Inference Algorithms for the Multiplicative Mixture Mallows Model

***Abstract*—A popular approach to obtain a consensus ranking from ranking data is based on the probabilistic, distance-based Mallows model comprising of a modal permutation and dispersion parameters. Often, the population consists of several subpopulations. As a result, finite mixture models are used to distinguish latent sub-groups of individuals in a heterogeneous population. Given a finite number of subpopulations each based on the Mallows model, a popular inference approach is the computationally intensive expectation maximization algorithm for additive models. We address the drawbacks of this model using a novel *multiplicative mixture Mallows model* (`M4`). Given complete ranking observations from a heterogeneous population, we derive inference algorithms for the joint estimation of the parameters and the consensus rankings of the component distributions. We numerically validate the permutation estimation performance of the proposed algorithms on synthetic datasets. We also demonstrate the goodness-of-fit using the Bayesian information criterion and the integrated complete likelihood on the real-world APA and Sushi datasets.**

I. Introduction

Ranking is an essential ingredient of a gamut of applications such as electoral preference learning, personalized advertisement targeting, recommender systems, etc. Typically ranking data is obtained from surveys and market studies where the participants provide complete or partial list of items in the order of preference, which we refer to as a *rankings*. The design of a ranking system involves learning an appropriate ranking model based on the nature of the rankings and subsequently obtaining the consensus ranking that best agrees with the given sample permutations.

Typically, given n items, the survey participants independently generate rankings of length $t \leq n$. There is substantial amount of work on models, algorithms and guarantees that consider observations in the form of pair-wise preferences ($t = 2$) [1], [2], [3] and for rankings of length $2 \leq t \leq n$ [4], [5]. We are interested in the Mallows model which is a probabilistic distance-based modeling approach to analyze rankings [5], [6]. Here, each observation is regarded as a *noisy* version of the ground truth permutation whose probability of occurence is inversely related to the distance between itself and the ground truth permutation [5]. The inference problem in Mallows' model consists of estimating the model parameters and the underlying consensus ranking using the *Kendall-Tau distance* metric [5], [7].

Real world survey data often consists of samples from *heterogeneous* subpopulations. It is pragmatic to model the global population as consisting of subpopulations of rankers who share a common preference behaviour and thus, each cluster is characterized by a unique consensus ranking. In the context of the permutation-based Mallows model, additive mixture models are commonly employed, where the expecatation maximization (EM) algorithm is used for inference [8], [9]. The critical drawback of these inference techniques is that the expectation requires weighted summations over all $n!$ possible permutations. For large n, the complexity of EM algorithm is prohibitive and the convergence is slow [9]. Furthermore, in [10], the authors note that in high-dimensions, the posterior distribution behaves similarly to the individual components of the additive mixture density, leading to a vacuous modelling of heterogeneous population.

To address the computational shortcomings of additive mixture models, we propose a novel, low-complexity *multiplicative mixture Mallows model* (`M4`), where the overall mixture distribution is an *exponentially weighted product* of the component distributions, akin to the product of experts model [10]. In the theory of hypothesis testing, such a product mixture distribution is a *tilting* from one component distribution to another [11, Chapter 11]. Based on the concept of tilting, we propose a sample-wise cluster assignment where every sample is a weighted product of two out of M component distributions. Our contributions are as follows:

- We propose a greedy approach for joint estimation of the dispersion parameters and consensus ranking based on the concave convex procedure (CCCP) and the Stochastic Gradient Descent (SGD) algorithm rendering the overall algorithm to be implementation friendly.
- We validate our methods on real-world datasets such as the APA and the Sushi datasets using the Bayesian information criterion (BIC) and Integrated Complete Likelihood (ICL) [12] and demonstrate the goodness-of-fit of `M4` vis-à-vis additive mixture models and single component distributions. We also contrast the computational ease of `M4` vis-à-vis additive models.
- We also validate the consensus ranking recovery performance of the proposed algorithms on synthetic datasets.

II. The Mallows Model and Preliminaries

We now describe the Mallows model, hence laying the foundation for inference in the `M4`. Let π denote the ranking of a survey participant when queried over n items in the decreasing order of preference, i.e., π is a permutation over the set $[n] = \{1, \ldots, n\}$, where $\pi(l)$ denotes the *rank* of l in π. Let Ω be the set of all $n!$ possible rankings and $d(\cdot, \cdot)$ be a distance function on $\Omega \times \Omega$, such that $d(\pi_i, \pi_j) \geq 0$ for every $\pi_i, \pi_j \in \Omega$, and $d(\pi_i, \pi_j) = 0$ iff $\pi_i = \pi_j$.

A. Mallows Model

In the vanilla Mallows model [6], the rankings are generated from a probability density function given by

$$p_\theta(\pi) = \frac{\exp(-\theta d(\pi, \pi_0))}{\psi(\theta)}, \quad \pi \in \Omega, \quad \theta > 0, \tag{1}$$

where π_0 represents the consensus ranking and θ is an inverse scale dispersion parameter such that when $\theta \to \infty$, $p_\theta(\pi)$ concentrated at the consensus ranking π_0 and when $\theta \to 0$, $p_\theta(\pi)$ is the uniform distribution over all permutations [5]. For the Kendall-Tau distance, every permutation π is uniquely determined from $n-1$ integers (sufficient statistics), given by

$$V_j(\pi) = \sum_{l>j} \mathbf{1}\{l \prec_\pi j\} \text{ and } d(\pi, \pi_0) = \sum_{j=1}^{n-1} V_j(\pi\pi_0^{-1}), \tag{2}$$

where $i \prec_\pi j$ means that i is ranked before j in π, $\mathbf{1}\{\cdot\}$ is the indicator function, and $V_j(\pi\pi_0^{-1}) \in \{0, \ldots, n-j\}$. In [5], it has been shown that that the model in (1) factors into a product of independent univariate exponential models, one for each $V_j(\pi)$ where $j = 1, \ldots, n-1$. A parametrized generalization of the Mallows model is given by

$$\Pr(V_j(\pi\pi_0^{-1}) = v_j) = \frac{\exp(-\theta_j v_j)}{\psi_j(\theta_j)}, \quad v_j = 0, 1, \ldots, n-j, \tag{3}$$

where $\psi_j(\theta_j) = (1 - \exp -((n-j+1)\theta_j))(1-\exp(-\theta_j))^{-1}$ is the normalization constant.[1] The joint Mallows distribution [5] is a product of independent univariate exponential models, one for each $V_j(\pi\pi_0^{-1})$ given as

$$\begin{aligned}\Pr(V_1(\pi\pi_0^{-1}) = v_1, \ldots, V_{n-1}(\pi\pi_0^{-1}) = v_{n-1}) \\ = \prod_{j=1}^{n-1} \frac{\exp(-\theta_j v_j)}{\psi_j(\theta_j)}.\end{aligned} \tag{4}$$

III. Permutation-based Multiplicative Mixture Mallows Model (M4)

We present the M4 model for learning the ground-truth permutations and the associated dispersion parameters of the Mallows model. Consider a heterogeneous population consisting of M clusters, where the m-th cluster is characterized by a ground-truth permutation π_{m0} and parameters $\boldsymbol{\theta}_m \in \mathbb{R}_+^{n-1}$. The Kendall-Tau distance [7] between a given sample π_k and the m-th consensus ranking π_{m0} is given by

$$V_{mj}(\tau_{k,m}) = \sum_{l>j} \mathbf{1}\big\{l \prec_{\tau_{k,m}} j\big\}, \text{ where } \tau_{k,m} = \pi_k \pi_{m0}^{-1}. \tag{5}$$

Given K independent and identically distributed (i.i.d.) sample permutations $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_K]$, M component distributions and $\mathbf{v}_m(k) = \{v_{mj}(k)\}_{j=1}^{n-1}$, a weighted additive mixture distribution is given by [13]

$$p(\mathbf{v}_m(k)|\boldsymbol{\theta}) = \sum_{m=1}^{M} w_m p_m(\mathbf{v}_m(k)|\boldsymbol{\theta}_m), \tag{6}$$

where w_m represents the mixing weight of the m-th component distribution given by $p_m(\mathbf{v}_m(k)|\boldsymbol{\theta}_m)$. The Kendall-Tau distance for the k-th sample, $\mathbf{v}_m(k)$ is an $n-1$ length vector whose j-th entry is given by $v_{mj}(k) = V_{mj}(\pi_k \pi_{m0}^{-1})$ (cf. (5)). Rewriting (6) in terms of the per-sample latent boolean weight vector $\mathbf{z}_k = [z_k(1), \ldots, z_k(M)]^T$ such that $\sum_{m=1}^M z_k(m) = 1$, and $p(z_k(1) = 0, \ldots, z_k(m) = 1, \ldots, z_k(M) = 0) = w_m$, we obtain [14]

$$p(\mathbf{v}_m(k)|\boldsymbol{\theta}) = \sum_{m=1}^{M} w_m \prod_{m=1}^{M} p_m(\mathbf{v}_m(k)|\boldsymbol{\theta}_m)^{z_k(m)}, \tag{7}$$

where $\boldsymbol{\theta} \in \mathbb{R}^{M \times (n-1)}$ consists of $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M$ as its columns. Setting $z_k(m) = 1$ implies that the k-th sample is a member of the m-th subpopulation [14]. Hence the distribution of the k-th sample $\mathbf{v}_m(k)$ conditioned on $\boldsymbol{\theta}$ and $\mathbf{z}_k$ is

$$p(\mathbf{v}_m(k)|\boldsymbol{\theta}, \mathbf{z}_k) = \frac{1}{c(\mathbf{z}_k, \boldsymbol{\theta})} \prod_{m=1}^{M} p_m(\mathbf{v}_m(k)|\boldsymbol{\theta}_m)^{z_k(m)}, \tag{8}$$

where $c(\mathbf{z}_k, \boldsymbol{\theta})$ is the partition function. The model in (8) resembles the product of experts model proposed in [10]. In fact, the ranking model for the homogeneous population in [7] is a special case of the M4 model when we set $M = 1$ and $z_k(1) = 1$ for all k.

We relax the constraint on the boolean nature of vector $\mathbf{z}_k$, and assume that its m-th entry $z_k(m) \in [0, 1]$. This model is partly inspired by the *tilting* of probability distributions, which is an ubiquitous concept in hypothesis testing [11, Ch. 11]. Instead of assigning a given sample to one of the M subpopulations, we assign it to one of the $\binom{M}{2}$ size-2 subpopulations, where each of the M hypotheses are characterized by distinct Mallows model. Accordingly, the weight vector $\mathbf{z}_k$ is 2-sparse (i.e., $\|\mathbf{z}_k\|_0 = 2$).

We assume that the distribution of permutations in each subpopulation is a Mallows distribution defined by

$$p_m(\mathbf{v}_m(k)|\boldsymbol{\theta}_m) = \prod_{j=1}^{n-1} \frac{\exp\{-\theta_{mj} v_{mj}(k)\}}{\psi_j(\theta_j)}. \tag{9}$$

Hence, the M4 can be written as

$$p(\mathbf{v}(k)|\boldsymbol{\theta}, \mathbf{z}_k) = \frac{\exp\big\{-\sum_{m=1}^M z_k(m) \sum_{j=1}^{n-1} \theta_{mj} v_{mj}(k)\big\}}{c(\mathbf{z}_k, \boldsymbol{\theta})}, \tag{10}$$

where $\mathbf{v}(k) = [\mathbf{v}_1(k), \ldots, \mathbf{v}_M(k)]$ and $\mathbf{z}_k = [z_k(m), \ldots, z_k(m)]$ and $\mathbf{z} = [\mathbf{z}_1, \ldots, \mathbf{z}_K]$. The partition function can be written as $c(\mathbf{z}_k, \boldsymbol{\theta}) = \prod_{j=1}^{n-1} c_j(\mathbf{z}_k, \boldsymbol{\theta}_j)$ where $c_j(\mathbf{z}_k, \boldsymbol{\theta}_j) = \sum_{v_{mj}(k)} \prod_{m=1}^M p_j(v_{mj}(k))|\theta_{mj})^{z_k(m)}$. Thus, the overall mixture distribution of the k-th sample is

$$p(\mathbf{v}(k)|\boldsymbol{\theta}, \mathbf{z}_k) = \frac{\exp\big\{-\sum_{m=1}^M z_k(m) \sum_{j=1}^{n-1} \theta_{mj} v_{mj}(k)\big\}}{\prod_{j=1}^{n-1} \psi_j\left(\sum_{m=1}^M z_k(m)\theta_{mj}\right)}. \tag{11}$$

IV. Majorization-Minimization (M-M) Algorithm

The M-M framework [15] solves difficult optimization problems by iteratively minimizing a majorizing function until a local optimum is obtained. We present an M-M algorithm for parameter estimation in the M4 model. Given K i.i.d. sample permutations, the goal is to estimate $(\mathbf{z}, \boldsymbol{\theta})$, assuming that the ground-truth permutations π_{m0} are known. We address the estimation of π_{m0} in Sec. IV-B. The log-likelihood is

$$\ell(\mathbf{z}, \boldsymbol{\theta}) = \sum_{k=1}^{K} \log p(\mathbf{v}(k)|\boldsymbol{\theta}, \mathbf{z}_k) \tag{12}$$

[1] We write $\Pr(V_j(\pi\pi_0^{-1}) = v_j)$ as $p(v_j)$ in the sequel.

Let $\alpha_{kj}(m) = z_k(m)\theta_{mj}$. The log-likelihood given above can be expressed as $\ell(\mathbf{z}, \boldsymbol{\theta}) = -\sum_{k=1}^{K}\sum_{j=1}^{n-1}\ell(\boldsymbol{\alpha}_{kj})$, where

$$\ell(\boldsymbol{\alpha}_{kj}) = \boldsymbol{\alpha}_{kj}^T\mathbf{v}_j(k) - \log\left(1 - \exp\Big[-\sum_{m=1}^{M}\alpha_{kj}(m)\Big]\right) + \log\left(1 - \exp\Big[-(n-j+1)\sum_{m=1}^{M}\alpha_{kj}(m)\Big]\right). \quad (13)$$

Proposition 1: The function $\ell(\boldsymbol{\alpha}_{kj})$ is a difference of convex (d.c.) functions.

A. Parameter Estimation in the M4

We now employ the CCCP which is a popular technique to obtain solutions to the difference of convex functions given in (13) [16]. CCCP converts such a function to a sequence of convex functions and iteratively solves the original optimization problem by obtaining the optima to the intermediate convex functions. In the p-th iteration, we construct a convex majorizing function $\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)})$, such that the $(p+1)$-st iterate of $\boldsymbol{\alpha}_{kj}$ is given by

$$\boldsymbol{\alpha}_{kj}^{(p+1)} = \underset{\boldsymbol{\alpha}_{kj}\in\mathbb{R}_+^M}{\arg\max}\,\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)}), \quad (14)$$

and $\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)}) \triangleq f(\boldsymbol{\alpha}_{kj}) - \boldsymbol{\alpha}_{kj}^T\nabla_{\boldsymbol{\alpha}_{kj}}g(\boldsymbol{\alpha}_{kj}^{(p)})$, $\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)}) \geq \ell(\boldsymbol{\alpha}_{kj})$ and $\mathcal{Q}(\boldsymbol{\alpha}_{kj}^{(p)}; \boldsymbol{\alpha}_{kj}^{(p)}) = \ell(\boldsymbol{\alpha}_{kj}^{(p)})$. We compute the majorizing function $\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)})$ as

$$\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)}) = \boldsymbol{\alpha}_{kj}^T\mathbf{v}_m(k) - c_{kj}^{(p)}\sum_{m=1}^{M}\alpha_{kj}(m) - \log\left(1 - \exp\Big[-\sum_{m=1}^{M}\alpha_{kj}(m)\Big]\right), \quad (15)$$

where $c_{kj}^{(p)}$ is given by

$$c_{kj}^{(p)} = -\frac{(n-j+1)\exp[-(n-j+1)\sum_{m=1}^{M}\alpha_{kj}(m)^{(p)}]}{1 - \exp(-(n-j+1)\sum_{m=1}^{M}\alpha_{kj}(m)^{(p)}]}. \quad (16)$$

Hence, the CCCP procedure leads to an affine approximation $g(\boldsymbol{\alpha}_{kj})$ about $\boldsymbol{\alpha}_{kj}^{(p)}$, and linearly combines it with the convex function $f(\boldsymbol{\alpha}_{kj})$ resulting in a convex majorizing function $\boldsymbol{\alpha}_{kj} \mapsto \mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)})$. To estimate $\mathbf{z}_k$ and $\boldsymbol{\theta}_j$ from $\boldsymbol{\alpha}_{kj}$ for $k \in [K]$ and $j \in [n-1]$, we use the *biconvexity* property [17] of $\mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)})$.

Proposition 2: The function $(\mathbf{z}_k, \boldsymbol{\theta}_j) \mapsto \mathcal{Q}(\boldsymbol{\alpha}_{kj}; \boldsymbol{\alpha}_{kj}^{(p)})$ is biconvex in $\mathbf{z}_k$ and $\boldsymbol{\theta}_j$. The feasible set $\Theta \times Z$ is also biconvex.

In a non-linear biconvex optimization setting, the alternate convex search (ACS) algorithm [17] is usually employed. Here, the set of unknown variables are divided into disjoint sub-blocks. In every iteration, only one sub-block is optimized while the other sub-block is held fixed [17]. Since fixing one of the sub-blocks results in a convex sub-problem, efficient algorithms can be used to solve biconvex programs.

In the context of M4, the convex sub-problem for optimizing $\mathbf{z}_k$ with $\boldsymbol{\theta}_j = \boldsymbol{\theta}_j^{(p)}$ is given by

$$\mathbf{z}_k^{(p+1)} = \underset{\mathbf{z}_k\in\mathbb{R}_+^M}{\arg\min}\sum_{j=1}^{n-1}\mathcal{Q}_j^{\boldsymbol{\theta}}(\mathbf{z}_k), \quad (17)$$

where $\mathcal{Q}_j^{\boldsymbol{\theta}}(\mathbf{z}_k) = \mathbf{z}_k^T\text{diag}(\boldsymbol{\theta}_j^{(p)})\mathbf{v}_j(k) - \log\big(1 - \exp(-\mathbf{z}_k^T\boldsymbol{\theta}_j^{(p)})\big) - c_{kj}^{(p)}\mathbf{z}_k^T\boldsymbol{\theta}_j^{(p)}$. Note that the optimization problem in (17) cannot be solved in closed form. However, the finite-sum property allows us to employ the *stochastic gradient descent* (SGD) technique to obtain $\mathbf{z}_k^{(p+1)}$ [18]. Using the SGD approach, the j-th (where $j \in [n-1]$) update for $\mathbf{z}_k$ is

$$\mathbf{z}_k^{(p+\frac{j+1}{n-1})} \triangleq \mathbf{z}_k^{(p+\frac{j}{n-1})} - \eta_{z_k}\nabla_{\mathbf{z}_k}\mathcal{Q}_j^{\boldsymbol{\theta}}\Big(\mathbf{z}_k^{(p+\frac{j}{n-1})}\Big), \quad (18)$$

where η_{z_k} is a learning rate parameter. Here, $\nabla_{\mathbf{z}_k}\mathcal{Q}_j^{\boldsymbol{\theta}}\Big(\mathbf{z}_k^{(p+\frac{j}{n-1})}\Big)$ is given by

$$\nabla_{\mathbf{z}_k}\mathcal{Q}_j^{\boldsymbol{\theta}}\Big(\mathbf{z}_k\Big) = \text{diag}(\boldsymbol{\theta}_j^{(p)})\mathbf{v}_j(k) - \frac{\boldsymbol{\theta}_j^{(p)}\exp(-\mathbf{z}_k^T\boldsymbol{\theta}^{(p)})}{1 - \exp(-\mathbf{z}_k^T\boldsymbol{\theta}^{(p)})} \quad (19)$$

Similarly, for the convex sub-problem in $\boldsymbol{\theta}_j$ with $\mathbf{z}_k = \mathbf{z}_k^{(p)}$ we follow the steps as outlined above and obtain the k-th update for $\boldsymbol{\theta}_j$ as

$$\boldsymbol{\theta}_j^{(p+\frac{k+1}{K})} \triangleq \boldsymbol{\theta}_j^{(p+\frac{k}{K})} - \eta_{\boldsymbol{\theta}_j}\nabla_{\boldsymbol{\theta}_j}\mathcal{Q}_k^{\mathbf{z}}\Big(\boldsymbol{\theta}_j^{(p+\frac{k}{K})}\Big), \quad (20)$$

where $\eta_{\boldsymbol{\theta}_j}$ is the learning rate and $k \in [K]$. Here $\nabla_{\boldsymbol{\theta}_j}\mathcal{Q}_k^{\mathbf{z}}\Big(\boldsymbol{\theta}_j^{(p+\frac{k}{K})}\Big)$ can be computed as

$$\nabla_{\boldsymbol{\theta}_j}\mathcal{Q}_k^{\mathbf{z}}\Big(\boldsymbol{\theta}_j\Big) = \text{diag}(\mathbf{z}_k^{(p)})\mathbf{v}_j(k) - \frac{\mathbf{z}_k^{(p)}\exp(-\boldsymbol{\theta}_j^T\mathbf{z}_k^{(p)})}{1 - \exp(-\boldsymbol{\theta}_j^T\mathbf{z}_k^{(p)})} \quad (21)$$

It is known that SGD converges to a local minimizer of the original objective [19].

B. Estimation of Consensus Rankings

In this section, we consider the problem of estimating the consensus rankings π_{m0} for $m \in [M]$. This is a well-known combinatorial optimization problem and several heuristics have been proposed to solve it approximately [5]. In the context of M4, the optimization problem is

$$(\hat{\pi}_{10}, \ldots, \hat{\pi}_{M0}) = \underset{[\pi_{10},\ldots,\pi_{M0}]\in\Omega^M}{\arg\min}\sum_{k=1}^{K}\sum_{m=1}^{M}z_k(m)\boldsymbol{\theta}_m^T\mathbf{v}_m(k), \quad (22)$$

where $\mathbf{v}_m(k)$ is a function of π_{m0} for all m, as given in (3). We propose a greedy approach for joint estimation of the consensus ranking, $\mathbf{z}$ and $\boldsymbol{\theta}$, along the lines of the algorithms proposed in [7]. Consider the M matrices $\mathbf{Q}^{(1)}, \ldots, \mathbf{Q}^{(M)}$, where the m-th matrix is obtained from a homogeneous population with consensus ranking π_{m0} as follows:

$$Q_{jl}^{(m)}(\boldsymbol{\pi}_{K_m}) = \frac{1}{K_m}\sum_{k=1}^{K_m}\mathbf{1}\{j \prec_{\pi_k} l\}. \quad (23)$$

That is, $Q_{jl}^{(m)}(\boldsymbol{\pi}_{K_m})$ is the empirical probability that j precedes l in the sample $\boldsymbol{\pi}_{K_m}$, where the sample consists of K_m permutations from the m-th component distribution. The mean of v_{m1} under the sampling distribution is given by

$$\bar{v}_{m1} = \sum_{j:j\neq r}Q_{jr}^{(m)}, \quad \text{whenever} \quad \pi_{m0}^{-1}(1) = r, \quad (24)$$

and hence, $\pi_{m0}^{-1}(1) = \arg\min_r\sum_{j\neq r}Q_{jr}^{(m)}$.

A tree-based search algorithm is derived by extending the above idea to all j. The $n!$ nodes of the tree represent partial orderings of π_{m0} given by $\rho_{mj} = (r_{m1}, \ldots, r_{mj})$,

i.e., each node has $n-j$ children. Hence, a particular level j of the search tree corresponds to the j-th position in the sample permutation, where $j \in [n-1]$. Further, any path of length n through the tree starting from the root represents a permutation. Given a consensus ranking π_{m0}, and associated parameters of the model $\boldsymbol{\theta}_m = [\theta_{m1}, \ldots, \theta_{m(n-1)}]$, the cost at node ρ_{mj} is given by

$$C_m(r_{m1}, \ldots, r_{mj}) = \sum_{l=1}^{j} z(m)\theta_{mj} V_{ml}(r_{m1}, \ldots, r_{ml}), \quad (25)$$

where $V_{ml}(r_{m1}, \ldots, r_{ml}) = \sum_{l \notin \{r_{m1}, \ldots, r_{mj}\}} Q^{(m)}_{l r_{mj}}$. The proposed algorithm chooses the permutation that leads to smallest cost at each level j. Under the sampling distribution, the mean of $\sum_{k=1}^{K}\sum_{m=1}^{M} z_k(m)\boldsymbol{\theta}_m^T \mathbf{v}_m(k)$ is given by $\sum_{j \neq r} z(m_1) Q^{(m_1)}_{jr} + z(m_2) Q^{(m_2)}_{jr}$, where $z(m_1)$ and $z(m_2)$ are the weights of the samples from the multiplicative mixture distribution consisting of m_1 and m_2 component distributions. We employ the mean as a *proxy* for the term $\sum_{k=1}^{K}\sum_{m=1}^{M} z_k(m)\boldsymbol{\theta}_m^T \mathbf{v}_m(k)$ in (18) and (20). Further, v_{mj} in (18) and (20) can be replaced by $Q^{(m_1)}_{jr}$, where the index r is obtained by minimizing the cost at the j-th level.

The steps of the proposed M4-SEARCHPI algorithm are given in Algorithm 2. Note that the per-sample weights $\mathbf{z}_k$ are updated for j SGD iterations at the j-th level. The computation of A is described in [7].

Algorithm 1 M4-SEARCHPI

Require: Obtain $(\pi_{10}, \ldots, \pi_{M0}, \boldsymbol{\theta}, \mathbf{z})$ from $(\mathbf{Q}^{(1)}, \ldots, \mathbf{Q}^{(M)})$.
Input: $\mathbf{Q}^{(1)}, \ldots, \mathbf{Q}^{(M)}$, $p_{\max}$.
Output: $\pi_{10}, \ldots, \pi_{M0}$, $\boldsymbol{\theta}$ and $\mathbf{z}$.
Initialization $\theta^{(0)}_{mj}(m)$ for all m, j, $z^{(0)}_k(m) = 1/M$ for all m, k.
Compute $\bar{V} = \sum_{j \neq r}\sum_{m=1}^{M} z(m) Q^{(m)}_{jr}$.
while $|\rho_{mj}| < n \ \forall\ m$ **do**
  for $m = 1 : M$ **do**
    for $r_{m(j+1)} \in [n] \backslash \rho_{mj}$ **do**
      Create node $\rho' = [\rho_{mj}, r_{m(j+1)}]$, $V_{m(j+1)}(\rho') = \sum_{l \in [n]\backslash \rho'} Q^{(m)}_{l r_{j+1}}$.
    end for
  end for
  while $p \leq p_{\max}$ **do**
    $j = |\rho'|$, $p = p + 1$.
    Solve (18) using $\bar{V}$, with a fixed $\boldsymbol{\theta}^{(p)}_j$ and (20) using $\bar{V}$, with a fixed $\mathbf{z}^{(p)}_k$.
  end while
  for $m = 1 : M$ **do**
    Compute $C_m(\rho')$ and $L_m(\rho') = C_m(\rho') + A$ and set $\rho_{m(j+1)} = \arg\min_{\rho \in S_m} L_m(\rho)$.
  end for
end while

V. Numerical Experiments

In this section, we numerically demonstrate the goodness-of-fit of the M4 using the popular American Psychological Association (APA) [20] and Sushi datasets [21]. We also demonstrate the permutation learning capability of the inference algorithms proposed in Secs. IV on synthetic datasets.

A. Goodness-of-fit: APA and Sushi Dataset

In this section, we use the M4 to model the real-world APA presidential election dataset [20] and the Sushi dataset [21]. Our goal is to highlight the advantages of M4 as compared to the homogeneous Mallows model [22] and the additive mixture Mallows model [8], [9]. For both datasets, we use the BIC and ICL [12] as measures of the goodness-of-fit. The BIC is defined as

$$\mathrm{BIC}(G) = 2\ell_G(\mathbf{z}, \boldsymbol{\theta}) - v_G \log(K), \quad (26)$$

where $\ell_G(\mathbf{z}, \boldsymbol{\theta})$ represents the log likelihood under the model G and v_G represents the number of free parameters in G. The ICL is used to measure the separation of the mixture components and is a popular criterion for clustering applications. The ICL is defined as

$$\mathrm{ICL}(G) = \mathrm{BIC}(G) - 2\sum_{k=1}^{K} \mathrm{Entropy}(\mathbf{z}_k), \quad (27)$$

where $\mathrm{Entropy}(\mathbf{z}_k) = -\sum_{m=1}^{M} z_k(m) \log(z_k(m))$.

M4	$M=1$	$M=2$	$M=3$
BIC	-5.48	-1.17	-1.21
ICL	-	-0.77	-0.823
Additive	$M=1$	$M=2$	$M=3$
BIC	-5.48	-2.10	-2.75
ICL	-	-1.71	-2.12

Table I
APA DATASET: BIC AND ICL (VALUES TO BE MULTIPLIED BY 10^4)

1) APA Dataset: The 1980, the APA presidential election consisted of five candidates (A, B, C, D, E) and voters were asked to rank the candidates in their order of preference. Among the 15449 votes that were cast, 5738 voters ranked all five candidates [20]. We demonstrate the goodness-of-fit of the M4 for permutation-based observations using the 5738 permutations. We simulate the additive mixture Mallows model using the EM algorithm for comparison. Note that the number of free parameters in the M4 is given by $M(n-1) + 2\binom{M}{2}$. Using the BIC and ICL we declare the model with the largest values as the best model.

Using the M-M based inference algorithm, we fit the M4 to this dataset, and computed the per-sample weights and the parameters, for $M = 2$ and $M = 3$. We also ran the EM algorithm to learn the additive mixture Mallows model [8], [9] and the homogeneous Mallows model [7] (with $M = 1$). It can be seen from Tables I that for both $M = 2$ and $M = 3$, M4 provides a better fit compared to the additive mixture model in both the complete and partial rankings scenarios. When $M = 3$, we obtain a lower value of BIC and ICL indicating that a 3 component model is better suited for this dataset.

2) Sushi Dataset: We now fit the M4 to the popular Sushi dataset. This dataset compares 10 types of Sushi. The data was collected by surveying 5000 individuals living in Japan about their preferences regarding the Sushi variants [21].

From the Condorcet ranking [23] we know that Fatty tuna is a common favorite and is ranked highest, while cucumber roll is the least liked. However, there is a divided opinion about sea urchin as 15-20% of the voters rank it as their most favourite or their least favourite item, hinting that this dataset is suited for mixture modelling.

The global search for the candidate permutation that leads to the smallest value of BIC is infeasible as the number of possible permutation choices is too large. We pick a

	$M=1$	$M=2$	$M=3$
M4 (BIC)	-1.487	-0.274	-0.226
M4 (ICL)	-1.487	-0.205	-0.184

Table II
SUSHI DATASET: BIC AND ICL (VALUES TO BE MULTIPLIED BY 10^4)

small subset of permutations that consists of the Condorcet permutation and permutations that capture the voters' divided opinion on Sea Urchin. We obtain the BIC values for the subset of permutations as given in Table II. In the case of the Sushi dataset, the EM algorithm for additive mixture density necessitates a weighted sum over 10! permutations, which is infeasible [9]. However, we are able to model the heterogeneous population using M4, hence substantiating our claim that the inference framework for the M4 is indeed computationally simple.

B. Consensus Ranking Estimation Performance

In this section, we demonstrate consensus ranking recovery performance of the proposed inference algorithms. We generated a synthetic dataset with $n = 8$ using Gibbs sampling. The consensus ranking was chosen randomly, ensuring that the distance between permutation is sufficiently large [24]. The experiments are repeated over 100 trials. In all our experiments, we set $\eta_\theta \propto i^{-\beta}$, where i is the iteration number of the SGD algorithm and $\beta = 2$. We display the success rates of the proposed algorithm, where success rate refers to the average number of times the algorithm recovers the consensus ranking perfectly ($\hat{\pi}_{m0} = \pi_{m0}$) for $M = 1, 2, 4$. We observe that as K increases, the success rate also increases. Finally, as M increases, the success rate decreases as there are more parameters to estimate.

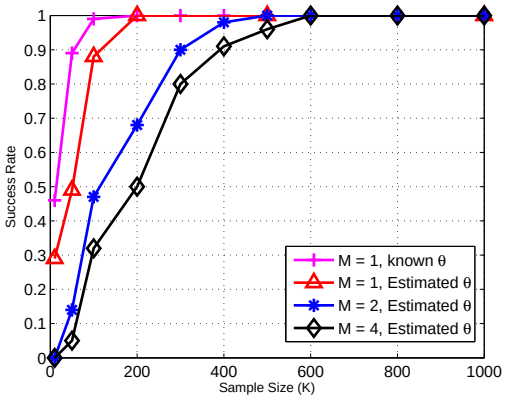

Figure 1. Plot of the success rates of the M4-SEARCHPI algorithm.

VI. FUTURE WORK

There are a couple of natural extensions of the present work.

1) First, it is known that tensor decomposition methods [24] have had tremendous success in disambiguating mixtures and latent variable models. Adapting such tensor methods to the M4 is a fruitful research direction.
2) A promising area of research consists in establishing the fundamental tradeoff between the number of samples and the probability of error in learning the consensus rankings.

REFERENCES

[1] S. Negahban, S. Oh, and D. Shah, "Iterative ranking from pair-wise comparisons," in *Advances in Neural Information Processing Systems*, 2012, pp. 2474–2482.
[2] S. Oh and D. Shah, "Learning mixed multinomial logit model from ordinal data," in *Advances in Neural Information Processing Systems*, 2014, pp. 595–603.
[3] C. Suh, V. Tan, and R. Zhao, "Adversarial top-K ranking," *IEEE Trans. on Information Theory*, vol. 63, no. 4, pp. 2201–2225, 2017.
[4] R. L. Plackett, "The analysis of permutations," *Applied Statistics*, pp. 193–202, 1975.
[5] M. A. Fligner and J. S. Verducci, "Distance based ranking models," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 359–369, 1986.
[6] C. L. Mallows, "Non-null ranking models: I," *Biometrika*, vol. 44, no. 1/2, pp. 114–130, 1957.
[7] M. Meila, K. Phadnis, A. Patterson, and J. Bilmes, "Consensus ranking under the exponential model," in *Proc. of the 23rd Annual Conference on Uncertainty in Artificial Intelligence*, 2007.
[8] T. B. Murphy and D. Martin, "Mixtures of distance-based models for ranking data," *Computational Statistics & Data Analysis*, vol. 41, no. 3, pp. 645–655, 2003.
[9] P. H. Lee and L. Philip, "Mixtures of weighted distance-based models for ranking data with applications in political studies," *Computational Statistics & Data Analysis*, vol. 56, no. 8, pp. 2486–2500, 2012.
[10] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural computation*, vol. 14, no. 8, pp. 1771–1800, 2002.
[11] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.
[12] C. Biernacki, G. Celeux, and G. Govaert, "Assessing a mixture model for clustering with the integrated completed likelihood," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 7, pp. 719–725, 2000.
[13] L. M. Busse, P. Orbanz, and J. M. Buhmann, "Cluster analysis of heterogeneous rank data," in *Proc. of International Conference on Machine Learning*. ACM, 2007, pp. 113–120.
[14] K. A. Heller and Z. Ghahramani, "A nonparametric Bayesian approach to modeling overlapping clusters." in *AISTATS*, 2007, pp. 187–194.
[15] D. R. Hunter and K. Lange, "Quantile regression via an MM algorithm," *Journal of Computational and Graphical Statistics*, vol. 9, pp. 60–77, 2000.
[16] G. R. Lanckriet and B. K. Sriperumbudur, "On the convergence of the concave-convex procedure," in *Advances in Neural Information Processing Systems*, 2009, pp. 1759–1767.
[17] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373–407, 2007.
[18] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, pp. 400–407, 1951.
[19] O. Shamir and T. Zhang, "Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes." in *Proc. of the International Conference on Machine Learning*, 2013, pp. 71–79.
[20] P. Diaconis, "A generalization of spectral analysis with application to ranked data," *The Annals of Statistics*, pp. 949–979, 1989.
[21] T. Kamishima, "Nantonac collaborative filtering: recommendation based on order responses," in *Proc. of the International Conference on Knowledge Discovery and Data Mining*. ACM, 2003, pp. 583–588.
[22] E. Irurozki, B. Calvo, and J. A. Lozano, "Permallows: An R package for Mallows and generalized Mallows models," *Journal of Statistical Software*, vol. 71, 2016.
[23] W. Chen, "How to order sushi," Ph.D. dissertation, Harvard University, 2014.
[24] P. Awasthi, A. Blum, O. Sheffet, and A. Vijayaraghavan, "Learning mixtures of ranking models," in *Advances in Neural Information Processing Systems*, 2014, pp. 2609–2617.