Generative Ratio Matching Networks

Akash Srivastava[*,1,2], Kai Xu[*,3], Michael U. Gutmann[3], Charles Sutton[3,4,5]

∗ equal contributions
[1] MIT-IBM Watson AI Lab [2] IBM Research [3] University of Edinburgh [4] Google AI [5] Alan Turing Institute

Introduction

Adversarial Generative Models(GANs, MMD-GANs)
✅ can generate high-dimensional data such as natural images.
❌ are very difficult to train due to the saddlepoint optimization problem

GRaM is a *stable* learning algorithm for *implicit* deep generative models that does **not** invlove a saddlepoint optimization problem and therefore is easy to train 🎉

Overview

1. Learn a projection function (f_θ)
   - that projects the data (p_x) and the model (q_x) densities into a low-dimensional manifold which,
   - preserves the difference between this pair of densities.
   - We use the ratio ($r(x) = \frac{p_x}{q_x}$) of the two densities as the measure of this difference.
2. Train the model (G_γ) in the low-dimensional manifold
   - using the *Maximum Mean Discrepancy* criterion as it work very well in low dimensional data.

GRaM: Algorithm

1. Learn the manifold projection function $f_\theta(x)$ by minimising the squared difference between the pair of density ratios:

$$
\begin{aligned}
D(\theta) &= \int q_x(x) \left(\frac{p_x(x)}{q_x(x)} - \frac{\bar{p}(f_\theta(x))}{\bar{q}(f_\theta(x))} \right)^2 dx \\
&= C - \mathrm{PD}(\bar{q}, \bar{p})
\end{aligned}
$$

2. Train the generator $G\gamma$ by minimizing the empirical estimator of MMD in the low-dimensional manifold,

$$\min_{\gamma}\left[\frac{1}{N^2}\sum_{i=1}^{N}\sum_{i'=1}^{N}k(f_\theta(x_i),f_\theta(x_{i'}))-\frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}k(f_\theta(x_i),f_\theta(G_\gamma(z_j)))\right.$$
$$\left.+\frac{1}{M^2}\sum_{j=1}^{M}\sum_{j'=1}^{M}k(f_\theta(G_\gamma(z_j)),f_\theta(G_\gamma(z_{j'})))\right]$$

Pearson Divergence Maximisation and Density Ratio Estimation

Monte Carlo approximation of PD,

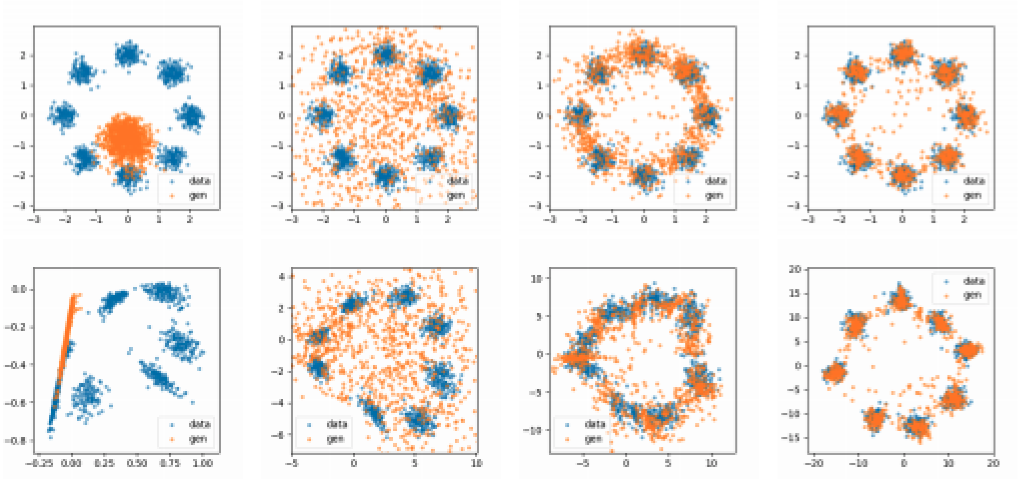
$$\mathrm{PD}(\bar{q}, \bar{p}) \approx \frac{1}{N} \sum_{i=1}^{N} \left(\frac{\bar{p}(f_\theta(x_i))}{\bar{q}(f_\theta(x_i))} \right)^2 - 1$$

where $x_i^q \sim q_x$.

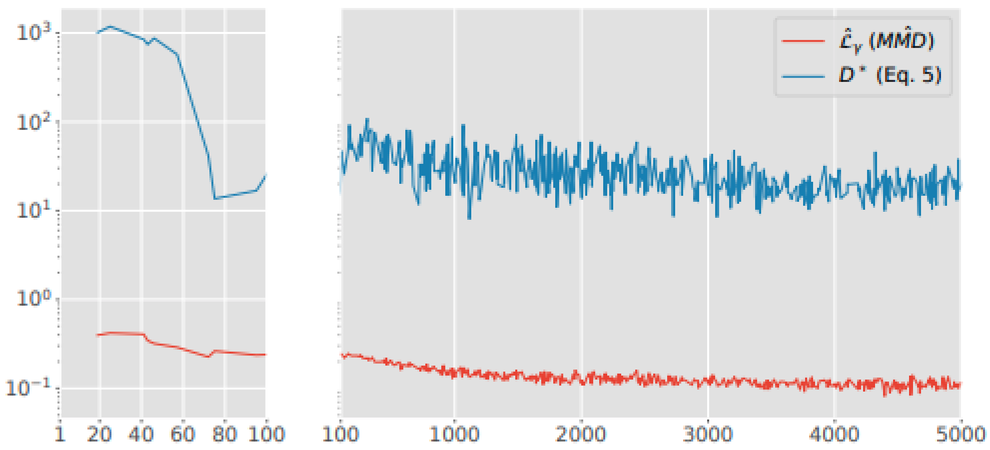
We use a MMD based density ratio estimator (Sugiyama et al., 2012) under the fixed-design setup: $\hat{r}_q = \mathbf{K}_{q,q}^{-1} \mathbf{K}_{q,p} \mathbf{1}$.

- $\mathbf{K}_{q,q}$ and $\mathbf{K}_{q,p}$ are Gram matrices defined by $[\mathbf{K}_{q,q}]_{i,j} = k(f_\theta(x_i^q), f_\theta(x_j^q))$ and $[\mathbf{K}_{q,p}]_{i,j} = k(f_\theta(x_i^q), f_\theta(x_j^p))$.

The Ring dataset: Illustration of the Method and Stability

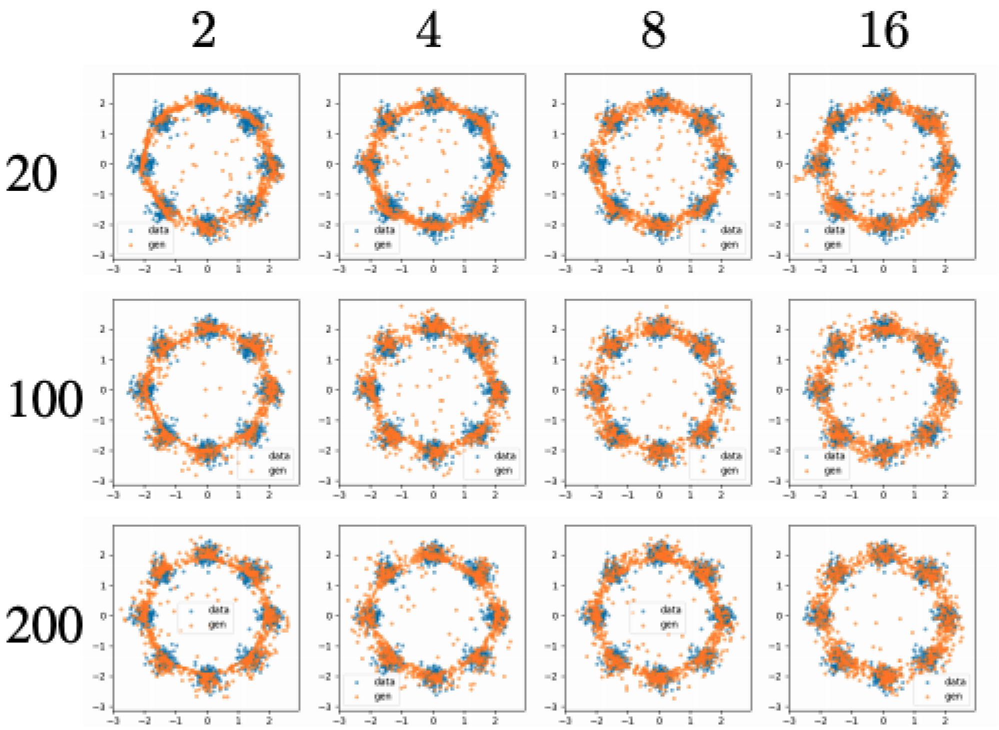


(a) Data and samples in the original (top) and projected space (bottom) during training; four plots are at iteration 10, 100, 1000 and 10, 000 respectively. Notice how the projected space separates $\bar{p}$ and $\bar{q}$.

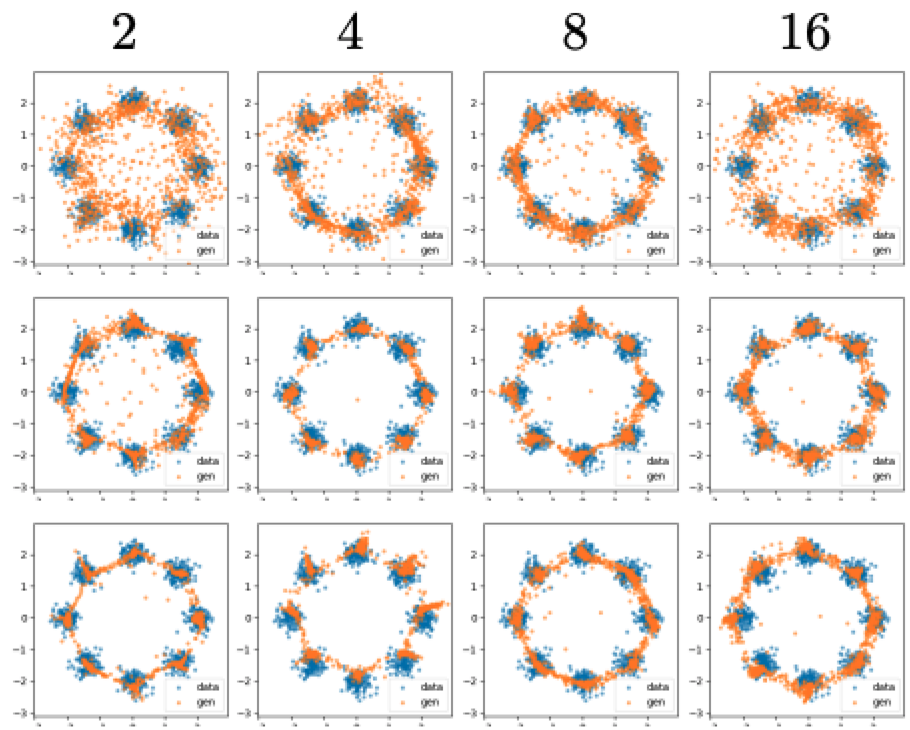


(b) Trace of $\hat{\mathcal{L}}_\gamma$ and D^* (equation (5)) during training. The left plot is for iteration 1 to 100 and the right plot is for 100 to 5,000, with the same y-axes in the log scale.

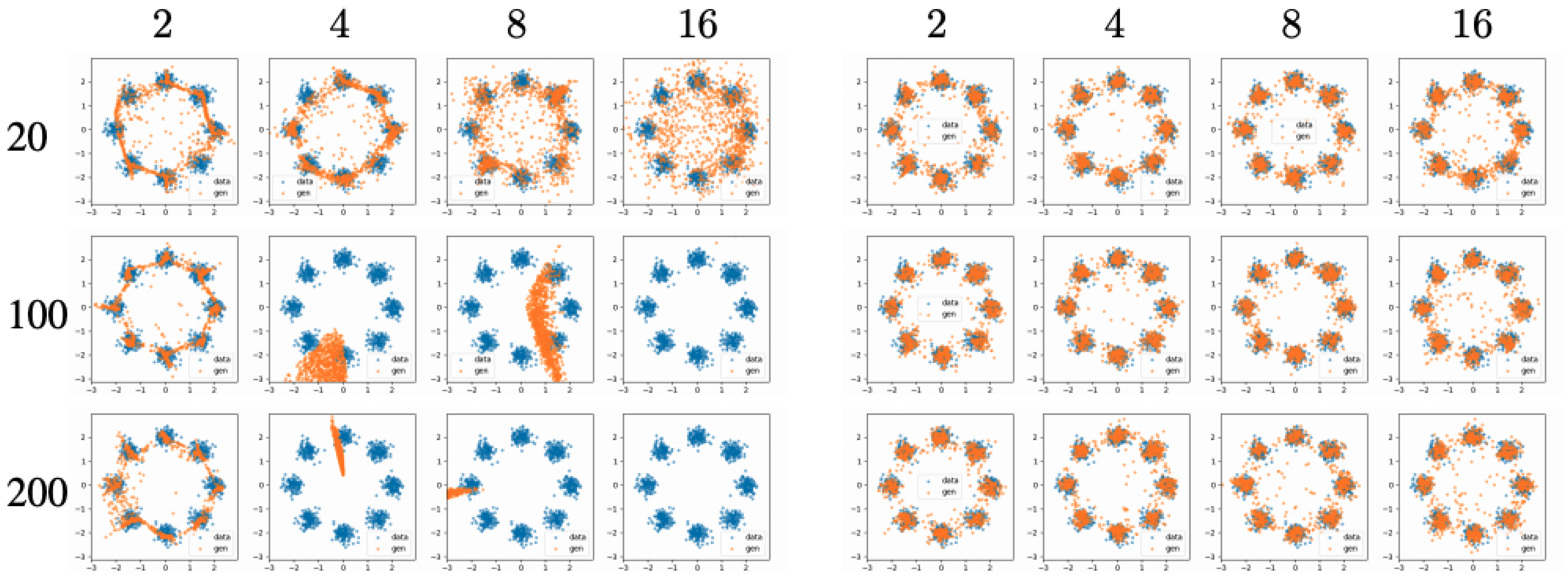
Figure 1: Training results with projected dimension fixed to 2.



(a) MMD-nets



(b) MMD-GANs



(a) GAN

(b) GRAM-net

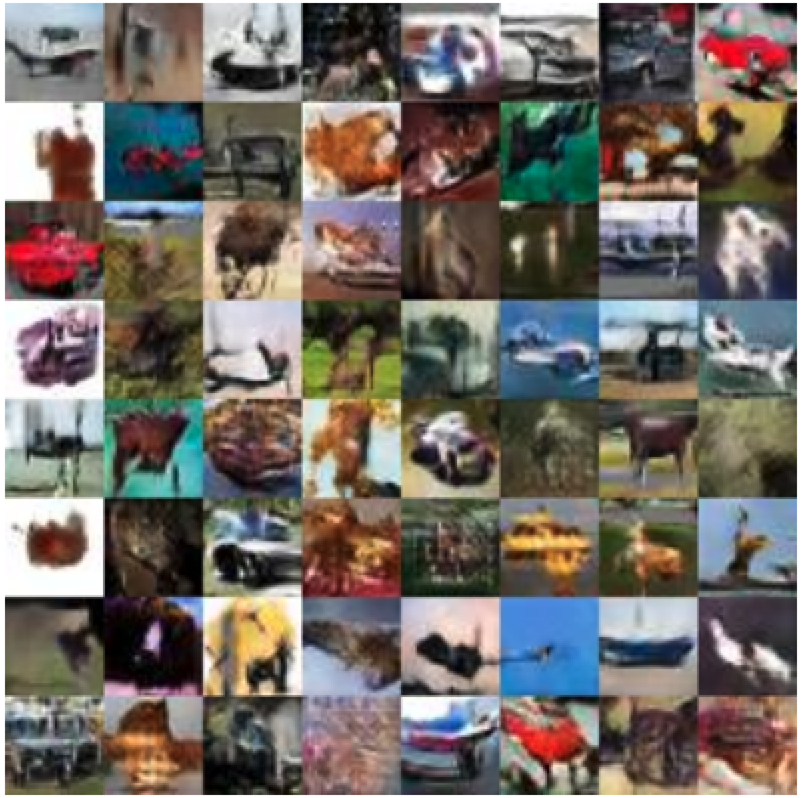
Figure 2: Training after 2,000 epochs by varying noise dimension h and the hidden layer size of critic model. For each model, each row is a different layer size in $[20, 100, 200]$ and each column is a different h in $[2, 4, 8, 16]$. Half of the GAN training diverges while all GRAM training converges.

Quantitative Results: Sample Quality

Table 1: Sample quality (measured by FID; lower is better) of GRAM-nets compared to GANs.

Arch.	**Dataset**	**MMD-GAN**	**GAN**	**GRAM-net**
DCGAN	Cifar10	40.00 ± 0.56	26.82 ± 0.49	$\mathbf{24.85 \pm 0.94}$
Weaker	Cifar10	210.85 ± 8.92	31.64 ± 2.10	$\mathbf{24.82 \pm 0.62}$
DCGAN	CelebA	41.105 ± 1.42	30.97 ± 5.32	$\mathbf{27.04 \pm 4.24}$

Qualitative Results: Random Samples



(a) CIFAR10

(b) CelebA

The End!

Extra slides to follow...

Density Ratio Estimation via (Infinite) Moment Matching

Maximum mean discrepancy

$$\mathrm{MMD}_{\mathcal{F}}(p, q) = \sup_{f \in \mathcal{F}} \left(\mathbb{E}_p[f(x)] - \mathbb{E}_q[f(x)]\right)$$

Gretton et al. (2012) show that it is sufficient to choose $\mathcal{F}$ to be a unit ball in an reproducing kernel Hilbert space $\mathcal{R}$ with a characteristic kernel k.

Using this definition of MMD, the density ratio estimator $r(x)$ can be derived as the solution to

$$\min_{r \in \mathcal{R}} \left\| \int k(x; .)p(x)dx - \int k(x; .)r(x)q(x)dx \right\|_{\mathcal{R}}^2.$$

Generator Training

- The generator $G\gamma$ is trained by minimizing the empirical estimator of MMD,

$$\min_{\gamma}\left[\frac{1}{N^2}\sum_{i=1}^{N}\sum_{i'=1}^{N}k(f_\theta(x_i),f_\theta(x_{i'}))-\frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}k(f_\theta(x_i),f_\theta(G_\gamma(z_j)))\right.$$
$$\left.+\frac{1}{M^2}\sum_{j=1}^{M}\sum_{j'=1}^{M}k(f_\theta(G_\gamma(z_j)),f_\theta(G_\gamma(z_{j'})))\right]$$

with respect to it's parameters γ.