Research Internship

Fabian Baier

May 2025

1 Introduction

The project was to implement a simulation study, which was provided in R code, and optimise it in Python. The simulation study is based on the paper: Distribution free tests for model selection based on maximum mean discrepancy with estimated parameters by Brueck, Fermanian and Min (https://arxiv.org/abs/2305.07549), in the following abbreviated by BFM. The goal of the BFM test statistic is to decide, given an underlying data set X and two data sets Y_1 and Y_2, which data sets of Y_1 or Y_2 is more similar to X. Alternatively when only provided one other data set Y we want to test, if Y is similar to X. For the latter case, which is called model specification, the null hypothesis is that X and Y are drawn from the same distribution. For the model specification test, the null hypothesis is:

$$\mathcal{H}_{0,\mathcal{M}} : \ \mathrm{MMD}(P_{\alpha_\star}, P) = 0, \tag{1}$$

where P is the underlying distribution of X, $\mathcal{M}$ is the model (a family of probability measures) and $P_{\alpha_\star}$ is the distribution of Y. This corresponds to the case in the BFM test, where the model selection is false. The output to look for in the BFM test is the spec-reject, if it is true, we reject the null hypothesis, meaning that X and Y are too different. In the case where we have two additional data sets Y_1 and Y_2, this case is called model selection. The null hypothesis is that Y_1 and Y_2 are too similar and we cannot decide. For the sake of simplicity, we assume Y= Y_1. Let now $Q_{\beta_\star}$ be the distribution of Y_2, then the null hypothesis is:

$$\mathcal{H}_{0,\mathcal{M}_1,\mathcal{M}_2} : \ \mathrm{MMD}(P_{\alpha_\star}, P) = \mathrm{MMD}(Q_{\beta_\star}, P). \tag{2}$$

This corresponds to the case in the BFM test, where model selection is true. The output to look at in the BFM test to decide which data set is more

similar to X is the selec-test-stat, when it is positive Y_1 is closer to X than Y_2, and if it is negative Y_2 is closer to X than Y_1. The output selec-reject indicates whether the difference between Y_1 and Y_2 is significant at level=0.05. The BFM test statistic is highly relevant for applications such as generative machine learning, transfer learning, Bayesian statistics, clustering adaptive MCMC methods. For the coding part of the project, I was expected to use Numpy and parallelise the code to avoid any for loops in computationally expensive parts of the algorithm. As I was curious, I also implemented the code using Cupy, a GPU-based package similar to Numpy. I was able to achieve a performance improvement of around 4200-4800 compared to the R code. For further improvements, memory management or some kind of probabilistic algorithm would be needed, as we could easily use approximate values for our calculations. These techniques must be used, otherwise the BFM test isn't feasible for large n, which are common in machine learning (such as sample size n=1 million or above). The main goal of the Research Internship was to recreate the figures from the BFM test, and write custom functions to generate normal distributed matrices and measure the time compared to the R code. Here, the performance improvements were about a factor of 60-70 when using the Numpy version with the cdist function from the SciPy package.

2 Maximum Mean Discrepancy (MMD) and MMD estimators

Maximum Mean Discrepancy (MMD) is a statistical measure used to quantify the difference between two probability distributions based on samples drawn from them. Given two distributions P and Q, with samples $X = \{x_i\}_{i=1}^n$ drawn from P, and $Y = \{y_j\}_{j=1}^m$ drawn from Q, MMD is defined using a kernel function $k(\cdot, \cdot)$. Formally, the MMD between distributions P and Q in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ associated with kernel k is:

$$\mathrm{MMD}^2(\mathcal{H}, P, Q) = \|\mu_P - \mu_Q\|_{\mathcal{H}}^2, \tag{3}$$

where μ_P and μ_Q are mean embeddings of P and Q into the RKHS:

$$\mu_P = \mathbb{E}_{x\sim P}[k(x, \cdot)], \quad \mu_Q = \mathbb{E}_{y\sim Q}[k(y, \cdot)]. \tag{4}$$

One can deduce that:

$$\mathrm{MMD}^2(P_1, P_2) = \mathbb{E}_{X,X'\sim P_1}\Big[\mathbb{E}_{Y,Y'\sim P_2}\big[k(X, X') - 2k(X, Y) + k(Y, Y')\big]\Big].$$

Thus, the computation of $\mathrm{MMD}(P_1, P_2)$ relies solely on expectations of known functionals w.r.t. P_1 and P_2.

Empirical Estimation of MMD: Biased vs. Unbiased Estimators

The empirical estimation of MMD is based on finite samples. The unbiased estimator, typically employed, is defined as:

$$\widehat{\mathrm{MMD}}_u^2 = \frac{1}{n(n-1)} \sum_{i \neq j}^{n} k(x_i, x_j) + \frac{1}{m(m-1)} \sum_{i \neq j}^{m} k(y_i, y_j) \tag{5}$$

$$- \frac{2}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} k(x_i, y_j). \tag{6}$$

A biased estimator, which includes diagonal terms and is simpler computationally, is:

$$\widehat{\mathrm{MMD}}_b^2 = \frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, x_j) + \frac{1}{m^2} \sum_{i,j=1}^{m} k(y_i, y_j) - \frac{2}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} k(x_i, y_j). \tag{7}$$

In the BFM paper, the MMD_z estimator is used since m=n and they also want to exclude cross-similarity terms. The MMD_z estimator is defined similarly to the unbiased estimator:

$$\widehat{\mathrm{MMD}}_z^2(P_1, P_2) := \frac{1}{n(n-1)} \sum_{i \neq j}^{n} \big\{ k(X_i, X_j) - 2k(X_i, Y_j) + k(Y_i, Y_j) \big\}. \tag{8}$$

Remark: For $m = n$, the estimates MMD_u and MMD_z differ as MMD_z^2 excludes terms $\{k(X_i, Y_i) : i = 1, ..., m\}$. Hence, changing the order of samples alters MMD_z^2. The statistic MMD_z^2 has slightly higher variance but can be computed more efficiently. The biased statistic MMD_b can be represented by replacing U-statistics with V-statistics, yielding faster computational complexity of $O((m+n)^2)$. These estimators and their properties are thoroughly discussed by Gretton et al. (2012). (See the master thesis of Tobias Solfronk: Maximum Mean Discrepancy for Model Comparisons)

2.1 $\widehat{\mathrm{MMD}}$ is degenerate under the null hypothesis

Under the null hypothesis $H_0 : P_1 = P_2$, let us denote $P := P_1 = P_2$. Then all samples X_i and Y_j are i.i.d. from the same distribution P.

Define the U-statistic kernel function:

$$h(z_1, z_2) := k(x_1, x_2) + k(y_1, y_2) - k(x_1, y_2) - k(x_2, y_1)$$

where $z_1 = (x_1, y_1)$ and $z_2 = (x_2, y_2)$.

Then the unbiased estimator of the squared MMD becomes:

$$\widehat{\mathrm{MMD}}^2 = \frac{1}{m(m-1)} \sum_{i \neq j} h(Z_i, Z_j)$$

where $\{Z_1, \ldots, Z_{m+n}\} \sim P$ is the combined sample.

A U-statistic is said to be *degenerate* if the expectation of its kernel function conditioned on one argument is zero:

$$\mathbb{E}_{Z_2}[h(Z_1, Z_2)] = 0 \quad \text{for all } Z_1$$

Let us verify this under H_0. Fix $z_1 = (x_1, y_1)$, and compute:

$$\begin{aligned}\mathbb{E}_{z_2}[h(z_1, z_2)] &= \mathbb{E}_{x_2, y_2}\left[k(x_1, x_2) + k(y_1, y_2) - k(x_1, y_2) - k(x_2, y_1)\right] \\ &= \mathbb{E}_x[k(x_1, x)] + \mathbb{E}_y[k(y_1, y)] - \mathbb{E}_y[k(x_1, y)] - \mathbb{E}_x[k(x, y_1)]\end{aligned}$$

Since $x, y \sim P$, all marginal expectations are equal:

$$\mathbb{E}_x[k(x_1, x)] = \mathbb{E}_y[k(x_1, y)], \quad \mathbb{E}_y[k(y_1, y)] = \mathbb{E}_x[k(x, y_1)]$$

Thus,

$$\mathbb{E}_{z_2}[h(z_1, z_2)] = 0$$

Therefore, the MMD U-statistic is degenerate under H_0.

2.2 Non-degeneracy of $\widehat{\mathrm{MMD}}_q$ as a U-statistic under H_0

We consider the quadratic MMD estimator defined via a U-statistic over block-pairs of data:

$$\begin{aligned}\widehat{\mathrm{MMD}}_q(P_1, P_2) := \frac{1}{m(m-1)} \sum_{\substack{i,j=1 \\ i \neq j}}^{m} \big\{ & k(X_{2i-1}, X_{2j-1}) - k(Y_{2j}, X_{2i}) \\ & - k(Y_{2i}, X_{2j}) + k(Y_{2i-1}, Y_{2j-1}) \big\} = \frac{1}{m(m-1)} \sum_{\substack{i,j=1 \\ i \neq j}}^{m} q\big([\boldsymbol{X}, \boldsymbol{Y}]_{2i-1,2i}, [\boldsymbol{X}, \boldsymbol{Y}]_{2j-1,2j}\big),\end{aligned} \tag{9}$$

where $m = n/2$, and the kernel q is defined as:

$$q\big([\boldsymbol{x},\boldsymbol{y}]_{1:2},[\boldsymbol{x},\boldsymbol{y}]_{3:4}\big) := k(x_1,x_3) - k(x_4,y_2) - k(x_2,y_4) + k(y_1,y_3).$$

Here, $[\boldsymbol{x},\boldsymbol{y}]_{i:j} := \big((x_i,y_i),(x_{i+1},y_{i+1})\big)$, for $i < j$, denotes a pair of jointly sampled observations.

Claim: $\widehat{\mathrm{MMD}}_q$ is a non-degenerate U-statistic under $H_0 : P_1 = P_2$

Assume $P_1 = P_2 = P$. Then all observations $X_1,\ldots,X_n \sim P$, and similarly $Y_1,\ldots,Y_n \sim P$, independently.

Each pair $Z_i := \big((X_{2i-1},Y_{2i-1}),(X_{2i},Y_{2i})\big)$ is i.i.d. from $(P\times P)^2$. Therefore, the statistic is a symmetric U-statistic of order 2 with kernel $q(Z_i,Z_j)$.

To determine whether the statistic is degenerate under H_0, consider the first-order projection of the kernel:

$$\psi(Z_1) := \mathbb{E}_{Z_2}[q(Z_1,Z_2)] = \mathbb{E}_{x_3,x_4,y_3,y_4}\big[k(x_1,x_3) - k(x_4,y_2) - k(x_2,y_4) + k(y_1,y_3)\big].$$

Let $Z_1 = \big((x_1,y_1),(x_2,y_2)\big)$, and $Z_2 = \big((x_3,y_3),(x_4,y_4)\big)$, with all $x_i,y_i \sim P$ independently.

We expand this as:

$$\psi(Z_1) = \mathbb{E}_x[k(x_1,x)] + \mathbb{E}_y[k(y_1,y)] - \mathbb{E}_x[k(x,y_2)] - \mathbb{E}_y[k(x_2,y)].$$

Under H_0, all variables are i.i.d. from P. Now, suppose $\psi(Z_1) = 0$ for some $Z_1 = ((x_1,y_1),(x_2,y_2))$. Then:

$$\mathbb{E}_x[k(x_1,x)] - \mathbb{E}_x[k(x,y_2)] = \mathbb{E}_y[k(x_2,y)] - \mathbb{E}_y[k(y_1,y)].$$

But since $x \sim P$, the function $f(a) := \mathbb{E}_x[k(a,x)]$ is the kernel mean embedding $\mu_P(a)$. Therefore, this simplifies to:

$$\mu_P(x_1) - \mu_P(y_2) = \mu_P(x_2) - \mu_P(y_1)$$

which implies:

$$\mu_P(x_1) + \mu_P(y_1) = \mu_P(x_2) + \mu_P(y_2)$$

In other words, $\psi(Z_1) = 0 \iff \mu_P(x_1) + \mu_P(y_1) = \mu_P(x_2) + \mu_P(y_2)$.

This condition is only satisfied when the images of x_1,y_1,x_2,y_2 under the kernel mean embedding satisfy a specific linear relationship. For any *characteristic kernel*, the mapping $x \mapsto \mu_P(x)$ is injective, so this equality only holds for a measure-zero set of $Z_1 \in \mathcal{Z}$ under continuous P. Therefore, the U-statistic $\widehat{\mathrm{MMD}}_q$ is *non-degenerate* under the null hypothesis $P_1 = P_2$, in the strong sense that $\psi(Z_1) \neq 0$ for **almost** every Z_1.

3 BFM test statistic

To understand what the BFM test statistic tries to accomplish, we first have to look at the components of it and which issues it solves. By standard U-statistic theory (Serfling, 1980) it is known that under the null hypothesis,

$$n\widehat{\mathrm{MMD}^2}(P_1, P_2) \xrightarrow{\text{law}} \sum_{i=1}^{\infty} \lambda_i \left(\mathcal{X}_i^2 - 2\right),$$

where $\mathcal{X}_i \sim \mathcal{N}(0, 2)$ and the λ_i denote the (possibly infinitely many) eigenvalues associated with the functional equation $\mathbb{E}\Big[\big\{k(X, y) - \mu_{P_1}(X) - \mu_{P_1}(y) + \mathbb{E}_Y[\mu_{P_1}(Y)]\big\}\psi(X)\Big] = \lambda\psi(y)$ for every $y \in \mathcal{S}$. The problem here is that estimating the eigenvalues is too complex or impossible, so a potential test should not be based on $n\widehat{\mathrm{MMD}^2}(P_1, P_2)$. The other possibilities discussed in the BFM paper would require a pre-test to test if $P_{\alpha_\star} = P = Q_{\beta_\star}$, but this would introduce new problems to solve. The BFM paper solves the problem to avoid pre-testing and provides a asymptotically distribution free approach for model selection and specification. They also show that their proposed test statistic is converging to a standard normal distribution under the null hypothesis. The main advantage of the BFM test statistic is that it is simpler to compute, and the asymptotic standard normal law is not influenced by the parameter estimation (i.e. $\alpha_\star$, the optimal α that minimises some custom loss function), which was a major problem in the other papers. Let's first discuss the advantages of the model specification test. The first problem the BFM test statistic solves is that in the Gretton paper (2006), where the test is only based on $\widehat{\mathrm{MMD}^2}(P_{\alpha_n}, P)$, which is an estimate of $\widehat{\mathrm{MMD}^2}(P_{\alpha_\star}, P)$. Assume that the estimator α_n of a value $\alpha_\star$ is obtained from the sample drawn under P, which is the natural setup when P_{α_n} is a model for P. Then, any sample from P_{α_n} is inherently dependent due to the common α_n and it is not independent of the initial sample from P. Note that Gretton requires i.i.d. samples from P_{α_n} to compute $\widehat{\mathrm{MMD}^2}(P_{\alpha_n}, P)$, which is problematic. Additionally one might hope that $n\widehat{\mathrm{MMD}^2}(P_{\alpha_n}, P)$ has the same asymptotic law as $n\widehat{\mathrm{MMD}^2}(P_{\alpha_\star}, P)$. But that isn't the case as shown in section 2.1 in the BFM paper, due to the influence of parameter estimation. Thus they introduce a distribution free test statistic based on

MMD, first they introduce another MMD variant,

$$\widehat{\mathrm{MMD}}_q^2(P_1,P_2) := \frac{1}{n/2(n/2-1)} \sum_{\substack{i,j=1 \\ i\neq j}}^{n/2} \big\{k(X_{2i-1},X_{2j-1}) - k(Y_{2j},X_{2i})$$

$$- k(Y_{2i},X_{2j}) + k(Y_{2i-1},Y_{2j-1})\big\} = \frac{1}{n/2(n/2-1)} \sum_{\substack{i,j=1 \\ i\neq j}}^{n/2} q\big([\boldsymbol{X},\boldsymbol{Y}]_{2i-1,2i}, [\boldsymbol{X},\boldsymbol{Y}]_{2j-1,2j}\big),$$

where $[\boldsymbol{X}]_{i:j} := \big(X_i, X_{i+1}, \ldots, X_j\big)$, $[\boldsymbol{Y}]_{i:j} := \big(Y_i, Y_{i+1}, \ldots, Y_j\big)$ and $[\boldsymbol{X},\boldsymbol{Y}]_{i:j} := \big((X_i,Y_i),(X_{i+1},Y_{i+1}),\ \ldots,\ (X_j,Y_j)\big)$ for $1 \le i < j \le n$, and

introducing the U-statistic kernel

$$q\big([\boldsymbol{x},\boldsymbol{y}]_{1:2}, [\boldsymbol{x},\boldsymbol{y}]_{3:4}\big) := k(x_1,x_3) - k(x_4,y_2) - k(x_2,y_4) + k(y_1,y_3).$$

The main idea of their test statistic is taking a linear combination of the standard MMD (meaning MMD_z) and MMD_q. It follows that $\widehat{\mathrm{MMD}}_q^2(P_1,P_2)$ is an unbiased estimator of $\mathrm{MMD}^2(P_1,P_2)$ and by U-statistic theory it follows that

$$\sqrt{n}\big\{\widehat{\mathrm{MMD}}_q^2(P_1,P_2) - \mathrm{MMD}^2(P_1,P_2)\big\} \xrightarrow{\text{law}} \mathcal{N}(0,\sigma_q^2),$$

with $\sigma_q^2 > 0$, essentially if and only if P_1 or P_2 is not a Dirac probability measure. Some may think that that implies, that a test should only be based on MMD_q, but since it doesn't use all pairs of observations there will be a power loss. That's why a linear combination of the two is the solution to approximately keep the power of the $\mathrm{MMD}^2(P_1,P_2)$. Thus they introduce:

$$\widehat{\mathrm{MMD}}_{\epsilon_n}^2(P_1,P_2) := \widehat{\mathrm{MMD}}^2(P_1,P_2) + \epsilon_n \widehat{\mathrm{MMD}}_q^2(P_1,P_2),$$

with weights $\epsilon_n > 0$.

If $\epsilon_n := \epsilon > 0$ is a constant, it is obvious that $\sqrt{n}\,\widehat{\mathrm{MMD}}_{\epsilon_n}^2(P_1,P_2) \xrightarrow{\text{law}} \epsilon\mathcal{N}(0,\sigma_q^2)$ under $\mathcal{H}_0$, since $\sqrt{n}\,\widehat{\mathrm{MMD}}^2(P_1,P_2)$ tends to zero in probability when $P_1 = P_2$. However, the choice $\epsilon_n = \epsilon > 0$ may lead to a power loss, similar to a test based on $\sqrt{n}\,\widehat{\mathrm{MMD}}_q^2(P_1,P_2)$. Therefore, we impose that ϵ_n tends to zero in probability hereafter.

So for the test in the specification test scenario follows:

$$\mathcal{T}_n(\mathcal{M},P) := \sqrt{n}\,\frac{\widehat{\mathrm{MMD}}_{\epsilon_n}^2(P_{\alpha_n},P)}{\hat{\sigma}_n}, \tag{10}$$

for some sequence of parameters $(\alpha_n)_{n\geq 1}$ that weakly converges to $\alpha_\star$ at rate $n^{-1/2}$ and $\hat{\sigma}_n$ is the estimator of the standard deviation. In the next chapter, it will be stated under which conditions this test statistic will converge to a standard normal distribution under the null hypothesis.
In case of model selection the test for $\mathcal{H}_{0,\mathcal{M}_1,\mathcal{M}_2} : \mathrm{MMD}(P_{\alpha_\star}, P) = \mathrm{MMD}(Q_{\beta_\star}, P)$, will be based on

$$\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P) := \sqrt{n} \frac{\widehat{\mathrm{MMD}^2_{\epsilon_n}}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}^2_{\epsilon_n}}(Q_{\beta_n}, P)}{\hat{\tau}_n}, \qquad (11)$$

where $\hat{\tau}_n^2 = \hat{\tau}_n^2(\epsilon_n, P_{\alpha_n}, Q_{\beta_n}, P)$ denotes a natural estimator of the asymptotic variance of $\sqrt{n}\big\{\widehat{\mathrm{MMD}^2_{\epsilon_n}}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}^2_{\epsilon_n}}(Q_{\beta_n}, P)\big\}$.

4 Central Limit Theorem

For the asymptotic behaviour of the BFM test statistic in the case of model specification they show the following theorem.

Theorem 1. *Assume $\epsilon_n \to 0$, $\epsilon_n\sqrt{n} \to \infty$ in probability, $\sqrt{n}(\alpha_n - \alpha_\star) = O_{\mathbb{P}}(1)$ and that some technical Assumptions as in the BFM paper hold.*

1. *If $P = P_{\alpha_\star}$, i.e. under $\mathcal{H}_{0,\mathcal{M}}$, we have*

$$\mathcal{T}_n(\mathcal{M}, P) = \sqrt{n} \frac{\widehat{\mathrm{MMD}^2_{\epsilon_n}}(P_{\alpha_n}, P)}{\hat{\sigma}_n} \xrightarrow{law} \mathcal{N}(0,1),$$

   where $\hat{\sigma}_n = \tilde{\sigma}_{\alpha_n} + \epsilon_n \tilde{\sigma}_{q,\alpha_n}$.

2. *If $P \neq P_{\alpha_\star}$, i.e. if $\mathcal{H}_{0,\mathcal{M}}$ is not true, then $\mathcal{T}_n(\mathcal{M}, P)$ tends to infinity in probability.*

The theorem shows how the problems of other papers and their approaches can be solved. The estimates for the variance will be discussed in the next chapter.
The algorithm for model specification is described in Algorithm 1. They also showed a similar asymptotic behavior for their test in case of model selection.

Theorem 2. *Assume that $\epsilon_n \to 0$, $\epsilon_n\sqrt{n} \to \infty$ in probability, $\alpha_\star \in \mathrm{argmin}_{\alpha\in\Theta_1} \mathrm{MMD}^2(P_\alpha, P)$ and $\beta_\star \in \mathrm{argmin}_{\beta\in\Theta_2} \mathrm{MMD}^2(Q_\beta, P)$, $\sqrt{n}(\alpha_n - \alpha_\star) = O_{\mathbb{P}}(1)$ and $\sqrt{n}(\beta_n - \beta_\star) = O_{\mathbb{P}}(1)$, the samples $(U_i)_{i\geq 1}$ and $(V_i)_{i\geq 1}$ are independent and that some technical Assumptions as in the BFM paper are satisfied by the competing models $\mathcal{M}_1$ and $\mathcal{M}_2$.*

Algorithm 1: MMD-based test of $\mathcal{H}_{0,\mathcal{M}} : \mathrm{MMD}(P_{\alpha_\star}, P) = 0$

Requirements: I.i.d. sample $(X_i)_{1\le i\le n}$ from P, generative model $F(U;\alpha) \sim P_\alpha$, estimator α_n of $\alpha_\star$, tuning parameter ϵ_n and confidence level γ.

1 Sample $\big(F(U_i,\alpha_n)\big)_{1\le i\le n}$, where $(U_i)_{1\le i\le n} \overset{i.i.d.}{\sim} P_U$;

2 Compute $\mathcal{T}_n(\mathcal{M}, P) = \sqrt{n}\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P)/(\tilde{\sigma}_{\alpha_n} + \epsilon_n \tilde{\sigma}_{q,\alpha_n})$;

3 Reject $\mathrm{MMD}(P_{\alpha_\star}, P) = 0$ when $\big|\mathcal{T}_n(\mathcal{M}, P)\big| > \Phi^{-1}(1-\gamma/2)$; otherwise, accept.

Algorithm 2: MMD based test of $\mathcal{H}_{0,\mathcal{M}_1,\mathcal{M}_2} : \mathrm{MMD}(P_{\alpha_\star}, P) = \mathrm{MMD}(Q_{\beta_\star}, P)$

Requirements: I.i.d. sample $(X_i)_{1\le i\le n}$ from P, generative models $F(U;\alpha) \sim P_\alpha$ and $G(V,\beta) \sim Q_\beta$, estimator α_n of $\mathrm{argmin}_{\alpha\in\Theta_1} \mathrm{MMD}(P_\alpha, P)$, estimator β_n of $\mathrm{argmin}_{\beta\in\Theta_2} \mathrm{MMD}(Q_\beta, P)$, tuning parameter ϵ_n and confidence level γ.

1 Sample $\big(F(U_i,\alpha_n)\big)_{1\le i\le n}$ and $\big(G(V_i,\beta_n)\big)_{1\le i\le n}$, where $(U_i)_{1\le i\le n} \overset{i.i.d.}{\sim} P_U$ and $(V_i)_{1\le i\le n} \overset{i.i.d.}{\sim} P_V$ are independent;

2 Compute $\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P) = \sqrt{n}\big\{\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}^2_{\epsilon_n}(Q_{\beta_n}, P)\big\}/(\tilde{\sigma}_{\alpha_n,\beta_n} + \epsilon_n \tilde{\sigma}_{q,\alpha_n,\beta_n})$;

3 Reject $\mathrm{MMD}(P_{\alpha_\star}, P) = \mathrm{MMD}(Q_{\beta_\star}, P)$ when $\big|\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P)\big| > \Phi^{-1}(1-\gamma/2)$; otherwise, accept.

1. *Under* $\mathcal{H}_{0,\mathcal{M}_1,\mathcal{M}_2} : \mathrm{MMD}(P_{\alpha_\star}, P) = \mathrm{MMD}(Q_{\beta_\star}, P)$, *we have*

$$\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P) = \sqrt{n}\frac{\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}^2_{\epsilon_n}(Q_{\beta_n}, P)}{\hat{\tau}_n} \xrightarrow{law} \mathcal{N}(0,1),$$

where $\hat{\tau}_n = \tilde{\sigma}_{\alpha_n,\beta_n} + \epsilon_n \tilde{\sigma}_{q,\alpha_n,\beta_n}$.

2. *If* $\mathrm{MMD}(P_{\alpha_\star}, P) > \mathrm{MMD}(Q_{\beta_\star}, P)$, *then*

$$\sqrt{n}\frac{\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}^2_{\epsilon_n}(Q_{\beta_n}, P)}{\hat{\tau}_n} \to +\infty \quad \textit{in probability.}$$

3. *If* $\mathrm{MMD}(P_{\alpha_\star}, P) < \mathrm{MMD}(Q_{\beta_\star}, P)$, *then*

$$\sqrt{n}\frac{\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}^2_{\epsilon_n}(Q_{\beta_n}, P)}{\hat{\tau}_n} \to -\infty \quad \textit{in probability.}$$

The algorithm of the BFM test for model selection is shown in Algorithm 2.

5 Estimates of Variance

The goal is to obtain a estimator of the asymptotic variance of $\sqrt{n}\widehat{\mathrm{MMD}}^2_{\epsilon_n}(P_{\alpha_n}, P)$. To achieve this we split it into two estimations, one of $\sqrt{n}\widehat{\mathrm{MMD}}^2(P_{\alpha_n}, P)$ and one of $\sqrt{n}\widehat{\mathrm{MMD}}^2_q(P_{\alpha_n}, P)$, which we will then combine. It is well known that the asymptotic variance of $\sqrt{n}\big\{\widehat{\mathrm{MMD}}^2(P_\alpha, P) - \mathrm{MMD}^2(P_\alpha, P)\big\}$ is

$$\sigma^2_\alpha := \mathrm{Var}\Big(2\tilde{h}\big(X, F(U;\alpha);\alpha\big)\Big).$$

The empirical estimate of it is

$$\tilde{\sigma}^2_\alpha := \frac{4}{n}\sum_{i=1}^{n}\left\{\frac{1}{n-1}\sum_{\substack{j=1\\ i\neq j}}^{n} h\Big(\big(X_i, F(U_i;\alpha)\big), \big(X_j, F(U_j;\alpha)\big)\Big) - \widehat{\mathrm{MMD}}^2(P_\alpha, P)\right\}^2.$$

Since our goal is to estimate $\sigma^2_{\alpha_\star}$ but $\alpha_\star$ is unknown, we replace $\alpha_\star$ with α_n and define an estimator of $\sigma^2_{\alpha_\star}$ by

$$\tilde{\sigma}^2_{\alpha_n} := \frac{4}{n}\sum_{i=1}^{n}\left\{\frac{1}{n-1}\sum_{\substack{j=1\\ i\neq j}}^{n} h\Big(\big(X_i, F(U_i;\alpha_n)\big), \big(X_j, F(U_j;\alpha_n)\big)\Big) - \widehat{\mathrm{MMD}}^2(P_{\alpha_n}, P)\right\}^2.$$

For the other asymptotic variance of the MMD_q-part, we define some shorthand notation:

$$\begin{aligned} q\big([\boldsymbol{x}, \boldsymbol{u}]_{1:4};\alpha\big) \;:=\; & k\big(x_1, x_3\big) - k\big(x_4, F(u_2;\alpha)\big) - k\big(x_2, F(u_4;\alpha)\big) \\ & + k\big(F(u_1;\alpha), F(u_3;\alpha)\big). \end{aligned} \tag{12}$$

Similarly, the asymptotic variance of $\sqrt{n}\big\{\widehat{\mathrm{MMD}}^2_q(P_\alpha, P) - \mathrm{MMD}(P_\alpha, P)\big\}$ for any $\alpha \in \Theta_1$ is

$$\sigma^2_{q,\alpha} := \mathrm{Var}\left(2\sqrt{2}\mathbb{E}_{[\boldsymbol{X},\boldsymbol{U}]_{3:4}}\big[q\big([\boldsymbol{X}, \boldsymbol{U}]_{1:4};\alpha\big)\big]\right).$$

Analogously, this allows to define an estimator of $\sigma^2_{q,\alpha_\star}$ via

$$\tilde{\sigma}^2_{q,\alpha_n} := \frac{16}{n}\sum_{i=1}^{n/2}\left\{\frac{1}{n/2-1}\sum_{\substack{j=1\\ i\neq j}}^{n/2} q\Big([\boldsymbol{X}, \boldsymbol{U}]_{2i-1,2i,2j-1,2j};\alpha_n\Big) - \widehat{\mathrm{MMD}}^2(P_{\alpha_n}, P)\right\}^2.$$

Notice that both estimators are always non-negative and of computational complexity $O(n^2)$. Additionally, $\sigma^2_{\alpha_\star} = 0$ when $P_{\alpha_\star} = P$, but $\sigma^2_{q,\alpha_\star}$ is always strictly positive under Assumption 2 of the BFM paper. In the BFM test statistic they combine those two estimators into $\hat{\sigma}_n = \tilde{\sigma}_{\alpha_n} + \epsilon_n\tilde{\sigma}_{q,\alpha_n}$.

6 Implementation in Python

For the programming part of the internship, the provided R code of Professor Min and a GitHub repository by https://github.com/sshekhar17/PermFreeMMD used for the kernel and MMD-z calculation, was used as a starting point. The first step was to research all the functions used in the R code and find equivalent packages in Numpy or Scipy, and replicate the exact logic, including also the for loops of the R code. To achieve this step, each function in the BFM paper was mapped to the R code to simplify the translation to Python code and further improvements, such as parallelisation. For the kernel function and the MMD calculation, the code of sshekhar and some parallelised version of MMD calculation using the cdist function, which calculates the distance between each point into a matrix, was used. This was quite a challenge since the MMD implementation of sshekhar was quite different from Professor Mins' R code, and also the logic was different. There was a need to exclude the cross-similarity diagonal elements, and also for the kernel function, the inputs theta1 and theta2 were changed to bandwidth and amplitude, where bandwidth is bw=sqrt(theta1/2). Additionally there were also included some safety nets to avoid unnecessary errors, such as making sure inputs are Numpy arrays with the right dimensions or that some inputs must be even integers. For the simulation study, there was a need to include functions to generate the data. Those functions use the Numpy Random Number Generation default setting to generate some normal distributed matrices. For the simulation study, the goal was to recreate four pictures with my code and run 1000 simulations for each of the varying parameters. Using cdist and numpy, it was possible to increase the performance compared to the R code by a factor of around 60. As I was curious if I could further improve my code to make it run faster, I also updated my code to use a GPU instead of a CPU. For this, I had to install the CUDA package provided by Nvidia and use the package Cupy instead of Numpy wherever I used Numpy. For this part, I had to make sure that each function I was using in Numpy is available in Cupy, which wasn't the case. So I had to rewrite some functions using the Numpy package documentation to recreate those functions using Cupy syntax. I had to do the same for cdist of the Scipy package. Of course, I only changed components which are computationally expensive, so rewriting Random Number Generation in Cupy wouldn't be necessary. Using the GPU had another speedup of around 70-80, so in total, compared to the R code, the GPU version is around 4200-4800 times faster. For the simulation study, I didn't use the R code provided at all, I only used section 5 of the BFM paper. For all figures the underlying sample X is

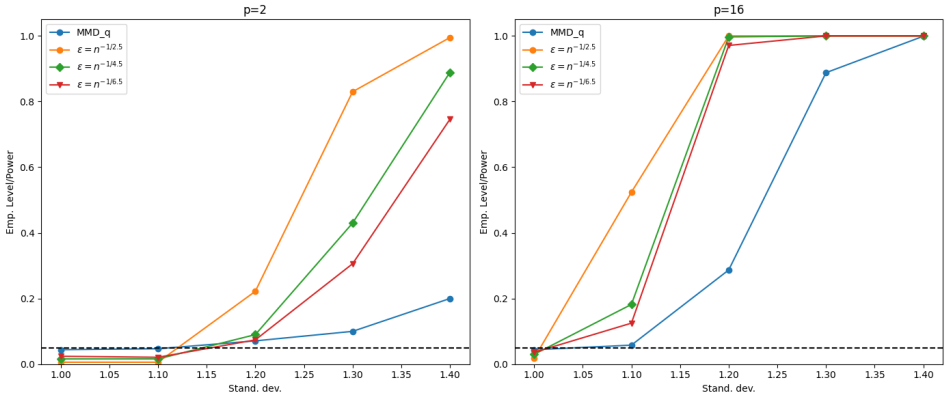

Figure 1: Empirical level and power of the tests based on $\mathcal{T}_n(\mathcal{M}, P)$ and $\sqrt{n}\widehat{\mathrm{MMD}}_q^2(P_{\alpha_n}, P)$ for dimensions $p = 2$ (left) and $p = 16$ (right), a sample size $n = 500$, as well as for different choices of ϵ_n (see the legend) and varying ***standard deviation***. The rejection probabilities are estimated using 1000 replications of the tests based on samples of size n. The black dashed line indicates the significance level 0.05.

generated using $X \sim P = \mathcal{N}(0, I_p)$, where I_p is the $p-$dimensional identity matrix.

6.1 Simulation for Figure 1

For figure 1, I needed to create a normal distribution with varying standard deviation, calculate the BFM-test for each simulation and count the rejects out of those 1000 simulations, same for the different epsilon. To generate the sample Y_1 , consider

$$Y_1(\alpha) = Y + \alpha \sim P_\alpha, \text{ with } Y \sim \mathcal{N}\big(0, \sigma^2 I_p\big),$$

for some known variance σ^2, and $\alpha := (\alpha_1, \ldots, \alpha_p)$ is a p-dimensional vector to be estimated. For every σ^2, the "optimal" parameter is $\alpha_\star = 0$. Moreover, $P = P_{\alpha_\star}$ if $\sigma^2 = 1$.

6.2 Simulation for Figure 2

For figure 2, epsilon is now only one value, and the mean is varying, also the sample sizes n are varying. Again, I needed to count the output rejects of the BFM test and plot those.

$$Y_1(\sigma) = \alpha_0 \mathbf{1} + diag(\sigma_1, \ldots, \sigma_p) Y \sim P_\sigma, ; \; Y \sim \mathcal{N}(0, I_p)$$

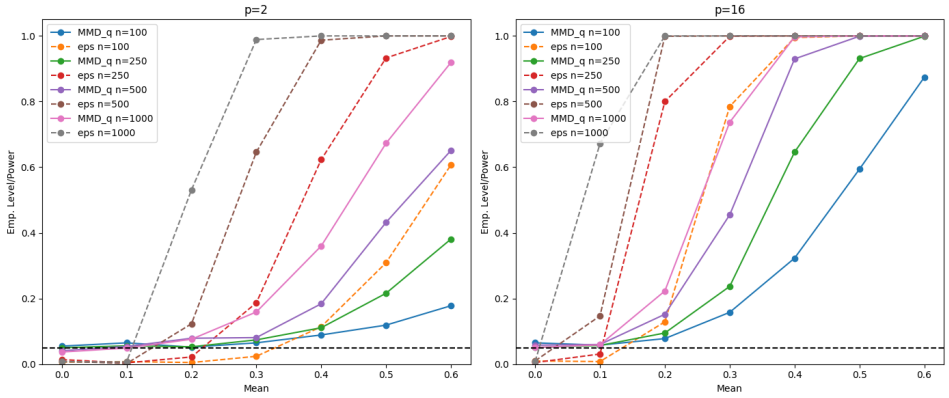

Figure 2: Empirical level and power of the tests based on $\mathcal{T}_n(\mathcal{M}, P)$ and $\sqrt{n}\widehat{\mathrm{MMD}}_q^2(P_{\sigma_n}, P)$ for dimensions $p = 2$ (left) and $p = 16$ (right) as well as for sample sizes $n = 100, 250, 500, 1000$ (see the legend), $\epsilon_n = n^{-1/2.5}$ and varying ***mean***. The rejection probabilities are estimated using 1000 samples of size n. The black dashed line indicates the significance level 0.05.

for some pre-specified marginal mean $\alpha_0 \in \mathbb{R}$, where the marginal standard deviations are $\sigma_1, \ldots, \sigma_p$. We set $\sigma := (\sigma_1, \ldots, \sigma_p)$ and $\mathbf{1} = (1, \ldots, 1)$. If we fix $\alpha_0 = 0$, then the "optimal" parameters are $\sigma_1^* = \ldots = \sigma_p^* = 1$ and $P = P_{\sigma^*}$, where $\sigma^* = (\sigma_1^*, \ldots, \sigma_p^*)$. Now, we vary the mean α_0 of the competing model $Y(\sigma)$ by setting $\alpha_0 \in \{0, 0.1, 0.2, \ldots, 0.6\}$. Furthermore, we estimate σ_j by the empirical standard deviation of the j–th marginal i.i.d. sample from P, namely $\sigma_{j,n}^2 = n^{-1} \sum_{i=1}^n (X_{ij} - \overline{X}_{ij})^2$, and set $\sigma_n := (\sigma_{1,n}, \ldots, \sigma_{p,n})$.

6.3 Simulation for Figure 3

For figure 3, the BFM test now has model-selection=True, and we need to generate three data sets instead of two. X is standard normal, Y_1=Y is for the first half standard normal and then for a varying standard deviation between 1.0 and 1.4, and Y_2=Z is also standard normal, same as X. That would be the degenerate U-statistic case. The first model $\mathcal{M}_1$ is defined by

$$Y(\alpha) = Y + \alpha \sim P_\alpha, ; \; Y \sim \mathcal{N}\left(0, diag(1, \ldots, 1, \sigma^2, \ldots, \sigma^2)\right),$$

for some pre-specified variance σ^2, where $\alpha = (\alpha_1, \ldots, \alpha_p)$. Thus, the first $p/2$ margins of $Y(\alpha)$ have variance 1 and the remaining $p/2$ margins have variance σ^2. If $\sigma^2 = 1$, the model $\mathcal{M}_1$ coincides with the true model when α equals the "optimal" parameter $\alpha_\star = 0$. The second model $\mathcal{M}_2$ is defined by

$$Z(\beta) = Z + \beta \sim Q_\beta ; \; Z \sim \mathcal{N}(0, I_p) ,$$

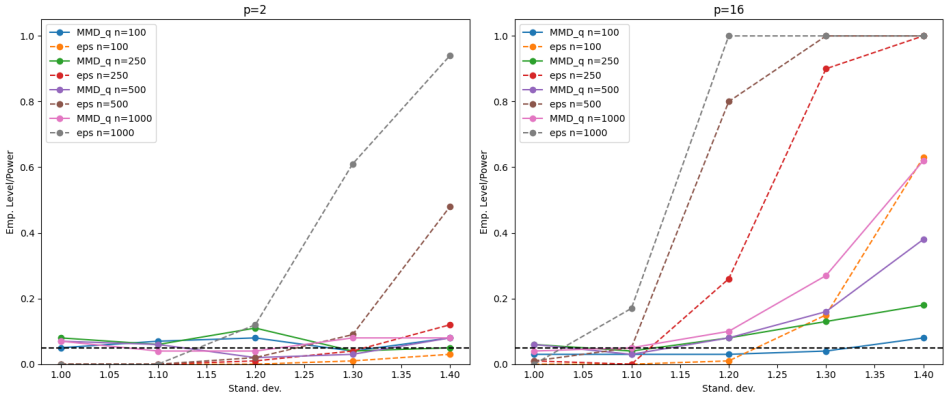

Figure 3: **Degenerate case** for comparison of two models: Empirical level and power of the tests based on $\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P)$ and $\sqrt{n}\big(\widehat{\mathrm{MMD}}_q^2(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}_q^2(Q_{\beta_n}, P)\big)$ for dimensions $p = 2$ (left) and $p = 16$ (right) as well as for sample sizes $n = 100, 250, 500, 1000$ (see the legend), $\epsilon_n = n^{-1/2.5}$ and varying ***standard deviation*** σ in Model $\mathcal{M}_1$. Model $\mathcal{M}_2$ coincides with the true model ($\beta = 0$). The rejection probabilities are estimated using 1000 replications of the tests based on samples of size n. The black dashed line indicates the significance level 0.05.

where $\beta = (\beta_1, \ldots, \beta_p)$. If $\beta = 0$, the model $\mathcal{M}_2$ also coincides with the law of the DGP. Therefore, we may be in the degenerate situation, when the two competing models with optimal parameters coincide with the law of the DGP. As in the first example, we vary the standard deviation σ by setting $\sigma \in \{1.0, 1.1, 1.2, 1.3, 1.4\}$. Further, we estimate α and β by the empirical mean of the i.i.d. sample from P, $\alpha_n = \beta_n = n^{-1} \sum_{i=1}^n X_i$.

6.4 Simulation for Figure 4

For figure 4, X is the same as in figure 3, but now instead of having a standard deviation of 1.0, Y1 has one of 1.2 for the first half and then 1.2 until 1.6, varying, and Y2 always has 1.2 standard deviation. For figures 1 and 2, where we test for model specification, the outputs of the BFM test to look at are the spec-reject and spec-reject-q ones, and for figures 3 and 4, where we test for model selection, the outputs of the BFM test to look at are the select-reject and select-reject-q. Now, the models $\mathcal{M}_1$ and $\mathcal{M}_2$ are given by

$$Y(\alpha) = Y + \alpha \sim P_\alpha, ; \; Y \sim \mathcal{N}\left(0, diag(1.2^2, \ldots, 1.2^2, \sigma^2, \ldots, \sigma^2)\right), \text{ and}$$

$$Z(\beta) = Z + \beta \sim Q_\beta ; \; Z \sim \mathcal{N}\left(0, 1.2^2 I_p\right),$$

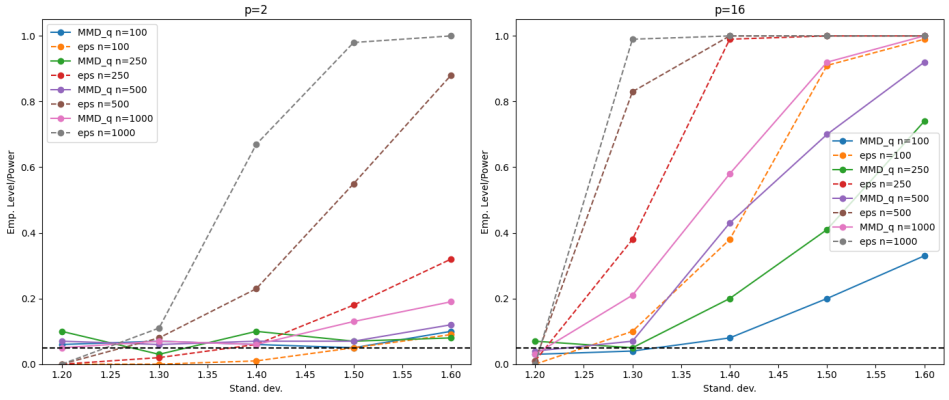

Figure 4: **Non-degenerate case** for comparison of two models: Empirical level and power of the tests based on $\mathcal{T}_n(\mathcal{M}_1, \mathcal{M}_2, P)$ and $\sqrt{n}\big(\widehat{\mathrm{MMD}}_q^2(P_{\alpha_n}, P) - \widehat{\mathrm{MMD}}_q^2(Q_{\beta_n}, P)\big)$ for dimensions $p = 2$ (left), $p = 16$ (right) as well as for sample sizes $n = 100, 250, 500, 1000$ (see the legend), $\epsilon_n = n^{-1/2.5}$ and varying ***standard deviation*** in Model $\mathcal{M}_1$. Both models do not coincide with the true model. The rejection probabilities are estimated using 1000 replications of the tests based on samples of size n. The black dashed line indicates the significance level 0.05.

respectively. Thus, both models cannot coincide with the DGP, reflecting the non-degenerate case. However, for $\sigma = 1.2$, they coincide and are therefore equally far away from the DGP. We vary the standard deviation σ by setting $\sigma \in \{1.2, 1.3, 1.4, 1.5, 1.6\}$.

6.5 Further potential improvements

To make further progress to improve the code of the BFM test, it would require getting some additional logic to avoid memory-bound errors with the GPU, since this test would be used for n=1 million or larger. I also considered probabilistic techniques since, for most of the code, approximate values are sufficient and could improve performance substantially.

7 References

Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schoelkopf, and Alex Smola. A kernel method for the two-sample-problem. In Advances in Neural Information Processing Systems, volume 19. Curran Associates, Inc., 2006.

Florian Brück, Jean-David Fermanian, and Aleksey Min. Distribution free MMD tests for model selection with estimated parameters. arXiv preprint arXiv:2305.07549, 2024.

master thesis of Tobias Solfronk: Maximum Mean Discrepancy for Model Comparisons.