Towards a Better Design of Online Marketplaces

by

Zhaohui Jiang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Business Administration)
in the University of Michigan
2020

Doctoral Committee:

Professor Damian Beil, Co-Chair
Assistant Professor Yan Huang, Co-Chair
Associate Professor Jun Li
Associate Professor Yesim Orhun
Associate Professor Cong Shi

Zhaohui Jiang

jiangzh@umich.edu

ORCID iD: 0000-0003-3354-5851

Acknowledgments

I am extremely grateful to my advisors for their guidance and support during my PhD— to Damian R. Beil for his support and trust at all times, and for being a role model to me as a caring teacher, a wise mentor, and a rigorous researcher; to Yan Huang for helping me grow as a researcher, for always believing in me, for fighting research challenges alongside me, and for enlightening me with her love and curiosity for knowledge and truth; to Jun Li for leading me to new explorations in research, and inspiring me to reach out to the world to connect research with industry practices. I also thank my thesis committee members – Yesim Orhun and Cong Shi, for their comments that significantly improved this dissertation. I would also like to give warm thanks to the faculty members at the Stephen M. Ross School of Business, in particular, Roman Kapuscinski, Stephen Leider, Joline Uichanco, and many others, for their endless support throughout my PhD. I am also grateful to my undergraduate thesis advisor, Xiaona Zheng for her inspiration and encouragement that made me apply for a PhD.

I cannot begin to express my thanks to Aravind Govindarajan, my cohort, my best friend, and my biggest supporter throughout PhD, who made rough days palatable and happy days memorable. I am extremely grateful to my seniors, especially Yixin (Iris) Wang and Fangyuan Ma, who are always there to guide me when I feel lost. I am so grateful for all the great moments I spent with my office friends—Eun Woo Kim, Rebecca Chae, and Frank Li, my roommates—Yuanyuan Chen and Qiansheng Hou; those fun moments are the greatest highlights in my PhD. I also want to thank and give my best wishes to my life-long friends, Siying Cao and Yu Zang—thank you for always be there for me. Last but not the least, I would like to thank my mother, Xiaojun Zheng, and father, Junshu Jiang, for their unconditional love and firm belief in me, which makes me who I am. I am also deeply grateful to my partner Jingyu Fan, who always understands and supports me, even through times that we spent apart from each other.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

ABSTRACT

Online markets are staggering in volume and variety. These online marketplaces are transforming lifestyles, expanding the boundaries of conventional businesses, and reshaping labor force structures. To fully realize their potential, online marketplaces must be designed carefully. However, this is a significant challenge. This dissertation studies individual behavior and interactions in online marketplaces, and examines how to enhance efficiency and outcomes of these online marketplaces by providing actionable operational policy recommendations.

An important question in the context of open-ended innovative service marketplaces is how to manage information when specifying design problems to achieve better outcomes. Chapter 1 investigates this problem in the context of online crowdsourcing contests where innovation seekers source innovative products (designs) from a crowd of competing solvers (designers). We propose and empirically test a theoretical model featuring different types of information in the problem specification (conceptual objectives, execution guidelines), and the corresponding impact on design processes and submission qualities. We find that, to maximize the best solution quality in crowdsourced design problems, seekers should always provide more execution guidelines, and only a moderate number of conceptual objectives.

Building on the same research setting, Chapter 2 looks into another important yet challenging problem—how the innovation seeker should provide interim performance feedback to the solvers in online service marketplaces where seekers and solvers can interact dynamically. In particular, we study whether and when the seeker should provide such interim performance feedback. We empirically examine these research questions

using a dataset from a crowdsourcing platform. We develop and estimate a dynamic structural model to understand contestants' behavior, compare alternative feedback policies using counter-factual simulations, and find providing feedback throughout the contest may not be optimal. The late feedback policy, i.e., providing feedback only in the second half of the contest, leads to a better overall contest outcome.

Moving to a wider application, Chapter 3 leverages consumer clickstream information in e-commerce marketplaces to help market organizers improve demand estimation and pricing decisions. These decisions can be challenging, as e-commerce marketplaces offer an astonishing variety of product choices and face extremely diversified consumer decision journeys. We provide a novel solution to these challenges by combining econometric and machine learning (Graphical Lasso) approaches, leveraging customer clickstream information to learn the product correlation network, and creating high-dimensional choice models that easily scale and allow for flexible substitution patterns. Our model offers better in- and out-of-sample demand forecasts and enhanced pricing recommendations in various synthetic datasets and in a real-world empirical setting.

CHAPTER 1

Introduction

Online markets are staggering in volume and variety: In 2019, $1.7 trillion was spent globally in major online marketplaces, ranging from specialized online labor (e.g., Topcoder and Mechanical Turk) to e-commerce (e.g., Amazon and Alibaba). By reducing transaction costs and facilitating better matches between supply and demand, these online marketplaces are transforming lifestyles, expanding the boundaries of conventional businesses, and reshaping labor force structures. To fully realize their potential, online marketplaces must be designed carefully. However, this is a significant challenge. Newly emerging online marketplaces create unprecedented market structures, these structures affect the decisions of both demand- and supply-side market participants, and these intricate and complex interactions in turn determine market outcomes.

To improve the design of online marketplaces, I use novel large-scale datasets (e.g., from crowdsourcing graphic design platforms and e-commerce marketplaces) and study individual behavior and interactions in these marketplaces. My goal is to improve marketplace designs by providing actionable operational policy recommendations to enhance the efficiency and outcomes of online marketplaces. In particular, I study the following problems in my dissertation: how seekers should optimally specify their "problem statement" at a crowdsourcing contest's outset (Chapter 1), how seekers should provide interim performance feedback to participants during the contest (Chapter 2), and how consumer clickstream information can help e-commerce marketplaces improve demand estimation and pricing decisions across a very large number of products (Chapter 3).

Online service marketplaces connect seekers and solvers. These online service marketplaces have been expanding from supplying basic labor to solving more open-ended problems, such as logo design, algorithm design, web design, etc. An important question in the context of open-ended innovative service marketplaces is how to manage information when specifying design problems to achieve better outcomes. In Chapter

1, I study this problem from both a theoretical and an empirical perspective, in the context of online crowdsourcing contests (also known as open innovation contests), where innovation seekers source innovative products (designs) from a crowd of competing solvers (designers). To understand how information impacts designer behavior in online marketplaces, I build a novel model to capture the design process based on qualitative theories in the design literature. To ensure that the observations and recommendations are realistic, I combine my game-theoretic model with econometric analysis and textual analysis, and I leverage a unique quasi-natural experiment to strengthen the causality arguments. I show theoretically and validate empirically that a very highly specified design problem statement may not be optimal, and that different types of information (e.g., conceptual objectives and executional guidance) may have significantly different impacts on designer behavior and should be managed differently. These findings can improve current marketplace practices, as the platforms now often encourage seekers to pack as much information as possible into their problem specification when describing the design they want. This is the first paper to study the role of different types of information in problem specifications in crowdsourcing contests, and the first to bring insights from the design research field into the study of crowdsourcing innovation contests.

Building on the same research setting as Chapter 1, I look into another important yet challenging problem— how the innovation seeker should provide interim performance feedback to the solvers (e.g., in the form of ratings) in online service marketplaces where seekers and solvers can interact dynamically. In Chapter 2, I study whether and when the seeker should provide such interim performance feedback. To answer this important research question, I take a structural modeling empirical approach in the context of graphic logo design contests. My structural empirical model features endogenous entries by new designers throughout the contest, exploitation and exploration behaviors by existing designers, and the designers' strategic choices among these decisions in a dynamic game. Using real-world contest data, I estimate the parameters governing designers' participation behavior. In particular, to distinguish between designers' exploratory and exploitative strategies, I use the computer vision Scale-Invariant Feature Transform algorithm to quantify the similarity between pairs of logo submissions. Based on these understandings of solver behavior, I conduct policy simulations to assess the contest outcomes under alternative feedback policies. My findings offer important insights for crowdsourcing platforms. In practice, most online platforms for crowdsourcing contests strongly encourage seekers to provide feedback throughout the contests. My findings underscore the value of providing feedback, but surprisingly they also suggest that the platform's current recommendation that "the more frequently the seeker provides feedback the better" may be misguided, depending on the seeker's objectives. If the seeker's objective is to maximize the number of high performers or the total number of designers in the contest, the late feedback policy (providing feedback only late in the contest) is the best option. This is one of the first studies to investigate the impact of feedback on the outcome of crowdsourcing contests, and the first study to propose a structural model to empirically analyze the impact.

Moving to a wider application, I investigate how information from online marketplaces can improve more conventional operational decisions. In Chapter 3, I leverage consumer clickstream information in e-commerce marketplaces to help market organizers improve demand estimation and pricing decisions. These decisions can be challenging, as e-commerce marketplaces offer an astonishing variety of product choices and face extremely diversified consumer decision journeys. I provide a novel solution to these challenges by combining econometric and machine learning (Graphical Lasso) approaches, leveraging customer clickstream information to learn the product correlation network, and creating high-dimensional choice models that easily scale and allow for flexible substitution patterns. Partnering with a leading international online retail marketplace organizer, I demonstrate significant value of incorporating clickstream information when making conventional operational decisions: Our model offers better in- and out-of-sample demand forecasts in various synthetic datasets in this real-world empirical setting. Based on these estimated substitution patterns, my model can provide better pricing recommendations for the marketplace organizer, boosting profit and revenue relative to current pricing practices. My model can be applied to different settings with large choice sets, and it can help with a variety of operational decisions, such as pricing, assortment, and inventory planning. This is one of the first studies to leverage clickstream data in choice models to help identify substitution patterns, and one of the first to introduce Graphical Lasso to the choice model framework.

In summary, my research generates new insights into how participants behave in and interact with online marketplaces, and it offers actionable policy recommendations for better designs of these online marketplaces.

CHAPTER 2

The Role of Problem Specification in Crowdsourcing Contests for Design Problems: A Theoretical and Empirical Analysis.

2.1 Introduction

Online crowdsourcing has become a popular channel for sourcing design (creative) products. A widely used form of organizing the crowdsourced innovation process is the *Crowdsourcing Contest*, used for products ranging from web to interior design. Compared with traditional innovation sourcing approaches, crowdsourcing contests allow seekers to access a large pool of designers, solicit a larger number of solutions from which to choose, and pay for only the most satisfying solutions. A typical crowdsourcing contest starts with a seeker specifying a design problem and associated award(s), based on which designers generate solutions, and compete for the award(s).

We focus on the seeker's problem specification: how the seeker specifies his design problem at the launch of his crowdsourcing contest. In the problem specification, a seeker can state his problem (e.g., he needs a logo for a real estate company), and communicate what he would like the solutions to achieve (e.g., the logo should convey professionalism and reliability; blue color and sharp edges are preferred). The information provided in the problem specification defines what constitutes a "high-quality" solution in the focal contest, which can potentially affect designer behavior and contest outcomes. Hence, it is important to understand the role of problem specifications in crowdsourcing contests.

Yet, the best approach to problem specification is not obvious. At first sight, one may think seekers should specify their problems in the most thorough manner possible. A detailed problem specification clarifies what the seeker is looking for. Without it, designers may miss important points when generating solutions, and they may also face increased uncertainty about how closely their solutions match the seeker's objectives. However, an overly specified problem may backfire, especially when the seeker is not careful about the types of information he provides. By taking a closer look at seekers' problem specifications, we find that problem specifications can contain multiple types of information, which are likely to affect designers' behavior differently. For example, a problem specification with a long list of objectives can overwhelm designers. Designers have to spend more time digesting the list, clarifying design objectives, and creating designs satisfying multiple objectives; consequently, designers may choose to not incorporate all of the design objectives, or even choose not to participate.

In this paper, we aim to address the following research questions: (1) provided with different types of information in a seeker's problem specification, how do designers decide whether to join the contest or not, and if so, how do they reflect information from the problem specification in their design solutions; and (2) how should seekers optimally specify their design problems?

To answer these questions, we first construct a game-theoretical model to capture designers' behavior given a problem specification. Our model distinguishes different types of information provided in problem specifications ("professionalism" and "reliability" can be considered as *conceptual objectives* that the seeker wants design solutions to achieve, whereas "blue color" and "sharp edges" as *execution guidance* that conveys the seeker's instructions for design details). To assess how such information influence decisions in designers' design processes, our model features distinct stages in design processes (i.e., *design problem framing*, *design concept formulating* and *design trials generating*), which are based upon an established framework in the qualitative *design research* literature (e.g., Schön (1984, 1988); Cross (2011)). Our theoretical model predicts that the number of participating designers in a contest eventually decreases with more conceptual objectives disclosed in the problem specification. In addition, participants' trial effort provision increases with more execution guidelines provided in the problem specification. We also offer insights on how seekers can optimize their problem specifications to maximize the quality of the best design: more execution guidance is always beneficial; however, seekers should not always disclose all their conceptual objectives.

The theoretical predictions of our model are empirically tested against a dataset of logo design contests from a major crowdsourcing platform. In addition, we avail ourselves of a recent "quasi-natural experiment" opportunity that arose on the platform, wherein changes were made to the platform problem specification template. This analysis further strengthens the reliability of our empirical results, directionally confirms our recommendations for the provision of information in problem specifications, and generates additional nuanced insights into the implementation of these recommendations.

Our study makes several contributions. First, it is one of the first papers studying the role of problem specifications in crowdsourcing contests, which holds great practical relevance. Second, we theoretically and empirically distinguish between different types of information contained in problem specifications, i.e., conceptual objectives and execution guidelines, and highlight their differential effects on designers' participation behavior and solution quality. Third, we bring findings from the *design research* field into the study of crowdsourcing innovation contests. In particular, we borrow from the qualitative *design research* literature, and formulate a mathematical model to capture three distinct stages in design processes. The incorporation of the first two stages, the *design problem framing* and *design concept formulating* stages, which are often omitted in theoretical models of crowdsourcing contests, is crucial for capturing a more complete picture of the designers' design process and the impact of information provided in problem specification on this process. Finally, we offer empirical evidence from the field to support the predictions and recommendations from our theoretical model, linking the theoretical and empirical research of crowdsourcing innovation contests.

2.2 Literature Review

Crowdsourcing contests have attracted interest from operations management researchers. Existing operations literature on the design of crowdsourcing contests has looked at the impact of award structure, contest size, open/closed entry, contest duration, and the presence of competing contests on the outcome of crowdsourcing contests. (See Chen et al. (2018) for a review of this literature.) There is also an emerging literature examining the role of *information* in crowdsourcing contests. A few recent studies investigate the impact of the disclosure of intermediate solutions (Boudreau and Lakhani, 2015; Wooten and Ulrich, 2017a; Bockstedt et al., 2016) and interim feedback (Jiang et al., 2016; Wooten and Ulrich, 2017b; Bimpikis et al., 2017; Mihm and Schlapp, 2018) on contest dynamics and outcomes. The present paper and Jiang et al. (2016) employ similar data from a crowdsourcing logo design platform. Jiang et al. (2016) utilizes a structural modeling approach to capture the dynamics of feedback during a contest, while the present paper introduces an analytical model motivated by the design literature, to mathematically frame the design process and incorporate it into an analytical exploration of innovation contests, and then empirically examines the analytical predictions and policy recommendations so derived; the research questions, methodology, contribution and insights are quite different across the two papers.

Another important occasion where seekers disclose information to solvers is *problem specification.* The information in a problem specification is a major part of what "defines" the contest, and as such can have a significant impact on participants' behavior and contest outcomes. Besides the present paper, to the best of our knowledge the only other study that looks at problem specification in crowdsourcing contests is Erat and Krishnan (2012). In that paper, the authors study contests where the seeker starts with a well-defined problem, and solvers choose from a set of known approaches. They focus on how the completeness of problem specifications helps provide a more precise valuation of those approaches, which narrows down the solvers' search by revealing which of the known solutions are more likely to be successful. By contrast, we study open-ended creative contests, where the seeker describes the problem, the "approach(es)" to solving it are generated by each expert designer, and a longer list of specifications may make it more difficult to find a suitable approach. Moreover, we study how different types of information disclosed in the problem specification affect designers' entry, design concept formation and design solution generation behavior. Not surprisingly, with our different model/setting and research focus, we arrive at different managerial insights. Erat and Krishnan (2012) find that the seeker may not want to fully specify their problem, in order to increase ambiguity that in turn increases the breadth of search that the solvers undertake within a set of known solution approaches. By contrast, we model two types of information, and show that the seeker always wants divulge all his execution guidelines, but might not want to divulge all his conceptual objectives because an overly long set of objectives may discourage creators from participating. Furthermore, we use real-world data to empirically test the predictions of our theoretical model, which not only helps ensure the validity of the theoretical model, but contributes to the empirical literature of crowdsourcing contests.

An innovation in our theoretical model of crowdsourcing contests is that our model borrows from the classic literature in *design research* and explicitly captures various stages of designers' solution generation process. The design research literature (e.g., French et al. (1985); Pahl and Beitz (1988); Hubka (1989); Roozenburg and Cross (1991)) often portrays the design process as a sequence of activities, which can be grouped into phases of *design problem framing* (clarifying objectives), *design concept formulating* (generating and refining design concepts), and *design trial generating* (embodying designs and detailing designs). The last phase, which corresponds to the stage where a designer generates actual solutions, is often the focus of analytical models of crowdsourcing contests in the operations and economics literature. The first two phases are often overlooked, possibly because they are less visible and more abstract. Researchers in design research realize the importance of these two phases in the design process, and call for attention to them (Schön, 1984, 1988; Pahl and Beitz, 1988; Cross, 2011). For example, Schön (1988) suggests that design problems are often "ill-defined", in that "in a design project it is often not at all clear what 'the problem' is"; hence, in order to solve those problems, "the designer must frame a problematic design situation", in which "the goal is set at a high level with clear objectives". Cross (2011) continues to stress the importance of the *design concept formulating* stage: "a clear concept of how to reach this goal is devised, ... and the solution details then cascade from the concept". In this stage, "designers select features of the problem space to which they choose to attend, and identify areas of the solution space in which they choose to explore" (Cross, 2001). The model to be presented in the next section reflects all these important stages of the design process, and captures how information in the problem specification influence designers' decisions in each of these stages.

2.3 Theoretical Model

In this section, we construct a theoretical model that characterizes seekers' and designers' decisions in a crowdsourcing design contest. Consider a situation where a seeker ("he") wishes to source solutions to a design problem from a group of designers through a crowdsourcing contest. The seeker has some *conceptual objectives* in mind (e.g., the design should convey reliability and helpfulness); each conceptual objective included in a design gives an equal, incremental quality w. (In Online Appendix EC.7.2 we consider an alternative model in which the conceptual objectives have diminishing weights.) Apart from the conceptual objectives, the seeker can also provide *execution guidelines* (e.g. what color or shape is/is not desired, etc.). Unlike conceptual objectives which the designer has to interpret, execution guidelines are more straightforward — e.g., "don't use the color red". (See Online Appendix EC.1 for several examples of conceptual objectives and execution guidelines in our data.) The sets of all the conceptual objectives and execution guidelines the seeker has in mind are denoted as $\overline{\mathbb{S}}_r$ and $\overline{\mathbb{S}}_g$ respectively, with the size of the two sets being $|\overline{\mathbb{S}}_r| := \overline{S}_r$ and $|\overline{\mathbb{S}}_g| := \overline{S}_g$. Given $\overline{\mathbb{S}}_r$ and $\overline{\mathbb{S}}_g$, the seeker decides which conceptual objectives and execution guidelines to disclose in his problem specification. (Note that we do not study how seekers come up with $\overline{\mathbb{S}}_r$ and $\overline{\mathbb{S}}_g$ in the first place.) The sequence of events in this contest is as follows:

- The seeker posts his design request, in which he announces the award amount (A) for the contest winner (we consider "single-winner" contests as those are the most common type of contests in our empirical setting), and specifies his design problem. In this problem specification, the seeker can specify some or all of his conceptual objectives and execution guidelines. The sets of disclosed conceptual objectives and execution guidelines are denoted as $\mathbb{S}_r$ ($\subseteq \overline{\mathbb{S}}_r$) and $\mathbb{S}_g$ ($\subseteq \overline{\mathbb{S}}_g$) respectively, with the sizes of the two sets being S_r ($\leq \overline{S}_r$) and S_g ($\leq \overline{S}_g$).

- Given the design request, designers ("she") first decide whether to enter the contest or not, based on their assessment of the expected net payoff they will receive if they join the contest. Those who decide to enter the contest then go through a *design process* (to be explained in Section 2.3.1) to develop their design submissions, and submit them to the seeker.

- Finally, the seeker evaluates all the submitted designs, claims the best-quality design among those submissions, and gives the award to the designer of the winning design.

As is common in the crowdsourcing literature (Terwiesch and Xu, 2008; Erat and Krishnan, 2012; Körpeoğlu et al., 2017), we model designers simultaneously making participation and other design decisions. Below we present a mathematical model reflecting a typical *design process*, drawing on the design research literature. Using this model, we analyze designers' behavior in Section 2.3.2.

2.3.1 Designers' Three-Stage Design Process

In the design literature, design processes are often considered to consist of several cognitive steps, which can be broadly classified into three stages: *design problem framing*, *design concept formulating* and *design trial generating* (van den Kroonenberg, 1986; Cross, 2001, 2011). To mathematically represent these three distinct stages, we formulate a stylized model, with simple functional forms, that tractably captures the key features of our setting; simplification by assuming functional forms, to ensure tractability, is an approach widely used in previous theoretical research on crowdsourcing or open innovation contests (Ales et al., 2017b; Korpeoglu et al., 2017; Mihm and Schlapp, 2018). Next we provide modeling details for each of the three stages in the design process.

Design Stage (I) — Framing the Design Problem. Design problems are nearly always "not all clear" and "may have been only loosely defined by the client (seeker)"

(Cross (2001) p.81). Hence, a key aspect of the design process lies in digesting and understanding the conceptual objectives ($\mathbb{S}_r$) in seekers' problem specifications. This stage is referred to as the *design problem framing* stage.

Framing the design problem is effort-consuming (Cross, 2001). In reality, conceptual objectives in a problem statement are often embedded in sentences or paragraphs, as seekers endeavor to communicate what they are looking for in a design. Designers need to exert efforts to understand the problem statement text and extract and comprehend the conceptual objectives conveyed. In general, the more conceptual objectives ($\mathbb{S}_r$) are embedded in the problem specification, the higher effort cost a solver has to incur in the design problem framing stage. We model this cost as $c_1 S_r$. On the other hand, when problem specifications contain too little information about the seeker's conceptual objectives, solvers need to incur a different form of *problem framing* cost – they may need to "guess" at the seeker's possible conceptual objectives. We assume that, to guess each conceptual objective, the solver incurs a cost c_g, and this unit cost of "guessing" possible objectives is higher than the cost of "understanding" disclosed objective ($c_g > c_1$). We also assume that the marginal quality improvement by incorporating a "guessed" conceptual objective is not as high as a disclosed one, represented by νw where $\nu \in (0, 1)$. Note that we assume the problem framing cost changes only with conceptual objectives but *not* with execution guidelines in problem specifications, as execution guidelines are mostly objective instructions in standardized design terms, and are therefore more straightforward for designers to understand. (Empirical evidence for this assumption is provided in Online Appendix EC.5.)

Design Stage (II) — Formulating the Design Concept. After framing the problem, participating designers "select features of the problem space to which they choose to attend", and then "identify areas in the solution space where they choose to explore" (Cross, 2001). We form a mathematical model for these two steps: (1) designer i chooses $\mathbb{D}_{r,i}$ to incorporate into her design(s), with the number of incorporated objectives being $r_i := |\mathbb{D}_{r,i}|$; (2) designer i searches for a design concept satisfying all conceptual objectives in $\mathbb{D}_{r,i}$.

We model the cost associated with step (2) as follows. Consider a potential design concept to be a "sample" (random draw). The probability that a sampled design concept satisfies any particular conceptual objective is $p \in [0, 1]$. Assuming the objectives are independent (we consider an extension capturing the level of overlap across objectives in Online Appendix EC.7.1), the probability of a sampled design concept being "successful", i.e., satisfying all r_i targeted objectives, is p^{r_i}. Hence, in expectation, designer i has to attempt $(\frac{1}{p})^{r_i}$ design concepts until she finds a "successful" one. If the cost associated with each attempt is c_2, the expected cost of the *design concept formulating* stage is $c_2(\frac{1}{p})^{r_i}$. Note that this cost increases exponentially with the number of objectives designer i incorporates (r_i), which captures the fact that it gets increasingly more challenging to find a design concept that *simultaneously* satisfies more objectives.

Design Stage (III) — Generating Design Trials. In the final stage, based on the design concept identified in Stage II, designers generate design trials, which are submitted to the seeker. (Hereafter, we use submissions, solutions, and trials interchangeably.) We assume designers incur a cost of c_3 to come up with a design trial — in this stage, designers need to figure out the execution details, such as shape, color, font, etc., which is effort-consuming. If the seeker provides execution guidelines, (i.e., recommends designers what fonts, colors, shapes, etc. to use), it will save designers' time and effort in determining such details. Hence, we expect the cost of each design trial (c_3) decreases with more execution guidelines (S_g). Correspondingly, in our model, we assume $c_3 = h(S_g)$, where $h(\cdot)$ is a decreasing function. (We provide empirical evidence for this assumption in Section 2.5.) Given c_3, designer i who decides to generate m_i design trials incurs a cost of $c_3 m_i$ in the *design trial generating* stage.

For focal designer i, the quality of each trial τ $(= 1, 2, ..., m_i)$, denoted as $V_{i\tau}$, is assumed to be the baseline value of designer i's design concept (v_i), plus a quality random shock ($\epsilon_{i\tau}$) (i.e., $V_{i\tau} = v_i + \epsilon_{i\tau}$). The baseline quality of designer i's design concept (v_i) is the sum of weights associated with all its satisfied conceptual objectives. The uncertainty captured by $\epsilon_{i\tau}$ may come from seeker taste uncertainty (the perceived quality is often subject to the taste of the seeker) and trial quality shock (the uncertainty associated with the execution of the design concept). Like Terwiesch and Xu (2008), we model trial shocks, $\epsilon_{i\tau}$'s, as Gumbel distributed with mean zero and scale parameter μ, *i.i.d* across design trials.

2.3.2 Designers' Problem

Combining the three design stages discussed above, when participating designer i incorporates r_i conceptual objectives and generates m_i design trials, her overall expected cost is:

$$C_i(r_i, m_i) = \underbrace{c_g \cdot (r_i - S_r)^+ + c_1 \cdot S_r}_{\text{Design Stage (I)}} + \underbrace{c_2 \cdot (1/p)^{r_i}}_{\text{Design Stage (II)}} + \underbrace{c_3 \cdot m_i}_{\text{Design Stage (III)}}, \text{ where } c_3 = h(S_g); \tag{2.1}$$

and the quality of designer i's best design is:

$$V_i(r_i, m_i) = \max_{\tau=1,\ldots,m_i} V_{i\tau} = \begin{cases} r_i w + \max\limits_{\tau=1,\ldots,m_i} \epsilon_{i\tau}, & r_i \leq S_r \text{ (without guessing);} \\ S_r w + (r_i - S_r)\,\nu w + \max\limits_{\tau=1,\ldots,m_i} \epsilon_{i\tau}, & r_i > S_r \text{ (with guessing).} \end{cases} \tag{2.2}$$

We are modeling a single-winner contest; therefore, if a designer submits multiple designs, only her highest-quality design matters. Note that because guessing is costly, designer i would only guess the minimum number of concepts needed to reach r_i, i.e., $(r_i - S_r)^+$. Note that we do not explicitly consider designers' choices on the amount of execution guidance to follow, because this decision is trivial: designers would always follow all the execution guidance to lower their design trial generation cost. This is captured by setting $c_3 = h(S_g)$.

We now analyze designers' entry decision, design concept formation and trial effort provision in a crowdsourcing contest. Consider a focal designer i facing a contest with award A, S_r conceptual objectives, and S_g execution guidelines. Let $j = 1, ..., N(\neq i)$ index the other designers who would be i's opponents participating in the contest, where r_j is the number of conceptual objectives j will incorporate and m_j is the number of design trials j will generate. Focal designer i makes the following decisions: whether to join the contest or not (i.e., a binary entry decision, denoted by d_i), how many conceptual objectives to incorporate in the *design concept formulating* stage (i.e., a concept formation decision, denoted by r_i), how many trials to generate in the *design trial generating* stage (i.e., a trial effort decision, denoted by m_i). Designer i makes those decisions to maximize her expected utility (her expected compensation minus her expect costs):

$$\max_{d_i, r_i, m_i} U_i(d_i, r_i, m_i) = \mathbb{I}_{d_i=1} \cdot [\Pr(\text{i wins}) \cdot A - C_i(r_i, m_i)] + \mathbb{I}_{d_i=0} \cdot s,$$
$$\text{where } \Pr(\text{i wins}) = \Pr(V_i > V_{j|j=1,\ldots,N(\neq i)}) = \Pr(V_i(r_i, m_i) > \max\{V_{j|j=1,\ldots,N(\neq i)}(r_j, m_j)\}). \tag{2.3}$$

If designer i decides not to join ($d_i = 0$), she earns utility s from choosing her outside option. (In other words, we consider s to be the opportunity cost of joining a contest.) If designer i's best design provides the highest value to the seeker (i.e. $V_i > V_j, \forall j \neq i$), she wins the contest and receives the award A; otherwise, she does not receive anything.

2.3.3 Designers' Equilibrium Behavior

Utilizing Equation (2.3), we solve for designers' equilibrium behavior. As is common in the crowdsourcing literature (Terwiesch and Xu, 2008; Erat and Krishnan, 2012;

Körpeoğlu et al., 2017; Ales et al., 2017b), we focus on symmetric pure strategy Nash equilibrium throughout the paper. In our analysis, we assume that the number of potential participants is sufficiently large that participants will enter the contest as long as it is profitable to do so. That is, the size of a contest is never limited by a lack of potential participants. (This assumption is natural in crowdsourcing contests — for example, in our dataset, on any given day, the average number of active designers (i.e., potential participants) on the platform is around 300, while the average number of participants in a contest is around 26; see Table 2.1.) Theorem 2.3.1 characterizes the equilibrium number of participants (N^*) in a crowdsourcing contest, the equilibrium number of objectives each participating designer incorporates (r^*), and the equilibrium number of design trials each participating designer generates (m^*). See Appendix A.1 for the proof. For simplicity, in our analysis we allow N^*, r^*, and m^* to be non-integer numbers; and to avoid trivial solutions, we confine our attention to problem parameters for which can assume $N^* > 1$, $m^* > 0$, seekers wish to disclose at least some conceptual objectives (i.e., w is not trivially small), and the seeker is able to induce designers to incorporate all disclosed conceptual objectives by disclosing sufficiently few conceptual objectives in the problem statement (i.e., $r^* \geq S_r$ if S_r is sufficiently small).

Theorem 2.3.1 *In a crowdsourcing contest, where the equilibrium number of participating designers equals N^*, the unique symmetric equilibrium for r^* and m^* are as follows. The equilibrium number of objectives a designer incorporates is $r^* = \begin{cases} \overline{r}^g & \text{if } S_r < \overline{r}^g \\ S_r & \text{if } \overline{r}^g \leq S_r \leq \overline{r} \\ \overline{r} & \text{if } S_r > \overline{r} \end{cases}$, where $\overline{r}^g = \frac{\ln\left(\frac{N^*-1}{(N^*)^2}\frac{\nu w}{\mu}\frac{A}{c_2}\frac{1}{\ln(1/p)} - \frac{c_g}{c_2}\frac{1}{\ln(1/p)}\right)}{\ln(1/p)}$ and $\overline{r} = \frac{\ln\left(\frac{N^*-1}{(N^*)^2}\frac{w}{\mu}\frac{A}{c_2}\frac{1}{\ln(1/p)}\right)}{\ln(1/p)}$; and the equilibrium number of design trials each designer generates is $m^* = \frac{A(N^*-1)}{(N^*)^2 c_3}$, where $c_3 = h(S_g)$.*

The equilibrium number of participating designers (N^*) first increases (when $S_r < \overline{r}^g$) and then decreases (when $S_r \geq \overline{r}^g$) with more disclosed conceptual objectives S_r; but N^* does not change with the amount of execution guidance S_g. (The exact formula for N^* is provided in Appendix A.1.)

Theorem 2.3.1 reveals that when there are very few conceptual objectives disclosed ($S_r < \overline{r}^g$), designers find it optimal to guess and incorporate additional conceptual objectives beyond those that have been disclosed, i.e., $r^* = \overline{r}^g$ ($> S_r$). As more conceptual objectives are disclosed ($\overline{r} \leq S_r \leq \overline{r}^g$), designers are going to incorporate all and only the disclosed conceptual objectives, i.e., $r^* = S_r$. As the number of disclosed conceptual objectives further increases ($S_r > \overline{r}$), designers are no longer willing to incorporate all the disclosed objectives; they only incorporate a subset of the disclosed objectives, i.e., $r^* = \overline{r}$ ($< S_r$).

Designers' equilibrium trial effort (m^*) increases with more execution guidance (larger S_g), because (per Section 2.3.1) this lowers the designers' cost of generating each design trial (c_3). Since designers tailor their equilibrium number of design trials (m^*) to the size of c_3 (or associated S_g), the size of c_3 (S_g) ends up not affecting the designers' entry decision.

Theorem 2.3.1 also characterizes the equilibrium number of participants. When there are few conceptual objectives disclosed ($S_r < \bar{r}^g$), designers are willing to "guess" and incorporate undisclosed objectives. In this case where designers are "guessing", as S_r increases, more designers will join a contest. The reason is that "guessing" is relatively more effort-consuming than "comprehending" ($c_g > c_1$), and more disclosed conceptual objectives save designers from guessing some conceptual objectives. However, when there are enough conceptual objectives disclosed ($S_r \geq \bar{r}^g$), designers no longer "guess" but instead just incorporate all or some the disclosed conceptual objectives. In these cases, the relationship between S_r and N^* reverses – fewer designers will join a contest as the seeker discloses even more conceptual objectives. The intuition is as follows. With more disclosed conceptual objectives, designers have to spend more time comprehending them, and more effort searching for design concepts that satisfy those objectives simultaneously, which leads to a higher participation cost (i.e., the total cost a designer incurs throughout the three stages of the design process). This is true even in cases where designers choose to only select a subset of the disclosed objectives to incorporate (i.e., the number of disclosed objectives S_r is larger than the number of objectives designers are willing to incorporate $\bar{r}$), because although designers' design concept formation cost does not increase when S_r exceeds $\bar{r}$, they would still need to spend more time understanding and digesting all disclosed objectives to frame the design problem at the beginning of the design process. Thus, the participation cost always increases when more conceptual objectives are disclosed, which leads to a lower equilibrium expected profit under the same level of competition, and correspondingly a smaller number of participants in equilibrium.

2.3.4 Seeker's Problem

In this section we analyze how seekers should provide information in their problem specification to maximize their "profit" in crowdsourcing design contests. A seeker's profit (Π_s) is defined as the expected highest quality among all the designs submitted to his contest, i.e., $\Pi_s = \mathbb{E}_{\boldsymbol{\epsilon}} \max_i V_i^*$, where V_i^* is the equilibrium quality of designer i's best design, and the expectation is taken over the vector of all participating designers' design trials' quality shocks $\boldsymbol{\epsilon}$. (For simplicity, the "profit" ignores the cost of the award A, as we are focusing on how the seeker optimizes the problem specification given an award size A.) The seeker maximizes Π_s by choosing *the number of conceptual objectives to disclose (S_r)* and *the amount of execution guidance to provide (S_g)*. We define the seeker's problem as:

$$\max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \Pi_s(S_r, S_g) = \max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \left[\mathbb{E}_{\boldsymbol{\epsilon}} \max_{i=1 \ldots N^*} V_i^*(r^*, m^*) \right], \tag{2.4}$$

where N^* is the equilibrium number of participating designers, and r^* and m^* are the equilibrium design concept formation and trial generation strategies of participating designers. Problem (2.4) is a joint problem in both S_r and S_g. From Theorem 2.3.1, we know the impacts from S_r and S_g can be separated: N^* and r^* are only affected by S_r but not by S_g; and given N^*, m^* is only affected by S_g but not by S_r. So, we can rewrite Problem (2.4) as the following problem separable in S_r and S_g (see Appendix A.2 for the proof):

Lemma 2.3.1

$$\max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \Pi_s(S_r, S_g) = \max_{S_r \le \overline{S}_r} \left[w \cdot r^*(S_r) + \mu \ln A^{\frac{N^*(S_r)-1)}{N^*(S_r)}} \right] + \max_{S_g \le \overline{S}_g} \left[-\mu \ln(h(S_g)) \right]. \tag{2.5}$$

With the seeker's decisions on S_r and S_g being separable (Problem (2.5)), we next discuss how the seeker should set S_r and S_g separately.

Seeker's Decision on Number of Conceptual Objectives to Disclose. We first consider the problem associated with the seeker's optimal choice of S_r, the number of conceptual objectives to disclose in the problem specification. The seeker's objective is to choose a S_r that maximizes the expected best design quality under any fixed S_g, i.e., $\max_{S_r \le \overline{S}_r} \Pi_s(S_r; S_g)$.

Proposition 2.3.1 $\Pi_s(S_r; S_g)$ *decreases in S_r once S_r becomes sufficiently large, and thus for sufficiently large $\bar{S}_r$, we have $S_r^* < \bar{S}_r$. Proof is provided in Appendix A.3.*

The seeker's profit Π_s is first increasing and eventually decreasing with S_r. The intuition is as follows. Disclosing more conceptual objectives has two countervailing effects. On the one hand, with more conceptual objectives being disclosed, designers are aware of and thus can incorporate more objectives, which leads to a higher expected quality of each design generated, positively affecting the best design quality $\Pi_s(S_r)$. We call this effect the "*quality effect*". On the other hand, with more disclosed objectives, the designers' participation cost increases (higher costs in the *design problem framing* and *design concept formulating* stages), leading to fewer designers participating in the contest. We call this the *"competition effect"*. This *competition effect* negatively affects the best design quality $\Pi_s(S_r)$, because $\Pi_s(S_r)$ is an extreme value of qualities of all design submissions, which will decrease if there are fewer participating designers.

As the seeker discloses more and more conceptual objectives (a larger S_r), the negative *competition effect* increases and eventually dominates the *quality effect*, because with only a handful of designers in the contest, a small decrease in the number of participating designers would have a severe impact on the extreme value $\Pi_s(S_r)$. As a result, $\Pi_s(S_r)$ will decrease when the seeker discloses too many conceptual objectives. Therefore, the seeker should not always disclose all the conceptual objectives he cares about. For reasonable parameter ranges, simulation results suggest that the optimal S_r^* is relatively small (from 2 to 5); see Appendix A.3.

Seeker's Decision on Number of Execution Guidelines to Provide. Next, we consider the seeker's problem of choosing a S_g that maximizes the expected best design quality under any fixed S_r, i.e., $\max_{S_g \leq \overline{S}_g} \Pi_s(S_g; S_r)$. This problem is trivial, since providing more execution guidelines (S_g) always increases best design quality. Intuitively, as the seeker provides more execution guidelines (a larger S_g), it is easier for designers to come up with design trials (a lower trial cost c_3), because the execution guidelines give directions for different aspects of design execution (such as color, shape, etc.), which designers otherwise need to spend effort figuring out and deciding on. Hence, with the lowered cost to generate each design trial, designers will come up with more trials (a larger m^*), which then in turn leads to a higher extreme value in submission quality (a higher Π_s). Therefore, the seeker should disclose all his execution guidelines (i.e., $S_g^* = \overline{S}_g$).

2.3.5 Summary of Key Takeaways and Model Extensions

Takeaway 1: The number of participating designers first increases and then decreases with the number of conceptual objectives disclosed in the problem specification; it does not change with the amount of execution guidance in the problem specification.

Takeaway 2: Given the equilibrium number of participating designers, more execution guidance leads to a higher level of designer trial effort provision from each participating designer. However, more conceptual objectives will not affect designers' trial effort provision.

Takeaway 3: The seeker should disclose as much execution guidance as possible, but only a moderate number of conceptual objectives.

We keep our main model parsimonious and focus on capturing a complete picture of designers' design process. The results from this parsimonious model are robust to alternative modeling assumptions and are supported by empirical data. We formally derive theoretical results for two extensions, which we have alluded to in Section 2.3.1.

(Details are in Online Appendix EC.7.) First, we consider overlaps among conceptual objectives (e.g., "friendly" and "welcoming" overlap more than "friendly" and "professional" do). Second, we allow for diminishing weight/importance among conceptual objectives (i.e., objectives are sorted in descending order in importance). In both extensions, the qualitative findings, i.e., Takeaways 1, 2 and 3, remain intact.

2.4 Hypotheses

Based on the theoretical predictions (summarized in Section 2.3.5), we develop testable hypotheses.

- **Hypothesis 1a**: *When the number of disclosed conceptual objectives (S_r) is very small, the number of participants (N^*) increases with S_r.*
- **Hypothesis 1b**: *When the number of disclosed conceptual objectives (S_r) is moderate or large, the number of participants (N^*) decreases with the number of conceptual objectives (S_r) specified in the problem specification.*
- **Hypothesis 1c**: *The number of participants (N^*) does not change with the amount of execution guidance (S_g) provided in the problem specification.*
- **Hypothesis 2a**: *Given the number of participants (N^*), more execution guidance (S_g) leads to more trial effort provision (m^*) from each participating designer.*
- **Hypothesis 2b**: *Given the number of participants (N^*), the number of conceptual objectives (S_r) does not affect the trial effort provision from each participating designer (m^*).*

We also wish to develop hypotheses related to our theoretical results on the relationship between problem specification and seeker profit – i.e., the best design quality. Yet, design quality is difficult to measure empirically for the following reasons. First, in design contests, the notion of quality is subjective – it is based on the matching between designers' submissions and seekers' private taste. Second, the quality of each design is not directly observable in our empirical setting. As we will explain in Section 2.5.5, seekers signal design quality using star ratings (on a scale of 1 to 5 star), however one still cannot perfectly infer the best design quality because (1) the ratings are truncated at 5-star, and quality differences among 5-star designs are not observed;

(2) the ratings are not completely comparable across different contests, because some seekers might be more strict in giving high ratings than others.

To proceed, we formulate hypotheses in terms of the number of submissions rated as 5-star by the seeker (instead of seekers' profit). Ignoring the in-comparability issue of ratings across contests mentioned above, a contest that receives more 5-star designs is likely to have a higher best design quality. Though not conclusive, testing this set of hypotheses can provide data evidence on our theoretical results for the seeker's problem.

- **Hypothesis 3a**: *The number of designs rated as 5-star eventually decreases with the number of conceptual objectives (S_r).*
- **Hypothesis 3b**: *The number of designs rated as 5-star increases with the amount of execution guidance (S_g).*

2.5 Empirical Analyses

2.5.1 Empirical Context and Data Description

The dataset we use for the empirical analysis is collected from a major online platform for crowdsourcing creative services in various areas, such as custom logo design, Web design, and writing services. We focus on logo design contests because it is a representative form of open-ended creative contests; it is also the largest category on the platform both in terms of the number of completed contests and the number of designers participating in the category.

A typical logo design contest on this platform proceeds as follows. First, a seeker in need of a design posts a design request, where he specifies the design problem by answering five questions:

- Q1: What is the exact name you would like in your logo?
- Q2: What is your industry?
- Q3: What are the top 3 things you would like to communicate through your logo?
- Q4: What logo styles do you like (image + text, image only, text only, etc.)?
- Q5: Do you have any other info or links you want to share?

The seeker also announces the award structure (e.g., the number of winners, and the award(s) for the winner(s)). Based on the seeker specified information, designers on the platform can join the contest and submit design(s). Finally, the seeker picks his favorite submission(s) and gives the pre-announced award to its (their) author(s). Note that, although the platform asks the seeker to list the "top 3 things" (in Q3), there is no hard limit on seekers' answers to this question. As can be seen from the illustrative example in Figure 3.1a on page 40 (and the additional examples in Online Appendix EC.1), some seekers specify more than "3 things" in their problem specification. The summary statistics to be shown in Table 2.1 reveal substantial variation in the length of seekers' answers to Q3 across different contests.

The main advantages of our data include the fact that (i) different types of information, namely, conceptual objectives (Q3) and execution guidelines (Q4 and Q5) are already separated by questions in a problem specification (the categorization is self-explanatory with how the questions are raised); (ii) the number of submissions made by each designer is available, which allows us to quantify the designers' trial efforts (often-unobserved in other contexts); (iii) the exact problem specification (textual information) provided in each contest is available, from which we can extract conceptual objectives and execution guidelines using either manual coding or natural language processing.

We collect data of "public-gallery" logo-design contests on this crowdsourcing platform from March, 2012 to November, 2014. For each contest, we record the seeker's problem specification and participating designers' submission activities. To facilitate the empirical analysis, we focus on 7-day contests where the design seekers promise to award $200 to one and only one final winner. This is because it has been documented that the contest length and award structure can affect designers' behavior and contest outcomes; since the objective of this study is to examine the effect of problem specification and to find the optimal way to specify a design problem, we purposefully minimize the heterogeneity among the contests in these other dimensions. The contests included in our sample are representative contests on the platform — 97% of the contests held on the platform have a single award, 61% are "assured", $200 is the most common award level, and 7-day is the most common length among all contests. The final working sample consists of 441 contests and 11,757 contest-designer combinations.

In the main empirical analyses, the measures for the amounts of different types of information provided in the problem specification are constructed using textual analysis and manual coding. Specifically, we use textual analysis to count the keywords related to each aspect of execution guidance (e.g., colors, fonts, usages, shapes, and styles) in the problem specification. For example, we count words such as "green", "round", "shining", etc., and use the total number of such keywords $(No.GuideWords)_q$ as the measure for the amount of execution guidance. However, it is challenging to apply this approach (textual analysis based on keywords) to extract conceptual objectives, because conceptual objectives are much less structured and involve a wide range of concepts, which are often embedded in sentences and sometimes implicitly mentioned. Hence, we hire 3 coders with design backgrounds, and ask each of them to independently read the problem specifications in our dataset and manually list down the conceptual objectives mentioned in the problem specification for each contest. (The complete coding instructions and resulting data are available upon request.) By then averaging the number of conceptual objectives identified across coders, we arrive at our measure for the number of conceptual objectives ($(No.Concepts)_q$) for each problem specification. The average inter-rater reliability (assessed using Weighted Kappa) is 0.91, which is considered very good, indicating strong agreement among different coders' assessments (Cohen, 1968). To make sure that our empirical results are robust and are not significantly affected by potential human or machine coding errors, we use the word-count of the seeker's answer(s) to Q3 (Q4 and Q5) as an alternative measure for the amount of conceptual objectives (execution guidance), and then as a robustness check we test our hypotheses using use these alternate measures (see Section 2.5.3 for details). Table 2.1 reports summary statistics of these contest-level characteristics.

Table 2.1 also reports summary statistics for variables characterizing the following: designers' behavior, including their entry ($No.Designers$) and effort ($Avg.SubPerDesigner$) decisions in each contest, the outcome of each contest, including the numbers of 5-star submissions and design styles ($(No.\ 5\text{-}Star\ Submission)_q$ and $(No.\ 5\text{-}Star\ DesignStyle)_q$) (these two variables are explained in Section 2.5.5), and the percentage of submissions rated by the seeker ($(Rated\ Ratio)_q$). We observe considerable variation in these variables across different contests. Our empirical analysis explores the relationship between characteristics of problem specifications and outcomes of the contest (including designers' participation and seekers' profit).

2.5.2 Empirical Evidence for Designer Behavior (Hypotheses 1-2)

We estimate Equation (**??**) (where q indexes contests) to test Hypotheses 1a-1b, which examines the effect of number of conceptual objective (measured by $(No.Concepts)_q$) on the number of participants ($(No.Designers)_q$) in contests with few vs. many conceptual objectives. To do that, we use 3 as the cutoff (i.e., $\mathbb{I}_q^{ShortCcpt} = \mathbb{I}\{(No.Concepts)_q \leq$

Table 2.1: **Summary Statistics**

Variable	Mean	Std	Min	Pcl(25)	Med	Pcl(75)	Max
$(No.Concepts)_q$	5.741	3.850	1	3	4	7	30
$Concept\ Similarity$	0.179	0.075	0.000	0.136	0.172	0.210	0.510
$(len_{Q3}/No.\ \text{Concept})_q$	3.507	3.648	0.625	1.250	2.000	4.000	28.909
$(No.GuideWords)_q$	6.515	4.494	0	3	6	9	26
$Length.\ Q\ 3$	25.376	34.564	2	5	11	33	317
$Length.\ Q\ 4$	20.701	31.604	2	4	9	27	442
$Length.\ Q\ 5$	44.723	48.080	2	12	30	64	346
$Avg.\ Sub\ Per\ Designer$	3.512	1.138	1.600	2.727	3.294	3.963	9.429
$No.Designers$	26.533	10.485	7	19	26	33	57
$No.\ Submissions$	71.998	36.624	14	48	65	88	385
$(Rated\ Ratio)_q$	0.544	0.313	0	0.3	0.6	0.8	1.0
$(No.\ 5\text{-}Star\ Submission)_q$	3.358	6.151	0	0	1	4	57
$(No.\ 5\text{-}Star\ DesignStyle)_q$	2.450	3.550	0	0	1	3	33

3}), and separately measure the impact from $(No.Concepts)_q$ when it is no more than 3 or above 3 concept keywords. (We tried alternative cutoffs and the qualitative results remain unchanged.) (2.6)

In the regression, we control for the degree of overlap among the manually coded conceptual objectives. (We compute the semantic similarity of coded objectives based on Wordnet, and use the average pairwise similarity ($Concept\ Similarity$; on a scale of 1-100) to account for the level of overlap among conceptual objectives.) Additionally, we control for the ratio $(len_{Q3}/No.\ \text{Concept})_q$, a proxy for the seeker's (lack of) conciseness, as well as the number of times the seeker updates the problem specification ($(No.Updates)_q$) during contest q. We include the industry fixed effect (i.e., the seeker's answer to Q2) to control for the possibility that different industries might have different levels of attractiveness to designers (seekers categorize themselves using 16 industry categories, including advertising and marketing, consulting and professional services, education and universities, etc.); we also include year dummies, day-of-week dummies and month fixed effects to control for possible seasonality effects or contemporaneous unobservables.

The estimation results for Equation (**??**) are presented in Column (1) of Table 2.2: the coefficient of $(No.Concepts)_q$ is significantly negative, and the coefficient of the interaction term $(No.Concepts)_q * \mathbb{I}_q^{ShortCcpt}$ is not significant, indicating that $(No.Designers)_q$ decreases with more disclosed conceptual objectives, in both contests with short and contests with long lists of disclosed conceptual objectives, and the speed of decrease is similar in contests with few and many disclosed objectives. These findings do not support Hypothesis 1a. This indicates a lack of empirical evidence that designers "guess" and incorporate undisclosed objectives in our empirical setting, which makes sense for the following reasons: (1) in open-ended design problems, designers might be less motivated to "guess": it could be that "guessing" a subjective design concepts can be relatively difficult (c_g is large), and the "guessed" objectives might not substantially improve design quality (ν is small); (2) the contests we focus on have "guaranteed" awards – the seeker must award a winner, regardless of design qualities, so only the designers' relative performance matters; (3) since in our empirical setting there are very few contests with a very small number of conceptual objectives, we might not be able to observe the "guessing" region where $S_r < \bar{r}^g$. (Note that that these reasons are mutually consistent; e.g., a large c_g implies a small $\bar{r}^g$.)

We now discuss Hypothesis 1b. In the paragraph above we learned that $(No.Designers)_q$ decreases with more disclosed conceptual objectives; this supports Hypothesis 1b. Moreover, we learned that dividing the data into contests with short and long lists of disclosed conceptual objectives made no difference as far as this effect is concerned; therefore, as an additional test we remove the "short conceptual list" variables ($\mathbb{I}_q^{ShortCcpt}$ and $(No.Concepts)_q * \mathbb{I}_q^{ShortCcpt}$) — thereby pooling contests with few and many conceptual objectives together — and re-run the regression in Equation (2.7). The updated estimation results are reported in Column (3) of Table 2.2, where we once again see that $(No.Designers)_q$ is significantly negatively associated with the measure of the number of conceptual objectives $(No.Concepts)_q$. This again supports Hypothesis 1b, and with a higher statistical significance due to the pooled data.

$$(No.Designers)_q = \beta_0 + \overbrace{\beta_1 (No.Concepts)_q + \beta_2 Concept\ Similarity}^{\text{Effect of Conceptual objectives}} + \overbrace{\beta_3 (No.GuideWords)_q}^{\text{Effect of Execution Guidance}} + \alpha_1 (No.Updates)_q + \alpha_2 (len_{Q3}/No.\text{ Concept})_q + \omega_{Industry} + \phi_{Week\ Day} + \delta_{Month} + \mu_{Year}. \quad (2.7)$$

It is also worth noting that $(No.Designers)_q$ is significantly positively associated with *Concept Similarity*, which supports an additional theoretical result from the extension (see Online Appendix EC.7.1) accounting for the overlap across conceptual objectives — the number of participating designers increases with the overlap across conceptual objectives, everything else equal.

Turning to Hypothesis 1c, Table 2.2's Columns (1) and (3) reveal that the number of participating designers in a contest ($(No.Designers)_q$) is not significantly associated with the amount of execution guidance $(No.GuideWords)_q$. Thus we also find support for Hypothesis 1c.

Next, we use Equation (2.8) to empirically test Hypotheses 2a-b. Designer i's trial effort in contest q is proxied by the number of submissions made by designer i to contest q ($(No.Submissions)_{i,q}$). (In Section 2.5.3, we consider alternative measures of designers' trial efforts.) We regress $(No.Submissions)_{i,q}$ on the amounts of different types of information in the problem specification (measured by $(No.Concepts)_q$ and $(No.GuideWords)_q$) in contest q. We control for $No.Designers_q$, $(No.Updates)_q$, and include industry dummies, day-of-week dummies, month dummies, year dummies and designer-specific dummies.

$$(No.Submissions)_{i,q} = \rho_0 + \overbrace{\rho_1 (No.Concepts)_q + \rho_2 Concept\ Similar}^{\text{Effect of Conceptual objectives}} + \overbrace{\rho_3 (No.GuideWords)_q}^{\text{Effect of Execution Guidance}}$$
$$+\zeta_1 (No.Updates)_q + \zeta_2 (No.Designers)_q + \upsilon_{Industry} + \chi_{Designer} + \eta_{Week\ Day} + \iota_{Month} + \psi_{Year}. \tag{2.8}$$

The estimation results for Equation (2.8) are presented in Column (5) of Table 2.2. These results show that $(No.Submissions)_{i,q}$ is significantly positively correlated with the amount of seeker execution guidance ($(No.GuideWords)_q$). This supports Hypothesis 2a — more seeker execution guidance leads to more submissions per designer. Moreover, $(No.Submissions)_{i,q}$ is not significantly associated with the number of conceptual objectives ($(No.Concepts)_q$), supporting Hypothesis 2b.

Table 2.2: Regression Results of Equations (4) (5) (6) (Including Alternative Measures)

Dependent Variable:	$(No.Designers)_q$				$(No.\ Submission)_{i,q}$	
	(1)	(2)	(3)	(4)	(5)	(6)
Conceptual Objectives						
$(No.Concepts)_q$	−0.357* (0.143)		−0.578*** (0.126)		0.013 (0.010)	
Concept Similarity	0.154* (0.069)		0.198** (0.064)		0.002 (0.005)	
$log(len_{Q3}+1)_q$		−2.281** (0.695)		−1.833*** (0.499)		0.082 (0.049)
Very Short Objectives						
$(No.Concepts)_q * \mathbb{I}_q^{ShortCcpt}$	−2.077 (2.236)					
$log(len_{Q3}+1)_q * \mathbb{I}_q^{ShortCcpt}$		−6.079 (8.641)				
$\mathbb{I}_q^{ShortCcpt}$	9.587 (6.522)	8.677 (14.560)				
Execution Guidance						
$(No.GuideWords)_q$	−0.015 (0.104)		−0.035 (0.105)		0.056*** (0.008)	
$log(len_{Q4}+1)_q$		−0.365 (0.512)		−0.369 (0.512)		0.113** (0.039)
$log(len_{Q5}+1)_q$		−0.212 (0.426)		−0.198 (0.425)		0.191*** (0.032)
Control Variables						
$(No.Designers)_q$					−0.009* (0.004)	−0.008* (0.004)
$(len_{Q3}/No.\text{Concept})_q$	−0.008 (0.133)		0.004 (0.131)			
$(No.Updates_q)$	0.014 (0.428)	−0.156 (0.439)	0.011 (0.432)	−0.163 (0.438)	0.197*** (0.034)	0.194*** (0.034)
Day-of-Week	Yes		Yes		Yes	
Month Fixed Effects	Yes		Yes		Yes	
Year Fixed Effects	Yes		Yes		Yes	
Industry Fixed Effects	Yes		Yes		Yes	
Creator Fixed Effects	No		No		Yes	
Observations	441	441	441	441	11,757	11,757
R^2 (Adjusted R^2)	0.271 (0.196)	0.220 (0.142)	0.253 (0.180)	0.217 (0.143)	0.479 (0.304)	0.480 (0.304)
Residual Std. Error	9.403 (df = 399)	9.711 (df = 400)	9.494 (df = 401)	9.706 (df = 402)	3.318 (df = 8796)	3.316 (df = 8796)
F Statistic	3.610***	2.822***	3.476***	2.932***	2.732***	2.738***

$^*p<0.05$; $^{**}p<0.01$; $^{***}p<0.001$

Note: (1) The qualitative results in this table remain unchanged with the inclusion of additional controls (including a dummy indicating whether a seeker describes himself in the problem specification as "open-minded" or not ($\mathbb{I}(OpenMind)_q$), as well as the likelihood of the seeker giving out ratings to submissions she receives ($\%Rated_q$)) or negative binomial count specifications.

(2) F-tests suggest that variables characterizing the problem specifications are important in explaining the contest outcomes (i.e., $(No.Designers)_q$ and $(No.Submissions)_{i,q}$).

(3) We conducted power analyses to ensure our tests have adequate statistical power to detect any significant variables. For the models represented in Columns (1)-(4) (in Columns (5)-(6)), our sample of 441 contests (11757 contest-designer combinations) can achieve a power that is above 0.95.

2.5.3 Robustness Checks (Hypotheses 1-2)

We perform a number of robustness checks to ensure the robustness of our empirical results for Hypotheses 1-2 with respect to the choices of measure, sample and model specification.

Alternative Measures for Conceptual Objectives and Execution Guidance: In our main empirical analyses, we use manual- and machine-coded keywords to measure the number of conceptual objectives and the amount of execution guidance in seekers' problem specifications. One may worry that there may be human and machine errors in the coding process, which can potentially affect the estimation results. To ensure our empirical results are robust to the measures of the main independent variables used, we consider an alternative measure, word-count, that does not suffer human or machine coding errors. In particular, we use the logarithm of length of the Q3 answer ($log(len_{Q3}+1)_q$) to proxy for the number of conceptual objectives, and the logarithm of length of the Q4 and Q5 answers ($log(len_{Q4}+1)_q$ and $log(len_{Q5}+1)_q$) to proxy for the amount of execution guidance. (Note that the log-transformation is applied to reduce the skewness of the distribution of the word count of seekers' answers to Q3/4/5 observed in the data, since some seekers are much more verbose in describing their conceptual objectives than others.) We then replace the original measure of conceptual objectives ($(No.Concepts)_q$) with $log(len_{Q3}+1)_q$, and the original measure of execution guidance ($(No.GuideWords)_q$) with $log(len_{Q4}+1)_q$ and $log(len_{Q5}+1)_q$. With these alternative measures, we re-estimate Equations (**??**)(2.7)(2.8) (the estimation results are reported in Columns (2)(4)(6) in Table 2.2, respectively), and find that all qualitative results remain unchanged.

Alternative Measures for Designer Trial Efforts: In this set of robustness checks, we construct two additional measures for a designer's trial effort in Equation (2.8): the number of distinct design styles submitted by designer i in contest q ($(No.DesignStyle)_{i,q}$), and the average number of minor variations designer i submits for each of her design styles in contest q ($(Variations/Style)_{i,q}$). The former represents the amount of designer efforts in creating designs of substantially different styles, and the latter represents the level of designer efforts in creating small variations of each distinct design style. (We classify designers' submissions into "distinct designs" and "variations" using an image comparison algorithm; see Online Appendix EC.2.) We rerun Regression (2.8) using $(No.DesignStyle)_{i,q}$ and $(Variations/Style)_{i,q}$ separately as the dependent variable, and find our qualitative findings are unchanged; see Table EC.6 in Online Appendix EC.5.

Sub-sample Analysis Using Contests that Did Not Have Updates: One may worry that the seeker's updates to her problem specification might be simultaneously correlated with the information provided in the initial problem specification and the contest outcome. To ensure the empirical results are not affected by this possibility, we estimate Equations (**??**)-(2.8) using a subsample containing only contests that had no updates, and find that the qualitative results remain unchanged; see Table EC.7 in Online Appendix EC.5.

Seemingly Unrelated Regression System: We further consider the possibility that certain contest-specific unobservables can affect both designer entry and trial effort decisions. If this indeed happens, the errors in Equation (**??**) or (2.7) and those in Equation (2.8) might be correlated. We model and estimate the SUR, and find that the results are almost the same as OLS regression results.

2.5.4 Additional Analyses (Hypotheses 1-2)

We further examine the differential effects of common (e.g., "experience" and "fun") and rare (e.g., "astrology" and "anti-mosquito") keywords for conceptual objectives and the differential effects of different categories of execution guidance (e.g., colors, logo styles, shapes. (See Online Appendix EC.4 for details of these analyses.) The results reveal that rare keywords are relatively less discouraging for designers' participation than common keywords, possibly because rare keywords are associated with clear and distinct definitions, whereas common keywords may be too vague and too general. This indicates that seekers should not be afraid to list specialty keywords. In addition, providing execution guidelines on "color", "shape", "art style", "usage" is more helpful in increasing the number of submissions by each designer, compared with providing execution guidelines on other aspects (such as "font" and "logo style").

We also explore whether the benefit (encouraging more submissions from each participating designer) from providing execution guidance is mostly coming from offering more categories of execution guidance, or from offering more detailed guidance in each category. We separate these two effects – (1) the number of categories and (2) the average number of keywords that fall into each category – and find that, it is providing execution guidance in various categories, rather than providing detailed execution guidance within each category, that helps reduce designers' trial costs and increase designers' trial efforts. (See Online Appendix EC.4 for details.)

2.5.5 Empirical Evidence for Seeker's Problem (Hypothesis 3)

Recall that we cannot directly observe seeker profit (i.e., the quality of the best design submitted), and therefore, Hypothesis 3 is about the relationship between the characteristics of the problem specification and the number of submissions the seeker rates as 5-star. As mentioned in Section 2.4, the seeker's rating (on a scale of 1 to 5 star) for each design is an imperfect signal of the quality of that design. Under the assumption that seeker rating behavior in different contests is relatively consistent and thus ratings in different contests are comparable (which is a fairly strong assumption), the more 5-star designs there are in a contest, the higher the expected "true" best design quality is likely to be. Therefore, any empirical evidence that supports Hypothesis 3 can also serve as indirect empirical support for the analytical findings about seeker information provision in problem specification presented in Section 2.3.4 (i.e., seekers should provide as many execution guidelines as possible, but only a moderate number of conceptual objectives). We first use the number of 5-star submissions in each contest $(No.\ 5\text{-}Star\ Submission)_q$ as the dependent variable in the empirical model to test Hypothesis 3. (In our sample, in only 0.25% of the contests, a design wins the contest without getting any rating, suggesting that non-rated submissions are typically of lower quality than 5-star submissions. Thus, the existence of submissions without ratings does not compromise our ability to measure contest-level quality by focusing only on 5-star submissions.)

First we examine contests with more than a moderate number of conceptual objectives ($(No.Concepts)_q \geq 4$). Based on the analytical model, we expect that in those contests, fewer conceptual objectives and longer execution guidance would lead to more 5-star submissions in contests. To test that, we regress $(No.\ 5\text{-}Star\ Submission)_q$ on the numbers of conceptual objectives ($(No.Concepts)_q$) and execution guidelines ($(No.GuideWords)_q$), along with the same control variables used in Table 2.2. The estimation results (reported in the first column of Table 2.3) suggest that $(No.\ 5\text{-}Star\ Submission)_q$ is negatively and significantly associated with $(No.Concepts)_q$, and is positively and significantly associated with $(No.GuideWords)_q$; these empirical results provide support for our analytically-driven recommendation that seekers should provide extensive execution guidance but not too many conceptual objectives. For completeness, we perform the same analysis on contests with few conceptual objectives (no more than 4 conceptual keywords), and find that the best design quality is not decreasing with more $(No.Concepts)_q$ (reported in the third column of

Table 2.3).

Robustness Check (Hypotheses 3). To account for the possibility that multiple submissions that are small variations of the same design may not meaningfully contribute to the best design quality, we also consider an alternative dependent variable – the number of distinctive design styles with 5-stars in the contest $(No.5\text{-}StarDesignStyle)_q$. (Online Appendix EC.2 explains how we identify distinctive design styles.) We re-run the aforementioned tests using $(No.5\text{-}StarDesignStyle)_q$, and report the results in Columns 2 and 4 in Table 2.3. As can be seen from the results, the qualitative findings remain the same under this new dependent variable.

Although they should be interpreted with due caution (as mentioned in Section 2.4), these empirical results provide suggestive evidence supporting our theoretical findings on the seeker's problem. While these findings on quality focus on the *contest*-level outcome, as this is what seekers ultimately care about, we also conducted an analogous analysis at the individual-level and again found empirical support (this additional analysis is omitted here).

Table 2.3: Best Design Quality with Problem Specification

	Long Sample ($(No.Concepts)_q \geq 4$)		**Short Sample** ($(No.Concepts)_q \leq 4$)	
Dependent Variable:	$(No.\ 5\text{-}Star\ Submission)_q$	$(No.\ 5\text{-}Star\ DesignStyle)_q$	$(No.\ 5\text{-}Star\ Submission)_q$	$(No.\ 5\text{-}Star\ DesignStyle)_q$
Conceptual Objectives				
$(No.Concepts)_q$	−0.213* (0.108)	−0.155* (0.070)	0.157 (1.224)	0.523 (0.696)
Concept Similarity	−0.125 (0.086)	−0.025 (0.056)	0.025 (0.086)	0.032 (0.049)
Execution Guidance				
$(No.GuideWords)_q$	0.217** (0.083)	0.178** (0.054)	0.177 (0.144)	0.083 (0.082)
Control Variables				
$(len_{Q3}/No.Concepts)_q$	0.901** (0.270)	0.606*** (0.176)	1.259* (0.580)	0.853* (0.330)
$(No.Updates)_q$	0.411 (0.388)	−0.093 (0.252)	0.805 (0.654)	0.180 (0.372)
$(Rated\ Ratio)_q$	6.498*** (1.745)	3.007** (1.135)	9.799*** (2.649)	5.005** (1.507)
DoW, Month, Year Dummies	Yes	Yes	Yes	Yes
Industry Fixed Effects	Yes	Yes	Yes	Yes
Observations	230	230	175	175
R^2	0.309	0.300	0.355	0.329
Adjusted R^2	0.163	0.151	0.169	0.135
Residual Std. Error	5.445 (df = 189)	3.540 (df = 189)	7.475 (df = 135)	4.251 (df = 135)
F Statistic	2.117***	2.022***	1.908**	1.697*

*p<0.05; **p<0.01; ***p<0.001

Note: We exclude contests with less than 20% submissions rated ($(Rated\ Ratio)_q \leq 0.20$), since in those contests, the $(No.\ 5\text{-}Star\ Submission)_q$ is not a good a measure for the best design quality. We also exclude 3 contests with extremely inconcise Q3 ($(len_{Q3}/No.Concepts)_q > 15$).

2.6 Further Evidence from the Field

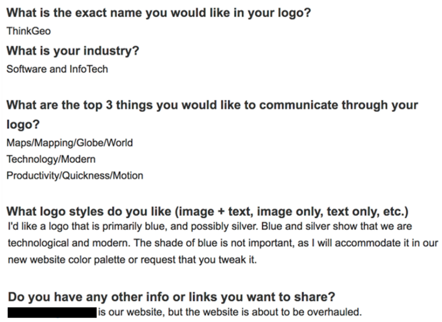
In this section, we discuss a recent change to the crowdsourcing platform we collect data from: the template for seekers' problem specifications was updated. This update, as to be shown below, not only validates our earlier recommendations on how seekers should provide information in problem specifications, but also introduces an exogenous shock to seeker problem specification behavior on the platform and thus provides an opportunity for us to further identify the effects of disclosing conceptual objectives and execution guidelines on designers' behavior.

The update took place on August 31, 2017. Figure 2.1b is a screenshot of the problem specification template after the website update; the original problem specification template is shown in Figure 3.1a. The update involves two major adjustments. First, the platform provided an example when prompting the seeker to specify conceptual objectives (i.e., "top 3 things to communicate"). As can be seen from the screenshot (Figure 2.1c), the example only includes three keywords: "modernity, professionalism, strength". Potentially, this short three-keyword example can lead seekers to shorten their list of conceptual objectives. Second, the updated template changed the way the seeker provides execution guidelines. Instead of providing a short text answer to the "what logo styles do you like" question in a text box, seekers are asked to answer four multiple-choice questions, including logo usage (e.g. screen/digital, clothing), preferred logo style (e.g. image+text, image only), preferred fonts (e.g. sans-serif, mono), colors to explore (e.g. aqua, green).

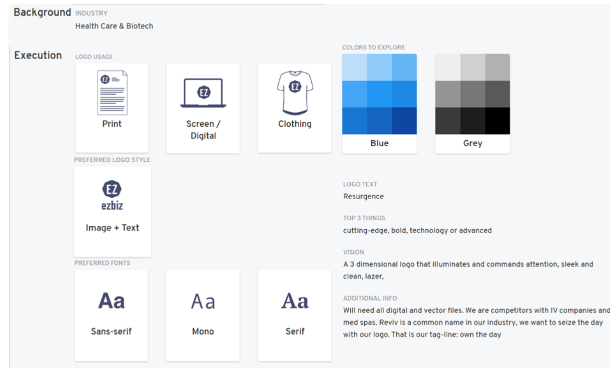
Since this update does not affect how seekers answer Question 5 (Q5) much, we do not consider Q5 in the following analysis. Besides the two main changes described above, we also observe that the platform added a new area called "vision" in the template. This new area is located below the "top 3 things" box, providing additional space for the seeker to elaborate what message he envisions a logo to convey. We find that most of the time, content provided in the "vision" area does not involve introducing *additional* conceptual objectives. In the manual coding of conceptual objectives for contests that took place after the website update, we incorporate the occasional additional objectives that are mentioned in the "vision" area.

This update shows that the platform recognizes problem specification as a crucial design element for crowdsourcing design contests, and distinguishes among different types of information conveyed through the problem specification, i.e., conceptual objectives and execution guidelines. Moreover, the example the platform provides for specifying the "top 3 things" (Figure 2.1c) is also aligned with our recommendation not to specify too many conceptual objectives.

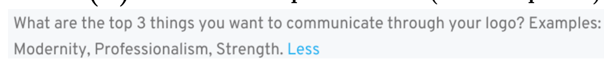
In addition, the website update provides an exogenous shock to how seekers specify their problems. In the empirical analysis presented in Section 2.5, the identification of the effects of disclosed conceptual objectives and execution guidelines relies on cross-contest variations in the amounts of these two types of information provided under the same website layout. One may argue that these variations could be driven by seekers' unobserved idiosyncratic characteristics, and if those seeker characteristics can also af-



(a) **A Problem Specification (Before Update)**



(b) **A Problem Specification (After Update)**



(c) **The Platform's Prompt for Conceptual objectives (After Update)**

Figure 2.1: Screen-shots for Problem Specifications Before and After the Platform's Update on Template

fect designers' participation behavior, then the regression models presented earlier may suffer from an endogeneity problem. The update to the problem specification template imposes an exogenous shock on the number of conceptual objectives and execution guidelines, which is unlikely to be correlated with seekers' idiosyncratic characteristics, allowing us to better identify the effects of them on designers' participation behavior.

Specifically, we collect additional data on all logo design contests that took place on the platform from June 01, 2017 to December 01, 2017 (three months before and three months after the update). As in the empirical analysis presented in Section 2.5, we focus on contests where the design seekers promise to give a \$200 award to one and only one winner in 7-day contests. The contests that took place during the transition of the platform website (i.e., 10 days before and after August 31, 2017) are excluded. This new sample consists of 102 contests (2,819 contest-designer combinations). Below, we first establish how the template update affects the numbers of conceptual objectives and execution guidelines seekers provide in their problem specification, and then study how these changes in seekers' problem specification affect designers' entry and trial effort decisions.

Conceptual objectives: We first test whether the number of conceptual objectives decreases, as expected, after the website update. Specifically, we compare the number of manually coded conceptual objectives before and after the update: the average number of conceptual keywords before the update ($(No.\,\text{Concept})_q^{\text{pre}}$) is 5.924, and the number after the update ($(No.\,\text{Concept})_q^{\text{post}}$) is 2.806, and the difference is statistically significant (p-value<0.001). Note that, in the manual coding process, the coders report the total number of conceptual keywords mentioned in both the "top 3 things" and "vision" areas. These results suggest that seekers indeed provide fewer conceptual objectives after the update.

Next we examine how fewer disclosed conceptual objectives affect designer behaviors. Specifically, we regress the number of designers in each contest on the dummy variable $\mathbb{I}(\text{post-update})_q$, which indicates whether the focal contest took place after the website update. We include day-of-week and industry dummies in the regression as control variables. The estimation results for this regression are shown in the first column of Table 2.4. The estimated coefficient of the dummy variable $\mathbb{I}(\text{post-update})_q$ is positive and significant, indicating that after the update, the number of participating designers increases. This result, combined with the finding that after the website update, seekers disclose fewer conceptual objectives, is again consistent with Hypothesis 1b. To make sure the increase in $(No.Designers)_q$ after the website update is indeed due to the decrease in the number of disclosed conceptual objectives, but not other concurrent changes to the platform, we conduct the following analysis. We regress the number of participants in each contest on not only $\mathbb{I}(\text{post-update})_q$, but also $(No.Concepts)_q$ and $\mathbb{I}(\text{post-update})_q * (No.Concepts)_q$. The estimation results are reported in the second column of Table 2.4. After controlling for $(No.Concepts)_q$, $\mathbb{I}(\text{post-update})_q$ no longer correlates with $(No.Designers)_q$, indicating that the increase in $(No.Designers)_q$ is explained entirely by the change in the number of disclosed objectives ($(No.Concepts)_q$). In addition, the interaction term $\mathbb{I}(\text{post-update})_q * (No.Concepts)_q$ is not significantly associated with $(No.Designers)_q$, suggesting that the effect of conceptual objectives on participation is not significantly different before and after the platform update.

Table 2.4: Regression Results for the Update's Impacts on Contests' Outcomes

Dependent Variable	$(No.Designers)_q$		$(No.Submission)_{i,q}$		$(No.5\text{-}StarSubmission)_q$
Overal Changes					
$\mathbb{I}(\text{post-update})_q$	6.795* (3.090)	8.454 (5.283)	−1.310** (0.479)		0.094 (0.859)
Conceptual Objectives					
$(No.Concepts)_q$		−0.879* (0.412)			
$\mathbb{I}(\text{post-update})_q * (No.Concepts)_q$		−1.208 (1.399)			
Execution Guidance					
$\mathbb{I}(\text{short Q4})_q * \mathbb{I}(\text{pre-update})_q$				−0.058 (0.177)	
$\mathbb{I}(\text{long Q4})_q * \mathbb{I}(\text{pre-update})_q$				0.692** (0.251)	
Control Variables					
$(No.Designers)_q$			0.002 (0.006)	0.005 (0.006)	
Industry Fixed Effects	Yes	Yes	Yes	Yes	Yes
Designer Fixed Effects	No	No	Yes	Yes	No
Day-of-Week Fixed Effects	Yes	Yes	Yes	Yes	Yes
Observations	102	102	2,819	2,819	102
R^2 (Adjusted R^2)	0.467 (0.263)	0.515 (0.309)	0.425 (0.257)	0.426 (0.259)	0.452 (0.241)
Residual Std. Error	12.059 (df = 73)	11.669 (df = 71)	2.827 (df = 2182)	2.824 (df = 2181)	3.354 (df = 73)
F Statistic	1.546	2.284**	2.532***	2.545***	2.146**

Note: *p<0.05; **p<0.01; ***p<0.001

Execution Guidelines: Since the platform updated the form of execution guidelines from open-ended descriptions to multiple-choice questions, we cannot directly measure the change in the number of execution guidelines; however, we notice some changes in the information provided. Prior to the update, the number of guidelines provided varies a lot from seeker to seeker — some seekers provide extremely detailed execution guidelines, including suggestions for shape, color, and pattern (e.g., "*Please use variations of red, black, silver and gold (matte, not shiny). Make sure that the logo doesn't rely on silver and gold effects, need a color palette that is easily translatable to web.*"), whereas others provide very brief guidelines (e.g., "*image+text*"). After the update, this seeker heterogeneity no longer exists, because all seekers are now required to answer the exact same set of multiple-choice questions.

We regress the number of submissions per designer on $\mathbb{I}(\text{post-update})_q$ and report the results in the third column of Table 2.4. Quite surprisingly, the coefficient of $\mathbb{I}(\text{post-update})_q$ turns out to be negative and significant, suggesting that on average each designer submits *fewer* designs after the update. Why is this the case? One possible explanation is that, as mentioned previously, the template update removes seeker heterogeneity in execution guideline provision. Prior to the update, some seekers are willing to provide detailed guidelines, and others are not. For seekers who would provide few execution guidelines in the open-ended Q4 template, the switch to the standardized questionnaire may have increased the number of guidelines they provide; whereas for seekers who would provide detailed guidelines in the open-ended Q4 template, the switch in fact limits the amount of execution guidance they can provide. An average negative effect reflected in the negative estimated coefficient of $\mathbb{I}(\text{post-update})_q$ seems to suggest that more seekers suffer the negative effect of the switch to the standardized questionnaire format than those who benefit from it. The test reported in the fourth column of Table (2.4), where we separately evaluate the difference in the number of submissions per designer between the post-update contests and pre-update contests with short ($\mathbb{I}(\text{short Q4})_q$) vs. long ($\mathbb{I}(\text{long Q4})_q$) execution guidelines, supports this explanation.

To fully understand the mechanism driving the effect of the format change for execution guidelines, we compare the aspects of the execution guidelines seekers provide before and after the update. Recall in Section 2.5.4, we explore the importance of seven categories of execution guidelines (colors, logo styles, shapes, font, usage, art styles, and resources). We now examine whether the multiple-choice questions after the platform update include the important categories. The comparisons reported in Table EC.2 in Online Appendix EC.4 generate several interesting observations. (1)

Providing guidelines for "font" and "logo style", two of the four categories included in the new multiple-choice format, is not very helpful in increasing the designer trial effort. (2) Providing guidelines for "shape" and "art style", categories that are not included in the new multiple-choice format, has a significant positive effect on the designer trial effort. (3) The multiple-choice format limits the amount of information the seeker can provide for each category. Consider "usage" as an example. After the update, the seeker can only choose from the five options provided (outdoor, clothing, screen/digital, print, signature); whereas, prior to the update, seekers could, and in fact did provide more detailed information about logo usage (e.g. embroidery, building, banner, sticker, letterhead, device, t-shirt, vest, uniform, hat, etc.). All these results suggest that the new multiple-choice format appears to be less effective for execution guideline provision than the old open-ended format, and therefore, this change may have increased designers trial cost (c_3 in our analytical model), which then leads to the decrease in the number of submissions per participant we observe in the data. This result further supports Hypothesis 2a.

We also examine whether the number of submissions rated as 5-star changes after the template update, and find that it does not (see the last columns of Table 2.4). This is not surprising: as discussed above, the update effects an increase in the number of participants, and a decrease in the number of submissions per participant, influencing the best design quality in opposite directions.

2.7 Conclusion

In this paper, we combine analytical and empirical methods to examine the effects of different types of information provided in seekers' problem specifications — namely, conceptual objectives and execution guidance — on designers' participation behavior and contest outcomes, and prescribe recommendations for optimal information provision in problem specifications.

Our study provides the following novel insights. First, our theoretical analysis suggests — and our empirical analysis confirms — that the number of participating designers in a contest eventually decreases with more conceptual objectives provided in the problem specification, that the trial effort provision increases with more execution guidance provided in the problem specification, and that providing more execution guidelines increases the quality of the best design sourced from the contest, whereas specifying too many conceptual objectives can negatively affect the best design quality. Further, we exploit a "quasi-natural experiment" (involving an update to the platform's problem specification template) to strengthen the reliability of our empirical results. In addition, the update made by the platform directionally supports our recommendations for seekers' problem specifications, and our detailed textual analysis provides further insights into what types of execution guidelines are more helpful in reducing designers' cost to come up with design trials.

Like any research, our study has a few limitations. First, our empirical analyses are based on data collected from a single crowdsourcing platform. Although we believe that contests on this platform are representative, as it is one of the largest crowdsourcing platforms for creative services, further empirical analysis of data collected from other crowdsourcing platforms would be helpful in establishing the external validity of our findings. Second, we focus on how the initial problem specification affects the contest outcome, and do not consider the updates to the problem specification that might take place during the contest period. Such updates are relatively rare in our data, but in other settings where updates to specifications are more prevalent, the effects of the problem specification updates may require special attention. Third, because analyzing a model with both the agent's output uncertainty and heterogeneity is considered intractable in the innovation-contest literature (e.g., Terwiesch and Xu (2008), Ales et al. (2017a), Korpeoglu et al. (2017)), we assume homogeneous designers and focus on the impact of the designers' solution uncertainty.

Despite these limitations, our study offers rich empirical evidence from the field to support the predictions and recommendations from our theoretical model, linking the theoretical and empirical research of crowdsourcing contests. This study also proposes a novel theoretical model building upon the design literature to characterize designers' design process, which allows us to examine the role of information in distinct stages of this design process and its impact on contest outcome. This theoretical modeling framework can be used to study other design problems in the context of open design/creative contests. As one of the first papers studying problem specification in crowdsourcing contests, this study contributes to the academic literature of crowdsourcing contests. It also provides rich managerial implications, especially for individuals or organizations that are using crowdsourcing contests to source creative solutions or products (i.e., seekers in crowdsourcing creative contests).

CHAPTER 3

The Role of Feedback in Dynamic Crowdsourcing Contests: A Structural Empirical Analysis

3.1 Introduction

Crowdsourcing contests have become a popular way for organizations to source innovations. Compared to traditional innovation sourcing approaches, crowdsourcing contests allow an innovation seeker to access and interact with a much larger pool of innovators, to choose from a large number of submissions, and to pay only for the successful ones. This new way of sourcing innovation increases the variety and novelty of innovations, and lowers the risk of innovation failure.

To perform well, crowdsourcing contests must be designed well. Researchers have studied the performance effects of award structure, competition size, problem specification, etc. Another important, but somewhat overlooked design element is the feedback disclosure policy. The feedback disclosure policy is important in *dynamic crowdsourcing contests*. Such contests are used in many application contexts, including algorithm design, web-design, logo designs, etc. In these dynamic crowdsourcing contests, solvers can enter a contest at any time before the contest ends; incumbent solvers (those who have already made submissions) can submit additional submissions that exploit their existing ideas or explore new ideas, and these two strategies require different levels of effort and produce different distributions of quality improvement; and there exist uncertainties in the relationship between effort level and performance.

By disclosing feedback, the seeker resolves uncertainty about solvers' performance. For example, Kaggle contests provide public leader boards displaying rankings based on the performance of submitted algorithms on a holdout data set. Many crowdsourcing design platforms allow design-seekers to rate submitted designs using a rating system, and the rating distribution is publicly viewable; such platforms typically strongly encourage seekers to provide feedback to solvers throughout the contest (for example, the FAQ site of one platform states “the more feedback you can give, the better!” and “be sure to score EVERY entry”). The intuition behind providing performance feedback in innovation contests is that feedback can guide solvers’ submissions towards better performance.

However, the effect of performance feedback in crowdsourcing contests is likely to be more complicated than that. In addition to informing solvers of how well they themselves are performing, feedback also discloses the current status of the competition, and thus may affect existing and potential solvers’ participation decisions. Dating back to Fudenberg et al. (1983), theoretical literature on feedback in small-scale (mostly two-player) contests has generally concluded that the revealed performance gap will lead to both the low performer and the high performer reducing their efforts due to decreased competition. But this conclusion may not hold in dynamic crowdsourcing contests, because such contests are large-scale; new entrants can join dynamically; incumbent solvers can dynamically choose between exploitative and exploratory strategies; and there exist uncertainties in the relationship between effort level and performance (e.g., in algorithm contests, how well the algorithm will perform on the holdout data; in design contests, how well the seeker will like a design). In the presence of these unique features, revealing performance gaps may encourage solvers to exert effort: new entrants are encouraged to join the contest when the feedback indicates a low level of competition; top performers want to secure their leading position as there are often multiple high performers competing against each other; and low performers may endeavor to catch up – due to the performance uncertainties, solvers whose current submissions fall far behind have the option of submitting a completely new innovation, which gives them a chance to leapfrog the competition.

Given the co-existence of all these effects, there is no easy answer to the important question of whether (and if so when) to release performance feedback during dynamic crowdsourcing contests. The goal of this study is to examine this complicated process and disentangle the intertwining effects that feedback may have on the outcome of such contests. More specifically, this study attempts to address the following research questions: (1) How does performance feedback affect incumbent solvers’ exploitation/exploration actions and potential entrants’ entry decisions? (2) What is the impact of the availability of feedback on the contest outcome, in terms of the total number of solvers participating in the contest, the highest quality achieved by their submissions, and the number of top performers/submissions? Does the timing of the feedback availability matter or not? (3) What is the value (disvalue) of providing feedback in crowdsourcing contests?

We take a structural modeling approach to answer these questions. The dynamic structural model we create explicitly captures how potential entrants decide whether to join an on-going contest, and how incumbents decide whether to make additional submissions, and if so, how to choose between the exploratory and exploitative strategies. We apply the structural model to a data set collected from a major platform for crowdsourced custom logo designs. (A generalizability discussion is provided in Section 3.7 paragraphs 4-6.) In order to classify observed incumbents' follow-up submissions into exploratory action (*redesign*) and exploitative action (*revision*), we employ the computer vision Scale-Invariant Feature Transform (SIFT) algorithm to quantify the similarity between each pair of submissions made by the same creator. We estimate parameters governing solvers' participation behavior, including solvers' entry costs, the cost associated with exploratory actions (*redesign*s), and the cost associated with exploitative actions (*revision*s).

Using the estimated parameters, we conduct counterfactual simulations to compare contest outcomes across four feedback policies: full/early/late/no feedback, where the seeker provides feedback throughout/only early/only late/not at all in the contest. In brief, the simulation results reveal that if all that the seeker cares about is the maximum quality achieved, both the full feedback policy and the late feedback policy perform quite well. Moreover, if the seeker's objective is to maximize the number of high performers, or the total number of participants in the contest, the late feedback policy is the best option. These findings underscore the value of providing feedback, but also suggest that the aphorism, "the more frequently the seeker provides feedback the better", may be misleading depending on the seeker's objectives.

Our study makes several contributions. First, to our knowledge, this is the one of the first studies to investigate the impact of feedback on the outcome of crowdsourcing contests, and the first study that that proposes a structural model to empirically analyze this impact. Second, the structural model presented in this paper is one of the most comprehensive models of crowdsourcing contests, as it explicitly models multiple stages, endogenous entry, and solvers' exploitation and exploration innovation strategies, unique features of crowdsourcing contests that, to our knowledge, have not been all incorporated in the existing literature. Third, our policy simulations provide rich managerial insights into whether, and if so, when should feedback be revealed, depending on the objective(s) that the seeker wants to achieve.

3.2 Literature Review

Emerging large-scale online markets have attracted increasing academic interest from the operations management community (see Chen et al. (2019) for a summary of recent papers, which cover a wide range of application contexts). The focus of this paper is the *crowdsourcing design contest* (Girotra et al., 2010), in which a large crowd of innovators/designers *compete* to solve design problems.

Crowdsourcing contests have been examined as platforms for sourcing innovation (Terwiesch and Xu, 2008). Existing literature on the design of crowdsourcing contests has looked at the impact of award structure (Ales et al., 2017b); competition size and open/closed entry (Boudreau et al., 2011, 2016; Ales et al., 2017a; Körpeoğlu and Cho, 2017); problem specification (Erat and Krishnan, 2012); solvers' choices among coexisting contests (Körpeoğlu et al., 2017); and the disclosure of intermediate solutions (Boudreau and Lakhani, 2015; Bockstedt et al., 2016; Wooten and Ulrich, 2017a) on the outcome (both the quality and quantity of the crowdsourced solutions) of crowdsourcing contests. Most of the previous research studies crowdsourcing contests as static problems; however, in reality, many crowdsourcing contests are dynamic in nature (Korpeoglu et al., 2017; Hu and Wang, 2017). Our paper explicitly models the dynamics of crowdsourcing contests, and focuses specifically on an important but less explored design element – feedback. To our knowledge, the only published study that looks at the role of feedback in crowdsourcing contest is Wooten and Ulrich (2017b), in which the authors conduct a field experiment to compare the effects of three different feedback treatments – no feedback, random feedback, and truthful feedback. However, their paper does not explicitly model the contest dynamics, nor study the detailed mechanisms driving participants' behavior. They run field experiments with six contests, and do not examine the impact of the timing of performance feedback on contest outcomes; getting sufficient observations on contest outcomes would require many more contests to be run, which would be prohibitively difficult using field experiments. Instead, in our paper we employ a structural estimation approach whereby we simulate contest outcomes under different feedback policies using an empirically estimated structural model.

Feedback in a different but related context, tournaments/contests, has been studied in the economics literature. The theoretical research focuses on small-scale contests, and until recently, most studies in this stream of research conclude that performance feedback reduces contestants' effort. A few recent papers (e.g., Aoyagi 2010; Goltsman and Mukherjee 2011) extend the literature by showing that the optimal feedback mechanism depends on the curvature of the agents' cost function, complementarity between effort and ability, complementarity between contestants' effort, etc. There are also laboratory experimental studies evaluating the role of feedback in contests. These studies have mixed results in terms of whether and how feedback will affect top-performing and low-performing contestants' effort provision in various experimental settings (see Dechenaux et al. (2015) for a detailed summary of this literature). There has been very little work featuring feedback in contests for innovation. To our knowledge, there are only a few recent game-theoretic papers on this topic. Bimpikis et al. (2017) and Halac et al. (2017) both study "innovation races" and consider technical uncertainties in innovation races – namely whether it is feasible to solve the problem at all. They illustrate that feedback on the one hand exposes a discouraging performance gap, but on the other hand updates contestants' perceptions of the feasibility of solving the problem. Mihm and Schlapp (2018) evaluates the average design quality and the best design quality under no feedback, public feedback and private feedback scenarios. They find that the best feedback strategy depends on the contest uncertainty and the contest holder's interest in average design quality or the best design quality. A recent work by Gross (2017) is one of the few papers that use field data to examine the effect of feedback on small-scale, fixed-size innovation contests. Concurrently but independently from our work, using a crowdsourcing data set similar to ours, the author estimates how feedback affects contestants' participation and the quality of their subsequent submissions, in a model that treats each submission by the same participant as an independent trial. Then the author uses the estimated model to simulate the dynamics of a three-player, sequential-play contest, under alternative feedback policies, including the public, private and partial feedback policies. The simulation results suggest that the net effect of feedback on the number of high-quality submissions is positive; therefore, the author concludes that feedback is desirable for a principal seeking innovation.

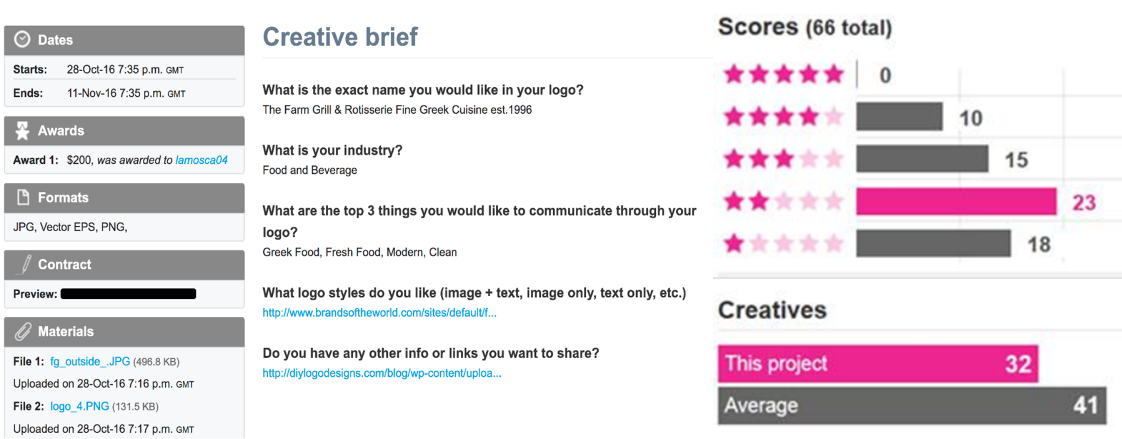
Our work considers a similar principal decision (feedback), but in a different setting – crowdsourcing contests held on large-scale online markets. The crowdsourcing contests we study differ from small-scale fixed-size contests in the following crucial aspects: (1) the size of the participant pool is large; (2) crowdsourcing contests allow endogenous entries of new participants throughout the contest; and (3) there is large uncertainty in the relationship between effort and solution quality. The last feature gives rise to exploration- and exploitation-type strategies; indeed, sourcing the best solution to an innovation problem has been modeled as a search process (Girotra et al., 2010; Kornish and Ulrich, 2011; Erat and Krishnan, 2012), where independent trials (exploration) and sequential trials (exploitation) contribute differently to the contest outcomes. In the presence of these unique features, we expect that the role of feedback will be quite different in crowdsourcing contests, and that findings in the literature about the role of feedback in traditional contests may not hold in the context of crowdsourcing contests.

The model we propose in this paper captures these unique features of crowdsourcing contests by explicitly modeling potential entrants' entry decisions and incumbents' choices between exploratory and exploitative strategies in this highly uncertain environment, and, as a result, provides novel insights into the role of feedback in crowdsourcing contests. In addition, our study contributes to the limited empirical literature on the design of crowdsourcing contests and on the sourcing of innovative products. Using a structural modeling approach, we are able to recover the parameters governing contest participants' behavior from real-world data and conduct counterfactual simulations to evaluate alternative feedback disclosure structures. Although the focus of our study is the role of feedback, our structural model of contest participants' behavior can be used to study other features of crowdsourcing contests and other large-scale online open platforms.

Methodologically, our paper is based on the dynamic game structural estimation literature (see Aguirregabiria and Mira (2010) for a detailed review). The most closely related papers are Sweeting (2009), and Igami (2016), each of which estimates a dynamic game using the nested fixed-point approach (NFXP) initially proposed by Rust (1987), as opposed to the two-step approach (Hotz and Miller, 1993). Our modeling and estimation approach diverges from the more conventional framework where the market is assumed to be in a stationary environment and the competition has an infinite horizon. Specifically, we embed discrete choice with private information into a non-stationary, finite-horizon dynamic game, focusing on type-symmetric strategies to avoid multiple equilibria. Our work also contributes to the growing empirical operations management literature that employs structural modeling to examine operations-related questions (e.g., Lu et al. (2013), Moon et al. (2016), Zheng (2016)).

3.3 Research Context and Data

The data we use for the empirical analysis is from crowdsourced creative contests hosted on an online platform. We focus on logo design contests, which are the largest category on the platform both in terms of the number of completed contests and the number of active solvers (hereafter we refer to solvers as "creators" in this context of graphic design contests). A typical logo design contest proceeds as follows. First, a seeker ("he") in need of a design posts a design request. In the posting, he describes what he needs, specifies when he needs it (i.e., the contest's start date and end date), chooses whether he wants to make all existing submissions publicly viewable, and announces the award structure (i.e., whether the award is guaranteed by the seeker, the number of winners, and the award(s) for the winner(s)). Once a contest is posted, all creators ("she") on the platform can join the contest and submit design(s) at any time before the contest ends; additional submissions by a creator do not replace her earlier submissions. The seeker can rate each submission based on a 5-star system.[1] A submission's rating is visible to its author. All ratings are summarized in a table, (Figure 3.1), which is accessible to all participating and potential creators.[2] At the end of the contest, the seeker picks his favorite submission(s) and gives the pre-announced award to its (their) author(s).



(a) Design Request (b) Project Stats

Figure 3.1: Illustrative Screenshots of the Design Request and Project Stats

Note: "Buyer Assured", "Public-gallery", "Award", "Start-date and End-date" and "Project length" are labeled on the contest list page and on a project's header.

We collect data of logo-design contests with publicly viewable submissions on this crowdsourcing platform from March 2012 to November 2014. For each contest, we record all participants' activities (including creators' submissions and the seeker's rat-

[1]In contests for creative products, the seeker is likely to discover new options after seeing more submissions. This is captured by our model. That said, our model cannot capture the possibility of seekers' preferences changing as they see more submissions (e.g., his evaluation of the same design changes as he sees more submissions); however, we found little support in the data for this possibility (please see Online Appendix B.2.3 for details).

[2]The platform strenuously polices copyright violations to prevent creators from copying each other's designs. As a result, we found little evidence in the data that creators copy each others' designs (see Figure B.2 in Online Appendix B.2.2), suggesting that the platform's anti-copying efforts are successful.

ings) with corresponding time stamps. To facilitate the empirical analysis, we focus on 7-day contests where the design seekers promise to reward one and only one final winner, because it has been documented (Moldovanu and Sela, 2001; Liu et al., 2014) that the contest length and award structure can affect creators' behavior and contest outcomes; since the objective of this study is to examine the role of feedback, we purposefully minimize the heterogeneity among the contests in these other dimensions. The contests included in our sample are representative contests on the platform – 97% of the contests held on the platform have a single award, 61% are guaranteed award, and 7-day is the most common length among all contests. The final working sample consists of 810 contests, 26,367 contest-creator combinations, 75,572 design images and 45,999 ratings. Table 3.1 reports key summary statistics of contest-level characteristics; to save space in the body of the paper, additional descriptive statistics are provided in Online Appendix B.2.2.

Table 3.1: Contest-Level Summary Statistics

Variables	Mean	St. Dev.	Min	Pctl(25)	Median	Pctl(75)	Max
Award ($)	260.84	97.57	200	200	200	300	1,000
No. Submissions	93.30	73.39	14	56	77	109	1,221
No. Creators	32.55	19.81	6	20	29	39	233
No. Submissions with Ratings	56.79	63.13	0	19	43	73	927
No. 1-Star Submissions	7.14	21.78	0	0	1	6	411
No. 2-Star Submissions	10.82	17.03	0	1	6	14	240
No. 3-Star Submissions	17.10	22.85	0	4	11	23	347
No. 4-Star Submissions	11.00	13.37	0	3	7	15	141
No. 5-Star Submissions	3.16	5.74	0	0	1	4	48

3.3.1 Classifying Creators' Actions

When exploring creators' submission patterns in our data, an existing creator's follow-up submissions are sometimes similar to her previous submissions, while other times very different. The former type of follow-up submissions can be considered "revisions" of a creator's previous submissions, akin to an "exploitative" innovation strategy; by contrast the latter type involves creating design(s) that are significantly different from any of the creator's existing designs (later defined as "redesigns"), corresponding to an "exploratory" innovation strategy.[3]

[3]There are also a small number of cases where a follow-up submission is almost identical to one of the creator's previous submissions. We classify these submissions as *replications*, and in the later analysis, *replications* are not counted as follow-up submissions and are removed from the data, since we assume "replicating" an existing design incurs very little cost, and does not benefit the design seeker very much.

Both types of follow-up submissions are "new" submissions in that they do not replace existing submissions, but distinguishing one type ("revision") from the other ("redesign") is important. For the seeker, these different submission types contribute differently to the pool of designs seekers can choose from – a "revision" may result in an incremental quality difference relative to prior designs, whereas a brand-new "redesign" may impart greater quality differences relative to previous designs and increases the variety of the submission portfolio. From a creator's standpoint, it could cost them different amounts of effort to make these two types of submissions: tweaking an existing design for a "revision" is likely to require less effort, while creating a brand-new "redesign" will possibly require much more effort.

To systematically classify creators' follow-up submissions observed in the data into "revisions" (exploitative) and "redesigns" (exploratory) on a large scale, we employ *Scale-Invariant Feature Transform (SIFT)*, an algorithm used to detect and describe local features in images proposed by Lowe (1999). Our approach consists of four steps (details with examples are provided in Online Appendix B.1). First, from a pair of design images (A and B), we extract descriptors of the key points by identifying *SIFT feature vectors* in scale space, which robustly capture the structural properties of the images. Second, we match SIFT feature vectors by calculating and comparing the Euclidean distance between each of the SIFT feature vectors in image A and image B. Third, using the obtained matched feature-vector pairs, we then calculate the Similarity Ratio (the percentage of matched SIFT features relative to the total number of SIFT features in images A and B). Finally, we classify the image pair (image A and image B) as either similar or different based on the Similarity Ratio. The higher the ratio is, the more similar the two images are. In our empirical analysis, we classify a pair of submissions as similar if the Similarity Ratio is greater or equal to 0.4.[4] Correspondingly, if a creator's new submission is very similar to any of her prior submissions (the similarity score between the two submissions is above 0.4), we classify the new submission as a "revision"; otherwise, we consider the new submission as a "redesign".

To facilitate our empirical analysis, we will divide the contest time horizon in the data into discrete intervals, or *periods.* (Details for discretizing the contest time horizon will be provided in Section 3.5.1.2.) Then based on a creator's submissions (there could be several) in a period being only "redesigns", only "revisions", or both "revisions" and "redesigns", we define her follow-up action within that period as *redesign*, *revise*, or

[4] We choose 0.4 as the cutoff for the main analysis because the distribution of the Similarity Ratio is roughly bimodal and 0.4 is its median. To ensure our results are robust to this cutoff choice, we estimate our model using a series of alternative cutoffs; see Section B.10.5.

do-both (*revise-and-redesign*), respectively. If she submitted nothing during the period, her action is *do-nothing*.[5]

3.3.2 Descriptive Regressions

Before constructing our structural model, we use regression to explore how disclosed ratings are associated with creators' participation behavior. We do not seek to derive causal inferences from this analysis; we seek only preliminary insights that we can later build upon using a more sophisticated and powerful structural analysis. We divide creators into "new entrants" and "incumbents", and test separately whether the number of new entrants or incumbents' follow-up actions are affected by the ratings that the seeker has disclosed by the end of the previous day. The details of the regression analysis are provided in Online Appendix B.2.1; highlights are provided below.

Descriptive Observation I – Feedback's Effect on Entries: A larger number of high ratings (5-star) disclosed in previous periods is associated with fewer entries, whereas a larger number of low ratings (1-star and 2-star) is associated with more entries. In addition, the number of existing participants is negatively correlated with the number of entries, which suggests that when a contest is already "crowded", potential entrants are discouraged from joining the contest.

Descriptive Observation II – Feedback's Effect on Incumbents' Follow-up Actions: Incumbents' own best rating is positively correlated with the probabilities of them choosing the *redesign*, *revise* and *do-both* actions (non-null actions), and this correlation is the highest with the *revise* action. Additionally, once we control for the focal incumbent's best rating, neither her average rating nor her second-best rating is significantly associated with any of the action probabilities. Further, when there are fewer creators or fewer existing submissions in the contest, and when there are more low ratings and fewer high ratings disclosed, the focal incumbent is more likely to follow-up with a non-null action.

The regression insights discussed above provide evidence for the correlation between performance feedback and the participation behavior of entrants and incumbents. However, reduced-form regressions alone are insufficient to address our research questions about the effect of performance feedback on contest outcomes. First, in the data there is not enough variation in the feedback disclosure policy to allow us to directly compare

[5] We do not granularize the actions further to capture the exact number of redesign and revision submissions in a period, which would increase the action space dramatically and render the model intractable without generating additional insight: In the data, there is not large variation in the number of designs a creator submits in each period, and submitting more revisions and/or more redesigns within one period does not significantly increase the quality improvement.

the outcome of contests with different feedback disclosure policies; this is because seekers in most contests follow the platform's suggestion and provide feedback throughout the entire contest. Second, from the regression results described above which pertain to individual creators' actions, it is difficult to infer what the overall contest-level impact would be from different feedback policies. For example, the above results suggest that revealing high ratings can encourage the authors of those highly-rated designs to improve upon their own submissions; but on the other hand, it discourages other participants' submission activities and entries. We cannot directly compare the magnitudes in the regression coefficients to see which effect dominates another.

Given the above, the approach we take is to build and estimate a structural model: We explicitly model the underlying decision-making mechanism, and estimate decision primitives which are likely to remain the same in different contest designs; doing so enables us to evaluate alternative feedback disclosure policies and measure their effects on contest dynamics and outcomes. The next section describes our structural model.

3.4 The Structural Model

We build a finite-horizon dynamic game model to capture creators' behavioral dynamics in a crowdsourcing design contest. Since in the data most seekers (¿70%) follow the platform's advice to provide full feedback (i.e., provide feedback throughout the contest), the model that we build for now assumes that seekers provide full feedback. In Section 3.5 we use this model to estimate parameters that govern creators' behaviors in design contests. We then use these estimates in Section 3.6 to run counterfactual simulations to study other feedback policies.

Our model divides time into periods indexed by t. In period t, the set of potential entrants who may choose to join the contest and the set of incumbents are denoted as M_t and N_t respectively.[6] We index potential entrants by j, and incumbents by i. We also attach superscript e to potential entrants' utility/response/etc. functions to distinguish them from incumbents' functions. In our model, the seeker provides his feedback at the end of each period; this reflects the reality that the seeker is not immediately able to give feedback on a one-for-one basis as each individual submission rolls in. The timing of the model is as follows (also illustrated in Figure 3.2):

1. The contest begins with the seeker posting a design request and announcing the award (R).

[6] We assume the pool of potential entrants is renewed every period.

2. In the first period ($t = 1$), potential entrants (M_1) arrive at the contest, and decide whether to join the contest by submitting their first design(s). Potential entrants' decision in each period (including the first period and periods $2, ..., T$) is a binary choice, denoted as $d_{j,t} \in \{enter, not\text{-}enter\} := \mathcal{D}$. The seeker evaluates the submitted designs, and at the end of the period, discloses the ratings. The rating is on the scale of 1-5 stars; if a design is not rated, we use "NA" to denote its rating.

3. At the beginning of each of the subsequent periods ($t = 2, ..., T$), incumbents (N_t) and potential entrants (M_t) observe all existing ratings. Based on this information, they make the following decisions simultaneously: potential entrants decide whether to join the contest ($d_{j,t}$ defined above), and incumbents choose their follow-up action $a_{i,t} \in \{do\text{-}nothing, redesign, revise, do\text{-}both\} := \mathcal{A}$, where, $a_{i,t} = do\text{-}nothing$, if incumbent i decides not to do anything in period t; $a_{i,t} = redesign$, if she creates one or more new designs that are significantly different from any of her existing designs; $a_{i,t} = revise$, if she submits one or more designs that are similar to one or more of her existing designs; $a_{i,t} = do\text{-}both$, if she both revises and redesigns.

4. In period $T + 1$, the creator of the best quality design wins the contest and receives the prize (R). In this terminal period, no entry is allowed, incumbents have no chance to take any action.[7]

Since the competition is quality-based and ratings reflect submissions' quality, existing ratings can be considered as state variables in our model, which not only capture the current status of the contest, affect creators' current and future utility, but also evolve as a function of creators' actions in each period. However, in reality, it is difficult for creators to track the ratings of each one of their own submissions and their competitors' submissions; as we learned from the descriptive regressions results, an incumbent's best rating significantly affects her follow-up actions, but once we control for the focal incumbent's best rating, neither her average rating nor her second-best rating is significantly associated with her follow-up actions. Moreover, incorporating all submissions' ratings and their evolution will make the model unmanageable. Hence, we define the individual-level state variable at time t (denoted as $x_{i,t}$) as the highest rating received by an incumbent i up to the beginning of period t.[8] Correspondingly,

[7] We add an arbitrary period $T+1$ at the end of the model to represent the time when the winner is announced.

[8] We use a creator's best rating observed on a 5-star scale as her individual-level state variable for

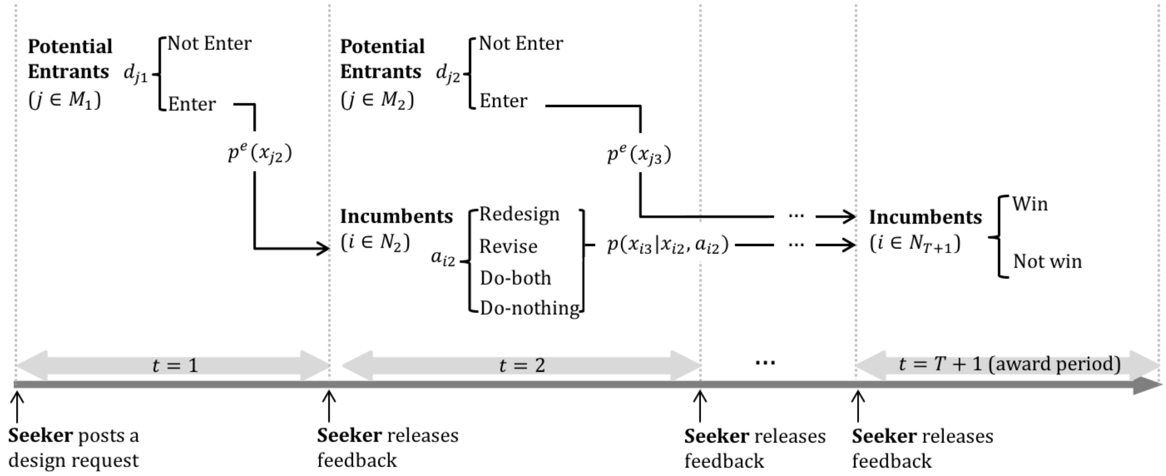


Figure 3.2: Timeline for Actions

we define the vector $s_t = \{s_t(x)\}_{x \in \{\text{NA},1,2,3,4,5\}}$ as the contest-level state variable, where $s_t(x)$ is the number of incumbents whose individual state takes value x in period t.

The state variables $x_{i,t}$ and s_t evolve as follows. A contest starts with zero incumbents and a contest state $s_1 = \{0, 0, 0, 0, 0, 0\}$. At the beginning of each period t where $t \in \{1, 2, ..., T\}$, potential entrants arrive and make entry decisions. If a potential entrant j enters the contest in period t, she will become an incumbent from $t + 1$, and the best rating her submitted design(s) receives in period t will become her individual state at the beginning of the next period ($x_{j,t+1}$), which is a random draw from the probability distribution $p^e(x_{j,t+1})$. For any incumbent i, her state variable $x_{i,t}$ evolves as a function of her action $a_{i,t}$. The action-specific transition probability of the state variable for an incumbent creator is then $p(x_{i,t+1}|x_{i,t}; a_{i,t})$. (Note that a creator's state transition is dependent on the contest-level state s_t, but we capture it through the creator's action choice $a_{i,t}$.[9]) The contest-level state variable, s_t, will evolve correspondingly, which can be expressed as $p(s_{t+1}|s_t; a_t, d_t)$, where a_t is the stack of all incumbents' actions in period t, and d_t is the stack of all potential entrants' entry decisions in period t. Note that $p^e(\cdot)$, $p(\cdot|\cdot)$, and s_t are common knowledge for all creators (Aguirregabiria and Mira, 2010).

now. One potential problem with this definition is that the ratings are truncated at 5. If a creator already has a 5-star design but revises her design, her best rating cannot be improved any further. To deal with this problem, we slightly alter the definition of the individual-level state variable, which will be explained later in Section 3.4.4.

[9] Specifically, creator i chooses her action $a_{i,t}$ upon seeing the opponent states. Given this action choice, we then model the *action-specific* transition probabilities as independent of opponent states (i.e., $p(x_{i,t+1}|x_{i,t}; s_t, a_{i,t}) = p(x_{i,t+1}|x_{i,t}; a_{i,t})$); empirical support for this assumption is provided in Online Appendix B.2.4 Table B.4.

3.4.1 Single Period Utility

As there are finitely many of periods in each contest, the per-period utility is t-dependent. We first explain the per-period utility in the terminal period $T+1$. In period $T+1$, no creator action is allowed; the creator with the best quality design is announced as the winner and receives award R; everyone else receives nothing. Given creator i's state $x_{i,T+1}$ and the contest-level state s_{T+1} from the seeker ratings, creator i's expected per-period utility in period $T+1$ can be expressed as

$$U_{i,T+1}(x_{i,T+1}, s_{T+1}) = \alpha R \cdot \Pr(i \text{ wins}|x_{i,T+1}, s_{T+1}), \tag{3.1}$$

where α is the marginal utility of money (or the number of utils a creator receives from getting an additional dollar of award). We will explain how to calculate $\Pr(i \text{ wins}|x_{i,T+1}, s_{T+1})$, the probability that creator i wins a contest, later in this section.

In periods $t = 1, ..., T$, incumbent i's per-period utility can be expressed as

$$U_{i,t}(\epsilon_{i,t}; a_{i,t}) = -c(a_{i,t}) + \epsilon_{i,t}(a_{i,t}),\ \ a_{i,t} \in \mathcal{A}, \tag{3.2}$$

where $c(a_{i,t})$ represents the cost associated with action $a_{i,t}$, and $\epsilon_{i,t}(a_{i,t})$ represents the individual-level choice-specific random shock. Likewise, potential entrant j's per-period utility can be expressed as

$$U^e_{j,t}(\varepsilon_{j,t}; d_{j,t}) = -c^e_t(d_{j,t}) + \varepsilon_{j,t}(d_{j,t}),\ \ d_{j,t} \in \mathcal{D}, \tag{3.3}$$

where $c^e_t(d_{j,t})$ represents the cost associated with action $d_{j,t}$ in period t. $c^e_t(d_{it} = \textit{enter})$ may include the costs of becoming aware of the contest, understanding the problem specifications, and coming up with the first design(s), and thus can be different across periods.[10] For normalization purposes, $c(a_{i,t} = \textit{do-nothing})$ and $c^e_t(d_{j,t} = \textit{not-enter})$ are assumed to be zero. The shocks $\epsilon_{i,t}(a_{i,t})$ $(\varepsilon_{j,t}(d_{j,t}))$ is private information observable to incumbent i (potential entrant j) in period t before she chooses which action to take, but are unobservable to other creators and the researchers. In addition, the shocks $\epsilon_{i,t}(a_{i,t})$ and $\varepsilon_{j,t}(d_{j,t})$ are assumed to be zero-mean, and *i.i.d.* with respect to individual, time and choice. $\epsilon_{i,t} = \{\epsilon_{i,t}(a_{i,t})|a_{i,t} \in \mathcal{A}\}$ follows a distribution whose probability density function is p_ϵ, and $E_{i,t}$ is the corresponding state space; $\varepsilon_{j,t} = \{\varepsilon_{j,t}(d_{j,t})|d_{j,t} \in \mathcal{D}\}$ follows a distribution whose probability density function is p_ε, and

[10] Notice that we assume the cost associated with incumbents' actions $(c(a_{i,t}))$ to be time invariant, but allow entry cost $(c^e_t(d_{j,t}))$ to vary across periods, due to the fact that the crowdsourcing platform displays contests nearing their end higher on the contest list webpage, and as a result the cost of discovering a contest may decrease over time.

$\mathcal{E}_{j,t}$ is the corresponding state space.

3.4.2 Creators' Decisions

In every period t, incumbents decide on follow-up actions, and potential entrants make entry decisions. Creators are forward-looking and take into account the implications of their decisions on future utilities and on the expected future reaction of competitors. Specifically, creators make these decisions to maximize expected discounted lifetime utility.

We assume creators play Markov strategies.[11] Formally, a Markov strategy for incumbent i in period t is a function $\rho_{i,t} : X_{i,t} \times S_t \times E_{i,t} \to \mathcal{A}$. Likewise, a Markov strategy for potential entrant j in period t is a function $\lambda_{j,t} : S_t \times \mathcal{E}_{j,t} \to \mathcal{D}$. Let $\sigma_t = \{\rho_t, \lambda_t\}$ summarize all existing incumbents' and potential entrants' strategies in period t, where $\rho_t = \{\rho_{i,t}\}_{i \in N_t}$ and $\lambda_t = \{\lambda_{j,t}\}_{j \in M_t}$. $\sigma = \{\sigma_t\}_{t=1,...,T}$ then summarizes all periods' strategies. Note that the strategies are time-varying because the contest has a finite horizon. Let $V^{\sigma}_{i,t}(x_{i,t}, s_t, \epsilon_{i,t})$ represent the value function for an incumbent i given that the other creators behave according to their respective strategies in σ, and given that incumbent i uses her best response strategy. That is,

$$V^{\sigma}_{i,t}(x_{i,t}, s_t, \epsilon_{i,t}) = \max_{a_{i,t}} \mathbb{E}\left\{ \left[\sum_{\tau=t}^{T} \beta^{\tau-t} U_{i,\tau}(\epsilon_{i,\tau}; a_{i,\tau}) + \beta^{T+1-t} U_{i,T+1}(x_{i,T+1}, s_{T+1}) \right] \middle| x_{i,t}, s_t, \epsilon_{i,t}; \sigma \right\}. \tag{3.4}$$

The expectation is taken over current-period private shocks of other contestants, as well as future values of the state variables and private shocks.[12] According to Bellman's principle of optimality, we can write $V^{\sigma}_{i,t}(x_{i,t}, s_t, \epsilon_{i,t})$ recursively as:

$$V^{\sigma}_{i,t}(x_{i,t}, s_t, \epsilon_{i,t}) = \begin{cases} \max_{a_{i,t}} \left\{ U_{i,t}(\epsilon_{i,t}; a_{i,t}) + \beta \mathbb{E}[V^{\sigma}_{i,t+1}(x_{i,t+1}, s_{t+1}, \epsilon_{i,t+1}) | a_{i,t}] \right\} & \text{if } t \in 1, 2, ..., T, \\ U_{i,T+1}(x_{i,T+1}, s_{T+1}) & \text{if } t = T+1; \end{cases} \tag{3.5}$$

[11] That is, potential entrant j's decision depends only on the current contest-level state s_t and her current private utility shock $\varepsilon_{j,t}$; incumbent i's behavior depends only on the current contest-level state s_t, her current own state $x_{i,t}$ and her current private utility shock $\epsilon_{i,t}$.

[12] These value functions and their corresponding strategies are also dependent on the award level. As we will discuss in the next section, the variation in the award level enables the identification of α. In this section, we suppress the award level (R) and focus on how creators make decisions under a given award level.

and for potential entrants (in $t = 1, ..., T$),

$$V^{e,\sigma}_{j,t}(s_t, \varepsilon_{j,t}) = \max_{d_{j,t}} \Big\{ U^e_{j,t}(\varepsilon_{j,t}; d_{j,t}) + \mathbb{I}(d_{j,t} = Enter) \cdot \beta \mathbb{E} V^{\sigma}_{j,t+1}(x_{j,t+1}, s_{t+1}, \epsilon_{j,t+1}) \Big\}. \tag{3.6}$$

The expectations in Equations (3.5) and (3.6) are taken over other creators' actions in the current period, values of the next period individual- and contest-state variables, as well as the next period private shocks.

3.4.3 Equilibrium Concept

We solve this finite-horizon dynamic discrete game with private information for a Markov Perfect Equilibrium (MPE) in type-symmetric pure strategies. For the proposed structural model, a strategy $\sigma^* = \{\rho^*, \lambda^*\}$ represents a MPE if, at any t, given everyone else is playing σ^*, an incumbent's and a potential entrant's best response are ρ^* and λ^*, respectively.

Following Milgrom and Weber (1985), we represent a MPE in probability space. We denote the *conditional choice probability* (*CCP*) corresponding to the MPE strategy σ^* as $\mathbf{P}^*$, in which the t^{th} element, P^*_t, characterizes creator strategies in the t^{th} period. Equilibrium probabilities are a fixed point $\mathbf{P}^* = \mathbf{\Gamma}(\mathbf{P}^*)$, where the function $\mathbf{\Gamma}$ is the creators' *best response probability function*. Assuming both the incumbents' private shock ($\epsilon_{i,t}$) and potential entrants' private shock ($\varepsilon_{j,t}$) follow the Type I extreme value distribution (Rust, 1987), we then get that in equilibrium incumbent i follows

$$\text{P}^*_{i,t}(a_{i,t}|x_{i,t}, s_t) = \Gamma_{i,t}(a_{i,t}|x_{i,t}, s_t; \mathbf{P}^* \setminus \text{P}^*_{i,t}) = \frac{\exp(v^{\mathbf{P}^*}_{i,t}(x_{i,t}, s_t; a_{i,t}))}{\sum_{a'_{i,t}} \exp(v^{\mathbf{P}^*}_{i,t}(x_{i,t}, s_t; a'_{i,t}))}, \tag{3.7}$$

and potential entrant j follows

$$\text{P}^*_{j,t}(d_{j,t}|s_t) = \Gamma^e_{j,t}(d_{j,t}|s_t; \mathbf{P}^* \setminus \text{P}^*_{j,t}) = \frac{\exp(v^{e,\mathbf{P}^*}_{j,t}(s_t; d_{j,t}))}{\sum_{d'_{j,t}} \exp(v^{e,\mathbf{P}^*}_{j,t}(s_t; d'_{j,t}))}, \tag{3.8}$$

where $v^{\mathbf{P}^*}_{i,t}$ and $v^{e,\mathbf{P}^*}_{j,t}$ are incumbent and potential entrants' *choice specific value functions*: $v^{\mathbf{P}^*}_{i,t}(x_{i,t}, s_t; a_{i,t}) = -c(a_{i,t}) + \beta \mathbb{E}[\tilde{V}^{\mathbf{P}^*}_{i,t+1}(x_{i,t+1}, s_{t+1}|a_{i,t})]$; $v^{e,\mathbf{P}^*}_{j,t}(s_t; d_{j,t}) = -c^e(d_{j,t}) + \mathbb{I}(d_{j,t} = Enter) \cdot \beta \mathbb{E}[\tilde{V}^{\mathbf{P}^*}_{j,t+1}(x_{j,t+1}, s_{t+1})|d_{j,t}]$, in which $\tilde{V}^{\mathbf{P}}_{i,t+1}$ is the *ex ante value function*: $\tilde{V}^{\mathbf{P}}_{i,t+1}(x_{i,t+1}, s_{t+1}) := \int V^{\sigma}_{i,t+1}(x_{i,t+1}, s_{t+1}, \epsilon_{i,t+1}) p_\epsilon(\epsilon_{i,t+1}) d\epsilon_{i,t+1}$. The derivation details are provided in Online Appendix B.3.

3.4.4 Winning Probability

The last element of our structural model to be explained is how creator i's winning probability ($\Pr(i \text{ wins}|x_{i,T+1}, s_{T+1})$) in period $T+1$ is calculated. Generally speaking, the creator who submits the highest quality design wins. So far, the quality of a design has been measured by its rating, which is observed in our data. One potential problem with this measure is that the observed ratings are truncated at 5-star – no matter how high a design's "true" quality is, the seeker can at most rate it as 5-star. If we neglect this problem, our model would fail to capture some important aspects of creator behavior observed in the data. For example, the model would predict that a creator whose current best rating is 5-star would have no incentive to make any additional submissions, as doing so only incurs cost but provides no benefit. However, in the data, we observe many cases where creators make additional submissions after receiving a 5-star rating; moreover, among creators who have received a 5-star rating, those who remain active in revising their design(s) or submitting new design(s) have a higher probability of winning the contest than those who become inactive (details are presented in Online Appendix B.5).

To resolve this problem and recover the underlying "true" quality (which we denote as $x_{i,t}^{true}$) of a creator's best design, we redefine the individual-level state as $\tilde{x}_{i,t} = (x_{i,t}, \boldsymbol{\varpi}_{i,t})$, with $x_{i,t}$ still denoting the best rating that creator i has received, and an additional variable $\boldsymbol{\varpi}_{i,t}$ recording the actions creator i has taken up to period t *after* receiving her first 5-star rating, if she has ever received any 5-star rating. $\boldsymbol{\varpi}_{i,t}$ is a vector of three elements, respectively representing the number of times that the *"redesign"*, *"revise"* or *"do-both"* action has been taken by creator i since she received her first 5-star rating (excluding the action that results in the first 5-star rating). There are then two scenarios:

1. When $x_{i,t} \in \{\text{NA}, 1, 2, 3, 4\}$, i.e., creator i's current best score has not reached 5-star:
   $\tilde{x}_{i,t} = (x_{i,t}, \boldsymbol{\varpi}_{i,t})$, where $\boldsymbol{\varpi}_{i,t} = (0, 0, 0)$;

2. When $x_{i,\tau+1} = ... = x_{i,t} = 5$, $x_{i,\tau} < 5$, i.e., creator i has her first 5-star in period $\tau + 1$: $\tilde{x}_{i,t} = (5, \boldsymbol{\varpi}_{i,t})$, where $\boldsymbol{\varpi}_{i,t} = (\varpi_{i,t,\text{redesign}}, \varpi_{i,t,\text{revise}}, \varpi_{i,t,\text{do-both}})$, with $\varpi_{i,t,a} = \sum_{l=\tau+1}^{t} \mathbb{I}(a_{i,l} = a)$.

The contest-level state is then redefined as $\tilde{s}_t$, which summarizes the number of creators whose individual-level state is $\tilde{x}_{i,t}$ for all possible values of $\tilde{x}_{i,t}$. $\tilde{x}_{i,t}$ transitions in the following way: $x_{i,t}$ still transitions in the same way as discussed in the previous subsection; $\boldsymbol{\varpi}_{it}$ transitions deterministically as a function of the action creator i takes in period t: $\varpi_{i,t+1,a} = \varpi_{i,t,a} + \mathbb{I}(a_{i,t} = a)$, where $a \in \{redesign, revise, do\text{-}both\}$. With this newly-constructed state variable $\tilde{x}_{i,t}$, we can recover the true quality of a creator's best design ($x_{i,t}^{true}$). We assume that the first time creator i receives a 5-star rating, the true quality of that 5-star design is $5 + \max(0, \xi_0)$, where $\xi_0 \in \mathbb{R}$ and $\max(0, \xi_0)$ is the part of quality that gets truncated by the integer rating of 5. After receiving a 5-star rating, every time a creator takes a non-null action $a_{i,t} \in \{redesign, revise, do\text{-}both\}$, there will be a corresponding stochastic improvement ($\max(0, \xi_{a_{i,t}})$), where $\xi_{a_{i,t}} \in \mathbb{R}$. Hence, the mapping between a creator's state variable ($\tilde{x}_{i,t} = (x_{i,t}, \boldsymbol{\varpi}_{i,t})$) and the quality of her best design ($x_{i,t}^{true}$) is as follows:

$$x_{i,t}^{true} = x_{i,t} \ \text{, if } x_{i,t} < 5; \tag{3.9a}$$

$$x_{i,t}^{true} = 5 + \max(0, \xi_0) + \sum_{a \in \{redesign, revise, do\text{-}both\}} \sum_{n=1}^{\varpi_{i,t,a}} \max(0, \xi_a^{(n)}) \ \text{, if } x_{i,t} = 5. \tag{3.9b}$$

where $\max(0, \xi_a^{(n)})$ is the n^{th} realization of the stochastic improvement, with the max operator reflecting the fact that the focal creator's new submissions do not replace her existing submissions so the quality rating of her best design can stay the same or increase, but never decreases.

We can now compute the probability of winning using the "true" quality in the terminal period $T+1$. Let $\bar{x}_{T+1}^{true} = \max_{i \in N_{T+1}}(x_{i,T+1}^{true})$ denote the maximum quality achieved by all creators in the game in period $T+1$, then creator i's winning probability can be expressed as:

$$\Pr(i \text{ wins} | \{x_{k,T+1}^{true}\}_{\forall k \in N_{T+1}}) = \mathbb{I}(x_{i,T+1}^{true} = \bar{x}_{T+1}^{true}) \frac{1}{\sum_{k \in N_{T+1}} \mathbb{I}(x_{k,T+1}^{true} = \bar{x}_{T+1}^{true})}. \tag{3.10}$$

Notice that Equation (3.10) implies two cases. (i) When $\bar{x}_{T+1}^{true} > 5$, $\bar{x}_{T+1}^{true}$ is a continuous variable. In this case, a tie is impossible – the designer of the highest-quality design wins the contest; (ii) when $\bar{x}_{T+1}^{true} \leq 5$, ties are possible. In case of ties, each creator whose best rating is $\bar{x}_{T+1}^{true}$ wins the contest with an equal chance.[13] In the data, the majority of the contests fall into the case (i).

[13] Here we assume in cases of ties, each creator has an equal chance, rather than each submission has an equal chance, to win the contest. The reason is that, when a creator decides to revise her existing submission(s), she typically picks her best performing style to revise; when a creator decides to redesign, she typically creates a new design style and abandons old design styles if this new design style is rated higher. Therefore, in period $T+1$, we effectively assume that each creator has one design, i.e., the best among her submissions, in the seeker's consideration set.

In the next section we describe our estimation strategy and results. We estimate non-parametrically all transition probabilities of the individual-level state variable ($x_{i,t}^{true}$) *within 1- to 4-star*. By contrast, we use the particular functional form in Equation (3.9b) only for estimating how $x_{i,t}^{true}$ transitions *after the creator gets a 5-star rating* given $\varpi_{i,t}$; Online Appendix B.5 discusses robustness of our results to alternative versions of Equation (3.9b).

3.5 Estimation and Results

3.5.1 Estimation Strategy

Our approach to estimate the structural model proceeds in two steps. In step 1, we estimate the set of parameters that describe the *action-specific* state transition process (θ_1). In step 2, we embed the estimated action-specific state transition parameters into the dynamic discrete game of creators' entry and follow-up actions, and estimate the parameters in creators' utility function, denoted by θ_2. These two sets of parameters can be estimated separately under the conditional independence assumption.[14,15] Intuitively, the action-specific state transition probabilities describe how a creator's best design quality improves conditional on the action she takes, which does not depend on the cost parameters, because the action is already taken.

3.5.1.1 Step 1: Estimating Parameters for Action-Specific State Transitions (θ_1)

We estimate the *rating* transition probabilities $p^e(x_{j,t+1})$ and $p(x_{i,t+1}|x_{i,t}, a_{i,t})$ using the frequency estimator (denoted as θ_{11}). Since $x_{i,t}^{true} = x_{i,t}$ when $x_{i,t} < 5$, $p^e(x_{j,t+1}^{true}) = p^e(x_{j,t+1})$ when $x_{j,t+1} < 5$, and $p(x_{i,t+1}^{true}|x_{i,t}^{true}, a_{i,t}) = p(x_{i,t+1}|x_{i,t}, a_{i,t})$ when $x_{i,t+1} < 5$.

[14]Under the conditional independence assumption (i.e., $p(x_{i,t+1}|x_{i,t}; a_{i,t})$ and ξ's are independent of $p_\epsilon(\epsilon_{i,t})$, $p^e(x_{j,t+1})$ and ξ's are independent of $p_\varepsilon(\varepsilon_{i,t})$), the transition probability functions $p(x_{i,t+1}|x_{i,t}; a_{i,t})$, $p^e(x_{j,t+1})$, and ξ's can be estimated from the transition data using a standard maximum likelihood method without solving for the game equilibrium; further discussion on this assumption is included in Online Appendix B.4.

[15]In the main estimation, we set $\beta = 0.9$, the number of potential entrants $|M_t| = 300, \forall t$ to get results in Table 3.2. β's identification is known to be impractical (Rust, 1987), so we do not intend to estimate the discount factor. For the number of potential entrants, it cannot be identified together with entry costs. Therefore, we fix it to be the number of total active creators on the platform at any given time – the average is around 300. We conducted sensitivity analyses on β and $|M_t|$, and find that the qualitative nature of our estimation results do not change; see Section B.10.5.

It is more challenging to estimate the parameters governing the unobserved quality that gets truncated at 5-star when a creator receives her first 5-star rating and makes further submissions, because we do not directly observe the quality state beyond 5-star. Yet, we can infer the distribution of the quality state from the observed probability of winning (Pr(i win)) given the combination of actions each creator takes after receiving her first 5-star rating, as follows. As discussed in Section 3.4.4 we introduce a series of random variables, $\xi = \{\xi_0, \xi_{redesign}, \xi_{revise}, \xi_{\textit{do-both}}\}$ to account for the rating system's truncation at 5 stars. Here we further parametrize the unobserved quality transition process and assume that $\xi.$ follows a normal distribution, $\xi. \sim N(\mu., \sigma.^2)$. The parameter vector $\{\mu_0, \sigma_0^2, \mu_{redesign}, \sigma_{redesign}^2, \mu_{revise}, \sigma_{revise}^2, \mu_{\textit{do-both}}, \sigma_{\textit{do-both}}^2\} := \theta_{12}$ can then be estimated by maximizing the following likelihood:

$$L_2(\theta_{12}) = \prod_{q=1}^{Q} \prod_{i=1}^{||N_{q,T+1}||} \Big\{ \Pr(i \text{ wins} | \{\tilde{x}_{k,T+1}\}_{k \in N_{q,T+1}}; \theta_{12})^{\mathbb{I}(i \text{ wins})} \\ [1 - \Pr(i \text{ wins} | \{\tilde{x}_{k,T+1}\}_{k \in N_{q,T+1}}; \theta_{12})]^{[1-\mathbb{I}(i \text{ wins})]} \Big\}, \quad (3.11)$$

where Q represents the number of contests in our estimation sample, and $||N_{q,T+1}||$ is the total number of creators that submitted designs to contest q.

3.5.1.2 Step 2: Estimating Parameters for Creators' Utility Function (θ_2)

Once we obtain a consistent estimate of $\theta_1 = \{\theta_{11}, \theta_{12}\}$, denoted as $\hat{\theta}_1$, we can plug these estimated state transition probabilities into the dynamic discrete game model and estimate the parameters in creators' utility function. We use the following notation: $c(a_{i,t}{=}a) := c_a$, $a \in$ {re-design, revision, and do-both} , and $c_t^e(d_{j,t}{=}enter) := c_t^e$, for $t \leq T$. These six parameters – $c_{\text{re-design}}$, c_{revision}, $c_{\text{do-both}}$, c_1^e, c_2^e, c_3^e – along with α, are the parameters in creators' utility function (θ_2) to be estimated. At a high-level, our step 2 estimation procedure contains four sub-steps:

(2.a) Fix a candidate vector of parameters θ_2, and start the numerical search on θ_2's parameter space.

(2.b) Given this candidate parameter vector, solve the dynamic game for the Markov Perfect Equilibrium (MPE) and compute the creators' conditional choice probabilities in MPE.

(2.c) Based on the equilibrium choice probabilities from the previous step, we compute the log likelihood of observing the actual creator *entry/revision/redesign/do-both* choices in the data.

(2.d) Using the nested fixed-point (NFXP) algorithm, iterate the first three steps in the parameter space, and obtain the maximum simulated likelihood (MSL) estimate of the parameters, $\hat{\theta}_2$, which best rationalizes the observed creator choices.

In step (2.b), we solve the dynamic game for the MPE, using the "best response mapping (BRM)" method, which solves an inter-related system of dynamic programs for the strategic interactions among creators: Given all other creator' strategies (in the form of conditional choice probabilities), we compute each creator's most favorable strategy; this process iterates, until a fixed point is found. This fixed point is a representation of MPE in the space of creator choice probabilities. (Online Appendix B.6 contains details about the BRM algorithm.)

Using BRM for a contest with a large number of players involves solving dynamic programs in an extremely large state space, which grows exponentially with the number of periods.[16] To ensure tractability we divide each 7-day contest into three periods – Period 1 (Days 1-2), Period 2 (Days 3-5), and Period 3 (Days 6-7).[17] In a so-defined three-period contest, the seeker can give feedback at the end of the first and the second periods (the feedback after the last period will not be able to affect creator behaviors). Dividing contests into three periods significantly reduces the computational burden, yet still allows us to capture not only the full feedback and the no feedback scenarios, but also the early feedback and the late feedback scenarios. However, the number of $(x_{i,t}, s_t)$ combinations is still extremely large in a three-period contest (see Footnote 16). To further reduce the computational burden, we employ Keane and Wolpin (1994) (KW) interpolation to expedite the calculation of BRM.[18]

For a candidate vector θ_2, we denote the equilibrium creator choice probabilities computed in step (2.b) as $\Pr(d_{jt}|s_t;\theta_2,\hat{\theta}_1)$ $(d_{jt} \in \mathcal{D})$ for potential entrant j and $\Pr(a_{it}|x_{it},s_t;\theta_2,\hat{\theta}_1)$ $(a_{it} \in \mathcal{A})$ for incumbent i. In step (2.c), the likelihood of observing m_{qt} out of $||M_{qt}||$ potential entrants joining the contest q in period t is: $f^e(m_{qt}|s_{qt};\theta_2,\hat{\theta}_1) = \binom{||M_{qt}||}{m_{qt}}\Pr(d_{qt} = enter|s_{qt};\theta_2,\hat{\theta}_1)^{m_{qt}}(1 - \Pr(d_{qt} =$

[16] For example, in a three-period contest with 30 participants, the number of possible values for $x_{i,t}$ is $5 + \binom{3-1+4-1}{4-1} = 15$, and the number of possible combinations of $s_{t=3}$ is therefore $\binom{30+15-1}{15-1} = 1.15 \times 10^{11}$; whereas in a seven-period contest with 30 participants, the number of possible values for $x_{i,t}$ is $5+\binom{7-1+4-1}{4-1} = 89$, and the number of possible combinations of $s_{t=7}$ is $\binom{30+89-1}{89-1} = 9.52 \times 10^{27}$.

[17] We also try two alternative ways of dividing a contest into three periods: (i) Period 1 = Days 1-3, Period 2 = Days 4-5, and Period 3 = Days 6-7; (ii) Period 1 = Days 1-2, Period 2 = Days 3-4, and Period 3 = Days 5-7. The observed conditional choice probabilities and choice-specific state transition probabilities exhibit little sensitivity to how we define the periods.

[18] KW preserves the strategic interactions among creators in the dynamic game; please refer to Online Appendix B.6.2 for details.

$enter|s_{qt};\theta_2,\hat{\theta}_1))^{||M_{qt}||-m_{qt}}$,[19] and the likelihood of an incumbent i choosing follow-up action $a_{qit} \in \mathcal{A}$ in period t of contest q is: $f(a_{qit}|x_{qit}, s_{qt};\theta_2,\hat{\theta}_1) = \prod_{a\in\mathcal{A}} \Pr(a_{qit} = a|x_{qit}, s_{qt};\theta_2,\hat{\theta}_1)^{\mathbb{I}(a_{qit}=a)}$. The joint likelihood to be maximized in step (2.d) is then:

$$L_1(\theta_2,\hat{\theta}_1) = \prod_{q=1}^{Q}\prod_{t=1}^{T}\left[f^e(m_{qt}|s_{qt};\theta_2,\hat{\theta}_1)\cdot\prod_{i=1}^{||N_{qt}||} f(a_{qit}|x_{qit}, s_{qt};\theta_2,\hat{\theta}_1)\right], \tag{3.12}$$

where $||N_{qt}||$ is the number of incumbents in contest q up to period t. The maximum simulated likelihood estimator of θ_2 is $\hat{\theta}_2 =_{\theta_2} \log L_1(\theta_2,\hat{\theta}_1)$.

3.5.1.3 Identification

We discuss the sources of identification. For step 1, θ_{11} is identified directly from the frequencies observed in the data; θ_{12} is identified from the variation in the winning probability resulting from different combinations of follow-up actions taken by creators after receiving their first 5-star rating (details are provided in Online Appendix B.7.4). These state transition processes, together with variations in award levels across contests and the observed entry and follow-up action choices in the panel data of creator activities, constitute the inputs for identifying the utility parameters in the second step.

For step 2, we discuss identification of the parameters in creators' utility functions (θ_2), starting from how the cost parameters are identified given a specific value of αR (the product of α and R). For simplicity, we suppress in the utility and value functions the dependence of these functions on αR for now. We can rewrite incumbents' (potential entrants') per-period utility function as the sum of a deterministic component and a random component, i.e., $U_{i,t}(a_{i,t},\epsilon_{i,t}) = u(a_{i,t}) + \epsilon_{i,t}(a_{i,t})$ ($U^e_{j,t}(d_{i,t},\varepsilon_{j,t}) = u^e(d_{j,t}) + \varepsilon_{j,t}(d_{j,t})$). Our model does not contain creator-specific cost-shifters. Initially it may seem that our model is under-identified, as cost-/payoff-shifters are usually needed when estimating a dynamic discrete game with a generic period payoff function $u_{i,t}(a_{i,t}, a_{-i,t}, s_t) + \epsilon(a_{i,t})$ (e.g., Bajari et al. (2010), Bajari et al. (2015)). However, in our setting, opponents' actions (new entries and incumbents' follow-up actions) only affect the evolution of their own state and the contest-level state and do not directly enter the focal creator's current period utility. This is because, prior to the rewarding stage $T+1$, the single-period utility is just the negative cost of the action plus the individual-level choice-specific random shock; the award only occurs in

[19] We cannot observe each potential new entrant j's decision; rather, we can only observe the number of potential entrants who join the contest in each period.

the terminal stage $T+1$ (at which time there are no new actions taken). Formally, $u_{i,t}(a_{i,t}, a_{-i,t}, s_t) = u_{i,t}(a_{i,t}, s_t)$; see Equations (3.2)-(3.3). Per Section 3.5 of Bajari et al. (2015) (part of the argument leading up to their Theorem 2, appropriately adapted for a finite-horizon game), we can identify $u_{i,t}(a_{i,t}, s_t)$, as long as the conditional choice probabilities $\mathrm{P}_{i,t'}(a_{i,t'}|s_{t'})$ ($t' \geq t$) are observed in the data. Thus, we do not need cost-shifters to identify the model.

In brief, identification of θ_2 is achieved as follows. (Please see Online Appendix B.7 the formal identification proof.) For the award period, we observe in the data the realized s_{T+1} and $x_{i,T+1}$, and who wins the contest. With this information, we can calculate the value function of the very last period $\tilde{V}_{i,T+1}(x_{i,T+1}, s_{T+1})$ (in particular, $\tilde{V}_{i,T+1}(x_{i,T+1}, s_{T+1}) = \alpha R \cdot \Pr(i \text{ winning in period T+1}|x_{i,T+1}, s_{T+1})$). With the last-period value function calculated, we can start performing backward induction to identify the per-period utility for periods $t \leq T$, leveraging the fact that this utility can be decoupled from the (discounted) future period value function. Once current-period utility is identified, we can further identify the costs of actions, because in our setting the current period utility ($u(x_{i,t}, s_t; a_{i,t} = k)$) is only a function of i's own action $a_{i,t}$, but not opponents' actions $a_{-i,t}$. (Note that opponents' actions $a_{-i,t}$ only enter the discounted expected future period value function by influencing the opponents' state transition.) Finally, the identification of α is enabled by variation in the award level in the data. Creators' behavior (more specifically their conditional choice probabilities) varies with the award amount (R), and the degree of this variation helps us identify α. Intuitively, if in the data, the probability of entry and the probability of follow-up submissions are not sensitive to changes in the award level, then α is small; otherwise, α is large.

3.5.2 Estimation Results

Estimates of Parameters for State Transitions (θ_1).

Here we highlight a few observations of the state transition probability estimates (relegated to Online Appendix B.8 to save space in the body). As might be expected, the state transition probability estimates reveal that *do-both* is the most effective in improving individuals' highest design rating, leading to on average larger improvements than *redesign* and *revision* do; however, the variation of the improvements from *do-both* is also high. In addition, for creators with low ratings (below or equal to 3-Star), *redesign* results in on average bigger, but more variable improvements than *revision.* For creators who have already received relatively high ratings (4-Star or 5-star), *revision* leads to on average slightly larger improvements than *redesign*.

Estimates of Parameters for Creators' Utility Function (θ_2).

Table 3.2 reports the estimation results for the parameters in creators' utility function. The estimates are generally consistent with intuition. Regarding period-specific entry costs, we have two observations. First, the estimated entry costs are always higher than costs associated with incumbents' follow-up actions. This is expected, because in addition to the effort required to submit a design, the entrant also spends effort discovering the contest, understanding the problem specification, etc. Second, the entry cost is decreasing over time. One plausible explanation is that, by default, contests that are closing soon rank higher on the site's contest list. A higher position makes it easier for creators to discover the contest, which reduces one component of the entry cost discussed above.

As might be expected, among incumbents' follow-up actions, the estimated cost associated with the *redesign* action is higher than the estimated cost associated with the *revise* action. Also, the cost of the *do-both* action is higher than the cost of *redesign* only or *revise* only, but lower than the sum of doing both separately. Notice that these cost numbers are not "per-submission costs"; rather, the costs should be interpreted as the cost of employing only the exploration strategy, only the exploitation strategy, and both strategies in one period. Each strategy can possibly involve multiple submissions. Finally, the marginal utility of money (α), or the number of utils a creator receives from getting an additional dollar of award, is estimated to be 0.034.

In summary, the parameter estimates show some interesting patterns. They are also generally aligned with our intuition, which could serve as a check to make sure our model generates reasonable results. In Section 3.6 we use the estimated structural model to perform counterfactual simulations, to address our key research questions about the effect of performance feedback on contest outcomes.

Table 3.2: Estimates for θ_2, Parameters in Creators' Utility Function

Parameter	Description	Estimate	Standard Error
α	Marginal Utility of Money	0.034	(0.002)
$c(a_{i,t} = \textit{redesign})$	Cost of Redesign	2.790	(0.010)
$c(a_{i,t} = \textit{revise})$	Cost of Revision	2.205	(0.021)
$c(a_{i,t} = \textit{do-both})$	Cost of Doing Both	3.319	(0.031)
$c_1^e(d_{i,t} = \textit{enter})$	Period 1 Entry Cost	4.958	(0.028)
$c_2^e(d_{i,t} = \textit{enter})$	Period 2 Entry Cost	4.326	(0.043)
$c_3^e(d_{i,t} = \textit{enter})$	Period 3 Entry Cost	3.599	(0.014)

Note: The numbers in parenthesis are standard errors, calculated with bootstrapping.

3.5.3 Model Performance

Below we show the performance of our model in predicting the overall probabilities of potential entrants' entry and incumbents' taking each follow-up action, and the contest outcomes. Such aggregate measures are commonly used to demostrate the performance of structural econometric models, especially for dynamic game models (e.g., Aksin et al. (2013); Collard-Wexler (2013); Takahashi (2015); Igami (2017)).

Cross-Validation on Contestant Actions: We partition the data randomly into four subsets and perform four rounds of cross-validation for all actions: *new entry*, *revise*, *redesign*, *do-both*, and *do-nothing* (Online Appendix B.9 details the cross-validation process). Table 3.3 reports prediction errors averaged across the four rounds. These errors are small, suggesting that our model performs well in predicting individuals' actions. (Note that because the absolute frequency of *redesign* actions is small, a small prediction error is magnified by the smaller denominator in the calculation of the relative error.)

Out-of-Sample Prediction of Contestant Actions: Next, we perform an out-of-sample test to illustrate the accuracy of the estimated model in predicting creators actions in a hold-out sample. We consider a hold-out sample of 418 contests and 11,578 contest-creator combinations.[20] We use the hold-out sample to test the out-of-sample prediction accuracy of the estimated model, and report the results for different actions in Table 3.4. As can be seen in table, the model estimated using the in-sample data produces fairly accurate predictions for the actions observed in the hold-out data – the values of the relative errors and absolute errors are small. Note that because the absolute frequency of *redesign* actions is small, a small absolute prediction error can lead to a large relative error (due to the small denominator). Hence, although the relative error may seem high in the case of the *redesign* action, the corresponding absolute error is small.

Performance on Predicting Contest Outcome: Apart from evaluating our model performance in predicting creators' actions, it is crucial to ensure our model performs well in predicting the contest outcomes, because the purpose of this paper is to shed light on how seekers should provide performance feedback to achieve better contest outcomes (e.g., more high-quality design submissions, a higher best-design quality, etc.). To test the model performance in predicting contest outcomes, we use the same hold-out sample as in the previous out-of-sample prediction analysis of creator actions.

[20]The contests in the hold-out sample are similar to those in the sample used for the estimation in terms of the award level, and are all with guaranteed award, single-winner, public and 7 days long. Additionally, the seeker in all the 418 contests provided feedback throughout the contest horizon.

We simulate the competition dynamics and contest-level outcomes for the 418 hold-out contests using the estimated structural model. For each contest, we draw 50,000 sample paths, and calculate the average numbers of creators whose highest rating at the end of the contest is NA, 1-star,..., and 5-star, respectively, across the 50,000 simulations, as the "simulated" numbers of NA, 1-star,..., and 5-star creators for that contest. In Figure 3.3, black columns represent the average simulated number of creators whose best rating is NA, 1-star,..., or 5-star at the end of each contest across the 418 contests. The grey columns represent the actual average numbers of creators whose best rating is NA, 1-star,..., and 5-star across the 418 contests observed in the data. Figure 3.3 shows that the actual numbers observed in the data closely match the simulated numbers. Thus, we conclude that our model performs well in predicting contest outcomes in the holdout sample.

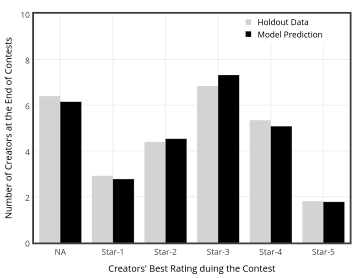


Figure 3.3: Model Performance

Table 3.3: Performance of Cross-Validation for All Actions

	Incumbent				Entrant
	Do-nothing	Redesign	Revision	Do-both	Entry
Relative Err	0.56%	9.85%	2.91%	3.40%	2.66%
Absolute Err	0.43%	0.60%	0.34%	0.17%	0.08%

Table 3.4: Out-of-Sample Testing of the Model With All Actions

	Incumbent				Entrant
	Do-nothing	Redesign	Revision	Do-both	Entry
Relative Err	1.13%	9.22%	11.68%	4.54%	5.53%
Absolute Err	0.87%	0.63%	1.33%	0.24%	0.17%

3.5.4 Robustness Checks

We designed our main model to be parsimonious but effective. To ensure that our empirical results are robust to our modeling choices and assumptions, we conduct a series of robustness checks, including: (1) considering and estimating two alternative "myopic" models of creator behavior; (2) considering and estimating an alternative model that accounts for potential entrants' strategic waiting behavior; (3) providing a stratified sample test and reduced-form evidence to support our assumption that creators are ex-ante homogeneous (in particular, we consider two most likely types of creator heterogeneity — heterogeneity in creator ability/experience and interest in the contest); (4) re-estimating the model allowing creators to receive a non-monetary reward from participating in the contest; (5) testing the sensitivity of the estimation results with respect to the choices of the discount factor β, the number of potential entrants $||M_t||$, and the SIFT cutoff. Overall, the results from the robustness checks either support our modeling assumptions, or suggest that the qualitative nature of the empirical results are the same under alternative assumptions. The details of each of the robustness checks are provided in Online Appendix B.10.

3.6 Counterfactual Simulations

Most crowdsourcing platforms suggest that seekers provide feedback throughout the contest horizon. One of these platforms even uses the maxim "give feedback early - give feedback often". Following the platform's suggestion, more than 70% of contest seekers provide feedback throughout the contest horizon in our data.

To assess this common practice and quantify the impact of feedback on the contest outcome, in this section we utilize the estimation results of our structural model to conduct policy simulations. Specifically, we experiment with four alternative feedback disclosure policies: (1) the seeker provides feedback throughout the contest (*full feedback*), (2) the seeker does not provide feedback at all (*no feedback*), (3) the seeker provides feedback only in the first half of the contest (*early feedback*), and (4) the seeker provides feedback only in the second half of the contest (*late feedback*).

Our policy simulations aim at studying the impact of feedback in the steady state after the feedback policy change, holding everything else equal. Thus, we vary only the feedback disclosure policy to examine its impact on creators' equilibrium choices, while fixing the action-specific quality transition parameters (θ_1) and the number of potential entrants in each period ($||M_t||$). Doing so, our policy simulations correspond more closely to situations where (1) all seekers on the platform shift to an alternative policy and thus the policy change does not affect some contest holders (seekers) favorably and others adversely in attracting creators, and (2) seekers commit to the alternative feedback policy, and creators are informed of when they will receive feedback and make decisions based on their *beliefs* about the rating distribution in periods in which feedback is not available.

Each of the four policies above requires its own model; the policies differ in the feedback scheme the seeker commits to and follows, and accordingly the creators' information set is different under each policy's model. Note that the information creators have (about their own and their competitors' submission quality) is the only way that the feedback policies differ; other model primitives (parameters in the utility functions and *action-specific* state transition probabilities) stay the same across all alternative feedback policies.[21] It is also important to note that, the overall quality transition probabilities (with action integrated out, i.e., $p(x_{i,t+1}|x_{i,t}, s_t) = \sum_{a_{i,t}} Pr_t(a_{i,t}|x_{i,t}, s_t) \cdot p(x_{i,t+1}|x_{i,t}, a_{i,t})$) will be different across the feedback scenarios — (1) we model the quality-transition probabilities as being dependent on creators' *actions*; (2) our counterfactual simulations capture the change in quality-transition probabilities via the equilibrium *action* choices (i.e., $Pr_t(a_{i,t}|x_{i,t}, s_t)$), *which depend on the feedback policy* (see Online Appendix B.12.2). Here is how we incorporate the information differences. For full feedback, we utilize the model already defined in Section 3.4, where state variables are based on directly leveraging the observed ratings (note that ratings are truncated at 5-star, so as discussed in Section 3.4.4, a creator's "true" quality $x_{i,t}^{true}$ is recovered also based on the number of actions she has taken after receiving her first 5-star rating). In contrast, under the alternative feedback policies we experiment with (i.e., late feedback, early feedback, and no feedback), creators can no longer observe the ratings throughout the contest horizon. Hence, we model creator behavior in these alternative feedback scenarios with different information structures, as follows. We assume that under the alternative feedback policies, creators track not only their own and their competitors' realized ratings (if they can see any), but also their own and their competitors' actions in periods where creators do not receive performance feedback, as these are the (only) information available to creators in contests with publicly viewable submissions. Based on this information, creators form rational beliefs about what the distribution of their own best rating and their opponents' best ratings might be in those periods. For example, in the no feedback scenario, if a potential entrant sees a lot of existing creators have made multiple submissions in the contest, she might form the belief that her opponents' best ratings are relatively high and that her chance of winning is low, and thus decide not to join the contest.

Operationally, the state variables are defined differently in different feedback scenarios based on the information structure in each particular case – early feedback: $\mathbf{x}_{i,t}^{\mathrm{EF}}$ and $\mathbf{s}_t^{\mathrm{EF}}$; late feedback: $\mathbf{x}_{i,t}^{\mathrm{LF}}$ and $\mathbf{s}_t^{\mathrm{LF}}$; no feedback: $\mathbf{x}_{i,t}^{\mathrm{NF}}$ and $\mathbf{s}_t^{\mathrm{NF}}$; and in the full

[21] We assume the *action-specific* quality transition probabilities are structural parameters – creator i chooses her action $a_{i,t}$ based on her individual and contest-level state variables; and **given her action choice**, the *action-specific* quality-transition probabilities no longer depend on the contest-level state. This is not surprising, because the differences in *action-specific* quality transition probabilities already reflect the different levels of improvement each action could produce. Empirical evidence supporting this argument is provided in Online Appendix B.2.4 Table B.4.

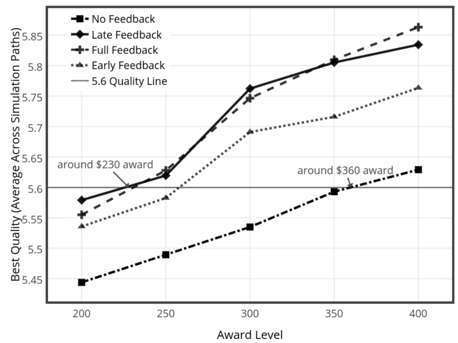
feedback case, state variables are directly based on quality state variables, i.e., $x_{i,t}^{true}$ and s_t^{true}. These $\mathbf{x}_{i,t}^{\cdot}$ and $\mathbf{s}_t^{\cdot}$ are mapped to the quality states $x_{i,t}^{true}$ and s_t^{true}, with the assumption that creators have rational expectations. With this modification, we can use the BRM algorithm described in Section 3.5.1.2 to solve for the equilibrium conditional choice probabilities (i.e., $\mathrm{P}_{i,t}^{*}(a_{i,t}|\mathbf{x}_{i,t}^{\cdot}, \mathbf{s}_t^{\cdot})$, where $a_{i,t} \in \{$*revision*, *redesign*, *do-both*, *do-nothing*$\}$), for each combination of $(\mathbf{x}_{i,t}^{\cdot}, \mathbf{s}_t^{\cdot})$ in each feedback scenario. We detail how we model creator behavior under each policy in Online Appendix B.12.1. In essence, for the four feedback disclosure policies, our counterfactual simulations involve solving four different versions of dynamic game.

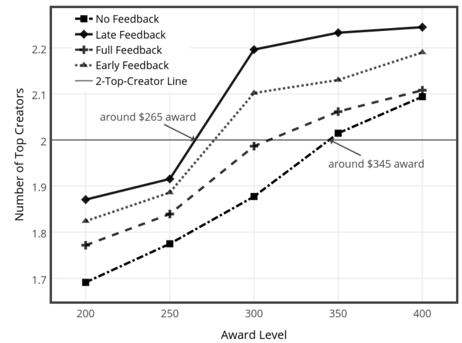
Based on the simulation results, we compare the feedback policy performance using three metrics: (i) the quality of the best design, (ii) the number of top performers (creators with at least one 5-star design), and (iii) the total number of creators participating in the contest. To capture the possibility that the relative performance of the four alternative policies may vary across different award levels, we conduct the simulations at a series of award levels ranging from \$200 to \$400, with steps of \$50. The award levels of most contests on the platform fall in this range. Given an award level and a feedback policy, we simulate 50,000 independent contest outcomes, and report the average value of the performance metrics across the 50,000 simulation paths.

3.6.1 Performance Metric I – The Maximum Quality

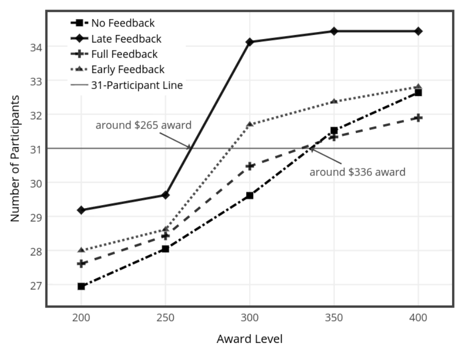
We first consider the best quality achieved by all contest participants. As discussed earlier, in contests with a single winner, the seeker is likely to care most about the quality of the best design (the extreme value). In Figure 3.4a, we can see that the no feedback policy is dominated by all the other three policies, confirming that giving feedback generally improves the contest outcome in terms of the maximum quality. This is not surprising, because feedback reveals how much the seeker likes a creator's existing submissions, and this information guides individual creators to choose the most productive follow-up action to hone, or revamp, their future design submissions towards the seeker's favorable direction (recall that actions exhibit different effectiveness conditional on different base-score; see Section 5.2).

To further visualize the value of providing feedback, we can set a target best quality, and compare how much monetary incentive (award) the seeker should provide to achieve the pre-specified target. For example, if a seeker wants to achieve a maximum quality of 5.6 (the horizontal line in Figure 3.4a), he can either (1) set the award at approximately \$360 and provide no feedback, or (2) set the award at around \$230 and provide feedback





(a) Maximum Quality **(b) Number of Top Performers**



(c) Number of Participants

Figure 3.4: Different Performance Measures for Contest Outcomes under Each Policy

throughout the contest, or only in the second half of the contest. In this example, giving feedback throughout or only in the second half of the contest can save the seeker $130, which is roughly one third of the award that the seeker has to pay if he decides not to provide performance feedback at all.

However, should performance feedback be disclosed "as early as possible and as frequently as possible" as suggested by crowdsourcing platforms? Not necessarily. Although the full feedback policy performs best most of the time, our simulation results also show that the late feedback policy performs as well as the full feedback policy at nearly all award levels experimented with, and its performance even exceeds the full feedback policy when the award is around $200 or $300.

3.6.2 Performance Metric II – The Number of Top Performers

In many cases, innovation seekers would also like to have a large number of top creators to provide a richer set of high-quality submissions. Therefore, we consider the number of top creators as the second performance metric; see Figure 3.4b. It turns out that the late feedback policy performs best at all award levels experimented with. This may sound counter-intuitive at first, but when we look closer at the mechanism through which feedback affects creators' activities, the result makes sense. Under the full feedback policy, feedback is provided from early on when there are only a handful of participants. Most incumbents (except the very few in the lead) are discouraged from taking follow-up actions, and potential entrants are also discouraged from entering the contest. In contrast, under the late feedback policy, the feedback is muted at the early stage of the contest – no creator is revealed to be in an advantageous position. More existing participants will remain active in making new submissions, and more potential entrants will be willing to join the contest. By the time when the performance feedback is disclosed, there will be more creators having good-quality submissions, and they will continue making submissions based on the guidance from the seeker's feedback (which helps creators choose the "most effective" follow-up strategy) in the last period. Hence, under the late feedback policy, there will be more top performers at the end of the contest.

As we did for the first performance metric, we can visualize the value of providing feedback on the second performance metric (the number of top performers), e.g., see the horizontal line drawn halfway up the graph in Figure 3.4b.

3.6.3 Performance Metric III – The Number of Participants

The results of our counterfactual simulations also suggest that the late feedback policy outperforms the other three policies in attracting creators to the contest at all award levels experimented with (Figure 3.4c). The reason for this is similar to the reason we provided in the previous subsection regarding the number of top performers – the disclosure of high ratings can discourage entry. However, one may ask, if the disclosure of the performance feedback always discourages the majority of the contest participants, why don't we always mute the performance feedback (choose the no feedback policy)? The issue with the no feedback policy is that, without the ability to distinguish between high-performers and low-performers, the expected probability for each person to win the contest decreases with the number of participants. Consequently, as the contest becomes more and more crowded, potential entrants are less willing to join the contest, and existing participants are less willing to take costly follow-up actions as well. As we did when analyzing the previous two performance metrics, we can also quantify the value of releasing feedback with the objective of attracting more participants, e.g., see the horizontal line drawn halfway up the graph in Figure 3.4c.

In summary, Sections 3.6.1-3.6.3 show the efficacy of late feedback as it strikes a balance between encouraging entries/incumbents' follow-up actions and provision of direction for creators.

3.7 Discussion and Conclusion

This paper studies how performance feedback policies affect the outcome of dynamic crowdsourcing contests. In such contests, new solvers can enter the contest at any time before the contest ends; incumbent solvers can submit additional submissions that exploit their existing ideas or explore new ideas, and these two strategies require different levels of effort and produce different distributions of quality improvement; and there exist uncertainties in the relationship between effort level and performance. The role of interim performance feedback on the outcome of dynamic crowdsourcing contests is an important, albeit somewhat understudied element of crowdsourcing contest design. We address this research question with a structural empirical analysis.

We develop a dynamic structural model to capture the economic processes that drive creators' participation behavior and how existing contest participants and potential new entrants react to the disclosed performance feedback. Our structural model explicitly considers potential entrants' endogenous entry processes, and distinguishes between incumbent creators' exploratory and exploitative follow-up actions. We recover the parameters in the structural model using a rich real-word data set on custom logo design contests collected from a major crowdsourcing platform.

Using the estimated structural model, we run counterfactual policy simulations that provide important insights into the role of feedback disclosure policy in crowdsourcing contests – they not only compare contest outcomes under alternative feedback disclosure policies, but also quantify the value of providing performance feedback on different performance metrics. In particular, we show that if all that the seeker cares about is the maximum quality achieved, both the full feedback policy and the late feedback policy perform quite well. If the seeker's objective is to maximize the number of high performers, or the total number of participants in the contest, the late feedback policy is the best option. Feedback helps guide creators' exploration and exploitation decisions, but can have a discouraging effect on entries and incumbents' follow-up actions. The late feedback policy attains the former benefit while mitigating the latter problem, by only giving feedback after many creators have had a chance to enter. If one considers the cost associated with monitoring submissions and providing performance feedback in real time (which we did not include in our formal analysis), the late feedback policy would become even more attractive for seekers. Given the above, our study may help seekers make better decisions about feedback policies in practice, by highlighting the merits of the late feedback policy. In fact, this late feedback policy has been used in some existing crowdsourcing contests. For example, programming contests organized by Mathworks start with a so-called “darkness” period, where participants do not receive any information about the performance of their own or other players’ entries. Participants are given access to the information about the status of the competition in a later stage of the contests (Balogh et al., 2010). On Topcoder (a popular crowdsourcing platform for programming and data science projects), coders are encouraged to first submit ‘test examples’ to assess whether the code is complete and generates reasonable results, and then submit ‘full submissions’ to access performance evaluations. Coders can only view their scores and rankings on the leaderboard, after making the first verified full submission, which resembles our late feedback policy.

Many contests settings share key features of our model. Consider algorithm design as an example. First, many algorithm contests allow participants to join the contest at any time and make multiple submissions throughout the contest period. Second, in algorithm contests, the notions of exploration (*redesign*) and exploration (*revision*) are still applicable. A *redesign* in algorithm contests would correspond to the action of abandoning the original algorithm and switching to a completely different approach, whereas a *revision* would correspond to the action of keeping the original approach and improving specific steps. Third, there is performance uncertainty: in some algorithm contests (e.g., Netflix and Kaggle), participants must submit the predictions generated by their algorithm onto the contest hosting platform to see the relative performance of their algorithm on a set of holdout labels in the form of its ranking on the public leader board. Since the holdout labels are not visible to algorithm contest participants, they do not know how well their algorithm performs until they submit their algorithm’s predictions online. Moreover, the winner is determined by the prediction accuracy of participants’ algorithms on a *final test set*, but not on the *holdout labels*, and therefore, the leader board ranking calculated based on the performance of submitted algorithms on *holdout labels* during the contest is not always the same as the final ranking that determines the winner.

In addition to algorithm design contests, other types of contests sharing the key features of our setting include, for example, web-design contests, print-design (e.g., designing communication brochure covers) contests, etc. Our structural model can be readily applied to such contests. In the logo design setting, our main result is that the late feedback policy performs well. The reason is that it strikes a balance between the discouraging and the guidance (explore vs. exploit) effects of feedback; intuitively this is a general principle, hence we would expect late feedback to perform well in the other settings as well. One note of caution is that in the logo design setting we found that full feedback and late feedback performed similarly along the dimension of highest quality score; thus although highest quality with late feedback was a bit better in the logo design setting, for other settings it would be worthwhile to run an empirical study and to do so one could utilize our structural model.

As one of the first empirical studies of large-scale crowdsourcing contests, our paper makes a number of important contributions but also has limitations. First, our results cannot be generalized to static, single-submission crowdsourcing contests, contests where the entry and effort decisions are made in separate stages; contests where creators collaborate rather than compete with each other; contests where the award is not guaranteed, etc. Second, in our current analysis, we assume the contests are independent of each other, and have not considered the creators' choices among concurrent on-going contests. A systematic analysis of what factors affect creators' choices of which contest to join could be a productive direction of future research. Third, similar to many prior studies of innovation-contest (e.g., Terwiesch and Xu (2008), Ales et al. (2017a), Korpeoglu et al. (2017)), we assume ex-ante homogeneous creators and creators' heterogeneity is captured by their realized interim submission ratings. We provide evidence that not modeling ex-ante creator heterogeneity (in creator experience and interest) does not significantly affect the estimation results in our research setting. However, this may not be true in other settings. It may be a fruitful future direction to look at how ex-ante (observed or unobserved) heterogeneity affects creators' activities in other types of crowdsourcing contests. Lastly, we focus on the role of quantitative performance feedback (i.e., ratings), and have not considered qualitative feedback, given through private messages, for two reasons: (1) qualitative feedback in the form of private messages is inaccessible to us; and (2) based on an interview with a marketing manager of a major online crowdsourcing platform, qualitative feedback occurs much less frequently than quantitative performance feedback. However, in other settings where the qualitative feedback is more prevalent and available to researchers, the effects of qualitative feedback could be a fertile direction for future research.

Despite these limitations, our paper is the first to provide a comprehensive dynamic structural framework to analyze creators' behavior in crowdsourcing contests. With the use of the structural model, we are able to disentangle intertwining effects of feedback on the outcome of crowdsourcing contests, helping both practitioners and researchers obtain a more comprehensive understanding of this increasingly popular new approach for sourcing innovation. Although the focus of our paper is the role of feedback, the structural framework we propose can be used to analyze other design issues in crowdsourcing contests. We hope that our work can pave the way for future research in this area.

CHAPTER 4

SKU Proliferation: High-Dimensional Choice Model and Online Retailing

Digital transformation is taking over the retail space. Yet, just as online shopping is becoming more and more convenient for consumers; it is also becoming more and more complex for retailers to manage. Great effort is spent on pricing, inventory and assortment decisions, as online retailers often offer an astonishing variety of choices. Moreover, given the existence of various advertisement and distribution channels, the online shopping environment is often complex, making consumer decision journey increasingly challenging to analyze.

To improve pricing and assortment decisions for these online retailers, an important first step is to achieve a realistic understanding of the substitution patterns (cross-effects) among a large number of products offered in the complex online environment. Although classic choice models offer an elegant framework for estimating substitution patterns among competing options, they have limited applicability and performance in settings with a large number of products. (We offer a review of classic choice models and their limitations in Section 2, and detailed performance comparisons of our proposed approach against alternative models in Section 4-6.)

We seek to provide a novel solution to address this problem, i.e., estimating flexible substitution patterns in presence of large choice sets. Before formulating the solution, we realize that there are three challenges in estimating flexible and realistic substitution patterns: First, the space of all the important product features can be extremely large (for example, Feldman et al. (2018) utilizes millions of features in recommendation system at Alibaba); Second, many of these product features are difficult to quantify (e.g., shape of the product, and color shades and patterns) or to observe (e.g., product information/advertisement from other online retailers or search engine could affect consumer purchasing decisions, but might not be available to the focal online retailer); Third, being exposed to various advertisement and distribution channels, customers in the complex online environment might exhibit extremely complicated purchasing behaviors, which is hard to capture based on researchers' prior knowledge. Due to the first two challenges, we often fail to use the observed product features to capture all the important factors determining consumer choices and product substitution patterns, which leaves unobserved/uncaptured features in error terms. Yet, the last challenge makes it hard for researchers to impose structures on these error terms by making assumptions a priori.

To tackle those challenges, we propose a high-dimensional choice model that allows for flexible substitution patterns and carries few assumptions a priori. We first leverage customers' clicking activities; based on the likelihood of products being clicked together, we use econometric and machine learning methods (i.e., Graphical Lasso) to estimate a product network which captures the conditional correlation among the underlying utility shocks of products. Next, we take the estimated product network as an informative initial guess for the structured precision matrix, in estimating a multinomial Probit model, where we allow for a flexible precision matrix among products' error terms in the utility specification. We support this method theoretically and show its performance in simulations and in an empirical setting. This method allows us to uncover product substitution patterns with improved accuracy, while maintaining computational feasibility. Lastly, based on estimated substitution patterns, we provide demand forecasts and pricing recommendations to online retailers.

We test the performance of our method on synthetic data under various scenarios classified based on precision matrix density, signal-to-noise ratio, high-low clicking levels, etc. Compared to classical multinomial Probit models with iid error terms (IID Probit models) and random coefficient models, our method consistently offers better in- and out-of-sample performances, and provides more accurate demand forecasts— out-of-sample MAPE of our model (18.48%) is significantly lower than that of the random coefficient model (45.31%) and the IID Probit model (46.44%), representing 59.21% and 60.21% reductions in demand forecasting errors, respectively. By doing so, we are able to recommend better pricing decisions— in comparison with the IID Probit model (the random coefficient Probit model), we provide an 13.47% (29.05%) increase in total profit when maximizing profits, and an average 17.40% (8.72%) increase in total revenue when maximizing revenues. We notice that our model does particularly well in improving profits among slow-moving products. Finally, we apply our method to an online retailing dataset. We show that, in this empirical setting, our method continues offering better in- and out-of-sample performances, and providing more accurate demand forecasts. Out-of-sample demand forecasts are improved by 41.44% on average in comparison with the random coefficient Probit model.

Our study makes several contributions. First, we offer an approach to estimate flexible substitution patterns for large choice sets. Realistic and flexible substitution patterns are inputs for making many retailing decisions. We demonstrate that our method offers a more accurate and realistic substitution pattern, helps retailers make better demand forecasting and pricing decisions, and could potentially be further generalized to many other retailing applications, including promotion and assortment decisions. Second, we combine the state-of-art Graphical Lasso method from machine learning to the choice modeling framework. By doing so, we contribute to a growing literature attempting to combine machine learning methods with economic methods on consumer choice. These growing attempts advance the traditional econometric models in answering new data challenges with the recent digital transformations. Finally, this paper also contributes to an emerging literature that leverages consumer clickstream data in choice models to help identify substitution patterns. With more data available, we can make fewer assumptions a priori, but let the data inform the modeling decisions.

4.1 Literature Review

Our paper proposes a novel method combining choice modeling and machine learning methods, and demonstrates the method's potential applications in dealing with large-scale problems in revenue management. We discuss how we contribute to four streams of literature: (1) *classic choice models*, (2) *large-scale choice models*, (3) *the application of machine learning methods in choice models*, and (4) more broadly, *empirical revenue management*.

Classic Choice models. Choice models with Logit form (i.e., multinomial logit model, or MNL) are widely adopted in demand estimation and revenue management, since the corresponding conditional choice probabilities enjoy the convenience of closed-form specifications. However, this appeal of the MNL's closed-form specification comes at a cost for demand analysis—the MNL model exhibits the well-known independence of irrelevant alternatives property (IIA) property, which can impose unrealistic substitution patterns in demand analysis. (For example, the IIA property implies that a focal product with a lower price gains share from all other products in proportion to their original shares, regardless of how similar they are to the focal product. More examples summarized in Train (2009).) This is problematic especially when it comes to revenue management, because obtaining optimal pricing strategy relies on realistic substitution patterns.

Two widely used solutions to achieve a non-IIA specification are (1) varying the error components in the latent utility specification (e.g., the nested logit or the generalized extreme value distribution model), or (2) shifting to a random coefficient specification (e.g., mixed logit/Probit model). However, both are with limitations. The nested logit model requires pre-knowledge about the product nest structure. The random coefficient model, though theoretically can approximate any random-utility model to any desired degree of accuracy (McFadden and Train (1997), often fail to achieve ideal performance in reality, due to the imperfect choice of explanatory variables and the distributions for the random parameters. (See Section 4-6 for demonstrations comparing our model with these alternative models.)

To achieve a fully flexible and realistic substitution pattern, an ultimate solution is to use multinomial Probit model (MNP) with the error term follows a normal distribution with a flexible covariance matrix (Train 2009 p.103). However, this classic model is rarely applied, because the number of parameters in the model is quadratic in the number of products, making it infeasible to be estimated in most practical settings. Moreover, unless we have tremendous amount of customers' purchasing observations, we cannot achieve efficient estimates of these extremely flexible models. We offer a solution to this computational challenge by leveraging customer clickstream data with Graphical Lasso, and incorporating the learnt information as structured covariance matrix in estimating the Probit model.

Large-scale choice model. While we estimate a choice model with large choice sets, we contribute to an growing literature of large-scale choice models. Studies here have several different emphasizes, including (A) estimating a *basket demand* as the number of potential product baskets increases drastically with more products in the choice set (Ruiz et al., 2017); (B) estimating models with an improved *computational efficiency* and potentially trading-off some accuracy in estimating substitution patterns among all choices (Fox, 2007; Chiong and Shum, 2016; Amano et al., 2019); and (C) recovering a *realistic and flexible substitution patterns* in presence of the large choice sets, while maintaining computational feasibility. The focus of our paper is (C), which we discuss in more details.

The closest to our method are papers estimating a structured covariance matrix. As mentioned above, estimating a full covariance matrix in Probit models allows for any substitution patterns, but generates huge computational burden. This computational burden can be alleviated with an imposed structure on the covariance matrix (instead of estimating a full covariance matrix for the errors). Yai et al. (1997) estimates a Probit model of route choice where thee covariance between any two routes depends on the length of shared routes segments. Dotson et al. (2018) estimates a Probit model of product choice where covariance between any two products depends on the perceptual distance between choice alternatives; and the distance is parameterized by the observed product attributes (i.e., the covariates). Our paper shares the same spirit—enjoying the flexibility from Probit model, and meanwhile cutting down the parameters in inverse covariance matrix (or precision matrix). But instead of making assumption a priori (i.e., parameterizing the covariance with the observed product attributes), we let the data tell us what structure to impose on the inverse covariance matrix by looking at what products consumers click at the same time.

There are other papers approaching the problem (i.e., allowing for a flexible substitution patterns in presence of the large choice sets) using aggregate demand models. Smith and Allenby (2019) and Smith et al. (2019) propose random partition in estimating a demand model and use a Bayesian approach to flexibly estimate these partitions using supermarket scanner data. Yet, the aggregate demand model does not capture more micro-level consumer choice, making it hard to run counterfactual analyses based on consumer behavior and to provide optimal solutions to retailers' pricing decisions. Moreover, the work considers single-layer, non-overlapping partitions, which could be important to relax.

Machine learning methods in choice models and graphical lasso. Our paper also contributes to an emerging literature on combining machine learning methods with choice modeling methods. A couple of above-mentioned papers in large-scale choice models are also in this area. Ruiz et al. (2017) apply machine learning techniques (in particular, stochastic gradient descent and variational inference) to make the choice model scalable for a large choice setting. Chiong and Shum (2016) use random projection to compress a high-dimensional choice set into a lower-dimensional Euclidean space when estimating the choice model. Wan et al. (2017) use a latent factorization approach that incorporates price variation. Our paper introduces the idea of using Graphical Lasso, a tool popular in machine learning field, in estimating the structure of the covariance matrix of errors in Probit models. Below, we briefly summarize this approach and its use in other areas.

To uncover the important connections among a large number of products, we utilize the consumer clickstream data and let the data inform us about the product network using a Gaussian graph model. The edges in these Gaussian graphical models can be interpreted as the direct influence between two nodes. Graphical Lasso method eliminates spurious or misleading relationships by removing non-existing direct links among a large number of nodes. Proposed by Friedman et al. (2008), Graphical Lasso is developed to estimate the precision matrix and obtain the graphical structure through a penalized maximum likelihood approach. It is soon applied to different fields, including neuro-science (Allen et al., 2014), information network (Gomez-Rodriguez et al., 2012), biostatistics (Cai et al., 2013), and transportation (Yan and Kung, 2018). Similar to Rothman et al. (2010), Yin and Li (2011), and Cai et al. (2013), we are aiming at estimating a sparse precision matrix after taking into account for the effects of the covariates on the mean utilities. It has not yet been applied to operations management problems, nor to choice models.

Empirical Revenue Management. Finally, our research contributes to empirical revenue management literature by providing realistic estimates for the substitution patterns among products, which is an important foundation for revenue management decisions in retailing context. Potential applications include dynamic pricing decisions (Ferreira et al., 2016; Papanastasiou and Savva, 2017; Fisher et al., 2018) and inventory decisions (Caro and Gallien, 2010; Gallego et al., 2020).

Closest to our work, there is a stream of literature in empirical revenue management, that uncovers products substitution patterns. To describe the competitive market structure and product substitution patterns, researchers have been utilizing the co-occurrence of products mentioned in online discussion forums (Netzer et al., 2012), customer reviews and the products they mention (Lee and Bradlow, 2011), and, similar to our approach, online search data (Ringel and Skiera, 2016; Kim et al., 2010). With the main focus being visualizing the connections among products, these work (most with descriptive methods) does not provide an explicit elasticity matrix nor a straight-forward solution for pricing decisions.

4.2 A Model with Flexible Substitution Patterns

In this section, we first present the choice model and discuss the challenges associated with estimating the model. We then provide two solutions to address the challenges of high dimensionality and limited variations in purchase data. Specifically, we estimate a sparse precision matrix defined as the inverse of covariance matrix, while leveraging denser substitution information provided through clickstream data.

4.2.1 The Model

Consider a product category with J products. Let u_{ij} denote the utility that consumer i obtains when purchasing product j ($j = 1, 2, ..., J$). Consumer i can also not purchasing any of the offered products, i.e., choose the outside option with a utility u_{i0}. Let y_{ij} be a binary variable, where $y_{ij} = 1$ denotes consumer i purchases project j, and $y_{ij} = 0$ otherwise. A customer will choose the option j that maximizes her utility, that is, $y_{ij} = \mathbf{I}\left(u_{ij} = \max_{k,k=0,1,2,...,J}\{u_{ik}\}\right)$.

In particular, here is our product utility specification,

$$\begin{aligned} u_{ij} &= v_{ij} + \epsilon_{ij} = \alpha + X_{ij}\boldsymbol{\beta} + \epsilon_{ij}, j = 1, 2, ..., J, \\ u_{i0} &= \epsilon_{i0}. \end{aligned} \tag{4.1}$$

The vector X_{ij} captures product features, which includes time-variant, person-specific characteristics (for example, product reviews, recommendations, prices and promotions), as well as characteristics that are constant over time and across person (for example, product specifications). The coefficient $\boldsymbol{\beta}$ captures sensitivities to observed product characteristics. Since only the differences across these utilities are identifiable, we normalize v_{i0} in the outside option to be zero. Error term ϵ_{ij} represents customer i's utility shock of purchasing product j. Since only differences in utilities are identifiable, we normalize the utility shocks of all products with respect to the utility shock of the outside option. Specifically, we define a vector $\tilde{\boldsymbol{\epsilon}}_i := \{\tilde{\epsilon}_{i1},\ \tilde{\epsilon}_{i2},\ ...,\ \tilde{\epsilon}_{iJ}\}$, where $\tilde{\epsilon}_{ij} = \epsilon_{ij} - \epsilon_{i0}, j = 1, 2, ..., J$. Vector $\tilde{\boldsymbol{\epsilon}}_i$ is a normally distributed $J \times 1$ vector, i.i.d. across individual customers, with zero mean and variance–covariance matrix

$$\Sigma = \begin{pmatrix} 1 & \sigma_{1,2} & \cdots & \sigma_{1,J} \\ & \sigma_{2,2} & \cdots & \sigma_{1,J} \\ & & \ddots & \vdots \\ & & & \sigma_{J,J} \end{pmatrix}. \tag{4.2}$$

We only specify the upper part of the matrix because it is symmetric. The elements in the lower part equal the corresponding elements in the upper part. Note σ_{11} is standardized to 1 because the scale of the utility is unidentifiable and irrelevant.

The model is very similar to the standard Probit model, except for that the latter assumes the variance–covariance matrix to be the identity matrix (i.e., error terms are i.i.d. across options). The advantage of allowing a fully flexible variance–covariance matrix is that it imposes minimal assumptions on substitution patterns (Train, 2009). Recover a flexible substitution pattern by estimating the variance-covariance matrix of error terms is greatly beneficial in online retail settings, since there might be a lot of important product features not included in the observed product feature X for the following two reasons. First, the space of all the important product features of a product can be extremely large. Second, many the product features are difficult to quantify (e.g., shape of the product, and color shades and patterns) or to observe (e.g., product information/advertisement from other online retailers or search engine could affect consumers purchasing decision, but might not be available to the focal online retailer). Later, we are going to see this benefit from estimating the variance-covariance matrix based on our simulation and empirical analyses.

However, the matrix Σ contains $[(J-1)J/2-1]$ free parameters to be estimated, which increases quadratically with the number of products. For example, if the product category includes 100 products, then there will be $(100-1)*100/2-1 = 4,949$ parameters to be estimated. In other words, the estimation problem becomes too large to solve as the number of products rises. On the other hand, in contrast with the scale of the problem, consumer purchase data are usually sparse, because a consumer only selects one of the many options. As a result, most of the outcome variables take zero as values. The limited variation among the outcome variables, or the sparsity of the data, further adds to the difficulty of estimating a flexible covariance matrix.

4.2.2 Overview of Our Solutions

We offer two solutions to tackle these challenges. To address the dimensionality challenge, we estimate a sparse precision matrix, which is defined as the inverse of the covariance matrix. To overcome the sparsity of purchase data, we leverage clickstream data which provide denser information on what consumers consider as substitutable products. We first summarize the insights and then discuss the details of the approach in Section 4.2.3.

To tackle the dimensionality challenge, we leverage one critical property of Gaussian variables and its implication in graph theory: the *precision matrix* of Gaussian variables indicates conditional dependence. In particular, an element of the precision matrix is non-zero if and only if the two corresponding variables are correlated conditional on all other variables. Theoretical details of this property are formally introduced in Section 4.2.3.1. In our setting, this property intuitively suggests a non-zero element in the precision matrix of the unobserved errors indicates: the two products' utility shocks are correlated *conditional on* utility shocks of all the other products, in other words, the two products experience common utility shocks that are unique to them but not to other products—for instance, two sweaters both have unique lace trims while others do not; two cosmetic products are both promoted by a Youtube influencer but not other products. While there could be tens of thousands or even millions of product pairs, we note that most product pairs likely do not share (sufficiently) unique utility shocks. Consequently, the precision matrix is likely sparse. Moreover, customers typically consider a handful of options in their decision process even though hundreds options are available, which also likely leads to a sparse precision matrix. As we will show in Section 4.4.1, we find empirical evidence supporting our conjecture of sparse precision matrices in various retailing contexts.

Next, to overcome the sparsity of purchase data, we leverage clickstream data which provide denser information on what consumers consider as substitutable products. We leverage consumer clickstream information to estimate the structure of the precision matrix, that is, which elements are non zero. Clickstream data are helpful for two reasons. First, clickstream data are typically denser than purchasing data. Many consumers with clicking activities do not end up purchasing. According to recent surveys,[1] the average conversion rate in online shopping is often less than 4% across many product categories and shopping channels. Even among consumers who decide to purchase, they click more options than they purchase. Second, if we use purchasing data to estimate the non-zero entries in the precision matrix, the identification needs to reply on aggregate-level variations across time and markets rather than consumer level variations, because each consumer only chooses one option. In practice, we may not have sufficient variations across time or markets to estimate the precision matrix. On the other hand, if we use clickstream data, we can exploit individual-level variations to estimate which products are close substitutes, because we can directly observe products that consumers consider at the same time. We prove that this intuition is true and explain how we achieve the estimation in Section 4.2.3.2.

4.2.3 Estimation

We first introduce the theoretical properties of the precision matrix. We then discuss the two stages of our estimation strategy. In the first stage, we use clickstream data to estimate which entries of the precision matrix are non zero. In the second stage, we solve the original estimation problem knowing which elements of the precision matrix are non zero.

[1]Adobe Digital Index 2020 report: https://www.slideshare.net/adobe/adi-consumer-electronics-report-2020.
Research from Episerver retail clients: https://www.episerver.com/reports/2019-b2c-ecommerce-benchmark-report/.
E-commerce 2019 KPI report: https://www.wolfgangdigital.com/kpi-2019/.

4.2.3.1 Theoretical properties of the Precision Matrix

The precision matrix Φ ($\Phi = \Sigma^{-1}$, where Σ^{-1} is from Equation (4.2)) have the following properties.

PROPERTY 1: *If ϵ follows multivariate Gaussian distribution, the conditional independence between ϵ_j and ϵ_k given other variables ($\epsilon_{-j,-k}$) is equivalent to $\Phi_{jk} = 0$. Mathematically, $\Phi_{jk} = 0 \iff \epsilon_j \perp \epsilon_k | \epsilon_{-j,-k}$. This property is based on Hammersley-Clifford Theorem (Hammersley and Clifford, 1971).*

PROPERTY 2: *Let $G = (V, E)$ be a graph representing conditional independence relations among ϵ, where V is the vertex set and E is the edge set. We denote $(j, k) \in E$, if there is an edge between ϵ_j and ϵ_k. The edge between ϵ_j and ϵ_k is excluded from E if and only if ϵ_j and ϵ_k are independent given $\epsilon_{-j,-k}$. Mathematically, $\epsilon_j \perp \epsilon_k | \epsilon_{-j,-k} \iff (j, k) \notin E$ (Lauritzen, 1996).*

We define the support of a matrix M as the (0,1)-matrix with ij^{th} entry equal to 1 if the ij^{th} entry of M is non-zero, and equal to 0, otherwise. Properties 1 and 2 suggest that the support of the precision matrix for Gaussian variables $\boldsymbol{\epsilon}$ represents the conditional dependence among products' unobservables. It also has implications in graph theory—the support of the precision matrix can be represented by a graph whose edges denote conditional dependence between corresponding products.

4.2.3.2 Stage I: Estimate the Location of Non-Zero Entries In The Precision Matrix

We leverage clickstream data to estimate the non-zero entries in the precision matrix. At the time when consumers are deciding which products to click, the utility of product j perceived by consumer i is denoted by u^b. Specifically,

ASSUMPTION 1-PERCEIVED UTILITY: *The perceived utility, u^b_{ij}, of product j for consumer i is $u^b_{ij} = h(u_{ij} - u_{i0})$, where $h(\cdot)$ is a strictly monotonic (increasing) function.* (We further relax this assumption and allow for the possibility that consumers might not be able to assess utilities precisely at the time when they decide which products to click in Section 4.3.4 and Appendix C.5.)

We also assume a customer has a higher probability to click products with a higher perceived utilities u^b. We denote customer i's clicking probability for product j by p^b_{ij}; in particular,

ASSUMPTION 2-MONOTONE INCREASING CLICKING PROBABILITY: *Customer i clicks product j with a clicking probability of p^b_{ij}, and p^b_{ij} is strictly monotonic increasing with u^b_{ij}, in particular, $p^b_{ij} = g(u^b_{ij})$ and $g(.)$ is a strictly monotonic increasing function.*

Assumption 1 assumes that a product's perceived utility is an increasing function of its actual utility and a decreasing function of the utility of the outside option. Assumption 2 assumes that the products with higher perceived utilities are more likely to be clicked than the less preferred ones. Under the two assumptions, a product with higher actual utility is more likely to be clicked, everything else being equal. Moreover, a customer with less preferred outside option will click more products, everything else being equal. Both assumptions are intuitive and likely satisfied. For simplicity of notations, we denote $\tilde{u}_{ij} = u_{ij} - u_{i0}$ and $p^b_{ij} = H(\tilde{u}_{ij}) = g(h(\tilde{u}_{ij}))$ in the rest of the paper. Based on Assumptions 1 and 2, $H(.)$ is strictly monotonic increasing.

Intuitively, clickstream data provide information about the structure of the precision matrix Φ in latent utilities. Suppose we observe consumers who are likely to click products j also tend to be more likely to click product k conditional on the probability of clicking for all other products, and such conditional co-clicking behavior is not well explained by the observed product features X_{ij} and X_{ik}, then it is likely that the unobserved utility shocks, $\tilde{\epsilon}_{ij}$ and $\tilde{\epsilon}_{ik}$ conditional on $\tilde{\epsilon}_{i,-j-k}$, are highly positively correlated. Correspondingly, the ik^{th} element in Φ is significantly positive. This aforementioned intuition is indeed proven to be true. Namely, clickstream data inform us the positions of non-zero entries in the precision matrix. Lemma 4.2.1 states this intuition formally.

Lemma 4.2.1 *Under "Assumption 1-Perceived Utility" and "Assumption 2-Monotone Increasing Clicking Probability", controlling for product features Xs, p^b_{ij} and p^b_{ik} are independent conditional on $p^b_{i,-j-k}$, if and only if $\tilde{\epsilon}_{ij}$ and $\tilde{\epsilon}_{ik}$ in Equation* (4.1) *are independent conditional on $\tilde{\epsilon}_{i,-j-k}$, i.e.,*

$$p^b_{ij} \perp p^b_{ik}|_{p^b_{i,-j-k},X} \Longleftrightarrow \tilde{\epsilon}_j \perp \tilde{\epsilon}_k|_{\tilde{\epsilon}_{i,-j-k},X}.$$

Proofs for the Lemma are provided in Appendix C.1.

Let us denote the precision matrix of p^b conditional on X as Φ^b with the jk^{th} entry being $p^b_{ij} \perp p^b_{ik}|_{p^b_{i,-j-k},X}$. Lemma 4.2.1 shows that Φ^{b} and Φ share the same support, where Φ^{b} denotes the conditional precision matrix of clicking probability among products in the clicking model and Φ denotes the precision matrix among the unobserved factors in the product latent utility in Equation (4.1). In particular, if and only if $\sigma(p^b_{ij}, p^b_{ik})|_{p^b_{i,-j-k},X}$ in Φ^b is non-zero (zero), $\sigma(\tilde{\epsilon}_{ij}, \tilde{\epsilon}_{ik})|_{\tilde{\epsilon}_{i,-j-k},X}$ in $\tilde{\Phi}$ is also non-zero (zero), suggesting the utility shocks of products j and k are significantly correlated (uncorrelated) conditional on the shocks of all other products. This is exactly how we are able to learn the positions of the non-zero entries in the precision matrix Φ by estimating the structure of Φ^{b} from the clickstream data.

As mentioned in Section 4.2.2, Φ is likely to be sparse; therefore, Φ^{b}, which shares the same support as Φ, is likely to be sparse too. We apply the Graphical Lasso algorithm developed by Friedman et al. (2008) to estimate the positions of the non-zero entries in a sparse precision matrix Φ^{b}. Specifically, $\hat{\Phi}^b$ is the solution to the following optimization problem over nonnegative definite matrices Φ^b:

$$\hat{\Phi}^b = \mathrm{argmin}_{\Phi^b}\big(\mathrm{tr}S_{pp}\Phi^b - \mathrm{logdet}\Phi^b + \lambda\|\Phi^b - \mathrm{diag}(\Phi^b)\|_1\big), \tag{4.3}$$

where S_{pp} is the empirical covariance matrix of vector $\boldsymbol{p^b}$ conditional on product features X.[2] The term $\mathrm{tr}S_{zz}\Phi^b - \mathrm{logdet}\Phi^b$ is the negative log-likelihood, where tr denotes the trace and det denotes the determinant. The term $\|\Phi^b - \mathrm{diag}(\Phi^b)\|_1$ is the sum of the absolute values of off-diagonal coefficients of Φ^b. See Appendix C.2 for proof.

The Graphical Lasso estimator uses an L_1 penalty to enforce sparsity on the precision matrix Φ^b through the tuning parameter λ: the larger the tuning parameter λ is, the more sparse the precision matrix. There are several methods to set the tuning parameter λ: cross-validation, BIC, or adaptive methods. We discuss these methods in detail in Appendix C.4. The optimization problem in Equation (4.3) can be solved using the blockwise coordinate descent approach in Banerjee et al. (2008). See Rothman et al. (2008) for the guaranteed convergence of this estimator. In particular, the convergence rate of this estimator depends on the sparsity of the true precision matrix.

Upon obtaining the estimated $\hat{\Phi}^b$, we also compute the corresponding covariance matrix associated with the sparse precision matrix ($\hat{\Sigma}^b = (\hat{\Phi}^b)^{-1}$). Now with this $\hat{\Phi}^b$ (i.e., for each element in the matrix, we know whether it is zero or not), we have learnt the structure of Φ in the latent utility, which shares the same structure as Φ^b (according to Lemma 1).

4.2.3.3 Stage II: Estimate The Probit Model with A Sparse Precision Matrix

Knowing which entries in the precision matrix take non-zero values, we now return to the original problem: estimating the choice model using purchase data. We fix the zero entries in Φ to be zero, and focus on estimating its non-zero elements and coefficients for product characteristics, i.e., α and $\boldsymbol{\beta}$. For simplicity of notation, we denote the vector of all the non-zero elements in Φ as $\boldsymbol{\gamma}$. Specifically, we estimate α , $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ by

[2] In particular, $S_{pp} = \frac{1}{N}\sum_{i=1}^{N}[(p_i^b - \hat{\beta}^b p_i^b)(p_i^b - \hat{\beta}^b p_i^b)^{\mathrm{T}}]$, in which $\hat{\beta}^b = S_{xp}S_{xx}^{-1}$ and $S_{xp} = \frac{1}{N}\sum_{i=1}^{N}(p_i^b - \bar{p^b})(x_i - \bar{x})^{\mathrm{T}}$ and $S_{xx} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(x_i - \bar{x})^{\mathrm{T}}$.

maximizing the following likelihood function

$$L(\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \prod_{i=1}^{N} \prod_{j=1}^{J} \Pr(y_{ij} = 1 | \alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})^{d_{ij}}, \tag{4.4}$$

where d_{ij} indicates whether consumer i purchases product j or not based on the real data, $\prod_{j=1}^{J} \Pr(y_{ij} = 1|\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})^{d_{ij}}$ is the probability of observing the actual choice d_{ij} made by consumer i, and $\Pr(y_{ij} = 1|\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})$ is the conditional choice probability (CCP) of observing consumer i purchase product j given a set of parameters $(\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma})$. Specifically, CCP is calculated as

$$\begin{aligned}
\Pr(y_{ij} = 1|\alpha, \boldsymbol{\beta}, \boldsymbol{\gamma}) &= \Pr(u_{ij} > u_{ik}, \forall k \neq j) \\
&= \Pr(v_{ij} + \epsilon_{ij} > v_{ik} + \epsilon_{ik}, \forall k \neq j) \\
&= \Pr(v_{ij} + \epsilon_{ij} - \epsilon_{i0} > v_{ik} + \epsilon_{ik} - \epsilon_{i0}, \forall k \neq j) \\
&= \Pr(v_{ij} + \tilde{\epsilon}_{ij} > v_{ik} + \tilde{\epsilon}_{ik}, \forall k > 0,\ k \neq j) \\
= \int_{\tilde{\epsilon}_{i1}=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}-v_{i1}} \cdots &\int_{\tilde{\epsilon}_{ij}=-v_{ij}}^{\infty} \cdots \int_{\tilde{\epsilon}_{iJ}=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}-v_{iJ}} F(\tilde{\epsilon}_i) \quad d\tilde{\epsilon}_{i1}\, d\tilde{\epsilon}_{i2} ... d\tilde{\epsilon}_{iJ},
\end{aligned} \tag{4.5}$$

where $F(\cdot)$ is the cumulative distribution function for $\tilde{\epsilon}_i$ $(= \{\tilde{\epsilon}_{i1}, \tilde{\epsilon}_{i2}, ..., \tilde{\epsilon}_{iJ}\})$.

We estimate the parameters in Equation (4.5) using simulated maximum likelihood (SML) estimator, because the conditional choice probability based on normally distributed error terms does not have a closed-form expression. In particular, the simulation of $F(\cdot)$ is performed using the Geweke-Hajivassiliou-Keane (GHK) simulator (Hajivassiliou, 1992).

Note we do not use clickstream data in this stage of estimation for three reasons. First, we would like to retain the generality of our method. To use clickstream data in this stage, we would need to make additional structural assumptions about (1) how consumers decide which options to click, (2) how a consumer's clicking decisions might inform us about his/her purchasing decisions, and (3) what product information is not available at the clicking stage and what is available at the purchasing stage. It is unclear whether these assumptions will be supported in each specific setting, as consumer decision process will vary by contexts and platform designs. Second, in the model applications, we run policy simulations by applying the model under alternative product features, such as, alternative pricing strategies. If we need the clickstream information to estimate the second stage, we would need to simulate all possible subsets of products clicked by a consumer, which is a combinatorial problem.

Lastly, using only purchase data and not making additional assumptions would allow us to make fair comparisons between our model and other choice models. It better allows us to assess the value of estimating a flexible precision matrix rather than ignoring unobserved substitutability or making assumptions about it.

4.2.3.4 Stage II Extension: Polynomial Approximation

Even with a relatively sparse precision matrix, sometimes we may still end up estimating a large number of parameters if the space is large or if the precision matrix is not sufficiently sparse. In this case, the estimation in Stage II may still take considerable time because it involves simulation. Therefore, to further alleviate the computational burden, we propose an approximation method.

Recall that in the first stage, we obtain an estimate of the precision matrix based on clickstream information ($\hat{\Phi}^b$) using the Graphical Lasso estimator. So far, we have only used it to obtain the positions of the non-zero entries in the precision matrix, but have not used the levels of the non-zero entries. A larger entry in $\hat{\Phi}^b$ would likely suggest a larger entry in the precision matrix for the latent utility $\hat{\Phi}$.[3] Therefore, we propose to approximate the precision matrix $\hat{\Phi}$ using a function of $\hat{\Phi}^b$. Specifically, instead of estimating all the non-zero elements in the precision matrix, we approximate it using a polynomial function. For a M-level polynomial approximation, we have:

$$\tilde{\phi}_{jk} = \sum_{m=1}^{M} a_m * (\phi^b_{jk})^{m-1}. \tag{4.6}$$

That is, we estimate M parameters in $\boldsymbol{a} = \{a_1, a_2, ..., a_M\}$.

Now, both the structure and the intensity of the precision matrix Φ are represented by a small number of parameters in $\boldsymbol{a}$. We thus revise the likelihood function in Equation (4.4), in particular,

$$L(\alpha, \boldsymbol{\beta}, \boldsymbol{a}) = \prod_{i=1}^{N} \prod_{j=1}^{J} \Pr(y_{ij} = 1 | \alpha, \boldsymbol{\beta}, \boldsymbol{a})^{d_{ij}}, \tag{4.7}$$

and

$$\Pr(y_{ij} = 1 | \alpha, \boldsymbol{\beta}, \boldsymbol{a}) = \int\limits_{\tilde{\epsilon}_{i1}=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}-v_{i1}} \cdots \int\limits_{\tilde{\epsilon}_{ij}=-v_{ij}}^{\infty} \cdots \int\limits_{\tilde{\epsilon}_{iJ}=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}-v_{iJ}} \tilde{F}_{\boldsymbol{a}}(\tilde{\epsilon}_i) \quad d\tilde{\epsilon}_{i1}\, d\tilde{\epsilon}_{i2} ... d\tilde{\epsilon}_{iJ}. \tag{4.8}$$

[3] This intuition is supported in our simulations (Section 4.3).

where $F_{\tilde{\boldsymbol{a}}}(\cdot)$ is the cumulative normal distribution of a vector of dimension J with mean zero and variance-covariance $\tilde{\Sigma}(\boldsymbol{a}) = \tilde{\Phi}^{-1}(\boldsymbol{a})$, and the jk^{th} entry in $\tilde{\Phi}^{-1}(\boldsymbol{a})$ is specified in Equation (4.6).

From now on, we refer to the original method without approximation as the *full model* and the approximation method as the *approximation model*. We evaluate the approximation model's performance using synthetic data and empirical data in Section 4 and Section 5 respectively.

4.3 Performance on Synthetic Data

We now demonstrate the performance of the proposed method and fine-tune the estimation procedure using simulated data. We first describe how we construct the synthetic data. We then discuss the following key points in the estimation process: (1) selection of the tuning parameters, (2) stage I performance in recovering the positions of non-zero entries in the precision matrix, (3) stage II performance in estimating model coefficients, and (4) product-level demand estimation accuracy of our method in comparison with existing methods. These results will guide our estimation strategy on the real data.

4.3.1 Synthetic Data Generation

Before discussing tuning and model performance, we first introduce how our synthetic data are generated. We simulate consumer clicking and purchasing data based on assumptions introduced in the previous section. That is, we intuitively assume that consumers are more likely to click products that are of higher utilities in comparison to the utility of the outside option (Assumptions 1 and 2), and purchase products with the highest utility.

For the baseline case, we consider a choice setting with 20 periods, 500 customers per period, 30 products, and 3 observed product features: $x_{1,j,t}$, $x_{2,j,t}$, and $x_{3,j,t}$, where $x_{3,j,t}$ denotes price. The scale of the underlying parameters and the distribution of product features are set based on a real-world empirical setting, which we introduce in Section 5. In particular, we set: $\alpha = 0$, $\beta_1 = 0.5631$, $\beta_2 = 1.2568$, $\beta_3 = -0.0144$. The precision matrix Φ is assumed to has 8% non-zero elements with the non-zero off-diagnol entry positions randomly determined. The product features are simulated based on their distributions in the real data. The resulting signal-to-noise ratio (SNR) is 5.339. When generating the clicking incidences, we assume that customer i clicks product j with a probability of p^b_{ij}, which is increasing with $\tilde{u}_{ij}$. In particular, we assume that $H(.)$ in Section 4.2.3.2 is a Sigmoid function, i.e., $p^b_{ij} = \frac{exp(\tilde{u}_{ij}+c^{\mathrm{b}})}{1+exp(\tilde{u}_{ij}+c^{\mathrm{b}})}$, where c^{b} adjusts the clicking level and is set to be zero in the baseline case. Later in alternative simulation settings, we conduct sensitivity analyses regarding this functional form assumption, the clicking level assumption, and other uncertainties during the clicking step.

Apart from the baseline choice setting, we also consider several variations of it in alternative simulation settings. These variations help us understand how our model performs under different simulation primitives. In particular, we experiment with different (1) signal-to-noise ratios (increase or decrease by 50%), (2) sample sizes (5,000, 10,000, and 20,000), (3) sparseness of the precision matrix (6%, 8%, and 10% of non-zero entries), (4) numbers of products (50, 80, and 100), (5) clicking levels for the clicking probability ($c^{\mathrm{b}} = -1.409$, 0, and 1.409)[4], (6) two alternative functional form assumptions for the clicking probability $H(.)$ (i.e., a cdf function and a linear function, refer to Appendix C.5 for more details), and (7) including uncertainties/noises associated with the perceived utilities while consumers are making clicking decisions (see Appendix C.5 for details).

4.3.2 Observing Only Clicking Dummies

In reality, we often cannot directly observe the underlying clicking probabilities p^b_{ij}, but only the clicking dummies realized from p^b_{ij}. In particular, a clicking dummy z_{ij} indicating whether a consumer i clicks product j, which is a draw from the Bernoulli distribution with a probability p^b_{ij}. In simulations, we demonstrate that, using the clicking dummies can achieve results almost as good as using the clicking probabilities when estimating Equation (4.3) in Stage-1. In particular, we solve for $\hat{\Phi}^{b,z}$ as the solution to the following optimization problem in Equation (4.9) over nonnegative definite matrices Φ^b:

$$\hat{\Phi}^{b,z} = \mathrm{argmin}_{\Phi^b}\big(\mathrm{tr}S_{zz}\Phi^b - \mathrm{logdet}\Phi^b + \lambda\|\Phi^b - \mathrm{diag}(\Phi^b)\|_1\big), \tag{4.9}$$

where S_{zz} is the empirical covariance matrix of vector $\boldsymbol{z}$ conditional on product features X.[5] In Section 4.3.4, we demonstrate that estimating Equation (4.3) and Equation (4.9) provide similar have similar Stage-1 performance in terms of the error rates, and both

[4]We choose -1.409 and 1.409 as the clicking levels to test, since they meaningfully vary the clicking probability: $0.5 * std(u_{ij})$ equals 1.409, where $std(u_{ij})$ is the standard deviation of the latent utilities.

[5]In particular, $S_{zz} = \frac{1}{N}\sum_{i=1}^{N}[(z_i - \hat{\beta}^b z_i)(z_i - \hat{\beta}^b z_i)^{\mathrm{T}}]$, in which $\hat{\beta}^b = S_{xz}S_{xx}^{-1}$ and $S_{xz} = \frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})(x_i - \bar{x})^{\mathrm{T}}$ and $S_{xx} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(x_i - \bar{x})^{\mathrm{T}}$.

$\hat{\Phi}^b$ and $\hat{\Phi}^{b,z}$ are close to the true precision matrix used in the underlying data generation process.

4.3.3 Tuning Parameter Selection

As we can see from the Graphical Lasso formulation in Equation (4.3), the tuning parameter (λ) controls the accuracy and sparsity of the model. Various approaches have been proposed to select the tuning parameter λ including cross-validation (CV), BIC, and adaptive methods. We provide details of these methods in Appendix C.4. We test the model performance under different tuning parameter selection methods. Specifically, we measure performance using false positive error rate and false negative error rate. The false positive error rate measures the percentage of zero entries in the precision matrix labeled as non-zero by our method. The false negative error rate measures the percentage of non-zero entries in precision matrix labeled as zero by our method. We specifically account for both error types because they have different consequences. Recall that in Stage II, we only estimate entries that are estimated to be non-zero from Stage I. Therefore, false positive errors result in wastes of computation time in Stage II, whereas false negative errors lead to biases in the estimation and inaccurate model predictions. We consider false negative errors to be more problematic.

The results are reported in Table 4.1, with columns representing different tuning parameter selection methods, and rows representing variations of simulated settings. Comparing columns in Table 4.1, in general the adaptive method provides the best performance. CV tends to select a large number of non-zero entries, which leads to a relatively low false negative rate but a high false positive rate. BIC, on the other hand, achieves significantly lower false positive rate, yet the corresponding false negative rate is high. The adaptive method leads to more balanced performance—the false positive rate and the false negative rate are both relatively low. Among the different variations of the adaptive methods, the adaptive cross-validation approach with squared-inverse, or A-CV, provides the best performance. Note that we test both inverse data-dependent weights and inverse-square weights, and find the performance is better when using squared inverse as weights. From now on, we use A-CV for the tuning parameter selection.

4.3.4 Stage I Estimation Performance

We now discuss the performance in Stage I. Recall that in this stage, we estimate the structure of the precision matrix, that is, the positions of non-zero entries. To demonstrate the performance of this step, we compare the structure of estimated precision matrix $\hat{\Phi}^b$ to the structure of the true underlying precision matrix Φ.

As shown in Table 4.1, our model does a good job recovering the structure of the underlying precision matrix. Focusing on the results using Adaptive-CV in Columns (5) and (6), our model achieves a false negative rate as low as 5.60% in the base case, which suggests we successfully identify most of the positions of the non-zero entries in the precision matrix. Our model also achieve a low false positive rate (6.50%) in the base case. Figure 4.1 illustrates the performance of our method graphically. The upper panel demonstrates how well we recover the true precision matrix, whereas the lower panel demonstrates how well we recover the true covariance matrix in the base case. From left to right in each panel, we present (1) the precision (covariance) matrix of consumer clicking activities, (2) the precision (covariance) matrix of consumer clicking activities conditional on product features, (3) the sparse precision matrix (or the corresponding covariance matrix) estimated using the Graphical Lasso algorithm, and (4) the true underlying precision (covariance) matrix. The zero elements in the matrix are not colored, or colored white. Positive elements are colored red. Negative elements are colored blue. The darkness of the color of each element represents how much the corresponding entry deviates from zero—darker colors indicate larger absolute values, while lighter colors indicate values closer to zero. As shown in the figure, precision matrices $\hat{\Phi}^b$ and Φ share a similar structure. Covariance matrices $\hat{\Sigma}^b$ and Σ share a similar structure too. Moreover, we notice that the estimated covariance matrix also shares a similar intensity structure with the true covariance matrix, as indicated by the darkness of the color. This observation suggests that besides using the position of non-zero entries in the second stage, we can potentially use the estimated covariance matrix itself from Stage-I to obtain a proxy for the true precision matrix in the following stage. We will test how well such approximation work in the second stage.

Next, we discuss how the model performance varies in different simulation settings. We again focus on the results using Adaptive-CV in Columns (5) and (6), since Adaptive-CV tend to be the best tuning parameter selection method compared to others.

- *Clicking Probabilities*: Our model performs better in terms of the false negative rate when we observe the clicking probabilities. This is expected, since the clicking probability gives us more information about the unobserved shocks in the latent utilities. Yet, we notice that, using the clicking dummies in the base case, we achieve similar performance—the false positive rate is even lower, and the false negative rate is not much higher. This suggests that we can use clicking

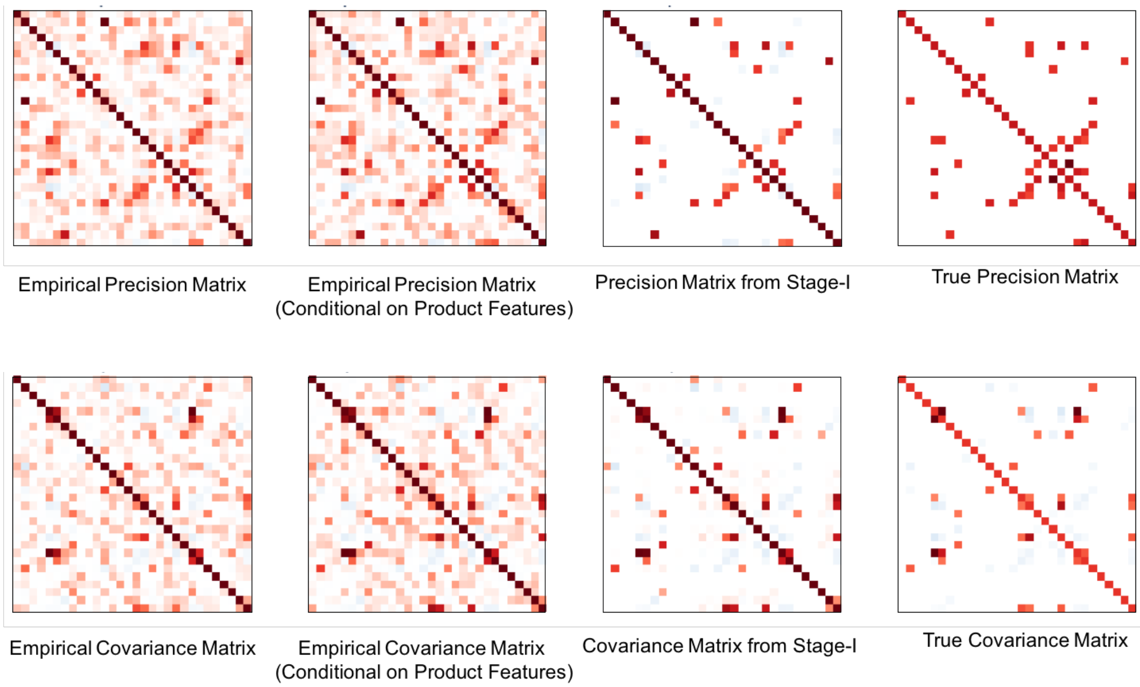


Figure 4.1: Figure Illustration of Stage I Performance [†]

Notes: The upper panel demonstrates the model performance in estimation the precision matrix, whereas the lower panel demonstrate the performance in estimating the covariance matrix. In both panels, the three figures starting from the left illustrate how we recover the true precision and covariance matrices step by step. In particular, we illustrate the empirical precision (covariance) matrix of consumer click activities (first left), the empirical precision (covariance) matrix conditional on the product features (second left), and the precision (covariance) matrix estimated using the proposed method (third left). The last figure in each panel demonstrate the true precision (covariance) matrix, simulated using the base model introduced in Section 4.3.1. The tuning parameter is selected using adaptive CV with inverse square weights. Red colors represent positive elements. Blue colors represent negative elements. The darker the color, the larger the absolute value of the corresponding element.

dummies in the Stage-1 of our model.

- *Signal-to-Noise Ratio (SNR)*: Our model performs well with higher and lower SNRs. We notice that the false negative rate is even lower with a lower SNR. This makes sense because in cases where the noise (unobserved shocks) is relative high compared to the signal (observed features), it is even easier to detect and estimate the precision matrix of the unobserved shocks.

- *Sample Size*: Our model maintains relatively low false positive and negative rates when we double the sample size or decrease it by half. As expected, a larger sample size leads to a slightly lower false negative rate.

- *Sparseness of the Underlying Precision Matrix*: Our model maintains low false negative rates in cases where there are less (6%) and more (10%) non-zero entries in the precision matrix. Yet, we note that the false positive rate is higher when the precision matrix is denser, for instance, the false positive rate is 11.50% when the precision matrix has 10% non-zero entries. This is will slightly increase the computational burden in the second stage. Even in this case, the false negative rate is still very small (7.00%).

- *Number of Products*: Our model is able to scale up to handle a large choice set. In cases where there are 50 or 80 products, the model keeps generating reasonably low false positive rates. Though the false positive rates when there are 50 or 80 products are higher, it is not as crucial, since it just means more wastes of computation time in step-2.

- *Clicking Levels (c^b)*: our model maintains relatively low false positive and negative rates with different clicking levels. We tend to have higher false negative rates in cases with lower clicking levels. This is also intuitively: lower clicking levels mean fewer products will be clicked, which reduces the number of clicking observations and leads to less accurate estimates.

- *Alternative Functional Forms $H(.)$ for Clicking Probabilities*: Our model can still identify the true precision matrix with relatively low false positive and negative rates when the clicking incidences are generated from clicking probabilities based on alternative functional forms. In fact, both low false positive and negative rates are comparable with the base case, suggesting that the performance results are not driven by the functional form assumption we made in the data generation process.

- *Uncertain Perceived Utilities*: Our model can still identify the true precision matrix with relatively low false positive and negative rates with uncertain perceived utilities. Note that x_{ij} measures potential noises associated with the perceived utilities while consumers are making clicking decisions, which is formally defined in Appendix C.5.

4.3.5 Stage II Estimation Performance

Knowing the positions of the non-zero entries in the precision matrix, we estimate their values in stage II. We compare the in- and out-of-sample performance of our model with alternative, commonly used discrete choice models: (1) the standard Probit model with i.i.d utility shocks; and (2) the random coefficient Probit model, also referred to as Mixed Probit model. Note that, in both alternative models, we keep the Probit specification ensure comparability with our model. Nevertheless, the comparison will not be affected if we use logit models. In addition to estimating our model and the two alternative models, we also estimate an approximation model as introduced in Section 4.2.3.4, which further alleviates the computational burden when the option space becomes exceptionally large.

We estimate all four models and compare the estimation results along the following dimensions: (a) how well each model recovers the true coefficients; (b) model fits in terms of in- and out-of-sample likelihoods; (c) in- and out-of-sample demand estimation accuracy. The sample used for in-sample estimation is same as the base case synthetic data. We generate another random sample of the same size and using the same model parameters for out-of-sample tests. The results from all models are discussed below.

We report the estimation results in Table 4.2, where the columns represent the true parameters and estimated parameters from four different models, namely, our full model, our approximation model, the standard Probit model with i.i.d. errors, and the random coefficient Probit model. As shown in the table, both of our models (the full and the approximation models) generate significantly more accurate estimates compared with the alternative models (the random coefficient model and the IID Probit model). The %error in parameter estimates is significantly smaller under our models (6.40% in the full model and 6.83% in the approximation model) compared to the standard Probit model (19.29%) and the random coefficient model (15.06%). The errors here %error are defined as the percentage deviation from the true parameters, i.e., %error $= |\beta - \beta^{true}|/|\beta^{true}|$.

Our method also yields good performance fitting the data. The negative in-sample and out-of-sample likelihoods are reported in the first two rows in Table 4.3. Both the full model and the approximation model achieve lower negative in- and out-of-sample likelihoods, i.e., higher in- and out-of-sample likelihoods, in comparison to the alternative models. This suggests that our proposed method better explains and predicts individual choices.

Most importantly, we are interested in how well our model predicts the demand for each product. This could be crucial building blocks for important decisions such as assortment and pricing decisions. We measure demand estimation error using MAPE (mean absolute percentage error) and MSE (mean squared error). In particular,

$$\text{MAPE} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{J}\sum_{j=1}^{J}\frac{|D^e_{j,t} - D^a_{j,t}|}{D^a_{j,t}}, \tag{4.10}$$

$$\text{MSE} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{J}\sum_{j=1}^{J}(D^e_{j,t} - D^a_{j,t})^2. \tag{4.11}$$

In Equations (4.10) and (4.11), $D^a_{j,t}$ summarizes the total actual demand of product j in period t, whereas $D^e_{j,t}$ is its counterpart in the estimation. In particular,

$$D^a_{j,t} = \sum_{i=1}^{N} d_{ij,t}, \tag{4.12}$$

where $d_{ij,t}$ indicates whether customer i arriving at time t purchases product j or not in the synthetic data; and

$$D^e_{j,t} = \sum_{i=1}^{N} \Pr(y_{ij,t} = 1), \tag{4.13}$$

where $\Pr(y_{ij,t} = 1)$ is the conditional choice probability predicted by the model.

The results are reported in the last four rows in Table 4.3. Note that the first column indicates the demand estimation accuracy using the true parameters, which help us understand the intrinsic level of variability beyond what any model can explain. Even when we know the underlying data generation process, we cannot always perfectly predict the realized demand—we cannot observe the error terms (or utility shocks) in individual consumers' utilities realized from the true data generation process, and thus, cannot perfectly predict the day-to-day demand realizations on the product level. In sum, our model demonstrates a performance that is very close to the true model, and is significantly better than the alternative models both in sample and out of sample. Taking out-of-sample MAPE for example, our model has an MAPE of 18.48%, and it is only slightly larger than the MAPE from the true model (16.81%). Our model's MAPE is significantly lower than that of the random coefficient model (45.31%) and the IID Probit model (46.44%), representing 59.21% and 60.21% reductions, respectively. Moreover, our approximation model also achieves much lower demand estimation error in comparison to the two alternative models (19.36% compared to 45.31% and 46.44%). The MAPE of the approximation model is also close to that of the true model, which is 16.81%. This result suggests that the approximation model preserves a desirable performance while saving significant computational time—in particular, from 2374 minutes to 336 minutes. The improvements in MAPE might be surprising large. Intuitively, we can provide such significant improvements in MAPE for the following reason. We do a better job in accurately estimating the changes in demands for a large number of products, many of which are slow-moving with low demand levels. For these slow-moving products, even a slightly improved *ab-*

solute demand estimation accuracy can translate into a large reduction in the *relative* demand estimation error.

4.4 An Empirical Example

We test our method using data from a leading international online retailer. We discuss the empirical setting, the estimation results and the in-sample and out-of-sample performance below.

4.4.1 Empirical Setting and Data

We obtain detailed information of product prices, product features, as well as customer clickstream and purchase decisions from a large international retailer. Specifically, the data are generated on the mobile app of the sponsor's e-commerce platform. There are three distinguishing features of our data. First, our retail partner recorded and made available to us product features and prices at the exact time when a consumer clicks or purchases a product. As product features and prices change dynamically over time, consumers substitute among the offered products. Such variations are instrumental for us to estimate the substitution patterns across products. Secondly, our retail partner provides product features capturing the matching score between a consumer and a product, and this score feeds into the retail's recommendation system. Controlling for this matching score allows us to control for the effects' of the recommendation system in real time. Lastly, the retail partner made available to us consumer IDs linking all consumer activities (clicks and purchases), which is essential for the first stage of our estimation. The co-click patterns provide information about which products are close substitutes after controlling for co-movement caused by observable product features.

Specifically, the dataset contains the clicking and purchasing history of all user searching kitchen appliances through the retailer's mobile app platform during the observation period from April 1, 2018 to April 14, 2018. There are 46 products with at least one impression during the study period and 32,217 consumers.[6] For each consumer impression, the dataset includes the product's features and its price at the time of the impression.[7] Note that the application of the discrete choice model typically requires a

[6]Note that there are in total 82 products in this category. However, if a product is never clicked by any consumer, then there is almost no information that one can use to estimate its substitution patterns with other products. Therefore, we do not include those products in our analyses.

[7]Like most retailers, our retail partner only archives a product's information when the product is viewed or purchased by a customer. Since product features and prices can vary dynamically (daily, in out setting), features and prices for products with low impressions are not always observable.

single choice setting, where consumer chooses only one option. The category of kitchen appliances is suitable for this analysis because most consumers purchase zero or one product. In particular, we observe only 10 out of 32,217 consumers purchase more than one products from this product category in a day. We therefore exclude them from the analyses. A second reason why the kitchen appliance category is a good fit for our analysis is that consumers purchase kitchen appliances infrequently. Unlike in other categories such as consumer packaged goods, consumers will not stockpile kitchen appliances. In particular, we only observe 1242 out of all 32,217 consumers, or 3.86%, have impressions on multiple days. For the purpose of our analysis, we treat the same consumer on different days as different consumers.

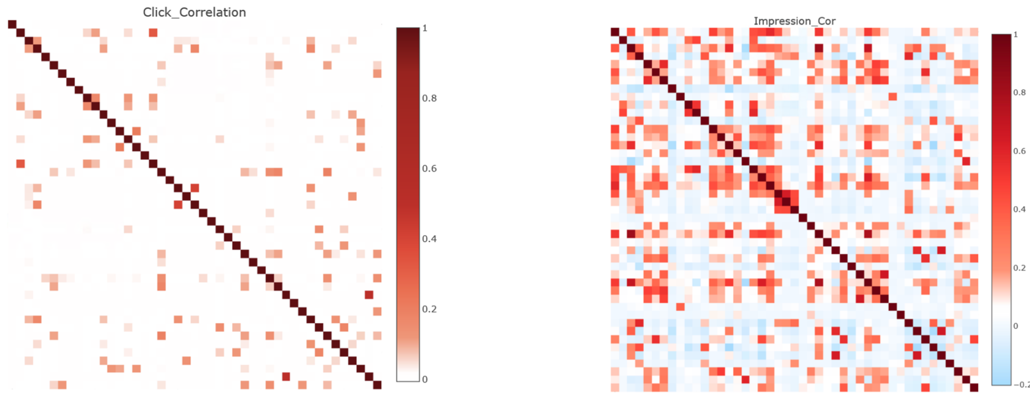
Similar to other retail settings, we observe a large portion of consumers with no-purchase. Specifically, around 96.11% of the consumers did not purchase any product. It is likely that some of the consumers who did not purchase anything are just casual "browser". With most of the consumers being "browsers", it is difficult for us to illustrate our model performance – there is little room left for any demand forecasting improvement, as we can just forecast that the consumers are not going to purchase anything and achieve a fairly high accuracy. To alleviate this issue, we refer to Feldman et al. (2018) and balance our sample,[8] so that we can focus on studying how well our model recovers the substitution patterns among all the offered products. Note that the balancing will not affect our MAPE measures, since both the denominators ($D^a_{j,t}$) and the numerators ($|D^e_{j,t} - D^a_{j,t}|$) in Equation (4.10) are being balanced proportionally.

On average, a customer clicks 0.593 products after seeing the initial product display. Among all consumers in our sample, 1,254 purchased an item. To understand the co-clicking behavior, we plot the click correlation matrix in Figure 4.2a, which represents the degree to which two products are clicked together by the same consumer. As shown in Figure 4.2a, the co-clicking matrix is fairly sparse. This supports our assumption that the underlying precision matrix is likely sparse. As a sanity check, we plot the products displayed for the same consumers at the same time in Figure 4.2b, and see that the co-impression plot is not sparse.

In our analysis, we focus on three most important product features selected using the gradient boosting method—(1) consumer preference score ($score_{ij,t}$): the

Therefore, for days on which we do not observe a product's features and price, we interpolate the missing values by taking the average of values before and after the missing period.

[8] In particular, we randomly discard 96% of the no-purchase events, and focus on the rest of the events as working sample. Namely, the sample consists of 4% of the consumers who did not purchase anything, and all the consumers who ended up purchasing the products. In this balanced sample of 2491 consumers, on average consumers buy 0.494 products.



(a) **Co-click** (b) **Co-impression**

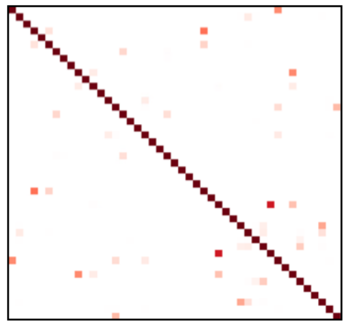
Figure 4.2: Co-click and Co-impression Behavior

individual-specific consumer preference score calculated based on *i2i* algorithm, which intuitively represents how likely a consumer is going to purchase this product based on his/her previous activities on the platform calculated by a machine learning algorithm, (2) display record ($display_{j,t}$): the number of times the product is displayed historically, (3) product price ($price_{j,t}$). The summary statistics for these product features are reported in Table in Appendix .

4.4.2 Estimation Results

We analyze this dataset using our two-stage method. We first estimate the positions of non-zero entries of a sparse precision matrix using clickstream data, which is shown in Figure 4.3. Next, we estimate the parameters in the choice model knowing the positions of the non-zero entries in the precision matrix. We estimate the parameters using the full model as well as the approximation model. Results of these two models are presented in the first two columns of Table 4.4. We also estimate the parameters using traditional models: the random coefficient model and the IID Probit model. The results are shown in the last two columns in Table 4.4. Our models have lower negative log likelihood—in other words, the highest likelihood—compared to traditional models, which suggests they fit better with the empirical observations.

We now assess the model performance in terms of demand prediction accuracy. We again compute both in-sample and out-of-sample MAPE and MSE as in the previous section. The results are reported in Table 4.5. Our models have lower in-sample and out-of-sample errors compared to traditional models. For example, the in-sample (out-of-sample) MAPE of our full model is 15.79% (16.38%), which is 37.59% (41.44%) lower



Precision Matrix of the Empirical Setting from Stage-I

Figure 4.3: Estimated Precision Matrix In Stage I

Notes: The figure demonstrates the precision matrix of the utility stocks in the empirical setting estimated from Stage I of our model. Red colors represent non-zero elements. The darker the color, the larger the absolute value of the corresponding element. In Stage II, we estimate the exact values of the non-zero elements in the precision matrix.

than the that under the random coefficient Probit model, and 57.02% (59.12%) lower than that under the standard Probit model with iid errors. Our approximation model also has similar performance to our full model, and is significantly more accurate than the alternative models. Finally, we see a similar performance gap using MSE as the performance measure.

4.5 An Application in Managing Products' Prices

So far we demonstrate our model out-performs traditional models in both simulated settings and a real-world empirical setting. Our model can be used to assist firms in making various business decisions such as assortment planning, inventory management, pricing and so forth. In this section, we illustrate how it can be used to make better pricing decisions. In particular, we show that our model obtains a better estimate of product substitution patterns compared to traditional models. As a result, we are able to make better price recommendations that lead to larger profit or revenue gains. For the purpose of evaluating potential profit/revenue gains, we utilize the synthetic data generated in Section 4.3. Since we know the underlying true demand model that generates the data, we can use it to simulate demands and profits/revenues under price recommendations made using our model and the traditional models, and then compare them.

4.5.1 Own- and Cross-Price Elasticities

As in Equation (4.13), the expected demand for product j predicted by model m is calculated as

$$\begin{aligned} D_j^m &= \sum_{i=1}^{N} \Pr^m(y_{ij}=1|X;\theta^m) \\ &= \sum_{i=1}^{N} \int\limits_{\tilde{\epsilon}_{i1}=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}(\theta^m)-v_{i1}(\theta^m)} \cdots \int\limits_{\tilde{\epsilon}_{ij}=-v_{ij}(\theta^m)}^{\infty} \cdots \int\limits_{\tilde{\epsilon}_{iJ}(\theta^m)=-\infty}^{\tilde{\epsilon}_{ij}+v_{ij}(\theta^m)-v_{iJ}(\theta^m)} F_{\theta^m}(\tilde{\epsilon}_i) \quad d\tilde{\epsilon}_{i1}\ d\tilde{\epsilon}_{i2}...d\tilde{\epsilon}_{iJ}, \end{aligned} \tag{4.14}$$

where X is the given set of product features, and θ^m summarizes the set of parameters under choice model m. In particular, we evaluate four models: (1) the full version of our model (i.e., m = Full Model), (2) our model with approximation (i.e., m = App Model), (3) standard Probit model with iid error terms (i.e., m = IID Probit), and (4) random coefficient Probit model (i.e., m = Random Coef). In our full model, the parameter space $\theta^{\text{Full Model}} = \{\alpha, \beta, \gamma\}$. Recall α and β represent the coefficients of observed product features, and γ represents the non-zero elements in the precision matrix. In our approximation model, $\theta^{\text{App Model}} = \{\alpha, \beta, \boldsymbol{a}\}$, where $\boldsymbol{a}$ is used to polynomial approximation of non-zero elements in the precision matrix. In the IID Probit model, $\theta^{\text{IID Probit}} = \{\alpha, \beta\}$). Finally, in the random coefficient model, $\theta^{\text{Random Coef}} = \{\mu(\alpha), \sigma(\alpha), \mu(\beta), \sigma(\beta)\}$, where μ and σ represent the mean and standard deviation of the corresponding parameter.

Based on these demand functions, we can estimate the substitution patterns across different products. Specifically, we compute own-price and cross-price elasticities, indicating how a marginal change in one product's price will affect its own demand and other products' demand. Let e_{kj} denote the elasticity of product j's demand to product k's price. With any given demand model m, we have

$$e_{kj}^m = \frac{\%\Delta D_j^m}{\%\Delta p_k} = \frac{\Delta D_j^m}{\Delta p_k} \cdot \frac{p_k}{D_j^m}. \tag{4.15}$$

Let matrix E^m denote the own- and cross-price elasticity matrix under model m. Recall that CCP under normal distribution does not have a close-form solution, and as a result, we calculate the elasticity matrices numerically. We first evaluate the expected demand under a specific price vector. We then change the price for one product by a small amount (i.e., $\Delta p = 0.001$), and re-evaluate the expected demand under the new price vector. To demonstrate how well our models recover the true substitution pattern compared to traditional models, we measure the deviation of the estimated elasticity matrices ($E^{\text{Full Model}}$, $E^{\text{App Model}}$, $E^{\text{IID Probit}}$, and $E^{\text{Random Coef}}$) from the true elasticity matrix (E^{True}). The true elasticity matrix is calculated based on the true model of the underlying data generate process—in particular, the kj^{th} entry e_{kj}^{True} in the elasticity matrix E^{True} is calculated as: $e_{kj}^{True} = \frac{\%\Delta D_j^{True}}{\%\Delta p_k} = \frac{\Delta D_j^{True}}{\Delta p_k} \cdot \frac{p_k}{D_j^{True}}$, where D_j^{True} is the aggregate demand for product j predicted by the true model. After obtaining the true elasticity matrix E^{True}, we measure how far estimated elasticity matrices deviates from E^{True}. Specifically, we measure the deviations using both the absolute and the euclidean distances:

$$Dist_{Abs}^{m} = \sum_{k} \sum_{j} |e_{kj}^{m} - e_{kj}^{True}|, \tag{4.16a}$$

$$Dist_{Euc}^{m} = \sqrt{\sum_{k} \sum_{j} (e_{kj}^{m} - e_{kj}^{True})^2}. \tag{4.16b}$$

The results are presented in Table 4.6. As shown in the table, the elasticity matrices estimated using our models are closer to the true elasticity matrix. In particular, the distances under our models are only half of those under the IID Probit and the random coefficient models. The improvement brought by our model in recovering the "true" substitution pattern is an important building block for making better pricing decisions, as illustrated in the next subsection.

4.5.2 Price Recommendations and Profit Gains

As described in the earlier subsection, our model better recovers the underlying substitution pattern among products. It has many business implications such as better demand forecasting, assortment planning, pricing and promotional decisions.[9] In this section, we illustrate one such application in making pricing decisions. Consider a retailer selling J substitutable products. The retailer either maximizes profit or maximizes revenue while subject to margin constraints. Given any demand model m, the retailer chooses optimal prices for J products over a time period $t = 1, 2, ..., T$,

[9]Note that, our model can capture the effect of a recommendation system by capturing in X how this system recommends a product to a consumer (for example, including *score* in the empirical model in Section 4.4.1). Estimating such model helps us understand consumer demands under alternative designs of the recommendation system, namely, alternative Xs. However, in other cases where multiple recommendation channels exist and it is hard to capture all systems' effects in X, our model predicts consumer demands and offers pricing strategies under the current recommendation systems. In these cases, we would need to re-estimate our model if the recommendation systems change.

$\boldsymbol{p} = \{p_{j,t}\}_{j=1,2,\ldots,J;t=1,2,\ldots,T}$, to either (1) maximize the total profit π^m_{profit} based on Equation (4.17), or (2) maximize the total revenue π^m_{revenue} based on Equation (4.18):

$$\boldsymbol{p}^{*m}_{\text{profit}} :=_{\boldsymbol{p}} \pi^m_{\text{profit}}(\boldsymbol{p}) =_{\boldsymbol{p}} \sum_{t=1}^{T} \sum_{j=1}^{J} D^m_{j,t}(\boldsymbol{p}_t)\,(p_{j,t} - c_j)\,, \tag{4.17}$$

or

$$\begin{aligned} \boldsymbol{p}^{*,m}_{\text{revenue}} :=_{\boldsymbol{p}} \pi^m_{\text{revenue}}(\boldsymbol{p}) =_{\boldsymbol{p}} \sum_{t=1}^{T} \sum_{j=1}^{J} D^m_{j,t}(\boldsymbol{p}_t) \cdot p_{j,t}, \\ \text{s.t., } p_{j,t} > (1 + MPM) \cdot c_j, \end{aligned} \tag{4.18}$$

where $D^m_{j,t}$ represents the demand for product j in period t predicted by model m (recall from Equation (4.14)), c_j is the unit marginal cost of product j, and MPM is the minimum profit margin for the retailer, and $\boldsymbol{p}^{*m}_{\text{profit}}$ and $\boldsymbol{p}^{*m}_{\text{revenue}}$ denotes the optimal price vectors for J products under demand model m when the retailer is maximizing profit and revenue, respectively. We solve the maximization problem under each demand model m, including our full model, our approximated model, the Probit model with iid error term, and the random coefficient model. With the price vectors recommended by each demand model m, we then simulate the "true" expected profit/revenue under each m. Note that since we know the "true" demand model, which is used in the synthetic data generation process, we can simulate the expected sales and the expected profit or revenue given any price vector.

We report the profit/revenue performance in Table 4.7. In the upper and lower two sections of Table 4.7, namely, "Profit when Maximizing Profits" and "Revenue when Maximizing Revenues", we report the average profit and revenue obtained under each model when the retailer's goal is to maximize profit and revenue, respectively. Our models, the full version and the approximated version, achieve better overall performance than the IID Probit model and the random coefficient model. Specifically, when the retailer's goal is to maximize profit, our full model achieves 32.267 profit, which is 13.47% higher than the IID Probit model (28.438) and 29.05% higher than the random coefficient model (25.003). We also divide the products into slow-moving products and fast-moving products, which are the one third of the products with the least and most amount of sales. We notice that our model does particularly well in improving profits among slow-moving products. Our full model achieves more than twice of the profit among slow-moving products than the IID Probit model, while trading-off a slightly lowered profit with fast-moving products. When the retailer's goal is to maximize revenue, our full model achieves 29.05 revenue, which is 17.40% higher than the IID Probit model (106.110) and 8.72% higher than the random coefficient model (114.583). Similar to the results of profit maximization, we find that our model does particularly well in improving revenues among slow-moving products.

4.6 Conclusion and Contribution

To improve demand estimation and pricing decisions, an important first step is to achieve a realistic understanding of the substitution patterns among products. Yet, it is challenging to do so among a large number of products and in the complex online environment. We propose a methodology that combines a machine learning tool (namely, Graphical Lasso) with choice models to tackle this challenge. There are two major innovations in our method. First, we leverage consumer clickstream data to identify the product network based on the error terms. Second, we introduce Graphical Lasso method to the choice modeling framework in identifying the product substitution patterns.

Our method performs well using synthetic data under various scenarios (classified based on precision matrix density, signal-to-noise ratio, high-low clicking levels, etc) in recovering the true underlying product network. The method is also robust to several types of noises in consumers clicking step—our method could still successfully recover the true underlying product network, even when we consider the fact that (1) consumers might not be able to evaluate the product utilities while clicking, or (2) might have a limited attention span to click all the good products. Compared to classical multinomial Probit model with iid error terms, our method consistently offers better in- and out-of-sample fit with the data (the out-of-sample negative Log likelihood reduced from 29097 to 27460 using synthetic data), and provides more accurate demand forecasts—out-of-sample MAPE is reduced from 46.44% to 18.48%, representing a 60.21% reduction. Our method even outperforms the more advanced random coefficient model—the out-of-sample MAPE reduced from 45.31% to 18.48% on average, suggesting an improved demand estimation accuracy by 59.21%. By doing so, we are able to recommend better pricing decisions—in comparison with the IID Probit model (the random coefficient Probit model), we provide an 13.47% (29.05%) increase in total profit when maximizing profits, and an average 17.40% (8.72%) increase in total revenue when maximizing revenues. Applying our method to the real online retail setting, we show that our method continues offering better in- and out-of-sample performances, and providing more accurate demand forecasts—out-of-sample demand forecasts are on average improved by 59.12% and 41.44% in comparison with the IID Probit model and the random coefficient model respectively.

Practically, our proposed method can be readily applied by online retailers to various business settings with large choice sets. This includes demand estimation for inventory and transshipment decisions, as well as a variety of other operational decisions, such as promotion and assortment planning decisions. We find that capturing the product network via Graphical Lasso approach will lead to a more accurate demand estimation (avoiding around 50% demand estimation error), and more accurate own-price and cross-price elasticity estimates. This highlights the importance of incorporating information learned from consumers' clicking activities as well as combining state-of-art machine learning methods with the choice modeling framework.

Our analysis is, of course, not without limitations. First, this study focuses on the demand forecasting on the product-level, which assists retailers making important operational decisions, such as demand forecasting and product pricing. It would be interesting to see how clickstream information could help inform demand estimation for each consumers, which could help retailers make promotion decisions for each individual consumers or groups of consumers. Second, we learn the product network based on error terms by observing the consumer co-click activities among the offered products. An important extension is learning the product network for new products based on pre-existing products. This could help retailers determine ordering quantities and making pricing decisions for newly introduced products.

Table 4.1: Precision Matrix Estimation Accuracy on Simulated Data

	CV		BIC		A-CV		A-BIC	
	FN (1)	FP (2)	FN (3)	FP (4)	FN (5)	FP (6)	FN (7)	FP (8)
Main	5.60%	8.90%	5.56%	2.90%	5.60%	6.50%	5.60%	10.40%
Observe Click Probability	0.00%	13.04%	0.00%	5.31%	0.00%	7.73%	0.00%	14.49%
Signal to Noise Ratio (SNR)								
SNR + 50%	5.60%	7.20%	13.89%	1.69%	5.60%	5.30%	5.60%	10.90%
SNR - 50%	0.00%	10.90%	2.78%	4.35%	0.00%	9.20%	0.00%	13.00%
Sample Size								
Sample = 5000	5.60%	7.20%	8.33%	2.17%	5.60%	4.10%	5.60%	9.90%
Sample = 20000	2.80%	12.80%	2.78%	3.14%	2.80%	8.70%	2.80%	11.80%
Sparseness of Precision Matrix								
Chol 6%	3.20%	8.40%	3.23%	2.39%	3.20%	6.90%	3.20%	11.00%
Chol 10%	4.70%	16.00%	9.30%	5.16%	7.00%	11.50%	4.70%	21.10%
Number of Products								
Prod = 50	3.70%	22.24%	23.46%	2.48%	6.17%	16.60%	3.70%	27.12%
Prod = 80	5.80%	36.92%	43.48%	3.64%	7.25%	28.50%	3.87%	45.01%
Clicking Levels								
$c^b = -1.409$	8.30%	3.60%	8.33%	3.38%	8.30%	3.40%	2.80%	8.50%
$c^b = 1.409$	2.80%	14.70%	2.78%	4.11%	2.80%	10.90%	2.80%	17.40%
Alternative Functional Forms $H(.)$								
cdf	5.60%	7.50%	25.00%	1.69%	5.60%	4.30%	5.60%	7.70%
linear	2.80%	11.80%	8.33%	2.42%	2.80%	7.50%	2.80%	14.00%
Uncertain Perceived Utilities								
$\xi_{ij} = 1.409$	2.80%	8.00%	13.89%	1.93%	2.80%	6.00%	2.80%	8.70%
$\xi_{ij} = 2.818$	5.60%	9.20%	33.33%	0.48%	8.30%	5.80%	2.80%	12.60%

Notes: The table reports false positive error rate (FP) and false negative rate (FN). FP measures the percentage of zero entries in the precision matrix labeled as non-zero by our method, and false negative error rate. FN measures the percentage of non-zero entries in precision matrix labeled as zero by our method. The columns represent different tuning parameter selection methods. In particular, Columns (1) and (2) represent results from cross validation (CV). Columns (3) and (4) represent results from BIC. Columns (5) and (6) represent results from Adaptive-CV with inverse-squared data-dependent weights. Columns (7) and (8) represent results from Adaptive-BIC with inverse-square data-dependent weights. We also test adaptive models with non-squared weights, and find their performance are worse. The rows represent different simulation settings. Among them, the base case is defined as: 30 products, 20 periods, 500 customers per period, $\alpha = 0$, $\beta_1 = 0.5631$, $\beta_2 = 1.2568$, $\beta_3 = -0.0144$, the sparseness of Φ equals 8%, SNR = 5.339, consumer i clicks product j with probability $p^b_{ij} = \frac{exp(u_{ij}-u_{i0})}{1+exp(u_{ij}-u_{i0})}$. Note that x_{ij} measures potential noises associated with the perceived utilities while consumers are making clicking decisions, which is formally defined in Appendix C.5.

Table 4.2: Accuracy of Model Estimates

	True Model	Model Estimates: Full Model		App Model		IID Probit		Random Coef	
β_1	0.563	0.584	(0.031)	0.585	(0.032)	0.544	(0.026)	0.560	(0.026)
β_2	1.257	1.203	(0.040)	1.170	(0.036)	0.855	(0.026)	0.936	(0.028)
β_p	-0.014	-0.013	(0.000)	-0.013	(0.000)	-0.011	(0.000)	-0.012	(0.000)
$\sigma_{\beta 1}$								-0.496	(0.064)
$\sigma_{\beta 2}$								0.642	(0.065)
$\sigma_{\beta p}$								0.003	(0.000)
%error		6.40%		6.83%		19.29%		15.06%	
Log likelihoood		26992.2		27230.3		29148.7		28959.4	

Note: (1) App Model is short for the approximation model defined in Section 4.2.3.4. (2) %error is defined as $|\beta - \beta^{true}|/|\beta^{true}|$. (3) Numbers in the parentheses are standard errors of the estimates.

Table 4.3: Model Fit and Demand Estimation Accuracy

	True Model	Full Model	App Model	IID Probit	Random Coef
In-Sample Negative LL	27555	27362	27395	29196	29196
Out-of-Sample Negative LL	27608	27460	27433	29097	29079
In sample MAPE	14.81%	16.89%	17.86%	45.42%	44.16%
Out-sample MAPE	16.81%	18.48%	19.36%	46.44%	45.31%
In sample MSE	5.22E-05	6.60E-05	7.44E-05	3.83E-04	3.72E-04
Out-sample MSE	6.15E-05	6.95E-05	7.62E-05	4.07E-04	3.94E-04

Note: (1) App Model is short for the approximation model defined in Section 4.2.3.4. (2) LL is short for Log Likelihood. (3) MAPE stands for mean absolute percentage error defined in Equation (4.11). MSE stands for mean squared error defined in Equation (4.10).

Table 4.4: Parameter Estimates from The Empirical Example

	Full Model		App Model		Random Coef		IID Probit	
β_{price}	-0.138	(0.005)	-0.112	(0.003)	-0.141	(0.006)	-0.080	(0.001)
$\beta_{display}$	6.068	(0.473)	4.620	(0.394)	10.249	(0.425)	6.185	(0.188)
β_{score}	12.622	(0.634)	9.126	(0.376)	12.644	(0.634)	6.752	(0.205)
$\sigma_{\beta price}$					-0.074	(0.003)		
$\sigma_{\beta display}$					-0.000	(2.318)		
$\sigma_{\beta score}$					10.003	(1.693)		
In-Sample Negative LL	1.95E+03		1.97E+03		2.12E+03		2.74E+03	

Note: (1) App Model is short for the approximation model defined in Section 4.2.3.4. (2) LL is short for Log Likelihood.

Table 4.5: Model Performance on The Empirical Example

	Our Model	App Model	Random Coef	IID Probit
In-Sample MAPE	15.79%	18.66%	25.30%	36.74%
Out-of-Sample MAPE	16.38%	18.65%	27.97%	40.07%
In-Sample MSE	2.59E-04	2.58E-04	4.11E-04	1.10E-03
Out-of-Sample MSE	2.05E-04	2.27e-04	4.40E-04	1.30E-03

Note: (1) App Model is short for the approximation model defined in Section 4.2.3.4. (2) MAPE stands for mean absolute percentage error defined in Equation (4.11). MSE stands for mean squared error defined in Equation (4.10).

Table 4.6: The Distance from the True Elasticity Matrix to the Elasticity Matrix of Alternative Models

	Full Model	App Model	IID Probit	Random Coef
All Products				
$Dist_{Abs}$	24.967	27.783	47.646	47.193
$Dist_{Euc}$	1.849	2.048	3.965	3.906
Own Price				
$Dist_{Abs}$	19.661	21.973	40.720	40.072
$Dist_{Euc}$	1.944	2.186	4.667	4.581
Cross Price				
$Dist_{Abs}$	20.442	22.607	34.212	34.278
$Dist_{Euc}$	1.231	1.307	1.641	1.653

Note: (1) App Model is short for the approximation model defined in Section 4.2.3.4. (2) "All Products" considers all elasticities among J products; "Own Price" considers only own price elasticity, namely, e^m_{kj} when $k = j$; "Cross Price" considers only cross price elasticity, namely, e^m_{kj} when $k \neq j$. (3) $Dist_{Abs}$ stands for the absolute distance between two elasticity matrices defined in Equation (4.16a). $Dist_{Euc}$ stands for the euclidean distance between two elasticity matrices defined in Equation (4.16b).

Table 4.7: Average Profit and Revenue Over Time Obtained by Each Demand Model

	Full Model	App Model	IID Probit	Random Coef
Profit when Maximizing Profits				
All Products	32.267	32.540	28.438	25.003
Slow Moving	2.132	1.983	0.914	0.575
Medium Moving	7.603	7.819	5.284	4.176
Fast Moving	22.532	22.737	22.239	20.252
Revenue when Maximizing Revenues				
All Products	124.572	125.080	106.110	114.583
Slow Moving	14.121	13.272	8.138	9.886
Medium Moving	38.530	39.041	23.406	27.044
Fast Moving	71.921	72.767	74.565	77.652

Note: (1) "Profit when Maximizing Profits" stands for the average profit across periods under the optimal pricing in maximizing profits recommended by each model, defined in Equation (4.17). "Revenue when Maximizing Revenues" stands for the average revenue across periods under the optimal pricing in maximizing revenues recommended by each model, defined in Equation (4.18). (2) App Model is short for the approximation model defined in Section 4.2.3.4. (3) "All Products" stands for the profit/revenues for all J products averaging across periods; "Slow Moving" ("Fast Moving") stands for the profit/revenues for the slow-moving (fast-moving) products averaging across periods.

APPENDIX A

Appendices to Chapter 1

A.1 Proofs for Theorem 2.3.1 — Designers' Equilibrium Decisions

To solve designers' problem (Equation (2.3)), we first solve for designers' equilibrium choices of the number of conceptual objectives to incorporate ($r^*(N)$) and the number of design trials to generate ($m^*(N)$), given a number of participating designers (N). Then we solve for the equilibrium number of participating designers (N^*). As a preparation, we exploit the Gumbel distribution's property (recall that $V_\cdot$'s in Equation (2.2) follow a Gumbel distribution), and compute designer i's probability of winning, given i's choices (r_i and m_i), and all other competing designers' choices (r and m):

$$\Pr(\text{i wins}) = \frac{m_i \exp(v(r_i)/\mu)}{m_i \exp(v(r_i)/\mu) + (N-1)m \exp(v(r)/\mu)}. \tag{A.1}$$

We analyze the symmetric pure strategy Nash equilibrium, where every designer chooses the same r^* and m^*. Such an equilibrium exists in our game, per Theorem 1.2 in Fudenberg and Tirole (2000).

Equilibrium $\mathbf{r}^*$: Designer i's first-order condition (F.O.C.) with respect to concept formation decision r_i (in Equation (2.3)) is $\frac{\partial U_i}{\partial r_i} = 0$. Substituting Pr(i wins) from Equation (A.1), and using symmetry ($m = m_i = m^*$ and $r = r_i = r^*$), we can simplify the marginal change in the winning chance with respect to r_i as $\frac{\partial \Pr\{\text{i wins}\}}{\partial r_i} = \frac{\nu w}{\mu} \frac{N-1}{N^2}$ ($S_r < r^*$); $\frac{\partial \Pr\{\text{i wins}\}}{\partial r_i} = \frac{w}{\mu} \frac{N-1}{N^2}$ ($S_r > r^*$). With that, we solve the F.O.C. and obtain a unique symmetric solution: $\overline{r}^g = \frac{\ln(\frac{N-1}{N^2} \frac{\nu w}{\mu} \frac{A}{c_2} \frac{1}{\ln(1/p)} - \frac{c_g}{c_2} \frac{1}{\ln(1/p)})}{\ln(1/p)}$ ($S_r < r^*$); $\overline{r} = \frac{\ln(\frac{N-1}{N^2} \frac{w}{\mu} \frac{A}{c_2} \frac{1}{\ln(1/p)})}{\ln(1/p)}$ ($S_r > r^*$). Since $c_g > 0$ and $\nu \in (0, 1)$, we have $\overline{r}^g < \overline{r}$, for any N. Note that if designers guess and incorporate more than the disclosed conceptual objectives, $r_i = \overline{r}^g$; if they incorporate all and only the disclosed ones, $r_i = S_r$; if they incorporate just a subset of the disclosed ones, $r_i = \overline{r}$. We have

$$r^*(N) = \begin{cases} \overline{r}^g, \text{ if } S_r < \overline{r}^g & \text{(Region 0)} \\ S_r, \text{ if } \overline{r}^g \le S_r \le \overline{r} & \text{(Region I)} \\ \overline{r} \quad \text{if } S_r > \overline{r} & \text{(Region II)} \end{cases} \tag{A.2}$$

Equilibrium $\mathbf{m}^*$: Designer i's F.O.C. with respect to trial effort decision m_i (in Equation (2.3)) is $\frac{\partial U_i}{\partial m_i} = 0$. Substituting Pr(i wins) from Equation (A.1) into the F.O.C., and using symmetry ($m = m_i = m^*$ and $r = r_i = r^*$), the F.O.C. reduces to $A\frac{(N-1)m^*}{(m^*N)^2} - c_3 = 0$, which has a unique solution:

$$m^*(N) = \frac{A(N-1)}{N^2 c_3}. \tag{A.3}$$

Equilibrium $\mathbf{N}^*$: Having derived participants' equilibrium choices ($r^*(N)$ and $m^*(N)$) under a given number of participants N, we compute the equilibrium number of participating designers (N^*). We add a subscript to N^* to help distinguish N^* in three scenarios (specified in Equation (A.2)): N_0^* stands for N^* when $S_r < \bar{r}^g$ (designers guess and incorporate more than S_r); N_1^* stands for N^* when $\bar{r}^g \leq S_r \leq \bar{r}$ (designers incorporate all and only S_r); N_2^* stands for N^* when $S_r > \bar{r}$ (designers incorporate only a subset of S_r).

- Region 0: Equilibrium conditions imply that designers are indifferent between participating or not, i.e., $\Pr(\text{i wins}) \cdot A - C_i(r_i{=}\bar{r}^g, m_i{=}m^*) = s$. In a symmetric equilibrium, $\Pr(\text{i wins}) = 1/N$. After substituting $\bar{r}^g$ (see Equation (A.2)), we can rearrange and express the equilibrium number of participating designers through the implicit function (we prove below that N_0^* is unique):

$$N^* = N_0^* = \frac{\sqrt{4Y(X+1)+X^2}-X}{2Y}, \text{ where } X = \frac{\nu w}{\mu \ln(1/p)} \text{ and } Y = \frac{s + c_g \cdot (\bar{r}^g - S_r)^+ + c_1 \cdot S_r - \frac{c_g}{\ln(1/p)}}{A}. \tag{A.4}$$

- Region I: In equilibrium, designers are indifferent between participating or not, i.e., $\Pr(\text{i wins}) \cdot A - C_i(r_i{=}S_r, m_i{=}m^*) = s$. In a symmetric equilibrium, $\Pr(\text{i wins}) = 1/N$. We can simplify and solve for the equilibrium number of participating designers as

$$N^* = N_1^* = \sqrt{\frac{A}{s + c_1 \cdot S_r + c_2 \cdot (1/p)^{S_r}}}. \tag{A.5}$$

- Region II: In equilibrium, designers are indifferent between participating or not, i.e., $\Pr(\text{i wins}) \cdot A - C_i(r_i{=}\bar{r}, m_i{=}m^*) = s$. In a symmetric equilibrium, $\Pr(\text{i wins}) = 1/N$. After substituting $\bar{r}$ (see Equation (A.2)), we can rearrange and solve for the equilibrium number of participating designers as:

$$N^* = N_2^* = \frac{\sqrt{4Y(X+1)+X^2}-X}{2Y}, \text{ where } X = \frac{w}{\mu \ln(1/p)} \text{ and } Y = \frac{s + c_1 \cdot S_r}{A}. \tag{A.6}$$

Next, we establish that the three regions are split by two thresholds: S_r^{gs} such that $S_r^{\text{gs}} = \bar{r}^g(N_0^*(S_r^{\text{gs}}))$ (where $\bar{r}^g$ is from Equation (A.2), and N_0^* is from Equation (A.4)); S_r^{ic} such that $S_r^{\text{ic}} = \bar{r}(N_2^*(S_r^{\text{ic}}))$ (where $\bar{r}$ is from Equation (A.2), and N_2^* is from Equation (A.6)). In words, S_r^{gs} is the number of conceptual objectives the seeker discloses, under which designers are not willing to guess any more conceptual objectives; S_r^{ic} is the number of conceptual objectives the seeker discloses, under which designers are willing to incorporate exactly what are disclosed. Online Appendix EC.6 formally characterizes S_r^{gs} and S_r^{ic}, and proves that we can discuss N_0^*, N_1^* and N_2^* separately in Region 0, Region I and Region II; therefore, the three regions are self-contiguous and do not overlap. Turning to N_0, its uniqueness follows from three observations: (i) S_r is a function of N_0^*, namely, $S_r = \frac{1}{c_g - c_1}\left(s - \frac{c_g}{\ln(1/p)} + c_g \bar{r}^g(N_0^*) - A \cdot \frac{X+1-N_0^* X}{(N_0^*)^2}\right)$ (by re-arranging Equation (A.4)). Hence, given N_0^* there is a unique S_r. (ii) From the analysis below, we know that N_0^* is strictly increasing with S_r. (iii) N^* is a continuous with S_r, and N_1^* is unique given S_r (per Equation (A.5)). Hence, when S_r increases to approach S_r^{gs} (recall that S_r^{gs} is the cutoff of Region 0 and Region I), $\lim_{S_r \to S_r^{\text{gs}-}} N^*(S_r)$ is unique. With (i)-(iii) established, suppose N_0^* is not unique. Per (ii), there will be more than one increasing trajectories of $N_0^*(S_r)$. Per (iii) the increasing trajectories will merge at S_r^{gs}. We then have multiple S_rs corresponding to a N_0^*, contradicting (i).

Region 0: the number of participants increases with more disclosed conceptual objectives. The proof is conducted using continuous N. Suppose with S_r disclosed objectives, the number of participating designers is N and designers are guessing and incorporating more than S_r (i.e., $N_0^*(S_r) = N$). This indicates the N^{th} designer's expected utility equals the opportunity cost (i.e., $U_i(N; S_r) = s$), whereas the designers after the N^{th} designer have lower expected utility than the opportunity cost ($U_i(N+1; S_r) < s$). If the seeker discloses slightly more conceptual objectives $S_r' = S_r + \Delta(S_r)$, let us see the change in $N_0^*(S_r)$. First note that, under the same level of participation, the designer's total participation cost is lower as more objectives are disclosed, since $c_g > c_1$ and r^* does not change with S_r in Region 0 (under the same N). Consequently, designers' expected utilities increase with S_r (given the same N). Specifically, with a higher S_r, the N^{th} designer's expected utility is even higher than before (i.e., $U_i(N; S_r') > U_i(N; S_r) = s$), implying the N^{th} designer will still join the contest under S_r'. Moreover, the expected utility is strictly greater than the opportunity cost (i.e., $U_i(N; S_r') > s$), indicating that there will be designer(s) join after the the N^{th} designer. Hence, as more conceptual objectives are disclosed, the number of designers participating in the contest strictly increases; i.e., N_0^* strictly increases in S_r.

Regions I and II: the number of participants decreases with more disclosed conceptual objectives. It is obvious that in Region I, N_1^* decreases with more disclosed conceptual objectives (S_r). In Region II, we have, $\frac{\partial N_2^*}{\partial S_r} = \frac{\partial N_2^*}{\partial Y} \cdot \frac{\partial Y}{\partial S_r} = -\frac{-X\sqrt{X^2+4XY+4Y}+X^2+2XY+2Y}{2Y^2\sqrt{X^2+4XY+4Y}} \cdot \frac{c_1}{A} \leq 0$. That is, N_2^* also decreases with more disclosed conceptual objectives (S_r) in Region II. Moreover, N^* is continuous with S_r. Hence, the number of participating designers always decreases with S_r.

Lastly, we can substitute N^* (Equations (A.5)-(A.6)) into Equations (A.2)-(A.3), and get m^* and r^* under the equilibrium number of participating designers.

A.2 Proof for Lemma 2.3.1

$$
\begin{aligned}
\max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \Pi_s(S_r, S_g) &= \max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \left[\mathbb{E}_{\boldsymbol{\epsilon}} \max_{i \in N^*} V_i^*(r^*, m^*) \right] \text{ (from Problem (2.4))} \\
&= \max_{S_r \le \overline{S}_r, S_g \le \overline{S}_g} \left[v(r^*(S_r)) + \mu \ln \left(m^*(S_g, N^*(S_r)) \cdot N^*(S_r) \right) \right], \\
&\quad \text{(Theorem 2.3.1 and Gumbel distribution property)} \\
&= \max_{S_r \le \overline{S}_r} \left[w \cdot r^*(S_r) + \mu \ln A \frac{N^*(S_r) - 1}{N^*(S_r)} \right] + \max_{S_g \le \overline{S}_g} \left[-\mu \ln (h(S_g)) \right].
\end{aligned}
$$

A.3 Proof and Additional Detail for Proposition 2.3.1

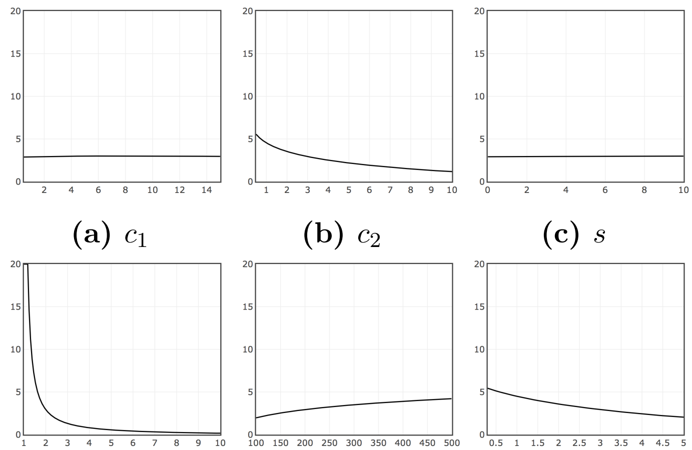
Theoretical Properties of S_r^*. We separately discuss the optimal number of conceptual objectives to disclose (S_r^*) under three scenarios: In Scenario 0, we assume that N^* follows Equation (A.4), m^* follows Equation (A.2), and $r^* = \overline{r}^g$. In Scenario 1, we assume that N^* follows Equation (A.5), m^* follows Equation (A.2), and $r^* = S_r$ regardless of the size of S_r. In Scenario 2, we assume that N^* follows Equation (A.6), m^* follows Equation (A.2), and $r^* = \overline{r}$ regardless of the size of S_r. In words, in Scenario 0 (1/2) we assume that the equilibrium formulas of Region 0 (I/II) will prevail regardless of the size of S_r. We define S_r^{1*} and S_r^{2*} as the optimal numbers of conceptual objectives the seeker should disclose in Scenario 1 and 2 under $\overline{S}_r = \infty$, and show their properties in Lemmas A.3.1 and A.3.2, respectively.

Lemma A.3.1 *In Scenario 1, the seeker's profit* Π_s *is first increasing and eventually decreasing with* S_r*. When* $c_1 = 0$ *(i.e., no designers' problem framing cost),* $\Pi_s(S_r)$ *is a concave function maximized at* $S_{r,c_1=0}^{1*} = \frac{\ln\left(2[\sqrt{A(A+s(\frac{\mu}{w})^2 \ln^2 \frac{1}{p} + 2s\frac{\mu}{w}\ln\frac{1}{p})} + (A-2s) - s\frac{\mu}{w}\ln\frac{1}{p}]\right) - \ln\left(c_2(\frac{\mu}{w}\ln\frac{1}{p}+2)^2\right)}{\ln(1/p)}$.
When $c_1 \ge 0$, $\Pi_s(S_r)$ *is maximized at* S_r^{1*}*, and* $S_r^{1*} \le S_{r,c_1=0}^{1*}$.

Lemma A.3.2 *In Scenario 2, the seeker's profit* Π_s *decreases with* S_r.

Lemmas A.3.1 and A.3.2 are proved in Online Appendices EC.8 and EC.9, respectively. Using EC.6, note that Scenario 1 corresponds to Region I in Lemma 5 when $S_r \le S_r^{\text{ic}}$; Scenario 2 corresponds to Region II in Lemma 5 when $S_r > S_r^{\text{ic}}$. With this observation and the continuity of $\Pi_s(S_r)$, Lemmas A.3.1 and A.3.2 imply that $S_r^* \le \min\{\overline{S}_r, S_r^{1*}, S_r^{\text{ic}}\}$; i.e., the seeker should not always disclose all the conceptual objectives he cares about.

Comparative Statics of S_r^*. Next, we examine how S_r^* changes with the nature of the design problem. Specifically, we simulate S_r^* at different values of the model primitives, including: cost of digesting each conceptual objective (in the design problem framing stage) (c_1); unit cost of each design concept formulating attempt (c_2); opportunity cost (s); concept formulating difficulty ($1/p$); award level (A); and relative problem uncertainty ($\lambda := \frac{\mu}{w}$). Figure A.1 illustrates the simulation results. The relationships between S_r^* and model primitives are consistent with intuition.



(a) c_1 **(b)** c_2 **(c)** s

(d) $1/p$ **(e)** A **(f)** λ

Figure A.1: Optimal Number of Conceptual Objectives S_r^* $(A = 200, s = 2, p = 0.5, c_2 = 3, \mu = 6, w = 2, c_1 = 2)$**.**

A.4 Examples of the Types of Information in Problem Specifications in the Data

Table A.1: Three Examples of the Types of Information in Problem Specifications in the Data

Company	**Industry**	**Q3** (Conceptual objectives)	**Q4** (Execution Guidance)	**Q5** (Execution Guidance)
Quackers	Retailing	It's a fun brand for kids. Very happy, upbeat and never quit on your dreams. Community and fun.	Image only. A cartoon duck with a sailor hat - not too much like Donald Duck. The hat should be green, the duck bill should be yellow.	Duck face a bit like this but with hat and less body visible. (A hyperlink to company website).
Intertwined	Consulting and Professional Services	We want to generate a sense of progress to our customers, as well as communicate that we are a professional organization. We're here to help our clients cut through industry baggage to help them really focus on what will help them do business well.	Image with the company name, or image only. We won't consider text only logos.	None
Rise	Social Media Advertising and Marketing	We are looking for a design that is professional but not too corporate. We want to convey reliability, fun and professionalism.	None.	Not really.

A.5 Image Comparison Algorithm – SIFT

To classify designers' submissions into "distinct designs" and "variations" in our large-scale data, we employ *Scale-Invariant Feature Transform (SIFT)*. SIFT is an algorithm proposed by Lowe (1999) that detects and describes local features in images, and comprises four steps: (1) *Extract SIFT Feature Vectors:* From a pair of images (A and B), descriptors of the key points are extracted by identifying *SIFT feature vectors* in scale space; these vectors are constructed to robustly capture the structural properties of the images. (2) *Match SIFT Feature Vectors:* For each feature vector D_i^A in image A, its shortest Euclidean distance ($d(D_i^A, D_j^B)$) to each of the SIFT feature vectors in image B (D_j^B) is calculated. Feature vectors D_i^A and D_k^B are defined as a matched pair if and only if $\frac{d(D_i^A,D_k^B)}{d(D_i^A,D_j^B)} < 2/3$ for all j. (This is the default threshold in Lowe's paper (1999), essentially the nearest-neighbor approach.) (3) *Compute Similarity Ratio:* Let $\gamma_{A,B} = \frac{N^m_{A,B}}{\min\{N_A,N_B\}}$, where $N^m_{A,B}$ is the number of matched pairs between image A and image B, and N_A (N_B) is the total number of feature vectors extracted from image A (B). (4) *Classify Images Pairs as Similar or Different:* Similarity Ratio $\gamma_{A,B}$ is used to classify images A and B as either similar or different. The higher the ratio, the more similar the two images are (a ratio of 1 means the two images are exactly the same). In our empirical analysis, we classify two submissions as similar if $\gamma_{A,B} \geq 0.4$. If a designer's new submission (a logo image) is very similar to any of her prior submissions (based on the SIFT score), we classify the new submission as a "variation"; otherwise, we consider the new submission a "distinct design".

A.6 Power Analysis

We use power analyses to evaluate the ability of our empirical study to detect variables characterizing the problem specification that have an impact on the number of participants ($(No.Designers)_q$) and designers' trial efforts ($(No.Submissions)_{i,q}$). Based on the R-squares before and after the inclusion of problem specification-related variables and an alpha level of 0.05 with Bonferroni adjustment, the sample size analysis indicates that we need sample sizes of 174, 182 and 2987 to achieve a power of 0.8 to detect variables characterizing the problem specification in Table 2.2's Columns (1)(3)(5) respectively. With our current sample sizes, we have enough data to detect any significant variables. Equivalently, we can calculate a post-hoc level of statistical power. With our current sample sizes, we can achieve a power above 0.95 in all regressions, indicating that our tests have adequate statistical power.

A.7 Additional Empirical Results about the Impact of Problem Specification

Table A.2: Impact of Different Categories of Execution Guidelines on No.Submissions Per Designer

	Dependent Variable: $(NO\ Submission)_{i,q}$	In Post-Update Multiple-Choice Questions?
Elements of Execution Guidance		
Colors Dummy	0.177* (0.083)	Yes
Logo Style Dummy	0.101 (0.105)	Yes
Shapes Dummy	0.214* (0.097)	No
Font Dummy	0.029 (0.116)	Yes
Usage Dummy	0.239** (0.078)	Yes
Art Styles Dummy	0.163* (0.077)	No
Resources Dummy	0.132· (0.076)	No
Control Variables		
$(No.Concepts)_q$	0.013 (0.010)	
Concept Similarity	0.149 (0.479)	
$(No.Designers)_q$	−0.009* (0.004)	
$(No.Updates)_q$	0.204*** (0.034)	
DoW, Month, Year Dummies	Yes	
Industry, Creator Dummies	Yes	
Observations	11,757	
R^2 (Adjusted R^2)	0.480 (0.304)	
Residual Std. Error (F Statistic)	3.317 (df = 8790)	
Note:	*p<0.05; **p<0.01; ***p<0.001	

Common vs. Rare Keywords We measure the keyword usage frequency with Zipf frequency ($ConceptFreq$, which ranges from 0-8, with more frequently used words having large $ConceptFreq$s). (The Zipf frequency of a word is the base-10 logarithm of the number of times it appears per billion words.) For each problem specification, we compute the average and the minimum usage frequency among the manually-coded conceptual objective keywords ($(Avg.ConceptFreq)_q$ and $((Min.ConceptFreq)_q)$. We consider and estimate models in Table A.3, and find that $(Avg.ConceptFreq)_q$ and $(Min.ConceptFreq)_q$ are both negatively and significantly associated with $(No.Designers)_q$, suggesting that rare conceptual objective keywords are relatively less discouraging for designers participation.

Categories vs. Intensities of Execution Guidance We separately consider the effect of offering more categories of execution guidance ($(No.GuideCats)_q$), and that of offering more detailed guidance in each category, i.e., more keywords per category ($(\frac{No.GuideWords}{No.GuideCats})_q$). We regress $(No.Submissions)_{i,q}$ on both $(No.GuideCats)_q$ and $(\frac{No.GuideWords}{No.GuideCats})_q$, report the results in Table A.4, and find that $(No.GuideCats)_q$ is significantly positively associated with $(No.Submissions)_{i,q}$, whereas $(\frac{No.GuideWords}{No.GuideCats})_q$ is not.

Table A.3: Nuance Findings for Conceptual Objectives – Does Frequency of a Keyword in English Affect Participation?

	Dependent variable: $(No.Designers)_q$	
Conceptual Objectives		
$(No.Concepts)_q$	−0.396** (0.137)	−0.604*** (0.124)
Concept Similarity	0.176** (0.064)	0.178** (0.064)
$(Avg.ConceptFreq)_q$	−2.776** (0.887)	
$(Min.ConceptFreq)_q$		−1.316** (0.405)
Execution Guidance		
$(No.GuideWords)_q$	−0.042 (0.104)	−0.040 (0.104)
Control Variables		
$(len_{Q3}/No.Concepts)_q$	0.043 (0.130)	−0.037 (0.130)
$(No.Updates)_q$	0.011 (0.427)	−0.012 (0.427)
DoW, Month, Year, Industry Dummies	Yes	Yes
Observations	441	441
R^2	0.271	0.272
Adjusted R^2	0.198	0.199
Residual Std. Error (df = 400)	9.392	9.383
F Statistic	3.709***	3.735***
Note:		*p<0.05; **p<0.01; ***p<0.001

Table A.4: Nuance Findings for Nested Structure of Execution Guidance

	Dependent variable: $(No.Submissions)_{i,q}$
Conceptual Objectives	
$(No.Concepts)_q$	0.010 (0.010)
Concept Similarity	0.002 (0.005)
Execution Guidance	
$(No.GuideCats)_q$	0.172*** (0.025)
$(\frac{No.GuideWords}{No.GuideCats})_q$	0.048 (0.052)
Control Variables	
$(No.Designers)_q$	−0.010* (0.004)
$(No.Updates)_q$	0.199*** (0.034)
Day-of-Week Fixed Effects	Yes
Month Fixed Effects	Yes
Year Fixed Effects	Yes
Industry Fixed Effects	Yes
Creator Fixed Effects	Yes
Observations	11,509
R^2	0.483
Adjusted R^2	0.306
Residual Std. Error	3.333 (df = 8576)
F Statistic	2.729***
Note:	*p<0.05; **p<0.01; ***p<0.001

A.8 Additional Tables

Data evidence for problem framing cost increasing with conceptual objectives. Table A.5 reports the results of the regression of coders' reading time for a problem specification on the characteristics of the problem specification. The results suggest that the coders spend more time reading problem specifications with more conceptual objectives, but not on those with more execution guidelines. This supports our modeling assumption that the problem-framing cost only changes with the number of conceptual objectives.

Table A.5: Regression of Reading Time

		Dependent Variable: $(Reading\,Time)_q$	
$(No.Concepts)_q$		0.071*** (0.004)	
$(len_{Q3}/No.Concepts)_q$		0.054*** (0.004)	
$(No.GuideWords)_q$		0.001 (0.003)	
Observations	441	R^2 (Adjusted R^2)	0.622 (0.587)
Residual Std. Error	0.304 (df = 403)	F Statistic	17.889***
Note:			*p<0.05; **p<0.01; ***p<0.001

Robustness checks – alternative measures for designer trial efforts. As we can see from Table A.6, the two new measures for designer i's trial effort, $(No.DesignStyle)_{i,q}$ and $(Variations/Style)_{i,q}$, are both significantly positively correlated with $(No.GuideWords)_q$, but neither is significantly associated with $(No.Concepts)_q$, which again supports Hypotheses 2a and 2b – more seeker execution guidance leads to more submissions from each designer (both in terms of "distinct designs" and their "variations"), but more conceptual objectives do not.

Robustness checks – sub-sample tests only with contests that did not have updates. The results reported in Table A.7 are qualitatively the same as those estimated with the full sample (reported in Table 2.2), indicating that the main findings are not affected by how we control for $(No.Updates)_q$.

Table A.6: Robustness Checks for reduced2 (Alternative Measures for Trial Efforts)

	Dependent Variable:	
	$(No.DesignStyle)_{i,q}$	$(Variations/Style)_{i,q}$
Conceptual Objectives		
$(No.Concepts)_q$	0.001 (0.003)	0.010 (0.006)
Concept Similarity	−0.000 (0.001)	−0.000 (0.003)
Execution Guidance		
$(No.GuideWords)_q$	0.006** (0.002)	0.028*** (0.005)
Control Variables		
$(len_{Q3}/No.Concepts)_q$	0.001 (0.003)	0.011 (0.006)
$(No.Designers)_q$	−0.0004 (0.001)	−0.006** (0.002)
$(No.Updates)_q$	0.060*** (0.009)	0.042* (0.019)
Day-of-Week Fixed Effects	Yes	Yes
Month Fixed Effects	Yes	Yes
Year Fixed Effects	Yes	Yes
Industry Fixed Effects	Yes	Yes
Creator Fixed Effects	Yes	Yes
Observations	11,757	11,757
R^2	0.396	0.403
Adjusted R^2	0.193	0.202
Residual Std. Error (df = 8795)	0.890	1.895
F Statistic (df = 2961; 8795)	1.949***	2.003***
Note:		*p<0.05; **p<0.01; ***p<0.001

Table A.7: Sub-Sample Tests Containing Only Contests that Did Not Have Updates

	Dependent Variables:		
	$(No.Designers)_q$		$(No.Submissions)_{i,q}$
Conceptual Objectives			
$(No.Concepts)_q$	-0.498^{*} (0.196)	-0.665^{***} (0.173)	0.025 (0.013)
Concept Similarity	0.173· (0.095)	0.163^{*} (0.077)	0.000 (0.006)
Very Short Objectives			
$(No.Concepts)_q * \mathbb{I}_q^{ShortCcpt}$	-5.101 (2.708)		
$\mathbb{I}_q^{ShortCcpt}$	16.513^{*} (7.822)		
Execution Guidance			
$(No.GuideWords)_q$	-0.016 (0.141)	-0.042 (0.141)	0.046^{***} (0.011)
Control Variables			
$(len_{Q3}/No.Concepts)_q$	-0.087 (0.191)	-0.002 (0.184)	0.019 (0.013)
$(No.Designers)_q$			0.002 (0.005)
Day-of-Week Fixed Effects	Yes	Yes	Yes
Month Fixed Effects	Yes	Yes	Yes
Year Fixed Effects	Yes	Yes	Yes
Industry Fixed Effects	Yes	Yes	Yes
Creator Fixed Effects	No	No	Yes
Observations	260	260	6,968
R^2	0.325	0.309	0.548
Adjusted R^2	0.201	0.190	0.338
Residual Std. Error	9.505 (df = 219)	9.573 (df = 221)	3.056 (df = 4758)
F Statistic	2.632^{***}	2.595^{***}	2.614^{***}

Note: *p<0.05; **p<0.01; ***p<0.001

A.9 Properties of $S_r^{\mathbf{gs}}$ and $S_r^{\mathbf{ic}}$

The following lemmas characterize S_r^{gs} and S_r^{ic}.

Lemma A.9.1 *If S_r^{gs} exists, then it is unique and divides the number of conceptual objectives the seeker discloses into two regions. Region 0: when $S_r \leq S_r^{gs}$, designers are willing to guess and incorporate more than the disclosed objectives (i.e., $S_r < \overline{r}^g(N_0^*(S_r))$); Region I: when $S_r > S_r^{gs}$, designers do not guess and only incorporate from the disclosed objectives (i.e., $S_r \geq \overline{r}^g(N_0^*(S_r))$).*

Lemma A.9.2 *There exists a unique S_r^{ic}. S_r^{ic} divides the number of conceptual objectives the seeker discloses into two regions. Region I: when $S_r \leq S_r^{ic}$, designers are willing to incorporate all the disclosed objectives (i.e., $S_r \leq \overline{r}(N_2^*(S_r))$); Region II: when $S_r > S_r^{ic}$, designers incorporate only a subset of all the disclosed objectives (i.e., $S_r > \overline{r}(N_2^*(S_r))$).*

We prove Lemmas A.9.1-A.9.2 by: (A) S_r^{ic} exists, (B) S_r^{ic} is unique, (C) if S_r^{gs} exists, $S_r^{\text{gs}} < S_r^{\text{ic}}$ and S_r^{gs} is unique.

(A) Existence of $S_r^{\mathbf{ic}}$. We prove S_r^{ic} exists by showing:

Lemma A.9.3 *(1) $\exists S_r \to 0$ s.t. $\bar{r}(N_2^*(S_r)) \geq S_r$, and (2) $\exists S_r > 0$ s.t. $\bar{r}(N_2^*(S_r)) < S_r$.*

We now provide proofs for (1) and (2) in Lemma A.9.3. (1) By assumption, $r^* = S_r$ if S_r is sufficiently small (see Section 2.3.3), hence we know that $\exists S_r \to 0$ s.t. $\bar{r}(N_2^*(S_r)) \geq S_r$. (2) We are interested in studying contests where $N^* > 1$ (see Section 2.3.3); hence, under the problem parameters we focus on, $\exists S_r$ s.t. $N^*(S_r) > 1$. This implies when S_r is sufficiently small, $N^* > 1$, since when S_r increases, N^* decreases (shown in Appendix A.1). On the other hand, when S_r becomes extremely large, the number of participating designers approaches zero: $\lim_{S_r\to\infty} N^*(S_r) = 0$ (both $\lim_{S_r\to\infty} N_1^*(S_r) = 0$, and $\lim_{S_r\to\infty} N_2^*(S_r) = \lim_{S_r\to\infty} \frac{\sqrt{4Y(X+1)+X^2}-X}{2Y} = 0$ (by L'Hopital's Rule)). Hence, given continuity of $N^*(S_r)$, $\exists S_r > 0$ s.t. $N^*(S_r) = 1$ (denoted as $S_{r,N=1}$). If the seeker discloses $S_{r,N=1}$ conceptual objectives, the number of objectives designers are willing to incorporate is $\lim_{S_r\to S_{r,N=1}} \bar{r}(N^*(S_r)) = \lim_{S_r\to S_{r,N=1}} \bar{r}(N^*{=}1) = -\infty$, which implies $\bar{r}(N^*(S_r)) < S_r$, and thus the number of participating designers in the equilibrium is N_2^*. Hence, $\exists S_r$ s.t. $\bar{r}(N_2^*(S_r)) < S_r$.

(B) Uniqueness of S_r^{ic}. If there are multiple S_r^{ic}'s s.t. $S_r^{\text{ic}} = \bar{r}(N_2^*(S_r^{\text{ic}}))$, call the smallest one among them $S_r^{\text{ic}(1)}$. Lemma A.9.3 implies that $\frac{\partial \bar{r}(N_2^*(S_r))}{\partial S_r}|_{S_r^{\text{ic}(1)}} \leq 1$. Next, at any point S_r, we can compare the left and right limits of $\frac{\partial \bar{r}(N_2^*(S_r))}{\partial S_r}$, namely left limit $\lim_{S_r\to S_r^-} \frac{\Delta \bar{r}(N_2^*(S_r))^-}{\Delta S_r} = \frac{\bar{r}(N_2^*(S_r)) - \bar{r}(N_2^*(S_r - \Delta S_r))}{\Delta S_r}$, and right limit $\lim_{S_r\to S_r^+} \frac{\Delta \bar{r}(N_2^*(S_r))^+}{\Delta S_r} = \frac{\bar{r}(N_2^*(S_r + \Delta S_r)) - \bar{r}(N_2^*(S_r))}{\Delta S_r}$. Note that $\bar{r}(N_2^*(S_r))$ is a function of S_r through the equilibrium number of participating designers (N_2^*). Hence, corresponding to the left and right limits, we can define the changes in N_2^* as ΔN_2^{*-} and ΔN_2^{*+} respectively. Algebra based on Equation (A.6) implies that, $\frac{\partial N_2^*}{\partial S_r} < 0$ and $\frac{\partial^2 N_2^*}{\partial S_r^2} > 0$, from which we know $|\Delta N_2^{*+}| < |\Delta N_2^{*-}|$ (i.e., the number of participants decreases with S_r at a decreasing speed). Based on the formula for $\bar{r}(N_2^*)$, we can write $\Delta \bar{r}$ as a function of ΔN_2^*: $\Delta \bar{r} = \left(\frac{N_2^{*-}|\Delta N_2^*|-1)(N_2^*)^2}{(N_2^{*-}|\Delta N_2^*|)^2(N_2^*-1)}\right)/(\ln \frac{1}{p})$. Now, we can compare $\lim_{S_r\to S_r^-} \frac{\Delta \bar{r}^-}{\Delta S_r}$ and $\lim_{S_r\to S_r^+} \frac{\Delta \bar{r}^+}{\Delta S_r}$:

- When $N_2^*{>}2$: $\frac{\partial \Delta \bar{r}}{\partial |\Delta N_2^*|}{>}0$. In this case, $\Delta \bar{r}^+{<}\Delta \bar{r}^-$ since $|\Delta N_2^{*+}|{<}|\Delta N_2^{*-}|$. Hence, $\lim_{S_r\to S_r^+} \frac{\Delta \bar{r}^+}{\Delta S_r} {<} \lim_{S_r\to S_r^-} \frac{\Delta \bar{r}^-}{\Delta S_r}$.
- When $N_2^*{<}2$: $\bar{r}$ increases with N_2^*, so decreases with S_r ($\Delta \bar{r}^-{<}0$, $\Delta \bar{r}^+{<}0$). So $\lim_{S_r\to S_r^-} \frac{\Delta \bar{r}^-}{\Delta S_r}{<}0$, $\lim_{S_r\to S_r^+} \frac{\Delta \bar{r}^+}{\Delta S_r}{<}0$.

The assessment of the left and right limits indicates that, $\frac{\partial \bar{r}}{\partial S_r}$ is either decreasing or negative. Therefore, we have that $\forall S_r > S_r^{\text{ic}(1)}$, and $\frac{\partial \bar{r}}{\partial S_r}|_{S_r} < 1$. This suggests $S_r^{\text{ic}(1)}$ is the only S_r, s.t. $\bar{r}(N_2^*(S_r)) = S_r$. This result, combined with Lemma A.9.3, suggests that S_r^{ic} divides the number of conceptual objectives the seeker discloses S_r into two regions: Region I, when $S_r \leq S_r^{\text{ic}}$, $S_r \leq \bar{r}(N_2^*(S_r))$; Region II, when $S_r > S_r^{\text{ic}}$, $S_r > \bar{r}(N_2^*(S_r))$.

(C) $S_r^{\mathbf{gs}} < S_r^{\mathbf{ic}}$, and $S_r^{\mathbf{gs}}$ is unique. If S_r^{gs} exists, $S_r^{\text{gs}} < S_r^{\text{ic}}$, since for any $N^* = N$, $\bar{r}^g(N) < \bar{r}(N)$. We prove that S_r^{gs} is unique by contradiction. Suppose there are multiple S_r^{gs}s, among which the very first one is $S_r^{\text{gs},1}$ (i.e., $\bar{r}^g(N_0^*(S_r^{\text{gs},1})) = S_r^{\text{gs},1}$). Then there should be $S_r' > S_r^{\text{gs},1}$, under which designers "guess". However, $\bar{r}^g(N_0^*(S_r')) < \bar{r}^g(N_0^*(S_r^{\text{gs},1})) = S_r^{\text{gs},1} < S_r'$, which suggests that designers cannot be "guessing" under S_r'. Note the first inequality is from the following facts. We know that $S_r' > S_r^{\text{gs},1}$ and under both points designers "guess", and we also know that when designers "guess", N_0^* increases with S_r; hence, $N_0^*(S_r') > N_0^*(S_r^{\text{gs},1})$. We further know that $\bar{r}^g$ is a decreasing function of N_0^*, hence, we get the first inequality.

A.10 Extensions

We extend our main model to allow for the following possibilities: (1) overlap/similarity across conceptual objectives; (2) diminishing weights/importance among conceptual objectives. We are able to show in both cases, the qualitative results from the model, hypotheses derived from the model predictions, and the managerial implications (the optimal way of providing problem specifications) remain intact.

A.10.1 Extension (I): Overlap Across Conceptual objectives

We extend our main model to consider the level of overlap across conceptual objectives. For example, designers might consider "friendly" and "welcoming" more overlapping than "friendly" and "professional". Intuitively, satisfying multiple conceptual objectives with a large overlap is likely to be easier than satisfying the same number of objectives with a small overlap; on the other hand, if a design already satisfies one objective, satisfying another objective that overlaps a lot with the first one might only generate limited marginal improvement to the design's quality. To capture these effects, we make the following adjustments to the main model. Given the level of overlap (denoted as $1 - \alpha$ where $\alpha \in [0, 1]$, i.e., when α is smaller, objectives overlap significantly), we assume that conditional on one objective being satisfied, (1) the probability that another objective is satisfied by a random design concept generated is p^α (when objectives are more overlapping, p^α is closer to 1); (2) the weight carried by any additional objective is αw (when objectives are more overlapping, αw is closer to 0). Correspondingly, when a designer incorporates r objectives, her cost of concept formulation is $(1/p)^{1+\alpha(r-1)}$; and the base quality of her designs is $w(1+\alpha(r-1))$. Another way to think about this is that the number of "orthogonal" objectives is $1+\alpha(r-1)$ (when objectives are almost completely overlapping, the number of "orthogonal" objectives approaches 1; on the other extreme, as the level of overlap goes to zero, the number approaches r). We solve for designers' equilibrium behavior, given the seeker's problem specification (S_r, S_g, and α):

Lemma A.10.1 *In a crowdsourcing contest, where the equilibrium number of participating designers equals $N^{\alpha,*}$, the unique symmetric equilibrium for $r^{\alpha,*}$ and $m^{\alpha,*}$ are as follows. The equilibrium number of objectives a designer incorporates is* $r^{\alpha,*} = \begin{cases} \overline{r}^{\alpha,g} & \text{if } S_r < \overline{r}^{\alpha,g} \\ S_r & \text{if } \overline{r}^{\alpha,g} \leq S_r \leq \overline{r} \\ \overline{r}^{\alpha} & \text{if } S_r > \overline{r}^{\alpha} \end{cases}$, *where* $\overline{r}^{\alpha,g} = \frac{ln(\frac{N^{\alpha,*}-1}{(N^{\alpha,*})^2} \cdot \frac{\nu w}{\mu} \cdot \frac{A}{c_2} \cdot \frac{1}{\ln 1/p} \cdot p^{1-\alpha} - \frac{c_g}{c_2} \frac{1}{\ln(1/p)} \cdot p^{1-\alpha})}{\alpha \ln \frac{1}{p}}$ *and* $\overline{r}^{\alpha} = \frac{ln(\frac{N^{\alpha,*}-1}{(N^{\alpha,*})^2} \cdot \frac{w}{\mu} \cdot \frac{A}{c_2} \cdot \frac{1}{\ln 1/p} \cdot p^{1-\alpha})}{\alpha \ln \frac{1}{p}}$*; and the equilibrium number of design trials each designer generates is* $m^{\alpha,*} = \frac{A(N^{\alpha,*}-1)}{(N^{\alpha,*})^2 c_3}$*, where* $c_3 = h(S_g)$*. The equilibrium number of participating designers ($N^{\alpha,*}$) first increases (when $S_r < \overline{r}^{\alpha,g}$) and then decreases (when $S_r \geq \overline{r}^{\alpha,g}$) with more disclosed conceptual objectives S_r; but $N^{\alpha,*}$ does not change with the amount of execution guidance S_g.*

Lemma A.10.1 generalizes Theorem 2.3.1, and considers the level of overlap among conceptual objectives. This lemma has the same intuition as Theorem 2.3.1 — the direction of the relationship between designers' equilibrium behaviors and the number of conceptual objectives and execution guidelines in seekers' problem specification remains unchanged; and Takeaways 1-2 also remain intact. Note that this extension provides an additional insight: when the number of disclosed conceptual objectives is moderate or large (designers do not guess and incorporate undisclosed conceptual objectives), as α decreases (i.e., the level of overlap increases), the number of participating designers increases (first strictly increases, and then stays the same). We in fact find empirical support for this additional insight: the number of participating designers in a contest increases with the semantic similarity (*Concept Similarity* on a scale of 1-100) among the manually coded keywords for conceptual objectives (see Table 2.2 Column (3) for the detailed regression results). Furthermore, our recommendation that seekers should disclose as much execution guidance as possible, but disclose conceptual objectives only up to a certain level stays the same (hence Takeaway 3 remain intact). The proofs are straight-forward generalization of proofs in Appendices A.1 and A.3, omitted to save space.

A.10.2 Extension (II): Descending Weights Among Conceptual Objectives

We extend our main model to consider the possibility that conceptual objectives could carry different weights/importance to the seeker. Each conceptual objective (denoted as s_r) carries a weight of w_{s_r}, which represents the quality improvement of a design if this additional objective s_r is satisfied. The seeker's objectives ($s_r = 1, ..., \bar{S}_r$) are sorted in descending importance, with smaller s_r indicating more important objectives (i.e., w_{s_r} is decreasing with s_r). We assume $w_{s_r} = w\Phi^{s_r-1}$, where $\Phi \in [0, 1]$. The parameter Φ captures how "skewed" the distribution of w_{s_r} is, i.e., when Φ is large, all the objectives are very similar in terms of their importance to the seeker; whereas when Φ is small, the importance drops quickly with s_r, and only a small number of objectives are important. We assume that w_{s_r} is common knowledge. As objectives are equally difficult to achieve but of different importance, the seeker would always want to disclose objectives in the order of decreasing importance (i.e., from the most important to the least important). Given the seeker's problem specification (S_r, S_g, and Φ) we solve for designers' equilibrium behavior:

Lemma A.10.2 *In a crowdsourcing contest, where the equilibrium number of participating designers equals $N^{\Phi,*}$, the unique symmetric equilibrium for $r^{\Phi,*}$ and $m^{\Phi,*}$ are as follows. The equilibrium number of objectives a designer incorporates is* $r^{\Phi,*} = \begin{cases} \overline{r}^{\Phi,g} & \text{if } S_r < \overline{r}^{\Phi,g} \\ S_r & \text{if } \overline{r}^{\Phi,g} \leq S_r \leq \overline{r} \\ \overline{r}^{\Phi} & \text{if } S_r > \overline{r}^{\Phi} \end{cases}$, *where* $\overline{r}^{\Phi,g} = \frac{ln(\frac{N^{\Phi,*}-1}{(N^{\Phi,*})^2} \cdot \frac{\nu w}{\mu(1-\Phi)} \cdot \frac{A}{c_2} \cdot \frac{\ln 1/\Phi}{\ln 1/p} - \frac{c_g}{c_2} \cdot \frac{\ln 1/\Phi}{\ln 1/p})}{\ln \frac{1}{p\Phi}}$ *and* $\overline{r}^{\Phi} = \frac{ln(\frac{N^{\Phi,*}-1}{(N^{\Phi,*})^2} \cdot \frac{w}{\mu(1-\Phi)} \cdot \frac{A}{c_2} \cdot \frac{\ln 1/\Phi}{\ln 1/p})}{\ln \frac{1}{p\Phi}}$; *and the equilibrium number of design trials each designer generates is* $m^{\Phi,*} = \frac{A(N^{\Phi,*}-1)}{(N^{\Phi,*})^2 c_3}$, *where* $c_3 = h(S_g)$. *The equilibrium number of participating designers ($N^{\Phi,*}$) first increases (when $S_r < \overline{r}^{\Phi,g}$) and then decreases (when $S_r \geq \overline{r}^{\Phi,g}$) with more disclosed conceptual objectives S_r; but $N^{\Phi,*}$ does not change with the amount of execution guidance S_g.*

Lemma A.10.2 generalizes Theorem 2.3.1, and considers the possible descending weights/importance among conceptual objectives. It has the same intuition as Theorem 2.3.1: the direction of the relationship between designers' equilibrium behaviors and the numbers of conceptual objectives and execution guidelines in the seeker's problem specification remains unchanged; and Takeaways 1-2 remain intact. Furthermore, our recommendation that it is optimal for seekers to disclose as much execution guidance as possible stays the same, and our suggestion that seekers should only disclose conceptual objectives up to a certain level becomes even more salient (hence Takeaway 3 remain intact). Intuitively, as the importance of the conceptual objectives decreases with s_r, the quality improvement from incorporating an additional objective is smaller (a lower "quality effect"). Yet, the negative "competition effect" from disclosing more conceptual objectives remains (a higher cost for designers to digest and incorporate disclosed conceptual objectives, which lowers the number of participating designers). The proofs are straightforward generalization of proofs in Appendix A.1 and Appendix A.3, which we omit here given the limited space.

A.11 Proof for Lemma A.3.1

We denote Π_s in Scenario 1 (defined in Appendix A.3) as Π_s^1. In Scenario 1, all designers incorporate all the disclosed objectives (S_r). In this case, the seeker solves the following optimization problem:

$$\max_{S_r \leq \bar{S}_r} \Pi_s^1(S_r; S_g) = \max_{S_r \leq \bar{S}_r} w \cdot S_r + \mu \ln A(1 - \frac{1}{N_1^*(S_r)}) - \mu \ln(h(S_g)), \text{ where } N_1^* \text{ is from Equation (A.5).} \tag{A.7}$$

A.11.1 With Zero Cost for Designers to Digest Each Conceptual objective (i.e., $c_1 = 0$)

We calculate the second derivative of $\Pi_s^1(S_r; S_g|c_1 = 0)$ with respect to S_r as follows:

$$\frac{\partial^2 \Pi_s^1(S_r;S_g|c_1=0)}{\partial {S_r}^2} = -\frac{\mu c_2^2 \ln^2 \frac{1}{p} (\frac{1}{p})^{S_r} [c_2 (\frac{1}{p})^{S_r} + 2s(1-B)]}{4A^2 B^3 (B-1)^2} < 0, \text{ where } B = \sqrt{\frac{c_2 (\frac{1}{p})^{S_r} + s}{A}} = \frac{1}{N^*} \in (0,1], \tag{A.8}$$

which shows $\Pi_s^1(S_r; S_g|c_1 = 0)$ is a concave function w.r.t S_r.

Maximum of $\Pi_s^1(S_r; S_g|c_1 = 0)$: For a concave function $\Pi_s^1(S_r; S_g|c_1 = 0)$, the global maximum is reached when $\frac{\partial \Pi_s^1(S_r;S_g|c_1=0)}{\partial S_r} = 0$. We can write this F.O.C. as $[x \ln \frac{1}{p} c_2 \mu + 2w(s + xc_2)]^2 = 4w^2 A(s + xc_2)$, where $x := (\frac{1}{p})^{S_r}$. The roots for this quadratic function are:

$$\begin{cases} x_1 = \frac{2[-\sqrt{{c_2}^2 A w^2 (A w^2 + s\mu^2 \ln^2 \frac{1}{p} + 2s\mu w \ln \frac{1}{p})} + (c_2 w^2 (A-2s) - c_2 s \mu w \ln \frac{1}{p})]}{{c_2}^2 (\mu \ln \frac{1}{p} + 2w)^2}; \\ x_2 = \frac{2[\sqrt{{c_2}^2 A w^2 (A w^2 + s\mu^2 \ln^2 \frac{1}{p} + 2s\mu w \ln \frac{1}{p})} + (c_2 w^2 (A-2s) - c_2 s \mu w \ln \frac{1}{p})]}{{c_2}^2 (\mu \ln \frac{1}{p} + 2w)^2}. \end{cases} \tag{A.9}$$

Because ${c_2}^2 A w^2 (A w^2 + s\mu^2 \ln^2 \frac{1}{p} + 2s\mu w \ln \frac{1}{p}) - (c_2 w^2 (A - 2s) - c_2 s\mu w \ln \frac{1}{p})^2 = {c_2}^2 s w^2 (A - s)(\mu \ln \frac{1}{p} - 2w)^2 > 0$, we know that $x_1 < 0$, $x_2 > 0$. Hence, x_2 is the unique maximum (by definition, $x = (\frac{1}{p})^{S_r}$ is positive). Therefore, the optimal number of conceptual objectives to disclose is $S_{r,c_1=0}^{1*} = \min\{\overline{S}_r, \frac{\ln x_2}{\ln(1/p)}\}$, where x_2 is from Equation (A.9).

A.11.2 With Positive Cost for Designers to Digest Each Conceptual Objective (i.e., $c_1 > 0$)

The Seeker's Profit Π_s^1 is Eventually Decreasing with S_r. As mentioned in Section 2.3.3, we are interested in studying contests where $N^* > 1$; hence, under the problem parameters we focus on, $\exists S_r$ s.t. $N^*(S_r) > 1$. This implies when S_r is sufficiently small, $N^* > 1$, since when S_r increases, N^* further decreases (shown in Appendix A.1). Also, when S_r is sufficiently small, specifically, $S_r \leq S_r^{\text{ic}}$, the equilibrium number of participating designers is N_1^*, and thus $N_1^* > 1$. On the other hand, when S_r becomes extremely large, N_1^* is approaching zero: $\lim_{S_r \to \infty} N_1^*(S_r) = 0$. Hence, given continuity of $N_1^*(S_r)$, $\exists S_r > 0$ s.t. $N_1^*(S_r) = 1$ (we denote it as $S_{r,N_1^*=1}$).

While S_r increases and approaches $S_{r,N_1^*=1}$, the seeker's profit is approaching to negative infinity, i.e., $\lim_{S_r \to S_{r,N_1^*=1}} \Pi_s^1(S_r; S_g) = \lim_{S_r \to S_{r,N_1^*=1}} w \cdot S_r + \mu \ln A(1 - \frac{1}{N_1^*(S_r)}) - \mu \ln(h(S_g)) = -\infty$. Therefore, the seeker profit Π_s^1 is eventually decreasing with a high enough S_r, which suggests $S_r^{1*} < \infty$, i.e., the seeker should not always disclose all of his conceptual objectives in Scenario 1 (i.e., even if designers are "required" to incorporate all the disclosed conceptual objectives).

Optimal Number of Conceptual Objectives to Disclose (S_r^{1*}) is Bounded Above by $S_{r,c_1=0}^{1*}$. We first show that the seeker's profit always decreases with the unit cost for designers to frame the design problem (i.e., a higher c_1). $\frac{\partial\Pi_s^1(S_r;S_g)}{\partial c_1} = \frac{\partial\Pi_s^1(S_r;S_g)}{\partial N_1^*}\frac{\partial N_1^*}{\partial c_1} < 0$ (according to Equation (A.7)). Hence, $\Pi_s^1(S_r;S_g|c_1 > 0) < \Pi_s^1(S_r;S_g|c_1 = 0)$, i.e., the seeker's profit is always lower when there is positive cost for designers to frame the design problem. In addition, based on algebra, we have $\frac{\partial\frac{\partial\Pi_s^1(S_r;S_g)}{\partial S_r}}{\partial c_1} = -\mu\ln'(1\text{-}\frac{1}{N_1^*})\frac{\partial\frac{1/N_1^*}{\partial S_r}}{\partial c_1} < 0$, suggesting

$$\frac{\partial\Pi_s^1(S_r;S_g|c_1\geq 0)}{\partial S_r} \leq \frac{\partial\Pi_s^1(S_r;S_g|c_1=0)}{\partial S_r}. \tag{A.10}$$

Recall that, $\Pi_s^1(S_r;S_g|c_1 = 0)$ is concave and maximized at $S_{r,c_1=0}^{1*}$; thus, $\frac{\partial\Pi_s^1(S_r;S_g|c_1=0)}{\partial S_r}|_{S_r>S_{r,c_1=0}^{1*}} < 0$. Combining this with Equation (A.10), we have $\frac{\partial\Pi_s^1(S_r;S_g|c_1\geq 0)}{\partial S_r}|_{S_r>S_{r,c_1=0}^{1*}} \leq \frac{\partial\Pi_s^1(S_r;S_g|c_1=0)}{\partial S_r}|_{S_r>S_{r,c_1=0}^{1*}} < 0$. In other words, when S_r is greater than $S_{r,c_1=0}^{1*}$, the seeker's profit is always decreasing with S_r. Hence, the global maximum of $\Pi_s^1(S_r;S_g)$ is bounded above by $S_{r,c_1=0}^{1*}$. Furthermore, based on simulations in Appendix A.3, we find that S_r^{1*} is not very sensitive to c_1, suggesting that the maximum $S_{r,c_1>0}^{1*}$ is relatively close to $S_{r,c_1=0}^{1*}$.

A.12 Proof of Lemma A.3.2

We denote Π_s in Scenario 2 (defined in Appendix A.3) as Π_s^2. In Scenario 2, designers incorporate the equilibrium subset ($\overline{r}$) of all the disclosed objectives. In this case, the seeker solves the following optimization problem:

$$\max_{S_r\leq\bar{S}_r}\Pi_s^2(S_r;S_g) = \max_{S_r\leq\bar{S}_r}[w\cdot\bar{r}(N_2^*(S_r)) + \mu\ln(m^*(S_g,N_2^*(S_r))\cdot N_2^*(S_r))],$$
$$\text{where } N_2^* \text{ is from Equation (A.6), and } m^* \text{ and } \overline{r} \text{ are from Theorem 2.3.1.} \tag{A.11}$$

Now we show $\frac{\partial\Pi_s^2}{\partial S_r} < 0$, i.e., $\Pi_s^2(S_r;S_g)$ is monotonically decreasing w.r.t S_r. Note that, $\Pi_s^2(S_r;S_g)$ is a function of S_r only through N_2^*; hence, we can write

$$\frac{\partial\Pi_s^2}{\partial S_r} = \frac{\partial\Pi_s^2}{\partial N_2^*}\cdot\frac{\partial N_2^*(S_r)}{\partial S_r}, \tag{A.12}$$

in which we know that $\frac{\partial N_2^*(S_r)}{\partial S_r} \leq 0$ (see the proof in Appendix A.1). Now we show $\frac{\partial\Pi_s^2}{\partial N_2^*} > 0$:

$$\frac{\partial\Pi_s^2}{\partial N_2^*} = w\frac{\partial\bar{r}(N_2^*)}{\partial N_2^*} + \mu\frac{1}{(N_2^*)^2-N_2^*} = \frac{1}{(N_2^*)^2-N_2^*}\cdot\frac{w}{\ln(1/p)}\cdot(\frac{\mu\ln(1/p)}{w} + 2 - N_2^*). \tag{A.13}$$

Based on Equation (A.6), the last term in Equation (A.13) can be re-written as $\frac{\mu\ln(1/p)}{w} + 2 - N_2^* = \frac{1}{X} + 2 - \frac{\sqrt{4Y(X+1)+X^2}-X}{2Y}$, which can be shown to be positive using algebra. With all the terms in Equation (A.13) being positive (by assumption, $N_2^* > 1$), $\frac{\partial\Pi_s^2}{\partial N_2^*} \geq 0$. Combining

$\frac{\partial N_2^*(S_r)}{\partial S_r} \leq 0$ and $\frac{\partial \Pi_s^2}{\partial N_2^*} \geq 0$, we have $\frac{\partial \Pi_s^2}{\partial S_r} \leq 0$ (Equation (A.12)). That is, the seeker profit is monotonically decreasing with respect to the number of disclosed objectives in the problem specification in Scenario 2.

APPENDIX B

Appendices to Chapter 2

B.1 Image Comparison Algorithm – SIFT

SIFT is an algorithm used to detect and describe local features in images proposed by Lowe (1999). Broadly speaking, the algorithm consists of four steps: (1) *Extracting SIFT Feature Vectors:* From a pair of design images (A and B), we extract descriptors of the key points by identifying *SIFT feature vectors* in scale space, which robustly capture the structural properties of the images[1]. (2) *Matching SIFT Feature Vectors:* For each feature vector D_i^A in image A, we calculate its shortest Euclidean distance ($d(D_i^A, D_j^B)$) to each of the SIFT feature vector in image B (D_j^B). Features D_i^A and D_k^B are defined as a matched pair if and only if the ratio $\frac{d(D_i^A, D_k^B)}{d(D_i^A, D_j^B)}$ is less than 2/3 for all j[2]. (3) *Computing the Similarity Ratio:* After obtaining the number of matched feature-vector pairs, we can calculate the percentage of matched SIFT features relative to the total number of SIFT features in images A and B as $\gamma_{A,B} = \frac{N_{A,B}^m}{\min\{N_A, N_B\}}$, where $N_{A,B}^m$ is the number of matched pairs between image A and image B, and N_A (N_B) is the total number of feature vectors extracted from image A (B). (4) *Classifying the Image Pair as Similar or Different:* Finally, we classify the image pair (image A and image B) as either similar or different based on the Similarity Ratio, $\gamma_{A,B}$. The higher the ratio is, the more similar the two images are. If the ratio is 1, the two images are exactly the same. In our empirical analysis, we classify a pair of submissions as similar if $\gamma_{A,B} \geq 0.4$.

Figure B.1 provides an example of the computed Similarity Ratios among six images (sourced from Microsoft Word's clip art). As we can see, the images that are relatively similar to each other have higher pairwise similarity ratios (0.600, 0.524, and 0.424 respectively).

---

[1] The scale-invariant features are efficiently identified by a staged filtering approach. In the first stage, the algorithm identifies key locations in scale space by looking for locations that are maxima or minima of a difference-of-Gaussian function. Then, each point is used to generate a feature vector that describes the local image region sampled relative to its scale-space coordinate frame.

[2] This is the default threshold in Lowe's paper (1999). The approach described here is essentially the nearest-neighbor approach.

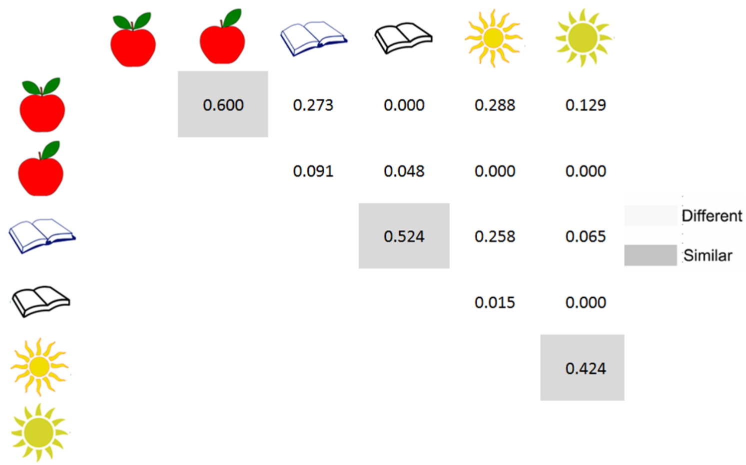


Figure B.1: Similarity Ratio Matrix for Six Designs

B.2 Data Evidence and Reduced-form Results

B.2.1 Descriptive Regressions and Regression Results

We use descriptive regression analysis to explore how disclosed ratings are associated with creators' participation behavior. We classify creators into "new entrants" and "existing creators"/"incumbents", and test separately whether the number of new entrants, and incumbents' follow-up actions, are affected by the ratings that the seeker has disclosed by the end of the previous day (Equations B.1 and B.2, respectively). In Section 3.3.2, we briefly discuss the observations from these descriptive regressions. We now provide details.

In the first regression (Equation B.1), we regress the number of entrants joining contest q on day t ($\Delta(No.\,Creators)_{qt}$) on the numbers of 1-star, 2-star,..., 5-star ratings disclosed up to day $t-1$ ($(No.\text{ 1-}Star)_{qt-1}$, $(No.\text{ 2-}Star)_{qt-1}$,..., $(No.\text{ 5-}Star)_{qt-1}$), while controlling for the number of creators already in the contest ($(No.\,Creators)_{qt-1}$) and the cumulative number of submissions made by all existing participants ($(No.\,Submissions)_{qt-1}$) up to day $t-1$, as well as the time dummies and contest-level fixed effects.

In the second regression (Equation B.2), we apply a multinomial logit regression model to incumbents' follow-up action choices. The dependent variable is a nominal variable denoting incumbent i's choice $Action_{iqt}$ among *redesign, revise, do-both*, or *do-nothing*, where the reference category is *do-nothing*. We include three main sets of independent variables in this multinomial logit regression: (1) the individual-level variables, including the number of submissions that the *focal* creator has made previously ($(No.\,Submissions)_{iqt-1}$), and among all her previous submissions her best rating ($(Best\ Rating)_{iqt-1}$), second-best rating ($(SecondBest\ Rating)_{iqt-1}$), and average rating ($(Avg\ Rating)_{iqt-1}$), (2) the contest-level rating variables, including $(No.\text{ 1-}Star)_{qt-1}$, $(No.\text{ 2-}Star)_{qt-1}$... $(No.\text{ 5-}Star)_{qt-1}$, and (3) control variables, including $(No.\,Submissions)_{qt-1}$ and $(No.\,Creators)_{qt-1}$. Additional controls include the amount of award for the contest ($Award_q$ in \$) and time dummies. To simplify our notation, we group independent variables into three vectors

$$\mathbf{W}_{qt} := \{(No.\,Creators)_{qt}, (No.\,Submissions)_{qt}\};$$
$$\mathbf{Y}_{qt} := \{1, (No.\text{ 1-}Star)_{qt}, (No.\text{ 2-}Star)_{qt}, ..., (No.\text{ 5-}Star)_{qt}\};$$
$$\mathbf{Z}_{iqt} := \{(No.\,Submissions)_{iqt}, (Avg\ Rating)_{iqt}, (Best\ Rating)_{iqt}, (SecondBest\ Rating)_{iqt}\}.$$

$$\Delta(No.\,Creators)_{qt} = \boldsymbol{\beta}\mathbf{Y}_{qt-1} + \boldsymbol{\Psi}\mathbf{W}_{qt-1} + \phi_q + \delta_t + \mu_{qt} \tag{B.1}$$

$$ln\frac{\Pr(Action_{iqt} = k)}{\Pr(Action_{iqt} = \text{do-nothing})} = \boldsymbol{\Gamma}_k\mathbf{Z}_{iqt-1} + \boldsymbol{\Lambda}_k\mathbf{Y}_{qt-1} + \boldsymbol{\alpha}_k\mathbf{W}_{qt-1} + \zeta_k Award_q + \rho_t + \nu_{iqtk},$$
$$\text{where } k = \textit{redesign}, \textit{revision}, \text{ or } \textit{do-both}. \tag{B.2}$$

We also provide the results from these descriptive regressions in Table B.1 and Table B.2 as follows.

Table B.1: Regression of the Number of Entries

	Dependent Variable: $\Delta Creator_{qt}$	
$(No.\,1\text{-}Star)_{qt-1}$	0.026**	(0.008)
$(No.\,2\text{-}Star)_{qt-1}$	0.022*	(0.010)
$(No.\,3\text{-}Star)_{qt-1}$	0.000	(0.008)
$(No.\,4\text{-}Star)_{qt-1}$	−0.021	(0.012)
$(No.\,5\text{-}Star)_{qt-1}$	−0.149***	(0.022)
Control Variables		
$(No.\,Submissions)_{qt-1}$	−0.004	(0.006)
$(No.\,Creators)_{qt-1}$	−0.202***	(0.015)
Contest-Level Fixed Effects	Yes	
Time Dummies	Yes	
Observations	5,607	
R^2	0.166	
Adjusted R^2	0.142	
F Statistic	136.369***	(df = 7)

Note: *p<0.05; **p<0.01; ***p<0.001; the numbers in parenthesis are standard errors

Table B.2: Multinomial Logit Regression of the Incumbent Follow-up Action Choice

			Depend Variable: $Action_{iqt}$			
	re-design		*revision*		*do-both*	
Individual-Level Variables						
$(No.\,Submissions)_{iqt-1}$	0.011	(0.014)	0.080***	(0.006)	0.075***	(0.009)
$AvgRating_{iqt-1}$	0.099	(0.135)	0.101	(0.074)	0.139	(0.120)
$BestRating_{iqt-1}$	0.269***	(0.081)	0.352***	(0.048)	0.188*	(0.076)
$SecondBestRating_{iqt-1}$	−0.095	(0.088)	−0.033	(0.051)	−0.076	(0.081)
Contest-Level Variables						
$Award_q(\$)$	0.001	(0.000)	0.001***	(0.000)	0.001***	(0.000)
$(No.\,1\text{-}Star)_{qt-1}$	0.010***	(0.003)	0.010***	(0.002)	0.011***	(0.003)
$(No.\,2\text{-}Star)_{qt-1}$	0.009**	(0.003)	0.008***	(0.002)	0.012***	(0.003)
$(No.\,3\text{-}Star)_{qt-1}$	0.005	(0.003)	0.008***	(0.002)	0.006*	(0.003)
$(No.\,4\text{-}Star)_{qt-1}$	0.002	(0.004)	0.005*	(0.003)	0.014***	(0.004)
$(No.\,5\text{-}Star)_{qt-1}$	−0.012	(0.009)	−0.027***	(0.005)	−0.030***	(0.008)
$(No.\,Submissions)_{qt-1}$	−0.007*	(0.003)	−0.002	(0.002)	−0.007*	(0.003)
$(No.\,Creators)_{qt-1}$	0.005	(0.006)	−0.018***	(0.004)	−0.009	(0.006)
Time Dummies			Yes			
Observations	24,085		Log Likelihood		-14,783.050	
R^2	0.040		LR Test		1,247.014*** (df = 54)	

Note: *p<0.05; **p<0.01; ***p<0.001; the numbers in parenthesis are standard errors.

B.2.2 Descriptive Patterns in Creator Behavior

Creators' Dynamic Behavior Over the Contest Horizon. We provide the summary statistics of creators' dynamic behavior over the 7-day contest horizon (Table B.3). Note that, these statistics show that creators enter a contest throughout the entire contest period, and also provide evidence that creators do make follow-up submissions throughout the contest horizon.[3]

Table B.3: Summary Statistics Over 7-Day Contests

	Day-1	Day-2	Day-3	Day-4	Day-5	Day-6	Day-7
Number of Entrants	6.469	3.825	3.060	2.961	3.296	3.509	7.162
Cummulative Number of Entrants	6.469	10.294	13.354	16.315	19.612	23.120	30.282
Cummulative Number of Incumbents	-	6.469	10.294	13.354	16.315	19.612	23.120
Number of Submissions	12.636	11.305	9.891	9.513	10.618	11.504	23.043
Number of Submissions (from Entrants)	12.636	7.194	5.476	5.303	5.799	6.645	15.649
Number of Submissions (from Incumbents)	-	4.111	4.416	4.210	4.819	4.859	7.394
Cumulative Number of Submissions	12.636	23.941	33.832	43.344	53.962	65.466	88.509
Number of Submission Per Incumbent	-	0.635	0.429	0.315	0.295	0.248	0.320
Number of Ratings	7.032	8.055	7.741	6.875	7.448	7.167	11.629
Ratio of Rated Submissions	0.557	0.632	0.679	0.709	0.717	0.724	0.672
Cumulative Number of Ratings	7.032	15.087	22.829	29.704	37.152	44.318	55.947
Cumulative Ratio of Rated Submissions	0.557	0.713	0.783	0.723	0.701	0.623	0.505

The Number of Submissions With and Without Receiving Feedback. We compare the number of submissions submitted per day with and without receiving feedback, and find that the average number of designs submitted by a creator on a day (irrespective of whether she has received feedback on her previous design(s) or not) is 0.37, but the submission number increases to 0.709 on the day after she receives feedback on her previous design(s). This result provides evidence that feedback does affect creators' subsequent submission behavior, and creators tend to become more active after receiving feedback.

Entry Score Distribution Over Time. As shown in Figure B.2, the rating distributions (conditional on being rated) of new entrants in the three contest periods is quite similar, providing empirical evidence that creators do not learn from existing design submissions to the contest.

[3]Nonetheless, these summary statistics are averages across all contests in the sample, and cannot capture the dynamic nature of the contests or the interaction among creators. For more thorough understanding about how feedback affects creators' behavior, please refer to our descriptive regression in Section 3.3.2 and Appendix B.2.1.

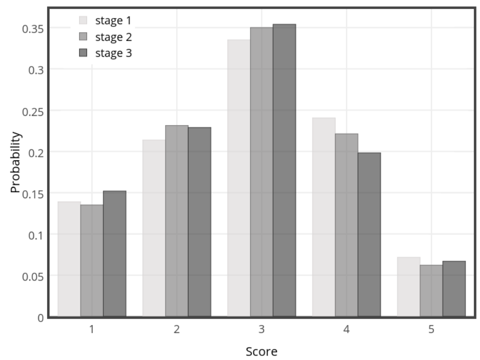


Figure B.2: New Entrants' Rating Distribution (Conditional On Being Rated) Over the Three Periods

B.2.3 Seeker Learning

It is possible that seekers do not know what kind of designs they like the most beforehand, and that there exists seeker learning in crowdsourcing contests for creative products.

There are two possible forms of seeker learning: (i) the seeker discovers new options he likes after seeing more submissions; (ii) the seeker updates his preferences after seeing more submissions (e.g., his evaluation of the same design changes as he sees more submissions).

Between these two forms of seeker learning, Form (i) is already captured by our model. Consider a hypothetical contest. In the first period, there are two fine submissions, and the seeker gives them a 3-star and a 4-star respectively. In the second period, a new creator enters the contest and submits a design; the seeker likes the new submission a lot and gives it a 5-star rating. This process of seeker discovering new designs he likes better is fully captured by our model.

Form (ii) learning – seekers' preferences change as they see more designs – is not captured by our model. For example, a seeker receives a design in the first period; he likes it and gives it a 5-star rating (based on his imperfect prior belief about what he wants). After seeing more designs and learning about his own preferences, the seeker thinks this design is only worth a 2-star rating. If such learning exists in seekers' behavior, we expect to observe cases in which (a) the seeker picks a design that is not the top-rated as the winner of the contest (e.g., a 4-star design submitted later in a contest is preferred to an earlier 5-star submission), or (b) similar designs submitted in different periods receive inconsistent ratings, or (c) the seeker rates a design and then later changes the rating he gave to that design. All of (a), (b) and (c) are very rare in our data. For (a), we find that in the data, the winning design is almost always the highest rated design (92% of contests in our sample). As for (b), we identify all pairs of very similar logo designs (SIFT Similarity Ratio ≥ 0.9) that were submitted in two different periods within the same contest (i.e., image A at t=1, image B at t=2; image A at t=2, image B at t=3; or image A at t=1, image B at t=3), and compare the seeker's ratings for each pair of designs. We find that the ratings of designs in a pair are rarely very different – only in 4.9% of the cases the seeker rates image A lower than (higher than) 3-star, and later rates a very similar design B higher than (lower than) 3-star. This percentage is significantly smaller than the percentage for any randomly generated pair of designs submitted to the same contest (20.1%) (Here, we use 3-star as the cutoff, as it is the middle value of the rating scale.) The difference in ratings between image A and a very similar image B is 0.69, which is again significantly smaller than the average rating difference (1.75) between any random pair of designs submitted to the same contest. As for (c), the platform allows seekers to change ratings for a design, but it happens extremely rarely – in 96% cases, designs are rated only once and the rating is not changed afterwards.

The above data evidence indicates that in our research setting, Form (ii) learning is unlikely to exist in seekers' behavior. That said, it is perhaps interesting to consider what might happen in other settings for which Form (ii) seeker learning does exist (i.e., the seeker can only give out meaningful ratings after seeing enough designs). For such settings, our intuition is that the main finding of our paper (i.e., the late feedback policy performs better than the commonly used full feedback policy) would remain. To the extent that early feedback is meaningless and this is known to the creators, it is a waste of time for the seeker to provide it and hence the late feedback policy becomes more attractive.

B.2.4 Empirical Evidence for the Independence of Creator Action-Specific State Transition Probabilities and the Contest-Level State

Recall that in our model, opponents' states, summarized by the contest-level state s_t, affect the focal creator's state transition probability ($p(x_{i,t+1}|x_{i,t}, s_t)$), through affecting her action choice $a_{i,t}$. Specifically,

$$p(x_{i,t+1}|x_{i,t}, s_t) = \sum_{a_{i,t}} Pr_t(a_{i,t}|x_{i,t}, s_t) \cdot p(x_{i,t+1}|x_{i,t}, a_{i,t}, s_t),$$

where $Pr_t(a_{i,t}|x_{i,t}, s_t)$ is the probability of choosing action $a_{i,t}$ in Period t, given individual-level state $x_{i,t}$ and contest-level state s_t. We assume that $p(x_{i,t+1}|x_{i,t}, a_{i,t}, s_t) = p(x_{i,t+1}|x_{i,t}, a_{i,t})$. That is, conditional on the chosen action, we assume that the contest-level state s_t does not significantly affect creators' state transition. This assumption is supported by the data.

We test whether creator i's choice-specific transition probability $p(x_{i,t+1}|x_{i,t}, a_{i,t})$ is affected by the elements of the contest-level state, s_t (i.e., the numbers of 1-, 2-, ... , 5-Star submissions in the contest). Specifically, for each type of non-null action (i.e., *revision*, *redesign* and *do-both*), we regress incumbent creators' individual state in a period on their previous period state and the elements of the previous period contest-level state. The regression results are reported in Table B.4. As can be seen in the table, none of the elements of the contest-level state significantly affects the choice-specific state transitions associated with each follow-up action. This suggests that given creators' actions, the individual-level state transition probabilities do not depend on the contest-level state s_t. This is not surprising, because the differences in the *action-specific* quality transition probabilities already reflect the different levels of improvement each action could produce, hence, conditional on the chosen action, individuals' quality-transition probabilities should not be significantly affected by opponents' states.

Table B.4: Action-Specific Transition Not Affected by Opponent States

	Dependent Variable: Next Period Individual-Level State: $BestRating_{iqt}$		
	revision	*re-design*	*do-both*
Individual-Level State			
$BestRating_{iqt-1} = 2$	0.666*** (0.135)	0.856*** (0.161)	1.225*** (0.292)
$BestRating_{iqt-1} = 3$	1.440*** (0.125)	1.601*** (0.154)	1.268*** (0.257)
$BestRating_{iqt-1} = 4$	2.221*** (0.124)	2.432*** (0.154)	1.958*** (0.254)
$BestRating_{iqt-1} = 5$	3.035*** (0.132)	3.249*** (0.184)	2.702*** (0.280)
Contest-Level State			
$(No.\,1\text{-}Star)_{qt-1}$	−0.002 (0.007)	−0.010 (0.011)	0.002 (0.019)
$(No.\,2\text{-}Star)_{qt-1}$	−0.010 (0.007)	−0.019 (0.013)	−0.011 (0.019)
$(No.\,3\text{-}Star)_{qt-1}$	−0.006 (0.006)	−0.006 (0.009)	−0.002 (0.016)
$(No.\,4\text{-}Star)_{qt-1}$	0.004 (0.007)	−0.005 (0.013)	0.007 (0.021)
$(No.\,5\text{-}Star)_{qt-1}$	0.019 (0.010)	0.029 (0.016)	0.047 (0.025)
Creator-Level Fixed Effects	Yes	Yes	Yes
Time Dummies	Yes	Yes	Yes
Observations	1,526	889	700
R^2	0.730	0.719	0.583
Adjusted R^2	0.272	0.201	0.118
F Statistic	170.611*** (df = 9; 568)	70.819*** (df = 9; 249)	22.051*** (df = 9; 142)

Note: *p<0.05; **p<0.01; ***p<0.001

B.3 Deriving Equilibrium CCP (Equations (3.7) and (3.8))

In the proposed structural model, a strategy $\sigma^* = \{\rho^*, \lambda^*\}$ representing a MPE equilibrium is characterized by, at any t, for any incumbent i:

$$\rho^*_{i,t} = arg\max_{a_{i,t}}\Big\{U_{i,t}(a_{i,t}, \epsilon_{i,t}) + \beta \sum_{x_{i,t+1}, s_{t+1}} \Big[\int V^{\sigma^*}_{i,t+1}(x_{i,t+1}, s_{t+1}, \epsilon_{i,t+1}) p_\epsilon d(\epsilon_{i,t+1})\Big] p^{\sigma^*}(x_{i,t+1}, s_{t+1}|x_{i,t}, s_t; a_{i,t})\Big\},$$

and for any potential entrant j:

$$\lambda^*_{j,t} = arg\max_{d_{j,t}}\Big\{U^e_{j,t}(d_{j,t}, \varepsilon_{j,t}) + \mathbb{I}(d_{j,t} = Enter)\cdot \beta \sum_{x_{j,t+1}, s_{t+1}} \Big[\int V^{\sigma^*}_{j,t+1}(x_{j,t+1}, s_{t+1}, \epsilon_{j,t+1}) p_\epsilon d(\epsilon_{j,t+1})\Big] p^{e,\sigma^*}(x_{j,t+1}, s_{t+1}|s_t)\Big\},$$

where $p^\sigma(x_{i,t+1}, s_{t+1}|x_{i,t}, s_t; a_{i,t}) = p(x_{i,t+1}|x_{i,t}; a_{i,t}) \cdot p(x_{\text{-}i,t+1}|x_{\text{-}i,t}; \sigma)$ (let $x_{\text{-}i,t} := s_t \setminus x_{i,t}$) and $p^{e,\sigma}(x_{j,t+1}, s_{t+1}|s_t) = p^e(x_{j,t+1}) \cdot p(x_{\text{-}j,t+1}|x_{\text{-}j,t}; \sigma)$, in which $p(x_{\text{-}i,t+1}|x_{\text{-}i,t}; \sigma) = \sum_{a_{\text{-}i,t}, d_t} p(x_{\text{-}i,t+1}|x_{\text{-}i,t}; a_{\text{-}i,t}, d_t) \cdot \Pr(a_{\text{-}i,t}, d_t|\sigma_t; x_{\text{-}i,t})$ (let $a_{\text{-}i,t} = a_t \setminus a_{i,t}$) and $p(x_{\text{-}j,t+1}|x_{\text{-}j,t}; \sigma) = \sum_{a_t, d_{\text{-}j,t}} p(x_{\text{-}j,t+1}|x_{\text{-}j,t}; a_t, d_{\text{-}j,t}) \cdot \Pr(a_t, d_{\text{-}j,t}|\sigma_t; x_{\text{-}j,t})$ (let $d_{\text{-}j,t} = d_t \setminus d_{j,t}$).

Representing MPE in probability space[4] (Milgrom and Weber, 1985), we get equilibrium CCP $\mathbf{P}^*$ as a fixed point $\mathbf{P}^* = \mathbf{\Gamma}(\mathbf{P}^*)$, where the function $\mathbf{\Gamma}$ is the creators' *best response probability function*, with the t^{th} element being $\mathbf{\Gamma}_t(\mathbf{P}) = \{\Gamma_t(\mathbf{P}), \Gamma^e_t(\mathbf{P})\}$ (with and without superscripts "e" denoting potential entrants' entry choices and incumbents' follow-up choices respectively). For incumbents, $\Gamma_t = \{\Gamma_{i,t}(\mathbf{P})\}_{i\in N_t}$, where $\Gamma_{i,t}(\mathbf{P}) = \{\Gamma_{i,t}(a_{i,t}|x_{i,t}, s_t; \mathbf{P} \setminus \mathrm{P}_{i,t})|a_{i,t} \in \mathcal{A}\}$, in which,

$$\Gamma_{i,t}(a_{i,t}|x_{i,t}, s_t; \mathbf{P} \setminus \mathrm{P}_{i,t}) = \int \mathbb{I}(a_{i,t} = arg\max_{a_{i,t}}\{v^{\mathbf{P}}_{i,t}(x_{i,t}, s_t; a_{i,t}) + \epsilon_{i,t}(a_{i,t})\}) p_\epsilon(\epsilon_{i,t}) d\epsilon_{i,t}, \quad \text{(B.3)}$$

where $v^{\mathbf{P}}_{i,t}(x_{i,t}, s_t; a_{i,t}) = -c(a_{i,t}) + \beta \sum_{x_{i,t+1}, s_{t+1}} \tilde{V}^{\mathbf{P}}_{i,t+1}(x_{i,t+1}, s_{t+1}) \cdot p(x_{i,t+1}, s_{t+1}|x_{i,t}, s_t; a_{i,t}, \mathbf{P}_{-i,t})$ is incumbent i's *choice specific value function*. Similarly, for potential entrants, $\Gamma^e_t = \{\Gamma^e_{j,t}(\mathbf{P})\}_{j\in M_t}$, where $\Gamma^e_{j,t}(\mathbf{P}) = \{\Gamma^e_{j,t}(d_{j,t}|s_t; \mathbf{P} \setminus \mathrm{P}_{j,t})|d_{j,t} \in \mathcal{D}\}$, in

[4] Associated with σ_t we can define a set of CCP for incumbents and potential entrants in period t, $P^\sigma_t = \{\{P^\sigma_{i,t}(a_{i,t}|x_{i,t}, s_t)\}, \{P^\sigma_{j,t}(d_{j,t}|s_t)\}|i \in N_t, j \in M_t\}$ such that,

$$P^\sigma_{i,t}(a_{i,t}|x_{i,t}, s_t) = \int I\{\rho_{i,t}(x_{i,t}, s_t, \epsilon_{i,t}) = a_{i,t}\} p_\epsilon(\epsilon_{i,t}) d\epsilon_{i,t}, \text{ and}$$

$$P^\sigma_{j,t}(d_{j,t}|s_t) = \int I\{\lambda_{j,t}(s_t, \varepsilon_{j,t}) = d_{j,t}\} p_\varepsilon(\varepsilon_{j,t}) d\varepsilon_{j,t}.$$

Note that the probabilities $P^\sigma_{i,t}(a_{i,t}|x_{i,t}, s_t)$ and $P^\sigma_{j,t}(d_{j,t}|s_t)$ represent the expected behavior of a creator from the point of view of the rest of the creators when this creator follows strategy σ_t.

which,

$$\Gamma^{e}_{j,t}(d_{i,t}|s_t;\mathbf{P}\setminus \mathrm{P}_{j,t}) = \int \mathbb{I}(d_{j,t} = arg\max_{d_{j,t}}\{v^{e,\mathbf{P}}_{j,t}(s_t;d_{j,t}) + \varepsilon_{jt}(d_{j,t})\})p_{\varepsilon}(\varepsilon_{j,t})d\varepsilon_{j,t}, \tag{B.4}$$

where $v^{e,\mathbf{P}}_{j,t}(s_t;d_{j,t}) = -c^e(d_{j,t}) + \mathbb{I}(d_{j,t} = Enter) \cdot \beta \sum_{x_{j,t+1},s_{t+1}} \tilde{V}^{\mathbf{P}}_{j,t+1}(x_{j,t+1},s_{t+1}) \cdot p(x_{j,t+1},s_{t+1}|s_t;\mathbf{P}_{-j,t},d_{j,t} = Enter)$ is the potential entrants' *choice specific value function.*

Assuming both incumbents' private shock ($\epsilon_{i,t}$) and potential entrants' private shock ($\varepsilon_{j,t}$) follow i.i.d. Type I extreme value distribution (Rust, 1987), we can get a closed-form solution for the equilibrium *CCP* ($\mathbf{P}^*$), i.e., incumbent i follows Equation (3.7), and potential entrant j follows Equation (3.8).

B.4 Conditional Independence

Conditional Independence (CI) is a conventional assumption made in structrual dynamic game models, such as in Jofre-Bonet and Pesendorfer (2003), Srisuma (2013), Igami (2016), Arcidiacono and Ellickson (2011), Egesdal et al. (2015), among others.[5]

Despite CI being an assumption that is conventional and challenging to relax, it could nonetheless be helpful to discuss the motivation for and the implication of the CI assumption in our setting, and show the robustness of our results to the assumption. CI corresponds to two assumptions; quoting from Aguirregabiria and Mira (2010) (the notation they use is different from ours, but the analogue to our notation is straightforward):

> "1. Assumption CI-X (Conditional Independence of Future x): Conditional on the current values of the decision and the observable state variables, next period observable state variables do not depend on current ϵ: i.e., $CDF(x_{i,t+1}|a_{it}, x_{it}, \epsilon_{it}) = F_x(x_{i,t+1}|a_{it}, x_{it})$.
> 2. Assumption IID (i.i.d Unobservables): The unobserved state variables in ϵ_{it} are independently and identically distributed over agents and over time with CDF $G_\epsilon(\epsilon_{it})$ which has finite first moments and is continuous and twice differentiable in ϵ_{it}."

We discuss the implications of these two assumptions in our setting below.

Assumption (1) CI-X:

First, we discuss the economic meaning of the CI-X assumption and provide the rationale for assuming CI-X in our research context. Recall that the last term ($\epsilon_{i,t}(a)$) in the creator utility function is a random shock to her current period utility from choosing an action among *revision*, *redesign*, *do-both*, and *do-nothing*. Those action-specific shocks $\epsilon_{i,t}(a)$ capture unobserved factors that affect creators' preference toward each action in a particular period. For example, if a creator is unusually busy in a period, this will be reflected in a positive shock specific to the *do-nothing* action; if a creator is in the mood to try new design styles, this will be reflected in a positive shock specific to the *redesign* action; if certain external events taking place in a period make a creator feel more like staying in her comfort zone, this will be reflected in a positive shock specific to the *revision* action; etc. With the economic meaning of the action-specific shocks in mind, let us see what the CI-X assumption implies in our setting. In general, CI-X refers to the assumption that $\epsilon_{i,t}(a)$ is independent of $x_{i,t}$ and all state variables in the past periods, and $\epsilon_{i,t}(a)$ is only correlated to x_{t+1} through the choice variables a_{it} (Aguirregabiria and Mira, 2010). In our research setting, although the action-specific shock $\epsilon_{i,t}(a)$ changes its corresponding action's cost and thus affects a creator's

[5]For example, Arcidiacono and Ellickson (2011) argues that "this assumption (CI) is standard in virtually all dynamic discrete choice papers"; and Srisuma (2013) states that "(CI is one of) the key restrictions commonly imposed on the class of games in this literature."

action choice (a_{it}), conditional on a_{it}, $\epsilon_{i,t}(a)$ is not likely to influence the size of the quality improvement. Therefore, we think CI-X is a reasonable assumption in our setting.

Assumption (2) IID:

Next, we perform the following Monte Carlo analysis to show the robustness of our parameter estimates to the presence of serially correlated unobserved state variables.

We follow the *Robustness to serially correlated unobservables* Section in Pakes et al. (2007) closely. The idea here is to show the robustness of our estimation results with respect to serial correlation in the shocks by testing the performance of our estimation technique — which ignores serially correlated shocks — in a setting where we know for sure that there are serially correlated shocks. Specifically, we first compute the equilibrium strategies (a mapping from states and shocks to player actions) of our dynamic game with *an additional* serially correlated state variable, $\gamma_{i,t}$, which shifts follow-up action costs from period to period. Next, we use the equilibrium strategies we just computed to *simulate* data for contests in which creators' actions are determined by the state described by four variables $(x_{it}, s_t, \gamma_{i,t}, \epsilon_{it})$. Finally, we "pretend" that this data is generated from a model with the state described by only three variables $(x_{it}, s_t, \epsilon_{it})$ (without the serially-correlated unobservables) and estimate the utility parameters assuming that the misspecified three-state-variable model is the true model.

In the first two steps of this investigation (solving for the equilibrium and simulating contest data), we incorporate the serial correlation as follows. In each period t, focal incumbent i's action costs (c_a, $\forall a \in \{redesign, revision, do\text{-}both\}$) are all subjected to a serially correlated shock $\gamma_{i,t}$ which takes on three values: a positive shock which increases all costs by 20%, a negative shock which decreases all costs by 20%, and a "zero" shock which leaves all costs unchanged. If in period t, creator i receives a positive or negative shock, then in period $t+1$, she will receive a shock with the same value with .75 probability, and a shock with a value of 0 with .25 probability. If in period t, creator i's shock has a value of 0, then in period $t+1$, she will receive a shock with a value of 0 with .5 probability, and positive or negative shock with .25 probability.

Table B.5 presents the results of this analysis, comparing the true parameters used to generate the data and the estimated parameters recovered from the generated data assuming the misspecified model. The results are encouraging. In particular, in large data sets (the generated data set consists of 1799 contests, 52,439 contest-creator combinations), all of the parameter estimates of the misspecified model are within 3% from the true parameter values. We therefore conclude that there is reason to believe that the size of the bias caused by serially correlated unobservables is quite small in our sample.

The above discussion on the CI-X and IID assumptions implies that our results are robust to the conventional Conditional Independence (CI) assumption.

Table B.5: Sensitivity to Serial Correlation

	True Parameter with known serial correlation	Estimate of the Misspecified Model which ignores serial correlation	Error of the Estimate
α	0.034	0.035	2.94%
$c(redesign)$	2.790	2.803	0.47%
$c(revise)$	2.205	2.201	-0.18%
$c(do\text{-}both)$	3.319	3.292	-0.81%
$c_1^e(enter)$	4.958	4.904	-1.09%
$c_2^e(enter)$	4.326	4.315	-0.25%
$c_3^e(enter)$	3.599	3.599	0.00%

B.5 Alternative Improvement Specification Beyond 5-Star

We use Equations (3.9b) to describe how $x_{i,t}^{true}$ transitions *after the creator gets a 5-star rating*. Although the state transition beyond 5-star occurs rarely, it is an important element that makes our model complete.

Data evidence for quality improvement from actions taken after creators receive a 5-star rating We first provide data evidence for whether and how much the winning probability increases with the number of *revisions/redesigns* after an individual has received her first 5-star rating. We find that after getting a 5-star, the probability of winning is 20.8% for creators who do not take any further follow-up action, 22.0% for those who make additional submissions (i.e., take either the *revision*, *redesign*, or *do-both* action) in one period, and 28.3% for those who make additional submissions in two periods. The differences are statistically significant. This confirms that, after a creator receives her first 5-star rating, the winning probability increases as the number of follow-up actions increases, which provides evidence that everything else equal, creators' best design quality increases with the number of follow-up actions they take.[6]

Robustness checks for alternative improvement specifications beyond 5-star First, note that the assumption implied by Equation (3.9b) is that every time the focal creator takes an non-zero action (i.e., *revision*, *redesign*, or *do-both*), the resulting improvement ($\max(0, \xi_a^{(n)})$) is drawn from the same action-specific distribution, and quality improvements are additive. For example, the improvement from the first *revision* and that from the second *revision* (after getting a 5-star) are drawn from the same distribution (for the *revision* action) and the total quality improvement equals the sum of the improvements resulting from each of these two revisions. (Later, we will show the robustness of our empirical results with respect to this assumption.) This assumption is necessary for the following reason. Estimating the exact functional form of quality improvement requires a sufficient number of observations of a creator receiving a 5-star in the first period, and taking follow-up actions in both the second and third periods. Unfortunately, we only have 46 such observations out of the 965 cases where a creator receives a 5-star(s) in the first period. The linear and additive assumption avoids model overfitting and weak estimation results. Hence, we decide to use the most parsimonious model – the linear additive model – in our main analysis.

However, one might think the marginal quality improvement an additional submission can contribute is decreasing or increasing with the number of follow-up actions the same creator has taken. We now check the robustness of our paper's key finding, the policy simulation

[6] Also note that what we identified here is the relationship between a creator's follow-up actions after she receives her first 5-star rating and her *probability of winning*. The direct relationship between the follow-up actions and *quality improvement* may look different, because (1) the probability of winning is not linear with respect to the quality improvement; (2) in addition to a creator's quality improvement, also others' quality improvements are involved in determining her probability of winning.

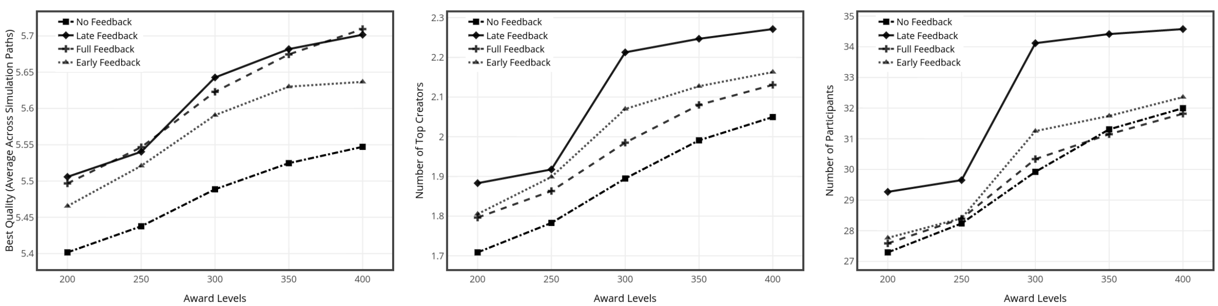
results, with respect to assuming decreasing marginal quality improvement and increasing marginal quality improvement.[7] Specifically, we examine cases in which the second follow-up action is assumed to be 80% and 120% as effective as the first. Mathematically, we adjust Equation (3.9b) on Page 51 as:

$$x_{i,t}^{true} = 5 + \max(0, \xi_0) + \max(0, \xi_a^{(1)}) + \mathbf{80\%} * \max(0, \xi_a^{(2)}), \text{ if } x_{i,t} = 5$$

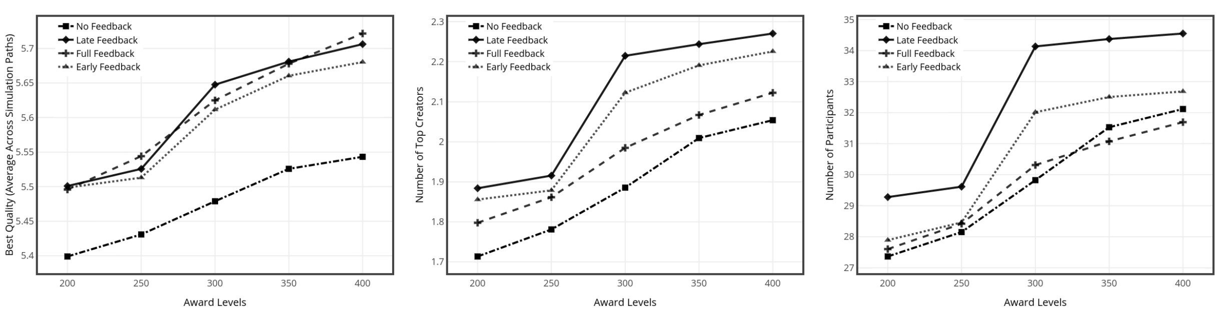
or as:

$$x_{i,t}^{true} = 5 + \max(0, \xi_0) + \max(0, \xi_a^{(1)}) + \mathbf{120\%} * \max(0, \xi_a^{(2)}), \text{ if } x_{i,t} = 5$$

With the new "decreasing/increasing marginal quality improvement" specifications, we re-estimate our model and re-conduct our policy simulations. As shown in Figure B.3, the qualitative nature of the policy simulation results remains the same, and the late feedback policy still outperforms other feedback policies.



(a) The Second Follow-up Action Is 80% Effective As the First One



(b) The Second Follow-up Action Is 120% Effective As the First One

Figure B.3: Different Performance Measures (Left: Maximum Quality, Middle: The Number of Top Performers, Right: The Number of Participants) for Contest Outcomes under Each Policy using Different Model Specification of Equation (3.9b)

[7] As we discussed before, because it is very rare that a creator receives a 5-star in the first period, and take follow-up actions in both the second and third periods, our modeling assumptions made on the unobserved quality improvement process beyond the 5-star rating have very little impact on the estimates of the parameters in creators' utility function.

B.6 Estimation Details

B.6.1 Computing MPE Given Candidate Model Parameters

We solve for the MPE of our finite-horizon dynamic game using backward induction. The following pseudo code details how we solve for a Best Response Mapping (BRM) in period t. Note that the algorithm stops only when the actions of all creators (both incumbents and potential entrants) in the current iteration is the best response to their opponents' actions in the iteration.

Computing MPE for Incumbents and Entrants under a given contest-level state s_t

procedure BEST RESPONSE MAPPING
  initialize iteration counter $n = 0$
  initialize entry CCP $P^0_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t) := \frac{1}{2}\left[\overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t) + P_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t)\right]$
  initialize incumbent CCP $P^0_{i,t}(a_{i,t}|x_{i,t}, s_t) := 1/4$ for $\forall x_{i,t}, \forall a_{i,t} \in \mathcal{A}$
  repeat
    for each incumbent $i \in N_t$ **do**
      best response $P^{'}_{i,t}(a_{i,t}|x_{i,t}, s_t) := \Gamma_{i,t}(a_{i,t}|x_{i,t}, s_t; \mathbf{P}^n_{\text{-}i})$
      update incumbent CCP
        $P^{n+1}_{i,t}(a_{i,t}|x_{i,t}, s_t) := P^n_{i,t}(a_{i,t}|x_{i,t}, s_t) + \alpha_n \cdot \left[P^{'}_{i,t}(a_{i,t}|x_{i,t}, s_t) - P^n_{i,t}(a_{i,t}|x_{i,t}, s_t)\right]$
    end for
    new entrant's best response $P^{'}_{j,t}(d_{j,t}|s_t) := \Gamma^e_{j,t}(d_{j,t}|s_t; \mathbf{P}^n_{\text{-}j})$
    if $P^{'}_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t) \geq P^n_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t)$ **then**
      $P_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t) := P^n_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t)$
    else
      $\overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t) := P^n_{j,t}(d_{j,t} = \mathit{Enter}|s_t)$
    end if
    update entry CCP $P^{n+1}_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t) := \frac{1}{2}\left[\overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t) + P_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t)\right]$
    $n := n + 1$
  until $|P^n_{i,t}(a_{i,t}|x_{i,t}, s_t) - P^{'}_{i,t}(a_{i,t}|x_{i,t}, s_t)| < \epsilon_1; |P^n_{j,t}(d_{j,t}|s_t) - P^{'}_{j,t}(d_{j,t}|s_t)| < \epsilon_2$
end procedure
MPE: $P^*_{j,t}(d_{j,t}|s_t) := P^n_{j,t}(d_{j,t}|s_t)$, $P^*_{i,t}(a_{i,t}|x_{i,t}, s_t) := P^n_{i,t}(a_{i,t}|x_{i,t}, s_t)$, $\forall a_{i,t} \in \mathcal{A}, \forall d_{j,t} \in \mathcal{D}, \forall x_{i,t}$

- α_n is the step size for searching the alternative best response.
- $\mathbf{P}^n_{\text{-}i}$ ($\mathbf{P}^n_{\text{-}j}$) summarizes all incumbent and entrant, except focal incumbent i (entrant j), players' conditional choice probability (CCP) in the n^{th} iteration.
- $\overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t)$ is the highest possible probability for potential entrants to enter a contest. It is calculated as the entry probability when the potential entrant believes the contest will contain no one else but this period's group of new entrants. Mathematically, it is calculated with: $\overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t) = Pr\{-c^e_t(d_{j,t}{=}\mathit{Enter}) + \varepsilon_{j,t}(d_{j,t}{=}\mathit{Enter}) + \beta R/\left[|M| * \overline{P_{j,t}}(d_{j,t}{=}\mathit{Enter}|s_t)\right] > 0\}$.
- $P_{j,t}(d_{j,t}{=}\mathit{Enter}|s_t)$ is the lowest possible probability for potential entrants to enter the contest. It is calculated as the entry probability when a potential entrant be-

lieves there is no chance for them to win. Mathematically, it is calculated with: $P_{j,t}(d_{j,t}{=}Enter|s_t) = Pr\{-c_t^e(d_{j,t}{=}Enter) + \varepsilon_{j,t}(d_{j,t}{=}Enter) > 0\}$.

- Incumbent i's best response to $\mathbf{P}^n_{\text{-}i}$ is given by

$$\Gamma_{i,t}(a_{i,t}|x_{i,t}, s_t; \mathbf{P}^n_{\text{-}i}) = \frac{exp(v_{i,t}^{\mathbf{P}^n_{\text{-}i}}(x_{i,t}, s_t; a_{i,t}))}{\sum_{a'_{i,t}} exp(v_{i,t}^{\mathbf{P}^n_{\text{-}i}}(x_{i,t}, s_t; a'_{i,t}))}, \tag{B.5}$$

where the *choice specific value function* is given by

$$v_{i,t}^{\mathbf{P}^n_{\text{-}i}}(x_{i,t}, s_t; a_{i,t}) = -c(a_{i,t}) + \beta \sum_{x_{i,t+1}, s_{t+1}} \tilde{V}_{i,t+1}^{\mathbf{P}^*}(x_{i,t+1}, s_{t+1}) \cdot p(x_{i,t+1}, s_{t+1}|x_{i,t}, s_t; a_{i,t}, \mathbf{P}^n_{\text{-}i}). \tag{B.6}$$

- Entrant j's best response to $\mathbf{P}^n_{\text{-}j}$ is given by

$$\Gamma^e_{j,t}(d_{j,t}|s_t; \mathbf{P}^n_{\text{-}j}) = \frac{exp(v_{j,t}^{e,\mathbf{P}^n_{\text{-}j}}(s_t; d_{j,t}))}{\sum_{d'_{j,t}} exp(v_{j,t}^{e,\mathbf{P}^n_{\text{-}j}}(s_t; d'_{j,t}))}, \tag{B.7}$$

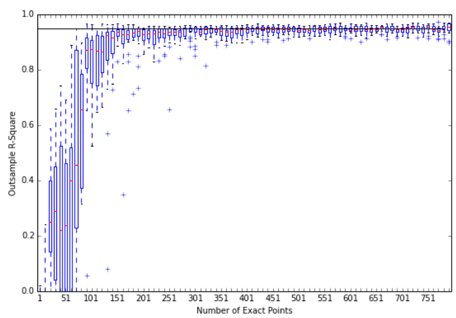
where

$$\begin{aligned} v_{j,t}^{e,\mathbf{P}^n_{\text{-}j}}(s_t; d_{j,t}) = &- c^e(d_{j,t}) + \mathbb{I}(d_{j,t} = Enter) \\ &\cdot \beta \sum_{x_{j,t+1}, s_{t+1}} \tilde{V}_{j,t+1}^{\mathbf{P}^*}(x_{j,t+1}, s_{t+1}) \cdot p(x_{j,t+1}, s_{t+1}|s_t; \mathbf{P}^n_{\text{-}j}, d_{j,t} = Enter). \end{aligned} \tag{B.8}$$

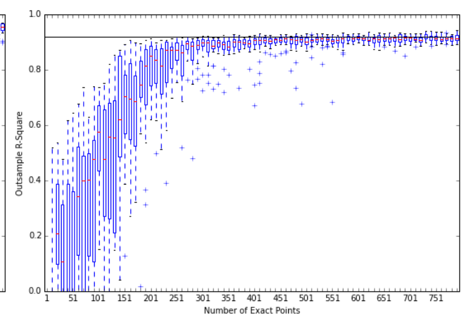
$\tilde{V}_{i,t+1}^{\mathbf{P}^*}(x_{i,t+1}, s_{t+1})$ is the integrated value function under equilibrium play from period $t+1$.

B.6.2 KW Interpolation

When solving for creator i's period-t best response, we need $\tilde{V}_{i,t+1}^{\mathbf{P}^*}(x_{i,t+1}, s_{t+1})$. Solving for period-3 best responses is relatively easy – we can directly use the exact values for $\tilde{V}_{i,4}^{\mathbf{P}^*}(x_{i,4}, s_4)$, as period 4 is the terminal period. However, when solving for her period-2 best response, it is computationally burdensome to compute $\tilde{V}_{i,3}^{\mathbf{P}^*}(x_{i,3}, s_3)$ by backward induction due to the "curse of dimensionality"; we therefore approximate the value using KW interpolation (literature using KW interpolation in solving dynamic game includes: Arcidiacono and Miller (2013); Richards-Shubik (2015); Fowlie et al. (2016); Zhou (2016)). Specifically, we first sample a subset of frequently visited state points $(x_{i,3}, s_3)$; for each of these sampled state points, we compute $\tilde{V}_{i,3}^{\mathbf{P}^*}(x_{i,3}, s_3)$ exactly using backward induction; using these exact values at the sampled state points, we fit a regression model with a third-order polynomial of state variables $(x_{i,3}, s_3)$, and use this fitted model to interpolate the value function at the non-sampled period-3 state points. Similarly, when solving for a period-1 best response, we sample a subset of period-2 state points $(x_{i,2}, s_2)$ and compute the corresponding $\tilde{V}_{i,2}^{\mathbf{P}^*}(x_{i,2}, s_2)$ by backward induction, and interpolate the value function at the non-sampled period-2 state points. The above-mentioned interpolation method provides a good approximation for the value functions. As we can see from Figure B.4, the out-of-sample R-squares can be as high as 95% (92%) for second (third) period's value functions when the number of sampled exact points is sufficiently large. Here, we sample 600 exact points for the interpolation. Note that we get a higher out-of-sample R-square in the second-period interpolation. This is because the number of possible state combinations in the second period is much smaller than that in the third period – with the same number of sampled exact points, we are computing the "exact" value functions for proportionally more points for the second period.



(a) t=2



(b) t=3

Figure B.4: Out-of-sample R-square for Value Function Interpolation

B.7 Identification

B.7.1 Structure of the Identification Proof in Online Appendix B.7.2

Let us recall the key assumptions made in our model:

- Assumption-1: We assume that ϵ and ε follow an extreme value distribution and are *i.i.d.*
- Assumption-2: We make the "outside good" assumption (or normalization assumption), that is, the single-period utility associated with "not-enter" and "do-nothing" action is zero.

For the purpose of identification, the population choice probabilities $P(a|x, s)$ is assumed to be known, which can be empirically estimated from the data. Also, the proof builds on the Hotz and Miller inversion, which provides the correspondence between the first difference of the choice-specific value functions and the first difference of logarithms of conditional choice probabilities. Below, we clarify the steps to identify our model. The equation references below refer to the body of the paper and Online Appendix B.7.2.

1. **Identifying Terminal-Period's Value Function ($\tilde{V}_{i,T+1}(x_{i,T+1}, s_{T+1})$):** In the terminal period (i.e., the reward period, or Period $T + 1$, value function $\tilde{V}_{i,T+1}(x_{i,T+1}, s_{T+1}) = U_{i,T+1}(x_{i,T+1}, s_{T+1}) = \alpha R \cdot \Pr(i \text{ wins}|x_{i,T+1}, s_{T+1})$ (Equations (3.1) and (3.5) in Section 3.4). As soon as α is identified (by exploiting the variation in award level), value function $\tilde{V}_{i,T+1}$ can be directly calculated, and therefore, identified.

2. **Identifying Last-Action-Period's Null-Action-Specific Value Function ($v_{i,T}(x_{i,T}, s_T; a_{i,T}{=}0)$):** The null-action-specific value function $v_{i,T}(x_{i,T}, s_T; a_{i,T}{=}0)$ can be identified for the last action period T (according to Equation (B.16)). Note that the expectation in Equation (B.16) is taken over $(x_{i,t+1}, s_{t+1})$. Given $a_{i,t} = 0$, $x_{i,t+1} = x_{i,t}$; and $P(s_{t+1}|s_t)$ can be calculated based on $P(a|x, s)$ and the action-specific state transition probabilities, both of which have been estimated directly from the data (in the first stage of the estimation process).

3. **Identifying Null-Action-Specific Value Function For All Periods ($v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ for all $t = 1, ..., T$):** From our data, we can empirically estimate the choice probabilities $\mathrm{P}_{i,t+1}(a_{i,t+1}{=}0|x_{i,t+1}, s_{t+1})$. From Step 2, we also know that $v_{i,T}(x_{i,T}, s_T; a_{i,T}{=}0)$ is identified. Then we can use Equation (B.17) to calculate $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ for all periods before the last action period ($t < T$) backwards. The entrant's null-action-specific value function is always zero (Equation (B.15)). That is, $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ are identified for all $t = 1, ..., T$.

4. **Identifying Action-Specific Value Function For All Actions and All Periods ($v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ and $v^e_{j,t}(s_t; d_{j,t} = d)$ for all $k \in \mathcal{A}$, $d \in \mathcal{D}$, and all $t = 1, ..., T$):** With

the identified null-action-specific value functions (i.e., $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$) for all $t = 1, ..., T$, based on Equation (B.10), we can identify action-specific value functions (i.e., $v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ and $v^e_{j,t}(s_t; d_{j,t} = d)$) for all $k \in \mathcal{A}$, $d \in \mathcal{D}$, and all $t = 1, ..., T$, given $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ identified in the previous step and $P_{i,t}(a_{i,t}|x_{i,t}, s_t)$ and $P_{i,t}(d_{i,t}|s_t)$ directly estimated from the data.

5. **Identifying Ex-Ante Value Function For (All Periods $\tilde{V}_{i,t}(x_{i,t}, s_t)$ and $\tilde{V}^e_{j,t}(s_t)$):** Building on Step 4, the ex-ante value functions ($\tilde{V}_{i,t}(x_{i,t}, s_t)$ and $\tilde{V}^e_{j,t}(s_t)$) can be identified by Equations (B.14a)-(B.14b) given that we have identified the choice-specific value functions ($v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ and $v^e_{j,t}(s_t; d_{j,t} = d)$).

6. **Identifying Per-Period Utility Functions ($u(x_{i,t}, s_t; a_{i,t} = k)$ and $u^e(s_t; d_{j,t} = d)$):** Based on the definition of the action-specific value function $v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ (specified in the paragraph after Equations (3.7)-(3.8) in Section 3.4.3), we can write the per-period utility as the difference between the action-specific value function ($v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$) and the discounted next-period ex-ante value function ($\beta\mathbb{E}[\tilde{V}_{i,t+1}(x_{i,t+1}, s_{t+1})|x_{i,t}, s_t; a_{i,t} = k]$) in Equation (B.18). Since both terms on the right hand side of Equation (B.18) are known from Steps 4 and 5, $u(x_{i,t}, s_t; a_{i,t} = k)$ and $u^e(s_t; d_{j,t} = d)$ are identified.

7. **Identifying Cost of Actions:** Since our per-period utility is only a function of creators' own action, but not opponents' actions, we can identify cost of actions.

Here we briefly point out how our paper relates to Bajari et al. (2015), which we closely follow by adapting their proof to a finite-horizon game. We want to highlight the fact that, in general, under Assumptions 1 and 2, $u_i(a_{i,t}, x_{i,t}, s_t)$ (with $a_{-i,t}$ integrated out) can be identified. Cost-shifters come in to play only in identifying $u_i(a_{i,t}, a_{-i,t}, x_{i,t}, s_t)$ ($\pi_i(a_{i,t}, a_{-i,t}, s_t)$ in Bajari et al. (2015)'s notation[8]) from $u_i(a_{i,t}, x_{i,t}, s_t)$ ($\pi_i(a_{i,t}, s_t)$ in Bajari et al. (2015)'s notation) (see page 14 of Bajari et al. (2015) for details). In our case, opponents' actions $a_{-i,t}$ do not affect the focal creator's single period utility u_i, hence $u_i(a_{i,t}, x_{i,t}, s_t)$ is all we need to back out the utility parameters. Therefore, we do not need cost-shifters.

B.7.2 Proof of Identification for Parameters in Creators' Utility Function

In this appendix, we show that it is possible to uniquely recover the deterministic part in the creator per-period utility, which then allows identification of the cost parameters (discussed in Section 3.5.1.3). Note that the choice probabilities, ex-ante value functions and choice-specific value functions all depend on the award amount R, and hence depend on αR; in this proof, we show the identification of creator cost parameters for a given αR value. Note that, the difference in creators behavior among contests with different award amounts in the data, which correspond to different values of single-period utility for the same action-state combination ($u(x_{it}, s_t; a_{it})$), contributes to the identification of α.

[8] In Bajari et al. (2015)'s notation, s_t is not just opponent state, but everyone's states

Per Section 3.4, the equilibrium choice probabilities and the choice-specific value functions are related through:

$$
\begin{aligned}
\mathrm{P}_{i,t}(a_{i,t}|x_{i,t},s_t) &= \frac{exp(v_{i,t}(x_{i,t},s_t;a_{i,t}))}{\sum_{a'_{i,t}} exp(v_{i,t}(x_{i,t},s_t;a'_{i,t}))}, \quad a_{i,t} \in \mathcal{A}(=\{0, redesign, revise, do\text{-}both\}),\\
\mathrm{P}_{j,t}(d_{j,t}|s_t) &= \frac{exp(v^e_{j,t}(s_t;d_{j,t}))}{\sum_{d'_{j,t}} exp(v^e_{j,t}(s_t;d'_{j,t}))}, \quad d_{j,t} \in \mathcal{D}(=\{0, enter\}).
\end{aligned}
\tag{B.9}
$$

For simplicity, we have indexed the actions *do-nothing* (in $\mathcal{A}$) and *not-enter* (in $\mathcal{D}$) as 0 in this identification discussion. The choice-specific value functions ($v_{i,t}(x_{i,t},s_t;a_{i,t})$ and $v^e_{j,t}(s_t;d_{j,t})$ are defined in Section 3.4.3).[9]

From the data, we observe conditional choice probabilities $\mathrm{P}_{i,t}(a_{i,t}|x_{i,t},s_t)$ and $\mathrm{P}_{j,t}(d_{j,t}|s_t)$. We first take the log of both sides of Equation (B.9). Algebra implies that

$$
\begin{aligned}
&\log(\mathrm{P}_{i,t}(a_{i,t}{=}k|x_{i,t},s_t)) - \log(\mathrm{P}_{i,t}(a_{i,t}{=}0|x_{i,t},s_t)) = v_{i,t}(x_{i,t},s_t;a_{i,t}{=}k) - v_{i,t}(x_{i,t},s_t;a_{i,t}{=}0),\\
&\log(\mathrm{P}^e_{j,t}(d_{j,t}=d|s_t)) - \log(\mathrm{P}^e_{j,t}(d_{j,t}{=}0|s_t)) = v^e_{j,t}(s_t;d_{j,t}=d) - v^e_{j,t}(s_t;d_{j,t}{=}0).
\end{aligned}
\tag{B.10}
$$

These equations show that we are able to recover the choice-specific value functions up to a first difference, once we know the empirical population choice probabilities.

Having identified the first differences of the choice-specific value functions, we next turn to the problem of identifying the choice-specific value functions. To do that, we define the ex-ante value functions for incumbents and potential entrants, denoted as $\tilde{V}_{i,t}(x_{i,t},s_t)$ and $\tilde{V}^e_{j,t}(s_t)$ respectively, as the following:

$$
\begin{aligned}
\tilde{V}_{i,t}(x_{i,t},s_t) &= \mathbb{E}V_{i,t}(x_{i,t},s_t,\epsilon_{i,t}) = \int_{\epsilon_{i,t}} V_{i,t}(x_{i,t},s_t,\epsilon_{i,t})p(\epsilon_{i,t})d\epsilon_{i,t}, \text{ and}\\
\tilde{V}^e_{j,t}(s_t) &= \mathbb{E}V^e_{j,t}(s_t,\varepsilon_{j,t}) = \int_{\varepsilon_{j,t}} V^e_{j,t}(s_t,\varepsilon_{j,t})p(\varepsilon_{j,t})d\varepsilon_{j,t}, \text{ where}
\end{aligned}
\tag{B.11}
$$

$V_{i,t}(x_{i,t},s_t,\epsilon_{i,t})$ and $V^e_{j,t}(s_t,\varepsilon_{j,t})$ are defined in Equations (3.5)-(3.6) in Section 3.4.2. We can write the relationship between the ex-ante value functions ($\tilde{V}_{i,t}(x_{i,t},s_t)$ and $\tilde{V}^e_{j,t}(s_t)$) and the

[9] Note that the ex-ante value functions and the choice-specific value functions depend on opponents' strategies, $\sigma_{\text{-}i}$. In the estimation, we assume creators are playing the equilibrium strategies (σ^*_i), and therefore the value functions are those that correspond to the scenario where opponents are playing equilibrium strategies. For simplicity, we suppress σ^*_i in the choice-specific value functions (in Equations (B.9)-(B.10)), and the ex-ante value functions (in Equation (B.11)).

choice-specific value functions ($v_{i,t}(x_{i,t}, s_t; a_{i,t})$ and $v^e_{j,t}(s_t; d_{j,t})$ as:

$$\begin{aligned}\tilde{V}_{i,t}(x_{i,t}, s_t) &= \mathbb{E}_{\epsilon_{i,t}} \max_{a_{i,t}}[v_{i,t}(x_{i,t}, s_t; a_{i,t}) + \epsilon(a_{i,t})], \text{ and} \\ \tilde{V}^e_{j,t}(x_{i,t}) &= \mathbb{E}_{\varepsilon_{j,t}} \max_{d_{j,t}}[v^e_{j,t}(s_t; d_{j,t}) + \varepsilon_(d_{i,t})].\end{aligned} \tag{B.12}$$

Based on Equation (B.12), and properties of the multinomial logit specification (derived, for example, in Train (2009)), we are able to get:

$$\begin{aligned}\tilde{V}_{i,t}(x_{i,t}, s_t) = \mathbb{E}_{\epsilon_{i,t}} \max_{a_{i,t}}[v_{i,t}(x_{i,t}, s_t; a_{i,t}) + \epsilon(a_{i,t})] = \log\left(\sum_{k\in\mathcal{A}} \exp[v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}k)]\right) \\ = \log\left(\sum_{k\in\mathcal{A}} \exp[v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}k) - v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)]\right) + v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0);\end{aligned} \tag{B.13a}$$

similarly, for potential entrants,

$$\tilde{V}^e_{j,t}(s_t) = \log\left(\sum_{d\in\mathcal{D}} \exp[v^e_{j,t}(s_t; d_{j,t}{=}d) - v^e_{j,t}(s_t; d_{i,t}{=}0)]\right) + v^e_{j,t}(s_t; d_{j,t}{=}0). \tag{B.13b}$$

Using Equation (B.10), Equations (B.13a)-(B.13b) can also be written as:

$$\begin{aligned}&\tilde{V}_{i,t}(x_{i,t}, s_t) \\ &\quad = \log\left(\sum_{k\in\mathcal{A}} \exp[\log(\mathrm{P}_{i,t}(a_{i,t}{=}k|x_{i,t}, s_t)) - \log(\mathrm{P}_{i,t}(a_{i,t}{=}0|x_{i,t}, s_t))]\right) + v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0) \\ &\quad = -\log(\mathrm{P}_{i,t}(a_{i,t}{=}0|x_{i,t}, s_t)) + v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0);\end{aligned} \tag{B.14a}$$

and

$$\tilde{V}^e_{j,t}(s_t) = -\log(\mathrm{P}^e_{j,t}(d_{j,t}{=}0|s_t)) + v^e_{j,t}(s_t; d_{j,t}{=}0). \tag{B.14b}$$

Equations (B.14a)-(B.14b) show that the ex-ante value functions ($\tilde{V}_{i,t}(x_{i,t}, s_t)$ and $\tilde{V}^e_{j,t}(s_t)$) are known as soon as the choice-specific value functions ($v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$) are.

Next, we will show how to identify the choice-specific value functions. Based on the

Bellman Equation, we can write $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ recursively as:

$$
\begin{aligned}
v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0) &= u(x_{i,t}, s_t; a_{i,t}{=}0) + \beta\mathbb{E}[\tilde{V}_{i,t+1}(x_{i,t+1}, s_{t+1})|x_{i,t}, s_t; a_{i,t}{=}0], \text{ and} \\
v^e_{j,t}(s_t; d_{j,t}{=}0) &= u^e(s_t; d_{j,t}{=}0) + \mathbb{I}(d_{j,t}{=}Enter) \cdot \beta\mathbb{E}[\tilde{V}_{j,t+1}(x_{j,t+1}, s_{t+1})|s_t]. \\
&= u^e(s_t; d_{j,t}{=}0) + 0.
\end{aligned}
\tag{B.15}
$$

The expectations are taken over current-period private shocks of opponents, as well as future values of the state variables and private shocks of both the focal creator and her opponents.

Recall that we normalize the costs of "do-nothing" action and "not enter" action as zero, hence, $u(x_{i,t}, s_t; a_{i,t}{=}0) = -c(a_{i,t}{=}\textit{do-nothing}) = 0$ and $u^e(s_t; d_{j,t}{=}0) = -c^e_t(d_{j,t}{=}\textit{not-enter}) = 0$, which allows further simplification of Equation (B.15) as:

$$
\begin{aligned}
v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0) &= \beta\mathbb{E}[\tilde{V}_{i,t+1}(x_{i,t+1}, s_{t+1})|x_{i,t}, s_t; a_{i,t}{=}0], \text{ and} \\
v^e_{j,t}(s_t; d_{j,t}{=}0) &= 0.
\end{aligned}
\tag{B.16}
$$

The value functions related to the "*not-enter*" action for potential entrants ($v^e_{j,t}(s_t; d_{j,t}{=}0)$) are identified trivially. Next, we discuss how we achieve identification of the incumbents' value functions related to "*do-nothing*" starting from the last action period (T) in our finite-horizon dynamic game. According to Equations (3.1) and (3.5) in Section 3.4 of the paper, the terminal period value function is $\tilde{V}_{i,T+1}(x_{i,T+1}, s_{T+1}) = U_{i,T+1}(x_{i,T+1}, s_{T+1}) = \alpha R \cdot \Pr(i \text{ wins}|x_{i,T+1}, s_{T+1})$. Therefore, based on Equation (B.16), $v_{i,T}(x_{i,T}, s_T; a_{i,T}{=}0)$ are identified. For all the periods before T ($t < T$), we can further combine Equations (B.14a)-(B.14b), and write Equation (B.16) as:

$$
\begin{aligned}
v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0) &= \beta\mathbb{E}[\tilde{V}_{i,t+1}(x_{i,t+1}, s_{t+1})|x_{i,t}, s_t; a_{i,t}{=}0] \\
&= \beta\mathbb{E}[-\log(\mathrm{P}_{i,t+1}(a_{i,t+1}{=}0|x_{i,t+1}, s_{t+1})) \\
&\quad + v_{i,t+1}(x_{i,t+1}, s_{t+1}; a_{i,t+1}{=}0)|x_{i,t}, s_t; a_{i,t}{=}0] \\
&= \beta\mathbb{E}[-\log(\mathrm{P}_{i,t+1}(a_{i,t+1}{=}0|x_{i,t+1}, s_{t+1}))|x_{i,t}, s_t; a_{i,t}{=}0] \\
&\quad + \beta\mathbb{E}[v_{i,t+1}(x_{i,t+1}, s_{t+1}; a_{i,t+1}{=}0)|x_{i,t}, s_t; a_{i,t}{=}0].
\end{aligned}
\tag{B.17}
$$

From our data, we can empirically estimate the choice probabilities $\mathrm{P}_{i,t+1}(a_{i,t+1}{=}0|x_{i,t+1}, s_{t+1})$. We also know that $v_{i,T}(x_{i,T}, s_T; a_{i,T}{=}0)$ is identified for the last action period. Then we can use Equation (B.17) to calculate $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ for all periods before the last action period ($t < T$) backwards. That is, $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ are identified for all $t = 1, ..., T$.

With these identified $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$, $v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ and $v^e_{j,t}(s_t; d_{j,t} = d)$ are identified for all $k \in \mathcal{A}$ and $d \in \mathcal{D}$ by substituting $v_{i,t}(x_{i,t}, s_t; a_{i,t}{=}0)$ and $v^e_{j,t}(s_t; d_{j,t}{=}0)$ into Equation (B.10). Also note that the ex-ante value functions ($\tilde{V}_{i,t}(x_{i,t}, s_t)$ and $\tilde{V}^e_{j,t}(s_t)$) can be identified by Equations (B.14a)-(B.14b) given that we have identified the choice-specific value functions ($v_{i,t}(x_{i,t}, s_t; a_{i,t} = k)$ and $v^e_{j,t}(s_t; d_{j,t} = d)$). Using the recursive nature of the value function in our dynamic setting, we have:

$$u(x_{i,t}, s_t; a_{i,t} = k) = v_{i,t}(x_{i,t}, s_t; a_{i,t} = k) - \beta\mathbb{E}[\tilde{V}_{i,t+1}(x_{i,t+1}, s_{t+1})|x_{i,t}, s_t; a_{i,t} = k], \text{ and}$$
$$u^e(s_t; d_{j,t} = d) = v^e_{j,t}(s_t; d_{j,t} = d) - \beta\mathbb{E}[\tilde{V}_{j,t+1}(x_{j,t+1}, s_{t+1})|s_t; d_{j,t} = d]. \tag{B.18}$$

Since both terms on the right hand side of Equation (B.18) are known, $u(x_{i,t}, s_t; a_{i,t} = k)$ and $u^e(s_t; d_{j,t} = d)$ are identified. Given $u(x_{i,t}, s_t; a_{i,t} = k)$ and $u^e(s_t; d_{j,t} = d)$, since our per-period utility is only a function of creators' own action, but not opponents' actions, we can identify cost of actions.

As a final note, creators' behavior, more specifically, their conditional choice probabilities, may vary with the award amount (R). If this is true, the identification procedure described above will yield different sets of value functions for different R values. Because the cost parameters and the per-period utility function in periods $1...T$ do not vary with R, any difference in the conditional choice probabilities observed in the data and their implied value functions across contests with different R's, can only be attributed to the difference in αR, which affects the value functions through the terminal-period utility function (Equation (1)). Since R is observed, the magnitude of the variation of creators' behavior across contests with different award amount identifies α.

B.7.3 Monte Carlo Simulation

To demonstrate the effectiveness of our estimation method, we conduct Monte Carlo experiments on a simulated sample. In what follows, we first describe the data-generating process and then present the estimation results on the simulated data.

We simulate 30,000 independent contests with the same horizon. For each empirically observed award level, we first numerically solve for the equilibrium strategies of creators and the corresponding conditional choice probabilities (CCPs) under possible combinations of individual-level state and contest-level state for each period. For each simulated contest, we sample an award-level from the empirical distribution of award-levels, and then use its numerically computed equilibrium CCPs to simulate creator actions for that simulated contest – in particular, we draw actions based on Equations (7)-(8), which depend on the state variables (x_{it}, s_t).[10] In addition, we use the same state transition probabilities estimated from our data to simulate the evolution of the individual- and contest-level states.

[10]Alternatively, we can also sample the unobserved state variables (i.e., ϵ and ε) for each creator in each period of each contest, and then use the equilibrium strategy σ^* to determine the creator actions in each period of each contest. Under the assumption that ϵ and ε are *i.i.d.* and follow the Type I extreme value distribution, this alternative approach is equivalent to the "CCP" based approach described above. The CCP based approach is more convenient to use.

We then use the simulated data to estimate the structural parameters in the creator's utility function (i.e., α, $c(redesign)$, $c(revise)$, $c(do\text{-}both)$, $c^e_1(enter)$, $c^e_2(enter)$, and $c^e_3(enter)$) with the four-step estimation approach described in Section 5.1.2. Table B.6 summarizes the estimation results. These results demonstrate that our approach can successfully back out the model primitives.

Table B.6: Monte Carlo Results

	True Parameter	Recovered Estimates
α	0.0340	0.0307
$c(redesign)$	2.7900	2.8139
$c(revise)$	2.2050	2.1969
$c(do\text{-}both)$	3.3190	3.3296
$c^e_1(enter)$	4.9580	5.0935
$c^e_2(enter)$	4.3260	4.3072
$c^e_3(enter)$	3.5990	3.5788

B.7.4 Identification for Action-Specific State Transition Probabilities Beyond 5-Star

The only difference between cases beyond 5-star (governed by θ_{12}) and those within 5-star (governed by θ_{11}) is that, we do not directly observe the quality state (measured by star ratings), when it goes beyond 5-star. Yet, we can infer the distribution of quality state from the empirical probability of winning ($\Pr(i \text{ win})$) (given the actions creators take after receiving their first 5-star rating).

In both cases, we can observe creators' action and the quality improvement (directly if the quality is below 5-star and indirectly if the quality is greater or equal to 5-star) associated with this action from the data. This information (the connection between an action and the associated action-specific quality improvement) does not depend on parameters in the utility function (e.g., the cost of action). Hence, we can separately estimate the parameters governing the transition probabilities conditional on actions.

A Numerical Example for Cases beyond 5-Star. To provide further detail for interested readers, we construct a simplified numerical example to help illustrate the spirit of how the estimation of θ_{12} works. For simplicity, this example is similar to, but simpler than, our main model. In particular, we consider contests with 2 creators ($i = 1, 2$) and 3 action periods ($T = 3$). For simplicity, we assume that both players join in the very first period; and at the beginning of the second period, both players are rated 5-star. We assume in this example that the players' "true" design qualities are $x_{i,t=2} \sim N(5, 1)$ $(i = 1, 2)$. (Note that this assumption about the distribution of the true design quality is different from the assumption made in our main model; see later in this paragraph for a detailed discussion about this difference and its implications.) In the next two periods, the creators can choose from two actions – make a follow-up submission ($a_{i,t} = 1$) or not ($a_{i,t} = 0$). If a creator i chooses to follow-up, her individual state evolves to reflect the quality improvement of her best design(s) – in particular, we assume $x_{i,t+1} = x_{i,t} + \Delta_i$ and $\Delta_i \sim N(\mu, \sigma)$ when $a_{i,t} = 1$; when $a_{i,t} = 0$, $x_{i,t+1} = x_{i,t}$. In the award period, the creator with the highest individual state $x_{i,T}$ (i.e., best design quality) wins the reward R, and everyone else gets nothing.[11] Note that the transition process in this example model is simplified from our main model in the paper as well. In order to obtain a closed-form representation of the probability of winning (see Equation (B.19)), this example assumes the "true" quality of the first 5-star rating follows a normal distribution (centered around 5), instead of a truncated normal distribution as in our main model; similarly, the quality improvements are assumed to follow a normal distribution instead of a truncated normal distribution. (The exact state transition process beyond 5-star is described by Equations (3.9a)-(3.9b) in the paper.) However, these differences (i.e., with or without "truncation", and two vs. four alternative follow-up actions) do not affect the logic behind the estimation and identification of the parameters governing the state transition process, but only affect the exact formula of quality distributions at the end of the contest and the exact formula of the winning probabilities predicted by the model.

We can now write a creator i's probability of winning ($\Pr(i \text{ wins})$) under different scenarios. (The scenarios are defined based on the two creators' chosen actions $a_{i=1,t=2}$, $a_{i=2,t=2}$, $a_{i=1,t=3}$ and $a_{i=2,t=3}$.) To simplify the notation, we define random variables $z^{\text{0-act}}$, $z^{\text{1-act}}$, and $z^{\text{2-act}}$ to capture the best design quality resulting from all actions a player takes in periods 2 and 3, when the player takes follow-up action in zero, one, and two of these periods, respectively: $z^{\text{0-act}} \sim (5, 1)$, $z^{\text{1-act}} \sim (5 + \mu, \sqrt{1 + \sigma^2})$, and $z^{\text{2-act}} \sim (5 + 2\mu, \sqrt{1 + 2\sigma^2})$.

$$\begin{cases} \Pr^{\text{scn-1}}(i \text{ wins}) = 0.5, & \text{scn-1: } \sum_t a_{i=1,t} = \sum_t a_{i=2,t}; \\ \Pr^{\text{scn-2}}(i \text{ wins}) = \Pr(z^{\text{2-act}} > z^{\text{1-act}}) = \Phi(\frac{\mu}{\sqrt{2+3\sigma^2}}), & \text{scn-2: } \sum_t a_{i=1,t} = 2 \text{ and } \sum_t a_{i=2,t} = 1; \\ \Pr^{\text{scn-3}}(i \text{ wins}) = \Pr(z^{\text{2-act}} > z^{\text{0-act}}) = \Phi(\frac{2\mu}{\sqrt{2+2\sigma^2}}), & \text{scn-3: } \sum_t a_{i=1,t} = 2 \text{ and } \sum_t a_{i=2,t} = 0; \\ \Pr^{\text{scn-4}}(i \text{ wins}) = \Pr(z^{\text{1-act}} > z^{\text{0-act}}) = \Phi(\frac{\mu}{\sqrt{2+\sigma^2}}), & \text{scn-4: } \sum_t a_{i=1,t} = 1 \text{ and } \sum_t a_{i=2,t} = 0; \\ \Pr^{\text{scn-5}}(i \text{ wins}) = 1 - \Pr^{\text{scn-4}}(i \text{ wins}), & \text{scn-5: } \sum_t a_{i=1,t} = 0 \text{ and } \sum_t a_{i=2,t} = 1; \\ \Pr^{\text{scn-6}}(i \text{ wins}) = 1 - \Pr^{\text{scn-3}}(i \text{ wins}), & \text{scn-6: } \sum_t a_{i=1,t} = 0 \text{ and } \sum_t a_{i=2,t} = 2; \\ \Pr^{\text{scn-7}}(i \text{ wins}) = 1 - \Pr^{\text{scn-2}}(i \text{ wins}), & \text{scn-7: } \sum_t a_{i=1,t} = 1 \text{ and } \sum_t a_{i=2,t} = 2; \end{cases} \tag{B.19}$$

In our empirical setting, creators' actions and the corresponding winning probabilities are observed. Therefore, we are able to identify the parameters governing the state transition

[11]Note that, we define the individual state beyond 5-star using continuous variables. Thus, the probability that there is a draw is zero.

probability (i.e., those that describe the distribution of $z^{\cdot}$) by matching the predicted and observed probability of winning for each of these scenarios. Specifically, the equations from scenario-2 through scenario-7 help identify the μ and σ. Note that, we actually have more variations to identify these parameters (governing the improvement distribution) in the real data: we can observe contests with different numbers of creators, and each scenario in different contests would generate a likelihood function that is a non-linear function of the parameters in θ_{12}.

B.8 Estimates for Action-Specific State Transition Probabilities

B.8.1 Frequency Estimates Within 5-star

Frequency estimates for the rating distribution of new entrants' first submission(s) and the action-specific state transition probabilities among states up to 5-star are reported below.

Table B.7: New Entrants Rating Distribution

NA	1-Star	2-Star	3-Star	4-Star	5-Star
0.23	0.109	0.172	0.265	0.172	0.052

Table B.8: Rating Improvement Resulting from "Revise" Action

		Post Rating					
		NA	1-Star	2-Star	3-Star	4-Star	5-Star
Base Rating	NA	0.869	0.035	0.026	0.044	0.026	0
	1-Star	0	0.615	0.187	0.121	0.066	0.011
	2-Star	0	0	0.659	0.239	0.09	0.012
	3-Star	0	0	0	0.785	0.177	0.038
	4-Star	0	0	0	0	0.861	0.139
	5-Star	0	0	0	0	0	1

Table B.9: Rating Improvement Resulting from "Redesign" Action

		Post Rating					
		NA	1-Star	2-Star	3-Star	4-Star	5-Star
Base Rating	NA	0.771	0.032	0.039	0.087	0.055	0.016
	1-Star	0	0.639	0.132	0.118	0.083	0.028
	2-Star	0	0	0.649	0.235	0.095	0.021
	3-Star	0	0	0	0.783	0.158	0.059
	4-Star	0	0	0	0	0.869	0.131
	5-Star	0	0	0	0	0	1

Table B.10: Rating Improvement Resulting from "Do-both" Action

		Post Rating					
		NA	1-Star	2-Star	3-Star	4-Star	5-Star
Base Rating	NA	0.679	0.038	0.094	0.094	0.038	0.057
	1-Star	0	0.551	0.192	0.167	0.077	0.013
	2-Star	0	0	0.523	0.189	0.212	0.076
	3-Star	0	0	0	0.607	0.301	0.092
	4-Star	0	0	0	0	0.783	0.217
	5-Star	0	0	0	0	0	1

B.8.2 State Transition Probability Estimates Beyond 5-Star

The estimation results for θ_{12}, the set of parameters governing the distribution of the unobserved quality improvements beyond 5-star, are displayed in Table B.11. Notice that we fix ξ_0 to follow the standard normal distribution to achieve identification for other elements of θ_{12}.[12] The estimation results suggest that after creators receive at least one 5-star rating, exploratory actions (*redesign*) are less likely to bring positive quality improvements, but the variance of these quality improvements is larger; by contrast, exploitative actions (*revise*) are more likely to bring positive quality improvements, but the variance in these quality improvements is smaller. Combining both exploratory and exploitative actions, *do-both* leads to a medium-level chance of positive quality improvements, but the variance of these quality improvements is very large.

[12]We are not able to simultaneously identify the distributions for $\xi_0, \xi_{redesign}, \xi_{revise}, \xi_{do\text{-}both}$, since shifting all ξ_a distributions horizontally or making them flatter/thinner simultaneously will not affect how we rationalize the winning realizations observed in the data.

Table B.11: Estimates for θ_{12}, Parameters Governing Quality Improvements beyond 5-Star

Mean of ξ_a	Estimate		S.t.d of ξ_a	Estimate	
$\mu(\xi_0)$	Fixed to 0		$\sigma(\xi_0)$	Fixed to 1	
$\mu(\xi_{redesign})$	-0.052	(0.008)	$\sigma(\xi_{redesign})$	1.351	(0.003)
$\mu(\xi_{revise})$	0.230	(0.006)	$\sigma(\xi_{revise})$	0.517	(0.008)
$\mu(\xi_{do\text{-}both})$	0.022	(0.005)	$\sigma(\xi_{do\text{-}both})$	1.426	(0.010)

Note: The numbers in parenthesis are standard errors. Recall in Equation (3.9b), $max(\xi_a, 0)$ represents the quality improvement beyond 5-star resulting from action a.

B.9 Details for Cross-Validation on Contestant Actions

We use 4-fold cross-validation to examine the ability of our model to predict creators' participation behavior. We partition the data randomly into four subsets and perform four rounds of cross-validation. In each round, we compare the prediction generated by the model estimated using three subsets of data (training set) with the fourth subset of observed data (testing set). We rotate the choice of the training and testing sets. After running all four rounds of cross-validation, we average the validation results over the rounds. This validation is done for all actions: *new entry*, *revise*, *redesign*, *do-both*, and *do-nothing*.

We first compare incumbents' actions predicted by the model estimated using the training data set with the observed actions in the testing data set. Let $\mathrm{P}^*_{i,t}(a_{i,t} = a'|x_{i,t}, s_t)$ be the probability of creator i choosing action a', which is calculated by solving the dynamic game based on the model estimated using the training set. Then for any incumbent i in period t in contest q observed in the testing set, having state $(x_{i,t}, s_t)$, the incumbent's predicted action $a^{predict}_{i,t}$ is then drawn from the $\mathrm{P}^*_{i,t}(a_{i,t}|x_{i,t}, s_t)$ distribution. Hence, the predicted number of incumbents taking action a' in period t in contest q is $m^{predict}_{qt}(a') = \sum_{i \in N_{qt}} \mathrm{P}^*_{i,t}(a_{i,t} = a')$ (As a reminder, N_{qt} denotes the set of all incumbents in contest q in period t; also note that, for readability, we are suppressing the state variables here and in the rest of this Appendix B.9). Let $a^{observe}_{i,t}$ denote the observed action by incumbent i in period t, and $m^{observe}_{qt}(a') = \sum_{i \in N_{qt}} \mathbb{I}(a^{observe}_{i,t} = a')$ is the actual number of incumbents taking action a' in period t in contest q. Let us use N $(N = \sum_q \sum_t ||N_{qt}||)$ to denote the sum of the numbers of incumbents over all periods and all contests observed in the testing set. (Note that N is different from the total number of participants who have ever submitted designs to a contest. For example, a creator who joins the contest at $t = 1$ will be counted twice in N, as she is an incumbent in both period $t = 2$ and period $t = 3$.) Following Akşin et al. (2013), we consider the estimated model's relative and absolute errors in predicting incumbent actions in the hold-out sample (testing set) as the performance metrics for the cross-validation. The relative and absolute errors in predicting action a' are defined as:

$$\begin{aligned}\text{relative error}_{a'} =& \frac{|\sum_q \sum_t m_{qt}^{predict}(a') - \sum_q \sum_t m_{qt}^{observe}(a')|}{\sum_q \sum_t m_{qt}^{observe}(a')},\ a' \in \mathcal{A},\\ \text{absolute error}_{a'} =& 1/N \cdot |\sum_q \sum_t m_{qt}^{predict}(a') - \sum_q \sum_t m_{qt}^{observe}(a')|,\ a' \in \mathcal{A},\end{aligned}$$

where $\mathcal{A} = \{revise, redesign, do\text{-}both, do\text{-}nothing\}$. For potential entrants' actions, we compare the number of entries predicted by the model estimated using the training set with the observed number of entries in the testing set. For potential entrant j in period t, let $\mathrm{P}_{j,t}^*(d_{j,t} = Enter|s_t)$ denote the probability of j entering the contest predicted by the model estimated using the training set, and $d_{j,t}^{observe}$ denote the observed entry decision by j. As for incumbents, we denote the predicted number of entries in period t in contest q as $m_{qt}^{e,predict}(Enter) = \sum_{j \in M_{qt}} \mathrm{P}_{j,t}^*(d_{j,t} = Enter)$, and the actual number of entries in period t in contest q is $m_{qt}^{e,observe}(Enter) = \sum_{j \in M_{qt}} \mathbb{I}(d_{j,t}^{observe} = Enter)$ (as a reminder, M_{qt} denotes the set of all potential entrants in contest q in period t). Let M ($M = \sum_q \sum_t ||M_{qt}||$) denote the total number of potential entrants; we have

$$\begin{aligned}\text{relative error}_{Entry} &= \frac{|\sum_q \sum_t m_{qt}^{e,predict}(Enter) - \sum_q \sum_t m_{qt}^{e,observe}(Enter)|}{\sum_q \sum_t m_{qt}^{e,observe}(Enter)},\\ \text{absolute error}_{Entry} &= 1/M \cdot |\sum_q \sum_t m_{qt}^{e,predict}(Enter) - \sum_q \sum_t m_{qt}^{e,observe}(Enter)|.\end{aligned}$$

B.10 Overview of the Robustness Checks

We designed our main model to be parsimonious but effective. To ensure that our empirical results are robust to our modeling choices and assumptions, we conduct a series of robustness checks.

B.10.1 Myopic Creators

We consider two alternative models of creator behavior: i) fully myopic and ii) partially myopic. In the "fully myopic" model, we assume that creators make decisions based only on the current-period expected utility. In the "partially myopic" model, we assume every creator makes her decision, believing that she is the last one in the contest to act. Using the two alternative models, we estimate the parameters of interest and compute the simulated log likelihood of observing the actual choices in the data. Both alternative models yield a considerably poorer fit in comparison with the strategic model, suggesting no evidence for the myopic models being superior (details are reported in Online Appendix B.11.1).

B.10.2 Heterogeneity in Creator Ability/Experience and Interest

In the main model, we assume creators are ex-ante homogeneous; that is, the heterogeneity among creators is captured by their realized ratings, and before the ratings of their submissions are disclosed, they are homogeneous. To ensure that our results are not sensitive to this homogeneity assumption, we perform the following robustness check with respect to two most likely types of creator heterogeneity — heterogeneity in (1) creator ability/experience and (2) creator interest in the contest.

We use creators' *Reputation Level* (on a scale of $0-100$) the platform assigns to creators to measure their ability/experience level.[13] If a creator's *Reputation Level* is above or equal to 70, she is classified as high-reputation (H); otherwise, she is classified as low-reputation (L).[14] First, we conduct regression analyses and show there is little evidence for heterogeneous entry or follow-up action behavior between the high-reputation and the low-reputation creators, indicating that creators' participation behavior is not significantly affected by their ability/experience. Second, we re-estimate our structural model using stratified samples – *High-Reputation Concentrated Contests*, *Low-Reputation Concentrated Contests*, and *Balanced Contests*, based on the percentage of high-reputation creators in the contest. We

[13] *Reputation Level* is computed by the platform and is displayed on every creator's profile page, and it summarizes the ratings of the creator's past submissions, her level of participation, her history on the platform, and her community behavior (e.g., frequencies of visiting the site, reporting problems, and participating in the forums).

[14] We use 70 as the cutoff score, because 70 is the starting-point score assigned to a new creator who just joined the platform, from where the system adjusts the creator's score upwards or downwards according to her performance and activity level.

compare the estimation results and are able to conclude that creator ability/experience heterogeneity does not meaningfully affect our estimation results. (The complete description of the regression procedure/results and stratified sample test can be found in Online Appendices B.11.2.1-B.11.2.3.) We find these results intuitive. First, the logo design contest we are studying is likely to be a so-called "ideation project" (Terwiesch and Xu, 2008), where the impact of participants' endowed expertise is attenuated by the fact that the notion of quality is highly subjective – it is based on seekers' private tastes rather than objective quality measures, as indicated by a popular platform: "These ratings are subjective. A star rating doesn't reflect your design skill – it indicates the personal preferences of the contest holder." Hence, a creator who performs well in other design contests, or has participated in a large number of contests, does not necessarily have an advantage in a new contest she chooses to participate in. Second, even though the creator population on the platform is highly diverse in their experience level and background, those who frequently participate in contests and thus contribute more to the model estimation are relatively homogeneous.

One may be concerned that the heterogeneity in creator interest in a contest may affect creators' state and their actions simultaneously. To test if creators' heterogeneous "interest" in a contest impacts their follow-up decisions, we perform a reduced-form empirical analysis. In this analysis, a creator i's interest in contest q is measured by the percentage of contests creator i participated in (throughout the sample period) that fall in the same industry as contest q. With this measure, we re-run the multinomial logit regression model of incumbents' follow-up action choices, and find little evidence for heterogeneity in creators' interest that affects their follow-up decisions. (The complete description of the analysis and results can be found in Online Appendix B.11.2.4.)

B.10.3 Strategic Waiting

We consider another alternative model that allows for the possibility that creators might strategically wait to participate in a contest. In this alternative model, the number of potential entrants in a particular period is endogenously determined as the number of active creators on the platform minus the number of creators who have already joined the focal contest; and these potential entrants can strategically wait — they will decide between entering the contest in the current period and waiting for another period. The estimation results of this alternative model show that the qualitative nature of our empirical findings remains the same under this alternative assumption on the entry process. We also provide reduced-form empirical evidence showing our main model (as opposed to the alternative model where the creator can strategically wait) is more consist with the data. (The complete description of the alternative model and the corresponding estimation results, as well as the reduced-form analysis can be found in Online Appendix B.11.3.)

B.10.4 Including Non-Monetary Incentives

Another assumption we make in our main model is that the positive utility a creator receives in the terminal period only comes from the financial reward given by the seeker (see Equation 3.1). One might argue that creators can also get non-monetary rewards (e.g., learning by participating, pure joy of designing, and building up design portfolios) from participating in these design contests. To see whether our main estimation results are robust to the inclusion of non-monetary incentives, we revise the utility function for the rewarding period $(T+1)$ as follows.

$$U_{i,T+1}(x_{i,T+1}, s_{T+1}) = \alpha R \cdot \Pr(i \text{ wins}|x_{i,T+1}, s_{T+1}) + R_{nm}, \tag{B.20}$$

where R_{nm} is the additional non-monetary reward creator i receives in the terminal period. Note that the financial reward R is only received when a creator wins the contest, while any creator receives the non-monetary reward R_{nm} as long as she participates. The estimation results of the revised model suggestion that the inclusion of the non-monetary incentives has little effect on the estimates of the utility parameters. (See Table B.19 in Online Appendix.)

B.10.5 Robustness with Respect to the Discount Factor β, the Number of Potential Entrants $||M_t||$, and the SIFT Cutoff

We further test the sensitivity of the estimation results with respect to the discount factor β, the number of potential entrants $||M_t||$, and the SIFT cutoff. Not surprisingly, most of the cost estimates increase with β, because a higher β increases the expected utility from future periods, and hence the model needs larger cost estimates to rationalize the observed patterns of *entry*, *redesign*, *revision*, and *do-both*. Additionally, entry costs for all periods increase with the assumed number of potential entrants, as models that assume more potential entrants need higher entry costs to rationalize the number of entrants observed in the data. Lastly, the cost of *revision* increases and the cost of *redesign* decreases as we tune up the SIFT cutoff; since higher SIFT cutoff categorizes more actions into *redesign*, the model needs a lower *redesign* cost estimate and a larger *revision* cost estimate to rationalize the observed patterns in creators' follow-up actions. Overall, the qualitative nature of the results are the same under different assumptions for the discount factor, the number of potential entrants, and the SIFT cutoff. (See Table B.23 in Online Appendix.)

B.11 Detailed Discussion and Results for Robustness Checks

To ensure that our empirical results are robust to our modeling choices and assumptions, we conduct a series of robustness checks. The findings from these checks are summarized in Online Appendix B.10. Below we provide the detailed results for each of the robustness checks.

B.11.1 Performance of the Strategic Model Compared with Myopic Models

We consider two alternative models of creator behavior: i) fully myopic and ii) partially myopic. In the "fully myopic" model, we assume that creators make decisions only based on the current-period expected utility. In the "partially myopic" model, we assume every creator makes her decision, believing that she is the last one in the contest to act. Using the two alternative models, we estimate the parameters of interest and compute the simulated log likelihood of observing the actual choices in the data. Table B.12 reports the fit of each model. As can be seen in Table B.12, both alternative models (fully myopic and partially myopic) yield a considerably poorer fit in comparison with the strategic model, suggesting no evidence for the myopic models being superior. Specifically, The Akaike information criterion (AIC) is 29512.08 for the fully myopic model, and 28966.97 for the strategic model; the difference of two AICs ($29512.08 - 28966.97 = 545.11$) represents the information loss experienced if we use the myopic model rather than the strategic model. This loss is of considerable size – according to rough guidelines (Raftery, 1996), greater than 10 difference in AIC indicates the myopic model has essentially no support.

Table B.12: Alternative Models

	Simulated Log Likelihood	AIC
Strategic Model	-14476.49	28966.97
Fully Myopic Model	-14749.04	29512.09
Partially Myopic Model	-14625.05	29264.10

B.11.2 Robustness Checks With Respect to Heterogeneity

We conduct several tests to ensure the robustness of our results with respect to the ex-ante homogeneity assumption, namely, the high- and low-reputation creators do not behave differently in the same contest.

B.11.2.1 Reduced-form Test for Heterogeneity

pointer:original:reg

We first regress the percentage of the high-reputation creators among all creators joining contests on day t ($\%\Delta H_{qt} = \frac{\Delta(No.\,High\,Type\,Creators)_{qt}+1}{\Delta(No.\,Creators)_{qt}+1}$[15]) on $\mathbf{W}_{qt}$ and $\mathbf{Y}_{qt}$ (defined in Online Appendix B.2.1), to test whether the high-reputation creators are more/less likely to join contests that are more/less competitive. The estimation results of this regression suggest that neither the number of existing submissions with high ratings, nor the number of existing submissions with low ratings has a significant effect on $\%\Delta H_{qt}$, and that the model is not significant with $F\text{-}Stats = 1.244$ ($df = 7; n = 4793$).[16] In other words, there are not disproportionately high- or low-reputation creators joining a contest when more high or low ratings are disclosed. Therefore, there is little evidence for any heterogeneous entry behavior between the high-reputation and the low-reputation creators.

Next, we test whether the high-reputation and low-reputation creators differ in their decisions on follow-up actions by estimating a variant of Equation (B.2), in which two additional independent variables – the focal creator's type dummy ($I(H)_i$)[17] and the percentage of high-reputation creators among all existing creators in contest q on day $t-1$ ($H_{qt-1} = \frac{(No.\,High\,Type\,Creators)_{qt}+1}{(No.\,Creators)_{qt}+1}$). The results of this multinomial regression suggest that $I(H)_i$ is not significantly correlated with the probabilities of *revision* and *do-both*, and is only marginally significantly correlated with *redesign*; $\%H_{qt-1}$ is not significantly correlated with any of the follow-up actions. This indicates that neither the focal incumbent creator i's type nor the percentage of high-reputation creators in the contest significantly affects creator i's choice of follow-up actions, after controlling for the individual-level and contest-level state variables. This finding, along with the results for the previous regression, supports our argument that creators' participation behavior is not significantly affected by either their own ability/experience, or that of their rivals. The complete regression results can be found in Tables B.13 and B.14.

[15] We added 1 to both the numerator and the denominator to avoid un-defined numbers.

[16] Among all independent variables, the only significant one is $(No.\,Submissions)_{qt-1}$: it is only marginally significant ($p\text{-}value = 0.028$) with a small magnitude (0.001).

[17] $I(H)_i = 1$, if $ReputationLevel_i \geq 70$; otherwise $I(H)_i = 0$.

Table B.13: Regression of The Percentage of High Type Among New Entrants

	Dependent variable: $\%\Delta H_{qt}$	
$(No.\,1\text{-}Star)_{qt-1}$	-0.001	(0.001)
$(No.\,2\text{-}Star)_{qt-1}$	-0.0003	(0.001)
$(No.\,3\text{-}Star)_{qt-1}$	0.0002	(0.001)
$(No.\,4\text{-}Star)_{qt-1}$	-0.001	(0.001)
$(No.\,5\text{-}Star)_{qt-1}$	0.001	(0.002)
$(No.\,Submissions)_{qt-1}$	0.001*	(0.0005)
$(No.\,Creators)_{qt-1}$	-0.002	(0.001)
Time Dummies	Yes	
Contest-level Fixed Effect	Yes	
Observations	5,607	R^2: 0.002
Adjusted R^2	0.002	F Statistic: 1.244 (df = 7; 4793)

Note: *p<0.05; **p<0.01; ***p<0.001; the numbers in parenthesis are standard errors.

Table B.14: Multinomial Logit Regression of Incumbent Follow-up Actions

	Depend Variable: $Action_{iqt}$					
	Re-design		*revision*		*do-both*	
Heterogeneity Variables						
$I(H)_i$	-0.162*	(0.076)	-0.043	(0.048)	-0.008	(0.075)
$\%H_{qt-1}$	0.322	(0.307)	0.045	(0.188)	-0.092	(0.291)
Individual-Level Variables						
$(No.\,Submissions)_{iqt-1}$	0.011	(0.014)	0.080***	(0.006)	0.075***	(0.009)
$AvgRating_{iqt-1}$	0.121	(0.135)	0.108	(0.075)	0.140	(0.120)
$BestRating_{iqt-1}$	0.271***	(0.080)	0.352***	(0.048)	0.189*	(0.076)
$SecondBestRating_{iqt-1}$	-0.101	(0.088)	-0.034	(0.051)	-0.077	(0.081)
Contest-Level Variables						
$Award_q(\$)$	0.001	(0.0004)	0.001***	(0.0002)	0.001***	(0.0003)
$(No.\,1\text{-}Star)_{qt-1}$	0.010***	(0.003)	0.010***	(0.002)	0.011***	(0.003)
$(No.\,2\text{-}Star)_{qt-1}$	0.009**	(0.003)	0.008***	(0.002)	0.011***	(0.003)
$(No.\,3\text{-}Star)_{qt-1}$	0.005	(0.003)	0.008***	(0.002)	0.006*	(0.003)
$(No.\,4\text{-}Star)_{qt-1}$	0.002	(0.004)	0.005*	(0.003)	0.014***	(0.004)
$(No.\,5\text{-}Star)_{qt-1}$	-0.012	(0.009)	-0.027***	(0.005)	-0.030***	(0.008)
$(No.\,Submissions)_{qt-1}$	-0.007*	(0.003)	-0.002	(0.002)	-0.007*	(0.003)
$(No.\,Creators)_{qt-1}$	0.005	(0.006)	-0.018***	(0.004)	-0.010	(0.006)
Time Dummies			Yes			
Observations	24,085		R^2		0.041	
Log Likelihood	-14,771.670		LR Test		1,250.716*** (df = 60)	

Note: *p<0.05; **p<0.01; ***p<0.001; the numbers in parenthesis are standard errors.

B.11.2.2 Alternative Reduced-form Test for Heterogeneity (with Two Separate Regressions for High/Low Reputation Creators)

pointer:new:reg

We separately analyze the potential entrants' entry decision and the incumbents' follow-up action choice, and study whether high-reputation and low-reputation creators behave differently.

We start with the *potential entrants*' entry decision. We regress the number of high- and low-reputation entrants separately on the state variables. Note that we cannot directly regress the number of high- and low-reputation entrants separately on the state variables and compare the estimated coefficients in these two regressions, because the magnitude of the coefficients are significantly affected by the number of active high- and low-reputation creators on the platform. For example, if the majority of active creators belong to the high type, then the estimated coefficients in the regression for the high-reputation creators would be larger in magnitude, even if the probability of a high- and low-reputation creator joining a contest is identical. Hence, we divide the number of high-reputation (low-reputation) entrants by the total number of times high-reputation (low-reputation) creators appear in contests held on the platform during the study period (i.e., the denominator for high-reputation is $\sum_{q \in \forall \text{Contest}} \sum_{j \in M_q} \mathbb{I}(j \text{ is a high-reputation})$, and the denominator for low-reputation is $\sum_{q \in \forall \text{Contest}} \sum_{j \in M_q} \mathbb{I}(j \text{ is a low-reputation})$), and use the "adjusted" entry number as the dependent variable in these regressions. Table B.15 compares the results for the low-reputation creators and those for high-reputation creators. As can be seen in the table, the significance and signs of the two sets of coefficients are mostly the same. To formally test whether the two sets of coefficients are significantly different, we perform a Z-test (Clogg et al., 1995; Paternoster et al., 1998) for the coefficients of each independent variable in the two regressions (e.g., for $(No.\,1\text{-}Star)_{qt\text{-}1}$, we test with $H_0 : \beta^{\text{High}}_{(No.\,1\text{-}Star)_{qt\text{-}1}} = \beta^{\text{Low}}_{(No.\,1\text{-}Star)_{qt\text{-}1}}$) and report the results in Table B.16. Except for $(No.\,3\text{-}Star)_{qt\text{-}1}$, we cannot reject the null hypothesis for the coefficient of any other independent variable, indicating that the two sets of coefficients are not significantly different. The significant difference in the coefficients of $(No.\,3\text{-}Star)_{qt\text{-}1}$ between the two regressions, is not very concerning, because $(No.\,3\text{-}Star)_{qt\text{-}1}$ is individually insignificant in both regressions (Table B.15).

Next, we regress the high- and low-reputation *incumbents*' follow-up action decisions on the state variables. Table B.17 compares the results for the high- and low-reputation incumbents (for example, "*redesign*:$(No.\,Submissions)_{iqt-1}$" stands for the effect of $(No.\,Submissions)_{iqt-1}$) on creator i's probability of choosing *redesign*). As can be seen in the table, between the two sets of coefficients, the signs and sizes are mostly the same, but the significance level varies. To formally test whether the two sets of coefficients are significantly different, again we perform a Z-test for the coefficients of each independent variable in the two regressions and report the results in Table B.18. Except for *revision*:$(No.\,Submissions)_{iqt-1}$, we do not find the coefficients of the other 38 independent variables are significantly different between the two regressions.

Moreover, combining this set of results with those presented in Table B.14, we find no systematic pattern indicating the creator ex-ante heterogeneity affects their participation behavior (e.g., systematic pattern indicating heterogeneity could be the two sets of tests consistently suggest that the vast majority of independent variable(s) have similar effects on different types of creators' behavior): in the earlier test presented in Table B.14, the only significant variable is *re-design*:$I(H)_i$ ($I(H)_i$ in the regression for the *redesign* action); whereas in the test presented in Table B.17, the only variable that has significantly different coefficients between the low- and high-reputation regressions is *revision*:$(No.\ Submissions)_{iqt-1}$ ($(No.\ Submissions)_{iqt-1}$ in the regression for the *revision* action). Therefore, the marginal significance of the coefficient of 1 (out of 39) independent variables in Table B.18 is likely due to randomness. It would be more concerning if the results from these two sets of regressions display systematic patterns.

However, we are aware that this set of results cannot completely rule out the possible effect of ex-ante heterogeneity on creator behavior, especially for the incumbents. That is why we conducted the following additional stratified sub-sample test, discussed next.

Table B.15: Separate Regressions of the Number of Entries for High- and Low-Type Creators (Adjusted by The Overall Entries by H- and L- Type Creators on The Platform)

	Dependent variable :	
	ΔL_{qt}(adjusted)	ΔH_{qt}(adjusted)
$(No.\ 1\text{-}Star)_{qt-1}$	1.971* (0.931)	1.131** (0.424)
$(No.\ 2\text{-}Star)_{qt-1}$	1.309 (1.129)	1.084* (0.514)
$(No.\ 3\text{-}Star)_{qt-1}$	−1.776 (0.930)	0.672 (0.424)
$(No.\ 4\text{-}Star)_{qt-1}$	−2.705* (1.355)	−0.525 (0.617)
$(No.\ 5\text{-}Star)_{qt-1}$	−7.345** (2.492)	−8.113*** (1.135)
$(No.\ Submissions)_{qt-1}$	0.048 (0.693)	−0.300 (0.316)
$(No.\ Creators)_{qt-1}$	−8.556*** (1.670)	−11.528*** (0.760)
Contest-Level Fixed Effects	Yes	Yes
Time Dummies	Yes	Yes
Observations	5,607	5,607
R^2	0.036	0.185
Adjusted R^2	−0.127	0.047
F Statistic (df = 7; 4793)	25.869***	155.623***
Note:		*p<0.05; **p<0.01; ***p<0.001

Table B.16: Difference of The Coefficients (Separate Regressions for Entries by High- and Low-Type Creators)

	$coef_H$	$coef_L$	se_H	se_L	$\|coef_H\text{-}coef_L\|$	t-value	p-value
$(No.\ 1\text{-}Star)_{qt-1}$	1.971	1.131	0.931	0.424	0.840	0.821	0.412
$(No.\ 2\text{-}Star)_{qt-1}$	1.309	1.084	1.129	0.514	0.225	0.181	0.856
$(No.\ 3\text{-}Star)_{qt-1}$	-1.776	0.672	0.930	0.424	2.448 *	-2.395	0.017
$(No.\ 4\text{-}Star)_{qt-1}$	-2.705	-0.525	1.355	0.617	2.180	-1.464	0.143
$(No.\ 5\text{-}Star)_{qt-1}$	-7.345	-8.113	2.492	1.135	0.768	0.280	0.779
$(No.\ Submissions)_{qt-1}$	0.048	-0.300	0.693	0.316	0.348	0.457	0.648
$(No.\ Creators)_{qt-1}$	-8.556	-11.528	1.670	0.760	2.972	1.620	0.105

Table B.17: Regressions of Incumbent Actions on Feedback by Creator Type

	Dependent variable:			
	Incumbents Follow-up Action Choice: $Action_{iqt}$			
	(High Reputation)		(Low Reputation)	
redesign:*Intercept*	-4.5877***	0.2439	-4.0970***	0.2892
revision:*Intercept*	-3.9091***	0.1519	-4.3582***	0.2003
do-both:*Intercept*	-4.1477***	0.2304	-4.5415***	0.285
redesign:$(No.\ Submissions)_{iqt-1}$	0.0098	0.0167	0.007	0.0262
revision:$(No.\ Submissions)_{iqt-1}$	0.0714***	0.0074	0.1038***	0.0112
do-both:$(No.\ Submissions)_{iqt-1}$	0.0712***	0.0112	0.0851***	0.0176
redesign:$AvgRating_{iqt-1}$	0.045	0.1686	0.2212	0.2273
revision:$AvgRating_{iqt-1}$	0.0429	0.0925	0.2228	0.1273
do-both:$AvgRating_{iqt-1}$	0.0652	0.1472	0.2643	0.209
redesign:$BestRating_{iqt-1}$	0.3267**	0.1019	0.1925	0.1326
revision:$BestRating_{iqt-1}$	0.3728***	0.0599	0.3051***	0.082
do-both:$BestRating_{iqt-1}$	0.2326*	0.0945	0.126	0.1277
redesign:$SecondBestRating_{iqt-1}$	-0.0705	0.1086	-0.14	0.1492
revision:$SecondBestRating_{iqt-1}$	-0.0095	0.0624	-0.0693	0.0891
do-both:$SecondBestRating_{iqt-1}$	-0.0345	0.0988	-0.1536	0.1428
redesign:$Award_q(\$)$	0.0006	0.0005	0.0005	0.0006
revision:$Award_q(\$)$	0.0015***	0.0003	0.0007	0.0004
do-both:$Award_q(\$)$	0.0016***	0.0004	0.0008	0.0005
redesign:$(No.\ 1\text{-}Star)_{qt-1}$	0.0091**	0.0035	0.0111*	0.005
revision:$(No.\ 1\text{-}Star)_{qt-1}$	0.0114***	0.0024	0.0073*	0.0033
do-both:$(No.\ 1\text{-}Star)_{qt-1}$	0.0148***	0.004	0.0052	0.0047
redesign:$(No.\ 2\text{-}Star)_{qt-1}$	0.0068	0.0043	0.0133*	0.0054
revision:$(No.\ 2\text{-}Star)_{qt-1}$	0.0069*	0.0027	0.0094**	0.0035
do-both:$(No.\ 2\text{-}Star)_{qt-1}$	0.0098*	0.0044	0.0142**	0.0049
redesign:$(No.\ 3\text{-}Star)_{qt-1}$	0.0012	0.0036	0.0097*	0.0045
revision:$(No.\ 3\text{-}Star)_{qt-1}$	0.0076***	0.0022	0.0081**	0.0027
do-both:$(No.\ 3\text{-}Star)_{qt-1}$	0.0099**	0.0037	0.0012	0.0041
redesign:$(No.\ 4\text{-}Star)_{qt-1}$	0.0008	0.0056	0.0045	0.0075
revision:$(No.\ 4\text{-}Star)_{qt-1}$	0.0048	0.0033	0.0043	0.0042
do-both:$(No.\ 4\text{-}Star)_{qt-1}$	0.0079	0.0056	0.0192***	0.0056
redesign:$(No.\ 5\text{-}Star)_{qt-1}$	-0.0216	0.0111	0.0039	0.0146
revision:$(No.\ 5\text{-}Star)_{qt-1}$	-0.0285***	0.0062	-0.0206*	0.0087
do-both:$(No.\ 5\text{-}Star)_{qt-1}$	-0.0315**	0.0107	-0.0262	0.0143
redesign:$(No.\ Submissions)_{qt-1}$	-0.0027	0.0034	-0.0137**	0.0046
revision:$(No.\ Submissions)_{qt-1}$	-0.0029	0.0021	-0.001	0.0028
do-both:$(No.\ Submissions)_{qt-1}$	-0.0071*	0.0035	-0.0066	0.004
redesign:$(No.\ Creators)_{qt-1}$	-0.0001	0.0077	0.0135	0.0101
revision:$(No.\ Creators)_{qt-1}$	-0.0187***	0.0048	-0.0148*	0.0064
do-both:$(No.\ Creators)_{qt-1}$	-0.0193*	0.0079	0.0044	0.0093
Time Dummies	Yes			
Observations	15,424		8,661	
R^2	0.0408		0.0452	
Log Likelihood	−9,605.70		−5,134.21	
LR Test (df = 54)	817.4216***		485.8946***	
Note:	*p<0.05; **p<0.01; ***p<0.001			

Table B.18: Difference in Coefficients between Regressions of Incumbents' Actions for Low- and High- Type Creators

	$coef_H$	$coef_L$	se_H	se_L	$\|coef_H - coef_L\|$	t-value	p-value
redesign:$Intercept$	-4.588	-4.097	0.244	0.289	0.491	-1.297	0.195
revision:$Intercept$	-3.909	-4.358	0.152	0.200	0.449	1.787	0.074
do-both:$Intercept$	-4.148	-4.542	0.230	0.285	0.394	1.075	0.283
redesign:$(No.\ Submissions)_{iqt-1}$	0.010	0.007	0.017	0.026	0.003	0.090	0.928
revision:$(No.\ Submissions)_{iqt-1}$	0.071	0.104	0.007	0.011	0.032	-2.414	0.016 *
do-both:$(No.\ Submissions)_{iqt-1}$	0.071	0.085	0.011	0.018	0.014	-0.666	0.505
redesign:$AvgRating_{iqt-1}$	0.045	0.221	0.169	0.227	0.176	-0.623	0.534
revision:$AvgRating_{iqt-1}$	0.043	0.223	0.093	0.127	0.180	-1.143	0.253
do-both:$AvgRating_{iqt-1}$	0.065	0.264	0.147	0.209	0.199	-0.779	0.436
redesign:$BestRating_{iqt-1}$	0.327	0.193	0.102	0.133	0.134	0.802	0.422
revision:$BestRating_{iqt-1}$	0.373	0.305	0.060	0.082	0.068	0.667	0.505
do-both:$BestRating_{iqt-1}$	0.233	0.126	0.095	0.128	0.107	0.671	0.502
redesign:$SecondBestRating_{iqt-1}$	-0.071	-0.140	0.109	0.149	0.070	0.377	0.706
revision:$SecondBestRating_{iqt-1}$	-0.010	-0.069	0.062	0.089	0.060	0.550	0.582
do-both:$SecondBestRating_{iqt-1}$	-0.035	-0.154	0.099	0.143	0.119	0.686	0.493
redesign:$Award_q(\$)$	0.001	0.001	0.001	0.001	0.000	0.128	0.898
revision:$Award_q(\$)$	0.002	0.001	0.000	0.000	0.001	1.600	0.110
do-both:$Award_q(\$)$	0.002	0.001	0.000	0.001	0.001	1.249	0.212
redesign:$(No.\ 1\text{-}Star)_{qt-1}$	0.009	0.011	0.004	0.005	0.002	-0.328	0.743
revision:$(No.\ 1\text{-}Star)_{qt-1}$	0.011	0.007	0.002	0.003	0.004	1.005	0.315
do-both:$(No.\ 1\text{-}Star)_{qt-1}$	0.015	0.005	0.004	0.005	0.010	1.555	0.120
redesign:$(No.\ 2\text{-}Star)_{qt-1}$	0.007	0.013	0.004	0.005	0.007	-0.942	0.346
revision:$(No.\ 2\text{-}Star)_{qt-1}$	0.007	0.009	0.003	0.004	0.003	-0.566	0.572
do-both:$(No.\ 2\text{-}Star)_{qt-1}$	0.010	0.014	0.004	0.005	0.004	-0.668	0.504
redesign:$(No.\ 3\text{-}Star)_{qt-1}$	0.001	0.010	0.004	0.005	0.009	-1.475	0.140
revision:$(No.\ 3\text{-}Star)_{qt-1}$	0.008	0.008	0.002	0.003	0.001	-0.144	0.886
do-both:$(No.\ 3\text{-}Star)_{qt-1}$	0.010	0.001	0.004	0.004	0.009	1.575	0.115
redesign:$(No.\ 4\text{-}Star)_{qt-1}$	0.001	0.005	0.006	0.008	0.004	-0.395	0.693
revision:$(No.\ 4\text{-}Star)_{qt-1}$	0.005	0.004	0.003	0.004	0.001	0.094	0.925
do-both:$(No.\ 4\text{-}Star)_{qt-1}$	0.008	0.019	0.006	0.006	0.011	-1.427	0.154
redesign:$(No.\ 5\text{-}Star)_{qt-1}$	-0.022	0.004	0.011	0.015	0.026	-1.390	0.164
revision:$(No.\ 5\text{-}Star)_{qt-1}$	-0.029	-0.021	0.006	0.009	0.008	-0.739	0.460
do-both:$(No.\ 5\text{-}Star)_{qt-1}$	-0.032	-0.026	0.011	0.014	0.005	-0.297	0.767
redesign:$(No.\ Submissions)_{qt-1}$	-0.003	-0.014	0.003	0.005	0.011	1.923	0.054
revision:$(No.\ Submissions)_{qt-1}$	-0.003	-0.001	0.002	0.003	0.002	-0.543	0.587
do-both:$(No.\ Submissions)_{qt-1}$	-0.007	-0.007	0.004	0.004	0.001	-0.094	0.925
redesign:$(No.\ Creators)_{qt-1}$	0.000	0.014	0.008	0.010	0.014	-1.071	0.284
revision:$(No.\ Creators)_{qt-1}$	-0.019	-0.015	0.005	0.006	0.004	-0.488	0.626
do-both:$(No.\ Creators)_{qt-1}$	-0.019	0.004	0.008	0.009	0.024	-1.942	0.052

B.11.2.3 Stratified-Sample Analysis

pointer:stratify

To further demonstrate that our estimation results are robust to the inclusion/exclusion of creator ex-ante heterogeneity in their ability/experience, and that not incorporating creator ex-ante heterogeneity does not affect the nature of our estimation results, we re-estimate the main model using stratified sub-samples. Operationally, we stratify contests in the complete sample into *High-Reputation Concentrated Contests*, *Low-Reputation Concentrated*

Contests, and *Balanced Contests*, based on the percentage of high-reputation creators in the contest,[18] and we estimate our structural model using *High-Reputation Concentrated* and *Low-Reputation Concentrated Contests* separately, and compare the estimation results based on the two sub-samples with the full-sample results.

As we can see in Table B.19, the *High-Reputation Concentrated Contests* and *Low-Reputation Concentrated* sub-samples yield similar structural estimates, and both sets of estimates are very similar to our main estimation results. We conclude that creator ability/experience heterogeneity does not meaningfully affect our estimation results.

Table B.19: Results for Stratified-Sample Analysis and the Alternative Model with Non-Monetary Incentives

Parameter	Main Model		High-Reputation Concentrated		Low-Reputation Concentrated		With Non-Monetary Incentive	
α	0.034	(0.002)	0.034	(0.001)	0.033	(0.001)	0.028	(0.001)
$c(redesign)$	2.790	(0.010)	2.790	(0.044)	2.790	(0.011)	2.784	(0.008)
$c(revise)$	2.205	(0.021)	2.157	(0.027)	2.254	(0.035)	2.204	(0.006)
$c(do\text{-}both)$	3.319	(0.031)	3.313	(0.023)	3.324	(0.023)	3.293	(0.010)
$c_1^e(enter)$	4.958	(0.028)	4.919	(0.030)	5.016	(0.053)	5.032	(0.097)
$c_2^e(enter)$	4.326	(0.043)	4.388	(0.016)	4.411	(0.014)	4.466	(0.025)
$c_3^e(enter)$	3.599	(0.014)	3.814	(0.032)	3.490	(0.020)	3.814	(0.020)
R_{nm}	–		–		–		0.233	(0.007)

Note: The numbers in parenthesis are standard errors.

B.11.2.4 Reduced-form Test for Hetereogeneity in Creator Interest

One may be concerned that the heterogeneity in creator interest in a contest may affect creators' state and their actions simultaneously. To test if creators' heterogeneous "interest" in a contest impacts their follow-up decisions, we perform the following reduced-form empirical analysis. In this analysis, we record how many times a creator participates in contests in each industry within the sampled period (e.g., retailing, software engineering, banking, etc.), and measure creator i's interest in contest q using $(\frac{No.\ Contest\ Same\ Industry}{No.\ All\ Contest})_{iq}$, i.e., the percentage of contests creator i participated in (throughout the sample period) that fall in the same industry as contest q. With this measure for a creator's interest in a contest, we re-run the regression reported in Online Appendix B.2.1 Table B.2 — a multinomial logit regression model of incumbents' follow-up action choices. Specifically, the dependent variable is a nominal variable denoting incumbent i's choice $Action_{iqt}$ among *redesign, revise, do-both*, and *do-nothing*, where the reference category is *do-nothing.* We add $(\frac{No.\ Contest\ Same\ Industry}{No.\ All\ Contest})_{iq}$ as an independent variable, in addition to the three main sets of independent variables in the original version of the multinomial logit regression (reported in

[18] *High-Reputation Concentrated Contests*: contests with a high proportion ($\geq$ upper quartile of the proportion observed in all contests in the data) of high ability creators; *Low-Reputation Concentrated Contests:* contests with a low proportion ($\leq$ lower quartile of the proportion observed in all contests in the data) of high ability creators; and *Balanced Contests:* the remaining contests.

Table B.2): (1) the individual-level variables, including the number of submissions that the *focal* creator has made previously ($(No.Submissions)_{iqt-1}$), and among all her previous submissions her best rating ($(BestRating)_{iqt-1}$), second-best rating ($(SecondBestRating)_{iqt-1}$), and average rating ($(Avg\ Rating)_{iqt-1}$), (2) the contest-level rating variables, including $(No.\ 1\text{-}Star)_{qt-1}$, $(No.\ 2\text{-}Star)_{qt-1}$... $(No.\ 5\text{-}Star)_{qt-1}$, and (3) control variables, including $(No.\ Submissions)_{qt-1}$ and $(No.\ Creators)_{qt-1}$, the amount of award for the contest ($Award_q$ in \$) and time dummies. We report the estimation results in Table B.20. As can be seen from the table, $(\frac{No.\ Contest\ Same\ Industry}{No.\ All\ Contest})_{iq}$, the proxy for the creator interest, does not significantly affect her choice among *redesign, revise, do-both* and *do-nothing* (after controlling for the individual- and contest-level states captured in the original model). To sum up, we find little evidence for heterogeneity in creators' interest that affects their follow-up decisions.

Table B.20: Is Creator Interest Affecting the Incumbent Follow-up Action Choice – Multinomial Logit Regression

	Depend Variable: $Action_{iqt}$					
	re-design		*revision*		*do-both*	
Creator Interest						
$(\frac{No.\ Contest\ Same\ Industry}{No.\ All\ Contest})_{iq}$	−0.042	(0.164)	0.015	(0.102)	0.133	(0.149)
Individual-Level Variables						
$(No.\ Submissions)_{iqt-1}$	0.011	(0.014)	0.080***	(0.006)	0.075***	(0.009)
$AvgRating_{iqt-1}$	0.099	(0.135)	0.101	(0.074)	0.139	(0.119)
$BestRating_{iqt-1}$	0.269***	(0.081)	0.353***	(0.048)	0.190*	(0.076)
$SecondBestRating_{iqt-1}$	−0.096	(0.088)	−0.033	(0.051)	−0.076	(0.081)
Contest-Level Variables						
$Award_q$(\$)	0.001	(0.0004)	0.001***	(0.0002)	0.001***	(0.0003)
$(No.\ 1\text{-}Star)_{qt-1}$	0.010***	(0.003)	0.010***	(0.002)	0.011***	(0.003)
$(No.\ 2\text{-}Star)_{qt-1}$	0.009**	(0.003)	0.008***	(0.002)	0.012***	(0.003)
$(No.\ 3\text{-}Star)_{qt-1}$	0.005	(0.003)	0.008***	(0.002)	0.006*	(0.003)
$(No.\ 4\text{-}Star)_{qt-1}$	0.002	(0.004)	0.005*	(0.003)	0.014***	(0.004)
$(No.\ 5\text{-}Star)_{qt-1}$	−0.012	(0.009)	−0.027***	(0.005)	−0.030***	(0.008)
$(No.\ Submissions)_{qt-1}$	−0.007*	(0.003)	−0.002	(0.002)	−0.007**	(0.003)
$(No.\ Creators)_{qt-1}$	0.005	(0.006)	−0.018***	(0.004)	−0.009	(0.006)
Time Dummies			Yes			
Observations	24,085		Log Likelihood		-14,782.610	
R^2	0.040		LR Test		1,247.877***	

Note: *p<0.05; **p<0.01; ***p<0.001; the numbers in parenthesis are standard errors.

B.11.3 Alternative Model Allowing for Creators' Strategic Waiting Behavior

B.11.3.1 Alternative Model of Strategic Waiting

We consider an alternative model that allows for the possibility that creators might strategically wait to join a contest. In this alternative model, the number of potential entrants in a particular period is endogenously determined. In particular, this new model differs from the main model (presented in the paper's Section 4) in the following ways: (1) We now assume that contests start with a fixed number of potential entrants (defined as all active creators on the platform on each day), and in each period, the potential entrants are those in the fixed pool who have not yet joined the contest. In other words, the number of potential entrants in any given period equals the number of all active creators on the platform ($|M_t|$), minus those who are already participating in the focal contest (i.e., $|M_t| = |M_1| - |N_t|$) (Recall that $|N_t|$ is the number of incumbents in period t.) (2) We also assume that potential entrants can strategically wait – they will decide between entering the contest in the current period and waiting for another period. To incorporate (2) in our model, we assign a continuation payoff (i.e., a value function) to a potential entrant who decides "not to enter" in the current period. That is, a potential entrant j will decide whether to enter or not ($d_{j,t}$) based on Equation (B.22) (as opposed to Equation (B.21) in the main model).

$$V_{j,t}^{e,\sigma}(s_t, \varepsilon_{j,t}) = \max_{d_{j,t}} \left\{ U_{j,t}^e(\varepsilon_{j,t}; d_{j,t}) + \mathbb{I}(d_{j,t} = Enter) \cdot \beta \mathbb{E} V_{j,t+1}^{\sigma}(x_{j,t+1}, s_{t+1}, \epsilon_{j,t+1}) \right\}. \quad \text{(B.21)}$$

$$V_{j,t}^{e,\sigma}(s_t, \varepsilon_{j,t}) = \max_{d_{j,t}} \begin{cases} U_{j,t}^e(\varepsilon_{j,t}; d_{j,t}) + \beta \mathbb{E} V_{j,t+1}^{\sigma}(x_{j,t+1}, s_{t+1}, \epsilon_{j,t+1}), & \text{if } d_{j,t} = Enter \\ 0 + \beta \mathbb{E} V_{j,t+1}^{e,\sigma}(s_{t+1}, \varepsilon_{j,t+1}), & \text{if } d_{j,t} = NotEnter. \end{cases} \quad \text{(B.22)}$$

We estimate this new model, and report the estimation results in Table B.21. As we can see from Table B.21, the estimation results of the utility parameters are similar to our main estimation results, and the qualitative conclusions remain unchanged. In particular, we still find that the entry costs are decreasing over time, suggesting that the decline in the entry cost estimates in time (t) is not driven by how we model potential entrants' decisions and that allowing for potential entrants' strategic waiting does not meaningfully affect our estimation results. However, we do notice that the rate at which the entry cost decreases is slightly slower under this alternative model. This suggests that strategic waiting and a smaller number of potential entrants in later periods can partially explain the entry patterns, but only to a limited degree.

Table B.21: Results for Stratified-Sample Analysis and the Alternative Model with Non-Monetary Incentives

Parameter	Main Model		Strategic Waiting	
α	0.034	(0.002)	0.030	(0.002)
$c(redesign)$	2.790	(0.010)	2.722	(0.073)
$c(revise)$	2.205	(0.021)	2.155	(0.042)
$c(do\text{-}both)$	3.319	(0.031)	3.252	(0.080)
$c_1^e(enter)$	4.958	(0.028)	4.649	(0.052)
$c_2^e(enter)$	4.326	(0.043)	4.244	(0.030)
$c_3^e(enter)$	3.599	(0.014)	3.531	(0.017)

Note: The numbers in parenthesis are standard errors.

B.11.3.2 Reduced-form Evidence for Strategic Waiting

Next, we use a reduced-form analysis to show which one of the following two assumptions is more likely to be true – (1) the number of potential entrants is endogenous (as in the strategic waiting model); (2) the number of potential entrants is fixed (as in our main model). The probability of any potential entrant entering the focal contest in period t is a function of the contest-level state at the beginning of the current period (s_t). Suppose (1) is true, then the number of potential entrants in a period t ($|M_t|$) would be affected by the number of actual entrants from the previous period $t-1$, and $|M_t|$ would depend on the contest-level state at the beginning of the previous period (s_{t-1}) as well. As a result, the number of actual entrants in period t ($\Delta Creator_{qt}$) – the product of the number of potential entrants and the probability of entering – would be a function of the contest-level state of both the previous period and the current period (i.e., s_{t-1} and s_t). By contrast, if (2) is true, then the number of actual entrants in period t ($\Delta Creator_{qt}$) would be a function of only the current period contest-level state s_t, but not the previous period contest-level state s_{t-1}.

To test which one of (1) and (2) is more consistent with our data, we run the following analysis. We regress the number of actual entrants in contest q in period t ($\Delta Creator_{qt}$) on both the current-period state variables s_t (measured by $(No.\ 1\text{-}Star)_{qt}$, $(No.\ 2\text{-}Star)_{qt}$, $(No.\ 3\text{-}Star)_{qt}$, $(No.\ 4\text{-}Star)_{qt}$, $(No.\ 5\text{-}Star)_{qt}$, $(No.\ Submissions)_{qt}$ and $(No.Creators)_{qt}$), and the previous-period state variables s_{t-1} (measured by $(No.1\text{-}Star)_{qt-1}$, $(No.2\text{-}Star)_{qt-1}$, $(No.3\text{-}Star)_{qt-1}$, $(No.4\text{-}Star)_{qt-1}$, $(No.5\text{-}Star)_{qt-1}$, $(No.Submissions)_{qt-1}$ and $(No.\ Creators)_{qt-1}$). The results are reported as Model 1 in Table B.22. (Model 0 only contains the set of variables describing the current period state.) From the results, we can see that most variables describing the previous period state are not significant (the only exception is $(No.\ 3\text{-}Star)_{qt-2}$), suggesting that assumption (2) seems to be more consistent with the data.

Table B.22: Regression of the Number of Entries With One v.s. Two Lags

	Dependent variable: $\Delta Creator_{qt}$	
	(Model 0)	(Model 1)
Current-Period State Variables s_t		
$(No.\ 1\text{-}Star)_{qt-1}$	0.026** (0.008)	0.015 (0.013)
$(No.\ 2\text{-}Star)_{qt-1}$	0.022* (0.010)	0.034* (0.016)
$(No.\ 3\text{-}Star)_{qt-1}$	0.0002 (0.008)	0.026 (0.014)
$(No.\ 4\text{-}Star)_{qt-1}$	−0.021 (0.012)	−0.049* (0.020)
$(No.\ 5\text{-}Star)_{qt-1}$	−0.149*** (0.022)	−0.121** (0.038)
$(No.\ Creators)_{qt-1}$	−0.202*** (0.015)	−0.203*** (0.023)
$(No.\ Submissions)_{qt-1}$	−0.004 (0.006)	0.007 (0.009)
Earlier-Period State Variables s_{t-1}		
$(No.\ 1\text{-}Star)_{qt-2}$		0.012 (0.015)
$(No.\ 2\text{-}Star)_{qt-2}$		−0.021 (0.018)
$(No.\ 3\text{-}Star)_{qt-2}$		−0.039* (0.016)
$(No.\ 4\text{-}Star)_{qt-2}$		0.032 (0.021)
$(No.\ 5\text{-}Star)_{qt-2}$		−0.039 (0.042)
$(No.\ Creators)_{qt-2}$		−0.014 (0.024)
$(No.\ Submissions)_{qt-2}$		−0.005 (0.010)
Observations	5,607	5,607
R^2	0.166	0.169
Adjusted R^2	0.025	0.027
F Statistic	136.369*** (df = 7; 4793)	69.650*** (df = 14; 4786)
Note:		*p<0.05; **p<0.01; ***p<0.001

B.11.4 Summary Table for Sensitivity Analysis for β, $||M_t||$, and SIFT-cutoff

Table B.23: Sensitivity Analysis with respect to β, $||M_t||$, and SIFT cutoff

	Main	β =**0.95**	β =**0.85**	$\|\|M_t\|\|$ =**350**	$\|\|M_t\|\|$ =**250**	SIFT **0.35**	SIFT **0.45**
$\|\|M_t\|\|$	300	300	300	350	250	300	300
β	0.9	0.95	0.85	0.9	0.9	0.9	0.9
SIFT cutoff	0.4	0.4	0.4	0.4	0.4	0.35	0.45
α	0.034	0.030	0.034	0.033	0.030	0.031	0.029
	(0.002)	(0.001)	(0.002)	(0.003)	(0.001)	(0.002)	(0.001)
$c(redesign)$	2.790	2.792	2.790	2.788	2.789	2.997	2.590
	(0.010)	(0.009)	(0.008)	(0.006)	(0.008)	(0.012)	(0.009)
$c(revise)$	2.205	2.205	2.204	2.205	2.212	2.047	2.217
	(0.021)	(0.011)	(0.006)	(0.011)	(0.015)	(0.031)	(0.052)
$c(do\text{-}both)$	3.319	3.298	3.296	3.297	3.294	3.476	3.267
	(0.031)	(0.025)	(0.013)	(0.015)	(0.027)	(0.183)	(0.059)
$c^e_1(enter)$	4.958	5.070	4.811	5.100	4.736	4.932	4.933
	(0.028)	(0.023)	(0.018)	(0.029)	(0.017)	(0.015)	(0.022)
$c^e_2(enter)$	4.326	4.292	4.270	4.483	4.148	4.291	4.245
	(0.043)	(0.023)	(0.025)	(0.010)	(0.019)	(0.025)	(0.026)
$c^e_3(enter)$	3.599	3.578	3.593	3.722	3.407	3.571	3.555
	(0.014)	(0.027)	(0.019)	(0.013)	(0.020)	(0.022)	(0.033)

Note: The numbers in parenthesis are standard errors.

B.12 Policy Simulation

B.12.1 Policy Simulation Model Specification

In policy simulations, the four feedback disclosure policies correspond to four different information structures. Here is how we reflect these differences in the creators' behavioral model for each alternative feedback policy.

No Feedback Policy Under the no feedback policy, a creator only gets to see prior actions taken by existing creators, but not the realized ratings. Each creator can only be characterized by the actions she has taken so far, and therefore, we define incumbent i's individual-level state variable under the no feedback policy as her actions up to the beginning of the current period (t) (denoted as $\mathbf{x}_{i,t}^{NF}$). Operationally, $\mathbf{x}_{i,t}^{NF}$ is simply a vector recording all the actions that have been taken by creator i in each period since she entered the contest. For example, if a creator i enters in Period 1 and takes the *"redesign"* action in Period 2, her individual-level state at the beginning of Period 3 is then $\mathbf{x}_{i,3}^{NF} = (\text{enter}, \text{redesign})$. The contest-level state is then redefined as s_t^{NF}, which summarizes the number of creators whose individual-level state is $\mathbf{x}_{i,t}^{NF}$ for all possible values of $\mathbf{x}_{i,t}^{NF}$. $\mathbf{x}_{i,t}^{NF}$ transitions deterministically into $\mathbf{x}_{i,t+1}^{NF}$ by simply appending $a_{i,t}$, i.e.,

$$\mathbf{x}_{i,t+1}^{NF} = (\mathbf{x}_{i,t}^{NF}, a_{i,t}). \tag{B.23}$$

Everything else is modeled in the same way as in the main model, except that we now have to update the calculation of creators' winning probabilities conditional on the newly defined state variables ($\Pr(i \text{ wins}|\mathbf{x}_{i,T+1}^{NF}, s_{T+1}^{NF})$, more specifically, $\Pr(i \text{ wins}|\mathbf{x}_{i,t=4}^{NF}, s_{t=4}^{NF})$ in our 3-period model). Like in the main model, to calculate the creator i's winning probability, we need a mapping between the creator's state variable ($\mathbf{x}_{i,4}^{NF}$) and the quality of her best design ($x_{i,4}^{true}$). We calculate the distribution of $x_{i,4}^{true}|\mathbf{x}_{i,4}^{NF}$ based on the distribution of the rating of creators' first submission and the action-dependent per-period state transition probabilities estimated from the data. For example, if $\mathbf{x}_{i,4}^{NF} = (\text{enter}, \text{redesign})$, then the cummulative distribution function of $x_{i,4}^{true}$ is as follows:

$$\Pr(x_{i,4}^{true} \le x|\mathbf{x}_{i,4}^{NF}) =$$
$$\begin{cases} \sum_{y\le x}^{x} \sum_{l\le y}^{y} p^e(l) p(x_{i,4}{=}y|x_{i,3}{=}l, \text{redesign}) & , \forall x \in \{nan, 1, 2, 3, 4\}; \\ \Pr(x_{i,4}^{true} \le 4|\mathbf{x}_{i,4}^{NF}) + & \\ \quad \sum_{l\le 5}^{5} p^e(l) p(x_{i,4}{=}5|x_{i,3}{=}l, \text{redesign}) & \\ \quad p(5{+}\max\{0, \xi_0\}{+}\max\{0, \xi_{\text{redesign}}^{(1)}\} \le x) & , \forall x \ge 5. \end{cases} \tag{B.24}$$

Late Feedback Policy Under the late feedback scenario, feedback is turned on after the second period. Hence, the late feedback scenario consists a "no-feedback" stage ($t = 2$) and a "with-feedback" stage ($t = 3$).[19] We define state variables differently for the two stages.

During the "no-feedback" stage ($t = 2$), creators only observe their own and other contest participants' actions, similar to the no feedback case. We thus define the state variable ($\mathbf{x}_{i,2}^{LF}$) as whether a creator enters the focal contest in the first period or not. The contest-level state (s_2^{LF}) is defined correspondingly as the number of entrants in the first period.

Once the seeker discloses the performance feedback at the beginning of Period 3, the individual-level state variable, $\mathbf{x}_{i,3}^{LF}$, is the observed individual best rating.

It is worth pointing out how the "no-feedback" stage and "with-feedback" stage are connected. The distribution of the best rating ($\mathbf{x}_{i,3}^{LF}$), which will be revealed at the beginning of Period 3, is calculated using on the distribution of the rating of creators' first submission and the action-dependent per-period state transition probabilities estimated from the data. We use these ingredients corresponding to the focal creator's entering time and any follow-up action. Specifically, if creator i enters in Period 2, $\mathbf{x}_{i,3}^{LF}$ is characterized by Equation (B.25a); if creator i enters in Period 1, then $\mathbf{x}_{i,3}^{LF}$ is characterized by Equation (B.25b).

$$\Pr(\mathbf{x}_{i,3}^{LF} = x | d_{i,2}\text{=enter}) \quad = \quad p^e(x); \tag{B.25a}$$

$$\Pr(\mathbf{x}_{i,3}^{LF} = x | d_{i,1}\text{=enter}, a_{i,2}) = \begin{cases} \sum_{l \leq x, l \neq 5}^{x} p^e(l) p(x_{i,3}{=}x | x_{i,2}{=}l, a_{i,2}) & , \forall x \in \{nan, 1, 2, 3, 4, 5\}; \\ p^e(5) & , \forall x = (5, a_{i,2}). \end{cases} \tag{B.25b}$$

Note that, as explained in Section 3.4.4, in the second line of Equation (B.25b), we record in $\mathbf{x}_{i,3}^{LF}$ the follow-up action(s) after a creator receives her first 5-star to recover the true best quality of her submissions truncated at 5-star.

Early Feedback Policy Under the early feedback policy, creators observe seeker ratings at the beginning of the second period, but not the third period. Hence, like the late feedback scenario, the early feedback scenario also consists of a "with-feedback" stage ($t = 2$) and a "no-feedback" stage ($t = 3$). We define state variables differently for the two stages.

In the "with-feedback" stage ($t = 2$), the information structure is the same as that under the full feedback policy, i.e., incumbent i's state variable ($\mathbf{x}_{i,2}^{EF}$) is defined as her observed best rating at the beginning of the second period.

The feedback is turned off after the second period. That is, no rating is disclosed at

[19] Note that, the "state" in period t is defined as the rating/action information at the beginning of period t; and in the very first stage ($t = 1$), since there has been no activity, the individual-level state is always empty, and the contest-level state (counting the number of creators with all possible individual-level states) is always a vector of zeros.

the end of Period 2, or equivalently, the beginning of Period 3. In this "no-feedback" stage ($t = 3$), creators have to make decisions based on the available information, including the realized ratings disclosed at the beginning of the second period ($x_{i,2}$), and participants' actions in the second period ($a_{i,2}$). Accordingly, we define an incumbent i's individual-level state variable in the third period as a vector of all the available information, i.e., $\mathbf{x}_{i,3}^{EF} = (x_{i,2}, a_{i,2})$. The contest-level state is redefined as s_t^{EF}, which summarizes the number of creators whose individual-level state is $\mathbf{x}_{i,t}^{EF}$ for all possible values of $\mathbf{x}_{i,t}^{EF}$.

In order to compute creator i's winning probability ($\Pr(i \text{ wins}|\mathbf{x}_{i,T+1}^{EF}, s_{T+1}^{EF})$, more specifically, $\Pr(i \text{ wins}|\mathbf{x}_{i,4}^{EF}, s_4^{EF})$ in our 3-period setting) in the early feedback case, we need to establish a mapping between the creator's state variable ($\mathbf{x}_{i,4}^{EF}$) and the true quality of her best design ($x_{i,4}^{true}$). Suppose creator i entered in the first period. We calculate the distribution of $x_{i,4}^{true}|\mathbf{x}_{i,4}^{EF}$ based on i's realized rating at the beginning of the second period, as well as the period-2 and period-3 action-dependent state transition probabilities estimated from the data. The cumulative distribution function for $x_{i,4}^{true}|\mathbf{x}_{i,4}^{EF}$ is:

$$
\begin{aligned}
&\Pr(x_{i,4}^{true} \leq x|\mathbf{x}_{i,4}^{EF}) \\
&= \begin{cases}
\sum_{y\leq x}^{x}\sum_{l\leq y}^{y} p(x_{i,3}{=}l|x_{i,2}, a_{i,2})p(x_{i,4}{=}y|x_{i,3}{=}l, a_{i,3}) & ,\forall x \in \{nan, 1, 2, 3, 4\}; \\
\Pr(x_{i,4}^{true} \leq 4) + p(x_{i,3}{=}5|x_{i,2}, a_{i,2})p(5{+}\max\{0, \xi_0\}{+}\max\{0, \xi_{a_{i,3}}^{(2)}\} \leq x) & \\
\qquad + \sum_{l\leq 4}^{4}\{p(x_{i,3}{=}l|x_{i,2}, a_{i,2})p(x_{i,4}{=}5|x_{i,3}{=}l, a_{i,3})p(5{+}\max\{0, \xi_0\} \leq x)\} & \\
& ,\forall x \geq 5, \forall x_{i,2} \in \{nan, 1, 2, 3, 4\}; \\
p(5{+}\max\{0, \xi_0\}{+}\max\{0, \xi_{a_{i,2}}^{(1)}\}{+}\max\{0, \xi_{a_{i,3}}^{(2)}\} \leq x) & ,\forall x \geq 5, \forall x_{i,2} = 5.
\end{cases}
\end{aligned}
\tag{B.26}
$$

B.12.2 State Transition Probabilities in Policy Simulation

In this appendix, we clarify our assumptions on state transition probabilities in the counterfactuals. The high-level summary is as follows: (1) we model the quality-transition probabilities as being dependent on creators' *actions*; (2) our counterfactual simulations capture the change in quality-transition probabilities via the equilibrium *action* choices (i.e., $Pr_t(a_{i,t}|x_{i,t}, s_t)$), *which depend on the feedback policy.* The upshot is that the overall quality transition probabilities (with action integrated out, i.e., $p(x_{i,t+1}|x_{i,t}, s_t) = \sum_{a_{i,t}} Pr_t(a_{i,t}|x_{i,t}, s_t) \cdot p(x_{i,t+1}|x_{i,t}, a_{i,t})$) will be different across the feedback scenarios. Below we provide more details.

We model quality-transition probabilities as being dependent on creators' *actions.* In our model, creator i chooses her action $a_{i,t}$ based on her individual and contest-level state variables. **Given her action choice**, the *action-specific* quality-transition probabilities no longer depend on the contest-level state. This is not surprising, because the differences in *action-specific* quality transition probabilities already reflect the different levels of improvement each action could produce (note that creators' action choice in our model can also be viewed as their choice of effort level in a discrete space, and therefore, the action-specific transition probabilities captures the mapping between effort and quality improvement, which can be reasonably assumed to be the same under different feedback policies); hence, conditional on the chosen action, individuals' quality-transition probability should not be significantly affected by opponents' states. Empirical evidence supports this argument. Details of the data evidence are provided in Table B.4 in Online Appendix B.2.4.

To summarize, in our model the quality-transition probabilities depend on creators' *actions*, but not opponents' states, and these action-specific quality-transition probabilities remain the same for all feedback policies. In the simulation for alternative feedback policies, we model the change in creators' strategies/actions (both for the focal creator and for her opponents) resulting from different information availability, but keep the quality-transition probabilities conditional on the chosen actions constant. (We provide details of how we model creators' strategies across different feedback policies in the next point.)

Our counterfactual simulations capture the effect of opponent behavior. Here is how we capture in our counterfactual analyses how different information structures under different feedback policies might lead to different creators' behaviors. Operationally, the state variables are defined differently in different feedback scenarios based on the information structure in each particular case – early feedback: $\mathbf{x}_{i,t}^{\mathrm{EF}}$ and $\mathbf{s}_t^{\mathrm{EF}}$; late feedback: $\mathbf{x}_{i,t}^{\mathrm{LF}}$ and $\mathbf{s}_t^{\mathrm{LF}}$; no feedback: $\mathbf{x}_{i,t}^{\mathrm{NF}}$ and $\mathbf{s}_t^{\mathrm{NF}}$; and in the full feedback case, states variables are directly based on quality state variables, i.e., $x_{i,t}^{true}$ and s_t^{true}. These $\mathbf{x}_{i,t}^{\cdot}$ and $\mathbf{s}_t^{\cdot}$ are mapped to the quality states $x_{i,t}^{true}$ and s_t^{true}, with the assumption that creators have **rational expectations**. (See Online Appendix B.12.1 for a detailed explanation of how we model each feedback scenario.) With this modification, we can use the same BRM algorithm as explained in Section 3.5.1.2 in the paper to solve for the equilibrium conditional choice probabilities (i.e. $\mathrm{P}_{i,t}^{*}(a_{i,t}|\mathbf{x}_{i,t}^{\cdot}, \mathbf{s}_t^{\cdot})$, where $a_{i,t} \in \{$*revision*, *redesign*, *do-both*, *do-nothing*$\}$), for each combination of $(\mathbf{x}_{i,t}^{\cdot}, \mathbf{s}_t^{\cdot})$ in each feedback scenario. Therefore, we are not keeping the unconditional transition probabilities fixed; instead, we solve for creators' corresponding equilibrium actions under different feedback policies, keeping the *action-specific* transition probabilities, which reflect the quality improvement resulting from the creators actions, fixed. As mentioned above, we provide empirical evidence that the quality-transition probabilities depend on creators' *actions*, but not opponents' states, and therefore, it is reasonable to assume that the action-specific quality-transition probabilities remain the same for all feedback policies.

Note that there may be efficiency loss when the creator acts without knowing the ratings under some alternative feedback policies. For example, a top performer, without knowing she has already been given a 5-star rating, might sub-optimally choose the *redesign* action or choose the *do-nothing* option; a low-performing creator, without knowing her previous designs are not favored by the buyer, might choose the *revision* action and continue working on her unsuccessful design style. That is, although the quality-transition probabilities are the same conditional on a specific action being chosen, the feedback policy might affect which action is chosen. Our model captures the effect that more feedback can better guide contestants' exploration and exploitation actions. In fact, as can be seen from the policy simulation results in Figure 3.4 on page 63 in the body of the paper, the no feedback policy performs very poorly in almost all performance measures. However, under late feedback this inefficiency is less severe and is offset by the positive effect of reduced competition, as is also illustrated in Figure 3.4.

APPENDIX C

Appendices to Chapter 3

C.1 A Proof for Lemma 4.2.1

We now prove that under "Assumption 1-Perceived Utility" and "Assumption 2-Monotone Increasing Clicking Probability", controlling for product features Xs, p^b_{ij} and p^b_{ik} are independent conditional on $p^b_{i,-j-k}$, if and only if $\tilde{\epsilon}_{ij}$ and $\tilde{\epsilon}_{ik}$ in Equation (4.1) are independent conditional on $\tilde{\epsilon}_{i,-j-k}$, i.e.,

$$p^b_{ij} \perp p^b_{ik}|_{p^b_{i,-j-k},X} \Longleftrightarrow \tilde{\epsilon}_j \perp \tilde{\epsilon}_k|_{\tilde{\epsilon}_{i,-j-k},X}.$$

The proof utilizes the following two lemmas for independent variables:

Lemma C.1.1 *Let R_1 and R_2 denote two random variables, and f_1 and f_2 be functions of R_1 and R_2, respectively. If R_1 and R_2 are independent, then $f_1(R_1)$ and $f_2(R_2)$ are also independent, i.e.,*

$$R_1 \perp R_2 \Longleftrightarrow f_1(R_1) \perp f_2(R_2),\ \forall f.$$

Lemma C.1.2 *Let R_1, R_2 and T denote random variables, and $f(\cdot)$ be a strictly monotonic function. If R_1 and R_2 are independent conditional on T, i.e., $R_1 \perp R_2|_T$, then R_1 and R_2 are independent conditional on $f(T)$, i.e.,*

$$R_1 \perp R_2|_T \Longleftrightarrow R_1 \perp R_2|_{f(T)},\ \forall \textit{ a strictly monotonic } f.$$

Lemma C.1.1 suggests functions of independent variables also independent. Lemma C.1.2 suggests if two random variables are independent conditional on a third variable, then they are also independent conditional on any strictly monotonic function of the third variable. Proofs for Lemmas C.1.1 and C.1.2 are provided in Appendix C.1.1 and C.1.2, respectively.

According to Assumption 1, given product features Xs, $u^b_{ij}|_X = h(u_{ij} - u_{i0}; X) = f_h(\tilde{\epsilon}_{ij}; X)$, where f_h is a strictly increasing function. According to Assumption 2, given product features Xs, $p^b_{ij}|_X = g(u^b_{ij}) = g(f_h(\tilde{\epsilon}_{ij}; X))$. Since both $f_h(.)$ and $g(.)$ is strictly increasing, the composite function $H(x) = g(f_h(x))$ is also strictly increasing. Therefore,

$p^b_{ij}|_X = H(\tilde{\epsilon}_{i\cdot}; X)$, i.e., p^b_{ij} strictly increases with $\tilde{\epsilon}_{ij}$, after controlling for X. We refer to this as Observation 1.

Combining Observation 1 with Lemma C.1.1, we have $\tilde{\epsilon}_{ij} \perp \tilde{\epsilon}_{ik}|_{\tilde{\epsilon}_{i,-j-k},X} \iff H(\tilde{\epsilon}_{ij}) \perp H(\tilde{\epsilon}_{ik})|_{\tilde{\epsilon}_{i,-j-k},X} \iff p^b_{ij} \perp p^b_{ik}|_{\tilde{\epsilon}_{i,-j-k},X}$. Then combining Lemma C.1.1, we have $p^b_{ij} \perp p^b_{ik}|_{\tilde{\epsilon}_{i,-j-k},X} \iff p^b_{ij} \perp p^b_{ik}|_{H\tilde{\epsilon}_{i,-j-k},X} \iff p^b_{ij} \perp p^b_{ik}|_{p^b_{i,-j-k},X}$. Therefore, summarizing the above-mentioned two steps, we have $\tilde{\epsilon}_{ij} \perp \tilde{\epsilon}_{ik}|_{\tilde{\epsilon}_{i,-j-k},X} \iff p^b_{ij} \perp p^b_{ik}|_{p^b_{i,-j-k},X}$.

C.1.1 A Proof for Lemma C.1.1

For any two measurable sets $\mathcal{A}_i$, $i = 1, 2$, $T_i \in \mathcal{A}_i$ if and only if $R_i \in \mathcal{B}_i$, where $\mathcal{B}_i$ are the sets $\{s : f_i(s) \in \mathcal{A}_i\}$. Hence, since the R_i are independent, $\Pr(T_1 \in \mathcal{A}_1, T_2 \in \mathcal{A}_2) = \Pr(T_1 \in \mathcal{A}_1) * \Pr(T_2 \in \mathcal{A}_2)$. Thus, the T_i are independent.

C.1.2 A Proof for Lemma C.1.2

We know that $R_1 \perp R_2 | T$ Since f(.) is strictly monotone, we have $\Pr(f(T)) = \Pr(T)$ and $\Pr(F(T)|X) = \Pr(T|X), \forall X$. With that, we can write down the probability of both R_1 and R_2 conditional on T using the following two forms. First,

$$
\begin{aligned}
\Pr(R_1 \cap R_2|T) &= \Pr(R_1|T)\Pr(R_2|T) \\
&\frac{\Pr(R_1 \cap T)}{\Pr(T)}\frac{\Pr(R_2 \cap T)}{\Pr(T)} \\
&\frac{\Pr(T|R_1)\Pr(R_1)}{\Pr(T)}\frac{\Pr(T|R_2)\Pr(R_2)}{\Pr(T)} \\
&\Pr(R_1|F(T))\Pr(R_2|F(T));
\end{aligned}
\tag{C.1}
$$

Or,

$$
\begin{aligned}
\Pr(R_1 \cap R_2|T) &= \frac{\Pr(R_1 \cap R_2 \cap T)}{\Pr(T)} \\
&= \frac{\Pr(F(T)|R_1 \cap R_2)\Pr(R_1 \cap R_2)}{\Pr(F(T))} \\
&= \Pr(R_1 \cap R_2|F(T)).
\end{aligned}
\tag{C.2}
$$

Since the above mentioned two forms in Equations (C.1) and (C.2) equal each other. We have $\Pr(R_1 \cap R_2|F(T)) = \Pr(R_1|F(T))\Pr(R_2|F(T))$.

C.2 Log Likelihood

We show that negative log-likelihood is $\mathcal{L}(Z|X; \Phi^b) = \mathrm{tr} S_{zz}\Phi^b - \mathrm{logdet}\Phi^b$.

$$
\begin{aligned}
f(Z|X;\beta^b,\Sigma) &= (2\pi)^{-\frac{J}{2}}\det(\Sigma)^{-\frac{1}{2}}e^{-\frac{1}{2}(Z-\beta^b X)^{\mathrm{T}}\Sigma^{-1}(Z-\beta^b X)}(\text{Normal distribution})\\
f(Z|X;\beta^b,\Phi^b) &= (2\pi)^{-\frac{J}{2}}\det(\Phi^b)^{\frac{1}{2}}e^{-\frac{1}{2}(Z-\beta^b X)^{\mathrm{T}}\Phi^b(Z-\beta^b X)}\\
\ln f(Z|X;\beta^b,\Phi^b) &= \frac{N}{2}\ln\det(\Phi^b)-\frac{1}{2}\sum_{i=1}^{N}(z_n-\beta^b x_n)^{\mathrm{T}}\Phi^b(z_n-\beta^b x_n)+\text{const}
\end{aligned} \tag{C.3}
$$

Taking its derivative w.r.t. β^b and setting it to zero we have $\hat{\beta}^b = S_{xz}S_{xx}^{-1}$, where $S_{xz} = \frac{1}{N}\sum_{i=1}^{N}(z_i-\bar{z})(x_k-\bar{x})^{\mathrm{T}}$ and $S_{xx} = \frac{1}{N}\sum_{i=1}^{N}(x_i-\bar{x})(x_i-\bar{x})^{\mathrm{T}}$. $S_{zz} = \frac{1}{N}\sum_{i=1}^{N}[(z_i-\hat{\beta}^b x_i)(z_i-\hat{\beta}^b x_i)^{\mathrm{T}}]$.

Rewrite the log-likelihood using "trace trick"[1] w.r.t. Φ^b, we have:

$$
\begin{aligned}
\log f(Z|X;\Phi^b) &= \frac{N}{2}\log\det(\Phi^b)-\frac{1}{2}\sum_{i=1}^{N}(z_i-\hat{\beta}^b x_i)^{\mathrm{T}}\Phi^b(z_i-\hat{\beta}^b x_i)+\text{const}\\
&\propto \frac{N}{2}\log\det(\Phi^b)-\frac{1}{2}\sum_{i=1}^{N}\mathrm{tr}\left((z_i-\hat{\beta}^b x_i)(z_i-\hat{\beta}^b x_i)^{\mathrm{T}}\Phi^b\right)\\
&= \frac{N}{2}\log\det(\Phi^b)-\frac{1}{2}\mathrm{tr}\left(\sum_{i=1}^{N}[(z_i-\hat{\beta}^b x_i)(z_i-\hat{\beta}^b x_i)^{\mathrm{T}}]\Phi^b\right)\\
&= \log\det(\Phi^b)-\mathrm{tr}\left(\frac{1}{N}\sum_{i=1}^{N}[(z_i-\hat{\beta}^b x_i)(z_i-\hat{\beta}^b x_i)^{\mathrm{T}}]\Phi^b\right)\\
&= \log\det(\Phi^b)-\mathrm{tr}\left(S_{zz}\Phi^b\right)
\end{aligned} \tag{C.4}
$$

Therefore, we have $\mathcal{L}(Z|X;\Phi^b) = -\ln f(Z|X;\Phi^b) = \mathrm{tr}\left(S_{zz}\Phi^b\right) - \log\det(\Phi^b)$.

C.3 Algorithms

Step 1: We regress consumer i's clicking decision for product j on product features for all available products (p_k and X_{ik} for all k). The estimates for p_k and X_{ik} are denoted as $\hat{\alpha}_k^{\mathrm{b}}$ and $\hat{\beta}_k^{\mathrm{b}}$ respectively for each product $k \in \{1,2,...,J\}$. In particular, the vector summarizing all estimates is: $\hat{\boldsymbol{\beta}}^b = S_{xz}S_{xx}^{-1}$, where $S_{xz} = \frac{1}{N}\sum_{i=1}^{N}(z_i-\bar{z})(X_k-\bar{X})^{\mathrm{T}}$ and $S_{xx} = \frac{1}{N}\sum_{i=1}^{N}(X_i-\bar{X})(X_i-\bar{X})^{\mathrm{T}}$.

Step 2: We obtain the residuals (denoted at e_{ij}) from the clicking regression above—we subtract the the predicted clicking level from the actual click, i.e., $e_{ij} = z_{ij} - \hat{z}_{ij} = z_{ij} - \sum_{k=1}^{J}(\hat{\alpha}_{jk}^{\mathrm{b}}p_k + \hat{\beta}_{jk}^{\mathrm{b}}X_{ik})$.

[1]The "trace trick" is that $x^T Ax = \mathrm{tr}[x^T Ax] = \mathrm{tr}[x^T xA]$, where the first equality follows from the fact that $x^T Ax$ is a scalar.

Step 3: We compute the empirical covariance of these residual terms, which is a $J * J$ matrix S_{zz} among the J available products, in particular,

$$S_{zz} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{e}_i \boldsymbol{e}_i^T = \frac{1}{N}\sum_{i=1}^{N}[(z_i - \hat{\beta}^b X_i)(z_i - \hat{\beta}^b X_i)^{\mathrm{T}}], \tag{C.5}$$

where $\boldsymbol{e}_i$ is a vector, summarizing the residual terms of all J products (i.e., $\boldsymbol{e}_i = (e_{i1}, e_{i2}, ..., e_{iJ})$).

Step 4: Lastly, we use the Graphical Lasso algorithm (Friedman et al., 2008; Rothman et al., 2008) and form an estimator based on Equation (4.3) to learn a sparse precision matrix Φ^{b}.

C.4 Description of Tuning Parameter Selection Methods

We discuss three commonly used methods for tuning parameter selection: cross-validation (CV), Bayesian information criterion (BIC), and adaptive methods.

CV: A common method to choose the tuning parameter λ is cross-validation. Specifically, given a grid of penalties and K folds of the data, CV estimates the model based on $K - 1$ folds of the data, and score the performance on K^{th} fold; after doing these repeatly, CV determines the best λ that minimizes the mean errors.

BIC: An alternative to cross-validation are models based on information criteria. Cross-validation is effective especially when the objective is the prediction accuracy. However, it may select too many variables when the primary interest is model selection. To obtain a more sparse model for the purpose of inference, we would rely on Bayesian information criterion (BIC). It leads to asymptotically consistent model selection in the setting with fixed number of variables and growing sample size (Schwarz et al., 1978; Foygel and Drton, 2010).[2]

$$EBIC_{\kappa} = -n\mathcal{L}(S_{zz}; \Sigma^{\mathrm{b}}) + |E(S_{zz})| \cdot \log(n), \tag{C.6}$$

where $\mathcal{L}(S_{zz}; \Sigma^{\mathrm{b}})$ denotes the log-likelihood (MLE) between the estimated and sample covariance. $\mathcal{L}(S_{zz}; \Sigma^{\mathrm{b}}) = \mathrm{tr}S_{zz}\Phi^b - \mathrm{logdet}\Phi^b$. $|E(S_{zz})|$ denotes number of estimated edges or non-zeros in S_{zz}.

Adaptive methods: Another approach is the two-step adaptive Lasso (Zhou et al., 2011; Meinshausen, 2007). Specifically, we start with an initial sparse estimate $\hat{\Sigma}^{\mathrm{b},0}$, based on which we can derive a new penalization matrix (Λ) using data dependent weights W (Equation (C.7)). In the second step, we use this new penalty to refit the graphical lasso based on the

[2]Our setting does not fall into the category where are far more variables relative to the number of observations ($p >> n$). Therefore, Extended BIC proposed by Foygel and Drton (2010) reduces to the conventional BIC, with $\kappa = 0$ in EBIC.

new penalization matrix Λ. This second step refitting can be done with either CV or BIC.

$$\Lambda_{jk} = \lambda \cdot W_{jk}, \text{ where } W_{jk} = W_{kj} > 0 \text{ for all } (j,k),\ j \neq k \tag{C.7}$$

where the data-dependent weights (W) can be computed given the coefficient $\hat{\sigma}_{jk}^{\mathrm{b},0}$ in the initial sparse estimate $\hat{\Sigma}^{\mathrm{b},0}$, using either (1) "inverse" method: $W_{jk} = \frac{1}{|\hat{\sigma}_{jk}^{\mathrm{b},0}|}$ for non-zero coefficients and $W_{jk} = \max\{\frac{1}{\hat{\sigma}_{jk}^{\mathrm{b},0}}\}$ for the zero-valued coefficients, or (2) "inverse-squared" method: $W_{jk} = \frac{1}{(\hat{\sigma}_{jk}^{\mathrm{b},0})^2}$ for non-zero coefficients and $W_{jk} = \max\{\frac{1}{(\hat{\sigma}_{jk}^{\mathrm{b},0})^2}\}$ for the zero-valued coefficients. In sum, with these two weighting approaches and two re-fitting methods, there are four options within adaptive methods: Adaptive CV inverse, Adaptive CV inverse-square, Adaptive BIC inverse, Adaptive BIC inverse-square. In general, adaptive graphical lasso yields more sparser solutions and can be used to further reduce the number of false positives. While we test all four adaptive methods, we find squared inverse has better performance (lower false positive and lower false negative rates), so we only display results with square inverse weights in the table.

C.5 Alternative Clicking Assumptions

C.5.1 Alternative Functional Forms for the Clicking Probability

We evaluate whether the performance of the stage-1 is robustness with respect to the functional form assumption we had in the base case of our simulation, i.e., the Sigmoid function. Specifically, in simulation scenario (6), we consider two alternative types of functions for the clicking probability. In other words, we consider two alternative relationships between $(u_{ij} - u_{i0})$ and p^b.

Alternative-1: we assume consumer i clicks product j with probability $p_{ij}^b = H(u_{ij} - u_{i0})$, where $H()$ is a cumulative density function of a normal distribution. In particular, we assume the mean of this normal distribution equals 80^{th} quartile of u_{ij}, and the standard deviation of this normal distribution equals $\text{std}(u_{ij})$, where $\text{std}(u_{ij})$ is the sample standard deviation of the latent utilities. The results are robust to alternative levels of means and standard deviations.

Alternative-2: we assume consumer i clicks product j with probability $p_{ij}^b = H(\tilde{u}_{ij})$, where $H()$ is a linear function. In particular, we assume $p_{ij}^b = \frac{\tilde{u}_{ij} - \min(u_.)}{\max(u_.) - \min(u_.)}$, where $\min(u_.)$ is the minimum among all u_{ij} ($\forall i$, $\forall j$) and $\max(u_.)$ is the maximum among all u_{ij} ($\forall i$, $\forall j$). To further avoid some extreme values of $u_.$, which might lead to p_{ij}^b clustering around 0.5, we make some adjustments to set $\max(u_.)$ as the 95^{th} quartile of u_{ij} and $\min(u_.)$ as the 5^{th} quartile of u_{ij}; and to ensure p^b falls into the range of 0 to 1, we refine p_{ij}^b as $p_{ij}^b = \min\left(1, \max\left(0, \frac{\tilde{u}_{ij} - \min(u_.)}{\max(u_.) - \min(u_.)}\right)\right)$.

C.5.2 Uncertainties in Consumers' Clicking Decisions

In reality, whether a customer clicks a product or not could potentially be a noisy process. We want to demonstrate the robustness of our approach to those noises. We model some potential forms of noises when generating the synthetic data, and assess to what extend our step-1 estimation results are affected by those noises. Specifically, when modeling customers clicking process in the synthetic data, we allow for an type of uncertainty that may arise from potential behavioral factors. When deciding which product to click, a customer i might not be able to assess the utility of purchasing product j accurately. To model this possibility, we assume the perceived utility when making the clicking decision is

$$u_{ij}^{b} = h(u_{ij} - u_{i0} + \xi_{ij}), \tag{C.8}$$

where the added term ξ_{ij} reflects the deviation from the "true" product utility.

We generate two sets of data with two levels of this uncertainty. In particular, we assume uncertainties in customers' perceived utilities follow a normal distribution $N(0, 1.409)$ or $N(0, 2.818)$, where $\xi_{ij} \sim N(0, 1.409)$ or $\xi_{ij} \sim N(0, 2.818)$ in Equation (C.8). We then estimate parameters from these two sets of "noisy" data using our model to assess the robustness of our approach with respect to this uncertainty. All other simulation primitives are same as the ones in the base setting in Table 4.1. The stage-1 estimation results are represented in the last two rows in Table 4.1.

C.6 Summary Statistics for Product Features in the Empirical Setting

	price		*display*		*score*	
Product	mean	std	mean	std	mean	std
1	300.881	(2.697)	0.002	(0.001)	0.542	(0.244)
2	181.159	(2.969)	0.015	(0.003)	0.331	(0.186)
3	109.681	(4.549)	0.800	(0.054)	0.339	(0.370)
4	141.551	(1.268)	0.259	(0.029)	0.319	(0.168)
5	337.945	(6.956)	0.002	(0.000)	0.571	(0.233)
6	552.103	(0.982)	0.060	(0.008)	0.274	(0.079)
7	236.036	(5.697)	0.002	(0.000)	0.589	(0.273)
8	555.674	(7.388)	0.038	(0.016)	0.216	(0.338)
9	205.237	(0.889)	0.002	(0.001)	0.429	(0.247)
10	185.715	(4.743)	0.075	(0.013)	0.223	(0.120)
11	130.216	(3.552)	0.032	(0.007)	0.242	(0.132)
12	95.908	(4.431)	0.037	(0.024)	0.275	(0.250)
13	401.052	(2.847)	0.013	(0.004)	0.287	(0.180)
14	316.881	(2.659)	0.063	(0.017)	0.345	(0.132)
15	140.362	(1.136)	0.371	(0.049)	0.165	(0.159)
16	275.299	(1.728)	0.027	(0.014)	0.261	(0.152)
17	364.121	(5.913)	0.047	(0.028)	0.034	(0.058)
18	191.915	(5.119)	0.061	(0.020)	0.287	(0.098)
19	345.325	(1.442)	0.346	(0.090)	0.156	(0.167)
20	200.428	(3.887)	0.004	(0.000)	0.490	(0.240)
21	249.567	(7.998)	0.009	(0.001)	0.217	(0.307)
22	215.134	(13.891)	0.016	(0.005)	0.182	(0.154)
23	163.751	(4.653)	0.022	(0.004)	0.213	(0.178)
24	213.474	(4.099)	0.001	(0.000)	0.500	(0.298)
25	364.034	(4.285)	0.003	(0.001)	0.317	(0.133)
26	152.176	(2.214)	1.000	0.000	0.471	(0.463)
27	204.280	(8.330)	0.004	(0.005)	0.495	(0.259)
28	177.244	(4.028)	0.095	(0.017)	0.200	(0.146)
29	279.407	(5.222)	0.031	(0.003)	0.360	(0.196)
30	361.226	(10.249)	0.033	(0.016)	0.121	(0.156)
31	128.851	(3.758)	0.006	(0.004)	0.436	(0.203)
32	251.615	(2.862)	0.746	(0.054)	0.170	(0.243)
33	373.471	(5.131)	0.012	(0.004)	0.474	(0.134)
34	120.223	(1.254)	0.004	(0.001)	0.454	(0.229)
35	274.252	(5.548)	0.007	(0.003)	0.336	(0.240)
36	296.588	(5.698)	0.001	(0.000)	0.436	(0.291)
37	133.319	(2.145)	0.072	(0.054)	0.258	(0.143)
38	167.758	(1.607)	0.255	(0.061)	0.354	(0.175)
39	343.022	(2.872)	0.064	(0.008)	0.275	(0.095)
40	1129.432	(39.665)	0.300	(0.101)	0.077	(0.100)
41	171.460	(0.774)	0.107	(0.027)	0.262	(0.213)
42	119.093	(6.584)	0.018	(0.007)	0.339	(0.180)
43	374.354	(1.940)	0.051	(0.003)	0.342	(0.088)
44	189.722	(7.315)	0.023	(0.009)	0.130	(0.253)
45	188.957	(4.466)	0.035	(0.009)	0.309	(0.118)

Table C.1: Summary Statistics of Product Features in the Empirical Setting

BIBLIOGRAPHY

Victor Aguirregabiria and Pedro Mira. Dynamic discrete choice structural models: A survey. *Journal of Econometrics*, 156(1):38–67, 2010.

Zeynep Akşin, Barış Ata, Seyed Morteza Emadi, and Che-Lin Su. Structural estimation of callers' delay sensitivity in call centers. *Management Science*, 59(12):2727–2746, 2013.

Zeynep Aksin, Baris Ata, Seyed Morteza Emadi, and Che-Lin Su. Structural estimation of callers' delay sensitivity in call centers. *Management Science*, 59(12):2727–2746, 2013.

Laurence Ales, Soo-Haeng Cho, and Ersin Körpeoğlu. Innovation and crowdsourcing contests. Working paper, 2017a.

Laurence Ales, Soo-Haeng Cho, and Ersin Körpeoğlu. Optimal award scheme in innovation tournaments. *Operations Research*, 65(3):693–702, 2017b.

Elena A Allen, Eswar Damaraju, Sergey M Plis, Erik B Erhardt, Tom Eichele, and Vince D Calhoun. Tracking whole-brain connectivity dynamics in the resting state. *Cerebral cortex*, 24(3):663–676, 2014.

Tomomichi Amano, Andrew Rhodes, and Stephan Seiler. Large-scale demand estimation with search data. 2019.

Masaki Aoyagi. Information feedback in a dynamic tournament. *Games and Economic Behavior*, 70(2):242–260, 2010.

Peter Arcidiacono and Paul B Ellickson. Practical methods for estimation of dynamic discrete choice models. *Annu. Rev. Econ.*, 3(1):363–394, 2011.

Peter Arcidiacono and Robert A Miller. Identifying dynamic discrete choice models off short panels. Working paper, 2013.

Patrick Bajari, Han Hong, and Stephen P Ryan. Identification and estimation of a discrete game of complete information. *Econometrica*, 78(5):1529–1568, 2010.

Patrick Bajari, Victor Chernozhukov, Han Hong, and Denis Nekipelov. Identification and efficient semiparametric estimation of a dynamic discrete game. Technical report, National Bureau of Economic Research, 2015.

Erin Balogh, Sharyl J Nass, Margie Patlak, et al. *Extending the Spectrum of Precompetitive Collaboration in Oncology Research: Workshop Summary*. National Academies Press, 2010.

Onureena Banerjee, Laurent El Ghaoui, and Alexandre d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine learning research*, 9(Mar):485–516, 2008.

Kostas Bimpikis, Shayan Ehsani, and Mohamed Mostagir. Designing dynamic contests. *Operations Research*, 2017. Forthcoming.

Jesse Bockstedt, Cheryl Druehl, and Anant Mishra. Heterogeneous submission behavior and its implications for success in innovation contests with public submissions. *Production and Operations Management*, 25(7):1157–1176, 2016.

Kevin J Boudreau and Karim R Lakhani. "open" disclosure of innovations, incentives and follow-on reuse: Theory on processes of cumulative innovation and a field experiment in computational biology. *Research Policy*, 44(1):4–19, 2015.

Kevin J Boudreau, Nicola Lacetera, and Karim R Lakhani. Incentives and problem uncertainty in innovation contests: An empirical analysis. *Management Science*, 57 (5):843–863, 2011.

Kevin J Boudreau, Karim R Lakhani, and Michael Menietti. Performance responses to competition across skill levels in rank-order tournaments: Field evidence and implications for tournament design. *The RAND Journal of Economics*, 47(1):140–165, 2016.

T Tony Cai, Hongzhe Li, Weidong Liu, and Jichun Xie. Covariate-adjusted precision matrix estimation with an application in genetical genomics. *Biometrika*, 100(1): 139–156, 2013.

Felipe Caro and Jérémie Gallien. Inventory management of a fast-fashion retail network. *Operations Research*, 58(2):257–273, 2010.

Ying-Ju Chen, Tinglong Dai, C Gizem Korpeoglu, Ersin Körpeoğlu, Ozge Sahin, Christopher S Tang, and Shihong Xiao. Innovative online platforms: Research opportunities. Working paper, 2018.

Ying-Ju Chen, Tinglong Dai, C Gizem Korpeoglu, Ersin Körpeoğlu, Ozge Sahin, Christopher S Tang, and Shihong Xiao. Innovative online platforms: Research opportunities. *Manufacturing & Service Operations Management*, 2019.

Khai Chiong and Matthew Shum. Random projection estimation of discrete-choice models with large choice sets. *Available at SSRN 2748700*, 2016.

Clifford C Clogg, Eva Petkova, and Adamantios Haritou. Statistical methods for comparing regression coefficients between models. *American Journal of Sociology*, 100 (5):1261–1293, 1995.

Jacob Cohen. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213, 1968.

Allan Collard-Wexler. Demand fluctuations in the ready-mix concrete industry. *Econometrica*, 81(3):1003–1037, 2013.

Nigel Cross. Design cognition: Results from protocol and other empirical studies of design activity. In *Design knowing and learning: Cognition in design education*, pages 79–103. Elsevier, 2001.

Nigel Cross. *Design thinking: Understanding how designers think and work*. Berg, 2011.

Emmanuel Dechenaux, Dan Kovenock, and Roman M Sheremeta. A survey of experimental research on contests, all-pay auctions and tournaments. *Experimental Economics*, 18(4):609–669, 2015.

Jeffrey P Dotson, John R Howell, Jeff D Brazell, Thomas Otter, Peter J Lenk, Steve MacEachern, and Greg M Allenby. A probit model with structured covariance for similarity effects and source of volume calculations. *Journal of Marketing Research*, 55(1):35–47, 2018.

Michael Egesdal, Zhenyu Lai, and Che-Lin Su. Estimating dynamic discrete-choice games of incomplete information. *Quantitative Economics*, 6(3):567–597, 2015.

Sanjiv Erat and Vish Krishnan. Managing delegated search over design spaces. *Management Science*, 58(3):606–623, 2012.

Jacob Feldman, Dennis Zhang, Xiaofei Liu, and Nannan Zhang. Taking assortment optimization from theory to practice: Evidence from large field experiments on alibaba. *Available at SSRN*, 2018.

Kris Johnson Ferreira, Bin Hong Alex Lee, and David Simchi-Levi. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management*, 18(1):69–88, 2016.

Marshall Fisher, Santiago Gallino, and Jun Li. Competition-based dynamic pricing in online retailing: A methodology validated with field experiments. *Management Science*, 64(6):2496–2514, 2018.

Meredith Fowlie, Mar Reguant, and Stephen P Ryan. Market-based emissions regulation and industry dynamics. *Journal of Political Economy*, 124(1):249–302, 2016.

Jeremy T Fox. Semiparametric estimation of multinomial discrete-choice models using a subset of choices. *The RAND Journal of Economics*, 38(4):1002–1019, 2007.

Rina Foygel and Mathias Drton. Extended bayesian information criteria for gaussian graphical models. In *Advances in neural information processing systems*, pages 604–612, 2010.

Michael J French, JT Gravdahl, and MJ French. *Conceptual design for engineers*. Springer, 1985.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

Drew Fudenberg, Richard Gilbert, Joseph Stiglitz, and Jean Tirole. Preemption, leapfrogging and competition in patent races. *European Economic Review*, 22(1): 3–31, 1983.

Guillermo Gallego, Anran Li, Van-Anh Truong, and Xinshang Wang. Approximation algorithms for product framing and pricing. *Operations Research*, 2020.

Karan Girotra, Christian Terwiesch, and Karl T Ulrich. Idea generation and the quality of the best idea. *Management Science*, 56(4):591–605, 2010.

Maria Goltsman and Arijit Mukherjee. Interim performance feedback in multistage tournaments: The optimality of partial disclosure. *Journal of Labor Economics*, 29 (2):229–265, 2011.

Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(4):1–37, 2012.

Daniel P Gross. Performance feedback in competitive product development. *The RAND Journal of Economics*, 48(2):438–466, 2017.

Vassilis A Hajivassiliou. *Simulation estimation methods for limited dependent variable models*. Yale University, 1992.

Marina Halac, Navin Kartik, and Qingmin Liu. Contests for experimentation. *Journal of Political Economy*, 125(5):1523–1569, 2017.

John M Hammersley and Peter Clifford. Markov fields on finite graphs and lattices. *Unpublished manuscript*, 46, 1971.

V Joseph Hotz and Robert A Miller. Conditional choice probabilities and the estimation of dynamic models. *The Review of Economic Studies*, 60(3):497–529, 1993.

Ming Hu and Lu Wang. Simultaneous vs. sequential crowdsourcing contests. Working paper, 2017.

V Hubka. *Principles of engineering design*. Springer Verlag, New York, 1989.

Mitsuru Igami. Industry dynamics of offshoring: The case of hard disk drives. *American Economic Journal: Microeconomics*, 2016. Forthcoming.

Mitsuru Igami. Estimating the innovator's dilemma: Structural analysis of creative destruction in the hard disk drive industry, 1981–1998. *Journal of Political Economy*, 125(3):798–847, 2017.

Zhaohui Zoey Jiang, Yan Huang, and Damian R Beil. The role of feedback in dynamic crowdsourcing contests: A structural empirical analysis. Working paper, 2016.

Mireia Jofre-Bonet and Martin Pesendorfer. Estimation of a dynamic auction game. *Econometrica*, 71(5):1443–1489, 2003.

Michael P Keane and Kenneth I Wolpin. The solution and estimation of discrete choice dynamic programming models by simulation and interpolation: Monte carlo evidence. *The Review of Economics and Statistics*, pages 648–672, 1994.

Jun B Kim, Paulo Albuquerque, and Bart J Bronnenberg. Online demand under limited consumer search. *Marketing science*, 29(6):1001–1023, 2010.

Laura J Kornish and Karl T Ulrich. Opportunity spaces in innovation: Empirical analysis of large samples of ideas. *Management Science*, 57(1):107–128, 2011.

C Gizem Korpeoglu, Ersin Körpeoğlu, and Sıdıka Tunç. Optimal duration of innovation contests. Working paper, 2017.

Ersin Körpeoğlu and Soo-Haeng Cho. Incentives in contests with heterogeneous solvers. *Management Science*, 64(6):2709–2715, 2017.

Ersin Körpeoğlu, C Gizem Korpeoglu, and Isa Emin Hafalir. Parallel innovation contests. Working paper, 2017.

Steffen L Lauritzen. *Graphical models*, volume 17. Clarendon Press, 1996.

Thomas Y Lee and Eric T Bradlow. Automated marketing research using online customer reviews. *Journal of Marketing Research*, 48(5):881–894, 2011.

Tracy Xiao Liu, Jiang Yang, Lada A Adamic, and Yan Chen. Crowdsourcing with all-pay auctions: A field experiment on taskcn. *Management Science*, 60(8):2020–2037, 2014.

David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

Yina Lu, Andrés Musalem, Marcelo Olivares, and Ariel Schilkrut. Measuring the effect of queues on customer purchases. *Management Science*, 59(8):1743–1763, 2013.

Nicolai Meinshausen. Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1): 374–393, 2007.

Jürgen Mihm and Jochen Schlapp. Sourcing innovation: On feedback in contests. *Management Science*, 2018. Articles in Advance.

Paul R Milgrom and Robert J Weber. Distributional strategies for games with incomplete information. *Mathematics of Operations Research*, 10(4):619–632, 1985.

Benny Moldovanu and Aner Sela. The optimal allocation of prizes in contests. *American Economic Review*, pages 542–558, 2001.

Ken Moon, Kostas Bimpikis, and Haim Mendelson. Randomized markdowns and online monitoring. *Management Science*, page Forthcoming, 2016.

Oded Netzer, Ronen Feldman, Jacob Goldenberg, and Moshe Fresko. Mine your own business: Market-structure surveillance through text mining. *Marketing Science*, 31 (3):521–543, 2012.

G Pahl and W Beitz. *Engineering Design*. Springer Verlag, New York, 1988.

Ariel Pakes, Michael Ostrovsky, and Steven Berry. Simple estimators for the parameters of discrete dynamic games (with entry/exit examples). *The RAND Journal of Economics*, 38(2):373–399, 2007.

Yiangos Papanastasiou and Nicos Savva. Dynamic pricing in the presence of social learning and strategic consumers. *Management Science*, 63(4):919–939, 2017.

Raymond Paternoster, Robert Brame, Paul Mazerolle, and Alex Piquero. Using the correct statistical test for the equality of regression coefficients. *Criminology*, 36(4): 859–866, 1998.

Adrian E Raftery. Approximate bayes factors and accounting for model uncertainty in generalised linear models. *Biometrika*, 83(2):251–266, 1996.

Seth Richards-Shubik. Peer effects in sexual initiation: Separating demand and supply mechanisms. *Quantitative Economics*, 6(3):663–702, 2015.

Daniel M Ringel and Bernd Skiera. Visualizing asymmetric competition among more than 1,000 products using big search data. *Marketing Science*, 35(3):511–534, 2016.

Norbert FM Roozenburg and NG Cross. Models of the design process: integrating across the disciplines. *Design studies*, 12(4):215–220, 1991.

Adam J Rothman, Peter J Bickel, Elizaveta Levina, Ji Zhu, et al. Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515, 2008.

Adam J Rothman, Elizaveta Levina, and Ji Zhu. A new approach to cholesky-based covariance regularization in high dimensions. *Biometrika*, 97(3):539–550, 2010.

Francisco JR Ruiz, Susan Athey, and David M Blei. Shopper: A probabilistic model of consumer choice with substitutes and complements. *arXiv preprint arXiv:1711.03560*, 2017.

John Rust. Optimal replacement of gmc bus engines: An empirical model of harold zurcher. *Econometrica: Journal of the Econometric Society*, pages 999–1033, 1987.

Donald A Schön. *The reflective practitioner: How professionals think in action*, volume 5126. Basic books, 1984.

Donald A Schön. Designing: Rules, types and words. *Design studies*, 9(3):181–190, 1988.

Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

Adam N Smith and Greg M Allenby. Demand models with random partitions. *Journal of the American Statistical Association*, pages 1–35, 2019.

Adam N Smith, Peter E Rossi, and Greg M Allenby. Inference for product competition and separable demand. *Marketing Science*, 38(4):690–710, 2019.

Sorawoot Srisuma. Minimum distance estimators for dynamic games. *Quantitative Economics*, 4(3):549–583, 2013.

Andrew Sweeting. The strategic timing incentives of commercial radio stations: An empirical analysis using multiple equilibria. *The RAND Journal of Economics*, 40 (4):710–742, 2009.

Yuya Takahashi. Estimating a war of attrition: The case of the us movie theater industry. *American Economic Review*, 105(7):2204–41, 2015.

Christian Terwiesch and Yi Xu. Innovation contests, open innovation, and multiagent problem solving. *Management science*, 54(9):1529–1543, 2008.

Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

HH van den Kroonenberg. Cad applications in the creative phases of the methodical design process. In *CAPE'86*, pages 339–348, 1986.

Mengting Wan, Di Wang, Matt Goldman, Matt Taddy, Justin Rao, Jie Liu, Dimitrios Lymberopoulos, and Julian McAuley. Modeling consumer preferences and price sensitivities from large-scale grocery shopping transaction logs. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1103–1112, 2017.

Joel O Wooten and Karl T Ulrich. The impact of visibility in innovation tournaments: Evidence from field experiments. Working paper, 2017a.

Joel O Wooten and Karl T Ulrich. Idea generation and the role of feedback: Evidence from field experiments with innovation tournaments. *Production and Operations Management*, 26(1):80–99, 2017b.

Chiwei Yan and Jerry Kung. Robust aircraft routing. *Transportation Science*, 52(1): 118–133, 2018.

Jianxin Yin and Hongzhe Li. A sparse conditional gaussian graphical model for analysis of genetical genomics data. *The annals of applied statistics*, 5(4):2630, 2011.

Fanyin Zheng. Spatial competition and preemptive entry in the discount retail industry. Working paper, 2016.

Shuheng Zhou, Philipp Rütimann, Min Xu, and Peter Bühlmann. High-dimensional covariance estimation based on gaussian graphical models. *Journal of Machine Learning Research*, 12(Oct):2975–3026, 2011.

Yiyi Zhou. Bayesian estimation of a dynamic model of two-sided markets: application to the us video game industry. *Management Science*, 63(11):3874–3894, 2016.