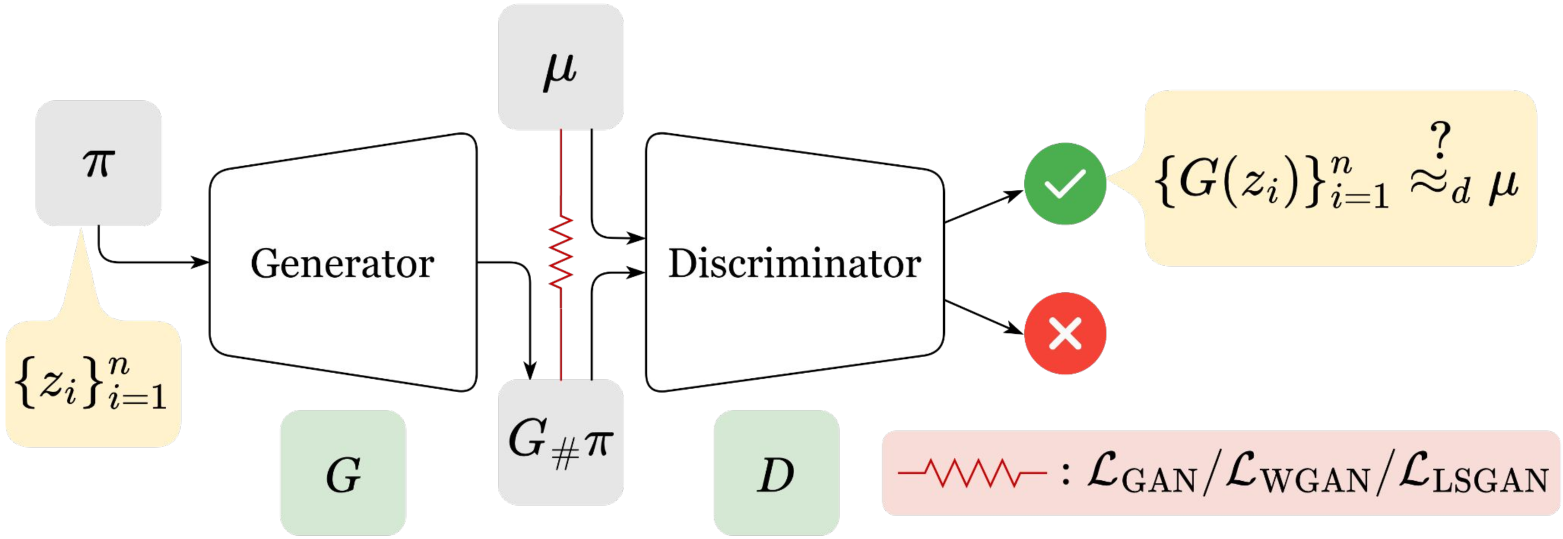
Lost in Translation: GANs' Inability to Generate Simple Probability Distributions

Debanjan Dutta, Anish Chakrabarty and Swagatam Das
Indian Statistical Institute, Kolkata

Push-forward Generation as Statistical Simulation



- Statistical simulation often deals with the problem of generating samples from a target probability distribution given random noise.
- For example, the inverse transform sampling generates $F_X^{-1}(u) =_d X$, given $u \sim U(0,1)$, where F_X^{-1} := the generalized inverse of the cdf of X.

Generative Adversarial Networks (GAN) do something similar.

Theoretical Motivation

- GANs aim to produce pseudo-random replicates from an unknown target distribution $F_X(\cdot)$, often *assumed* to possess corresponding density $f_X : \mathcal{X} \to \mathbb{R}$

- The task boils down to the search for the best estimate of f_X amongst the class of generated laws $g_\theta(\cdot)$, $\theta \in \Theta$.

- GANs tend to perform well when the target distribution is *regular* or has *intrinsic 'patterns'*.

GANs excel at estimating Besov densities (Liang, 2021)

- For images, estimating some of the responsible semantic features in representations accurately results in perceptually acceptable simulations.

Tengyuan Liang. How well generative adversarial networks learn distributions. JMLR, 2021.

Questions

- It seems natural to ask whether there exists a lower limit to the amount of *semantic information* a target law needs to possess to be estimated accurately by GANs.

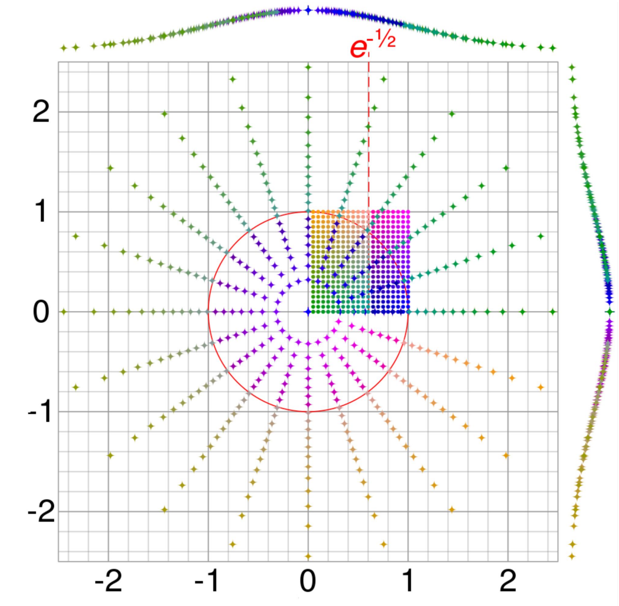
Theoretical guarantees (due to Brenier, Caffarelli, Figalli etc.) only show a handful of possibilities of feasible transport maps.

- We check GANs' ability to perform elementary simulation tasks based on distributions characterized by a small set of parameters.

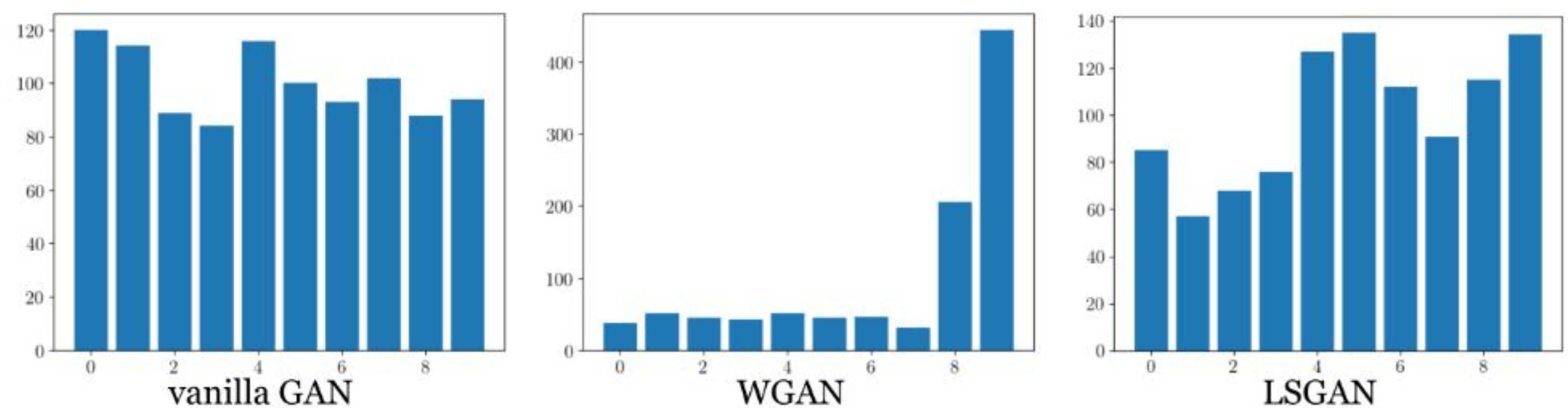
When seen from a model selection viewpoint, the answer seems straightforward.

Experiment: $U(0,1) \xrightarrow{G} N(0,1)$

- Box-Muller transform provides a deterministic pathway, based on radial maps



Can 2-deep vanilla GANs (also, WGAN or LSGAN) equipped with *ReLU*, *leaky-ReLU*, or *tanh* (to span the support) do the same?



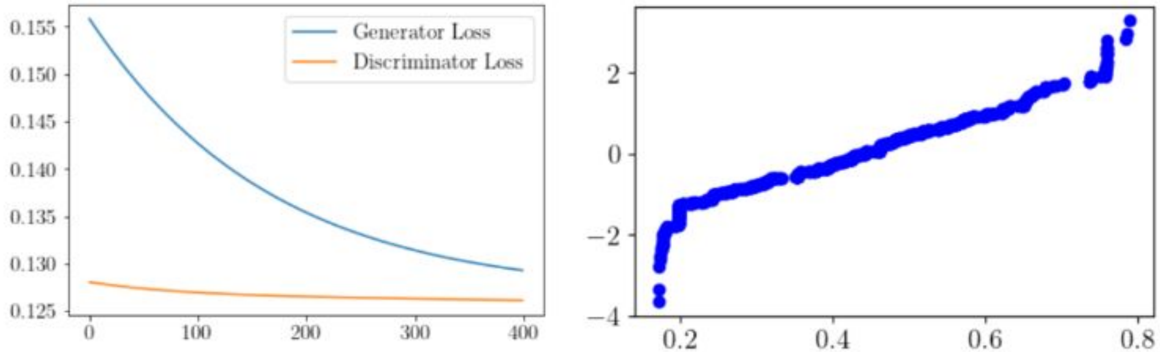
Histograms of generated distribution: Leaky-ReLU-activated 2-deep G and D.

Experiment: $U(0,1) \xrightarrow{G} N(0,1)$

Architecture	KS Test	AD Test
vanilla GAN	×	×
WGAN	×	×
LSGAN	×	×

Table 1: Tests of Normality on $g_{\theta^*}(u)$ at 5% level of significance.

Training instability or overfitting are not to blame either



(a) (b)

(a) Loss landscape and, (b) QQ plot corresponding to $g_{\theta^*}(u)$ using Dropout in LSGAN.

Experiment: $N(0,1) \xrightarrow{G} N(\mu, \sigma^2)$

- Similar trait follows for Gaussian translation and scaling.

Architecture	$N(0,1) \xrightarrow{G} N(-200,1)$	$N(0,1) \xrightarrow{G} N(-200,25)$
	Test of equality of mean	Tests of equality of mean and variance
vanilla GAN	×	×
WGAN	×	×
LSGAN	×	×

Fig 6: Tests of equality of parameters between generated and target distributions.

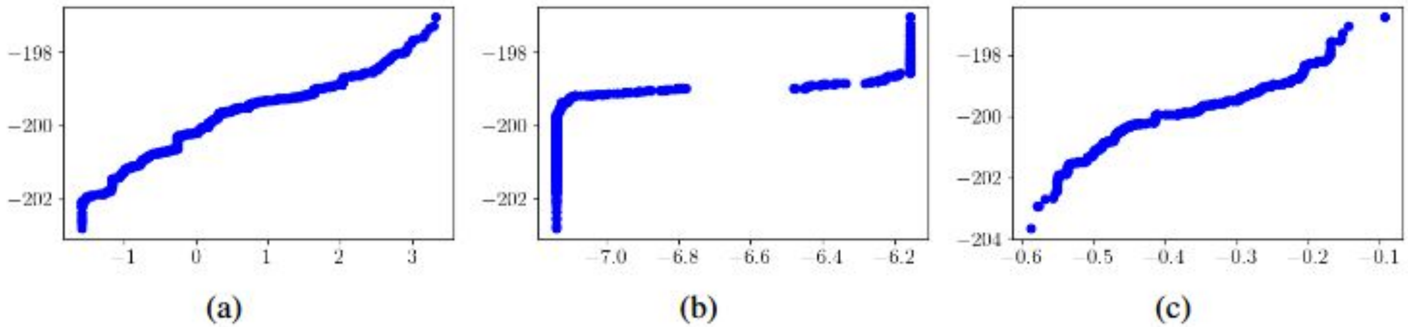


Figure : QQ plots corresponding to generated distributions using (a) vanilla GAN, (b) WGAN, and (c) LSGAN on input $N(0,1)$ to achieve $N(-200,1)$; all deploying leaky-ReLU-activated 2-deep generators with Dropout.

Experiment: $N(0,1) \xrightarrow{G} N(\mu, \sigma^2)$

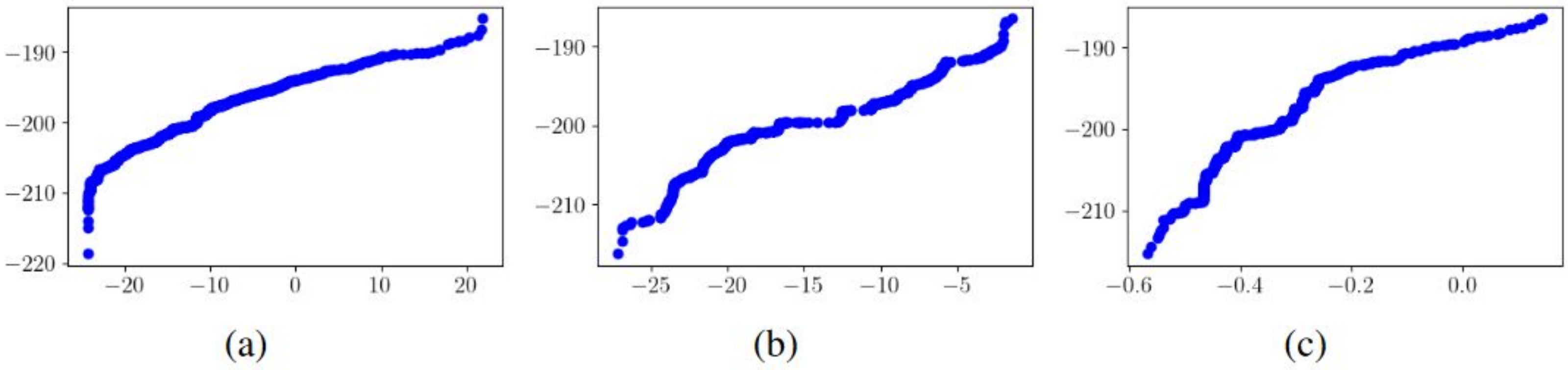


Figure 6: QQ plots corresponding to generated distributions using (a) vanilla GAN (b) WGAN, and (c) LSGAN all deploying Tanh-activated dropped-out 2-deep generators; on input $N(0,1)$ to achieve $N(-200, 25)$.

- Significance is hardly achieved despite fine-tuning *learning rates*, *discriminator's capacity*, *activations* even in *larger sample regimes*.

Check out our poster and paper for details!

Thank You.