Learning Populations of Preferences via Pairwise Comparison Queries

Gokcan Tatli [1] Yi Chen [1] Ramya Korlakai Vinayak [1]

Abstract

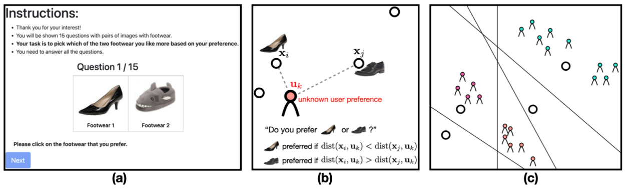
Ideal point based preference learning using pairwise comparisons of type ”Do you prefer a or b?” has emerged as a powerful tool for understanding how we make preferences which plays a key role in many areas. Existing preference learning approaches assume homogeneity and focus on learning preference on average over the population or require a large number of queries per individual to localize individual preferences. However, in practical scenarios with heterogeneous preferences and limited availability of responses, these approaches are impractical. Therefore, we introduce the problem of *learning the distribution of preferences over a population* via pairwise comparisons using only one response per individual. In this scenario, learning each individual's preference is impossible. Hence the question of interest is: what can we learn about the distribution of preferences over the population? Due to binary answers from comparison queries, we focus on learning the mass of the underlying distribution in the regions (polytopes) created by the intersection of bisecting hyperplanes between queried pairs of points. We investigate this fundamental question in both 1-D and higher dimensional settings with noiseless response to comparison queries. We show that the problem is identifiable in 1-D setting and provide recovery guarantees. We also show that the problem is not identifiable for higher dimensional settings. We propose using a regularized recovery for higher dimensional settings and provide guarantees on the total variation distance between the true mass in each of the regions and the distribution learned via regularized constrained optimization problem. We validate our findings through simulations and experiments on real datasets. We also introduce a new dataset for this task collected on a real crowdsourcing platform.

1. Introduction

Learning user preferences via pairwise comparison queries of type “Do you prefer item a or b?” (Figure 1(a)) is widely used in various applications, such as political science, to model voters' political preferences and to predict their voting behavior, and in recommendation systems, to model users' preferences for products or services (Saaty & Vargas, 2012; Fichtner, 1986; Abildtrup et al., 2006; Hopkins & Noel, 2022; Oishi et al., 2005). Let $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ be the known feature representation of concepts (items, objects, images, choices etc.). Preference learning based on ideal point model (Coombs, 1950; Jamieson & Nowak, 2011; Ding, 2016; Singla et al., 2016; Xu & Davenport, 2020; Canal et al.) assumes that there is an *unknown ideal preference point* $\boldsymbol{u} \in \mathcal{X}$ that represents the reference point people use for their preference judgements based on distances. When presented a preference query, “Do you prefer a or b?”, denoted by $\mathrm{Q}(a, b)$, the answer $y_{(a,b)}$ is 1 if the individual prefers the item a over b and -1 otherwise. The ideal point model assumes that $y_{(a,b)} = 1$ if the individual's preference point $\boldsymbol{u}$ is closer to the representation of item a, $\boldsymbol{x}_a$ than item b, $\boldsymbol{x}_b$ (Figure 1(b)). That is, $||\boldsymbol{x}_a - \boldsymbol{u}||_2 < ||\boldsymbol{x}_b - \boldsymbol{u}||_2$. The goal of preference learning is to use the responses for pairwise comparison queries from people and learn the preference point $\boldsymbol{u}$. Once we learn $\boldsymbol{u}$, we can predict the choices people make between new unseen pairs. Many works on preference learning have focused on universal model, where the data from everyone is pooled in together to learn a single preference point on average for the population (Green, 1975; Johnson, 1971; Bhargava et al., 2016). However, different individuals can have different preferences. While one can focus on learning an individual's preference separately, it takes $\mathcal{O}(d \log(d/\varepsilon))$ queries in $\mathbb{R}^d$ to learn an individual's preference point within an ε-ball (Massimino & Davenport, 2021). This can be a prohibitively large number of queries per individual due to cost, cognitive overload or privacy concerns. Therefore, we introduce the problem of *learning the distribution of preferences over a population* via pairwise comparisons using only one response per individual.

*Equal contribution [1]Department of Electrical and Computer Engineering, University of Wisconsin-Madison. Correspondence to: Gokcan Tatli <gtatli@wisc.edu>.

The Many Facets of Preference Learning Workshop at the International Conference on Machine Learning (ICML), Honolulu, Hawaii, USA, 2023. Copyright 2023 by the author(s).



Instructions:

- Thank you for your interest!
- You will be shown 15 questions with pairs of images with footwear.
- **Your task is to pick which of the two footwear you like more based on your preference.**
- You need to answer all the questions.

Question 1 / 15

Footwear 1

Footwear 2

Please click on the footwear that you prefer.

Next

(a)
$\mathbf{x}_i$
$\mathbf{x}_j$
$\mathbf{u}_k$
unknown user preference
“Do you prefer or ?”
preferred if $\mathrm{dist}(\mathbf{x}_i, \mathbf{u}_k) < \mathrm{dist}(\mathbf{x}_j, \mathbf{u}_k)$
preferred if $\mathrm{dist}(\mathbf{x}_i, \mathbf{u}_k) > \mathrm{dist}(\mathbf{x}_j, \mathbf{u}_k)$
(b)
(c)

Figure 1. (a) Example of pairwise comparison query. (b) Ideal point model based response to a comparison query. The colorless circles denote the known representation for items being compared and the human denotes the unknown user preference point. (c) Example of the regions formed by the bisecting hyperplanes between pairs queried and mass of user preferences in different regions.

In this scenario, learning each individual's preference is impossible. In many applications, learning the distribution of user preferences (Figure 1(c)) can be useful for many downstream tasks. E.g., if an ice cream company wants to come up with new flavors, knowing which regions of flavor profiles have more mass is beneficial in the discovery of new ice cream flavors. The learned distribution can thus be helpful in various tasks ranging from finding the more preferred items for cold-start in recommendation, testing difference between preferences of different populations to using the prior to efficiently learn new user preferences. We focus on learning the mass of the underlying distribution in the regions (polytopes) defined by the intersection of the bisecting hyperplanes pairs of items (see Figure 1(c)). If we could have queried $\mathcal{O}(d \log d)$, we can localize the user preference point to one of the regions. So, if we sample a large number of individuals from the population and if we can query each of them with a sufficiently large number of queries, we can build a histogram of the underlying distribution in these regions. However, querying large number of comparison pairs per individual can be prohibitive due to privacy issues, limited interaction of individual with the platform, cognitive overload and cost.

Goal: Develop fundamental understanding on what we can learn about the distribution of user preferences with only one comparison query per individual.

Our contribution: We introduce the novel problem of learning the distribution of user preferences over the population via pairwise comparison queries with only one response per individual and investigate the fundamental questions of identifiability and recovery guarantees leading to the following contributions:

- We show that the problem is identifiable in 1D setting and is not identifiable in higher dimensional setting.
- For the 1D setting, we provide recovery guarantees for the mass in the regions defined by the intersection of hyperplanes at the mid-point of pairs of items used for queries.
- For the higher dimensional setting, we propose to use regularized recovery and provide guarantees on the total variation distance between the true mass in each of the regions and the estimated mass in terms of the regularization parameter and the interplay between the true mass and regularization.
- We provide experiments on synthetic datasets and real datasets that validate our results and observations. We also introduce a new dataset for this task collected on a real crowdsourcing platform [1].

In addition to the above contributions, our work leads to several interesting open questions regarding learning from diverse populations in preference learning.

2. Problem Setup

Let $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ denote known feature representation of items[2]. Under the ideal point model (Coombs, 1950), each individual preference is also modeled as an *unknown* point in the same space. Let $P^\star$ denote the unknown underlying distribution of user preferences. Each individual l has an unknown preference $\boldsymbol{u}_l \in \mathbb{R}^d$. We assume that $\boldsymbol{u}_l \overset{i.i.d.}{\sim} P^\star$. Let $\mathcal{T}$ denote the set of pairs of items (i, j) that are queried. We consider pairwise comparison queries of the form “do you prefer item i or item j?”. We assume that the answer to the pairwise query (i, j) from an individual l is $y_{ij}^{(l)} = 1$ if $||\boldsymbol{x}_i - \boldsymbol{u}_l||_2 < ||\boldsymbol{x}_j - \boldsymbol{u}_l||_2$ and $y_{ij}^{(l)} = -1$ otherwise. Note that each pair of items (i, j) in $\mathcal{T}$ creates a hyperplane perpendicular at the midpoint joining the two items. We will slightly abuse the notation and use $\mathcal{T}$ to denote the set of hyperplanes as well as there is an one-to-one correspondence between each pair and the respective hyperplane. Let h_{ij} denote the hyperplane perpendicular to the midpoint of the pairs (i, j) with $i < j$. The intersection of these hyperplanes carve out regions in $\mathbb{R}^d$ that are polytopes. Let $\mathcal{H}(\mathcal{T})$ denote the set of partitions of $\mathbb{R}^d$ that is created by the set of all hyperplanes in $\mathcal{T}$. Note that for $|\mathcal{T}|$ hyperplanes in $\mathbb{R}^d$, $|\mathcal{H}(\mathcal{T})| = \mathcal{O}(|\mathcal{T}|^d)$ (Buck, 1943). For each pair of items $(i, j) \in \mathcal{T}$, let $q_{ij}^\star$ denote the mass of $P^\star$ on the side of $\boldsymbol{x}_i$ of the hyperplane h_{ij} and $q_{ji}^\star = 1 - q_{ij}^\star$ is the mass to the other side of h_{ij}. Let $\mathbf{q}^\star \in \mathbb{R}^{2|\mathcal{H}(\mathcal{T})|}$ denote the vector that stacks q_{ij}'s for the ordered pairs $(i, j) \in \mathcal{T}$, followed by the corresponding q_{ji}'s. We note $\mathbf{q}^\star$ is the conditional distribution of user preferences on either side of the hyperplanes in $\mathcal{T}$ and can be written as a linear combination of the mass $\mathbf{p}_{\mathcal{H}(\mathcal{T})}^\star$ in the regions via the following linear system of equations,

$$\mathbf{M}\, \mathbf{p}_{\mathcal{H}(\mathcal{T})}^\star = \mathbf{q}^\star, \tag{1}$$

[1] Codes for our methods and synthetic datasets are available in the supplementary material. We will make the anonymized crowdsourced dataset available to the public upon publication.

[2] This is a reasonable assumption, especially with the availability of large pre-trained foundation models.

where $\mathbf{M}$ is a $2|\mathcal{H}(\mathcal{T})| \times \mathcal{O}(|\mathcal{T}|^d)$ binary matrix where in each row, the 1's indicate the regions that contribute to the side of the hyperplane. Each column of $\mathbf{M}$ corresponds to a region (polytope) created by the intersection of the hyperplanes. For each pair $(i, j) \in \mathcal{T}$, we can estimate the mass on either side of the hyperplane h_{ij} by querying a random sample of people. Given these estimates, the question of interest is: can we estimate $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$, the mass in the regions of intersections of hyperplanes in $\mathcal{T}$ induced by the underlying distribution of preferences $P^\star$?

Note that $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ *identifiable* if it is the unique probability vector of size $|\mathcal{H}(\mathcal{T})|$ that gives rise to $\mathbf{q}^\star$. So, $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ is *not identifiable* if there exist a $\mathbf{p} \neq \mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ such that $\mathbf{M}\mathbf{p} = \mathbf{M}\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$.

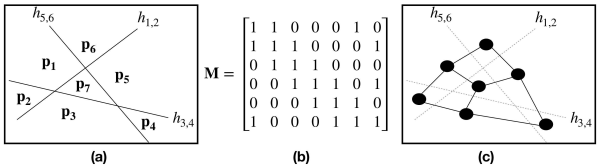

$h_{5,6}$
$h_{1,2}$
$\mathbf{p_6}$
$\mathbf{p_1}$
$\mathbf{p_5}$
$\mathbf{p_7}$
$\mathbf{p_2}$
$\mathbf{p_3}$
$\mathbf{p_4}$
$h_{3,4}$
(a)
$\mathbf{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$
(b)
$h_{5,6}$
$h_{1,2}$
$h_{3,4}$
(c)

Figure 2. (a) Example of regions (polytopes) formed by intersection of three hyperplanes. (b) The corresponding matrix $\mathbf{M}$. (c) The corresponding graph where the regions are the nodes.

Figure 2 shows an example of partition of $\mathbb{R}^2$ with 3 hyperplanes $h_{1,2}, h_{3,4}$ and $h_{5,6}$. With the enumeration of the regions shown in the figure, we can construct the binary matrix $\mathbf{M}$, where the first 3 rows represent regions corresponding to $h_{1,2}$ towards the side of item 1, $h_{3,4}$ towards the side of item 3 and $h_{5,6}$ towards the side of item 5 respectively. Similarly, the last 3 rows represent regions corresponding to the other side of each of the hyperplanes. We also note that each column gives positions of the corresponding region $\mathbf{p}_i$ in terms of hyperplanes $h_{1,2}, h_{3,4}$ and $h_{5,6}$.

3. One Dimensional Setting

We first study the problem in the 1D setting and provide results on identifiability and recovery guarantees.

Identifiability: In 1D setting, $|\mathcal{T}|$ pairs creates $|\mathcal{T}|+1$ intervals. Measuring the fraction of mass on either side of each of the hyperplanes in $\mathcal{T}$ is equivalent to measuring the cumulative distribution function (CDF) of the distribution $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$. The linear system of equations (1) has $|\mathcal{T}|+1$ unknowns and $|\mathcal{T}|+1$ equations. Corresponding binary matrix $\mathbf{M}$ can be written as a concatenation of two triangular matrices where one is a lower triangular and the other is an upper triangular matrix. E.g., 2 pairs create 3 regions and the corresponding matrix $\mathbf{M} = [1,\ 0,\ 0;\ 1,\ 1,\ 0;\ 0,\ 1,\ 1;\ 0,\ 0,\ 1]$. Any such $\mathbf{M}$ is full column rank by construction. Therefore, the linear system of equations $\mathbf{q}^\star = \mathbf{M}\ \mathbf{p}$ has a unique solution in terms of the true $\mathbf{q}^\star$, given by $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{q}^\star$. This is summarized in the following proposition.

Proposition 1. (Identifiability in 1D) In 1D setting, the mass $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ in the regions of intersection of hyperplanes in $\mathcal{T}$ induced by the underlying distribution of preferences can be uniquely determined by only measuring the fraction of the population on either side of each of the hyperplanes in $\mathcal{T}$.

Recovery Guarantees: As we do not have access to true $q^\star$, we have to learn $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ from $\hat{\mathbf{q}}$ estimated by querying pairs in $\mathcal{T}$. We use the following constrained optimization problem:

$$\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} := \arg\min_{\mathbf{p} \geq 0, \mathbf{1}^\top\mathbf{p}=1} \quad \frac{1}{2}||\mathbf{M}\mathbf{p} - \hat{\mathbf{q}}||_2^2.$$

The objective function is strongly convex and therefore the above optimization problem is guaranteed to have a unique solution. We provide recovery guarantees for the 1D noiseless setting below:

Theorem 3.1. *(Recovery in 1D) With probability at least 1−δ, the total variation distance between $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and the recovered mass $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ is bounded as follows,*

$$\mathit{TV}\left(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}\right) \leq \sqrt{\frac{1}{2}\left(1+\frac{1}{|\mathcal{T}|}\right)} \mathit{cond}(\mathbf{M},1) \max\left\{\sqrt{\frac{40}{n}}, \sqrt{\frac{25\log(3/\delta)}{n}}\right\},$$

where cond$(\mathbf{M}, 1)$ is the condition number of $\mathbf{M}$ with respect to l_1-norm, and n is the total number of users queried.

As the total number of users increases, total variation distance betweeen $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ goes to 0. Proof details are available in the appendix.

4. Higher Dimensional Settings

In this section, we discuss the identifiability results and recovery guarantees for $\mathbb{R}^d$ with $d \geq 2$. The details of the proofs are deferred to appendix due to space limitations.

Identifiability: Assuming items and users are supported on $\mathbb{R}^d$, with $d \geq 2$, we note that the number of regions $|\mathcal{H}(\mathcal{T})|$ created by the hyperplanes in $\mathcal{T}$ is of $\mathcal{O}(|\mathcal{T}|^d)$. So, the linear system of equations (1) has order of magnitude more unknowns than the number of equations. We show the following with regards to identifiability.

Proposition 2. For $d \geq 2$, the binary matrix M which of size $2|\mathcal{T}| \times \mathcal{O}(|\mathcal{T}|^d)$ has rank$(\mathbf{M}) = |\mathcal{T}| + 1$ and the solution to the linear system of equations (1) is not unique and hence $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ is not identifiable.

Sparsity: From Proposition 2, for $d \geq 2$, we cannot hope to recover $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ in general. However, M is a fat matrix and a natural question that arises is what if $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ is sparse? That is, if only $k \ll \mathcal{O}(|\mathcal{T}|^d)$ entries of $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ are non-zero. We note that, since $\text{rank}(\mathbf{M}) = |\mathcal{T}| + 1$, for any $k > |\mathcal{T}| + 1$, there exists at least another solution to the equation (1). This follows from noting that the conditional distribution of preferences on either side of each of the hyperplanes can in fact be re-written as convex combination of a set of vectors in $\mathbb{R}^{|\mathcal{T}|+1}$ corresponding to the columns of $\mathbf{M}$, and then using the Carathéodory theorem which guarantees that there always exists a solution with exactly expressed in terms of $|\mathcal{T}| + 1$ columns (see appendix for a detailed discussion of this).

Exploiting sparsity as a side information in signal reconstruction (underlying distribution here) stands for one of the fundamental approaches for a couple of decades in compressed sensing (Candes & Wakin, 2008; Baraniuk, 2007; Elad, 2010). Results from this literature, in particular, from Theorem 2.13 in (Foucart & Rauhut, 2013), any k-sparse solution of an underdetermined linear system of equations is unique if and only if every set of 2k columns of measurement matrix is linearly independent. Given the structure of our matrix $\mathbf{M}$ in equation (1), we show the following.

Proposition 3. For the problem setting in (1), we can always find linearly dependent k columns of $\mathbf{M}$ as long as $k \geq 4$.

As a result, we note even with large sparsity of $k \geq 4$, uniqueness cannot be guaranteed for all k-sparse distributions. For example, consider the partition in the Figure 2(a). We suppose that the solution is 4-sparse and $\mathbf{p}^\star = [0.3,\ 0,\ 0,\ 0,\ 0.4,\ 0.1,\ 0.2]^T$. Therefore, we have $\mathbf{M}\mathbf{p}^\star = \mathbf{q}$, where $\mathbf{q} = [0.4,\ 0.5,\ 0,\ 0.6,\ 0.5,\ 1]^T$. However, $\mathbf{q} = \mathbf{M}\mathbf{p}$ holds also for $\mathbf{p} = [0.2,\ 0,\ 0,\ 0,\ 0.3,\ 0.2,\ 0.3]^T$ Additionally, we make the following remark based on the fact that $\mathbf{M}$ is a column-regular matrix, i.e. each column of $\mathbf{M}$ has exactly the same number of 1's.

Remark 4.1. Robust Null Space Property (RNSP) has been proposed as a sufficient condition for basis pursuit approach (a popular recovery algorithm in compressed sensing literature) (Foucart & Rauhut, 2013; Foucart, 2014). Recently, Lotfi et al. (Lotfi & Vidyasagar, 2020) proposed sufficient conditions for a column-regular binary matrix to achieve RNSP, which are the best sufficient conditions for column-regular binary matrices to the best of our knowledge. According to Theorem 9 in (Lotfi & Vidyasagar, 2020), a column-regular binary matrix satisfies RNSP when k< d_L/ρ, where d_L is the number of 1's in each column and ρ is the maximum inner product among columns. Our binary matrix $\mathbf{M}$ is column-regular binary matrix with $|\mathcal{T}|$ 1's in each column. Since there are neighboring regions, i.e., regions that has only one different coordinate, maximum inner product among columns is $|\mathcal{T}| - 1$. Therefore, RNSP is achieved when $k = 1$.

Bounds on the mass in the regions: Given the non-identifiability of $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ in higher dimensions, we cannot hope to recover it from binary answers to pairwise comparison queries. Here, we show that we can obtain lower and upper bounds for each entry of $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ from the estimated $\hat{\mathbf{q}}$ without requiring any additional assumptions. To state these bounds, we need some notations. Let $\mathbf{M}_{i,:}$ be the i-th row of $\mathbf{M}$ (corresponding to the $i-$th hyperplane) and let (a_i, b_i) denote the pair queried corresponding to this row. Let $\hat{\mathbf{q}}_{a_i,b_i}$ denote the estimated mass on the side of a_i of the i-th hyperplane. Let $\mathcal{K}_j$ denote the position of rows of $\mathbf{M}$ whose j-th column entry is 1. $\hat{\mathbf{Q}}_0^j := [\min_{i\in\mathcal{K}_1} \hat{\mathbf{q}}_{a_ib_i}, \ldots \min_{i\in\mathcal{K}_{j-1}} \hat{\mathbf{q}}_{a_ib_i},\ 0,\ \min_{i\in\mathcal{K}_{j+1}} \hat{\mathbf{q}}_{a_ib_i}, \ldots \min_{i\in\mathcal{K}_{|\mathcal{H}(\mathcal{T})|}} \hat{\mathbf{q}}_{a_ib_i}]^T$.

Proposition 4. With probability at least $1 - \delta$, each entry of $\mathbf{p}_{\mathcal{H}(\mathcal{T})}$ can be bounded below and above as follows:

$$\max\left\{0, \max_{i\in\mathcal{K}_j} \hat{\mathbf{q}}_{a_ib_i} - \mathbf{M}_{i,:}^T \hat{\mathbf{Q}}_0^j - (|\mathbf{M}_{i,:}|_1 + 1)\gamma\right\} \leq \mathbf{p}^\star_{\mathcal{H}(\mathcal{T})_j} \leq \min_{i\in\mathcal{K}_j} \hat{\mathbf{q}}_{a_ib_i} + \gamma,$$

where $\gamma = \sqrt{\frac{\log(4|\mathcal{T}|/\delta)}{2n_p}}$ and n_p is the number of people answering each pairwise query.

Graph Regularization: In the face of non-identifiability, additional structural assumptions are needed for learning the mass in the regions, i.e., polytopes, $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$. We note that while $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ is a $\mathcal{O}(|\mathcal{T}|^d)$-dimensional probability vector, the entries correspond to mass in regions that have a geometry in the space $\mathcal{X} \subseteq \mathbb{R}^d$ (recall Figure 2(a)) that gives a notion of near-by and far-away regions. We construct a connected undirected graph with the polytopes as the nodes and two nodes are connected by an edge if they share a $(d-1)$-dimensional face between them (see Figure 2(c)). We propose using a graph regularizer (normalized by volume to account for differences in the sizes of the regions) to recover $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$. Intuitively, this means that we expect preferences to accumulate in spatially nearby regions (Figure 1(c)). Several works in signal recovery have used graph regularization to exploit local invariance in data as a side information and find a locally invariant representation of the data (Belkin & Niyogi, 2001; Cai et al., 2011; Hadsell et al., 2006).

We note that this proposed graph structure can be constructed using the matrix $\mathbf{M}$. Recall that the rows of $\mathbf{M}$ correspond to hyperplanes and the columns correspond to the regions (polytopes) in $\mathcal{H}(\mathcal{T})$ providing a binary encoding for them by construction. That is, each entry of a given column of $\mathbf{M}$ determines which side of a hyperplane the corresponding region is located on. Therefore, there exists an edge between nodes corresponding to the regions that has only two different entries in their hyperplane coordinates, i.e., only if one pairwise comparison yields opposite results. Accordingly, neighboring regions have common $(d-1)$-dimensional faces in between.

We define the weight matrix $\mathbf{W}$ for the graph regularization as

$$\mathbf{W}_{i,j} = \frac{\|\mathbf{M}_{:,i} - \mathbf{M}_{:,j}\|_1^{-1}}{\alpha_i \alpha_j}, \tag{2}$$

where $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_{|\mathcal{H}(\mathcal{T})|}]^T$ represent volumes of regions with corresponding mass $\mathbf{p} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{|\mathcal{H}(\mathcal{T})|}]^T$ respectively.

Each entry of $\mathbf{W}$ is the weighted inverse of the Hamming distance between corresponding nodes i and j, where $\mathbf{M}_{:,i}$ is the i-th column of the matrix $\mathbf{M}$. Furthermore, since the regions in $\mathcal{H}(\mathcal{T})$ are not equal in sizes, we normalize with the volumes of the regions.

One can similarly construct different weight matrices for regularization as long as the entries are inversely proportional to the distances between nodes. Heat kernel weighting (Belkin & Niyogi, 2001), 0-1 weighting (Cai et al., 2011) are some of the widely used ones in the literature. We use $\mathbf{W}$ defined above and form following graph Laplacian regularizer:

$$\frac{1}{2}\sum_{i=1}^{|\mathcal{H}(\mathcal{T})|}\sum_{j=1}^{|\mathcal{H}(\mathcal{T})|} |\mathbf{p}_i - \mathbf{p}_j|^2 \mathbf{W}_{i,j} =: \mathbf{p}^T\mathbf{D}\mathbf{p} - \mathbf{p}^T\mathbf{W}\mathbf{p} \tag{3}$$

$$=: \mathbf{p}^T\mathbf{L}\mathbf{p}, \tag{4}$$

where $\mathbf{D}_{i,i} = \sum_{j=1}^{|\mathcal{H}(\mathcal{T})|} \mathbf{W}_{i,j}$, $\mathbf{D}_{i,j} = 0$ when $i \neq j$ and $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Using this regularizer, we propose following optimization problem for recovering $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$:

$$\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} := \arg\min_{\mathbf{p}\geq 0, \mathbf{1}^\top\mathbf{p}=1} \frac{1}{2}||\mathbf{M}\mathbf{p} - \hat{\mathbf{q}}||_2^2 + \frac{\lambda}{2}\mathbf{p}^T\mathbf{L}\mathbf{p} \tag{5}$$

$$:= \arg\min_{\mathbf{p}\geq 0, \mathbf{1}^\top\mathbf{p}=1} \frac{1}{2}||\mathbf{R}\mathbf{p} - \hat{\mathbf{b}}||_2^2, \tag{6}$$

where $\mathbf{R}^T\mathbf{R} = \mathbf{M}^T\mathbf{M} + \lambda\mathbf{L}$ by Cholesky decomposition and $\hat{\mathbf{b}} = \mathbf{R}^{-T}\mathbf{M}^T\hat{\mathbf{q}}$.

The regularizer in equation (4) encourages the changes in nearby regions to be smooth, which is similar to the local invariance property considered in (Belkin & Niyogi, 2001; Cai et al., 2011; Hadsell et al., 2006). Weighted Laplacian regularizer $\mathbf{L}$ imposes a penalty on $\mathbf{p}$ in such a way that potential values correlated with eigenvectors of $\mathbf{L}$ are diminished. Therefore, eigenvectors corresponding to larger eigenvalues cause more penalty. Note that the eigenvectors of $\mathbf{L}$ are mutually orthogonal by spectral theorem. So, we conclude that orthogonal eigenvectors of nonzero eigenvalues force the potential solution to be close to the distribution $\overline{\alpha}$ by diminishing possible directions other than α, where $\overline{\alpha}$ is the normalized α.

We provide the following recovery guarantee using the solution to the proposed regularized optimization problem.

Theorem 4.2. *The convex optimization problem in (6) has a unique solution. Furthermore, with probability at least $1-\delta$, the total variation distance between $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and the recovered mass $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ is bounded as follows,*

$$TV(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}) \leq \frac{\lambda}{2}\sqrt{|\mathcal{H}(\mathcal{T})|}\|\mathbf{R}^{-1}\|_2\|\mathbf{L}\|_2\,\|\mathbf{p}^\star - \overline{\alpha}\|_2 + \sqrt{\frac{|\mathcal{H}(\mathcal{T})||\mathcal{T}|}{2}}\|\mathbf{R}^{-1}\|_2\|\mathbf{M}\|_2 \max\left\{\sqrt{\frac{40}{n}}, \sqrt{\frac{25\log(3/\delta)}{n}}\right\},$$

where n is the total number of users.

The maximum singular value of $\mathbf{L}$ and the minimum singular value of $\mathbf{R}$ play an important role on determining the first component of the bound. On the other hand, the second component tends towards 0, as the number of users increases.

5. Experimental Results

We evaluate the proposed approaches for both simulated and real datasets. We quantify the total variation distance (TV) and Wasserstein distance between $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and the recovered mass in partitions $\mathcal{H}(\mathcal{T})$. For 1D setting, we use Wasserstein-1 distance and for higher dimensional settings, we use the graph Wasserstein distance with normalized cost matrix is written as follows,

$$\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}) := \min_{\mathbf{K}\geq 0,\ \mathbf{K}\mathbf{1}=\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})},\ \mathbf{K}^T\mathbf{1}=\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}} \sum_{i=1}^{|\mathcal{H}(\mathcal{T})|}\sum_{j=1}^{|\mathcal{H}(\mathcal{T})|} \mathbf{K}_{i,j}\mathbf{C}_{i,j},$$

where $\mathbf{C}_{i,j}$ is the ratio of distance between nodes i and j to the maximum length on the graph induced by matrix $\mathbf{M}$. Note that the total variation distance does not differentiate between whether mass is moved between neighbor regions or any far away region. Whereas Wasserstein distances take into account the geometry and hence distinguish between these scenarios.

For simulations, we consider the following four distributions as true user distributions: uniform, Gaussian, a mixture of two Gaussians, and a mixture of three Gaussians. We also consider two types of noises. (a) $\mathrm{Bernoulli}(p_{\text{flip}})$ that flips a simulated user's answer with probability p_{flip}. (b) flipping the answer of user $\boldsymbol{u}$ for pair $\boldsymbol{x}_a, \boldsymbol{x}_b$ with probability $\frac{1}{1+e^{-cd_{\text{diff}}}}$, where c is a scaling factor and $d_{\text{diff}} =$

$-\text{abs}(\text{dist}(\boldsymbol{x}_a, \boldsymbol{u}) - \text{dist}(\boldsymbol{x}_b, \boldsymbol{u}))$. We sample $m = 5$ items uniformly at random from $[-1, 1]^d$ and subsample n_h hyperplanes repeating 10 times from all possible hyperplanes. Similarly, for each set of pairwise comparisons corresponding to the hyperplanes, we sample users from the underlying distribution repeating 10 times. Then, we use CVXPY to solve optimization problems and run all simulations on Python 3.9.

6. Related Works

We briefly review related work, deferring a more detailed discussion to the Appendix. Preference learning based on ideal point model (Coombs, 1950; Jamieson & Nowak, 2011; Ding, 2016; Singla et al., 2016; Xu & Davenport, 2020; Massimino & Davenport, 2021; Canal et al.) has been studied by several works. A key limitation of these works is that they either focus on learning an individual preference by making many queries per individual or assume homogeneity and learn a single preference point using data from all the users. A recent work by Tatli et al. (Tatli et al., 2022) introduced the problem of learning distribution of preferences over the population in 1D setting using distance queries rather than pairwise comparisons.

Another line of work in preference learning involves ranking based models, e.g., Bradley-Terry-Luce model (Bradley & Terry, 1952; Luce, 1959), stochastic transitivity models (Shah et al., 2016), that focus on finding ranking of m items or finding top-k items by pairwise comparisons (Hunter, 2004; Kenyon-Mathieu & Schudy, 2007; Braverman & Mossel, 2007; Negahban et al., 2012; Eriksson, 2013; Rajkumar & Agarwal, 2014; Shah & Wainwright, 2017). Ranking m items in these settings requires $\mathcal{O}(m \log m)$ queries. Under the ideal point based models, this query complexity reduces to $\mathcal{O}(d \log m)$, where d is the dimension of the domain of representations which is usually much smaller than the number of items being ranked (Jamieson & Nowak, 2011). This is due to the fact that once the preference point is learned, it can then be used to predict rankings of new items without needing more comparisons. We focus on preference learning that is based on ideal point model.

7. Conclusions and Future Work

We propose a novel problem of learning distribution of user preferences from pairwise comparison queries. We focus on fundamental questions regarding what we can learn about the underlying distribution from a single query per user. We show that the problem is identifiable in 1D setting and provide recovery guarantees under the total variation distance. We show that this problem is not identifiable in dimensions $d \geq 2$. We provide upper and lower bounds on the masses in the regions (polytopes) formed by the intersecting hyperplanes corresponding to the queried pairs. We proposed using graph regularization for recovery of the masses in these regions and provide bound on the total variation distance between the true distribution and the estimated distribution. We validate these fundamental results on extensive numerical simulations. Furthermore, we show the efficacy of the proposed methods on real datasets. As a byproduct of this work, we introduce two new datasets for learning distribution of user preferences. In the future, we would like to mathematically characterize how large the set of underlying preference distribution that lead to the same answers to pairwise queries in terms of the TV and Wasserstein distances. We would also like to further explore what other structures on the underlying distributions make it amenable to overcome non-identifiability and develop recovery guarantees under the graph Wasserstein distance which takes into account the geometry of the feature space.

References

Abildtrup, J., Audsley, E., Fekete-Farkas, M., Giupponi, C., Gylling, M., Rosato, P., and Rounsevell, M. Socio-economic scenario development for the assessment of climate change impacts on agricultural land use: a pairwise comparison approach. *Environmental science & policy*, 9(2):101–115, 2006.

AMT. Amazon mechanical turk. https://www.mturk.com/. Accessed: 2023-04-27.

Baraniuk, R. G. Compressive sensing [lecture notes]. *IEEE Signal Processing Magazine*, 24(4):118–121, 2007. doi: 10.1109/MSP.2007.4286571.

Belkin, M. and Niyogi, P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Dietterich, T., Becker, S., and Ghahramani, Z. (eds.), *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001. URL https://proceedings.neurips.cc/paper/2001/file/f106b7f99d2cb30c3db1c3cc0fde9ccb-Paper.pdf.

Bhargava, A., Ganti, R., and Nowak, R. D. Active algorithms for preference learning problems with multiple populations. *arXiv: Machine Learning*, 2016.

Bradley, R. A. and Terry, M. E. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Braverman, M. and Mossel, E. Noisy sorting without resampling. *arXiv preprint arXiv:0707.1051*, 2007.

Buck, R. C. Partition of space. *The American Mathematical Monthly*, 50(9):541–544, 1943. doi: 10.1080/00029890.1943.11991447. URL https://doi.org/10.1080/00029890.1943.11991447.

Cai, D., He, X., Han, J., and Huang, T. S. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011. doi: 10.1109/TPAMI.2010.231.

Canal, G., Mason, B., Vinayak, R. K., and Nowak, R. D. One for all: Simultaneous metric and preference learning over multiple users. In *Advances in Neural Information Processing Systems, 2022*.

Candes, E. J. and Wakin, M. B. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2): 21–30, 2008. doi: 10.1109/MSP.2007.914731.

Coombs, C. H. Psychological scaling without a unit of measurement. *Psychological review*, 57(3):145, 1950.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Devroye, L. The equivalence of weak, strong and complete convergence in l1 for kernel density estimates. *The Annals of Statistics*, pp. 896–904, 1983.

Ding, C. Evaluating change in behavioral preferences: Multidimensional scaling single-ideal point model. *Measurement and Evaluation in Counseling and Development*, 49 (1):77–88, 2016.

Elad, M. Sparse and redundant representations: From theory to applications in signal and image processing, 2010.

Eriksson, B. Learning to top-k search using pairwise comparisons. In *Artificial Intelligence and Statistics*, pp. 265–273. PMLR, 2013.

Fichtner, J. On deriving priority vectors from matrices of pairwise comparisons. *Socio-Economic Planning Sciences*, 20(6):341–345, 1986.

Foucart, S. Stability and robustness of ℓ1-minimizations with weibull matrices and redundant dictionaries. *Linear Algebra and its Applications*, 441:4–21, 2014. ISSN 0024-3795. doi: https://doi.org/10.1016/j.laa.2012.10.003. URL https://www.sciencedirect.com/science/article/pii/S0024379512007008. Special Issue on Sparse Approximate Solution of Linear Systems.

Foucart, S. and Rauhut, H. A mathematical introduction to compressive sensing. In *Applied and Numerical Harmonic Analysis*, 2013.

Green, P. E. Marketing applications of mds: Assessment and outlook. *Journal of Marketing*, 39(1):24–31, 1975. ISSN 00222429. URL http://www.jstor.org/stable/1250799.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 1735–1742, 2006. doi: 10.1109/CVPR.2006.100.

Hopkins, D. J. and Noel, H. Trump and the shifting meaning of "conservative": Using activists' pairwise comparisons to measure politicians' perceived ideologies. *American Political Science Review*, 116(3):1133–1140, 2022.

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

IMDB. Imdb non-commercial dataset. https://developer.imdb.com/non-commercial-datasets/, 2023. URL https://developer.imdb.com/non-commercial-datasets/. Accessed: 2023-5-12.

Jamieson, K. G. and Nowak, R. Active ranking using pairwise comparisons. *Advances in neural information processing systems*, 24, 2011.

Johnson, R. M. Market segmentation: A strategic management tool. *Journal of Marketing Research*, 8(1):13–18, 1971. ISSN 00222437. URL http://www.jstor.org/stable/3149720.

Kenyon-Mathieu, C. and Schudy, W. How to rank with few errors. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pp. 95–103, 2007.

Lotfi, M. and Vidyasagar, M. Compressed sensing using binary matrices of nearly optimal dimensions. *IEEE Transactions on Signal Processing*, 68:3008–3021, 2020. doi: 10.1109/TSP.2020.2990154.

Luce, R. D. *Individual choice behavior: A theoretical analysis*. New York: Wiley, 1959.

Massimino, A. K. and Davenport, M. A. As you like it: Localization via paired comparisons. *J. Mach. Learn. Res.*, 22:186–1, 2021.

Negahban, S., Oh, S., and Shah, D. Iterative ranking from pair-wise comparisons. *Advances in neural information processing systems*, 25, 2012.

Oishi, S., Hahn, J., Schimmack, U., Radhakrishan, P., Dzokoto, V., and Ahadi, S. The measurement of values across cultures: A pairwise comparison approach. *Journal of research in Personality*, 39(2):299–305, 2005.

OpenAI. text-embedding-ada-002. https://openai.com/blog/new-and-improved-embedding-model, 2023. URL https://openai.com/blog/new-and-improved-embedding-model. Accessed: 2023-5-12.

Palmer, S. E. and Schloss, K. B. An ecological valence theory of human color preference. *Proceedings of the National Academy of Sciences*, 107(19):8877–8882, 2010.

Palmer, S. E., Schloss, K. B., and Sammartino, J. Visual aesthetics and human preference. *Annual review of psychology*, 64:77–107, 2013.

Poignard, C., Pereira, T., and Pade, J. P. Spectra of laplacian matrices of weighted graphs: Structural genericity properties. *SIAM Journal on Applied Mathematics*, 78 (1):372–394, 2018. doi: 10.1137/17M1124474. URL https://doi.org/10.1137/17M1124474.

Rajkumar, A. and Agarwal, S. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *International conference on machine learning*, pp. 118–126. PMLR, 2014.

Saaty, T. L. and Vargas, L. G. The possibility of group choice: pairwise comparisons and merging functions. *Social Choice and Welfare*, 38(3):481–496, 2012.

Schneider, R. *Convex Bodies: The Brunn–Minkowski Theory*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2 edition, 2013. doi: 10.1017/CBO9781139003858.

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

Shah, N. B. and Wainwright, M. J. Simple, robust and optimal ranking from pairwise comparisons. *The Journal of Machine Learning Research*, 18(1):7246–7283, 2017.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Singla, A., Tschiatschek, S., and Krause, A. Actively learning hemimetrics with applications to eliciting user preferences. In *International Conference on Machine Learning*, pp. 412–420. PMLR, 2016.

Tatli, G., Nowak, R., and Vinayak, R. K. Learning preference distributions from distance measurements. In *2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1–8, 2022. doi: 10.1109/Allerton49937.2022.9929404.

Tomatoes, R. Rotten tomatoes. https://www.rottentomatoes.com/, 2023. URL https://www.rottentomatoes.com/. Accessed: 2023-5-12.

Wang, Y., Huang, H., Rudin, C., and Shaposhnik, Y. Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. *Journal of Machine Learning Research*, 22(201):1–73, 2021a. URL http://jmlr.org/papers/v22/20-1061.html.

Wang, Y., Huang, H., Rudin, C., and Shaposhnik, Y. Understanding how dimension reduction tools work: an empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. *The Journal of Machine Learning Research*, 22(1):9129–9201, 2021b.

Wikipedia. Wikipedia. https://en.wikipedia.org/, 2023. URL https://en.wikipedia.org/. Accessed: 2023-5-12.

Xu, A. and Davenport, M. Simultaneous preference and metric learning from paired comparisons. *Advances in Neural Information Processing Systems*, 33:454–465, 2020.

Yu, A. and Grauman, K. Fine-grained visual comparisons with local learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 192–199, 2014.

Yu, A. and Grauman, K. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5570–5579, 2017.

A. Limitations

We study the novel problem of *learning populations of preferences via pairwise comparison queries* when we are limited to making one query per individual. We show that the problem is identifiable in 1D setting and provide recovery guarantees. Further, we show that the problem is not identifiable in dimensions $d \geq 2$. Linear system of equations in (1) is underdetermined in dimensions $d \geq 2$. So, we cannot recover $\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})}$ in dimensions $d \geq 2$. Therefore, we propose using graph regularization for recovery of masses in $\mathcal{H}(\mathcal{T})$ and provide recovery guarantees. Our recovery guarantees are limited to the noiseless setting. For noisy settings, we show simulation results that are promising. Furthermore, the suitability of the regularizer depends on the property of underlying distribution of preferences. We have explored one such regularization technique in this work. Theoretical analysis of noisy setting and other regularizers suited for different properties would be interesting to study in the future.

In this work, we focus on the case where we can only make one comparison query per individual. On the other end, if we can make $\tilde{\mathcal{O}}(d)$ queries per individual, we can estimate individual preferences. We expect there is a trade-off between these two regimes, that is, single query per individual to enough queries to learn individual preference points, in terms of information gain regarding the underlying distribution of preferences, which is left to future work for further investigation.

B. Proofs

B.1. Proof of Theorem 3.1

We recall that $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ is the solution to the constrained least square optimization problem with unit simplex constraint in Section 3. Then, we note that $\mathbf{M}\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ is the projection of $\hat{\mathbf{q}}$ onto the closed convex set $C_{\mathbf{M}}$ under ℓ_2 distance, which we call $\text{Proj}_{C_M}(\hat{\mathbf{q}})$, where

$$C_{\mathbf{M}} := \text{conv}(\mathbf{M}e_1, \ldots, \mathbf{M}e_m).$$

Therefore, we can write

$$\begin{aligned} ||\mathbf{p}^{*}_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}||_2 &= ||\mathbf{M}^{\dagger}(\mathbf{q}^{\star} - \text{Proj}_{C_M}(\hat{\mathbf{q}}))||_2 \\ &\leq ||\mathbf{M}^{\dagger}||_2||\mathbf{q}^{\star} - \text{Proj}_{C_M}(\hat{\mathbf{q}})||_2 \\ &\overset{(a)}{\leq} ||\mathbf{M}^{\dagger}||_2||\mathbf{q}^{\star} - \hat{\mathbf{q}}||_2 \\ &\leq ||\mathbf{M}^{\dagger}||_2||\mathbf{q}^{\star} - \hat{\mathbf{q}}||_1, \end{aligned} \tag{7}$$

where the inequality (a) is due the fact that the projection onto closed convex sets is contracting (Thm. 1.2.2.(Schneider, 2013)). Then, we note that $2\ \text{TV}(\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}) = ||\mathbf{p}^{*}_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}||_1$, and use $l_1 - l_2$ norm inequality to obtain the following from (7),

$$\begin{aligned} ||\mathbf{p}^{*}_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}||_1 &\leq \sqrt{|\mathcal{T}|+1}||\mathbf{p}^{*}_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}||_2 \\ &\leq \sqrt{|\mathcal{T}|+1}||\mathbf{M}^{\dagger}||_{\mathbf{2}}||\hat{\mathbf{q}} - \mathbf{q}^{\star}||_{\mathbf{2}} \\ &\leq \sqrt{2|\mathcal{T}|(|\mathcal{T}|+1)}||\mathbf{M}^{\dagger}||_{\mathbf{1}}||\hat{\mathbf{q}} - \mathbf{q}^{\star}||_{\mathbf{1}} \\ &= \sqrt{2|\mathcal{T}|(|\mathcal{T}|+1)}\frac{\text{cond}(\mathbf{M}, \mathbf{1})}{||\mathbf{M}||_{\mathbf{1}}}||\hat{\mathbf{q}} - \mathbf{q}^{\star}||_1. \end{aligned}$$

Note that the term $||\hat{\mathbf{q}} - \mathbf{q}^{\star}||_1$ is the sum of l_1- distances between the empirical and the true conditional distributions of pairwise comparisons for set of pairs of items in $\mathcal{T}$. We then note that $||\mathbf{M}||_{\mathbf{1}} = |\mathcal{T}|$ and use the bound on the l_1-norm between empirical distribution and the true distribution for discrete distributions on finite support from following Lemma to complete the proof.

Lemma B.1. *(Lemma 3 in (Devroye, 1983)) Let $\mathbf{v} \in \mathbb{R}^z$ be the probabilities corresponding to a multinomial random vector with support size z. Let $\hat{\mathbf{v}}$ denote the empirical estimate of these probability values from N i.i.d. samples drawn from $\mathbf{p}$. Then, for all $\varepsilon \geq \sqrt{20z/N}$,*

$$Pr\left(||\hat{\mathbf{v}} - \mathbf{v}||_1 > \varepsilon\right) \leq 3e^{-N\varepsilon^2/25}.$$

B.2. Proof of Proposition 2

We first note that we can write following

$$\mathbf{e}_j = \prod_{i \in \mathcal{K}_j}^{\odot} \mathbf{M}_{i,:}, \quad j = 1, \dots, |\mathcal{H}(\mathcal{T})|. \tag{8}$$

Then, considering the structure of matrix $\mathbf{M}$, we note that

$$\sum_{i=1}^{2|\mathcal{T}|} \lambda_i \mathbf{M}_{i,:} = \sum_{i=1}^{|\mathcal{T}|} (\lambda_i - \lambda_{|\mathcal{T}|+i}) \mathbf{M}_{i,:} + \left(\sum_{i=1}^{|\mathcal{T}|} \lambda_{|\mathcal{T}|+i} \right) \mathbf{1}.$$

If $\sum_{i=1}^{|\mathcal{T}|}(\lambda_i - \lambda_{|\mathcal{T}|+i})\mathbf{M}_{i,:} + \left(\sum_{i=1}^{|\mathcal{T}|} \lambda_{|\mathcal{T}|+i}\right)\mathbf{1} = 0$ holds only when $\lambda_i - \lambda_{|\mathcal{T}|+i} = 0$ for all $i = 1, \dots, |\mathcal{T}|$ and $\sum_{i=1}^{|\mathcal{T}|} \lambda_{|\mathcal{T}|+i} = 0$, we can claim that $\mathbf{1}$ and $\mathbf{M}_{i,:}$'s for $i = 1, \dots, |\mathcal{T}|$ are linearly independent. Therefore, we suppose that

$$\sum_{i=1}^{2|\mathcal{T}|} \lambda_i \mathbf{M}_{i,:} = \sum_{i=1}^{|\mathcal{T}|} (\lambda_i - \lambda_{|\mathcal{T}|+i}) \mathbf{M}_{i,:} + \left(\sum_{i=1}^{|\mathcal{T}|} \lambda_{|\mathcal{T}|+i} \right) \mathbf{1} = 0.$$

Now, we take $|\mathcal{T}|$-th power of the left-hand side with respect to Hadamard product and write it as follows:

$$\left(\sum_{i=1}^{2|\mathcal{T}|} \lambda_i \mathbf{M}_{i,:} \right)^{\odot|\mathcal{T}|} = 0 \tag{9}$$

Considering results of all products in given expression, we can write following Lemma.

Lemma B.2. *Given the binary matrix $\mathbf{M} \in \{0,1\}^{2|\mathcal{T}| \times |\mathcal{H}(\mathcal{T}))|}$ in (1) and real coefficients λ_i's, we can write following*

$$\left(\sum_{i=1}^{2|\mathcal{T}|} \lambda_i \mathbf{M}_{i,:} \right)^{\odot|\mathcal{T}|} = \sum_{j=1}^{2|\mathcal{T}|} \left(\sum_{i \in \mathcal{K}_j} \lambda_i \right)^{|\mathcal{T}|} \mathbf{e}_j,$$

where $\mathcal{K}_j$ is the position of rows of $\mathbf{M}$ whose j-th entry is 1 and $\mathbf{e}_j$'s are standard basis vectors.

Lemma B.3. *Given the binary matrix $\mathbf{M}$ in Section 2, for any $j \leq |\mathcal{T}|$, we can find two columns $\mathbf{M}_{:,j_1}$ and $\mathbf{M}_{:,j_2}$ such that only j-th and ($|\mathcal{T}|$+j)-th entries of $\mathbf{M}_{:,j_1}$ and $\mathbf{M}_{:,j_2}$ differ.*

Proof: *Each hyperplane has to form neighboring regions by construction. Therefore, there exists two columns $\mathbf{M}_{:,j_1}$ and $\mathbf{M}_{:,j_2}$ such that only j-th and ($|\mathcal{T}|$+j)-th entries differ. To understand it better, we can consider a scenario where we delete j-th row of the matrix $\mathbf{M}$ and call $\mathbf{M}^j$ to this new matrix. $\mathbf{M}^j$ has to have a pair of same columns. Otherwise, we would conclude that j-th hyperplane does not form new regions, which is not possible by construction, when we consider adding one hyperplane at a time to end up with final partition. We can also refer to the fact that each hyperplane has to divide at least one previous region into two, when that specific hyperplane is added.*

Then, from Lemma B.2, (9) yields that

$$\sum_{j=1}^{2|\mathcal{T}|} \left(\sum_{i \in \mathcal{K}_j} \lambda_i \right)^{|\mathcal{T}|} \mathbf{e}_j = 0,$$

which happens only if

$$\sum_{i \in \mathcal{K}_j} \lambda_i = 0, \quad j = 1, \dots, |\mathcal{H}(\mathcal{T})|,$$

since standard basis vectors are linearly independent. From Lemma B.3, it follows that we can find two numbers j_1 and j_2 for all $j = 1, \dots, |\mathcal{H}(\mathcal{T})|$ such that

$$\sum_{i \in \mathcal{K}_{j_1}} \lambda_i = \sum_{i \in \mathcal{K}_{j_2}} \lambda_i = 0,$$

where $j \in \mathcal{K}_{j_1}$, $|\mathcal{T}| + j \in \mathcal{K}_{j_2}$ and $\mathcal{K}_{j_1} \backslash \{j\} = \mathcal{K}_{j_2} \backslash \{|\mathcal{T}| + j\}$. Therefore, we conclude that $\lambda_j = \lambda_{|\mathcal{T}|+j}$ for all $j = 1, \dots, |\mathcal{H}(\mathcal{T})|$. Now, (9) implies $\sum_{i=1}^{|\mathcal{T}|} \lambda_{|\mathcal{T}|+i} = 0$, which confirms the claim that $\text{rank}(\mathbf{M}) = |\mathcal{T}| + 1$.

For the nonuniquness of the solution to the linear system of equations (1), we refer to the discussion below(Section B.3), where we argue that the solution is not unique even for sparse cases, and complete the proof of Proposition 2.

B.3. Sparsity

We first recall that half of the rows among $2|\mathcal{T}|$ rows of $\mathbf{M}$ reflect the mass on the other side of each hyperplane. Basically, adding a row of all ones makes half of the rows redundant, since the rows representing the mass on the other side of each hyperplane are just flipped versions of rows representing the mass on the first side, i.e., $\mathbf{M}_{i+|\mathcal{T}|,:} = \mathbf{1}^T - \mathbf{M}_{i,:}$. We call $\mathbf{M}_{\text{half}}$ to the simplified version of $\mathbf{M}$. Then, we note that $\text{rank}(\mathbf{M}_{\text{half}}) = \text{rank}(\mathbf{M}) = |\mathcal{T}| + 1$ from Proposition 2. Therefore, we cannot make further simplifications on $\mathbf{M}$ to get redundant rows.

Now, we consider the simplified version $\mathbf{M}_{\text{half}}$ and recall that any solution $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ to the problem setting in (1) has to be in the probability simplex. Therefore, all possible $\hat{\mathbf{q}}_{\text{half}}$ vectors belong to the convex hull of columns of matrix $\mathbf{M}_{\text{half}}$, which we call $\text{conv}(\mathbf{M}_{\text{half}})$. Then, we apply Carathéodory's Theorem and write following expression. *Each element in conv*$(\mathbf{M}_{\textit{half}})$ *can be written as a convex combination of at most* $|\mathcal{T}| + 1$ *columns of* $\mathbf{M}_{\textit{half}}$. We can easily observe that the same property also applies to $\text{conv}(\mathbf{M})$ and $\hat{\mathbf{q}}$, as they share a one-to-one correspondence with $\mathbf{M}_{\text{half}}$ and $\hat{\mathbf{q}}_{\text{half}}$, respectively.

B.4. Proof of Proposition 3

Given any two neighboring regions in the partition $\mathcal{H}(\mathcal{T})$, we suppose that $\mathbf{M}_{:,i_1}$ and $\mathbf{M}_{:,i_2}$ are corresponding columns to those regions, where only j-th hyperplane differ in between. We observe that it is possible to find another pair of columns, satisfying the same condition, separated solely by the j-th hyperplane as long as there exists another hyperplane that intersects with j-th hyperplane. Therefore, we can find linearly dependent 4 columns except the trivial case when all hyperplanes are parallel to each other. The problem setting boils down to the 1D setting, when all hyperplanes are parallel. The binary measurement matrix $\mathbf{M}$ becomes full rank and we can uniquely recover underlying distribution of regions in the partition $\mathcal{H}(\mathcal{T})$ separated by parallel hyperplanes.

B.5. Proof of Proposition 4

Recall that $\mathbf{M}_{i,:}$ and (a_i, b_i) are the row and the pair corresponding to i-th hyperplane. $\mathbf{q}^\star_{a_i,b_i}$ denotes the true mass on the side of a_i of the i-th hyperplane and $\mathcal{K}_j$ is the position of rows of $\mathbf{M}$ whose j-th column entry is 1. $\mathbf{q}^\star_{a_i,b_i}$ has $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})_j}$, j-th entry of $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$, as a nonnegative summand when $i \in \mathcal{K}_j$. Therefore, we can write following:

$$\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})_j} \leq \min_{i \in \mathcal{K}_j} \mathbf{q}^\star_{a_i b_i}, \quad j = 1, \dots, |\mathcal{H}(\mathcal{T})|. \tag{10}$$

Using those upper bounds and nonnegativity of entries of matrix $\mathbf{M}$, we can write following set of inequalities:

$$\mathbf{M} \underbrace{\begin{bmatrix} \min_{i \in \mathcal{K}_1} \mathbf{q}^\star_{a_i b_i} \\ \vdots \\ \min_{i \in \mathcal{K}_{j-1}} \mathbf{q}^\star_{a_i b_i} \\ \mathbf{p}^\star_{\mathcal{H}(\mathcal{T})_j} \\ \min_{i \in \mathcal{K}_{j+1}} \mathbf{q}^\star_{a_i b_i} \\ \vdots \\ \min_{i \in \mathcal{K}_l} \mathbf{q}^\star_{a_i b_i} \end{bmatrix}}_{\mathbf{Q}^j} \geq \mathbf{q}^\star, \quad j = 1, \dots, |\mathcal{H}(\mathcal{T})| \tag{11}$$

which enables us to lower bound each entry $\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})_j}$ for $j = 1, \dots, |\mathcal{H}(\mathcal{T})|$. Here, $\mathbf{Q}^j$ represents the vector constructed with minimum $\mathbf{q}^{\star}_{a_i b_i}$'s over different sets and $\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})_j}$. We also define $\mathbf{Q}^j_0$ as the vector that jth entry of $\mathbf{Q}^j$ is replaced with 0. Note that each inequality in (11) can be rewritten as follows

$$\mathbf{M}^T_{k,:}\mathbf{Q}^j \geq \mathbf{q}^{\star}_{a_k b_k}, \quad k = 1, \dots, |\mathcal{T}|.$$

We can also write an alternative expression by using standard basis vectors, i.e., $\mathbf{e}_j$'s,:

$$\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})_j}\mathbf{M}^T_{k,:}\mathbf{e}_j \geq \mathbf{q}^{\star}_{a_k b_k} - \mathbf{M}^T_{k,:}\mathbf{Q}^j_0, \quad k = 1, \dots, |\mathcal{T}|,$$

which provides us following bound

$$\mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})_j} \geq \max\{\max_{i \in \mathcal{K}_j} \mathbf{q}^{\star}_{a_i b_i} - \mathbf{M}^T_{i,:}\mathbf{Q}^j_0, 0\}, \quad j = 1, \dots, |\mathcal{H}(\mathcal{T})|. \tag{12}$$

Combining (10) and (12), we obtain following expression

$$\max_{i \in \mathcal{K}_j} \mathbf{q}^{\star}_{a_i b_i} - \mathbf{M}^T_{i,:}\mathbf{Q}^j_0 \leq \mathbf{p}^{\star}_{\mathcal{H}(\mathcal{T})_j} \leq \min_{i \in \mathcal{K}_j} \mathbf{q}^{\star}_{a_i b_i}. \tag{13}$$

Below we expand on estimation errors to replace $\mathbf{q}^{\star}_{a_i b_i}$'s with corresponding estimates. For any $\mathbf{q}^{\star}_{a_i b_i}$, we can say that

$$|\hat{\mathbf{q}}_{a_i b_i} - \mathbf{q}^{\star}_{a_i b_i}| \leq \sqrt{\frac{\log(2/\delta')}{2n_p}} \tag{14}$$

holds with probability at least $1 - \delta'$ by Hoeffding's Inequality. Therefore, we want to bound the probability that

$$|\hat{\mathbf{q}}_{a_i b_i} - \mathbf{q}^{\star}_{a_i b_i}| \geq \sqrt{\frac{\log(2/\delta')}{2n_p}}$$

holds at least for one i, where n_p is the number of people answering each pairwise query. Therefore, we want to bound

$$\Pr\left(\bigcup_i \left\{ |\hat{\mathbf{q}}_{a_i b_i} - \mathbf{q}^{\star}_{a_i b_i}| \geq \sqrt{\frac{\log(2/\delta')}{2n_i}} \right\}\right) \leq \sum_i \Pr\left(\left\{ |\hat{\mathbf{q}}_{a_i b_i} - \mathbf{q}^{\star}_{a_i b_i}| \geq \sqrt{\frac{\log(2/\delta')}{2n_i}} \right\}\right) \tag{15}$$

$$\leq 2|\mathcal{T}|\delta', \tag{16}$$

where (15) is from union bound and (16) is due to (14). Picking $\delta = 2|\mathcal{T}|\delta'$, we conclude that

$$|\hat{\mathbf{q}}_{a_i b_i} - \mathbf{q}^{\star}_{a_i b_i}| \leq \sqrt{\frac{\log(4|\mathcal{T}|/\delta)}{2n_p}}, \quad \forall i,$$

holds with probability at least $1 - \delta$. Inserting it to the result in 13, we complete the proof of Proposition 4.

B.6. Graph Regularization

In this section, we discuss about the graph structure and the graph regularization that we proposed using in Section 4. We note that proposed graph structure can be constructed using the matrix $\mathbf{M}$. Recall that the rows of $\mathbf{M}$ correspond to hyperplanes and the columns correspond to the regions (polytopes) in $\mathcal{H}(\mathcal{T})$ providing a binary encoding for them by construction. That is, each entry of a given column of $\mathbf{M}$ determines which side of a hyperplane the corresponding region is located on. Therefore, there exists an edge between nodes corresponding to the regions that has only two different entries in their hyperplane coordinates, i.e., only if one pairwise comparison yields opposite results. Accordingly, neighboring regions have common $(d-1)$-dimensional faces in between.

We provide a standard graph regularizer without using volume weighting here to give a better intuition about graph regularizers and why we used volume weighting in Section 4. We start by defining following weight matrix $\mathbf{W}^{\text{unif}}$:

$$\mathbf{W}^{\text{unif}}_{i,j} = \|\mathbf{M}_{:,i} - \mathbf{M}_{:,j}\|_1^{-1}, \tag{17}$$

which is the inverse of the Hamming distance between nodes i and j. Accordingly, we can write following graph Laplacian regularizer:

$$\begin{aligned} R &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}|\mathbf{p}_i - \mathbf{p}_j|^2\mathbf{W}^{\text{unif}}_{i,j} \\ &= \sum_{i=1}^{n}\mathbf{p}_i\mathbf{p}_i\mathbf{D}^{\text{unif}}_{i,i} - \sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{p}_i\mathbf{p}_j\mathbf{W}^{\text{unif}}_{i,j} = \mathbf{p}^T\mathbf{D}^{\text{unif}}\mathbf{p} - \mathbf{p}^T\mathbf{W}^{\text{unif}}\mathbf{p} = \mathbf{p}^T\mathbf{L}^{\text{unif}}\mathbf{p}, \end{aligned}$$

where $\mathbf{D}^{\text{unif}}_{i,i} = \sum_{j=1}^{n}\mathbf{W}^{\text{unif}}_{i,j}$, $\mathbf{D}^{\text{unif}}_{i,j} = 0$ when $i \neq j$ and $\mathbf{L}^{\text{unif}} = \mathbf{D}^{\text{unif}} - \mathbf{W}^{\text{unif}}$. Now, suppose that the spectral decomposition of $\mathbf{L}^{\text{unif}}$ can be written as $\mathbf{L}^{\text{unif}} = \sum_{i=1}^{l}\mu_i\mathbf{v}_i\mathbf{v}_i^T$, where $\mathbf{v}_i$'s are eigenvectors and μ_i's are the corresponding eigenvalues. We now further elaborate on spectral properties of Laplacian matrices and use following Lemma.

Lemma B.4. *Graph Laplacian matrices are positive semi-definite by the Gershgorin circle theorem. Furthermore, the eigenvectors of the Laplacian matrix $\mathbf{L}^{\text{unif}}$ corresponding to zero eigenvalues are spanned by $\mathbf{1}$, which is referred to as constant vectors in (Poignard et al., 2018).*

Then, we can rewrite Laplacian regularizer in (4) as

$$\mathbf{p}^T\mathbf{A}^T\mathbf{L}^{\text{unif}}\mathbf{A}\mathbf{p} = \mathbf{p}^T\sum_{i=1}^{l}\mu_i\mathbf{A}^T\mathbf{v}_i\mathbf{v}_i^T\mathbf{A}\mathbf{p} = \sum_{i=1}^{l}\mu_i(\mathbf{p}^T(\mathbf{A}^T\mathbf{v}_i))^2,$$

where $\mathbf{A}$ is a diagonal matrix with the entries $\mathbf{A}_{i,i} = \frac{1}{\alpha_i}$ and $\sum_i\mathbf{A}_{i,i} = 1$. Laplacian regularizer $\mathbf{L} = \mathbf{A}^T\mathbf{L}^{\text{unif}}\mathbf{A}$ penalizes $\mathbf{p}$ so that potential $\mathbf{p}$ values correlated to vectors $\mathbf{A}^T\mathbf{v}_i$'s are diminished. We can rephrase it as follows: regularizer penalizes $\mathbf{p}$ so that potential $\mathbf{A}^{-1}\mathbf{p}$ values correlated to eigenvectors $\mathbf{v}_i$'s are diminished. Therefore, $\mathbf{v}_i$'s corresponding to larger eigenvalues cause more penalty. From Lemma B.4, it follows that Laplacian matrix $\mathbf{L}$ corresponding to zero eigenvalues are spanned by $\mathbf{A}^{-1}\mathbf{1}$. In (Poignard et al., 2018), authors also point out that the multiplicity of the eigenvalue is equal to the number of connected components in the graph, which is clearly 1 in our graph structure induced by $\mathbf{M}$, since the regions in $\mathcal{H}(\mathcal{T})$ are connected. We note that the eigenvectors of $\mathbf{L}^{\text{unif}}$ are mutually orthogonal by spectral theory. We observe that orthogonal eigenvectors of nonzero eigenvalues would force the candidate of the solution $\mathbf{p}$ to be similar to uniform distribution by punishing possible directions other than $\mathbf{1}$. However, we note that regions in $\mathcal{H}(\mathcal{S}_m)$ are not similar to an equally spaced grid. Therefore, we use a weighted version of the regularizer in (4) with respect to the volumes of the regions in $\mathcal{H}(\mathcal{T})$ instead of $\mathbf{L}^{\text{unif}}$ and punish possible directions other than $\mathbf{A}^{-1}\mathbf{1}$.

In short, the regularizer in equation (4) encourages the changes in nearby regions to be smooth, which is similar to the local invariance property considered in (Belkin & Niyogi, 2001; Cai et al., 2011; Hadsell et al., 2006). Weighted Laplacian regularizer $\mathbf{L}$ imposes a penalty on $\mathbf{p}$ in such a way that potential values correlated with eigenvectors of $\mathbf{L}$ are diminished. Therefore, eigenvectors corresponding to larger eigenvalues cause more penalty. Since the eigenvectors of $\mathbf{L}$ are mutually orthogonal by spectral theorem. So, we conclude that orthogonal eigenvectors of nonzero eigenvalues force the potential solution to be close to the distribution $\overline{\alpha}$ by diminishing possible directions other than α, where $\overline{\alpha}$ is the normalized α.

B.7. Proof of Theorem 4.2

We first show that the solution to the convex optimization problem in (6) is unique. Let $f(\mathbf{p})$ be the objective function $\frac{1}{2}||\mathbf{M}\mathbf{p} - \hat{\mathbf{q}}||_2^2 + \frac{\lambda}{2}\mathbf{p}^T\mathbf{L}\mathbf{p}$. If we can guarantee that

$$\frac{\partial^2 f}{\partial \mathbf{p}^2} = 2\mathbf{M}^T\mathbf{M} + 2\lambda\mathbf{L} \succ 0, \tag{18}$$

we deduce that solution to the convex optimization problem in (6) is unique. Therefore, we first focus on matrix $\mathbf{L}$. From Lemma B.4, null space of $\mathbf{L}^{\text{unif}}$ is spanned by $\mathbf{1}$. Since $\mathbf{A}$ is a full rank matrix, null space of $\mathbf{L} = \mathbf{A}^T\mathbf{L}^{\text{unif}}\mathbf{A}$ is spanned by $\mathbf{A}^{-1}\mathbf{1}$. All entries of $\mathbf{A}^{-1}\mathbf{1}$ are nonnegative since $\mathbf{A}^{-1}$ is a diagonal matrix with nonnegative entries. Now, we have following

$$\begin{aligned} \mathbf{M}^T\mathbf{M} &\succeq 0, \\ \mathbf{L} &\succeq 0, \\ \mathbf{M}^T\mathbf{M} + \lambda\mathbf{L} &\succeq 0. \end{aligned}$$

If $ker(\mathbf{M}^T\mathbf{M}) \neq ker(\mathbf{L})$, we can guarantee that $\mathbf{M}^T\mathbf{M} + \lambda\mathbf{L} \succ 0$. $\mathbf{M}^T\mathbf{M}$ is already positive semidefinite and $\mathbf{A}^{-1}\mathbf{1}$ cannot be an eigenvector for $\mathbf{M}^T\mathbf{M}$, since all nonzero entries of $\mathbf{M}^T\mathbf{M}$ have same sign. Therefore, $\mathbf{M}^T\mathbf{M} + \lambda\mathbf{L}$ is always positive definite.

Now, we recall that $\mathbf{R}^T\mathbf{R} = \mathbf{M}^T\mathbf{M} + \lambda\mathbf{L}$ and note that multiplication of each element in the unit simplex with matrix $\mathbf{R}$ defines following closed convex set,

$$C_{\mathbf{R}} := \mathrm{conv}(\mathbf{R}\mathbf{e}_1, \mathbf{R}\mathbf{e}_2 \ldots, \mathbf{R}\mathbf{e}_{|\mathcal{H}(\mathcal{T})|}).$$

Then, the unique solution $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ to the optimization setting in (6) can be expressed as

$$\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} = \mathbf{R}^{-1}\mathrm{Proj}_{C_{\mathbf{R}}}(\mathbf{b}), \tag{19}$$

where $\mathbf{b} = \mathbf{R}^{-T}\mathbf{M}^T\mathbf{M}\mathbf{p}^*$. Therefore,

$$\mathbf{R}\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} = \mathrm{Proj}_{C_{\mathbf{R}}}(\mathbf{R}^{-T}\mathbf{M}^T\hat{\mathbf{q}}).$$

We start with bounding ℓ_2 norm error and write

$$\begin{aligned}
\|\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} - \mathbf{p}^*_{\mathcal{H}(\mathcal{T})}\|_2 &\leq \|\mathbf{R}^{-1}\|_2\|\mathbf{R}\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})} - \mathbf{R}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})}\|_2 \\
&= \|\mathbf{R}^{-1}\|_2\|\mathbf{R}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathrm{Proj}_{C_{\mathbf{R}}}(\mathbf{R}^{-T}\mathbf{M}^T\hat{\mathbf{q}})\|_2 \\
&\leq \|\mathbf{R}^{-1}\|_2\|\mathbf{R}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{R}^{-T}\mathbf{M}^T\hat{\mathbf{q}}\|_2 \qquad (20) \\
&\leq \|\mathbf{R}^{-1}\|_2(\|\mathbf{R}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{R}^{-T}\mathbf{M}^T\mathbf{M}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} \qquad (21) \\
&+ \mathbf{R}^{-T}\mathbf{M}^T\mathbf{M}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{R}^{-T}\mathbf{M}^T\hat{\mathbf{q}}\|_2) \\
&\leq \|\mathbf{R}^{-1}\|_2(\|\mathbf{R}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{R}^{-T}\mathbf{M}^T\mathbf{M}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})}\|_2 \qquad (22) \\
&+ \|\mathbf{R}^{-T}\mathbf{M}^T(\mathbf{M}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{q}})\|_2) \\
&\leq \|\mathbf{R}^{-1}\|_2(\|\lambda\mathbf{R}^{-T}\mathbf{L}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})}\|_2 + \|\mathbf{R}^{-T}\mathbf{M}^T(\mathbf{M}\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \hat{\mathbf{q}})\|_2) \qquad (23) \\
&\leq \|\mathbf{R}^{-1}\|_2^2(\lambda\|\mathbf{L}(\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \overline{\alpha})\|_2 + \|\mathbf{M}^T\|_2\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_2) \qquad (24) \\
&\leq \|\mathbf{R}^{-1}\|_2^2(\lambda\|\mathbf{L}\|_2\left\|\mathbf{p}^\star - \overline{\alpha}\right\|_2 + \|\mathbf{M}^T\|_2\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_2), \qquad (25)
\end{aligned}$$

where (20) is due to contracting property of projection onto closed convex sets, (23) is because $\mathbf{M}^T\mathbf{M} = \mathbf{R}^T\mathbf{R} - \lambda\mathbf{L}$, and (24) follows from $\mathbf{M}\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})} = \mathbf{q}^\star$ and $\mathbf{L}\overline{\alpha} = 0$. Then, by using $\ell_1 - \ell_2$ norm inequality, we can simply write following inequalities

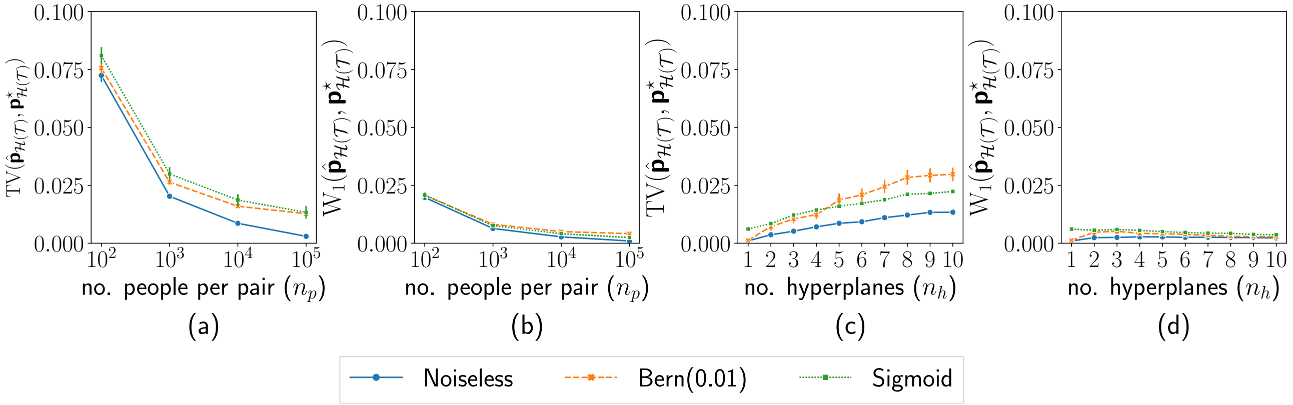
$$\begin{aligned}
\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \mathbf{p}_{sol}) = \frac{1}{2}||\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{p}_{sol}||_1 &\leq \frac{\sqrt{|\mathcal{H}(\mathcal{T})|}}{2}||\mathbf{p}^*_{\mathcal{H}(\mathcal{T})} - \mathbf{p}_{sol}||_2 \\
&\leq \frac{\sqrt{|\mathcal{H}(\mathcal{T})|}}{2}\|\mathbf{R}^{-1}\|_2(\lambda\|\mathbf{L}\|_2\left\|\mathbf{p}^\star - \overline{\alpha}\right\|_2 \\
&+ \|\mathbf{M}^T\|_2\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_2) \\
&\leq \frac{\lambda}{2}\sqrt{|\mathcal{H}(\mathcal{T})|}\|\mathbf{R}^{-1}\|_2\|\mathbf{L}\|_2\left\|\mathbf{p}^\star - \overline{\alpha}\right\|_2 \\
&+ \frac{\sqrt{|\mathcal{H}(\mathcal{T})|}}{2}\|\mathbf{R}^{-1}\|_2\|\mathbf{M}^T\|_2\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_2 \\
&\leq \frac{\lambda}{2}\sqrt{|\mathcal{H}(\mathcal{T})|}\|\mathbf{R}^{-1}\|_2\|\mathbf{L}\|_2\left\|\mathbf{p}^\star - \overline{\alpha}\right\|_2 \\
&+ \sqrt{\frac{|\mathcal{H}(\mathcal{T})||\mathcal{T}|}{2}}\|\mathbf{R}^{-1}\|_2\|\mathbf{M}^T\|_2\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_1.
\end{aligned}$$

Lastly, we apply Lemma B.1 for $\|\mathbf{q}^\star - \hat{\mathbf{q}}\|_1$ and complete the proof.

C. Simulations and Experimental Results

We present simulation results with a mixture of three Gaussians here and defer the rest to Appendix D.

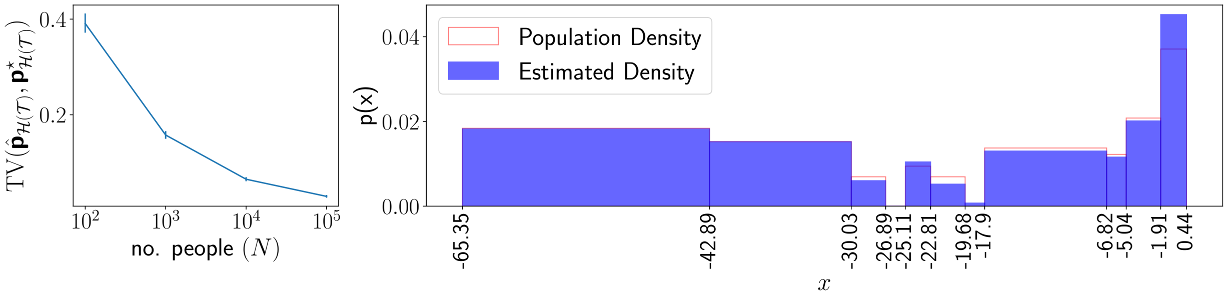
1D Simulations: Figure (3) shows the relationship between the number of hyperplanes, n_h, and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$ by varying $n_h \in \{1, \ldots, 10\}$, as well as between the number of people asked per query, n_p, and the error, by varying $n_p \in \{10^2, 10^3, 10^4, 10^5\}$. As shown in our analysis, the recovery gets better as the number of users increases. Note that as the number of pairs, (equivalently, the number of hyperplanes) increase, size of the length of $\mathbf{p}$ increases, and hence we expect an increase in the TV.



(a) (b) (c) (d)

Figure 3. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_1(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for mixture of 3 Gaussians in 1D.

Colors dataset: Colors dataset (Palmer & Schloss, 2010; Palmer et al., 2013) consists of answers to pairwise queries from 48 different users and 37 colors, where each person was asked all $\binom{37}{2}$ pairwise comparisons. In this dataset, each color is considered as a 3-dimensional vector in CIELAB color space (lightness, red vs. green, blue vs. yellow). For our experiment, we use the 1D user embedding of the colors dataset learned from (Canal et al.). We then project the CIELAB color space onto this 1D user embedding space. We consider a subset of $m = 5$ colors sampled from this space and use all 10 pairs for comparison. Then, we uniformly sample $\{10^2, 10^3, 10^4, 10^5\}$ users from all 48 user preference points learned from (Canal et al.) with replacement for each pair to estimate $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ and form $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ as shown in Figure (4)(a). Figure (4)(b) shows the true distribution of people (computed using making multiple queries to each user on a separate user set) and the distribution recovered using our method.



(a) (b)

Figure 4. (a) $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ by varying number of people, (b) $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ for Colors dataset.

Simulations for $d \geq 2$: Figure (5) shows the relationship between the number of hyperplanes, n_h, and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$, as well as between the number of people asked per query, n_p, and the error for $d = 2$. We use $\lambda = 1$ here and defer results with varying regularization parameter λ to Appendix D. Additionally, we provide simulation results using l_1- and l_2-norm regularizations and the maximum likelihood estimate (KL) as baselines (Figure (6)(c)). We defer more detailed results to Appendix D. Figures (6)(a) and (b) show the relationship between the feature dimension d and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$.

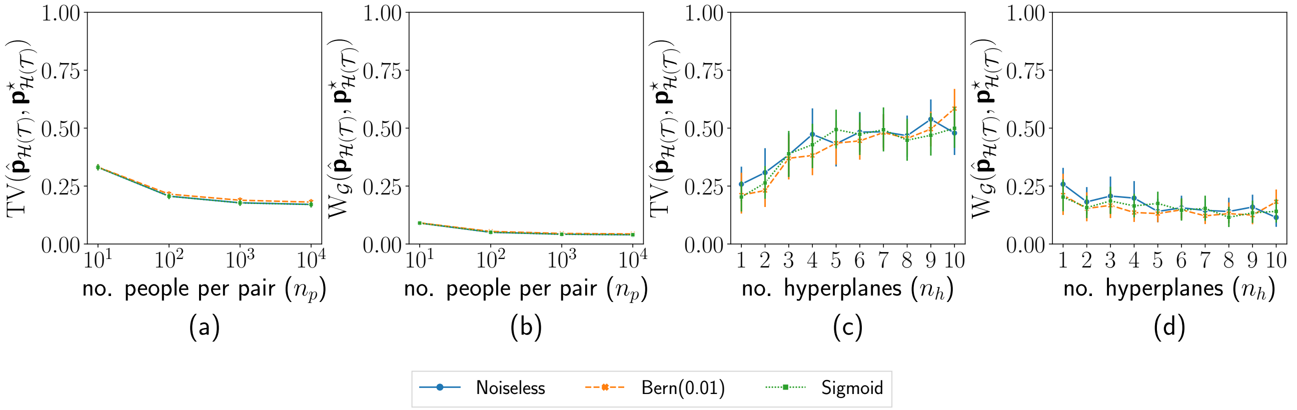


Figure 5. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for mixture of 3 Gaussians in 2D

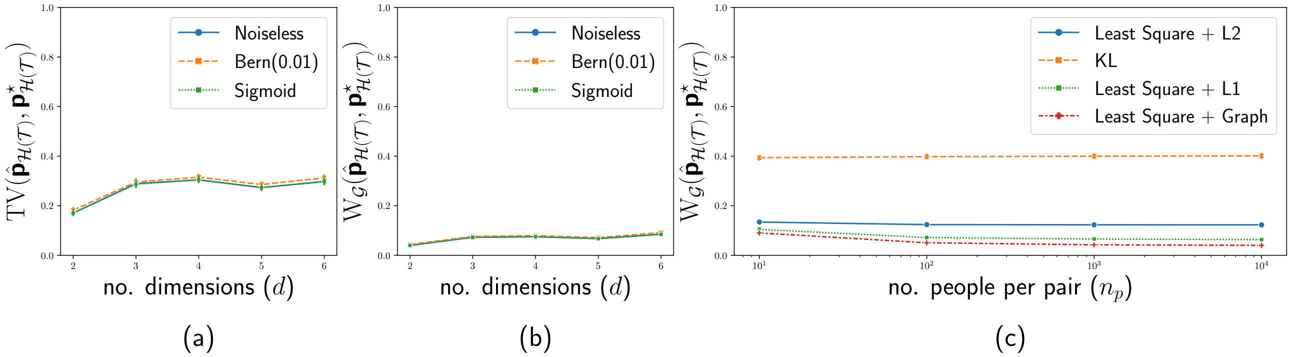


Figure 6. (a) $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$, (b) $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for mixture of 3 Gaussians with varying d. (c) $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for different objective functions with varying n_p.

Bounds on the Mass: We generate the true underlying preferences from a mixture of 3 Gaussians in 2D. We query for 5 pairs of items and $10,000$ users per pair. Figure (7) shows the upper and lower bounds on the mass in each of the regions in the intersection of the 5 hyperplanes using equations in Proposition 4. We also show the true mass induced by the underlying distribution in these regions which highlights the efficacy of our bounds.

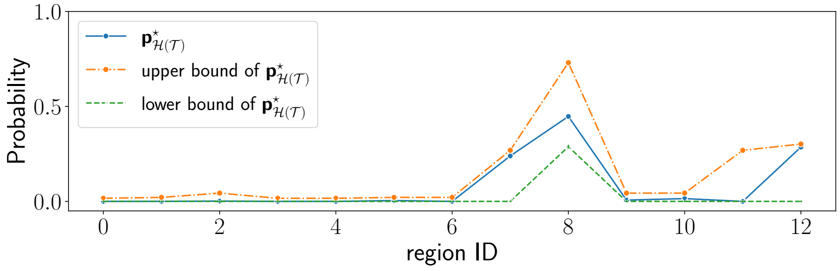


Figure 7. Lower and upper bounds with the true underlying distribution, a mixture of 3 Gaussians.

Zappos: UT Zappos50K (Yu & Grauman, 2014; 2017) is a large dataset with $50,025$ catalog images of shoes in different categories, such as shoes, sandals, slippers, and boots. We manually pick five shoes from this dataset and collect responses from 6000 Amazon Mechanical Turk (AMT) workers for each possible pairwise query. With a subset of workers' answers to each possible pairs, we estimate $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and use the remaining workers to answer pairwise comparison queries using only one response per worker to estimate $\hat{\mathbf{p}}$. We defer details of the setting to Appendix D. Figures 8(a) and (b) show the results of our experiments on this dataset.

Movies: We create a new dataset comprising $4,266$ movies from different countries, produced between 2013 and 2022, inclusive. Each movie is associated with its plot and info scrapped from Wikipedia (Wikipedia, 2023). We utilize OpenAI's *text-embedding-ada-002* model (OpenAI, 2023) to generate an embedding for each movie. Then, we train a regression neural network, where the target is each movie's average IMDB rating (IMDB, 2023). We use the output of the penultimate layer from the network as an intermediate embedding, which has 100 dimensions. Lastly, we reduce it to 2D using PaCMAP (Wang et al., 2021a). In the subsequent experiment, we consider the 2D embedding as the coordinates for the movies. We scrape the ratings of reviewers, both critics and audience, from Rotten Tomatoes (Tomatoes, 2023), and use these ratings to create answers to pairwise comparison queries. We run our experiment on a set of 13 DC Comics superhero movies. We consider 9 pairwise comparisons questions and assign each pair to 50 reviewers. Our results are presented in Figures 8(c) and (d).

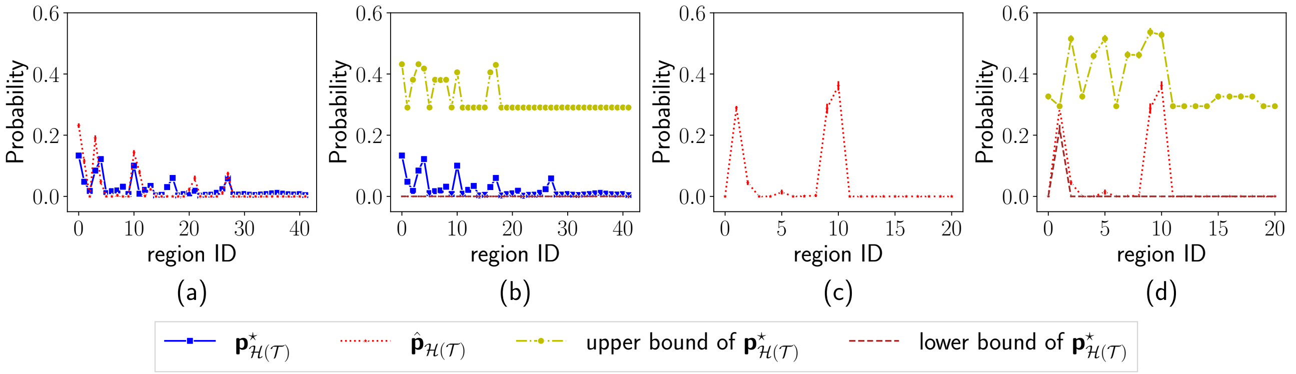


Figure 8. (a) $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ and $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ for Zappos dataset. (b) Estimated upper and lower bounds for $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$ in Zappos dataset. (c) $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ in movies dataset. (d) Estimated upper and lower bounds for $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ in movies dataset.

D. Additional Simulations and Experimental Results with Details

D.1. Simulations for $d = 1$

We provide simulation results for following group of user distributions: uniform, Gaussian, a mixture of 2 Gaussians, and a mixture of 3 Gaussians. We also present simulations results for varying amount of noises in both noise models.

Figure 9-12 show the relationship between the number of hyperplanes, n_h, and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$ by varying $n_h \in \{1, \dots, 10\}$, as well as between the number of people asked per query, n_p, and the error, by varying $n_p \in \{10^2, 10^3, 10^4, 10^5\}$, under the 4 user distributions and different noises. As it can be seen in our analysis, the recovery gets better as the number of users increases. Note that as the number of pairs, (equivalently, the number of hyperplanes) increase, size of $\mathbf{p}$ increases, and hence we expect an increase in the TV.

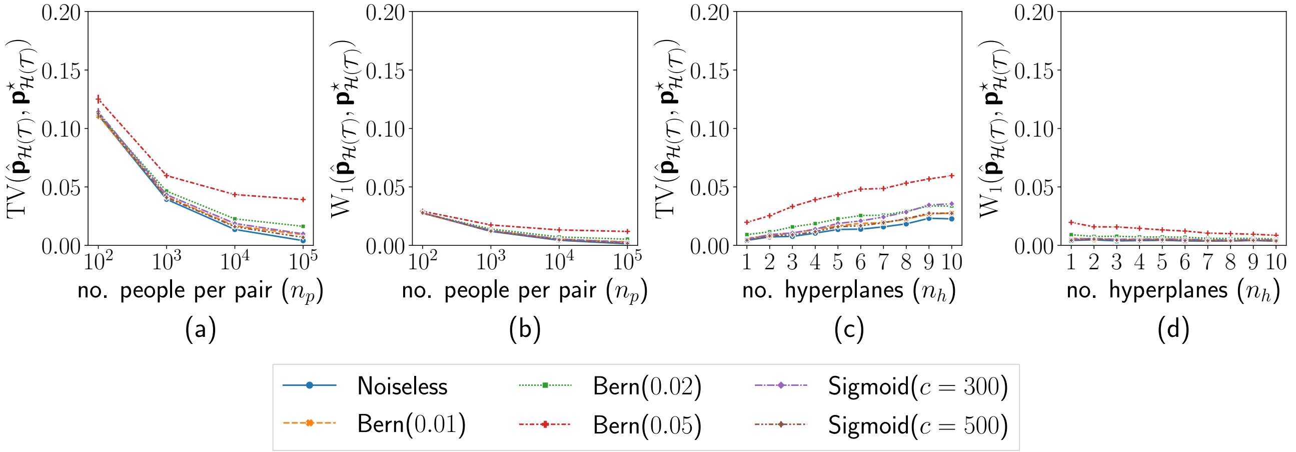


Figure 9. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_1(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for uniform user distribution in 1D.

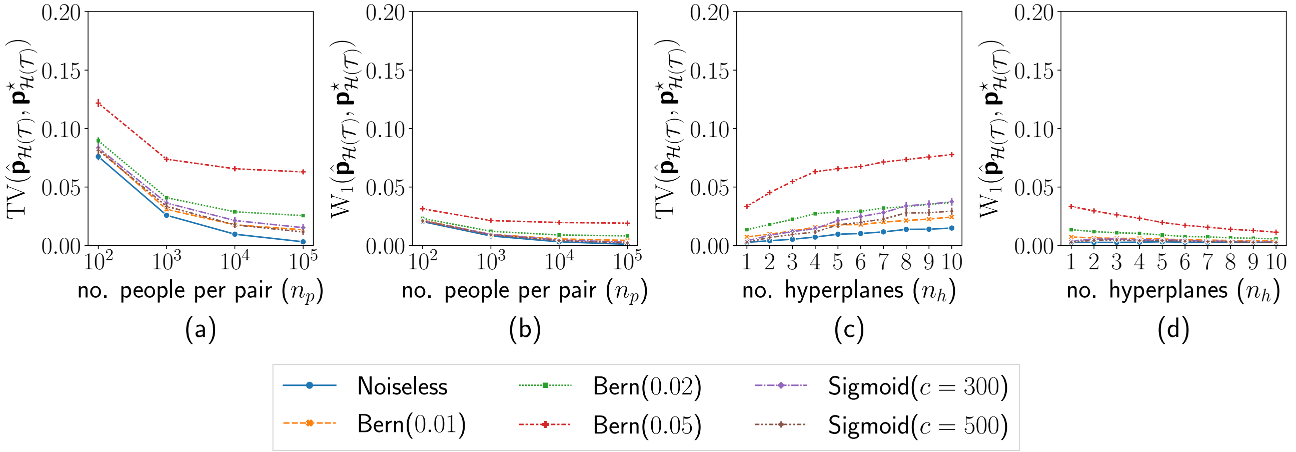


Figure 10. TV($\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$) and W$_1$($\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$) for Gaussian user distribution in 1D.

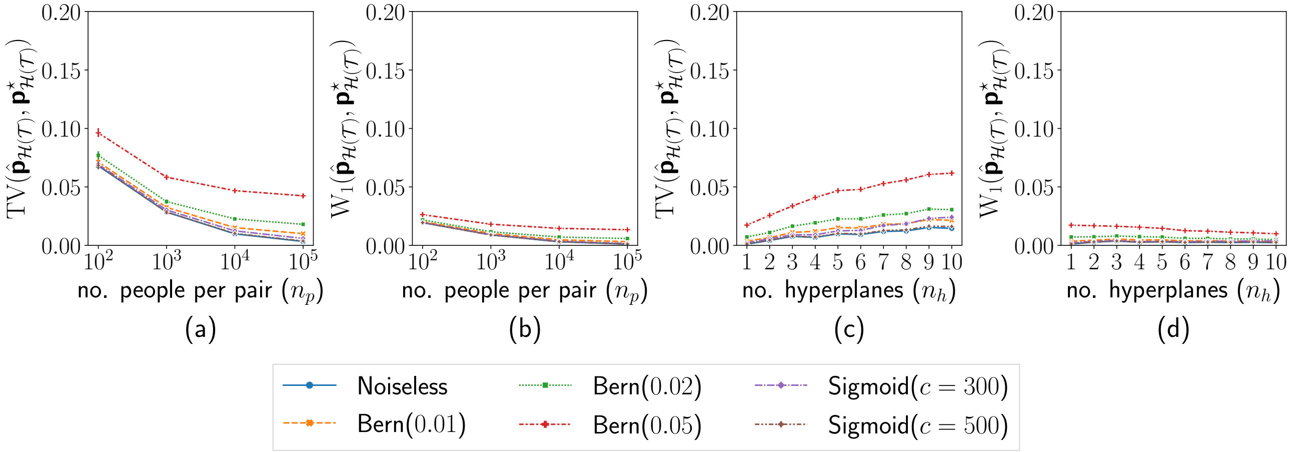


Figure 11. TV($\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$) and W$_1$($\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$) for a mixture of 2 Gaussians user distribution in 1D.

D.2. Construction of M in dimensions $d \geq 2$

Unlike 1D, algorithmic construction of binary matrix $\mathbf{M}$ is not straightforward in dimensions $d \geq 2$. We need to figure out which polytopes, i.e., regions, are on the left side of a given hyperplane. We recall that these polytopes are defined by the halfspaces induced by the bisecting hyperplanes of item pairs in $\mathcal{T}$.

Hence, our problem can be formally described as follows: Given a set of halfspaces $\mathbb{H}_s = \{\boldsymbol{a}_{ij}^\top \boldsymbol{x} + b_{ij} < 0 : h_{ij} = \boldsymbol{a}_{ij}^\top \boldsymbol{x} + b_{ij} = 0, i < j\}$, where h_{ij} is the bisecting hyperplanes of pair $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{T}$, we want to find all polytopes in $\mathcal{H}(\mathcal{T})$ that are in the halfspace h_s, for each $h_s \in \mathbb{H}_s$. To make our life easier, we define a bounding box $[-1, 1]^d$, so that we can only look at the polytopes within this box and avoid unbounded polytopes. For simplicity, we use the vector $[\boldsymbol{a}_{ij} \quad b_{ij}]$ to represent a halfspace.

Let $\mathbb{B}_s$ denote the set of halfspaces that defines the bounding box $[-1, 1]^d$. Let $\mathbb{P}_t$ denote the set of polytopes that we have discovered. Let $\mathbb{H}_s^u$ denote the set of halfspaces we have not explored yet. Our algorithm works as follows:

```
$\mathbb{P}_t \leftarrow \{\mathbb{B}_s\}$
for $h_s \in \mathbb{H}_s \setminus \mathbb{H}_s^o$ do
    for $p_t \in \mathbb{P}_t$ do
        if $h_s$ intersects with $p_t$ then
```

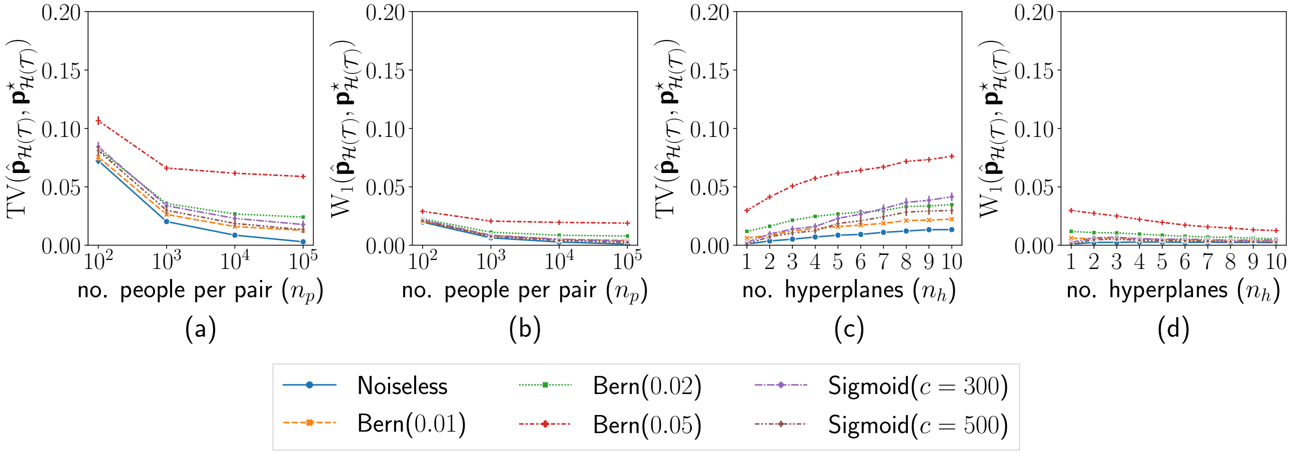


Figure 12. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_1(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 3 Gaussians user distribution in 1D.

```
            $p_t^l \leftarrow p_t \cup \{h_s\}$
            $p_t^r \leftarrow p_t \cup \{-h_s\}$
            $\mathbb{P}_t \leftarrow \mathbb{P}_t \setminus p_t$
            $\mathbb{P}_t \leftarrow \mathbb{P}_t \cup \{p_t^l, p_t^r\}$
        end if
    end for
end for
```

To check if h_s intersects with p_t, we first assume that h_s splits p_t into two polytopes, namely, $p_t^l := p_t \cup \{h_s\}$ and $p_t^r := p_t \cup \{-h_s\}$. If p_t^l or p_t^r degenerate, then the assumption does not hold and therefore h_s does not intersect with p_t. To verify if p_t^l or p_t^r degenerate, we check if they have a Chebyshev center, which can be found by solving the following linear program twice:

$$\begin{aligned} \max_{\boldsymbol{y}, r} \quad & r \\ \text{subject to} \quad & \boldsymbol{a}_i^T \boldsymbol{y} + ||a_i|| r \leq b_i, \quad \forall i \in [|p_t| + 1] \end{aligned}$$

where $[\boldsymbol{a}_i \quad b_i]$ is the i^{th} halfspace in p_t^l(or p_t^r) and $\boldsymbol{y}$ is the Chebyshev center (when solved). If the two linear programs have (bounded) solutions and $\boldsymbol{y}$ is in p_t, we can say that p_t^l and p_t^r have Chebyshev centers and therefore h_s intersects with p_t. Otherwise, we can conclude that h_s does not intersect with p_t. To determine the position of any polytope with respect to hyperlanes (halfspaces), we check whether the Chebyshev center of that polytope is on the left or right of the hyperplane (is in the halfspace).

D.3. Simulations for $d \geq 2$

We first present the results with varying regularization parameter λ. Figure 13 and 14 present the behavior of TV and $W_{\mathcal{G}}$ under the 4 user distributions while we vary λ when $n_p = 10,000$, $m = 5$, $n_h = 10$, and $d = 2$. No noise model is introduced in this set of simulation. 4 different colored lines in Figures refer to the 4 different objective we used. Least Square + Graph means that the objective is least square with graph regularization; Least Square + L1 means that the objective is least square with ℓ_1 regularization; Least Square + L2 means that the objective is least square with ℓ_2 regularization, and KL means that the objective is the KL divergence of $\hat{\mathbf{q}}$ from $\mathbf{Mp}$, $D_{\mathrm{KL}}(\hat{\mathbf{q}}, \mathbf{Mp})$, where the solution is maximum likelihood estimate.

We also present the behavior of TV and $W_{\mathcal{G}}$ when we use a different formulation of the optimization problem, where we use the regularization term in the original optimization objective as the sole objective, and use $||\mathbf{Mp} - \hat{\mathbf{q}}||_2^2 \leq \varepsilon$ ($D_{\mathrm{KL}}(\hat{\mathbf{q}}, \mathbf{Mp}) \leq \varepsilon$) as an additional constraint. We set $\varepsilon = 10^{-5}$ in simulation. Figure 15 and 16 present the results with

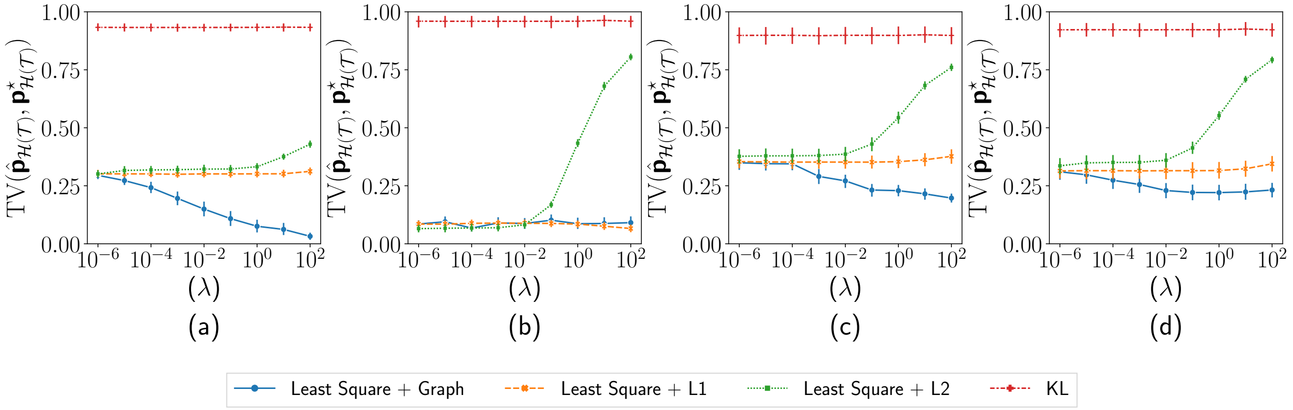


Figure 13. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for (a) uniform (b) Gaussian (c) mixture of 2 Gaussians (d) mixture of 3 Gaussians user distribution while varying the regularization parameter λ.

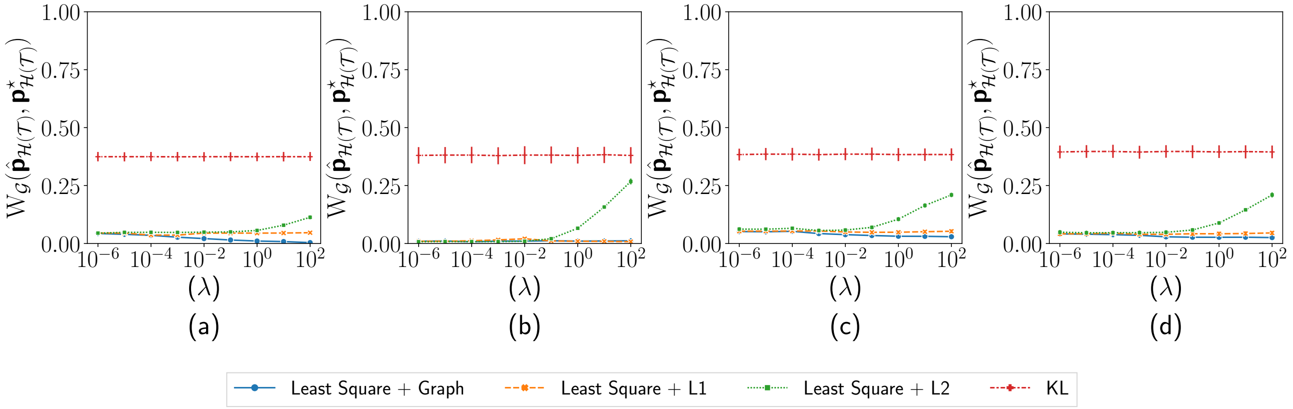


Figure 14. $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for (a) uniform (b) Gaussian (c) mixture of 2 Gaussians (d) mixture of 3 Gaussians user distribution while varying the regularization parameter λ.

the new formulation of the optimization problem under the same setting as above.

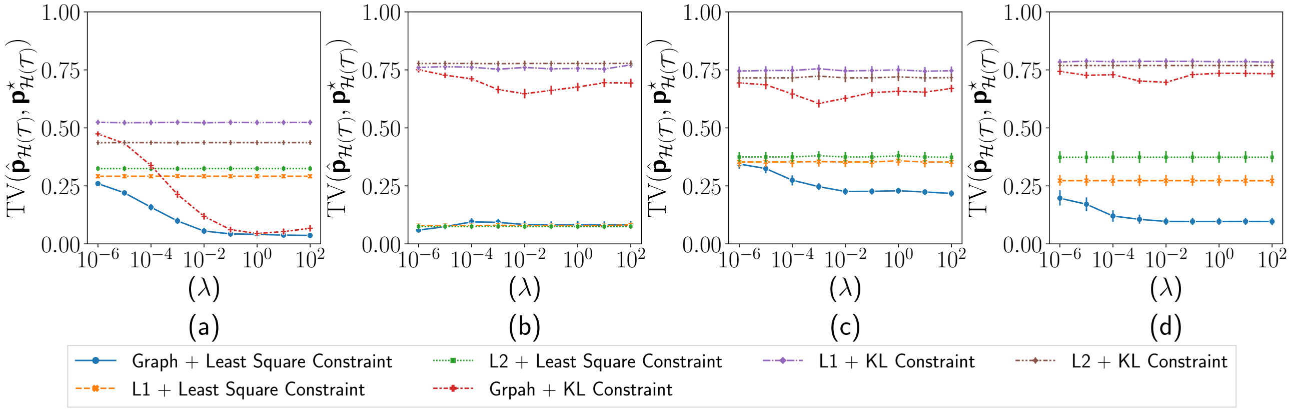


Figure 15. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for (a) uniform (b) Gaussian (c) mixture of 2 Gaussians (d) mixture of 3 Gaussians user distribution while varying the regularization parameter λ.

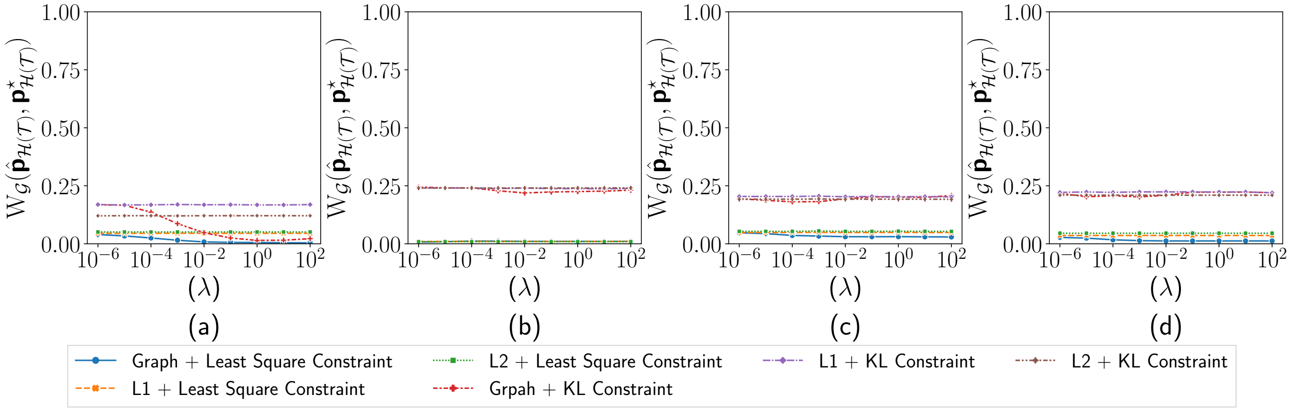


Figure 16. $W_\mathcal{G}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for (a) uniform (b) Gaussian (c) mixture of 2 Gaussians (d) mixture of 3 Gaussians user distribution while varying the regularization parameter λ.

In the subsequent simulation, we fix $\lambda = 1$. We now provide simulation results for following group of users distributions: uniform, Gaussian, a mixture of 2 Gaussians, and a mixture of 3 Gaussians. We also present simulations results for varying amount of noises in both noise models. Figure 17-20 show the relationship between the number of hyperplanes, n_h, and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$, as well as between the number of people asked per query, n_p, and the error for $d = 2$, with the 4 user distributions and different noise models.

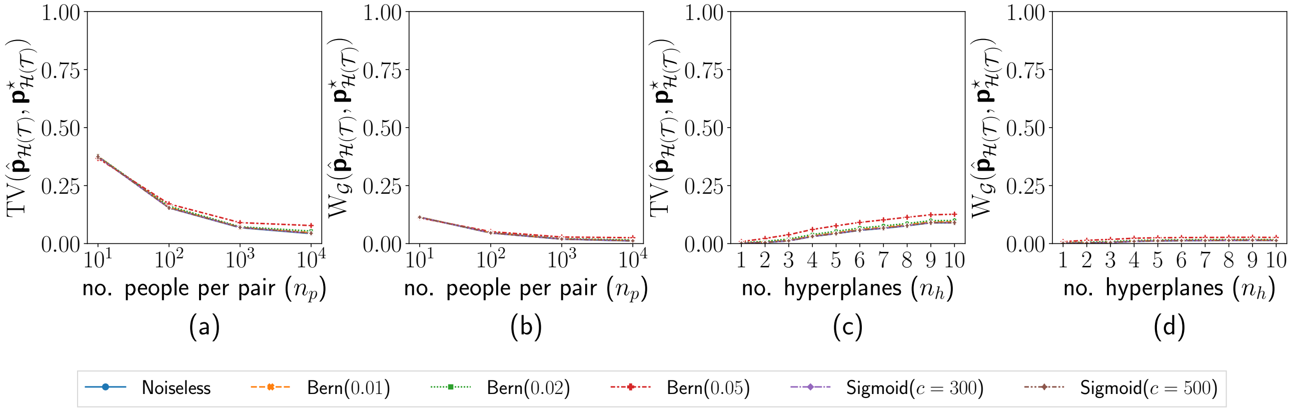


Figure 17. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $W_\mathcal{G}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for uniform user distribution in 2D.

Additionally, Figure 21- 24 show the relationship between the feature dimension d and the error in recovered mass in the partitions $\mathcal{H}(\mathcal{T})$.

Lastly, Figure 25- 28 provide simulation results in terms of $W_\mathcal{G}$ using all optimization methods while varying the number of people per pair, n_p, under all 4 user distributions, $d = 2$, $n_h = 5$, with no noise model introduced.

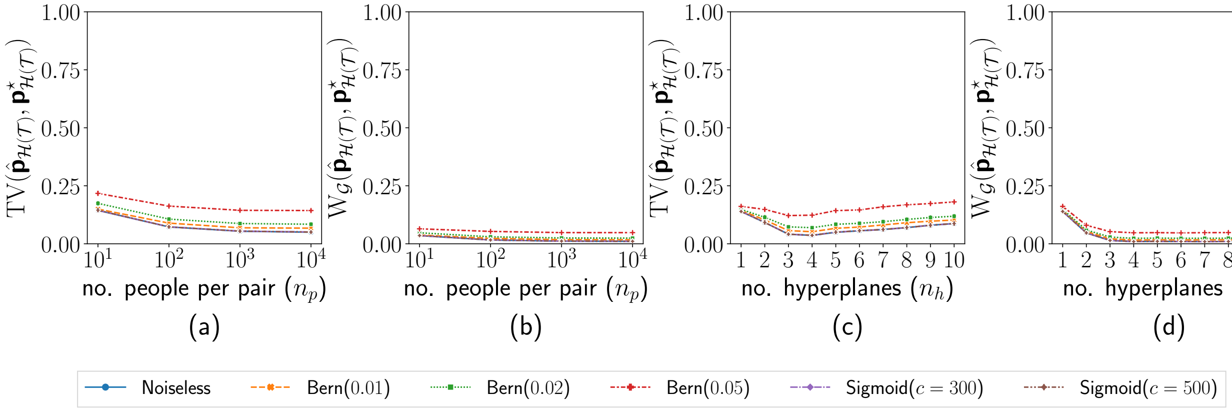


Figure 18. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for Gaussian user distribution in 2D.

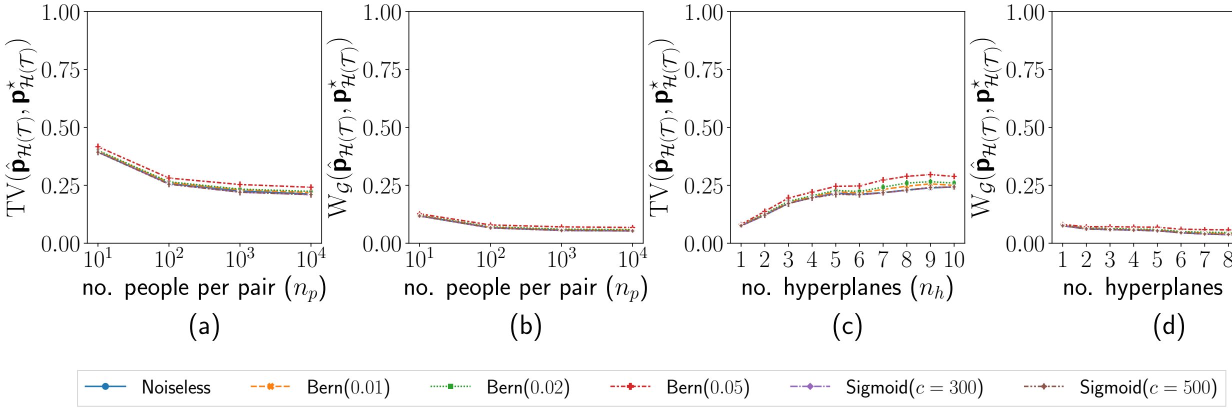


Figure 19. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 2 Gaussian user distribution in 2D.

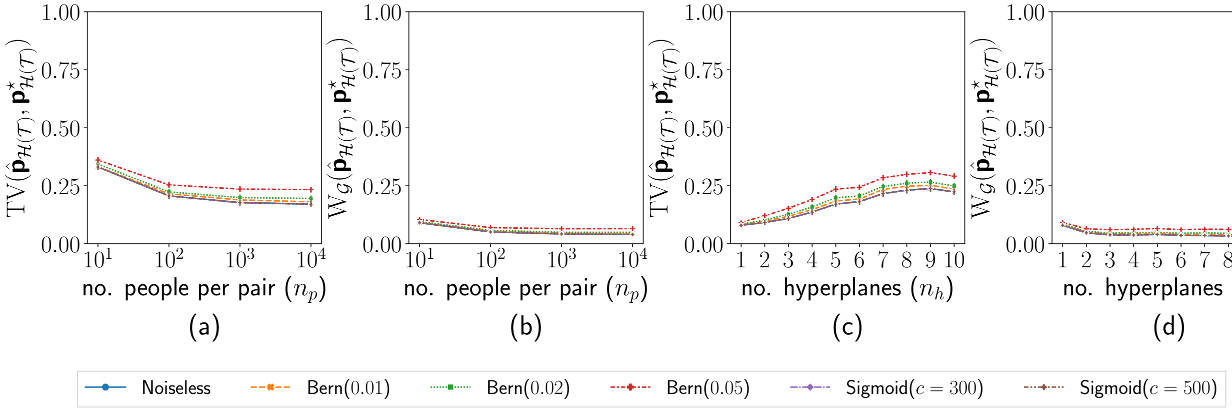


Figure 20. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 3 Gaussian user distribution in 2D.

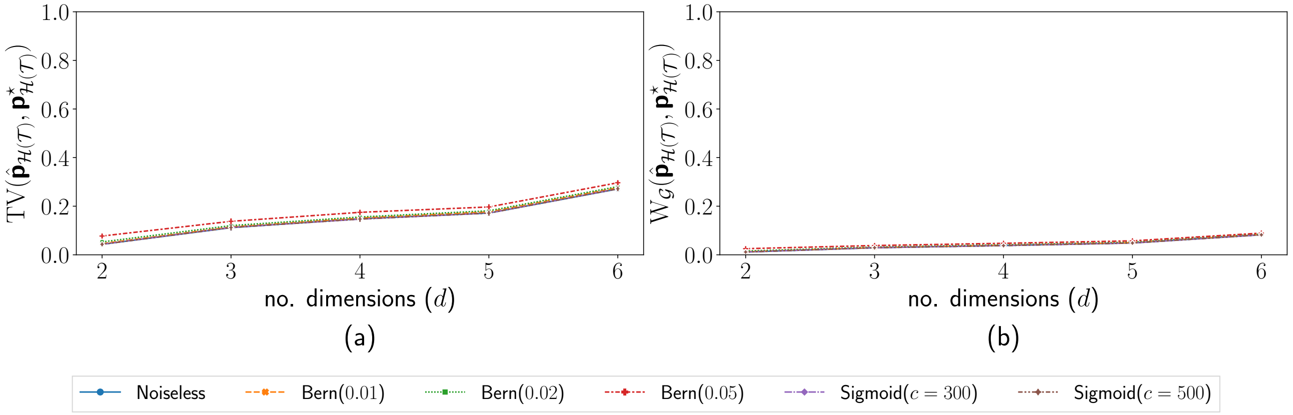


Figure 21. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for uniform user distribution in 2D with varying d.

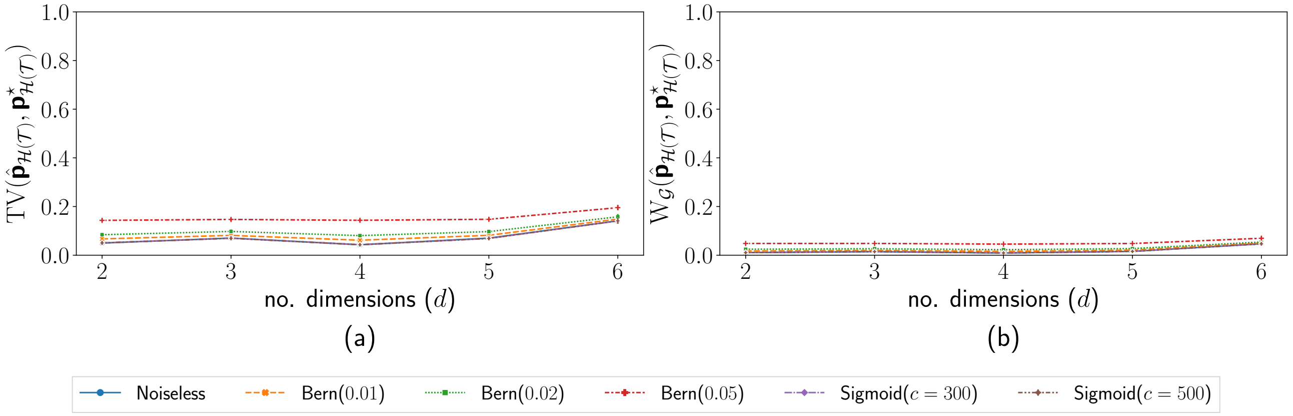


Figure 22. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for Gaussian user distribution in 2D with varying d.

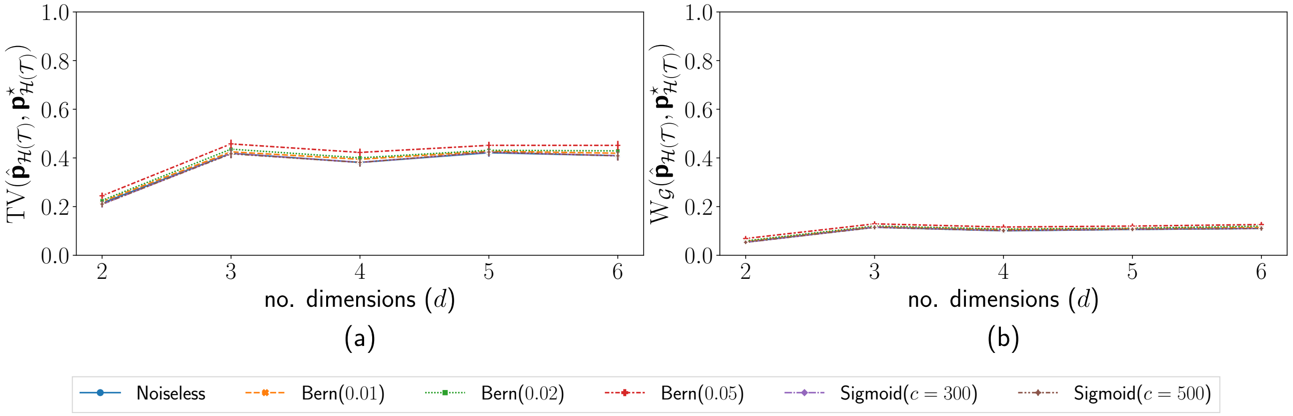


Figure 23. $\mathrm{TV}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and $\mathrm{W}_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 2 Gaussian user distribution in 2D with varying d.

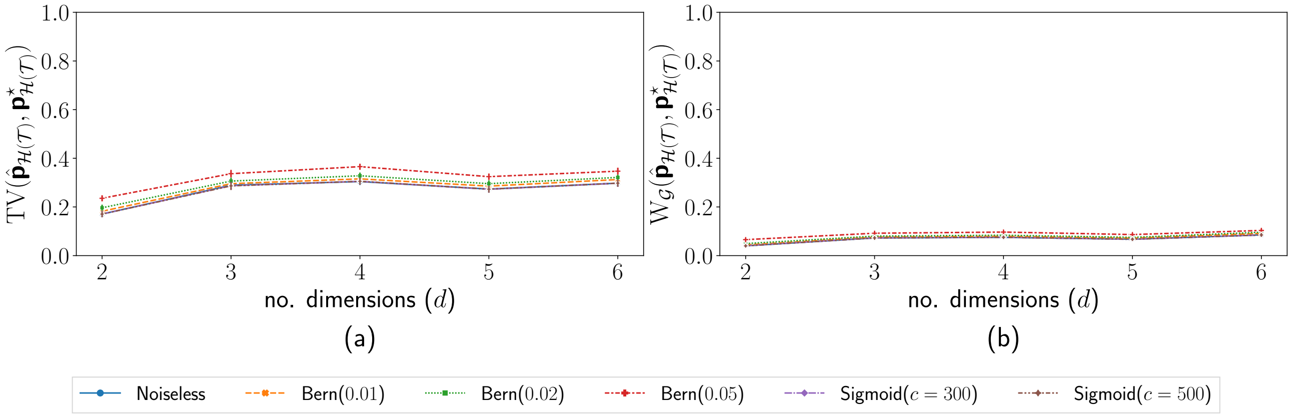


Figure 24. TV$(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ and W$_\mathcal{G}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 3 Gaussian user distribution in 2D with varying d.

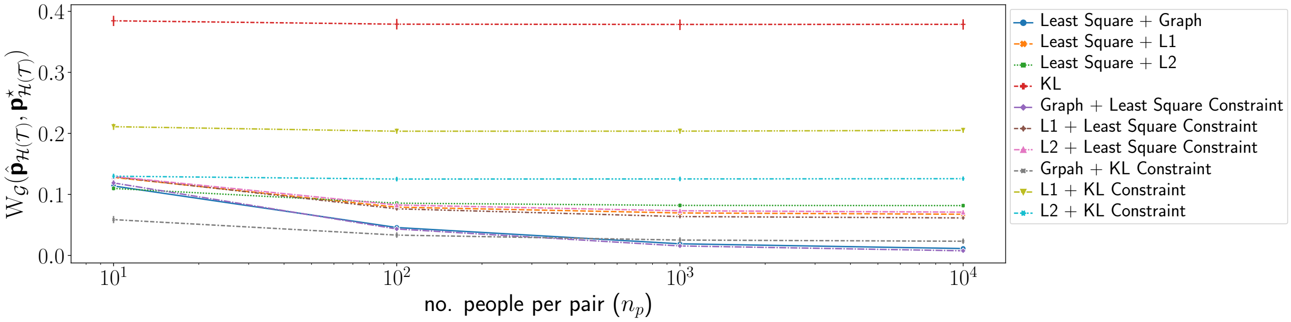


Figure 25. W$_\mathcal{G}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for uniform user distribution in 2D with varying n_p using all optimization methods.

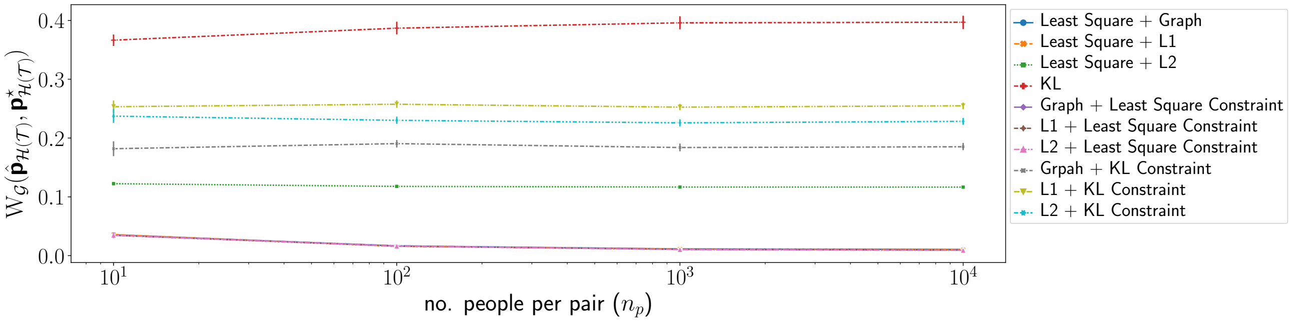


Figure 26. W$_\mathcal{G}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for Gaussian user distribution in 2D with varying n_p using all optimization methods.

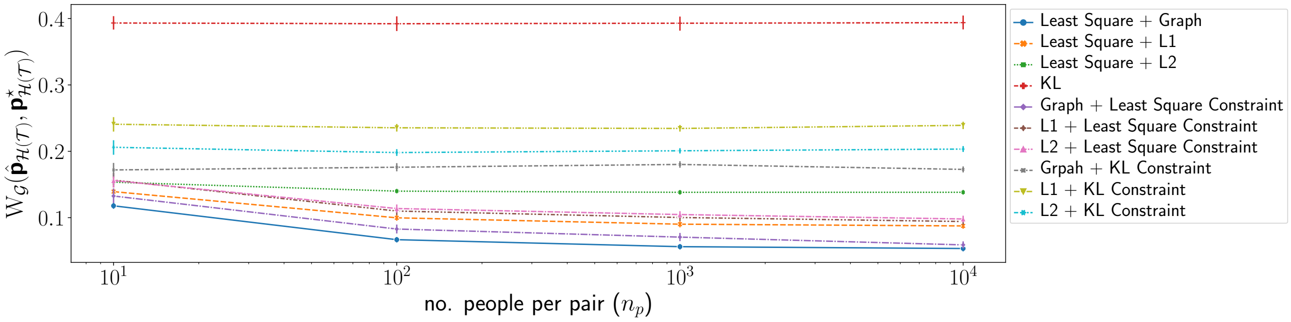


Figure 27. $W_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 2 Gaussian user distribution in 2D with varying n_p using all optimization methods.

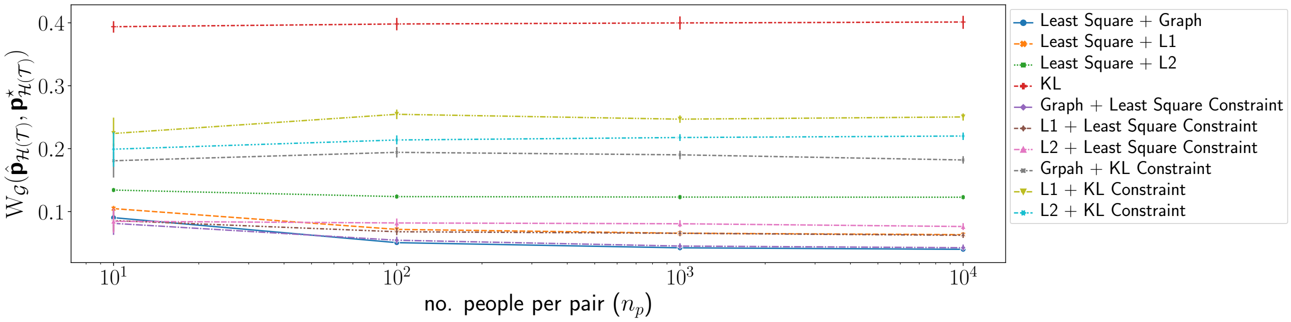


Figure 28. $W_{\mathcal{G}}(\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}, \hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})})$ for a mixture of 3 Gaussian user distribution in 2D with varying n_p using all optimization methods.

D.4. Zappos

The Zappos dataset (UT Zappos50K) (Yu & Grauman, 2014; 2017) comprises 4 major categories of shoes: Boots, Sandals, Shoes, and Slippers. Each major category includes several minor categories. For instance, within the Boots category, you can find Ankle, Knee High, Mid-Calf, Over the Knee, and Prewalker Boots. Table 1 shows the major and minor categories in the Zappos dataset.

Major Category	**Minor Category**
Boots	Ankle, Knee High, Mid-Calf, Over the Knee, Prewalker Boots
Sandals	Athletic, Flat, Heel
Shoes	Boat Shoes, Clogs and Mules, Crib Shoes, Firstwalker, Flats, Heels, Loafers, Oxfords, Prewalker, Sneakers and Atheletic Shoes
Slippers	Boot, Slipper Flats, Slipper Heels

Table 1. Major and minor categories in the Zappos dataset.

Data Preprocessing: We consider minor category, that has a cardinality of 21, as the label space. To ensure that all the images have the same dimension, we use the Zappos image square dataset. Then, we resize them to 135 × 135. Lastly, we convert the images into grey scale.

We train a modified VGG11 convolutional neural network (Simonyan & Zisserman, 2014) on the Zappos dataset. VGG11 is intended to be trained on ImageNet (Deng et al., 2009), which has 1000 classes. We modify the last layer of the network so that it works with 21 classes. We insert a new layer as the penultimate layer of the network. This is because the original penultimate layer has an output of dimension 4096, which is too large. By reducing it to 512, we can employ the output of this penultimate layer as the embedding for the Zappos dataset.

Training: We use 80% of the dataset as the training set and the rest as the test set, both with a batch size of 64. We use SGD optimizer with learning rate 0.01, momentum 0.9, and weight decay 0.0005. After 12 epochs, we achieve a train accuracy of 94.05% and a test accuracy of 86.81%. To generate an embedding for the Zappos dataset, we feed the entire dataset into the trained network and extract the output from the penultimate layer, resulting in a matrix of dimensions 50066×512. We use PaCMAP (Wang et al., 2021b) with default parameters to obtain the 2D embedding of shoes as shown in Figure 29.

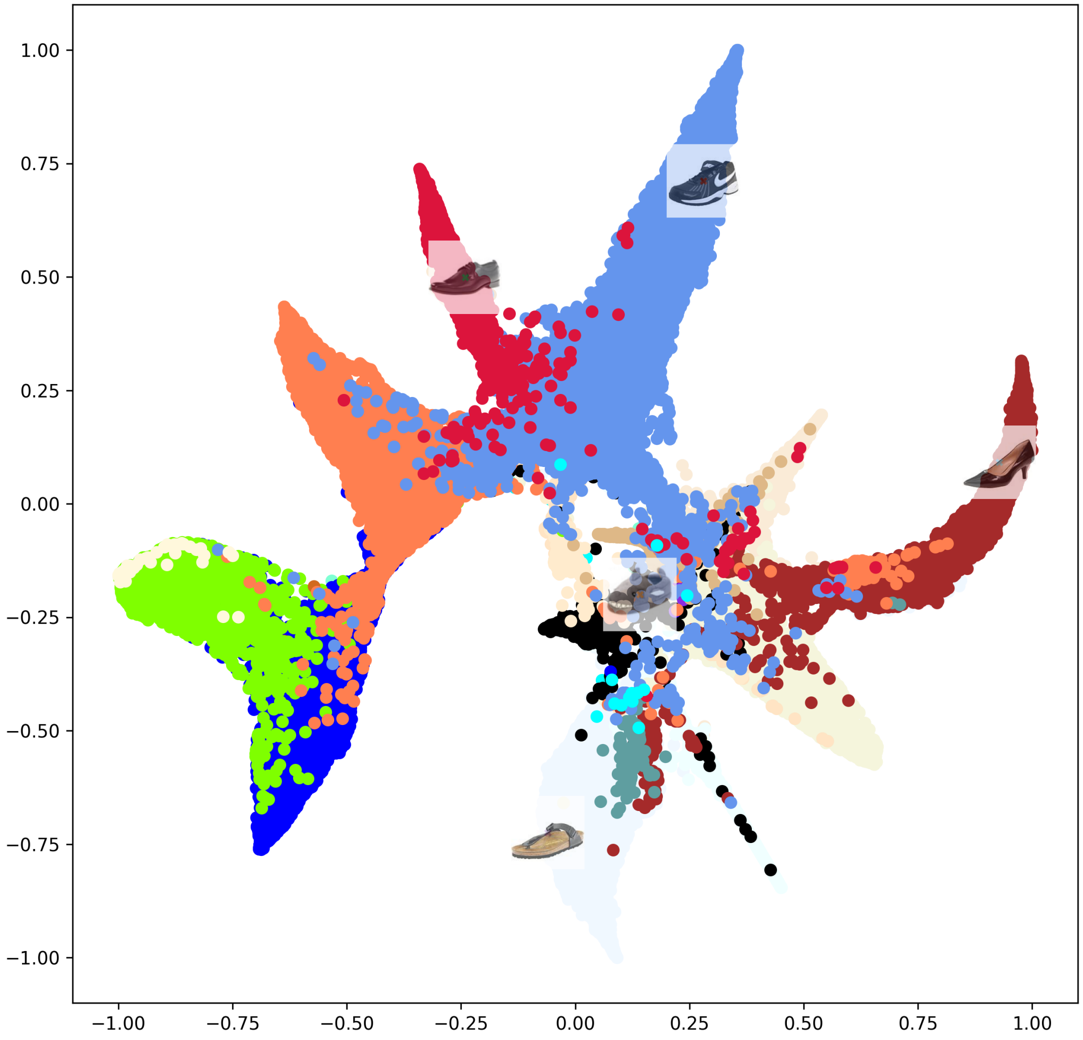


Figure 29. 2D embedding of the 5 shoes obtained using penultimate layer of modified VGG11 and PaCMAP. Each color represents a minor category. 5 shoes we used for experiments are also located.

Data Collection via Crowdsourcing: We pick 5 shoes as our query item set (30).

Figure 31 shows the instructions provided and the interface for answering pairwise comparison queries.

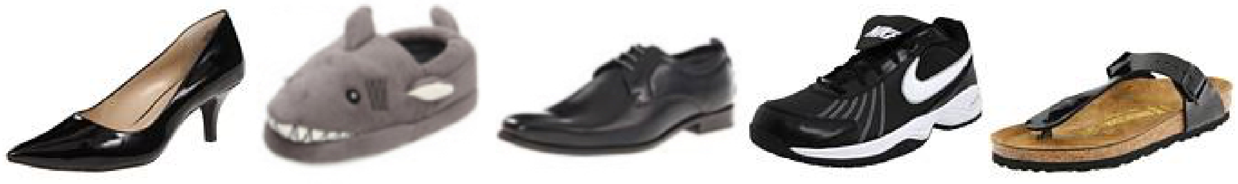
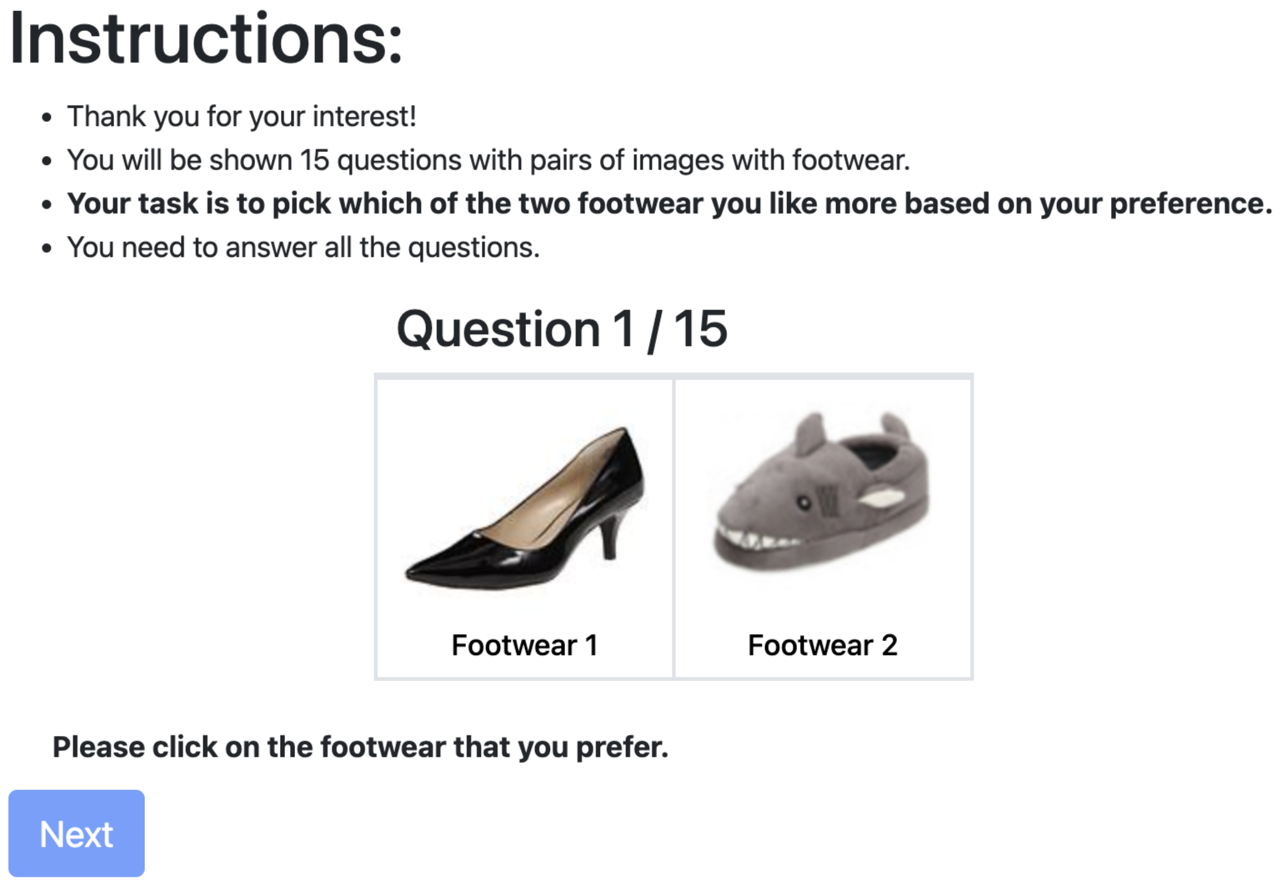


Figure 30. The 5 shoes we pick for pairwise comparison task on Amazon Mechanical Turk.



Instructions:

- Thank you for your interest!
- You will be shown 15 questions with pairs of images with footwear.
- **Your task is to pick which of the two footwear you like more based on your preference.**
- You need to answer all the questions.

Question 1 / 15

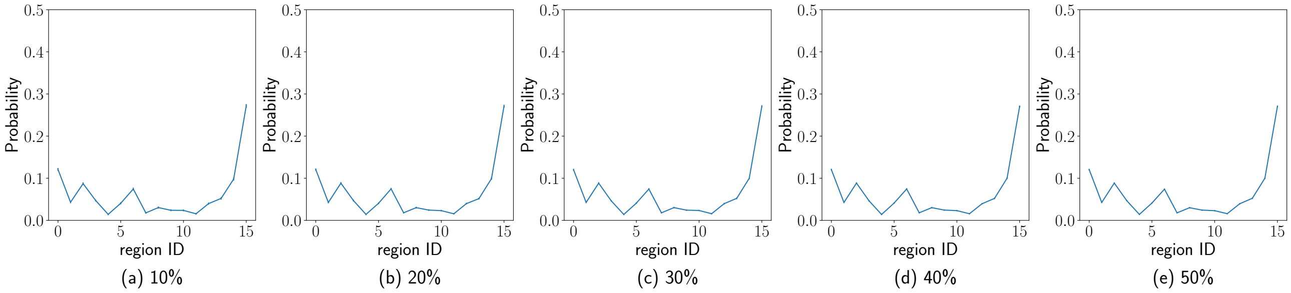
Please click on the footwear that you prefer.

Figure 31. Amazon Mechanical Turk Task interface

We posted this task on Amazon Mechanical Turk (AMT) (MTurk). Each task has 15 pairwise comparison queries (10 pairs and 5 repeats). The median time taken per query is around $2.58s$ and for the task (15 pair comparisons) is $\sim 47s$. Each worker is paid 15 cents per task. This is roughly $\sim \$7$ per hour. We did not restrict the task to master workers. The task was open to all those who had at least 500 HITs approved and 95% approval rate.

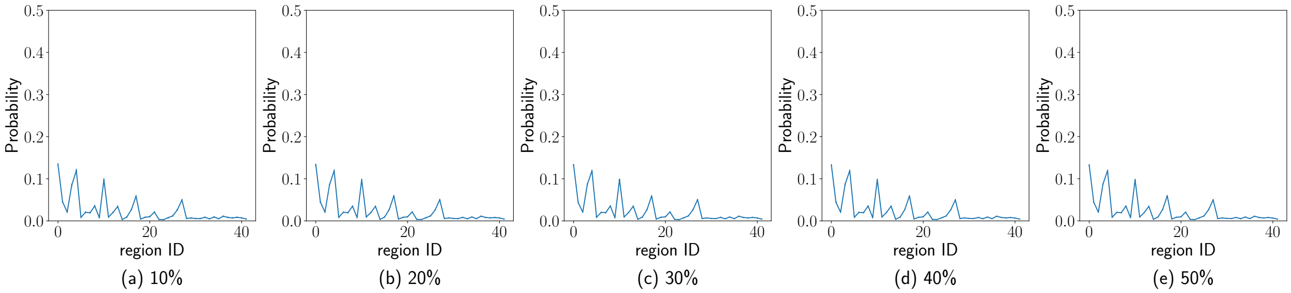
We first bootstrap $10\%, 20\%, 30\%, 40\%, 50\%$ of all workers, repeating the process 100 times for each percentage. Then, we use answers to all possible queries from these workers to estimate the true mass with $n_h = 5$ and $n_h = 10$. We create a global bin (initialized to 0) whose size equals to the number of regions formed by the hyperplanes. Each worker has its own local bin (initialized to 0) that has the same size as the global bin. For each pairwise comparison query, the worker can only be on one side of the corresponding hyperplane. Consider all polytopes on the side of the hyperplane related to worker's answer. We increase the corresponding entries of these polytopes in the bin by 1. After we examine all queries, a set of entries has the maximum value among all entries in the bin. Ideally, this set has a cardinality of 1. However, due to noises and worker's inconsistency, it is possible for the cardinality to be greater than 1. We increase the corresponding entries of this set in the global bin by $\frac{1}{\text{cardinality of the set}}$. After we examine all workers we bootstrapped, we normalize the global bin and obtain a probability vector, which is our estimate of the $\mathbf{p}^\star$.

Figure 32 and 33 illustrate the $\mathbf{p}^\star$ we estimated via bootstrapping. It can be seen that the true distribution $\mathbf{p}^\star$ that we estimated is stable across different bootstrap settings.



(a) 10% (b) 20% (c) 30% (d) 40% (e) 50%

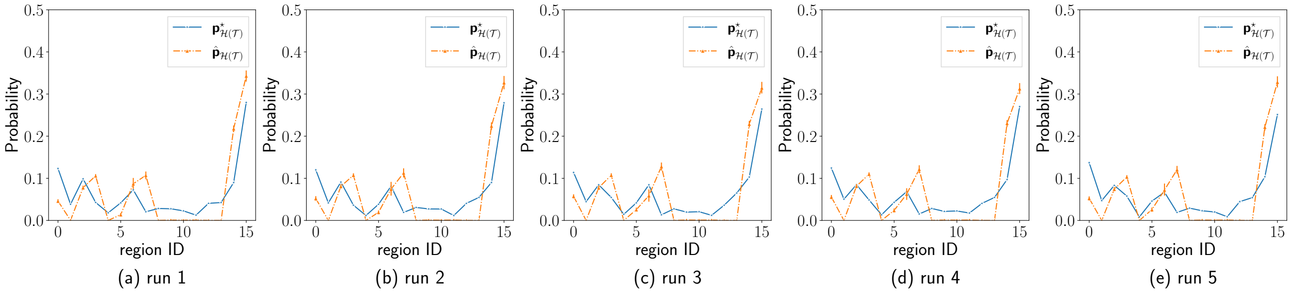
Figure 32. $\mathbf{p}^\star$ obtained via bootstrap (a) 10% (b) 20% (c) 30% (d) 40% (e) 50% of all crowdworkers when $n_h = 5$.



(a) 10% (b) 20% (c) 30% (d) 40% (e) 50%

Figure 33. $\mathbf{p}^\star$ obtained via bootstrap (a) 10% (b) 20% (c) 30% (d) 40% (e) 50% of all crowdworkers when $n_h = 5$.

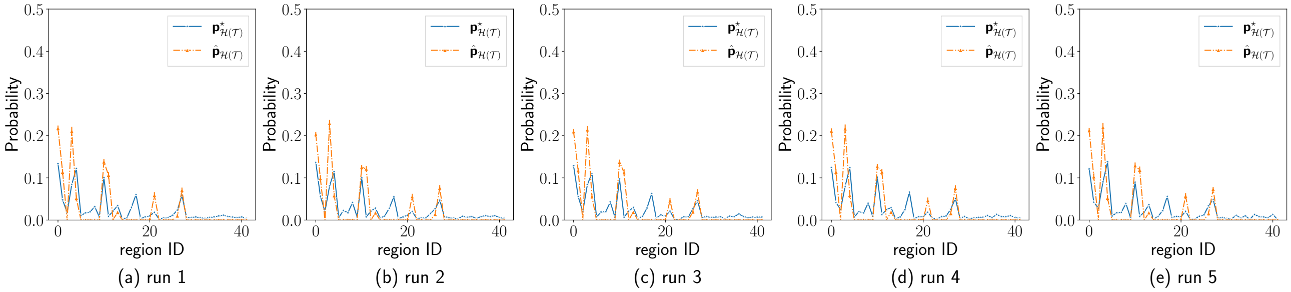
We also present $\hat{\mathbf{p}}$ that we obtain via our method. We first use 20% of crowdworkers to estimate $\mathbf{p}^\star$. Then, use the remaining 80% of crowdworkers to answer the pairwise comparisons and estimate $\hat{\mathbf{p}}$ using our method. We shuffle the remaining 80% of crowdworkers 100 times to obtain 100 different $\hat{\mathbf{q}}$ and hence 100 different $\hat{\mathbf{p}}$. We repeat the above process 5 times (each time with different 20% of crowdworkers to estimate $\mathbf{p}^\star$) for both $n_h = 5$ and $n_h = 10$. The results are presented in Figure 34 and 35. The bounds for $\mathbf{p}^\star$ are presented in Figure 36 and 37 for $n_h = 5$ and $n_h = 10$, respectively.



(a) run 1 (b) run 2 (c) run 3 (d) run 4 (e) run 5

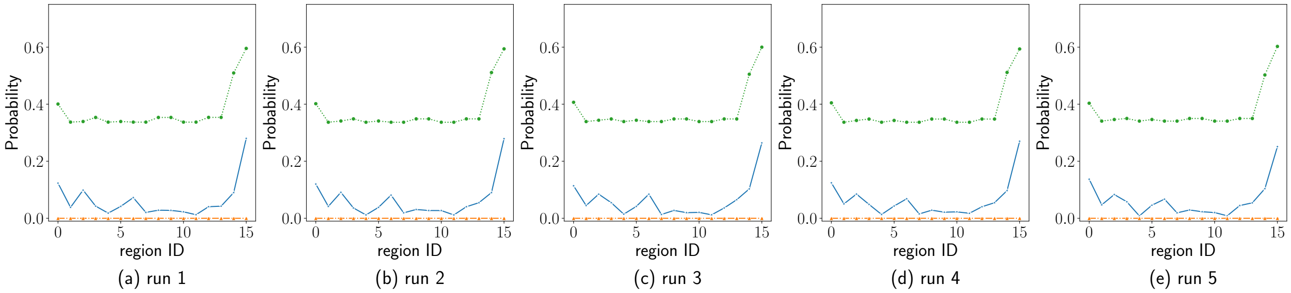
Figure 34. $\mathbf{p}^\star$ obtained using 20% of crowdworkers and $\hat{\mathbf{p}}$ obtained using the remaining 80% of the crowdworkers 100 times. Each of the (a)-(e) uses different set of 20% of all crowdworkers to obtain $\mathbf{p}^\star$, when $n_h = 5$.

Figure 38 shows the TV and W_G between $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ while we vary n_h from 1 to 10.



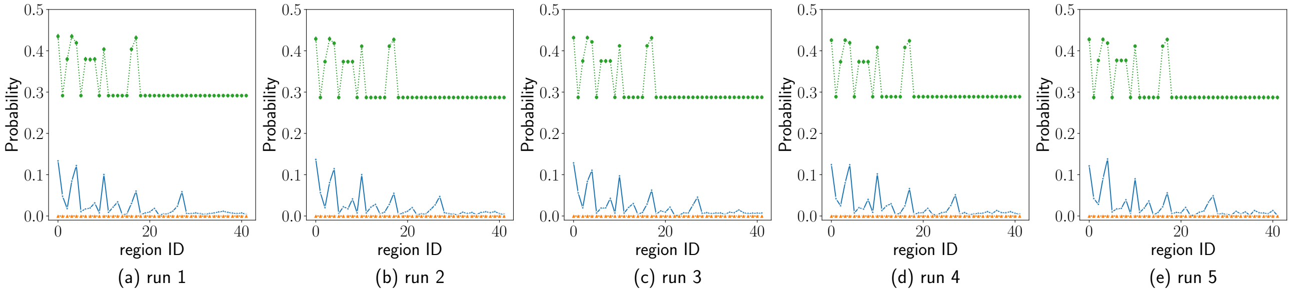
(a) run 1 (b) run 2 (c) run 3 (d) run 4 (e) run 5

Figure 35. $\mathbf{p}^\star$ obtained using 20% of crowdworkers and $\hat{\mathbf{p}}$ obtained using the remaining 80% of the crowdworkers 100 times. Each of the (a)-(e) uses different set of 20% of all crowdworkers to obtain $\mathbf{p}^\star$, when $n_h = 10$.



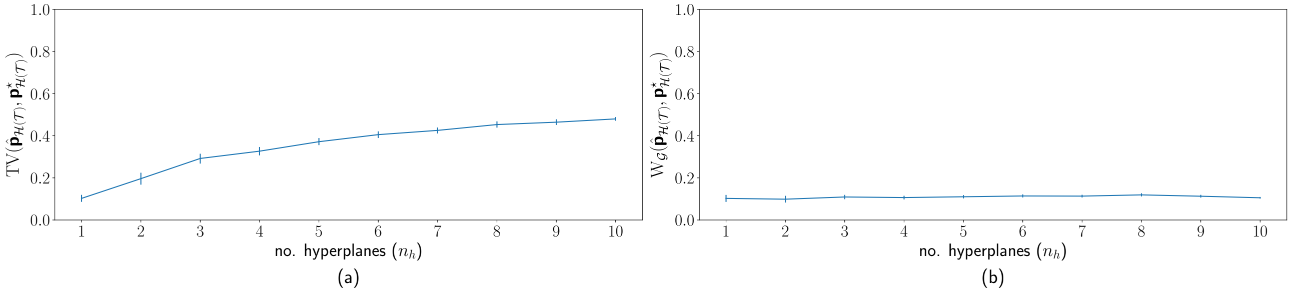
(a) run 1 (b) run 2 (c) run 3 (d) run 4 (e) run 5

Figure 36. Upper and lower bound for $\mathbf{p}^\star$ when $n_h = 5$.



(a) run 1 (b) run 2 (c) run 3 (d) run 4 (e) run 5

Figure 37. Upper and lower bound for $\mathbf{p}^\star$ when $n_h = 10$.



(a) (b)

Figure 38. TV and $W_{\mathcal{G}}$ when we vary n_h.

Lastly, Figure 39-48 illustrate the polytopes formed by the hyperplanes as well as $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ while we vary n_h from 1 to 10.

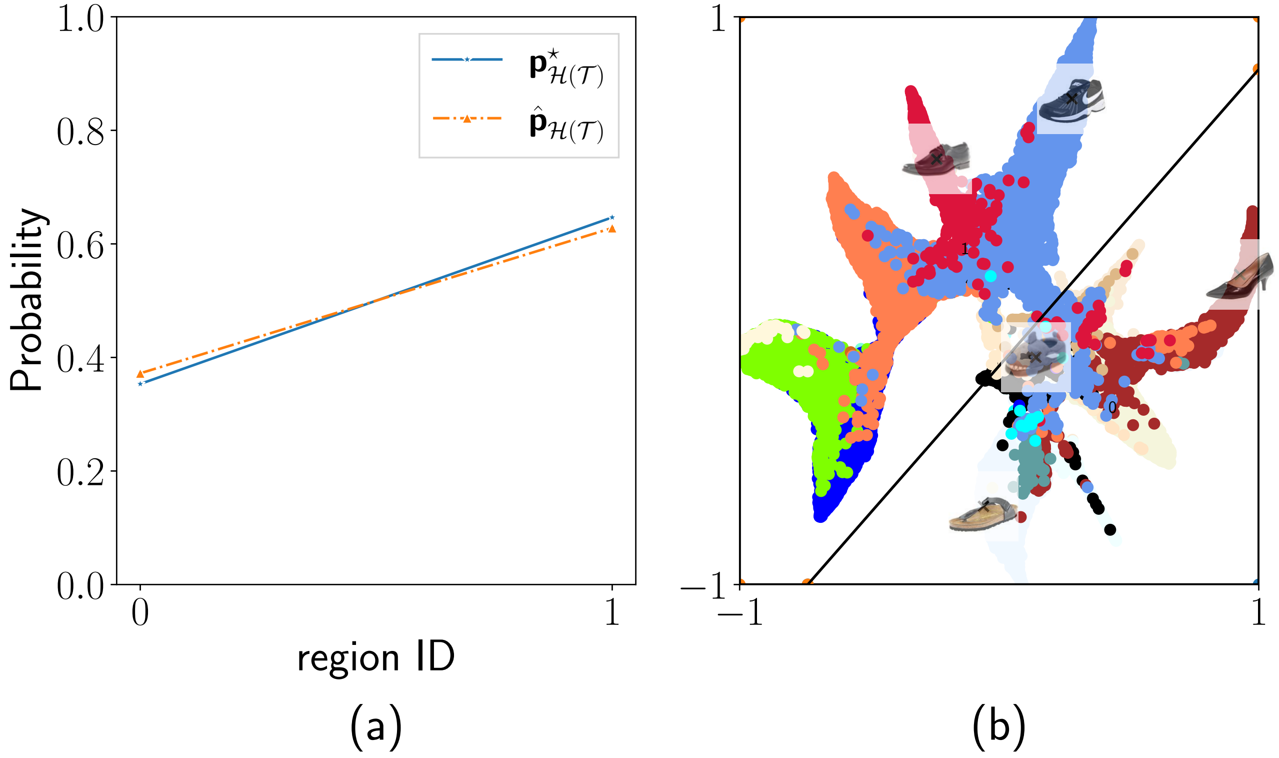


Figure 39. (a) $\mathbf{p}^{\star}$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 1$. (b) regions formed by 1 hyperplane. The numbers in each region corresponds to the region ID.

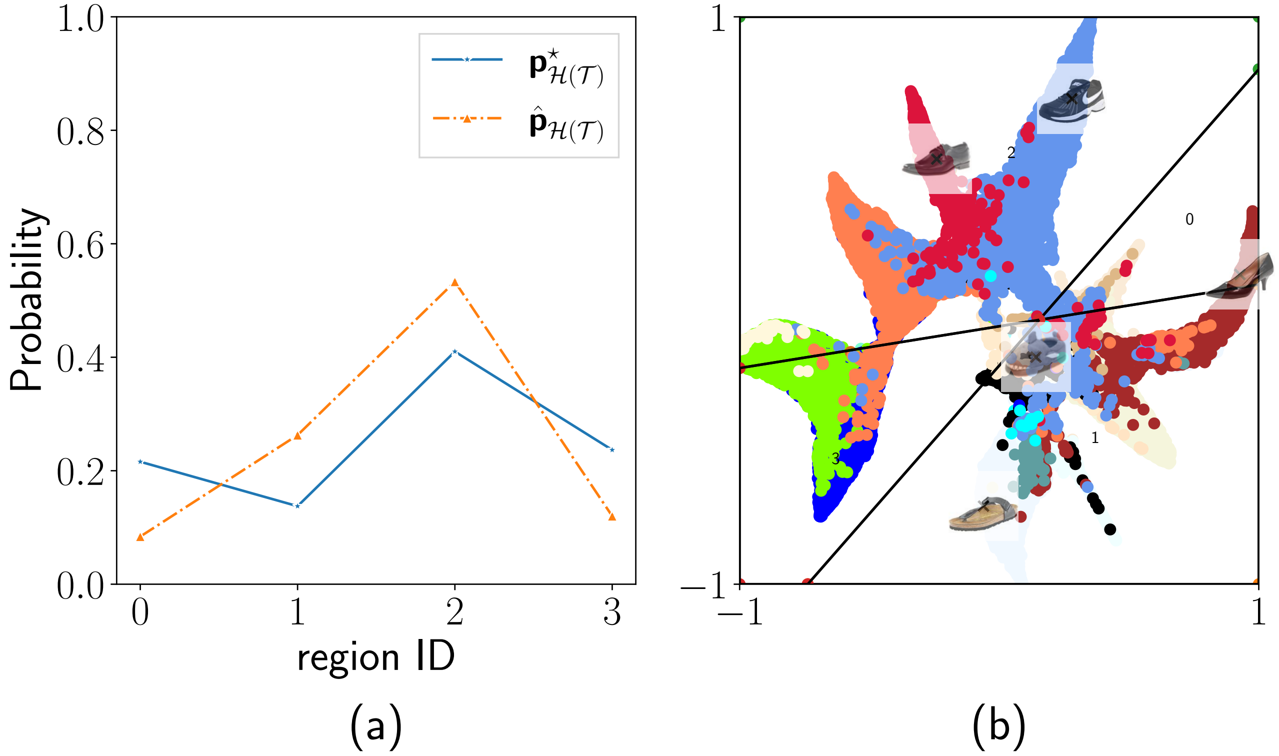
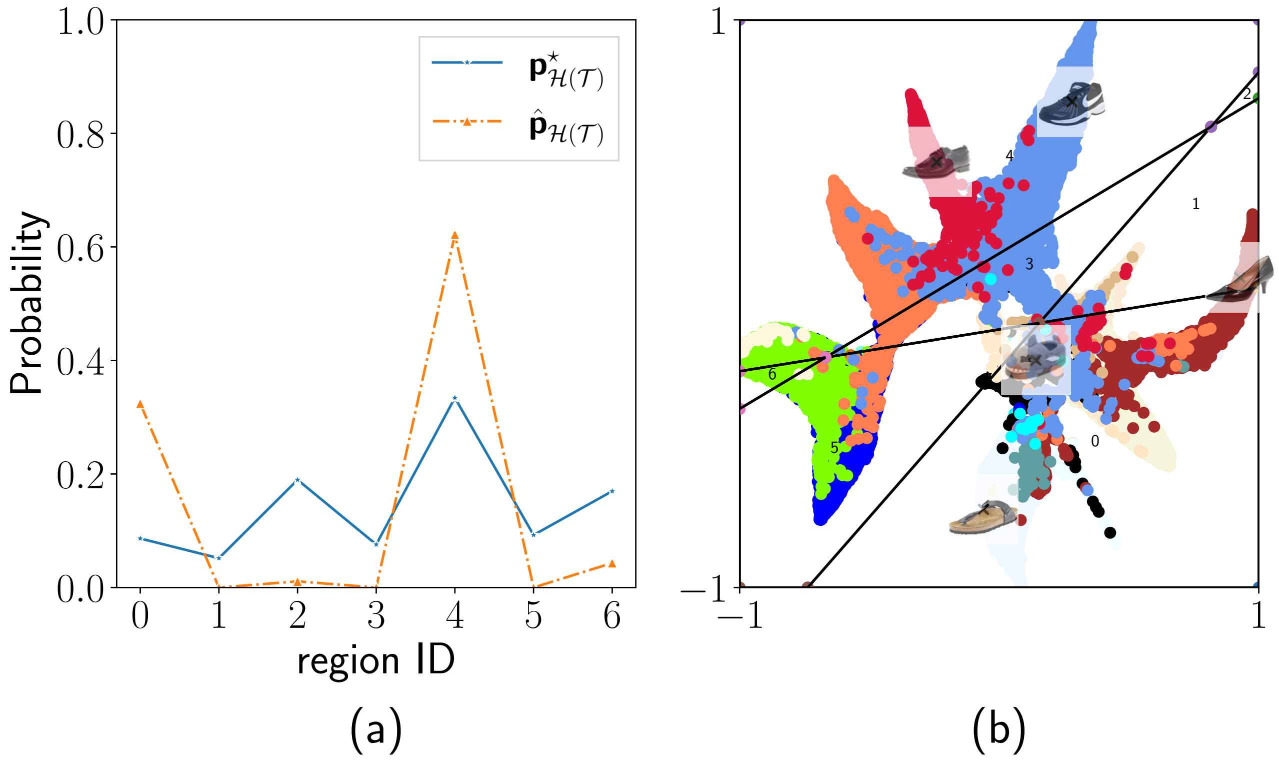


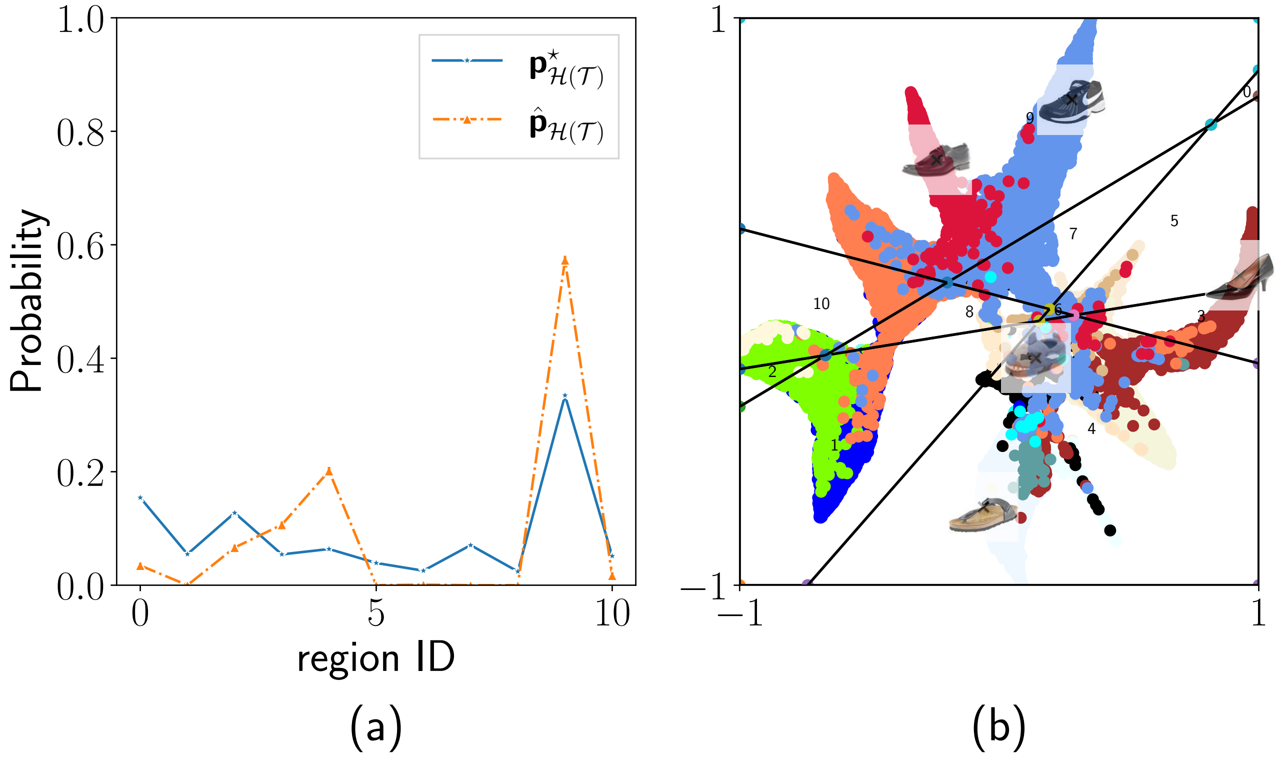
Figure 40. (a) $\mathbf{p}^{\star}$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 2$. (b) regions formed by the 2 hyperplanes. The numbers in each region corresponds to the region ID.



(a)

(b)

Figure 41. (a) $\mathbf{p}^{\star}$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 3$. (b) regions formed by the 3 hyperplanes. The numbers in each region corresponds to the region ID.



(a)

(b)

Figure 42. (a) $\mathbf{p}^{\star}$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 4$. (b) regions formed by the 4 hyperplanes. The numbers in each region corresponds to the region ID.

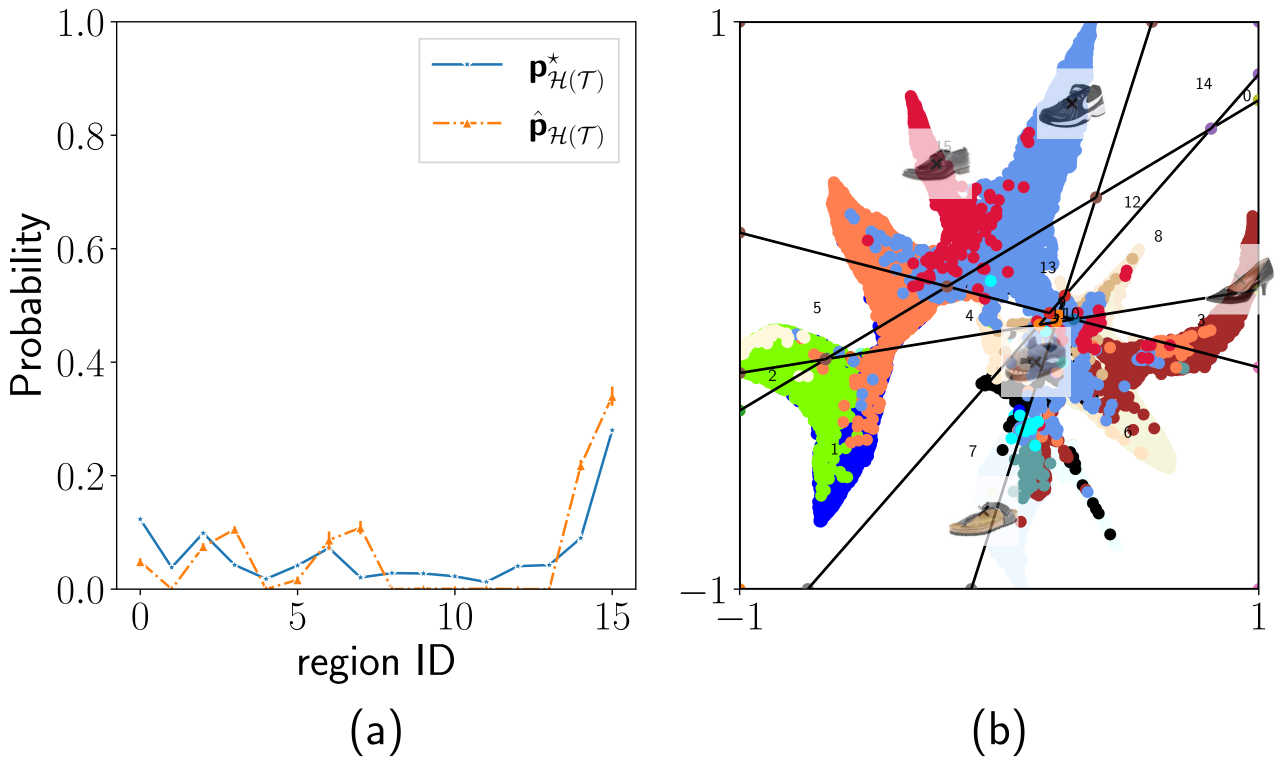


Figure 43. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 5$. (b) regions formed by the 5 hyperplanes. The numbers in each region corresponds to the region ID.

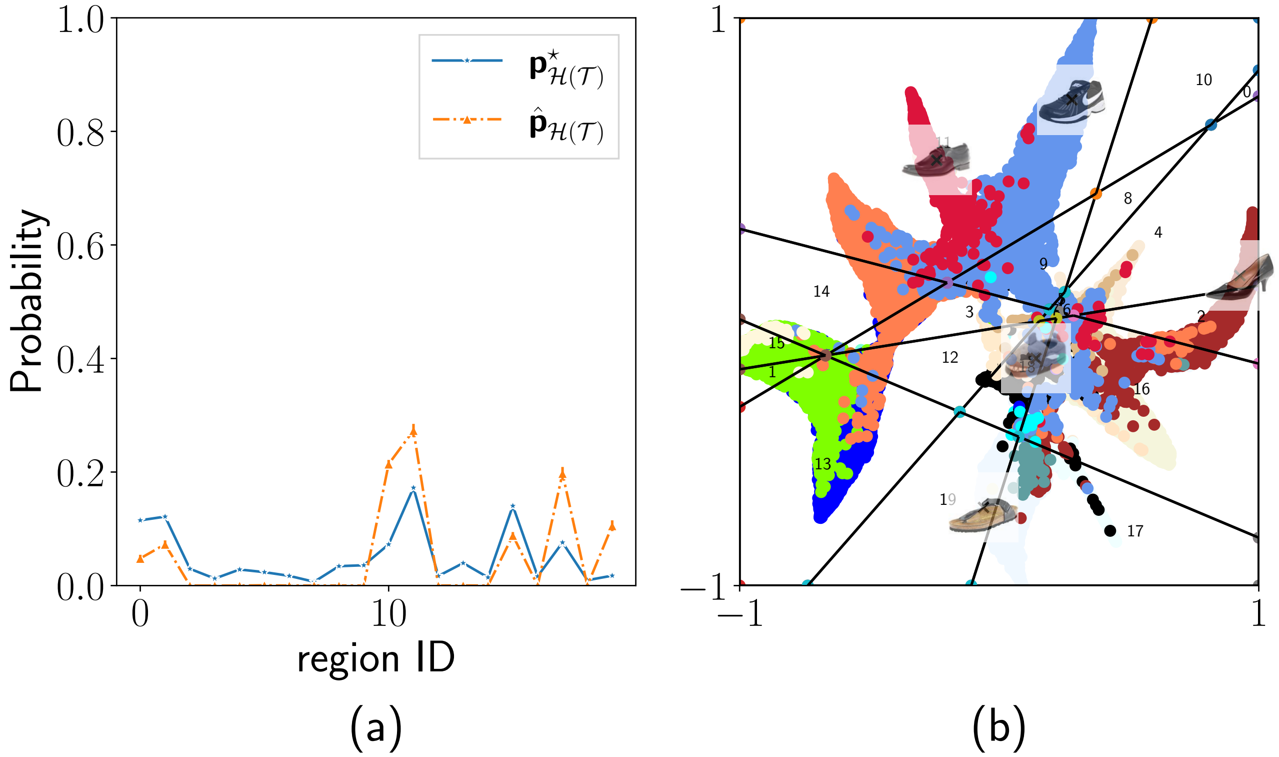
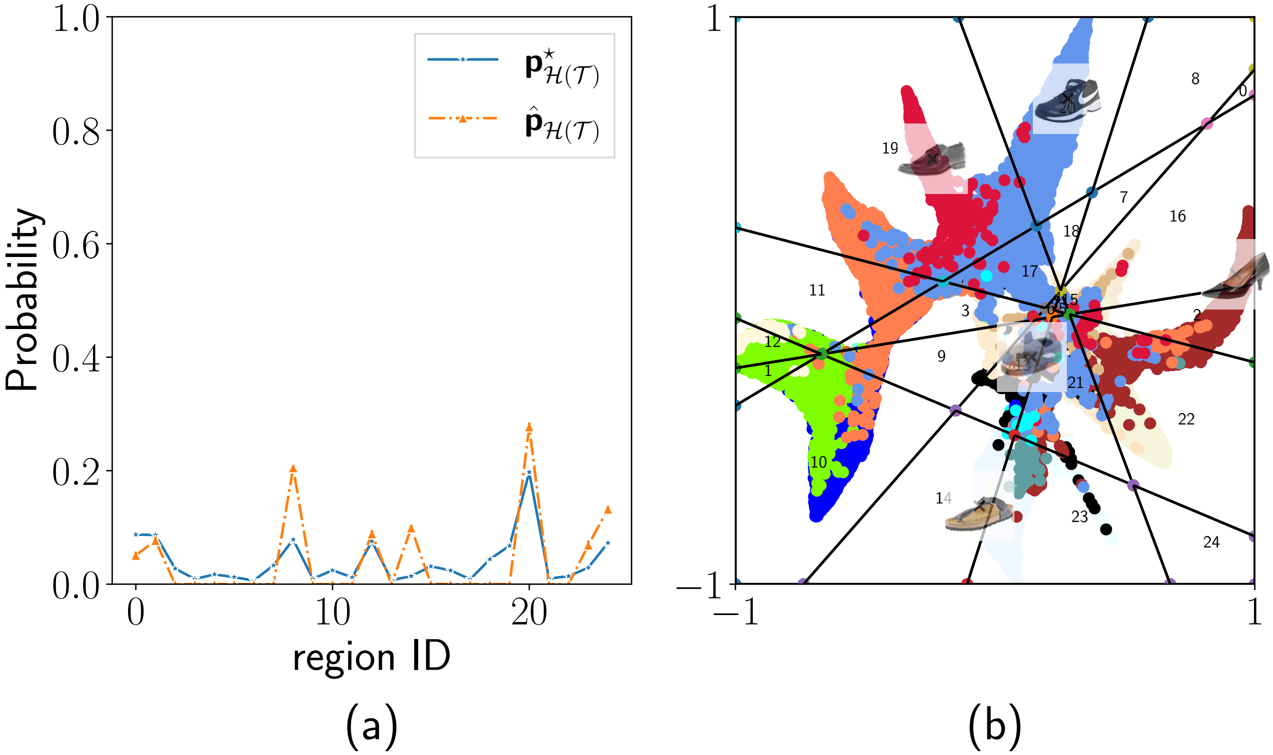
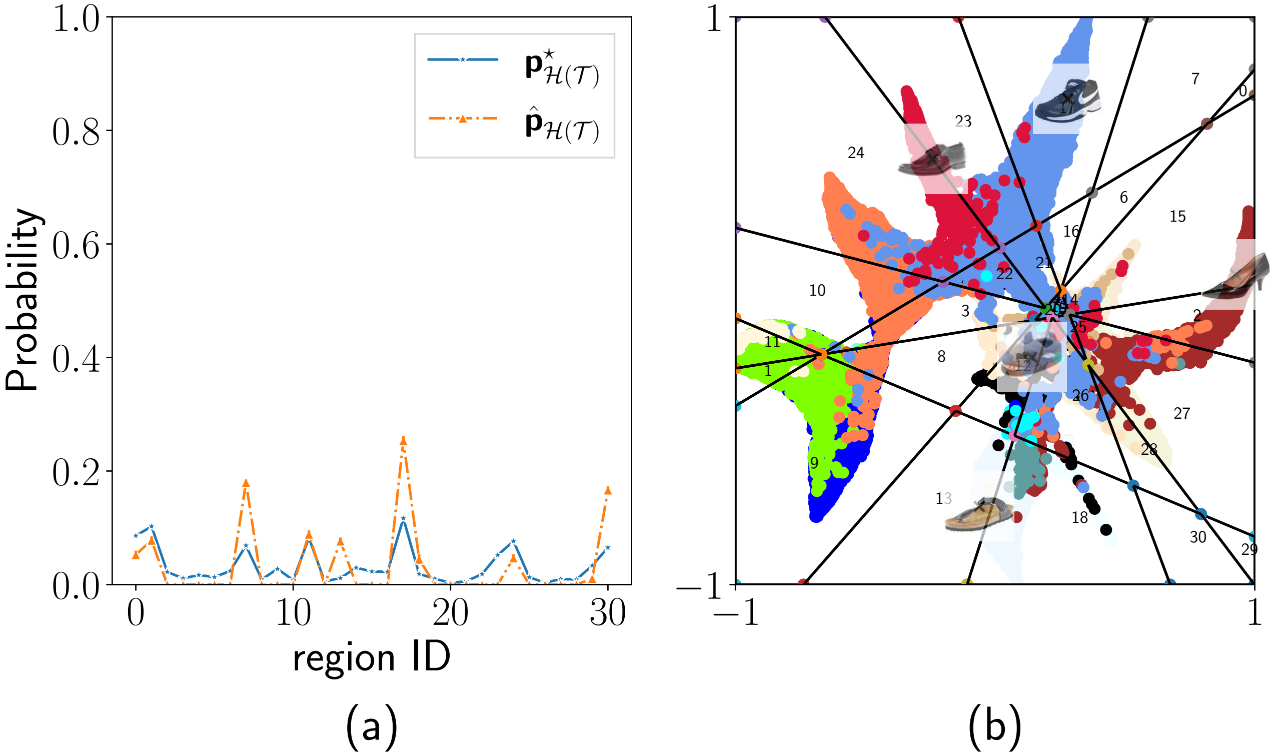


Figure 44. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 6$. (b) regions formed by the 6 hyperplanes. The numbers in each region corresponds to the region ID.



(a) (b)

Figure 45. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 7$. (b) regions formed by the 7 hyperplanes. The numbers in each region corresponds to the region ID.



(a) (b)

Figure 46. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 8$. (b) regions formed by the 8 hyperplanes. The numbers in each region corresponds to the region ID.

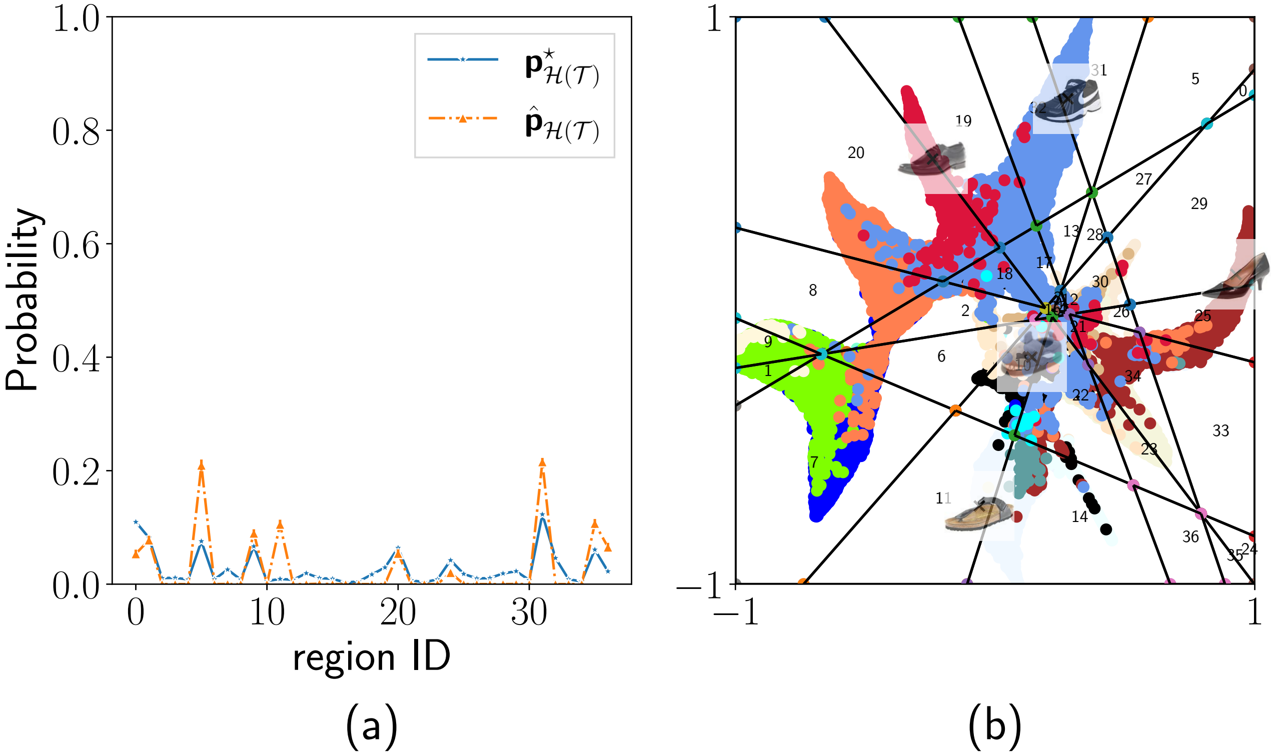


Figure 47. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 9$. (b) regions formed by the 9 hyperplanes. The numbers in each region corresponds to the region ID.

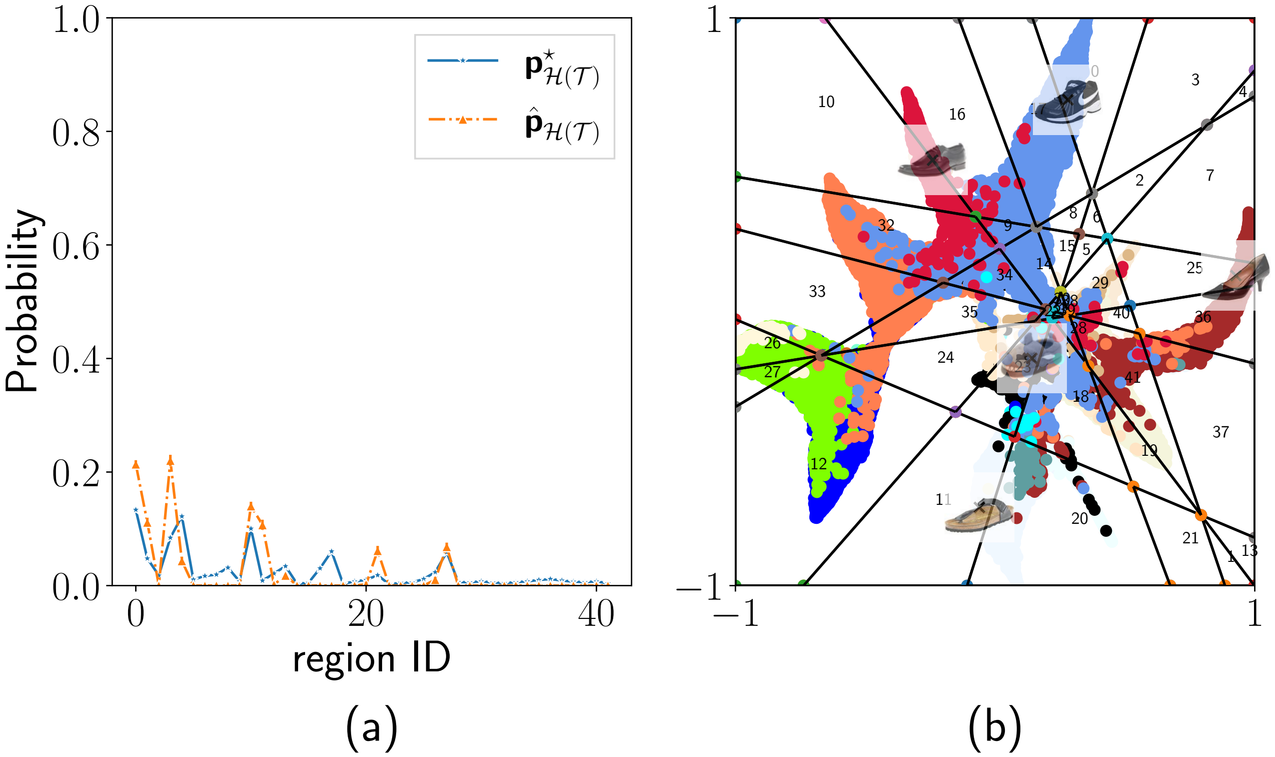


Figure 48. (a) $\mathbf{p}^\star$ and $\hat{\mathbf{p}}$ recovered by our algorithm when $n_h = 10$. (b) regions formed by the 10 hyperplanes. The numbers in each region corresponds to the region ID.

D.4.1. Movies

We create a new dataset of 4266 movies. We use OpenAI's *text-embedding-ada-002* model to generate an 1536 dimensional embedding for each movie. Then, we train a regression neural network, where the target is each movie's average IMDB rating. For this, we use 80% of the movies as the training set and the rest as the test set, where batch size 4. We use SGD optimization with learning rate 0.0001, momentum 0.9, Huber loss. After 250 epochs, we reach a mean average error of 0.63 on the test set.

For pairwise comparisons task, we pick the following 2 sets of movies in a way that most of the movies in those 2 sets have unbalanced opinion in terms of critics by general audience:

- DCEU superheroes (12 DC superhero movies)
- Movie2 (7 movies from US, China, and South Korea)

We scrap critics and audience ratings for selected movies from Rotten Tomatoes. Then, we construct a set of users for each movie from its reviewers. We look for intersections of user sets for each pair of movies. If the size of intersection is small, we discard the corresponding pair. Since movies in DCEU have the similar type and are from the same franchise, it is more likely that we encounter common reviewers. Hence, the definition of small size of intersection is different for the 2 sets. For DCEU, we discard pairs whose size of intersection is less than 200. For Movie2, we discard pairs whose size of intersection is less than 100. This process leaves us with 9 pairs of movies for DCEU and 3 pairs of movies for Movie 2.

For a given pairwise comparison query based on movie pairs, a randomly selected reviewer picks the one that has a higher rating (rated by the same reviewer). If both movies in a pair have the same rating, we pick the movie on the left of the pair. After a reviewer answered one query, we are done with this reviewer. We perform 100 repetitions of calculating $\hat{\mathbf{p}}_{\mathcal{H}(\mathcal{T})}$ by reshuffling users each time, where $n_p = 50$ for DCEU and $n_p = 25$ for Movie2. Since we do not query any user more than once, we do not have enough information to estimate $\mathbf{p}^\star_{\mathcal{H}(\mathcal{T})}$, unlike our experimental work on the Zappos dataset.

Figure 49 (a) (b) show the $\hat{\mathbf{p}}$ recovered using our method and the bounds for $\mathbf{p}^\star$ for the DCEU movie set. Figure 49 (c) (d) show the $\hat{\mathbf{p}}$ recovered using our method and the bounds for $\mathbf{p}^\star$ for the Movie2 movie set. Figure 50 and 51 shows the regions formed by the hyperplanes and the movies' location in the embedding space for DCEU amd Movie2 movie set, respectively.

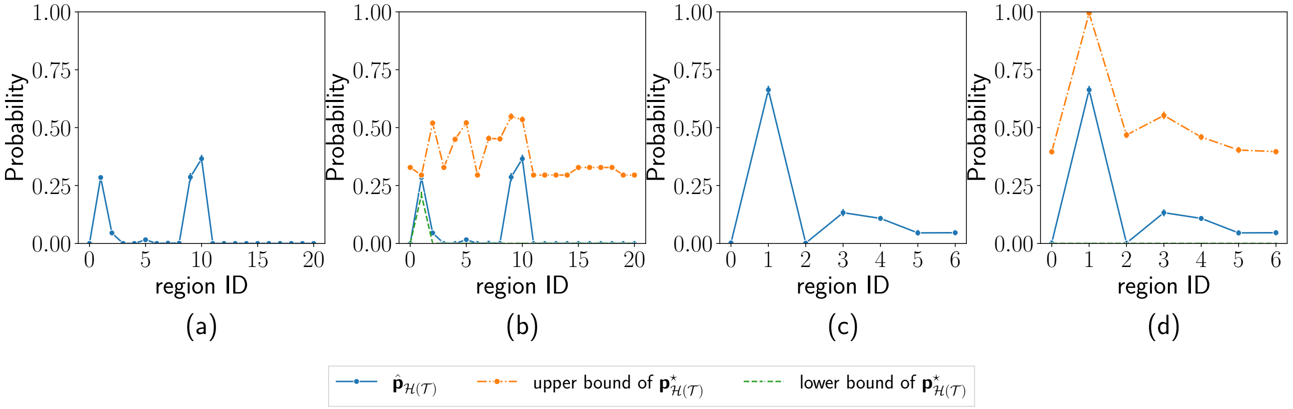


Figure 49. (a) $\hat{\mathbf{p}}$ recovered for the DCEU movie set (b) Bounds for $\mathbf{p}^\star$ for the DCEU movie set (c) $\hat{\mathbf{p}}$ recovered for the Movie2 movie set (d) Bounds for $\mathbf{p}^\star$ for the Movie2 movie set.

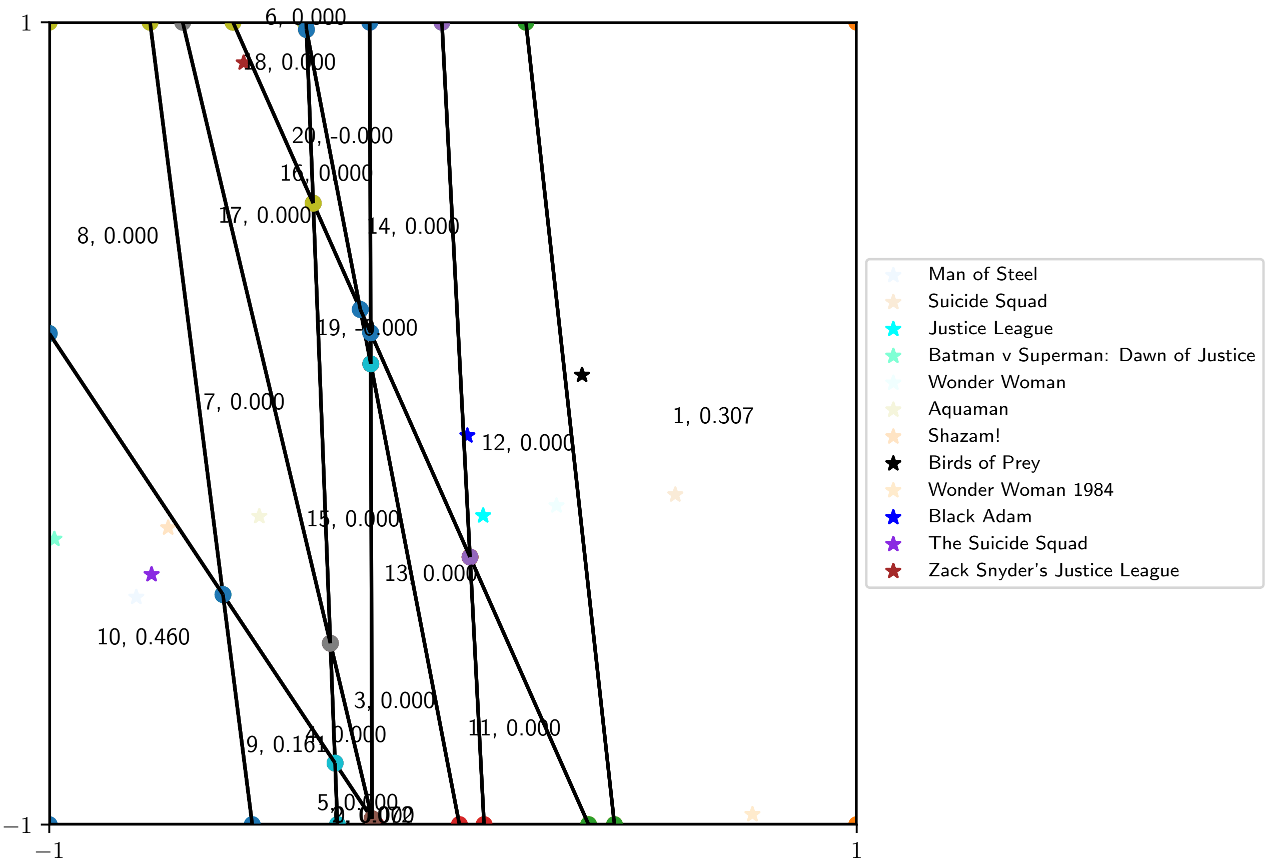


Figure 50. Regions formed by the hyperplanes from DCEU movie set. The numbers in each region represent the region ID as well as the probability mass recovered by our method in that region. Movies in the DCEU movie set are also labeled using their corresponding embedding.

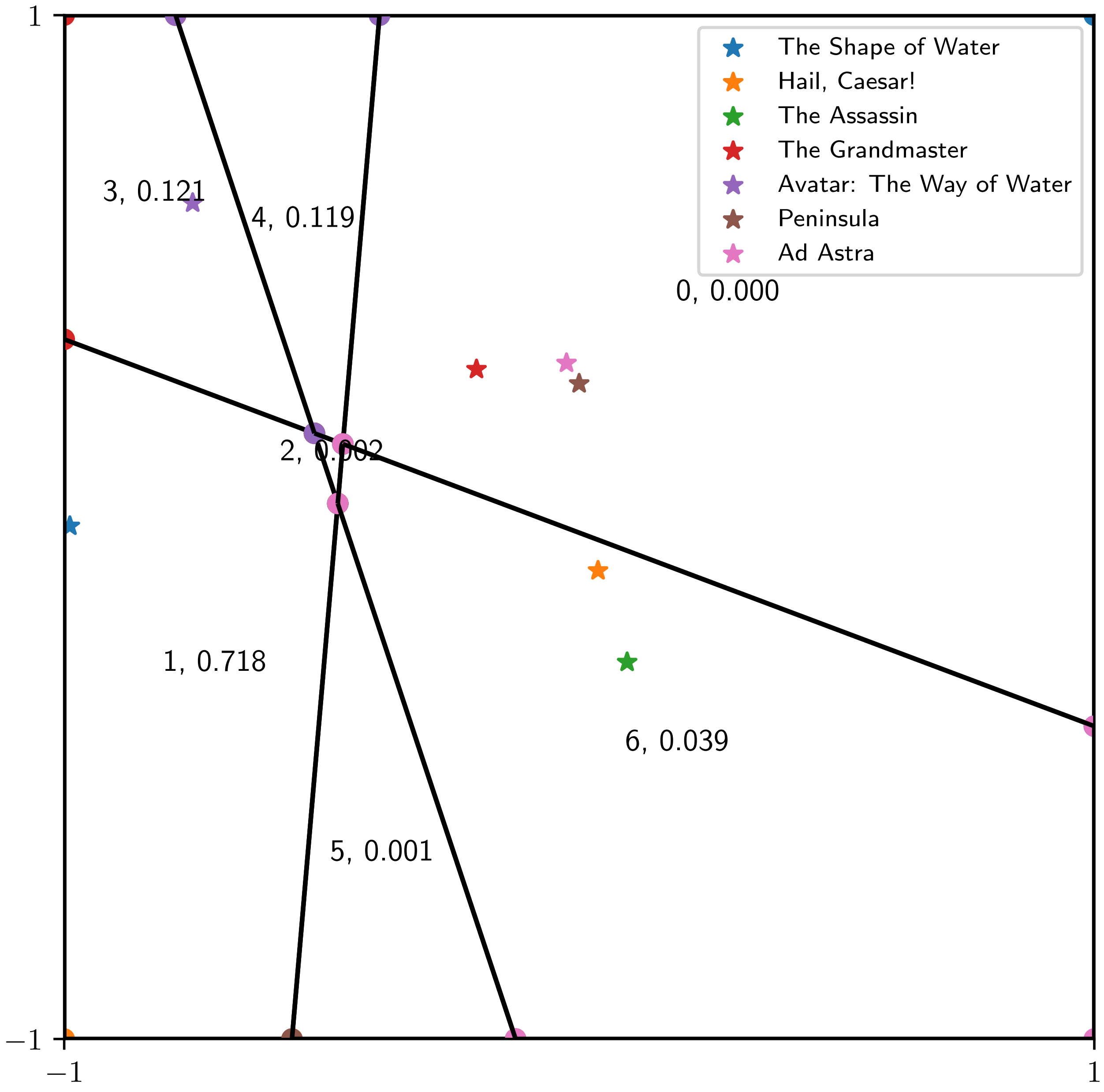


Figure 51. Regions formed by the hyperplanes from Movie2 movie set. The numbers in each region represent the region ID as well as the probability mass recovered by our method in that region. Movies in the Movie2 movie set are also labeled using their corresponding embedding.