Maximum Mean Discrepancy Test is Aware of Adversarial Attacks

Ruize Gao [*1 2] **Feng Liu** [*3] **Jingfeng Zhang** [*4] **Bo Han** [1] **Tongliang Liu** [5] **Gang Niu** [4] **Masashi Sugiyama** [4 6]

Abstract

The *maximum mean discrepancy* (MMD) test could in principle detect any distributional discrepancy between two datasets. However, it has been shown that the MMD test is unaware of *adversarial attacks*—the MMD test failed to detect the discrepancy between *natural* and *adversarial data*. Given this phenomenon, we raise a question: are natural and adversarial data really from different distributions? The answer is affirmative—the previous use of the MMD test on the purpose missed three key factors, and accordingly, we propose three components. Firstly, *Gaussian kernel* has limited *representation power*, and we replace it with an effective *deep kernel*. Secondly, *test power* of the MMD test was neglected, and we maximize it following *asymptotic statistics*. Finally, adversarial data may be *non-independent*, and we overcome this issue with the *wild bootstrap*. By taking care of the three factors, we verify that *the MMD test is aware of adversarial attacks*, which lights up a novel road for adversarial data detection based on two-sample tests.

1. Introduction

The *maximum mean discrepancy* (MMD) aims to measure the closeness between two distributions $\mathbb{P}$ and $\mathbb{Q}$:

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

where $\mathcal{F}$ is a set containing all continuous functions (Gretton et al., 2012a). To obtain an analytic solution regarding the sup in Eq. (1), Gretton et al. (2012a) restricted $\mathcal{F}$ to be a unit ball in the *reproducing kernel Hilbert space* (RKHS) and obtain the kernel-based MMD defined in the following.

$$\begin{aligned}\mathrm{MMD}(\mathbb{P},\mathbb{Q};\mathcal{H}_k) := &\sup_{f\in\mathcal{H},\|f\|_{\mathcal{H}_k}\le 1} |\mathbb{E}[f(X)]-\mathbb{E}[f(Y)]| \\ =&\|\mu_{\mathbb{P}}-\mu_{\mathbb{Q}}\|_{\mathcal{H}_k},\end{aligned} \tag{2}$$

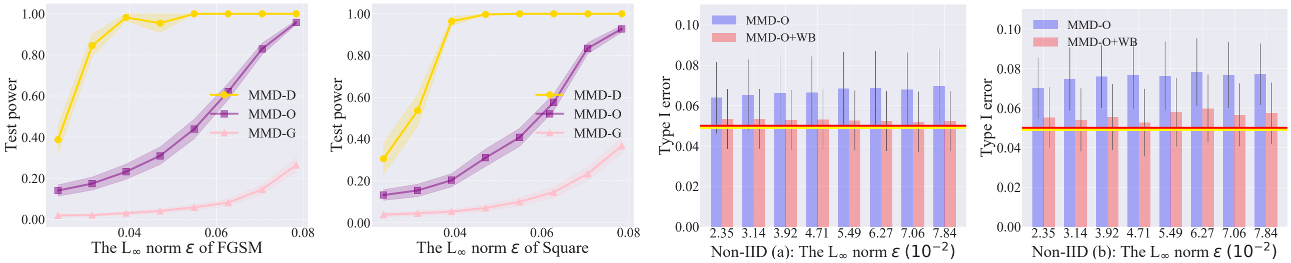
where k is a bounded kernel regarding a RKHS $\mathcal{H}_k$ (i.e., $|k(\cdot,\cdot)| < +\infty$), and $X \sim \mathbb{P}$, $Y \sim \mathbb{Q}$ are two random variables, and $\mu_{\mathbb{P}} := \mathbb{E}[k(\cdot,X)]$ and $\mu_{\mathbb{Q}} := \mathbb{E}[k(\cdot,Y)]$ are kernel mean embeddings of $\mathbb{P}$ and $\mathbb{Q}$, respectively (Gretton et al., 2005; 2012a; Jitkrittum et al., 2016; 2017; Sutherland et al., 2017; Liu et al., 2020b). According to Eq. (1), it is clear that MMD equals zero *if and only if* $\mathbb{P}=\mathbb{Q}$ (Gretton et al., 2008). As for the MMD defined in Eq. (2), Gretton et al. (2012a) also prove this property. Namely, we could *in principle* use the MMD to show whether two distributions are the same, which drives researchers to develop the MMD-based two-sample test (Gretton et al., 2012a).

In the MMD test, we are given two samples observed from $\mathbb{P}$ and $\mathbb{Q}$ and aim to check whether two samples come from the same distribution. Specifically, we first *estimate* MMD value from two samples, and then compute the p-value corresponding to the estimated MMD value (Sutherland et al., 2017). If the p-value is above a given threshold α, then two samples are from the same distribution. In the last decade, MMD test has been used to detect the distributional discrepancy within several real-world datasets, including high-energy physics data (Chwialkowski et al., 2015), amplitude modulated signals (Gretton et al., 2012b), and challenging image datasets, e.g., the *MNIST* and the *CIFAR-10* (Sutherland et al., 2017; Liu et al., 2020b).

However, it has been empirically shown that the MMD test, as one of the most powerful two-sample tests, is unaware of *adversarial attacks* (Carlini & Wagner, 2017a). Specifically, Carlini & Wagner (2017a) input adversarial and natural data into the MMD test, then the MMD test outputs a p-value that is greater than the given threshold α with a high probability. Namely, the MMD test agrees that adversarial and natural data are from the same distribution. Given the success of MMD test in many fields (Liu et al., 2020b), this phenomenon seems a *paradox* regarding the homogeneity between nature and adversarial data.

[*]Equal contribution [1]Department of Computer Science, Hong Kong Baptist University [2]Department of Computer Science and Engineering, The Chinese University of Hong Kong [3]DeSI Lab, AAII, University of Technology Sydney [4]RIKEN-AIP [5]TML Lab, University of Sydney [6]Graduate School of Frontier Sciences, University of Tokyo. Correspondence to: Bo Han <bhanml@comp.hkbu.edu.hk>, Tongliang Liu <tongliang.liu@sydney.edu.au>, Gang Niu <gang.niu@riken.jp>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).



(a) Test power (b) Test power (c) Type I error (d) Type I error

Figure 1. Consequences of missing the three key factors when using the MMD on adversarial data detection. The subfigure (a) and (b) illustrate the test power of *the MMD test with deep kernel* (MMD-D test (Liu et al., 2020b)), *the MMD test with optimized Gaussian kernel* (MMD-O test (Sutherland et al., 2017)) and *the MMD test with Gaussian kernel* (MMD-G test), respectively. Adversarial data is generated by a white-box attack *fast gradient sign method* (FGSM) (Goodfellow et al., 2015) and a black-box attack *Square attack* (Square) (Andriushchenko et al., 2020) with different L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$ (following (Madry et al., 2018; Zhang et al., 2020a)). Clearly, MMD-D and MMD-O tests perform much better than MMD-G test (previously used by (Grosse et al., 2017) and (Carlini & Wagner, 2017a)). The failure of MMD-G test takes root in Factors 1 and 2 in Section 1. The (c) and (d) show type I error within two typical non-IID adversarial data (see detailed generation in Section 5), where type I error of MMD-O test is abnormal (higher than the red line that $\alpha = 0.05$, while the type I error within natural data is the yellow line). The main reason is the Factor 3 in Section 1. If we apply the *wild bootstrap* (WB) process to MMD-O test, it brings type I error to normality (MMD-O+WB).

In this paper, we raise a question regarding this paradox: *are natural data and adversarial data really from different distributions?* The answer is affirmative, and we find the previous use of MMD missed *three* factors. As a result, previous MMD-based adversarial data detection methods not only have a low detection rate when detecting attacks (due to the first two factors), but also are invalid detection methods (due to the third factor).

Factor 1. The Gaussian kernel (used by previous MMD-based adversarial data detection methods) has *limited* representation power and cannot measure the similarity between two multidimensional samples (e.g., images) well (Wenliang et al., 2019). Although $\mathrm{MMD}(\mathbb{P}, \mathbb{Q})$ is a perfect statistic to see if $\mathbb{P}$ equals $\mathbb{Q}$, test power (i.e., the detection rate when detecting adversarial attacks) of its empirical estimation (Eq. (3)) depends on the form of used kernels (Sutherland et al., 2017; Liu et al., 2020b). Since a Gaussian kernel only looks at data uniformly rather than focuses on areas where two distributions are different, it requires many observations to distinguish the two distributions (Liu et al., 2020b). As a result, the test power of *the MMD test with a Gaussian kernel* (MMD-G test used by Grosse et al. (2017) and Carlini & Wagner (2017a)) is *limited*, especially when facing complex data (Sutherland et al., 2017; Liu et al., 2020b).

We replace the Gaussian kernel with a simple and effective semantic-aware deep kernel to take care of the first factor. We call this semantic-aware deep kernel based MMD as *semantic-aware MMD* (SAMMD). The SAMMD is motivated by the recent advances in nonparametric two-sample tests, i.e., *the MMD test with deep kernel* (MMD-D). In MMD-D, the kernel is parameterized by deep neural nets (Liu et al., 2020b) and measures the distributional discrepancy between two sets of images using raw features (i.e., pixels in images). Compared to the deep kernel used in MMD-D, semantic-aware deep kernel uses *semantic features* extracted by a well-trained classifier on natural data. Figure 2 (see Section 6) shows that natural and adversarial data are quite different in the view of semantic features, showing that semantic features can help distinguish between natural and adversarial data, taking care of the first factor.

Factor 2. Previous MMD-based adversarial data detection methods overlook the optimization of parameters of the used kernel. In MMD-G test, its test power is related to the choice of the bandwidth of the Gaussian kernel (Sutherland et al., 2017). Once we overlook the optimization of the kernel bandwidth, the test power of MMD-G test will *drop significantly* (Gretton et al., 2012b; Sutherland et al., 2017). Furthermore, recent studies have shown that Gaussian kernel with an optimized bandwidth still has limited representation power for complex distributions (e.g., multi-modal distributions used in (Wenliang et al., 2019; Liu et al., 2020b)). Namely, it is important to take care of Factor 1 and Factor 2 simultaneously, which is verified in Figures 1a-1b.

To take care of the second factor, we analyze the asymptotics of the SAMMD when detecting adversarial attacks. According to the asymptotics of SAMMD, we can compute the approximate test power of SAMMD using two datasets and then optimize the parameters of the deep kernel by maximizing the approximate test power.

Factor 3. The adversarial data are probably not *independent and identically distributed* (IID) due to their unknown generation process, which breaks a basic assumption of the MMD tests used by (Grosse et al., 2017; Carlini & Wagner, 2017a). Once there exists dependence within the observations, the type I error of ordinary MMD tests will surpass the given threshold α. Note that, type I error is the probability of rejecting the null hypothesis ($\mathbb{P} = \mathbb{Q}$) when the null hypothesis is true. If the type I error of a test is much higher than α, this test will always reject the null hypothesis. Namely, for two datasets that come from the same distribution, the test will always show that they are different, which means that the test is *meaningless* (Chwialkowski et al., 2014).

To take care of the third factor, the wild bootstrap is used to resample the value of SAMMD (with the optimized kernel), which ensures that we can get correct p-values in non-IID/IID scenarios (Figures 1c-1d). Here, we show two scenarios where the dependence within adversarial data exists: 1) the adversary attacks the data used for training the target model, where the target model depends on the attacked data. Thus, generated adversarial data are highly dependent (the Non-IID (a) in Figure 1c); and 2) the adversary attacks one instance many times to generate many adversarial instances (the Non-IID (b) in Figure 1d).

The above study is not of purely theoretical interest; it has also practical consequences. The considered detection problem is also known as *statistical adversarial data detection* (SADD). In SADD, we care about how to find out a dataset that only contains natural data. That will bring benefits to users who are *only* interested in a model that has high accuracy on the *natural data*. For example, as an artificial-intelligence service provider, we need to acquire a client by modeling his/her task well, such as modeling the risk level of a factory. In this task, the client only cares about the accuracy on the natural data. Thus, we need to use MMD test to ensure that our training data only contain natural data. In Appendix B, we have demonstrated SADD in detail.

2. Preliminary

This section presents four concepts used in this paper.

Two-sample test. Let $\mathcal{X} \subset \mathbb{R}^d$ and $\mathbb{P}$, $\mathbb{Q}$ be Borel probability measures on $\mathcal{X}$. Given IID samples $S_X = \{\boldsymbol{x}_i\}_{i=1}^n \sim \mathbb{P}^n$ and $S_Y = \{\boldsymbol{y}_j\}_{j=1}^m \sim \mathbb{Q}^m$, in the two-sample test problem, we aim to determine if S_X and S_Y come from the same distribution, i.e., if $\mathbb{P} = \mathbb{Q}$.

Estimation of MMD. We can estimate MMD (Eq. (2)) using the U-statistic estimator, which is unbiased for MMD^2 and has nearly minimal variance among unbiased estimators (Gretton et al., 2012a):

$$\widehat{\mathrm{MMD}}_u^2(S_X, S_Y; k) = \frac{1}{n(n-1)} \sum_{i \neq j} H_{ij}, \tag{3}$$

$$H_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j) + k(\boldsymbol{y}_i, \boldsymbol{y}_j) - k(\boldsymbol{x}_i, \boldsymbol{y}_j) - k(\boldsymbol{y}_i, \boldsymbol{x}_j),$$

where $\boldsymbol{x}_i, \boldsymbol{x}_j \in S_X$ and $\boldsymbol{y}_i, \boldsymbol{y}_j \in S_Y$.

Adversarial data generation. Let $(\mathcal{X}, d_\infty)$ be the input feature space $\mathcal{X}$ with the infinity distance metric $d_{\inf}(\boldsymbol{x}, \boldsymbol{x}') = \|\boldsymbol{x} - \boldsymbol{x}'\|_\infty$, and

$$\mathcal{B}_\epsilon[\boldsymbol{x}] = \{\boldsymbol{x}' \in \mathcal{X} \mid d_{\inf}(\boldsymbol{x}, \boldsymbol{x}') \leq \epsilon\} \tag{4}$$

be the closed ball of radius $\epsilon > 0$ centered at $\boldsymbol{x}$ in $\mathcal{X}$. Let $D = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^n$ be a dataset, where $\boldsymbol{x}_i \in \mathcal{X}$, $l_i \in \mathcal{C}$ is ground-truth label of $\boldsymbol{x}_i$, and $\mathcal{C} = \{1, \ldots, C\}$ is a label set. Then, adversarial data regarding $\boldsymbol{x}_i$ is

$$\mathcal{G}_{\ell, \hat{f}}(\boldsymbol{x}_i) = \arg\max_{\tilde{\boldsymbol{x}} \in \mathcal{B}_\epsilon[\boldsymbol{x}_i]} \ell(\hat{f}(\tilde{\boldsymbol{x}}), l_i), \tag{5}$$

where $\tilde{\boldsymbol{x}}$ is a sample within the ϵ-ball centered at $\boldsymbol{x}$, $\hat{f}(\cdot) : \mathcal{X} \to \mathcal{C}$ is a well-trained classifier on D, and $\ell : \mathcal{C} \times \mathcal{C} \to \mathbb{R}_{\geq 0}$ is a loss function.

There are many methods to solve Eq. (5) and generate adversarial data, e.g., white-box attacks including *fast gradient sign method* (FGSM) (Goodfellow et al., 2015), *basic iterative methods* (BIM) (Kurakin et al., 2017), *project gradient descent* (PGD) (Madry et al., 2018), *AutoAttack* (AA) (Croce & Hein, 2020), *Carlini and Wagner attack* (CW) (Carlini & Wagner, 2017b) and a score-based black-box attack: *Square attack* (Square Andriushchenko et al. (2020)).

Wild bootstrap process. The wild bootstrap process has been proposed to resample observations from a stochastic process $\{Y_i\}_{i \in \mathbb{Z}}$ (Shao, 2010), where $\mathbb{E}(Y_i) = 0$ for each $i \in \mathbb{Z}$. Through multiplying the given observations with random numbers from the wild bootstrap process, we can obtain new samples that can be regarded as resampled observations from $\{Y_i\}_{i \in \mathbb{Z}}$ (Leucht & Neumann, 2013; Chwialkowski et al., 2014). After resampling observations many times, we can use such resampled observations to estimate the distribution of statistics regarding the random process $\{Y_i\}_{i \in \mathbb{Z}}$, such as the null distribution of MMD over two time series (Chwialkowski et al., 2014). Following (Chwialkowski et al., 2014) and (Leucht & Neumann, 2013), this paper uses the following wild bootstrap process:

$$W_t = e^{-1/l} W_{t-1} + \sqrt{1 - e^{-2/l}} \epsilon_t, \tag{6}$$

where $W_0, \epsilon_0, \ldots, \epsilon_t$ are independent standard normal random variables.

3. Problem Setting

Following Grosse et al. (2017), we aim to address the following problem (i.e., SADD mentioned in Section 1).

Problem 1 (SADD). *Let $\mathcal{X}$ be a subset of $\mathbb{R}^d$ and $\mathbb{P}$ be a Borel probability measure on $\mathcal{X}$, and $S_X = \{\boldsymbol{x}_i\}_{i=1}^n \sim \mathbb{P}^n$ be IID observations from $\mathbb{P}$, and $f(\cdot) : \mathbb{R}^d \to \mathcal{C}$ be the ground-truth labeling function on observations from $\mathbb{P}$, where $\mathcal{C} = \{1, \ldots, C\}$ is a label set. Assume that attackers can obtain a well-trained classifier $\hat{f}$ on S_X and*

Table 1. Average values of dependence scores (HSIC) within natural data ($\epsilon = 0$) and non-IID adversarial data (the L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$). The adversarial data of the Non-IID (a) are generated by FGSM on the training set of *CIFAR-10*. The Non-IID (b) consists of the adversarial data generated by Square on CIFAR-10's testing set (for each natural image, Square generates four different adversarial images). We can see that the dependence within non-IID adversarial data is stronger than that within IID natural data.

Perturbation bound ϵ	0.0000	0.0235	0.0314	0.0392	0.0471	0.0549	0.0627	0.0706	0.0784
Non-IID (a) (10e-5)	2.1948	2.2214	2.2409	2.2650	2.3067	2.3320	2.3727	2.4234	2.4805
Non-IID (b) (10e-5)	2.1948	2.2146	2.2346	2.2614	2.2952	2.3359	2.3835	2.4381	2.4998

IID observations S'_X from $\mathbb{P}$, we aim to determine if the upcoming data $S_Y = \{\boldsymbol{y}_i\}_{i=1}^m$ come from the distribution $\mathbb{P}$, where S_X and S'_X are independent, and we do not have any prior knowledge regarding the attacking methods. Note that, in SADD, S_Y may be IID data from $\mathbb{P}$ or non-IID data generated by attackers.

In Problem 1, if S_Y are IID observations from $\mathbb{P}$, given a threshold α, we aim to accept the null hypothesis H_0 (i.e., S_X and S_Y are from the same distribution) with the probability $1 - \alpha$. If S_Y contains adversarial data (i.e., S_X and S_Y are from different distributions), we aim to reject the null hypothesis H_0 with a probability near to 1. Please note that, an invalid test method could be "rejecting all upcoming data", which can perform very well when S_X and S_Y being from different distributions but fail when S_Y being from $\mathbb{P}$.

4. Failure of Gaussian-kernel MMD Test for Adversarial Data Detection

We reimplement the experiment in (Carlini & Wagner, 2017a) and (Grosse et al., 2017) to test the performance of MMD-G test on *CIFAR-10* dataset. Adversarial data with different perturbation bound ϵ are generated by FGSM, BIM, PGD, AA, CW and Square. Figure 4 shows how test power changes as the ϵ value increases in each attacking method. Through our implementations, we draw the same conclusion with Carlini & Wagner (2017a). Namely, MMD-G test (the pink line) fails to detect adversarial data.

As demonstrated in Section 1, MMD-G test has the following issues: 1) the limited representation power of the Gaussian kernel (Wenliang et al., 2019; Liu et al., 2020b); and 2) the overlook of optimization of the kernel bandwidth (Liu et al., 2020b); and 3) the non-IID property of adversarial data (Shao, 2010; Leucht & Neumann, 2013; Chwialkowski et al., 2014). Since the third issue is crucial, we first analyze whether there exists dependence within adversarial data in the following section and then propose a novel test to address the above issues simultaneously (see Section 6).

5. Dependence in Adversarial Data

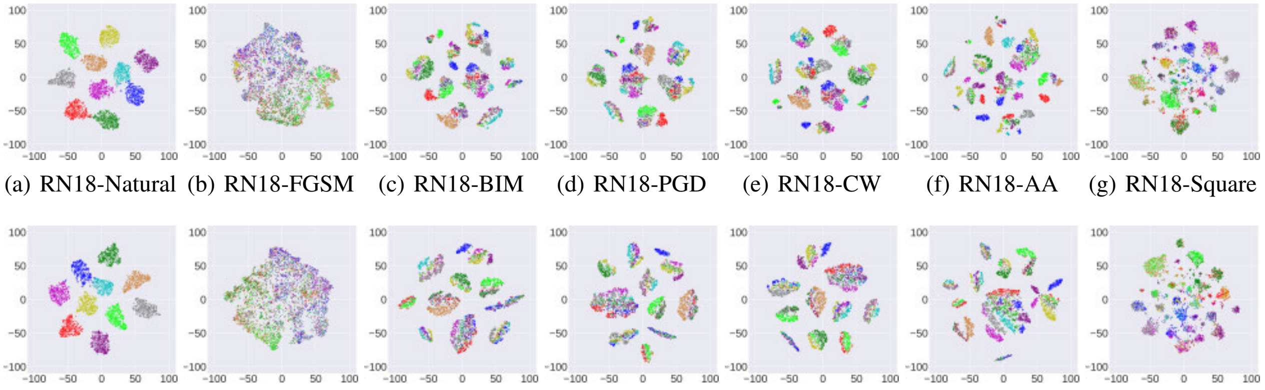
As discussed in Section 4, this section investigates whether there is dependence within adversarial data.

Dependence within adversarial data. In the real world, since we do not know the attacking strategies of attackers, the dependence within adversarial data probably exists. For example, if attackers use one natural image to generate many adversarial images, the adversarial data is obviously not independent (the Non-IID (b) in Table 1). To empirically show the dependence within adversarial data, we use HSIC (Gretton et al., 2005) as the statistic to represent the dependence score within adversarial data (Appendix C presents detailed procedures to compute the HSIC values between two datasets). The larger value of HSIC represents the stronger dependence.

We generated two typical non-IID adversarial datasets that the Non-IID (a) and the Non-IID (b). Given natural images from the *CIFAR-10* training set, we generated the Non-IID (a) using FGSM with the L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$. Given natural images from the *CIFAR-10* testing set, we used Square with the L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$ to generate the adversarial data four times and mixed them into the Non-IID (b). For each dataset, we randomly selected two disjoint subsets (containing 500 images) and compute the HSIC value over the two subsets. Repeating the above process 100 times, we obtained the average value of the 100 HSIC values in Table 1. Since the HSIC value of adversarial data is higher than that of natural data (i.e., $\epsilon = 0$), the dependence within adversarial data is stronger than that within IID natural data.

Dependence meets MMD tests. Grosse et al. (2017) and Carlini & Wagner (2017a) used the permutation based bootstrap (Odén et al., 1975) to implement MMD-G test (i.e., the ordinary MMD test). Specifically, they initialize a by $\mathrm{MMD}(S_X, S_Y)$. Then, they shuffle the elements of S_X and S_Y into two new sets G_X and G_Y, and let $b = \mathrm{MMD}(G_X, G_Y)$. Repeating the shuffling process K times, they can obtain a sequence $\{b_k\}_{k=1}^K$. If a is greater than the $1 - \alpha$ quantile of $\{b_k\}_{k=1}^K$, then the null hypothesis is rejected (Grosse et al., 2017; Carlini & Wagner, 2017a). Namely, the adversarial attacks are detected.

However, the permutation-based MMD-G test only works when facing IID data (Chwialkowski et al., 2014). Since adversarial data may not be IID, according to Chwialkowski et al. (2014), *the permutation based MMD-G* (previously



(a) RN18-Natural (b) RN18-FGSM (c) RN18-BIM (d) RN18-PGD (e) RN18-CW (f) RN18-AA (g) RN18-Square

(h) RN34-Natural (i) RN34-FGSM (j) RN34-BIM (k) RN34-PGD (l) RN34-CW (m) RN34-AA (n) RN34-Square

Figure 2. Visualization of outputs using t-SNE. This figure visualizes outputs of the second to last layers in ResNet-18 and ResNet-34. Different colors represent different semantic meanings (i.e., different classes in the testing set of the *CIFAR-10*). Apparently, semantic information contained in natural data is lost in adversarial data. This phenomenon can help us distinguish adversarial data and natural data.

used by Grosse et al. (2017) and Carlini & Wagner (2017a)) could be invalid to detect adversarial data.

6. A Semantic-aware MMD Test

To take care of three factors missed by previous studies (Grosse et al., 2017; Carlini & Wagner, 2017a), we design a simple and effective test motivated by the most important characteristic of adversarial data. Namely, semantic meaning of adversarial data (in the view of a well-trained classifier on natural data) is very different from that of natural data. Based on this characteristic, *semantic-aware MMD* (SAMMD) is proposed to measure the discrepancy between natural and adversarial data in this section.

Semantic features. As mentioned above, the semantic meaning of data plays an important role to distinguish between natural and adversarial data. Thus, we will first introduce how to represent the semantic meaning of each image, i.e., to construct *semantic features* of images, in this part. Since the success of deep learning mainly takes roots in its ability to extract features that can be used to classify images well, outputs of the layers of a well-trained deep neural network have already contained semantic meaning. Namely, we can construct semantic features of images using outputs of the layers of the well-trained network.

Figure 2 visualizes outputs of the second to last full connected layers of a well-trained ResNet-18 and ResNet-34 using t-SNE (Maaten & Hinton, 2008), showing that these outputs indeed contain clear semantic meanings (in the view of natural data). Thus, we use these outputs as semantic features in this paper. This figure also shows that natural data and adversarial data are quite different in the view of semantic features. In addition, we also show the MMD values between semantic features of natural and adversarial data in Figure 3. Results show that, in the second to last full connected layer of ResNet-18, outputs of natural and adversarial data have the largest distributional discrepancy. Thus, the semantic features we constructed can help us distinguish adversarial data and natural data well.

Semantic-aware MMD. Based on the semantic features, we consider the following semantic-aware deep kernel $k_\omega(\boldsymbol{x},\boldsymbol{y})$ to measure the similarity between two images:

$$k_\omega(\boldsymbol{x},\boldsymbol{y}) = \Big[(1-\epsilon_0)s_{\hat{f}}(\boldsymbol{x},\boldsymbol{y}) + \epsilon_0\Big]q(\boldsymbol{x},\boldsymbol{y}), \tag{7}$$

where $s_{\hat{f}}(\boldsymbol{x},\boldsymbol{y}) = \kappa(\phi_p(\boldsymbol{x}),\phi_p(\boldsymbol{y}))$ is a deep kernel function that measures the similarity between $\boldsymbol{x}$ and $\boldsymbol{y}$ using semantic features extracted by $\hat{f}$; we use ϕ_p, the second to the last fully connected layer in $\hat{f}$, to extract semantic features (according to Figure 3); the κ is the Gaussian kernel (with bandwidth σ_{ϕ_p}); $\epsilon_0 \in (0,1)$ and $q(\boldsymbol{x},\boldsymbol{y})$ (the Gaussian kernel with bandwidth σ_q) are key components to ensure that $k_\omega(\boldsymbol{x},\boldsymbol{y})$ is a characteristic kernel (Liu et al., 2020b) (ensuring that, SAMMD equals zero if and only if two distributions are the same (Liu et al., 2020b)). Since $\hat{f}$ is fixed, the set of parameters of k_ω is $\omega = \{\epsilon_0, \sigma_{\phi_p}, \sigma_q\}$. Based on $k_\omega(\boldsymbol{x},\boldsymbol{y})$ in Eq. (7), SAMMD$(\mathbb{P},\mathbb{Q})$ is

$$\sqrt{\mathbb{E}\left[k_\omega(X,X') + k_\omega(Y,Y') - 2k_\omega(X,Y)\right]},$$

where $X, X' \sim \mathbb{P}$, $Y, Y' \sim \mathbb{Q}$. We can estimate SAMMD$(\mathbb{P},\mathbb{Q})$ using the U-statistic estimator, which is unbiased for SAMMD$^2(\mathbb{P},\mathbb{Q})$:

$$\widehat{\mathrm{SAMMD}}_u^2(S_X,S_Y;k_\omega) = \frac{1}{n(n-1)}\sum_{i\neq j} H_{ij}, \tag{8}$$

where $H_{ij} = k_\omega(\boldsymbol{x}_i,\boldsymbol{x}_j) + k_\omega(\boldsymbol{y}_i,\boldsymbol{y}_j) - k_\omega(\boldsymbol{x}_i,\boldsymbol{y}_j) - k_\omega(\boldsymbol{y}_i,\boldsymbol{x}_j)$.

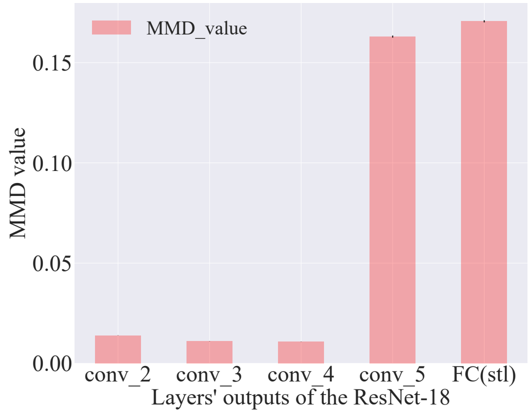


Figure 3. Discrepancy of MMD value between different layers' outputs in $\hat{f}$. The figure shows MMD value between outputs of 5 different layers of ResNet-18. It is clear that, in the conv_5 layers, outputs of natural and adversarial data have larger distributional discrepancy compared to outputs of other 3 convolutional layers. FC(stl) is the second to the last fully-connected layer and also an average pooling layer. Compared to the conv_5 layer, the FC(stl) layer has fewer dimensions and its outputs can help measure the discrepancy between natural and adversarial data well.

Asymtotics and test power of SAMMD. In this part, we analyze the asymtotics of SAMMD when S_Y are adversarial data. Based on the asymtotics of SAMMD, we can estimate its test power that can be used to optimize the SAMMD (i.e., optimizing parameters in $k_\omega(\boldsymbol{x}, \boldsymbol{y})$).

Theorem 1 (Asymptotics under H_1)**.** *Under the alternative $H_1 : S_Y$ are from a stochastic process $\{Y_i\}_{i=1}^{+\infty}$, under mild assumptions, we have*

$$\sqrt{n}(\widehat{\mathrm{SAMMD}}_u^2 - \mathrm{SAMMD}^2) \xrightarrow{d} \mathcal{N}(0, C_1^2\sigma_{H_1}^2),$$

where $Y_i = \mathcal{G}_{\ell,\hat{f}}(\mathcal{B}_\epsilon[X_i]) \sim \mathbb{Q}$, $X_i \sim \mathbb{P}$, $\sigma_{H_1}^2 = 4(\mathbb{E}_Z[(\mathbb{E}_{Z'}h(Z,Z'))^2] - [(\mathbb{E}_{Z,Z'}h(Z,Z'))^2])$, $h(Z,Z') = k_\omega(X,X') + k_\omega(Y,Y') - k_\omega(X,Y') - k_\omega(X',Y)$, $Z := (X,Y)$ and $C_1 < +\infty$ is a constant for a given ω.

The detailed version of Theorem 1 can be found in Appendix D. Using Theorem 1, we have

$$\Pr_{H_1,r}^{\mathrm{SAMMD}} \to \Phi\Big(\frac{\sqrt{n}\mathrm{SAMMD}^2}{C_1\sigma_{H_1}} - \frac{r}{\sqrt{n}\,C_1\sigma_{H_1}}\Big), \tag{9}$$

where $\Pr_{H_1,r}^{\mathrm{SAMMD}} = \Pr_{H_1}\Big(n\widehat{\mathrm{SAMMD}}_u^2 > r\Big)$ is the test power of SAMMD, Φ is the standard normal CDF and r is the rejection threshold related to $\mathbb{P}$ and $\mathbb{Q}$. Via Theorem 1, we know that r, $\mathrm{SAMMD}(\mathbb{P},\mathbb{Q})$, and σ_{H_1} are constants. Thus, for reasonably large n, the test power of SAMMD is dominated by the first term (inside Φ), and we can optimize k_ω by maximizing

$$J(\mathbb{P},\mathbb{Q};k_\omega) = \mathrm{SAMMD}^2(\mathbb{P},\mathbb{Q};k_\omega)/\sigma_{H_1}(\mathbb{P},\mathbb{Q};k_\omega).$$

Algorithm 1 The SAMMD Test

```
Input: $S_X$, $S_Y$, $\hat{f}$, various hyperparameters used below;
$\omega \leftarrow \omega_0$; $\lambda \leftarrow 10^{-8}$;
Split the data as $S_X = S_X^{tr} \cup S_X^{te}$ and $S_Y = S_Y^{tr} \cup S_Y^{te}$;
# Phase 1: train the kernel parameters $\omega$ and $\beta$; on $S_X^{tr}$ and $S_X^{tr}$
for $T = 1, 2, \dots, T_{max}$ do
    $S'_X \leftarrow$ minibatch from $S_X^{tr}$;
    $S'_Y \leftarrow$ minibatch from $S_Y^{tr}$;
    $k_\omega \leftarrow$ kernel function with parameters $\omega$ using Eq. (7);
    $M(\omega) \leftarrow \widehat{\mathrm{SAMMD}}_u^2(S'_X, S'_Y; k_\omega)$ using Eq. (8);
    $V_\lambda(\omega) \leftarrow \hat{\sigma}_{H_1,\lambda}^2(S'_X, S'_Y; k_\omega)$ using Eq. (11);
    $\hat{J}_\lambda(\omega) \leftarrow M(\omega)/\sqrt{V_\lambda(\omega)}$ using Eq. (10);
    $\omega \leftarrow \omega + \eta\nabla_{\mathrm{Adam}}\hat{J}_\lambda(\omega)$;    # maximize $\hat{J}_\lambda(\omega)$
end for
# Phase 2: testing with $k_\omega$ on $S_X^{te}$ and $S_Y^{te}$
$est \leftarrow \widehat{\mathrm{SAMMD}}_b^2(S_X^{te}, S_Y^{te}; k_\omega)$
for $i = 1, 2, \dots, n_{perm}$ do
    Generate $\{W_i^X\}_{i=1}^n$ and $\{W_i^Y\}_{i=1}^m$ using Eq. (6);
    $\{\tilde{W}_i^X\}_{i=1}^n \leftarrow \{W_i^X\}_{i=1}^n - \frac{1}{n}\sum_{i=1}^n W_i^X$;
    $\{\tilde{W}_i^Y\}_{i=1}^m \leftarrow \{W_i^Y\}_{i=1}^m - \frac{1}{m}\sum_{i=1}^m W_i^Y$;
    $perm_i \leftarrow \frac{1}{n(n-1)}\sum_{i,j} H_{ij}\tilde{W}_i^X\tilde{W}_j^Y$;    # resample
end for
Output: $k_\omega$, $est$, $p$-value: $\frac{1}{n_{perm}}\sum_{i=1}^{n_{perm}} \mathbf{1}(perm_i \geq est)$
```

Note that, we omit C_1 in $J(\mathbb{P},\mathbb{Q};k_\omega)$, since C_1 can be upper bounded by a constant C_0 (see Appendix D).

Optimization of SAMMD. Although the higher value of criterion $J(\mathbb{P},\mathbb{Q};k_\omega)$ means higher test power of SAMMD, we cannot directly maximize $J(\mathbb{P},\mathbb{Q};k_\omega)$, since $\mathrm{SAMMD}^2(\mathbb{P},\mathbb{Q};k_\omega)$ and $\sigma_{H_1}(\mathbb{P},\mathbb{Q};k_\omega)$ depend on the particular $\mathbb{P}$ and $\mathbb{Q}$ that are unknown. However, we can estimate it with

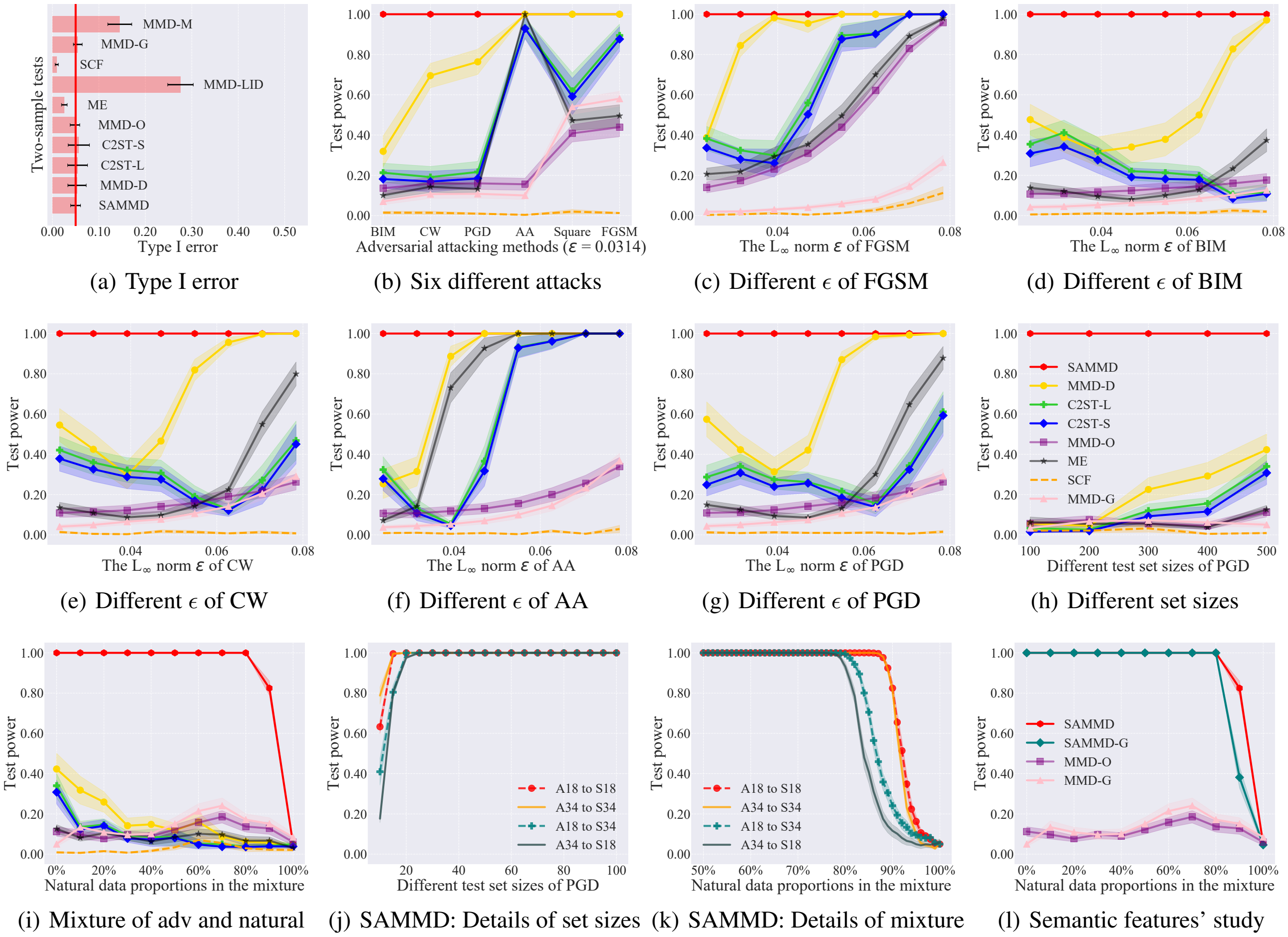
$$\hat{J}_\lambda(S_X,S_Y;k_\omega) := \frac{\widehat{\mathrm{SAMMD}}_u^2(S_X,S_Y;k_\omega)}{\hat{\sigma}_{H_1,\lambda}(S_X,S_Y;k_\omega)}, \tag{10}$$

where $\hat{\sigma}_{H_1,\lambda}^2$ is a regularized estimator of $\sigma_{H_1}^2$ (Liu et al., 2020b):

$$\frac{4}{n^3}\sum_{i=1}^n\left(\sum_{j=1}^n H_{ij}\right)^2 - \frac{4}{n^4}\left(\sum_{i=1}^n\sum_{j=1}^n H_{ij}\right)^2 + \lambda. \tag{11}$$

Then we can optimize SAMMD by maximizing $\hat{J}_\lambda(S_X,S_Y;k_\omega)$ on the training set (see Algorithm 1). Note that, although Sutherland et al. (2017) and Sutherland (2019) have given an unbiased estimator for $\sigma_{H_1}^2$, it is much more complicated to implement.

The SAMMD test. Since adversarial data are probably not IID, we cannot simply use a permutation-based bootstrap method to simulate the null distribution of the SAMMD test



(a) Type I error
(b) Six different attacks
(c) Different ϵ of FGSM
(d) Different ϵ of BIM
(e) Different ϵ of CW
(f) Different ϵ of AA
(g) Different ϵ of PGD
(h) Different set sizes
(i) Mixture of adv and natural
(j) SAMMD: Details of set sizes
(k) SAMMD: Details of mixture
(l) Semantic features' study

Figure 4. Results of adversarial data detection. Subfigure (a) reports the type I error when S_Y are natural data. The ideal type I error should be around α (red line, $\alpha = 0.05$ in this paper). Subfigures (b)-(l) report the test power (i.e., the detection rate) when S_Y are adversarial data (or the mixture of adversarial and natural data). The ideal test power is 1 (i.e., 100% detection rate). Subfigures (b) - (i) share the same legend presented in subfigure (h). Details of subfigures are explained in Section 7.

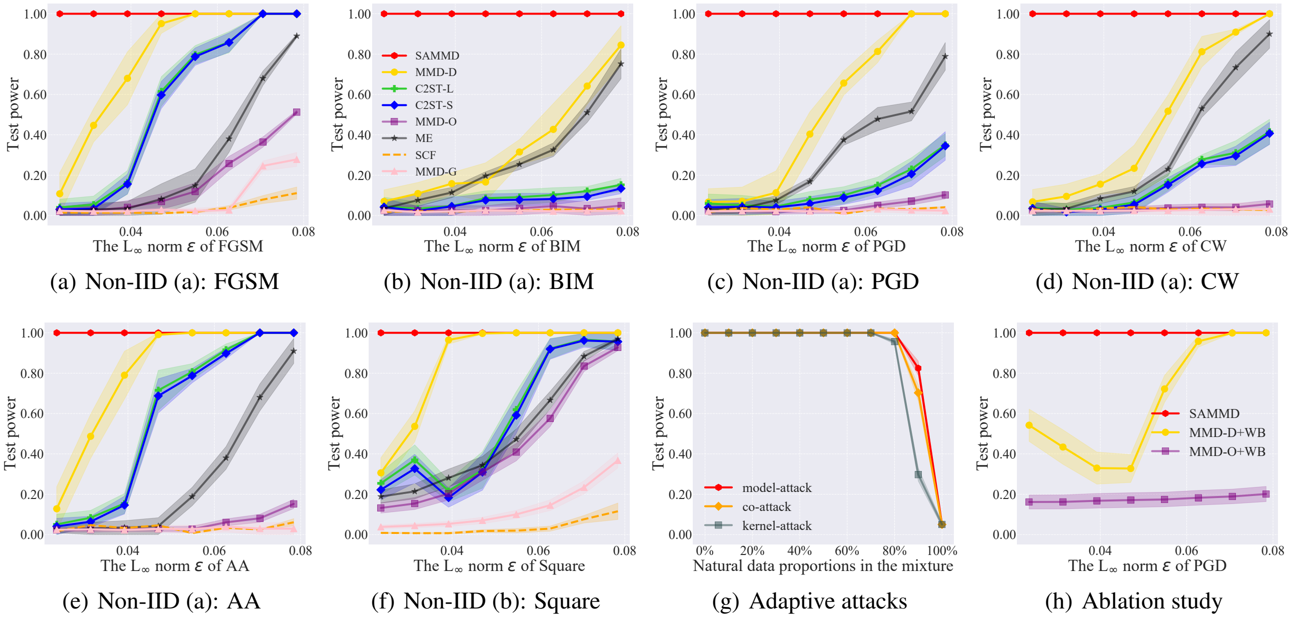
(Chwialkowski et al., 2014). To address this issue, wild bootstrap (Shao, 2010) is used to help simulate the null distribution of SAMMD, then we can test if S_Y are from $\mathbb{P}$. To the end, the Algorithm 1 shows the whole procedure of the SAMMD test. In Appendix D, it has been shown that, under mild assumptions, the proposed SAMMD test is a provably consistent test to detect adversarial attacks.

7. Experiments

We verify detection methods on the ResNet-18 and ResNet-34 trained on the *CIFAR-10* and the *SVHN*. We also validate performance of SAMMD on the large network Wide ResNet (WRN-32-10) (Zagoruyko & Komodakis, 2016) and the large dataset *Tiny-Imagenet*. Configuration of all experiments is in Appendix E. Detailed experimental results are presented in Appendix F. The code of our SAMMD test is available at `github.com/Sjtubrian/SAMMD`.

Baselines. We compare SAMMD test with 6 existing two-sample tests: 1) MMD-G test used by (Grosse et al., 2017); 2) MMD-O test (Sutherland et al., 2017); 3) Mean embedding (ME) test (Jitkrittum et al., 2016); 4) Smooth characteristic functions (SCF) test (Chwialkowski et al., 2015); 5) Classifier two-sample test (C2ST) (Liu et al., 2020b; Lopez-Paz & Oquab, 2017); 6) MMD-D test (Liu et al., 2020b).

Besides, we also try to construct two new MMD tests based on features commonly used by adversarial data classification methods: 1) MMD-LID: the MMD with a Gaussian kernel whose inputs are *local intrinsic dimensionality* (LID) features (Ma et al., 2018) of two samples. Then we optimize the Gaussian kernel by maximizing its test power; and 2) MMD-M: the MMD with a Gaussian kernel whose inputs are mahalanobis distance based features (Lee et al., 2018) of two samples. Then, we optimize the bandwidth of the Gaussian kernel by maximizing the test power.



(a) Non-IID (a): FGSM (b) Non-IID (a): BIM (c) Non-IID (a): PGD (d) Non-IID (a): CW

(e) Non-IID (a): AA (f) Non-IID (b): Square (g) Adaptive attacks (h) Ablation study

Figure 5. Results of adversarial data detection. Subfigures (a)-(f) report the test power (i.e., the detection rate) when S_Y are non-IID adversarial data. Subfigure (g) reports the test power when S_Y are adaptive adversarial data. Subfigure (h) reports an ablation study. Subfigures (a)-(f) share the same legend presented in subfigure (b). Details of subfigures are explained in Section 7.

Test power on different attacks. We first report the type I error of our SAMMD test and 9 baselines when S_Y are natural data in Figure 4a. It is clear that MMD-LID test and MMD-M test have much higher type I error than the given threshold $\alpha = 0.05$ (the red line in Figure 4a). That is, both baselines are invalid two-sample tests. The main reason is that LID features and mahalanobis-distance features are sensitive to any perturbation. The sensitivity leads to that MMD-LID test and MMD-M test will recognize natural data as adversarial data. Other methods except for SCF maintain reasonable type I errors. Since MMD-LID test and MMD-M test are invalid two-sample tests, we do not validate the test power of them in the remaining experiments.

For 6 different attacks, FGSM, BIM, PGD, AA, CW and Square (Non-IID(b)), we report the test power of all tests when S_Y are adversarial data (L_∞ norm $\epsilon = 0.0314$; set size $= 500$) in Figure 4b. Results show that SAMMD test performs the best and achieves the highest test power.

Test power on different ϵ. In addition to different adversarial attacks, different perturbation bound ϵ can also affect the adversarial data generation process. If the adversarial attack is within a small perturbation bound, the generated adversarial data is not sufficient to fool the well-trained natural-data classifier (Tramèr et al., 2020). However, if the adversarial attack is within a big perturbation bound, natural information contained in images will be completely lost (Tramèr et al., 2020; Zhang et al., 2020a). Following previous studies (Carlini & Wagner, 2017b; Madry et al., 2018; Wang et al., 2019; Tramèr et al., 2020; Zhang et al., 2020a; Chen et al., 2020; Wu et al., 2020; Zhang et al., 2020d), we set the L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$. The lower bound of 0.0235 is calculated by 6/255, i.e., the maximum variation of each pixel value is 6 intensities, and the upper bound of 0.0784 is calculated by 20/255. This range covers all possible ϵ used in the literature (Madry et al., 2018; Zhang et al., 2020a).

We report the average test power (with its standard error) on different ϵ of FGSM, BIM, CW, AA and PGD (set size $= 500$) in Figure 4(c)-(g). For the non-IID adversarial data mentioned in Section 5, we also report the average test power on different ϵ of the Non-IID (a) and the Non-IID (b) in Figure 5(a)-(f). Given the training set of *CIFAR-10*, we use FGSM, BIM, CW, AA and PGD to generate the Non-IID (a). Given the testing set of *CIFAR-10*, we use Square to generate the adversarial data four times and mix them into the Non-IID (b). Results show that our SAMMD test also achieves the highest test power all the ϵ.

Test power on different set sizes. The effective of previous kernel non-parametric two-sample tests like C2ST and MMD-D test (Lopez-Paz & Oquab, 2017; Liu et al., 2020b) depends on a large size of data. Namely, they can only measure the discrepancy well when there are a large batch of data. Hence, we evaluate the performance of our SAMMD test and baselines with different set sizes. Experiments results are reported in Figure 4h, which shows that our SAMMD test is suitable to different data sizes.

Test power on the mixture of adversarial data and natural data. For practical concerns, it is often that only part of data is adversarial. We analyze test power of the SAMMD test and baselines in this case, with natural data mixture proportion ranging from 0% to 100%. The experimental results of PGD (L_∞ norm $\epsilon = 0.0314$; set size $= 500$) are presented in Figure 4i. Results show that the performance of our SAMMD test is much better than all baselines.

Semantic featurizers meet unknown adversarial data. In the above setting, the semantic featurizers $\phi_p(\cdot)$ are also the classifiers subjected to adversarial attacks. In this part, we also consider that a dataset to be tested contains the adversarial data acquired by unknown classifiers. Hence, we analyze the performance when adversarial data and semantic features are acquired by different classifiers. We train two classifiers (ResNet-18 and ResNet-34) on natural data. One is used to acquire adversarial data (A18/A34), and the other is used to acquire semantic features (S18/S34).

Experiments results of our SAMMD test with different set sizes (from 10 to 100) are presented in Figure 4j. Experiments results of our SAMMD test with mixture proportion (from 0% to 100%) are presented in Figure 4k. In Figures 4j and 4k, the attack method is PGD (L_∞ norm $\epsilon = 0.0314$; set size $= 500$). Results clearly show that our SAMMD test can also work well in this case. It is the existence of adversarial transferability that can help our SAMMD test defend against such attacks.

Study of Semantic features. In order to verify that semantic features are better to help measure the distribution discrepancy between adversarial data and natural data than raw features, we also test the semantic features (the same with the SAMMD test) and the Gaussian kernel of the fixed bandwidth (SAMMD-G in Figure 4l). Experiments in Figure 4l confirms the importance of semantic features.

The SAMMD test meets adaptive attacks. In the case where the attacker is aware of our SAMMD kernel, we evaluate our SAMMD test from the security standpoint. Compared to other defenses, the advantage of our method in security is adaptive defense. In our detection mechanism, the semantic-aware deep kernel is trained on part of unknown data (to be tested), that is to say, for each input of data in the test, parameters of our semantic-aware deep kernel can be adaptively trained to be powerful. Therefore, the target of the adaptive attack can only be the SAMMD-G mentioned above which has fixed parameters.

First, we use the PGD white-box attack to minimize the $M(\omega)/\sqrt{V_\lambda(\omega)}$ in Eq. (10) and obtain examples (kernel-attack in Figure 5g), and 89.08% of them can fool the pre-trained ResNet-18. Then, we obtain examples using the PGD white-box attack to minimize the $M(\omega)/\sqrt{V_\lambda(\omega)}$ in Eq. (10) and maximize the cross entropy loss in Eq. (5) (co-attack in Figure 5g), and 61.34% of them can fool the pre-trained ResNet-18. The examples acquired by model attack is the adversarial examples, 100.00% of them can fool the pre-trained ResNet-18. Experiments in Figure 5g show that these examples fail to fool our SAMMD test. And attacking such a statistic test will also reduce the ability of adversarial data to mislead a well-trained classifier.

Ablation study. To illustrate the effectiveness of semantic features, we compare our SAMMD test with MMD-O test and MMD-D test after wild bootstrap process (MMD-O+WB, MMD-D+WB). Experiments results are reported in Figure 5h, which verifies that semantic features are better to help measure the distribution discrepancy between adversarial data and natural data than raw features (MMD-O+WB) and the learned features (MMD-D+WB) (Liu et al., 2020b).

8. Conclusion

Two-sample tests could in principle detect any distributional discrepancy between two datasets. However, previous studies have shown that the MMD test, as the most powerful two-sample test, is unaware of adversarial attacks. In this paper, we find that previous use of MMD on adversarial data detection missed three key factors, which *significantly* limits its power. To this end, we propose a simple and effective test that is cooperated with a new semantic-aware kernel—*semantic-aware MMD* (SAMMD) test, to take care of the three factors simultaneously. Experiments show that our SAMMD test can successfully detect adversarial attacks. Thus, we argue that *MMD is aware of adversarial attacks*, which lights up a novel road for adversarial attack detection based on two-sample tests. We also recommend practitioners to use our SAMMD test when they wish to check whether the dataset they acquired contains adversarial data.

Acknowledgements

RZG and BH were supported by HKBU Tier-1 Start-up Grant and HKBU CSD Start-up Grant. BH was also supported by the RGC Early Career Scheme No. 22200720, NSFC Young Scientists Fund No. 62006202 and HKBU CSD Departmental Incentive Grant. TLL was supported by Australian Research Council Project DE-190101473. JZ, GN and MS were supported by JST AIP Acceleration Research Grant Number JPMJCR20U3, Japan, and MS was also supported by Institute for AI and Beyond, UTokyo. FL would like to thank Dr. Yiliao Song and Dr. Wenkai Xu for productive discussions.

References

Andriushchenko, M., Croce, F., Flammarion, N., and Hein, M. Square attack: a query-efficient black-box adversarial attack via random search. In *ECCV*, 2020.

Bai, Y., Feng, Y., Wang, Y., Dai, T., Xia, S.-T., and Jiang, Y. Hilbert-based generative defense for adversarial examples. In *ICCV*, 2019.

Barreno, M., Nelson, B., Joseph, A. D., and Tygar, J. D. The security of machine learning. *Machine Learning*, 81 (2):121–148, 2010.

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

Borgwardt, K. M., Gretton, A., Rasch, M. J., Kriegel, H.-P., Schölkopf, B., and Smola, A. J. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.

Carlini, N. and Wagner, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 3–14, 2017a.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pp. 39–57. IEEE, 2017b.

Chen, C., Seff, A., Kornhauser, A., and Xiao, J. Deepdriving: Learning affordance for direct perception in autonomous driving. In *ICCV*, 2015.

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, T., Liu, S., Chang, S., Cheng, Y., Amini, L., and Wang, Z. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *CVPR*, 2020.

Chwialkowski, K., Sejdinovic, D., and Gretton, A. A wild bootstrap for degenerate kernel tests. In *NeurIPS*, 2014.

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

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Denker, M. and Keller, G. On u-statistics and v. mise'statistics for weakly dependent processes. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 64(4):505–522, 1983.

Fang, T., Lu, N., Niu, G., and Sugiyama, M. Rethinking importance weighting for deep learning under distribution shift. In *NeurIPS*, 2020a.

Fang, Z., Lu, J., Liu, F., Xuan, J., and Zhang, G. Open set domain adaptation: Theoretical bound and algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 2020b.

Feinman, R., Curtin, R. R., Shintre, S., and Gardner, A. B. Detecting adversarial samples from artifacts. *arXiv:1703.00410*, 2017.

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

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

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. Measuring statistical dependence with Hilbert-Schmidt norms. In *ALT*, 2005.

Gretton, A., Fukumizu, K., Teo, C. H., Song, L., Schölkopf, B., and Smola, A. J. A kernel statistical test of independence. In *NeurIPS*, 2008.

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.

Grosse, K., Manoharan, P., Papernot, N., Backes, M., and McDaniel, P. On the (statistical) detection of adversarial examples. *arXiv:1702.06280*, 2017.

He, W., Li, B., and Song, D. Decision boundary analysis of adversarial examples. In *ICLR*, 2018.

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.

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

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

Kloft, M. and Laskov, P. Security analysis of online centroid anomaly detection. *The Journal of Machine Learning Research*, 13(1):3681–3724, 2012.

Kurakin, A., Goodfellow, I., Bengio, S., et al. Adversarial examples in the physical world. In *ICLR*, 2017.

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018.

Leucht, A. and Neumann, M. H. Dependent wild bootstrap for degenerate U-and V-statistics. *Journal of Multivariate Analysis*, 117:257–280, 2013.

Li, B. and Vorobeychik, Y. Feature cross-substitution in adversarial classification. In *NeurIPS*, 2014.

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

Li, X. and Li, F. Adversarial examples detection in deep networks with convolutional filter statistics. In *ICCV*, 2017.

Liu, F., Lu, J., Han, B., Niu, G., Zhang, G., and Sugiyama, M. Butterfly: A panacea for all difficulties in wildly unsupervised domain adaptation. In *NeurIPS LTS Workshop*, 2019.

Liu, F., Lu, J., and Zhang, G. Multi-source heterogeneous unsupervised domain adaptation via fuzzy-relation neural networks. *IEEE Trans. Fuzzy Syst.*, 2020a.

Liu, F., Xu, W., Lu, J., Zhang, G., Gretton, A., and Sutherland, D. J. Learning deep kernels for non-parametric two-sample tests. In *ICML*, 2020b.

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

Ma, X., Li, B., Wang, Y., Erfani, S. M., Wijewickrema, S., Schoenebeck, G., Song, D., Houle, M. E., and Bailey, J. Characterizing adversarial subspaces using local intrinsic dimensionality. In *ICLR*, 2018.

Ma, X., Niu, Y., Gu, L., Wang, Y., Zhao, Y., Bailey, J., and Lu, F. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, 2021.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov): 2579–2605, 2008.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Metzen, J. H., Genewein, T., Fischer, V., and Bischoff, B. On detecting adversarial perturbations. *arXiv:1702.04267*, 2017.

Nguyen, A., Yosinski, J., and Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Odén, A., Wedel, H., et al. Arguments for fisher's permutation test. *The Annals of Statistics*, 3(2):518–520, 1975.

Oneto, L., Donini, M., Luise, G., Ciliberto, C., Maurer, A., and Pontil, M. Exploiting MMD and Sinkhorn divergences for fair and transferable representation learning. In *NeurIPS*, 2020.

Rouhani, B. D., Samragh, M., Javidi, T., and Koushanfar, F. Curtail: Characterizing and thwarting adversarial deep learning. *arXiv:1709.02538*, 2017.

Shao, X. The dependent wild bootstrap. *Journal of the American Statistical Association*, 105(489):218–235, 2010.

Sriperumbudur, B. K., Gretton, A., Fukumizu, K., Schölkopf, B., and Lanckriet, G. R. Hilbert space embeddings and metrics on probability measures. *The Journal of Machine Learning Research*, 11:1517–1561, 2010.

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

Sugiyama, M., Suzuki, T., Itoh, Y., Kanamori, T., and Kimura, M. Least-squares two-sample test. *Neural Networks*, 24(7):735–751, 2011.

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

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.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Tramèr, F., Behrmann, J., Carlini, N., Papernot, N., and Jacobsen, J.-H. Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. In *ICML*, 2020.

Wang, H., Chen, T., Gui, S., Hu, T.-K., Liu, J., and Wang, Z. Once-for-all adversarial training: In-situ tradeoff between robustness and accuracy for free. In *NeurIPS*, 2020a.

Wang, Y., Ma, X., Bailey, J., Yi, J., Zhou, B., and Gu, Q. On the convergence and robustness of adversarial training. In *ICML*, 2019.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020b.

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

Wu, D., Xia, S.-T., and Wang, Y. Adversarial weight perturbation helps robust generalization. In *NeurIPS*, 2020.

Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., and Sugiyama, M. Relative density-ratio estimation for robust distribution comparison. In *NeurIPS*, 2011.

Yoshihara, K.-i. Limiting behavior of u-statistics for stationary, absolutely regular processes. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 35 (3):237–252, 1976.

Yu, X., Liu, T., Gong, M., Zhang, K., Batmanghelich, K., and Tao, D. Label-noise robust domain adaptation. In *ICML*, 2020.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *BMVC*, 2016.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.

Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., and Kankanhalli, M. Attacks which do not kill training make adversarial learning stronger. In *ICML*, 2020a.

Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. Geometry-aware instance-reweighted adversarial training. In *ICLR*, 2021.

Zhang, K., Gong, M., Stojanov, P., Huang, B., LIU, Q., and Glymour, C. Domain adaptation as a problem of inference on graphical models. In *NeurIPS*, 2020b.

Zhang, T., Yamane, I., Lu, N., and Sugiyama, M. A one-step approach to covariate shift adaptation. In *ACML*, 2020c.

Zhang, Y., Li, Y., Liu, T., and Tian, X. Dual-path distillation: A unified framework to improve black-box attacks. In *ICML*, 2020d.

Zhang, Y., Liu, F., Fang, Z., Yuan, B., Zhang, G., and Lu, J. Clarinet: A one-step approach towards budget-friendly unsupervised domain adaptation. In *IJCAI*, 2020e.

Zhong, L., Fang, Z., Liu, F., Lu, J., Yuan, B., and Zhang, G. How does the combined risk affect the performance of unsupervised domain adaptation approaches? In *AAAI*, 2021.

Zhu, J., Zhang, J., Han, B., Liu, T., Niu, G., Yang, H., Kankanhalli, M., and Sugiyama, M. Understanding the interaction of adversarial training with noisy labels. *arXiv preprint arXiv:2102.03482*, 2021.

A. Related Works

In this section, we briefly review related works used in our paper.

A.1. Adversarial attacks

A growing body of research shows that neural networks are vulnerable to adversarial attacks, i.e., test inputs that are modified slightly yet strategically to cause misclassification (Carlini & Wagner, 2017a; Kurakin et al., 2017; Wang et al., 2019; Zhang et al., 2020d), which seriously threaten the security-critical computer vision systems, such as autonomous driving and medical diagnostics (Chen et al., 2015; Ma et al., 2021; Nguyen et al., 2015; Szegedy et al., 2013). Thus, it is crucial to defend against adversarial attacks (Chen et al., 2020; Wang et al., 2020a; Zhu et al., 2021), for example, by injecting adversarial examples into training data, adversarial training methods have been proposed in recent years (Madry et al., 2018; Bai et al., 2019; Wang et al., 2020b; Zhang et al., 2020a). However, these defenses can generally be evaded by *optimization-based* (Opt) attacks, either wholly or partially (Carlini & Wagner, 2017a; He et al., 2018; Li & Vorobeychik, 2014)

A.2. Adversarial data detection

For the adversarial defense, in addition to improving models' robustness by more effective adversarial training (Chen et al., 2020; Wang et al., 2019; Wu et al., 2020; Zhang et al., 2021), recent studies have instead focused on detecting adversarial data. Based on features extracted from DNNs, most works train classifiers to discriminate adversarial data from both natural and adversarial data. Recent studies include, a cascade detector based on the PCA projection of activations (Li & Li, 2017), detection subnetworks based on activations (Metzen et al., 2017), a logistic regression detector based on Kernel Density KD, and Bayesian Uncertainty (BU) features (Grosse et al., 2017), an augmented neural network detector based on statistical measures, a learning framework that covers unexplored space invulnerable models (Rouhani et al., 2017), a *local intrinsic dimensionality* (LID) based characterization of adversarial data (Ma et al., 2018), a generative classifier based on Mahalanobis distance-based score (Lee et al., 2018).

A.3. Statistical adversarial data detection

In the safety-critical system, it is important to find reliable data (i.e., natural data) and eliminate adversarial data that is statistically different from natural data distribution. Thus, statistical detection methods are also proposed to detect if the upcoming data contains adversarial data (or saying that if upcoming data is from natural data distribution in the view of statistics). A number of these methods have been introduced, including the use of the Maximum Mean Discrepancy (MMD) (Gretton et al., 2012a; Borgwardt et al., 2006) with a simple polynomial-time approximation to test whether the upcoming data are all adversarial data, or all natural images (Grosse et al., 2017), and a kernel density estimation defense used a Gaussian Mixture Model to model outputs from the final hidden layer of a neural network, to test whether the upcoming data belongs to a different distribution than that of natural data (Feinman et al., 2017). However, recent studies have shown these statistical detection failed to work under attack evaluations (Carlini & Wagner, 2017a).

A.4. Two-sample Tests

Two-sample tests aim to check whether two datasets come from the same distribution. Traditional tests such as t-test and Kolmogorov-Smirnov test are the mainstream of statistical applications, but require strong assumptions on the distributions being studied. Researchers in statistics and machine learning have been focusing on relaxing these assumptions, with methods specific to various real-world domains (Sugiyama et al., 2011; Yamada et al., 2011; Kanamori et al., 2012; Gretton et al., 2012a; Jitkrittum et al., 2016; Sutherland et al., 2017; Chen & Friedman, 2017; Ghoshdastidar et al., 2017; Lopez-Paz & Oquab, 2017; Li & Wang, 2018; Kirchler et al., 2020; Liu et al., 2020b). In order to involve distributions with complex structure such as images, deep kernel approaches has been proposed (Sutherland et al., 2017; Wenliang et al., 2019; Jean et al., 2018), the foremost study has shown that kernels parameterized by deep neural nets, can be trained to maximize test power in high-dimensional distribution such as images (Liu et al., 2020b). They propose statistical tests of the null hypothesis that the two distributions are equal against the alternative hypothesis that the two distributions are different. Such tests have applications in a variety of machine learning problems such as domain adaptation, covariate shift, label-noise learning, generative modeling, fairness and causal discovery (Binkowski et al., 2018; Zhang et al., 2020c; Fang et al., 2020a; Gong et al., 2016; Fang et al., 2020b; Liu et al., 2019; Zhang et al., 2020e;b; Liu et al., 2020a; Zhong et al., 2021; Yu et al., 2020; Stojanov et al., 2019; Lopez-Paz & Oquab, 2017; Oneto et al., 2020).

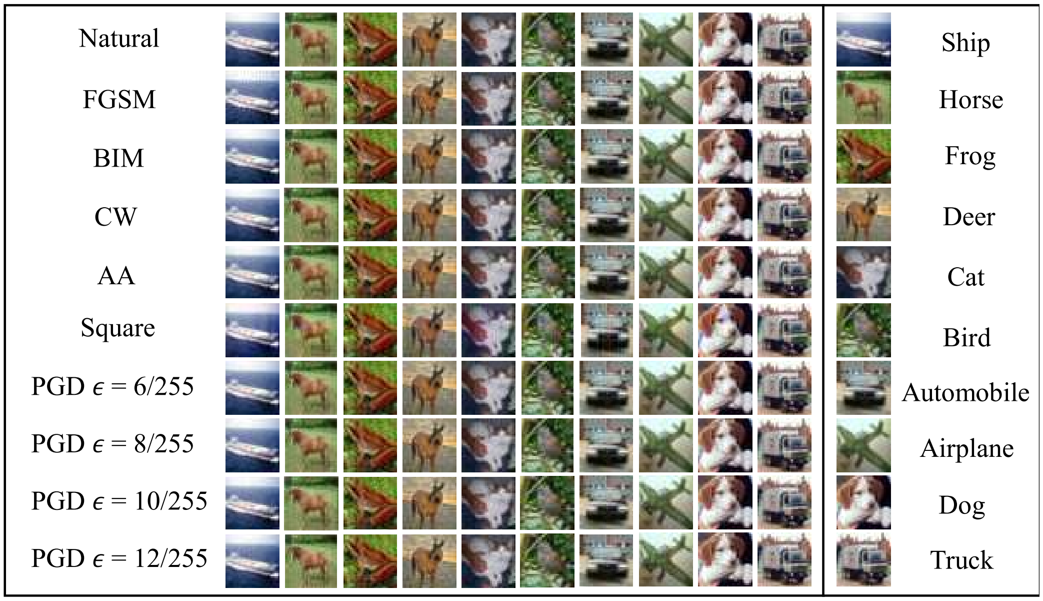


Figure 6. Adversarial data on *CIFAR-10*. We output adversarial examples on a pre-trained ResNet-18, which are attacked by different methods or under different bounded perturbation ϵ.

B. Real-world Scenarios regarding SADD

Scenario 1. As an artificial-intelligence service provider, we need to acquire a client by modeling his/her task well, such as modeling the risk level of manufacturing factory. To finish this task, we need to hire distributed annotators to obtain labeled natural data regarding the risk level in the factory. However, our competitors may *conspire* with several annotators against us, poisoning this training data by injecting malicious adversarial data (Barreno et al., 2010; Kloft & Laskov, 2012). If training data contains adversarial ones, the test accuracy will drop (Zhang et al., 2019), which makes us lose the client unexpectedly. To beware of such adversarial attacks, we can use the MMD test to find reliable annotators providing natural training data.

Scenario 2. As a client, we need to purchase artificial-intelligence services to model our task well, such as modeling the risk level of manufacturing factory mentioned above. Given a variety of models offered by many providers, we should select the optimal one and need to hire distributed annotators to obtain labeled natural data regarding the risk level in our factory. However, some artificial-intelligence service providers may *conspire* with several our annotators, poisoning our testing data by injecting malicious adversarial data (Barreno et al., 2010; Kloft & Laskov, 2012). If the testing data contains adversarial ones which are only in the training set of those conspired providers, the test accuracy of conspired providers' models will surpass that of their competitors (Madry et al., 2018), which makes us fail to select the optimal provider. To beware of such adversarial attacks, we can use the MMD test to find reliable annotators providing natural test data.

C. Hilbert-Schmidt Independence Criteria

The HSIC (Gretton et al., 2008; 2005) is a test statistic to work on independence testing (Gretton et al., 2005). HSIC can be interpreted as the distance between embeddings of the joint distribution and the product of the marginals in a RKHS. More importantly, HSIC between two random variables is zero if and only if the two variables are independent (Sriperumbudur et al., 2010). Under the null hypothesis of independence, $P_{XY} = P_X P_Y$, the minimum variance estimate of HSIC is a degenerate U-statistic. The formulation of the HSIC is as follows (more details can be found in (Gretton et al., 2005)). Given two sets of data S_X and S_Y (with size n), the HSIC can be computed using

$$\begin{aligned}\mathrm{HSIC}(S_X, S_Y) = &\ \mathbb{E}_{(x_1,y_1)\sim p(x,y),(x_2,y_2)\sim p(x,y)}[\kappa_X(x_1,x_2)\kappa_Y(y_1,y_2)] \\ &+ \mathbb{E}_{x_1\sim p(x),x_2\sim p(x),y_1\sim p(y),y_2\sim p(y)}[\kappa_X(x_1,x_2)\kappa_Y(y_1,y_2)] \\ &- 2\mathbb{E}_{(x_1,y_1)\sim p(x,y),x_2\sim p(x),y_2\sim p(y)}[\kappa_X(x_1,x_2)\kappa_Y(y_1,y_2)],\end{aligned} \tag{12}$$

where κ_X and κ_Y are two Gaussian kernel functions whose bandwidths are set to two constants, and

$$\mathbb{E}_{(x_1,y_1)\sim p(x,y),(x_2,y_2)\sim p(x,y)}[\kappa_X(x_1,x_2)\kappa_Y(y_1,y_2)] = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}[\kappa_X(x_i,x_j)\kappa_Y(y_i,y_j)]. \tag{13}$$

D. Asymptotics of the SAMMD

In this section, we will first prove the asymptotics of the SAMMD by assuming that the adversarial data $\{Y_i\}_{i\in\mathbb{Z}^+}$ are an absolutely regular process with mixing coefficients $\{\beta_k\}_{k>0}$ defined in the following.

Definition 1 (Absolutely regular process). *(i) Let $(\Omega, \mathcal{A}, \mathbb{Q})$ be a probability space, and let $\mathcal{A}_1, \mathcal{A}_2$ be sub-σ-field of $\mathcal{A}$. We define*

$$\beta(\mathcal{A}_1, \mathcal{A}_2) = \sup_{A_1,\ldots,A_n,B_1,\ldots,B_m}\sum_{i=1}^{n}\sum_{j=1}^{m}|\mathbb{Q}(A_i\cap B_j) - \mathbb{Q}(A_i)\mathbb{Q}(B_j)|, \tag{14}$$

where the supremum is taken over all partitions $A_1,\ldots,A_n$ and $B_1,\ldots,B_m$ of Ω into elements of $\mathcal{A}_1$ and $\mathcal{A}_2$, respectively. (ii) Given a stochastic process $\{Y_i\}_{i\in\mathbb{Z}^+}$ and integers $1\le a\le b$, we denote by $\mathcal{A}_a^b$ the σ-field generated by the random variables $Y_{a+1},\ldots,Y_b$. We define the mixing coefficients of absolute regularity by

$$\beta_k = \sup_{n\in\mathbb{Z}^+}\beta(\mathcal{A}_1^n, \mathcal{A}_{n+k}^\infty). \tag{15}$$

The process $\{Y_i\}_{i\in\mathbb{Z}^+}$ is called absolutely regular if $\lim_{k\to\infty}\beta_k = 0$.

Then, we can obtain the main theorem in the following.

Theorem 2 (Asymptotics under H_1). *Under the alternative, $H_1 : S_Y$ are from a stochastic progress $\{Y_i\}_{i=1}^{+\infty}$, if $\{Y_i\}_{i=1}^{+\infty}$ is an absolutely regular process with mixing coefficients $\{\beta_k\}_{k>0}$ satisfying $\sum_{k=1}^{+\infty}\beta_k^{\delta/(2+\delta)} < +\infty$ for some $\delta > 0$, then $\widehat{\mathrm{SAMMD}}_u^2$ is $\mathcal{O}_P(1/n)$, and in particular*

$$\sqrt{n}(\widehat{\mathrm{SAMMD}}_u^2 - \mathrm{SAMMD}^2) \xrightarrow{d} \mathcal{N}(0, C_1^2\sigma_{H_1}^2),$$

where $Y_i = \mathcal{G}_{\ell,\hat{f}}(\mathcal{B}_\epsilon[X_i''])\sim\mathbb{Q}$, $X_i''\sim\mathbb{P}$, $\sigma_{H_1}^2 = 4(\mathbb{E}_Z[(\mathbb{E}_{Z'}h(Z,Z'))^2] - [(\mathbb{E}_{Z,Z'}h(Z,Z'))^2])$, $h(Z,Z') = k_\omega(X,X') + k_\omega(Y,Y') - k_\omega(X,Y') - k_\omega(X',Y)$, $Z := (X,Y)$, $X\sim\mathbb{P}$ and X'' are independent and $C_1 < +\infty$ is a constant for a given ω.

Proof. Without loss of generality, let Z be a random variable on a probability space $(\Omega^Z, \mathcal{A}^Z, \mathbb{Q}^Z)$. We will first prove that $\{Z\}_{i=1}^{+\infty}$ is an absolutely regular process. According to Eq. (14), we have

$$\beta^Z(\mathcal{A}_1^Z, \mathcal{A}_2^Z) = \sup_{A_1^Z,\ldots,A_n^Z,B_1^Z,\ldots,B_m^Z}\sum_{i=1}^{n}\sum_{j=1}^{m}|\mathbb{Q}^Z(A_i^Z\cap B_j^Z) - \mathbb{Q}^Z(A_i^Z)\mathbb{Q}^Z(B_j^Z)|, \tag{16}$$

where $\mathcal{A}_1^Z, \mathcal{A}_2^Z$ are sub-σ-field of $\mathcal{A}^Z$ generated by $\{Z\}_{i=1}^{+\infty}$ and the supremum is taken over all partitions $A_1^Z,\ldots,A_n^Z$ and $B_1^Z,\ldots,B_m^Z$ of Ω into elements of $\mathcal{A}_1^Z$ and $\mathcal{A}_2^Z$, respectively. Since X and Y are independent and $Z = (X,Y)$, $\mathbb{Q}^Z(Z\in A^Z) = \mathbb{Q}^Z(X\in A^X, Y\in A) = \mathbb{P}(A^X)\mathbb{Q}(A)$. Thus, we have $\mathbb{Q}^Z(A_i^Z\cap B_j^Z) = \mathbb{P}(A_i^X\cap B_i^X)\mathbb{Q}(A_i\cap B_i)$, $\mathbb{Q}^Z(A_i^Z) = \mathbb{P}(A_i^X)\mathbb{Q}(A_i)$, and $\mathbb{Q}^Z(B_i^Z) = \mathbb{P}(B_i^X)\mathbb{Q}(B_i)$. Since X and X' are independent, we have $\mathbb{P}(A_i^X\cap B_i^X) = \mathbb{P}(A_i^X)\mathbb{P}(B_i^X)$, meaning that

$$\beta^Z(\mathcal{A}_1^Z, \mathcal{A}_2^Z) = \sup_{A_1^Z,\ldots,A_n^Z,B_1^Z,\ldots,B_m^Z}\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbb{P}(A_i^X\cap B_i^X)|\mathbb{Q}(A_i\cap B_j) - \mathbb{Q}(A_i)\mathbb{Q}(B_j)|. \tag{17}$$

Due to the supremum, we can safely make $\mathbb{P}(A_i^X\cap B_i^X)$ be 1. Thus, we have $\beta^Z(\mathcal{A}_1^Z, \mathcal{A}_2^Z) = \beta(\mathcal{A}_1, \mathcal{A}_2)$. Namely, $\{Z\}_{i=1}^{+\infty}$ is an absolutely regular process with mixing coefficients $\{\beta_k\}_{k>0}$ satisfying $\sum_{k=1}^{+\infty}\beta_k^{\delta/(2+\delta)} < +\infty$. Based on Theorem 1 in (Denker & Keller, 1983), since $h(\cdot,\cdot)\le 2$, we know that

$$\sqrt{n}(\widehat{\mathrm{SAMMD}}_u^2 - \mathrm{SAMMD}^2) \xrightarrow{\mathbf{d}} \mathcal{N}(0, 4\sigma^2), \tag{18}$$

where

$$\sigma^2 = \underbrace{\mathbb{E}[h_1(Z_1)]^2}_{\sigma^2_{H_1}} + 2\sum_{j=1}^{+\infty}\mathrm{cov}(h_1(Z_1), h_1(Z_j)), \tag{19}$$

$h_1(Z_j) = \mathbb{E}_{Z_i} h(Z_i, Z_j) - \theta$, and $\theta = \mathbb{E}_{Z_i,Z_j} h(Z_i, Z_j)$. Note that, due to $\mathbb{P} \neq \mathbb{Q}$, we know $\sigma > 0$; due to the absolute regularity, $\sigma < +\infty$. Since the possible dependence between Z_1 and Z_j are caused by Y_1 and Y_j, we will calculate the second term in the right side of Eq. (19) in the following. First, we introduce two notations for the convenience.

$$\mathbb{E}_X^{(i)} = \mathbb{E}_X[k_\omega(X_i, X) - k_\omega(Y_i, X)], \tag{20}$$

$$\mathbb{E}_Y^{(i)} = \mathbb{E}_Y[k_\omega(X_i, Y) - k_\omega(Y_i, Y)]. \tag{21}$$

Thus, we know

$$h_1(Z_1) = \underbrace{\mathbb{E}_X^{(1)} + \mathbb{E}_Y^{(1)}}_{\tilde{h}_1(Z_1)} - \theta,\ h_1(Z_j) = \underbrace{\mathbb{E}_X^{(j)} + \mathbb{E}_Y^{(j)}}_{\tilde{h}_1(Z_j)} - \theta,\ \theta = \mathbb{E}_{Z_1}[\mathbb{E}_X^{(1)} + \mathbb{E}_Y^{(1)}], \tag{22}$$

and

$$\theta^2 = \mathbb{E}_{Z_1}[\mathbb{E}_X^{(1)} + \mathbb{E}_Y^{(1)}]\mathbb{E}_{Z_j}[\mathbb{E}_X^{(j)} + \mathbb{E}_Y^{(j)}] = \Big(\mathbb{E}_{Z_1}[\mathbb{E}_X^{(1)}] + \mathbb{E}_{Z_1}[\mathbb{E}_Y^{(1)}]\Big)\Big(\mathbb{E}_{Z_j}[\mathbb{E}_X^{(j)}] + \mathbb{E}_{Z_j}[\mathbb{E}_Y^{(j)}]\Big). \tag{23}$$

Then, we can compute the $\mathrm{cov}(h_1(Z_1), h_1(Z_j))$.

$$\begin{aligned}\mathrm{cov}(h_1(Z_1), h_1(Z_j)) =& \mathbb{E}_{Z_1,Z_j}[(\tilde{h}_1(Z_1) - \theta)(\tilde{h}_1(Z_j) - \theta)]\\ =& \mathbb{E}_{Z_1,Z_j}[\tilde{h}_1(Z_1)\tilde{h}_1(Z_j) - \theta\tilde{h}_1(Z_j) - \theta\tilde{h}_1(Z_1) + \theta^2]\\ =& \mathbb{E}_{Z_1,Z_j}[\tilde{h}_1(Z_1)\tilde{h}_1(Z_j)] - \theta^2\\ =& \mathbb{E}_{Z_1,Z_j}[(\mathbb{E}_X^{(1)} + \mathbb{E}_Y^{(1)})(\mathbb{E}_X^{(j)} + \mathbb{E}_Y^{(j)})] - \theta^2\\ =& \mathbb{E}_{Z_1,Z_j}[\mathbb{E}_X^{(1)}\mathbb{E}_X^{(j)} + \mathbb{E}_X^{(1)}\mathbb{E}_Y^{(j)} + \mathbb{E}_Y^{(1)}\mathbb{E}_X^{(j)} + \mathbb{E}_Y^{(1)}\mathbb{E}_Y^{(j)}] - \theta^2\\ =& \mathbb{E}_{Z_1}[\mathbb{E}_X^{(1)}]\mathbb{E}_{Z_j}[\mathbb{E}_X^{(j)}] + \mathbb{E}_{Z_1}[\mathbb{E}_X^{(1)}]\mathbb{E}_{Z_j}[\mathbb{E}_Y^{(j)}] + \mathbb{E}_{Z_1}[\mathbb{E}_Y^{(1)}]\mathbb{E}_{Z_j}[\mathbb{E}_X^{(j)}] + \mathbb{E}_{Z_1,Z_j}[\mathbb{E}_Y^{(1)}\mathbb{E}_Y^{(j)}] - \theta^2.\end{aligned} \tag{24}$$

Substituting Eq. (23) into Eq. (24), we have

$$\mathrm{cov}(h_1(Z_1), h_1(Z_j)) = \mathbb{E}_{Z_1,Z_j}[\mathbb{E}_Y^{(1)}\mathbb{E}_Y^{(j)}] - \mathbb{E}_{Z_1}[\mathbb{E}_Y^{(1)}]\mathbb{E}_{Z_j}[\mathbb{E}_Y^{(j)}]. \tag{25}$$

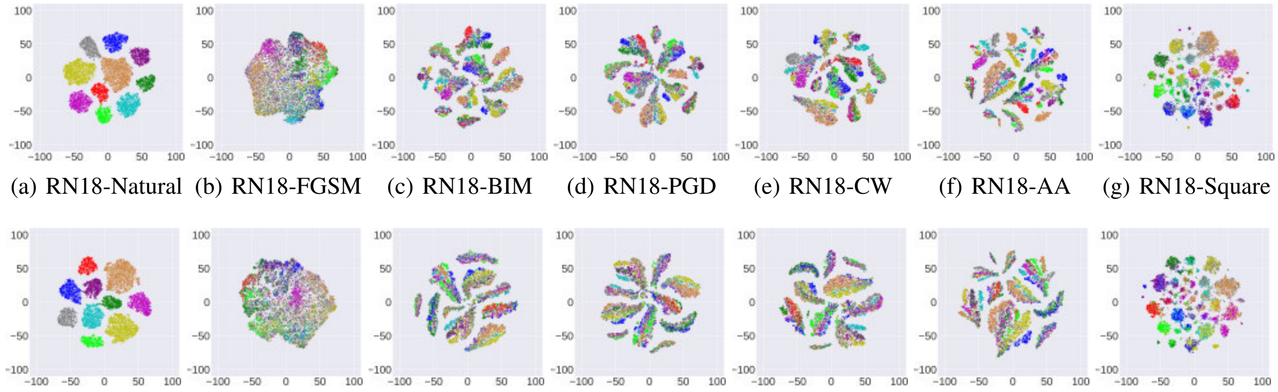
Then, substituting Eq. (21) into Eq. (25), we have

$$\begin{aligned}\mathrm{cov}(h_1(Z_1), h_1(Z_j)) =& \mathbb{E}_{Y_1,Y_j}\big[\mathbb{E}_Y\mathbb{E}_Y[k_\omega(Y_1, Y)k_\omega(Y_j, Y)]\big] - \mathbb{E}_{Y_1}\big[\mathbb{E}_Y[k_\omega(Y_1, Y)]\big]\mathbb{E}_{Y_j}\big[\mathbb{E}_Y[k_\omega(Y_j, Y)]\big]\\ =& \mathbb{E}_Y\mathbb{E}_Y\big[\mathbb{E}_{Y_1,Y_j}[k_\omega(Y_1, Y)k_\omega(Y_j, Y)] - \mathbb{E}_{Y_1}[k_\omega(Y_1, Y)]\mathbb{E}_{Y_j}[k_\omega(Y_j, Y)]\big].\end{aligned} \tag{26}$$

Since $k_\omega(\cdot,\cdot) \leq 1$, according to Lemma 1 in (Yoshihara, 1976), we have $\mathrm{cov}(h_1(Z_1), h_1(Z_j)) < 4\beta_j^{\delta/(2+\delta)}$. Because $\sum_{k=1}^{+\infty}\beta_k^{\delta/(2+\delta)} < +\infty$, we know, $\forall\epsilon' \in (0,1)$, there exists an N such that $\sum_{k=N+1}^{+\infty}\beta_k^{\delta/(2+\delta)} < \epsilon'$. Hence

$$\sum_{j=1}^{+\infty}\mathrm{cov}(h_1(Z_1), h_1(Z_j)) = \sum_{j=1}^{N}\mathbb{E}_Y\mathbb{E}_Y\big[\mathbb{E}_{Y_1,Y_j}[k_\omega(Y_1, Y)k_\omega(Y_j, Y)] - \mathbb{E}_{Y_1}[k_\omega(Y_1, Y)]\mathbb{E}_{Y_j}[k_\omega(Y_j, Y)]\big] + c', \tag{27}$$

where c' is a small constant. Without loss of generality, we assume the small constant c' is smaller than $\mathbb{E}[h_1(Z_1)]^2$. Thus, there exists a constant $C_1^2 - 1$ such that $2\sum_{j=1}^{+\infty}\mathrm{cov}(h_1(Z_1), h_1(Z_j)) = (C_1^2 - 1)\mathbb{E}[h_1(Z_1)]^2$. Namely, $\sigma^2 = C_1^2\sigma_{H_1}^2$, which completes the proof. □



(a) RN18-Natural (b) RN18-FGSM (c) RN18-BIM (d) RN18-PGD (e) RN18-CW (f) RN18-AA (g) RN18-Square

(h) RN34-Natural (i) RN34-FGSM (j) RN34-BIM (k) RN34-PGD (l) RN34-CW (m) RN34-AA (n) RN34-Square

Figure 7. Visualization of outputs using t-SNE. This figure visualizes outputs of the second to last layers in ResNet-18 and ResNet-34. Different colors represent different semantic meanings (i.e., different classes in the testing set of the *SVHN*).

Next, we will show that the bootstrapped SAMMD (shown in the following) has the same asymptotic null distribution as the empirical SAMMD. First, we restate the bootstrapped SAMMD in the following.

$$\begin{aligned}&\widehat{\mathrm{SAMMD}}_w(S_X,S_Y;k_\omega)\\ =&\ \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\tilde{W}_i^x\tilde{W}_j^x k_\omega(x_i,x_j)+\frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\tilde{W}_i^y\tilde{W}_j^y k_\omega(y_i,y_j)-\frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\tilde{W}_i^x\tilde{W}_j^y k_\omega(x_i,y_j),\end{aligned} \tag{28}$$

where

$$\{\tilde{W}_i^X\}_{i=1}^n=\{W_i^X\}_{i=1}^n-\frac{1}{n}\sum_{i=1}^{n}W_i^X,\ \{\tilde{W}_i^Y\}_{i=1}^m=\{W_i^Y\}_{i=1}^m-\frac{1}{m}\sum_{i=1}^{m}W_i^Y. \tag{29}$$

The W_i^X and W_j^Y are generated by

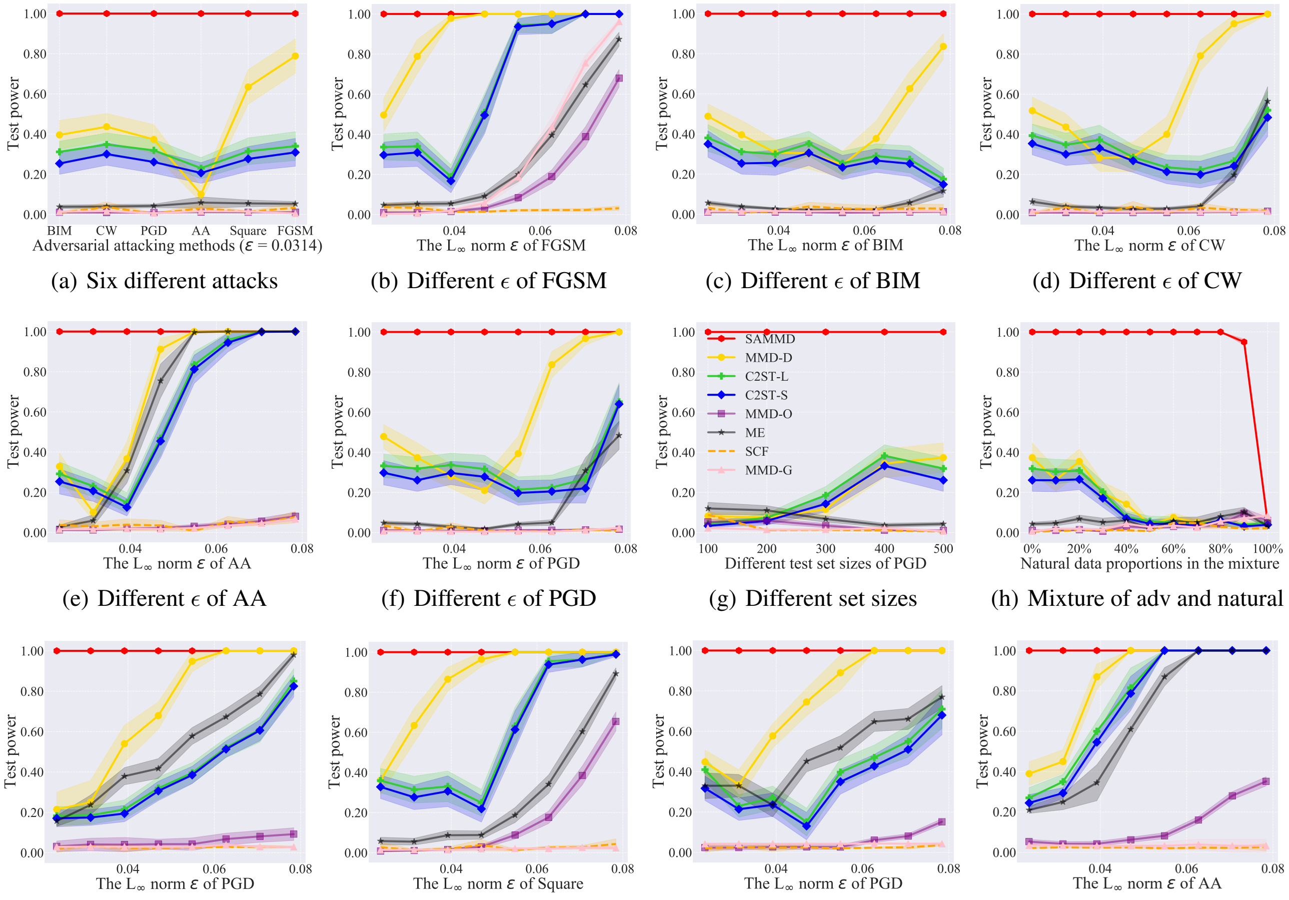
$$W_t=e^{-1/l}W_{t-1}+\sqrt{1-e^{-2/l}}\epsilon_t, \tag{30}$$

where $W_0,\epsilon_0,\ldots,\epsilon_t$ are independent standard normal random variables. Then, following the Proposition 1 in (Chwialkowski et al., 2014), we can directly obtain the following proposition using the relation between β-mixing and τ-mixing presented in Eq. (18) in (Chwialkowski et al., 2014).

Proposition 1. *Let $\{Y_i\}_{i=1}^{+\infty}$ be an absolutely regular process with mixing coefficients $\{\beta_k\}_{k>0}$ satisfying $\beta_k = O(k^{(-6-\epsilon'')(1+\delta)})$ for some $\epsilon > 0$ and $\delta > 0$, $n=\rho_x n'$ and $m=\rho_y n'$, where $n'=n+m$. Then, under the null hypothesis $Y_i\sim\mathbb{P}$, $\psi(\rho_x\rho_y n'\widehat{\mathrm{SAMMD}}_w(S_X,S_Y;k_\omega),\rho_x\rho_y n'\widehat{\mathrm{SAMMD}}(S_X,S_Y;k_\omega))\to 0$ in probability as $n'\to+\infty$, where ψ is the Prokhorov metric.*

E. Experiments Setup

We implement all methods on Python 3.7 (Pytorch 1.1) with a NVIDIA GeForce RTX2080 Ti GPU. The *CIFAR-10* dataset and the *SVHN* dataset can be downloaded via Pytorch. See the codes submitted. Given the $50,000$ images from the *CIFAR-10* training set and $73,257$ digits from the *SVHN* training set, we conduct a standard training on ResNet-18 and ResNet-34 for classification. Given the $100,000$ images from the *Tiny-Imagenet* training set, we conduct a standard training on WRN-32-10 classification. DNNS are trained using SGD with 0.9 momentum, the initial learning rate of 0.01 and the batch size of 128 for 150 epochs. Based on these pre-trained models, adversarial data is generated from *fast gradient sign method* (FGSM) (Goodfellow et al., 2015), *basic iterative methods* (BIM) (Kurakin et al., 2017), *project gradient descent* (PGD) (Madry et al., 2018), *Carlini and Wagner attack* (CW) (Carlini & Wagner, 2017b), *AutoAttack* (AA) (Croce & Hein, 2020) and *Square attack* (Square) (Andriushchenko et al., 2020).



(a) Six different attacks

(b) Different ϵ of FGSM

(c) Different ϵ of BIM

(d) Different ϵ of CW

(e) Different ϵ of AA

(f) Different ϵ of PGD

(g) Different set sizes

(h) Mixture of adv and natural

(i) Non-IID (a): PGD

(j) Non-IID (b): Square

(k) WRN-Tiny: PGD

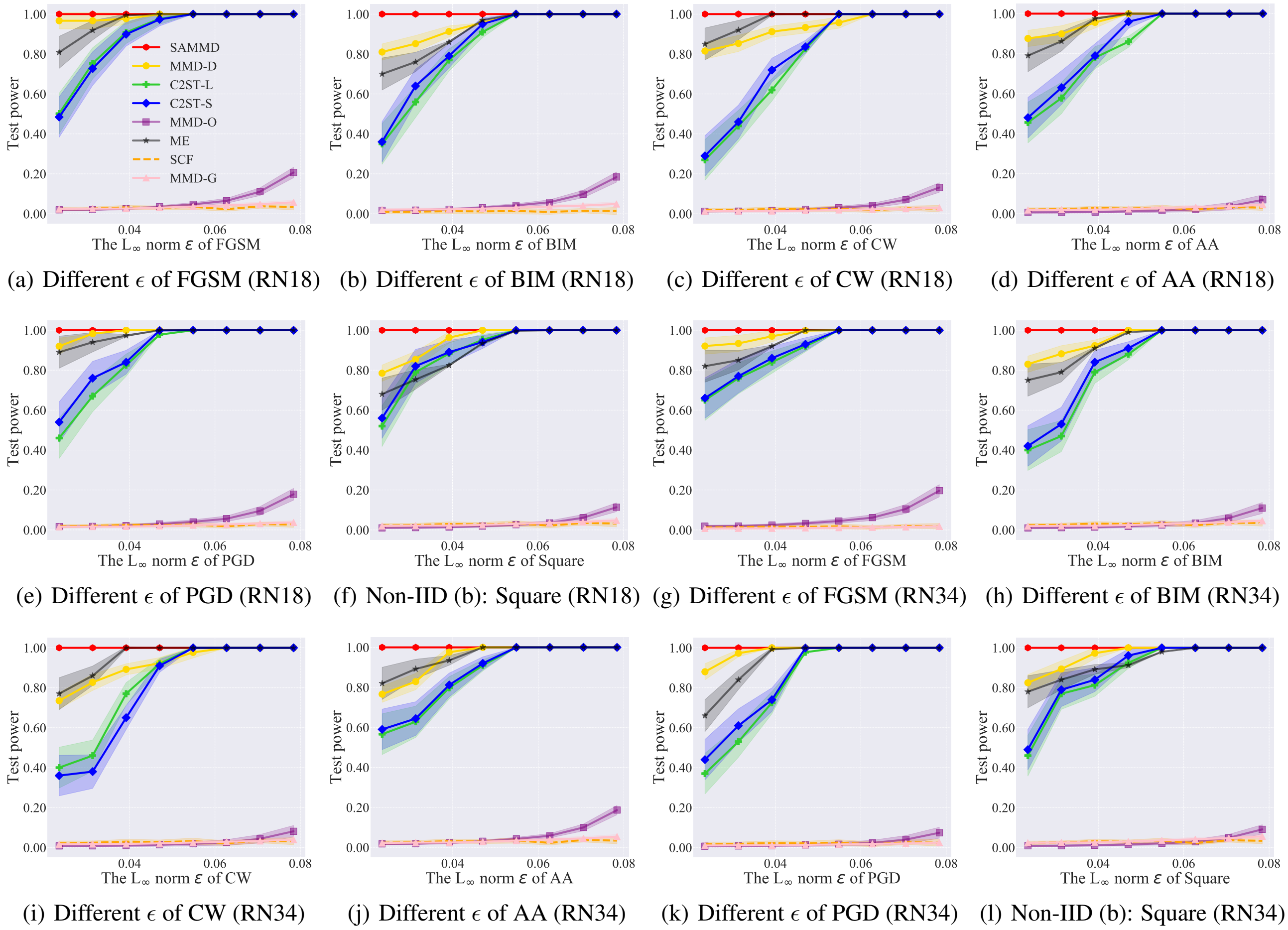
(l) WRN-Tiny: AA

Figure 8. Results of adversarial data detection. Subfigures (a)-(l) report the test power (i.e., the detection rate) when S_Y are adversarial data. The ideal test power is 1 (i.e., 100% detection rate). Subfigure (a) - (j) are the experiments on the adversarial data of the *CIFAR-10* acquired by ResNet-34. Subfigure (k)-(l) are the experiments on the adversarial data of the *Tiny-Imagenet* acquired by WRN-32-10.

For each attack method, we generate eight adversarial datasets with the L_∞-norm bounded perturbation $\epsilon \in [0.0235, 0.0784]$. For attack methods except FGSM, maximum step $K = 20$ and step size $\alpha = \epsilon/10$. For the $10,000$ images from the *CIFAR-10* testing set and $26,032$ digits from the *SVHN* testing set, we only choose the adversarial data, whose original images are correctly classified by the pre-trained models. We extract semantic features from the second to last full connected layer of the well-trained ResNet-18 and ResNet-34. In the wild bootstrap process of SAMMD, given an alternative value set $\{0.1, 0.2, 0.5, 1, 5, 10, 15, 20\}$, we choose the optimal l (in Eq. (6)) depending on whose type I error on natural data is the most close to α ($\alpha = 0.05$ in the paper). The optimal value of l we choose is 0.2. For images from the *CIFAR-10* (or the *SVHN*) testing set and adversarial datasets generated above, we select a subset containing 500 images of the each for S_p^{tr} and S_q^{tr}, and train on that; we then evaluate on 100 random subsets of each, disjoint from the training set, of the remaining data. We repeat this full process 20 times, and report the mean rejection rate of each test. The learning rate of our SAMMD test and all baselines is 0.0002.

F. Additional Experiments

Results of ResNet-34 on the *CIFAR-10*. For the adversarial data acquired by ResNet-34 on *CIFAR-10*, we also compare our SAMMD test with baselines in Figure 8. Figure 8a is a supplement of ResNet-18 to Figure 4 that different ϵ of PGD. For 6 different attacks, FGSM, BIM, PGD, AA, CW and Square (the Non-IID (b)), Figure 8b reports the test power of all tests when S_Y are adversarial data (L_∞ norm $\epsilon = 0.0314$; set size $= 500$). Figure 8(c)-(h) report the average test power on



(a) Different ϵ of FGSM (RN18) (b) Different ϵ of BIM (RN18) (c) Different ϵ of CW (RN18) (d) Different ϵ of AA (RN18)

(e) Different ϵ of PGD (RN18) (f) Non-IID (b): Square (RN18) (g) Different ϵ of FGSM (RN34) (h) Different ϵ of BIM (RN34)

(i) Different ϵ of CW (RN34) (j) Different ϵ of AA (RN34) (k) Different ϵ of PGD (RN34) (l) Non-IID (b): Square (RN34)

Figure 9. Results of adversarial data detection on the *SVHN*. Subfigure (a)-(l) report the test power (i.e., the detection rate) when S_Y are adversarial data. The ideal test power is 1 (i.e., 100% detection rate).

Table 2. Average type I error within natural data and natural data on the *SVHN* and *Tiny-ImageNet*.

Attack	SAMMD	MMD-D	C2ST-L	C2ST-S	MMD-O	ME	SCF	MMD-G
SVHN	0.053±0.017	0.037±0.011	0.036±0.010	0.043±0.012	0.017±0.004	0.022±0.005	0.022±0.006	0.015±0.004
Tiny-ImageNet	0.049±0.015	0.046±0.019	0.051±0.023	0.052±0.016	0.048±0.010	0.039±0.007	0.021±0.008	0.047±0.013

different ϵ of FGSM, BIM, AA, CW and Square (set size $= 500$). Figure 8i reports the average test power on different set sizes. Figure 8j reports the average test power when adversarial data and natural data mix. Results show that our SAMMD test also achieves the highest test power.

Results of ResNet-18 and ResNet-34 on the *SVHN*. We compare the SAMMD test with 6 existing two-sample tests on the *SVHN*. All baselines and experiments setting are the same as those stated in Section 7. We report the type I error in Table 2. The ideal type I error should be around α (0.05 in this paper). For 6 different attacks that FGSM, BIM, PGD, CW, AA, Square and different L_∞-norm bounded perturbation ϵ, we report the test power of all tests when S_Y are adversarial data in Figure 9. Results show that our SAMMD test also performs the best. Compared to results on *CIFAR-10*, adversarial data generated on the *SVHN* is more easily detected by these state-of-the-art tests.

Results of Wide ResNet on the *Tiny-Imagenet*. We also validate the effectiveness of SAMMD on the larger network WRN-32-10 and the larger dataset *Tiny-Imagenet*. All baselines and experiments setting are the same as those stated in

Table 3. The average runtime of the SAMMD test and baselines.

Attack	SAMMD	MMD-D	C2ST-L	C2ST-S	MMD-O	ME	SCF	MMD-G
Runtime(s)	12.51±2.97	47.26±5.92	48.82±4.28	160.78±13.47	11.13±2.15	56.25±8.34	3.59±1.08	1.23±0.17

Section 7. We report the type I error in Table 2. For different attacks PGD and AA, we report the test power of all tests when S_Y are adversarial data in Figure 8 (k)-(l). Results show that our SAMMD test also performs the best.

Time complexity of the SAMMD test. Let E denote the cost of computing an embedding $\phi_p(\boldsymbol{x})$, and K denote the cost of computing $s_{\hat{f}}(\boldsymbol{x}, \boldsymbol{y})$ given $\phi_p(\boldsymbol{x})$, $\phi_p(\boldsymbol{y})$ in Eq. (7). Then each iteration of training in Algorithm 1 costs $O(mE + m^2K)$, where m is the minibatch size.

The average runtime. For images from the *CIFAR-10* testing set and adversarial datasets generated by PGD, we select the subset containing 500 images of the each for S_p^{tr} and S_q^{tr}, and train on that; we then evaluate on 100 random subsets of each, disjoint from the training set, of the remaining data. We repeat this full process 1 times and report the average runtime of our SAMMD test and baselines in Table 3, and the units are seconds.