CLASSIFICATION LOGIT TWO-SAMPLE TESTING BY NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

The recent success of generative adversarial networks and variational learning suggests training a classifier network may work well in addressing the classical two-sample problem. Network-based tests have the computational advantage that the algorithm scales to large samples. This paper proposes to use the difference of the logit of a trained neural network classifier evaluated on the two finite samples as the test statistic. Theoretically, we prove the testing power to differentiate two smooth densities given that the network is sufficiently parametrized, by comparing the learned logit function to the log ratio of the densities, the latter maximizing the population training objective. When the two densities lie on or near to low-dimensional manifolds embedded in possibly high-dimensional space, the needed network complexity is reduced to only depending on the intrinsic manifold geometry. In experiments, the method demonstrates better performance than previous network-based tests which use the classification accuracy as the test statistic, and compares favorably to certain kernel maximum mean discrepancy (MMD) tests on synthetic and hand-written digits datasets.

1 INTRODUCTION

The two-sample test problem concerns the comparison of two unknown distributions p and q from finitely observed data samples (Lehmann & Romano, 2006). As a central problem in statistics, it is widely encountered in general data analysis of biomedical data, audio and imaging data, etc. (Borgwardt et al., 2006; Chwialkowski et al., 2015; Jitkrittum et al., 2016; Lopez-Paz & Oquab, 2016; Cheng et al., 2017), and particularly, in the machine learning application of training and evaluating generative models (Li et al., 2015; 2017; Goodfellow et al., 2014; Arjovsky et al., 2017; Nowozin et al., 2016; Lloyd & Ghahramani, 2015; Sutherland et al., 2016; Chwialkowski et al., 2016; Liu et al., 2016; Jitkrittum et al., 2017). Many existing tests are based on certain estimators of a distance or divergence between p and q. Important examples include Maximum Mean Discrepancy (MMD), especially kernel-based MMD (Anderson et al., 1994; Gretton et al., 2012a) and distance of Reproducing Kernel Hilbert Space Mean Embedding (Chwialkowski et al., 2015; Jitkrittum et al., 2016), divergence based methods which may involve non-parametric estimation of density difference or density ratio (Sriperumbudur et al., 2009; Wornowizki & Fried, 2016; Sugiyama et al., 2013; Kanamori et al., 2012). While these methods have been intensively studied and theoretically well-understood, the application is often restricted to data of small dimensionality and/or small sample size, or certain specific classes of densities p and q, due to model and computational limitations.

The powerful expressiveness of neural networks and the recent progress in optimization of deep networks suggest the natural idea of using a network for the two-sample problem, as revisited in (Lopez-Paz & Oquab, 2016). In the training of generative adversarial networks (GAN) (Goodfellow et al., 2014; Arjovsky et al., 2017; Nowozin et al., 2016), in each iteration a discriminative network (D-net) is trained to distinguish the model density q produced by a generative network from the data density p which is only accessible via observed samples, that is, a two-sample problem. Strictly speaking, the task of D-net is a goodness-of-fit problem as the model density q is analytically given (Chwialkowski et al., 2016; Liu et al., 2016; Jitkrittum et al., 2017). Since batch sampling is commonly used in training GAN and other generative networks, the ability of a trained network, such as the D-net in GAN, to detect the difference between two densities from finite samples is crucial for such applications. We review these connections in more detail in Section 1.1.

The current paper studies the method of training a network for the two-sample problem, where the test statistic is the log ratio of the class probabilities averaged over samples, which can be computed once a classifier network is trained. Our contributions include: (1) We introduce a network-based

two-sample test statistic based on classification logit, and the algorithm inherits the scalable computational efficiency of neural networks; (2) Theoretical guarantee of testing power is proved for smooth densities p and q in $\mathbb{R}^D$, and the needed network complexity is reduced to be intrinsic when p and q lie on or near to low-dimensional manifolds embedded in possibly high dimensional ambient space. (3) Numerical experiments show that the proposed test compares favorably to kernel MMD tests and earlier neural network classifier test based on classification accuracy, on both synthetic manifold data and hand-written digits datasets.

The proposed statistic belongs to a general class of f-divergence between p and q, which means that the method may extend to a broad class of classification networks as suggested by f-GAN (Nowozin et al., 2016) beyond maximizing softmax loss. Since KL divergence (corresponding to softmax loss) is a prototypical case of f-divergence, we focus on softmax classifier network in this paper.

1.1 Related works

Classification and two-sample testing. The relations between two-sample testing, divergence estimation and binary classification has been pointed out in earlier statistical literature Friedman (2004); Sriperumbudur et al. (2009); Reid & Williamson (2011). (Ramdas et al., 2016) studied Fisher LDA classifier used for testing mean shift of Gaussian distributions. Discriminative approach has also been used to detect and correct covariant shifts (Bickel et al., 2007; 2009). Training classifier provides an estimator of density ratio, as has been pointed out in (Menon & Ong, 2016) and in the formulation of learning generative models (Mohamed & Lakshminarayanan, 2016). While distribution divergence estimation has been studied and used for two-sample problems (Kanamori et al., 2011; 2012; Sugiyama et al., 2013; Wornowizki & Fried, 2016), the use of neural network as a divergence estimator for two-sample testing was less investigated. In terms of theoretical guarantee of test power, the analysis in (Lopez-Paz & Oquab, 2016) assumes a non-zero population test statistic under $\mathcal{H}_1$ which is not specified, along with other approximations. Theoretical analysis of neural network two-sample testing power remains limited.

MMD and kernel MMD tests. MMD encloses a wide class of two-sample statistics such as Kolmogorov-Smirnov statistic, Wasserstein metric, and general integral probability metrics. Particularly, kernel-based MMD (Anderson et al., 1994; Gretton et al., 2012a) has been widely applied due to its non-parametric form, and recently in training moment matching networks (MMN) (Li et al., 2015; 2017) and evaluating generative models (Sutherland et al., 2016). To optimize kernel parameters, (Gretton et al., 2012a) considers the selection of kernel bandwidth from data, (Jitkrittum et al., 2016) studies the optimization of reference locations in the Mean Embedding test, (Gretton et al., 2012b) optimizes the kernel via a convex combination of multiple kernels, (Cheng et al., 2017) choses anisotropic kernels. The combination of neural network feature learning and kernel MMD has been studied in (Li et al., 2017), where the training is typically more costly than that of a classifier network. Compared to kernel methods, neural networks are algorithmically more efficient and scalable, and the current paper investigates if it also has advantage in testing power.

Relation to goodness-of-fit test and GAN. The goodness-of-fit problem differs from the two-sample problem in that one of the two densities is analytically accessible. Using the explicit formula of q, methods based on kernel Stein discrepancy have been developed in (Chwialkowski et al., 2016; Liu et al., 2016; Jitkrittum et al., 2017) and applied to generative model evaluation. However, the computation of the *score function* $\nabla \log q$ may be difficult for certain generative models, including many generative networks. Meanwhile, in many generative models including MMN and GAN the goodness-of-fit is evaluated by batch sampling, i.e. the two-sample setting: Kernel MMD is used in MMN, and GAN, Wasserstein GAN (Arjovsky et al., 2017) and f-GAN (Nowozin et al., 2016) estimate density divergence by a trained network (the D-net). Since the success of GAN training relies on the discriminative power of the D-net, the efficiency of using a neural network for the two-sample test is important for the training and evaluating of such models.

2 Log-ratio test by network classifier

2.1 Two-sample problem

Formally, the two-sample problem asks to test the null hypothesis $\mathcal{H}_0 : p = q$ given datasets $X = \{x_i\}_{i=1}^{n_X}$ and $Y = \{y_j\}_{j=1}^{n_Y}$ where $x_i \sim p$ i.i.d., $y_j \sim q$ i.i.d., and X is independent from Y. The test method is usually based upon a statistic $\hat{T} = \hat{T}(X, Y)$, which is computed from the two datasets, and a test threshold τ, and the null hypothesis $\mathcal{H}_0$ is rejected if $\hat{T} > \tau$. To control the false

discovery under the null, the threshold τ is usually set to the smallest value s.t. $\Pr[\hat{T} > \tau|\mathcal{H}_0] \leq \alpha$, where $\alpha \in (0, 1)$ is a pre-specified number called the *significance level* of the test (typically $\alpha = 0.05$). Algorithm-wise, τ is given either by theory (the probabilistic distribution of $\hat{T}$ under $\mathcal{H}_0$) or computed from data (Higgins, 2003; Gretton et al., 2012a).

2.2 Test statistic and density log ratio estimation

The proposed test statistic is computed in the following way: given X and Y as above, without loss of generality suppose $n = n_X + n_Y$ is even integer. Same as in (Lopez-Paz & Oquab, 2016), we split the dataset $\mathcal{D} = \{(x_i, 0)\}_{i=1}^{n_X} \cup \{(y_j, 0)\}_{j=1}^{n_Y} = \{(z_i, l_i)\}_{i=1}^{n}$, $l_i \in \{0, 1\}$, into two halves, i.e. $\mathcal{D} = \mathcal{D}_{\text{tr}} \cup \mathcal{D}_{\text{te}}$, $|\mathcal{D}_{\text{tr}}| = |\mathcal{D}_{\text{te}}| = \frac{n}{2}$, $\mathcal{D}_{\text{te}} = X_{\text{te}} \cup Y_{\text{te}}$ and similarly for the training set. A binary classification neural network is trained on $\mathcal{D}_{\text{tr}}$ using softmax loss, which gives estimated class probabilities as $\Pr[l = 0|z] = \frac{e^{u_\theta(z)}}{e^{u_\theta(z)} + e^{v_\theta(z)}}$, $\Pr[l = 1|z] = \frac{e^{v_\theta(z)}}{e^{u_\theta(z)} + e^{v_\theta(z)}}$, where u_θ and v_θ are activations in the last hidden layer of the network, θ denoting the network parametrization. We define $f_\theta := u_\theta - v_\theta$, which is the *logit*, and let the test statistic be

$$\hat{T} = \frac{1}{|X_{\text{te}}|} \sum_{x \in X_{\text{te}}} f_\theta(x) - \frac{1}{|Y_{\text{te}}|} \sum_{y \in Y_{\text{te}}} f_\theta(y). \tag{1}$$

which can be written as $\hat{T} = \int f_\theta(x)(\hat{p}_{\text{te}}(x) - \hat{q}_{\text{te}}(x))dx$ where $\hat{p}_{\text{te}}$ and $\hat{q}_{\text{te}}$ stand for the empirical density of X_{te} and Y_{te} respectively.

As has been pointed out in (Menon & Ong, 2016), the training of the classifier can be interpreted as minimizing a Bregman divergence between the estimated logit f_θ and the true log ratio $f^* = \log \frac{p}{q}$. Thus the proposed statistic $\hat{T}$ can be viewed as estimating the symmetric KL divergence $\int (p - q) \log \frac{p}{q} = \text{KL}(p||q) + \text{KL}(q||p)$, which is an f-divergence with $f(u) = (u - 1) \log u$ (Kanamori et al., 2011; Nowozin et al., 2016). The testing power of (1) will be theoretically analyzed in Section 4, particularly when p and q lie on or near to low-dimensional manifolds.

2.3 Algorithm in practice

Threshold τ. In practice, the test threshold τ can be computed by a *permutation test* (Higgins, 2003): randomly permute the $|\mathcal{D}_{\text{te}}|$ many labels l_i on the test set, and recompute the test statistics for m_{perm} times, typically a few tens. This gives an empirical distribution of $\hat{T}$ under the null hypothesis where both densities equal a mixture of the original p and q. Then τ is set to be the (1-α)-quantile of the empirical distribution so as to control the type-I error to be at most α.

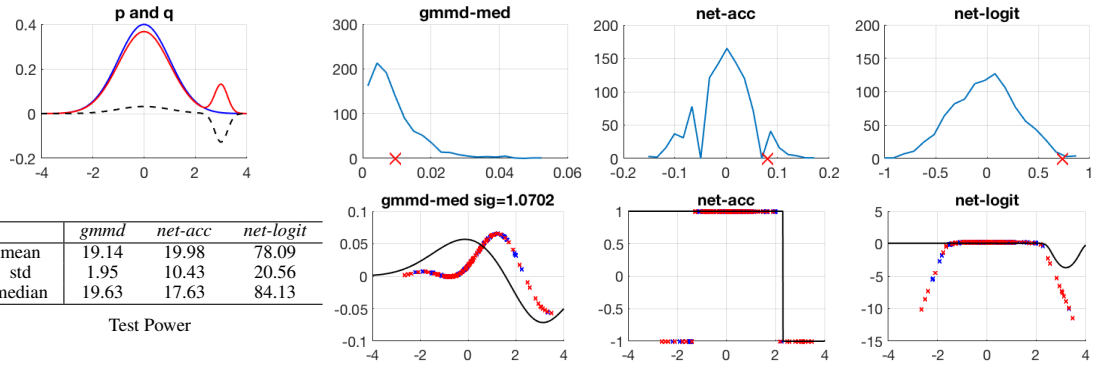
Density difference indicator. Many two-sample methods also provide an indication of where q differs from p, which is often of more application interest, e.g., via the *witness function* in kernel MMD (Gretton et al., 2012a). For the proposed test, such a differential indicator is provided by the logit f_θ of the trained classifier network, which can be viewed as an estimator of log density ratio f^* as discussed above. Following kernel MMD, we call f_θ (and f^*) the empirical (and population) witness function of the proposed classification logit test.

Network training and computational complexity. Our training of network is conducted via Adam (Kingma & Ba, 2014), the convergence of which has been analyzed in many places such as (Reddi et al., 2019). We use fixed learning rate over a fixed number of epochs, and it is entirely possible that our training procedure is over simplified and better usage of stochastic gradient descent method as studied in (Bottou, 2010; Zeiler, 2012; Sutskever et al., 2013; Shamir & Zhang, 2013; Du et al., 2018) may lead to improved performance. Given n data samples, evaluating the network output on each sample takes a fixed amount of flops, and thus computing the test statistic takes $O(n)$ operations. The permutation test to determine τ adds negligible cost since f_θ has been evaluated at each test sample, and permuting the class labels only reorders these computed values. The training phase is certainly more expensive, though theoretically the overall complexity is $O(n)$ assuming that training is terminated after a fixed number of epochs. Note that the computation can be conducted by batch sampling so the algorithm scales to large sample size and also to multiple sample problems.

3 A One-dimensional Example

Setting-up. We compare the proposed test based on log ratio (*net-logit*) to (1) gaussian kernel MMD (*gmmd*) and (2) test based on classification accuracy (*net-acc*) (Lopez-Paz & Oquab, 2016), on 1D example where the distributions are

$$x_i \sim \mathcal{N}(0, 1), \quad y_j \sim (1 - \delta)\mathcal{N}(0, 1) + \delta\mathcal{N}(3, \frac{1}{16}), \tag{2}$$



	gmmd	*net-acc*	*net-logit*
mean	19.14	19.98	78.09
std	1.95	10.43	20.56
median	19.63	17.63	84.13

Test Power

Figure 1: **Plots**: Top-left: Two densities p and q as in (2); Right three columns: The test statistic $\hat{T}$ on $|X_{\text{te}}| = |Y_{\text{te}}| = 100$ samples i.e. under $\mathcal{H}_1$ (red cross) and the histogram of $\hat{T}$ under 1000 permutation tests i.e. $\mathcal{H}_0$ (blue curve). The population witness function (black curve) and the empirical one evaluated on test data (red cross for X_{te}, blue crosses for Y_{te}) of the three methods are shown in the bottom row. **Table**: The mean, standard deviation ("std") and median of the test power computed over $n_{\text{rep}} = 20$ replicas, as described in Sec. 3.

the number $\delta \in [0, 1]$ controls the difference of the densities. We set $\delta = 0.08$, and p and q are illustrated in Fig. 1. For both network tests, 200 training and 200 testing samples are used, and same samples in X and Y. Testing power is computed by the frequency of rejecting $\mathcal{H}_0$ in $n_{\text{run}} = 400$ test runs, and $n_{\text{rep}} = 20$ replicas of whole experiment are used to compute mean and standard deviation of the estimated power. The kernel bandwidth σ in *gmmd* is set to be the median of the pairwise distances among all samples (Gretton et al., 2012a). *gmmd* can make use of the training set for a more fair comparison, see below. More experimental details in Appendix A.

Test power. The table in Fig. 1 lists the power for the three methods, where *net-logit* gives significantly better average power $\sim 80\%$, and the power of *net-acc* and *gmmd* are similar, both $\sim 20\%$. The variation of the power is much larger for the two net-based tests though (c.f. Fig. A.2). We note that such large variation is due to the instability of network training, possibly due to small training size, and is a limitation of the current net-based methods.

One may observe that the above comparison to kernel MMD is not fair: First, kernel MMD with median-distance σ does not use the training set, thus it would be a more fair comparison if *gmmd* can use all the data samples without training-test splitting. Second, the median setting of σ may not be optimal and can be improved by existing methods in literature. We thus repeat the experiment of *gmmd* with median-distance σ which uses all the 400 samples ("*gmmd*+"), and also test over a range of values of σ which are $\{2^{-3}, 2^{-2}, \cdots 2^{3}\}$, and select the best test power ("*gmmd*++"). Other experimental setting being the same as above. The results are reported in Table A.1, where *gmmd*+ achieves a test power of 47% and *gmmd*++ a power of 57%, remaining inferior to *net-logit*, while both with small variation (std $\lesssim 2$) and thus are more stable than net-based tests. Results with other values of δ and sample sizes are reported in Sec. 5.1, Fig. 2.

Witness function. The three tests all provide witness functions to indicate where q differs from p: The population witness functions for *gmmd* is $w_\sigma(x) := \int g_\sigma(x - y)(p(y) - q(y))dy$, $g_\sigma(z) := e^{-|z|^2/(2\sigma^2)}$, and its empirical counterpart is by replacing p and q with the empirical densities of X_{te} and Y_{te} respectively in the integral. Recall that the population and empirical witness function for *net-logit* test are $f^*(x) = \log \frac{p(x)}{q(x)}$ and f_θ respectively. For *net-acc*, when $|X_{\text{te}}| = |Y_{\text{te}}|$, it is equivalent to using the sign (taking value of ± 1) of f_θ instead of f_θ in computing the test statistic in (1), as shown in (A.1). Thus we call $\text{Sign}(f_\theta(x))$ the empirical witness function for the *net-acc* test, and $\text{Sign}(f^*(x))$ its population version.

The population and empirical witness functions (in one test run) are plotted in Fig. 1. Comparing to *gmmd*, the witness function of *net-logit*, i.e., the log density ratio, weighs larger at the differential region which is at the tail of the density p. This is also reflected in the empirical witness functions. Taking the sign of f_θ as done in classification accuracy test introduces discontinuity of at the decision boundary neat $x = 2$, which leads to comparatively larger variance in view of the mean of the statistic under $\mathcal{H}_1$. This intuitively explains why the *net-logit* test is more powerful here, and a quantitative comparison of mean vs. standard deviation of the three tests is given in Appendix A.2.

4 Analysis of Test Power

In this section we prove the power of the network logit test assuming that f_θ is identified by minimizing the population classification loss. The proof is based on network approximation analysis,

and a key question is the needed network complexity, particularly how it scales with the data dimensionality. We first analyze the general case of smooth densities p and q in $\mathbb{R}^D$, and then consider the important case where p and q lie on or near a smooth low-dimensional manifold, where we reduce the needed network complexity to depend on the intrinsic manifold geometry only.

4.1 IDENTIFICATION OF f_θ BY POPULATION LOSS

Training with the population loss of the classification network can be expressed as (samples from X and Y are of same number)

$$\max_{f\in\mathcal{F}_\Theta} L[f] = \frac{1}{2}\left(\int p\log\frac{2e^f}{1+e^f} + \int q\log\frac{2}{1+e^f}\right), \tag{3}$$

where $\mathcal{F}_\Theta$ denotes the class of functions that can be expressed as the difference of the outputs in the last hidden layer of the classification network. A direct verification shows that $f^* = \arg\max_f L[f] = \log\frac{p}{q}$ (see e.g. (Goodfellow et al., 2014) where it is proved in terms of $D = \frac{e^f}{1+e^f}$), which characterizes the solution of (3) when the function class is arbitrarily large or something large enough to contains f^*. Then $L[f^*] = \frac{1}{2}\left(\int p\log\frac{2p}{p+q} + \int q\log\frac{2q}{p+q}\right) = \text{JSD}(p,q)$, which is the Jensen-Shannon divergence. For certain classification network of finite complexity, f^* may not be contained in $\mathcal{F}_\Theta$, and instead the optimization (3) finds some f_θ s.t.

$$L[f_\theta] = \max_{f\in\mathcal{F}_\Theta} L[f]. \tag{4}$$

We analyze the exact optimization of the population loss only. Since we do not address training from finite samples in this section, we abuse the notation of n (which used to denote $|D_{tr}| + |D_{te}|$) to be $n = |X_{te}| = |Y_{te}|$. Then after $f = f_\theta$ is identified by (4), the test statistic $T_n = \hat{T}$ can be computed as in (1), and we denote its expectation as $T[f_\theta] = \int f_\theta(p-q)$.

4.2 TESTING POWER FOR DISTINGUISHING GENERAL p AND q IN $\mathbb{R}^D$

We assume that p and q are smooth densities in $\mathbb{R}^D$ compactly supported on a bounded region Ω, and without loss of generality $\Omega = \{x\in\mathbb{R}^D, |x| < 1\}$. The analysis proceeds in the following three steps, all proofs in Appendix B:

Step 1. Use neural network approximation result to construct a network function $f_{\text{con}} \in \mathcal{F}_\Theta$ that uniformly approximates f^*, which bounds $|L[f_{\text{con}}] - L[f^*]|$ proportionally. This implies that $L[f_{\text{con}}] > C$ for some $C \approx L[f^*] = \text{JSD}(p,q) > 0$ (c.f. Proposition B.1). The relation $L[f_\theta] \geq L[f_{\text{con}}]$ then gives that $L[f_\theta] > C > 0$.

Step 2. The lower bound of $L[f_\theta]$ serves to show that $T[f_\theta] > 4C$ under $\mathcal{H}_1$ via a relation between the two (c.f. Lemma B.2).

Step 3. We compute and bound the variance of T_n to be at order $O(n^{-1})$, under both H_0 and H_1 (c.f. Proposition B.4). This together with the proved $O(1)$ mean (bias) of T_n under H_1 will prove the test power to be asymptotically 1.

Combining these steps gives the following theorem, where we set $r = 2$ in Proposition B.1:

Theorem 4.1. *Let the densities $p = e^u$, $q = e^v$ be C^2 and supported on Ω, the unit ball in $\mathbb{R}^D$, and $u, v \in C^2(\Omega)$, $f^* = u - v$. If $p \neq q$ such that JSD$(p,q) > 0$, then for any small $0 < \varepsilon_1 <$ JSD(p,q), there is a neural network architecture Θ with $O(\varepsilon_1^{-D/2})$ many trainable parameters, where the constant depends on the regularity of the second derivative of f^*, such that f_θ identified by* (3) *satisfies that*

(1) $\mathbb{E}T_n = T[f_\theta] > 4C$, where $C =$ JSD$(p,q) - \varepsilon_1 > 0$.

(2) For all n, Var$(T_n) = \frac{1}{n}($Var$_{x\sim p}(f_\theta(x)) +$ Var$_{x\sim q}(f_\theta(x))) \leq \frac{2}{n}B_\Theta^2$, B_Θ is a constant depending on the network function family Θ, as defined in (A.3)*.*

(3) Under $\mathcal{H}_0$, $\sqrt{n}T_n \to \mathcal{N}(0, \sigma^2_{\mathcal{H}_0})$ in distribution; Under $\mathcal{H}_1$, $\sqrt{n}(T_n - T[f_\theta]) \to \mathcal{N}(0, \sigma^2_{\mathcal{H}_1})$ in distribution; $\sigma^2_{\mathcal{H}_0} = 2Var_{x\sim p}(f_\theta(x))$, $\sigma^2_{\mathcal{H}_1} =$ Var$_{x\sim p}(f_\theta(x)) +$ Var$_{x\sim q}(f_\theta(x))$, both $\leq 2B_\Theta^2$.

We discuss the extension to less regular f^* after Proposition B.1, including the case where p and q nearly non-overlap. In particular, the more relevant situation for two-sample problem is where p and q weakly differ. The above Theorem directly gives the asymptotic test consistency:

Corollary 4.2. *Notations and settings as in Theorem 4.1, and for given p, q, the network architecture Θ has been fixed to satisfy $C = JSD(p,q) - \varepsilon_1 > 0$. Let $0 < \alpha < 1$ be the two-sample test level, typically $\alpha = 0.05$, and the test threshold be $\tau_n = \frac{\sigma_{\mathcal{H}_0}}{\sqrt{n}}\Psi^{-1}(\alpha)$, where $\Psi(x) := \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$, then, as $n \to \infty$, $\Pr[T_n > \tau_n | \mathcal{H}_0] \to \alpha$ and*

$$\Pr[T_n > \tau_n | \mathcal{H}_1] \to 1 - \Psi\left(\sqrt{n}\frac{T[f_\theta] - \tau_n}{\sigma_{\mathcal{H}_1}}\right) \geq 1 - \Psi\left(\frac{\sqrt{n}(4C - \tau_n)}{\sigma_{\mathcal{H}_1}}\right)$$

which $\geq 1 - c_0 e^{-c_1 n}$ for positive constants c_0 and c_1.

The analysis reveals that the critical regime for two-sample test is when the divergence between p and q is $\sim O(n^{-1/2})$. Actually, one may obtain a lower bound of testing power based on the bound of $\mathbb{E}T_n$ and $\mathrm{Var}(T_n)$ in Theorem 4.1 and Chebyshev inequality to control the large deviation. This will prove a finite-sample testing power which is positive and approaching 1 as long as $\mathrm{JSD}(p,q)$ exceeds a constant multiple of $n^{-1/2}$, where the constant depends on the choice of Θ which guarantees both small ε_1 and B_Θ. We omit the details here.

4.3 Reducing Network Complexity to Be Intrinsic

The above analysis is for general p and q in $\mathbb{R}^D$, and when D is large the needed network complexity grows exponentially. We now show that when p and q are on/near manifold $\mathcal{M}$ as in many applications, the network complexity can be reduced to be intrinsic, i.e., depending on the manifold only, and replacing D by the intrinsic dimension d in Theorem 4.1.

On-manifold densities. When both p and q are degenerate and support on the manifold $\mathcal{M}$, all the integrals above, $L[f]$ and $T[f]$, are carried out on the manifold only. Specifically, assume that $\mathcal{M}$ is compact smooth manifold without boundary, and p and q are smooth on $\mathcal{M}$ with respect to the Riemannian geometry, then the log ratio f^* is also smooth on $\mathcal{M}$. This allows to apply the manifold function approximation result in (Shaham et al., 2018) to obtain $\|f_{\text{con}} - f^*\|_{L^\infty(\mathcal{M})} < \varepsilon_1$ with needed network complexity of $O(\epsilon_1^{-d/2})$ (c.f. Theorem B.6). The rest of the proof remains the same by replacing all the integrals in $\Omega \subset \mathbb{R}^D$ to be on $\mathcal{M}$.

Near-manifold densities. The analysis extends when p and q decay sufficiently fast away from the manifold, as proved in Proposition B.5. This is because the manifold is locally near Euclidean and there exists differentiable one-to-one mapping between the manifold chart and the local tangent plane on local neighborhoods in $\mathbb{R}^D$ of radius $\delta > 0$, which is an absolute constant determined by the manifold $\mathcal{M}$. This allows to approximate integrals $L[f_{\text{con}}]$ and $L[f^*]$ in $\mathbb{R}^D$ by their counterparts on $\mathcal{M}$, using the off-manifold decay of the densities, and the two integrals on $\mathcal{M}$ are close due to uniform approximation of f^* on the manifold. Proofs in Appendix B.

5 Experiments

This section conducts numerical experiments of the proposed two-sample test and compares with alternatives, on synthetic 1D and manifold densities and evaluating hand-written digits generating models. Codes to produce all experiments will be publicly online.

5.1 Synthetic data

1D normal density departure. The following three examples all have $p = \mathcal{N}(0,1)$, and Eg.1. Mean shift, $q = \mathcal{N}(\delta, 1)$; Eg.2. Dilation of variance, $q = \mathcal{N}(0, (1+\delta)^2)$. Eg.3. Mixture with bump at tail, q as in (2). We examine the tests: (1) *net-acc*, (2) *net-logit*, (3) *gmmd* which sets σ to be median distance, (4) *gmmd-ad* which selects σ by maximizing kernel MMD discrepancy on the training set, and (5) *gmmd+*, (6) *gmmd++* as described in Sec. 3. (1)-(4) only use the test set when measuring the power, while (5)-(6) access both the training and test sets. More details about experimental setting-up in Appendix A.

The test powers of all the methods are plotted in Fig. 2 for the three examples. For Eg.1 and Eg.2, *gmmd+*, *gmmd++* are performing consistently better than the other four which only access the test data set, particularly in Eg.1. Eg.3 has been discussed in Sec. 3, and *net-logit* gives stronger power than *gmmd+*, *gmmd++* when $n_{\text{all}} > 200$, where *net-acc* remains inferior to the two. Note that *gmmd-ad* does worse than *gmmd*, that is, the median-distance choice of kernel bandwidth σ works better than the adaptive choice here. Among the four methods (1)-(4), the performances on Eg.1 are comparable, and *net-logit* gives better power on Eg.2 and Eg. 3. This is especially the case of Eg. 3, where *net-logit* shows the most significant advantage.

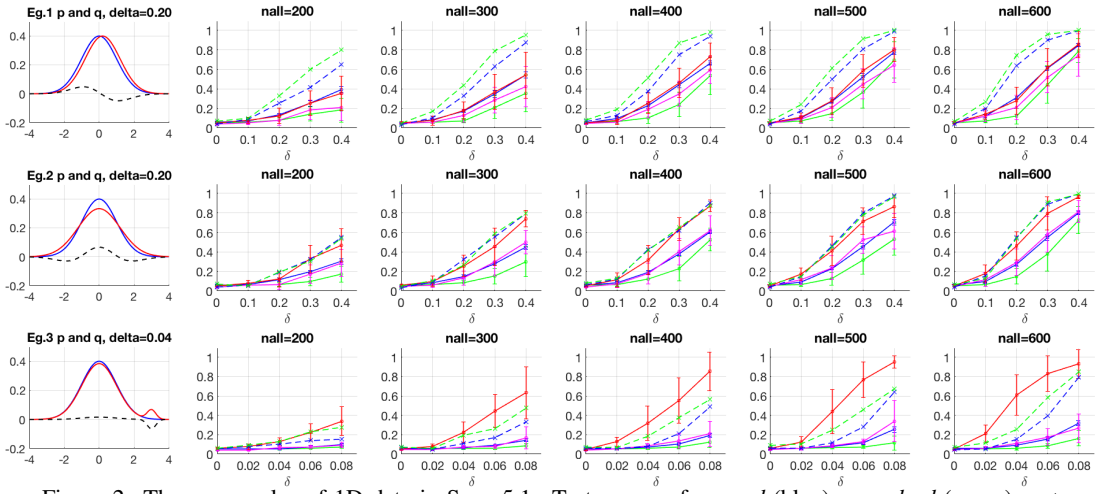


Figure 2: Three examples of 1D data in Sec. 5.1. Test power of: *gmmd* (blue), *gmmd-ad* (green), *net-acc* (pink), *net-logit* (red), error bar standing for the standard deviation of the estimated power over 20 replicas, and *gmmd+* (blue dash), *gmmd++* (green dash). $n_{all} = |X| + |Y|$ including half-half training-testing split.

Densities on a 2D manifold. The example consists of p and q which lie on the sphere S^2, a 2-dimensional manifold embedded in $\mathbb{R}^3$. A realization of samples X and Y is shown in the left of Fig. 3. Fig. 3 plots the test power of the 5 methods over increasing density departure δ and sample size. It can been seen that *net-logit* gives the fastest growth of power as δ increases and the strongest average power for all n_{all}, but the variation can be large if the power is not close to 1. Unlike some of the 1D cases, *gmmd* with median σ does not do better than the trivial power for all cases (blue solid), even with access to the full data samples *gmmd+* the power is only 0.2 with the largest n_{all} (blue dash). The adaptive choice by maximizing MMD discrepancy on training set improves the power significantly (*gmmd-ad*, green solid), but does not do as well as *net-acc*, which again performs inferior to *net-logit*. The optimal choice of σ (*gmmd++*, green dash) achieves better power than *net-acc* at $n_{\text{all}} = 200$ and comparable performance with larger n_{all}. *net-logit* performs better than *gmmd++* and the advantage is more evident when $n_{\text{all}} > 200$. This indicates that larger sample size can be particularly in favor of network-based tests, which rely on the search in the network parameter space optimized on a separated training set.

5.2 Generated vs authentic MNIST data

As a real-world data example, we study the task of distinguishing "faked" MNIST samples produced by a pre-trained generative network from authentic ones. The MNIST dataset consists of gray-scale hand-written digits of size 28×28 falling into 10 classes, which is relatively simple and thus is viewed to lie near to low-dimensional manifolds in the ambient space of $\mathbb{R}^{784}$. The classifier network used in *net-logit* test is a convolutional neural network (CNN) with 2 convolutional layers. More details about the generative and classification networks in Appendix A.

We compare (1) *net-acc* (2) *net-logit* (3) *gmmd* (4) *gmmd-ad* on two samples X and Y, half of $\mathcal{D} = X \cup Y$ used for training. X consists of authentic MNIST samples, and Y of a mixture of authentic and faked ones, i.e. $p = p_{\text{data}}$ and $q = (1-\delta)p_{\text{data}} + \delta p_{\text{model}}$, $\delta \in [0, 1]$. The test power is evaluated on 400 test runs and the training is repeated for 20 replicas. The results for increasing δ and sample size $n_{all} = |\mathcal{D}|$ up to 500 is shown in Fig. 5, where *net-logit* gives the strongest power throughout all cases, and the two network-based tests significantly outperforms the other two when $n_{all} \geq 300$. The adaptive choice of kernel bandwidth also improves the power over the median-distance choice, shown by the better power of *gmmd-ad* than *gmmd*. The standard deviation of the *net-acc* and *net-logit* power is less than that of *gmmd-ad* power when $n_{\text{all}} = 300$ and $\delta \geq 0.4$, when the former two give near to 1 power. We also observe that the training of the CNN classifier in this experiment is more stable than that of the previous fully-connected network on low-dimensional synthetic data, as revealed in the training error evolution plots, c.f. Fig. A.1 Fig. A.5. With another pre-trained model which generates faked images that are closer to authentic ones, *net-logit* again

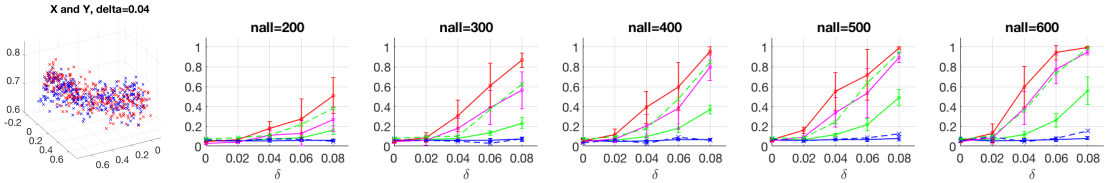
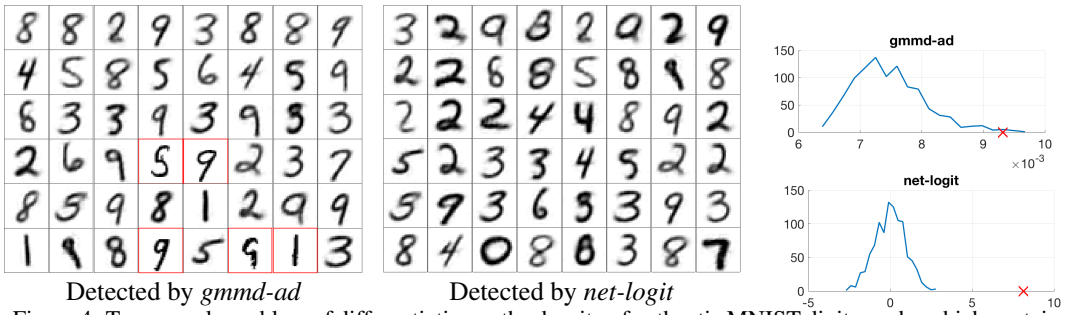


Figure 3: Test power of the different tests on data on sphere in $\mathbb{R}^3$ in Sec. 5.1. Markers same as in Fig. 2.



Detected by *gmmd-ad* Detected by *net-logit*

Figure 4: Two-sample problem of differentiating p, the density of authentic MNIST digits, and q which contains a $\delta = 0.4$ fraction of digits "faked" by a generative model. $|X| = |Y| = 500$. The *gmmd-ad* and *net-logit* tests use half as training set, and test on the other $|\mathcal{D}_{\text{te}}| = 500$ samples. Left and middle: the most likely fake digits identified by the empirical witness functions of the two tests, red box indicates authentic digits incorrectly identified. Right: The test statistic $\hat{T}$ ($\mathcal{H}_1$) and the histogram of its value under 1000 permutation tests ($\mathcal{H}_0$).

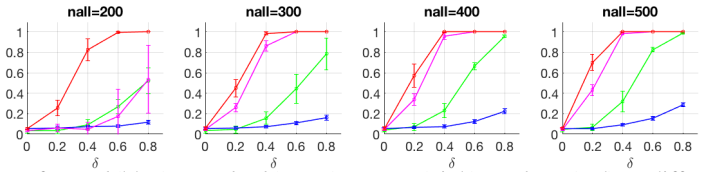


Figure 5: Test power of *gmmd* (blue) *gmmd-ad* (green) *net-acc* (pink) *net-logit* (red) on differentiating authentic vs synthesized MNIST digits produced by a generative model, where sample X has all authentic ones, and δ stands for the fraction of synthesized ones in Y, $n_{all} = |X| + |Y|$ including half-half training-testing split.

shows the best discriminative power, *net-acc* gives comparable performance starting $n_{\text{all}} = 300$, while *gmmd* and *gmmd-ad* gives trivial power up to $n_{all} = 500$, c.f. Fig. A.4.

Setting $n_{\text{all}} = 1000$, $\delta = 0.4$, the results of *gmmd-ad* and *net-logit* in one test run is shown in Fig. 4. Based on the $n_{\text{all}} = 500$ plot in Fig. 5, both tests shall have non-trivial power, and that of *net-logit* shall be close to 1. In this test, both methods correctly rejects $\mathcal{H}_0$, yet the *net-logit* statistic deviates from the distribution of $\hat{T}|\mathcal{H}_0$ more significantly, indicating stronger power (shown in the histogram plots). To compare the detecting ability of the empirical witness function $\hat{w}$ of *gmmd-ad* and *net-logit*, for each method, we sort the 250 samples in Y_{te} (among which 100 are faked ones) in ascending order of the value of $\hat{w}$ and select the first 100 samples. These are samples which the model views as most likely to be faked ones. The success rate of identifying faked samples is ~ 60 by *gmmd-ad* $\hat{w}$, and ~ 90 by *net-logit* $\hat{w}$. The first 48 most likely faked digits identified with both witness functions are plotted in Fig. 4, where *gmmd-ad* $\hat{w}$ incorrectly includes 5 authentic samples, and none by *net-logit* $\hat{w}$.

6 Conclusion

The paper proposes to use estimated log ratio to compute a two-sample statistic once a classification network has been trained on a split training set. The proposed statistic empirically demonstrates stronger testing power than previously studied neural network classifier tests based on classification accuracy. It also compares favorable to gaussian kernel-based MMD in certain settings, especially for higher dimensional data, including distinguishing generated MNIST digits from authentic ones. The proposed test has more advantage with large samples, due to that larger training set makes the training more stable, as well as its linear computational complexity and scalable algorithm. Theoretically, we prove the power of the proposed test when the network is sufficiently parametrized, and reduce the needed network complexity to be intrinsic when p and q lie on or sufficiently near to low-dimensional manifolds in possibly high-dimensional space,

The analysis in this paper gives a positive result towards justifying the power of two-sample tests based on training a classification network. However, the proof is based on a network approximation analysis and optimization of population loss, which means that the derived testing power only applied when the training perfectly identifies the global optimizer, a situation not necessarily happening in practice. In experiments we observe that the performance of the network-based tests has larger variance than traditional methods like kernel MMD, due to the instability of the training, particularly with small training size. Thus, more understanding of the network optimization, which is not addressed in the current paper, is needed so as to better understand network classification two-sample tests and to develop better methods. At last, we have not systematically explored the effects of different network architecture on the performance, which surely has an influence.

REFERENCES

Niall H Anderson, Peter Hall, and D Michael Titterington. Two-sample test statistics for measuring discrepancies between two multivariate probability density functions using kernel-based density estimates. *Journal of Multivariate Analysis*, 50(1):41–54, 1994.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pp. 214–223, 2017.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative learning for differing training and test distributions. In *Proceedings of the 24th international conference on Machine learning*, pp. 81–88. ACM, 2007.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(Sep):2137–2155, 2009.

Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22 (14):e49–e57, 2006.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer, 2010.

Xiuyuan Cheng, Alexander Cloninger, and Ronald R Coifman. Two-sample statistics based on anisotropic kernels. *arXiv preprint arXiv:1709.05006*, 2017.

Kacper Chwialkowski, Heiko Strathmann, and Arthur Gretton. A kernel test of goodness of fit. JMLR: Workshop and Conference Proceedings, 2016.

Kacper P Chwialkowski, Aaditya Ramdas, Dino Sejdinovic, and Arthur Gretton. Fast two-sample testing with analytic representations of probability measures. In *Advances in Neural Information Processing Systems*, pp. 1981–1989, 2015.

Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018.

Jerome Friedman. On multivariate goodness-of-fit and two-sample testing. Technical report, Stanford Linear Accelerator Center, Menlo Park, CA (US), 2004.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012a.

Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems*, pp. 1205–1213, 2012b.

James J Higgins. Introduction to modern nonparametric statistics. 2003.

Wittawat Jitkrittum, Zoltán Szabó, Kacper P Chwialkowski, and Arthur Gretton. Interpretable distribution features with maximum testing power. In *Advances in Neural Information Processing Systems*, pp. 181–189, 2016.

Wittawat Jitkrittum, Wenkai Xu, Zoltán Szabó, Kenji Fukumizu, and Arthur Gretton. A linear-time kernel goodness-of-fit test. In *Advances in Neural Information Processing Systems*, pp. 262–271, 2017.

Takafumi Kanamori, Taiji Suzuki, and Masashi Sugiyama. f-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models. *IEEE Transactions on Information Theory*, 58(2): 708–720, 2011.

Takafumi Kanamori, Taiji Suzuki, and Masashi Sugiyama. f-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models. *IEEE Transactions on Information Theory*, 58(2): 708–720, 2012.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Erich L Lehmann and Joseph P Romano. *Testing statistical hypotheses*. Springer Science & Business Media, 2006.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network. In *Advances in Neural Information Processing Systems*, pp. 2203–2213, 2017.

Yujia Li, Kevin Swersky, and Rich Zemel. Generative moment matching networks. In *International Conference on Machine Learning*, pp. 1718–1727, 2015.

Qiang Liu, Jason Lee, and Michael Jordan. A kernelized stein discrepancy for goodness-of-fit tests. In *International Conference on Machine Learning*, pp. 276–284, 2016.

James R Lloyd and Zoubin Ghahramani. Statistical model criticism using kernel two sample tests. In *Advances in Neural Information Processing Systems*, pp. 829–837, 2015.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*, 2016.

Aditya Menon and Cheng Soon Ong. Linking losses for density ratio and class-probability estimation. In *International Conference on Machine Learning*, pp. 304–313, 2016.

Shakir Mohamed and Balaji Lakshminarayanan. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pp. 271–279, 2016.

Aaditya Ramdas, Aarti Singh, and Larry Wasserman. Classification accuracy as a proxy for two sample testing. *arXiv preprint arXiv:1602.02210*, 2016.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

Mark D Reid and Robert C Williamson. Information, divergence and risk for binary experiments. *Journal of Machine Learning Research*, 12(Mar):731–817, 2011.

Robert J Serfling. Approximation theorems of mathematical statistics, 1981.

Uri Shaham, Alexander Cloninger, and Ronald R Coifman. Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis*, 44(3):537–557, 2018.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, pp. 71–79, 2013.

Bharath K Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Gert RG Lanckriet. On integral probability metrics,\phi-divergences and binary classification. *arXiv preprint arXiv:0901.2698*, 2009.

Masashi Sugiyama, Takafumi Kanamori, Taiji Suzuki, Marthinus Christoffel du Plessis, Song Liu, and Ichiro Takeuchi. Density-difference estimation. *Neural Computation*, 25(10):2734–2775, 2013.

Dougal J Sutherland, Hsiao-Yu Tung, Heiko Strathmann, Soumyajit De, Aaditya Ramdas, Alex Smola, and Arthur Gretton. Generative models and model criticism via optimized maximum mean discrepancy. *arXiv preprint arXiv:1611.04488*, 2016.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pp. 1139–1147, 2013.

Max Wornowizki and Roland Fried. Two-sample homogeneity tests based on divergence measures. *Computational Statistics*, 31(1):291–313, 2016.

Dmitry Yarotsky. Error bounds for approximations with deep relu networks. *Neural Networks*, 94:103–114, 2017.

Dmitry Yarotsky. Optimal approximation of continuous functions by very deep relu networks. *arXiv preprint arXiv:1802.03620*, 2018.

Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

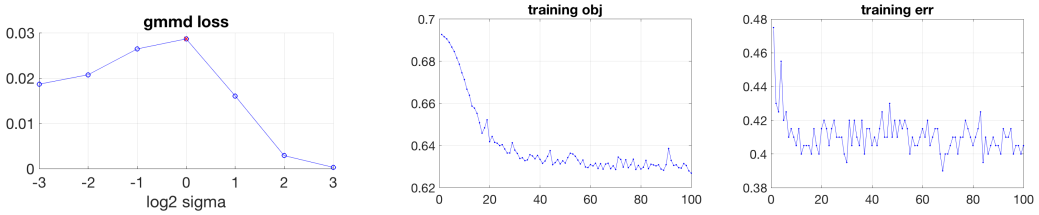


Figure A.1: Left: MMD discrepancy on trained set used by *gmmd-ad* to select kernel bandwidth σ. Middle and right: training of the classification network used in net-based tests. On the synthetic 1D dataset.

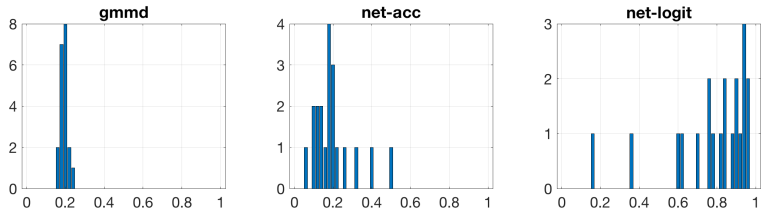


Figure A.2: Histogram of estimated test power from 400 test runs of *gmmd*, *net-acc* and *net-logit* over 20 replicas of training (no training for *gmmd*), on the example in Sec. 3, Fig. 1.

A Details of Experiments

A.1 Training in Sec. 3 and 5.1

Training of networks. In all the experiments, the classifier network used by *net-acc* and *net-logit* is a two-layer fully-connected neural network with 32 hidden nodes in each hidden layer, and the bottom layer has the same dimension as the input data. The training of the network is conducted via 100 epochs of Adam with learning rate 10^{-3}, and batch size 100 when the size of training set > 100. A typical plot of evolution of training loss and training error is given in Fig. A.1. Training via SGD with momentum 0.9 produces similar result. The result is qualitatively the same when the number of hidden units varies from 16 to 1024. We have not investigated the optimal choice of network architecture hyperparameters for the two-sample problem.

Adaptive choice of σ in *gmmd-ad*. In the training phase, the algorithm computes the gaussian kernel MMD discrepancy

$$\hat{T}_{\text{MMD}}(X,Y) = \frac{1}{|X|^2}\sum_{x,x'\in X} k_\sigma(x,x') + \frac{1}{|Y|^2}\sum_{y,y'\in Y} k_\sigma(y,y') - \frac{2}{|X||Y|}\sum_{x\in X,\, y\in Y} k_\sigma(x,y)$$

on the training set $X = X_{\text{tr}}$, $Y = Y_{\text{tr}}$, for a range of values of the kernel bandwidth σ, i.e. $\sigma = \{2^{-3}, \cdots, 2^3\}$. $k_\sigma(x,y) = \exp\{-\frac{|x-y|^2}{2\sigma^2}\}$ is the isotropic gaussian kernel. A plot of MMD discrepancy as a function of varying σ is given in Fig. A.1. The σ which maximizes the MMD discrepancy on the training set is then chosen to compute the test statistic on the test set. The method is equivalent to the training process in (Li et al., 2017) with only one trainable parameter which is the kernel bandwidth σ. The MMD test statistic also takes the form as $\hat{T}_{\text{MMD}}$ in (Gretton et al., 2012a; Li et al., 2017).

A.2 More details about results in Sec. 3

Test power. The way of computing the test power is empirical and has randomness: for kernel mmd the variation is due to the finite number of runs (n_{run} times), and for network based tests there is extra variation due to the stochastic optimization of the network. Thus we use experiment replicas to recored the variations of the test power.

The empirical distribution of the test power of the three methods over training replicas is given in A.2, corresponding to the experiment in Fig. 1. The plots of the two net-based methods indicate large variation of the power given by each trained network, that is, the "quality" of the trained net to discriminate the two densities varies. This instability is due to limited training samples as well as the randomness in the optimization algorithm. We observe decreased power variation with larger

	gmmd	*gmmd+*	*gmmd++*	*net-acc*	*net-logit*
mean	19.14	46.63	57.29	19.98	78.09
std	1.95	2.49	1.578	10.43	20.56
median	19.63	47.13	57.38	17.63	84.13

Table A.1: The mean, standard deviation ("std") and median of the test power of the various methods computed from $n_{run} = 400$ test runs over $n_{\text{rep}} = 20$ replicas on the 1D example in Sec. 3. The *gmmd*, *net-acc*, *net-logit* tests are computed on $|X_{\text{te}}| = |Y_{\text{te}}| = 100$ samples, where *net-acc* and *net-logit* train a classification network on another training set of size $|X_{\text{tr}}| = |Y_{\text{tr}}| = 100$. *gmmd* only uses the test set and sets the kernel bandwidth σ to be the median distance. *gmmd+* and *gmmd++* accesses both the training and test sets, where *gmmd+* uses the median distance as σ, and *gmmd++* reports the best power over varying range of choices of σ, as described in Sec. 3. The results of *gmmd*, *net-acc*, *net-logit* are also reported in Fig. 1.

training set, where the stochastic optimization converges to solutions of lower error and the resulted net gives better two-sample test power. However, the two-sample problem itself is expected to be easier with larger n too.

Table A.1 gives the full table of test power including that of the methods *gmmd+* and *gmmd++*.

Equivalent form of *net-acc* test. Here we show that the *net-acc* test studied in (Lopez-Paz & Oquab, 2016) is equivalent to using $\text{Sign}(f_\theta)$ instead of f_θ in (1) when $n_X = n_Y$, up to multiplying and adding constants. Specifically, by the definition of test statistic in (Lopez-Paz & Oquab, 2016), and recall that $|X_{te}| = |Y_{te}| = \frac{1}{2}|\mathcal{D}_{te}|$, $\text{Sign}(z) = 1$ if $z \geq 0$ and -1 if $z < 0$,

$$\begin{aligned}
\hat{T}_{\text{net-acc}} &= \frac{1}{2}\left(\frac{1}{|X_{te}|}\sum_{x\in X_{te}} \mathbf{1}_{\{f_\theta(x)\geq 0\}} + \frac{1}{|Y_{te}|}\sum_{y\in Y_{te}} \mathbf{1}_{\{f_\theta(y)<0\}}\right) \\
&= \frac{1}{2}\left(\frac{1}{|X_{te}|}\sum_{x\in X_{te}} \frac{1}{2}(1+\text{Sign}(f_\theta(x))) + \frac{1}{|Y_{te}|}\sum_{y\in Y_{te}} \frac{1}{2}(1-\text{Sign}(f_\theta(x)))\right) \qquad \text{(A.1)} \\
&= \frac{1}{2} + \frac{1}{4}\left(\frac{1}{|X_{te}|}\sum_{x\in X_{te}} \text{Sign}(f_\theta(x)) - \frac{1}{|Y_{te}|}\sum_{y\in Y_{te}} \text{Sign}(f_\theta(x))\right).
\end{aligned}$$

Quantitative comparison of mean and std of test statistics. Let w be the population witness function of the three methods respectively, and define

$$\begin{aligned}
\textbf{Mean} &:= \mathbb{E}_{x\sim p, Y\sim q}(w(X) - w(Y)), \\
\textbf{Std} &:= \sqrt{\text{Var}_{x\sim p}(w(X)) + \text{Var}_{Y\sim q}(w(Y))}.
\end{aligned}$$

For tests using $\hat{T} = \int w(\hat{p} - \hat{q})$ as the statistic, such as in *net-logit* and *net-acc*, by independence of the samples the mean and variance of $\hat{T}$ are **Mean** and $\textbf{Std}/\sqrt{n}$ respectively. For kernel MMD, the actually test statistic is computed via quadratic sums, however the mean remains the same, and $\textbf{Std}/\sqrt{n}$ will be a lower bound of the standard deviation of the MMD statistic (Serfling, 1981). Strictly speaking, the test statistic is computed from empirical rather than the population witness function. For *net-logit* and *net-acc*, considering population witness function is as if the training is able to identify the exact optimizer which lies inside the representable function family of the network, an idealized scenario. With this idealization, the relation of **Mean** and **Std** to the test power can be made rigorous making use of the asymptotic normality of the test statistic (as independent sums), as done in Section 4. The conclusion gives that the larger the **Mean**, and the smaller the **Std**, the more powerful the test is going to be. For kernel MMD, these two quantities similarly indicate the testing power, see e.g. (Serfling, 1981; Cheng et al., 2017). Thus we will use **Mean** and **Std** for all three methods for comparison.

To remove the scaling equivalence of test statistics (a test statistic multiplied by a positive constant gives the same test power), we will use the ratio of **Mean** and **Std** as an indicator of test power. For the 1D example in Section 3, due to the explicit formula of p and q the values of **Mean** and **Std** can be analytically computed, which are shown in Table A.2. The *net-logit* gives the largest ratio in this comparison. The normalized witness functions are plotted in Fig. A.3, where a constant

	Mean	**Std**	**Mean/Std**
gmmd	0.0087	0.0421	0.2075
net-acc	0.1579	0.6087	0.2594
net-logit	0.2445	0.9011	0.2714

Table A.2: The values of **Mean**, **Std**, and their ratio of the three tests, where p and q are as in the 1D example in Sec. 3, c.f. Fig. 1.

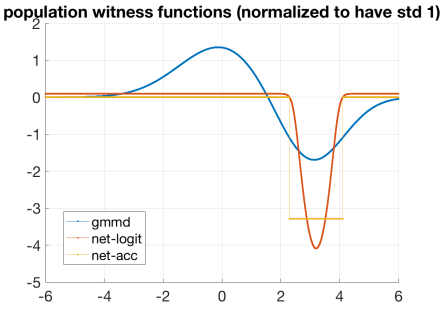


Figure A.3: Plots of the population witness functions normalized to have **Std** $= 1$, of the three tests in Sec. 3, c.f. Fig. 1, Table A.2.

is multiplied to each w respectively to enforce **Std** $= 1$. It can be seen that the *net-logit* witness function gives the largest weights to the differential region of p and q in this example.

A.3 Experimental details in Section 5.1

1D normal density departure experiment. The experiments with (1)-(4) use 400 test runs to estimate the power, and are repeated for 20 replicas. The test with (5) and (6) uses 200 test runs to estimate the power, since these gmmd methods demonstrate less variation in estimated power. Training and testing split is half-and-half in all cases.

Densities p and q on the 2D manifold. The construction of $x_i \sim p$ and $y_j \sim q$ are as below: $x_i = T(u_i), y_j = T(v_j)$ where $T : \mathbb{R}^2 \to \mathbb{R}^3$ is a smooth mapping from unit square to the spherical surface given by

$$T(x_1, x_2) = \frac{1}{R}\left(x_1, x_2, \sqrt{R^2 - x_1^2 - x_2^2}\right), \quad R = 1.5,$$

and u_i, v_j are i.i.d. copies of random variables u and v in $\mathbb{R}^2$ distributed as

$$u = t_u + \eta_u, \quad v = t_v + \eta_v, \quad \eta_u, \eta_v \sim \mathcal{N}(0, \epsilon^2 I_2), \quad \epsilon = 0.05,$$
$$t_u \sim \text{ uniformly on a quater circle in } [0,1] \times [0,1],$$
$$t_v \sim \text{ the distribution of } t_u \text{ rotated around } (\tfrac{1}{\sqrt{2}}, \tfrac{1}{\sqrt{2}}) \text{ by angle } \delta,$$

where the 4 random variables are all independent.

A.4 Experiment on MNIST data in Sec. 5.2

The pre-trained generated model is based on a convolutional auto-encoder:

c5x5x1x16 - re - ap 2x2 - c5x5x16x32 - re - ap 2x2 - fc128 - re
- fc10 - re $\quad \leftarrow$ **code space** $\mathbb{R}^{10}$
- fc128 - re - ct 5x5x128x32 - re
- ct5x5x32x16 (upsample 2x2) - re - ct5x5x16x1 (upsample 2x2) - Euclidean loss

where "c" stands for convolutional layer, "ct" for transposed convolutional layers, "re" for Relu activation, and "ap" for average pooling. The auto-encoder is trained on 50000 MNIST dataset for 20 epochs using Adam with learning rate decreasing from 10^{-3} to 10^{-6} and batch size 100.

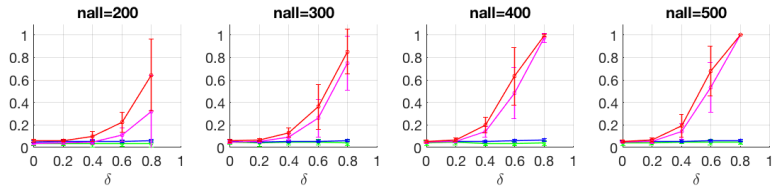


Figure A.4: Same plot as Fig. 5 with another pre-trained generative model which produces faked images that are closer to the authentic ones.

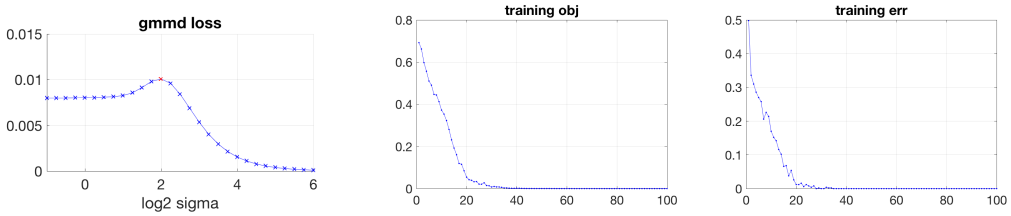


Figure A.5: Same plot as Fig. A.1 on MNIST data.

The sampling of generative model is conducted by adding a small isotropic gaussian noise ("giggering") to the 10-dimensional codes of authentic MNIST digits computed by an encoder, and then mapping through the decoder to $\mathbb{R}^{784}$.

We also prepare another generative model by removing the bottleneck layer in the above auto-encoder architecture and retrain the model, which gives smaller reconstruction error and a higher-dimensional code space of $\mathbb{R}^{128}$. The generative model is conducted in the same way by sampling in the code space using gaussian noise of smaller variance per coordinate. This produces faked images that are closer to the authentic ones in Euclidean distance in $\mathbb{R}^{784}$, however less explore the "manifold" of p_{data}. The test power of the four methods is shown in Fig. A.4.

The classification network used in *net-logit* is the following CNN

c5x5x1x16 - re - ap 2x2
- c5x5x16x32 - re - ap 2x2
- fc128 - re - fc2 - softmax loss

where dropout is used between the last 2 fully-connected layers. The classification CNN is trained for 100 epochs using Adam with learning rate 10^{-3} and batch size 100. A typical plot of evolution of training loss and training error is given in Fig. A.5.

The procedure of adaptive selection of σ by *gmmd-ad* is same as in Sec. 5.1, where the bandwidth search range is $\sigma = \{2^{-1}, \cdots, 2^{6}\}$.

B Proofs and Details of the Analysis in Section 4

B.1 Proofs and Details for Theorem 4.1

B.1.1 Construction of f_{con} by network

Suppose that p and q are non-vanishing inside Ω, and let $p = e^u$ and $q = e^v$ for smooth potential functions u and v, then $f^* = u - v$ is also smooth. We first assume that f^* is has properly bounded derivatives, then standard network approximation theory, e.g. that in (Yarotsky, 2017), guarantees that one can approximate f^* by a network output function f_{con} of a multi-layer fully-connected network with

$$\|f_{\text{con}} - f^*\|_{L^\infty(\Omega)} = O(N^{-r/D})$$

where N is the number of parameters in the network, and the constant depends on the regularity of r-th derivative of f^*. This leads to the following guaranteed positive lower bound of $L[f_{\text{con}}]$:

Proposition B.1. *Let the densities $p = e^u$, $q = e^v$ be C^r and supported on Ω, the unit ball in $\mathbb{R}^D$, and $u, v \in C^r(\Omega)$, $f^* = u - v$. If $p \neq q$ such that JSD$(p, q) > 0$, then for any $0 < \varepsilon_1 <$ JSD(p, q),*

there is a neural network architecture Θ with $O(\varepsilon_1^{-D/r})$ many trainable parameters and $f_{con} \in \mathcal{F}_\Theta$ such that

$$L[f_{con}] > \textit{JSD}(p, q) - \varepsilon_1 := C > 0.$$

Note that C can be made arbitrarily close to largest possible value of $L[f]$ which is $\text{JSD}(p, q)$ given sufficient network complexity. More recent approximation result which improves the approximation rate, such as (Yarotsky, 2018), will improve the complexity needed accordingly.

We now consider the case that f^* is less regular in terms of having derivatives of larger magnitude. While mathematically the previous argument is still valid, the worse constant in the approximation bound indicates difficulty for the network to approximate such f^*. This can be improved by observing that in the previous proof it suffices to make $\|f^* - f_{\text{con}}\|_{L^1}$ (with the measure p and q) small rather than in the L^∞ norm. This allows, e.g., using the network to approximate $\tilde{f}^*$, which is a regularized version of f^*, and thus can be more efficiently parametrized by the network, as long as $\|f^* - \tilde{f}^*\|_{L^1} < \varepsilon_2$, and it will give $L[f_{\text{con}}] > \text{JSD} - \varepsilon_2 - \varepsilon_1$. This argument will extend to the case where $f^* = u - v$ has places of discontinuity or other singularity as long as such places are of small measure under $(p + q)$.

A specific type of singularity of f^* is when its magnitude diverges to ∞ or very large. This happens when p almost vanishes on a region where q takes significantly large value and vice versa. The nearly divergent value of f^* surely creates a difficulty for network approximation, however, in this case f^* again can be replaced by a bounded and regularized version $\tilde{f}^*$, at least locally, which will produce an a comparably large $L[\tilde{f}^*]$. We then approximate $\tilde{f}^*$ by the network function f_{con} and obtains $L[f_{\text{con}}] > C > 0$ for some sufficiently large C.

When the regions where p and q nearly non-overlap are large, the above argument may lead to C much less than $\text{JSD}(p, q)$. However, note that such situation is actually a trivial case for two-sample testing: if p and q already differ significantly then many test statistics will give strong testing power. Because that two-sample test is trying to distinguish differential densities reliably with as small number of samples as possible, the the more interesting situation is the "weak-separation" regime of p and q, that is, the departure of q from p is small and then f^* is not far from zero.

Due to above reasons, in what follows we focus on f^* which is regular, bounded and has properly bounded derivatives. We will see that the critical regime for two-sample detection is when the magnitude of $(p - q)$ and thus that of $f^* \sim O(n^{-1/2})$ which is asymptotically 0 as n increases.

B.1.2 Bounding $T[f]$ by $L[f]$

Lemma B.2. *For any f so that the integrals are defined, $T[f] \geq 4L[f]$.*

The relaxation in Lemma B.2 may not be sharp, particularly, when p and q nearly non-overlap on their supports, $T[f^*] = 2\text{SKL}(p, q)$ diverges to infinity, while $L[f^*]$ remains bounded (by $2 \log 2$). When f is close to zero, which as discussed above is the more relevant scenario for two-sample test, the following lemma quantifies the tightness of the relaxation

Lemma B.3. *For any f s.t. f^2 is integrable w.r.t p and q,*

$$0 \leq \frac{1}{2}T[f] - 2L[f] \leq \int (p + q)\frac{f^2}{2}.$$

B.1.3 Bounding the variance of T_n

By the definition of T_n and the independence of X_i's and Y_i's, the random variable T_n is asymptotically normal by Central Limit Theorem, and

$$\text{Var}(T_n) = \frac{1}{n}(\text{Var}_{X\sim p}(f(X)) + \text{Var}_{Y\sim q}(f(Y))). \tag{A.2}$$

The following Proposition proves that both of the variances of $f(X)$ and $f(Y)$ are bounded by $O(1)$ constants depending on the network function family.

Proposition B.4. *Given network function family $\mathcal{F}_\Theta$, suppose that*

$$B_\Theta^{(0)} = \sup_{f\in\mathcal{F}_\Theta} \|f\|_{L^\infty(\Omega)}, \quad B_\Theta^{(1)} = \sup_{f\in\mathcal{F}_\Theta} \textit{Lip}(f),$$

are both finite, Ω is the unit sphere in $\mathbb{R}^D$, where densities p and q are supported on. Then for any $f \in \mathcal{F}_\Theta$,

$$\textit{Var}_{x\sim p}(f(x)),\ \textit{Var}_{x\sim q}(f(x)) \leq \min\{2B_\Theta^{(0)}, B_\Theta^{(1)}\}^2 := B_\Theta^2. \tag{A.3}$$

The boundedness of $B_\Theta^{(0)}$ and $B_\Theta^{(1)}$ relies on the regularization of the network function family $\mathcal{F}_\Theta$. For a given Θ, $B_\Theta^{(0)}$ and $B_\Theta^{(1)}$ may be very large compared to $\|f\|_{L^\infty(\Omega)}$ and $Lip(f)$ of the trained f_θ, which leads to a loose upper bound of the variance. In practice, regularization techniques may lead to smaller values of $|f|$ and $Lip(f)$ while at a price of smaller $L[f]$, revealing the trade-off between stability of the test statistic and the sensitivity to detect differential density departure at large samples. We do not further pursue this problem in the current paper.

B.1.4 PROOFS

Proof of Proposition B.1. Under the assumptions, f^* can by approximated by $f_{con} \in \mathcal{F}_\Theta$ such that

$$\|f_{\text{con}} - f^*\|_{L^\infty(\Omega)} \leq \varepsilon_1.$$

Writing f_{con} as f, we then have that

$$\begin{aligned}
|L[f] - L[f^*]| &\leq \frac{1}{2}\left|\int_\Omega p\left(\log\frac{2e^f}{1+e^f} - \log\frac{2e^{f^*}}{1+e^{f^*}}\right) + \int_\Omega q\left(\log\frac{2}{1+e^f} - \log\frac{2}{1+e^{f^*}}\right)\right| \\
&\leq \frac{1}{2}\left(\int_\Omega p(x)|f(x) - f^*(x)| + \int_\Omega q|f(x) - f^*(x)|\right), \\
&\leq \varepsilon_1,
\end{aligned}$$

where the second inequality uses that $\log\frac{e^\xi}{1+e^\xi}$ and $\log(1+e^\xi)$ are Lip 1. This implies the claim. □

Proof of Lemma B.2. By definition,

$$\begin{aligned}
2L[f] &= \int p\log\frac{2}{1+e^{-f}} + \int q\log\frac{2}{1+e^f} \\
&= -\int p\log\frac{1+e^{-f}}{2} - \int q\log\frac{1+e^f}{2} \\
&\leq -\int p\log e^{-\frac{f}{2}} - \int q\log e^{\frac{f}{2}} \quad \text{(for any real number } \xi,\ \frac{1+e^\xi}{2} \geq e^{\frac{\xi}{2}}) \\
&= \int \frac{f}{2}(p-q) = \frac{1}{2}T[f].
\end{aligned}$$

□

Proof of Lemma B.3.

$$\begin{aligned}
\frac{1}{2}T[f] - 2L[f] &= \int p\frac{f}{2} - \int q\frac{f}{2} - \int p\log\frac{2e^f}{1+e^f} - \int q\log\frac{2}{1+e^f} \\
&= \int p\log\frac{1+e^f}{2e^{f/2}} + \int q\log\frac{1+e^f}{2e^{f/2}} \\
&= \int (p+q)\log\frac{e^{-f/2}+e^{f/2}}{2},
\end{aligned}$$

and by that $\frac{e^x+e^{-x}}{2} \leq e^{x^2/2}$,

$$\frac{1}{2}T[f] - 2L[f] \leq \int (p+q)\log e^{f^2/2} = \int \frac{f^2}{2}(p+q).$$

□

Proof of Proposition B.4. Let $L = Lip(f)$,

$$\begin{aligned}\text{Var}_{x\sim p}(f(x)) \le \mathbb{E}_x|f(x)-f(0)|^2 &= \int_\Omega |f(x)-f(0)|^2 p(x)dx \\ &\le L^2\int_\Omega |x|^2 p \le L^2\int_\Omega p = L^2.\end{aligned}$$

This proves that $\text{Var}_{x\sim p}(f(x)) \le (B_\Theta^{(1)})^2$ as $L \le B_\Theta^{(1)}$. Meanwhile,

$$\int_\Omega |f(x)-f(0)|^2 p(x)dx \le (2\|f\|_{L^\infty(\Omega)})^2 \le (2B_\Theta^{(0)})^2,$$

which proves the other upper bound. Same proof for q. □

Proof of Theorem 4.1. Using Proposition B.1 with $r = 2$, there exists $f_{\text{con}} \in \mathcal{F}_\Theta$ such that $L[f_{\text{con}}] > C > 0$. The optimization (4) then gives that $L[f_\theta] \ge L[f_{\text{con}}] > C$. Lemma B.2 then gives that $T[f_\theta] \ge 4L[f_\theta] > 4C$. This proves (1). (2) and (3) follow from (A.2) and Proposition B.4, and Central Limit Theorem. □

B.2 Extension to near-manifold densities

Like before, suppose f_θ is the minimizer of population training loss, and $f_{\text{con}} \in \mathcal{F}_\Theta$ is to be constructed to approximate the log density ratio f^*. The Step 2 and 3 of proving Theorem 4.1 remain the same, thus it suffices to establish the "manifold intrinsic complexity" version of Proposition B.1, which is the following

Proposition B.5. *Let the densities $p = e^u$, $q = e^v$ be C^2 and supported on Ω, the unit ball in $\mathbb{R}^D$, and $u, v \in C^2(\Omega)$, $f^* = u - v$. Let $\mathcal{M} \subset \Omega$ be a compact smooth manifold of dimension d, and p, q decay exponentially fast away from $\mathcal{M}$, that is, p, $q \in \mathcal{P}_\sigma$ defined to be*

$$\mathcal{P}_\sigma = \{p \text{ smooth density supported on } \Omega, \text{ s.t. } Pr_{X\sim p}[d(X,\mathcal{M}) > t] \le c_1 e^{-c_2\frac{t}{\sigma}}\}, \tag{A.4}$$

where $d(x,\mathcal{M}) := \inf_{y\in\mathcal{M}}\|x-y\|_2$ for any $x \in \mathbb{R}^D$, and c_1, c_2 are absolute positive constants. We will need σ to be a small constant. Suppose $p \ne q$, JSD$(p,q) > 0$, then for any $\varepsilon_1 > 0$, there is a neural network architecture Θ with $O(\varepsilon_1^{-d/2})$ many trainable parameters and $f_{con} \in \mathcal{F}_\Theta$ such that

$$L[f_{con}] > \text{JSD}(p,q) - (10\varepsilon_1 + c(f^*,\Theta,\mathcal{M})\sigma) := C > 0,$$

where we need ε_1 and σ to be small enough to guarantee that

$$\tilde{c}_4(\mathcal{M})\sigma < 9, \quad 10\varepsilon_1 + c(f^*,\Theta,\mathcal{M})\sigma < \text{JSD}(p,q)$$

where $\tilde{c}_4$ is a constant determined by the manifold and atlas, $c(f^*,\Theta,\mathcal{M})$ a constant determined by f^*, Θ, and manifold atlas.

Again we only consider sufficiently regular f^* which has properly bounded 2nd derivative, by the comments below Proposition B.1.

The main elements of proving Proposition B.5 are

(1) Replacing the integrals in $\mathbb{R}^D$ by a counterpart on $\mathcal{M}$, using the exponential away-manifold decay of the densities p and q.

(2) The uniform approximation of f^* by f_{con} on $\mathcal{M}$.

The second argument is also used to prove the "on-manifold" case in Section 4.3.

To proceed, we reproduce the needed result in Shaham et al. (2018), including the construction of atlas, the δ-wide neighborhood around manifold, and other notations, for completeness.

B.2.1 Result and set-up from Shaham et al. (2018)

We first establish some notation for the manifold and atlas cover. Recall that $\mathcal{M}$ be a smooth, compact manifold embedded in $\Omega \subset \mathbb{R}^D$. We cover $\mathcal{M}$ with an atlas $\{(U_i,\phi_i)\}_{i=1}^K$, where $U_i = B(x_i,\delta)\cap\mathcal{M}$ is an open set on $\mathcal{M}$ and $\phi_i : U_i \to \mathbb{R}^d$ is the map that takes U_i to the local tangent

space around $x_i \in U_i$. We also define the map $\psi_i : \phi_i(U_i) \to U_i$, which is the inverse of ϕ_i due to the one-to-one correspondence between U_i and $\phi_i(U_i)$.

We can choose δ small enough such that for any $x, x' \in U_i$, there exist positive α_i and β_i s.t.

$$\alpha_i \|\phi_i(x) - \phi_i(x')\|_2 \leq d_{\mathcal{M}}(x, x') \leq \beta_i \|\phi_i(x) - \phi_i(x')\|_2, \tag{A.5}$$

and for all i, $\alpha_i \geq \alpha_{\mathcal{M}}$, $\beta_i \leq \beta_{\mathcal{M}}$, and $\alpha_{\mathcal{M}}$, $\beta_{\mathcal{M}}$ are absolute constants. In particular, if the manifold is locally near Euclidean, then α_i, β_i are close to 1. For each neighborhood, this is possible for some $\delta_i > 0$ due to manifold smoothness, and the constants in that neighborhood will depend on the local curvature of the manifold. There exist global $\delta, \alpha_{\mathcal{M}}, \beta_{\mathcal{M}}$ due to compactness of the manifold.

Using the covering atlas, there exists a partition of unity $\{\eta_i\}_{i=1}^K$ such that supp$(\eta_i) \subset U_i$, $\eta_i \in C^\infty(\mathcal{M})$, and $\sum_{i=1}^K \eta_i(x) = 1$ for all $x \in \mathcal{M}$. The following theorem has been established under this setting:

Theorem B.6 (Shaham et al. (2018))**.** *Notations and assumptions as above, let $h \in C^2(\mathcal{M})$ and have a bounded Hessian. Then there exists a four layer feed network h_N with rectified linear unit activations, DK nodes in the first layer, $8dN$ nodes in the second layer, and $2N$ nodes in the third layer, such that*

$$\|h - h_N\|_{L^\infty(\mathcal{M})} \leq \frac{C_h}{N^{2/d}},$$

where C_h depends on $\|h\|_2$, $\|\nabla^2 h\|_2$ and the manifold and atlas. The total number of trainable parameters in the network is $O(N)$.

.

The proof of Theorem B.6 also constructs for each U_i a rectangle neighborhood N_i in $\mathbb{R}^D$ which is $\phi(U_i) \times (-\delta, \delta)^{D-d}$, thus $\phi(N_i) = \phi(U_i)$. Then the partition of unity function η_i is extended to N_i, given by $\tilde{\eta}_i(x) = \eta_i(\psi_i \circ \phi_i(x))$, for any $x \in N_i$. The union $N_\delta := \cup_{i=1}^K N_i$ forms a neighborhood of $\mathcal{M}$ in Ω. For any $p \in \mathcal{P}_\sigma$, we consider σ sufficiently small such that $\int_{\Omega \setminus N_\delta} p$ is exponentially small and negligible - when c_1, c_2 in (A.4) are 1, then $\sigma < \frac{1}{10}\delta$ suffices, and generally, we need $c_4(\mathcal{M})\sigma < 1$ where $c_4(\mathcal{M})$ is a constant depending on manifold and atlas (the δ) and c_1, c_2. By this truncation argument, in the following analysis we assume that p and q are supported on N_δ.

B.2.2 Technical lemmas

This implies that the extended partition of unity function $\tilde{\eta}_i$ is Lip in $\mathbb{R}^D$, that is

Lemma B.7. *For $i = 1, \cdots, K$, Lip$(\tilde{\eta}_i) \leq L_{\eta,\mathcal{M}}$ which is an absolute constant.*

Proof of Lemma B.7. For a fixed i, Since η_i is smooth on $\mathcal{M}$ and compactly supported on U_i, we assume that

$$|\eta_i(y_1) - \eta_i(y_2)| \leq c d_{\mathcal{M}}(y_1, y_2), \quad \forall y_1, y_2 \in U_i.$$

Now for $x_1, x_2 \in N_i$, let $y_1 = \psi_i \circ \phi_i(x_1)$, $y_2 = \psi_i \circ \phi_i(x_2)$, thus

$$\begin{aligned} |\tilde{\eta}_i(x_1) - \tilde{\eta}_i(x_2)| &= |\eta_i(y_1) - \eta_i(y_2)| \leq c d_{\mathcal{M}}(y_1, y_2) \\ &\leq c\beta_i \|\phi_i(y_1) - \phi_i(y_2)\|_2 \quad \text{(by (A.5))} \\ &= c\beta_i \|\phi_i(x_1) - \phi_i(x_2)\|_2 \leq c\beta_i \|x_1 - x_2\|_2, \end{aligned}$$

this proves that Lip$(\tilde{\eta}_i) \leq c\beta_i$, where c is the Lip(η_i) w.r.t. manifold geometry. Taking maximum over i gives $L_{\eta,\mathcal{M}}$ which is absolute content determined by the atlas and partition of unity construction. □

Lemma B.8. *There is c_3 an absolute constant s.t. for any $p \in \mathcal{P}_\sigma$, $\sigma > 0$,*

$$\int_{\mathbb{R}^D} d(x, \mathcal{M}) p(x) dx < c_3 \sigma.$$

Proof of Lemma B.8. Let $X \sim p$, then $d(X, \mathcal{M})$ is a non-negative random variable, and

$$\begin{aligned} \int_{\mathbb{R}^D} d(x, \mathcal{M}) p(x) dx = \mathbb{E} d(X, \mathcal{M}) &= \int_0^\infty \Pr[d(X, \mathcal{M}) > t] dt \\ &\leq \int_0^\infty c_1 e^{-c_2 \frac{t}{\sigma}} dt = \sigma \frac{c_1}{c_2}, \end{aligned}$$

which proves the claim with $c_3 = c_1/c_2$. □

This immediately gives the following lemma

Lemma B.9. *For any $\xi : \Omega \to \mathbb{R}$ which is Lip continuous, and $\xi|_{\mathcal{M}} = 0$, then for any $p \in \mathcal{P}_\sigma$,*

$$\int_\Omega |\xi(x)|p(x)dx < \textit{Lip}(\xi)c_3\sigma.$$

Proof of Lemma B.9. By compactness and smoothness of $\mathcal{M}$, for any $x \in \Omega$, there exists $\psi(x) \in \mathcal{M}$ s.t. $\|\psi(x) - x\|_2 = d(x, \mathcal{M})$. thus

$$|\xi(x)| = |\xi(x) - \xi(\psi(x))| \leq \text{Lip}(\xi)\|x - \psi(x)\|_2 = \text{Lip}(\xi)d(x, \mathcal{M}).$$

Then

$$\int_\Omega |\xi(x)|p(x)dx \leq \int_\Omega \text{Lip}(\xi)d(x, \mathcal{M})p(x)dx < \text{Lip}(\xi) \cdot c_3\sigma,$$

where the last $<$ is by Lemma B.8. □

The following Lemma fulfills element (1) in the proof.

Lemma B.10. *Let $g : \Omega \to \mathbb{R}$ has finte Lip(g) and $\|g\|_{L^\infty(\Omega)}$, $p \in \mathcal{P}_\sigma$, and define*

$$\tilde{p}(x) = \sum_{i=1}^K \eta_i(x)\tilde{p}_i(x),$$

where $\tilde{p}_i$ is an atlas dependent projection of the density p to U_i, the explicit formula to be given below, then

$$\left|\int_\Omega g(x)p(x)dx - \int_{\mathcal{M}} g(x)\tilde{p}(x)d_{\mathcal{M}}(x)\right| \leq K(\|g\|_{L^\infty(\Omega)}L_{\eta,\mathcal{M}} + \textit{Lip}(g)(1 + \beta_{\mathcal{M}}))c_3\sigma.$$

Proof of Lemma B.10. Let $H_i := \phi_i(U_i) = \phi_i(N_i)$ for each $i = 1, \cdots, K$,

$$\begin{aligned}
\int_\Omega p(x)g(x)dx &\approx \int_\Omega p(x)g(x)\sum_{i=1}^K \tilde{\eta}_i(x) \quad \text{(error 1)} \\
&= \sum_{i=1}^K \int_{N_i} g(x)\tilde{\eta}_i(x)p(x)dx \quad (p \text{ supported on } \cup_i N_i, \text{ c.f. comment after Theorem B.6}) \\
&\approx \sum_i \int_{N_i} g(\psi_i \circ \phi_i(x))\tilde{\eta}_i(x)p(x)dx \quad \text{(error 2)} \\
&= \sum_i \int_{H_i} (g \cdot \eta_i)(\psi_i(u)) \int_{[-\delta,\delta]^{D-d}} p(u, v)dudv \\
&=: \sum_i \int_{U_i} g(z)\eta_i(z)\tilde{p}_i(z)d_{\mathcal{M}}(z) \quad \text{(definition of } \tilde{p}_i) \\
&= \int_{\mathcal{M}} g(z)\left(\sum_i \eta_i(z)\tilde{p}_i(z)\right)d_{\mathcal{M}}(z) = \int_{\mathcal{M}} g(z)\tilde{p}(z)d_{\mathcal{M}}(z),
\end{aligned}$$

where $d_{\mathcal{M}}(z)$ stands for the Reimannian volume measure on $\mathcal{M}$. Thus it suffices to bound the error in (error 1) and (error 2) and show that the sum $\leq$ the right hand side of the claim in the Lemma.

Bound of (error 1):

$$\left|\int_\Omega p(x)g(x)dx - \int_\Omega p(x)g(x)\sum_{i=1}^K \tilde{\eta}_i(x)\right| \leq \|g\|_{L^\infty(\Omega)} \int_\Omega p(x)\left|1 - \sum_{i=1}^K \tilde{\eta}_i(x)\right|dx,$$

and the Lip constant of the function $\xi := (1 - \sum_{i=1}^K \tilde{\eta}_i)$ is upper bounded by $\sum_{i=1}^K \text{Lip}(\tilde{\eta}_i) \leq KL_{\eta,\mathcal{M}}$ by Lemma B.7. Also ξ vanishes on $\mathcal{M}$. Applying Lemma B.9 gives that

$$\text{(error 1)} \leq \|g\|_{L^\infty(\Omega)}KL_{\eta,\mathcal{M}}c_3\sigma.$$

Bound of (error 2):

$$\left|\sum_{i=1}^{K}\int_{N_i}(g(x)-g(\psi_i\circ\phi_i(x)))\tilde{\eta}_i(x)p(x)dx\right|\leq\sum_{i=1}^{K}\int_{N_i}|g(x)-g(\psi_i\circ\phi_i(x))|p(x)dx, \quad \text{(A.6)}$$

using $\tilde{\eta}_i(x)\leq 1$. For each i, consider $\xi(x):=g(x)-g(\psi_i\circ\phi_i(x))$, one can verify that

$$\text{Lip}(\xi)\leq\text{Lip}(g)+\text{Lip}(g)\beta_i,$$

by (A.5), thus each term in the summation of the r.h.s of (A.6) $\leq \text{Lip}(g)(1+\beta_{\mathcal{M}})c_3\sigma$. This proves that

$$\text{(error 2)}\leq K\text{Lip}(g)(1+\beta_{\mathcal{M}})c_3\sigma.$$

Combining the two bounds of (error 1) and (error 2) proves the claim. □

B.2.3 Proof of Proposition B.5

We are now ready to prove the main result in this section.

Proof of Proposition B.5. Since $\text{JSD}(p,q)=L[f^*]$, it suffices to control $|L[f_{\text{con}}]-L[f^*]|$ as stated in the claim.

We first consider $f=f^*$ which has a finite $\text{Lip}(f^*)$, and we have that

$$L[f]=\frac{1}{2}\left(\int_\Omega p\log\frac{2e^f}{1+e^f}+\int_\Omega q\log\frac{2}{1+e^f}\right):=\frac{1}{2}(L_p+L_q).$$

Define $g:=\log\frac{2e^f}{1+e^f}$, then

$$L_p=\int_\Omega pg,$$

and by that $\log\frac{2e^\xi}{1+e^\xi}$ as a function of $\xi\in\mathbb{R}$ is Lip 1, we have that

$$\text{Lip}(g)\leq\text{Lip}(f^*).$$

To bound $|g(x)|$, note that there is at least one point $x_0\in\Omega$ s.t. $f^*(x_0)=0$, thus $g(x_0)=0$. Then $\forall x\in\Omega$,

$$|g(x)|=|g(x)-g(x_0)|\leq\text{Lip}(g)\|x-x_0\|\leq 2\text{Lip}(g).$$

Applying Lemma B.10, we have that

$$\begin{aligned}&L_p=\int_{\mathcal{M}}\tilde{p}g+r_1=\int_{\mathcal{M}}\tilde{p}\log\frac{2e^f}{1+e^f}+r_1,\\&|r_1|\leq\text{Lip}(g)K(2L_{\eta,\mathcal{M}}+1+\beta_{\mathcal{M}})c_3\sigma\leq c_{\mathcal{M}}\text{Lip}(f^*)\sigma,\end{aligned}\quad \text{(A.7)}$$

where

$$c_{\mathcal{M}}:=K(2L_{\eta,\mathcal{M}}+1+\beta_{\mathcal{M}})c_3 \quad \text{(A.8)}$$

is an absolute constant only depending on manifold and atlas.

Similarly, we can show that

$$L_q=\int_{\mathcal{M}}\tilde{q}\log\frac{2}{1+e^f}+r_2,\quad |r_2|\leq c_{\mathcal{M}}\text{Lip}(f^*)\sigma.$$

This gives that

$$L[f^*]=\frac{1}{2}\left(\int_{\mathcal{M}}\tilde{p}\log\frac{2e^{f^*}}{1+e^{f^*}}+\int_{\mathcal{M}}\tilde{q}\log\frac{2}{1+e^{f^*}}\right)+r_{1,2},\quad |r_{1,2}|\leq c_{\mathcal{M}}\text{Lip}(f^*)\sigma. \quad \text{(A.9)}$$

We then consider $f=f_{\text{con}}$, where f_{con} is constructed by Theorem B.6 to uniformly approximate f^* on $\mathcal{M}$ up to ε_1. Following Proposition B.4, $\sup_{x\in\Omega}|f_{\text{con}}(x)|\leq B_\Theta^{(0)}$, and $\text{Lip}(f_{\text{con}})\leq B_\Theta^{(1)}$. Similar as before, $g:=\log\frac{2e^f}{1+e^f}$ or $\log\frac{2}{1+e^f}$ both have

$$\text{Lip}(g)\leq\text{Lip}(f)\leq B_\Theta^{(1)}.$$

Also observe the relation that $|F(\xi)| \leq |\xi|$ for $\xi \in \mathbb{R}$, where $F(\xi) = \log \frac{2e^{\xi}}{1+e^{\xi}}$ or $\log \frac{2}{1+e^{\xi}}$, which gives that $|g(x)| \leq |f(x)|$, for all $x \in \Omega$. This gives that $\|g\|_{L^{\infty}(\Omega)} \leq B_{\Theta}^{(0)}$. It is generally valid that $f_{\text{con}}(x_0) = 0$ for some $x_0 \in \mathcal{M} \subset \Omega$, and then $g(x_0) = 0$, thus we also have that $|g(x)| \leq 2\text{Lip}(g)$. Thus

$$\|g\|_{L^{\infty}(\Omega)} \leq \min\{B_{\Theta}^{(0)}, 2B_{\Theta}^{(1)}\} := B_{\Theta}^{'}. \tag{A.10}$$

Putting together, we then have

$$\begin{aligned} L[f_{\text{con}}] &= \frac{1}{2}\left(\int_{\mathcal{M}} \tilde{p} \log \frac{2e^{f_{\text{con}}}}{1+e^{f_{\text{con}}}} + \int_{\mathcal{M}} \tilde{q} \log \frac{2}{1+e^{f_{\text{con}}}}\right) + r_3, \\ |r_3| &\leq K(B_{\Theta}^{'} L_{\eta,\mathcal{M}} + B_{\Theta}^{(1)}(1+\beta_{\mathcal{M}}))c_3\sigma. \end{aligned} \tag{A.11}$$

Comparing (A.9), (A.11) gives that

$$|L[f^*] - L[f_{\text{con}}]| \leq \frac{1}{2}\varepsilon_1 \int_{\mathcal{M}} (\tilde{p} + \tilde{q}) + |r_{1,2}| + |r_3|,$$

where we used that $|f^*(x) - f_{\text{con}}(x)| \leq \varepsilon_1, \forall x \in \mathcal{M}$. Observe that

$$\int_{\mathcal{M}} \tilde{p} \leq \int_{\Omega} p + KL_{\eta,\mathcal{M}}c_3\sigma = 1 + KL_{\eta,\mathcal{M}}c_3\sigma,$$

by applying Lemma B.10 with $g(x) = 1$, and same for $\int_{\mathcal{M}} \tilde{q}$. This proves that

$$\begin{aligned} |L[f^*] - L[f_{\text{con}}]| &\leq (1 + KL_{\eta,\mathcal{M}}c_3\sigma)\varepsilon_1 + \text{Lip}(f^*)K(2L_{\eta,\mathcal{M}} + 1 + \beta_{\mathcal{M}})c_3\sigma \\ &\quad + K(B_{\Theta}^{'} L_{\eta,\mathcal{M}} + B_{\Theta}^{(1)}(1+\beta_{\mathcal{M}}))c_3\sigma \\ &< 10\varepsilon_1 + c(f^*, \Theta, \mathcal{M})\sigma, \end{aligned} \tag{A.12}$$

where we assume that σ is small enough to make $KL_{\eta,\mathcal{M}}c_3\sigma < 9$, and $c(f^*, \Theta, \mathcal{M})$ is a positive constant the formula of which is given in the previous term. At last, the comment after Theorem B.6) needs that $c_4(\mathcal{M})\sigma < 1$, and we let $\tilde{c}_4$ be the maximum of $9c_4(\mathcal{M})$ and $KL_{\eta,\mathcal{M}}c_3$. □