Minimax Rate for Learning From Pairwise Comparisons in the BTL Model

Julien Hendrickx [1] **Alex Olshevsky** [2] **Venkatesh Saligrama** [2]

Abstract

We consider the problem of learning the qualities $w_1, \ldots, w_n$ of a collection of items by performing noisy comparisons among them. A standard assumption is that there is a fixed "comparison graph" and every neighboring pair of items is compared k times. We will study the popular Bradley-Terry-Luce model, where the probability that item i wins a comparison against j equals $w_i/(w_i + w_j)$. The goal is to understand how the expected error in estimating the vector $w = (w_1, \ldots, w_n)$ behaves in the regime when the number of comparisons k is large. Our contribution is the determination of the minimax rate up to a constant factor. We show that this rate is achieved by a simple algorithm based on weighted least squares, with weights determined from the empirical outcomes of the comparisons. This algorithm can be implemented in nearly linear time in the total number of comparisons.

1. Introduction

Estimation of item qualities from user preferences is a common problem across multiple domains in e-commerce, health care, and social science. The dominant approach is to rely on raw scores provided by users; for instance, Amazon asks customers for ratings on a scale ranging from 1-5 stars, which are then aggregated to produce an average rating for each item.

Unfortunately, such user-provided scores can be poorly calibrated. Users could differ substantially in how they reach the decision to assign scores; worse, different items could be popular among different classes of users, and these user classes could have statistical differences in the way they assign ratings. It is challenging to deal with this disparity in a principled manner.

[1]ICTEAM, UCLouvain [2]Boston University. Correspondence to: Alex Olshevsky <alexols@bu.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

An alternative line of research has explored data fusion based on better calibrated measures, such as the outcomes of comparisons among items. In many contexts, comparison data is readily available. When a user chooses to purchase one of several items recommended by a webpage, it is natural to view this as the outcome of an implicit comparison. The outcome of a sports game can be viewed as the result of a noisy comparison of the strengths of the two teams. Finally, when users click on a particular webpage in response to a list of sites provided by a search engine, this may be viewed as the outcome of a comparison between user estimates of the informativeness of the corresponding webpages. Many additional examples can be given and we refer the reader to (Cattelan, 2012) for an extensive overview of comparison models and their uses.

The simplest and most common model is the Bradley-Terry-Luce (BTL) model (Bradley & Terry, 1952; Luce, 2012) which posits n items with quality measures $w_1, \ldots, w_n$, with item i winning each comparison against item j independently with probability $w_i/(w_i + w_j)$. All comparisons in this model are pairwise. The BTL model is extremely well-studied; for a sampling of its uses, we mention its applications to an empirical analysis of sports tournaments (Cattelan *et al.* , 2013), measurements of pain among patients (Matthews & Morris, 1995), estimating driver crash risks (Li & Kim, 2000), and testing the power of arguments in referendums (Loewen *et al.* , 2012), among many others.

This paper is concerned with estimating the vector $w = (w_1, \ldots, w_n)$ from the outcomes of comparisons carried out according to the BTL model. It is standard to assume that there is a given undirected graph $G = (\{1, \ldots, n\}, E)$, and every pair of neighbors in G is compared k times. The goal is to recover the vector of true weights vector w. Note, however, that since scaling every entry of w does not change the probability distribution of the outcomes under the BTL model, what can actually be recovered is a normalized version of w.

This problem formulation is standard in the literature; in particular, its analysis has been the subject of a number of recent papers, e.g., (Negahban *et al.* , 2012; Rajkumar & Agarwal, 2014; Negahban *et al.* , 2016; Agarwal *et al.* , 2018; Hendrickx *et al.* , 2019). One can, of course, introduce a number of complicating factors (e.g., more general

comparisons models, active comparisons, different numbers of comparisons across each edge, simultaneous comparisons of multiple items, etc), and we below survey a number of works analyzing these extensions. However, surprisingly it turns out that, despite literature on the BTL model dating back to the 1950s, many fundamental questions in this simplest setting remain open.

In this paper, we address one of those questions, namely understanding the rate at which the error in the recovery of w decays with the number of comparisons per edge k in terms of the graph G and the true weight vector w.

We will propose an algorithm for the recovery of w based on nonlinearly scaled weighted least-squares. Our main contribution is to show that, up to a constant factor, this algorithm achieves the asymptotic minimax rate for this problem, which we characterize in terms of the trace of a certain matrix depending both on the graph G and the weights w.

1.1. Previous work

The earliest references on the BTL model are (Bradley & Terry, 1952; Rao & Kupper, 1967; Davidson, 1970; Beaver & Gokhale, 1975) dating back to 1950s-1970s. These works focused on maximum likelihood estimation and hypothesis testing. We mention in particular (Beaver, 1977), which proposed doing so with a least squares approach, which is in the same spirit as the method proposed in this paper. The problem was first introduced in the context of internet search in the now-classic paper (Dwork *et al.* , 2001). Several methods for the general class of problems of rank aggregation were proposed in (Dwork *et al.* , 2001), particularly a method based on encoding qualities as the stationary distribution of a Markov chain built from the outcomes of comparisons.

An extremely large literature on analysis of pairwise comparisons has sprung within the statistics and machine learning literature in the past two decade and, a a result, it is not possible to survey all the work that has been done. There are many variations of the problem that have been studied, from more sophisticated models such as Thurstone and Placket-Luce (Hajek *et al.* , 2014; Maystre & Grossglauser, 2015), to online or bandit versions (Szörényi *et al.* , 2015; Yue *et al.* , 2012), to models with active learning (Jamieson & Nowak, 2011; Ailon, 2012), to models with multiple users with potentially different preferences among items (Wu *et al.* , 2015). We next focus only on papers most directly related to our work, namely papers concerned with rates for recovery of the true weights w in the BTL model.

The first rigorous analysis of the error rate in the pairwise case appeared in (Negahban *et al.* , 2012) in the case of a random comparison graph and in (Negahban *et al.* , 2016) for an arbitrary graph. The underlying method recovered an estimate $\hat{W}$ from the stationary distribution of a Markov chain constructed based on the outcomes of the comparisons. By construction, the elements of $\hat{W}$ summed to one, which made it natural to compare $\hat{W}$ with the normalized version of the true weights $w/||w||_1$.

It was shown in (Negahban *et al.* , 2016) that, for a number of comparisons k large enough as a function of the graph G, assuming that the weight imbalance is bounded as

$$\max_{i,j} \frac{w_i}{w_j} \leq b, \tag{1}$$

then with high probability we have that

$$\frac{\left|\left|\frac{w}{||w||_1} - \hat{W}\right|\right|_2^2}{\left|\left|\frac{w}{||w||_1}\right|\right|_2^2} \leq O\left(\frac{1}{k}\right) \frac{b^5 \log n}{(1-\rho)^2} \frac{d_{\max}}{d_{\min}^2}, \tag{2}$$

where $d_{\max}, d_{\min}$ are the largest/smallest degrees in the comparison graph and $1-\rho$ is the spectral gap of the random walk on the comparison graph G.

To understand how this scales in terms of the number of nodes n, we can use the results of (Landau & Odlyzko, 1981) which show that $1/(1-\rho)$ for a simple random walk on any graph will have worst-case scaling of $O(n^3)$. Thus the right-hand side above has a worst-case scaling of $O(n^7 \log n)/k$.

To our knowledge (Negahban *et al.* , 2012; 2016) represent the first understanding of how error bounds for w scale in terms of the corresponding graph. A consequence of those results is that a good approximation to the (scaled) true weights w can be found using a polynomial number of samples. Moreover, the results of (Negahban *et al.* , 2016) suggest a natural open problem: to understand just how fast the error decays for the best possible method.

The bounds of (Negahban *et al.* , 2016) were recently improved in (Agarwal *et al.* , 2018), resulting in a better scaling with b and replacing $d_{\text{avg}}/d_{\min}$ with $d_{\text{avg}}/d_{\min}$, among other improvements. Moreover, improved bounds in the somewhat more restrictive setting when comparisons are made over the complete graph, but with each pair of edges sampled independently (at rates that could differ across edges) were obtained in (Rajkumar & Agarwal, 2014).

Considerably more general models of ranking are quite common in the literature; in particular, we mention the papers (Rajkumar & Agarwal, 2016; Shah *et al.* , 2016; Negahban *et al.* , 2018), discussed next. In (Rajkumar & Agarwal, 2016), the class of ranking models learnable from a random comparison graph G with average degree that scales as $\log(n)$ was studied, and it was shown that this possible under a certain "low-rank" condition on the underlying model. In (Shah *et al.* , 2016) namely estimating w under a general ranking model parametrized by a nonlinear function which

included the BTL model was a special case. Adopting the normalization condition $\sum_{i=1}^n \log w_i = 0$, upper and lower bounds were shown in (Shah *et al.* , 2016) after m comparisons for $E\left|\left|\hat{W} - \log w\right|\right|_2^2$; the upper bound scaled with $(n/m)\lambda_2(L)^{-1}$, where L is the Laplacian of the comparison graph, and the lower bound had a complicated dependence on the Laplacian spectrum. In (Negahban *et al.* , 2018) upper and lower bounds depending on the Laplacian spectrum were derived for the multinomial logit model, which is much more general than the BTL model.

For the BTL model specifically, progress towards the best rate was made in the recent paper (Hendrickx *et al.* , 2019). The error measure considered in that paper was the sine of the angle made by $\hat{W}$ and w, which can be expressed as

$$|\sin(\hat{W}, w)| = \inf_{\alpha} \frac{||\alpha\hat{W} - w||_2}{||w||_2}.$$

The sine of the angle is a standard way to measure distance between subspaces and, as the above identity suggests, it can be thought of as the relative distance between w and the best normalized version of $\hat{W}$. Moreover, because $\sin(\theta) \approx \theta$ for small θ, this error measure is essentially the same (provided the number of samples is large) as measuring the angle between $\hat{W}$ and w. Additionally, as remarked in (Hendrickx *et al.* , 2019) it can be shown that

$$\frac{1}{\sqrt{2}}\left|\left|\frac{x}{||x||_2} - \frac{y}{||y||_2}\right|\right|_2 \leq |\sin(x,y)| \leq \left|\left|\frac{x}{||x||_2} - \frac{y}{||y||_2}\right|\right|_2$$

so that the sine is, up to a constant, the error in the two-norm after normalization. Finally, the sine is also equivalent, up to polynomial factors of b, to previous metrics used in this problem. In particular, it was shown in (Hendrickx *et al.* , 2019) that the sine is within a $\sqrt{b}$ multiplicative factor of the norm used in (Negahban *et al.* , 2012) (see left-hand side of Eq. (2) and it can be shown it is within a polynomial factor of $E\left|\left|\hat{W} - \log w\right|\right|_2^2$ used in (Shah *et al.* , 2016).

Upper and lower bounds were established (Hendrickx *et al.* , 2019) on $\sin^2(\hat{W}, w)$, both holding when the number of samples per edge k is large enough. As far as upper bounds, it was shown that, for large enough k as a function of G and δ, with probability $1-\delta$ we have the bounds

$$\sin^2(\hat{W}, w) = O\left(\frac{b^2 R_{\max}(1+\log(1/\delta))}{k}\right) \quad (3)$$

$$\sin^2(\hat{W}, w) = O\left(\frac{b^4 R_{\text{avg}}(1+\log(1/\delta))}{k}\right), \quad (4)$$

where $R_{\text{avg}}, R_{\max}$ are, respectively, the average and largest electrical resistance[1] of the comparison graph G. A corresponding lower bound was proved showing that, for large enough k as a function of the graph G,

$$E\left[\sin^2(\hat{W}, w)\right] \geq \frac{R_{\text{avg}}}{k}. \quad (5)$$

These results come close, but do not quite characterize, the asymptotic minimax rate. Putting all the bounds together, it becomes clear that the electrical resistance is the key graph-theoretic quantity. However, there are gaps between the upper and lower bounds, both in terms of scaling with b and in terms of the difference between average and maximum resistance[2].

1.2. Our contribution

The purpose of the present paper is to present a new algorithm, coupled with new upper and lower bounds, which characterize the minimax rate for this problem (using the sine as a measure of distance). We will need the following definition: we set

$$\gamma(i,j) = \frac{1}{(w_i + w_j)^2},$$

and we use L_γ to mean the Laplacian of the graph G where edge (i,j) has weight $\gamma(i,j)$. We next state two theorems, the first providing an upper bound and the second providing a lower bound, which are the main results of this paper. Note that by definition of L_γ, the quantity $\frac{\text{Tr}\left(L_\gamma^\dagger\right)}{||w||_2^2}$ appearing in both results is invariant under scaling of the weights w_i.

Theorem 1. *For large enough k, there is a near-linear time method which produces a estimate $\hat{W}$ which satisfies*

$$E\left[\sin^2(\hat{W}, w)\right] \leq O\left(\frac{1}{k}\right)\frac{\text{Tr}\left(L_\gamma^\dagger\right)}{||w||_2^2}, \quad (6)$$

where $L_\gamma^\dagger$ refers to the Moore-Penrose pseudoinverse of L_γ. The method which accomplishes this is the WLSM described in Section 2.

Theorem 2. *Fix any $w \in \mathbb{R}^n$ and fix any map $\hat{w}$ from the outcomes of k comparisons across each edge to $\mathbb{R}^n$. There is a way to generate w_z randomly from a ball of radius $O_{w,G}(1/\sqrt{k})$ around w such that as long as k is large enough,*

$$E\left[\sin^2(\hat{w}(\mathbf{Y}), w_z)\right] \geq \Omega\left(\frac{1}{k}\right)\frac{\text{Tr}\left(L_\gamma^\dagger\right)}{||w||_2^2}, \quad (7)$$

where $\mathbf{Y}$ is the outcome of k comparisons across each edge generated according to weights w_z and $L_\gamma^\dagger$ refers to the Moore-Penrose pseudoinverse of L_γ.

[1] Resistances are defined in terms of the circuit obtained by replacing every edge in a graph by a resistor of unit resistance.

[2] There is also a gap in terms of the $\log(1/\delta)$ factor present in Eq. (3) and Eq. (4) but not in Eq. (5). However, this gap is not important, as can be expected to go away when integrating the high-probability bounds of Eq. (3) and Eq. (4) over δ to obtain a bound on the expectation.

We note that the constants in $O(\cdot)$ and $\Omega(\cdot)$ notations are absolute constants, i.e., they do not depend on any of the problem parameters, and in particular, they do not depend on b. However, the notation $O_{w,G}(1/\sqrt{k})$ in Theorem 2 means that the constant in the $O(\cdot)$-notation depends on w and the graph G. A key point is that the estimator $\hat{w}(\mathbf{Y})$ is arbitrary. Intuitively, this means that we can think of this estimator as "knowing" w as well as the distribution of w_z. Finally, we remark that it is easy to derive both the upper and lower bounds of (Hendrickx *et al.* , 2019) from these theorems using the well-known fact that the average graph resistance is proportional to the trace of the Laplacian pseudoinverse (see e.g., (Vishnoi, 2013))[3].

2. Our approach

The underlying intuition of approach is best explained by using a series of non-rigorous approximations. While our method will be formally analyzed in the supplementary information, in this section we make free use of such approximations.

For every pair of neighbors i, j in G, we will use F_{ij} to denote the fraction of times node i wins the comparisons against its neighbor j. It will be helpful sometimes to turn G into a directed graph by orienting every edge arbitrarily; we will use $\overrightarrow{E}$ to refer to the edge set of this directed graph.

Across each edge, we also define the ratio $R_{ij} = F_{ij}/F_{ji}$ which captures the imbalance between item qualities across the edge (i, j); indeed, by the strong law of large numbers,

$$R_{ij} \to \frac{w_i/(w_i + w_j)}{w_j/(w_i + w_j)} = \frac{w_i}{w_j}, \tag{8}$$

where the convergence would happen with probability one if we were to take the number of comparisons $k \to \infty$.

Our goal is to figure out the weights w_i from knowledge of the quantities R_{ij} for large but nevertheless finite k. One approach is to take the logarithm of both sides of Eq. (8) to obtain that

$$\log R_{ij} \approx \log w_i - \log w_j, \quad \text{for all edges } (i,j) \in \overrightarrow{E}.$$

The $\approx$ symbol hides the error that occurs from taking k finite. This is now a linear system of equations in the quantities $\log w_i$, so a natural approach is to solve the collection of equations

$$\log R_{ij} = z_i - z_j, \quad \text{for all edges } (i,j) \in \overrightarrow{E},$$

in the least-squares sense. In other words, we can try to find

$$z^* = \arg \min_{z_1,\ldots,z_n} \sum_{(i,j)\in E} (\log R_{ij} - (z_i - z_j))^2. \tag{9}$$

We can then build an estimator $\hat{W}$ of the item quality vector w by setting $\hat{W}_i = e^{z_i^*}$. This is exactly what is done in (Beaver, 1977; Hendrickx *et al.* , 2019) and broadly similar to the approach taken earlier in (Jiang *et al.* , 2011).

One disadvantage of this algorithm is that it does not take into consideration the differences in variance in comparisons across different edges. Indeed, observe that the variance in outcomes in comparing items i and j depends on the weights w_i and w_j. Furthermore, if the variance across an edge (i, j) is relatively low, then the corresponding squared term in Eq. (9) should have higher weight. This motivates a weighted least squares approach: we will divide each term in Eq. (9) by the standard deviation of $\log R_{ij}$.

In general, the standard deviation of $\log R_{ij}$ does not have a simple formula, but when k is large we can repeatedly write Taylor expansions of all quantities involved to turn everything approximately linear. The calculation is relatively simple and we perform it in the next few paragraphs; the uninterested reader may feel free to skip ahead to Eq. (11) to see the outcome.

Defining $\rho_{ij} = w_i/w_j$ to be the true ratio between qualities of items i and j, we can use $(\log x)' = 1/x$ to write

$$\begin{aligned}
\log R_{ij} &\approx \log \rho_{ij} + \frac{1}{\rho_{ij}}(R_{ij} - \rho_{ij}) \\
&= \log \rho_{ij} + \frac{1}{\rho_{ij}}\left(\frac{F_{ij}}{F_{ji}} - \rho_{ij}\right) \\
&= \log \rho_{ij} + \frac{1}{\rho_{ij}}\left(\frac{1 - F_{ji}}{F_{ji}} - \rho_{ij}\right) \\
&= \log \rho_{ij} + \frac{1}{\rho_{ij}}\left(\frac{1}{F_{ji}} - 1 - \rho_{ij}\right) \\
&\approx \log \rho_{ij} + \frac{1}{\rho_{ij}}\left(\frac{1}{p_{ji}} - \frac{1}{p_{ji}^2}(F_{ji} - p_{ji}) - \rho_{ij}\right),
\end{aligned}$$

where $p_{ji} = w_j/(w_i + w_j)$ is the correct probability of j winning against i, and the final step takes the linear Taylor approximation of $1/F_{ji}$ around its limit of $1/p_{ji}$.

[3] Indeed, the relation referred to is

$$R_{\text{avg}} = \frac{\text{Tr}(L^\dagger)}{n},$$

where L is the plain (unweighted) graph Laplacian (see e.g., (Vishnoi, 2013)). Thus taking $w = (1, \ldots, 1)$ in Theorem (2) we immediately recover Eq. (5). Similarly, rescaling w so that $\min_i w_i = 1$ we have that Eq. (1) implies that $\max_i w_i \le b$, and using the implications $||w||_2^2 \ge n$ and $(w_i + w_j)^2 = O(b^2)$, Theorem 1 immediately implies an upper bound of $O(b^2 R_{\text{avg}}/k)$, actually improving upon both Eq. (3) and Eq. (4).

The advantage of these manipulations is that they imply

$$\begin{aligned}\mathrm{var}(\log R_{ij}) &\approx \frac{1}{\rho_{ij}^2}\frac{1}{p_{ji}^4}\mathrm{var}(F_{ji}-p_{ji}),\\ &= \frac{w_i/w_j + w_j/w_i + 2}{k},\end{aligned}$$

where the last step follows by some simple algebraic manipulations. For simplicity, let us define

$$v_{ij} = \frac{w_i}{w_j} + \frac{w_j}{w_i} + 2. \tag{10}$$

Then what we really should do is solve the weighted least squares problem

$$\arg\min_{z_1,\ldots,z_n} \sum_{(i,j)\in\overrightarrow{E}} \frac{(\log R_{ij} - (z_i - z_j))^2}{\sqrt{v_{ij}/k}} \tag{11}$$

which properly accounts for the different variances of different comparisons. Indeed, each term in Eq. (11) now has the same variance as k gets large. Naturally, we can omit $1/\sqrt{k}$ from the denominator since it multiplies every term.

The big problem with this approach, of course, is that the quantities v_{ij} are actually unknown to us because we (obviously) do not know the true weights $w_1, \ldots, w_n$ a-priori. Thus as written Eq. (11) cannot be implemented.

Nevertheless, even though we do not know the quantities v_{ij}, we can construct estimates of them based on the data. Glancing at Eq. (10), a natual approach is to define

$$\hat{V}_{ij} = \frac{F_{ij}}{F_{ji}} + \frac{F_{ji}}{F_{ij}} + 2. \tag{12}$$

Indeed, if we consider what happens if we were to take $k \to \infty$, it follows from the strong law of large numbers that $\hat{V}_{ij} \to v_{ij}$ with probability one. Thus we simply replace each v_{ij} in Eq. (11) by its estimated counterpart:

$$z^* = \arg\min_{z_1,\ldots,z_n} \sum_{(i,j)\in\overrightarrow{E}} \frac{(\log R_{ij} - (z_i - z_j))^2}{\sqrt{\hat{V}_{ij}}}. \tag{13}$$

As before, constructing the estimator $\hat{W}$ will be done by setting $\hat{W}_i = e^{z_i^*}$.

We need to take one final step to have a well-defined algorithm. Clearly, we could have a problem when some $F_{kl} = 0$ because then we might run into the problem of using $\log R_{kl} = \log 0 = -\infty$ in our least squares objective. We resolve this problem by setting F_{kl} to be some small positive number (specifically, $F_{kl} = (1/2)/k$) in this case. Intuitively, when k is sufficiently large compared to b, the probability that some $F_{kl} = 0$ is exponentially small, so it doesn't really matter what we do; nevertheless, we need to do something in order to have a well-defined method.

This is the algorithm we will analyze in the remainder of this paper. We state it formally in the algorithm box below. We will refer to it as the Weighted Least Squares Method, or the WLSM for short.

Algorithm 1 Weighted Least Squares Method

Input: results of k independent comparisons across each edge in E.
for all $(i,j) \in E$ **do**
  Compute F_{ij}, the fraction of times item i wins.
  if $F_{ij} = 0$ **then**
    Set $F_{ij} = (1/2)/k$.
  else if $F_{ij} = 1$ **then**
    Set $F_{ij} = 1 - (1/2)/k$.
  end if
  Set $R_{ij} = F_{ij}/F_{ji}$.
end for
Compute the quantities $\hat{V}_{ij}$ using Eq. (12).
Solve Eq. (13) for the vector z^*.
For all $i = 1, \ldots, n$, set $\hat{W}_i = e^{z_i^*}$.

Finally, as we discuss in the Supplementary Information, this algorithm can be implemented in nearly linear time in the number of edges of G.

Unfortunately, the final procedure we have ended up with involves taking the ratio of two random variables constructed from the data (i.e., the quantities $\log R_{ij}$ and $\hat{V}_{ij}$), which will make analysis of the error in *expectation* challenging. To preview the analsysis of this method (which is available in the supplementary information) we will need to perform a large-deviations analysis of the outcome of the WLSM, which will then need to be integrated to obtain a bound on the expectation of $\sin^2(\hat{W}, w)$.

2.1. Linear time solvability

We now rewrite our algorithm in compact form; this rewriting will be needed later section to discuss the technical novelty in the proof, and will also, as a consequence, show that our algorithm can be implemented in (nearly) linear time.

First we discuss some notation. We let $\overrightarrow{E}$ denote the set of directed edges obtained by orienting every edge in E arbitrarily. We let M be the edge-vertex incidence matrix of the resulting directed graph $(\{1,\ldots,n\}, \overrightarrow{E})$; note that the graph Laplacian L satisfies $L = MM^T$. The matrices L_V and $L_{\hat{V}}$ correspond to weighted graph Laplacians, where the edge $(i,j) \in E$ is weighted by v_{ij}^{-1} or $\hat{V}_{ij}^{-1}$, respectively. We will omit the subscripts when we stack the above quantities into vectors. For example, the notation R represents the vector in $\mathbb{R}^{|\overrightarrow{E}|}$ obtained by stacking up the quantities $R_{ij}, (i,j) \in \overrightarrow{E}$.

With this notation in place, inspecting Eq. (13), we see that z^* is a least squares solution to the system of equations

$$\hat{V}^{-1/2} M^T z = \hat{V}^{-1/2} \log R.$$

Recall that $\hat{W}$ is our notation for $\hat{W} = e^{z^*}$; thus $\log \hat{W}$ is the least squares solution of

$$\hat{V}^{-1/2} M^T \log W = \hat{V}^{-1/2} \log R.$$

Writing out the least-squares solution explicitly, we have that $\log \hat{W}$ is an exact solution of the equation

$$(\hat{V}^{-1/2} M^T)^T \hat{V}^{-1/2} M^T \log \hat{W} = (\hat{V}^{-1/2} M^T)^T \hat{V}^{-1/2} \log R,$$

that is, of

$$L_{\hat{V}} \log \hat{W} = M \hat{V}^{-1} \log R. \tag{14}$$

Thus we have that one solution to Eq. (14) is

$$\log \hat{W} = L_{\hat{V}}^{\dagger} M \hat{V}^{-1} \log R. \tag{15}$$

Observe that since, by connectivity of G the null space of $L_{\hat{V}}$ is just $\mathrm{span}\{\mathbf{1}\}$, this picks up the solution of Eq. (14) which satisfies $\sum_{i=1}^n \log \hat{W}_i = 0$ or $\prod_{i=1}^n \hat{W}_i = 1$.

Concluding, we see that Eq. (15) is one way to represent a solution $\hat{W}$ we seek to compute. We can now observe that this is a Laplacian linear system, i.e., it requires multiplication by the pseudoinverse of a weighted graph Laplacian. We can now directly apply the results of (Spielman & Teng, 2014), which showed that it is possible to solve Eq. (15) in nearly-linear time, specifically in $O(|E| \log^c n \log(1/\epsilon))$ to accuracy ϵ.

Linear time solvability is important in the context of ranking from comparisons because it allows the underlying algorithm to potentially scale up to very large data-sets, such as those built from counts of web activity (i.e., clicks) or from systems with millions of users and many times that comparisons.

2.2. Main innovation in the proof

At a general level, there is a natural way to try to prove the main results of this paper: on the one hand, there are a variety of "two point estimates" which lower bound the expected error by finding pairs of weights that are as different as possible while giving rise to similar distributions on outcomes; more sophisticated approaches do the same over a distribution of weights. In the reverse direction, we can do a large deviations analysis of Eq. (15), which will involve having an accurate analysis of the behavior of a pseudoinverse of a random matrix. Once the lower and upper bounds obtained this way match, the optimal error rate will have been found. Most of the previous literature on the subject, e.g., (Negahban *et al.* , 2016; Shah *et al.* , 2016; Hendrickx *et al.* , 2019) used such an approach to derive upper or lower bounds.

Unfortunately, this appears to be difficult to carry out directly. Our analysis relies on a "trick" of analyzing a suitably regularized version of the problem, which we informally describe next; the full proof is of course available in the supplementary information.

Our starting point is Eq. (14), which shows that $\log \hat{W}$ is a solution of

$$L_{\hat{V}} \log W = M \hat{V}^{-1} \log R. \tag{16}$$

Of course, this equation has many solutions as $\mathbf{1}$ belongs to the null space of $L_{\hat{V}}$. The previous section defined a solution $\hat{W}$ of this equation, which was the solution with the elements of $\log \hat{W}$ summing to zero. For our analysis, we will find it convenient to "pick out" a different solution of Eq. (16). We proceed as follows.

First, we multiply both sides of Eq. (16) by $\mathrm{diag}(w)^{-1}$:

$$\mathrm{diag}(w)^{-1} L_{\hat{V}} \log W = \mathrm{diag}(w)^{-1} M \hat{V}^{-1} \log R.$$

We next introduce a new variable Y and reparametrize $\log W = \mathrm{diag}(w)^{-1} Y$ so that the last equation can be rewritten more symmetrically as

$$\mathrm{diag}(w)^{-1} L_{\hat{V}} \mathrm{diag}(w)^{-1} Y = \mathrm{diag}(w)^{-1} M \hat{V}^{-1} \log R.$$

As before, this equation has many solutions and we pick one arguably the most "natural" one by setting

$$\hat{Y} = \left(\mathrm{diag}(w)^{-1} L_{\hat{V}} \mathrm{diag}(w)^{-1}\right)^{\dagger} \mathrm{diag}(w)^{-1} M \hat{V}^{-1} \log R, \tag{17}$$

Our analysis will proceed by analyzing the quantity $\hat{Y}$. Naturally, we can use the relation $\log W = \mathrm{diag}(w)^{-1} Y$ to obtain that the quantity $\mathrm{diag}(w)^{-1} \hat{Y}$ is a solution of Eq. (16). It will be helpful to introduce new notation for the latter quantity:

$$\log \hat{W}^r = \mathrm{diag}(w)^{-1} \hat{Y}. \tag{18}$$

The quantity $\hat{W}^r$ is, of course, a rescaled version of $\hat{W}$ (because all solutions of Eq. (15) are rescaled versions of $\hat{W}$). It is possible to be more precise and observe that since the null space of $\mathrm{diag}(w)^{-1} L_{\hat{V}} \mathrm{diag}(w)^{-1}$ is span of w, we have that $\hat{Y}$ is orthogonal to w; which implies that $\log \hat{W}^r$ is orthogonal to w^2 (where the square is understood elementwise) or

$$\prod_{i=1}^n (\hat{W}^r)_i^{w_i^2} = 1. \tag{19}$$

Observe that, because we do not know the true weights w, we cannot compute $\hat{W}^r$. Nevertheless, we can still consider

it and analyze its properties, and whatever upper and lower bounds we obtain for the sine of the angle between $\hat{W}^r$ and w will apply to the solutions we can actually compute, since the angle between two vectors is unchanged if one of them is scaled. *It turns out that minimax optimal bounds come out of the analysis only after analyzing the solution $\hat{W}^r$ defined in Eq. (19).* Attempts based on other solutions of Eq. (16) resulted in upper and lower bounds that do not match (unless one introduces a scaling akin to considering $\hat{W}^r$ in the analysis). This is the main proof ingredient present in this paper that was not used in earlier works.

Analyzing the quantity $\hat{W}^r$ is the same as analyzing the solution $\log \widehat{W}$ of the underlying least-squares problem of Eq. (16) with smallest norm relative to the inner product $\langle x, x\rangle_w = \sum_{i=1}^n w_i^2 x_i^2$. Our approach may thus be viewed as part of a long line of research suggesting that the key is often to choose a metric that is natural for the problem. It is analysis with respect to this (scaled) inner product that ultimately leads to the weighted Laplacian L_γ appearing in our main results and not the ordinary Laplacian L.

3. Simulations and Two Conjectures

We perform a number of experiments designed to gauge the accuracy of the WLSM relative to competing methods. Since we are not aware of any real data sets involving comparisons where the true weights are known, we will use synthetic data. As we will see shortly, two conjectures are suggested by our results. We simulate five methods:

1. The least-squares method. This is the method that solves Eq. (9) for z^* and then sets $\hat{W}_i = e^{z_i^*}$. In the figures below, it is abbreviated as "LS."

2. Least squares with artificial weights. This solves for z^* using Eq. (11) and then sets $\hat{W}_i = e^{z_i^*}$ as above. It cannot be implemented in practice because we do not know the true variances v_{ij} used in Eq. (11), but it can be used as a useful benchmark to measure degradation in performance from using estimates of these variances. This is abbreviated "artif weight" in the figures.

3. Iterative least squares. This method begins by solving Eq. (11) by setting $v_{ij} = 1$. It then uses the computed w_{ij} to compute v_{ij} using Eq. (10), and then proceeds to re-solve Eq. (11). This cycle (new w_{ij} leading to new v_{ij} then leading to new w_{ij}) is then repeated. This is abbreviated by "iter weight" in the figures.

4. Our main algorithm, the WLSM method, which is abbreviated with "emp weight" in the figures.

5. The eigenvector-based algorithm of (Negahban *et al.* , 2012; 2016).

In general, we do not see much of a difference between any of the methods on simple graphs. Representative results are shown in Figure 1 for the 2D grid, the 3D grid, and the Erdos-Renyi random graph. While the method we propose in this paper is usually the best, the gains are extremely modest in the neighborhood of a few percent, as can be eyeballed from the figures. Only three graphs are shown because the pattern is the same on all graphs we have simulated.

However, with some experimentations we have found that the WLSM (along with other least-squares methods) has a significant advantage as compared against the eigenvector-based method in terms of accurately recovering *all* the weights, especially when there are many nodes of small weight. We give one example of such a graph in Figure 2. We take a line graph, pick two nodes that are a neighbor appart, and connect them through a complete bipartite graph with newly introduced nodes (on the right-hand side of the figure). The key idea is that the nodes on the right-hand side (labeled u_1, u_2, u_3 in the figure) will be assigned weight w_i of of 1, while the nodes on the left hand side will have weights that increase geometrically from 1 to b. Thus, for large b, the nodes $u_1, \ldots, u_3$ are not very relevant to (any notion of) distance between normalized versions of $\hat{W}$ and w due to their comparatively small weights. However, neglecting them has the effect of neglecting a large number of paths between w_3 and w_5 which can be used to help estimate the weights on the left-hand side.

Figure 3 shows the difference between $\hat{W}_3 - \hat{W}_5$ when $w_3 = w_5$ and there are approximately 50 nodes u_i on the right-hand side. We compare the difference $\hat{W}_3 - \hat{W}_5$ for both the WLSM and the eigenvector-based method of (Negahban *et al.* , 2012; 2016). Each number represents a single run of the algorithm with new random comparisons. We see that the WLSM outperforms by about an order of magnitude.

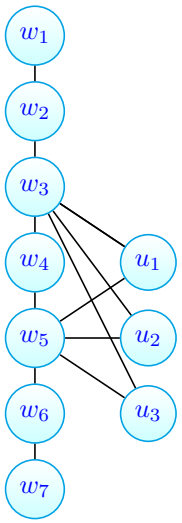
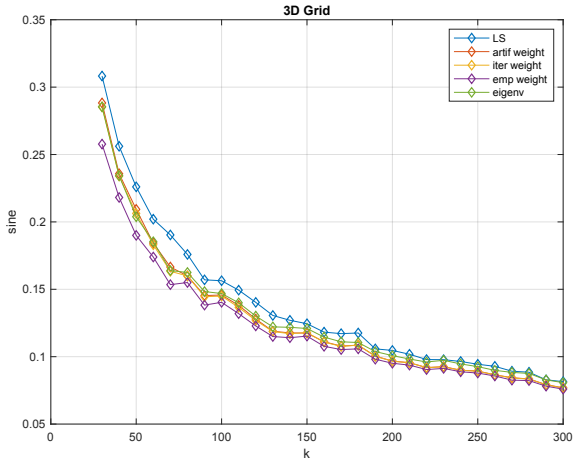
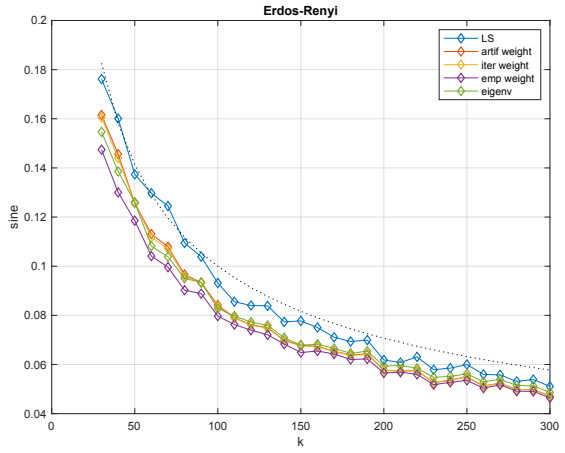


Figure 2. A graph on which the eigenvector-based approach underperform least-squares methods.

Our simulations thus point to two conjectures which can be the subject of further work. The first conjecture is that the earlier eigenvector based methods also achieve either the minimax scalings we have identified here, or something very





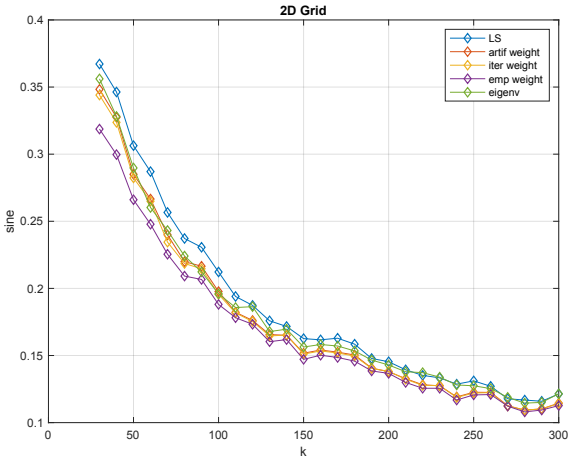


Figure 1. Performance on the 2D grid, 3D grid, and Erdos-Renyi graph. All three plots show $|\sin(\hat{W}, w)|$ on the y-axis vs the number of samples per edge on the x-axis. For the plots, the weights were generated randomly in the interval $[1, 20]$. The 2D and the E-R graph have 100 nodes, while the 3D grid has 125 nodes; the average degree of the E-R graph is 10. Each data point is the average of 50 simulations.

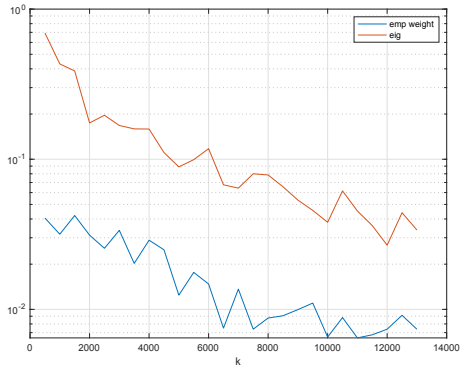


Figure 3. $\hat{W}_3 - \hat{W}_5$ for the eigenvector method in red and the WLSM in blue on the graph of Figure 2.

close to them, as our simulations do not appear to detect any significant difference in performance. Indeed, note that the 3D grid has a very strong divergence between average resistance (constant) and spectral gap ($\simeq n^{2/3}$), and yet our simulation on the 3D grid showed no difference between the eigenvector based method (which has been upper bounded in terms of scaling with the spectral gap) and the WLSM (which we know to scale with resistance).

Moreover, a plausible conjecture is that the methods in question achieve optimal performance not just in distance between the vectors $\hat{W}, w$ but also among $\hat{W}_i - w_i$ for each node i (after appropriate normalization). We conjecture this is indeed the case for the WLSM. However, our simulation suggests this may not be the case for the eigenvector method, as we have constructed an example (Figures 2 and 3) where it underperforms in this metric.

4. Conclusions

Our main contribution is the determination of the asymptotic minimax rate for inference from pairwise comparisons. In contrast to previous work, our result is exact up to constant factors.

Besides the conjectures discussed in Section 3, the most natural open question raised by our work is to understand how big the number of samples per edge k has to be for the minimax rate derived in this paper to kick in. We would actually conjecture that $\mathrm{tr}(L_\gamma^\dagger)/||w||_2^2$ is, up to constant factors, not only the minimax rate but also the sample complexity of recovering (a scaled version of) w.

Acknowledgements

This work was supported by the "*Learning from Pairwise Comparisons*" incentive grant (MIS) of the F.R.S.-FNRS, and by NSF grants ECCS-1933027, 2007350, 1527618 and 1955981.

References

Agarwal, Arpit, Patil, Prathamesh, & Agarwal, Shivani. 2018. Accelerated spectral ranking. *Pages 70–79 of: International Conference on Machine Learning*.

Ailon, Nir. 2012. An active learning algorithm for ranking from pairwise preferences with an almost optimal query complexity. *Journal of Machine Learning Research*, **13**(Jan), 137–164.

Beaver, Robert J. 1977. Weighted least-squares analysis of several univariate Bradley-Terry models. *Journal of the American Statistical Association*, **72**(359), 629–634.

Beaver, Robert J, & Gokhale, DV. 1975. A model to incorporat within-pair order effects in paired comparisons. *Communications in statistics-theory and methods*, **4**(10), 923–939.

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

Brualdi, Richard A, & Ryser, Herbert John. 1991. *Combinatorial matrix theory*. Vol. 39. Springer.

Bubeck, Sébastien. 2011. Introduction to online optimization. *Lecture Notes*, 1–86.

Cattelan, Manuela. 2012. Models for paired comparison data: A review with emphasis on dependent data. *Statistical Science*, 412–433.

Cattelan, Manuela, Varin, Cristiano, & Firth, David. 2013. Dynamic Bradley–Terry modelling of sports tournaments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **62**(1), 135–150.

Davidson, Roger R. 1970. On extending the Bradley-Terry model to accommodate ties in paired comparison experiments. *Journal of the American Statistical Association*, **65**(329), 317–328.

Dwork, Cynthia, Kumar, Ravi, Naor, Moni, & Sivakumar, Dandapani. 2001. Rank aggregation methods for the web. *Pages 613–622 of: Proceedings of the 10th international conference on World Wide Web*. ACM.

Foster, Ronald M. 1949. The average impedance of an electrical network. *Contributions to Applied Mechanics (Reissner Anniversary Volume)*, 333–340.

Golub, Gene H, & Pereyra, Victor. 1973. The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. *SIAM Journal on Numerical Analysis*, **10**(2), 413–432.

Hajek, Bruce, & Raginsky, Maxim. 2019. *Statistical Learning Theory*. http://maxim.ece.illinois.edu/teaching/SLT/SLT.pdf. Book draft.

Hajek, Bruce, Oh, Sewoong, & Xu, Jiaming. 2014. Minimax-optimal inference from partial rankings. *Pages 1475–1483 of: Advances in Neural Information Processing Systems*.

Hendrickx, Julien M, Olshevsky, Alex, & Saligrama, Venkatesh. 2019. Graph Resistance and Learning from Pairwise Comparisons. *In: International Conference on Machine Learning*.

Hsu, Daniel, Kakade, Sham, & Zhang, Tong. 2012. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability*, **17**.

Jamieson, Kevin G, & Nowak, Robert. 2011. Active ranking using pairwise comparisons. *Pages 2240–2248 of: Advances in Neural Information Processing Systems*.

Jiang, Xiaoye, Lim, Lek-Heng, Yao, Yuan, & Ye, Yinyu. 2011. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming*, **127**(1), 203–244.

Landau, Henry, & Odlyzko, Andrew. 1981. Bounds for eigenvalues of certain stochastic matrices. *Linear algebra and its Applications*, **38**, 5–15.

Li, Lel, & Kim, Karl. 2000. Estimating driver crash risks based on the extended Bradley–Terry model: an induced exposure method. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, **163**(2), 227–240.

Loewen, Peter John, Rubenson, Daniel, & Spirling, Arthur. 2012. Testing the power of arguments in referendums: A Bradley–Terry approach. *Electoral Studies*, **31**(1), 212–221.

Luce, R Duncan. 2012. *Individual choice behavior: A theoretical analysis*. Courier Corporation.

Matthews, JNS, & Morris, KP. 1995. An Application of Bradley-Terry-Type Models to the Measurement of Pain. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **44**(2), 243–255.

Maystre, Lucas, & Grossglauser, Matthias. 2015. Fast and accurate inference of Plackett–Luce models. *Pages 172–180 of: Advances in neural information processing systems*.

Negahban, Sahand, Oh, Sewoong, & Shah, Devavrat. 2012. Iterative ranking from pair-wise comparisons. *Pages 2474–2482 of: Advances in Neural Information Processing Systems*.

Negahban, Sahand, Oh, Sewoong, & Shah, Devavrat. 2016. Rank centrality: Ranking from pairwise comparisons. *Operations Research*, **65**(1), 266–287.

Negahban, Sahand, Oh, Sewoong, Thekumparampil, Kiran K, & Xu, Jiaming. 2018. Learning from comparisons and choices. *The Journal of Machine Learning Research*, **19**(1), 1478–1572.

Rajkumar, Arun, & Agarwal, Shivani. 2014. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. *Pages 118–126 of: International Conference on Machine Learning*.

Rajkumar, Arun, & Agarwal, Shivani. 2016. When can we rank well from comparisons of O (n\log (n)) non-actively chosen pairs? *Pages 1376–1401 of: Conference on Learning Theory*.

Rao, PV, & Kupper, Lawrence L. 1967. Ties in paired-comparison experiments: A generalization of the Bradley-Terry model. *Journal of the American Statistical Association*, **62**(317), 194–204.

Shah, Nihar B, Balakrishnan, Sivaraman, Bradley, Joseph, Parekh, Abhay, Ramchandran, Kannan, & Wainwright, Martin J. 2016. Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence. *The Journal of Machine Learning Research*, **17**(1), 2049–2095.

Spielman, Daniel A, & Teng, Shang-Hua. 2014. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *SIAM Journal on Matrix Analysis and Applications*, **35**(3), 835–885.

Szörényi, Balázs, Busa-Fekete, Róbert, Paul, Adil, & Hüllermeier, Eyke. 2015. Online rank elicitation for plackett-luce: A dueling bandits approach. *Pages 604–612 of: Advances in Neural Information Processing Systems*.

Tetali, Prasad. 1994. An extension of Foster's network theorem. *Combinatorics, Probability and Computing*, **3**(3), 421–427.

Vishnoi, Nisheeth. 2013. Lx=b. *Foundations and Trends in Theoretical Computer Science*, **8**(1–2), 1–141.

Wu, Rui, Xu, Jiaming, Srikant, Rayadurgam, Massoulié, Laurent, Lelarge, Marc, & Hajek, Bruce. 2015. Clustering and inference from pairwise comparisons. *Pages 449–450 of: ACM SIGMETRICS Performance Evaluation Review*, vol. 43. ACM.

Yue, Yisong, Broder, Josef, Kleinberg, Robert, & Joachims, Thorsten. 2012. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, **78**(5), 1538–1556.