A Primer on Maximum Mean Discrepancy and its Applications

Zijian Liu
Courant Institute of Mathematical Sciences
New York University
zl3067@nyu.edu

Ruoyi Wang
Courant Institute of Mathematical Sciences
New York University
rw2676@nyu.edu

Hanyu Zhou
Courant Institute of Mathematical Sciences
New York University
hz1649@nyu.edu

Abstract

In many different areas of machine learning, a way to measure the distance between two probability measures is a fundamental requirement for optimization, sampling, and hypothesis testing. In this survey, we will give a primer for a special kind of probability metric: the Maximum Mean Discrepancy (MMD), which has an analytic expression and is closely related to kernel methods. We then discuss some prominent applications to demonstrate the power and importance of MMD to machine learning.

1 Introduction

In this survey, we discuss the Maximum Mean Discrepancy (MMD), an important measure of the distance between two distributions. MMD is a special member of the Integral Probability Metric (IPM) family since it has an analytic solution, with the help of Reproducing Kernel Hilbert Space (RKHS). Due to its close connection to kernel methods and nice properties of IPM as optimization objectives, MMD is widely applicable in a variety of scenarios, including hypothesis testing, sampling, and GAN training, where we discuss one prominent application from each category to demonstrate why MMD is an important concept and how MMD are used in practice. Section 3 introduces MMD with the help of IPM and RKHS. Section 4.1 discusses using MMD for hypothesis testing. Section 4.2 discusses using MMD as an optimization objective for generative modeling, with fixed or learned kernels. Section 4.3 discusses using MMD to derive a novel sampling by optimization algorithm. Section 4.4 discusses how MMD naturally arises in unexpected applications such as neural style transfer.

2 Notation

In this survey, $\mathcal{X}$ is the domain of our function. $\langle \cdot, \cdot \rangle$ represents the Euclidean inner product. $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$ represents a RKHS $\mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$ equipped with the inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. $k_{\mathcal{H}}(x, x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the kernel of $\mathcal{H}$. Denote the space of vector-valued functions $f = [f_1, \cdots, f_d]$ with $f_i \in \mathcal{H}$ by $\mathcal{H}^d$, equipped with inner product $\langle f, g \rangle_{\mathcal{H}^d} = \sum_{i=1}^{d} \langle f_i, g_i \rangle_{\mathcal{H}}$. Given a metric space $\mathcal{M}$, we use $B(\mathcal{M})$ to represent the unit ball in that space. Given a probability distribution $\mathbb{P}$, and a function $T : \mathcal{X} \to \mathbb{R}^d$, we use $T_{\#}\mathbb{P}$ to represent the pushforward measure defined by $T_{\#}\mathbb{P}(A) = \mathbb{P}(\boldsymbol{T}^{-1}(A))$ where A is a measurable set. For two probability distribution $\mathbb{P}$ and $\mathbb{Q}$, $\mathbb{P} \otimes \mathbb{Q}$ means the product

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

measure. $\mathrm{KL}(\mathbb{P}\|\mathbb{Q})$ represents the Kullback-Leibler (KL) divergence. In this paper, we always assume that our probability distribution has density. Sometimes, we may abuse notations, which means all the notations used for probability distribution could be used for probability density. We will use $JT(x)$ to represent the Jacobian matrix of $T(x)$.

3 Maximum Mean Discrepancy

We will first introduce the IPM[12], then derive the MMD[8] and its property, and present the conditions under which it is a metric on the space of probability distributions.

3.1 Definition of the Integral Probability Metric

Suppose we want to answer the following question: given two probobility measures $\mathbb{P}$ and $\mathbb{Q}$, we want to measure how close is $\mathbb{P}$ with respect to $\mathbb{Q}$. How can we do that? One such measure is IPM.

Definition (Integral Probability Metric): Let $\mathcal{F}$ be as class of functions $f : \mathcal{X} \to \mathbb{R}$ and let $\mathbb{P}, \mathbb{Q}$ be two probability measures defined on $\mathcal{X}$. We define the IPM as:

$$\mathrm{IPM}[\mathcal{F}, \mathbb{P}, \mathbb{Q}] = \sup_{f \in \mathcal{F}} |\mathbb{E}_{\mathbb{P}}[f] - \mathbb{E}_{\mathbb{Q}}[f]|$$

The notable thing is we always need some constraints on $\mathcal{F}$ to prevent IPM $= +\infty$. For example, if $\mathcal{F} = \{f \in \mathbb{R}^{\mathcal{X}} : \|f\|_{\text{Lip}} \leq 1\}$, we can recover the Wasserstein-1 distance[14] between $\mathbb{P}$ and $\mathbb{Q}$.

3.2 The IPM in Reproducing Kernel Hilbert Spaces

Now, we restrict our IPM function class $\mathcal{F}$ to the unit ball in a RKHS $\mathcal{H}$, named this special kind of IPM as MMD. Then we can establish conditions under which the MMD can be used to distinguish between probability measures. We will give the definition for the reproducing kernel Hilbert space and derive MMD.

3.2.1 Reproducing Kernel Hilbert Spaces

Definition (Kernel): A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel on $\mathcal{X}$ if there exists a Hilbert Space $\mathcal{H}$ and a map $\phi : \mathcal{X} \to \mathcal{H}$ such that for all $x, x' \in \mathcal{X}$ we have

$$k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$$

The map ϕ is called a feature map and $\mathcal{H}$ is called a feature space associated with k. A kernel is said to be positive definite symmetric if the matrix $K_{ij} = k(x_i, x_j)$ is symmetric positive semidefinite .

Definition (Reproducing Kernel Hilbert Space): Let $\mathcal{H}$ be a Hilbert space of real valued functions defined on a non-empty set $\mathcal{X}$. A function $k_{\mathcal{H}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a reproducing kernel of $\mathcal{H}$, and $\mathcal{H}$ is a reproducing kernel Hilbert space, if $k_{\mathcal{H}}$ satisfies:

$$k_{\mathcal{H}}(\cdot, x) \in \mathcal{H}, \forall x \in \mathcal{X}$$
$$\langle f, k_{\mathcal{H}}(\cdot, x) \rangle_{\mathcal{H}} = f(x), \forall x \in \mathcal{X}, \forall f \in \mathcal{H}$$

By Moore-Aronszaj theorem, we know every positive definite kernel k is associated with a unique Reproducing Kernel Hilbert Space $\mathcal{H}$. We can show that reproducing kernels are kernels. Let $\mathcal{H}$ be a Hilbert space on $\mathcal{X}$ with a reproducing kernel $k_{\mathcal{H}}$. Then $\mathcal{H}$ is an RKHS and is also a feature space of $k_{\mathcal{H}}$, the feature map $\phi : \mathcal{X} \to \mathcal{H}$ is defined by:

$$\phi(x) = k_{\mathcal{H}}(\cdot, x)$$

ϕ is called the canonical feature map. The proof begins with fixing an $x' \in \mathcal{X}$ and defining $f = k_{\mathcal{H}}(\cdot, x')$. Then for $x \in \mathcal{X}$ the reproducing property implies:

$$f(x) = \langle f, k_{\mathcal{H}}(\cdot, x) \rangle_{\mathcal{H}} = \langle k_{\mathcal{H}}(\cdot, x'), k_{\mathcal{H}}(\cdot, x) \rangle_{\mathcal{H}} = \langle \phi(x'), \phi(x) \rangle_{\mathcal{H}} = k_{\mathcal{H}}(x, x')$$

This completes the proof.

3.2.2 Mean Embedding and Maximum Mean Discrepancy

Next we extend the notion of feature map to the embedding of a probability distribution.

Definition (Kernel Mean Embedding)[8]: Given a probability measure $\mathbb{P}$. A kernel mean embedding is defined by

$$\mu : \mathbb{P} \mapsto \int k_{\mathcal{H}}(\cdot, x) d\mathbb{P}(x)$$

We call this the mean embedding of $\mathbb{P}$. But we need to show $\mu_{\mathbb{P}} \in \mathcal{H}$. Now we will establish conditions for the existence of $\mu_{\mathbb{P}}$ and show the most important property of $\mu_{\mathbb{P}}$: $\mathbb{E}_{\mathbb{P}}[f] = \langle f, \mu_{\mathbb{P}} \rangle_{\mathcal{H}}$, by following lemma.

Lemma[8]: If $k_{\mathcal{H}}(\cdot, \cdot)$ is $\mathbb{P}$-measurable and $\mathbb{E}_{\mathbb{P}}[\sqrt{k_{\mathcal{H}}(x, x)}] < \infty$ then $\mu_{\mathbb{P}} \in \mathcal{H}$ and $\mathbb{E}_{\mathbb{P}}[f] = \langle f, \mu_{\mathbb{P}} \rangle_{\mathcal{H}}$.

Proof. The linear operator $\mathcal{T}_{\mathbb{P}} f = \mathbb{E}_{\mathbb{P}}[f]$ for all $f \in \mathcal{H}$ is bounded under the assumption, since

$$|\mathcal{T}_{\mathbb{P}} f| = |\mathbb{E}_{\mathbb{P}}[f]| \leq \mathbb{E}_{\mathbb{P}}[|f|] = \mathbb{E}_{\mathbb{P}}[|\langle f, \phi(x) \rangle_{\mathcal{H}}|] \leq \mathbb{E}_{\mathbb{P}}[\sqrt{k_{\mathcal{H}}(x, x)}] \|f\|_{\mathcal{H}}$$

We use Jensen's inequality in the first inequality and Cauchy-Schwarz inequality in the second inequality. By the Riesz representation theorem, there exists $\nu_{\mathbb{P}} \in \mathcal{H}$ such that $\mathcal{T}_{\mathbb{P}} f = \langle f, \nu_{\mathbb{P}} \rangle_{\mathcal{H}}$. As the result, we have

$$\nu_{\mathbb{P}}(x) = \langle \nu_{\mathbb{P}}, k_{\mathcal{H}}(\cdot, x) \rangle_{\mathcal{H}} = \mathbb{E}_{y \sim \mathbb{P}}[k_{\mathcal{H}}(y, x)] = \mu_{\mathbb{P}}(x)$$

Therefore, we know $\mu_{\mathbb{P}} = \nu_{\mathbb{P}} \in \mathcal{H}$ and prove the most important property of $\mu_{\mathbb{P}}$: $\mathbb{E}_{\mathbb{P}}[f] = \langle f, \mu_{\mathbb{P}} \rangle_{\mathcal{H}}$. □

We will further assume that the map $\mathbb{P} \mapsto \mu_{\mathbb{P}}$ is injective. Then when the supremum in the definition of IPM is taken over functions in the unit ball in an universal RKHS $\mathcal{H}$, i.e $\mathcal{F} := \{f : ||f||_{\mathcal{H}} \leq 1\}$, the resulting metric is known as the Maximum Mean Discrepancy.

Definition (Maximum Mean Discrepancy): Given two probability measures $\mathbb{P}$ and $\mathbb{Q}$, a RKHS $\mathcal{H}$ with kernel $k_{\mathcal{H}}$ satisfying our lemma, we define MMD as:

$$\text{MMD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}] = \sup_{f \in B(\mathcal{H})} |\mathbb{E}_{\mathbb{P}}[f] - \mathbb{E}_{\mathbb{Q}}[f]|$$

MMD can be expressed as the distance in RKHS between two mean embeddings:

$$\begin{aligned}
\text{MMD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}] &= \sup_{f \in B(\mathcal{H})} |\mathbb{E}_{\mathbb{P}}[f] - \mathbb{E}_{\mathbb{Q}}[f]| \\
&= \sup_{f \in B(\mathcal{H})} |\langle f, \mu_{\mathbb{P}} \rangle_{\mathcal{H}} - \langle f, \mu_{\mathbb{Q}} \rangle_{\mathcal{H}}| \\
&= \sup_{f \in B(\mathcal{H})} |\langle f, \mu_{\mathbb{P}} - \mu_{\mathbb{Q}} \rangle_{\mathcal{H}}| \\
&= ||\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}||_{\mathcal{H}}
\end{aligned}$$

Since we assume the mean embedding is injective, $\text{MMD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}] = 0$ if and only if $\mathbb{P} = \mathbb{Q}$.

Finally, we introduce the notion of characteristic kernel which is related to the mean element.

Definition (Characteristic Kernel)[11]: Let $(\mathcal{X}, \Sigma_{\mathcal{X}})$ be a measurable space and $(\mathcal{H}, k_{\mathcal{H}})$ be an RKHS over $\mathcal{X}$ with the kernel $k_{\mathcal{H}}$ measurable and bounded. Let $\mathcal{P}$ be a space of all probability measures on $(\mathcal{X}, \Sigma_{\mathcal{X}})$, then the kernel is said to be characteristic with respect to $\Sigma_{\mathcal{X}}$ if the following map is injective:

$$\mathbb{P} \mapsto \mu_{\mathbb{P}}(x) = \mathbb{E}_{y \sim \mathbb{P}}[k_{\mathcal{H}}(x, y)] \in \mathcal{H}, \ \mathbb{P} \in \mathcal{P}$$

where $\mu_{\mathbb{P}}$ is the mean element of the random variable $k_{\mathcal{H}}(\cdot, Y)$ with law $\mathbb{P}$. As we discovered above, this is equivalent to say

$$||\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}||_{\mathcal{H}} = 0 \iff \mathbb{P} = \mathbb{Q}.$$

In other words, any two different probability measures can be distinguished by their kernel means $\mu_{\mathbb{P}}, \mu_{\mathbb{Q}} \in \mathcal{H}$. Gaussian kernel $k(x, y) = e^{-\frac{||x-y||_2^2}{2\sigma^2}}$ is an example of characteristic kernels. Laplacian kernel $k(x, y) = e^{-\frac{\|x-y\|_1}{\sigma^2}}$ is also a characteristic kernel.

4 Applications in Machine Learning

In this section, we will describe several applications of MMD. Readers can find MMD is a powerful tool in many different areas of machine learning.

4.1 A Kernel Two-Sample Test

In practice, given i.i.d observations $X = \{x_1, ..., x_M\}$ and $Y = \{y_1, ..., y_N\}$ from two probability measures $\mathbb{P}$ and $\mathbb{Q}$, we may want to decide whether $\mathbb{P} \neq \mathbb{Q}$. This problem can be recognized as as a hypothesis testing problem, where the null hypothesis is $H_0 : \mathbb{P} = \mathbb{Q}$, and the alternative hypothesis is $H_1 : \mathbb{P} \neq \mathbb{Q}$. Here, we introduce the kernel two-sample test proposed by [8].

In the definition of MMD, we have

$$\begin{aligned}\text{MMD}^2[\mathcal{H}, \mathbb{P}, \mathbb{Q}] &= \|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|_{\mathcal{H}}^2 \\ &= \mathbb{E}_{\mathbb{P}\otimes\mathbb{P}}[k_{\mathcal{H}}(x, x')] - 2\mathbb{E}_{\mathbb{P}\otimes\mathbb{Q}}[k(x, y)] + \mathbb{E}_{\mathbb{Q}\otimes\mathbb{Q}}[k_{\mathcal{H}}(y, y')]\end{aligned}$$

We can construct the following unbiased and biased estimator:

$$\widehat{\text{MMD}}_u^2[\mathcal{H}, \mathbb{P}, \mathbb{Q}] = \frac{1}{M(M-1)} \sum_{i \neq j}^{M} k_{\mathcal{H}}(x_i, x_j) - \frac{2}{MN} \sum_{i,j=1}^{M,N} k_{\mathcal{H}}(x_i, y_j) + \frac{1}{N(N-1)} \sum_{i \neq j}^{N} k_{\mathcal{H}}(y_i, y_j)$$

$$\widehat{\text{MMD}}_b^2[\mathcal{H}, \mathbb{P}, \mathbb{Q}] = \frac{1}{M^2} \sum_{i,j=1}^{M} k_{\mathcal{H}}(x_i, x_j) - \frac{2}{MN} \sum_{i,j=1}^{M,N} k_{\mathcal{H}}(x_i, y_j) + \frac{1}{N^2} \sum_{i,j=1}^{N} k_{\mathcal{H}}(y_i, y_j)$$

Consider the additional assumption $0 \leq k_{\mathcal{H}}(x, y) \leq K$ and $M = N$. Under the null hypothesis,[8] shows a hypothesis test of level α for unbiased estimator $\widehat{\text{MMD}}_u$ has the acceptance region:

$$\widehat{\text{MMD}}_u[\mathcal{H}, \mathbb{P}, \mathbb{Q}] < \frac{4K}{\sqrt{M}} \log \frac{1}{\alpha}$$

for biased estimator $\widehat{\text{MMD}}_b$ has the acceptance region:

$$\widehat{\text{MMD}}_b[\mathcal{H}, \mathbb{P}, \mathbb{Q}] < \sqrt{\frac{2K}{M}} \left(1 + \sqrt{2 \log \frac{1}{\alpha}}\right)$$

For the detailed proof, readers can refer to the appendix of [8]. Hence, using MMD, we can construct a new kind of hypothesis testing.

4.2 MMD-GAN

4.2.1 Generative Adversarial Networks

The generative adversarial networks (GAN) [6] involves a generator G and a discriminator (or critic) D. The goal of the generator is to transform latent codes $z \sim p(z)$ into generated samples which the discriminator cannot distinguish from real data samples, and the goal of the discriminator is to distinguish real data samples and generated samples as good as possible. With i.i.d. data samples, the optimal generator parameter θ_g and discriminator parameter θ_d is the solution to the following min-max game:

$$\min_{\theta_g} \max_{\theta_d} V(\theta_g, \theta_d) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x, \theta_d)] + \mathbb{E}_{z \sim p(z)}[\log (1 - D(G(z, \theta_g), \theta_d))]$$

For a fixed θ_g, the optimal θ_d^* satisfies:

$$D^*_{\theta_g}(x, \theta_d^*) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$$

Then, suppose in each iteration, the discriminator can be trained to optimal. Then, the generator is minimizing the following virtual loss, obtained by plugging the expression of the optimal discriminator to the min-max game:

$$C(\theta_g) = \max_{\theta_d} V(\theta_g, \theta_d) = -2\log 2 + \text{JS}(p_{data} \| p_g)$$

where JS is the Jensen-Shannon divergence. Therefore, assuming the discriminator can be trained to optimal in each step, GAN training is divergence minimization between the data distribution and the generated distribution.

4.2.2 Generative Modeling with MMD minimization

The problem of using Jensen-Shannon divergence, and any f-divergence in general, to train the generator is that the gradient does not provide useful information when the support does not match [2]. One solution, proposed by [3], is the Wasserstein GAN, which uses the Wasserstein distance to replace f-divergence. The observation here is Wasserstein distance is a particular member of the IPM family, and the good properties of Wassersetein distance are also shared by MMD. Therefore, one can train the generator with MMD^2 as the critic [5]:

$$\begin{aligned}\min_{\theta_g} \text{MMD}^2[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g] &= \min_{\theta_g} (\max_{f \in B(\mathcal{H})} \mathbb{E}_{\mathbb{P}_{data}}[f] - \mathbb{E}_{\mathbb{P}_g}[f])^2 \\ &= \min_{\theta_g} \mathbb{E}_{x,x' \sim p_{data}; y,y' \sim p_g}[k_{\mathcal{H}}(x,x) - 2k_{\mathcal{H}}(x,y) + k_{\mathcal{H}}(y,y')]\end{aligned}$$

Using an unbiased empirical estimator for the MMD^2:

$$\widehat{\text{MMD}}^2_u[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g] = \frac{1}{N(N-1)} \sum_{n \neq n'}^{N} k_{\mathcal{H}}(x_n, x'_n) + \frac{1}{M(M-1)} \sum_{m \neq m'}^{M} k_{\mathcal{H}}(y_m, y'_m) - \frac{2}{MN} \sum_{m,n}^{M,N} k_{\mathcal{H}}(x_n, y_m)$$

where M, N is the number of generated samples and real samples, respectively.

Removing terms not containing θ_g, the new objective for the generator is:

$$C(Y_{\theta_g}, X) = \frac{1}{M(M-1)} \sum_{m \neq m'} k_{\mathcal{H}}(y_m, y'_m) - \frac{2}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} k_{\mathcal{H}}(x_n, y_m)$$

which is differentiable w.r.t. θ_g if the kernel is differentiable.

If we pick a fixed kernel (i.e. RBF kernel), the critic is fixed and no longer requires training. Learning the kernel leads to the MMD-GAN and will be discussed in detail in the next section. Here we want to present an interesting generalization bound for this formulation.

Because we are using an empirical estimate of the MMD, the optimal parameter minimizing the empirical MMD might not also minimizing the true MMD. Let $\hat{\theta_g}$ and θ_g^* be the empirical minimizer and true minimizer respectively:

$$\begin{aligned}\hat{\theta_g} &= \inf_{\theta} \widehat{\text{MMD}}[\mathcal{H}, X, Y_\theta] \\ \theta_g^* &= \inf_{\theta} \text{MMD}[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g(\theta)]\end{aligned}$$

Assume $M = N$ and the kernel is bounded by 1. Define

$$\mathcal{G}_{k_{\mathcal{H}}+} = \{g = k_{\mathcal{H}}(G(z;\theta), G(\cdot;\theta)) : z \in Z, \theta \in \Theta\}$$

as all functions over z obtained by fixing the first parameter of the kernel as a generated sample and fixing the generator parameter.

Similarly, define

$$\mathcal{G}^{\mathbb{X}}_{k_{\mathcal{H}}+} = \{g = k_{\mathcal{H}}(x, G(\cdot;\theta)) : x \in \mathbb{X}, \theta \in \Theta\}$$

as all functions over z obtained by fixing the first parameter of the kernel as a real sample and fixing the generator parameter. $\mathbb{X}$ is the set of all possible real samples which X is a subset of.

Assume the fat-shattering dimension $fat_\epsilon(\cdot)$ of the two function classes can be exponentially bounded. Formally, assume

$$\exists \gamma_1 > 1, p_1 \in \mathbb{N} \text{ s.t. } fat_\epsilon(\mathcal{G}_{k_{\mathcal{H}}+}) \leq \gamma_1 \epsilon^{-p_1}$$
$$\exists \gamma_2 > 1, p_2 \in \mathbb{N} \text{ s.t. } fat_\epsilon(\mathcal{G}^{\mathbb{X}}_{k_{\mathcal{H}}+}) \leq \gamma_2 \epsilon^{-p_2}$$

Then, with probability at least $1 - \delta$:

$$\text{MMD}[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g(\hat{\theta})] < \text{MMD}[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g(\theta^*)] + \epsilon$$

where

$$\epsilon = r(p_1, \gamma_1, M) + r(p_2, \gamma_2, M-1) + 12M^{-\frac{1}{2}}\sqrt{\log\frac{2}{\delta}}$$

where

$$r(p, \gamma, M) = C_p\sqrt{\gamma}\begin{cases} M^{-\frac{1}{2}} \text{ if } p < 2 \\ M^{-\frac{1}{2}}\log(m)^{\frac{2}{3}} \text{ if } p = 2 \\ M^{-\frac{1}{p}} \text{ if } p > 2 \end{cases}$$

where constants C_{p_1} and C_{p_2} only depends on p_1 and p_2 respectively.

The key takeaway is the generalization error is bounded by the square root of $1/M$ where M is the data size. Interested readers are advised to refer to the appendix of [5] for the full proof.

4.2.3 MMD-GAN

MMD-GAN [9] can be obtained by minimizing MMD with a learned kernel, since MMD already has a min-max form:

$$\min_{\theta_g} \text{MMD}[\mathcal{H}, \mathbb{P}_{data}, \mathbb{P}_g] = \min_{\theta_g} \max_{f \in B(\mathcal{H})} \mathbb{E}_{\mathbb{P}_{data}}[f] - \mathbb{E}_{\mathbb{P}_g}[f]$$

The caveats are there are constraints for the MMD kernel. In particular, the kernel must be *characteristic* to ensure $\text{MMD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}] \iff \mathbb{P} = \mathbb{Q}$, otherwise the MMD will no longer serve as a good distance measure of distributions. The trick is, if f is an injective function and k is a characteristic kernel, then $k(f(x), f(y))$ is also a characteristic kernel. Therefore, we can pick a characteristic kernel (i.e. RBF kernel), and transform the kernel learning problem into learning an injective function f.

To learn an injective function f, we make use of the fact that if f is injective, then f is invertible if we limit its codomain to its range. In other words, there exists f^{-1} s.t. $\forall x, f^{-1}(f(x)) = x$. Using this as a hint, we learn f and f^{-1} jointly with an autoencoder, and add it to the training objective:

$$\text{MMD}[\mathcal{H}_{f_e}, \mathbb{P}_{data}, \mathbb{P}_g] - \lambda \mathbb{E}_{y \in \mathbb{X} \cup G(Z)} ||y - f_d(f_e(y))||^2$$

where $\mathcal{H}_{f_e}$ is the RKHS of $k(f_e(x), f_e(y))$ where k is any characteristic kernel.

Another very important constraint is that the RKHS feature f of every kernel must satisfy $||f||_{\mathcal{H}} \leq 1$. This constraint, unfortunately, is much harder to enforce. People have borrowed ideas from gradient penalty of Wasserstein GANs and are trying to draw a rigorous connection between gradient penalty and constraints on the IPM witness functions [1].

Empirical study suggests MMD-GAN can be simpler and faster to train because it requires a smaller critic compared to Wasserstein GANs [4]. MMD-GAN has also inspired a more generalized measure for generative models called Kernel Inception Distance (KID). We do not discuss them in further details because they are very specific to GANs and diverges from our discussion of MMD.

4.3 Stein Variational Gradient Descent

In this section, we will introduce another application of MMD distance: Stein Variational Gradient Descent (SVGD)[10], a kind of Quasi-Monte Carlo method.

4.3.1 Kernelized Stein Discrepancy

First, we will give a special kind of MMD distance. In the definition of MMD, we use a unit ball of some RKHS to construct a distance between two probability measures $\mathbb{P}$ and $\mathbb{Q}$. Now our goal is to find some special RKHS $\mathcal{H}_\mathbb{Q}$ satisfying $\forall f \in B(\mathcal{H}_\mathbb{Q}), \mathbb{E}_\mathbb{Q}[f] = 0$. In this case, we have

$$\text{MMD}[\mathcal{H}_\mathbb{Q}, \mathbb{P}, \mathbb{Q}] = \sup_{f \in B(\mathcal{H}_\mathbb{Q})} |\mathbb{E}_\mathbb{P}[f] - \mathbb{E}_\mathbb{Q}[f]| = \sup_{f \in B(\mathcal{H}_\mathbb{Q})} |\mathbb{E}_\mathbb{P}[f]|$$

To find such a RKHS, given a vector-valued function family $\mathcal{F} \subset (\mathbb{R}^d \to \mathbb{R}^d)$, we introduce the Stein operator $\mathcal{T}_\mathbb{Q} : \mathcal{F} \to (\mathbb{R}^d \to \mathbb{R})$ which is a differential operator satisfying $\forall f \in \mathcal{F}, \mathbb{E}_\mathbb{Q}[\mathcal{T}_\mathbb{Q} f] = 0$. Under some assumption of $\mathcal{F}$. One choice of the Stein operator is defined as following

$$\mathcal{T}_\mathbb{Q} f = \frac{\langle \nabla, qf \rangle}{q}$$

If we take $\mathcal{F} = B(\mathcal{H}^d)$ and $k_\mathcal{H}$ is the kernel of $\mathcal{H}$, under additional conditions [13] on $k_\mathcal{H}$ we can show that the image $\mathcal{T}_\mathbb{Q} B(\mathcal{H}^d)$ is a unit ball of another RKHS equipped with the kernel $k_\mathbb{Q}$ defined as

$$\begin{aligned} k_\mathbb{Q}(x, y) =& \langle \nabla_x, \nabla_y k_\mathcal{H}(x, y) \rangle + k_\mathcal{H}(x, y) \langle \nabla \log q(x), \nabla_y \log q(y) \rangle \\ &+ \langle \nabla_x k_\mathcal{H}(x, y), \nabla_y \log q(y) \rangle + \langle \nabla_y k_\mathcal{H}(x, y), \nabla_x \log q(x) \rangle \end{aligned}$$

Hence, we can take $\mathcal{H}_\mathbb{Q} = \mathcal{T}_\mathbb{Q} B(\mathcal{H}^d)$. By the previous result, we know $\text{MMD}[\mathcal{H}_\mathbb{Q}, \mathbb{P}, \mathbb{Q}] = \|\mu_\mathbb{P}\|_{\mathcal{H}_\mathbb{Q}}$ where $\mu_\mathbb{P} \in \mathcal{H}_\mathbb{Q}$ is the mean embedding of $\mathbb{P}$. However, there is an alternative way to calculate $\text{MMD}[\mathcal{H}_\mathbb{Q}, \mathbb{P}, \mathbb{Q}]$

$$\begin{aligned} \text{MMD}[\mathcal{H}_\mathbb{Q}, \mathbb{P}, \mathbb{Q}] &= \sup_{f \in B(\mathcal{H}^d)} |\mathbb{E}_\mathbb{P}[\mathcal{T}_\mathbb{Q} f]| \\ &= \sup_{f \in B(\mathcal{H}^d)} |\mathbb{E}_\mathbb{P}[\langle \nabla \log q, f \rangle + \langle \nabla, f \rangle]| \\ &= \sup_{f \in B(\mathcal{H}^d)} |\mathbb{E}_\mathbb{P}[\sum_{i=1}^d \nabla_i \log q f_i + \nabla_i f_i]| \\ &= \sup_{f \in B(\mathcal{H}^d)} |\mathbb{E}_\mathbb{P}[\sum_{i=1}^d \langle f_i(\cdot), k_\mathcal{H}(x, \cdot) \nabla_i \log q(x) + \nabla_i k_\mathcal{H}(x, \cdot) \rangle_\mathcal{H}]| \\ &= \sup_{f \in B(\mathcal{H}^d)} |\mathbb{E}_\mathbb{P}[\langle f(\cdot), k_\mathcal{H}(x, \cdot) \nabla \log q(x) + \nabla k_\mathcal{H}(x, \cdot) \rangle_{\mathcal{H}^d}]| \\ &= \sup_{f \in B(\mathcal{H}^d)} |\langle f(\cdot), \mathbb{E}_\mathbb{P}[k_\mathcal{H}(x, \cdot) \nabla \log q(x) + \nabla k_\mathcal{H}(x, \cdot)] \rangle_{\mathcal{H}^d}| \\ &= \|\mathbb{E}_\mathbb{P}[k_\mathcal{H}(x, \cdot) \nabla \log q(x) + \nabla k_\mathcal{H}(x, \cdot)]\|_{\mathcal{H}^d} \end{aligned}$$

Readers can check this result is equal to $\|\mu_\mathbb{P}\|_{\mathcal{H}_\mathbb{Q}}$. In this way, we can easily find the function $f^* \in B(\mathcal{H}^d)$ making the equation reach the upper bound.

$$f^*(\cdot) \propto \mathbb{E}_\mathbb{P}[k_\mathcal{H}(x, \cdot) \nabla \log q(x) + \nabla k_\mathcal{H}(x, \cdot)] \text{ and } \|f^*\|_{\mathcal{H}^d} = 1$$

People call this special kind of MMD distance as Kernelized Stein Discrepancy (KSD)[7], we will denote KSD between $\mathbb{P}$ and $\mathbb{Q}$ by $\text{KSD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}]$ instead of $\text{MMD}[\mathcal{H}_\mathbb{Q}, \mathbb{P}, \mathbb{Q}]$.

4.3.2 Construct SVGD

In practice, given a probability measure $\mathbb{Q}$ whose density is q, we may want to find a probability measure $\mathbb{P}$ which makes they are as close as possible under some measurement. The intuition of SVGD is, given $\mathbb{P}$, to minimize the KL divergence between $\mathbb{P}$ and $\mathbb{Q}$. However, this is a hard task and it is not practical to achieve the goal at once. The idea of SVGD is to find a vector field $T(x) = x + \varepsilon f(x)$ where $f \in \mathcal{F}$, $\mathcal{F}$ is a function family with some constraints, which can make:

$$\text{KL}(T_\# \mathbb{P} \| \mathbb{Q}) \leq \text{KL}(\mathbb{P} \| \mathbb{Q})$$

Then we can repeat this procedure until the divergence between our probability measure and the target probability measure is as small as we want. However, how to choose the direction function

f? By the similar intuition of steepest descent, SVGD proposes to find the best direction function $f^* \in \mathcal{F}$ which makes

$$\lim_{\varepsilon \to 0} \frac{\mathrm{KL}(T_{\#}\mathbb{P}\|\mathbb{Q}) - \mathrm{KL}(\mathbb{P}\|\mathbb{Q})}{\varepsilon}$$

as small as possible. By calculation, the following amazing equation holds

$$\lim_{\varepsilon \to 0} \frac{\mathrm{KL}(T_{\#}\mathbb{P}\|\mathbb{Q}) - \mathrm{KL}(\mathbb{P}\|\mathbb{Q})}{\varepsilon} = -\mathbb{E}_{\mathbb{P}}[\langle \nabla \log q, f \rangle + \langle \nabla, f \rangle]$$

This result inspires us to set $\mathcal{F} = B(\mathcal{H}^d)$ and $k_{\mathcal{H}}$ is the kernle of $\mathcal{H}$. By the previous introduction of KSD, we know

$$\inf_{f \in B(\mathcal{H}^d)} \lim_{\varepsilon \to 0} \frac{\mathrm{KL}(T_{\#}\mathbb{P}\|\mathbb{Q}) - \mathrm{KL}(\mathbb{P}\|\mathbb{Q})}{\varepsilon} = -\mathrm{KSD}[\mathcal{H}, \mathbb{P}, \mathbb{Q}]$$

and the best direction function f^* is

$$f^* \propto \mathbb{E}_{\mathbb{P}}[k_{\mathcal{H}}(x, \cdot)\nabla \log q(x) + \nabla k_{\mathcal{H}}(x, \cdot)] \text{ and } \|f^*\|_{\mathcal{H}^d} = 1$$

Hence given a Probability measure $\mathbb{P}$, we can take

$$T(x) = x + \varepsilon \mathbb{E}_{y \sim \mathbb{P}}[k_{\mathcal{H}}(y, x)\nabla \log q(y) + \nabla k_{\mathcal{H}}(y, x)]$$

as our update rule to get a new probability measure $T_{\#}\mathbb{P}$, where ϵ is our step size. In practice, we can use n discrete points to represent our probability measure $\mathbb{P}$. By Monte Carlo estimation, we have the following SVGD algorithm.

Algorithm 1: Stein Variational Gradient Descent

Input : A target distribution with density function $q(x)$ and a set of initial particles $\{x_i^0\}_{i=1}^n$.
Output : A set of particles$\{x_i\}_{i=1}^n$ that approximates the target distribution.
for *iteration* l **do**
  $x_i^{l+1} \leftarrow x_i^l + \varepsilon_l \hat{f}^*(x_i^l)$ where $\hat{f}^*(\cdot) = \frac{1}{n}\sum_{j=1}^n [k_{\mathcal{H}}(x_j^l, \cdot)\nabla \log q(x_j^l) + \nabla k_{\mathcal{H}}(x_j^l, \cdot)]$
  where ε_l is the step size at the l-th iteration.

4.4 Neural Style Transfer

The two applications above are directly inspired by MMD. In some cases, such as neural style transfer, MMD was not the starting point but arises naturally in retrospect.

4.4.1 Canonical Neural Style Transfer

We begin by introducing the original algorithm for neural style transfer. In neural style transfer, one wants to transfer the style of an content image $\mathbf{x}_c$ using a style image $\mathbf{x}_s$ as reference but preserve the content. This is done by the key intuition that in a deep convolutional neural network, the first few convolutional layers extract 'shallow' features which can be viewed as the style, while the deeper convolutional layers capture the 'deep' features which can be viewed as contents.

We can then use this intuition to define a *style loss* and a *content loss* by matching shallow features and deep features, respectively, from a trained deep convolutional neural network on a classification task. Denote the (generated) stylized image as $\mathbf{x}^*$. Denote the feature maps of $\mathbf{x}^*$, $\mathbf{x}_c$ and $\mathbf{x}_s$ in the layer l of a CNN as $\mathbf{F}^l, \mathbf{P}^l, \mathbf{S}^l \in \mathbb{R}^{N_l \times M_l}$ respectively, where N_l is the number of the feature maps in the layer l and M_l is the dimension the feature map.

Then, the content loss is the MSE loss between the features of the stylized image and the content image for a particular layer, typically one of the deep layers:

$$\mathcal{L}_{content}^l = \frac{1}{2}\sum_{i=1}^{N_l}\sum_{j=1}^{M_l}(F_{ij}^l - P_{ij}^l)^2$$

The style loss is a weighed sum of layer-wise style losses:

$$\mathcal{L}_{style} = \sum_l w_l \mathcal{L}^l_{style}$$

For each layer, the style loss matches the Gram matrix between the features of the stylized image and the style image:

$$\mathcal{L}^l_{style} = \frac{1}{4N_l^2 M_l^2} \sum_{i=1}^{N_l} \sum_{j=1}^{N_l} (G^l_{ij} - A^l_{ij})^2$$

where G and A are Gram matrices from the feature maps:

$$G^l_{ij} = \sum_{k=1}^{M_l} F^l_{ik} F^l_{jk}$$

$$A^l_{ij} = \sum_{k=1}^{M_l} S^l_{ik} S^l_{jk}$$

The total loss is:

$$\mathcal{L} = \mathcal{L}_{content} + \lambda \mathcal{L}_{style}$$

MSE loss is well-studied and widely used, but why the style loss take the given specific form was poorly understood. However, the style loss actually corresponds to MMD between the stylized image and the style image, which can give us some intuition for why the style loss makes sense.

4.4.2 Neural Style Transfer as MMD Minimization

MMD can be recovered from the style loss by simply reorganizing the terms. Define $\mathbf{f}^l_{\cdot k}$ and $\mathbf{s}^l_{\cdot k}$ is the k-th column of $\mathbf{F}^l$ and $\mathbf{S}^l$. Then,

$$\begin{aligned}
\mathcal{L}^l_{style} &= \frac{1}{4N_l^2 M_l^2} \sum_{i=1}^{N_l} \sum_{j=1}^{N_l} (\sum_{k=1}^{M_l} F^l_{ik} F^l_{jk} - \sum_{k=1}^{M_l} S^l_{ik} S^l_{jk})^2 \\
&= \frac{1}{4N_l^2 M_l^2} \sum_{i=1}^{N_l} \sum_{j=1}^{N_l} \Big((\sum_{k=1}^{M_l} F^l_{ik} F^l_{jk})^2 + (\sum_{k=1}^{M_l} S^l_{ik} S^l_{jk})^2 - 2(\sum_{k=1}^{M_l} F^l_{ik} F^l_{jk})(\sum_{k=1}^{M_l} S^l_{ik} S^l_{jk}) \Big) \\
&= \frac{1}{4N_l^2 M_l^2} \sum_{k_1=1}^{M_l} \sum_{k_2=1}^{M_l} \Big((\sum_{i=1}^{N_l} F^l_{ik_1} F^l_{ik_2})^2 + (\sum_{i=1}^{N_l} S^l_{ik_1} S^l_{ik_2})^2 - 2(\sum_{i=1}^{N_l} F^l_{ik_1} S^l_{ik_2})^2 \Big) \\
&= \frac{1}{4N_l^2 M_l^2} \sum_{k_1=1}^{M_l} \sum_{k_2=1}^{M_l} \Big(({\mathbf{f}^l_{\cdot k_1}}^T \mathbf{f}^l_{\cdot k_2})^2 + ({\mathbf{s}^l_{\cdot k_1}}^T \mathbf{s}^l_{\cdot k_2})^2 - 2({\mathbf{f}^l_{\cdot k_1}}^T \mathbf{s}^l_{\cdot k_2})^2 \Big) \\
&= \frac{1}{4N_l^2 M_l^2} \sum_{k_1=1}^{M_l} \sum_{k_2=1}^{M_l} \Big(k(\mathbf{f}^l_{\cdot k_1}, \mathbf{f}^l_{\cdot k_2}) + k(\mathbf{s}^l_{\cdot k_1}, \mathbf{s}^l_{\cdot k_2}) - 2k(\mathbf{f}^l_{\cdot k_1}, \mathbf{s}^l_{\cdot k_2}) \Big) \\
&= \frac{1}{4N_l^2} \text{MMD}^2[\mathcal{F}^l, \mathcal{S}^l]
\end{aligned}$$

where $k(x, y) = (x^T y)^2$ is the second-order polynomial kernel.

Therefore, for each layer, the style loss of neural style transfer can be viewed as minimizing the MMD between the stylized image and the style image with a polynomial kernel over the feature space implicitly defined by a trained convolutional neural network. Replacing the polynomial kernel with other kernels (i.e. RBF kernel) gives similar successful results.

Connecting this finding to the other applications mentioned before, one can possibly try to learn the kernel rather than using a pretrained network to define a kernel. It is unclear whether the style transfer ability of CycleGAN [15] is related to MMD and MMD-GAN, and uncovering their connections can be an interesting work.

5 Conclusion

In this survey, we introduced MMD and its applications in machine learning. Note that the applications we covered is far from exhaustive, and MMD is applied in a much wider scope, even if we limit the area to machine learning only. We also want to mention that the particular constraints on MMD witness functions can be loosely connected to spectral roughness penalty, and optimizing MMD with an RBF kernel is connected to infinite-order moment matching. The conclusion is, this survey is but a very short introduction to a very rich area, and we hope this survey can motivate and help the reader interested in MMD to learn more about it.

References

[1] Michael Arbel, Dougal J. Sutherland, Mikołaj Bińkowski, and Arthur Gretton. On gradient regularizers for mmd gans, 2018.

[2] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks, 2017.

[3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[4] Mikołaj Bińkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans, 2018.

[5] Gintare Karolina Dziugaite, Daniel M. Roy, and Zoubin Ghahramani. Training generative neural networks via maximum mean discrepancy optimization, 2015.

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[7] Jackson Gorham and Lester Mackey. Measuring sample quality with stein's method, 2019.

[8] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[9] Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network, 2017.

[10] Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm, 2019.

[11] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, 2017. ISSN 1935-8245. doi: 10.1561/2200000060. URL `http://dx.doi.org/10.1561/2200000060`.

[12] Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, pages 429–443, 1997.

[13] Chris J. Oates, Mark Girolami, and Nicolas Chopin. Control functionals for monte carlo integration, 2016.

[14] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[15] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.