Learning Deep Kernels for Non-Parametric Two-Sample Tests

Feng Liu [*1 2] **Wenkai Xu** [*2] **Jie Lu** [1] **Guangquan Zhang** [1] **Arthur Gretton** [2] **Danica J. Sutherland** [3]

Abstract

We propose a class of kernel-based two-sample tests, which aim to determine whether two sets of samples are drawn from the same distribution. Our tests are constructed from kernels parameterized by deep neural nets, trained to maximize test power. These tests adapt to variations in distribution smoothness and shape over space, and are especially suited to high dimensions and complex data. By contrast, the simpler kernels used in prior kernel testing work are spatially homogeneous, and adaptive only in lengthscale. We explain how this scheme includes popular classifier-based two-sample tests as a special case, but improves on them in general. We provide the first proof of consistency for the proposed adaptation method, which applies both to kernels on deep features and to simpler radial basis kernels or multiple kernel learning. In experiments, we establish the superior performance of our deep kernels in hypothesis testing on benchmark and real-world data. The code of our deep-kernel-based two sample tests is available at `github.com/fengliu90/DK-for-TST`.

1. Introduction

Two sample tests are hypothesis tests aiming to determine whether two sets of samples are drawn from the same distribution. Traditional methods such as t-tests and Kolmogorov-Smirnov tests are mainstays of statistical applications, but require strong parametric assumptions about the distributions being studied and/or are only effective on data in extremely low-dimensional spaces. A broad set of recent work in statistics and machine learning has focused on relaxing these assumptions, with methods either generally applicable or specific to various more complex domains (Gretton et al., 2012a; Székely & Rizzo, 2013; Heller & Heller, 2016; Jitkrittum et al., 2016; Ramdas et al., 2017; Lopez-Paz & Oquab, 2017; Chen & Friedman, 2017; Gao et al., 2018; Ghoshdastidar et al., 2017; Ghoshdastidar & von Luxburg, 2018; Li & Wang, 2018; Kirchler et al., 2020). These tests have also allowed application in various machine learning problems such as domain adaptation, generative modeling, and causal discovery (Binkowski et al., 2018; Gong et al., 2016; Stojanov et al., 2019; Lopez-Paz & Oquab, 2017).

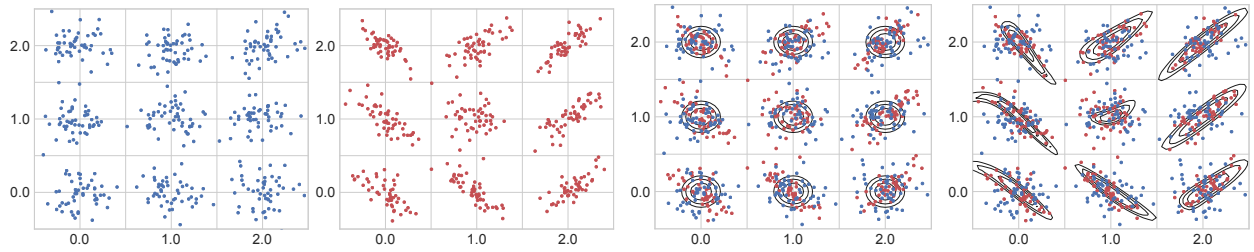
A popular class of non-parametric two-sample tests is based on kernel methods (Smola & Schölkopf, 2001): such tests construct a *kernel mean embedding* (Berlinet & Thomas-Agnan, 2004; Muandet et al., 2017) for each distribution, and measure the difference in these embeddings. For any *characteristic* kernel, two distributions are the same if and only if their mean embeddings are the same; the distance between mean embeddings is the *maximum mean discrepancy* (MMD) (Gretton et al., 2012a). There are also several closely related methods, including tests based on checking for differences in mean embeddings evaluated at specific locations (Chwialkowski et al., 2015; Jitkrittum et al., 2016) and kernel Fisher discriminant analysis (Harchaoui et al., 2007). These tests all work well for samples from simple distributions when using appropriate kernels.

Problems that we care about, however, often involve distributions with complex structure, where simple kernels will often map distinct distributions to nearby (and hence hard to distinguish) mean embeddings. Figure 1a shows an example of a multimodal dataset, where the overall modes align but the sub-mode structure varies differently at each mode. A translation-invariant Gaussian kernel only "looks at" the data uniformly within each mode (see Figure 1b), requiring many samples to correctly distinguish the two distributions. The distributions can be distinguished more effectively if we understand the structure of each mode, as with the more complex kernel illustrated in Figure 1c.

To model these complex functions, we adopt a *deep kernel* approach (Wilson et al., 2016; Sutherland et al., 2017; Li et al., 2017; Jean et al., 2018; Wenliang et al., 2019),

[*]Equal contribution [1]Australian Artificial Intelligence Institute, University of Technology Sydney, Sydney, NSW, Australia [2]Gatsby Computational Neuroscience Unit, University College London, London, UK [3]Toyota Technological Institute at Chicago, Chicago, IL, USA. Correspondence to: Feng Liu <Feng.Liu@uts.edu.au>, Wenkai Xu <wenkaix@gatsby.ucl.ac.uk>, Danica J. Sutherland <djs@djsutherland.ml>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.



(a) Samples drawn from $\mathbb{P}$ (left) and $\mathbb{Q}$ (right). (b) Contour of Gaussian (c) Contour of deep kernel

Figure 1. In the Blob dataset, $\mathbb{P}$ and $\mathbb{Q}$ are each equal mixtures of nine Gaussians with the same modes (a), but each component of $\mathbb{P}$ is an isotropic Gaussian whereas the covariance of $\mathbb{Q}$ differs in each component. Panels (b) and (c) show the contours of a kernel, $k(x, \mu_i)$ for each of the nine modes μ_i; contour values are 0.7, 0.8 and 0.9. A Gaussian kernel (b) treats points isotropically throughout the space, based only on $\|x - y\|$. A deep kernel (c) learned by our methods behaves differently in different parts of the space, adapting to the local structure of the data distributions and hence allowing better identification of differences between $\mathbb{P}$ and $\mathbb{Q}$.

building a kernel with a deep network. In this paper, we use

$$k_\omega(x, y) = [(1 - \epsilon)\kappa(\phi_\omega(x), \phi_\omega(y)) + \epsilon]q(x, y), \quad (1)$$

where the deep neural network ϕ_ω extracts features of samples, and κ is a simple kernel (e.g., a Gaussian) on those features, while q is a simple characteristic kernel (e.g. Gaussian) on the input space. With an appropriate choice of ϕ_ω, this allows for extremely flexible kernels which can learn complex behavior very different in different parts of space. This choice is discussed further in Section 5.

These complex kernels, though, cannot feasibly be specified by hand or simple heuristics, as is typical practice in kernel methods. We select the parameters ω by maximizing the ratio of the MMD to its variance, which maximizes test power at large sample sizes. This procedure was proposed by Sutherland et al. (2017), but we establish for the first time that it gives consistent selection of the best kernel in the class, whether optimizing our deep kernels with hundreds of thousands of parameters or simply choosing lengthscales of a Gaussian as did Sutherland et al. Previously, there were no guarantees this procedure would yield a kernel which generalized at all from the training set to a test set.

Another way to compare distributions is to train a classifier between them, and evaluate its accuracy (Lopez-Paz & Oquab, 2017). We show, perhaps surprisingly, that our framework encompasses this approach, but deep kernels allow for more general model classes which can use the data more efficiently. We also train representations directly to maximize test power, rather than a cross-entropy surrogate.

We test our method on several simulated and real-world datasets, including complex synthetic distributions, high-energy physics data, and challenging image problems. We find convincingly that learned deep kernels outperform simple shallow methods, and learning by maximizing test power outperforms learning through a cross-entropy surrogate loss.

2. MMD Two-Sample Tests

Two-sample testing. Let $\mathcal{X}$ be a separable metric space – in this paper, typically a subset of $\mathbb{R}^d$ – and $\mathbb{P}$, $\mathbb{Q}$ be Borel probability measures on $\mathcal{X}$. We observe independent identically distributed (*i.i.d.*) samples $S_\mathbb{P} = \{x_i\}_{i=1}^n \sim \mathbb{P}^n$ and $S_\mathbb{Q} = \{y_j\}_{j=1}^m \sim \mathbb{Q}^m$. We wish to know whether $S_\mathbb{P}$ and $S_\mathbb{Q}$ come from the same distribution: does $\mathbb{P} = \mathbb{Q}$?

We use the null hypothesis testing framework, where the null hypothesis $\mathfrak{H}_0 : \mathbb{P} = \mathbb{Q}$ is tested against the alternative hypothesis $\mathfrak{H}_1 : \mathbb{P} \neq \mathbb{Q}$. We perform a two-sample test in four steps: select a significance level $\alpha \in [0, 1]$; compute a test statistic $\hat{t}(S_\mathbb{P}, S_\mathbb{Q})$; compute the p-value $\hat{p} = \Pr_{\mathfrak{H}_0}(T > \hat{t})$, the probability of the two-sample test returning a statistic as large as $\hat{t}$ when $\mathfrak{H}_0$ is true; finally, reject $\mathfrak{H}_0$ if $\hat{p} < \alpha$.

Maximum mean discrepancy (MMD). We will base our two-sample test statistic on an estimate of a distance between distributions. Our metric, the MMD, is defined in terms of a kernel k giving point-level "similarities" on $\mathcal{X}$.

Definition 1 (Gretton et al., 2012a). *Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be the kernel of a reproducing kernel Hilbert space $\mathcal{H}_k$, with feature maps $k(\cdot, x) \in \mathcal{H}_k$. Let $X, X' \sim \mathbb{P}$ and $Y, Y' \sim \mathbb{Q}$, and define the* kernel mean embeddings $\mu_\mathbb{P} := \mathbb{E}[k(\cdot, X)]$ *and $\mu_\mathbb{Q} := \mathbb{E}[k(\cdot, Y)]$. Under mild integrability conditions,*

$$\mathrm{MMD}(\mathbb{P}, \mathbb{Q}; \mathcal{H}_k) := \sup_{f \in \mathcal{H}, \|f\|_{\mathcal{H}_k} \leq 1} |\mathbb{E}[f(X)] - \mathbb{E}[f(Y)]|$$

$$= \|\mu_\mathbb{P} - \mu_\mathbb{Q}\|_{\mathcal{H}_k} = \sqrt{\mathbb{E}\left[k(X, X') + k(Y, Y') - 2k(X, Y)\right]}.$$

For characteristic *kernels, $\mu_\mathbb{P} = \mu_\mathbb{Q}$ implies $\mathbb{P} = \mathbb{Q}$, hence $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}; \mathcal{H}_k) = 0$ if and only if $\mathbb{P} = \mathbb{Q}$.*

The first form shows that the MMD is an integral probability metric (Müller, 1997), along with such popular distances as the Wasserstein and total variation.

There are several natural estimators of the MMD from samples. We will assume $n = m$ and use the U-statistic estima-

tor, which is unbiased for MMD^2 and has nearly minimal variance among unbiased estimators (Gretton et al., 2012a):

$$\widehat{\mathrm{MMD}}_u^2(S_{\mathbb{P}}, S_{\mathbb{Q}}; k) := \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij} \tag{2}$$

$$H_{ij} := k(X_i, X_j) + k(Y_i, Y_j) - k(X_i, Y_j) - k(Y_i, X_j).$$

The similar $\widehat{\mathrm{MMD}}_b^2 := \frac{1}{n^2} \sum_{ij} H_{ij}$ is the squared MMD between the empirical distributions of $S_{\mathbb{P}}$ and $S_{\mathbb{Q}}$.[1]

Testing with the MMD. It can be shown that under $\mathfrak{H}_0$, $n\widehat{\mathrm{MMD}}_u^2$ converges to a distribution depending on $\mathbb{P}$ and k; we thus use this as our test statistic.

Proposition 2 (Asymptotics of $\widehat{\mathrm{MMD}}_u^2$)**.** *Under the null hypothesis, $\mathfrak{H}_0 : \mathbb{P} = \mathbb{Q}$, we have if $Z_i \sim \mathcal{N}(0, 2)$,*

$$n\widehat{\mathrm{MMD}}_u^2 \xrightarrow{\boldsymbol{d}} \sum_i \sigma_i (Z_i^2 - 2);$$

here σ_i are the eigenvalues of the $\mathbb{P}$-covariance operator of the centered kernel (Gretton et al., 2012a, Theorem 12), and $\xrightarrow{\boldsymbol{d}}$ denotes convergence in distribution.

Under the alternative, $\mathfrak{H}_1 : \mathbb{P} \neq \mathbb{Q}$, a standard central limit theorem holds (Serfling, 1980, Section 5.5.1):

$$\sqrt{n}(\widehat{\mathrm{MMD}}_u^2 - \mathrm{MMD}^2) \xrightarrow{\boldsymbol{d}} \mathcal{N}(0, \sigma_{\mathfrak{H}_1}^2)$$
$$\sigma_{\mathfrak{H}_1}^2 := 4\left(\mathbb{E}[H_{12}H_{13}] - \mathbb{E}[H_{12}]^2\right)$$

where H_{12}, H_{13} refer to H_{ij} above.

Although it is possible to construct a test based on directly estimating this null distribution (Gretton et al., 2009), it is both simpler and, if implemented carefully, faster (Sutherland et al., 2017) to instead use a permutation test. This general method (Dwass, 1957; Alba Fernández et al., 2008) observes that under $\mathfrak{H}_0$, the samples from $\mathbb{P}$ and $\mathbb{Q}$ are interchangeable; we can therefore estimate the null distribution of our test statistic by repeatedly re-computing it with the samples randomly re-assigned to $S_{\mathbb{P}}$ or $S_{\mathbb{Q}}$.

Test power. The main measure of efficacy of a null hypothesis test is its *power*: the probability that, for a particular $\mathbb{P} \neq \mathbb{Q}$ and n, we correctly reject $\mathfrak{H}_0$. Proposition 2 implies, where Φ is the standard normal CDF, that

$$\Pr_{\mathfrak{H}_1}\left(n\widehat{\mathrm{MMD}}_u^2 > r\right) \to \Phi\left(\frac{\sqrt{n}\,\mathrm{MMD}^2}{\sigma_{\mathfrak{H}_1}} - \frac{r}{\sqrt{n}\,\sigma_{\mathfrak{H}_1}}\right);$$

we can find the approximate test power by using the rejection threshold, found via (e.g.) permutation testing, as r. We also know via Proposition 2 that this r will converge to a constant, and MMD, $\sigma_{\mathfrak{H}_1}$ are also constants. For reasonably large n, the power is dominated by the first term, and the kernel yielding the most powerful test will approximately maximize (Sutherland et al., 2017)

$$J(\mathbb{P}, \mathbb{Q}; k) := \mathrm{MMD}^2(\mathbb{P}, \mathbb{Q}; k)/\sigma_{\mathfrak{H}_1}(\mathbb{P}, \mathbb{Q}; k). \tag{3}$$

Selecting a kernel. The criterion $J(\mathbb{P}, \mathbb{Q}; k)$ depends on the particular $\mathbb{P}$ and $\mathbb{Q}$ at hand, and thus we typically will neither be able to choose a kernel *a priori*, nor exactly evaluate J given samples. We can, however, estimate it with

$$\hat{J}_\lambda(S_{\mathbb{P}}, S_{\mathbb{Q}}; k) := \frac{\widehat{\mathrm{MMD}}_u^2(S_{\mathbb{P}}, S_{\mathbb{Q}}; k)}{\hat{\sigma}_{\mathfrak{H}_1,\lambda}(S_{\mathbb{P}}, S_{\mathbb{Q}}; k)}, \tag{4}$$

where $\hat{\sigma}_{\mathfrak{H}_1,\lambda}^2$ is a regularized estimator of $\sigma_{\mathfrak{H}_1}^2$ given by[2]

$$\frac{4}{n^3}\sum_{i=1}^n\left(\sum_{j=1}^n H_{ij}\right)^2 - \frac{4}{n^4}\left(\sum_{i=1}^n\sum_{j=1}^n H_{ij}\right)^2 + \lambda. \tag{5}$$

Given $S_{\mathbb{P}}$ and $S_{\mathbb{Q}}$, we could construct a test by choosing k to maximize $\hat{J}_\lambda(S_{\mathbb{P}}, S_{\mathbb{Q}}; k)$, then using a test statistic based on $\widehat{\mathrm{MMD}}(S_{\mathbb{P}}, S_{\mathbb{Q}}; k)$. This sample re-use, however, violates the conditions of Proposition 2, and permutation testing would require repeatedly re-training k with permuted labels.

Thus we split the data, get $k^{tr} \approx \arg\max_k \hat{J}_\lambda(S_{\mathbb{P}}^{tr}, S_{\mathbb{Q}}^{tr}; k)$, then compute the test statistic and permutation threshold on $S_{\mathbb{P}}^{te}$, $S_{\mathbb{Q}}^{te}$ using k^{tr}. This procedure was proposed for $\widehat{\mathrm{MMD}}_u^2$ by Sutherland et al. (2017), but the same technique works for a variety of tests (Gretton et al., 2012b; Jitkrittum et al., 2016; 2017; Lopez-Paz & Oquab, 2017). Our paper adopts this framework (Section 5) and studies it further.

Relationship to other approaches. One common scheme is to pick a kernel k_ω based on some proxy task, such as a related classification problem (e.g. Kirchler et al. 2020 or the KID score of Binkowski et al. 2018). Although this approach can work quite well, it depends entirely on features from the proxy task applying well to the differences between $\mathbb{P}$ and $\mathbb{Q}$, which can be hard to know in general.

An alternative is to maximize simply $\widehat{\mathrm{MMD}}_u$ (Sriperumbudur et al. 2009; proposed but not evaluated by Kirchler

[1] Including $k(X_i, Y_i)$ terms in $\widehat{\mathrm{MMD}}_u$ gives the minimal variance unbiased estimator, and allows $m \neq n$. The U-statistic is more convenient for analysis and for efficient permutations; in our settings it behaves similarly to the MVUE and $\widehat{\mathrm{MMD}}_b^2$.

[2] This estimator, as a V-statistic, is biased even when $\lambda = 0$ (although this bias is only $O(1/N)$; see Lemma 18). Although Sutherland et al. (2017); Sutherland (2019) give a quadratic-time estimator unbiased for $\sigma_{\mathfrak{H}_1}^2$, it is much more complicated to implement and analyze, likely has higher variance, and (being unbiased) can be negative, especially e.g. when the kernel is poor.

et al.). Ignoring $\sigma_{\mathfrak{H}_1}$ means that, for instance, this approach would choose to simply scale $k \to \infty$, even though this does not change the test at all. Even when this is not possible, Sutherland et al. (2017) found this approach notably worse than maximizing (4); we confirm this in our experiments.

MMD-GANs (Li et al., 2017; Binkowski et al., 2018) also simply maximize $\widehat{\mathrm{MMD}}_u$ to identify the differences between their model $\mathbb{Q}_\theta$ and target $\mathbb{P}$. If $\mathbb{Q}_\theta$ is quite far from $\mathbb{P}$, however, an MMD-GAN requires a "weak" kernel to identify a path for improving $\mathbb{Q}_\theta$ (Arbel et al., 2018), while our ideal kernel is one which perfectly distinguishes $\mathbb{P}$ and $\mathbb{Q}_\theta$ and would likely give no signal for improvement. Our algorithm, theoretical guarantees, and empirical evaluations thus all differ significantly from those for MMD-GANs.

3. Limits of Simple Kernels

We can use the criterion $\hat{J}_\lambda$ of (4) even to select parameters among a simple family, such as the lengthscale of a Gaussian kernel. Doing so on the *Blob* problem of Figure 1 illustrates the limitations of using MMD with these kernels.

In Figure 2c, we show how the maximal value of $\hat{J}$ changes as we see more samples from $\mathbb{P}$ and $\mathbb{Q}$, for both a family of Gaussian kernels (green dashed line) and a family (1) of deep kernels (red line). The optimal $\hat{J}$ is always higher for the deep kernels; as expected, the empirical test power (Figure 2a) is also higher for deep kernels.

Most simple kernels used for MMD tests, whether the Gaussian we use here or Laplace, inverse multiquadric, even automatic relevance determination kernels, are all translation invariant: $k(x, y) = k(x-t, y-t)$ for any $t \in \mathbb{R}^d$. (All kernels used by Sutherland et al. (2017), for instance, were of this type.) Hence the kernel behaves the same way across space, as in Figure 1b. This means that for distributions whose behavior varies through space, whether because principal directions change (as in Figure 1) so the shape should be different, or because some regions are much denser than others and so need a smaller lengthscale (e.g. Wenliang et al., 2019, Figures 1 and 2), any single global choice is suboptimal.

Kernels which are not translation invariant, such as the deep kernels (1) shown in Figure 1c, can adapt to the different shapes necessary in different areas.

4. Relationship to Classifier-Based Tests

Another popular method for conducting two-sample tests is to train a classifier between $S^{tr}_{\mathbb{P}}$ and $S^{tr}_{\mathbb{Q}}$, then assess its performance on $S^{te}_{\mathbb{P}}$, $S^{te}_{\mathbb{Q}}$. If $\mathbb{P} = \mathbb{Q}$, the classification problem is impossible and performance will be at chance.

The most common performance metric is the accuracy (Lopez-Paz & Oquab, 2017); this scheme is fairly common among practitioners, and Kim et al. (2020) showed it to be optimal in rate, but suboptimal in constant, in one limited setting (linear discriminant analysis between high-dimensional elliptical distributions, e.g. Gaussians, with identical covariances). We will call this approach a Classifier Two-Sample Test based on Sign, C2ST-S. Letting $f : \mathcal{X} \to \mathbb{R}$ output classification scores, the C2ST-S statistic is $\widehat{\mathrm{acc}}(S_{\mathbb{P}}, S_{\mathbb{Q}}; f)$ given by

$$\frac{1}{2n} \sum_{X_i \in S_{\mathbb{P}}} \mathbb{1}(f(X_i) > 0) + \frac{1}{2n} \sum_{Y_i \in S_{\mathbb{Q}}} \mathbb{1}(f(Y_i) \leq 0).$$

Let $\mathrm{acc}(\mathbb{P}, \mathbb{Q}; f) := \frac{1}{2} \Pr(f(X) > 0) + \frac{1}{2} \Pr(f(Y) \leq 0)$; $\widehat{\mathrm{acc}}$ is unbiased for acc and has a simple asymptotically normal null distribution.

Although it is perhaps not immediately obvious this is the case, C2ST-S is almost a special case of the MMD. Let

$$k^{(S)}_f(x, y) = \frac{1}{4} \mathbb{1}(f(x) > 0) \mathbb{1}(f(y) > 0). \tag{6}$$

A C2ST-S with f is equivalent to an MMD test with $k^{(S)}_f$:

Proposition 3. *It holds that*

$$\mathrm{MMD}(\mathbb{P}, \mathbb{Q}; k^{(S)}_f) = |\mathrm{acc}(\mathbb{P}, \mathbb{Q}; f) - \frac{1}{2}|$$
$$\widehat{\mathrm{MMD}}_b(S_{\mathbb{P}}, S_{\mathbb{Q}}; k^{(S)}_f) = |\widehat{\mathrm{acc}}(S_{\mathbb{P}}, S_{\mathbb{Q}}; f) - \frac{1}{2}|.$$

Proof. The mean embedding $\mu_{\mathbb{P}}$ under $k^{(S)}_f$ is simply $\frac{1}{2} \mathbb{E}\, \mathbb{1}(f(X) > 0) = \frac{1}{2} \Pr(f(X) > 0)$, so the MMD is

$$\frac{1}{2} \Big| \Pr(f(X) > 0) - \Pr(f(Y) > 0) \Big| = \Big| \mathrm{acc}(\mathbb{P}, \mathbb{Q}; f) - \frac{1}{2} \Big|.$$

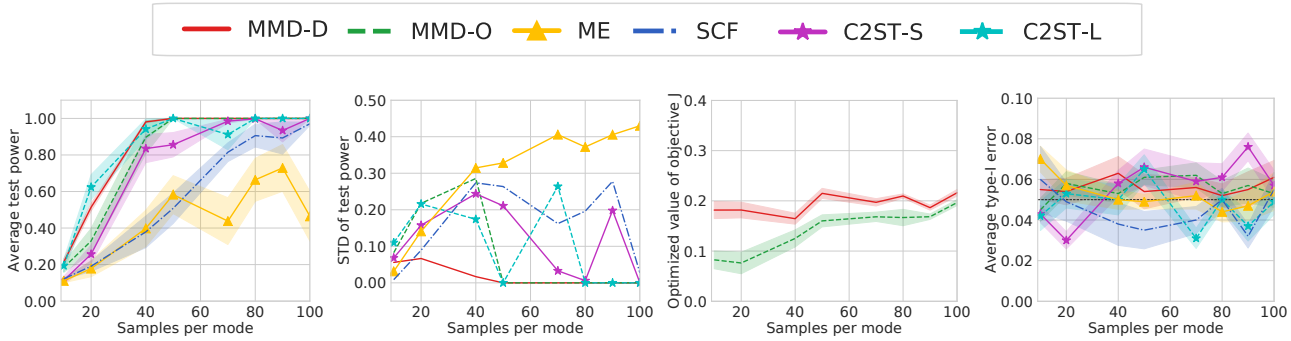
Moreover, $\widehat{\mathrm{acc}}$ is acc on empirical distributions. □

The C2ST-S, however, selects f to maximize cross-entropy (approximately maximizing $\widehat{\mathrm{acc}}$), while we maximize $\hat{J}_\lambda$ (4). Although $k^{(S)}_f$ is not differentiable, maximizing (3) would exactly maximize acc and hence maximize test power (Lopez-Paz & Oquab, 2017, Theorem 1).

Accessing f only through its sign allows for a simple null distribution, but it ignores f's measure of confidence: a highly confident output extremely far from the decision boundary is treated the same as a very uncertain one lying in an area of high overlap between $\mathbb{P}$ and $\mathbb{Q}$, dramatically increasing the variance of the statistic. A scheme we call C2ST-L instead tests difference in means of f on $\mathbb{P}$ and $\mathbb{Q}$ (Chen & Cloninger, 2019). Let

$$k^{(L)}_f(x, y) = f(x) f(y). \tag{7}$$

A C2ST-L is equivalent to an MMD test with $k^{(L)}_f$:



(b) Average test power (c) STD of test power (d) $\hat{J}_\lambda$ value (4) (e) Type-I error

Figure 2. Results on *Blob-S* and *Blob-D* given $\alpha = 0.05$; see Section 7 for details. n_b is the number of samples at each mode, so $n_b = 100$ means drawing 900 samples from each of $\mathbb{P}$ and $\mathbb{Q}$. We report, when increasing n_b, (a) average test power, (b) standard deviation of test power, (c) the value of $\hat{J}_\lambda$, and (d) average type-I error. (a), (b) and (c) are on *Blob-D*, and (d) is on *Blob-S*. Shaded regions show standard errors for the mean, and the black line shows α.

Proposition 4. *It holds that*

$$\mathrm{MMD}(\mathbb{P}, \mathbb{Q}; k_f^{(L)}) = |\mathbb{E}\, f(X) - \mathbb{E}\, f(Y)|$$
$$\widehat{\mathrm{MMD}}_b(S_\mathbb{P}, S_\mathbb{Q}; k_f^{(L)}) = |\frac{1}{n}\sum_{X_i \in S_\mathbb{P}} f(X_i) - \frac{1}{n}\sum_{Y_i \in S_\mathbb{Q}} f(Y_i)|.$$

Proof. This kernel's feature map is $k_f^{(L)}(x, \cdot) = f(x)$. □

Now maximizing accuracy (or a cross-entropy proxy) no longer directly maximizes power. This kernel is differentiable, so we can directly compare the merits of maximizing (4) to maximizing cross-entropy; we will see in Section 7.2 that our more direct approach is empirically superior.

Compared to using $k_f^{(L)}$, however, Section 7.2 shows that learned MMD tests also obtain better performance using kernels like (1). This is analogous to a similar phenomenon observed in other problems by Binkowski et al. (2018) and Wenliang et al. (2019): C2STs learn a full discriminator function on the training set, and then apply only that function to the test set. Learning a deep kernel like (1) corresponds to learning only a powerful *representation* on the training set, and then *still learning* f itself from the test set – in a closed form that makes permutation testing simple.

5. Learning Deep Kernels

Choice of kernel architecture. Most previous work on deep kernels has used a kernel κ directly on the output of a featurization network ϕ_ω, $k_\omega(x, y) = \kappa(\phi_\omega(x), \phi_\omega(y))$. This is certainly also an option for us. Any such k_ω, however, is characteristic if and only if ϕ_ω is injective. If we select our kernel well, this is not really a concern.[3] Even so, it would be reassuring to know that, even if the optimization goes awry, the resulting test will still be at least consistent. More importantly, it can be helpful in optimization to add a "safeguard" preventing the learned kernel from considering extremely far-away inputs as too similar. We can achieve these goals with the form (1), repeated here:

$$k_\omega(x, y) = [(1-\epsilon)\kappa(\phi_\omega(x), \phi_\omega(y)) + \epsilon]\, q(x, y).$$

Here ϕ_ω is a deep network (with parameters ω) that extracts features, and κ is a kernel on those features; we use a Gaussian with lengthscale σ_ϕ, $\kappa(a, b) = \exp\left(-\frac{1}{2\sigma_\phi^2}\|a - b\|^2\right)$. We choose $0 < \epsilon < 1$ and q a Gaussian with lengthscale σ_q.

Proposition 5. *Let k_ω be of the form (1) with $\epsilon > 0$ and q characteristic. Then k_ω is characteristic.*

Learning the deep kernel. The kernel optimization and testing procedure is summarized in Algorithm 1. For larger datasets, or when $n \neq m$, we use minibatches in the training procedure; for smaller datasets, we use full batches. We use the Adam optimizer (Kingma & Ba, 2015). Note that the parameters ϵ, σ_ϕ, and σ_q are included in ω, all parameterized in log-space (i.e. we optimize ϵ' where $\epsilon = \exp(\epsilon')$).

Time complexity. Let E denote the cost of computing an embedding $\phi_\omega(x)$, and K the cost of computing (1) given $\phi_\omega(x)$, $\phi_\omega(y)$. Then each iteration of training in Algorithm 1 costs $\mathcal{O}\left(mE + m^2K\right)$, where m is the minibatch size; for the moderate m that fit in a GPU-sized minibatch anyway, the mE term typically dominates, matching the complexity of a C2ST. Testing takes time $\mathcal{O}\left(nE + n^2K + n^2\, n_{perm}\right)$, compared to $\mathcal{O}\left(nE + n\, n_{perm}\right)$ for permutation-based C2STs. In either case, the quadratic factors could if necessary be reduced

[3] A characteristic kernel on top of even $\phi_\omega(x) = \omega^\mathsf{T} x$ with a *random* ω will be almost surely consistent (Heller & Heller, 2016), and in general the existence of even one good ϕ_ω for a particular $\mathbb{P}$, $\mathbb{Q}$ pair is enough that a perfect optimizer would be able to distinguish the distributions (Arbel et al., 2018, Proposition 1).

Algorithm 1 Testing with a learned deep kernel

Input: $S_{\mathbb{P}}$, $S_{\mathbb{Q}}$, various hyperparameters used below;
$\omega \leftarrow \omega_0$; $\lambda \leftarrow 10^{-8}$;
Split the data as $S_{\mathbb{P}} = S^{tr}_{\mathbb{P}} \cup S^{te}_{\mathbb{P}}$ and $S_{\mathbb{Q}} = S^{tr}_{\mathbb{Q}} \cup S^{te}_{\mathbb{Q}}$;
 # Phase 1: train the kernel parameters ω on $S^{tr}_{\mathbb{P}}$ and $S^{tr}_{\mathbb{Q}}$
for $T = 1, 2, \ldots, T_{max}$ **do**
  $X \leftarrow$ minibatch from $S^{tr}_{\mathbb{P}}$; $Y \leftarrow$ minibatch from $S^{te}_{\mathbb{Q}}$;
  $k_\omega \leftarrow$ kernel function with parameters ω; *# as in (1)*
  $M(\omega) \leftarrow \widehat{\mathrm{MMD}}^2_u(X, Y; k_\omega)$; *# using (2)*
  $V_\lambda(\omega) \leftarrow \hat{\sigma}^2_{\mathfrak{H}_1,\lambda}(X, Y; k_\omega)$; *# using (5)*
  $\hat{J}_\lambda(\omega) \leftarrow M(\omega)/\sqrt{V_\lambda(\omega)}$; *# as in (4)*
  $\omega \leftarrow \omega + \eta \nabla_{\text{Adam}} \hat{J}_\lambda(\omega)$; *# maximize $\hat{J}_\lambda(\omega)$*
end for
 # Phase 2: permutation test with k_ω on $S^{te}_{\mathbb{P}}$ and $S^{te}_{\mathbb{Q}}$
$est \leftarrow \widehat{\mathrm{MMD}}^2_u(S^{te}_{\mathbb{P}}, S^{te}_{\mathbb{Q}}; k_\omega)$
for $i = 1, 2, \ldots, n_{perm}$ **do**
  Shuffle $S^{te}_{\mathbb{P}} \cup S^{te}_{\mathbb{Q}}$ into X and Y
  $perm_i \leftarrow \widehat{\mathrm{MMD}}^2_u(X, Y; k_\omega)$
end for
Output: k_ω, est, p-value $\frac{1}{n_{perm}} \sum_{i=1}^{n_{perm}} \mathbb{1}(perm_i \geq est)$

with the block estimator approach of Zaremba et al. (2013), at the cost of some test power. In our experiments in Section 7, the overall runtime of our methods was scarcely different from the overall runtime of C2STs.

6. Theoretical Analysis

We now show that optimizing the regularized test power criterion based on a finite number of samples works: as n increases, our estimates converge uniformly over a ball in parameter space, and therefore if there is a unique best kernel, we converge to it. Sutherland et al. (2017) gave no such guarantees; this result allows us to trust that, at least for reasonably large n and if our optimization process succeeds, we will find a kernel that generalizes nearly optimally rather than just overfitting to S^{tr}.

We first state a generic result, then show some choices of kernels, particularly deep kernels (1), satisfy the conditions.

Theorem 6. *Let ω parameterize uniformly bounded kernel functions k_ω in a Banach space of dimension D, with $|k_\omega(x, y) - k_{\omega'}(x, y)| \leq L_k \|\omega - \omega'\|$. Let $\bar{\Omega}_s$ be a set of ω for which $\sigma^2_{\mathfrak{H}_1}(\mathbb{P}, \mathbb{Q}; k_\omega) \geq s^2 > 0$ and $\|\omega\| \leq R_\Omega$. Take $\lambda = n^{-1/3}$. Then, with probability at least $1 - \delta$,*

$$\sup_{\omega \in \bar{\Omega}_s} |\hat{J}_\lambda(S_{\mathbb{P}}, S_{\mathbb{Q}}; k_\omega) - J(\mathbb{P}, \mathbb{Q}; k_\omega)| = \mathcal{O}\left(\frac{1}{s^2 n^{1/3}}\left[\frac{1}{s} + \sqrt{D \log(R_\Omega n) + \log\frac{1}{\delta}} + L_k\right]\right).$$

If there is a unique best kernel ω^*, the maximizer of $\hat{J}_\lambda$ converges in probability to ω^* as $n \to \infty$.

A version with explicit constants and more details is given in Appendix A (as Theorem 11 and Corollary 12); the proof is based on uniform convergence of the MMD and variance estimators using an ϵ-net argument.

The following results are shown in Appendix A.4. We first show a result on simple Gaussian bandwidth selection.

Proposition 7. *Suppose each $x \in \mathcal{X}$ has $\|x\| \leq R_X$, and we choose the bandwidth of a Gaussian kernel among a set whose minimum is at least $1/R_\Omega$. Then the conditions of Theorem 6 are met with $D = 1$ and $L_k = 2R_X/\sqrt{e}$.*

Our results also apply to multiple kernel learning, where in fact the exact maximizer of $\hat{J}_\lambda$ is efficiently available (Proposition 27).

Proposition 8. *Let $\{k_i\}_{i=1}^D$ be a fixed set of kernels, with $\sup_x k_i(x, x) \leq K$ for all i. Then picking $k_\omega = \sum_{i=1}^D \omega_i k_i$ among some set of ω with $\sum_{i=1}^D \omega_i^2 \leq R_\Omega^2$ satisfies the conditions of Theorem 6 with $L_k = K\sqrt{D}$.*

We finally establish our results for fully-connected deep kernels; it also applies to convolutional networks with a slightly different R_Ω (Remark 25). The constants in L_k are given in Proposition 23.

Proposition 9. *Take k_ω as in Section 5, with ϕ_ω a fully-connected network with depth Λ and D total parameters, whose activations are 1-Lipschitz with $\sigma(0) = 0$ (e.g. ReLU). Suppose the operator norm of each weight matrix and L_2 norm of each bias vector are is at most R_Ω, and each $x \in \mathcal{X}$ has $\|x\| \leq R_X$. Then k_ω meets the conditions of Theorem 6 with dimension D and $L_K = \mathcal{O}\left(\Lambda R_\Omega^{\Lambda-1} \frac{R_X+1}{\sigma_\phi}\right)$.*

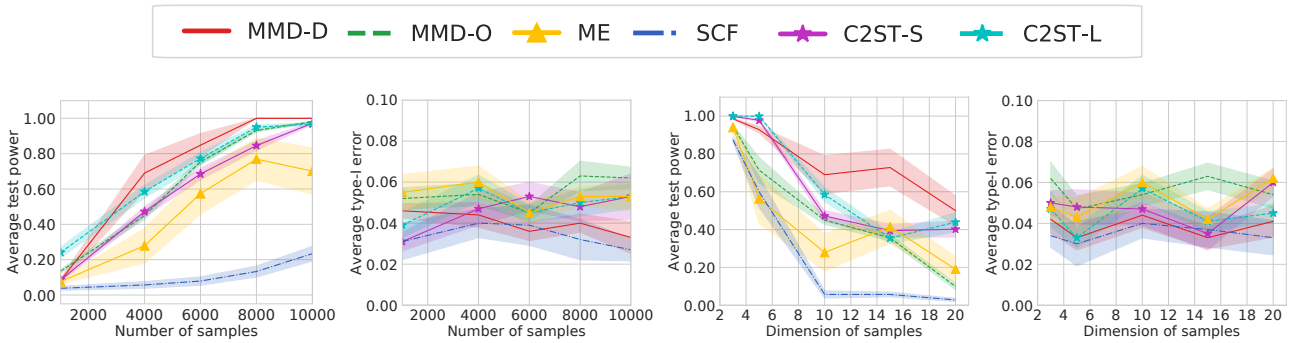
The dependence on s in Theorem 6 is somewhat unfortunate, but the ratio structure of J means that otherwise, errors in very small variances can hurt us arbitrarily. Even so, "near-perfect" kernels (with reasonably large MMD and very small variance) will likely still be chosen as the maximizer of the regularized criterion, even if we do not estimate the (extremely large) ratio accurately. Likewise, near-constant kernels (with very small variance but still small J) will generally have their J *under*estimated, and so are unlikely to be selected when a better kernel is available. The ϵq component in (1) may also help avoid extremely small variances.

Given N data points, this result also gives insight into how many we should use to train the kernel and how many to test. With perfect optimization, Corollary 14 shows a bound on the asymptotic power of the test is maximized by training on $\Theta\left(\left(N\sqrt{\log N}\right)^{\frac{3}{4}}\right)$ points, and testing on the remainder.

7. Experimental Results

7.1. Comparison on Benchmark Datasets

We compare the following tests on several datasets:



(b) Power vs. N; $d = 10$ (c) Level vs. N; $d = 10$ (d) Power vs. d; $N = 4\,000$ (e) Level vs. d; $N = 4\,000$

Figure 3. Results on *HDGM-S* and *HDGM-D* for $\alpha = 0.05$ (black line). Left: average test power (a) and Type I error (b) when increasing the number of samples N, keeping $d = 10$. Right: average test power (c) and Type I error (d) when increasing the dimension d, keeping $N = 4\,000$. Shaded regions show standard errors for the mean.

- MMD-D: MMD with a deep kernel; our method described in Section 5.
- MMD-O: MMD with a Gaussian kernel whose length-scale is optimized as in Section 5. This gives better results than standard heuristics.
- Mean embedding (ME): a state-of-the-art test (Chwialkowski et al., 2015; Jitkrittum et al., 2016) based on differences in Gaussian kernel mean embeddings at a set of optimized points.
- Smooth characteristic functions (SCF): a state-of-the-art test (Chwialkowski et al., 2015; Jitkrittum et al., 2016) based on differences in Gaussian mean embeddings at a set of optimized frequencies.
- Classifier two-sample tests, including C2STS-S (Lopez-Paz & Oquab, 2017) and C2ST-L (Chen & Cloninger, 2019) as described in Section 4. We set the test thresholds via permutation for both.

For synthetic datasets, we take a single sample set for $S^{tr}_{\mathbb{P}}$ and $S^{tr}_{\mathbb{Q}}$ and learn a kernel/test locations/etc once for each method on that training set. We then evaluate its rejection rate on 100 new sample sets $S^{te}_{\mathbb{P}}$, $S^{te}_{\mathbb{Q}}$ from the same distribution. For real datasets, we select a subset of the available data for $S^{tr}_{\mathbb{P}}$ and $S^{tr}_{\mathbb{Q}}$ and train on that; we then evaluate on 100 random subsets, disjoint from the training set, of the remaining data. We repeat this full process 10 times, and report the mean rejection rate of each test. Table 5 shows significance tests. Further details are in Appendix B.

***Blob* dataset.** *Blob-D* is the dataset shown in Figure 1; *Blob-S* has $\mathbb{Q}$ also equal to the distribution shown in Figure 1a, so that the null hypothesis holds. Details are given in Table 6 (Appendix B.1).

Results are shown in Figure 2. MMD-D and C2ST-L are the clear winners in power, with MMD-D better in the higher-sample regime, and MMD-D is more reliable than C2STs. Figure 2c shows that J is higher for MMD-D than MMD-O, in addition to the actual test power being better, as discussed in Section 3. All methods have expected Type I error rates.

High-dimensional Gaussian mixtures. Here we study bimodal Gaussian mixtures in increasing dimension. Each distribution has two Gaussian components; in *HDGM-S*, $\mathbb{P}$ and $\mathbb{Q}$ are the same, while in *HDGM-D*, $\mathbb{P}$ and $\mathbb{Q}$ differ in the covariance of a single dimension pair but are otherwise the same. Details are in Table 6 (Appendix B.1). We consider both increasing N while keeping $d = 10$ and increasing d while keeping $N = 4\,000$, with results shown in Figure 3. Again, MMD-D has generally the best test power across a range of problem settings, with reasonable type I error.

***Higgs* dataset (Baldi et al., 2014).** We compare the jet ϕ-momenta distribution ($d = 4$) of the background process, $\mathbb{P}$, which lacks Higgs bosons, to the corresponding distribution $\mathbb{Q}$ for the process that produces Higgs bosons, following Chwialkowski et al. (2015). As discussed in these previous works, ϕ-momenta carry very little discriminating information for recognizing whether Higgs bosons were produced. We consider a series of tests with increased number of samples N.

We report average test power (comparing $\mathbb{P}$ to $\mathbb{Q}$) in Table 1, and average type-I error (comparing $\mathbb{P}$ to $\mathbb{P}$ or $\mathbb{Q}$ to $\mathbb{Q}$) in Table 7 (Appendix B.6). As before, MMD-D generally performs the best; although the improvement over MMD-O here is not dramatic, MMD-D does notably outperform C2ST. All methods maintain reasonable Type I errors.

***MNIST* generative model.** The *MNIST* dataset contains 70 000 handwritten digit images (LeCun et al., 1998). We compare true *MNIST* data samples $\mathbb{P}$ to samples $\mathbb{Q}$ from a pretrained deep convolutional generative adversarial network (DCGAN) (Radford et al., 2016). Samples from both distributions are shown in Figure 4 (in Appendix B.2).

We consider tests for increasing numbers of samples N, and report average test power (for $\mathbb{P}$ to $\mathbb{Q}$) in Table 2 and

Table 1. Higgs ($\alpha = 0.05$): average test power±standard error for N samples. Bold represents the highest mean per row.

N	ME	SCF	C2ST-S	C2ST-L	MMD-O	MMD-D
1 000	0.120±0.007	0.095±0.022	0.082±0.015	0.097±0.014	**0.132±0.005**	0.113±0.013
2 000	0.165±0.019	0.130±0.026	0.183±0.032	0.232±0.017	0.291±0.012	**0.304±0.035**
3 000	0.197±0.012	0.142±0.025	0.257±0.049	0.399±0.058	0.376±0.022	**0.403±0.050**
5 000	0.410±0.041	0.261±0.044	0.592±0.037	0.447±0.045	0.659±0.018	**0.699±0.047**
8 000	0.691±0.067	0.467±0.038	0.892±0.029	0.878±0.020	0.923±0.013	**0.952±0.024**
10 000	0.786±0.041	0.603±0.066	0.974±0.007	0.985±0.005	**1.000±0.000**	**1.000±0.000**
Avg.	0.395	0.283	0.497	0.506	0.564	**0.579**

Table 2. MNIST ($\alpha = 0.05$): average test power±standard error for comparing N real images to N DCGAN samples.

N	ME	SCF	C2ST-S	C2ST-L	MMD-O	MMD-D
200	0.414±0.050	0.107±0.018	0.193±0.037	0.234±0.031	0.188±0.010	**0.555±0.044**
400	0.921±0.032	0.152±0.021	0.646±0.039	0.706±0.047	0.363±0.017	**0.996±0.004**
600	**1.000±0.000**	0.294±0.008	**1.000±0.000**	0.977±0.012	0.619±0.021	**1.000±0.000**
800	**1.000±0.000**	0.317±0.017	**1.000±0.000**	**1.000±0.000**	0.797±0.015	**1.000±0.000**
1 000	**1.000±0.000**	0.346±0.019	**1.000±0.000**	**1.000±0.000**	0.894±0.016	**1.000±0.000**
Avg.	0.867	0.243	0.768	0.783	0.572	**0.910**

average Type I error ($\mathbb{P}$ to $\mathbb{P}$) in Table 8 (in Appendix B.6). MMD-D substantially outperforms its competitors in test power, with the desired Type I error. ME also does well in this case: it is perhaps particularly suited to this problem, since it is capable of identifying either modes dropped by the generative model or spurious modes it inserts.

***CIFAR-10* vs *CIFAR-10.1*.** *CIFAR-10.1* (Recht et al., 2019) is an attempt to collect a new test set for the very popular *CIFAR-10* image classification dataset (Krizhevsky, 2009). Normally, when evaluating a supervised model, we consider the test set an independent sample from the training distribution, ideally never-before-seen by the training algorithm. But modern computer vision model architectures and training procedures have been developed based on repeatedly evaluating on the *CIFAR-10* test set ($\mathbb{P}$), so it is possible that current models themselves are dependent on $\mathbb{P}$. *CIFAR-10.1* ($\mathbb{Q}$) is an attempt at an independent sample from this distribution, collected after the models were trained, so that they are truly independent of $\mathbb{Q}$. These models do obtain substantially lower accuracies on $\mathbb{Q}$ than on $\mathbb{P}$ – but this drop is surprisingly consistent across models, which seems unlikely to be due to the expected overfitting. The main potential explanation proposed by Recht et al. is dataset shift, but their attempt (in their Appendix C.2.8) at what amounts to a C2ST-S did not reject $\mathfrak{H}_0$.[4] Samples from each distribution are shown in Figure 5 (Appendix B.2).

We train on 1 000 images from each dataset and test on 1 031, so that we use the entirety of *CIFAR-10.1* each time, and average over ten repetitions. These tests provide strong evidence (Table 3) that images in the *CIFAR-10.1* test set *are* statistically different from the *CIFAR-10* test set, with MMD-D again strongest and ME still performing well.

Table 3. CIFAR-10.1 ($\alpha = 0.05$): mean rejection rates.

ME	SCF	C2ST-S	C2ST-L	MMD-O	MMD-D
0.588	0.171	0.452	0.529	0.316	**0.744**

Our learned kernel also helps provide some ability to interpret the difference between $\mathbb{P}$ and $\mathbb{Q}$, particularly if we use it for an ME test. Appendix C explores this.

Recht et al. (2019) also provide a new ImageNetV2 test set for the ImageNet dataset, with similar properties; we defer this more challenging problem to future work.

7.2. Ablation Study

We now study in more detail the difference between MMD-D and closely related methods. Recall from Section 4 that there are two main differences between MMD-D and C2STs: first, using a "full" kernel (1) rather than the sign-based kernel (6) or the intermediate linear kernel (7). Second, training to maximize $\hat{J}_\lambda$ (4) rather than a cross-entry surrogate. MMD-D uses a full kernel (1) trained for test power; C2ST-S effectively uses the sign kernel (6) trained for cross entropy.

In this section, we consider the performance of several intermediate models empirically, demonstrating that both factors help in testing. All are based on the same feature extraction architecture ϕ_ω; some models add a classification layer with new parameters w and b,

$$f_\omega(x) = w^\mathsf{T}\phi_\omega(x) + b,$$

[4] Assuming pretrained classifiers are independent of $\mathbb{P}$, Figure 1 of Recht et al. (2019) indicates that the joint (images, labels) distribution certainly differs between *CIFAR-10* and *CIFAR-10.1*. We test here whether the marginal image distribution differs.

Table 4. Mean test power on *Blob* ($n_b = 40$), *HDGM* ($N = 4000, d = 10$), *Higgs* ($N = 3000$) and *MNIST* ($N = 400$) for $\alpha = 0.05$. See Section 7.2 for the naming scheme; S+C corresponds to C2ST-S, L+C to C2ST-L, and D+J to MMD-D. L+M is the method proposed by Kirchler et al. (2020).

	S+C	L+C	G+C	D+C	L+M	G+M	D+M	L+J	G+J	D+J
Blob	0.835	0.942	0.901	0.900	0.851	0.960	0.906	0.952	0.966	**0.985**
HDGM	0.472	0.585	0.287	0.302	0.494	0.223	0.539	0.635	0.604	**0.659**
Higgs	0.257	0.399	0.353	0.384	0.321	0.254	0.379	0.295	0.364	**0.403**
MNIST	0.646	0.706	0.784	0.803	0.845	0.680	0.760	0.935	0.976	**0.996**
Avg.	0.553	0.658	0.581	0.597	0.628	0.529	0.646	0.704	0.727	**0.761**

Table 5. Paired t-test results ($\alpha = 0.05$) for the results of Section 7.1. For *HDGM*, we fix $d = 10$ (corresponding to Figure 3a). ✓ indicates MMD-D achieved statistically significantly higher mean test power than the other method, × that it did not.

Dataset	ME	SCF	C2ST-S	C2ST-L	MMD-O
Blob	✓	✓	✓	×	×
HDGM	✓	✓	✓	✓	✓
Higgs	✓	✓	✓	×	×
MNIST	✓	✓	✓	✓	✓

which is treated as outputting classification logits. The model variants we consider are

S A kernel $\mathbb{1}(f_\omega(x) > 0)\mathbb{1}(f_\omega(y) > 0)$; corresponds to a test statistic of the accuracy of f (Proposition 3).

L A kernel $f_\omega(x)f_\omega(y)$; corresponds to a test statistic comparing the mean value of f (Proposition 4).

G A Gaussian kernel $\kappa(\phi_\omega(x), \phi_\omega(y))$.

D The deep kernel (1) based on ϕ_ω.

We combine these model variants with a suffix describing the optimization objective:

J Choose ω, including possibly w and b, to optimize the approximate test power (4).

M Choose ω, including possibly w and b, to maximize the value of the empirical MMD between two samples.[5]

C Choose ω, including w and b, to optimize cross-entropy using the classifier that specifies the probability of x belonging to $\mathbb{P}$ as $1/\left(1 + \exp(-f_\omega(x))\right)$.[6]

Table 4 presents results for all of these methods (except for S+J, which is non-differentiable and hence difficult to optimize). Performance generally improves as we move from S to L to G to D, and from C to J, and from M to J.

[5]If a deep kernel is unbounded, directly maximizing MMD will make optimized parameters of ϕ_ω be infinite. Thus, for L+M, we consider a normalized linear deep kernel: $\tanh(f_\omega(x)/\|S\|_F)\tanh(f_\omega(y)/\|S\|_F)$, where $S = [S_\mathbb{P}; S_\mathbb{Q}]$ and $\|\cdot\|_F$ is the Frobenius norm.

[6]G+C and D+C take the fixed ϕ_ω embeddings, then find the optimal lengthscale/etc by optimizing $\hat{J}_\lambda$.

7.3. Architecture design of deep kernels

For *Blob*, *HDGM* and *Higgs*, ϕ_ω is a five-layer fully-connected neural network, with softplus activations. the number of neurons in hidden and output layers of ϕ_ω are set to 50 for *Blob*, $3d$ for *HDGM* and 20 for *Higgs*, where d is the dimension of samples. in general, we expect similar fully-connected networks, to be reasonable choices for datasets where strong structural assumptions are not known, perhaps with $3d$ as a baseline width for datasets of at least moderate dimension.

For *MNIST* and *CIFAR*, ϕ_ω is a *convolutional neural network* (CNN) that contains four convolutional layers and one fully-connected layer. The structure of the CNN follows the structure of the feature extractor in the DCGAN's discriminator (Radford et al., 2016) (see Figures 6 and 8 for the structure of ϕ_ω in MMD-D, and Figures 7 and 9 for the structure of classifier F in C2ST-S and C2ST-L). In general, we expect GAN discriminator architectures to work well for image datasets, as the problem is closely related.

8. Conclusions

The test power of MMD is limited by simple kernels (e.g., Gaussian kernel or other translation-invariant kernels) when facing complex-structured distributions, but we can avoid this problem with richer *deep kernels*, which is no longer translation-invariant. We show that optimizing the parameters of these kernels to maximize the test power, as proposed by Sutherland et al. (2017), outperforms state-of-the-art alternatives even when considering large, deep kernels with hundreds of thousands of parameters, rather than the simple shallow kernels they considered. We provide theoretical guarantees that this process is reasonable to conduct on finite samples, and asymptotically selects the most powerful kernel. We also give deeper insight into the relationship between this approach and classifier two-sample tests (Lopez-Paz & Oquab, 2017), explaining why this approach outperforms that one.

We thus recommend practitioners to use optimized deep kernel methods when they wish to check if two distributions are the same, rather than indirectly training a classifier.

Acknowledgements

This work was supported by the Australian Research Council under FL190100149 and DP170101632, and by the Gatsby Charitable Foundation. FL, JL and GZ gratefully acknowledge the support of the NVIDIA Corporation with the donation of two NVIDIA TITAN V GPUs for this work. FL also acknowledges the support from UTS-FEIT and UTS-AAII. DJS would like to thank Aram Ebtekar, Ameya Velingker, and Siddhartha Jain for productive discussions.

References

Alba Fernández, V., Jiménez Gamero, M., and Muñoz García, J. A test for the two-sample problem based on empirical characteristic functions. *Computational Statistics & Data Analysis*, 52(7):3730–3748, 2008.

Arbel, M., Sutherland, D. J., Binkowski, M., and Gretton, A. On gradient regularizers for MMD GANs. In *NeurIPS*, 2018.

Baldi, P., Sadowski, P., and Whiteson, D. Searching for exotic particles in high-energy physics with deep learning. *Nature communications*, 5:4308, 2014.

Berlinet, A. and Thomas-Agnan, C. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer, 2004.

Bibi, A., Ghanem, B., Koltun, V., and Ranftl, R. Deep layers as stochastic solvers. In *ICLR*, 2019.

Binkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. Demystifying MMD GANs. In *ICLR*, 2018.

Callaert, H. and Janssen, P. The Berry-Esseen theorem for u-statistics. *The Annals of Statistics*, 6(2):417–421, 1978.

Chen, H. and Friedman, J. H. A new graph-based two-sample test for multivariate and object data. *Journal of the American Statistical Association*, 112(517):397–409, 2017.

Chen, X. and Cloninger, A. Classification logit two-sample testing by neural networks, 2019. arXiv:1909.11298.

Chwialkowski, K., Ramdas, A., Sejdinovic, D., and Gretton, A. Fast two-sample testing with analytic representations of probability measures. In *NeurIPS*, 2015.

Cucker, F. and Smale, S. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1):1–49, 2001.

Dwass, M. Modified randomization tests for nonparametric hypotheses. *The Annals of Mathematical Statistics*, 28 (1):181–187, 03 1957.

Gao, R., Xie, L., Xie, Y., and Xu, H. Robust hypothesis testing using Wasserstein uncertainty sets. In *NeurIPS*, 2018.

Ghoshdastidar, D. and von Luxburg, U. Practical methods for graph two-sample testing. In *NeurIPS*, 2018.

Ghoshdastidar, D., Gutzeit, M., Carpentier, A., and von Luxburg, U. Two-sample tests for large random graphs using network statistics. In *COLT*, 2017.

Gönen, M. and Alpaydn, E. Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12: 2211–2268, 2011.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Systems, I. Domain adaptation with conditional transferable components. In *ICML*, 2016.

Gretton, A., Fukumizu, K., Harchaoui, Z., and Sriperumbudur, B. K. A fast, consistent kernel two-sample test. In *NeurIPS*, 2009.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. J. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012a.

Gretton, A., Sriperumbudur, B., Sejdinovic, D., Strathmann, H., and Pontil, M. Optimal kernel choice for large-scale two-sample tests. In *NeurIPS*, 2012b.

Harchaoui, Z., Bach, F., and Moulines, E. Testing for homogeneity with kernel Fisher discriminant analysis. In *NeurIPS*, 2007.

Heller, R. and Heller, Y. Multivariate tests of association based on univariate tests. In *NeurIPS*, 2016.

Jean, N., Xie, S. M., and Ermon, S. Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance. In *NeurIPS*, 2018.

Jitkrittum, W., Szabo, Z., Chwialkowski, K., and Gretton, A. Interpretable distribution features with maximum testing power. In *NeurIPS*, 2016.

Jitkrittum, W., Xu, W., Szabo, Z., Fukumizu, K., and Gretton, A. A linear-time kernel goodness-of-fit test. In *NeurIPS*, 2017.

Kim, I., Ramdas, A., Singh, A., and Wasserman, L. Classification accuracy as a proxy for two sample testing. *Annals of Statistics*, 2020. arXiv:1602.02210.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Kirchler, M., Khorasani, S., Kloft, M., and Lippert, C. Two-sample testing using deep learning. In *AISTATS*, 2020. arXiv:1910.06239.

Korolyuk, V. S. and Borovskikh, Y. V. Asymptotic theory of U-statistics. *Ukrainian Mathematical Journal*, 40(2): 142–154, 1988.

Krizhevsky, A. Learning multiple layers of features from tiny images, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y., and Póczos, B. MMD GAN: Towards deeper understanding of moment matching network. In *NeurIPS*, 2017.

Li, S. and Wang, X. Fully distributed sequential hypothesis testing: Algorithms and asymptotic analyses. *IEEE Trans. Information Theory*, 64(4):2742–2758, 2018.

Lopez-Paz, D. and Oquab, M. Revisiting classifier two-sample tests. In *ICLR*, 2017.

Muandet, K., Fukumizu, K., Sriperumbudur, B., and Schölkopf, B. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, May 2017.

Müller, A. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29 (2):429–443, 1997.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

Ramdas, A., García Trillos, N., and Cuturi, M. On Wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19(2):47, January 2017.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do ImageNet classifiers generalize to ImageNet? In *ICML*, 2019.

Sedghi, H., Gupta, V., and Long, P. M. The singular values of convolutional layers. In *ICLR*, 2019.

Serfling, R. J. *Approximation Theorems of Mathematical Statistics*. John Wiley & Sons, 1980.

Smola, A. J. and Schölkopf, B. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

Sriperumbudur, B. K., Fukumizu, K., Gretton, A., Lanckriet, G. R., and Schlkopf, B. Kernel choice and classifiability for rkhs embeddings of probability distributions. In *NeurIPS*, 2009.

Stojanov, P., Gong, M., Carbonell, J. G., and Zhang, K. Data-driven approach to multiple-source domain adaptation. In *AISTATS*, 2019.

Sutherland, D. J. Unbiased estimators for the variance of MMD estimators, 2019. arXiv:1906.02104.

Sutherland, D. J., Tung, H.-Y., Strathmann, H., De, S., Ramdas, A., Smola, A., and Gretton, A. Generative models and model criticism via optimized maximum mean discrepancy. In *ICLR*, 2017.

Székely, G. J. and Rizzo, M. L. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013.

Torralba, A., Fergus, R., and Freeman, W. T. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

Van der Vaart, A. W. *Asymptotic Statistics*. Cambridge University Press, 2000.

Wenliang, L., Sutherland, D. J., Strathmann, H., and Gretton, A. Learning deep kernels for exponential family densities. In *ICML*, 2019.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In *AISTATS*, 2016.

Zaremba, W., Gretton, A., and Blaschko, M. B-tests: Low variance kernel two-sample tests. In *NeurIPS*, 2013.