Learning and Testing Powerful Hypotheses

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Jonas Matthias Kübler

aus Filderstadt

Tübingen

2022

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	17.07.2023
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter/-in:	Prof. Dr. Robert Williamson
2. Berichterstatter/-in:	Prof. Dr. Bernhard Schölkopf

Learning and Testing Powerful Hypotheses

Jonas Matthias Kübler

2022

Abstract

Progress in science is driven through the formulation of hypotheses about phenomena of interest and by collecting evidence for their validity or refuting them. While some hypotheses are amenable to deductive proofs, other hypotheses can only be accessed in a data-driven manner. For most phenomena, scientists cannot control all degrees of freedom and hence data is often inherently stochastic. This stochasticity disallows to test hypotheses with absolute certainty. The field of statistical hypothesis testing formalizes the probabilistic assessment of hypotheses, enabling researchers to control the error rates, for example, at which they reject a true hypothesis, while aiming to reject false hypotheses as often as possible.

But how do we come up with promising hypotheses, and how can we test them efficiently? Can we use machine learning systems to automatically generate promising hypotheses? This thesis studies different aspects of this question.

A simple rule for statistical hypothesis testing states that one should not peek at the data when formulating a hypothesis. This is indeed true if done naively, that is, when the hypothesis is then simply tested with the data as if one had not looked at it yet. However, we show that in principle using the same data for learning the hypothesis and testing it is feasible if we can correct for the selection of the hypothesis. We treat this in the case of the two-sample problem. Given two samples, the hypothesis to be tested is whether the samples originate from the same distribution. We can reformulate this by testing whether the maximum mean discrepancy over a (unit ball of a) reproducing kernel Hilbert space is zero. We show that we can learn the kernel function, hence the exact test we use, and perform the test with the same data, while still correctly controlling the Type-I error rates. Likewise, we demonstrate experimentally that taking all data into account can lead to more powerful testing procedures than the data splitting approach. However, deriving the formulae that correct for the selection procedure requires strong assumptions, which are only valid for a specific, the linear-time, estimate of the maximum mean discrepancy. In more general settings it is difficult, if not impossible, to adjust for the selection.

We thus also analyze the case where we split the data and use part of it to learn a test statistic. The maximum mean discrepancy implicitly optimizes a mean discrepancy over the unit ball of a reproducing kernel Hilbert space, and often the kernel itself is optimized on held-out data. We instead propose to optimize a witness function directly on held-out data and use its mean discrepancy as a test statistic. This allows us to directly maximize the test power, simplifies the theoretical treatment, and makes testing more efficient. We provide and implement algorithms to learn the test statistics. Furthermore, we show analytically that the optimization objective to learn powerful tests for the two-sample problem is closely related to the objectives used in standard supervised learning tasks, namely the least-square loss and cross-entropy loss. This allows us to indeed use existing machine learning tools when learning powerful hypotheses. Furthermore, since we use held-out data for learning the test statistic, we can use any kind of model-selection and cross-validation techniques to maximize the performance. To facilitate this for practitioners, we provide an open-source Python package `'autotst'` implementing an interface to existing libraries and running the whole testing pipeline, including the learning of the hypothesis. Our presented methods reach state-of-the-art performance on two-sample testing tasks. We also show how to trade off the computational resources required for the test by sacrificing some statistical power, which can be important in practice. Furthermore, our test easily allows interpreting the results.

Having more computational power potentially allows extracting more information from data and thus obtain more significant results. Hence, investigating whether quantum computers can help in machine learning tasks has gained popularity over the past years. We investigate this in light of the two-sample problem. We define the quantum mean embedding, mapping probability distributions onto quantum states, and analyze when this mapping is injective. While this is conceptually interesting on its own, we do not find a straight-forward way of harnessing any speed-up. The main problem here is that there is no known way to efficiently create the quantum mean embedding. On the contrary, fundamental results in quantum information theory show that this might generally be hard to do.

For two-sample testing, the usage of reproducing kernel Hilbert spaces has been established for many years and proven important both theoretically and practically. In this case, we thus focused on practically relevant aspects to make the tests as powerful and easy to use as possible. For other hypothesis testing tasks, the usage of advanced machine learning tools still lags far behind. For specification tests based on conditional moment restrictions, popular in econometrics, we do the first steps by defining a consistent test based on kernel methods. Our test already has promising performance, but optimizing it, potentially with the other insights gained in this thesis, is an open task.

Zusammenfassung

The 'Zusammenfassung' is a machine translated version of the abstract via www.DeepL.com/Translator and slightly adopted.

Fortschritt in der Wissenschaft wird durch die Formulierung von Hypothesen über Phänomene von Interesse und durch das Sammeln von Evidenz für deren Gültigkeit oder deren Widerlegung erzielt. Während einige Hypothesen deduktiv bewiesen werden können, lassen sich andere Hypothesen nur auf datengestützte Weise überprüfen. Bei den meisten Phänomenen können Wissenschaftler nicht alle Freiheitsgrade kontrollieren, und daher sind die Daten oft von Natur aus stochastisch. Diese Stochastik macht es unmöglich, Hypothesen mit absoluter Gewissheit zu prüfen. Der Feld der statistischen Hypothesentests formalisiert die probabilistische Bewertung von Hypothesen und ermöglicht es Forscher:innen, beispielsweise die Fehlerquoten zu kontrollieren, mit denen sie eine wahre Hypothese ablehnen, während sie versuchen, falsche Hypothesen so oft wie möglich abzulehnen.

Aber wie kommen wir zu vielversprechenden Hypothesen und wie können wir sie effizient testen? Können wir maschinelle Lernsysteme einsetzen, um automatisch vielversprechende Hypothesen zu generieren? In dieser Arbeit werden verschiedene Aspekte dieser Frage untersucht.

Eine einfache Regel für statistische Hypothesentests besagt, dass man bei der Formulierung einer Hypothese nicht auf die Daten schauen sollte. Das stimmt tatsächlich, wenn man es naiv macht, d. h. wenn man die Hypothese einfach mit den selben Daten testet, als hätte man sie noch nicht angeschaut. Wir zeigen jedoch, dass es prinzipiell möglich ist, dieselben Daten zum Lernen der Hypothese und zum Testen derselben zu verwenden, wenn wir für Auswahl der Hypothese korrigieren können. Wir behandeln dies am Beispiel des Zwei-Stichproben-Problems. Gegeben zwei Stichproben besteht die zu prüfende Hypothese darin, ob die Stichproben aus der gleichen Verteilung stammen. Wir können dies umformulieren, indem wir testen, ob die maximale mittlere Diskrepanz über einem (Einheitsball eines) reproduzierenden Kernel-Hilbert-Raums Null ist. Wir zeigen, dass wir die Kernel-Funktion lernen können, also den genauen Test, den wir verwenden, und den Test mit denselben Daten durchführen können, während wir die Typ-I-Fehlerraten immer noch korrekt kontrollieren. Ebenso zeigen wir experimentell, dass die Berücksichtigung aller Daten zu leistungsfähigeren Testverfahren führen kann als der Ansatz der Datenaufteilung. Die Ableitung der Formeln, die für das Auswahlverfahren korrigieren, erfordert jedoch strenge Annahmen, die nur für eine bestimmte, nämlich die linear-skalierende Schätzung der maximalen mittleren Diskrepanz gültig sind. In allgemeineren Situationen ist es schwierig, wenn nicht gar unmöglich, die Auswahl zu korrigieren.

Wir analysieren daher auch den Fall, dass wir die Daten aufteilen und einen Teil davon zum Lernen einer Teststatistik verwenden. Die maximale mittlere Diskrepanz optimiert implizit eine mittlere Diskrepanz über die Einheitskugel eines reproduzierenden Kernel-Hilbert-Raums, und oft wird der Kernel selbst auf separaten Daten optimiert. Wir schlagen stattdessen vor, eine Zeugenfunktion direkt auf separaten Daten zu optimieren und ihre mittlere Diskrepanz als Teststatistik zu verwenden. Dies ermöglicht eine direkte Maximierung der Testleistung, vereinfacht die theoretische Behandlung und macht das Testen effizienter. Wir stellen Algorithmen zum Lernen der Teststatistiken bereit und implementieren sie. Darüber hinaus zeigen wir analytisch, dass das Zielfunktion zum Erlernen leistungsfähiger Tests für das Zwei-Stichproben-Problem eng mit den Zielfunktionen verwandt ist, die bei Standardaufgaben des überwachten Lernens verwendet werden, nämlich der kleinsten quadratischen Abweichung und der logistischen Regression. Dies ermöglicht es uns, beim Lernen leistungsfähiger Hypothesen tatsächlich bestehende maschinelle Lernwerkzeuge zu verwenden. Da wir für das Lernen der Teststatistiken separate Daten verwenden, können wir außerdem alle Arten von Modellauswahl- und Kreuzvalidierungstechniken einsetzen, um die Leistung zu maximieren. Um dies den Praktikern zu erleichtern, stellen wir ein quelloffenes Python-Paket "`autotst`" zur Verfügung, das eine Schnittstelle zu bestehenden Bibliotheken implementiert und die gesamte Testpipeline, einschließlich des Lernens der Hypothese, ausführt. Die von uns vorgestellten Methoden erreichen die beste Leistung bei Zwei-Stichproben-Tests. Wir zeigen auch, wie man die für den Test erforderlichen Rechenressourcen durch den Verzicht auf eine gewisse statistische Aussagekraft verringern kann, was in der Praxis wichtig sein kann. Außerdem lassen sich die Ergebnisse unseres Tests leicht interpretieren.

Mit mehr Rechenleistung lassen sich potenziell mehr Informationen aus den Daten extrahieren und somit aussagekräftigere Ergebnisse erzielen. Daher hat die Untersuchung, ob Quantencomputer bei Aufgaben des maschinellen Lernens helfen können, in den letzten Jahren an Popularität gewonnen. Wir untersuchen dies im Hinblick auf das Zwei-Stichproben-Problem. Wir definieren das Quantum Mean Embedding, das Wahrscheinlichkeitsverteilungen auf Quantenzustände abbildet, und analysieren, wann diese Abbildung injektiv ist. Obwohl dies für sich genommen konzeptionell interessant ist, finden wir keinen einfachen Weg, um einen Rechenzeitvorteil zu nutzen. Das Hauptproblem dabei ist, dass es keine bekannte Methode gibt, das quantum mean embedding effizient zu erstellen. Im Gegenteil, grundlegende Ergebnisse der Quanteninformationstheorie zeigen, dass dies im Allgemeinen schwer zu bewerkstelligen sein dürfte.

Für Zwei-Stichproben-Tests hat sich die Verwendung von reproduzierenden Kern-Hilbert-Räumen seit vielen Jahren etabliert und sowohl theoretisch als auch praktisch als wichtig erwiesen. In diesem Fall haben wir uns daher auf praktisch relevante Aspekte konzentriert, um die Tests so leistungsfähig und benutzerfreundlich wie möglich zu gestalten. Bei anderen Hypothesentests hinkt der Einsatz von fortgeschrittenen Methoden des maschinellen Lernens noch weit hinterher. Für Spezifikationstests auf der Grundlage von bedingten Momentenrestriktionen, die in der Ökonometrie weit verbreitet sind, haben wir die ersten Schritte unternommen, indem wir einen konsistenten Test auf der Grundlage von Kernelmethoden definiert haben. Unser Test hat bereits eine vielversprechende Leistung, aber seine Optimierung, möglicherweise mit den anderen in dieser Arbeit gewonnenen Erkenntnissen, ist eine offene Aufgabe.

Thank You

I am deeply thankful that I have been able to work on this thesis and other research projects over the past four years. Particularly, I am thankful to our society for valuing basic research and the trust put in researchers like me. There are many individuals that I would like to thank directly:

Bernhard Schölkopf for trusting in my abilities as a researcher and pushing me both to think big and to bring concrete ideas to paper. It is hard to imagine a place where I could have set my research agenda more freely and still have complete support and great advice from my supervisor. Initially, that freedom was slightly overwhelming, and I am grateful to *Krikamol Muandet* for providing a close supervision at the beginning of my PhD journey and all the projects we worked on together. I thank *Philipp Hennig* and *Caterina De Bacco* for serving on my thesis advisory committee and their productive feedback, and Philipp Hennig particularly also for help in navigating the PhD expectations. I am grateful to *Robert Williamson* for reviewing this thesis.

Many projects were only possible because of my great collaborators. *Wittawat Jitkrittum* and *Simon Buchholz* always had an open door and were willed to discuss my arbitrary and often half-baked ideas. I am truly impressed that after some minutes of discussion, they often understood my ideas better than I had. Especially towards the end of the past four years, I realized how much fun it is to closely collaborate on projects and thank *Vincent Stimper, Luigi Gresele, Julius von Kügelgen, and Simon Buchholz* (once more) for this. I also thank all my collaborators on projects in quantum machine learning. Although some of them did not end up in this thesis, it was always great fun to explore this new area together.

I would also like to thank *Daniel Braun*, who supervised my Bachelor and Master theses, for laying the foundation of my research path. And *Dominik Janzing* without whose talk on quantum causality I might have never thought about doing a PhD at the MPI.

At times, doing a PhD can also be pretty tough. But if you have colleagues (that then become friends) like *Sebastian Weichwald* that take you on board and accompany you through this journey, coming to work is always a great pleasure no matter how stuck a project seems. This also holds true for all other colleagues from the *Empirical Inference* department. Also, many thanks to our incredibly supportive and qualified administrative staff, in particular *Sabrina Rehbaum*, whom I always first asked when I did not know better.

I always enjoyed coming home after work and living with a diverse group of people, having dinner together, playing games and sharing part of our life. Shoutout to the *hardy 66* crew, and especially to *Eric* and *Federica* for a record-breaking 7(?) years of living together. Thank you *Annika* for sharing the past years with me, climbing many mountains, and discussing what is needed for a good life. Lastly, my *parents* and my *brother* for their continuous support in all facets of life.

Danke.

Contents

Introduction

0.1. Introduction

Imagine you would have to asses whether a (six-sided) die is fair, i.e., the probability of all outcomes is equally given as 1/6. What is a good strategy to test the *null hypothesis* H_0: 'the die is fair' against the *alternative hypothesis* 'the die is unfair'? How can we make this quantitative? What errors could we make?

Suppose we are allowed to role the die 20 times. A conceptually very simple approach would be to role the die ten times, look at the outcomes, and formulate a concise hypothesis about what is unfair. Then role the die another ten times, and test whether this is indeed statistically significant.

Let's do the experiment: The first ten roles yield ⚄, ⚃, ⚀, ⚂, ⚂, ⚅, ⚂, ⚂, ⚂, ⚂. Without much knowledge about statistics an intuitive and concise alternative hypothesis would be 'the probability of obtaining a ⚂ is larger than 1/6'. We then role the die another ten times and obtain ⚄, ⚂, ⚂, ⚀, ⚅, ⚄, ⚂, ⚃, ⚂, ⚅. Denoting by K the number of sixes, we observe $K = 4$. Is this statistically significant? To check this, we can look at the distribution under the null hypothesis, where we have that the probability of rolling a ⚂ is 1/6. Under the null hypothesis K follows a binomial distribution, i.e.,

$$\Pr[K = k] = \binom{10}{k} \left(\frac{1}{6}\right)^k \left(\frac{5}{6}\right)^{10-k}.$$

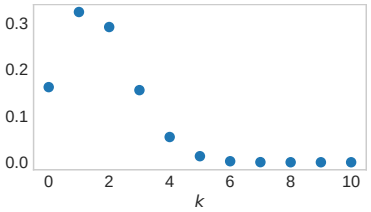


Figure 1.: Probability mass function for observing k sixes when rolling a fair die ten times.

Thus the p-value[1] is given as $\Pr[K \geq 4] \approx 7\%$. The question of whether or not to reject the null hypothesis now depends on two types of error that we need to balance:

Type-I error: We reject H_0 although it was true.
Type-II error: We do not reject H_0 although it was false.

In this thesis our goal is to devise tests that control the Type-I error at (or below) a specified *significance level* α, oftentimes set to 5%,[2] and given Type-I error control the goal is to make the Type-II error rate as small as possible. We will also speak of *test power*, which is simply the rate of rejection given the null hypothesis is wrong, and thus simply 1 − rate of Type-II errors. In our example, given $\alpha = 5\%$ we could not reject the null hypothesis.[3]

Although, this thesis will not be about rolling dice, this small example nicely brings up questions that we will discuss in the following. Among others these are:

Is data splitting necessary for two-stage hypothesis tests?
How can we learn the most promising hypothesis in the first stage?
How applicable is a test for non-expert users?
How many computational resources does the test require?

Let us briefly comment on those four questions. Data splitting is *not necessary* but very convenient. Nevertheless, data is often split to prevent (accidental) 'p-value hacking'.[4] p-value hacking happens when one takes the same data to formulate a hypothesis (via a test statistic) and tests on the same data without adjusting for it. This leads to unreliable p-values and thus useless results. In the above example, this would have happened,

1: The p-value is defined as the probability under H_0 of observing a test statistic (here K) at least as large as the actually observed one.

2: 5% is an arbitrary choice that is common in the literature, but should be set depending on the practical application.

3: In fact this would be a Type-II error, as a ⚂ was rolled with probability 1/2 the way the data was generated.

Is data splitting necessary for two-stage hypothesis tests?

4: xkcd.com/882/ gives a nice illustration of p-value hacking.

if we tested the significance of 'the probability of obtaining a ⚄ is larger than 1/6' with the prescribed method, but on the first dataset. There exist methods to obtain reliable tests even without data splitting. A classic way is to use Bonferroni correction [1, Theorem 9.1.1]: we can simultaneously test whether any of the six outcomes has probability larger than 1/6 and adjust the significance level for each test to $\alpha/6$. A more modern approach is to derive a distribution of the test statistic *conditionally* on it being selected [2]. While the first approach can be quite conservative, especially with a large number of different tests, the latter requires strong analytic understanding of the problem. In this work, we will extend the second approach, but also fall back to data splitting, since this allows us to define more flexible tests.

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

[2]: Fithian et al. (2017), *Optimal Inference After Model Selection*

How can we learn the most promising hypothesis in the first stage?

In the die example, one could also uniformly at random pick one of the sides of the die in the first stage and then test whether its probability is too large on the second sample. Clearly, this does not make good use of the data of the first sample. Since we want to maximize the test power (minimize Type-II error), we should instead pick a hypothesis for which, based on the data in the first sample, we see the highest chances of the test rejecting on the second sample. If the hypotheses we consider are simply to pick one number and check its probability, then our choice of picking ⚄ was optimal; based on the data in the first sample it is most likely that checking probability of ⚄ is leads to rejection. However, if our set of hypotheses is not that simple, finding the optimal one might be challenging. In this work, we focus on strategies that optimize the *asymptotic* test power, i.e., are optimal in the large data regime.

How applicable is a test for non-expert users?

Arguably the testing procedure above is relatively simple and applicable with minimal statistical knowledge and tools. But what if learning the right hypothesis requires advanced machine learning tools? Most literature in hypothesis testing focuses on statistical and theoretical aspects, while the implementation details and practical aspects are often less worked out. Arguably, engineering matters just as much for hypothesis testing as it does for standard machine learning (say regression and classification tasks). One reason that the engineering part is not so popular, might be that it will usually require data splitting in order to prevent (accidental) p-value hacking. But if we split the data, we are on the safe side. The `autotst` Python package[5] developed during this doctoral studies, builds on existing automated machine learning techniques and uses them for two-sample testing. It requires minimal knowledge of the user and automates the engineering part when learning a powerful hypothesis.

5: github.com/jmkuebler/auto-tst

How much computational resources does the test require?

When working with larger datasets not only statistical but also computational considerations come into play. This will sometimes result in a trade-off between statistical aspects in form of test power and computational aspects like runtime and space requirements. As a bold example: simply ignoring half of the available data and running a test only on the other half will be faster but also (likely) lead to worse results. Fortunately, we will present more sophisticated aspects to balance this trade-off in this work. Ideally, an unexperienced user could easily state their available computational resources, which we also achieve in the `autotst` package, simply by integrating existing methods.

Two-sample tests. The major part of this thesis deals with a specific hypothesis testing problem: Testing whether two samples originate from the same distribution. This is an important problem in scientific discovery, for example, testing whether two differently treated groups of patients show different characteristics is a two-sample problem. More recently, the problem of detecting distribution shift has also been a use case of two-sample tests. Let us denote by $\mathbb{X}$ an i.i.d. sample drawn from the distribution P and by $\mathbb{Y}$ an i.i.d. sample drawn from distribution Q. The null hypothesis will then be $P = Q$. While there exist classical tests like Student's two-sample t-test or the Kolmogorov-Smirnov test, for complex datasets as in Figure 2 one needs more flexible approaches. In this thesis we will work with approaches that rely on machine learning and will now introduce two flexible approaches.

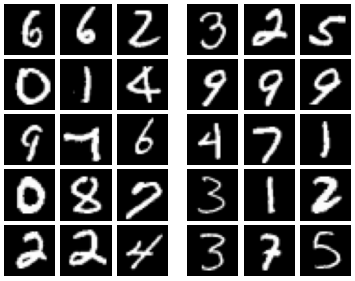


Figure 2.: The two-sample problem: Are the images on the left drawn from the same distribution as the images on the right? (Subsampled from MNIST [3].)

For two distributions P, Q, the Maximum Mean Discrepancy (MMD) is defined as

$$\mathrm{MMD}(P, Q \mid \mathcal{H}) = \max_{h \in \mathcal{H}, \|h\| \leq 1} \mathbb{E}_{X \sim P}\left[h(X)\right] - \mathbb{E}_{Y \sim Q}\left[h(Y)\right].$$

Here $\mathcal{H}$ is commonly a reproducing kernel Hilbert space (RKHS) [4]. [5] use an empirical estimate of the (squared) MMD as a test statistic for two-sample testing. One can think of the MMD as an implicit two-stage procedure, where first the function that maximizes the mean discrepancy is found and then its (squared) mean discrepancy is taken as a test statistic. However, in practice, one directly estimates its squared value. On the other hand, choosing a good RKHS $\mathcal{H}$ has also been done in a data-driven way, which then relied on data splitting once more [6–8] making MMD into an explicit two-stage procedure. Furthermore, computational-statistical trade-offs have also found consideration, since [5] propose both a linear- and a quadratic-runtime estimator. We will show that in the linear-time case, we can learn $\mathcal{H}$ even without data splitting. Furthermore, we will argue that if we use data-splitting, we should rather directly estimate a witness function whose mean-discrepancy has best test power, instead of learning a kernel and its associated RKHS.

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

[5]: Gretton et al. (2012), *A kernel two-sample test*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

An other way to test the two-sample hypothesis is to train a classifier between the two-samples and then to check whether its accuracy is significantly above chance [9]. This procedure is intrinsically tied to a two-stage procedure and to data splitting. While the MMD basically is its own branch of machine learning, the classifier two-sample test can use existing frameworks for classification. However, using a binary classifier was criticized for two reasons: First, binary outcomes are discrete and introduce significant noise in areas where the decision is uncertain. Second, optimizing classification accuracy does not really optimize test power.

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*

We will connect classification-based tests to tests based on the mean discrepancy of a witness function (similar to MMD), and will show that optimizing a cross-entropy loss and taking the predicted probabilities as witness function indeed optimizes asymptotic test power.

While most of the presented research of this thesis has a direct practical impact, we also investigated more fundamental questions in machine learning during my doctoral studies. In particular, we investigated the role and impact future quantum computers could have on machine learning

[10–13]. Most of this is not of direct relevance for this dissertation, except for [10]. We will thus also discuss in this work whether we can use quantum computers to speed-up two-sample tests. We will see that with the current state of the theory this does not seem to be the case.

[10]: Kübler et al. (2019), *Quantum mean embedding of probability distributions*; [11]: Kübler et al. (2020), *An adaptive optimizer for measurement-frugal variational algorithms*; [12]: Kübler* et al. (2021), *The inductive bias of quantum kernels*; [13]: Jerbi et al. (2023), *Quantum machine learning beyond kernel methods*

Other hypothesis testing problems. Beyond two-sample testing there are other hypothesis testing problems that are nowadays tackled via machine learning methods, in particular kernel methods. These are for example, goodness-of-fit tests [14], independence testing [15], and conditional independence testing [16]. We add a novel test to this collection called the *kernel conditional moment test*. This test can check whether a given model violates constraints given as conditional moment restrictions and has applications in econometrics and causality.

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*

[15]: Gretton et al. (2007), *A kernel statistical test of independence*

[16]: Zhang et al. (2011), *Kernel-based conditional independence test and application in causal discovery*

0.2. Outline

This thesis will be split in three parts. The first part covering Chapters 1 to 4 will focus on the two-sample problem and the question 'how can we learn test statistics that lead to powerful tests.' The first part can also be read independently of the other two parts. We give a general introduction to two-sample testing and review related work in Chapter 1. We will particularly focus on related work that considers the maximization of test power and will already give outlooks how our later results fit in. Furthermore a focus will be put on kernel-based hypothesis tests working with the maximum mean discrepancy.

In Chapter 2 we will present an approach that allows us to learn a powerful hypothesis test without splitting the data. The test will be based on linear-time estimates of the MMD. Linear-time MMD tests are very scalable, as the name indicates, their runtime only scales linearly with the sample size. On the downside, the test is statistically not very efficient as it ignores some informative terms. The test considers a finite set of base kernels and optimizes their linear combination by optimizing an asymptotic test power criterion corresponding to a signal-to-noise ratio. What enables this method is that the linear-time MMD estimate is asymptotically normally distributed under the null and alternative hypothesis. Our procedure follows the same idea as in [6], but we overcome the need to split the data. The main theoretical contribution of Chapter 2 is a generalization of the post-selection inference framework [17], shown in Theorem 2.3.2, which is of independent interest. This allows us to derive the distribution of the test statistic under the null hypothesis, conditional on it being selected. Hence we can find reliable test thresholds without data splitting. We provide experiments showing the improvements over the previously existing approach based on data splitting. Interestingly, we managed to obtain improved performance over [6], without increased computational cost.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

The approach of Chapter 2, however, crucially relies on asymptotic normality of the test statistic under the null hypothesis and is thus tied to the linear-time MMD estimates. The downside of the linear-time MMD estimate is that it has a very high variance and thus requires large datasets for powerful tests. At present it is not feasible to extend the post-selection

inference framework also to settings working with the quadratic-time MMD estimate. In Chapter 3 we thus focus on approaches to optimize the kernel for the quadratic-time MMD estimate. Prior work [7, 8] derived a test power criterion based solely on the asymptotic distribution under the alternative hypothesis, which is also normal for the quadratic time estimate. Hence the optimization objective is also a signal-to-noise ratio. We propose that instead of learning a kernel, we should directly learn a one-dimensional witness function. In fact, in the simplest case, we use previously proposed methods to optimize the kernel function but additionally use the training data again to estimate the MMD-witness function. This leads us to the more general definition of witness two-sample tests (WiTS-tests). WiTS tests use the mean discrepancy of a witness function h as a test statistic

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$\tau(P, Q) = \mathbb{E}_{X\sim P}\left[h(X)\right] - \mathbb{E}_{Y\sim Q}\left[h(Y)\right]. \tag{0.1}$$

The witness function itself is optimized on held-out data via an asymptotic test power criterion, which once more correspond to a signal-to-noise ratio, i.e., the empirical estimate of $\tau(P, Q)$ divided by the empirical standard deviation of the estimator. We show how to solve the corresponding optimization problem over functions in a reproducing kernel Hilbert space via kernel Fisher discriminant analysis. We also show how to solve this optimization procedure efficiently using the Nyström approximation and conjugate gradient. This once more allows us to trade some statistical significance for computational efficiency. The WiTS tests are an example where in the first stage we can use any sort of engineering we want without the danger of (accidental) p-value hacking. As a simple example, we use cross-validation to select the kernel and the regularization in the optimization stage. We also obtain insights into the kernel optimization when using MMD. In fact, optimizing the RKHS to maximize test power, corresponds to tweaking the RKHS such that its MMD-witness function has optimal test power when used in a WiTS test. Empirically, the WiTS test, although conceptually simpler, can outperform existing approaches.

Chapter 4 has a very practically oriented agenda. The main shortcoming of Chapter 3 is that the signal-to-noise ratio determining the test power of a witness function is a rather uncommon optimization objective. While we were able to solve it over an RKHS, it is unclear how to optimize this in other machine learning frameworks. As we find, it is unnecessary to develop new methods specifically for optimizing the witness, since we show that in fact optimizing the signal-to-noise ratio is (asymptotically) equivalent to finding the function minimizing a squared loss or a cross-entropy loss. This suddenly enables us to deploy WiTS tests with any existing machine learning framework. In particular, it allows to harvest recent advancements in automated machine learning and building on well-engineered machine learning pipelines. We call the resulting test AutoML two-sample test. Besides the theoretical insights, we discuss the application of two-sample tests to detect distribution shift and run a large distribution shift benchmark [18], where we use the same method for all experiments. This is a great advantage over prior work, which often uses different hyperparameters on different testing problems. To make testing as user friendly as possible, we provide the open-source Python package `autotst`, which wraps existing machine learning frameworks

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

`autotst` is installable via `pip install autotst` and is available on github.com/jmkuebler/auto-tst

and implements the witness two-sample testing pipeline. With its default setting, given two sample arrays `sample_P` and `sample_Q`, computing p-values with `autotst` requires only 3 lines of code:

```
import autotst
tst = autotst.AutoTST(sample_P, sample_Q)
p_value = tst.p_value()
```

Beyond the default settings, `autotst` is easy to customize and to use with other machine learning frameworks. We will not include a detailed discussion of the package in this thesis. Instead we refer to the documentation github.com/jmkuebler/auto-tst.

In Chapters 1 to 4 the computational cost of the various tests plays an important role: In Chapter 2 we use a linear-time test, in Chapter 3 we introduce a test that naively runs in cubic time, but propose an approximate solution, and in Chapter 4 we leverage existing AutoML frameworks that explicitly allow control of the resources, by setting runtime and/or storage limits. In the second part of the thesis (Chapter 5) we briefly explore whether we can use quantum computers to speed up the estimation of the maximum mean discrepancy. To do this we define the quantum mean embedding, which generalizes the kernel mean embedding [19] by mapping a probability distribution onto a pure quantum state. We show under which conditions it is a one-to-one representation of probability distributions. We then discuss the constraints of quantum information that seem to block the road to a quantum speedup. In particular, it is known that there cannot exist a machine which given as input some quantum states, creates the superposition of those states. But this is exactly what is needed to create the quantum mean embedding. So although the initial idea of using a quantum computer to speed up MMD estimation seems promising, there are conceptual problems that hinder this. Although not directly relevant to this thesis, we further explored other aspects of quantum machine learning during my doctoral studies [11–13]. In particular the NeurIPS 2021 paper [12] provided a thorough statistical analysis of learning with quantum kernels and its limitations.

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

[11]: Kübler et al. (2020), *An adaptive optimizer for measurement-frugal variational algorithms*; [12]: Kübler* et al. (2021), *The inductive bias of quantum kernels*; [13]: Jerbi et al. (2023), *Quantum machine learning beyond kernel methods*

The third part of the thesis (Chapter 6) presents a new type of kernel-based hypothesis test, i.e., for a task different than two-sample testing. The kernel conditional moment test represents conditional moment restrictions in the reproducing kernel Hilbert space. This leads to a handy test statistic to assess whether a given model violates a given set of conditional moment restrictions.

The thesis ends with a summarizing conclusion and outlooks for potential future work.

0.3. Underlying manuscripts and contributions

The research covered in this thesis was published in diverse papers with different collaborators. The chapters of this thesis are often built **verbatim** on these publications. The following lists the papers and describes my personal contributions. Papers I contributed to during my PhD studies that are not part of this thesis are also included at the end.

Articles included in this thesis:

Chapter 2:
J. M. Kübler, W. Jitkrittum, B. Schölkopf, and K. Muandet. 'Learning Kernel Tests Without Data Splitting'. In: *NeurIPS*. 2020
Contributions: The original idea to address data splitting in kernel-based two-sample testing came from my co-authors. I generalized the existing theory, proved the main results, implemented and ran the experiments. All authors were involved in discussions and contributed to the writing of the paper.

Chapter 3:
J. M. Kübler, W. Jitkrittum, B. Schölkopf, and K. Muandet. 'A Witness Two-Sample Test'. In: *AISTATS*. 2022
Contributions: I had the original idea, showed the main results, implemented and ran the experiments. My collaborators helped with technical details of some proofs in the appendix. All authors were involved in discussions and contributed to the writing of the paper.

Chapter 4:
J. M. Kübler, V. Stimper, S. Buchholz, K. Muandet, and B. Schölkopf. 'AutoML Two-Sample Test'. In: *NeurIPS*. 2022
Contributions: I had the original idea, showed the main results, implemented the experiments, and wrote major parts of the paper. Simon Buchholz helped in the discussion leading to the main theoretical insights. Ultimately he also wrote down the proof. Vincent Stimper helped running the experiments on the cluster, helped with the preparation of the `autotst` package, and helped with discussion. All authors were involved in discussions and contributed to the writing of the paper.

Chapter 5:
J. M. Kübler, K. Muandet, and B. Schölkopf. 'Quantum mean embedding of probability distributions'. In: *Phys. Rev. Research* 1 (2019)
Contributions: The original idea was jointly developed by all authors. I stated and proved the main results. All authors were involved in the discussions and contributed to the writing of the paper.

Chapter 6:
K. Muandet, W. Jitkrittum, and J. Kübler. 'Kernel Conditional Moment Test via Maximum Moment Restriction'. In: *UAI*. 2020
Contributions: The original idea was developed by Krikamol Muandet. I helped with the discussion, the proofs and the writing of the paper.

Articles not included in this thesis:

1. J. M. Kübler*, S. Buchholz*, and B. Schölkopf. 'The inductive bias of quantum kernels'. In: *NeurIPS*. 2021 (*equal contribution of JMK and SB)
2. L. Gresele*, J. von Kügelgen*, J. M. Kübler*, E. Kirschbaum, B. Schölkopf, and D. Janzing. 'Causal Inference Through the Structural Causal Marginal Problem'. In: *ICML*. 2022 (*LG, JvK, and JMK are co-first authors)

3. J. M. Kübler, A. Arrasmith, L. Cincio, and P. J. Coles. 'An adaptive optimizer for measurement-frugal variational algorithms'. In: *Quantum* 4 (2020)
4. S. Jerbi, L. J. Fiderer, H. Poulsen Nautrup, J. M. Kübler, H. J. Briegel, and V. Dunjko. 'Quantum machine learning beyond kernel methods'. In: *Nature Communications* 14.1 (2023)

Learning powerful test statistics for two-sample testing

1. Introduction to two-sample testing and related work

This chapter introduces the two-sample problem and related literature in depth and prepares for the subsequent three chapters.

1.1. Hypothesis testing

Testing statistical hypotheses is at the core of scientific discovery. Given a hypothesis about phenomenon it prescribes a principled way of using data to find statistical evidence against such a hypothesis [1]. We will test a *null hypothesis* H_0 against a specific *alternative hypothesis* H_1. While the hypotheses state something about a data-generating process, we will use the collected data to compute a real-valued *test statistic*, often denoted as τ. Thus τ will be a random variable and we will translate the null and alternative hypotheses into hypotheses about the distribution of τ. The hypotheses we usually consider is that the mean of τ is zero under the H_0 and positive under H_1. We will thus reject the null hypothesis if the observed value of τ is *significantly* larger than what we would expect under the null hypothesis. Since τ is a random variable, this can lead to two types of errors:

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

Type-I error: We reject H_0 although it was true.
Type-II error: We do not reject H_0 although it was false.

Our tests will be designed to control the Type-I error at a prespecified significance level $\alpha \in (0, 1)$ and we will then aim to minimize the Type-II error. For a given test statistic and significance level α[1], we define the *test threshold* t_α such that

$$\Pr[\tau \geq t_\alpha \mid H_0] \leq \alpha. \tag{1.1}$$

1: We will mostly use $\alpha = 5\%$ in our experiments. In practice, users should always think carefully how to choose the significance level. This has to take into account the severity of Type-I and Type-II errors. Increasing α decreases the Type-II error at an increased Type-I error and vice versa [1, p. 57].

The test then *rejects* the null hypothesis if $\tau \geq t_\alpha$. Note that such a testing procedure is inherently asymmetric in the sense that failing to reject does not provide direct evidence that null hypothesis is true.

Alternatively to computing a threshold, we can also assign a p-value to the observed test statistic, which is defined as the "smallest significance level [...] at which the hypothesis could be rejected for the given observation" [1, Chapter 3.3].

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

Choice of test statistic. A central question in the practical application of hypothesis tests is how to best choose the test statistic. As mentioned earlier our goal will be to maximize the test power, i.e., minimize the rate of Type-II errors, while controlling the Type-I error at level α. In this part of the thesis, we will do this for the two-sample problem, which we introduce in more detail next. We will explicitly learn powerful tests by using the observed data. Note that for some people 'choosing' a test statistic might be slightly disconcerting. However, using a prefixed test statistic only superficially is different. After all some scientist had to choose this test statistic. Usually this process is neither questioned nor justified. Since this choice happens (hopefully) independently of the data it is not a conceptual problem. But this illustrates that a data-driven approach of choosing a test statistic is conceptually nothing different, it is just more explicit than the standard approach.

1.2. The two-sample problem

We now introduce the two-sample problem. Let X and Y be random variables with (unknown) probability distributions P and Q over $\mathcal{X} \subseteq \mathbb{R}^d$ for some $d \in \mathbb{N}$.[2] Given samples $\mathbb{X} = \{x_1, \dots, x_n\}$ drawn i.i.d. from P and $\mathbb{Y} = \{y_1, \dots, y_m\}$ drawn i.i.d. from Q, with sample sizes $n, m \in \mathbb{N}$, our goal is to test the null hypothesis

$$H_0 : P = Q$$

against the alternative hypothesis

$$H_1 : P \neq Q.$$

We will usually not (explicitly) assume much more about the distributions that we consider. Generally, and particularly without strong parametric assumptions it will not be possible to define a *uniformly most powerful test* (UMP) test, i.e., a test that has highest power for all possible distributions that fall under the alternative hypothesis [1, Chapter 3]. An example, with strong assumptions is the following:

2: For conciseness, whenever we use X or Y in expressions without explicit specification, we intend that they are random variables distributed according to P and Q, respectively, e.g., $\mathbb{E}[f(X)] = \mathbb{E}_{X\sim P}[f(X)]$.

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

Example 1.2.1 Let $P = \mathcal{N}(\xi, 1)$ and $Q = \mathcal{N}(\eta, 1)$. Consider $H_0 : \xi = \eta$ and the **one-sided** alternative $H_1 : \xi > \eta$. Then using the mean discrepancy as test statistic

$$\tau(\mathbb{X}, \mathbb{Y}) = \frac{1}{n}\sum_{i=1}^{n} x_i - \frac{1}{m}\sum_{j=1}^{m} y_j$$

and rejecting the null hypothesis when it is too large, leads to a uniformly most powerful test against all possible alternative hypotheses [1, Problem 3.61]. To derive the test threshold t_α, we use that under the null hypothesis $\tau(\mathbb{X}, \mathbb{Y}) \sim \mathcal{N}(0, \frac{1}{n} + \frac{1}{m})$ and thus set $t_\alpha = \sqrt{\frac{1}{n} + \frac{1}{m}}\Phi^{-1}(1-\alpha)$, where Φ denotes the cumulative distribution function (CDF) of the standard normal, and Φ^{-1} its inverse. We would then reject the test whenever $\tau(\mathbb{X}, \mathbb{Y}) \geq t_\alpha$ and thus by definition control the Type-I error at α. We can even directly compute the test power because under a fixed alternative $\xi > \eta$ we have $\tau(\mathbb{X}, \mathbb{Y}) \sim \mathcal{N}(\xi - \eta, \frac{1}{n} + \frac{1}{m})$ and thus

$$\begin{aligned}
&\Pr\left[\tau(\mathbb{X}, \mathbb{Y}) \geq t_\alpha \mid \xi > \nu)\right] \\
&= \Pr\left[\frac{\tau(\mathbb{X}, \mathbb{Y})}{\sqrt{\frac{1}{n} + \frac{1}{m}}} \geq \frac{t_\alpha}{\sqrt{\frac{1}{n} + \frac{1}{m}}} \mid \xi > \nu)\right] \\
&= \Pr\left[Z \geq -\left(\frac{\xi - \eta}{\sqrt{\frac{1}{n} + \frac{1}{m}}} - \Phi^{-1}(1-\alpha)\right) \mid Z \sim \mathcal{N}(0,1)\right] \\
&= \Phi\left(\frac{\xi - \eta}{\sqrt{\frac{1}{n} + \frac{1}{m}}} - \Phi^{-1}(1-\alpha)\right).
\end{aligned}$$

We illustrate this graphically in Figure 1.1. As might intuitively be expected, the larger the difference $\xi - \eta$ the higher the test power.

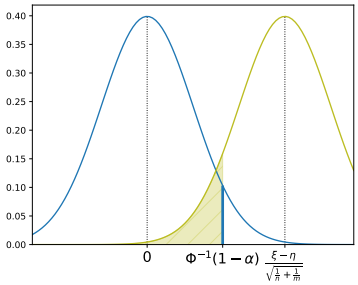


Figure 1.1.: Test power for Example 1.2.1. The green area corresponds to the rate of Type-II errors. The larger the difference of means $\xi - \eta$ and the larger the samplesize, the larger the test power is.

We emphasize that the existence of a UMP test for the problem in Example 1.2.1 is only possible through strong assumptions. If for example, we had $H_1 : \eta \neq \xi$, i.e., a two-sided alternative there does not exists a UMP test anymore. A reasonable test would then be to use $\left(\frac{1}{n}\sum_{i=1}^{n} x_i - \frac{1}{m}\sum_{j=1}^{m} y_j\right)^2$ as test statistic and reject for too large values. This test is reasonable in the sense that it is *consistent* against any fixed alternative, i.e., its power approaches 1 as the sample sizes go to infinity. However, this *two-sided* test would have lower power than the test in Example 1.2.1 for any alternative with $\xi > \eta$, thus it is not UMP. Choosing a good test statistic is thus a delicate task: We can choose a test that is consistent against all alternative hypotheses, or choose a test that is inconsistent against some alternatives but provides larger test power on others. This is reminiscent of the *bias-variance tradeoff* in supervised machine learning [25, Chapter 3.2].

[25]: Bishop et al. (2006), *Pattern recognition and machine learning*

Given this insight and that we will be interested in testing on complex datasets, the goal of this work cannot be to find *the best* two-sample test. Rather, we will provide procedures that find good test statistics in a principled way and allow to incorporate prior knowledge, for example by choosing a particular model class or machine learning framework.

p-values based on permutations. For many test statistics we will encounter in this thesis it is infeasible to directly derive the finite sample distribution under the null hypothesis. In two-sample testing we can alternatively obtain p-values by permuting the samples. We will now illustrate this in detail and use it many times throughout this thesis. Let τ be an arbitrary test statistic. Let $\mathbb{Z} = \{x_1, \dots, x_n, y_1, \dots, y_m\}$ denote the pooled sample and let $\tau(\mathbb{Z}) = \tau(\mathbb{X}, \mathbb{Y})$ be a test statistic that is computed by taking the first n elements of $\mathbb{Z}$ as $\mathbb{X}$ and the last m elements as $\mathbb{Y}$. Let Π be the uniform distribution over all permutations $\pi : \{1, \dots, n+m\} \to \{1, \dots, n+m\}$ and denote by $\mathbb{Z}^\pi$ the corresponding permutation of $\mathbb{Z}$. Assume that the null hypothesis holds and thus $\mathbb{Z} \sim P^{(n+m)}$, then for $\alpha \in (0, 1)$

$$\Pr_{\mathbb{Z} \sim P^{(n+m)}} \left[\Pr_{\pi \sim \Pi} \left[\tau(\mathbb{Z}) \leq \tau(\mathbb{Z}^\pi)\right] \leq \alpha\right] \leq \alpha. \tag{1.2}$$

The statement follows since $P^{(n+m)}$ is invariant under permutations (see proof of Lemma 1.2.1 below). Equation 1.2 implies that for a given realization $\mathbb{Z}$ we can define a p-value as

$$p\left(\tau(\mathbb{Z})\right) = \Pr_{\pi \sim \Pi}\left[\tau(\mathbb{Z}) \leq \tau(\mathbb{Z}^\pi)\right].$$

Since computing all possible permutations quickly gets infeasible (there are $(n+m)!$ different permutations of $n+m$ elements), we will approximate p-values through $B \in \mathbb{N}$ random permutations $\pi_1, \dots, \pi_B$ and use a biased[3] estimator

$$\hat{p}\left(\tau(\mathbb{Z})\right) = \frac{1}{B+1}\left(1 + \sum_{i=1}^{B} \mathbf{I}\left[\tau(\mathbb{Z}) \leq \tau(\mathbb{Z}^{\pi_i})\right]\right), \tag{1.3}$$

where $\mathbf{I}$ is the indicator function. As we state in the next Lemma, rejecting the null hypothesis whenever $\hat{p} \leq \alpha$ correctly controls the Type-I error for any sample size $n, m \in \mathbb{N}$ and number of permutations $B \in \mathbb{N}$.

3: Removing the bias term, leads to uncontrollable Type-I errors [26]. For illustration consider $B = 1$. The probability that $\tau(\mathbb{Z}) \leq \tau(\mathbb{Z}^{\pi_1})$ can be arbitrary close to $1/2$. We would then have $\hat{p} = \frac{1}{2}$ with probability close to $1/2$ and we can thus not control Type-I errors for $\alpha < 1/2$. Forgetting the bias term is a common issue and also happened to us in the original publications of [21, 22]. It also happens in popular packages like `SciPy`, which we fixed with pull request #16469.

Lemma 1.2.1 *Let $n, m, B \in \mathbb{N}$ and P be an arbitrary distribution over $\mathcal{X}$. Then for $\hat{p}(\tau(\mathbb{Z}))$ defined as in Equation 1.3*

$$Pr_{\mathbb{Z}\sim P^{(n+m)}}\left[\hat{p}(\tau(\mathbb{Z})) \leq \alpha\right] \leq \alpha.$$

Since this is quite fundamental, and of general relevance, we include the proof here in the main part.

Proof. If we have $B+1$ real random variables $A_1, \ldots, A_B$ and A^* that are independently and identically distributed according to an arbitrary distribution D. Then we have for $K \in \{0, \ldots, B\}$

$$\Pr_{\substack{A^*\sim D\\ A_i\sim D}}\left[\sum_{i=1}^{B}\mathbf{I}\left[A^* \leq A_i\right] \leq K\right] \leq \frac{K+1}{B+1}. \tag{1.4}$$

Next, observe that $P^{(n+m)}$ is invariant under permutations, meaning that if $\mathbb{Z} \sim P^{(n+m)}$ and $\pi^* \sim \Pi$, then $\mathbb{Z}^{\pi^*} \sim P^{(n+m)}$ as well. Thus we have that $\hat{p}(\tau(\mathbb{Z}))$ follows the same distribution as $\hat{p}(\tau(\mathbb{Z}^{\pi^*}))$ and we can write

$$\begin{aligned}
&\Pr_{\mathbb{Z}\sim P^{(n+m)}}\left[\hat{p}(\tau(\mathbb{Z})) \leq \alpha\right]\\
&= \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\pi^*\sim\Pi}\left[\hat{p}(\tau(\mathbb{Z}^{\pi^*})) \leq \alpha\right]\\
&= \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\substack{\pi^*\sim\Pi\\ \pi_i\sim\Pi}}\left[\frac{1}{B+1}\left(1+\sum_{i=1}^{B}\mathbf{I}\left[\tau(\mathbb{Z}^{\pi^*}) \leq \tau(\mathbb{Z}^{\pi_i\pi^*})\right]\right) \leq \alpha\right]\\
&\overset{(a)}{=} \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\substack{\pi^*\sim\Pi\\ \pi_i\sim\Pi}}\left[\frac{1}{B+1}\left(1+\sum_{i=1}^{B}\mathbf{I}\left[\tau(\mathbb{Z}^{\pi^*}) \leq \tau(\mathbb{Z}^{\pi_i})\right]\right) \leq \alpha\right]\\
&\overset{(b)}{=} \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\substack{A^*\sim D(\mathbb{Z})\\ A_i\sim D(\mathbb{Z})}}\left[\frac{1}{B+1}\left(1+\sum_{i=1}^{B}\mathbf{I}\left[A^* \leq A_i\right]\right) \leq \alpha\right]\\
&= \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\substack{A^*\sim D(\mathbb{Z})\\ A_i\sim D(\mathbb{Z})}}\left[\sum_{i=1}^{B}\mathbf{I}\left[A^* \leq A_i\right] \leq \alpha(B+1)-1\right]\\
&= \Pr_{\mathbb{Z}\sim P^{(n+m)}}\Pr_{\substack{A^*\sim D(\mathbb{Z})\\ A_i\sim D(\mathbb{Z})}}\left[\sum_{i=1}^{B}\mathbf{I}\left[A^* \leq A_i\right] \leq \lfloor\alpha(B+1)-1\rfloor\right]\\
&\overset{(1.4)}{\leq} \frac{\lfloor\alpha(B+1)-1\rfloor+1}{B+1} = \frac{\lfloor\alpha(B+1)\rfloor}{B+1}\\
&\leq \alpha.
\end{aligned}$$

In (a) we used that the distribution of a permutation is invariant under concatenation with another permutation and in (b) we defined the random variable $A^* = \tau(\mathbb{Z}^{\pi^*})$ for a given $\mathbb{Z}$ and denoted its distribution with $D(\mathbb{Z})$. Analogously , we define A_i for $i = 1\ldots, B$. Where we used (1.4), we also used that this holds for all $\mathbb{Z}$. □

Remark 1.2.1 Three points should be considered when choosing the number of permutations B:

1. To make the last inequality in the proof tight (and thus make the rate of rejections highest) one should choose B such that $\lfloor\alpha(B+1)\rfloor$ is as close as possible to $\alpha(B+1)$, ideally equal.

2. $\hat{p} \geq \frac{1}{B+1}$ by Equation 1.3. Hence, to even have the possibility to reject the null we should have that $\frac{1}{B+1} \leq \alpha$ which is the case whenever $B \geq \frac{1}{\alpha} - 1$. Thus we should choose B at least of this size.
3. $\hat{p}$ is random given $\mathbb{Z}$. It is desirable to minimize this additional randomness and choose B as large as possible. Of course this should be traded-off with the computational cost of using many permutations.

1.3. Maximum mean discrepancy

As we have seen in the previous section using the mean discrepancy in the one-dimensional case can be UMP under strong assumptions (Example 1.2.1). However, such a test is unable to reliably detect differences of distributions when their means are the same. A classic test that consistently detects arbitrary (fixed) differences in univariate distributions is the *Kolmogorov-Smirnov* test, using as a test statistic the maximal difference of the empirical cumulative distribution function of $\mathbb{X}$ and $\mathbb{Y}$. In this work we focus on test built with machine learning models. We will now introduce kernel methods [4] to then later define flexible tests based on the maximum mean discrepancy.

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

Definition 1.3.1 (Positive Definite Kernel, see Defintion 2.5 of [4]) *Let $\mathcal{X}$ be a nonempty set. A symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that for all $n \in \mathbb{N}$ and $x_1, \ldots, x_n \in \mathcal{X}$, and all $c_1, \ldots, c_n \in \mathbb{R}$*

$$\sum_{i,j} c_i c_j k(x_i, x_j) \geq 0$$

is called positive definite kernel.[4]

4: We will usually simply say 'kernel'. Furthermore, we limit ourselves here to real kernels, although the definition generalizes to complex kernels.

Associated with a kernel is a unique *reproducing kernel Hilbert space* (RKHS) $\mathcal{H}$ of functions $f : \mathcal{X} \to \mathbb{R}$ for which the reproducing property

$$\langle f, k(\cdot, x) \rangle = f(x)$$

holds for all $f \in \mathcal{H}$ and $x \in \mathcal{X}$ [4, Chapter 2.2.3].

Instead of embedding a single point in the RKHS via $x \mapsto k(\cdot, x)$ we can also embed a whole probability distribution in the RKHS, via the kernel mean embedding [19, Chapter 3]

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

$$\mu_P = \mathbb{E}_{X \sim P}[k(\cdot, X)]. \tag{1.5}$$

We have that $\mu_P \in \mathcal{H}$ if $\mathbb{E}_{X \sim P}[k(X, X)]$ is finite [19, Lemma 3.1]. In this case the expectation of a function $f \in \mathcal{H}$ is given by $\mathbb{E}_{X \sim P}[f(X)] = \langle f, \mu_P \rangle$. We will use $\mu_{\mathbb{X}}$ to denote the mean embedding of the empirical distribution of a sample $\mathbb{X}$.

We will now use kernel methods to define a test statistic that can be consistent against arbitrary fixed alternatives. The underlying idea is that if two distributions differ, then there exists a function such that the mean under P is different than the mean under Q. The idea of the *maximum*

mean discrepancy (MMD) is to find the function that maximizes the mean discrepancy over a unit ball in an RKHS [5].[5]

[5]: Gretton et al. (2012), *A kernel two-sample test*

5: The MMD was introduced in a series of works [27–29] which was then summarized and extended in [5].

Definition 1.3.2 (Maximum mean discrepancy, Definition 2 of [5]) *Let $\mathcal{F}$ be a class of functions $f : \mathcal{X} \to \mathbb{R}$ and let P and Q be two distributions over $\mathcal{X}$. We define the MMD as*

$$MMD(P, Q \mid \mathcal{F}) := \sup_{f \in \mathcal{F}} \mathbb{E}_{X \sim P}[f(X)] - \mathbb{E}_{Y \sim Q}[f(Y)]. \tag{1.6}$$

While the definition holds for arbitrary function spaces $\mathcal{F}$,[6] it becomes particularly useful when using a unit ball of an RKHS $\mathcal{H}$ for which μ_P and μ_Q exist. In this case we have $\mathbb{E}[f(X)] - \mathbb{E}[f(Y)] = \langle \mu_P - \mu_Q, f \rangle$ for all $f \in \mathcal{H}$. The function that *witnesses* the maximum mean discrepancy is then aligned with the difference in mean embeddings and we will refer to it as MMD witness. The squared MMD then simply is [5, Lemma 4]

6: See Chapter 7.1 of [5]. We also note that the MMD is an integral probability metric [30].

$$\mathrm{MMD}^2(P, Q \mid \mathcal{H}) := \left[\sup_{\substack{f \in \mathcal{H} \\ \|f\| \leq 1}} \mathbb{E}[f(X)] - \mathbb{E}[f(Y)] \right]^2 = \|\mu_P - \mu_Q\|^2_{\mathcal{H}}. \tag{1.7}$$

Whenever, we write $\mathrm{MMD}(\cdot, \cdot \mid \mathcal{H})$ for an RKHS $\mathcal{H}$ we leave implicit that we constrain the supremum to the unit ball. We will also occasionally write $\mathrm{MMD}(\cdot, \cdot \mid k)$, for a kernel k instead of using its associated RKHS or simply $\mathrm{MMD}(P, Q)$ and leave the dependence implicit.

From Equation 1.7 it follows that the squared MMD is a metric on probability distributions if the mean embedding is injective. This has been formalized through the notion of *characteristic* kernels [31]. In particular we can then translate our abstract null and alternative hypothesis into concise mathematical theses, i.e., $H_0 : \mathrm{MMD}^2(P, Q \mid \mathcal{H}) = 0$ versus $H_1 : \mathrm{MMD}^2(P, Q \mid \mathcal{H}) > 0$. Then we can use an empirical estimate of the squared MMD to perform a two-sample test and consistently detect any (fixed) different distributions. Of course, this does not imply that at a fixed sample size the test will actually have high power against arbitrary alternatives [5, Chapter 3.2], otherwise this thesis would not be necessary.

[31]: Fukumizu et al. (2008), *Kernel Measures of Conditional Dependence*

Empirical estimates. To define a hypothesis test, we will use an empirical estimate of the MMD as test statistic.[7] Here, we solely introduce common test statistics. Approaches to derive test thresholds will be introduced later when relevant. For an overview we refer to [5]. Using the reproducing property and denoting with $X' \sim P, Y' \sim P$ independent copies of X, Y, we can rewrite Equation 1.7 as

7: We will now simply say MMD, even though we will always work with the squared MMD

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\mathrm{MMD}^2(P, Q) = \mathbb{E}[k(X, X') - k(X, Y') - k(X', Y) + k(Y, Y')]. \tag{1.8}$$

A straight-forward (biased) estimator of the MMD is given by replacing the expectation with the empirical expectation leading to

$$\begin{aligned}\widehat{\mathrm{MMD}}_b^2(\mathbb{X},\mathbb{Y}) = &\frac{1}{n^2}\sum_{i,j=1}^{n} k(x_i,x_j)\\ &-\frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} k(x_i,y_j)\\ &+\frac{1}{m^2}\sum_{i,j=1}^{m} k(y_i,y_j).\end{aligned} \tag{1.9}$$

We can think of the biased estimator as using the available data first to 'learn' the witness function and then use the same data to estimating the mean discrepancy. A minimum-variance unbiased estimator is given by leaving out the terms $k(x_i,x_i)$ and $k(y_i,y_i)$ [5, Lemma 6]

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\begin{aligned}\widehat{\mathrm{MMD}}_u^2(\mathbb{X},\mathbb{Y}) = &\frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{j\neq i}^{n} k(x_i,x_j)\\ &-\frac{2}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} k(x_i,y_j)\\ &+\frac{1}{m(m-1)}\sum_{i=1}^{m}\sum_{j\neq i}^{m} k(y_i,y_j).\end{aligned} \tag{1.10}$$

One can then use these quantities as test statistic and there exist different approaches to estimate test thresholds [5]. A popular approach is to permute the data and reestimate the quantities as described in Lemma 1.2.1.

Both of the above estimators have a cost scaling quadratically in the sample size. If we have equal sample size $n = m$ we can also define an estimator whose cost only scales linearly with the sample size. Let us define $Z = (X, X', Y, Y') \sim P \otimes P \otimes Q \otimes Q$ and $h(Z) = k(X,X') - k(X,Y') - k(X',Y) + k(Y,Y')$. Then we can rewrite Equation 1.8 as

$$\mathrm{MMD}^2(P,Q) = \mathbb{E}\left[h(Z)\right]. \tag{1.11}$$

Assuming for simplicity that n is even, we can split the samples and define $z_i = (x_i, x_{n/2+i}, y_i, y_{n/2+i})$ for $i = 1,\dots,n/2$. The linear time estimate is then simply [5, Lemma 14]

$$\widehat{\mathrm{MMD}}_{\mathrm{lin}}^2(\mathbb{X},\mathbb{Y}) = \frac{1}{n/2}\sum_{i=1}^{n/2} h(z_i). \tag{1.12}$$

Since this estimator is simply an empirical mean its asymptotic distribution is characterized by the Central Limit Theorem and is asymptotically normal [32] [5, Corollary 16]. This allows to define asymptotic thresholds in closed-form, a property we will exploit in Chapter 2.

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

[5]: Gretton et al. (2012), *A kernel two-sample test*

1.4. Classifier two-sample tests

Another machine-learning based approach to two-sample testing is built on classification accuracy. Intuitively, if there exists a classifier that

achieves accuracy significantly above chance level when classifying data from P and Q we can conclude that $P \neq Q$ [9, 33]. For simplicity we will assume equal sample size $m = n$. The idea is to split the available data $\mathbb{X}$, $\mathbb{Y}$ into disjoint training and test sets $\mathbb{X}_{\text{tr}}, \mathbb{X}_{\text{te}}$, and $\mathbb{Y}_{\text{tr}}, \mathbb{Y}_{\text{te}}$. One labels data from $\mathbb{X}$ with '1' and data from $\mathbb{Y}$ with '0' and trains a classifier $f : \mathcal{X} \to \{0, 1\}$ on the training samples. The classification accuracy on the test set then serves as test statistic:

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*

$$\hat{\tau}(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}} \mid f) = \frac{1}{2}\left(\frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}} f(x_i) + \frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}}(1 - f(y_i))\right)$$

p-values can then be estimated either by permuting the test data or directly by using the distribution of $\hat{\tau}(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}} \mid f)$ under the null hypothesis, where the accuracy equals $1/2$ [9, Section 3.1]. Since p-values are only estimated on the test data, one can use cross-validation or any other technique in the training phase [9, Section 3.2].

1.5. Related work

The two prior subsections introduced the MMD and classifiers as popular examples of two-sample tests. We shall now introduce a few more works that cover details about such tests and which are of particular relevance to this thesis. We will start by introducing methods that optimize the kernel for MMD-based two sample tests.

Initially people used the median heuristic or simply maximized the MMD [34] when choosing a good kernel function for two-sample testing. [6] was the first work that introduced a principled optimization of the test power when choosing the kernel function. They worked with the linear-time MMD estimate (1.12). Since this statistic is asymptotically normally distributed, deriving an asymptotic test power criterion is quite simple. Essentially it is the same idea presented in Example 1.2.1 and Figure 1.1. To minimize the rate of Type-II errors (green area in Figure 1.1) one maximizes the signal-to-noise ratio of the distribution under the alternative hypothesis. Visually this means that one maximizes the distance of means of the two distributions, normalized by their width – hence the signal-to-noise ratio (SNR). When using $d \in \mathbb{N}$ different base kernels, their corresponding linear-time MMD estimates will asymptotically jointly follow a multivariate normal distribution. Using that the sum of kernel functions is again a kernel function, [6] propose to learn the (positive) linear combination of base kernels that minimizes the SNR. They used data splitting when learning the kernel combination and observe that their principled approach indeed leads to improved performance. We will show how to overcome data splitting in this approach in Chapter 2. While the linear-time estimates are fast to compute even on large data and have appealing asymptotic distributions, they use the information inefficiently.

[34]: Sriperumbudur et al. (2009), *Kernel Choice and Classifiability for RKHS Embeddings of Probability Distributions*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[7] generalized the approach of optimizing the asymptotic SNR to the quadratic-time MMD estimate (1.10). We here use a slightly modified

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*

version given as

$$\widehat{\mathrm{MMD}}_u^2 = \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}, \tag{1.13}$$

with $H_{ij} = \langle k(x_i, \cdot) - k(y_i, \cdot), k(x_j, \cdot) - k(y_j, \cdot) \rangle$ and again assuming $n = m$. This is a U-statistic. Under the null hypothesis, the asymptotic distribution of this test statistic is a (potentially infinite) sum of weighted independent chi-square variables [8, Proposition 2][5, Theorem 12]

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$n\widehat{\mathrm{MMD}}_u^2 \xrightarrow{d} \sum_l \sigma_l (\chi_l^2 - 2), \tag{1.14}$$

where $\chi_l^2 = Z_l^2$ with $Z_l \sim \mathcal{N}(0, 1)$, and σ_l depend on the kernel and on P. It is usually thus infeasible to directly compute quantiles of this distribution. Under the alternative hypothesis and assuming for the chosen kernel $\mu_P \neq \mu_Q$ the asymptotic distribution is asymptotically normal [32, Section 5.5.1], [8, Proposition 2]

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

$$\sqrt{n}\left(\widehat{\mathrm{MMD}}_u^2 - \mathrm{MMD}^2\right) \xrightarrow{d} \mathcal{N}(0, \sigma_{H_1}^2), \tag{1.15}$$

with $\sigma_{H_1}^2 = 4(\mathbb{E}\left[H_{12}H_{13}\right] - \mathbb{E}\left[H_{12}\right]^2)$. Assuming that t_α was the $1-\alpha$ quantile[8] of the asymptotic null distribution (1.14) used as threshold and that the alternative hypothesis holds, by a reasoning similar to Figure 1.1 one can compute the asymptotic test power [8]

8: In practice this is usually estimated via permutations, see Section 1.2.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$\Pr\left[n\widehat{\mathrm{MMD}}_u^2 > r\right] \to \Phi\left(\frac{\sqrt{n}\mathrm{MMD}^2}{\sigma_{H_1}} - \frac{t_\alpha}{\sqrt{n}\sigma_{H_1}}\right). \tag{1.16}$$

Since the second term asymptotically goes to zero, it is the first term that dominates the asymptotic test power. Therefore [7, 8] use an empirical estimate of

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$J = \frac{\mathrm{MMD}^2}{\sigma_{H_1}} \tag{1.17}$$

to optimize the kernel function. They also rely on data splitting. [8] use a gradient-based optimization to continuously optimize a deep kernel. These works conclude that it is beneficial to learn a kernel function and to use Equation 1.13 as a test statistic. In Chapters 3 and 4 we will challenge this idea, by showing that only learning a one-dimensional witness function and using a WiTS test can lead to more powerful tests that are furthermore easier to implement and use.

Learning kernel tests without data splitting

2.

Modern large-scale kernel-based tests such as maximum mean discrepancy (MMD) and kernelized Stein discrepancy (KSD) optimize kernel hyperparameters on a held-out sample via data splitting to obtain the most powerful test statistics. While data splitting results in a tractable null distribution, it suffers from a reduction in test power due to smaller test sample size. Inspired by the selective inference framework, we propose an approach that enables learning the hyperparameters and testing on the full sample without data splitting. Our approach can correctly calibrate the test in the presence of such dependency, and yield a test threshold in closed form. At the same significance level, our approach's test power is empirically larger than that of the data-splitting approach, regardless of its split proportion.

2.1. Introduction

Traditionally, test statistic for a hypothesis test are usually fixed prior to the testing phase. In modern-day hypothesis testing, however, practitioners often face a large family of test statistics from which the best one must be selected before performing the test. For instance, the popular kernel-based two-sample tests [5, 6] (Section 1.3) and goodness-of-fit tests [14, 35] require the specification of a kernel function and its parameter values. Abundant evidence suggests that finding good parameter values for these tests improves their performance in the testing phase [6, 7, 36, 37]. As a result, several approaches have recently been proposed to learn optimal tests directly from data using different techniques such as optimized kernels [6, 8, 37–40], classifier two-sample tests [9, 33], and deep neural networks [41, 42], to name a few. In other words, the modern-day hypothesis testing has become a two-stage "learn-then-test" problem.

Special care must be taken in the subsequent testing when optimal tests are learned from data. If the same data is used for both learning and testing, it becomes harder to derive the asymptotic null distribution because the selected test and the data are now dependent. In this case, conducting the tests as if the test statistics are independent from the data leads to an uncontrollable false positive rate, see, e.g., our experimental results. While permutation testing (Section 1.2) can be applied [43], it is too computationally prohibitive for real-world applications. Up to now, the most prevalent solution is *data splitting*: the data is randomly split into two parts, of which the former is used for learning the test while the latter is used for testing. Although data splitting is simple and in principle leads to the correct false positive rate, its downside is a potential loss of power.

In this chapter, we investigate the two-stage "learn-then-test" problem in the context of modern kernel-based tests [5, 6, 14, 35] where the choice of kernel function and its parameters play an important role.

[5]: Gretton et al. (2012), *A kernel two-sample test*; [6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [36]: Scetbon et al. (2019), *Comparing distributions: L1 geometry improves kernel two-sample testing*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*; [38]: Jitkrittum et al. (2018), *Informative Features for Model Comparison*; [39]: Jitkrittum et al. (2017), *A Linear-Time Kernel Goodness-of-Fit Test*; [40]: Jitkrittum et al. (2017), *An Adaptive Test of Independence with Analytic Kernel Embeddings*

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*

[41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*; [42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*

[43]: Fisher (1935), *The design of experiments*

[5]: Gretton et al. (2012), *A kernel two-sample test*; [6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

The key question is *whether it is possible to employ the full sample for both learning and testing phase without data splitting, while correctly calibrating the test in the presence of such dependency*. We provide an affirmative answer if we learn the test from a vector of jointly normal base test statistics, e.g., the linear-time MMD estimates (Equation 1.12) of multiple kernels. The empirical results suggest that, at the same significance level, the test power of our approach is larger than that of the data-splitting approach, regardless of the split proportion (cf. Section 2.5). The code for the experiments of this chapter is available at `https://github.com/MPI-IS/tests-wo-splitting`.

2.2. Preliminaries

We start with some background material on conventional hypothesis testing and review linear-time kernel two-sample tests. In what follows, we will use $[d] := \{1, \dots, d\}$ to denote the set of natural numbers up to $d \in \mathbb{N}$, $\boldsymbol{\mu} \geq \mathbf{0}$ to denote that all entries of $\boldsymbol{\mu} \in \mathbb{R}^d$ are non-negative, e_i to denote the i-th Cartesian unit vector, and $\|\cdot\| := \|\cdot\|_2$.

Statistical hypothesis testing. Let Z be a random variable taking values in $\mathcal{Z} \subseteq \mathbb{R}^p$ distributed according to a distribution P. The goal of statistical hypothesis testing is to decide whether some *null hypothesis* H_0 about P can be rejected in favor of an *alternative hypothesis* H_A based on empirical data [1, 44]. Let h be a real-valued function such that $0 < \mathbb{E}\left[h^2(Z)\right] < \infty$. In this chapter, we consider testing the null hypothesis $H_0 : \mathbb{E}\left[h(Z)\right] = 0$ against the one-sided alternative hypothesis $H_1 : \mathbb{E}\left[h(Z)\right] > 0$ for reasons which will become clear later. To do so, we define the *test statistic* $\tau(\mathbb{Z}_n) = \frac{1}{n}\sum_{i=1}^n h(z_i)$ as the empirical mean of h based on a sample $\mathbb{Z}_n := \{z_1, \dots, z_n\}$ drawn i.i.d. from P^n. We reject H_0 if the observed test statistic $\hat{\tau}(\mathbb{Z}_n)$ is *significantly* larger than what we would expect if H_0 was true, i.e., if $P(\tau(\mathbb{Z}_n) < \hat{\tau}(\mathbb{Z}_n) \mid H_0) > 1 - \alpha$. Here α is a *significance level* and controls the probability of incorrectly rejecting H_0 (Type-I error). For sufficiently large n we can work with the asymptotic distribution of $\tau(\mathbb{Z}_n)$, which is characterized by the Central Limit Theorem [32].

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*; [44]: Anderson (2003), *An Introduction to Multivariate Statistical Analysis*

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

Lemma 2.2.1 *Let $\mu := \mathbb{E}\left[h(Z)\right]$ and $\sigma^2 := Var\left[h(Z)\right]$. Then, the test statistic converges in distribution to a Gaussian distribution, i.e., $\sqrt{n}\left(\tau(\mathbb{Z}_n) - \mu\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2)$.*

Let Φ be the CDF of the standard normal and Φ^{-1} its inverse. We define the test threshold $t_\alpha = \sqrt{n}\sigma\Phi^{-1}(1-\alpha)$ as the $(1-\alpha)$-quantile of the null distribution so that $P\left(\tau(\mathbb{Z}_n) < t_\alpha \mid H_0\right) = 1 - \alpha$ and we reject H_0 simply if $\hat{\tau}(\mathbb{Z}_n) > t_\alpha$. Besides correctly controlling the Type-I error, the test should also reject H_0 as often as possible when P actually satisfies the alternative H_1. The probability of making a Type-II error is defined as $P\left(\tau(\mathbb{Z}_n) < t_\alpha \mid H_1\right)$, i.e., the probability of failing to reject H_0 when it is false. A powerful test has a small Type-II error while keeping the Type-I error at α. Since Lemma 2.2.1 holds for any μ, and thus both under null and alternative hypotheses, the asymptotic probability of a Type-II error is [6] (see also Example 1.2.1 and Figure 1.1)

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

$$P(\tau(\mathbb{Z}_n) < t_\alpha \mid H_1) \approx \Phi\left(\Phi^{-1}(1-\alpha) - \frac{\mu\sqrt{n}}{\sigma}\right). \tag{2.1}$$

Since Φ is monotonic, this probability decreases with μ/σ, which we interpret as a signal-to-noise ratio (SNR). It is therefore desirable to find test statistics with high SNR.

Kernel two-sample testing. As an example that can be expressed in the above form we present kernel two-sample tests, which we already introduced in Chapter 1 and shortly review here. Given two samples $\mathbb{X}_n$ and $\mathbb{Y}_n$ drawn from distributions P and Q, the two-sample test aims to decide whether P and Q are different, i.e., $H_0 : P = Q$ and $H_1 : P \neq Q$. A popular test statistic for this problem is the maximum mean discrepancy (MMD) of [5]. We use the form of Equation 1.8

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\begin{aligned}\mathrm{MMD}^2(P,Q) &= \mathbb{E}\left[k(X,X') - k(X,Y') - k(X',Y) + k(Y,Y')\right] \\ &= \mathbb{E}\left[h(Z)\right],\end{aligned}$$

where X, X' are independent draws from P, Y, Y' are independent draws from Q, and $h(X,X',Y,Y') := k(X,X') + k(Y,Y') - k(X,Y') - k(Y,X') = h(Z)$. A minimum-variance unbiased estimator of MMD^2 is given by a second-order U-statistic (Equation 1.10). However, this estimator scales quadratically with the sample size, and the distribution under H_0 (Equation 1.14) is not available in closed form. Thus it has to be simulated either via a bootstrapping approach or via a permutation of the samples. For large sample size, the computational requirements become prohibitive [5]. In this chapter, we assume we are in this regime. To circumvent these computational burdens, [5] suggest a 'linear-time' MMD estimate that scales linearly with sample size and is asymptotically normally distributed under both null and alternative hypotheses (Section 1.3). Specifically, let $\mathbb{X}_{2n} = \{x_1, \ldots, x_{2n}\}$ and $\mathbb{Y}_{2n} = \{y_1, \ldots, y_{2n}\}$, i.e., the samples are of the same (even) size. We can define $z_i := (x_i, x_{n+i}, y_i, y_{n+i})$ and $\tau(\mathbb{Z}_n) := \frac{1}{n}\sum_{i=1}^{n} h(z_i)$ as the test statistic, which by Lemma 2.2.1 is asymptotically normally distributed. Furthermore, if the kernel k is characteristic [45], it is guaranteed that $\mathrm{MMD}^2(P,Q) = 0$ if $P = Q$ and $\mathrm{MMD}^2(P,Q) > 0$ otherwise. Therefore, a one-sided test is sufficient.

[5]: Gretton et al. (2012), *A kernel two-sample test*

[45]: Sriperumbudur et al. (2010), *Hilbert Space Embeddings and Metrics on Probability Measures*

Other well-known examples are goodness-of-fit tests based on the kernelized Stein discrepancy (KSD), which also has a linear-time estimate [14, 35]. In our experiments, we focus on the kernel two-sample test, but point out that our theoretical treatment in Section 2.3 is more general and can be applied to other problems, e.g., KSD goodness-of-fit tests, but also beyond kernel methods.

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

2.3. Selective hypothesis tests

Statistical lore tells us *not to use the same data for learning and testing.* We now discuss whether it is indeed possible to use the same data for selecting a test statistic from a candidate set and conducting the selected test [2]. The key to controllable Type-I errors is that we need to adjust the test threshold to account for the selection event. As before, let $\mathbb{Z}_n$ denote the data we collected. Let $T = \{\tau_i\}_{i\in\mathcal{I}}$ be a countable set of candidate test

[2]: Fithian et al. (2017), *Optimal Inference After Model Selection*

statistics that we evaluate on the data $\mathbb{Z}_n$, and $\{t^i_\alpha\}_{i\in\mathcal{I}}$ the respective test thresholds. Assume that $\{A_i\}_{i\in\mathcal{I}}$ are disjoint *selection events* depending on $\mathbb{Z}_n$ and that their outcomes determine which test statistic out of T we apply. Thus, all the tests and events are generally dependent via $\mathbb{Z}_n$. To define a *well-calibrated* test, we need to control the overall Type-I error, i.e., $P(\text{reject} \mid H_0)$. Using the law of total probability, we can rewrite this in terms of the selected tests

$$P(\text{reject} \mid H_0) = \sum_{i\in\mathcal{I}} P(\tau_i > t^i_\alpha \mid A_i, H_0) P(A_i \mid H_0). \tag{2.2}$$

To control the Type-I error $P(\text{reject} \mid H_0) \leq \alpha$, it thus suffices to control $P(\tau_i > t^i_\alpha \mid A_i, H_0) \leq \alpha$ for each $i \in \mathcal{I}$, i.e., the test thresholds need to take into account the conditioning on the selection event A_i. A *naive* approach would wrongly calibrate the test such that $P(\tau_i > t^i_\alpha \mid H_0) \leq \alpha$, not accounting for the selection A_i and thus would result in an uncontrollable Type-I error. On the other hand, this reasoning directly tells us why data splitting works. There A_i is evaluated on a split of $\mathbb{Z}_n$ that is independent of the split used to compute τ_i and hence $P(\tau_i > t^i_\alpha \mid A_i, H_0) = P(\tau_i > t^i_\alpha \mid H_0)$.

Selecting tests with high power. Our objective in selecting the test statistic is to maximize the power of the selected test. To this end, we start from $d \in \mathbb{N}$ different *base functions* $h_1, \ldots, h_d$. Based on observed data $\mathbb{Z}_n = \{z_1, \ldots, z_n\} \sim P^n$, we can compute d *base* test statistics $\tau_u := \tau_u(\mathbb{Z}_n) = \frac{1}{n}\sum_{i=1}^n h_u(z_i)$ for $u \in [d]$. Let $\boldsymbol{\tau} := (\tau_1, \ldots, \tau_d)^\top$ and $\boldsymbol{\mu} := \mathbb{E}[\boldsymbol{h}(Z)]$, where $\mathbf{h}(Z) = (h_1(Z), \ldots, h_d(Z))^\top$. Asymptotically, we have $\sqrt{n}(\boldsymbol{\tau} - \boldsymbol{\mu}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \Sigma)$, with the variance of the asymptotic distribution given by $\Sigma = \text{Cov}[\boldsymbol{h}(Z)]$.[1] Now, for any $\boldsymbol{\beta} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$ that is independent of $\boldsymbol{\tau}$, the normalized test statistic $\tau_\beta := \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$ is asymptotically normal, i.e., $\sqrt{n}\left(\tau_\beta - \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}\right) \xrightarrow{d} \mathcal{N}(0,1)$. Following our considerations of Section 2.2, the test with the highest power is defined by

$$\boldsymbol{\beta}^\infty := \operatorname*{argmax}_{\|\boldsymbol{\beta}\|=1} \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \frac{\Sigma^{-1}\boldsymbol{\mu}}{\|\Sigma^{-1}\boldsymbol{\mu}\|}, \tag{2.3}$$

where the constraint $\|\boldsymbol{\beta}\| = 1$ is to ensure that the solution is unique, since the objective of the maximization is a homogeneous function of order zero in $\boldsymbol{\beta}$. The explicit form of $\boldsymbol{\beta}^\infty$ is proven in Appendix A.3.2. Obviously, in practice, $\boldsymbol{\mu}$ is not known, so we use an estimate of $\boldsymbol{\mu}$ to select $\boldsymbol{\beta}$. The standard strategy to do so is to split the sample $\mathbb{Z}_n$ into two independent sets and estimate $\boldsymbol{\tau}_{\text{tr}}$ and $\boldsymbol{\tau}_{\text{te}}$, i.e., two independent training and test realizations [6, 8, 36, 37]. One can then choose a suitable $\boldsymbol{\beta}$ by using $\boldsymbol{\tau}_{\text{tr}}$ as a proxy for $\boldsymbol{\mu}$. Then one tests with this $\boldsymbol{\beta}$ and $\boldsymbol{\tau}_{\text{te}}$. However, to our knowledge, there exists no principled way to decide in which proportion to split the data, which will generally influence the power, as shown in our experimental results in Section 2.5.

Our approach to maximizing the utility of the observed dataset is to use it for both learning and testing. To do so, we have to derive an adjustment to the distribution of the statistic under the null, in the spirit of the selective hypothesis testing described above. We will consider

1: In practice, we work with an estimate $\hat{\Sigma}$ of the covariance obtained from $\mathbb{Z}_n$, which is justified since $\sqrt{n}\hat{\Sigma}^{-\frac{1}{2}}(\boldsymbol{\tau} - \boldsymbol{\mu}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, I_d)$ for consistent estimates of the covariance.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [36]: Scetbon et al. (2019), *Comparing distributions: L1 geometry improves kernel two-sample testing*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

three different candidate sets T of test statistics, which are all constructed from the base test statistics $\boldsymbol{\tau}$. To do so, we will work with the asymptotic distribution of $\boldsymbol{\tau}$ under the null. To keep the notation concise, we include the $\sqrt{n}$ dependence into $\boldsymbol{\tau}$. Thus, we will assume $\boldsymbol{\tau} \sim \mathcal{N}(\mathbf{0}, \Sigma)$, where Σ is known and strictly positive. We provide the generalization to singular covariance in Appendix A.5.

To select the test statistics, we maximize the SNR $\tau_{\boldsymbol{\beta}} = \boldsymbol{\beta}^\top \boldsymbol{\tau} / (\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}$ and thus the test power over three different sets of candidate test statistics:

1. $T_{\text{base}} = \left\{ \tau_{\boldsymbol{\beta}} \,|\, \boldsymbol{\beta} \in \{e_1, \dots, e_d\} \right\}$, i.e., we directly select from the base test statistics,
2. $T_{\text{Wald}} = \left\{ \tau_{\boldsymbol{\beta}} \,|\, \|\boldsymbol{\beta}\| = 1 \right\}$, where we allow for arbitrary linear combinations,
3. $T_{\text{OST}} = \left\{ \tau_{\boldsymbol{\beta}} \,|\, \Sigma\boldsymbol{\beta} \geq \mathbf{0}, \|\Sigma\boldsymbol{\beta}\| = 1 \right\}$, where we constrain the allowed values to increase the power (see below).

The rule for selecting the test statistic from these sets is simply to select the one with the highest value. To design selective hypothesis tests, we need to derive suitable selection events and the distribution of the maximum test statistic conditioned on its selection.

2.3.1. Selection from a finite candidate set

We start with $T_{\text{base}} = \left\{ \tau_{\boldsymbol{\beta}} \,|\, \boldsymbol{\beta} \in \{e_1, \dots, e_d\} \right\}$ and use the test statistic $\tau_{\text{base}} = \max_{\tau \in T_{\text{base}}} \tau$. Since the selection is from a countable set and the selected statistic is a projection of $\boldsymbol{\tau}$, we can use the polyhedral lemma of [17] to derive the conditional distributions. Therefore, we denote $u^* = \operatorname{argmax}_{u \in [d]} \frac{\tau_u}{\sigma_u}$, with $\sigma_u := (\Sigma_{uu})^{\frac{1}{2}}$, and obtain $\tau_{\text{base}} = \frac{\tau_{u^*}}{\sigma_{u^*}}$. The following corollary characterizes the conditional distribution. The proof is given in Appendix A.3.1.

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

Corollary 2.3.1 *Let* $\boldsymbol{\tau} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$,

$$\boldsymbol{z} := \boldsymbol{\tau} - \frac{\Sigma e_{u^*} \tau_{u^*}}{\sigma_{u^*}^2}, \quad \mathcal{V}^-(\hat{\boldsymbol{z}}) = \max_{j \in [d], j \neq u^*} \frac{\sigma_{u^*} \hat{\boldsymbol{z}}_j}{\sigma_u^* \sigma_j - \Sigma_{u^* j}}, \tag{2.4}$$

and $TN(\mu, \sigma^2, a, b)$ *denote a normal distribution with mean* μ *and variance* σ^2 *truncated at* a *and* b*. Then the following statement holds:*

$$\left[\frac{\tau_{u^*}}{\sigma_{u^*}} \middle| u^* = \operatorname*{argmax}_{u \in [d]} \frac{\tau_u}{\sigma_u}, \boldsymbol{z} = \hat{\boldsymbol{z}} \right] \overset{d}{=} TN\left(\frac{\mu_{u^*}}{\sigma_{u^*}}, 1, \mathcal{V}^-(\hat{\boldsymbol{z}}), \mathcal{V}^+ = \infty \right). \tag{2.5}$$

This scenario arises, for example, in kernel-based tests when the kernel parameters are chosen from a grid of predefined values determining for example kernel type and bandwidth [5, 6]. Corollary 2.3.1 allows us to test using the same set of data that was used to select the test statistic, by providing the corrected asymptotic distribution (2.5). The only downside is its dependence on the parameter grid. To overcome this limitation, several works have proposed to optimize for the parameters directly [6, 37–40]. Unfortunately, we cannot apply Corollary 2.3.1 directly to this scenario.

[5]: Gretton et al. (2012), *A kernel two-sample test*; [6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*; [38]: Jitkrittum et al. (2018), *Informative Features for Model Comparison*; [39]: Jitkrittum et al. (2017), *A Linear-Time Kernel Goodness-of-Fit Test*; [40]: Jitkrittum et al. (2017), *An Adaptive Test of Independence with Analytic Kernel Embeddings*

2.3.2. Learning from an uncountable candidate set

To allow for more flexible tests, in the following we consider the candidate sets T_{Wald} and T_{OST} that contain uncountably many tests. For these sets, we cannot directly use Equation 2.2 to derive conditional tests, since the probability of selecting some given tests is 0. However, we show that it is possible in both cases to rewrite the test statistic such that we can build conditional tests based on Equation 2.2. First, for T_{Wald}, we rewrite the entire test statistic including the maximization in closed form. Second, for T_{OST} we derive suitable measurable selection events that allow us to rewrite the conditional test statistic in closed form and derive their distributions in Theorem 2.3.2.

Wald Test. We first allow for arbitrary linear combinations of the base test statistics $\boldsymbol{\tau}$. Therefore, define $T_{\text{Wald}} = \left\{\tau_\beta \mid \|\boldsymbol{\beta}\| = 1\right\}$ and $\tau_{\text{Wald}} := \max_{\tau \in T_{\text{Wald}}} \tau$. We denote the optimal $\boldsymbol{\beta}$ for this set as $\boldsymbol{\beta}_{\text{Wald}} := \operatorname{argmax}_{\|\boldsymbol{\beta}\|=1} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$. This optimization problem is the same as in Equation 2.3, hence $\boldsymbol{\beta}_{\text{Wald}} = \frac{\Sigma^{-1}\boldsymbol{\tau}}{\|\Sigma^{-1}\boldsymbol{\tau}\|}$, and we can rewrite the "Wald" test statistic as

$$\tau_{\text{Wald}} = \frac{\boldsymbol{\beta}_{\text{Wald}}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}_{\text{Wald}}^\top \Sigma \boldsymbol{\beta}_{\text{Wald}})^{\frac{1}{2}}} = (\boldsymbol{\tau}^\top \Sigma^{-1} \boldsymbol{\tau})^{\frac{1}{2}} = \|\Sigma^{-\frac{1}{2}} \boldsymbol{\tau}\|. \tag{2.6}$$

Note that T_{Wald} contains uncountably many tests. However, instead of deriving individual conditional distributions, we can directly derive the distribution of the maximized test statistic, since τ_{Wald} can be written in closed form. In fact, under the null, we have $\Sigma^{-\frac{1}{2}}\boldsymbol{\tau} \sim \mathcal{N}(\mathbf{0}, I_d)$ and τ_{Wald} follows a chi distribution with d degrees of freedom. Surprisingly, the presented approach results in the classic Wald test statistic [46], which originally was defined directly in closed form.

[46]: Wald (1943), *Tests of Statistical Hypotheses Concerning Several Parameters When the Number of Observations is Large*

One-sided test (OST). The original Wald test was defined to optimally test $H_0 : \boldsymbol{\mu} = \mathbf{0}$ against the alternative $H_1 : \boldsymbol{\mu} \neq \mathbf{0}$ [46]. Thus, it ignores the fact that we only test against the "one-sided" alternative $\boldsymbol{\mu} \geq \mathbf{0}$, which suffices since we consider linear-time estimates of the squared MMD as test statistics and their population values are non-negative. Multiplying Equation 2.3 with Σ yields $\Sigma\boldsymbol{\beta}^\infty = \frac{\boldsymbol{\mu}}{\|\Sigma^{-1}\boldsymbol{\mu}\|}$. Using $\boldsymbol{\mu} \geq \mathbf{0}$, we find $\Sigma\boldsymbol{\beta}^\infty \geq \mathbf{0}$. Thus, we have prior knowledge over the asymptotically optimal combination $\boldsymbol{\beta}^\infty$. To incorporate this, we a priori constrain the considered values of $\boldsymbol{\beta}$ by the condition $\Sigma\boldsymbol{\beta} \geq \mathbf{0}$. Thus we define $T_{\text{OST}} = \left\{\tau_\beta \mid \Sigma\boldsymbol{\beta} \geq \mathbf{0}, \|\Sigma\boldsymbol{\beta}\| = 1\right\}$, where the norm constraint $\|\Sigma\boldsymbol{\beta}\| = 1$ is added to make the maximum unique. We suggest using the test statistic $\tau_{\text{OST}} := \max_{\tau \in T_{\text{OST}}} \tau$. Before we derive suitable conditional distributions for this test statistic, we rewrite it in a *canonical form*.

Remark 2.3.1 Define $\boldsymbol{\alpha} := \Sigma\boldsymbol{\beta}$, $\boldsymbol{\rho} := \Sigma^{-1}\boldsymbol{\tau}$, and $\Sigma' := \Sigma^{-1}\Sigma\Sigma^{-1} = \Sigma^{-1}$. This implies $\boldsymbol{\rho} \sim \mathcal{N}(\mathbf{0}, \Sigma')$ and $\tau_{\text{OST}} := \max_{\|\Sigma\boldsymbol{\beta}\|=1, \Sigma\boldsymbol{\beta}\geq\mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \max_{\|\boldsymbol{\alpha}\|=1, \boldsymbol{\alpha}\geq\mathbf{0}} \frac{\boldsymbol{\alpha}^\top \boldsymbol{\rho}}{(\boldsymbol{\alpha}^\top \Sigma' \boldsymbol{\alpha})^{\frac{1}{2}}}$.

Thus in the following, we focus on the canonical form, where the constraints are simply positivity constraints. For ease of notation, we stick with $\boldsymbol{\tau}$ and Σ instead of $\boldsymbol{\rho}$ and Σ'. We will thus analyze the distribution of

$$\max_{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta}\geq \mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}}, \tag{2.7}$$

where $\boldsymbol{\beta}^*(\boldsymbol{\tau}) := \operatorname{argmax}_{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta}\geq \mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$. We emphasize that $\boldsymbol{\beta}^*(\boldsymbol{\tau})$ is a random variable that is determined by $\boldsymbol{\tau}$. For conciseness, however, we will use $\boldsymbol{\beta}^*$ and keep the dependency implicit. We find the solution of Equation 2.7 by solving an equivalent convex optimization problem, which we provide in Appendix A.2. We need to characterize the distribution of Equation 2.7 under the null hypothesis, i.e., $\boldsymbol{\tau} \sim \mathcal{N}(\mathbf{0}, \Sigma)$. Since we are not able to give an analytic form for $\boldsymbol{\beta}^*$, it is hard to directly compute the distribution of τ_{OST} as we did for the Wald test. In Subsection 2.3.1 we were able to work around this by deriving the distribution conditioned on the selection of $\boldsymbol{\beta}^*$. In the present case, however, there are uncountably many values that $\boldsymbol{\beta}^*$ can take, so for some the probability is zero. Hence, the reasoning of Equation 2.2 does not apply and we cannot use the PSI framework of [17].

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

Our approach to solving this is the following. Instead of directly conditioning on the explicit value of $\boldsymbol{\beta}^*$, we condition on the *active set*. For a given $\boldsymbol{\beta}^*$, we define the active set as $\mathcal{U} := \{u \mid \beta^*_u \neq 0\} \subseteq [d]$. Note that the active set is a function of $\boldsymbol{\tau}$, defined via Equation 2.7. In Theorem 2.3.2 we show that given the active set, we can derive a closed-form expression for $\boldsymbol{\beta}^*$, and we can characterize the distribution of the test statistic conditioned on the active set. Figure 2.1 depicts the intuition behind Theorem 2.3.2 and Appendix A.1 contains the full proof. In the following, let χ_l denote a chi distribution with l degrees of freedom and $\mathrm{TN}(0, 1, a, \infty)$ denote the distribution of a standard normal RV truncated from below at a, i.e., with CDF $F^a(x) = \frac{\Phi(x)-\Phi(a)}{1-\Phi(a)}$.

Theorem 2.3.2 *Let $\boldsymbol{\tau} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ be a normal RV in $\mathbb{R}^d$ with positive definite covariance matrix Σ. Let $\boldsymbol{\beta}^*$ be defined as in Equation 2.7, $\mathcal{U} := \{u \mid \beta^*_u \neq 0\}$, $l := |\mathcal{U}|$, $\boldsymbol{z} := \left(I_d - \frac{\Sigma \boldsymbol{\beta}^* \boldsymbol{\beta}^{*\top}}{\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*}\right) \boldsymbol{\tau}$, and $\mathcal{V}^-$ as in Corollary 2.3.1. Then, the following statements hold.*

1.) *If $l = 1$:* $\left[\max_{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta}\geq \mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} \,\middle|\, \mathcal{U}, \boldsymbol{z} = \hat{\boldsymbol{z}}\right] \overset{d}{=} TN(0, 1, \mathcal{V}^-(\hat{\boldsymbol{z}}), \infty)$.

2.) *If $l \geq 2$:* $\left[\max_{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta}\geq \mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} \,\middle|\, \mathcal{U}\right] \overset{d}{=} \chi_l$.

With Theorem 2.3.2 and Remark 2.3.1, we are able to define conditional hypothesis tests with the test statistic τ_{OST}. First, we transform our observation $\hat{\boldsymbol{\tau}}$ according to Remark 2.3.1 to obtain it in canonical form, i.e., $\hat{\boldsymbol{\tau}} \to \Sigma^{-1}\hat{\boldsymbol{\tau}}$ and $\Sigma \to \Sigma^{-1}$. Then we solve the optimization problem of Equation 2.7 to find $\boldsymbol{\beta}^*$. Next, we define the active set $\mathcal{U}$, by checking which entries of $\boldsymbol{\beta}^*$ are non-zero. Theorem 2.3.2 characterizes the distribution τ_{OST} conditioned on the selection. We can then define a test threshold t_α

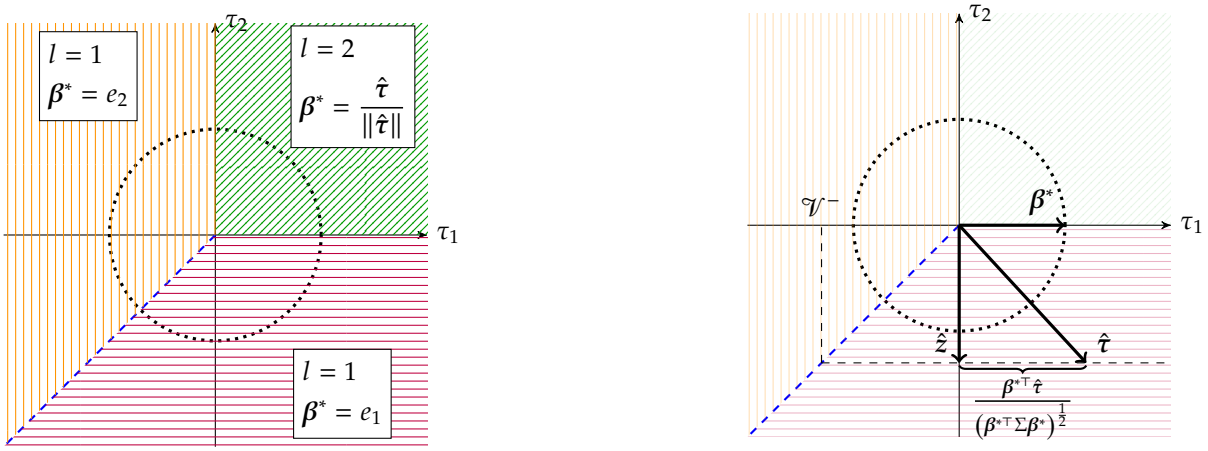


Figure 2.1.: Geometric interpretation of Theorem 2.3.2 for $d = 2$ and unit covariance $\Sigma = I$ (denoted by the black dotted unit-circle). **Left:** If $\hat{\tau}$ is in the positive quadrant (green), the constraints of the optimization are not active and the optimal direction is the same as for the Wald test, hence the distribution of the test statistic follows χ_2. When $\hat{\tau}$ is in the orange or purple zone, one of the constraints is active and β^* is a canonical unit-vector. **Right:** If $l = 1$, for example when only the first direction is active, we additionally condition on $z = \hat{z}$, which is independent of the value of $\beta^{*\top}\tau$ since z is orthogonal to β^*. For the observed value $\hat{z}$, we only select $\beta^* = e_1$ if $\beta^{*\top}\tau \geq \mathcal{V}^-$. If this was not the case, then τ would lie in the orange/vertically lined region and we would select $\beta^* = e_2$. This explains the truncated behavior and is in analogy to the results of [17].

that accounts for the selection of $\mathcal{U}$, i.e.,

$$t_\alpha = \begin{cases} \Phi^{-1}\left((1-\alpha)(1-\Phi(\mathcal{V}^-)) + \Phi(\mathcal{V}^-)\right) & \text{if } |\mathcal{U}| = 1, \\ \Phi_{\chi_l}^{-1}(1-\alpha) & \text{if } |\mathcal{U}| = l \geq 2, \end{cases} \tag{2.8}$$

with $\Phi_{\chi_l}^{-1}$ being the inverse CDF of a chi distribution with l degrees of freedom, which we can evaluate using standard libraries, e.g., [47]. We can then reject the null, if the observed value of the optimized test statistic exceeds this threshold, i.e., $\hat{\tau}_{\text{OST}} > t_\alpha$. We summarize the entire approach in Algorithm 1.

2.4. Related work

The present chapter is best positioned in the context of modern statistical tests with tunable hyperparameters. [6] were the first to propose a kernel two-sample test that optimizes the kernel hyperparameters by maximizing the test power. This influential work has led to further development of optimized kernel-based tests [7, 8, 36–40]. Since any universally consistent binary classifier can be used to construct a valid two-sample test [34, 48], [9, 33] used classification accuracy as a proxy to train machine learning models for two-sample tests. [42, 49] studied this further, and [41] proposed using the difference of a trained deep network's expected logit values as the test statistic for two-sample tests.

All the aforementioned "learn-then-test" approaches optimize hyperparameters (e.g., kernels, weights in a network) on a training set which is split from the full dataset. While the null distribution becomes tractable due to the independence between the optimized hyperparameters and the test set, there is a potential reduction of test power because of a smaller test set. This observation is the main motivation for our consideration of selective hypothesis tests, which allow the full dataset to be used for

[47]: Jones et al. (2001), *SciPy: Open source scientific tools for Python*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [36]: Scetbon et al. (2019), *Comparing distributions: L1 geometry improves kernel two-sample testing*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*; [38]: Jitkrittum et al. (2018), *Informative Features for Model Comparison*; [39]: Jitkrittum et al. (2017), *A Linear-Time Kernel Goodness-of-Fit Test*; [40]: Jitkrittum et al. (2017), *An Adaptive Test of Independence with Analytic Kernel Embeddings*

[34]: Sriperumbudur et al. (2009), *Kernel Choice and Classifiability for RKHS Embeddings of Probability Distributions*; [48]: Friedman (2003), *On multivariate goodness of fit and two sample testing*

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*

[42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*; [49]: Cai et al. (2020), *Two-sample test based on classification probability*

[41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*

both training and testing by correcting for the dependency, as we discuss in Section 2.3.

More broadly, properly assessing the strength of potential associations that have been previously learned from the data falls under an emerging subfield of statistics known as *selective inference* [50]. A seminal work of [17] proposed a post-selection inference (PSI) framework to characterize the valid distribution of a post-selection estimator where model selection is performed by the Lasso [51]. The PSI framework has been applied to kernel tests, albeit in different context, for selecting the most informative features for supervised learning [52, 53], selecting a subset of features that best discriminates two samples [54], as well as selecting a model with the best fit from a list of candidate models [55]. All these applications of the PSI framework consider a finite candidate set. Our Theorem 2.3.2 can be seen as an extension of the previously known results of [17] to uncountable candidate sets. To our knowledge, the presented work is the first to explicitly maximize test power by using the same data for selecting and testing.

[50]: Taylor et al. (2015), *Statistical learning and selective inference*

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

[51]: Tibshirani (1996), *Regression Shrinkage and Selection via the Lasso*

[52]: Yamada et al. (2018), *Post Selection Inference with Kernels*; [53]: Slim et al. (2019), *kernelPSI: a Post-Selection Inference Framework for Nonlinear Variable Selection*

[54]: Yamada et al. (2019), *Post Selection Inference with Incomplete Maximum Mean Discrepancy Estimator*

[55]: Lim et al. (2019), *Kernel Stein Tests for Multiple Model Comparison*

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

Unfortunately, we cannot directly use our results to optimize tests based on complete U-statistics estimates of the MMD, which would be desirable since those estimates have lower variance than the *linear* version we use. The difficulty arises since our method requires asymptotic normality under the null, which is not the case for complete U-statistics, see Equation 1.14. To circumvent this problem, [54] considered incomplete U-statistics [56] and [57] used a Block estimate of the MMD. Under the null, these approaches either have approximately asymptotic normal distribution [54] or require a higher sample size to reach the asymptotic normality [57]. In principle thus our approach is applicable with these methods if one is willed to assume asymptotic normality and to neglect the induced errors. Besides that, since the linear-time estimate has lowest computational cost, it should generally be used in the *large-data, constraint-computation* regime [6]. On the other hand one should consider the other approaches when the computational efforts are not the limiting factor.

[54]: Yamada et al. (2019), *Post Selection Inference with Incomplete Maximum Mean Discrepancy Estimator*

[56]: Janson (1984), *The asymptotic distributions of incomplete U-statistics*

[57]: Zaremba et al. (2013), *B-test: A non-parametric, low variance kernel two-sample test*

[54]: Yamada et al. (2019), *Post Selection Inference with Incomplete Maximum Mean Discrepancy Estimator*

[57]: Zaremba et al. (2013), *B-test: A non-parametric, low variance kernel two-sample test*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

Moreover, under the assumption that $\boldsymbol{\tau} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, similar scenarios have previously been investigated in the traditional statistical literature, but the idea of data splitting is not considered there. In particular, our construction of τ_{Wald} turned out to coincide with the test statistic suggested in [46]. The one-sided version τ_{OST} also has a twin named "*chi-bar-square*" test previously considered in [58]. While their test statistic is constructed to be always non-negative, our τ_{OST} can be negative. Furthermore, they derived the distribution of the test statistic by decomposing the distribution into 2^d selection events, which, however, "*may represent a quite difficult problem*" [59, p. 54]. The approach presented in this chapter circumvents this difficulty by defining a conditional test, which does not require calculating any probability of the selection events. Another difference is that our approach only defines $2^d - 1$ different active sets, by enforcing $\boldsymbol{\beta} \neq \mathbf{0}$. It is instructive to note that there exist other more complicate settings of "learn-then-test" scenarios in which the normality assumption may not hold [9, 41, 42, 49]. Extending our work towards these scenarios remains an open, yet promising problem to consider.

[46]: Wald (1943), *Tests of Statistical Hypotheses Concerning Several Parameters When the Number of Observations is Large*

[58]: Kudo (1963), *A multivariate analogue of the one-sided test*

[59]: Shapiro (1988), *Towards a Unified Theory of Inequality Constrained Testing in Multivariate Analysis*

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*; [42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*; [49]: Cai et al. (2020), *Two-sample test based on classification probability*

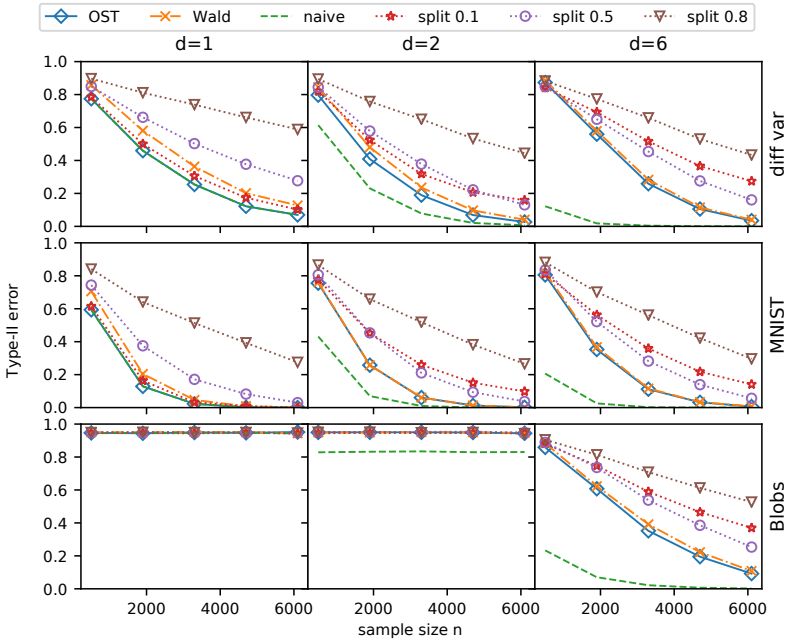


Figure 2.2.: Type-II errors from different experiments. The rows (columns) correspond to different datasets (sets of base kernels). For all considered cases, OST outperforms all the (well-calibrated) competing methods, i.e., SPLIT and WALD.

2.5. Experiments

We demonstrate the advantages of OST over data-splitting approaches and the Wald test with kernel two-sample testing problems as described in Section 2.2. For an extensive description of the experiments we refer to Appendix A.4. We consider three different datasets with different input dimensions p.

1. `DIFF VAR` ($p = 1$): $P = \mathcal{N}(0, 1)$ and $Q = \mathcal{N}(0, 1.5)$.
2. `MNIST` ($p = 49$): We consider downsampled 7x7 images of the MNIST dataset [3], where P contains all the digits and Q only uneven digits.
3. `Blobs` ($p = 2$): A mixture of anisotropic Gaussians where the covariance matrix of the Gaussians have different orientations for P and Q.

[3]: LeCun et al. (2010), *MNIST handwritten digit database*

We denote by k_{lin} the linear kernel, and k_σ the Gaussian kernel with bandwidth σ. For each dataset we consider three different base sets of kernels $\mathcal{K}$ and choose $\tilde{\sigma}$ with the median heuristic:

1. $d = 1$: $\mathcal{K} = [k_{\tilde{\sigma}}]$,
2. $d = 2$: $\mathcal{K} = [k_{\tilde{\sigma}}, k_{\text{lin}}]$,
3. $d = 6$: $\mathcal{K} = [k_{0.25\tilde{\sigma}}, k_{0.5\tilde{\sigma}}, k_{\tilde{\sigma}}, k_{2\tilde{\sigma}}, k_{4\tilde{\sigma}}, k_{\text{lin}}]$.

From the base set of kernels we estimate the base set of test statistics using the linear-time MMD estimates. We compare four different approaches:

1. OST,
2. WALD,
3. SPLIT: Data splitting similar to the approach in [6], but with the same constraints as OST. SPLIT0.1 denotes that 10% of the data are used for learning $\boldsymbol{\beta}^*$ and 90% are used for testing,
4. NAIVE: Similar to splitting but all the data is used for learning and testing without correcting for the dependency. The NAIVE approach is not a well-calibrated test.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

For all the setups we estimate the Type-II error for various sample sizes at a level $\alpha = 0.05$. Error rates are estimated over 5000 independent trials and the results are shown in Figure 2.2. In Appendix A.4.1, we also investigate the Type-I error and show that all methods except for NAIVE correctly control the Type-I error at a rate α. Note that all of the methods scale with $\mathcal{O}(n)$ and the difference in computational cost are negligible.

The experimental results in Figure 2.2 support the main claims of this chapter. First, comparing OST with SPLIT, we conclude that using all the data in an integrated approach is always better (or equally good) than any data splitting approach. Second, comparing OST to WALD, we conclude that adding a priori information ($\boldsymbol{\mu} \geq \mathbf{0}$) to reduce the class of considered tests in a sensible way leads to higher (or equally high) test power. Another interesting observation is in the results of the data-splitting approach. Looking at the DIFF VAR experiment, in the leftmost plot, we can see that the errors are monotonically increasing with the portion of data used to select the test. Since there is only one test, the more data we use to select the test, the higher the error (less data remains for testing). In the middle plot, selection becomes important. Hence, we can see that the gap in performance between all data-splitting approach reduces. However, the order is still consistent with the previous plot. Interestingly, in the rightmost plot, learning becomes even more important. Now, the order changes. If we use too little data for learning the test (SPLIT0.1), the error is high. However, if we use too much data for learning the test (SPLIT0.8), the error will be high as well. That is, there is a trade-off in how much data one should use for selecting the test, and for conducting the test. The optimal proportion depends on the problem and can thus in general not be determined a priori.

In Appendix A.4.3 we also compare τ_{base} to a selection of a base test via the data-splitting approach. Here, SPLIT0.1 consistently performs better than the other split approaches, which is plausible, since the class of considered tests T_{base} is quite small. SPLIT0.1 can even be better than τ_{base}, see discussion in Appendix A.4.3.

In Figure 2.3, we additionally consider a constructed 1-D dataset where the distributions share the first three moments and all uneven moments vanish (see Figure A.4). We compare the results for different sets of $d \in [5]$ base kernels $\mathcal{K} = [k^1_{\text{pol}}, \ldots, k^d_{\text{pol}}]$, where $k^u_{\text{pol}}(x, y) = (x \cdot y)^u$ denotes the homogeneous polynomial kernel of order u. By construction, k^u_{pol} does not contain any information about the difference of P and Q, for $u \neq 4$. Thus, for $d \leq 3$ the well-calibrated methods have a Type-II error of $1 - \alpha$. Only the NAIVE approach already overfits to the noise. Adding the fourth order polynomial adds helpful information and all the methods improve performance. However, adding the fifth order, which again only contains noise, leads to an increased error rate. We interpret this as bias-variance tradeoff that should be considered in the choice of the base set $\mathcal{K}$.

In Appendix A.4.2 we compare how the constraints $\boldsymbol{\beta} \geq \mathbf{0}$, as suggested in [6], work in comparison to the OST approach. We find that while the constraints $\Sigma\boldsymbol{\beta} \geq \mathbf{0}$ lead to consistently higher power than the Wald test, the simple positivity constraints can lead to both, better or worse power depending on the problem. We thus recommend using the OST.

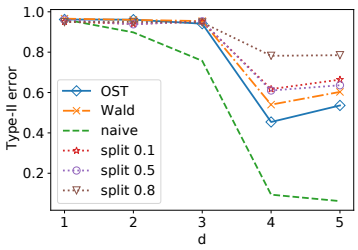


Figure 2.3.: Type-II errors when the first d polynomial kernels are used for a two-sample test with symmetric distributions with the equal covariance (Figure A.4 in the appendix). OST outperforms all the (well-calibrated) competitors.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

Algorithm 1 One-Sided Test (OST)

Input: Σ, $\hat{\boldsymbol{\tau}} = \sqrt{n\widehat{\mathrm{MMD}}^2}(P, Q)$, α
$\hat{\boldsymbol{\tau}} = \Sigma^{-1}\hat{\boldsymbol{\tau}}$ ▷ Apply Remark 2.3.1
$\Sigma = \Sigma^{-1}$ ▷ Apply Remark 2.3.1
$\boldsymbol{\beta}^* = \mathrm{argmax}_{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta}\geq \mathbf{0}} \frac{\boldsymbol{\beta}^\top \hat{\boldsymbol{\tau}}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$
$\mathcal{U} = \{u | u \in [d], \beta^*_u > 0\}$
$\hat{\boldsymbol{z}} = \hat{\boldsymbol{\tau}} - \Sigma\boldsymbol{\beta}^* \frac{\boldsymbol{\beta}^{*\top}\hat{\boldsymbol{\tau}}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*}$
$l = |\mathcal{U}|$
if $l \geq 2$ **then**
  $t_\alpha = \Phi^{-1}_{\chi l}(1-\alpha)$
if $l = 1$ **then**
  $\mathcal{V}^- = \max_{u \notin \mathcal{U}} \frac{\hat{z}_u (\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}{\Sigma_{uu}^{\frac{1}{2}}(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}} - (\Sigma\boldsymbol{\beta}^*)_u}$
  $t_\alpha = \Phi^{-1}\left((1-\alpha)(1-\Phi(\mathcal{V}^-)) + \Phi(\mathcal{V}^-)\right)$
if $t_\alpha < \frac{\boldsymbol{\beta}^{*\top}\hat{\boldsymbol{\tau}}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}$ **then**
  Reject H_0

2.6. Chapter conclusion

Previous work used data splitting to exclude dependencies when optimizing a hypothesis test. This chapter provided the first step towards using all the data for learning and testing. Our approach uses asymptotic joint normality of a predefined set of test statistics to derive the conditional null distributions in closed form. We investigated the example of kernel two-sample tests, where we use linear-time MMD estimates of multiple kernels as a base set of test statistics. We experimentally verified that an integrated approach outperforms the existing data-splitting approach of [6]. Thus data splitting, although theoretically easy to justify, does not efficiently use the data. Further, we experimentally showed that a one-sided test (OST), using prior information about the alternative hypothesis, leads to an increase in test power compared to the more general Wald test. Since the estimates of the base test statistics are linear in the sample size and the null distributions are derived analytically, the whole procedure is computationally cheap. However, it is an open question whether and how this work can be generalized to problems where the class of candidate tests is not directly constructed from a base set of jointly normal test statistics.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

3. A witness two-sample test

The Maximum Mean Discrepancy (MMD) has been the state-of-the-art nonparametric test for tackling the two-sample problem. Its statistic is given by the difference in expectations of the witness function, a real-valued function defined as the mean of kernel evaluations on a set of basis points. Typically the kernel is optimized on a training set, and hypothesis testing is performed on a separate test set to avoid overfitting (i.e., control Type-I error). That is, the test set is used to simultaneously estimate the expectations and define the basis points, while the training set only serves to select the kernel and is discarded. In this chapter, we propose to use the training set to also define the weights and the basis points for better data efficiency. We show that 1) the new test is consistent and has a well-controlled Type-I error; 2) the optimal witness function is given by a precision-weighted mean in the reproducing kernel Hilbert space associated with the kernel; and 3) the test power of the proposed test is comparable or exceeds that of the MMD and other modern tests, as verified empirically on challenging synthetic and real problems (e.g., Higgs data).

3.1. Introduction

In this chapter we continue to tackle the *two-sample problem*: given two samples, do they differ significantly enough that we can conclude they originate from two different distributions (see Section 1.2)? This is a common task in many life sciences such as bioinformatics and cancer diagnosis [27]. To decide the two-sample problem, one can perform a *two-sample test*, whose goal is to reject the *null hypothesis* "the probability distributions are the same" in favor of the *alternative hypothesis* "the probability distributions are not the same" based on data [1]. To quantitatively assess this, one defines a *test statistic* and estimates its value on the observed samples. If we know (or are able to simulate) the distribution of this test statistic under the null, we can reject the null if the observed value is significantly larger than what we would expect if the null was true. Traditional hypothesis tests have test statistics that are defined a priori. A simple example are t- or z-tests, which only test whether the empirical means of both samples differ significantly [1] (see Example 1.2.1). However, such a simple approach is not sufficient to detect differences of distributions with the same mean but, for example, different variance, skewness, or kurtosis.

[27]: Borgwardt et al. (2006), *Integrating structured biological data by Kernel Maximum Mean Discrepancy*

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

[1]: Lehmann et al. (2005), *Testing statistical hypotheses*

To detect any differences between two distributions we focus on two categories of tests closely tied to machine learning, but note that various other methods exist [60, 61]. The former first transforms data into a high-dimensional feature space based on a pre-defined feature map, e.g., kernel function. The test statistics can then be defined in terms of the embeddings of the two distributions in the feature space [5, 62]. The

[60]: Friedman et al. (1979), *Multivariate Generalizations of the Wald-Wolfowitz and Smirnov Two-Sample Tests*; [61]: Chen et al. (2017), *A New Graph-Based Two-Sample Test for Multivariate and Object Data*

[5]: Gretton et al. (2012), *A kernel two-sample test*; [62]: Harchaoui et al. (2008), *Testing for homogeneity with kernel Fisher discriminant analysis*

second approach instead learns to distinguish the two distributions by training a classifier, e.g., via a deep neural network. Based on the learned model, the test statistics is then computed on an independent set of samples, e.g., through data splitting [9, 33, 41, 48].

The popular kernel two-sample test based on the *Maximum Mean Discrepancy* (MMD) in principle does not require data splitting and is completely determined a priori by a positive definite kernel function [5]. However, recent research has shown that optimizing the kernel function on a held-out dataset increases the power of the MMD-based tests [6–8, 42]. Thus most modern MMD-based tests are used as two-stage procedures with data splitting, although it is in principle possible to use the entire dataset for kernel selection and testing [63, 64] as we have seen in Chapter 2. In particular, as opposed to Chapter 2 we will now focus on the quadratic time MMD estimates (Section 1.3). [65] recently proposed an aggregated MMD two-sample test working without data splitting.

To obtain maximally significant results in the testing phase, we advocate that in a 'two-stage' two-sample test, it is more appropriate to learn a test statistic that is as problem-specific as possible. For the MMD tests, this means that we advocate to learn a one-dimensional witness function and not a kernel. To formalize this, we propose a general two-stage witness two-sample test (WiTS test). The introduced WiTS test has the following properties:

- The test statistic is the difference in means of a one-dimensional function called the *witness function* and is thus asymptotically normal under *both* the null and alternative hypotheses. This allows for a simple theoretical treatment (cf. Theorem 3.3.1 and Proposition 3.3.2).
- Compared to [7] and [8], the WiTS test has a simpler test power criterion as a training objective and test thresholds can be simulated more efficiently (cf. Section 3.3 & Equation 3.7).
- The WiTS tests empirically outperform the benchmark tests of [8] and classification-based tests on challenging synthetic and real problems, e.g., Higgs data (cf. Figure 3.3).

The rest of this chapter is organized as follows. Section 3.2 reviews MMD based two-sample tests with a focus on the witness function and discusses our motivation. We then present the general WiTS test framework in Section 3.3, followed by a specific example in Section 3.4. Next, we discuss related work in detail in Section 3.5. Finally, Section 3.6 provides the empirical results comparing the proposed WiTS tests to existing ones on several benchmark datasets. The code to reproduce the experiments of this chapter is published under `https://github.com/jmkuebler/wits-test`.

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*; [41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*; [48]: Friedman (2003), *On multivariate goodness of fit and two sample testing*

[5]: Gretton et al. (2012), *A kernel two-sample test*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*

[63]: Fromont et al. (2012), *Kernels Based Tests with Non-asymptotic Bootstrap Approaches for Two-sample Problems*; [64]: Fromont et al. (2013), *The two-sample problem for Poisson processes: Adaptive tests with a nonasymptotic wild bootstrap approach*

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

3.2. Background and motivation

Notation and definitions. We again consider the two-sample problem as introduced in Chapter 1. When we consider data splitting, we use $\mathbb{X}_{\text{tr}}, \mathbb{X}_{\text{te}}$ and $\mathbb{Y}_{\text{tr}}, \mathbb{Y}_{\text{te}}$ to denote the disjoint training and test sets with $n = n_{\text{tr}} + n_{\text{te}}$, $m = m_{\text{tr}} + m_{\text{te}}$. We define the shorthands $[n] := \{1, \dots, n\}$, $\mathbb{Z} = \{\mathbb{X}, \mathbb{Y}\}$, $\mathbb{Z}_{\text{tr}} = \{\mathbb{X}_{\text{tr}}, \mathbb{Y}_{\text{tr}}\}$ and $\mathbb{Z}_{\text{te}} = \{\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}\}$.

Although most of our analysis applies to more general function spaces, we will consider a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ with positive definite kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (Definition 1.3.1). By the Riesz representation theorem, we have that $f(x) = \langle f, k(x, \cdot)\rangle$ for all $x \in \mathcal{X}$ and $f \in \mathcal{H}$. We assume that

$$\textbf{(A1):}\ \mathbb{E}\left[k(X,X)\right] < \infty,\ \mathbb{E}\left[k(Y,Y)\right] < \infty$$

holds. **(A1)** ensures the kernel mean embeddings of P and Q exist, i.e., $\mu_P = \mathbb{E}\left[k(X,\cdot)\right], \mu_Q = \mathbb{E}\left[k(Y,\cdot)\right]$, and that we can write $\mathbb{E}\left[f(X)\right] = \langle f, \mu_P\rangle$ for all $f \in \mathcal{H}$ [19]. For a sample $\mathbb{X}$, we define the empirical mean embedding as $\mu_{\mathbb{X}} = \frac{1}{|\mathbb{X}|}\sum_{x\in\mathbb{X}} k(x,\cdot)$.

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

MMD and witness function. We introduced the MMD in Section 1.3. The function that *witnesses* the MMD is $\operatorname{argmax}_{f\in\mathcal{H},\|f\|\leq 1}\{\mathbb{E}\left[f(X)\right] - \mathbb{E}\left[f(Y)\right]\} = (\mu_P - \mu_Q)/\|\mu_P - \mu_Q\|$ [5, Sec. 2.3]. We define its unnormalized version as $h_k^{P,Q} = \mu_P - \mu_Q$ and obtain

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\begin{aligned}\mathrm{MMD}^2 &= \langle \mu_P - \mu_Q, \mu_P - \mu_Q\rangle = \langle \mu_P - \mu_Q, h_k^{P,Q}\rangle \\ &= \mathbb{E}\left[h_k^{P,Q}(X)\right] - \mathbb{E}\left[h_k^{P,Q}(Y)\right].\end{aligned} \tag{3.1}$$

With a *characteristic* kernel [45], $\mu_P = \mu_Q$ if and only if $P = Q$. Hence, the squared MMD (3.1) can be used to test the hypothesis $H_0 : P = Q$ against $H_1 : P \neq Q$.

[45]: Sriperumbudur et al. (2010), *Hilbert Space Embeddings and Metrics on Probability Measures*

MMD-BOOT test statistics. We can estimate the squared MMD (3.1) by replacing the witness $h_k^{P,Q}$ and the expectations in Equation 3.1 with their empirical counterparts $h_k^{\mathbb{Z}} = \mu_{\mathbb{X}} - \mu_{\mathbb{Y}}$ and obtain a biased estimate (Equation 1.9)

$$\begin{aligned}&\widehat{\mathrm{MMD}}^2_{'boot'}(\mathbb{Z}|k) \\ =&\frac{1}{n}\sum_{x\in\mathbb{X}} h_k^{\mathbb{Z}}(x) - \frac{1}{m}\sum_{y\in\mathbb{Y}} h_k^{\mathbb{Z}}(y) \\ =&\left\langle \frac{1}{n}\sum_{x\in\mathbb{X}} k(x,\cdot) - \frac{1}{m}\sum_{y\in\mathbb{Y}} k(y,\cdot), h_k^{\mathbb{Z}}(\cdot)\right\rangle \\ =&\frac{1}{n^2}\sum_{x,x'\in\mathbb{X}} k(x,x') + \frac{1}{m^2}\sum_{y,y'\in\mathbb{Y}} k(y,y') - \frac{2}{nm}\sum_{x\in\mathbb{X},y\in\mathbb{Y}} k(x,y).\end{aligned} \tag{3.2}$$

[5]: Gretton et al. (2012), *A kernel two-sample test*

The latter expression is a sum of V-statistics and up to the biased terms where $x = x'$ or $y = y'$ equals the unbiased U-statistic (Equation 1.10), which is the standard MMD estimate [5]. The witness itself depends on the same data $\mathbb{Z}$ used to evaluate the test statistic (3.2). Hence to compute the test threshold, the null distribution has to be simulated via permutation of the samples (or bootstrapping) [5]. Thus, we refer to this approach as 'mmd-boot'.[1]

1: Our naming convention should emphasizes that the asymptotic distribution cannot be evaluated in closed-form and hence we necessarily need to simulate it. Note, however, that in practice often permutations are used [7] and it is not necessary to completely simulate the distribution from scratch.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*; [7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

OPT-MMD-BOOT test statistics. A drawback of 'mmd-boot' is that the kernel k has to be chosen a priori before observing the data. Kernel choice, however, critically affects the performance of MMD based two-sample tests [6–8, 37] as we have also seen in the previous chapter. Since for

the quadratic-time MMD estimates no similar procedure to Chapter 2 is known, it is common to split the data into $\mathbb{Z} = (\mathbb{Z}_{\text{tr}}, \mathbb{Z}_{\text{te}})$ and optimize the kernel only on the held-out set $\mathbb{Z}_{\text{tr}}$. For the moment, without specifying how the kernel is optimized, we denote the resulting optimized kernel as k_{tr} with a subscript tr to indicate that it depends on the training data. After optimizing the kernel, the standard 'mmd-boot' test is conducted on $\mathbb{Z}_{\text{te}}$ with the optimized kernel k_{tr} [7, 8]. Hence, the empirical expectations and witness function in Equation 3.2 are still dependent on the same data $\mathbb{Z}_{\text{te}}$, and the null distribution still has to be bootstrapped, for the same reason as in the case of 'mmd-boot'. We will refer to this approach as 'opt-mmd-boot' with the test statistic

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$\widehat{\text{MMD}}^2_{'opt-boot'}(\mathbb{Z}_{\text{te}}|k_{\text{tr}}) = \frac{1}{n_{\text{te}}} \sum_{x \in \mathbb{X}_{\text{te}}} h^{\mathbb{Z}_{\text{te}}}_{k_{\text{tr}}}(x) - \frac{1}{m_{\text{te}}} \sum_{y \in \mathbb{Y}_{\text{te}}} h^{\mathbb{Z}_{\text{te}}}_{k_{\text{tr}}}(y). \tag{3.3}$$

Our Motivation. This is the starting point of our investigations: Although the kernel is optimized, it is still a multidimensional representation of the data. While this makes the test statistic applicable to other problems [8, 42], features that contain little information about the differences of P and Q will mainly add noise to the test statistic. Generally, the noisier the test statistic, the harder it is to obtain significant test results. Motivated by this drawback, we propose to formulate a test statistic that is more specific to the observed difference in $\mathbb{Z}_{\text{tr}}$. Being more specific to the training data (that is all we know about P and Q), comes at the risk of overfitting, which we mitigate via regularization and model selection (cf. Section 3.3). Specifically for MMD, after the kernel is optimized, *we define the witness directly on the training data* by replacing $h^{\mathbb{Z}_{\text{te}}}_{k_{\text{tr}}}$ with $h^{\mathbb{Z}_{\text{tr}}}_{k_{\text{tr}}} = \frac{1}{n_{\text{tr}}} \sum_{x \in \mathbb{X}_{\text{tr}}} k_{\text{tr}}(x, \cdot) - \frac{1}{m_{\text{tr}}} \sum_{y \in \mathbb{Y}_{\text{tr}}} k_{\text{tr}}(y, \cdot)$. We call this 'opt-mmd-witness':

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*

$$\widehat{\text{MMD}}^2_{'opt-witness'}\left(\mathbb{Z}_{\text{te}}|h^{\mathbb{Z}_{\text{tr}}}_{k_{\text{tr}}}\right) = \frac{1}{n_{\text{te}}} \sum_{x \in \mathbb{X}_{\text{te}}} h^{\mathbb{Z}_{\text{tr}}}_{k_{\text{tr}}}(x) - \frac{1}{m_{\text{te}}} \sum_{y \in \mathbb{Y}_{\text{te}}} h^{\mathbb{Z}_{\text{tr}}}_{k_{\text{tr}}}(y). \tag{3.4}$$

This test statistic comes with numerous advantages. Firstly, the expectations (defined via $\mathbb{Z}_{\text{te}}$) are now independent of the witness function (defined via $\mathbb{Z}_{\text{tr}}$). Thus, the test statistic is asymptotically normal. Secondly, as we will see in the following sections, Equation 3.4 allows us to compute asymptotic test thresholds in closed form and allows for a simpler derivation of a test power criterion than in the case of 'opt-mmd-boot' [7, 8]. Lastly, our empirical results suggest that 'opt-mmd-witness' outperforms 'opt-mmd-boot' on datasets considered in [8].

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

3.3. Witness two-sample test (WiTS test)

Similar to Equation 3.4, the WiTS tests we propose are by design two-stage procedures: In *Stage I*, we learn the witness function h with the training data $\mathbb{Z}_{\text{tr}}$. This ensures that h is independent of the test data $\mathbb{Z}_{\text{te}}$, used in

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Stage II to define a test statistic

$$\hat{\tau}(\mathbb{Z}_{\text{te}}|h) \propto \frac{1}{n_{\text{te}}} \sum_{x \in \mathbb{X}_{\text{te}}} h(x) - \frac{1}{m_{\text{te}}} \sum_{y \in \mathbb{Y}_{\text{te}}} h(y). \tag{3.5}$$

We reject the null hypothesis $H_0 : P = Q$ if the observed value is larger than a test threshold. We start presenting Stage II and analyze the test's asymptotic power for a given function h. Then, we will use this test power criterion as the objective when optimizing the witness function in Stage I.

3.3.1. Stage II - testing with the witness function

We start with a basic result on asymptotic normality of empirical means ([32], Proof in Appendix B.1.1).

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

Theorem 3.3.1 (Asymptotic normality of WiTS test) *For a witness function* $h : \mathcal{X} \to \mathbb{R}$, *let* $\sigma_P^2 := Var[h(X)]$ *and* $\sigma_Q^2 := Var[h(Y)]$ *such that* $0 < \sigma_P^2, \sigma_Q^2 < \infty$. *Let* $\{X_i\}_{i \in [n]} \overset{i.i.d.}{\sim} P$, $\{Y_j\}_{j \in [m]} \overset{i.i.d.}{\sim} Q$, *and* $c := \frac{n}{n+m} \in (0,1)$ *as* $n + m \to \infty$. *Denote by* $\bar{h}_P := \mathbb{E}[h(X)]$ *and* $\bar{h}_Q := \mathbb{E}[h(Y)]$. *We define the empirical means* $\hat{h}_P^n := \frac{1}{n} \sum_{i \in [n]} h(X_i)$, $\hat{h}_Q^m := \frac{1}{m} \sum_{i \in [m]} h(Y_i)$ *and denote the sample variance as* $\hat{\sigma}_c^2(h) := \hat{\sigma}_P^2/c + \hat{\sigma}_Q^2/(1-c)$. *Then*

$$\frac{\sqrt{n+m}}{\hat{\sigma}_c(h)} \left[\left(\hat{h}_P^n - \bar{h}_P \right) - \left(\hat{h}_Q^m - \bar{h}_Q \right) \right] \xrightarrow{d} \mathcal{N}(0,1).$$

For any *fixed* h and for sufficiently large sample sizes, we can thus work with the asymptotic distribution of test statistics of the form $\tau(\cdot|h)$ in Equation 3.5 to compute test thresholds and derive an asymptotic test-power objective for choosing h based on the training data $\mathbb{Z}_{\text{tr}}$ in Stage I. Data splitting ensures that h is independent of $\mathbb{Z}_{\text{te}}$, which is necessary for Theorem 3.3.1 to hold. In the following, to make the comparison between different choices of h easier, we consider the standardized test statistic on the test samples $\mathbb{Z}_{\text{te}}$

$$\tau(\mathbb{Z}_{\text{te}}|h) = \sqrt{n_{\text{te}} + m_{\text{te}}} \frac{\frac{1}{n_{\text{te}}} \sum_{x \in \mathbb{X}_{\text{te}}} h(x) - \frac{1}{m_{\text{te}}} \sum_{y \in \mathbb{Y}_{\text{te}}} h(y)}{\hat{\sigma}_c(h)},$$

where $c = \frac{n_{\text{te}}}{n_{\text{te}} + m_{\text{te}}}$ and $\hat{\sigma}_c(h)$ is the empirical estimate of the pooled variance as in Theorem 3.3.1 based on $\mathbb{Z}_{\text{te}}$. To control the Type-I error at a significance level α, we need to find a *test threshold* t_α such that $P(\tau(\mathbb{Z}_{\text{te}}|h) > t_\alpha | H_0) \leq \alpha$. By Theorem 3.3.1, we can define the threshold to be the $(1-\alpha)$ quantile of the asymptotic null distribution. Under the null hypothesis we have $\bar{h}_P = \bar{h}_Q$ and obtain $t_\alpha = \Phi^{-1}(1-\alpha)$ where Φ^{-1} denotes the inverse CDF of the standard normal.

Note that we only consider a "one-sided" test, since we choose h in stage I with the appropriate sign, i.e., such that it has larger expectation under $\mathbb{X}_{\text{tr}}$ than under $\mathbb{Y}_{\text{tr}}$. A "two-sided" test ignores this and may lead to a reduction in test power.

We reject the null hypothesis $H_0 : P = Q$ if $\tau(\mathbb{Z}_{\text{te}}|h) > t_\alpha$. As an advantage of the asymptotic normality under the alternative and the closed form of

the threshold of our test, we can write the asymptotic Type-II error rate in closed form, similar as in Example 1.2.1 and Figure 1.1

$$P(\tau(\mathbb{Z}_{\text{te}}|h) < t_\alpha) \approx \Phi\left(\Phi^{-1}(1-\alpha) - \sqrt{n_{\text{te}} + m_{\text{te}}}\,\frac{\bar{h}_P - \bar{h}_Q}{\sigma_c(h)}\right). \tag{3.6}$$

An important consideration in designing a hypothesis test is test consistency. A hypothesis test is called consistent, if for a fixed alternative hypothesis, its test power converges to one as sample size goes to infinity. With Equation 3.6, we can characterize for which functions h the statistic τ_h leads to a consistent test.

Proposition 3.3.2 (Consistency of WiTS test) *Assume $0 < \sigma_c(h) < \infty$, where $\sigma_c(h)$ is defined in Theorem 3.3.1. A WiTS test based on h is consistent against a fixed alternative hypothesis $P \neq Q$ if and only if $\bar{h}_P > \bar{h}_Q$.*

Proposition 3.3.2 ensures that, for a given alternative hypothesis, our proposed test will eventually (in the limit of the sample size) reject the null hypothesis H_0 when it is false. Associated with this notion is the *test power*, the probability that the test rejects H_0 when it is false; this quantity is equivalent to 1− Type-II error. Defining the *signal-to-noise* ratio $\text{SNR}(h) = \frac{\bar{h}_P - \bar{h}_Q}{\sigma_c(h)}$, it follows from Equation 3.6 that the asymptotic test power of our test is

$$\beta_h \approx 1 - \Phi\left(\Phi^{-1}(1-\alpha) - \sqrt{n_{\text{te}} + m_{\text{te}}}\,\text{SNR}(h)\right). \tag{3.7}$$

Since Φ increases monotonically, the test power grows monotonically with the *signal-to-noise* ratio (SNR).

3.3.2. Stage I - finding an optimal witness

We now propose an objective to find an optimal witness function. Based on our test power consideration, we argue that in the first stage one should find a witness by maximizing a, possibly regularized, empirical estimate of the SNR in Equation 3.7. Let $\mathcal{F}$ be a function class containing candidates for the witness. We propose using the witness $\hat{h}_\lambda$ defined as

$$\begin{aligned} \hat{h}_\lambda &= \operatorname*{argmax}_{f \in \mathcal{F}} \frac{\bar{f}_{\mathbb{X}_{\text{tr}}} - \bar{f}_{\mathbb{Y}_{\text{tr}}}}{\sigma_{c,\lambda}^{\mathbb{Z}_{\text{tr}}}(f)}, \\ \text{with } \bar{f}_{\mathbb{X}_{\text{tr}}} &= \frac{1}{n_{\text{tr}}} \sum_{x \in \mathbb{X}_{\text{tr}}} f(x),\ \bar{f}_{\mathbb{Y}_{\text{tr}}} = \frac{1}{m_{\text{tr}}} \sum_{y \in \mathbb{Y}_{\text{tr}}} f(y), \end{aligned} \tag{3.8}$$

and $\sigma_{c,\lambda}^{\mathbb{Z}_{\text{tr}}}(f) = ((\sigma_c^{\mathbb{Z}_{\text{tr}}}(f)^2 + \lambda\Omega(f))^{\frac{1}{2}}$, where $\sigma_c^{\mathbb{Z}_{\text{tr}}}(f)$ corresponds to $\hat{\sigma}_c(h)$ defined in Theorem 3.3.1 and Ω is a regularizer. We remark that the optimal witness is generally not uniquely defined since the SNR is invariant to rescaling the function. Correctly rejecting H_0 when it is false is at the core of hypothesis testing. Our choice of maximizing the SNR in Equation 3.7 is in line with this principle: it leads to a test that maximizes the asymptotic test power. By contrast, while other objectives such as classification loss[9, 33], softmax loss [41], or the MMD statistic itself

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*

[41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*

[5]: Gretton et al. (2012), *A kernel two-sample test*

[5], can be used to learn the witness function, their relationship to the test power may be indirect. We will come back to other loss functions in Chapter 4.

OPT-MMD-Witness. In Section 1.5 we discussed the asymptotic test power criterion used by [7, 8] to optimize the kernel for quadratic time MMD estimates. This is given by $J = \frac{\text{MMD}^2}{\sigma_{H_1}}$ in Equation 1.17. In Appendix B.1.5, we examine this quantity in more detail, and show that

$$J(P, Q|k) = 1/\sqrt{2}\,\text{SNR}\left(h_k^{P,Q}\right). \tag{3.9}$$

For a given class of kernels and corresponding (empirical) MMD witnesses, this implies that selecting the optimal witness according to our SNR criterion leads to the same function as first optimizing the kernel with the J criterion and defining the MMD witness afterwards.

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Algorithm 2 WiTS test with 'kfda-witness'

Input: $\mathbb{X}, \mathbb{Y}, \alpha$, paramGrid, r
$\mathbb{X}_{\text{tr}}, \mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{tr}}, \mathbb{Y}_{\text{te}} \leftarrow \text{RandomSplit}(\mathbb{X}, \mathbb{Y}, r)$
Optionally perform model selection
$k, \lambda \leftarrow \text{GridSearchCV}(\text{paramGrid}, \mathbb{Z}_{\text{tr}})$
Stage I - Optimize Witness
$h \leftarrow \text{kfdaWitness}(\mathbb{Z}_{\text{tr}}, k, \lambda)$ ▷ App. Alg.4
Stage II - Test
return: $\text{witnessTest}(\mathbb{Z}_{\text{te}}, h, \alpha)$

function $\text{witnessTest}(\mathbb{Z}_{\text{te}}, h(\cdot), \alpha, B = 200)$
  $h_{\mathbb{Z}_{\text{te}}} \leftarrow [h(z) \text{ for } z \text{ in } \mathbb{Z}_{\text{te}}]$
  $\tau \leftarrow \text{mean}(h_{\mathbb{Z}_{\text{te}}}[: n_{\text{te}}]) - \text{mean}(h_{\mathbb{Z}_{\text{te}}}[n_{\text{te}} :])$
  $p \leftarrow 1/(B+1)$ ▷ simulate p-value via permutations
  for i in $[B]$ **do**
    $h_{\mathbb{Z}_{\text{te}}} \leftarrow \text{Permute}(h_{\mathbb{Z}_{\text{te}}})$
    if $\text{mean}(h_{\mathbb{Z}_{\text{te}}}[: n_{\text{te}}]) - \text{mean}(h_{\mathbb{Z}_{\text{te}}}[n_{\text{te}} :]) \geq \tau$ **then**
      $p \leftarrow p + 1/(B+1)$
  if $p \leq \alpha$ **then** return: 1 **else** return: 0

Model selection and optimization. The choice of function class $\mathcal{F}$ and regularization parameter λ affects the learned witness in Equation 3.8. We therefore recommend that practitioners use standard tools for model selection such as cross-validation (CV) for finding suitable "hyperparameters" and to validate that the learned witness actually has a high SNR (see also Chapter 4). CV ensures that the witness actually learns the differences between P and Q and does not solely overfit the training data. Model selection on $\mathbb{Z}_{\text{tr}}$ is legit since in Stage II we only use $\mathbb{Z}_{\text{te}}$, which are independent of $\mathbb{Z}_{\text{tr}}$. While this is also possible in classifier two-sample tests [9], in the standard 'mmd-boot' this is not done.

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*

Our objective Equation 3.7 can be used with a variety of function classes $\mathcal{F}$. For instance, $\mathcal{F}$ can be defined based on an RKHS, or parameterized by a deep neural network. Note that optimization methods to maximize Equation 3.8 are generally function class specific, and may require an iterative procedure. In Chapter 4 we will show that equivalently to maximizing the SNR we can also minimize a squared loss, which makes it easy to use standard libraries. When $\mathcal{F}$ is an RKHS, we can derive the closed-form solution to Equation 3.8, as shall be explained in Section 3.4. Algorithm 2 shows the general procedure for the two-stage WiTS test.

Permutation-based thresholds. For our theoretical analysis we used the asymptotic threshold. However, the witness is also chosen in a data-dependent manner. Thus, we generally recommend to simulate the

threshold via *permutations* in order to ensure Type-I error control at finite sample size. In this case, for simplicity and ease of implementation, we compute the test statistic without normalization and simply take the difference in means. We first compute the value of the witness function on all points in $\mathbb{Z}_{\text{te}}$ and store it in an array. Then we compute the simplified test statistic by taking the difference in means of $\mathbb{X}_{\text{te}}$ and $\mathbb{Y}_{\text{te}}$ (as computed from the array that stores all the witness evaluations). We then iterate over $B \in \mathbb{N}$ permutation runs to estimate the p-value of the computed test statistic. For each run, we permute the array storing the witness evaluations, and then compute the difference in means of the first n_{te} and the last m_{te} entries. We then estimate the p-value as detailed in Equation 1.3 and Lemma 1.2.1. After all permutations, if the p-value is smaller or equal than α, we reject (see Algorithm 2). By Lemma 1.2.1 this correctly controls Type-I errors. Since for this procedure we only need to compute the witness once on each data point the overall cost is $O((n_{\text{te}} + m_{\text{te}})B)$. Note that simulating the null for 'mmd-boot' instead has cost $O((n_{\text{te}} + m_{\text{te}})^2 B)$ [8, Sec. 5].

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

3.4. KFDA-witness

In this section, we consider the function class in Equation 3.8 to be an RKHS, and show that this choice leads to a closed form solution for the optimal witness. To start, let $\mathcal{H}$ be an RKHS associated with a positive definite kernel k (see Section 3.2). Additionally to the mean embeddings μ_P, μ_Q, we define the (centered) covariance operator $\Sigma_P = \mathbb{E}\left[k(X,\cdot) \otimes k(X,\cdot)\right] - \mu_P \otimes \mu_P$ (analogously for Q) whose existence is ensured by Assumption **(A1)** [19, Sec. 3]. For any function in the RKHS we then have $\mathbb{E}\left[f(X)\right] = \langle \mu_P, f \rangle$ and $\text{Var}[f(X)] = \langle f, \Sigma_P f \rangle$, and analogously for Q. We define the pooled covariance operator $\Sigma = \frac{\Sigma_P}{c} + \frac{\Sigma_Q}{1-c}$. Then for all $f \in \mathcal{H}$ with non-zero variance we have

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

$$\text{SNR}(f) = \frac{\langle \mu_P - \mu_Q, f \rangle}{\langle f, \Sigma f \rangle^{\frac{1}{2}}}, \tag{3.10}$$

where SNR is defined in Equation 3.7. This objective corresponds to Kernel Fisher discriminant analysis (KFDA)'s learning objective [66]. For singular covariance operator the SNR can diverge, and for infinite-dimensional RKHS, the empirical estimation of the covariance operator is ill-posed. In the following, we therefore consider a regularized ($\lambda > 0$) version of Equation 3.10 and call its solution *(regularized) KFDA witness*:

[66]: Mika et al. (1999), *Fisher discriminant analysis with kernels*

$$h_\lambda = \underset{f \in \mathcal{H}}{\text{argmax}} \frac{\langle \mu_P - \mu_Q, f \rangle}{\langle f, (\Sigma + \lambda I) f \rangle^{\frac{1}{2}}}. \tag{3.11}$$

The solution of Equation 3.11 is given by the solution to the generalized eigenvalue problem $(\Sigma + \lambda I) h_\lambda = \gamma(\mu_P - \mu_Q)$ [67, Sec.3.2], thus

[67]: Mika (2003), *Kernel Fisher Discriminants*

$$h_\lambda = \gamma(\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q), \tag{3.12}$$

where $\gamma > 0$ is an arbitrary positive constant we fix to 1, unless stated otherwise. We will refer to the test with the witness function h_λ as the 'kfda-witness' test. Next, we show how we can estimate the KFDA-witness with the training data.

Estimation of the KFDA witness. Let $\mathbb{Z}_{\text{tr}} = \{x_1, \ldots, x_{n_{\text{tr}}}, y_1, \ldots, y_{m_{\text{tr}}}\}$ denote the pooled training sample and K denote the kernel matrix such that $K_{ij} = k(z_i, z_j)$ for $i, j \in [n_{\text{tr}} + m_{\text{tr}}]$. Further, we define $\delta = (\frac{1}{n_{\text{tr}}}, \ldots, \frac{1}{n_{\text{tr}}}, -\frac{1}{m_{\text{tr}}}, \ldots, -\frac{1}{m_{\text{tr}}})^\top \in \mathbb{R}^{n_{\text{tr}}+m_{\text{tr}}}$. For $l \in \{n_{\text{tr}}, m_{\text{tr}}\}$, we define the idempotent centering matrix $P_l = I_l - l^{-1}\mathbf{1}_l\mathbf{1}_l^\top$, where I_l denotes the identity operator and $\mathbf{1}_l$ the l dimensional vector with all ones. With this we define the $(n_{\text{tr}} + m_{\text{tr}}) \times (n_{\text{tr}} + m_{\text{tr}})$ matrix $N_c = \begin{pmatrix} \frac{1}{c}P_{n_{\text{tr}}} & 0 \\ 0 & \frac{1}{1-c}P_{m_{\text{tr}}} \end{pmatrix}$. Using the representer theorem [68], we can empirically estimate the KFDA witness (more detail in Appendix B.1.3) as

[68]: Schölkopf et al. (2001), *A Generalized Representer Theorem*

$$\hat{h}_\lambda(\cdot) = \sum_{i=1}^{n_{\text{tr}}+m_{\text{tr}}} \hat{\alpha}_i k(z_i, \cdot), \tag{3.13}$$

$$\hat{\alpha} = \left(\frac{KN_cK}{n_{\text{tr}} + m_{\text{tr}}} + \lambda K\right)^{-1} K\delta. \tag{3.14}$$

$\hat{h}_\lambda(\cdot)$ can be viewed as a precision-weighted (inverse covariance) mean of the embeddings of the basis points $\mathbb{Z}_{\text{tr}}$ in the RKHS. Since $\mu_{\mathbb{X}_{\text{tr}}}$, $\mu_{\mathbb{Y}_{\text{tr}}}$, and $\hat{\Sigma}$ are consistent estimates of μ_P, μ_Q, and Σ, for fixed regularization, we have $\hat{h}_\lambda \to h_\lambda = (\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q)$ (see Appendix B.1.4). For the asymptotic witness h_λ we can compute the difference in expectation under P and Q in closed form: $\bar{h}_{\lambda,P} - \bar{h}_{\lambda,Q} = \langle \mu_P - \mu_Q, h_\lambda \rangle = \langle \mu_P - \mu_Q, (\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q) \rangle$. This difference is positive, and hence by Proposition 3.3.2 we obtain a consistent WiTS test, if and only if $\mu_P \neq \mu_Q$. We can ensure this for arbitrary $P \neq Q$ by using a *characteristic* kernel [45], the same condition as for MMD-based tests.

[45]: Sriperumbudur et al. (2010), *Hilbert Space Embeddings and Metrics on Probability Measures*

Based on our experimental results, we observe that, for a fixed kernel k, fixed regularization $\lambda > 0$, and sufficiently large sample size, the splitting ratio $r = 1/2$ appears to give the highest test power in many cases, compared to other values of r. Generally, identifying the optimal splitting ratio remains an open problem. We observe (middle panel of Figure 3.1) that if we include model selection in stage I, it is favorable to use more than half of the data for the first stage, i.e., $r > 1/2$. However, since we cannot quantify how much "*more*" data we should use, we generally recommend using a 50/50 split.

The cost of computing the exact solution $\hat{\alpha}$ in Equation 3.13 is $O((n_{\text{tr}} + m_{\text{tr}})^2)$ in space (storing the kernel matrix) and $O((n_{\text{tr}} + m_{\text{tr}})^3)$ time (matrix inversion). In Appendix B.3, we adopt recent advances in large-scale kernel machines [69, 70] to obtain approximate solutions with lower time and space complexity and thus scale to large datasets. Using the Nyström approximation [71] to approximate the solution and approximately solving it with conjugate gradient, we obtain a complexity of $O((n_{\text{tr}} + m_{\text{tr}})Mt + M^3)$ in time and $O(M^2)$ in space, where M denotes the number of Nyström centers and t the number of conjugate gradient iterations. For stage II we then only need $(n_{\text{te}} + m_{\text{te}})M$ kernel evaluations to compute the test statistic. This makes our approach scalable to large-scale dataset.[2]

[69]: Rudi et al. (2017), *FALKON: An Optimal Large Scale Kernel Method*; [70]: Meanti et al. (2020), *Kernel Methods Through the Roof: Handling Billions of Points Efficiently*

[71]: Williams et al. (2000), *Using the Nyström Method to Speed Up Kernel Machines*

2: Recently, [72] also proposed a Nyström approximation of the kernel mean embedding to speed up the MMD estimation.

Connection of 'opt-mmd-witness' and 'kfda-witness'. To emphasize the relationship between optimizing the MMD and using KFDA, consider a fixed kernel k and denote by $\mathcal{A}$ the set of bounded positive operators on $\mathcal{H}_k$. We consider the nonparametric class of kernels

Table 3.1.: Overview of kernel-based two-sample tests. *a priori* means that the kernel/regularization is chosen independently of the data. The present work proposes the "witness" methods.

Method	kernel choice	reg. λ	witness obj.	witness estim.	test data	threshold
'kfda-witness'(proposed)	CV	CV	SNR	$\mathbb{Z}_{\text{tr}}$	$\mathbb{Z}_{\text{te}}$	analytic
'kfda-boot'[62]	a priori	a priori	SNR	$\mathbb{Z}$ (implicit)	$\mathbb{Z}$	bootstrap
'mmd-boot'[5]	a priori	-	MMD	$\mathbb{Z}$ (implicit)	$\mathbb{Z}$	bootstrap
'opt-mmd-witness'(proposed)	J with $\mathbb{Z}_{\text{tr}}$	-	MMD	$\mathbb{Z}_{\text{tr}}$	$\mathbb{Z}_{\text{te}}$	analytic
'opt-mmd-boot'[7]	J with $\mathbb{Z}_{\text{tr}}$	-	MMD	$\mathbb{Z}_{\text{te}}$ (implicit)	$\mathbb{Z}_{\text{te}}$	bootstrap

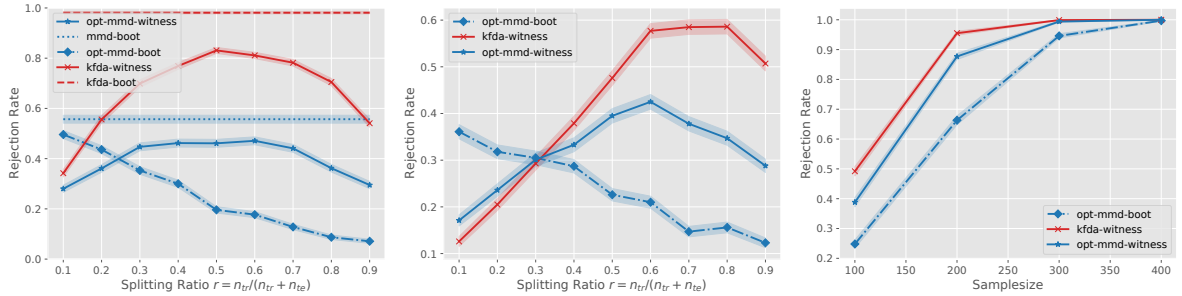


Figure 3.1.: Instructive experiments on "Blobs" dataset. **Left:** Fixed kernel and fixed regularization for sample size $n = m = 100$. **Middle:** For multiple candidate kernels ($\mathcal{K}_{10}$) kernel optimization becomes more important and the difference of 'kfda-witness' and 'opt-mmd-witness' becomes smaller. Further, 'opt-mmd-witness' already outperforms 'opt-mmd-boot'. **Right:** Same kernels as in the middle figure and $r = 1/2$. All the tests are consistent, i.e., converge to power equal 1.

$\mathcal{K} = \{k_A | k_A(x, y) = \langle Ak(x, \cdot), Ak(y, \cdot)\rangle, A \in \mathcal{A}\}$. For this class of kernels, we show in Appendix B.1.6 that using 'opt-mmd-witness' leads to the same witness function as using 'kfda-witness'.

'kfda-boot'. It turns out that KFDA-like test statistics were considered before [62], but in settings without data splitting. Indeed, for a fixed k and $\lambda > 0$, we can use the whole data, i.e., $\mathbb{X}, \mathbb{Y}$ for learning the witness $(\hat{\Sigma} + \lambda)^{-1}(\mu_{\mathbb{X}} - \mu_{\mathbb{Y}})$ and computing the test statistic (empirical mean difference). The test statistic thus is $\tau_{'kfda-boot'} = \langle \mu_{\mathbb{X}} - \mu_{\mathbb{Y}}, (\hat{\Sigma} + \lambda)^{-1}(\mu_{\mathbb{X}} - \mu_{\mathbb{Y}})\rangle$, and we call its population version $\text{KFDA}^2(P, Q | k, \lambda)$. This, is the test statistic as studied by [62]. As for 'mmd-boot', the same data is used for estimating the witness and computing the mean difference, hence Theorem 3.3.1 does not hold anymore. We thus need to bootstrap the null distribution via permutations of the samples; thus, we refer to it as 'kfda-boot'. 'kfda-boot' has similar drawbacks as 'mmd-boot': 1. simulating the null distribution via permutations has cost $\mathcal{O}((n + m)^3 B)$ for $B \in \mathbb{N}$ draws from the null distribution; and 2. we have to fix k and λ a priori, and their choices strongly affect the test power. [62] do not provide guidance for how to choose k and λ.

[62]: Harchaoui et al. (2008), *Testing for homogeneity with kernel Fisher discriminant analysis*

[62]: Harchaoui et al. (2008), *Testing for homogeneity with kernel Fisher discriminant analysis*

[62]: Harchaoui et al. (2008), *Testing for homogeneity with kernel Fisher discriminant analysis*

3.5. Related work

Besides the kernel-based tests we discussed so far, [73] proposed tests based on *smooth characteristic functions* (SCF), and projected *mean embeddings* (ME) of the distributions where the mean embeddings are projected to J-dimensional Euclidean vectors for $J \in \mathbb{N}$. In fact, the normalized ME statistic in [73, Eq. 13] can be seen as a variant of the KFDA where the function classes is restricted by the J projection directions. Note that

[73]: Chwialkowski et al. (2015), *Fast Two-Sample Testing with Analytic Representations of Probability Measures*

[73]: Chwialkowski et al. (2015), *Fast Two-Sample Testing with Analytic Representations of Probability Measures*

for a finite-dimensional RKHS and without regularization, 'kfda-boot' corresponds to the Hotelling's T^2 statistic [74]. [37] improve the approach of [73] by optimizing the features in the first stage. However, they also discard the training data after learning the J projection directions. [42] propose to learn a deep finite-dimensional representation of the data and to use this for a subsequent MMD or KFDA test. However, their training objective does not directly maximize the test power [42, Sec. 3.1.1]. [8] propose a deep version of 'opt-mmd-boot'. They learn a deep-kernel ('mmd-d') of the form

[74]: Hotelling (1931), *The Generalization of Student's Ratio*

[37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

[42]: Kirchler et al. (2020), *Two-sample Testing Using Deep Learning*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$k_\omega(x, x') = \left[(1-\epsilon)\kappa(\phi_\omega(x), \phi_\omega(x'))) + \epsilon\right] q(x, x'), \tag{3.15}$$

where $\epsilon \in (0, 1)$, κ and q are Gaussian kernels and ϕ_ω is a deep representation optimized via the criterion J, see Appendix B.1.5. They also consider a version called 'mmd-o' which is $k_\omega(x, x') = \kappa(\phi_\omega(x), \phi_\omega(x'))$ and conclude that learning a full kernel (they advocate 'mmd-d') is better than learning a one-dimensional representation.

Most of the aforementioned works focus on developing a practical testing procedure for a specific dataset at hand. However, there also exist more theoretical work on the statistical optimality of different kernel-based approaches. [75] show that a *moderated* MMD approach (which is related to KFDA) leads to optimal rates when testing against local alternatives. A similar discussion can be found in the long version of [76, Sec.5.1]. This resonates our findings, that a witness based on KFDA is more powerful than simply using the MMD witness. Furthermore, [77] show how the choice of scaling parameter in Gaussian kernels affects the statistical optimality. However, such theoretically optimal tests oftentimes are unpractical to use. [75], for example, requires the eigendecomposition of the kernel function, which generally is hard to obtain. Furthermore, without data splitting also these works cannot find a good kernel function.

[75]: Balasubramanian et al. (2021), *On the Optimality of Kernel-Embedding Based Goodness-of-Fit Tests*

[76]: Harchaoui et al. (2008), *Testing for Homogeneity with Kernel Fisher Discriminant Analysis*

[77]: Li et al. (2019), *On the Optimality of Gaussian Kernel Based Nonparametric Tests against Smooth Alternatives*

Since our proposed witness function is one-dimensional, it is closely related to classification based two-sample tests [9, 33, 41, 48, 49]. [9] proposed learning a deep classifier and using its classification accuracy as test statistic. We refer to this as 'c2st-s', where 's' stands for sign. The method has two drawbacks. First, classification loss does optimize the 0-1 loss, whereas we directly maximize test power [9, Remark 2].[3] Second, it only uses the sign of the classification function and thus neglects information by weighting all points equally. [41] address the second issue by considering the network's output before thresholding the function into a classifier. They train with a softmax loss, which also does not directly address test power. The connections of these methods to kernel-based tests were also thoroughly discussed by [8] and, in accordance, we refer to the approach of [41] as 'c2st-l'.

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*; [41]: Cheng et al. (2019), *Classification Logit Two-sample Testing by Neural Networks*; [48]: Friedman (2003), *On multivariate goodness of fit and two sample testing*; [49]: Cai et al. (2020), *Two-sample test based on classification probability*

3: In Chapter 4 we will show that optimizing a cross-entropy loss actually optimizes test power.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

3.6. Experiments

We empirically assess the test power of the proposed WiTS tests in two settings. First, we perform instructive experiments to highlight the differences of the methods summarized in Table 3.1. Second, we perform benchmark experiments on two challenging datasets and compare the

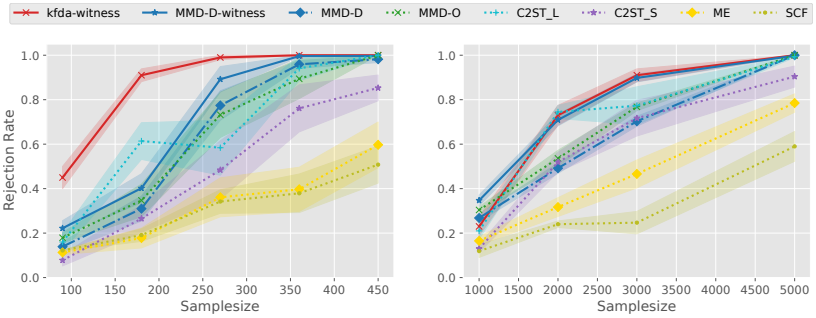


Figure 3.3.: Benchmark experiments adapted from [8] **Left:** Blobs, **Right:** HIGGS. Computing the MMD witness after kernel optimization and performing a witness test ('mmd-d-witness') improves the test power over 'mmd-d'. Directly learning the 'kfda-witness' also leads to high power.

performance of the introduced WiTS tests ('kfda-witness' and 'opt-mmd-witness') to the benchmarks ('mmd-d', 'mmd-o', 'me', 'scf', 'c2st-s', 'c2st-l') introduced in Section 3.5. For the benchmarks, we reuse the implementation provided by [8] without changing any hyperparameters. Throughout our experiments we set the level $\alpha = 0.05$. Appendix B.2 contains experiments for correct Type-I error control. The shaded regions contain ± one standard error of the estimates.[4]

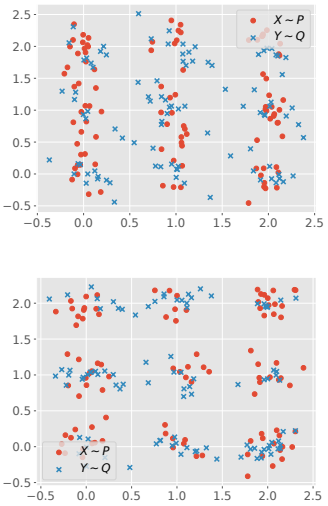


Figure 3.2.: Top: Draws from Blobs dataset for the instructive experiments. The distributions are mixtures of nine Gaussians, with anisotropic covariance (but the same covariance matrix across blobs). The covariance matrix of Q is rotated by $\theta = \pi/4$ relative to the covariance matrix of P. To simulate the null hypothesis we use $\theta = 0$, which corresponds to drawing both samples from P. **Bottom:** Blobs dataset used for Figure 3.3 as suggested by [8, Figure 1]. In this case, P has isotropic Gaussian, the blobs in Q are anisotropic and have different covariance matrices. To simulate the null hypothesis, we draw both samples from P.

Instructive experiments. In Figure 3.1, we consider a **Blobs** dataset [6] where P and Q are mixtures of nine anisotropic 2-d Gaussians with Q having the covariance matrix rotated by an angle $\theta = \pi/4$, see Figure 3.2. For the left panel of Figure 3.1, we consider a single Gaussian kernel $k_\sigma(x, x') = \exp\left(-\|x - x'\|^2/\sigma^2\right)$ with bandwidth $\sigma = 0.2$ and a regularization parameter for the 'kfda' methods of $\lambda = 10^{-2}$ (in Appendix B.2 we additionally show the effect of the regularization in Figure B.2. Note that for $\lambda \to \infty$, 'kfda' and 'mmd' methods coincide). We showcase the effect of varying splitting ratios r when the kernel is fixed a-priori (thus we can apply 'mmd-boot' and 'kfda-boot'). With fixed kernel, 'opt-mmd-boot' essentially discards the training data. We estimate the test power (rejection rate) with fixed overall sample size $n = m = 100$. We observe that the witness methods achieve highest power for a 50/50 split, given a fixed kernel and fixed regularization. We also observe that the boot approaches outperform the witness methods in this case.

However, in practice, it is unlikely that we can pick a powerful kernel and regularization *a priori*. Therefore, for the middle panel of Figure 3.1, we optimize the kernel function over a class of kernels $\mathcal{K}_{10}$ consisting of ten Gaussian kernels with bandwidths on a logarithmic range from 10^{-3} to 10^{1}. Additionally, for 'kfda-witness' we cross-validate over five candidate regularizations on a log range from 10^{-4} to 10^{3}. In this case, the witness methods attain the highest power at a splitting ratio $r > 1/2$, and 'opt-mmd-witness' outperforms 'opt-mmd-boot' for the majority of splitting ratios and also globally. For the right panel, we use the same setting, but fix the splitting ratio at $r = 1/2$ and vary the sample size. As we expect, all tests are consistent and we observe that both WiTS test approaches outperform 'opt-mmd-boot' at a 50/50 split.

Benchmark Experiments. [8] benchmarked several deep classification two-sample tests ('c2st-l', 'c2st-c') against MMD with an optimized deep kernel ('mmd-d', 'mmd-o') and the optimized tests ('me', 'scf') of [37]. We implement 'opt-mmd-witness' on top of their proposed method

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

4: Note that in Figure 3.3 we used different approaches to estimate the rejection rates, see Appendix B.2. This explains that at the same rejection rate we can have differently large errors.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

'mmd-d', which optimizes a deep kernel [8, Section 5]. Therefore after the kernel optimization, we use the training data to define the MMD witness function (Equation 3.4) and then proceed with 'WitnessTest' from Algorithm 2. We also run 'kfda-witness' with grid search over the same kernels and regularization as for the previous experiments. We run the experiments on two benchmarks. First, an adopted **Blobs** problem, with multiple different covariances [8, Figure 1] (see Figure 3.2), introduced to show the limitations of MMD with translation-invariant kernels. Second, the **Higgs** dataset [78] where "we compare the jet ϕ-momenta distribution ($d = 4$) of the background process, P, which lacks Higgs bosons, to the corresponding distribution Q for the process that produces Higgs bosons" (cited from [8]). For the Higgs dataset we consider sample sizes larger than a thousand per class. To speed up the computation of the 'kfda-witness', we approximate the solution with $M = 500$ Nyström centers, see Appendix B.3, which underlines the scalability of our approach. For both datasets we observe higher power of the WiTS tests we propose, see Figure 3.3. We emphasize that we used the implementation of [8], without changing the deep architecture or any hyperparameters.

[78]: Baldi et al. (2014), *Searching for exotic particles in high-energy physics with deep learning*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

3.7. Chapter Conclusion

We introduced a principled approach to learn optimal witness functions for two-sample testing. The approach consists of two-stages: First, we learn a witness on a subset of the observations by maximizing a test-power criterion. In the second stage, we simply test whether the witness function attains the same mean on the test samples, and efficiently simulate the null distribution via permutations. We further showed how to adopt recent tests based on optimized Maximum Mean Discrepancy into a witness two-sample test. [8] advocated optimizing a (deep) kernel in the training stage. Our experiments show, however, that explicitly learning a one-dimensional witness can perform better than learning a high-dimensional representation (a kernel function) in the training stage.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Our results extend beyond kernel methods since we derive a principled objective to train a one-dimensional function optimal for two-sample testing. This objective and the proposed testing procedure can be applied with any function class. The proposed framework thus not only allows domain experts to perform two-sample tests with the models most suitable to the data at hand, but can also easily incorporate model selection techniques developed for classification and regression tasks to optimize for the best parameter settings. This will be the focus of the next chapter.

4. AutoML two-sample test

Two-sample tests are important in statistics and machine learning, both as tools for scientific discovery as well as to detect distribution shifts. This led to the development of many sophisticated test procedures going beyond the standard supervised learning frameworks, whose usage can require specialized knowledge about two-sample testing. We use a simple test that takes the mean discrepancy of a witness function as the test statistic and prove that minimizing a squared loss leads to a witness with optimal testing power. This allows us to leverage recent advancements in AutoML. Without any user input about the problems at hand, and using the same method for all our experiments, our AutoML two-sample test achieves competitive performance on a diverse distribution shift benchmark as well as on challenging two-sample testing problems.

We provide an implementation of the AutoML two-sample test in the Python package `autotst`.

4.1. Introduction

Testing whether two distributions are the same based on data is a fundamental problem in data science. A classical application is to test whether two differently treated groups have the same characteristics or not [79–81]. Testing independence of two random variables can also be phrased as a two-sample problem by testing whether the joint distribution equals the product of the marginal distributions [82]. A more recent application in machine learning is to detect distribution shifts, i.e., whether the distribution a model was trained on equals the distribution the model is deployed on [18, 83, 84].

Classical methods have a fixed test statistic that makes strong parametric assumptions. For example, Student's two-sample t-test only tests whether the distributions have equal mean, assuming both distributions follow a normal distribution with the same (but unknown) variance (Example 1.2.1). With modern datasets, which are often high-dimensional, such test cannot be applied because the strong assumptions are often not justified. Nonparametric kernel-based test such as the Maximum Mean Discrepancy (MMD) [5] are very flexible and, theoretically, can detect differences of any kind given enough data. However, this generality often harms test power at finite data size. This can simply be understood in terms of a classical bias-variance tradeoff. As we have discussed in the prior chapters, it is common to optimize a kernel (Section 1.5) or a witness function (Chapter 3). However, the derived objective as well as optimizing a kernel function are no standard tasks in machine learning and no automated packages exist, making it hard for practitioners to apply them.

[79]: Student (1908), *The probable error of a mean*; [80]: Welch (1947), *The generalization of 'STUDENT'S' problem when several different population varlances are involved*; [81]: Golland et al. (2003), *Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies*

[82]: Gretton et al. (2005), *Measuring statistical dependence with Hilbert-Schmidt norms*

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*; [83]: Lipton et al. (2018), *Detecting and Correcting for Label Shift with Black Box Predictors*; [84]: Koch et al. (2022), *Hidden in Plain Sight: Subgroup Shifts Escape OOD Detection*

[5]: Gretton et al. (2012), *A kernel two-sample test*

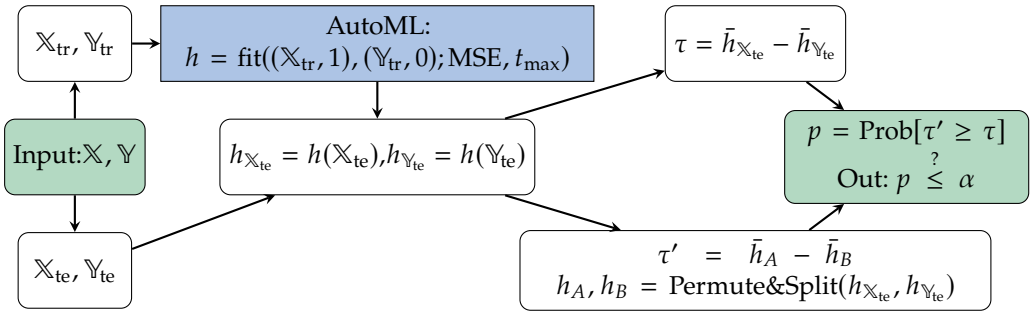


Figure 4.1.: AutoML two-sample test: $\mathbb{X}, \mathbb{Y}$ denotes the available data from P and Q, which is first split into two parts of equal size. A witness $h : \mathcal{X} \to \mathbb{R}$ is trained using a (weighted) squared loss Equation 4.6, denoted by MSE, and using AutoML to maximize predictive performance. Users can easily control important properties, for example the maximal runtime $t_{\max}$. The test statistic τ is the difference in means on the test sets. Permuting the data and recomputing τ allows the estimation of the p-values. The null hypothesis $P = Q$ is rejected if $p \leq \alpha$.

Tests that fit well into the standard machine learning pipeline are based on the classification accuracy. First, a classifier is trained to detect the difference between the two samples, and then its accuracy on a held-out set is taken as a test statistic [9, 33, 49, 81, 85]. [8] argued, however, that optimizing classification accuracy does not directly optimize test power and considered this one reason why kernel-based test outperform classifier tests. Our results of Chapter 3 challenged this and we considered the mean of an optimized witness function as test statistic finding that kernels are not necessary for good performance. Generally, such two-stage procedures are very intuitive and arguably also how a human would approach the two-sample problem on complicated data. One could look at some part of the data, try to come up with a simple hypothesis, and then try to test its significance on held-out data (Section 0.1).

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*; [49]: Cai et al. (2020), *Two-sample test based on classification probability*; [81]: Golland et al. (2003), *Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies*; [85]: Hediger et al. (2022), *On the use of random forest for two-sample testing*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Despite the recent progress in the theoretical understanding of machine learning-based two-sample tests [8, 33], there is still little guidance on how to apply these tests in practice and a substantial amount of engineering and expertise is required to implement them. On the contrary, in supervised learning, namely regression and classification, the past years have shown tremendous advancements in making machine learning models applicable essentially without any expert knowledge leading to the field of Automated Machine Learning (AutoML) [86–88]. The goal of AutoML is to automate the full machine learning pipeline: Data cleaning, feature engineering and augmentation, model search, hyperparameter optimization, and model ensembling [89]. All of it with the goal of achieving the best possible predictive performance on unseen data.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*

[86]: Feurer et al. (2015), *Efficient and Robust Automated Machine Learning*; [87]: Hutter et al. (2019), *Automated machine learning: methods, systems, challenges*; [88]: He et al. (2021), *AutoML: A survey of the state-of-the-art*

[89]: Dietterich (2000), *Ensemble methods in machine learning*

The goal of this chapter is to bring the advancements of AutoML research to the field of two-sample testing. Our main contributions are:

1. We prove that minimizing a squared loss is equivalent to maximizing the unwieldy signal-to-noise ratio, which determines the asymptotic test power of a witness two-sample test (Subsection 4.3.1).
2. Thanks to the former result we can use AutoML to learn the test statistic, thereby harnessing the power of many advancements in machine learning such as hyperparameter optimization, bagging, and ensemble learning in a user-friendly manner (Subsection 4.3.2).
3. Our test is usable without any specific knowledge and skills in two-sample testing. Users can easily specify how many resources they want to use when learning the test, for example the maximal

training time (Subsection 4.3.2 and Section 4.5). Furthermore, one can easily interpret the results (Subsection 4.3.3).
4. We extensively study the empirical performance of our approach first by considering the two low-dimensional datasets Blob and Higgs followed by running a large benchmark on a variety of distribution shifts on MNIST and CIFAR10 data. We observe very competitive performance without any manual adjustment of hyper-parameters. Our experiments also show that a continuous witness outperforms commonly used binary classifiers (Section 4.5).
5. We provide the Python Package `autotst` implementing our testing pipeline.

The proposed testing pipeline is described in Figure 4.1: First, the two samples are split into training and test sets. Then a *witness* function h is trained by first labeling samples in $\mathbb{X}_{\text{tr}}$ with 1 and samples from $\mathbb{Y}_{\text{tr}}$ with 0 and then minimizing a (weighted) Mean Squared Error (MSE) to maximize test power, see Section 4.3 for further details. To maximize the predictive performance and to require as little user input as possible, we use AutoGluon [90], an existing AutoML framework, when optimizing the witness. Our test statistic is then simply the difference in means of the test sets $\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}$, see Section 4.2. p-values are computed via permutation of the samples [81], which is a standard technique in two-sample testing (Section 1.2).

[90]: Erickson et al. (2020), *Autogluon-tabular: Robust and accurate automl for structured data*

[81]: Golland et al. (2003), *Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies*

4.2. Preliminaries

We use the same notation as introduced in Section 3.2. Unless otherwise stated, we assume that the data is split in equal halves, which is the default approach [8, 9].

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*

Witness two-sample test. We consider a witness-based hypothesis test as introduced in the previous chapter, however, only consider the unnormalized version. Given a function $h : \mathcal{X} \to \mathbb{R}$, called *witness*, the *mean discrepancy* is

$$\tau(P, Q \mid h) = \mathbb{E}_{X\sim P}\left[h(X)\right] - \mathbb{E}_{Y\sim Q}\left[h(Y)\right], \tag{4.1}$$

and we use its empirical estimate on the test set as test statistic

$$\tau(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}} \mid h) = \frac{1}{n_{\text{te}}} \sum_{x\in\mathbb{X}_{\text{te}}} h(x) - \frac{1}{m_{\text{te}}} \sum_{y\in\mathbb{Y}_{\text{te}}} h(y). \tag{4.2}$$

As we show in Section 4.4, this test statistic can be seen as a continuous extension of classifier two-sample tests [9]. We assume that $c = \frac{n_{\text{te}}}{n_{\text{te}}+m_{\text{te}}}$ converges to a constant. With $\sigma_c^2(h) = \frac{(1-c)\text{Var}_{X\sim P}[h(X)]+c\text{Var}_{Y\sim Q}[h(Y)]}{c(1-c)}$ we showed in Theorem 3.3.1 that the test statistic is asymptotically normally distributed

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*

$$\sqrt{n_{\text{te}} + m_{\text{te}}}\left[\tau(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}} \mid h) - \tau(P, Q \mid h)\right] \xrightarrow{d} \mathcal{N}\left(0, \sigma_c^2(h)\right). \tag{4.3}$$

Let us for now assume that we know $\sigma_c(h)$. For any level $\alpha \in (0, 1)$ we can set the *analytic* test threshold to $t_\alpha = \frac{\sigma_c(h)}{\sqrt{n_{\text{te}}+m_{\text{te}}}}\Phi^{-1}(1-\alpha)$, where Φ

denotes the CDF of a standard normal and Φ^{-1} its inverse. We can then compute the asymptotic probability of rejecting as:

$$\begin{aligned}\Pr[\text{reject}] &= \Pr[\tau(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}|h) > t_\alpha] \\ &\to \Phi\left(\sqrt{n_{\text{te}} + m_{\text{te}}}\frac{\tau(P, Q \mid h)}{\sigma_c(h)} - \Phi^{-1}(1-\alpha)\right).\end{aligned} \tag{4.4}$$

Under the null hypothesis $P = Q$ we have $\tau(P, Q \mid h) = 0$. Therefore, Equation 4.4 reduces to $\Phi\left(-\Phi^{-1}(1-\alpha)\right) = 1 - \Phi\left(\Phi^{-1}(1-\alpha)\right) = \alpha$. Hence, the asymptotic test correctly controls Type-I error. On the other hand, for given $P \neq Q$, Equation 4.4 corresponds to the test power. Since Φ is a monotonically increasing function, the test power is maximized by the witness h that maximizes

$$\text{SNR}(h) = \frac{\tau(P, Q \mid h)}{\sigma_c(h)}, \tag{4.5}$$

where SNR is the Signal-to-Noise Ratio. In Section 3.4 we showed that the optimal witness can be learned when using kernel methods (using kernel Fisher Discriminant Analysis), but it was left open how this can be done efficiently with other machine learning frameworks. Such an SNR is not commonly implemented and also common approaches like mini-batching are not easily adapted, as an estimate of the SNR based on a mini batch would be a biased estimate. In the next section, we show how to circumvent this and optimize a squared loss instead.

4.3. The AutoML two-sample test

4.3.1. Equivalence of squared loss and signal-noise ratio

Since it is known for linear models that minimizing a squared loss over two labelled samples is equivalent to Fisher Discriminant Analysis [67, 91], we attempt to find a more general relation between the squared loss and the SNR. Our goal is to use the squared loss as the optimization objective when learning the witness. Let $c = \frac{n_{\text{tr}}}{n_{\text{tr}}+m_{\text{tr}}}$ analogously to the above. Let us mark all data from P with a label '1' and all data from Q with a label '0'. We define the following (weighted) squared loss

[67]: Mika (2003), *Kernel Fisher Discriminants*; [91]: Duda et al. (2001), *Pattern classification, 2nd Edition*

$$L_{P,Q,c}(h) = \quad (1-c)\,\mathbb{E}_{X\sim P}\left[(1-h(X))^2\right] + c\,\mathbb{E}_{Y\sim Q}\left[(0-h(Y))^2\right], \tag{4.6}$$

Note that the weights $(1-c)$ and c are swapped as it will be more important to fit the set with fewer samples. Given a function h, notice that shifting and scaling it leaves the SNR (4.5) invariant. We can then show the following relationship of its squared loss and its SNR.

Lemma 4.3.1 *Let the function h be fixed. We apply the linear transformation $h \to \gamma h + \nu$ with $\gamma \in \mathbb{R}$ and $\nu \in \mathbb{R}$. Let (γ^*, ν^*) be the minimum of the quadratic function $(\gamma, \nu) \mapsto L(\gamma h + \nu)$. Then, the following holds true:*

$$L(\gamma^* h + \nu^*) = \frac{c(1-c)}{1 + SNR(h)^2}.$$

We defer the proof to Appendix C.1.

Let us assume that the supports of the two distributions P, Q overlap. Hence, for any function the loss $L_{P,Q,c}$ is strictly positive. Assume that h^* is the function that minimizes the loss over all possible functions. This implies that $\gamma^* = 1$ and $\nu^* = 0$, as otherwise one could still improve the loss by scaling or shifting. Thus, by Lemma 4.3.1 we have:

Proposition 4.3.2 *Assume that h^* minimizes the squared loss (4.6). Then h^* maximizes the signal-to-noise ratio, i.e.,*

$$L(h^*) = \min_h L(h) \implies SNR(h^*) = \max_h SNR(h).$$

Proof. A solution that minimizes the loss has $\bar{h}^*_P \geq \bar{h}^*_Q$ and hence a non-negative SNR. Assume there exists $\tilde{h}$ such that $\mathrm{SNR}(\tilde{h}) > \mathrm{SNR}(h^*)$. Then Lemma 4.3.1 implies the existence of $\tilde{\gamma}, \tilde{\nu}$ such that $L(\tilde{\gamma}\tilde{h} + \tilde{\nu}) < L(h^*)$, which is a contradiction. □

We can further derive a closed-form solution for the population optimal witness:

Proposition 1 (Optimal Witness) Assume P and Q have densities $p(x)$ and $q(x)$. The function minimizing Equation 4.6 is

$$h^*(x) = \frac{(1-c)p(x)}{(1-c)p(x) + c\, q(x)}. \tag{4.7}$$

Proof. We rewrite Equation 4.6 as

$$L(h) = \int_{\mathcal{X}} (1-c)p(x)(1-h(x))^2 + c\, q(x)h^2(x)\, dx.$$

Minimizing the integrand for each x yields the claimed result. A similar result was obtained by [92]. □

[92]: Mao et al. (2019), *On the Effectiveness of Least Squares Generative Adversarial Networks*

Remark 4.3.1 Consider the balanced case $c = 1/2$, i.e., equal prior probabilities of labels '1' and '0'. Then $h^*(x)$ is the posterior probability that the example x came from P, or, using our defined labels, $h^*(x) = \Pr[1|x]$. Thus, minimizing a log loss, i.e. the binary cross-entropy, and using its output probability for class 1 as witness function also maximizes test power.

Notice that for $c \neq 1/2$, we need to weight our samples with the inverse weights, i.e., it is more important to get the less frequent samples right.

Proposition 4.3.2 and Remark 4.3.1 lead to the main conclusion of this chapter: *To find an optimal witness, we can simply optimize the (weighted) squared error or a cross-entropy loss.* This allows us to seamlessly integrate existing AutoML frameworks, which are designed to solve this task in an automated fashion, to learn powerful witnesses. In the following we mainly focus on the squared error.

4.3.2. Practical implementation

Stage I - optimization. In the first stage, we optimize the witness function to minimize the MSE via the training data $\mathbb{X}_{\text{tr}}$ and $\mathbb{Y}_{\text{tr}}$, as motivated in the previous section. We simply label the data with '1' or '0' depending on whether they come from P or Q. We can then use any library that implements an optimization of a squared loss. If $c \neq 1/2$ we additionally need to specify weights according to Equation 4.6. Note that, unsurprisingly, the relevant quantity for the test power is the loss on the test data and not on the training data. Thus, it is of crucial importance to find a witness with good generalization performance. To make this as simple as possible for practitioners, we propose to use an AutoML framework. This also has the advantage that users can specify runtime and memory limits, and can explicitly trade computational resources for better statistical significance.

Although we strongly argue towards using AutoML for the test, this can of course not circumvent the no-free-lunch theorem. Thus, whenever users have good intuition about how their two samples might differ, we strongly encourage taking this into account when designing the test. To put it to the extreme: If one knows that their (one-dimensional) data follows a normal distribution and only differs in mean (if at all), one should use a classic t-test rather than our approach.

Stage II - testing. Given a witness function h learned as detailed in the previous section, we compute the test statistic as in Equation 4.2. To compute a p-value or decide whether to reject the null hypothesis $P = Q$, we can either approximate the asymptotic distribution or use permutations (Chapter 3). To estimate an asymptotically valid p-value[1] we first estimate $\sigma_c^2(h)$ (see Equation 4.3) based on $\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}$, which we denote as $\hat{\sigma}_c^2(h)$. The p-value is then given as $1 - \Phi\left(\sqrt{n_{\text{te}} + m_{\text{te}}}\tau(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}|h)/\hat{\sigma}_c(h)\right)$.

1: Asymptotic p-values are strictly speaking only valid for fixed h as the size of $\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}$ goes to infinity.

For two-sample tests, a cheap alternative that guarantees correct Type-I error control even at finite sample size is based on permutations as we described in Section 1.2. In case of witness functions, one can simply permute the values $h(x_1), \ldots, h(x_{n_{\text{te}}}), h(y_1), \ldots, h(y_{m_{\text{te}}})$ and split them in two sets of size n_{te} and m_{te}, respectively. One then recomputes the test statistic and estimates the p-value based on the empirical quantile over $B \in \mathbb{N}$ iterations. We reject whenever $p \leq \alpha$. We emphasize that we do not need to retrain the model, and it even suffices to evaluate the witness once on all elements of the test sets. We can then directly permute the witness' values.

Runtime. The overall runtime of the AutoML based witness test is the sum of the runtimes of the training phase, the evaluation of the witness, and the evaluation of the test statistic. We denote the scaling of the former by $s_{\text{train}}[n_{\text{tr}} + m_{\text{tr}}]$, where the square brackets emphasize that this is not a product. It will depend on the AutoML framework but can usually be controlled by setting a time limit. Even with a limit of one minute or less AutoGluon can already train powerful models on large datasets and even performs model-selection, hyperparameter optimization, and so on. In contrast, deep kernel-based methods typically train a neural network with a fixed architecture, which can be expensive. Although

neural networks belong to the suite of models AutoGluon trains, they are optimized for speed and if the runtime limit does not permit training them another faster model will be selected.

The scaling of evaluating h, denoted by $s_{\text{eval}}[n_{\text{tr}} + m_{\text{tr}}]$, is usually linear in the dataset size, but it can be sublinear if the evaluation is parallelized. It can also be controlled with AutoGluon by using different hyperparameter presets which might optimize the model selection towards fast inference times. Compared to that, kernel-based tests have a quadratic runtime. Furthermore, the test statistic has to be evaluated on the original partition of the data as well as B permutations requiring $(n_{\text{tr}} + m_{\text{tr}})(B + 1)$ steps. In practice, this is usually the cheapest step, but it could also be further reduced by parallelization. The overall runtime is given by

$$O\left(s_{\text{train}}[n_{\text{tr}} + m_{\text{tr}}] + s_{\text{eval}}[n_{\text{te}} + m_{\text{te}}] + (n_{\text{te}} + m_{\text{te}})(B + 1)\right). \tag{4.8}$$

Generally, training the witness will be the most expensive step of our test. A main advantage of our test over others is that practitioners can easily trade-off spending more time and resources on the training phase to potentially get a better witness and thus to more significant results. Thanks to AutoML, specifying the time and resources does not require any detailed knowledge of the underlying algorithm and is hence easily done.

4.3.3. Interpretability

Suppose our test finds a significant difference between $\mathbb{X}$ and $\mathbb{Y}$. An additional task would be to *interpret* how the distributions differ. This is particularly simple in our framework and shown in Figure 4.2: We can check which examples attained the highest value of the witness to find which inputs are much more likely under P than under Q. On the other hand, inputs with small witness values are more likely under Q. Similar procedures were used in [9, 18, 37]. An additional advantage of using the AutoML framework AutoGluon is that it allows to compute feature importance values easily. Therefore, for datasets which are hard to visualize the important features of data points with high or low witness values can be identified.

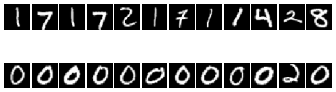


Figure 4.2.: Testing MNIST against shifted MNIST with '0's *knocked out*. The optimized witness assigns the highest values to the images on the left, and lowest values to the images on the right, allowing us to interpret the difference.

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*; [37]: Jitkrittum et al. (2016), *Interpretable Distribution Features with Maximum Testing Power*

4.4. Related work

We already discussed most related work in previous chapters. In Appendix C.1.2 we show that our results in Subsection 4.3.1 similarly apply to learning kernels [7, 8]. Concretely, instead of using the signal-to-noise ration Equation 1.17 one can also use a squared loss or cross-entropy loss when optimizing the kernel and the asymptotically optimal kernel is given as

$$k^*(x, x') = h^*(x)h^*(x'), \tag{4.9}$$

with h^* given in Equation 4.7.

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

We shortly also mention two newer works. [65] test with a finite collection of different kernels and reject if one of these MMD-based tests rejects. To

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*

ensure correct Type-I error control, they need to *aggregate* the test and modify the test thresholds to account for the multiple testing (MMDAgg). This allows them to use the full dataset, without having to split in train and test sets, but in turn this only enables using a countable candidate set. Recently, [93] proposed a general framework that also includes MMD.

[93]: Zhao et al. (2022), *Comparing Distributions by Measuring Differences that Affect Decision Making*

We now discuss the relation to classifier two-sample tests (C2ST) in more detail. They also rely on a data splitting approach and have extensively been studied in the literature [9, 33, 48, 49, 81, 85]. For simplicity, we focus on the balanced case. A C2ST trains a classifier with $\mathbb{X}_{\text{tr}}$, labelled with '1' and $\mathbb{Y}_{\text{tr}}$ labelled with '0' and then estimates its classification accuracy on $\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}}$. If the estimated accuracy is significantly above chance (that's what it would be under the null hypothesis), the test rejects. Let $f : \mathcal{X} \to \{0, 1\}$ denote the binary classifier, then we can write the accuracy as $\frac{1}{2} + \varepsilon$ and estimate it as

[9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [33]: Kim et al. (2021), *Classification accuracy as a proxy for two-sample testing*; [48]: Friedman (2003), *On multivariate goodness of fit and two sample testing*; [49]: Cai et al. (2020), *Two-sample test based on classification probability*; [81]: Golland et al. (2003), *Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies*; [85]: Hediger et al. (2022), *On the use of random forest for two-sample testing*

$$\begin{aligned}\frac{1}{2} + \hat{\varepsilon} &= \frac{1}{2}\left(\frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}} f(x_i) + \frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}}(1 - f(y_i))\right) \\ &= \frac{1}{2} + \frac{1}{2}\left(\frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}} f(x_i) - \frac{1}{n_{\text{te}}}\sum_{i=1}^{n_{\text{te}}} f(y_i)\right) \\ &= \frac{1}{2} + \frac{1}{2}\tau(\mathbb{X}_{\text{te}}, \mathbb{Y}_{\text{te}} \mid f).\end{aligned}$$

Thus using the classification accuracy as test statistic is equivalent to using the mean discrepancy as test statistic with the binary classifier f as witness function in Equation 4.2. However, using binary classifiers is quite limiting and results in quite high variance. Using continuous witness functions allows for higher power.[2] Some might also speak of 'classifier' test when referring to a witness test, but using the term 'witness' emphasizes that it is continuous.

2: [9] also observe that using a binary classifier might be too restrictive (see their Remark 2), but they did not investigate this in detail.

[83] proposed to use a pretrained classifier to detect label shift. [18] extended this to detect covariate shift. They investigate different ways of reducing the dimensionality and then applying different (classical) hypothesis test on them. While they also consider a basic C2ST, their best performing method uses the softmax outputs of a pretrained image classifier. They then run a *univariate* Kolmogorov-Smirnov test on each of the output 'probabilities' separately and correcting via Bonferroni correction. We refer to this as (univariate) BBSDs (black box shift detection - soft). For more details on their other methods, we refer the reader to their work directly.

[83]: Lipton et al. (2018), *Detecting and Correcting for Label Shift with Black Box Predictors*

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

4.5. Experiments

To show the power of utilizing AutoML we use the same setup for all datasets we consider. The data is split into two equally sized parts since this is the standard approach [8, 9, 18]. We label data from P with '1', data from Q with '0' and fit a least square regression with AutoGluon's `TabularPredictor` [90]. We use the configuration `presets='best_quality'` and by default optimize with a five-minute time limit. For more details, we refer to the AutoGluon documentation. We run all experiments with significance level $\alpha = 5\%$. Results of correct Type-I

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*; [9]: Lopez-Paz et al. (2017), *Revisiting Classifier Two-Sample Tests*; [18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

[90]: Erickson et al. (2020), *Autogluon-tabular: Robust and accurate automl for structured data*

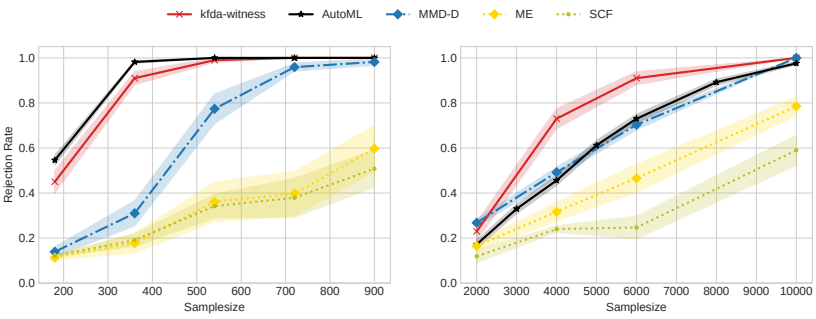


Figure 4.3.: Experiments on low dimensional problems. The simple approach of learning a one-dimensional witness function with AutoML or optimizing a witness via kfda and a grid search can outperform more involved approaches. **Left:** Blob, **Right:** Higgs.

Table 4.1.: Shift detection on MNIST and CIFAR10 based on [18].

(a) Test power across all simulated shifts on MNIST and CIFAR10. We propose the AutoML methods, and additionally run new baselines (MMDAgg, MMD-D).

Test	DR	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
Univ. tests	NoRed	0.03	0.15	0.26	0.36	0.41	0.47	0.54	0.72
	PCA	0.11	0.15	0.30	0.36	0.41	0.46	0.54	0.63
	SRP	0.15	0.15	0.23	0.27	0.34	0.42	0.55	0.68
	UAE	0.12	0.16	0.27	0.33	0.41	0.49	0.56	0.77
	TAE	0.18	0.23	0.31	0.38	0.43	0.47	0.55	0.69
	BBSDs	0.19	0.28	**0.47**	**0.47**	0.51	**0.65**	**0.70**	0.79
χ^2	BBSDh	0.03	0.07	0.12	0.22	0.22	0.40	0.46	0.57
Bin	Classif	0.01	0.03	0.11	0.21	0.28	0.42	0.51	0.67
Multiv. tests	NoRed	0.14	0.15	0.22	0.28	0.32	0.44	0.55	–
	PCA	0.15	0.18	0.33	0.38	0.40	0.46	0.55	–
	SRP	0.12	0.18	0.23	0.31	0.31	0.44	0.54	–
	UAE	0.20	0.27	0.40	0.43	0.45	0.53	0.61	–
	TAE	0.18	0.26	0.37	0.38	0.45	0.52	0.59	–
	BBSDs	0.16	0.20	0.25	0.35	0.35	0.47	0.50	–
AutoML (raw)		0.17	0.24	0.37	0.46	0.50	0.62	0.67	**0.87**
AutoML (pre)		0.18	**0.29**	0.42	**0.47**	0.47	0.64	0.65	0.72
AutoML (class)		0.19	0.19	0.38	0.46	**0.52**	0.61	0.67	**0.87**
AutoML (bin)		0.03	0.14	0.31	0.43	0.49	0.51	0.59	0.86
MMDAgg		0.19	0.24	0.31	0.32	0.40	–	–	–
MMD-D		**0.22**	0.19	0.25	0.36	0.40	0.48	0.56	0.65

(b) Test power depending on the shift for the AutoML test on the raw features (raw) vs. the AutoML test on the output of pretrained features (pre).

Shift	Test	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
s_gn	**raw**	0.20	0.27	0.33	0.40	0.43	0.50	0.63	0.80
	pre	0.00	0.03	0.10	0.03	0.00	0.10	0.03	0.03
m_gn	**raw**	0.27	0.23	0.33	0.43	0.43	0.53	0.63	0.83
	pre	0.00	0.03	0.17	0.00	0.00	0.13	0.07	0.13
l_gn	**raw**	0.23	0.33	0.53	0.67	0.70	0.77	1.00	1.00
	pre	0.17	0.27	0.50	0.57	0.60	0.73	0.80	0.90
s_img	raw	0.13	0.27	0.30	0.33	0.40	0.50	0.53	0.83
	pre	0.20	0.30	0.60	0.57	0.67	0.83	0.83	1.00
m_img	raw	0.03	0.00	0.03	0.00	0.10	0.20	0.30	0.57
	pre	0.07	0.03	0.13	0.10	0.13	0.33	0.47	0.60
l_img	raw	0.20	0.07	0.27	0.37	0.40	0.50	0.47	0.83
	pre	0.10	0.03	0.07	0.23	0.27	0.57	0.63	0.70
adv	raw	0.07	0.10	0.37	0.37	0.43	0.70	0.67	0.90
	pre	0.27	0.33	0.53	0.67	0.60	0.83	0.80	0.87
ko	raw	0.17	0.33	0.37	0.50	0.60	0.83	0.83	0.97
	pre	0.27	0.47	0.57	0.77	0.67	0.87	0.87	0.97
m_img	raw	0.00	0.03	0.23	0.53	0.53	0.67	0.67	1.00
+ko	**pre**	0.17	0.43	0.50	0.73	0.80	1.00	1.00	1.00
oz	raw	0.37	0.77	0.97	1.00	1.00	1.00	1.00	1.00
+m_img	**pre**	0.60	0.93	1.00	1.00	1.00	1.00	1.00	1.00

error control are provided in Appendix C.2. The sample size we report is always the size of the datasets before splitting, i.e, $n = m$, since we only consider balanced problems.

All experiments in this chapter were done on servers having only CPUs and we spend around 100k CPU hours on doing all the experiments reported in this chapter, which is mainly because we did various configurations and many repetitions for all the test cases we consider. Further details are given in Appendix C.2.

Blob & Higgs. We first compare the performance on the two low-dimensional datasets Blob and Higgs that we already used for the benchmark experiments in Section 3.6. As baselines, we use MMD-D, ME, SCF, and kfda-witness as reported in Section 3.6. We report the results in Figure 4.3, where ±1 standard error are shown as shaded regions. Since we estimated the performance over 500 runs, we obtain a smaller error than the other methods. We observe that both approaches based on the mean difference of a witness function (kfda-witness, AutoML) perform competitively. AutoML performs best on Blob, and kfda-witness is best on Higgs.

Detecting distribution shift. [18] introduced a large benchmark for the detection of distribution shifts. We repeat their experiments by considering the datasets MNIST [3] and CIFAR10 [94]. We consider sample sizes $n, m \in \{10, 20, 50, 100, 200, 500, 1000, 10000\}$. Each shift is applied on a fraction $\delta \in \{0.1, 0.5, 1.0\}$ of the second sample in different runs. We consider the following shifts: **Adversarial (*adv*)**: Turn some images into adversarial examples via FGSM [95]; **Knock-out (*ko*)**: Remove samples from class 0; **Gaussian noise (*gn*)**: Add gaussian noise to images with standard deviation $\sigma \in \{1, 10, 100\}$ (denoted *s_gn*, *m_gn*, and *l_gn*); **Image (*img*)**: Natural shifts to images through combinations of random rotations, (x, y)-axis-translation, as well as zoom-in with different strength (denoted *s_img*, *m_img*, and *l_img*); **Image + knock-out (*m_img+ko*)**: Fixed medium image shift and a variable knock-out shift; **Only-zero + image (*oz+m_img*)**: Only images from class 0 in combination with a variable medium image shift. More details are given in [18]. In total, we run 33 different shift experiments on MNIST and CIFAR10 each and for each sample size. Every setting is repeated for 5 times.

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

[3]: LeCun et al. (2010), *MNIST handwritten digit database*

[94]: Krizhevsky (2009), *Learning Multiple Layers of Features from Tiny Images*

[95]: Goodfellow et al. (2015), *Explaining and Harnessing Adversarial Examples*

The methods of [18] perform a dimensionality reduction by using the whole training set (50.000 images for MNIST, 40.000 images for

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

10). The actual tests compare examples from the validation set (10.000 images) to examples from the shifted test set (10.000 images). They also consider a C2ST trained on the raw features, i.e. without seeing the whole training set.

We add four univariate AutoML witness tests: a) **AutoML (raw)** trains a regression model on the raw data with MSE, which is our default, b) **AutoML (pre)** uses the same setting, but trains on the softmax output of a pretrained classifier for MNIST/CIFAR10 respectively, which is the same representation as BBSDs used, c) **AutoML (class)** trains a classifier and uses its predicted probabilities of class '1' as witness function, d) **AutoML (bin)** uses the same as c) but only considers binary outputs.

As additional baselines we also, for the first time, run the shift detection pipeline with MMD-D [8] and MMDAgg [65], where we use the settings recommend in their paper. For MMD-D we use the exact architectures and hyperparameters that [8] used for their MNIST and CIFAR10 Tasks. For MMDAgg, we use Gaussian kernels with bandwidth in $\{2^c \lambda_{\text{med}} \mid c \in \{10, 11, \ldots, 19, 20\}$, as recommended for MNIST [65, p.26]. With the implementation of MMDAgg, we received a memory error for sample size larger than 200, so results are only reported up to 200.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*

Our findings are reported in Table 4.1. From Table 4.1a we see that AutoML (raw) achieves overall very competitive performance in detecting the shifts, especially for large sample sizes. Moreover, we see that AutoML (raw) and AutoML (class) achieve comparable performance which confirms our findings of Remark 4.3.1. Thresholding the classification probabilities to binary outputs always harms the performance, see AutoML (class) vs. AutoML (bin). We can also compare AutoML (bin) with 'classif', as reported by [18]. While both use binary classifiers for the testing, 'classif' used a fixed architecture across all shifts. This illustrates the power of using AutoML, as we find significantly better performance across all sample sizes. If instead of training on the raw features we start from the ten dimensional pretrained features, i.e. AutoML (pre), the

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

performance is improved when the sample size is small. For large sample sizes, instead working with the raw features gives higher power.

We also see that the AutoML test outperforms MMDAgg and MMD-D except for very small sample size.

In Table 4.1b we report the test power for comparing AutoML (raw) with AutoML (pre) for the different shifts. Using the pretrained probabilities of the softmax output, it is extremely hard to detect Gaussian noise, while AutoML (raw) does a fairly good job here. This is consistent with the findings of [18, Table 1(b)]. Apparently, the output probabilities of the pretrained models are quite invariant under small and medium noise on the inputs. For the other shifts, such as knock-outs, using the pretrained features improves performance, particularly at small sample sizes.

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

The code to reproduce our experiments of this chapter is provided at: github.com/jmkuebler/autoML-TST-paper.

4.6. Discussion

Bias-variance tradeoff. Our results on the distribution shift benchmark indicate a bias-variance tradeoff when optimizing the witness in stage-I. Learning the witness function over a ten dimensional pretrained representation gives good test power for some shifts even for small sample sizes, however, at the cost of being almost unable to detect other shifts, such as local Gaussian noise. Thus, learning on pretrained features introduces a strong bias. On the other hand, learning directly on the raw features introduces little bias, even more so since we used AutoGluon's `TabularPredictor`, which is not specifically designed for images. This has the effect that on small sample sizes the test power is reduced, but when large data is available, we observe good test power across almost all shifts. For practical applications this implies that using models with the right bias when learning hypothesis tests is just as important as in any other supervised learning setting.

Stand on the shoulders of giants. As we see from the Blob and Higgs experiments the conceptually simple witness two-sample test can outperform more sophisticated test statistics like the deep MMD. This is possible through both the use of cross-validation (kfda-witness) or a full AutoML pipeline. In the distribution shift benchmark, we saw much better performance even when comparing a binary classifier (AutoML (bin)) with a classifier having a prespecified architecture (classif). Furthermore, using an AutoML framework allows practitioners to stand on the shoulders of giants and removes the need for specialized expertise. Instead, they can directly control how much time and resources to spend on optimizing the witness, which can lead to improved significance and/or inference time.

Which test to use? Obviously, there is no general answer to this question, and we are not claiming that our AutoML two-sample test should always be used. In special settings, a simple parametric test would perform much better than our AutoML witness test. Similarly, using MMD with a

kernel can be the right choice in some settings. Nevertheless, a few points should be considered. For example, we demonstrate that a test using binary outputs of a classifier underperforms a test using the predicted probabilities of the same classifier. Therefore, we do recommend choosing the latter instead of the former. Furthermore, when using data splitting we should ensure that in the first stage we are actually optimizing the test power or a directly related proxy loss. To this end, it is important to use techniques that ensure good predictive performance and prevent overfitting. This brings us to the last point: We should also consider the resources available, both computational and human, that are relevant when implementing the test. That is, a testing framework should be easy to apply by a large group of users and should be adaptable to the computational resources the user is willed to spend on the test. The AutoML witness test can tick off all boxes. It learns a continuous witness function to optimize test power, leverages well-engineered toolboxes to maximize predictive performance, and requires little engineering expertise to apply and gives easy control over the computational resources used to learn the test, by setting a time limit and providing the available hardware.

4.7. Chapter conclusion

We showed that optimizing a squared loss or cross-entropy loss leads to a witness function that maximizes test power, when using the mean discrepancy of the witness as a test statistic. This allows us to harness the advances in Automated Machine Learning, where regression and classification are the standard tasks, for two-sample testing. Although less studied, the use of a well-engineered toolbox to maximize the predictive performance of the learned function is just as important for hypothesis testing as it is for supervised learning tasks. The result is a testing pipeline that is theoretically justified, leads to competitive performance, and is simple to apply in various settings. Our work thus constitutes a step towards fully automated statistical analysis of complex data [96].

[96]: Steinruecken et al. (2019), *The automatic statistician*

Can quantum computers speed-up two sample tests?

Quantum mean embedding of probability distributions

5.

In Equation 1.5 we introduced the kernel mean embedding of probability distributions. As we discussed it is used in machine learning as an injective mapping from distributions to functions in an infinite dimensional Hilbert space. It allows us, for example, to define a distance measure between probability distributions, called maximum mean discrepancy (MMD). In this chapter we propose to represent probability distributions in a pure quantum state of a system that is described by an infinite dimensional Hilbert space and prove that the representation is unique if the corresponding kernel function is c_0-universal. This enables us to work with an explicit representation of the mean embedding, whereas classically one can only work implicitly with an infinite dimensional Hilbert space through the use of the kernel trick. We show how this explicit representation can speed up methods that rely on inner products of mean embeddings and discuss the theoretical and experimental challenges that need to be solved in order to achieve these speedups.

5.1. Introduction

In machine learning, kernel methods are used to implicitly evaluate inner products in high dimensional feature spaces. Popular linear algorithms such as the support vector machine [97, 98] or principal component analysis [99] can be expressed solely in terms of inner products between data points. These methods become more expressive if the data is first mapped onto a high dimensional feature space. Instead of evaluating the inner product explicitly in the feature space, whose cost scales linearly with the feature space dimension, a more efficient evaluation can be done implicitly in the original space using a positive definite kernel function. This is known as the *kernel trick* [4]. Since it does not require an explicit feature map, the kernel trick even allows us to work with infinite dimensional feature spaces, e.g., using a Gaussian kernel. The downside of most kernel-based methods is that they scale polynomially with the size of the data sets. This problem has been tackled in the realm of quantum computation and exponential speedups have been conjectured [100, 101]. However, such speedups are still highly controversial [12, 102, 103].

Only recently has the cost of a single kernel evaluation been the target of quantum computing research [104–106]. Speedups might be possible, since the cost of explicitly evaluating inner products of quantum states only grows logarithmically with the system size [107], as opposed to linear on a classical computer. Schuld and Killoran further conjecture the usage of continuous variable quantum systems for working with classically intractable, i.e., hard to compute, kernels in infinite dimensions [105], but it is unclear whether problems exist for which such kernels can lead to an improvement [12]. Furthermore, the recent suggestions do

[97]: Cortes et al. (1995), *Support-Vector Networks*; [98]: Steinwart et al. (2008), *Support Vector Machines*

[99]: Hotelling (1933), *Analysis of a complex of statistical variables into principal components.*

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

[100]: Rebentrost et al. (2014), *Quantum Support Vector Machine for Big Data Classification*; [101]: Lloyd et al. (2014), *Quantum principal component analysis*

[12]: Kübler* et al. (2021), *The inductive bias of quantum kernels*; [102]: Aaronson (2015), *Read the fine print*; [103]: Ciliberto et al. (2018), *Quantum machine learning: A classical perspective*

[104]: Chatterjee et al. (2017), *Generalized Coherent States, Reproducing Kernels, and Quantum Support Vector Machines*; [105]: Schuld et al. (2019), *Quantum Machine Learning in Feature Hilbert Spaces*; [106]: Havlicek et al. (2019), *Supervised learning with quantum-enhanced feature spaces*

[107]: Cincio et al. (2018), *Learning the quantum algorithm for state overlap*

[105]: Schuld et al. (2019), *Quantum Machine Learning in Feature Hilbert Spaces*

not address the polynomial scaling of kernel methods with the sample size, leaving the application of quantum computing in large-scale kernel methods a challenging problem.

The idea of explicitly representing an infinite dimensional feature vector as a quantum state opens a way to tackle this problem. While it is impossible classically to sum two infinite dimensional vectors, a quantum mechanical *superposition* of two states can be constructed explicitly, even for infinite dimensional systems, see, e.g., [108]. On the other hand, *the evaluation of inner products in an infinite dimensional quantum Hilbert space is independent of the number of states in a superposition*. We identify methods involving the *kernel mean embedding* [19, 28, 109] as a branch of machine learning techniques that suffer from the fact that on a classical computer the cost of the evaluation of inner products of sums of feature maps is not independent of the number of data points involved.

[108]: Vlastakis et al. (2013), *Deterministically Encoding Quantum Information Using 100-Photon Schrödinger Cat States*

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*; [28]: Smola et al. (2007), *A Hilbert space embedding for distributions*; [109]: Berlinet et al. (2004), *Reproducing Kernel Hilbert Spaces in Probability and Statistics*

The contribution of this chapter is to adapt the notion of kernel mean embedding to quantum mechanics, point out how quantum mechanics can lead to speedups, and make transparent what the challenges are in order to realize this in an experiment. The chapter is organized as follows. We start by introducing the kernel mean embedding from a classical perspective, point out the main problem it has in big data applications, and present its relevance in current machine learning research through some real-world applications. We then define the *quantum mean embedding* as a modified version of the kernel mean embedding, which makes it suitable for investigation in the context of quantum computation, and show that this modification still allows for the usage in conventional applications. We present how the quantum mean embedding can be used, in principle, to overcome the problems faced classically. Since this cannot be done on nowadays hardware, we continue with a section on the challenges left. Finally, we sum up with a discussion of our results.

5.2. Kernel mean embedding

Let $\mathcal{X}$ be a locally compact and Hausdorff space. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{C}$ is called a positive definite kernel function, or kernel function for brevity, if for all $n \in \mathbb{N}$, $x_1, ..., x_n \in \mathcal{X}$, and $c_1, ..., c_n \in \mathbb{C}$, it holds that $\sum_{i,j=1}^{n} c_i^* c_j k(x_i, x_j) \geq 0$ [4]. For every kernel function there exists a unique reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ such that $k(\cdot, x) \in \mathcal{H}_k$ for all $x \in \mathcal{X}$ and the *reproducing property* $f(x) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}_k}$ holds for all $f \in \mathcal{H}_k$ and $x \in \mathcal{X}$. We call the mapping $\phi : \mathcal{X} \to \mathcal{H}_k$ given by $\phi(x) := k(\cdot, x)$ the *canonical* feature map of k, i.e., $k(x, y) = \langle \phi(y), \phi(x)\rangle$ [110].

So far we constraint ourselves to real-valued kernels. Since complex kernels arise quite naturally in quantum mechanics, we will introduce kernels more generally here.

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

[110]: Aronszajn (1950), *Theory of Reproducing Kernels*

Let P be a probability measure over $\mathcal{X}$. The kernel mean embedding (KME) of P is defined as [28, 109]

$$\mu_P := \int_{\mathcal{X}} k(\cdot, x) \mathrm{d}P(x) = \int_{\mathcal{X}} \phi(x) \mathrm{d}P(x). \tag{5.1}$$

[28]: Smola et al. (2007), *A Hilbert space embedding for distributions*; [109]: Berlinet et al. (2004), *Reproducing Kernel Hilbert Spaces in Probability and Statistics*

The embedding μ_P exists and is a function in $\mathcal{H}_k$ if $\mathbb{E}_{X\sim P}[k(X, X)] < \infty$ [28]. In practice we do not have access to the true probability distribution P. Instead, we observe a finite i.i.d. sample $\mathbb{X} = \{x_1, ..., x_n\}$ drawn from

[28]: Smola et al. (2007), *A Hilbert space embedding for distributions*

P. Based on the sample $\mathbb{X}$, an empirical estimate of μ_P is given by the KME of the empirical distribution $\hat{P} = \frac{1}{n}\sum_{i=1}^{n}\delta_{x_i}$:

$$\mu_{\mathbb{X}} := \frac{1}{n}\sum_{i=1}^{n}\phi(x_i), \tag{5.2}$$

which converges to the true embedding of P in the Hilbert space metric at a rate of $n^{-\frac{1}{2}}$ [19].

The kernel function k is said to be *characteristic* if the map $\mu : P \mapsto \mu_P$ is injective [31, 45]. In other words, working with a characteristic kernel enables us to represent (all properties of) a probability distribution by a function in the RKHS, which is why the notion of characteristic kernels plays an important role in kernel methods [111]. The notion of characteristic kernels is closely related to the notion of universal kernels [112]. Here we call a kernel *universal* if the corresponding RKHS is dense in the space of continuous functions over $\mathcal{X}$ that vanish at infinity, which corresponds to c_0-universality [111]. For c_0-universal kernels the KME is injective even for finite signed measures [111]. Popular universal kernels include the Gaussian kernel $k(x,y) = \exp(-\|x-y\|^2/2\sigma^2)$ and Laplacian kernel $k(x,y) = \exp(-\|x-y\|_1/\sigma)$, where σ is a bandwidth parameter [45, 113].

The expressiveness of characteristic kernels comes at a price. Since there exist distributions with infinite moments, the corresponding RKHS must have infinite dimensions to ensure no information loss. Consequently, it is impossible for a classical computer to represent and manipulate $\mu_{\mathbb{X}}$ directly. However, if we only care about inner products of mean embeddings, which is usually the case in most algorithms, we can resort to the "kernel trick" and replace inner products with kernel evaluations [4]. That is, given i.i.d. samples $\mathbb{X} = \{x_1, \ldots, x_n\}$ from P and $\mathbb{Y} = \{y_1, \ldots, y_n\}$ from Q [1], we can evaluate

$$\begin{aligned}\langle \mu_{\mathbb{X}}, \mu_{\mathbb{Y}} \rangle &= \frac{1}{n^2}\sum_{i,j=1}^{n}\langle \phi(x_i), \phi(y_j)\rangle = \frac{1}{n^2}\sum_{i,j=1}^{n}k(x_i,y_j)\\ &=: K(\mathbb{X},\mathbb{Y}).\end{aligned} \tag{5.3}$$

The inevitable drawback of this trick is that algorithms based on $K(\mathbb{X},\mathbb{Y})$ have a runtime complexity that scales at least quadratically with the number of data points n.

In the following we present essential applications of the KME, which suffer from the above limitation.

1. *Learning on probability distributions:* Classical machine learning algorithms were originally developed for training data consisting of *points* in some vector space. In several domains such as astronomy and high-energy physics, however, data are represented naturally as probability distributions, e.g., clusters of galaxies and groups of collision events. The KME (5.1) allows us to generalize algorithms to the space of probability distributions [114–117] through the *distributional* kernel function

$$K(P,Q) = \langle \mu_P, \mu_Q \rangle_{\mathcal{H}_k} = \iint_{\mathcal{X}} k(x,y)\,\mathrm{d}P(x)\,\mathrm{d}Q(y). \tag{5.4}$$

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

[31]: Fukumizu et al. (2008), *Kernel Measures of Conditional Dependence*; [45]: Sriperumbudur et al. (2010), *Hilbert Space Embeddings and Metrics on Probability Measures*

[111]: Simon-Gabriel et al. (2018), *Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions*

[112]: Steinwart (2001), *On the Influence of the Kernel on the Consistency of Support Vector Machines*

[111]: Simon-Gabriel et al. (2018), *Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions*

[45]: Sriperumbudur et al. (2010), *Hilbert Space Embeddings and Metrics on Probability Measures*; [113]: Fukumizu et al. (2004), *Dimensionality Reduction for Supervised Learning with Reproducing Kernel Hilbert Spaces*

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

1: For simplicity we assume the sample sizes to be equal.

[114]: Muandet et al. (2012), *Learning from Distributions via Support Measure Machines*; [115]: Muandet et al. (2013), *One-class Support Measure Machines for Group Anomaly Detection*; [116]: Lopez-Paz et al. (2015), *Towards a Learning Theory of Cause-Effect Inference*; [117]: Szabó et al. (2016), *Learning Theory for Distribution Regression*

Given i.i.d samples $\mathbb{X} = \{x_1, \ldots, x_n\}$ from P and $\mathbb{Y} = \{y_1, \ldots, y_n\}$ from Q, $K(P, Q)$ can be approximated by

$$K(P, Q) \approx \frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, y_j) = K(\mathbb{X}, \mathbb{Y}). \tag{5.5}$$

The main drawback of (5.5) is that given samples $\mathbb{X}_1, \ldots, \mathbb{X}_N$ from N input distributions, each of size n, the runtime complexity of evaluating the kernels $K(\mathbb{X}_i, \mathbb{X}_j)$ for all $i, j = 1, \ldots, N$ is $O(N^2 n^2)$. This is prohibitive for many real-world applications of learning problems on probability distributions.

2. *Maximum mean discrepancy (MMD):* The MMD is a discrepancy measure between any two distributions P and Q [5, 27] that we extensively discussed in the previous chapters. Recall that is given by the distance of the corresponding mean embeddings of the distributions [5, Lemma 4] and can be expressed solely in terms of inner products of mean embeddings (assuming a real kernel):

[5]: Gretton et al. (2012), *A kernel two-sample test*; [27]: Borgwardt et al. (2006), *Integrating structured biological data by Kernel Maximum Mean Discrepancy*

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\begin{aligned} \mathrm{MMD}^2(P, Q \mid \mathscr{H}_k) &= \|\mu_P - \mu_Q\|^2 \\ &= \langle \mu_P, \mu_P \rangle - 2 \langle \mu_P, \mu_Q \rangle + \langle \mu_Q, \mu_Q \rangle . \end{aligned} \tag{5.6}$$

For characteristic kernels, $\mathrm{MMD}^2(P, Q \mid \mathscr{H}_k) = 0 \Leftrightarrow P = Q$ [5, Theorem 5]. Given i.i.d. samples $\mathbb{X} = \{x_1, ..., x_n\}$ drawn from P and $\mathbb{Y} = \{y_1, ..., y_n\}$ drawn from Q, it is possible to estimate the MMD by evaluating (5.6) with the embeddings $\mu_{\mathbb{X}}$ and $\mu_{\mathbb{Y}}$ [5, Eq. (5)]:

[5]: Gretton et al. (2012), *A kernel two-sample test*

[5]: Gretton et al. (2012), *A kernel two-sample test*

$$\begin{aligned} \mathrm{MMD}^2(\mathbb{X}, \mathbb{Y} \mid \mathscr{H}_k) &= \|\mu_{\mathbb{X}} - \mu_{\mathbb{Y}}\|^2 \\ &= K(\mathbb{X}, \mathbb{X}) - 2K(\mathbb{X}, \mathbb{Y}) + K(\mathbb{Y}, \mathbb{Y}), \end{aligned} \tag{5.7}$$

whose cost is determined by that of evaluating $K(\mathbb{X}, \mathbb{Y})$, $K(\mathbb{X}, \mathbb{Y})$, and $K(\mathbb{Y}, \mathbb{Y})$.

3. *Deep learning:* The applications of KMEs in deep learning have gained a lot of attention in the past few years. Notably, the MMD has been used as an objective function for training deep generative models [118–120]. For a deep generative model G_θ parametrized by a parameter vector θ, the idea is to learn θ by minimizing the $\mathrm{MMD}^2(P, Q_\theta \mid \mathscr{H}_k)$, where P is the data distribution and Q_θ is the distribution induced by the generative model G_θ. Again, the downside of the MMD in this area is its computational cost as we usually have to deal with huge amount of data [121].

[118]: Dziugaite et al. (2015), *Training Generative Neural Networks via Maximum Mean Discrepancy Optimization*; [119]: Li et al. (2015), *Generative moment matching networks*; [120]: Li et al. (2017), *MMD GAN: Towards deeper understanding of moment matching network*

[121]: LeCun et al. (2015), *Deep Learning*

All of the above applications require the estimation of terms like $K(\mathbb{X}, \mathbb{Y})$, which scale quadratically with the sample size n, and hence become prohibitive for large n. To enable large-scale learning with KMEs, a common approach is to approximate $\mu_{\mathbb{X}}$ by a finite dimensional representation, e.g., using random Fourier features [122] or the Nyström method [71], after which it can be manipulated directly in a classical computer without resorting to the kernel trick. For a d dimensional approximation, the cost drops to $O(n + d)$, which is linear in n. The downside is that the embedding defined in terms of this representation can no longer be injective, which is an essential requirement in most applications of the KME.

[122]: Rahimi et al. (2008), *Random Features for Large-Scale Kernel Machines*

[71]: Williams et al. (2000), *Using the Nyström Method to Speed Up Kernel Machines*

Recent work [105, 106] showed how one can in principle evaluate a d

[105]: Schuld et al. (2019), *Quantum Machine Learning in Feature Hilbert Spaces*; [106]: Havlicek et al. (2019), *Supervised learning with quantum-enhanced feature spaces*

dimensional approximation of the kernel function using only $O(\log d)$ qubits. Furthermore, [123] has investigated quantum kernels in the context of the MMD. [124] formulates quantum graphical models in terms of the kernel mean embedding and uses a density matrix as a mean map. On the contrary, we focus on the quadratic scaling when using an infinite dimensional feature map which has not been addressed before in the quantum community.

[123]: Coyle et al. (2020), *The Born supremacy: quantum advantage and training of an Ising Born machine*

[124]: Srinivasan et al. (2018), *Learning and Inference in Hilbert Space with Quantum Graphical Models*

5.3. Quantum mean embedding

Let $\mathcal{H}$ be the Hilbert space of a quantum system and $\varphi : \mathcal{X} \to \mathcal{H}, x \mapsto |\varphi(x)\rangle$ a quantum feature map that assigns a quantum state $|\varphi(x)\rangle$, i.e., a normalized function in $\mathcal{H}$, to each point in the input domain $x \in \mathcal{X}$.[2] This defines a kernel $k(x, x') = \langle\varphi(x)|\varphi(x')\rangle$ [105, 106] with the constraint $k(x, x) = 1$ for all $x \in \mathcal{X}$, due to the normalization of quantum states [125].

Let P be a probability distribution over the input domain. We define the *quantum mean embedding* (QME)

$$|\nu_P\rangle := \frac{1}{\mathcal{N}_P} \int_{\mathcal{X}} |\varphi(x)\rangle \, \mathrm{d}P(x), \tag{5.8}$$

where the normalization $\mathcal{N}_P$ ensures the physicality of the state and is given by the norm of the corresponding KME (5.1), i.e., $\mathcal{N}_P := \|\mu_P\|_{\mathcal{H}_k}$.

The QME exists for all probability distributions due to the constraint $k(x, x) = 1$. *A subtle difference between the KME and the QME are the spaces in which the embeddings live.* While the KME is a function in the RKHS $\mathcal{H}_k$ and uniquely defined by the kernel k, the QME depends on the quantum systems Hilbert space $\mathcal{H}$ and the choice of the feature map φ. Even though the embeddings live in different spaces, for any two probability distributions P and Q we have

$$\langle\mu_{\mathbb{P}}, \mu_Q\rangle_{\mathcal{H}_k} = \mathcal{N}_P \cdot \mathcal{N}_Q \, \langle\nu_P|\nu_Q\rangle_{\mathcal{H}} \, . \tag{5.9}$$

That is, their inner products have a fixed relation independent of $\mathcal{H}$. Hence, the important difference is that the QME maps every probability distribution on the unit sphere in a Hilbert space, whereas the KME does not enforce this, see Figure 5.1. In the following theorem we show that if the kernel is universal we do not lose information about a probability measure when using the QME.

Theorem 5.3.1 ***Injectivity of the QME***
Let $\varphi : \mathcal{X} \to \mathcal{H}, x \mapsto |\varphi(x)\rangle$ be a mapping such that $k(x, y) = \langle\varphi(x)|\varphi(y)\rangle$ is a universal kernel for the space of continuous functions over $\mathcal{X}$ that converge to zero at infinity $\mathcal{C}_0(\mathcal{X})$. Let $\mathcal{P}$ be the space of Borel probability measures over the measurable space $(\mathcal{X}, \mathcal{A})$, where $\mathcal{A}$ denotes the Borel sigma algebra. For a universal kernel k, the QME (5.8), is injective over $\mathcal{P}$, i.e., $|\nu_P\rangle = |\nu_Q\rangle \Leftrightarrow P = Q$ for any $P, Q \in \mathcal{P}$.

The proof is included in Appendix D.1. For a finite sample $\mathbb{X}$ we define

2: In order to emphasize that we deal with a quantum state, we shall abuse notation by denoting the image of a point x under the mapping φ as $|\varphi(x)\rangle$ instead of $\varphi(x)$. Mathematically $|\varphi(x)\rangle$ denotes the same function in $\mathcal{H}$ as $\varphi(x)$

[105]: Schuld et al. (2019), *Quantum Machine Learning in Feature Hilbert Spaces*; [106]: Havlicek et al. (2019), *Supervised learning with quantum-enhanced feature spaces*

[125]: Nielsen et al. (2010), *Quantum Computation and Quantum Information*

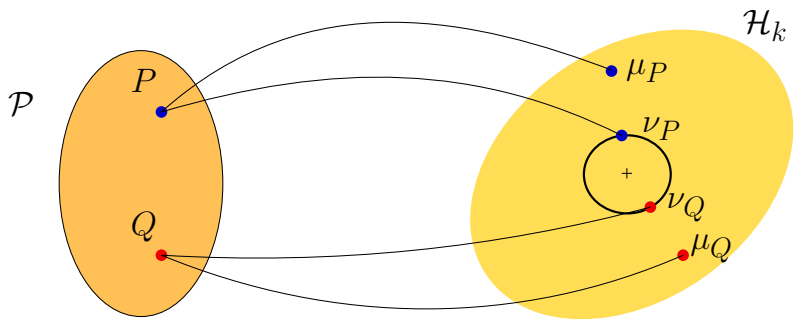


Figure 5.1.: Schematic comparison of the classical KME and the QME: The KME maps probability distributions P onto functions in the RKHS $\mathcal{H}_k$. The QME additionally enforces that the mapping is onto the unit ball (denoted by the circle) in the RKHS. Theorem 5.3.1 shows the injectivity of the QME for universal kernels. For visualization we choose $\mathcal{H} = \mathcal{H}_k$.

an empirical QME as

$$|\nu_{\mathbb{X}}\rangle := \frac{1}{\mathcal{N}_{\mathbb{X}}} \frac{1}{n} \sum_{i=1}^{n} |\varphi(x_i)\rangle \,, \tag{5.10}$$

with the normalization constant

$$\mathcal{N}_{\mathbb{X}} = \|\mu\mathbb{X}\|_{\mathcal{H}_k} = \sqrt{\frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, x_j)}. \tag{5.11}$$

As discussed before, for infinite dimensional feature maps, the KME cannot be described explicitly and only used via inner products. The advantage of the QME is that it is possible, in principle, to explicitly create $|\nu_{\mathbb{X}}\rangle$ in the lab, even for infinite dimensional cases. Here it is important that an experimenter only needs to create a state that is proportional to $\sum_{i=1}^{n} |\varphi(x_i)\rangle$. The prefactor (5.11) is enforced by the laws of physics and is not required for the state preparation. Given this explicit representation, it allows us to decouple the cost of the inner product evaluation from the sample size n, see Figure 5.2.

Conjecture 5.3.2 *Suppose we are given a routine that prepares states of the form Equation 5.10 with cost $O(n)$ for a feature map φ. In addition we are given a routine that can evaluate inner products of arbitrary states in $\mathcal{H}$ in constant time. Then for two samples $\mathbb{X} = \{x_1, \ldots, x_n\}$ and $\mathbb{Y} = \{y_1, \ldots, y_n\}$ one can evaluate $K(\mathbb{X}, \mathbb{Y})$, defined in (5.3), with cost $O(n)$, whereas a classical computer scales with $O(n^2)$.*

Proof. By assumption we can prepare $|\nu_{\mathbb{X}}\rangle$ and $|\nu_{\mathbb{Y}}\rangle$ with linear cost in n. Furthermore we can evaluate $\langle\nu_{\mathbb{X}}|\nu_{\mathbb{Y}}\rangle$ in constant time, given the individual states. Together the cost of evaluating the term $\langle\nu_{\mathbb{X}}|\nu_{\mathbb{Y}}\rangle$ scales at most with $O(n)$. The normalizations $\mathcal{N}_{\mathbb{X}}$ and $\mathcal{N}_{\mathbb{Y}}$ can also be estimated with cost $O(n)$, see Section 5.4. Using relation (5.9), we obtain

$$K(\mathbb{X}, \mathbb{Y}) = \langle\mu_{\mathbb{X}}, \mu_{\mathbb{Y}}\rangle_{\mathcal{H}_k} = \mathcal{N}_{\mathbb{X}}\mathcal{N}_{\mathbb{Y}} \langle\nu_{\mathbb{X}}|\nu_{\mathbb{Y}}\rangle_{\mathcal{H}} \,. \tag{5.12}$$

□

Compared to the classical KME, this conjecture implies that under the stated assumptions it is possible to simultaneously reduce the cost of the QME while preserving its expressibility guarantee given in Theorem 5.3.1.

Given an efficient evaluation of $K(\mathbb{X}, \mathbb{Y})$, it is possible to speed up the methods presented earlier, which rely on inner products of the KMEs. In the next section we discuss the assumptions of Conjecture 5.3.2. Apart from using the QME to speed up the evaluation of inner products of the KMEs, it follows from the proof of Theorem 5.3.1 that the QME is also important on its own, as it can uniquely represent probability distributions. However, it is unclear to what extend the applications of the KME could be rephrased solely in terms of inner products of the QME instead of taking the detour over $K(\mathbb{X}, \mathbb{Y})$, where we need to determine the normalizations.

5.4. Challenges

In order to harvest a potential quantum speedup it is necessary to create the QME efficiently, i.e., with resources and time linear in the sample size. We phrase this as the first challenge:

> *Given a quantum feature map φ, find an experimental strategy, denoted E_φ, such that for an arbitrary input sample $\mathbb{X} = \{x_1, \ldots, x_n\}$, with $n \in \mathbb{N}$, it creates $|\nu_\mathbb{X}\rangle$, using resources that scale at most linear in n.*

In case of coherent states as feature map (see Appendix D.2), superpositions similar to $|\nu_\mathbb{X}\rangle$ have already been experimentally realized for specific cases and are known as "cat-states" [108, 126, 127]. However, it is an open question how these approaches scale, even theoretically, for superposing a large number of states, see [128] for an overview on similar experimental approaches. In general, the rigorous study of resources required to construct superpositions of quantum states and the connections to entanglement are subject of current research [129, 130]. Particularly for the case of superpositions of nonorthogonal states, as it is the case for our proposed embedding, the theory becomes more involved [130, III.K.4]. Note that we explicitly allow for an experimental setup E_φ that is specific to the given quantum feature map φ, i.e., a specific kernel function. This is necessary because a universal machine that builds a superposition of completely arbitrary and unknown quantum states cannot exist [131, 132]. Furthermore, we emphasize that this work does not require a qRAM [133].

[108]: Vlastakis et al. (2013), *Deterministically Encoding Quantum Information Using 100-Photon Schrödinger Cat States*; [126]: Deléglise et al. (2008), *Reconstruction of non-classical cavity field states with snapshots of their decoherence*; [127]: Ourjoumtsev et al. (2007), *Generation of optical 'Schrödinger cats' from photon number states*

[128]: Andersen et al. (2015), *Hybrid discrete-and continuous-variable quantum information*

[129]: Theurer et al. (2017), *Resource Theory of Superposition*; [130]: Streltsov et al. (2017), *Colloquium: Quantum coherence as a resource*

[130]: Streltsov et al. (2017), *Colloquium: Quantum coherence as a resource*

[131]: Alvarez-Rodriguez et al. (2015), *The Forbidden Quantum Adder*; [132]: Oszmaniec et al. (2016), *Creating a Superposition of Unknown Quantum States*

[133]: Giovannetti et al. (2008), *Quantum Random Access Memory*

[134]: Buhrman et al. (2001), *Quantum Fingerprinting*

Given the QMEs, at the core of our approach lies the estimation of the inner product of two arbitrary quantum states in $\mathcal{H}$. Formally, this can be done by using the *swap test* routine of [134], see right side of Figure 5.2. The swap test works independently of the input states, which for our purpose we denote by $|\nu_\mathbb{X}\rangle, |\nu_\mathbb{Y}\rangle \in \mathcal{H}$. These inputs are each in one register and a single ancilla qubit in the state $|0\rangle$ in an additional register. The test itself consists of a Hadamard transformation H on the qubit, followed by a controlled swap of the two states conditioned on the state of the qubit, and another Hadamard transformation on the qubit. This circuit maps the initial state $|0\rangle |\nu_\mathbb{X}\rangle |\nu_\mathbb{Y}\rangle$ onto

$$\frac{|0\rangle (|\nu_\mathbb{Y}\rangle |\nu_\mathbb{X}\rangle + |\nu_\mathbb{X}\rangle |\nu_\mathbb{Y}\rangle) + |1\rangle (|\nu_\mathbb{Y}\rangle |\nu_\mathbb{X}\rangle - |\nu_\mathbb{X}\rangle |\nu_\mathbb{Y}\rangle)}{2},$$

see [134, Eq. (4)]. At the end, the qubit is measured in the computational

[134]: Buhrman et al. (2001), *Quantum Fingerprinting*

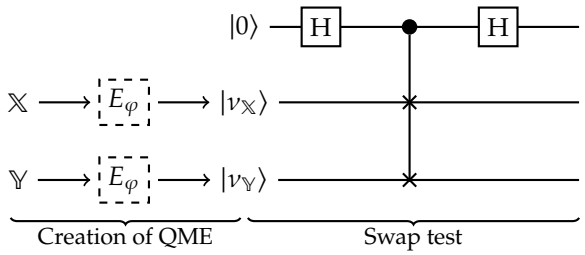


Figure 5.2.: The quantum approach separates the creation of the QME from the inner product estimation. It requires two subroutines. First, on the left, an experimental setup E_φ that creates the QME efficiently. Second, on the right, a circuit to estimate inner products of arbitrary states in $\mathcal{H}$ whose runtime is independent of the states. Here we chose the swap test, which uses an ancillary qubit. This approach detaches the estimation of the inner product from the sample size.

basis. This results in outcome 0 with probability $p_0 = (1 + |\langle \nu_\mathbb{X} | \nu_\mathbb{Y} \rangle|^2)/2$ and outcome 1 with probability $p_1 = 1 - p_0$. Repetitive application of this routine allows for an estimation of p_0 and p_1 from which one can infer $|\langle \nu_\mathbb{X} | \nu_\mathbb{Y} \rangle|^2 = 2p_0 - 1$. When using a Gaussian kernel, we know a priori that $\langle \nu_\mathbb{X} | \nu_\mathbb{Y} \rangle > 0$, thus $\langle \nu_\mathbb{X} | \nu_\mathbb{Y} \rangle = \sqrt{2p_0 - 1}$. If we cannot guarantee the positivity of $\langle \nu_\mathbb{X} | \nu_\mathbb{Y} \rangle$, we need a phase sensitive estimation of inner products, as discussed in the supplemental material of [105]. Crucially, the swap test works independently of the size of the samples $\mathbb{X}$ and $\mathbb{Y}$.

[105]: Schuld et al. (2019), *Quantum Machine Learning in Feature Hilbert Spaces*

For finite dimensional systems, [107] recently proposed an implementation that scales logarithmically with the dimension of the Hilbert space. But this approach does not translate to systems of infinite dimension. The infinite dimensional case has been studied in [135–137]. However, they do not give an explicit solution and we are not aware of any experimental realization of a universal swap test for the infinite dimensional case. This marks the second challenge arising from this chapter. At the stage of preparing superpositions in the form of Equation 5.10 on a quantum device, it is not necessary to know the value of the normalization $\mathcal{N}_\mathbb{X}$. However, if the goal is to estimate $K(\mathbb{X}, \mathbb{Y})$ with the help of a quantum device, then knowledge of the normalizations is needed, see (5.12). The naive approach, i.e., using its definition (5.11), takes $O(n^2)$ operations and would prohibit the polynomial advantage. In Appendix D.3 we show how one can estimate $\mathcal{N}_\mathbb{X}$. The suggested strategy only relies on the previous two challenges and hence does not pose a difficulty by itself.

[107]: Cincio et al. (2018), *Learning the quantum algorithm for state overlap*

[135]: Filip (2002), *Overlap and entanglement-witness measurements*; [136]: Pregnell (2006), *Measuring Nonlinear Functionals of Quantum Harmonic Oscillator States*; [137]: Jeong et al. (2014), *Detecting the degree of macroscopic quantumness using an overlap measurement*

5.5. Chapter conclusion

In this chapter, we adapted the concept of kernel mean embeddings to quantum mechanics, by defining the quantum mean embedding. While the kernel mean embedding maps a probability distribution to a function in a reproducing kernel Hilbert space, the quantum mean embedding can only map onto the unit sphere of a Hilbert space, a necessity that arises due to the normalization of quantum states. Despite this additional constraint, we showed that the quantum mean embedding is still injective if the induced kernel is c_0-universal. Since the quantum mean embedding can, in principle, be created in the lab, it allows for a polynomial speedup when computing inner products between mean embeddings of empirical distributions. We highlighted the relevance of this task by describing use cases in recent machine learning applications. We made explicit which requirements need to be fulfilled by the quantum hardware in order to harvest the polynomial advantage.

These insights open multiple paths for further research. On the quantum side, the experimental creation of superpositions of a large number of states and the estimation of inner products thereof. Furthermore, the quantum mean embedding is a new way of encoding probability distributions in quantum states, which allows us to use the results known from the kernel theory. For machine learning research, it is an open question what the possible applications of the embedding of probability distributions onto the unit sphere in the reproducing kernel Hilbert space could be.

Embedding conditional moment restrictions in the RKHS

6. Kernel conditional moment test

In this chapter we leave the two-sample problem behind and propose a new family of specification tests called kernel conditional moment (KCM) tests. Our tests are built on a novel representation of conditional moment restrictions in a reproducing kernel Hilbert space (RKHS) called conditional moment embedding (CMME). After transforming the conditional moment restrictions into a continuum of unconditional counterparts, the test statistic is defined as the maximum moment restriction (MMR) within the unit ball of the RKHS. We show that the MMR not only fully characterizes the original conditional moment restrictions, leading to consistency in both hypothesis testing and parameter estimation, but also has an analytic expression that is easy to compute as well as closed-form asymptotic distributions. Our empirical studies show that the KCM test has a promising finite-sample performance compared to existing tests.

6.1. Introduction

Many problems in causal inference, economics, and finance are often formulated as a conditional moment restriction (CMR): for correctly specified models, the conditional mean of certain functions of data is almost surely equal to zero [138, 139]. Rational expectation models—widely used in many fields of macroeconomics—specify how economic agents exploit available information to form their expectations in terms of conditional moments [140]. Recent advances in causal machine learning also rely on the CMR including a generalized random forest (GRF) [141], orthogonal random forest (ORF) [142], double machine learning (DML) [143], and nonparametric instrumental variable regression [144, 145] among others; see also [146–148] and references therein.

Checking the validity of these moment restrictions is the first and foremost step to ensure that a model is correctly specified which constitutes a fundamental assumption for its estimation and inference. A model misspecification often creates biases to parameter estimates, inconsistency of standard errors, and invalid asymptotic distributions that hinder our subsequent inference based on the model. An overidentifying restriction test in the generalized method of moments (GMM) framework is one of the standard approaches to test a *finite* number of *unconditional* moment conditions [149, 150]. The J-test is an example of such tests [149, 151], and numerous tests have been developed in econometrics to deal with various sources of misspecification; see, e.g., [152] for a review. This chapter focuses on an important class of CMR-based specification tests known as the conditional moment (CM) tests [153, 154] which have a long history in econometrics [152, 155, 156].

Testing *conditional* moment restrictions becomes more challenging as an *infinite* number of equivalent unconditional moment restrictions (UMR) must be examined simultaneously (Section 6.3). At first, [153]

[138]: Newey (1993), *Efficient estimation of models with conditional moment restrictions*; [139]: Ai et al. (2003), *Efficient Estimation of Models with Conditional Moment Restrictions Containing Unknown Functions*

[140]: Muth (1961), *Rational Expectations and the Theory of Price Movements*

[141]: Athey et al. (2019), *Generalized random forests*

[142]: Oprescu et al. (2019), *Orthogonal Random Forest for Causal Inference*

[143]: Chernozhukov et al. (2018), *Double/debiased machine learning for treatment and structural parameters*

[144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*; [145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*

[146]: Hartford et al. (2017), *Deep IV: A Flexible Approach for Counterfactual Prediction*; [147]: Singh et al. (2019), *Kernel Instrumental Variable Regression*; [148]: Muandet et al. (2020), *Dual instrumental variable regression*

[149]: Hansen (1982), *Large Sample Properties of Generalized Method of Moments Estimators*; [150]: Hall (2005), *Generalized Method of Moments*

[149]: Hansen (1982), *Large Sample Properties of Generalized Method of Moments Estimators*; [151]: Sargan (1958), *The Estimation of Economic Relationships using Instrumental Variables*

[152]: Bierens (2017), *Econometric Model Specification: Consistent Model Specification Tests and Semi-nonparametric Modeling and Inference*

[153]: Newey (1985), *Maximum Likelihood Specification Testing and Conditional Moment Tests*; [154]: Tauchen (1985), *Diagnostic testing and evaluation of maximum likelihood models*

[152]: Bierens (2017), *Econometric Model Specification: Consistent Model Specification Tests and Semi-nonparametric Modeling and Inference*; [155]: Hausman (1978), *Specification Tests in Econometrics*; [156]: White (1981), *Consequences and Detection of Misspecified Nonlinear Regression Models*

[153]: Newey (1985), *Maximum Likelihood Specification Testing and Conditional Moment Tests*

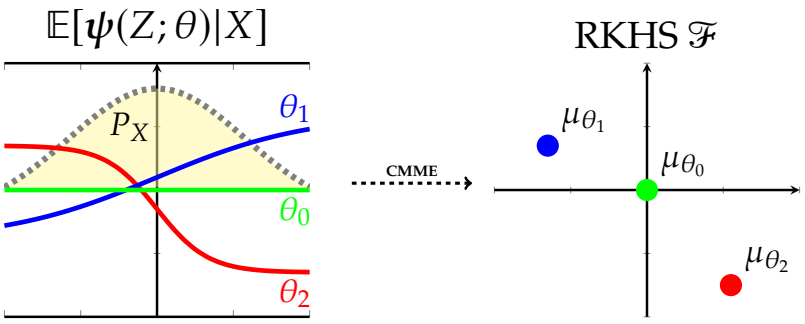


Figure 6.1.: Conditional moment embedding (CMME): The conditional moments $\mathbb{E}[\psi(Z;\theta)|X]$ for different parameters θ are *uniquely* (P_X-almost surely) embedded into the RKHS. The RKHS norm of μ_θ measures to what extent these restrictions are violated and hence is used as a test statistic for conditional moment tests.

and [154] proposed to perform the overidentifying restriction test on a finite subset of the UMR. Unfortunately, the CM tests that rely only on a finite number of moment conditions cannot be consistent against all alternatives. Additional assumptions such as the global identification of selected moment conditions and sample-size dependent moment conditions are required to guarantee consistency [157, 158]. To overcome this limitation, [159] introduced the first consistent CM tests—known as integrated conditional moment (ICM) tests—by checking *all* moment conditions simultaneously [160]. However, the ICM test depends on parametric weighting functions and nuisance parameters that limit its practical use. An alternative class of consistent CM tests, known as smooth tests, employ nonparametric kernel estimation [161, 162] which also forms a basis for the generalized empirical likelihood approach [163, 164]. However, they have non-trivial power only against local alternatives that approach the null at a slower rate than $1/\sqrt{n}$, and are susceptible to the curse of dimensionality (see Section 6.5 for the discussion).

[154]: Tauchen (1985), *Diagnostic testing and evaluation of maximum likelihood models*

[157]: Jong (1996), *The Bierens test under data dependence*; [158]: Donald et al. (2003), *Empirical Likelihood Estimation and Consistent Tests with Conditional Moment Restrictions*

[159]: Bierens (1982), *Consistent model specification tests*

[160]: Bierens et al. (1997), *Asymptotic Theory of Integrated Conditional Moment Tests*

[161]: Zheng (1996), *A consistent test of functional form via nonparametric estimation techniques*; [162]: Li et al. (1998), *A simple consistent bootstrap test for a parametric regression function*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*; [164]: Tripathi et al. (2003), *Testing conditional moment restrictions*

Inspired by a surge of kernel-based tests [5, 14, 35], we propose to embed the CMR in a reproducing kernel Hilbert space (RKHS). By transforming CMR into a continuum of UMR in RKHS, the test statistic is defined as the maximum moment restriction (MMR) within the unit ball of the RKHS (Section 6.3). We then show that the MMR corresponds to the RKHS norm of a Hilbert space embedding of conditional moments. *Not only can the MMR capture all information about the original CMR, but it also has a closed-form expression that enables the practical ease of implementation* (Theorems 6.3.3 and 6.3.4). The MMR allows us to develop a class of consistent CM tests that we call kernel conditional moment (KCM) tests (Section 6.4). Furthermore, it considerably simplifies the parameter estimation problems based on the CMR. Our framework has relationships to existing methods in econometrics and machine learning (Section 6.5). To the best of our knowledge, the Hilbert space embedding of conditional moment restrictions has not appeared elsewhere in the literature.[1]

[5]: Gretton et al. (2012), *A kernel two-sample test*; [14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

1: [165] and their follow-up work are the most relevant works from the econometric literature. We discuss this connection in Section 6.5.

All proofs can be found in Appendix E.4. The code of the experiments of this chapter is available at `https://github.com/krikamol/kcm-test`.

6.2. Background

We introduce the CMR in Subsection 6.2.1 and then review the concepts of kernels and RKHS in Subsection 6.2.2. Finally, we discuss the main assumptions in Subsection 6.2.3.

6.2.1. Conditional moment restrictions

Let Z be a random variable taking values in $\mathcal{Z} \subseteq \mathbb{R}^p$ with distribution P_Z, X a subvector of Z taking values in $\mathcal{X} \subseteq \mathbb{R}^d$ with distribution P_X, and $\Theta \subset \mathbb{R}^r$ a parameter space. Following [138], we consider models where the only available information about the unknown parameter $\theta_0 \in \Theta$ is a set of conditional moment restrictions

[138]: Newey (1993), *Efficient estimation of models with conditional moment restrictions*

$$\mathcal{M}(X;\theta_0) := \mathbb{E}[\boldsymbol{\psi}(Z;\theta_0)|X] = \mathbf{0}, \quad P_X\text{-a.s.}, \tag{6.1}$$

where $\boldsymbol{\psi} : \mathcal{Z} \times \Theta \to \mathbb{R}^q$ is a vector of *generalized residual functions* whose functional forms are known up to the parameter $\theta \in \Theta$. The expectation is always taken over all random variables that are not conditioned on. Note that there can be two different models that are *observationally equivalent* on the basis of (6.1) alone although an ideal parameter θ_0 is unique.

Several statistical problems can be formulated as Equation 6.1. In nonparametric regression models, $Z = (X, Y)$ where $Y \in \mathbb{R}$ is a dependent variable and $\boldsymbol{\psi}(Z;\theta) = Y - f(X;\theta)$. For conditional quantile models, $Z = (X, Y)$ and $\boldsymbol{\psi}(Z;\theta) = \mathbf{1}\{Y < f(X;\theta)\} - \tau$ for the target quantile $\tau \in [0, 1]$. In heterogeneous effect estimation, $Z = (X, T, Y)$ where T is a vector of treatments and $\boldsymbol{\psi}(Z;\theta(X)) = (Y - \langle\theta(X), T\rangle)T$. For instrumental variable regression, $Z = (X, W, Y)$ where W is an instrumental variable and $\boldsymbol{\psi}(Z;\theta) = (Y - f_\theta(X))$ and $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|W] = 0$ almost surely. When Z admits the density $p(z;\theta)$, we can define the moment conditions in terms of the *score function* as $\boldsymbol{\psi}(Z;\theta) = \nabla_\theta \log p(Z;\theta)$ and use it for local maximum likelihood estimation.

Conditional moment tests. Given an independent sample $(x_i, z_i)_{i=1}^n$ drawn from a distribution that satisfies the conditional moments (6.1) and an estimate $\hat{\theta}$ of θ_0, our goal is to perform specification testing: *Given a function $\boldsymbol{\psi}$ and a parameter estimate $\hat{\theta}$, we test the null hypothesis*

$$H_0 : \ \mathbb{E}[\boldsymbol{\psi}(Z;\hat{\theta}) \,|\, X] = \mathbf{0}, \quad P_X\text{-a.s.} \tag{6.2}$$

For instance, in the test of functional form of the nonlinear regression model [155], the null hypothesis can be expressed as $H_0 : \mathbb{E}[Y - f(X;\hat{\theta})|X] = 0$ where $\hat{\theta} = \arg\min_{\theta\in\Theta} \mathbb{E}[(Y - f(X;\theta))^2]$. In this case, $Z = (Y, X)$ and $\boldsymbol{\psi}(Z;\theta) = Y - f(X;\theta)$. This test allows us to detect misspecifications of the functional form of f.

[155]: Hausman (1978), *Specification Tests in Econometrics*

In this work, we assume that $\hat{\theta}$ is obtained independently of the data that is used to test the null hypothesis (6.2). In many cases, however, $\hat{\theta}$ is estimated using this data and hence the test performance is also subject to the estimation error. A generalization of our framework to those cases will require more involved analyses, and we leave it to future work.

6.2.2. Reproducing kernels

We superficially introduced reproducing kernels in Chapter 1. Here we give a more detailed introduction to relevant concepts. Let $\mathcal{X}$ be a nonempty set and $\mathcal{F}$ a Hilbert space consisting of functions on $\mathcal{X}$ with $\langle\cdot,\cdot\rangle_\mathcal{F}$ and $\|\cdot\|_\mathcal{F}$ being its inner product and norm, respectively. The Hilbert space $\mathcal{F}$ is called a reproducing kernel Hilbert space (RKHS) if there

exists a symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ called the reproducing kernel of $\mathcal{F}$ such that (i) $k(x, \cdot) \in \mathcal{F}$ for all $x \in \mathcal{X}$ and (ii) $f(x) = \langle f, k(x, \cdot)\rangle_{\mathcal{F}}$ for all $f \in \mathcal{F}$ and $x \in \mathcal{X}$. The latter is called the *reproducing property* of $\mathcal{F}$. Every positive definite kernel k uniquely determines the RKHS for which k is a reproducing kernel [110].

[110]: Aronszajn (1950), *Theory of Reproducing Kernels*

Let $\{(\lambda_j, e_j)\}$ be pairs of positive eigenvalues and orthonormal eigenfunctions of k, i.e., $\int e_i(x)e_j(x)\,\mathrm{d}x = 1$ if $i = j$ and zero otherwise. By Mercer's theorem [98, Thm 4.49], the kernel k has the spectral decomposition

[98]: Steinwart et al. (2008), *Support Vector Machines*

$$k(x, x') = \sum_j \lambda_j e_j(x) e_j(x'), \quad x, x' \in \mathcal{X}, \tag{6.3}$$

where the convergence is absolute and uniform. As a result, for any $f \in \mathcal{F}$, we have $f(x) = \sum_j f_j e_j(x)$ with $\sum_j f_j^2/\lambda_j < \infty$ where $f_j = \langle f, e_j\rangle_{\mathcal{F}}$, $\langle f, g\rangle_{\mathcal{F}} = \sum_j f_j g_j/\lambda_j$, and $\|f\|_{\mathcal{F}}^2 = \langle f, f\rangle_{\mathcal{F}} = \sum_j f_j^2/\lambda_j$.

Next, we introduce the notion of integrally strictly positive definite (ISPD) kernels and Bochner's characterization.

Definition 6.2.1 *A kernel $k(x, x')$ is integrally strictly positive definite (ISPD) if for any function f that satisfies $0 < \|f\|_2^2 < \infty$,*

$$\int_{\mathcal{X}} f(x)k(x, x')f(x')\,\mathrm{d}x\,\mathrm{d}x' > 0.$$

ISPD kernels are an important notion in kernel methods and are closely related to characteristic and universal kernels, see, e.g., [111].

[111]: Simon-Gabriel et al. (2018), *Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions*

The next result characterizes shift-invariant kernels $k(x, x') = \varphi(x - x')$ for some positive definite φ.

Theorem 6.2.1 (Bochner) *A continuous function $\varphi : \mathbb{R}^d \to \mathbb{C}$ is positive definite if and only if it is the Fourier transform of a finite nonnegative Borel measure Λ on $\mathbb{R}^d$:*

$$\varphi(t) = \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} e^{-it^\top \omega}\,\mathrm{d}\Lambda(\omega)$$

for $t \in \mathbb{R}^d$.

Examples of popular kernels are the Gaussian RBF kernel $k(x, x') = \exp(-\|x - x'\|_2^2/2\sigma^2), \sigma > 0$, Laplacian kernel $k(x, x') = \exp(-\|x - x'\|_1/\sigma), \sigma > 0$, and inverse multiquadric (IMQ) kernel $k(x, x') = (c^2 + \|x - x'\|_2^2)^{-\gamma}, c, \gamma > 0$. See, e.g., [98, Ch. 4] for more examples.

[98]: Steinwart et al. (2008), *Support Vector Machines*

6.2.3. Main assumptions

Our subsequent analyses rely on these key assumptions.

(A1) The random vector (X, Z) forms a strictly stationary process with the probability measure P_{XZ}.

(A2) *Regularity conditions*: (i) the function $\boldsymbol{\psi} : \mathcal{Z} \times \Theta \to \mathbb{R}^q$ where $q < \infty$ is continuous on Θ for each $z \in \mathcal{Z}$; (ii) $\mathbb{E}[\boldsymbol{\psi}(Z; \theta)|x]$ exists and

is finite for every $\theta \in \Theta$ and $x \in \mathcal{X}$ for which $P_X(x) > 0$; (iii) $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|x]$ is continuous on Θ for all $x \in \mathcal{X}$ for which $P_X(x) > 0$.

(A3) *Global identification*: there exists a unique $\theta_0 \in \Theta$ for which $\mathbb{E}[\boldsymbol{\psi}(Z;\theta_0)|X] = \mathbf{0}$ a.s., and $P(\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|X] = \mathbf{0}) < 1$ for all $\theta \in \Theta, \theta \neq \theta_0$.

(A4) The kernel k is ISPD, continuous, and bounded, i.e.,

$$sup_{x\in\mathcal{X}} \sqrt{k(x,x)} < \infty.$$

Assumption **(A1)** ensures that all expectations of functions of (X, Z) are independent of time. The regularity conditions **(A2)** are standard assumptions [150, Ch. 3] which ensure that $\boldsymbol{\psi}$ is well-defined, and hold in most models considered in the literature [150]. By contrast, **(A3)** may not hold, especially in non-linear models. A *local* identifiability can be assumed instead by imposing additional constraints on Θ. Testing whether the constraints are sufficient can then be done, for example, by examining the Jacobian at some parameter values [150, pp. 54]. Lastly, **(A4)** implies that the RKHS $\mathcal{F}$ consists of bounded continuous functions [98, Sec. 4.3] and is expressive enough (Theorem 6.3.3).

[150]: Hall (2005), *Generalized Method of Moments*

[98]: Steinwart et al. (2008), *Support Vector Machines*

6.3. Mamximum moment restriction

This section presents the RKHS representation of the CMR. Let $\mathcal{F}$ be a set of measurable functions on $\mathcal{X}$. Then,

$$\mathbb{E}_{XZ}[\boldsymbol{\psi}(Z;\theta)f(X)] = \mathbb{E}_X[\mathbb{E}_Z[\boldsymbol{\psi}(Z;\theta)f(X)|X]] = \mathbb{E}_X[\mathcal{M}(X;\theta)f(X)]$$

for any $f \in \mathcal{F}$ by the law of iterated expectation. That is, the CMR in Equation 6.1 implies an infinite set of unconditional moment restrictions

$$\mathbb{E}[\boldsymbol{\psi}(Z;\theta_0)f(X)] = \mathbf{0}, \quad \forall f \in \mathcal{F}. \tag{6.4}$$

Equivalently, any $\theta_0 \in \Theta$ that satisfies Equation 6.4 must also satisfy what we call a *maximum moment restriction* (MMR)

$$\sup_{f\in\mathcal{F}} \|\mathbb{E}[\boldsymbol{\psi}(Z;\theta_0)f(X)]\|_2^2 = 0. \tag{6.5}$$

It is known that the implied moment restrictions (6.4) and (6.5) can be insufficient to globally identify the parameters of interest. We call $\mathcal{F}$ for which Equation 6.5 implies Equation 6.1 a *sufficient class of instruments*. In the context of this work, $\mathcal{F}$ must consist of infinitely many instruments for the CM test to be consistent against all alternatives. However, the sup operator also makes it hard to optimize Equation 6.5. We resolve these issues by choosing $\mathcal{F}$ to be a unit ball in a RKHS, which we show to be a sufficient class of instruments. As a result, Equation 6.5 can be solved analytically, the parameters of interest can be consistently estimated, and the resulting CM test is consistent against all fixed alternatives.

Recently, [145] and [144] also propose to estimate θ_0 based on Equation 6.5 and $\mathcal{F}$ that is parameterized by deep neural networks. While they consider an estimation problem, we focus on hypothesis testing problems. Nevertheless, our formulation of CMR can also be used to estimate θ_0 (Subsection 6.3.2 and Appendix E.2). Note that the algorithms proposed

[145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*

[144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*

in [145] and [144] require solving a minimax game, whereas our approach for estimation is simply a minimization problem.

[145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*

[144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*

6.3.1. Conditional moment embedding

To express Equation 6.5 using the RKHS, we first develop a representation of the CMR in a vector-valued RKHS of functions $f : \mathcal{X} \to \mathbb{R}^q$ [166]. Let $\mathcal{F}$ be the RKHS of real-valued functions on $\mathcal{X}$ with reproducing kernel k and $\mathcal{F}^q$ the product RKHS of functions $f := (f_1, \ldots, f_q)$ where $f_i \in \mathcal{F}$ for all i with an inner product $\langle f, g \rangle_{\mathcal{F}^q} = \sum_{i=1}^q \langle f_i, g_i \rangle_{\mathcal{F}}$ and norm $\|f\|_{\mathcal{F}^q} = \sqrt{\sum_{i=1}^q \|f_i\|_{\mathcal{F}}^2}$. For $\theta \in \Theta$, we define an operator M_θ on $\mathcal{F}^q$ as

[166]: Álvarez et al. (2012), *Kernels for Vector-Valued Functions: A Review*

$$M_\theta f := \mathbb{E}[\boldsymbol{\psi}(Z;\theta)^\top f(X)] = \sum_{i=1}^q \mathbb{E}[\psi_i(Z;\theta) f_i(X)],$$

where ψ_i denotes the i-th component of $\boldsymbol{\psi}$. This operator takes an instrument $f \in \mathcal{F}^q$ as input and returns the corresponding conditional moment restrictions.

The following lemma shows that M_θ satisfies the property of the original conditional moment restrictions.

Lemma 6.3.1 *For all $f \in \mathcal{F}^q$, $M_{\theta_0} f = 0$.*

Moreover, by Assumption **(A2)** and **(A4)**,

$$M_\theta f| \leq \sum_{i=1}^q \|f_i\|_{\mathcal{F}_i} \sqrt{\mathbb{E}[\psi_i(Z;\theta)\psi_i(Z';\theta)k(X,X')]} < \infty,$$

where (X', Z') is an independent copy of (X, Z). Hence, M_θ is a bounded linear operator. By Riesz's representation theorem, there exists a unique element $\boldsymbol{\mu}_\theta$ in $\mathcal{F}^q$ such that $M_\theta f = \langle f, \boldsymbol{\mu}_\theta \rangle_{\mathcal{F}^q}$ for all $f \in \mathcal{F}^q$. Indeed, by the reproducing property,

$$M_\theta f = \sum_{i=1}^q \left\langle f_i, \mathbb{E}[\xi_\theta^i(X,Z)] \right\rangle_{\mathcal{F}_i} = \langle f, \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)] \rangle_{\mathcal{F}^q},$$

where $\boldsymbol{\xi}_\theta(x,z) := \big(\psi_1(z;\theta)k(x,\cdot), \ldots, \psi_q(z;\theta)k(x,\cdot)\big)$ is the feature map in $\mathcal{F}^q$ and ξ_θ^i denotes the i-th element of $\boldsymbol{\xi}_\theta$. The equalities above are well-defined since $\boldsymbol{\xi}_\theta(x,z)$ is Bochner integrable [98, Def. A.5.20], i.e., $\mathbb{E}\|\boldsymbol{\xi}_\theta(X,Z)\|_{\mathcal{F}^p} \leq \sqrt{\mathbb{E}\|\boldsymbol{\xi}_\theta(X,Z)\|_{\mathcal{F}^p}^2} = \sqrt{\mathbb{E}[\boldsymbol{\psi}(Z;\theta)^\top \boldsymbol{\psi}(Z;\theta)k(X,X)]} < \infty$.

[98]: Steinwart et al. (2008), *Support Vector Machines*

In other words, $\boldsymbol{\mu}_\theta := \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)]$ is a *representer* of M_θ in $\mathcal{F}^q$. We define $\boldsymbol{\mu}_\theta$ as *conditional moment embedding* (CMME) of $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|X]$ in $\mathcal{F}^q$ relative to P_X.

Definition 6.3.1 *For each $\theta \in \Theta$, let*

$$\boldsymbol{\xi}_\theta(x,z) := \big(\psi_1(z;\theta)k(x,\cdot), \ldots, \psi_q(z;\theta)k(x,\cdot)\big) \in \mathcal{F}^q.$$

The conditional moment embedding *(CMME) is defined as*

$$\mu_\theta := \int_{\mathcal{X}\times\mathcal{Z}} \xi_\theta(x,z)\, dP_{XZ}(x,z) \in \mathcal{F}^q. \tag{6.6}$$

The CMME μ_θ takes the form of a kernel mean embedding of P_{XZ} with ξ_θ as the feature map [19]. This is illustrated in Figure 6.1. Hence, given an i.i.d. sample $(x_i, z_i)_{i=1}^n$ from P_{XZ}, we can estimate μ_θ simply by $\widehat{\mu}_\theta := \frac{1}{n}\sum_{i=1}^n \xi_\theta(x_i, z_i)$. The following theorem establishes the $\sqrt{n}$-consistency of this estimator.

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

Theorem 6.3.2 *Let* $\sigma_\theta^2 := \mathbb{E}\|\xi_\theta(X,Z)\|_{\mathcal{F}^q}^2$ *and assume that*

$$\|\xi_\theta(X,Z)\|_{\mathcal{F}^q} < C_\theta < \infty$$

almost surely. Then, for any $0 < \delta < 1$*, with probability at least* $1-\delta$*,*

$$\left\|\widehat{\mu}_\theta - \mu_\theta\right\|_{\mathcal{F}^p} \le \frac{2C_\theta \log\frac{2}{\delta}}{n} + \sqrt{\frac{2\sigma_\theta^2 \log\frac{2}{\delta}}{n}}. \tag{6.7}$$

Remarkably, $\widehat{\mu}_\theta$ converges at a rate $O_p(n^{-1/2})$ that is independent of the dimension of (X,Z) and the RKHS $\mathcal{F}^q$. This is an appealing property because estimation and inference based on $\widehat{\mu}_\theta$ become less susceptible to the *curse of dimensionality* (see, e.g., [167] and references therein for the discussion). Under certain assumptions, [168] established the minimax optimal rate for the kernel mean estimators like $\widehat{\mu}_\theta$.

[167]: Khosravi et al. (2019), *Non-Parametric Inference Adaptive to Intrinsic Dimension*

[168]: Tolstikhin et al. (2017), *Minimax Estimation of Kernel Mean Embeddings*

The next theorem shows that μ_θ provides a *unique* representation of the CMR $\mathcal{M}(X,\theta)$ in $\mathcal{F}^q$ relative to P_X.

Theorem 6.3.3 *Assume that the kernel* k *is ISPD. Then, for any* $\theta_1, \theta_2 \in \Theta$*,* $\mathcal{M}(x;\theta_1) = \mathcal{M}(x;\theta_2)$ *for* P_X*-almost all* x *if and only if* $\mu_{\theta_1} = \mu_{\theta_2}$*.*

To better understand Theorem 6.3.3, consider when $q = 1$ and $k(x,x') = \varphi(x - x')$ is a shift-invariant kernel. First, we have

$$\begin{aligned}\mu_\theta(\cdot) &= \mathbb{E}_X[\mathbb{E}_Z[\psi(Z;\theta)k(X,\cdot)|X]] \\ &= \mathbb{E}_X[\mathbb{E}_Z[\psi(Z;\theta)|X]k(X,\cdot)] \\ &= \mathbb{E}_X[\mathcal{M}(X;\theta)k(X,\cdot)].\end{aligned}$$

It is then easy to show using Theorem 6.2.1 that

$$\mu_\theta(\cdot) = \int_{\mathbb{R}^d} \phi(\omega;\theta)c(\omega,\cdot)\,\mathrm{d}\Lambda(\omega)$$

where $c(\omega, y) = \exp(i\omega^\top y) \neq 0$ and

$$\phi(\omega;\theta) := \mathbb{E}_X[\mathcal{M}(X;\theta)\exp(i\omega^\top X)]$$

is the Fourier transform (or characteristic function) of the Borel measurable function $\mathcal{M}(x;\theta)$ relative to P_X. Hence, if $\mathrm{supp}(\Lambda) = \mathbb{R}^d$, the uniqueness of μ_θ follows from the uniqueness of $\phi(\omega;\theta)$. [159] was the first to observe the characterization of the CMR in terms of the integral transform and then used it to construct the consistent CM tests of functional form (cf. Section 6.5).

[159]: Bierens (1982), *Consistent model specification tests*

Theorem 6.3.3 shows that $\boldsymbol{\mu}_\theta$ captures all information about $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|x]$ for every $x \in \mathcal{X}$ for which $P_X(x) > 0$. Consequently, estimation and inference on CMR can be performed by means of $\boldsymbol{\mu}_\theta$ using the existing kernel arsenal. As mentioned earlier, for each $f \in \mathcal{F}^q$ and $\theta \in \Theta$, the inner product $\langle f, \boldsymbol{\mu}_\theta \rangle_{\mathcal{F}^q} = \langle f, \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)] \rangle_{\mathcal{F}^q}$ can be interpreted as a restriction of conditional moments with respect to f. Moreover, the investigator can inspect $\boldsymbol{\mu}_\theta(x,z)$, which measures to what extent the moment conditions are violated at (x,z), i.e., structural instability, in order to understand the nature of misspecification.

6.3.2. Maximum moment restriction with reproducing kernels

Based on the CMME $\boldsymbol{\mu}_\theta$, we can now define the MMR as

$$\mathbb{M}(\theta) := \sup_{\|f\|_{\mathcal{F}^q} \leq 1} M_\theta f = \sup_{\|f\|_{\mathcal{F}^q} \leq 1} \langle f, \boldsymbol{\mu}_\theta \rangle_{\mathcal{F}^q} = \|\boldsymbol{\mu}_\theta\|_{\mathcal{F}^q}. \tag{6.8}$$

By Theorem 6.3.3, $\mathbb{M}(\theta) \geq 0$ and $\mathbb{M}(\theta) = 0$ if and only if $\theta = \theta_0$. Put differently, $\mathbb{M}(\theta)$ measures how much the models associated with θ violate the original CMR in Equation 6.1.

To obtain an expression for $\mathbb{M}(\theta)$, we define a real-valued kernel $h_\theta : (\mathcal{X} \times \mathcal{Z}) \times (\mathcal{X} \times \mathcal{Z}) \to \mathbb{R}$ based on the feature map $\boldsymbol{\xi}_\theta : \mathcal{X} \times \mathcal{Z} \to \mathcal{F}^q$ as follows:

$$\begin{aligned} h_\theta((x,z),(x',z')) &:= \langle \boldsymbol{\xi}_\theta(x,z), \boldsymbol{\xi}_\theta(x',z') \rangle_{\mathcal{F}^q} \\ &= \boldsymbol{\psi}(z;\theta)^\top \boldsymbol{\psi}(z';\theta) k(x,x'). \end{aligned} \tag{6.9}$$

Then, a closed-form expression for $\mathbb{M}(\theta)$ in terms of the kernel h_θ follows straightforwardly.

Theorem 6.3.4 *Assume that* $\mathbb{E}[h_\theta((X,Z),(X,Z))] < \infty$. *Then,* $\mathbb{M}^2(\theta) = \mathbb{E}[h_\theta((X,Z),(X',Z'))]$ *where* (X',Z') *is independent copy of* (X,Z) *with the same distribution.*

Finally, Mercer's representation (6.3) of k allows us to interpret h_θ and $\mathbb{M}(\theta)$ in terms of a continuum of unconditional moment restrictions.

Theorem 6.3.5 *Let* $\{(\lambda_j, e_j)\}$ *be eigenvalue/eigenfunction pairs associated with the kernel* k *and* $\zeta^j_\theta(x,z) := \big(\psi_1(z;\theta)e_j(x), \ldots, \psi_q(z;\theta)e_j(x)\big)$. *Then, for each* $\theta \in \Theta$, $h_\theta((x,z),(x',z')) = \sum_j \lambda_j \zeta^j_\theta(x,z)^\top \zeta^j_\theta(x',z')$ *and* $\mathbb{M}^2(\theta) = \sum_j \lambda_j \|\mathbb{E}[\zeta^j_\theta(X,Z)]\|_2^2$.

That is, we can interpret $\mathbb{E}[\zeta^j_\theta(X,Z)]$ as the UMR with e_j acting as an instrument. Moreover, $\mathbb{M}^2(\theta)$ can be viewed as a weighted sum of moment restrictions based on the sequence of weights and instruments $(\lambda_j, e_j)_j$. As a result, the CM test based on $\mathbb{M}^2(\theta)$ as a test statistic examines an infinite number of moment restrictions. Note that $(\lambda_j, e_j)_j$ are defined implicitly by the choice of k.

6.4. Kernel conditional moment test with bootstrapping

By virtue of Theorem 6.3.3, we can reformulate the CM testing problem (6.2) in terms of the MMR as

$$H_0 : \mathbb{M}^2(\theta) = 0, \quad H_1 : \mathbb{M}^2(\theta) \neq 0.$$

Given an i.i.d. sample $\{(x_i, z_i)\}_{i=1}^n$ from the distribution P_{XZ}, we consider the test statistic

$$\widehat{\mathbb{M}}_n^2(\theta) = \frac{1}{n(n-1)} \sum_{1 \leq i \neq j \leq n} h_\theta((x_i, z_i), (x_j, z_j)), \tag{6.10}$$

which is in the form of U-statistics [32, Section 5]. Although there exist several potential estimators for $\mathbb{M}^2(\theta)$, we focus on Equation 6.10 as it is a minimum-variance unbiased estimator with appealing asymptotic properties. Moreover, Equation 6.10 also provides a basis for the estimation of θ_0 simply by minimizing $\widehat{\mathbb{M}}_n^2(\theta)$ with respect to $\theta \in \Theta$. Preliminary results on estimation are given in Appendix E.2.

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

Next, we characterize the asymptotic distributions of $\widehat{\mathbb{M}}_n^2(\theta)$ under the null and alternative hypotheses.[2]

2: Note that the asymptotic distribution takes the same form as the asymptotic distribution of the quadratic time MMD estimates that we introduced in Section 1.5.

Theorem 6.4.1 *Assume that* $\mathbb{E}[h_\theta^2((X, Z), (X', Z'))] < \infty$ *for all* $\theta \in \Theta$. *Let* $U := (X, Z)$ *and* $U' := (X', Z')$. *Then, the following statements hold.*

(1) *If* $\theta \neq \theta_0$, $\widehat{\mathbb{M}}_n^2(\theta)$ *is asymptotically normal with*

$$\sqrt{n}\left(\widehat{\mathbb{M}}_n^2(\theta) - \mathbb{M}^2(\theta)\right) \xrightarrow{d} \mathcal{N}(0, 4\sigma_\theta^2),$$

where $\sigma_\theta^2 = \mathrm{Var}_U\left[\mathbb{E}_{U'}[h_\theta(U, U')]\right]$.

(2) *If* $\theta = \theta_0$, *then* $\sigma_\theta^2 = 0$ *and*

$$n\widehat{\mathbb{M}}_n^2(\theta) \xrightarrow{d} \sum_{j=1}^{\infty} \tau_j \left(W_j^2 - 1\right), \tag{6.11}$$

where $W_j \sim \mathcal{N}(0, 1)$ *and* $\{\tau_j\}$ *are the eigenvalues of* $h_\theta(u, u')$, *i.e., they are the solutions of* $\tau_j \phi_j(u) = \int h_\theta(u, u')\phi_j(u')dP(u')$ *for non-zero* ϕ_j.

As we can see, $n\widehat{\mathbb{M}}_n^2(\theta) < \infty$ with probability one under the null $\theta = \theta_0$ and diverts to infinity at a rate $O(\sqrt{n})$ under any fixed alternative $\theta \neq \theta_0$. Hence, a consistent CM test can be constructed as follows: if $\gamma_{1-\alpha}$ is the $1 - \alpha$ quantile of the CDF of $n\widehat{\mathbb{M}}_n^2(\theta)$ under the null $\theta = \theta_0$, we reject the null with significance level α if $n\widehat{\mathbb{M}}_n^2(\theta) \geq \gamma_{1-\alpha}$.

Proposition 6.4.2 ([169]; p. 671) *Assume the conditions of Theorem 6.4.1. The test that rejects the null* $\theta = \theta_0$ *when* $n\widehat{\mathbb{M}}_n^2(\theta) > \gamma_{1-\alpha}$ *is consistent against any fixed alternative* $\theta \neq \theta_0$, *i.e., the limiting power of the test is one.*

Unfortunately, the limiting distribution in Equation 6.11 and its $1 - \alpha$ quantile do not have an analytic form. Following recent work on kernel-

Algorithm 3 KCM Test with bootstrapping

Input: Bootstrap sample size B, significance level α
for $t \in \{1, \dots, B\}$ **do**
  Draw $(w_1, \dots, w_n) \sim \text{Mult}(n; \frac{1}{n}, \dots, \frac{1}{n})$
  $\rho_i \leftarrow (w_i - 1)/n$ for $i = 1, \dots, n$
  $\widehat{\mathbb{M}}_n^*(\theta) \leftarrow \sum_{i \neq j} \rho_i \rho_j h_\theta((x_i, z_i), (x_j, z_j))$
  $a_t \leftarrow n\widehat{\mathbb{M}}_n^*(\theta)$
$\hat{\gamma}_{1-\alpha}$:= empirical $(1-\alpha)$-quantile of $\{a_t\}_{t=1}^B$
Reject H_0 if $\hat{\gamma}_{1-\alpha} < n\widehat{\mathbb{M}}_n^2(\theta)$ (see (6.10))

based tests [5, 14, 35], we propose to approximate the critical values using the bootstrap method proposed by [169, 170], which was previously used in [35]. Specifically, we first draw multinomial random weights $(w_1, \dots, w_n) \sim \text{Mult}(n; \frac{1}{n}, \dots, \frac{1}{n})$ and compute the bootstrap sample $\widehat{\mathbb{M}}_n^*(\theta) = (1/n^2) \sum_{1 \leq i \neq j \leq n} (w_i - 1)(w_j - 1) h_\theta((x_i, z_i), (x_j, z_j))$. We then calculate the empirical quantile $\hat{\gamma}_{1-\alpha}$ of $n\widehat{\mathbb{M}}_n^*(\theta)$. For degenerate U-statistics, $\hat{\gamma}_{1-\alpha}$ is a consistent estimate of $\gamma_{1-\alpha}$ [169, 170].

[5]: Gretton et al. (2012), *A kernel two-sample test*; [14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

[169]: Arcones et al. (1992), *On the Bootstrap of U and V Statistics*; [170]: Huskova et al. (1993), *Consistency of the Generalized Bootstrap for Degenerate U-Statistics*

We summarize our bootstrap kernel conditional moment (KCM) test in Algorithm 1. Note that the proposed test checks the CMR for a *given* parameter θ and does not take into account the estimation error of θ. We defer a full treatment of interplay between parameter estimation and hypothesis testing to future work.

6.5. Related work

Existing CM tests can generally be categorized into two classes. The former is based on a transformation of CMR into a continuum of unconditional counterparts, e.g., [159, 171], [157], [160], and [158] to name a few. The latter employs nonparametric kernel estimation which includes [161, 162, 172] among others. While both classes lead to consistent tests, they exhibit different asymptotic behaviors; see, e.g., [163, 172] for detailed comparisons.

[159]: Bierens (1982), *Consistent model specification tests*; [171]: Bierens (1990), *A Consistent Conditional Moment Test of Functional Form*

[157]: Jong (1996), *The Bierens test under data dependence*

[160]: Bierens et al. (1997), *Asymptotic Theory of Integrated Conditional Moment Tests*

[158]: Donald et al. (2003), *Empirical Likelihood Estimation and Consistent Tests with Conditional Moment Restrictions*

[161]: Zheng (1996), *A consistent test of functional form via nonparametric estimation techniques*; [162]: Li et al. (1998), *A simple consistent bootstrap test for a parametric regression function*; [172]: Fan et al. (2000), *Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*; [172]: Fan et al. (2000), *Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests*

A continuum of unconditional moments. One of the classical approaches is to find a parametric weighting function $w(x, \eta)$ such that

$$\mathbb{E}[\boldsymbol{\psi}(Z; \theta)|X] = \mathbf{0} \text{ a.s.} \iff \mathbb{E}[\boldsymbol{\psi}(Z; \theta) w(X, \eta)] = \mathbf{0},$$

for almost all $\eta \in \Xi \subseteq \mathbb{R}^m$ where η is a nuisance parameter. [153] and [154] proposed the so-called M-test using a finite number of weighting functions. Since it imposes only a finite number of moment conditions, the test cannot be consistent against all possible alternatives and power against specific alternatives depends on the choice of these weighting functions. [157] and [158] showed that this issue can be circumvented by allowing the number of moment conditions to grow with sample size. Although our KCM test generally relies on infinitely many moment conditions, one can impose finitely many conditions using the finite dimensional RKHS such as those endowed with linear and polynomial kernels or resorting to finite-dimensional kernel approximations.

[153]: Newey (1985), *Maximum Likelihood Specification Testing and Conditional Moment Tests*

[154]: Tauchen (1985), *Diagnostic testing and evaluation of maximum likelihood models*

[157]: Jong (1996), *The Bierens test under data dependence*

[158]: Donald et al. (2003), *Empirical Likelihood Estimation and Consistent Tests with Conditional Moment Restrictions*

[173] showed that there exists a wide range of $w(x, \eta)$ that lead to consistent CM tests. They call these functions "totally revealing". For instance, [159] proposed the first consistent specification test for nonlinear regression models using $w(x, \eta) = \exp(i\eta^\top x)$ for $\eta \in \mathbb{R}^d$. Similarly, [171] used $w(x, \eta) = \exp(\eta^\top x)$ for $\eta \in \mathbb{R}^d$. An indicator function $w(x, \eta) = \mathbf{1}(\alpha^\top x \leq \beta)$ with $\eta = (\alpha, \beta) \in \mathbb{S}^d \times (-\infty, \infty)$ where $\mathbb{S}^d = \{\alpha \in \mathbb{R}^d : \|\alpha\| = 1\}$ was used in [174] and [163]. Other popular weighting functions include power series, Fourier series, splines, and orthogonal polynomials, for example. In light of Theorem 6.3.5, the KCM test falls into this category where weighting functions are eigenfunctions associated with the kernel k.

[173]: Stinchcombe et al. (1998), *Consistent Specification Testing with Nuisance Parameters Present Only under the Alternative*

[159]: Bierens (1982), *Consistent model specification tests*

[171]: Bierens (1990), *A Consistent Conditional Moment Test of Functional Form*

[174]: Escanciano (2006), *A Consistent Diagnostic Test for Regression Models Using Projections*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*

Since $w(x, \eta)$ depends on the nuisance parameter η, [159] suggested to integrate η out, resulting in an *integrated conditional moment* (ICM) test statistic:

$$\widehat{T}_n(\theta) = \int_\Xi \|\widehat{Z}_n(\eta)\|_2^2 \, \mathrm{d}\nu(\eta), \tag{6.12}$$

where Ξ is a compact subset of $\mathbb{R}^d$, $\nu(\eta)$ is a probability measure on Ξ, and $\widehat{Z}_n(\eta) := (1/\sqrt{n}) \sum_i \boldsymbol{\psi}(z_i; \theta) w(x_i, \eta)$. The limiting null distribution of the ICM test was proven to be a zero-mean Gaussian process [171]. [160] also characterizes the asymptotic null distribution of a general class of real-valued weighting functions.

[159]: Bierens (1982), *Consistent model specification tests*

[171]: Bierens (1990), *A Consistent Conditional Moment Test of Functional Form*

[160]: Bierens et al. (1997), *Asymptotic Theory of Integrated Conditional Moment Tests*

The following theorem establishes the connection between the KCM and ICM test statistics.

Theorem 6.5.1 *Let $k(x, x') = \varphi(x - x')$ be a shift-invariant kernel on $\mathbb{R}^d$. Then, we have*

$$\mathbb{M}^2(\theta) = \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} \left\| \mathbb{E}[\boldsymbol{\psi}(Z; \theta) \exp(i\omega^\top X)] \right\|_2^2 \, \mathrm{d}\Lambda(\omega)$$

where Λ is a Fourier transform of k.

This theorem is quite insightful as it describes the KCM test statistic as the ICM test statistic $\widehat{T}_n(\theta)$ of [159] where the distribution on the nuisance parameter ω is a Fourier transform of the kernel. For instance, the Gaussian kernel $k(x, x') = \exp(-\|x - x'\|_2^2 / 2\sigma^2)$ corresponds to the Gaussian density $\Lambda(\omega) = \exp(-\sigma^2 \|\omega\|_2^2 / 2)$; see [19, Table 2.1] for more examples. Note that both weighting functions and integrating measures are implicitly determined by the kernel k. Unlike ICM tests, KCM tests can be evaluated without solving the high-dimensional numerical integration in Equation 6.12 explicitly. Moreover, KCM tests can be easily generalized to $\mathcal{X}$ that is not necessarily a subset of $\mathbb{R}^d$.

[159]: Bierens (1982), *Consistent model specification tests*

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

[165] also considers a similar setting that involves a continuum of moment conditions in RKHS. Their approach, however, differs significantly from ours. First, they consider a specific case where the Hilbert space is a set of square integrable functions of a scalar $t \in [0, T]$ with the unconditional moment conditions $\mathbb{E}[\boldsymbol{\psi}_t(X, \theta_0)] = \mathbf{0}$ for all $t \in [0, T]$. Second, their key question is to identify the optimal choice of weighting matrix in GMM. Third, estimation is actually based on a truncation of infinite moment conditions. Lastly, they also proposed the CM test similar to the ICM tests, but it can handle only the case with $Z \in \mathbb{R}$, while our test is applicable to any domain with a valid kernel.

[165]: Carrasco et al. (2000), *Generalization of GMM to a Continuum of Moment Conditions*

Nonparametric kernel estimation. The second class of tests, known as *smooth tests* [161, 162, 172], adopts the statistic of the form

$$T(\theta) = \mathbb{E}[\boldsymbol{\psi}(Z;\theta)^\top \mathbb{E}[\boldsymbol{\psi}(Z;\theta)|X]f(X)]. \tag{6.13}$$

[161]: Zheng (1996), *A consistent test of functional form via nonparametric estimation techniques*; [162]: Li et al. (1998), *A simple consistent bootstrap test for a parametric regression function*; [172]: Fan et al. (2000), *Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests*

Based on the kernel estimator of $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)|X]f(X)$, the empirical estimate of (6.13) can be expressed as

$$\widehat{T}_n(\theta) = \frac{1}{n(n-1)h^d} \sum_{1\le i\ne j\le n} \boldsymbol{\psi}(z_i;\theta)^\top \boldsymbol{\psi}(z_j;\theta) K_{ij} \tag{6.14}$$

where $K_{ij} = K((x_i - x_j)/h)$, $K(\cdot) : \mathbb{R}^d \to \mathbb{R}$ is a normalized kernel function and h is a smoothing parameter. Here, we emphasize that existing smooth tests rely on the kernel density estimator (KDE) in which the kernel used is not necessarily a reproducing kernel. For the smooth test to be consistent, h must vanish as $n \to \infty$, whereas our KCM test is consistent even when the kernel is fixed. Nevertheless, if $K(\cdot)$ is a reproducing kernel, the test statistic $\widehat{T}_n(\theta)$ with a fixed smoothing parameter h resembles the KCM test statistic (6.10). In fact, [172] has shown that the ICM test is a special case of the kernel-based test with a fixed smoothing parameter. However, the critical drawback of the nonparametric kernel-based tests is that they have non-trivial power only against local alternatives that approach the null at a slower rate than $1/\sqrt{n}$, due to the slower rate of convergence of kernel density estimators, i.e., $O((nh^{d/2})^{-1/2})$ as $h \to 0$ [172]. Moreover, these tests are susceptible to the curse of dimensionality.

[172]: Fan et al. (2000), *Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests*

[172]: Fan et al. (2000), *Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests*

Last but not least, the kernel estimator is also a key ingredient in empirical likelihood-based CM tests [164, 175, 176].

[164]: Tripathi et al. (2003), *Testing conditional moment restrictions*; [175]: Kitamura et al. (2004), *Empirical Likelihood-Based Inference in Conditional Moment Restriction Models*; [176]: Dominguez et al. (2004), *Consistent Estimation of Models Defined by Conditional Moment Restrictions*

Kernelized Stein discrepancy (KSD). Stein's methods [177] are among the most popular techniques in statistics and machine learning. One notable example is the Stein discrepancy which aims to characterize complex, high-dimensional distribution $p(x) = \tilde{p}(x)/N$ with intractable normalization constant $N = \int \tilde{p}(x)\,\mathrm{d}x$ using a *Stein operator* $\mathcal{A}_p$ such that

$$p = q \quad \Leftrightarrow \quad \mathbb{E}_{x\sim q}[\mathcal{A}_p f(x)] = 0, \quad \forall f, \tag{6.15}$$

[177]: Stein (1972), *A bound for the error in the normal approximation to the distribution of a sum of dependent random variables*

where $\mathcal{A}_p f(x) := \nabla_x \log p(x) f(x) + \nabla_x f(x)$. Here, we assume for simplicity that $x \in \mathbb{R}$. The Stein operator $\mathcal{A}_p$ depends on the density p through its *score function* $s_p(x) := \nabla_x \log p(x) = \frac{\nabla_x p(x)}{p(x)}$, which is independent of N. When $p \ne q$, the expectation in (6.15) gives rise to a discrepancy

$$\begin{aligned} \mathbb{S}_f(p,q) &:= \mathbb{E}_{x\sim q}[\mathcal{A}_p f(x)] \\ &= \mathbb{E}_{x\sim q}[(s_p(x) - s_q(x))f(x)]. \end{aligned} \tag{6.16}$$

See, also, [35, Lemma 2.3]. The Stein discrepancy has led to numerous applications such as variance reduction [178] and goodness-of-fit testing [14, 35], among others.

[35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

[178]: Oates et al. (2017), *Control functionals for Monte Carlo integration*

Like Equation 6.4, we can observe that Equation 6.15 is indeed a set of unconditional moment conditions. To make an explicit connection between Stein discrepancy and CMR, we need to assume access to the probability densities. Let $\mathcal{P}_\Theta$ be a space of probability densities $p(z;\theta)$

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

such that $\theta \mapsto p(z;\theta)$ is injective. We choose $\boldsymbol{\psi}(z;\theta) = \nabla_z \log p(z;\theta) =: s_\theta(z)$ as the associated score function.[3] This yields the following CMR:

$$\mathbb{E}[\nabla_z \log p(Z;\theta_0) \mid X] = \mathbf{0}, \quad P_X\text{-a.s.} \tag{6.17}$$

3: This differs from the standard definition of score function as $\nabla_\theta \log p(z|\theta)$ in the interpretation of maximum likelihood as generalized method of moments [150].

For any $\theta \in \Theta$, it follows that $\mathbb{E}[\boldsymbol{\psi}(Z;\theta)^\top f(X)] = \mathbb{E}[s_\theta(Z)^\top f(X) - s_{\theta_0}(Z)^\top f(X)] = \mathbb{E}[(s_\theta(Z) - s_{\theta_0}(Z))^\top f(X)] =: \Delta_f(\theta,\theta_0)$. While $\Delta_f(\theta,\theta_0)$ resembles the Stein discrepancy in Section 6.5, we highlight the key differences. First, this characterization requires that the model is correctly specified, i.e., $p(z;\theta_0)$ is observationally indistinguishable from the underlying data distribution. Second, like the Stein discrepancy, it can be interpreted as the $f(x)$-weighted expectation of the score difference $s_\theta - s_{\theta_0}$. In contrast, the weighting function $f(x)$ in our setting depends only on X, which is a subvector of Z. We provide further discussion about this discrepancy measure in Appendix E.1. The following theorem follows directly from the preceeding observation.

Theorem 6.5.2 *Let $\mathcal{P}_\Theta$ be a space of probability densities $p(z;\theta)$. Assume that $\theta \mapsto p(z;\theta)$ is injective and $\theta_0 \in \Theta$. If $\boldsymbol{\psi}(z;\theta) = \nabla_z \log p(z;\theta)$ and $X = Z$, we have $\mathbb{S}_f(p(z;\theta), p(z;\theta_0)) = \Delta_f(\theta,\theta_0)$.*

Mostly related to our work are the RKHS-based Stein's methods [14, 35]. Specifically, if we assume the conditions of Theorem 6.5.2 and that f belongs to the RKHS, it follows that $\Delta(\theta,\theta_0) := \sup_f \|\Delta_f(\theta,\theta_0)\|_2$ coincides with the kernelized Stein discrepancy (KSD) proposed in [35] and [14].

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*; [35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

[35]: Liu et al. (2016), *A Kernelized Stein Discrepancy for Goodness-of-fit Tests*

[14]: Chwialkowski et al. (2016), *A Kernel Test of Goodness of Fit*

6.6. Experiments

We report the finite-sample performance of the KCM test against two well-known consistent CM tests, namely ICM test and smooth test, as discussed in Section 6.5. We evaluate all tests with a bootstrap size $B = 1000$ and a significance level $\alpha = 0.05$.

(1) **`KCM`**: The bootstrap KCM test using U-statistic in Algorithm 3. We use the RBF kernel with bandwidth chosen by the median heuristic.

(2) **`ICM`**: The test based on an integration over weighting functions. Following [179] and [163], we use Equation 6.12 as the test statistic with $w(x,\eta) = \mathbf{1}(x \le \eta) = \prod_{j=1}^d \mathbf{1}(x_j \le \eta_j)$ where $\mathbf{1}(\cdot)$ is an indicator function. The density ν is chosen to be the empirical distribution of X. This leads to a simple test statistic $t_n = \sum_{i=1}^n r_n(x_i)^\top r_n(x_i)$ where $r_n(x) := \frac{1}{n}\sum_{i=1}^n \boldsymbol{\psi}(z_i;\theta)\mathbf{1}(x_i \le x)$. We follow the bootstrap procedure in [163, Sec. 4.3] to compute the critical values.

(3) **`Smooth`**: The test based on nonparametric kernel estimation. We use Equation 6.14 as the test statistic. The kernel is the standard Gaussian density function whose bandwidth is chosen by the rule-of-thumb $h = n^{-1/5}$. Note that the median heuristic is not applicable here because the bandwidth h does not vanish, as required. The critical values are obtained using the same bootstrap procedure as in [163, Sec. 4.2].

[179]: Stute (1997), *Nonparametric model checks for regression*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*

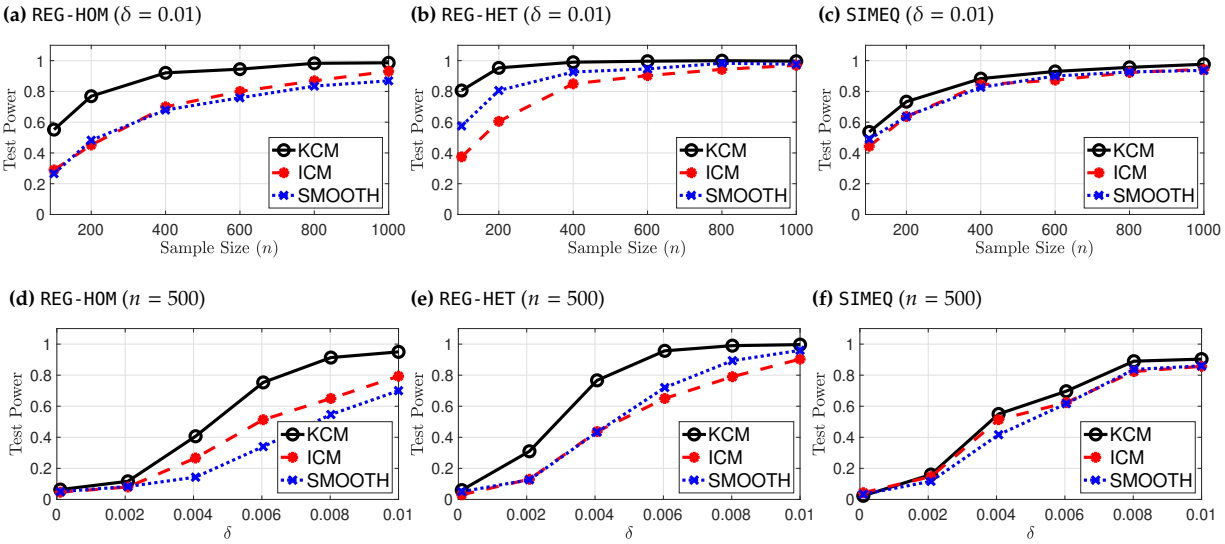


Figure 6.2.: The test powers of KCM, ICM, and smooth tests averaged over 300 trials as we vary the values of n (top) and δ (bottom). Type-I errors of these tests are shown in Figure E.1 in Appendix E.3.2. See main text for the interpretation.

Testing a regression function (REG). We follow a similar simulation of regression model used in [180]. In this setting, for a given estimate $\hat{\boldsymbol{\beta}}$ of the regression parameters, the null hypothesis is

$$H_0 : \mathbb{E}[Y - \hat{\boldsymbol{\beta}}^\top X \mid X] = 0 \quad \text{a.s.}$$

where $X \in \mathbb{R}^d$ and Y is a univariate random variable, i.e., $Z = (Y, X)$. The data are generated from the data generating process (DGP):

$$Y = \boldsymbol{\beta}_0^\top X + e.$$

We set $\boldsymbol{\beta}_0 = \mathbf{1}$, and $X \sim \mathcal{N}(0, I_d)$. For the error term e, we consider two scenarios: (i) *Homoskedastic* (HOM): $e = \epsilon, \epsilon \sim \mathcal{N}(0, 1)$ and (ii) *Heteroskedastic* (HET): $e = \epsilon\sqrt{0.1 + 0.1\|X\|_2^2}$. In each trial, we obtain an estimate of $\boldsymbol{\beta}_0$ by $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_0 + \gamma$ where $\gamma \sim \mathcal{N}(\mathbf{0}, \delta^2 I_d)$. In this experiment, we set $d = 5$. When $\delta = 0$, the CMR are fulfilled, whereas they are violated, i.e., H_0 is false, if $\delta \neq 0$. Different values of δ correspond to different degrees of deviation from the null.

[180]: Lavergne et al. (2016), *A Hausman Specification Test of Conditional Moment Restrictions*

Testing the simultaneous equation model (SIMEQ). Following [181] and [163], we consider the equilibrium model

$$Q = \alpha_d P + \beta_d R + U, \quad \alpha_d < 0, \qquad \text{(Demand)}$$
$$Q = \alpha_s P + \beta_s W + V, \quad \alpha_s > 0, \qquad \text{(Supply)}$$

where Q and P denote quantity and price, respectively, R and W are exogeneous variables, and U and V are the error terms. In this setting, $Z = (Q, P, R, W)$ and $X = (R, W)$. The null hypothesis can be expressed as

$$H_0 : \mathbb{E}\left[\begin{array}{c} Q - \alpha_d P - \beta_d R \\ Q - \alpha_s P - \beta_s W \end{array} \middle| \, X \right] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

a.s. for some $\theta_0 = (\alpha_d, \beta_d, \alpha_s, \beta_s)$. We generate data according to $Q = \lambda_{11} R + \lambda_{12} W + V_1$ and $P = \lambda_{21} R + \lambda_{22} W + V_2$ where R and

[181]: Newey (1990), *Efficient Instrumental Variables Estimation of Nonlinear Models*

[163]: Delgado et al. (2006), *Consistent Tests of Conditional Moment Restrictions*

W are independent standard Gaussian random variables while V_1 and V_2 are correlated standard Gaussian random variables with 10^{-3} variance and $10^{-3}/\sqrt{2}$ covariance, and independent of (R, W). We set $(\lambda_{11}, \lambda_{12}, \lambda_{21}, \lambda_{22}) = (1, -1, 1, 1)$ and provide the details on how to find the true parameters θ_0 in Appendix E.3.1. The estimate $\hat{\theta}$ is obtained as in the previous experiment. The null hypothesis corresponds to $\delta = 0$ and different values of δ corresponds to alternative hypotheses. Rejecting H_0 means that the functional form of the supply and demand curves are misspecified.

Figure 6.2 depicts the empirical results for $n \in \{1, 2, 4, 6, 8, 10\} \times 10^2$ and $\delta \in \{10^{-4}, 2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}, 10^{-2}\}$. First, it can be observed that KCM, ICM, and smooth tests are all capable of detecting the misspecification as the sample size and δ are sufficiently large. Second, the KCM test tends to outperform both ICM and smooth tests in terms of the test power, especially in a low sample regime (see Figure 6.2a–6.2c) and a small deviation regime (see Figure 6.2d–6.2f). In addition, the smooth test and the ICM test are competitive: there is no substantial evidence to conclude that one is always better than the other. Lastly, Figure E.1 in Appendix E.3.2 depicts that the Type-I errors of all tests are correctly controlled at $\alpha = 0.05$.

Lastly, we point out that this work does not elaborate on the effect of parameter estimation. In practice, the candidate parameter $\hat{\theta}$ has to be estimated from the observed data, which changes the asymptotic distribution of the test statistic. We envision the interplay between parameter estimation and hypothesis testing as an important arena for future work.

6.7. Chapter conclusion

To conclude this chapter: we propose a new conditional moment test called the KCM test whose statistic is based on a novel representation of the conditional moment restrictions in a reproducing kernel Hilbert space. This representation captures all necessary information about the original conditional moment restrictions. Hence, the resulting test is consistent against all fixed alternatives, is easy to use in practice, and also has connections to existing tests in the literature. It also has an encouraging finite-sample performance compared to those tests. While the conditional moment restrictions have a long history in econometrics and so does the concept of reproducing kernel Hilbert spaces in machine learning, the intersection of these concepts remains unexplored. We believe that this work gives rise to a new and promising framework for conditional moment restrictions which constitute numerous applications in econometrics, causal inference, and machine learning.

Conclusion

7. Conclusion and outlook

Hypothesis testing is a subfield of statistics that until today remained much more in the hand of rigorous theory than other statistics-related fields like for example supervised machine learning. While also in supervised learning great progress has been made through theoretical analysis, much of its empirical success is due to persistent engineering and clever heuristics.

In this thesis we attempted to improve hypothesis tests somewhere in between. We did not make any assumptions about how our distributions differ and hence also were not able to theoretically show that our procedures are statistically optimal in any sense. Other works are able to derive optimal results under appropriate assumptions [65, 182]. Nevertheless in practice, it can be either hard to make such strong assumptions or there are still some hyperparameters left for the user to choose. We thus focused on deriving theoretically well-motivated strategies with the goal of improving the testing pipeline in *practice*. These contributions were quite conceptual and should be applicable to a wide range of problems. Incorporating expert knowledge about the problem at hand is thus still left to the practitioners. Therefore, we did also not put much focus on engineering for a particular type of data. After all, one of our main insights (Chapter 4) was that we can reuse approaches developed for supervised learning.

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*; [182]: Kim et al. (2022), *Minimax optimality of permutation tests*

We now give an outlook on a few (non exhaustive) aspects that are worthwhile and should be considered in the future:

1. What strategies other than selective inference can be used to avoid data splitting?
2. Do the practical benefits of data splitting outweigh the decrease of the set on which the significance is computed?
3. If we split the data, is there a way to choose in which proportion to split the training and test set?
4. How do we need to adapt existing heuristics?

Avoiding data splitting. In Chapter 2 we focused on the post-selection inference framework to circumvent data splitting. This required strong assumptions about the distribution of the test statistic under the null hypothesis, which limits its applications. It turns out, however, that one can think of other approaches to circumvent data-splitting. One way is to define a test statistic that completely contains the optimization. In principle, it could even contain model selection and cross-validation et cetera. In the two-sample problem, we could then still estimate valid p-values via permutations (Lemma 1.2.1). Note that in this case, however, for each permutation *the whole* pipeline has to be reran, including all engineering steps. This likely yields a prohibitive cost for many applications. Instead, in our proposed witness test one only needs to train once, evaluate the witness on the test set once and the cost of computing the $B \in \mathbb{N}$ permutations is in practice negligibly small (Equation 4.8). We are not aware of any work that investigated the possibility of simulating the whole pipeline in detail.

As we briefly introduced in the introduction using a Bonferroni correction is a simple way to combine multiple tests, but that commonly leads to overly conservative tests. [65, 183] recently proposed a procedure to aggregate multiple kernel tests that determines the threshold by a joint simulation of the aggregated test. They make an initial (empirical) comparison to the OST introduced in Chapter 2, but from a theoretical viewpoint it remains to be understood which approach is better.

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*; [183]: Schrab et al. (2022), *Efficient Aggregated Kernel Tests using Incomplete U-statistics*

Practical benefits of data splitting. The procedure of [65] also works with the quadratic-time MMD estimates, thus overcoming a limitation of our OST which relied on the less accurate linear-time estimate. However, the aggregated test also has the limitation of only combining finitely many predefined test statistics. This blocks to harness the benefits of gradient-based optimization, which in turn has also been argued to be helpful for hypothesis testing [8]. As mentioned above, if simulating the p-value by repeating the entire optimization pipeline, of course gradient-based optimization becomes feasible again. But beyond this, it is hard to imagine a procedure that does not repeat the entire pipeline and still can appropriately adjust for continuous or even non-convex optimization strategies. As we have seen, being able to adjust the tests required well-behaved null distributions (Chapter 2) or finitely many tests [65]. Either way, it is important to analytically understand the optimization. This is not the case for more general optimization strategies. We thus conclude that for the time being, the only practically feasible method to include complex optimization strategies is to split the data. Beyond this, data splitting also prevents from accidental p-value hacking, which might indeed be a problem for practitioners when handling to complex procedures without data splitting.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[65]: Schrab et al. (2021), *MMD Aggregated Two-Sample Test*

Is online partitioning of data useful? Let us come back to the data-splitting approaches once more. As we have seen in Chapters 2 and 3, the proportion in which the data is split into training and testing part influences the test power. Qualitatively speaking, the larger the class of tests we optimize over, the more data should be used for training. In the extreme case, if we do not learn anything, i.e., the test is fixed upfront, clearly we should use no data for learning, and instead all the data for testing. Beyond these qualitative insights, we were not able to give guidance on choosing the splitting ratio in practice and defaulted to a 50/50 split. But due to its relevance for the performance, investigating the splitting ratio in more detail, seems a promising and relevant direction for both theoretical and practical research. On the theoretical side, understanding how the splitting ratio should be scaled as a function of the sample size could bring important insights. For the practical aspect, we shortly sketch a potential strategy.

Firstly, note that if we use data splitting, the testing phase should control the Type-I error independently of the sample size used for testing. We only need to ensure that the testing data is independent of the training data. This allows us to dynamically assign more and more data for training. If, say, after using 10% of the data, we estimate that using more data for training, there is nothing that stops us from decreasing the testing set and increasing the training set. Note that, of course, the other

way is not allowed, i.e., once some data was in the training set, we cannot put it back into the test set.

A rough idea for a strategy in case of a witness two-sample test could be the following: during training one keeps an estimate of the SNR of the witness function. Using the asymptotic distribution and the size of the held-out samples, one could then estimate the probability that the test would reject given the prespecified level α. On the other hand, one could try to track how much the SNR would improve if more data was used for training. Based on this information one could estimate whether improving the SNR at the cost of a decreased test set could improve the test power. If this is the case, one could assign more data to the learning phase. Otherwise one stops the training and continues with the testing phase. Working out such a strategy would give us yet another practical tool to optimize hypothesis tests.

Adaption of heuristics. While our contributions are theoretically founded, this thesis is nevertheless also a call for pragmatism in hypothesis testing. We showed that we can largely incorporate tools and heuristics that initially are developed for other tasks. While our empirical results show that this works well, it is nevertheless conceivable that some heuristics need adaption. One reason is that when testing hypotheses, the signal might generally be much lower than, say, in a standard classification task, where one knows that the classes are different and solely maximizes performance.

Statistical significance does not imply relevance. We end this thesis with some remarks of cautiousness. We focused on making tests as powerful as possible. If a test rejects, we conclude that there is a statistically significant violation of the null hypothesis. But this does not necessarily imply that this violation is *relevant* in the considered setting. Firstly, notice that even the faintest violation can be detected if the datasets are appropriately large. Secondly, the detected violation, although strong, might be irrelevant for us. In the two-sample problem, imagine we would compare image data, collected at two hospitals. Our goal would be to test whether there is a difference in the groups from which the data was collected. But additionally there is a defect in one pixel at the first hospitals imaging device. Our test might then spot this and reject the null hypothesis, although this is not what we initially looked for. There are a few ways to prevent such situations. After detecting a significant violation of the null hypothesis, one could interpret the data to understand how this violation came about. In the two-sample problem, this could be done by interpreting the data as in Subsection 4.3.3 and Figure 4.2. Experts should then be able to determine whether the found difference is indeed relevant to them. An alternative way is to test the hypothesis on pretrained features that are relevant to the task at hand. In the distribution shift benchmark, for example, we did this by using pretrained features. [18] also classified shifts into *harmful* and *harmless* changes. Recently, [93] also tried to detect relevant changes in distributions.

[18]: Rabanser et al. (2019), *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift*

[93]: Zhao et al. (2022), *Comparing Distributions by Measuring Differences that Affect Decision Making*

Appendix

Appendix of Chapter 2 A.

A.1. Proof of Theorem 2.3.2

In this section we prove the main theorem. The outline of the proof is as follows: We first characterize the "selection event", i.e., we characterize under which conditions each active set $\mathcal{U}$ is selected. This is done with Lemmas A.1.1 and A.1.2. For the case $l = 1$ we then show that the PSI framework of [17] can be applied and we recover the result of Corollary 2.3.1. It is not surprising, that for the case $l = 1$ the PSI framework works, since $\mathcal{U}$ corresponds to a single fixed $\boldsymbol{\beta}^*$ and the probability of selecting it is greater than 0. For the case $l \geq 2$, we show, that the considered test statistic essentially takes the same form as the Wald test but only on the active dimensions. Thus it follows a χ_l distribution. This distribution does not change even if we explicitly condition on the selection of $\mathcal{U}$. This is because the randomness that determines which active set is selected is independent of the value of the selected test statistic. Before we start with the proof we collect some notation we introduce for the proof.

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

Notation:

- The objective of the optimization $f(\boldsymbol{\beta}) := \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$.
- Projector onto the active subspace (leaving the dependency on $\mathcal{U}$ implicit):

$$\Pi := \sum_{u \in \mathcal{U}} e_u e_u^\top,$$

  where e_u denotes the u-th Cartesian unit vector in $\mathbb{R}^d$.
- $z := \left(I_d - \frac{\Sigma \boldsymbol{\beta}^* \boldsymbol{\beta}^{*\top}}{\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*}\right) \boldsymbol{\tau} = \boldsymbol{\tau} - \Sigma \boldsymbol{\beta}^* \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*}$.
- $\bar{\Sigma}$ denotes the pseudoinverse of $\Pi \Sigma \Pi$.

As a first step, we need to characterize which values of $\boldsymbol{\tau}$ correspond to which active set $\mathcal{U}$. This is done with Lemma A.1.1, which we prove separately in Appendix A.1.1.

Lemma A.1.1 *Let $\mathcal{U} := \{u \mid \beta_u^* \neq 0\}$. Then,*

$$\boldsymbol{\beta}^* = \underset{\|\boldsymbol{\beta}\|=1, \boldsymbol{\beta} \geq 0}{\operatorname{argmax}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$$

if and only if all of the following conditions hold:

1. $\frac{\partial}{\partial \beta_u} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^*} \begin{cases} \leq 0 \text{ if } u \notin \mathcal{U} & (a), \\ = 0 \text{ if } u \in \mathcal{U} & (b), \end{cases}$
2. $\frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \geq \frac{\tau_u}{\sqrt{\Sigma_{uu}}}, \qquad \forall u \notin \mathcal{U},$
3. $\begin{aligned} &\boldsymbol{\beta}_u^* = 0 \quad \forall u \notin \mathcal{U} \quad (a), \\ &\boldsymbol{\beta}_u^* > 0 \quad \forall u \in \mathcal{U} \quad (b), \\ &\|\boldsymbol{\beta}^*\| = 1 \quad (c). \end{aligned}$

Intuitively, Condition 1(b) ensures that $\boldsymbol{\beta}^*$ is a local maximum of the objective function for the active dimensions. Condition 1(a) ensures that if $u \notin \mathcal{U}$, increasing β^*_u does not improve the SNR. Condition 2 is harder to interpret, but is needed in cases where all entries of $\boldsymbol{\tau}$ are negative. Condition 3 enforces that $\boldsymbol{\beta}^*$ lies in the feasible set of Equation 2.7.

Note that $\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}$ is essentially a one-dimensional RV. We define another random variable

$$z := \left(I_d - \frac{\Sigma\boldsymbol{\beta}^*\boldsymbol{\beta}^{*\top}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*}\right)\boldsymbol{\tau} = \boldsymbol{\tau} - \Sigma\boldsymbol{\beta}^*\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*}. \tag{A.1}$$

In Appendix A.1.2, we show that z is closely related to the partial derivatives of the objective function and we have

$$\frac{\partial}{\partial\beta_u}\frac{\boldsymbol{\beta}^\top\boldsymbol{\tau}}{(\boldsymbol{\beta}^\top\Sigma\boldsymbol{\beta})^{\frac{1}{2}}}\bigg|_{\beta=\boldsymbol{\beta}^*} = \frac{z}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}. \tag{A.2}$$

We can then rewrite the conditions of Lemma A.1.1 as follows.

Lemma A.1.2 *The conditions of Lemma A.1.1 are equivalent to*

1. $\begin{cases} z_u \le 0 & \forall u \notin \mathcal{U} \quad (a), \\ z_u = 0 & \forall u \in \mathcal{U} \quad (b), \end{cases}$
2. $\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}} \ge \mathcal{V}^-(z)$, *with*
   $\mathcal{V}^-(z) := \max_{u\notin\mathcal{U}} \frac{z_u(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}{\Sigma_{uu}^{\frac{1}{2}}(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}} - (\Sigma\boldsymbol{\beta}^*)_u}$,
3. $\begin{aligned} &\boldsymbol{\beta}^*_u = 0 \quad \forall u \notin \mathcal{U} \quad (a), \\ &\boldsymbol{\beta}^*_u > 0 \quad \forall u \in \mathcal{U} \quad (b), \\ &\|\boldsymbol{\beta}^*\| = 1 \quad (c). \end{aligned}$

Proof of Lemma A.1.2. Condition 1 directly follows from (A.2). The second condition follows by inserting the definition of z

$$\begin{aligned}
&\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}} \ge \frac{\tau_u}{\sqrt{\Sigma_{uu}}} \\
\Leftrightarrow &\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}} \ge \frac{z_u}{\sqrt{\Sigma_{uu}}} + e_u^\top\Sigma\boldsymbol{\beta}^*\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*\sqrt{\Sigma_{uu}}} \\
\Leftrightarrow &\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}\left(1 - \frac{e_u^\top\Sigma\boldsymbol{\beta}^*}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}\sqrt{\Sigma_{uu}}}\right) \ge \frac{z_u}{\sqrt{\Sigma_{uu}}} \\
\Leftrightarrow &\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}\left((\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}\sqrt{\Sigma_{uu}} - e_u^\top\Sigma\boldsymbol{\beta}^*\right) \ge z_u(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}} \\
\Leftrightarrow &\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}} \ge \frac{z_u(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}{\left((\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}\sqrt{\Sigma_{uu}} - e_u^\top\Sigma\boldsymbol{\beta}^*\right)},
\end{aligned}$$

where we used $\Sigma_{uu}^{\frac{1}{2}}(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}} - (\Sigma\boldsymbol{\beta}^*)_u > 0$, which holds since Σ is positive and we only consider u such that $e_u \neq \boldsymbol{\beta}^*$. □

Note that $\mathcal{V}^-(z)$ is always non-positive by Condition 1 and the positivity of Σ. With the above two lemmas we are able to prove Theorem 2.3.2.

Proof of Theorem 2.3.2. We prove the two cases $l = 1$ and $l \geq 2$ separately.

1.): Let $u^* \in [d]$ such that $\mathcal{U} = \{u^*\}$. In this case, by Condition 3, $\boldsymbol{\beta}^* = e_{u^*}$. We shall now see how Lemma A.1.2 constrains the distribution of τ_{u^*}. For Condition 1(b), we have $z_{u^*} = 0$ by the definition of $\boldsymbol{z}$. So there only remain the constraints 1(a) and 2. Using the definition (A.1) of $\boldsymbol{z}$, we can rewrite 1(a) as

$$\left(\left(I_d - \Sigma e_{u^*} \frac{e_{u^*}^\top}{\Sigma_{u^*u^*}}\right)\boldsymbol{\tau}\right)_u \leq 0 \quad \forall u \notin \mathcal{U} \iff A^{[1(b)]}\boldsymbol{\tau} \leq 0,$$

where $A^{[1(b)]}$ is the matrix $\left(I_d - \Sigma e_{u^*} \frac{e_{u^*}^\top}{\Sigma_{u^*u^*}}\right)$ and we used that its u-th row contains only zeros. Note that Condition 2 is the same as used in Subsection 2.3.1. Thus we can define the matrix $A^{[2]}$ as we do in the proof of Corollary 2.3.1. We have now all the remaining constraints as linear inequalities of $\boldsymbol{\tau}$ and thus we can find the conditional distribution by applying Theorem A.3.1. Defining $\boldsymbol{\eta} = \frac{e_{u^*}}{(\boldsymbol{\beta}^{*T}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}$ and $\boldsymbol{c} := \Sigma\boldsymbol{\eta}\left(\boldsymbol{\eta}^\top\Sigma\boldsymbol{\eta}\right)^{-1}$, we get $A^{[1(b)]}\boldsymbol{c} = \boldsymbol{0}$. Note that whenever $(A\boldsymbol{c})_j = 0$, the constraint does not change anything in Theorem A.3.1. Thus the result follows by using $A = A^{[2]}$ and application of Theorem A.3.1.

An alternative proof can be done by noting that $\boldsymbol{z}$ is independent of $\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}$ if we consider $\boldsymbol{\beta}^* = e_{u^*}$ as fixed. Thus, the fulfillment of Condition 1b) is independent of $\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}$. Since the unconditional distribution of $\frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}$ follows a standard normal, adding Condition 2 results in a truncated normal.

2.) Next, we consider the case $|\mathcal{U}| \geq 2$. Again we will be considering the conditions as stated in Lemma A.1.2. As we state in Equation A.9, we have $\boldsymbol{\beta}^{*\top}\boldsymbol{\tau} \geq 0$ and thus Condition 2 is fulfilled, since $\mathcal{V}^-$ is always non-positive. Thus, we can neglect Condition 2. Our first step will be to find a closed form function $h_{\mathcal{U}}$ such that $\boldsymbol{\beta}^* = h_{\mathcal{U}}(\boldsymbol{\tau})$ (this function will only hold true if $\mathcal{U}$ is actually the active set). Defining the projector onto the active subspace $\Pi := \sum_{u\in\mathcal{U}} e_u e_u^\top$, by Condition 3(a) we have $\boldsymbol{\beta}^* = \Pi\boldsymbol{\beta}^*$. Using Equation A.1, we can rewrite Condition 1(b) as

$$\Pi\boldsymbol{z} = \boldsymbol{0} \overset{(A.1)}{\Leftrightarrow} \Pi\boldsymbol{\tau} = \Pi\Sigma\boldsymbol{\beta}^* \frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*} \overset{3(a)}{\Leftrightarrow} \Pi\boldsymbol{\tau} = \Pi\Sigma\Pi\boldsymbol{\beta}^* \frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*}. \tag{A.3}$$

This defines a system of l non-trivial equations and by Condition 3, $\boldsymbol{\beta}^*$ has l free parameters. We define $\bar{\Sigma}$ as the pseudoinverse of $\Pi\Sigma\Pi$.[1] For the pseudoinverse it is easy to show $\bar{\Sigma} = \Pi\bar{\Sigma} = \bar{\Sigma}\Pi$. Since Σ has full rank, a possible solution of Equation A.3 necessarily has to be of the form $\boldsymbol{\beta}^* = c \cdot \bar{\Sigma}\boldsymbol{\tau}$ for some $c \in \mathbb{R}$. Plugging this into Equation A.3, we get $c = \frac{\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*}{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}$. Using Equation A.9 we get $0 \leq \frac{\boldsymbol{\beta}^{*\top}\boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top}\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}} = \frac{1}{c}$. Hence, $c \geq 0$. Using $\|\boldsymbol{\beta}^*\| = 1$ we get $c = \frac{1}{\|\bar{\Sigma}\boldsymbol{\tau}\|}$. Thus, given that the active set is $\mathcal{U}$, we

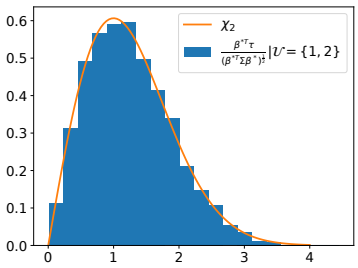


Figure A.1.: Numerical verification of Theorem 2.3.2. For the histogram, we generate a random covariance matrix $\Sigma \in \mathbb{R}^{4\times4}$ and sample $\boldsymbol{\tau} \sim \mathcal{N}(\boldsymbol{0}, \Sigma)$. We solve (2.7) and only accept the samples for which the active set is $\mathcal{U} = \{1, 2\}$. The orange line is the theoretical distribution according to Theorem 2.3.2, which is given by a chi distribution with two degrees of freedom. For the specific example the acceptance rate is $P(\mathcal{U} = \{1, 2\}) \approx 4\%$.

1: For intuition, assume without loss of generality that $\mathcal{U} = \{1, \ldots, l\}$. The pseudoinverse is then simply the inverse of the $l \times l$ blockmatrix padded with zeros.

found a closed-form solution for $\boldsymbol{\beta}^*$ as a function of $\boldsymbol{\tau}$, i.e.,

$$\boldsymbol{\beta}^* = h_{\mathcal{U}}(\boldsymbol{\tau}) := \frac{\bar{\Sigma}\boldsymbol{\tau}}{\|\bar{\Sigma}\boldsymbol{\tau}\|}. \tag{A.4}$$

Note that so far we did not use Condition 3(b), so this formula itself does not ensure the positivity of $\boldsymbol{\beta}^*$.

Replacing $\boldsymbol{\beta}^*$ in the definition (A.1) of $\boldsymbol{z}$ with its closed form, the constant c cancels, and we get

$$\boldsymbol{z} = \boldsymbol{\tau} - \Sigma\bar{\Sigma}\boldsymbol{\tau}.$$

Note that $\bar{\Sigma}\Pi\Sigma\Pi\bar{\Sigma} = \bar{\Sigma}$ and $(\Sigma\bar{\Sigma})_{uu'} = \delta_{uu'}$ if $u, u' \in \mathcal{U}$. This implies that $z_u = 0$ if $u \in \mathcal{U}$ and thus also $\boldsymbol{z}^\top\bar{\Sigma}\boldsymbol{\tau} = 0$.

Let us now define $\tilde{X} := (\bar{\Sigma})^{\frac{1}{2}}\boldsymbol{\tau}$, resulting in $\tilde{X}_u = 0$ for all $u \notin \mathcal{U}$. Since $\tilde{X}$ and $\boldsymbol{z}$ are both linear transformations of $\boldsymbol{\tau}$ they are jointly normally distributed. In Appendix A.1.3 we show that $\tilde{X}$ and $\boldsymbol{z}$ are uncorrelated. This, together with the joint normality, implies that they are independent, i.e.,

$$\tilde{X} \perp\!\!\!\perp \boldsymbol{z}. \tag{A.5}$$

Further the non-zero coordinates of $\tilde{X}$ are jointly distributed according to a l-dimensional standard normal distribution. Hence, its euclidean norm follows a chi-distribution

$$\|\tilde{X}\|_2 \sim \chi_l. \tag{A.6}$$

Let us summarize how we used all the conditions of Lemma A.1.2 and finish the proof. We used 1(b), 3(a), and 3(c) to show Equation A.4. We thus still need to condition on 1(a), and 3(b). Conditioning on 1(a) can be done using the independence of $\boldsymbol{z}$ and $\tilde{X}$. To condition on 3(b), we rewrite it in terms of $\tilde{X}$, i.e., for all $u \in \mathcal{U}$ we have

$$\boldsymbol{\beta}^*_u > 0 \Leftrightarrow \left(\bar{\Sigma}\boldsymbol{\tau}\right)_u \Leftrightarrow \left((\bar{\Sigma})^{\frac{1}{2}}\tilde{X}\right)_u > 0 \Leftrightarrow \left((\bar{\Sigma})^{\frac{1}{2}}\frac{\tilde{X}}{\|\tilde{X}\|}\right)_u > 0.$$

Thus it only depends on the direction of $\tilde{X}$. Since the non-trivial entries of $\tilde{X}$ follow a standard normal, the direction of $\tilde{X}$ is independent of its norm, i.e.,

$$\|\tilde{X}\|_2 \perp\!\!\!\perp \frac{\tilde{X}}{\|\tilde{X}\|_2}. \tag{A.7}$$

In the end we get

$$\left[\frac{\boldsymbol{\beta}^*\boldsymbol{\tau}}{(\boldsymbol{\beta}^*\Sigma\boldsymbol{\beta}^*)^{\frac{1}{2}}}\middle|\text{Conditions } 1,2,3\right] \overset{(A.4)}{\to} \overset{d}{=} \left[\frac{\boldsymbol{\tau}^\top\bar{\Sigma}\boldsymbol{\tau}}{(\boldsymbol{\tau}\bar{\Sigma}\boldsymbol{\tau})^{\frac{1}{2}}}\middle|\text{Conditions } 1(a), 3(b)\right]$$
$$\overset{d}{=} \left[\|\tilde{X}\|_2\middle|\begin{cases} z_u \le 0 & \forall u \notin \mathcal{U}, \\ \left((\bar{\Sigma})^{\frac{1}{2}}\frac{\tilde{X}}{\|\tilde{X}\|}\right)_u > 0 & \forall u \in \mathcal{U} \end{cases}\right] \underset{(A.7)}{\overset{(A.5)}{\to}} \overset{d}{=} \left[\|\tilde{X}\|_2\right] \overset{d}{=} \chi_l.$$

□

A.1.1. Proof of Lemma A.1.1

Proof of Lemma A.1.1. Since the objective is a homogeneous function of order zero in $\boldsymbol{\beta}$, we can make the proof by considering the optimization without the constraint $\|\boldsymbol{\beta}\| = 1$.

The necessity of the conditions is trivial to show. We thus only show the sufficiency. The fourth condition ensures that $\boldsymbol{\beta}^*$ is in the feasible set. For the other conditions, assume there exists $\boldsymbol{\xi} \in \mathbb{R}^d$ such that $\xi_u \geq 0$ for all $u \in [d]$ and $\frac{\boldsymbol{\xi}^\top \boldsymbol{\tau}}{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} > \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}}$. In the following we show that this implies that at least one of the conditions above is violated, and hence the conditions are sufficient. We separate two cases, *i*) where $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} \geq 0$, and *ii*) $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} < 0$.

i) Assume $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} \geq 0$. We have

$$
\begin{aligned}
&\boldsymbol{\xi}^\top \nabla_\beta \left. \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} \right|_{\beta=\beta^*} \\
&= \sum_{u \in [d]} \xi_u \frac{\partial}{\partial \beta_u} \left. \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} \right|_{\beta=\beta^*} \\
&= \frac{\boldsymbol{\xi}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} - \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{3}{2}}} \boldsymbol{\xi}^\top \Sigma \boldsymbol{\beta}^* \\
&= \frac{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \left(\frac{\boldsymbol{\xi}^\top \boldsymbol{\tau}}{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} - \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \frac{\boldsymbol{\xi}^\top \Sigma \boldsymbol{\beta}^*}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}} (\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} \right) \\
&> \frac{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \left(\frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} - \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \frac{\boldsymbol{\xi}^\top \Sigma \boldsymbol{\beta}^*}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}} (\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} \right) \\
&= \frac{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \left(1 - \frac{\boldsymbol{\xi}^\top \Sigma \boldsymbol{\beta}^*}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}} (\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} \right) \\
&\geq 0,
\end{aligned}
$$

where we used the assumption $\frac{\boldsymbol{\xi}^\top \boldsymbol{\tau}}{(\boldsymbol{\xi}^\top \Sigma \boldsymbol{\xi})^{\frac{1}{2}}} > \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}}$ for the first inequality and $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} \geq 0$ and the Cauchy-Schwarz inequality to arrive at the last line. Since, by assumption, $\xi_u \geq 0$ for all u, this implies $\frac{\partial}{\partial \beta_u} \left. \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} \right|_{\beta=\beta^*} > 0$ for some u and thus is a contradiction to Condition 1.

ii) Assume $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} < 0$. We define $u^* = \underset{u \in [d]}{\operatorname{argmax}} \frac{\tau_u}{(e_u^\top \Sigma e_u)^{\frac{1}{2}}}$. By the third condition and the assumption $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} < 0$, we have

$$
0 > \frac{\boldsymbol{\beta}^{*\top} \boldsymbol{\tau}}{(\boldsymbol{\beta}^{*\top} \Sigma \boldsymbol{\beta}^*)^{\frac{1}{2}}} \geq \frac{\tau_{u^*}}{(e_{u^*}^\top \Sigma e_{u^*})^{\frac{1}{2}}}.
$$

This implies $\tau_{u^*} < 0$. We then get

$$\begin{aligned}
\frac{\xi^\top \tau}{(\xi^\top \Sigma \xi)^{\frac{1}{2}}} &= \sum_{u\in[d]} \xi_u \frac{\tau_u}{(\xi^\top \Sigma \xi)^{\frac{1}{2}}} = \sum_{u\in[d]} \xi_u \frac{\tau_u \, (e_u^\top \Sigma e_u)^{\frac{1}{2}}}{(\xi^\top \Sigma \xi)^{\frac{1}{2}} \, (e_u^\top \Sigma e_u)^{\frac{1}{2}}} \\
&\leq \sum_{u\in[d]} \xi_u \frac{\tau_{u^*} \, (e_u^\top \Sigma e_u)^{\frac{1}{2}}}{\left(e_{u^*}^\top \Sigma e_{u^*}\right)^{\frac{1}{2}} \, (\xi^\top \Sigma \xi)^{\frac{1}{2}}} \\
&= \frac{\tau_{u^*}}{\left(e_{u^*}^\top \Sigma e_{u^*}\right)^{\frac{1}{2}}} \frac{\sum_{u\in[d]} \xi_u (e_u^\top \Sigma e_u)^{\frac{1}{2}}}{(\xi^\top \Sigma \xi)^{\frac{1}{2}}} \\
&\leq \frac{\tau_{u^*}}{\left(e_{u^*}^\top \Sigma e_{u^*}\right)^{\frac{1}{2}}} \leq \frac{\beta^{*\top} \tau}{\left(\beta^{*\top} \Sigma \beta^*\right)^{\frac{1}{2}}},
\end{aligned}$$

where to arrive at the last line we used $\tau_{u^*} < 0$ and the triangle inequality $\sum_{u\in[d]} \xi_u (e_u^\top \Sigma e_u)^{\frac{1}{2}} = \sum_{u\in[d]} \xi_u \|\Sigma^{\frac{1}{2}} e_u\| \geq \|\sum_{u\in[d]} \xi_u \Sigma^{\frac{1}{2}} e_u\| = \|\Sigma^{\frac{1}{2}} \xi\| = (\xi^\top \Sigma \xi)^{\frac{1}{2}}$. Thus this violates the assumption $\frac{\xi^\top \tau}{(\xi^\top \Sigma \xi)^{\frac{1}{2}}} > \frac{\beta^{*\top} \tau}{\left(\beta^{*\top} \Sigma \beta^*\right)^{\frac{1}{2}}}$.

Note that the above inequalities also hold for $\frac{\beta^{*\top} \tau}{\left(\beta^{*\top} \Sigma \beta^*\right)^{\frac{1}{2}}}$. Thus we get that $\frac{\beta^{*\top} \tau}{\left(\beta^{*\top} \Sigma \beta^*\right)^{\frac{1}{2}}} = \frac{\tau_{u^*}}{\left(e_{u^*}^\top \Sigma e_{u^*}\right)^{\frac{1}{2}}}$. This implies that $l = |\mathcal{U}| = 1$. Thus the following statements hold true:

$$i) \quad \beta^{*\top} \tau < 0 \quad \Rightarrow \quad l = 1, \tag{A.8}$$

$$ii) \quad l \geq 2 \quad \Rightarrow \quad \beta^{*\top} \tau \geq 0. \tag{A.9}$$

□

A.1.2. Gradient of objective

We overload the notation and define $z := \tau - \Sigma\beta \frac{\beta^\top \tau}{\beta^\top \Sigma \beta}$ similar as in Equation A.1 but for any β. Then

$$\begin{aligned}
\nabla_\beta f(\beta) &= \nabla_\beta \left(\frac{\beta^\top \tau}{(\beta^\top \Sigma \beta)^{\frac{1}{2}}} \right) \\
&= \frac{(\beta^\top \Sigma \beta)^{\frac{1}{2}} \nabla_\beta (\beta^\top \tau) - \beta^\top \tau \nabla_\beta ((\beta^\top \Sigma \beta)^{\frac{1}{2}})}{\beta^\top \Sigma \beta} \\
&= \frac{(\beta^\top \Sigma \beta)^{\frac{1}{2}} \tau - \frac{1}{2} \beta^\top \tau ((\beta^\top \Sigma \beta)^{-\frac{1}{2}}) \cdot 2\beta^\top \Sigma}{\beta^\top \Sigma \beta} \\
&= \frac{1}{(\beta^\top \Sigma \beta)^{\frac{1}{2}}} \left(\tau - \Sigma\beta \left(\frac{\beta^\top \tau}{(\beta^\top \Sigma \beta)} \right) \right) \\
&= \frac{1}{(\beta^\top \Sigma \beta)^{\frac{1}{2}}} z.
\end{aligned} \tag{A.10}$$

A.1.3. Proof of Equation A.5

In the proof of Theorem 2.3.2 we used that $\tilde{X}$ and z are independent. Which we prove here. Since $\tilde{X}$ and z are jointly normal, we only need to

show that they are uncorrelated. To do so recall that we are only interested in the distribution under the null and hence $\mathbf{0} = \mathbb{E}[\boldsymbol{\tau}] = \mathbb{E}\left[\tilde{X}\right] = \mathbb{E}[\boldsymbol{z}]$. Since $\tilde{X}_u = 0$ for all $u \notin \mathcal{U}$ and $\boldsymbol{z}'_u = 0$ for all $u' \in \mathcal{U}$, it suffices to show that $\tilde{X}_j$ is uncorrelated with z_i for all $i \notin \mathcal{U}, j \in \mathcal{U}$.

$$\begin{aligned}
&\mathrm{Cov}\left[z_i, \tilde{X}_j\right] = \mathbb{E}\left[z_i \tilde{X}_j\right] = \mathbb{E}\left[\left(\tau_i - (\Sigma\bar{\Sigma}\boldsymbol{\tau})_i\right)\left((\bar{\Sigma})^{\frac{1}{2}}\boldsymbol{\tau}\right)_j\right] \\
&= \sum_{u\in\mathcal{U}}((\bar{\Sigma})^{\frac{1}{2}})_{ju}\mathbb{E}[\tau_i, \tau_u] - \sum_{s,t,u\in\mathcal{U}}((\bar{\Sigma})^{\frac{1}{2}})_{ju}\Sigma_{is}\bar{\Sigma}_{st}\mathbb{E}[\tau_t\tau_u] \\
&= \sum_{u\in\mathcal{U}}((\bar{\Sigma})^{\frac{1}{2}})_{ju}\Sigma_{iu} - \sum_{s,t,u\in\mathcal{U}}((\bar{\Sigma})^{\frac{1}{2}})_{ju}\Sigma_{is}\bar{\Sigma}_{st}\Sigma_{tu} \\
&= \left((\bar{\Sigma})^{\frac{1}{2}}\Sigma\right)_{ji} - \left(\Sigma\bar{\Sigma}\Sigma(\bar{\Sigma})^{\frac{1}{2}}\right)_{ij} \\
&= \left((\bar{\Sigma})^{\frac{1}{2}}\Sigma\right)_{ji} - \left(\Sigma(\bar{\Sigma})^{\frac{1}{2}}\right)_{ij} = 0.
\end{aligned}$$

Thus $\tilde{X}$ and $\boldsymbol{z}$ are uncorrelated and independent.

A.2. Solution of the continuous optimization problem

The presented solution is similarly described in Section 4 of [6]. There an $L1$ norm constraint was used, which, however does not change anything. For completeness we include it here. We define

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

$$f(\boldsymbol{\beta}) := \frac{\boldsymbol{\beta}^\top\boldsymbol{\tau}}{(\boldsymbol{\beta}^\top\Sigma\boldsymbol{\beta})^{\frac{1}{2}}},$$

and we want to find

$$\boldsymbol{\beta}^* = \underset{\boldsymbol{\beta}\geq\mathbf{0},\|\boldsymbol{\beta}\|=1}{\mathrm{argmax}} \frac{\boldsymbol{\beta}^\top\boldsymbol{\tau}}{(\boldsymbol{\beta}^\top\Sigma\boldsymbol{\beta})^{\frac{1}{2}}}.$$

Since f is a homogeneous function of order 0 in $\boldsymbol{\beta}$ we have $f(c\boldsymbol{\beta}) = f(\boldsymbol{\beta})$ for any $c > 0$. We can thus solve the relaxed problem (we implicitly exclude $\boldsymbol{\beta} = \mathbf{0}$)

$$\boldsymbol{\beta}' = \underset{\boldsymbol{\beta}\geq\mathbf{0}}{\mathrm{argmax}}\, f(\boldsymbol{\beta}).$$

The solution of the original problem is then simply given as a rescaled version of the relaxed problem $\boldsymbol{\beta}^* = \frac{\boldsymbol{\beta}'}{\|\boldsymbol{\beta}'\|}$. We shall solve the relaxed problem for two different cases.

i) $\exists u \in [d] : \tau_u \geq 0$.
In this case, we know that $\max_{\boldsymbol{\beta}\geq\mathbf{0}} f(\boldsymbol{\beta}) \geq 0$ and hence $\boldsymbol{\beta}' = \underset{\boldsymbol{\beta}\geq\mathbf{0}}{\mathrm{argmax}}\, f(\boldsymbol{\beta}) \Leftrightarrow \boldsymbol{\beta}' = \underset{\substack{\boldsymbol{\beta}\geq\mathbf{0} \\ f(\boldsymbol{\beta})\geq 0}}{\mathrm{argmax}}\, f^2(\boldsymbol{\beta})$. The set $S := \{\boldsymbol{\beta} \in \mathbb{R}^d | \boldsymbol{\beta} \geq \mathbf{0}, f(\boldsymbol{\beta}) \geq 0\}$ is convex and the functions $g_1(\boldsymbol{\beta}) := (\boldsymbol{\beta}^\top\boldsymbol{\tau})^2$ and $g_2(\boldsymbol{\beta}) := \boldsymbol{\beta}^\top\Sigma\boldsymbol{\beta}$ are convex (recall that Σ is a positive matrix). Thus

our problem becomes

$$\boldsymbol{\beta}' = \underset{\boldsymbol{\beta} \in S}{\operatorname{argmax}} \frac{g_1(\boldsymbol{\beta})}{g_2(\boldsymbol{\beta})},$$

which is a concave fractional program. In our implementation we solve it by fixing $\boldsymbol{\beta}^\top \boldsymbol{\tau} = a$ for some $a > 0$ and then minimizing the denominator. Thus we are solving the quadratic optimization problem

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta} \\ \text{subject to:} \quad & \boldsymbol{\beta} \geq \mathbf{0} \\ & \boldsymbol{\beta}^\top \boldsymbol{\tau} = a. \end{aligned}$$

We solve this problem with the CVXOPT python package [184].

[184]: Vandenberghe (2010), *The CVXOPT linear and quadratic cone program solvers*

ii) $\tau_u < 0 \, \forall u \in [d]$.
In this case we have $\boldsymbol{\beta}^{*\top} \boldsymbol{\tau} < 0$. By Equation A.9 we have $l = 1$. Thus we simply have $\boldsymbol{\beta}^* = e_{u^*}$, where $u^* = \underset{u \in [d]}{\operatorname{argmax}} \frac{\tau_u}{\Sigma_{u,u}}$.

Note that in the case $\boldsymbol{\tau} = \mathbf{0}$, $\boldsymbol{\beta}^*$ is not well defined and we could randomly select any $\boldsymbol{\beta}^*$. However, the probability of this happening is 0.

A.3. Other proofs

A.3.1. Proof of Corollary 2.3.1

As we pointed out in the Subsection 2.3.1, when selecting a test from a countable number of test that can be written as projections of the base tests $\boldsymbol{\tau}$ we can use the results of [17]. For completeness we explicitly include their relevant Theorem 5.2.

[17]: Lee et al. (2016), *Exact post-selection inference, with application to the lasso*

Theorem A.3.1 (Polyhedral Lemma) *Let $\boldsymbol{\tau} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, $\boldsymbol{\eta}, \boldsymbol{\mu} \in \mathbb{R}^d$, $\Sigma \in \mathbb{R}^{d \times d}$ positive definite, and $A \in \mathbb{R}^{s \times d}$, $\boldsymbol{b} \in \mathbb{R}^s$ for some $s \in \mathbb{N}$. Define $\boldsymbol{c} := \Sigma \boldsymbol{\eta} \left(\boldsymbol{\eta}^\top \Sigma \boldsymbol{\eta}\right)^{-1}$ and $\boldsymbol{z} := \left(I_d - \boldsymbol{c}\boldsymbol{\eta}^\top\right) \boldsymbol{\tau}$. Then we have*

$$\left[\boldsymbol{\eta}^\top \boldsymbol{\tau} | A\boldsymbol{\tau} \leq \boldsymbol{b}, \boldsymbol{z} = \hat{\boldsymbol{z}}\right] \overset{d}{=} TN\left(\boldsymbol{\eta}^\top \boldsymbol{\mu}, \boldsymbol{\eta}^\top \Sigma \boldsymbol{\eta}, \mathcal{V}^-(\hat{\boldsymbol{z}}), \mathcal{V}^+(\hat{\boldsymbol{z}})\right),$$

where $TN(\mu, \sigma^2, a, b)$ denotes a Gaussian distribution with mean μ and variance σ^2 that is truncated at a and b. Here

$$\mathcal{V}^-(\boldsymbol{z}) := \max_{j:(A\boldsymbol{c})_j < 0} \frac{\boldsymbol{b}_j - (A\boldsymbol{z})_j}{(A\boldsymbol{c})_j}, \quad \mathcal{V}^+(\boldsymbol{z}) := \min_{j:(A\boldsymbol{c})_j > 0} \frac{\boldsymbol{b}_j - (A\boldsymbol{z})_j}{(A\boldsymbol{c})_j}.$$

Note that $\boldsymbol{c}$ is simply a fixed vector. $\boldsymbol{z}$ is a random variable that can be shown to be independent of $\boldsymbol{\eta}^\top \boldsymbol{\tau}$. The result enables us to draw a realization $\hat{\boldsymbol{\tau}}$ of the random variable (RV) $\boldsymbol{\tau}$ and select $\boldsymbol{\eta}$ if $A\hat{\boldsymbol{\tau}} \leq \boldsymbol{b}$. Since the truncation points of the Gaussian only depend on $\hat{\boldsymbol{z}}$, and $\boldsymbol{z}$ is independent of $\boldsymbol{\eta}^\top \boldsymbol{\tau}$, we can compute a reliable p-value of $\boldsymbol{\eta}^\top \hat{\boldsymbol{\tau}}$ by using Theorem A.3.1.

Proof of Corollary 2.3.1. We need the distribution of $\frac{\tau_{u^*}}{\sigma_{u^*}}$ after conditioning on the selection of u^*. To obtain this distribution we first need to

characterize the event that leads to the selection of u^*. The selection event simply is $u^* = \operatorname{argmax}_{u\in[d]} \frac{\tau_u}{\sigma_u} \Leftrightarrow \frac{\tau_{u^*}}{\sigma_{u^*}} \geq \frac{\tau_u}{\sigma_u}$ for all $u \in [d]$. Therefore, define the matrix $A := \operatorname{diag}(\frac{1}{\sigma_1}, \dots, \frac{1}{\sigma_d}) - \frac{1}{\sigma_{u^*}} A(u^*)$, where $\operatorname{diag}(\cdot)$ defines a $d \times d$ matrix with the arguments on its diagonal and zeros everywhere else and $A(\cdot)$ is a $d \times d$ matrix with ones in the column given by its argument and zeros everywhere else. It follows that $(A\boldsymbol{\tau})_j = \frac{\tau_j}{\sigma_j} - \frac{\tau_{u^*}}{\sigma_{u^*}}$, and $u^* = \operatorname{argmax}_{u\in[d]} \frac{\tau_u}{\sigma_u}$ is equivalent to $A\boldsymbol{\tau} \leq \mathbf{0} =: \boldsymbol{b}$. Apart from this we define $\boldsymbol{\eta} := \frac{e_{u^*}}{\sigma_{u^*}}$, so that $\boldsymbol{\eta}^\top \boldsymbol{\tau} = \frac{\tau_{u^*}}{\sigma_{u^*}}$. Then we can define $\boldsymbol{c} := \Sigma\boldsymbol{\eta}\,(\boldsymbol{\eta}^\top \Sigma \boldsymbol{\eta})^{-1}$ and $\boldsymbol{z} := (I_d - \boldsymbol{c}\boldsymbol{\eta}^\top)\,\boldsymbol{\tau}$ as in Theorem A.3.1, and denote by $\hat{\boldsymbol{z}}$ the value of the random variable $\boldsymbol{z}$ that we observed (note that this coincides with the definition we used for $\boldsymbol{z}$ in the Corollary). By our definitions we have $(A\boldsymbol{c})_j = \frac{\Sigma_{ju^*}/\sigma_j - \sigma_{u^*}}{\sigma_{u^*}} = \frac{1}{\sigma_{u^*}\sigma_j}\left(\Sigma_{u^*j} - \sigma_u^*\sigma_j\right)$. Since Σ is positive definite, $(A\boldsymbol{c})_j < 0$ if $j \neq u^*$ and $(A\boldsymbol{c})_{u^*} = 0$. Thus according to Theorem A.3.1, $\mathcal{V}^+$ is an optimization over an empty set and we can set it to ∞. Further $(A\boldsymbol{z})_j = \frac{1}{\sigma_{u^*}\sigma_j}\left(\tau_j\sigma_{u^*} - \frac{\Sigma_{u^*j}}{\sigma_{u^*}}\tau_{u^*}\right)$. Combining the previous two expressions we obtain $\frac{-(A\boldsymbol{z})_j}{(A\boldsymbol{c})_j} = \frac{\tau_j\sigma_{u^*} - \frac{\Sigma_{u^*j}}{\sigma_{u^*}}\tau_{u^*}}{\sigma_u^*\sigma_j - \Sigma_{u^*j}} = \frac{\sigma_{u^*}z_j}{\sigma_u^*\sigma_j - \Sigma_{u^*j}}$. We can then directly apply Theorem A.3.1 and the result follows. □

A.3.2. Proof of Equation 2.3

In Section 2.3 we omitted the proof of the closed form solution of $\boldsymbol{\beta}^\infty$. We thus need to show

$$\operatorname*{argmax}_{\|\boldsymbol{\beta}\|=1} \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \frac{\Sigma^{-1}\boldsymbol{\mu}}{\|\Sigma^{-1}\boldsymbol{\mu}\|}.$$

Proof. We are only interested in $\boldsymbol{\beta}^\infty$ if the alternative hypothesis is true and thus at least one entry of $\boldsymbol{\mu}$ is positive. We further assume that the covariance Σ has full rank. Hence there exists a $b > 0$ such that $\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta} > b$ for all $\boldsymbol{\beta}$ with $\|\boldsymbol{\beta}\| = 1$, i.e., the denominator $(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}$ is strictly positive and has a lower bound. Since $\boldsymbol{\mu} \neq \mathbf{0}$, this implies that $\max_{\|\boldsymbol{\beta}\|=1} \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} > 0$. Also the nominator has an upper bound which is given by $\boldsymbol{\beta}^\top \boldsymbol{\mu} \leq \boldsymbol{\mu}^\top \boldsymbol{\mu}/\|\boldsymbol{\mu}\|$ if $\|\boldsymbol{\beta}\| = 1$. Hence the whole maximization is upper bounded. Since the unit sphere in $\mathbb{R}^d$ is a compact set, we can conclude that the maximum of the objective is attained. Thus it suffices to show that for all $\boldsymbol{\beta} \neq \boldsymbol{\beta}^\infty$ the objective is not maximized. In the following, we use that the objective of the maximization is a homogeneous function of order 0 in $\boldsymbol{\beta}$ and hence we can relax the constraint $\|\boldsymbol{\beta}\| = 1$ to $\boldsymbol{\beta} \neq \mathbf{0}$ (note that this not affect the existence of the maximum). As we showed in Appendix A.1.2, the gradient of the objective function is given by

$$\nabla_\beta \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \frac{1}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}\left(\boldsymbol{\mu} - \Sigma\boldsymbol{\beta}\left(\frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})}\right)\right).$$

Setting the gradient to zero we obtain

$$\nabla_\beta \frac{\boldsymbol{\beta}^\top \boldsymbol{\mu}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \mathbf{0} \Leftrightarrow \boldsymbol{\beta} = c \cdot \Sigma^{-1}\boldsymbol{\mu} \text{ for some } c \in \mathbb{R}.$$

If $c < 0$ the objective attains a negative value, since Σ^{-1} is a strictly positive matrix, and thus does not correspond to the global maximum, which we already know to be positive. Thus, the maximum has to be attained for some $c > 0$. Using the constraint $\|\boldsymbol{\beta}\| = 1$ it follows that the global optimum is attained at $\boldsymbol{\beta}^{\infty}$. □

A.4. Experimental details and further experiments

We first give some details on the experiments we showed in Section 2.5. For all the experiments we start with a set of d base kernels $\mathcal{K} = [k_1, \ldots, k_d]$ that are chosen independently of the observed data samples $\mathbb{X} = \{x_1, \ldots, x_{2n}\} \sim P^{2n}$ and $\mathbb{Y} = \{y_1, \ldots, y_{2n}\} \sim Q^{2n}$. First, we define $z_i := (x_i, x_{n+i}, y_i, y_{n+i})$ and compile $\mathbb{X}$ and $\mathbb{Y}$ into $= \{z_1, \ldots, z_n\}$. For each kernel we define $h_i(z) := h_i(x, x', y, y') := k_i(x, x') + k_i(y, y') - k_i(x, y') - k_i(y, x')$. For all the methods we estimate the covariance matrix on the whole dataset as

$$\hat{\Sigma}_{ij} = \frac{1}{n}\sum_{k=1}^{n} h_i(z_k)h_j(z_k) - \frac{1}{n}\sum_{k=1}^{n} h_i(z_k)\frac{1}{n}\sum_{k'=1}^{n} h_j(z_{k'}).$$

We then further assume that $\Sigma = \hat{\Sigma}$ which is justified since the CLT also works with a consistent estimate of the covariance. For all the methods that do not split the data (OST, Wald, and Naive) we estimate the entries of $\hat{\boldsymbol{\tau}}$ as

$$\hat{\tau}_i = \sqrt{n}\,\widehat{\mathrm{MMD}}^2_{\mathrm{lin}}(P, Q) = \sqrt{n}\,\frac{1}{n}\sum_{k=1}^{n} h_i(z_k),$$

i.e., we directly absorb the $\sqrt{n}$ dependence of the asymptotic distribution into $\boldsymbol{\tau}$. For data splitting we estimate $\hat{\boldsymbol{\tau}}_{\mathrm{tr}}$ on a split of the data and $\hat{\boldsymbol{\tau}}_{\mathrm{te}}$ on the other split. For example split0.3 means that 30% of the data are used to estimate $\hat{\boldsymbol{\tau}}_{\mathrm{tr}}$ and 70% used to estimate $\hat{\boldsymbol{\tau}}_{\mathrm{te}}$. We assume that the number of samples in the respective subsets are even and otherwise neglect some samples.

Methods We compare four different methods:

1. OST: The test we recommend to use, as described in Algorithm 1.
2. Wald: The Wald test, which does not take into account the prior information $\boldsymbol{\mu} \geq \mathbf{0}$.
3. split: Data splitting similar to the approach in [6]. split0.3 denotes that 30% of the data are used for learning $\boldsymbol{\beta}^*$ and 70% are used for testing. Here we first, learn $\boldsymbol{\beta}^*$ on the training sample, i.e., $\boldsymbol{\beta}^* = \underset{\|\Sigma\boldsymbol{\beta}\|=1, \Sigma\boldsymbol{\beta}\geq 0}{\mathrm{argmax}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}_{tr}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}}$. We then use the test statistic $\frac{\hat{\boldsymbol{\beta}}^\top \boldsymbol{\tau}_{te}}{(\hat{\boldsymbol{\beta}}^\top \Sigma \hat{\boldsymbol{\beta}})^{\frac{1}{2}}}$, which follows a standard normal under the null. This differs from the approach in [6], since we optimize with the constraints $\Sigma\boldsymbol{\beta} \geq \mathbf{0}$, whereas [6] suggested a simple positivity constraint $\boldsymbol{\beta} \geq \mathbf{0}$. We discuss this in Appendix A.4.2.
4. naive: Two stage procedure where all the data is used for learning and testing without correcting for the dependency, i.e., without

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

splitting the data. Thus the test statistic is the same as for OST, but we work with the wrong null distribution, i.e., the one that is only valid for data splitting. This approach is not a well-calibrated test, see Figure A.5 and hence is useless.

Datasets The `DIFF VAR` dataset is a simple one-dimensional toy dataset, where $P = \mathcal{N}(0, 1)$ and $Q = \mathcal{N}(0, 1.5)$.

The `Blobs` dataset was constructed using a mixture of 2D Gaussians on a 3×3 grid. The centers of the Gaussians are set to $\mu_1, \ldots, \mu_9 = (0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (2,0), (2,1), (2,2)$ and the covariances are $\Sigma_P = \text{diag}(0.1, 0.3)$ and $\Sigma_Q = \text{diag}(0.3, 0.1)$. Samples from P and Q are shown in Figure A.2. The `Blobs` dataset is constructed such that the main variance in the data does not reflect the difference between P and Q, which happens on a smaller length scale. This is inspired by [6], where similar data has been considered to showcase that such problems benefit from careful kernel choice. We can reproduce this behavior with our results, which show that for this dataset the performance is bad if one only considers the median heuristic Gaussian kernel together with a linear kernel.

The `MNIST` dataset was constructed by first downsampling all the images to 7×7 pixels (originally 28×28), by simply averaging over fields of 4×4 pixels. We define P to contain all the digits, while Q only contains uneven digits. For our experiments we draw with replacement from the images in the database. Some samples from both distributions are shown in Figure A.3.

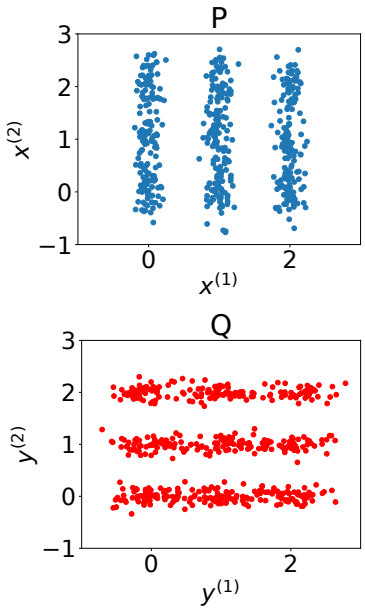


Figure A.2.: Samples from Blobs dataset.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

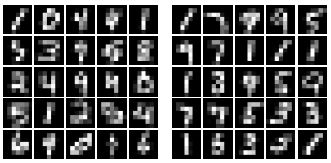


Figure A.3.: Samples from downsampled MNIST dataset. P (left) contains all digits, while Q (right) only contains uneven digits.

Experiments for Figure 2.3 For Figure 2.3 we constructed a 1-D data set such that both P and Q are symmetric (thus all uneven moments vanish) and have the same variance, see Figure A.4.

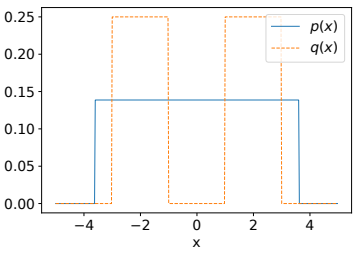


Figure A.4.: Probability density functions used for the experiment in Figure 2.3 of the main paper. Both distributions are symmetric and are constructed to have the same variance.

[3]: LeCun et al. (2010), *MNIST handwritten digit database*

A.4.1. Type-I errors

To verify which methods are theoretically justified, i.e., control the Type-I error at a level $\alpha = 0.05$, we run the following experiments, similar to the experiments in the main paper, where $P = Q$.

1. DIFF VAR ($p = 1$): $P = \mathcal{N}(0, 1)$ and $Q = \mathcal{N}(0, 1)$.
2. MNIST ($p = 49$): We consider downsampled 7x7 images of the MNIST dataset [3], where P contains all the digits and $Q = P$.
3. Blobs ($p = 2$): A mixture of anisotropic Gaussians and $P = Q$.

The results are in Figure A.5. All the methods except NAIVE correctly control the Type-I error at a rate $\alpha = 0.05$ even for relatively small sample sizes. Note that all the described approaches rely on the asymptotic distribution. The critical sample size, at which it is safe to use, generally depends on the distributions P and Q and also the kernel functions. A good approach to simulating Type-I errors in in two-sample testing problems is to merge the samples and then randomly split them again. If the estimated Type-I error is significantly larger that α, working with the asymptotic distribution is not reliable.

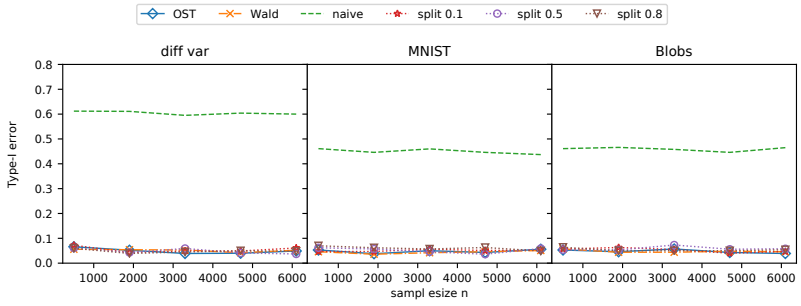


Figure A.5.: Type-I errors for similar distributions as the one considered in the main paper. To simulate type-I errors we choose distributions $P = Q$ that are similar to the ones considered for the Type-II errors. We see that all well-calibrated methods reliably control the Type-I error at a rate $\alpha = 0.05$, and conclude that working with the asymptotic distributions is well justified for the considered examples. The NAIVE approach fails to control the error, as it overfits in the training phase without a correction in the testing phase.

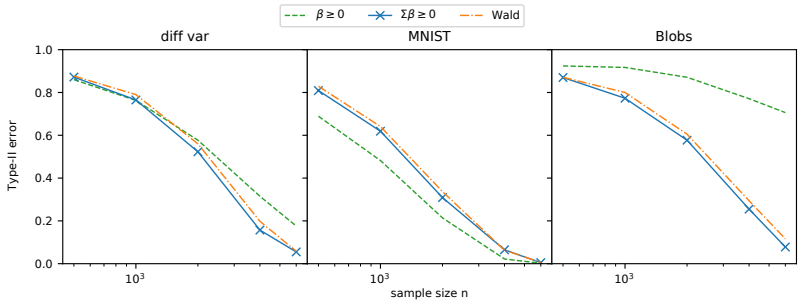


Figure A.6.: Comparison of the different constraints: In the main paper we argue that OST is a principled approach to constraint the class of considered tests, when $\mu \geq 0$ is guaranteed. [6] suggested a different constraint $\beta \geq 0$. With Theorem 2.3.2, we can also work with these constraints without data-splitting. The results suggest that indeed OST is a meaningful way to constrain the class of tests, as it consistently outperforms the Wald test. On the other hand the constraint suggested by [6], can only be seen as a heuristic. For some cases it performs better than the Wald test and the OST, but it can also perform worse.

A.4.2. Comparison of the constraints

In Subsection 2.3.2 we motivate to constrain the set of considered β to obey $\Sigma\beta \geq 0$, thus incorporating the knowledge $\mu \geq 0$. All our experiments suggest that this constraint indeed improves test power as compared to the general Wald test. In [6] a different constraint was chosen. There β is constrained to be positive, i.e., $\beta \geq 0$. The motivation for their constraint is that the sum of positive definite (pd) kernel functions is again a pd kernel function [4]. Thus, by constraining $\beta \geq 0$ one ensures that $k = \sum_{u=1}^{d} \beta_u k_u$ is also a pd kernel. While this is sensible from a kernel perspective, it is unclear whether this is smart from a hypothesis testing viewpoint. From the latter perspective we do not necessarily care whether or not β^* defines a pd kernel. Our approach instead was purely motivated to increase test power over the Wald test. In Figure A.6 we thus compare the two different constraints to the Wald test on the examples that were also investigated in the main paper with $d = 6$ kernels (again five Gaussian kernels and a linear kernel).

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

From Figure A.6 we observe that the positivity constraint of [6] does not allow for general conclusions. Depending on the problem, the positivity constraint can both lead to higher or lower test power than the Wald test or tests with the constraint $\Sigma\beta \geq 0$. It will thus generally depend on the problem at hand which constraint is better. However, at least the approach we recommend ($\Sigma\beta \geq 0$) seems to guarantee a test power at least as high as the Wald test, whereas the positivity constraint can also be worse. As long as one has not a clear indication that the positivity constraint leads to better performance, we thus recommend the constraint $\Sigma\beta \geq 0$.

[6]: Gretton et al. (2012), *Optimal kernel choice for large-scale two-sample tests*

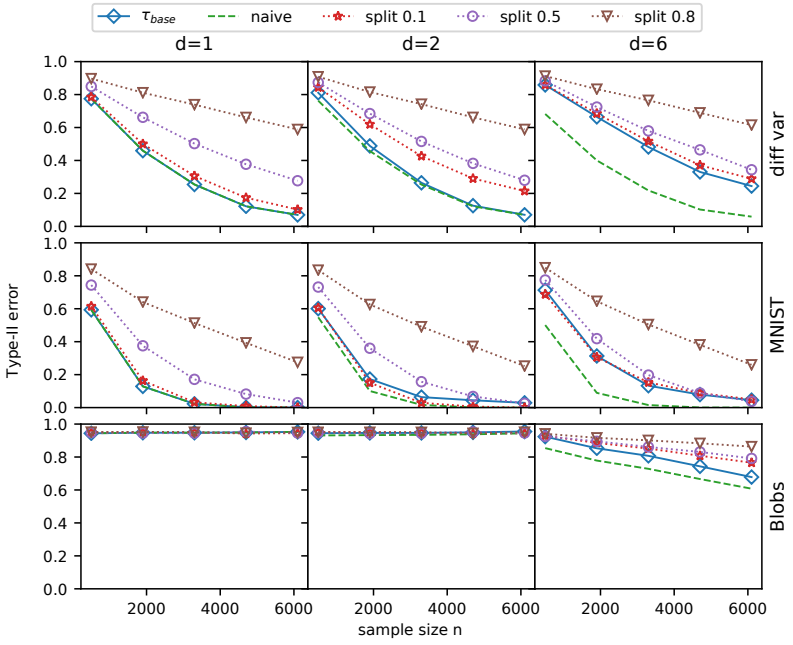


Figure A.7.: Type-II errors for discrete selection, i.e., the class of considered tests is T_{base}. The rows (columns) correspond to different datasets (sets of base kernels). Similar as in Figure 2.2, our approach τ_{base} outperforms the splitting approaches in most cases. However, for the `MNIST` dataset and $d = 2$ we see that the splitting approach with 10% training and 90% testing data (SPLIT0.1) performs better.

A.4.3. Discrete selection from T_{base}

In this experiment, we use the same datasets and base kernels as for the experiment in Section 2.5. Instead of considering T_{Wald} and T_{OST}, we consider T_{base}. We thus only compare to a data-splitting approach where also one of the base test statistics is selected. For completeness, we also include the NAIVE approach, which again overfits for $d > 1$. Note that the thresholds for τ_{base} can be computed with Corollary 2.3.1 and do not rely on Theorem 2.3.2. The results are shown in Figure A.7, again averaged over 5000 independent trials. In most of the cases, we observe that τ_{base} outperforms the data-splitting approaches. However, for the MNIST dataset and $d = 2$, the splitting approach that uses 10% for learning and 90% for testing does perform slightly better. Our attempt to explain this behavior lies in the truncation $\mathcal{V}^-$ of the conditional distribution. While for OST, we can show that $\mathcal{V}^- \leq 0$ (see proof of Theorem 2.3.2), for Corollary 2.3.1, $\mathcal{V}^-$ cannot be bounded. If $\mathcal{V}^-$ is very large, the selected test is very conservative. We acknowledge that this is not a sufficient analysis of this phenomenon, but leave a more theoretical treatment for future work.

A.5. Singular covariance matrices

In Section 2.3 we assumed that Σ is strictly positive, i.e., non-singular. However, in practice, some eigenvalues of the covariance matrix can be sufficiently close to zero to cause numerical problems. In the case of the kernel two-sample test, this can happen if we consider kernels that are too similar and thus cause redundancy in our observations. In practice, this happens for example if we consider Gaussian kernels with too similar bandwidths on an easy problem.

Note on regularization: One strategy to recover the numerical stability of the algorithm is to regularize the covariance matrix $\Sigma \to \Sigma + \lambda I$.

Doing this indeed increases the numerical stability, since it leads to a well-behaved condition number. However, it also makes the whole approach more conservative, since the (artificially) increased variance decreases the value of the test statistic compared to the threshold. This leads to an increase of Type-II error and thus a loss of power. To evade this, we suggest the more elaborate strategy below.

Since Σ is symmetric, there exists an orthonormal basis $\{v_i\}_{i\in[d]}$ and non-negative numbers $\{\lambda_i\}_{i\in[d]}$ such that

$$\Sigma = \sum_{i=1}^{d} \lambda_i v_i v_i^\top.$$

If Σ is singular, we can assume without loss of generality that there exists $d_0 \in [d]$ such that $\lambda_i = 0$ if $i \leq d_0$ and hence

$$\Sigma = \sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top.$$

Now if $v_i^\top \boldsymbol{\tau} \neq 0$ for some $i \in [d_0]$, we immediately know that $\boldsymbol{\mu} \neq \mathbf{0}$ and could reject. In other words the signal-to-noise ratio along this direction is infinite. Thus, in the following we assume $v_i^\top \boldsymbol{\tau} = 0$ for all $i \in [d_0]$, and hence, $\sum_{i=d_0+1}^{d} v_i v_i^\top \boldsymbol{\tau} = \boldsymbol{\tau}$. We can then rewrite the objective as follows

$$\max_{\Sigma\boldsymbol{\beta}\geq\mathbf{0}} \frac{\boldsymbol{\beta}^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma \boldsymbol{\beta})^{\frac{1}{2}}} = \max_{\sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top \boldsymbol{\beta}\geq\mathbf{0}} \frac{\boldsymbol{\beta}^\top \sum_{i=d_0+1}^{d} v_i v_i^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top \boldsymbol{\beta})^{\frac{1}{2}}}.$$

Now define $\boldsymbol{\alpha} := \sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top \boldsymbol{\beta}$. Since Σ is symmetric its pseudoinverse is given as $\Sigma^+ = \sum_{i=d_0+1}^{d} \frac{1}{\lambda_i} v_i v_i^\top$ and we get

$$\max_{\sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top \boldsymbol{\beta}\geq\mathbf{0}} \frac{\boldsymbol{\beta}^\top \sum_{i=d_0+1}^{d} v_i v_i^\top \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \sum_{i=d_0+1}^{d} \lambda_i v_i v_i^\top \boldsymbol{\beta})^{\frac{1}{2}}} = \max_{\boldsymbol{\alpha}\geq\mathbf{0}} \frac{\boldsymbol{\alpha}^\top \Sigma^+ \boldsymbol{\tau}}{(\boldsymbol{\beta}^\top \Sigma^+ \boldsymbol{\beta})^{\frac{1}{2}}}.$$

Similar as in Remark 2.3.1 we can define $\boldsymbol{\rho} := \Sigma^+ \boldsymbol{\tau}$ and $\Sigma' = \Sigma^+$. However, in Theorem 2.3.2 we assumed that the covariance is not singular. Therefore in Theorem 2.3.2 we used $l = |\mathcal{U}|$, which corresponded to the rank of $\Pi\Sigma\Pi$ (see Appendix A.1). However, in the present case the rank of $\Pi\Sigma^+\Pi$ does not equal the number of non-zero entries of $\boldsymbol{\beta}$. Therefore we use $l = \operatorname{rank}(\Pi\Sigma^+\Pi)$. With this we can apply Theorem 2.3.2 and get the conditional distribution under the null.

In practice, we have to treat the covariance matrix as singular if its condition number is below some threshold, as otherwise the numerical precision does not suffice to invert matrices faithfully.

Appendix of Chapter 3 B.

B.1. Proofs

B.1.1. Proof of Theorem 3.3.1

Proof. Theorem 3.3.1 follows by the application of the CLT; see, e.g., Theorem A, Chapter 1.9.1 in [32]. The CLT implies $\sqrt{n+m}(\hat{h}_P^n - \bar{h}_P) = \sqrt{n/c}(\hat{h}_P^n - \bar{h}_P) \xrightarrow{d} \mathcal{N}(0, \sigma_P^2/c)$, analogously for Q and the variances add up. Since $\hat{\sigma}_c^2(h) \xrightarrow{p} \sigma_c := \sigma_P^2/c + \sigma_Q^2/(1-c)$, the result follows from Slutsky's theorem. □

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

B.1.2. Proof of Proposition 3.3.2

Proof. Since we assume $\sigma_c(h) > 0$, it follows that

$$\lim_{n_{\text{te}}+m_{\text{te}}\to\infty} \Phi\left(\Phi^{-1}(1-\alpha) - \sqrt{n_{\text{te}}+m_{\text{te}}}\frac{\bar{h}_P - \bar{h}_Q}{\sigma_c(h)}\right) = 0, \tag{B.1}$$

i.e., the asymptotic rate of type-II errors goes to zero, if and only if $\bar{h}_P > \bar{h}_Q$. □

B.1.3. Derivation of Equation 3.13

We use the following definitions: Let $Z = \{x_1, \ldots, x_{n_{\text{tr}}}, y_1, \ldots, y_{m_{\text{tr}}}\}$ denote the pooled training sample and K denote the kernel matrix such that $K_{ij} = k(z_i, z_j)$ for $i, j \in [n_{\text{tr}} + m_{\text{tr}}]$. Let us define $G \in \mathcal{H}^{n_{\text{tr}}+m_{\text{tr}}}$ such that $G_i = k(z_i, \cdot)$. And we write $K = G^\top G$. Further we define $v_1 = (\frac{1}{n_{\text{tr}}}, \ldots, \frac{1}{n_{\text{tr}}}, 0, \ldots, 0)^\top \in \mathbb{R}^{n_{\text{tr}}+m_{\text{tr}}}$, $v_2 = (0, \ldots, 0, \frac{1}{m_{\text{tr}}}, \ldots, \frac{1}{m_{\text{tr}}})^\top \in \mathbb{R}^{n_{\text{tr}}+m_{\text{tr}}}$, and $\delta = v_1 - v_2$. For $l = n_{\text{tr}}, m_{\text{tr}}$ we define the idempotent centering operator $P_l = I_l - l^{-1}\mathbf{1}_l\mathbf{1}_l^\top$, where I denotes the identity operator and $\mathbf{1}_l$ the l dimensional vector with all ones. With this we define the $(n_{\text{tr}}+m_{\text{tr}})\times(n_{\text{tr}}+m_{\text{tr}})$ matrix $N_c = \begin{pmatrix} \frac{1}{c}P_{n_{\text{tr}}} & 0 \\ 0 & \frac{1}{1-c}P_{m_{\text{tr}}} \end{pmatrix}$. With the preceding definitions, we obtain $\hat{\mu}_P - \hat{\mu}_Q = G\delta$, $\hat{\Sigma} = \frac{1}{n_{\text{tr}}+m_{\text{tr}}} G N_c G^\top$.

Starting from Equation 3.11 we estimate the KFDA witness based on the empirical estimates of μ_P, μ_Q, Σ, i.e.,

$$\hat{h}_\lambda = \operatorname*{argmax}_{f\in\mathcal{H}} \frac{\langle \mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}, f\rangle}{\langle f, (\hat{\Sigma} + \lambda I) f\rangle^{\frac{1}{2}}}. \tag{B.2}$$

We first show a *representer Theorem* for KFDA [67, Sec. 3.4.3]. Therefore, we decompose possible candidate functions $f = f_1 + f_2 \in \mathcal{H}$ into a part f_1 that lies in the span of the training data $S_{\text{tr}} = \text{span}(\{k(z_i, \cdot) | i \in [n_{\text{tr}}+m_{\text{tr}}]\})$ and f_2 which lies in the span's orthogonal complement. Thus, by definition, we have $\langle f_2, k(z_i, \cdot)\rangle = 0$ for all $i \in [n_{\text{tr}} + m_{\text{tr}}]$. Since $\mu_{\mathbb{X}_{\text{tr}}}$ and $\mu_{\mathbb{Y}_{\text{tr}}}$ are within S_{tr}, we have $\langle \mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}, f\rangle = \langle \mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}, f_1\rangle$. Similarly, since $\hat{\Sigma}$ is only defined via the training samples in Z, $\hat{\Sigma}$ maps functions from S_{tr}

[67]: Mika (2003), *Kernel Fisher Discriminants*

to S_{tr} and we have $\hat{\Sigma} f_2 = 0$. Thus for the denominator of Equation B.2 we get

$$\langle f, (\hat{\Sigma} + \lambda I) f \rangle = \langle f_1, (\hat{\Sigma} + \lambda I) f_1 \rangle + \lambda \|f_2\|^2 \geq \langle f_1, (\hat{\Sigma} + \lambda I) f_1 \rangle . \quad \text{(B.3)}$$

We have shown that the nominator of Equation B.2 stays constant, if we add a function f_2 that is not is not in S_{tr} and the denominator can only grow. This implies that the maximum in Equation B.2 is attained for a function in S_{tr} and we can expand it as $\hat{h}_\lambda(\cdot) = \sum_{i=1}^{n_{\text{tr}}+m_{\text{tr}}} \hat{\alpha}_i k(z_i, \cdot)$. Hence the solution is

$$\hat{\alpha} = \underset{\alpha \in \mathbb{R}^{n_{\text{tr}}+m_{\text{tr}}}}{\operatorname{argmax}} \frac{\langle \mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}, \sum_{i=1}^{n_{\text{tr}}+m_{\text{tr}}} \alpha_i k(z_i, \cdot) \rangle}{\langle \sum_{i=1}^{n_{\text{tr}}+m_{\text{tr}}} \alpha_i k(z_i, \cdot), (\hat{\Sigma} + \lambda I) \sum_{i=1}^{n_{\text{tr}}+m_{\text{tr}}} \alpha_i k(z_i, \cdot) \rangle^{\frac{1}{2}}} \quad \text{(B.4)}$$

$$= \underset{\alpha \in \mathbb{R}^{n_{\text{tr}}+m_{\text{tr}}}}{\operatorname{argmax}} \frac{\delta^\top K \alpha}{\left(\alpha^\top \left(\frac{K N_c K}{n_{\text{tr}}+m_{\text{tr}}} + \lambda K \right) \alpha \right)^{\frac{1}{2}}}. \quad \text{(B.5)}$$

The solution to this is [67, Sec. 3.2][1]

$$\left(\frac{K N_c K}{n_{\text{tr}}+m_{\text{tr}}} + \lambda K \right) \hat{\alpha} = K\delta \qquad \Longleftrightarrow \qquad \hat{\alpha} = \left(\frac{K N_c K}{n_{\text{tr}}+m_{\text{tr}}} + \lambda K \right)^{-1} K\delta.$$

[67]: Mika (2003), *Kernel Fisher Discriminants*

1: For a sanity check, simply compute the gradient of Equation B.4 and set it to zero.

B.1.4. Convergence of $\hat{h}_\lambda$

We will show that $\hat{h}_\lambda \to h_\lambda = (\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q)$ in probability.

Proof. First, we observe that

$$\begin{aligned}
\hat{h}_\lambda - h_\lambda &= (\hat{\Sigma} + \lambda I)^{-1}(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q) \\
&= (\hat{\Sigma} + \lambda I)^{-1}(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\hat{\Sigma} + \lambda I)^{-1}(\mu_P - \mu_Q) \\
&\quad + (\hat{\Sigma} + \lambda I)^{-1}(\mu_P - \mu_Q) - (\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q) \\
&= (\hat{\Sigma} + \lambda I)^{-1} \left[(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\mu_P - \mu_Q) \right] \\
&\quad + \left[(\hat{\Sigma} + \lambda I)^{-1} - (\Sigma + \lambda I)^{-1} \right] (\mu_P - \mu_Q).
\end{aligned}$$

Thus it follows that

$$\begin{aligned}
\|\hat{h}_\lambda - h_\lambda\|_{\mathcal{H}} &\leq \|(\hat{\Sigma} + \lambda I)^{-1}[(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\mu_P - \mu_Q)]\|_{\mathcal{H}} \\
&\quad + \|[(\hat{\Sigma} + \lambda I)^{-1} - (\Sigma + \lambda I)^{-1}](\mu_P - \mu_Q)\|_{\mathcal{H}} \\
&= (A) + (B).
\end{aligned}$$

Probabilistic bound on (A). By the triangle inequality,

$$\begin{aligned}
&\|(\hat{\Sigma} + \lambda I)^{-1}[(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\mu_P - \mu_Q)]\|_{\mathcal{H}} \\
&\leq \|(\hat{\Sigma} + \lambda I)^{-1}\| \|(\mu_{\mathbb{X}_{\text{tr}}} - \mu_{\mathbb{Y}_{\text{tr}}}) - (\mu_P - \mu_Q)\|_{\mathcal{H}} \\
&\leq \|(\hat{\Sigma} + \lambda I)^{-1}\| (\|\mu_{\mathbb{X}_{\text{tr}}} - \mu_P\|_{\mathcal{H}} + \|\mu_Q - \mu_{\mathbb{Y}_{\text{tr}}}\|_{\mathcal{H}}).
\end{aligned}$$

By the spectral theorem, $\|(\hat{\Sigma} + \lambda I)^{-1}\| = \sup_{\hat{l} \in (\hat{l}_k)_{k=1}^{\infty}} \frac{1}{\hat{l}+\lambda} \leq 1/\lambda$ where $(\hat{l}_k)_{k=0}^{\infty}$ are the eigenvalues of $\hat{\Sigma}$ and by definition non-negative. Then, the $\sqrt{n}$-convergence of (A) follows from the $\sqrt{n}$-convergence of the

kernel mean embeddings $\|\mu_{\mathbb{X}_{\text{tr}}} - \mu_P\|_{\mathcal{H}} = \mathcal{O}_p(n_{\text{tr}}^{-1/2})$ and $\left\|\mu_Q - \mu_{\mathbb{Y}_{\text{tr}}}\right\|_{\mathcal{H}} = \mathcal{O}_p(m_{\text{tr}}^{-1/2})$; see, e.g., [19, Theorem 3.4]. That is, $(A) = \mathcal{O}_p(\min(n_{\text{tr}}, m_{\text{tr}})^{-1/2})$.

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

Probabilistic bound on (B). Using the identity $C^{-1} - D^{-1} = C^{-1}(D - C)D^{-1}$, we can rewrite (B) as

$$\begin{aligned}
&\|[(\hat{\Sigma} + \lambda I)^{-1} - (\Sigma + \lambda I)^{-1}](\mu_P - \mu_Q)\|_{\mathcal{H}} \\
&\quad = \|(\hat{\Sigma} + \lambda I)^{-1}(\hat{\Sigma} - \Sigma)(\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q)\|_{\mathcal{H}} \\
&\quad \leq \|(\hat{\Sigma} + \lambda I)^{-1}\|\|\hat{\Sigma} - \Sigma\|\|(\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q)\|_{\mathcal{H}} \\
&\quad \leq \|(\hat{\Sigma} + \lambda I)^{-1}\|\|\hat{\Sigma} - \Sigma\|_{\text{HS}}\|(\Sigma + \lambda I)^{-1}(\mu_P - \mu_Q)\|_{\mathcal{H}},
\end{aligned}$$

where we used that the operator norm is upper bounded by the Hilbert-Schmidt norm. Let $n := n_{\text{tr}} + m_{\text{tr}}$. Then, since $\|(\hat{\Sigma} + \lambda I)^{-1}\| \leq 1/\lambda$, the $\sqrt{n}$-convergence of (B) follows from the $\sqrt{n}$-convergence of the covariance operator, i.e., $\|\hat{\Sigma} - \Sigma\|_{\text{HS}} = \mathcal{O}_p(n^{-1/2})$ [185, Lemma 4]. That is, $(B) = \mathcal{O}_p((n_{\text{tr}} + m_{\text{tr}})^{-1/2})$.

[185]: Fukumizu et al. (2005), *Statistical Convergence of Kernel CCA*

Combining the rates of (A) and (B) yields the overall rate of convergence: $\|\hat{h}_\lambda - h_\lambda\|_{\mathcal{H}} = \mathcal{O}_p(\min(n_{\text{tr}}, m_{\text{tr}})^{-1/2})$. □

B.1.5. Witness objective vs. kernel optimization objective in MMD tests

In MMD-based two sample tests, the most common estimate of the MMD is the U-statistic estimate, defined as [5]

$$\widehat{\text{MMD}_u^2} = \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}, \tag{B.6}$$

[5]: Gretton et al. (2012), *A kernel two-sample test*

with $H_{ij} = \langle k(x_i, \cdot) - k(y_i, \cdot), k(x_j, \cdot) - k(y_j, \cdot)\rangle$. The objective function used in [7, 8] bases on the asymptotic variance of the estimator under the alternative hypothesis. If the population value of MMD^2 is positive, then the distribution of the estimate is asymptotically normal [32, Section 5.5.1], $\sqrt{n}\left(\widehat{\text{MMD}_u^2} - \text{MMD}^2\right) \xrightarrow{d} \mathcal{N}(0, \sigma_{H_1}^2)$, with $\sigma_{H_1}^2 = 4(\mathbb{E}\left[H_{12}H_{13}\right] - \mathbb{E}\left[H_{12}\right]^2)$ [8]. This can be used to derive an asymptotic test power criterion, which is given as the signal-to-noise ratio $J = \frac{\text{MMD}^2}{\sigma_{H_1}}$ [7, Sec. 2.1].

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*

We show, that the power criterion $J = \frac{\text{MMD}^2}{\sigma_{H_1}}$ corresponds to the SNR criterion we derived in Equation 3.8. It is an easy exercise to show that

$$\begin{aligned}
\sigma_{H_1}^2 = 4\Big(&\mathbb{E}_{X\sim P}\left[\langle \mu_P - \mu_Q, k(X, \cdot)\rangle^2\right] + \mathbb{E}_{Y\sim Q}\left[\langle \mu_P - \mu_Q, k(Y, \cdot)\rangle^2\right] \\
&- \langle \mu_P - \mu_Q, \mu_P\rangle^2 - \langle \mu_P - \mu_Q, \mu_Q\rangle^2\Big).
\end{aligned}$$

Recalling the definition of the covariance operator

$$\Sigma_P = \mathbb{E}\left[k(X, \cdot) \otimes k(X, \cdot)\right] - \mu_P \otimes \mu_P,$$

we obtain

$$\begin{aligned}\sigma_{H_1}^2 &= 4\left\langle \mu_P - \mu_Q, (\Sigma_P + \Sigma_Q)(\mu_P - \mu_Q)\right\rangle \\ &= 2\left\langle \mu_P - \mu_Q, (2\Sigma_P + 2\Sigma_Q)(\mu_P - \mu_Q)\right\rangle \\ &= 2\left\langle \mu_P - \mu_Q, \Sigma(\mu_P - \mu_Q)\right\rangle,\end{aligned}$$

where we used $\Sigma = \Sigma_P/c + \Sigma_Q/(1-c)$ and $c = 1/2$ for balanced samples.

Using $h_k^{P,Q} = \mu_P - \mu_Q$, we have

$$J(P,Q|k) = \frac{\mathrm{MMD}^2}{\sigma_{H_1}} = \frac{\langle \mu_P - \mu_Q, \mu_P - \mu_Q\rangle}{\sqrt{2}\left\langle \mu_P - \mu_Q, \Sigma(\mu_P - \mu_Q)\right\rangle^{\frac{1}{2}}} \tag{B.7}$$

$$= \frac{\langle \mu_P - \mu_Q, h_k^{P,Q}\rangle}{\sqrt{2}\left\langle h_k^{P,Q}, \Sigma h_k^{P,Q}\right\rangle^{\frac{1}{2}}} \tag{B.8}$$

$$= \frac{1}{\sqrt{2}}\mathrm{SNR}(h_k^{P,Q}). \tag{B.9}$$

B.1.6. MMD of nonparametrically optimized kernel corresponds to KFDA

Consider a fixed kernel k and denote by $\mathcal{A}$ the set of bounded positive operators on $\mathcal{H}_k$. For the nonparametric class of kernels $\mathcal{K} = \{k_A | k_A(x,y) = \left\langle Ak(x,\cdot), Ak(y,\cdot)\right\rangle, A \in \mathcal{A}\}$ using 'opt-mmd-witness' leads to exactly the same witness function as using 'kfda-witness'.

Proof. Writing inner products in the original RKHS with kernel k for kernel k_A we have the regularized J criterion

$$J_A^\lambda = \frac{\langle A(\mu_P - \mu_Q), A(\mu_P - \mu_Q)\rangle}{\langle A(\mu_P - \mu_Q), A(\Sigma + \lambda I)AA(\mu_P - \mu_Q)\rangle^{\frac{1}{2}}}.$$

We define $\delta_A := A^2(\mu_P - \mu_Q)$ and obtain

$$J_A^\lambda = \frac{\langle \mu_P - \mu_Q, \delta_A\rangle)}{\langle \delta_A, (\Sigma + \lambda I)\delta_A\rangle^{\frac{1}{2}}}, \tag{B.10}$$

which looks almost like Equation 3.11. The solution to Equation 3.11 is Equation 3.12 which implies that $\tilde{A}_\lambda = (\Sigma + \lambda I)^{-\frac{1}{2}}$ defines the optimal kernel

$$\begin{aligned}\tilde{k}_\lambda(x,x') &:= \langle (\Sigma + \lambda 1)^{-\frac{1}{2}}k(x,\cdot), (\Sigma + \lambda 1)^{-\frac{1}{2}}k(x',\cdot)\rangle_{\mathcal{H}} \\ &= \left\langle k(x,\cdot), (\Sigma + \lambda 1)^{-1}k(x',\cdot)\right\rangle_{\mathcal{H}}.\end{aligned}$$

Based on the empirical estimates the MMD witness of the optimized kernel would be (expressed in terms of the original kernel k)

$$h_{\tilde{k}_\lambda}^{\mathbb{Z}_{\mathrm{tr}}} = (\hat{\Sigma} + \lambda 1)^{-1}(\mu_{\mathbb{X}_{\mathrm{tr}}} - \mu_{\mathbb{Y}_{\mathrm{tr}}}) = \hat{h}_\lambda, \tag{B.11}$$

i.e., the witness of 'opt-mmd-witness' coincides with the 'kfda-witness' in the original RKHS. □

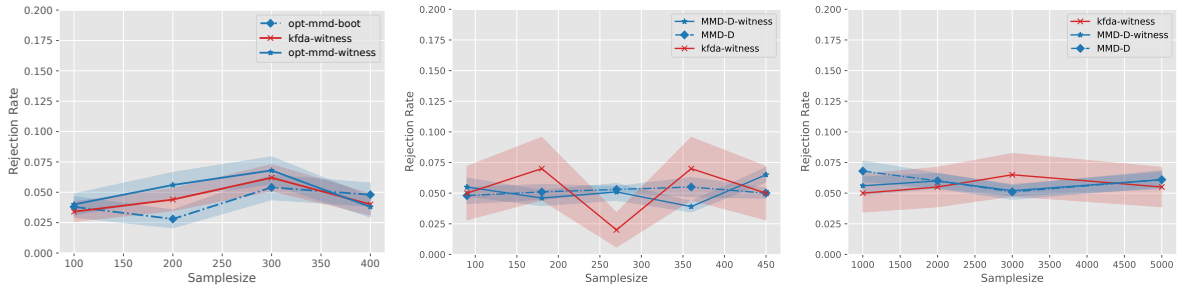


Figure B.1.: Rejection Rates for true null hypothesis (TypeII error) at $\alpha = 0.05$. **Left:** Standard Blobs dataset (500 iterations). **Middle:** Blobs dataset of [8], 'kfda-witness' is only average over 100 trials the others over 10×100, therefore 'kfda-witness' has higher variance. **Right:** Higgs dataset

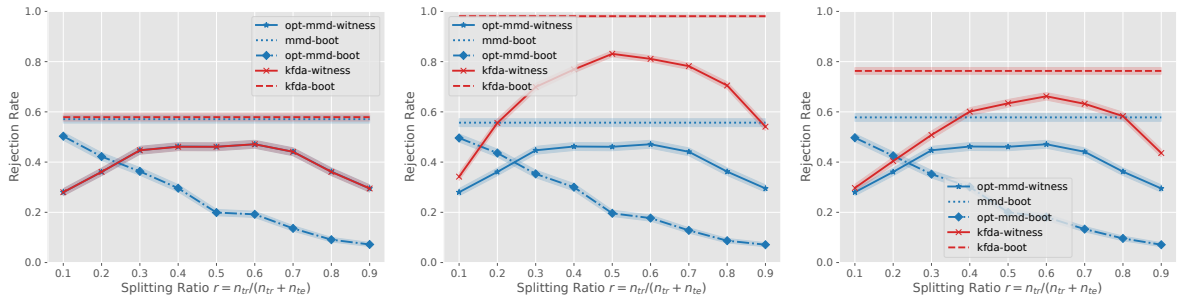


Figure B.2.: Effect of regularization on KFDA. We consider the same setting as in the left panel of Figure 3.1 (fixed kernel and fixed regularization and $n = m = 100$) but for different regularization. **Left** ($\lambda = 10^3$)**:** For large regularization KFDA converges to MMD. **Middle** ($\lambda = 10^{-2}$)**:** For a good regularization the KFDA approaches clearly outperform the corresponding MMD approaches. **Right** ($\lambda = 10^{-4}$)**:** If the regularization is too small for a given sample size (here $n = 100$) , then KFDA overfits in the training phase, which leads to a reduction in test power.

B.2. Further experiments and details

This section provides supplementary information on our experiments.

Datasets. We used two different versions of the Blobs dataset. We showed random draws for both cases in Figure 3.2. For the benchmark experiments we also used the Higgs dataset [78], which is part of the *UCI Machine Learning Repository* (`https://archive.ics.uci.edu/ml/datasets/HIGGS`). We used a version that is ready for Python usage provided by [8] (`https://drive.google.com/open?id=1sHIIFCoHbauk6Mkb6e8a_tp1qnvuUOCc`). To ensure the comparability we follow the implementation of [8] and draw samples from the Higgs dataset *without replacement*.

[78]: Baldi et al. (2014), *Searching for exotic particles in high-energy physics with deep learning*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Effect of regularization of 'kfda-witness'. In the left panel of Figure 3.1, we chose a fixed regularization $\lambda = 10^{-2}$ for the KFDA methods. In Figure B.2, we show the effect of choosing a bad regularization. If the regularization is too large (left), then KFDA coincides with MMD. On the other hand, if the regularization is too small (right), then the effect of inaccurately estimating the covariance operator might as well lead to a reduced test power. For good performance it is thus important to chose a suitable regularization. This can be automated by including a model selection procedure, such as cross-validation, in the training stage.

Estimation of Rejection Rates. For the instructive experiments in Figure 3.1 we estimate the rejection rates by repeating the whole two-stage procedure 1000 times. For the benchmark experiments we use 100 iterations of the two-stage procedure for 'kfda-witness'. For all the other methods in the benchmark experiments, we follow the implementation of [8] and estimate the rejection rates by running the first stage ten times and estimating the rejection rate over 100 independent test sets for each run of the first stage. The reason for this is, that the first stage is quite slow (training a neural network).

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

Type-I errors. We report Type-I errors for all three different datasets in Figure B.1.

B.3. Approximate computation of the KFDA witness

In this section we will use n instead of n_{tr} and m instead of m_{tr} to keep the notation more concise. In Appendix B.1.3, we showed that the exact solution for the estimate of the KFDA witness is given by

$$\hat{h}_\lambda(\cdot) = \sum_{i=1}^{n+m} \hat{\alpha}_i k(z_i, \cdot), \tag{B.12}$$

$$\hat{\alpha} = \left(\frac{K N_c K}{n+m} + \lambda K \right)^{-1} K\delta. \tag{B.13}$$

Remark B.3.1 The problem with computing the KFDA witness is that a naive implementation scales cubically with the pooled sample size. In this section, we thus derive an approach that builds on recent results, that show that one can essentially get optimal convergence guarantees while only using $O((n+m)^{3/2})$ time. Therefore two steps are needed. First, the solution is approximated with $M = O((n+m)^{\frac{1}{2}})$ Nystrom centers. Second the solution with for the Nystrom centers is found via conjugate gradient, where a preconditioner is computed again with only M datapoints.

We take an approach similar to [69, 70].[2] We will thus explicitly assume that the function h has the parametric form

$$h_{\tilde{\alpha}}(x) = \sum_{m=1}^{M} \tilde{\alpha}_i k(x, \tilde{z}_i), \tag{B.14}$$

with $M = \{\tilde{z}_1, \dots, \tilde{z}_M\} \subseteq \{x_1, \dots, x_n, y_1, \dots, y_m\}$ (we overload notation and use M to denote the set itself as well as its size). We take the notation introduced in Section 3.4 and constrain to the case $c = \frac{1}{2}$. In this case we can use $N = \begin{pmatrix} P_n & 0 \\ 0 & P_m \end{pmatrix} = \frac{N_c}{2}$, instead of N_c. Note that this only affects the scaling of the solution (if we also scale λ accordingly), which is unimportant for WiTS tests. Using N instead of N_c has the advantage that N itself is idempotent $N = NN^\top$, which makes the following easier.

[69]: Rudi et al. (2017), *FALKON: An Optimal Large Scale Kernel Method*; [70]: Meanti et al. (2020), *Kernel Methods Through the Roof: Handling Billions of Points Efficiently*

2: By our results of Chapter 4, we conclude that using Kernel Ridge Regression instead of KFDA is perfectly suitable. Nevertheless, the implementation of an approximated KFDA procedure might still be of independent interest.

Algorithm 4 Pseudocode for the FdaFalkon algorithm. Adopted for KFDA from [70]

function FdaFalkon($Z, \boldsymbol{y}, k, \lambda, m$, t)
  $Z_m, \boldsymbol{y}_m \leftarrow$ RandomSubsample($(Z, \boldsymbol{y}), m$)
  $T, A \leftarrow$ Preconditioner($Z_m, \boldsymbol{y}_m, \lambda$)
  function LinOp($\boldsymbol{\beta}$)
    $\boldsymbol{v} \leftarrow A^{-1}\boldsymbol{\beta}$
    $\boldsymbol{c} \leftarrow k(Z_m, Z)NN^\top k(Z, Z_m)T^{-1}\boldsymbol{v}$
    return $A^{-\top}(T^{-\top}\boldsymbol{c} + \lambda n \boldsymbol{v})$
  $R \leftarrow A^{-\top}T^{-\top}k(Z_m, Z)\boldsymbol{y}$
  $\boldsymbol{\beta} \leftarrow$ ConjugateGradient(LinOp, R, t)
  return $T^{-1}A^{-1}\boldsymbol{\beta}, Z_m$

function Preconditioner($Z_m, \boldsymbol{y}_m, \lambda$)
  $K_{mm} \leftarrow k(Z_m, Z_m)$
  $T \leftarrow$ chol(K_{mm})
  $K_{mm} \leftarrow \frac{1}{m}TN_mN_mT^\top + \lambda \boldsymbol{I}$
  $A \leftarrow$ chol(K_{mm})
  return T, A
function kfdaWitness($\mathbb{Z}_{\text{tr}}, k, \lambda$)
  $Z \leftarrow$ Concatenate($\mathbb{Z}_{\text{tr}}$)
  $\boldsymbol{y} = [1] * \text{len}(\mathbb{X}_{\text{tr}}) + [-1] * \text{len}(\mathbb{Y}_{\text{tr}})$
  $m = \text{len}(Z)$ ▷ # Nyström centers
  $\alpha, Z \leftarrow$ FdaFalkon($Z, \boldsymbol{y}, k, \lambda, m$)
  return $h_\lambda = \sum_{i=1}^m \alpha_i k(z_i, \cdot)$

Nevertheless, it is straightforward to use the below algorithm for any $c \in (0, 1)$, simply by using $N_c = \begin{pmatrix} \frac{1}{\sqrt{c}}P_n & 0 \\ 0 & \frac{1}{\sqrt{1-c}}P_m \end{pmatrix}\begin{pmatrix} \frac{1}{\sqrt{c}}P_n & 0 \\ 0 & \frac{1}{\sqrt{1-c}}P_m \end{pmatrix}$.

In the following we denote with K_{ZM} the $(n+m) \times M$ matrix of entries $k(z_i, \tilde{z}_j)$ and K_{MZ} its transpose. We can then rewrite the terms in our objective

$$\langle \hat{\mu}_P - \hat{\mu}_Q, h_{\tilde{\alpha}} \rangle = \delta^\top K_{ZM}\tilde{\alpha}, \tag{B.15}$$

$$\langle h_{\tilde{\alpha}}, (\hat{\Sigma} + \lambda\, 1)h_{\tilde{\alpha}} \rangle = \tilde{\alpha}^\top \left(\frac{1}{n+m} K_{MZ}NN^\top K_{ZM} + \lambda K_{MM} \right) \tilde{\alpha}. \tag{B.16}$$

Let us define $R_{MZ} := K_{MZ}N$. This is a $M \times (n+m)$ matrix. Note that N is the sum of the identity and two 1-sparse matrices, hence computing R_{MZ} requires only $\mathcal{O}((n+m) \cdot M)$ operations.

With our considerations from above we can write the optimal coefficients as

$$\tilde{\alpha}^* = \left(R_{MZ}R_{MZ}^\top + (n+m)\lambda K_{MM} \right)^{-1} K_{MZ}\delta, \tag{B.17}$$

$$\Leftrightarrow \left(R_{MZ}R_{MZ}^\top + (n+m)\lambda K_{MM} \right) \tilde{\alpha}^* = K_{MZ}\delta \tag{B.18}$$

Computing $R_{MZ}R_{MZ}^\top$ explicitly costs $O((n+m)M^2)$ operations and would thus dominate the cost of our previous operations. However, Equation B.18 is now exactly in the same form as Eq. (8) in [69]. Thus from this point onwards we can build on their results to efficiently find a solution.

[69]: Rudi et al. (2017), *FALKON: An Optimal Large Scale Kernel Method*

The key idea of [69] is to find an efficient way to precondition the system of linear equations in Equation B.18. In analogy, we propose to use the following preconditioner

[69]: Rudi et al. (2017), *FALKON: An Optimal Large Scale Kernel Method*

$$BB^\top = \left(\frac{n+m}{M} R_{MM}R_{MM}^T + \lambda(n+m)K_{MM} \right)^{-1}, \tag{B.19}$$

where $R_{MM} := K_{MM}N_M$ and N_M is defined in analogy to N but only with the M Nyström centers. The preconditioner (B.19) thus corresponds to the ideal preconditioner of the problem without Nyström approximation but only M points to start with.

Using this preconditioner we use t conjugate gradient steps to solve

$$B^\top \left(R_{MZ}R_{MZ}^\top + (n+m)\lambda K_{MM} \right) B\beta = B^\top K_{MZ}\delta. \tag{B.20}$$

If $\hat{\beta}$ is the approximate solution after t steps, we obtain an approximate solution as

$$\hat{\alpha} = B\hat{\beta}. \tag{B.21}$$

The algorithm is described in Algorithm 4 and has overall complexity of $O((n_{\text{tr}} + m_{\text{tr}})Mt + M^3)$ in time and $O(M^2)$.

Appendix of Chapter 4 C.

C.1. Equivalence of squared loss and signal-to-noise ratio

C.1.1. Proof of Lemma 4.3.1

We now prove Lemma 4.3.1. While the simplicity of the relation suggests that there is an instructive proof we here give a proof based on direct calculation.

Proof of Lemma 4.3.1. After renaming we can assume that the minimizer of $(\gamma, \nu) \to L(\gamma h + \nu)$ is $(\gamma^*, \nu^*) = (1, 0)$, i.e., h itself minimizes the loss. We use the shorthand

$$\bar{h}_P = \mathbb{E}_P[h(X)] \quad \text{and} \quad \bar{h}_Q = \mathbb{E}_Q[h(Y)] \tag{C.1}$$

for the mean of h under P and Q. Note that

$$0 = \frac{\mathrm{d}}{\mathrm{d}\nu}\bigg|_{\nu=0} L(h+\nu) = 2(1-c)\mathbb{E}_P[h(X)-1] + 2c\mathbb{E}_Q[h(X)]. \tag{C.2}$$

This implies

$$c\bar{h}_Q = (1-c)(1-\bar{h}_P). \tag{C.3}$$

Similarly we get

$$0 = \frac{\mathrm{d}}{\mathrm{d}\gamma}\bigg|_{\gamma=1} L(\gamma h) = 2(1-c)\mathbb{E}_P[h(X)(h(X)-1)] + 2c\mathbb{E}_Q\left[h(Y)^2\right]. \tag{C.4}$$

We conclude that

$$(1-c)\mathbb{E}_P\left[h(X)^2\right] + c\mathbb{E}_Q\left[h(Y)^2\right] = (1-c)\bar{h}_P. \tag{C.5}$$

We observe using (C.5) and (C.3) that

$$\begin{aligned} L(h) &= (1-c)\left(\mathbb{E}_P\left[h(X)^2\right] - 2\mathbb{E}_P[h(X)] + 1\right) + c\mathbb{E}_Q\left[h(Y)^2\right] \\ &= (1-c) + (1-c)\bar{h}_P - 2(1-c)\bar{h}_P \\ &= (1-c)(1-\bar{h}_P) = c\bar{h}_Q. \end{aligned} \tag{C.6}$$

Recall that

$$\sigma_c^2(h) = \frac{(1-c)\mathrm{Var}_{X\sim P}[h(X)] + c\mathrm{Var}_{Y\sim Q}[h(Y)]}{c(1-c)}. \tag{C.7}$$

Using $\mathrm{Var}_P(h(X)) = \mathbb{E}_P\left[h(X)^2\right] - \bar{h}_P^2$ and (C.5) we derive

$$\begin{aligned} c(1-c)\sigma_c^2(h) &= (1-c)\mathbb{E}_P\left[h(X)^2\right] + c\mathbb{E}_Q\left[h(Y)^2\right] - (1-c)\bar{h}_P^2 - c\bar{h}_Q^2 \\ &= (1-c)\bar{h}_P - (1-c)\bar{h}_P^2 - c\bar{h}_Q^2 \\ &= (1-c)\bar{h}_P(1-\bar{h}_P) - c\bar{h}_Q^2 \\ &= c\bar{h}_P\bar{h}_Q - c\bar{h}_Q^2 \\ &= L(h)(\bar{h}_P - \bar{h}_Q) \end{aligned} \tag{C.8}$$

where we used (C.3) in the penultimate and (C.6) in the last step. Using the second step from the last display we obtain

$$\begin{aligned} &c(1-c)\left(\sigma_c^2(h) + (\bar{h}_P - \bar{h}_Q)^2\right) \\ &= \left((1-c)\bar{h}_P - (1-c)\bar{h}_P^2 - c\bar{h}_Q^2\right) + c(1-c)(\bar{h}_P^2 + \bar{h}_Q^2 - 2\bar{h}_P\bar{h}_Q) \\ &= (1-c)\bar{h}_P - (1-c)^2\bar{h}_P^2 - c^2\bar{h}_Q^2 - 2c(1-c)\bar{h}_P\bar{h}_Q \\ &= (1-c)\bar{h}_P - ((1-c)\bar{h}_P + c\bar{h}_Q)^2. \end{aligned} \tag{C.9}$$

Now we use (C.3) which implies $1-c = (1-c)\bar{h}_P + c\bar{h}_Q$ and get

$$\begin{aligned} c(1-c)\left(\sigma_c^2(h) + (\bar{h}_P - \bar{h}_Q)^2\right) &= (1-c)\bar{h}_P - (1-c)((1-c)\bar{h}_P + c\bar{h}_Q) \\ &= (1-c)(c\bar{h}_P - c\bar{h}_Q). \end{aligned} \tag{C.10}$$

Recall that $\mathrm{SNR}^2 = \sigma_c(h)^{-2}(\bar{h}_P - \bar{h}_Q)^2$. We thus get using (C.8) and (C.10),

$$\frac{1}{1+\mathrm{SNR}^2} = \frac{\sigma_c(h)^2}{\sigma_c(h)^2 + (\bar{h}_P - \bar{h}_Q)^2} = \frac{L(h)(\bar{h}_P - \bar{h}_Q)}{(1-c)c(\bar{h}_P - \bar{h}_Q)} = \frac{L(h)}{c(1-c)}. \tag{C.11}$$

This completes the proof. □

C.1.2. Implications for testing with MMD with an optimized kernel

As we discussed, using the mean discrepancy as a test statistic is closely connected to tests based on the MMD [5]. We now briefly discuss the implications of our findings in Subsection 4.3.1 for MMD-based tests with optimized kernel functions [7, 8].

[5]: Gretton et al. (2012), *A kernel two-sample test*

[7] showed that the asymptotic test power of an MMD-based two sample test is determined by its kernel function k via the criterion $J(P, Q; k) = \mathrm{MMD}^2(P, Q; k)/\sigma(P, Q; k)$, where $\sigma(P, Q; k)$ is the standard deviation of the MMD estimator, see Proposition 2 and Eq. (3) of [8]. Hence, they use an empirical estimate of J when optimizing the kernel function. In Appendix B.1.5 we showed that J is directly related to the SNR Equation 4.5 of the MMD-witness function:

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*; [8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

$$J(P, Q; k) = \frac{1}{\sqrt{2}}\mathrm{SNR}(h_k^{P,Q}), \tag{C.12}$$

where $h_k^{P,Q} = \mu_P - \mu_Q$ is the MMD-witness[1] of kernel k, and μ_P, μ_Q

1: Note that the MMD-witness is not defined to maximize test power.

denote the kernel mean embeddings. Hence, we can think of optimizing the kernel for an MMD two-sample test as trying to optimize the kernel such that its MMD-witness has maximal testing power in a witness two-sample test. Given this insight, we argued in Chapter 3 that maximize a witness is a more direct approach as opposed to optimizing a kernel and then using MMD. When committing to MMD nevertheless, our insights of Subsection 4.3.1 are directly applicable when optimizing the asymptotic test power of MMD-based tests:

1. Instead of optimizing J one can also optimize the kernel function by minimizing the squared loss or cross-entropy loss of its associated MMD-witness function (Proposition 4.3.2 and Remark 4.3.1). We are not aware of any work that considered these choices before, see also [7, Section 2.2] for an overview of previously used (heuristic) approaches.
2. An asymptotically optimal kernel function is $k^*(x, x') = h^*(x)h^*(x')$, with h^* given in Equation 4.7.

[7]: Sutherland et al. (2017), *Generative models and model criticism via optimized maximum mean discrepancy*

To see the second point, note that for $k^*(x, x') = h^*(x)h^*(x')$ the corresponding MMD-witness is

$$\begin{aligned} h^{P,Q}_{k^*}(x') &= h^*(x') \left(\mathbb{E}_{X\sim P}\left[h^*(X)\right] - \mathbb{E}_{Y\sim Q}\left[h^*(Y)\right] \right) \\ &\propto h^*(x'). \end{aligned} \tag{C.13}$$

Since h^* is the optimal witness and the SNR is invariant to scaling, $h^{P,Q}_{k^*}$ maximizes the right side of Equation C.12, and thus no kernel function can lead to a larger J criterion.

C.2. Further experiments and details

C.2.1. Type-I error control

In Subsection 4.3.2 we discussed two methods to obtain p-values. Based on the asymptotic distribution or based on permutations of the witness values. Since using permutations does not lead to a critical increase in computational resources, we recommend this approach by default since it controls Type-I error also at finite sample size. We empirically show this by running two experiments with the Blob and Higgs dataset with significance level $\alpha = 5\%$ and maximal training time $t_{\max} = 1\text{min}$. We follow [8] and sample $\mathbb{X}$ and $\mathbb{Y}$ from the same distributions. For each sample size we estimate the Type-I error rate over 500 independent runs and report the results in Figure C.1. Overall, on Blob we estimate a type-I error of $4.8\% \pm 0.4\%$ and Higgs of $4.3\% \pm 0.3\%$, demonstrating that our test correctly controls Type-I error.

[8]: Liu et al. (2020), *Learning Deep Kernels for Non-Parametric Two-Sample Tests*

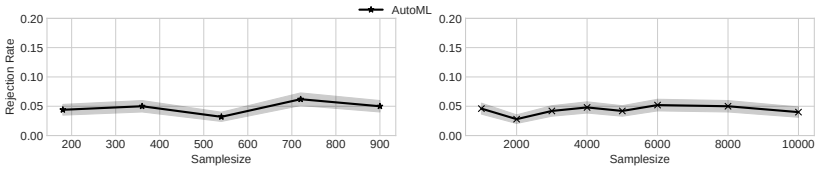


Figure C.1.: Type-I error rates with specified level $\alpha = 0.05$. **Left:** Blob **Right:** Higgs.

C.2.2. Further experiments

In Section 4.5, the default setting of our reported results was to use AutoGluon with `presets='best_quality'` and training with the MSE. We set the maximal runtime to $t_{\max} = 5$ minutes. We now report further experiments with different settings and a more fine-grained analysis for the shift detection datasets.

Blob dataset. We run different variants of the AutoML two-sample test on the Blob dataset. We use different maximal training times $t_{\max}$ and besides our default approach 'AutoML' that uses the MSE, we also consider training a classifier with AutoGluon and using its probability of class '1' as witness 'AutoML (class)'. We also consider the binary outputs of the classifier as witness 'AutoML (bin)'.

We report the test power averaged over 500 trials in Table C.1. Consistently with Remark 4.3.1 and our observations, using 'AutoML (class)' performs comparably to training with the MSE. However, thresholding the classifier to binary values drastically decreases performance. We do not observe any significant effect of allowing longer training times on this simple dataset.

All experiments were run on servers with Intel Xeon Platinum 8360Y processors, having 18 cores and 64 GB of memory each.

Table C.1.: Test power on Blob dataset.

$t_{\max}$	Test	Sample Size				
		180	360	540	720	900
1	AutoML	**0.56±0.02**	**0.98±0.01**	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (class)	0.54±0.02	0.95±0.01	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (bin)	0.39±0.02	0.84±0.02	0.99±0.00	**1.00±0.00**	**1.00±0.00**
5	AutoML	0.55±0.02	**0.98±0.01**	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (class)	0.54±0.02	0.96±0.01	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (bin)	0.37±0.02	0.83±0.02	0.98±0.01	**1.00±0.00**	**1.00±0.00**
10	AutoML	**0.56±0.02**	**0.98±0.01**	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (class)	0.53±0.02	0.97±0.01	**1.00±0.00**	**1.00±0.00**	**1.00±0.00**
	AutoML (bin)	0.36±0.02	0.84±0.02	0.99±0.01	**1.00±0.00**	**1.00±0.00**

Higgs dataset. We run the AutoML two-sample test (using MSE) for different maximal training times $t_{\max} = 1, 5, 10$ minutes on the Higgs dataset. We report our findings in Table C.2. Notice that the Blob dataset is much simpler than Higgs, since we achieve unit test power with much smaller sample size. For Higgs, we observe that the performance indeed depends on the training time. We observe that for smaller sample size, using less training time leads to increased test power. On the other hand, for larger sample size using more time is better. Although generally AutoGluon should mitigate overfitting, it seems that for small sample sizes it overfits the validation set, within the training stage. We believe that this happens because the signal in the Higgs dataset is extremely small, and the heuristics AutoGluon is using are not designed for this. For larger sample size, the general recommendation of 'allowing more time leads to better results' is recovered.

All experiments were run on the same servers as those used for the experiments on the Blob dataset.

Table C.2.: Test power on Higgs dataset.

t_{max}	Test	Sample Size							
		1000	2000	3000	4000	5000	6000	8000	10000
1	AutoML	**0.13±0.02**	**0.2±0.02**	**0.33±0.02**	**0.48±0.02**	0.59±0.02	0.72±0.02	0.84±0.02	0.94±0.01
5	AutoML	0.09±0.01	0.17±0.02	**0.33±0.02**	0.46±0.02	0.62±0.02	0.73±0.02	0.89±0.01	0.98±0.01
10	AutoML	0.09±0.01	0.17±0.02	0.25±0.02	0.40±0.02	**0.63±0.02**	**0.80±0.02**	**0.93±0.01**	**0.99±0.00**

Detecting distribution shift. All AutoML results reported in Table 4.1 were run with t_{max} = 5 minutes, we show detailed performance depending on the shift type, shift strength, and percentage of affected examples (shift frequency) in Table C.3. For completeness, in Table C.4 we also show summary results for AutoML (raw), i.e., using MSE on the raw features for 1 and 10 minute maximal runtime.

All experiments were run on servers with Intel Xeon Gold 6148 processors, having 20 cores and 48 GB of memory each.

Table C.3.: Test power for the AutoML test with different methods all run with maximal training time of t_{max} = 5 minutes.

(a) Test power depending on shift type.

Shift	Test	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
s_gn	raw 5	0.20	0.27	0.33	0.40	0.43	0.50	0.63	0.80
	pre 5	0.00	0.03	0.10	0.03	0.00	0.10	0.03	0.03
	class 5	0.20	0.17	0.30	0.37	0.47	0.50	0.53	0.80
	bin 5	0.00	0.17	0.27	0.40	0.40	0.33	0.40	0.73
m_gn	raw 5	0.27	0.23	0.33	0.43	0.43	0.53	0.63	0.83
	pre 5	0.00	0.03	0.17	0.00	0.00	0.13	0.07	0.13
	class 5	0.20	0.20	0.33	0.40	0.43	0.53	0.73	0.83
	bin 5	0.00	0.17	0.30	0.40	0.43	0.37	0.53	0.83
l_gn	raw 5	0.23	0.33	0.53	0.67	0.70	0.77	1.00	1.00
	pre 5	0.17	0.27	0.50	0.57	0.60	0.73	0.80	0.90
	class 5	0.33	0.23	0.57	0.70	0.73	0.83	0.93	1.00
	bin 5	0.03	0.17	0.43	0.67	0.70	0.67	0.80	1.00
s_img	raw 5	0.13	0.27	0.30	0.33	0.40	0.50	0.53	0.83
	pre 5	0.20	0.30	0.60	0.57	0.67	0.83	0.83	1.00
	class 5	0.23	0.10	0.30	0.37	0.43	0.50	0.50	0.87
	bin 5	0.10	0.17	0.30	0.33	0.40	0.43	0.50	0.83
m_img	raw 5	0.03	0.00	0.03	0.00	0.10	0.20	0.30	0.57
	pre 5	0.07	0.03	0.13	0.10	0.13	0.33	0.47	0.60
	class 5	0.10	0.03	0.07	0.07	0.17	0.20	0.30	0.53
	bin 5	0.00	0.00	0.07	0.10	0.10	0.03	0.20	0.50
l_img	raw 5	0.20	0.07	0.27	0.37	0.40	0.50	0.47	0.83
	pre 5	0.10	0.03	0.07	0.23	0.27	0.57	0.63	0.70
	class 5	0.07	0.07	0.33	0.33	0.47	0.43	0.47	0.83
	bin 5	0.03	0.00	0.23	0.27	0.43	0.37	0.43	0.83
adv	raw 5	0.07	0.10	0.37	0.37	0.43	0.70	0.67	0.90
	pre 5	0.27	0.33	0.53	0.67	0.60	0.83	0.80	0.87
	class 5	0.10	0.07	0.33	0.33	0.40	0.67	0.70	0.90
	bin 5	0.00	0.03	0.20	0.33	0.37	0.57	0.63	0.87
ko	raw 5	0.17	0.33	0.37	0.50	0.60	0.83	0.83	0.97
	pre 5	0.27	0.47	0.57	0.77	0.67	0.87	0.87	0.97
	class 5	0.20	0.23	0.37	0.53	0.60	0.80	0.80	0.97
	bin 5	0.07	0.13	0.30	0.43	0.63	0.73	0.73	0.97
m_img +ko	raw 5	0.00	0.03	0.23	0.53	0.53	0.67	0.67	1.00
	pre 5	0.17	0.43	0.50	0.73	0.80	1.00	1.00	1.00
	class 5	0.10	0.07	0.23	0.53	0.53	0.60	0.73	1.00
	bin 5	0.00	0.03	0.13	0.43	0.43	0.60	0.67	1.00
oz +m_img	raw 5	0.37	0.77	0.97	1.00	1.00	1.00	1.00	1.00
	pre 5	0.60	0.93	1.00	1.00	1.00	1.00	1.00	1.00
	class 5	0.33	0.77	0.97	1.00	1.00	1.00	1.00	1.00
	bin 5	0.07	0.53	0.87	0.93	1.00	1.00	1.00	1.00

(b) Test power depending on shift intensity.

Test	Intensity	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
raw 5	Small	0.14	0.11	0.21	0.26	0.31	0.40	0.47	0.73
	Medium	0.16	0.20	0.33	0.38	0.42	0.58	0.61	0.86
	Large	0.19	0.37	0.53	0.68	0.71	0.82	0.88	0.99
pre 5	Small	0.14	0.06	0.03	0.10	0.12	0.13	0.33	0.38
	Medium	0.16	0.16	0.22	0.43	0.41	0.42	0.60	0.57
	Large	0.19	0.30	0.53	0.64	0.77	0.77	0.90	0.92
class 5	Small	0.14	0.12	0.09	0.23	0.26	0.37	0.38	0.43
	Medium	0.16	0.18	0.12	0.32	0.37	0.42	0.57	0.64
	Large	0.19	0.24	0.33	0.53	0.69	0.72	0.81	0.87
bin 5	Small	0.01	0.06	0.19	0.26	0.31	0.24	0.34	0.69
	Medium	0.03	0.12	0.27	0.36	0.40	0.46	0.56	0.84
	Large	0.04	0.22	0.43	0.62	0.69	0.75	0.80	0.99

(c) Test power depending on shift frequency.

Test	Percentage	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
raw 5	10%	0.09	0.15	0.14	0.24	0.27	0.45	0.52	0.68
	50%	0.15	0.17	0.45	0.52	0.58	0.66	0.72	0.94
	100%	0.26	0.40	0.53	0.62	0.66	0.75	0.78	1.00
pre 5	10%	0.15	0.17	0.31	0.28	0.19	0.41	0.45	0.53
	50%	0.14	0.27	0.40	0.48	0.53	0.70	0.69	0.79
	100%	0.26	0.42	0.54	0.64	0.70	0.81	0.81	0.84
class 5	10%	0.07	0.10	0.16	0.23	0.34	0.43	0.50	0.68
	50%	0.16	0.13	0.44	0.54	0.58	0.68	0.71	0.94
	100%	0.33	0.35	0.54	0.62	0.65	0.71	0.80	1.00
bin 5	10%	0.02	0.08	0.12	0.22	0.23	0.26	0.32	0.66
	50%	0.02	0.08	0.29	0.51	0.58	0.61	0.69	0.91
	100%	0.05	0.26	0.52	0.56	0.66	0.66	0.76	1.00

$t_{\max}$	Test	Number of samples from test							
		10	20	50	100	200	500	1,000	10,000
5	AutoML (raw)	0.17	0.24	0.37	0.46	0.50	0.62	0.67	0.87
	AutoML (pre)	0.18	0.29	0.42	0.47	0.47	0.64	0.65	0.72
	AutoML (class)	0.19	0.19	0.38	0.46	0.52	0.61	0.67	0.87
	AutoML (bin)	0.03	0.14	0.31	0.43	0.49	0.51	0.59	0.86
1	AutoML (raw)	0.19	0.21	0.37	0.46	0.49	0.60	0.66	0.81
10	AutoML (raw)	0.15	0.24	0.38	0.46	0.51	0.61	0.67	0.88

Table C.4.: Shift detection on MNIST and CIFAR10 based on [18]. The performance of the 5-minute runtime was reported in Table 4.1. We additionally show the effect of varying the maximal runtime $t_{\max}$.

Appendix of Chapter 5 D.

D.1. Proof of Theorem 5.3.1

Proof. We make the proof in terms of the canonical feature map ϕ, which maps into the RKHS. The validity for any mapping $\varphi : \mathcal{X} \to \mathcal{H}$ that leads to the same kernel function is then trivial.

Let $\mathcal{M}(\mathcal{X}, \mathcal{A})$ denote the set of finite non-negative measures on the measurable space $(\mathcal{X}, \mathcal{A})$, i.e., $\xi(\mathcal{X}) < \infty$ for all $\xi \in \mathcal{M}(\mathcal{X}, \mathcal{A})$. We can extend the definition of the kernel mean embedding (5.1) to $\mathcal{M}(\mathcal{X}, \mathcal{A})$ by defining

$$\mu_\xi = \int_{\mathcal{X}} k(\cdot, x) \mathrm{d}\xi(x) = \int_{\mathcal{X}} \phi(x) \mathrm{d}\xi(x), \tag{D.1}$$

for any $\xi \in \mathcal{M}(\mathcal{X}, \mathcal{A})$ that fulfills $\int_{\mathcal{X}} k(x, x) \mathrm{d}\xi(x) < \infty$. Let ξ_1 and ξ_2 be arbitrary measures in $\mathcal{M}(\mathcal{X}, \mathcal{A})$. By assumption, k is universal over $\mathcal{C}_0(\mathcal{X})$ and thus characteristic over $\mathcal{M}(\mathcal{X}, \mathcal{A})$, i.e., $\mu_{\xi_1} = \mu_{\xi_2} \Leftrightarrow \xi_1 = \xi_2$; see Theorem 6 in [111].

[111]: Simon-Gabriel et al. (2018), *Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions*

Define ν_P as the mean embedding onto the unit sphere of the RKHS

$$\nu_P := \frac{1}{\mathcal{N}_P} \mu_P, \tag{D.2}$$

with $\mathcal{N}_P \in \mathbb{R}^+$ such that $\|\nu_P\|_{\mathcal{H}_k} = 1$. Let P and Q be probability measures for which the embedding onto the unit sphere (D.2) coincide, i.e., $\nu_P = \nu_Q$. We can relate this to the kernel mean embeddings as

$$\mu_P = \mathcal{N}_P \, \nu_Q = \frac{\mathcal{N}_P}{\mathcal{N}_Q} \mu_Q = \mu_\xi, \tag{D.3}$$

where we defined the finite non-negative measure $\xi = \frac{\mathcal{N}_P}{\mathcal{N}_Q} Q$, using the linearity of Equation D.1. With the injectivity of the embedding (D.1) this implies $P = \xi = \frac{\mathcal{N}_P}{\mathcal{N}_Q} Q$. By assumption, P and Q are probability measures and fulfill $P(\mathcal{X}) = Q(\mathcal{X}) = 1$. This implies $\frac{\mathcal{N}_P}{\mathcal{N}_Q} = 1$ and thus $P = Q$, which proves the injectivity of ν for the set of probability distributions. □

D.2. Coherent states and Gaussian kernel

In this section, we consider an explicit example, previously reported in [104]. Let $\mathcal{H}$ be an infinite dimensional (complex) Hilbert space, with orthonormal basis $\{|n\rangle\}_{n \in \mathbb{N}_0}$. This could for example be the space corresponding to a single mode of the electro-magnetic field [186]. For simplicity we consider $\mathcal{X} = \mathbb{R}$ and define the feature map $\varphi : \mathbb{R} \to \mathcal{H}$ as

[104]: Chatterjee et al. (2017), *Generalized Coherent States, Reproducing Kernels, and Quantum Support Vector Machines*

[186]: Strekalov et al. (2019), *Nonlinear Interactions and Non-classical Light*

$$|\varphi(x)\rangle = e^{-\frac{1}{2}x^2} \sum_{n=0}^{\infty} \frac{x^n}{\sqrt{n!}} |n\rangle \,. \tag{D.4}$$

In quantum optics, the states $|n\rangle$ are called Fock states. States of the form of Equation D.4 are called coherent states and are well studied [187]. In the context of this paper, however, the nature of the basis and hence the exact form of the Hilbert space are unimportant. The important part is the orthonormality of the basis states, which implies

$$\langle\varphi(x)|\varphi(x')\rangle = e^{-\frac{1}{2}(x-x')^2} =: k(x,x'), \tag{D.5}$$

for arbitrary $x, x' \in \mathbb{R}$ and defines the popular Gaussian kernel [4]. By composing the mapping (D.4) with the mapping $x \mapsto \frac{x}{\sigma}$, for some $\sigma > 0$, it is also possible to include a bandwidth parameter σ. The Gaussian kernel fulfills the requirements of Theorem 5.3.1 (see, [111, theorem 17]). Therefore, it is possible to construct an injective embedding of probability distributions over the real numbers in a superposition of coherent states.

Coherent states are commonly considered the *most classical* states in quantum optics, and are easy to simulate on a classical device. Working with a quantum device becomes interesting when the states become *non-classical* [186]. When using the coherent feature map (D.4), the embedding of a sample (5.10) corresponds to the so-called *cat-states* [108, 126, 127]. Cat-states are considered nonclassical, as their Wigner function attains negative values. From a quantum perspective, this already hints to the difficulties encountered when working with such states on a classical devices.

[187]: Agarwal (2012), *Quantum optics*

[4]: Schölkopf et al. (2001), *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*

[111]: Simon-Gabriel et al. (2018), *Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions*

[186]: Strekalov et al. (2019), *Nonlinear Interactions and Non-classical Light*

[108]: Vlastakis et al. (2013), *Deterministically Encoding Quantum Information Using 100-Photon Schrödinger Cat States*; [126]: Deléglise et al. (2008), *Reconstruction of non-classical cavity field states with snapshots of their decoherence*; [127]: Ourjoumtsev et al. (2007), *Generation of optical 'Schrödinger cats' from photon number states*

D.3. Estimation of $\mathcal{N}_{\mathbb{X}}$

In order to obtain $\mathcal{N}_{\mathbb{X}}$ without explicitly calculating Equation 5.11, we can evaluate $\mathcal{N}_{\mathbb{X}}$ by estimating the inner product with a reference state $|\psi_{\text{ref}}\rangle = |\varphi(x_{\text{ref}})\rangle$ for some reference value $x_{\text{ref}} \in \mathcal{X}$. To this end, we analytically calculate

$$c := \frac{1}{n}\sum_{i=1}^{n} \langle\psi_{\text{ref}}|\varphi(x_i)\rangle = \frac{1}{n}\sum_{i=1}^{n} k(x_{\text{ref}}, x_i), \tag{D.6}$$

using $O(n)$ operations. Now given the preparation of $|\nu_{\mathbb{X}}\rangle$ and of $|\psi_{\text{ref}}\rangle$ we can experimentally evaluate the inner product $\langle\psi_{\text{ref}}|\nu_{\mathbb{X}}\rangle$ and from this obtain the normalization $\mathcal{N}_{\mathbb{X}} = c\langle\psi_{\text{ref}}|\nu_{\mathbb{X}}\rangle^{-1}$. Obviously, in order to make this well defined, we need to choose the reference function such that $\langle\psi_{\text{ref}}|\nu_{\mathbb{X}}\rangle \neq 0$. This strategy relies on the two challenges phrased in the main text, i.e., the preparation of $|\nu_{\mathbb{X}}\rangle$ and the estimation of inner products, but apart from this does not pose an extra difficulty by itself. We emphasize again that due to Theorem 5.3.1 it should be possible to come up with algorithms that directly work with the QME and hence make the estimation of the normalization superfluous.

Appendix of Chapter 6 | E.

E.1. Conditional moment discrepancy (CMMD)

The maximum moment restriction (MMR) also allows us to compare two different models based on the conditional moment restriction (CMR). Let $\mathcal{M}_{\theta_1}$ and $\mathcal{M}_{\theta_2}$ be two models parameterized by $\theta_1, \theta_2 \in \Theta$, respectively. Then, we can define a CMR-based discrepancy measure between these two models as follows.

Definition E.1.1 *For $\theta_1, \theta_2 \in \Theta$, a conditional moment discrepancy (CMMD) is defined as $\Delta(\theta_1, \theta_2) := \|\boldsymbol{\mu}_{\theta_1} - \boldsymbol{\mu}_{\theta_2}\|_{\mathcal{F}^p}$.*

By Theorem 6.3.3, $\Delta(\theta_1, \theta_2) \geq 0$ and $\Delta(\theta_1, \theta_2) = 0$ if and only if the two models $\mathcal{M}_{\theta_1}$ and $\mathcal{M}_{\theta_2}$ are indistinguishable in terms of the CMR alone. Moreover, if the global identifiability **(A3)** holds, $\Delta(\theta_0, \theta) = \mathbb{M}(\theta)$ for all $\theta \in \Theta$. Since

$$\Delta(\theta_1, \theta_2) = \|\mathbb{E}[\boldsymbol{\xi}_{\theta_1}(X, Z) - \boldsymbol{\xi}_{\theta_2}(X, Z)]\|_{\mathcal{F}^q} = \left\|\mathbb{E}[\bar{\boldsymbol{\xi}}(X, Z)]\right\|_{\mathcal{F}^q}$$

where $\bar{\boldsymbol{\xi}}(x, z) := \boldsymbol{\xi}_{\theta_1}(x, z) - \boldsymbol{\xi}_{\theta_2}(x, z) = (\boldsymbol{\psi}(z; \theta_1) - \boldsymbol{\psi}(z; \theta_2))k(x, \cdot)$, the CMMD can be viewed as the MMR defined on a *differential residual function* $\boldsymbol{\psi}(z; \theta_1) - \boldsymbol{\psi}(z; \theta_2)$. As a result, $\Delta(\theta_1, \theta_2)$ also has a closed-form expression similar to that in Theorem 6.3.4.

Corollary E.1.1 *For $\theta_1, \theta_2 \in \Theta$, let*

$$h((x, z), (x', z')) := (\boldsymbol{\psi}(z; \theta_1) - \boldsymbol{\psi}(z; \theta_2))^\top (\boldsymbol{\psi}(z'; \theta_1) - \boldsymbol{\psi}(z'; \theta_2))k(x, x')$$

and assume that $\mathbb{E}[h((X, Z), (X, Z))] < \infty$. Then, we have $\Delta^2(\theta_1, \theta_2) = \mathbb{E}[h((X, Z), (X', Z'))]$ where (X', Z') is independent copy of (X, Z) with an identical distribution.

Proof. The result follows by applying the proof of Theorem 6.3.4 to the feature map $\bar{\boldsymbol{\xi}}(x, z) := \boldsymbol{\xi}_{\theta_1}(x, z) - \boldsymbol{\xi}_{\theta_2}(x, z) = (\boldsymbol{\psi}(z; \theta_1) - \boldsymbol{\psi}(z; \theta_2))k(x, \cdot)$. □

Furthermore, we can express the empirical CMMD as

$$\Delta_n^2(\theta_1, \theta_2) := \frac{1}{n(n-1)} \sum_{1 \leq i \neq j \leq n} h((x_i, z_i), (x_j, z_j))$$

where

$$\begin{aligned} &h((x_i, z_i), (x_j, z_j)) \\ &\quad := (\boldsymbol{\psi}(z_i; \theta_1) - \boldsymbol{\psi}(z_i; \theta_2))^\top (\boldsymbol{\psi}(z_j; \theta_1) - \boldsymbol{\psi}(z_j; \theta_2))k(x_i, x_j). \end{aligned}$$

As we can see, the RKHS norm, inner product, and function evaluation computed with respect to $\boldsymbol{\mu}_\theta$ all have meaningful economic interpretations. Table E.1 summarizes these interpretations.

Operation	Interpretation
$\|\boldsymbol{\mu}_\theta\|_{\mathcal{F}^q}$	conditional moment violation
$\langle f, \boldsymbol{\mu}_\theta \rangle_{\mathcal{F}^q}$	violation w.r.t. the instrument f
$\boldsymbol{\mu}_\theta(x, z)$	structural instability at (x, z)
$\|\boldsymbol{\mu}_{\theta_1} - \boldsymbol{\mu}_{\theta_2}\|_{\mathcal{F}^q}$	discrepancy between $\mathcal{M}_{\theta_1}$ and $\mathcal{M}_{\theta_2}$

Table E.1.: Interpretations of different operations on $\boldsymbol{\mu}_\theta$ in $\mathcal{F}^q$.

E.2. Parameter estimation

Besides hypothesis testing, another important application of the CMR is parameter estimation. That is, given the CMR as in Equation 6.1, we aim to find an estimate of θ_0 that satisfies Equation 6.1 from the observed data $(x_i, z_i)_{i=1}^n$. Based on the MMR, we define the estimator of θ_0 as the parameter that minimizes Equation 6.10:

$$\hat{\theta}_n := \arg\min_{\theta \in \Theta} \widehat{\mathbb{M}}_n^2(\theta) = \arg\min_{\theta \in \Theta} \frac{1}{n(n-1)} \sum_{1 \le i \ne j \le n} h_\theta((x_i, z_i), (x_j, z_j)). \tag{E.1}$$

We call $\hat{\theta}_n$ a *minimum maximum moment restriction* (MMMR) estimate of θ_0. Note that it is also possible to adopt V-statistic in Equation E.1 instead of the U-statistic. Previously, [145] and [144] proposed to estimate θ_0 based on Equation 6.5 and $\mathcal{F}$ that is parameterized by deep neural networks. However, their algorithms require solving a minimax game, whereas our approach for estimation is merely a minimization problem.

[145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*

[144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*

The following theorem shows that $\hat{\theta}_n$ is a consistent estimate of θ_0. The proof can be found in Appendix E.4.6.

Theorem E.2.1 (Consistency of $\hat{\theta}_n$) *Assume that the parameter space Θ is compact. Then, we have $\hat{\theta}_n \xrightarrow{p} \theta_0$.*

Despite the consistency, we suspect that $\hat{\theta}_n$ may not be asymptotically efficient and there exist better estimators. Theorem 6.3.5 shows that $\mathbb{M}(\theta)$ depends on a continuum of moment conditions reweighted by the non-uniform eigenvalues $(\lambda_j)_j$, which suggests that a *reweighting matrix* must also be incorporated in order to achieve the optimality [150]. Constructing an optimal choice of reweighting matrix in an infinite dimensional RKHS is an interesting topic [165], and we leave it to future work.

[150]: Hall (2005), *Generalized Method of Moments*

[165]: Carrasco et al. (2000), *Generalization of GMM to a Continuum of Moment Conditions*

E.2.1. Maximum moment restriction for instrumental variable regression

To illustrate one of the advantages of the MMR for parameter estimation, let us consider the nonparametric instrumental variable regression problem [144, 145, 147, 148, 188]. Let X be a treatment (endogeneous) variable taking values in $\mathcal{X} \subseteq \mathbb{R}^d$ and Y a real-valued outcome variable. Our goal is to estimate a function $g : \mathcal{X} \to \mathbb{R}$ from a structural equation model (SEM) of the form

$$Y = g(X) + \varepsilon, \quad X = h(Z) + f(\varepsilon) + \nu, \tag{E.2}$$

[144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*; [145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*; [147]: Singh et al. (2019), *Kernel Instrumental Variable Regression*; [148]: Muandet et al. (2020), *Dual instrumental variable regression*; [188]: Angrist et al. (2008), *Mostly Harmless Econometrics: An Empiricist's Companion*

where we assume that $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\nu] = 0$. Unfortunately, as we can see from Equation E.2, ε is correlated with the treatment X, i.e., $\mathbb{E}[\varepsilon|X] \neq 0$, and hence standard regression methods cannot be used to estimate g. This setting often arises when there exist unobserved confounders between the treatment X and outcome Y.

In instrumental variable regression, we assume access to an *instrumental* variable Z which is associated with the treatments X, but not with the outcome variable Y, other than through its effect on the treatments. Moreover, the instrument Z is assumed to be uncorrelated with ε. This implies the conditional moment restriction $\mathbb{E}[\varepsilon \mid Z] = \mathbb{E}[Y - g(X) \mid Z] = 0$ for P_Z-almost all z [138, 144, 145]. Given an i.i.d. sample $(x_i, y_i, z_i)_{i=1}^n$ from $P(X, Y, Z)$, the MMR allows us to reduce the problem of estimating g to a regularized empirical risk minimization (ERM) problem

$$\begin{aligned} \widehat{g}_\lambda &:= \arg\min_{g \in \mathcal{G}_l} \widehat{\mathbb{M}}_n^2(g) + \lambda \|g\|_{\mathcal{G}_l}^2 \\ &= \arg\min_{g \in \mathcal{G}_l} \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n (y_i - g(x_i))(y_j - g(x_j)) k(z_i, z_j) + \lambda \|g\|_{\mathcal{G}_l}^2 \end{aligned} \tag{E.3}$$

where λ is a positive regularization parameter and $\mathcal{G}_l$ is a reproducing kernel Hilbert space (RKHS) of real-valued functions on $\mathcal{X}$ with the reproducing kernel $l : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Note that we adopt the V-statistic instead of the U-statistic in Appendix E.2.1. By the representer theorem, the optimal solution to Appendix E.2.1 can be expressed as a linear combination

$$\widehat{g}_\lambda(x) = \sum_{i=1}^n \alpha_i l(x, x_i) \tag{E.4}$$

for some $(\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$. Let $K = [k(z_i, z_j)]_{i,j}$ and $L = [l(x_i, x_j)]_{i,j}$ be the kernel matrices in $\mathbb{R}^{n \times n}$ of $\boldsymbol{z} = [z_1, \ldots, z_n]^\top$ and $\boldsymbol{x} = [x_1, \ldots, x_n]^\top$, respectively, and $\boldsymbol{y} := [y_1, \ldots, y_n]^\top$. Substituting Equation E.4 back into Appendix E.2.1 yields a *generalized ridge regression* (GRR) problem

$$\boldsymbol{\alpha}_\lambda := \arg\min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{n^2} (\boldsymbol{y} - L\boldsymbol{\alpha})^\top K (\boldsymbol{y} - L\boldsymbol{\alpha}) + \lambda \boldsymbol{\alpha}^\top L \boldsymbol{\alpha}. \tag{E.5}$$

That is, the optimal coefficients $\boldsymbol{\alpha}_\lambda$ can be obtained by solving the first-order stationary condition $(LKL + n^2 \lambda L)\boldsymbol{\alpha} = LK\boldsymbol{y}$ and if L is positive definite, the solution has a *closed-form* expression, i.e.,

$$\widehat{g}_\lambda(x) = \sum_{i=1}^n \alpha_{\lambda,i} l(x, x_i), \qquad \boldsymbol{\alpha}_\lambda = (LKL + n^2 \lambda L)^{-1} LK\boldsymbol{y}. \tag{E.6}$$

Similar techniques have been considered in [147] and [148]. In [147], the authors extended the two-stage least square (2SLS) by modeling the first-stage regression with the conditional mean embedding of $P(X|Z)$ [19] which is then used in the second-stage kernel ridge regression. In [148], the authors showed that the two-stage procedure can be reformulated as a convex-concave saddle-point problem. When the solutions lie in the RKHS, the closed-form solution similar to Equation E.6 and the one in [147] can be obtained. By contrast, the MMR-based approach allows us to reformulate the problem directly as a generalized ridge regression (GRR) in which the values of hyperparameters, e.g., the regularization parameter λ, can be chosen via the popular cross-validation procedures.

[138]: Newey (1993), *Efficient estimation of models with conditional moment restrictions*; [144]: Bennett et al. (2019), *Deep Generalized Method of Moments for Instrumental Variable Analysis*; [145]: Lewis et al. (2018), *Adversarial Generalized Method of Moments*

[147]: Singh et al. (2019), *Kernel Instrumental Variable Regression*

[148]: Muandet et al. (2020), *Dual instrumental variable regression*

[147]: Singh et al. (2019), *Kernel Instrumental Variable Regression*

[19]: Muandet et al. (2017), *Kernel Mean Embedding of Distributions: A Review and Beyond*

[148]: Muandet et al. (2020), *Dual instrumental variable regression*

[147]: Singh et al. (2019), *Kernel Instrumental Variable Regression*

E.3. Experiments

In this section, we provide further description of our experiments as well as additional experimental results.

E.3.1. Simultaneous equation models

A simultaneous equation model (SEM) is a fundamental concept in economics. In one of our experiments, we consider the following SEM:

$$\begin{aligned} Q &= \alpha_d P + \beta_d R + U, \quad \alpha_d < 0, \qquad \text{(Demand)} \\ Q &= \alpha_s P + \beta_s W + V, \quad \alpha_s > 0, \qquad \text{(Supply)} \end{aligned} \tag{E.7}$$

where Q and P denote quantity and price, respectively, R and W are exogeneous variables, and U and V are the error terms. To obtain *reduced-form equations* of (E.7), we must solve for the endogeneous variables P and Q. First, we solve for P by equating the two equations in (E.7):

$$P = \left[\frac{\beta_s}{\alpha_d - \alpha_s}\right] W - \left[\frac{\beta_d}{\alpha_d - \alpha_s}\right] R + \frac{V - U}{\alpha_d - \alpha_s}. \tag{E.8}$$

Then, we can solve for Q by plugging in P to the supply equation in (E.7):

$$Q = \left[\frac{\alpha_s \beta_s}{\alpha_d - \alpha_s} + \beta_s\right] W - \left[\frac{\alpha_s \beta_d}{\alpha_d - \alpha_s}\right] R + \frac{\alpha_s}{\alpha_d - \alpha_s}(V - U) + V. \tag{E.9}$$

By comparing Equation E.8 and Equation E.9 to the data generating process in our experiment, we obtain the following system of equations:

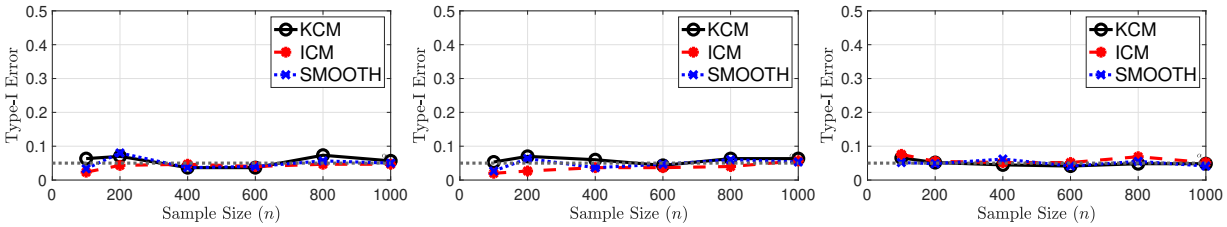
$$\begin{aligned} \lambda_{11} &= -\frac{\alpha_s \beta_d}{\alpha_d - \alpha_s}, & \lambda_{21} &= -\frac{\beta_d}{\alpha_d - \alpha_s} \\ \lambda_{12} &= \frac{\alpha_s \beta_s}{\alpha_d - \alpha_s} + \beta_s, & \lambda_{22} &= \frac{\beta_s}{\alpha_d - \alpha_s}. \end{aligned} \tag{E.10}$$

Finally, setting $(\lambda_{11}, \lambda_{12}, \lambda_{21}, \lambda_{22}) = (1, -1, 1, 1)$ and then solving the system of equations (E.10) results in a non-trivial solution $(\alpha_d, \beta_d, \alpha_s, \beta_s) = (-1, 2, 1, -2)$. This solution coincides with the one obtained from the two-stage least square (2SLS) procedure [188, Ch. 4].

[188]: Angrist et al. (2008), *Mostly Harmless Econometrics: An Empiricist's Companion*

E.3.2. Type-I errors

The KCM test with bootstrapping is based on the asymptotic distribution of the test statistic under H_0 (cf. Theorem 6.4.1). Hence, the test reliably controls the Type-I error when the sample size is sufficiently large, i.e., we are in the asymptotic regime. For the considered examples, this is the case already for moderate sample sizes. We report the Type-I error at a significance level $\alpha = 0.05$ for $n \in \{100, 200, 400, 600, 800, 1000\}$ in Figure E.1.



(a) REG-HOM **(b)** REG-HET **(c)** SIMEQ

Figure E.1.: The Type-I errors averaged over 300 trials of KCM, ICM, and smooth tests under the null hypothesis ($\delta = 0$) as we vary the sample size n.

E.4. Proofs

This section collects all the proofs of the results presented in Chapter 6.

E.4.1. Proof of Lemma 6.3.1

Proof. We have $M_{\theta_0} f = \sum_{i=1}^{q} \mathbb{E}[\psi_i(Z;\theta_0) f_i(X)]$ and, for all $i = 1, \ldots, q$,

$$\begin{aligned}\mathbb{E}_{XZ}[\psi_i(Z;\theta_0) f_i(X)] &= \mathbb{E}_X[\mathbb{E}_Z[\psi_i(Z;\theta_0) f_i(X)|X]] \\ &= \mathbb{E}_X[\mathbb{E}_Z[\psi_i(Z;\theta_0)|X] f_i(X)] \\ &= 0\end{aligned}$$

by the law of iterated expectation. The last equality follows from the definition of θ_0 and the continuity of f_i, i.e., by Assumption **(A4)**. □

E.4.2. Proof of Theorem 6.3.2

Our result follows directly from [189, Lemma 2] and [190, Theorem 3.4] which rely on the Bennett inequality for vector-valued random variables. We reproduce the proof here for completeness.

[189]: Smale et al. (2007), *Learning Theory Estimates via Integral Operators and Their Approximations*

[190]: Pinelis (1994), *Optimum Bounds for the Distributions of Martingales in Banach Spaces*

Proof. First, recall that $\mu_\theta = \mathbb{E}[\xi_\theta(X, Z)]$ and $\widehat{\mu}_\theta = \frac{1}{n}\sum_{i=1}^{n} \xi_\theta(x_i, z_i)$ for the independent random variables $\{\xi_\theta(x_i, z_i)\}_{i=1}^{n}$. Then, for any $\varepsilon > 0$, it follows from [189, Lemma 1] that

[189]: Smale et al. (2007), *Learning Theory Estimates via Integral Operators and Their Approximations*

$$P\left\{\left\|\frac{1}{n}\sum_{i=1}^{n} \xi_\theta(x_i, z_i) - \mu_\theta\right\|_{\mathcal{F}^q} \geq \varepsilon\right\} \leq 2\exp\left\{-\frac{n\varepsilon}{2C_\theta}\log\left(1 + \frac{C_\theta \varepsilon}{\sigma_\theta^2}\right)\right\}.$$

Taking $t := C_\theta \varepsilon / \sigma_\theta^2$ and applying the inequality $\log(1+t) \geq t/(1+t)$ for all $t > 0$ yield

$$\begin{aligned}P\left\{\left\|\frac{1}{n}\sum_{i=1}^{n} \xi_\theta(x_i, z_i) - \mu_\theta\right\|_{\mathcal{F}^q} \geq \varepsilon\right\} &\leq 2\exp\left\{-\frac{n\varepsilon}{2C_\theta}\left(\frac{C_\theta\varepsilon}{C_\theta\varepsilon + \sigma_\theta^2}\right)\right\} \\ &= 2\exp\left\{-\frac{n\varepsilon^2}{2C_\theta\varepsilon + 2\sigma_\theta^2}\right\}.\end{aligned}$$

The value of $\varepsilon > 0$ for which this probability equal to δ can be obtained by solving the quadratic equation $n\varepsilon^2 = \log(2/\delta)(2C_\theta\varepsilon + 2\sigma_\theta^2)$. As a

result, we have with confidence $1-\delta$ that

$$\left\| \frac{1}{n}\sum_{i=1}^{n} \xi_\theta(x_i, z_i) - \mu_\theta \right\|_{\mathcal{F}^q} \leq \frac{2C_\theta \log\frac{2}{\delta}}{n} + \sqrt{\frac{2\sigma_\theta^2 \log\frac{2}{\delta}}{n}}, \tag{E.11}$$

as required. □

It remains to show that, for each $\theta \in \Theta$, there exists a constant $C_\theta < \infty$ such that $\|\xi_\theta(X,Z)\|_{\mathcal{F}^q} < C_\theta$ almost surely. Note that for any $(x,z) \in \mathcal{X}\times\mathcal{Z}$ for which $P_{XZ}(x,z) > 0$,

$$\begin{aligned} \|\xi_\theta(x,z)\|_{\mathcal{F}^p} &= \sqrt{\|\xi_\theta(x,z)\|_{\mathcal{F}^p}^2} \\ &= \sqrt{\psi(z;\theta)^\top \psi(z;\theta) k(x,x)} \\ &\leq \sup_{x,z} \sqrt{\psi(z;\theta)^\top \psi(z;\theta) k(x,x)} < \infty, \end{aligned}$$

where the last inequality follows from Assumptions **(A2)** and **(A4)**.

E.4.3. Proof of Theorem 6.3.3

Proof. If $\mathcal{M}(x;\theta_1) = \mathcal{M}(x;\theta_2)$ for P_X-almost all x, then the equality $\mu_{\theta_1} = \mu_{\theta_2}$ follows straightforwardly. Suppose that $\mu_{\theta_1} = \mu_{\theta_2}$ and let $\delta(x) := \mathcal{M}(x;\theta_1) - \mathcal{M}(x;\theta_2)$. Then, we have

$$\begin{aligned} & \|\mu_{\theta_1} - \mu_{\theta_2}\|_{\mathcal{F}^q}^2 \\ = & \left\| \int \xi_{\theta_1}(x,z)\,\mathrm{d}P_{XZ}(x,z) - \int \xi_{\theta_2}(x,z)\,\mathrm{d}P_{XZ}(x,z) \right\|_{\mathcal{F}^q}^2 \\ = & \left\| \int \mathcal{M}(x;\theta_1) k(x,\cdot)\,\mathrm{d}P_X(x) - \int \mathcal{M}(x;\theta_2) k(x,\cdot)\,\mathrm{d}P_X(x) \right\|_{\mathcal{F}^q}^2 \\ = & \left\| \int (\mathcal{M}(x;\theta_1) - \mathcal{M}(x;\theta_2)) k(x,\cdot)\,\mathrm{d}P_X(x) \right\|_{\mathcal{F}^q}^2 \\ = & \iint \delta(x)^\top k(x,x')\delta(x')\,\mathrm{d}P_X(x)\,\mathrm{d}P_{X'}(x') = 0, \end{aligned} \tag{E.12}$$

where X' is an independent copy of X. It follows from Appendix E.4.3 and Assumption **(A2)** that the function $g(x) := \delta(x)p_X(x)$ has zero L2-norm, i.e., $\|g\|_2^2 = 0$ where p_X denotes the density of P_X. As a result, $\delta(x) = \mathbf{0}$ a.e. P_X implying that $P_X(B_0) = 1$ where $B_0 := \{x \in \mathcal{X} : \mathcal{M}(x;\theta_1) - \mathcal{M}(x;\theta_2) = \mathbf{0}\}$. Therefore, $\mathcal{M}(x;\theta_1) = \mathcal{M}(x;\theta_2)$ for P_X-almost all x. This completes the proof. □

E.4.4. Proof of Theorem 6.3.4

Proof. By the definition of $\mathbb{M}(\theta)$ and the Bochner integrability of $\boldsymbol{\xi}_\theta$,

$$\begin{aligned}
\mathbb{M}^2(\theta) &= \|\boldsymbol{\mu}_\theta\|_{\mathcal{F}^q}^2 \\
&= \langle \boldsymbol{\mu}_\theta, \boldsymbol{\mu}_\theta \rangle_{\mathcal{F}^q} \\
&= \langle \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)], \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)] \rangle_{\mathcal{F}^q} \\
&= \mathbb{E}[\langle \boldsymbol{\xi}_\theta(X,Z), \mathbb{E}[\boldsymbol{\xi}_\theta(X,Z)] \rangle_{\mathcal{F}^q}] \\
&= \mathbb{E}[\langle \boldsymbol{\xi}_\theta(X,Z), \boldsymbol{\xi}_\theta(X',Z') \rangle_{\mathcal{F}^q}] \\
&= \mathbb{E}[h_\theta((X,Z),(X',Z'))],
\end{aligned}$$

where (X', Z') is an independent copy of (X, Z) with an identical distribution. □

E.4.5. Proof of Theorem 6.3.5

Proof. By Mercer's theorem [98, Theorem 4.49], we have $k(x,x') = \sum_j \lambda_j e_j(x) e_j(x')$ where the convergence is absolute and uniform. Recall that $\boldsymbol{\zeta}_\theta^j(x,z) := \big(\psi_1(z;\theta)e_j(x), \ldots, \psi_q(z;\theta)e_j(x)\big)$. Hence, we can express the kernel h_θ as

[98]: Steinwart et al. (2008), *Support Vector Machines*

$$\begin{aligned}
h_\theta((x,z),(x',z')) &= \boldsymbol{\psi}(z;\theta)^\top \boldsymbol{\psi}(z';\theta) k(x,x') \\
&= \boldsymbol{\psi}(z;\theta)^\top \boldsymbol{\psi}(z';\theta) \left(\sum_j \lambda_j e_j(x) e_j(x') \right) \\
&= \sum_j \lambda_j \boldsymbol{\psi}(z;\theta)^\top \boldsymbol{\psi}(z';\theta) e_j(x) e_j(x') \\
&= \sum_j \lambda_j \left[\boldsymbol{\psi}(z;\theta) e_j(x) \right]^\top \left[\boldsymbol{\psi}(z';\theta) e_j(x') \right] \\
&= \sum_j \lambda_j \boldsymbol{\zeta}_\theta^j(x,z)^\top \boldsymbol{\zeta}_\theta^j(x',z').
\end{aligned}$$

Since $\lambda_j > 0$, the function h_θ is positive definite. Then, we can express $\mathbb{M}^2(\theta)$ as follows:

$$\begin{aligned}
\mathbb{M}^2(\theta) &= \mathbb{E}[h_\theta((X,Z),(X',Z'))] \\
&= \mathbb{E}\left[\sum_j \lambda_j \boldsymbol{\zeta}_\theta^j(X,Z)^\top \boldsymbol{\zeta}_\theta^j(X',Z') \right] \\
&= \sum_j \lambda_j \mathbb{E}_{XZ}\left[\boldsymbol{\zeta}_\theta^j(X,Z) \right]^\top \mathbb{E}_{X'Z'}\left[\boldsymbol{\zeta}_\theta^j(X',Z') \right] \\
&= \sum_j \lambda_j \left\| \mathbb{E}_{XZ}\left[\boldsymbol{\zeta}_\theta^j(X,Z) \right] \right\|_2^2.
\end{aligned}$$

This completes the proof. □

E.4.6. Proof of Theorem E.2.1

In order to show the consistency of $\hat{\theta}_n := \arg\min_{\theta\in\Theta} \widehat{\mathbb{M}}_n^2(\theta)$, we need the uniform consistency of $\widehat{\mathbb{M}}_n^2(\theta)$ and the continuity of $\theta \mapsto \mathbb{M}^2(\theta)$. The

following lemma gives these two results.

Lemma E.4.1 *Assume that there exists an integrable and symmetric function F_ψ such that $\|\boldsymbol{\psi}(z,\theta)\|_2 \le F_\psi(z)$ for any $\theta \in \Theta$ and $z \in \mathcal{Z}$. If Assumption* **(A4)** *holds, $\sup_{\theta\in\Theta} |\mathbb{M}_n^2(\theta) - \mathbb{M}^2(\theta)| \xrightarrow{p} 0$ and $\theta \mapsto \mathbb{M}^2(\theta)$ are continuous.*

Proof. Recall that

$$\begin{aligned}\mathbb{M}^2(\theta) &= \mathbb{E}[h_\theta((X,Z),(X',Z'))],\\ \widehat{\mathbb{M}}_n^2(\theta) &= \frac{1}{n(n-1)}\sum_{1\le i\ne j\le n} h_\theta((x_i,z_i),(x_j,z_j)),\end{aligned}$$

where

$$h_\theta((x,z),(x',z')) = \langle \boldsymbol{\xi}_\theta(x,z), \boldsymbol{\xi}_\theta(x',z')\rangle_{\mathcal{F}^q} = \boldsymbol{\psi}(z;\theta)^\top \boldsymbol{\psi}(z';\theta) k(x,x').$$

Then, it follows that

$$\begin{aligned}&|h_\theta((x,z),(x',z'))|\\ =&|\langle \boldsymbol{\xi}_\theta(x,z), \boldsymbol{\xi}_\theta(x'z')\rangle_{\mathcal{F}^q}|\\ \le&\|\boldsymbol{\xi}_\theta(x,z)\|_{\mathcal{F}^q}\cdot\|\boldsymbol{\xi}_\theta(x',z')\|_{\mathcal{F}^q}\\ =&\sqrt{\boldsymbol{\psi}(z;\theta)^\top\boldsymbol{\psi}(z;\theta)k(x,x)}\sqrt{\boldsymbol{\psi}(z';\theta)^\top\boldsymbol{\psi}(z';\theta)k(x',x')}\\ =&\|\boldsymbol{\psi}(z;\theta)\|_2\|\boldsymbol{\psi}(z';\theta)\|_2\sqrt{k(x,x)k(x',x')}\\ \le&F_\psi(z)F_\psi(z')\sqrt{k(x,x)k(x',x')},\end{aligned}$$

where F_ψ is an integrable and symmetric function. By Assumption **(A4)**, $(x,x') \mapsto \sqrt{k(x,x)k(x',x')}$ is also an integrable function. Hence, h_θ is integrable. Since Θ is compact, it then follows from [191, Lemma 2.4] that $\sup_{\theta\in\Theta} |\widehat{\mathbb{M}}_n^2(\theta) - \mathbb{M}^2(\theta)| \xrightarrow{\mathrm{P}} 0$ and $\theta \mapsto \mathbb{M}^2(\theta)$ is continuous. □

[191]: Newey et al. (1994), *Large sample estimation and hypothesis testing*

Now, we are in the position to present the proof of Theorem E.2.1.

Proof of Theorem E.2.1. By Assumption **(A3)** and Theorem 6.3.3, $\mathbb{M}^2(\theta) = 0$ if and only if $\theta = \theta_0$. Thus $\mathbb{M}^2(\theta)$ is uniquely minimized at θ_0. Since Θ is compact, $\mathbb{M}^2(\theta)$ is continuous and $\widehat{\mathbb{M}}_n^2(\theta)$ converges uniformly in probability to $\mathbb{M}^2(\theta)$ by Lemma E.4.1. Then, $\hat{\theta}_n \xrightarrow{\mathrm{P}} \theta_0$ by [191, Theorem 2.1]. □

[191]: Newey et al. (1994), *Large sample estimation and hypothesis testing*

E.4.7. Proof of Theorem 6.4.1

Proof. First, we need to check that $\sigma_h^2 \ne 0$ when $\theta \ne \theta_0$ and $\sigma_h^2 = 0$ when $\theta = \theta_0$. Then, the results follow directly from [32, Sec. 5.5.1 and Sec. 5.5.2].

[32]: Serfling (1980), *Approximation theorems of mathematical statistics*

Note that

$$\begin{aligned}\mathbb{E}_{u'}[h_\theta(u,u')] &= \mathbb{E}_{u'}[\langle \xi_\theta(u), \xi_\theta(u')\rangle_{\mathcal{F}^q}] \\ &= \langle \xi_\theta(u), \mathbb{E}_{u'}[\xi_\theta(u')]\rangle_{\mathcal{F}^q} \\ &= \langle \xi_\theta(u), \mu_\theta\rangle_{\mathcal{F}^q} \\ &= M_\theta \xi_\theta(u).\end{aligned}$$

When $\theta = \theta_0$, it follows that $\mathbb{E}_{u'}[h_{\theta_0}(u,u')] = 0$ by Lemma 6.3.1, and hence $\sigma_h^2 = 0$.

Next, suppose that $\theta \neq \theta_0$. Then, $\mathbb{E}_{u'}[h_\theta(u,u')] = M_\theta \xi_\theta(u) =: c(u)$. Since $\sigma_h^2 = \mathrm{Var}_u[c(u)] = \mathbb{E}_u[(c(u) - \mathbb{E}_{u'}[c(u')])^2]$, $\sigma_h^2 = 0$ if and only if $c(u)$ is a constant function. Note that we can write $c(u) = c(x,z) = \mathbb{E}_{X'Z'}[\psi(Z';\theta)^\top \psi(z;\theta) k(x,X')]$. Therefore, by Assumptions **(A3)** and **(A4)**, $c(u)$ cannot be a constant function, implying that $\sigma_h^2 > 0$. □

E.4.8. Proof of Theorem 6.5.1

Proof. Since the kernel $k(x,x') = \varphi(x - x')$ is a shift-invariant kernel on $\mathbb{R}^d$, it follows from Theorem 6.2.1 that

$$\varphi(x - x') = (2\pi)^{-d/2} \int_{\mathbb{R}^d} e^{-i(x-x')^\top \omega} \, \mathrm{d}\Lambda(\omega).$$

Therefore, we can express $\mathbb{M}^2(\theta)$ as

$$\begin{aligned}\mathbb{M}^2(\theta) &= \mathbb{E}[\psi(Z;\theta)^\top \psi(Z';\theta) k(X,X')] \\ &= \mathbb{E}[\psi(Z;\theta)^\top \psi(Z';\theta) \varphi(X - X')] \\ &= (2\pi)^{-d/2} \mathbb{E}\left[\psi(Z;\theta)^\top \psi(Z';\theta) \left(\int_{\mathbb{R}^d} e^{-i(X-X')^\top \omega} \, \mathrm{d}\Lambda(\omega)\right)\right] \\ &= (2\pi)^{-d/2} \mathbb{E}\left[\psi(Z;\theta)^\top \psi(Z';\theta) \left(\int_{\mathbb{R}^d} e^{-i\omega^\top X} \cdot e^{i\omega^\top X'} \, \mathrm{d}\Lambda(\omega)\right)\right] \\ &= (2\pi)^{-d/2} \mathbb{E}\left[\int_{\mathbb{R}^d} \psi(Z;\theta)^\top \psi(Z';\theta) e^{-i\omega^\top X} e^{i\omega^\top X'} \, \mathrm{d}\Lambda(\omega)\right] \\ &= (2\pi)^{-d/2} \mathbb{E}\left[\int_{\mathbb{R}^d} \left[\psi(Z;\theta) e^{-i\omega^\top X}\right]^\top \left[\psi(Z';\theta) e^{i\omega^\top X'}\right] \, \mathrm{d}\Lambda(\omega)\right] \\ &= (2\pi)^{-d/2} \int_{\mathbb{R}^d} \mathbb{E}\left[\psi(Z;\theta) e^{-i\omega^\top X}\right]^\top \mathbb{E}\left[\psi(Z';\theta) e^{i\omega^\top X'}\right] \, \mathrm{d}\Lambda(\omega) \\ &= (2\pi)^{-d/2} \int_{\mathbb{R}^d} \left\| \mathbb{E}\left[\psi(Z;\theta) \exp(i\omega^\top X)\right] \right\|_2^2 \, \mathrm{d}\Lambda(\omega).\end{aligned}$$

This completes the proof. □

Bibliography

[1] E. L. Lehmann and J. P. Romano. *Testing statistical hypotheses*. Third. Springer Texts in Statistics. Springer, 2005 (cited on pages 3, 11, 12, 21, 32).

[2] W. Fithian, D. Sun, and J. Taylor. *Optimal Inference After Model Selection*. arXiv:1410.2597v4. 2017 (cited on pages 3, 22).

[3] Y. LeCun, C. Cortes, and C. Burges. 'MNIST handwritten digit database'. In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010) (cited on pages 4, 29, 54, 98).

[4] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001 (cited on pages 4, 15, 58–60, 99, 117).

[5] A. Gretton et al. 'A kernel two-sample test'. In: *Journal of Machine Learning Research* 13 (2012), pp. 723–773 (cited on pages 4, 16, 17, 19, 20, 22, 24, 32–34, 37, 38, 41, 45, 61, 69, 77, 104, 111).

[6] A. Gretton et al. 'Optimal kernel choice for large-scale two-sample tests'. In: *NeurIPS*. 2012 (cited on pages 4, 5, 18, 20, 21, 23, 24, 27–31, 33, 34, 43, 94, 97–99).

[7] D. J. Sutherland et al. 'Generative models and model criticism via optimized maximum mean discrepancy'. In: *ICLR*. 2017 (cited on pages 4, 6, 18–20, 27, 33–35, 38, 41, 51, 104, 111, 112).

[8] F. Liu et al. 'Learning Deep Kernels for Non-Parametric Two-Sample Tests'. In: *ICML*. 2020 (cited on pages 4, 6, 19, 20, 23, 27, 33–35, 38, 39, 42–44, 46, 47, 51, 52, 54, 85, 104, 106, 107, 111, 112).

[9] D. Lopez-Paz and M. Oquab. 'Revisiting Classifier Two-Sample Tests'. In: *ICLR*. 2017 (cited on pages 4, 18, 20, 27, 28, 33, 37, 38, 42, 46, 47, 51, 52).

[10] J. M. Kübler, K. Muandet, and B. Schölkopf. 'Quantum mean embedding of probability distributions'. In: *Phys. Rev. Research* 1 (2019) (cited on pages 5, 8).

[11] J. M. Kübler et al. 'An adaptive optimizer for measurement-frugal variational algorithms'. In: *Quantum* 4 (2020) (cited on pages 5, 7, 9).

[12] J. M. Kübler*, S. Buchholz*, and B. Schölkopf. 'The inductive bias of quantum kernels'. In: *NeurIPS*. 2021 (cited on pages 5, 7, 8, 58).

[13] S. Jerbi et al. 'Quantum machine learning beyond kernel methods'. In: *Nature Communications* 14.1 (2023), p. 517 (cited on pages 5, 7, 9).

[14] K. Chwialkowski, H. Strathmann, and A. Gretton. 'A Kernel Test of Goodness of Fit'. In: *ICML*. 2016 (cited on pages 5, 20, 22, 69, 77, 79, 80).

[15] A. Gretton et al. 'A kernel statistical test of independence'. In: *NeurIPS*. 2007 (cited on page 5).

[16] K. Zhang et al. 'Kernel-based conditional independence test and application in causal discovery'. In: *UAI*. 2011 (cited on page 5).

[17] J. D. Lee et al. 'Exact post-selection inference, with application to the lasso'. In: *Ann. Statist.* 44.3 (June 2016), pp. 907–927 (cited on pages 5, 24, 26–28, 88, 95).

[18] S. Rabanser, S. Günnemann, and Z. Lipton. 'Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift'. In: *NeurIPS*. 2019 (cited on pages 6, 45, 51–55, 86, 115).

[19] K. Muandet et al. *Kernel Mean Embedding of Distributions: A Review and Beyond*. Vol. 10. Foundations and Trends in Machine Learning. 2017, pp. 1–141 (cited on pages 7, 15, 34, 39, 59, 60, 74, 78, 104, 120).

[20] J. M. Kübler et al. 'Learning Kernel Tests Without Data Splitting'. In: *NeurIPS*. 2020 (cited on page 8).

[21] J. M. Kübler et al. 'A Witness Two-Sample Test'. In: *AISTATS*. 2022 (cited on pages 8, 13).

[22] J. M. Kübler et al. 'AutoML Two-Sample Test'. In: *NeurIPS*. 2022 (cited on pages 8, 13).

[23] K. Muandet, W. Jitkrittum, and J. Kübler. 'Kernel Conditional Moment Test via Maximum Moment Restriction'. In: *UAI*. 2020 (cited on page 8).

[24] L. Gresele* et al. 'Causal Inference Through the Structural Causal Marginal Problem'. In: *ICML*. 2022 (cited on page 8).

[25] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006 (cited on page 13).

[26] B. Phipson and G. K. Smyth. 'Permutation P-values should never be zero: calculating exact P-values when permutations are randomly drawn'. In: *Statistical applications in genetics and molecular biology* 9.1 (2010) (cited on page 13).

[27] K. M. Borgwardt et al. 'Integrating structured biological data by Kernel Maximum Mean Discrepancy'. In: *Bioinformatics* 22.14 (2006), pp. 49–57 (cited on pages 16, 32, 61).

[28] A. Smola et al. 'A Hilbert space embedding for distributions'. In: *Algorithmic Learning Theory*. 2007, pp. 13–31 (cited on pages 16, 59).

[29] A. Gretton et al. 'A kernel method for the two-sample-problem'. In: *NeurIPS*. 2006 (cited on page 16).

[30] A. Müller. 'Integral Probability Metrics and Their Generating Classes of Functions'. In: *Advances in Applied Probability* 29.2 (1997), pp. 429–443 (cited on page 16).

[31] K. Fukumizu et al. 'Kernel Measures of Conditional Dependence'. In: *NeurIPS*. 2008 (cited on pages 16, 60).

[32] R. J. Serfling. *Approximation theorems of mathematical statistics*. John Wiley & Sons, 1980 (cited on pages 17, 19, 21, 36, 76, 102, 104, 125).

[33] I. Kim et al. 'Classification accuracy as a proxy for two-sample testing'. In: *The Annals of Statistics* 49.1 (2021), pp. 411–434 (cited on pages 18, 20, 27, 33, 37, 42, 46, 52).

[34] B. K. Sriperumbudur et al. 'Kernel Choice and Classifiability for RKHS Embeddings of Probability Distributions'. In: *NeurIPS*. 2009 (cited on pages 18, 27).

[35] Q. Liu, J. Lee, and M. Jordan. 'A Kernelized Stein Discrepancy for Goodness-of-fit Tests'. In: *ICML*. 2016 (cited on pages 20, 22, 69, 77, 79, 80).

[36] M. Scetbon and G. Varoquaux. 'Comparing distributions: L1 geometry improves kernel two-sample testing'. In: *NeurIPS*. 2019 (cited on pages 20, 23, 27).

[37] W. Jitkrittum et al. 'Interpretable Distribution Features with Maximum Testing Power'. In: *NeurIPS*. 2016 (cited on pages 20, 23, 24, 27, 34, 42, 43, 51).

[38] W. Jitkrittum et al. 'Informative Features for Model Comparison'. In: *NeurIPS*. 2018 (cited on pages 20, 24, 27).

[39] W. Jitkrittum et al. 'A Linear-Time Kernel Goodness-of-Fit Test'. In: *NeurIPS*. 2017 (cited on pages 20, 24, 27).

[40] W. Jitkrittum, Z. Szabó, and A. Gretton. 'An Adaptive Test of Independence with Analytic Kernel Embeddings'. In: *ICML*. 2017 (cited on pages 20, 24, 27).

[41] X. Cheng and A. Cloninger. 'Classification Logit Two-sample Testing by Neural Networks'. In: *arXiv:1909.11298* (2019) (cited on pages 20, 27, 28, 33, 37, 42).

[42] M. Kirchler et al. 'Two-sample Testing Using Deep Learning'. In: *AISTATS*. 2020 (cited on pages 20, 27, 28, 33, 35, 42).

[43] R. Fisher. *The design of experiments*. Oliver and Boyd, 1935 (cited on page 20).

[44] T. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley Series in Probability and Statistics. Wiley, 2003 (cited on page 21).

[45] B. K. Sriperumbudur et al. 'Hilbert Space Embeddings and Metrics on Probability Measures'. In: *Journal of Machine Learning Research* 11 (2010), pp. 1517–1561 (cited on pages 22, 34, 40, 60).

[46] A. Wald. 'Tests of Statistical Hypotheses Concerning Several Parameters When the Number of Observations is Large'. In: *Transactions of the American Mathematical Society* 54.3 (1943), pp. 426–482 (cited on pages 25, 28).

[47] E. Jones, T. Oliphant, P. Peterson, et al. *SciPy: Open source scientific tools for Python*. 2001 (cited on page 27).

[48] J. H. Friedman. 'On multivariate goodness of fit and two sample testing'. In: *Stanford Linear Accelerator Center–PUB–10325* (2003). Ed. by L. Lyons, R. Mount, and R. Reitmeyer (cited on pages 27, 33, 42, 52).

[49] H. Cai, B. Goggin, and Q. Jiang. 'Two-sample test based on classification probability'. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 13.1 (2020), pp. 5–13 (cited on pages 27, 28, 42, 46, 52).

[50] J. Taylor and R. J. Tibshirani. 'Statistical learning and selective inference'. In: *Proceedings of the National Academy of Sciences* 112.25 (2015), pp. 7629–7634 (cited on page 28).

[51] R. Tibshirani. 'Regression Shrinkage and Selection via the Lasso'. In: *Journal of the Royal Statistical Society (Series B)* 58 (1996), pp. 267–288 (cited on page 28).

[52] M. Yamada et al. 'Post Selection Inference with Kernels'. In: *AISTATS*. 2018 (cited on page 28).

[53] L. Slim et al. 'kernelPSI: a Post-Selection Inference Framework for Nonlinear Variable Selection'. In: *ICML*. 2019 (cited on page 28).

[54] M. Yamada et al. 'Post Selection Inference with Incomplete Maximum Mean Discrepancy Estimator'. In: *ICLR*. 2019 (cited on page 28).

[55] J. N. Lim et al. 'Kernel Stein Tests for Multiple Model Comparison'. In: *NeurIPS*. 2019 (cited on page 28).

[56] S. Janson. 'The asymptotic distributions of incomplete U-statistics'. In: *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 66.4 (Sept. 1984), pp. 495–505 (cited on page 28).

[57] W. Zaremba, A. Gretton, and M. Blaschko. 'B-test: A non-parametric, low variance kernel two-sample test'. In: *NeurIPS*. 2013, pp. 755–763 (cited on page 28).

[58] A. Kudo. 'A multivariate analogue of the one-sided test'. In: *Biometrika* 50.3/4 (1963), pp. 403–418 (cited on page 28).

[59] A. Shapiro. 'Towards a Unified Theory of Inequality Constrained Testing in Multivariate Analysis'. In: *International Statistical Review / Revue Internationale de Statistique* 56.1 (1988), pp. 49–62 (cited on page 28).

[60] J. H. Friedman and L. C. Rafsky. 'Multivariate Generalizations of the Wald-Wolfowitz and Smirnov Two-Sample Tests'. In: *The Annals of Statistics* 7.4 (1979), pp. 697–717 (cited on page 32).

[61] H. Chen and J. H. Friedman. 'A New Graph-Based Two-Sample Test for Multivariate and Object Data'. In: *Journal of the American Statistical Association* 112.517 (2017), pp. 397–409 (cited on page 32).

[62] Z. Harchaoui, F. R. Bach, and E. Moulines. 'Testing for homogeneity with kernel Fisher discriminant analysis'. In: *NeurIPS*. 2008 (cited on pages 32, 41).

[63] M. Fromont et al. 'Kernels Based Tests with Non-asymptotic Bootstrap Approaches for Two-sample Problems'. In: *COLT*. 2012 (cited on page 33).

[64] M. Fromont, B. Laurent, and P. Reynaud-Bouret. 'The two-sample problem for Poisson processes: Adaptive tests with a nonasymptotic wild bootstrap approach'. In: *The Annals of Statistics* 41.3 (2013), pp. 1431–1461 (cited on page 33).

[65] A. Schrab et al. 'MMD Aggregated Two-Sample Test'. In: *arXiv:2110.15073* (2021) (cited on pages 33, 51, 54, 84, 85).

[66] S. Mika et al. 'Fisher discriminant analysis with kernels'. In: *Neural Networks for Signal Processing IX*. 1999, pp. 41–48 (cited on page 39).

[67] S. Mika. 'Kernel Fisher Discriminants'. Doctoral Thesis. Berlin: Technische Universität Berlin, 2003 (cited on pages 39, 48, 102, 103).

[68] B. Schölkopf, R. Herbrich, and A. J. Smola. 'A Generalized Representer Theorem'. In: *COLT*. 2001 (cited on page 40).

[69] A. Rudi, L. Carratino, and L. Rosasco. 'FALKON: An Optimal Large Scale Kernel Method'. In: *NeurIPS*. 2017 (cited on pages 40, 107, 108).

[70] G. Meanti et al. 'Kernel Methods Through the Roof: Handling Billions of Points Efficiently'. In: *NeurIPS*. 2020 (cited on pages 40, 107, 108).

[71] C. K. I. Williams and M. W. Seeger. 'Using the Nyström Method to Speed Up Kernel Machines'. In: *NeurIPS*. 2000 (cited on pages 40, 61).

[72] A. Chatalic et al. 'Nyström Kernel Mean Embeddings'. In: *arXiv:2201.13055* (2022) (cited on page 40).

[73] K. Chwialkowski et al. 'Fast Two-Sample Testing with Analytic Representations of Probability Measures'. In: *NeurIPS*. 2015 (cited on pages 41, 42).

[74] H. Hotelling. 'The Generalization of Student's Ratio'. In: *The Annals of Mathematical Statistics* 2.3 (1931), pp. 360–378 (cited on page 42).

[75] K. Balasubramanian, T. Li, and M. Yuan. 'On the Optimality of Kernel-Embedding Based Goodness-of-Fit Tests'. In: *Journal of Machine Learning Research* 22.1 (2021), pp. 1–45 (cited on page 42).

[76] Z. Harchaoui, F. Bach, and E. Moulines. 'Testing for Homogeneity with Kernel Fisher Discriminant Analysis'. In: *arXiv:0804.1026* (2008) (cited on page 42).

[77] T. Li and M. Yuan. 'On the Optimality of Gaussian Kernel Based Nonparametric Tests against Smooth Alternatives'. In: *arXiv:1909.03302* (2019) (cited on page 42).

[78] P. Baldi, P. Sadowski, and D. Whiteson. 'Searching for exotic particles in high-energy physics with deep learning'. In: *Nature communications* 5.1 (2014), pp. 1–9 (cited on pages 44, 106).

[79] Student. 'The probable error of a mean'. In: *Biometrika* (1908), pp. 1–25 (cited on page 45).

[80] B. L. Welch. 'The generalization of 'STUDENT'S'problem when several different population varlances are involved'. In: *Biometrika* 34.1-2 (1947), pp. 28–35 (cited on page 45).

[81] P. Golland and B. Fischl. 'Permutation Tests for Classification: Towards Statistical Significance in Image-Based Studies'. In: *Information Processing in Medical Imaging*. Ed. by C. Taylor and J. A. Noble. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 330–341 (cited on pages 45–47, 52).

[82] A. Gretton et al. 'Measuring statistical dependence with Hilbert-Schmidt norms'. In: *International conference on algorithmic learning theory*. Springer. 2005, pp. 63–77 (cited on page 45).

[83] Z. Lipton, Y.-X. Wang, and A. Smola. 'Detecting and Correcting for Label Shift with Black Box Predictors'. In: *ICML*. 2018 (cited on pages 45, 52).

[84] L. Koch et al. 'Hidden in Plain Sight: Subgroup Shifts Escape OOD Detection'. In: *Medical Imaging in Deep Learning*. 2022 (cited on page 45).

[85] S. Hediger, L. Michel, and J. Näf. 'On the use of random forest for two-sample testing'. In: *Computational Statistics and Data Analysis* 170 (2022), p. 107435 (cited on pages 46, 52).

[86] M. Feurer et al. 'Efficient and Robust Automated Machine Learning'. In: *NeurIPS*. 2015 (cited on page 46).

[87] F. Hutter, L. Kotthoff, and J. Vanschoren. *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019 (cited on page 46).

[88] X. He, K. Zhao, and X. Chu. 'AutoML: A survey of the state-of-the-art'. In: *Knowledge-Based Systems* 212 (2021), p. 106622 (cited on page 46).

[89] T. G. Dietterich. 'Ensemble methods in machine learning'. In: *International workshop on multiple classifier systems*. Springer. 2000, pp. 1–15 (cited on page 46).

[90] N. Erickson et al. 'Autogluon-tabular: Robust and accurate automl for structured data'. In: *arXiv:2003.06505* (2020) (cited on pages 47, 52).

[91] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern classification, 2nd Edition*. Wiley, 2001 (cited on page 48).

[92] X. Mao et al. 'On the Effectiveness of Least Squares Generative Adversarial Networks'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.12 (Dec. 2019), pp. 2947–2960 (cited on page 49).

[93] S. Zhao et al. 'Comparing Distributions by Measuring Differences that Affect Decision Making'. In: *ICLR*. 2022 (cited on pages 52, 86).

[94] A. Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*. 2009 (cited on page 54).

[95] I. Goodfellow, J. Shlens, and C. Szegedy. 'Explaining and Harnessing Adversarial Examples'. In: *ICLR*. 2015 (cited on page 54).

[96] C. Steinruecken et al. 'The automatic statistician'. In: *Automated Machine Learning*. Springer, Cham, 2019, pp. 161–173 (cited on page 56).

[97] C. Cortes and V. Vapnik. 'Support-Vector Networks'. In: *Machine Learning* 20.3 (1995), pp. 273–297 (cited on page 58).

[98] I. Steinwart and A. Christmann. *Support Vector Machines*. Springer, 2008 (cited on pages 58, 71–73, 124).

[99] H. Hotelling. 'Analysis of a complex of statistical variables into principal components.' In: *Journal of Educational Psychology* 24.6 (1933), pp. 417–441 (cited on page 58).

[100] P. Rebentrost, M. Mohseni, and S. Lloyd. 'Quantum Support Vector Machine for Big Data Classification'. In: *Phys. Rev. Lett.* 113 (13 Sept. 2014), p. 130503 (cited on page 58).

[101] S. Lloyd, M. Mohseni, and P. Rebentrost. 'Quantum principal component analysis'. In: *Nature Physics* 10 (July 2014), pp. 631–633 (cited on page 58).

[102] S. Aaronson. 'Read the fine print'. In: *Nature Physics* 11 (Apr. 2015), pp. 291–29 (cited on page 58).

[103] C. Ciliberto et al. 'Quantum machine learning: A classical perspective'. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 474.2209 (2018), p. 20170551 (cited on page 58).

[104] R. Chatterjee and T. Yu. 'Generalized Coherent States, Reproducing Kernels, and Quantum Support Vector Machines'. In: *Quantum Info. Comput.* 17.15-16 (Dec. 2017), pp. 1292–1306 (cited on pages 58, 116).

[105] M. Schuld and N. Killoran. 'Quantum Machine Learning in Feature Hilbert Spaces'. In: *Phys. Rev. Lett.* 122 (4 Feb. 2019), p. 040504 (cited on pages 58, 61, 62, 65).

[106] V. Havlicek et al. 'Supervised learning with quantum-enhanced feature spaces'. In: *Nature* 567.7747 (2019), p. 209 (cited on pages 58, 61, 62).

[107] L. Cincio et al. 'Learning the quantum algorithm for state overlap'. In: *New Journal of Physics* 20.11 (2018), p. 113022 (cited on pages 58, 65).

[108] B. Vlastakis et al. 'Deterministically Encoding Quantum Information Using 100-Photon Schrödinger Cat States'. In: *Science* 342.6158 (2013), pp. 607–610 (cited on pages 59, 64, 117).

[109] A. Berlinet and C. Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Springer, Boston, MA, 2004 (cited on page 59).

[110] N. Aronszajn. 'Theory of Reproducing Kernels'. In: *Transactions of the American Mathematical Society* 68.3 (1950), pp. 337–404 (cited on pages 59, 71).

[111] C.-J. Simon-Gabriel and B. Schölkopf. 'Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions'. In: *Journal of Machine Learning Research* 19.44 (2018), pp. 1–29 (cited on pages 60, 71, 116, 117).

[112] I. Steinwart. 'On the Influence of the Kernel on the Consistency of Support Vector Machines'. In: *Journal of Machine Learning Research* 2 (2001), pp. 67–93 (cited on page 60).

[113] K. Fukumizu, F. R. Bach, and M. I. Jordan. 'Dimensionality Reduction for Supervised Learning with Reproducing Kernel Hilbert Spaces'. In: *Journal of Machine Learning Research* 5 (2004), pp. 73–99 (cited on page 60).

[114] K. Muandet et al. 'Learning from Distributions via Support Measure Machines'. In: *NeurIPS*. 2012 (cited on page 60).

[115] K. Muandet and B. Schölkopf. 'One-class Support Measure Machines for Group Anomaly Detection'. In: *UAI*. 2013 (cited on page 60).

[116] D. Lopez-Paz et al. 'Towards a Learning Theory of Cause-Effect Inference'. In: *ICML*. 2015 (cited on page 60).

[117] Z. Szabó et al. 'Learning Theory for Distribution Regression'. In: *Journal of Machine Learning Research* 17.152 (2016), pp. 1–40 (cited on page 60).

[118] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani. 'Training Generative Neural Networks via Maximum Mean Discrepancy Optimization'. In: *UAI*. 2015 (cited on page 61).

[119] Y. Li, K. Swersky, and R. Zemel. 'Generative moment matching networks'. In: *ICML*. 2015 (cited on page 61).

[120] C.-L. Li et al. 'MMD GAN: Towards deeper understanding of moment matching network'. In: *NeurIPS*. 2017 (cited on page 61).

[121] Y. LeCun, Y. Bengio, and G. Hinton. 'Deep Learning'. In: *Nature* 521 (May 2015), pp. 436–44 (cited on page 61).

[122] A. Rahimi and B. Recht. 'Random Features for Large-Scale Kernel Machines'. In: *NeurIPS*. 2008 (cited on page 61).

[123] B. Coyle et al. *The Born supremacy: quantum advantage and training of an Ising Born machine*. 2020 (cited on page 62).

[124] S. Srinivasan, C. Downey, and B. Boots. 'Learning and Inference in Hilbert Space with Quantum Graphical Models'. In: *NeurIPS*. 2018 (cited on page 62).

[125] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. 10th anniv. Cambridge: Cambridge University Press, 2010 (cited on page 62).

[126] S. Deléglise et al. 'Reconstruction of non-classical cavity field states with snapshots of their decoherence'. In: *Nature* 455 (Sept. 2008), pp. 510–514 (cited on pages 64, 117).

[127] A. Ourjoumtsev et al. 'Generation of optical 'Schrödinger cats' from photon number states'. In: *Nature* 448 (Aug. 2007), pp. 784–786 (cited on pages 64, 117).

[128] U. L. Andersen et al. 'Hybrid discrete-and continuous-variable quantum information'. In: *Nature Physics* 11 (Sept. 2015), pp. 713–719 (cited on page 64).

[129] T. Theurer et al. 'Resource Theory of Superposition'. In: *Physical Review Letters* 119 (2017), p. 230401 (cited on page 64).

[130] A. Streltsov, G. Adesso, and M. B. Plenio. 'Colloquium: Quantum coherence as a resource'. In: *Rev. Mod. Phys.* 89 (4 Oct. 2017), p. 041003 (cited on page 64).

[131] U. Alvarez-Rodriguez et al. 'The Forbidden Quantum Adder'. In: *Scientific Reports* 5 (2015), p. 11983 (cited on page 64).

[132] M. Oszmaniec et al. 'Creating a Superposition of Unknown Quantum States'. In: *Physical Review Letters* 116 (2016), p. 110403 (cited on page 64).

[133] V. Giovannetti, S. Lloyd, and L. Maccone. 'Quantum Random Access Memory'. In: *Phys. Rev. Lett.* 100 (16 Apr. 2008), p. 160501 (cited on page 64).

[134] H. Buhrman et al. 'Quantum Fingerprinting'. In: *Phys. Rev. Lett.* 87 (16 Sept. 2001), p. 167902 (cited on page 64).

[135] R. Filip. 'Overlap and entanglement-witness measurements'. In: *Phys. Rev. A* 65 (6 June 2002), p. 062320 (cited on page 65).

[136] K. L. Pregnell. 'Measuring Nonlinear Functionals of Quantum Harmonic Oscillator States'. In: *Phys. Rev. Lett.* 96 (6 Feb. 2006), p. 060501 (cited on page 65).

[137] H. Jeong et al. 'Detecting the degree of macroscopic quantumness using an overlap measurement'. In: *J. Opt. Soc. Am. B* 31.12 (Dec. 2014), pp. 3057–3066 (cited on page 65).

[138] W. Newey. 'Efficient estimation of models with conditional moment restrictions'. In: *Handbook of Statistics*. Vol. 11. 1993. Chap. 16, pp. 419–454 (cited on pages 68, 70, 120).

[139] C. Ai and X. Chen. 'Efficient Estimation of Models with Conditional Moment Restrictions Containing Unknown Functions'. In: *Econometrica* 71.6 (2003), pp. 1795–1843 (cited on page 68).

[140] J. Muth. 'Rational Expectations and the Theory of Price Movements'. In: *Econometrica* 29.3 (1961), pp. 315–335 (cited on page 68).

[141] S. Athey, J. Tibshirani, and S. Wager. 'Generalized random forests'. In: *The Annals of Statistics* 47.2 (Apr. 2019), pp. 1148–1178 (cited on page 68).

[142] M. Oprescu, V. Syrgkanis, and Z. S. Wu. 'Orthogonal Random Forest for Causal Inference'. In: *ICML*. 2019 (cited on page 68).

[143] V. Chernozhukov et al. 'Double/debiased machine learning for treatment and structural parameters'. In: *The Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on page 68).

[144] A. Bennett, N. Kallus, and T. Schnabel. 'Deep Generalized Method of Moments for Instrumental Variable Analysis'. In: *NeurIPS*. 2019 (cited on pages 68, 72, 73, 119, 120).

[145] G. Lewis and V. Syrgkanis. 'Adversarial Generalized Method of Moments'. In: *ArXiv:1803.07164* (2018) (cited on pages 68, 72, 73, 119, 120).

[146] J. Hartford et al. 'Deep IV: A Flexible Approach for Counterfactual Prediction'. In: *ICML*. 2017 (cited on page 68).

[147] R. Singh, M. Sahani, and A. Gretton. 'Kernel Instrumental Variable Regression'. In: *NeurIPS*. 2019 (cited on pages 68, 119, 120).

[148] K. Muandet et al. 'Dual instrumental variable regression'. In: *NeurIPS*. 2020 (cited on pages 68, 119, 120).

[149] L. P. Hansen. 'Large Sample Properties of Generalized Method of Moments Estimators'. In: *Econometrica* 50.4 (1982), pp. 1029–1054 (cited on page 68).

[150] A. Hall. *Generalized Method of Moments*. Advanced texts in econometrics. Oxford University Press, 2005 (cited on pages 68, 72, 80, 119).

[151] J. Sargan. 'The Estimation of Economic Relationships using Instrumental Variables'. In: *Econometrica* 26.3 (1958), pp. 393–415 (cited on page 68).

[152] H. Bierens. *Econometric Model Specification: Consistent Model Specification Tests and Semi-nonparametric Modeling and Inference*. World Scientific, 2017 (cited on page 68).

[153] W. Newey. 'Maximum Likelihood Specification Testing and Conditional Moment Tests'. In: *Econometrica* 53.5 (1985), pp. 1047–1070 (cited on pages 68, 77).

[154] G. Tauchen. 'Diagnostic testing and evaluation of maximum likelihood models'. In: *Journal of Econometrics* 30.1 (1985), pp. 415–443 (cited on pages 68, 69, 77).

[155] J. Hausman. 'Specification Tests in Econometrics'. In: *Econometrica* 46.6 (1978), pp. 1251–71 (cited on pages 68, 70).

[156] H. White. 'Consequences and Detection of Misspecified Nonlinear Regression Models'. In: *Journal of the American Statistical Association* 76.374 (1981), pp. 419–433 (cited on page 68).

[157] R. de Jong. 'The Bierens test under data dependence'. In: *Journal of Econometrics* 72.1-2 (1996), pp. 1–32 (cited on pages 69, 77).

[158] S. Donald, G. Imbens, and W. Newey. 'Empirical Likelihood Estimation and Consistent Tests with Conditional Moment Restrictions'. In: *Journal of Econometrics* 117.1 (2003), pp. 55–93 (cited on pages 69, 77).

[159] H. Bierens. 'Consistent model specification tests'. In: *Journal of Econometrics* 20.1 (1982), pp. 105–134 (cited on pages 69, 74, 77, 78).

[160] H. Bierens and W. Ploberger. 'Asymptotic Theory of Integrated Conditional Moment Tests'. In: *Econometrica* 65.5 (1997), pp. 1129–1152 (cited on pages 69, 77, 78).

[161] J. Zheng. 'A consistent test of functional form via nonparametric estimation techniques'. In: *Journal of Econometrics* 75.2 (1996), pp. 263–289 (cited on pages 69, 77, 79).

[162] Q. Li and S. Wang. 'A simple consistent bootstrap test for a parametric regression function'. In: *Journal of Econometrics* 87.1 (1998), pp. 145–165 (cited on pages 69, 77, 79).

[163] M. Delgado, M. Domínguez, and P. Lavergne. 'Consistent Tests of Conditional Moment Restrictions'. In: *Annales d'Économie et de Statistique* 81 (2006), pp. 33–67 (cited on pages 69, 77, 78, 80, 81).

[164] G. Tripathi and Y. Kitamura. 'Testing conditional moment restrictions'. In: *The Annals of Statistics* 31.6 (2003), pp. 2059–2095 (cited on pages 69, 79).

[165] M. Carrasco and J.-P. Florens. 'Generalization of GMM to a Continuum of Moment Conditions'. In: *Econometric Theory* 16.6 (2000), pp. 797–834 (cited on pages 69, 78, 119).

[166] M. Álvarez, L. Rosasco, and N. Lawrence. 'Kernels for Vector-Valued Functions: A Review'. In: *Foundation and Trends in Machine Learning* 4.3 (2012), pp. 195–266 (cited on page 73).

[167] K. Khosravi, G. Lewis, and V. Syrgkanis. 'Non-Parametric Inference Adaptive to Intrinsic Dimension'. In: *ArXiv:1901.03719* (2019) (cited on page 74).

[168] I. Tolstikhin, B. Sriperumbudur, and K. Muandet. 'Minimax Estimation of Kernel Mean Embeddings'. In: *Journal of Machine Learning Research* 18 (2017), 86:1–86:47 (cited on page 74).

[169] M. Arcones and E. Gine. 'On the Bootstrap of U and V Statistics'. In: *The Annals of Statistics* 20.2 (June 1992), pp. 655–674 (cited on pages 76, 77).

[170] M. Huskova and P. Janssen. 'Consistency of the Generalized Bootstrap for Degenerate U-Statistics'. In: *The Annals of Statistics* 21.4 (Dec. 1993), pp. 1811–1823 (cited on page 77).

[171] H. Bierens. 'A Consistent Conditional Moment Test of Functional Form'. In: *Econometrica* 58.6 (1990), pp. 1443–1458 (cited on pages 77, 78).

[172] Y. Fan and Q. Li. 'Consistent model specification tests: Kernel-Based tests versus Bierens' ICM tests'. In: *Econometric Theory* 16 (Dec. 2000), pp. 1016–1041 (cited on pages 77, 79).

[173] M. Stinchcombe and H. White. 'Consistent Specification Testing with Nuisance Parameters Present Only under the Alternative'. In: *Econometric Theory* 14.3 (1998), pp. 295–325 (cited on page 78).

[174] J. C. Escanciano. 'A Consistent Diagnostic Test for Regression Models Using Projections'. In: *Econometric Theory* 22.6 (2006), pp. 1030–1051 (cited on page 78).

[175] Y. Kitamura, G. Tripathi, and H. Ahn. 'Empirical Likelihood-Based Inference in Conditional Moment Restriction Models'. In: *Econometrica* 72.6 (2004), pp. 1667–1714 (cited on page 79).

[176] M. Dominguez and I. Lobato. 'Consistent Estimation of Models Defined by Conditional Moment Restrictions'. In: *Econometrica* 72.5 (2004), pp. 1601–1615 (cited on page 79).

[177] C. Stein. 'A bound for the error in the normal approximation to the distribution of a sum of dependent random variables'. In: *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*. University of California Press, 1972, pp. 583–602 (cited on page 79).

[178] C. Oates, M. Girolami, and N. Chopin. 'Control functionals for Monte Carlo integration'. In: *Journal of the Royal Statistical Society Series B* 79.3 (2017), pp. 695–718 (cited on page 79).

[179] W. Stute. 'Nonparametric model checks for regression'. In: *The Annals of Statistics* 25.2 (Apr. 1997), pp. 613–641 (cited on page 80).

[180] P. Lavergne and P. Nguimkeu. *A Hausman Specification Test of Conditional Moment Restrictions*. TSE Working Papers 16-743. Toulouse School of Economics (TSE), 2016 (cited on page 81).

[181] W. Newey. 'Efficient Instrumental Variables Estimation of Nonlinear Models'. In: *Econometrica* 58.4 (1990), pp. 809–837 (cited on page 81).

[182] I. Kim, S. Balakrishnan, and L. Wasserman. 'Minimax optimality of permutation tests'. In: *The Annals of Statistics* 50.1 (2022), pp. 225–251 (cited on page 84).

[183] A. Schrab et al. 'Efficient Aggregated Kernel Tests using Incomplete U-statistics'. In: *arXiv preprint arXiv:2206.09194* (2022) (cited on page 85).

[184] L. Vandenberghe. 'The CVXOPT linear and quadratic cone program solvers'. In: (2010) (cited on page 95).

[185] K. Fukumizu, F. R. Bach, and A. Gretton. 'Statistical Convergence of Kernel CCA'. In: *NeurIPS*. 2005 (cited on page 104).

[186] D. V. Strekalov and G. Leuchs. 'Nonlinear Interactions and Non-classical Light'. In: *Quantum Photonics: Pioneering Advances and Emerging Applications*. Ed. by R. Boyd, S. Lukishova, and V. Zadkov. Springer Nature Switzerland, 2019 (cited on pages 116, 117).

[187] G. S. Agarwal. *Quantum optics*. Cambridge University Press, 2012 (cited on page 117).

[188] J. Angrist and J.-S. Pischke. *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press, 2008 (cited on pages 119, 121).

[189] S. Smale and D.-X. Zhou. 'Learning Theory Estimates via Integral Operators and Their Approximations'. In: *Constructive Approximation* 26 (Aug. 2007), pp. 153–172 (cited on page 122).

[190] I. Pinelis. 'Optimum Bounds for the Distributions of Martingales in Banach Spaces'. In: *The Annals of Probability* 22.4 (Oct. 1994), pp. 1679–1706 (cited on page 122).

[191] W. Newey and D. McFadden. 'Large sample estimation and hypothesis testing'. In: vol. 4. Handbook of Econometrics. Elsevier, 1994, pp. 2111–2245 (cited on page 125).