Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Modelling Uncertainty and Vagueness within Recommender Systems via Nonparametric Predictive Inference

Angela McCourt

A Dissertation submitted to the University of Dublin, Trinity College

in fulfilment of the requirements for the degree of

Doctor of Philosophy (Statistics)

2019

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work.

I agree to deposit this thesis in the University's open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

_____

Angela McCourt

Dated:

Acknowledgements

You are braver than you believe, stronger than you seem, and smarter than you think.

– Christopher Robin to Winnie the Pooh

The Chinese proverb states that "a single conversation with a wise man is better than ten years of study" and I have been very fortunate to converse with several wise men and women during my years in academia. First, and foremost, I wish to thank my supervisor, Prof. Brett Houlding for offering me support, expertise, encouragement and humorous conversations during the process of my Ph.D. studies. He offered kind words during dark days and kept me focused on my goals. Dr. Ann O'Shea was another wise individual whom I have had the pleasure of gaining insight, wisdom and encouragement to pursue mathematics and statistics.

A special thank you to Dr Donnacha Bolger for answering my call which was the first step on my Ph.D. journey. He provided much laughter and support during my first year. Thank you to Joy, Asmaa, Michael, Arthur, Myra, Gernot, Séan, Riccardo, Cristina and Jason for their conversations related to my work and in particular those that provided a break from it. Finally, to my new colleagues in the CSO, thank you for assuring me there is life after a Ph.D. and for your proof reading expertise.

Thank you to my entire family, especially my parents Peter and Margaret and my sister Teresa, who have supported and encouraged me during my many years of study. They also helped to remind me that there is more to life than academia. A special acknowledgement of my brother, Peter. I dedicate this thesis to his wife, Samantha and their wonderful son, Graham. The world is less bright without you both.

Martin, Kevin and Conor, simply thank you for being wonderful. You provide me with strength, happiness and purpose everyday.

Abstract

The way in which we learn is the subject of considerable research within multiple disciplines. There is also a vast amount of on-line material available to us, causing decision-making to become increasingly difficult. Learning preferences for decision-making processes has been an area of substantial research in recent years given the introduction of Recommender Systems (RSs). RSs help in decision-making processes by recommending items of interest and filtering out undesired items, they need to learn preferences by extracting information about both the user and the item. This thesis presents a novel approach of incorporating vagueness and uncertainty into recommendations via Nonparametric Predictive Inference (NPI). This approach is termed the Uncertainty Interval (UI); it is a modified version of Nonparametric Predictive Utility Intervals. There are four UI approaches presented: $UI_{Untrans}$, UI_{Abs}, UI_{Sq} and UI_{Rt}. Each algorithm is evaluated and compared with a similar technique, Robust Bayesian Correlation Learning. The UI_{Abs} algorithm has superior performance to the other UI approaches and is applied to real world data. The width of the interval reflects the amount of information available to the RS, with a wider interval indicating little or no information. The interval narrows as more information is incorporated into the UI algorithm.

Contents

List of Tables

List of Figures

1

Introduction

This thesis presents what I will name the Uncertainty Interval (UI) approach, a novel algorithm that incorporates vagueness and uncertainty into recommendations. The UI algorithm enables a Recommender System (RS) to calculate an interval utility function for users' preferences. This is achieved using a modified statistical approach, Nonparametric Predictive Inference (NPI), to allow pairwise correlation coefficients to be combined in order to produce a recommendation for a novel item/user. This chapter describes the motivations behind the work; the challenges to the development of this algorithm in relation to RSs are also outlined, before presenting the goals and contributions of this work. Finally, the thesis structure of the rest of this thesis is described.

1.1 Motivation

The way in which individuals learn is the subject of considerable research within multiple disciplines, including statistics, psychology, economics, computer science, philosophy, political and social science, *etc.* What individuals learn and the context in which they learn may affect decision-making processes. That is, individuals learn about their preferences and these preferences, along with judgements (assessments about the external state of the world), influence decisions. The decision-making process requires the ability to assess and filter a large amount of information and is generally done so in the face of uncertainty, *i.e.*, decisions made with limited information about their potential consequences. Psychology research in the last four decades has devoted a good deal of attention to an examination of how individuals process information.

One of the most cited studies on the significant ways in which human behaviour deviates from predicted behaviour due to cognitive biases and heuristics is that of Kahneman & Tversky (1979). The implications of cognitive biases and heuristics have been explored to challenge the assumption that individuals can properly assess the likelihood of various outcomes so that they can make the utility-maximizing choice among outcomes (Gigerenzer & Gaissmaier, 2011; Simon *et al.*, 1987).

The study of decision-making is important as individuals are makers of a choice and must suffer the consequences of such decisions (Lindley, 1991). The majority of decisions are not momentary, with some decisions requiring a great deal of thought, especially if those decisions have long term effects or have substantial costs attached. Decision theory, much like learning, encompasses a variety of disciplines. Normative decision theory is concerned with the study of strategies for decision-making under conditions of uncertainty in such a way as to maximize expected utility, and is concerned with the type of decisions that *ought* to be made. This approach uses rationality in order to make a decision. Although, there are those from the Normative school of thought that may disagree that expected utility is Normative, *e.g.*, Allais (1953), who argues that *higher moments* of utility are also important. Whilst descriptive approaches to decision-making attempts to describe what people will *actually* do and how decisions are made in reality.

As individuals, we are spending increasingly more time on-line as we have more access to the Internet via smart phones, watches, tablets, *etc.*, see Gunter (2017); Perrin (2015); Yonker *et al.* (2015). There is also a vast amount of on-line material available to us, causing decision-making to become increasingly difficult. Learning preferences for decision-making processes has been an area of substantial research in recent years given the introduction of RSs and Context Aware Recommender Systems (CARS). These systems encompass many statistical techniques, multiple data sets and heuristics.

In addition, Relational Frame Theory (RFT) was a primary motivation behind this research. RFT is a modern behaviour-analytic account of human language and cognition which builds on Skinner's (1938) observations on reinforcement (Hayes *et al.*, 2001). It is based on the assumption that higher-cognitive functioning and language are composed of relational frames, which are patterns of generalised relational responding. That is, humans are capable of learning

through complex untrained (derived) relations and, learning occurs in relation to contextual cues via a history of multiple-exemplar training which is non-linear (see Barnes & Roche, 1996; Hayes & Hayes, 1989). The usefulness of RFT is dependent on its ability to predict how people will behave given generalised contexts and circumstances.

A natural extension to learning and cognition in humans, was to explore *learning* by Artificial Intelligence (AI) and machine learning, in particular RSs. Given that knowledge management is a rapidly developing area that is fast changing in order to meet the demands of more and more individuals using a myriad of on-line services and products, RSs must learn about preferences. Statistical techniques have been employed to predict and recommend the needs of such individuals with many companies, such as Amazon, Netflix, and TripAdvisor, *etc.*, gathering data about an individual's on-line behaviour to help their customers make decisions about future purchases, to recommend movies, places to visit, *etc.* The information gathered about on-line users is often accessible by large on-line retailers, *i.e.*, retained preferences, social media profiles and social tags[1]. This information is often utilised by RSs that assist us with our decision-making processes. The techniques that are used by RSs are wide and varied, coming from multiple disciplines to solve a common problem of recommendation (Jannach *et al.*, 2010). Sophisticated information systems are utilised in dialogue-based recommendation via natural language processes to evaluate reviews (Kang *et al.*, 2017). These systems must learn preferences, sometimes with little or no data available, and they must adapt to the user's needs over time. In addition, the user may also have a desire to *personalise* the information that they receive. This may be in the form of their spam filter, the news stories that are promoted on news sites or the clothes that on-line retailers suggest for them, in short, providing a tailor-made experience for the user. Therefore the user wishes the system to learn about their preferences just as much as businesses want their systems to learn the user's preferences.

In addition to learning preferences, most current RSs also incorporate context. Context is simply the environment and situation that the user is in when making an online decision. Context is very important to how we, as humans, learn. It is also possible for the user to have two conflicting preferences when the context is not the same. For example, the user may have a preference for *thriller* novels but will select a *romance* novel as it is a gift. Therefore it is

[1]Social tags are keywords generated by individuals on a platform that are used to describe and categorise an object, concept or idea.

important that context be accessed and incorporated into recommendations.

Finally, with the introduction of the General Data Protection Regulation (GDPR: 25th May, 2018), which is a comprehensive regulation that unifies data protection laws across all European Union member states, the way in which AI can learn is changing. As a result of the implementation of GDPR personal data is to be processed lawfully, fairly and in a transparent manner. Article 22 of GDPR is related to "Automated individual decision-making, including profiling."

> *"The data subject should have the right not to be subject to a decision, which may include a measure, evaluating personal aspects relating to him or her which is based solely on automated processing and which produces legal effects concerning him or her or similarly significantly affects him or her, such as automatic refusal of an online credit application or e-recruiting practices without any human intervention. Such processing includes 'profiling' that consists of any form of automated processing of personal data evaluating the personal aspects relating to a natural person, in particular to analyse or predict aspects concerning the data subject's performance at work, economic situation, health, personal preferences or interests, reliability or behaviour, location or movements, where it produces legal effects concerning him or her or similarly significantly affects him or her."*

Often RSs utilise large data stores that are used for exploration and experimentation. Such stores allow for the identification of relevant variables to produce accurate predictions. According to GDPR, data must be stored for a specified, limited amount of time. Some RSs are like black box systems in that they are ambiguous in relation to how the recommendation was made and offer no insight into the logic used to produce the recommendation. However, the concept of collaborative filtering, explained in Section 3.3.1, is easy to understand and can easily be explained to users even if the mathematics behind the concept may difficult to grasp. This is at odds with GDPR as the regulation requires models to be explainable, thus transparent. The last impact of GDPR on RSs is the certification of every party handling the data. As a result, there are specific requirements for procedures, documentation, certification, and reporting which binds every party. This affects RSs with respect to comparison and evaluate of models. RSs rely on data sharing across the internet, between multiple parties, and sometimes

in multiple countries. Therefore, it may become necessary to create a recommendation on a single source of information.

This regulation will help combat breeches in data privacy such as the Cambridge Analytica controversy that saw approximately 50 million Facebook users' data harvested without their consent and it was also discovered that Facebook had been collecting the data from users phone calls and text messages. In addition, Kosinski *et al.* (2013) demonstrated that digital records of behaviour, such as Facebook Likes, are easily accessible and can be used to automatically and accurately predict a range of highly sensitive personal attributes including: sexual orientation, ethnicity, religious and political views, personality traits, intelligence, happiness, use of addictive substances, parental separation, age, and gender. Kosinski *et al.* (2013) demonstrated that it was possible to correctly discriminate between homosexual and heterosexual men in 88% of cases, African Americans and Caucasian Americans in 95% of cases and between Democrat and Republican voters in 85% of cases.

1.2 Considerations for Proposed Approach

RSs are one of the most well-known Web Intelligence applications. Their primary aim is to assist in decision-making processes by alleviating the information overload problem. User preferences are typically collected via the users' implicit (online behaviours, *e.g.*, clicks, views and purchases) and explicit (*e.g.* ratings and reviews) feedback. This feedback can be collected from several different domains and in various formats. Information in relation to context may also be collected, *e.g*, the time of day, date, type of devise used, *etc.* Therefore, incorporating information that would not typically be considered for the purpose of a recommendation is ubiquitous in RSs. However, RSs face a number of challenges in relation to providing accurate recommendations in relation to vagueness and uncertainty in light of novelty. Displaying transparency has also become an issue for RSs which now is required by law in the European Union. Finally, producing a recommendation, for the majority of RSs, requires auxiliary information. One of the primary challenges for this thesis is to create an accurate algorithm that utilities only explicit feedback. Hence, the challenges faced in this thesis are:

- **Consideration 1: Incorporating Vagueness and Uncertainty**

RSs utilise a number of statistical techniques that combine different approaches in order to generate a recommendation. These systems produce a point-wise recommendation that represents the users' preferences based on previous ratings and/or previous activity. Such systems do not offer any interval type recommendation that would reflect vagueness and uncertainty. In this context, *uncertainty* is characterised as a state in which the individual has only imperfect knowledge and incomplete information but is still able to assign probability estimates to the possible outcomes of a decision. Whereas, *vagueness* refers to lack of precision, it is a state in which the individual does not have even the information to make subjective probability assessments. Both vagueness and uncertainty can be found when novelty is introduced. Novelty of recommendations is defined as the proportion of known and unknown content in the recommended list (Tweedale *et al.*, 2016). Novelty may be found in context, which may be measured in the various forms of physical (location), social (who you are with/who is the receiver of the item), interaction media (what device you are using) and modal (your current state of mind/mood), *etc.* Novelty can also be present when there is a new user/item where there is no previous information available. Given that there is vagueness and uncertainty within the system, within the users' preferences and within the context, a point-wise estimate cannot be reflective of the users' preferences. An interval recommendation is required to allow the user to explicitly see that the recommendation is reflective of the amount of information available to the system and the novelty of the context/item/user to the system.

- **Consideration 2: Transparency within the Recommender System**

  RSs also face challenging questions regarding how transparent they ought to be in relation to what information is collected and used when making a recommendation. Transparency, by definition, means that it (in this case a recommendation) is clear and easy to understand, *i.e.*, unambiguous or obvious. Most RSs are black boxes, in that, they offer the user no insight into the logic or justification for the recommendations (Sinha & Swearingen, 2002). According to Sinha & Swearingen (2002), a good algorithm *" that generates accurate recommendations is not enough to constitute a useful system from the users' perspective. The system needs to convey to the user its inner logic and why a particular recommendation is suitable for them"*. In addition, Jones & Pu (2008) found that some

users were curious to know more about how the system was achieving such good recommendations. According to Swearingen & Sinha (2002); Tintarev & Masthoff (2007); Pu *et al.* (2011) the role of transparency in a RS is very important, and the RS can convey its inner logic to the user via the explanation interface. Tintarev & Masthoff (2007) expanded on this stating that explanations showing how the system works make RSs more transparent and good explanations could help increase users satisfaction, making it quicker and easier for users to find what they want, and persuade them to try or purchase recommendations. In Pu *et al.* (2011), they argued that the contribution of explanation is not only to convince consumers to adopt recommendations, but also allow them to make more informed and accurate decisions about which recommendations to utilise, and eventually affect users satisfaction of RSs. Explanations for recommendations are provided by several RSs, *e.g.*, Amazon (Mooney & Roy, 2000).

In addition, according to Article 12 of the GDPR data subjects (*i.e.*, users) have the right to transparent information, communication and modalities.

> *"The principal of transparency requires that any information addressed to the public or to the data subject be concise, easily accessible and easy to understand, and that clear and plain language and, additionally, where appropriate, visualisation be used. Such information could be provided in electronic form, for example, when addressed to the public, through a website. This is of particular relevance in situations where the proliferation of actors and the technological complexity of practice make it difficult for the data subject to know and understand whether, by whom and for what purpose personal data relating to him or her are being collected, such as in the case of online advertising. Given that children merit specific protection, any information and communication, where processing is addressed to a child, should be in such a clear and plain language that the child can easily understand."*

Therefore, even though transparency has been considered an important aspect of RSs, it is now a necessary aspect under GDPR. By allowing a system to produce an interval recommendation, the user is given information via the width of the interval which, over a number of interactions, demonstrates the more explicit information the user provides

the narrower the interval becomes. Hence, it is obvious to the user how the interval is narrowing thus, providing some transparency in how explicit feedback from the user relates to a recommendation. This allows users to meaningfully revise their input in order to improve recommendations, producing an efficient interaction with the system.

- **Consideration 3: Single Source Data**

  According to Google's chief economist, Hal Varian *"we really do have essentially free and ubiquitous data"* (Champkin, 2011). The information gathered about on-line users is often accessible by large on-line retailers, utilising a range of statistical techniques to produce a recommendation. Nowadays, hybrid RSs are common place. A hybrid system combines two or more recommendation approaches to leverage the strengths of each individual approach and to address data sparsity and cold-start problems. To achieve this type of recommendation, ratings, features and demographic information are utilised, accessing multiple social media accounts.

  In light of new legislation, users of RSs have more control over their data - the data is theirs and they have the right to be forgotten. As a result, the *"ubiquitous data"* may not be so ubiquitous. While many RSs utilise such auxiliary information, the proposed algorithm will use only the original data. Therefore the third consideration for the proposed approach is to produce an algorithm, that not only incorporates vagueness and uncertainty whilst being easy to understand, but to produce one that is accurate whilst only incorporating data from a single source. That is not to say that our proposed methodology is a RS in of itself, but simply that this is a first attempt at developing an interval recommendation.

1.3 Contribution

This thesis investigates how the UI algorithm can contribute to RSs by allowing such systems to incorporate vagueness and uncertainty and to be more transparent. This research contributes to the body of knowledge by:

Interval Based Recommendations

The primary contribution of this thesis is to incorporate vagueness and uncertainty into recom-

mendations. Existing RSs provide point-wise estimates regardless of the amount of novelty. As accurate as current RSs have become a point-wise recommendation cannot address vagueness and uncertainty. This can only be achieved via an interval approach based on imprecise probability. The concept of NPI and its specialisation, Nonparametric Predictive Utility Interval (NPUI), are utilised to update recommendations in light of new information. This ensures, that when there are few or no observations (*e.g.*, explicit ratings), the interval is wide only becoming narrower as more feedback is observed.

1.4 Thesis Structure

The following outlines the structure of the Thesis.

- **Key Concepts**

  In Chapter 2 the key concepts utilised in the proposed methodology are introduced. These include the concepts of probability, utility, vagueness and uncertainty. The notion of exchangeability, which is a key assumption of the UI approach, is outlined, along with Nonparametric Predictive Inference (NPI), which also builds on exchangeability. One of the main challenges of RSs is missingness. Therefore this chapter defines the patterns of missing values. Next, as the UI algorithm utilises Pearson's correlation coefficient, a brief overview of this statistical technique is necessary, again with emphasis on how it relates to exchangeability.

- **Literature Review**

  This chapter focuses on the concepts from the literature that provide relevant insights for the topics considered in this thesis. The concept of Relational Frame Theory from behavioural and cognitive psychology is briefly introduced, as are Bayesian Networks from statistics. This leads to measuring uncertainty, decision-making under uncertainty and heuristics in decision-making. The concept of vagueness is expanded on and methods of incorporating vagueness and uncertainty are discussed. Finally, the concept of Recommender Systems (RSs) is introduced, with a brief introduction to the various types of RSs.

- **Proposed Methodology**

  In Chapter 4 an existing technique, Robust Bayesian Correlation Learning (RBCL) (Troffaes *et al.*, 2014), is introduced. This chapter outlines the RBCL technique and highlights its limitations. Next, it describes in detail the proposed UI algorithm, outlining its various flavours including $UI_{Untrans}$, UI_{Abs}, UI_{Sq}, and UI_{Rt}. The chapter concludes by explaining how this solution addresses the issue of vagueness and uncertainty, which arises as a result of novelty.

- **Evaluation on Simulated Data**

  Chapter 5 evaluates the performance of the UI algorithm, in each of the four UI approaches, on simulated data. It first describes the methods used to simulate data. Next, the RBCL and UI algorithms are applied to both complete and sparse data sets, evaluated by the interval percentage that captures the true correlation coefficient and the absolute minimum distance from the true correlation coefficient. Finally, these algorithms are applied to and evaluated on various distributions.

- **Evaluation on Real-World Data**

  In Chapter 6 the best performing UI algorithm is applied to the MovieLens data set, providing a real-world example of its application. It first describes the MovieLens dataset and its contribution to RSs. Next, as with the previous chapter, the performance of UI and RBCL is evaluated and presented.

- **Conclusion**

  Finally, Chapter 7 summaries the thesis by discussing the strengths and weaknesses of the UI algorithm and identifying a number of potential areas for future work.

1.5 Chapter Summary

Vagueness and uncertainty are present in decision-making related to novel context/user/item. They arise due to a lack of or imprecise information and/or the inability to attach a subjective probability assessment. Vagueness and uncertainty may give rise to inaccurate recommendations. NPI has successfully incorporated vagueness and uncertainty into decision-making processes by utilising imprecise probabilities and exchangeability. Therefore, developing a

specialised version of NPI would allow vagueness and uncertainty to be integrated into the decision-making process of RSs.

Lack of transparency has been found to be an issue within RSs. Transparency is not only required by law, but users want more transparency on how recommendations are made and how their data is utilised. Transparency engages the user, allowing for more effective interaction between the user and the system.

This thesis presents the UI algorithm, a novel interval approach for incorporating vagueness and uncertainty into recommendations. The UI algorithm enables the system, to not only provide an interval recommendation, but also provide transparency in relation to feedback. This algorithm builds on existing statistical techniques that incorporate imprecise probabilities that are predictive in nature. It also utilises the simple, yet commonly used, statistical technique of Pearson's correlation coefficient found in RSs. The following chapters describe how this algorithm achieves combining various statistical approaches, stemming from the multi disciplinary theories of decision-making and learning, to produce an interval recommendation.

2

Key Concepts

In this chapter the concepts used in the proposed methodology are defined. These can be considered as two sections. The first section defines probability, both precise and imprecise, utility, vagueness and uncertainty. These concepts relate to decision theory and decision-making processes. The second section defines exchangeability, missingness and correlation; all of which are incorporated into the proposed Uncertainty Interval (UI) algorithm.

2.1 Probability

In this section there is a brief outline to the theory of probability. This introduction is focused towards understanding the key concepts that will follow. First, some basic definitions from measure theory are given. Next the concept of probability is introduced. The definitions outlined in this section are taken from Schervish (2012).

A *measure* is a way of assigning numerical values to the *"sizes"* of sets.

Definition 2.1 Field: A nonempty collection of subsets $\mathcal{A}$ of a set Ω is called a *field* if

- $A \in \mathcal{A}$ implies $A^c \in \mathcal{A}$,
- $A_1, A_2 \in \mathcal{A}$ implies $A_1 \cup A_2 \in \mathcal{A}$.

A field $\mathcal{A}$ is called a σ-field if $\{A_i\}_{i=1}^{\infty}$ implies $\cup_{i=1}^{\infty} A_i \in \mathcal{A}$. Therefore, if $\mathcal{A}$ is a σ-field, then it is nonempty, closed under compliments and closed under countable unions.

Definition 2.2 Borel σ-field: Let $\mathcal{C}$ be a collection of intervals in $\mathbb{R}$. The smallest σ-field containing $\mathcal{C}$ is called the *Borel σ-field*. More formally, let $\mathcal{B}$ be the smallest σ-field that contains all open subintervals in Ω. This σ-field is commonly referred to as the *Borel* σ-algebra $\mathcal{B}$ and accordingly each event in $\mathcal{B}$ is called a Borel set. As $\mathcal{B}$ is a σ-field, it is closed under complement, countable unions, and countable intersections and contains many subsets other than open intervals.

Definition 2.3 Measurable Space: A pair $(\Omega, \mathcal{A})$, where Ω is a set and $\mathcal{A}$ is a σ-field is called a *measurable space*.

Definition 2.4 Measure: A function $P : \mathcal{A} \to [0, \infty]$ is called a *measure* if

- $P(\emptyset) = 0$,
- $\{A\}_{i=1}^{\infty}$ mutually disjoint implies $P(\cup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$.

Definition 2.5 Measure Space: If P is a measure, the triplet $(\Omega, \mathcal{A}, P)$ is called a *measure space*.

Definition 2.6 Measurable Function: Suppose that Ω_1 is a set with a σ-field $\mathcal{A}_1$ of subsets, and let Ω_2 be another set with a σ-field $\mathcal{A}_2$ of subsets. Suppose that $f : \Omega_1 \to \Omega_2$ is a function. We say that f is *measurable* if for every $A \in \mathcal{A}_2$, $f^{-1}(A) \in \mathcal{A}_1$.

Definition 2.7 Probability Space: A *probability space* is a measure space $(\Omega, \mathcal{A}, P)$ with $P(\Omega) = 1$. Each element of $\mathcal{A}$ is called an *event*. If $(\Omega, \mathcal{A}, P)$ is a probability space, $(\mathcal{X}, \mathcal{B})$ is a measurable space, and $X : \Omega \to \mathcal{X}$, then X is called a *random quantity*. If $\mathcal{X} = \mathbb{R}$ and $\mathcal{B}$ is a Borel σ-field, then X is called a *random variable*. Let Pr be the probability measure induced on $(\mathcal{X}, \mathcal{B})$ by X from P. The probability measure is called the *distribution of* X. The distribution of X is said to be *discrete* if there exists a countable set $A \subseteq \mathcal{X}$ such that $Pr(A) = 1$. The distribution of $\mathcal{X}$ is *continuous* if $Pr(\{x\}) = 0$ for all $x \in \mathcal{X}$.

Coherence in precise probability: Kolmogorov (1950) axioms of probability state:

- $Pr(\Omega) = 1$
- $Pr(A) \geq 0$ for all $A \in \mathcal{A}$

- Countable additivity: If $A_1, A_2, \ldots \in \mathcal{A}$ are pairwise disjoint (*i.e.*, $A_i \bigcap A_j = \emptyset$, for all $i \neq j$), then $Pr(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} Pr(A_i)$

Any collection of probabilities failing to meet the three Kolmogorov axioms leads to incoherence and is at risk of becoming a Dutch Book argument (Kemeny, 1955). This argument is a system of bets that guarantees a net loss, *i.e.*, an individual takes a gamble that guarantees a loss irrespective of what outcome occurs.

Definition 2.8 Probability Mass Function: The probability mass function (pmf) is the probability function which is defined for the discrete random variable X. It is denoted by $f_X(x)$.

$$f_X(x) = Pr(X = x) \geq 0 \text{ for all } x$$

$$\sum_{x \in \Omega} f_X(x) = 1$$

Definition 2.9 Probability Density Function: The probability density function (pdf) is the probability function which is defined for the continuous random variable X. It is denoted by $f_X(x)$.

$$f_X(x) = Pr(x \in \Omega) \geq 0 \text{ for all } x$$

$$\int_{-\infty}^{\infty} f_X(x)dx = 1$$

Definition 2.10 Expectation: The expectation of a random variable is a weighted average of the possible values that X can take, each value being weighted according to the probability of that event occurring. The expectation is defined differently for continuous and discrete random variables. Let X be a *discrete* random variable with probability mass function $f_X(x)$. The expected value of X is:

$$\mathbb{E}[X] = \sum_{x \in \Omega} x f_X(x) = \sum_{x \in \Omega} x Pr(x)$$

Let X be a *continuous* random variable with probability density function $f_X(x)$. The expected value of X is:

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f_X(x)dx.$$

Probability as an expectation: Let A be any event. We can write $Pr(A)$ as an expectation

(prevision), as follows. Define the *indicator function*:

$$I_A = \begin{cases} 1 & \text{if event } A \text{ occurs} \\ 0 & \text{otherwise} \end{cases}$$

Then I_A is a *random variable* and

$$\begin{aligned} \mathbb{E}(I_A) &= \sum_{x=0}^{1} xPr(I_A = x) \\ &= 0 \times Pr(I_A = 0) + 1 \times Pr(I_A = 1) \\ &= Pr(I_A = 1) \\ &= Pr(A) \end{aligned}$$

2.2 Previsions and Imprecise Probability

The term *imprecise probabilities* categorises a variety of mathematical representations such as convex sets of probabilities (Good, 1962; Levi, 1974), lower previsions (Walley, 1991) and interval probabilities (Kuznetsov, 1991; Weichselberger, 2000). In this section we will focus on imprecise probabilities as modelled by lower previsions. While probabilities are defined on event spaces, previsions exist on the space of real-valued gambles. The definitions in this section are taken from Augustin *et al.* (2014)

Definition 2.11 Gamble: A *gamble* f on a random variable ω is defined as a mapping $f : \Omega \to \mathbb{R}$. It represents an uncertain payoff (reward) $f(\omega)$. This reward is expressed in units of some linear utility scale, *e.g.*, if we receive double the payoff, we consider this to be twice as good (or bad).

Definition 2.12 Linear space of gambles: Let $\mathcal{L}(\mathcal{A})$ denote the set of all gambles, then $\mathcal{L}(\mathcal{A})$ is a *linear lattice* with respect to point-wise addition, the point-wise scalar multiplication and the point-wise ordering, *i.e.*, every pair of elements has a least upper bound and a greatest

lower bound.

$$(f+g)(\omega) = f(\omega) + g(\omega), \text{and}$$
$$(\lambda f)(\omega) = \lambda f(\omega)$$

for any pair of gambles f and g on $\mathcal{A}$, any real λ and all $\omega \in \Omega$, and

$$f \geq g \text{ if for all } \omega \in \Omega, f(\omega) \leq g(\omega)$$

The supremum $f \vee g$ of two gambles g and f on $\mathcal{A}$ is defined by their point-wise maximum for all $\omega \in \Omega$.

$$f \vee g = \max\{f(\omega), g(\omega)\}$$

The infimum $f \wedge g$ of two gambles g and f on $\mathcal{A}$ is defined by their point-wise minimum for all $\omega \in \Omega$.

$$f \wedge g = \min\{f(\omega), g(\omega)\}$$

Definition 2.13 $I_A(\omega)$**:** If A is an event ($A \subseteq \Omega$), we can associate a $\{0,1\}$-valued gamble I_A, where I_A is the *indicator* of A. This gamble is defined as:

$$I_A(\omega) = \begin{cases} 1 & \text{if event } A \text{ occurs} \\ 0 & \text{otherwise} \end{cases}$$

Definition 2.14 Previsions: The *lower prevision* of a gamble f, denoted $\underline{P}(f)$, represents an individual's supremum acceptable buying price for f. Similarly, an individual's *upper prevision*, $\overline{P}(f)$, is his or her infinmum acceptable selling price for f. Therefore, this individual is willing to buy a gamble f for all prices $s < \underline{P}(f)$ and sell f for all prices $t > \overline{P}(f)$. For prices $\underline{P}(f) \leq p \leq \overline{P}(f)$ the individual is undecided.

If an individual assesses a lower prevision $\underline{P}(f)$ for all gambles f on some subset $\mathcal{K}$ of the set $\mathcal{L}(\Omega)$ (the set of all gambles), then this defines a real functional mapping $\underline{P} : \mathcal{K} \to \mathbb{R}$, called a lower prevision with *domain* $\mathcal{K}$

$\underline{P}(f)$ represents a so-called *fair price* for a gamble f. As buying the gamble f for a price s is the same as selling the gamble $-f$ for the price $-s$, the lower and upper previsions are

conjugate functions. That is to say: $\underline{P}(f) = -\overline{P}(-f)$ for any gamble f. Therefore, only lower previsions need to be considered as upper previsions can immediately be derived from them.

$\underline{P}(f)$ is superadditive, *i.e.*, $\underline{P}(-f) + \underline{P}(f) \leq 0$, hence $\underline{P}(f) \leq \overline{P}(f)$. In addition, since $\underline{P}(f)$ is superadditive, $\overline{P}(f)$ is subadditive.

Definition 2.15 Previsions as probabilities: The lower probability $\underline{Pr}(A)$ of an event $A \subseteq \Omega$ is defined as the lower prevision of its indicator $I_A(\omega) : \underline{Pr}(A) = \underline{P}(I_A)$. For the upper probability $\overline{Pr}(A)$ of A, we similarly have that $\overline{Pr}(A) = \overline{P}(I_A)$.

Definition 2.16 Coherent Lower Prevision: A lower prevision is said to be coherent if it avoids sure loss; this is the most important rationality criterion for lower previsions. A lower prevision is coherent on a linear space of bounded gambles, $\mathcal{K}$, if and only if it satisfies the following conditions

- $\underline{P}(f) \geq \inf_{\omega \in \Omega} f(\omega)$ (accepting a sure gain).
- $\underline{P}(\lambda f) = \lambda \underline{P}(f)$, for gambles f and $g \in \mathcal{K}$, $\lambda > 0$ (non-negative homogeneity).
- $\underline{P}(f + g) \geq \underline{P}(f) + \underline{P}(g)$ (superadditivity).

Definition 2.17 Precise Prevision: If the lower prevision $\underline{P}(f)$ and the upper prevision $\overline{P}(f)$ for a gamble f happen to coincide, then the value $P(f) = \underline{P}(f) = \overline{P}(f)$ is called a *precise* prevision.

Definition 2.18 Linear Prevision: A real-valued functional P defined on a set of gambles $\mathcal{K}$ is a *linear prevision* if and only if for all natural numbers $m, n \geq 0$ and all gambles $f_1, \ldots, f_m, g_1, \ldots g_m$ in the domain of $\mathcal{K}$,

$$\sup_{\omega \in \Omega} \Big[\sum_{i=1}^{m} [f_i(\omega) - P(f_i)] - \sum_{i=1}^{n} [g_i(\omega) - P(g_i)] \Big] \geq 0$$

2.3 Utility

Utility is the name given to the concept of subjective value of *usefulness* that a consumer gains from any good or service beyond the explicit monetary value of those goods or services.

Definition 2.19 Utility Function: A utility function, $U : R \to \mathbb{R}$ is a measure of an individual's subjective preferences from returns $r \in R$.

In order to define *expected utility*, some notion from normative decision theory must be introduced. An individual must choose a *decision* for a non-empty, finite set of admissible decisions $\mathcal{D} = \{d_1, \ldots, d_n\}$. The consequence of every decision depends on the true, but unknown *state of nature* $\theta \in \Theta = \{\theta_1, \ldots, \theta_m\}$. The corresponding outcome is evaluated by the utility function $u : (\mathcal{D} \times \Theta) \to \mathbb{R}$. The utility resulting from making decision d_i and the occurrence of θ_j is denoted $u(d_i, \theta_j)$.

Definition 2.20 Preference Relations: *Preference relations* can be defined between outcomes. Let d_1 and d_2 be two possible decisions then:

- $d_1 \succ d_2$ implies that decision d_1 is strictly preferred to d_2.
- $d_1 \sim d_2$ implies indifference; the individual is indifferent between d_1 and d_2.
- $d_1 \succeq d_2$ implies d_1 is deemed at least as preferable as decision d_2.

Preference of a rational individual (agent) must obey the axioms of rationality:

- **Completeness:** $(d_1 \succeq d_2)$, $(d_2 \succeq d_1)$ or $(d_1 \sim d_2)$, *i.e.*, an individual has well defined preferences and that all relevant decisions can be definitively compared to one another.
- **Transitivity**[1]: $\forall\ d_1, d_2, d_3$, if $(d_1 \succeq d_2)$ and $(d_2 \succeq d_3)$ then $(d_1 \succeq d_3)$. This ensures that there are no cycles of exact preference.
- **Continuity:**[2]: If $d_1 \succeq d_2 \succeq d_3$ then $\exists$ a probability $p \in [0,1]$ such that $pd_1 +_g (1-p)d_3 \sim d_2$ *i.e.*, there exists a probability p making an individual indifferent between a decision between the best and worst outcome (with probabilities p and $1-p$ respectively) and a guaranteed intermediate outcome. Note that the operator $+_g$ is used to denote a gamble between two outcomes rather than standard addition.
- **Independence:** $\forall\ d_1, d_2, d_3$ and $p \in [0,1]$, $d_1 \succeq d_2 \Leftrightarrow pd_1 +_g (1-p)d_3 \succeq pd_2 +_g (1-p)d_3$ *i.e.*, preference is invariant to the introduction of independent alternatives.

Definition 2.21 Expected Utility: The expected utility of a choice decision depends on the probabilities and utilities of its possible rewards and is defined for a descrete distribution as:

$$\mathbb{E}[u(d_i)] = \sum_{j=1}^{m} u(d_i, \theta_j) Pr(\theta_j)$$

[1]Without transitivity, *i.e.*, coherence, we end up with a money pump situation (Lindley, 1991).

[2]An alternative axiom for continuity is the *Archimedean property* which states if $d_1 \succ d_2 \succ d_3$ then $\exists\ \alpha, \beta \in (0,1)$ such that $\alpha d_1 +_g (1-\alpha)d_3 \succ d_2 \succ \beta d_1 +_g (1-\beta)d_3$.

The utility function u is not necessarily linear and loss is expressed as negative utility but may not be its mirror image. In addition, a utility is often re-scaled to the closed interval $[0, 1]$ with 0 being the least favoured outcome and 1 being the most optimal outcome. The product of a rewards' probability and utility indicates the subjective value to the expectation of that reward. From the above axioms it is possible to show the existence of a unique utility function such that:

- $\forall\ d_1, d_2, u(d_1) \geq u(d_2) \Leftrightarrow d_1 \succeq d_2$
- $\forall\ d_1, d_2,$ and $p \in (0, 1),\ u(p(d_1) +_g (1-p)d_2) = pu(d_1) + (1-p)u(d_2)$

Definition 2.22 Maximum Expected Utility: The optimal decision, d_{opt}, which *maximises* expected utility is calculated as:

$$d_{opt} = \arg\max_i \mathbb{E}[u(d_i)]$$

2.4 Uncertainty

There can be many sources of uncertainty, but these can be categorised by the character of uncertainties as either aleatory or epistemic. The word *aleatory* derives from the Latin *alea*, which refers to the rolling of dice. Therefore, aleatory uncertainty is uncertainty that is presumed to be the intrinsic randomness of a phenomenon; it refers to variation which is inherent to a given system, typically as a result of the random nature of model inputs. Aleatory uncertainties are typically modelled as random variables described by probability distributions, where decision makers typically make assumptions about the distribution's descriptive statistics (*i.e.*, its mean and variance). This type of uncertainty is considered *objective*. The word *epistemic* derives from the Greek $\epsilon\pi\iota\sigma\tau\eta\mu$ (episteme), which refers to knowledge. Therefore, epistemic uncertainty is uncertainty that is presumed as being caused by lack of knowledge (or data). Epistemic uncertainty often becomes an issue when expert opinion is required to solve a problem; it is also referred to as reducible uncertainty and subjective uncertainty.

Definition 2.23 Uncertainty: A decision d_i, with outcome space $\mathcal{R}$ is said to be uncertain if $Pr(d_i)$ is defined for all d_i, *i.e.*, it can be assigned a θ_j, and $Pr(d_i)$<1 for all d_i.

2.5 Vagueness

Vagueness, which Walley (1991) terms imprecision, is a property related to the content of a statement concerning a questionable feature of the actual problem; either more than one world or no world is compatible with the available information. It is essentially a property of the given information, with partial ignorance and conflicting information being common in practice. It refers to a lack of clarity in meaning: when information is vague it is difficult to form any interpretation at the desired level of specificity. Vagueness, in philosophical terms, is defined as: *The characteristic of words or phrases whose meaning is not determined with precision. Use of one or more vague terms typically renders it impossible to establish the truth or falsity of the sentences in which they appear.*

Definition 2.24 Vagueness: A decision d_i, with outcome space $\mathcal{R}$ is said to be vague if $Pr(d_i)$ cannot be associated with a certain singleton θ_j.

2.6 Exchangeability

Exchangeability is a fundamental assumption in Bayesian analysis and reflects how variables do not depend on their indices even though they may be dependent among themselves. Therefore, exchangeability is about symmetry in a way that does not require independence. Hence, one can swap around, or reorder, variables in a sequence without changing their joint distribution.

Definition 2.25 Infinitely exchangeable: Consider a process $X = (X_1, X_2, \ldots)$, taking two values. A probability distribution Pr for the processes is said to be an *infinitely exchangeable* sequence of random variables if, for any k, the joint probability $Pr(x_1, \ldots, x_k)$ is invariant to permutation of the the indices, *i.e.*, for any permutation π, $Pr(X_1, \ldots, X_k) = Pr(X_{\pi(1)}, \ldots, X_{\pi(k)})$. Therefore, a sequence is infinitely exchangeable if and only if every finite subsequence is exchangeable.

Theorem 1 De Finetti's (1931) representation theorem:
Let $(\Omega, \mathcal{A}, P)$ be a probability space, and let $(\mathcal{X}, \mathcal{B})$ be a Borel space. For each i, let $X_i : \Omega \to \mathcal{X}$ be measurable. The sequence $X_{i\infty}^{i=1}$ is exchangeable if and only if there is a random probability

measure μ on $(\mathcal{X}, \mathcal{B})$ such that e $X_{i\infty}^{i=1}$ are i.i.d. with distribution μ. Furthermore, if the sequence is exchangeable, then the distribution of μ is unique.

In this theorem, de Finetti demonstrates if $X_1, X_2, \ldots$ are infinity exchangeable then the joint probability $f_X(x_1, x_2, \ldots)$ has a representation as a mixture. He demonstrates that with probability 1 the proportion of each type tends to be a limiting frequency and that there is a unique measure μ on the unit interval, *i.e.* a probability distribution, such that for any give n, and a sequence $x_1, \ldots, x_n$,

$$Pr(x_1, x_2, \ldots) = \int_{\Omega_\theta} x^a (1-x)^b \mu d(x),$$

where a is the number of x_i of type 1 among $x_1, \ldots, x_n$ and b is the number of type 2.

Although de Finetti's theorem does not hold exactly for finite sequences, it does hold approximately for sufficiently large finite sequence. Exchangeability involves complete symmetry in beliefs. Often such beliefs are not warranted across all observables, but are reasonable for subsets. In 1938 de Finetti broadened this concept of exchangeability to *partial exchangeability.*

Definition 2.26 Partially exchangeable: Consider the case of two sets of observations $X_1, X_2, \ldots; Y_1, Y_2, \ldots$. Each set of observations takes two values. The covariate is judged as potentially meaningful, the X_i's are judged exchangeable between themselves and the Y_i's are judged exchangeable between themselves. Mathematically, the joint law must be invariant under permutations within the X's and Y's:

$$Pr(X_1, \ldots, X_n; Y_1, \ldots, Y_m) = Pr(X_{\pi(1)}, \ldots, X_{\pi(n)}; Y_{\sigma(1)}, \ldots, Y_{\sigma(m)})$$

This must hold for all n and m, and permutations π and σ. They can be generalised to more than two cases of observations.

Here de Finetti proved that for an infinite process $\{X_i, Y_j\}$ partially exchangeable implies

$$\frac{X_1 + \ldots X_n}{n}, \frac{Y_1 + \ldots + Y_m}{m} \to (\mu_X, \mu_Y)$$

Definition 2.27 Finitely exchangeable: Finite exchangeability is often more natural than infinite exchangeability and is defined as

$X = (X_1, \ldots, X_n)$ is said to be a *finitely exchangeable* if each of the $n!$ permutations of $X_1, \ldots, X_n$ has the same n-dimensional joint probability distribution, *i.e.*, $Pr(X_1, \ldots, X_n) = Pr(X_{\pi(1)}, \ldots, X_{\pi(n)})$, where $X_{\pi(1)}, \ldots, X_{\pi(n)}$ is any of the $n!$ permutations of $X_1, \ldots, X_n$. Note, for a finite sequence of exchangeable random variables $\rho \geq 0$.

2.7 Correlation

Correlation measures the degree and direction of linear association between quantitative variables.

Definition 2.28 Pearson's correlation coefficient: *Pearson's correlation coefficient*, ρ, is the ratio between the covariance and the product of the standard deviations of both sets of random variables. ρ has the following properties:

- $-1 \leq \rho \leq 1$.
- $\rho>0$ when there is a positive linear association, $\rho<0$ when there is a negative linear association.
- The sign of the linear correlation coefficient is shared by the covariance. A covariance is a measure of how much two variables change together. $\rho = 1$ when there is a perfect linear association, $\rho = -1$ when there is a perfect negative linear association.
- The value of ρ is a measure of the extent to which two variables are *linearly* related. $\rho = 0$ does not imply that the two variables have no association, just that they have no linear association.
- Correlation is unit free, *i.e.*, the units of measurement of the two variables do not matter.
- Correlation does not imply causation.

Pearson's correlation coefficient, ρ, is calculated as:

$$\rho = \rho_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

$$= \frac{Cov(x, y)}{\sigma_x \sigma_y}$$

Where σ_x and σ_y represents the standard deviation of x and y respectively. Therefore, we can see that the correlation coefficient is simply the covariance normalised.

2.8 Missingness

Missing data is defined as data that are missing for some (but not all) variables and for some (but not all) cases. Little & Rubin (1987, 2014) offered an explication of various missing patterns; when considering a single variable x, values of x can be missing:

- randomly;
- below some cut value of x, or;
- subject to some form of probabilistic censoring proportional to the value of X.

The type of missingness is likely to bias sample estimates of the population mean of x, *i.e.*, when data is missing at random the bias is negligible but not necessarily negligible when data is missing systematically. There are even more possible missing data patterns when considering multivariate analysis with nonrandom missingness creating bias for both variable means and covariance estimates (Rubin, 1976).

Definition 2.29 Missing completely at random (MCAR): Missing data are *MCAR* when the probability of missing data on a variable is unrelated to any other measured variable and is unrelated to the variable with missing values itself. In other words the missingness on the variable is completely unsystematic. There are two criteria that must be met for missingness to be described as MCAR - the data must be classified as *missing at random* (MAR) and *observed at random* (OAR). MAR is described below and OAR occurs if the missingness pattern does not depend on the values of the data that are observed.

Definition 2.30 Missing at random (MAR): Missing data are *MAR* when the probability of missing data on a variable is related to some other measured variable in the model, but not to the value of the variable with missing values itself. In the context of a movie rating, if users like the film "Frozen" (animation), they typically do not submit ratings for the movie "The Conjuring" (horror), in which the missingness of the movie "The Conjuring" depends on the observed ratings for the movie "Frozen".

Definition 2.31 Not missing at random (NMAR): Data are *NMAR* when the missing values on a variable are related to the values of that variable itself, even after controlling for other variables. For example, if a user does not like the movie "The Conjuring", the user will not rate it.

2.9 Chapter Summary

This chapter has outlined the formal definitions of the concepts that are utilised in this thesis. The first section was concerned with key elements related to beliefs and preferences via probabilities and utilities. The latter part defined the underlying properties of the proposed methodology, UI approach, which incorporates correlation coefficients and missingness. The UI approach makes the assumption of partial exchangeability, allowing for future events to be predicted from past observations.

The next chapter introduces the themes from the literature that have influenced the UI approach. These influences come from the elements of psychology and statistics that focus on learning, forming relations and imprecision.

3

Literature Review

In this chapter we introduce the concepts from the literature that provide relevant information and insights for the topics considered in this thesis. As with the previous chapter, there are two distinct themes: firstly, the development of concepts related to precise and imprecise probabilities and; secondly, the development of Recommender Systems (RSs). In the first section, a brief history of modelling uncertainty is discussed, outlining how precise and imprecise probabilities are incorporated into decision-making processes. This section also describes how heuristics play a role in decision-making and the various types of heuristics utilised by individuals making decisions are described. The next section outlines the various types of RSs and allows for a brief history of recommendation and preference techniques. This leads to the role of heuristics in recommendations.

3.1 Beginnings

From the literature in psychology, it is argued that humans learn novel behaviour without direct reinforcement of that behaviour, where a reinforcer is anything that increases the frequency of a behaviour; this is known as stimulus equivalence (Fields *et al.*, 1993). It is the fundamental building block of Relational Frame Theory (RFT). RFT is concerned with all possible derived relations and as such examines derived relational responding which involves the ability to relate stimuli in a variety of ways even though one has never been trained/reinforced for relating those stimuli in those specific ways (Blackledge, 2003). Not only does RFT consider reflexivity (*e.g.* A to A, B to B, *etc.*), symmetry (*e.g.* AB matching), transitivity (*e.g.* untrained AC

matching), and coordination (same/similar), it encompasses non-coordination (*e.g.* difference, oppositeness), evaluation (*e.g.* better than, less than), causal (*e.g.* if-then), temporal (*e.g.* before-after) and many other types of relations. In addition, RFT distinguishes between two main types of relational responding, namely non-arbitrary relations and arbitrary relations. Non-arbitrary relations involve physical properties, formal dimensions, whereas arbitrary relations are abstract concepts and dimensions such as love, justice, worth/value. For example, an elephant is larger than a mouse, but we can also state that a 10¢ coin is larger than a 5¢ even though it is smaller in size. The relation is arbitrarily applicable since they are manipulated based on social convention (Quiñones *et al.*, 2000). In arbitrarily applicable derived relational responding, mutually entailed relational responses (*i.e.* if a stimulus A is related to stimulus B in a specific way, then B is related to A in a complimentary way) are controlled by contextual cues rather than physical properties of the stimuli being related. RFT accounts for a human's ability to derive additional relationships between stimuli given limited information. For example, given the following four trained relations:

$$\text{Trained Relations} = \begin{cases} \text{Coin } A \text{ is worth less than Coin } B \\ \text{Coin } E \text{ is worth less than Coin } A \\ \text{Coin } E \text{ is worth more than Coin } C \\ \text{Coin } C \text{ is worth more than Coin } D \end{cases}$$

From these four trained relations, 16 relations are derived. These trained and derived relations are easier to understand as a graphical representation as in Fig. 3.1.1.

This interpretation of learning via relations is reliant on context, and by considering context, RSs have been able to produce more relevant and accurate recommendations (these systems are discussed in **??**). Empirical research on RFT has shown how this approach to language and cognition allows for the exploration of implicit beliefs via the Implicit Relational Aptitude Procedure (IRAP) (Barnes-Holmes *et al.*, 2006; Nicholson *et al.*, 2013). Therefore, this theory is not only built on relational learning and context but also evaluates implicit (unconscious knowledge) belief systems. Implicit beliefs (and biases) are brought about unconsciously by typical

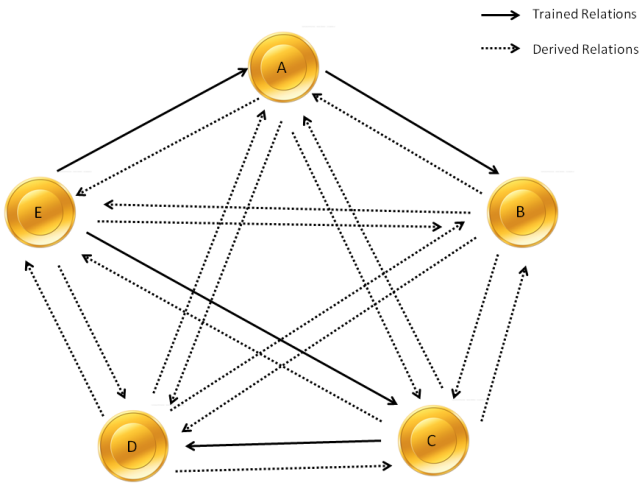

Figure 3.1.1: Trained and Derived Relations.

conditioned responses: *"the science of implicit cognition suggests that actors do not always have conscious, intentional control over the processes of social perception, impression formation, and judgement that motivate their actions"* (Greenwald & Krieger, 2006). In addition, Kovács *et al.* (2014) found that on-line interactions are implicit and automatic. This suggests that relational learning is objective, as derived relational learning requires historical learning that is generalised to novel contexts/items/experiences. However, it may be argued that, given language is an arbitrary construct and prone to subjectivity (and even vagueness), judgements based on language are also subjective. Psychologists have utilised the IRAP to develop evaluations of implicit beliefs in order to modify maladaptive behaviours (Arch & Craske, 2008), explore and understand bias (Hussey *et al.*, 2015), as well as to develop methods to improve learning (Tarbox *et al.*, 2013).

3.2 Measuring Uncertainty

In general, uncertainty occurs whenever information pertaining to a situation is incomplete, contradictory or fluctuating. Probability theory provides a well-founded mathematical framework for quantifying uncertainty. The mathematical theory of classical probability arises from gambling and games of chance which came under examination in the 17th century by Pascal and de Fermat (Edwards, 1982). This approach consists of repeating a game of chance a large number of times under the same conditions. Therefore, the probability of winning is then ap-

proximately equal to the proportion of wins in the repeats. Thus, is named a *frequentist* or *objective* probability.

De Finetti (1931) stated *"probability does not exist"*, by which he meant probability does not exist objectively but that only *subjective* probabilities exist, *i.e.*, the *degree of belief* in the occurrence of an event attributed by a given person at a given instant and with a given set of information. Therefore, a *subjective* interpretation of probability asserts that there is no such thing as a true or objective probability, but rather probabilities are reflections of a *rational* individual's beliefs, and involves personal judgements (Ramsey, 1924; De Finetti, 1974; Lindley, 1982).

In this thesis uncertainty within the RS is assumed to arise from a lack of information but it may also be the result of certain fluctuations which are dependent on the context, *e.g.*, depending on the particular situation, mood, media literacy, and other biasing factors such as the interface (Jasberg & Sizov, 2018). Therefore a Bayesian concept of uncertainty is utilised. This bodes well with subjective utility, as defined in Section 2.3. An individual's preference (or utility) for an item may be considered in RFT terms; it is the arbitrary relation of abstract concepts, they are subjective beliefs that can be evaluated via the individual's implicit and explicit on-line behaviour. Implicit beliefs are not subjective probabilities *per se*, but rather constitute the basis for subjective probabilities , *i.e.*, derived relational responding with respect to implicit beliefs that usually involves noticing and coordinating implicit beliefs after/as they occur (personal correspondence with Dr Nigel Vahey). The RFT literature on *"pragmatic verbal analysis"* provides a justification for a Bayesian approach to probability from the point of view that it allows for adaptive (*i.e.* learning) systems that change perhaps continually over time and according to context, as opposed to the more absolutist assumptions made by frequentists.

3.2.1 Decision-making under Uncertainty

The analysis of decision-making under uncertainty varies depending on which discipline is carrying out the research. Statisticians and mathematicians take a normative view and are interested in what people *ought* to do, leading to the optimal outcome, consequence or reward. Whilst psychologists take a descriptive approach and are interested in what people *actually* do in real world decision-making processes.

The leading normative theory is known as Expected Utility (EU) and was first proved by von Neumann & Morgenstern (1947). This theory deals with objective probabilities and represents an axiomatic approach to decision-making. Their method assume that an individual is rational and, as such, wishes to maximise expected utility. However, the von Neumann-Morgenstern approach does not lend itself to subjective probabilities. Savage (1954), following on from von Neumann-Morgenstern's work, proposed Subjective Expected Utility (SEU) Theory. SEU theory is also based on a set of *rationality axioms* for individual preferences among a set of available decisions, which guarantee the existence of such a subjective probability.

The normative models of von Neumann-Morgenstern and Savage sets out how a rational individual *ought* to makes decisions and does not truly represent how individuals *actually* choose between decisions given uncertainty. The Allais (Allais, 1953) and Ellsberg (Ellsberg, 1961) paradoxes demonstrate that there is incoherence as a preference does not hold independently of considerations of other possible rewards; that is, humans act in a way that contradicts independence. This lead to the development of a descriptive approach known as Prospect Theory (Kahneman & Tversky, 1979), which is a descriptive account of decision-making processes.

Following on from a series of experiments that tested the Allais paradox, Kahneman & Tversky (1979) found that individuals overweight outcomes that are certain, relative to outcomes which are probable, which was termed the *certainty effect*. Under the classic model of decision under risk it is assumed that individuals are risk-averse (Wu *et al.*, 2004). Kahneman & Tversky (1979) were able to demonstrate that individuals are risk-averse and risk-seeking[1]. In their experiment they found that 84% of participants preferred \$500 with certainty to a 50% chance at \$1,000. However 72% of the same participants preferred a 0.1% chance at winning \$5,000 to \$5 with certainty. In the first case, the participants are risk-averse whilst in the second case they are risk-seeking. In addition, when the actions involve losses 69% preferred a 50% chance of losing \$1,000 to a certainty of losing \$500 whilst 83% preferred to lose \$5 for sure compared to a 0.1% chance of losing \$5,000. Therefore the reverse occurs and demonstrates that participants were risk-averse for small probabilities and risk-seeking for moderate probabilities. Risk attitude is nothing more than a descriptive label for the shape of the utility function presumed to underlie an individual's choices. An individual's risk attitude describes the shape of his or

[1]A very simple concrete example of this would be individuals who purchase mobile phone insurance and also play the national lottery.

her utility function (derived from a series of risky choices) for the outcomes in question. Within the EU framework it is hypothesized that domain differences in apparent risk attitude might be the result of differences in marginal value for outcomes in different domains (Dyer & Sarin, 1982). Individuals do not appear to be consistently risk-seeking or risk-averse across different domains and situations even when using the same assessment method (Schoemaker, 1990).

In addition to the certainty effect, Kahneman & Tversky (1979) show a *reflection effect*. Through their experiments they found that preferences for losses are a mirror-image reflection of preferences for gains. Therefore, when faced with decision problems, individuals involve not only possible gains, but also possible losses. An assumption of EU theory is that individuals who are risk-averse would be just as averse in the face of losses as they would in the face of gains, and so Kahneman & Tversky (1979) have demonstrated that this assumption has been violated. This means that we cannot consider risk-aversion as an explanation for the certainty effect as individuals are not risk-averse when confronting losses.

Finally, Kahneman & Tversky (1979) comment on what they term the *isolation effect*. The isolation effect explains the fact that individuals making decisions tend to disregard the components that the alternatives share, and tend to focus on what distinguishes them. This occurs whether those components share common pay-off or share common probabilities. Gigerenzer & Brighton (2009) found that individual's cognitive processes rely on simple heuristics to reach accurate responses, they found that the more information that is analysed, the less decision accuracy was observed. This can perhaps account for the isolation effect.

3.2.2 Heuristics in Decision-making

What is meant by heuristics in decision making? The word heuristic simply means to discover and is often referred to as a rule of thumb. Heuristics are the result of experience and the intensity of experience. They are an important part of the decision making process (Kahneman & Tversky, 1979). When individuals make decisions, short cuts are taken, *e.g.*, probabilities may be rounded up and satisficing[2] heuristics used. In fact, individuals often evaluate alternative outcomes sequentially until the first satisfactory or acceptable outcome is found (Simon, 1956).

[2]Satisficing is a decision-making strategy that enables the decision maker to choose an outcome that is good enough, as opposed to the optimal decision outcome (Simon, 1956).

Therefore, as individuals we do not always act as rational agents when making decisions.

According to Kahneman & Tversky (1979) an individual may use stereotypic behaviour to make a decision in an uncertain situation, this is known as *representativeness heuristic*. Another heuristic often employed by individuals is *overconfidence*, this refers to overestimating one's knowledge, underestimating risks and amplifying one's ability to control events. *Anchoring* refers to how an individual allows the advice of influential individuals, such as experts, to influence decision processes, however their influence may only hold at the initial stages of decision-making. When an individual is slow to update in light of new information, this heuristic is known as *conservatism*. Finally, there is *aversion to ambiguity*, which is a bias related to risk-aversion. This refers to why an individual will take little risk under uncertainty. Charles & Kasilingam (2015) state that an investor's emotions, *i.e.* a feeling which is directed to a particular object or event, affects heuristics.

Prospect theory has additional insight on decision-making in that it accounts for utility curves differing in domains of gain from those of domain of loss. A domain effect is simply the commodity being evaluated and Prospect Theory explains how situational variability in the way a decision is framed can have a dramatic impact on the decisions people make. These decisions are not restricted to any particular domain. Such an approach offers insights as to why individuals make non-optimising decisions rather than those that maximise expected utility.

Subjective probabilities might be elicited from expert knowledge or summarise the result of a statistical processing of historical data. Such probabilities are typically quantified via *precise* or *point-wise* values. In many situations, these probabilities cannot provide a satisfactory evaluation of uncertainty, by this very reason, a convincing estimate in a normative decision making context. Therefore, a set-valued, *imprecise*, quantification might offer a better, or at least more cautious specification (de Campos & Antonucci, 2015).

3.2.3 Modelling Vagueness

An early attempt to provide solutions to problems that could not be solved satisfactorily with precise probabilities was Fuzzy Logic (FL) (Zadeh, 1965). FL extends from the theory of fuzzy sets, a theory which relates to classes of objects with unsharp boundaries in which membership is a matter of degree. This is an organised method for dealing with imprecise data with

fuzzy sets allowing partial membership and applies a more human-like way of thinking in the programming of computers. It is based on *"degrees of truth"* which lie on the [0,1] interval rather than the usual *true or false* (1 or 0) Boolean logic on which the modern computer is based. For example instead of a system defining just *hot*=0 and *cold*=1, it may apply *hot*=0, *warm*=.3, *chilly*=.6 and *cold*=1. FL has been incorporated into RSs and one such system is discussed in Section 3.3.

The theory of imprecise probability and interval probability has grown via nonparametric approaches. Nonparametric methods are widely used in statistics, they allow more flexibility than their parametric counterparts as they require fewer assumptions to be met. Nonparametric Predictive Inference (NPI) is based on the assumption $A_{(n)}$ which was proposed by Hill (1968). This assumption allows for prediction in the case of extremely vague prior knowledge. Let $x_1, \ldots, x_n$, be the data values obtained by sampling from a population, with values ordered as $x_{(1)}<x_{(2)}<\ldots<x_{(n)}$ (in increasing order of magnitude). Let $X_1, \ldots, X_n$ be the corresponding pre-data random quantities, so that the data consist of the observed random quantities $X_1 = x_1, \ldots, X_n = x_n$ $A_{(n)}$ is defined as follows:

- The observable random quantities $X_1, \ldots, X_n$ are exchangeable.
- Ties have probability 0. (This was later generalised to include possible ties which tend to zero[3]).
- Given data $x_1, \ldots, x_n$ the probability that the next observation falls in the open interval $I_j = (x_{(j)}, x_{(j+1)})$ is $1/(n+1)$, for all $j = 0, \ldots, n$, where $x_{(0)} = -\infty$ (or 0 when dealing with non-negative random quantities) and $x_{(n+1)} = \infty$.

Therefore, the assumption $A_{(n)}$ is a distribution free, post-data assumption related to finite exchangeability (De Finetti, 1931). It may be interpreted in terms of ranks; the rank of the next observation amongst all observations will be equal to any possible value with probability $1/(n+1)$. De Finetti's (1931) representation theorem uses a similar argument to justify a Bayesian framework for learning about an underlying parameter and its probability distribution. A Bayesian approach explicitly requires a specified prior and conditional independence of future observations and so is not as flexible. Although the assumption $A_{(n)}$ is not sufficient to derive precise probabilities, it does provide bounds for probabilities via the application of De Finetti's

[3]Ties can be dealt with in NPI by assuming that tied observations differ by small amounts (Hill, 1988).

(1931) *"Fundamental Theorem of Probability"*. Therefore, some related predictive inferences may be expressed using imprecise probability.

In NPI, uncertainty about the future observation X_{n+1} is quantified by lower and upper probabilities (Coolen, 2011). These probabilities are such that all orderings of the future random quantities among the observed random quantities are equally likely. This is a low structure statistical technique, which is predictive in nature. The theory of imprecise probability makes it clear that bounds provide valuable information on the uncertainty of events caused by restricted information (Augustin & Coolen, 2004; Walley, 1991; Weichselberger, 2000).

There are several advantages to incorporating imprecise probabilities into a model. There is greater flexibility when quantifying uncertainty, it allows us to deal with conflicting evidence, weaker assumptions are required and it allows for a simpler way of eliciting subjective judgements (Coolen *et al.*, 2011). Many applications of NPI have been presented in the literature, *e.g.*, accelerated life testing (Yin *et al.*, 2016), stock returns (Baker *et al.*, 2017), survival signature (Coolen *et al.*, 2014), right-censored data (Coolen & Yan, 2004), and as an alternative to the imprecise Dirichlet model (Coolen & Augustin, 2009). There have also been many techniques that have incorporated imprecise probability in the literature. For example, sensitivity analysis in engineering (Oberguggenberger *et al.*, 2009), explaining preference reversals (Bayrak & Hey, 2017), structural assessments (Mohammadi *et al.*, 2016; Zhang *et al.*, 2017) and weapons uncertainty qualification (Picard & Vander Wiel, 2016).

Nonparametric Predictive Utility Inference (NPUI) is a specialisation of NPI and incorporates uncertain utility (Houlding & Coolen, 2012). This models extreme vagueness. In NPUI utilities are scaled to the unit interval. This method allows an individual to assign a utility to a novel outcome when he or she has experienced n exchangeable outcomes that (after observation) are believed similar to this. For example, an individual has watched n *romantic* movies which he or she assigns varying utilities to these movies. The individual now wants to predict the satisfaction derived from watching a new *romantic* movie. Let real-valued $u_{(1)}, \ldots, u_{(n)}$, with $u_{(i)} \in (0, 1)$ be the known ordered values of the utilities $u_1, \ldots, u_n$ representing preferences over outcomes $\mathcal{O} = \{o_1, \ldots, o_n\}$. Let $\mathcal{U}_n = \{U_1, \ldots, U_n\}$ denote the set of random quantities representing the utilities of the elements within $\mathcal{O}$ before they are experienced, and suppose that the element of $\mathcal{U}_n$ are considered exchangeable. Then, given a new outcome o_{new} whose

utility value $U_{new} \in (0,1)$ is unknown but considered exchangeable with the elements of $\mathcal{U}_n$, the NPUI models states the following:

$$P\Big(U_{new} \in (0, u_{(1)}]\Big) = P\Big(U_{new} \in [u_{(i)}, u_{(i+1)}]\Big) = P\Big(U_{new} \in [u_{(n)}, 1)\Big) = \frac{1}{n+1}$$

NPUI leads to the following rules:

- Lower expected utility bound:

$$\underline{E}[U_{new}] = \frac{1}{n+1} \sum_{i=1}^{n} u_i$$

- Upper expected utility bound:

$$\overline{E}[U_{new}] = \frac{1}{n+1}\Big(1 + \sum_{i=1}^{n} u_i\Big)$$

- Difference in utility bounds:

$$\Delta E[U_{new}] = \overline{E}[U_{new}] - \underline{E}[U_{new}] = \frac{1}{n+1}$$

In this instance, learning occurs over time with new observations incorporated into the upper and lower bound equations. As n increases the interval width decreases. Therefore the more experiences an individual has, the more confident he or she will be in making a prediction for a future utility value from previous similar experiences and outcomes.

Both RSs and CARSs typically give a single value for the recommended item, and rank items based on their point-wise value. Therefore the precise probability of the user liking such an item is quantified via a single (*precise*) probability that satisfies Kolmogorov's axioms (Augustin & Cattaneo, 2011). Under severe uncertainty, it may be hard to identify a unique probability distribution which honestly represents the limited information available. Therefore using a precise probability may have severe limitations (Coolen *et al.*, 2011) and this approach maybe considered unreasonable (Bradley, 2014). The approach of imprecise probabilities is an alternative to the standard Bayesian epistemology, *i.e.*, the degrees of partial belief of a person or system (Walley, 1991).

The next section introduces RSs, these are modern day decision-making aids. These systems seek to maximise expected utility for the user by filtering through all available items to produce a recommendation. Such recommendations are quantified as a point-wise representation.

3.3 Recommender Systems

Knowledge management within e-commerce is a rapidly developing area that is fast changing in order to meet the demands of more and more individuals using a myriad of on-line services and products. Since the famous PageRank algorithm of the Google search engine many entities and companies have utilised recommender systems. Statistical techniques have been employed to predict and recommend the needs of such individuals with many companies, such as Amazon, Netflix, and TripAdvisor, to name but a few, gathering data about an individual's on-line behaviour to help their customers make decisions about future purchases, to recommend movies, places to visit *etc.* The information gathered about on-line users is often available at no extra cost, *i.e.*, retaining preferences, social media profiles and social tags (*i.e.*, keywords generated by individuals on a platform that are used to describe and categorise an object, concept or idea). Such information is often available to sites as they use links, share and like facilities that are available. This information is often utilised by Recommender Systems (RSs) that assist us with our decision-making processes. This feedback can be collected from several different domains and in various formats. Information in relation to context may also be collected, *e.g*, the time of day, date, type of devise used, *etc.* There are several definitions for RSs but the definition given by Adomavicius & Tuzhilin (2005) is a generalised one:

> *"More formally, the recommendation problem can be formulated as follows: Let C be the set of all users and let S be the set of all possible items that can be recommended. Let u be a utility function that measures the usefulness of item s to user c, that is, $u : C \times S \Rightarrow R$, where R is a totally ordered set (for example, nonnegative integers or real numbers within a certain range). Then, for each user $c \in C$, we want to choose such item $s' \in S$ that maximizes the users utility. More formally:* $\forall \quad c \in C, \quad s'_c = \arg\max_{s \in S} u(c, s)$."

The utility function here is the rating achieved by an item from the given individual or as

predicted by a RS. The more utils an individual assigns an item, the more they prefer that item. A user's utility function is an important aspect of RSs as it predicts the user's preferences and the subjectivity of an individual's utility function has been explored in decision-theory from both statistical and psychological viewpoints (*e.g.*, Kahneman & Tversky (1979), Savage (1954), von Neumann & Morgenstern (1947)).

3.3.1 Categories of Recommender Systems

There are three main categories of RSs which are:

- **Content-based**: Such a RS recommends items that are similar to those preferred by the user in the past; filtering is based on the novel items description and the user's profile. The utility of a novel item is therefore based on the utilities already assigned by the user to similar items. It does not consider the other users' preferences. The statistical techniques often employed for such a RS are cluster analysis such as K-means, hierarchical clustering and decision-trees, *e.g.*, Van den Oord *et al.* (2013), Natsev *et al.* (2007).
- **Collaborative**: This type of RS recommends items that people with similar preferences have liked in the past; it recommends items based on the user's similarity to other users. The utility of a novel item is therefore based on the utilities assigned by users that are deemed to be similar to the user. It does not consider the user's own preferences. The statistical techniques often employed for such a RS are k-nearest neighbours, matrix factorization and Pearson Correlation, *e.g.*, Hill *et al.* (1995), Sarwar *et al.* (2001), Koren (2010).
- **Hybrid**: This approach combines collaborative and content-based methods and allows for the use of both the individuals' ratings as well as those ratings of similar individuals. There are several ways in which these approaches may be combined. For example, running the analysis separately and then combining the outcomes; incorporate some characteristics from a content-based/collaborative method into a a collaborative/content-based approach; or create a model that incorporates both methods equally, *e.g.*, Debnath *et al.* (2008).

Domain characteristics have a strong influence over the types of methods employed to pro-

duce a recommendation as well as the type of data sources that are accessed (Burke *et al.*, 2011). For example, a context-based approach is often used for text-based applications as it has roots in information retrieval (Baeza-Yates & Ribeiro-Neto, 1999) and is analogous to machine learning. The most common algorithm used is the term frequency/inverse document frequency (TF-IDF) measure, which looks at how important a word is to a document and is often used as a weighting factor in information retrieval and text-mining.

For movie and book recommendations a collaborative filtering approach may be taken, with Pearson's coefficient (see Definition 2.28) being a popular similarity measure for two users. It is possible to use such techniques, not only to find similar users, but also to find similar items.

Both of the above approaches may employ data that is gathered from social sources, *i.e*, utilising demographic data, data gathered from previous ratings, reviews of products, on-line purchasing behaviour and/or social tags. However, in order to produce an accurate recommendation for, say banking and financial products, a more individual data source would be required which must also incorporate, not only demographics, behaviour and opinions, but also more personal details such as savings, financial constraints and commitments and a measure of the user's risk aversion.

It is worth noting that an individual's utility for a particular item may vary more rapidly than for another item. For example, an individual's utility associated with, say, a smart device may fluctuate more as next generation devices are released in quick succession. Whereas, an individual's utility for a particular genre of music will fluctuate very little as their music tastes tend to change slowly over time (Burke *et al.*, 2011). Therefore by using domain specific techniques, a more personalized approach to on-line activity is achieved allowing for greater customer satisfaction and as a result these sites retain customers and increase customer loyalty. Maintaining user's profiles, on-line activity, and preferences is one way to achieve a personalized recommendation. However, according to Mansoury *et al.* (2016) the vast amounts of information available to *e*-businesses has led to an information overload and this may lead to customer dissatisfaction.

These very sophisticated algorithms stem from simple statistical techniques such as Pearson's correlation, utilised by Konstan *et al.* (1997) to recommend news group articles via collaborative filtering.

3.3.2 Cold-Start Problem

But what if it is a new-user or new-item? This is known as a cold-start problem and is a data sparsity issue. A new-item cold-start problem is when an item is completely new and as such has not been rated by any individuals. This type of cold-start issue is not isolated to RSs, but may occur in many decision-making processes where there is vagueness and unfamiliarity about a novel item that does not allow for a utility function to be assigned (Houlding & Coolen, 2011, 2012). A new-user cold start problem is when a new individual has not rated anything in the past and so there is no previous information available about this individual's preferences.

New-User

RSs may make use of additional data sources (*e.g.*, demographic data), use the data available from a prominent group of analogous users or use a combination of methods which is known as hybrid methods (Son, 2016). Making use of additional data sources, such as Facebook and Twitter profiles, allows the system to build a demographic profile of the individual. For example, Spotify (a music, podcast and video streaming service) utilises collaborative filtering, Natural Language Processing (NLP), deep learning and analysis on new songs to make recommendations. By incorporating user information (previous songs listened to), similar user informations (based on similar song history), music blogs and elements of music such as melody, harmony, rhythm *etc.* For new songs, Spotify are able to produce highly personalised recommendations in the form of Daily Mix and Discover Weekly (Knight, 2015). In relation to NLP, it is important to note that vagueness is ubiquitous in all natural languages and is still neglected by most natural language processing systems (Fermüller & Roschger, 2014).

Another example is the use of social tags to generate more personalized recommendations (Zhang *et al.*, 2010). One of the most accurate algorithms that incorporates such additional data is the MIPFGWC-CS agorithm which uses Modified Intuitionistic Possibilistic Fuzzy Geographically Weighted Clustering (MIPFGWC) (Son *et al.*, 2013). MIPFGWC uses a fuzzy[4] geography algorithm which allows the system to determine similar users with respect to all aspects of the demographic data. Here each area is allocated a membership value in each of

[4]Fuzzy simply refers to a method that allows one piece of data to belong to two or more groups.

the clusters and so rules out issues related to ecological fallacy[5]. This approach makes use of additional data and can more accurately determine users that are similar to the new user based on their profile's opinions and social tags. Just as with a collaborative approach, once a set of neighbours are found to be analogous with the new user, if the item is rated by these users, then their ratings are considered representative of how a new user would rate them. If an item has not been rated by this group, then a Pearson coefficient is calculated between it and a similar item already rated by the group.

What happens when there is a new user and there is no additional information available? According to Stigler & Becker (1977) *"tastes neither change capriciously nor differ importantly between people"* and therefore it appears reasonable to use other users as a proxy to generate a rating for a new-user (Liu *et al.*, 2014). The New Heuristic Similarity Model (NHSM) was proposed by Liu *et al.* (2014) which is an algorithm that utilises the ratings of the more prominent group of analogous users. This algorithm also addresses some of the limitations of the Pearson coefficient[6] which does not accurately measure the relationship between two items if that relationship is non-linear. The similarity measure, sim^{PSS}, between a new user and an active user is improved by using PSS: Proximity (distance between two ratings), Significance (ratings are more significant if they lie far from the median rating) and Singularity (represents how two ratings are different with other ratings). In turn, this improves how to determine a group of analogous users. By combining sim^{PSS} with another common similarity measure, Jaccard[7] the authors are able to integrate user rating and thus enhance the accuracy of the algorithm.

The final algorithm in relation to the new-user cold-start problem to be considered is one that makes use of hybrid methods. The HU-FCF++, which is an extension of the Hybrid User-based Fuzzy Collaborative Filtering (HU-FCF) (Son, 2015), combines different methods to achieve a better accuracy of predictions and to eliminate some of the limitations that have been found with other techniques. It calculates the number of clusters needed for finding similar

[5]An ecological fallacy occurs when conclusions are drawn based only on the analyses of group data and then these qualities are attributed at the individual level.

[6]Pearson coefficient, cosine similarity and Euclidean distance are the most commonly used similarity measures in collaborative RSs.

[7]Jaccard coefficient (Jaccard, 1912), is defined as the quotient between the intersection and the union of the pairwise compared variables among two objects; it only considers the number of common ratings between two users.

users (this is one of the limitations of the MIPFGWC-CS approach). Next the MIPFGWC algorithm is run which specifies the group of analogous users for a new user. Following this, another algorithm, namely Association Rules Mining (ARM) (to address the limitation of the Pearson coefficient), is utilised to find similar items of a given one, then a pre-ratings set is generated for the Complete Rating data based on the most popular rating of all users. Finally the NHSM algorithm is used to predict from the complete rating dataset. The computational time for this approach is one of its drawbacks (Son *et al.*, 2013).

New-Item

As with the new-user cold-start problem, this is a data sparsity issue. As the item is new, it has yet to be rated and so how does a RS recommend such an item to a user?

Regression type models have been proposed by Agarwal & Chen (2009), Park & Chu (2009) and Xu *et al.* (2014) to mitigate new-item cold-start problems. Xu *et al.* (2014) proposed a Feature-based Regression Model with Baseline Estimates (FRBE) which adopts previous regression approaches. This algorithm combines a regression approach that is based on users' profiles and combines it with baseline estimates. The regression model utilises all available information of users and items to build a predictive model that incorporates a baseline estimate for user and item effects b_{ui}, which is defined as:

$$b_{ui} = r + d_u + d_i$$

where r is the overall average rating of a ratings dataset, d_u is the observed deviations of a user from the average rating and d_i is the observed deviations of an item. This approach can be taken when recommending to a new-user, recommending a new-item and also when recommending a new-item to a new-user as it builds a joint feature space for user/item pairs.

Clustering with Hybrid Features Selection Method (CHFSM) was proposed by Hdioud *et al.* (2016) and, as the name suggests, uses a combination of techniques to solve this problem. This approach incorporates both semantic and statistical dependencies of the items in order to compute more accurate similarities between them. This algorithm consists of a k-means clustering, weighting the keywords using the TD-IDF formula and then calculating the hybrid

measure of the features. This process is repeated until the centroids no longer move and thus provide a more accurate recommendation.

Novelty

The concept of *novelty* is a fundamental field of recommendation effectiveness and added-value according to Zhang (2013): *"There is not a unified definition about novelty, therefore, novel recommendation algorithms are different with respect to the definition of novelty."* Novel recommendations are recommendations for items that the user did not know about (Konstan *et al.*, 2006), it has been defined with respect to the end-user as the proportion of known and unknown relevant items in a recommendation list (Baeza-Yates & Ribeiro-Neto, 1999). Zhang (2013) states that a novel item should have three characteristics: unknown (the item is unknown to the user), satisfactory (the item is satisfied for the user) and dissimilarity (the item is dissimilar to items in the profile of the user). Zhang (2013) believes that novelty should be used as one of the key metrics to measure customer satisfaction. Vargas & Castells (2011) identified three critical ground concepts as the core of novelty and diversity; discovery (an item is seen by, or is familiar to, a user), choice (an item is used, selected, consumed, *etc.*, by a user) and relevance (an item is liked, useful, enjoyed, *etc.*, by a user). Novel recommendation may simply be defined as recommended items users don't know, hence the simplest way to novel recommend is to filter items in profile of the user.

The accuracy of predictions and the quality of recommendations is extremely important as it leads to repeat customers. Such accuracy is measured by the Mean Absolute Error (MAE), which computes the deviation between predicted ratings and actual ratings, and the Root Mean Square Error (RMSE), which also computes the deviation between predicted ratings and actual ratings but places more emphasis on larger deviation. A RS can make two types of errors: it may recommend an item to a user that the user does not like (false positive), or it may fail to recommend an item to a user that they would like (false negative) (Cho *et al.*, 2002). According to Burke *et al.* (2011) *"A recommendation technique that optimizes for high accuracy over the entire data set therefore contains an implicit bias toward well-known items, and therefore may fail to capture aspects of utility related to novelty."*

Misssing Values

Ignoring missing data is a common practice (de Campos & Antonucci, 2015). As seen in Section 2.8, missingness may result for various reasons. Ignoring missing values can be only justified under specific assumptions about the process making the output of an observation/measurement missing. According to Su *et al.* (2008), most collaborative filtering algorithms currently used in RSs implicitly assume MAR, whilst others assume MCAR. However, Marlin *et al.* (2007) tested the hypothesis that the probability of observing a rating of an item is dependent on it's values. The results show a clear dependence of rating frequency on the underlying preference level, this creates a systematic bias towards observing a disproportionate number of high rating values. This indicates that missing values are NMAR.

In relation to exchangeability, unmeasured (*i.e.*, missing values) can be considered exchangeable with measured *(observed)* values (Draper *et al.*, 1993). In this instance, it is appropriate to consider measured values $y_i, i = 1, \ldots, m$ and unmeasured $Y_i, i = m+1, \ldots, n$ values as exchangeable in a sample of the population as opposed to units in the general population but not included in the sample. If these two groups of units are exchangeable, then inferences made on units in the sample may be generalized to the units in the larger population.

It is also important to consider missing values in relation to Pearson's correlation coefficient ρ, see Definition 2.28. There are several ways to calculate ρ when it comes to dealing with missing values. The most common technique is to analyse only those cases for which the data is available on all variables; this is known as listwise or casewise deletion. This method omits any cases that are not entirely complete from the analysis. For example, if we consider the following matrix:

$$M = \begin{bmatrix} -2 & 1.5 & MV & 1 \\ -1 & 2 & 1 & MV \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \end{bmatrix} \quad Cor(M) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Listwise deletion provides relatively unbiased estimates under MCAR but can lead to parameter bias when it is not MCAR (Glasser, 1964). A drawback of using listwise deletion is that it often discards a great deal of potentially usable data. This loss of data leads to larger standard errors, wider confidence intervals, and a loss of power in testing hypotheses. However, the estimated standard errors produced by listwise deletion are usually accurate estimates of the true standard errors. In our example, as we have identical columns in our M matrix, we have perfect correlation. The deletion of the two rows have had rather a large impact on the correlation matrix and we have two rows that had no correlation, now having perfect correlation.

In real life, an individual does not simply disregard information because it is incomplete, often information is filtered based on its relevance (Borlund, 2003). An alternative to listwise deletion is pairwise deletion. This technique calculates the covariance amongst each pair of variables using all available cases for each pair. This has the advantage of including information in the covariance matrix that would have been discarded under listwise deletion. Going back to our example, in this case the first two rows are removed when calculating ρ_{m_1,m_3}, ρ_{m_1,m_4}, ρ_{m_2,m_3}, ρ_{m_2,x_4} and ρ_{m_3,m_4}.

$$M = \begin{bmatrix} -2 & 1.5 & MV & 1 \\ -1 & 2 & 1 & MV \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \end{bmatrix} \quad Cor(M) = \begin{bmatrix} 1 & 0 & 0.63 & 0.48 \\ 0 & 1 & 0.85 & 0.97 \\ 0.63 & 0.85 & 1 & 1 \\ 0.48 & 0.96 & 1 & 1 \end{bmatrix}$$

Given that pairwise deletion utilises more data when producing estimates, it is often more efficient than listwise deletion. Pairwise deletion is unbiased in large samples under MCAR but may suffer from serious parameter bias when data are MAR (Glasser, 1964; Allison, 2002). It can breakdown because the estimated correlation matrix is not positive definite and cannot be inverted to calculate the parameters. Its more common problem, however, is the difficulty in getting accurate estimates of the standard errors. That is because each correlation may be based on a different sample size, depending on the missing-data pattern. Therefore, we may have non-negative eigenvalues (non-positive) and the matrix may contain at least one zero eigenvalue (semi-definite). In the words of Donald Rumsfeld, we have converted *"known*

unknowns" into *"unknown knowns"* - incorrect knowns.

In relation to exchangeability, it is possible to consider a correlation structure as exchangeable, since clearly there is dependence between measurements on the same individual at different times, but the exact form of the correlation is not clear (Liang & Zeger, 1986).

3.3.3 Recommendations via Heuristics

In addition to sophisticated algorithms and statistical functions, these systems incorporate recommendation heuristics to reduce computational time, make the data scalable and may be applied pre- or post-filtering (Burke & Ramezani, 2011). Heuristic-based algorithms essentially are heuristics that make rating predictions based on the entire collection of previously rated items by the users (Adomavicius *et al.*, 2005). The commonly used heuristic-based recommendations for content-based RSs are TF-IDF and clustering; for collaborative RSs they are nearest neighbour techniques (cosine, correlation), clustering and graph theory; and for hybrid RSs they utilise linear combination of predicted ratings, voting schemes and incorporating one component as a part of the heuristic for the other (Adomavicius & Tuzhilin, 2005).

3.4 Chapter Summary

The quantification of uncertainty via precise probability has been incorporated into normative decision theories. Such theories set up an axiomatic approach as to how rational individuals can and must maximise expected utility. However, limitations of these Normative approaches were exposed via the Allais (Allais, 1953) and Ellsberg (Ellsberg, 1961) paradoxes, which demonstrated that humans do not always act in a rational way. Kahneman & Tversky (1979) went on to propose their own theory of decision-making, this time from a psychological perspective and empirical research. What they found was that humans often use heuristics, can be risk averse and risk seeking and language plays a role in decision-making processes. Given, that there are a variety of sources of information, a point-wise approach to assessing uncertainty may not be suitable. Thus, a better and perhaps more cautious approach is to use an interval based quantification for uncertainty. This can be achieved via NPI. NPI assumes exchangeability and has been successfully applied to a number of research areas.

Recommender systems, much like normative decision theories, wish to maximise the user's expected utility whilst filtering the vast amount of information available now-a-days via the Internet. These recommender systems also utilise a number of sources, yet give a point-wise estimate of the predicted utility. In a similar vein to decision-theory, perhaps a better and more cautious approach would be to use an interval quantification for predicted utility. It is possible to assume exchangeability for missing values and for correlation coefficients, therefore it appears to be appropriate to apply the NPI technique to recommendations.

The next chapter introduces the Robust Bayesian Correlation Learning (RBCL) technique of Troffaes *et al.* (2014). This approach is used to estimate Pearson's correlation coefficient. It will be used as a comparison to the proposed technique, the Uncertainty Interval (UI), of this thesis, which is subsequently discussed.

4

Proposed Methodology

Thus far probability, both precise and imprecise, and utility have been discussed. The key concepts that our proposed methodology are built on have also been discussed, with the concepts of Nonparametric Predictive Inference (NPI), exhangeability, missing values and correlation explained. We have also seen how a Recommender System (RS) is a modern day decision-making aid, which learns via statistical techniques and may utilise all available data on a user and related users/items. In this chapter we first examine an existing technique, Robust Bayesian Correlation Learning (RBCL), that offers a similar approach to our proposed technique. Next, the proposed technique is presented and justified.

4.1 Robust Bayesian Correlation Learning

Learning under severe uncertainty in relation to correlation coefficients have been explored by Troffaes *et al.* (2014). Robust Bayesian Correlation Learning (RBCL) introduced two models, one continuous and one categorical, to learn about dependence between two random variables under extreme uncertainly using only limited joint observations. The objective of this method is to estimate a correlation coefficient by utilising the pairwise comparisons between sets of variables. Inference is made on a multivariate normal random variable (MVN) Z_i with $z_i = (z_{i1}, \ldots, z_{ik})$ being a particular realisation of Z_i. The mean is known and rescaled to zero, whilst the covariance matrix Σ is unknown.

The likelihood function of an i.i.d. sample $z_1, \ldots, z_n$ is:

$$f(z_1,\ldots,z_n|\Sigma) \propto |\Sigma|^{-\frac{n}{2}} \prod_{i=1}^{n} \exp\Big[-\frac{1}{2}(z_i^T\Sigma^{-1}z_i)\Big]$$
$$= |\Sigma|^{-\frac{n}{2}} \exp\left[-\frac{1}{2}\sum_{i=1}^{n} tr(z_i z_i^T\Sigma^{-1})\right]$$

The Inverse Wishart (IW) is the natural conjugate prior for a covariance matrix and is commonly used with normal sampling models as a result. The IW distribution guarantees to produce positive definite draws and generates random covariance matrices. ν_0 acts as a parameter for learning speed, which is similar to Walley's s value in the imprecise Dirichlet model (Walley, 1996), acting as a tuning parameter. The larger ν_0>0, the more hypothetical prior observations are considered and the more the prior distribution influences the posterior distribution of Σ. Ψ_0 is the corresponding prior covariance matrix, Σ_0, for a particular value of ν_0.

The likelihood for the IW is:

$$f(\Sigma|\nu_0,\Psi_0) \propto |\Sigma|^{-\frac{\nu_0+k+1}{2}} exp\Big[-\frac{1}{2}tr(\Psi_0\Sigma^{-1})\Big]$$

The hyperparameters are updated as follows:

$$\nu_n = \nu_0 + n \qquad\qquad \Psi_n = \Psi_0 + \sum_{i=1}^{n} z_i z_i^T$$

Σ_0 denotes the prior covariance matrix and its expectation is $E(\Sigma_0|\nu_0,\Psi_0) = \Psi_0/(\nu_0+k+1)$. Therefore, the posterior covariance matrix Σ_n has expectation:

$$E(\Sigma_n|z_1,\ldots,z_n,\nu_0,\Psi_0) = E(\Sigma_n|\nu_n,\Psi_n) = \frac{\Psi_n}{\nu_n-k-1}$$
$$= \frac{(\nu_0-k-1)\Sigma_0 + \sum_{i=1}^{n} z_i z_i^T}{\nu_0+n-k-1}$$

For a bivariate case, let $Z_i = (Z_{i1}, Z_{i2})$ and $\sigma_{Z_{i1}}$ and $\sigma_{Z_{i2}}$ be the variance of Z_{i1} and Z_{i2} respectively. The mean of each is $\mu_{Z_{i1}}$ and $\mu_{Z_{i2}}$. Both the variances and means are normalised. We have $k = 2$ and the prior covariance matrix is:

$$\Psi_0 = (\nu_0 - 3) \begin{bmatrix} \sigma_X^2 & \rho_0\sigma_X\sigma_Y \\ \rho_0\sigma_X\sigma_Y & \sigma_Y^2 \end{bmatrix}$$

In RBCL the prior correlation is $\rho_0 \in [-1, 1]$. Both prior σ_X and σ_Y are considered to be well known, but ρ_0 is unknown. Now there are two unknown hyperparameters: ν_0 and ρ_0. The posterior covariance matrix is then defined as:

$$\Sigma_n = \frac{1}{\nu_0' + n} \begin{bmatrix} \nu_0' + \Sigma_{i=1}^n x_i^2 & \nu_0'\rho_0 + \Sigma_{i=1}^n x_i y_i \\ \nu_0'\rho_0 + \Sigma_{i=1}^n x_i y_i & \nu_0' + \Sigma_{i=1}^n y_i^2 \end{bmatrix}$$

where $\nu_0' = \nu_0 - k - 1 = \nu_0 - 3$. The posterior bounds on the correlations are calculated as follows:

$$\underline{\rho}_n = \frac{-\nu_0' + \Sigma_{i=1}^n x_i y_i}{\nu_0' + n}$$

$$\overline{\rho}_n = \frac{\nu_0' + \Sigma_{i=1}^n x_i y_i}{\nu_0' + n}$$

However, there are at least three problems with using an IW prior in this setting:

1. The uncertainty for all variance parameters are controlled by the single degree of freedom parameter, and hence, provides no flexibility to incorporate different amounts of prior knowledge to different variance components (Gelman *et al.*, 2003).
2. When ν>1, the marginal distribution for the variance is an inverse gamma, which has a region near zero with extremely low density and causes a bias toward larger variances when the true variance is small (Gelman, 2006).
3. There is a prior dependency between the variances and correlations such that larger variances are associated with correlations near +/- 1 while small variances are associated with correlations near zero. Thus, when the true variance is small, the correlation will be estimated to be zero regardless of the true value of the correlation and this bias remains even for relatively large sample sizes (Tokuda *et al.*, 2011).

In addition, this approach utilises pairwise comparisons and provides a model to estimate the dependence between two random variables under extreme uncertainty. As such, it produces an estimate of a correlation in the form of an interval, with the width of the interval dependent on the number of pairwise comparisons available. Hence, when there are no pairwise comparisons,

this algorithm gives the vacuous information $[-1, 1]$. However, this approach ignores other information in a multivariate data setting. As individuals, we do not learn in a vacuum and knowledge for how two items are related is not solely assessed on these two items alone. Rather some decision-making processes would incorporate how these two items relate to other items and if some analogy may be formed.

For example, Relational Frame Theory (RFT) is a modern behaviour-analytic account of human language and cognition which builds on Skinner's (1938) observations on reinforcement (Hayes *et al.*, 2001). RFT is based on the assumption that higher-cognitive functioning and language are composed of relational frames, which are patterns of generalised relational responding. That is, humans are capable of learning through complex untrained (derived) relations and, learning occurs in relation to contextual cues via a history of multiple-exemplar training, see Barnes & Roche (1996); Hayes & Hayes (1989). Relational responses emerge early on in development as a result of socio-verbal interactions and develop across time (Barnes-Holmes *et al.*, 2011). A coherent relational network is then formed, given sufficient time, through additional relational responses.

4.2 Uncertainty Interval

Building on the concept of NPUIs, which contains elements of NPIs and uncertain utility, the proposed approach is to adapt the NPUI equations to estimate the correlation coefficient between two items. Recall the lower and upper NPUIs are calculated as:

$$\underline{E}[U_{new}] = \frac{1}{n+1}\sum_{i=1}^{n} u_{(i)} = \frac{1}{n+1}\sum_{i=1}^{n} u_i$$
$$\overline{E}[U_{new}] = \frac{1}{n+1}\Big(1 + \sum_{i=1}^{n} u_{(i)}\Big) = \frac{1}{n+1} + \underline{E}[U_{new}]$$

The correlation coefficient was selected as this is a similarity measure that is often used in RSs and it is possible to generate pairwise correlations even for very sparse matrices. In addition, as utilities can be restricted to the unit interval, there is a strong similarity to the magnitude of a correlation coefficient.

Recall that Pearson's correlation coefficient is calculated as follows:

$$\begin{aligned}\rho_{x,y} &= \frac{1}{n-1}\sum_{i=1}^{n}\frac{x_i-\bar{x}}{\sigma_x}\frac{y_i-\bar{y}}{\sigma_y} \\ &= \sum_{i=1}^{n}\frac{(x_i-\bar{x})(y_i-\bar{y})}{n-1}\left(\frac{1}{\sigma_x}\right)\left(\frac{1}{\sigma_x}\right) \\ &= \frac{Cov(x,y)}{\sigma_x\sigma_y}\end{aligned}$$

As in the previous section, the correlation coefficient was also estimated using Bayesian techniques by Troffaes *et al.* (2014). Their approach required a prior as well as pairwise comparisons between items; it utilises pairwise comparisons and provides a model to estimate the dependence between two random variables under extreme uncertainty. As such, it produces an estimate of a correlation in the form of an interval, with the width of the interval dependent on the number of pairwise comparisons available. Hence, when there are no pairwise comparisons, this algorithm gives the vacuous information [-1, 1]. However, this approach ignores other information in a multivariate data setting. It also requires a prior distribution to be calculated and that variables are normally distributed, whereas NPUI does not have such requirements.

There a several limitations to using a correlation coefficient to predict similarities according to Sheugh & Alizadeh (2015):

1. Correlation between two user profiles can only be computed based on items that both users have rated. Within the context of a RS, users can select among thousands of items to rate, it is likely that overlap of rated items between two users will be small in many cases. Therefore, the correlation coefficient cannot be regarded as a trustworthy measure of similarity because it is based on just a few observations.
2. Two users can only be similar if there is overlap among the rated items, that is, if users did not rate any common items, their user profiles cannot be correlated.
3. The Correlation technique induces one global model of similarities between users, rather than separate models for classes of ratings (*e.g.* positive rating versus negative rating).

The proposed approach in this chapter attempts to overcome these limitations and to provide an accurate estimate of the correlation coefficient by utilising more observations and calculating an estimate between non-overlapping user ratings as well as defining positive relations.

In order to develop what we term an Uncertainty Interval (UI), it was proposed to incorporate information that would not normally be utilised to estimate a relationship between two variables. This is very much in keeping with techniques used by RSs when solving a cold-start problem. It is not a RS in of itself, but an introduction to incorporating an interval approach on one of the basic and most utilised statistics in RSs, namely correlations. It was noted that pairwise correlations generated missing values (MV) given very sparse matrices. Therefore, it is proposed to utilise the correlations that could be calculated in order to estimate a correlation coefficient of a MV. If a MV is returned for a correlation calculation between item a and item b, the correlations of item a and item i and the correlations between item b and item i, where item i represents all other items in the matrix, would provide information about how item a and item b are related. Hence, the UI approach is also a pairwise comparison method, see Fig. 4.2.1 and in Fig. 4.2.2 the graph resembles a Directed Acyclic Graph.

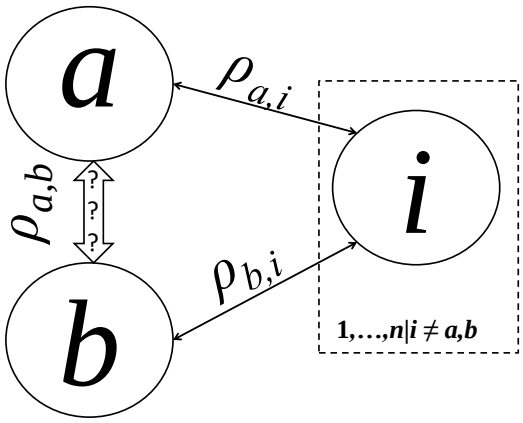

Figure 4.2.1: Representation of correlation.

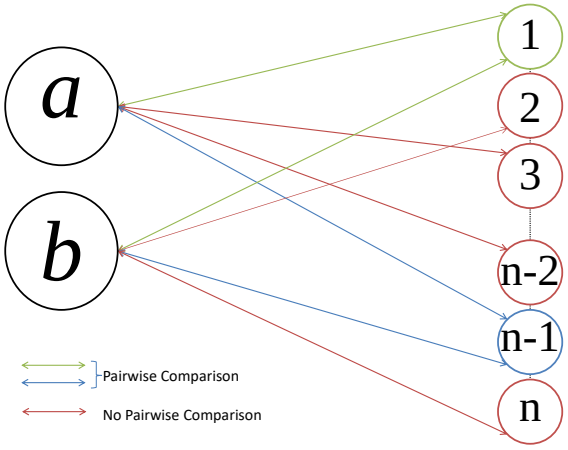

Figure 4.2.2: Pairwise comparison.

The first step in creating the UI was to modify the NPUI equations for lower and upper expectations of utility to incorporate our measure of choice, correlation. This approach seeks to estimate the upper and lower bound of a correlation, $\rho_{a,b}$, such that:

$$\underline{\rho}_{a,b} = \frac{1}{n+1} \sum_{i=1, i \neq a,b}^{n} (\text{information of how item } a \text{ relates to } i \text{ and how } b \text{ relates to } i)$$

$$\overline{\rho}_{a,b} = \frac{1}{n+1} \Big(1 + \sum_{i=1, i \neq a,b}^{n} (\text{information of how item } a \text{ relates to } i \text{ and how } b \text{ relates to } i)\Big)$$

Where the correlation between items a and b is unknown and i is all the other items in the matrix. In Section 4.2 the least amount of information is displayed as the lower bound, whilst

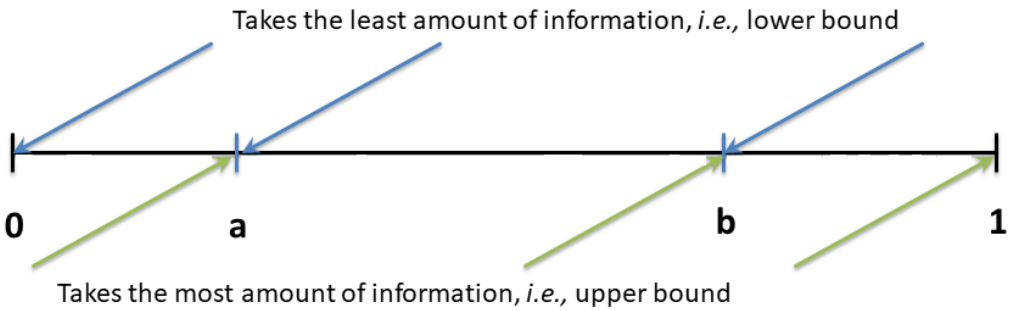

Figure 4.2.3: Information about a and b

the most amount of information is displayed as the upper bound. This is illustrated in the example below, which represents a correlation matrix of a ratings matrix. Information of how item a relates to i is highlighted green and information of how item b relates to i is highlighted blue. Information on how item a/b is related to itself is not utilised as this always has perfect positive correlation.

$$\rho_{M_{Rating}} = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} \begin{array}{c} \begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\ \begin{bmatrix} \rho_{a,a}=1 & \rho_{a,b} & \rho_{a,i=1} & \rho_{a,i=2} & \rho_{a,i=3} \\ \rho_{b,a} & \rho_{b,b}=1 & \rho_{b,i=1} & \rho_{b,i=2} & \rho_{b,i=3} \\ \rho_{i=1,a} & \rho_{i=1,b} & \rho_{i=1,i=1}=1 & \rho_{i=1,i=2} & \rho_{i=1,i=3} \\ \rho_{i=2,a} & \rho_{i=2,b} & \rho_{i=2,i=1} & \rho_{i=2,i=2}=1 & \rho_{i=2,i=3} \\ \rho_{i=3,a} & \rho_{i=3,b} & \rho_{i=3,i=1} & \rho_{i=3,i=2} & \rho_{i=3,i=3}=1 \end{bmatrix} \end{array}$$

By building on and extending NPI and NPUI techniques, the UI approach assumes that the correlations are partially exchangeable if the domain and context of the items are the same. For example, with a movie ratings matrix, the correlations for this matrix may not be considered exchangeable with each other but when the ratings matrix is broken down by say genre, the correlations within this sub-ratings matrix may be assumed exchangeable. That is to say, that the correlation coefficients are exchangeable within a particular genre but not between genres. By assuming partial exchangeability in this way, we are also assuming that the items in a sub-ratings matrix are positively related. The next challenge is to combine this information in a meaningful manner as to estimate the true correlation coefficient of item a and item b.

4.2.1 Combining Pairwise Comparisons

In this section, the methods considered to combine the pairwise comparisons will be discussed. As stated above, the primary aim is to combine information about how item a relates to item(s) i with information about how item b relates to item(s) i in order to estimate the true correlation coefficient of items a and b. To distinguish between the pairwise comparison utilised in the calculation of the correlation coefficient (*i.e.*, the pairwise comparisons from M_{Rating}) and the pairwise comparison utilised to calculate a MV (*i.e.*, the pairwise comparisons from $\rho_{M_{Rating}}$) the former will be referred to as a *pairwise correlation* and the latter, a *pairwise comparison*.

In order to illustrate what is meant by pairwise comparisons, let us consider the following example. Let M_{Rating} be a 5×5 rating matrix where the rows correspond to five users, $u_1, \ldots, u_5$, and the columns represent five different movies, $m_1, \ldots, m_5$. Each movie is rated between 1 and 10, 1 representing the user did not enjoy the movie and 10 representing the user enjoyed the movie. Not every movie is rated by every user, these are represented as blank spaces in M_{Rating}.

$$M_{Rating} = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{array} \begin{array}{c} \begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\ \begin{bmatrix} 10 & 10 & 10 & 10 & 10 \\ 8 & 8 & 6 & & 4 \\ 6 & 7 & & 5 & 8 \\ 5 & 9 & 9 & & 10 \\ 7 & 10 & & 4 & 10 \end{bmatrix} \end{array} \qquad \rho_{M_{Rating}} = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} \begin{array}{c} \begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\ \begin{bmatrix} 1 & 0.42 & 0.13 & 0.92 & -0.12 \\ 0.42 & 1 & 0.96 & 0.36 & 0.62 \\ 0.13 & 0.96 & 1 & MV & 0.97 \\ 0.92 & 0.36 & MV & 1 & 0.36 \\ -0.12 & 0.62 & 0.97 & 0.36 & 1 \end{bmatrix} \end{array}$$

As there are missing variables in M_{Rating}, a pairwise correlation is calculated. This results in the matrix $\rho_{M_{Rating}}$ which has one MV, indicating that no pairwise correlation can be calculated between m_3 and m_4. The MV is the result of a single observation for $(u_1, m_3) = 10$ paired with a single observation for $(u_1, m_4) = 10$. Therefore, given a single point, no correlation coefficient can be calculated.

Before proceeding with how the UI technique can estimate this MV, consider how the RBCL

approach would go about calculating an interval for ρ_{m_3,m_4}. Setting $\nu_0 = 2$, then:

$$\underline{\rho}_1 = \frac{1 + 100}{-1 + 1}$$

$$\overline{\rho}_1 = \frac{-1 + 100}{-1 + 1}$$

Both $\underline{\rho}_1$ and $\overline{\rho}_1$ are undefined. As it is not possible to calculate a pairwise correlation, it is not possible to apply the RBCL approach and as such a vacuous interval of [-1,1] is returned.

In order to estimate the true correlation coefficient of a MV, the UI approach utilises all pairwise comparisons for m_3 and m_4 in $\rho_{M_{Rating}}$ to estimate the ρ_{m_3,m_4}. These pairwise comparisons are highlighted in the matrix below. In the instance where a pairwise comparison contains a MV and/or is a correlation of an item with itself, *e.g.*, $\rho_{a,a} = 1$, these comparisons are removed from the algorithm. In Fig. 4.2.4 the light shaded area represents the information from $\rho_{m_{3,i}}, i = 1, 2, 5$ which is calculated as follows:

$$\begin{aligned}
\underline{\rho_{3,new}} &= \frac{1}{n+1} \sum_{i=1, i\neq 3}^{n} \rho_{m_{3,i}} = \frac{1}{4}(.13 + .96 + .97) = .51 \\
\overline{\rho_{3,new}} &= \frac{1}{n+1}\Big(1 + \sum_{i=1, i\neq 3}^{n} \rho_{m_{3,i}}\Big) = \frac{1}{4}(1 + .13 + .96 + .97) = .76 \\
\Delta(\rho_{3,new}) &= \underline{\rho_{3,new}} - \overline{\rho_{3,new}} = \frac{1}{n+1} = .25
\end{aligned}$$

This generates the interval (0.51, 0.76) which represents the estimate for the correlation between item 3 and a *new* item. This provides information on how item 3 relates to all the other items in the matrix, ignoring missing values.

The darker shaded area represents the information from $\rho_{m_{4,i}}, i = 1, 2, 5$ which is calculated in a similar manner:

$$\begin{aligned}
\underline{\rho_{4,new}} &= \frac{1}{n+1} \sum_{i=1, i\neq 4}^{n} \rho_{m_{4,i}} = \frac{1}{4}(.92 + .396 + .36) = .41 \\
\overline{\rho_{4,new}} &= \frac{1}{n+1}\Big(1 + \sum_{i=1, i\neq 4}^{n} \rho_{m_{4,i}}\Big) = \frac{1}{4}(1 + .92 + .36 + .36) = .66 \\
\Delta(\rho_{4,new}) &= \underline{\rho_{4,new}} - \overline{\rho_{4,new}} = \frac{1}{n+1} = .25
\end{aligned}$$

In this instance, the estimate for the correlation between item 4 and a *new* item lies in the interval (0.41, 0.66). Again, this provides information on how item 4 relates to all other items in the matrix, ignoring missing values.

The ordering of the pairwise correlations does not affect the final interval as it is an additive calculation. As the UI approach is only concerned with the final interval that is produced, the ordering of the pairwise correlations is irrelevant. Notably, $\Delta\rho_{3,new} = \Delta\rho_{4,new}$ as, even though the information is different, the amount of information is the same, *i.e.*, there are three correlation coefficients in each case.

$$
\begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 0.42 & 0.13 & 0.92 & -0.12 \\
0.42 & 1 & 0.96 & 0.36 & 0.62 \\
0.13 & 0.96 & 1 & MV & 0.97 \\
0.92 & 0.36 & MV & 1 & 0.36 \\
-0.12 & 0.62 & 0.97 & 0.36 & 1
\end{array}\right]
\end{array}
$$

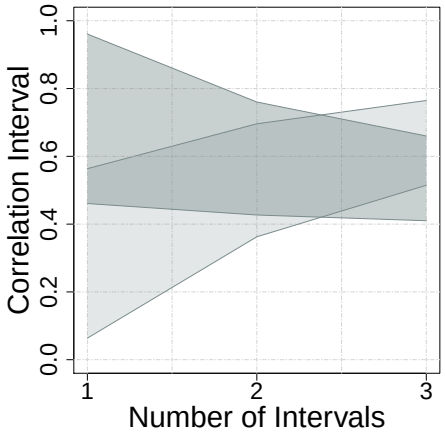

Figure 4.2.4: Information from $\rho_{3,i}$ and $\rho_{4,i}$.

The next challenge is to decide how to combine the information from $\rho_{a,i}$ with $\rho_{b,i}$ in order to estimate $\rho_{a,b}$. There are a number of ways to combine this information, one possibility is to considered $\max(\rho_{a,i}, \rho_{b,i})$. Therefore, going back to our example, in order to estimate the correlation coefficient for m_3 and m_4, $\rho_{m_1,m_4} = 0.92$, $\rho_{m_2,m_3} = 0.96$ and $\rho_{m_5,m_3} = 0.97$ are utilised. In doing so, information on how m_3 relates to m_1 is ignored, as is the information about how m_4 relates to m_2 and m_5. This approach may be considered as an *optimistic* one as it takes the highest value at each interval.

$$
\begin{aligned}
\underline{\rho}_{3,4} &= \frac{1}{n+1} \sum_{i=1, i\neq 3,4}^{n} \max(\rho_{m_{3,i}}, \rho_{m_{4,i}}) = \frac{1}{4}(.92 + .96 + .97) = .71 \\
\overline{\rho}_{3,4} &= \frac{1}{n+1}\Big(1 + \sum_{i=1, i\neq 3,4}^{n} \max(\rho_{m_{3,i}}, \rho_{m_{4,i}}) = \frac{1}{4}(1 + .92 + .96 + .97) = .96 \\
\Delta(\rho_{3,4}) &= \underline{\rho}_{3,4} - \overline{\rho}_{3,4} = \frac{1}{n+1} = .25
\end{aligned}
$$

In our example it returns the interval $(0.71, 0.96)$, see Fig. 4.2.5. As before, the difference between the upper and lower intervals is .25. The amount of information is still three pairwise comparisons, thus has not changed. However, the information itself has changed, thus returning a different interval.

$$
\begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 0.42 & \max(0.13, & 0.92) & -0.12 \\
0.42 & 1 & \max(0.96, & 0.36) & 0.62 \\
0.13 & 0.96 & 1 & MV & 0.97 \\
0.92 & 0.36 & MV & 1 & 0.36 \\
-0.12 & 0.62 & \max(0.97, & 0.36) & 1
\end{array}\right]
\end{array}
$$

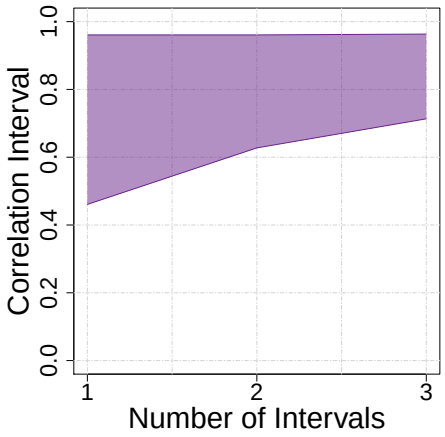

Figure 4.2.5: Maximum value of each pairwise comparison.

In a similar vein, if $\min(\rho_{a,i}, \rho_{b,i})$ is considered then $\rho_{m_1,m_3} = 0.13$, $\rho_{m_2,m_4} = 0.36$ and $\rho_{m_5,m_4} = 0.36$ are utilised.

$$
\begin{aligned}
\underline{\rho}_{3,4} &= \frac{1}{n+1} \sum_{i=1, i\neq 3,4}^{n} \min(\rho_{m_3,i}, \rho_{m_4,i}) = \frac{1}{4}(.13 + .36 + .36) = .21 \\
\overline{\rho}_{3,4} &= \frac{1}{n+1}\Big(1 + \sum_{i=1, i\neq 3,4}^{n} \min(\rho_{m_3,i}, \rho_{m_4,i}) = \frac{1}{4}(1 + .13 + .36 + .36) = .46 \\
\Delta(\rho_{3,4}) &= \underline{\rho}_{3,4} - \overline{\rho}_{3,4} = \frac{1}{n+1} = .25
\end{aligned}
$$

$$
\begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 0.42 & \min(0.13, & 0.92) & -0.12 \\
0.42 & 1 & \min(0.96, & 0.36) & 0.62 \\
0.13 & 0.96 & 1 & MV & 0.97 \\
0.92 & 0.36 & MV & 1 & 0.36 \\
-0.12 & 0.62 & \min(0.97, & 0.36) & 1
\end{array}\right]
\end{array}
$$

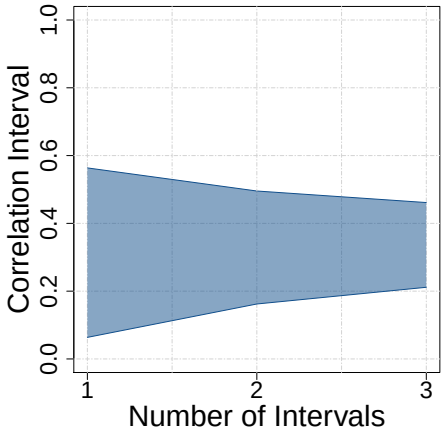

Figure 4.2.6: Minimum value of each pairwise comparison.

Once again, this is not ideal as information on how m_3 relates to m_2 and m_5 and how m_4 relates to m_1 is not incorporated in our example. This may be considered a *pessimistic* approach to estimating the true correlation coefficient ρ_{m_3,m_4}. In our example, taking the minimum value of pairwise comparisons gives the interval $(0.21, 0.46)$, see Fig. 4.2.6.

The concept of taking maximum or minimum values is akin to Houlding and Coolen's (2012) approach to solving sequential decisions that incorporates NPUI via a rule that is based on an attitude of Extreme Pessimism (EP) and a rule that is based on an attitude of Extreme Optimism (EO). EP occurs when the decision-maker will always select the outcome or sequential decision path whose lower expected utility bound is greatest, whereas EO occurs when the decision-maker will always select the outcome or sequential decision path whose upper expected utility bound is greatest. However the UI approach seeks to incorporate information about how items a and b are related to all other items in the matrix in order to estimate how they are related to each other and taking either the maximum or minimum value ignores some information.

To overcome the shortfalling of taking the maximum or minimum value, the average of the pairwise comparisons, $\frac{\rho_{a,i}+\rho_{b,i}}{2}$ may be considered. In doing so, it assures that information about the relationship between item a and item(s) i and the relationship between item b and item(s) i is incorporated in the building of the interval.

$$\begin{aligned}
\underline{\rho}_{3,4} &= \frac{1}{n+1} \sum_{i=1, i\neq 3,4}^{n} \text{avg}(\rho_{m_{3,i}}, \rho_{m_{4,i}}) = \frac{1}{4}(.60 + .66 + .66) = .48 \\
\overline{\rho}_{3,4} &= \frac{1}{n+1}\Big(1 + \sum_{i=1, i\neq 3,4}^{n} \text{avg}(\rho_{m_{3,i}}, \rho_{m_{4,i}}) = \frac{1}{4}(1 + .60 + .66 + .66) = .73 \\
\Delta(\rho_{3,4}) &= \underline{\rho}_{3,4} - \overline{\rho}_{3,4} = \frac{1}{n+1} = .25
\end{aligned}$$

In our example, the UI approach estimates the lower bound of ρ_{m_3,m_4} as 0.46 and its upper bound as 0.73, see Fig. 4.2.7. As the NPUI equations calculate an expectation, the modified NPUI equations for the UI approach may also be considered as measuring an expectation. Therefore, in effect this takes the average of a pairwise correlation and then the UI approach takes the average of these averages.

$$
\begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 0.42 & \text{avg}(0.13, & 0.92) & -0.12 \\
0.42 & 1 & \text{avg}(0.96, & 0.36) & 0.62 \\
0.13 & 0.96 & 1 & MV & 0.97 \\
0.92 & 0.36 & MV & 1 & 0.36 \\
-0.12 & 0.62 & \text{avg}(0.97, & 0.36) & 1
\end{array}\right]
\end{array}
$$

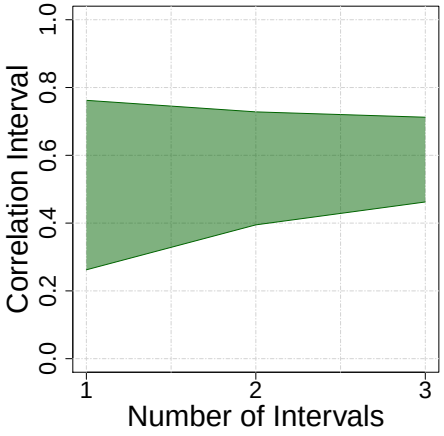

Figure 4.2.7: Average value of each pairwise comparison.

In considering how to combine the information from the correlations, what is being estimated needs to be considered, *i.e.*, $\hat{\rho}_{a,b}$. As $\rho_{m_{a,i}}$ and $\rho_{m_{b,i}}$ are summary statistics, they may be considered standardised random variables. The product of the two correlation coefficients combines the linear relationship between item a and item(s) i with the linear relationship of item b with item(s) i by the following:

$$
\begin{aligned}
\underline{\rho}_{a,b} &= \frac{1}{n+1} \sum_{i=1, i\neq a,b}^{n} \frac{Cov(a,i)}{\sigma_a \sigma_i} \times \frac{Cov(b,i)}{\sigma_b \sigma_i} \\
&= \frac{1}{n+1} \sum_{i=1, i\neq a,b}^{n} \rho_{m_{a,i}} \times \rho_{m_{b,i}} \\
\overline{\rho}_{a,b} &= \frac{1}{n+1} \sum_{i=1, i\neq a,b}^{n} \left(1 + \frac{Cov(a,i)}{\sigma_a \sigma_i} \times \frac{Cov(b,i)}{\sigma_b \sigma_i}\right) \\
&= \frac{1}{n+1}\left(1 + \sum_{i=1, i\neq a,b}^{n} \rho_{m_{a,i}} \times \rho_{m_{b,i}}\right) \\
\Delta(\rho_{a,b}) &= \underline{\rho}_{a,b} - \overline{\rho}_{a,b} = \frac{1}{n+1}
\end{aligned}
$$

Going back to our example, this results in the interval $(0.20, 0.45)$, see Fig. 4.2.8.

$$
\begin{aligned}
\underline{\rho}_{3,4} &= \frac{1}{n+1} \sum_{i=1, i\neq 3,4}^{n} \rho_{m_{3,i}} \times \rho_{m_{4,i}} = \frac{1}{4}(.12 + .35 + .35) = .20 \\
\overline{\rho}_{3,4} &= \frac{1}{n+1}\left(1 + \sum_{i=1, i\neq 3,4}^{n} \rho_{m_{3,i}} \times \rho_{m_{4,i}}\right) = \frac{1}{4}(1 + .12 + .35 + .35) = .45 \\
\Delta(\rho_{3,4}) &= \underline{\rho}_{3,4} - \overline{\rho}_{3,4} = \frac{1}{n+1} = .25
\end{aligned}
$$

$$
\begin{array}{c}
 & m_1 & m_2 & m_3 & & m_4 & m_5 \\
m_1 & 1 & 0.42 & (0.13 & \times & 0.92) & -0.12 \\
m_2 & 0.42 & 1 & (0.96 & \times & 0.36) & 0.62 \\
m_3 & 0.13 & 0.96 & 1 & & MV & 0.97 \\
m_4 & 0.92 & 0.36 & MV & & 1 & 0.36 \\
m_5 & -0.12 & 0.62 & (0.97 & \times & 0.36) & 1
\end{array}
$$

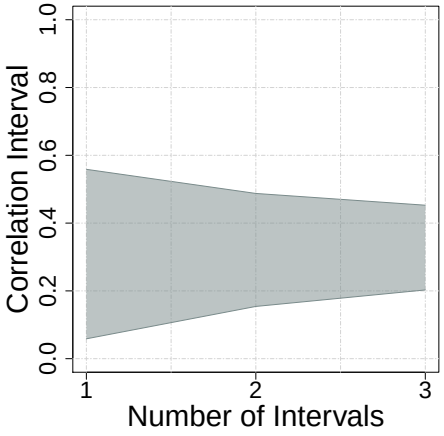

Figure 4.2.8: Product of each pairwise comparison.

As the UI approach will incorporate the products of correlations, it is necessary to understand the how these products will effect $\underline{\rho}_{a,b}$ and $\overline{\rho}_{a,b}$. Given that exchangeable random variables are assumed to have positive correlations, only positive correlations shall be considered. If we consider a strong correlation as ρ_{strong} = .9, a medium correlation as ρ_{medium} = .5 and a weak correlation as ρ_{weak} = .1, representing this as a simple matrix:

	ρ_{strong}	ρ_{medium}	ρ_{weak}
ρ_{strong}	ρ_{strong}	ρ_{medium}	ρ_{weak}
ρ_{medium}	ρ_{medium}	ρ_{weak}	ρ_{weak}
ρ_{weak}	ρ_{weak}	ρ_{weak}	ρ_{weak}
	0.9	0.5	0.1
0.9	0.81	0.45	0.09
0.5	0.45	0.25	0.05
0.1	0.09	0.05	0.01

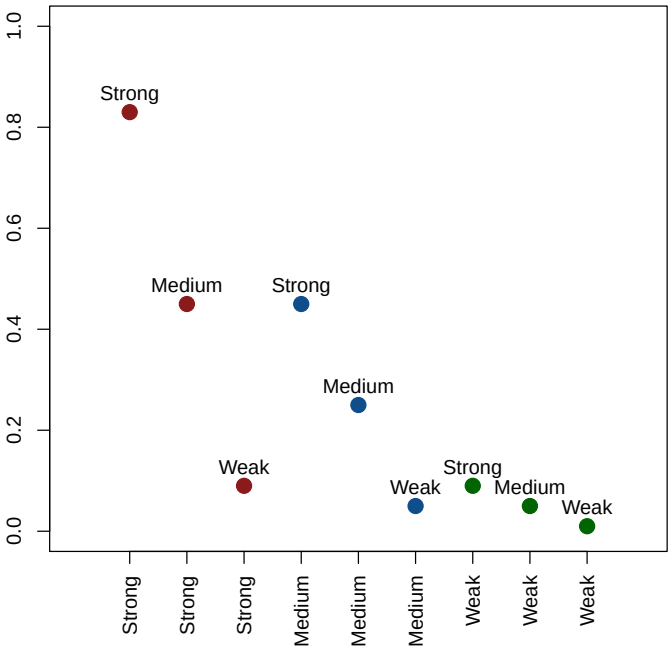

Figure 4.2.9: The product of correlations by strength of correlation.

In Fig. 4.2.9 it is evident that $\rho_{strong} \times \rho_{strong}$ produces ρ_{strong}, and $\rho_{strong} \times \rho_{medium}$ produces ρ_{medium}. Whereas $\rho_{strong/medium/weak} \times \rho_{weak}$ produces ρ_{weak}. Given that the UI approach assumes

partial exchangeability and positive association, this approach is concerned with averaging over the product of correlations. As such, there is assumed to be a higher proportion of $\rho_{strong} \times \rho_{strong}$ and $\rho_{strong} \times \rho_{medium}$ in comparison to $\rho_{strong/medium/weak} \times \rho_{weak}$.

4.2.2 Development of the Uncertainty Interval

In this section, having proposed that the information should be combined via a product, focus is given on how to modify the NPUI equations as to estimate correlation coefficients. Further consider the MVs of a sub-ratings matrix as a *unknown* relationship between two items and the goal is to estimate how these items are related to each other. By understanding how these two items are related, a recommendation may be made for the item not experienced by an individual.

The first UI to consider is simply to replace the known utilities in NPUI equations with the product of the pairwise comparisons. This is denoted $UI_{Untrans}$ as it is an untransformed version of UI and is calculated as follows:

$$\begin{aligned} \underline{\rho}_{a,b} &= \frac{1}{n+1} \sum_{i=1, i \neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right) \\ \overline{\rho}_{a,b} &= \frac{1}{n+1}\left(1 + \sum_{i=1, i \neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right)\right) \\ \Delta(\rho_{a,b}) &= \underline{\rho}_{a,b} - \overline{\rho}_{a,b} = \frac{1}{n+1} \end{aligned} \tag{4.1}$$

The $UI_{Untrans}$ equations will calculate the interval $[0, 1]$ when no information is provided, *i.e.*, vacuous information. This is in line with the assumption of exchangeability and the fact that exchangeable random variables are positively correlated. The $UI_{Untrans}$ equations do not take into account that pairwise correlations may return negative correlation coefficients for a pair of items that are actually positively correlated. Therefore, it is possible that the interval may contain negative values and the final interval produced from the pairwise comparisons may not contain the true correlation coefficient in a similar way that the correlation coefficient of pairwise correlations is not the true correlation coefficient if complete information was available. The $UI_{Untrans}$ equations may calculate an interval $(-1, -1)$ when the limit is taken. For example, consider all correlations between item a and item(s) i to be perfectly negatively correlated (*i.e.*, $\rho_{a,i} = -1$ always) and all correlations between item b and item(s) i are perfectly positively

correlated (*i.e.*, $\rho_{b,i} = 1$ always). Then, for an infinite number of pairwise comparisons the following is true:

$$\lim_{n\to\infty} \frac{1}{n+1} \sum_{i=1}^{\infty} 1 \times -1 = -1$$

$$\lim_{n\to\infty} \frac{1}{n+1} (1 + \sum_{i=1}^{\infty} 1 \times -1) = -1$$

If this occurred, the assumption of exchangeability has been violated.

As the product of two correlations is calculated, *i.e.*, $\rho_i \in (-1, 1) \times \rho_j \in (-1, 1)$, the magnitude of estimated lower and upper intervals are reduced and the sign of the product is dependent on the sign of the correlation coefficients. In our example, as there are no negative correlations utilised in generating an interval to estimate the true correlation coefficient, the outcome is the same as Fig. 4.2.8.

However, this would not be the case if ρ_{m_1,m_2} was a MV as $\rho_{m_1,m_5} = -0.12$. Applying the $UI_{Untrans}$ algorithm to estimate ρ_{m_1,m_2} generates a final interval (.09,0.34), see Fig. 4.2.10. In this instance ρ_{m_1,m_2} is known and it is evident that the negative correlation ρ_{m_1,m_5} impacts on the lower and upper bound equations. Ideally the algorithm will provide bounds for an unknown relationship between items a and b in [0, 1] as exchangeable random variables have a correlation coefficient greater or equal to 0.

	m_1	m_2	m_3	m_4	m_5
m_1	1	MV	0.13	0.92	−0.12
m_2	MV	1	0.96	0.36	0.62
m_3	(0.13	× 0.96)	1	MV	0.97
m_4	(0.92	× 0.36)	MV	1	0.36
m_5	(−0.12	× 0.62)	0.97	0.36	1

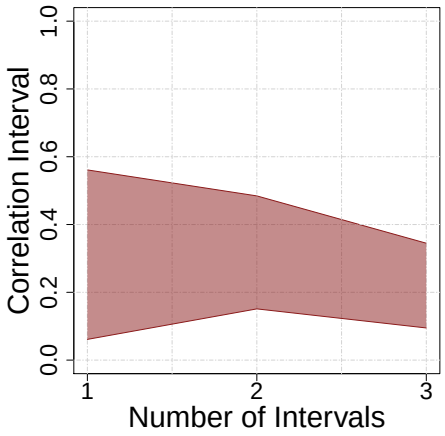

Figure 4.2.10: Product of each pairwise comparison with a negative correlation.

A correlation coefficient delivers three pieces of information:

1. Whether two variables are linearly related to one another;
2. Whether the relationship is positive or negative and;
3. The magnitude of the relationship.

Knowing the magnitude of the relationship may be more informative than knowing the sign of the correlation. Therefore, the absolute value of the product of pairwise comparisons can be incorporated into the UI approach. This is termed UI_{Abs} and is represented in Eq. (4.2):

$$\begin{aligned} \underline{\rho}_{a,b} &= \frac{1}{n+1} \sum_{i=1, i\neq a,b}^{n} \left(|\rho_{a,i} \times \rho_{b,i}| \right) \\ \overline{\rho}_{a,b} &= \frac{1}{n+1} \left(1 + \sum_{i=1, i\neq a,b}^{n} \left(|\rho_{a,i} \times \rho_{b,i}| \right) \right) \\ \Delta(\rho_{a,b}) &= \underline{\rho}_{a,b} - \overline{\rho}_{a,b} = \frac{1}{n+1} \end{aligned} \tag{4.2}$$

As with $UI_{Untrans}$, the product of the two correlations will reduce the magnitude of the relationship. However, as all values are in absolute form, the sign will always be positive. Often RSs are interested in positive relationships, as these are the basis for recommendation, therefore it would appear justifiable to only consider positive correlations. Further justification to take the absolute value of the pairwise correlations comes for the calculations of these correlations. A pairwise correlation is used to deal with missing data and computes the correlation coefficient for each *pair* of columns using vectors generated by omitting rows with missing values on a *pairwise* basis. According to Lewis (2015), this may not be a reasonable approach with known unknowns being converted to unknown knowns. Therefore, even if the underlying data is highly correlated, applying a pairwise comparison may lead to zero or even high negative correlations.

Going back to the example with a negative correlation, the final interval is now calculated as (0.20,0.45). In this instance, taking the absolute value of the pairwise comparison better reflects the true correlation coefficient of ρ_{m_1,m_2}, see Fig. 4.2.11.

	m_1	m_2	m_3	m_4	m_5
m_1	1	MV	0.13	0.92	−0.12
m_2	MV	1	0.96	0.36	0.62
m_3	$\lvert 0.13 \times 0.96 \rvert$		1	MV	0.97
m_4	$\lvert 0.92 \times 0.36 \rvert$		MV	1	0.36
m_5	$\lvert -0.12 \times 0.62 \rvert$		0.97	0.36	1

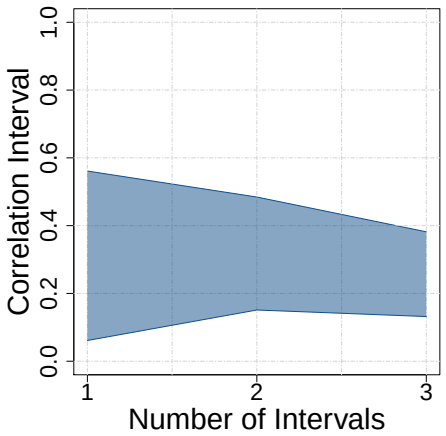

Figure 4.2.11: Product of each absolute pairwise comparison.

In a similar approach to UI_{Abs}, ρ^2 is considered in order to produce intervals between [0,1]. The *coefficient of determination* is another term used for ρ^2 and is used to explain the ratio of two variances, *i.e.*, the proportion/percentage of variability accounted for (Nagelkerke, 1991). It represents the linear relationship between two or more variables and that is why it is often reported in regression analysis. This is denoted UI_{Sq} with the lower and upper bounds calculated as follows:

$$\begin{aligned} \underline{\rho}_{a,b} &= \frac{1}{n+1} \sum_{i=1, i\neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right)^2 \\ \overline{\rho}_{a,b} &= \frac{1}{n+1}\left(1 + \sum_{i=1, i\neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right)^2\right) \\ \Delta(\rho_{a,b}) &= \underline{\rho}_{a,b} - \overline{\rho}_{a,b} = \frac{1}{n+1} \end{aligned} \tag{4.3}$$

This approach also scales the correlation coefficient between [0,1]. However, calculating UI_{Sq} produces a lower final interval than UI_{Abs} as, not only is $\rho \in [-1, 1]$ squared, it is then multiplied by a value $\in [0, 1]$. Returning to our example incorporating a negative correlation, the final interval is (0.03,0.28).

	m_1	m_2	m_3	m_4	m_5
m_1	1		0.42	0.13	0.92	−0.12
m_2	0.42		1	0.96	0.36	0.62
m_3	$(0.13$	$\times$	$0.96)^2$	1	MV	0.97
m_4	$(0.92$	$\times$	$0.36)^2$	MV	1	0.36
m_5	$(-0.12$	$\times$	$0.62)^2$	0.97	0.36	1

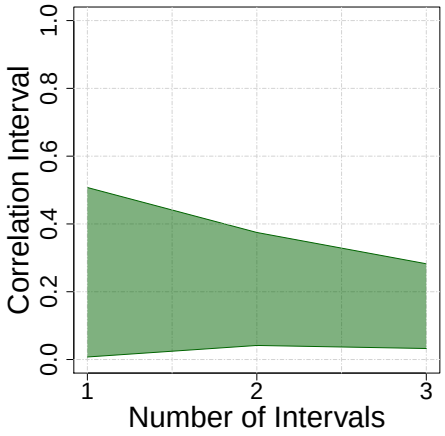

Figure 4.2.12: Product of each squared pairwise comparison.

The last approach considered is UI_{Rt}. In order to transform the intervals produced in Eq. (4.4) the square root is taken. This is similar to a transformation in linear regression, as the output is interpreted in its original form. This approach also scales the correlation coefficient between [0,1], but adjusts the lower and upper bounds so that the information about

how a is related to i and how b is related to i is better reflected. As we can see from Fig. 4.2.13, the interval is now (0.18, 0.53).

$$\begin{aligned} \underline{\rho}_{a,b} &= \sqrt{\frac{1}{n+1} \sum_{i=1, i \neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right)^2} \\ \overline{\rho}_{a,b} &= \sqrt{\frac{1}{n+1}\left(1 + \sum_{i=1, i \neq a,b}^{n} \left(\rho_{a,i} \times \rho_{b,i}\right)^2\right)} \\ \Delta(\rho_{a,b}) &= \underline{\rho}_{a,b} - \overline{\rho}_{a,b} \leq \frac{1}{n+1} \end{aligned} \tag{4.4}$$

	m_1	m_2	m_3	m_4	m_5	
m_1	1		0.42	0.13	0.92	−0.12
m_2	0.42		1	0.96	0.36	0.62
m_3	$(0.13$	$\times$	$0.96)^2$	1	MV	0.97
m_4	$(0.92$	$\times$	$0.36)^2$	MV	1	0.36
m_5	$(-0.12$	$\times$	$0.62)^2$	0.97	0.36	1

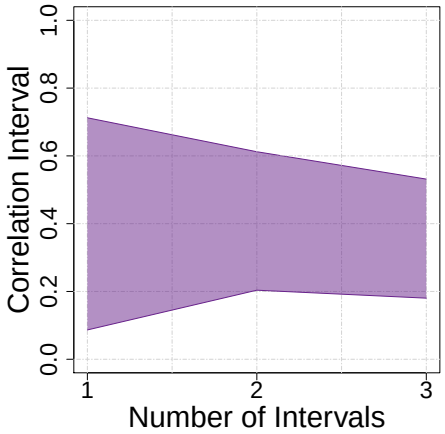

Figure 4.2.13: Product of each root pairwise comparison.

The four UI approaches are considered heuristic in a recommender setting in that they make a rating prediction based on the entire collection of previously rated items by users. These algorithms also represent relational learning as described in RFT, *i.e.*, certain relationships are known (trained) and it is possible to derive relationships between unknown (untrained) items. Recall that there were three challenges outlines in Chapter 1. The challenge of incorporating vagueness and uncertainty into recommendations is addressed in the interval estimate of UI approaches, the wider the interval the more vague and/or uncertain the system is about how two items are related. In addition, the more information available the narrower the interval becomes representing a form of learning. Thus, there is transparency in the system as the system is *learning* preferences based on explicit feedback and displaying this knowledge as an interval which represents the amount of knowledge in the system. Finally, utilising a single data source is, in and of itself, a challenge. The UI approaches are capable of estimating an interval that is representative of the true correlation coefficient from a single ratings matrix.

4.3 Chapter Summary

This chapter discussed how the RBCL technique is used to calculate an interval estimate for a correlation coefficient. The limitations and potential issues related to this technique was highlighted before introducing our proposed method. In order to develop our technique, a number of issues needed to be addressed. Firstly, how to combine information about how item a relates to item(s) i and how item b relates to item(s) i. The next challenge was to modify the existing NPUI equations in order to calculate an interval that would estimate the unknown correlation $\rho_{a,b}$. The four proposed UI algorithms are UI_{Untran}, UI_{Abs}, UI_{Sq} and UI_{Rt}. These are to be evaluated and compared with the RBCL approach using a distance and percent metrics, as well as by the number of pairwise comparisons in each set of intervals.

The following chapter introduces the evaluation methods and simulated data, describing the method used to simulate complete and sparse matrices. The four UI approaches are also evaluated and comparisons made with RBCL where possible.

5

Performance Evaluation via Simulation

In this chapter the method used to evaluate and simulate a ratings matrix are discussed. A single ratings matrix is generated, allowing for the evaluation of the UI approaches and a comparison, where possible, with the RBCL technique. Next, the UI approaches are applied to matrices with differing amounts of missing variables and evaluated. Finally, matrices are simulated from various distributions and the application of the UI approach on these are explored and evaluated. The aim of this chapter is to distinguish which of the UI approaches performs most accurately.

5.1 Evaluation of the UI approaches

Within a RS setting, algorithms and statistical techniques are often evaluated by calculating the Mean Absolute Error (MAE), which computes the deviation between predicted ratings and actual ratings, and the Root Mean Square Error (RMSE), which also computes the deviation between predicted ratings and actual ratings but places more emphasis on larger deviation. However, the UI approach is not a RS in and of itself. It is an algorithm that estimates an interval for a missing value and should be evaluated as such. Therefore, in order to evaluate the UI algorithm, the imputation of the MVs is evaluated.

In Newman (2003), the author related to modelling and simulating missing data. The average error of the parameter estimates, or missing data error (correlation in this instance), is calculated as the mean absolute difference between estimates derived from complete data and those derived from missing data techniques (the UI approach). As the four UI approaches have

a lower and upper bound they are evaluated by calculating the absolute minimum distance the final lower and upper bound is from the true correlation coefficient:

$$\text{Distance} = \sum \frac{\min |\text{complete data coefficient} - \text{incomplete data coefficient}|}{n}$$

The percent of the total number of intervals that the true correlation coefficient is contained inside the UI intervals is also recorded. As the UI approach will be evaluated alongside the RBCL technique, this is another way to find similarities/differences between the two. In order to calculate the percentage, let

$$x_i = \begin{cases} 1 & \text{if the true correlation coefficient is contained within the interval} \\ 0 & \text{otherwise} \end{cases}$$

Then the percentage is expressed as:

$$\text{Percent} = \left(\sum_i \frac{x_i}{n} \right) \times 100$$

Finally, again to draw comparisons between the UI approach and the RBCL technique, the number of pairwise comparisons that are utilised in generating a final interval is recorded. In addition to drawing comparisons, this number allows for the interpretation of how informative the UI interval is. That is, a narrower interval indicates more information has been corporated into building the final interval, thus is more informative. This also allows for the calculation of the percent of missingness in each correlation matrix, as well as the percent of missingness that remains once the UI algorithms are run.

5.2 Simulation of Correlated Data

The first matrix considered is one which is exchangeable and positively correlated. The following is a procedure for randomly generating correlated raw variables, there is 1,000 simulations for each UI algorithm. These raw variables are simulated, $R_{i.} \sim MVN(\mu, \Sigma)$, such that the vector μ contains only zeros and the covariance matrix Σ contain values between .95 and .99 to ensure

high correlation[1]. The raw variables generated formed a matrix which has no missing variables and is positive definite, it contains 500×500 entries and is denoted $C_{complete}$. In an ideal setting few or no variables would be missing and so the calculated correlation coefficient would be representative of the data and these positive correlations are calculated using all the data and are exchangeable. To illustrate this, in Table 5.1 the first ten correlations generated for items one to ten of a randomly generated matrix, which will act as our example matrix, is displayed.

Table 5.1: First ten correlations calculated from $C_{complete}$

item	1	2	3	4	5	6	7	8	9	10
1	1.00									
2	0.90	1.00								
3	0.89	0.90	1.00							
4	0.89	0.90	0.89	1.00						
5	0.90	0.90	0.88	0.91	1.00					
6	0.89	0.91	0.88	0.92	0.90	1.00				
7	0.89	0.90	0.89	0.90	0.89	0.89	1.00			
8	0.90	0.90	0.91	0.91	0.91	0.90	0.91	1.00		
9	0.91	0.91	0.89	0.90	0.89	0.90	0.91	0.90	1.00	
10	0.91	0.90	0.89	0.90	0.91	0.89	0.90	0.91	0.90	1.00

In reality rating matrices are sparse, they consist of few variables and any correlations calculated are done so on a pairwise basis. In order to create a matrix that would reflect a real world setting, a mask matrix was generated at random. A mask matrix is generated , *i.e.*, a matrix comprising of success=1 and failure=0 such that a Hadamrad (entry-wise) product with a complete matrix produces a new matrix with missing values=0. This allowed the generation of a matrix with a percentage of missing values (MVs). Real world data sets, such as MovieLens, Netflix and YouTube, contain a low proportion of observed ratings; this proportion is typically between 1% and 5% (Marlin *et al.*, 2011).

A number of considerations is made when generating the missingness of data, see Section 2.8 for formal definitions of patterns of missingness. One possible approach is to generate Missing At Random (MAR) data which assumes the pattern of missingness depends on the value of the data that is missing. However, the UI approach assumes the data is exchangeable and that there is a strong relationship between the items in the ratings sub-matrix. Therefore, the missingness does not depend on the values of the data that are missing. Another approach

[1]All R code for generating simulated data is presented in the appendix.

is Not Missing At Random (NMAR), which assumes the pattern of missingness is due to a variable, for example genre. Given that the UI approach assumes partial exchangeability; generating NMAR data would be similar to generating a ratings matrix and then creating a sub-matrix in order to satisfy the partial-exchangeability assumption. As collaborative filtering algorithms may assume missingness is Missing Completely At Random (MCAR) (Su *et al.*, 2008) the sparse matrix generated in this section is MCAR. The probability of success in the mask matrix was set to 0.05 and so we have a matrix with approximately 95% MVs, this matrix is defined as $C_{5\%}$. Next, pairwise correlations are calculated. Using a pairwise calculation can result in correlation matrices which are not positive semi-definite[2]. In addition, only paired correlations are utilised, if there are no pairs then a MV is produced. The first ten of these correlations in our example are displayed in Table 5.2, from which it is evident that such a sparse matrix leads to no pairwise comparisons being computed in some cases, thus resulting in MVs.

Table 5.2: First ten correlations calculated from $C_{5\%}$

Item	1	2	3	4	5	6	7	8	9	10
1	1.00									
2	0.86	1.00								
3	MV	1.00	1.00							
4	MV	-0.67	1.00	1.00						
5	0.88	0.98	MV	MV	1.00					
6	MV	0.93	MV	MV	-1.00	1.00				
7	MV	MV	MV	1.00	-1.00	0.97	1.00			
8	0.99	1.00	MV	0.93	-1.00	MV	0.91	1.00		
9	MV	MV	MV	MV	MV	MV	MV	0.63	1.00	
10	MV	MV	MV	-0.98	-1.00	MV	MV	1.00	MV	1.00

[2]As such a pairwise calculation may lead to negative eigenvalues which may imply negative variance - which is a contradiction of the definition of variance.

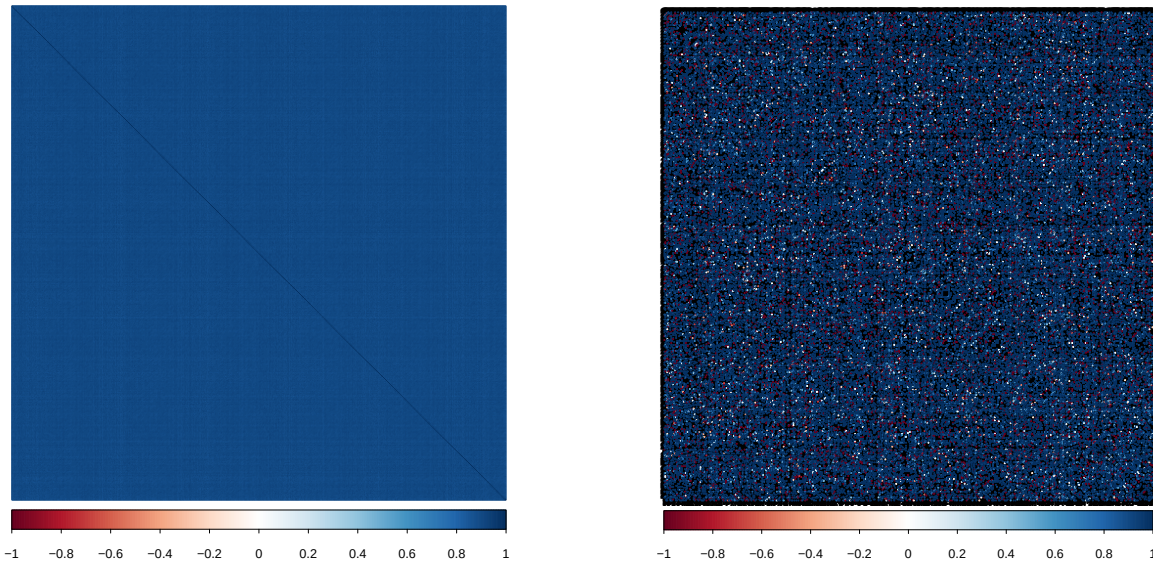

Figure 5.2.1: (a) Correlation Plots: $C_{complete}$ correlations; (b) Correlation Plots: $C_{5\%}$ correlations

In Fig. 5.2.1 (a) the correlations produced by the completed data are strongly positive and uniform in colour, whereas, in the correlations produced from the sparse data (Fig. 5.2.1 (b)) there is a combination of positive and negative, weak and strong correlations with a non-uniform pattern displayed.

While $C_{complete}$ generates high positive correlations between all the variables, this is not the case in $C_{5\%}$. Instead, in addition to MVs, high negative correlations are produced, see Fig. 5.2.1(b). A high negative correlation may be the result of only two pairwise comparisons, leading to a line fitted in a negative direction. A single pairwise comparison in a ratings matrix may lead to a zero correlation coefficient as the fitted line has only one point to pass through. In addition, if there is a pairwise comparison, this may result in a missing value if all values are the same, *i.e.*, variance would be zero. As stated, this is one of the limitations of using pairwise correlations. From $C_{5\%}$ a pairwise correlation produced a matrix with approximately 64% missing values. Therefore, the RBCL approach could not be employed to calculate an interval estimate of the true correlation coefficient for these missing values.

5.2.1 $C_{complete}$: Complete Data

In order to evaluate the UI approaches and to make comparisons with the RBCL technique, the example matrix described above is employed. The four UI approaches and the RBCL technique are used to estimate the lower and upper bounds of two correlations:

1. The correlation between item 1 and item 2, $\rho_{C_{complete}}(1,2)$: when missing values are introduced, a pairwise correlation is calculated. The UI approaches and the RBCL technique will be applied to estimate the true correlation value of item 1 and item 2.
2. The correlation between item 1 and item 3, $\rho_{C_{complete}}(1,3)$: when missing data is simulated, a pairwise correlation produced a missing value (MV). In this instance, only the UI approaches can be applied to estimate the true value of the correlation coefficient between item 1 and item 3.

Firstly, $\rho_{C_{complete}}(1,2)$ is designated as a MV and the $UI_{Untrans}$, UI_{Abs}, UI_{Sq} and UI_{Rt} are calculated to estimate the true value of the correlation coefficient, $\rho_{C_{complete}}(1,2) = .90$. In order to evaluate the four UI approaches, the RBCL algorithm is also calculated. The hyperparameter ν_0 was set to the default of 2 for the calculation of posterior expected covariance matrix.

Fig. 5.2.2 displays the performance of the UI approaches to estimate an interval for $\rho_{C_{complete}}(1,2)$. The UI approaches generate intervals that do not take negative values. The performance of the RBCL technique is displayed in Fig. 5.2.3, where it is obvious that the first interval generated is $[-1,1]$. Both the UI and RBCL techniques quickly narrow as more information is introduced. However, unlike the RBCL technique, each of the UI approaches underestimate the true correlation coefficient. In addition, the UI approach utilises pairwise correlations and, given the sparsity of the matrix, these pairwise correlations may produce negative or zero correlations, even though the true underlying correlation coefficient is strongly positive. Hence, these correlations contain errors as demonstrated in Section 3.3.2.

The RBCL builds an interval based on paired observations, it generates a paired correlation for the first set of intervals and continues to update the correlation coefficient as more and more paired sets are entered into the model. Therefore, this technique does not contain the errors generated by pairwise correlations on missing values. In addition, the final interval calculated using the RBCL approach contains the true correlation coefficient.

As this is a complete matrix simulation, the number of pairwise comparisons utilised in building the final interval for the UI approaches and the RBCL technique is 499. This in turn allows for the calculation of the interval, which is 0.002. Therefore, the level of informativeness of these intervals is high.

Repeating the process to calculate the correlation coefficient between item 1 and item 3

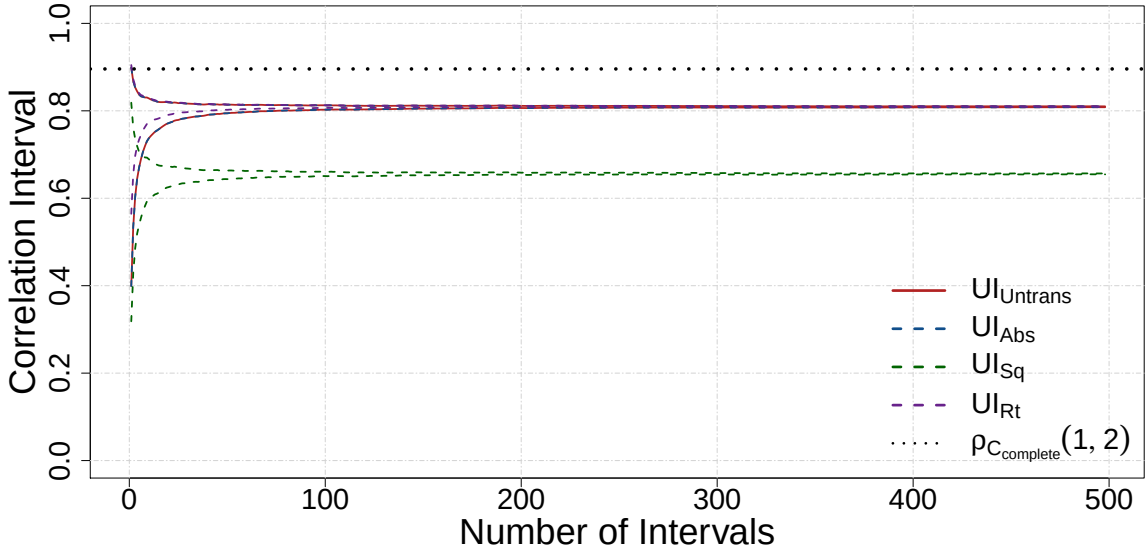

Figure 5.2.2: Application of Eq. 4.1, Eq. 4.2, Eq. 4.3 and Eq. 4.4 to estimate $\rho_{C_{complete}}(1,2)$

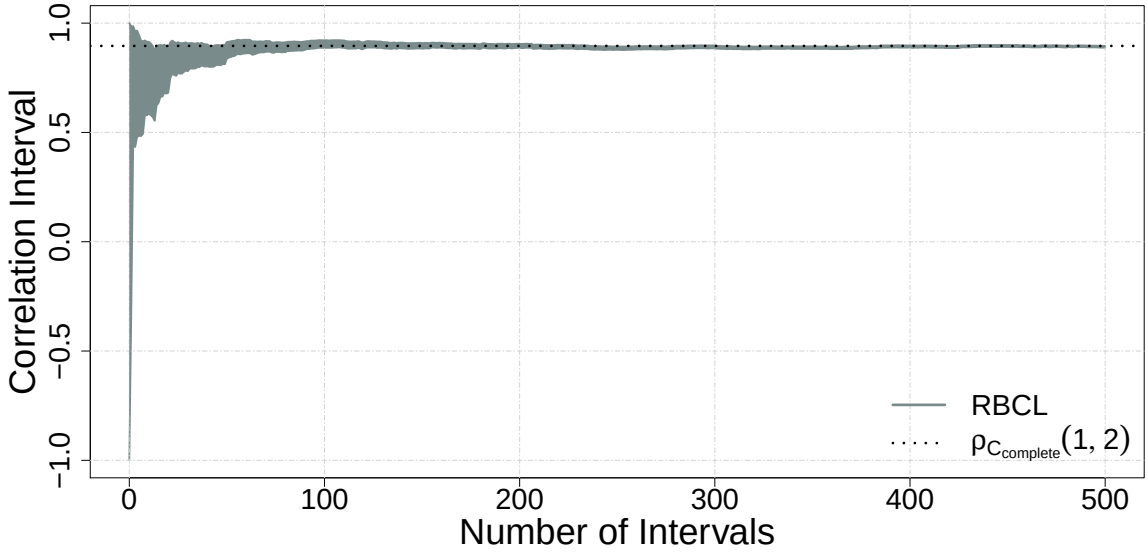

Figure 5.2.3: Application of RBCL technique to estimate $\rho_{C_{complete}}(1,2)$

produces very similar results as the previous example, see Fig. 5.2.4 for UI approaches and Fig. 5.2.5 for RBCL technique. A summary of the distance metric, *i.e.*, the absolute minimum distance the true correlation lies outside the interval, is displayed in Table 5.3. This table also contains the percentage of intervals that contain the true correlation coefficient. The RBCL technique contains not only the true correlation coefficient in the final interval, but also has the highest percentage of intervals that contain the true correlation coefficient. Hence, given complete information, the RBCL technique is superior to the UI approaches.

A total of 1,000 matrices were simulated in the above manner and a randomly selected entry was assigned MV. All four UI algorithms and the RBCL technique were applied to estimate

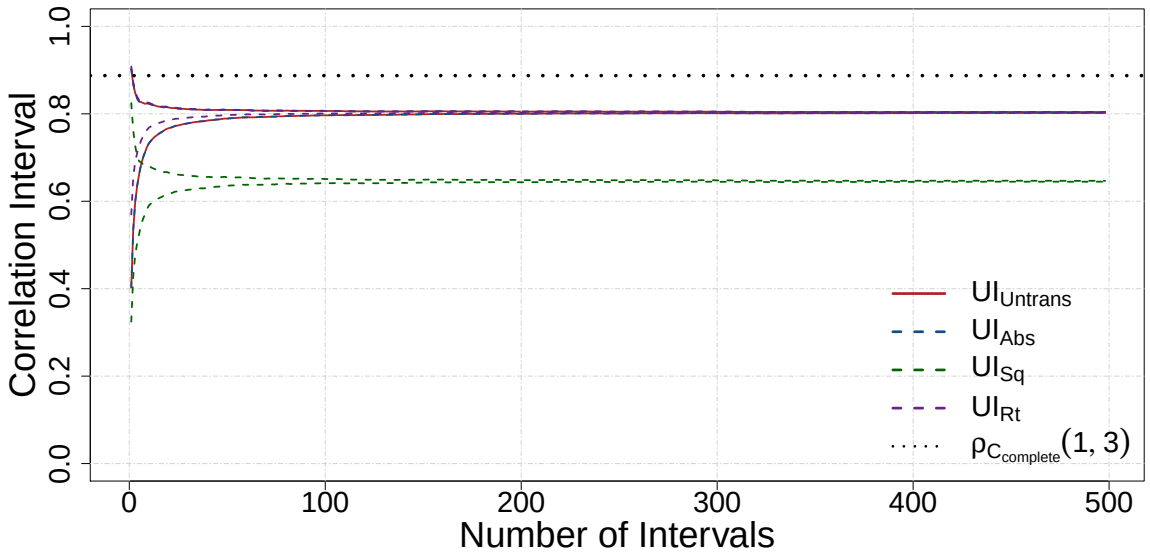

Figure 5.2.4: Application of Eq. 4.1, Eq. 4.2, Eq. 4.3 and Eq. 4.4 to estimate $\rho_{C_{complete}}(1,3)$

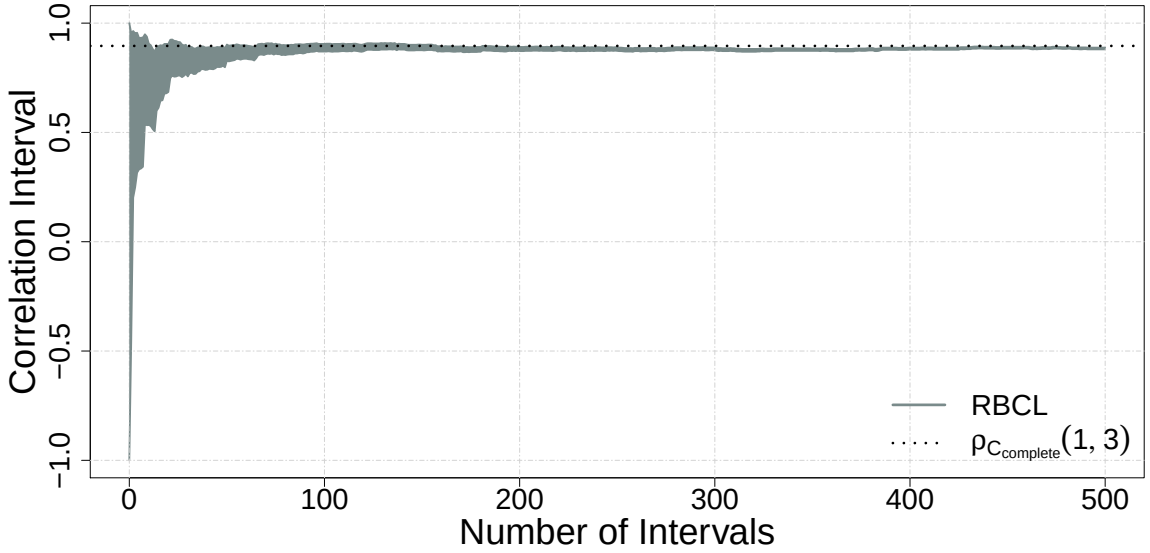

Figure 5.2.5: Application of RBCL technique to estimate $\rho_{C_{complete}}(1,3)$

Table 5.3: Final Intervals for Estimated Correlation $\rho_{C_{complete}}(1,2)$ and $\rho_{C_{complete}}(1,3)$

	$\boldsymbol{\rho_{C_{complete}}(1,2)}$				$\boldsymbol{\rho_{C_{complete}}(1,3)}$			
	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Distance	Percent	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Distance	Percent
$\boldsymbol{UI_{Untrans}}$	0.81	0.81	0.09	0.20	0.80	0.80	0.08	0.20
$\boldsymbol{UI_{Abs}}$	0.81	0.81	0.09	0.20	0.80	0.80	0.08	0.20
$\boldsymbol{UI_{Sq}}$	0.66	0.66	0.24	0.00	0.65	0.65	0.24	0.00
$\boldsymbol{UI_{Rt}}$	0.81	0.81	0.09	0.20	0.80	0.80	0.08	0.20
RBCL	0.89	0.90	**0**	74.85	0.89	0.90	**0**	76.25

Table 5.4: Summary statistics of 1,000 simulations of $C_{complete}$

	$UI_{Untrans}$		UI_{Abs}		UI_{Sq}		UI_{Rt}		RBCL	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	.00	.06	.00	.06	.00	.21	.00	.06	63.40	0
Max	.40	.21	.40	.21	.00	.38	.40	.21	78.20	0
Median	.20	.09	.20	.09	.00	.25	.20	.09	70.80	0
Mean	.14	.09	.14	.09	.00	.25	.17	.09	71.00	0

the true correlation coefficient of the MV and a summary of 1,000 simulations of a complete correlation matrix with a MV random generated and estimated can be found in Table 5.4. Notably, the intervals generated by UI_{Sq} never contain the true correlation coefficients and its absolute average distance from the true correlation coefficients is higher than the other three UI approaches. $UI_{Untrans}$, UI_{Abs} and UI_{Rt} all preform almost identically when applied to a complete dataset. This reinforces the findings from the above examples of $\rho_{C_{complete}}(1,2)$ and $\rho_{C_{complete}}(1,3)$ that the RBCL techniques has superior performance when applied to complete data. There is little insight as to which UI approach performs best when applied to a complete correlation matrix. The following section explores their application to sparse matrices and evaluates the performance in relation to percentage of overall intervals that contain the true correlation coefficient and the absolute minimum distance from the interval to the true correlation coefficient.

5.2.2 $C_{5\%}$: Sparse Data

The correlation between item 1 and item 2 is $\rho_{C_{complete}}(1,2) = .90$ from $C_{complete}$ and $\rho_{C_{5\%}}(1,2) = .86$ from $C_{5\%}$. The pairwise correlation $\rho_{C_{5\%}}(1,2)$ is set to MV in order to explore how the UI approaches perform. The aim of the UI approach is to estimate $\rho_{C_{complete}}(1,2)$ using the products of the correlations produced from $C_{5\%}$.

As the UI technique is a pairwise comparison method, it should be noted that it calculates products of pairwise correlation coefficients. In Fig. 5.2.6 the first interval calculated by a UI approach, to estimate $\rho_{C_{complete}}(1,2)$, incorporates a pairwise correlation that consisted of two pairwise comparisons multiplied by a pairwise correlation that consisted of three pairwise

comparisons. Therefore, not only does the number of pairwise comparisons used to construct a lower and upper bound affect the final interval, but also the number of pairwise comparisons used for the correlation calculations. When the product of two correlations produces a MV, this value is removed and not utilised in the building of the intervals.

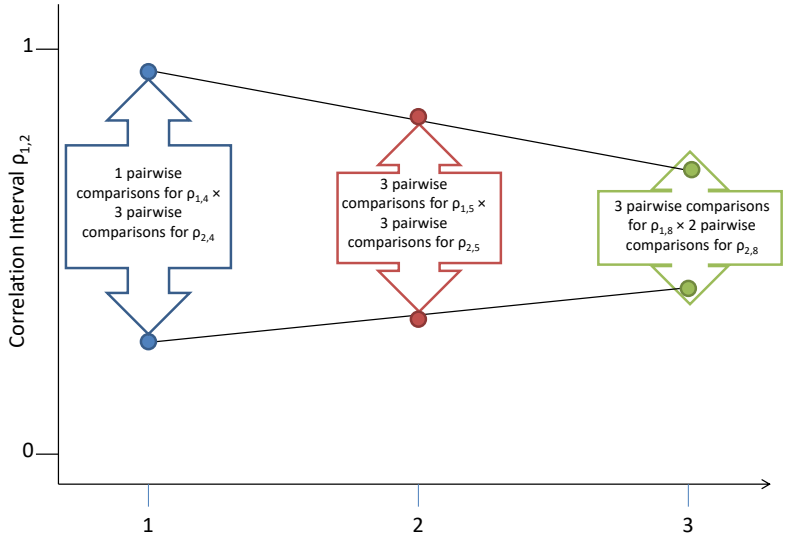

Figure 5.2.6: Graph showing the first three intervals to estimate $\rho_{C_{complete}}(1,2)$

As the UI approaches are primarily heuristics, it is necessary to explore how these approaches work. In Fig. 5.2.7 the correlation coefficients for item 1 are plotted against the correlation coefficients for item 2 for both complete (displayed as red points) and sparse (displayed as blue points) correlation matrices. It is evident, in this example, that in the complete correlation matrices for untransformed, absolute and squared correlations, the points are all clustered closely together in the top right corner of each graph, indicating strong positive correlations. Looking at the correlation produced from the sparse data set, even through there are some strong negative pairwise correlations for item 1 and item 2, the majority of item 1 pairwise correlations versus item 2 correlations are plotted in the top right corner, indicating both sets of correlations are strong positive. In the current example, 86% of the sparse correlation matrix contains values greater than 0.7. The untransformed correlations can contain values between -1 and 1, whereas the absolute and squared correlations are contained within a [0,1] interval. The points are more *spread out* in the squared transformation. Therefore, the majority of paired correlations are representative of the true correlation coefficient.

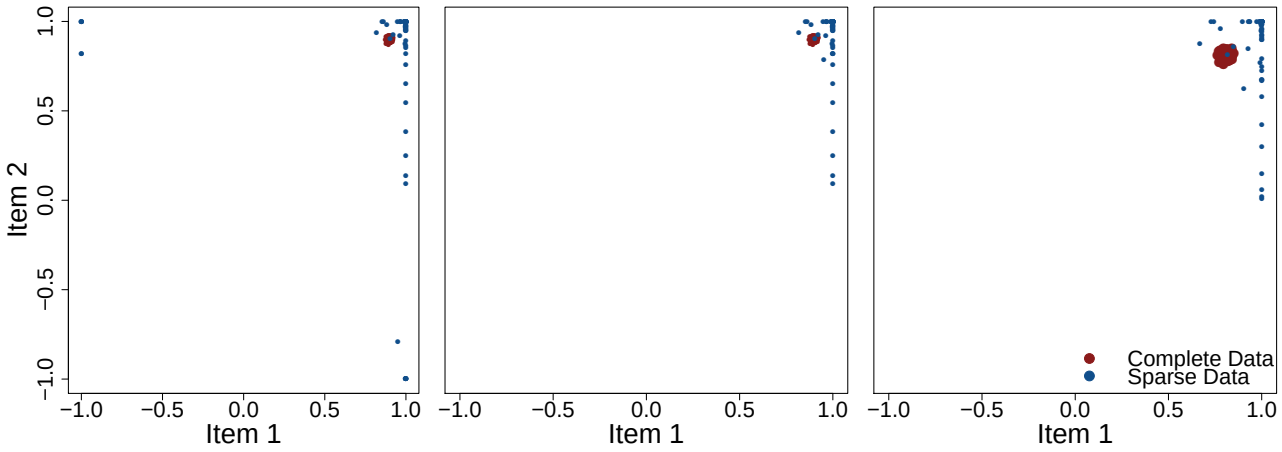

Figure 5.2.7: $\rho(1,i)$ vs $\rho(2,i)$ for complete and sparse data. From left to right we have (a) Untransformed correlations (b) Absolute correlations (c) Squared correlations.

The above graphs display both complete and sparse correlations for items 1 and 2. It is perhaps more relevant to explore the correlation matrix produced from the sparse matrix $C_{5\%}$ in some more detail. In Fig. 5.2.8, the untransformed pairwise correlations produced for item $a = 1$ and all other items are plotted on the left-hand side of the graph whilst the pairwise correlations produced for item $b = 2$ and all other items are plotted on the right-hand side of the graph. The connecting lines indicate a pairwise comparison and clearly there exists correlations for item 2 with some item i where there isn't a corresponding correlation for item 1, as indicated by no line. In this example, when there is a negative pairwise correlation between item a and item i, it is paired with a positive correlation between item b and item i and *vice versa*. Recall that the pairwise correlations are assumed to be exchangeable and, even though a negative pairwise correlation is produced, it is not reflective of the true correlation coefficient given complete information. In addition, 86% of the pairwise correlations are 0.7 or above, which is more reflective of the true correlation coefficient given complete information. Hence, positive pairwise correlations should outnumber negative pairwise correlations and once paired these negative pairwise correlations are mitigated.

To further mitigate the effect of these negative correlations, the absolute value of each correlation is calculated and these are displayed in Fig. 5.2.9. In this instance, the values the pairwise correlations take are between 0 and 1. There are still pairwise correlations that, although not negative, are indicating no correlation between item 2 and some item(s) i. As before, these correlations are paired with strong positive correlation for item 1. Another approach to mitigate

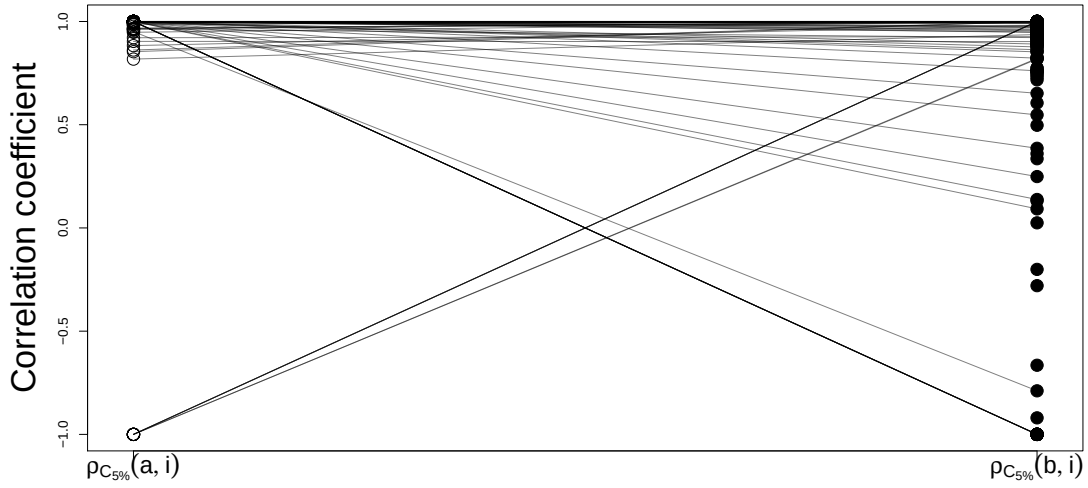

Figure 5.2.8: Untransformed paired correlation comparisons utilised in estimating $\rho_{complete}(1,2)$

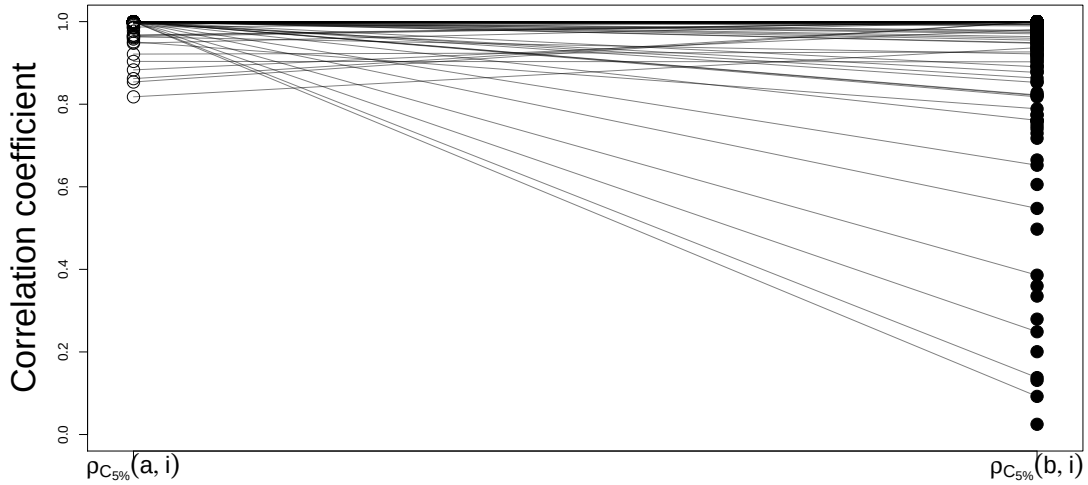

Figure 5.2.9: Absolute paired correlation comparisons utilised in estimating $\rho_{complete}(1,2)$

the affect of these negative correlations is to square each correlation. In Fig. 5.2.10 it is evident that this causes strong positive pairwise correlations to become less so and the *spread* is more evident in this graph. Therefore, the UI_{Abs} is superior to the UI_{Sq} as it manages to remove the effects of negative pairwise correlations without comprising the high pairwise correlations.

Applying the four UI approaches to estimate $\rho_{complete}(1,2)$, there are 77 paired correlations that are inputted into building the intervals. Therefore the width of the final interval is 0.012. In Fig. 5.2.11 the intervals for each UI approach is displayed. The black horizontal line represents the true correlation coefficient as calculated from the complete raw data matrix, whilst the dashed black line represents the pairwise correlation coefficient as calculated from the sparse raw data matrix. $UI_{Untrans}$ underestimates the true correlation coefficient, as does UI_{Sq}; this is

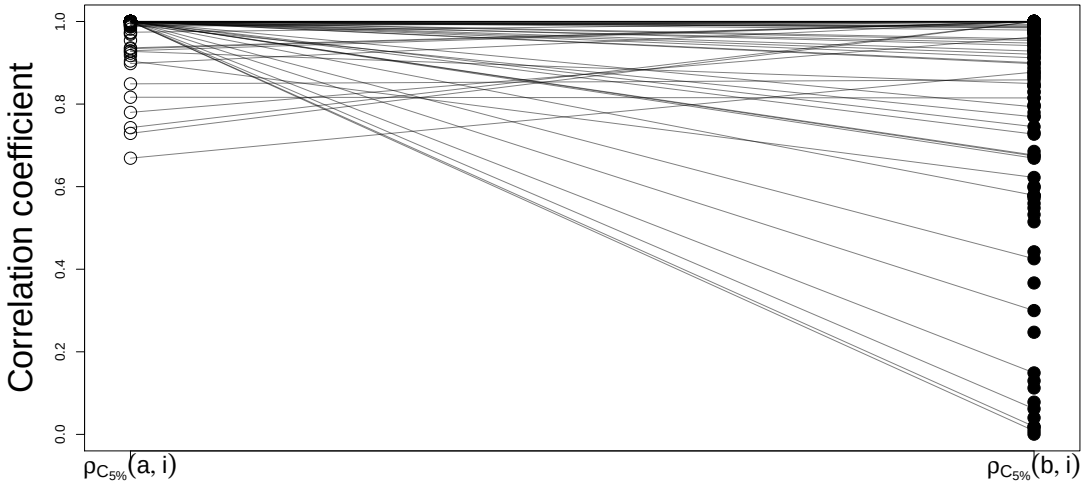

Figure 5.2.10: Squared paired correlation comparisons utilised in estimating $\rho_{complete}(1,2)$

reflective of the effect negative pairwise correlations and the *spreading* effect respectively. The final interval for UI_{Abs} contains the true correlation coefficient, with 98.7% of the total number of these intervals containing the true correlation coefficient. The UI_{Rt} performs slightly worse in that it is a distance of .02 away from the true correlation coefficient at the last interval. However, it contains the true correlation coefficient only 13% of the intervals. When the RBCL technique is applied to estimate $\rho_{complete}(1,2)$, a total of three intervals are built, see Fig. 5.2.12. The final interval, although it contains both $\rho_{complete}(1,2)$ and $\rho_{5\%}(1,2)$, is wide ranging from -.068 to .931.

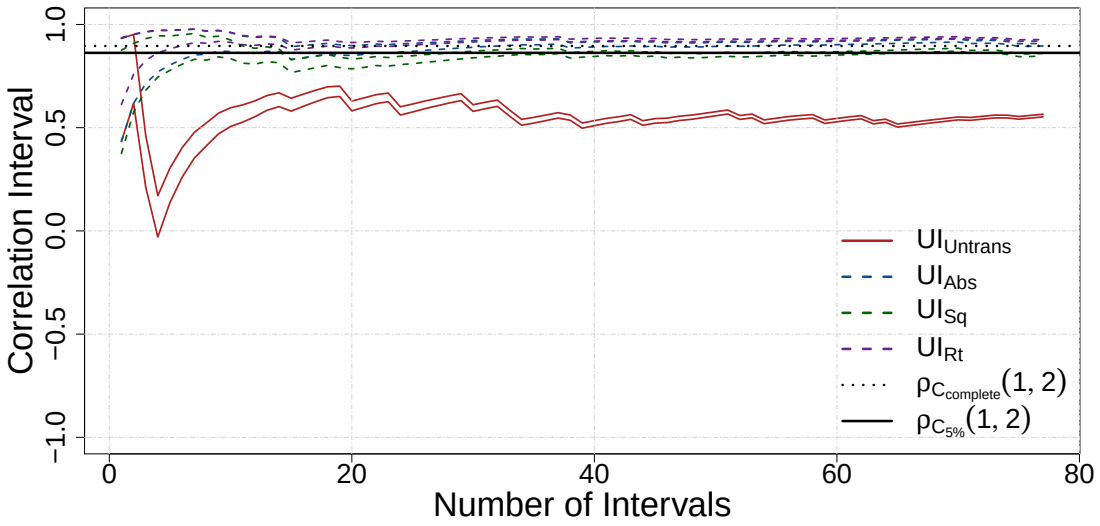

Figure 5.2.11: Application of Eq. 4.1, Eq. 4.2, Eq. 4.3 and Eq. 4.4 to estimate $\rho_{C_{complete}}(1,2)$

When the UI approaches are applied to estimate $\rho_{complete}(1,3)$ the results are similar to

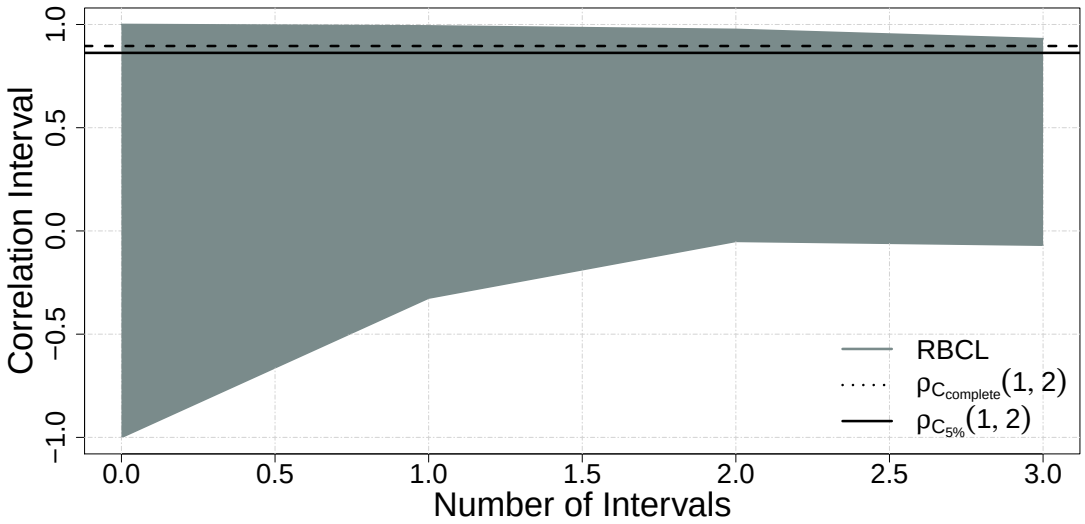

Figure 5.2.12: Application of RBCL to estimate $\rho_{C_{complete}}(1,2)$

those of $\rho_{complete}(1,2)$. The most notable difference is the number of paired correlations used to build these intervals, in this instance 18. $UI_{Untrans}$ underestimates the true correlation coefficient, whereas the other three UI approaches overestimate it. The RBCL technique is unable to estimate the true correlation coefficient in this instance are there are no paired raw data, hence no pairwise correlation can be calculated.

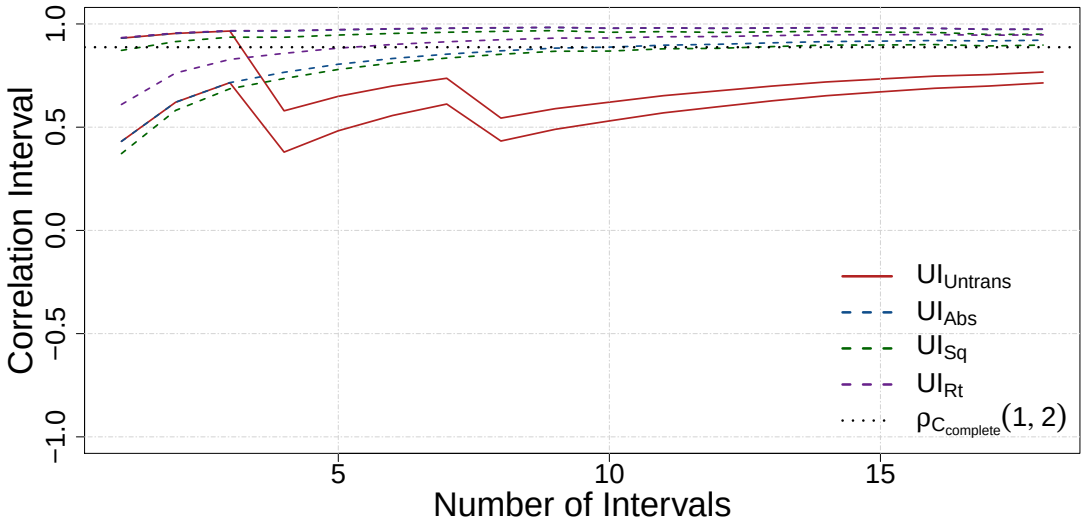

Figure 5.2.13: Application of Eq. 4.1, Eq. 4.2, Eq. 4.3 and Eq. 4.4 to estimate $\rho_{C_{complete}}(1,3)$

In Table 5.5 the final lower and upper bounds for $\rho_{C_{complete}}(1,2)$ and $\rho_{C_{complete}}(1,3)$ are displayed, as well as the minimum distance of the interval from the true correlation coefficient and the percentage of intervals that this correlation was contained within the interval. In this example, the UI intervals are narrower than the RBCL interval for $\rho_{C_{complete}}(1,2)$. This is the result of the number of paired correlations available to build the UI intervals ($n = 77$) compared to the number of paired raw variables available to build the RBCL interval ($n = 3$). Therefore, the width of the final interval for the UI approaches is 0.012, whilst the width of the final interval or the RBCL technique is 0.998. For $\rho_{C_{complete}}(1,3)$, the width of the final interval is 0.052 for the UI algorithms and as no interval can be calculated for the RBCL technique, a vacuous result (*i.e.*, an interval width of 2) is returned. The UI approaches are utilising more information from the correlation matrix and, even through they do not perfectly estimate the true correlation coefficient, some are producing relatively close estimates. The RBCL technique utilises the paired raw variables, which contains less information compared to the correlation matrix. Notably, RBCL cannot estimate the true correlation coefficient when there are no paired raw variables in the matrix.

Table 5.5: Final Intervals for Estimated Correlation $\rho_{C_{complete}}(1,2)$ and $\rho_{C_{complete}}(1,3)$

	$\boldsymbol{\rho_{C_{complete}}(1,2)}$				$\boldsymbol{\rho_{C_{complete}}(1,3)}$			
	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Distance	Percent	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Distance	Percent
$\boldsymbol{UI_{Untrans}}$	0.55	0.56	0.33	2.60	0.71	0.77	-0.12	16.67
$\boldsymbol{UI_{Abs}}$	0.90	0.91	**0**	98.70	0.92	0.97	0.03	66.67
$\boldsymbol{UI_{Sq}}$	0.85	0.86	0.04	12.99	0.90	0.95	0.01	61.11
$\boldsymbol{UI_{Rt}}$	0.92	0.93	0.02	18.18	0.95	0.98	0.06	27.78
RBCL	-0.068	0.93	**0**	100				

5.2.3 Generalising to the entire example matrix

In the previous section the two examples for a sparse correlated matrix were concerned with:

1. Estimating the true correlation coefficient for a known and unknown entry in the sparse correlation matrix;
2. Exploring the performance of each UI approach and;

3. Comparing the UI approaches with the RBCL technique.

When the UI approaches are applied to 80,810 MVs (64.9%) within the example matrix $C_{5\%}$, the number of pairwise comparisons, the percentage of the total interval in which the true correlation coefficient was contained within the interval and the absolute minimum distance from the true correlation to the UI approach are recorded. These are used to evaluate the performance of each UI approach. The average number of paired comparisons utilised to generate a pairwise correlations is 2.47, σ = 0.13. Therefore, the RBCL technique would produce a large interval in estimating these known pairwise correlations. The estimate probability density function (pdf) of these comparisons is displayed in Fig. 5.2.14.

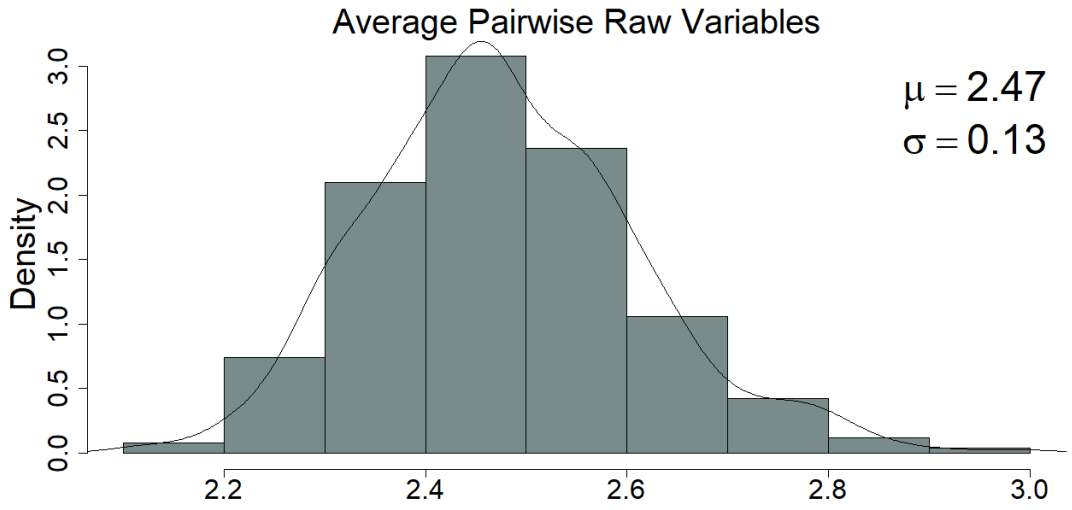

Figure 5.2.14: PDF of paired raw variable from sparse matrix $C_{5\%}$

These pairwise correlations calculated from the sparse raw data, $C_{5\%}$, are then utilised by the UI approaches. Even though these pairwise correlations are generated from few paired raw variables, the pairwise comparisons are generated from a higher number of paired pairwise correlation, see Fig. 5.2.15. The average number of these is 65.6, σ = 16.25. Hence, the average width of the final interval for the UI algorithms is 0.015.

Considering each UI algorithm separately, the average percentage of the $UI_{Untrans}$ intervals that captures the true correlation coefficient is 6.62% (σ = 9.13). Whilst the average distance of the final interval from the true correlation coefficient is .32 (σ = .12), see Fig. 5.2.16. In relation to UI_{Abs}, the average percentage of intervals that captures the true correlation coefficient is

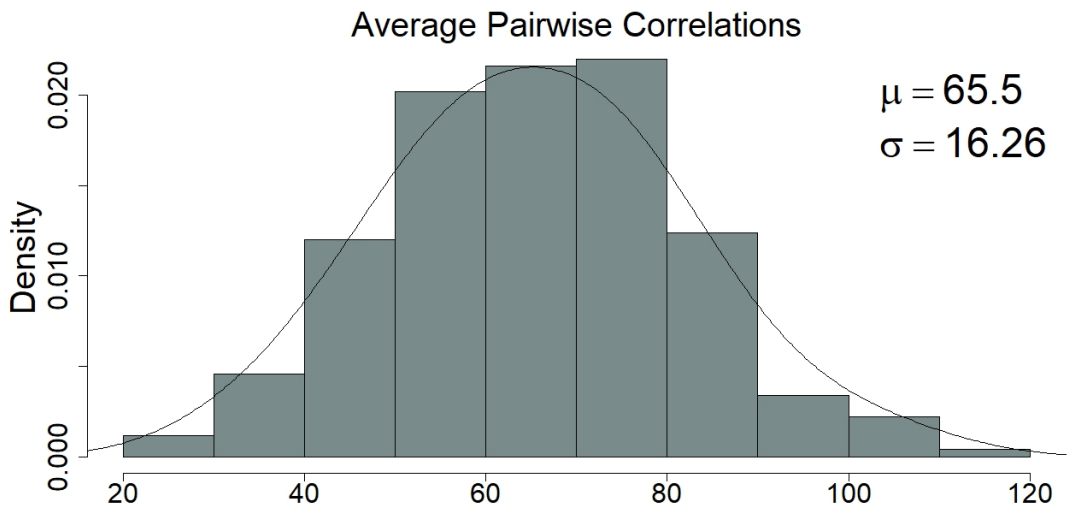

Figure 5.2.15: PDF of pairwise correlations in $Cor(C_{5\%})$

42.67% (σ = 24.43). The average absolute minimum distance of the final interval is .017 (σ = .015), see Fig. 5.2.17. The densities of the average percentage of intervals that captures the true correlation coefficient (μ = 28.29, σ = 26.44) and the average absolute minimum distance of the final interval (μ = .148, σ = .034) for UI_{Sq} are displayed in Fig. 5.2.18. Finally, UI_{Rt} has an average percentage of intervals that captures the true correlation coefficient of 24.97% (σ = 17.24) and the average absolute minimum distance of the final interval is .025 (σ = .017), see Fig. 5.2.19.

As expected, $UI_{Untrans}$ is the worse performing algorithm given that it includes negative correlations which affect the intervals by increasing their distance from the true correlation coefficient. UI_{Sq} performs marginally better, but the affect of squaring each correlation coefficient is decreasing the correlation coefficient that is entered into the algorithm. The is the result of taking a fraction of a fraction as correlations lie between [-1,1]. The UI_{Rt} algorithm attempts to reduce the affect of squaring and it performs better than just squaring alone; it is reducing the distance metric but not increasing the percentage metric. This leaves the UI_{Abs} algorithm, which not only reduces the distance metric but also increases the percentage metric, making it the best performing and most accurate of the four UI algorithms.

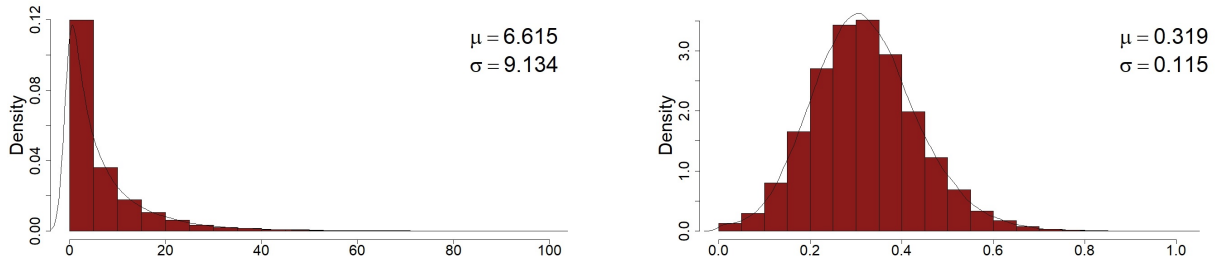

Figure 5.2.16: (a) PDF of percent metric and (b) PDF of distance metric of $UI_{Untrans}$

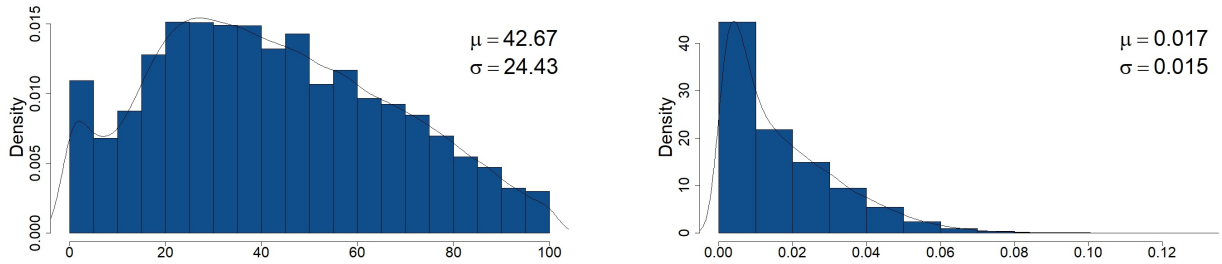

Figure 5.2.17: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Abs}

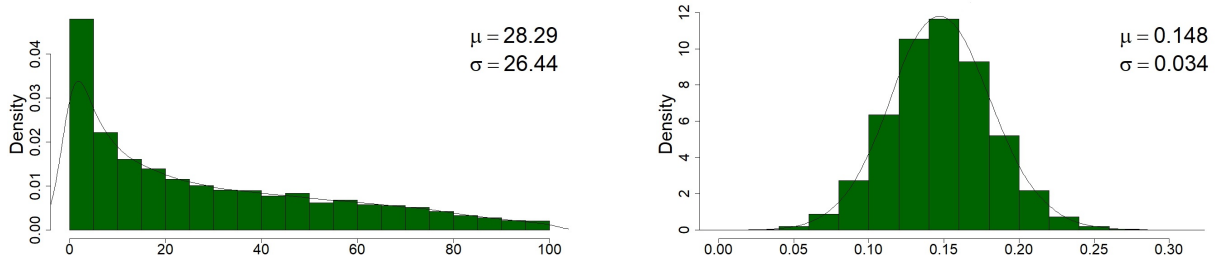

Figure 5.2.18: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Sq}

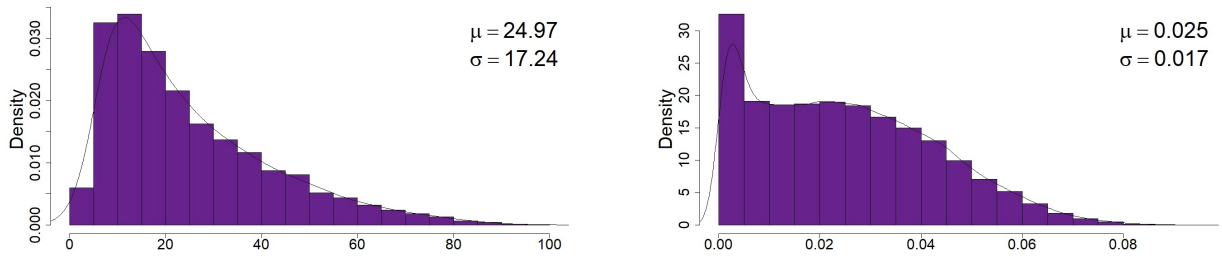

Figure 5.2.19: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Rt}

5.2.4 Comparison of UI_{Abs} to Singular Value Decomposition

In this section, the best preforming UI algorithm UI_{Abs} is compared to the Singular Value Decomposition (SVD). The SVD algorithm is perhaps one of the most famous recommendation algorithms, winning the 2006 Netflix prize for movie recommendation. This algorithm employs

the use of gradient descent to minimise the squared error between predicted rating and actual rating, eventually getting the best model. The purpose of such a comparison is to calculate the Root Mean Square Error (RMSE) that is traditionally used by RSs to evaluate the recommender model. The RMSE is the standard deviation of the prediction errors.

Firstly, to evaluate $C_{complete}$, the ratings matrix was split into a training (5% of ratings) and test set (95% of ratings) to demonstrate the performance of the SVD algorithm. This produced a RMSE of 0.894. This preformed in a similar way to $C_{5\%}$ which was the training data set and $C_{complete}$ was the test dataset. The correlation matrix produced from $C_{complete}$ was also split into train and test set, 5% and 95% respectively. This performed better than the previous two, resulting in a RMSE of 0.122. The correlation matrix produced by calculating pairwise correlations in $C_{5\%}$ was also used to train the SVD algorithm. It was evaluated using the correlation matrix produced from $C_{Complete}$ and had a RMSE of 0.1183. Lastly, the final interval of the UI_{Abs} algorithm was evaluated. As there is an upper and lower interval, the SVD algorithm was applied to both of them. The upper interval resulted in a RMSE of 0.1163, whilst the lower had a RMSE of 0.1162. This was evaluated on the complete correlation matrix. These values are displayed in Fig. 5.2.20.

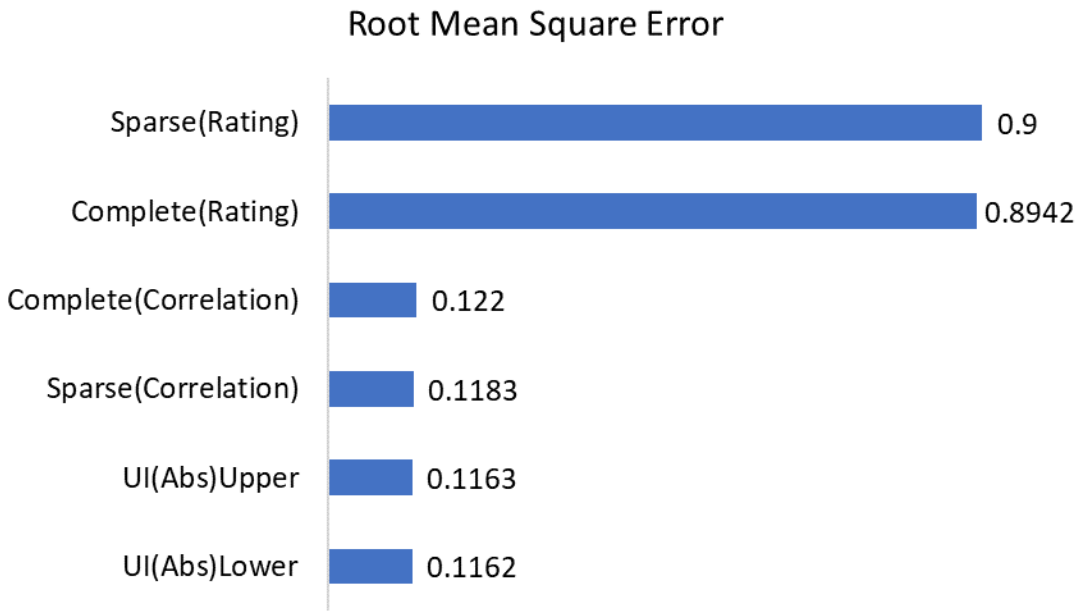

Figure 5.2.20: RSME of Complete, Sparse, Imputed and UI_{Abs}

5.2.5 Increasing Missingness

The procedure to produce the above example matrix was followed to randomly simulate 1,000 matrices with approximately 95%, 96%, 97%, 98% and 99% MVs producing $C_{5\%}$, $C_{4\%}$, $C_{3\%}$, $C_{2\%}$ and $C_{1\%}$ respectively. To evaluate the four UI approaches the average error was calculated for each individual simulation. The percent of the total number of intervals that the true correlation coefficient is contained inside the UI intervals is also recorded.

In Table 5.6 summary statistics of each set of 1,000 simulations by missingness are displayed. The $UI_{Untrans}$ algorithm, for all levels of missiningness, has the highest error in relation to average absolute minimum distance and the lowest average percentage of the UI algorithms. The UI_{Abs} algorithm consistently performs better than the $UI_{Untrans}$ and UI_{Sq} algorithms as expected. As the amount of missingness increases, the UI_{Rt} algorithm improves and is performing similarly to UI_{Abs}. It is evident that the higher the percentage of missingness, the higher the percentage of intervals that contain the true correlation coefficient. This is due to the reduced number of pairwise comparison, leading to wider intervals. In Fig. 5.2.21 the effect of the number of intervals built via the pairwise comparisons on the distance between the lower and upper bound of the final interval is displayed. From $C_{1\%}$ the average number of pairwise comparisons is 1, which results in a final interval width of 0.5. As the amount of missing decreases, the number of pairwise comparisons increases and at 95% missingness the final interval is calculated using an average of 50 comparisons giving a final width of 0.19. This demonstrates that the amount of vagueness and uncertainty within the prediction is decreasing with the amount of information provided.

For each set of the 1,000 simulations the minimum and maximum number of MVs in the correlation matrices are displayed in Table 5.7, these figures remain more or the same across the four UI approaches. The percentage of MVs that remain MVs for each level of missingness is also displayed. This value may be distorted due to the differing number of MVs in each of the simulated matrices as they range from approximately 61.6% to 66.9% MVs for 95% missingness. Wider intervals provide information to the user that more information is required and reflects the vagueness and uncertainty that exists due to a lack of information.

Table 5.6: Summary statistics of 1,000 simulations of $C_{5\%}$, $C_{4\%}$, $C_{3\%}$, $C_{2\%}$, $C_{1\%}$

	95 % Missing Values							
	$\boldsymbol{UI_{Untrans}}$		$\boldsymbol{UI_{Abs}}$		$\boldsymbol{UI_{Sq}}$		$\boldsymbol{UI_{Rt}}$	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	5.03	.29	37.34	.01	20.81	.03	19.56	.02
Max	7.90	.36	47.33	.02	33.43	.05	30.90	.04
Median	6.42	.32	41.90	.02	26.63	.04	25.19	.02
Mean	6.46	.32	**41.89**	**.02**	26.67	.04	25.14	.02
	96 % Missing Values							
	$\boldsymbol{UI_{Untrans}}$		$\boldsymbol{UI_{Abs}}$		$\boldsymbol{UI_{Sq}}$		$\boldsymbol{UI_{Rt}}$	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	16.94	.29	55.84	.02	16.08	.02	37.48	.02
Max	26.68	.36	68.25	.03	27.25	.03	55.52	.04
Median	21.62	.33	61.50	.02	21.94	.03	45.69	.03
Mean	31.64	.33	**61.46**	**.02**	21.96	.03	45.60	.03
	97 % Missing Values							
	$\boldsymbol{UI_{Untrans}}$		$\boldsymbol{UI_{Abs}}$		$\boldsymbol{UI_{Sq}}$		$\boldsymbol{UI_{Rt}}$	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	51.41	.28	90.41	.07	84.69	.07	78.80	.04
Max	61.95	.35	94.58	.09	90.41	.09	91.10	.06
Median	56.63	.31	92.61	.08	87.71	.08	87.25	.05
Mean	56.66	.31	**92.59**	**.08**	87.66	.08	**87.14**	**.05**
	98 % Missing Values							
	$\boldsymbol{UI_{Untrans}}$		$\boldsymbol{UI_{Abs}}$		$\boldsymbol{UI_{Sq}}$		$\boldsymbol{UI_{Rt}}$	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	68.14	.25	95.29	.01	93.22	.08	95.30	.02
Max	78.12	.33	98.91	.05	98.09	.12	98.62	.07
Median	72.75	.29	97.15	.02	95.53	.10	97.31	.04
Mean	72.73	.29	**97.16**	**.02**	95.53	.10	97.31	.04
	99 % Missing Values							
	$\boldsymbol{UI_{Untrans}}$		$\boldsymbol{UI_{Abs}}$		$\boldsymbol{UI_{Sq}}$		$\boldsymbol{UI_{Rt}}$	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	56.71	.27	91.39	0	20.81	.08	92.03	0
Max	92.86	.35	100	.01	33.43	.12	100	.01
Median	75.11	.32	99.23	0	26.63	.10	99.25	0
Mean	75.07	.32	**100**	**0**	26.67	.10	**100**	**0**

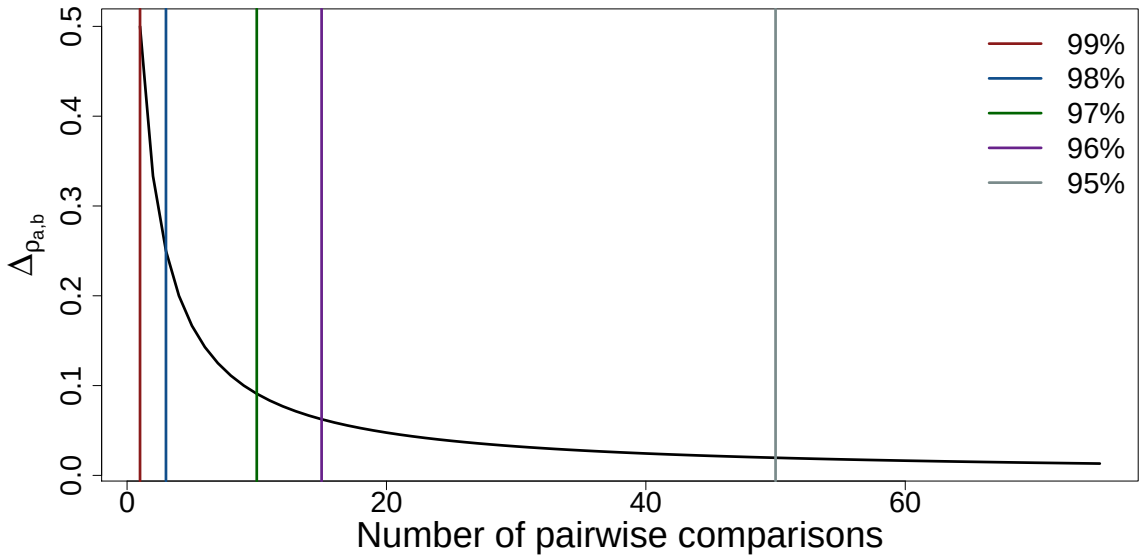

Figure 5.2.21: Distance between lower and upper bounds: $\Delta(\rho_{a,b}) = \frac{1}{n+1}$

It is evident that the UI approaches do not successfully complete a sparse correlation matrix when the level of missingness increases from 95%. They do provide transparency, in that, the width of the final bounds reveal the amount of uncertainty and vagueness in the estimate of the correlation coefficient due to a smaller amount of information. The RBCL technique also provide the user with this feedback via the width of the final interval. However, it is incapable of estimating bounds for correlation coefficients when there are not paired raw variables in the sparse matrix, which is a high percentage at the 95% missingness level, increasing with the level of missingness.

The UI approaches perform discrete tasks, *i.e.*, it is possible to estimate each correlation coefficient independent or another. Hence, this approach can be performed on parallel processes. In addition, the time complexity appears not to be an issue given that 80,000 MVs (without being parallelised or optimised) can be calculated in 49.87 seconds. In Fig. 5.2.22 the length of time taken to calculate various numbers of MVs are displayed for both user and time elapsed for system time.

5.2.6 Distributions

In this section, the application of all four UI approaches is explored on various distributions and then evaluated as before. Three distributions will be generated: $Poisson(\lambda = 5)$ $(\mu_P = 5, \sigma_P^2 = 5)$, $Beta(\alpha = 0.25, \beta = 0.25)$ $(\mu_B = 5, \sigma_B^2 = 1.6)$ and $Skewed\ Normal(\epsilon(\text{location}) = 5, \omega(\text{scale}) =$

Table 5.7: The minimum and maximum number of MV in simulations of $C_{5\%}$, $C_{4\%}$, $C_{3\%}$, $C_{2\%}$, $C_{1\%}$

	95 % Missing Values			
	$\boldsymbol{UI_{Untans}}$	$\boldsymbol{UI_{Abs}}$	$\boldsymbol{UI_{Sq}}$	$\boldsymbol{UI_{Rt}}$
Min(%)	76835 (61.6)	76826 (61.6)	77329 (61.9)	77208 (61.9)
Max(%)	83456 (66.8)	83678 (67.1)	83334 (66.8)	83438 (66.9)
% still MV	**0**	**0**	**0**	**0**
	96 % Missing Values			
	$\boldsymbol{UI_{Untans}}$	$\boldsymbol{UI_{Abs}}$	$\boldsymbol{UI_{Sq}}$	$\boldsymbol{UI_{Rt}}$
Min(%)	98485 (78.9)	98480 (78.9)	98481 (78.9)	98481 (78.9)
Max(%)	102939 (82.5)	103345 (82.8)	103346 (82.8)	103346 (82.8)
% still MV	**.08**	**.08**	**.08**	**.08**
	97 % Missing Values			
	$\boldsymbol{UI_{Untans}}$	$\boldsymbol{UI_{Abs}}$	$\boldsymbol{UI_{Sq}}$	$\boldsymbol{UI_{Rt}}$
Min(%)	114004 (91.3)	113904 (91.3)	113904 (91.3)	113904 (91.3)
Max(%)	116553 (93.4)	116697 (93.5)	116697 (93.5)	116697 (93.5)
% still MV	**13.49**	**13.45**	**13.45**	**13.45**
	98 % Missing Values			
	$\boldsymbol{UI_{Untans}}$	$\boldsymbol{UI_{Abs}}$	$\boldsymbol{UI_{Sq}}$	$\boldsymbol{UI_{Rt}}$
Min(%)	122179 (97.9)	122155 (98.1)	122122 (97.8)	122122 (97.8)
Max(%)	122918 (98.5)	122907 (98.5)	122944 (98.5)	122944 (98.5)
% still MV	**83.87**	**83.83**	**83.84**	**83.84**
	99 % Missing Values			
	$\boldsymbol{UI_{Untans}}$	$\boldsymbol{UI_{Abs}}$	$\boldsymbol{UI_{Sq}}$	$\boldsymbol{UI_{Rt}}$
Min(%)	124543 (99.8)	124542 (99.8)	124542 (99.8)	124542 (99.8)
Max(%)	124651 (99.9)	124647 (99.9)	124650 (99.9)	12450 (99.9)
% still MV	**99.87**	**99.87**	**99.87**	**99.87**

$2, \alpha(\text{shape}) = 5)$ $(\mu_{SN} = 5.04, \sigma^2_{SN} = 1.61)$. The parameters for these distributions are chosen as to have similar mean and, in the case of the *Beta* and *Skewed Normal* distributions, a similar variance.

As before, each matrix contains 500×500 entries and were simulated 1,000 times for each distribution and each UI algorithm (12,000 simulations in total). The percentage of missingness

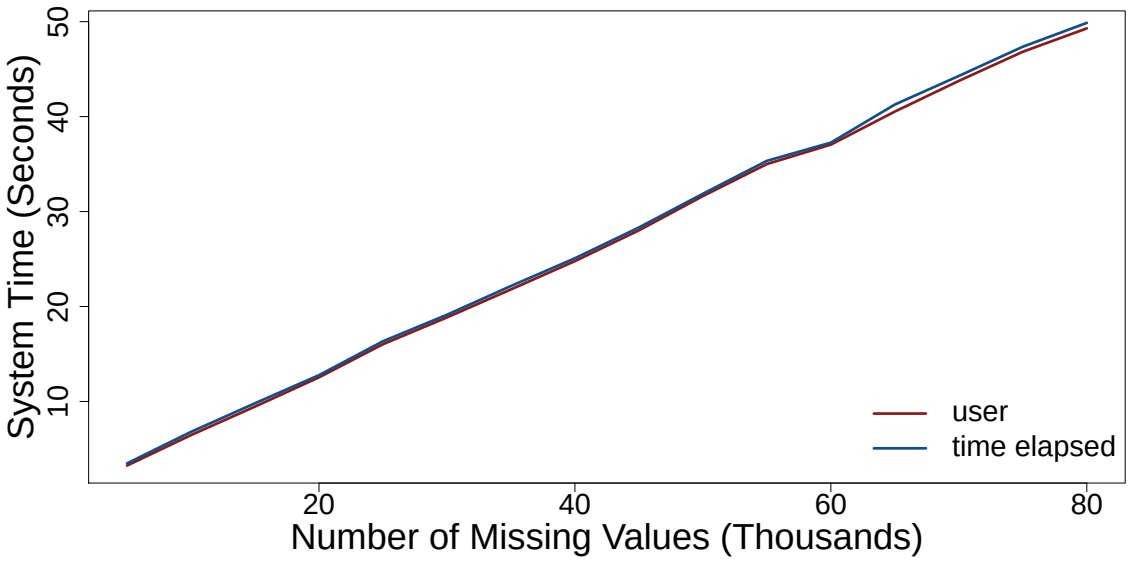

Figure 5.2.22: System time of user and time elapsed for MVs calculated by UI_{Abs}

for each of these matrices is set at 95% and then a correlation matrix is produced via pairwise correlations. This produces correlation matrices with approximately 64% MVs still present, which is the same for the raw variables generated as Multivariate Normal (MVN) in Section 5.2. In Table 5.8 the summary of these simulations is presented, along with those for the MVN distribution with 95% missingness. The minimum of the absolute minimum distance increases no more than 0.01 for each distribution when compared with MVN, with the exception of UI_{Un} which has a minimum of the absolute minimum distance increase of .03. Given that this algorithm has the largest absolute minimum distance, an increase of .03 appears reasonable. The one distribution and algorithm that appears to vary from the MVN is the $UI_{Untrans}$ algorithm applied to *Poisson* distribution. This differs in both absolute minimum distance and percentage, being more accurate and a higher percentage of intervals in which the true correlation coefficient is contained. The $UI_{Untrans}$ was applied to another 1,000 simulations; similar results were observed (Percent: min =11.98, max = 18.12, mean = 15.53, median = 15.49 whilst for Distance: min = 0.14, max = 0.19, mean = 0.17, median = 0.17). Even though the $UI_{Untrans}$ performs superior when applied to a *Poisson* distribution, it does not outperform the UI_{Abs} algorithm. Finally, the mean percentage of intervals that the true correlation coefficient is contained in varies between approximately 27% and 29% for UI_{Sq} and between approximately 19% and 29% for UI_{Rt}. For UI_{Abs}, the mean percentage of intervals that the true correlation coefficient is contained within varies between 35% and 44%, depending on the distribution; the

Table 5.8: Summary statistics of 1,000 simulations of various distributions

MVN	$UI_{Untrans}$		UI_{Abs}		UI_{Sq}		UI_{Rt}	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	5.03	.29	**37.34**	**.01**	20.81	.03	19.56	.02
Max	7.90	.36	**47.33**	**.02**	33.43	.05	30.90	.04
Median	6.42	.32	**41.90**	**.02**	26.63	.04	25.19	.02
Mean	6.46	.32	**41.89**	**.02**	26.67	.04	25.14	.02
Poisson	$UI_{Untrans}$		UI_{Abs}		UI_{Sq}		UI_{Rt}	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	12.12	.14	**40.19**	**.02**	20.33	.04	24.36	.02
Max	19.38	.20	**50.52**	**.03**	31.97	.07	34.43	.04
Median	15.60	.17	**44.44**	**.02**	26.84	.05	29.44	.03
Mean	15.58	.17	**44.45**	**.02**	26.84	.05	29.49	.03
Beta	$UI_{Untrans}$		UI_{Abs}		UI_{Sq}		UI_{Rt}	
	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	4.92	.26	**21.64**	**.02**	25.14	.03	14.65	.03
Max	8.27	.32	**29.43**	**.03**	32.83	.04	24.82	.05
Median	6.50	.29	**29.46**	**.02**	29.19	.03	19.52	.04
Mean	6.52	.29	**35.61**	**.02**	29.19	.03	19.56	.04
Skewed	$UI_{Untrans}$		UI_{Abs}		UI_{Sq}		UI_{Rt}	
Normal	%	*Distance*	%	*Distance*	%	*Distance*	%	*Distance*
Min	4.98	.29	**35.11**	**.01**	20.61	.03	17.79	.02
Max	8.58	.36	**45.87**	**.03**	34.30	.05	30.16	.04
Median	6.41	.32	**40.98**	**.02**	27.24	.04	23.53	.03
Mean	6.45	.32	**41.01**	**.02**	27.14	.04	23.51	.03

mean absolute minimum distance changes by a maximum of 0.01. Therefore, given that the UI_{Abs} estimate is the most accurate of all the UI algorithms, and that the absolute minimum distance from the true correlation coefficient is more important than the percentage metric, the UI_{Abs} algorithm performs in a very similar manner on these datasets.

5.3 Chapter Summary

This chapter demonstrated the four UI techniques on sparse data sets with various amounts of missingness. It is clear that the UI_{Abs} algorithm is the best performing algorithm with respect

to the number of intervals that contained the true correlation coefficient and the absolute minimum distance for it. This chapter has presented some of the limitations of the RBCL technique when applied to sparse datasets; when RBCL can generate an interval, the interval is much wider than the interval calculated under the UI approaches. It has also highlighted the limitations of the $UI_{Untrans}$ and UI_{Sq} algorithms; these are unable to mitigate the affects of negative pairwise correlations and *spread* due to squaring respectively.

The UI_{Abs} algorithm performed well for each level of missingness. As demonstrated, this algorithm is capable of reducing the affect of negative pairwise correlations by only inputting absolute values. Given that the pairwise correlation matrix generated from the sparse matrix $C_{5\%}$ is assumed to be exchangeable and the raw variables are assumed to be positively correlated, this approach is justified. It is worth noting that the UI_{Rt} approach becomes more comparable to UI_{Abs} when the level of missingness is increased.

The various distributions generated in this section provide information on the performance of the UI algorithms. It was found that the UI_{Abs} algorithm performed best in relation to the absolute minimum distance from the true correlation coefficient and the percentage of intervals that contained this correlation coefficient. In the next section, the UI_{Abs} approach is applied to genres of movies taken from the MovieLens dataset, providing a real-world case study of its performance.

6

Case Study with MovieLens Data

In this chapter the selected algorithm UI_{Abs} is applied to the MovieLens data. This data is one of the most utilised datasets in evaluating recommender systems. As the UI_{Abs} algorithm assumes partial exchangeability, the algorithm is applied to subsets of the MovieLens data based on the assigned genre of *Action*, *Children* and *Horror* movies. As in previous sections, the UI_{Abs} algorithm is evaluated in relation to the percentage of intervals that capture the true correlation coefficient (in this case the calculated pairwise coefficient) and the absolute minimum distance the final intervals lies from the true correlation coefficient.

6.1 MovieLens Data

The MovieLens data is often employed in exploring new techniques in RSs and is referred to as a stable benchmark dataset (Harper & Konstan, 2016). During Spring in 2015, a search for "movielens" produced 2,750 results in Google Books and 7,580 in Google Scholar (Harper & Konstan, 2016). The MovieLens 100K data consists of 100,000 ratings from 943 users on 1682 movies, being released in April 1998. It is freely available[1] and was collected by the GroupLens Research Project. Each rating is a discrete value between 1 and 5. There is no explicit definition as to what these ratings represent, but one may assume that: 1=*I hated it*; 2=*I didn't it*; 3=*It was OK*; 4=*I liked it* and; 5=*I loved it*. This data has been cleaned, *i.e.*, users who had less than 20 ratings or did not have complete demographic information were removed from this data set. The data files consist of *u.data*; this is the full ratings data set, consisting of ratings by

[1]https://grouplens.org/datasets/movielens/100k/ Downloaded 22 November 2016.

users on movies. A list of genres that movies may belong to is given in *u.genre*; a movie may belong to more than one genre. Demographic information about the users, *i.e.*, user ID, age, gender, occupation and zip code, are located in *u.user*. This information is not utilised by the UI approaches.

There is approximately 95% of the MovieLens data with no entries. The individual's rating behaviours are highly associated with missing patterns; movies with higher average rating scores attract more ratings from individuals, whist individuals who frequently rate movies tends to be more critical and give lower ratings. Therefore, the pattern of missingness is not Missing Completely At Random (MCAR) and as such may produce a higher percentage of negative or zero pairwise correlations than a ratings matrix that is MCAR.

When the Singular Value Decomposition (SVD) algorithm, as introduced in Section 5.2.4 above, is applied to the MovieLens data, the average Root Mean Square Error (RMSE) produced is 0.934 (80% training set). A pairwise correlation was applied to the entire MovieLens data and this created a matrix which contained no missing values. As a result, the UI_{Abs} algorithm could not be applied. When the SVD algorithm was applied to only the correlation matrix produced by the pairwise correlation of the MovieLens matrix, again with 80 training set, this resulted in a RSME of 0.938.

6.1.1 Specific Examples by Genre

In order to create a matrix of exchangeable correlation coefficients, the MovieLens data was separated by genre. Genres refers to recurring, repeating and similar, familiar or instantly-recognizable patterns, styles, themes, syntax, templates, paradigms, motifs or rules. Themes, even though normally incorporated into genres, are distinct and refers to the issues or concepts that the movie revolves around (*e.g.*, science fiction, sports, or crime). The mood of a movie is its emotional tone (*e.g.*, comedy, horror, or tearjerker) and may also be considered distinct from genre.

The first genre explored in this section is *Action*, with a total of 251 movies in this category. The Internet Movie Database (IMDb) is an online database of information related to films, television programs and video games. The ratings are generated by registered individuals who contribute info to the site. The movie names and IMDb ratings are referred to so that the reader

may gain some context regarding the movies. The correlation between Movie 2 (Goldeneye (1995), IMDb(7.2)) and Movie 22 (Braveheart (1995), IMDb(8.4)) is to be estimated. In this sub-matrix, there are approximately 89% missing data. This correlation was arbitrarily chosen and it is assumed that it represents the true correlation coefficient. The minimum rating assigned to a movie in this genre is 1, with a maximum rating of 5. The average rating in this genre is 3.48 (median=4).

Table 6.1: First ten correlations of *Action* movies

Movie	2	4	17	21	22	24	27	28	29	33
2	1.00									
4	0.24	1.00								
17	-0.17	0.06	1.00							
21	0.36	0.22	-0.27	1.00						
22	0.10	0.10	-0.02	0.23	1.00					
24	0.25	0.06	0.39	0.14	0.23	1.00				
27	0.30	0.27	-0.04	0.09	-0.04	0.13	1.00			
28	0.26	0.01	-0.15	0.20	0.43	-0.13	0.11	1.00		
29	0.25	0.07	0.16	0.46	0.17	-0.05	0.14	0.09	1.00	
33	0.20	0.14	0.44	-0.13	0.04	0.35	0.34	0.04	0.28	1.00

As before, the pairwise correlations of the *Action* movies are calculated with Table 6.1 displaying the first ten correlations of this sub-dataset. For the pairwise correlation matrix, there are approximately 31% correlations that can not be calculated due to no available pairwise comparisons. There are only 8% of entries with a pairwise correlation coefficient of 0.7 or above. The minimum pairwise correlation coefficient is -1, the maximum is 1, with a mean of 0.20 (median=0.22). Therefore, the RBCL technique cannot be applied to estimate an interval for these correlation coefficients. The use of pairwise comparisons has limitations, but the percentage of correlation coefficients below 0.7 is high and so the movies in the *Action* sub-matrix are explored further. For example, $\rho_{Movie}(2, 17) = -0.17$; Movie 17 (From Dust Till Dawn (1996), IMDb(7.3)) is classed as an *Action/Comedy/Crime/Horror/Thriller* movie by MovieLens and is described as a modest success at the box office which has since become a cult film. Therefore, this negative correlation may be a true reflection of the true correlation coefficient and not a result of the limitations of pairwise correlation. If the former is true, then the correlation matrix from *Action* is not truly exchangeable. Many movies currently do not fit into one genre classification. Many movies are considered hybrids, *i.e.*, they straddle several

film genres. It is possible to further categorise movies into sub-matrices of multiple genres, *Action* and *Horror*, but this sub-matrix contains very few entries and, either there are two few pairwise correlations produced to apply the UI algorithm, or there is no need to apply the UI algorithm as there are no MV found in the pairwise correlation matrix.

The correlation coefficient $\rho_{Movie}(2, 22)$ is set to MV and both the UI_{Abs} and RBCL techniques were applied. In Fig. 6.1.1 the performance of both approaches are graphically displayed. The correlation coefficient ρ_{Movie} for RBCL was calculated using ninety-nine pairwise comparisons with ν_0 = 2. We can clearly see that the final interval produced $(0.074, 0.114)$ captures the pairwise correlation coefficient $\rho_{movie}(2, 22)$. This has a final interval width of 0.04. UI_{Abs} utilises 204 pairwise correlation products, and even though it lies outside the true correlation coefficient a higher percentage of the time, it gives an excellent estimate of $\rho_{movie}(2, 22)$ and a narrower final interval than the RBCL, see Table 6.2.

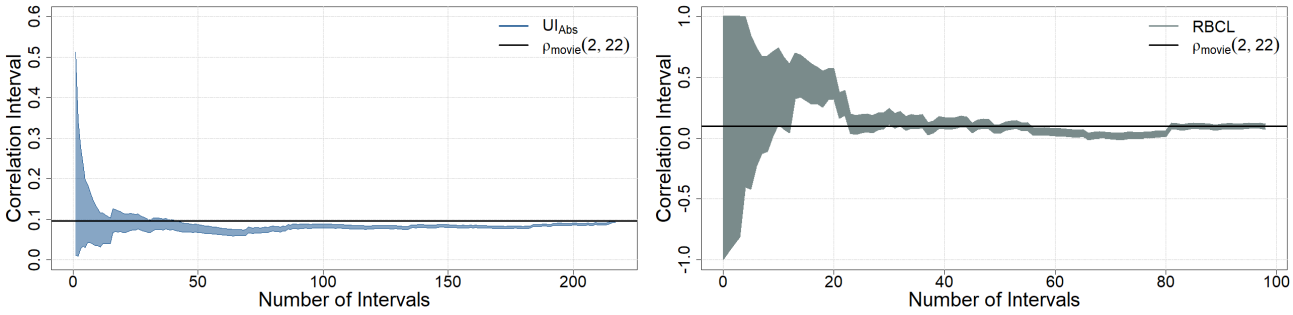

Figure 6.1.1: Upper and lower bounds for $\rho_{Movie}(2, 22)$ via (a) UI_{Abs} and (b) RBCL

Table 6.2: Final Intervals for Estimated Correlation $\rho_{Movie}(2, 22)$

	$\boldsymbol{UI_{Abs}}$				**RBCL**			
	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Dis	%	$\underline{\rho}_{n}$	$\overline{\rho}_{n}$	Dis	%
$\boldsymbol{\rho_{C_{movie}}}(2, 22)$	0.09	0.10	**0**	3.60	0.07	0.11	**0**	61.61

The next genre that is considered is *Children's* movies. This sub-rating matrix consisted of 122 movies. The level of missingnesss for this genre is 94%, reducing to 32% when pairwise correlations are calculated. The average rating in this sub-matrix is 3.35 (minimum=1, maximum=5, median=3). In relation to the pairwise correlation matrix, the mean pairwise correlation coefficient is 0.28 (minimum=-1, maximum=1, median=0.32). The first ten pairwise correlation coefficients are displayed in Table 6.3.

The pairwise correlation between movie 1 (Toy Story (1995), IMDb(8.3)) and movie 71 (The

Table 6.3: First ten correlations of *Children* movies

Movie	1	8	35	63	71	78	91	94	95	99
1	1.00									
8	0.25	1.00								
35	0.45	0.46	1.00							
63	0.38	0.20	0.76	1.00						
71	0.43	0.14	0.47	0.31	1.00					
78	0.24	0.39	0.93	0.40	0.48	1.00				
91	0.14	0.18	-0.62	-0.04	-0.07	-0.11	1.00			
94	0.27	0.13	0.61	0.25	0.04	0.38	-0.03	1.00		
95	0.41	0.22	0.09	0.38	0.14	0.51	0.35	-0.03	1.00	
99	0.32	0.20	0.52	0.235	0.26	0.28	0.23	0.20	0.23	1.00

Lion King (1994), IMDb(8.5)) was set to MV. These movies have a pairwise correlation of 0.43, which indicates a strong positive correlation. The UI_{Abs} algorithm has failed to capture this pairwise correlation in its final interval, having a minimum distance of 0.28. This indicates that UI_{Abs} underestimated the pairwise correlation. Unlike in the previous chapter, where 86% of the entries in the pairwise correlation matrix were above 0.7, only 20% of the *Children's* matrix is 0.7 or greater. Given that a high proportion of the matrix is not showing high positive correlations, it is likely there is a violation of the assumption of exchangeability.

In Fig. 6.1.2 it is evident that the RBCL technique performance is superior to that of UI_{Abs}, with Table 6.4 displaying the evaluation metrics. Based on the ratings from IMDb and their genre, these movies should be highly correlated. The fact that these movies are not positively correlated to other movies in this genre is affecting the UI_{Abs} intervals, thus the final interval does not reflect the pairwise correlation coefficient. If the assumption of exchangeability was not violated, and the pattern of missingness was MCAR, then it would be anticipated that the UI_{Abs} algorithm would produce an interval estimate of the true correlation coefficient that is higher than the pairwise correlation coefficient. It is worth noting that the number of intervals in the UI_{Abs} algorithm (n = 87) is less than the number of intervals in the RBCL technique (n = 173). Clearly there are more pairwise comparisons in the ratings matrix than in the correlation matrix. The reason being, even though there are 122 movies classed as *Children's*, there are 943 users who rated these movies, 173 of which rated both Movie 1 and Movie 71.

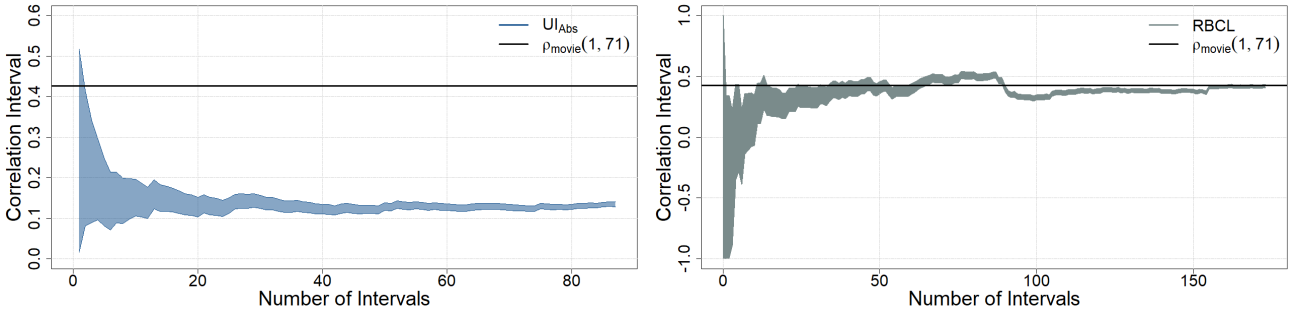

Figure 6.1.2: Upper and lower bounds for $\rho_{Movie}(1,71)$ via (a) UI_{Abs} and (b) RBCL

Table 6.4: Final Intervals for Estimated Correlation $\rho_{Movie}(1,77)$

	$\boldsymbol{UI_{Abs}}$				**RBCL**			
	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Dis	%	$\underline{\rho}_{n}$	$\overline{\rho}_{n}$	Dis	%
$\boldsymbol{\rho_{C_{movie}}}(2,22)$	0.13	0.14	0.28	1.15	0.5	0.114	**0**	12.56

Exploring the movies in this sub-matrix, $\rho_{Movie}(35, 91) = -0.62$. Movie 63 (The Santa Clause (1994), IMDb(6.4)) is classified as *Children's/Comedy* by MovieLens and as *Comedy/Drama/Family* by IMDb. Movie 93 (The Nightmare Before Christmas (1993), IMDb(8.0)) is classified as *Animation/Children's/Musical* by MovieLens and as *Animation/Family/Fantasy* by IMDb. Neither movie is classed as *Chridren's* by IMDb and they have a strong negative pairwise correlation. Hence, these movies may not be classed in the correct genre in the MovieLens data. This leads the pairwise correlations produced for this sub-matrix to violate the assumption of exchangeability.

The final genre to be considered is *Horror*. This consists of a total of 92 movies with approximately 94% MVs in the ratings matrix. The minimum rating given for a *Horror* movie is 1, with a maximum rating of 5. The mean rating in this sub-ratings matrix is 3.29 (median =4). The percentage of misssingness is reduced to 37% when the pairwise correlations were calculated. The minimum pairwise correlation coefficient calculated is -1, the maximum is 1, with a mean of 0.27 (median=.030); with approximately 14% of entries with a pairwise correlation coefficient 0.7 or higher. In Table 6.5 the first ten correlations are displayed.

Table 6.5: First ten correlations of *Horror* movies

Movie	17	84	101	123	183	184	185	200	201	208
17	1.00									
84	0.03	1.00								
101	0.16	-0.08	1.00							
123	0.30	0.76	0.35	1.00						
183	0.12	0.03	0.36	0.36	1.00					
184	0.30	0.40	0.26	0.18	0.23	1.00				
185	-0.03	-0.53	0.06	0.08	0.25	0.06	1.00			
200	0.06	-0.21	0.42	-0.08	0.33	0.09	0.22	1.00		
201	0.48	-0.54	0.09	0.14	0.22	0.64	0.21	0.44	1.00	
208	0.04	0.54	0.11	0.25	0.07	0.18	0.24	0.17	0.03	1.00

Movie 17 (From Dusk Till Dawn (1996), IMDb (7.3)) and Movie 183 (Alien (1979), IMDb (8.5) are examined. Both the UI_{Abs} and RBCL algorithms deliver similar results as the *Action* genre, see Fig. 6.1.3 and Table 6.6.

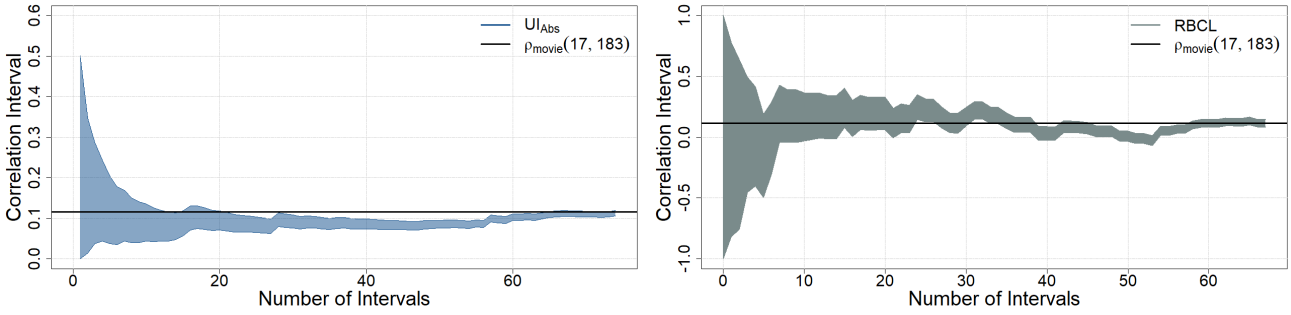

Figure 6.1.3: Upper and lower bounds for $\rho_{Movie}(17, 183)$ via (a) UI_{Abs} and (b) RBCL

Table 6.6: Final Intervals for Estimated Correlation $\rho_{Movie}(17, 183)$

	$\boldsymbol{UI_{Abs}}$				**RBCL**			
	$\underline{\rho}_{a,b}$	$\overline{\rho}_{a,b}$	Dis	%	$\underline{\rho}_{n}$	$\overline{\rho}_{n}$	Dis	%
$\boldsymbol{\rho_{C_{movie}}}(17, 183)$	0.10	0.12	**0**	36.49	.08	0.14	**0**	70.59

In Fig. 6.1.4 the frequency of ratings (1-5) for each of the genres explored above is displayed. Clearly the majority of users assigned a rating of 4 to the genres of *Action* and *Horror*, with *Children's* movies receiving a similar number of 3 (n = 2218) and 4 (n = 2212) ratings.

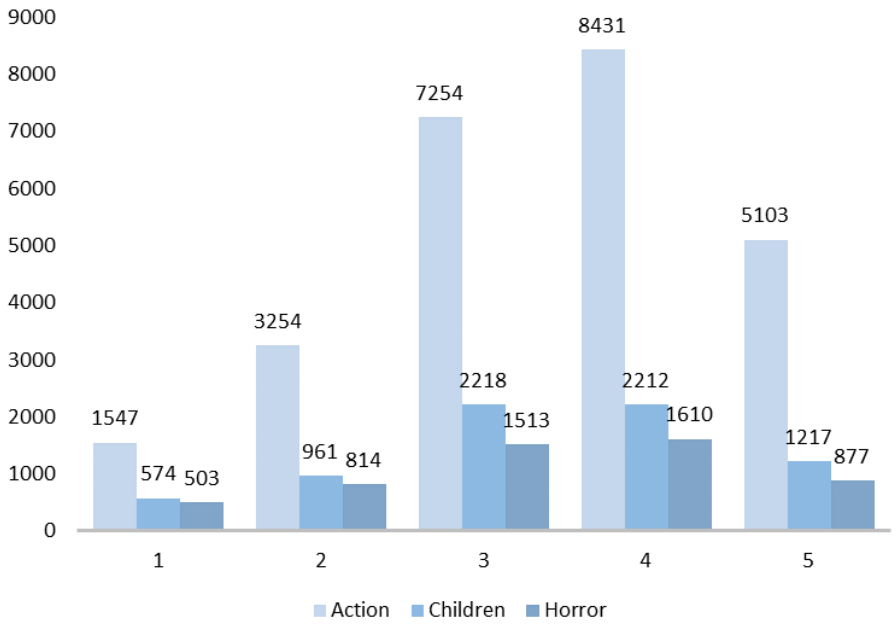

Figure 6.1.4: Frequency of movie ratings by genre

6.1.2 Evaluating Complete Sub-matrix by Genre

The specific examples in the previous section were used to not only evaluate the UI_{Abs} approach, but also to compare it with the RBCL technique. In this section, as in Section 5.2.1, each entry of the sub-ratings matrix for *Action*, *Children's* and *Horror* are separately assigned as MV. The percent and distance metric is calculated for each entry in order to evaluate the performance of the UI_{Abs} algorithm. Firstly, for the genre *Action*, the estimated pdf for both metrics is displayed in Fig. 6.1.5. The UI_{Abs} estimates the correlation coefficient with a distance of .1 or less in 33% of cases. Clearly the percentage metric indicates that the UI_{Abs} algorithm is not capturing the pairwise correlation coefficient in a high percentage of intervals. However, as shown with the specific examples above, the interval can lie very close to the pairwise correlation, but not actually contain it. There are very similar findings for *Children's* (Fig. 6.1.6) and *Horror* (Fig. 6.1.7), with 26.7% and 27.3% of the final interval having an absolute minimum distance of 0.1 or less respectively.

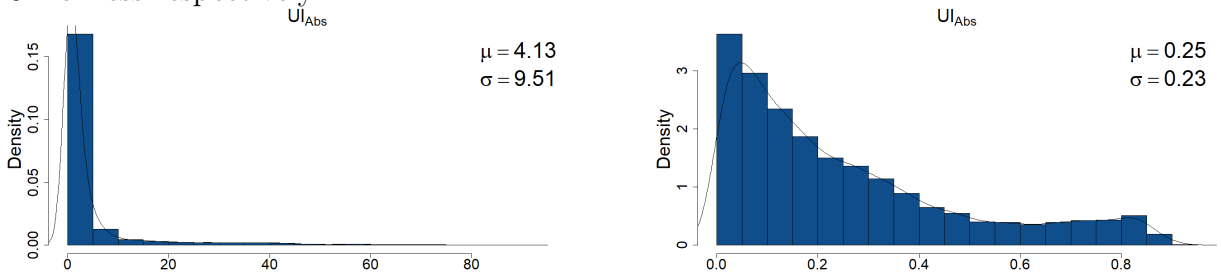

Figure 6.1.5: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Abs} for *Action*

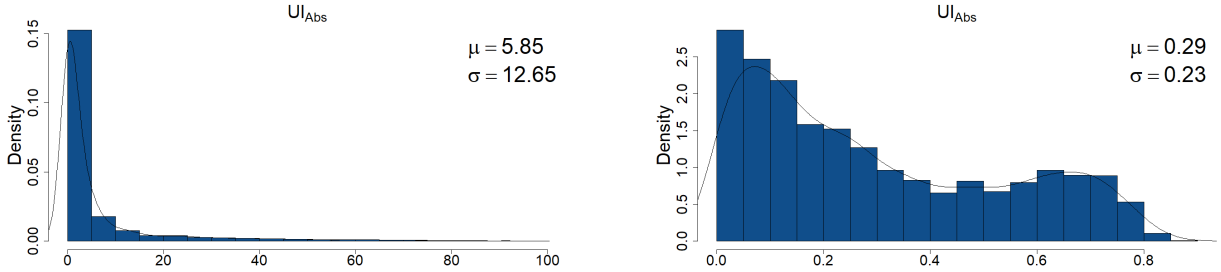

Figure 6.1.6: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Abs} for *Children*

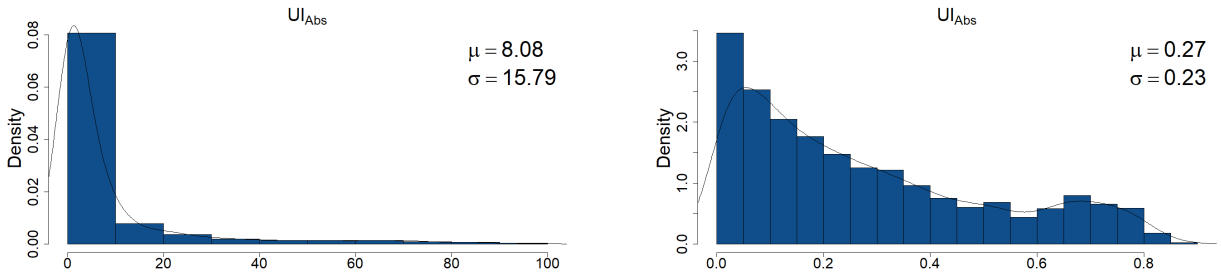

Figure 6.1.7: (a) PDF of percent metric and (b) PDF of distance metric of UI_{Abs} for *Horror*

As the UI_{Abs} approach is designed to calculate MVs via pairwise correlations, this next section will present the findings from calculating the true MVs for each sub-matrix via UI_{Abs}, *i.e.*, not replacing a pairwise correlation with a MV. In Fig. 6.1.8 the percentage of each rating (values between 1 and 5) are displayed for each *Action* movie, of which there are 251 such movies. Each movie has a unique bar, with each bar displaying this percentage. Note, some movies have 100% of a particular rating, *e.g.*,the last last movie displayed has a 100% rating of 2. There is two reasons why this may occur: 1) only one person rated this movie or; 2) all users rated this movie the same.

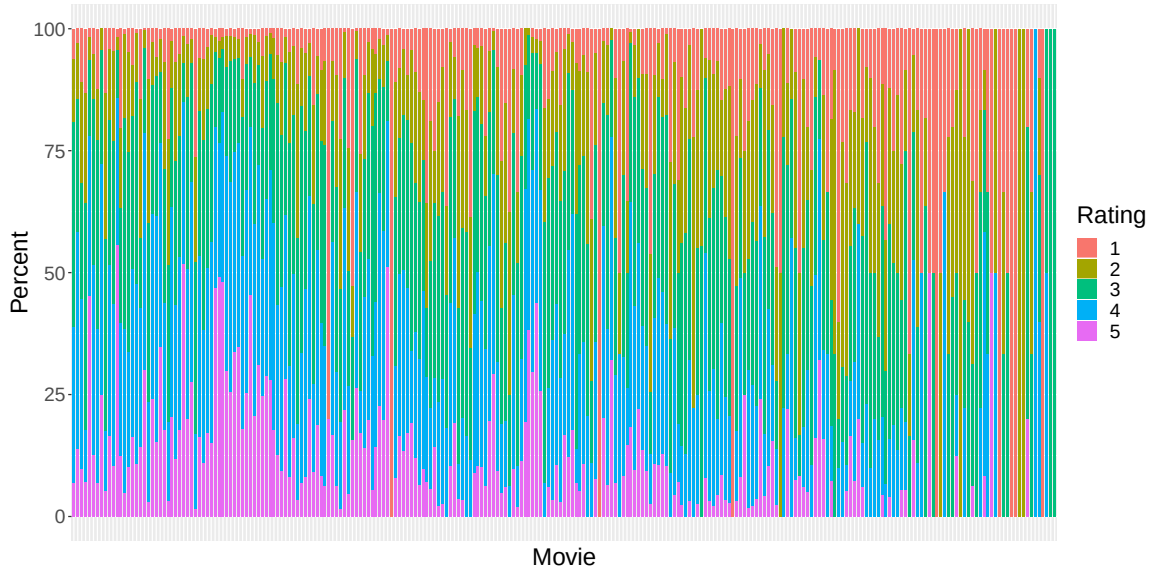

Figure 6.1.8: Percentage of ratings (1-5) for each *Action* movies

The estimated pdf of the final lower and upper bounds of the UI_{Abs} intervals are displayed in Fig. 6.1.9. The number of pairwise comparisons ranges from 1 to 225 with an average number of 64 pairwise comparisons, see Table 6.7. The mean value of the final UI_{Abs} intervals that estimate the true correlation coefficient is 0.24, recall the mean pairwise correlation for the

sub-matrix is 0.20. The UI_{Abs} algorithm is unable to calculate 36% of the MVs ($n = 3,452$) due to the lack of pairwise comparisons in the pairwise correlation matrix

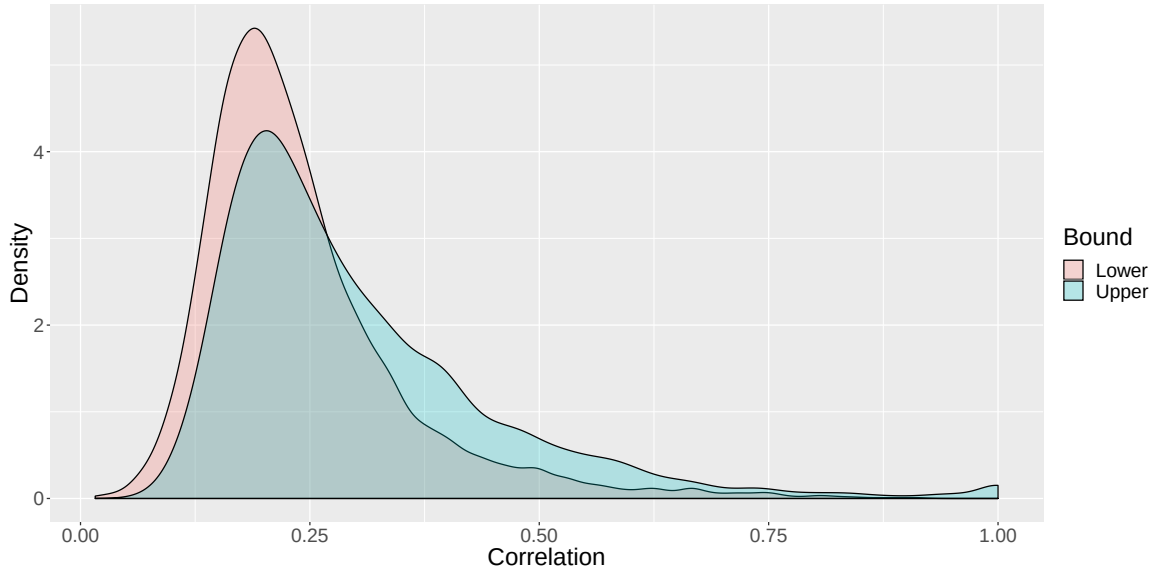

Figure 6.1.9: PDF of Upper and Lower bounds of UI_{Abs} for *Action*

Table 6.7: Summary statistics for *Action* genre

	Minimum	**Maximum**	**Mean**	**Median**
$\underline{\boldsymbol{\rho}}_{\boldsymbol{movie}}\boldsymbol{(Action)}$	.039	.789	.120	.113
$\overline{\boldsymbol{\rho}}_{\boldsymbol{movie}}\boldsymbol{(Action)}$	.044	.804	.127	.119
# Comparisons	1	225	181	191

The same trend in the *Action* pairwise correlation matrix can be seen in the *Children's* pairwise correlation matrix. There are 122 movies in this matrix, with the majority of movies receiving a value of each rating, see Fig. 6.1.10. Once more, the majority of the pairwise correlations fell below 0.7 and the estimated pdfs of the lower and upper bounds are displayed in Fig. 6.1.11. Recall, the mean pairwise correlation for this genre is 0.28, for the lower bound it is 0.32 and the upper bound it is 0.38, see Table 6.8. In this instance, the UI_{Abs} algorithm is able to calculate 88% ($n = 2032$) of the total number of MVs.

As there are fewer movies in the *Horror* genre, there is less of an overlap of the estimated pdfs for the lower and upper bounds, see Fig. 6.1.13. The average number of pairwise comparison is 24 as displayed in Table 6.9, which also displays the mean of the final lower and upper

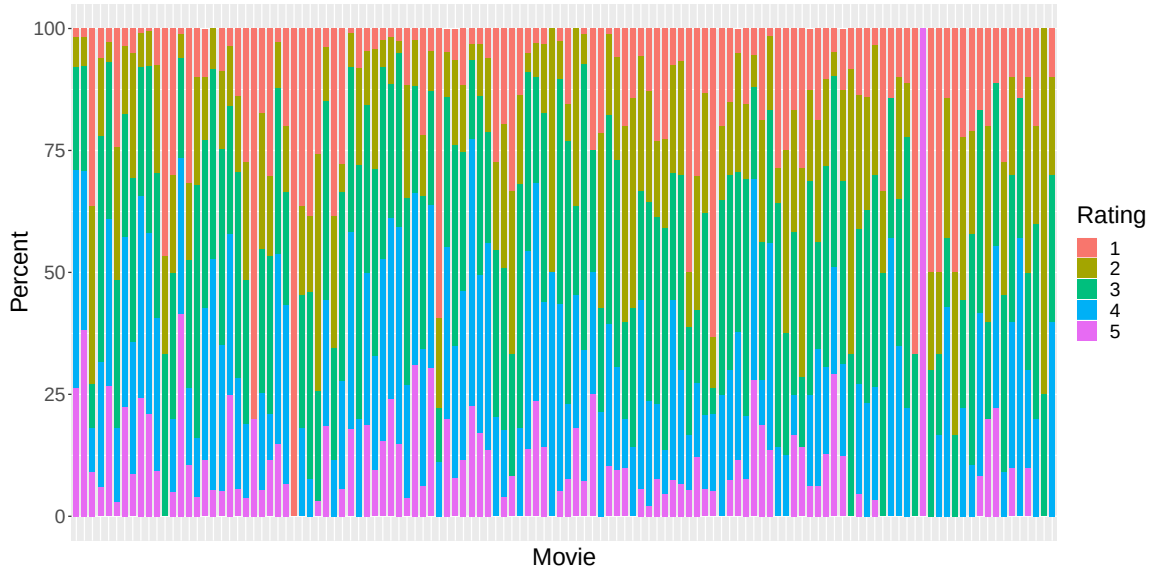

Figure 6.1.10: Percentage of ratings (1-5) for each *Children's* movies

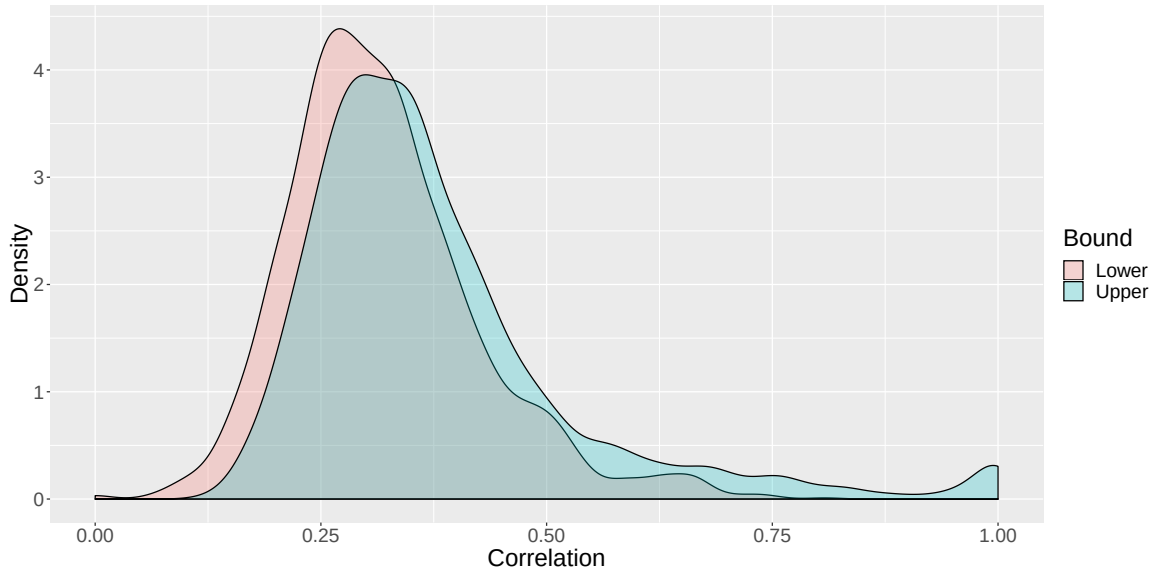

Figure 6.1.11: PDF of Upper and Lower bounds of UI_{Abs} for *Children's*

Table 6.8: Summary statistics for *Children's* genre

	Minimum	**Maximum**	**Mean**	**Median**
$\underline{\rho}_{movie}(Children)$	0	.81	.32	.30
$\overline{\rho}_{movie}(Children)$	.12	1	.38	.34
# Comparisons	1	85	36	35

bound. This demonstrates that there is more vagueness and uncertainty in these intervals in comparison with the previous example. As there are fewer movies, it is easier to see that each movie received a rating of each value, as displayed in Fig. 6.1.12. As this genre has a high

percentage of MVs in the sub-ratings matrix (97%), it is not surprising that the number of MVs that still remain, once the UI_{Abs} algorithm is run, is 41% ($n = 630$).

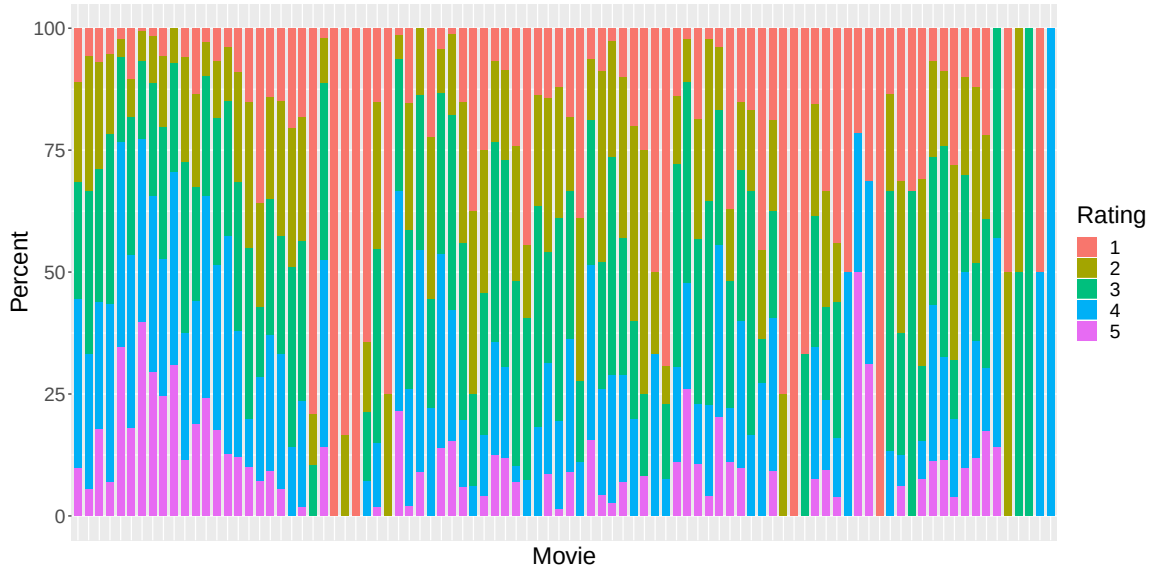

Figure 6.1.12: Percentage of ratings (1-5) for each *Horror* movies

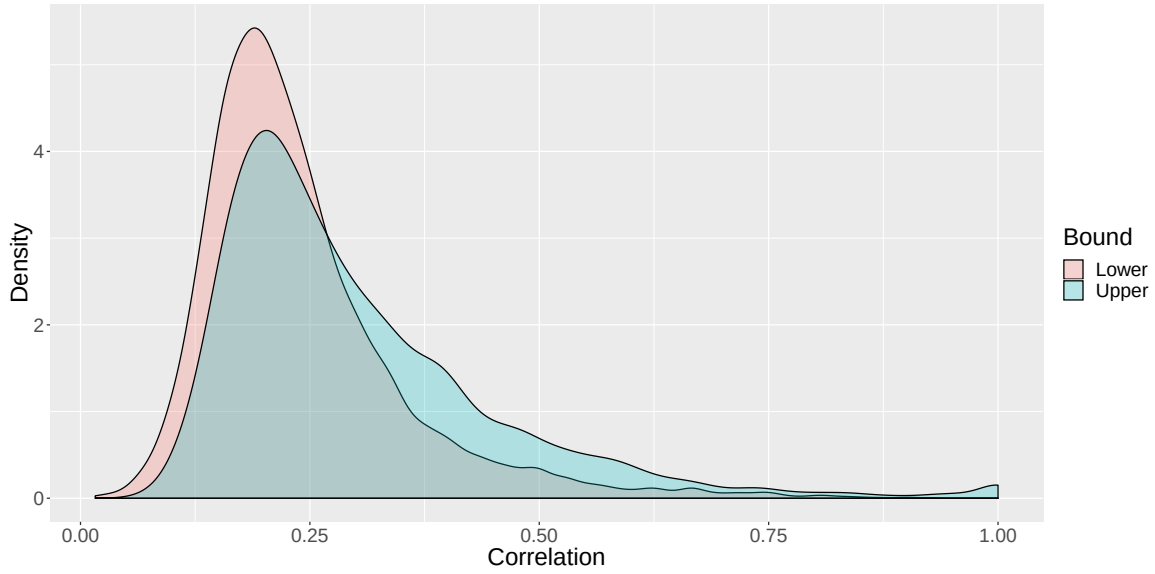

Figure 6.1.13: PDF of Upper and Lower bounds of UI_{Abs} for *Horror*

It is evident that for these selected genres, not all movies can be considered similar, hence not all pairwise correlations are exchangeable. There are many ways to create sub-matrices from ratings data. For example, a sub-matrix may be created using the director of a movie, the lead actors, the mood or the theme of the movie. As the MovieLens provides information relating to the genre of the movie, and no other information related to the film, creating different types

Table 6.9: Summary statistics for *Horror* genre

	Minimum	**Maximum**	**Mean**	**Median**
$\underline{\rho}_{movie}(Horror)$	0	.83	.26	.24
$\overline{\rho}_{movie}(Horror)$	.12	1	.34	.31
# Comparisons	1	64	24	25

of sub-matrices is not within the scope of the current research.

In relation to the application of the UI_{Abs} algorithm as an additional tool for RSs, the algorithm provides information to the user that is easy to assess; the wider the interval the less information the system knows whereas the narrower the interval the more information the system knows. By providing an interval, instead of a point-wise estimate, it is more likely to capture the true estimate of the user liking an item, hence improving the user's satisfaction of the recommendation. It may be used as a first step for new-user cold-start problems, as it encourages explicit feedback via the interval width. Therefore allowing the RS to learn the preferences of the user when no other information is available. It also provides transparency of how the recommendation is generated so that the user builds trust and an interactive relationship with the system.

6.2 Chapter Summary

This chapter set out to explore how the UI_{Abs} algorithm would perform on a real ratings dataset. The MovieLens dataset is perhaps one of the most utilised ratings dataset in articles relate to recommender systems, personalisation and preference decision-making. However, it may not be the most suitable dataset to apply the UI_{Abs} algorithm as its pattern of missingness is not MCAR and thus, generates pairwise correlations that are not exchangeable. It is also important to note that the UI_{Abs} approach does not, in and of itself, purport to be a recommender system. Therefore, given this dataset, the performance of the UI_{Abs} algorithm reflected the quality of the pairwise correlations supplied to it. It managed to perform well approximately 30% of the time (*i.e.* the absolute minimum distance from the pairwise correlation was .1 or less), even though it had a high percentage of intervals that did not capture the pairwise correlation.

As this algorithm is designed to generate an estimate interval for MVs in a correlation

matrix, it was reflective of the pairwise correlations that already existed in that matrix. This is evident from the pdfs generated from the final intervals when compared to the pdf of the pairwise comparisons. The diversity of the ratings for each movie within a sub-ratings matrix impacted on the ability of the algorithm, as one of the assumptions is that the correlations are exchangeable and exchangeable; random variables have positive correlations. The UI_{Abs} approach highlights vagueness and uncertainty as is demonstrated by the width of the final interval. This was evident with the *Horror* genre. Given that a single source of information is used, *i.e.* a ratings matrix, this vagueness and uncertainty is more transparent than if more information was incorporated into a recommendation. Simply put, more explicit feedback about your likes and dislikes, the narrower the interval becomes.

7

Discussion

The development and use of imprecise probabilities to new areas of research is gathering momentum. In addition, the research on Recommender Systems (RSs) is fast paced and rapidly changing due to *e*-commerce and Big data. This research is crossing domains and the usefulness of RSs to aid decision-making has gone beyond that of a movie, book or music recommendation. RSs are found in many areas, ranging from systems that help entire cities to those that aid decision-making for health professionals.

This thesis introduced concepts from psychology, statistics and computer science which is very much in-line with modern RSs. The elements of relational learning from a cognitive-behavioural perspective combined with decision-making and updating techniques from statistics provide an interesting and unique union. The application of Nonparametric Predictive Inference (NPI) to estimating an upper and lower bound for a correlation was proposed and demonstrated via four UI algorithms. The UI approaches are novel approaches to addressing vagueness and uncertainty within recommendation, with one of the main challenges of RSs being the cold-start problem. This has been addressed, in part, by incorporating information into the system that would not ordinarily be utilised via the UI approaches: $UI_{Untrans}$, UI_{Abs}, UI_{Sq} and UI_{Rt}.

Pearson's correlation coefficient is a popular statistical technique employed by RSs and is a well established and understood technique. Hence, it is a good starting point to introduce interval estimates. The UI approaches utilised products of pairwise correlations in order to build a final interval that estimates the true correlation coefficient. As stated previously, pairwise calculations may not always be an accurate reflection of the true underlying correlation coeffi-

cient. Given this method relies on pairwise products of pairwise correlations, the *quality* of such estimates may not be truly accurate. However, this approach appears to sit well with findings from psychology in that, as individuals, derived relations are formed based on trained relations and are updated as more information is received. In this respect, the UI approaches are *learning* and should be updated as more and more information is gathered. Another limitation of the use of a correlation coefficient is that it can only measure a linear relationship. Perhaps future research can explore the application of NPI on other statistical techniques employed by RSs.

Each of the UI approaches were evaluated on simulated data and compared with a similar technique, Robust Bayesian Correlation Learning (RBCL). The UI approaches do not require any pairwise correlations between two items in order to estimate bounds, which is an advantage over the RBCL technique. Therefore it is possible to generate a correlation estimate even when two items do not generate a pairwise correlation. The UI approaches presented here attempt to utilise information from known sources but, in addition, tries to explicitly represent vagueness within a recommendation by providing an interval recommendation rather than a precise recommendation. The UI_{Abs} performance was superior to the other UI approaches when applied to simulated data. It consistently had the highest percentage of intervals that contained the true correlation coefficient and the final intervals had the smallest absolute minimum distance from the true correlation coefficient. In the absence of any knowledge an interval of [0,1] would be produced, accurately reflecting the lack of information available. Therefore, given no data, no personalised recommendations can be produced. Much like other RS techniques, this method is proposed to be used in conjunction with other methods in order to generate a hopefully more reflective estimate of a relationship between items when there is uncertainty and vagueness in an individual's preference. However, once the level of misssingness increases, the number of interval estimates produced by the UI_{Abs} decreases. Therefore, an interval reflecting complete lack of information is produced. One of the artefacts of exchangeability, hence affecting the proposed technique, is that, unlike RBCL, only the magnitude of the correlation is possible. Hence, UI_{Abs} may be considered more of a distance measure than a correlation coefficient.

In addition to incorporating vagueness and uncertainty, the UI_{Abs} reflected transparency in the final interval via the width of the interval. As outlined in Section 1.2, one of the primary

motivations for developing an algorithm that would incorporate uncertainty and vagueness that utilises a relatively simple, widely known, statistical technique on a single data set was due to the new General Data Protection Regulation (GDPR: 25th May, 2018). Considering the algorithms discussed in the Literature Review related to cold-start problems, Chapter 3, it is apparent that they are complex in nature, incorporate a myriad of statistical techniques and multiple datasets. As stated, Article 12 of GDPR requires transparency and that these algorithms, be it with the aid of visualisation, must be easily accessible and easy to understand. Given the battery of techniques, this does not appear to be a simple task. It is evident that users do not understand how their personal data from social media is utilised nor how statistical techniques are employed to influence their behaviours and attitudes via recommendations on social media sights in light of the Cambridge Analytica scandal.

Numerous academics have demonstrated how digital footprints (every phone call or SMS; every Facebook like or Tweet; every email; every online purchase; every Wi-Fi network joined or every work-out session recorded) can combine together to create a detailed picture of the user that can be mined to determine preferences, opinions, and desires. This may provide unsettling to users and they may choose to exercise their *right to be forgotten* under the new regulation. This may have a huge impact on the statistical techniques utilised and the access to personal data. Therefore, a significant challenge to this thesis was the reliance on a single source of information. In this respect, the UI_{Abs} algorithm performed extremely well on stimulated data that produced exchangeable pairwise correlations. When applied to real data, the limitations of this approach were apparent as it was performing well in approximately 30% of the iterations. This is a reflection of the input data, the pattern of missingness was not Missing Completely At Random (MCAR) which the UI_{Abs} algorithm was created from. Nonetheless, the final intervals generated from the algorithm were reflective of the pairwise correlations found in the sub-ratings matrices.

In addition to exploring new statistical techniques to incorporate NPI for the purpose of recommendation, future research may wish to explore the utility of the UI_{Abs} in relation to generating feedback. Today's RSs are extremely accurate at predicting preference for items but one shortcoming may be the lack of feedback they require from users. This may become an issue if the availability of personal data becomes restricted under GDPR. If a RS is capable of

building confidence with and interaction in the form of explicit feedback, then both the system and the user gain. The individual user will clearly see the impact of explicit feedback, thus providing transparency as well as building a *relationship* with the user. Recall, a good algorithm *" that generates accurate recommendations is not enough to constitute a useful system from the users' perspective. The system needs to convey to the user its inner logic and why a particular recommendation is suitable for them"* (Sinha & Swearingen, 2002). Therefore, exploring the affect of interval based recommendations on explicit feedback may be worth investigating via empirical experiments.

As the level of missingness increased, the percentage of Missing Values (MVs) that remained a MV increased. Therefore, future research may explore the ability to incorporate the intervals estimates into the UI_{Abs} algorithm, allowing for the still remaining MVs to be estimated. This would require exploring what value should be taken form the interval estimate to make it point-wise, if required. This is not necessarily a trivial task and the width of the interval may need to be a determining factor.

Bibliography

Adomavicius, G, & Tuzhilin, A. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, **17**(6), 734–749.

Adomavicius, G, & Tuzhilin, A. 2011. Context-aware recommender systems. *Pages 217–253 of: Recommender systems handbook*. Springer.

Adomavicius, G, Sankaranarayanan, R, Sen, S, & Tuzhilin, A. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, **23**(1), 103–145.

Agarwal, D, & Chen, B. 2009. Regression-based latent factor models. *Pages 19–28 of: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

Allais, M. 1953. Le comportement de l'homme rationnel devant le risque: Critique des postulats et axiomes de l'école américaine. *Econometrica: Journal of the Econometric Society*, **21**(4), 503–546.

Allison, Paul D. 2002. Missing data: Quantitative applications in the social sciences. *British Journal of Mathematical and Statistical Psychology*, **55**(1), 193–196.

Arch, J J, & Craske, M G. 2008. Acceptance and commitment therapy and cognitive behavioral therapy for anxiety disorders: Different treatments, similar mechanisms? *Clinical Psychology: Science and Practice*, **15**(4), 263–279.

Augustin, T, & Cattaneo, M E G V. 2011. Foundations of Probability. *Pages 542–544 of: International Encyclopedia of Statistical Science*. Springer.

Augustin, T, & Coolen, F P A. 2004. Nonparametric predictive inference and interval probability. *Journal of Statistical Planning and Inference*, **124**(2), 251–272.

Augustin, T, Coolen, F P A, de Cooman, G, & Troffaes, M C M. 2014. *Introduction to imprecise probabilities*. John Wiley & Sons.

Baeza-Yates, R, & Ribeiro-Neto, B. 1999. *Modern information retrieval*. Vol. 463. ACM press New York.

Baker, R M, Coolen-Maturi, T, & Coolen, F P A. 2017. Nonparametric predictive inference for stock returns. *Journal of Applied Statistics*, **44**(8), 1333–1349.

Barnes, D, & Roche, B. 1996. Stimulus equivalence and relational frame theory are fundamentally different: A reply to Saunders commentary. *The Psychological Record*, **46**, 489–507.

Barnes-Holmes, D, Barnes-Holmes, Y, Power, P, Hayden, E, Milne, R, & Stewart, I. 2006. Do you really know what you believe? Developing the Implicit Relational Assessment Procedure (IRAP) as a direct measure of implicit beliefs. *The Irish Psychologist*, **32**(7), 169–177.

Barnes-Holmes, D, Barnes-Holmes, Y, Stewart, I, & Boles, S. 2011. A sketch of the Implicit Relational Assessment Procedure (IRAP) and the Relational Elaboration and Coherence (REC) model. *The Psychological Record*, **60**(3), 10.

Bayrak, O K, & Hey, J D. 2017. Expected utility theory with imprecise probability perception: explaining preference reversals. *Applied Economics Letters*, **24**(13), 906–910.

Blackledge, J T. 2003. An introduction to relational frame theory: Basics and applications. *The Behavior Analyst Today*, **3**(4), 421–432.

Borlund, P. 2003. The concept of relevance in IR. *Journal of the American Society for information Science and Technology*, **54**(10), 913–925.

Bradley, S. 2014. Imprecise probabilities. *The Stanford Encyclopedia of Philosophy (Winter 2014 ed.)*.

Burke, R, & Ramezani, M. 2011. Matching recommendation technologies and domains. *Pages 367–386 of: Recommender systems handbook*. Springer.

Burke, R, Felfernig, A, & Göker, M H. 2011. Recommender systems: An overview. *AI Magazine*, **32**(3), 13–18.

Champiri, Z D, Shahamiri, S R, & Salim, S S B. 2015. A systematic review of scholar context-aware recommender systems. *Expert Systems with Applications*, **42**(3), 1743–1758.

Champkin, J. 2011. Hal Varian and the sexy profession. *Significance*, **8**(1), 31–34.

Charles, A, & Kasilingam, R. 2015. Does Investor's Heuristics Determines Their Investment Decisions? *IIMS Journal of Management Science*, **6**(2), 113–128.

Cho, Y H, Kim, J K, & Kim, S H. 2002. A personalized recommender system based on web usage mining and decision tree induction. *Expert systems with Applications*, **23**(3), 329–342.

Codina, V, Ricci, Fo, & Ceccaroni, L. 2016. Distributional semantic pre-filtering in context-aware recommender systems. *User Modeling and User-Adapted Interaction*, **26**(1), 1–32.

Coolen, F P A. 2011. Nonparametric predictive inference. *Pages 968–970 of: International encyclopedia of statistical science*. Springer.

Coolen, F P A, & Augustin, T. 2009. A nonparametric predictive alternative to the Imprecise Dirichlet Model: the case of a known number of categories. *International Journal of Approximate Reasoning*, **50**(2), 217–230.

Coolen, F P A, & Yan, K J. 2004. Nonparametric predictive inference with right-censored data. *Journal of Statistical Planning and Inference*, **126**(1), 25–54.

Coolen, F P A, Troffaes, M C M, & Augustin, T. 2011. Imprecise probability. *Pages 645–648 of: International encyclopedia of statistical science*. Springer.

Coolen, F P A, Coolen-Maturi, T, & Al-Nefaiee, A H. 2014. Nonparametric predictive inference for system reliability using the survival signature. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, **228**(5), 437–448.

de Campos, C P, & Antonucci, A. 2015. Imprecision in Machine Learning and AI. *IEEE Intelligent Informatics Bulletin*, **16**(1), 20–22.

De Finetti, B. 1931. Funzione caratteristica di un fenomeno aleatorio. *Atti della R. Academia Nazionale dei Lincei, Serie 6. Memorie, Classe di Scienze Fisiche, Mathematice e Naturale*, **4**.

De Finetti, B. 1938. Sur la condition dequivalence partielle, Actualit& scientifques et industrielles, No. 739 (Paris: Hermann & Cie). Translated by P. Benacerraf and R. Jeffrey as On the Condition of Partial Exchangeability. *Studies in Inductive Logic and Probability*, **2**, 193–205.

De Finetti, B. 1974. *Theory of Probability (2 volumes)*. Wiley, London.

Debnath, S, Ganguly, N, & Mitra, P. 2008. Feature weighting in content based recommendation system using social network analysis. *Pages 1041–1042 of: Proceedings of the 17th international conference on World Wide Web*. ACM.

Dey, A K. 2001. Understanding and using context. *Personal and ubiquitous computing*, **5**(1), 4–7.

Draper, D, Hodges, J S, Mallows, C L, & Pregibon, D. 1993. Exchangeability and data analysis. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 9–37.

Dyer, J S, & Sarin, R K. 1982. Relative risk aversion. *Management Science*, **28**(8), 875–886.

Edwards, A W F. 1982. Pascal and the problem of points. *International Statistical Review/Revue Internationale de Statistique*, 259–266.

Ellsberg, D. 1961. Risk, ambiguity, and the Savage axioms. *The quarterly journal of economics*, **75**(4), 643–669.

Fermüller, C G, & Roschger, C. 2014. Randomized game semantics for semi-fuzzy quantifiers. *Logic Journal of the IGPL*, **22**(3), 413–439.

Fields, L, Adams, B J, Verhave, T, & Newman, S. 1993. Are stimuli in equivalence classes equally related to each other? *The Psychological Record*, **46**, 85–105.

Gelman, A. 2006. Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper). *Bayesian Analysis*, **1**(3), 515–534.

Gelman, A, Carlin, J B, Stern, H S, & Rubin, D B. 2003. *Bayesian Data Analysis, (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC.

Gigerenzer, G, & Brighton, H. 2009. Homo heuristicus: Why biased minds make better inferences. *Topics in Cognitive Science*, **1**(1), 107–143.

Gigerenzer, G, & Gaissmaier, W. 2011. Heuristic decision making. *Annual review of psychology*, **62**, 451–482.

Glasser, M. 1964. Linear regression analysis with missing observations among the independent variables. *Journal of the American Statistical Association*, **59**(307), 834–844.

Good, I J. 1962. *Subjective probability as the measure of a non-measurable set*. na.

Greenwald, A G, & Krieger, L H. 2006. Implicit bias: Scientific foundations. *California Law Review*, **94**(4), 945–967.

Gunter, B. 2017. Media Use: United Kingdom. *The International Encyclopedia of Media Effects*.

Harper, F M, & Konstan, J A. 2016. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, **5**(4), 19.

Hayes, S C, & Hayes, L J. 1989. The verbal action of the listener as a basis for rule-governance. *Pages 153–190 of: Rule-governed behavior*. Springer.

Hayes, S C, Barnes-Holmes, D, & Roche, B. 2001. *Relational frame theory: A post-Skinnerian account of human language and cognition*. Springer Science & Business Media.

Hdioud, F, Frikh, B, Benghabrit, A, & Ouhbi, B. 2016. Collaborative Filtering with Hybrid Clustering Integrated Method to Address New-Item Cold-Start Problem. *Pages 285–296 of: Intelligent Distributed Computing IX*. Springer.

Hill, B M. 1968. Posterior distribution of percentiles: Bayes' theorem for sampling from a population. *Journal of the American Statistical Association*, **63**(322), 677–691.

Hill, B M. 1988. De Finettis Theorem, Induction, and A (n) or Bayesian nonparametric predictive inference (with discussion). *Bayesian statistics*, **3**, 211–241.

Hill, W, Stead, L, Rosenstein, M, & Furnas, G. 1995. Recommending and evaluating choices in a virtual community of use. *Pages 194–201 of: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co.

Houlding, B, & Coolen, F P A. 2011. Adaptive utility and trial aversion. *Journal of Statistical Planning and Inference*, **141**(2), 734–747.

Houlding, B, & Coolen, F P A. 2012. Nonparametric predictive utility inference. *European Journal of Operational Research*, **221**(1), 222–230.

Hussey, I, Barnes-Holmes, D, & Barnes-Holmes, Y. 2015. From Relational Frame Theory to implicit attitudes and back again: Clarifying the link between RFT and IRAP research. *Current Opinion in Psychology*, **2**, 11–15.

Jaccard, P. 1912. The distribution of the flora in the alpine zone. *New phytologist*, **11**(2), 37–50.

Jannach, D, Zanker, M, Felfernig, A, & Friedrich, G. 2010. *Recommender systems: an introduction*. Cambridge University Press.

Jasberg, K, & Sizov, S. 2018. Human uncertainty and ranking error: fallacies in metric-based evaluation of recommender systems. *Pages 1358–1365 of: Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. ACM.

Jones, N, & Pu, P. 2008. *User Acceptance Issues in Music Recommender Systems*. Tech. rept. EPFL Technical Report HCI-REPORT-2009-001.

Kahneman, D, & Tversky, A. 1979. Prospect theory: An analysis of decision under risk. *Econometrica: Journal of the Econometric Society*, 263–291.

Kang, J, Condiff, K, Chang, S, Konstan, J A, Terveen, L, & Harper, F M. 2017. Understanding How People Use Natural Language to Ask for Recommendations. *Pages 229–237 of: Proceedings of the Eleventh ACM Conference on Recommender Systems*. ACM.

Kemeny, J G. 1955. Fair bets and inductive probabilities. *The Journal of Symbolic Logic*, **20**(03), 263–273.

Knight, W. 2015. The Hit Charade: An algorithm might create a playlist you enjoy,but don't mistake that for creativity. *MIT Technology Review*, **118**(6), 78–82.

Kolmogorov, A N. 1950. *Foundations of the Theory of Probability*.

Konstan, J A, Miller, B N, Maltz, D, Herlocker, J L, Gordon, L R, & Riedl, J. 1997. GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, **40**(3), 77–87.

Konstan, J A, McNee, S M, Ziegler, C-N, Torres, R, Kapoor, N, & Riedl, J. 2006. Lessons on applying automated recommender systems to information-seeking tasks. *Pages 1630–1633 of: AAAI*, vol. 6.

Koren, Y. 2010. Collaborative filtering with temporal dynamics. *Communications of the ACM*, **53**(4), 89–97.

Kosinski, M, Stillwell, D, & Graepel, T. 2013. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, 201218772.

Kovács, Á M, Kühn, Si, Gergely, G, Csibra, G, & Brass, M. 2014. Are all beliefs equal? Implicit belief attributions recruiting core brain regions of theory of mind. *PLoS One*, **9**(9), e106558.

Kuznetsov, V P. 1991. *Interval statistical models*.

Levi, I. 1974. On Indeterminate Probabilities." The Journal of Philosophy 71391–418.. 1986a. *The Paradoxes of Allais and Ellsberg." Economics and Philosophy*, 2–23.

Lewis, B W. 2015. Pairwise-complete correlation considered dangerous. *http://bwlewis.github.io/covar/missing.html*. Accessed: 17 April 2017.

Liang, K-Y, & Zeger, S L. 1986. Longitudinal data analysis using generalized linear models. *Biometrika*, **73**(1), 13–22.

Lindley, D V. 1982. The Subjectivist View of Decision-Making. *European Journal of Operational Research*, **9**(3), 213.

Lindley, D V. 1991. *Making Decisions*. Wiley New York.

Little, R J A, & Rubin, D B. 1987. *Analysis with missing data*. John Wiley & Sons, New York.

Little, R J A, & Rubin, D B. 2014. *Statistical analysis with missing data*. Vol. 333. John Wiley & Sons.

Liu, H, Hu, Zand Mian, A, Tian, Hi, & Zhu, X. 2014. A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, **56**, 156–166.

Mansoury, M, Shajari, M, Taniar, D, & Pardede, E. 2016. Improving Recommender Systems Performance on Cold-start Users and Controversial Items by a New Similarity Model. *International Journal of Web Information Systems*, **12**(2).

Marlin, B, Zemel, R S, Roweis, S, & Slaney, M. 2007. Collaborative filtering and the missing at random assumption. *Pages 267–275 of: Proceedings of UAI'07 Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*. ACM.

Marlin, B M, Zemel, R S, Roweis, S T, & Slaney, M. 2011. Recommender Systems, Missing Data and Statistical Model Estimation. *Page 2686 of: IJCAI proceedings-international joint conference on artificial intelligence*, vol. 22.

Mohammadi, J, Modares, M, & Bergerson, J. 2016. Structural Condition Assessment Using Imprecise Probability. *Pages 319–326 of: Proceedings of the 7th International Workshop on Reliable Engineering Computing (REC)*. REC.

Mooney, R J, & Roy, L. 2000. Content-based book recommending using learning for text categorization. *Pages 195–204 of: Proceedings of the fifth ACM conference on Digital libraries*. ACM.

Mowday, R T, & Sutton, R I. 1993. Organizational behavior: Linking individuals and groups to organizational contexts. *Annual review of psychology*, **44**(1), 195–229.

Nagelkerke, N J D. 1991. A note on a general definition of the coefficient of determination. *Biometrika*, **78**(3), 691–692.

Natsev, A P, Haubold, A, Tešić, J, Xie, L, & Yan, R. 2007. Semantic concept-based query expansion and re-ranking for multimedia retrieval. *Pages 991–1000 of: Proceedings of the 15th ACM international conference on Multimedia*. ACM.

Newman, D A. 2003. Longitudinal modeling with randomly and systematically missing data: A simulation of ad hoc, maximum likelihood, and multiple imputation techniques. *Organizational Research Methods*, **6**(3), 328–362.

Nicholson, E, McCourt, A, & Barnes-Holmes, D. 2013. The Implicit Relational Assessment Procedure (IRAP) as a measure of obsessive beliefs in relation to disgust. *Journal of Contextual Behavioral Science*, **2**(1-2), 23–30.

Oberguggenberger, M, King, J, & Schmelzer, B. 2009. Classical and imprecise probability methods for sensitivity analysis in engineering: A case study. *International Journal of Approximate Reasoning*, **50**(4), 680–693.

Park, S, & Chu, W. 2009. Pairwise preference regression for cold-start recommendation. *Pages 21–28 of: Proceedings of the third ACM conference on Recommender systems*. ACM.

Perrin, A. 2015. Social media usage. *Pew Research Center*.

Picard, R R, & Vander Wiel, S A. 2016. *Imprecise Probability Methods for Weapons UQ*. Tech. rept. Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Pu, P, Chen, L, & Hu, R. 2011. A Consumer-Centric Evaluation Framework for Recommender Systems. *Page 157 of: Proceedings of the fifth ACM conference on Recommender systems RecSys*.

Quiñones, R, Hayes, L J, & Hayes, S C. 2000. On the benefits of collaboration: Consumer psychology, behavioral economics and relational frame theory. *Managerial and Decision Economics*, **21**(3-4), 159–165.

Ramsey, F P. 1924. An Imaginary Conversation with John Stuart Mill. *Frank Plumpton Ramsey: Notes on Philosophy, Probability and Mathematics. Naples, Italy: Bibliopolis*, **1991**, 25.

Rubin, D B. 1976. Inference and missing data. *Biometrika*, **63**(3), 581–592.

Sarwar, B, Karypis, G, Konstan, J, & Riedl, J. 2001. Item-based collaborative filtering recommendation algorithms. *Pages 285–295 of: Proceedings of the 10th international conference on World Wide Web*. ACM.

Savage, L J. 1954. *The foundations of statistics*. New York: Wiley.

Schervish, M J. 2012. *Theory of statistics*. Springer Science & Business Media.

Schoemaker, P J. 1990. Are risk-attitudes related across domains and response modes? *Management science*, **36**(12), 1451–1463.

Sheugh, L, & Alizadeh, S H. 2015. A note on pearson correlation coefficient as a metric of similarity in recommender system. *Pages 1–6 of: AI & Robotics (IRANOPEN), 2015*. IEEE.

Simon, H A. 1956. Rational choice and the structure of the environment. *Psychological review*, **63**(2), 129–138.

Simon, H A, Dantzig, G B, Hogarth, R, Plott, C R, Raiffa, H, Schelling, T C, Shepsle, K A, Thaler, R, Tversky, A, & Winter, S. 1987. Decision making and problem solving. *Interfaces*, **17**(5), 11–31.

Sinha, R, & Swearingen, K. 2002. The role of transparency in recommender systems. *Pages 830–831 of: CHI'02 extended abstracts on Human factors in computing systems*. ACM.

Skinner, B F. 1938. *The behavior of organisms: An experimental analysis.* Appleton-Century.

Son, L H. 2015. HU-FCF++. *Engineering Applications of Artificial Intelligence*, **41**(C), 207–222.

Son, L H. 2016. Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems*, **58**, 87–104.

Son, L. H., Cuong, K. M., Minh, N. T. H., & Canh, N. Van. 2013 (Dec). An application of fuzzy geographically clustering for solving the Cold-Start problem in recommender systems. *Pages 44–49 of: Soft Computing and Pattern Recognition (SoCPaR), 2013 International Conference of.*

Stigler, G J, & Becker, G S. 1977. De gustibus non est disputandum. *The American Economic Review*, **67**(2), 76–90.

Su, X, Khoshgoftaar, Ti M, & Greiner, R. 2008. A Mixture Imputation-Boosted Collaborative Filter. *Pages 312–316 of: FLAIRS conference*.

Suhaimin, M S M, Hijazi, M H A, Alfred, R, & Coenen, F. 2017. Natural language processing based features for sarcasm detection: An investigation using bilingual social media texts. *Pages 703–709 of: Information Technology (ICIT), 2017 8th International Conference on.* IEEE.

Swearingen, K, & Sinha, R. 2002. Interaction design for recommender systems. *Pages 312–334 of: Designing Interactive Systems*, vol. 6.

Tarbox, J, Stewart, I, Roche, B, & O'Hora, D. 2013. Education, intellectual development, and relational frame theory. *In: Advances in Relational Frame Theory & Contextual Behavioral Science: Research & Application.* New Harbinger Publications.

Tintarev, N, & Masthoff, J. 2007. A survey of explanations in recommender systems. *Pages 801–810 of: Data Engineering Workshop, 2007 IEEE 23rd International Conference on.* IEEE.

Tokuda, T, Goodrich, B, Van Mechelen, I, Gelman, A, & Tuerlinckx, F. 2011. Visualizing distributions of covariance matrices. *Columbia Univ., New York, USA, Tech. Rep*, 18–18.

Troffaes, M C M, Coolen, F P, & Destercke, S. 2014. A Note on Learning Dependence Under Severe Uncertainty. *Pages 498–507 of: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems.* Springer.

Tweedale, J W, Neves-Silva, R, Jain, L C, Phillips-Wren, G, Watada, J, Howlett, R J, *et al.* 2016. *Intelligent Decision Technology Support in Practice.* Springer.

Van den Oord, A, Dieleman, S, & Schrauwen, B. 2013. Deep content-based music recommendation. *Pages 2643–2651 of: Advances in Neural Information Processing Systems.*

Vargas, S, & Castells, P. 2011. Rank and relevance in novelty and diversity metrics for recommender systems. *Pages 109–116 of: Proceedings of the fifth ACM conference on Recommender systems.* ACM.

von Neumann, J, & Morgenstern, O. 1947. Theory of Games and Economic Behavior. *Princeton: Princeton University Press.*

Walley, P. 1991. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall.

Walley, P. 1996. Inferences from multinomial data: learning about a bag of marbles. *Journal of the Royal Statistical Society. Series B (Methodological)*, 3–57.

Weichselberger, K. 2000. The theory of interval-probability as a unifying concept for uncertainty. *International Journal of Approximate Reasoning*, **24**(2), 149–170.

Wu, G, Zhang, J, & Gonzalez, R. 2004. Decision under risk. *Chap. 20, pages 399–423 of:* Koehler, D J, & Harvey, N (eds), *Blackwell handbook of judgment and decision making*. Blackwell.

Xu, X, Lizhong, Z, Xiaowei, Zo, Zhenzhen, X, & Yu, L. 2014. A feature-based regression algorithm for cold-start recommendation. *Journal of Industrial and Production Engineering*, **31**(1), 17–26.

Yin, Y, Coolen, F P A, & Coolen-Maturi, T. 2016. Combining nonparametric predictive inference and power-Weibull model for accelerated life testing. *Risk, Reliability and Safety: Innovating Theory and Practice: Proceedings of ESREL 2016 (Glasgow, Scotland, 25-29 September 2016)*, 44.

Yonker, L M, Zan, S, Scirica, C V, Jethwani, K, & Kinane, T B. 2015. Friending teens: systematic review of social media in adolescent and young adult health care. *Journal of Medical Internet Research*, **17**(1).

Zadeh, L A. 1965. Fuzzy logic and its applications. *New York, NY, USA*.

Zhang, H, Ha, L, Li, Q, & Beer, M. 2017. Imprecise probability analysis of steel structures subject to atmospheric corrosion. *Structural Safety*, **67**, 62–69.

Zhang, L. 2013. The Definition of Novelty in Recommendation System. *Journal of Engineering Science & Technology Review*, **6**(3).

Zhang, Z, Liu, C, Zhang, Y, & Zhou, T. 2010. Solving the cold-start problem in recommender systems with social tags. *EPL (Europhysics Letters)*, **92**(2), 28002.

Appendices

Appendix A. R Code

Code for EXAMPLE MATRIX

```
m=matrix(c(10,8,6,5,7,10,8,7,9,10,10,6,NA,9,NA,10,NA,5,NA,4,10,4,8,10,10), nrow=5, byrow=F)
m
mcor<-cor(m, use='pairwise')
mcor
HighCor_corvars_un<-mcor
n_top_un<-rep(0,5)

for (i in 1:length(n_top_un)){
  n_top_un[i]<-max(HighCor_corvars_un[3,(i)],HighCor_corvars_un[4,(i)])
}
length(sort(n_top_un))
n_top_un<-n_top_un[!is.na(n_top_un)]
length(n_top_un)

E_lower_n_top_un<-rep(0,length(n_top_un))
E_upper_n_top_un<-rep(0,length(n_top_un))
k<-0
for(i in seq(1:length(n_top_un))){
  k<-k+1
  E_lower_n_top_un[k]<-1/(1+i)*sum(n_top_un[1:i])
  E_upper_n_top_un[k]<-1/(1+i)*(1+sum(n_top_un[1:i]))
}

n_top_un1<-rep(0,5)

for (i in 1:length(n_top_un1)){
  n_top_un1[i]<-min(HighCor_corvars_un[3,(i)],HighCor_corvars_un[4,(i)])
}
length(sort(n_top_un1))
n_top_un1<-n_top_un1[!is.na(n_top_un1)]
length(n_top_un1)

E_lower_n_top_un1<-rep(0,length(n_top_un1))
E_upper_n_top_un1<-rep(0,length(n_top_un1))
k<-0
for(i in seq(1:length(n_top_un1))){
  k<-k+1
  E_lower_n_top_un1[k]<-1/(1+i)*sum(n_top_un1[1:i])
  E_upper_n_top_un1[k]<-1/(1+i)*(1+sum(n_top_un1[1:i]))
}

n_top_un2<-rep(0,5)

for (i in 1:length(n_top_un2)){
  n_top_un2[i]<-(HighCor_corvars_un[3,(i)]+HighCor_corvars_un[4,(i)])/2
}
```

```
length(sort(n_top_un2))
n_top_un2<-n_top_un2[!is.na(n_top_un2)]
length(n_top_un2)

E_lower_n_top_un2<-rep(0,length(n_top_un2))
E_upper_n_top_un2<-rep(0,length(n_top_un2))
k<-0
for(i in seq(1:length(n_top_un2))){
  k<-k+1
  E_lower_n_top_un2[k]<-1/(1+i)*sum(n_top_un2[1:i])
  E_upper_n_top_un2[k]<-1/(1+i)*(1+sum(n_top_un2[1:i]))
}


n_top_un3<-rep(0,5)

for (i in 1:length(n_top_un3)){
  n_top_un3[i]<-HighCor_corvars_un[3,(i)]*HighCor_corvars_un[4,(i)]
}
length(sort(n_top_un3))
n_top_un3<-n_top_un3[!is.na(n_top_un3)]
length(n_top_un3)

E_lower_n_top_un3<-rep(0,length(n_top_un3))
E_upper_n_top_un3<-rep(0,length(n_top_un3))
k<-0
for(i in seq(1:length(n_top_un3))){
  k<-k+1
  E_lower_n_top_un3[k]<-1/(1+i)*sum(n_top_un3[1:i])
  E_upper_n_top_un3[k]<-1/(1+i)*(1+sum(n_top_un3[1:i]))
}


win.graph()
par(mar=c(5,5,1,1))
plot(E_upper_n_top_un, type='l', ylim=c(0,1),
    xlab='Number of Intervals',  ylab='Correlation Interval',
    cex.lab=2.5, cex.axis=2, col='darkorchid4')
grid (NULL,NULL, lty = 6, col = "lightgrey")
polygon(c(1:(length(E_upper_n_top_un)), rev(1:(length(E_lower_n_top_un)))),
      c(E_upper_n_top_un,rev(E_lower_n_top_un)),col=rgb(.408,.132,.545,0.5), border=NA)
lines(E_lower_n_top_un,lty=1, col='darkorchid4')

win.graph()
par(mar=c(5,5,1,1))
plot(E_upper_n_top_un1, type='l', ylim=c(0,1),
    xlab='Number of Intervals',  ylab='Correlation Interval',
    cex.lab=2.5, cex.axis=2, col='dodgerblue4')
```

```
grid (NULL,NULL, lty = 6, col = "lightgrey")
polygon(c(1:(length(E_upper_n_top_un1)), rev(1:(length(E_lower_n_top_un1)))),
      c(E_upper_n_top_un1,rev(E_lower_n_top_un1)),col=rgb(.064,.305,.545,0.5), border=NA)
lines(E_lower_n_top_un1,lty=1, col='dodgerblue4')

win.graph()
par(mar=c(5,5,1,1))
plot(E_upper_n_top_un2, type='l', ylim=c(0,1),
    xlab='Number of Intervals',  ylab='Correlation Interval',
    cex.lab=2.5, cex.axis=2, col='darkgreen')
grid (NULL,NULL, lty = 6, col = "lightgrey")
polygon(c(1:(length(E_upper_n_top_un2)), rev(1:(length(E_lower_n_top_un2)))),
      c(E_upper_n_top_un2,rev(E_lower_n_top_un2)),col=rgb(0,.392,0,0.5), border=NA)
lines(E_lower_n_top_un2,lty=1, col='darkgreen')

win.graph()
par(mar=c(5,5,1,1))
plot(E_upper_n_top_un3, type='l', ylim=c(0,1),
    xlab='Number of Intervals',  ylab='Correlation Interval',
    cex.lab=2.5, cex.axis=2, col='lightcyan4')
grid (NULL,NULL, lty = 6, col = "lightgrey")
polygon(c(1:(length(E_upper_n_top_un3)), rev(1:(length(E_lower_n_top_un3)))),
      c(E_upper_n_top_un3,rev(E_lower_n_top_un3)),col=rgb(.478,.545,.545,0.5), border=NA)
lines(E_lower_n_top_un3,lty=1, col='lightcyan4')

n_top_un4<-rep(0,5)
for (i in 1:length(n_top_un4)){
  n_top_un4[i]<-HighCor_corvars_un[3,(i)]
}
length(sort(n_top_un4))
n_top_un4<-n_top_un4[!is.na(n_top_un4)]
n_top_un4<-n_top_un4[-3]
length(n_top_un4)

E_lower_n_top_un4<-rep(0,length(n_top_un4))
E_upper_n_top_un4<-rep(0,length(n_top_un4))
k<-0
for(i in seq(1:length(n_top_un4))){
  k<-k+1
  E_lower_n_top_un4[k]<-1/(1+i)*sum(n_top_un4[1:i])
  E_upper_n_top_un4[k]<-1/(1+i)*(1+sum(n_top_un4[1:i]))
}


n_top_un5<-rep(0,5)
for (i in 1:length(n_top_un5)){
  n_top_un5[i]<-HighCor_corvars_un[4,(i)]
}
```

```
length(sort(n_top_un5))
n_top_un5<-n_top_un5[!is.na(n_top_un5)]
n_top_un5<-n_top_un5[-3]
length(n_top_un5)

E_lower_n_top_un5<-rep(0,length(n_top_un5))
E_upper_n_top_un5<-rep(0,length(n_top_un5))
k<-0
for(i in seq(1:length(n_top_un5))){
  k<-k+1
  E_lower_n_top_un5[k]<-1/(1+i)*sum(n_top_un5[1:i])
  E_upper_n_top_un5[k]<-1/(1+i)*(1+sum(n_top_un5[1:i]))
}

win.graph()
par(mar=c(5,5,1,1))
plot(E_upper_n_top_un4, type='l', ylim=c(0,1),
    xlab='Number of Intervals',  ylab='Correlation Interval',
    cex.lab=2.5, cex.axis=2, col='lightcyan4')
grid (NULL,NULL, lty = 6, col = "lightgrey")
polygon(c(1:(length(E_upper_n_top_un4)), rev(1:(length(E_lower_n_top_un4)))),
      c(E_upper_n_top_un4,rev( )),col=rgb(.478,.545,.545,0.2), border=NA)
lines(E_lower_n_top_un4,lty=1, col='lightcyan4')
lines(E_upper_n_top_un5,lty=1, col='lightcyan4')
polygon(c(1:(length(E_upper_n_top_un5)), rev(1:(length(E_lower_n_top_un5)))),
      c(E_upper_n_top_un5,rev(E_lower_n_top_un5)),col=rgb(.478,.545,.545,0.4), border=NA)
lines(E_lower_n_top_un5,lty=1, col='lightcyan4')
```

Code for SIMMULATION MATRIX (NORMAL)

```
#### HIGH CORRELATIONS ######
#HighCor_mu <- rep(0,500)
#set.seed(7732)
#x<-runif(250000, .95, .99)
#HighCor_sigma <- matrix(x, nrow=500, ncol=500)
#is.positive.definite(HighCor_sigma)
#HighCor_sigma<-make.positive.definite(HighCor_sigma)
#is.positive.definite(HighCor_sigma)
#set.seed(7732)
#HighCor_rawvars <- mvrnorm(n=500, mu=HighCor_mu, Sigma=HighCor_sigma)
#HighCor_corvars<-cor(HighCor_rawvars)
#dim(HighCor_rawvars)
#round(HighCor_corvars[1:10, 1:10],2)
#dim(HighCor_corvars)
```

```
#write.csv(HighCor_corvars,file='7732HighCor_corvars.csv')
#write.csv(HighCor_rawvars,file='7732HighCor_rawvars.csv')

#win.graph()
#corrplot(HighCor_corvars, method="color",cl.pos="b", tl.pos="n")

#print(xtable(HighCor_corvars[1:12,1:12],digits=2), type="latex", file="1cormat_correlates.tex")

#set.seed(7732)
#mask<-matrix(rbinom(250000,1,.05),nrow=500,ncol=500)
#HighCor_rawvars95NA<-mask*HighCor_rawvars
#HighCor_rawvars95NA <- ifelse(mask, HighCor_rawvars, NA)

#HighCor_corvars95NA<-cor(HighCor_rawvars95NA, use='pairwise')
#write.csv(HighCor_rawvars95NA,file='7732HighCor_rawvars95NA.csv')
#write.csv(HighCor_corvars95NA,file='7732HighCor_corvars95NA.csv')
#win.graph()
#corrplot(HighCor_corvars95NA, method="color",cl.pos="b", tl.pos="n")

#print(xtable(HighCor_corvars95NA[1:12,1:12],digits=2), type="latex",
file="cormat95NA1_correlates.tex")
#HighCor_rawvars95NA[1:10, 1:10]
#round(HighCor_corvars95NA[1:10, 1:10],2)

HighCor_corvars<-read.csv('7732HighCor_corvars.csv', sep=',',header = T)
HighCor_corvars<-as.matrix(HighCor_corvars[,-1])
HighCor_rawvars<-read.csv('7732HighCor_rawvars.csv', sep=',',header = T)
HighCor_rawvars<-as.matrix(HighCor_rawvars[,-1])
HighCor_corvars95NA<-read.csv('7732HighCor_corvars95NA.csv', sep=',',header = T)
HighCor_corvars95NA<-as.matrix(HighCor_corvars95NA[,-1])
HighCor_rawvars95NA<-read.csv('7732HighCor_rawvars95NA.csv', sep=',',header = T)
HighCor_rawvars95NA<-as.matrix(HighCor_rawvars95NA[,-1])

HighCor_corvars_fill<-HighCor_corvars95NA
#I want just the lower triange to find where NAs are
HighCor_corvars_fill[lower.tri(HighCor_corvars_fill,diag=TRUE)]<-0
MV<-as.matrix(which(is.na(HighCor_corvars_fill), arr.ind=TRUE))#location of i and j
#I don't want the diagonal to be used
diag(HighCor_corvars95NA) <- NA

#location of ith row
locationi<-MV[,1]
#location of jth column
locationj<-MV[,2]

###Untrans
#Function to calculate the lower bound
lower<-function(t,top){
```

```
  1/(1+t)*sum(top[1:t])
}

#Function to calculate the upper bound
upper<-function(t,top){
  1/(1+t)*(1+sum(top[1:t]))
}

sparse<-HighCor_corvars95NA

#####Absolute
#Function to calculate the lower bound
lower<-function(t,top){
  1/(1+t)*sum(top[1:t])
}

#Function to calculate the upper bound
upper<-function(t,top){
  1/(1+t)*(1+sum(top[1:t]))
}

sparse<-abs(HighCor_corvars95NA)

#####Squared
#Function to calculate the lower bound
lower<-function(t,top){
  1/(1+t)*sum(top[1:t])
}

#Function to calculate the upper bound
upper<-function(t,top){
  1/(1+t)*(1+sum(top[1:t]))
}

sparse<-HighCor_corvars95NA*HighCor_corvars95NA

#####Root
#Function to calculate the lower bound
lower<-function(t,top){
  sqrt(1/(1+t)*sum(top[1:t]))
}

#Function to calculate the upper bound
upper<-function(t,top){
  sqrt(1/(1+t)*(1+sum(top[1:t])))
}

sparse<-HighCor_corvars95NA*HighCor_corvars95NA
```

```
truecor<-c()
#vector of number of pairwise comaparisons
pairwise<-c()
#vector for the absolute minimum distance of last interval from true correlation
dis<-c()
#vector of the lower bounds for each j, i
lowerbound<-c()
#vector of the upper bounds for each j, i
upperbound<-c()
#vector of the percent the true correlation is captured by the interval
percent<-c()



#Function to evaluate the intervals
eval<-function(s,j){
  ifelse(((lbound[s]<= truecor[j]) && (ubound[s]>= truecor[j])), 1, 0)
}
for (j in 1:length(locationi)){
  # let this be the x value for calculating bounds
  x<-locationi[j]
  # let this be the y value for calculating bounds
  y<-locationj[j]
  # keep a record of the true correlation
  truecor[j] <- HighCor_corvars[x,y]
  #fill in the n-top vector
  n_top<-(sparse[x,]*sparse[y,])
  #remove any NAs
  top<-n_top[!is.na(n_top)]
  #apply upper bound fuction
  ubound<-sapply(1:length(top),upper,top=top)
  #apply lower bound function
  lbound<-sapply(1:length(top),lower,top=top)
  #set up a vector for evaluating the intervals
  a<-sapply(1:length(ubound),eval,j=j)
  #calculate the percentage of time it is inside
  per<-sum(a/length(ubound)*100)
  #calculate the distance from upper and lower bound of the final interval
  b<-c(lbound[length(lbound)],ubound[length(ubound)])
  #produce a distance vector
  distance<-round(min(abs(lbound[length(lbound)]-truecor[j]),abs(ubound[length(ubound)]-
truecor[j])),3)
  lowerbound[j]<-round(lbound[length(lbound)],3)# return last lower bound
  upperbound[j]<-round(ubound[length(ubound)],3)# return last upper bound
  pairwise[j]<-length(top) #return number of pairwise comparisons
  percent[j] <- round(per,3) # return percent of intervals true correlations captured
  dis[j]<-distance # return min absolute distance from true correlation
```

```
  print(j)
}


######Untrans
Results<-cbind(locationi,locationj,truecor,pairwise,lowerbound,upperbound,percent,dis)
write.csv(Results,file='Untrans_Results.csv')

######Absolute
Results<-cbind(locationi,locationj,truecor,pairwise,lowerbound,upperbound,percent,dis)
write.csv(Results,file='Absolute_Results.csv')

######Squared
Results<-cbind(locationi,locationj,truecor,pairwise,lowerbound,upperbound,percent,dis)
write.csv(Results,file='Squared_Results.csv')

######Root
Results<-cbind(locationi,locationj,truecor,pairwise,lowerbound,upperbound,percent,dis)
write.csv(Results,file='Root_Results.csv')
```

Code for SIMMULATION MATRIX (OTHER DISTRIBUTIONS)

```
HighCor_mu <- rep(0,500)
set.seed(7732)
x<-runif(250000, .95, .99)
HighCor_sigma <- matrix(x, nrow=500, ncol=500)
is.positive.definite(HighCor_sigma)
HighCor_sigma<-make.positive.definite(HighCor_sigma)
is.positive.definite(HighCor_sigma)
set.seed(7732)
HighCor_rawvars <- mvrnorm(n=500, mu=HighCor_mu, Sigma=HighCor_sigma)
HighCor_corvars<-cor(HighCor_rawvars)
win.graph()
corrplot(HighCor_corvars, method="color",cl.pos="b", tl.pos="n")

HighCor_pvars <- pnorm(HighCor_rawvars)
HighCor_corpvars<-cor(HighCor_pvars)
win.graph()
corrplot(HighCor_corpvars, method="color",cl.pos="b", tl.pos="n")

#Poisson
HighCor_poisvars <- qpois(HighCor_pvars, 5)
HighCor_corpoisvars<-cor(HighCor_poisvars)
win.graph()
corrplot(HighCor_corpoisvars, method="color",cl.pos="b", tl.pos="n")

#binomial
```

```
HighCor_binomvars <- qpois(1-HighCor_pvars, 3, .25)
HighCor_corbinomvars<-cor(HighCor_binomvars)
win.graph()
corrplot(HighCor_corbinomvars, method="color",cl.pos="b", tl.pos="n")

#exponential
HighCor_expvars <- qexp(HighCor_pvars)
HighCor_corexpvars<-cor(HighCor_expvars)
win.graph()
corrplot(HighCor_corexpvars, method="color",cl.pos="b", tl.pos="n")

#skewed normal
HighCor_snvars <- qsn(HighCor_pvars, 5, 2, 5)
HighCor_snvars<-matrix(HighCor_snvars, 500,500)
dim(HighCor_snvars)
HighCor_corsnvars<-cor(HighCor_snvars)
win.graph()
corrplot(HighCor_corsnvars, method="color",cl.pos="b", tl.pos="n")

#Hyper
HighCor_hypvars <- qhyper(HighCor_pvars, 495,5,100)
HighCor_corhypvars<-cor(HighCor_hypvars)
win.graph()
corrplot(HighCor_corhypvars, method="color",cl.pos="b", tl.pos="n")

##beta
HighCor_betavars <- qbeta(HighCor_pvars, .5, .5)
HighCor_corbetavars<-cor(HighCor_betavars)
win.graph()
corrplot(HighCor_corbetavars, method="color",cl.pos="b", tl.pos="n")
```

Code for MOVIELENS MATRIX

```
user_col_names <- c('user_id', 'age', 'sex', 'occupation', 'zip_code')
users <- read.csv('u.user', sep='|', col.names=user_col_names, header=FALSE)
rating_col_names<-c('user_id', 'movie_id', 'rating', 'timestamp')
rating<-read.csv('u.data', sep='', col.names=rating_col_names, header=FALSE)
rating[1:10,]
dim(rating) #1000 x 4

class(rating) #df, need to know what I am sorting
rating<-rating[order(rating$movie_id),] # put it in order so that I can keep track of the movie ids
rating[1:10,]
rating<-rating[order(rating$user_id),] #need to sort it again by users
rating[1:10,]

genre_col_names<-c('Genre', 'Genre Code')
```

```
genre<-read.csv('u.genre', sep='|', col.names=genre_col_names, header=FALSE)
item_col_names<-c( 'movie id', 'movie title', 'release date',
              'video release date','IMDb URL','unknown',
              'Action','Adventure', 'Animation','Children',
              'Comedy','Crime','Documentary','Drama', 'Fantasy',
              'Film-Noir','Horror', 'Musical', 'Mystery', 'Romance',
              'Sci-Fi','Thriller', 'War', 'Western')
item<-read.csv('u.item', sep='|', col.names=item_col_names, header=FALSE)
dim(item) # 1682 x 24

names(item)

##### Make overall matrix #####
userid <- length(unique(rating$user_id)) #943 users
movieid <- length(unique(rating$item_id)) #1682 movies
rating<-as.matrix(rating) #make it a matrix


#make it a matrix so that row=user ID and column=movie ID
ratmat<-matrix(0,nrow=943, ncol=1682) #use 0 to make it easier to remove empty columns
for (i in 1:100000){
  ratmat[rating[i,1],rating[i,2]]=rating[i,3]
}

dim(ratmat) # 943 user IDs 1682 Movie IDs
ratmat[ratmat==0]<-NA
rownames(ratmat)<-rownames(ratmat, do.NULL = FALSE, prefix = "User") #label rows
colnames(ratmat)<-colnames(ratmat, do.NULL = FALSE, prefix = "Movie") #label columns
#this means that my movie IDs stay the same when I remove the empty columns

action<-c()

for (i in 1:1682){
  if (item$Action[i]==1){
    action[i]<-item$movie.id[i]
  }
}
action<-na.omit(action)
action[1:10]
length(action) #251 action movies
m<-rep('Movie', 251)
action<-paste(m, action, sep="")

#extract these movies
library(Hmisc)

action_ratmat<-ratmat[, colnames(ratmat) %in% action]#retain only
#the colnames that match the names in action
```

```
dim(action_ratmat) #943 x 251
action_ratmat[1:6, 1:6]

cor_action<-cor(action_ratmat, method="pearson", use="pairwise")
dim(cor_action)
pMiss <- function(x){sum(is.na(x))/length(x)*100}
CorNAs<-apply(cor_action, 2, pMiss)
diag(cor_action) <- NA
cor_action_fill<-cor_action
cor_action_fill[lower.tri(cor_action_fill,diag=TRUE)]<-NA
cor_action_fill[1:5,1:5]
MV<-as.matrix(which(!is.na(cor_action_fill), arr.ind=TRUE))#location of i and j
#I don't want the diagonal to be used
MV[1:5,]

#location of ith row
locationi<-MV[,1]
#location of jth column
locationj<-MV[,2]

HighCor_corvars95NA<-cor_action
#####Absolute
#Function to calculate the lower bound
lower<-function(t,top){
  1/(1+t)*sum(top[1:t])
}

#Function to calculate the upper bound
upper<-function(t,top){
  1/(1+t)*(1+sum(top[1:t]))
}

sparse<-abs(HighCor_corvars95NA)

truecor<-c()
#vector of number of pairwise comaparisons
pairwise<-c()
#vector for the absolute minimum distance of last interval from true correlation
dis<-c()
#vector of the lower bounds for each j, i
lowerbound<-c()
#vector of the upper bounds for each j, i
upperbound<-c()
#vector of the percent the true correlation is captured by the interval
percent<-c()
```

```
#Function to evaluate the intervals
eval<-function(s,j){
  ifelse(((lbound[s]<= truecor[j]) && (ubound[s]>= truecor[j])), 1, 0)
}
for (j in 1:length(locationi)){
  # let this be the x value for calculating bounds
  x<-locationi[j]
  # let this be the y value for calculating bounds
  y<-locationj[j]
  # keep a record of the true correlation
  truecor[j] <- abs(HighCor_corvars95NA[x,y])
  missing<-HighCor_corvars95NA[x,y]
  HighCor_corvars95NA[x,y]<-NA
  HighCor_corvars95NA[y,x]<-NA
  #fill in the n-top vector
  n_top<-(sparse[x,]*sparse[y,])
  #remove any NAs
  top<-n_top[!is.na(n_top)]
  #apply upper bound fuction
  ubound<-sapply(1:length(top),upper,top=top)
  #apply lower bound function
  lbound<-sapply(1:length(top),lower,top=top)
  #set up a vector for evaluating the intervals
  a<-sapply(1:length(ubound),eval,j=j)
  #calculate the percentage of time it is inside
  per<-sum(a/length(ubound)*100)
  #calculate the distance from upper and lower bound of the final interval
  b<-c(lbound[length(lbound)],ubound[length(ubound)])
  #produce a distance vector
  distance<-round(min(abs(lbound[length(lbound)]-truecor[j]),abs(ubound[length(ubound)]-
truecor[j])),3)
  lowerbound[j]<-round(lbound[length(lbound)],3)# return last lower bound
  upperbound[j]<-round(ubound[length(ubound)],3)# return last upper bound
  pairwise[j]<-length(top) #return number of pairwise comparisons
  percent[j] <- round(per,3) # return percent of intervals true correlations captured
  dis[j]<-distance # return min absolute distance from true correlation
  HighCor_corvars95NA[x,y]<-missing
  HighCor_corvars95NA[y,x]<-missing
  print(j)
}

Results<-cbind(locationi,locationj,truecor,pairwise,lowerbound,upperbound,percent,dis)
write.csv(Results,file='ABS_Movie_Lens_Absolute_Results.csv')
```