# Near-Neighbor Methods in Random Preference Completion

**Ao Liu**
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12180, USA
liua6@rpi.edu

**Qiong Wu, Zhenming Liu**
Department of Computer Science
College of William and Mary
Williamsburg, VA 23187, USA
qwu05@email.wm.edu and zliu@cs.wm.edu

**Lirong Xia**
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12180, USA
xial@cs.rpi.edu

## Abstract

This paper studies a stylized, yet natural, learning-to-rank problem and points out the critical incorrectness of a widely used nearest neighbor algorithm. We consider a model with $n$ agents (users) $\{x_i\}_{i\in[n]}$ and $m$ alternatives (items) $\{y_l\}_{l\in[m]}$, each of which is associated with a latent feature vector. Agents rank items nondeterministically according to the Plackett-Luce model, where the higher the utility of an item to the agent, the more likely this item will be ranked high by the agent. Our goal is to identify near neighbors of an arbitrary agent in the latent space for prediction.

We first show that the Kendall-tau distance based kNN produces incorrect results in our model. Next, we propose a new anchor-based algorithm to find neighbors of an agent. A salient feature of our algorithm is that it leverages the rankings of many other agents (the so-called "anchors") to determine the closeness/similarities of two agents. We provide a rigorous analysis for one-dimensional latent space, and complement the theoretical results with experiments on synthetic and real datasets. The experiments confirm that the new algorithm is robust and practical.

## 1 Introduction

In a learning-to-rank problem, there is a set of agents (users) $\mathcal{X} = \{x_1, \ldots x_n\}$ and a set of alternatives (items) $\mathcal{Y} = \{y_1, \ldots y_m\}$. Each agent reveals her preferences over a subset of alternatives. The goal is to infer agents' preferences over all alternatives, including those that are not rated or ranked. This fundamental machine learning problem has many practical applications. For example, recommender systems use an agent' revealed preferences to discover other alternatives she might be interested in; product designers learn from consumers' past choices to estimate the demand curve of a new product; defenders can predict terrorists' preferences based on their past behavior; and political parties can evaluate campaign options based on voters' preferences. See (Liu and others 2009) for a recent survey.

**Rating vs. ranking.** Agents' preferences can be represented by either a *rating* for each alternative (e.g., an integer rating in Netflix), or a *ranking* over the alternatives (i.e., complete ordering). Rating-based approaches have many known drawbacks (Liu and Yang 2008; Katz-Samuels and Scott 2018), including (i) agents often have different scales for ratings; and (ii) numeric values are often less robust than ranking-based approaches. In fact, rating data can always be converted to ranking data (e.g., $y_1$ is ranked higher than $y_2$ if $y_1$ has a higher rating) and thus ranking-based models and algorithms are more general. We focus on ranking data.

A common approach to infer an agent's preference is to first identify near neighbors of the agent in terms of the *Kendall-Tau* (KT) distance, then aggregate their rankings to produce a prediction. The KT distance is a metric that counts the number of pairwise disagreements between two ranking lists. This approach was proposed by Liu and Yang (2008), and their algorithm will be referred to as KT-kNN in this paper. Many subsequent work are based on the following assumption (Hwang and Lee 2009; Wang et al. 2012; Fan and Lin 2013; Wang et al. 2014; Park et al. 2015).

**Assumption 1.** KT distance is a good measure of similarity between agents.

No theoretical justification for this assumption was known until recently. Katz-Samuels and Scott (2018) proposed a latent utility model to justify the assumption. In their model, each agent or alternative is associated with a latent feature. Alternative $j$'s utility to agent $i$ is controlled by a deterministic function of the similarity in their latent features. Under this model, consistency result is established for the KT-kNN algorithms.

However, this model assumes that agents' preferences are deterministic, which is unrealistic in many settings. For example, an agent can exhibit irrational behavior, or provide only a noisy version of her preferences. In fact, human preferences are often highly non-deterministic. Various statistical models have been built to model such randomness, pioneered by the Nobel Laureate McFadden (McFadden 2000) among many other researchers. Therefore, the following question remains open.

*How can we learn an agent's random preferences from other agents' random preferences?*

This question can be answered by designing algorithms for two closely-related problems: *(i) preference completion (PC):* given each agent's preferences over a subset of alternatives, the goal is to estimate its preference over all the alternatives. *(ii) near neighbors (NN):* given an agent $x_i$, the goal is to find agents $x_{i'}$ close to $x_i$ in the latent space.

Standard techniques exist to use algorithms for the NN problem to solve a PC problem ( (Katz-Samuels and Scott 2018; Liu and Yang 2008); see also Appendix D). Therefore, we focus on the NN problem in this paper.

**Our Contributions.** Our main conceptual contribution is the combination of a distance-based latent model and random preferences for learning to rank. To the best of our knowledge, while there is a large literature in each component, we are the first to consider both. See related work for more discussions.

Our model is called *distance-based random preference model*. Let the latent feature of agent $i$ (alternative $j$) be $x_i$ ($y_j$). Agent $i$'s preferences are determined by a utility function $u(x_i, y_j) = \theta(x_i, y_j) + \epsilon_{i,j}$, where $\theta(x_i, y_j)$ is a deterministic monotonically decreasing distance-based function and $\epsilon_{i,j}$ is a zero mean independent random variable. Our model captures two pervasive characteristics of ranking datasets: *Ch1. Economically meaningful $\theta(\cdot, \cdot)$ function.* $u(x_i, y_j)$ is high in expectation when $x_i$ and $y_j$ are close. An agent is more likely to prefer alternatives with similar latent features to itself. *Ch2. Random preference model.* The function $u(x_i, y_j)$ contains a noise term $\epsilon_{i,j}$ to capture uncertainties in agents' behaviors.

Our technical contributions are two-fold. First, we prove that Assumption 1 does not hold anymore in our distance-based random preference model. More precisely, we prove that the agents found by the KT-kNN algorithm (Liu and Yang 2008) is far away from the given agents with high probability, even when $n, m \to \infty$.

Second, we design an "anchor-based" algorithm for finding an agent's near neighbors under random preferences. The algorithm is based on the following natural idea: if two agents $i_1$ and $i_2$ are close, then their KT distance to any other agent $j$ (an anchor) should also be close. The algorithm proceeds by using the KT distance to other agents as an agent's feature, and measures the closeness between two agents by the $L_1$ distance of their features. We prove that asymptotically our algorithm identifies an agent's near neighbors with high probability when the latent space is 1-dimensional. Many techniques we developed can be generalized to high-dim settings.

Experiments on synthetic data verify our theoretical findings, and demonstrate that our algorithm is robust in high-dim spaces. Experiments on Netflix data shows that our anchor-based algorithm is superior to the KT-kNN algorithm and a standard collaborative filter (using the cosine-similarities to determine neighbors).

**Related Work and Discussions.** While using random utility models in learning-to-rank problems is not new (Lu and Negahban 2015; Park et al. 2015; Oh, Thekumparampil, and Xu 2015; Zhao, Piech, and Xia 2016; Zhao, Villamil, and Xia 2018; Liu et al. 2019; Katz-Samuels and Scott 2018), we are not aware of any that simultaneously achieves both *Ch1* and *Ch2*.

Random utility-based ranking algorithms (Lu and Negahban 2015; Park et al. 2015; Oh, Thekumparampil, and Xu 2015) address *Ch2*, but the function $\theta(x_i, y_j)$ often does not have an explicit economics interpretation. For example, let $\Theta \in \mathbf{R}^{n \times m}$ be a matrix such that $\Theta_{i,j} = \theta(x_i, y_j)$. (Park et al. 2015; Oh, Thekumparampil, and Xu 2015) assume that $\Theta$ is low rank. But the low rank assumption does not have explicit economically interpretation.

While recent non-parametric models (e.g., (Katz-Samuels and Scott 2018)) allow one to use economically interpretable functions $\theta$ (addressing *Ch1*), they operate only under deterministic utility models.

Parametric preference learning has been extensively studied in machine learning, especially learning to rank (Azari Soufiani et al. 2013; Azari Soufiani, Parkes, and Xia 2013; 2014; Cheng, Hüllermeier, and Dembczynski 2010; Hughes, Hwang, and Xia 2015; Khetan and Oh 2016; Maystre and Grossglauser 2015). These works are different with ours as it is often assumed that agents' preferences are generated from a parametric model.

## 2 Preliminaries

**Distance-Based Random Preference Model.** Let $\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbf{R}^d$ denote the set of agents and let $\mathcal{Y} = \{y_1, \ldots, y_m\} \subset \mathbf{R}^d$ denote the set of alternatives. We slightly abuse the notation and use $x_i$ to refer to both agent $i$ and her latent features. Each agent $x_i$ has a ranking (preference list) $R_i = [y_{j_1} \succ \cdots \succ y_{j_m}]$ over $\mathcal{Y}$, where $\succ$ means "prefer to". We observe only a subset of $R_i$ for each $i \in [n]$.

*Utility functions and the random utility model.* Agent $i$'s expected utility on alternative $j$ is determined by a function $\theta(x_i, y_j)$. Throughout this paper, we use $\theta(x_i, y_j) = \exp(-\|x_i - y_j\|_2)$, where $\|.\|_2$ is the $\ell_2$-norm.

Agent $i$'s ranking $R_i$ is determined by the widely-used *Plackett-Luce model* (Plackett 1975; Luce 1977). The realized utility of alternative $y_j$ for agent $i$ is generated by $u(x_i, y_j) \equiv \theta(x_i, y_j) + \epsilon_{i,j}$, where $\epsilon_{i,j}$ is a zero mean independent random variable that follows the Gumbel distribution. Then, agent $i$ ranks the alternatives in decreasing order of their realized utilities. The density function of the Plackett-Luce model has a closed-form formula. Let $y_{j_1} \succ_i y_{j_2}$ represent that $y_{j_1}$ is ahead of $y_{j_2}$ in $R_i$ and let $j_1, j_2, \ldots j_m$ be a permutation of $[m]$. We have

$$\Pr\left[y_{j_1} \succ_i \cdots \succ_i y_{j_m}\right] = \prod_{t=1}^{m} \frac{\theta(x_i, y_{j_t})}{\sum_{t^*=t}^{m} \theta(x_i, y_{j_{t^*}})}. \qquad (1)$$

The marginal distribution between alternatives $j_1$ and $j_2$ is $\Pr[y_{j_1} \succ_i y_{j_2}] = \frac{\theta(x_i, y_{j_1})}{\theta(x_i, y_{j_1}) + \theta(x_i, y_{j_2})}$.

*Distributions of $\mathcal{X}$ and $\mathcal{Y}$.* $x_i$ and $y_j$ are i.i.d. generated from distributions $\mathcal{D}_X$ and $\mathcal{D}_Y$. The supports of $\mathcal{D}_X$ and $\mathcal{D}_Y$ are a cube $\mathbf{B}(d)$ in $\mathbf{R}^d$, where $\mathbf{B}(d) = \{v \in \mathbf{R}^d : \|v\|_\infty \leq c\}$, where $c$ is a constant. We adopt the standard "near uniform" assumption for $\mathcal{D}_X$ and $\mathcal{D}_Y$ (Abraham et al. 2015; Hoff, Raftery, and Handcock 2002; Kleinberg 2000; Sarkar, Chakrabarti, and Moore 2011).

**Definition 1.** *Consider a continuous distribution $\mathcal{D}$ on $\mathbf{B}(d)$ with probability density function $f_{\mathcal{D}}(x)$. $\mathcal{D}$ is near-uniform if $\frac{\sup f_{\mathcal{D}}(x)}{\inf f_{\mathcal{D}}(x)}$ is bounded by a constant, where $x \in \mathbf{B}(d)$.*

Let $f_X$ and $f_Y$ be the PDFs of $\mathcal{D}_X$ and $\mathcal{D}_Y$ respectively. Define $c_X = \frac{\sup f_X(x)}{\inf f_X(x)}$ and $c_Y = \frac{\sup f_Y(x)}{\inf f_Y(x)}$.

*Observation model.* We observe only agent $i$'s ranking over a subset $\mathcal{O}_i \subseteq [m]$ of alternatives. Each alternative $j$ is in $\mathcal{O}_i$ independently with probability $p$. The $\mathcal{O}_i$'s are also independently generated across different agents. Let $R^{\mathcal{O}_i}$ be the ordered list over $\mathcal{O}_i \subseteq \mathcal{Y}$ that is consistent with $R$ (i.e., $R^{\mathcal{O}_i}$ is the partial ranking of $R$ over $\mathcal{O}_i$). For each agent $i$, we observe $R_i^{\mathcal{O}_i}$.

**The near neighbor problem.** Here, an algorithm needs to find near neighbors of an input agent. An algorithm is a $k(n,m)$-NN solver with parameter $\tau(n)$ if

- for any input agent $i$, the algorithm outputs $k$ agents $i_1, i_2, \dots i_k$, and
- with overwhelming probability, $|x_i - x_{i_j}| \le \tau(n)$, where $\tau(n) = o(1)$.

We often write $k$-NN or $k$NN instead of $k(n,m)$-NN when $k$'s dependencies on $m$ and $n$ are not critical.

**Additional notations and examples.** For an arbitrary ordered list $R$, we use it calligraphic form $\mathcal{R}$ to extract the rank of an alternative. For example, suppose $y_j$ is the top-ranked alternative in $R$, then $\mathcal{R}(y_j) = 1$. Let $\mathbf{I}(v)$ be an indicator that sets to 1 if the argument $v$ is true; if false, it sets to 0.

Let $|R|$ be the length of the list $R$. Let $R_1$ and $R_2$ be two ordered lists over the same set of alternatives. The normalized Kendall-Tau distance between $R_1$ and $R_2$ is

$$\begin{aligned}\mathrm{NK}(R_1, R_2) = & \frac{1}{\binom{|R_1|}{2}} \sum_{j_1 \neq j_2 \in R_1} \mathbf{I}\Big(\big(\mathcal{R}_1(j_1) - \mathcal{R}_1(j_2)\big) \\ & \big(\mathcal{R}_2(j_1) - \mathcal{R}_2(j_2)\big) < 0\Big).\end{aligned} \tag{2}$$

When $R_1$ and $R_2$ do not have the same support, the normalized KT distance is defined as $\mathrm{NK}(R_1^{\mathcal{O}}, R_2^{\mathcal{O}})$, where $\mathcal{O} = R_1 \cap R_2$.

To facilitate analysis, sometimes we need to introduce new agents outside $\mathcal{X}$. For a new agent with latent features $x$, let $R_x$ denote its ranking over $\mathcal{Y}$ and let $R_x^{\mathcal{O}_x}$ denote the observed ranking.

*Conditional probability and expectations.* There are multiple levels of randomness for producing the rankings $R_i$'s: (i) $x_i$ and $y_j$ are random and (ii) $u(x_i, y_j)$ consists of a random component (i.e., randomness from the Plackett-Luce model). Care must be taken when operating the conditional random variables defined in our process. For example,

- $\mathbb{E}[\mathrm{NK}(R_{i_1}, R_{i_2} \mid \mathcal{X}]$ refers to fixing the latent positions of the agents and taking expectations over $\mathcal{Y}$ and randomness from the Plackett-Luce model.
- $\mathbb{E}[\mathrm{NK}(R_{i_1}, R_{i_2}) \mid \mathcal{X}, \mathcal{Y}]$ refers to fixing the latent positions of both alternatives and agents and taking expectations over randomness from the Plackett-Luce model.

## 3 Inefficacy of KT-kNN

In this section, we will prove the inefficacy of KT-kNN algorithm by Liu and Yang (2008) (Algorithm 1) in our distanced-based random preference model. This implies that Assumption 1 does not hold in our model.

**Algorithm 1:** KT-kNN (it produces **incorrect** results)

```
Input: $\{R_j^{\mathcal{O}_j}\}_{j \in [n]}$, $k$, and an agent $x_i$.
Output: $k$ neighbors near agent $i$ in the latent space.
Find $j_1, j_2, \cdots, j_{n-1} (\in [n]/\{i\})$ such that
    $\mathrm{NK}(R_i^{\mathcal{O}_i}, R_{j_1}^{\mathcal{O}_{j_1}}) \le \cdots \le \mathrm{NK}(R_i^{\mathcal{O}_i}, R_{j_{n-1}}^{\mathcal{O}_{j_{n-1}}})$
Return $\mathcal{X}_{\text{KT-kNN}} \leftarrow \{j_1, \dots j_k\}$
```

Recall that KT-kNN uses KT distances to find an agent's neighbors based on the intuition that when $x_i$ and $x_j$ are close, their "opinion" on alternatives' utilities should be similar. The next theorem show that this intuition does not hold in our model, by proving that KT-kNN **does not return any near neighbors** for a large fraction of $x_i$.

**Theorem 1.** *Consider Algorithm* KT-kNN *under* distance-based random preference model *in which $d = 1$ and $p = 1$. Let $\mathcal{D}_Y$ and $\mathcal{D}_X$ be uniform distributions on* $[-1, 1]$*. For any constant $\epsilon$, any $x_i \in [-1+\epsilon, -0.5] \cup [0.5, 1-\epsilon]$, and any $k \le n/\ln^5 n$, we have*

$$\min_{x \in \text{KT-kNN}(\{R_j^{\mathcal{O}_j}\}_j, k, x_i)} ||x - x_i||_2 \ge \epsilon = \Omega(1). \tag{3}$$

*with high probability. The probability comes from random $\mathcal{X}/\{x_i\}$, random $\mathcal{Y}$, and random preferences.*

**Remarks.** Theorem 1 states that KT-kNN fails to work *even for the simple case where $d = 1$* and $\mathcal{D}_X = \mathcal{D}_Y =$ Uniform$([-1, 1])$. Eq. (3) is a strong result because trivial algorithms exist to find an agent $x_j$ whose distance to $x_i$ is $\Theta(1)$ (just picking up an arbitrary $x_j$). In addition, this result continues to hold for large populations (e.g., when $n, m \to \infty$ and $p = 1$), suggesting that the limitation of the KT-based approach roots at the structural properties of the NK function. In addition, *if we use* KT-kNN *to solve PC problem by applying standard techniques, it will also produce poor results* (see Appendix D and Lemma 8 there).

**Comparison to (Katz-Samuels and Scott 2018).** Katz-Samuels and Scott (2008) proved that KT-kNN is effective under the *deterministic* utility model. This suggests that with the presence of uncertainties in the utility function (a more realistic assumption), the algorithmic structure of the NN problem is significantly altered.

**Intuitions behind Theorem 1.** The following example highlights the salient structures of KT-distances.

**Example 1** (Near-neighbors in expectation)**.** Let $x_1 = 0$, $y_1 = -0.5$, and $y_2 = 1$. Let $x^* = \arg\min_x \mathbb{E}[\mathrm{NK}(R_x, R_1) \mid x_1, x, \mathcal{Y}]$ (e.g., where would we place an agent that minimizes its KT distance to $x_1$?). One would hope that when $x^*$ and $x_1$ are close, $R_{x^*}$ and $R_1$ is close, but here $x^* = -0.5$. Specifically, let $a$ be the probability $x_1$ prefers $y_1$ to $y_2$ (i.e., $a \equiv \frac{\theta(x_1, y_1)}{\theta(x_1, y_1) + \theta(x_1, y_2)}$) and let $b(x)$ be the probability $x$ prefers $y_1$ to $y_2$. Recall that agents' support is $[-1, 1]$. We need to solve

$$\begin{aligned} x^* &= \arg\min_x \mathbb{E}[\mathrm{NK}(R_x, R_1) \mid x_1, x, \mathcal{Y}] \\ &= \arg\min_x a(1 - b(x)) + (1 - a)b(x). \end{aligned}$$

Here, we aim to minimize the weighted sum of $a \in [0,1]$ and $(1-a)$ via controlling $b(x)$. The optimal solution has a simple structure: when $a > (1-a)$ (equivalently, $a > 0.5$), we need to set the weight associated with $a$ as small as possible, which means setting $b(x)$ to the largest possible value. When $a < (1-a)$, $b(x)$ needs to be minimized. Thus, the optimal solution uses the following threshold rules (assume $a \neq 0.5$ for simplicity).

$$x^* \in \begin{cases} (-1, y_1) & \text{if } a > 0.5 \\ (y_2, 1) & \text{if } a < 0.5 \end{cases}.$$

This minimizer is far from $x_1$. □

See also Example 3 in Appendix B.5 for another small and concrete example, in which KT-kNN produces poor output.

### 3.1 Proof sketch of Theorem 1

We use intuitions from Example 1 to prove the theorem. Specifically, define

$$\mathcal{G}_i(x) \equiv \mathbb{E}[\mathrm{NK}(R_i, R_x) \mid x_i, x].$$

First, note that $\mathrm{NK}(R_i, R_x)$ concentrates at $\mathcal{G}_i(x)$ when $m$ is sufficiently large. This comes from the concentration behavior of the NK function:

**Lemma 1.** *Let $\mu = \mathbb{E}[\mathrm{NK}(R_i, R_j) \mid \mathcal{X}] = \mathcal{G}_i(x_j)$. We have*

$$\Pr\left[|\mathrm{NK}(R_i, R_j) - \mu| \geq \delta\mu \mid \mathcal{X}\right] \leq 4m \exp\left(-\frac{\delta^2 m \mu}{6}\right).$$

See Appendix B.1 for the proof. The terms in NK are *not* independent terms so we cannot directly apply Chernoff bounds. Our proof uses the combinatorial structure of the NK function to decouple the dependencies among terms. The technique we develop can be of independent interests.

Let $x^* = \arg\min_x \mathcal{G}_i(x)$. Below is our main lemma:

**Lemma 2.** *Let $\mathcal{D}_Y$ be uniform distribution on $[-1,1]$. Let $x_i$ be any agent in $[-1,-0.5]$. We have*

$$\arg\min_x \mathcal{G}_i(x) = -1.$$

*Similarly, when $x_i \in [0.5, 1]$, $\arg\min_x \mathcal{G}_i(x) = 1$.*

For any $x_i \in [-1,-0.5] \cup [0.5,1]$, Lemma 1 and Lemma 2 give us:

$$\min_{i^*} \mathrm{NK}(R_{i^*}, R_i) \approx \min_{i^*} \mathcal{G}_i(x_{i^*}) \approx \mathcal{G}_i(-1),$$

where the first approximation comes from the concentration bound of NK and the second approximation comes from the fact that there must exist one agent close to -1 when the number of agent is large. Therefore, all the neighbors produced by KT-kNN are far from $x_i$ (see Appendix B.4 for a rigorous analysis).

*Proof of Lemma 2.* By linearity of expectation, we have

$$\begin{aligned} \mathcal{G}_i(x) &= \frac{1}{\binom{m}{2}} \sum_{\ell_1 \neq \ell_2} \mathbb{E}\left[\mathrm{NK}\left(R_i^{\{y_{\ell_1}, y_{\ell_2}\}}, R_x^{\{y_{\ell_1}, y_{\ell_2}\}}\right) \middle| x, x_i\right] \\ &= \mathbb{E}\left[\mathrm{NK}\left(R_i^{\{y_{\ell_1}, y_{\ell_2}\}}, R_x^{\{y_{\ell_1}, y_{\ell_2}\}}\right) \middle| x, x_i\right]. \end{aligned}$$

The last equality holds because $y_j$'s are i.i.d. samples from $\mathcal{D}_Y$. Define

$$p_i(y_1, y_2) \equiv \Pr[y_1 \succ_i y_2 \mid y_1, y_2, x_i] = \frac{\theta(x_i, y_1)}{\theta(x_i, y_1) + \theta(x_i, y_2)}.$$

$$p_x(y_1, y_2) \equiv \Pr[y_1 \succ_x y_2 \mid y_1, y_2, x] = \frac{\theta(x, y_1)}{\theta(x, y_1) + \theta(x, y_2)}.$$

When the context is clear, we shall refer to $p_i(y_1, y_2)$ and $p_x(y_1, y_2)$ as $p_i$ and $p_x$, respectively. We have

$$\mathcal{G}_i(x) = \mathbb{E}_{y_1, y_2}\left[p_x(1-p_i) + (1-p_x)p_i \mid x_i, x\right].$$

One can see that $\mathcal{G}_i(x)$ is a smooth function (the first derivative exists). Our proof consists of three parts. *Part 1.* When $x \in (-1, x_i]$, $\partial\mathcal{G}_i(x)/\partial x > 0$, *Part 2.* When $x \in [x_i, -x_i]$, $\mathcal{G}_i(x) - \mathcal{G}_i(x_i) > 0$, and *Part 3.* When $x \in [-x_i, 1)$, $\partial\mathcal{G}_i(x)/\partial x > 0$.

The proof for part 3 is similar to part 1. Proving part 2 is also simpler. Therefore, we focus only on the proof for part 1. Proof for part 2 and 3 is deferred to Appendix B.2.

We now show that when $x \in (-1, x_i]$, $\partial\mathcal{G}_i(x)/\partial x > 0$. We have (see Fact 2 in Appendix B.2):

$$\frac{\partial \mathcal{G}_i(x)}{\partial x} = \mathbb{E}\left[\Phi(y_1, y_2, x, x_i) \middle| x_i, x\right], \tag{4}$$

where

$$\begin{cases} \Phi(y_1, y_2, x, x_i) & \equiv \frac{e^{-\Delta^{(i)}_{1,2}} - 1}{e^{-\Delta^{(i)}_{1,2}} + 1} \cdot \frac{\mathrm{sign}(y_1 - x) - \mathrm{sign}(y_2 - x)}{4\cosh^2\left(\Delta^{(x)}_{1,2}/2\right)} \\ \Delta^{(i)}_{1,2} & \equiv |y_2 - x_i| - |y_1 - x_i| \\ \Delta^{(x)}_{1,2} & \equiv |y_2 - x| - |y_1 - x| \end{cases}.$$

Here, $\Delta^{(i)}_{1,2}$ ($\Delta^{(x)}_{1,2}$) measures whether $x_i$ ($x$) is closer to $y_2$ or $y_1$. Similar to Example 1, they serve as important quantities determining the structure of $\partial\mathcal{G}_i(x)/\partial x$ (and therefore the optimal solution).

One can check that $\Phi(y_1, y_2, x, x_i) = \Phi(y_2, y_1, x, x_i)$. Therefore,

$$\frac{\partial \mathcal{G}(x)}{\partial x} = \mathbb{E}_{y_1, y_2}\left[\Phi(y_1, y_2, x, x_i) \mid x, x_i, (y_1 \leq y_2)\right].$$

Central to our analysis is carefully partitioning the event $y_1 \leq y_2$ into four disjoint (sub)-events. Under each event, the conditional expectation of $\Phi$ can be computed in a straightforward manner. Specifically, define

- $\mathcal{E}_1$: when $x \leq y_1 \leq y_2$ or $y_1 \leq y_2 \leq x$. Thus, $\Pr[\mathcal{E}_1 \mid y_1 \leq y_2] = \frac{x^2+1}{2}$.
- $\mathcal{E}_2$: when $y_1 < x$ and $y_2 \geq 1 + 2x_i$. Thus, $\Pr[\mathcal{E}_2 \mid y_1 \leq y_2] = -(x+1)x_i$.
- $\mathcal{E}_3$: when $y_1 < x$ and $x < y_2 < 2x_i - x$. Thus, $\Pr[\mathcal{E}_3 \mid y_1 \leq y_2] = (x+1)(x_i - x)$.
- $\mathcal{E}_4$: when $y_1 < x$ and $2x_i - x \leq y_2 < 1 + 2x_i$. Thus, $\Pr[\mathcal{E}_4 \mid y_1 \leq y_2] = \frac{(x+1)^2}{2}$.

Figure 3(a) in Appendix A visualizes the events to complement the analysis. We now interpret the meaning of these events.

**Event $\mathcal{E}_1$.** Event $\mathcal{E}_1$ represents the case in which $y_1$ and $y_2$ are on the same side of $x$. In this case, any movement

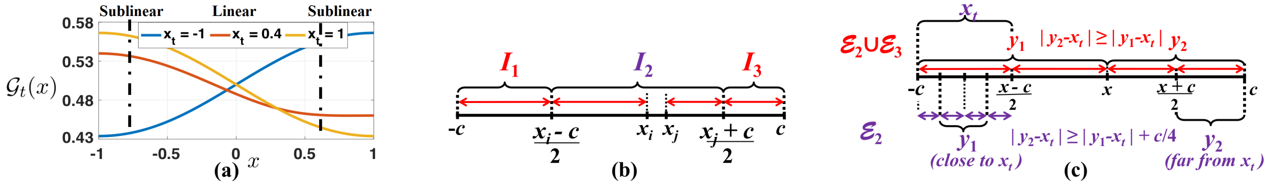

Figure 1: **(a)** Functions $\mathcal{G}_t(x)$ for $c = 1$, and $x_t = -1, 0.4$, and 1. Observations: *(i)* when $x_t = \pm 1$, the function $\mathcal{G}_t(x)$ is a monotone function; *(ii)* for all $x_t$, $\mathcal{G}_t(x)$ grows in sublinear manner when $x$ is close to $\pm 1$. **(b)** Definition of $I_1, I_2$ and $I_3$ used in the lower bound proof for Lemma 3 (Section 4). Agents in $I_1$ and $I_3$ are effective anchor agents. **(c)** Intuition of $\mathcal{E}_2$ and $\mathcal{E}_3$ in the proof of Lemma 4 (Section 4).

of $x$ without passing $y_1, y_2$ will not change the probability that $y_1 \succ_i y_2$ occurs (i.e., $\Pr[y_1 \succ_i y_2 \mid y_1, y_2, x, \mathcal{E}_1] = \Pr[y_1 \succ_i y_2 \mid y_1, y_2, x \pm \delta, \mathcal{E}_1]$ for any sufficiently small $\delta$). Therefore, $\mathcal{E}_1$ does not impact $\mathbb{E}[\Phi]$ and $\mathbb{E}[\Phi \mid \mathcal{E}_1, x, x_i] = 0$.

**Event $\mathcal{E}_2$.** Under this event, one can check that $|x_i - y_1| < |x_i - y_2|$ (i.e., $x_i$ is closer to $y_1$ than to $y_2$). In this case, when we increase the value of $x$ (recall that $y_1 < x < x_i$), $\mathrm{NK}(R_i^{\{y_1,y_2\}}, R_x^{\{y_1,y_2\}})$ will increase. This is equivalent to $\mathbb{E}[\Phi \mid \mathcal{E}_2, x, x_i] > 0$.

**Event $\mathcal{E}_3$.** Under this event, one can check that $|x_i - y_1| > |x_i - y_2|$.In this case, when we increase the value of $x$, $\mathrm{NK}(R_i^{\{y_1,y_2\}}, R_x^{\{y_1,y_2\}})$ will decrease. This is equivalent to $\mathbb{E}[\Phi \mid x, x_i, \mathcal{E}_3] < 0$.

**Event $\mathcal{E}_4$.** Under this event $|x_i - y_1| - |x_i - y_2|$ is positive (we call this a positive event) with probability 0.5 and is negative (we call this a negative event) with probability 0.5. The first order term conditioned under the positive event cancels out that conditioned under the negative event. Therefore, $\mathbb{E}[\Phi \mid x, x_i, \mathcal{E}_4]$ will be a "small" term.

With the above intuition, we have

$$\begin{aligned}\mathbb{E}[\Phi \mid x, x_i] \approx\ & \mathbb{E}[\Phi \mid x, x_i, \mathcal{E}_2] \Pr[\mathcal{E}_2 \mid x, x_i] \\ & + \mathbb{E}[\Phi \mid x, x_i, \mathcal{E}_3] \Pr[\mathcal{E}_3 \mid x, x_i].\end{aligned}$$

We now relate $\mathcal{E}_2$ and $\mathcal{E}_3$ to Example 1. Event $\mathcal{E}_2$ corresponds to the setting where $x_i$ is closer to $y_1$ than to $y_2$. As explained in Example 1, when we move $x$ to the right, we *increase* the expected KT distance, whereas in $\mathcal{E}_3$ when we move $x$ to the right, we *decrease* the expected KT distance. The crucial observation is that $\mathcal{E}_2$ is much more likely to happen than $\mathcal{E}_3$. The observation becomes clear with visualization in Fig. 3b (i.e., the area for $\mathcal{E}_2$ is much larger than that for $\mathcal{E}_3$). Using these intuitions, we have (see Appendix B.2 for the full analysis):

**Fact 1.** *Using notations above, we have ,*

$$\mathbb{E}[\Phi \mid x, x_i] = \sum_{t=2,3,4} \mathbb{E}[\Phi \mid x, x_i, \mathcal{E}_t] \Pr[\mathcal{E}_t \mid x, x_i] > 0.$$

This completes the proof of Lemma 2. □

## 4 Anchor-based nearest neighbor algorithms

This section develops a new (high-dimensional) feature $\vec{F}_i$ for each agent $i$ so that $|\vec{F}_i - \vec{F}_j|$ is small if and only if $x_i$ and $x_j$ are close. We present the positive results in the most general form in 1-dim latent space (i.e., we allow $\mathcal{D}_X$ and $\mathcal{D}_Y$ to be any near-uniform distribution, $p = o(1)$, and the latent space is $[-c, c]$ for any constant $c$).

**Index convention.** This section uses $i$, $j$, and $t$ to index agents and $\ell$ (include $\ell_1$, $\ell_2$, etc.) to index alternatives.

**Intuition of the design of $\vec{F}_i$.** Our key idea is to leverage a third agent, namely an *anchor agent*, to determine the closeness of two agents $x_i$ and $x_j$. Let $x_t$ be a third agent. Compute $\mathrm{NK}(R_t, R_i)$ and $\mathrm{NK}(R_t, R_j)$. If $x_t$ were chosen appropriately, then $\mathrm{NK}(R_t, R_i) \approx \mathrm{NK}(R_t, R_j)$ if and only if $x_i$ and $x_j$ are close. For example, if $x_t = -c$, then $\mathrm{NK}(R_t, R_i) \approx \mathcal{G}_t(x_i)$ and the function $\mathcal{G}_t(x)$ is a monotone function (see the blue curve in Fig. 1a). We have $\mathcal{G}_t(x_i) \approx \mathcal{G}_t(x_j)$ if and only if $x_i \approx x_j$.

We face two key technical challenges: *C1.* Not all agents can be served as effective anchor agents. For example, when $x_t = 0$, $\mathcal{G}_t(x) = \mathcal{G}_t(-x)$, which cannot separate $x_i$ and $x_j$ when $x_i = -x_j$. *C2.* the efficacy of the anchor agent is sensitive to $x_i$ and $x_j$. For example, when $x_t = -c$ (see again Fig. 1a), the function $\mathcal{G}_t(x)$ grows in a sub-linear manner so it is less effective in detecting the closeness of $x_i$ and $x_j$ when they are both close to $-c$.

To address C1, we use *all possible* agents as anchor agents. To address C2, we need to develop new probabilistic techniques to analyze $\mathcal{G}_t(x)$.

**Our features.** Let $\vec{F}_i = (F_{i,1}, F_{i,2}, \cdots F_{i,n})$, where

$$F_{i,t} \equiv \mathbb{E}_{R_i, R_t, \mathcal{Y}}[\mathrm{NK}(R_i, R_t) \mid \mathcal{X}] = \mathcal{G}_t(x_i).$$

Then we use the $L_1$-distance between $\vec{F}_i$ and $\vec{F}_j$ to determine whether $x_i$ and $x_j$ are close. Define

$$D(x_i, x_j) \equiv \frac{1}{n-2} \sum_{t \notin \{i,j\}} |F_{i,t} - F_{j,t}|. \tag{5}$$

The summation excludes $i$ and $j$ because $x_i$ and $x_j$ themselves cannot be used as anchor agents.

**Replacing the features $\vec{F}$ by estimates.** $F_{i,t}$ is not directly given so we use the empirical estimate as the plug-in estimator. Define $\hat{F}_{i,t} = \mathrm{NK}(R_i^{\mathcal{O}_i}, R_t^{\mathcal{O}_t})$ and our estimator is $\hat{D}(x_i, x_j) = \frac{1}{n-2} \sum_{t \notin \{i,j\}} |\hat{F}_{i,t} - \hat{F}_{j,t}|$ (see Algorithm 2).

**Theorem 2.** *Consider Algorithm* Anchor-kNN *under* distance-based random preference model. *Let $\mathcal{D}_X$ and $\mathcal{D}_Y$ be any near-uniform distribution on $[-c, c]$. Let $\tau(n)$ be*

**Algorithm 2:** Anchor-kNN

```
Input: $\{R_j^{\mathcal{O}_j}\}_{j\in[n]}$, $k$, and agent $x_i$.
Output: $k$ neighbors near agent $i$.
Compute $\hat{F}_{i,j} = \mathrm{NK}(R_i^{\mathcal{O}_i}, R_j^{\mathcal{O}_j})$ for all $i, j \in [n]$.
Compute $\hat{D}(x_i, x_j) = \frac{1}{n-2}\sum_{t\neq i,j} |\hat{F}_{i,t} - \hat{F}_{j,t}|$.
Find $j_1, j_2, \cdots, j_{n-1}$ ($\in [n]/\{i\}$) such that
  $\hat{D}(x_i, x_{j_1}) \le \hat{D}(x_i, x_{j_2}) \cdots \le \hat{D}(x_i, x_{j_{n-1}})$.
Return $\mathcal{X}_{\text{Anchor-kNN}} \leftarrow \{j_1, \cdots, j_k\}$.
```

*an arbitrary quality parameter so that* $\tau(n) = o(1) \wedge \tau(n) = \omega\left(n^{-\frac{1}{4}}\sqrt{\ln n}\right)$. *For any* $x_i \in [-c, c]$, *any* $m = \omega\left(\frac{\ln^3 n}{p^2\cdot\tau^4(n)} \cdot \ln^2\left(\frac{\ln^3 n}{p^2\cdot\tau^4(n)}\right)\right)$ *and any* $k = o(n \cdot \tau^2(n) \cdot \ln^{-1} n)$, *we have*

$$\max_{x\in \text{Anchor-kNN}(\{R_j^{\mathcal{O}_j}\}_j, k, x_i)} ||x - x_i||_2 \le \tau(n)$$

*with high probability. The probability comes from random* $\mathcal{X}/\{x_i\}$, *random* $\mathcal{Y}$, *and random preferences.*

**Remark.** $\tau(n)$ cannot be too small because our function $D(x_i, x_j)$ cannot measure the distance of two agents well if they are too close. $m$ needs to grow when $p$ (fewer samples) or $\tau(n)$ decreases (higher quality requirement), which is intuitive. $k$ measures the number of numbers an algorithm can find so that their distance is within $\tau(n)$; larger $k$ means the algorithm is more powerful.

Let $\mathbb{D}(x_i, x_j) = \mathbb{E}[D(x_i, x_j)]$. In the remainder of this section, we analyze the behavior of of $\mathbb{D}$. The function $\hat{D}$ concentrates at $\mathbb{D}$ and can be shown by using simple Chernoff bounds (see Appendix C.2 for a complete analysis).

**Lemma 3.** *For any near-uniform* $\mathcal{D}_X, \mathcal{D}_Y$ *on* $[-c, c]$ *and any two agents* $x_i, x_j$, *we have*

$$c_3(c) \cdot \left(\ln^{-1} n\right) \cdot |x_i - x_j|^2 \le \mathbb{D}(x_i, x_j) \le |x_i - x_j|,$$

*where* $c_3$ *is a constant that depends only on* $c$.

**Upper bound proof for Lemma 3.** The upper bound requires only a straightforward calculation. Recall that $p_i = p_i(y_1, y_2) = \Pr[y_1 \succ_i y_2 \mid y_1, y_2, x_i]$. We have

$$\begin{aligned}\mathbb{D}(x_i, x_j) &= \mathbb{E}_{x_t}\left[\left|\mathbb{E}_{y_1,y_2}[(p_i - p_j)(1 - 2p_t) \mid x_i, x_j, x_t]\right|\right] \\ &\le \mathbb{E}_{y_1,y_2}\left[|p_i - p_j| \mid x_i, x_j\right] \le |x_i - x_j|.\end{aligned} \tag{6}$$

The last inequality is shown in Fact 6 in Appendix C.1.

**Lower bound bound proof for Lemma 3.** Here we analyze only the case $|x_i - x_j| \le 2c - 2\ln n$. When $|x_i - x_j| > 2c - 2\ln n$ (e.g., $x_i$ and $x_j$ are around the boundaries $-c$ and $c$, respectively), the result is trivial.

Wlog, assume that $x_i < x_j$. We partition $[-c, c]$ into three intervals and consider anchor agents in each of these intervals. Specifically, define (see also Figure 1(b))

$$I_1 \equiv \left[-c, \frac{x_i - c}{2}\right],\ I_2 \equiv \left(\frac{x_i - c}{2}, \frac{x_j + c}{2}\right) \;\&\; I_3 \equiv \left[\frac{x_j + c}{2}, c\right].$$

The agents in $I_2$ are "less effective anchors" (C2). We use trivial bound for terms in $I_2$ ($|F_{i,t} - F_{j,t}| \ge 0$). Focusing on $I_1$ and $I_3$, we have

$$\begin{aligned}&\mathbb{D}(x_i, x_j) \\ \ge\ &\frac{x_i + c}{4c \cdot c_X} \cdot \mathbb{E}_{x_t}\left[\left|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)\right| \mid x_t \in I_1, x_i, x_j\right] + \\ &\frac{c - x_j}{4c \cdot c_X} \cdot \mathbb{E}_{x_t}\left[\left|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)\right| \mid x_t \in I_3, x_i, x_j\right].\end{aligned} \tag{7}$$

Note that $\frac{x_i+c}{4c\cdot c_X} + \frac{c-x_j}{4c\cdot c_X}$ is at least $\tilde{\Omega}(1)$. Now we show that $\mathbb{E}_{x_t}\left[\left|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)\right| \mid x_t \in I_1, x_i, x_j\right]$ is at least in the order of $|x_i - x_j|^2$. The analysis for the other term is similar. Below is the lemma we need (related to C2):

**Lemma 4.** *For any near-uniform* $\mathcal{D}_X, \mathcal{D}_Y$ *on* $[-c, c]$ *such that* $|x_i - x_j| \ge 2c - 2\ln n$ *and* $x_t \in I_1$, *we have*

$$|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)| \ge c_4(c) \cdot |x_i - x_j|^2,$$

*where* $c_4(c) = \frac{1-e^{-c/4}}{1+e^{-c/4}} \cdot \frac{1}{96c^2\cdot\cosh^2(c)\cdot c_Y^2} \in (0, \frac{1}{2c})$.

*Proof of Lemma 4.* Note that $|x_i - x_j| \ge 2c - 2\ln n$ and $x_t \in I_1$ imply $x_j \ge x_i \ge 2x_t + c$. Because $|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)| = \left|\int_{x_i}^{x_j} \frac{\partial \mathcal{G}_t(x)}{\partial x} dx\right|$, we aim to give a bound for $\frac{\partial \mathcal{G}_t(x)}{\partial x}$. Specifically,

$$\frac{\partial \mathcal{G}_t(x)}{\partial x} \ge 3c_4(c) \cdot (c^2 - x^2).$$

when $x \ge 2x_t + c$ (or equivalently, $x_t \le \frac{x-c}{2}$). Re-cycle the definition of $\Phi$, $\Delta_{1,2}^{(i)}$, and $\Delta_{1,2}^{(x)}$ used in the analysis of Theorem 1 (see also Appendix A for the notation summary) so that $\frac{\partial \mathcal{G}(x)}{\partial x} = \mathbb{E}_{y_1<y_2}[\Phi(y_1, y_2, x, x_t) \mid x, x_t]$

We partition the positions of $\{y_1, y_2\}$ into events (see also Fig. 1c for a visualization):

- $\mathcal{E}_1$: when $x \le y_1 \le y_2$ or $y_1 \le y_2 \le x$.
- $\mathcal{E}_2$: when $y_1 \in [\frac{x-7c}{8}, \frac{3x-5c}{8}]$ and $y_2 \ge \frac{x+c}{2}$. We have $\Pr[\mathcal{E}_2 \mid y_1 < y_2] \ge \frac{c^2-x^2}{16c^2c_Y^2}$.
- $\mathcal{E}_3$: when $(y_1, y_2) \notin \mathcal{E}_1 \cup \mathcal{E}_2$. As explained below, we do not need to explicitly calculate $\Pr[\mathcal{E}_3 \mid y_1 \le y_2]$.

We now explain the intuition associated with these events.

**Event $\mathcal{E}_1$.** Because $y_1$ and $y_2$ are on the same side of $x$, we have $\mathbb{E}[\Phi \mid \mathcal{E}_1, x, x_t] = 0$ (see also Lemma 2).

**Event $\mathcal{E}_2$ and event $\mathcal{E}_3$.** When $(y_1, y_2) \in \mathcal{E}_2 \cup \mathcal{E}_3$, we have $|y_1 - x_t| \le |y_2 - x_t|$ ($x_t$ is closer to $y_1$ than to $y_2$). This is because *(i)* $|y_2 - x_t| \ge x - x_t$ when $y_2 \ge x$ and $x \in I_1$, and *(ii)* $|y_1 - x_t| \le \max\{x_t - c, x - x_t\} \le x - x_t$ since $y_1 \le x$ and $x \ge 2x_t + c$. This conclusion is trivial when the positions of the points are visualized (Figure 1(c)).

In these events, an increment in $x$ will result in an increment in $\mathcal{G}_t(x)$. Therefore $\mathbb{E}[\Phi \mid \mathcal{E}_2 \cup \mathcal{E}_3, x, x_t] \ge 0$.

**Event $\mathcal{E}_2$.** Knowing $\Phi$ can be arbitrarily close to 0 when $\mathcal{E}_2 \cup \mathcal{E}_3$ happens. Here, we need also identify an event so that $\Phi$ is at least a positive constant. Event $\mathcal{E}_2$ serves for this purpose because it ensures $x$ to be far from the mid-point

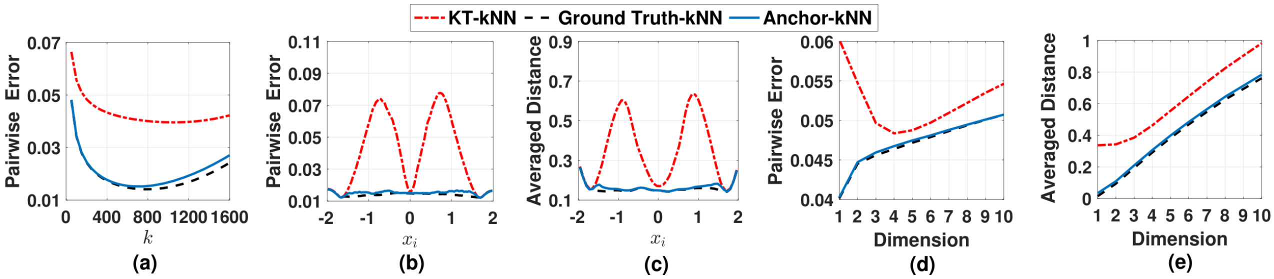

Figure 2: **(a)** Average pairwise probability prediction error (or pairwise error, see Appendix E.1 for the definition) for different $k \in [101, 1601]$. **(b)** Pairwise error when $k = 751$ (optimal $k$ for ground-truth kNN). **(c)** The average latent distance between $x_i$ and its predicted 751 near neighbors. **(d)** Validation for high dimension ($1 \sim 10$ dimensional latent space). Let $k = 73 \approx \ln^2 n$ and measure the pairwise errors when $k = 73$. **(e)** The average latent distance when $k = 73$.

of $y_1$ and $y_2$, so $\mathcal{G}(x)$ behaves like a linear function in this region (and thus its derivative behaves like a constant).

**Intuition on dependencies on $|x_i - x_j|^2$.** We now explain why $|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)|$ depends on $|x_i - x_j|^2$ instead of $|x_i - x_j|$. Only if $(x_i, x_j) \notin \mathcal{E}_1$, $\Phi$ will be non-zero. This requires $y_1 \leq x \leq y_2$. Recall we only interests to $x_i \leq x \leq x_j$. When $x_i$ and $x_j$ get too close to $-c$, $\Pr[y_1 < x < y_2 \mid x] = \Theta(x_i + c) \approx \Theta(|x_i - x_j|)$. When we carry out integration over $\partial \mathcal{G}_t(x)/\partial x$, this term will lead to an additional factor of $|x_i - x_j|$ (see also Figure 1(a) for simulated $\mathcal{G}_t(x)$).

Using the above intuition, we have

$$\mathbb{E}_{y_1<y_2}[\Phi \mid x, x_t] \geq \mathbb{E}_{y_1<y_2}[\Phi \mid \mathcal{E}_2, x, x_t] \cdot \Pr_{y_1<y_2}[\mathcal{E}_2 \mid x, x_t]$$
$$= \Omega(1) \cdot \frac{c^2 - x^2}{16c^2 c_Y^2}.$$

The last inequality uses the fact that $\mathbb{E}_{y_1 \leq y_2}[\Phi \mid \mathcal{E}_2, x, x_i] = \Omega(1)$ (Fact 7 in Appendix C). By carrying out an integration over $\partial \mathcal{G}_t(x)$, we get $|\mathcal{G}_t(x_i) - \mathcal{G}_t(x_j)| \geq c_4(c) \cdot |x_i - x_j|^2$. □

Lemma 4 and its similar result for $I_3$ suffice to establish the lower bound.

## 5 Experiments

Our experiments aim to *(i)* confirm the behaviors of KT-kNN and Anchor-kNN for finite sample size when $d = 1$, *(ii)* understand the behavior of Anchor-kNN in high-dim settings, and *(iii)* validate the practicality of our algorithm over real-world datasets.

Details of all experiments are in Appendix E.1. Note that this is a *theoretical* paper. Extensive evaluations on real-world data are an promising direction for future work.

**1-dim synthetic data.** In this experiment we compare the efficacy of KT-kNN, Anchor-kNN, and Ground-Truth-kNN. Ground-Truth-kNN assumes an oracle access to an input's (ground-truth) $k$-nearest neighbors so this is an optimal kNN. Figure 2(a) plots the performance of Anchor-kNN,KT-kNN and Ground-truth-kNN in completing the preferences for different $k$. Figure 2(b) plots the completion/prediction error for different $x_i$ using an optimal $k$ (chosen by using cross-validations for Ground-Truth-kNN). Figure 2(c) shows the average latent distance between the return set and $x_i$. The experiments confirm that *(i)* Anchor-kNN is consistently better, *(ii)* KT-kNN does not return near neighbors when $x_i \approx \pm\frac{c}{2}$, and *(iii)* when $x_i \approx \pm\frac{c}{2}$, KT-kNN is the poorest at completing preferences. Appendix E.1 provides additional experiments for different settings on $m$.

**High-dim synthetic data.** We repeat the experiments for $d = 1, \cdots, 10$. Figure 2(d) is the prediction error and Figure 2(e) is the average distance. Anchor-kNN continues to outperform KT-kNN, suggesting that our algorithm works for $d > 1$. The performance of all algorithms deteriorate for large $d$ because of the curse-of-dimensionality problem in neighbor-based algorithms (Radovanović, Nanopoulos, and Ivanović 2010).

**Real dataset.** We examine the performance of Anchor-kNN using the standard Netflix dataset (Bell and Koren 2007; Bennett, Lanning, and others 2007). The baselines are KT-kNN and a standard collaborative filtering algorithm using cosine-similarity (Terveen and Hill 2001; Breese, Heckerman, and Kadie 1998). Table 2 in Supplementary materials presents the results. Anchor-kNN consistently outperforms KT-kNN and collaborative filtering using cosine-similarity. Our experiments suggests that our distance-based random preference model and Anchor-kNN seem to be quite practical.

## 6 Conclusion

This paper introduced a natural learning-to-rank model, and showed that under this model a widely used KT-distance based kNN algorithm failed to find similar agents (users), challenging the assumptions made in many prior preference completion algorithms. To fix the problem, we introduced a new anchor-based algorithm Anchor-kNN that uses all the agents' ranking data to determine the closeness of two agents. Our approach is in sharp contrast to most existing feature engineering methods. We provided a rigorous analysis for Anchor-kNN for 1-dim latent space, and performed experiments on both synthetic and real datasets. Our experiments showed that Anchor-kNN works in high dim space and promises to outperform other widely used techniques.

## 7 Acknowledgments

We thank all anonymous reviewers for helpful comments and suggestions. AL and LX are supported by NSF #1453542 and ONR #N00014-17-1-2621. QW and ZL are supported by NSF CRII:III #1755769.

## References

Abraham, I.; Chechik, S.; Kempe, D.; and Slivkins, A. 2015. Low-distortion inference of latent similarities from a multiplex social network. *SIAM Journal on Computing* 44(3):617–668.

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

Azari Soufiani, H.; Parkes, D. C.; and Xia, L. 2013. Preference elicitation for general random utility models. *arXiv preprint arXiv:1309.6864*.

Azari Soufiani, H.; Parkes, D. C.; and Xia, L. 2014. Computing parametric ranking models via rank-breaking. In *ICML*, 360–368.

Bell, R. M., and Koren, Y. 2007. Lessons from the netflix prize challenge. *Acm Sigkdd Explorations Newsletter* 9(2):75–79.

Bennett, J.; Lanning, S.; et al. 2007. The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, 35. New York, NY, USA.

Breese, J. S.; Heckerman, D.; and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 43–52. Morgan Kaufmann Publishers Inc.

Cheng, W.; Hüllermeier, E.; and Dembczynski, K. J. 2010. Label ranking methods based on the plackett-luce model. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 215–222.

Fan, C., and Lin, Z. 2013. Collaborative ranking with ranking-based neighborhood. In *Asia-Pacific Web Conference*, 770–781. Springer.

Hoff, P. D.; Raftery, A. E.; and Handcock, M. S. 2002. Latent space approaches to social network analysis. *Journal of the american Statistical association* 97(460):1090–1098.

Hughes, D.; Hwang, K.; and Xia, L. 2015. Computing optimal bayesian decisions for rank aggregation via mcmc sampling. In *UAI*, 385–394.

Hwang, S.-w., and Lee, M.-W. 2009. A uncertainty perspective on qualitative preference. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Katz-Samuels, J., and Scott, C. 2018. Nonparametric preference completion. In *International Conference on Artificial Intelligence and Statistics*, 632–641.

Khetan, A., and Oh, S. 2016. Data-driven rank breaking for efficient rank aggregation. *The Journal of Machine Learning Research* 17(1):6668–6721.

Kleinberg, J. 2000. The small-world phenomenon: An algorithmic perspective. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, 163–170. ACM.

Liu, T.-Y., et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval* 3(3):225–331.

Liu, N. N., and Yang, Q. 2008. Eigenrank: a ranking-oriented approach to collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 83–90. ACM.

Liu, A.; Zhao, Z.; Liao, C.; Lu, P.; and Xia, L. 2019. Learning plackett-luce mixtures from partial preferences. In *Proceedings of the Thirty-third AAAI Conference on Artificial Intelligence (AAAI-19)*.

Lu, Y., and Negahban, S. N. 2015. Individualized rank aggregation using nuclear norm regularization. In *Communication, Control, and Computing (Allerton), 2015 53rd Annual Allerton Conference on*, 1473–1479. IEEE.

Luce, R. D. 1977. The choice axiom after twenty years. *Journal of mathematical psychology* 15(3):215–233.

Maystre, L., and Grossglauser, M. 2015. Fast and accurate inference of plackett–luce models. In *Advances in neural information processing systems*, 172–180.

McFadden, D. L. 2000. Economic Choice. Nobel Prize Lecture.

Oh, S.; Thekumparampil, K. K.; and Xu, J. 2015. Collaboratively learning preferences from ordinal data. In *Advances in Neural Information Processing Systems*, 1909–1917.

Park, D.; Neeman, J.; Zhang, J.; Sanghavi, S.; and Dhillon, I. 2015. Preference completion: Large-scale collaborative ranking from pairwise comparisons. In *International Conference on Machine Learning*, 1907–1916.

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

Radovanović, M.; Nanopoulos, A.; and Ivanović, M. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research* 11(2487—2531).

Sarkar, P.; Chakrabarti, D.; and Moore, A. W. 2011. Theoretical justification of popular link prediction heuristics. In *IJCAI proceedings-international joint conference on artificial intelligence*, volume 22, 2722.

Terveen, L., and Hill, W. 2001. Beyond recommender systems: Helping people help each other. *HCI in the New Millennium* 1(2001):487–509.

Wang, S.; Sun, J.; Gao, B. J.; and Ma, J. 2012. Adapting vector space model to ranking-based collaborative filtering. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, 1487–1491. ACM.

Wang, S.; Sun, J.; Gao, B. J.; and Ma, J. 2014. Vsrank: A novel framework for ranking-based collaborative filtering. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5(3):51.

Zhao, Z.; Piech, P.; and Xia, L. 2016. Learning mixtures of plackett-luce models. In *International Conference on Machine Learning*, 2906–2914.

Zhao, Z.; Villamil, T.; and Xia, L. 2018. Learning mixtures of random utility models. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*.