Fast and Accurate Ranking Regression

İlkay Yıldız
Dept. of ECE
Northeastern Univ.
yildizi@ece.neu.edu

Jennifer Dy
Dept. of ECE
Northeastern Univ.
jdy@ece.neu.edu

Deniz Erdoğmuş
Dept. of ECE
Northeastern Univ.
erdogmus@ece.neu.edu

Jayashree Kalpathy-Cramer
Dept. of Radiology
MGH/Harvard Medical School
kalpathy@nmr.mgh.harvard.edu

Susan Ostmo
Dept. of Ophthalmology
Casey Eye Inst., OHSU
ostmo@ohsu.edu

J. Peter Campbell
Dept. of Ophthalmology
Casey Eye Inst., OHSU
campbelp@ohsu.edu

Michael F. Chiang
Dept. of Ophthalmology
Casey Eye Inst., OHSU
chiangm@ohsu.edu

Stratis Ioannidis
Dept. of ECE
Northeastern Univ.
ioannidis@ece.neu.edu

Abstract

We consider a ranking regression problem in which we use a dataset of ranked choices to learn Plackett-Luce scores as functions of sample features. We solve the maximum likelihood estimation problem by using the Alternating Directions Method of Multipliers (ADMM), effectively separating the learning of scores and model parameters. This separation allows us to express scores as the stationary distribution of a continuous-time Markov Chain. Using this equivalence, we propose two spectral algorithms for ranking regression that learn model parameters up to 579 times faster than the Newton's method.

1 Introduction

Learning from ranked choices has a long history in domains such as econometrics (McFadden, 1973; Ryzin and Mahajan, 1999), transportation (McFadden, 2000), psychometrics (Thurstone, 1927; Bradley and Terry, 1952), and sports (Elo, 1978), to name a few. The Plackett-Luce choice model (Plackett, 1975) is a popular parametric model used for inference in this setting: each sample is parametrized by a score and the probability that a sample is ranked higher than a set of alternatives is proportional to this score.

Plackett-Luce scores are traditionally learned from ranking observations via Maximum Likelihood Estimation (MLE) (Dykstra, 1960; Hunter, 2004; Hajek et al., 2014; Negahban et al., 2018); under a reparametrization, the negative log-likelihood is convex and Plackett-Luce scores can be estimated via, e.g., Newton's method (Nocedal and Wright, 2006). Nevertheless, for large datasets, Newton's method can be prohibitively slow (Hunter, 2004). Recently, Maystre and Grossglauser (2015) proposed a highly efficient iterative spectral method, termed Iterative Luce Spectral Ranking (ILSR), that estimates Plackett-Luce scores significantly faster than state-of-the-art methods. ILSR relies on the fact that ML estimates of Plackett-Luce scores constitute the stationary distribution of a Markov chain with transition rates defined by ranking observations.

The above approaches learn Plackett-Luce scores in the absence of sample features, which precludes rank predictions on samples outside the training set. A natural variant of the above setting is *ranking regression*, whereby Plackett-Luce scores are parametrized functions of sample features. This problem has received considerable attention in the literature, via both shallow (Joachims, 2002; Pahikkala et al., 2009; Tian et al., 2019) and deep models (Burges et al., 2005; Chang et al., 2016; Dubey et al., 2016; Han, 2018; Yıldız et al., 2019). Nevertheless, virtually all existing work on ranking regression relies on classic optimization methods for parameter inference. To the best of our knowledge, the opportunity to accelerate learning in ranking regression via spectral methods has not yet been explored.

We make the following contributions.

- We solve the ranking regression problem by using the Alternating Directions Method of Multipliers (ADMM) (Boyd et al., 2011) to perform MLE, effectively separating the learning of scores and model parameters. This separation allows us to express scores as the stationary distribution of a modified Markov Chain, and to devise spectral algorithms for ranking regression akin to ILSR.
- In particular, we propose two iterative algorithms, PLADMM and PLADMM-log, that jointly estimate

Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

model parameters and Plackett-Luce scores via a spectral method. Though the problems solved are non-convex, we establish conditions that yield convergence guarantees, as well as initializations tailored to the Plackett-Luce objective.
- Our algorithms yield significant performance dividends in terms of both speed and accuracy on synthetic and real-life datasets. PLADMM and PLADMM-log are up to 579 times faster than traditional optimization methods regressing Plackett-Luce scores from features, including Newton's method. Furthermore, for large datasets, PLADMM and PLADMM-log outperform feature-less methods, including ILSR, by 13% in maximal choice prediction accuracy and 9% in ranking prediction Kendall-Tau correlation.

From a technical standpoint, we show that the Plackett-Luce negative log-likelihood *augmented with a proximal penalty* has stationary points that satisfy the balance equations of a Markov chain (c.f. Thm 4.2). In turn, ADMM allows us to reduce ranking regression to a regularized MLE with precisely such a penalty. The remainder of this paper is organized as follows. We review related literature in Sec. 2. We formulate our problem in Sec. 3 and summarize ILSR. We describe our main contributions and proposed algorithms in Sec. 4. We present our experiments in Sec. 5 and conclude with future work in Sec. 6.

2 Related Work

The problem of *rank aggregation* (Dwork et al., 2001), in which a total ordering of samples is regressed from ranking observations, is classic; literature on the subject is vast—see, e.g., the surveys by Fligner and Verducci (1993), Cattelan (2012) and Marden (2014). Probabilistic inference in this setting typically assumes (a) that a "true" total ordering of samples exists, and (b) that ranking observations exhibit a *stochastic transitivity property* (Agarwal, 2016): a sample is more likely to be ranked higher than another when this event is consistent with the underlying total ordering.

The noisy permutation model is a non-parametric model for this setting: pairwise comparisons consistent with the underlying total ordering are observed under i.i.d. Bernoulli noise. Maximum likelihood estimation (MLE) is NP-hard in this setting. A polytime algorithm by Braverman and Mossel (2008) recovers the underlying ordering in $\Theta(n \log n)$ comparisons, w.h.p.; this is tightened by several recent works (Wauthier et al., 2013; Mao et al., 2017, 2018). The Mallows model (Mallows, 1957) assumes that the probability of a ranking observation is a decreasing function of its distance from the underlying total ordering, under appropriate notions of distance (e.g., Kendall-Tau); MLE can be approached, e.g., via EM (Lu and Boutilier, 2011). Shah et al. (2016a) learn the full matrix of pairwise comparison probabilities via a minimax optimal estimator requiring $\Theta(\log^2 n)$ comparisons. Rajkumar and Agarwal (2016) learn the matrix via matrix completion, requiring $\Theta(nr \log n)$ comparisons, where $r \ll n$ is the rank. Ammar and Shah (2011) assume that comparisons are sampled from an unknown distribution over total orderings and propose an entropy maximization algorithm requiring $\Theta(n^2)$ comparisons.

We focus on parametric models, as they are more natural in the context of regressing rankings from sample features. In both Plackett-Luce (Plackett, 1975) and Thurstone (Thurstone, 1927) each sample is parametrized by a score. In the Thurstone model, observations result from comparing scores after the addition of Gaussian noise. Vojnovic and Yun (2016) and Shah et al. (2016b) estimate Thurstone scores via MLE and provide sample complexity bounds that are inversely proportional to the smallest non-zero eigenvalue of the Laplacian of a graph modeling comparisons. In Plackett-Luce, the probability that a sample is chosen over a set of alternatives is proportional to its score. Hunter (2004) proposes a Minorization-Maximization (MM) approach to estimate Plackett-Luce scores via MLE, earlier used by Dykstra (1960) on pairwise comparisons (i.e., on the Bradley-Terry (BT) setting). Hajek et al. (2014) provide an upper bound on the error in estimating the Plackett-Luce scores via MLE and show that the latter is minimax-optimal. Negahban et al. (2018) propose a latent factor model, estimating parameters via a convex relaxation of the corresponding rank penalty and providing sample complexity guarantees. Assuming score priors, Guiver and Snelson (2009), Caron and Doucet (2012) and Azari et al. (2012) estimate Plackett-Luce scores via Bayesian inference.

Our focus on Plackett-Luce is due to the recent emergence of spectral algorithms for inference in this setting. Negahban et al. (2012) propose the Rank Centrality (RC) algorithm for the BT setting and derive a minimax error bound. Chen and Suh (2015) propose a spectral MLE algorithm extending RC with an additional stage that cyclically performs MLE for each score. Soufiani et al. (2013) and Jang et al. (2017) extend RC to rankings of two or more samples by breaking rankings into independent comparisons. Improved bounds, applying also to broader noise settings, are provided by Rajkumar and Agarwal (2014). Khetan and Oh (2016) generalize the work by Soufiani et al. (2013) by breaking rankings into independent shorter rankings, and building a hierarchy of tractable and consistent estimators. Blanchet et al. (2016) model sequential choices by state transitions in a Markov chain (MC), where transitions are functions of choice probabilities.

Bridging the above approaches with MLE, Maystre and

Grossglauser (2015) show that the MLE of Plackett-Luce scores can be expressed as the stationary distribution of an MC. Their proposed Iterative Luce Spectral Ranking (ILSR) algorithm estimates the Plackett-Luce scores faster than traditional optimization methods, such as, e.g., Hunter's (Hunter, 2004) and Newton's method, and more accurately than prior spectral rank aggregation methods. Ragain and Ugander (2016) show that a spectral approach applies even after relaxing the assumption that the relative order of any two samples is independent of the alternatives. Agarwal et al. (2018) propose another spectral method called accelerated spectral ranking by departing from the exact equivalence between MLE and MC approximation and demonstrate faster convergence than ILSR.

We depart from all aforementioned methods by regressing ranked choices from sample features. Closer to our work, RankSVM (Joachims, 2002) learns a target ranking from features via a linear Support Vector Machine (SVM), with constraints imposed by all possible comparisons. Pahikkala et al. (2009) propose a regularized least-squares based algorithm for learning to rank from comparisons. Several works learn comparisons from features via MLE over logistic BT models (Guo et al., 2018; Tian et al., 2019); deeper models have also been considered (Burges et al., 2005; Chang et al., 2016; Dubey et al., 2016; Han, 2018; Yıldız et al., 2019). Niranjan and Rajkumar (2017) assume that features are low-dimensional and use matrix completion to recover the BT scores. Saha and Rajkumar (2018) propose a least squares based algorithm called f-BTL to regress the BT scores; we adjust this to initalize our algorithm. To the best of our knowledge, we are the first to use a spectral method akin to ILSR to (a) regress Plackett-Luce scores from features, and (b) to establish a significant speedup over prior art.

3 Problem Formulation

Plackett-Luce Model. We consider a dataset of n samples indexed by $i \in \mathcal{N} \equiv \{1, \dots, n\}$. Every sample $i \in \mathcal{N}$ has a corresponding p-dimensional feature vector $\boldsymbol{x}_i \in \mathbb{R}^p$. There exists an underlying total ordering of these n samples. A labeler of this dataset acts as a (possibly noisy) oracle revealing this total ordering: when presented with a query $A \subseteq \mathcal{N}$, i.e., a set of alternative samples, the noisy labeler chooses the *maximal* sample in A w.r.t. the underlying total ordering.

Formally, our "labeled" dataset $\mathcal{D} = \{(c_\ell, A_\ell) \,|\, \ell \in \mathcal{M} = \{1, ..., M\}\}$ consists of M *observations* (c_ℓ, A_ℓ), $\ell \in \mathcal{M}$, where $A_\ell \subseteq \mathcal{N}$ is the ℓ-th *query* submitted to the labeler and $c_\ell \in A_\ell$ is her respective ℓ-th *maximal choice* (i.e., the label). We tackle the problem of regressing such choices c_ℓ from the features $\boldsymbol{x}_i$ of the samples $i \in A_\ell$. To do so, we assume that choices are governed by the Plackett-Luce model (Plackett, 1975). The model asserts that every sample $i \in \mathcal{N}$ is associated with a non-negative deterministic score $\pi_i \in \mathbb{R}_+$. Given scores $\boldsymbol{\pi} = [\pi_i]_{i \in \mathcal{N}} \in \mathbb{R}_+^n$, then (a) observations $(c_\ell, A_\ell), \ell \in \mathcal{M}$ are independent, and (b) given query A_ℓ,

$$\mathbf{P}(c_\ell \,|\, A_\ell, \boldsymbol{\pi}) = \pi_{c_\ell} / \textstyle\sum_{j \in A_\ell} \pi_j = \pi_\ell / \sum_{j \in A_\ell} \pi_j. \quad (1)$$

Abusing notation, we write the score of the chosen sample as $\pi_\ell \equiv \pi_{c_\ell}$. Note that $\mathbf{P}(c_\ell \,|\, A_\ell, \boldsymbol{\pi}) = \mathbf{P}(c_\ell \,|\, A_\ell, s\boldsymbol{\pi})$, for all $s > 0$; thus, w.l.o.g., we may additionally assume (or enforce via rescaling) that Plackett-Luce scores satisfy $\mathbf{1}^\top \boldsymbol{\pi} = 1$.

Plackett-Luce also applies to *ranking* data. In the ranking setting, when presented with a query $A_\ell \subseteq \mathcal{N}$, the labeler ranks the samples in A_ℓ into an ordered sequence $\alpha_1^\ell \succ \alpha_2^\ell \succ \cdots \succ \alpha_{|A_\ell|}^\ell$. Under the Plackett-Luce model, this ranking is expressed as $|A_\ell| - 1$ maximal choice queries: α_1^ℓ over A_ℓ, α_2^ℓ over $A_\ell \setminus \{\alpha_1^\ell\}$, etc., so that:

$$\mathbf{P}(\alpha_1^\ell \succ \alpha_2^\ell \succ \cdots \succ \alpha_{|A_\ell|}^\ell \,|\, A_\ell, \boldsymbol{\pi}) = \prod_{t=1}^{|A_\ell|-1} \big(\pi_{\alpha_t^\ell} / \sum_{s=t}^{|A_\ell|} \pi_{\alpha_s^\ell}\big). \quad (2)$$

The product form of (2) implies that rankings of a query A_ℓ can be converted to $|A_\ell| - 1$ maximal-choice observations, each governed by (1), that have the same joint probability: ranking $(\alpha_1^\ell \succ \alpha_2^\ell \succ \cdots \succ \alpha_{|A_\ell|}^\ell)$ can be seen as the outcome of α_1^ℓ being chosen as the top within the query set A_ℓ, α_2^ℓ being the top among $A_\ell \setminus \{\alpha_1^\ell\}$, etc. Keeping this reduction from ranking to maximal-choice datasets in mind, we focus on the latter in our exposition below.

Parameter Inference and Regression. Given observations $\mathcal{D}$, Maximum Likelihood Estimation (MLE) of the Plackett-Luce scores $\boldsymbol{\pi} \in \mathbb{R}_+^n$ amounts to minimizing the negative log-likelihood:

$$\mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) \equiv \textstyle\sum_{\ell=1}^{M} \big(\log \sum_{j \in A_\ell} \pi_j - \log\, \pi_\ell\big). \quad (3)$$

To regress scores $\boldsymbol{\pi}$ from sample features $\boldsymbol{x}_i$, $i \in \mathcal{N}$, we consider two cases:

Affine Case. We assume that there exist $\boldsymbol{\beta} \in \mathbb{R}^p$ and $b \in \mathbb{R}$ such that $\boldsymbol{\pi} = \boldsymbol{\pi}_{\texttt{AFF}}(\boldsymbol{\beta}, b; \boldsymbol{X}) \equiv \boldsymbol{X}\boldsymbol{\beta} + b\mathbf{1}$. Then, MLE of parameters $(\boldsymbol{\beta}, b) \in \mathbb{R}^{p+1}$ amounts to solving:

$$\min_{(\boldsymbol{\beta}, b): \boldsymbol{\pi}_{\texttt{AFF}}(\boldsymbol{\beta}, b; \boldsymbol{X}) \geq \mathbf{0}} \mathcal{L}\left(\mathcal{D} \,|\, \boldsymbol{\pi}_{\texttt{AFF}}(\boldsymbol{\beta}, b; \boldsymbol{X})\right), \quad (4)$$

where $\mathcal{L}$ is given by (3), and $\boldsymbol{X} = [\boldsymbol{x}_1, .., \boldsymbol{x}_n]^T \in \mathbb{R}^{n \times p}$. We note that Problem (4) is *not* convex, as the objective is not convex in $(\boldsymbol{\beta}, b) \in \mathbb{R}^{p+1}$ (c.f. Appendix A).

Logistic Case. In the logistic case, we assume that there exists $\boldsymbol{\beta} \in \mathbb{R}^p$ s.t. $\boldsymbol{\pi} = \boldsymbol{\pi}_{\texttt{LOG}}(\boldsymbol{\beta}; \boldsymbol{X}) \equiv [e^{\boldsymbol{\beta}^\top \boldsymbol{x}_i}]_{i \in \mathcal{N}}$. As $\boldsymbol{\pi}_{\texttt{LOG}}(\boldsymbol{\beta}; \boldsymbol{X}) \geq 0$ by definition, MLE corresponds to:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}_{\texttt{LOG}}(\boldsymbol{\beta}; \boldsymbol{X})). \quad (5)$$

The objective of (5) is convex in $\boldsymbol{\beta}$ (c.f. Appendix A), so an optimal solution can be found via, e.g., Newton's method (Nocedal and Wright, 2006).

Plackett-Luce Without Features and a Spectral Method. We wish to construct highly efficient algorithms for solving regression problems (4) and (5). To do so, we first briefly review the state of the art for learning the scores $\boldsymbol{\pi}$ in the absence of features. In this case, MLE amounts to:

$$\min_{\boldsymbol{\pi} \in \mathbb{R}^n_+} \mathcal{L}(\mathcal{D} \mid \boldsymbol{\pi}). \tag{6}$$

As is the case for (5), reparametrizing the scores as $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$ makes the negative log-likelihood $\mathcal{L}$ convex in $\boldsymbol{\theta} = [\theta_i]_{i \in \mathcal{N}}$, which in turn enables computing the Plackett-Luce scores via Newton's method. Nevertheless, Newton's method can be prohibitively slow for large n and M (Hunter, 2004). Recently, Maystre and Grossglauser (2015) proposed a novel spectral algorithm that is significantly faster than Newton's method. Their algorithm relies on the following theorem which, for completeness, we re-prove in Appendix B:

Theorem 3.1 (Maystre and Grossglauser (2015))**.** *An optimal solution $\boldsymbol{\pi} \in \mathbb{R}^n_+$ to (6) satisfies:*

$$\textstyle\sum_{j \neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) = \sum_{j \neq i} \pi_i \lambda_{ij}(\boldsymbol{\pi}), \quad \text{for all } i \in \mathcal{N}, \tag{7}$$

where, for all $i, j \in \mathcal{N}$, with $i \neq j$,

$$\lambda_{ji}(\boldsymbol{\pi}) = \textstyle\sum_{\ell \in W_i \cap L_j} \left(\sum_{t \in A_\ell} \pi_t \right)^{-1} \geq 0, \tag{8}$$

for $W_i = \{\ell \,|i \in A_\ell, c_\ell = i\}$ the observations where sample $i \in \mathcal{N}$ is chosen and $L_i = \{\ell \,|i \in A_\ell, c_\ell \neq i\}$ the observations where sample $i \in \mathcal{N}$ is not chosen.

Eq. (7) are the *balance equations* of a continuous-time Markov Chain (MC) with transition rates:

$$\boldsymbol{\Lambda}(\boldsymbol{\pi}) = [\lambda_{ji}(\boldsymbol{\pi})]_{i,j \in \mathcal{N}}, \tag{9}$$

where $\lambda_{ji}(\boldsymbol{\pi})$ are given by Eq. (8). Hence, $\boldsymbol{\pi}$ is the stationary distribution of the MC defined by transition rates $\boldsymbol{\Lambda}(\boldsymbol{\pi})$ (Gallager, 2013). Let $\text{ssd}(\boldsymbol{\Lambda})$ be the stationary distribution of an MC with transition rates $\boldsymbol{\Lambda}$. When matrix $\boldsymbol{\Lambda}$ is fixed (i.e., the transition rates are known), the vector $\text{ssd}(\boldsymbol{\Lambda})$ is a solution to the linear system defined by the balance equations (7) and $\mathbf{1}^\top \boldsymbol{\pi} = 1$, as it is a distribution.[1] If (9) is irreducible, the linear system has a unique solution $\boldsymbol{\pi} > \mathbf{0}$ (Gallager, 2013).

However, the transition matrix $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}(\boldsymbol{\pi})$ in Theorem 3.1 *is itself a function of $\boldsymbol{\pi}$, and is therefore a priori unknown.* Maystre and Grossglauser (2015) find $\boldsymbol{\pi}$ through an iterative algorithm. Starting from the uniform distribution $\boldsymbol{\pi}^0 = \frac{1}{n}\mathbf{1}$, they compute:

$$\boldsymbol{\pi}^{l+1} = \text{ssd}\left(\boldsymbol{\Lambda}(\boldsymbol{\pi}^l)\right), \quad \text{for } l = 0, 1, 2, \ldots, \tag{10}$$

where $\boldsymbol{\Lambda}(\cdot)$ is given by (8), (9). Maystre and Grossglauser (2015) refer to Eq. (10) as the *Iterative Luce Spectral Ranking* (ILSR) algorithm. They also establish that (10) converges to an optimal solution of (6) under mild assumptions. Most importantly, as mentioned above, ILSR significantly outperforms state-of-the-art MLE algorithms in computational efficiency.

4 Plackett-Luce ADMM (PLADMM) Algorithm

Given ILSR's significant computational benefits, we wish to develop analogues in the regression setting. In contrast to the feature-less setting, it is not a priori evident how to solve Problems (4) and (5) via a spectral approach. Taking the affine case as an example, and momentarily ignoring issues of non-convexity, the stationary points of the Lagrangian of the optimization problem (4) *cannot be expressed via the balance equations of an MC.* Our main contribution is to circumvent this problem by using the Alternating Directions Method of Multipliers (ADMM) (Boyd et al., 2011). Intuitively, ADMM allows us to decouple the optimization of scores $\boldsymbol{\pi}$ from model parameters $\boldsymbol{\beta}$ and b, encapsulating them in a quadratic penalty: the latter becomes amenable to a spectral approach after a series of manipulations that we outline below (see Thm. 4.2). We focus here on the affine case, extending our method to the logistic case in Appendix E.

An ADMM Approach. We rewrite Problem (4) as:

$$\begin{aligned} \text{Minimize} \quad & \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) && \text{(11a)} \\ \text{subject to:} \quad & \boldsymbol{\pi} = \boldsymbol{X}\boldsymbol{\beta} + b\mathbf{1}, \quad \boldsymbol{\pi} \geq \mathbf{0}. && \text{(11b)} \end{aligned}$$

To simplify our notation, we introduce $\tilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}, b) \in \mathbb{R}^{p+1}$ and $\tilde{\boldsymbol{X}} = [\boldsymbol{X}|\mathbf{1}] \in \mathbb{R}^{n \times (p+1)}$, so that $\boldsymbol{\pi} = \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}$. ADMM solves (11) by minimizing the following *augmented Lagrangian*:

$$\begin{aligned} L_\rho(\tilde{\boldsymbol{\beta}}, \boldsymbol{\pi}, \boldsymbol{y}) = \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) + \boldsymbol{y}^T(\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}) \\ + \frac{\rho}{2} \,\|\, \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi} \,\|_2^2, \end{aligned} \tag{12}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ is a dual variable corresponding to the equality constraints in Eq. (11) and $\rho > 0$ is a penalty parameter. ADMM alternates between optimizing $\tilde{\boldsymbol{\beta}}$ and $\boldsymbol{\pi}$, thereby decoupling these two variables. Using a rescaling $\boldsymbol{u} = \frac{1}{\rho}\boldsymbol{y} \in \mathbb{R}^n$ for convenience, applying ADMM on problem (11) yields the following iterative algorithm (see Appendix C for a detailed derivation):

$$\tilde{\boldsymbol{\beta}}^{k+1} = \underset{\tilde{\boldsymbol{\beta}} \in \mathbb{R}^{p+1}}{\arg\min} \,\|\, \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}^k + \boldsymbol{u}^k \,\|_2^2, \tag{13a}$$

[1] In practice, $\text{ssd}(\boldsymbol{\Lambda})$ can be computed by uniformizing $\boldsymbol{\Lambda}$, i.e., increasing self-transition rates until all states have the same outgoing rate, and finding the leading left eigenvector via, e.g., the power method (Lei et al., 2016).

$$\boldsymbol{\pi}^{k+1}=\underset{\boldsymbol{\pi}\in\mathbb{R}^n_+}{\arg\min}\big(\mathcal{L}(\mathcal{D}|\boldsymbol{\pi})+\tfrac{\rho}{2}\|\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1}-\boldsymbol{\pi}+\boldsymbol{u}^k\|_2^2\big), \quad (13b)$$

$$\boldsymbol{u}^{k+1}=\boldsymbol{u}^k+\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1}-\boldsymbol{\pi}^{k+1}. \quad (13c)$$

This has the following immediate computational advantages. First, step (13a) is a quadratic minimization and admits a closed form solution. Crucially, step (13b) is amenable to a spectral approach, though the corresponding MC is not as apparent as in ILSR; we outline its construction below.

An MC for Step (13b)**.** We first establish the following auxiliary lemma, proved in Appendix D.1.

Lemma 4.1. *Given $\tilde{\boldsymbol{\beta}}^{k+1}\in\mathbb{R}^{p+1}$, $\boldsymbol{u}^k\in\mathbb{R}^n$, let $\boldsymbol{\pi}\in\mathbb{R}^n_+$ be such that:*

$$\nabla_{\boldsymbol{\pi}}L_\rho(\tilde{\boldsymbol{\beta}}^{k+1},\boldsymbol{\pi},\boldsymbol{u}^k)=\mathbf{0}. \quad (14)$$

For $\boldsymbol{\sigma}=[\sigma_i]_{i\in\mathcal{N}}\equiv\rho(\boldsymbol{\pi}-\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1}-\boldsymbol{u}^k)$, and $[\lambda_{ij}(\boldsymbol{\pi})]_{i,j\in\mathcal{N}}$ given by (8)*,* (14) *is equivalent to:*

$$\textstyle\sum_{j\neq i}\pi_j\lambda_{ji}(\boldsymbol{\pi})-\sum_{j\neq i}\pi_i\lambda_{ij}(\boldsymbol{\pi})=\pi_i\sigma_i, \quad (15)$$

for all $i\in\mathcal{N}$.

Although Eq. (15) looks similar to Eq. (7), it is not evident that it corresponds to the balance equations of an MC as, in general, $\boldsymbol{\sigma}\neq\mathbf{0}$. Nevertheless, we prove that this is indeed the case:

Theorem 4.2. *Eq.* (15) *are the balance equations of a continuous-time MC with transition rates:*

$$\mu_{ji}(\boldsymbol{\pi})=\begin{cases}\lambda_{ji}(\boldsymbol{\pi})+\frac{2\pi_i\sigma_i\sigma_j}{\sum_{t\in\mathcal{N}_-}\pi_t\sigma_t-\sum_{t\in\mathcal{N}_+}\pi_t\sigma_t} & \\ \qquad\qquad \textit{if } j\in\mathcal{N}_+ \textit{ and } i\in\mathcal{N}_- & \\ \lambda_{ji}(\boldsymbol{\pi}) \quad \textit{otherwise,} & \end{cases} \quad (16)$$

where $\boldsymbol{\sigma}=[\sigma_i]_{i\in\mathcal{N}}\equiv\rho(\boldsymbol{\pi}-\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1}-\boldsymbol{u}^k)$, $[\lambda_{ij}(\boldsymbol{\pi})]_{i,j\in\mathcal{N}}$ are given by (8)*, and $(\mathcal{N}_+,\mathcal{N}_-)$ is a partition of $\mathcal{N}$ such that $\sigma_i\geq 0$ for all $i\in\mathcal{N}_+$ and $\sigma_i<0$ for all $i\in\mathcal{N}_-$.*

The proof is in App. D.2. By Lemma 4.1 and Theorem 4.2, we conclude that a stationary $\boldsymbol{\pi}\in\mathbb{R}^n_+$ satisfying (14) is also the stationary distribution of the continuous-time MC with transition rates:

$$\boldsymbol{M}(\boldsymbol{\pi})=[\mu_{ji}(\boldsymbol{\pi})]_{i,j\in\mathcal{N}}, \quad (17)$$

where $\mu_{ji}(\boldsymbol{\pi})$ are given by Eq. (16). Motivated by these observations, and mirroring ILSR (Eq. (10)), we compute a solution to (13b) via:

$$\boldsymbol{\pi}^{l+1}=\operatorname{ssd}\left(\boldsymbol{M}(\boldsymbol{\pi}^l)\right), \quad \text{for } l=0,1,2,\ldots, \quad (18)$$

where $\boldsymbol{M}(\cdot)$ is given by Eq. (17). We refer to this procedure as ILSRX ("ILSR with features").

Overall Algorithm. Putting everything together, our Plackett-Luce ADMM (PLADMM) solving Eq. (11) is summarized in Algorithm 1. We iteratively update $\tilde{\boldsymbol{\beta}}$, $\boldsymbol{\pi}$, and $\boldsymbol{u}$ via Eq. (13) until convergence, with $\tilde{\boldsymbol{\beta}}$ updated via Eq. (13a) and $\boldsymbol{\pi}$ updated via ILSRX (Eq. (18)). At iteration k, we initialize ILSRX with $\boldsymbol{\pi}^{k-1}$. We note that, as Problem (11) is non-convex, selecting a good initialization point is important in practice. We discuss initialization, additional computational issues, and theoretical guarantees below.

Algorithm 1 PLADMM

procedure ADMM($\tilde{\boldsymbol{X}}$, $\mathcal{D}=\{(c_\ell,A_\ell)\,|\,\ell\in\mathcal{M}\}$, ρ)
  Initialize $\tilde{\boldsymbol{\beta}}$ via Eq. (20); $\boldsymbol{\pi}\leftarrow\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}$; $\boldsymbol{u}\leftarrow\mathbf{0}$
  repeat
    $\boldsymbol{\pi}\leftarrow$ ILSRX($\rho,\boldsymbol{\pi},\tilde{\boldsymbol{X}},\tilde{\boldsymbol{\beta}},\boldsymbol{u}$)
    $\boldsymbol{u}\leftarrow\boldsymbol{u}+\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}-\boldsymbol{\pi}$
    $\tilde{\boldsymbol{\beta}}\leftarrow(\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^T(\boldsymbol{\pi}-\boldsymbol{u})$
  until convergence
return $\tilde{\boldsymbol{\beta}}$, $\boldsymbol{\pi}$
end procedure
procedure ILSRX($\rho,\boldsymbol{\pi},\tilde{\boldsymbol{X}},\tilde{\boldsymbol{\beta}},\boldsymbol{u}$)
  repeat
    $\boldsymbol{\sigma}\leftarrow\rho(\boldsymbol{\pi}-\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}-\boldsymbol{u})$
    Calculate $\boldsymbol{M}(\boldsymbol{\pi})=[\mu_{ji}(\boldsymbol{\pi})]_{i,j\in\mathcal{N}}$ via Eq. (16)
    $\boldsymbol{\pi}\leftarrow\operatorname{ssd}(\boldsymbol{M}(\boldsymbol{\pi}))$
  until convergence
return $\boldsymbol{\pi}$
end procedure

Initialization. We initialize $\tilde{\boldsymbol{\beta}}$ so that $\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}$ is a good approximation of Plackett-Luce scores. We use a technique akin to Saha and Rajkumar (2018), applied to our affine setting. Given a distribution over queries $A\subseteq\mathcal{N}$, let $P_{ij}=\mathbb{E}_A[c=i|\{i,j\}\subseteq A]$ be the probability that i is chosen given a query A that contains both i and j. By (1), for $i,j\in\mathcal{N}$, $\frac{P_{ij}}{P_{ji}}=\frac{\pi_i}{\pi_j}=\frac{\boldsymbol{x}_i^\top\boldsymbol{\beta}+b}{\boldsymbol{x}_j^\top\boldsymbol{\beta}+b}$, or:

$$\delta_{ij}(\tilde{\boldsymbol{\beta}})\equiv(P_{ij}\boldsymbol{x}_j-P_{ji}\boldsymbol{x}_i)^\top\boldsymbol{\beta}+(P_{ij}-P_{ji})b=0. \quad (19)$$

Motivated by (19), we estimate P_{ij} empirically from $\mathcal{D}$, and obtain our initialization $\tilde{\boldsymbol{\beta}}^0=(\boldsymbol{\beta}^0,b^0)\in\mathbb{R}^{p+1}$ by solving (19) in the least-square sense; that is,

$$\tilde{\boldsymbol{\beta}}^0=\underset{\tilde{\boldsymbol{\beta}}\in\mathbb{R}^{p+1}:\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}\geq\mathbf{0}\wedge\mathbf{1}^\top\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}=1}{\arg\min}\textstyle\sum_{i,j}\delta_{ij}^2(\tilde{\boldsymbol{\beta}}). \quad (20)$$

Note that this is a convex quadratic program. Finally, we also set the initial dual variable as $\boldsymbol{u}^0=\mathbf{0}$.

Computational Complexity. Each iteration of PLADMM involves the three steps in Eq. (13). One iteration of ILSRX is $O(\sum_{\ell\in\mathcal{D}}|A_\ell|+n^2)$ for constructing the transition matrix $\boldsymbol{M}(\boldsymbol{\pi})$ via Eq.(17) and for finding the stationary distribution $\boldsymbol{\pi}$ via, e.g., a power method (Lei et al., 2016), respectively. Updates of $\boldsymbol{u}$ and $\tilde{\boldsymbol{\beta}}$ are both $O\big(n(p+1)\big)$ as matrix-vector multiplications, since the matrix $(\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^T$ can be precomputed.

Theoretical Guarantees. In general, condition (14) is *not* sufficient for optimality w.r.t. step (13b). To show this, we require the following technical assumption:

Assumption 4.1. *For $\{\boldsymbol{\pi}^k\}_{k\in\mathbb{N}}$, given by* (13b)*, there exists an $\epsilon>0$ such that $\pi_i^k>\epsilon$ for all $i\in\mathcal{N}$ and $k\in\mathbb{N}$.*

Spec.	Dataset			
	ROP	FAC	Pairwise Sushi	Triplet Sushi
n	100	1000	100	100
p	143	50	18	18
M	29705	728	450	1200
$\lvert A_\ell \rvert$	2	2	2	3
$n_{\texttt{fold}}$	10	10	3	10
Type	Choice	Choice	Choice	Ranking

Table 1: No. of samples (n), no. of parameters (p), no. of observations (M), query size ($|A_\ell|$), no. of cross validation folds ($n_{\texttt{fold}}$), and type of observations for real data

Under this assumption, we show that stationarity implies optimality w.r.t. (13b) for large enough ρ:

Theorem 4.3. *Under Assumption 4.1, for* $\rho \geq \frac{2}{\epsilon^2}\max_i \sum_{\ell | i \in A_\ell} \frac{1}{|A_\ell|^2}$, *a* $\boldsymbol{\pi} > \mathbf{0}$ *satisfying condition* (14) *is a minimizer of* (13b).

The proof is in Appendix D.3. Moreover, although problem (11) is non-convex, we establish the following convergence guarantee for the ADMM steps (13). We provide the proof in Appendix D.4.

Theorem 4.4. *Suppose that there exists* $\kappa > 0$ *such that* $\tilde{\boldsymbol{X}}^T \tilde{\boldsymbol{X}} \succeq \kappa \boldsymbol{I}$ *and the sequence* $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^{k+1})\}_{k\in\mathbb{N}}$ *generated by* (13) *is bounded. Then, under Assumption 4.1, for* $\rho > \frac{2\max_i |W_i|}{\epsilon^2}$ *where* W_i *is defined in Theorem 3.1, the sequence* $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^{k+1})\}_{k\in\mathbb{N}}$ *generated by* (13) *converges to a point that satisfies the Karush-Kuhn-Tucker (KKT) conditions of* (11).

5 Experiments

Experiment Setup. We evaluate PLADMM and PLADMM-log (the spectral algorithm for the logistic case, described in Appendix E) on synthetic and real-life datasets, summarized in Table 1. Additional details on our datasets are in Appendix F.1. We perform 10-fold cross validation (CV) for each dataset, except Pairwise Sushi, for which we use 3 folds. For synthetic datasets, we also repeat experiments over 5 random generations. We partition each dataset into training and test sets in two ways. In *observation CV*, we partition the dataset w.r.t. *observations* $\mathcal{M}$, using 90% of the M observations for training and the remaining 10% for testing. In *sample CV*, we partition samples $\mathcal{N}$, using 90% of the n samples for training and the remaining 10% for testing. When partitioning w.r.t. samples, *observations containing samples from both training and test partitions are discarded.* As the Pairwise Sushi dataset contains few observations (c.f. Table 1), we perform 3-fold cross validation in this case.

We implement[2] seven inference algorithms described in detail in Appendix F.2. Four are *feature methods*, i.e., algorithms that regress Plackett-Luce scores from features: PLADMM described in Algorithm 1, PLADMM-log described in Appendix E, sequential least-squares quadratic programming (SLSQP), that solves (4), and Newton on $\boldsymbol{\beta}$, that solves the convex problem (5) via Newton's method. The remaining three are *featureless methods*, i.e., algorithms that learn the Plackett-Luce scores from the choice observations alone: Iterative Luce Spectral Ranking (ILSR) described by Eq.(10), the Minorization-Maximization (MM) algorithm (Hunter, 2004), and Newton on $\boldsymbol{\theta}$ that solves Eq. (6) via the reparametrization $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$ using Newton's method on $\boldsymbol{\theta} = [\theta_i]_{i\in\mathcal{N}}$.

Performance Metrics. We run each algorithm until convergence (see App. F.2 for criteria). We measure the elapsed time, including time spent in initialization, in seconds (Time) and the number of iterations (Iter). We measure the prediction performance by Top-1 accuracy (Top-1 Acc.) and Kendall-Tau correlation (KT) on the test set; formulas are provided in App. F.3. For synthetic datasets, we also measure the quality of convergence by the norm of the difference between estimated and true Plackett-Luce scores ($\triangle\boldsymbol{\pi}$); lower values indicate better estimation. We report averages and standard deviations over folds.

Execution Environment. For Tables 2 - 4, we measure timing on an Intel Xeon CPU E5-2680v2 2.8GHz with 128GB RAM. Particularly for experiments on larger synthetic datasets (c.f. Figures 1 - 2), we use an Intel Xeon CPU E5-2680v4 2.4GHz with 500GB RAM.

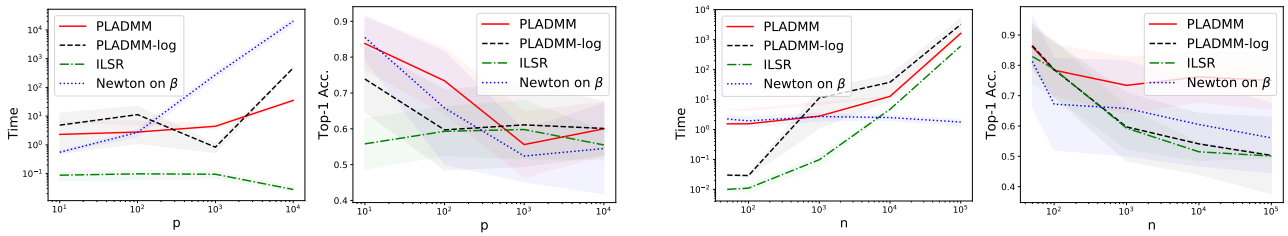
Sample CV. We begin with the experiments on sample CV, in which we partition samples $\mathcal{N}$ into training and test sets. Table 2 shows the evaluations of all algorithms trained on a synthetic dataset with $n = 1000$ samples, $p = 100$ features, $M = 1000$ observations, and query size $|A_\ell| = 2$. PLADMM and PLADMM-log converge $4-27$ times faster than other feature methods, i.e., Newton on $\boldsymbol{\beta}$ and SLSQP. Recall that in sample CV partitioning, training observations contain *only* training samples: test samples do not participate in any of the training observations. Thus, featureless methods ILSR, MM, and Newton on $\boldsymbol{\theta}$ *are no better than random predictors*, with 0.5 Top-1 Acc. and 0.0 KT. By regressing the Plackett-Luce scores from features, PLADMM and PLADMM-log significantly outperform the predictions of ILSR, MM, and Newton on $\boldsymbol{\theta}$, by $16\% - 33\%$ Top-1 Acc. and $16\% - 30\%$ KT.

Real datasets. We observe an equally significant speed gain on real datasets; Table 3 shows the evaluations on four real datasets partitioned w.r.t. sample CV. PLADMM and PLADMM-log are $3 - 18$ times faster than Newton on $\boldsymbol{\beta}$ and SLSQP. This speed gain is fundamentally due to the smaller per iteration complexity of PLADMM and PLADMM-log (c.f. Section 4). For instance, compared to Newton on $\boldsymbol{\beta}$, PLADMM-log re-

[2] Our code is publicly available at https://github.com/neu-spiral/FastAndAccurateRankingRegression

Partitioning	Method	Training Metrics			Performance Metrics on the Test Set	
		Time (s) ↓	Iter. ↓	$\triangle\boldsymbol{\pi}$ ↓	Top-1 Acc. ↑	KT ↑
Sample CV	**PLADMM**	0.237 ± 0.006	4 ± 0	0.717 ± 0.207	0.831 ± 0.119	0.609 ± 0.247
	PLADMM-log	1.428 ± 2.595	49 ± 79	0.845 ± 0.204	0.668 ± 0.159	0.335 ± 0.318
	ILSR (no $\boldsymbol{X}$)	0.045 ± 0.002	2 ± 0	0.718 ± 0.207	0.5 ± 0.0	-1.0 ± 0.0
	MM (no $\boldsymbol{X}$)	9.728 ± 0.487	500 ± 0	1.2 ± 0.1	0.5 ± 0.0	0.0 ± 0.0
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	4.537 ± 0.729	14 ± 3	1.236 ± 0.132	0.5 ± 0.0	-0.08 ± 0.272
	Newton on $\boldsymbol{\beta}$	6.406 ± 2.104	14 ± 5	0.808 ± 0.462	0.844 ± 0.148	0.688 ± 0.296
	SLSQP	43.908 ± 24.469	229 ± 132	0.718 ± 0.206	0.796 ± 0.106	0.592 ± 0.211
Observation CV	**PLADMM**	0.48 ± 0.24	4 ± 0	0.717 ± 0.207	0.837 ± 0.037	0.569 ± 0.072
	PLADMM-log	1.58 ± 2.027	29 ± 14	0.883 ± 0.208	0.699 ± 0.066	0.398 ± 0.132
	ILSR (no $\boldsymbol{X}$)	0.098 ± 0.056	2 ± 0	0.718 ± 0.208	0.708 ± 0.045	0.389 ± 0.088
	MM (no $\boldsymbol{X}$)	11.302 ± 0.515	500 ± 0	0.864 ± 0.19	0.685 ± 0.037	0.354 ± 0.074
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	8.218 ± 1.782	14 ± 3	1.244 ± 0.121	0.506 ± 0.029	0.01 ± 0.05
	Newton on $\boldsymbol{\beta}$	7.696 ± 2.35	14 ± 4	0.804 ± 0.463	0.871 ± 0.087	0.742 ± 0.173
	SLSQP	47.824 ± 28.585	219 ± 138	0.718 ± 0.206	0.819 ± 0.035	0.637 ± 0.07

Table 2: Evaluations on a synthetic dataset with $n = 1000$, $p = 100$, and $M = 1000$, partitioned w.r.t. sample CV and observation CV. We report the convergence time (Time), number of iterations until convergence (Iter), norm error in estimating true Plackett-Luce scores ($\triangle\boldsymbol{\pi}$), top-1 accuracy on the test set (Top-1 Acc.), and Kendall-Tau correlation on the test set (KT). ILSR, MM, and Newton on $\boldsymbol{\theta}$ do not use the features $\boldsymbol{X}$. Newton on $\boldsymbol{\beta}$ and SLSQP regress $\boldsymbol{\pi}$ from $\boldsymbol{X}$.



(a) Time and Top-1 Acc. vs. p

(b) Time and Top-1 Acc. vs. n

Figure 1: Convergence time (Time) and top-1 test accuracy (Top-1 Acc.) vs. n and p for PLADMM, PLADMM-log, ILSR, and Newton on $\boldsymbol{\beta}$. Evaluations are on synthetic datasets containing $M = 250$ observations partitioned w.r.t. observation CV. Number of samples changes in $n \in \{50, 100, 1000, 10000, 100000\}$ when number of features is $p = 100$, and number of features changes in $p \in \{10, 100, 1000, 10000\}$ when number of samples is $n = 1000$.

Dataset	Method	Training Metrics		Performance Metrics on the Test Set	
		Time (s) ↓	Iter. ↓	Top-1 Acc. ↑	KT ↑
FAC	**PLADMM**	0.301 ± 0.048	4 ± 0	0.654 ± 0.237	0.307 ± 0.473
	PLADMM-log	0.298 ± 0.466	10 ± 15	0.685 ± 0.237	0.369 ± 0.474
	ILSR (no $\boldsymbol{X}$)	0.059 ± 0.016	2 ± 0	0.5 ± 0.0	-1.0 ± 0.0
	MM (no $\boldsymbol{X}$)	5.905 ± 0.282	500 ± 0	0.5 ± 0.0	0.0 ± 0.0
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	7.604 ± 0.805	18 ± 2	0.5 ± 0.0	-0.4 ± 0.49
	Newton on $\boldsymbol{\beta}$	0.859 ± 0.077	6 ± 1	0.67 ± 0.17	0.34 ± 0.339
	SLSQP	14.332 ± 5.684	178 ± 67	0.675 ± 0.147	0.349 ± 0.293
ROP	**PLADMM**	1.708 ± 0.166	4 ± 0	0.783 ± 0.03	0.565 ± 0.06
	PLADMM-log	0.325 ± 0.028	1 ± 0	0.724 ± 0.105	0.448 ± 0.209
	ILSR (no $\boldsymbol{X}$)	0.649 ± 0.053	2 ± 0	0.5 ± 0.0	-1.0 ± 0.0
	MM (no $\boldsymbol{X}$)	0.001 ± 0.001	1 ± 0	0.5 ± 0.0	-1.0 ± 0.0
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	68.924 ± 5.521	8 ± 0	0.497 ± 0.012	-0.988 ± 0.036
	Newton on $\boldsymbol{\beta}$	47.563 ± 8.342	2 ± 1	0.552 ± 0.048	0.103 ± 0.096
	SLSQP	4.823 ± 4.914	2 ± 1	0.769 ± 0.052	0.538 ± 0.104
Pairwise Sushi	**PLADMM**	0.046 ± 0.01	4 ± 0	0.451 ± 0.082	-0.09 ± 0.177
	PLADMM-log	0.141 ± 0.025	27 ± 13	0.532 ± 0.076	0.064 ± 0.152
	ILSR (no $\boldsymbol{X}$)	0.014 ± 0.006	2 ± 0	0.5 ± 0.0	-1.0 ± 0.0
	MM (no $\boldsymbol{X}$)	1.513 ± 0.587	352 ± 183	0.5 ± 0.0	-1.0 ± 0.0
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	1.282 ± 0.924	18 ± 9	0.5 ± 0.0	-0.666 ± 0.472
	Newton on $\boldsymbol{\beta}$	0.21 ± 0.115	4 ± 2	0.665 ± 0.035	0.33 ± 0.069
	SLSQP	4.619 ± 6.321	168 ± 235	0.624 ± 0.065	0.248 ± 0.13
Triplet Sushi	**PLADMM**	0.091 ± 0.02	4 ± 0	0.358 ± 0.805	-0.333 ± 0.924
	PLADMM-log	0.556 ± 0.276	40 ± 25	0.393 ± 0.826	0.096 ± 1.069
	ILSR (no $\boldsymbol{X}$)	0.033 ± 0.012	2 ± 0	0.334 ± 0.0	-0.047 ± 1.089
	MM (no $\boldsymbol{X}$)	1.824 ± 0.701	267 ± 151	0.334 ± 0.0	0.0 ± 0.0
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	2.728 ± 1.475	13 ± 3	0.334 ± 0.0	-0.047 ± 1.089
	Newton on $\boldsymbol{\beta}$	1.966 ± 3.158	10 ± 19	0.322 ± 0.802	-0.261 ± 0.956
	SLSQP	1.656 ± 1.793	20 ± 30	0.608 ± 0.826	0.358 ± 0.928

Table 3: Evaluations on real datasets partitioned w.r.t. sample CV (c.f. Sec. 5). We report the convergence time (Time), number of iterations until convergence (Iter), top-1 accuracy on the test set (Top-1 Acc.), and Kendall-Tau correlation on the test set (KT). ILSR, MM, and Newton on $\boldsymbol{\theta}$ do not use the features $\boldsymbol{X}$. Newton on $\boldsymbol{\beta}$ and SLSQP regress $\boldsymbol{\pi}$ from $\boldsymbol{X}$.

quires about 2 times more iterations, but still converges 3 times faster than Newton on $\boldsymbol{\beta}$ on FAC. Moreover, while significantly decreasing the convergence time, PLADMM or PLADMM-log consistently attain similar prediction performance to Newton on $\boldsymbol{\beta}$ and SLSQP, except for Sushi, for which they perform slightly worse (by $13\% - 20\%$ Top-1 Acc.), though the convergence time dividends are striking in comparison ($78\% - 95\%$).

Aligned with the prediction performance on synthetic datasets, featureless methods ILSR, MM, and Newton on $\boldsymbol{\theta}$ can only attain 0.5 Top-1 Acc. and 0.0 KT. By regressing the Plackett-Luce scores from features, PLADMM and PLADMM-log significantly outperform the predictions of ILSR, MM, and Newton on $\boldsymbol{\theta}$, by $3\% - 31\%$ Top-1 Acc. and $5\% - 78\%$ KT.

Observation CV. A sample can appear in both training and test observations in observation CV. Hence, featureless methods ILSR, MM, and Newton on $\boldsymbol{\theta}$ should fare better than in sample CV. Nonetheless, in Table 2, as $n = 1000$ is larger than $p = 100$, there are more scores to learn than parameters. As a result, feature methods are advantageous for good predictions compared to featureless methods. Particularly, PLADMM and PLADMM-log outperform the predictions of ILSR, MM, and Newton on $\boldsymbol{\theta}$ in observation CV on Table 2, by 13% Top-1 Acc. and 9% KT. The relative performance of feature vs. featureless methods is governed by the relationship among n, p, and M. We therefore explore the effect of n and p below; the effect of M is discussed in Appendix F.4. We do not include Newton on $\boldsymbol{\theta}$ and MM in this analysis, as they are too slow.

Impact of p. To assess the impact of number of parameters, we fix $n = 1000$, $M = 250$, $|A_\ell| = 2$ and generate synthetic datasets with $p \in \{10, 100, 1000, 10000\}$. Fig. 1a shows the Time and Top-1 Acc. of PLADMM, PLADMM-log, ILSR, and Newton on $\boldsymbol{\beta}$. As $M = 250$ observations are not enough to learn $n = 1000$ scores, PLADMM leads to significantly better Top-1 Acc. compared to ILSR. When $p = 10$, PLADMM and PLADMM-log outperform ILSR by $18\% - 28\%$ Top-1 Acc. Moreover, PLADMM and PLADMM-log are consistently faster than Newton on $\boldsymbol{\beta}$, for all $p > 100$. Particularly, for $p = 10000$, PLADMM and PLADMM-log converge *42-579 times faster than Newton on $\boldsymbol{\beta}$*. Interestingly, the convergence time of PLADMM-log can even decrease with increasing p. This is because the number of iterations until convergence decreases. While significantly decreasing the convergence time, PLADMM consistently attains better Top-1 Acc. than Newton on $\boldsymbol{\beta}$, up to 8% for $p = 100$.

Impact of n. To assess the impact of number of samples, we fix $p = 100$, $M = 250$, $|A_\ell| = 2$ and generate synthetic datasets with $n \in \{50, 100, 1000, 10000, 100000\}$; Fig. 1b shows evaluations on the resulting datasets. For $n > p = 100$, i.e., when there are more scores to learn than parameters, PLADMM leads to significantly better Top-1 Acc. compared to ILSR. Particularly, for $n = 100000$, PLADMM outperforms ILSR by 25% Top-1 Acc. This confirms that, especially when the number of observations M is not sufficient to learn n scores, exploiting the features associated with the samples is crucial in attaining good prediction performance. As expected, convergence time of Newton on $\boldsymbol{\beta}$ is not significantly affected by n. Despite this, PLADMM and PLADMM-log are faster than Newton on $\boldsymbol{\beta}$ for all $n < 1000$. Particularly, for $n = 50$, PLADMM and PLADMM-log converge *2-75 times faster than Newton on $\boldsymbol{\beta}$*. While decreasing the convergence time, PLADMM consistently attains better Top-1 Acc. than Newton on $\boldsymbol{\beta}$, up to 19% for $n = 100000$.

Real datasets. We include the evaluations on real datasets partitioned w.r.t. observation CV in the Appendix (c.f.Table 4). Performance agrees with observations above regarding the dependence on n an p. For datasets where $n > M > p$, e.g., FAC, PLADMM and PLADMM-log significantly outperform the predictions of ILSR, by 10% Top-1 Acc. and 25% KT. For datasets where M is much larger than n (c.f. Table 1), feature methods lead to similar prediction performance to each other and slightly lower performance than ILSR, MM, and Newton on $\boldsymbol{\theta}$. Overall, PLADMM and PLADMM-log consistently converge faster than Newton on $\boldsymbol{\beta}$ and SLSQP, by $3 - 27$ times across all real datasets.

6 Conclusions

We solve the maximum likelihood estimation problem for the Plackett-Luce scores via ADMM. We show that the scores are equivalent to the stationary distribution of a Markov Chain and propose spectral algorithms, PLADMM and PLADMM-log. We model the Plackett-Luce scores as affine and logistic functions of features. Extending these to more complex models, particularly to deep neural networks, is an interesting open problem. Our approach has the potential of training a neural network over a linear penalty w.r.t. scores, where the latter are regressed efficiently via a spectral method over the *quadratic* pairwise ranking data. This can lead to significant improvements over training time, making an epoch linear rather than quadratic in sample size.

Acknowledgments

We are supported by NIH (R01EY019474), NSF (SCH-1622542 at MGH; SCH-1622536 at Northeastern; SCH-1622679 at OHSU), and by unrestricted departmental funding from Research to Prevent Blindness (OHSU).

Bibliography

Agarwal, A., Patil, P., and Agarwal, S. (2018). Accelerated spectral ranking. In *International Conference on Machine Learning*, pages 70–79.

Agarwal, S. (2016). On ranking and choice models. In *IJCAI*, pages 4050–4053.

Ammar, A. and Shah, D. (2011). Ranking: Compare, don't score. In *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 776–783. IEEE.

Ataer-Cansızoğlu, E. (2015). *Retinal image analytics: A complete framework from segmentation to diagnosis*. Northeastern University.

Azari, H., Parks, D., and Xia, L. (2012). Random utility theory for social choice. In *Advances in Neural Information Processing Systems*, pages 126–134.

Blanchet, J., Gallego, G., and Goyal, V. (2016). A markov chain approximation to choice modeling. *Operations Research*, 64(4):886–905.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

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 *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 268–276. Society for Industrial and Applied Mathematics.

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., and Hullender, G. (2005). Learning to rank using gradient descent. In *Proceedings of the International Conference on Machine Learning*, pages 89–96. ACM.

Caron, F. and Doucet, A. (2012). Efficient bayesian inference for generalized bradley–terry models. *Journal of Computational and Graphical Statistics*, 21(1):174–196.

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

Chang, H., Yu, F., Wang, J., Ashley, D., and Finkelstein, A. (2016). Automatic triage for a photo series. *ACM Transactions on Graphics (TOG)*, 35(4):148.

Chartrand, R. and Wohlberg, B. (2013). A nonconvex ADMM algorithm for group sparsity with sparse groups. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6009–6013.

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

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Dubey, A., Naik, N., Parikh, D., Raskar, R., and Hidalgo, C. A. (2016). Deep learning the city: Quantifying urban perception at a global scale. In *European Conference on Computer Vision*, pages 196–212. Springer.

Dwork, C., Kumar, R., Naor, M., and Sivakumar, D. (2001). Rank aggregation methods for the web. In *Proceedings of the 10th International Conference on World Wide Web*, pages 613–622. ACM.

Dykstra, O. (1960). Rank analysis of incomplete block designs: A method of paired comparisons employing unequal repetitions on pairs. *Biometrics*, 16(2):176–188.

Elo, A. E. (1978). *The rating of chessplayers, past and present*. Arco Pub.

Fligner, M. A. and Verducci, J. S. (1993). *Probability models and statistical analyses for ranking data*, volume 80. Springer.

Gallager, R. G. (2013). *Stochastic Processes: Theory for Applications*. Cambridge University Press.

Guiver, J. and Snelson, E. (2009). Bayesian inference for plackett-luce ranking models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 377–384. ACM.

Guo, K., Han, D., and Wu, T.-T. (2017). Convergence of alternating direction method for minimizing sum of two nonconvex functions with linear constraints. *International Journal of Computer Mathematics*, 94(8):1653–1669.

Guo, Y., Tian, P., Kalpathy-Cramer, J., Ostmo, S., Campbell, J. P., Chiang, M. F., Erdogmus, D., Dy, J. G., and Ioannidis, S. (2018). Experimental design under the bradley-terry model. In *IJCAI*, pages 2198–2204.

Hajek, B., Oh, S., and Xu, J. (2014). Minimax-optimal inference from partial rankings. In *Advances in Neural Information Processing Systems*, pages 1475–1483.

Han, B. (2018). Dateline: Deep plackett-luce model with uncertainty measurements. *arXiv preprint arXiv:1812.05877*.

Hong, M. (2018). A distributed, asynchronous, and incremental algorithm for nonconvex optimization: An ADMM approach. *IEEE Transactions on Control of Network Systems*, 5(3):935–945.

Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Cambridge university press.

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

Jang, M., Kim, S., Suh, C., and Oh, S. (2017). Optimal sample complexity of m-wise data for top-k ranking. In *Advances in Neural Information Processing Systems*, pages 1686–1696.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142. ACM.

Jolliffe, I. T. (1986). Principal component analysis and factor analysis. In *Principal Component Analysis*. Springer.

Kamishima, T., Hamasaki, M., and Akaho, S. (2009). A simple transfer learning method and its application to personalization in collaborative tagging. In *ICDM*.

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

Khetan, A. and Oh, S. (2016). Computational and statistical tradeoffs in learning to rank. In *Advances in Neural Information Processing Systems*, pages 739–747.

Lei, Q., Zhong, K., and Dhillon, I. S. (2016). Coordinate-wise power method. In *Advances in Neural Information Processing Systems*, pages 2064–2072.

Lu, T. and Boutilier, C. (2011). Learning mallows models with pairwise preferences. In *Proceedings of the 28th International Conference on Machine Learning (icml-11)*, pages 145–152.

Mallows, C. L. (1957). Non-null ranking models. i. *Biometrika*, 44(1/2):114–130.

Mao, C., Pananjady, A., and Wainwright, M. J. (2018). Breaking the $1/\sqrt{n}$ barrier: Faster rates for permutation-based models in polynomial time. In *Conference On Learning Theory*, pages 2037–2042.

Mao, C., Weed, J., and Rigollet, P. (2017). Minimax rates and efficient algorithms for noisy sorting. *arXiv preprint arXiv:1710.10388*.

Marden, J. I. (2014). *Analyzing and modeling rank data*. Chapman and Hall/CRC.

Maystre, L. and Grossglauser, M. (2015). Fast and accurate inference of plackett-luce models. In *Advances in Neural Information Processing Systems*, pages 172–180.

McFadden, D. (1973). Conditional logit analysis of qualitative choice behavior.

McFadden, D. (2000). Disaggregate behavioral travel demand's rum side. *Travel Behaviour Research*, pages 17–63.

Negahban, S., Oh, S., and Shah, D. (2012). Iterative ranking from pair-wise comparisons. In *Advances in Neural Information Processing Systems*, pages 2474–2482.

Negahban, S., Oh, S., Thekumparampil, K. K., and Xu, J. (2018). Learning from comparisons and choices. *The Journal of Machine Learning Research*, 19(1):1478–1572.

Niranjan, U. and Rajkumar, A. (2017). Inductive pairwise ranking: going beyond the n log (n) barrier. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Nocedal, J. and Wright, S. (2006). *Numerical optimization*. Springer Science & Business Media.

Pahikkala, T., Tsivtsivadze, E., Airola, A., Järvinen, J., and Boberg, J. (2009). An efficient algorithm for learning to rank from preference graphs. *Machine Learning*, 75(1):129–165.

Plackett, R. L. (1975). The analysis of permutations. *Applied Statistics*, pages 193–202.

Ragain, S. and Ugander, J. (2016). Pairwise choice markov chains. In *Advances in Neural Information Processing Systems*, pages 3198–3206.

Rajkumar, A. and Agarwal, S. (2014). A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *International Conference on Machine Learning*, pages 118–126.

Rajkumar, A. and Agarwal, S. (2016). When can we rank well from comparisons of o (n\log (n)) non-actively chosen pairs? In *Conference on Learning Theory*, pages 1376–1401.

Ryzin, G. v. and Mahajan, S. (1999). On the relationship between inventory costs and variety benefits in retail assortments. *Management Science*, 45(11):1496–1509.

Saha, A. and Rajkumar, A. (2018). Ranking with features: Algorithm and a graph theoretic analysis. *arXiv preprint arXiv:1808.03857*.

Shah, N., Balakrishnan, S., Guntuboyina, A., and Wainwright, M. (2016a). Stochastically transitive models for pairwise comparisons: Statistical and computational issues. In *International Conference on Machine Learning*, pages 11–20.

Shah, N. B., Balakrishnan, S., Bradley, J., Parekh, A., Ramchandran, K., and Wainwright, M. J. (2016b).

Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence. *The Journal of Machine Learning Research*, 17(1):2049–2095.

Soufiani, H. A., Chen, W., Parkes, D. C., and Xia, L. (2013). Generalized method-of-moments for rank aggregation. In *Advances in Neural Information Processing Systems*, pages 2706–2714.

Sun, W.-T., Chao, T.-H., Kuo, Y.-H., and Hsu, W. H. (2017). Photo filter recommendation by category-aware aesthetic learning. *IEEE Transactions on Multimedia*, 19(8):1870–1880.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Computer Vision and Pattern Recognition, 2015*.

Thurstone, L. L. (1927). The method of paired comparisons for social values. *The Journal of Abnormal and Social Psychology*, 21(4):384.

Tian, P., Guo, Y., Kalpathy-Cramer, J., Ostmo, S., Campbell, J. P., Chiang, M. F., Dy, J., Erdogmus, D., and Ioannidis, S. (2019). A severity score for retinopathy of prematurity. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1809–1819. ACM.

Vojnovic, M. and Yun, S. (2016). Parameter estimation for generalized thurstone choice models. In *International Conference on Machine Learning*, pages 498–506.

Wang, Y., Yin, W., and Zeng, J. (2019). Global convergence of ADMM in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78(1):29–63.

Wauthier, F., Jordan, M., and Jojic, N. (2013). Efficient ranking from pairwise comparisons. In *International Conference on Machine Learning*, pages 109–117.

Yıldız, İ., Tian, P., Dy, J., Erdoğmuş, D., Brown, J., Kalpathy-Cramer, J., Ostmo, S., Campbell, J. P., Chiang, M. F., and Ioannidis, S. (2019). Classification and comparison via neural networks. *Neural Networks*.

Zeng, J., Ouyang, S., Lau, T. T.-K., Lin, S., and Yao, Y. (2018). Global convergence in deep learning with variable splitting via the kurdyka-lojasiewicz property. *arXiv preprint arXiv:1803.00225*.

A On the Convexity of the PL Negative Log-Likelihood

The Hessian of Eq. (3), given by Eq. (48), is not in general positive semidefinite (PSD) (Hunter, 2004). A simple counterexample is as follows: consider $n = 2$ samples and a single observation, i.e., $M = 1$. The Hessian in this case is negative-definite for all $\pi_1, \pi_2 > 0$. Thus, Problem (4) with objective (3) is in general non-convex in $(\boldsymbol{\beta}, b) \in \mathbb{R}^{p+1}$. On the other hand, (3) under parametrization $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$ is convex as a consequence of the convexity of the log of the sum of exponentials, which is well known (see (Boyd and Vandenberghe, 2004)). The convexity of Problem (5) w.r.t. $\boldsymbol{\beta}$ follows by this observation and also the fact that the composition of convex and affine is convex.

B Proof of Theorem 3.1 (Maystre and Grossglauser, 2015)

We start by showing that $\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi})}{\partial \pi_i} = 0$, $i \in \mathcal{N}$ is the optimality condition to minimize Eq. (3). Consider the reparametrization $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$. Eq. (3) under this reparametrization is given by:

$$\mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\theta}) = \sum_{\ell=1}^{M} \left(\log \sum_{j \in A_\ell} e^{\theta_j} - \theta_\ell \right), \tag{21}$$

which is convex w.r.t. $\boldsymbol{\theta} = [\theta_i]_{i \in \mathcal{N}}$, i.e., even though Eq. (3) is not convex w.r.t. $\boldsymbol{\pi}$, it is convex under the reparametrization $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$. This implies that $\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\theta})}{\partial \theta_i} = 0$, $i \in \mathcal{N}$ is the optimality condition to minimize Eq. (21) w.r.t. $\boldsymbol{\theta}$. By the chain rule, this condition can be written in terms of $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$ as:

$$\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\theta})}{\partial \theta_i} = \frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi})}{\partial \pi_i} e^{\theta_i} = 0 \quad \forall i \in \mathcal{N}. \tag{22}$$

Note that $e^{\theta_i} > 0$, $i \in \mathcal{N}$. Then, $\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\theta})}{\partial \theta_i} = 0$ is equivalent to $\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi})}{\partial \pi_i} = 0$, $i \in \mathcal{N}$, i.e., $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$ satisfies Eq.(22) if and only if θ_i, $i \in \mathcal{N}$ is the minimizer of Eq. (21). Hence, the stationarity condition $\frac{\partial \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi})}{\partial \pi_i} = 0$, $i \in \mathcal{N}$ is also the optimality condition for problem (6).

The optimality condition is given explicitly by:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \pi_i} = &\sum_{\ell \in W_i} \left(\frac{1}{\sum_{t \in A_\ell} \pi_t} - \frac{1}{\pi_i} \right) \\ &+ \sum_{\ell \in L_i} \frac{1}{\sum_{t \in A_\ell} \pi_t} = 0 \quad \forall i \in \mathcal{N}, \end{aligned} \tag{23}$$

where $W_i = \{\ell \,|\, i \in A_\ell, c_\ell = i\}$ is the set of observations where sample $i \in \mathcal{N}$ is chosen and $L_i = \{\ell \,|\, i \in A_\ell, c_\ell \neq i\}$ is the set of observations where sample $i \in \mathcal{N}$ is not chosen. Multiplying both sides of Eq. (23) with π_i, $i \in \mathcal{N}$, we have:

$$\sum_{\ell \in L_i} \left(\frac{\pi_i}{\sum_{t \in A_\ell} \pi_t} \right) - \sum_{\ell \in W_i} \left(\frac{\sum_{j \neq i \in A_\ell} \pi_j}{\sum_{t \in A_\ell} \pi_t} \right) = 0, \tag{24}$$

for all $i \in \mathcal{N}$. Note that $\sum_{\ell \in W_i} \sum_{j \neq i \in A_\ell} \cdot = \sum_{j \neq i} \sum_{\ell \in W_i \cap L_j} \cdot$ and $\sum_{\ell \in L_i} \cdot = \sum_{j \neq i} \sum_{\ell \in W_j \cap L_i} \cdot$. Accordingly, we rewrite Eq. (24) as:

$$\begin{aligned} &\sum_{j \neq i} \sum_{\ell \in W_j \cap L_i} \left(\frac{\pi_i}{\sum_{t \in A_\ell} \pi_t} \right) \\ &= \sum_{j \neq i} \sum_{\ell \in W_i \cap L_j} \left(\frac{\pi_j}{\sum_{t \in A_\ell} \pi_t} \right) \quad \forall i \in \mathcal{N}. \end{aligned} \tag{25}$$

Then, an optimal solution $\boldsymbol{\pi} \in \mathbb{R}^n_+$ to Eq. (6) satisfies:

$$\sum_{j \neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) = \sum_{j \neq i} \pi_i \lambda_{ij}(\boldsymbol{\pi}) \quad \forall i \in \mathcal{N}, \tag{26}$$

where $\lambda_{ji}(\boldsymbol{\pi})$, $i, j \in \mathcal{N}, i \neq j$ are given by Eq. (8).

C Alternating Directions Method of Multipliers

We employ Alternating Directions Method of Multipliers (ADMM) to solve the problem in Eq.(11) (Boyd et al., 2011). ADMM is a primal-dual algorithm designed for problems with decoupled objectives, i.e., objectives that can be written as a sum of functions where each function depends on only one of the optimized variables. In our case, we solve Eq.(11) for $\tilde{\boldsymbol{\beta}}$ and $\boldsymbol{\pi}$, and the objective $\mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi})$ is a function of $\boldsymbol{\pi}$ only.

ADMM solves a constrained optimization problem by minimizing the *augmented Lagrangian*, rather than the standard Lagrangian. The difference of augmented Lagrangian from the standard Lagrangian is the additional quadratic penalty on the equality constraint. This additional penalty is shown to greatly improve convergence properties of the algorithm (Boyd et al., 2011). The augmented Lagrangian of Eq. (11) is:

$$\begin{aligned} L_\rho(\tilde{\boldsymbol{\beta}}, \boldsymbol{\pi}, \boldsymbol{y}) = &\mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi}) \\ &+ \boldsymbol{y}^T(\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}) + \frac{\rho}{2} \| \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi} \|_2^2, \end{aligned} \tag{27}$$

where $\rho > 0$ is the penalty parameter, $\boldsymbol{y} \in \mathbb{R}^n$ is the dual variable, $\tilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}, b) \in \mathbb{R}^{p+1}$ and $\tilde{\boldsymbol{X}} = [\boldsymbol{X}|\mathbf{1}] \in \mathbb{R}^{n \times (p+1)}$, so that $\boldsymbol{\pi} = \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}$.

ADMM alternates between optimizing the primal variables $\tilde{\boldsymbol{\beta}}$ and $\boldsymbol{\pi}$, and the dual variable $\boldsymbol{y}$. Applying

ADMM on problem (11) yields the following iterative algorithm:

$$\tilde{\boldsymbol{\beta}}^{k+1} = \underset{\tilde{\boldsymbol{\beta}}\in\mathbb{R}^{p+1}}{\arg\min}\, \boldsymbol{y}^{k^T}(\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}^k) + \tfrac{\rho}{2}\,\|\,\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}^k\,\|_2^2$$
$$= (\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^T(\boldsymbol{\pi}^k - \frac{1}{\rho}\boldsymbol{y}^k), \tag{28a}$$
$$\boldsymbol{\pi}^{k+1} = \underset{\boldsymbol{\pi}\in\mathbb{R}^n_+}{\arg\min}\big(\mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi}) + \boldsymbol{y}^{k^T}(\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{\pi}) + \frac{\rho}{2}\,\|\,\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{\pi}\,\|_2^2\,\big), \tag{28b}$$
$$\boldsymbol{y}^{k+1} = \boldsymbol{y}^k + \rho(\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{\pi}^{k+1}). \tag{28c}$$

For convenience in calculations, the augmented Lagrangian in (27) can be written in a different form, by introducing a scaled dual variable $\boldsymbol{u} = \frac{1}{\rho}\boldsymbol{y}$ and combining the linear and quadratic terms. By doing so, Eq. (27) is equivalent to the final form of the augmented Lagrangian:

$$L_\rho(\tilde{\boldsymbol{\beta}}, \boldsymbol{\pi}, \boldsymbol{u}) = \mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi}) + \frac{\rho}{2}\,\|\,\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi} + \boldsymbol{u}\,\|_2^2 - \frac{\rho}{2}\,\|\,\boldsymbol{u}\,\|_2^2\,. \tag{29}$$

Having formed the final augmented Lagrangian in Eq. (29), applying ADMM on problem (11) yields the iterative steps:

$$\tilde{\boldsymbol{\beta}}^{k+1} = \underset{\tilde{\boldsymbol{\beta}}\in\mathbb{R}^{p+1}}{\arg\min}\,\|\,\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} - \boldsymbol{\pi}^k + \boldsymbol{u}^k\,\|_2^2$$
$$= (\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^T(\boldsymbol{\pi}^k - \boldsymbol{u}^k), \tag{30a}$$
$$\boldsymbol{\pi}^{k+1} = \underset{\boldsymbol{\pi}\in\mathbb{R}^n_+}{\arg\min}\big(\mathcal{L}(\mathcal{D}\,|\,\boldsymbol{\pi}) + \tfrac{\rho}{2}\,\|\,\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{\pi} + \boldsymbol{u}^k\,\|_2^2\,\big), \tag{30b}$$
$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{\pi}^{k+1}. \tag{30c}$$

For convex problems, there are well-established convergence properties for ADMM. If the objective is closed, proper, and convex, and the standard Lagrangian of the problem has a saddle point, then the ADMM iterations are guaranteed to converge to a point where (a) the equality constraint is satisfied, and (b) objective and dual variable attain optimal values. Moreover, in many applications, ADMM has been shown to converge to a modest accuracy in a few tens of iterations (Boyd et al., 2011). For nonconvex problems, there are few convergence analyses for ADMM, which focus on a restricted class of problems (Guo et al., 2017). In general, ADMM is not guaranteed to converge for non-convex problems, and even if it does, it may not converge to the optimal point of the problem. Nevertheless, ADMM is extensively used to also solve nonconvex problems similar to the one we study (Chartrand and Wohlberg, 2013; Guo et al., 2017; Hong, 2018; Wang et al., 2019).

D Proofs

D.1 Proof of Lemma 4.1

At the k-th iteration of ADMM, gradient of the augmented Lagrangian in (29) w.r.t. $\boldsymbol{\pi}$ is:

$$\nabla_{\boldsymbol{\pi}} L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k) = \nabla_{\boldsymbol{\pi}}\mathcal{L} + \rho(\boldsymbol{\pi} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{u}^k). \tag{31}$$

To simplify the rest of the calculations, we introduce $\boldsymbol{\sigma} = \rho(\boldsymbol{\pi} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{u}^k) \in \mathbb{R}^n$. Then, the stationarity condition $\frac{\partial L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)}{\partial \pi_i} = 0$, $i \in \mathcal{N}$, is equivalent to:

$$\frac{\partial L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)}{\partial \pi_i} = \frac{\partial \mathcal{L}}{\partial \pi_i} + \sigma_i = 0 \quad \forall i \in \mathcal{N}. \tag{32}$$

Setting $\frac{\partial \mathcal{L}}{\partial \pi_i}$ from Eq. (23) to Eq. (32), we have:

$$\frac{\partial L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)}{\partial \pi_i} = \sum_{\ell\in W_i}\left(-\frac{1}{\pi_i} + \frac{1}{\sum_{t\in A_\ell}\pi_t}\right) + \sum_{\ell\in L_i}\frac{1}{\sum_{t\in A_\ell}\pi_t} + \sigma_i = 0, \tag{33}$$

for all $i \in \mathcal{N}$. Multiplying both sides of Eq. (33) with $-\pi_i$, $i \in \mathcal{N}$, we have:

$$\sum_{\ell\in W_i}\left(\frac{\sum_{j\neq i\in A_\ell}\pi_j}{\sum_{t\in A_\ell}\pi_t}\right) - \sum_{\ell\in L_i}\left(\frac{\pi_i}{\sum_{t\in A_\ell}\pi_t}\right) - \pi_i\sigma_i = 0 \quad \forall i \in \mathcal{N}. \tag{34}$$

Note that $\sum_{\ell\in W_i}\sum_{j\neq i\in A_\ell}\cdot = \sum_{j\neq i}\sum_{\ell\in W_i\cap L_j}\cdot$ and $\sum_{\ell\in L_i}\cdot = \sum_{j\neq i}\sum_{\ell\in W_j\cap L_i}\cdot$. Accordingly, we rewrite Eq. (34) as:

$$\sum_{j\neq i}\sum_{\ell\in W_i\cap L_j}\left(\frac{\pi_j}{\sum_{t\in A_\ell}\pi_t}\right) - \sum_{j\neq i}\sum_{\ell\in W_j\cap L_i}\left(\frac{\pi_i}{\sum_{t\in A_\ell}\pi_t}\right) - \pi_i\sigma_i = 0 \;\forall i \in \mathcal{N}. \tag{35}$$

Then, the stationarity condition $\frac{\partial L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)}{\partial \pi_i} = 0$, $i \in \mathcal{N}$ is equivalent to:

$$\sum_{j\neq i}\pi_j\lambda_{ji}(\boldsymbol{\pi}) - \sum_{j\neq i}\pi_i\lambda_{ij}(\boldsymbol{\pi}) = \pi_i\sigma_i \quad \forall i \in \mathcal{N}, \tag{36}$$

where $\lambda_{ji}(\boldsymbol{\pi})$, $i, j \in \mathcal{N}, i \neq j$ are given by Eq. (8).

D.2 Proof of Theorem 4.2

Summing Eq. (15) for $i \in \mathcal{N}$, we get:

$$\sum_i\sum_j(\pi_j\lambda_{ji}(\boldsymbol{\pi}) - \pi_i\lambda_{ij}(\boldsymbol{\pi}))\,\mathbb{1}_{j\neq i} = \sum_i\pi_i\sigma_i = 0. \tag{37}$$

Since the Plackett-Luce scores are non-negative, i.e. $\pi_i \geq 0$, $i \in \mathcal{N}$, Eq. (37) implies that $\boldsymbol{\sigma} \equiv [\sigma_i]_{i\in\mathcal{N}}$ contains both positive and negative elements. Let $(\mathcal{N}_+, \mathcal{N}_-)$ be a partition of $\mathcal{N}$ such that $\sigma_i \geq 0$ for all $i \in \mathcal{N}_+$ and $\sigma_i < 0$ for all $i \in \mathcal{N}_-$. Then, for $i \in \mathcal{N}_+$ in Eq. (15), we have:

$$\sum_{j\neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) = \pi_i \left(\sum_{j\neq i} \lambda_{ij}(\boldsymbol{\pi}) + \sigma_i \right), \ \forall i \in \mathcal{N}_+, \quad (38)$$

where $\lambda_{ij}(\boldsymbol{\pi}) + \sigma_i \geq 0$, $i \in \mathcal{N}_+$ and $j \in \mathcal{N}$. Eq. (38) shows that from each state $i \in \mathcal{N}_+$ into the states in $\mathcal{N}_-$, there exists a total of σ_i "additional outgoing rate", compared to Eq. (7). At the same time, for $i \in \mathcal{N}_-$ in Eq. (15), we have:

$$\begin{aligned} &\sum_{j\in\mathcal{N}_+} \pi_j \lambda_{ji}(\boldsymbol{\pi}) + \sum_{j\in\mathcal{N}_-|j\neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) \\ &\quad = \pi_i \sum_{j\neq i} \lambda_{ij}(\boldsymbol{\pi}) + \pi_i \sigma_i, \quad \forall i \in \mathcal{N}_-. \end{aligned} \quad (39)$$

Since $\pi_i \sigma_i < 0$, for $i \in \mathcal{N}_-$, we distribute these terms into the first sum on the left hand side. Then, Eq. (39) is equivalent to:

$$\begin{aligned} &\sum_{j\in\mathcal{N}_+} \pi_j \left(\lambda_{ji}(\boldsymbol{\pi}) - \frac{\pi_i \sigma_i c_j}{\pi_j} \right) + \sum_{j\in\mathcal{N}_-|j\neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) \\ &= \pi_i \sum_{j\neq i} \lambda_{ij}(\boldsymbol{\pi}), \quad \forall i \in \mathcal{N}_-, \end{aligned} \quad (40)$$

where $\sum_{j\in\mathcal{N}_+} c_j = 1$.

To determine the $\{c_j\}_{j\in\mathcal{N}_+}$, recall from Eq. (38) that from each state $j \in \mathcal{N}_+$ into the states $i \in \mathcal{N}_-$, there exists a total of σ_j additional outgoing rate. Then, Eq. (40) implies that $\sum_{i\in\mathcal{N}_-} -\frac{\pi_i \sigma_i c_j}{\pi_j} = \sigma_j$, i.e., $c_j = \frac{-\pi_j \sigma_j}{\sum_{i\in\mathcal{N}_-} \pi_i \sigma_i}$, $j \in \mathcal{N}_+$. Using $-\sum_{i\in\mathcal{N}_-} \pi_i \sigma_i = \sum_{i\in\mathcal{N}_+} \pi_i \sigma_i$ from Eq. (37), we confirm that $\sum_{j\in\mathcal{N}_+} c_j = \frac{-\sum_{j\in\mathcal{N}_+} \pi_j \sigma_j}{\sum_{i\in\mathcal{N}_-} \pi_i \sigma_i} = 1$, and rewrite $\{c_j\}_{j\in\mathcal{N}_+}$ as:

$$c_j = \frac{-2\pi_j \sigma_j}{\sum_{i\in\mathcal{N}_-} \pi_i \sigma_i - \sum_{i\in\mathcal{N}_+} \pi_i \sigma_i}, \quad \forall j \in \mathcal{N}_+. \quad (41)$$

Finally, setting $\{c_j\}_{j\in\mathcal{N}_+}$ into Eq. (40), we have:

$$\begin{aligned} &\sum_{j\in\mathcal{N}_+} \pi_j \left(\lambda_{ji}(\boldsymbol{\pi}) + \frac{2\pi_i \sigma_i \sigma_j}{\sum_{t\in\mathcal{N}_-} \pi_t \sigma_t - \sum_{t\in\mathcal{N}_+} \pi_t \sigma_t} \right) \\ &+ \sum_{j\in\mathcal{N}_-|j\neq i} \pi_j \lambda_{ji}(\boldsymbol{\pi}) = \pi_i \sum_{j\neq i} \lambda_{ij}(\boldsymbol{\pi}), \quad \forall i \in \mathcal{N}_-, \end{aligned} \quad (42)$$

where $\lambda_{ji}(\boldsymbol{\pi}) + \frac{2\pi_i \sigma_i \sigma_j}{\sum_{t\in\mathcal{N}_-} \pi_t \sigma_t - \sum_{t\in\mathcal{N}_+} \pi_t \sigma_t} \geq 0$, $j \in \mathcal{N}_+$ and $i \in \mathcal{N}_-$.

Eq. (15), partitioned as Eq. (38) and Eq. (42), is the balance equations of a continuous-time MC with transition rates given by:

$$\mu_{ji}(\boldsymbol{\pi}) = \begin{cases} \lambda_{ji}(\boldsymbol{\pi}) + \frac{2\pi_i \sigma_i \sigma_j}{\sum_{t\in\mathcal{N}_-} \pi_t \sigma_t - \sum_{t\in\mathcal{N}_+} \pi_t \sigma_t} \\ \qquad \text{if } j \in \mathcal{N}_+ \text{ and } i \in \mathcal{N}_- \\ \lambda_{ji}(\boldsymbol{\pi}) \quad \text{otherwise.} \end{cases} \quad (43)$$

Hence, $\boldsymbol{\pi}$ is the stationary distribution of this MC (Gallager, 2013).

D.3 Proof of Theorem 4.3

We use the following definition.

Definition D.1 (Diagonal dominance)**.** *A matrix $\boldsymbol{H}$ is diagonally dominant if $|\boldsymbol{H}_{ii}| \geq \sum_{j\neq i} |\boldsymbol{H}_{ij}|$, $i \in \mathcal{N}$, i.e., for every row, magnitude of the diagonal element is larger than the sum of magnitudes of all off-diagonal elements (Horn and Johnson, 2012).*

Eq. (33) is equivalent to:

$$\begin{aligned} \frac{\partial L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)}{\partial \pi_i} =& \sum_{\ell\in W_i} -\frac{1}{\pi_i} + \sum_{\ell|i\in A_\ell} \frac{1}{\sum_{t\in A_\ell} \pi_t} \\ &+ \rho(\boldsymbol{\pi} - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k+1} - \boldsymbol{u}^k)_i \ , \end{aligned} \quad (44)$$

for all $i \in \mathcal{N}$. At the k-th iteration of (13), let $\nabla^2 L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ be the Hessian of the augmented Lagrangian w.r.t. $\boldsymbol{\pi}$. Differentiating Eq. (44) w.r.t. π_j, $\nabla^2 L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ has the following form:

$$\begin{aligned} &\nabla^2_{ij} L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k) \\ &= \begin{cases} \sum_{\ell\in W_i} \frac{1}{\pi_i^2} - \sum_{\ell|i\in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2} + \rho, & i = j \\ -\sum_{\ell|i,j\in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2}, & i \neq j. \end{cases} \end{aligned} \quad (45)$$

Consider $\rho \geq \frac{2}{\epsilon^2} \max_i \sum_{\ell|i\in A_\ell} \frac{1}{|A_\ell|^2}$. By Assumption 4.1, we have:

$$\begin{aligned} &\rho \geq \frac{2}{\epsilon^2} \sum_{\ell|i\in A_\ell} \frac{1}{|A_\ell|^2} \ \forall i \in \mathcal{N}, \\ &\Leftrightarrow \ \rho \geq \sum_{\ell|i\in A_\ell} \frac{2}{\left(\sum_{t\in A_\ell} \pi_t\right)^2} \ \forall i \in \mathcal{N}, \\ &\Leftrightarrow \ \rho + \sum_{\ell\in W_i} \frac{1}{\pi_i^2} > \sum_{\ell|i\in A_\ell} \frac{2}{\left(\sum_{t\in A_\ell} \pi_t\right)^2} \ \forall i \in \mathcal{N}, \quad (46a) \\ &\Leftrightarrow \ \rho + \sum_{\ell\in W_i} \frac{1}{\pi_i^2} > \sum_{\ell|i\in A_\ell} \frac{1}{\left(\sum_{t\in A_\ell} \pi_t\right)^2} \\ &\qquad + \sum_{j\neq i} \sum_{\ell|i,j\in A_\ell} \frac{1}{\left(\sum_{t\in A_\ell} \pi_t\right)^2} \ \forall i \in \mathcal{N}. \quad (46b) \end{aligned}$$

Eq. (46a) implies that all diagonal elements of $\nabla^2 L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ are positive. Also, by Eq. (46b),

$\nabla^2 L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ is diagonally dominant (c.f. Definition D.1). Thus, $\nabla^2 L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ is positive definite (Horn and Johnson, 2012), i.e., $L_\rho(\tilde{\boldsymbol{\beta}}^{k+1}, \boldsymbol{\pi}, \boldsymbol{u}^k)$ is convex w.r.t. $\boldsymbol{\pi}$. As a result, under Assumption 4.1, for $\rho \geq \frac{2}{\epsilon^2} \max_i \sum_{\ell | i \in A_\ell} \frac{1}{|A_\ell|^2}$, a stationary $\boldsymbol{\pi} > \mathbf{0}$ satisfying condition (14) is also a minimizer of step (13b).

D.4 Proof of Theorem 4.4

We make use of the following lemmas.

Lemma D.1 (Zeng et al. (2018)). *Logarithm and polynomials are Kurdyka–Łojasiewicz (KL) functions. Moreover, sums, products, compositions, and quotients (with denominator bounded away from 0) of KL functions are also KL.*

Lemma D.2 (Guo et al. (2017)). *Consider the optimization problem:*

$$\begin{aligned} &\underset{\tilde{\boldsymbol{\beta}}, \boldsymbol{\pi}}{\text{minimize}} && g(\boldsymbol{\pi}) \\ &\text{subject to} && \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}} = \boldsymbol{\pi}, \end{aligned} \tag{47}$$

and solve Eq. (47) *via Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2011). Let* $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^k)\}_{k\in\mathbb{N}}$ *be the sequence generated by the ADMM algorithm, and* ρ *be the penalty parameter of ADMM. Suppose that there exists* $\kappa > 0$ *such that* $\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}} \succeq \kappa \boldsymbol{I}$*, and the sequence* $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^k)\}_{k\in\mathbb{N}}$ *is bounded.*

If there exist solutions for the minimization steps of ADMM w.r.t. both $\boldsymbol{\pi}$ *and* $\tilde{\boldsymbol{\beta}}$*,* $g(\boldsymbol{\pi})$ *is a continuous differentiable function with an L-Lipschitz continuous gradient at* $\boldsymbol{\pi}^k$*,* $k \in \mathbb{N}$ *where* $L > 0$*, and the augmented Lagrangian of Eq.* (47) *is a KL function, then, for* $\rho > 2L$*,* $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^k)\}_{k\in\mathbb{N}}$ *converges to a point that satisfies the Karush-Kuhn-Tucker (KKT) conditions of Eq.* (47)*.*

To begin with, there exist solutions for the minimization steps in (13): $\tilde{\boldsymbol{\beta}}$ update has the closed form solution given by Eq. (13a) and $\boldsymbol{\pi}$ update admits a minimizer for large enough ρ by Lemma 4.3.

By Assumption 4.1, $\nabla_{\boldsymbol{\pi}}\mathcal{L}$ given by Eq. (23) exists, i.e. $\mathcal{L}$ is continuous differentiable at $\boldsymbol{\pi}^k$, $k \in \mathbb{N}$ generated by (13b). Let $\nabla^2(\mathcal{L})$ be the Hessian of $\mathcal{L}$. Differentiating Eq. (23) w.r.t. π_j, $\nabla^2(\mathcal{L})$ has the following form:

$$\begin{aligned} &\nabla^2_{ij}(\mathcal{L}) \\ &= \begin{cases} \sum_{\ell \in W_i} \frac{1}{\pi_i^2} - \sum_{\ell | i \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2}, & i = j \\ - \sum_{\ell | i,j \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2}, & i \neq j. \end{cases} \end{aligned} \tag{48}$$

Consider $L = \frac{\max_i |W_i|}{\epsilon^2}$, where W_i is the set of observations where sample $i \in \mathcal{N}$ is chosen. By Assumption 4.1, we have:

$$\begin{aligned} L = \frac{\max_i |W_i|}{\epsilon^2} &\geq \sum_{\ell \in W_i} \frac{1}{\pi_i^2} \ \forall i \in \mathcal{N}, \\ \Leftrightarrow \ L - \sum_{\ell \in W_i} \frac{1}{\pi_i^2} + \sum_{\ell | i \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2} &\geq \sum_{\ell | i \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2} \ \forall i \in \mathcal{N}, \\ \Leftrightarrow \ L - \sum_{\ell \in W_i} \frac{1}{\pi_i^2} + \sum_{\ell | i \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2} &\geq \sum_{j \neq i} \sum_{\ell | i,j \in A_\ell} \frac{1}{(\sum_{t\in A_\ell} \pi_t)^2} \ \forall i \in \mathcal{N}. \end{aligned} \tag{49}$$

Now, consider the matrix $L\boldsymbol{I}_{n\times n} - \nabla^2(\mathcal{L})$. By Eq. (49), $L\boldsymbol{I}_{n\times n} - \nabla^2(\mathcal{L})$ is diagonally dominant (c.f. Definition D.1) and all of its diagonal elements are positive, i.e., $\nabla^2(\mathcal{L})$ is upper bounded by $L\boldsymbol{I}_{n\times n}$. Thus, the objective function of Eq. (11), i.e. $\mathcal{L}$, has an L-Lipschitz continuous gradient at $\boldsymbol{\pi}^k$, $k \in \mathbb{N}$, where $L = \frac{\max_i |W_i|}{\epsilon^2} > 0$.

Moreover, the augmented Lagrangian given by Eq. (29) is a sum of three functions: logarithm of the ratio of two polynomials where the denominator is bounded away from 0 for all $\boldsymbol{\pi}^k$, $k \in \mathbb{N}$ by Assumption 4.1, and two other polynomial functions. By Lemma D.1, these three functions and their sum is KL on the set $\{\boldsymbol{\pi}^k \,|\, \pi_i^k > \epsilon \,, i \in \mathcal{N}, k \in \mathbb{N}\}$. As a result, the augmented Lagrangian of Eq. (11) is a KL function. Putting it all together, by Lemma D.2, for $\rho > \frac{2\max_i |W_i|}{\epsilon^2}$, the sequence $\{(\boldsymbol{\pi}^k, \boldsymbol{u}^k, \tilde{\boldsymbol{\beta}}^{k+1})\}_{k\in\mathbb{N}}$ generated by (13) converges to a point that satisfies the KKT conditions (Nocedal and Wright, 2006) of Problem (11).

E Extension to the Logistic Case

We describe here how to apply our approach to regress model parameters in the logistic case. Recall that Problem (5) is, in this case, convex, and can thus be solved by Newton's method. Nevertheless, we would like to accelerate its computation via a spectral method akin to ILSR. Following the steps we took in the affine case, we re-write (5) as:

$$\text{Minimize} \quad \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) \tag{50a}$$

$$\text{subject to:} \quad \log \boldsymbol{\pi} = \boldsymbol{X}\boldsymbol{\beta}, \quad \boldsymbol{\pi} \geq \mathbf{0}, \tag{50b}$$

where $\log \boldsymbol{\pi} = [\log \pi_i]_{i\in\mathcal{N}}$ is the $\mathbb{R}^n$ vector generated by applying log to $\boldsymbol{\pi}$ element-wise. The augmented Lagrangian corresponding to Eq. (50) is:

$$\begin{aligned} L_\rho(\boldsymbol{\beta}, \boldsymbol{\pi}, \boldsymbol{u}) = \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) \\ + \frac{\rho}{2} \|\boldsymbol{X}\boldsymbol{\beta} - \log \boldsymbol{\pi} + \boldsymbol{u}\|_2^2 - \frac{\rho}{2} \|\boldsymbol{u}\|_2^2, \end{aligned} \tag{51}$$

Algorithm 2 PLADMM-log

procedure ADMM($\boldsymbol{X}$, $\mathcal{D} = \{(c_\ell, A_\ell) \,|\, \ell \in \mathcal{M}\}$, ρ)
  Initialize $\boldsymbol{\beta}$ via Eq. (55); $\boldsymbol{\pi} \leftarrow [e^{\boldsymbol{x}_i^T \boldsymbol{\beta}}]_{i \in \mathcal{N}}$; $\boldsymbol{u} \leftarrow \boldsymbol{0}$
  repeat
    $\boldsymbol{\pi} \leftarrow \text{ILSRX}(\rho, \boldsymbol{\pi}, \boldsymbol{X}, \boldsymbol{\beta}, \boldsymbol{u})$
    $\boldsymbol{u} \leftarrow \boldsymbol{u} + \boldsymbol{X}\boldsymbol{\beta} - \log \boldsymbol{\pi}$
    $\boldsymbol{\beta} \leftarrow (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T (\log \boldsymbol{\pi} - \boldsymbol{u})$
  until convergence
return $\boldsymbol{\beta}$, $\boldsymbol{\pi}$
end procedure
procedure ILSRX($\rho, \boldsymbol{\pi}, \boldsymbol{X}, \boldsymbol{\beta}, \boldsymbol{u}$)
  repeat
    $\sigma_i \leftarrow \rho \frac{(\log \pi_i - \boldsymbol{x}_i^T \boldsymbol{\beta} - u_i)}{\pi_i}, \quad i \in \mathcal{N}$
    Calculate $\boldsymbol{M}(\boldsymbol{\pi}) = [\mu_{ji}(\boldsymbol{\pi})]_{i,j \in \mathcal{N}}$ via Eq. (16)
    $\boldsymbol{\pi} \leftarrow \text{ssd}\,(\boldsymbol{M}(\boldsymbol{\pi}))$
  until convergence
return $\boldsymbol{\pi}$
end procedure

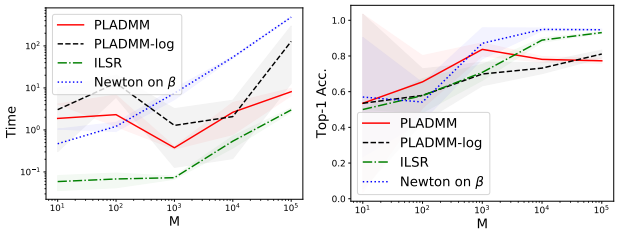


Figure 2: Convergence time (Conv. Time) and top-1 test accuracy (Top-1 Acc.) of PLADMM, PLADMM-log, ILSR, and Newton on $\boldsymbol{\beta}$ evaluated on synthetic datasets vs. the number of observations $M \in \{10, 100, 1000, 10000, 100000\}$. Observations are partitioned w.r.t. observation CV (c.f. Sec. 5), where number of samples is $n = 1000$, number of features is $p = 100$, and query size is $|A_\ell| = 2$.

and applying ADMM on problem (50) yields:

$$\boldsymbol{\beta}^{k+1} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min}\, L_\rho(\boldsymbol{\beta}, \boldsymbol{\pi}^k, \boldsymbol{u}^k) = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T (\log \boldsymbol{\pi}^k - \boldsymbol{u}^k), \tag{52a}$$

$$\boldsymbol{\pi}^{k+1} = \underset{\boldsymbol{\pi} \in \mathbb{R}^n_+}{\arg\min}\, \mathcal{L}(\mathcal{D} \,|\, \boldsymbol{\pi}) + \frac{\rho}{2} \| \boldsymbol{X}\boldsymbol{\beta}^{k+1} - \log \boldsymbol{\pi} + \boldsymbol{u}^k \|_2^2, \tag{52b}$$

$$\boldsymbol{u}^{k+1} = \boldsymbol{u}^k + \boldsymbol{X}\boldsymbol{\beta}^{k+1} - \log \boldsymbol{\pi}^{k+1}. \tag{52c}$$

Mutatis mutandis, following the same manipulations in Lemma 4.1, a stationary point of the objective in each step (52b) can be cast as the stationary distribution of the continuous-time MC with transition rates $\mu_{ji}(\boldsymbol{\pi})$, $i, j \in \mathcal{N}$, given by Eq. (16), the only difference being that vector $\boldsymbol{\sigma} = [\sigma_i]_{i \in \mathcal{N}}$ is now given by:

$$\sigma_i = \rho \frac{(\log \pi_i - \boldsymbol{x}_i^T \boldsymbol{\beta} - u_i)}{\pi_i}, \quad i \in \mathcal{N}. \tag{53}$$

Having adjusted the transition matrix $\boldsymbol{M}(\boldsymbol{\pi})$ thusly, $\boldsymbol{\pi}$ can again be obtained by repeated iterations of (18).

The resulting algorithm, which we refer to as Plackett-Luce ADMM-log (PLADMM-log), is summarized in Algorithm 2; the algorithm is almost identical to Algorithm 1, using $\log \boldsymbol{\pi}$ instead of $\boldsymbol{\pi}$, defining $\boldsymbol{\sigma}$ via (53), and having a different initialization. We discuss the latter below.

Initialization. Similar to the initialization of PLADMM (c.f. Eq. (20)), we initialize $\boldsymbol{\beta}$ so that the initial scores obey the Plackett-Luce model, mirroring the approach by Saha and Rajkumar (2018). Defining P_{ij}, $i, j \in \mathcal{N}$ the same way, and using the logistic parametrization in Sec.3, we have that:

$$\frac{P_{ij}}{P_{ji}} = \frac{\pi_i}{\pi_j} = e^{\boldsymbol{\beta}^T (\boldsymbol{x}_i - \boldsymbol{x}_j)}. \tag{54}$$

Accordingly, we initialize $\boldsymbol{\beta}$ as:

$$\boldsymbol{\beta}^0 = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \sum_{(i,j) \in \mathcal{D}} \left(\boldsymbol{\beta}^T (\boldsymbol{x}_i - \boldsymbol{x}_j) - \log \left(\frac{\hat{P}_{ij}}{\hat{P}_{ji}} \right) \right)^2, \tag{55}$$

where $\hat{P}_{ij}$, $i, j \in \mathcal{N}$, are again empirical estimates obtained from dataset $\mathcal{D}$. Given $\boldsymbol{\beta}^0$, we generate the initial Plackett-Luce scores via the logistic parametrization $\boldsymbol{\pi}^0 = [e^{\boldsymbol{x}_i^T \boldsymbol{\beta}^0}]_{i \in \mathcal{N}}$. Finally, we initialize the dual variable as $\boldsymbol{u}^0 = \boldsymbol{0}$.

F Experiments

F.1 Datasets

Synthetic Datasets. We generate the feature vectors $\boldsymbol{x}_i \in \mathbb{R}^p, \quad i \in \mathcal{N}$ from $\mathcal{N}(\boldsymbol{0}, \sigma_x^2 \boldsymbol{I}_{p \times p})$ and a common parameter vector $\boldsymbol{\beta} \in \mathbb{R}^p$ from $\mathcal{N}(\boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}_{p \times p})$. Then, we generate the Plackett-Luce scores via the logistic parametrization $\boldsymbol{\pi} = [e^{\boldsymbol{x}_i^T \boldsymbol{\beta}}]_{i \in \mathcal{N}}$. We normalize the resulting scores, so that $\boldsymbol{1}^\top \boldsymbol{\pi} = 1$. We set $\sigma_x^2 = \sigma_\beta^2 = 0.8$ in all experiments. Given $\boldsymbol{\pi}$, we generate each observation in $\mathcal{D}$ as follows: we first select $|A_\ell| = 2$ samples out of n samples uniformly at random. Then, we generate the choice c_l, $l \in \mathcal{M}$ from the Plackett-Luce model given by Eq. (1).

Filter Aesthetic Comparison (FAC). The Filter Aesthetic Comparison (FAC) dataset (Sun et al., 2017) contains 1280 unfiltered images pertaining to 8 different categories. Twenty-two different image filters are applied to each image. Labelers are provided with two filtered images and are asked to identify which image has better quality. We select $n = 1000$ images within one category, as only the filtered image pairs that are within the same category are compared. The resulting dataset contains $M = 728$ pairwise comparisons. Moreover, for each image, we extract features via a

state-of-the-art convolutional neural network architecture, namely GoogLeNet (Szegedy et al., 2015), with weights pre-trained on the ImageNet dataset (Deng et al., 2009). We select $p = 50$ of these features by Principal Component Analysis (Jolliffe, 1986).

Retinopathy of Prematurity (ROP). The Retinopathy of Prematurity (ROP) dataset contains $n = 100$ retina images with $p = 143$ features (Ataer-Cansızoğlu, 2015). Experts are provided with two images and are asked to choose the image with higher severity of the ROP disease. Five experts independently label 5941 image pairs; the resulting dataset contains $M = 29705$ pairwise comparisons. Note that some pairs are labelled more than once by different experts.

SUSHI. The SUSHI Preference dataset (Kamishima et al., 2009) contains $n = 100$ sushi ingredients with $p = 18$ features. Each of the 5000 customers independently ranks 10 ingredients according to her preferences. We select the rankings provided by 10 customers, where an ingredient is ranked higher if it precedes the other ingredients in a customer's ranked list. We generate two datasets: triplet Sushi containing $M = 1200$ rankings of $|A_\ell| = 3$ ingredients, and pairwise Sushi containing $M = 450$ pairwise comparisons.

F.2 Algorithms

We implement four algorithms that regress Plackett-Luce scores from features, which we call as *feature methods*.

PLADMM. PLADMM solves the problem in Eq. (11) and is summarized in Algorithm 1. We compute the stationary distribution at each iteration of ILSRX (c.f. Eq. (18)) using the power method (Lei et al., 2016). As the stopping criterion, we use $\|\boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1}\|_2 < r_{\texttt{tol}} \| \boldsymbol{\pi}^k\|_2$ and $\|\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^k - \tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^{k-1}\|_2 < r_{\texttt{tol}} \|\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}^k\|_2$. We set the relative tolerance $r_{\texttt{tol}} = 10^{-4}$ for all experiments. We use the same relative tolerance for the stopping criterion of the power method. We set $\rho = 1$ in our experiments, which is a standard choice in the ADMM literature (Boyd et al., 2011). In our experiments, we consistently observe that Eq.(37) is satisfied. That is why, we use $c_j = \frac{-\pi_j \sigma_j}{\sum_{i \in \mathcal{N}_-} \pi_i \sigma_i}$, $j \in \mathcal{N}_+$ instead of Eq.(41) to calculate the transition rates (16).

PLADMM-log. PLADMM-log solves the problem in Eq. (50) and is summarized in Algorithm 2. As the stopping criterion, we use $\|\boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1}\|_2 < r_{\texttt{tol}} \| \boldsymbol{\pi}^k\|_2$ and $\|e^{\boldsymbol{X}\boldsymbol{\beta}^k} - e^{\boldsymbol{X}\boldsymbol{\beta}^{k-1}}\|_2 < r_{\texttt{tol}} \|e^{\boldsymbol{X}\boldsymbol{\beta}^k}\|_2$, where exponentiation is applied elementwise.

SLSQP. SLSQP solves the problem in Eq. (4) via the sequential least-squares quadratic programming (SLSQP) algorithm (Nocedal and Wright, 2006). We initialize SLSQP the same as PLADMM (c.f. Algorithm 1). As stopping criterion, we use $\| \boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1} \|_2 < r_{\texttt{tol}} \| \boldsymbol{\pi}^k \|_2$, where $\boldsymbol{\pi}^k = \boldsymbol{X}\boldsymbol{\beta}^k + b^k\mathbf{1}$, $k \in \mathbb{N}$. Each iteration of SLSQP is $O\Big(\sum_{\ell \in \mathcal{D}} \big(|A_\ell|(p+1)\big) + (p+1)^2 \Big)$ for constructing the gradient of Eq. (3) w.r.t. $\tilde{\boldsymbol{\beta}}$ and updating $\tilde{\boldsymbol{\beta}}$, respectively.

Newton on $\boldsymbol{\beta}$. Newton on $\boldsymbol{\beta}$ solves the convex problem in Eq. (5) via Newton's method (Nocedal and Wright, 2006). We initialize Newton on $\boldsymbol{\beta}$ the same as PLADMM-log (c.f. Algorithm 2). As stopping criterion, we use $\| \boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1} \|_2 < r_{\texttt{tol}} \| \boldsymbol{\pi}^k \|_2$, where $\boldsymbol{\pi}^k = [e^{\boldsymbol{x}_i^T \boldsymbol{\beta}^k}]_{i \in \mathcal{N}}$, $k \in \mathbb{N}$. Each iteration of Newton on $\boldsymbol{\beta}$ is $O\Big(\sum_{\ell \in \mathcal{D}} \big(|A_\ell|\, p^2\big) + p^2 \Big)$ for constructing the Hessian of Eq. (3) w.r.t. $\boldsymbol{\beta}$ and updating $\boldsymbol{\beta}$, respectively.

We implement three algorithms that learn the Plackett-Luce scores from the choice observations alone, which we call as *featureless methods*.

ILSR. Iterative Luce Spectral Ranking (ILSR) algorithm solves the problem in Eq. (6) and is described by the iterations in Eq.(10). We initialize ILSR with $\boldsymbol{\pi}^0 = \frac{1}{n}\mathbf{1}$. We compute the stationary distribution at each iteration of ILSR using the power method. As the stopping criterion, we use $\|\boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1}\|_2 < r_{\texttt{tol}} \|\boldsymbol{\pi}^k\|_2$. Each iteration of ILSR is $O\Big(\sum_{\ell \in \mathcal{D}} \big(|A_\ell|\big) + n^2 \Big)$ for constructing the transition matrix $\boldsymbol{\Lambda}(\boldsymbol{\pi})$ (c.f. Eq.(9)) and finding the stationary distribution $\boldsymbol{\pi}$, respectively.

MM. The Minorization-Maximization (MM) algorithm (Hunter, 2004) solves the problem in Eq. (6). We initialize MM with $\boldsymbol{\pi}^0 = \frac{1}{n}\mathbf{1}$. As the stopping criterion, we use $\|\boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1}\|_2 < r_{\texttt{tol}} \|\boldsymbol{\pi}^k\|_2$. Each iteration of MM is $O\Big(\sum_{\ell \in \mathcal{D}} \big(|A_\ell|\big) \Big)$.

Newton on $\boldsymbol{\theta}$. Newton on $\boldsymbol{\theta}$ algorithm solves the problem in Eq. (6) by reparametrizing the scores as $\pi_i = e^{\theta_i}$, $i \in \mathcal{N}$. It solves the resulting convex problem by Newton's method (Nocedal and Wright, 2006). We initialize Newton on $\boldsymbol{\theta}$ with $\boldsymbol{\theta}^0 = [\theta_i^0]_{i \in \mathcal{N}} = \mathbf{0}$. As stopping criterion, we use $\|\boldsymbol{\pi}^k - \boldsymbol{\pi}^{k-1}\|_2 < r_{\texttt{tol}} \|\boldsymbol{\pi}^k\|_2$, where $\pi_i^k = e^{\theta_i^k}$, $i \in \mathcal{N}$, $k \in \mathbb{N}$. Each iteration of Newton on $\boldsymbol{\theta}$ is $O\Big(\sum_{\ell \in \mathcal{D}} \big(|A_\ell|^2\big) + n^2 \Big)$ for constructing the Hessian of Eq. (3) w.r.t. $\boldsymbol{\theta}$ and updating $\boldsymbol{\theta}$, respectively.

F.3 Top-1 Accuracy and Kendall-Tau Correlation

We measure the prediction performance by Top-1 accuracy (Top-1 Acc.) and Kendall-Tau correlation (KT) on the test set. Let the test set be $\mathcal{D}_{\texttt{choice}} = \{(c_\ell, A_\ell) \,|\, \ell \in \{1, ..., M_{\texttt{test}}\}\}$ for the choice setting and $\mathcal{D}_{\texttt{rank}} = \{(\alpha^\ell, A_\ell) \,|\, \ell \in \{1, ..., M_{\texttt{test}}\}\}$ for the rank-

ing setting, where $\alpha^\ell = \alpha^\ell_1 \succ \alpha^\ell_2 \succ \cdots \succ \alpha^\ell_{|A_\ell|}$ is an ordered sequence of the samples in A_ℓ. For both settings, given A_ℓ, we predict the ℓ-th choice as $\hat{c}_\ell = \arg\max_{i \in A_\ell} \pi_i$. We calculate the Top-1 accuracy (Top-1 Acc.) for the choice setting as:

$$\text{Top-1 Acc.} = \frac{\sum_{\ell=1}^{M_{\texttt{test}}} \mathbb{1}(\hat{c}_\ell = c_\ell)}{M_{\texttt{test}}} \in [0, 1], \tag{56}$$

and for the ranking setting as:

$$\text{Top-1 Acc.} = \frac{\sum_{\ell=1}^{M_{\texttt{test}}} \mathbb{1}(\hat{c}_\ell = \alpha^\ell_1)}{M_{\texttt{test}}} \in [0, 1]. \tag{57}$$

For the ranking setting, given A_ℓ, we also predict the ranking as $\hat{\alpha}^\ell = \arg\text{sort}[\pi_i]_{i \in A_\ell}$, i.e. sequence of the samples in A_ℓ ordered w.r.t. their scores. We calculate Kendall-tau correlation (KT) (Kendall, 1938) as a measure of the correlation between each true ranking α^ℓ and predicted ranking $\hat{\alpha}^\ell$, $\ell \in \{1, ..., M_{\texttt{test}}\}$. For observation ℓ, let $T_\ell = \sum_{t=1}^{|A_\ell|} \sum_{s=1}^{|A_\ell|} \mathbb{1}(\hat{\alpha}^\ell_t \succ \hat{\alpha}^\ell_s \wedge \alpha^\ell_t \succ \alpha^\ell_s)$ be the number correctly predicted ranking positions, and $F_\ell = \sum_{t=1}^{|A_\ell|} \sum_{s=1}^{|A_\ell|} \mathbb{1}(\hat{\alpha}^\ell_t \succ \hat{\alpha}^\ell_s \wedge \alpha^\ell_s \succ \alpha^\ell_t)$ be the number incorrectly predicted ranking positions. Then, KT is computed by:

$$\text{KT} = \frac{\sum_{\ell=1}^{M_{\texttt{test}}} (T_\ell - F_\ell) / \binom{|A_\ell|}{2}}{M_{\texttt{test}}} \in [-1, 1], \tag{58}$$

where $\binom{|A_\ell|}{2}$ is the number of sample pairs in a query of size $|A_\ell|$.

F.4 Impact of Number of Observations

Fig. 2 shows the convergence time (Time) and top-1 test accuracy (Top-1 Acc.) of PLADMM, PLADMM-log, ILSR, and Newton on $\boldsymbol{\beta}$ when trained on synthetic datasets with number of observations $M \in \{10, 100, 1000, 10000, 100000\}$. Observations are partitioned w.r.t. observation CV (c.f. Sec. 5), where number of samples is $n = 1000$, number of parameters is $p = 100$, and size of each query is $|A_\ell| = 2$. As $n > p$, PLADMM benefits from being able to regress n scores from a smaller number of p parameters and leads to significantly better Top-1 Acc compared to ILSR in Fig. 2. Especially when M is not enough to learn $n = 1000$ scores, but to learn $p = 100$ parameters, PLADMM gains the most performance advantage over ILSR, up to 13% Top-1 Acc. Moreover, PLADMM and PLADMM-log are consistently faster than Newton on $\boldsymbol{\beta}$, for all number of observations $M > 100$. Particularly, for $M = 100000$, PLADMM and PLADMM-log converge $4 - 60$ times faster than Newton on $\boldsymbol{\beta}$.

Dataset	Method	Training Metrics		Performance Metrics on the Test Set	
		Time (s) ↓	Iter. ↓	Top-1 Acc. ↑	KT ↑
FAC	**PLADMM**	0.352 ± 0.044	4 ± 0	0.68 ± 0.048	0.35 ± 0.089
	PLADMM-log	0.17 ± 0.033	4 ± 0	0.691 ± 0.054	0.378 ± 0.11
	ILSR (no $\boldsymbol{X}$)	0.066 ± 0.012	2 ± 0	0.591 ± 0.067	-0.13 ± 0.164
	MM (no $\boldsymbol{X}$)	10.7 ± 0.501	500 ± 0	0.544 ± 0.046	0.046 ± 0.087
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	9.152 ± 1.284	17 ± 3	0.5 ± 0.0	0.0 ± 0.0
	Newton on $\boldsymbol{\beta}$	1.531 ± 0.169	6 ± 1	0.701 ± 0.04	0.398 ± 0.08
	SLSQP	22.73 ± 19.151	160 ± 135	0.689 ± 0.063	0.375 ± 0.125
ROP	**PLADMM**	1.953 ± 0.217	4 ± 0	0.896 ± 0.005	0.791 ± 0.009
	PLADMM-log	0.359 ± 0.027	1 ± 0	0.904 ± 0.005	0.807 ± 0.01
	ILSR (no $\boldsymbol{X}$)	0.716 ± 0.058	2 ± 0	0.891 ± 0.005	0.781 ± 0.009
	MM (no $\boldsymbol{X}$)	356.497 ± 29.11	500 ± 0	0.905 ± 0.004	0.81 ± 0.008
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	85.42 ± 6.849	9 ± 0	0.906 ± 0.004	0.811 ± 0.008
	Newton on $\boldsymbol{\beta}$	55.718 ± 6.293	2 ± 0	0.904 ± 0.005	0.808 ± 0.009
	SLSQP	9.595 ± 7.136	2 ± 1	0.683 ± 0.049	0.366 ± 0.098
Pairwise Sushi	**PLADMM**	0.061 ± 0.002	4 ± 0	0.669 ± 0.034	0.338 ± 0.068
	PLADMM-log	0.764 ± 1.192	58 ± 30	0.634 ± 0.075	0.267 ± 0.15
	ILSR (no $\boldsymbol{X}$)	0.027 ± 0.003	2 ± 0	0.763 ± 0.039	0.521 ± 0.084
	MM (no $\boldsymbol{X}$)	5.191 ± 0.345	490 ± 31	0.773 ± 0.048	0.543 ± 0.094
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	2.342 ± 0.689	18 ± 5	0.735 ± 0.095	0.465 ± 0.185
	Newton on $\boldsymbol{\beta}$	0.176 ± 0.17	2 ± 2	0.685 ± 0.044	0.369 ± 0.087
	SLSQP	16.198 ± 8.728	245 ± 134	0.64 ± 0.06	0.28 ± 0.119
Triplet Sushi	**PLADMM**	0.127 ± 0.007	4 ± 0	0.569 ± 0.035	0.218 ± 0.045
	PLADMM-log	0.804 ± 0.349	36 ± 18	0.487 ± 0.034	0.19 ± 0.072
	ILSR (no $\boldsymbol{X}$)	0.054 ± 0.003	2 ± 0	0.678 ± 0.036	0.454 ± 0.06
	MM (no $\boldsymbol{X}$)	15.349 ± 0.617	500 ± 0	0.715 ± 0.035	0.522 ± 0.059
	Newton on $\boldsymbol{\theta}$ (no $\boldsymbol{X}$)	5.122 ± 0.34	14 ± 1	0.73 ± 0.036	0.496 ± 0.089
	Newton on $\boldsymbol{\beta}$	1.12 ± 0.659	3 ± 2	0.605 ± 0.058	0.285 ± 0.062
	SLSQP	21.738 ± 39.761	107 ± 197	0.521 ± 0.043	0.191 ± 0.059

Table 4: Evaluations on real datasets partitioned w.r.t. observation CV (c.f. Sec. 5). We report the convergence time in seconds (Time), number of iterations until convergence (Iter), top-1 accuracy on the test set (Top-1 Acc.), and Kendall-Tau correlation on the test set (KT). ILSR, MM, and Newton on $\boldsymbol{\theta}$ learn the Plackett-Luce scores $\boldsymbol{\pi}$ from the choice observations alone and do not use the features $\boldsymbol{X}$. Newton on $\boldsymbol{\beta}$ and sequential least squares quadratic programming (SLSQP) regress $\boldsymbol{\pi}$ from $\boldsymbol{X}$. (c.f. Sec. F.2).