On the Exploration of Local Significant Differences For Two-Sample Test

Zhi-Jian Zhou, Jie Ni, Jia-He Yao, Wei Gao
National Key Laboratory for Novel Software Technology, Nanjing University, China
School of Artificial Intelligence, Nanjing University, China
{zhouzj, nij, yaojh, gaow}@lamda.nju.edu.cn

Abstract

Recent years have witnessed increasing attentions on two-sample test with diverse real applications, while this work takes one more step on the exploration of local significant differences for two-sample test. We propose the ME$\text{MaBiD}$, an effective test for two-sample testing, and the basic idea is to exploit local information by multiple Mahalanobis kernels and introduce bi-directional hypothesis for testing. On the exploration of local significant differences, we first partition the embedding space into several rectangle regions via a new splitting criterion, which is relevant to test power and data correlation. We then explore local significant differences based on our bi-directional masked $p$-value together with the ME$\text{MaBiD}$ test. Theoretically, we present the asymptotic distribution and lower bounds of test power for our ME$\text{MaBiD}$ test, and control the familywise error rate on the exploration of local significant differences. We finally conduct extensive experiments to validate the effectiveness of our proposed methods on two-sample test and the exploration of local significant differences.

1 Introduction

Two-sample test has attracted much attention with diverse applications such as cancer detection [1], distribution-shift detection [2], generative modeling [3, 4], etc. The basic problem is to assess whether two i.i.d. samples are drawn from the same distribution. Various kernel-based methods have been developed for two-sample test such as Maximum Mean Discrepancy (MMD) [5–8] and Mean Embedding (ME) [9–11]. Another relevant approach is to construct a binary classifier and assess two samples according to classification performance [12–20]. For an overview of two-sample test, we refer to a survey [21, and references therein].

In many real applications, however, it is necessary to take one more step to explore and understand local significant differences, rather than only two-sample test. For example, a scientific problem in galaxy morphology is to identify some local regions of significant differences between two kinds of galaxies, which is important to discover galaxy formation and evolution history [22]. On the analysis of mass cytometry data in cell biology, researchers are always interested in finding local regions of significantly different abundance between disease and healthy samples [23].

Several attempts have been made to explore local significant differences in the past years. A feasible solution is to partition space into several regions, and identify significantly different regions according to their cardinalities of samples [24–27]. Another relevant work is to identify local significant differences by estimating kernel densities [28, 29] and conditional probabilities [30]. Generally, it is not easy to deal with complex data by simply counting cardinalities of samples, regardless of data intrinsic correlations, and it is also difficult to make accurate estimation of kernel densities and conditional probabilities without sufficient data, especially for many regions with finite samples.
This work presents a new two-sample test from local and directional information, and further explore local significant differences. The main contributions can be summarized as follows:

- We propose the effective ME_MaBiD test for two-sample testing, and the basic idea is to exploit local information by multiple Mahalanobis kernels and introduce bi-directional hypothesis for testing. Intuitively, Mahalanobis kernels are more flexible to exploit local differences from neighborhoods and feature maps, and the bi-directional hypothesis is beneficial to improve the sensitivity of two-sample test with proper parameter adaptation.
- We partition the embedding space into several rectangle regions based on a new splitting criterion, which is relevant to test power and data correlation. We introduce the bi-directional masked \( p \)-value for each rectangle region, and finally explore local regions with significant difference based on our bi-directional masked \( p \)-value together with the ME_MaBiD test.
- We present theoretical guarantees for our ME_MaBiD test via the asymptotic distribution, as well as the lower bounds on the test power for our test. We also present the upper bounds on familywise error rate for our exploration of local significant differences.
- We conduct extensive experiments to validate the effectiveness and efficiency of our methods. Specifically, our methods achieve better performance on most datasets for two-sample test and exploring local significant differences, along with comparable or smaller running time.

The rest of this work is organized as follows: Section 2 presents our ME_MaBiD test. Section 3 explores local significant differences. Section 4 conducts extensive experiments, and Section 5 concludes with future work. All technical proofs are given in Appendix A.

2 Our ME_MaBiD Test for Two-Sample Testing

Let \( \mathbb{P} \) and \( \mathbb{Q} \) denote two (unknown) Borel probability measures over an instance space \( \mathcal{X} \subseteq \mathbb{R}^d \), and \( X = \{x_1, \ldots, x_m\} \) and \( Y = \{y_1, \ldots, y_n\} \) are two i.i.d. samples from \( \mathbb{P} \) and \( \mathbb{Q} \), respectively. The goal of two-sample test is to assess whether \( X \) and \( Y \) are drawn from the same distribution; in other words, we aim to assess whether \( \mathbb{P} = \mathbb{Q} \) from two samples \( X \) and \( Y \).

We introduce some necessary notations used in this work. Write \( \left\{ \tau \right\} = \{1, 2, \ldots, \tau\} \) for integer \( \tau \geq 2 \), and \(|A|\) denotes the cardinality of set \( A \). Let \( I_d \) be the identity matrix of size \( d \times d \). For a vector \( \mathbf{a} = [a_1, a_2, \ldots, a_d] \), denote by \( \text{sgn}(\mathbf{a}) = [\text{sgn}(a_1), \text{sgn}(a_2), \ldots, \text{sgn}(a_d)] \) with \( \text{sgn}(a_i) = a_i/|a_i| \) for \( a_i \neq 0 \); otherwise, \( \text{sgn}(a_i) = 0 \). Let \( \chi^2_{d,\alpha} \) be the \( \chi^2 \) distribution with \( \ell \) degree of freedom, as well as the \( p \)-value function \( \chi^2_{d,\alpha}(\cdot) \). Denote by \( \chi^2_{d,\alpha} \) the \( \alpha \)-quantile of distribution \( \chi^2_{d,\alpha} \) for \( \alpha \in (0, 1) \).

Learning multiple Mahalanobis kernels via maximizing test power in training

Following ME test [9, 10], we begin with a set of test locations \( \mathcal{V} = \{v_1, v_2, \ldots, v_\ell\} \subset \mathcal{X} \) to construct discriminative features. For every \( v_i \in \mathcal{V} \), we introduce a Mahalanobis kernel as follows:

\[
\kappa_i(x, v_i) = \exp \left( -\left( x - v_i \right)^	op M_i \left( x - v_i \right) / 2\gamma_i^2 \right) \quad \text{for } \gamma_i > 0 \text{ and positive definite matrix } M_i. \quad (1)
\]

Here, we propose multiple Mahalanobis kernels for two-sample test, which is motivated from multiple kernel learning [31, 32] and Mahalanobis distance [33–35]. The advantage of multiple Mahalanobis kernels is to exploit intrinsic structures and correlations from different directions and regions, and adjust geometrical distribution of data so as to enlarge the distance between different samples [36, 37].
This is different from previous Gaussian kernel \( \kappa_i(x, v_i) = \exp(-\|x - v_i\|^2 / 2\gamma^2) \) [10, 38], which deals with every direction isotropically without difference. Figure 1 presents an illustration of different contours of Mahalanobis and Gaussian kernels for two-sample test. As we can see, Mahalanobis kernels are more flexible to exploit different directional information than Gaussian kernels. Our work is also different from previous deep kernel approaches [4, 39], which train single one deep neural network combined with Gaussian kernel for variations in distribution smoothness and shape.

We then embed each element in \( X = \{x_i\}_{i=1}^m \) and \( Y = \{y_j\}_{j=1}^n \) into an \( \ell \)-dimensional space as

\[
\hat{x}_i = (\kappa_i(x_i, v_1), \cdots, \kappa_i(x_i, v_\ell))^T \quad \text{and} \quad \hat{y}_j = (\kappa_j(y_j, v_1), \cdots, \kappa_j(y_j, v_\ell))^T,
\]

Denote by \( \hat{X} = \{\hat{x}_i\}_{i=1}^m \) and \( \hat{Y} = \{\hat{y}_j\}_{j=1}^n \). We define the pooled covariance matrix as

\[
\Sigma_{\hat{X}, \hat{Y}} = \sum_{i=1}^m (\hat{x}_i - c_{\hat{X}})(\hat{x}_i - c_{\hat{X}})^T + \sum_{j=1}^n (\hat{y}_j - c_{\hat{Y}})(\hat{y}_j - c_{\hat{Y}})^T + \epsilon I_d,
\]

where \( c_{\hat{X}} = \sum_{i=1}^m \hat{x}_i / m \) and \( c_{\hat{Y}} = \sum_{j=1}^n \hat{y}_j / n \), and \( \epsilon I_d \) is introduced to guarantee the positive definiteness for small constant \( \epsilon > 0 \). We consider the Hotelling \( T^2 \) statistic, as in [40–42],

\[
T(\hat{X}, \hat{Y}) = mn(c_{\hat{X}} - c_{\hat{Y}})^T \Sigma_{\hat{X}, \hat{Y}}^{-1}(c_{\hat{X}} - c_{\hat{Y}}) / (m + n).
\]

Test power is the probability of correctly identifying two different samples. Maximizing \( T(\hat{X}, \hat{Y}) \) is essentially equivalent to maximizing a lower bound of test power \([10, 11]\), and we learn test locations and Mahalanobis kernels as follows:

\[
\{\mathcal{V}, \mathcal{M}_1, \cdots, \mathcal{M}_r, \gamma_1, \cdots, \gamma_{\ell}\} \in \arg \max \{T(\hat{X}, \hat{Y})\}.
\]

We decompose \( \Sigma_{\hat{X}, \hat{Y}} = LL^T \) via the Schur method [44] to remove feature correlations, and it follows

\[
T(\hat{X}, \hat{Y}) = \frac{mn}{m+n} (c_{\hat{X}} - c_{\hat{Y}})^T \Sigma_{\hat{X}, \hat{Y}}^{-1}(c_{\hat{X}} - c_{\hat{Y}}) = \frac{mn}{m+n} \|L^{-1}c_{\hat{X}} - L^{-1}c_{\hat{Y}}\|_2^2.
\]

Hence, \( T(\hat{X}, \hat{Y}) \) essentially measures the difference between two samples via the \( L_2 \)-norm of vector \( L^{-1}c_{\hat{X}} - L^{-1}c_{\hat{Y}} \). We further exploit their inference direction, defined by

\[
F = \text{sgn} (L^{-1}c_{\hat{X}} - L^{-1}c_{\hat{Y}}) \in \{-1, 0, +1\}^{\ell}.
\]

**Bi-directional hypothesis for testing**

Let \( \hat{P} \) and \( \hat{Q} \) be the corresponding embedding distributions from the original \( P \) and \( Q \), respectively. Denote by \( \mu_{\hat{P}} = E_{\hat{P}}[\hat{x}'] \) and \( \mu_{\hat{Q}} = E_{\hat{Q}}[\hat{y}'] \). We consider the following null hypothesis

\[
H_0: \mu_{\hat{P}} = \mu_{\hat{Q}}.
\]

The null hypothesis \( H_0 \) can be used to test whether \( P = Q \) by the following lemma:

**Lemma 1.** We have \( \mu_{\hat{P}} = \mu_{\hat{Q}} \iff P = Q \), for bounded Mahalanobis kernels \( \{\kappa_j\}_{j=1}^\ell \) and for test locations \( \{v_i\}_{i=1}^r \) drawn i.i.d. from a absolutely-continuous distribution w.r.t. Lebesgue measure.

Let \( \hat{X}' = (\hat{x}_i')_{i=1}^m \) and \( \hat{Y}' = (\hat{y}_j')_{j=1}^n \) denote two embedding testing samples. We make similar Schur decomposition \( \Sigma_{\hat{X}', \hat{Y}'} = L'L', \) and calculate testing statistic \( T(\hat{X}', \hat{Y}') \) according to Eqn. (4).

Based on Lemma 1, we can present the asymptotic distribution of statistic \( T(\hat{X}', \hat{Y}') \) as follows:

**Theorem 2.** The testing statistic \( T(\hat{X}', \hat{Y}') \) is almost surely asymptotically distributed as \( \chi^2_{\ell} \) if \( P = Q \); otherwise, \( \chi^2_{\ell}(T(\hat{X}', \hat{Y}')) \to 0 \) as \( m'n'/(m' + n') \to \infty. \)

From Theorem 2, we propose the bi-directional hypothesis, by considering inference direction,

\[
h(\hat{X}', \hat{Y}') = \left\{ \begin{array}{ll}
\| \chi^2_{\ell}(T(\hat{X}', \hat{Y}')) \| \leq \beta \alpha & \text{for } F^T L^{-1}(c_{\hat{X}'} - c_{\hat{Y}'}) \geq 0 \\
\| \chi^2_{\ell}(T(\hat{X}', \hat{Y}')) \| \leq (2 - \beta) \alpha & \text{for } F^T L^{-1}(c_{\hat{X}'} - c_{\hat{Y}'}) < 0,
\end{array} \right.
\]

(7)
where $\alpha \in (0, 1)$ is the significance level of hypothesis test, and $\beta \in [1, 2]$ is an adaptive parameter. Our bi-directional hypothesis is essentially about designing a rejection region of null hypothesis [45]. How to design an efficient rejection region is an interesting problem from the early work [46], and some techniques has been developed for selecting rejection regions [47–49]. We consider the most discriminative directions $F$ and $-F$ in our bi-directional hypothesis, which could improve the sensitivity for two-sample test by selecting appropriate parameters according to different datasets.

Our bi-directional hypothesis can be viewed as a generalization of previous hypotheses, that is,

- By setting $\beta = 1$, our test has been the non-directional hypothesis [9, 11, 50] regardless of direction information, which is also referred to as two-sided/tailed hypothesis [49, 51, 52];
- By setting $\beta = 2$ and $F = (1, 1, \ldots, 1)^T$, our test has become one-directional hypothesis [53–55], which is also referred to as one-sided/tailed hypothesis [52, 56].

Notice that previous non/one-directional hypotheses fix the structures of rejection region for a given significance level $\alpha$, whereas our bi-directional hypothesis could adjust rejection region according to inference direction $F$ w.r.t. different datasets, which can be illustrated in Figure 2. Here, we consider an illustrative dataset, and our bi-directional hypothesis gives the rejection region adaptive to dataset, which could yield higher test power in two-sample test, as shown in Figure 5 (in Section 4).

We can also present some distribution and probability information for $F^\top L^{-1}(c_{\hat{X}}, c_{\hat{Y}}) \geq 0$ in Eqn. (7). Denote by $L_{\hat{X},\hat{Y}} = E_{\hat{X},\hat{Y}}\sim\tilde{\mu}m'\sim\tilde{\nu}n'\sim\tilde{\lambda}m'\sim\tilde{\omega}n'\sim\tilde{\xi}[L']$ and $\xi = \Pr[F^\top L^{-1}(c_{\hat{X}}, c_{\hat{Y}}) \geq 0]$. We have

**Lemma 3.** For inference direction $F$ in Eqn. (6) and for embedding samples $\hat{X}'$ and $\hat{Y}'$, we have

$$L^{-1}(c_{\hat{X}'}, c_{\hat{Y}'}) \sim N(L_{\hat{X},\hat{Y}}^{-1}(\mu_{\hat{X}} - \mu_{\hat{Y}}), \omega^{-1}I) \text{ and } \text{sgn}(F^\top L^{-1}(c_{\hat{X}'}, c_{\hat{Y}'})) \sim TP(\xi)$$

with $\omega = m'n'/((m''+n')$ and $\xi = 1 - \Phi(-\sqrt{\omega/l}F^\top L_{\hat{X},\hat{Y}}^{-1}(\mu_{\hat{X}} - \mu_{\hat{Y}}))$. Here, $\Phi(\cdot)$ is the distribution function of standard Gaussian, and $TP(\xi)$ denotes a distribution over $\{-1, +1\}$ with probability $\xi$ on the selection of $+1$.

The selection of parameter $\beta$ is highly positive-relevant to the probability $\xi$. This is because a larger $\xi$ implies larger difference between two samples in the inference direction $F$, and we should select a larger $\beta$ to enlarge the rejection region and improve the sensitivity of dataset. Figure 6 (in Section 4) shows this positive relevance between the optimal parameter $\beta$ and probability $\xi$ empirically.

We finally present theoretical analysis on test power and type-I error of our bi-directional hypothesis. Let $f(x; n_1, n_2, \lambda)$ be the density function of noncentral $F$-distribution with $n_1$ and $n_2$ degrees of freedom and non-centrality parameter $\lambda$, and denote by $F_{n_1, n_2, \alpha}$ the $\alpha$-quantile of central $F$-distribution with $n_1$ and $n_2$ degrees of freedom for $\alpha \in (0, 1)$. We define the following probability, from the work of [57],

$$q(n_1, n_2, \lambda, \alpha) = \int_{F_{n_1, n_2, \alpha}} \infty f(x; n_1, n_2, \lambda) \, dx .$$

**Theorem 4.** For our bi-directional hypothesis $h(\hat{X}', \hat{Y}')$, the test power can be lower bounded by

$$q(\ell, \nu - \ell, \lambda, \beta\alpha) \cdot \xi + \Phi\left(\frac{\xi^2}{\ell^2(\ell - \beta)}\right) .$$
We call our test as Algorithm 1 Construction of partition tree

**Input**: Two embedding samples \( \bar{X} \) and \( \bar{Y} \)

**Output**: Partition tree \( T \)

**Initialize**: Tree \( T \) with only a root \([0, 1]^\ell\)

1: for \( \ell = 1, \ldots, s - 1 \) do

2: Randomly select one of the leaf node \( B \) of the largest \(|\bar{X} \cap B|\)

3: Initialize the best splitting feature \( j^* = 0 \)

4: for \( j = 1, \ldots, \ell \) do

5: Calculate the median value in \( j \)-th dimension \( \tau_j = \text{median}\{\hat{x}_{i,j} : \hat{x}_i \in \bar{X} \cap B\} \)

6: Split two samples \( \bar{X}_B \) and \( \bar{Y}_B \) into \( \bar{X}_{B_{j}} \), \( \bar{Y}_{B_{j}} \) and \( \bar{X}_{B_{\bar{j}}} \), \( \bar{Y}_{B_{\bar{j}}} \) according to \( \tau_j \)

7: Calculate \( T(\bar{X}_{B_{j}}, \bar{Y}_{B_{j}}) \) and \( T(\bar{X}_{B_{\bar{j}}}, \bar{Y}_{B_{\bar{j}}}) \) based on Eqn. (3)

8: Update \( j^* = j \) if \( \tau_j = \text{median}\{\hat{x}_{i,j} : \hat{x}_i \in \bar{X} \cap B\} \)

9: end for

10: Update node \( B \) with two children \( B_{j^*} \) and \( B_{\bar{j}^*} \) w.r.t. the splitting feature \( j^* \) and position \( \tau_{j^*} \)

11: end for

If \( F^T L_{p,\cdot,Q}^{-1} \mu_{\cdot} \geq F^T L_{p,\cdot,Q}^{-1} \mu_{Q} \), and the test power can also be lower bounded by

\[
q(\ell, \nu - \ell, \lambda, (2 - \beta)\alpha)(1 - \xi) + 1 - \Phi \left( \left( \chi^2_{p} \right)^{1/2} - (\omega/\ell)^{1/2} F^T L_{p,\cdot,Q}^{-1}(\mu_{\cdot} - \mu_{Q}) \right)
\]

If \( F^T L_{p,\cdot,Q}^{-1} \mu_{\cdot} < F^T L_{p,\cdot,Q}^{-1} \mu_{Q} \), and the type-I error rate is equal to \( \alpha \) if \( \mu_{\cdot} = \mu_{Q} \). Here, \( \omega = m'n'/m' + n' \), \( \nu = m' + n' - 1 \) and \( \lambda = \omega L_{p,\cdot,Q}^{-1}(\mu_{\cdot} - \mu_{Q}) \|_2^2 \).

This theorem presents lower bounds on test power, and the performance of the statistical test is maintained under the general condition. Notice that the type-I error in our hypothesis test is controlled only by the significant level \( \alpha \), regardless of different \( \xi \) and \( \beta \).

We call our test as \( ME_{MaBiD} \) test because of multiple Mahalanobis kernels in training and our bi-directional hypothesis in testing.

### 3 Explore Local Significant Differences for Two-sample Test

On the exploration of local significant difference, most previous studies [24–28] partition the instance space into several regions, and then exploit the difference on each region from two samples. Motivated from pòlya tree method [26, 58], we first partition the embedding instance space with a new splitting criterion, which is relevant to test power and data correlations. We then exploit local regions (i.e., leaves nodes of partition tree) with significant difference.

#### Partition of the embedding instance space

Our partition tree is constructed iteratively as follows: We initiate the tree root with embedding space \([0, 1]^\ell\). In each iteration, each node is associated with a rectangle region, and all leaves constitute a partition of embedding instance space. The following procedure is repeated \( s - 1 \) iterations (\( s \geq 2 \)):

- Randomly select a leaf node, denote by \( B \), uniformly over leaf nodes of the largest \(|\bar{X} \cap B|\).
- Let \( \tau_j = \text{median}\{\hat{x}_{i,j} : \hat{x}_i \in \bar{X} \cap B\} \) for \( j \in [\ell] \). We select the best splitting feature

\[
\begin{equation}
\tau_j = \text{arg max}_{j \in [\ell]} \left\{ T(\bar{X}_{B_j}, \bar{Y}_{B_j}) \times T(\bar{X}_{B_{\bar{j}}}, \bar{Y}_{B_{\bar{j}}}) \right\},
\end{equation}
\]

with \( \bar{X}_{B_j} = B_j \cap \bar{X}, \bar{X}_{B_{\bar{j}}} = B_{\bar{j}} \cap \bar{X}, \bar{Y}_{B_j} = B_j \cap \bar{Y} \) and \( \bar{Y}_{B_{\bar{j}}} = B_{\bar{j}} \cap \bar{Y} \). Here, \( B_j \) and \( B_{\bar{j}} \) are left and right children of \( B \) w.r.t. the \( j \)-th splitting feature and splitting position \( \tau_j \), respectively, and \( T(\cdot, \cdot) \) is defined by Eqn. (3).

- Select the splitting position \( \tau_{j^*} = \text{median}\{\hat{x}_{i,j^*} : \hat{x}_i \in \bar{X} \cap B\} \).
We finally get the partition tree with $s$ leaf nodes, associated with $s$ rectangle regions $B_1, B_2, \ldots, B_s$. Algorithm 1 presents the detailed description on tree construction and rectangle region splitting.

We take the statistic $T(\cdot, \cdot)$ as a splitting criterion, relevant to test power and data correlations, and it is helpful to exploit local significant difference directly. We also adopt the median splitting position with equal probabilities on partitioned regions, i.e., balanced examples for each partition region, and this could yield better performance than regular grids, as shown empirically in [59, 60].

Our partition tree is different from previous $p$-value histogram based on Chi-square test [61, 62], where the difference is measured by cardinalities of elements in two samples over a local rectangle region. Our splitting criterion is also different from that of previous decision trees [63–66], which consider some information-theoretic criterions such as entropy, Gini index, information gain, etc. In comparisons, our statistic $T(\cdot, \cdot)$ is more essential to reflect the test power for two-sample test.

For each rectangle region $B_i$, let $c_{\hat{X}_{B_i}}$ and $c_{\hat{Y}_{B_i}}$ be the means of $\hat{X}_{B_i} = B_i \cap \hat{X}$ and $\hat{Y}_{B_i} = B_i \cap \hat{Y}$, respectively. We make similar Schur decomposition $L_{B_i}^{-1} = \Sigma_{\hat{X}_{B_i}, \hat{Y}_{B_i}}$ for covariance matrix $\Sigma_{\hat{X}_{B_i}, \hat{Y}_{B_i}}$, and introduce the local inference direction for each $B_i$ as follow:

$$F_{B_i} = \text{sgn} \left( L_{B_i}^{-1} c_{\hat{X}_{B_i}} - L_{B_i}^{-1} c_{\hat{Y}_{B_i}} \right) \in \{-1, 0, +1\}^\ell. \quad (9)$$

For different rectangle regions, we could have different or even contrary inference directions, which is helpful to exploit local differences from distributional shapes of two samples.

**Exploration of local significant differences**

For testing embedding samples $\hat{X}'$ and $\hat{Y}'$, we denote by $\hat{X}'_{B_i} = B_i \cap \hat{X}'$ and $\hat{Y}'_{B_i} = B_i \cap \hat{Y}'$ with their respective means $c_{\hat{X}'_{B_i}}$ and $c_{\hat{Y}'_{B_i}}$ for each rectangle region $B_i$, and calculate testing statistic $T_{B_i} = T(\hat{X}'_{B_i}, \hat{Y}'_{B_i})$ by Eqn. (4).

We propose the new bi-directional masked $p$-value for each rectangle region $B_i$ as follows:

$$g(\hat{X}'_{B_i}, \hat{Y}'_{B_i}) = \begin{cases} \min \left\{ \frac{\chi_2^2(T_{B_i})}{\beta}, \frac{p_s (1 - \chi_2^2(T_{B_i}))}{1 - \beta p_s} \right\} & \text{for } F_{B_i}^{-1} L_{B_i}^{-1} c_{\hat{X}'_{B_i}} - c_{\hat{Y}'_{B_i}} \geq 0, \\ \min \left\{ \frac{\chi_2^2(T_{B_i})}{2 - \beta}, \frac{p_s (1 - \chi_2^2(T_{B_i}))}{1 - (2 - \beta)p_s} \right\} & \text{for } F_{B_i}^{-1} L_{B_i}^{-1} c_{\hat{X}'_{B_i}} - c_{\hat{Y}'_{B_i}} < 0, \end{cases} \quad (10)$$

\begin{table}
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>dna</td>
<td>3,186</td>
<td>180</td>
<td>kropt</td>
<td>27,705</td>
<td>6</td>
<td>santan</td>
<td>200,000</td>
<td>200</td>
<td>adult</td>
<td>2,000,000</td>
<td>14</td>
</tr>
<tr>
<td>agnos</td>
<td>3,468</td>
<td>970</td>
<td>diamon</td>
<td>53,940</td>
<td>9</td>
<td>codrna</td>
<td>487,867</td>
<td>8</td>
<td>labor</td>
<td>2,000,000</td>
<td>16</td>
</tr>
<tr>
<td>topp21</td>
<td>8,885</td>
<td>266</td>
<td>cifar10</td>
<td>60,000</td>
<td>3072</td>
<td>blob</td>
<td>1,000,000</td>
<td>2</td>
<td>poker</td>
<td>1,025,010</td>
<td>10</td>
</tr>
<tr>
<td>har</td>
<td>10,299</td>
<td>561</td>
<td>mnist</td>
<td>56,000</td>
<td>784</td>
<td>sea50</td>
<td>1,000,000</td>
<td>3</td>
<td>higgs</td>
<td>11,000,000</td>
<td>4</td>
</tr>
</tbody>
</table>

**Table 1: Datasets**

where $\chi_2^2(T_{B_i})$ follows a uniform distribution in such case [68].

Based on such recognition, we resort rectangle regions as $B_{(1)}, B_{(2)}, \ldots, B_{(s)}$ according to their bi-directional masked $p$-value, i.e.,

$$g(\hat{X}'_{B_{(1)}}, \hat{Y}'_{B_{(1)}}) \leq g(\hat{X}'_{B_{(2)}}, \hat{Y}'_{B_{(2)}}) \leq \cdots \leq g(\hat{X}'_{B_{(s)}}, \hat{Y}'_{B_{(s)}}).$$
We compare our ME

We conduct experiments on 16 datasets

We then take our bi-directional hypothesis

Theorem 5.

Table 2: Comparisons of test powers (mean±std) on two-sample test. Bold denotes the highest mean in per row.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Our ME_{MaBiD}</th>
<th>ME</th>
<th>MMDAgg</th>
<th>MMD-D</th>
<th>C2ST-L</th>
<th>C2ST-S</th>
<th>AutoMLTST</th>
</tr>
</thead>
<tbody>
<tr>
<td>blob</td>
<td>.985±.009</td>
<td>.823±.000</td>
<td>.935±.012</td>
<td>.963±.010</td>
<td>.972±.078</td>
<td>.946±.037</td>
<td>.980±.029</td>
</tr>
<tr>
<td>dna</td>
<td>.717±.068</td>
<td>.536±.059</td>
<td>.659±.070</td>
<td>.628±.009</td>
<td>.699±.028</td>
<td>.505±.044</td>
<td>.603±.085</td>
</tr>
<tr>
<td>agnos</td>
<td>.812±.018</td>
<td>.602±.033</td>
<td>.779±.046</td>
<td>.734±.006</td>
<td>.742±.012</td>
<td>.679±.051</td>
<td>.632±.077</td>
</tr>
<tr>
<td>topos21</td>
<td>.692±.006</td>
<td>.526±.058</td>
<td>.605±.077</td>
<td>.633±.062</td>
<td>.679±.046</td>
<td>.517±.046</td>
<td>.591±.006</td>
</tr>
<tr>
<td>har</td>
<td>.858±.065</td>
<td>.816±.015</td>
<td>.814±.026</td>
<td>.728±.064</td>
<td>.761±.093</td>
<td>.738±.063</td>
<td>.740±.058</td>
</tr>
<tr>
<td>kropt</td>
<td>.992±.012</td>
<td>.875±.027</td>
<td>.971±.024</td>
<td>.916±.066</td>
<td>.946±.013</td>
<td>.929±.031</td>
<td>.971±.026</td>
</tr>
<tr>
<td>diamon</td>
<td>.837±.066</td>
<td>.697±.068</td>
<td>.676±.047</td>
<td>.755±.056</td>
<td>.747±.086</td>
<td>.727±.076</td>
<td>.831±.062</td>
</tr>
<tr>
<td>cifar</td>
<td>.893±.022</td>
<td>.859±.075</td>
<td>.866±.091</td>
<td>.878±.090</td>
<td>.834±.099</td>
<td>.798±.019</td>
<td>.882±.086</td>
</tr>
<tr>
<td>mnist</td>
<td>.985±.017</td>
<td>.926±.056</td>
<td>.932±.068</td>
<td>.972±.051</td>
<td>.969±.042</td>
<td>.930±.029</td>
<td>.963±.074</td>
</tr>
<tr>
<td>santan</td>
<td>1.00±.000</td>
<td>.896±.060</td>
<td>1.00±.000</td>
<td>.887±.021</td>
<td>.911±.084</td>
<td>.850±.021</td>
<td>.954±.012</td>
</tr>
<tr>
<td>codma</td>
<td>1.00±.000</td>
<td>.946±.085</td>
<td>.926±.037</td>
<td>.914±.076</td>
<td>1.00±.000</td>
<td>1.00±.000</td>
<td>1.00±.000</td>
</tr>
<tr>
<td>sea50</td>
<td>.993±.018</td>
<td>.993±.018</td>
<td>.982±.012</td>
<td>.993±.018</td>
<td>.993±.018</td>
<td>.970±.053</td>
<td>.989±.029</td>
</tr>
<tr>
<td>adult</td>
<td>.996±.002</td>
<td>.875±.034</td>
<td>.967±.029</td>
<td>.908±.072</td>
<td>.761±.091</td>
<td>.854±.058</td>
<td>.992±.006</td>
</tr>
<tr>
<td>labor</td>
<td>.992±.012</td>
<td>.807±.078</td>
<td>.988±.010</td>
<td>.930±.093</td>
<td>.756±.059</td>
<td>.791±.031</td>
<td>1.00±.000</td>
</tr>
<tr>
<td>poker</td>
<td>.821±.079</td>
<td>.719±.096</td>
<td>.712±.033</td>
<td>.701±.056</td>
<td>.743±.039</td>
<td>.731±.052</td>
<td>.832±.048</td>
</tr>
<tr>
<td>higgs</td>
<td>.979±.024</td>
<td>.818±.090</td>
<td>.938±.047</td>
<td>.953±.055</td>
<td>.968±.043</td>
<td>.933±.013</td>
<td>.969±.030</td>
</tr>
<tr>
<td>Average</td>
<td>.909±.026</td>
<td>.795±.053</td>
<td>.859±.039</td>
<td>.843±.050</td>
<td>.842±.052</td>
<td>.806±.039</td>
<td>.863±.043</td>
</tr>
</tbody>
</table>

We then take our bi-directional hypothesis \( h(\hat{X}^{t}_B, \hat{Y}^{t}_B) \) with parameter \( \beta \) and \( \alpha = p_s \) as in Eqn. (7), and finally get the local regions with significant differences as

\[
\{ B(i) : i \leq t^* \text{ and } h(\hat{X}^{t}_B(i), \hat{Y}^{t}_B(i)) = 1 \},
\]

where

\[
t^* = \arg \max_{t \in [s]} \left\{ t - \{ i \in [t] : h(\hat{X}^{t}_B(i), \hat{Y}^{t}_B(i)) = 1 \} \right\} + 1 \leq \frac{\ln(1 - \alpha_s)}{\ln(1 - p_s)}.
\]

Here, \( p_s \) is selected as in Eqn. (10), and \( \alpha_s \in [p_s, 1) \) is a parameter to control the probability of mis-identifying at least one rectangle region without significant difference, also called familywise error rate [69–72]. We present theoretical analysis for familywise error rate as follows:

**Theorem 5.** For our exploration, the familywise error rate is upper bounded by \( \alpha_s \), if 1) the \( p \)-values of local regions without differences are mutually independent; and 2) the \( p \)-values of local regions with differences are independent to those \( p \)-values of local regions without differences.

Our method is different from previous space partition methods of trees or clusters [24–27, 60], where the splitting criterion is taken as the cardinalities of samples in each region. Duong [28] partitioned and searched local regions from the estimated density function, and Kim et al. [30] identified local regions by clustering data samples from the estimated conditional probabilities. It is not easy to make accurate estimation for density and conditional probabilities without sufficient data, particularly for multiple small regions. Other relevant studies detected local differences implicitly based on interactive rank test [73] or a learned classifier [74].

### 4 Experiments

We conduct experiments on 16 datasets\(^1\) as summarized in Table 1. Most dataset have been studied in previous two-sample test, and features have been scaled to \([0, 1]\) for all datasets. All experiments are performed with Python on nodes of a computational cluster with a single CPU (Intel Core i9-10900X 3.7GHz) and a single GPU (GeForce RTX 2080 Ti), running Ubuntu with 128GB main memory.

**Experimental comparisons for two-sample test**

We compare our ME_{MaBiD} with the state-of-the-art approaches on two-sample test as follows:

\(^1\)Dataset blob is downloaded from [github.com/fengliu90/DK-for-TST](https://github.com/fengliu90/DK-for-TST), and other datasets are downloaded from [www.openml.org](http://www.openml.org).
We compare with the state-of-the-art approaches on exploring local significant difference as follows:

We further compare the average running time (in seconds) for different methods on two-sample test, Appendix C. We take parameter settings for other methods as in their respective inferences.

Table 2 summarizes the average of test powers and standard deviations. It is evident that our ME takes better performance than ME and MMDAgg, because they both take Gaussian kernels with isotropic scale, and ignore the distributional differences from different directions. Our method is still better than MMD-D with a deep kernel, and a reason is that multiple Mahalanobis kernels are more flexible than a deep kernel to capture local difference from multiple neighborhoods and directions.

From Table 2, it is also observed that our ME outperforms three classifier-based methods C2ST-S, C2ST-L and AutoML expect for datasets labor and poker, since those methods focus merely on the prediction information from outputs of classifiers, rather than local and directional information among data samples. For datasets labor and poker, AutoML generates new features automatically from the original mixture of continuous and symbolic features, and thus achieves better performance.

We further compare the average running time (in seconds) for different methods on two-sample test, as shown in Figure 3. As expected, ME takes the least running time since it considers only one Gaussian kernel, yet with the smallest average of test powers in Table 2. Our ME method takes smaller and comparable running time in contrast to other methods since our method takes relatively smaller time on training Mahalanobis kernels without permutation test in the testing process.

Experiments on the exploration of local significant differences

We compare with the state-of-the-art approaches on exploring local significant difference as follows:

- ME: Mean Embeddings over multiple test locations and a single Gaussian kernel [9, 10];
- MMD-D: Maximum Mean Discrepancy based on a Deep kernel [39];
- MMDAgg: Maximum Mean Discrepancy with Aggregating of multiple Gaussian kernels [75];
- C2ST-S: Train a binary classification network and test its accuracy on a hold-out set [13];
- C2ST-L: Train a binary classification network with a statistic about class probabilities [14, 18];
- AutoMLTST: Train a binary classifier based on AutoML method with a statistic as C2ST-L [19].

Following [39, 76], we train on a subset of each available data, and test on 100 random subsets from the remaining dataset, and the ratio is set as 4 : 1 for training and testing. We repeat such process 10 times for each dataset. More details are given in Appendix C. For our ME, we set \( \alpha = 0.05 \) and take 5-fold cross validation to select \( \beta \in [1 : 0.2 : 2] \). We limit the cardinality of test locations within 20 for ME and ME as in [9–11], and optimization parameters of Eqn. (5) is presented in Appendix C. We take parameter settings for other methods as in their respective inferences.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Our method</th>
<th>FDG</th>
<th>KPRIM</th>
<th>MRS</th>
<th>MMDT</th>
<th>BTLDD</th>
<th>TEAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>blob</td>
<td>.945±.082</td>
<td>.902±.075</td>
<td>.879±.045</td>
<td>.849±.075</td>
<td>.909±.091</td>
<td>.932±.046</td>
<td>.877±.067</td>
</tr>
<tr>
<td>diamond</td>
<td>.974±.054</td>
<td>.852±.010</td>
<td>.895±.089</td>
<td>.876±.066</td>
<td>.867±.104</td>
<td>.951±.070</td>
<td>.947±.022</td>
</tr>
<tr>
<td>codrna</td>
<td>.969±.026</td>
<td>.936±.037</td>
<td>.876±.061</td>
<td>.966±.045</td>
<td>.884±.105</td>
<td>.905±.050</td>
<td>.863±.036</td>
</tr>
<tr>
<td>sea50</td>
<td>.977±.056</td>
<td>.975±.062</td>
<td>.933±.074</td>
<td>.928±.058</td>
<td>.985±.045</td>
<td>.982±.059</td>
<td>.944±.065</td>
</tr>
<tr>
<td>adult</td>
<td>.953±.048</td>
<td>.862±.044</td>
<td>.838±.109</td>
<td>.911±.085</td>
<td>.927±.101</td>
<td>.875±.074</td>
<td>.880±.070</td>
</tr>
<tr>
<td>labor</td>
<td>.959±.062</td>
<td>.894±.080</td>
<td>.905±.016</td>
<td>.900±.037</td>
<td>.911±.093</td>
<td>.922±.048</td>
<td>.932±.067</td>
</tr>
<tr>
<td>poker</td>
<td>.945±.030</td>
<td>.901±.030</td>
<td>.882±.027</td>
<td>.925±.023</td>
<td>.927±.057</td>
<td>.894±.028</td>
<td>.884±.064</td>
</tr>
<tr>
<td>higgs</td>
<td>.946±.016</td>
<td>.932±.011</td>
<td>.918±.001</td>
<td>.940±.000</td>
<td>.927±.026</td>
<td>.926±.027</td>
<td>.937±.002</td>
</tr>
<tr>
<td>Average</td>
<td>.959±.047</td>
<td>.907±.044</td>
<td>.891±.053</td>
<td>.912±.049</td>
<td>.917±.078</td>
<td>.912±.050</td>
<td>.908±.049</td>
</tr>
</tbody>
</table>

- MMDT: Partition space into equal grids and test density difference via Welch’s statistic [29].
Datasets
100
101
102
103
Running time (seconds)
Our MEMaBiD ME MMD-D MMDAgg C2ST-S C2ST-L AutoMLTST

Figure 3: Comparisons of running time for different methods on two-sample test. Note that y-axis is in log-scale.

Figure 4: Visualization of the local significant differences with partition tree of \( s = 8 \) (left) and \( s = 16 \) (right) rectangle regions. The bigger the values, the larger the density difference.

For fair comparisons with some methods of density estimations, we select eight large datasets: diamond, codrna, blob, sea50, adult, labor, poker and higgs, whose instance numbers are more than 50,000 and feature numbers are smaller than 20. We randomly select 10,000 instances during the exploration of local significant differences. For the BTLDD method, we set the percentage of selected samples as 6.25\% for local regions in a cluster, while for other methods, we partition the space into 64 rectangle regions, and identify 4 rectangle regions with the most local differences.

We take density differences [78] between two samples in a local region as an evaluation measure for local significant differences, and follow the works of [79, 80] based on \( k \)-NN density estimator with \( k = 20 \). We calculate the difference between two estimated density functions in an identified local region, and normalize the returned values into \([0, 1]\).

Table 3 summarizes the average of density differences and standard deviations. As can be seen, our method takes better explorations on local significant difference than FDG, MRS and TEAM, since those methods simply take cardinalities of samples to measure significant difference. Our exploration is still better than K-PRIM, BTLDD and MMDT except for dataset sea50, because those methods estimate density or conditional probabilities from two samples, and it is not easy to make accurate estimation without sufficient data, especially for small regions with finite samples. MMDT could estimate density function well on dataset sea50, and achieves better exploration via Welch’s statistic.

We further visualize the local significant differences on mnist (sample \( X \)) and Fake-mnist (sample \( Y \)) in Figure 4. Here, we take partition trees with \( s = 8 \) (left) and \( s = 16 \) (right) rectangle regions. As can be seen, our method achieves the largest local differences via new splitting criterion for partition tree, and our proposed ME\textsubscript{MaBiD} test and bi-directional masked \( p \)-values.

Parameter analysis

We now present some parameter influence for our ME\textsubscript{MaBiD} test and the exploration of local significant difference. We only present the results on four datasets due to pages limit, but the trends are similar on other datasets, and more results can be found in Appendix D.
Figure 5: The comparisons of test power vs sample size for our bi-directional hypothesis and previous one/non-directional hypothesis.

Figure 6: The relationship between the optimal adaptive parameter $\beta$ and probability $\xi$ for our $\text{ME}_{\text{MaBiD}}$.

Figure 7: The type-I error is limited about $\alpha = 0.05$ w.r.t different $\beta$ for our $\text{ME}_{\text{MaBiD}}$.

Figure 8: The FWER is limited about $\alpha_s = 0.05$ w.r.t different $s$ for exploring local significant difference.

Figure 5 illustrates the test power versus sample size for our bi-directional hypothesis, non-directional hypothesis and one-directional hypothesis. As can be seen, our bi-directional hypothesis achieves higher test power by considering the inference and its contrary direction and adaptive parameter selection. Figure 6 exploits the relationship between the optimal parameter $\beta$ and the probability $\xi = \Pr [F^\top L^{-1} (c_{X^\top} - c_{Y^\top}) \geq 0]$ for our $\text{ME}_{\text{MaBiD}}$ method. We can easily find the positive relevance between $\beta$ and $\xi$: the larger the parameter $\xi$, the larger the parameter $\beta$.

Figure 7 indicates that the type-I error is limited about $\alpha = 0.05$ for different $\beta$ in our experiments, as shown in Theorem 4, and thus our method could effectively control the rate of falsely reject the null hypothesis, which empirically verify the trustworthiness of our $\text{ME}_{\text{MaBiD}}$ test. Figure 8 empirically shows the familywise error rate is limited about $\alpha_s = 0.05$ for different number of local regions $s$; therefore, our exploring method could control the rate of incorrectly exploiting the local regions with significant difference, and this is nicely in accordance with Theorem 5.

5 Conclusion

This work takes one more step on the exploration of local significant differences. We propose the $\text{ME}_{\text{MaBiD}}$ test by exploiting local information from multiple Mahalanobis kernels and introducing bi-directional hypothesis for testing. We partition embedding space via a new splitting criterion, and then identify local significant differences based on our bi-directional masked $p$-value and $\text{ME}_{\text{MaBiD}}$ test. We verify the effectiveness of our proposed methods both theoretically and empirically. An interesting work is to explore other local and directional information for local significant differences.

Acknowledgments and Disclosure of Funding

The authors want to thank the reviewers for their helpful comments and suggestions. This research was supported by National Key R&D Program of China (2021ZD012802), NSFC (61921006, 62376119) and CAAI-Huawei MindSpore Open Fund. W. Gao is corresponding author of this paper.
References


A Detailed Proofs for Our Theoretical Results

A.1 Proof of Lemma 1

We begin with some useful definitions and lemmas as follows:

Definition 6. (Random Metric [9]). We say that $\rho$ is a random metric with values in $\mathbb{R}$, indexed with pairs from the set of probability measures $\mathcal{M}$, i.e., $\rho = \{\rho(\mathcal{P}, \mathcal{Q}) : \mathcal{P}, \mathcal{Q} \in \mathcal{M}\}$, if it satisfies the conditions for a metric with qualification ‘almost surely’. Formally, for every $\mathcal{P}, \mathcal{Q}, \mathcal{U} \in \mathcal{M}$, random variables $\rho(\mathcal{P}, \mathcal{Q})$, $\rho(\mathcal{P}, \mathcal{U})$ and $\rho(\mathcal{Q}, \mathcal{U})$ satisfy

1. $\rho(\mathcal{P}, \mathcal{Q}) \geq 0$ a.s.
2. if $\mathcal{P} = \mathcal{Q}$, then $\rho(\mathcal{P}, \mathcal{Q}) = 0$ a.s., if $\mathcal{P} \neq \mathcal{Q}$ then $\rho(\mathcal{P}, \mathcal{Q}) \neq 0$ a.s.
3. $\rho(\mathcal{P}, \mathcal{Q}) = \rho(\mathcal{Q}, \mathcal{P})$ a.s.
4. $\rho(\mathcal{P}, \mathcal{Q}) \leq \rho(\mathcal{P}, \mathcal{U}) + \rho(\mathcal{U}, \mathcal{Q})$ a.s.

Lemma 7. [11] Let $\mu$ be an absolutely continuous Lebesgue measure on $\mathbb{R}^d$. A non-zero analytic function $f$ can be zero at most in the set of measure zero w.r.t. $\mu$.

Lemma 8. [9] If $\kappa$ is a bounded and analytic kernel on $\mathbb{R}^d \times \mathbb{R}^d$, then it is analytic for every function in the RKHS associated with this kernel.

Definition 9. For test locations $\mathcal{V} = \{v_1, v_2, \cdots, v_\ell\}$ and Mahalanobis kernels $\kappa_1, \kappa_2, \cdots, \kappa_\ell$ given by Eqn. (1), we define

$$\rho_{\mu_1, \mu_2, \cdots, \mu_\ell}(\mathcal{P}, \mathcal{Q}) = \sum_{j=1}^{\ell} ([\mu_j \mathcal{P}](v_j) - [\mu_j \mathcal{Q}](v_j))^2,$$  \hspace{1cm} (12)

where $[\mu_j \mathcal{P}](v_j) = E_{x \sim \mathcal{P}}[\kappa_j(x, v_j)]$ and $[\mu_j \mathcal{Q}](v_j) = E_{y \sim \mathcal{Q}}[\kappa_j(y, v_j)]$.

We show that the above distance metric for two probability measures is a random metric as follows:

Lemma 10. If $v_1, v_2, \cdots, v_\ell$ are drawn i.i.d. from a absolutely continuous distribution $\mathcal{G}$, then $\rho_{\mu_1, \mu_2, \cdots, \mu_\ell}(\mathcal{P}, \mathcal{Q})$ is a random metric for bounded kernels $\{\kappa_1, \kappa_2, \cdots, \kappa_\ell\}$.

Proof. For $j \in [\ell]$, we first introduce a function

$$\rho_{\mu_j}(\mathcal{P}, \mathcal{Q}) = ([\mu_j \mathcal{P}](v_j) - [\mu_j \mathcal{Q}](v_j))^2,$$

and it is sufficient to prove that $\rho_{\mu_j}(\mathcal{P}, \mathcal{Q})$ is a random metric for each $j \in [\ell]$ from Eqn. (12).

It is well-known that Mahalanobis kernels in Eqn. (1) are characteristic and analytic from [81], and the corresponding mapping

$$\mu_j : \mathcal{P} \to \mu_j \mathcal{P} \quad \text{and} \quad \mu_j : \mathcal{Q} \to \mu_j \mathcal{Q}$$

are injective for $\kappa_j$, where $\mu_j \mathcal{P}$ and $\mu_j \mathcal{Q}$ denote the images of measures $\mathcal{P}$ and $\mathcal{Q}$, respectively. Hence, the image of $\mu_j$ is a subset of analytic functions for analytic and bounded $\kappa_j$, according to Lemma 8.

If $\mathcal{P} = \mathcal{Q}$, then we have

$$f_j = \mu_j \mathcal{P} - \mu_j \mathcal{Q} = 0 \quad \text{and} \quad \rho_{\mu_j}(\mathcal{P}, \mathcal{Q}) = ([\mu_j \mathcal{P}](v_j) - [\mu_j \mathcal{Q}](v_j))^2 = 0.$$

We now prove that if $\mathcal{P} \neq \mathcal{Q}$ then $f_j \neq 0$ almost surely, by applying Lemma 7 to analytic function $f_j = \mu_j \mathcal{P} - \mu_j \mathcal{Q}$ with distribution $\mathcal{G}$. For injective map $\mu_j$, there exists at least one point $a$ such that $f_j(a) \neq 0$, and there exists a ball around $a$ with non-zero $f_j$ from the continuity of $f_j$. Hence, $f$ is almost everywhere nonzero based on Lemma 7, and this follows that

$$\rho_{\mu_j}(\mathcal{P}, \mathcal{Q}) = ([\mu_j \mathcal{P}](v_j) - [\mu_j \mathcal{Q}](v_j))^2 > 0 \quad \text{a.s. for} \quad \mathcal{P} \neq \mathcal{Q}.$$

Hence, $\rho_{\mu_j}$ is random metric from Definition 6 from the symmetry and triangle inequality of $\rho_{\mu_j}$. \hfill $\square$

Proof of Lemma 1. We first have

$$\hat{\mu}_\mathcal{P} = E_{x \sim \mathcal{P}}[\hat{x}] = ([\mu_1 \mathcal{P}](v_1), [\mu_2 \mathcal{P}](v_2), \cdots, [\mu_\ell \mathcal{P}](v_\ell)),$$

$$\hat{\mu}_\mathcal{Q} = E_{y \sim \mathcal{Q}}[\hat{y}] = ([\mu_1 \mathcal{Q}](v_1), [\mu_2 \mathcal{Q}](v_2), \cdots, [\mu_\ell \mathcal{Q}](v_\ell)),$$
We then prove that \( \chi \) We first prove that testing statistic \( T \)

We begin with a useful lemma from [11] as follows.

The matrix \( P \) is symmetric and invertible in the Schur decomposition

This follows that \( \rho \chi \) standard normal distribution. This proves the

From Lemma 10, we can see that \( \rho \mu_1, \ldots, \mu_\ell(\mathbb{P}, \mathbb{Q}) = 0 \) if and only if \( \mathbb{P} = \mathbb{Q} \), and this implies that \( \mu_\ell = \mu_\ell \) if and only if \( \mathbb{P} = \mathbb{Q} \). This completes the proof of Lemma 1.

A.2 Proof of Theorem 2

We begin with a useful lemma from [11] as follows.

Lemma 11. For symmetric and positive definite matrix \( \Sigma \), function \( h(\Sigma) = \Sigma^{-1/2} \) is continuous and well-defined on the positive definite space.

Based on this lemma, we present the detailed proof of Theorem 2 as follows:

Proof of Theorem 2. For embedding testing sample \( \hat{X}' = \{ \hat{x}_1', \ldots, \hat{x}_{m'}' \} \) and \( \hat{Y}' = \{ \hat{y}_1', \ldots, \hat{y}_{n'}' \} \), recall the pooled covariance matrix in Eqn. (3) as

and we also define

We first prove that testing statistic \( T(\hat{X}', \hat{Y}') \) is almost surely asymptotically distributed as \( \chi_\ell^2 \) with \( \ell \) degrees of freedom under the condition \( \mu_\ell = \mu_\ell \) for small \( \epsilon \).

For i.i.d. samples \( \hat{X}' \) and \( \hat{Y}' \), \( \nu_\ell \) is independent to \( c_\nu \). If \( \mu_\ell = \mu_\ell \), then \( \nu = \nu - c_\nu \nu \) follows a multivariate normal distribution with mean \( \mu = \mu_\ell - \mu_\ell = 0 \) and covariance matrix \( (m' + n')\Sigma_\nu_\ell \), from [41, 82] and the Slutsky’s theorem [83], that is

where \( \nu \) denotes convergence in distribution.

The matrix \( L \) is symmetric and invertible in the Schur decomposition \( \Sigma_{\hat{X}', \hat{Y}'} = LL \). Our statistic can be formalized as

and by applying the Slutsky’s theorem, we have

This follows that

where \( (L^{-1} \nu)_j \) is the \( j \)-th dimension value of \( L^{-1} \nu \), and \( \sqrt{m'n'/(m' + n')} \) follows the standard normal distribution. This proves the \( \chi_\ell^2 \) distribution for \( T(\hat{X}', \hat{Y}') \) from the sum of \( \ell \) squares of standard (i.i.d.) normal random variables.

We then prove that \( \chi_\ell^2(T(\hat{X}', \hat{Y}')) \to 0 \) as \( m'n'/(m' + n') \to \infty \) if \( \mu_\ell \neq \mu_\ell \), and hence our test rejects \( H_0 \) almost surely.
From Lemma 11, we have
\[
\lim_{m' n'/ (m' + n') \to \infty} L^{-1} = \lim_{m' n' / (m'+n') \to \infty} \Sigma_{X', Y'}^{-1/2} = \Sigma_{\hat{P}, \hat{Q}}^{-1/2} = L^{-1}_{\hat{P}, \hat{Q}}.
\]
In a similar manner, \( \hat{z} = c_X - c_Y \) converges to \( \mu_{\hat{P}} - \mu_{\hat{Q}} \) in probability. For \( \mu_{\hat{P}} \neq \mu_{\hat{Q}} \), we have
\[
\left\| \Sigma_{\hat{P}, \hat{Q}}^{-1/2} (\mu_{\hat{P}} - \mu_{\hat{Q}}) \right\|_2^2 > 0.
\]
Then, \( \| L^{-1} \hat{z} \|_2^2 \) is a continuous function with entries \( \hat{z} \) and \( L^{-1} \), and it is convergent to some positive constant. We have \( (m' + n') / (m' n') \hat{z} \alpha \to 0 \), and
\[
\Pr \left( \frac{m' n'}{m' + n'} \| L^{-1} \hat{z} \|_2^2 > \frac{m' + n'}{m' n'} \hat{z} \alpha \right) \to 1.
\]
This follows that
\[
\chi_{\hat{P}}^2 \left( \mathcal{T}(\hat{X}', \hat{Y}') \right) = \chi_{\hat{P}}^2 \left( \frac{m' n'}{m' + n'} \| L^{-1} \hat{z} \|_2^2 \right) \to 0,
\]
and hence \( \chi_{\hat{P}}^2 (\mathcal{T}(\hat{X}', \hat{Y}')) \to 0 \) as \( m' n' / (m' + n') \to \infty \) if \( \mu_{\hat{P}} \neq \mu_{\hat{Q}} \). \( \square \)

### A.3 Proof of Lemma 3
Recall the pooled covariance matrix \( \Sigma_{X', Y'} \) and \( \Sigma_{\hat{P}, \hat{Q}} = E_{\hat{X}', \hat{Y}' \sim \hat{P}, \hat{Q}'} \left[ \Sigma_{X', Y'} \right] \) in the proof of Theorem 2, and \( \hat{z} = (c_X - c_Y) \) follows a multivariate normal distribution with mean \( \mu_{\hat{P}} - \mu_{\hat{Q}} \) and covariance matrix \( (m' + n') \Sigma_{\hat{P}, \hat{Q}} / m' n' \). We first observe
\[
L_{\hat{P}, \hat{Q}} = E_{\hat{X}', \hat{Y}' \sim \hat{P}, \hat{Q}'} [L'] = \Sigma_{\hat{P}, \hat{Q}}^{1/2},
\]
and this follows that
\[
L^{-1} \hat{z} \sim \mathcal{N} \left( L^{-1}_{\hat{P}, \hat{Q}} (\mu_{\hat{P}} - \mu_{\hat{Q}}), \frac{m' + n'}{m' n'} I_{\ell} \right).
\]
This is because \( E[L^{-1} \hat{z}] = L^{-1}_{\hat{P}, \hat{Q}} (\mu_{\hat{P}} - \mu_{\hat{Q}}) \) and the covariance matrix is given by
\[
\text{Cov}(L^{-1} \hat{z}, L^{-1} \hat{z}) = L^{-1}_{\hat{P}, \hat{Q}} \left( \frac{m' + n'}{m' n'} \Sigma_{\hat{P}, \hat{Q}} \right) L^{-1}_{\hat{P}, \hat{Q}} = \frac{m' + n'}{m' n'} I_{\ell}.
\]
Write \( B = \sqrt{m' n' / (m' + n')} L^{-1} \hat{z} \), and we have
\[
B^T B = \left( \sqrt{\frac{m' n'}{m' + n'}} L^{-1} \hat{z} \right)^T \left( \sqrt{\frac{m' n'}{m' + n'}} L^{-1} \hat{z} \right)
\]
\[
= \frac{m' n'}{m' + n'} \hat{z}^T \text{diag}(F) (L^{-1})^T L^{-1} \text{diag}(F) \hat{z}
\]
\[
= \frac{m' n'}{m' + n'} \hat{z}^T \text{diag}(F) \Sigma_{\hat{P}, \hat{Q}}^{-1} \text{diag}(F) \hat{z}
\]
\[
= \frac{m' n'}{m' + n'} \hat{z}^T \Sigma_{\hat{P}, \hat{Q}}^{-1} \hat{z}
\]
by using the symmetry of \( \Sigma_{\hat{P}, \hat{Q}}^{-1} \). It is easy to get
\[
E[B] = \sqrt{m' n' / (m' + n')} L^{-1}_{\hat{P}, \hat{Q}} \text{diag}(F) (\mu_{\hat{P}} - \mu_{\hat{Q}})
\]
and covariance matrix is given by
\[
\text{Cov}(B, B) = \sqrt{\frac{m' n'}{m' + n'}} L^{-1}_{\hat{P}, \hat{Q}} \left( \frac{m' + n'}{m' n'} \Sigma_{\hat{P}, \hat{Q}} \right) \sqrt{\frac{m' n'}{m' + n'}} \text{diag}(F) L^{-1}_{\hat{P}, \hat{Q}}
\]
\[
= \frac{m' n'}{m' + n'} \text{diag}(F) L^{-1}_{\hat{P}, \hat{Q}} (m' n' \Sigma_{\hat{P}, \hat{Q}} L^{-1}_{\hat{P}, \hat{Q}}) \sqrt{\frac{m' n'}{m' + n'}} \text{diag}(F) L^{-1}_{\hat{P}, \hat{Q}}
\]
\[
= I_{\ell}.
\]
This yields that

\[ B \sim \mathcal{N} \left( \sqrt{m'n'/(m' + n')} L_{\hat{p}, \hat{q}}^{-1} \text{diag}(F)(\mu_\hat{p} - \mu_\hat{q}), I_\ell \right), \]

and all random variables in \( B \) are mutually independent. Define

\[ \bar{B} = 1^T B/\ell = \sum_{i=1}^\ell B_i/\ell \quad \text{and} \quad S^2 = \sum_{i=1}^\ell (B_i - \bar{B})^2, \]

and \( \bar{B} \) is normally distributed with mean \( \sqrt{m'n'/(m' + n')} F^T L_{\hat{p}, \hat{q}}^{-1}(\mu_\hat{p} - \mu_\hat{q})/\ell \) and variance \( 1/\ell \). It is easy to see that

\[ F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}}) = \sqrt{(m' + n')/m'n'} 1^T B, \]

which yields that

\[ \Pr[\text{sgn}(F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}}))] = 1 = \Pr \left[ \sqrt{\frac{m' + n'}{m'n'}} 1^T B > 0 \right] = \Pr[\bar{B} > 0]. \tag{13} \]

We further have

\[ \Pr[\text{sgn}(F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}}))] = 1 - \Phi \left( -\sqrt{\frac{m'n'}{(m' + n')\ell}} F^T L_{\hat{p}, \hat{q}}^{-1}(\mu_\hat{p} - \mu_\hat{q}) \right), \]

where \( \Phi(\cdot) \) is the cumulative distribution function of standard Gaussian distribution. For continuous normal distribution, we have

\[ \Pr[\text{sgn}(F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}}))] = 0 = \Pr[\bar{B} = 0], \]

and this follows that

\[ \Pr[\text{sgn}(F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}}))] = -1 = \Pr[\bar{B} < 0] = \Phi \left( -\sqrt{\frac{m'n'}{(m' + n')\ell}} F^T L_{\hat{p}, \hat{q}}^{-1}(\mu_\hat{p} - \mu_\hat{q}) \right). \]

Hence, the \( \text{sgn}(F^T L'^{-1}(c_{\tilde{X}}, -c_{\tilde{Y}})) \) follows a two-point distribution \( T\mathcal{P}(\xi) \) with parameter

\[ \xi = 1 - \Phi \left( -\sqrt{\frac{m'n'}{(m' + n')\ell}} F^T L_{\hat{p}, \hat{q}}^{-1}(\mu_\hat{p} - \mu_\hat{q}) \right). \]

This completes the proof. \( \square \)

### A.4 Proof of Theorem 4

Recall the pooled covariance matrix \( \Sigma_{\tilde{X}, \tilde{Y}} \) and \( \Sigma_{\hat{p}, \hat{q}} = E_{\tilde{X}, \tilde{Y}, \tilde{X}', \tilde{Y}'} \Sigma_{\tilde{X}, \tilde{Y}} \) in the proof of Theorem 2, and \( \bar{z} = (c_{\tilde{X}}, -c_{\tilde{Y}}) \) follows a multivariate normal distribution with mean \( \mu_\hat{p} - \mu_\hat{q} \) and covariance matrix \( (m' + n')\Sigma_{\hat{p}, \hat{q}}/m'n' \). We have \( L_{\hat{p}, \hat{q}}^1 L_{\hat{p}, \hat{q}} = \Sigma_{\hat{p}, \hat{q}} \).

We begin with a useful lemma and corollary as follows.

**Lemma 12.** If \( F^T L_{\hat{p}, \hat{q}}^{-1} \mu_\hat{p} > F^T L_{\hat{p}, \hat{q}}^{-1} \mu_\hat{q} \) or null hypothesis \( H_0: \mu_\hat{p} = \mu_\hat{q} \), we have

\[ \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma_{\tilde{X}, \tilde{Y}} \bar{z} \geq c \mid F^T L'^{-1} \bar{z} \geq 0 \right] \geq \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma_{\tilde{X}, \tilde{Y}} \bar{z} \geq c \right]. \tag{14} \]

**Proof.** Recall \( B = \sqrt{m'n'/(m' + n')} L'^{-1} \text{diag}(F) \bar{z} \) and \( F^T L'^{-1} \bar{z} = \sqrt{(m' + n')/m'n'} 1^T B \) in the proof of Lemma 3, and Eqn. (14) is equivalent to

\[ \Pr \left[ B^T \bar{B} \geq c \mid 1^T B \geq 0 \right] \geq \Pr \left[ B^T B \geq c \right]. \tag{15} \]

Recall that

\[ \bar{B} = 1^T B/\ell = \sum_{i=1}^\ell B_i/\ell \quad \text{and} \quad S^2 = \sum_{i=1}^\ell (B_i - \bar{B})^2, \]
and from Eqn. (15), we have
\[
\Pr \left[ \hat{B}^2 \geq (c - S^2)/\ell | \hat{B} \geq 0 \right] \geq \Pr \left[ \hat{B}^2 \geq (c - S^2)/\ell \right].
\]
From the independence of \( S^2 \) and \( \hat{B} \), it is sufficient to prove that, for every \( \delta \geq 0 \),
\[
\Pr \left[ \hat{B}^2 \geq \delta | \hat{B} \geq 0 \right] \geq \Pr \left[ \hat{B}^2 \geq \delta \right]. \tag{16}
\]
It’s easy to see that \( \hat{B} \) is normally distributed with mean \( \sqrt{m'n'/(m' + n')}F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}})/\ell \) and variance \( 1/\ell \). We define
\[
a = \sqrt{\ell \delta} \quad \text{and} \quad b = \sqrt{\frac{m'n'}{(m' + n')\ell}}F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}}),
\]
and from Eqn. (16), we have
\[
\frac{\Phi(-b)}{\Phi(-b - a)} \geq \frac{\Phi(b)}{\Phi(b - a)},
\]
where the equality holds from \( b = 0 \), i.e., \( \mu_{\hat{p}} = \mu_{\hat{Q}} \). This completes the proof. \( \square \)

**Corollary 13.** If \( F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}}) < F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{Q}}) \), we have
\[
\Pr \left[ \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \geq c \mid F^\top L^{-1}z < 0 \right] \geq \Pr \left[ \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \geq c \right],
\]

**Lemma 14.** If \( F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}}) > F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{Q}}) \), then the test power of our bi-directional hypothesis can be lower bounded by
\[
q(\ell, \nu - \ell, \lambda, \beta) : \xi + \Phi \left( -\left(\chi^2_{\ell,(2-\beta)\alpha}\right)^{1/2} - (\omega/\ell)^{1/2} F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}}) \right),
\]
where \( \omega = m'n'/(m' + n') \), \( \nu = m' + n' - 1 \) and \( \lambda = \omega\|L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}})\|_{\mathbb{Z}}^2 \).

**Proof.** From Lemma 12, we first have, for \( F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}}) > F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{Q}}), \)
\[
\Pr[h = 1 \mid F^\top L^{-1}z \geq 0] = \Pr \left[ \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \geq \chi^2_{\ell, \beta\alpha} \mid F^\top L^{-1}z \geq 0 \right] > \Pr \left[ \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \geq \chi^2_{\ell, \beta\alpha} \right], \tag{17}
\]
by substituting \( c = \chi^2_{\ell, \beta\alpha} \) into Eqn. (14). From \( \mu_{\hat{p}} \neq \mu_{\hat{Q}} \), we also have, from the work of [84],
\[
\frac{m' + n' - \ell - 1}{(m' + n' - 2)\ell} T(X', \hat{Y}') = \frac{m' + n' - \ell - 1}{(m' + n' - 2)\ell} \times \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \sim F(\ell, m' + n' - 1 - \ell, \lambda),
\]
where \( \lambda = m'n'(\mu_{\hat{p}} - \mu_{\hat{Q}})^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}})/(m' + n') \). Given the significance level \( \beta\alpha \), we have
\[
\Pr \left[ \frac{m' + n' - \ell - 1}{(m' + n' - 2)\ell} T(X', \hat{Y}') \geq \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \right] = q(\ell, m' + n' - 1 - \ell, \lambda, \beta\alpha) \tag{18}
\]
and this follows that
\[
\Pr \left[ \frac{m'n'}{m' + n'}z^\top \Sigma^{-1}_{X', \hat{Y}, \hat{Z}} \right] = q(\ell, m' + n' - 1 - \ell, \lambda, \beta\alpha). \]
Recall \( \hat{B} = \sqrt{m'n'/(m' + n')}L^{-1} \text{diag}(F)z \) and \( F^\top L^{-1}z = \sqrt{(m' + n')/m'n'} \mathbf{B} \) in the proof of Lemma 3, and that \( \hat{B} = 1^\top \mathbf{B}/\ell \) is normally distributed with mean \( \sqrt{m'n'/(m' + n')}F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}})/\ell \) and variance \( 1/\ell \). We have
\[
\Pr[F^\top L^{-1}z > 0] = \Pr[\hat{B} \geq 0] = 1 - \Phi \left( -\sqrt{\frac{m'n'}{(m' + n')\ell}}F^\top L_{\hat{p}, \hat{Q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{Q}}) \right), \tag{19}
\]
since \( \bar{B} \) is normally distributed with mean \( \sqrt{m'n'/(m' + n')}F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q})/\ell \) and variance 1/\( \ell \). Combining with Eqs. (17)-(19), we have

\[
\text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, \beta \alpha} \right] = \Phi \left( \sqrt{\frac{m'n'}{(m' + n')\ell}} F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \right).
\]

For \( F_{\hat{p} \hat{q}}^{-1} \mu_\bar{p} > F_{\hat{p} \hat{q}}^{-1} \mu_\bar{Q} \), we substitute \( c = \lambda_{\ell, (2 - \beta) \alpha} \) into Eqn. (14), and it holds that

\[
\text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, (2 - \beta) \alpha} \right] = \Phi \left( \sqrt{\frac{m'n'}{(m' + n')\ell}} F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \right).
\]

Combining with Eqs. (20)-(21), we give a lower bound for the test power of bi-directional hypothesis

\[
\text{Pr}[h = 1] = \text{Pr}[h = 1 \cap F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \zeta > 0] + \text{Pr}[h = 1 \cap F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \zeta < 0]
\]

\[
= \text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, \beta \alpha} \right] + \text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, (2 - \beta) \alpha} \right]
\]

\[
> q(\ell, m' + n' - 1 - \ell, \lambda, \beta \alpha) \left( 1 - \Phi \left( \sqrt{\frac{m'n'}{(m' + n')\ell}} F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \right) \right)
\]

\[
+ \Phi \left( \sqrt{\frac{m'n'}{(m' + n')\ell}} F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \right).
\]

\( \square \)

**Lemma 15.** If \( F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \mu_\bar{p} < F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \mu_\bar{Q} \), then the test power of our bi-directional hypothesis can be lower bounded by

\[
q(\ell, \nu - \ell, \lambda, (2 - \beta) \alpha)(1 - \xi) + 1 - \Phi \left( \left( \lambda_{\ell, \beta \alpha} \right)^{1/2} (\omega/\ell)^{1/2} F_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \right),
\]

where \( \omega = m'n'/(m' + n') \), \( \nu = m' + n' - 1 \) and \( \lambda = \omega \| L_{\hat{p} \hat{q}}^{-1}(\mu_\bar{p} - \mu_\bar{Q}) \|^2_2 \).

**Proof.** From Corollary 13, we have, for \( F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \mu_\bar{p} < F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \mu_\bar{Q} \),

\[
\text{Pr}[h = 1 \mid F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \zeta \leq 0] = \text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, \beta \alpha} \mid F_{\hat{p} \hat{q}} L_{\hat{p} \hat{q}}^{-1} \zeta < 0 \right]
\]

\[
> \text{Pr} \left[ \frac{m'n'}{m' + n'} \zeta^\top \Sigma_n^{-1} \zeta' \geq \lambda_{\ell, (2 - \beta) \alpha} \right],
\]

21
Recall that
\[
\Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,(2-\beta)\alpha} \cap F^T L^{-1} \tilde{z} < 0 \right] = q(\ell, m'+n'-1-\ell, \lambda, (2-\beta)\alpha)
\]
in the proof of Lemma 14, and this follows that
\[
\Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,(2-\beta)\alpha} \cap F^T L^{-1} \tilde{z} < 0 \right] = \Pr[\tilde{B}^2 \geq (\chi^2_{\ell,(2-\beta)\alpha} - S^2)/\ell \cap \tilde{B} \geq 0] = \Pr[\tilde{B} \geq \sqrt{(\chi^2_{\ell,(2-\beta)\alpha} - S^2)/\ell}],\]
which is bounded above by
\[
1 - \Phi \left( \sqrt{(\chi^2_{\ell,(2-\beta)\alpha} - S^2)/\ell} \right) + \Phi \left( \sqrt{(\chi^2_{\ell,(2-\beta)\alpha} - S^2)/\ell} \right).
\]
Combining with Eqs. (22)-(23), we give a lower bound for the test power of bi-directional hypothesis
\[
\Pr[h = 1] = \Pr [h = 1 \cap F^T L^{-1} \tilde{z} \geq 0] + \Pr [h = 1 \cap F^T L^{-1} \tilde{z} < 0] = \Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,\beta\alpha} \cap F^T L^{-1} \tilde{z} \geq 0 \right] + \Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,(2-\beta)\alpha} \cap F^T L^{-1} \tilde{z} < 0 \right] > q(\ell, m'+n'-1-\ell, \lambda, (2-\beta)\alpha)\Phi \left( -\sqrt{\frac{m'n'}{(m'+n')\ell}} F^T L^{-1}_{\tilde{p},\tilde{q}} (\mu_{\tilde{p}} - \mu_{\tilde{q}}) \right) + 1 - \Phi \left( \sqrt{(\chi^2_{\ell,\beta\alpha} - S^2)/\ell} \right) \cdot \left( \Phi \left( \sqrt{(\chi^2_{\ell,(2-\beta)\alpha} - S^2)/\ell} \right) \right).
\]
This completes the proof. \(\square\)

**Lemma 16.** For our bi-directional hypothesis, the type-I error rate is equal to \(\alpha\) if \(\mu_{\tilde{p}} = \mu_{\tilde{q}}\).

**Proof.** We first consider the case \(F^T L^{-1} \tilde{z} > 0\). By substituting \(c = \chi^2_{\ell,\beta\alpha}\) into Eqn. (14), we have
\[
\Pr[h = 1 | F^T L^{-1} \tilde{z} \geq 0] = \Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,\beta\alpha} \cap F^T L^{-1} \tilde{z} \geq 0 \right] = \Pr \left[ \frac{m'n'}{m'+n'} z^T \Sigma^{-1}_{\tilde{X}, \tilde{Y}} z \geq \chi^2_{\ell,\beta\alpha} \right] = \beta\alpha.
\]
Recall $B = \sqrt{m'n'/(m' + n')} L^{-1} \text{diag}(F) \bar{z}$ and $F^T L^{-1} \bar{z}$ in the proof of Lemma 3, and that $\bar{B} = 1^T B / \ell$ is normally distributed with mean $\sqrt{m'n'/(m' + n')} F^T L_{\theta, \mathcal{Q}}^{-1} (\mu_{\theta} - \mu_{\mathcal{Q}}) / \ell$ and variance $1/\ell$. We have

$$\Pr[F^T L^{-1} \bar{z} \geq 0] = \Pr[\bar{B} \geq 0] = 1/2,$$

and we have

$$P \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, \alpha} \cap F^T L^{-1} \bar{z} \geq 0 \right]$$

$$= \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, \alpha} | F^T L^{-1} \bar{z} \geq 0 \right] \Pr[F^T L^{-1} \bar{z} \geq 0] = \frac{\beta \alpha}{2}, \quad (24)$$

since $\bar{B}$ is normally distributed with mean 0.

For the case $F^T L^{-1} \bar{z} < 0$, we similarly substitute $c = \chi^2_{l, (2-\beta)\alpha}$ into Eqn. (14), and it follows that

$$\Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \cap F^T L^{-1} \bar{z} < 0 \right]$$

$$= \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \right] - \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \cap F^T L^{-1} \bar{z} \geq 0 \right]$$

$$= \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \right] - \frac{1}{2} \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \right]$$

$$= \frac{1}{2} \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \right] = (1 - \beta/2) \alpha. \quad (25)$$

From Eqns. (24)-(25), we have

$$\Pr[h = 1] = \Pr[h = 1 \cap F^T L^{-1} \bar{z} > 0] + \Pr[h = 1 \cap F^T L^{-1} \bar{z} < 0]$$

$$= \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, \alpha} \cap F^T L^{-1} \bar{z} \geq 0 \right]$$

$$+ \Pr \left[ \frac{m'n'}{m' + n'} \bar{z}^T \Sigma^{-1}_{X', Y', \bar{z}} \geq \chi^2_{l, (2-\beta)\alpha} \cap F^T L^{-1} \bar{z} < 0 \right]$$

$$= \beta \alpha + (1 - \beta/2) \alpha = \alpha.$$

This completes the proof. \hfill \Box

Theorem 4 follows from Lemmas 14-16.

### A.5 Proof of Theorem 5

Recall that $B_{(1)}, B_{(2)}, \ldots, B_{(s)}$ are rectangle regions of a non-increasing order w.r.t. $g(\cdot, \cdot)$. For each rectangle region $B_{(i)}$, we could define its local null hypothesis

$$H_{0_{(i)}}: \mu_{B_{(i)}} = \mu_{Q_{B_{(i)}}} \quad \text{with} \quad \mu_{B_{(i)}} = E_{\mathcal{Z}_{B_{(i)}}} [\mathcal{Z}] \quad \text{and} \quad \mu_{Q_{B_{(i)}}} = E_{\mathcal{Z}_{\mathcal{Q}_{B_{(i)}}}} [\mathcal{Z}].$$

From Theorem 2, the testing statistic $T(X_{B_{(i)}}, Y_{B_{(i)}})$ follows the $\chi^2$ distribution with $\ell$ degrees under the local null hypothesis $H_{0_{(i)}}$. Denote by $\chi^2_{\ell}(T(X_{B_{(i)}}, Y_{B_{(i)}}))$ the p-value, and we have
We also define the \( \chi^2 \) distribution \( \mu_{\hat{B}_{(i)}} = \mu_{\tilde{B}_{(i)}} \).

We take an interactive multi-step testing procedure to identify the index set \( \text{rectangle regions of local significant differences} \). Define the candidate rejection set \( \mathcal{R}(t) = \{ \mathcal{H}_{0,(i)} \}_{i=1}^s \) for \( 1 \leq t \leq s \) with \( \mathcal{R}(0) = \emptyset \), and exclude one null hypothesis \( \mathcal{H}_{0,(i)} \) at the \( t \)-th step. We could generate a sequence as follows:

\[
\{ \mathcal{H}_{0,(i)} \}_{i=1}^s = \mathcal{R}(s) \supseteq \mathcal{R}(s-1) \supseteq \mathcal{R}(s-2) \supseteq \cdots \supseteq \mathcal{R}(0) = \emptyset ,
\]

and it holds that \( \mathcal{H}_{0,(i)} = \mathcal{R}(i) \setminus \mathcal{R}(i-1) \). Recall the local bi-directional hypothesis

\[
h_i = h(\hat{X}_{(i)}', \hat{X}_{(i)}) \quad \text{for} \quad i \in [s].
\]

Denote by \( p_* \) the parameter of significant level for the local two-sample test and masked \( p \)-value. From Theorem 4, we have \( \Pr[h_i = 1] = p_* \) under the null local hypothesis \( \mathcal{H}_{0,(i)} \). We further present some useful lemmas as follows.

**Lemma 17.** [68] The \( p \)-value \( \chi^2 \left( T(\tilde{X}_{(i)}', \tilde{Y}_{(i)}') \right) \) follows a uniform distribution \( U[0, 1] \) under the local null hypothesis \( \mathcal{H}_{0,(i)} ; \mu_{\hat{B}_{(i)}} = \mu_{\tilde{B}_{(i)}} \).

**Lemma 18.** [67] We have \( E[h_i] = p_* \) for \( i \in [s] \) and \( h_1, h_2, \ldots, h_s \) are mutually independent, if \( \mu_{\hat{B}_{(i)}} = \mu_{\tilde{B}_{(i)}} \) for every \( i \in [s] \) and the \( p \)-values are uniformly distributed.

Denote by \( \tilde{B} \) the set of rectangle regions that the local two samples \( \hat{X}_{(i)}' \) and \( \hat{Y}_{(i)}' \) are actually drawn from one identical distribution, and we define

\[
\mathcal{H}_0 = \{ \mathcal{H}_{0,(i)} : B_{(i)} \in \tilde{B} \}.
\]

**Lemma 19.** [67] If there is some rectangle region such that \( \mu_{\hat{B}_{(i)}} \neq \mu_{\tilde{B}_{(i)}} \), and if the \( p \)-value follows a uniform distribution under null hypothesis \( \mathcal{H}_{0,(i)} \), then we have

\[
E \left[ h_i \mid \{ h_k \}_{k=i+1}^s , \{ \mathbb{1}(\mathcal{H}_{0,(k)} \in \mathcal{H}_0) \}_{k=i+1}^s , \mathcal{H}_{0,(i)} \in \mathcal{H}_0 \right] = p_* \quad \text{for} \quad i \in [s].
\]

We also define the weighted mirror-conservativeness, motivated from [85, 67], as follows.

**Definition 20.** We say that a density function \( f(\cdot) \) satisfies the weighted mirror-conservativeness if it holds that, for some given \( p \in (0, 1) \),

\[
f(aw) \leq f(1 - (1 - wp)/p) \quad \text{for} \quad w \in [1, 2], \quad a \in [0, p].
\]

Here, we introduce an additional parameter \( w \) to incorporate two directions for our method, which is different from the previous mirror-conservativeness [85, 67]. We could also present two sufficient conditions for weighted mirror-conservativeness from [85, 67]: i) the non-decreease of \( f \) and ii) the convexity of cumulative density function of \( p \)-value.

**Lemma 21.** If the density function of the \( p \)-value satisfies the weighted mirror-conservativeness under local null hypothesis \( \mathcal{H}_{0,(i)} \), and if there is some rectangle region such that \( \mu_{\hat{B}_{(i)}} \neq \mu_{\tilde{B}_{(i)}} \), then we have, for every \( i \in [s] \),

\[
E \left[ h_i \mid \{ h_k \}_{k=i+1}^s , \{ \mathbb{1}(h_k \in \mathcal{H}_0) \}_{k=i+1}^s , \mathcal{H}_{0,(i)} \in \mathcal{H}_0 , \{ g_k \}_{k=1}^s \right] \leq p_* ,
\]

where \( g_k = g(\hat{X}_{(k)}, \hat{Y}_{(k)}) \).

**Proof.** We first prove

\[
E[h_i \mid g_i = a] \leq p_* \quad \text{for} \quad \mathcal{H}_{0,(i)} \in \mathcal{H}_0 ,
\]

from our bi-directional hypothesis and masked \( p \)-value with \( \beta \in [1, 2] \) in Eqn. (7), and it is sufficient to consider the following two cases:
We further have, since $\frac{\beta a}{(1 - p_*) f(1 - \frac{1 - \beta p_*}{p_*})} \geq 0$, we have
\[
E \left[ h(i) \mid g(i) = a, F_{\mathcal{B}(i)}^T L_{\mathcal{B}(i)}^{-1} (c_{\hat{X}'_\mathcal{B}(i)} - c_{\hat{Y}'_\mathcal{B}(i)}) > 0 \right] 
\]
\[
= \frac{p_* f(\beta a)}{p_* f(\beta a) + (1 - p_*) f(1 - \frac{1 - \beta p_*}{p_*})} 
\]
\[
= \frac{p_*}{1 + \frac{(1 - p_*) f(1 - \frac{1 - \beta p_*}{p_*})}{f(\beta a)}} 
\]
\[
\leq p_* ,
\]
where the last inequality holds from the weighted mirror-conservativeness with $p$-value’s uniform distribution $U(0, 1]$.

- For $F_{\mathcal{B}(i)}^T L_{\mathcal{B}(i)}^{-1} (c_{\hat{X}'_\mathcal{B}(i)} - c_{\hat{Y}'_\mathcal{B}(i)}) < 0$, we similarly have
\[
E \left[ h(i) \mid g(i) = a, F_{\mathcal{B}(i)}^T L_{\mathcal{B}(i)}^{-1} (c_{\hat{X}'_\mathcal{B}(i)} - c_{\hat{Y}'_\mathcal{B}(i)}) < 0 \right] 
\]
\[
= \frac{p_* f((2 - \beta)a)}{p_* f((2 - \beta)a) + (1 - p_*) f(1 - \frac{1 - (2 - \beta)p_*}{p_*})} 
\]
\[
= \frac{p_*}{1 + \frac{(1 - p_*) f(1 - \frac{1 - (2 - \beta)p_*}{p_*})}{f((2 - \beta)a)}} 
\]
\[
\leq p_* .
\]

We define the information available for choosing $H_{0,(i)}$ as a filtration (sequence of nested $\sigma$-fields)
\[
\mathcal{F}_{(i)} = \sigma \left( \{B(k), g(k)\}_{k=1}^s, \{\chi^2(T(\hat{X}'_{\mathcal{B}(k)}, \hat{Y}'_{\mathcal{B}(k)}))\}_{k=1}^s \right) ,
\]
and also define the filtration
\[
\mathcal{F}^h_{(i)} = \sigma \left( \{h(k)\}_{k=1}^s, \{I(H_{0,(k)} \in \mathcal{H}_0)\}_{k=1}^s \right) .
\]
This follows that
\[
E \left[ h(i) \mid \{h(k)\}_{k=1}^s, \{I(H_{0,(k)} \in \mathcal{H}_0)\}_{k=1}^s, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= E \left[ h(i) \mid \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= E \left[ E \left[ h(i) \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] \mid \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
where the last equality holds from the law of total expectation. We have
\[
E \left[ h(i) \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= \sum_{H_{0,j} \in \mathcal{R}(i) \cap \mathcal{H}_0} E[h(i) \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s] \times Pr \left[ H_{0,(i)} = H_{0,j} \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= \sum_{H_{0,j} \in \mathcal{R}(i) \cap \mathcal{H}_0} E \left[ h(i) \mid \mathcal{F}_{(i)} \right] Pr \left[ H_{0,(i)} = H_{0,j} \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] ,
\]
where the last equation holds from the fact that $\{\mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s\}$ is a subset of $\mathcal{F}_{(i)}$.

We further have, since $h(i)$ is independent of other information in $\mathcal{F}_{(i)}$,
\[
E \left[ h(i) \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= \sum_{H_{0,j} \in \mathcal{R}(i) \cap \mathcal{H}_0} E[h(i) \mid g(j)] Pr \left[ H_{0,(i)} = H_{0,j} \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
\leq p_* \sum_{H_{0,j} \in \mathcal{R}(i) \cap \mathcal{H}_0} Pr \left[ H_{0,(i)} = H_{0,j} \mid \mathcal{F}_{(i)}, \mathcal{F}^h_{(i)}, H_{0,(i)} \in \mathcal{H}_0, \{g(k)\}_{k=1}^s \right] 
\]
\[
= p_* ,
\]
We further introduce a weighted version of Lemma 23 as follows:

\[ \Pr[Z = 1] = p \quad \text{and} \quad \Pr[Z = 0] = 1 - p. \]

We also say that a random variable \( Z \) follows a negative binomial distribution with parameters \( r \) and \( p \), denoted by \( Z \sim NB(r, p) \), if

\[ \Pr[Z = k] = \binom{k + r - 1}{k} (1 - p)^r p^k. \]

It is necessary to introduce a definition as follows:

**Definition 22.** We say that a random variable \( Z \) is stochastically dominated by a distribution \( G \), denoted by

\[ Z \preceq G, \]

if for random variable \( X \sim G \), it holds that

\[ \Pr[Z \geq x] \leq \Pr[X \geq x] \quad \text{for} \quad x \in (-\infty, +\infty). \]

We further introduce some useful lemmas as follows:

**Lemma 23.** [67] Let \( Z_1, \ldots, Z_s \) be i.i.d random variables with \( Z_i \sim Bern(p_s) \) for some \( p_s > 0 \), and write \( \bar{N}_t = \sum_{j=1}^t Z_j \) and \( \bar{G}_t = \sigma(\bar{N}_t, \{Z_j\}_{j=t+1}^s) \) for \( t \in [s] \). We have

\[ \bar{N}_t \preceq NB(v, p_s), \]

where the stopping index \( \bar{\tau} \) is parameterized by some constant \( v(\geq 1) \), defined by

\[ \bar{\tau} = \max \left\{ 0 < t \leq s : t - \bar{N}_t < v \text{ or } t = 1 \right\}. \]

We further introduce a weighted version of Lemma 23 as follows:

**Lemma 24.** [67] Let \( \{W_j\}_{j=1}^s \) be a sequence of weights, drawn from a Bernoulli distribution, s.t. \( \sum_{j=1}^s W_j = u \) for fixed constant \( u \leq s \); and \( Z_j \mid \sigma(\{Z_k, W_k\}_{k=j+1}^s, W_j = 1) \sim Bern(p_s) \). Write \( N^w_t = \sum_{j=1}^t W_j Z_j \), and we have

\[ N^w_t \preceq NB(v, p_s), \]

where the stopping index \( \tau^w \) is parameterized by some constant \( v(\geq 1) \), defined by

\[ \tau^w = \max \left\{ 0 < t \leq s : \sum_{j=1}^t W_j - N^w_t < v \text{ or } t = 1 \right\}. \]

We now introduce a different version of Lemma 24 by considering different parameter for Bernoulli distribution as follows:

**Lemma 25.** [67] Let \( Z_j \mid \sigma(\{Z_k, W_k\}_{k=j+1}^s, W_j = 1) \) follow a Bernoulli distribution with parameter \( p(\{Z_k, W_k\}_{k=j+1}^s) \) for \( j \in [s] \), respectively. We have

\[ N^w_t \preceq NB(v, p(\{Z_k, W_k\}_{k=j+1}^s)), \]

\[ N^w_t \preceq NB(v, p_s), \]

if \( p(\{Z_k, W_k\}_{k=j+1}^s) \leq p_s \) for every \( j \in [s] \).

We now present the detailed proof of Theorem 5 as follows.

**Proof of Theorem 5.** We first consider \( \mu_{\bar{\tau}^\text{EB}(i)} = \mu_{\bar{\tau}^\text{EB}(i)} \) for every \( i \in [s] \). From Lemma 18, \( \{h(i)\}_{i=1}^s \) are s i.i.d. random variables with \( h(i) \sim Bern(p_s) \). Recall that the stopping rule in our testing, i.e., Eqn. (11), which is equivalent to

\[ 1 - (1 - p_s)^{t - |Z(i)| + 1} \leq \alpha_s, \]
where \( \alpha_s \) is a parameter to control familywise error rate and \( \mathcal{I}(t) = \{ i \in [t] : h(\hat{X}_{B_i}, \hat{Y}_{B_i}) = 1 \} \).

The stopping rule can be rewritten as
\[
v = \frac{[\ln(1 - \alpha_s)/\ln(1 - p_s)]}{0 < t \leq s : t - \tilde{N}_i < v \text{ or } t = 1}.
\]  

(29)

Let \( Z_j = h_{(j)} \) and \( \tilde{N}_i = \sum_{j=1}^{t} Z_j \). We define the stopping index \( \tilde{\tau} \) as follows
\[
\tilde{\tau} = \max \left\{ 0 < t \leq s : t - \tilde{N}_i < v \text{ or } t = 1 \right\}.
\]

From Lemma 23, the number of rejections at the stopping index is given by
\[
|\mathcal{I}(\tilde{\tau})| = \sum_{j=1}^{\tilde{\tau}} h_{(j)} = \tilde{N}_{\tilde{\tau}} \leq \mathcal{NB}(v, p_s).
\]

If \( \mu_{\hat{X}_{B_i}} = \mu_{\hat{X}_{B_i}} \) for every \( i \in [s] \), then the number of false rejections is
\[
|\mathcal{I}(\tilde{\tau}) \cap \mathcal{H}_0| = |\mathcal{I}(\tilde{\tau})| \leq \mathcal{NB}(v, p_s),
\]
and hence the familywise error rate (FWER) is upper bounded by
\[
\Pr[|\mathcal{I}(\tilde{\tau}) \cap \mathcal{H}_0| \geq 1] \leq 1 - (1 - p_s)^v \leq \alpha_s,
\]
where the last inequality follows from Eqn. (29).

We now consider that there is some rectangle region with \( \mu_{\hat{X}_{B_i}} \neq \mu_{\hat{X}_{B_i}} \). In such case, we provide an upper bound for familywise error rate without the information of masked \( p \)-values, and prove that the number of false rejections is stochastically dominated by \( \mathcal{NB}(v, p_s) \).

Let \( Z_j = h_{(j)} \) and \( W_j = \mathbb{I}(H_{0,(j)} \in \mathcal{H}_0) \). We define the stopping index \( \tau_w \) as follows:
\[
\tau_w = \max \left\{ 0 < t \leq s : \sum_{j=1}^{t} \mathbb{I}[h_{(j)} = 0 \cap H_{0,(j)} \in \mathcal{H}_0] = \sum_{j=1}^{t} W_j (1 - Z_j) < v \text{ or } t = 1 \right\},
\]
where \( v \) is given in Eqn. (29). It is easy to see that
\[
Z_j | \sigma(\{Z_k, W_k\}_{k=j+1}, W_j = 1) \sim \text{Bern}(p_s)
\]
from Lemma 19. Denote by \( u = |\mathcal{H}_0| \), and we have \( \sum_{j=1}^{t} W_j = u \) and \( u \leq s \). From Lemma 24, we have the number of false rejections
\[
\sum_{j=1}^{\tau_w} \mathbb{I}[h_{(j)} = 1 \cap H_{0,(j)} \in \mathcal{H}_0] = \sum_{j=1}^{\tau_w} W_j Z_j = N_{\tau_w} \leq \mathcal{NB}(v, p_s) \quad (30).
\]

Recall that \( t - |\mathcal{I}(t)| < v \) is our stopping rule on the exploration of local significant differences. Denote by \( \tau_{\tau_w}^t \) the stopping index in our exploration, and we have
\[
\sum_{j=1}^{\tau_{\tau_w}^t} \mathbb{I}[h_{(j)} = 0 \cap H_{0,(j)} \in \mathcal{H}_0] \leq \sum_{j=1}^{\tau_{\tau_w}^t} \mathbb{I}[h_{(j)} = 0] = \tau_{\tau_w}^t - \sum_{j=1}^{\tau_{\tau_w}^t} \mathbb{I}[h_{(j)} = 1] = \tau_{\tau_w}^t - \mathcal{I}(\tau_w) < v.
\]

Since \( N_{\tau_w}^w \) is non-decreasing with respect to \( t \), it is easy to obtain
\[
\tau_{\tau_w}^t \leq \tau_w \quad \text{and} \quad N_{\tau_w}^w \leq N_{\tau_w}^w,
\]
and we have the number of false rejections
\[
|\mathcal{I}(\tau_{\tau_w}^t) \cap \mathcal{H}_0| = \sum_{j=1}^{\tau_{\tau_w}^t} \mathbb{I}[h_{(j)} = 1 \cap H_{0,(j)} \in \mathcal{H}_0] = N_{\tau_w}^w \leq N_{\tau_w}^w \leq \mathcal{NB}(v, p_s) \quad (31).
\]

27
We finally upper bound the familywise error rate without considering the masked $p$-values as follows:

$$
\Pr \left[ |\mathcal{I}(\tau^w) \cap \mathcal{H}_0| \geq 1 \right] \leq \Pr \left[ |\mathcal{I}(\tau^w) \cap \mathcal{H}_0| \geq 1 \right] \leq 1 - (1 - p_*)^{v} \leq \alpha_* .
$$

We finally take masked $p$-values $\{g(k)\}_{k=1}^{s}$ into consideration. From Lemma 21, it is easy to observe

$$
Z_j | \sigma (\{Z_k, W_k\}_{k=j+1}, W_j = 1) \sim \text{Bern}(p(\{Z_k, W_k\}_{k=j+1})),
$$

where $p(\{Z_k, W_k\}_{k=j+1}) = E \left[ \{h(k)\}_{k=j+1}, \{I(\mathcal{H}(k) \in \mathcal{H}_0)\}_{k=j+1}, \mathcal{H}_0, (g(k))_{k=1}^{s} \right] \leq p_* .
$$

This follows that

$$
\mathcal{N}(v, p (\{Z_k, W_k\}_{k=j+1})) \leq \mathcal{N}(v, p_*),
$$

and we further have, from Lemma 25,

$$
|\mathcal{I}(\tau^w) \cap \mathcal{H}_0| = \sum_{j=1}^{v} W_j Z_j = N^{\text{temp}} \leq \mathcal{N}(v, p (\{Z_k, W_k\}_{k=j+1}))) \leq \mathcal{N}(v, p_*).
$$

We finally upper bound the familywise error rate by considering the masked $p$-values $\{g(k)\}_{k=1}^{s}$ as

$$
\Pr \left[ |\mathcal{I}(\tau^w) \cap \mathcal{H}_0| \geq 1 \mid \{g(k)\}_{k=1}^{s} \right] \leq \mathcal{P} \left( |\mathcal{I}(\tau^w) \cap \mathcal{H}_0| \geq 1 \mid \{g(k)\}_{k=1}^{s} \right) \\
\leq 1 - (1 - p_*)^{v} \leq \alpha_* .
$$

This completes the proof. 

\[\square\]

### B Optimization for Test Locations and Mahalanobis Kernels

We take gradient method [43] for the optimization of Eqn. (5) as in the work of [10]. Specifically, we calculate gradients, and update test locations and Mahalanobis kernels iteratively. In the following of this section, we present the calculation of some crucial gradients in optimization.

For test location $v_j$ with $j \in [\ell]$, we have

$$
\nabla_{v_j} T(\hat{X}, \hat{Y}) = \left( \frac{\partial T(\hat{X}, \hat{Y})}{\partial v_{j,1}}, \frac{\partial T(\hat{X}, \hat{Y})}{\partial v_{j,2}}, \ldots, \frac{\partial T(\hat{X}, \hat{Y})}{\partial v_{j,\ell}} \right)^\top ,
$$

(32)

where, for $i \in [\ell]$,

$$
\frac{\partial T(\hat{X}, \hat{Y})}{\partial v_{j,i}} = \frac{\partial T(\hat{X}, \hat{Y})}{\partial c_{\hat{X}}} \frac{\partial c_{\hat{X}}}{\partial v_{j,i}} + \frac{\partial T(\hat{X}, \hat{Y})}{\partial c_{\hat{Y}}} \frac{\partial c_{\hat{Y}}}{\partial v_{j,i}} + \text{Tr} \left[ \frac{\partial T(\hat{X}, \hat{Y})}{\partial \Sigma_{\hat{X}, \hat{Y}}} \frac{\partial \Sigma_{\hat{X}, \hat{Y}}}{\partial v_{j,i}} \right],
$$

(33)

where $\text{Tr}[\cdot]$ denotes the trace.

We further have

$$
\frac{\partial T(\hat{X}, \hat{Y})}{\partial c_{\hat{X}}} = 2mn\Sigma_{\hat{X}, \hat{Y}}^{-1}(e_{\hat{X}} - c_{\hat{Y}})/(m + n) ,
$$

$$
\frac{\partial c_{\hat{X}}}{\partial v_{j,i}} = \frac{1}{m} \sum_{r=1}^{m} \frac{\partial \hat{x}_r}{\partial v_{j,i}} \text{ with } \frac{\partial \hat{x}_r}{\partial v_{j,i}} = \left( 0, \ldots, 0, \frac{\partial \hat{x}_r}{\partial v_{j,i}}, 0, \ldots, 0 \right)^\top ,
$$

where all elements are zeros except for the $j$-th element. We also have

$$
\frac{\partial \kappa_j(x_r, v_j)}{\partial v_{j,i}} = \frac{\partial}{\partial v_{j,i}} \left\{ \exp \left\{ -(x_r - v_j)^\top M_j(x_r - v_j)/2 \gamma_j^2 \right\} \right\}
$$

$$
= \kappa_j(\hat{x}_r - v_j)(M_j(\hat{x}_r - v_j))^\top (0, \ldots, 1, \ldots, 0)^\top / \gamma_j^2 ,
$$

where the $i$-th element is 1. We similarly calculate $\partial T(\hat{X}, \hat{Y})/\partial c_{\hat{Y}} \times \partial c_{\hat{Y}}/\partial v_{j,i}$. 

28
For the third term in Eqn. (33), we have

\[
\frac{\partial T(\hat{X}, \hat{Y})}{\partial \Sigma_{\hat{X}, \hat{Y}}} = \Sigma_{\hat{X}, \hat{Y}}^{-1}(e_{\hat{X}} - c_{\hat{Y}})(e_{\hat{X}} - c_{\hat{Y}})^\top \Sigma_{\hat{X}, \hat{Y}}^{-1},
\]

\[
\frac{\partial \Sigma_{\hat{X}, \hat{Y}}}{\partial v_{j,i}} = \frac{1}{m + n - 2} \left( \sum_{r=1}^{m} \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top}{\partial v_{j,i}} + \sum_{r=1}^{n} \frac{\partial (\hat{y}_r - c_{\hat{Y}})(\hat{y}_r - c_{\hat{Y}})^\top}{\partial v_{j,i}} \right),
\]

where

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top}{\partial v_{j,i}} = \begin{pmatrix}
0 & \cdots & \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top}{\partial v_{j,i}} \\
\vdots & \ddots & \vdots \\
0 & \cdots & 0
\end{pmatrix}.
\]

Here, \((\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top_{j,t}\) denotes the element in the \(j\)-th row and \(t\)-th column, and we have

\[
(\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top_{j,t} = \left( m - \frac{1}{m} \kappa_t(\hat{x}_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_t(\hat{x}_s, v_j) \right) \left( m - \frac{1}{m} \frac{\partial \kappa_t(\hat{x}_r, v_j)}{\partial v_{j,i}} - \frac{1}{m} \sum_{s \neq r} \frac{\partial \kappa_t(\hat{x}_s, v_j)}{\partial v_{j,i}} \right),
\]

and this follows that

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top_{j,t}}{\partial v_{j,i}} = \left( m - \frac{1}{m} \kappa_t(\hat{x}_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_t(\hat{x}_s, v_j) \right) \left( m - \frac{1}{m} \frac{\partial \kappa_t(\hat{x}_r, v_j)}{\partial v_{j,i}} - \frac{1}{m} \sum_{s \neq r} \frac{\partial \kappa_t(\hat{x}_s, v_j)}{\partial v_{j,i}} \right).
\]

We similarly have

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^\top_{j,i}}{\partial v_{j,i}} = 2 \left( m - \frac{1}{m} \kappa_j(\hat{x}_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_j(\hat{x}_s, v_j) \right) \left( m - \frac{1}{m} \frac{\partial \kappa_j(\hat{x}_r, v_j)}{\partial v_{j,i}} - \frac{1}{m} \sum_{s \neq r} \frac{\partial \kappa_j(\hat{x}_s, v_j)}{\partial v_{j,i}} \right).
\]

For gamma parameter \(\gamma_j\) with \(j \in [\ell]\), we have

\[
\nabla_{\gamma_j} T(\hat{X}, \hat{Y}) = \frac{\partial T(\hat{X}, \hat{Y})}{\partial c_{\hat{X}}} \frac{\partial c_{\hat{X}}}{\partial \gamma_j} + \frac{\partial T(\hat{X}, \hat{Y})}{\partial c_{\hat{Y}}} \frac{\partial c_{\hat{Y}}}{\partial \gamma_j} + \text{Tr} \left[ \frac{\partial T(\hat{X}, \hat{Y})}{\partial \Sigma_{\hat{X}, \hat{Y}}} \frac{\partial \Sigma_{\hat{X}, \hat{Y}}}{\partial \gamma_j} \right].
\]

(34)

We further have

\[
\frac{\partial c_{\hat{X}}}{\partial \gamma_j} = \frac{1}{m} \sum_{r=1}^{m} \frac{\partial \hat{x}_r}{\partial \gamma_j} \quad \text{with} \quad \frac{\partial \hat{x}_r}{\partial \gamma_j} = \left( 0, \ldots, \frac{\partial \kappa_j(\hat{x}_r, v_j)}{\partial \gamma_j}, \ldots, 0 \right)^\top,
\]

where

\[
\frac{\partial \kappa_j(\hat{x}_r, v_j)}{\partial \gamma_j} = \frac{\partial \exp \left( -(\hat{x}_r - v_j)^\top M_j(\hat{x}_r - v_j)/2\gamma_j^2 \right)}{\partial \gamma_j}
\]

\[
= \kappa_j(\hat{x}_r, v_j)(\hat{x}_r - v_j)^\top M_j(\hat{x}_r - v_j)\gamma_j^{-3}.
\]
We similarly calculate

\[
\frac{\partial \Sigma_{\hat{X}, \hat{Y}}}{\partial \gamma_j} = \frac{1}{m + n - 2} \left( \sum_{r=1}^{m} \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} + \sum_{r=1}^{n} \frac{\partial (\hat{y}_r - c_{\hat{Y}})(\hat{y}_r - c_{\hat{Y}})^T}{\partial \gamma_j} \right),
\]

where

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} = \begin{pmatrix}
0 & \cdots & \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} & \cdots & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} & \cdots & \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} & \cdots & \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & \frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} & \cdots & 0
\end{pmatrix},
\]

and this follows that

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} = \begin{pmatrix}
\frac{m - 1}{m} \kappa_j(x_r, v_t) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_t) \\
\vdots \\
\frac{m - 1}{m} \kappa_j(x_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_j) \end{pmatrix}.\]

We similarly have

\[
\frac{\partial (\hat{x}_r - c_{\hat{X}})(\hat{x}_r - c_{\hat{X}})^T}{\partial \gamma_j} = 2 \begin{pmatrix}
\frac{m - 1}{m} \kappa_j(x_r, v_t) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_t) \\
\vdots \\
\frac{m - 1}{m} \kappa_j(x_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_j) \end{pmatrix}.\]

For Mahalanobis matrix \( M_j \) with \( j \in \mathbb{I} \), we have

\[
\nabla_{M_j} \mathcal{T}(\hat{X}, \hat{Y}) = \begin{pmatrix}
\frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial M_{j,1,1}} & \cdots & \frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial M_{j,j,t}} \\
\vdots & \ddots & \vdots \\
\frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial M_{j,t,1}} & \cdots & \frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial M_{j,t,t}}
\end{pmatrix},
\]

where we denote by \( M_{j,a,b} \) the element in \( a \)-th row and \( b \)-th column in \( M_j \).

We further have

\[
\frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial M_{j,a,b}} = \frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial c_{\hat{X}}} \frac{\partial c_{\hat{X}}}{\partial M_{j,a,b}} + \frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial c_{\hat{Y}}} \frac{\partial c_{\hat{Y}}}{\partial M_{j,a,b}} + \text{Tr} \left[ \frac{\partial \mathcal{T}(\hat{X}, \hat{Y})}{\partial \Sigma_{\hat{X}, \hat{Y}}} \frac{\partial \Sigma_{\hat{X}, \hat{Y}}}{\partial M_{j,a,b}} \right],
\]

with

\[
\frac{\partial c_{\hat{X}}}{\partial M_{j,a,b}} = \frac{1}{m} \sum_{r=1}^{m} \frac{\partial \hat{x}_r}{\partial M_{j,a,b}} \quad \text{and} \quad \frac{\partial \hat{x}_r}{\partial M_{j,a,b}} = \left( 0, \ldots, \frac{\partial \kappa_j(x_r, v_j)}{\partial M_{j,a,b}}, \ldots, 0 \right)^T,
\]

and

\[
\frac{\partial \kappa_j(x_r, v_j)}{\partial M_{j,a,b}} = \frac{\partial \exp \left( -(x_r - v_j)^T M_j (x_r - v_j) / 2\gamma_j^2 \right)}{\partial M_{j,a,b}} = \frac{-\kappa_j(x_r, v_j)}{2\gamma_j^2} \text{Tr} \left[ \frac{\partial (x_r - v_j)^T M_j (x_r - v_j)}{\partial M_j} \right] = \frac{-\kappa_j(x_r, v_j)}{2\gamma_j^2} \text{Tr} \left[ (x_r - v_j)(x_r - v_j)^T J_{a,b} \right],
\]

30
where \( J_{a,b} \) is the single-entry matrix (1 at \((a, b)\) and zero elsewhere). We similarly calculate \( \frac{\partial c_y}{\partial M_{j,a,b}} \) and have

\[
\frac{\partial \Sigma_{X,Y}}{\partial M_{j,a,b}} = \frac{1}{m + n - 2} \left( \sum_{r=1}^{m} \frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} + \sum_{r=1}^{n} \frac{\partial (y_r - c_Y)(y_r - c_Y)^T}{\partial M_{j,a,b}} \right),
\]

where

\[
\frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} = \begin{pmatrix}
0 & \cdots & \frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} & \cdots & 0 \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
\frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} & \cdots & 0 & \ddots & \vdots \\
\vdots & \ddots & \vdots & \ddots & \ddots \\
0 & \cdots & \frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} & \cdots & 0
\end{pmatrix}.
\]

Here, we have

\[
\frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} = \left( \frac{m - 1}{m} \kappa_j(x_r, v_1) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_1) \right) \left( \frac{m - 1}{m} \frac{\partial \kappa_j(x_r, v_j)}{\partial M_{j,a,b}} - \frac{1}{m} \sum_{s \neq r} \frac{\partial \kappa_j(x_s, v_j)}{\partial M_{j,a,b}} \right).
\]

We similarly have \( \frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} \) and

\[
\frac{\partial (\hat{x}_r - c_X)(\hat{x}_r - c_X)^T}{\partial M_{j,a,b}} = 2 \left( \frac{m - 1}{m} \kappa_j(x_r, v_j) - \frac{1}{m} \sum_{s \neq r} \kappa_j(x_s, v_j) \right) \left( \frac{m - 1}{m} \frac{\partial \kappa_j(x_r, v_j)}{\partial M_{j,a,b}} - \frac{1}{m} \sum_{s = r} \frac{\partial \kappa_j(x_s, v_j)}{\partial M_{j,a,b}} \right).
\]

**Project Mahalanobis matrix onto a positive definite cone**

We can not guarantee the positive-definiteness of Mahalanobis matrices during the optimization process via gradient ascend. Motivated from [43], we project Mahalanobis matrix onto a positive definite cone as follows:

- Present the spectral (eigenvalue) decomposition of a Mahalanobis matrix \( M \) as

\[
M = \sum_{i=1}^{d} \lambda_i p_i p_i^T.
\]

where \( \lambda_1, \lambda_2, \ldots, \lambda_d \) are their eigenvalues with corresponding eigenvectors \( p_1, p_2, \ldots, p_d \).

- Project the Mahalanobis matrix \( M \) onto a positive definite cone

\[
M = \sum_{i=1}^{d} \max \{ \lambda_i, \delta \} v_i v_i^T
\]

for small positive constant \( \delta \).

**C Datasets and Parameter Setting**

**Datasets**

We partition datasets into several disjoint subsets, and then randomly draw data elements from each subset based on the sample fraction, i.e., the proportion of samples to be selected from each
We provide the details of constructing two samples for each dataset as follows:

- **blob** is constructed as the mixture of nine Gaussian modes. We first write
  \[ u_1 = [0, 0], \quad u_2 = [0, 1], \quad u_3 = [0, 2], \]
  \[ u_4 = [1, 0], \quad u_5 = [1, 1], \quad u_6 = [1, 2], \]
  \[ u_7 = [2, 0], \quad u_8 = [2, 1], \quad u_9 = [2, 2], \]
  and
  \[ \Delta_i = \begin{cases} -0.02 - 0.002 \times (i - 1) & \text{for } i < 5 \\ 0 & \text{for } i = 5 \\ 0.02 + 0.002 \times (i - 6) & \text{for } i > 5. \end{cases} \]
  To construct different distributions \( \mathbb{P} \) and \( \mathbb{Q} \), we adapt different covariance structures and
  \[
  \mathbb{P} = \sum_{i=1}^{9} \frac{1}{9} \mathcal{N}(u_i, 0.03 \times I_2) \quad \text{and} \quad \mathbb{Q} = \sum_{i=1}^{9} \frac{1}{9} \mathcal{N}\left(u_i, \begin{bmatrix} 0.03 & \Delta_i \\ \Delta_i & 0.03 \end{bmatrix}\right).
  \]
  To construct identical distribution for two samples, i.e., \( \mathbb{P} = \mathbb{Q} \), we have
  \[
  \mathbb{P} = \sum_{i=1}^{9} \frac{1}{9} \mathcal{N}(u_i, 0.03 \times I_2) \quad \text{and} \quad \mathbb{Q} = \sum_{i=1}^{9} \frac{1}{9} \mathcal{N}(u_i, 0.03 \times I_2).
  \]
  We set the sample size for training to 900 and for testing to 224.

- **dna** is a categorical dataset with 3 classes. For constructing two different samples, we set the sample fraction for one sample to \([0.30, 0.35, 0.35]\) and for the other sample to \([0.45, 0.25, 0.3]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.30, 0.35, 0.35]\) for two samples. We set the sample size for training to 1000 and for testing to 250.

- **agnos** (agnostic) is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to \([0.35, 0.65]\) and for the other sample to \([0.65, 0.35]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.35, 0.65]\) for two samples. We set the sample size for training to 1000 and for testing to 250.

- **topo21** is a regression dataset with continuous target variables. Based on the sorted target variables, we divide the data into 4 equal parts. For constructing two different samples, we set the sample fraction for one sample to \([0.1, 0.3, 0.2, 0.4]\) and for the other sample to \([0.5, 0.2, 0.1, 0.2]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.1, 0.3, 0.2, 0.4]\) for two samples. We set the sample size for training to 2200 and for testing to 550.

- **har** is a categorical dataset with 6 classes. For constructing two different samples, we set the sample fraction for one sample to \([0.10, 0.20, 0.10, 0.20, 0.10, 0.10]\) and for the other sample to \([0.15, 0.15, 0.15, 0.20, 0.15, 0.15]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.10, 0.20, 0.10, 0.20, 0.20, 0.10]\) for two samples. We set the sample size for training to 2200 and for testing to 550.

- **kropt** is a categorical dataset with 18 classes, where we only consider categories 13, 14, 15 and 16 which have the majority of the data. For constructing two different samples, we set the sample fraction for one sample to \([0.15, 0.2, 0.3, 0.35]\) and for the other sample to \([0.35, 0.3, 0.15, 0.15]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.25, 0.25, 0.25, 0.25]\) for two samples. We set the sample size for training to 2000 and for testing to 500.

- **diamonds** is a regression dataset with continuous target variables. Based on the sorted target variables, we divide the data into 4 equal parts. For constructing two different samples, we set the sample fraction for one sample to \([0.35, 0.2, 0.2, 0.25]\) and for the other sample to \([0.2, 0.3, 0.3, 0.2]\); To construct two samples with one identical distribution, we set the same sample fraction \([0.25, 0.25, 0.25, 0.25]\) for two samples. We set the sample size for training to 2000 and for testing to 500.
Table 4: Optimization parameters of our ME\textsubscript{L/B/D} test for different datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Test Locations</th>
<th>Learning Rate</th>
<th>Optimization Epoch</th>
</tr>
</thead>
<tbody>
<tr>
<td>blob</td>
<td>17</td>
<td>0.0007</td>
<td>1000</td>
</tr>
<tr>
<td>dna</td>
<td>15</td>
<td>0.0004</td>
<td>1000</td>
</tr>
<tr>
<td>agnos</td>
<td>15</td>
<td>0.001</td>
<td>200</td>
</tr>
<tr>
<td>topo21</td>
<td>15</td>
<td>0.001</td>
<td>1000</td>
</tr>
<tr>
<td>har</td>
<td>15</td>
<td>0.001</td>
<td>100</td>
</tr>
<tr>
<td>kropt</td>
<td>1</td>
<td>0.002</td>
<td>1000</td>
</tr>
<tr>
<td>diamon</td>
<td>2</td>
<td>0.0013</td>
<td>1000</td>
</tr>
<tr>
<td>cifar10</td>
<td>2</td>
<td>0.003</td>
<td>200</td>
</tr>
<tr>
<td>mnist</td>
<td>2</td>
<td>0.001</td>
<td>500</td>
</tr>
<tr>
<td>santan</td>
<td>10</td>
<td>0.001</td>
<td>1000</td>
</tr>
<tr>
<td>codrna</td>
<td>2</td>
<td>0.01</td>
<td>1000</td>
</tr>
<tr>
<td>sea50</td>
<td>1</td>
<td>0.01</td>
<td>1000</td>
</tr>
<tr>
<td>adult</td>
<td>1</td>
<td>0.001</td>
<td>1000</td>
</tr>
<tr>
<td>labor</td>
<td>1</td>
<td>0.001</td>
<td>1000</td>
</tr>
<tr>
<td>poker</td>
<td>3</td>
<td>0.002</td>
<td>200</td>
</tr>
<tr>
<td>higgs</td>
<td>15</td>
<td>0.001</td>
<td>1000</td>
</tr>
</tbody>
</table>

- Original cifar10 (samples $P$) is compared to adversarial-cifar10 (samples $Q$) following [19], where adversarial-cifar10 is constructed by [86] for distribution shift detection. To construct two samples with one identical distribution, we select randomly between original cifar10 and adversarial-cifar10 and draw two samples from the same dataset. The image is scaled to $32 \times 32$ and we set the sample size for training to 200 for testing to 50.

- mnist contains 70000 handwritten digit images, we compare true mnist data (samples $P$) to Fake-mnist data (samples $Q$) following [39], where the Fake-mnist data is drawn from a pre-trained deep convolutional generative adversarial network [87] and the image is scaled to $32 \times 32$. To construct two samples with one identical distribution, we select randomly between original mnist and Fake-mnist and draw two samples from the same dataset. We set the sample size for training to 600 and for testing to 150.

- santan (santandercustomersatisfaction) is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.8, 0.2]$ and for the other sample to $[0.25, 0.75]$; To construct two samples with one identical distribution, we set the same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 1000 and for testing to 250.

- codrna is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.7, 0.3]$ and for the other sample to $[0.35, 0.65]$; To construct two samples with one identical distribution, we set same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 1000 and for testing to 250.

- sea50 is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.7, 0.3]$ and for the other sample to $[0.3, 0.7]$; To construct two samples with one identical distribution, we set same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 5000 and for testing to 1250.

- adult is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.6, 0.4]$ and for the other sample to $[0.4, 0.6]$; To construct two samples with one identical distribution, we set same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 5000 and for testing to 1250.

- labor is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.6, 0.4]$ and for the other sample to $[0.45, 0.55]$; To construct two samples with one identical distribution, we set same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 5000 and for testing to 1250.

- poker is a categorical dataset with 2 classes. For constructing two different samples, we set the sample fraction for one sample to $[0.2, 0.8]$ and for the other sample to $[0.8, 0.2]$; To construct two samples with one identical distribution, we set same sample fraction $[0.5, 0.5]$ for two samples. We set the sample size for training to 3000 and for testing to 750.
We now present the details for our compared methods as follows:

where

\[
S_{j,i} = \{ x_1 - v_j, ..., x_m - v_j, y_1 - v_j, ..., y_n - v_j \}, \quad S_{j,i} = i-th element of S_j, \quad cS_j = \sum_{i=1}^{n+m} S_{j,i} / (m+n), \quad I_d is an identity matrix of size d \times d to guarantee the positive definiteness with small constant \delta > 0. Here, the elements x and y are come from training samples X and Y, and m and n are the numbers of elements in X and Y respectively.

For initialization of test locations, we first fit a Gaussian distribution for each sample and draw half of the number of test locations from each distribution. This could be expensive for high dimensional dataset, and we can simplify the process by directly sampling test locations from original dataset.

For initialization of the bandwidth parameters of Mahalanobis kernels, we let \( \gamma_j = \gamma \) for every \( j \in [\ell] \) and then linearly search for \( \gamma \) that maximize the statistic from a candidate list with fixed test locations and Mahalanobis matrices, which can be formalized as follows:

\[
\gamma = \arg\max_{\gamma} T(\hat{X}, \hat{Y}) \quad for \quad \gamma \in \{ med^2 \cdot 2^{-4}, med^2 \cdot 2^{-3.6}, \ldots, med^2 \cdot 2^{-3.6} \}, \quad (36)
\]

where \( med \) denotes the median of pairwise Euclidean distances of points in X and Y.

**Compared methods**

We now present the details for our compared methods as follows:

- **ME**\(^2\): The mean embedding method learns a set of test locations and a single Gaussian kernel, and then measures the difference between two mean embeddings with a statistic following \( \chi^2 \) distribution [9, 10];

\(^2\)The code is downloaded from github.com/wittawatj/interpretable-test.
We implement methods for exploring local significant differences in Python, following their respective guidelines as follows:

**Algorithm 3 ME-MaBiD testing**

**Input:** Two testing samples $X'$ and $Y'$, test locations $\{v_i\}_{i=1}^T$, Mahalanobis kernels parameters $\{M_i\}_{i=1}^T$ and $\{\gamma_i\}_{i=1}^T$, inference direction $F$, significance level $\alpha$ and parameter $\beta$

**Output:** $h$

1. Calculate the embedding testing samples $\hat{X}'$ and $\hat{Y}'$ based on Eqn. (2)
2. Calculate mean embeddings $c_{\hat{X}'} = \frac{\sum_{i=1}^{n'} \hat{x}'_i}{n'}$ and $c_{\hat{Y}'} = \frac{\sum_{i=1}^{n} \hat{y}'_i}{n}$
3. Calculate the pooled covariance matrix $\Sigma_{\hat{X}', \hat{Y}'}$ based on Eqn. (3)
4. Calculate the Schur decomposition of the pooled covariance matrix: $L'L' = \Sigma_{\hat{X}', \hat{Y}'}$
5. if $F^T(c_{\hat{X}'} - c_{\hat{Y}'}) \geq 0$ then
6. $h = \|[\chi^2(T(\hat{X}', \hat{Y}'))] \leq \beta \alpha$
7. else
8. $h = \|[\chi^2(T(\hat{X}', \hat{Y}'))] \leq (2 - \beta \alpha)]$
9. end if
10. return $h$

**Algorithm 4 Exploring the local significant differences**

**Input:** Two testing samples $X, Y$, Mahalanobis kernels’ parameters $\{M_i\}_{i=1}^T$ and $\{\gamma_i\}_{i=1}^T$, test locations $\{v_i\}_{i=1}^T$, partition tree $T$

**Output:** $t^*$

1. Calculate the embedding testing samples $\hat{X}'$ and $\hat{Y}'$ based on Eqn. (2)
2. Get the local two samples $\hat{X}_{B_i}'$ and $\hat{Y}_{B_i}'$ for every $i \in [s]$ based on partition Tree $T$
3. Calculate bi-directional masked p-value $g(\hat{X}_{B_i}', \hat{Y}_{B_i}')$ for each rectangle region $B_i$
4. Resort rectangle regions as $B_{(1)}, B_{(2)}, \ldots, B_{(s)}$ in a non-increasing order based on $g(\cdot, \cdot)$
5. Calculate bi-directional hypothesis $h(\hat{X}_{B_i}', \hat{Y}_{B_i}')$ for each rectangle region $B_i$ by Eqn. (7)
6. Calculate the index set $t^*$ based on Eqn. (11)
7. return $t^*$

- **C2ST-S:** A binary classification neural network is trained and the statistic is computed as the accuracy over a hold-out set of two samples [13];
- **MMDAgg:** A solution for the fundamental kernel selection problem involves the aggregation of a large number of kernels with several bandwidths, where the incomplete $U$-statistics are used to measure the difference between two samples Schrab et al. [75]. Notice that we set the testing sample size for MMDAgg to 5 times that of the other methods, since it does not require training;
- **MMD-D:** A deep kernel approach for Maximum Mean Discrepancy (MMD), where the parameters of a neural network, two lengthscales of Gaussian kernels and a regularization parameter are optimized [39];
- **C2ST-L:** A binary classification neural network is trained and the statistic is computed as the difference between outputs of the logit function corresponding to two samples [14, 18];
- **AutoML:** A binary classifier is trained based on Automated Machine Learning techniques and the statistic is same to C2ST-L approach [19].

We implement methods for exploring local significant differences in Python, following their respective guidelines as follows:

- **FDG:** Partition the sample space based on probability binning and then compare the cardinalities of two samples over rectangle regions, where a normalized chi-squared value is computed for each bin to measure the local difference [24];

---

1. The code is downloaded from [github.com/lopezpaz/classifier_tests](https://github.com/lopezpaz/classifier_tests).
2. The code is downloaded from [github.com/autoninschlabaggine-paper](https://github.com/autoninschlabaggine-paper).
3. The code is downloaded from [github.com/fengliu90/DK-for-TST](https://github.com/fengliu90/DK-for-TST).
4. The code is downloaded from [github.com/xycheng/net_logit_test](https://github.com/xycheng/net_logit_test).
Step IV: 1) Split node 4 at y-axis with median position
2) Split node 5 at x-axis with median position
3) Split node 6 at x-axis with median position
4) Split node 7 at y-axis with median position

Partition tree
1
2 3

Step III: 1) Split node 2 at x-axis with median position
2) Split node 3 at y-axis with median position

Partition tree
1
2 3

Step II: Split node 1 at x-axis with median position

Partition tree
1
2 3

Step I: Initialization with root 1

Partition tree
1

Figure 9: An illustration for our partition tree with splitting regions.

- **K-PRIM**: Partition the space based on patient rule induction method along with the information of kernel density estimation, and use a statistic following $\chi^2$ distribution to identify local distributional difference [28];
- **MRS**: Partition the sample space based on poly tree method, which splits a leaf node based on the median of randomly selected feature, and then measure the difference between local two samples based on Binomial distribution [26];
- **TEAM**: Partition the sample space on the median of the feature with largest sample variance, and then measure differences between local two samples with Binomial distribution [27];
- **BTLDD**: Estimate the conditional probabilities of two samples based on a regression model and then cluster those elements with significant different conditional probabilities [77];
- **MMDT**: Partition the sample space into multiple equal grids based on quantile values of the features, and then estimate the kernel densities of two samples and measure local differences based on Welch’s two-sample t-test statistic [29].

For the exploration of local significant differences, we take density differences [78] between two samples in a local region as an evaluation measure for local significant differences, and follow the works of [79, 80] based on k-NN density estimator with $k = 20$. Here, denote by $X$ and $Y$ the available data for density evaluation, we have estimated density functions as follows:

$$f_X(z) := \frac{20}{N \cdot v_d \cdot r_X(z)^d} \quad \text{and} \quad f_Y(z) := \frac{20}{N \cdot v_d \cdot r_Y(z)^d},$$

where $z \in [0, 1]^d$, $N$ denotes sample size for density estimation and $v_d$ is the volume of a unit ball in $\mathbb{R}^d$. Denote by $r_X(z)^d$ and $r_Y(z)^d$ the distances from $z$ to its $20$-th nearest neighbors in $X$ and $Y$, respectively. For density estimation, we use all data of diamond, and $1,000,000$ data of other datasets due to the limitation of time complexity.

For our ME$_{MaBiD}$ test, we present the detailed training procedure in Algorithm 2 and testing procedure in Algorithm 3. Figure 9 is a pictorial illustration to present the region-splitting method and clarify that our partition tree is constructed iteratively: We initiate tree root with embedding space, and during each iteration, we select randomly one of those leaves with the largest size of training data points, and select the feature of the largest statistics value and with the median splitting position. We present the detailed description on the exploration of local significant differences in Algorithm 4.
D Experimental comparisons

Running time

For fair comparisons, we run all experiments on a single core without parallel optimizations, and experiments are performed with Python on nodes of a computational cluster with a single CPU (Intel Core i9-10900X 3.7GHz) and a single GPU (GeForce RTX 2080 Ti), running Ubuntu with 128GB main memory. For these methods based on deep neural networks, such as C2ST-S, C2ST-L and MMD-D, we run experiments on GPU. For other methods except for AutoMLTST, we run experiments on CPU. For AutoMLTST, we run experiments on GPU for cifar and mnist, and run experiments on CPU for other datasets.

We further compare the average training time and testing time (in seconds) for different methods on two-sample test, as shown in Figure 10 and Figure 11. Notice that, the training time of MMDAgg is 0, since it has no training procedure. In testing procedure, our ME and MaBiD methods calculate the rejection threshold based on the asymptotic distribution \( \chi^2_{\ell} \), whereas other methods perform permutation test or wild bootstrap.

Additional experimental comparisons

Table 5 presents the comparisons of type-I error for different methods. As can be seen, type-I error is limited about \( \alpha = 0.05 \) for all compared methods, which shows the effectiveness of test power in experiments for different methods.

Figure 12 shows the experimental comparisons between Mahalanobis kernels and Gaussian kernels, and it is obvious that Mahalanobis kernels achieves higher test power with better performance by exploiting local regions and directional information. Figure 13 presents experimental comparisons of test power with different training sample sizes. The testing sample size is set as 200 for blob, 1000 for higgs, 200 for codrna and 150 for sea50. As can be seen, our ME test takes better test powers empirically than other methods w.r.t different training sample sizes by incorporating correlation and directional information of two samples.

Figure 10: Comparisons of training time for different methods on two-sample test. Note that y-axis is in log-scale.

Figure 11: Comparisons of testing time for different methods on two-sample test. Note that y-axis is in log-scale.
Table 5: Comparisons of type-I error (mean±std).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Ours</th>
<th>ME</th>
<th>MMDAgg</th>
<th>MMD-D</th>
<th>C2ST-L</th>
<th>C2ST-S</th>
<th>AutoMLTST</th>
</tr>
</thead>
<tbody>
<tr>
<td>blob</td>
<td>0.04 ± 0.25</td>
<td>0.06 ± 0.56</td>
<td>0.06 ± 0.54</td>
<td>0.04 ± 0.18</td>
<td>0.03 ± 0.32</td>
<td>0.004 ± 0.06</td>
<td>0.19 ± 0.16</td>
</tr>
<tr>
<td>dna</td>
<td>0.05 ± 0.36</td>
<td>0.05 ± 0.68</td>
<td>0.06 ± 0.35</td>
<td>0.02 ± 0.04</td>
<td>0.03 ± 0.25</td>
<td>0.059 ± 0.54</td>
<td>0.03 ± 0.034</td>
</tr>
<tr>
<td>agnostic</td>
<td>0.047 ± 0.23</td>
<td>0.049 ± 0.34</td>
<td>0.051 ± 0.43</td>
<td>0.02 ± 0.04</td>
<td>0.042 ± 0.29</td>
<td>0.063 ± 0.45</td>
<td>0.046 ± 0.39</td>
</tr>
<tr>
<td>top21</td>
<td>0.03 ± 0.19</td>
<td>0.049 ± 0.32</td>
<td>0.038 ± 0.34</td>
<td>0.049 ± 0.39</td>
<td>0.019 ± 0.22</td>
<td>0.051 ± 0.24</td>
<td>0.038 ± 0.24</td>
</tr>
<tr>
<td>har</td>
<td>0.046 ± 0.03</td>
<td>0.033 ± 0.26</td>
<td>0.040 ± 0.30</td>
<td>0.045 ± 0.28</td>
<td>0.022 ± 0.14</td>
<td>0.061 ± 0.04</td>
<td>0.052 ± 0.41</td>
</tr>
<tr>
<td>kropt</td>
<td>0.033 ± 0.16</td>
<td>0.021 ± 0.06</td>
<td>0.013 ± 0.00</td>
<td>0.030 ± 0.29</td>
<td>0.046 ± 0.16</td>
<td>0.033 ± 0.12</td>
<td>0.033 ± 0.21</td>
</tr>
<tr>
<td>diamon</td>
<td>0.042 ± 0.24</td>
<td>0.029 ± 0.24</td>
<td>0.017 ± 0.12</td>
<td>0.046 ± 0.23</td>
<td>0.013 ± 0.10</td>
<td>0.021 ± 0.16</td>
<td>0.033 ± 0.31</td>
</tr>
<tr>
<td>cifar</td>
<td>0.015 ± 0.10</td>
<td>0.012 ± 0.11</td>
<td>0.025 ± 0.14</td>
<td>0.078 ± 0.68</td>
<td>0.021 ± 0.16</td>
<td>0.027 ± 0.17</td>
<td>0.020 ± 0.14</td>
</tr>
<tr>
<td>mnist</td>
<td>0.043 ± 0.038</td>
<td>0.013 ± 0.16</td>
<td>0.041 ± 0.28</td>
<td>0.092 ± 0.68</td>
<td>0.032 ± 0.41</td>
<td>0.057 ± 0.26</td>
<td>0.049 ± 0.39</td>
</tr>
<tr>
<td>santan</td>
<td>0.037 ± 0.031</td>
<td>0.092 ± 0.43</td>
<td>0.029 ± 0.06</td>
<td>0.058 ± 0.19</td>
<td>0.025 ± 0.18</td>
<td>0.037 ± 0.027</td>
<td>0.042 ± 0.36</td>
</tr>
<tr>
<td>codrna</td>
<td>0.013 ± 0.10</td>
<td>0.017 ± 0.16</td>
<td>0.075 ± 0.80</td>
<td>0.042 ± 0.28</td>
<td>0.046 ± 0.47</td>
<td>0.042 ± 0.012</td>
<td>0.008 ± 0.12</td>
</tr>
<tr>
<td>sea50</td>
<td>0.032 ± 0.27</td>
<td>0.021 ± 0.12</td>
<td>0.058 ± 0.26</td>
<td>0.088 ± 0.59</td>
<td>0.029 ± 0.41</td>
<td>0.058 ± 0.056</td>
<td>0.050 ± 0.47</td>
</tr>
<tr>
<td>adult</td>
<td>0.024 ± 0.16</td>
<td>0.021 ± 0.16</td>
<td>0.029 ± 0.16</td>
<td>0.038 ± 0.10</td>
<td>0.033 ± 0.16</td>
<td>0.033 ± 0.26</td>
<td>0.062 ± 0.020</td>
</tr>
<tr>
<td>labor</td>
<td>0.017 ± 0.16</td>
<td>0.037 ± 0.10</td>
<td>0.054 ± 0.059</td>
<td>0.056 ± 0.29</td>
<td>0.029 ± 0.24</td>
<td>0.029 ± 0.26</td>
<td>0.042 ± 0.26</td>
</tr>
<tr>
<td>poker</td>
<td>0.029 ± 0.16</td>
<td>0.013 ± 0.10</td>
<td>0.054 ± 0.41</td>
<td>0.078 ± 0.47</td>
<td>0.029 ± 0.06</td>
<td>0.058 ± 0.031</td>
<td>0.050 ± 0.054</td>
</tr>
<tr>
<td>higgs</td>
<td>0.046 ± 0.11</td>
<td>0.024 ± 0.16</td>
<td>0.043 ± 0.31</td>
<td>0.066 ± 0.28</td>
<td>0.068 ± 0.098</td>
<td>0.048 ± 0.17</td>
<td>0.034 ± 0.039</td>
</tr>
</tbody>
</table>

Figure 12: The comparisons of test power vs sample size for Mahalanobis kernel and Gaussian kernel.

Figure 13: The comparisons of test power with different training sample sizes.

Figure 14: The comparisons of test power vs sample size for our bi-directional masked p-value and prior Sidak.

Figure 15: The comparisons of $\xi$ with different sample size and number of test locations.

Figure 14 presents experimental comparisons between our bi-directional masked p-value and previous Sidak [67]. As can be seen, our bi-directional masked p-value achieves higher test power by exploiting the significant level of local difference. Figure 15 analyzes the relationship between probability $\xi = \Pr[\text{sgn}(F_B^T L_B^{-1} (c_{X_B^T} - c_{Y_B^T}) \geq 1)]$ and sample size, by considering different numbers of test locations $\ell \in \{2, 6, 10, 14, 18\}$. As can be seen, $\xi$ gradually increases as the sample size for different number of test locations, which shows the necessity of directional information in two-sample test.

Figure 16 analyzes the relationship between the number of local regions $s$ and probability $\xi = \Pr[\text{sgn}(F_B^T L_B^{-1} (c_{X_B^T} - c_{Y_B^T}) \geq 1)]$. The probability $\xi$ increases with $s$, and hence can effectively partition the space into local regions with different inference directions. Figure 17 presents
Figure 16: The comparisons of $\xi$ for partition trees with $s \in [2^0, 2^3]$. The value of $\xi$ increases as $s$ increases.

Figure 17: The correlation between the optimal parameter $\beta$ and probability $\xi = \Pr[\text{sgn}(F_L^\top B_L^{-1} (c_{X}^\top - c_{Y}^\top)) \geq 1]$. Here, $N$ denotes sample size.

additional experiments on the relationship between the optimal parameter $\beta$ and the probability $\xi = \Pr[F_L^\top L_B^{-1} (c_{X}^\top - c_{Y}^\top) \geq 0]$, as shown in Figure 6.