CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Investigating the Applicability of the Bayesian Plackett-Luce Model in Software Engineering Problems

Master's thesis in Software Engineering and Technology

Vallisha Somayagi

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2021

Master's thesis 2021

Investigating the Applicability of the Bayesian Plackett-Luce Model in Software Engineering Problems

Vallisha Somayagi

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2021

Investigating the Applicability of the Bayesian Plackett-Luce Model in Software Engineering Problems

Vallisha Somayagi

Supervisor: David Issa Mattos, Department of Computer Science and Engineering
Examiner: Jennifer Horkoff, Department of Computer Science and Engineering

Master's Thesis 2021
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Investigating the Applicability of the Bayesian Plackett-Luce Model in Software Engineering Problems.

Vallisha Somayagi
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The statistical ranking has been used for ordering the artifacts based on the importance or priority in different problems in software engineering (SE) research. While frequentist statistical ranking methods such as the Friedman test and area under the curve are commonly found in research, these methods face many limitations. For instance, the Friedman test may lack power if the sample size is small and focuses on hypothesis testing rather than estimating effects. Similarly, the area under the curve method is inconsistent and unreliable in choosing confidence scales. Frequentist methods can lead to lower conclusion validity and interpretation pitfalls.

To address these limitations, we introduce the Bayesian Plackett-Luce model and examine its applicability to SE research. Following a design science methodology, iteratively developed an R package for the BPL model. We examined the applicability of this package with three SE datasets and compared it with the other ranking models. Further evaluation with SE researchers confirms the suitability of the Bayesian Plackett-Luce model for ranking in SE. This thesis shows that:
First, the Bayesian Plackett-Luce model is suitable for ranking software engineering problems. Second, the additional information about the data given by the density plot in the Bayesian Plackett-Luce model is the advantage compared with other ranking models. The additional information is vital for making someone consider using the BPL model instead of other ranking models.

Keywords: ranking, software engineering, SE, statistical, frequentist, Bayesian, Plackett-Luce model, researchers, datasets.

Acknowledgements

The author of this thesis would like to thank the supervisor Mr David Issa Mattos for his invaluable guidance and the examiner Jennifer Hofkoff.

Vallisha Somayagi, Gothenburg, September 2021

Contents

List of Figures

List of Tables

1
Introduction

The act or an instance of listing artifacts, tools, or objects in the order of importance, quality, priority is known as ranking. A ranking is a relationship between a set of artifacts [7] and is helpful to gain information on data and identify relevant artifacts. It is always challenging for a researcher to provide correct and proper priority among data. A researcher can rank based on benchmark data, survey data, or any other measurements.

While ranking based on benchmark data, the common goal is to derive an overall ranking of all the artifacts in a benchmark experiment [36]. The ranking based on benchmark data has several uses. However, the most prominent one is identifying an overall best artifact or object. When ranking based on survey data [37], a ranking question asks the respondent to set things in order, usually order of preference. So the respondent has to think wisely and choose an object based on the object's importance. This ranking based on a survey helps researchers to understand the relationship between the objects.

These types of ranking have been used in various fields like sports, economics, politics, and many other fields. For example, a study is conducted to determine hospital consultants' preferences for the format and content of radiology reports [38]. Ninety-nine questionnaires were sent to forty-nine consultant staff to rank various hypothetical reports in order of preference. They were asked whether they felt other commonly included features of a radiology report were of value. Rank data were analyzed using the Friedman statistic test. This type of ordering of artifacts is often also seen in the area of Software Engineering research.

Usage of rankings in Software Engineering research helps present the artifacts' usefulness more understandably. The ranking is helpful to measure software data or artifacts to know how good it is. For example, a study is conducted to rank the classifiers by applying them on different datasets [24]. This study uses the Friedman test to rank the classifiers. In this research example, researchers designed three different versions of the Friedman test and applied them on thirty real-world datasets. The above mentioned is just an example to show the applicability of ranking in the Software Engineering field.

In ranking the data, there are many statistical methods and non-statistical methods. In some specific cases, it requires to use of statistical methods to rank the data. Statistical methods for ranking are helpful to understand and create ranks under

uncertainty. To say one is better than another object, we get conclusive evidence in statistics. Also, statistics give an unbiased view of what the data tells us.

In statistics, ranking is the data transformation in which numerical or ordinal values are replaced by their rank when the data is sorted. For example, the numerical data 3.3, 2.7, 5.7, 7.2 were observed. The ranks of these data would be 2, 1, 3, and 4, respectively. The same way the ordinal data good, best, better would be replaced by 3, 1, 2. In these examples, there would be many rows on this dataset. All these data would be given to the statistical ranking model to get a final rank list. Also, statistical methods help in designing experiments, analyzing, and interpreting data.

Statistical ranking methods are also practically helpful in the software engineering industry. For example, social media platforms can use the statistical ranking method to rank upcoming features in the pipeline based on the users' feedback. Software companies can survey users to rank the future updates plans. The result of this can be given as the input to the ranking model. The results of this ranking model guide the company to work on new features based on the priority given by the users. The above mentioned is just one use case of the ranking in Software Engineering.

Many statistical methods are used for ranking in the Software Engineering area, such as the Friedman test, Area under Curve, and other techniques. The Friedman test is a nonparametric test with repeated measures [4]. The procedure involves ranking each row together, then considering the values of ranks by columns. There are many nonparametric tests like the Friedman test, Wilcoxon signed-ranked test, Mann- Whitney U. Some of the disadvantages in using nonparametric statistical methods are [22]:

- Nonparametric tests may lack power if the sample size is small.
- These methods are focused on hypothesis testing rather than estimation of effects.
- Nonparametric tests require more computing time compared to some other statistical methods.

The area under curve method considers sensitivity and specificity equally important. This method is inconsistent and unreliable in choosing confidence scales [25]. These are the disadvantages of using the area under the curve for ranking. Despite using all these statistical models for ranking, SE research has not used the Plackett-Luce model (PL) for ranking SE problems.

The Plackett–Luce model is based on Luce's axiom of choice. This hypothesis states that the probability of choosing one item over another does not depend on the other items available for selection in that choice [1]. The Plackett-Luce model accommodates both ties and partial rankings. This model accepts ranked lists and numerical values as the input. The Plackett-Luce model has many advantages over nonpara-

metric and other statistical models with uncertainties (PageRank, ordered AUC). Benefits of the Plackett-Luce model are [1]:

- Scalability of the algorithm.
- Avoids overfitting sparse data.

Additionally, researchers can easily extend the Plackett-Luce model to consider different problem structures such as repeated measures, random effects, comparable effect size measures, and extension to Bayesian models.

Also, SE research has not thoroughly utilized the potential of Bayesian statistics for ranking and has focused on frequentist ranks. Frequentist statistics is a type of statistical approach in which conclusions are drawn based on the frequency of an event. Frequentist statistics is about repeatability, gathering more data, and regard the population value as fixed. There are research works on the frequentist Plackett-Luce model for ranking. However, frequentist statistics have been widely misused and have many pitfalls. Some of the drawbacks of the frequentist statistics are [13]:

- Lack of information regarding the null hypothesis [18].
- Misinterpretation of meaning of confidence intervals [15].
- Lack of transparency in reporting statistical procedures [16].

Bayesian data analysis techniques are used to overcome the shortcomings of frequentist statistics. Bayesian data analysis treats all unknown quantities in the statistical model as random variables, contrasting with the fixed constants from the frequentist approach. Bayesian statistical methods start from existing prior beliefs and update them with data to obtain posterior beliefs that can be used as the basis for inference decisions. They can provide better results that are simultaneously robust and nuanced. Many empirical studies prove how Bayesian techniques can overcome frequentist pitfalls [14]. Bayesian inference advocates the usage of priors, regularisation, handling models with many parameters or latent variables, and uncertainty propagation. Also, it provides easily interpretable results and a convenient setting for a wide range of models. Although other areas have been using a Bayesian Plackett-Luce model, this model has not been used in Software Engineering research [17].

Despite the well-known and documented advantages of Bayesian data analysis [2][3], [13]-[18] SE research has not used the Bayesian version of the Plackett-Luce model. A preliminary investigation of tools in the R ecosystem (Comprehensive R Archive Network) suggests many tools are available for ranking with nonparametric tests. However, there are no tools that implement the Bayesian Plackt-Luce model. Moreover, to the best of our knowledge, no SE research publications have utilized the BPL model for ranking.

In some scenarios, user-given rank lists have to be considered as the input for ranking. In this case, we only have rank information. Also, it is possible to convert accuracy metrics or continuous values into a ranked list as the input for ranking models for ranking. The statistical model must satisfy these SE scenarios to say the model is suitable for SE research. There is a clear research scope for discussion of the suitability of the Plackett-Luce model in software engineering along with the development of the Bayesian Plackett-Luce package.

The Bayesian Plackett-Luce model is developed as the package in the R language because it is easier to reuse and share code in the public domain. The development of the package and testing with SE problems helps to know the applicability of the Plackett-Luce model for ranking in Software Engineering. Comparing the Bayesian Plackett-Luce model with other ranking models provides more evidence about the model's applicability for SE problems. Also, the evaluation with SE researchers confirms the suitability of the Bayesian Plackett-Luce model for ranking in SE. This study can lead to significant improvements in how statistical ranking results are presented and interpreted.

The remainder of this thesis is organized as follows. Chapter 2 contains related topics to our thesis. Chapter 3 discusses the Bayesian Plackett-Luce model in detail. Chapter 4 presents the objectives of this thesis, the research questions, and the methodology used in the study. Also, it introduces collected datasets for the analyses of the developed Bayesian Plackett-Luce model. Chapter 5 presents the implementation of the package along with libraries used and demonstrates the usage of the developed BPL package. Chapter 6 discusses the results of the analyses and evaluation. Chapter 7 discusses various aspects of the thesis, and finally, chapter 8 concludes this thesis and discusses future work.

2
Related Works

Ranking data commonly arise from situations that are desired to rank a set of individuals or objects under some criterion. Examples of ranking data are found in various areas like politics [26], voting and elections [27], and health economics [28]. These ranking methods use descriptive statistics to present an overall picture of ranking data. Descriptive statistics do not just provide a summary of the ranking data. However, they are also often suggestive of the appropriate direction to analyze the data. Alvo et al. [2] investigated the probability modeling for ranking the uniform distribution or non-uniform distribution.

Statistical modeling for ranking data is an efficient way to understand people's perception and preference on different objects [2]. Various statistical models for ranking data have been developed, where many problems involving many objects emerged. In a review paper on statistical models for ranking data, Critchlow et al. [29] broadly categorized these models into four classes:

- Order statistics models.
  The basic idea behind this approach is that a judge may have a taste that varies from one moment to the next, depending on how he perceives the object in question. This taste of the judge is not perfectly predictable, and hence it is a random variable. The order of these random variables then determines the judge's evaluation of the objects.

- Paired comparison models.
  This model is motivated by the connection between a ranking of objects and all pairwise comparisons of objects. Paired comparison models aim at combining models for paired comparisons to generate a probabilistic model for ranking data. Also, this model does not admit ties.

- Distance-based models.
  A distance function is useful in measuring the discrepancy between two rankings. The usual properties of a distance function between two rankings are reflexivity, positivity, and symmetry. Distance-based models can handle partial ranking, with some modifications in the distance measures.

- Multistage models.
  The class of multistage models includes ranking data models that postulate the ranking process can be decomposed into a sequence of independent stages.

> For a ranking of t objects, the ranking process can be decomposed into t-1 stages; at stage i, the ith object is selected.

Plackett-Luce model falls under the Order statistics model, which follows luce's axiom. Order statistics model handles big ranking data with many objects. Among the four probability models, order statistics models have the most extended history in the statistical literature. The basic idea behind the order statistics model is that a choice of user/respondent may fluctuate from one instant to another according to the perception of each object. The ordering of these random variables determines the user/respondent's ranking of the objects.

2.1 Use of ranking in SE

Various ranking methods are used in the Software Engineering area to address different problems. Most of the ranking methods accept the continuous metric and ranked list as input. Below, four examples of ranking in software engineering have been discussed. In these examples, statistical methods are used to generate the ranks.

Campos et al. [4] did an empirical evaluation to rank algorithms for test case generation. They ranked algorithms based on metrics such as overall coverage. They utilize bootstrap to calculate the uncertainty in these ranks. The bootstrap method is used to estimate statistics on a population by sampling a dataset with replacement. This research discusses the Friedman test, a nonparametric test to rank the algorithms for test case generation. Nonparametric means the test does not assume the data comes from a particular distribution. Code coverage of different classes using various algorithms was given as the input.

Altidor et al. [6] consider feature ranking to software engineering datasets. This study considered various classifiers like Naiıve Bayes (NB) and k-Nearest Neighbors (kNN) to classify the features on different datasets. Different performance metrics were used as ranking techniques like Overall Accuracy (OA), Geometric Mean (GM), Arithmetic Mean (AM). This study ranks various classifiers based on performance metrics using the area under the curve method.

Calvo and Santafe [33] did a study to compare and rank different algorithms based on the performance data of different algorithms. They were designed to simplify the statistical analysis of the results obtained in comparisons of algorithms in multiple problems. Many datasets were considered in this study. The algorithms were ranked based on the performance metrics of different algorithms. First, Iman Davenport's correction of Friedman's rank-sum test was conducted to test the hypothesis, i.e., whether all the algorithms perform equally or not. Based on the p-value, it is understandable whether the hypothesis is rejected or not. If at least one algorithm performs differently than the rest, then using the critical difference given by the Nemenyi test, the ranks of different algorithms were analyzed.

Garcia and Herrera [32] did a study on comparison of different classifiers based on the performance of classifiers over multiple data sets. Researchers use Bergmann-Hommel's procedure to find all elementary hypotheses that cannot be rejected. This study illustrates the study with the dataset of performance metrics of classifiers. Bergmann-Hommel's dynamic procedure first tests the hypothesis and then gives the ranks of classifiers.

2.2 Statistical analysis

Statistical analysis is the process of analyzing, summarizing, interpreting, and presenting the quantitative data collected. When data to be ranked is ranked list provided by the users, it requires to use of statistical methods to rank such data. Statistical methods are helpful when there is uncertainty on the data. During the comparison of objects, using statistics researcher gets conclusive evidence to rank the data. In this study, we discuss the two paradigms of statistical analysis, i.e., frequentist and Bayesian.

2.2.1 Frequentist statistics

Frequentist statistics test whether an event (hypothesis) occurs or not. In this type of inference, parameters and hypotheses are seen as unknown fixed quantities that we want to estimate. Frequentist statistics is about repeatability and gathering more data. Frequentist statistics is the standard approach for evaluating experimental results in online experiments where large amounts of data and many metrics and hypotheses are conducted simultaneously. Frequency estimates are usually based on the maximum likelihood estimator (MLE) or variations, such as the quasi- or penalized maximum likelihood estimator. Unfortunately, frequentist methods for null hypothesis testing have often been misused by scientists and practitioners. There are many shortcomings to this inference, and it is listed as follows:

- Lack of information regarding the null hypothesis [18].
- Misinterpretation of meaning of confidence intervals [15].
- Lack of transparency in the reporting of the statistical procedures [16].
- Misinterpretation of the actual meaning of the p-value [18].
- Lack of separation between the effect size and sample size in the p-value.

The ranking examples discussed in section 2.1 are frequentist methods and have their disadvantages. Campos et al. [4] used the Friedman test to rank in their study. However, this test may lack power if the sample size is small [22]. These non-parametric methods are geared toward hypothesis testing rather than estimation of effects. Altidor et al. [6] used the area under the curve methodology to rank the classifiers. But, this method is inconsistent and unreliable in choosing confidence

scales [25].

Calvo and Santafe [33] used the Friedman test and Nemenyi test to rank the algorithms based on the performances. The Nemenyi test is very conservative, has low power. In many cases, this test cannot control maximum Type I error. Garcia and Herrera [32] did a study on comparison of different classifiers using Bergmann-Hommel's dynamic procedure to rank the classifiers. Bergmann-Hommel's dynamic procedure may lack power if the sample size is small, and this method is the most difficult to understand and computationally expensive [32].
In this context, Bayesian statistics has gained attention from researchers as it naturally solves many of the problems listed above.

2.2.2 Bayesian Statistics

Bayesian analysis is a statistical model that answers research questions about unknown parameters using probability statements. Bayesian Data Analysis treats all unknown quantities in the statistical model as random variables, contrasting with the fixed constants from the frequentist approach. Depending on the chosen prior distribution and likelihood model, the posterior distribution is either available analytically or approximated using the Markov chain Monte Carlo (MCMC) methods.

The main idea behind Bayesian data analysis is to redistribute credibility among different possibilities [18]. In practice, we start with a prior explanation of the results before seeing any data and a model for generating the data. As we gather new data, our beliefs about the system are reallocated. The probability of candidate explanations that do not fit the data well is therefore reduced. In this updating process, we get a probability distribution of each possible explanation of the data. This procedure followed in bayesian data analysis allows us to obtain credible intervals [14].

The formula for Bayes' Theorem is stated as:

$$P(\theta|x) = \frac{P(x|\theta)P(\theta)}{P(x)}$$

Where x represents the data, θ the explanation (or hypothesis), P(θ|x) is the conditional probability of the hypothesis given the observed data. Below are common names for the factors in the Bayes theorem:

- P(x|θ) is the likelihood of the data x under the hypothesis θ.
- P(θ) denotes the prior distribution of the parameters.
- P(θ|x) is called the posterior. The posterior represents the probability distribution of each parameter estimate (our hypothesis h) given our observed data.

- P(x) is called the marginal likelihood, and it is a constant that is often impossible to compute analytically.

As I discussed, there are many shortcomings in the frequentist statistics. These include lack of flexibility, unintuitive results, and difficulty in interpretation limit their effectiveness in dealing with the diverse data available for empirical analysis of software engineering practice. Furia et al. [14] present Bayesian data analysis techniques and their benefits. This study stresses the role of Bayesian statistical techniques in empirical software engineering research and practice.

Plenty of Bayesian researchers have done their study and argument on the advantages of Bayesian statistics. In chapter 3, I discuss the Bayesian Plackett-Luce model in detail.

3

Bayesian Plackett-Luce Model

The Plackett-Luce model is a statistical model for ranking that estimates the strength variable of each item during comparison [1]. The Plackett-Luce model collects some features that make it suitable to model rankings of artifacts. The generalization of the model accommodates both ties and partial rankings. The output of the model is an estimated worth for each item that appears in the rankings. The parameters are generally presented on the log scale for inference [1]. A log-linear model is a mathematical model that takes a function whose logarithm equals a linear combination of the model's parameters.

For example, let us consider a car racing game with four cars. Suppose a probability distribution P represents the ability to win a race. The rank of these cars can be understood as a generative procedure. The result can be computed as follows: Let us assume Car 2 as the champion car with the probability of p2 among four cars, i.e., p2/(p1+p2+p3+p4). First, chooses a top-ranked item from all the items, then select a second-ranked item from the remaining items, and so on. If car1 becomes the runner-up, the probability p1 has to be normalized among the remaining three cars, which leads to p1/ (p1 + p3 + p4). So the probability of the rank2, 1 is their product. It is easy to observe that the most likely rank is that all cars are ranked by their winning chance.

When modeling rankings of size k, the Plackett-Luce model has k parameters, one per artifact. We refer to these parameters as weights, w_i. If we represent rankings as $\sigma = (\sigma_1, ..., \sigma_k)$ where σ_i = j means that the j-th algorithm is ranked in the i-th position, the probability of a given ranking σ under the Plackett-Luce model is:

$$P(\sigma|w) = \prod_{i=1}^{n} \frac{w_{\sigma_i}}{\sum_{j=i}^{n} w_{\sigma_j}}$$

Most research works use the frequentist version of the Plackett-Luce model. There are a couple of pitfalls to the frequentist Plackett-Luce model [1]. First, when the variance-covariance matrix is large, estimating the model parameters' standard errors may take a long time or be impossible due to memory constraints. Second, for data with higher-order ties, this method does not explicitly model tied events. A Bayesian version can be used to overcome the drawbacks. They can provide better results that are simultaneously robust and faster.

Bayesian statistics provide important advantages that make researchers choose during modeling.

- Ability to specify priors. Priors include knowledge that we have about data collection. Researchers can incorporate past information about a parameter and form a prior distribution for future analysis.

- It provides a probabilistic interpretation. That is, it is not based on repeated sampling.

- During inference, it offers a distribution of values of uncertainties. For example, In linear regression identifies a family of possible slopes, i.e., distribution of slopes.

- It provides a convenient setting for a wide range of models. MCMC, along with other numerical methods, makes computations tractable for virtually all parametric models.

Bayesian statistics simultaneously considers all possible distributions of a particular family and assigns a probability to each distribution. Before observing any data, the probability of the parameters is represented by the prior distribution. This distribution represents our prior belief about the actual parameters of the distribution from where the data comes. When we have access to actual data, and we have to update this belief accordingly. This update is done using Bayes' rule.

As in any Bayesian model, the three distributions are mentioned as follows: the likelihood function, the prior distribution, and the posterior distribution. In the model, the posterior is calculated as the product of likelihood function and prior distribution.

The prior distribution of the weights is modeled with the Dirichlet distribution, which is the generalization of the Beta distribution. The beta distribution is a family of continuous probability distributions defined on the interval [0, 1] denoted by α and β, which appear as exponents of the random variable [9]. The generalization to multiple beta variables is called a Dirichlet distribution. The Dirichlet distribution is used as a prior distribution in Bayesian statistics.

Dirichlet distribution is the conjugate prior to an important probability distribution, i.e., the categorical and multinomial distributions. In Bayesian probability theory, if the posterior distribution and the prior distribution are from the same probability distribution family, then prior is the conjugate prior for the likelihood function. During the modeling phase, we already know the posterior will also be a Dirichlet distribution. Therefore, after carrying out experiments, we can compute the posterior simply by adding the number of acceptances and rejections to the existing parameters α, β respectively, instead of multiplying the likelihood with the prior distribution.

Briefly, the Dirichlet distribution models probability distributions over real-valued as R = $\sigma(1), ..., \sigma(N)$, the posterior distribution of the weights can be computed as:

$$P(w|R) = \frac{P(R|w)P(w)}{p(R)}$$

Markov chain Monte Carlo method is available to approximate the posterior distribution of a parameter of interest by random sampling. MCMC provides ways to sample from a probability distribution and is mainly needed to sample from a posterior distribution. MCMC allows the algorithm to narrow in on the quantity approximated from the distribution, even with many random variables. When applied to the ranking data derived from the comparison, obtain an approximation of the posterior distribution of weights that can be used to answer different questions. The Bayesian inference approach allows us to make more precise statements.

The below model represents the Bayesian Plackett-Luce model:

$$P(y) = \prod_{i=1}^{N} \frac{w_i * Rating}{\sum_{j=i}^{N} w_j * Rating} \qquad \text{(Likelihood)}$$

$$Ratings \sim Dirichlet([1, 1, 1, 1, 1, ..., N]) \qquad \text{(Ratings Prior)}$$

In the model we have the following notation:

- P(y) is the posterior value of the artifact.
- w_i indicates weight of each artifact.
- The prior for Ratings is assumed to be distributed Dirichlet value as one.

The inference process is as follows. First, the ranked list given by the users or continuous performance metric is given as the input to the model. These inputs are used to produce a sample from the posterior distribution of weights. Once the posterior distribution of weights is defined, the model represents several aspects of the distribution of rankings. We analyze the weights using the mean factor to know the ranks of the items. Also, the BPL model provides the density plot, which gives additional information about the data.

4
Research Method

This chapter presents the objectives of this thesis, the research questions, and the methodology used in the study. Also, it introduces collected datasets for the analyses of the developed Bayesian Plackett-Luce model.

4.1 Research Objective

Research objectives narrate neatly what the research is trying to achieve. They summarize the accomplishments a researcher wishes to achieve through the project.

The study's objective is to **examine** the applicability of the Bayesian Plackett-Luce model in SE by creating a tool (an R package) and analyzing the results. The developed tool is helpful to compare the ranking results with the existing research to show the suitability of the Bayesian PL package in SE.

4.2 Research questions

A research question is a question that a research project sets out to answer. Choosing a research question is an essential element of the research. Below, presented the research questions of the study.

- (RQ1): How suitable is the Bayesian Plackett-Luce model for software engineering research?

  I use the dataset of other research or datasets from digital archives to reanalyze the developed tool. During reanalyses, I compare the Bayesian Plackett-Luce model results with the results of other ranking methods. The ranking list produced by the Bayesian Plackett-Luce model should be the same as the ranking lists given by other ranking methods. Also, this study considers different kinds of datasets to analyze the applicability of the model in various scenarios. These results help to analyze whether the Bayesian Plackett-Luce model for software engineering research is suitable or not. Further, request three SE researchers to evaluate the relevance of the Bayesian Plackett-Luce model for Software Engineering. SE researchers evaluate the model by analyzing the tool with the datasets that have been used in our reanalysis. Discuss with these SE researchers to know whether they see the value in the tool. The demonstration of analyses given to SE researchers helps them decide the suitability of the

Bayesian Plackett-Luce model for software engineering research.

- (RQ2): What are the different pieces of information that the BPL provides that make someone consider using the BPL instead of others in SE?

  After developing the tool, the Bayesian Plackett-Luce model is analyzed with the SE dataset used in other SE research works. Compare the ranking results obtained from the Bayesian Plackett-Luce model with other models. Also, the study compares the information provided by the Bayesian version of the Plackett-Luce model against other models. The additional information given by the density plot is analyzed in the Bayesian Plackett-Luce model. Also, analyze the advantages of the BPL against the other ranking models.

4.3 Design Science Methodology

This project is conducted according to the design science research methodology [30]. Design science is the design and investigation of artifacts in software engineering [8]. The two defining characteristics of design research are its interest in field problems and its solution focus which solves field problems with interventions or systems [10]. There are five phases in the methodology, i.e., awareness of the problem, suggestion, development, evaluation, and conclusion. The first three phases can be categorized into a building step. The last two phases can be categorized into the evaluation step. Figure 4.1 shows a consolidated version of the five phases. It is followed by a description of how each stage in different iterations has been observed in our study.

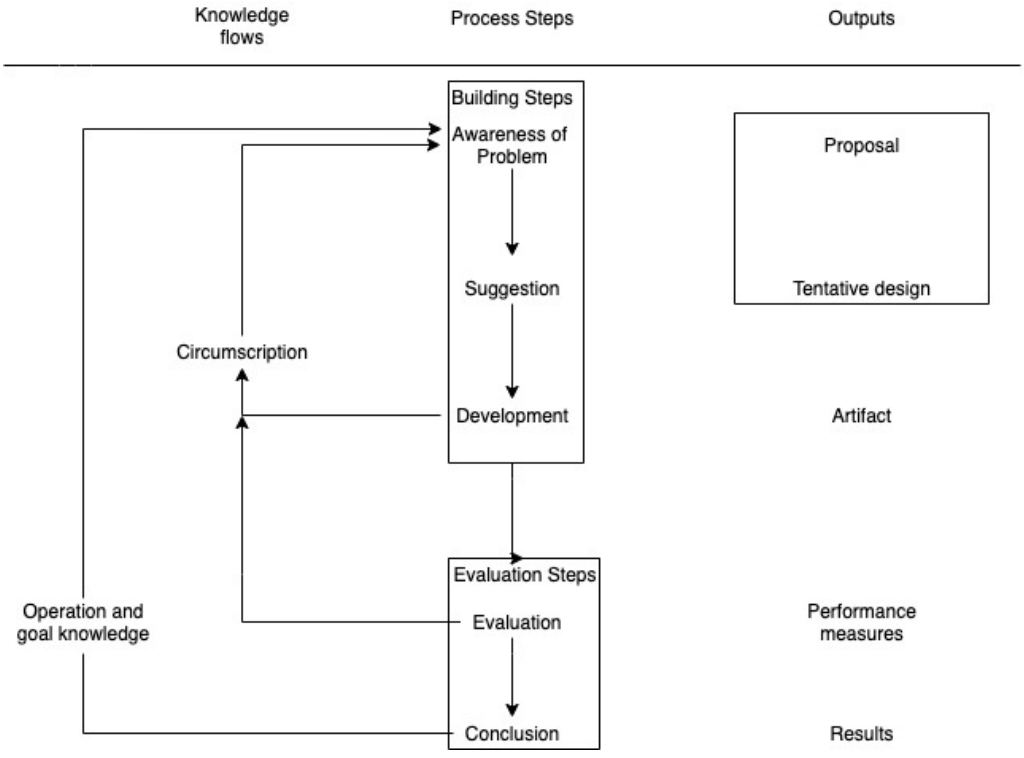

Figure 4.1: High Level Design Science Research Process

Design Science research methodology is suitable for this research because examining the applicability of the Bayesian Plackett-Luce model for SE problems is intuitively iterative. This approach helps to understand the problem first, then design and develop the solution. After developing a solution, this approach also accommodates the evaluation of the intention. Another advantage of the design science approach is that every research iteration does not always have to start from the first step (i.e., awareness of the problem).

4.3.1 First iteration of Design Science

The first iteration is conducted in the following sequence. Initially, the objective of the study is analyzed. An investigative review of related studies and topics is conducted, followed by the designing and creating Bayesian Plackett-Luce model.

In this iteration, the developed model is validated with the same dataset used in the frequentist Plackett-Luce model analysis. The results of the Bayesian Plackett-Luce model are compared with the frequentist Plackett-Luce model as a part of validation.

4.3.1.1 Awareness of the problem

This phase of the design science started with identifying the problem and scope. This phase of the design science helps to understand the objective of this study. In this phase, I tried understanding the primary intention of the study, i.e., to analyze the Bayesian Plackett-Luce model for ranking in Software Engineering is relevant or not.

4.3.1.2 Suggestion

This phase of the design science was helpful to understand the tentative idea that might address the problem. This phase aims to determine key concepts needed to solve the problem and produce a tentative package design. After designing the proposal, I started to go through the related research works to know the key concepts required for developing the solution. Tried understanding how different data were ranked using the various models of the statistics. Later, I got examples of many Software Engineering data ranking instances in various research works. The study of statistical inference also helped to design the solution and analyze the study's objective. In section 2, related works to our research have been explained in detail. Also, I learned the vocabulary of the basic concepts needed for developing the Bayesian Plackett-Luce model. I decided to implement the model using R and Stan to design the Bayesian model of the Plackett-Luce model.

4.3.1.3 Development

This phase is where the outcome of the suggestion phase is enriched to develop an artifact that addresses the problem identified in the first phase. In the development phase, the model's features were divided into tasks. The underlying techniques for

executing these tasks are built. I chose the R language to develop the Bayesian Plackett-Luce model for ranking in SE. I designed the model by selecting the appropriate parameters and priors. The designed model fitted using the CmdStanR. CmdStanR is a lightweight interface to the Stan probabilistic programming language for R users. In section 5, the implementation of the model is discussed in detail.

4.3.1.4 Evaluation

The main aim of this phase is to check how well does the model work. I evaluated the BPL model using reanalyses. Initially, I analyzed the Bayesian Plackett-Luce model with the dataset that has been used in the analysis of the frequentist Plackett-Luce model. The rank sequence of both versions of the Plackett-Luce model should match. To verify our Bayesian model, I collected the same dataset to perform analysis.

Dataset

A dataset is a collection of data. It consists of two components, which are rows and columns. Additionally, a vital feature of a data set is that it is organized so that each row contains one observation.

Here we compare the Bayesian Plackett-Luce model with the frequentist Plackett-Luce model to check the validity of our developed package of the Bayesian Plackett-Luce model. I have collected the dataset that has been used in the study of the frequentist Plackett-Luce model[1]. Here the goal is to compare the results of both Bayesian and frequentist Plackett-Luce models. Both the versions of the Plackett-Luce model should give the same sequence of ranking. Below, we can see the sample of that dataset.

1.Mean Girls	2.Beverly Hills	3.Mummy Returns	4. Mission:Impossible
2	1	4	3
1	2	4	3
2	1	3	4
1	2	3	4
2	4	1	3

Table 4.1: Sample Netflix users movie ratings data from PrefLib, (Bennett and Lanning 2007)

Each row corresponds to a unique order of the four movies in this dataset. The number of Netflix users assigned this order is given in the first column, followed by the four movies in order of preference. For example, the first user ranked Beverly Hills as first, Mean girls as second, Mission Impossible as third, and Mummy returns as fourth. The evaluation results are discussed in detail in chapter 6.

4.3.2 Second Iteration

In the second iteration, I continued with the analysis of the Bayesian Plackett-Luce model. Analyses are carried out to examine the applicability of the BPL by analyzing the model with the three suitable SE datasets.

4.3.2.1 Awareness of Problem

The focus of the design cycle continues from the previous cycle with the investigation of the applicability of the Bayesian Plackett-Luce model for SE. Along with this, focus on finding the advantages of using the Bayesian Plackett Luce model against other ranking methods on SE problems.

4.3.2.2 Suggestion

This iteration phase helped design the density plot that clearly shows the advantages of the Bayesian Plackett-Luce model. This plot is also helpful during the comparison between the Bayesian Plackett-Luce model and nonparametric models. A literature review also helped to know the disadvantages of nonparametric methods used for ranking in SE.

4.3.2.3 Development

I developed a function to plot the density plot according to the suggestion phase in the development phase. The underlying techniques for executing these tasks are built. In chapter 5, the usage section shows the code of the density plot function.

4.3.2.4 Evaluation

In this iteration, evaluated the BPL model using reanalyses with three Software Engineering datasets. Picked three SE datasets for the reanalysis based on the criteria that these SE datasets have to be either from research works or digital archives. Also, these selected datasets were of different kinds to examine the applicability of the Bayesian Plackett-Luce model for different scenarios. Further, I analyzed the additional information about the data provided by the Bayesian Plackett-Luce model.

Dataset1

To reanalyze the model, I collected the dataset from García and Herrera [21]. This dataset contains the performance of supervised classification algorithms in a set of 30 datasets. This specific dataset has been used in two other research works to rank the algorithms based on the performance data. Garcia and Herrera [32] considers this dataset to rank the classification algorithms using Bergmann-Hommel's procedure to rank the data. Calvo and Santafé [33] uses the same dataset but different methods to rank the classification algorithms. They used Iman and Davenport omnibus test and the Nemenyi test to show the difference between the algorithms. I compared the reanalysis results with other ranking methods used to analyze this

specific dataset.

The goal of the reanalysis was to rank the classification algorithms and determining which is the better algorithm among others. Table 4.2 is the head of the dataset.

	C4.5	k-NN(k=1)	NaiveBayes	Kernal	CN2
Abalone*	0.219	0.202	0.249	0.165	0.261
Adult*	0.803	0.750	0.813	0.692	0.798
Australian	0.859	0.814	0.845	0.542	0.816
Autos	0.809	0.774	0.673	0.275	0.785
Balance	0.768	0.790	0.727	0.872	0.706

Table 4.2: Sample data to measure the performance of supervised classification algorithms (Gracia and Herrera)

Each row corresponds to the performance of the five different supervised classification algorithms on specific datasets. In the first row, these five different supervised algorithms were applied to the Abalone dataset. The classification performance of these algorithms on Abalone is mentioned in the first row. Same way, classification performance is measured on 30 different datasets. So this dataset has 30 rows.

Dataset2

The goal of reanalysis on the following collected dataset was to rank the most popular programming languages [34]. This dataset has the popularity data of 28 different programming languages from July 2004 to May 2021. This corresponding dataset was collected from the Kaggle. The popularity metric was converted to a ranked list and given as input to the model. This reanalysis using this dataset helps to examine the applicability of the model in a different scenario. Table 4.3 is the head of this dataset and does not include all the columns of the dataset.

Date	Abap	Ada	C/C++	C#	Cobol	Dart	Delphi	Go	Groovy
July 2004	0.3399	0.36	10.08	4.71	0.43	0	2.82	0	0.82
August 2004	0.36	0.36	9.81	4.99	0.4599	0	2.67	0	0.069
September 2004	0.41	0.41	9.62	5.06	0.51	0	2.65	0	0.08
October 2004	0.4	0.38	9.5	5.31	0.53	0	2.77	0	0.09
November 2004	0.38	0.38	9.52	5.24	0.549	0	2.76	0	0.069

Table 4.3: Sample data of most popular programming languages

Each row in the above dataset represented in table 4.3 has programming language popularity values in percentage form out of 100 % along with the date. This dataset has 204 rows and 29 columns. The first row represents the popularity of different programming languages in percentage form out of 100% as of July 2004. The same

way all other rows have the popularity of different programming languages measured on various dates.

Dataset 3

The goal of reanalysis on the below-collected dataset was to rank the most popular databases in the current year [35]. This corresponding dataset was collected from the Stack Overflow Annual Developer Survey datasets for 2020. The conducted survey had 64621 respondents to answer the databases they have worked. Fourteen different databases were named in the dataset. Below table 4.4 is the head of this dataset.

This reanalysis considers only the data of the databases with which users were currently working. This reanalysis ranks databases based on the user's current usage. This dataset has a ranked list and was given as the input to the BPL model. Even though this dataset deals with the databases, this dataset is relevant to SE because the reanalysis results would be helpful to database companies to understand the trend. Since most SE research work or SE company projects deal with the database, reanalysis results would help them pick suitable databases for their respective projects. These were the reasons this dataset was picked for reanalysis to examine the suitability of the BPL model.

Each row is the response of each user. In each row, the databases with value 1 represent the databases where the respondent has worked and other databases as 2.

DynamoDB	PostgreSQL	Elasticsearch	Firebase	MongoDB	SQLite	MySQL
2	2	1	2	2	2	2
2	1	2	2	2	1	1
2	2	2	2	2	2	2
2	2	2	2	2	2	1
2	1	2	2	1	1	1
2	1	1	2	2	1	1

Table 4.4: Sample dataset of user has worked with databases.

Each row in the dataset represented in table 4.4 has an answer given by the respondents. Each row contains the names of all databases. Some of the respondents have not worked with any databases. The first row represents the answer of one respondent. The first respondent has worked with Oracle. In the same way, all other rows have the answers of other respondents to the survey.

4.3.3 Third Iteration

In the third iteration, further continued with the analysis of the Bayesian Plackett-Luce model. This iteration intends to evaluate the study's objective further by taking the feedback from the Software Engineering researchers.

4.3.3.1 Awareness of the Problem

The focus of the design cycle continues from the previous cycle with the investigation of the applicability of the Bayesian Plackett-Luce model for SE. The only difference was that the intention of the study was evaluated with the SE researchers.

4.3.3.2 Suggestion

This phase of the iteration helped to design the evaluation process with the SE researchers. The SE researchers' opinions helped to evaluate the intention of the study. However, there should be some process to do the evaluation interview with the SE researchers. This phase helped to design that process of evaluation. A literature review of thematic analysis helped to know the process of analyzing the opinions of SE researchers.

4.3.3.3 Development

In this iteration, did not change or develop any features in the package. This third iteration focused more on getting feedback about the developed package.

4.3.3.4 Evaluation

In this iteration, I took feedback from three SE researchers by demonstrating the analyses. I considered SE researchers for evaluation based on the criteria of having deep knowledge in SE and statistics. Initially, I approached the SE researchers whose literature was helpful during development and contacted them through email. I considered the researchers only from Sweden because contacting and convincing them was a bit easy. The researchers' interviews intended to know whether they see value in the Bayesian Plackett-Luce model in SE or not.

Interviews

Interviewing is a fundamental methodology for both quantitative and qualitative evaluation. However, the study conducted a qualitative method of assessment. It is critical to get feedback from scholars to know the suitability of the developed package in SE. In evaluation process requested three SE researchers who had done their research on ranking or related areas. Explained the evaluation process to researchers and tried convincing SE researchers who had done their study on ranking or related fields to be part of this evaluation process. These three SE researchers evaluated the model by analyzing the tool.

Presented the context of the thesis and discussed the Bayesian Plackett-Luce model for ranking during the evaluation process with SE researchers. The presentation can be found in Appendix A.1. I considered the datasets that have been used in previous reanalyses of the study to demonstrate the BPL model to SE researchers. Conducted a semi-structured interview using video calls with these SE researchers. This interview helped to know SE researchers' opinions on the model, presentation of the results, and thoughts on future improvements. I asked the following questions

with the SE researchers during a semi-structured interview.

1) How important to use ranking in SE after going through the examples taken in the study?
2) Knowing the advantages of Bayesian inference and the Plackett-Luce model from many research works, Is this BPL model relevant to SE?
3) How relevant do you think the datasets taken in the study are to validate the model's applicability for SE use cases?
4) Were there any discrepancies in results inferred from the research? Do the inferred results align well with ideal/expected results?
5) How useful do you find a tool like this? What other real-world use cases of a tool like Bayesian Plackett-Luce serve?
6) Why it is important to use statistics while ranking the data in SE?

During an evaluation with SE researchers, I got an opinion on whether they see the value in the tool or not. The analyses results helped SE researchers to give an opinion on the suitability of the Bayesian Plackett-Luce model for software engineering research (RQ1). The interview was recorded and converted that into a transcript. This interview transcript helped with the thematic analysis. Based on the feedback given by the SE researchers and thematic analysis, I was able to conclude whether the Bayesian Plackett-Luce model for ranking in SE is applicable or not.

Thematic analysis

Thematic analysis is a method of analyzing qualitative data. It is usually applied to a set of texts, such as interview transcripts. I closely examine the text to repeatedly find the familiar pattern, ideas, or topics of that meaning. There are different approaches to thematic analysis. However, I followed an inductive approach. An inductive approach involves allowing the data to determine themes [31]. In the inductive approach, coding occurs without trying to fit the data into a pre-existing theory.

In the thematic analysis, I followed four steps to analyze the interview of scholars. As a first step, I familiarised the interview transcript by reading it several times and took a note while reading the interview text. Next step, I did open coding by identifying the meaning piece in the interview data. Coding means highlighting sections of our text and coming up with shorthand labels to describe the content. Then merged the codes into multiple categories. Finally, combined the categories into themes.

I took the interview text of three experts in the ranking area and did the thematic analysis of the text separately. I analyzed the themes and got to know the positive and negative opinions given by the scholars during the evaluation process. The answers given by the three researchers were straightforward and wholly connected to the study topic. I did the thematic analysis alone and followed the four-step systematic procedure to analyze the interview data to get reliable results.

Concluded the suitability of the Bayesian Plackett-Luce model for ranking in Software Engineering problems after studying the themes of all the scholar's interview transcripts.

5

Implementation of the Package

After understanding the problem and analyzing it in the first phase of design science, we started to design a tentative solution for the problem. I chose the R language to develop the Bayesian Plackett-Luce model for ranking in SE. Because R is an open-source programming language, and it is mainly known as the language of statistics. I designed the model to formulate the mathematical formula of the Plackett-Luce model. R language has many packages for statistics and data analytics. Development of the package in R will help researchers to use the package in Bayesian data analysis.

I am using the Rstudio environment to work on the R language. RStudio is an Integrated Development Environment for R. It is straightforward and convenient to download the RStudio from their official website. It has a very minimum requirement to install on our system. While working on the package, I had a 1.4.1717 version of R studio. The package is an assembly of files and information about those files.

After the design of the solution, I started with the development of the package. In the process of development of the package, I started with downloading the devtools package. In recent years, R package development has become substantially easier by introducing a package by Hadley Wickham called devtools. The main goal of devtools is to make package development more manageable by providing R functions that simplify common tasks. As the package name suggests, this includes various functions that facilitate software development in R. It is supported with the functions to load and document the package.

There are different packages supported by R to make the developer job easy. The packages which were used in our implementation process are discussed here.

- roxygen2
  Documentation is the critical aspect of the code, and it is useful for the developers in the future. The roxygen2 package provides the standard way of documenting the code and documents the code as easily as possible.

- dplyr
  dplyr is another useful package that provides a consistent set of verbs that help to solve the most common data manipulation challenges. For example: summarise() function reduces multiple values down to a single value.

- tidyr
  tidyr is a package useful to us that makes it easy to tidy our data. Tidy data is data that's easy to work with, visualize, and model.

- stringr
  Stringr is a package that is not glamorous. However, they do play a significant role in data cleaning and preparation tasks. Therefore, this package is a fundamental one to work with strings.

- posterior
  The posterior R package is intended to provide valuable tools for fitting Bayesian models or working with output from Bayesian models. The primary goal of the posterior package is to convert between many different valuable formats of draws from the posterior.

- testthat
  testthat package tries to make testing as straightforward as possible. This package provides functions that make it easy to describe what we expect a function to do, including catching warnings and errors. It also displays test progress visually, showing a pass, fail, or error for every expectation.

- bayesplot
  The bayesplot is an R package providing an extensive library of plotting functions after fitting Bayesian models. bayesplot offers a variety of plots of posterior draws and graphical posterior predictive checking.

- ggplot2
  ggplot2 is a package that helps visualize the data, graphics for communicating a meaning of posterior data. We provide data and tell 'ggplot2' how to map variables and what graphical primitives to use. Then the ggplot2 package will give the appropriate results.

Initially, I designed the model by selecting the appropriate parameters and priors. The designed model fitted using the CmdStanR. CmdStanR is a lightweight interface to the Stan probabilistic programming language for R users. I Utilized a Bayesian probabilistic programming language called Stan. Stan is a probabilistic programming language for specifying statistical models. It is open-source software that offers a straightforward approach to implement Bayesian models that can fit data structures in R. There are significant advantages for CmdStanR. This package is compatible with the latest versions of Stan, creates less memory overhead and more permissive licenses than other stan packages like Rstan.

To run Stan in R, we need to have a suitable C++ toolchain. The C++ toolchain consists of a modern C++ compiler and the GNU-Make utility. Later, we installed the CmdStanR from GitHub. CmdStan translates Stan programs to C++ using the Stan compiler program, which is included in the CmdStan release bin directory as

program stanc. This package provides high flexibility with only a few limitations.

I selected Dirichlet distribution as an optimal choice as a prior in Bayesian statistics. It is discussed in section 4. The Stan performs fitting a Stan model, translates Stan program to C++, and creates compiled executable. $sample() method runs stan's MCMC algorithm on CmdStanModel objects.

There are many functionalities of this package that will benefit researchers during the research activity. This developed package creates CmdStanMCMC objects, which have many associated methods for given data. The posterior summary() method summarizes all the variables after the compilation of the model. The $draws() method extracts the posterior draws as a 3-D array. With the extracted data, the rank of the given data can be finalized based on the mean, median, or standard deviation. Visual representation of the outcome will provide more clarity on the ranking of the data.

The developed package bayesPL is available in GitHub, i.e.,
https://github.com/vallisha/bayesPL

The package can be downloaded from the above link.

Usage

Here I consider a dataset of ranks of browsers given by users as input to the Bayesian Plackett-Luce model for ranking. This specific section illustrates the working of the BPL package. This section tries to explain the meaning of code lines used in the BPL package and explains how to use the package.

To use this package, we need to load the package into Rstudio. Then, we have to load the devtools library. load_all() command loads all the required packages to compile the bayesPL package.

```
>devtools::load_all()
```

Dataset is loaded as the table with the total number of rows and header. Later it data is passed to bpl to fit the Plackett-Luce model.

```
>data1 <- read.table("data/data1.csv", header=TRUE, sep=",", nrows=20)
>print(head(data1))
```

Table 5.1 is a sample set of data that has six rows only. In this dataset, all four browsers are ranked by 30 users. Each user's response is represented in rows.

Chrome	Safari	Opera	Mozilla
1	2	3	4
1	4	3	4
1	3	2	4
1	3	4	2
1	4	2	3
1	2	3	4

Table 5.1: Sample dataset of rank of browsers given by users.

>res <- bpl(data1, min=FALSE, nsim=1000, nchains=3)

Understanding the generated results is critical to know the meaning of the results. Without understanding the results, it is hard to conclude the research question. The below code snippet shows how we compute the rank based on the mean of the posterior. Initially, the posterior value is converted to a matrix. The apply function takes a matrix as an input and gives output in the list.

```
#convert to matrix
posterior <- posterior::as_draws_matrix(fit$draws("ratings"))
colnames(posterior) <- colnames(ranking.matrix)
#apply() takes matrix as an input and gives output in vector, list or array.
posterior.calculator <- t(apply(posterior, MARGIN=1,
FUN=function(i) {
return (rank(-i))
}
))

#Computation of rank based on the mean of the posterior.
mean.rank <- colMeans(posterior.calculator)
```

Chrome	Safari	Opera	Mozilla
0.354	0.218	0.205	0.221

Table 5.2: Posterior mean values of the browsers.

Table 5.2 represents the ranks based on posterior mean values. In table 5.2, if the posterior mean value is high, that browser is ranked first. The next highest value is considered as the second rank. In the same way, the lowest value movie is ranked as last.

The ranking sequence of browsers according to the Bayesian Plackett-Luce model as Chrome is ranked as first, Mozilla is ranked as second, Safari is ranked as third,

and Opera is ranked as fourth.

```
plot_graph <- function(posterior, columns) {
orange_scheme <- c("#ffebcc", "#ffcc80",
"#ffad33", "#e68a00",
"#995c00", "#663d00")
color_scheme_set(orange_scheme)
color_scheme_view()
transformation_function <- function(x) min(colMeans(posterior))
+ max(colMeans(posterior)) - x
mcmc_areas(posterior, pars=c(columns),
transformations=transformation_function) + scale_y_discrete(labels=c(columns))
}
```

For models fit using MCMC, we can compute posterior uncertainty intervals in various ways. To show the uncertainty intervals as shaded areas under the estimated posterior density curves, we can use the mcmc_areas function. The above code snippet is used to compute the density plot of the model.

```
>plot_graph(res, colnames(res))
```

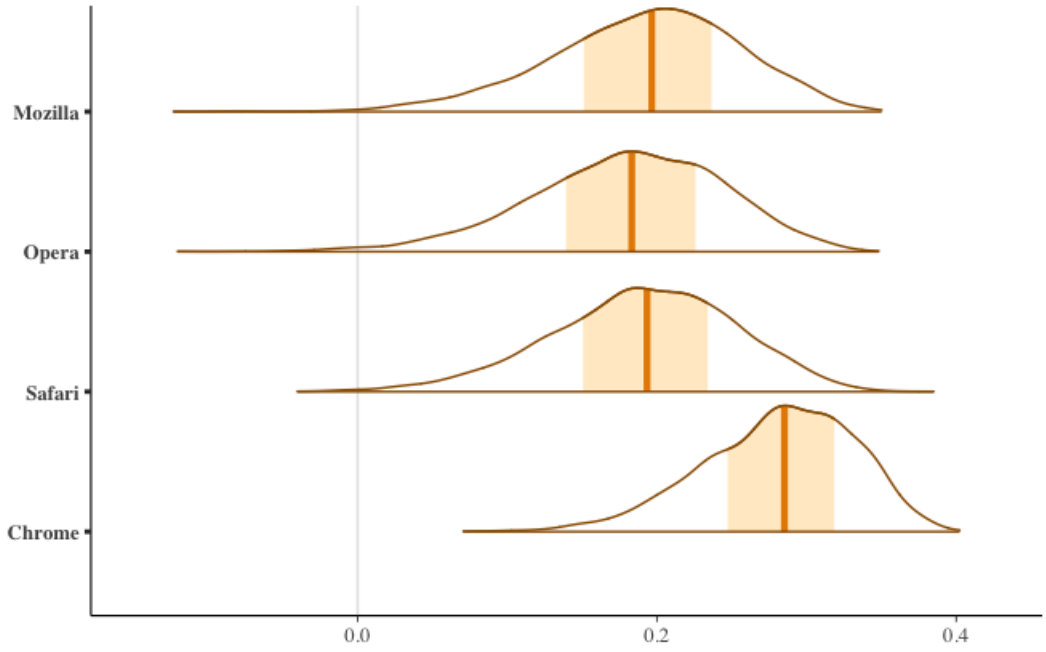

Figure 5.1: Density plot computed from posterior draws.

Figure 5.1 shows the uncertainty intervals of each browser in the plot. This plot is a great advantage because it gives the distribution of values of movies, not just a single value. By looking at the plot, it is easy to recognize how users have ranked the browsers. I have discussed the density plot in chapter 6.2.1.

6
Results

In this chapter, the results of all the iterations of design science are presented. Initially, the dataset used in the frequentist Plackett-Luce model was considered to reanalyze the Bayesian Plackett-Luce model. In the next stage, three SE datasets were considered to analyze the model and intention of the study. Later part, evaluation interviews were conducted with three SE researchers to get their feedback on the model.

6.1 First Iteration of Design Science

During this iteration, the reanalysis aimed to check the validity of the Bayesian Plackett-Luce model I developed. To do this, I took the dataset that has been used in the analysis of the frequentist Plackett-Luce model. The ranking pattern provided by both versions of the Plackett-Luce model should be the same because the mathematical model of the Plackett-Luce model remains the same. Here I took the dataset provided by Netflix about movie rankings given by the users. This dataset also has been used in the analysis of the frequentist Plackett-Luce model.

Mean Girls	Beverly Hills	Mummy Returns	Mission:Impossible
0.230	0.45	0.168	0.14

Table 6.1: Latent strength value of movies in frequentist Plackett-Luce model.

The ranking sequence of movies according to the frequentist Plackett-Luce model are Beverly Hills is ranked as first, Mean Girls is ranked as second, Mummy Returns is ranked as third, and Mission: Impossible is ranked as fourth.

Beverly Hills > Mean Girls > Mummy Returns > Mission: Impossible

Mean Girls	Beverly Hills	Mummy Returns	Mission:Impossible
0.299	0.400	0.2001	0.100

Table 6.2: Latent strength variable of movies in Bayesian Plackett-Luce model.

The ranking sequence of movies according to the Bayesian Plackett-Luce model are Beverly Hills is ranked as first, Mean Girls is ranked as second, Mummy Returns is

ranked as third, and Mission: Impossible is ranked as fourth.

Beverly Hills > Mean Girls > Mummy Returns > Mission: Impossible

In both frequentist and Bayesian versions of the Plackett-Luce model, the rank sequence remains the same. This result indicates that our Bayesian Plackett-Luce model works as expected theoretically. This analysis assures the validity of our model. Latent strength variable values of movies are different between the two versions. Also, the Bayesian Plackett-Luce model provides a density plot, which helps to understand the results.

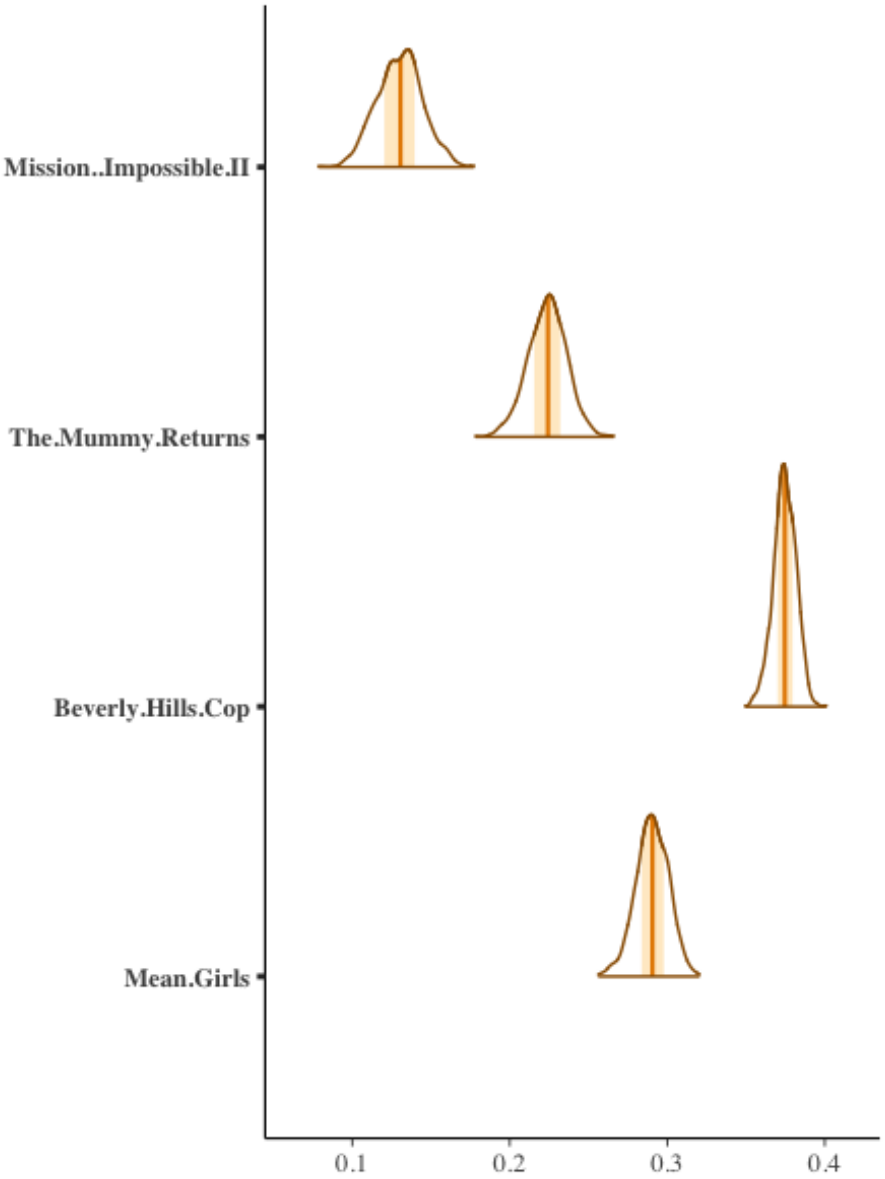

Figure 6.1: Density plot of dataset 1 parameters computed from posterior draws.

The Bayesian version of the Plackett-Luce model also gives the density plot of movies computed from the posterior draws. Figure 6.1 shows how the uncertainty intervals

of each movie are shown in the plot. This plot is a great advantage because it offers the distribution of uncertainty values. Narrow distribution in the plot indicates more consistency in movie ratings. Wider distribution plots indicate less consistency in movie ratings. By looking at the plot, it is easy to recognize how users have ranked the movies. This plot helps to identify which movie has the highest value and is ranked first by the users. Beverly Hills Cop is ranked first by the users. This interpretation of results is another advantage of the Bayesian Plackett-Luce model.

6.2 Second Iteration of Design Science

To know the suitability of the developed Bayesian Plackett-Luce package, I started analyzing the package with the different kinds of SE datasets. I have conducted three reanalyses to study our model. Reanalyses helped to study the suitability of the Bayesian Plackett-Luce model for Software Engineering problems. These reanalyses are explained below in detail.

6.2.1 Dataset 1

The sample of the dataset is shown in table 4.2. I used the dataset presented in that paper [32], which is available in GitHub [21]. This dataset has the performance of 5 supervised classification algorithms which are applied on 30 datasets. The goal of reanalysis was to check which classification algorithms outperform others, i.e., ranking these classification algorithms based on the performance data.

C4.5	k-NN(k=1)	Naive Bayes	Kernel	CN2
1.2995	3.9010	1.7385	5.0	3.061

Table 6.3: posterior mean value of 5 algorithms in the Bayesian Plackett-Luce model

Table 6.3 has values of the posterior data of classification algorithms. Here posterior value was not constrained to 1. According to Garcia and Herrera [32], the algorithm which has the least posterior mean value is the best classification algorithm. The algorithm with the highest performance value is the worst classification algorithm.

C4.5 algorithm is ranked first, Naive Bayes is ranked second, CN2 is ranked third, K-NN(k=1) is ranked fourth, and Kernel is ranked fifth.

C4.5 > Naive Bayes > CN2 > k-NN(k=1) > Kernel

6.2.1.1 Comparison of BPL results with Nonparametric methods

Garcia and Herrera [32] considers above-mentioned dataset to rank the classification algorithms. This study uses **Bergmann-Hommel's procedure** to rank the

data which is the most powerful post-hoc test [32].

C4.5	k-NN(k=1)	Naive Bayes	Kernel	CN2
2.100	3.250	2.2	4.333	3.117

Table 6.4: Computation of the rankings for the five algorithms using Bergmann-Hommel's procedure.

Table 6.4 has average values of the performances of the algorithms. Bergmann-Hommel's dynamic procedure allows ranking the classifiers based on the performance metric. C4.5 algorithm is ranked first, Naive Bayes is ranked second, CN2 is ranked third, K-NN(k=1) is ranked fourth, and Kernel is ranked fifth.

C4.5 > Naive Bayes > CN2 > k-NN(k=1) > Kernel

In both the Bayesian version of the Plackett-Luce model and Bergmann-Hommel's dynamic procedure, the rank sequence of the algorithms remains the same. However, computed performance values were different in the two models. Since Bergmann-Hommel's procedure is a nonparametric test, it has its disadvantages. Bergmann-Hommel's procedure may lack power if the sample size is small. This method is focused on hypothesis testing rather than the estimation of effects. This test requires more computing time compared to some other statistical methods.

Calvo and Santafé [33] uses the same dataset but different methods to compute which is the best algorithm. The authors used **Iman and Davenport omnibus test and the Nemenyi test** to show the difference between the algorithms. Also, it indicates which algorithm is the best among them. In this study, the first hypothesis was to test whether all the algorithms perform equally or not. In contrast, some of them had significantly different behavior. The authors used the Nemenyi test to interpret the results.

Iman Davenport's correction of Friedman's rank sum test.
Corrected Friedman's chi-squared = 14.3087, df1 = 4, df2 = 116
p-value = 1.593e-09
The p-value shown above denotes that there is at least one algorithm that performs differently than the rest. Therefore, proceeds with the post-hoc analysis of the results. Nemenyi test was used as the post-hoc analysis.

Nemenyi test
Critical difference = 1.1277, k = 5, df = 145

This procedure determines the critical difference. Any two algorithms whose performance difference is more significant than the critical difference are regarded as significantly different.
In results, it gives average ranking of the algorithms as:
C4.5 > Naive Bayes > CN2 > k-NN (k=1) > Kernel

In both the Bayesian version of the Plackett-Luce model and Nemenyi'test, the rank sequence of the algorithms remains the same. However, Nemenyi's test has many pitfalls. This test is very conservative and has low power. In many cases, this test cannot control maximum Type I error.

While comparing the results of the Bayesian Plackett-Luce model with the other two nonparametric methods, all these methods gave similar results. However, the Bayesian Plackett-Luce model provides more information about the data and easily interprets the results. Bayesian Plackett-Luce model gives a density plot, which is an advantage over other nonparametric methods.

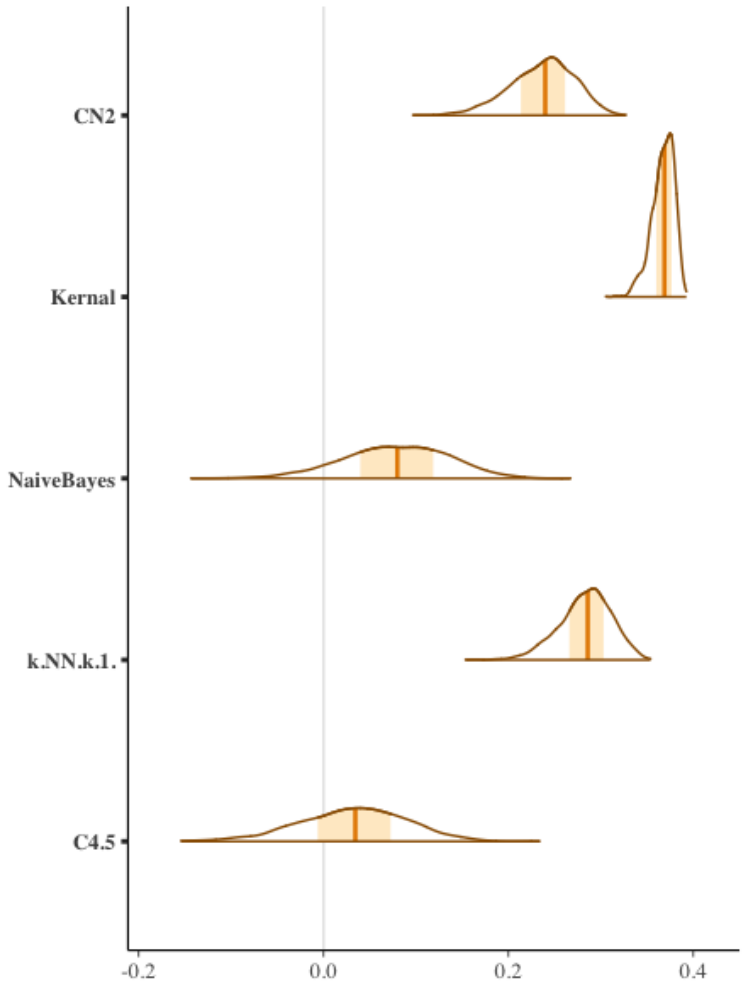

Figure 6.2: Density plot of dataset 2 parameters computed from posterior draws.

The Bayesian Plackett-Luce model, Bergmann-Hommel's dynamic procedure, and Nemenyi test give the same rank sequence on the same dataset 2. However, Bergmann-Hommel's dynamic procedure and the Nemenyi test have their disadvantages. The Bayesian version of the Plackett-Luce model gives the density plot of the performance of classification algorithms computed from the posterior draws.

Figure 6.2 shows the uncertainty intervals as shaded areas under the estimated posterior density curves. If the curve of the algorithm is narrow, then that indicates that the algorithm has low uncertainty. Even though the Kernal algorithm is the worst classifier algorithm among the given algorithms, this Kernal algorithm has low uncertainty. Suppose there is much overlapping between values of algorithms. In that case, there is a higher probability that it is hard to differentiate two algorithms since they have similar distributions. In figure 6.2, NaiveBayes and C4.5 algorithms have overlapping value distributions. This overlapping indicates that the rank is not absolute as in other methods. This plot is a great advantage because it offers the distribution of values of algorithms, not just a single value. Also, By looking at the density plot, it is easy to recognize which algorithm performs best out of 5 algorithms. This easy interpretation is another advantage of the Bayesian Plackett-Luce model.

6.2.2 Dataset 2

The goal of reanalysis on the following collected dataset was to rank the most popular programming languages [34]. This dataset has the popularity data of 28 different programming languages from July 2004 to May 2021. These reanalysis results were compared with the results of Simpson's rule. This corresponding dataset was collected from the Kaggle. Table 4.3 is the head of this dataset and does not include all the columns of the dataset.

Table 6.5 shows the posterior mean values of the programming languages. The highest valued programming language is ranked first, and the lowest valued programming language is last. The ranking sequence is as follows-

Java > PHP > Python > C/C++ > Javascript > C# > Visual.Basic > Perl > Objective.C > Matlab > R > Ruby > VBA > Delphy > Swift > Abap > Lua > Cobol > Scala > Typescript > Groovy > Haskell > Ada > Go > Rust > Kotlin > Dart > Julia.

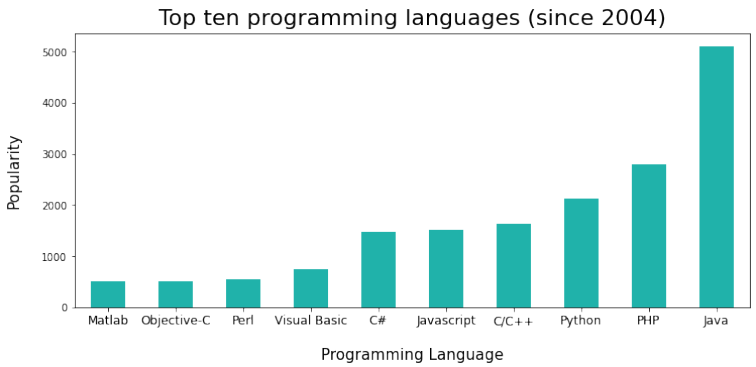

Figure 6.3: Top ten programming languages(Since 2004)

Zack Mufti also ranks the above dataset to rank the top 10 programming languages using Simpson's rule. Figure 6.3 shows the top ten programming languages based

on Simpson's rule. The ranking list with Simpson's rule is the same as the Bayesian Plackett-Luce model.

Abap	12.84
Ada	6.49
C/C++	24.50
Cobol	10.32
Dart	1.78
Delphy	14.63
Go	4.51
Groovy	7.79
Haskell	6.78
Java	28.00
Javascript	24.42
Julia	1.21
Kotlin	3.70
Lua	12.03
Matlab	19.43
Objective.C	19.74
Perl	20.77
PHP	27.00
Python	26.00
R	17.96
Ruby	17.07
Rust	3.78
Scala	10.77
Swift	14.36
Typescript	8.94
VBA	15.99
Visual.Basic	22.00
C#	23.07

Table 6.5: The posterior value of programming languages in the Bayesian Plackett-Luce model.

The Bayesian version of the Plackett-Luce model gives the density plot of programming languages computed from the posterior draws. Figure 6.4 shows how the uncertainty intervals of each programming language are shown in the plot. This plot is a great advantage because it offers the distribution of uncertainty values. Narrow distribution in the plot indicates more consistency in the popularity of programming language. Wider distribution plots indicate less consistency in the popularity of programming language. Plotting the data of all the 28 programming languages in the density plot is skewing the graph. That is Figure 6.4 is the density plot of 5 programming languages.

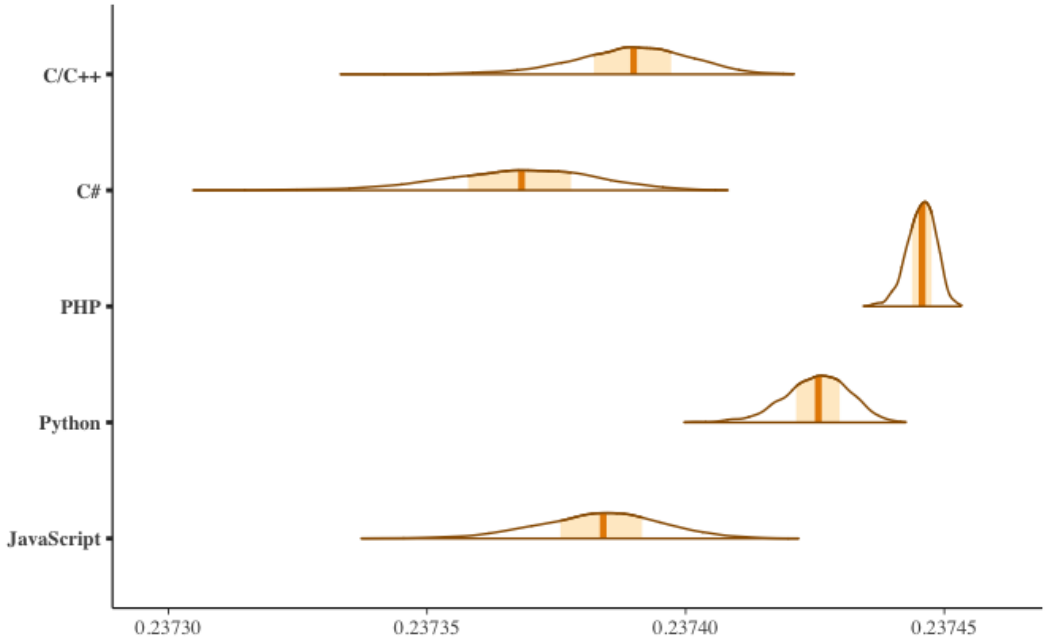

Figure 6.4: Density plot of programming languages computed from posterior draws.

6.2.3 Dataset 3

DynamoDB	4.00
PostgreSQL	13.00
Elasticsearch	8.00
Firebase	7.00
MongoDB	12.00
SQLite	10.946
MySQL	14.00
Oracle	5.86
Microsoft.SQL.Server	10.05
Cassandra	3.00
Redis	9.00
MariaDB	5.139
Couchbase	1.326
IBM.DB2	1.673

Table 6.6: Mean value of databases based on the posterior in the Bayesian Plackett-Luce model.

This dataset is the collection of opinions of 64641 users about the databases. This reanalysis focuses mainly on the data of databases that the user has worked. The reanalysis aimed to get the ranking of the databases and not compare the results with the results of other models. Here ranks were given as the input to the BPL

model. Also, to show the BPL model can handle this type of SE dataset. The sample dataset is mentioned in table 4.4. Each row is the response of each user.

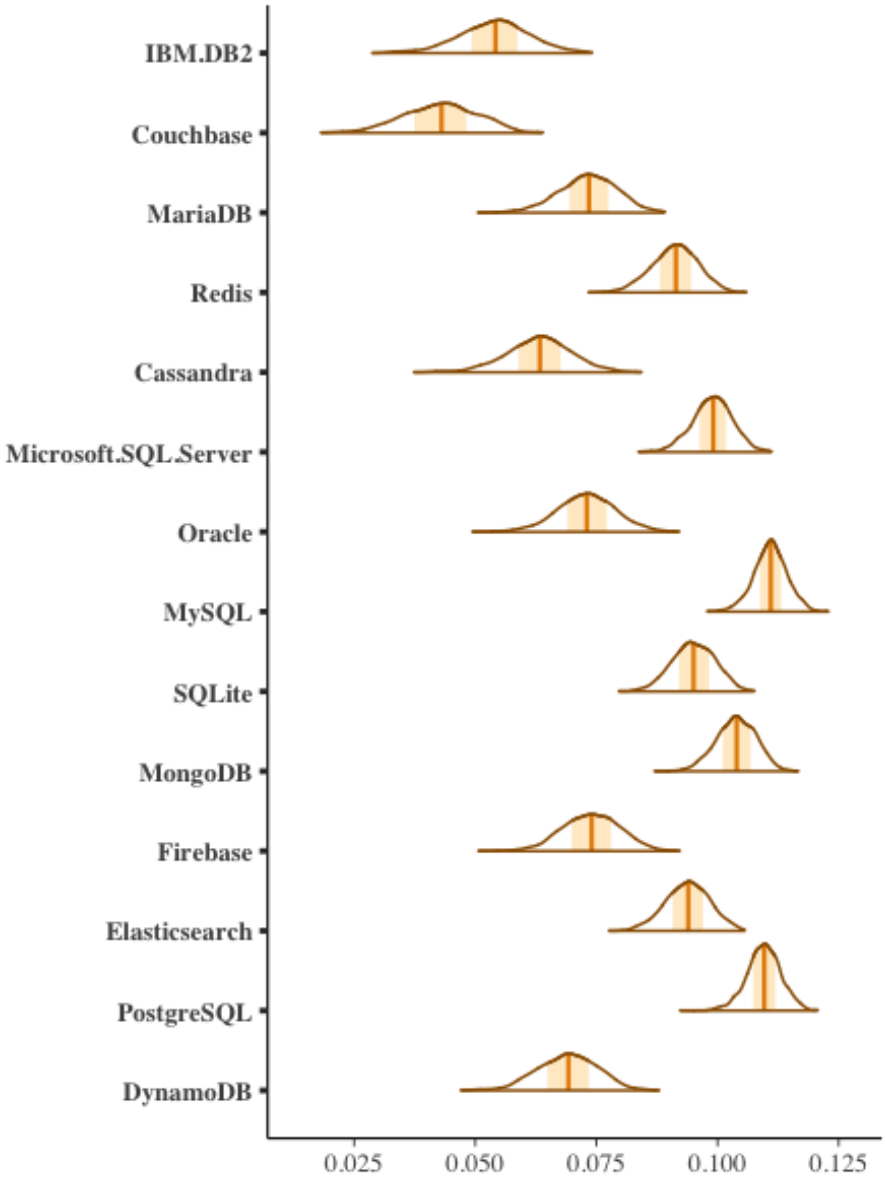

Figure 6.5: Density plot of databases computed from posterior draws.

This data is given as input to the Bayesian Plackett-Luce model. Table 6.6 is the posterior mean data of the model.

The value with the highest value is ranked as the first or most popular database. The lowest mean posterior value is ranked as the last or least popular database. The ranking sequence is as follows:

MySQL > PostgreSql > MongoDB > SQLite > Microsoft.SQL.Server > Redis > Elasticsearch > Firebase > Oracle > MariaDB > DynamoDB > Cassandra >

IBM.DB2 > Couchbase.

The Bayesian version of the Plackett-Luce model gives the density plot of databases computed from the posterior draws. Figure 6.5 shows how the uncertainty intervals of each popularity of the database are shown in the plot. This plot is a great advantage because it offers the distribution of uncertainty values. Narrow distribution in the plot indicates more consistency in the popularity of the database based on the user's usage. Wider distribution plots indicate less consistency in the popularity of the database. Also, By looking at the density plot, it is easy to recognize which database is popular among users. This easy interpretation is another advantage of the density plot in the Bayesian Plackett-Luce model.

6.3 Third Iteration of Design Science

This chapter presents the feedback from the SE researchers. It starts with a general presentation of the respondents, followed by their opinions on the study. The results were analyzed using thematic analysis, which is explained in detail in the previous chapter 4.

6.3.1 Respondents

Three respondents were interviewed in this study to get their opinion on the study. These researchers were considered for evaluation based on their research works in SE. All three researchers were statistical expertise along with SE researchers. Their analysis and opinion on the model have great value because all the researchers have deep knowledge about the ranking. Respondent A has done several research works in Bayesian analysis. Respondent B has been a senior researcher at the intersection of software engineering and applied artificial intelligence. Respondent C has done his SE research works with industrial collaboration.

6.3.2 Analysis

All three respondents that were interviewed for this study have done their researches in Software Engineering. I interviewed the researchers through video calls and recorded the call. This recorded interview was converted to the transcript.

The interviews were analyzed using the thematic analysis approach, which is described in detail in chapter 4. The results of the analysis are presented here in this chapter. This chapter presents the findings of this study, and it is structured around the main themes identified during the thematic analysis. Those main themes are the Importance of ranking in SE, Relevance of the Bayesian Plackett-Luce model to SE, Relevant datasets, Alignment of the results with the expected values, Usefulness of the BPL tool, Statistical methods for ranking.

6.3.2.1 Importance of ranking in SE

SE researchers interviewed in this study seem to be very positive towards the importance of using ranking in Software Engineering. I demonstrated the analyses with the datasets we have used in the study. Even considering that, all the three researchers felt using ranking is essential in SE. They believe that it is pretty common to use ranking in SE research.

Respondent A explained that:
"Ranking is important. Lot of time you compare different approaches and techniques to see which is the best. Ranking is the frequently used in the research by researchers".

6.3.2.2 Relevence of Bayesian Plackett-Luce model to SE

All three researchers found that using the Bayesian Plackett-Luce model is relevant to Software Engineering. Respondent A and respondent B were thoroughly convinced about using the Bayesian Plackett-Luce model for ranking in SE. These two researchers stressed the advantages provided by the bayesian approach and how the Bayesian approach makes uncertainties explicit.

Respondent B said that:
"Yeah, what particularly I like with any Bayesian approach is to make uncertainties explicit, and it is always a very nice thing to do. You said it is easy to say which one is best, and I am not entirely sure. There was a lot of overlap in the density plot over there. Observation can be done through a density plot. That is also good".

Researchers C was convinced with the model. However, he felt he needed to think deeply to agree on the applicability factor completely.

6.3.2.3 Relevant datasets

All the researchers felt the three SE datasets were relevant to the study. One researcher opinioned that datasets used in the study are relevant because some are small, and some are big among these datasets. This different size of datasets tests the ability of the model during reanalyses.
Respondent B felt:
"Datasets are relevant. You used datasets either from research studies or stack overflow".

Respondent A commented on the goal of the one dataset, i.e., finding the popular programming languages from 2004 to 2021 is less relevant.

6.3.2.4 Alignment of the results with the expected values

Two of the researchers agreed that the outcome results align well with the expected results. One of the researchers liked the density plot and the information conveyed by the plot.

Respondent B responded as:
"With the knowledge I have, and from what I observed, the results seem to be as expected. Nothing looks off in any reanalyses. I liked the idea of adding a plot. Nothing I would request beyond what you did".

Respondent A advised doing more study on prior analysis as future work. However, respondent C was not sure about this, and he did not answer this specific question.

6.3.2.5 The usefulness of the BPL tool

All three researchers found this BPL tool helpful in SE research. Respondent C felt this tool would be helpful to rank the interview data. Respondent B opinioned that this tool is better than just having the values of means and also valuable for ranking web pages, preferences, and recommendations in an e-commerce system.

Researcher A has done many research works using Bayesian inference. He felt, "I myself prefer to develop the package myself to understand the data and do the analysis myself. However, now these kinds of tools are getting popular, and many of them are using these packages as ready-to-use models. To researchers who do not have deep knowledge in statistics, these tools can be very useful for them to build more advanced concepts using this model".

6.3.2.6 Statistical methods for ranking

All researchers agreed with the importance of statistics during ranking. Every researcher had their reasons to use statistics during ranking. Respondent B opinioned that to decide the best among others requires conclusive evidence. Statistics provide that during ranking the objects.

According to respondent C, "It depends on the use case. When ranking the opinions based on the ranked list these types of methods are useful".

Respondent A said, "It gives an unbiased view of what the data tells you. It does not influence your suggestion too much. Bayesian models give a distribution of values".

7
Discussion

By following the methodology presented in this report, I now discuss the study after getting the results.

7.1 Applicability of the BPL model on SE problems

Bayesian Plackett-Luce model accepts ranked list and continuous values as the input. I analyzed the applicability of the BPL model with different kinds of SE problems. Consideration of different scenarios helped to know the applicability of the BPL model.

Initially, I analyzed the BPL model with the dataset that has been used in two different research works. Using this SE dataset, a few researchers have used different nonparametric to rank the SE data. Even though this specific dataset is an accuracy metric, the results given by the BPL model were compared with the results of other nonparametric models. The ranking pattern given by all these methods was the same. However, the additional information given by the BPL model provides advantages over nonparametric models. This comparison and analysis also indicated the applicability of the Bayesian Plackett-Luce model.

For the second analysis, a different scenario was considered for testing the relevancy of the Bayesian Plackett-Luce model for SE problems. Ranked lists were given as the input to the model. Most of the non-statistical models and nonparametric models do not accept ranked lists as the input. The ranking results of databases were generated using the Bayesian Plackett-Luce model based on the rank information given by several users in the Stackoverflow survey. This scenario also indicates the applicability of the Bayesian Plackett-Luce model in SE problems. Also, the size of the dataset considered for the analysis was large.

For the third analysis, the popularity values of the programming languages were converted into ranks and given as the input to the Bayesian Plackett-Luce model. The BPL model can also handle these types of datasets. The popularity of the programming languages from May 2004 to June 2021 was given as a ranked list as input to the model, and it provides the ranking list of the popularity of programming languages as the output. Also, these results were compared with the results of Simpson's rule. Both the methods yield the same pattern of ranking. This scenario

also indicates the applicability of the Bayesian Plackett-Luce model in SE problems.

Apart from all these analyses, evaluation by SE researchers of the Bayesian Plackett-Luce model reconfirms the outcome of the analyses. These SE researchers who participated in the evaluation gave their opinion by going through the context of the study and analyzing the demonstration of the model using 3 SE datasets. This opinion also plays a vital part in concluding the applicability of the Bayesian Plackett-Luce model on SE problems.

7.2 Benefits provided by density plots in Bayesian Plackett-Luce model

Visualization is probably the most important tool in an applied statistician's toolbox. It is an essential complement to quantitative statistical procedures. Bayesian data analysis provides the advantage of visualizing data along with evaluation of the model. Bayesian Plackett-Luce model provides density plot, but other nonparametric methods do not give density plot. This plot is a great advantage compared with the other nonparametric methods. The density plot shows the uncertainty intervals as shaded areas under the estimated posterior density curves. Also, this plot easily interprets the ranking results. This plot plays a vital role during the comparison of the Bayesian Plackett-Luce model with the nonparametric methods.

7.3 Interview results

In the evaluation process of the model, interviewed three SE researchers to know their opinion on the intention of the study. The interviewees were selected based on their knowledge in statistics and research works in the SE field. These interviews were conducted in the video calls in a semi-structured manner.

During interviews, presented the context of the study, discussed the BPL model, and demonstrated the reanalyses to SE researchers. Questions were direct and connected to the study. Researchers responded with straightforward answers as feedback to the thesis. Their responses were converted into transcripts and analyzed results using thematic analysis. Since interviewees responded to the point, it was easy to analyze their opinions.

During the thematic analysis, after following the four-step procedure, I got six themes. These themes reflected the opinions of the three researchers. The main themes were:

- The importance of ranking in SE.
- Relevance of the Bayesian Plackett-Luce model to SE.

- Relevant datasets.
- Alignment of the results with the expected values.
- Usefulness of the BPL tool.
- Statistical methods for ranking.

To summarise the feedback given by the researchers, all the interviewees opinioned that the Bayesian Plackett-Luce model is suitable for ranking SE problems, and the BPL tool is helpful to rank the data.

7.4 Bayesian Plackett-Luce tool

To check the applicability of the Bayesian Plackett-Luce model, developed the Bayesian Plackett-Luce package using the R language. CmdStanR was used to specify the model. This package is available on GitHub and is accessible to everyone. This tool would be helpful to researchers to build more advanced concepts using this model. Also, this tool could be beneficial to professionals to use to rank their data using the Bayesian Plackett-Luce package.

7.5 Validity threats

To investigate the Bayesian Plackett-Luce model in the software engineering area, I developed the Bayesian Plackett-Luce package in the R language. Due to the limited time frame of the project, I had to reduce the scope to investigate the model using a limited number of datasets. Also, I had to consider a limited number of SE scholars for the evaluation process.

This section presents the different threats to validity for this study. Validity threats are divided into four types: construct validity, internal validity, external validity, and conclusion validity.

7.5.1 Construct validity

Construct validity is necessarily the degree to which our scales and instruments measure the properties they are supposed to measure. Construct validity primarily concerns earlier phases of a thesis and is related to the research design being conducted. Construct validity is widely considered a critical quality criterion for most empirical research in the software engineering research area. Still, many software engineering studies assume that proposed measures are valid and make no attempt to assess construct validity. Because evaluating it is very difficult or due to a lack of specific guidance for conducting the evaluation process. Even in this study, due to time constraints, I was able to assess the construct validity only theoretically.

To study the suitability of the Bayesian Plackett-Luce model for ranking in software engineering problems, I developed the BPL package to analyze the purpose of the thesis. Suitable SE datasets were needed to test the intention of the study. Usually, Real-world data is often incomplete, inconsistent, or lacking in certain behaviors and is likely to contain many errors. Data pre-processing is a standard method of resolving such issues. The input data given to the BPL model has to be processed. This package does not have any data processing methodologies. The user/researcher has to be aware of pre-processing the dataset. Suppose the dataset is given as input without pre-processing. In that case, the model may give the wrong output and may risk the construct validity.

7.5.2 Internal validity

The testing process of the developed package may not be perfect. There is already a frequentist version of the Plackett-Luce model. When I compared the results of the Bayesian Plackett-Luce model with the frequentist version of the Plackett-Luce model, the ranking sequence of both models was identical. However, latent strength values of output may vary between Bayesian and frequentist versions of Plackett-Luce models. If the developed package has bugs, mistakes may occur when analyzing the results of the various models. The bugs without the researcher's knowledge may introduce bias. Evaluation of the package with more datasets may help to reduce the biases.

In the evaluation process with researchers, interviewed three SE researchers to get their opinion on the applicability of the BPL model. Picked researchers based on their expertise in statistics and research works in SE. Even though interviewees were picked based on their research works, they may give a biased opinion or feedback without proper knowledge on this study. This may threaten the validity of the interviews.

In the evaluation, I interviewed the researchers through video calls and recorded their opinions. I converted that into the transcript and analyzed researchers' opinions using thematic analysis. I alone did the thematic analysis in the study. Mistakes in the analysis without the researcher's knowledge may introduce bias.

7.5.3 External validity

External validity is a state that would probably hinder the researchers from establishing their experiment results.

There are chances that the applicability shown in the study is only specific to the cases. These chosen cases may not cover all the aspects and possible limitations of the study. These may threaten external validity.

7.5.4 Conclusion validity

We can draw correct conclusions from our analysis. Conclusion validity is analyzing data or finding patterns in the data and making inferences at the end of the research.

Three SE datasets were considered for reanalysis to study the intention of the study. As a result, there may be some lack of confidence in the obtained results. However, due to time constraints, I could reanalyze only three datasets and compare them with some other ranking models. Nevertheless, I had to conclude the study based on our reanalysis and evaluation results.

The number of scholars considered for the evaluation process was less, threatening this thesis's conclusion validity. However, taking time constraints into account, it is not easy to involve more people than this study. During the SE scholars' evaluation, the semi-structured interview and thematic analysis helped reduce the threat.

8

Conclusion

This thesis was conducted to examine the applicability of the Bayesian Plackett-Luce model for ranking Software Engineering problems. To analyze the intention developed the Bayesian Packett-Luce package. The implemented BPL tool is an excellent tool for others to rank their data. Also, the BPL package would be helpful to researchers to build more advanced concepts using this model. This study was conducted using design science methodology to answer the following two research questions:

> **RQ 1:** How suitable is the Bayesian Plackett-Luce model for software engineering research?
>
> **RQ 2:** What are the different pieces of information that the BPL provides that make someone consider using the BPL instead of others in SE?

To answer **RQ 1**, I analyzed the model with three SE datasets after developing the Bayesian Plackett-Luce model. In one scenario, ranked lists were given as input to the model. In another scenario, an accuracy metric was given as the input to the model. In the third scenario, the continuous value metric was converted to ranks and given input to the model. Different kinds of datasets were considered for testing the applicability in reanalyses. Table 6.4, 6.6, and 6.7 shows the results of the reanalyses of the three SE datasets. During reanalyses compared the Bayesian Plackett-Luce model results with the results of other ranking methods. These results helped to conclude that the Bayesian Plackett-Luce model for software engineering problems is relevant.

Further to reconfirm the BPL model's suitability, three SE researchers requested to evaluate the relevance of the Bayesian Plackett-Luce model for Software Engineering. In the evaluation process, the BPL package was demonstrated with the datasets that have been used in the reanalyses. This demonstration and analysis helped SE researchers to give feedback on the applicability of the BPL model. After discussion with these SE researchers and thematic analysis of their feedback, all three researchers see the value in the tool. Reanalyses and feedback from SE researchers helped to conclude that the Bayesian Plackett-Luce model is applicable for ranking Software Engineering problems.

To answer **RQ 2**, the Bayesian Plackett-Luce model was analyzed with the SE dataset used in other SE research works. After reanalysis, I compared the rank-

ing results obtained from the Bayesian Plackett-Luce model with other models. Bayesian Plackett-Luce model provides density plot as additional information compared to other ranking models. Figure 6.2 shows the density plot provided by the Bayesian Plackett-Luce model. This plot gives information on the uncertainties and provides an easy interpretation of the results. The advantages provided by Bayesian inference, additional information provided by the BPL model through density plot make someone consider the BPL model on SE problems instead of other ranking models.

8.1 Future work

This thesis is only an initial step in investigating the applicability of the Bayesian Plackett-Luce model on SE problems. There are many ways to continue the research conducted in this thesis.

As I have mentioned throughout the report, the model was analyzed with the three SE datasets due to time constraints. It is good to collaborate with companies with ranking requirements to investigate the BPL model on SE problems further. This collaboration will help to examine the applicability of the BPL model on SE problems with practical use cases. In reality, we will also know more about how useful this tool is for companies.

This thesis intended to develop the model and examine the applicability of the BPL model on SE problems. In the future, a researcher could develop a framework with many ranking methods that give ranking results. Using this framework user/researcher could easily select his choice of ranking method and compare it with other ranking methods. This framework may also help to compare various parameters like efficiency with ranking methods.

Bibliography

[1] Turner H.L., van Etten, J., Firth, D. and Kosmidis, I., 2020. Modelling rankings in R: The PlackettLuce package. Computational Statistics, pp.1-31.

[2] Alvo, M. and Philip, L.H., 2014. Statistical methods for ranking data. New York: Springer.

[3] Kashyap, S., Tripathi, A. and Sharma, K., 2015, March. Analysis and Ranking of Software Engineering metrics. In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 1654-1659). IEEE.

[4] Campos, J., Ge, Y., Albunian, N., Fraser, G., Eler, M. and Arcuri, A., 2018. An empirical evaluation of evolutionary algorithms for unit test suite generation. Information and Software Technology, 104, pp.207-235.

[5] Hu, J. and Li, P., 2016, December. Improved and scalable bradley-terry model for collaborative ranking. In 2016 IEEE 16th International Conference on Data Mining (ICDM) (pp. 949-954). IEEE.

[6] Altidor, W., Khoshgoftaar, T.M. and Napolitano, A., 2009, December. Wrapper-based feature ranking for software engineering metrics. In 2009 International Conference on Machine Learning and Applications (pp. 241-246). IEEE.

[7] Perin, F., Renggli, L. and Ressia, J., 2010. Ranking software artifacts. In 4th Workshop on FAMIX and Moose in Reengineering (FAMOOSr 2010) (Vol. 120).

[8] Design Science, 13 June 2021, <en.wikipedia.org/wiki/Design_science_(methodology)>

[9] Beta distribution, 20 June 2021, <https://en.wikipedia.org/wiki/Beta_distribution>

[10] Ardakan, M.A. and Mohajeri, K., 2009. Applying design research method to it performance management: Forming a new solution. Journal of Applied Sciences, 9(7), pp.1227-1237.

[11] Eric Knauss, 2021, February. Constructive Master's Thesis Work in Industry: Guidelines for Applying Design Science Research.

[12] Liddell, T.M. and Kruschke, J.K., 2018. Analyzing ordinal data with metric models: What could possibly go wrong?. Journal of Experimental Social Psychology, 79, pp.328-348.

[13] Mattos, D.I., Bosch, J. and Olsson, H.H., 2021. Statistical Models for the Analysis of Optimization Algorithms with Benchmark Functions. IEEE Transactions on Evolutionary Computation.

[14] Carlo Alberto Furia, Robert Feldt, Richard Torker, Bayesian Data Analysis in Empirical Software Engineering Research. 205 in IEEE Transactions on Software Engineering.

[15] Gill, J., 1999. The insignificance of null hypothesis significance testing. Political research quarterly, 52(3), pp.647-674.

[16] Cumming, G., 2013. Understanding the new statistics: Effect sizes, confidence intervals, and meta-analysis. Routledge.

[17] Calvo, B., Ceberio, J. and Lozano, J.A., 2018, July. Bayesian inference for algorithm ranking analysis. In Proceedings of the Genetic and Evolutionary Computation Conference Companion (pp. 324-325).

[18] Kruschke, J.K. and Liddell, T.M., 2018. Bayesian data analysis for newcomers. Psychonomic bulletin and review, 25(1), pp.155-177.

[19] Maystre, L. and Grossglauser, M., 2015. Fast and accurate inference of Plackett-Luce models (No. CONF).

[20] Turner H.L., van Etten, J., Firth, D. and Kosmidis, I., 2020. Modelling rankings in R: The PlackettLuce package dataset. https://www.preflib.org/data/election/netflix/ED-00004-00000138.soc

[21] Garcia, S. and Herrera, F., 2008. An Extension on" Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons, <https://github.com/b0rxa/scmamp/blob/master/data/data_gh_2008.RData>.

[22] Jim Frost, NonParametric Tests Vs. Parametric Tests, <https://statisticsbyjim.com/hypothesis-testing/nonparametric-parametric-tests/>.

[23] Elise Whitley and Jonathan Ball, 2002, September. Statistics review 6: Nonparametric methods,

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC153434/>.

[24] Yu, Z., Wang, Z., You, J., Zhang, J., Liu, J., Wong, H.S. and Han, G., 2016. A new kind of nonparametric test for statistical comparison of multiple classifiers over multiple datasets. IEEE transactions on cybernetics, 47(12), pp.4418-4431.

[25] Halligan, S., Altman, D.G. and Mallett, S., 2015. Disadvantages of using the area under the receiver operating characteristic curve to assess imaging tests: a discussion and proposal for an alternative approach. European radiology, 25(4), pp.932-939.

[26] Inglehart, R. (1977). The silent revolution: Changing values and political styles among western publics. Princeton: Princeton University Press.

[27] Diaconis, P. (1988). Group representations in probability and statistics. Hayward: Institute of Mathematical Statistics.

[28] Salomon, J. A. (2003). Reconsidering the use of rankings in the valuation of health states: A model for estimating cardinal values from ordinal data. Population Health Metrics, 1, 1–12.

[29] Critchlow, D. E., Fligner, M. A., Verducci, J. S. (1991). Probability models on rankings. Journal of Mathematical Psychology, 35, 294–318.

[30] Kuechler, W. and Vaishnavi, V., 2012. A framework for theory development in design science research: multiple perspectives. Journal of the Association for Information systems, 13(6), p.3.

[31] Jack Caulifield, 2019. How to do Thematic Analysis, <https://www.scribbr.com/methodology/thematic-analysis/>.

[32] Garcia, S. and Herrera, F., 2008. An Extension on“ Statistical Comparisons of Classifiers over Multiple Data Sets” for all Pairwise Comparisons. Journal of machine learning research, 9(12).

[33] Calvo, B. and Santafé Rodrigo, G., 2016. scmamp: Statistical comparison of multiple algorithms in multiple problems. The R Journal, Vol. 8/1, Aug. 2016.

[34] Muhammad Khalid, 2021. Most Popular Programming Languages Since 2004, <https://www.kaggle.com/muhammadkhalid/most-popular-programming-languages-since-2004>.

[35] Stack Overflow Annual Developer Survey datasets for the year 2020, <https://drive.google.com/file/d/1dfGerWeWkcyQ9GX9x20rdSGj7WtEpzBB/view>.

[36] Mersmann, O., Preuss, M., Trautmann, H., Bischl, B. and Weihs, C., 2015. Analyzing the BBOB results by means of benchmarking concepts. Evolutionary computation, 23(1), pp.161-185.

[37] Rank order using a rank order questions in your surveys, Question Pro, viewed 03 August 2021, <https://www.questionpro.com/features/rank-order.html>.

[38] Plumb, A.A.O., Grieve, F.M. and Khan, S.H., 2009. Survey of hospital clinicians' preferences regarding the format of radiology reports. Clinical radiology, 64(4), pp.386-394.

[39] Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A. and Rubin, D.B., 2013. Chapter 7. Evaluating, comparing, and expanding models. Bayesian data analysis, pp.165-196.

[40] Ilhasme, Murtada. (2019). The importance of statistical methods. 10.13140/RG.2.2.22139.87842.

[41] Zack Mufti, 2020. Quick Look: Top Ten Programming Languages, <https://www.kaggle.com/zmufti/quick-look-top-ten-programming-languages>.

A

Appendix 1

A.1 Presentation to researchers during the evaluation process

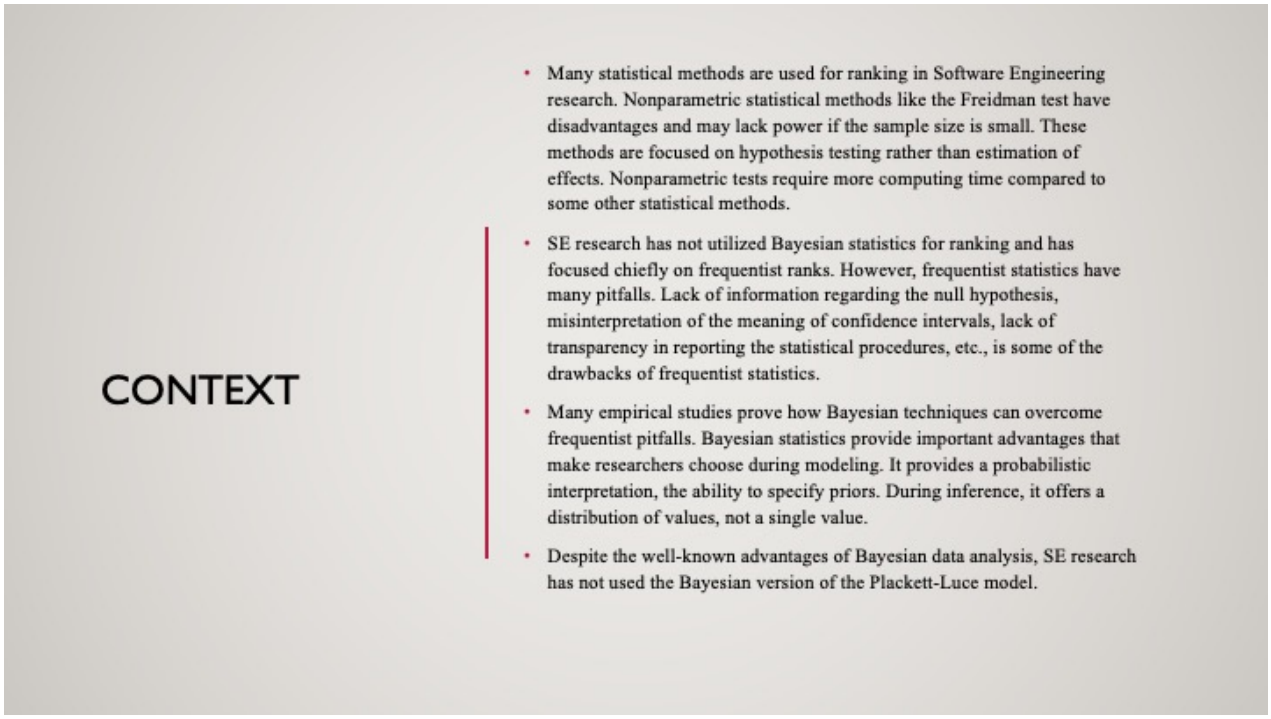

Figure A.1: Context of the thesis presented in evaluation.

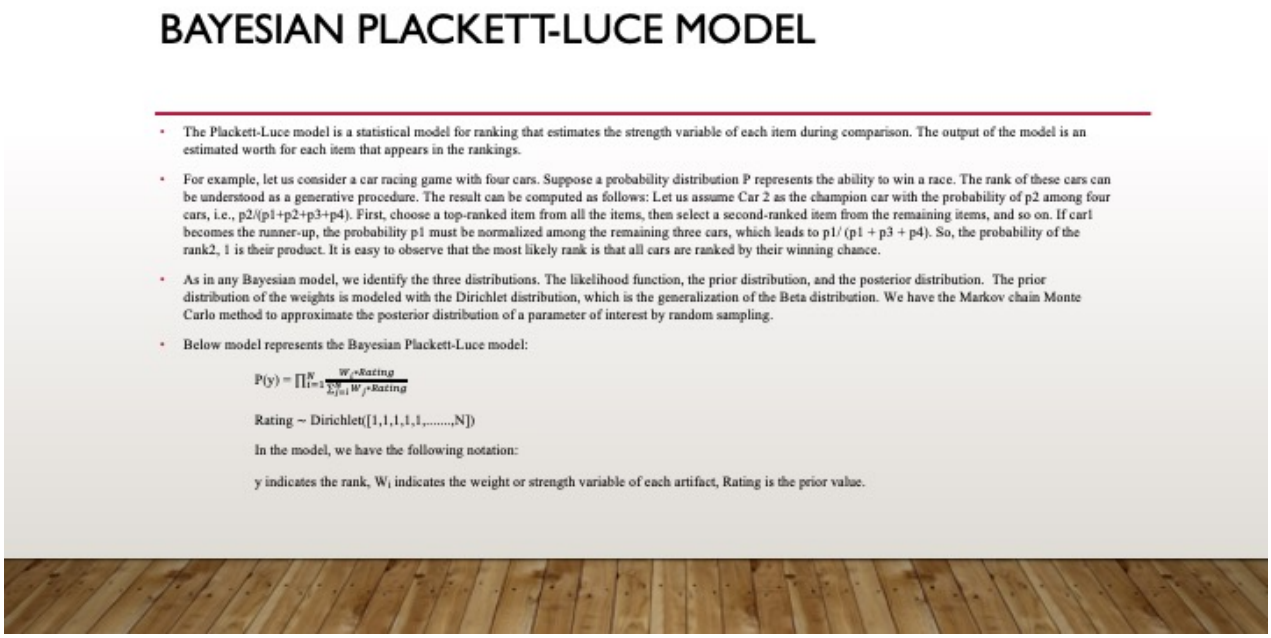

Figure A.2: Bayesian Plackett-Luce model presented in evaluation.

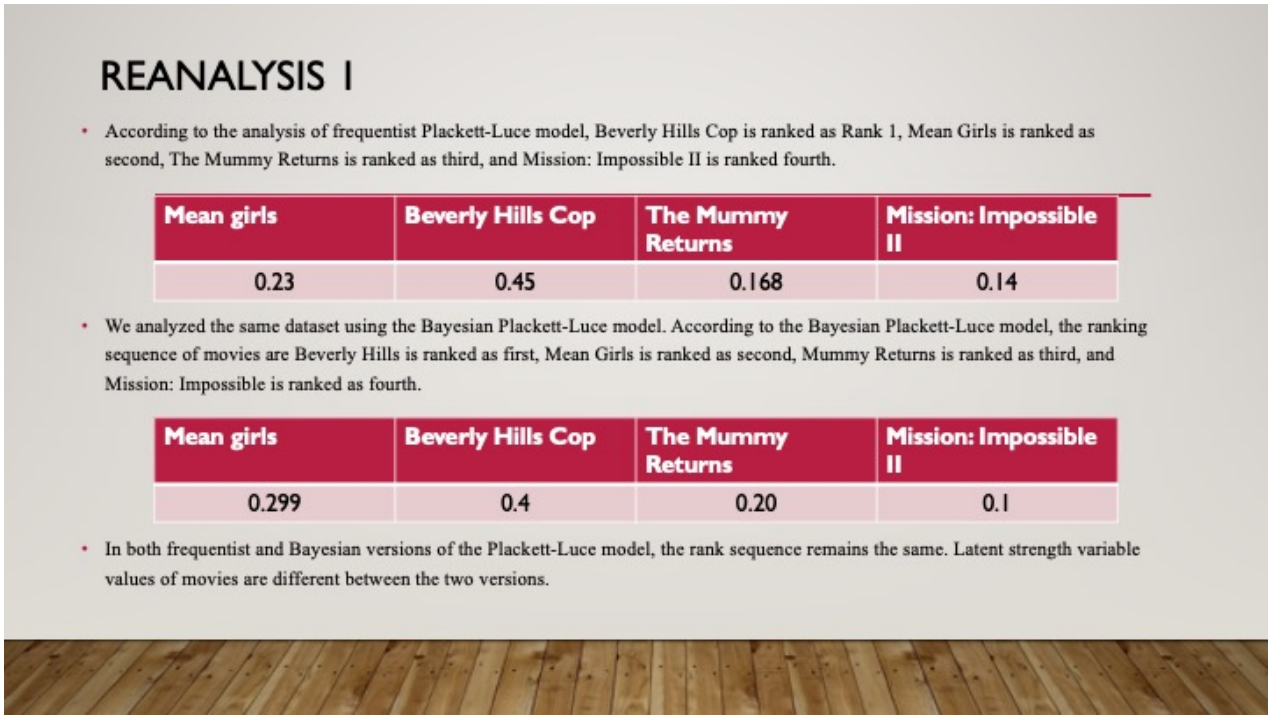

Figure A.3: Result of the reanalysis 1 presented in evaluation.

- The Bayesian version of the Plackett-Luce model gives the density plot of movies computed from the posterior draws. Figure 6.1 shows how the uncertainty intervals of each movie are shown in the plot. This plot is a great advantage because it offers the distribution of values of movies, not just a single value. By looking at this plot, it's easy to recognize how users have ranked the movies. This plot helps to identify which movie has the highest value and is ranked first by the users. Clearly, Beverly Hills Cop is ranked first by the users. This is another advantage of the Bayesian Plackett-Luce model.

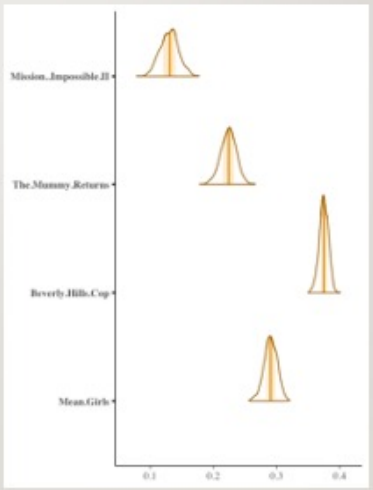

Figure A.4: Explanation to density plot of BPL for Netflix data presented in evaluation.

REANALYSIS 2 (DATASET 1)

- We have reanalyzed the model with 3 SE datasets with the Bayesian Plackett-Luce model to investigate the study's intention further.
- Dataset 1 has collected the performance of 5 supervised classification algorithms in a set of 30 datasets. The goal of reanalysis was to check which classification algorithms outperform others, i.e., ranking these classification algorithms based on the performance data.

C4.5	K-NN(K=1)	Naïve Bayes	Kernal	CN2
1.299	3.90	1.73	5.0	3.061

- The above table shows the mean performance values of algorithms when we analyze dataset 1 using the Bayesian Plackett-Luce model. According to Garcia and Herrera, the algorithm with the least mean posterior value is the best classification algorithm. The algorithm with the highest performance value is the worst classification algorithm.
- C4.5 algorithm is ranked first, Naive Bayes is ranked second, CN2 is ranked third, K-NN(k=1) is ranked fourth, and Kernel is ranked fifth.

Figure A.5: Result of the reanalysis 2 presented in evaluation.

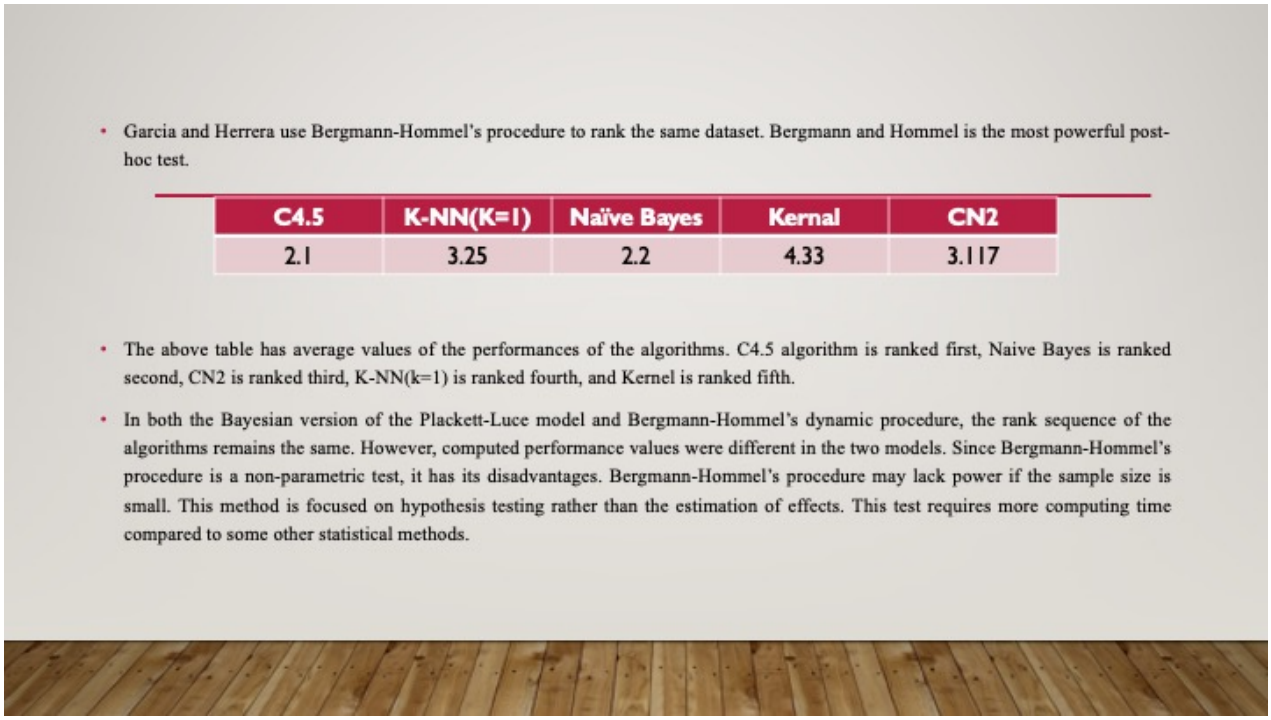

Figure A.6: Results of Bergmann-Hommel's procedure presented in evaluation.

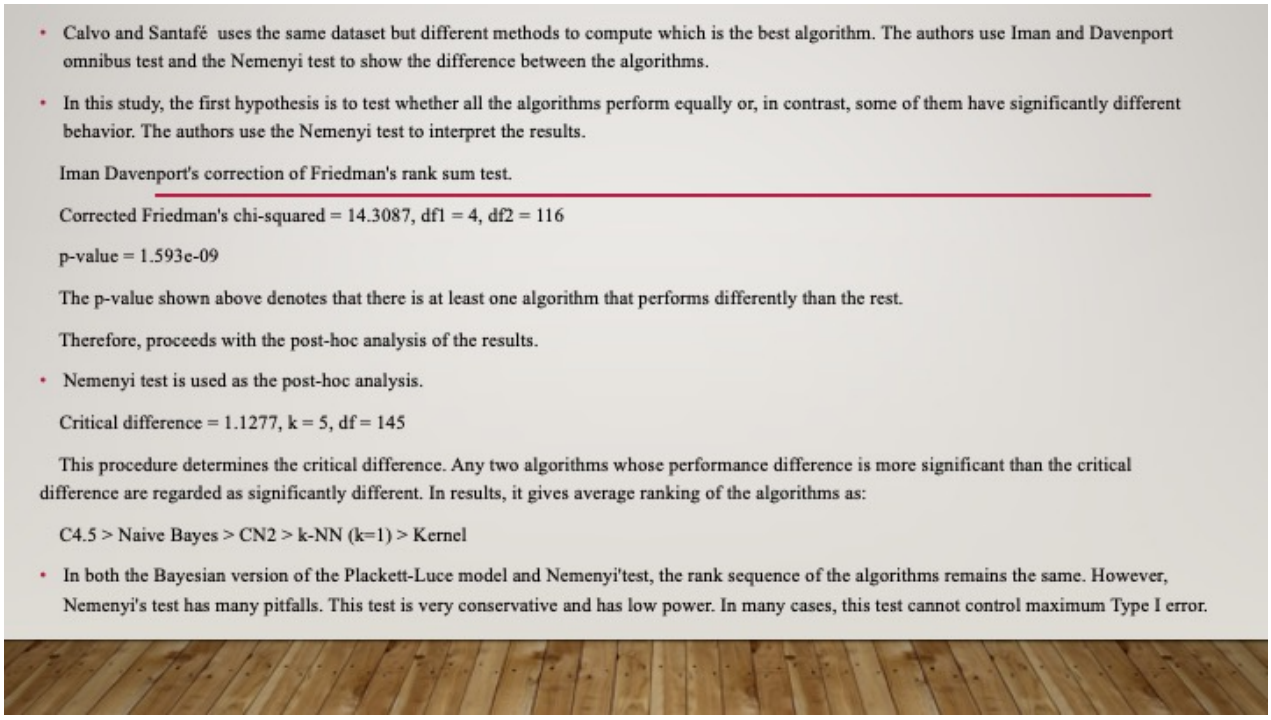

Figure A.7: Results of Nemenyi test presented in evaluation

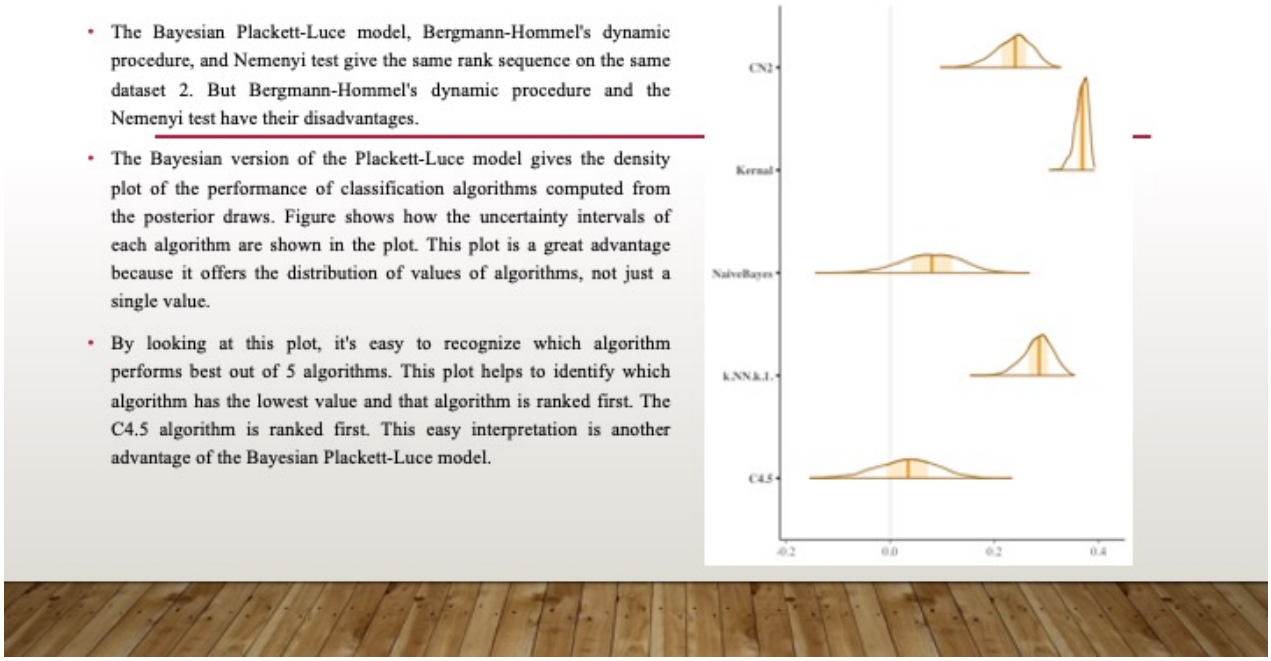

Figure A.8: Comparison of BPL and other methods presented in evaluation.

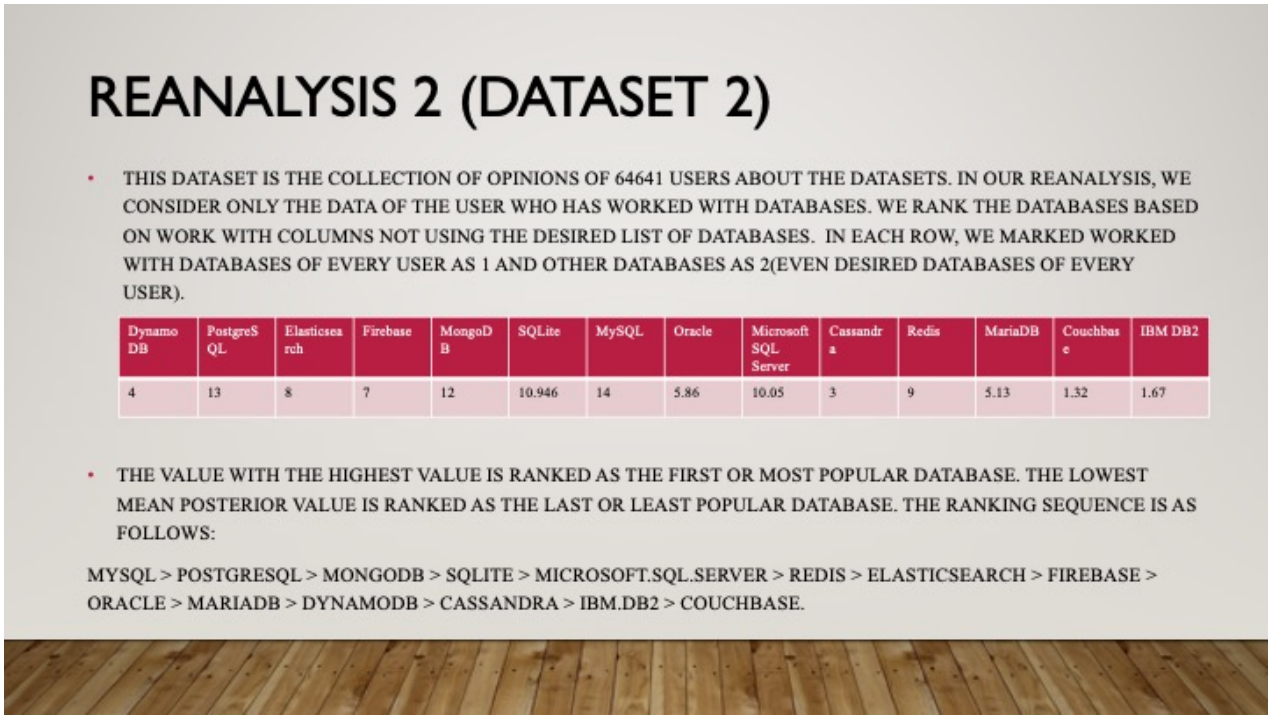

Figure A.9: Results of reanalysis of SE dataset 2 presented in evaluation.

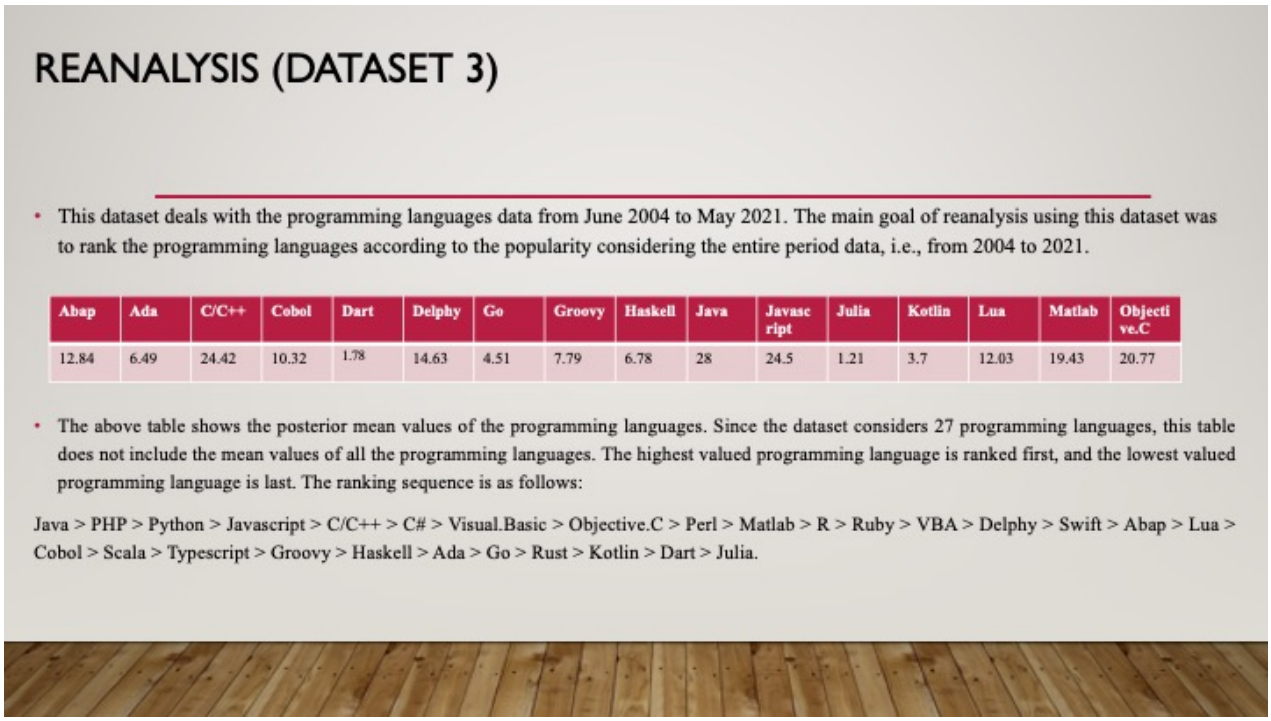

Figure A.10: Results of reanalysis of SE dataset 3 presented in evaluation.

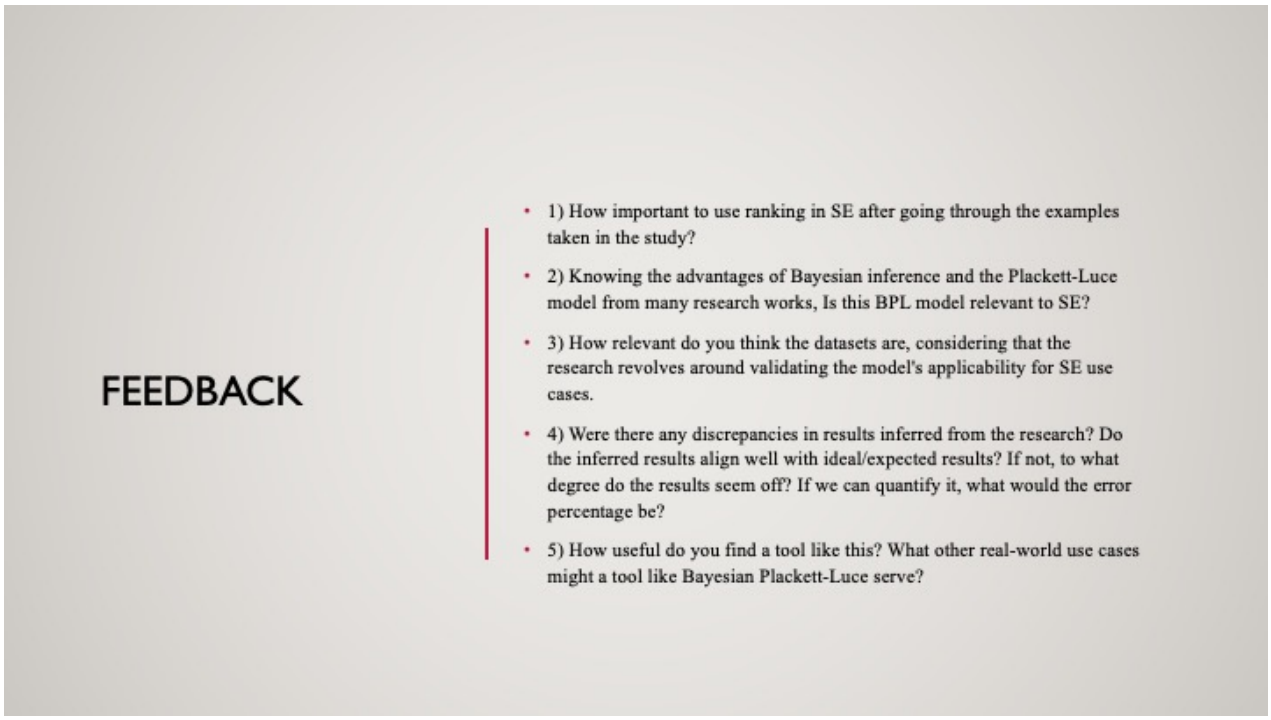

Figure A.11: Questions to researchers in evaluation.