Flow-GAN

Combining Maximum Likelihood and Adversarial Learning in Generative Models

Aditya Grover, Manik Dhar, Stefano Ermon

Presented by:

Amy Zhang

Kyle Kastner

Lluís Castrejón

Motivation

GANs can generate pretty pictures...

Progressive Growing of GANs for Improved Quality, Stability, and Variation.
Karras et al. ICLR 2018

Motivation

… but how do you quantify their performance?

?

http://torch.ch/blog/2015/11/13/gan.html

Motivation

In this presentation we'll see:

➔ A GAN model with a tractable likelihood

Motivation

In this presentation we'll see:

- ➔ A GAN model with a tractable likelihood
- ➔ A comparative analysis with models trained with Maximum Likelihood Estimation (MLE)

Outline

- **Quantitative evaluation of generative models**
- Alternatives for computing the data likelihood
- Normalizing Flows and FlowGAN
- Experiments and Analysis
- Conclusions

Evaluating a Generative Model

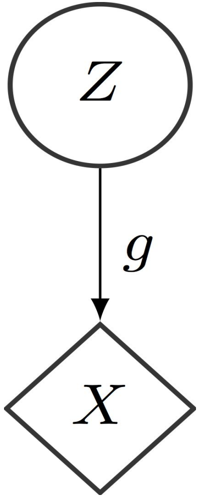
Compute the test data probability:

$$p(\mathbf{x_{test}}; \theta) = \prod_{i=1}^{N} p(x_i; \theta)$$

How likely the test data is under our model, i.e. what is the probability of our model generating the test data

Computing the data probability with latent variables

Marginalize over the latent variables



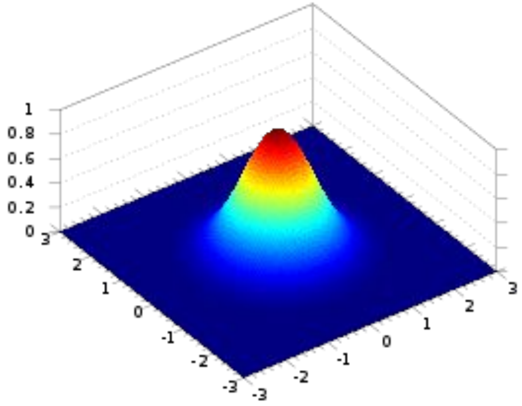
Graphics Credit: Laurent Dinh

$$p(x) = \int p(x|z)p(z)dz$$

Generative Adversarial Networks (GANs)

Prior

$z \sim \mathcal{N}(0, I)$



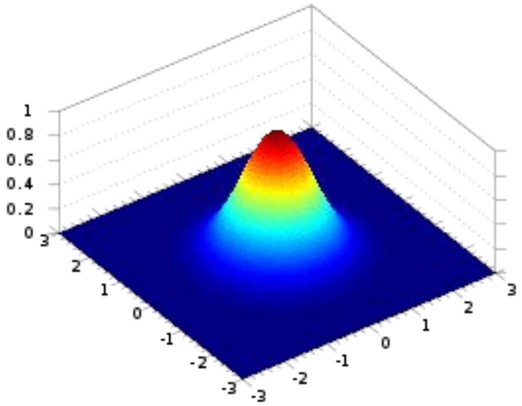
Generator$_\theta$

Sample x

Variational Autoencoders (VAEs)

Prior

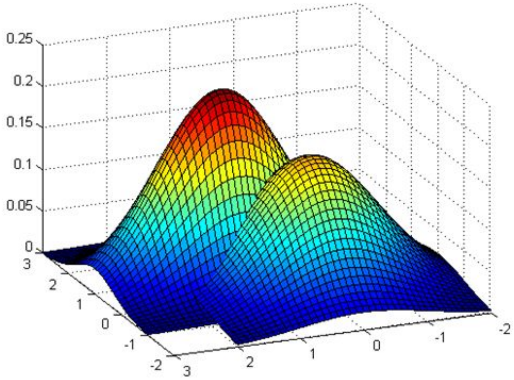
$$z \sim \mathcal{N}(0, I)$$



Decoder_θ

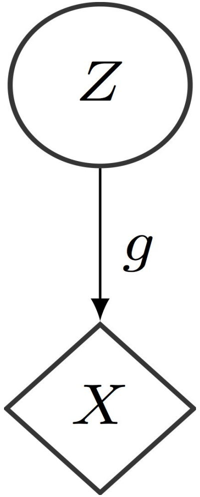
Likelihood

$$p(x|z) = \mathcal{N}(\mu_\theta, \sigma_\theta)$$



Computing the data probability with latent variables

Marginalize over the latent variables



Graphics Credit: Laurent Dinh

$$p(x) = \int p(x|z)p(z)dz$$

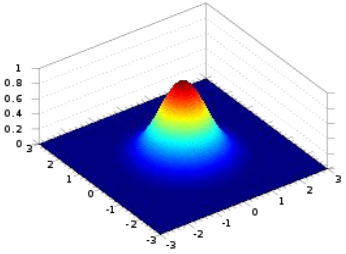
Variational Autoencoders (VAEs)

Monte Carlo estimate of the integral:

$$p(x) = \int p(x|z)p(z)dz = \mathbb{E}_{z\sim p(z)}[p(x|z)] \approx \sum_{i=1}^{N} p(x|z_i)$$

Prior

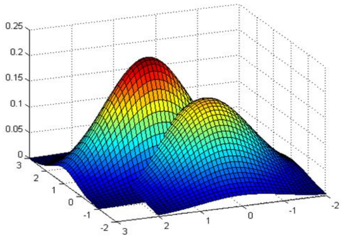
$$z \sim \mathcal{N}(0, I)$$



Decoder

Likelihood

$$p(x|z) = \mathcal{N}(\mu_\theta, \sigma_\theta)$$

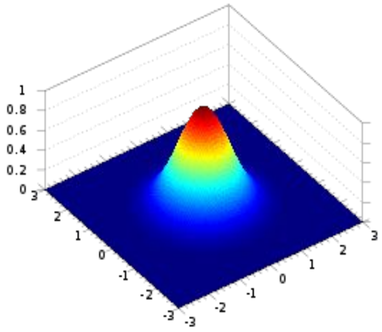


Generative Adversarial Networks (GANs)

We don’t have access to p(x|z), just samples!

Prior

$z \sim \mathcal{N}(0, I)$



Generator

Sample x

Outline

- Quantitative evaluation of generative models
- **Alternatives for computing the data likelihood**
- Normalizing Flows and FlowGAN
- Experiments and Analysis
- Conclusions

Alternatives for computing the data likelihood

- Kernel Density Estimation (KDE)
- Annealed Importance Sampling (AIS)

Alternatives for computing the data likelihood

- Kernel Density Estimation (KDE)
- Annealed Importance Sampling (AIS)
- Reversible Decoders/Normalizing Flows

Alternatives for computing the data likelihood

- Kernel Density Estimation (KDE)
- Annealed Importance Sampling (AIS)
- Reversible Decoders/Normalizing Flows

Proxys for Sample Quality

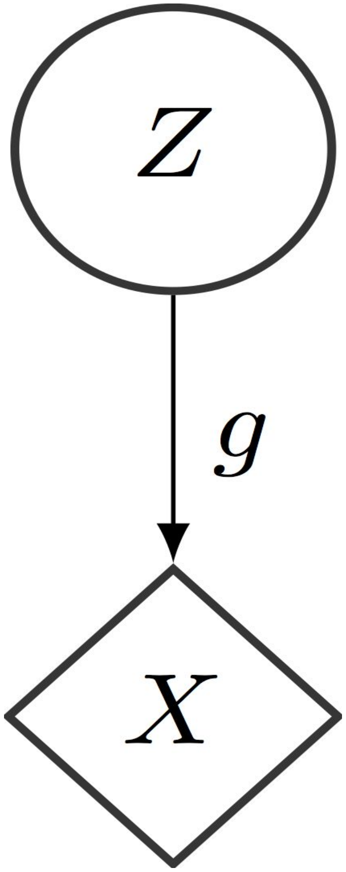
- Inception score
- MODE

Outline

- Quantitative evaluation of generative models
- Alternatives for computing the likelihood
- **Normalizing Flows and FlowGAN**
- Experiments and Analysis
- Conclusions
- Future directions

Thinking in Transformations

- Introduce a latent variable Z
- Choose simple distribution for Z
- Sample ~Z, transform into ~X
  - As in VAE, GAN, many more
- What about data (log) likelihood?

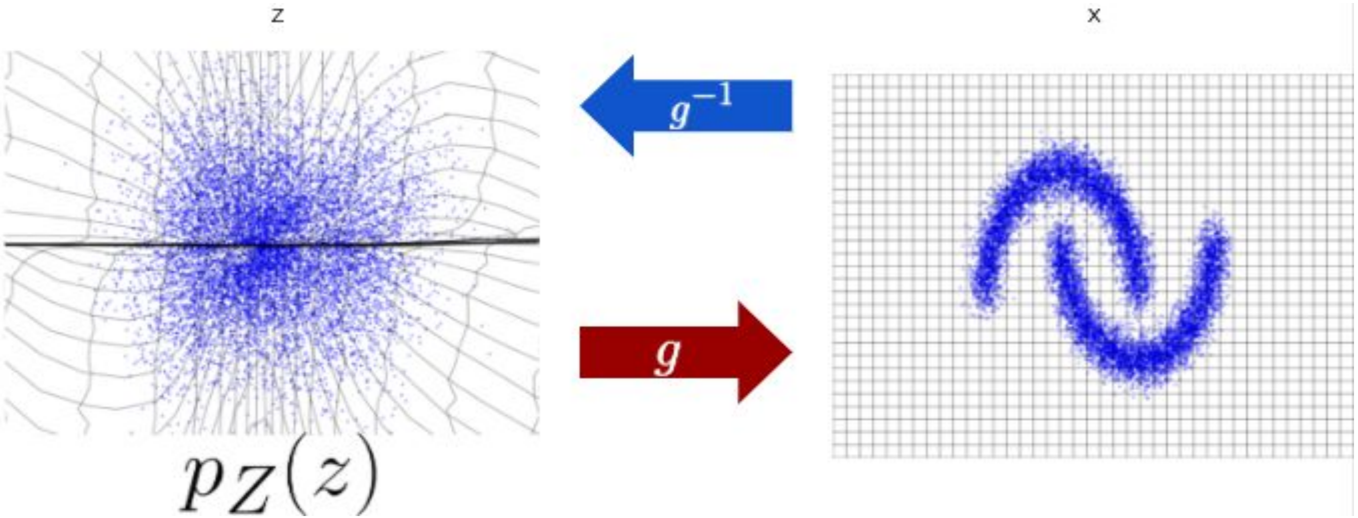


Graphics Credit: Laurent Dinh

Transformation

- What if g is invertible?
- How can we craft invertible g?

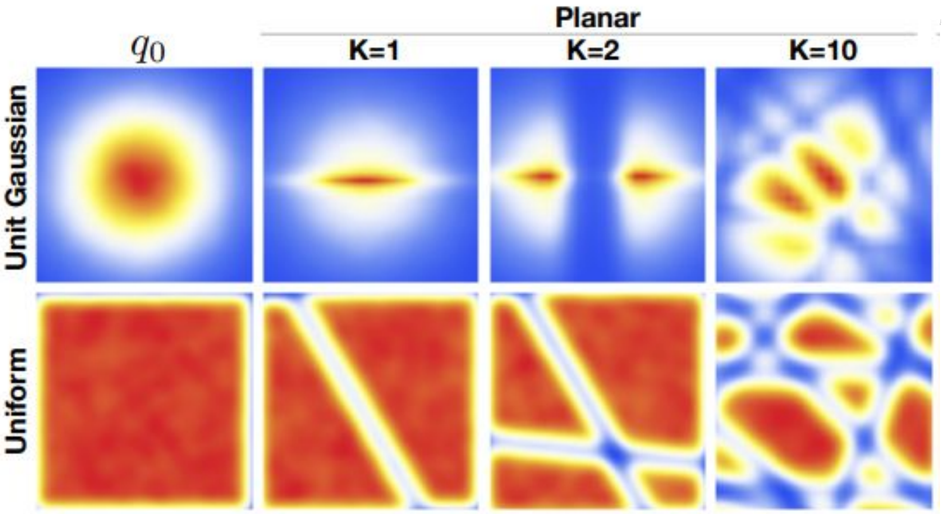
$$x = g(z)$$
$$z = g^{-1}(x)$$



Graphics Credit: Laurent Dinh

Normalizing Flows

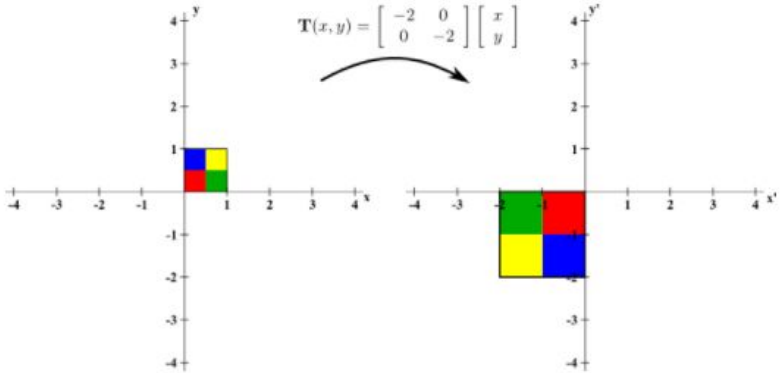
- Density “flows” through invertible transforms
- Still a valid (log) probability: “normalizing flow”



Graphics Credit: Danilo Rezende and Shakir Mohamed

Change of Variables

$$p_X(x) = p_Z(f(x)) \left| \det \left(\frac{\partial f(x)}{\partial x^T} \right) \right|$$



- Requirements: f is bijective, differentiable at x
- Determinants **can** be expensive to compute
- But certain functions have trivial determinants!

Graphics Credit: mathinsight.com

* See Matrix Determinant Lemma for examples
** invertible iff bijective https://math.stackexchange.com/questions/289452/invertible-if-and-only-if-bijective

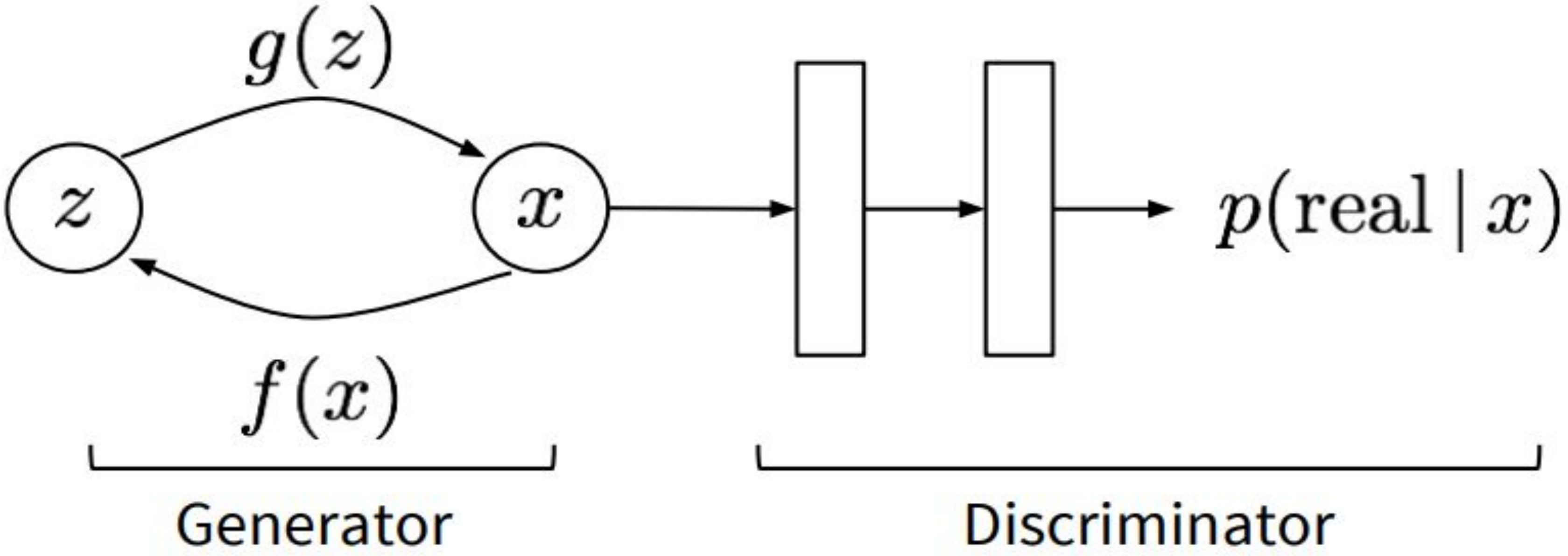
Bringing it Back To FlowGAN

- Use a normalizing flow for the generator
  - Real NVP in this paper
- This means learning can be done using
  - Only the generator (Real NVP, disc. unused)
  - GAN style training, adversarial loss (WGAN)
  - Hybrid combining each loss

Historical - see section 6.1, Yoshua Bengio's PhD thesis (1991) about change of variables

Normalizing Flows: https://math.nyu.edu/faculty/tabak/publications/Tabak-Turner.pdf

Visually



Graphics Credit: David Duvenaud
https://www.cs.toronto.edu/~duvenaud/courses/csc2541/slides/lec3-autoregressive.pdf

Coupling Layer, Real Non-Volume Preserving Transform

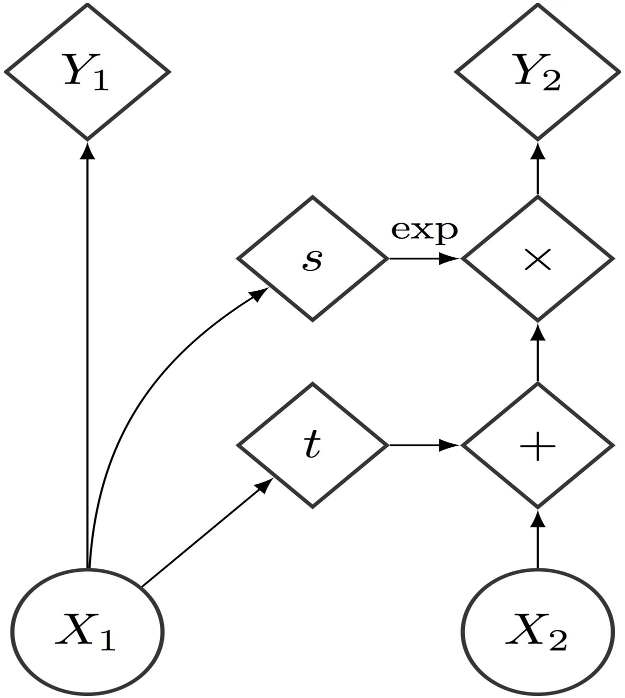
$$b \odot x + (1 - b) \odot \Big(x \odot \exp\left(s(b \odot x)\right) + t(b \odot x) \Big)$$

- has Jacobian determinant

$$\exp\left[\sum_j s\left(x_{1:d}\right)_j\right]$$

- Part unchanged, so chain them

$$det(AB) = det(A)det(B)$$



Graphics Credit: Laurent Dinh

Outline

- Quantitative evaluation of generative models
- Alternatives for computing the likelihood
- Normalizing Flows and FlowGAN
- **Experiments and Analysis**
- Conclusions

Results and Evaluation

Inception:

Run the generated sample through a classifier (Inception model) to get p(y|x), and p(y) is typically assumed uniform. Higher scores are better.

$$\exp\left(\mathbb{E}_{\mathbf{x}\in P_\theta}[KL(p(y|\mathbf{x})\|p(y)]\right)$$

MODE:

Inception score including ground truth distribution of labels

$$\exp\left(\mathbb{E}_{\mathbf{x}\in P_\theta}[KL(p(y|\mathbf{x})\|p^*(y)] - KL(p^*(y)\|p(y))\right)$$

Hybrid Objective:

$$\min_{\theta} \max_{\phi} V(G_\theta, D_\phi) - \lambda \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}} [\log p_\theta(\mathbf{x})]$$

- Analyze three models using Real NVP
- MNIST: Hybrid is best of both worlds
- CIFAR-10: Hybrid is in between MLE and ADV for both metrics

Table 1: Best MODE scores and test negative log-likelihood estimates for Flow-GAN models on MNIST.

Objective	MODE Score	Test NLL (in nats)
MLE	7.42	−3334.56
ADV	9.24	−1604.09
Hybrid ($\lambda = 0.1$)	**9.37**	**−3342.95**

Table 2: Best Inception scores and test negative log-likelihood estimates for Flow-GAN models on CIFAR-10.

Objective	Inception Score	Test NLL (in bits/dim)
MLE	2.92	**3.54**
ADV	**5.76**	8.53
Hybrid ($\lambda = 1$)	3.90	4.21

- Training curves wrt NLL
- NLL goes down (as expected) for MLE
- NLE goes UP for ADV even after WGAN loss stabilizes

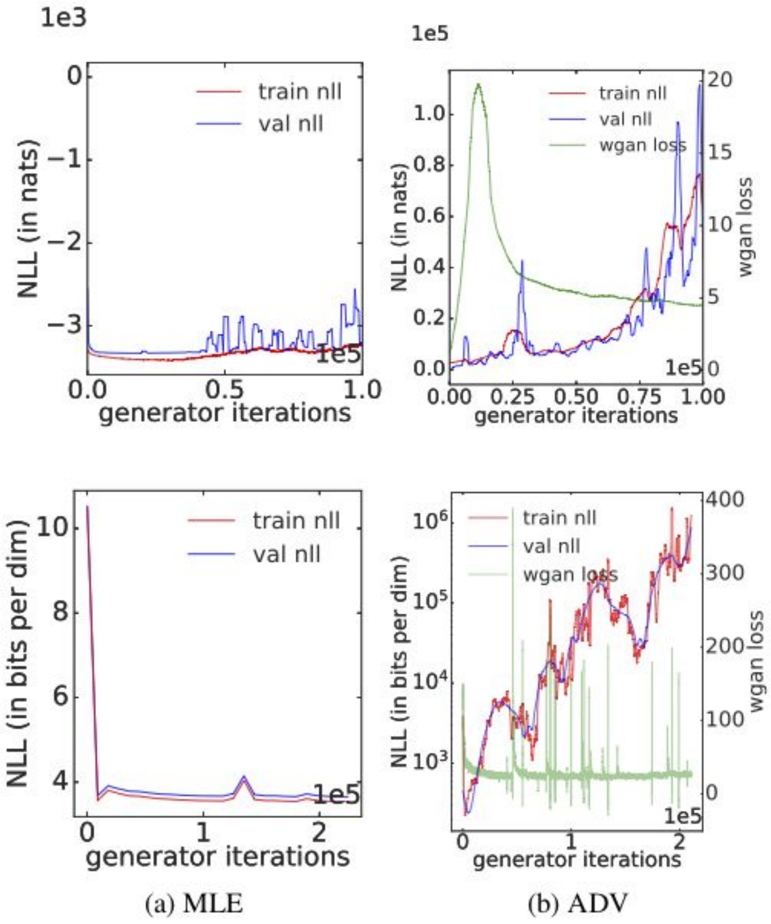


Figure 2: Learning curves for negative log-likelihood (NLL) evaluation on MNIST (**top**, in nats) and CIFAR (**bottom**, in bits/dim). Lower NLLs are better.

Explaining log-likelihood trends: Analyzing the Jacobian

Adversarial methods have ill-conditioned Jacobians, likely due to mode collapse.

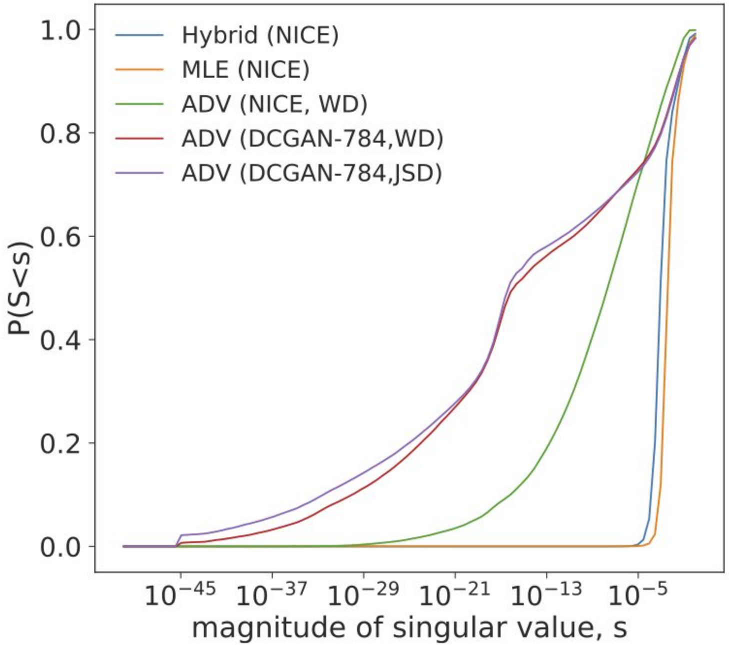


Figure 4: CDF of the singular values magnitudes for the Jacobian of the generator functions trained on MNIST.

True NLL vs. AIS and KDE estimates

AIS and KDE don’t give nll estimates that have the same ranking!

AIS: ADV > Hybrid > MLE

KDE: Hybrid > MLE > ADV

Flow-GAN: MLE > Hybrid > ADV

Table 3: Comparison of inference techniques for negative log-likelihood estimation of Flow-GAN models on MNIST.

Objective	Flow-GAN NLL	AIS	KDE
MLE	-3287.69	-2584.40	-167.10
ADV	26350.30	-2916.10	-3.03
Hybrid	-3121.53	-2703.03	-205.69

Outline

- Quantitative evaluation of generative models
- Alternatives for computing the data likelihood
- Normalizing Flows and FlowGAN
- Experiments and Analysis
- **Conclusions**

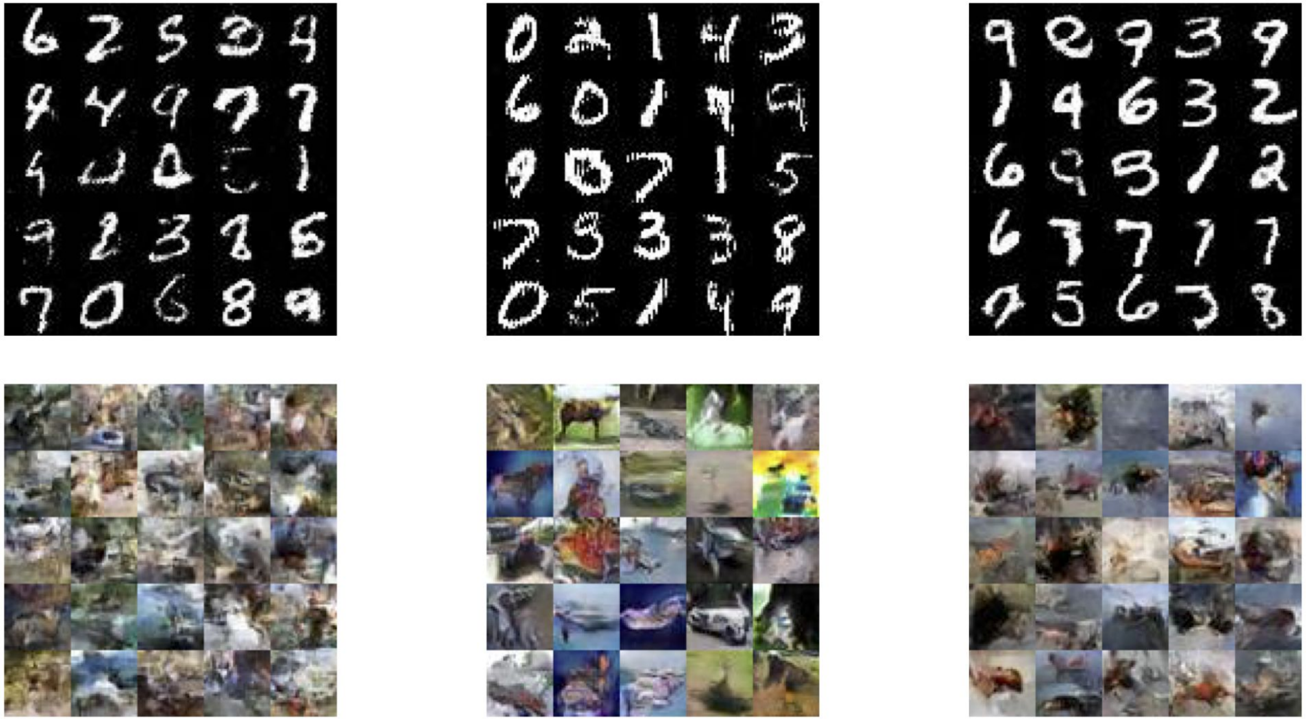
Conclusions

- Regular GANs have intractable data likelihoods
- Using Normalizing Flows we can estimate $p(x|z)$ in a GAN
- FlowGAN: RealNVP (normalizing flows) + GAN
- GANs have high NLL (mode collapse?) but produce better sample quality
- Hybrid model offers a trade-off between MLE and ADV models

References

- Review of determinants
  https://mathinsight.org/determinant_linear_transformation
- A family of non-parametric density estimation algorithms
  https://math.nyu.edu/faculty/tabak/publications/Tabak-Turner.pdf
- Tutorial on Generative Models, Shakir Mohamed
  http://auai.org/uai2017/media/tutorials/shakir.pdf
- NICE https://arxiv.org/abs/1410.8516
- Variational Inference with Normalizing Flows https://arxiv.org/abs/1505.05770
- Density Estimation using Real NVP https://arxiv.org/abs/1605.08803
- DCGAN https://arxiv.org/abs/1511.06434
- Autoencoding beyond pixels https://arxiv.org/abs/1512.09300

Thank You!



(a) MLE | (b) ADV | (c) Hybrid

Figure 1: Samples generated by Flow-GAN models with different objectives for MNIST (**top**) and CIFAR-10 (**bottom**).