Exponential Family Models for Rich Preference Ranking Data

Annelise Wagner

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2023

Reading Committee:
Marina Meilă, Chair
Adrian Dobra
Tyler McCormick

Program Authorized to Offer Degree:
Statistics

University of Washington

Abstract

Exponential Family Models for Rich Preference Ranking Data

Annelise Wagner

Chair of the Supervisory Committee:
Professor Marina Meilă
Statistics

Preferences can be found in a wide array of contexts, from recommender systems, to opinion polls, consumer habits, and elections. The specific method of data collection, and the types of data collected can greatly vary the tools available for analysis. We seek to expand the class of exponential family ranking models by considering two types of more rich preference data. We first look at the Recursive Inversion Model, a highly flexible exponential ranking model that can reflect high level trends in ranking data with informative parameters for inference. We expand these models for *partial rankings*, rankings that more accurately reflect the true opinions of most individuals by allowing for non-strict orderings of preference. While this addition of partial rankings accounts for increased overhead in algorithmic and computational complexity of maximum likelihood estimation, we detail methods and algorithms that ensure tractability. We also utilize this same theory to provide algorithms for calculating conditional and marginal probabilities for the Recursive Inversion Model. Using this new theory, we demonstrate the usefulness of expression ratings and rankings, highlighting a novel method of data analysis for preference data expressed as ratings. We also expand on this further by proposing a new data structure, *rankings with landmarks*, which combine the relative and absolute preferences expressed in rankings and ratings into one. This new class of rankings requires the construction of new ranking models, of which the Landmark Generalized Mallows Model (L-GMM^s) appears the most promising. We detail algorithms for maximum likelihood

estimation of the L-GMM^s, providing a solution to creating exponential ranking models containing non-invertible subsets, and demonstrate them on real world data.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1
INTRODUCTION

Wherever there is choice, there is preference, and understanding those preferences has utility from advertising, to sociology, to recommender systems, and elsewhere. How we express these preference, how we represent those expressions as data, and the subsequent statical techniques for modeling, inference, and analysis are all interlinked. Here we expand the tools available for modeling and understanding preferences by providing models for more rich and realistic preference expressions.

1.1 Expressing preferences

Preference can be expressed in many different ways, but perhaps one of the most simple is the pairwise comparison. Given a choice of two options, which do you prefer? Coke, or Pepsi? Such comparisons are simple and limited, only allowing for a binary choice to express relative preference. One might collect many pairwise comparisons of multiple options from the same individual, but this introduces new complications. Individuals may answer in a way that produces intransitivity of preferences, and the simple methods for summarizing, studying, or accurately reflecting those preference become more involved. Rankings are another way of expressing these preferences while maintaining transitivity.

Rankings, like pairwise comparisons, offer a relative way to measure preference amongst a finite or infinite set of options. Automated systems are concerned with ranking documents, websites, or files for relevancy when searching. News organizations rank universities for the best schools for engineering, business, or parties. Consumers rank their favorite movies, games, and blenders. In each of these scenarios, individuals or systems are expressing an ordering, one item preceding another, of many options. Methods and models are needed

that can allow for understanding preferences expressed as rankings on a population level. Such models can offer insight into the consensus ordering of items within a population or subpopulation, while also offering insight into how and how likely it is for individuals within the population to deviate from this consensus.

In practice, rankings come in many different forms. A strict ordering of a finite number of elements represents a *total order*, a set of pairwise preferences for every pairing and displaying strict transitivity. In a ranked choice voting system, for example, voters express preferences over a set of candidates by ranking them from first to last [11]. While some voters express total orders, more common is the case of voters only ranking a subset of candidates and leaving the remainder tied for last. This *top-t* ranking is a type of *partial ranking*, rankings that relax the strict ordering requirement and allow some items being ranked to rank equally. While democracies have strict rules for determining winners in such elections, data analysis can aid in capturing the trends in voting behaviors in the population, such as identifying which candidates voters feel similarly about, or using mixture models to reflect different types of voters.

Partial rankings are able to more accurately represent respondents' preference - they require respondents to rank all items, but introduce more flexibility by allowing for equal rankings of subsets of items. This includes top-t rankings, as noted, as well as bottom-t, and split rankings consisting of a ranking of most preferred items, a ranking of most disliked items, sandwiching in the middle items of which the respondent does not have strong preference either way. More often than these special cases, partial rankings can come in any format or structure, such as a movie fan ranking movies but allowing for an arbitrary number and size of ties in the ranking. Many such partial rankings can be inferred by looking at preferences collected as ratings.

Ratings are a more common way of expressing preference, and can be utilized in many of the same contexts. Netflix users previously rated movies on a scale of one star ($\star$) to five ($\star\star\star\star\star$) [4]. These ratings were then used to estimate ratings on movies a viewer hasn't seen, providing a way for consumers to find more films and shows they may enjoy. Likert

scales, often used to gather opinion data, use ratings framed as negative (disagreement), neutral, or positive (agreement), with varying levels of strength as needed [19].

Ratings express preference in absolute, rather than relative, terms, but the methods for studying these preferences are divorced from the methods used for rankings. Typically such models are focused on a single rating at a time, inferring for example how other covariates influence an individuals rating of the item. Other tools exist that look at multiple ratings, typically for the task of predicting or estimating unknown ratings, such as was the case with Netflix's recommender systems [41].

These two methods of expressing preferences carry complementary information, and looking at them in concert can provide more insight. Committee members of a journal utilized both rankings and ratings to assess potential submissions for publication, hoping to inform their decisions on which papers to publish via both metrics [36]. Ratings can also be used to infer partial rankings (for discrete ratings) or total orders (for continuous ratings).

In this work we treat rankings and ratings in a unified way. We show that ratings, expressed as or alongside rankings, can be used to understand and summarize the underlying preferences of a population. We start by expressing ratings as partial rankings. Ordinal ratings can be expressed as partial rankings, a class of ranking that inversion models can only handle in special cases such as top-t rankings [31] or for limited model constructions [24]. We expand the class of inversion based exponential ranking models to partial rankings, demonstrating the ability to recover consensus ordering and produce models that are informative about the underlying population for this data type.

We also introduce *rankings with landmarks*, a novel method of combining rankings and ratings that integrates the ratings into the rankings without a loss of the absolute preference utility expressed by ratings. These rankings with landmarks represent a more rich expression of preferences, synthesizing consistent information provided by the relative preferences of rankings and the more absolute preferences of ratings. We believe that this more accurately represents how people intuitively represent their preferences, and models that can reflect these information rich preferences can lead to more in depth analysis and encourage more

thorough and accurate elicitation of preferences when conducting data collection. We look specifically to the class of exponential ranking models for their ability to capture preference trends at a high level, while remaining tractable for maximum likelihood estimation, and interpretable for inference.

1.2 Models for ranking data

The task of modeling rankings encounters a number of complications. Rankings display a complex dependence structure, with each item assigned a rank restricting other items from that rank and restricting other ranks from that item. The space of possible rankings of n items also covers $n!$ possible discrete rankings, and any model over this space must assign likelihood to every possible ranking. The approaches used to combat these challenges vary, producing a wide array of approaches to modeling the space of rankings.

Thurstonian Models [5] represent rankings via a latent space of scores assigned to each object by each ranker. Scores for an item are modeled as multivariate normal distributions, themselves consisting of a sum of two multivariate normals representing the underlying distribution of scores of each item, and the individual preferences of the judge ranking the items. A ranking is produced from the scores for each item by simply ordering them from greatest score to least. This produces a model with a large number of parameters but constraints, such as assumptions about the underlying covariance, can reduce the number of parameters and simplify the fitting procedures.

Placket-Luce Models [28, 37] similarly assign a score to each item, but here the relative magnitude of scores correspond to the likelihood of a particular item being selected over another. This can be more simply framed as a multinomial distribution, sampled without replacement, with each item being represented by a single parameter p_i. The likelihood of selecting any particular item at each rank is the likelihood of selecting that item over the remaining items that have not been selected for earlier ranks. This simple parameterization is easy to work with, but produces a dependence structure between each selection.

Bradley-Terry Models [7] are similar to Placket-Luce models, but with a focus on pairwise

comparisons. Again, each item is assigned a score value which influences its likelihood of preceding other items, though the likelihood of one item preceding another depends only on the scores of the two items, and not the other items being ranked. This framing around pairwise comparisons makes the model a natural choice given pairwise preference data.

Riffle Independence Models [22] are much closer to the models that will be explored in this work. For these models, items to be ranked are decomposed into a binary tree structure with each item being a leaf node on the tree. At any node of the tree items are riffle shuffled, shuffling two sets together without modifying the order of elements within each set, and every possible riffle shuffling is parameterized by its probability of occurring. This produces a very complex model with many parameters, the number of which depends on the structure of the decomposition of items [24].

1.2.1 Inversion models

The class of models we build upon in this work represent *inversion models*. While the Riffle Independence Model considers inversions between sets of items in its riffle shuffles, inversion models as a class formalize this more concretely by mediating the likelihood by the number of inversions.

Such models can be natural approach to rankings, as any ranking can be described by a reference ranking and a number of inversions required to achieve the former from the latter [8]. This property leads to decomposability of rankings, allowing us to in turn construct decomposable likelihoods [24]. Constructing the models around a modal reference ranking leads to models that can capture underlying consensus orderings in the population [33]. Lastly, rankings expressed as inversions have simple statistics that serve as sufficient statistics for the following models, and are used throughout the models developed here. Section 2.1 details the calculation of these sufficient statistics.

Mallows Models [29] represent the most basic of inversion models. The model is parameterized by a central ranking with a dispersion parameter θ determining the probability of inversions. The likelihood of any ranking depends on the total number of inversions required

to arrive at the reference ranking. The likelihood is parameterized exponentially in the number of inversions, which simplifies maximum likelihood estimation. We review these models in Section 2.2.

Generalized Mallows Models [12] expand the Mallows Models in two similar but distinct ways. Both involve expanding the number of dispersion parameters to the number of items or ranks, minus one. This allows for more nuanced distribution of rankings, and individual parameters can be seen as controlling the dispersion of individual items or ranks. This produces not only a more accurate and flexible model, but one that is more useful and informative in inference. We review these models in Section 2.3 and Section 2.4.

Recursive Inversion Models [30] are a superclass of the Mallows and Generalized Mallows Models, and a subclass of the Riffle Independence Models. Similar to the Riffle Independence Models, a ranking is decomposed into a tree of inversions between sets of items penalized by an inversion parameter. Unlike the Riffle Independence Models, which parameterize every possible riffle shuffle of elements at each node, the Recursive Inversion Model places an exponential likelihood mediated by the number of inversions between items in the two sets to be shuffled. This added constraint produces more tractable model estimation and more interpretable models. Section 2.5 reviews the Recursive Inversion Model for total orders.

All of the inversion models explored in this work are parameterized around a central ranking, offering a means of representing and interpreting consensus in a population. These represent simple unimodal models, but can serve as building blocks for mixtures of models for describing more complex populations. In each of these scenarios the search for the optimal central ranking is believed to be NP-hard [3], but methods for efficient estimation can be utilized [30].

1.3 Motivation and contributions

We were interested in producing exponential ranking models for more rich and realistic preference data. Partial rankings have been considered by previous works, either in the limited case of top-t rankings [31] or more broadly for the Riffle Independence Models [24].

The former case was obviously quite limited (in terms of utilizing partial rankings), while the latter case involved considerable constraints on the model to maintain tractability of maximum likelihood estimation.

Our first contribution was to expand the Recursive Inversion Model, an exponential ranking model containing the Mallows and Generalized Mallows models as subclasses, to include the use of partial rankings. For such a model to be useful, it must likewise overcome the issue of tractable maximum likelihood estimation, which we demonstrate as possible with great detail on the necessary algorithms. [34] The Recursive Inversion Model allow for complex inference [30], and expanding the theory to handle partial rankings will open a wide range of data for similar inference.

Alongside this task we provide methods for estimating marginal distributions of rankings of items and inversions between items, represented by the rank matrix and inversion matrix. Estimating these marginal matrices will allow for more detailed inference of the underlying population, but also opens the door to estimating conditional rank and inversion matrices conditioned on a partial ranking [35].

In estimating RIMs from partial ranking data, a need for maintaining the information present in ratings was highlighted. Our solution came in three parts: first a new data representation, followed by a need for new models, and in turn requiring a new algorithmic details for estimation. In order to create models for handling more information rich rankings, we define *rankings with landmarks*, a new type of ranking that introduces new challenges for modeling and inference. This approach differs from previous approaches to combining ranking and rating information, ensuring that both metrics are consistent with each other rather than allowing them to be distinct [36].

For this new class of ranking we produce models that both accurately describe the underlying rankings with landmarks, but ideally do so in a way that allows for meaningful and useful inference about the population and their rankings and ratings [1]. As these rankings with landmarks represent a new constraint on the space of possible rankings, current models for rankings are unable to handle them in an informative way. It should be noted that,

to the best of her knowledge, the author is not aware of any other methods for combining consistent ranking and rating data and the models developed here are the first of their kind, thus allowing for the first time analysis of more rich preference data containing both rankings and ratings that are consistent.

1.4 Outline of thesis

We begin this work with a review of the relevant models, the class of Mallows models including Generalized Mallows, and Recursive Inversion Models. Chapter 2 details these models, upon which the other models explored in this work build.

Chapter 3 expands upon the RIM for the class of partial rankings. We start by reiterating the definition of partial rankings in Section 3.1, and continue into Section 3.2 which defines the likelihood function for a RIM utilizing partial rankings. We introduce theory and algorithms for estimating marginal and conditional rank and inversion matrices in Section 3.3 [35]. Likelihood estimation of RIMs from partial ranking data introduces new computational challenges. We detail methods for overcoming these challenges and ensuring that maximum likelihood estimation remains tractable for use, all of which we detail in Section 3.4 [34]. To assess the accuracy, efficacy, and computational overhead of utilizing partial rankings, we conduct a number of experiments on synthetic datasets in Section 3.5. Section 3.6 utilizes real world datasets to compare and contrast the RIM on partial rankings compared to total orders, and provides examples of the types of inference that can be conducted with a RIM on partial ranking datasets.

We address the task of combining rating and ranking information in Chapter 4, starting with a definition of rankings with landmarks in Section 4.1. For this new class of rankings we introduce a number of models, of which the Landmark Generalized Mallows Model represents the most promising, and detail their construction in Section 4.2. Again, the task of maximum likelihood estimation presents challenges to maintain tractability, but we detail methods of estimation of the Landmark Generalized Mallows Model in Section 4.3 and the remaining models in Section 4.4 [1]. As before, we test these methods on synthetic datasets to ensure

recoverability and assess runtime, which can be found in Section 4.5. We demonstrate inference on rankings with landmarks utilizing the Landmark Generalized Mallows Model on two real world datasets in Section 4.6.

We conclude with discussion of the works presented here and of potential future work in Chapter 5.

Chapter 2

EXPONENTIAL MODELS FOR RANKING DATA

A number of models exist that define distributions over the space of possible rankings, typically relying on a handful of principles to maintain a tractable level of complexity. The Bradley-Terry model [7] and the Plackett-Luce model [28, 37] both rely on a simple parameterization on items and pairs of items to summarize the large space. The Riffle Independence model [22] approaches the problem somewhat differently, introducing a more complex parameterization but supplementing it with a highly structured model. All of these models also rely on independence assumptions, where the order of two items does not influence the order of two unrelated items.

The class of exponential models also rely on some of these same principles. These models reflect a simple parameterization - typically one parameter for each item in the ranking that remain interpretable, plus a modal, central, or reference ranking π_0 that captures the underlying consensus of the population [33]. These models also rely on independence assumptions between the parts of the likelihood containing each of these parameters, allowing simple maximum likelihood estimation. Lastly, these models approach the problem of the complex dependence structure of rankings but expressing them as a series of inversions, taking advantage of the decomposability of rankings [24].

This work builds primarily upon the Generalized Mallows Model (GMM) [12] and the Recursive Inversion Model (RIM) [30]. We detail here the simpler Mallows Model (MM) [29], the two parameterizations of the GMM, and the original construction of the RIM.

2.1 Ranking data

Consider a set of n items to be ranked by a group of individuals, such as a selection of movies, restaurants, or authors. We will denote these items by $\mathcal{E} = \{e_1, e_2, ..., e_n\}$. The respondents create a *total order* ranking, a strict ordering of all available elements, by listing their most to least preferred items. We denote the space of all possible rankings by the notation $\mathbb{S}_\mathcal{E}$ (or $\mathbb{S}_n$ by identifying e_i with $i = 1, ..., n$), and can trivially see that $\mathbb{S}_n$ represents a sample space of $n!$ discrete total orders.

We represent a ranking by $\pi = [\pi(1), \pi(2), ..., \pi(n)]$. We refer to the item $e \in \mathcal{E}$ at rank i of π by $\pi(i)$ while its inverse $\pi^{-1}(e)$ represents the rank of e in π. In the context of preference, items that have low rank are preferred to items of high rank. For two items $e, e' \in \mathcal{E}$, if $\pi^{-1}(e) < \pi^{-1}(e')$ then e is ranked lower (and thus preferred) to e'. We say that e *precedes* e' in π which we denote by $e \succ_\pi e'$.

2.1.1 The inversion matrix Q

The models considered in this work build a distribution over $\mathbb{S}_n$ by starting from a central, modal, or reference ranking π_0, where the likelihood of any π depends on the number and structure of *inversions* of items between the two rankings.

An *inversion* of items between two rankings π and π_0 occurs when any two items $e, e' \in \mathcal{E}, e \neq e'$ appear in a different order in the two rankings. That is to say, $e \succ_\pi e'$ but $e \prec_{\pi_0} e'$. It can be useful to summarize these inversions in the *inversion matrix*, a matrix $Q(\pi, \pi_0)$ indexed by the items $e \in \mathcal{E}$.

$$Q(\pi, \pi_0) \;=\; [Q_{ee'}]_{e,e' \in \mathcal{E}}, \quad Q_{ee'} = 1 \text{ if } e \succ_\pi e', e \prec_{\pi_0} e' \text{ and 0 otherwise, for all } e, e' \in \mathcal{E}; \tag{2.1}$$

Example 1 *Consider the rankings* $\pi = [c, a, d, b, f, e]$*, and* $\pi_0 = [a, b, c, d, e, f]$.

$$Q(\pi;\pi_0) \;=\; \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & - & 1 & 0 & 1 & 1 & 1 \\ b & 0 & - & 0 & 0 & 1 & 1 \\ c & 1 & 1 & - & 1 & 1 & 1 \\ d & 0 & 1 & 0 & - & 1 & 1 \\ e & 0 & 0 & 0 & 0 & - & 0 \\ f & 0 & 0 & 0 & 0 & 1 & - \end{array} \tag{2.2}$$

The order of the rows and columns of $Q(\pi, \pi_0)$ follow the ordering of π_0. For total order rankings, the inversion matrix can be defined with only the lower triangle, noting that $Q_{e,e'}(\pi, \pi_0) = 1 - Q_{e',e}(\pi, \pi_0)$. While the matrix itself is a larger data structure than the original rankings, it proves useful as sufficient statistic for many of the models that will be explored.

Consider the *Kendall tau* distance [26], which measures the number of inversions between two rankings. The measure $d(\pi, \pi_0)$ can be defined as

$$d(\pi, \pi_0) = \sum_{e \in \mathcal{E}} \sum_{e' \in \mathcal{E}} 1_{[e \prec_{\pi_0} e']} \cdot 1_{[e \succ_{\pi} e']}. \tag{2.3}$$

If we instead express the same value using the lower triangle of the inversion matrix, we can express the Kendall tau distance as

$$d(\pi, \pi_0) \;=\; \sum_{e' \in \mathcal{E}} \sum_{e \succ_{\pi_0} e'} Q_{e'e}(\pi, \pi_0). \tag{2.4}$$

For the rankings given in Example 1, we can see that $d(\pi, \pi_0) = 4$, the sum of the lower triangle of the matrix, taking nonzero value for $c \succ_\pi a, c \succ_\pi b, d \succ_\pi b,$ and $f \succ_\pi e$.

While the data structure is larger than the original rankings used to construct it, the usefulness of the inversion matrix becomes more evident when considering multiple rankings.

For a reference ranking π_0 and set of rankings $\pi_1, ..., \pi_N \in \mathbb{S}_n$, the inversion matrix $\bar{Q}(\pi_0)$ is defined as

$$\bar{Q}(\pi_0) = \sum_{i=1}^{N} Q(\pi_i, \pi_0) \tag{2.5}$$

Using this matrix one can determine, for example, the sum of the Kendall tau distance $d(\pi, \pi_0)$ for all π_i simultaneously. Here $\bar{Q}_{e,e'}(\pi_0)$ measures the number of times item $e \succ_\pi e'$ in all the rankings, given $e \prec_{\pi_0} e'$.

For a second reference permutation $\pi_0{}'$, the matrix $Q(\pi, \pi_0{}')$ can be calculated from the matrix $Q(\pi, \pi_0)$ by reordering the rows and columns, and the same holds true for $\bar{Q}(\pi_0)$ and $\bar{Q}(\pi_0{}')$. This property will prove useful for managing the significant statistics of all models explored in depth here. For this reason, in cases where the specific ordering of rows and columns is not relevant the inversion matrix will be expressed as $Q(\pi)$.

2.2 Mallows Model (MM)

The exponential ranking models we explore in this work seek to define a probability for each $\pi \in \mathbb{S}_n$, w.r.t. some central (typically modal) ranking denoted by π_0. The most basic of these models is the Mallows model.

The Mallows model [29] is defined by a modal ranking π_0, and a single parameter $\theta \in [0, 1]$ which represents the probability of an inversion w.r.t. the modal ranking. Since the total number of inversions is the Kendall tau distance, the probability of a ranking π follows an exponential distribution, i.e.

$$P^{\mathrm{MM}}(\pi|\pi_0, \theta) = \frac{1}{Z(\theta)}\theta^{d(\pi,\pi_0)}, \tag{2.6}$$

where $Z(\theta)$, is the normalization constant and can be easily derived as

$$Z(\theta) = \prod_{k=0}^{n-1}\sum_{i=0}^{k}\theta^k = (1)(1+\theta)(1+\theta+\theta^2)...(1+\theta+...+\theta^{(n-1)}) \tag{2.7}$$

which can be computed in closed form [29]. This tractable normalization constant is one of the most remarkable properties of Mallows and Generalized Mallows Models.

It can be observed from the probability in equation (2.6), under this parameterization, that $\theta = 1$ represents the uniform distribution over $\mathbb{S}_n$ with no preference of item orders. Values of θ near 1 correspond to models with large dispersion of rankings. Values of θ near 0 imply stricter orderings, where items are unlikely to invert and rankings remain close (in inversion space) to π_0.

Example 2 *For a Mallows model parameterized by $\pi_0 = [a, b, c, d, e, f]$ and θ, the probability of the ranking π from Example 1 depends only on the number of inversions between π and π_0, which is 4. Thus, the probability of the ranking π*

$$P^{\mathrm{MM}}(\pi|\pi_0, \theta) = \frac{1}{Z(\theta)}\theta^4 \tag{2.8}$$

and $P(\pi_0|\pi_0, \theta) = \frac{1}{Z(\theta)}$.

One limitation of the Mallows Model is that it weighs each inversion equally according to a single factor θ, regardless of the item or rank. We can overcome this limitation and improve the flexibility of the model by replacing the single parameter θ with a vector $\boldsymbol{\theta} \in [0, 1]^{n-1}$. This leads to two alternative *Generalized Mallows Models* [14], with both models exploiting the decomposition of the inversion distance $d(\pi, \pi_0)$ into a sum of inversion counts, each term corresponding to one parameter in $\boldsymbol{\theta}$. Section 2.3 explores the GMM^v which can be thought of as parameterizing each item in the ranking, while Section 2.4 explores the GMM^s which parameterizes each rank of the model.

2.3 V-parameterized Generalized Mallows Model (GMM^v)

In this model, the probability of a ranking can be decomposed into the product of factors, one for each item in $\mathcal{E}$ (excepting one). One natural way to define the GMM^v is to imagine constructing the permutation π in stages, as suggested by the model's original name, the Stagewise Ranking Model [14].

Items are removed from π_0 one by one, and inserted into π, starting with the last item $\pi_0(n)$, and continuing backwards to $\pi_0(1)$. The first stage, $i = n$, is deterministic. Each subsequent item $\pi_0(i)$, for $i = n-1, n-2, \ldots 1$ is inserted into the new ranking π at rank $1 + v_i$, where $v_i \sim \exp(\theta_j, n-j)$; here $\exp(\theta, k)$ denotes the geometric distribution with parameter θ and range $\{0, 1, \ldots k\}$. To show the dependency of v_i on π_0 and π, we may use the more formal notation $v_i(\pi; \pi_0)$. For any π_0, the set of values $v_{1:n-1}(\pi; \pi_0)$, called the *code* of π, uniquely determines π, and $\sum_{i=1}^{n-1} v_i(\pi; \pi_0) = d(\pi, \pi_0)$ [14].

Parameter θ_i penalizes the inversions at stage i. Thus, the value of θ_i represents the probability of item $(\pi_0)_i$ being selected and inverting with any of the items that succeed it in π_0. Consequently, the probability of a ranking π is

$$P^{\mathrm{GMM}^v}(\pi|\pi_0, \vec{\theta}) = \prod_{i=1}^{n-1} \frac{1}{Z_{n-i}(\theta_i)} \theta_i^{v_i(\pi;\pi_0)}. \tag{2.9}$$

The expressions $Z_{n-i}(\theta_i)$ are the normalization constants of the respective geometric distributions, which can be expressed in closed form as

$$Z_j(\theta) = \sum_{k=0}^{j} \theta^k = \begin{cases} j+1 & \text{if } \theta_j = 1 \\ \frac{\theta^{j+1}-1}{\theta-1} & \text{otherwise} \end{cases} \tag{2.10}$$

Furthermore, if $\theta_1 = \theta_2 = \ldots = \theta_{n-1} = \theta$, the GMM^v in equation (2.9) becomes the Mallows model. Its normalization constant is equal to

$$Z(\theta) = \prod_{i=1}^{n-1} Z_i(\theta). \tag{2.11}$$

It has been previously demonstrated [33] that the $v_{1:n-1}$ code can be calculated, like the Kendall tau distance, from the inversion matrix $Q(\pi, \pi_0)$ by

$$v_i(\pi; \pi_0) \;=\; \sum_{e' \prec_{\pi_0} e} Q_{e'e}(\pi; \pi_0) \quad \text{where } e = \pi_0(i). \tag{2.12}$$

Example 3 *Consider the ranking* $\pi = [c, a, d, b, f, e]$ *and a* GMM^v *model parameterized by* $\pi_0 = [a, b, c, d, e, f]$ *and* $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$. *Refer back to equation* (2.2) *for the matrix* $Q(\pi; \pi_0)$. *Because* $Q(\pi; \pi_0)$ *is ordered by* π_0, *each* v_i *is the sum of columns of the lower triangle of* $Q(\pi; \pi_0)$. *From this matrix we can calculate the code for* π, *which is* $(1, 2, 0, 0, 1)$, *meaning* $v_1 = 1$, $v_2 = 2, \ldots$. *Below we show how to reconstruct* π *from* π_0 *and its code.*

i	v_i	θ_i	π *after insertion*	π_0 *after deletion*	*Factor in Eq.* (2.9)
–	–	–	$[f]$	$[a, b, c, d, e]$	1
5	1	θ_5	$[f, e]$	$[a, b, c, d]$	$\theta_5^1 / Z_5(\theta_5)$
4	0	θ_4	$[d, f, e]$	$[a, b, c]$	$\theta_4^0 / Z_4(\theta_4)$
3	0	θ_3	$[c, d, f, e]$	$[a, b]$	$\theta_3^0 / Z_3(\theta_3)$
2	2	θ_2	$[c, d, b, f, e]$	$[a]$	$\theta_2^2 / Z_2(\theta_2)$
1	1	θ_1	$[c, a, d, b, f, e]$	–	$\theta_1^1 / Z_1(\theta_1)$

(2.13)

The probability $P(\pi; \pi_0, \vec{\theta})$ *is the product of the independent probabilities for each stage, matching equation* (2.9).

2.4 S-parameterized Generalized Mallows Model (GMM^s)

The GMM^s is the dual of the GMM^v, and is similarly parameterized by π_0 and $\boldsymbol{\theta} = (\theta_1, \ldots \theta_{n-1})$. Where previously $d(\pi, \pi_0)$ was decomposed as a sum of inversions with respect to ranks of π_0, here we decompose $d(\pi, \pi_0) = \sum_{i=1}^{n-1} s_i(\pi; \pi_0)$, where each s_i counts inversions w.r.t. rank i of π.

This decomposition can also be viewed as a stagewise construction of π from π_0. Starting from an empty π; at stage $i = 1, 2, \ldots n-1$ rank i of π is filled with a random item extracted (and subsequently removed) from the modal ranking π_0. One samples $s_i \sim \exp(\theta_i; n-i)$ and selects the item on rank $1+s_i$ of π_0, removing it and placing it at ranking i of π (alternatively this can be viewed as skipping over s_i items in π_0 to choose $\pi(i)$). The last item remaining in π_0 becomes $\pi(n)$.

Consequently, the probability of a ranking π is

$$P^{\mathrm{GMM}^s}(\pi|\pi_0,\vec{\theta}) = \prod_{i=1}^{n-1} \frac{1}{Z_{n-i}(\theta_i)} \theta_i^{s_i(\pi;\pi_0)} \tag{2.14}$$

where again $Z_{n-i}(\theta_i)$ represents the normalization constant of the respective geometric distribution.

In has also been demonstrated [31] that the $s_i(\pi;\pi_0)$ terms can be calculated, like the Kendall tau distance, from the inversion matrix $Q(\pi,\pi_0)$ by

$$s_i(\pi;\pi_0) \;=\; \sum_{e' \succ_{\pi_0} e} Q_{ee'}(\pi;\pi_0) \quad \text{where } e = \pi(i). \tag{2.15}$$

Note that s_i are now row sums, and that, again, only entries in the lower triangle of Q are counted.

Example 4 *Let $\pi = [c,a,d,b,f,e]$ as before, and a* GMM^s *parameterized by $\pi_0 = [a,b,c,d,e,f]$ and $\vec{\theta} = (\theta_1,\theta_2,\theta_3,\theta_4,\theta_5)$. Refer back to equation (2.2) for the matrix $Q(\pi;\pi_0)$.*

From this matrix we can calculate $s_{1:5}(\pi;\pi_0)$ as follows: $\pi(1) = c$, hence $s_1 = \sum_{e' \succ_{\pi_0} c} Q_{ce'}(\pi;\pi_0)$, the sum of row c in $Q(\pi;\pi_0)$, up to the diagonal; further, $\pi(2) = a$, hence $s_2 = \sum_{e' \succ_{\pi_0} a} Q_{ae'}(\pi;\pi_0) = 0$ (since there is no item before a in π_0), and so on. Below we show how to reconstruct π from π_0 and its code $s_{1:n-1}$.

rank	s_i	π *after insertion*	π_0 *after deletion*	*Factor in Eq* (2.14)
1	2	$[c]$	$[a,b,d,e,f]$	$\frac{\theta_1^2}{Z_1(\theta_1)}$
2	0	$[c,a]$	$[b,d,e,f]$	$\frac{\theta_2^0}{Z_2(\theta_2)}$
3	1	$[c,a,d]$	$[b,e,f]$	$\frac{\theta_3^1}{Z_2(\theta_3)}$
4	1	$[c,a,d,b]$	$[e,f]$	$\frac{\theta_4^1}{Z_3(\theta_4)}$
5	0	$[c,a,d,b,f]$	$[e]$	$\frac{\theta_5^0}{Z_4(\theta_5)}$
6	–	$[c,a,d,b,f,e]$	–	1

(2.16)

Finally, the probability of the ranking π is the product of the factors in the last column,

matching equation (2.14).

2.5 Recursive Inversion Model (RIM)

The Recursive Inversion Model (RIM) [30] is a class of models that builds upon the ideas of, and contains as a subclass, the GMM^v. Like the GMM^v, a RIM over n items is parameterized by a ranking denoted $\pi_\tau \in \mathbb{S}_n$ and a set of dispersion parameters $\theta_{1:n-1} \in [0,1]$. Unlike the GMM^v, which merges each item into a ranking one at a time, the Recursive Inversion Model follows a binary tree structure τ with sets of items being shuffled together at each node.

Given the set of items $\mathcal{E} = \{e_1, ..., e_n\}$, the structure of the Recursive Inversion Model $\tau(\boldsymbol{\theta})$ decomposes the set $\mathcal{E}$ recursively in a binary tree until each leaf of the tree contains just one e_i. The nodes of the tree are denoted by $\mathcal{I}_i$, and take the form $\mathcal{I}_i = (\mathcal{I}_{i_L}, \mathcal{I}_{i_R}, \theta_i)$, with the left subtree $\mathcal{I}_{i_L}$, the right subtree $\mathcal{I}_{i_R}$, and the node parameter θ_i. Each subtree consists of similar internal nodes, with the leaves of the tree consisting of the elements $\mathcal{L}_i \subset \mathcal{E}, L_i = |\mathcal{L}_i|$ (or $\mathcal{R}_i \subset \mathcal{E}, R_i = |\mathcal{R}_i|$), or is itself just a leaf node.

The *reference ranking* of the model, denoted for the RIM by π_τ, can be found by traversing the tree in *preorder*, starting at the root node and progressing down first the left, then the right subtree at each node, and assembling a ranking in the order in which leaf nodes e_i are encountered.

A ranking of the elements $\mathcal{E}$ can be assembled from the tree by similarly traversing it in preorder, with each internal node producing a ranking of the subset of elements it contains, and passing it up the tree toward the root node. At any internal node $\mathcal{I}_i = (\mathcal{I}_{i_L}, \mathcal{I}_{i_R}, \theta_i)$, rankings of the subsets $\mathcal{L}_i$ and $\mathcal{R}_i$ are returned by the nodes $\mathcal{I}_{i_L}, \mathcal{I}_{i_R}$. The rankings of subsets $\mathcal{L}_i$ and $\mathcal{R}_i$ are placed with the left preceding the right and *interleaved* (or mixed together) with one another following a likelihood of inversions mediated by the dispersion parameter θ_i. This interleaving follows a *riffle shuffle* which allows for two sets of elements to be shuffled together without allowing the elements within sets to change order. This implies that the likelihood at each node $\mathcal{I}_i$ does not depend on the likelihood of the subtrees below it, and was the same principle utilized in the similarly structured Riffle Independence Models [21].

More specifically, at an internal node $\mathcal{I}_i$, the likelihood under the RIM depends on the number of inversions of items in the left set (denoted $l \in \mathcal{L}_i$) and items in the right set (denoted $r \in \mathcal{R}_i$) at that vertex. The *vertex discrepancy* at the node is defined as

$$v_i(\pi, \pi_\tau) = \sum_{l \in \mathcal{L}_i} \sum_{r \in \mathcal{R}_i} Q_{rl}(\pi), \tag{2.17}$$

which counts the number of instances where items in the right set precede items in the left set, irrespective of the ordering of items in the two sets. While this appears distinct from the $v_i(\pi, \pi_0)$ defined in Equation 2.12, the GMM^v can be viewed as a RIM with a structure such that each node's left child is a leaf, in which case Equation 2.17 yields Equation 2.12.

Much like the GMM^v, the likelihood of the internal node is penalized by the number of inversions following a geometric distribution. For a ranking $\pi \in \mathcal{E}$ and a RIM $\tau(\vec{\theta})$,

$$P_{\tau(\vec{\theta})}(\pi) \propto \prod_{\mathcal{I}_i \in \mathcal{I}} \theta_i^{v_i(\pi, \pi_\tau)} / Z_{L_i, R_i}(\theta_i) \quad \text{with } Z_{n,m}(\theta) = \frac{(\theta)_{n+m}}{(\theta)_n (\theta)_m} \text{ and } (\theta)_n = \prod_{k=1}^{n} \frac{1 - \theta^{-k}}{1 - \theta}. \tag{2.18}$$

The normalization constants $Z_{L_i, R_i}(\theta_i)$ depend only on the size of the left and right subsets, $L_i = |\mathcal{L}_i|$ and $R_i = |\mathcal{R}_i|$, and on the dispersion parameter θ_i. It has been previously demonstrated that these are *Gaussian Polynomials* and can be computed recursively in polynomial time [30].

It should be noted that while our definition listed here implies $\boldsymbol{\theta} \in [0, 1]^{n-1}$, values of θ_i greater than 1 also produce a valid likelihood. These instances indicate a preference of items on the right subtree $\mathcal{I}_{R_i}$ to come before items on the left subtree $\mathcal{I}_{L_i}$. If one were to invert these two subtrees in the model, a new value of θ_i can be found to produce the same likelihood distribution.

This is more evident under an alternate parameterization where $\theta_i = e^{-\phi_i}$ with $\phi \in [0, \infty)$, values of $\phi_i \in (-\infty, 0)$ would correspond to $\theta_i \in (1, \infty)$. These values can be corrected by inverting the two subtrees and replacing ϕ_i with $-\phi_i$.

This process is known as *canonicalization*, where a canonical tree is uniquely defined for

Example of RIM
reference permutation
$\pi_0 = [a,\ b,\ c,\ d,\ e,\ f]$
tree structure
$(((a, b),\ (c,\ d)),\ (e,\ f))$
parameters (in pre-order)
$\boldsymbol{\theta} = (\theta_1,\ \theta_2,\ \theta_3,\ \theta_4,\ \theta_5)$
RIM model
$\tau(\boldsymbol{\theta}) = (((a,\ b,\ \theta_3),\ (c,\ d,\ \theta_4),\ \theta_2),\ (e,\ f,\ \theta_5),\ \theta_1)$

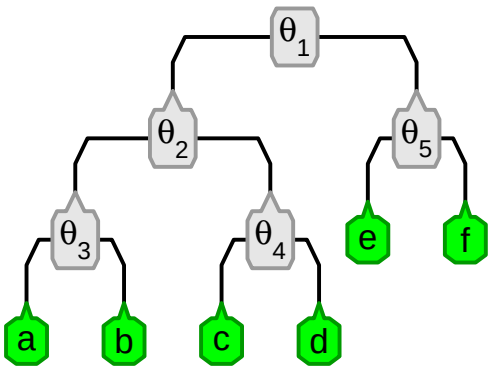


Figure 2.1: We can express all parameter, reference permutation, and structure information about a given RIM using nested parentheses. Here we traverse the RIM in pre-order from the root node and we use this ordering to identify internal nodes $\mathcal{I}_i \in \mathcal{I}$ and their parameters θ_i.

any model where no $\theta_i = 1$ and no $\theta_{root} = \theta_{child}$. Cases where $\theta_i = 1$ would indicate a uniform preference of interchangeable left and right items at a node, meaning no unique canonical tree can exist. Likewise a structure τ where any node $\mathcal{I}_i$ has parameter θ_i equal to $\theta_{i'}$ of one of its children can be restructured (maintaining relative order of children) to τ' with identical likelihood. If a RIM has a canonical structure, then the reference ranking π_τ is also the modal ranking.

Figure 2.1 gives an example of a RIM over the elements $\mathcal{E} = \{a, b, c, d, e, f\}$. As can be seen, we express the entirety of a RIM using a series of nested parenthetical expressions for each node. If $\theta_2 = \theta_3$, replacing $((a, b, \theta_3), \mathcal{I}_4, \theta_2)$ with $(a, (b, \mathcal{I}_4, \theta_3), \theta_2)$ produces an equivalent model.

For the purpose of completeness, and for utilization in expanding the RIM later in this work, we include the recursive algorithm to compute $Z_{L,R}$, which is repeated from [30] under the name VCDF. As we will see, it is advantageous to cache all the intermediate results of Algorithm VCDF in the (zero indexed) $(L+1) \times (R+1)$ matrix Z, a step that will benefit the problem of fitting partial rankings but is unnecessary for a model over total orders.

Algorithm 1 Algorithm VCDF

Input L, R, θ
for $l \leftarrow 1, \dots L$ **do**
  $Z_{l,0} \leftarrow 1$
for $r \leftarrow 1, \dots R$ **do**
  $Z_{1,r} \leftarrow \frac{1-\theta^{r+1}}{1-\theta}$
for $l \leftarrow 2, \dots L$ **do**
  for $r \leftarrow 1, \dots R$ **do**
    $Z_{l,r} \leftarrow Z_{l,r-1} + \theta^r Z_{l-1,r}$
Output $[Z_{l,r}]_{l=1:L,r=1:R}$

2.6 Conclusion

The class of exponential ranking models have a number of favorable properties for our purposes. The models reflect rankings through a series of independent codes, with each unique set of codes corresponding 1-to-1 to a ranking through various methods of translating codes to rankings according to the reference ranking (and structure, in the case of the RIM). This creates a method of producing rankings and estimating the likelihood of rankings that does not suffer from the complex dependence structure of rankings.

Framing rankings around inversion is also a natural approach, leading to models that are interpretable (with θ parameters corresponding directly to the likelihood of inversions), and models that can be estimated from sufficient statistics, in the form of the inversion matrix.

As these models all depend on a central modal ranking π_0, they inherently produce consensus rankings as part of the model. This aids greatly in the task of inference on rankings, as it not only summarizes the consensus in the population, but in the case of the RIM can also inform as to the relationship between different items being ranked in the structure of the model.

It should be noted that these models are a family of models, of which the RIM is the superclass. A Recursive Inversion Model consisting entirely of nodes that split one item on the left and the remainder of tree on the right can be expressed as a Generalized Mallows Model. Similarly, if we limit this model to a single θ value, it becomes a Mallows Model,

thus the work that we do to expand the Recursive Inversion Model will be applied to the subclass of models representable as RIMs.

Chapter 3

RECURSIVE INVERSION MODELS FOR PARTIAL RANKING DATA

The Recursive Inversion Model represents an extremely flexible exponential ranking models. While it describes rankings around inversions much like the Riffle Independence Model, it parameterizes these inversions in strict but tractable way. This results in a model that can capture not just consensus rankings in a population, but also reflect relationships between item preferences in the structure of the model. It accomplishes all of this while being defined by parameters that don't just describe the underlying distribution of rankings in the population, but do so in an inferentially informative way.

Naturally, given the usefulness of this model, we would hope to be able to apply it to a wide array of ranking data. Prior to this work, it was shown that the hierarchical decomposition of the Riffle Independence Model could be utilized to describe *partial rankings* [24]. As the Recursive Inversion Model is a subclass of the Riffle Independence Model, it follows that partial rankings could also be described by the RIM. Exponential models have previously been expanded to the space of top-t rankings, a special case of partial rankings [31].

The ability to fit RIMs to partial rankings would open the door to applying this informative and flexible model to a wide array of datasets. In practice, people are likely to express only partial preferences between a fixed set of items. This may come from a lack of familiarity with other items, as is often seen with ranked choice elections where many electors will select a subset of candidates they support and leave the remainder unranked [17]. Likewise voters may have an idea of who they don't want to win, have preference consisting of both a top-t and bottom-t, with items between being equally ranked. A person ranking their

favorite movies might be able to strictly order a few, but would likely express uncertainty with some equally preferred items, thus allowing for a more realistic measure of preferences when eliciting partial rankings as opposed to total orders. A wealth of partial rankings can also be inferred from any ordinal rating data, such as Likert responses to surveys [19], or star ($\star$) ratings of movies [20].

This work seeks to express the likelihood of a partial ranking on a Recursive Inversion Model. From here, this leads naturally to a representation of the marginal distribution of rank and inversion matrices. The task of fitting a RIM to a set of partial rankings introduces a number of challenges and issues. We detail methods for estimation of the RIM, following closely the methods originally presented [30] with modifications to improve them generally and for the specific case of partial rankings.

We test these methods on synthetic data, ensuring the recoverability of the model and model parameters, before similarly demonstrating the same recoverability on real world full ranking data synthetically reduced to partial rankings. We also analyze real world data that exists in part or entirety as partial rankings, particularly in the context of ratings inferred as rankings. We demonstrate the usefulness of the model in describing the underlying population, allowing inference about movies, political candidates, and opinion poll questions that would have previously been inaccessible to the RIM.

3.1 Partial rankings

A *partial ranking* [24], as opposed to a total order, is a type of ranking wherein the ranker provides a ranking of items $\mathcal{E} = \{e_1, ..., e_n\}$, but is able to rank some items equally in the ordering. More formally, a partial ranking divides the items $\mathcal{E}$ into disjoint subsets $\mathcal{E}_1, ..., \mathcal{E}_K$ and provides a total order of these subsets. We denote a partial ranking by $\sigma = (\mathcal{E}_1|\mathcal{E}_2|...|\mathcal{E}_K)$, where each $\mathcal{E}_i$ represents a subset of one or more items from $\mathcal{E}$.

Partial rankings have been previously defined for Mallows Models [27] and for Riffle Independence Models [24] where it was shown that partial rankings display *complete decomposability*. This property allows us to expand the RIM, and subsequently the GMM^v, for

partial rankings.

When discussing partial rankings, we refer to each subset $\mathcal{E}_i$ as a *grade*. Using the size of each grade, with $n_i = |\mathcal{E}_i|$ and $n = \sum_k^K n_k$, the structure or *shape* of a partial ranking can be described by $(n_1, n_2, ..., n_K)$.

One common example of partial rankings are top-t rankings, such as those collected in ranked choice voting, which consist of a strict ordering of t items with the remainder of items equally ranked last. These have shape $(1, ..., 1, n - t)$ and depending on context may have fixed or variable t.

Partial rankings can also be inferred from many common data sources; any set of items given ordinal ratings can be used to produce a partial ranking of those items. Simple examples include ratings of movies, restaurants, etc on a 1-star to 5-star rating or questions which have been rated with Likert responses. In either case, rankings are likely to have a maximum of 5 grades (or 7 for some Likert scales), but a ranking may also utilize fewer. For our purposes, we place no constraints on the number, size, or consistency of grades.

Partial rankings, compared to total orders, represent a loss of some of the information available in a ranking. This can be seen clearly by exploring the sufficient statistics for describing a ranking, the inversion matrix $Q(\pi)$.

Any total order π can be completely described by the inversion matrix $Q(\pi)$, but as previously noted only the lower triangle of $Q(\pi)$ is needed due to symmetry. For a total order of n items, the lower triangle of $Q(\pi)$ represents $\frac{n^2-n}{2}$ entries, one for each unique relation between items $e \neq e'$.

For a partial ranking $\sigma = (\mathcal{E}_1|\mathcal{E}_2|...|\mathcal{E}_K)$, any $e, e' \in \mathcal{E}_k$ have unknown (or marginalized) order, thus one entry in the inversion matrix is lost. For a group of size $|\mathcal{E}_k| = n_k$ the entire subset of the matrix corresponding to elements in $\mathcal{E}_k$ is missing, resulting in $\frac{n_k^2-n_k}{2}$ missing entries in the inversion matrix Q.

One can then get a measure of the percentage of data 'missingness' for a partial ranking $\sigma = (\mathcal{E}_1|\mathcal{E}_2|...|\mathcal{E}_K)$ with

$$M(\sigma) = \frac{\sum_k \frac{n_k^2+n_k}{2}}{\frac{n^2+n}{2}}. \tag{3.1}$$

3.2 Recursive Inversion Model for partial rankings

As noted, partial rankings display complete decomposability. This property allows us to decompose the likelihood of a partial ranking σ into a product of the likelihoods of interleavings at each node. The likelihood at each node then must account for all possible interleavings of elements within each grade.

Theorem 1 *Let $\sigma = (\mathcal{E}_1|\mathcal{E}_2|\dots|\mathcal{E}_K)$ be a partial ranking, and $\tau(\boldsymbol{\theta})$ a RIM model. Choose an internal node $\mathcal{I}_i$ of τ, and let $\mathcal{L}, \mathcal{R}$ be sets of items in the left and right subtrees of $\mathcal{I}_i$ (for simplicity, we drop the subscript i from the following equation). Denote $L_k = \mathcal{E}_k \cap \mathcal{L}$, $R_k = \mathcal{E}_k \cap \mathcal{R}$, $l_k = |L_k|, r_k = |R_k|, L = |\mathcal{L}|, R = |\mathcal{R}|$, $\bar{l} = (l_1, \dots l_K)$, $\bar{r} = (r_1, \dots r_K)$ and let θ be the parameter at node $\mathcal{I}_i$. Define*

$$g(\bar{l}, \bar{r}, \theta) \;=\; \frac{\prod_{k=1}^{K} Z_{l_k, r_k}(\theta)}{Z_{L,R}(\theta)} \theta^{\sum_{k=2}^{K} \sum_{k'<k} l_k r_{k'}}. \tag{3.2}$$

Then the likelihood of σ in the model $\tau(\boldsymbol{\theta})$ is

$$P_{\tau(\boldsymbol{\theta})}(\sigma) \;=\; \prod_{\mathcal{I}_i \in \mathcal{I}} g(\bar{l}^i, \bar{r}^i, \theta_i) \tag{3.3}$$

Sketch of Proof The term in the exponent of θ can be seen to replicate the observed portion of the vertex discrepancy v seen in equation (2.17), where items from $\mathcal{R}$ in $\mathcal{E}_k$ are inverted with items in $\mathcal{L}$ for all $\mathcal{E}_{k+i}, i > 0$. This produces a total of $r_k(l_{k+1} + ... + l_K)$ inversions. The Z_{l_k, r_k} terms represent the normalization constant for an interleaving of l_k and r_k items, thus providing the marginalization of all possible interleavings of the items in $\mathcal{E}_k$ at the node. Expanding, for clarity, all the terms of Equation 3.2 yields

$$g(\bar{l}, \bar{r}, \theta) \;=\; \frac{Z_{l_1,r_1}(\theta)\theta^{(l_2+\ldots l_K)r_1} Z_{l_2,r_2}(\theta)\theta^{(l_3+\ldots l_K)r_2} \ldots Z_{l_{K-1},r_{K-1}}(\theta)\theta^{l_K r_{K-1}} Z_{l_K,r_K}}{Z_{L,R}(\theta)}. \qquad (3.4)$$

□

Excluding for now the exponent on the θ term in equation 3.2, we can see that we introduce K new terms to the node level likelihood when working with partial rankings in comparison to total orders. As was previously noted, the $Z_{L,R}$ term in the denominator is computed recursively meaning, with proper caching, the Z_{l_i,r_i} terms in the likelihood only represent at most K additional computations.

This overhead can be further reduced by noting three facts. The first is that all $Z_{l,0}$ and $Z_{0,r}$ are trivially equal to 1 and can thus be ignored. Likewise, due to symmetry $Z_{l,r}(\theta) = Z_{r,l}(\theta)$. Lastly, any time $l_i = l_j, r_i = r_j, i \neq j$, the term $Z_{l_i,r_i} Z_{l_j,r_j}$ can simplify to $2Z_{l_j,r_j}$. While this presents minimal to no improvement for a single ranking, for a set of N rankings we will show that only the count of the number of (l, r) pairs is needed to reduce the likelihood of the set to a constant and a single $Z_{l,r}$ term.

Combining these facts together, for a large sample of partial rankings, the top level node introduces no more than $(n-1)^2/2$ $Z_{l,r}$ terms in the numerator. Given that there are $n-2$ nodes below the root node, and that each would have less items than the root node. This implies, given all relevant quantities, at most an $\mathcal{O}(n^3)$ computational overhead for calculating the likelihood for partial rankings compared to total orders.

3.3 Conditional rank and inversion matrices

There are two common methods of describing total orders in more mathematically useful forms. The first we have seen, in the form of the inversion matrix $Q(\pi)$, whose indices $e, e' \in \mathcal{E}$ indicate whether or not $e \succ_\pi e'$. The second method is the rank matrix, $P(\pi)$. Here the indices of the matrix are denoted by e, j, with $e \in \mathcal{E}$ and $j \in [1, n]$. As one might assume from the name, the rank matrix indicates the rank of each element in $\mathcal{E}$. For a total order

$\pi \in \mathbb{S}_n$,

$$P_{e,j}(\pi) \;=\; [P_{ej}]_{e\in\mathcal{E},j\in[1,n]}, \quad P_{ej} = 1 \text{ if } \pi^{-1}(j) = e \text{ and 0 otherwise, for all } e, \in \mathcal{E}, j \in [1,n]. \tag{3.5}$$

As we have shown, partial rankings represent a loss of information in the inversion matrix, with some inversions between elements within a grade $\mathcal{E}_k$ being unknown. The rank matrix $P(\sigma)$ is similarly missing entries, with the elements of $\mathcal{E}_k$ with $|\mathcal{E}_k| > 1$ covering the ranks $[1 + \sum_{k'<k} n'_k, \sum_{k'\leq k} n'_k]$, but the specific rank of each item $e \in \mathcal{E}_k$ unknown.

One can then ask of both of these descriptive methods: given a model $\tau(\boldsymbol{\theta})$ and a partial ranking σ, what is the conditional distribution of $Q(\sigma)$ and $P(\sigma)$ implied by $\tau(\boldsymbol{\theta})$? In the context of the inversion matrix, we wish to know for a pair of equally ranked items $e, e' \in \mathcal{E}_k$ what the probability that $e \succ e'$ is under $\tau(\boldsymbol{\theta})$. Likewise for a $e \in \mathcal{E}_k$ covering ranks $[j, j+n_k]$, we wish to find the probability that item e takes rank $j' \in [j, j+n_k]$.

Here we explore a recursive method of calculating these conditional probabilities. We again take advantage of complete decomposability, building the conditional distributions over subtrees of the model until arriving at the final conditional distributions at the root.

Let us first define some notation. For any partial ranking σ and model $\tau(\boldsymbol{\theta})$ we denote by $Q_{ee'}(\sigma, \tau(\boldsymbol{\theta}))$ the probability $P_{\tau(\boldsymbol{\theta})}[e \succ e' \,|\, \sigma]$. $Q_{ee'}(\sigma, \tau(\boldsymbol{\theta}))$ represents the probability of $e \succ e'$ under model $\tau(\boldsymbol{\theta})$, conditioned on the partial information in σ. We call the matrix of these probabilities the *conditional inversion* matrix $Q(\sigma, \tau(\boldsymbol{\theta})) = [Q_{ee'}(\sigma, \tau(\boldsymbol{\theta}))]_{e,e'\in\mathcal{E}}$.

We now define by $P_{ej}(\sigma, \tau(\boldsymbol{\theta}))$ the conditional probability that item $e \in \mathcal{E}$ has rank j, given model $\tau(\boldsymbol{\theta})$ and partial ranking σ. More specifically, $P_{ej}(\sigma, \tau(\boldsymbol{\theta})) = P_{\tau(\boldsymbol{\theta})}[\text{rank } e \text{ is } j \,|\, \sigma]$. We call the matrix of these probabilities the *rank distribution* matrix $P(\sigma, \tau(\boldsymbol{\theta})) = [P_{ej}(\sigma, \tau(\boldsymbol{\theta}))]_{e\in\mathcal{E}}^{j=1:n}$.

The algorithms we present, FillPQand MarginalSubsetrely on the following facts noted in Proposition 2.

Proposition 2 *Let $\tau(\boldsymbol{\theta})$ be a RIM over $\mathcal{E}$, and $\sigma = (\mathcal{E}_1|\ldots|\mathcal{E}_K)$ be a partial ranking of $\mathcal{E}$, and let $P(\sigma, \tau(\boldsymbol{\theta}))$, $Q(\sigma, \tau(\boldsymbol{\theta}))$ be defined as above. Let $m_k = \sum_{k'<k} n_{k'}$, and let $\pi \in \mathbb{S}_n$*

represent any total order compatible with σ. Where necessary, the $(\sigma, \tau(\boldsymbol{\theta}))$ of $P(\sigma, \tau(\boldsymbol{\theta}))$ or $Q(\sigma, \tau(\boldsymbol{\theta}))$ is dropped for readability. Then,

1. *For any $\mathcal{E}_k = \{e\}$, e takes rank $m_k + 1$, defining the row $P_{e\cdot}$ and column $P_{\cdot m_k+1}$ of $P(\sigma, \tau(\boldsymbol{\theta}))$, and defining the row $Q_{e\cdot}$ and column $Q_{\cdot e}$ of $Q(\sigma, \tau(\boldsymbol{\theta}))$.*

2. *Let $m_k = \sum_{k'<k} n_{k'}$. For all $e \in \mathcal{E}_k$, and for any $\pi \in \mathbb{S}_n$ compatible with σ, the rank of e takes values in $m_k + 1 : m_k + n_k$.*

3. *The only undetermined entries in $P(\sigma, \tau(\boldsymbol{\theta}))$ are in the blocks $P_{\mathcal{E}_k, m_k+1:m_k+n_k}$, for each non-trivial grade $\mathcal{E}_k$.*

4. *For all $e \in \mathcal{E}_k$ and $e \in \mathcal{E}_{k'}$ with $k < k'$, $Q_{ee'} = 1 = 1 - Q_{e'e}$*

5. *The only undetermined entries in $Q(\sigma, \tau(\boldsymbol{\theta}))$ are in blocks $Q_{\mathcal{E}_k, \mathcal{E}_k}$ for each non-trivial grade $\mathcal{E}_k$.*

6. *Conditioned on σ, the order over the elements of $\mathcal{E}_k$ are independent on the order of elements of $\mathcal{E}_{k'}$ for any $k \neq k'$. Therefore, $P_{\mathcal{E}_k, m_k+1:m_k+n_k}$ and Q_{E_k, E_k} can be computed by the* MARGINALS *algorithm on a RIM $\tau(\boldsymbol{\theta})|_{\mathcal{E}_k}$. $\tau(\boldsymbol{\theta})|_{\mathcal{E}_k}$ can be constructed from $\tau(\boldsymbol{\theta})$ by replacing all nodes $\mathcal{I}_i$ for which $\mathcal{L}_i \cap \mathcal{E}_k = \emptyset$ or $\mathcal{R}_i \cap \mathcal{E}_k = \emptyset$ with the child containing items from $\mathcal{E}_k$, or a dummy leaf if neither child qualifies.*

Statements 1–5 are straightforward from relevant definitions. In Statement 6, the independence follows from the likelihood factorization proved by [23]. At any node $\mathcal{I}_i$ of $\tau(\boldsymbol{\theta})$ where $\mathcal{L}_i \cap \mathcal{E}_k = \emptyset$ (or $\mathcal{R}_i \cap \mathcal{E}_k = \emptyset$), the rank and relative order of the elements of $\mathcal{E}_k$ are unchanged. Hence, the node can be removed from $\tau(\boldsymbol{\theta})$. If the other child of $\mathcal{I}_i$ contains items from $\mathcal{E}_k$, this child replace the node $\mathcal{I}_i$. □.

Based on these remarks, the algorithm FILLPQ below first fills in the deterministic parts of $P(\sigma, \tau(\boldsymbol{\theta}))$ and $Q(\sigma, \tau(\boldsymbol{\theta}))$ given by σ, then for each non-trivial $\mathcal{E}_k$ it completes $P_{\mathcal{E}_k, m_k+1:m_k+n_k}$ and $Q_{\mathcal{E}_k, \mathcal{E}_{k'}}$. Instead of recomputing $\tau(\boldsymbol{\theta})|_{\mathcal{E}_k}$, MARGINALSUBSET always uses

the original model $\tau(\boldsymbol{\theta})$, but does not do any operation at an eliminated node. This avoids redundant computations, copying or setting up data structures, with only minimal overhead in the depth of the recursion tree.

Algorithm 2 Algorithm FillPQ

Input Model $\tau(\boldsymbol{\theta})$, partial ranking $\sigma = (\mathcal{E}_1|\mathcal{E}_2|\dots|\mathcal{E}_K)$.
Initialize $Q, P \in \mathbb{R}^{n\times n}$
for $k, k' = 1, \dots K$, $k < k'$ **do**
  for $e \in \mathcal{E}_k$, $e' \in \mathcal{E}_{k'}$ **do** {set deterministic portions of Q}
    $Q_{ee'} \leftarrow 1$, $Q_{e'e} \leftarrow 0$
$m_1 \leftarrow 0$
for $k = 1, \dots K$ **do**
  if $\mathcal{E}_k = \{e\}$ **then** {set deterministic portions of P}
    $P_{e,m_k+1} \leftarrow 1$, $P_{ej} \leftarrow 0$ for all $j \neq m_k + 1$
  else
    $P_{\mathcal{E}_k, m_k+1:m_k+n_k} \leftarrow$MarginalSubset$(root, \mathcal{E}_k)$
  $m_{k+1} \leftarrow m_k + n_k$
Output P, Q

The recursive algorithm MarginalSubset is given in Algorithm 3. The algorithm takes as input a node $\mathcal{I}_i$ and a set of items $\mathcal{E}' \subseteq \mathcal{E}$. It is assumed that a RIM model is given, that $\mathcal{I}_i$ is a node in this RIM, and that $\mathcal{E}'$ is the set of leaves under $\mathcal{I}_i$.

Algorithm 3, MarginalSubset, is called recursively starting from the leaves of $\tau(\boldsymbol{\theta})$. At each node $\mathcal{I}_i \in \mathcal{I}$, MarginalSubset computes the block of Q (and the intermediate term P^i) corresponding to the elements in the subtree rooted at $\mathcal{I}_i$. These are obtained from the corresponding blocks $Q_{\mathcal{E}_{i_L}}, Q_{\mathcal{E}_{i_R}}$ (and intermediary P^{i_L}, P^{i_R}) of its children nodes and the output of InterleavProb, found in Algorithm 4, which marginalizes locally over all the interleavings at node $\mathcal{I}_i$; InterleavProb, in turn, uses the output Z of the VCDF algorithm found in Algorithm 1.

The last steps of MarginalSubset implement the non-trivial part of the algorithm, i.e. the case when both $\mathcal{L}_i$ and $\mathcal{R}_i$ intersect with the current $\mathcal{E}_k$. In this case, we need to consider the distribution over all possible interleavings of the left and right items, i.e. of $\mathcal{L}_i$ and $\mathcal{R}_i$, and to calculate the induced distributions over ranks $\tilde{P}^i$, and over inversions $\tilde{Q}^i$. The

Algorithm 3 Algorithm MarginalSubset

Input node $\mathcal{I}_i$, set $\mathcal{E}' \neq \emptyset$
if i is a leaf **then**
  $P^i = [1]$
  Return P^i
if $\mathcal{E}' \cap \mathcal{L}_i \neq \emptyset$ **then** {fills Q^{i_L}, returns P^{i_L}}
  $L_i = |\mathcal{E}' \cap \mathcal{L}_i|$
  $P^{i_L} \leftarrow$MarginalSubset$(i_L, \mathcal{E} \cap \mathcal{L}_i)$
if $\mathcal{E}' \cap \mathcal{R}_i \neq \emptyset$ **then** {fills Q^{i_R}, returns P^{i_R}}
  $R_i = |\mathcal{E} \cap \mathcal{R}_i|$
  $P^{i_R} \leftarrow$MarginalSubset(i_R)
if $L_i = 0$ or $R_i = 0$ **then**
  Return P^{i_L} or P^{i_R}, whichever is non-empty
else
  $[Z^i] \leftarrow \text{VCDF}(L_i, R_i, \theta_i)$
  $\tilde{P}^i, \tilde{Q}^i \leftarrow$ InterleavProb(L_i, R_i, θ_i)
  for $e \in \mathcal{L}_i$, $e' \in \mathcal{R}_i$ **do**
    $Q_{ee'} \leftarrow \sum_{l=1}^{L_i} \sum_{r=1}^{R_i} P^L_{el} P^R_{e'r} \tilde{Q}_{lr}$
    $Q_{e'e} \leftarrow 1 - Q_{ee'}$
  $P^i \leftarrow \begin{bmatrix} P^{i_L} & 0 \\ 0 & P^{i_R} \end{bmatrix} \tilde{P}^i$

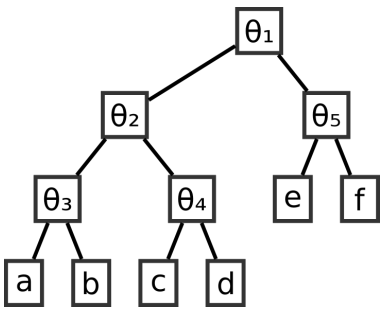
InterleavProb algorithm was introduced by [35] and is reproduced here for completeness. The computation of P^i, Q^i essentially convolves the rank (or inversion) distributions in the left and right children before interleaving with the interleaving marginals $\tilde{P}^i, \tilde{Q}^i$.

Algorithm 4 Algorithm InterleavProb

Input θ, $L \times R$, matrix Z obtained from VCDF(L, R, θ)
Initialize $\tilde{P}_{lj} \leftarrow 0$ for $j < l, j > l + R$, $\tilde{P}_{L+r,j} \leftarrow 0$ for $j < r, j > L + r$
for $l = 1, \ldots L$ **do**
  for $r = 0, \ldots R$ **do**
    $\tilde{P}_{l,l+r} \leftarrow \theta^{r(L-(l-1))} Z_{l-1,r} Z_{L-l,R-r} / Z_{L,R}$
for $r = 1, \ldots R$ **do**
  for $l = 0, \ldots L$ **do**
    $\tilde{P}_{l,l+r} \leftarrow \theta^{rl} Z_{L-l,r-1} Z_{l,R-r} / Z_{L,R}$
for $l = 1, \ldots L$ **do**
  $\tilde{Q}_{l,1} \leftarrow \tilde{P}_{l,l}$
  for $r = 2, \ldots R$ **do**
    $\tilde{Q}_{l,r} \leftarrow \tilde{Q}_{l,r-1} + \tilde{P}_{l,l+r-1}$
Output $\tilde{Q}$, $\tilde{P}$

3.3.1 Marginal rank and inversion matrices

If one is interested in the marginal distribution of the rank or inversion matrices, the same algorithms can be applied. For a partial ranking $\sigma = (\mathcal{E})$ demonstrating no preferences, the conditional distributions will be the *rank marginal* and *inversion marginal* matrices of $\tau(\boldsymbol{\theta})$, denoted as $P(\tau(\boldsymbol{\theta}))$ and $Q(\tau(\boldsymbol{\theta}))$. A simplified version of the MarginalSubset algorithm, simply named Marginals, was included in [35] that computes $P(\tau(\boldsymbol{\theta}))$ and $Q(\tau(\boldsymbol{\theta}))$ simultaneously, by a recursion starting from the leaves and working up the tree in post-order. As this represents a simpler, special case of the MarginalSubset algorithm (with $\sigma = (\mathcal{E})$), we do not repeat it here.



Original Q

	a	b	c	d	e	f
a	–	?	0	0	?	0
b		–	0	0	?	0
c			–	1	1	?
d				–	1	0
e					–	0
f						–

Original P

	1	2	3	4	5	6
a	0	0	0	?	?	?
b	0	0	0	?	?	?
c	?	?	0	0	0	0
d	0	0	1	0	0	0
e	0	0	0	?	?	?
f	?	?	0	0	0	0

Figure 3.1: **Left** a model structure $\tau(\boldsymbol{\theta}) = (((a, b, \theta_3), (c, d, \theta_4), \theta_2), (e, f, \theta_5), \theta_1)$. In the **center** and on the **right** are the deterministic portions of the inversion and rank matrices, respectively, for $\sigma = (c, f|d|a, b, e)$. The entries of either matrix not determined by σ are marked with "?", and the complementary lower triangle of Q is ignored.

3.3.2 FillPQ Example

We will consider the model $\tau(\boldsymbol{\theta})$ displayed in Figure 3.1 and the partial ranking $\sigma = (c, f|d|a, b, e)$; hence $K = 3$, $\mathcal{E}_1 = \{c, f\}$, $\mathcal{E}_2 = \{d\}$, $\mathcal{E}_3 = \{a, b, e\}$. We set $\theta_1 = 1$, $\theta_2 = 1/3$, $\theta_3 = 1/2$, $\theta_4 = \theta_5 = 1$. As will be seen, only θ_1, θ_2, and θ_3 are needed in calculating the conditional distributions. We will represent by P^i, $\tilde{P}^i$, and $\tilde{Q}^i$ the relevant intermediary terms at node $\mathcal{I}_i \in \mathcal{I}$, dropping $(\sigma|\tau(\boldsymbol{\theta}))$ in all places for clarity.

We traverse the tree in post order, starting with node $\mathcal{I}_3$. Immediately we note that in σ, we do not observe the preference between items a and b. Calculation of P^3, the rank marginal of node $\mathcal{I}_3$, will require we traverse to the root node of the tree. The rank marginals for all leaf nodes are equal to a 1×1 matrix containing a singular value of 1. We calculate $\tilde{P}^3$ and $\tilde{Q}^3$ according to Algorithm 4 parameterized by θ_3.

$$\tilde{P}^3 = \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix} \qquad \tilde{Q}^3 = \begin{bmatrix} 2/3 \end{bmatrix} \tag{3.6}$$

We can then calculate the node's marginal rank matrix P^3 as well as entry a, b of the matrix

Q.

$$P^3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix} = \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix} \quad Q_{(\{a\},\{b\})} = [1][2/3][1] = [2/3] \tag{3.7}$$

At node $\mathcal{I}_4$ we note that the relative ranking of c and d is known, thus no calculation is required. Similarly at node $\mathcal{I}_2$ there are no unobserved entries in the inversion matrix, hence

$Q_{(\{a,b,c,d\},\{a,b,c,d\})}$

	a	b	c	d
a	$-$	$2/3$	0	0
b		$-$	0	0
c			$-$	1
d				$-$

P^2

	1	2	3	4
a	0	0	$2/3$	$1/3$
b	0	0	$1/3$	$2/3$
c	1	0	0	0
d	0	1	0	0

(3.8)

Similarly at node $\mathcal{I}_5$ we know the order of items e and f, thus there are no unobserved entries in the inversion matrix or in P^5. At the root node of the tree, we have unobserved entries for the rank and inversion between c and f, the rank of a, b, and e, and the inversion between a and e, and b and e.

It is worth noting that we can resolve the remainder of the missing marginal information with a single application of Algorithm 2 and Algorithm 4, but from both a computational and exposition standpoint it is easier to view each grade $\mathcal{E}_k$ as its own calculation. With a parameter of $\theta_1 = 1$, we expect no preference of ordering for items on the left and right branches. We will start with the rank and inversion of c and f.

$$\tilde{P}^1_{c,f} = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} \qquad \tilde{Q}^1_{(\{c\},\{f\})} = \begin{bmatrix} 1/2 \end{bmatrix} \tag{3.9}$$

$$P^1_{c,f} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} \tag{3.10}$$

$$Q_{(\{c\},\{f\})} = [1][1/2][1] = [1/2] \tag{3.11}$$

While this calculation is unsurprising, the case for items a, b, and e proves more interesting.

$$\tilde{P}^1_{a,b,e} = \begin{bmatrix} 2/3 & 1/3 & 0 \\ 0 & 1/3 & 2/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \qquad \tilde{Q}^1_{(\{a,b\},\{e\})} = \begin{bmatrix} 2/3 \\ 1/3 \end{bmatrix} \tag{3.12}$$

Note that the 0 values in $\tilde{P}$ correspond to the fact that the first element on the left cannot occupy the last index, and similarly the second item from the left cannot occupy the first index. The ordering of these items (in this case a and b) is determined at node $\mathcal{I}_3$, as expressed by P^3.

$$P^1_{a,b,e} = \begin{bmatrix} 2/3 & 1/3 & 0 \\ 1/3 & 2/3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2/3 & 1/3 & 0 \\ 0 & 1/3 & 2/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} = \begin{bmatrix} 4/9 & 3/9 & 2/9 \\ 2/9 & 3/9 & 4/9 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \tag{3.13}$$

$$Q_{(\{a,b\},\{e\})} = \begin{bmatrix} 2/3 & 1/3 \\ 1/3 & 2/3 \end{bmatrix} \begin{bmatrix} 2/3 \\ 1/3 \end{bmatrix} [1] = \begin{bmatrix} 5/9 \\ 4/9 \end{bmatrix} \tag{3.14}$$

Despite $\theta_1 = 1$ representing no preference between items on the left and right, we note that e is more likely to be inverted with b than with a due to the fact that we are more likely to have a precede b. We can now fill the remaining items of Q and the calculated marginal values of the rank distribution matrix P.

$$Q = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline a & - & 2/3 & 0 & 0 & 5/9 & 0 \\ b & & - & 0 & 0 & 4/9 & 0 \\ c & & & - & 1 & 1 & 1/2 \\ d & & & & - & 1 & 0 \\ e & & & & & - & 0 \\ f & & & & & & - \end{array} \qquad P = \begin{array}{c|cccccc} & 1 & 2 & 3 & 4 & 5 & 6 \\ \hline a & 0 & 0 & 0 & 4/9 & 3/9 & 2/9 \\ b & 0 & 0 & 0 & 2/9 & 3/9 & 4/9 \\ c & 1/2 & 1/2 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 1 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\ f & 1/2 & 1/2 & 0 & 0 & 0 & 0 \end{array} \tag{3.15}$$

3.4 Maximum likelihood estimation for RIMs from partial ranking

The task of maximum likelihood estimation for partial rankings follows closely the same process originally developed for fitting RIMs [30]. The process is broken down into steps - given a reference ranking π_τ, a dynamic programming algorithm attempts to fit every possible structure consistent with π_τ. From this, the best possible structure is saved as the best model. From here, a new value of π_τ is sampled and the process repeated.

With inclusion of partial rankings, the task changes little. A more thorough local search was added that results in easy gains in the likelihood when possible. Otherwise, compared with the method as previously developed, this work goes into detail about the recommended steps for fast estimation with large samples of partial rankings.

3.4.1 Log-likelihood decomposition

We now turn to the problem of finding the Maximum Likelihood RIM given a dataset consisting of partial ranking observations $\mathcal{D} = \{\sigma_1, \dots \sigma_N\}$. We assume that the data were generated as follows. For $i = 1, \dots N$, a complete ranking π_j was sampled from a fixed, unknown distribution $P_{\tau(\boldsymbol{\theta})}$; then σ_j is obtained from π_j following a shape $(n_1, \dots, n_{K_j})$ given by some other (observed) process. We refer to this process as *partialization* and assume independence between partialization and the ranking, as well as independence between par-

tialization of each ranking. Importantly, this implies that the shapes are independent of the complete data whose distribution we are aiming to estimate.

In previous work [30], the ML estimation problem was studied for total orders. It was demonstrated that the estimation of $\boldsymbol{\theta}$ given a structure τ consists of independent, one-dimensional convex optimization problems. Estimation of the structure τ for a specific reference permutation π_τ can be done in polynomial time using the dynamic programming algorithm StructByDP.

Here, we demonstrate that all the tractable steps required for ML estimation for total orders can be performed in polynomial time for partial rankings as well, despite the fact that a partial ranking σ stands for an exponentially large set of permutations. The results hinge on Theorem 1, which demonstrates that we can compute probabilities for partial rankings in polynomial time.

These methods do not in any way constrain the structure of partial rankings, and rankings can be similar (such as all top-t rankings) or of arbitrary relative shape. In some places in the ML estimation process, similarities between rankings can be exploited to reduce computational overhead. Where relevant, we comment on these options, but do not go into detail on their implementation. Even without utilizing specific advantages, the algorithms will not incur more overhead than a low degree polynomial in the sample size N, the number of items ranked n, and the number of groups per ranking K.

According to the Maximum Likelihood (ML) paradigm, we propose to find the model τ and parameters $\boldsymbol{\theta}$ that maximize the data log-likelihood

$$\begin{aligned}\frac{1}{N}\sum_{j=1}^{N}\ln P_{\tau,\theta}(\sigma_i) &= \sum_{i\in\mathcal{I}}\underbrace{\tfrac{1}{N}\sum_{j=1}^{N}\ln g(\bar{l}^i(\sigma_j,\tau),\bar{r}^i(\sigma_j,\tau),\theta_i)}_{\texttt{score}_i} \\ &= \sum_{i\in\mathcal{I}}\left[\left(\tfrac{1}{N}\sum_{j=1}^{N}v_i(\sigma_j,\tau)\right)\ln\theta_i + \tfrac{1}{N}\sum_{j=1}^{N}\sum_{k=1}^{K_j}\ln Z_{l^i_k(\sigma_j,\tau),r^i_k(\sigma_j,\tau)}(\theta_i) - \ln Z_{L_i,R_i}(\theta_i)\right]\end{aligned} \tag{3.16}$$

In the above, the score for node $\mathcal{I}_i$ decomposes into three terms. The first term depends

only on the observed inversions in the data. For this term, the data can be summarized into a single discrepancy matrix Q, independently of the model τ, which can be precomputed and used to calculate the necessary terms. The third term is the normalization constant for node $\mathcal{I}_i$, which is independent of the amount of missing information in the data. Finally, the second term is the log-likelihood of the unobserved inversions; this term depends on the structure of the current model τ and must be recomputed every time τ changes, by iterating over the dataset. This is the only part of the log-likelihood for which sufficient statistics are not available.

In the upcoming sections, we show how to carefully organize the data to obtain efficient algorithms for each of the estimation steps, and we analyze their overhead in storage and run time w.r.t. the same estimation steps for total orders. Similarly to [30], we present algorithmic components that break down into separate steps; estimation of $\boldsymbol{\theta}$ parameters given τ, estimation of τ given π_τ, and end with putting them together into a stochastic search algorithm dubbed SASEARCH that explores the space of potential π_τ.

3.4.2 ML estimation of the $\boldsymbol{\theta}$ parameters

From Theorem 1 it is easy to see that the ML estimate of each θ_i depends only on statistics available at node $\mathcal{I}_i$ of a tree τ, namely `LR` calculated from the vectors in `G`, and `V` calculated from `Q`. Therefore, w.l.o.g., we will consider only the root node. All the results immediately apply to any other node in the tree, by replacing $\mathcal{E}$ with the subset of items under that node.

In the total order case, the ML estimate $\vec{\theta}$ is obtained by directly maximizing the univariate and concave log-likelihood function w.r.t θ [30]. Now, we ascertain that the current log-likelihood is also concave.

Theorem 3 *The function* $\ln g(\bar{l}, \bar{r}, \theta)$ *is concave in* θ.

This is true since g is the sum of the likelihoods of all permutations π consistent with a given σ; each π has a concave log-likelihood, hence $\ln g$ will be concave as well [6].

This means that maximizing the log-likelihood of θ_i at node $\mathcal{I}_i$ of a tree τ (for now we

assume the structure of τ is fixed) can be done by univariate gradient descent, just like in the case of total orders. We will consider now the amount of operations needed to compute the gradient of the log-likelihood (3.16) w.r.t. θ_i. We do not go into details, which are standard calculus, except for making two observations.

First, the gradient of the log-likelihood is a sum over N samples. In principle, computing it may take up to N times more than the computation of a single gradient, due to the lack of sufficient statistics. However, sufficient statistics exist at the node level that save considerable computation by tabulating the number of $Z_{l,r}$ factors that appear for each l, r combination from row sums of the matrix `G`, reducing the computation at a node to requiring $\mathcal{O}(LR)$ total operations after the calculation of `LR`, regardless of N.

Second, because we are able to calculate each $Z_{l,r}$ recursively (see VCDF), we can similarly use the chain rule to calculate derivatives of $\ln Z_{l,r}$ efficiently in a recursive manner. Note that the $Z_{L,R}$ represent normalization constants in the fully observed model; computing the derivatives of $\ln Z_{L,R}$ w.r.t. θ produces the derivatives of all necessary $Z_{l,r}$ factors as intermediate results. Hence, the overhead of this optimization step when working with partial ranking data is negligible.

3.4.3 Maximizing the log-likelihood over (sub)trees $\tau(\vec{\theta})$

Here, it is assumed that a reference permutation π_τ is given. The task is to find the tree structure τ of maximum likelihood, compatible with π_τ. As seen in Theorem 1 the likelihood of any subtree of a RIM is a product of factors corresponding to the nodes in the subtree. From this factorization, it follows that one can maximize the likelihood over structures recursively over subtrees, with a dynamic programming algorithm that closely follows the algorithm for total orders introduced by [30].

This is Algorithm 6, PartialStructByDP. Compared to the original algorithm found in [30], the differences are in the data structures `Q`, `G` and `LR`, as well as in the steps that use these, namely GetLR, and the `score` computation.

At the higher level, the Dynamic Programming algorithm can be described succinctly as

follows.

Starting with the leaves of the tree (with likelihood 1), we (trivially) find the most likely subtree for all size $n' = 2$ sets of consecutive items in π. Then recursively from the bottom of the tree up to the root, for each subset of n' items, we find the optimal θ for the $n' - 1$ possible splits of the items. The best of these is cached with the optimal θ and node level likelihood, to be used when evaluating potential subtrees on higher level nodes. Then, from the root down, we can assemble the optimal $\tau(\vec{\theta})$ using the best subtrees found for each node.

The algorithm stores, for each $l : m$ subtree, its `cost`, the index at which to split the children in `back`, the optimal $\texttt{theta}_{l,m}$, and whether we need to invert the children (i.e. if $\texttt{theta}_{l,m} > 1$) in `flip`; here `cost`, `theta`, `back`, `flip` are triangular arrays of size $n \times n$ containing respectively real, integer, and binary $\{0, 1\}$ values as needed.

To construct the optimal tree, we follow `back` recursively from the root node, using the split specified to find the best left and right nodes. If a node is discovered where `flip` is 1, we swap the children and invert the estimated parameter following the steps for canonicalization. Any tree that utilizes a node requiring canonicalization will produce a tree with a reference permutation that does not match the permutation used to fit the tree. As we will detail in Section 3.4.4, these cases can be utilized in the search for the optimal π_τ.

For example, consider a small tree of 4 items, under a reference permutation $\pi_\tau = (1, 2, 3, 4)$. At the root node, we have 3 possible splits, (1), (2,3,4); (1,2), (3,4); and (1,2,3), (4). We must calculate 3 different versions of `LR`, one for each split; we use the `G` vectors to enumerate the `LR` matrices, $\texttt{G}_{(1,1)}$ and $\texttt{G}_{(2,4)}$, $\texttt{G}_{(1,2)}$ and $\texttt{G}_{(3,4)}$, and lastly $\texttt{G}_{(1,3)}$ and $\texttt{G}_{(4,4)}$. These vectors are precomputed and stored before the Dynamic Programming is run.

The calculation of likelihood includes enumeration of $Z_{l,r}(\theta)$ for all $l \leq L, r \leq R$ at a given node. The algorithm GETLR parses through the vectors `G` associated with the left and right subtrees of a node. For a set of items $\mathcal{E}_L = \{e_{l_1}, e_{l_2}, .., e_{l_L}\}$ and $\mathcal{E}_R = \{e_{r_1}, e_{r_2}, .., e_{r_R}\}$, we calculate the matrix `LR` where $\texttt{LR}_{l,r} = \sum_{n=1}^{N} \sum_{\mathcal{E}_i \in \pi_n} 1_{[|\mathcal{E}_i \cap \mathcal{E}_L| = l, |\mathcal{E}_i \cap \mathcal{E}_R| = r]}$. This tells us the number of $Z_{l,r}$ terms needed for $l \in 1..L, r \in 1..R$. Note that the `LR` are computed only once during the run of the algorithm; thus they need not be cached.

Even with proper caching, the enumeration of LR still contributes significant calculation overhead to the optimal structure search. For a set of partial rankings, all observed ordering information is stored in the matrix Q, thus we need only concern ourselves with the unobserved orders, i.e., grades $\mathcal{E}_k$ with $|\mathcal{E}_k| > 1$. If there are a total of K_N such grades in all the partial rankings combined, calculating LR for each node requires $\mathcal{O}(K_N)$ total calculations. When searching for an optimal subtree, a node with n' elements will require $\mathcal{O}(K_N n')$ operations to enumerate all possible LR.

The calculation of V from Q is at worst $\mathcal{O}(n^2)$, but the calculation of V must only be performed once before estimating the optimum θ, which requires many more $\mathcal{O}(n^2)$ computations, as noted.

3.4.4 Reference ranking search

The search for the best reference ranking $\hat{\pi}_\tau$ is the only problem for which no tractable algorithm is known. We use a slightly modified version of the stochastic search algorithm, SASearch introduced by previous works [30].

The algorithm is initialized heuristically with a π determined by sorting the row sums of Q in increasing order. An initial model (and sampling model) is estimated, by maximizing the likelihood of $\tau(\vec{\theta})$ over all possible model structures consistent with π, using PartialStructByDP. It is at this point that a key modification was included to improve the model.

Empirically we found that search can be improved by adding one or more deterministic hill-climbing steps after a new $\tau(\vec{\theta})$ was fit to π. If the model requires canonicalization, we obtain a new π_τ that is different from π. Instead of sampling a new π' from $\tau(\vec{\theta})$, we call PartialStructByDP for π_τ. The new model is guaranteed to do as well (producing an identical model) or better than the previous model $\tau(\vec{\theta})$. This optimization step can be repeated until the model obtained requires no canonicalization.

After the hill-climbing steps have concluded, if the resulting model is the best model thus far, it is saved. If the model does better than the current sampling model or if a Metropolis-

Hastings like step passes (with probability inversely proportional to the loss in likelihood from the current sampling model) the resulting model replaces the sampling model.

From this point on, a new candidate π' is obtained by sampling from the current sampling model $\tau(\vec{\theta})$), evaluated by running PartialStructByDP, and accepting or rejecting π' as the best or sampling model as before. This is repeated for a predefined fixed number of steps. The best model found during search is returned at the end of the exploration. Note that the sampling of π' and the acceptance step are heuristic, as we are not trying to sample from any given distribution, but to *explore* the space efficiently, in order to find the optimal π_τ.

The name of the algorithm, SASearch, is reminiscent of simulated annealing, but in this simple version the "annealing" takes place at constant temperature. With the same algorithmic building blocks, other variants of stochastic search can be easily designed. The full algorithm is outlined in Algorithm 7.

3.4.5 Data structures for efficient computations on large samples

In this section, a variable in typescript font, like $\mathtt{Q}$, denotes an array data structure.

The partial inversion matrix $\mathtt{Q}$ As shown, when fitting a RIM we can reduce a set of N total orders into an aggregate inversion matrix Q. In the context of partial rankings, we define the *partial inversion matrix*, $\mathtt{Q}$, which counts at index e, e' the proportion of times item e is known to precede e', i.e. $e \in \mathcal{E}_k$, $e \in \mathcal{E}_{k'}$ and $k < k'$. Note that $\mathtt{Q}_{e,e'} + \mathtt{Q}_{e',e} \leq 1$. This matrix serves as a sufficient statistic for the terms of Equation 3.2 in the exponent of θ, or equivalently to the first term in (3.17).

Statistics dependent on the current π and τ With the exponential terms accounted for by the partial inversion matrix $\mathtt{Q}$, the remainder of Equation 3.2 requires knowing, at each node $\mathcal{I}_i$, how many instances of every possible (l_i, r_i) pair we have in the data. We store this information in the *pair counts* matrix $\mathtt{LR}_i$ of size $L_i \times R_i$. Note that unlike $\mathtt{Q}$, which is only

computed once, the pair counts matrices $\mathtt{LR}$ depend on the current structure τ, and must be recomputed every time a new τ is considered in the model search. We can however save time if we fix a permutation π and consider all structures τ consistent with π as a modal permutation.

For this we need an additional data structure, $\mathtt{G}$, which consists of $n(n+1)/2$ vectors of equal length. For a fixed π the item at rank m of π is associated with $\mathtt{G}_{m,m}$. If τ is a tree with reference permutation π, and $\mathcal{I}_i$ is an internal node of τ, the subtree under $\mathcal{I}_i$ will have leaves $l, l+1, \ldots m$ for some l, m with $1 \leq l < m \leq n$; thus we associate node $\mathcal{I}_i$ with $\mathtt{G}_{l,m}$.

We call $\mathcal{E}_k$ *trivial* if $n_k = |\mathcal{E}_k| = 1$. Obviously, $\mathtt{LR}_i$ needs to be computed only for non-trivial $\mathcal{E}_k$ grades, because for grades $\mathcal{E}_k$ with only one item all inversion information is stored in $\mathtt{Q}$. Consider partial ranking $\sigma_j = (\mathcal{E}_1|\mathcal{E}_2|...|\mathcal{E}_K)$. Let $K'_j = \sum_{k=1}^{K} 1_{|E_k|>1}$ be the number of non-trivial grades of σ_j, $j = 1, \ldots N$. The vector $\mathtt{G}_i$ has length $K_{tot} = \sum_{j=1}^{N} K'_j$; each entry in $\mathtt{G}_i$ is a count of how many items in $\mathcal{E}_k$ appear under node i, i.e. $|\mathcal{E}_k \cap \mathcal{E}_{l,m}|$. For the purposes of ML estimation, there is no need to differentiate $\mathcal{E}_k$ from different σ_j.

For a leaf node $e = \pi_\tau(m)$, the vector $\mathtt{G}_{m,m}$ is a binary vector indicating whether item e is in grade $\mathcal{E}_k$ of σ. The vector $\mathtt{G}_{l,m}$ for $l < m$ can be computed recursively by $\mathtt{G}_{l,m} = \sum_{k=l}^{m-1}(\mathtt{G}_{l,k} + \mathtt{G}_{k+1,m})$, a sum of vectors of length K_{tot}.

All vectors $\mathtt{G}_{l,m}$, for $l < m$ can be obtained recursively over $m - l$ in a total of $\mathcal{O}(n^3)$ operations. Given a tree structure τ compatible with σ, the vectors $\mathtt{G}_{l,m}$ corresponding to its internal nodes are among the comprehensive set of precomputed $\mathtt{G}$ vectors. This caching saves compute time if the goal is to optimize over tree structures τ.

Once $\mathtt{G}$ is obtained, the calculation of the necessary sufficient statistics $\mathtt{LR}$ is done as follows.

For a node $\mathcal{I}_i$ of some τ compatible with σ, $\mathcal{I}_i$ and its children are identified respectively with ranges $l : m$, with nodes $l : m'$ and $m' + 1, m$. Hence $\mathtt{LR}_i$ can be tabulated from the vectors $\mathtt{G}_{l,m'}$ and $\mathtt{G}_{m',m}$; $\mathtt{LR}_i$ is a 0 indexed matrix with shape $(L+1) \times (R+1)$. Algorithm 5, which we call GETLR, steps through vectors $\mathtt{G}_{l,m'}$ and $\mathtt{G}_{m'+1,m}$ while incrementing the corresponding $\mathtt{LR}$ element at each step.

Algorithm 6, dubbed PartialStructByDP, demonstrates how the G data structure is used. Note that for the likelihood computation of a single tree with known structure, there is no need nor benefit to this preprocessing.

Initialization and updates The length of the G vectors depends on the dataset only, hence these vectors can be pre-allocated, but they must be recomputed at every change of π_τ. We can pre-initialize the node level $\mathtt{G}_{m,m}$ values, which are independent of π_τ. When π_τ changes, these vectors must be re-indexed according to the new π_τ. The rows and columns of the partial observation matrix Q must also be reordered to agree with the new π_τ.

Computing the log-likelihood of a model $\tau(\vec{\theta})$ To calculate $\mathtt{score}_i$, the log-likelihood at a given node $\mathcal{I}_i \in \mathcal{I}$ for a model $\tau(\vec{\theta})$ given the sample $\mathcal{D}$, first a tabulation of all $Z_{l,r}(\theta)$ factors is obtained by VCDF, the LR matrix stores their counts, and the number of inversions v_i at each node $\mathcal{I}_i \in \mathcal{I}$ are calculated by summing the relevant indices of Q. Then the values of $\mathtt{score}_i$ for $\mathcal{I}_i \in \mathcal{I}$ can be calculated according to (3.17).

Note that the data structures Q and G contain more information than needed for a single model. In fact, they contain the sufficient information to optimize over all Recursive Inversion Models τ which have a given π_τ as reference permutation (including non-canonical models). This optimization is described in Section 3.4.3.

Algorithm 5 Algorithm getLR

Input G, item ranks l, m', m
$L \leftarrow m' - m$, $R \leftarrow m - m' - 1$
Initialize $\mathtt{LR} \leftarrow \mathbf{0}_{L+1,R+1}$ (with zero-based indexing)
for $k \in 1...K_{tot}$ **do**
  $++\mathtt{LR}_{\mathtt{G}_{l,m'}[k],\mathtt{G}_{m'+1,m}[k]}$
  Output $\mathtt{LR}_{l,m}$

Additional optimizations Though it was not done here, one could further speed up the evaluation of LR at each node, as follows, by taking into account that the same subset

Algorithm 6 Algorithm PartialStructByDP

Input $\mathtt{Q}$, $\mathtt{G}$
for $l \in 2...n$ **do**
  for $m \in 1..n-l$ **do**
    for $m' \in 2..l+1$ **do**
      $\mathtt{cost}(l,m) \leftarrow -\infty$
      calculate $\mathtt{V} = \sum_{l`=m}^{m+m'} \sum_{r`=m+m'}^{m+l+1} \mathtt{Q}_{l',r'}$
      $\mathtt{LR} \leftarrow \text{getLR}(\mathtt{G}, l, m'-1, m)$
      estimate $\mathtt{theta}(l,m)$ from $\mathtt{V}$ and $\mathtt{LR}$ according to Section 3.4.2
      calculate $\mathtt{score}(l,m)$ according to (3.17)
      $s \leftarrow \mathtt{cost}(m'-1,m) + \mathtt{cost}(l-m', m+m') - \mathtt{score}(l,m)$
      if $s < \mathtt{cost}(l,m)$ **then**
        $\mathtt{cost}(l,m) \leftarrow s$, $\mathtt{back}(l,m) \leftarrow m'$, $\mathtt{LR}(l,m) \leftarrow \mathtt{LR}(m'-1,l) + \mathtt{LR}(l-m', m+m')$
        $\mathtt{flip}(l,m) \leftarrow 1_{\theta_{l,m}>1}$; store $\theta_{l,m}$
Assemble τ from $\mathtt{back}, \mathtt{flip}$ and $\vec{\theta}$
Output $P(\mathtt{Q}, \mathtt{G}|\tau), \tau(\vec{\theta})$

Algorithm 7 Algorithm SASearch

Input Item set $\mathcal{E}$ of n items, N partial or complete rankings of $\mathcal{E}$
Initialize observed inversion matrix $\mathtt{Q}$ according to Section 3.4.5
Initialize $K_{tot} \times n(n+1)/2$ partial data matrix $\mathtt{G}$ according to Section 3.4.5
Initialize π from row sum of $\mathtt{Q}$, $P(\mathtt{Q}, \mathtt{G}|\tau_0), P(\mathtt{Q}, \mathtt{G}|\tau_{best}) \leftarrow 0$, $accept = FALSE$
for $t = 1, 2, ..., t_{max}$ **do**
  while $accept = FALSE$ **do**
    Create $\mathtt{Q}_\pi$ by reordering $\mathtt{Q}$ rows and columns consistent with π
    Reorder $\mathtt{G}_\mathcal{E}$ consistent with π
    Calculate all vectors in $\mathtt{G}_{l,m}$ recursively according to Section 3.4.5
    $P(\mathtt{Q}, \mathtt{G}|\tau'), \tau' \leftarrow$PartialStructByDP($\mathtt{Q}'_\pi$,$\mathtt{G}$)
    while $\pi \neq$ modal order of τ' **do** {local hill-climbing step}
      $\pi \leftarrow$ modal order of τ'
      Create $\mathtt{Q}_\pi$ by reordering $\mathtt{Q}$ consistent with π
      Calculate all vectors $\mathtt{G}_{l,m}$
      $P(\mathtt{Q}, \mathtt{G}|\tau'), \tau' \leftarrow$PartialStructByDP($\mathtt{Q}_\pi$,$\mathtt{G}$)
    $accept \leftarrow TRUE, u \sim uniform[0,1)$
    if $e^{-\beta(P(\mathtt{Q},\mathtt{G}|\tau_{t-1}) - P(\mathtt{Q},\mathtt{G}|\tau'))} < u$ **then** {annealing step}
      $accept \leftarrow FALSE$
  $\tau_t \leftarrow \tau'$
  if $P(\mathtt{Q}, \mathtt{G}|\tau_t) > P(\mathtt{Q}, \mathtt{G}|\tau_{best})$ **then**
    $\tau_{best} \leftarrow \tau_t$
Return τ_{best}

(representing $\mathcal{E}_k \cap \mathcal{L}_i$ or $\mathcal{E}_k \cap \mathcal{R}_i$) can appear multiple times in $\mathtt{G}_{k,l}$. For instance, take two partial rankings $\sigma = (...|e_1, e_2|...)$ and $\sigma' = (e_1, e_2|...)$. The entries in $\mathtt{G}$ corresponding to the grades $\mathcal{E}_k = (e_1, e_2)$ are identical for σ and σ'. By finding all instances of repeated grades of size 2 or greater, we can reduce each vector $\mathtt{G}_{k,l}$ of length K_{tot} to a vector of length K'_{tot}, where K'_{tot} is the number of *unique* groups $\mathcal{E}_k$ with sizes greater than 2.

This will necessitate the creation of a length K'_{tot} vector $\mathtt{C}$ matched to the grade indexing of $\mathtt{G}$ which counts the number of occurrences of each repeated group, and will be passed through the functions. Algorithmically very little changes, aside from the GETLR algorithm where, instead of incrementing $\mathtt{LR}_{\mathtt{G}^L[k'],\mathtt{G}^R[k']}$ by one, we increment it by $\mathtt{C}[\mathtt{k}']$, the number of copies of group k'. One could take this a step further, creating two versions of the vector $\mathtt{G}$, one which tracks the K'_{tot} non-unique groups that occur multiple times in $\mathcal{D}$, and one that tracks the remaining unique groups.

This preprocessing of the data will save time proportional to the total number of grade repetitions. Hence, the gains from this approach are largely dependent on the data. Instances with small n and many grades would benefit more, as large numbers of small grades are more likely to be repeated. In cases with large n or with grades consisting of many items, repetitions of a grade are less likely, thus speed up over the normal implementation detailed here will be small.

This process would also require considerable overhead at initialization, but would only need to be performed once for the entire dataset, and could subsequently be used to fit many different models to the data.

3.5 Experiments on synthetic data

We wish to confirm and test the methods developed for maximum likelihood estimation of RIMs on partial ranking data. To this end we sample total orders from known models, convert them into partial rankings, and attempt to recover the original models from the simulated data. This serves to confirm model recovery under stochasticity, and allows for estimation the needed runtime for recovering the model structure and reference ranking.

3.5.1 Goals

A number of different experiments were conducted to study the recoverability of various aspects of the RIM; the true parameter vector $\boldsymbol{\theta}$ which is subject to recovery of the true structure τ which in turn is subject to recovery of the true reference permutation π_0. While we do not detail the recovery of $\boldsymbol{\theta}$ (who's estimates $\hat{\theta}$ display mundane, normally distributed deviations), we do conduct a number of tests to study the recovery of both π_0 and τ.

The first was the recovery of structure. Given the correct reference ranking π_0, is the information present in the partial rankings (under stochasticity) sufficient to recover the original structure? To answer this question, models were initialized with a search starting at π_0; if the model produces another reference ranking π_τ, it means that the data lacks sufficient information to recover the true structure.

This process assumes that the correct reference ranking was already arrived at - a second set of tests seeks to resolve this issue by running the SASEARCH algorithm from a random initialization. This serves to test the recoverability of the reference ranking under different levels of partial rankings.

For these experiments, rankings were converted to partial rankings with varying levels of partialization. The goal of these experiments was to determine how many steps are needed to recover the true π_0, and to compare how both sample sizes and loss on information with increasing p_{merge} influences the number of steps needed to recover the tree for different model structures.

Each of these experiments under each of the different experimental conditions is repeated a total of 500 times.

3.5.2 Data generation

The first set of tests were conducted on a model with fixed structure and parameterization, with a balanced structure τ. The true model was

$$\tau_0 = ((0, ((1, 2|e^{-.2}), (3, 4|e^{-.4})|e^{-.6})|e^{-.8}), (((5, 6|e^{-.2}), (7, 8|e^{-.4})|e^{-.6}), 9|e^{-.8})|e^{-.5}). \quad (3.17)$$

To further test the SASEARCH algorithm seeks to measure the runtime. We test twos model structures τ_1 and τ_2 with $n = 10$ items, with randomized θ values sampled according to $\theta = e^{-u}, u \sim U(0.2, 1.2)$ with

$$\tau_1 = ((0, ((1, 2|\theta_4), (3, 4|\theta_b)|\theta_5)|\theta_2), (5, ((6, 7|\theta_8), (8, 9|\theta_9)|\theta_7)|\theta_6)|\theta_1); \quad (3.18)$$

$$\tau_2 = (((((0, 1|\theta_5), (2, 3|\theta_6)|\theta_4), (4, 5|\theta_7)|\theta_3), (6, 7|\theta_8)|\theta_2), (8, 9|\theta_9)|\theta_1). \quad (3.19)$$

Samples sizes varied between $N_{train} = 200$ and $N_{train} = 5000$. We are interested in seeing how well the algorithm can recover the reference permutation, as well as recovering the true structure and parameters.

To generate partial rankings, total orders are sampled from a RIM and elements of the total order are merged together into grades $\mathcal{E}_k$ following one of two processes. We refer to the process of converting total orders into partial rankings as *partialization*. Two different methods of partialization were utilized to explore recoverability of the model and allow for analyzing how missing data in the observed inversion matrix Q influences the number of steps needed to complete the SASEARCH portion of the model fitting.

The first approach for producing partial rankings from sampled total orders was to merge each e_i and e_{i+1}, for $i = 1, \ldots n-1$, into the same grade with probability p_{merge}. This produces $\mathcal{E}_i$ with sizes approximately following a geometric distribution, with range on $\{1, ..., n - \sum_{j<i} |\mathcal{E}_j|\}$. We are interested in looking at how increasing probability of merging items together impacts model recoverability for varying sample sizes of partial rankings. We will refer to this specific approach to producing partial rankings as *geometric partialization*.

Each of the models used maintains a constant $n = 10$ items, which allows estimating how partialization reduces the information from total orders. For this sample size, missingness of

data caused by geometric partialization was approximately 30.3% on average for $p_{merge} = .45$, jumping to approximately 58.4% on average for $p_{merge} = .8$ (with approximately 13.4% producing a single grade; useless data), and 86.4% on average for $p_{merge} = .95$ (with 63% producing useless single grade data). Cases with a single grade have likelihood 1 on any model, thus effectively reducing the sample size.

The second approach used to produce partial rankings was to a fixed shape $(n_1, \ldots n_K)$. For example, a top-5 sample of $n = 10$ items in this notation has shape $(1, 1, 1, 1, 1, 5)$. The shapes used were $(1, 3, 1, 5)$ ($\sim$42% of values in Q removed), $(1, 4, 4, 1)$ (40%), $(1, 2, 3, 4)$ ($\sim$36%), $(2, 2, 2, 2, 2)$ ($\sim$27%), and $(1, 2, 2, 2, 2, 1)$ ($\sim$25%).

3.5.3 Results - recoverability

The results of the first set of synthetic experiments are displayed in Figure 3.2, broken down into the recoverability of π_0 and the recoverability of τ for each partialization technique. This allows us to see how the loss of information affects our ability to recover the permutation and structure from the data.

The case where $p_{merge} = 0$ represents the situation where we are dealing with total orders. We see in the top left plot of Figure 3.2 that we were able to reliably recover the true ranking in almost all 500 cases with sample sizes around $N = 500$ or $N = 1000$. We see very little loss of recoverability until p_{merge} approaches 50%, with the falloff becoming more and more extreme as p approaches 1. Naturally, one can assume that as p_{merge} approaches 1 in limit, recoverability approaches 0 for all sample sizes, as observations would converge to the shape $(\mathcal{E}_1)$, $|\mathcal{E}_1| = n$ and every model becomes equally likely.

For those samples for which the true ranking π_0 was recoverable, the top right plot of Figure 3.2 demonstrates that reliable structure recovery requires much larger sample sizes. Even looking at total orders ($p_{merge} = 0$)with samples of size $N = 5000$, the data yielded a structure with better likelihood than the true structure for approximately 10% of samples that produced the correct reference permutation. Again, we see recoverability drop more rapidly as p_{merge} increases, with almost zero structure recovery when $p_{merge} = .95$ even at

sample size $N = 5000$.

The bottom left plot in Figure 3.2 shows the recoverability of π_0 under various structured partializations, though contrary to the geometric partialization, there is no clear ordering to determine which shape should produce better recoverability. For example, (1,3,1,5) and (1,4,4,1) both have 4 grades, both with the same average grade size, and both have similar levels of data missingness, yet perform quite differently from one another.

We see a bit more interesting trends looking at the structure recoverability in Figure 3.2. Again comparing (1,3,1,5) and (1,4,4,1), we see that the more symmetric structure has a higher recoverability rate with larger sample sizes. This is despite the fact that the amount of data lost in Q from each of these two structures is approximately 40% in both cases. It is quite possible that these two would behave differently under a different true structure, noting that our true structure τ_0 was also symmetric. It is important to note that for structured partialization, certain shapes of partialization may inherently perform better due to the structure of the underlying model τ.

It is of some interest to note that both (1,2,2,2,2,1) and (2,2,2,2,2) follow somewhat closely with a $p = .45$ or $p = .55$ merge probability, with the net loss of information being comparable between the structured and the more random geometric partialization. Noting that, we see that the recoverability of the tree follows a similar trend in the structured and geometric partialization.

3.5.4 Results - search effectiveness

The results of the secondary experiments meant to assess the runtime of SASEARCHcan be seen in Figure 3.3. The plot on the left shows the average number of steps to find π_0 using data generated from the more randomized models τ_1 and τ_2. In these experiments, SASEARCH was run until 1000 canonical structures were evaluated, noting the first time the algorithm encounters π_0.

The averages shown here for different sample sizes and merge probability only account for those tests that were able to find π_0. Of the 65,000 sample sets taken for each RIM model,

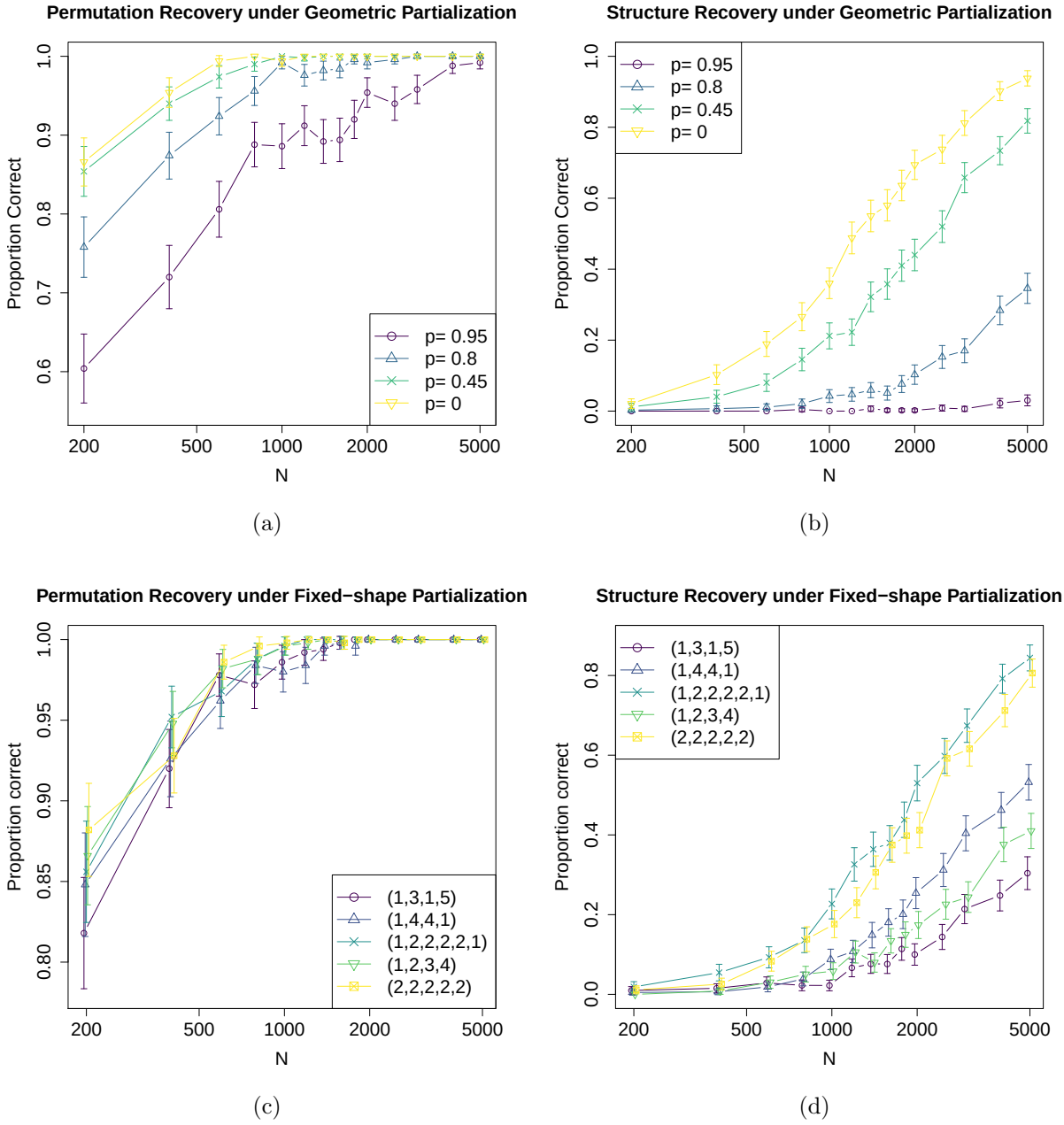
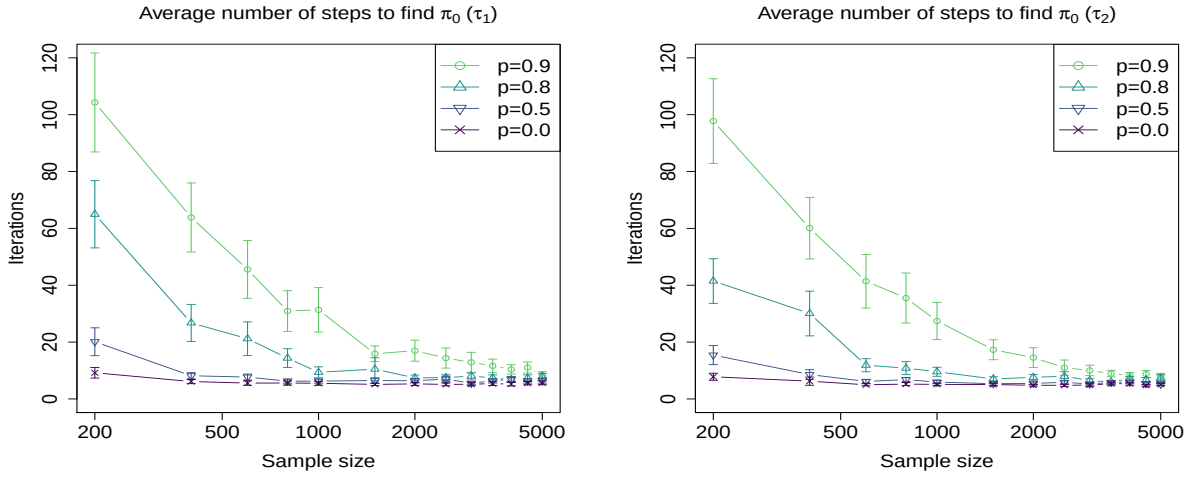


Figure 3.2: The recoverability of the true ranking π_0 (left) and the recoverability of the true structure τ_0 (right) for the model seen in Equation 3.17. Both geometric partialization (top) and structured partialization (bottom) are shown for comparison. Some points have been slightly displaced on the x-axis for readability.

SASEARCH was unable to encounter the true π_0 on less than 150; these are largely samples with small sizes (N =200-400) and high merge probability (p =80%-90%). For searches that were unable to find π_0, all but a handful yielded a model τ' that produced a higher likelihood than the true model τ. The extremely high rate of encountering π_0 demonstrates the excellent ability of SASEARCH to explore the model space and converge to optimal models.

We can see from Figure 3.3 that the two different tree structures studied produce similar results. Even in the most extreme cases the majority of samples were able to arrive at π_0 given sufficient time. As before, the effect of information lost due to partialization ramps up rapidly as we approach the higher merge probabilities, with greater loss of information leading to longer searches before encountering π_0.



(a) (b)

Figure 3.3: Average number of tested π until correct π_0 was encountered vs sample size N for various levels of geometric partialization. The left plot displays the results for the model τ_1 seen in Equation 3.18 while results for τ_2 from Equation 3.19 are displayed on the right.

3.6 Experiments on real data

We approach the problem of fitting RIMs to real world partial rankings with two goals in mind. The first task we explore is the utilization of RIMs with partial rankings for the task

of inference. We explore this with datasets that cover social surveys [19], movie preferences [20], and elections [17]. We discuss at length the information that can be inferred from both the parameters and underlying structure of the RIM.

The second task looks to compare how partial rankings vs total orders influence our ability to fit the models. We assess recoverability utilizing synthetic partialization on total orders of sushi preference [25], but expand on the task of inference with the election data by comparing the results of models fit only to electors producing total orders with those sampled uniformly from all voters, including both total orders and partial rankings.

3.6.1 ISSP data

The International Social Survey Programme[1] conducts annual surveys on various topics across a diverse selection of countries. Of particular interest for us are the questions with answers on a $k = 5$ point Likert scale, symmetrically ranging from "Strongly agree" to "Strongly disagree". The opinions of each respondent induce a partial ranking with up to five grades on the $n = 20$ questions collected. The top grade contains all opinions a respondent strongly agreed with, followed by all opinions a respondent agreed with, etc. Two things are worth noting here. First, a respondent who does not assign one or more of the $k = 5$ Likert scale responses to any questions will produce a partial ranking with fewer grades. A respondent who "strongly agrees" with the first question and "strongly disagrees" with the remainder is indistinguishable from a respondent who "agrees" with the first question and "disagrees" with the remainder. The induced partialization can result in some loss of information. Secondly, because we have $n = 20$ items and a max of $k = 5$ grades, none of the observations will be full rankings.

We studied specifically the results of the 2012 *Family and Changing Gender Roles* survey [19]. In total there were 20 Likert scale questions on spousal duties, children and child rearing, marriage, gender roles, and other related issues. A brief sample of relevant questions

[1] http://issp.org

0. A working mother can establish just as warm and secure a relationship with her children as a mother who does not work.
9. It is all right [sic] for a couple to live together without intending to get married.
11. One parent can bring up a child as well as two parents together.
12. A same sex female couple can bring up a child as well as a male-female couple.
13. A same sex male couple can bring up a child as well as a male-female couple.
14. Watching children grow up is life's greatest joy.
18. Having children increases peoples social standing in society.
19. Adult children are an important source of help for elderly parents.

Table 3.1: A subselection of questions found in the ISSP survey on Family and Changing Gender Roles.

are listed in Table 3.1.

We fit two RIM models, using data separated according to self-identified male and female respondents, as it was believed that due to the subject nature of the survey, opinions would depend on a respondents self identified gender. This allows us to see how these different opinions relate to one another among the two respective groups, and also compare and contrast how men and women differ in their responses.

Due to the surfeit of data, $N_{train} = 15,000$ observations on men and $N_{train} = 15,000$ observations on women were randomly selected and used to train the models. Three sets of testing data were selected from the remaining data, one of only females, one of only males, as well as a third set comprised equally of male and female respondents. Each test set was of size $N_{test} = 4,000$. Runs were initialized from a π_0 determined by sorting the row sums of the observed Q matrix (the so-called Borda order), and a total of 10,000 canonical models were fit for each dataset during SASEARCH.

Results - inference on survey question preferences

A cursory review of the raw data did seem to indicate a difference in responses from men and women. Despite these differences, there are a number of strong similarities between the Recursive Inversion Models fit to these two subsets of the data, which we will denote τ_{Male} and τ_{Female}. Figure 3.4 presents the optimal models returned by the SASEARCH Algorithm.

Drawing our attention first to the θ parameters closest to 0, which reflect the strongest preferences, we note that both men and women agree more with question (12) than question (13). This would imply that people in general feel that two women would be better parents than two men. Inaccuracy of the belief aside, this does fit with the greater societal view of women as caregivers for children, and it is not surprising that many would believe that a lesbian couple would be better for children than a gay male couple. It should be noted that as this is an analysis of partial rankings, the strong dispersion parameter implies only that *when there is a preference* of one question over the other, respondents are much more likely to agree more strongly with question (12) than question (13). The analysis of these preferences at this fine level can be done independent of all other item preferences thanks to the flexibility of the RIM over less informative sub-models such as the GMM.

Questions (0), (11), and (9) are all grouped with and receive more agreement then questions (12) and (13). Questions (0) and (11) both have to deal with raising children as a working or single parent, so it seems sensible that these opinions would fall close to those on gay and lesbian parents. Question (9) seems somewhat less related, asking about unmarried couples living together. Together, this subset of five questions appears at the end of the reference ranking, but with a dispersion parameter near 1 indicating a more uniform preference of these questions over others.

Questions (19) and (18) both have to do with what children bring to their parents' lives, with more agreement that adult children help elderly parents than children increase the parents' social standing. Question (14), falling as the first or second element of π_0, seems to show strong support from both men and women, that raising children is one of life's greatest

Table 3.2: Number of grades K distribution by gender (training data)

	K				
	1	2	3	4	5
Females	0	16	169	5000	9815
Males	1	24	245	5641	9089

joys. That's kind of adorable, right?

More can also be learned by looking not to similarities but to differences. For example, male respondents' opinions on question (17), (16), and (15) are independent of their preferences of other items aside from (10). Each of these questions (aside from question (10)) has to do with the burden children place on their parents' lives. Meanwhile female respondents, those more likely to be directly burdened by the introduction of children, express these preferences mixed with questions (19), (18), and (3), all questions that relate to the positives of having children. While this work does not seek to explain this distinction between the two sexes, it does reflect common societal assumptions and trends.

Results - comparing model estimation on testing data

We also note an interesting fact when we look at the mixed gender testing data. First we note that of the 10,000 models fit in the search process, the models which performed best for the respective gender's training data was also the model with the highest likelihood on the mixed data. While the female model τ_{female} produced a higher log-likelihood on mixed testing data (-18.62 vs. -18.64 for τ_{male}), the trend was reversed when comparing the gender specific test data to their own models. On male only testing data, τ_{male} produced a likelihood of -18.37, while on female testing data τ_{female} has a likelihood of -18.79.

The cause of this becomes quite apparent when we look at the number of grades a respondents partial ranking is broken into. Table 3.2 shows the frequency of each numbers of grades K in the male and female training data.

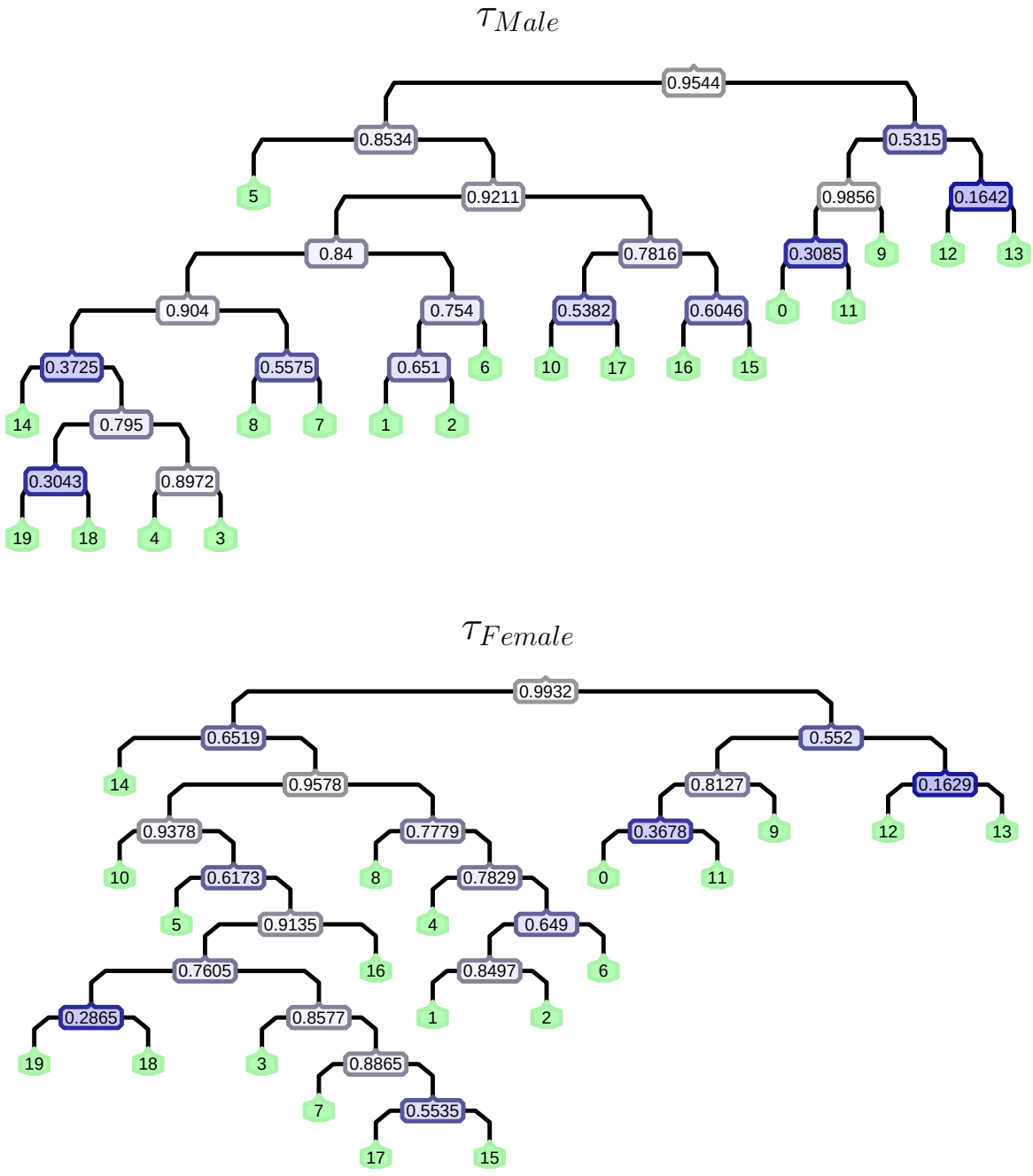


Figure 3.4: The best models selected fit to the ISSP surveys data, stratified by self-identified male respondents (top) and female respondents (bottom).

From Table 3.2 we immediately see that female respondents were more likely to use all five of the Likert-scale options and had less respondents using fewer than five grades at every level. Thus female respondents appeared more informative than male respondents, from the point of view of modeling rankings.

Looking at the missing data due to partialization also helps illustrate this point. The percentage of missing data due to partial rankings ranged from 23.8% to 100% for both genders, but men had on average 37.1% data missingness (SD=9.39%), while women had on average 36.1% data missingness (SD=8.83%).

In a RIM, of two partial rankings with different levels of partialization, the ranking with more unknowns in the inversion matrix will generally have higher likelihood. The male non-respondent, for example, who gave a single response to every question will have a likelihood of 1 on any model. This illustrates why τ_{female} fit on the more refined female data is able to predict less refined (e.g. male) data well in the mixed data, even if these data are coming from a slightly different model.

3.6.2 County Meath ranked voting

The Irish parliamentary voting system provides a good source of real world partial ranking data that was collected explicitly with rankings in mind [16]. The former County Meath voting constituency used a Single Transferable Vote system [16] in which each vote is a ranking of the candidates.

In the 2002 general election there were $n = 14$ candidates vying for 5 seats on the parliament. While over 64,000 votes were cast in total, the vast majority of voters did not rank all candidates, instead providing a variable length top-t ranking. About 5% of voters only selected a single candidate, and less than half of voters ranked more than 5 candidates in total. Only about 5% of ballots (3,166) were completed with a full or effectively full (all candidates but one) ranking.

We fit RIM models on samples from the entire data, as well as on datasets restricted to only full rank ballots studied by previous works [30, 22, 24]. The candidates and their

Candidate	Party	Candidate	Party
Johnny Brady	Fianna Fáil	Tom Kelly	Independent
John Bruton	Fine Gael	Pat O'Brien	Independent
Jane Colwell	Independent	Fergal O'Byrne	Green Party
Noel Dempsey	Fianna Fáil	Michael Redmond	Christian Solidarity
Damien English	Fine Gael	Joe Reilly	Sinn Féin
John Farrelly	Fine Gael	Mary Wallace	Fianna Fáil
Brian Fitzgerald	Independent	Peter Ward	Labour Party

Table 3.3: The candidates and their associated parties for the Meath county election dataset.

political affiliations are listed in Table 3.3.

In a first experiment we compare a model trained on total order data (denoted $\tau_{fullrank}$) with a model trained on a random sample of the same size (denoted τ_{rand}) to predict new samples from the population. Because the total order data are less than 5% of the whole data, the random samples contain a small proportion of full rankings.

We set aside 30% of the entire data for testing ($N_{test} \approx 19,000$); this leaves $N_{fullrank} = 2,273$ total orders. We train $\tau_{fullrank}$ on these data, and fit another 100 models to training datasets of identical size ($N = 2,273$) sampled randomly from the 70% of remaining original data. Each model represents the best model found after 4000 canonical models tried by SASEARCH.

Results

Part of the goal in analyzing the County Meath voting data was to determine how utilizing only total orders available in the data produces differing results from utilizing the top-t rankings as well.

If the total orders come from the same distribution as the whole population, i.e. if the missingness is random, we expect $\tau_{fullrank}$ fit on total orders to be a better predictor than

τ_{rand} fit to mixed data. This is purely due to the fact that, absent bias between the two datasets, random data should represent total order data that has been converted to partial rankings. What we observe is that the test set log-likelihood for $\tau_{fullrank}$ was -9.963, a lower value than *any of the 100 models* built from randomly sampled data. For these, the mean, standard deviation and minimum test log-likelihoods were respectively -9.817, 0.002, and -9.823.

Hence, even though total orders contain more information than partial rankings, the models trained on random samples from the population model the population better. This indicates that the voters who respond with total orders as opposed to partial top-t rankings are not a representative sample of the whole voter population.

In the next experiment, by taking larger samples from the population, we explore this difference further. We took a sample of $N_{fullrank} = 2,000$ total orders to fit $\tau_{fullrank}$ but used samples of variable N_{train} from the original distribution, with N_{train} ranging from 2,000 to 50,000.

Figure 3.5 shows the structure and parameters estimated using the total order data $\tau_{fullrank}$ and using random samples of training data τ_{rand}, in this case with a sample size of $N_{train} = 50,000$.

The top plot in Figure 3.5 shows the model produced using only full rank data. The five eventual winners of the election appear on the first ranks in π_0, with the biggest standout being candidate Reilly who falls at the very end of the reference ranking. Though Reilly was not one of the winners, he was the last candidate to be removed from the running using the single transferable vote. While interesting that the winners are first in the reference ranking, this fact is not an expected results of the model fit. Given the context of these rankings, it is possible that Reilly's rankings were either bimodal (high or low) or uniform in rank.

The model estimated when including the incomplete ballots, τ_{rand}, is found in the bottom plot of Figure 3.5. Here we see that the top split of the tree splits the parties into blocks. The parameter of 0.8776 indicates that the preferences between the two parties are weak, indicative of a lack of consensus likely caused by different voters preferring different parties.

In each branch, the candidates that eventually were elected are ranked highest. Reilly (the only Sinn Fein candidate) is placed in direct comparisons with the Fianna Fàil candidates (red) and "loses".

The two models also present a number of similarities, as one would hope. For example, the Fianna Fàil party and Fine Gael party are well grouped in both instances, with the Fine Gael party in turn being grouped with the remainder of smaller parties and independents. The strongest "preferences" seem to correspond to less popular candidates, with strong preferences ($\hat{\theta} = .466, \hat{\theta} = 0.323$) for Michael Redmond to fall after other independent candidates, and a strong preference ($\hat{\theta} = .542, \hat{\theta} = .589$) against John Farrelly among other Fine Gael candidates. This matches the final results of the election with Redmond being one of the least popular candidates, and Farrelly, despite performing well as an individual, receiving far less votes than other members of his party.

Increasing the sample size produced small gains in the log-likelihood (-9.946 at $N_{train} = 2,000$, -9.9405 at $N_{train} = 17,000$) as sample sizes increased, but sample sizes of $N_{train} = 20,000$ and higher did not see much gain. Most of the resulting models matched the model τ_{rand} in Figure 3.5, with the biggest variations being in the structure on non-primary party candidates, and on the placement of candidates with weak preference ordering, such as Fergal O'Brien.

We also fit Mallows models to these data, again comparing the full ranking only model with a model fit to 2000 random rankings. Similar to the RIM we find that the model built from a random sample has higher log-likelihood on testing data (-10.117 vs -10.291).

This experiment demonstrates that fitting models to only the segments of the population who have entered total orders carries real risk of biasing the models and inferences based on them towards the opinions of individuals producing full rankings. While we do not attempt to stipulate exactly why or in what way these individuals differ, for any self-selecting subset of any population, bias is always a concern.

Code	Movie	Code	Movie
0	Forrest Gump	10	The Fugitive
1	The Shawshank Redemption	11	Apollo 13
2	The Silence of the Lambs	12	Independence Day
3	Jurassic Park	13	The Usual Suspects
4	Star Wars: A New Hope	14	Star Wars: Return of the Jedi
5	Braveheart	15	Batman (1989)
6	Terminator 2: Judgement Day	16	Star Wars: The Empire Strikes Back
7	The Matrix	17	American Beauty
8	Schindler's List	18	Twelve Monkeys
9	Toy Story	19	Dances with Wolves

Table 3.4: List of movies in the Movielens dataset, and their associated numerical code

3.6.3 MovieLens data

A second source of real world partial ranking data is the MovieLens dataset[20], a collection of user movie ratings on a 5 star scale (with half ranks and no 0 star rating) for a total of $K_{max} = 10$ different grades. A partial ranking was produced by grouping and ordering movies which a user gave equal ratings. As with the ISSP data, a user failing to rate any movies at a specific rating results in a loss of information.

Not every movie has been rated by every user, and there are thousands of movies a user could rate. We selected a collection of movies that would produce a fully rated dataset, with each user rating each of the selected movies. For ease of dataset selection, we decided to look at the $n = 20$ most rated movies which had complete reviews for $N = 2467$ individuals.

We selected $N_{train} = 2,000$ users, leaving the remaining data for testing, and estimated a RIM by SASEARCH, iterating over 5,000 canonical models.

It is worth noting, by looking at the most rated movies, we are inherently biasing our sampling toward popular or critically acclaimed films. We are also specifically looking for users who have rated all of these films, so our sample represents avid and diverse movie goers, as opposed to the general population.

Results

Figure 3.6 shows the best model selected from the testing data, and we see that most parameters are near 1, reflecting weak preferences. Compared with the ISSP data which had a similar number of items to rank, and fewer possible grades to distribute those rankings, much of the underlying structure of τ_{Movies} lacks any immediate interpretability.

Much of the tree is arranged in two sets of GMM-like structures, with the most standout grouping represented by the subtree $(16, (4, 14))$. While others may hotly debate the topic, the model suggests a preference for Empire Strikes Back, followed by A New Hope, then lastly Return of the Jedi. Aside from the cluster of Star Wars films, with other science fiction (Terminator 2 and Twelve Monkeys), the remaining structure does not appear to group similar movies together.

Despite having a total of 10 possible ratings to give (0.5,1.0,...,5.0 stars), on average raters used approximately 4.15. About a third of respondents (33.2%) used three or less ratings, and only slightly more than a third (37.9%) gave 5 or more different ratings. On average the largest n_k for a user was 6.95. Note that in a grade of size n_k we are missing a proportion of comparisons equal to $(n_k^2 - n_k)/(n^2 - n)$. Thus a single grade of size 7 corresponds to a loss of over 11% of Q.

The reader is by now cautioned that this dataset, being limited to oft rated movies, inherently means more good or critically acclaimed movies. Moreover, it is not possible to eliminate the possibility that the sample of raters is a biased sample of the population. Finally, it is likely that the weak structure may also be indicative of lack of consensus.

3.6.4 Sushi preferences data

We looked at a set of sushi preference data, where respondents ranked $n = 10$ different types of sushi [25]. This dataset has been previously explored using RIM [30], and contains complete preference data from each respondent. In order to expand on previous research, we applied geometric partialization to the dataset. This provides a real world dataset for

exploring how the loss of information affects the resulting model.

From a total of $N = 5,000$ respondents, an 80:20 split of the data was used to produce a training and testing set, respectively. The training set was subject to geometric partialization with p_{merge} ranging from 0 to 0.95; for each p_{merge}, 1,000 canonical models were fit in SASEARCH.

Results

In these data, the partial rankings are created from a set of total orders. Hence, it is interesting to compare the RIM fitted to partial rankings against that fitted to total orders, which was also studied in previous work [30]. We find that the estimated RIM is remarkably robust under data partialization.

Figure 3.7 displays the test log-likelihood of the estimated models versus the degree of partialization p_{merge}. We note that, similar to synthetic tests, the model predictive power is effectively the same for p_{merge} up to 60%, then rapidly declines as p_{merge} approaches 1. Figure 3.8 displays two estimated RIMs, with $p_{merge} = 90\%$ and $p_{merge} = 0\%$ (total orders). While the reference permutation π_0 differs between the models, a closer look at model structure and parameters reveals high similarity. In both models, the top three splits are associated with single items uni, sake, and anago. The θ parameters are all nearly 1, indicating that these items' ranks are almost uniformly distributed w.r.t the items lower in the tree. Both models contain a subtree with toro, maguro, and tekka-maki (all tuna based) with similar parameterizations. As [30] shows, when a subtree has internal nodes with equal θ values, the structure of the subtree is not identifiable. Hence, the variation in structure for these nodes reflects weak information in the data. Hence, we see that estimation algorithm can recover all the features of the total order model, even under heavily partialized data. The models found for other p_{merge} values exhibit the similar characteristics.

3.7 Conclusion

The work here seeks to expand the theory of the Recursive Inversion Model to the space of partial ranking data, a common and realistic measure of people's preferences. We accomplishes this task and demonstrate that the overhead of partial ranking data with respect to total orders remains tractable for maximum likelihood estimation. We demonstrate the usefulness of this in a wide array of analyses, with the Meath data strongly highlighting the benefit of utilizing all partial ranking data rather than just the subset of total orders.

This expansion of the model also allows for a wide new space of opinion data to be utilized as rankings for inference and analysis, by converting ordinal ratings to rankings. This opens the door to preference ranking analysis with the Recursive Inversion Model on data that was previously inaccessible, allowing for the representation of a consensus ordering, rich inference about the clustering of preference items, and interpretable parameters that reflect the strength of preference of some items over others.

In establishing this theory for partial ranking likelihood calculations, a natural extension yields methods for marginal rank and inversion matrices estimation. We detail these methods with algorithms for calculation, allowing for both marginal rank and inversion matrices, and conditional rank and inversion matrices for partial rankings.

As previously noted, the RIM represents a superclass containing the Mallows and Generalized Mallows Models. Given proper constraints to the structure and/or parameters, the theory developed here for the RIM can likewise be applied to the subclass of MMs and GMMs.

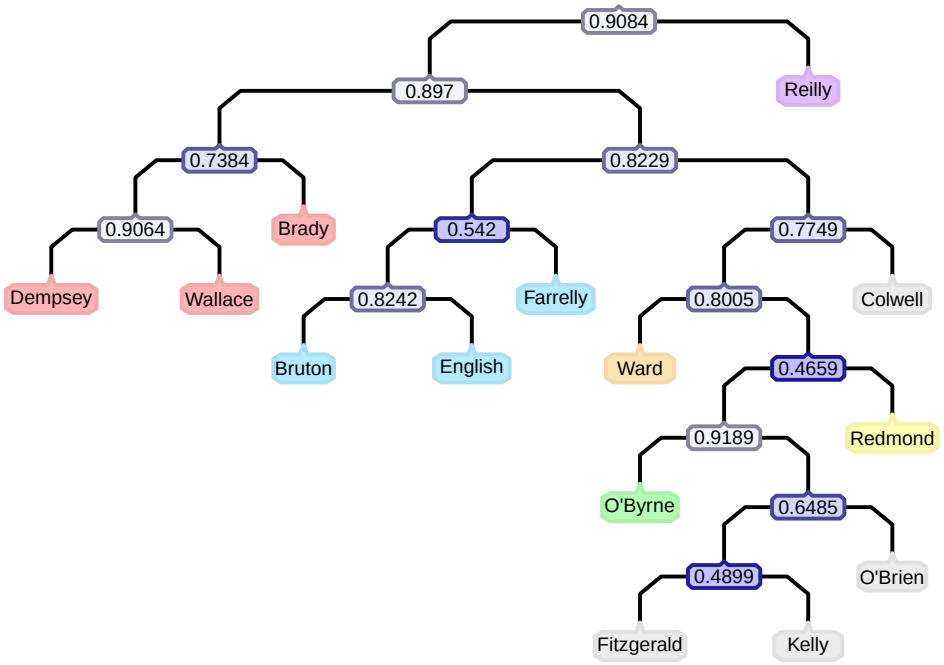
Relaxation of the canonical requirement can also open the door to new types of inference. One can, for example, fit two models on different datasets but constrained to identical structure. Relaxation of canonicalization of these models allows for inversions of branches between the two different models, in turn allowing for inference of the difference between the two populations represented.

This is similar to the case of a mixture model, but with the added requirement that all models in the mixture share structure. This allows for more direct comparison of the populations in question. If that inference is not of interest, then mixture models of dis-

tinct structures can instead be found, which can more accurately describe populations with multiple modes of consensus rankings.

The work here seeks to assemble the tools needed for inference, analysis, and modeling utilizing a more rich and realistic representation of peoples' preferences. We present this work in the the hope that it will inspire the gathering and use of more realistic preference ranking data, as well as ratings expressed as ranking data.

$\tau_{FullRank}$



τ_{Random}

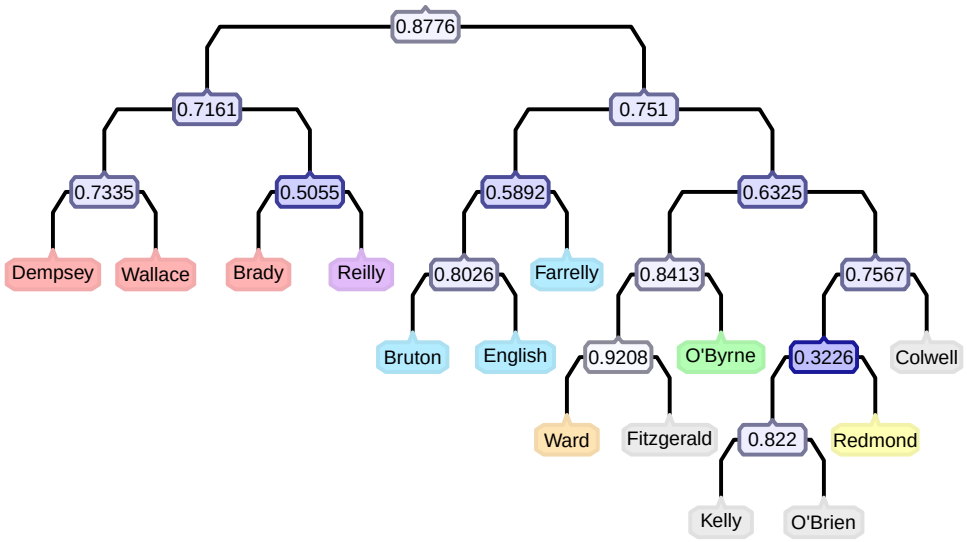


Figure 3.5: The models estimated from the County Meath voting data, limited only to those voters who completed the entire ballot (top) and including a random sample of mixed total order and partial ballots (bottom).

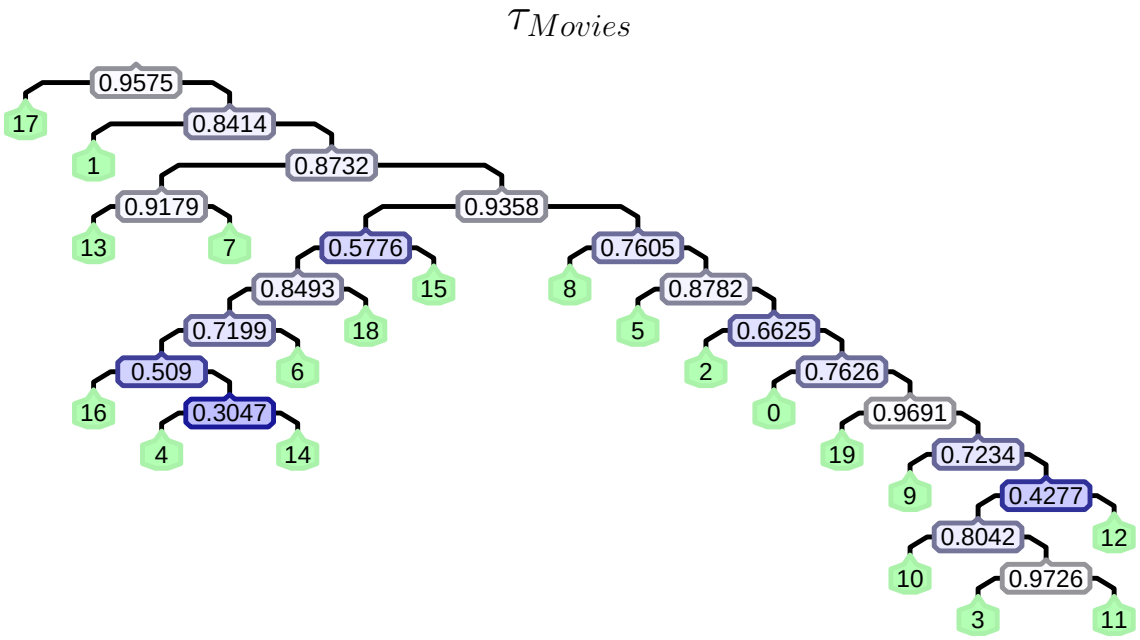
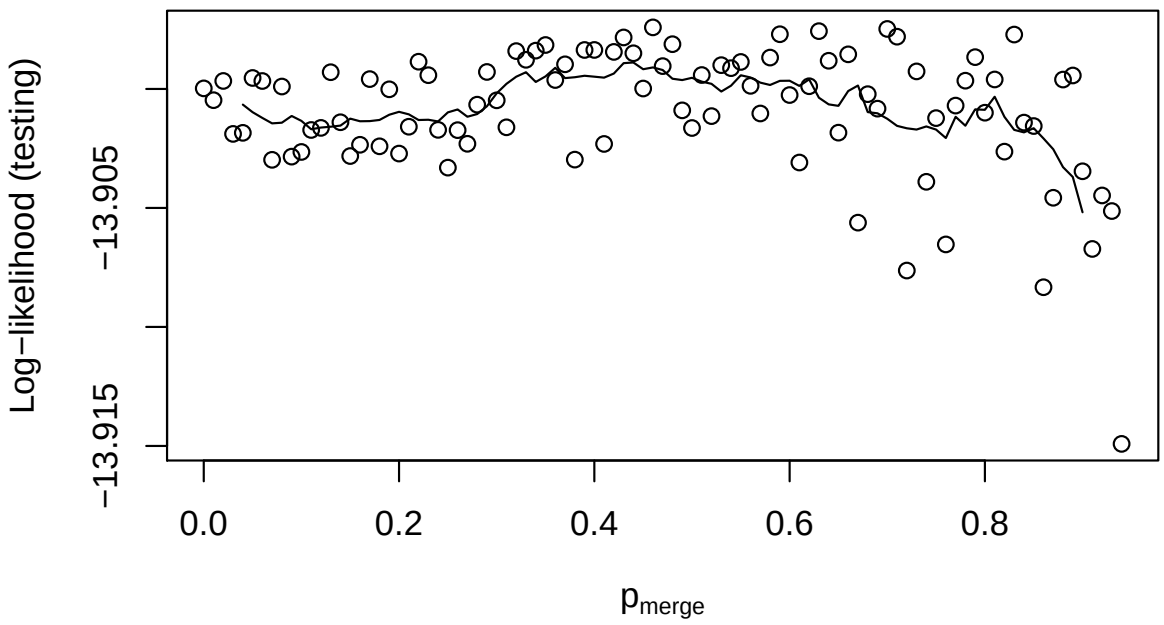


Figure 3.6: The best model selected from the MovieLens dataset. Much of the structure consists of two GMM-like subtrees, with the left subtree comprising only Scifi and Action films.



(a)

Figure 3.7: Sushi test log-likelihood as a function of the merge probability p_{merge}, with a superimposed moving 9-point average.

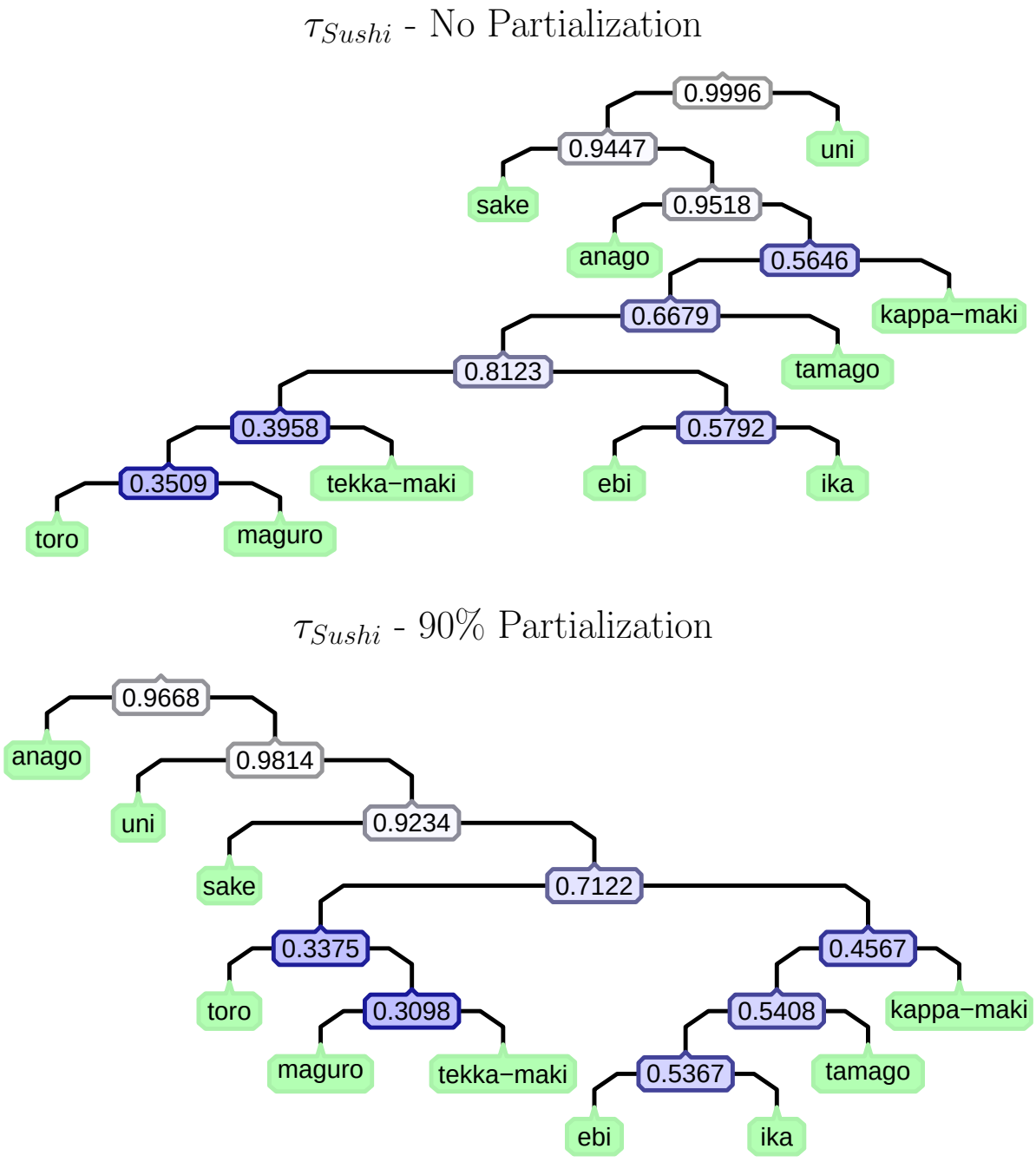


Figure 3.8: The best models selected from testing data for 0% partialization or full rankings (top) and 90% partialization (bottom).

Chapter 4

MODELS FOR RANKINGS WITH LANDMARKS

Rankings and ratings are both methods of expressing preferences that capture distinct but complimentary pieces of information. Despite this, very little work has been dedicated to the task of synergizing these two different modalities of expression preferences [36].

We seek to combine these two data types by introducing rankings with landmarks, a type of ranking that also losslessly incorporates the information present in ratings. We describe this new space of rankings with landmarks, which introduce new challenges in the context of modeling rankings.

To address the new challenges presented in rankings with landmarks, we develop a number of models that can incorporate these rankings without conflicting with the requirements added by landmarks. The standout of the models considered, the Landmark Generalized Mallows Model, represents likelihoods over the space of rankings with landmarks with a model that, importantly, utilizes interpretable parameters. Of biggest significance, the model utilizes a modal reference ranking π_0 that can reflect the relationship between items being ranked and the ratings those items receive under consensus.

4.1 Rankings with landmarks

We start with a running example, in which data is rated on a ordinal scale with $m + 1 = 4$ grades, and the set of items (for example, movies) is $\mathcal{E} = \{a, b, c, d, e, f\}$. In this paper we assume that items are *rated* with a rating function $r : \mathcal{E} \to \{0, \dots m\}$, in which the most preferred rating is 0 and the least is m. Additionally, a permutation or ranking π of the items in $\mathcal{E}$ is given.

In our model, we assume that a respondent provides both a ranking and a rating of all

elements $i, i' \in \mathcal{E}$, and that the ranking is *consistent* with the corresponding rating; that is, $i \succ_\pi i'$ whenever $r(i) < r(i')$. Hence, when a rating is given, the ranking disambiguates between items with the same rating. For example, assume that $r(c) = 0$, $r(a) = r(d) = 1$, $r(b) = r(f) = 2$ and $r(e) = 3$, which is consistent with $\pi = [c, a, d, b, f, e]$ as above. Then, π adds the information that $a \succ d$ and $b \succ f$. Conversely, given a ranking π, a rating r consistent with π groups consecutive items into grades. There are multiple ratings consistent with any π; for example rating a, b, c, d with 0 and the rest of the items with 3 would also be consistent with π. Since the two modalities of expressing preferences convey dependent but complementary information, we set out to model this information in a unified way.

We propose to do this by creating, in addition to $\mathcal{E}$ the set of items to be ranked, a set of m *landmarks* $\mathcal{L}$. A permutation with landmarks is an ordered list π of length $n + m$ containing a permutation of $\mathcal{E}$ interleaved with the m landmarks. The relative ordering of the landmarks is, by definition, always the identity permutation.

The role of the landmarks is to act as "separators" in the list π, such that the first landmark, denoted $\boxed{1}$, follows after all the items rated 0 (these are at the top of the list π) and precedes all the items rated 1. The second landmark $\boxed{2}$ then follows all the items rated 1 and precedes the items rated 2, and so on.

Given a consistent pair (π, r), denote by

$$\mathcal{E}_k^{\pi,r} = \{e \in \mathcal{E} \,|\, e \text{ is preceded by exactly } k \text{ landmarks}\}, \text{ for } k = 0 : m. \tag{4.1}$$

It is easy to see that, $\mathcal{E}_k^{\pi,r} = r^{-1}(k)$, the set of items rated k, which will be called *grade* k of π.

Example 5 *Let* $\mathcal{E} = \{a, b, c, d, e, f\}$ *as before,* $m = 3$, $\mathcal{L} = \{\boxed{1}, \boxed{2}, \boxed{3}\}$. *Let* $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$ *convey both the ranking* $\pi_\mathcal{E} = [c, a, d, b, f, e]$ *and the rating* $r(c) = 0$, $r(a) = r(d) = 1$, $r(b) = r(f) = 2$ *and* $r(e) = 3$; π *is in* $\mathbb{S}_6^3$. *The grades are* $\mathcal{E}_0^{\pi,r} = \{c\}, \mathcal{E}_1^{\pi,r} = \{a, d\}$, *and so*

on, as shown below

$$\pi_{\mathcal{E}} \;=\; [\underbrace{c}_{r=0},\, \underbrace{a,\, d}_{r=1},\, \underbrace{b,\, f}_{r=2},\, \underbrace{e}_{r=3}]\,. \tag{4.2}$$

The locations of the $m = 3$ landmarks are given by $\pi_{\mathcal{L}}^{-1} = [\,2\,5\,8\,]$ We will continue to use $\pi_0 = [a, b, \boxed{1}, c, d, \boxed{2}, e, \boxed{3}, f]$, and $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$ in the examples throughout the paper.

The list π is a permutation over $\mathcal{E} \cup \mathcal{L}$, in which the order of the landmarks in $\mathcal{L}$ is fixed to the identity permutation. We denote the set of all such permutations as $\mathbb{S}_n^m$, which we refer to as the set of *permutations over n items with m landmarks*.

We modify our notation here for ease of communication. We define $\pi_{\mathcal{E}}$ as the permutation on $\mathcal{E}$ induced by π, represented as a list. Finally, $\pi_{\mathcal{L}}^{-1}$ denotes the m-tuple of ranks of landmarks in π; $\pi_{\mathcal{L}}^{-1}$ is increasing from left to right, i.e., $\pi_{\mathcal{L}}^{-1}(k) < \pi_{\mathcal{L}}^{-1}(k+1)$. This is a slight abuse of notation and should be more rigorously denoted $(\pi^{-1}(\mathcal{L}))_k$; note also that $(\pi_{\mathcal{L}})^{-1}(k) = k$ always.

The next step is to define Mallows-like distributions over $\mathbb{S}_n^m$. Our primary model will be similar to GMM^s, but other models based on the GMM^v and RIM are also explored here, with details on the procedures needed to fit them where possible and relevant.

4.2 Models for rankings with landmarks

The task of modeling rankings with landmarks introduces new challenges that previous models are not designed to address. Namely, the models must be designed around addressing the problem of non-invertability of landmarks. We propose a number of models, starting with the most promising Landmark GMM^s, each of which address the problem of landmarks in different ways. We base these models on the class of exponential models around inversions, maintaining the interpretability of parameters of these models in the process.

4.2.1 The Landmark GMM^s (L-GMM^s)

Similarly to the GMM^s, this model has a central permutation $\pi_0 \in \mathbb{S}_n^m$ called the modal permutation, and a set of parameters, $\vec{\theta} = (\theta_1, ..., \theta_{n+m-1})$, one for each rank of π except for the last, whose role is to penalize inversions w.r.t. the modal permutation. We adopt the following model for sampling a permutation with landmarks. For each rank $j = 1$ to $m + n - 1$, we sample a position in π_0, by sampling s_j according to $\exp(\theta_j, n + m - j)$, where $\exp(\theta, k)$ denotes the geometric distribution with parameter θ and range $\{0, 1, \ldots k\}$ as before. If $\pi_0(s_j + 1) \in \mathcal{E}$, the item at this position is placed in rank j of π and deleted from π_0. Hence, s_j is the number of positions in π_0 before π_j that are being skipped. If $\pi_0(s_j + 1) \in \mathcal{L}$, then *the first landmark still in* π_0 is deleted and placed in π (rather than the landmark at $\pi_0(s_j + 1)$, if it is not the first).

Thus, the probability of a permutation is

$$P^{\text{L-GMM}^s}(\pi|\pi_0, \vec{\theta}) \;=\; \frac{1}{\prod_{j=1}^{n+m-1} Z_{n+m-j}(\theta_j)} \cdot \prod_{j=1}^{n+m-1} \begin{cases} \theta_j^{s_j(\pi)} & \text{if } \pi^{-1}(j) \in \mathcal{E} \\ \sum_{l=k}^{m} \theta_j^{s_j^{kl}} & \text{if } \pi^{-1}(j) = k \in \mathcal{L}. \end{cases} \tag{4.3}$$

In the above, $s_j(\pi)$ has the same meaning as for the standard GMM^s, namely the sum of row $\pi(j)$ of $Q(\pi; \pi_0)$ at stage j. When the k-th landmark is rank j, the probability of picking it is the total probability of picking any of the $m-k+1$ remaining landmarks; thus, the variables s_j^{kl} are the s_j's of the landmarks $l = k : m$ (i.e. 1 less than their ranks in the current π_0). The $(n-1) + m(m-1)/2$ variables (s_j, $j \in \pi_{\mathcal{E}}^{-1}$; s_j^{kl}, for $j \in \pi_{\mathcal{L}}^{-1}$, $1 \le l \le k \le m$) represent the *code* of $\pi \in \mathbb{S}_n^m$.

For the L-GMM^s there are multiple sequences $s_1, \ldots s_{n+m-1}$ that produce the same π. However, since any sequence corresponds to some π, the normalization constant is the same as for the standard GMM^s with the same number of parameters. As we shall see, this is not the case for other apparently natural Mallows models with landmarks.

Example 6 *Let* $\pi_0 = [a, b, \boxed{1}, c, d, \boxed{2}, e, \boxed{3}, f]$ *with* $n = 6$ *and* $m = 3$*;* $\mathcal{E} = \{a, b, c, d, e, f\}$, $\mathcal{L} = \{\boxed{1}, \boxed{2}, \boxed{3}\}$. *Set* $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$.

The table below shows the construction of π for a L-GMM^s *and the probability of each stage in the sampling of π.*

rank	*item*	π_0 *after deletion*	s_j	s_j^{kl}	*Factor in eq 4.3*
1	c	$[a, b, \boxed{1}, \not{c}, d, \boxed{2}, e, \boxed{3}, f]$	$s_1 = 3$	$-$	$\frac{\theta_1^3}{Z_8(\theta_1)}$
2	$\boxed{1}$	$[a, b, \boxed{\not{1}}, d, \boxed{2}, e, \boxed{3}, f]$	$-$	$s_2^{11} = 2,\ s_2^{12} = 4,\ s_2^{13} = 6$	$\frac{\theta_2^2+\theta_2^4+\theta_2^6}{Z_2(\theta_7)}$
3	a	$[\not{a}, b, d, \boxed{2}, e, \boxed{3}, f]$	$s_3 = 0$	$-$	$\frac{\theta_3^0}{Z_6(\theta_3)}$
4	d	$[b, \not{d}, \boxed{2}, e, \boxed{3}, f]$	$s_4 = 1$	$-$	$\frac{\theta_4^1}{Z_5(\theta_4)}$
5	$\boxed{2}$	$[b, \boxed{\not{2}}, e, \boxed{3}, f]$	$-$	$s_5^{22} = 1,\ s_5^{23} = 3$	$\frac{\theta_5^1+\theta_5^3}{Z_4(\theta_5)}$
6	b	$[\not{b}, e, \boxed{3}, f]$	$s_6 = 0$	$-$	$\frac{\theta_6^0}{Z_3(\theta_6)}$
7	f	$[e, \boxed{3}, \not{f}]$	$s_7 = 2$	$-$	$\frac{\theta_7^2}{Z_2(\theta_7)}$
8	$\boxed{3}$	$[e, \boxed{\not{3}}]$	$-$	$s_8^{33} = 1$	$\frac{\theta_8^1}{Z_1(\theta_8)}$
9	e	$[\not{e}]$	$s_9 = 0$	*always*	1

(4.4)

Any sequence that generated π can be obtained from the s_j and s_j^{kl} columns, with any combination of s_j^{kl} values at each rank in $\pi_{\mathcal{L}}^{-1}$. For example, the sequences $\vec{s} = (3, 2, 0, 1, 1, 0, 2, 1)$ and $\vec{s} = (3, 6, 0, 1, 3, 0, 2, 1)$ both produce the same π.

For two permutations with landmarks, $\pi, \pi_0 \in \mathbb{S}_n^m$, the inversion matrix $Q(\pi; \pi_0)$ is a $(n+m)\times(n+m)$ matrix with rows and columns indexed by $\mathcal{E}\cup\mathcal{L}$ defined as in equation (2.1).

Similarly to the standard GMM^s model, we see that the s_j inversion counts can be obtained by summing elements in the $\pi^{-1}(j)$-th row of $Q(\pi)$. Namely, we have

$$s_j(\pi) \ = \sum_{e' \succ_{\pi_0} e} Q_{ee'}(\pi) \quad \text{with } e = \pi^{-1}(j) \in \mathcal{E}, \tag{4.5}$$

and for landmark $k \in \mathcal{L}$ on rank j,

$$s_j^{kl}(\pi) \ = \sum_{e \in \mathcal{E}\cup\mathcal{L}, e \succ_{\pi_0} l} Q_{el}(\pi), \quad \text{for } l \in \{k, \dots m\} \subseteq \mathcal{L}. \tag{4.6}$$

Note that as before, $Q(\pi)$ can be reordered according to a given π_0, thus we drop the notation $Q(\pi;\pi_0)$ unless specifically relevant.

It is known that on $\mathbb{S}_n$, the code $s_{1:n-1}(\pi|\pi_0)$ uniquely defines π. It can be shown that this holds also for permutations with landmarks. Moreover, a permutation π can be recovered from the inversion counts $s_j(\pi|\pi_0)$ corresponding to items only. More formally, set $s_j = -1$ whenever a landmark is on rank j. Then, the inversion counts $s_{1:n+m-1}(\pi|\pi_0)$, with $s_j \in \{0 : n-j\} \cup \{-1\}$ uniquely determine $\pi \in \mathbb{S}_n^m$. The proof of this statement is straightforward and left to the reader.

Example 7 *The inversion matrix $Q(\pi;\pi_0)$ corresponding to Example 5 is*

$$Q(\pi) \;=\; \begin{array}{c|ccccccccc} & a & b & \boxed{1} & c & d & \boxed{2} & e & \boxed{3} & f \\ \hline a & - & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ b & 0 & - & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ \boxed{1} & 1 & 1 & - & 0 & 1 & 1 & 1 & 1 & 1 \\ c & 1 & 1 & 1 & - & 1 & 1 & 1 & 1 & 1 \\ d & 0 & 1 & 0 & 0 & - & 1 & 1 & 1 & 1 \\ \boxed{2} & 0 & 1 & 0 & 0 & 0 & - & 1 & 1 & 1 \\ e & 0 & 0 & 0 & 0 & 0 & 0 & - & 0 & 0 \\ \boxed{3} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & - & 0 \\ f & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & - \end{array} \tag{4.7}$$

For our π, and for $k = 1$, and $j = 2$, landmark $\boxed{1}$ is on rank 2; s_j could have been any of 2, 4, or 6, corresponding the respective inversion counts for $\boxed{1}$, $\boxed{2}$, $\boxed{3}$ in π_0 after deleting the first item c. Thus, s_2^{1l} corresponds to summing the elements in row $\boxed{1}$ of $Q(\pi|\pi_0)$ up to $\boxed{l}$.

Furthermore, the code $s_{1:8}(\pi|\pi_0) = (s_1 = 3, s_2 = -1, s_3 = 0, s_4 = 1, s_5 = -1, s_6 = 0, s_7 = 2, s_8 = -1)$ is sufficient to reconstruct π as $\pi = [c, \boxed{T}, a, d, \boxed{T}, b, f, \boxed{T}, e]$, then replacing the placeholder landmarks $\boxed{T}$ with $\mathrm{id}_{\mathcal{L}}$ to recover π.

In addition to the L-GMM^s presented here, other possibilities of extending the GMM to

permutations with landmarks exist, and we enumerate some of them here.

4.2.2 The GMM^s with Censored Landmarks (CL-GMM^s)

The following stagewise sampling model over $\mathbb{S}_n^m$ is perhaps more intuitive than the above L-GMM^s. Under this model, at most one landmark is made available for sampling at any step. Put another way, a landmark is censored from π_0 until all landmarks before have been placed in π.

Similarly to the L-GMM^s, the CL-GMM^s has a central permutation $\pi_0 \in \mathbb{S}_n^m$ (also called modal ranking), and a set of parameters, $\theta_{1:m+n-1}$, one for each rank of π except for the last.

CL-GMM^s adopts the following model for sampling a permutation with landmarks. First, all landmarks aside from the first are removed from π_0. They will be reinserted at a later stage, therefore their rank is stored, and is updated at every stage, as elements are deleted from π_0. For each rank j from 1 to $m+n-1$, we sample code s_j in π_0 according the the exponential distribution $\exp(\theta_j, n-j+1_{[\mathcal{L}\setminus\pi(j:n+m)\neq\emptyset]})$; the expression $1_{[\mathcal{L}\setminus\pi(j:n+m)\neq\emptyset]}$ is 0 if no landmarks remain in π_0 (to be placed in π), and 1 otherwise. If $\pi_0(s_j+1)$ is in $\mathcal{E}$, the item at this position is placed in rank j of π and deleted from π_0. If $\pi_0(s_j+1) \in \mathcal{L}$, then this landmark is removed from π_0 and the next landmark is placed at its relative index according to the original π_0.

Example 8

rank	*item*	π_0 *after deletion*	s_j
1	c	$[a, b, \boxed{1}, \not{c}, d, e, f]$	$s_1 = 3$
2	$\boxed{1}$	$[a, b, \not{\boxed{1}}, d, e, f]$	$s_2 = 2$
3	a	$[\not{a}, b, d, \boxed{2}, e, f]$	$s_3 = 0$
4	d	$[b, \not{d}, \boxed{2}, e, f]$	$s_4 = 1$
5	$\boxed{2}$	$[b, \not{\boxed{2}}, e, f]$	$s_5 = 1$
6	b	$[\not{b}, e, \boxed{3}, f]$	$s_6 = 0$
7	f	$[e, \boxed{3}, \not{f}]$	$s_7 = 2$
8	$\boxed{3}$	$[e, \not{\boxed{3}}]$	$s_8 = 1$
9	e	$[\not{e}]$	$s_9 = 0$

(4.8)

Note how at rank 2, $\boxed{1}$ is deleted from its position after b and before d, and $\boxed{2}$ is inserted between d and e. The length of π_0 does not decrease when landmarks $\boxed{1}, \boxed{2}$ are sampled, but decreases by 1 in all other cases.

This yields the probability for a ranking π,

$$P^{\text{CL}-\text{GMM}^s}(\pi|\pi_0, \vec{\theta}) \propto \prod_{j=1}^{n+m-1} \theta_j^{s_j}. \tag{4.9}$$

Unfortunately, the normalization constant associated to sampling rank j is not always $Z_{n-j+1}(\theta_j)$ as in the L-GMM^s, since the number of items in the π_0 list at step j depends on the number of landmarks already inserted in π. The length of the list π_0 stays the same when a landmark $l = 1 : m-1$ is sampled, while in the other cases it decreases by 1. Hence, while the CL-GMM^s is a valid sampling model, it is not known how to obtain its normalization constant, and therefore interpreting and fitting the model to data would be much more challenging. We do not further consider the CL-GMM^s in this paper.

4.2.3 GMM^v with Holes (H-GMM^v)

The H-GMM^v is a simple variation to the standard Generalized Mallows Model. Unlike the previous example, the H-GMM^v does not represent the landmarks explicitly.

The central permutation of H-GMM^v is over $\mathcal{E}$ only, $\pi_0 \in \mathbb{S}_n$, the set of parameters is $\theta_{1:n}$, one for each item in $\mathcal{E}$. The idea is to sample stagewise, in the same way as for GMM^v, by placing item $\pi_0(j)$ in the $v_j + 1$'th free rank in π, but to allocate $n + m$ spaces. Hence, $v_j \in 0 : n + m - j$. Therefore, after sampling $v_{1:n}$ the permutation π will have m "holes" which indicate the landmarks' locations.

The sample space is the same $\mathbb{S}_n^m$, but the distribution $P^{\mathrm{H-GMM}^v}$ has only n parameters instead of $n+m-1$ like in the L-GMM^s. The distribution of the holes is implicit from m, n, and the parameters, while in the latter model, by explicitly placing the holes in the central permutation π_0, one can control the partition of the items into grades.

The probability follows a standard GMM^v with a modified normalization constant, corresponding to the larger range of the v_j variables. Namely, the probability of a particular v_j is

$$P^{\mathrm{H-GMM}^v}(v_j) = \theta_j^{v_j} / Z_{n+m-j}(\theta_j), \quad \text{for } j = 1 : n. \tag{4.10}$$

This gives a ranking π a probability of

$$P^{\mathrm{H-GMM}^v}(\pi|\pi_0, \vec{\theta}) = \prod_{j=1}^{n} \frac{\theta_j^{v_j}}{Z_{n+m-j}(\theta_j)}. \tag{4.11}$$

Define $\pi_0 = [\pi_{0|\mathcal{E}}, \mathrm{id}_{|\mathcal{L}}]$, an extended central permutation, in which the landmarks are placed at the end. It can be easily seen that H-GMM^v is identical to a GMM^v model with central permutation π_0 and parameters $[\,\theta_1, \ldots \theta_n, \underbrace{0, \ldots 0}_{\times m-1}]$. For any $\pi \in \mathbb{S}_n^m$ let the inversion matrix $Q(\pi|\pi_0)$ be defined as in equation (4.7). It can be shown that the code $(v_{1:n})$ can be calculated

based on this inversion matrix as

$$v_j \;=\; \sum_{i \prec_{\pi_0} j} Q_{ij}(\pi|\pi_0), \text{ for } j = 1:n. \tag{4.12}$$

Note that the last m columns of $Q(\pi|\pi_0)$, corresponding to the landmarks, are never used and need not be stored.

Example 9 *Let* $\pi_0 = [a, b, c, d, e, f]$ *with* $n = 6$ *and* $m = 3$*;* $\mathcal{E} = \{a, b, c, d, e, f\}$*,* $\mathcal{L} = \{\boxed{1}, \boxed{2}, \boxed{3}\}$*. Set* $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$*. The table below shows the construction of* π *for a* H-GMM^v *and the probability of each stage in the sampling of* π*.*

item	v_j *range*	v_j	π *after insertion*
a	$0:8$	2	$[-, -, a, -, -, -, -, -, -]$
b	$0:7$	4	$[-, -, a, -, -, b, -, -, -]$
c	$0:6$	0	$[c, -, a, -, -, b, -, -, -]$
d	$0:5$	1	$[c, -, a, d, -, b, -, -, -]$
e	$0:4$	4	$[c, -, a, d, -, b, -, -, e]$
f	$0:3$	2	$[c, -, a, d, -, b, f, -, e]$

(4.13)

One can also map π_0 *to the* $\mathbb{S}_n^m$ *element* $[a, b, c, d, e, f, \boxed{1}, \boxed{2}, \boxed{3}]$*. The inversion matrix* $Q(\pi|\pi_0)$ *can be used to calculate all necessary* v_j *values and is identical to the matrix shown in equation* (4.7)*.*

This model is more restrictive compared to the L-GMM^s, and aside from a brief comment on how to fit it, the H-GMM^v is not a focus of this work.

4.2.4 Independent Interleaving Landmarks GMM (IIL-GMM)

This model is based on the concept of *interleaving* (or *shuffle*) between two permutations. An interleaving π of $\pi_{\mathcal{E}}$ and $\pi_{\mathcal{L}}$ over disjoint sets $\mathcal{E}, \mathcal{L}$ is a permutation of $\mathcal{E} \cup \mathcal{L}$ that preserves

the relative order of the items in $\mathcal{E}$, as well as that of the items in $\mathcal{L}$. That is $\pi_{|\mathcal{E}} = \pi_{\mathcal{E}}$, $\pi_{|\mathcal{L}} = \pi_{\mathcal{L}}$. In our case, $\pi_{\mathcal{L}}$ will be always the identity $\mathrm{id}_{\mathcal{L}}$.

The IIL-GMM is thus a GMM^{v}described by a central permutation over the items $\pi_0 \in \mathbb{S}_n$ and the parameter vector $\vec{\theta} \in [0,1]^{n-1}$, combined with a probabilistic model P^{interl} for the interleaving with the identity permutation of landmarks $\mathrm{id}_{\mathcal{L}}$.

The probability of an interleaving P^{interl} is exponential in the total number of inversions incurred [2, 35]. For a landmark $k \in \mathcal{L}$, define the number of inversions u_k as the number of items in $\mathcal{E}$ ranked by π after k,

$$u_k \;=\; n - (\pi^{-1}(k) - k). \tag{4.14}$$

The probability of the interleaving is given by

$$P^{\mathrm{interl}}(u_{1:m};\theta_0) \;=\; \frac{\theta_0^{\sum_{k=1}^{m} u_k}}{Z_{n,m}(\theta_0)} \quad \text{with } Z_{n,m}(\theta_0) \;=\; \frac{(\theta_0)_{n+m}}{(\theta_0)_m (\theta_0)_m}, \quad \text{and } (\theta)_n = \prod_{k=1}^{n} \frac{1-\theta^k}{1-\theta}\,. \tag{4.15}$$

The probability of a permutation $\pi \in \mathbb{S}_n^m$ is given by

$$P^{\mathrm{IIL-GMM}}(\pi) \;=\; P^{\mathrm{GMM}^s}(\pi_{\mathcal{E}}) P^{\mathrm{interl}}(u_{1:m};\theta_0), \tag{4.16}$$

As has been shown in previous work, P^{interl} is log-concave in θ_0 [30].

The single parameter $\theta_0 \in [0,1]$ of P^{interl} controls the distribution of the landmarks in the interleaving. If $\theta_0 = 1$, the landmarks are uniformly distributed, in the sense that any interleaving has equal probability. If θ_0 is near 0, the landmarks are placed closer to the beginning of the interleaving, and the most likely interleaving is the one where the landmarks preceed all items.

Like previous models the IIL-GMM is parameterized by a reference permutation π_0, but the model construction dictates that the landmarks must always come first in the ranking. Thus π_0 will always have the form $\pi_0 = [\mathrm{id}_{\mathcal{L}}, \pi_{0|\mathcal{E}}]$.

Example 10 *Let* $\pi_0 = [a, b, c, d, e, f]$ *with* $n = 6$ *and* $m = 3$*;* $\mathcal{E} = \{a, b, c, d, e, f\}$*,* $\mathcal{L} = \{\boxed{1}, \boxed{2}, \boxed{3}\}$*. Set* $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$*.*

The table below shows the construction of π *by the interleaving of* $\pi_{\mathcal{E}} = [c, a, d, b, f, e]$ *and* $\pi_{\mathcal{L}} = [\boxed{1}, \boxed{2}, \boxed{3}]$ *in the* IIL-GMM*. The construction of* $\pi_{\mathcal{E}}$ *is assumed to follow a standard* GMM*.*

item	u_k *range*	u_k	π	P^{interl} *Factor*
$\boxed{3}$	$0:6$	5	$[c, a, d, b, f, e]$	θ_0^5
$\boxed{2}$	$0:5$	3	$[c, a, d, b, f, \boxed{3}, e]$	θ_0^3
$\boxed{1}$	$0:3$	1	$[c, a, d, \boxed{2}, b, f, \boxed{3}, e]$	θ_0^1
$-$	$-$	$-$	$[c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$	$1/Z_{6,3}(\theta_0)$

(4.17)

The values of u_k *can be calculated from* $Q(\pi|\pi_0)$ *which can be found in equation* (4.7)*. One can also map* π_0 *to the* $\mathbb{S}_n^m$ *element* $[\boxed{1}, \boxed{2}, \boxed{3}, a, b, c, d, e, f]$*.*

In the IIL-GMM defined above, the modal permutation ranks all the landmarks *before* any other item. A symmetric model, in which the landmarks are ranked after all items can be defined similarly. Therefore, effectively, the central permutation of the IIL-GMM is formed similarly to the H-GMM^v, from the central permutation of the GMM, padded with $\text{id}_{\mathcal{L}}$ at the end (or, equivalently, at the top). This form of bias in the landmarks' location is a draw-back of this model.

The IIL-GMM does have remarkable flexibility. Namely, the place of the GMM model in the IIL-GMM can be taken by any ranking model. This larger class of models will be called *Independent Interleaving Landmark Models (IILM).*

Furthermore, without modification the IIL-GMM is a type of constrained Recursive Inversion Model (RIM) [35], in which the top node, denoted *root*, models the interleaving of $\pi_{\mathcal{E}} \in \mathbb{S}_n$ with the landmarks, the right child of *root* is the GMM, which is a type of RIM, and the left child is a dummy child, that always outputs $\text{id}_{\mathcal{L}}$. Hence, IIL-GMM inherits some of the flexibility and utility of the Recursive Inversion Model, including the ability to

quickly evaluate the probability of marginal rankings, and the ability to utilize partial rankings (with no further considerations for landmarks, as they always display strict ordering with all natural grades and each other).

4.2.5 Multiple Parameter IIL-GMM (DIL-GMM)

The multiple parameter IIL-GMM or the *Dependent Interleaving* GMM (DIL-GMM) expands on the IIL-GMM by introducing multiple parameters to define the interleaving.

The single parameter θ_0 of the IIL-GMM can be replaced by a parameter vector of dimension $\min(m, n)$. If $m < n$, then each landmark has a different parameter $\theta_{0,l}$ that controls its insertion location, conditioned on the locations of the previous landmarks. When $n < m$, the parameters are associated to inserting the items amidst the landmarks.

The DIL-GMM explores this modification, focusing on the former case where landmarks are inserted into non-landmarks ($m < n$), where the latter case becomes a H-GMM^v.

Similarly to the IIL-GMM, the probability of the interleaving is a function of the inversions w.r.t. landmark items. For a landmark $k \in \mathcal{L}$, define the number of inversions u_k as in equation (4.14), and u_k is sampled from a geometric distribution over its range.

Then P^{interl} can be derived recursing over k backwards. Starting with u_m, the last landmark, we have

$$P^{\text{interl}}(u_m|\theta_{0,m}) = \theta_{0,m}^{u_m}/Z_n(\theta_{0,m}), \tag{4.18}$$

which is identical to the factors of equation (2.9). Subsequent landmarks are interleaved so that $u_k \leq u_{k+1}$, hence

$$P^{\text{interl}}(u_k|u_{k+1}, \theta_{0,k}) = \theta_{0,k}^{u_k}/Z_{u_{k+1}}(\theta_{0,k}), \tag{4.19}$$

where the only change from equation (4.18) is the modification of the normalization constant to account for the variable range u_{k+1}. This displays a striking similarity between this stagewise interleaving model and the standard GMM^v. The probability of a permutation

$\pi \in \mathbb{S}_n^m$ is now given by

$$P^{\text{DIL}-\text{GMM}}(\pi) \;=\; P(\pi_{\mathcal{E}})P^{\text{interl}}(u_{1:m};\vec{\theta}_{0,1:m}) \;=\; P(\pi_{\mathcal{E}})P^{\text{interl}}(u_m|\theta_{0,m})\prod_{k=1}^{m-1}P^{\text{interl}}(u_k|u_{k+1},\theta_{0,k}). \tag{4.20}$$

As with the IIL-GMM the central permutation of the DIL-GMM will always place landmarks first and take the form $\pi_0 = [\text{id}_{\mathcal{L}}, \pi_{0|\mathcal{E}}]$. A symmetric model where landmarks are last in π_0 is also possible.

Example 11 *Let* $\pi_0 = [\boxed{1}, \boxed{2}, \boxed{3}, a, b, c, d, e, f]$ *with* $n = 6$ *and* $m = 3$*;* $\mathcal{E} = \{a, b, c, d, e, f\}$*,* $\mathcal{L} = \{\boxed{1}, \boxed{2}, \boxed{3}\}$*. Set* $\pi = [c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$*.*

The table below shows the construction of π *by interleaving* $\pi_{\mathcal{E}}$ *and* $\pi_{\mathcal{L}}$ *in the* DIL-GMM*. The construction of* $\pi_{\mathcal{E}}$ *is assumed to follow a standard* GMM*.*

item	u_k *range*	u_k	π	P^{interl} *Factor*
$\boxed{3}$	$0:6$	5	$[c, a, d, b, f, e]$	$\theta_{0,1}^5/Z_n(\theta_{0,1})$
$\boxed{2}$	$0:5$	3	$[c, a, d, b, f, \boxed{3}, e]$	$\theta_{0,2}^3/Z_5(\theta_{0,2})$
$\boxed{1}$	$0:3$	1	$[c, a, d, \boxed{2}, b, f, \boxed{3}, e]$	$\theta_{0,3}^1/Z_3(\theta_{0,3})$
–	–	–	$[c, \boxed{1}, a, d, \boxed{2}, b, f, \boxed{3}, e]$	–

(4.21)

The values of u_k *can be calculated following equation* (4.14)*, which for a single ranking can be calculated from* $Q(\pi|\pi_0)$ *which can be found in equation* (4.7)*.*

4.2.6 Alternative Models

Alternative 1 – GMM^s *or* GMM^v *based*

When s_j selects an element of π_0 that is in the set of landmarks, we remove specifically that landmark (rather than the first) and label it as such (rather than labeling it in order) allowing inversions over the elements of $\mathcal{L}$. This is no different from a standard GMM. The sample space would be $\mathbb{S}_{n+m}$ rather than $\mathbb{S}_n^m$. In consequence, the semantics of the landmarks

would be rather confusing; for instance, what is the rating of an item preceded by $\boxed{3}$ and followed by $\boxed{1}$? A model with similar properties can be obtained with GMM^v.

Alternative 2 – GMM^s *based*

When choosing an s_j that is in the set of landmarks, we remove specifically that landmark (rather than the first) but label it as the first available landmark, to avoid inversions between landmarks. In this case, the sample space remains $\mathbb{S}_n^m$.

When calculating the probability of a $\pi \in \mathbb{S}_n^m$, or the normalization constant, after the first landmark is been added to π, there are $m-1$ versions of π_0 to track, depending on s_j. This number grows $m-2$ fold with the next landmark, etc.

Alternative 3 – AL-GMM

This model samples from a GMM^v over $\mathcal{E} \cup \mathcal{L}$, but in each output π the landmarks are relabeled to have the identity permutation. Hence, any $\pi \in \mathbb{S}_n^m$ is obtained as the marginal over $m!$ permutations in $\mathbb{S}_{n+m}$. This model is interesting, but computation poses complex challenges, hence we leave it for future work. Note that this model can be derived as the marginal of any ranking model, hence we call it *Anonymous Landmark (GM) Model (AL-GMM).*

Related works - the Mallows Binomial Model

To our knowledge, the models developed in this paper are the first to model preference data expressed as mutually consistent ratings and rankings.

A previous attempt to combine ranking and rating data is the Mallows-Binomial (M-B) model [36]. This model is also a distribution over rankings and ratings, but unlike our models, it imposes consistency only at the parameter level, while in the individual samples, ratings are allowed to be inconsistent with the rankings.

The Mallows-Binomial model consists of a Mallows model for the rankings and a binomial

distribution on ratings, with dependence in parameterization between the two models. We can express the likelihood of a complete ranking and set of ratings $(\pi, \vec{r})$ as

$$P^{\mathrm{M-B}}(\pi) = \frac{\theta^{d(\pi,\pi_0)}}{Z(\theta)} \times \prod_{j=1}^{n} \binom{m}{r_j} p_j^{r_j} (1-p_j)^{m-r_j}. \tag{4.22}$$

In other words, the ratings are sampled from binomial distributions, $\mathrm{Binom}(m, p_j)$, while rankings follow a standard Mallows model parameterized by θ and π_0 with π_0 is constrained to sort $p_{1:n}$ in increasing order, $p_{(\pi_0)_i} < p_{(\pi_0)_{i+1}}$.

4.3 Maximum likelihood estimation of the L-GMM^s

Given a data set $\mathcal{D}$ consisting of a set of N permutations with landmarks of length $n+m$, we wish to find the L-GMM^s model that maximizes the data likelihood.

We will focus first on the log-likelihood of a single data point $\pi \in \mathbb{S}_n^m$. For notational simplicity, we define the function

$$t_j(\pi; \pi_0, \theta_j) = \sum_{l=k+1}^{m} \theta_j^{s_j^{kl}(\pi;\pi_0)} \tag{4.23}$$

for any rank j of π with $\pi(j) = k \in \mathcal{L}_{[1:(m-1)]}$ (all of the landmarks excepting the last). Thus, in this case, in equation (4.3), the factor $\sum_{l=k}^{m} \theta_j^{s_j^{kl}}$ is equal to $t_j(\pi; \pi_0, \theta_j)$. We also set $s_j(\pi; \pi_0) = 0$ for any rank j of π with $\pi(j) = k \in \mathcal{L}_{[1:(m-1)]}$ and $s_j(\pi; \pi_0) = s_j^{mm}$ whenever $\pi(j) = m \in \mathcal{L}$. We can then express the log-likelihood of π as

$$\ln P^{\text{L-GMM}^s}(\pi|\pi_0) = \sum_{j=1}^{n+m-1} s_j(\pi;\pi_0) \ln \theta_j + \sum_{j=1}^{n+m-1} 1_{[\pi(j)=k\in\mathcal{L}_{[1:(m-1)]}]} \ln t_j(\pi;\pi_0,\theta_j) - \sum_{j=1}^{n+m-1} \ln Z_{n-j}(\theta_j) \tag{4.24}$$

Proposition 4 *For any π and π_0 the log-likelihood function given by equation* (4.24) *is a concave function of variables $\theta_{1:n+m-1}$.*

Proof It is easy to see that $\ln P^{\text{L-GMM}^s}(\pi)$ is a sum of terms each depending on a single θ_j.

If $\pi(j) \notin \mathcal{L}$, then the contribution to the second term is zero and the remaining expression is concave in θ_j, as has previously been proven for the GMM^s[14, 33]. If on rank j we have a landmark, then by examining equation (4.3) we see that the corresponding factor in $P^{\text{L-GMM}^s}(\pi)$ is a sum of probabilities of individual outcomes. Each of these probabilities is a log-concave function of θ_j; it is known that summation preserves log-concavity [6, Chapter 3], hence the corresponding factor in the likelihood is log-concave, and therefore the entire likelihood is log-concave. □.

The log-likelihood of the entire data set $\mathcal{D}$, similarly a sum of concave terms, decomposes as

$$\sum_{\pi\in\mathcal{D}} \ln P^{\text{L-GMM}^s}(\pi) \;=\; \sum_{j=1}^{n+m-1} \ln\theta_j \sum_{\pi\in\mathcal{D}} s_j(\pi;\pi_0) + \sum_{j=1}^{n+m-1}\sum_{\pi\in\mathcal{D}} 1_{[\pi(j)=k\in\mathcal{L}]} \ln t_j(\pi;\pi_0,\theta_j) - N\sum_{j=1}^{n+m-1} \ln Z_{n-j}(\theta_j) \quad (4.25)$$

In what follows, we will first examine the ML estimation of the dispersion parameters $\vec{\theta}$ given a fixed π_0. As will be seen, the first term in the likelihood can easily be summarized with sufficient statistics, while the middle term for landmarks (excepting the last) cannot. We detail methods for summarizing the data for efficient calculation of this term, which in turn allows efficient estimation of $\vec{\theta}$ for a fixed π_0. Finally we build upon these steps with a search algorithm that utilizes these algorithmic innovations for efficient calculation of the log-likelihood in the search for the Maximum Likelihood π_0.

4.3.1 Optimizing θ_j for fixed π_0

It can be seen from equation (4.25) that, because the log-likelihood can be decomposed into a sum over the ranks j, the optimization of θ_j does not depend on any other θ_i values. The problem thus reduces to a univariate concave maximization.

A gradient descent method will be able to find the Maximum Likelihood estimate. However, with the sample space constrained to $\theta_j \in [0, 1]$, a zero-th order optimization [38] or

even binary search will work well too. They guarantee an arbitrary level of precision in a fixed number of steps, without requiring the tuning of step size.

4.3.2 Sufficient statistics for computing the first term of equation (4.25)

The first term in the expression of the log-likelihood can be calculated from sufficient statistics. This can be seen by comparing equation (4.24) with the expression of the log-likelihood for the GMM^s model in previous works [31]. It can easily be seen that the two expression are the same, therefore, as was shown previously [31], the matrices $Q_j(\pi)$ defined below are sufficient statistics for this term.

$$Q_j(\pi) \;=\; [Q_{j,ee'}]_{e\in\mathcal{E}\cup\{m\},e'\in\mathcal{E}\cup\mathcal{L}}, \quad Q_{j,ee}=0,\; Q_{j,ee'}=1 \text{ if } e\succ_\pi e', \pi(e)=j \text{ and } 0 \text{ otherwise.} \tag{4.26}$$

It is easy to see that

$$Q(\pi) \;=\; \sum_{j=1}^{n+m} Q_j(\pi), \quad \text{and} \quad s_j(\pi) \;=\; \sum_{e\succ_{\pi_0}e'} Q_{j;ee'}(\pi), \text{ with } e={\pi_0}^{-1}(j)\,. \tag{4.27}$$

With this decomposition, the sufficient statistics for all of $\mathcal{D}$ are given by

$$\bar{Q}_j = \frac{1}{N}\sum_{\pi\in\mathcal{D}} Q_j(\pi). \tag{4.28}$$

Let $\bar{s}_j(\pi_0) \;=\; \frac{1}{N}\sum_{\pi\in\mathcal{D})} Q_j(\pi;\pi_0)$. Then according to (4.5) and [31],

$$\bar{s}_j(\pi_0) \;=\; \sum_{i'<i} \bar{Q}_{j;ii'}(\pi_0), \tag{4.29}$$

where $\bar{Q}(\pi_0)$ is the matrix $\bar{Q}$ with rows and columns ordered by π_0. From a computational point of view, the matrices $\bar{Q}_{1:m+n-1}$ in (4.27) need to be computed only once. To compute $\bar{s}_{1:m+n-1}(\pi_0)$ for a given π_0, it suffices to read the entries in $\bar{Q}_j$ in the order prescribed by π_0, for a number of $(n+m)(n+m-1)/2$ operations, independently of N.

For the last landmark, the likelihood is functionally similar to the likelihood of the items in $\mathcal{E}$, and its contribution to the likelihood can be included in $\bar{s}_j(\mathcal{D})$ with the addition of a row in Q_j, as demonstrated above.

4.3.3 Data Structures for the second term of equation (4.25)

It remains to be shown how to efficiently compute the values $t_j(\pi; \pi_0, \theta_j)$ which determine the second term of equation (4.25). While we will not be able to summarize these terms with sufficient statistics, we can construct an intermediary data structure that will allow for efficient and algorithmically simple iteration over the data to calculate $t_j(\pi, \pi_0, \theta_j)$ for any π_0 and θ_j.

We construct tables $T_{j,k,l}$, where $j = 2 : n+m-1$ ranges over ranks, $k = 1 : m-1$ ranges over landmarks (excepting the last), and $l = k+1 : m$ ranges over the landmarks that follow k. For each rank j and for each landmark k that could be on this rank, we keep tables $T_{j,k,l}$ containing a row $T_{j,k,l}(\pi; \pi_0)$ for each permutation π in the data. This row contains a 1 for each item ranked by π between k and l, with the columns of the table ordered according to central permutation π_0. Tables are only constructed for the ranks j corresponding to landmarks $1 : m-1$.

Example 12 *For $\pi = (c\,\boxed{1}\,a\,d\,\boxed{2}\,b\,f\,\boxed{3}\,e)$, landmark $\boxed{1}$ is on rank $j = 2$; hence, the rows of $T_{2,\boxed{1},\boxed{2}}$ and $T_{2,\boxed{1},\boxed{3}}$ for central permutation $\pi_0 = [a, b, \boxed{1}, c, d, \boxed{2}, e, \boxed{3}, f]$ are given below.*

$$T_{2,\boxed{1},\boxed{2},\boxed{3}} = \begin{array}{c|ccccccccc} & a & b & \boxed{1} & c & d & \boxed{2} & e & \boxed{3} & f \\ \hline \boxed{2} & 1 & 0 & 1 & 0 & 1 & - & 0 & 0 & 0 \\ \boxed{3} & 1 & 1 & 1 & 0 & 1 & 1 & 0 & - & 1 \end{array} \tag{4.30}$$

Landmark $\boxed{2}$ is on rank j=5, meaning $T_{5,\boxed{2},\boxed{3}}$ is the only remaining entry in the table, with values

$$\begin{array}{c|ccccccccc} & a & b & \boxed{1} & c & d & \boxed{2} & e & \boxed{3} & f \\ \hline \boxed{3} & 0 & 1 & 0 & 0 & 0 & - & 0 & 1 & 1 \end{array} \tag{4.31}$$

No table is necessary for $\boxed{3}$ *since it is not followed by any other landmark.*

Note that for l increasing, the values in each column of $T_{j,kl}$ can only increase from 0 to 1 or stay the same.

When the columns of $T_{j,k,l}$ are sorted by a permutation π_0,

$$s_j^{kl}(\pi;\pi_0) \;=\; \sum_{i \succ \pi_0(e)} T_i^{j,k,l}(\pi;\pi_0) + (l-k), \tag{4.32}$$

Thus, $t_j(\pi;\pi_0,\theta_j)$ can be computed efficiently by traversing tables $T_{\pi^{-1}(k),k,(k+1):m}$ in column order π_0.

By equation (4.23), $t_j^m = 1$ hence no table is required for the last landmark, as its contribution to the likelihood is accounted for in the first term of equation (4.25). Moreover, on the very first and last ranks, only a restricted number of landmarks can be found; for example, only $\boxed{1}$ and $\boxed{2}$ can occupy rank 2. In general, for $j < m-1$, only $T_{j,1:j,:}$ are needed, for $j > n+m-1$, only $T_{j,n+m-j,:}$ are needed, and for $j = m-1 : n+m-1$ all $T_{j,1:m-1,:}$ are needed.

With this we can calculate $t_j(\pi;\pi_0,\theta_j)$ in equation (4.23) more efficiently when calculating the log-likelihood in equation (4.32).

Further optimization of the second term

The introduction of the landmarks introduces a $t_j(\pi;\pi_0,\theta_j)$ term for every landmark (excepting the last) for every ranking in the data set $\mathcal{D}$. For the purposes of fitting the maximal likelihood model to a given set of rankings, these terms will be repeatedly used in finding the optimum $\hat{\theta}_j$ parameters. The construction of the tables T allow for a more straightforward expression of the likelihood, but result in a data structure that is larger than the original data.

In order to speed up calculation on this more computationally intensive part of the likelihood (or its derivative), all necessary values from $\mathcal{D}$ are summarized as succinctly as possible.

This is initialized for a specific π_0 as will be shown. The second term of equation (4.25) requires every set of $s_j^{k,\cdot}$ values to calculate, thus we calculate and save these values once for a given π_0 for quick and repeated reference.

For a ranking π with landmark $k < m$ at index j, we save the values of s_j^{kl} with $l \in [k{+}1, m]$ in a vector $\mathcal{S}_{j,-}^{k,l}$, indexed the same way as s_j^{kl}, where j indexes the rank, $k \in [1, m{-}1]$ indexes for the landmark at $\pi^{-1}(j) = \pi_{\mathcal{L}}^{-1}(k)$, and $l \in [k{+}1, m]$ enumerates over the remaining landmarks in $\pi_{\mathcal{L}}$.

For a large dataset, the values of $\mathcal{S}_{j,-}^{k,l}$ for all rankings with landmark k at rank j are stacked into an $N_{j,k} \times (m-k)$ matrix. The addition of $N_{j,k}$ represents the number of rankings for which rank j contains landmark k. This ensures that, for a given rank/landmark pair (j,k), rather than enumerating over all rankings N in the data set $\mathcal{D}$ we instead only enumerate over those rankings which contain landmark k at rank j.

The calculation of the $\ln t_j(\pi_d; \pi_0)$ terms now becomes

$$\sum_{j=1}^{n+m-1} \sum_{\pi \in \mathcal{D}} 1_{[\pi(j)=k \in \mathcal{L}]} \ln t_j(\pi_d; \pi_0, \theta_j) \;=\; \sum_{j=1}^{n+m-1} \sum_{k=1}^{m-1} \sum_{N'=1}^{N_{j,k}} \ln \left(\sum_{l=k}^{m} \theta_j^{\mathcal{S}_{j,N'}^{k,l}} \right) \tag{4.33}$$

This is an area where some computational gains could be made under the right circumstances. Consider a simple case of two landmarks, where similar rankings produce rows of $\mathcal{S}$ that are identical at a given index j. If $\mathcal{S}_{j,1}^{1,1:2} = \mathcal{S}_{j,2}^{1,1:2} = (3,5)$ these repeated values could be instead stored in a $(n+m-1-j, n+m-1-j)$ matrix with index $(3,5)$ equal to 2, recording the number of instances of $\ln(\theta^3 + \theta^5)$ in the log-likelihood, and their contribution to the log-likelihood can be scaled by the number of repeated instances. This approach is best suited for cases where the number of landmarks m is low and the number of observations N is high, but won't be further considered here.

4.3.4 Summary of log-likelihood calculation

Combining Sections 4.3.2 and 4.3.3 with equation (4.25) the likelihood equation becomes

$$\sum_{\pi\in\mathcal{D}} \ln P^{\text{L-GMM}^s}(\pi) \;=\; \sum_{j=1}^{n+m-1}\left[N\ln\theta_j\left(\sum_{i'\succ i=\pi_0(j)}\bar{Q}_{j;ii'}(\pi_0)\right)+\sum_{k=1}^{m-1}\sum_{N'=1}^{N_{j,k}}\ln\left(\sum_{l=k}^{m}\theta_j^{\mathcal{S}^{k,l}_{j,N'}}\right)-N\ln Z_{n-j}(\theta_j)\right] \quad (4.34)$$

Estimation of the maximum likelihood $\hat{\theta}$ for a given π_0 requires straightforward convex optimization. When calculating the likelihood for a new ranking π_0, the columns of the tables T, and the rows and columns of the $\bar{Q}_j$ matrices must be reordered by the new π_0. In practice, these matrices stay in place, and we simply re-index the columns via pointers.

The values of $\mathcal{S}^{k,l}_{j,N'}$, also change with π_0; they must be recalculated in their entirety from the re-indexed tables T. This update represents the majority of the computational overhead in preparing to optimize $\hat{\theta}$ for a new π_0.

4.3.5 Finding the optimal π_0

In the Mallows and GMM, the search for the maximum likelihood π_0 is NP–hard. For these models, π_0 can be estimated by Branch and Bound methods [33], which assemble the modal ranking one item at a time. These algorithms are exact. Their run-time depends on the amount of consensus in the data; when $\mathcal{D}$ is concentrated around the Maximum Likelihood π_0, then the algorithms are efficient, otherwise they become intractable.

For the L-GMM^s the likelihood of a landmark at any given rank depends on the number of landmarks remaining and all of their ranks, values that are undetermined when assembling π_0 one rank at a time. While it would be possible to branch off of every single landmark configuration, this large branching factor would explode too quickly to be useful. Instead we utilize a stochastic search combined with a localized greedy search of $\mathbb{S}_n^m$ which has been demonstrated to work well for ranking models [35, 39].

The search begins by initializing a sampling model $(\pi_{\text{sample}}, \vec{\theta})$ with a uniform parameterization $\vec{\theta} = \vec{1}$. As the parameterization is uniform, π_{sample} is arbitrary. From the sampling distribution a ranking π_{new}, is sampled from $\mathbb{S}_n^m$. For π_{new}, the maximum likelihood estimate $\hat{\theta}(\pi_{\text{new}})$ is fit and the data log-likelihood is computed as demonstrated in previous sections. From π_{new}, a localized greedy search is done. Every adjacent $\pi_{\text{neighbor}} \in \mathbb{S}_n^m$ that can be created by a single inversion of adjacent elements in π_{new} is fit and the log-likelihood estimated. Let $(\pi_{\text{greedybest}}, \hat{\theta}(\pi_{\text{greedybest}}))$ be the neighbor of π_{new} with the highest likelihood. If this value is higher than that of $(\pi_{\text{new}}, \hat{\theta}(\pi_{\text{new}}))$, then $(\pi_{\text{new}}, \hat{\theta}(\pi_{\text{new}}))$ is replaced with $(\pi_{\text{greedybest}}, \hat{\theta}(\pi_{\text{greedybest}}))$ and the local greedy search is repeated. This continues until a ranking is found with no neighboring rankings producing a higher log-likelihood.

Then, in an annealing-like step, the new model is accepted probabilistically and becomes the new sampling model $(\pi_{\text{sample}}, \hat{\theta}(\pi_{\text{sample}}))$. This is repeated t_{max} times.

To prevent the annealing search getting stuck in local maxima, we repeat the above search k_{max} times, each time starting from a random π_{sample}, for a total of $k_{max} * t_{max}$ greedy searches. The annealing step also requires the selection of a temperature parameter β.

The procedure is detailed in Algorithm 8.

4.4 Maximum likelihood estimation of other models

While the Landmark GMM^s represents a significant deviation of fitting methods previously utilized for the GMM^s, the remainder of valid and tractable models remain much simpler to resolve. We detail here the steps needed for maximum likelihood estimation for completeness.

4.4.1 ML estimation for the IIL-GMM

As can be seen in equation (4.16), the log-likelihood of the data under the IIL-GMM decomposes into the log-likelihood of the permutation $\pi_{\mathcal{E}}$, the restriction of $\pi \in \mathbb{S}_n^m$ to $\mathcal{E}$ and the log-likelihood of the interleaving. Each of these terms can be maximized separately over its parameters.

Algorithm 8 Algorithm FIT L-GMM^s

Input Data set $\mathcal{D}$, consisting of N rankings from $\mathbb{S}_n^m$
Input annealing temperature parameter β, annealing steps per restart $\mathtt{k}_{\mathtt{max}}$, max restarts $\mathtt{t}_{\mathtt{max}}$
Initialize $\mathtt{Q}$ containing rank level inversion matrices $\mathtt{Q}_\mathtt{j}$ following Section 4.3.2
Initialize rank level tables $\mathtt{T}_\mathtt{j}$ following Section 4.3.3
$\pi_{\mathtt{best}} \leftarrow \mathtt{id}_\pi$
$\hat{\theta}_{\mathtt{best}} \leftarrow \overrightarrow{1}$
for $\mathtt{t} = 1, 2, ..., \mathtt{t}_{\mathtt{max}}$ **do** {iterate over restarts}
  $\pi_{\mathtt{sample}} \leftarrow \mathtt{id}_\pi$
  $\hat{\theta}_{\mathtt{sample}} \leftarrow \overrightarrow{1}$
  for $\mathtt{k} = 1, 2, ..., \mathtt{k}_{\mathtt{max}}$ **do** {iterate over stochastic search steps}
    $\pi_{\mathtt{new}}$ is sampled from L-GMM$^s(\pi_{\mathrm{sample}}, \hat{\theta}_{sample})$ following Section 4.2.1
    $\overrightarrow{s}, \mathcal{S} \leftarrow \mathtt{initalize}(\pi_{\mathtt{new}}, \mathtt{Q}, \mathtt{T})$ following Section 4.3.2 and Section 4.3.3
    $\hat{\theta}_{\mathtt{new}} \leftarrow \mathtt{fitTheta}(\overrightarrow{s}, \mathcal{S})$ following Section 4.3.1
    $\mathtt{greedyComplete} \leftarrow \mathtt{FALSE}$
    while $\mathtt{greedyComplete} == \mathtt{FALSE}$ **do** {greedy local search}
      $\pi_{\mathtt{greedybest}} \leftarrow \pi_{\mathtt{new}}$; $\hat{\theta}_{\mathtt{greedybest}} \leftarrow \hat{\theta}_{\mathtt{new}}$
      for $\mathtt{i} = 1, 2, ..., \mathtt{m} + \mathtt{n} - 1$ **do**
        if $\pi_{\mathtt{new}}(\mathtt{i}) \notin \mathcal{L}$ or $\pi_{\mathtt{new}}(\mathtt{i}+1) \notin \mathcal{L}$ **then**
          $\pi_{\mathtt{neighbor}} \leftarrow \pi_{\mathtt{new}}$
          $\mathtt{swap}(\pi_{\mathtt{neighbor}}(\mathtt{i}), \pi_{\mathtt{neighbor}}(\mathtt{i}+1))$
          $\overrightarrow{s}_{\mathtt{neighbor}}, \mathcal{S}_{\mathtt{neighbor}} \leftarrow \mathtt{initalize}(\pi_{\mathtt{neighbor}}, \mathtt{Q}, \mathtt{T})$
          $\hat{\theta}_{\mathtt{neighbor}} \leftarrow \mathtt{fitTheta}(\overrightarrow{s}_{\mathtt{neighbor}}, \mathcal{S}_{\mathtt{neighbor}})$
          if $\mathtt{P}(\mathcal{D}|\pi_{\mathtt{neighbor}}, \hat{\theta}_{\mathtt{neighbor}}) > \mathtt{P}(\mathcal{D}|\pi_{\mathtt{greedybest}}, \hat{\theta}_{\mathtt{greedybest}})$ **then**
            $\pi_{\mathtt{greedybest}} \leftarrow \pi_{\mathtt{neighbor}}$; $\hat{\theta}_{\mathtt{greedybest}} \leftarrow \hat{\theta}_{\mathtt{neighbor}}$
      if $\pi_{\mathtt{greedybest}} == \pi_{\mathtt{new}}$ **then**
        $\mathtt{greedyComplete} \leftarrow \mathtt{TRUE}$
      else
        $\pi_{\mathtt{new}} \leftarrow \pi_{\mathtt{greedybest}}$; $\hat{\theta}_{\mathtt{new}} \leftarrow \hat{\theta}_{\mathtt{greedybest}}$
    $\mathtt{u} \sim \mathtt{uniform}[0, 1)$
    if $\mathtt{e}^{-\beta(\mathtt{P}(\mathcal{D}|\pi_{\mathtt{new}}, \hat{\theta}_{\mathtt{new}}) - \mathtt{P}(\mathcal{D}|\pi_{\mathtt{sample}}, \hat{\theta}_{\mathtt{sample}}))} < \mathtt{u}$ **then** {"annealing" step}
      $\pi_{\mathtt{sample}} \leftarrow \pi_{\mathtt{new}}$; $\hat{\theta}_{\mathtt{sample}} \leftarrow \hat{\theta}_{\mathtt{new}}$
    if $\mathtt{P}(\mathcal{D}|\pi_{\mathtt{new}}, \hat{\theta}_{\mathtt{new}}) > \mathtt{P}(\mathcal{D}|\pi_{\mathtt{best}}, \hat{\theta}_{\mathtt{best}})$ **then** {save the best model so far}
      $\pi_{\mathtt{best}} \leftarrow \pi_{\mathtt{new}}$; $\hat{\theta}_{\mathtt{best}} \leftarrow \hat{\theta}_{\mathtt{new}}$
Output $\pi_{\mathtt{best}}$ and $\hat{\theta}_{\mathtt{best}}$

The search for $(\pi_0, \vec{\theta})$ amounts to estimating a GMM^s, which can be done via Branch and Bound [33] or by any other method for fitting ranking models.

Maximum Likelihood estimation for interleavings was introduced in previous works [30, 35] but we review it here. The likelihood of the interleaving is independent of the order of the non-landmark items, and depends only on the parameter θ_0 and of $u_{1:m}$ the number of items in $\mathcal{E}$ ranked after each landmark in an observed ranking π. For a landmark $k \in \mathcal{L}$, equation (4.14) defines u_k, counting inversions of π_0 w.r.t. landmark k. We can further note that $\sum_{k=1}^{m} u_k = d([\pi_\mathcal{L}, \pi_\mathcal{E}], \pi)$, the total number of inversions introduced by the interleaving. Then the likelihood of the interleaving becomes

$$\ln P^{\text{interl}}(\pi|\theta_0) = d([\pi_\mathcal{L}, \pi_\mathcal{E}], \pi) \ln \theta_0 - \ln Z_{n,m}(\theta_0) \tag{4.35}$$

where the denominator term $Z_{n,m}(\theta_0)$ is defined as in equation (4.16). The normalization constant is a product of log-concave functions and thus log-concave [30, 35].

For a set of rankings $\mathcal{D}$, the log-likelihood becomes

$$\ln \prod_{\pi \in \mathcal{D}} P^{\text{interl}}(\pi|\theta_0) = \left(\sum_{\pi \in \mathcal{D}} d([\pi_\mathcal{L}, \pi_\mathcal{E}], \pi) \right) \ln \theta_0 - N \ln Z_{n,m}(\theta_0). \tag{4.36}$$

The first term is a sum of all u_k terms over all rankings, which can be calculated from the inversion matrix $\bar{Q}$. The latter term in the likelihood depends on the data set only through N. Finding θ_0 amounts now to maximizing the above concave function. Note that this optimization, being independent of $(\pi_0, \vec{\theta})$, is done only once.

4.4.2 ML estimation for the DIL-GMM

From equation (4.20) it follows that the likelihood of the data given the DIL-GMM decomposes similarly to the IIL-GMM, into a factor for items $\mathcal{E}$, depending on π_0 and $\vec{\theta}$, and a factor for the interleaving of the landmarks $\mathcal{L}$, a function of θ_0 and the inversion counts $u_{1:m}$. As noted previously, the likelihood over the non-landmark items follows a standard GMM or

other models and its maximization is not discussed here.

The log-likelihood of the interleaving has two distinct terms. The first depends on the location of the last landmark and of its corresponding parameter $\theta_{0,m}$.

$$\ln P^{\text{DIL}-\text{GMM}}(u_m|\theta_{0,m}) = u_m \ln \theta_{0,m} - \ln(Z_n(\theta_{0,m})). \tag{4.37}$$

This term is no different from those of the standard GMM^v log-likelihood in equation (2.9). Thus we have a convex optimization to estimate $\theta_{0,m}$.

The log-likelihood for the subsequent mergers in the interleaving depends on the previous merger, with the value of u_k constrained to $0, \dots u_{k+1}$ and the normalization constant adjusted to account for this range, i.e.,

$$\ln P^{\text{DIL}-\text{GMM}}(u_k|\theta_{0,k}, u_{k+1}) = u_k \ln \theta_{0,k} - \ln(Z_{u_{k+1}}(\theta_{0,k})). \tag{4.38}$$

By analogy with equation (2.9) and Proposition 4, we see that P^{interl} can be expressed as a product of log-concave functions, yielding a concave log-likelihood. Additionally, for computational efficiency, we can simplify the r.h.s of equation (4.20), the log-likelihood of the interleaving at an index, as

$$\ln \prod_{\pi \in \mathcal{D}} P^{\text{DIL}-\text{GMM}}(u_k|\theta_{0,k}, u_{k+1}) = \sum_{\pi \in \mathcal{D}} u_k \ln \theta_{0,k} - \sum_{\pi \in \mathcal{D}} \ln(Z_{u_{k+1}}(\theta_{0,k})) \tag{4.39}$$

$$= \bar{u}_k \ln \theta_{0,k} - \sum_{j=0}^{j<n+k} N_{k,j} \ln(Z_{u_{k+1}}(\theta_{0,k})), \tag{4.40}$$

where the $N_{k,j}$ counts occurrences of $\ln(Z_{u_{k+1}}(\theta_{0,k}))$ with $u_{k+1} = j$. This summarizes the N denominators at an index into no more than $n+k$ potential distinct denominators. The first term in equation (4.39) can be computed as usual from the matrix Q.

4.4.3 ML estimation for the H-GMM^v

The maximum likelihood estimation algorithm for the H-GMM^v is a slightly modified version of standard GMM^v model fitting.

This model can be constructed as a valid GMM^v model where some parameters are fixed, i.e., $\vec{\theta}_{\mathcal{L}} = 0$ for all landmarks, allowing utilization of techniques for fitting a GMM^v.

The expression of the likelihood can be seen in equation (4.11) and differs from equation (2.9) for the standard GMM only in the normalization constant. This change does not affect the log-concavity and independencies of the remaining parameters, thus one can proceed as usual with Branch and Bound or similar approaches to estimate the remainder of the modal ranking [33].

4.5 Experiments on synthetic data

As the class of rankings with landmarks represents a new type of ranking, no analysis exists to ensure their recoverability under stochasticity. Likewise, the introduction of the greedy local search makes predicting the runtime of the L-GMM^s maximum likelihood estimation algorithm impossible. We set out to address these issues with testing on synthetic data.

4.5.1 Goals

The first task of our synthetic data tests is to ensure the recoverability of the model under stochasticity with different structures, item and landmark counts, and sample sizes. We test this by producing a variety of models and seeking to recover their original parameterization for various samples sizes.

The latter task of estimating the runtime of the reference ranking search is done empirically, again varying the sample sizes, structure of reference rankings, and number and ratio of items and landmarks. This allows for tracking the number of steps needed to complete the search algorithms.

4.5.2 Data generation

Synthetic data sets consisted of either $n = 4$, 12, or 16 items and either $m = 2$, 4, or 6 landmarks. For each of the scenarios, sample sizes ranged from $N = 100$, doubling in size up to a maximum size of $N = 3200$. Values for $\vec{\theta}$ were sampled uniformly between $[.75, .9]$. Note that values of $\theta = 1$ correspond to the uniform distribution over $\mathbb{S}_n^m$, thus the range of θ values we use represents models with weak consensus, and reasonably difficult to recover. The modal ranking π_0 were sampled uniformly from $\mathbb{S}_n^m$.

Under each permutation of item count n, landmark count m, and sample size N, 25 models were generated and sampled to the requested sample size. A total of 50 ($t_{max} = 5$) simulated annealing steps with localized greedy search were permitted on each sample, with restarts every 10 steps ($k_{max} = 10$).

Restarts were recorded separately (as they do not depend on one another), meaning that the results can also be seen as containing 5 independent ($t_{max} = 1$), 10 step ($k_{max} = 10$) annealing runs (without restarts) on each data set. When searching for the maximum likelihood model only the model with the best scoring log-likelihood is kept, but analyzing the individual restarts helps evaluate the stochastic search.

4.5.3 Results - recoverability

Our primary goal in the synthetic experiments is recovering the modal ranking π_0, but under stochasticity we are not guaranteed that the maximum likelihood estimate will be the true modal ranking. This leads to two scenarios that can be considered a success for our search procedure: recovering the original modal ranking π_0 or discovering a ranking with a higher likelihood than the true π_0. The latter evaluates the ability of the ML algorithm to fit the data, the former also evaluates the sample complexity of the models.

In all 6,750 total restarts of all runs in every possible scenario, there was exactly one single 10 step annealing search which produced a π_0 with lower likelihood than the true modal ranking. Thus, a single failure of the annealing algorithm, and no failure of the search

algorithm, as this was just one restart of the five used to fit this data set. Unsurprisingly, this case occurred for a large number of items $n+m = 16+2$ and a small sample size ($N = 100$).

The remainder of the restarts returned either a modal ranking that produced better likelihood than the true ranking or captured the true modal ranking. Figure 4.1 summarizes these results for a selection of synthetic sampling parameters, ranging from the smallest and largest sets of items. It can be seen that at smaller sample sizes, the true π_0 was less likely to coincide with the maximum likelihood estimate. This is more common when the number of items plus landmarks is large, but even for $n + m = 16 + 6$, at sample sizes of $N = 1600$ the true modal ranking was returned in every single restart of the annealing steps.

4.5.4 Results - search runtime

While the choice of annealing steps and frequency of restarts can be controlled, the number of steps in the greedy neighbor search can't be easily estimated. The number of potential neighbors for each ranking increases linearly with the number of items and landmarks in the model, but the depth of the search before terminating is unknown.

We assess the length of the greedy search in Figure 4.2, which shows the average number of steps taken by the greedy search for a single annealing step, versus numbers of items and landmarks. The growth in the number of steps per item appears linear in m, with the slope and intercept tied to the number of items n. The case where $n + m = 18$ is particularly revealing in this respect. This behavior can at least partially be explained by the fact that more potential neighbors in the greedy search are rejected for containing inverted landmarks when the proportion of landmarks is large.

4.6 Experiments on real data

We explore two real world datasets derived from continuous scores in two contexts. The first, a large dataset consisting of scores of many different jokes is used to create multiple data subsets to compare the ability of our models to reflect held out testing data accurately [15].

A second set of data, consisting of Elo scores of NBA teams [40, 10], helps to highlight

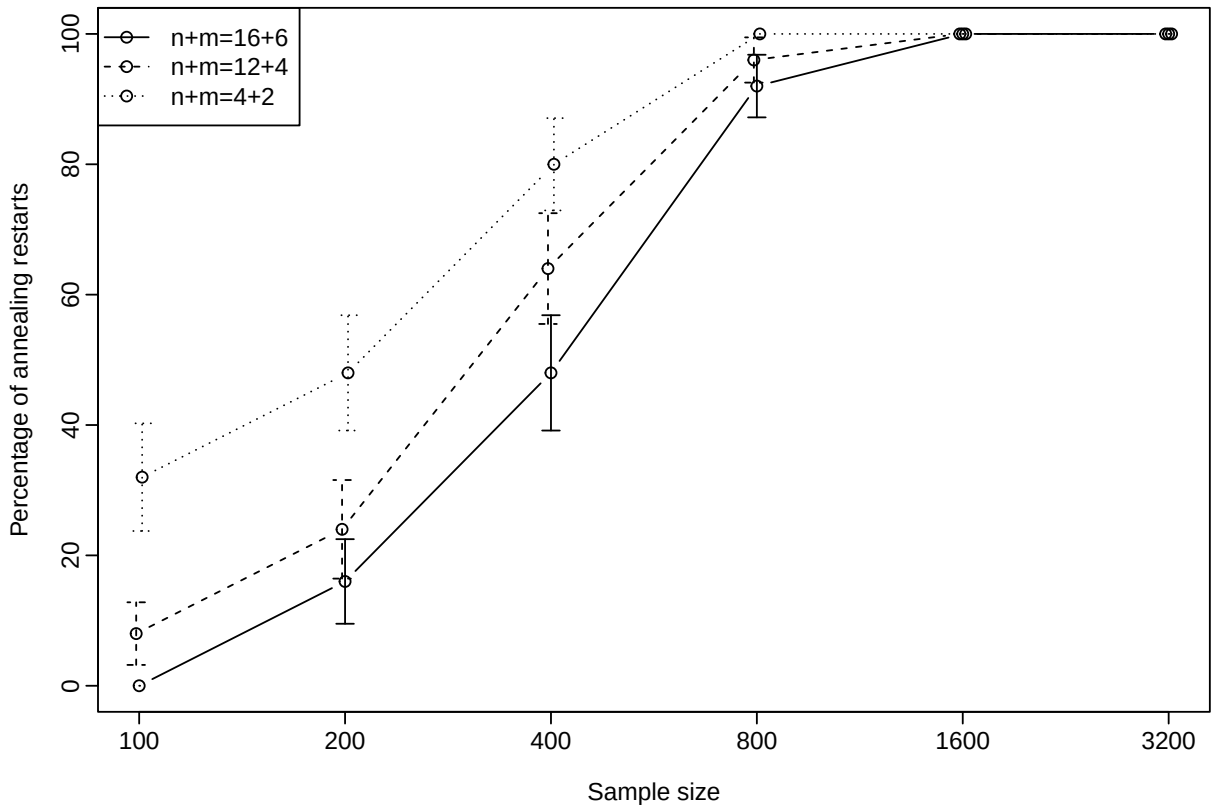


Figure 4.1: Recoverability of the true modal ranking as sample sizes change. Each point in this plot represents the average of 125 restarts; 5 restarts on 5 repeated maximum likelihood estimates on 5 data sets. Bars represent a 95% confidence interval. Points are slightly displaced on the x-axis for readability.

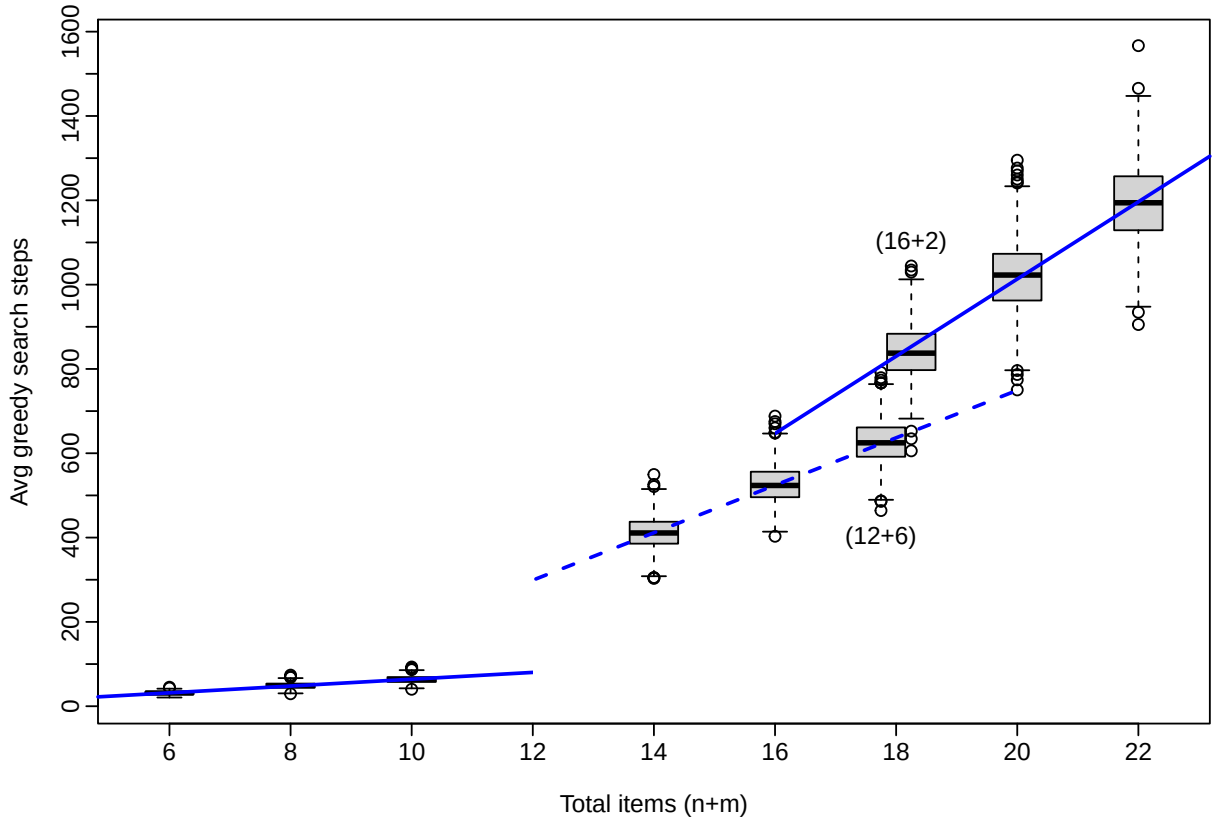


Figure 4.2: The average number of greedy steps per annealing step on synthetic data for different total item $(n + m)$ counts, grouped by value of n. The box plots for cases ($n = 12$, $m = 6$) and ($n = 16$, $m = 2$), both with $n + m = 18$, have been separated, labeled, and slightly displaced on the x-axis for readability. Linear regression lines are added to each group. Each box plot represents a total of 750 averages.

the ability of the model to capture consensus ranking (where it exists), alongside the joke rating dataset. We seek to explore (some of) the results of these models to discuss their use in inference and highlight the information added by the addition of rating data to rankings.

4.6.1 Jester data

For testing the model on real world data, we turn to the Jester data set. The Jester data set was gathered over a period of 4 years (April 1999 - May 2003). Respondents were prompted to view a wide range of jokes (100+ in total) and assess their level of humor using a continuous score between -10 and 10 (with higher scores corresponding to jokes the user enjoyed more). Scores were not calibrated to specific values excepting the midpoint 0, thus users were encouraged to rate things more accurately via an otherwise unmarked slider. Users can be seen expressing preferences over the entire space, implying that users were able to express not just a positive or negative preference, but also the magnitude of that preference [15].

Not every respondent scored every joke, but many dense subsets could be constructed using the 100 jokes. We curated these data as follows. First, a subsample of 25 jokes were selected for which over 4,000 individuals provided scores. From this set of 25 jokes, we selected four sets of ten jokes, and four sets of fifteen jokes yielding eight distinct sets of items $\mathcal{E}$. For each of these joke sets $\mathcal{E}$, four disjoint data sets were created, each consisting of $N = 1,000$ users, with three used as training data sets, and the fourth left out for testing purposes.

We convert the continuous scores to ratings by adding $m = 3$ landmarks at scores of $-2.5, 0, +2.5$ or scores of $-5, 0, +5$. We convert the same continuous scores to rankings by sorting the jokes by increasing score.

Combining the eight joke sets, four data samples, and two differing landmark locations yields a total of 48 distinct training data sets and 16 associated testing data sets.

We compare the L-GMM^s, IIL-GMM and DIL-GMM on these data. Following the procedures detailed for maximum likelihood estimation, the estimation of the modal ranking

was done via searches with $t_{max} = 50$ annealing steps, restarting from a uniform distribution every $k_{max} = 50$ steps, for a total of 2500 annealing steps. The model best fitting the training data is then evaluated using the left out testing data.

We also model the data using the Mallows-Binomial model, on the data sets with landmarks at $-5, 0, 5$. As the Mallows-Binomial model is less flexible, for a fair comparison we also implemented the IIL-MM, an independent interleaving model with a Mallows model (rather than a GMM) over $\mathcal{E}$, and compared it against Mallows-Binomial.

Results

In Figure 4.3, we compare the log-likelihood of the L-GMM^s model against both the IIL-GMM and DIL-GMM on held out testing data. It is immediately obvious that the log-likelihood of the L-GMM^s is much higher in all cases. This means the added complexity in the L-GMM^s, and specifically, the ability to control the location of the landmarks w.r.t. the items in $\mathcal{E}$, are needed in the modeling of real data.

We can see for all $n = 15$ item models produce less variation in the difference between log-likelihoods. We don't see a discernible difference between the performance of the IIL-GMM and the DIL-GMM. A careful assessment shows that most of the $\theta_{0,j}$ interleaving parameters DIL-GMM are close 1, with only the first landmark $\boxed{1}$ typically having a different θ value. A value of $\theta = 1$ makes it likely that later landmarks are ranked after the majority of items. Likewise the single θ_0 parameter for the IIL-GMM were also close to 1 in all instances. These details help to highlight the usefulness of the L-GMM^s in not just more accurately modeling the data, but also in producing models that are more informative as to the relation between landmark and items.

To learn more about how these models are alike and differ, we can compare the modal ranking on items, $\pi_{0|\mathcal{E}}$. Note that while the L-GMM^s contains a modal ranking with interspersed landmarks, both the IIL-GMM and the DIL-GMM place all landmarks at the front of the ranking. The modal ranking of the L-GMM^s on one of the data sets is shown below,

with both rating ± 2.5 and rating ± 5 landmarks.

$$\pi_0{}^{\pm 2.5} = [9, 1, 8, 7, 3, 4, \boxed{1}, \boxed{2}, \boxed{3}, 6, 10, 5, 2] \tag{4.41}$$

$$\pi_0{}^{\pm 5} = [9, 1, 8, 7, 3, \boxed{1}, 4, \boxed{2}, 6, 10, \boxed{3}, 5, 2] \tag{4.42}$$

$$\pi_{0|\mathcal{E}} = [9, 1, 8, 7, 3, 4, 6, 10, 5, 2] \text{ in both cases} \tag{4.43}$$

Here, $\pi_{0|\mathcal{E}}$ the ordering of the items is the same for all models; this was not the case for all data sets (nor do we necessarily expect the models to match). However, this case highlights some differences of L-GMM^s over the other models explored. Even when all models return the same information about the relationship between items, only the L-GMM^s (and H-GMM^v) includes a modal ranking that is explicitly informative about the ratings.

Comparing modal rankings for different landmark locations also highlights an important consideration. When converting a data set from continuous scores to ratings, care must be taken to find landmark locations that are informative. The choice of a landmark at 0 here makes sense, given the context, as separations between 'good' and 'bad' jokes.

An analysis of the resulting modal rankings of the L-GMM^s results reveals a tendency to produce 'empty' grades. This was most frequent for the case of landmarks at ± 2.5, as can be seen above. For the cases with landmarks at ± 5, empty grades were less common, and they usually involved placing the last landmark as the last item, implying no jokes received the lowest rating.

Figure 4.4, helps illustrates the issues. For many of the jokes, very few respondents provided ratings in the range $(-2.5, 2.5)$ (excepting the noticeable modes seen at 0), and likewise we see cases where $(-10, -5)$ or $(5, 10)$ are similarly sparse. This highlights the need for consideration in choosing the grades when they are not a-priori defined.

In comparison with the IIL-MM, the Mallows-Binomial model had much lower likelihood: the IIL-MM $\log_{10}$-likelihood was on average 2.94 higher (sd=0.068) for $n = 10$, and 5.32 higher (sd=0.061) for $n = 15$.

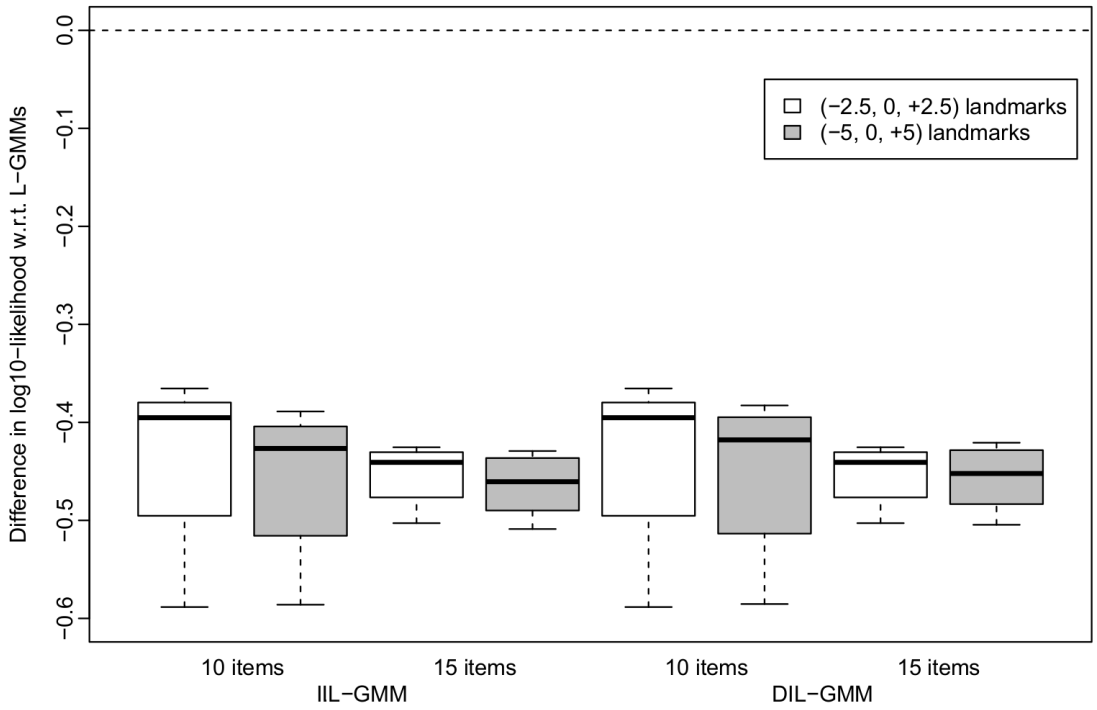


Figure 4.3: The difference in average testing log-likelihood between the L-GMM^s and both the IIL-GMM and DIL-GMM on the Jester data set. Differences below 0 indicate that the log-likelihood was higher for the L-GMM^s. Each boxplot represents the results on 12 data sets.

The IIL-MM and Mallows-Binomial are identical in the way they model the rankings, and both model the ratings independently of π. But the models in this paper enforce consistency between r and π, while Mallows-Binomial allows inconsistent (π, r) pairs. Consequently, Mallows-Binomial spreads its probability mass over a much larger combinatorial space, and this is the main cause for its weak performance on these data. Given that the IIL-MM represents the least flexible of the models developed in this work, we do not pursue comparisons between Mallows-Binomial and the other models.

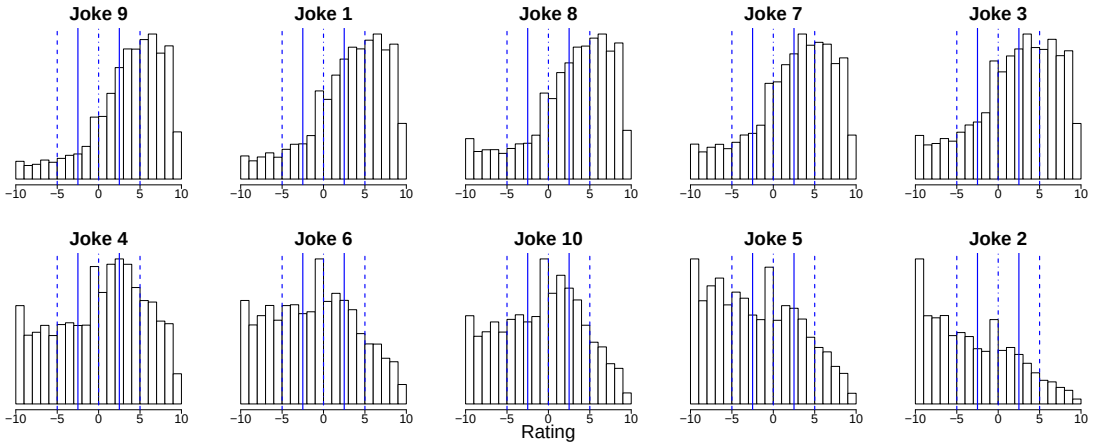


Figure 4.4: Ratings provided by approximately 50,000 respondents for each of the 10 jokes of one of the joke sets. Also indicated are lines at the potential landmark locations for ratings of either ± 5 or ± 2.5 and 0. The jokes are ordered according to the modal ranking $\pi_{0|\mathcal{E}}$ found in equation (4.43).

4.6.2 NBA Elo data

Elo ratings, originally developed for competitive chess ratings and ranking, offer a system for constructing continuous ratings of competitors in a game [9]. Ratings start at 1500 and are zero sum, meaning any game resulting in a win will raise the Elo ratings of the winner and lower the rating of the loser by the same amount. In the same vein, the gap between ratings of two competitors determines the change in Elo, with the lower rated player winning resulting in a bigger change than the higher rated player winning.

For the purposes of this research, the space of chess players is too large, with far too few games per player. Fortunately the construction of Elo ratings is not specific to chess and can be applied to any competitive environment. To this end the authors turn to the NBA Elo ratings assembled by fivethirtyeight.com [40, 10].

It should be noted that the authors of this paper are not familiar with the intricacies of the NBA, its teams, its general structure, or its usual schedules. As such, the sampling schemes utilized do not account for the tournaments, playoffs, or other details specific to the NBA. Rather, the authors approached it as they would any other chronologically correlated

Eastern Conference

Code	Team	Code	Team
BOS	Boston Celtics	CLE	Cleveland Cavaliers
NYK	New York Knicks	DET	Detroit Pistons
PHI	Philidelphia 76rs	IND	Indiana Pacers
CHI	Chicago Bulls	MIL	Milwaulke Bucks
		ATL	Atlanta Hawks

Western Conference

Code	Team	Code	Team
MIN	Minnestoa Timberwolves	LAL	Los Angeles Lakers
POR	Portland Trailblazers	PHO	Phoenix Suns
UTA	Utah Jazz	SAC	Sacramento Kings
GSW	Golden State Warriors	DAL	Dallas Mavericks
LAC	LA Clippers	HOU	Houston Rockets
		SAS	San Antonio Spurs

Table 4.1: The remaining teams in each respective conferences dataset and their associated 3 letter codes.

observations.

NBA teams have not been constant over the decades, so a consistent set of teams over a fixed span of years was required. The NBA is primarily divided into two conferences, east and west. For both of these conferences, the present day teams were selected. As some of these teams have only been formed in the previous few decades, the youngest teams were dropped to produce a large enough sample of games. The remaining teams can be seen in Table 4.1.

The Eastern Conference was narrowed down to 9 teams, with games starting in the latter half of 1976 (the start of the '76-'77 season). The Western Conference contained 11 teams and starts with the season in 1989. Both sets of teams existed through the end of the available data (end of the 2014-'15 season) and still exist at time of our analysis.

Observations in the 538 dataset [10] consist of individual games and their changes in Elo rating. Each game is listed twice, once for each team playing. This list of games was reduced

to only those games where a team that will remain in the sample is the primary team listed. As a consequence, games with teams playing against teams outside of the sample have a single observation listed, while games consisting of two teams within the sample list the game twice. What results after pruning can be viewed as a list of shifting Elo scores for all relevant teams.

From this list of Elo scores, a systematic sample of each 25th observation was taken. For the games associated with these changes, the ratings and rankings of all teams at the end of the games were recorded. This resulted in a total of 1216 rankings with ratings for the Eastern Conference and 983 rankings with ratings for the Western Conference.

It should be noted that these rankings are clearly highly autocorrelated, with successive rankings representing only a slight deviation from the previous rankings. The models developed here assume independence between rankings, thus the authors acknowledge the shortcoming of this analysis and examine this data despite.

While it is difficult to directly measure the level of autocorrelation of rankings, the Kendall tau distances between successive rankings do hint somewhat at this problem. Within the east conference, all sampled rankings had an average distance of 17.7, while the distance between successive rankings averaged 8.3 (10.5 at lag 2, 11.0 at lag 3). The Western Conference averaged a distance of 26.1 between all rankings, but only 11.0 between successive rankings (14.4 at lag 2, 16.7 at lag 3).

There exists a baseline value that makes a natural candidate as a landmark; an Elo of 1500 represents a player (or team) just joining the ranking system. With the Elo rating system being a zero-sum system, it means that the average Elo rating should remain around 1500. This isn't exactly the case, as some teams have been dropped from the NBA (taking their Elo delta with them). None the less, a landmark at 1500 will represent a baseline team.

The authors of the NBA Elo ratings note key Elo breakdowns for every 100 points, with 1600 described as "playoff bound" and 1400 described as "in the lottery". More extreme landmarks at 1700 and 1300 would also appear appropriate, but a brief analysis of the ratings in the data show a distribution that is approximately normal, centered around 1500, with a

standard deviation only slightly higher than 100. This implies that ratings less than 1300 or greater than 1700 are fairly rare, which would (uninformatively) produce landmarks at the beginning and end of the reference ranking.

The chosen landmarks for both models were placed at Elo ratings of 1600, 1500, and 1400. Elo ratings are intended to accurately represent the relative skill of competitors, but in a way that is relatable to humans. Thus landmarks separating the midpoint (1500) and scores at steps of 100 are sensible.

Results

Maximum likelihood estimation algorithms were run with a maximum run length of 100, and a total of 250 restarts, for a total of 25000 models fit to each dataset. In the absence of sufficient data for validation or testing, we report the model that produce the lowest likelihood on training data.

We can see the results for the Western Conference in Table 4.2. The results are organized according to the reference permutation π_0, with listed landmarks at Elo ratings of 1600, 1500, and 1400. Columns display the marginal distributions of grade within the rankings, and immediately the results appear to support the resulting reference permutation.

For every team except for one (DAL), the team's grade position in the reference permutation corresponds to their plurality grade within the rankings as a whole. The exception to this is DAL, which can be seen to have a bimodal distribution favoring Elo ratings greater than 1600 or less than 1400.

The listed average Elo ratings also help to highlight the usefulness of this model over other approaches. Calling attention to LAL, SAC, LAC, and MIN, we see that each have an average Elo rating that would place them in a grade lower (for LAL) or higher (for SAC, LAC, and MIN), than where they appear in the reference ranking. If we instead look to the distribution of grade membership for these teams, we note that their placement corresponds to the modal rating. Thus the ranking model is able to capture trends in scores without being strongly influenced by the extreme values skewing the distribution of scores, and more

closely reflects the mode of the rankings informed by ratings, rather than the averages of the scores on their own absent the rankings. Likewise some teams, such as PHO and POR, are out of order in their average score, again highlighting the models preference for reflecting the mode of rankings.

Looking finally to the values of $\hat{\theta}$ found for the Western Conference, we see values between (.7,.9). It is important to remember that contrary to how the values are displayed, each θ value corresponds to a rank, not an item. Thus a lower value of $\hat{\theta}_i$ indicates a strong preference for the item at rank i to come from the items earlier in the remaining π_0, while the very last value of $\hat{\theta}_{13} = .962$ indicates that the distinction between the last two items is (nearly) irrelevant (a common occurrence in our real world examples).

The results for the Eastern Conference found in Table 4.3 prove to be far less conclusive, but this same data breakdown helps to highlight why. Compared to the teams found in the Western Conference, the distribution of grades for Eastern Conference teams is much less peaked. This implies a more uniform relationship between the teams and landmarks, and results in an overall less informative π_0 containing many empty grades.

The $\hat{\theta}$ values found in the Eastern Conference data also reflect the more uniform, less distinguished rankings. Aside from a handful at early ranks, many of the $\hat{\theta}$ values were estimated at or near 1. This implies a uniform preference between items at the respective rank, while the $\hat{\theta}$ values near .75 for later ranks provide little information after the numerous uniformly chosen items at previous ranks.

We can also compare and contrast these results using the scatter plots of Elo scores found in Figure 4.5 and Figure 4.6. We see much the same trends we see in the tables, but with further distribution details on the scores. While it was noted that Elo scores collectively for all teams are normally distributed, we see that for the joint distributions of team scores, this distributional assumption does not generally hold. Thus, while the initial marginal distribution of scores might lead one to utilize a Thurstonian model for these rankings, such a model would operate under incorrect assumptions. Looking first to Figure 4.5, we can see that the data lends itself fairly well to a clear ordering. We would expect the upper triangle

of the matrix of scatter plots to contain more points above the diagonal, a reflection of the calculation of the inversion matrix Q.

Looking at the first team, the San Antonio Spurs (SAS), we see a consistent trend of points above the diagonal with a spattering of points crossing downward. These downward trends correspond to one of two (short) spans of losses the SAS suffered in otherwise very strong set of seasons where their rating rarely dropped below 1500. We can also see how other teams scores change alongside - while the Los Angeles Lakers (LAL) maintain a high score through the periods of SAS losses, we note that the Dallas Mavericks (DAL) appear to suffer in the same timespans as SAS, with scores aligning along the diagonal.

Teams like the Dallas Mavericks (DAL) highlight the usefulness of our models when there is a consensus ordering. We can closely see that the rows or columns containing Dallas have a clear disparity of weight across the diagonal, despite the varied Elo scores of DAL. The scores themselves in aggregate cannot capture the relative trends between teams like the inversions can.

In contrast the eastern conference teams found in Figure 4.6 do not display the same strong trends, indicating a lack of consensus ordering. We see much less clear distinction across the diagonal of each plot, with many plots indicating an almost even distribution of points across the diagonal, with little to no trend in ordering of teams. Compared with the trends seen in Figure 4.5, where we see that teams often maintain longer win streaks, and maintain more consistent scores.

π_0	SAS	LAL	1600	UTA	POR	PHO	HOU	DAL	1500	GSW	1400	SAC	LAC	MIN
Thetas	0.736	0.834	0.841	0.788	0.725	0.701	0.770	0.808	0.788	0.727	0.718	0.834	0.962	NA
Proportion Grade 1	73.45%	48.73%	–	35.50%	27.16%	39.27%	26.25%	35.20%	–	7.63%	–	16.28%	10.89%	6.61%
Proportion Grade 2	20.14%	29.91%	–	44.46%	49.85%	43.74%	50.76%	23.60%	–	21.57%	–	13.12%	15.67%	23.30%
Proportion Grade 3	2.64%	14.55%	–	15.97%	15.26%	15.77%	22.48%	13.53%	–	44.46%	–	34.38%	33.37%	22.58%
Proportion Grade 4	3.76%	6.82%	–	4.07%	7.73%	1.22%	0.51%	27.67%	–	26.35%	–	36.22%	40.08%	47.51%
Average Elo	1626	1578	–	1564	1549	1570	1553	1504	–	1460	–	1456	1437	1426

Table 4.2: A breakdown of the distribution of items in each grade for the NBA West conference Elo dataset.

π_0	BOS	DET	CHI	1600	1500	PHI	IND	MIL	NYK	ATL	CLE	1400
Thetas	0.834	0.914	0.992	0.988	0.968	0.997	0.906	0.718	0.766	1.000	1.000	NA
Proportion Grade 1	32.89%	22.78%	23.93%	–	–	20.81%	13.32%	17.11%	13.57%	12.42%	12.25%	–
Proportion Grade 2	25.33%	33.47%	30.02%	–	–	34.29%	35.61%	27.96%	32.32%	39.06%	32.15%	–
Proportion Grade 3	32.73%	25.16%	27.71%	–	–	26.56%	41.78%	37.99%	36.68%	33.06%	30.92%	–
Proportion Grade 4	9.05%	18.59%	18.34%	–	–	18.34%	9.29%	16.94%	17.43%	15.46%	24.67%	–
Average Elo	1538	1512	1517	–	–	1502	1503	1495	1491	1497	1480	–

Table 4.3: A breakdown of the distribution of items in each grade for the NBA East conference Elo dataset.

4.7 Conclusion

We created a model for simultaneously reflecting the preference information present in both rankings and ratings. To this end we have created the class of rankings with landmarks, a unique ranking space with increased numbers of constraints in the form of landmarks.

For this new space of rankings we propose a number of models that can handle rankings with landmarks. For our each of our models we develop or adapt methods for maximum likelihood estimation with tractable overhead in computational complexity. For some of these models we show that the task can be completed with simple convex optimization, while others require methods similar to previous approach such as Branch & Bound [33], or the simulated annealing search of the RIM [30].

We emphasize models that are inferentially informative as to the relationship between rankings and landmarks, but a number of natural extensions can forgo ease of inference in favor of more accurate representations of the underlying population. This could be easily achieved with the interleaving approaches to modeling landmarks with rankings, the IIL-GMM and DIL-GMM. One could move the location of the interleaving to any point in the reference ranking aside from the start, or alternative (or additionally) allow the dispersion parameters of the interleaving to exceed 1, allowing for preference of right items over left.

It has also been shown by previous works that Mallows-type models can utilize conjugate priors on both the modal rankings π_0 and the parameters θ [13, 32, 33]. This presents a natural extension to the work here, and may allow for Bayesian inference of models with appropriately defined prior distributions.

These models are meant to serve as building blocks for more in depth analysis. As with other consensus ordering ranking models, in a population displaying multimodal preference, mixture models can be utilized to both more accurately reflect the population and cluster similar preferences for each model in the mixture. Such mixtures are a natural reflection of the underly population, such as in elections where voters politics lead them to prefer specific parties of candidates over others [17], or the sushi data where the data author suggests that

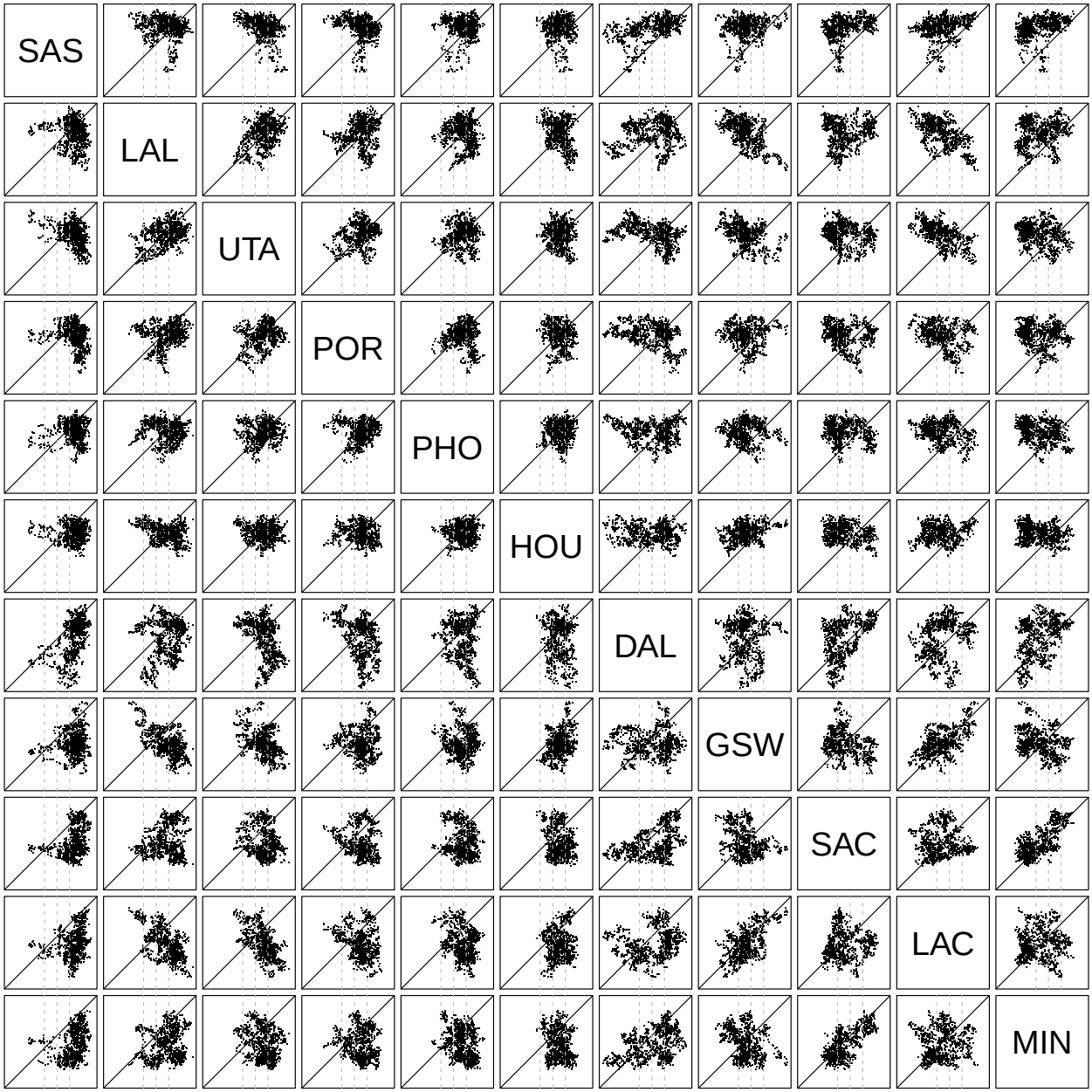


Figure 4.5: The pairwise scattering of Elo scores (range approximately 1110 to 1790) for teams in the western conference, ordered according to the reference ranking π_0. Vertical lines indicate landmark scores, which are removed from the horizontal for readability but replicated across the figure diagonal. Where points lie in respect to the solid diagonal line represents a preference for one team over another, thus this collection of scatter plots also captures the inversion matrix Q.

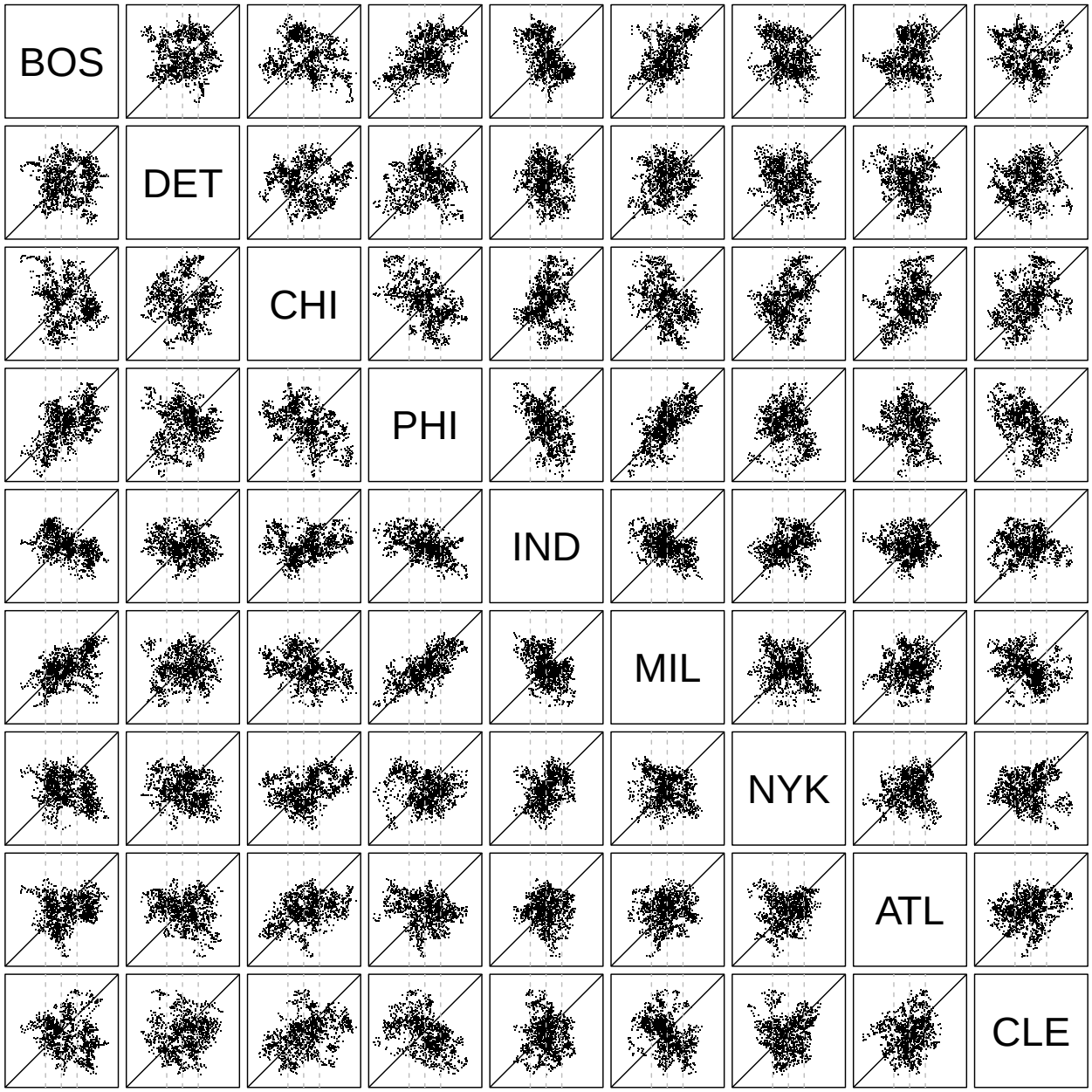


Figure 4.6: The pairwise scattering of Elo scores (range approximately 1170 to 1835) for teams in the eastern conference, ordered according to the reference ranking π_0. Vertical lines indicate landmark scores, which are removed from the horizontal for readability but replicated across the figure diagonal. Where points lie in respect to the solid diagonal line represents a preference for one team over another, thus this collection of scatter plots also captures the inversion matrix Q.

respondents are likely to display preferences differently based on geography [25].

We also develop these tools to enable and demonstrate the benefit of collecting more rich preference data allowing future researchers interested in assessing preference of large populations to more accurately and completely assess those preferences.

Chapter 5

DISCUSSION

5.1 Methodological contributions

The class of exponential ranking models contain a number of desirable properties that we sought to take advantage of to produce more flexible, accurate, and useful models for analysis of preference data. These models have been shown to capture consensus ranking, representing the consensus ordering in the underlying parametrization of the models [33]. Likewise the dispersion parameters in these models can be interpreted as dispersion for an item, a rank, or a set of inversions depending on the type of model being used.

We sought to expand these models to the class of partial rankings, a class of rankings containing total orders but allowing for a more diverse, more accurate, and more realistic representation of preferences. These rankings have been studied in various limited contexts for other models [27, 23, 31]. We have expanded the theory of the RIM to include the class of partial rankings, and have gone into great detail on the necessary steps, methods, and techniques needed for efficient estimation of the models from partial ranking data [34].

Alongside the introduction of partial rankings we have also detailed methods for estimating the marginal distribution of the rank and inversion matrices for a RIM. The techniques developed for this task allow for the estimation of conditional rank and inversion matrices for partial rankings, which can be viewed as marginal distribution estimation problems on a more limited RIM [35].

We demonstrated the use of the RIM for partial rankings on array of partial ranking datasets, and confirmed the efficacy of model recovery by comparing total orders and artificial partial rankings on real world data. We provided examples of the types of inference that achieved with such models in our analysis of these results.

In the analysis of rating data expressed as partial rankings, we exposed a shortcoming of this method. While rankings and ratings both represent methods of presenting preferences, the use of one over the other, or the conversion of one into the other, inherently carries a loss of some information provided in the preferences. We sought to address this shortcoming with the introduction of rankings with landmarks.

These new types of rankings carry with them a new set of challenges. Namely, rankings that contain non-invertible items in the ranking are ill suited to most ranking models. While models such as the Riffle Independence Model could be utilized for such rankings, these models are extremely limited in their usefulness due in part from their own massive flexibility. Thus we define a number of models constrained to the specific task of reflecting rankings containing non-invertible landmark items. These landmark ranking models can be utilized to describe the shared information in rankings and ratings, representing a contribution to the small space of ranking-rating models [36], and the only model ensuring consistency in rankings and ratings.

As the L-GMM^s builds upon the class of Generalized Mallows Models, they allow for the same interpretations to be conducted as any standard GMM. The construction inherently captures the underlying modal ranking of the population, summarizing the preferences into a single preference order reflecting the population mode [33]. Defined around this mode, rankings deviate according to parameters that are likewise themselves interpretable, reflecting the probability of a particular rank containing later elements of the reference ranking.

As before we demonstrated these methods on both synthetic and real world data, analyzing the efficacy of these models in both capturing the underlying consensus of the population, and representing that consensus in a way that reflects the underlying magnitude of preferences alongside the relative preferences.

We present these tools in to demonstrate and encourage more in depth analysis of preferences via more thorough, accurate, and complete representations of those preferences utilizing full rankings, partial rankings, and ratings. These tools allow for those expressing preferences to do so in a way not artificially constrained, thus producing more rich and meaningful

results.

5.2 Significance and future work

This work set out to demonstrate the need for and usefulness of more rich preference ranking data. Using the tools developed for analysis on this rich preference data, we have done exactly that.

Looking first to the expansion of the Recursive Inversion Model for partial rankings, we demonstrated on total orders of sushi preference data [25] that losing information in the form of partial rankings did not greatly impede our ability to recover the model at any point before the level of partialization became unreasonable. On the Meath election data [17] we found that the bias in the resultant model when looking only at total orders lead to considerable differences when compared to a truly random sample from the whole population of voters.

Analyzing rating data as rankings helped to highlight the usefulness of reflecting preferences under different paradigms. By constructing the ratings as partial rankings and analyzing them with the RIM, we revealed the dependency structure of preferences for social survey questions [19] and popular movie preferences [20]. This has opened the door to further RIM analyses of the wide range of partial ranking data and ratings that can similarly be expressed as partial rankings.

With the introduction of rankings with landmarks we have constructed an entirely new preference representation that can reflect the relative preferences in rankings while capturing the absolute preferences of ratings. Utilizing this rich preference information, we constructed rankings with landmarks for scores of jokes [15] and NBA team Elos [10, 40]. We demonstrated how such ranking models can reflect the underlying distribution of rankings without, for example, letting extreme skew in Elo scores influence the rankings of teams, but while simultaneously using those scores as ratings to inform the rankings.

This work overall represents a considerable contribution to the field of ranking models, expanding the modeling of preferences to new data paradigms and constructing new paradigms that required novel models thus allowing for entirely new inferences, and new

ways to represent preferences of populations.

There yet remain a number of problems that could serve to further expand upon the works herein.

5.2.1 L-GMM^s for partial ranking data

Partial rankings can be inferred, as was demonstrated, from various sources of ordinal rating datasets. The preservations of the magnitude of ratings in such a context was a major motivating factor in the original formulation of rankings with landmarks. Such a rating system carries with it inherent landmarks, and due to the popularity of ordinal rating data, there is a surfeit of available data sources. In the context of the L-GMM^s, partial rankings present additional challenges.

The task of likelihood estimation is fairly straight forward algorithmically, but follows a fairly brute force approach. Consider a partial ranking derived from ratings of the form $\sigma = (E_1, E_2, E_3)$ where $r(e) = k, e \in E_k$. The placement of landmarks in such a context is natural, with a landmark falling between each grade. Thus the partial ranking with landmarks takes the form $\sigma = (E_1, \boxed{1}, E_2, \boxed{2}, E_3)$.

This presents an obvious advantage; the treatment of landmarks does not appear to change with or without the presence of partial rankings, as landmarks will always fall between grades consisting of equally rated and ranked items. This implies that no additional considerations need to be made for landmarks beyond what the L-GMM^s is already managing.

We must then consider what the likelihood is for the partial component of the ranking. Consider a ranking σ containing $\sigma = (..., \boxed{\text{k-1}}, E_k, \boxed{\text{k}}, ...)$, with $E_k = (e_1, e_2, e_3)$ for which we must find the marginal likelihood over E_k. For some reference ranking π_0, assume that $\pi_0^{-1}(e_i) = j_i$ and w.l.o.g. that $j_i < j_{i+1}$, after the addition of any landmarks or items preceding E_k in σ. Lastly let the parameters $(\theta_1, \theta_2, \theta_3)$ represent the dispersion parameters for the ranks of σ containing E_k.

We can then resolve marginal probability of E_k by enumerating all $|E_k|!$ potential con-

figurations of the items (e_1, e_2, e_3).

$$\begin{aligned} P(E_k|\sigma,\pi_0,\boldsymbol{\theta}) =& P((e_1,e_2,e_3)|\sigma,\pi_0,\boldsymbol{\theta}) + P((e_1,e_3,e_2)|\sigma,\pi_0,\boldsymbol{\theta}) + P((e_2,e_1,e_3)|\sigma,\pi_0,\boldsymbol{\theta}) \\ &+ P((e_2,e_3,e_1)|\sigma,\pi_0,\boldsymbol{\theta}) + P((e_3,e_1,e_2)|\sigma,\pi_0,\boldsymbol{\theta}) + P((e_3,e_2,e_1)|\sigma,\pi_0,\boldsymbol{\theta}) \end{aligned} \tag{5.1}$$

$$\begin{aligned} =& (\theta_1^{j_1}\theta_2^{j_2-1}\theta_3^{j_3-2} + \theta_1^{j_1}\theta_2^{j_3-1}\theta_3^{j_2-1} + \theta_1^{j_2}\theta_2^{j_1}\theta_3^{j_3-2} \\ &+ \theta_1^{j_2}\theta_2^{j_3-1}\theta_3^{j_1} + \theta_1^{j_3}\theta_2^{j_1}\theta_3^{j_2-1} + \theta_1^{j_3}\theta_2^{j_2}\theta_3^{j_1})/(Z_{n'}(\theta_1)Z_{n'-1}(\theta_2)Z_{n'-2}(\theta_3)) \end{aligned} \tag{5.2}$$

While such a brute force approach to calculating the marginal probability of a partial ranking in an L-GMM^s appears to present an intractable computational overhead, for datasets with sufficiently small $|E_k|$ a number of considerations could be made to improve calculation. For any set of items E_k, given the rank of each item in the rating, clear patterns of terms in the exponents will repeat themselves for every ranking of similar size. This implies that the codes that define each ranking can be enumerated via brute force.

For a large dataset, methods of optimizing this particular calculation could greatly improve overhead. Independence between individual rankings is maintained, and the independence of the s_j codes that define the ranking outside of E_k remain. This implies that enumerating all codes consistent with E_k can still be done tractably (for small $|E_k|$), and the marginal probability of a partial ranking with landmarks can be found.

While the overhead may be manageable with respect to calculating the probability of partial rankings with landmarks, the likelihood calculation for optimizing the parameters introduces a dependence structure that appears to be intractable. While the likelihood can certainly be calculated, the ability to perform maximum likelihood estimation would require the simultaneously optimization of, in the example presented above, $(\theta_1, \theta_2, \theta_3)$. For a large dataset of many items, it is unlikely that any parameters would remain independent in all samples, thus one would be tasked with optimizing $\vec{\theta}$ in its entirety.

5.2.2 Expanding the L-GMM^s for d-way decomposition

It was previously shown for Riffle Independence Models, a super class containing the RIM and thus Mallows Models as a subclass, that any Riffle Independence Model that can be expressed as a hierarchical decomposition can also be expressed as a d-way decomposition [22]. For the purposes of the Riffle Independence Models, such decompositions caused a non-tractable increase in the number of parameters needed to describe the model.

The Landmark Generalized Mallows Model presents a simply parameterized approach to producing a decomposition consisting of a structure $(m, 1, 1, ..., 1)$ that merges landmark items with a large number of individual items without inverting the landmark elements. This only introduces $n + m - 1$ dispersion parameters, and the reference ranking π_0, as opposed to the factorial number of parameters needed for the Riffle Independence Model.

Such a limited decomposition presents similarly limited use cases, but a natural extension of the approach used in the construction of the L-GMM^s exists. Consider the case where there is one set of landmarks $\mathcal{L}$, but rather multiple independent sets of landmarks $\mathcal{L}_i$. Each individual set of landmarks would then behave the same; for example when sampling, selecting any landmark from a set implies the selection and removal of the first landmark in the set. One could express the likelihood of a ranking with d sets of landmarks as

$$P^{\text{L-GMM}^s}(\pi|\pi_0, \vec{\theta}) = \frac{1}{\prod_{j=1}^{n+m-1} Z_{n+m-j}(\theta_j)} \cdot \prod_{j=1}^{n+m-1} \begin{cases} \theta_j^{s_j(\pi)} & \text{if } \pi^{-1}(j) \in \mathcal{E} \\ \sum_{l=k}^{m_1} \theta_j^{s_j^{kl}} & \text{if } \pi^{-1}(j) = k \in \mathcal{L}_1 \\ \sum_{l=k}^{m_2} \theta_j^{s_j^{kl}} & \text{if } \pi^{-1}(j) = k \in \mathcal{L}_2 \\ \dots & \\ \sum_{l=k}^{m_d} \theta_j^{s_j^{kl}} & \text{if } \pi^{-1}(j) = k \in \mathcal{L}_d. \end{cases} \tag{5.3}$$

As before, the $s_j(\pi)$ terms are same as for the standard GMM^s, namely the sum of row $\pi(j)$ of $Q(\pi; \pi_0)$ at stage j. When the k-th landmark in $\mathcal{L}_i$ is rank j, the probability of picking it is the total probability of picking any of the $m_i - k + 1$ remaining landmarks; thus,

the variables s_j^{kl} are the s_j's of the landmarks $l = k : m_i$ of the landmark set $\mathcal{L}_i$. The now $(n-1) + m_1(m_1-1)/2 + ... + +m_d(m_d-1)/2$ variables

$$(s_j,\, j \in \pi_{\mathcal{E}}^{-1};\, s_j^{kl}, \text{ for } j \in \pi_{\mathcal{L}_1}^{-1},\, 1 \leq l \leq k \leq m_1; ...;\, s_j^{kl}, \text{ for } j \in \pi_{\mathcal{L}_d}^{-1},\, 1 \leq l \leq k \leq m_d)$$

represent the *code* of π.

This represents a decomposition of n individual items with $\sum_{i=1}^{d} m_i$ items in sets of various sizes. Items within each set cannot inverted, representing a valid riffle shuffle between all the elements of the d-way decomposition. This can be formalized further by removing the individual items and instead representing them as sets of landmarks of size 1. As previously noted, the last landmark of any set is indistinguishable from a non-landmark item and can be treated as such, thus this modification costs nothing.

One could then imagine a Riffle Independence Model of arbitrary branching factor, which can handle branches of arbitrary size with overhead that grows tractably in the size of decomposed sets. Each node in the decomposition would be represented by a set of $\boldsymbol{\theta}$ of length equal to the number of items being interleaved minus 1, and a reference ranking.

The search for the reference ranking in such a model could also be considerably simplified. As any ranking inverting any items within any of the landmark sets is ruled out, each landmark set could be considered as only a single item repeated multiple times in the reference ranking. Thus the space of possible rankings is limited not to the factorial many orderings of all items, but the combinatorial many orderings of sets of items.

Such an approach would still carry many computational challenges, including the enumeration of each individual landmark set similarly to how such enumeration was handled for the single set of landmarks used here. There also remains the problem of determining the best decompositions, a problem that also hindered the fitting of Riffle Independence Models and required simple decompositions to compensate [21, 23].

A more likely case for such a tool is one in which the decomposition lends itself naturally from the data. The authors of this work do not consider such datasets, nor do they seek to

hypothesize what such datasets may look like, but rather propose this as a potential method of creating arbitrary ranking models with non-invertible subsets of items for any applications in which such methods would be useful.

5.2.3 Recursive Inversion Models with predictors

Mixture models represent a natural extension to any statical model over a population that is believed to be better represented by a mixture of subpopulations. This can be seen in the context rankings in cases like the Meath election data [18], as well as other contexts [32]. In the context of ranking models, such mixtures can be fairly complex, with each subpopulation being identified with a unique reference ranking over the models. For a complexly structured model such as the Recursive Inversion Model, such mixtures can more accurately reflect the underlying population, but direct comparison of the models for subpopulations can be difficult.

One approach one might take to producing more informative RIMs for comparison may be to constrain the structure and reference ranking of the model. For example, in the context of the ISSP dataset [19], one might be compelled to constrain models for men and women to a single structure and reference ranking, but let individual parameters represent dispersion for the two subsets. Such a fit can be easily achieved by simultaneously optimizing a model over both subsets of the data on a shared structure.

Such an approach introduces some issues. Where one subset of the data may prefer items on the left over items on the right for a specific decomposition, the other subset may invert these preferences. This forces the model to parameterize one of the subsets canonically, constraining the other to uniform interleavings to accommodate the inverted preferences.

Relaxing the canonicalization requirement opens the door to more flexible fitting of underlying subpopulations, but hints at a great class of potential RIMs. We first revert to an alternative parameterization, replacing the dispersion parameters θ_i with a new parameterization of the form $e^{-\theta_i}$. This does not produce any significant change in the likelihood, representing purely a monotonic change in the parameter definition.

In such a parametrization, canonical models are defined for $e^{-\theta_i} \in [0, 1)$ as before, but can alternatively be defined defined around $\theta_i \in (0, \infty)$. In this context, a value of $\theta_i = 0$ implies a uniform dispersion, and a value of $\theta_i < 0$ implies that items on the right come before items on the left.

Keeping to the example of the ISSP data, again split on men and women as before, we can define the node level dispersion parameters as $\theta_i = \beta_{0,i} + \beta_{1,i} 1_{female}$ where the variable 1_{female} takes value 1 for rankings from female respondents and 0 otherwise. Alternatively one can view this as having a $\theta_{i,male}$ and $\theta_{i,female}$, and the parameterization can be found by optimizing both subsets over a fixed structure but allowing for either $\theta_{i,\cdot}$ to take values over the entire real line.

This raises the question; can more complex models be fit that introduce multiple parameters per ranking? For the case of fitting parameters to disjoint subsets of the population the task is fairly trivial, fitting the $\theta_{i,k}$ directly for each subset and ignoring the extended parameterization. Of more interest would be the handling of continuous predictors and non-disjoint discrete predictors, which would allow for rich models on which inference could be performed.

In such a model, predictors with positive parameterization would correspond to a stronger preference of left items over right, while predictors with negative parameterization would correspond to the opposite. The allowance of non-canonicalization and the modification of the parameterization both serve to ensure the space for optimizing the parameters is unconstrained. This could allow for simple coordinate descent methods for optimizing parameters, but it presently remains an open question whether such methods would remain computationally tractable for real world datasets.

BIBLIOGRAPHY

[1] Modeling preferences that integrate ratings with rankings. *Journal of Machine Learning Research*, In review.

[2] G.E. Andrews. *The Theory of Partitions*. Cambridge University Press, 1985.

[3] J. Bartholdi, C. A. Tovey, and M. Trick. Voting schemes for which it can be difficult to tell who won. *Social Choice and Welfare*, 6(2):157–165, 1989.

[4] J. Bennett and S. Lanning. The netflix prize. In *Proceedings of the KDD Cup Workshop 2007*, pages 3–6, New York, August 2007. ACM.

[5] Ulf Böckenholt. Applications of thurstonian models to ranking data. In *Probability Models and Statistical Analyses for Ranking Data*, pages 157–172, New York, NY, 1993. Springer New York.

[6] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[7] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[8] Persi Diaconis. Group representations in probability and statistics. *Lecture Notes-Monograph Series*, 11:i–192, 1988.

[9] Arpad E. Elo. The proposed USCF rating system. *Chess Life*, 22(8):242–247, 1967.

[10] Reuben Fischer-Baum and Nate Silver. The complete history of the nba, May 2015.

[11] Zack Fitzsimmons and Martin Lackner. Incomplete preferences in single-peaked electorates. *CoRR*, abs/1907.00752, 2019.

[12] M. A. Fligner and J. S. Verducci. Distance based ranking models. *Journal of the Royal Statistical Society B*, 48:359–369, 1986.

[13] Michael Fligner and Joseph Verducci. Posterior probabilities for a consensus ordering. *Psychometrika*, 55:53–63, 02 1990.

[14] Michael A. Fligner and Joseph S. Verducci. Multistage ranking models. *Journal of the American Statistical Association*, 83(403):892–901, 1988.

[15] Kenneth Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4:133–151, 07 2001.

[16] I. C. Gormley and T. B. Murphy. A latent space model for rank data. In *Proceedings of the 24th Annual International Conference on Machine Learning*, pages 90–102, New York, 2007. ACM.

[17] Isobel Claire Gormley and Thomas Brendan Murphy. A latent space model for rank data.

[18] Isobel Claire Gormley and Thomas Brendan Murphy. Exploring voting blocs within the irish electorate: A mixture modeling approach. *Journal of the American Statistical Association*, 103(483):1014–1027, 2008.

[19] ISSP Research Group. International Social Survey Programme: Family and changing gender roles IV - ISSP 2012, 2016.

[20] F. Maxwell Harper and Joseph A. Konstan. The MovieLens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4):19:1–19:19, 2015.

[21] Jonathan Huang and Carlos Guestrin. Riffled independence for ranked data. In *Advances in Neural Information Processing Systems 22*, pages 799–807. Curran Associates, Inc., 2009.

[22] Jonathan Huang and Carlos Guestrin. Uncovering the riffled independence structure of ranked data. *Electron. J. Statist.*, 6:199–230, 2012.

[23] Jonathan Huang, Ashish Kapoor, and Carlos Guestrin. Riffled independence for efficient inference with partial rankings. *Journal of Artificial Intelligence Research*, 44:491–532, 2012.

[24] Jonathan Huang, Ashish Kapoor, and Carlos Guestrin. Riffled independence for efficient inference with partial rankings. *CoRR*, abs/1401.6421, 2014.

[25] Toshihiro Kamishima. Nantonac collaborative filtering: recommendation based on order responses. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 583–588. ACM, 2003.

[26] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[27] Guy Lebanon and Yi Mao. Non-parametric modeling of partially ranked data. In *NIPS*, 2007.

[28] R.D. Luce. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959.

[29] C. L. Mallows. Non-null ranking models. *Biometrika*, 44:114–130, 1957.

[30] Christopher Meek and Marina Meila. Recursive inversion models for permutations. In *Advances in Neural Information Processing Systems*, pages 631–639, 2014.

[31] Marina Meilă; and Le Bao. An exponential model for infinite rankings. *Journal of Machine Learning Research*, 11(113):3481–3518, 2010.

[32] Marina Meilă and Harr Chen. Dirichlet process mixtures of generalized mallows models. In *Proceedings of the 26st Conference on Uncertainty in AI*, Los Angeles, California, 2010. AUAI Press.

[33] Marina Meilă, Kapil Phadnis, Arthur Patterson, and Jeff Bilmes. Consensus ranking under the exponential model. In *Proceedings of the 23rd Conference on Uncertainty in AI*, volume 23, 2007.

[34] Marina Meilă and Annelise Wagner. Recursive inversion models for partially ranked data. Revised for resubmission for review.

[35] Marina Meilă, Annelise Wagner, and Christopher Meek. Recursive inversion models for permutations. *Statistics and Computing*, 32(4), 2022.

[36] Michael Pearce and Elena A. Erosheva. A unified statistical learning model for rankings and scores with application to grant panel review. *Journal of Machine Learning Research*, 23(210):1–33, 2022.

[37] R. L. Plackett. The analysis of permutations. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(2):193–202, 1975.

[38] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 3 edition, 2007.

[39] Frans Schalekamp and Anke van Zuylen. Rank aggregation: Together we're strong. In Irene Finocchi and John Hershberger, editors, *Proceedings of the Workshop on Algorithm Engineering and Experiments, ALENEX 2009, New York, New York, USA, January 3, 2009*, pages 38–51. SIAM, 2009.

[40] Nate Silver and Reuben Fischer-Baum. How we calculate nba elo ratings, May 2015.

[41] Gábor Takács, István Pilászy, Bottyán Németh, and Domonkos Tikk. Matrix factorization and neighbor based algorithms for the netflix prize problem. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, RecSys '08, page 267–274, New York, NY, USA, 2008. Association for Computing Machinery.

Appendix A

APPENDIX

A.1 ISSP Questions

Table A.1 contains the full set of questions that were included in the analysis of the ISSP Family and Changing Gender Roles survey [19].

0. A working mother can establish just as warm and secure a relationship with her children as a mother who does not work.
1. A pre-school child is likely to suffer if his or her mother works.
2. All in all, family life suffers when the woman has a full-time job.
3. A job is all right [sic], but what most women really want is a home and children.
4. Being a housewife is just as fulfilling as working for pay.
5. Both the man and woman should contribute to the household income.
6. A man's job is to earn money; a woman's job is to look after the home and family.
7. Married people are generally happier than unmarried people.
8. People who want children ought to get married.
9. It is all right [sic] for a couple to live together without intending to get married.
10. Divorce is usually the best solution when a couple cant seem to work out their marriage problems.
11. One parent can bring up a child as well as two parents together.
12. A same sex female couple can bring up a child as well as a male-female couple.
13. A same sex male couple can bring up a child as well as a male-female couple.
14. Watching children grow up is life's greatest joy.
15. Having children interferes too much with the freedom of parents.
16. Children are a financial burden on their parents.
17. Having children restricts the employment and career chances of one or both parents.
18. Having children increases peoples social standing in society.
19. Adult children are an important source of help for elderly parents.

Table A.1: A full list of questions found in the ISSP survey on Family and Changing Gender Roles.