Rank Aggregation via Heterogeneous Thurstone Preference Models

Tao Jin,[1*] **Pan Xu,**[2*] **Quanquan Gu,**[2†] **Farzad Farnoud**[1†]
[1] University of Virginia, [2] University of California, Los Angeles
taoj@virginia.edu, panxu@cs.ucla.edu, qgu@cs.ucla.edu, farzad@virginia.edu

Abstract

We propose the Heterogeneous Thurstone Model (HTM) for aggregating ranked data, which can take the accuracy levels of different users into account. By allowing different noise distributions, the proposed HTM model maintains the generality of Thurstone's original framework, and as such, also extends the Bradley-Terry-Luce (BTL) model for pairwise comparisons to heterogeneous populations of users. Under this framework, we also propose a rank aggregation algorithm based on alternating gradient descent to estimate the underlying item scores and accuracy levels of different users simultaneously from noisy pairwise comparisons. We theoretically prove that the proposed algorithm converges linearly up to a statistical error which matches that of the state-of-the-art method for the single-user BTL model. We evaluate the proposed HTM model and algorithm on both synthetic and real data, demonstrating that it outperforms existing methods.

1 Introduction

Rank aggregation refers to the task of recovering the order of a set of objects given pairwise comparisons, partial rankings, or full rankings obtained from a set of users or experts. Compared to rating items, comparison is a more natural task for humans which can provide more consistent results, in part because it does not rely on arbitrary scales. Furthermore, ranked data can be obtained not only by explicitly querying users, but also through passive data collection, i.e., by observing user behavior, for example product purchases, clicks on search engine results, choice of movies in streaming services, etc. As a result, rank aggregation has a wide range of applications, from classical social choice applications (de Borda 1781) to information retrieval (Dwork et al. 2001), recommendation systems (Baltrunas, Makcinskas, and Ricci 2010), and bioinformatics (Aerts et al. 2006; Kim, Farnoud, and Milenkovic 2015).

In aggregating rankings, the raw data is often noisy and inconsistent. One approach to arrive at a single ranking is to assume a generative model for the data whose parameters include a true score for each of the items. In particular, Thurstone's preference model (Thurstone 1927) assumes that comparisons or partial rankings result from comparing versions of the true scores corrupted by additive noise. Special cases of Thurstone's model include the popular Bradley-Terry-Luce (BTL) model for pairwise comparisons and the Placket-Luce (PL) model for partial rankings. In these settings, estimating the true scores from data will allow us to identify the true ranking of the items. Various estimation and aggregation algorithms have been developed for Thurstone's preference model and its special cases, including (Hunter 2004; Guiver and Snelson 2009; Hajek, Oh, and Xu 2014; Chen and Suh 2015; Vojnovic and Yun 2016; Negahban, Oh, and Shah 2017).

Conventional models of ranked data and aggregation algorithms that rely on them make the assumption that the data is either produced by a single user[1] or from a set of users that are similar. In real-world datasets, however, users that provide the raw data are usually diverse with different levels of familiarity with the objects of interest, thus providing data that is not uniformly reliable and should not have equal influence on the final result. This is of particular importance in applications such as aggregating expert opinions for decision-making and aggregating annotations provided by workers in crowd sourcing settings.

In this paper, we study the problem of rank aggregation for heterogeneous populations of users. We present a generalization of Thurstone's model, called the *heterogeneous Thurstone model* (HTM), which allows users with different noise levels, as well as a certain class of adversarial users. Unlike previous efforts on rank aggregation for heterogeneous populations such as (Chen et al. 2013; Kumar and Lease 2011), the proposed model maintains the generality of Thurstone's framework and thus also extends its special cases such as BTL and PL models. We evaluate the performance of the method using simulated data for different noise distributions. We also demonstrate that the proposed aggregation algorithm outperforms the state-of-the-art method for real datasets on evaluating the difficulty of English text

[*] Equal contribution.
[†] Co-corresponding authors.

[1] We use the term user to refer to any entity that provides ranked data. In specific applications other terms may be more appropriate, such as voter, expert, judge, worker, and annotator.

and comparing the population of a set of countries.

Our Contributions: Our main contributions are summarized as follows

- We propose a general model called the heterogeneous Thurstone model (HTM) for producing ranked data based on heterogeneous sources, which reduces to the heterogeneous BTL (HBTL) model when the noise follows the Gumbel distribution and to the heterogeneous Thurstone Case V (HTCV) model when the noise follows the normal distribution respectively.
- We develop an efficient algorithm for aggregating pairwise comparisons and estimating user accuracy levels for a wide class of noise distributions based on minimizing the negative log-likelihood loss via alternating gradient descent.
- We theoretically show that the proposed algorithm converges to the unknown score vector and the accuracy vector at a locally linear rate up to a tight statistical error under mild conditions.
- For models with specific noise distributions such as the HBTL and HTCV, we prove that the proposed algorithm converges linearly to the unknown score vector and accuracy vector up to statistical errors in the order of $O(n^2 \log(mn^2)/(mk))$, where k is sample size, n is the number of items and m is the number of users. When $m = 1$, the statistical error matches the error bound in the state-of-the-art work for single user BTL model (Negahban, Oh, and Shah 2017).
- We conduct thorough experiments on both synthetic and real world data to validate our theoretical results and demonstrate the superiority of our proposed model and algorithm.

The reminder of this paper is organized as follows. In Section 2, we review the most related work in the literature. In Section 3, we propose a family of heterogeneous Thurstone models. In Section 4, we propose an efficient algorithm for learning the ranking from pairwise comparisons. We theoretically analyze the convergence of the proposed algorithm in Section 5. Thorough experimental results are presented in Section 6 and Section 7 concludes the paper.

2 Additional Related Work

The problem of rank aggregation has a long history, dating back to the works of (de Borda 1781) and (de Condorcet 1785) in the 18th century, where the problems of social choice and voting were discussed. More recently, the problem of aggregating pairwise comparisons, where comparisons are incorrect with a given probability p, was studied by (Braverman and Mossel 2008) and (Wauthier, Jordan, and Jojic 2013). Instead of assuming the same probability for all comparisons to be incorrect, it is natural to assume that the comparison of similar items is more likely to be noisy than those items that are distinctly different. This intuition is reflected in the random utility model (RUM), also known as *Thurstone's model* (Thurstone 1927), where each item has a true score, and users provide rankings of subsets of items by comparing approximate version of these scores corrupted by additive noise.

When restricted to comparing pairs of items, Thurstone's model reduces to the BTL model (Zermelo 1929; Bradley and Terry 1952; Luce 1959; Hunter 2004) if the noise follows the Gumbel distribution, and to the Thurstone Case V (TCV) model (Thurstone 1927) if the noise is normally distributed. Recently, (Negahban, Oh, and Shah 2012) proposed Rank Centrality, an iterative method with a random walk interpretation and showed that it performs as well as the maximum likelihood (ML) solution (Zermelo 1929; Hunter 2004) for BTL models and provided non asymptotic performance guarantees. (Chen and Suh 2015) studied identifying the top-K candidates under the BTL model and its sample complexity.

Thurstone's model can also be used to describe data from comparisons of multiple items. (Hajek, Oh, and Xu 2014) provided an upper bound on the error of the ML estimator and studied its optimality when data consists of partial rankings (as opposed to pairwise comparisons) under the PL model. (Yu 2000) studied order statistics under the normal noise distribution with consideration of item confusion covariance and user perception shift in a Bayesian model. (Weng and Lin 2011) proposed a Bayesian approximation method for game player ranking with results from two-team matches. (Guiver and Snelson 2009) studied the ranking aggregation problem with partial ranking (PL model) in a Bayesian framework. However, due to the nature of Bayesian method, above mentioned work provided few theoretical analysis. (Vojnovic and Yun 2016) studied the parameter estimation problem for Thurstone models where first choices among a set of alternatives are observed. (Raman and Joachims 2014; 2015) proposed the peer grading methods for solving a similar problem as ours, while the generative models to aggregate partial rankings and pairwise comparisons are completely different. Very recently, (Zhao, Villamil, and Xia 2018) proposed the k-RUM model which assumes that the rank distribution has a mixture of k RUM components. They also provided the analyses of identifiability and efficiency of this model.

Almost all aforementioned works assume that all the data is provided by a single user or that all users have the same accuracy. However, this assumption is rarely satisfied in real-world datasets. The accuracy levels of different users are considered in (Kumar and Lease 2011), which assumes that each user is correct with a certain probability and studies the problem via simulation methods such as naive Bayes and majority voting. In their pioneering work, (Chen et al. 2013) studied rank aggregation in a crowd-sourcing environment for pairwise comparisons, modeled via the BTL or TCV model, where noisy BTL comparisons are assumed to be further corrupted. They are flipped with a probability that depends on the identity of the worker. The k-RUM model proposed by (Zhao, Villamil, and Xia 2018) considered a mixture of ranking distributions, without using extra information on who contributed the comparison, it may suffer from common mixture model issues.

3 Modeling Heterogeneous Ranked Data

Before introducing our Heterogeneous Thurstone Model, we start by providing some preliminaries of Thurstone's preference model in further detail. Consider a set of n items. The score vector for the items is denoted by $\boldsymbol{s} = (s_1, \ldots, s_n)^\top$. These items/objects are evaluated by a set of m independent users. Each user may be asked to express their preference concerning a subset of items $\{i_1, \ldots, i_h\} \subseteq [n]$, where $2 \leq h \leq n$. For each item i, the user first estimates an empirical score for it as

$$z_i = s_i + \epsilon_i, \tag{3.1}$$

where ϵ_i is a random noise introduced by this evaluation process. This coarse estimate of score z_i is still implicit and cannot be queried or observed by the ranking algorithm. Instead, the user only produces a ranking of these h items by sorting the scores z_i. We thus have

$$\Pr(\pi_1 \succ \pi_2 \succ \cdots \succ \pi_h) = \Pr(z_{\pi_1} > z_{\pi_2} > \cdots > z_{\pi_h}), \tag{3.2}$$

where $i \succ j$ indicates that i is preferred to j by this user and $\{\pi_1, \ldots, \pi_h\}$ is a permutation of $\{i_1, \ldots, i_h\}$. Each time item i is compared with other items, a new score estimate z_i is produced by the user for are commonly assumed to be i.i.d. (Braverman and Mossel 2008; Negahban, Oh, and Shah 2012; Wauthier, Jordan, and Jojic 2013).

The Heterogeneous Thurstone Model

In real-world applications, users often have different levels of expertise and some may even be adversarial. Therefore, it is natural for us to propose an extension of the Thurstone's model presented above, referred to as the *Heterogeneous Thurstone Model* (HTM), which has the flexibility to reflect the different levels of expertise of different users. Specifically, we assume that each user has a different level of making mistakes in evaluating items, i.e., the evaluation noise of user u is controlled by a scaling factor $\gamma_u > 0$. The proposed model is then represented as follows:

$$z_i^u = s_i + \epsilon_i/\gamma_u. \tag{3.3}$$

Based on the estimated scores of each user for each item, the probability of a certain ranking of h items provided by user u is again given by (3.2). While this extension actually applies to both pairwise comparisons and multi-item orderings, we mainly focus on pairwise comparisons in this paper.

When two items i and j are compared by user u, we denote by Y_{ij}^u the random variable representing the result,

$$Y_{ij}^u = \begin{cases} 1 & \text{if } i \succ j; \\ 0 & \text{if } i \prec j. \end{cases} \tag{3.4}$$

Observation of $Y_{ij}^u = 1$ event is due to random variables $z_i^u > z_j^u$. Let F denote the CDF of $\epsilon_j - \epsilon_i$, where ϵ_i and ϵ_j are two i.i.d. random variables. We have

$$\begin{aligned} \Pr(Y_{ij}^u = 1; s_i, s_j, \gamma_u) &= \Pr(\epsilon_j - \epsilon_i < \gamma_u(s_i - s_j)) \\ &= F(\gamma_u(s_i - s_j)). \end{aligned} \tag{3.5}$$

It is clear that the larger the value of γ_u, the more accurate the user is, since large $\gamma_u > 0$ increases the probability of preferring an item with higher score to one with lower score.

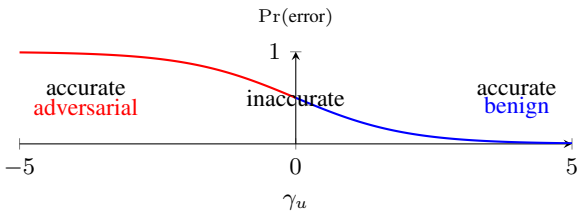


Figure 1: The effect of γ_u on the probability of error for a BTL comparison in which items have scores 0 and 1. In particular, for large negative values of γ_u, the user is accurate (with a high level of expertise) but adversarial.

We now consider several special cases arising from specific noise distributions. First, if ϵ_i follows a Gumbel distribution with mean 0 and scale parameter 1, then we obtain the following *Heterogeneous BTL* (HBTL) model:

$$\begin{aligned} \log \Pr(Y_{ij}^u = 1; s_i, s_j, \gamma_u) &= \log \frac{e^{\gamma_u s_i}}{e^{\gamma_u s_i} + e^{\gamma_u s_j}} \\ &= -\log(1 + \exp(-\gamma_u(s_i - s_j))), \end{aligned} \tag{3.6}$$

which follows from the fact that the difference between two independent Gumbel random variables has the logistic distribution. We note that setting $\gamma_u = 1$ recovers the traditional BTL model (Bradley and Terry 1952).

If ϵ_i follows the standard normal distribution, we obtain the following *Heterogeneous Thurstone Case V* (HTCV) model:

$$\log \Pr(Y_{ij}^u = 1; s_i, s_j, \gamma_u) = \log \Phi\left(\frac{\gamma_u(s_i - s_j)}{\sqrt{2}}\right), \tag{3.7}$$

where Φ is the CDF of the standard normal distribution. Again, when $\gamma_u = 1$, this reduces to Thurstone's Case V (TCV) model for pairwise comparisons (Thurstone 1927).

Adversarial users: Under our heterogeneous framework, we can also model a certain class of adversarial users, whose goal is to make the estimated ranking be the opposite of the true ranking, so that, for example, an inferior item is ranked higher than the alternatives. We assume for adversarial users, the score of item i is $C - s_i$, for some constant C. Changing s_i to $C - s_i$ in (3.5) is equivalent to assuming the user has a negative accuracy γ_u. In this way, the accuracy of the user is determined by the magnitude $|\gamma_u|$ and its trustworthiness by $\text{sign}(\gamma_u)$, as illustrated in Figure 1. When adversarial users are present, this will facilitate optimizing the loss function, since instead of solving the combinatorial optimization problem of deciding which users are adversarial, we simply optimize the value of γ_u for each user.

One relevant work to ours is the CrowdBT algorithm proposed by (Chen et al. 2013), where they also explored the accuracy level of different users in learning a global ranking. In particular, they assume that each user has a probability η_u of making mistakes in comparing items i and j: $\Pr(Y_{ij}^u = 1; s_i, s_j, \eta_u) = \eta_u \Pr(i \succ j) + (1 - \eta_u) \Pr(j \succ i)$, where $\Pr(i \succ j)$ and $\Pr(j \succ i)$ follow the BTL model. This translates to introducing a parameter in the likelihood function to quantify the reliability of each pairwise comparison.

This parameterization, however, deviates from the additive noise in Thurstonian models defined as in (3.1) such as BTL and Thurstone's Case V. Specifically, the Thurstonian model explains the noise observed in pairwise comparisons as resulting from the additive noise in estimating the latent item scores. Therefore, the natural extension of Thurstonian models to a heterogeneous population of users is to allow different noise levels for different users, as was done in (3.3). As a result, CrowdBT cannot be easily extended to settings where more than two items are compared at a time. In contrast, the model proposed here is capable to describe such generalizations of Thurstonian models, such as the PL model.

4 Optimization and Rank Aggregation

In this section, we define the pairwise comparison loss function for the population of users and propose an efficient and effective optimization algorithm to minimize it. We denote by $\mathcal{D}_u$ the set of all pairwise comparisons made by user u on any two distinct items from set $[n]$. We denote by $\boldsymbol{Y}^u$ the matrix containing all pairwise preferences Y_{ij}^u of user u on items i and j. The entries of $\boldsymbol{Y}^u$ are $0/1/?$, where $?$ indicates that the pair was not compared by the user. We define the loss function for each user u as

$$\begin{aligned}\mathcal{L}_u\left(\boldsymbol{s}, \gamma_u; \boldsymbol{Y}^u\right) &= -\frac{1}{k_u} \sum_{(i,j)\in\mathcal{D}_u} \log \Pr(Y_{ij}^u = 1; s_i, s_j, \gamma_u) \\ &= -\frac{1}{k_u} \sum_{(i,j)\in\mathcal{D}_u} \log F\left(\gamma_u(s_i - s_j)\right),\end{aligned}$$

where $k_u = |\mathcal{D}_u|$ is the number of comparisons by user u. Then, the total loss function for m users is

$$\mathcal{L}\left(\boldsymbol{s}, \boldsymbol{\gamma}; \boldsymbol{Y}\right) = \frac{1}{m} \sum_{u=1}^{m} \mathcal{L}_u\left(\boldsymbol{s}, \gamma_u; \boldsymbol{Y}^u\right), \tag{4.1}$$

where $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_m)^\top$, $\boldsymbol{Y} = (\boldsymbol{Y}^1, \ldots, \boldsymbol{Y}^m)$. We denote the unknown true score vector as $\boldsymbol{s}^*$ and the true accuracy vector as $\boldsymbol{\gamma}^*$. Given observation $\mathcal{D}$, our goal is to recover $\boldsymbol{s}^*$ and $\boldsymbol{\gamma}^*$ via minimizing the loss function in (4.1). To ensure the identifiability of $\boldsymbol{s}^*$, we follow (Negahban, Oh, and Shah 2017) to assume that $\mathbf{1}^\top \boldsymbol{s}^* = \sum_{i=1}^n s_i^* = 0$, where $\mathbf{1} \in \mathbb{R}^n$ is the all one vector. The following proposition shows that the loss function $\mathcal{L}$ is convex in $\boldsymbol{s}$ and in $\boldsymbol{\gamma}$ separately if the PDF of ϵ_i is log-concave.

Proposition 4.1. If the distribution of the noise ϵ_i in (3.3) is log-concave, then the loss function $\mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}; \boldsymbol{Y})$ given in (4.1) is convex in $\boldsymbol{s}$, and in $\boldsymbol{\gamma}$ respectively.

The log-concave family includes many well-known distributions such as normal, exponential, Gumbel, gamma and beta distributions. In particular, the noise distributions used in BTL and Thurstone's Case V (TCV) models fall into this category. Although the loss function $\mathcal{L}$ is non convex with respect to the joint variable $(\boldsymbol{s}, \boldsymbol{\gamma})$, Proposition 4.1 inspires us to perform alternating gradient descent (Jain, Netrapalli, and Sanghavi 2013) on $\boldsymbol{s}$ and $\boldsymbol{\gamma}$ to minimize the loss function. As is shown in Algorithm 1, we perform alternating gradient descent update on $\boldsymbol{s}$ (or $\boldsymbol{\gamma}$) while fixing $\boldsymbol{\gamma}$ (or $\boldsymbol{s}$) at each iteration. In addition to the alternating gradient descent steps,

Algorithm 1 HTMs with Alternating Gradient Descent

1: **input:** learning rates $\eta_1, \eta_2 > 0$, initial points $\boldsymbol{s}^{(0)}$ and $\boldsymbol{\gamma}^{(0)}$ satisfying $\|\boldsymbol{s}^{(0)} - \boldsymbol{s}^*\|_2^2 + \|\boldsymbol{\gamma}^{(0)} - \boldsymbol{\gamma}^*\|_2^2 \le r$, number of iteration T, comparison results by users $\boldsymbol{Y}$.
2: **for** $t = 0, \ldots, T-1$ **do**
3: $\quad \widetilde{\boldsymbol{s}}^{(t+1)} = \boldsymbol{s}^{(t)} - \eta_1 \nabla_{\boldsymbol{s}} \mathcal{L}\big(\boldsymbol{s}^{(t)}, \boldsymbol{\gamma}^{(t)}; \boldsymbol{Y}\big)$
4: $\quad \boldsymbol{s}^{(t+1)} = (\mathbf{I} - \mathbf{1}\mathbf{1}^\top/n)\widetilde{\boldsymbol{s}}^{(t+1)}$
5: $\quad \boldsymbol{\gamma}^{(t+1)} = \boldsymbol{\gamma}^{(t)} - \eta_2 \nabla_{\boldsymbol{\gamma}} \mathcal{L}\big(\boldsymbol{s}^{(t)}, \boldsymbol{\gamma}^{(t)}; \boldsymbol{Y}\big)$
6: **end for**
7: **output:** $\boldsymbol{s}^{(T)}$, $\boldsymbol{\gamma}^{(T)}$.

we shift $\boldsymbol{s}^{(t)}$ in Line 4 of Algorithm 1 such that $\mathbf{1}^\top \boldsymbol{s}^{(t)} = 0$ to avoid the aforementioned identifiability issue of $\boldsymbol{s}^*$. After T iterations, given the output $\boldsymbol{s}^{(T)}$, the estimated ranking of the items is obtained by sorting $\{s_1^{(T)}, \ldots, s_n^{(T)}\}$ in descending order (item with the highest score in $\boldsymbol{s}^{(T)}$ is the most preferred).

As we will show in the next section, the convergence of Algorithm 1 to the optimal points $\boldsymbol{s}^*$ and $\boldsymbol{\gamma}^*$ is guaranteed if an initialization such that $\boldsymbol{s}^{(0)}$ and $\boldsymbol{\gamma}^{(0)}$ are close to the unknown parameters is available. In practice, to initialize $\boldsymbol{s}$, we can use the solution provided by the rank centrality algorithm (Negahban, Oh, and Shah 2012) or start from uniform or random scores. In this paper, we initialize $\boldsymbol{s}$ and $\boldsymbol{\gamma}$, as $\boldsymbol{s}^{(0)} = \mathbf{1}$ and $\boldsymbol{\gamma}^{(0)} = \mathbf{1}$. We note that multiplying $\boldsymbol{s}$ or $\boldsymbol{\gamma}$ by a negative constant does not alter the loss but reverses the estimated ranking. Implicit in our initialization is the assumption that the majority of the users are trustworthy and thus have positive γ. When data is sparse, there may be subsets of items that are not compared directly or indirectly. In such cases, regularization may be necessary, which is discussed in further detail in Section 6.

5 Theoretical Analysis of the Proposed Algorithm

In this section, we provide the convergence analysis of Algorithm 1 for the general loss function defined in (4.1). Without loss of generality, we assume the number of observations $k_u = k$ for all users $u \in [m]$ throughout our analysis. Since there's no specific requirement on the noise distributions in the general HTM model, to derive the linear convergence rate, we need the following conditions on the loss function $\mathcal{L}$, which are standard in the literature of alternating minimization (Jain, Netrapalli, and Sanghavi 2013; Zhu et al. 2017; Xu, Zhang, and Gu 2017; Xu, Ma, and Gu 2017; Zhang, Wang, and Gu 2018; Chen et al. 2018). Note that all these conditions can actually be verified once we specify the noise distribution in specific models. Due to the space limit, we provide the justifications of these conditions in the longer version of the paper.

Condition 5.1 (Strong Convexity)**.** $\mathcal{L}$ is μ_1-strongly convex with respect to $\boldsymbol{s} \in \mathbb{R}^n$ and μ_2-strongly convex with respect to $\boldsymbol{\gamma} \in \mathbb{R}^m$. In particular, there is a constant $\mu_1 > 0$ such that

for all $\boldsymbol{s}, \boldsymbol{s}' \in \mathbb{R}^n$,

$$\mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) \geq \mathcal{L}(\boldsymbol{s}', \boldsymbol{\gamma}) + \langle \nabla_{\boldsymbol{s}} \mathcal{L}(\boldsymbol{s}', \boldsymbol{\gamma}), \boldsymbol{s} - \boldsymbol{s}' \rangle + \mu_1/2 \|\boldsymbol{s} - \boldsymbol{s}'\|_2^2.$$

And there is a constant $\mu_2 > 0$ such that for all $\boldsymbol{\gamma}, \boldsymbol{\gamma}' \in \mathbb{R}^m$, it holds

$$\mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) \geq \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}') + \langle \nabla_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}'), \boldsymbol{\gamma} - \boldsymbol{\gamma}' \rangle + \mu_2/2 \|\boldsymbol{\gamma} - \boldsymbol{\gamma}'\|_2^2.$$

Condition 5.2 (Smoothness)**.** $\mathcal{L}$ is L_1-smooth with respect to $\boldsymbol{s} \in \mathbb{R}^n$ and L_2-smooth with respect to $\boldsymbol{\gamma} \in \mathbb{R}^m$. In particular, there is a constant $L_1 > 0$ such that for all $\boldsymbol{s}, \boldsymbol{s}' \in \mathbb{R}^n$, it holds

$$\mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) \leq \mathcal{L}(\boldsymbol{s}', \boldsymbol{\gamma}) + \langle \nabla_{\boldsymbol{s}} \mathcal{L}(\boldsymbol{s}', \boldsymbol{\gamma}), \boldsymbol{s} - \boldsymbol{s}' \rangle + L_1/2 \|\boldsymbol{s} - \boldsymbol{s}'\|_2^2.$$

And there is a constant $L_2 > 0$ such that for all $\boldsymbol{\gamma}, \boldsymbol{\gamma}' \in \mathbb{R}^m$, it holds

$$\mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) \leq \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}') + \langle \nabla_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}'), \boldsymbol{\gamma} - \boldsymbol{\gamma}' \rangle + L_2/2 \|\boldsymbol{\gamma} - \boldsymbol{\gamma}'\|_2^2.$$

The next condition is a variant of the usual Lipschitz gradient condition. It is worth noting that the gradient is derived with respect to $\boldsymbol{s}$ (or $\boldsymbol{\gamma}$), while the upper bound is the difference of $\boldsymbol{\gamma}$ (or $\boldsymbol{s}$). This condition is commonly imposed and verified in the analysis of expectation-maximization algorithms (Wang et al. 2015) and alternating minimization (Jain, Netrapalli, and Sanghavi 2013).

Condition 5.3 (First-order Stability)**.** There are constants $M_1, M_2 > 0$ such that $\mathcal{L}$ satisfies

$$\|\nabla_{\boldsymbol{s}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) - \nabla_{\boldsymbol{s}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}')\|_2 \leq M_1 \|\boldsymbol{\gamma} - \boldsymbol{\gamma}'\|_2,$$
$$\|\nabla_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) - \nabla_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{s}', \boldsymbol{\gamma})\|_2 \leq M_2 \|\boldsymbol{s} - \boldsymbol{s}'\|_2,$$

for all $\boldsymbol{s}, \boldsymbol{s}' \in \mathbb{R}^n$ and $\boldsymbol{\gamma}, \boldsymbol{\gamma}' \in \mathbb{R}^m$.

Note that the loss function in (4.1) is defined based on finitely many samples of observations. The next condition shows how close the gradient of the sample loss function is to the expected loss function.

Condition 5.4. Denote $\bar{\mathcal{L}}$ as the expected loss, where the expectation of $\mathcal{L}$ is taken over the random choice of the comparison pairs and the observation $\mathcal{D}$. With probability at least $1 - 1/n$, we have

$$\|\nabla_{\boldsymbol{s}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) - \nabla_{\boldsymbol{s}} \bar{\mathcal{L}}(\boldsymbol{s}, \boldsymbol{\gamma})\|_2 \leq \epsilon_1(k, n),$$
$$\|\nabla_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{s}, \boldsymbol{\gamma}) - \nabla_{\boldsymbol{\gamma}} \bar{\mathcal{L}}(\boldsymbol{s}, \boldsymbol{\gamma})\|_2 \leq \epsilon_2(k, n),$$

where n is the number of items and k is the number of observations for each user. In addition, $\epsilon_1(k, n)$ and $\epsilon_2(k, n)$ will go to zero when sample size k goes to infinity.

$\epsilon_1(k, n)$ and $\epsilon_2(k, n)$ in Condition 5.4 are also called the statistical errors (Wang et al. 2015; Xu, Ma, and Gu 2017) between the sample version gradient and the expected (population) gradient.

Now we deliver our main theory on the linear convergence of Algorithm 1 for general HTM models. Due to the space limit, full proofs can be found in the the longer version of the paper.

Theorem 5.5. For a general HTM model, assume Conditions 5.1, 5.2, 5.3 and 5.4 hold and that $M_1, M_2 \leq \sqrt{\mu_1 \mu_2}/4$. Denote that $\|\boldsymbol{s}^*\|_\infty = s_{\max}$ and $\|\boldsymbol{\gamma}^*\|_\infty = \gamma_{\max}$. Suppose the initialization guarantees that $\|\boldsymbol{s}^{(0)} - \boldsymbol{s}^*\|_2^2 + \|\boldsymbol{\gamma}^{(0)} - \boldsymbol{\gamma}^*\|_2^2 \leq r^2$, where $r = \min\{\mu_1/(2M_1), \mu_2/(2M_2)\}$. If we set the step size $\eta_1 = \eta_2 = \mu/(12(L^2 + M^2))$, where $L = \max\{L_1, L_2\}$, $\mu = \min\{\mu_1, \mu_2\}$ and $M = \max\{M_1, M_2\}$, then the output of Algorithm 1 satisfies

$$\|\boldsymbol{s}^{(T)} - \boldsymbol{s}^*\|_2^2 + \|\boldsymbol{\gamma}^{(T)} - \boldsymbol{\gamma}^*\|_2^2 \leq r^2 \rho^T + \frac{\epsilon_1(k, n)^2 + \epsilon_2(k, n)^2}{\mu^2}$$

with probability at least $1 - 1/n$, where the contraction parameter is $\rho = 1 - \mu^2/(48(L^2 + M^2))$.

Remark 5.6. Theorem 5.5 establishes the linear convergence of Algorithm 1 when the initial points are close to the unknown parameters. The first term on the right-hand side is called the optimization error, which goes to zero as iteration number t goes to infinity. The second term is called the statistical error of the HTM model, which goes to zero when sample size mk goes to infinity. Hence, the estimation error of our proposed algorithm converges to the order of $O((\epsilon_1(k, n)^2 + \epsilon_2(k, n)^2)/\mu^2)$ after $t = O(\log((\epsilon_1(k, n)^2 + \epsilon_2(k, n)^2)/\mu^2 r^2)/\log \rho)$ iterations.

Note that the results in Theorem 5.5 hold for any general HTM models with Algorithm 1 as a solver. In particular, if we run the alternating gradient descent algorithm on the HBTL and HTCV models proposed in Section 3, we will also obtain linear convergence rate to the true parameters up to a statistical error in the order of $O(n^2 \log(mn^2)/(mk))$, which matches the state-of-the-art statistical error for such models (Negahban, Oh, and Shah 2017). Due to space limit, we provide the implications of Theorem 5.5 on specific models in the longer version of the paper.

6 Experiments

In this section, we present experimental results to show the performance of the proposed algorithm on heterogeneous populations of users. The experiments are conducted on both synthetic and real data with both benign users and adversarial users. We use the Kendall's tau correlation (Kendall 1948) between the estimated and true rankings to measure the similarity between rankings, which is defined as $\tau = \frac{2(c-d)}{n(n-1)}$, where c and d are the number of pairs on which the two rankings agree and disagree, respectively. Pairs that are tied in at least one of the rankings are not counted in c or d.

Baseline methods: In Gumbel noise setting, we compare Algorithm 1 based on our proposed HBTL model with (1) the BTL model that can be optimized through iterative maximum-likelihood methods (Negahban, Oh, and Shah 2012) or spectral methods such as Rank Centrality (Negahban, Oh, and Shah 2017); and (2) the CrowdBT algorithm (Chen et al. 2013), which is a variation of BTL that allows users with different levels of accuracy. In the normal noise setting, we compare Algorithm 1 based on our proposed HTCV model with TCV model. We also implemented a TCV equivalent of CrowdBT and report its performance as CrowdTCV.

Experimental Results on Synthetic Data

We first evaluate our algorithms on synthetic data produced by a heterogeneous population of users.

Data generation: We set number of items $n = 20$, number of users $m = 9$ and set the ground truth score vector $\boldsymbol{s}$ to be uniformly distributed in $[0, 1]$. We divide the m users into groups A and B, consisting of 3 and 6 users respectively. These two groups of users generate heterogeneous data in the sense that users in group A are more accurate than those in group B. We vary γ_A in the range of $\{2.5, 5, 10\}$ and γ_B in the range of $\{0.25, 1, 2.5\}$, which leads to in total 9 configurations of data generation. For each configuration, we conduct the experiment under the following two settings:

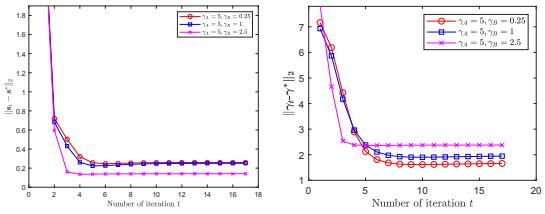
(1) **Benign:** $\gamma_1, \ldots, \gamma_3 = \gamma_A$ (Group A); $\gamma_4, \ldots, \gamma_9 = \gamma_B$ (Group B).

(2) **Adversarial:** $\gamma_1 = -\gamma_A$, $\gamma_2, \gamma_3 = \gamma_A$ (Group A); $\gamma_4, \gamma_5 = -\gamma_B$, $\gamma_6, \ldots, \gamma_9 = \gamma_B$ (Group B).

For each user and a given pair of items, pairwise comparison data is generated by comparing values produced according to the HTM model (3.3) with noise. Each pair of items is sent to the user 2 times for evaluation and is observed in the training dataset with probability $\alpha \in (0, 1)$. Due to the space limit, we only present the experimental results with $\alpha = 0.4$ here and defer more results to the the longer version of the paper.

Under setting (1), we perform rank aggregation using all the baseline methods. The experiment is repeated 100 times with different random seeds. We plot the estimation error of Algorithm 1 v.s. number of iterations for HBTL model in Figures 2(a)-2(b). In all settings, our algorithm enjoys a linear convergence rate to the true parameters up to statistical errors, which is well aligned with the theoretical results in Theorem 5.5.

Under setting (1), the ranking results for Gumbel noises under different configurations of γ_A and γ_B are shown in the first part of Table 1, where each cell presents the Kendall's tau correlation between the aggregated ranking and the ground truth, averaged over 100 trials. For each experimental setting, we use the bold text to denote the method which achieved the highest performance. We also underline the highest score whenever there is a tie. It can be observed that in almost all cases, HBTL provides much more accurate rankings than BTL. In particular, the larger the difference between γ_A and γ_B is, the more significant the improvement is. The only exception is when $\gamma_A = \gamma_B = 2.5$, in which case the data is not heterogeneous and our HTM model has no advantage.



(a) Estimation error for $\boldsymbol{s}^*$ (b) Estimation error for $\boldsymbol{\gamma}^*$

Figure 2: Evolution of estimation errors vs. number of iterations t for HBTL model.

Nevertheless, our method still achieve comparable performance as BTL for non-heterogeneous data. It can also be observed that HBTL generally outperforms CrowdBT. But the advantage is not large, as CrowdBT also includes the different accuracy levels of different users. Importantly, however, as discussed in Section 3, CrowdBT is not compatible with the additive noise in Thurstonian models and cannot be extended naturally to ranked data other than pairwise comparison. In addition, unlike CrowdBT, our method enjoys strong theoretical guarantees while maintaining a good performance. Table 1 also illustrates an important fact: If there are users with high accuracy, the presence of low quality data does not significantly impact the performance of Algorithm 1.

Under setting (2), we consider adversarial users whose accuracy level γ_u may take negative values as discussed above. The results for Gumbel noise under setting (2) are shown in the second part of Table 1. It can be seen that in this case, the difference between the methods is even more pronounced.

Due to the space limit, we defer the results with normal noise to the the longer version of the paper.

Experimental Results on Real-World Data

We evaluate our method on two real-world datasets. The first one named "Reading Level" (Chen et al. 2013) contains English text excerpts whose reading difficulty level is compared by workers. 624 workers annotated 490 excerpts which resulting in a total of 12,728 pairwise comparisons. Another dataset named "Country Population," collected by the authors, contains responses for pairwise comparison of populations of 15 countries from 199 workers. Each annotator was asked to provide 16 responses randomly generated from all possible country pairs. A detailed description is given in the the longer version of the paper. These two datasets were both collected in online crowdsourcing environments so that we can expect varying worker accuracy where effectiveness of our approach can be demonstrated.

In real-world datasets, it may happen that two items from two subsets are never compared with each other, directly or indirectly. In such cases, the ranking will not be unique. Furthermore, if data is sparse, the estimates may suffer from overfitting. To address these issues, regularization is often used. While this can be done in a variety of ways, for the sake of comparison with CrowdBT, we use virtual node regularization (Chen et al. 2013). Specifically, it is assumed that there is a virtual item of utility $s_0 = 0$ which is compared to all other items by a virtual user. This leads to the loss function $\mathcal{L} + \lambda_0 \mathcal{L}_0$, where $\mathcal{L}_0 = -\sum_{i \in [n]} \log F(s_0 - s_i) - \sum_{i \in [n]} \log F(s_i - s_0)$ and $\lambda_0 \geq 0$ is a tuning parameter.

We evaluate the performance of the methods for $\lambda_0 = 0, 1, 5, 10$. For different values of λ_0, HBTL performs best more often than any other method and, in particular, it performs best for $\lambda_0 = 0$. Table 2 reports the best performance of each method across different regularization values. It can be observed that HBTL and HTCV outperform their counterparts, CrowdBT and CrowdTCV, as well as the uniform models, BTL and TCV. Complete results are presented in the longer version of the paper.

Table 1: Kendall's tau correlation for different methods under Gumbel noise. Group A users all have the accuracy level γ_A and Group B users all have the accuracy level γ_B. In setting (1), i.e., the *Benign* setting, all the users have positive accuracy levels. In setting (2), i.e., the *Adversarial* setting, $1/3$ of the users in both groups have negative accuracy levels.

Settings	γ_B	Methods	γ_A		
			2.5	5	10
Benign	0.25	BTL	0.671±0.062	0.761±0.053	0.812±0.048
		CrowdBT	0.764±0.065	0.872±0.037	0.933±0.024
		HBTL	**0.769**±0.061	**0.873**±0.034	**0.934**±0.022
	1.0	BTL	0.791±0.051	0.844±0.043	0.866±0.035
		CrowdBT	0.798±0.050	0.889±0.029	0.934±0.027
		HBTL	**0.806**±0.051	**0.891**±0.031	**0.936**±0.026
	2.5	BTL	**0.882**±0.034	0.910±0.030	0.919±0.027
		CrowdBT	0.879±0.034	0.912±0.026	0.943±0.022
		HBTL	0.880±0.032	**0.916**±0.028	**0.945**±0.020
Adversarial	0.25	BTL	0.323±0.130	0.405±0.132	0.485±0.109
		CrowdBT	0.742±0.169	0.877±0.033	**0.934**±0.025
		HBTL	**0.766**±0.059	0.877±0.035	0.933±0.024
	1.0	BTL	0.448±0.118	0.544±0.096	0.583±0.094
		CrowdBT	0.810±0.044	0.886±0.031	0.934±0.026
		HBTL	**0.819**±0.045	**0.891**±0.031	0.934±0.029
	2.5	BTL	0.627±0.087	0.660±0.075	0.698±0.063
		CrowdBT	0.879±0.034	0.913±0.027	0.939±0.023
		HBTL	**0.880**±0.032	**0.914**±0.029	**0.948**±0.022

Table 2: Performance of ranking algorithms on real-world dataset.

Dataset	BTL	TCV	CrowdBT	CrowdTCV	HBTL	HTCV
Reading Level	0.3472	0.3452	0.3737	0.3672	**0.3763**	0.3729
Country Population	0.7524	0.7524	0.7714	0.7714	**0.7905**	0.7714

7 Conclusions and Future Work

In this paper, we propose the heterogeneous Thurstone model for pairwise comparisons and partial rankings when data is produced by a population of users with diverse levels of expertise, as is often the case in real-world applications. The proposed model maintains the generality of Thurstone's framework and thus also extends common models such as Bradley-Terry-Luce, Thurstone's Case V, and Plackett-Luce. We also developed an alternating gradient descent algorithm to estimate the score vector and expertise level vector simultaneously. We prove the local linear convergence of our algorithm for general HTM models satisfying mild conditions. We also prove the convergence of our algorithm for the two most common noise distributions, which leads to the HBTL and HTCV models. Experiments on both synthetic and real data show that our proposed model and algorithm generally outperforms the competing methods, sometimes by a significant margin.

There are several interesting future directions that could be explored. First, it would be of great importance to devise a provable initialization algorithm since our current analysis relies on certain initialization methods that are guaranteed to be close to the true values. Another direction is extending the algorithm and analysis to the case of partial ranking such as the Plackett-Luce model. Finally, lower bounds on the estimation error would enable better evaluating algorithms for rank aggregation in heterogeneous Thurstone models.

Acknowledgment

We would like to thank the anonymous reviewers for their helpful comments. We would like to thank Ashish Kumar for the collection of "Country Population" dataset. PX and QG are supported in part by the NSF grants CIF-1908544, III-1904183 and CAREER Award 1906169. TJ and FF are supported in part by the NSF grants CIF-1911168 and CCF-1908544. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

Aerts, S.; Lambrechts, D.; Maity, S.; Van Loo, P.; Coessens, B.; De Smet, F.; Tranchevent, L.-C.; De Moor, B.; Marynen, P.; Hassan, B.; Carmeliet, P.; and Moreau, Y. 2006. Gene prioritization through genomic data fusion. *Nature Biotechnology* 24(5):537–544.

Baltrunas, L.; Makcinskas, T.; and Ricci, F. 2010. Group Recommendations with Rank Aggregation and Collaborative

Filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, 119–126. New York, NY, USA: ACM.

Bradley, R. A., and Terry, M. E. 1952. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika* 39(3/4):324–345.

Braverman, M., and Mossel, E. 2008. Noisy Sorting Without Resampling. In *ACM-SIAM Symp. Discrete Algorithms (SODA)*, 268–276. San Francisco, California: Society for Industrial and Applied Mathematics.

Chen, Y., and Suh, C. 2015. Spectral MLE: Top-k rank aggregation from pairwise comparisons. In *International Conference on Machine Learning*, 371–380.

Chen, X.; Bennett, P. N.; Collins-Thompson, K.; and Horvitz, E. 2013. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 193–202. ACM.

Chen, J.; Xu, P.; Wang, L.; Ma, J.; and Gu, Q. 2018. Covariate adjusted precision matrix estimation via nonconvex optimization. In *International Conference on Machine Learning*, 921–930.

de Borda, J.-C. 1781. Mémoire sur les élections au scrutin. *Histoire de l'Académie royale des sciences*.

de Condorcet, M. 1785. *Essai Sur l'application de l'analyse à La Probabilité Des Décisions Rendues à La Pluralité Des Voix*. L'imprimerie royale.

Dwork, C.; Kumar, R.; Naor, M.; and Sivakumar, D. 2001. Rank aggregation methods for the web. In *Proc. 10th Int. Conf. World Wide Web*, 613–622. ACM.

Guiver, J., and Snelson, E. 2009. Bayesian Inference for Plackett-Luce Ranking Models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, 377–384. New York, NY, USA: ACM.

Hajek, B.; Oh, S.; and Xu, J. 2014. Minimax-optimal Inference from Partial Rankings. In *Advances in Neural Information Processing Systems 27*, 1475–1483.

Hunter, D. R. 2004. MM algorithms for generalized Bradley-Terry models. *The Annals of Statistics* 32(1):384–406.

Jain, P.; Netrapalli, P.; and Sanghavi, S. 2013. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, 665–674. ACM.

Kendall, M. 1948. *Rank Correlation Methods*. London: Griffin.

Kim, M.; Farnoud, F.; and Milenkovic, O. 2015. HyDRA: Gene prioritization via hybrid distance-score rank aggregation. *Bioinformatics* 31(7):1034–1043.

Kumar, A., and Lease, M. 2011. Learning to Rank from a Noisy Crowd. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, 1221–1222. New York, NY, USA: ACM.

Luce, R. D. 1959. *Individual Choice Behavior: A Theoretical Analysis*. New York: John Wiley & Sons, Inc.

Negahban, S.; Oh, S.; and Shah, D. 2012. Iterative ranking from pair-wise comparisons. In *Advances in Neural Information Processing Systems 25*.

Negahban, S.; Oh, S.; and Shah, D. 2017. Rank Centrality: Ranking from Pairwise Comparisons. *Operations Research* 65(1):266–287.

Raman, K., and Joachims, T. 2014. Methods for ordinal peer grading. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1037–1046. ACM.

Raman, K., and Joachims, T. 2015. Bayesian ordinal peer grading. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*, 149–156. ACM.

Thurstone, L. L. 1927. A law of comparative judgment. *Psychological Review* 34(4):273–286.

Vojnovic, M., and Yun, S. 2016. Parameter Estimation for Generalized Thurstone Choice Models. In *PMLR*, 498–506.

Wang, Z.; Gu, Q.; Ning, Y.; and Liu, H. 2015. High dimensional em algorithm: Statistical optimization and asymptotic normality. In *Advances in neural information processing systems*, 2521–2529.

Wauthier, F.; Jordan, M.; and Jojic, N. 2013. Efficient Ranking from Pairwise Comparisons. In *PMLR*, 109–117.

Weng, R. C., and Lin, C.-J. 2011. A Bayesian approximation method for online ranking. *Journal of Machine Learning Research* 12(Jan):267–300.

Xu, P.; Ma, J.; and Gu, Q. 2017. Speeding up latent variable Gaussian graphical model estimation via nonconvex optimization. In *Advances in Neural Information Processing Systems*, 1933–1944.

Xu, P.; Zhang, T.; and Gu, Q. 2017. Efficient algorithm for sparse tensor-variate Gaussian graphical models via gradient descent. In *Artificial Intelligence and Statistics*, 923–932.

Yu, P. L. H. 2000. Bayesian analysis of order-statistics models for ranking data. *Psychometrika* 65(3):281–299.

Zermelo, E. 1929. Die Berechnung der Turnier-Ergebnisse als ein Maximumproblem der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift* 29:436–460.

Zhang, X.; Wang, L.; and Gu, Q. 2018. A unified framework for nonconvex low-rank plus sparse matrix recovery. In *International Conference on Artificial Intelligence and Statistics*, 1097–1107.

Zhao, Z.; Villamil, T.; and Xia, L. 2018. Learning mixtures of random utility models. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Zhu, R.; Wang, L.; Zhai, C.; and Gu, Q. 2017. High-dimensional variance-reduced stochastic gradient expectation-maximization algorithm. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 4180–4188. JMLR. org.