Computational Approaches for Estimating Life Cycle Inventory Data

by

Jiarui Cai

A thesis submitted in partial fulfillment
Of the requirements for the degree of
Masters of Science
(Natural Resources and Environment)
in the University of Michigan

December 2016

Thesis Committee:

Assistant Professor Ming Xu, Chair
Research Fellow Dr. Sai Liang

Acknowledgements

After an intense learning period, it goes to the finishing touch on my thesis. Writing this thesis has had a big positive impact on me in my graduate education in the University of Michigan. I would like to reflect on the people who have supported and helped me so much throughout this period.

Foremost, I would like to express my sincere gratitude to my thesis advisor Prof. Ming Xu of the School of Natural Resources and Environment at the University of Michigan – Ann Arbor. Prof. Xu consistently provided his entire expertise and support to guide and encourage me in academic research. The thesis would not have been possible without Prof. Xu's help and enlightenment.

In addition, I would provide my special thanks to Dr. Sai Liang as the thesis co-advisor for providing me valuable comments and suggestions, and helping me move to progress. I would also acknowledge those who provided their thoughtful feedback towards this thesis including Prof. Gregory Keoleian, Prof. Shellie Miller of School of Natural Resources and Environment at University of Michigan – Ann Arbor, and my research partners Ping Hou and Syne Salem. It was also appreciated that the U.S National Science Foundation (NSF) supported this thesis under Grant No. 1554349.

My thanks also go to those who provided me support and companionship in daily life during this period including my colleagues Zeqi Zhu, Dr. Shen Qu, Helaine Hunscher, Maryam Arbabzadeh; my friends Dr. Zicheng Bi, Yuanyuan Yang, Yuqiao Liu, Ke Yang, Xiajun Zhang, Da Zhu, Hangjian Zhao, Eric Krawczyk, Emily Durand and my cat Mini.

Last but not least, I must express my very profound gratitude to my parents who provided me unfailing support, entire understanding and perpetual love. Hope they would forgive me for not accompanying with them during the years of my study. Hope they would be proud of their daughter not stopping studying and contributing.

Contents

Abstract

Data gaps in life cycle inventory (LCI) are stumbling blocks for investigating the life cycle performance and impact of emerging technologies. It can be tedious, expensive and time consuming for LCI practitioners to collect LCI data or to wait for experimental data become available. I propose a computational approach to estimate missing LCI data using link prediction techniques in network science. LCI data in Ecoinvent 3.1 is used to test the method.

The proposed approach is based on the similarities between different processes or environmental interventions in the LCI database. By comparing two processes' material inputs and emission outputs, I measure the similarity of these processes. I hypothesize that similar processes tend to have similar material inputs and emission outputs which are life cycle inventory data I want to estimate. In particular, I measure similarity using four metrics, including average difference, Pearson correlation coefficient, Euclidean distance, and SimRank with or without data normalization. I test these four metrics and normalization method for their performance of estimating missing LCI data.

The results show that processes in the same industrial classification have higher similarities, which validate the approach of measuring the similarity between unit processes. I remove a small set of data (from one data point to 50) for each process and then use the rest of LCI data as to train the model for estimating the removed data. It is found that approximately 80% of removed data can be successfully estimated with less than 10% errors. This study is the first attempt in the searching for an effective computational method for estimating missing LCI data. It is anticipated that this approach will significantly transform LCI compilation and LCA studies in future.

Chapter 1. Introduction

1.1 Life cycle assessment

Life Cycle Assessment (LCA) is an environmental management tool started from 1960s. It aims to establish an environmental profile of a product system by compiling and evaluating the inputs, outputs and the accompanying environmental impact of the system throughout its life cycle. The inputs include material, and energy. And the outputs often refer to byproduct and waste. Under a raising environmental concern in society, LCA is widely used by decision makers to assess the sustainability of a product, and become an important role in environmental management.

A life cycle for a product usually means the whole life period of the product start from raw material acquisition through materials processing, manufacture, distribution, use, repair and maintenance and disposal or recycling. LCA practitioner studies the life cycle of a product and produces a report of the environmental impact in different stages of this product. Traditionally, LCA people often talk about is conventional LCA. But there is also Economic Input-output LCA (EIO-LCA). EIO-LCA is more a method from a macro perspective. It concentrates on the environmental impact from economic transactions through the supply chain of an economic sector based on its financial input-output model. Conventional LCA and EIO-LCA are useful in different areas. This thesis only focuses on conventional LCA.

Generally, there are four phases in completing a conventional Life Cycle Assessment. **Goal and Scope Definition**: the scope of the assessment should be fixed after clarifying the purpose of the assessment, in order to keep the final result corresponding to expectation. **Inventory Analysis**: build the input-output list respect to the target system of the product or service that will be compiled. The input-output list concludes resources and energy input, and air, water, solid contaminant emissions. Inventory analysis including information collection and calculation, is designed for quantifying inputs and outputs of the product. **Impact Assessment:** based on the results from inventory analysis, analysts assess the potential environmental impact of the product. **Interpretation:** guiding by the

goal and scope of the assessment project, analysis results and suggestions are given based on inventory analysis and impact assessment.

LCA is widely implemented in policy making, academic researches and consultative analysis.

Curran has written a review of the effect of LCA in U.S. governmental policy making [1]. One example is that in 2007, California EPA had minimized the hazardous waste by implementing green chemical product based on a systematical life cycle assessment of multiple chemical products [2]. In academia, Center for Sustainable System (CSS) has been working on LCA researches for decades. In recent years, Gregory A. Keoleian et al applied LCA to compare sunlight-based and sugar-based algae for biodiesel [3], and use LCA model to analyze the difference of energy use and greenhouse gas emission between Plug-in and wireless charging for electric bus systems [4]. However, the first LCA study was conducted by a commercial company. In 1969, Coca-Cola conducted an assessment through out the whole life cycle of their products to compared the environmental performance of different beverage can materials, and finally choose the material with least environmental impact [5]. More recently, a LCA from Nestlé compared plastic pot and glass jar for baby food carrying and a small but significant environmental benefit of the plastic pot was observed since the plastic pot is much lighter than glass jar and have a better performance in reducing environmental burden at the phase of transportation [6].

As time goes by, LCA is not only used in environmental management, but also in sustainable system design, green building schemes and other areas[7] [8]. Steven Skerlos from University of Michigan used LCA to guide the manufacturing design. He provided four principles to evaluate the sustainability of a technology in product design before it is made. LCA was used to benchmark and address the challenges in sustainability design.

There are mature LCA software designed for increasing LCA practitioners. SimaPro and GaBi are two most widely used softwares among others. These two tools supported by strong database covering majority of industrial processes in many countries. There are many other softwares designed for specific industries or certain districts. Such as eFootprint is developed in China and mainly

designed for Chinese industries since it is based on the core database of Chinese Life Cycle Database (CLCD).

1.2 Life cycle inventory (LCI)

For a certain LCA project, data collection and validation are always the most important phase in its whole process. The ensemble of all environmental-related values of a product's life cycle is called Life Cycle Inventory (LCI). Inventory analysis characterizes the life cycle of a product as many interconnected unit processes, and then quantifies environmental interventions (i.e., materials/energy used and emission/waste generated) associated with each unit process. LCI is the "pillar stone" for further impact assessment and interpretation in LCA.

Generally, there are many significant sources of raw data to generate LCI database. For many instances, people can collect data from meter readings from equipment, industry data reports, laboratory test results, government documents, publicly available databases, journal papers, books, associations etc.. Once raw data are collected, people need to transform them to unit process input output information. LCA analysts usually build a mathematical model (Sometime this model are made in software) to realize this transformation and establish the product life cycle inventory according to defined functional unit, reference flow and emission factors.

Nowadays, well-developed LCI database such as Ecoinvent (www.ecoinvent.org) is the prevalent source for LCI construction for LCA projects. Among serial types of database, Ecoinvent is the most comprehensive and international product. It not only collects data from a wide-range of geographies and industries, but also encourages users provide operation data to update the database. There are also other LCI databases like US LCI, ELCD, LCAfood. Either of them is limited in a specific district or for a specific industry.

Data collection and validation in constructing a LCI database is always time-consuming and labor intensive. It requires large investment in time, human resources and capital while site monitoring or experimental simulation may be needed for data collection. Even though the dedicated LCI database can be helpful for most of industrial products, many advanced technology and emerging

industrial processes, such as renewable energy, still ask for field exploration and site testing. Transforming the traditional time-consuming, expensive practice of LCI database development into a faster and cheaper process becomes more and more necessary.

1.3 Network science

Network Science is an interdisciplinary academic field which studies on the properties of different types of networks including technology network, social network, information network, biology network etc.. For instance, telephone network, friendships between high school teenagers, the World Wide Web and food chain are all typical networks in real world. Network science was developed in disciplines as diverse as mathematics, statistics, physics, social science, information science and computer science [9]. Technically, there are nodes connected by edges in a given network. If there's probability or intensity assigned to edges between nodes, it's called weighted network. Most networks we know are multipartite graph which means there are many participants or nodes in the network and they connect to other random node(s) in the network. There are also bipartite networks such as users-products network, in which there are only two types of nodes (users, products) in this network, and each node in one category will only connect with node(s) in the other category. LCI data is a bipartite weighted network, which will be carefully illustrated in the following chapters.

Link Prediction is one of the sub-fields of Network Science. The studies of Link prediction have a remarkable increasing in recent years. Link prediction is a method to estimate the likelihood of the existence of a link between two nodes based on observed links and the attributes of nodes [10].

Link prediction is mainly applied in three areas. First, it could find the missing links in a network. Secondly, by the same principle, it could detect the perturbance in a database, that is the false data collected by mistakes. Finally, link prediction could be used in predict future development of a dynamic database by predicting new data in it.

All three applications are relevant to LCI database development (Table 1). Firstly, uncompleted LCI database is often a defect in LCA process. Link

prediction can be used as a way to estimate the missing data in LCI database. Secondly, the uncertainty in data information in LCA is widely discussed among LCA practitioners, thus link prediction can move in to detect the accuracy of current data by divide the database into training section and validation section and use data in training section to estimate data in validation section. The error between original data and estimated data can reveal the distrustful data points. Lastly, there always are emerging technologies that cannot find available data in current LCI database. Link prediction provided a pathway to predict environmental interventions of these newly generated industrial process.

Table 1 Link prediction applications, examples, and relevance to LCI database development

Application	Examples	Potential application to LCI database development
Predicting missing links	Scientific collaboration networks [11]; Protein interaction networks [12]	Estimating missing data to complete the LCI database
Identifying false links	Peer advice networks in companies [13]; Protein interaction networks [14]	Evaluating the quality of data points; Guiding directions for future development
Predicting future links	Internet [15], [16]; Online user-object networks [17]	Estimating LCI data for emerging technologies

1.4 Structure of the thesis

Inspiring by the willing to estimate LCI development and the mature discipline of network science, I conduct this research to explore the possibility of computational approach to estimate LCI missing data based on link prediction. The following parts are organized in 6 sections: 1) Literature Review in Chapter 2 gives a broad overview of the applications of link prediction in different areas. Most papers provide creative standpoints of link prediction application in certain studies, which give us the significance in guiding the research. 2) Chapter 3 talks about material, basic model and research structure. It describes the database used in this research in the aspects of developing history, components and data structure of the target LCI inventory database. It lists the fundamental prediction model to estimate missing LCI data. In addition, this chapter also draws the research framework. 3) Starting from Chapter 4 to Chapter 6, three research experiments are elaborated. In chapter 4, it investigates the accuracy of the method by finding out the minimum estimation error that can be approached. The corresponding size of training data to find out the minimum error is also provided to describe the efficiency of the method. 4) Chapter 5 discusses other 4

algorithms to replace the algorithm used in basic method provided in Chapter 4. A normalization method is also provided. The same accuracy and efficiency testing are conducted to test the performance of these 4 algorithms and normalization method. 5) Chapter 6 moves to finalize the prediction model by defining the size of training data that can have the model to meet its best performance. 6) In Chapter 7, I extend this research to a discussion on the future development and application of this method. Furthermore, opinions on future research of computational approach of LCI data are provided.

Chapter 2. Literature Review

The data deficiency and unreliability were widely discussed in LCI research papers. In some LCA projects, data gap and data uncertainty even were main barrier to successful decision making. A lot of solutions were provided to fill this gas in academia and application areas, and each one of them had their own application limitations.

This chapter overviews academic literatures in LCA data gap and applications of link prediction. By learning from published research works, it's easier for readers to understand methods in this research and it inspires future thoughts on this topic.

2.1 Data gaps in life cycle assessment

LCA practitioners largely rely on dedicated LCI database when completing a LCA project [18]. Therefore, data availability is essentially crucial for project success. However, in practice, dedicated database may not be sufficient enough for some LCA projects. For instance, dedicated database usually uses LCI data in one district to estimate LCI data in another district which introduces uncertainty to data quality. In addition, there always are emerging technologies that never have been tested for environmental information. Therefore dedicated LCI database often doesn't have the perfect data for unusual certain processes.

Anna Bjorklund elaborated sources of uncertainty in life cycle assessment. Data inaccuracy, data gaps are two major sources of uncertainty that affect LCA quality [19]. Beverly Sauer discussed data missing problem in detail in the book Life Cycle Assessment Handbook: A Guide for Environmentally Sustainable Products. He mentioned that water use is barely collected over the years. However, water use is essential and important to understand many processes but was missed in first years to develop LCI database. The same situation happens to carbon dioxide emission which is only recognized as important information when global warming potential become recognized as an important environmental issue [20]. In future LCA work, there probably will be other

parameters that were not included in current database. This type of data missing is critical to the success of many researches.

Many scholars proposed their thoughts on dealing with data issues in LCA projects. Shelie Miller and Gregory Keoleian have pointed out data challenge in analyzing transformative technology (emerging technology that has the potential to change existing situation in many aspects). They gave the idea that transformative technology lacks of data and has inherent uncertainties regarding to their development. They proposed a framework to manage data quality in transformative technologies LCA by analyzing 10 factors that affect uncertainty in these LCA projects [21].

Anna Bjorklund's survey paper gave readers an outlook about how people deal with LCA uncertainty. For example requiring LCA practitioner to assign Data quality goals (DQG) and Data quality indicators (DOI) along with LCA project. It also pointed out some reasonable methods to estimate missing parameters such as deriving missing data on mass balance in material flows, using data from similar technologies or average industry data [19].

Sangwon Suh and Gjalt Huppes proposed a missing data estimation method called Missing Inventory Estimation Tool (MIET) in 2002. They introduced Input-Output Analysis to LCA by transforming IOA information into LCI data with specific metrics. Although this method has been further developed after publish of this paper, authors have pointed out that the shortcomings of IOA would affect this method. For example, data in IOA has a high level of aggregation and data uncertainty, and it would be transferred to the estimation of LCI data [22]. This method combining IO table to fill in the gaps of LCI is first mentioned by Treloar, G.J at 1997 [23], and Robert H. Crawford did a research in 2008 to analyze the development of it and called it as hybrid LCI analysis method [24].

Scholars never stop finding new methods to deal with data gaps and uncertainty issues. Some methods such as mass balance study require a lot of information that makes the process even more complicated and not so cheap. Other methods like hybrid LCI analysis method do not always provide accurate estimation, and sometimes increase data uncertainty. Transforming the traditional

time-consuming, expensive practice of LCI database development into a faster, more accurate and less expensive process has been called for.

2.2 Link prediction

2.2.1 Network science

Network Science is an interdisciplinary field learning about structure, development and weaknesses of information webs. A network is consisted of nodes and internal links between nodes. There are different networks such as friendship network, world-wide-web and food chain. Network science was developed in disciplines as diverse as mathematics, statistics, physics, social network analysis, information science and computer science [9].

Two books are recommended as fundamental interpretation of network science. *Networks: An introduction* composed by Newman is a thorough work on introducing network theory. It categorized four main types of network – technological network, social network, network of information and biological network. In addition, it expounded the mathematics fundamentals, computer algorithms, models of networks and analyze processes of networks[25]. The other book names *Network Science* cooperatively created by Albert-Lasszlo Barabasi et al. is published in August 2016. It introduces network science from graph theory and emphasize on properties of networks.

Network theory was greatly developed in many disciplines since there are many kinds of network in real world and they exist in many different fields. A lot of network theory application can be found in academia. James McNerney et al. revealed the relationship between inter-industry flows by learning about money flows between industries in 45 national economies [26]. Sai Liang et al. used network theory to identify important sectors and communities that are responsible for carbon dioxide emission [27]. Friendship network analysis is a good and fun example of taking advantage of networks science in social researches. K. Lewis et al. revealed factors that affect online relationships by using Facebook network database [28]. In addition, network theory is also widely used in computer science, biology and chemistry.

2.2.2 Link prediction

Link Prediction is a branch of Network Science. It is a technique to estimate the likelihood of the existence of a link between two nodes, based on observed links and the attributes of nodes [10].

Linyuan Lv et al. wrote an review paper elaborated link prediction and three types of LP techniques in detail [29]. This paper classified link prediction techniques into three groups. They are Similarity-based Algorithm, Maximum Likelihood Methods and Probabilistic Model. Similarity-based Algorithm is the simplest framework of link prediction methods. It aims to score links between two nodes by evaluating their similarities according to the attributes of the nodes. Maximum Likelihood Method is used on the organization principles of the structure of the network. Probabilistic Models, instead of directly analyze network structure like the Maximum Likelihood, it tries to use existent information to abstract the underlying structure, and then predict the missing links by extended new model.

Link prediction are also used in many research fields such as anomalous email detection [30], terrorists identification [31] and protein function prediction [12].

2.2.3 Link prediction in LCI database development

There are weighted network and unweighted network in terms of link types. There are also multipartite and bipartite network in terms of relationship structure. LCI database is often regarded as a bipartite weighted network while the two elements are industrial unit processes and environmental interventions. In addition, the weighted link between unit process and environmental intervention is the corresponding value of environmental intervention from this unit processes. The LCI networks allow studying structural features of complex systems in distinct natures. Link prediction in network science offers potentially transformative capabilities for developing LCI databases with only limited observed data without relying on time-consuming, expensive empirical data collection.

Link prediction was texted well performed in multipartite unweighted networks such as Facebook friends recommendation system. However, network techniques like Link Prediction were still in a fast but still developing state in more complicated networks especially bipartite weighted networks such as LCI.

Increasing demand of techniques to do bipartite network prediction attracts researchers' attentions.

Jérôme Kunegis et al. studied deeply on bipartite network in their 2010 published paper. They defined bipartite networks and specialized general algorithms for this types of network [32]. Oussama Allali, et al. proposed a method to transform bipartite network into conventional multipartite graphs. They defined a concept as internal links and project these internal links into a multipartite networks by weighted method [33].

In terms of weighted network, Cai Gao et al. studied the methodology to identify the influential nodes in weighted network, they proposed parameters of Evidential Semi-local Centrality (ESC) and Existing Evidential Centrality (EVC). These two parameters were used to generate new degree of nodes and further model the uncertainty of new links. This method is applied in weighted networks [34].

Collaborative Filtering is a similarity-based link prediction algorithm designed for system recommendations. It was well known in recommending movies for users in Netflix [35] [36]. Netflix's user – movie database is exactly a bipartite weighted network since every user gives rating scores to movies. John Breese et al. had a profound research on collaborative filtering. They regarded weight *ij* as the probability that user *i* would purchase products *j*, and then gave three types of algorithms to estimate unknown weights between nodes by collaborative filtering. The first one is Memory-based Algorithms, it used user's consumption database to calculate relevant parameter for this user, and then took advantage of the parameter to predict the user's future purchase. The second one is extension to Memory-based Algorithm. It modified standard algorithm to improve the performance according to intuitive database perceive. The third one is Model-based Methods which created a probability model that assumes active user will have a particular vote value for a certain item given the previously observed votes [37].

Jing Zhao et al. gave us a way to calculate the weighted similarity index by adding a parameter called S_z in similarity-based algorithm. S_z is the strength of node z (the sum of weights of its associated link). Jing Zhao et al. claimed that this method has a good performance on weighted networks [38].

Inspiring by link prediction theory, more complicated methods were proposed for network prediction. Linyuan Lv et al. have proposed a matrix-algebra-based examination method to define the predictability of a complex network. Along with the examination method, they also gave out an algorithm to estimate missing value in a predictable network by calculating a new parameter related to the eigenvalue and eigenvector of its adjacent matrix [39].

SimRank is another newly developed method used in complex networks. It measures similarity of the structural context in object-to-object relationships based on their relationship to other objects. The basic assumption under SimRank is that two objects are similar to each other if they have similar neighbors. In addition, SimRank can be used combining with relational structural-context similarity for an overall similarity measure. Glen Jeh and Jennifer Widom from Stanford University have firstly introduced the basic concepts and basic algorithms of SimRank in their paper [40].

Efforts have already been made to estimate LCI data instead of collecting empirical data. In Wenert's research, based on neural networks, his team proposed a Molecular-Structure-based Models (MSMs) which use molecular features as input to estimate LCI data for certain chemical products[41][42]. However, the procedure still relies on a large amount of information such as reaction equations and process characteristics, which also requires a lot of prework such as chemical studies. In addition, this method is limited to chemical processes and relying on extensive chemical domain knowledge.

Chapter 3. Method and Materials

Beginning from this chapter, I elaborate the research map in a logical manner. The basic prediction method is introduced following the description of research database. After that, the paper structure is laid out. Readers can follow the structure provided in this chapter to find and dig into the information they need.

3.1 Database - Ecoinvent 3.1

I use Ecoinvent 3.1 as the target LCI database in this research. Ecoinvent is one of the biggest LCI databases in world LCA practice. It is also utilized in leading LCA softwares such as SimaPro, Gabi and Umberto. Ecoinvent was started off as a joint initiative of the ETH Domain and Swiss Federal Offices. It has been developed for 20 years dedicating in LCA methodology and LCI data compilation for different industrial sectors. While there are other LCI databases such as U.S. Life Cycle Inventory Database, European Life Cycle Data and LCA Food, Ecoinvent is relatively the most consistent and integrated database. People are supposed to purchase license to use Ecoinvent, and all industrial organizations are encouraged to contribute their own data to Ecoinvent.

There are default model, cut-off model and consequential model in Ecoinvent database. These three system models use the same data of real-world processes, but they use different methods to deal with the co-products and materials need to be treated. I simply choose the default model for analysis. Ecoinvent provides unit process dataset (UPR), cumulative inventories (LCI in Ecoinvent, it's called CI in the following parts) and life cycle impact assessment results (LCIA). UPR records the data of energy/material input, resource inputs and emission outputs of every process. CI is the aggregated data with all the upstream energy/material inputs of UPR converted into the resource inputs and emission outputs. LCIA is the impact category resulting from converting resources and emissions into environmental impact parameters.

CI database is the only material used in this research paper. UPR data is also analyzed by research group members, which will be discussed in future deliverables. Ecoinvent 3.1 CI database is basically a 1869 by 11332 matrix. Each

11332 column represents one industrial process and each 1869 row denotes value of environmental interventions such as material, energy inputs and emissions. Since CI database aggregates all the upstream material and energy consumption, the CI matrix is a dense matrix with most entries filled by numbers. One thing we should notice is that there are some negative values in CI database, they represent that the treatment activity is supplying the service of removing a certain mass of the reference product for treatment (either final disposal or recycling). The reference product is physically an input to the treatment activity, but since the reference product is placed on the output side, it needs the negative sign to maintain the mass balance of the activity [43].

Figure 1 clearly descripts the format of LCI database. Each element of the matrix indicates the amount of a particular type of resource or waste flow (row) associated with the unitary output of a particular unit process (column), e.g., 2 g CO2 emissions per 1 kg cement. An LCI database can also be represented as a network, using the LCI matrix as the adjacency matrix (Figure 1). In particular, there are two types of nodes (or vertices) in an LCI network respectively representing unit processes and environmental interventions. Unit process nodes are connected with environmental intervention nodes by links (or edges) indicating how much and what type of environmental interventions each unit process is associated with. This network is a weighted, bipartite network [44], as its links have strengths (the amount of environmental interventions) and nodes are divided into two disjoint sets.

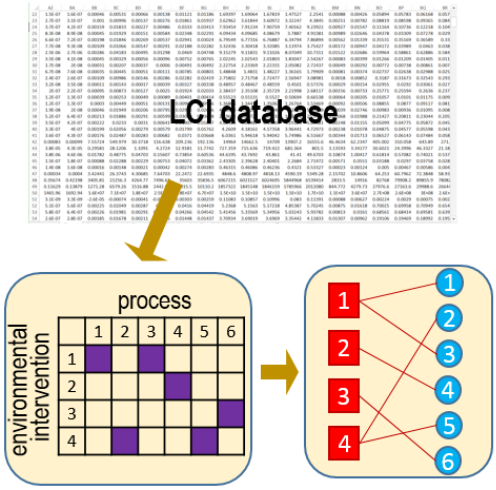


Figure 1. Data structure of LCI database

3.2 Data processing

In order to avoid interferences in the database, I remove empty columns and rows in CI database. These empty columns indicate that these processes does not have any information in CI database, they are listed there on purpose for future development. The same idea goes to empty rows. In addition, only one process is kept while the rest are removed among a set of processes that have identical or similar values in each environmental flows. Here similar is defined as the element-wise difference between the two processes are less than 10% of the maximum value of these two.

After data process, the new matrix of CI database has a dimension of 1799 by 8100. Finally, reformatted and processed Excel spreadsheets are imported into MATLAB as adjacency matrices to represent Ecoinvent 3.1 Cumulative Inventory network.

3.3 Similarity-based link prediction

Similarity-based link prediction is one of the most common methods used in network analysis [29] and has been taken as the basic prediction method in this research. Similarity-based link prediction methods first measure the similarity (or proximity) between each pair of nodes in the network. For bipartite networks, similarity is measured for the same type of nodes. Two nodes

that are similar tend to have similar patterns of linkages with other nodes in the network. Based on appropriate measures of similarity, one can then evaluate the likelihood of unknown links that exist for a node by comparing it with other similar nodes.

Note that in LCI networks, predicting the existence of links between processes is not enough. One also needs to predict the strength of particular links. This is different from simply applying existing link prediction methods in network science which are mostly developed for unweighted networks; however, the same principles still apply. Introducing link and/or node weights can generally modify models developed for unweighted networks for applying to weighted networks [45][46]. That's the philosophy of the prediction method for this bipartite weighted network.

In this research, the prediction process is divided into three steps: Calculate the similarity between processes, use the similarity to estimate missing data and test the estimation by examining its accuracy. Figure 2 shows the logic of prediction process.

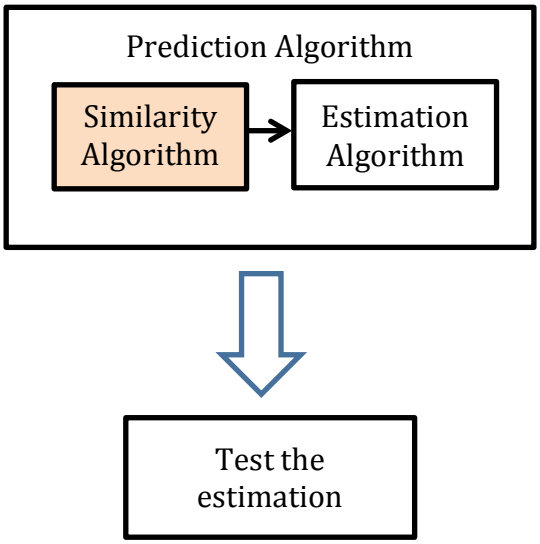


Figure 2.Graph of prediction process.

3.3.1 Similarity between the processes

The similarity of two unit processes s_{ij} is simply defined as the mean of normalized differences of all environmental interventions of the two processes at the first step (Equation 1):

$$s_{ij} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{|a_{ki} - a_{kj}|}{\max(|a_{ki}|, |a_{kj}|)} \quad (1)$$

where k is the index of environmental interventions, n is the number of environmental interventions, and a_{ki} is the environmental intervention k for unit process i. I named this similarity calculation method as **Average Difference**. I have also tried other methods typically used to measure similarity, such as Euclidean Distance, Pearson Correlation Coefficient, and SimRank [40], they will be discussed in Chapter 5.

3.3.2 Estimation for one missing data entry

For each missing data point in a certain process, the remaining data in this process are considered as known observations. They are used as the training set to compute similarities between this process and all other processes using Equation 1. The missing data point is then estimated by

$$e_{ki} = \frac{\sum_{j=1}^{t} a_{kq_j} s_{iq_j}}{\sum_{j=1}^{t} s_{iq_j}} \quad (2)$$

where t ($1 \le t \le m$-1) represents the number of most similar processes used to estimate the missing data point, and m indicates the total number of unit processes, I define t as the size of the training data. q_j is the column number of the jth similar process with process i. For each missing data point, there are m-1 different estimations with t ranging between 1 and m-1. Too less training data is not adequate enough to take advantage of similar processes, but too much training data may bring noise into the prediction. Both issues lead to inaccurate prediction that people want to avoid.

3.3.3 Performance of the estimation

I define the accuracy to be the distance (error) between the estimation and the "true" value from the original LCI database, which are calculated by Equation 3.

$$r_{ki} = \frac{|e_{ki} - a_{ki}|}{a_{ki}} \quad (3)$$

with m-1 different estimations of e_{ki}, I get m-1 different values of r_{ki}. I define the according t with the minimum r_{ki} as the size of training data corresponding to the minimum error.

The distribution of the errors indicates the accuracy of the estimation. The size of training data of the minimum error is the same meaning of the number of most similar processes that should be used to have the most accurate estimation for a single missing data. The size of training data of the minimum error indicates the computational efficiency, the smaller size of training data used to estimate, the more efficiency the calculation would be.

3.4 Research structure

All of the researches are built on an intact CI matrix from Ecoinvent 3.1 CI database. According to the situation, I divide this database into validation dataset and training dataset. I regard validation data as missing data, use training data to estimate missing data, and use validation data to test the accuracy of the estimation (Figure 3).

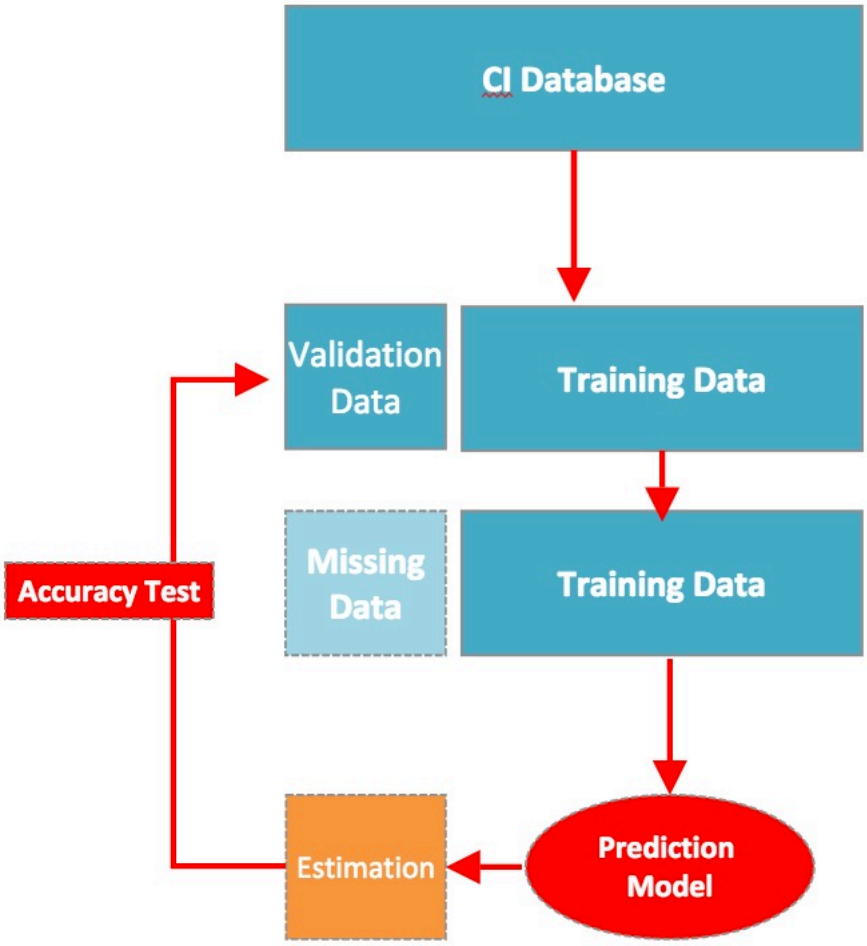


Figure 3 Graph of prediction steps

At the first phase, I set the size of validation dataset as 1 entry and 50 entries respectively to test the accuracy and efficiency of the proposed prediction method. In phase 2, instead of the similarity algorithm of Average Difference, I propose other three algorithms and a normalization method to obtain similarity matrix. Setting the validation dataset as 1 entry, I test the accuracy and efficiency from these new similarity approaches. Before finalizing the prediction method, I find that in estimation step, I don't need all training dataset to acquire a most accurate prediction, so in phase 3, I test the prediction accuracy by using different size of training data. The prediction method is finalized by using the amount of training data which can result in most accurate prediction. The research structure is listed in Figure 4**Error! Reference source not found.** Phase 1, 2 and 3 are elaborated in chapter 4, 5 and 6.

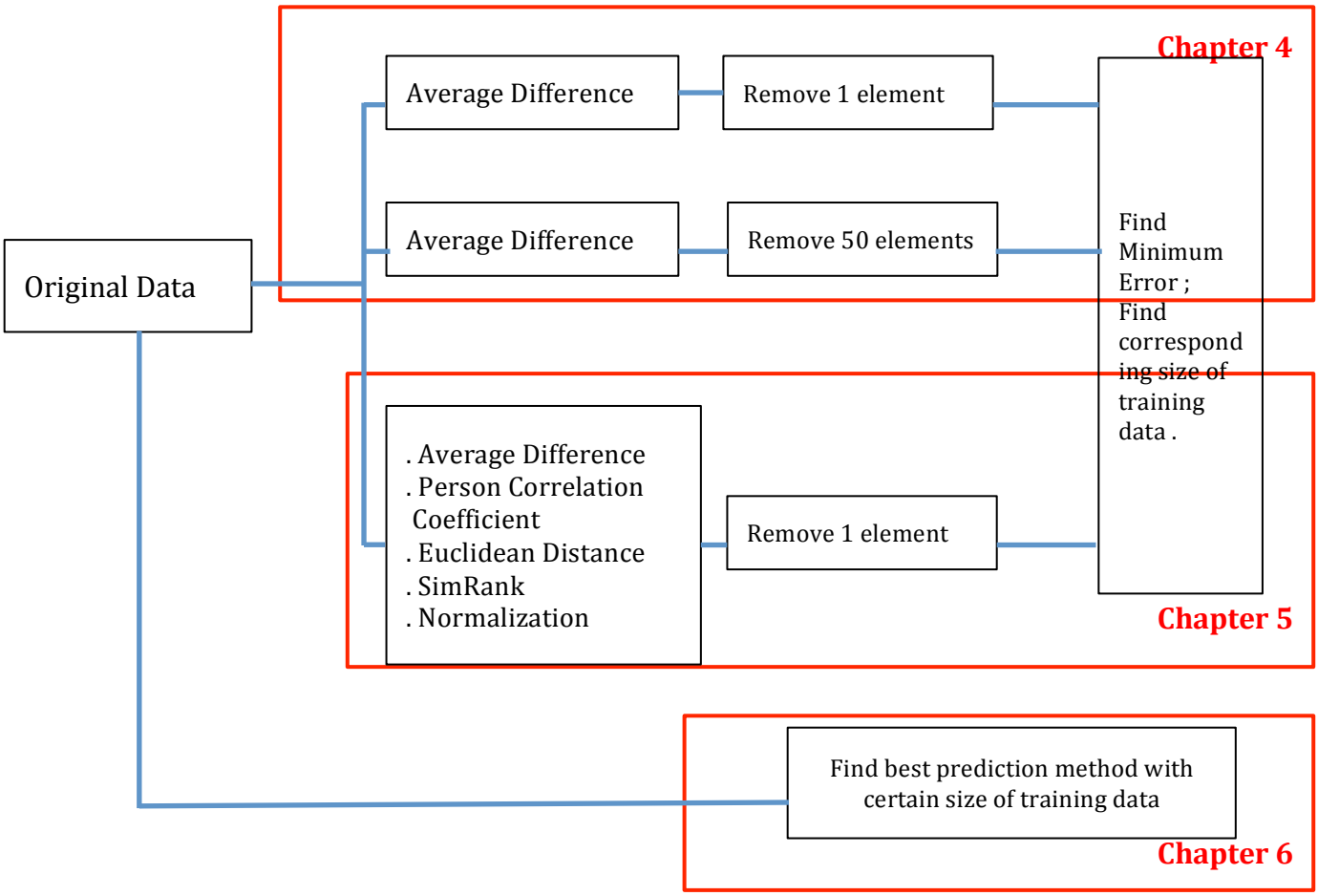


Figure 4 Research Structure

Chapter 4 Accuracy and Efficiency of Estimating 1 and a Block of 50 Missing Data Entries

I proposed the basic prediction method in section 3.3. In this chapter, I test the accuracy and efficiency of the prediction model by using it to estimate missing data entry/entries and calculate prediction errors.

I start the accuracy and efficiency test from an easy way that just remove one data entry, estimate it and test the accuracy and efficiency. If the estimation is accurate and efficient, I have the motivation to remove a block of 50 data entries and try to predict them in the same way. If the method still works well, it proves the validation of this approach.

4.1 Methods

4.1.1 Estimate one missing data entry

I remove each one data entry in CI matrix (1799 by 8100) one by one. Use the rest values of the matrix as training data to estimate the missing value. In every iteration, the removed data entry will be taken as both missing data and validation data. When one certain data entry removed, say, data entry in column A and row B, obtain the similarity between column A and all other columns by using equation (1) with data in all rows except row B. Using equation (2), with different size of training data, it will gives out m-1 estimations. Find out m-1 estimation errors, pick out the smallest one and record it as **Minimum Error**, the corresponding amount of training data used will also be recorded as **Size of Training Data**. In this way, result will be two 1799 by 8100 matrixes. One records minimum error from each iteration, and the other records corresponding size of training data.

4.1.2 Estimate a block of 50 missing data entries

I repeat this process when a block of 50 data entries are moved as missing data each time, with a revising that I intended to find out the minimum average error of 50 estimations as well as the corresponding size of training data. Unlike

1 data entry removing experiment, the matrix of minimum average error and corresponding size of training data are 36 by 8100 matrices in this experiment (1799 divided by 50 rounds up to 36).

4.1.3 Approximation of similarity matrix

One thing I should clarify is that theoretically, for each missing data, I should use the remaining data to calculate a specific similarity matrix by Equation (1), and then plug this group of similarity in Equation (2) to estimate this missing data. However, this process would be tedious and clumsy in calculation. The comparison between the similarity matrix of the original entire matrix and the matrix with one data missing shows neglectable difference. So I use the similarity of the original entire matrix as approximation in all the one missing data estimation, which helps save time to recalculate similarity matrix each time. However, the difference of the similarity matrix with a block of data missing and the original entire matrix cannot be neglected, especially when a large group of data is missing. Therefore similarity matrix should be recalculated each time when a large group of data is missing each time. Proven are discussed as following.

Name the similarity matrix calculated by entire original matrix as matrix A, and the similarity matrixes calculated by the rest of values when 1 data entry was removed each time as matrix Bs. 1799 similarity matrices Bs are calculated by removing one of 1799 rows each time. Then the average value of 1799 difference matrix Cs which equals to A-Bs are recorded. Mean value of these 1799 average value is 0.0004. It is very small and neglectable. Therefore, I claim that I can use the similarity matrix calculated by entire original LCI database as the approximation of similarity matrix in section 4.1.1.

4.2 Results

Before discussing experiment results from 4.1.1 and 4.1.2, I sketched a heat map of the process similarity matrix acquired from original entire database, which is a 8100 by 8100 square matrix.

Then I drew the empirical cumulative distribution graph of minimum error matrix and size of training data matrix. The distribution of the errors indicates the accuracy of the estimation. The distribution of size of training data

indicates the computational efficiency. The less processes used to get the best estimate, the more efficiency the calculation would be.

4.2.1 Similarity of processes

The similarity-based link prediction method is built on the assumption that similar processes have similar environmental interventions. The heat map (Figure 5) confirms the assumption that when the process in both rows and columns are ranked by International Standard Industrial Classification of All Economic Activities, Rev.4 (ISIC) identification number from 01 to 99.

The diagonal of the heat map is a white line, because the similarity between a process and itself is always 1. In addition, there are light squares around the diagonal, which indicates that processes in the same category generally have higher similarity.

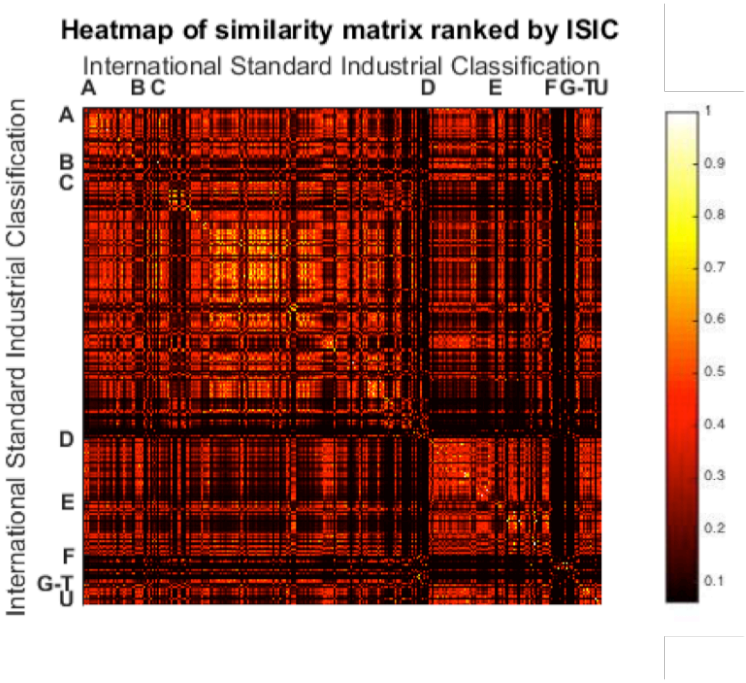


Figure 5 Heatmap of similarity of CI database

4.2.2 Results of estimating 1 missing data entry

4.2.2.1 Minimum errors

For CI database, Figure 6 shows that the minimum error the method could capture are distributed from very small (less than 10^{-10}) to very large (more than 10^{10}). But most of minimum errors are among 10^{-5} to 1 and nearly 89%

of minimum errors are less than 0.1. In addition, 4% minimum errors equal to 0, which means 4% of values could be precisely estimated.

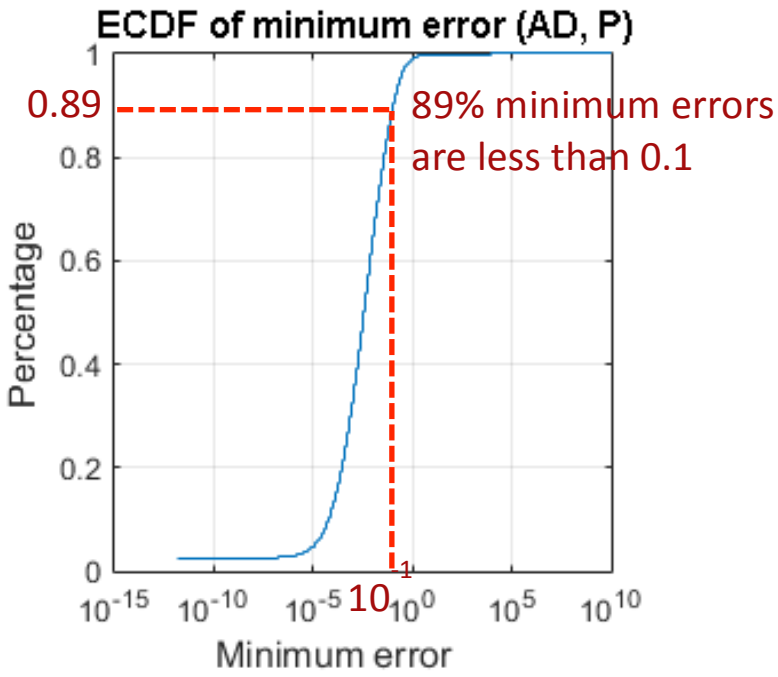


Figure 6 Distribution of minimum error (with one data missing)

4.2.2.2 Corresponding size of training data

Figure 7 shows the cumulative distribution of corresponding size of training data, from which I can tell that nearly 24% minimum error are hit by only using value from the top 1 most similar process as training dataset. In addition, around 84% minimum errors can be found out with only less than 100 similar processes in the training dataset. The calculation is relatively efficient.

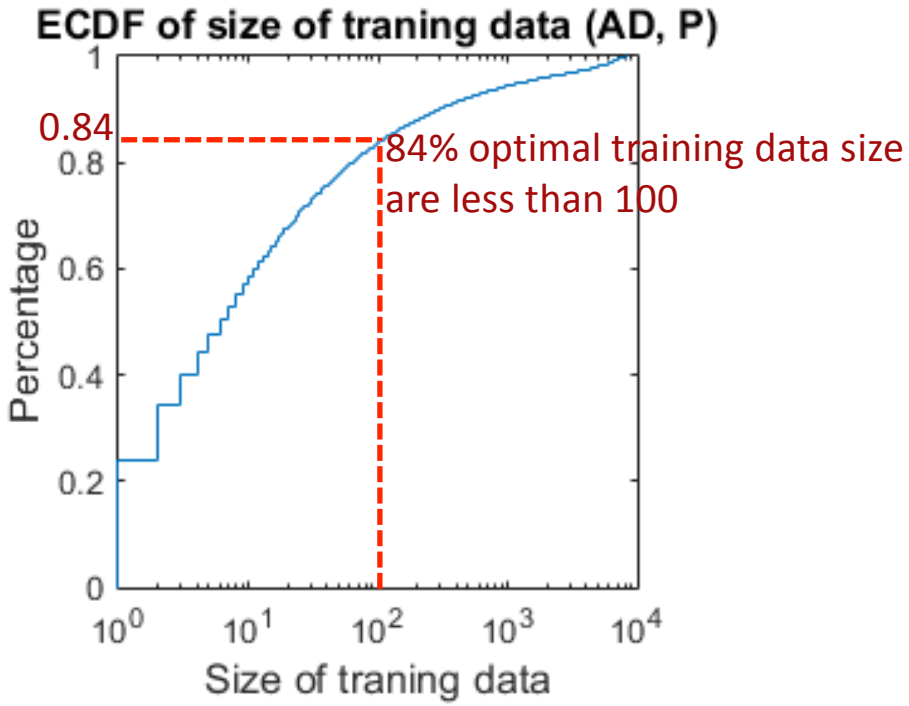


Figure 7 Distribution of size of training data (with one data missing)

4.2.3 Results of estimating a block of 50 missing data entries

4.2.3.1 Minimum error

When 50 data simultaneously miss in LCI database, the results are displayed in Figure 8 and Figure 9. Figure 8 shows the empirical cumulative distribution of average minimum error when a block of 50 data entries removed each time. It's obvious that the value of minimum error is mainly greater than the values of minimum error when only 1 data entry removed each time. In this situation, most of minimum errors are between 10^{-3} to 1 and the proportion of value less than 0.1 decreases to 58%.

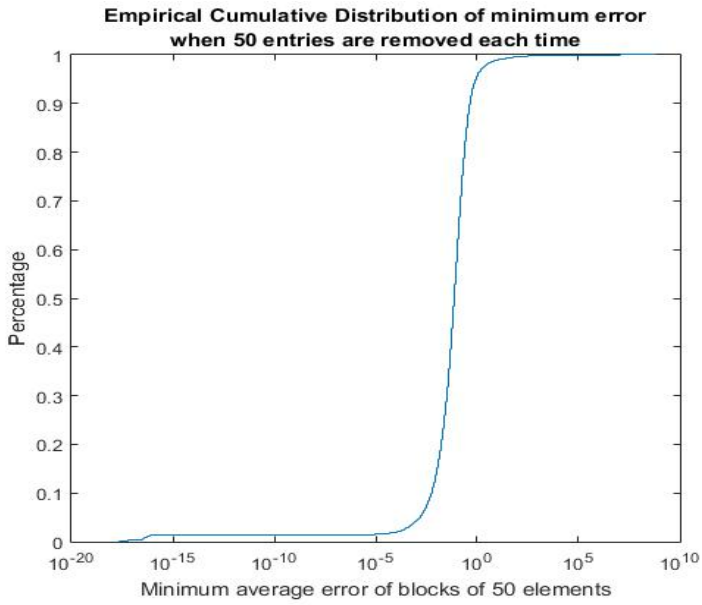


Figure 8. Distribution of minimum error (with a block of 50 data missing)

4.2.3.2 Corresponding size of training data

Figure 9 shows that nearly 48% average minimum errors are found by only using the most similar process as training data (size = 1). More than 99% average minimum error can be found with no more than 100 most similar processes.

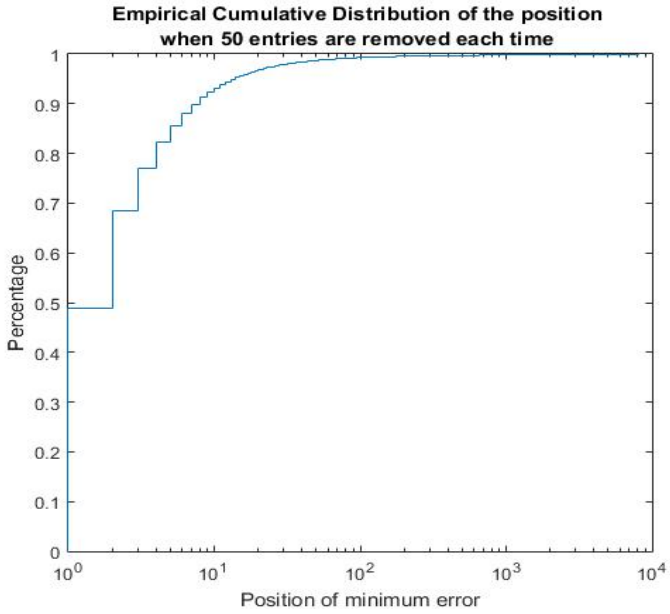


Figure 9. Distribution of the size of training data (with a block of 50 data missing)

Results of these two experiments show the promising potential of the proposed prediction model to estimate missing LCI data. For most of estimations, not only could the prediction method results in very small minimum error, but also could it meet less calculation complexity by only using small size of training data. In addition to a good performance of estimating 1 missing data, this prediction method doesn't collapse when estimate multiply missing data.

This prediction model needs to be finalized as a completed system to predict by defining optimal size of training data. This will be discussed in Chapter 6.

Chapter 5 Other Methods to Measure Similarity

As I mentioned in chapter 3, in this chapter, I propose another three algorithms to capture similarity matrix. In addition, I provide the method to do data normalization before similarity calculation. I also try to calculate similarities not only between industrial unit processes, but also between environmental interventions. I mean to find the best combination of methods in the prediction model that can give us the best estimation of CI missing data.

5.1 Methods

5.1.1 Additional methods to measure similarity

Despite Average Distance which is used in Chapter 4 to calculate similarity, three other algorithms are proposed to capture the similarity between processes. In this chapter, in spite of using new similarity matrix, other sections of the experiment in this chapter are the same with 4.1.1. Minimum errors and the corresponding size of training data are recorded as test parameters for accuracy and efficiency analysis.

In addition to only calculate similarities between processes, in this chapter, I also calculate similarities between environmental interventions for estimation. The prediction model is similar. After calculating the similarity matrix of environmental interventions, use estimation equation (2) to estimate missing data by using all environmental intervention values in the same process that the missing data belongs to. Estimation error, minimum error and corresponding size of training data are also recorded for comparing.

These three new algorithms are described below and their mathematic functions are listed in Table 2.

Pearson Correlation Coefficient

Pearson Correlation Coefficient is a statistical measure reveling the linear relationship between two variables. The equation for calculating Pearson Correlation Coefficient between variable x and y is:

$$r_{xy} = \frac{\sum xy - \frac{\sum x \sum y}{N}}{\sqrt{(\sum x^2 - \frac{(\sum x)^2}{N})(\sum y^2 - \frac{(\sum y)^2}{N})}} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \cdot \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

rxy is a value between -1 to 1, with 1 indicates total positive correlation, 0 is no correlation and -1 is total negative correlation. Pearson Correlation Coefficient measures the linear dependence between two variables.

Pearson Correlation Coefficient could be an algorithm to describe the relationship between two processes in LCI database. It could be developed in the research by simply assigned $S_{PJ}=r_{PJ}$ where P and J are two processes.

Euclidean Distance

Euclidean distance is a mathematic concept that widely used in describing ordinary distance between two points in Euclidean space. Mathematically, Euclidean distance calculates the summation of square difference between two points. For example, in a N dimension space, the Euclidean distance between point *A [A1, A2, …, An]'* and *B [B1, B2, …, Bn]'* is:

$$D_{ab} = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2}$$

Larger distance suggests a smaller similarity. Therefore I take the negative value of distance to adjust the function. And by using exponential function, the value of this function falls into an interval between 0 to 1. Processes with smaller distance will be assigned a higher similarity score. The finalized similarity calculation equation is:

$$S_{ij} = e^{\frac{-\sqrt{\sum_{k=1}^{n}(a_{ki}-a_{kj})^2}}{\sigma}}$$

Where:

a_{ki} is the value of environmental flow k in process i;

a_{kj} is the value of environmental flow k in process j;

σ is an adjustment parameter to make similarity results span well from 0 to 1.

If the two processes are identical, the summation of $\sqrt{\sum_{k=1}^{n}(a_{ki} - a_{kj})^2}$ should equal to zero and therefore the similarity score between two processes should have the score 1.

SimRank

SimRank is a method first proposed for website network analysis. It was aimed to measure the "similarity" between objects in a given web network. Potentially, we regards "two objects are similar if they are related to similar objects." And this general similarity measure is called SimRank [40]. In general, SimRank is a method to calculate the average of similarities between the neighbors of two objects. Together with other domain-specific similarity measures, it could be adjusted to satisfy different types of network, like homogenous domain and bipartite domain.

Based on similarity matrix established by previous algorithm, an adjusted SimRank algorithm could be used in this research. Firstly, I harvest similarity matrix by Average Difference algorithm between processes, and I regard the network as a homogenous domain with only industrial processes. Next, by setting a threshold, the neighbors of a process is defined according to the similarity from Average Difference algorithm (in this research, I take the most N similar processes as the neighbors of a certain process). Finally, I use SimRank algorithm to calculate new similarity between two processes *i* and *j*:

$$S_{ij} = \frac{C}{|I(i)||I(j)|} \sum_{a=1}^{|I(i)|} \sum_{b=1}^{|I(j)|} s(I_a(i), I_b(j))$$

Where:

C is a parameter to adjust the similarity locating between 0 to 1.

I(i) and *I(j)* are neighbors of *i* and *j*.

In this research for CI database, when calculate similarity between processes, I set a=20, C=1 and N=5. But when calculate similarity between environmental interventions, I set a=10, C=1 and N=5.

Table 2 List of similarty algorithms

	Similarity Algorithms for	
	Processes (i and j are two processes)	**Environmental interventions** (p and q are two environmental interventions)
Average Difference	$s_{ij} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{\lvert a_{ki} - a_{kj}\rvert}{\max(\lvert a_{ki}\rvert, \lvert a_{kj}\rvert)}$	$s_{pq} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{\lvert a_{kp} - a_{kq}\rvert}{\max(\lvert a_{kp}\rvert, \lvert a_{kq}\rvert)}$
Pearson Correlation Coefficient	$s_{ij} = \frac{\sum_{k=1}^{n}(a_{ki} - \overline{a_i})(a_{kj} - \overline{a_j})}{\sqrt{\sum_{k=1}^{n}(a_{ki} - \overline{a_i})^2 \cdot \sum_{k=1}^{n}(a_{kj} - \overline{a_j})^2}}$	$s_{pq} = \frac{\sum_{k=1}^{m}(a_{kp} - \overline{a_p})(a_{kq} - \overline{a_q})}{\sqrt{\sum_{k=1}^{m}(a_{kp} - \overline{a_p})^2 \cdot \sum_{k=1}^{m}(a_{kq} - \overline{a_q})^2}}$

Euclidean Distance	$S_{ij} = e^{\frac{-\sqrt{\sum_{k=1}^{n}(a_{ki}-a_{kj})^2}}{\sigma}}$	$S_{pq} = e^{\frac{-\sqrt{\sum_{k=1}^{n}(a_{kp}-a_{kq})^2}}{\sigma}}$
SimRank	$S_{ij} = \frac{C}{\lvert I(i)\rvert\lvert I(j)\rvert}\sum_{a=1}^{\lvert I(i)\rvert}\sum_{b=1}^{\lvert I(j)\rvert} s(I_a(i), I_b(j))$ where I(i) and I(j) are neighbors of i and j	$S_{pq} = \frac{C}{\lvert I(p)\rvert\lvert I(q)\rvert}\sum_{a=1}^{\lvert I(p)\rvert}\sum_{b=1}^{\lvert I(q)\rvert} s(I_a(p), I_b(q))$ where I(i) and I(j) are neighbors of i and j

5.1.2 Normalization of LCI data

5.1.2.1 The significance of normalization

There are many types of environmental interventions in LCI database representing input as resources, materials and output as emissions. Different types of environmental intervention have different unit, for example, unit of electricity input is *MJ*, but unit of CO2 emission is *ton*. Therefore, the values in the database have a large range of magnitude. The cumulative distribution of all values in CI database is shown in Figure 10.

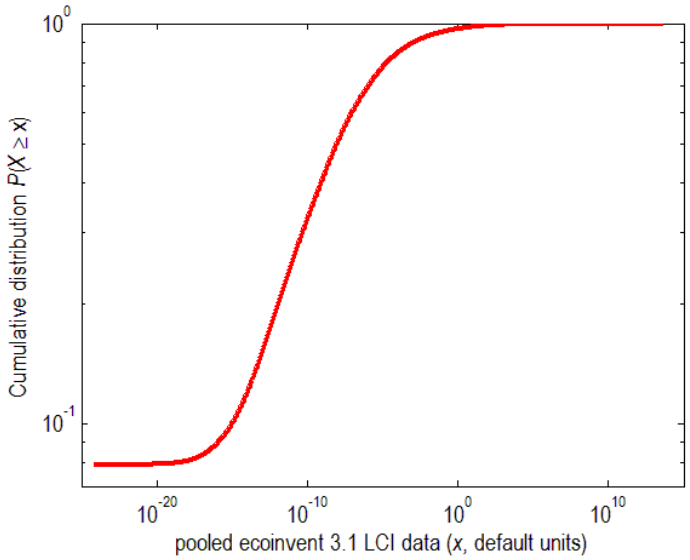


Figure 10 Cumulative distribution of values in CI database

Data magnitude may largely affect the estimation. For example if I use Euclidean Distance to calculate the similarity, data value with larger magnitude will have heavier impact to the similarity score than value with smaller magnitude. In order to eliminate impact from data magnitude, I proposed data normalization method to process data, so that all data would have the same degree of magnitude.

5.1.2.2 Normalization method

In this research, I first find the maximum value in each environmental intervention, and then use this maximum divide all value in this environmental intervention to normalize the database. That is to say, every data entry in the matrix is divided by the maximum value of all value in its row. It could be written in the matrix form as:

$$Normalized\ CI = (I \times (MAX))^{-1} \times CI$$

Where,

MAX is the vector of maximum value in each types of environmental intervention. i.e. vector of maximum value of each row;

CI is the original database;

I is the identity matrix.

Nevertheless, this normalization would not make any difference when use Average Difference to calculate similarity between processes and when use Pearson Correlation Coefficient to calculate similarity between environmental interventions. The deduction to get this conclusion is listed in Appendix A.

5.1.3 Evaluation of methods

I intend to test the performance of three new similarity algorithms, data normalization method and also similarity matrix between environmental interventions, and compare the results with prediction model using Average Difference similarity algorithm without data normalization to obtain similarity between processes which I talked in chapter 4.

Therefore, I have in total 16 combination scenarios to text. They are listed in Table 3. Taking all 16 scenarios, I went through the same process of chapter 4.2.2 that is to remove 1 data entry one by one and predict the missing data each time while recording their minimum errors and corresponding size of training data. In this chapter, after I have the minimum error and size of training data records, I counted the percentage of minimum error less than 0.1 and equals to 0. For size of training data, I counted the percentage that less than 100 and equals to 1 for every scenario for comparison. I define a good estimation is minimum error less than 0.1 with size of training data less than 100.

Table 3 List of all combination scenarios

Scenario	Similarity algorithm	Data normalization	Similarity between
1	Average Difference	NO	Processes
2	Average Difference	NO	Environmental interventions
3	Pearson Correlation Coefficient	NO	Processes
4	Pearson Correlation Coefficient	NO	Environmental interventions
5	Euclidean Distance	NO	Processes
6	Euclidean Distance	NO	Environmental interventions
7	SimRank	NO	Processes
8	SimRank	NO	Environmental interventions
9	Average Difference	YES	Processes
10	Average Difference	YES	Environmental interventions
11	Pearson Correlation Coefficient	YES	Processes
12	Pearson Correlation Coefficient	YES	Environmental interventions
13	Euclidean Distance	YES	Processes
14	Euclidean Distance	YES	Environmental interventions
15	SimRank	YES	Processes
16	SimRank	YES	Environmental interventions

5.2 Preliminary results

Table 4 gives the overview of the performance in minimum error and corresponding size of training data for the proposed 16 scenarios and Figure 11 shows their performance on minimum error less than 0.1 and size of training data less than 100.

From Table 4 and Figure 11, It can be found that by using Average Difference without data normalization to obtain similarities between processes (scenario1), 89.41% minimum errors are less than 0.1, 2.54% missing data could be accurately estimated with 0 estimation error. Its performance in these two parameters are much better than most of other scenarios, only Euclidean

Distance with data normalization calculating similarity between processes (Scenario 13) can have the relative competitive performance with it.

In addition, 83.63% minimum error could be found with a small size of training data defined as less than 100 most similar processes, so it also well performs in calculation efficiency. Although many other scenarios have better efficiency than scenario 1 since a lot more scenarios have higher percentage in size of training data less than 100, the efficiency of Scenario 1 is adequate enough for computing process and it's better than Scenario 13, which is competitive in the previous parameter.

Therefore it's reasonable to take Average Difference without data normalization to obtain similarity between processes (Scenario 1) as the best combination in prediction model. And this combination will be prioritized in the following research. There are graphs of minimum error and size of training data distribution from different scenarios listed in Appendix B.

Table 4 Parameter comparison between different scenarios

		Minimum error		Size of training data	
		0-0.1	0	0-100	1
1	Average Difference	89.41%	2.54%	83.63%	23.87%
2	Average Difference E	72.02%	0.55%	90.26%	18.38%
3	Correlation Coefficient	55.59%	1.52%	96.09%	36.92%
4	Correlation Coefficient E	17.73%	0.18%	95.40%	31.80%
5	Euclidean Distance	22.27%	0.00%	87.17%	16.76%
6	Euclidean Distance E	16.92%	0.00%	95.58%	50.70%
7	SimRank	51.45%	0.02%	70.88%	14.06%
8	SimRank E	40.92%	0.00%	84.23%	18.45%
9	Average Difference N				
10	Average Difference NE	19.84%	0.30%	95.70%	30.30%
11	Correlation Coefficient N	64.96%	0.87%	97.39%	35.91%
12	Correlation Coefficient NE				
13	Euclidean Distance N	86.76%	0.98%	58.72%	22.07%
14	Euclidean Distance NE	18.06%	0.36%	95.17%	31.89%
15	SimRank N				
16	SimRank NE	13.46%	0.00%	95.71%	30.24%

Note: Here E means it used similarity between environmental interventions to predict. N means it processed data normalization at the first step.

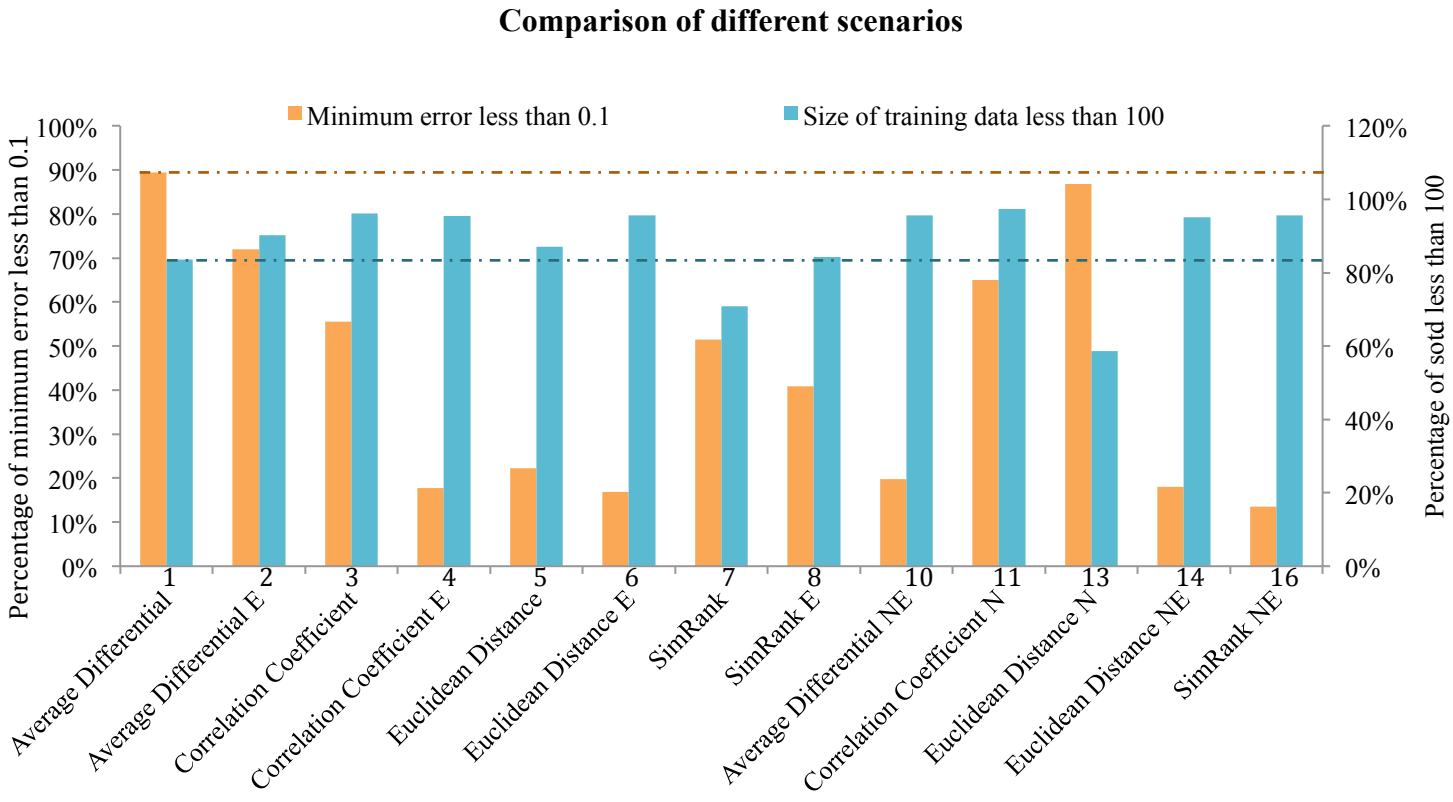


Figure 11 Comparison graph of minimum error between 0-0.1 and size of training data between 0-100 of different scenarios

Chapter 6 Optimal Size of Training Data – Finalize Prediction Method

The previous two chapters have carefully elaborated that for Ecoinvent 3.1 CI database, the proposed prediction model could result in a relative accurate and efficient estimation when 1 data entry miss, and a little less accurate but more efficient estimation when a block of 50 data entries miss. Result also shows that the model loaded Average Different similarity algorithm, calculating similarities between processes without normalization performs best among all methods.

Noticed that when I conduct these experiments, since I have an intact CI database, i.e. all missing data have their related validation database, it's realizable to test the estimation accuracy and efficiency by comparing estimation with validation data. However, in real world, there isn't "true value" of missing data or validation dataset when we want to predict it. Therefore, we cannot decide the certain amount of training data that can generate minimum estimation error(s) and also cannot test whether the estimation is accurate or not.

In this chapter, I use CI database to find the optimal size of training data that generate most accurate estimation. I take Average Difference as the similarity algorithm to calculate similarities between processes and not use normalization method. This size of training data can become a reference for real missing data estimation.

6.1 Methods

Firstly and most importantly, I propose the definition of optimal estimation as: when we use a certain size of training data, we can get the smallest average number for all estimation errors when 1 data entry removed and

estimated in the given database one by one. And this certain size of training data is called optimal size of training data.

So in this chapter, similar but unlike chapter 4.1.1, after I obtain the similarity matrix, I do not record minimum errors and corresponding size of training data. Instead, I set the size of training data from 1 to m-1 (m processes in total, and m = 8100 in CI database), and calculate the average value of estimation error matrix each time by equation (4). This value of d_t is also regarded as the difference between original value and estimation.

$$d_t = \frac{1}{s} * \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{|e_{ij} - a_{ij}|}{|a_{ij}|} \qquad (a_{ij} \neq 0) \qquad (4)$$

where *i*'s are environmental interventions and *j*'s are processes. e_{ij} is the estimation of entry *ij* and a_{ij} is the original value of entry *ij*.

In equation (4), *s* is the total elements number of the matrix excluding infinite number. Since there may be no consumption or emission of *i* in process *j*, the denominator a_{ij} could be 0 and estimation error which equals to $\frac{|e_{ij}-a_{ij}|}{|a_{ij}|}$ might be infinite, I intended to eliminate these infinite numbers. With *t* increases from 1 to *m*-1, there are *m* -1 predict matrices, and m-1 d_t's. The *t* gives us smallest d_t is the optimal size of training data.

6.2 Results

Figure 12 shows the plot between *t* and *dt*. It tells that as the size of training data increase, the estimation difference decrease first, but after t=7, the estimation difference starts to increase. After around t = 7000, the estimation difference grows sharply. The curve finally shows when the size of training data is 7, I can get the smallest estimation difference, i.e. the best estimation.

Therefore, I claim that when one uses my proposed similarity-based link prediction method to estimate missing data in Ecoinvent 3.1 CI database, seven most similar processes is the optimal size of training data to procure a relative accurate prediction. In addition, seven processes comparing with total 8100

processes is relatively a small value, thus this prediction method is also an efficiency approach.

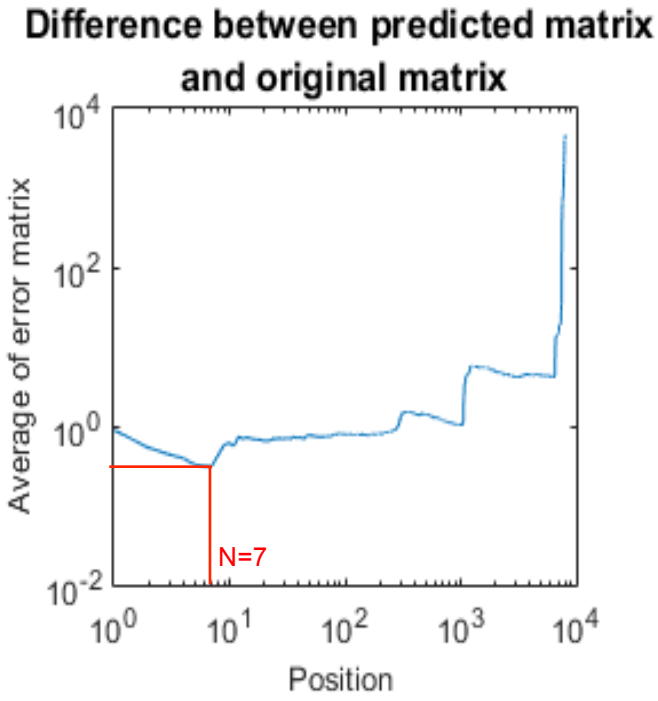


Figure 12 Relationship between size of training data and difference of prediction

This size of training data is only a reference for real world application because it was produced under the situation that only 1 data entry removed as missing data each time in Ecoinvent 3.1 CI database. However, people can repeat this experiment to find optimal size of training data under different situations. If this similarity-based link prediction method is used to predict future developing of CI database, it is recommended to use Average Difference algorithm to calculate similarity between processes, and take 7 most similar processes as training data.

Chapter 7 Discussion and Future Development

7.1 Discussion

From all the experiment results, it shows that these preliminary similarity-based link prediction models are promising and have strong potentials in LCA data estimation. This method can be applied to LCI database to improve the efficiency of database development, evaluate data quality and predict LCI data for emerging technologies.

First, empirically collecting LCI data is expensive and time-consuming. Predict missing links in the network constructed based on limited known LCI data is equivalent to estimating missing data in the LCI database without empirical data. This will significantly reduce the cost of and save time for creating an LCI database.

Second, data in an LCI database come from various sources with different quality and accuracy. One can use a portion of the LCI database that are trustworthy to predict the rest of observed links in the LCI network. By comparing the predicted results with the observed data, one can evaluate the quality of those observed data. The distance (error) between the best estimation and the true value indicates the quality of the particular data point, i.e., the quality of a data point is poor if the error is large. The dataset developer may need to recheck the data if there are typo mistake exist, such as wrong unit, etc. This can help LCA practitioners assess the quality of their LCI data, identify inaccurate data, and guide future improvements[47].

Lastly, LCI databases are always expanding due to the addition of new unit processes. It is often the case that environmental intervention data for new unit processes are incomplete, especially for processes representing emerging technologies. The technology system is constantly evolving in the way that new unit processes and products are invented all the time. Predicting emerging links between the new process and environmental interventions can help reasonably

estimate LCI data for unit processes of emerging technologies. This is particularly useful for LCA of emerging technologies for which empirical LCI data are less available[48].

7.2 Future research

Besides Ecoinvent, more LCI database could be tested with this method. Candidate databases for further consideration includes GREET (greet.es.anl.gov) and the US LCI Database (www.lcacommons.gov/nrel). These are initially selected because they represent a variety of LCI databases. In particular, Ecoinvent is one of the proprietary databases with comprehensive coverage; GREET is developed for a particular sector (transportation); and the US LCI Database is a national reference LCI database that provides industrial-representative LCI data for a particular country. GREET (latest 1 Series for fuel cycle and 2 Series for vehicle cycle) databases come as Excel spreadsheets that are ready to be processed and converted into MATLAB files. The US LCI Database has recently been compiled into Excel spreadsheets by [49] with open access.

Besides similarity-based link prediction, other methods, such as maximum likelihood model in link prediction, matrix completion etc. could also be used to predict the missing data. Maximum likelihood methods predict missing links based on presupposed structure features of the network. Matrix completion is to fill in the missing data entries of a partially observed matrix. Future study will explore the potential application of these methods in LCI database construction.

Reference

[1] M. A. Curran, "Life-cycle based government policies a survey," *Int. J. Life Cycle Assess.*, vol. 2, no. 1, pp. 39–43, 1997.

[2] B. Boughton, "California's green chemistry initiative: Application of life cycle assessment in public policy.," in *Abstracts of Papers, 239th ACS National Meeting, San Francisco, CA, United States, March 21-25, 2010*, 2010, p. CINF-22.

[3] N. D. Orfield, R. B. Levine, G. A. Keoleian, S. A. Miller, and P. E. Savage, "Growing algae for biodiesel on direct sunlight or sugars: A comparative life cycle assessment," *ACS Sustain. Chem. Eng.*, vol. 3, no. 3, pp. 386–395, 2015.

[4] Z. Bi, L. Song, R. De Kleine, C. C. Mi, and G. A. Keoleian, "Plug-in vs. wireless charging: Life cycle energy and greenhouse gas emissions for an electric bus system," *Appl. Energy*, vol. 146, pp. 11–19, 2015.

[5] D. L. Reed, "Life-Cycle Assessment in Government Policy in the United States," University of Tennessee, 2012.

[6] S. Humbert, V. Rossi, M. Margni, O. Jolliet, and Y. Loerincik, "Life cycle assessment of two baby food packaging alternatives: Glass jars vs. plastic pots," *Int. J. Life Cycle Assess.*, vol. 14, no. 2, pp. 95–106, 2009.

[7] S. J. Skerlos, "Promoting effectiveness in sustainable design," in *Procedia CIRP*, 2015, vol. 29, pp. 13–18.

[8] W. B. Trusty, S. Horst, T. A. Thena, S. Materials, J. Street, and P. O. Box, "Integrating LCA Tools in Green Building Rating Systems," *Athena*, pp. 1–7, 2007.

[9] K. Börner, S. Sanyal, and A. Vespignani, "Network science.," *Annu. Rev. Inf. Sci. Technol.*, vol. 41, pp. 537–607, 2007.

[10] L. Getoor and C. P. Diehl, "Link mining: A survey," *ACM SIGKDD Explor. Newsl.*, vol. 7, no. 2, pp. 3–12, 2005.

[11] G. Kossinets, "Effects of missing data in social networks," *Soc. Networks*, vol.

28, no. 3, pp. 247–268, 2006.

[12] P. Holme and M. Huss, “Role-similarity based functional prediction in networked systems: application to the yeast proteome.,” *J. R. Soc. Interface*, vol. 2, no. 4, pp. 327–33, 2005.

[13] C. T. Butts, “Network inference, error, and informant (in)accuracy: A Bayesian approach,” *Social Networks*, vol. 25, no. 2. pp. 103–140, 2003.

[14] C. von Mering, R. Krause, B. Snel, M. Cornell, S. G. Oliver, S. Fields, and P. Bork, “Comparative assessment of large-scale data sets of protein-protein interactions.,” *Nature*, vol. 417, no. 6887, pp. 399–403, 2002.

[15] S. Zhou and R. J. Mondragón, “Accurately modeling the internet topology,” *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, vol. 70, no. 6 2, 2004.

[16] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir, “A model of Internet topology using k-shell decomposition,” *Proc. Natl. Acad. Sci.*, vol. 104, no. 27, pp. 11150–11154, 2007.

[17] M.-S. S. and L. L. and Y.-C. Z. and T. Zhou, “Empirical analysis of web-based user-object bipartite networks,” *EPL (Europhysics Lett.*, vol. 90, no. 4, p. 48006, 2010.

[18] M. A. Curran, “Sourcing Life Cycle Inventory Data,” in *Life Cycle Assessment Handbook*, 2012, pp. 105–141.

[19] A. E. Björklund, “Survey of approaches to improve reliability in lca,” *Int. J. Life Cycle Assess.*, vol. 7, no. 2, pp. 64–72, 2002.

[20] B. Sauer, “Life Cycle Inventory Modeling in Practice,” in *Life Cycle Assessment Handbook: A Guide for Environmentally Sustainable Products*, 2012, pp. 43–66.

[21] S. A. Miller and G. A. Keoleian, “Framework for analyzing transformative technologies in life cycle assessment,” *Environ. Sci. Technol.*, vol. 49, no. 5, pp. 3067–3075, 2015.

[22] S. Suh and G. Huppes, “Missing inventory estimation tool using extended input-output analysis,” *Int. J. Life Cycle Assess.*, vol. 7, no. 3, pp. 134–140, 2002.

[23] G. J. Treloar, “Extracting embodied energy paths from input-output tables: towards an input-output-based hybrid energy analysis method,” *Econ. Syst. Res.*, vol. 9, no. 4, pp. 375–391, 1997.

[24] R. H. Crawford, “Validation of a hybrid life-cycle inventory analysis method,” *J. Environ. Manage.*, vol. 88, no. 3, pp. 496–506, 2008.

[25] M. E. J. Newman, *Networks: An introduction*. 2014.

[26] J. McNerney, B. D. Fath, and G. Silverberg, “Network structure of inter-industry flows,” *Phys. A Stat. Mech. its Appl.*, vol. 392, no. 24, pp. 6427–6441, 2013.

[27] S. Liang, Y. Feng, and M. Xu, “Structure of the Global Virtual Carbon Network: Revealing Important Sectors and Communities for Emission Reduction,” *J. Ind. Ecol.*, vol. 19, no. 2, pp. 307–320, 2015.

[28] K. Lewis, M. Gonzalez, and J. Kaufman, “Social selection and peer influence in an online social network,” *Proc. Natl. Acad. Sci.*, vol. 109, no. 1, pp. 68–72, 2012.

[29] L. Lü and T. Zhou, “Link prediction in complex networks: A survey,” *Phys. A Stat. Mech. its Appl.*, vol. 390, no. 6, pp. 1150–1170, 2011.

[30] Z. Huang and D. D. Zeng, “A link prediction approach to anomalous email detection,” in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2007, vol. 2, pp. 1131–1136.

[31] V. E. Krebs, “Mapping Networks of Terrorist Cells,” *Connections*, vol. 24, no. 3, pp. 43–52, 2002.

[32] J. Kunegis, E. W. De Luca, and S. Albayrak, “The link prediction problem in bipartite networks,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010, vol. 6178 LNAI, pp. 380–389.

[33] O. Allali, C. Magnien, and M. Latapy, “Link prediction in bipartite graphs using internal links and weighted projection,” in *2011 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2011*, 2011, pp. 936–941.

[34] C. Gao, D. Wei, Y. Hu, S. Mahadevan, and Y. Deng, “A modified evidential methodology of identifying influential nodes in weighted networks,” *Phys. A Stat. Mech. its Appl.*, vol. 392, no. 21, pp. 5490–5500, 2013.

[35] Y. Koren and R. Bell, “Advances in Collaborative Filtering,” *Recomm. Syst. Handb.*, pp. 145–186, 2011.

[36] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso, “Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems,” *ACM Trans. Web*, vol. 5, no. 1, p. 2:1–2:33, 2011.

[37] J. S. Breese, D. Heckerman, and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” *Proc. 14th Conf. Uncertain. Artif. Intell.*, vol. 461, no. 8, pp. 43–52, 1998.

[38] J. Zhao, L. Miao, J. Yang, H. Fang, Q.-M. Zhang, M. Nie, P. Holme, and T. Zhou, “Prediction of Links and Weights in Networks by Reliable Routes.,” *Sci. Rep.*, vol. 5, p. 12261, 2015.

[39] L. Lü, L. Pan, T. Zhou, Y.-C. Zhang, and H. E. Stanley, “Toward link predictability of complex networks,” *Proc. Natl. Acad. Sci.*, vol. 112, no. 8, p. 201424644, 2015.

[40] G. Jeh and J. Widom, “SimRank : A Measure of Structural-Context Similarity ∗,” *Proc. eighth ACM SIGKDD Int. …*, pp. 1–11, 2001.

[41] G. Wernet, S. Papadokonstantakis, S. Hellweg, and K. Hungerbühler, “Bridging data gaps in environmental assessments: Modeling impacts of fine and basic chemical production,” *Green Chem.*, vol. 11, no. 11, p. 1826, 2009.

[42] G. Wernet, S. Hellweg, S. Papadokonstantakis, and K. Hungerbuhler, “Molecular-Structure-Based Models of Chemicsl Inventories using Neural Networks,” *Environemental Sci. Technol.*, vol. 42, no. 17, pp. 6717–6722, 2008.

[43] B. P. Wedema, C. Bauer, R. Hischier, C. Mutel, T. Nemecek, J. Reinhard, C. O. Vadenbo, and G. Wenet, “Overview and methodology. Data quality

guideline for the ecoinvent database version 3," 2013.

[44] W. Souma, Y. Fujiwara, and H. Aoyama, "Complex networks and economics," in *Physica A: Statistical Mechanics and its Applications*, 2003, vol. 324, no. 1–2, pp. 396–401.

[45] A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, "The architecture of complex weighted networks.," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 101, no. 11, pp. 3747–3752, 2004.

[46] M. E. J. Newman, "Analysis of weighted networks," *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, vol. 70, no. 5 2, 2004.

[47] J. S. Cooper and E. Kahn, "Commentary on issues in data quality analysis in life cycle assessment," *International Journal of Life Cycle Assessment*, vol. 17, no. 4. pp. 499–503, 2012.

[48] T. E. McKone, W. W. Nazaroff, P. Berck, M. Auffhammer, T. Lipman, M. S. Torn, E. Masanet, A. Lobscheid, N. Santero, U. Mishra, A. Barrett, M. Bomberg, K. Fingerman, C. Scown, B. Strogen, and A. Horvath, "Grand challenges for life-cycle assessment of biofuels," *Environ. Sci. Technol.*, vol. 45, no. 5, pp. 1751–1756, 2011.

[49] D. H. M. Matthews, H. Scott, C.T. Hendrickson, "Life Cycle Assessment: Quantitative Approaches for Decisions That Matter," 2014.

Appendix A

Deduction of normalization effect on four similarity algorithms

Average Difference

$$s_{ij} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{|a_{ki} - a_{kj}|}{\max(|a_{ki}|, |a_{kj}|)} \quad (1)$$

After the data normalization, I can find that the normalization will be offset if I calculate similarity between processes by Average Difference. Because:

$$s'_{ij} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{\left|\frac{a_{ki}}{MAX_k} - \frac{a_{kj}}{MAX_k}\right|}{\max\left(\left|\frac{a_{ki}}{MAX_k}\right|, \left|\frac{a_{kj}}{MAX_k}\right|\right)} = 1 - \frac{1}{n}\sum_{k=1}^{n} \frac{\left|\frac{1}{MAX_k}\right| \cdot |a_{ki} - a_{kj}|}{\left|\frac{1}{MAX_k}\right| \cdot \max(|a_{ki}|, |b_{kj}|)}$$

$$= 1 - \frac{1}{n}\sum_{k=1}^{m} \frac{|a_{ki} - a_{kj}|}{\max(|a_{ki}|, |a_{kj}|)} = s_{ij}$$

where *i,j* represents two processes,

n is the number of environmental interventions,

a_{ki} is the environmental intervention *k* for unit process *i*.

But for similarity calculation between environmental interventions by Average Difference, the normalization effect will hold because a_{ki} and a_{kj} come from two different environmental interventions thus will be devided by two different maximum values (MAX_i, MAX_j).

Pearson Correlation Coefficient

Different with Average Difference, when I use Pearson Correlation Coefficient, normalization effect will offset when it applied to calculate similarities between environmental interventions. It also can be proven as an equation:

$$s'_{pq} = \frac{\sum_{k=1}^{m}(\frac{a_{kp}}{MAX_p} - \frac{\overline{a_p}}{MAX_p})(\frac{a_{kq}}{MAX_q} - \frac{\overline{a_q}}{MAX_q})}{\sqrt{\sum_{k=1}^{m}\left(\frac{a_{kp}}{MAX_p} - \frac{\overline{a_p}}{MAX_p}\right)^2 \cdot \sum_{k=1}^{m}\left(\frac{a_{kq}}{MAX_q} - \frac{\overline{a_q}}{MAX_q}\right)^2}}$$

$$= \frac{\frac{1}{MAX_p MAX_q}\sum_{k=1}^{m}(a_{kp} - \overline{a_p})(a_{kq} - \overline{a_q})}{\frac{1}{MAX_p MAX_q}\sqrt{\sum_{k=1}^{m}(a_{kp} - \overline{a_p})^2 \cdot \sum_{k=1}^{m}(a_{kq} - \overline{a_q})^2}}$$

$$= \frac{\sum_{k=1}^{m}(a_{kp} - \overline{a_p})(a_{kq} - \overline{a_q})}{\sqrt{\sum_{k=1}^{m}(a_{kp} - \overline{a_p})^2 \cdot \sum_{k=1}^{m}(a_{kq} - \overline{a_q})^2}} = s_{pq}$$

But when it is applied between processes, each term in the polynomial is supposed to be divided by a different MAX. In the equation, MAX_p and MAX_q would be replaced by MAX_k. Therefore, the normalization effect does not be offset.

Euclidean Distance

However, the difference in magnitude between data would dramatically change the similarity result from Euclidean Distance algorithm. Since the algorithm of Euclidean Distance is:

$$S_{ij} = e^{\frac{-\sqrt{\sum_{k=1}^{n}(a_{ki} - a_{kj})^2}}{\sigma}}$$

Data with large magnitude have much heavier weights in term $\sum_{k=1}^{n}(a_{ki} - a_{kj})^2$. So with a negative sign before this term, the similarity between i and j is determined by data with small magnitude in i and j in some degree. I could also prove this phenomenon through following deduction:

$$e^{\frac{-\sqrt{(100a_{ki} - 100a_{kj})^2}}{\sigma}} = e^{\frac{-100\sqrt{(a_{ki} - a_{kj})^2}}{\sigma}} \ll e^{\frac{-\sqrt{(a_{ki} - a_{kj})^2}}{\sigma}}$$

SimRank

SimrRank is built based on Average Difference. It uses similarity from Average Difference as the original information to determine “neighbors”. Therefore, how data normalization affect basic method similarity will affect results from SimRank in the same way.

Appendix B

Graphs of minimum error and size of training data by different similarity algorithms used for processes/environmental interventions and with/without data normalization

1. Use Average Difference to calculate similarity between processes without data normalization

See Chapter 4.2.2

2. Use Average Difference to calculate similarity between environmental interventions without data normalization

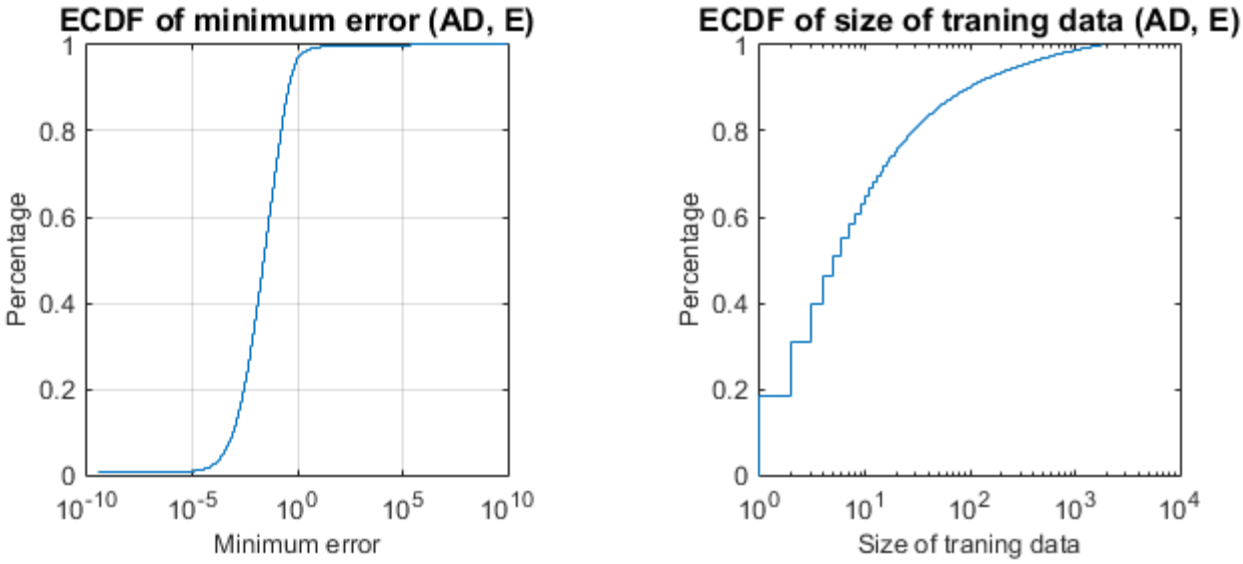


Figure 13 ECDF of minimum error and size of training data using Average Difference to calculate similarity between processes without data normalization in one data missing scenario

3. Use Pearson Correlation Coefficient to calculate similarity between processes without data normalization

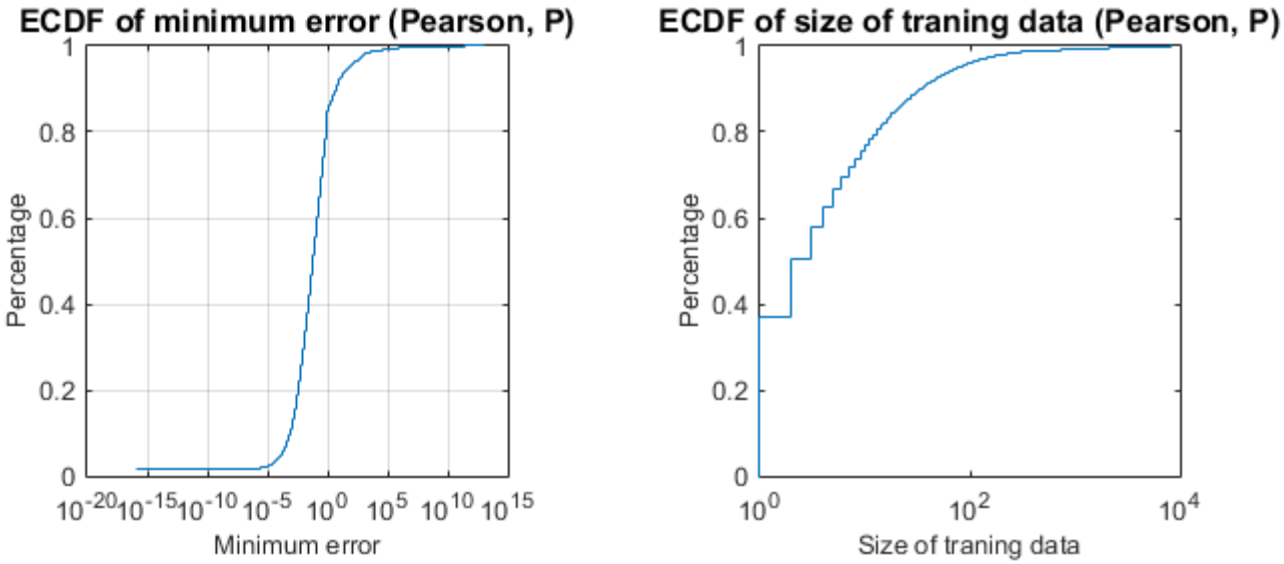


Figure 14 ECDF of minimum error and size of training data using Person Correlation Coefficient to calculate similarity between processes without data normalization in one data missing scenario

4. Use Person Correlation Coefficient to calculate similarity between environmental interventions without data normalization

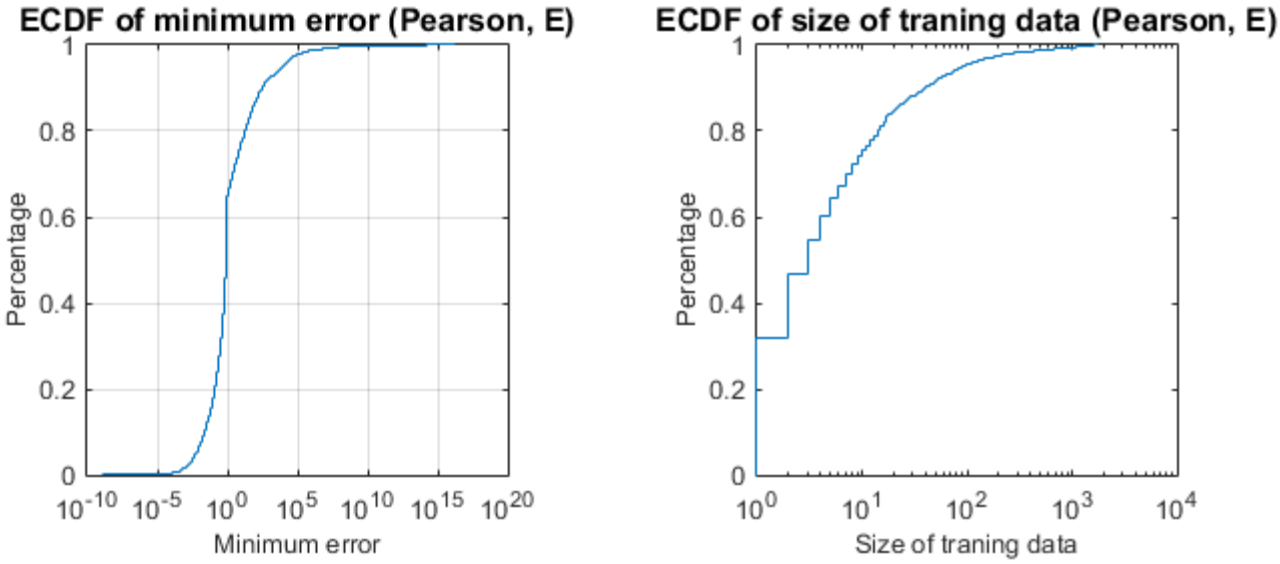


Figure 15 ECDF of minimum error and size of training data using Pearson Correlation Coefficient to calculate similarity between environmental interventions without data normalization in one data missing scenario

5. Use Euclidean Distance to calculate similarity between processes without data normalization

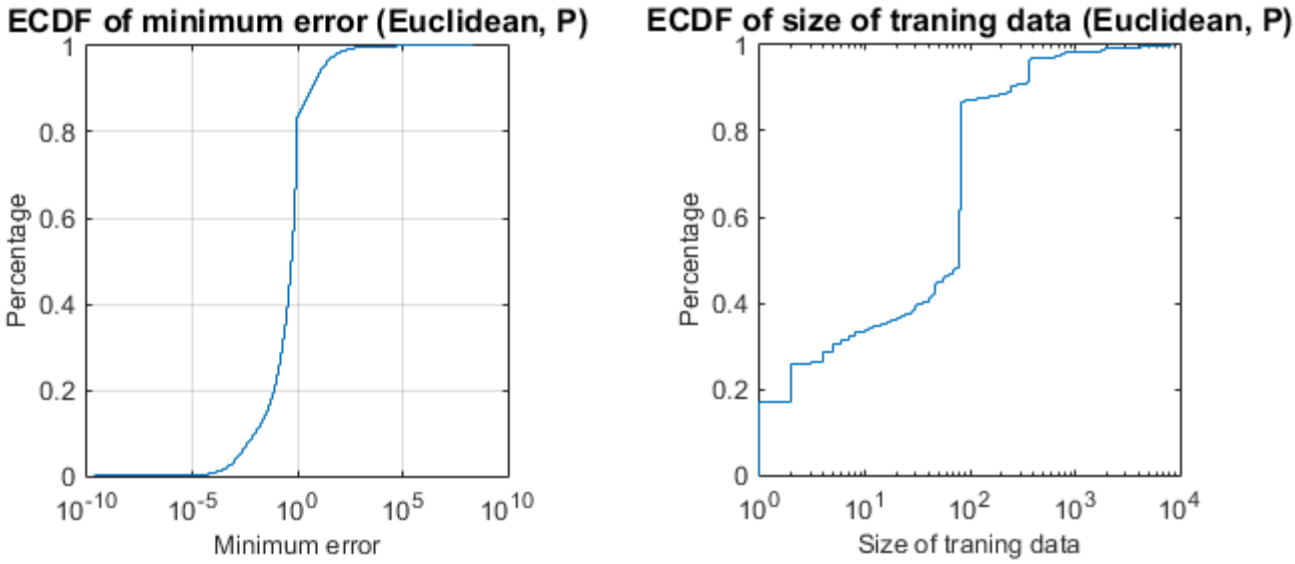


Figure 16 ECDF of minimum error and size of training data using Euclidean Distance to calculate similarity between processes without data normalization in one data missing scenario

6. Use Euclidean Distance to calculate similarity between environmental interventions without data normalization

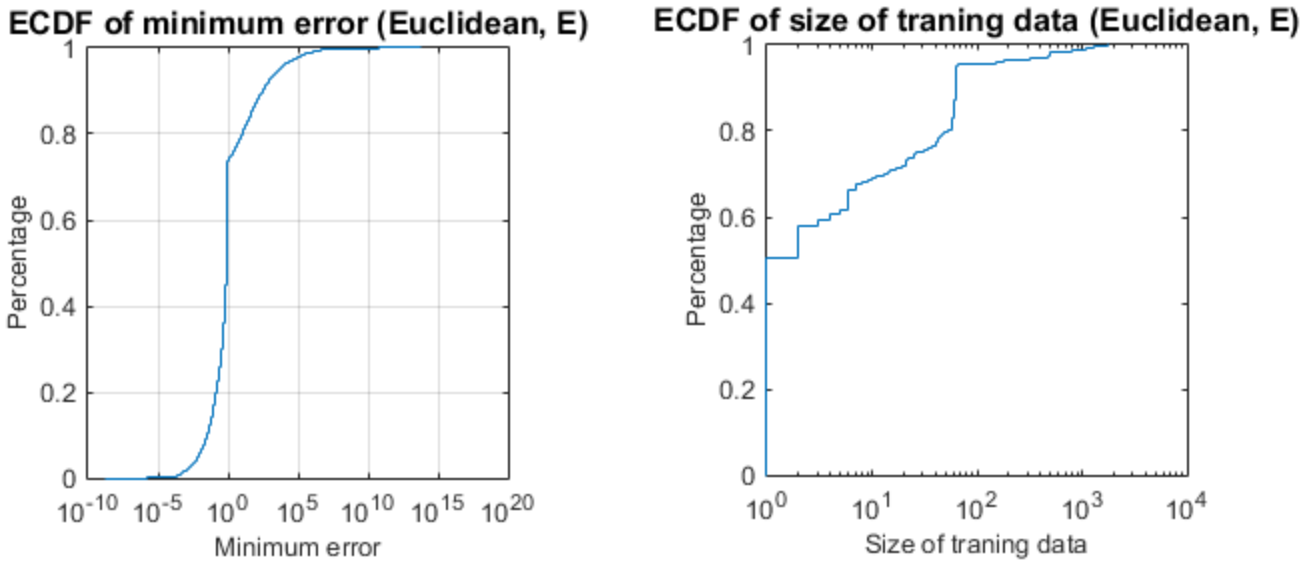
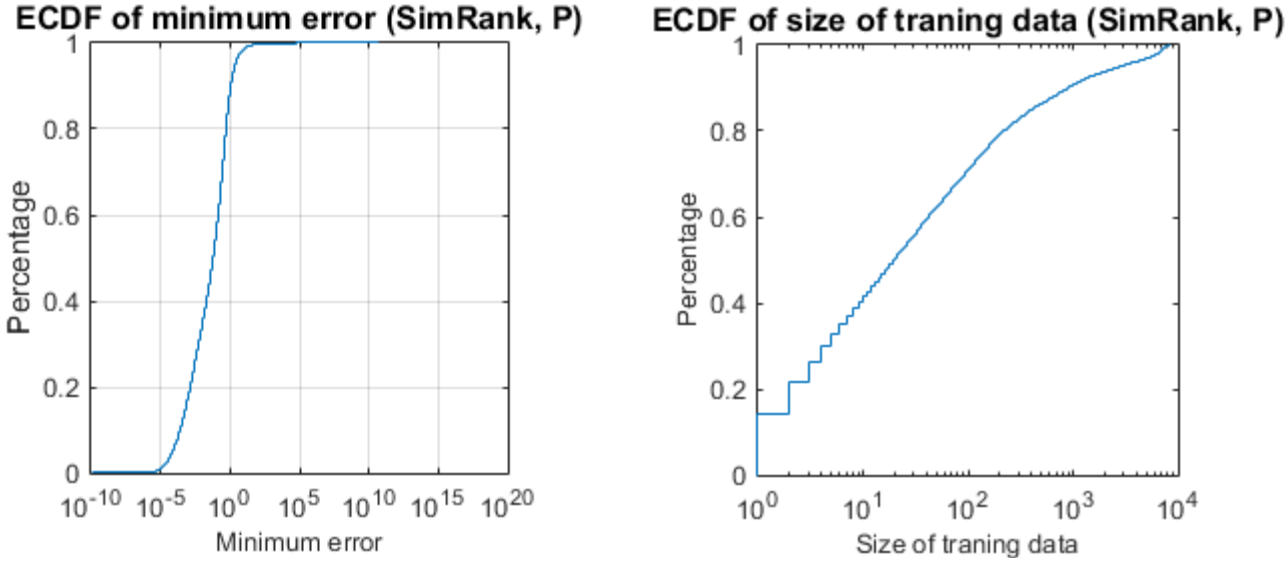


Figure 17 ECDF of minimum error and size of training data using Euclidean Distance to calculate similarity between environmental interventions without data normalization in one data missing scenario

7. Use SimRank to calculate similarity between processes without data normalization



8. Use SimRank to calculate similarity between environmental interventions without data normalization

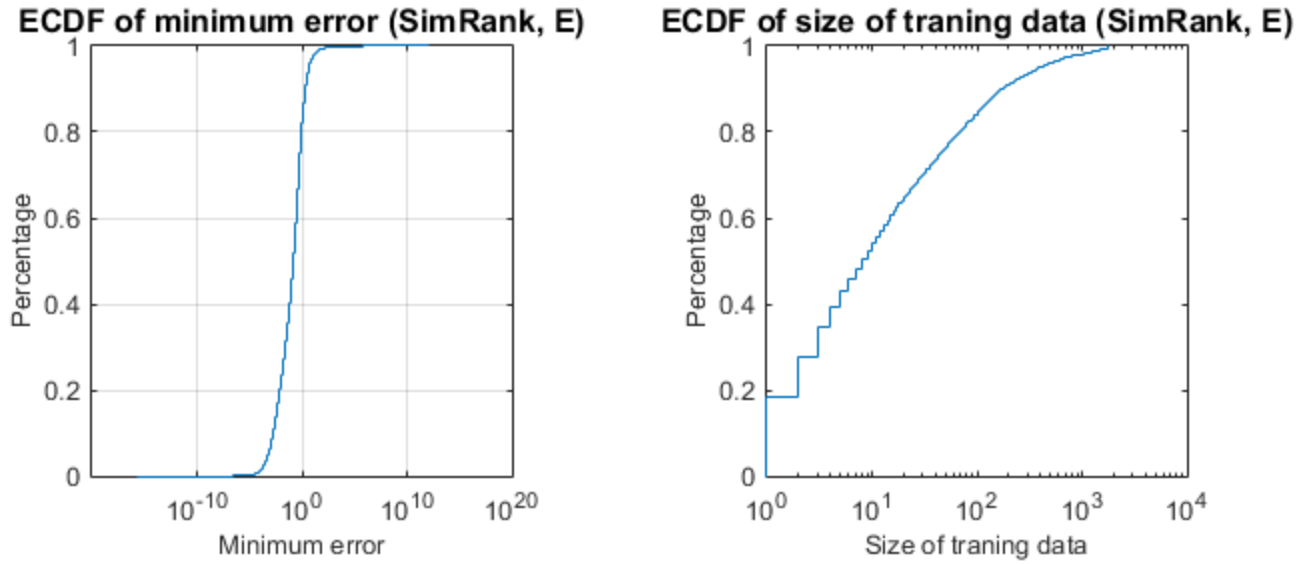


Figure 18 ECDF of minimum error and size of training data using SimRank to calculate similarity between environmental interventions without data normalization in one data missing scenario

9. Use Average Difference to calculate similarity between processes with data normalization

Results are same as this method without normalization.

10. Use Average Difference to calculate similarity between environmental interventions with data normalization

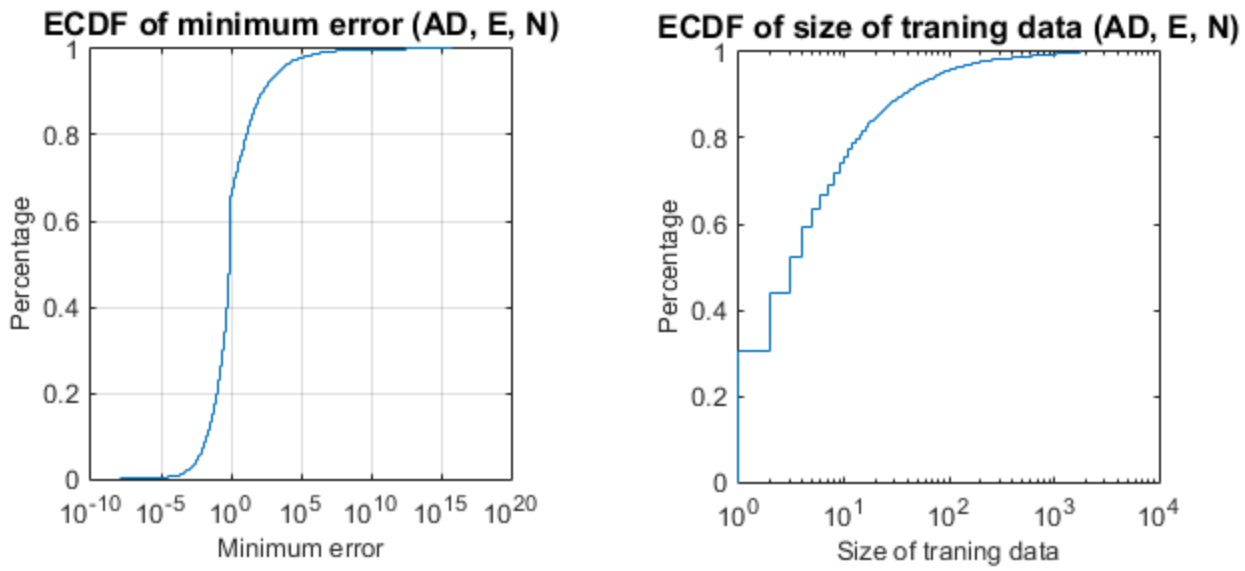


Figure 19 ECDF of minimum error and size of training data using Average Difference to calculate similarity between environmental interventions with data normalization in one data missing scenario

11. Use Pearson Correlation Coefficient to calculate similarity between processes with data normalization

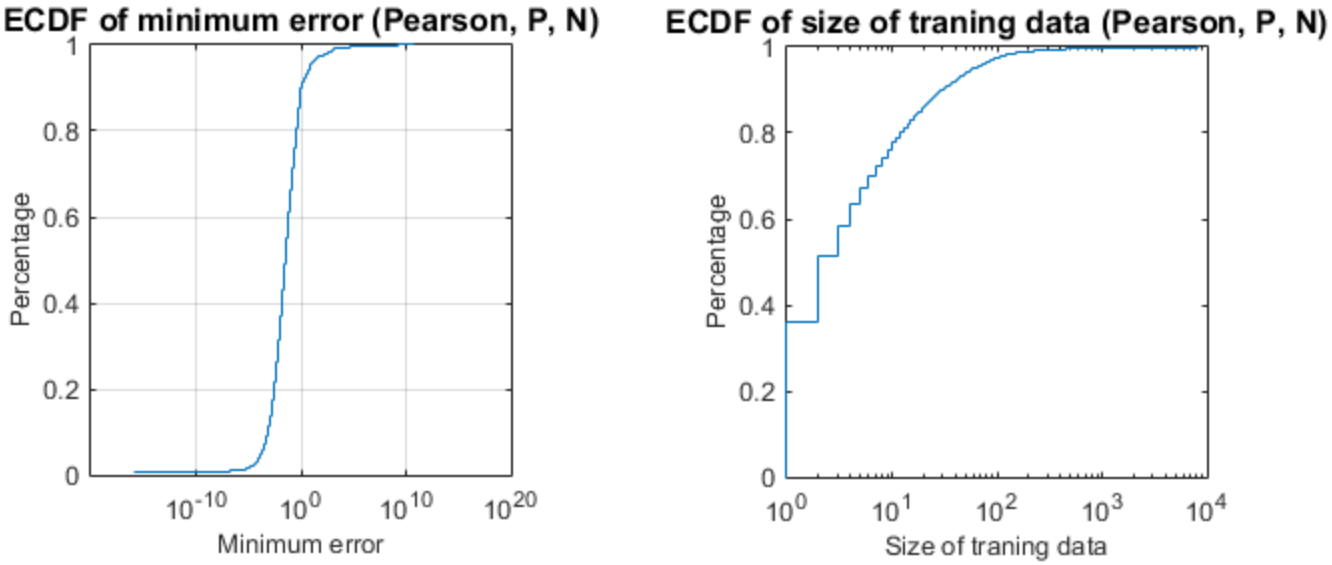


Figure 20 ECDF of minimum error and size of training data using Pearson Correlation Coefficient to calculate similarity between processes with data normalization in one data missing scenario

12. Use Pearson Correlation Coefficient to calculate similarity between environmental interventions with data normalization

Results are same as this method without normalization.

13. Use Euclidean Distance to calculate similarity between Processes with data normalization

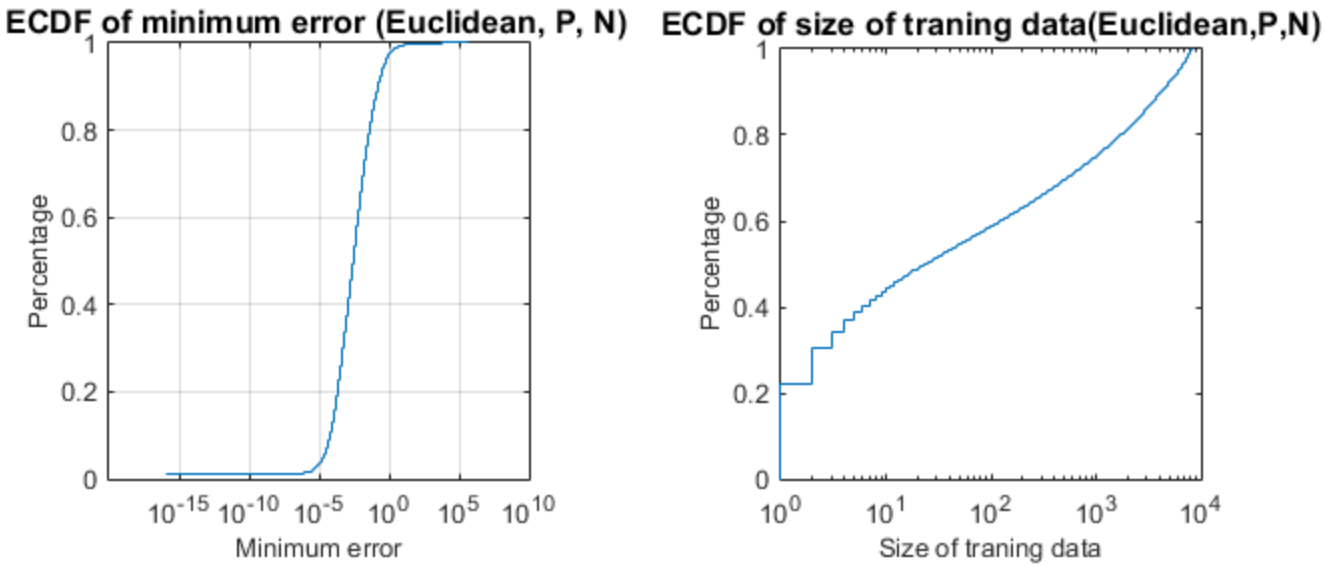


Figure 21 ECDF of minimum error and size of training data using Euclidean Distance to calculate similarity between processes with data normalization in one data missing scenario

14. Use Euclidean Distance to calculate similarity between environmental interventions with data normalization

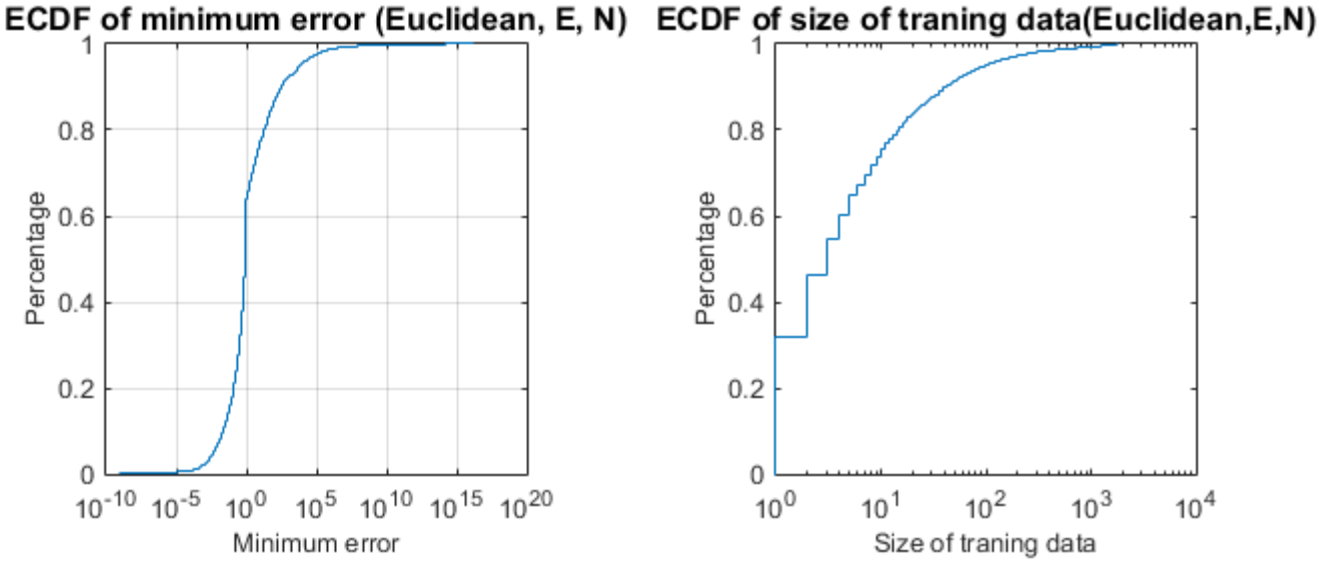


Figure 22 ECDF of minimum error and size of training data using Euclidean Distance to calculate similarity between environmental interventions with data normalization in one data missing scenario

15. Use Euclidean Distance to calculate similarity between Processes with data normalization

Results are same as this method without normalization.

16. Use SimRank to calculate similarity between environmental interventions with data normalization

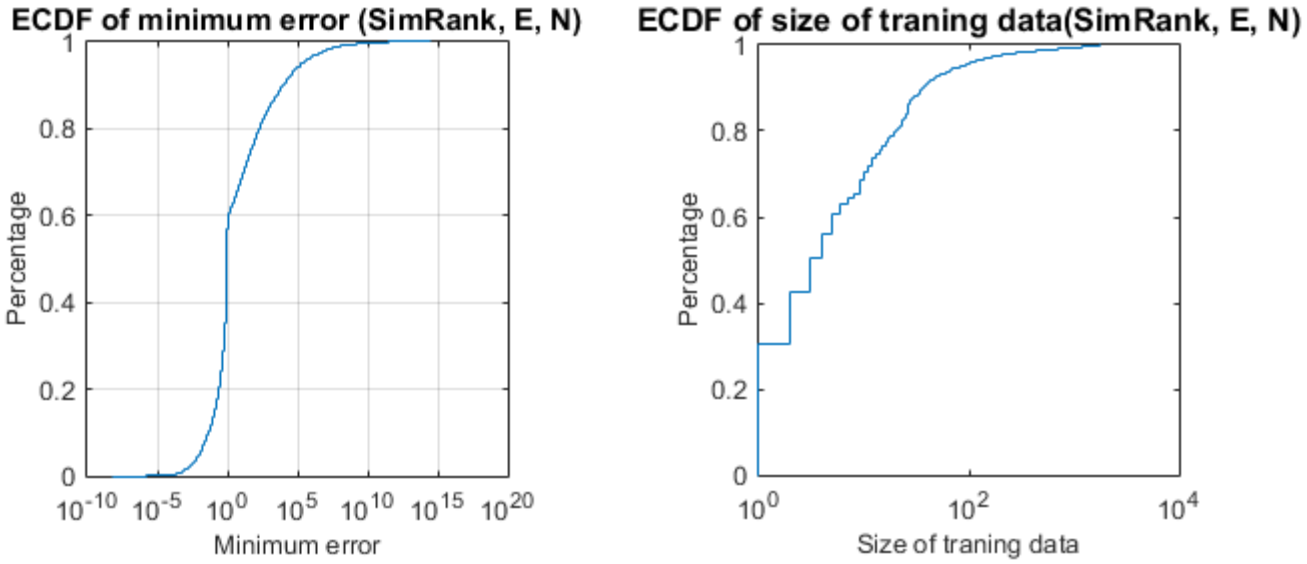


Figure 23 ECDF of minimum error and size of training data using SimRank to calculate similarity between processes with data normalization in one data missing scenario