ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200250

Coulomb Autoencoders

Emanuele Sansone[1] and **Hafiz Tiomoko Ali**[2] and **Jiacheng Sun**[3]

Abstract. Learning the true density in high-dimensional feature spaces is a well-known problem in machine learning. In this work, we consider generative autoencoders based on maximum-mean discrepancy (MMD) and provide theoretical insights. In particular, (i) we prove that MMD coupled with Coulomb kernels has optimal convergence properties, which are similar to convex functionals, thus improving the training of autoencoders, and (ii) we provide a probabilistic bound on the generalization performance, highlighting some fundamental conditions to achieve better generalization. We validate the theory on synthetic examples and on the popular dataset of celebrities faces, showing that our model, called Coulomb autoencoders, outperform the state-of-the-art.

1 Introduction

Deep generative models, like generative adversarial networks (GANs) and autoencoder-based models, represent the most promising research directions to learn the underlying density of data. Each of these families have their own limitations. On one hand, generative adversarial networks are difficult to train due to the mini-max nature of the optimization problem. On the other hand, autoencoder-based models, while more stable to train, often produce samples of lower quality compared to GANs. In this work, we attempt to address the issues of generative autoencoders.

Learning the unknown density in autoencoders requires to minimize two terms, namely the error between the input data and their reconstructed version, together with a distance between a prior density and the density induced by the encoder function. Note that by choosing different distances, we obtain different families of autoencoders. For example, when using the Kullback-Leibler divergence (KL), the corresponding models are variational autoencoders (VAEs) [5, 25], while when choosing the maximum-mean discrepancy (MMD), we obtain Wasserstein autoencoders (WAEs) [29]. The main advantage of WAEs over VAEs is that MMD allows using encoders with deterministic outputs, while, by definition, KL requires only encoders with stochastic outputs. In fact, the stochastic encoder in VAEs is driven to produce latent representations that can be similar among different input samples, thus generating conflicts during reconstruction [29], while the deterministic encoder in WAEs is driven to learn latent representations that are different for different input samples. Therefore, MMD should be preferred over KL, when using deterministic encoders. This work focuses on MMD-based autoencoders and provides two theoretical insights. Regarding the first contribution, we study the critical points of MMD coupled with Coulomb kernels and show that all local extrema are global and that the set of saddle points has zero Lebesgue measure. This result is particularly interesting from the optimization perspective, as MMD coupled with Coulomb kernels can be maximized/minimized through local-search algorithms (like gradient descent), without being trapped into local minima or saddle points. In the context of autoencoders, using MMD with Coulomb kernels allows to mitigate the problem of local minima and achieve better generalization performance, as validated through experiments on synthetic and real-world datasets. Regarding the second contribution, we provide a probabilistic bound on the generalization performance for MMD-autoencoders, highlighting the fact that the reconstruction error is crucial to achieve better generalization and that architecture design is the most important aspect to control it.

The rest of the paper is organized as follows. In Section 2, we provide the two theoretical results. In Section 3, we review the literature of recent generative models. Finally we discuss the experiments in Section 4.

2 Formulation and Theoretical Analysis

This section deals with the problem of density estimation. The goal is to estimate the unknown density function $p_{\mathbf{X}}(\mathbf{x})$, whose support is defined by $\Omega_{\mathbf{x}} \subset \mathbb{R}^d$.

We consider two continuous functions $f : \Omega_{\mathbf{x}} \to \Omega_{\mathbf{z}}$ and $g : \Omega_{\mathbf{z}} \to \Omega_{\mathbf{x}}$, where $\Omega_{\mathbf{z}} \subseteq \mathbb{R}^h$ with h equal to the intrinsic dimensionality of $\Omega_{\mathbf{x}}$. Furthermore, we consider that $g(f(\mathbf{x})) = \mathbf{x}$ for every $\mathbf{x} \in \Omega_{\mathbf{x}}$, namely that g is the left inverse for f on domain $\Omega_{\mathbf{x}}$. In this work, f and g are neural networks parameterized by vectors $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$, respectively. f is called the encoding function, taking a random input $\mathbf{x}$ with density $p_{\mathbf{X}}(\mathbf{x})$ and producing a random vector $\mathbf{z}$ with density $q_{\mathbf{Z}}(\mathbf{z})$, while g is the decoding function taking $\mathbf{z}$ as input and producing the random vector $\mathbf{y}$ distributed according to $q_{\mathbf{Y}}(\mathbf{y})$. Note that, $p_{\mathbf{X}}(\mathbf{x}) = q_{\mathbf{Y}}(\mathbf{y})$, since $\mathbf{y} = g(\mathbf{z}) = g(f(\mathbf{x})) = \mathbf{x}$ for every $\mathbf{x} \in \Omega_{\mathbf{x}}$. This is already a density estimator, but it has the drawback that in general $q_{\mathbf{Z}}(\mathbf{z})$ cannot be written in closed form. Now, define $p_{\mathbf{Z}}(\mathbf{z})$ an arbitrary density with support $\Omega_{\mathbf{z}}$, that has a closed form.[4] Our goal is to guarantee that $q_{\mathbf{Z}}(\mathbf{z}) = p_{\mathbf{Z}}(\mathbf{z})$ on the whole support, while maintaining $g(f(\mathbf{x})) = \mathbf{x}$ for every $\mathbf{x} \in \Omega_{\mathbf{x}}$. This allows us to use the decoding function as a generator and produce samples distributed according to $p_{\mathbf{X}}(\mathbf{x})$. Therefore, the problem of density estimation in a high-dimensional feature space is converted into a problem of estimation in a lower dimensional vector space, thus overcoming the curse of dimensionality.

The objective of our minimization problem is defined as follows:

$$\mathcal{L}(f,g) = \int_{\Omega_{\mathbf{x}}} \|\mathbf{x} - g(f(\mathbf{x}))\|^2 p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} + \lambda \int_{\Omega_{\mathbf{z}}} \int_{\Omega_{\mathbf{z}}} \phi(\mathbf{z})\phi(\mathbf{z}')k(\mathbf{z},\mathbf{z}')d\mathbf{z}d\mathbf{z}' \qquad (1)$$

[1] Huawei Noah's Ark Lab, UK , London, email: emanuele.sansone@huawei.com
[2] Huawei Noah's Ark Lab, UK , London
[3] Huawei Noah's Ark Lab, China, Shenzhen

[4] In this work we consider $p_{\mathbf{Z}}(\mathbf{z})$ as a standard multivariate Gaussian density.

where $\phi(\mathbf{z}) = p_{\mathbf{Z}}(\mathbf{z}) - q_{\mathbf{Z}}(\mathbf{z})$, $k(\cdot,\cdot)$ is a kernel function and λ is a positive scalar hyperparameter weighting the two addends. Note that the first term in (1) reaches its global minimum when the encoding and the decoding functions are invertible on support $\Omega_{\mathbf{x}}$, while the second term in (1) is globally optimal when $q_{\mathbf{Z}}(\mathbf{z})$ equals $p_{\mathbf{Z}}(\mathbf{z})$. Therefore, the global minimum of (1) satisfies our initial requirements and the optimal solution corresponds to the case where $q_{\mathbf{Y}}(\mathbf{y}) = p_{\mathbf{X}}(\mathbf{x})$.

2.1 Convergence properties

The integrals in (1) cannot be computed exactly since $p_{\mathbf{X}}(\mathbf{x})$ is unknown and $q_{\mathbf{Z}}(\mathbf{z})$ is not defined explicitly. As a consequence, we use the unbiased estimate of (1) as a surrogate for optimization, namely:

$$\hat{\mathcal{L}}(f,g) = \sum_{\mathbf{x}_i \in \mathcal{D}_{\mathbf{x}}} \frac{\|\mathbf{x}_i - g(f(\mathbf{x}_i))\|^2}{N} + \lambda \Bigg\{ \frac{1}{N(N-1)} \sum_{\substack{\mathbf{z}_i, \mathbf{z}_j \in \mathcal{D}_{\mathbf{z}} \\ j \neq i}} k_{i,j} - \frac{2}{N^2} \sum_{\mathbf{z}_i \in \mathcal{D}_{\mathbf{z}}} \sum_{\mathbf{z}_j \in \mathcal{D}^f_{\mathbf{z}}} k_{i,j} + \frac{1}{N(N-1)} \sum_{\substack{\mathbf{z}_i, \mathbf{z}_j \in \mathcal{D}^f_{\mathbf{z}} \\ j \neq i}} k_{i,j} \Bigg\} \tag{2}$$

where $k_{i,j} \doteq k(\mathbf{z}_i, \mathbf{z}_j)$ and $\mathcal{D}_{\mathbf{x}} = \{\mathbf{x}_i\}_{i=1}^N$, $\mathcal{D}_{\mathbf{z}} = \{\mathbf{z}_i\}_{i=1}^N$ and $\mathcal{D}^f_{\mathbf{z}} = \{f(\mathbf{x}_i)\}_{i=1}^N$ are three finite set of samples drawn from $p_{\mathbf{X}}(\mathbf{x})$, $p_{\mathbf{Z}}(\mathbf{z})$ and $q_{\mathbf{Z}}(\mathbf{z})$, respectively.

Note that the MMD term, corresponding to the last three addends in (2), is not convex in the set of unknowns $\mathcal{D}^f_{\mathbf{z}}$. This means that it is not possible in general to ensure convergence to the global minimum. Nevertheless, we can prove that, for a specific family of kernels, this property can be achieved. In fact,

Theorem 1. *Assume that*

1. $N > h$.
2. $\forall \mathbf{z}_i, \mathbf{z}_j \in \mathcal{D}_{\mathbf{z}}, \mathbf{z}_i \neq \mathbf{z}_j$
3. *The kernel function satisfies the Poisson's equation, namely* $-\nabla^2_{\mathbf{z}} k(\mathbf{z}, \mathbf{z}') = \lambda \delta(\mathbf{z} - \mathbf{z}'), \quad \forall \mathbf{z}, \mathbf{z}' \in \mathbb{R}^h$, *and the kernel can be written in the following form*

$$k(\mathbf{z}, \mathbf{z}') = \begin{cases} -\frac{\lambda}{2\pi} \ln \|\mathbf{z} - \mathbf{z}'\| & h = 2 \\ \frac{\lambda}{\beta \mathcal{S}_h \|\mathbf{z} - \mathbf{z}'\|^\beta} & \beta = h - 2, h > 2 \end{cases} \tag{3}$$

where $\mathcal{S}_h$ is the surface area of a h-dimensional unit ball. We refer to (3) as the family of Coulomb kernels [11, 30].

Then, the MMD term in (2) satisfies the following general properties:

1. *all local extrema are global.*
2. *the set of saddle points have zero Lebesgue measure.*

Furthermore, the set of all global minima is finite and consists of all possible permutations of the elements in $\mathcal{D}_{\mathbf{z}}$. In other words, $\mathcal{D}^f_{\mathbf{z}} = \mathcal{D}_{\mathbf{z}}$.

Proof. Define the MMD term as:

$$\begin{aligned} &\hat{\mathcal{G}}(\{\mathbf{x}_i\}_{i=1}^N, \{\mathbf{z}_i\}_{i=1}^N) \qquad (4) \\ &= \frac{1}{N(N-1)} \sum_{i=1}^N \sum_{j \neq i} k(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{N(N-1)} \sum_{i=1}^N \sum_{j \neq i} k(\mathbf{z}_i, \mathbf{z}_j) \\ &- \frac{2}{N^2} \sum_{i=1}^N \sum_{j=1}^N k(\mathbf{x}_i, \mathbf{z}_j) \end{aligned}$$

By the definition of Poisson's equation, we get the Laplacian of $\hat{\mathcal{G}}$ as

$$\nabla^2_{\mathbf{x}_i} \hat{\mathcal{G}} = -\frac{2}{N(N-1)} \sum_{j \neq i} \delta(\mathbf{x}_i - \mathbf{x}_j) + \frac{2}{N^2} \sum_{j=1}^N \delta(\mathbf{x}_i - \mathbf{z}_j) \tag{5}$$

Thus, $\hat{\mathcal{G}}$ is harmonic except on the set $H = \{\mathbf{x} : \mathbf{x}_i \neq \mathbf{x}_j, \mathbf{x}_i \neq \mathbf{z}_j, \forall i, j = 1, ..., N\}$. By the Maximal Principle of harmonic functions, $\hat{\mathcal{G}}$ has no local extrema and all the extrema are global.
On the other hand, the saddle points of $\hat{\mathcal{G}}$ satisfy

$$\nabla_{\mathbf{x}_i} \hat{\mathcal{G}} = -\frac{1}{N(N-1)} \sum_{j \neq i} \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|^h} + \frac{1}{N^2} \sum_{j=1}^N \frac{\mathbf{x}_i - \mathbf{z}_j}{\|\mathbf{x}_i - \mathbf{z}_j\|^h} = 0 \tag{6}$$

This implies that

$$F(\mathbf{x}_i) \triangleq \sum_{j=1}^N \frac{\mathbf{x}_i - \mathbf{z}_j}{\|\mathbf{x}_i - \mathbf{z}_j\|^h} = \frac{N}{N-1} \sum_{j \neq i} \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|^h} \tag{7}$$

As $\hat{\mathcal{G}}$ is harmonic except on the set H, F is analytic except on H. By the symmetry of the right hand side, we have $\sum_{i=1}^N F(\mathbf{x}_i) = 0$. Define

$$A = \{(\mathbf{x}_1, ..., \mathbf{x}_N) : \mathbf{x}_i \in \mathbb{R}^h \backslash \{\mathbf{z}_i\}_{i=1}^N, \sum_{i=1}^N F(\mathbf{x}_i) = 0\}$$

If $(\mathbf{x}_1, ..., \mathbf{x}_N) \in A$, then $\mathbf{x}_N \in F^{-1}(-\sum_{i=1}^{N-1} F(\mathbf{x}_i))$. By considering that F is analytic and nonconstant and by using the Fubini Theorem we get that the measure of A is

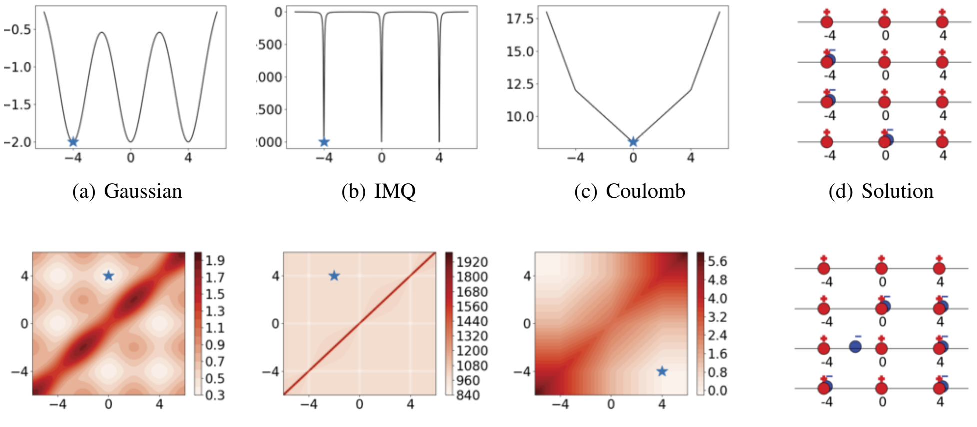
$$\begin{aligned} |A| &= \int_{(\mathbb{R}^h)^N} \chi_A d\mathbf{x}_1 ... d\mathbf{x}_N \\ &= \int_{(\mathbb{R}^h)^{N-1}} (\int_{\mathbb{R}^h} \chi_A d\mathbf{x}_N) d\mathbf{x}_1 ... d\mathbf{x}_{N-1} \\ &= \int_{(\mathbb{R}^h)^{N-1}} |F^{-1}(-\sum_{i=1}^{N-1} F(\mathbf{x}_i))| d\mathbf{x}_1 ... d\mathbf{x}_{N-1} = 0 \end{aligned} \tag{8}$$

where χ_A is the characteristic function, equals 1 on A, otherwise equals 0 and $|\cdot|$ denotes the Lebesgue measure operator. The third equality in (8) holds because we know that for a nonconstant analytic function, its inverse image at a value is of zero measure (w.r.t h-dimensional Lebesgue measure). As the saddle point is a subset of A, so the set of saddle points have zero Lebesgue measure. □

The assumptions of the theorem are quite general in practice. In fact, the requirement $N > h$ is generally valid in applications involving autoencoders. The second assumption is valid with probability 1, as long as the elements in $\mathcal{D}_{\mathbf{z}}$ are drawn independently from p_Z. While the third assumption restricts the choice of kernel function to the family of Coulomb kernels. And this is not a limiting factor as the choice of kernel function is usually arbitrary.

Note that, since the set of saddle points has zero measure, optimization through local search methods can converge to global minima.[5] This is an important characteristic which is similar to convex functionals. Another important remark is that at optimality, the sets

[5] In order to avoid numerical instabilities due to the potential singularities occurring at $\mathbf{z} = \mathbf{z}'$, we add a small constant $\epsilon = 1e - 3$ to the norms in (3), namely using $\|\mathbf{z} - \mathbf{z}'\| + \epsilon$ or $\|\mathbf{z} - \mathbf{z}'\|^\beta + \epsilon$.



(a) Gaussian (b) IMQ (c) Coulomb (d) Solution

(e) Gaussian (f) IMQ (g) Coulomb (h) Solution

Figure 1. Monodimensional cases with single ((a)-(d)) and pair of negative charged particles ((e)-(h))). (a-c) and (e-g) are the plots of the regularizer in (1) over different locations of the negative particles for the Gaussian, the inverse quadratic and the Coulomb kernels, respectively. (d) and (h) show possible minimizers (for the respective kernels).

$\mathcal{D}_{\mathbf{z}}^f$ and $\mathcal{D}_{\mathbf{z}}$ are equal, independently of the sampling from p_Z and of the choice of N. Therefore, MMD with Coulomb kernel forces q_Z to be equal to p_Z.

It is important to mention that Coulomb kernels represent a generalization of the Coulomb potential to any h-dimensional Euclidean space.[6] Therefore, samples from $p_{\mathbf{Z}}$ and $q_{\mathbf{Z}}$ can be regarded as positive and negative charged particles, respectively, while the Coulomb kernels induce some global attraction and repulsion forces between them. As a consequence, the minimization of the regularizer in (2), with respect to the location of the negative charged particles, allows to find a configuration where the two sets of particles balance each other. Based on this interpretation, we can highlight the differences between Coulomb and other kernels from previous work [29] using two simple mono-dimensional cases ($h = 1$). The first example consists of three positive particles, located at $-4, 0$ and 4, and a single negative particle, that is allowed to move freely. In this case, $p_Z(z) = \delta(z+4) + \delta(z) + \delta(z-4)$ and $q_Z(z) = \delta(z - z_1)$, where z_1 represents the variable location of the negative particle. Figure 1(a) and Figure 1(b) represent the plots of the regularizer in (1) evaluated at different z_1 for the Gaussian, the inverse multiquadratic and the Coulomb kernels, respectively. The Gaussian and the inverse multiquadratic kernels introduce new local optima and the negative particle is attracted locally to one of the positive charges without being affected by the remaining ones. On the contrary, the Coulomb kernel has only a single minimum. This minimal configuration is the best one, if one considers that all positive particles exert an attraction force on the negative one. As a result the Coulomb kernel induces **global attraction forces**. The second example consists of the same three positive particles and a pair of free negative charges. In this case, $q_Z(z) = \delta(z - z_1) + \delta(z - z_2)$, where z_1, z_2 are the locations of the two negative particles. Figure 1(d) and Figure 1(e) represent the plots of the regularizer in (1) evaluated at different z_1, z_2 for the Gaussian, the inverse multiquadratic and the Coulomb kernels, respectively. Following the same reasoning of the previous example, we conclude that the Coulomb kernel induces **global repulsion forces**.[7] Note that the fact that the Coulomb kernels induce both global attraction and global repulsion forces between the particles is essential to guarantee that there is a unique configuration of particles which is globally optimal, thus avoiding local minima.

It is worth mentioning that these theoretical results are valid when the optimization is performed on the function space, namely when minimizing with respect to f and g. In reality, the training is performed on the parameter space of neural networks, which may introduce local optima due to their non-convex nature. Solving the problem of local minima in the parameter space of neural networks is a very general problem common to deep learning approaches, which is out of the scope of this work. Our aim is to provide a principled objective function with better convergence properties with respect to existing works.

2.2 Generalization bound

The following theorem provides a probabilistic bound on the estimation error between $\hat{\mathcal{L}}(f,g)$ and $\mathcal{L}(f,g)$ in (2).

Theorem 2. *Given the objective in (2), $h > 2$, $\Omega_{\mathbf{z}}$ a compact set, $\Omega_{\mathbf{x}} = [-M, M]^d$ for positive scalar M, and a symmetric, continuous and positive definite kernel $k : \Omega_{\mathbf{z}} \times \Omega_{\mathbf{z}} \to \mathbb{R}$, where $0 \leq k(\mathbf{z}, \mathbf{z}') \leq K$ for all $\mathbf{z}, \mathbf{z}' \in \Omega_{\mathbf{z}}$ with $K = k(\mathbf{z}, \mathbf{z})$. If the reconstruction error $\|\mathbf{x} - g(f(\mathbf{x}))\|^2$ can be made small $\forall \mathbf{x} \in \Omega_{\mathbf{x}}$, such that it can be bounded by a small value ξ.*

Then, for any $s, u, v, t > 0$

$$Pr\Big\{|\hat{\mathcal{L}} - \mathcal{L}| > t + \lambda(s+u+v)\Big\} \leq 2\exp\left\{-\frac{2Nt^2}{\xi^2}\right\} + 2\exp\left\{-\frac{2\lfloor N/2\rfloor s^2}{K^2}\right\} + 2\exp\left\{-\frac{2\lfloor N/2\rfloor u^2}{K^2}\right\} + 2\exp\left\{-\frac{2Nv^2}{K^2}\right\}$$

Proof. In order to prove the theorem, we first derive the statistical bounds for the reconstruction and the MMD terms separately, and then combine them to obtain the final bound.

[6] In order to see this, consider that for $h = 3$ the kernel function in (3) is proportional to the Coulomb potential.

[7] In this case, there are a pair of minima, corresponding to the permutation of a single configuration.

Consider the reconstruction error term and define $\xi_{\mathbf{x}} \doteq \|\mathbf{x}-g(f(\mathbf{x}))\|^2$. Note that $\Omega_{\mathbf{x}} = [-M,M]^d$ and therefore $\xi_{\mathbf{x}}$ is bounded in the interval $[0,4M^2d]$. By considering $\xi_{\mathbf{x}}$ a random variable, we can apply the Hoeffding's inequality (see Theorem 2 in [32]) to obtain the following statistical bound:

$$\mathrm{P}_0 \doteq \Pr\Bigg\{\Bigg|\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}}\xi_{\mathbf{x}} - \int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x}\Bigg| \geq 2\exp\left\{-\frac{2Nt^2}{\xi^2}\right\} \tag{9}$$

where t is an arbitrary small positive constant.

We can then proceed to find the bound for the other terms in (2). In particular, using the one-sample and two sample U statistics in [32] (see pag. 25), we obtain the following bounds:

$$\begin{aligned}
\mathrm{P}_1 \doteq &\Pr\Bigg\{\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(p_{\mathbf{Z}},p_{\mathbf{Z}})\Bigg| \\
&\geq s\Bigg\} \leq 2\exp\left\{\frac{-2\lfloor N/2\rfloor s^2}{K^2}\right\} \\
\mathrm{P}_2 \doteq &\Pr\Bigg\{\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(q_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \\
&\geq u\Bigg\} \leq 2\exp\left\{\frac{-2\lfloor N/2\rfloor u^2}{K^2}\right\} \\
\mathrm{P}_3 \doteq &\Pr\Bigg\{\Bigg| - \frac{2}{N^2}\sum_{\mathbf{z}_i\in\mathcal{D}_{\mathbf{z}}}\sum_{\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}} k_{i,j} + 2D(p_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \\
&\geq v\Bigg\} \leq 2\exp\left\{\frac{-2Nv^2}{K^2}\right\}
\end{aligned} \tag{10}$$

where $D(p_{\mathbf{Z}},q_{\mathbf{Z}}) \doteq \int_{\Omega_{\mathbf{z}}}\int_{\Omega_{\mathbf{z}}} p_{\mathbf{Z}}(\mathbf{z})q_{\mathbf{Z}}(\mathbf{z}')k(\mathbf{z},\mathbf{z}')d\mathbf{z}d\mathbf{z}'$. Then, we can get the following lower bound:

$$\begin{aligned}
\sum_{i=0}^{3}\mathrm{P}_i \geq &\Pr\Bigg\{\Bigg|\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}}\xi_{\mathbf{x}} - \int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x}\Bigg| \geq t \quad \cup \\
&\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(p_{\mathbf{Z}},p_{\mathbf{Z}})\Bigg| \geq s \quad \cup \\
&\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(q_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \geq u \quad \cup \\
&\Bigg| - \frac{2}{N^2}\sum_{\mathbf{z}_i\in\mathcal{D}_{\mathbf{z}}}\sum_{\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}} k_{i,j} + 2D(p_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \geq v\Bigg\} \\
= &\Pr\Bigg\{\Bigg|\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}}\xi_{\mathbf{x}} - \int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x}\Bigg| \geq t \quad \cup \\
&\lambda\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(p_{\mathbf{Z}},p_{\mathbf{Z}})\Bigg| \geq \lambda s \quad \cup \\
&\lambda\Bigg|\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}\\ j\neq i}} k_{i,j} - D(q_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \geq \lambda u \quad \cup \\
&\lambda\Bigg| - \frac{2}{N^2}\sum_{\mathbf{z}_i\in\mathcal{D}_{\mathbf{z}}}\sum_{\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}} k_{i,j} + 2D(p_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \geq \lambda v\Bigg\} \\
\geq &\Pr\Bigg\{\Bigg|\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}}\xi_{\mathbf{x}} - \int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x} + \lambda\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}_{\mathbf{z}}\\ j\neq i}} k_{i,j} \\
&- \lambda D(p_{\mathbf{Z}},p_{\mathbf{Z}}) + \frac{1}{N(N-1)}\lambda\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}\\ j\neq i}} k_{i,j} \\
&- \lambda D(q_{\mathbf{Z}},q_{\mathbf{Z}}) - \frac{2}{N^2}\lambda\sum_{\mathbf{z}_i\in\mathcal{D}_{\mathbf{z}}}\sum_{\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}} k_{i,j} \\
&+ 2\lambda D(p_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg| \geq t + \lambda(s+u+v)\Bigg\} \\
\geq &\Pr\Bigg\{\Bigg|\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{D}_{\mathbf{x}}}\xi_{\mathbf{x}} + \lambda\Bigg[\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}_{\mathbf{z}}\\ j\neq i}} k_{i,j} \\
&\frac{1}{N(N-1)}\sum_{\substack{\mathbf{z}_i,\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}\\ j\neq i}} k_{i,j} - \frac{2}{N^2}\sum_{\mathbf{z}_i\in\mathcal{D}_{\mathbf{z}}}\sum_{\mathbf{z}_j\in\mathcal{D}^f_{\mathbf{z}}} k_{i,j}\Bigg] \\
&-\int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x} - \lambda\Bigg[D(p_{\mathbf{Z}},p_{\mathbf{Z}}) + D(q_{\mathbf{Z}},q_{\mathbf{Z}}) + \\
&- 2D(p_{\mathbf{Z}},q_{\mathbf{Z}})\Bigg]\Bigg| \geq t + \lambda(s+u+v)\Bigg\} \\
= &\Pr\Bigg\{\Bigg|\hat{\mathcal{L}} - \int_{\Omega_{\mathbf{x}}}\xi_{\mathbf{x}}p_{\mathbf{X}}(\mathbf{x})d\mathbf{x} \\
&- \lambda\Bigg[\int_{\Omega_{\mathbf{z}}}\int_{\Omega_{\mathbf{z}}}(p_{\mathbf{Z}}(\mathbf{z}) - q_{\mathbf{Z}}(\mathbf{z}))(p_{\mathbf{Z}}(\mathbf{z}') \\
&- q_{\mathbf{Z}}(\mathbf{z}'))k(\mathbf{z},\mathbf{z}')d\mathbf{z}d\mathbf{z}'\Bigg]\Bigg| \geq t + \lambda(s+u+v)\Bigg\} \\
= &\Pr\Bigg\{\Bigg|\hat{\mathcal{L}} - \mathcal{L}\Bigg| \geq t + \lambda(s+u+v)\Bigg\}
\end{aligned}$$

where the first inequality is obtained by applying the union bound. Finally, by exploiting also the results in (9), (10) we get the desired bound. □

Theorem 2 provides a probabilistic bound on the estimation error between $\hat{\mathcal{L}}(f,g)$ and $\mathcal{L}(f,g)$. The bound consists of four terms which vanish when N is large. It is important to mention that, while the last three terms can be made arbitrarily small, by choosing appropriate values for s,u,v and λ, the first term depends mainly on on the value of ξ, which can be controlled by modifying the capacity of the encoding and the decoding networks. Therefore, **we can improve the generalization performance of the model by controlling the capacity of the networks as long as ξ can be made small**. This is confirmed also in practice, as shown in the experimental section.

3 Related Work

The most promising research directions for implicit generative models are generative adversarial networks (GANs) and autoencoder-based models.

GANs [8] cast the problem of density estimation as a mini-max game between two neural networks, namely a discriminator, that tries to distinguish between true and generated samples, and a generator, that tries to produce samples similar to the true ones, to fool the discriminator. GANs are notoriously difficult to train, usually requiring careful design strategies for network architectures in [3]. Some of the most known issues are (i) the problem of vanishing gradients in [20], which happens when the output of the discriminator is saturated, because true and generated data are perfectly classified, and no more gradient information is provided to the generator, (ii) the problem of mode collapse in [15], which happens when the samples from the generator collapse to a single point corresponding to the maximum output value of the discriminator, and (iii) the problem of instability associated with the failure of convergence, which is due to the intrinsic nature of the mini-max problem. Different line of works [8] [26] [12] [22] [15] [28], have proposed effective solutions to overcome the aforementioned issues with GANs. However, all these strategies have either poor theoretical motivation or they are guaranteed to converge only locally.

Another research direction for GANs consists on using integral probability metrics [2] as optimization objective. In particular, the maximum mean discrepancy [9] can be used to measure the distance between $p_{\mathbf{X}}$ and $q_{\mathbf{Y}}$ and train the generator network. The general problem is formulated in the following way:

$$\inf_{g\in\mathcal{G}} \sup_{f\in\mathcal{F}} \left\{E_{\mathbf{x}\sim p_{\mathbf{X}}}[f(\mathbf{x})] - E_{\mathbf{y}\sim q_{\mathbf{Y}}}[f(\mathbf{y})]\right\}$$

In generative moment matching networks [18, 7] $\mathcal{F}$ is a RKHS, which is induced by the Gaussian kernel. Note that a major limitation of these models is the curse of dimensionality, since the similarity scores associated with the kernel function are directly computed in the sample space, as explained in [24]. The work of [16] introduces an encoding function to represent data in a more compact way and distances are computed in the latent representation, thus solving the problem of dimensionality. [23] propose to extend the maximum mean discrepancy and include also covariance statistics to ensure better stability. The work of [29] generalizes the computation of the distance between the encoded distribution and the prior to other divergences, thus proposing two different solutions: the first one consists of using the Jensen-Shannon divergence, showing also the equivalence to adversarial autoencoders, and the second one consists of using the maximum-mean discrepancy. The choice of the kernel function in this second case is of fundamental importance to ensure the global convergence of gradient-descent algorithms. As we have already shown in previous section, suboptimal choices of the kernel function, like the ones used by the authors, introduce local optima in the function space and therefore do not have the same convergence property of our model. The work by [30] use Coulomb kernels under the GANs' framework. Nevertheless, the computation of distances is performed directly in the sample space, thus being negatively affected by the curse of dimensionality.

There exists other autoencoder-based models that are inspired by the adversarial game of GANs. [4] add an autoencoder network to the original GANs for reconstructing part of the latent code. The identical works of [13] and [31] propose to add an encoding function together with the generator and perform an adversarial game to ensure that the joint density on the input/output of the generator agrees with the joint density of the input/output of the encoder. They prove that the optimal solution is achieved when the generator and the encoder are invertible. In practice, they fail to guarantee the convergence to that solution due to the adversarial nature of the game. [27] extend the previous works by explicitly imposing the invertibility condition. They achieve this by adding a term to the generator objective that computes the reconstruction error on the latent space. Adversarial autoencoders by [1] are similar to these approaches with the only differences that the estimation of the reconstruction error is performed in the sample space, while the adversarial game is performed only in the latent space. It is important to mention that all of these works are based on a mini-max problem, while our method solves a simple minimization problem, which behaves better in terms of training convergence.

Variational autoencoders (VAEs) by [5, 25] represent another family of autoencoder-based models. The framework is based on minimizing the Kullback-Leibler (KL) divergence between the approximate posterior distribution defined by the encoder and the true prior $p_{\mathbf{Z}}$ (which consists of a surrogate for the negative log-likelihood of training data). Practically speaking, the stochastic encoder used in variational autoencoders is driven to produce latent representations that can be similar among different input samples, thus generating conflicts during reconstruction. A deterministic encoder could ideally solve this problem, but unfortunately the KL divergence is not defined for such case. There are several variations for VAEs. For example, the work in [21] proposes to use the adversarial game of GANs to learn better approximate posterior distributions in VAEs. Nevertheless, the method is still based on a mini-max problem. Recently, [6] propose a training strategy based on a cascade of two VAEs to deal with the limitations implied by the KL divergence. In particular, the authors train a first VAE on the training data and then train a second VAE on the learnt latent representations. This second step is fundamental to improve the matching between the posterior density and the prior with respect to what is done in the first stage. Therefore, this solution implicitly considers the mitigation of local minima from the level of architecture design. However, the 2-stage procedure does not prevent local minima induced by the combination of the two addends in the two objectives. To the best of our knowledge, only [19] are aware of the problem of local minima in generative autoencoders. The authors analyze theoretically the behaviour of simple linear VAEs and show that the phenomenon known as posterior collapse[8] is due to the problem of local minima (or equivalently local maxima, when considering to maximize the ELBO).

4 Experiments

We evaluate the performance of our model (CouAE) against the baseline of Variational Autoencoders (VAE) [5, 25] and Wasserstein Autoencoders (WAE) [29]. All experiments are performed on two synthetic datasets, to simulate scenarios with low and high dimensional feature spaces and on a real-world faces' dataset, namely CelebA 64x64.[9]

We distinguish between two sets of experiments. The first set confirms the usefulness of using the MMD coupled with the Coulomb kernel, while the second one aims at validating the generalization error bound in Theorem 2.

4.1 Comparison with other autoencoders

We start by comparing the approaches on a two-dimensional dataset consisting of 25 isotropic Gaussians placed according to a grid (see Figure 2(a)), hereafter called the grid dataset [14]. The training dataset contains 500 samples generated from the true density.

Following the methodology of other works (see for example [14, 30], we choose fully connected Multilayer Perceptrons with two hidden layers (128 neurons each) at both the encoder and the decoder and set $h = 2$. All models are trained for 3.10^6 iterations using Adam optimizer with learning rate 10^{-3}. Models are evaluated qualitatively by visually inspecting generated samples and quantitatively by computing the log-likelihood on test data. To compute the log-likelihood, we first apply kernel density estimation using a Gaussian kernel on 10^4 generated samples[10] and then evaluate the log-likelihood on 10^4

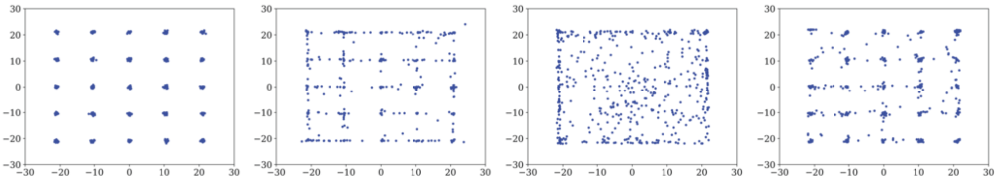
[8] i.e. the posterior over some latent variables matches the prior, with the consequence that those latent variables ignore encoder inputs.

[9] We choose $\lambda = 100$ for all experiments except the ones on the low-dimensional embedding dataset, in which we use $\lambda = 1$ to avoid numerical instabilities.

[10] Bandwidth is selected from a set of 10 values logarithmically spaced in $[10^{-3}, 10^{1.5}]$.

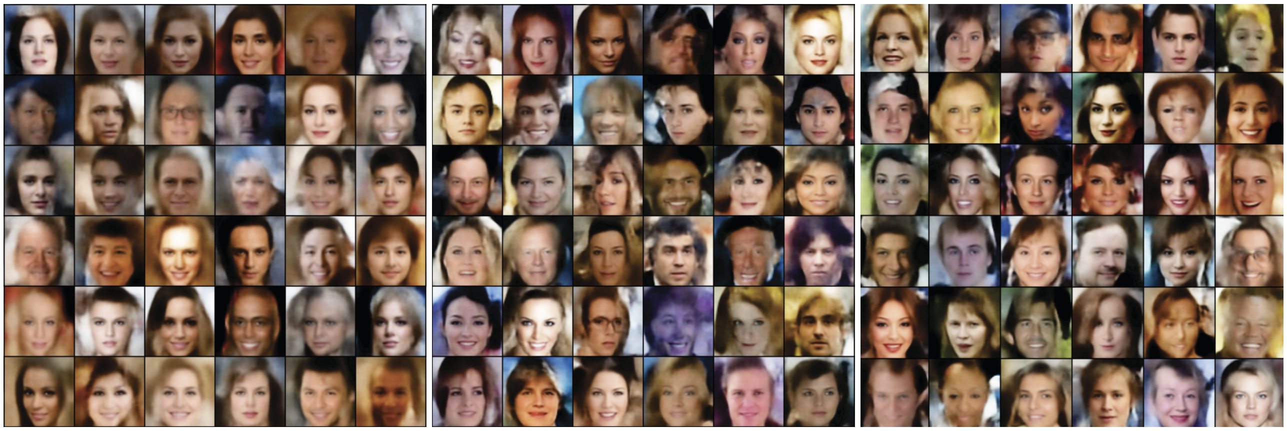
Table 1. Comparison among different autoencoders on different datasets.

Eval. Metric	**Data/Method**	VAE	WAE	CouAE
Test Log-likel.	Grid	-4.4±0.2	-6.4±1.1	**-4.3±0.1**
FID	CelebA	63	55	**47**



(a) True Data (b) VAE (c) WAE (d) CouAE

Figure 2. Generated data from different models on grid dataset.



(a) VAE (b) WAE (c) CouAE

Figure 3. Generated samples on CelebA.

test samples from the true distribution. Results are averaged over 10 repetitions.

Figure 2 shows samples generated by all models, while Table 1 provides quantitative results in terms of test log-likelihood. These experiments highlight the fact that an improper choice of the kernel function may lead to worse performance. In fact, note that WAE does not perform as good as our proposed solution.

For the experiments on CelebA, we follow the settings used in [29]. In particular, we choose a DCGAN architecture [3] and train all models for 10^5 iterations with a learning rate of 0.0005.[11] For the competitors, we run the simulations using the implementation of [3]. Figure 3 shows samples generated by all models, while Table 1 provides quantitative results in terms of test FID [10]. These experiments confirm the findings observed on the grid and the low-dimensional embedding datasets, namely that the choice of using the MMD coupled with the Coulomb kernel provides significant improvements over VAEs and WAEs.

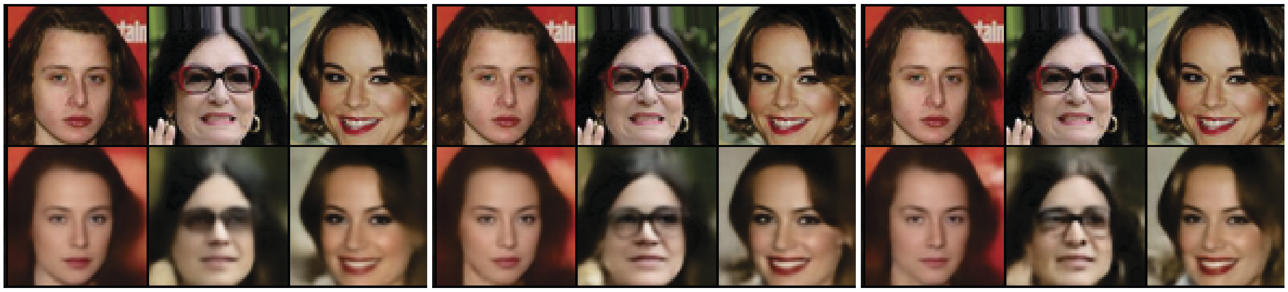
4.2 Validation of generalization bound

To validate the properties of the generalization bound in Theorem 2, we perform experiments on the same datasets used in the previous set of experiments and analyze the performance of CouAE as the capacity of the encoder and the decoder networks changes. In particular, we use the total number of hidden neurons as a proxy to measure the capacity of the model and vary this number according to different scaling factors. Table 2 provides a quantitative analysis of the generalization performance of CouAE. In all cases, we see that an increase of capacity translates into an improvement of the performance. Nevertheless, it is important to mention that there is a limit on the growth of the networks' capacity. As suggested by Theorem 2, the growth is mainly limited by ξ, which could be estimated averaging the reconstruction error on the train and the validation data. In fact, there is no additional benefit to consider larger networks, once ξ is at its minimum, as the bound in Theorem 2 is dominated by the MMD term, which can be improved only by increasing the number of samples. Figure 4 provides a more qualitative analysis on the relation between generalization and reconstruction error. In particular, we visualize

[11] Similarly to the low-dimensional embedding dataset, we choose $\beta = 2$.

Table 2. Experimental validation of generalization bound for CouAE. A scaling factor is applied to the number of neurons in each hidden layer to control the capacity of the encoder and the decoder.

Eval. Metric	**Data/Width factor**	×0.25	×0.5	×1
Test Log-likel.	Grid	-5.8±0.4	-4.8±0.4	-4.3±0.1
Eval. Metric	**Data/Width factor**	×0.25	×0.5	×1
FID	CelebA	53	51	47



(a) ×0.25

(b) ×0.5

(c) ×1

Figure 4. Reconstruction of test images for different width factors. The top and the bottom row of each case contains the original and reconstructed test images.

some reconstructed test images from the model and see that an increase of the network capacity allows to capture more details about the original images.

It is important to mention that there are also other architectural factors, which may affect the estimation of ξ, and which is worth considering in future research. Some examples are the depth of the networks and the use of residual connections (to mitigate the problem of local minima [17]).

5 Conclusions

In this work, we have proposed new theoretical insights on MMD-based autoencoders. In particular, (i) we have proved that MMD coupled with Coulomb kernels has convergence properties similar to convex functionals and shown that these properties have also an impact on the performance of autoencoders, and (ii) we have provided a probabilistic bound on the generalization performance and given principled insights on it.

REFERENCES

[1] A. Makhzani and J. Shlens and N. Jaitly and I. Goodfellow and B. Frey, 'Adversarial Autoencoders', in *International Conference on Learning Representations (ICLR)*, (2013).

[2] A. Müller, 'Integral Probability Metrics and Their Generating Classes of Functions', in *Advances in Applied Probability*, volume 29, pp. 429–443, (1997).

[3] A. Radford and L. Metz and S. Chintala, 'Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks', in *arXiv preprint arXiv:1511.06434*, (2015).

[4] X. Chen, X.Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, 'InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 2172–2180, (2016).

[5] D. P. Kingma and M. Welling, 'Auto-Encoding Variational Bayes', in *International Conference on Learning Representations (ICLR)*, (2014).

[6] B. Dai and D. Wipf, 'Diagnosing and Enhancing VAE Models', in *International Conference on Learning Representations (ICLR)*, (2019).

[7] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, 'Training Generative Neural Networks via Maximum Mean Discrepancy Optimization', in *Uncertainty in Artificial Intelligence (UAI)*, pp. 258–267, (2015).

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, 'Generative Adversarial Nets', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 2672–2680, (2014).

[9] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola, 'A Kernel Method for the Two Sample Problem', in *Max Planck Institute for Biological Cybernetics*, pp. 513–520, (2008).

[10] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, 'GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 6629–6640, (2017).

[11] S. Hochreiter and K. Obermayer, 'Optimal Kernels for Unsupervised Learning', in *IEEE International Joint Conference on Neural Networks (IJCNN 2005)*, pp. 1895–1899, (2005).

[12] I. Durugkar and I. Gemp and S. Mahadevan, 'Generative Multi-Adversarial Networks', in *International Conference on Learning Representations (ICLR)*, (2017).

[13] J. Donahue and P. Krähenbühl and T. Darrell, 'Adversarial Feature Learning', in *International Conference on Learning Representations (ICLR)*, (2017).

[14] J. H. Lim and J. C. Ye, 'Geometric Gan', in *arXiv preprint arXiv:1705.02894*, (2017).

[15] L. Metz and B. Poole and D. Pfau and J. Sohl-Dickstein, 'Unrolled Generative Adversarial Networks', in *International Conference on Learning Representations (ICLR)*, (2017).

[16] C. L. Li, W. C. Chang, Y. Cheng, Y. Yang, and B. Póczos, 'MMD GAN: Towards Deeper Understanding of Moment Matching Network', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 2200–2210, (2017).

[17] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, 'Visualizing the loss landscape of neural nets', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 6389–6399, (2018).

[18] Y. Li, K. Swersky, and R. Zemel, 'Generative Moment Matching Networks', in *International Conference on Machine Learning (ICML)*, pp. 1718–1727, (2015).

[19] J. Lucas, G. Tucker, R. Grosse, and M. Norouzi, 'Understanding Posterior Collapse in Generative Latent Variable Models', in *International Conference on Learning Representations (ICLR)*, (2019).

[20] M. Arjovsky and L. Bottou, 'Towards Principled Methods for Training Generative Adversarial Networks', in *International Conference on Learning Representations (ICLR)*, (2017).

[21] L. Mescheder, S. Nowozin, and A. Geiger, 'Adversarial Variational Bayes: Unifying Variational Autoencoders and Generative Adversarial Networks', in *International Conference on Machine Learning (ICML)*, pp. 2391–2400, (2017).

[22] L. Mescheder, S. Nowozin, and A. Geiger, 'The Numerics of GANs', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 1823–1833, (2017).

[23] Y. Mroueh, T. Sercu, and V. Goel, 'McGan: Mean and Covariance Feature Matching GAN', in *International Conference on Machine Learning (ICML)*, pp. 2527–2535, (2017).

[24] A. Ramdas, S. J. Reddi, B. Poczos, A. Singh, and L. Wasserman, 'On the Decreasing Power of Kernel and Distance Based Nonparametric Hypothesis Tests in High Dimensions', in *AAAI Conference on Artificial Intelligence*, pp. 3571–3577, (2015).

[25] D. J. Rezende, S. Mohamed, and D. Wierstra, 'Stochastic Backpropagation and Approximate Inference in Deep Generative Models', in *International Conference on Machine Learning (ICML)*, pp. 1278–1286, (2014).

[26] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, 'Improved Techniques for Training GANs', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 2234–2242, (2016).

[27] A. Srivastava, L. Valkoz, C. Russell, M. U. Gutmann, and C. Sutton, 'VEEGAN: Reducing Mode Collapse in GANs Using Implicit Variational Learning', in *Advances in Neural Information Processing Systems (NIPS)*, pp. 3310–3320, (2017).

[28] T. Karras and T. Aila and S. Laine and J. Lehtinen, 'Progressive Growing of GANs for Improved Quality, Stability, and Variation', in *International Conference on Learning Representations (ICLR)*, (2018).

[29] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf, 'Wasserstein Auto-Encoders', in *International Conference on Learning Representations (ICLR)*, (2018).

[30] T. Unterthiner, B. Nessler, G. Klambauer, M. Heusel, H. Ramsauer, and S. Hochreiter, 'Coulomb GANs: Provably Optimal Nash Equilibria via Potential Fields', in *International Conference on Learning Representations (ICLR)*, (2018).

[31] V. Dumoulin and I. Belghazi and B. Poole and A. Lamb and M. Arjovsky and O. Mastropietro and A. Courville, 'Adversarially Learned Inference', in *International Conference on Learning Representations (ICLR)*, (2017).

[32] W. Hoeffding, 'Probability Inequalities for Sums of Bounded Random Variables', in *Journal of the American Statistical Association*, pp. 13–30, (1963).