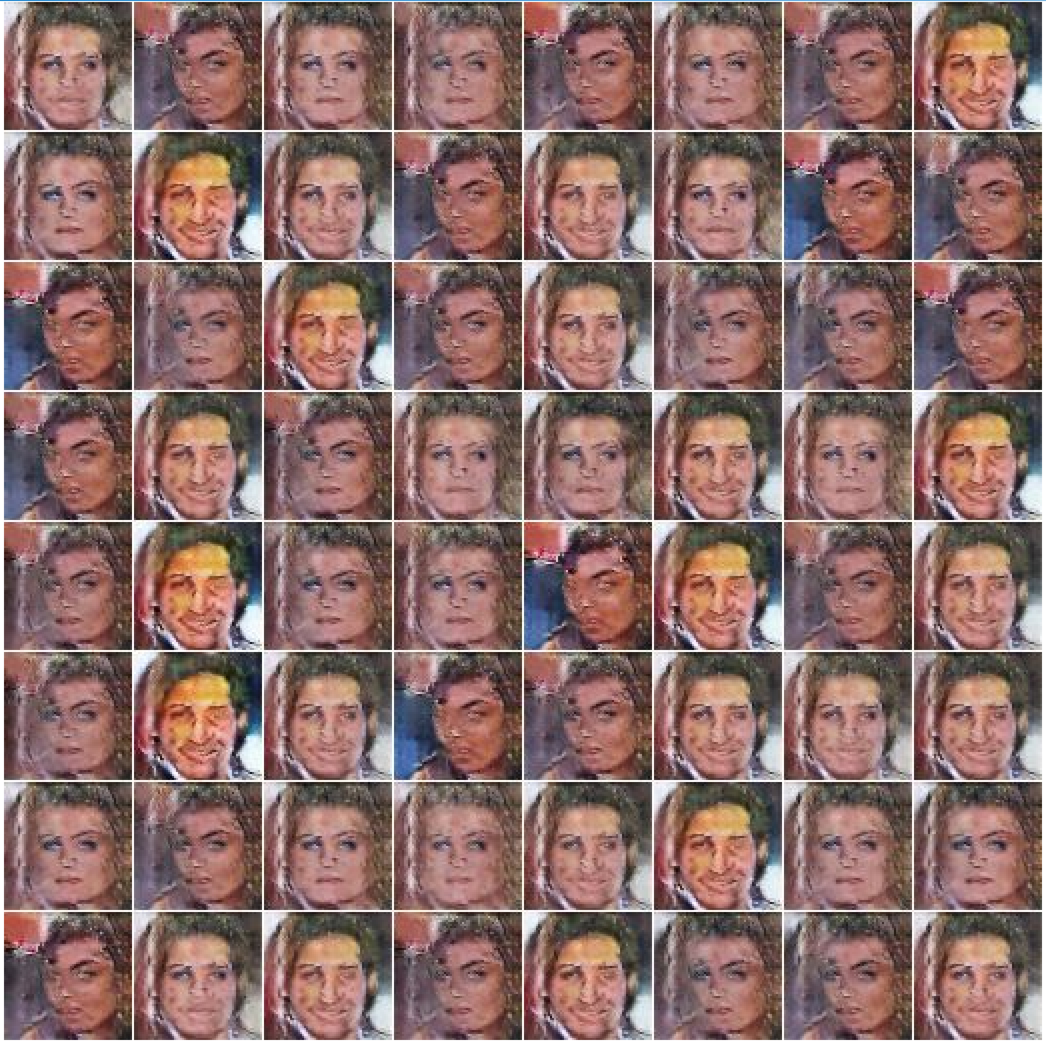
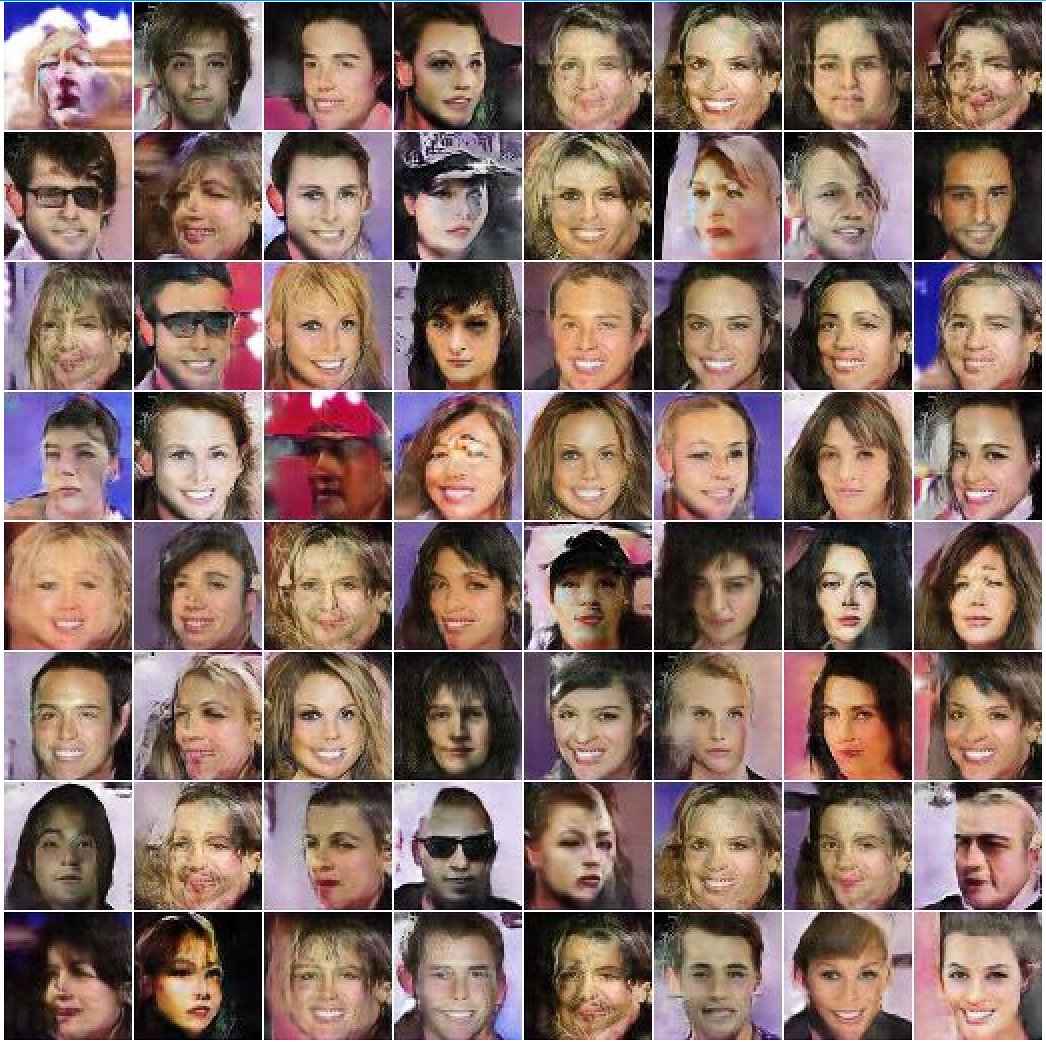
Solving mode collapse with Autoencoder GANs

Mihaela Rosca

DeepMind

Thanks to: Balaji Lakshminarayanan, David Warde-Farley, Shakir Mohamed





Autoencoders

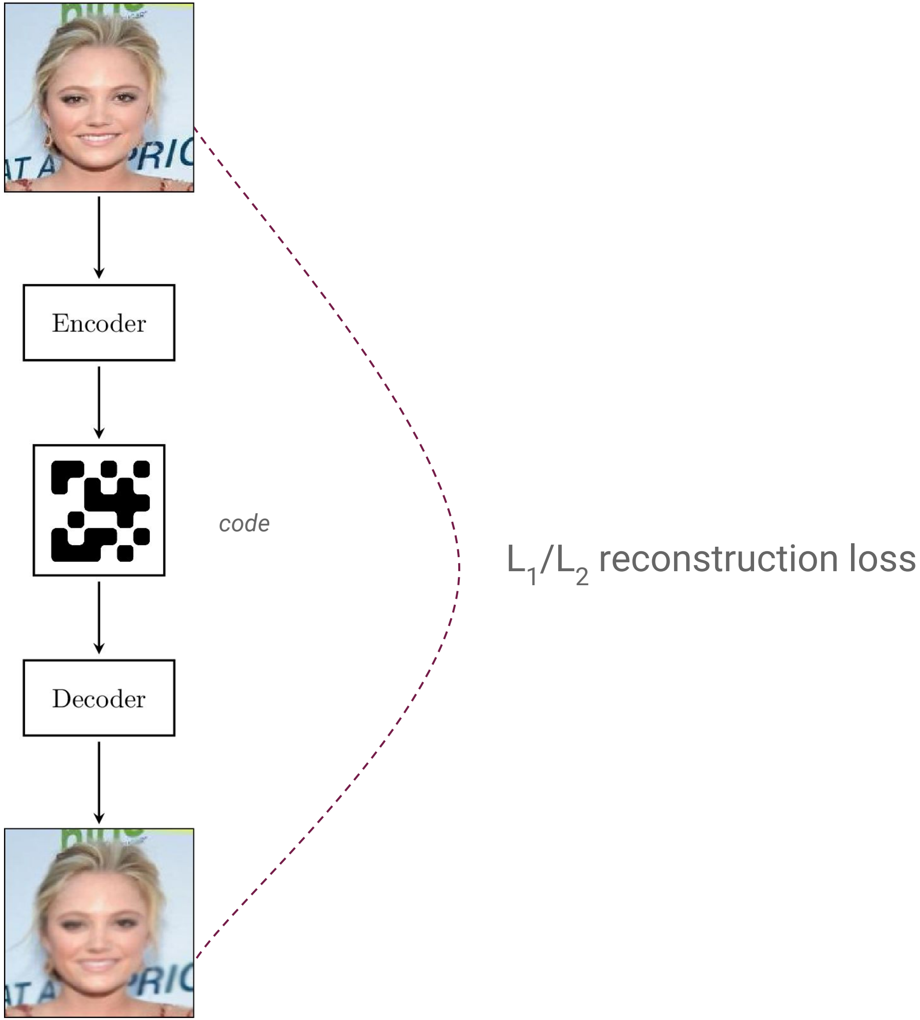
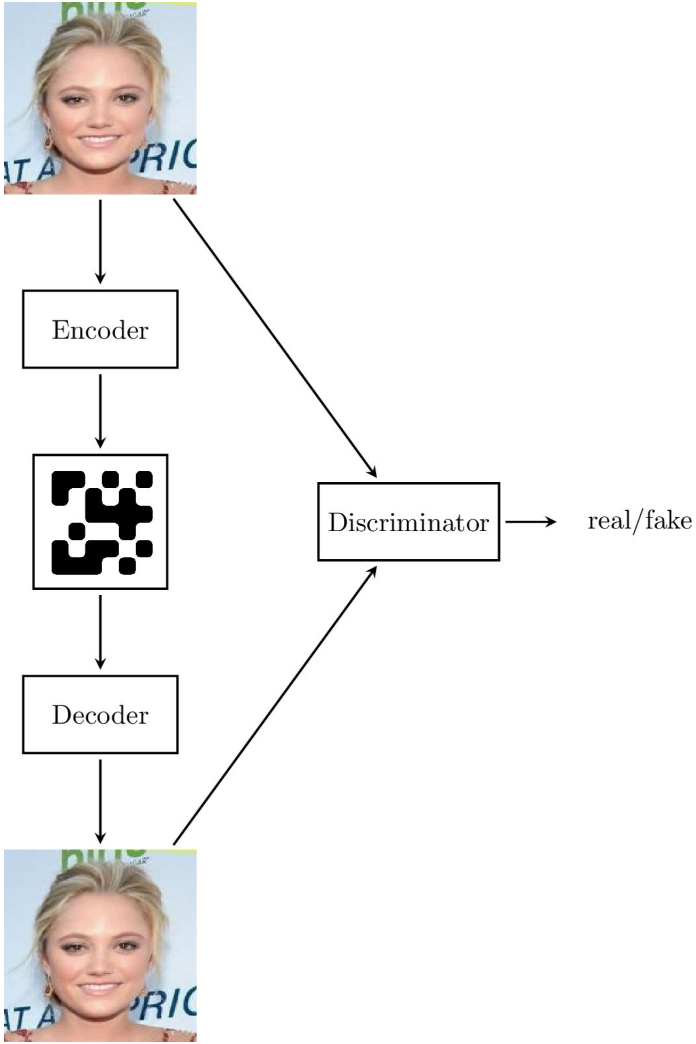


Image Credit: mikicon, the Noun Project

Adversarial autoencoders

Improve reconstruction quality by adding a GAN loss.



Adversarial Autoencoders
A. Makhzani, J. Shlens, N.Jaitly, I. Goodfellow, B. Frey

By construction, autoencoders learn to cover the entire training data.

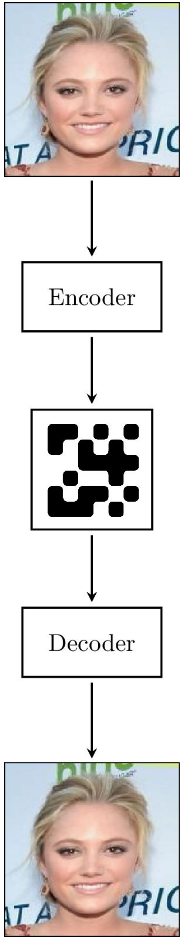
Autoencoder GANs

Combine the reconstruction power of autoencoders with the sampling power of GANs!

How to sample?

Work on the code space
- not data space.

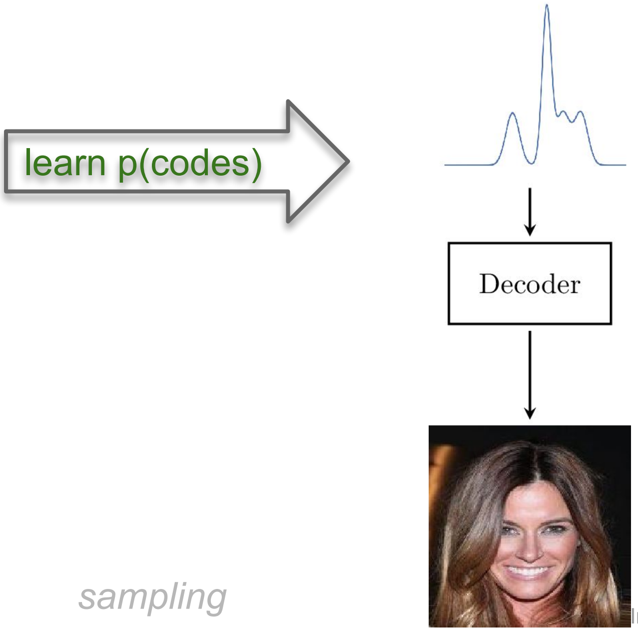
1) Learning the code distribution



reconstructing

Assumption

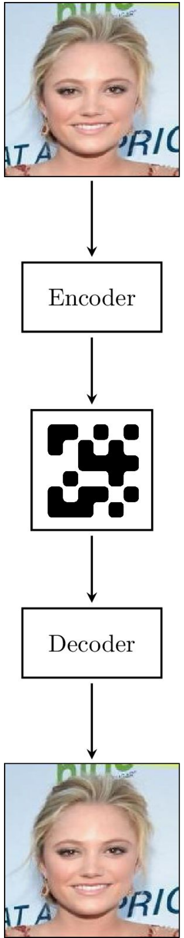
learning the code distribution is simpler than learning data distribution



sampling

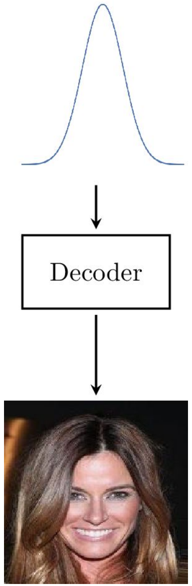
Image Credit: mikicon, the Noun Project

2) Match the code distribution to a desired prior



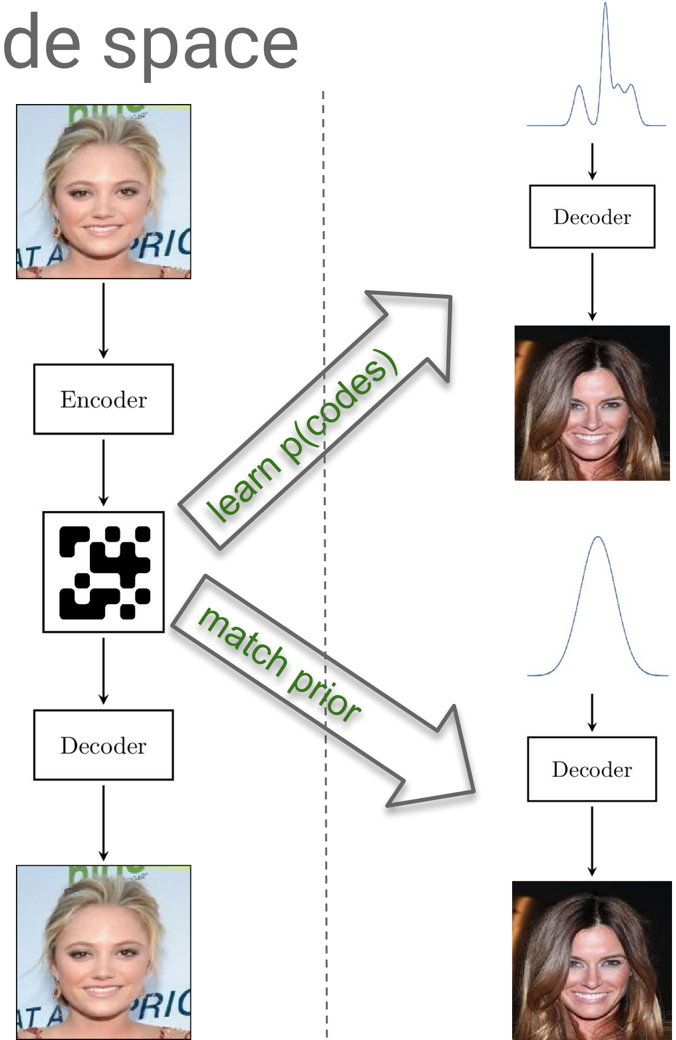
reconstructing

match prior



sampling

Working on the code space

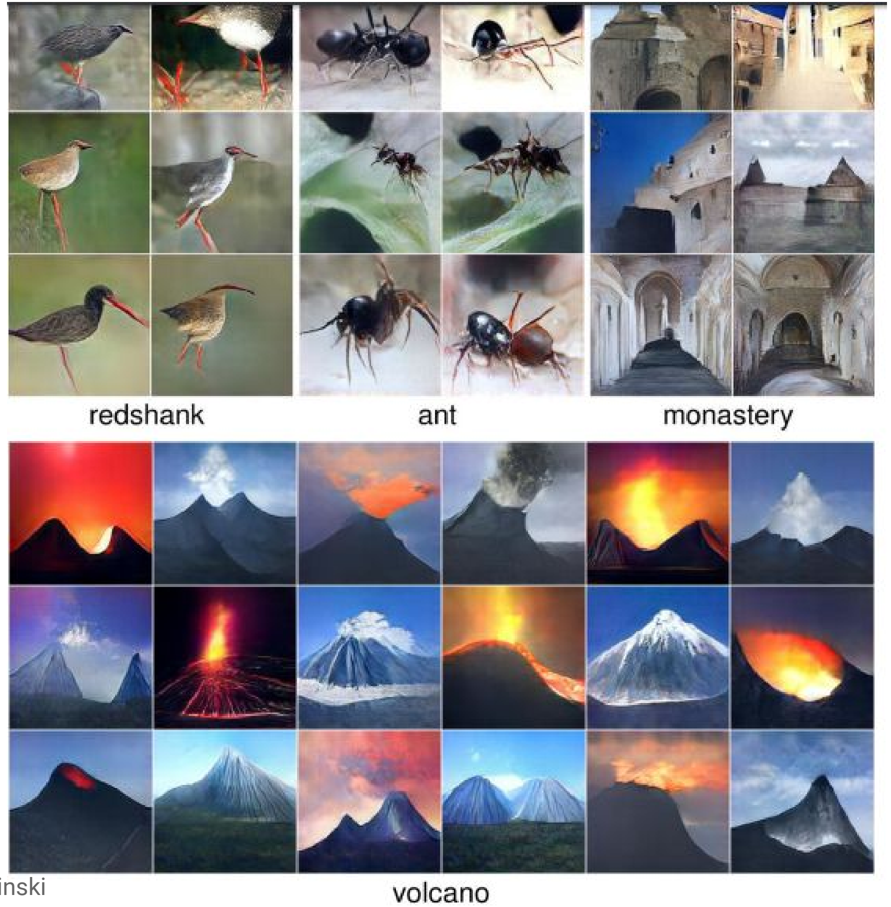


reconstructing

sampling

Learning the code distribution

Learning the code distribution: PPGN



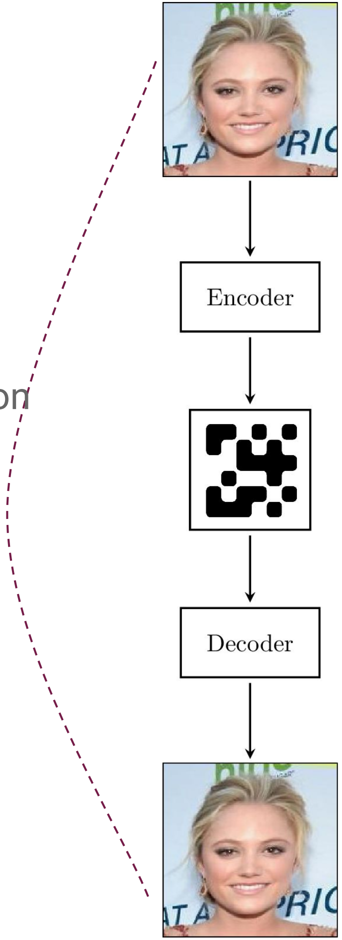
Plug and play generative models

A. Nguyen, J. Clune, Y, Bengio, A. Dosovitskiy, J. Yosinski

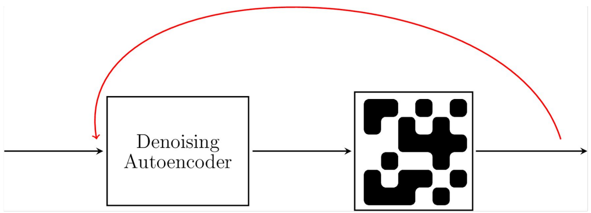
PPGN - key ingredients

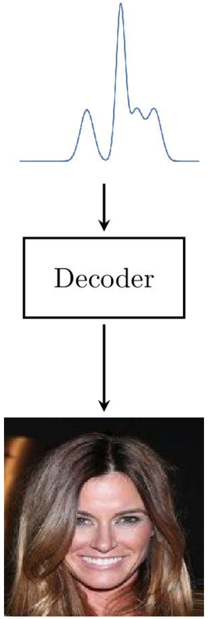
- Reconstructions
  - Autoencoder + GAN + Perceptual loss in feature space
- Samples
  - Markov Chain
  - Conditioning

PPGN



L_1/L_2 reconstruction
GAN loss
Perceptual loss

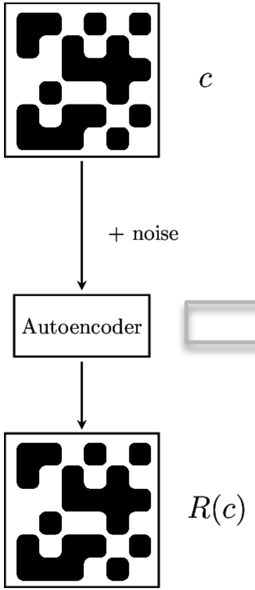




Not depicted:

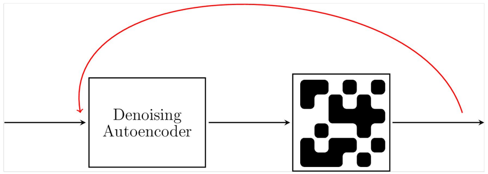
- Activation Maximization
- Encoder is pretrained classifier

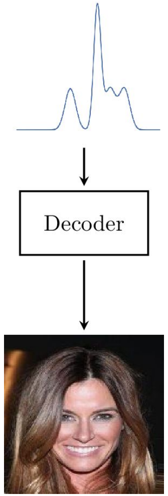
PPGN: sampling using a denoising autoencoder



$$\frac{\partial \log p(c)}{\partial c} \approx R(c) - c$$

$$c_{t+1} = c_t + \epsilon \frac{\partial \log p(c_t)}{\partial c_t} + N(0, \delta)$$





Learn $\frac{\partial \log p(c)}{\partial c}$ using a DAE, then sample using MCMC

Limitations of PPGN

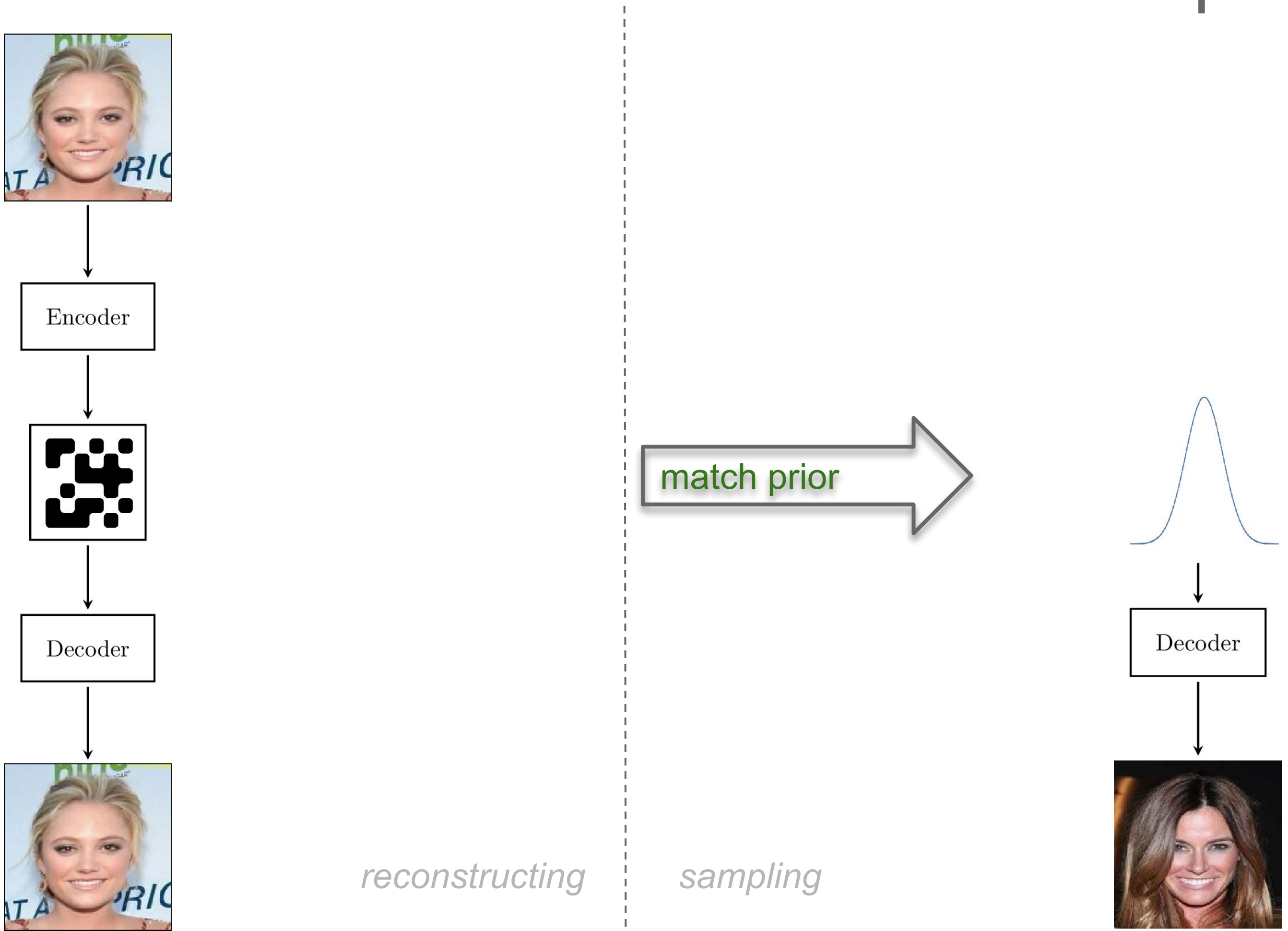
Learning the code distribution

- Not end to end
  - Need to pretrain an encoder on the same dataset
- Depends on labels for:
  - Conditioning samples
  - Pretrained encoder

- Markov Chains
  - when to stop?
  - missing rejection step

Matching a desired prior

2) Match the code distribution to a desired prior



Sounds familiar?

Variational inference - the ELBO

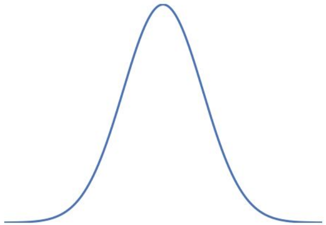
likelihood term

KL term

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) = \log \int p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} \geq \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] - \mathbf{KL}[q_\eta(\mathbf{z}|\mathbf{x})||p(\mathbf{z})]$$

good reconstructions

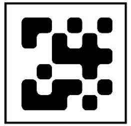
match the prior



Variational autoencoders

$\mathbf{x}$

Encoder



$q_\eta(\mathbf{z}|\mathbf{x})$

Decoder

computed analytically

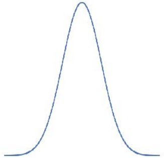
$\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})]$

reconstructing

computed analytically

$\mathrm{KL}[q_\eta(\mathbf{z}|\mathbf{x})||p(\mathbf{z})]$

match prior



$p(\mathbf{z})$

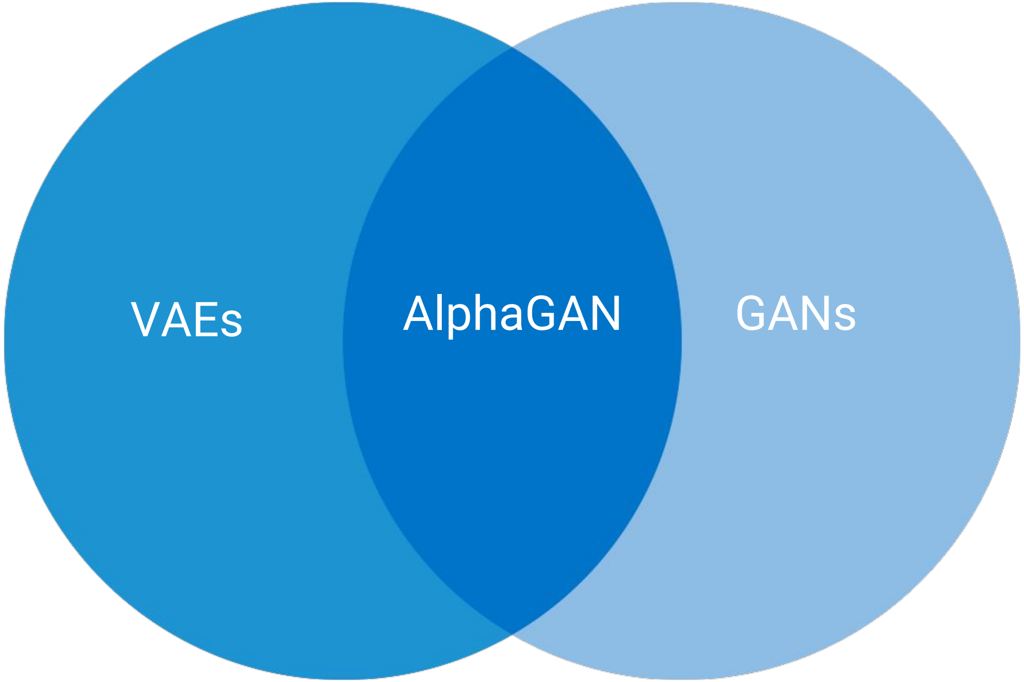
Decoder

sampling

AlphaGAN

combining GANs and VAEs

- variational
- inference
  - reconstructions
  - encoder network
- the posterior latent matches the prior

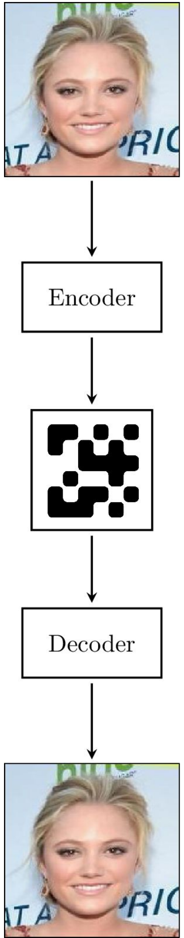


- implicit
  - encoder
  - decoder
- discriminators used to match distributions

DeepMind

Variational Approaches for Auto-Encoding Generative Adversarial Networks
M. Rosca, B. Lakshminarayanan, D. Warde-Farley, S. Mohamed

AlphaGAN



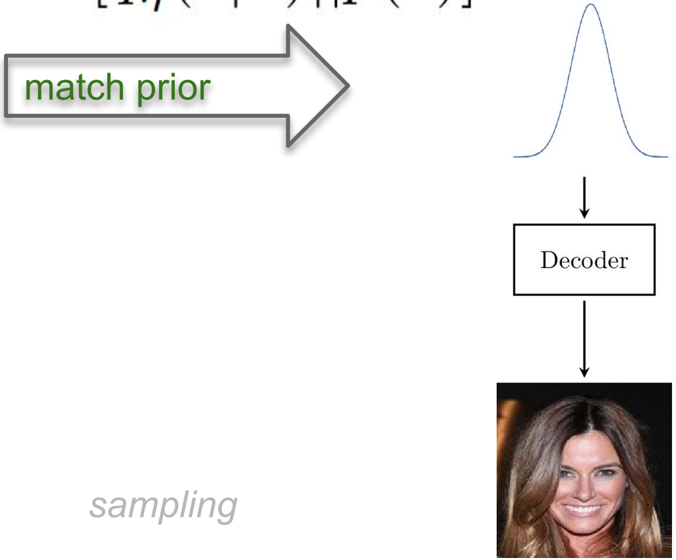
estimated from samples
data discriminator

$$\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})]$$

reconstructing

estimated from samples
codes discriminator

$$\mathrm{KL}[q_\eta(\mathbf{z}|\mathbf{x})||p(\mathbf{z})]$$



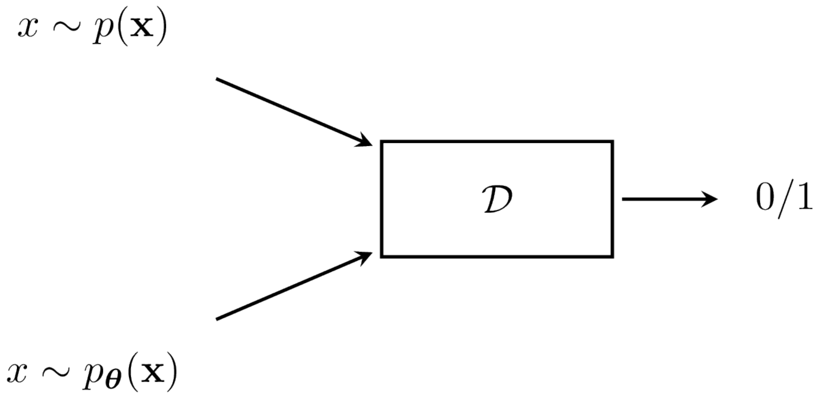
sampling

Q: how do we estimate the terms in the ELBO using GANs?

Density ratio trick

Estimate the ratio of two distributions only from samples, by building a binary **classifier** to distinguish between them.

$$\frac{p(\mathbf{x})}{p_{\boldsymbol{\theta}}(\mathbf{x})} = \frac{\mathcal{D}(x)}{1 - \mathcal{D}(x)}$$



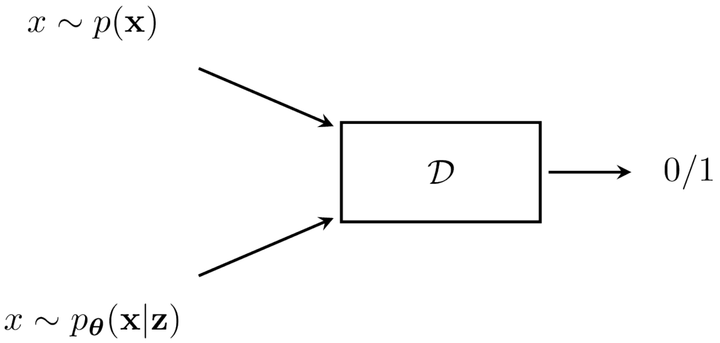
Using GANs for variational inference - the ELBO

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) = \log \int p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} \geq \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] - \mathbf{KL}[q_\eta(\mathbf{z}|\mathbf{x})||p(\mathbf{z})]$$

ELBO - likelihood term

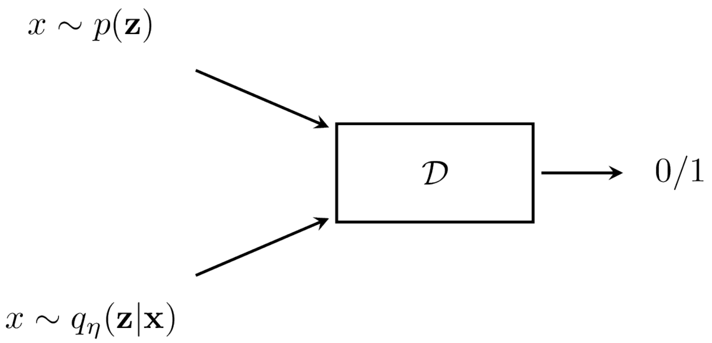
$$\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] = \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log(\frac{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})}{p(\mathbf{x})}p(\mathbf{x}))]$$

$$= \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log \underbrace{\frac{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})}{p(\mathbf{x})}}_{\text{ratio}}] + \underbrace{\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x})]}_{\text{constant}}$$



ELBO - the KL term

$$-\mathbf{KL}[q_\eta(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})] = \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}\left[\log \frac{p(\mathbf{z})}{q_\eta(\mathbf{z}|\mathbf{x})}\right]$$



From the ELBO to loss functions

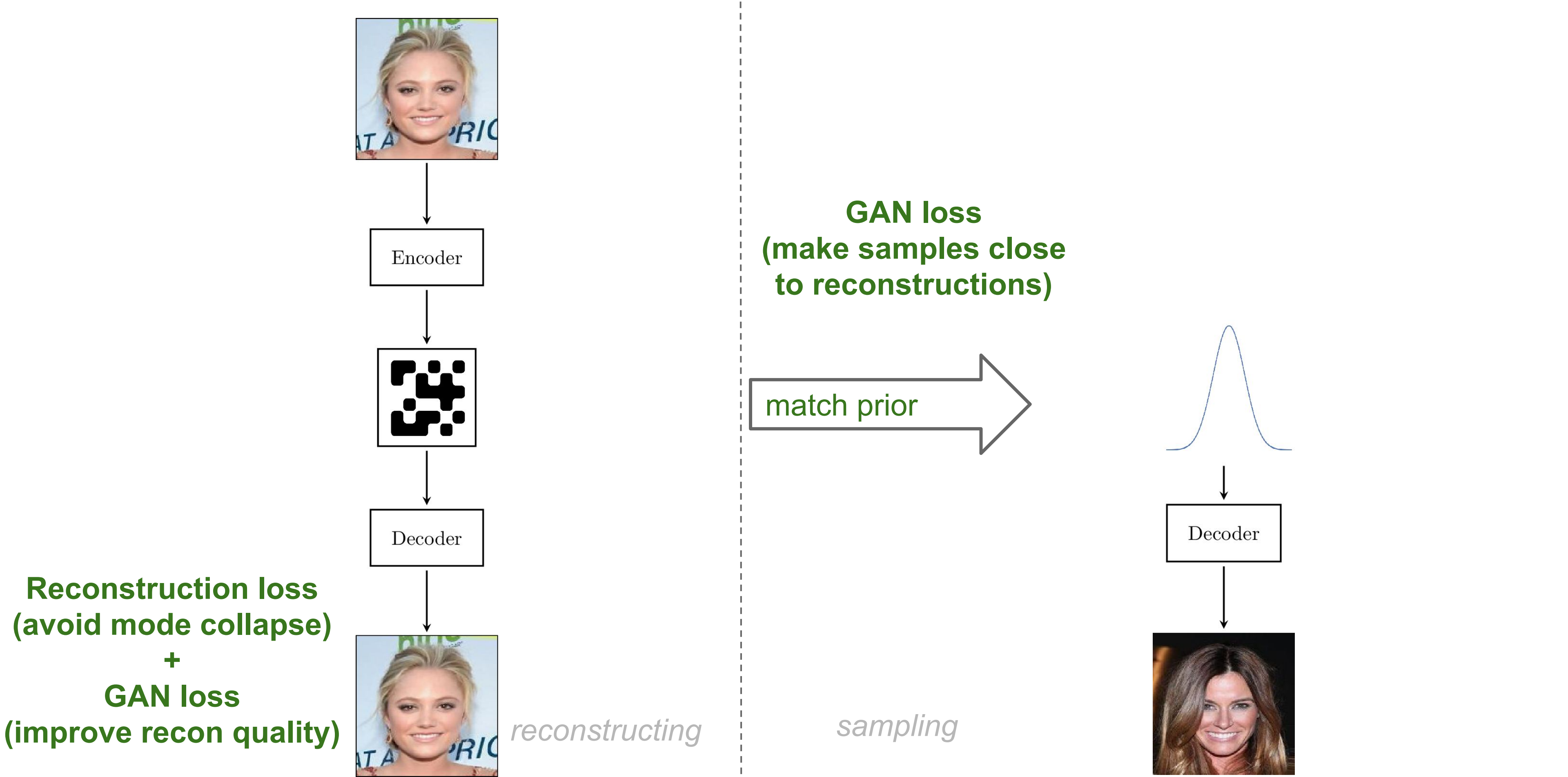
We want to match:

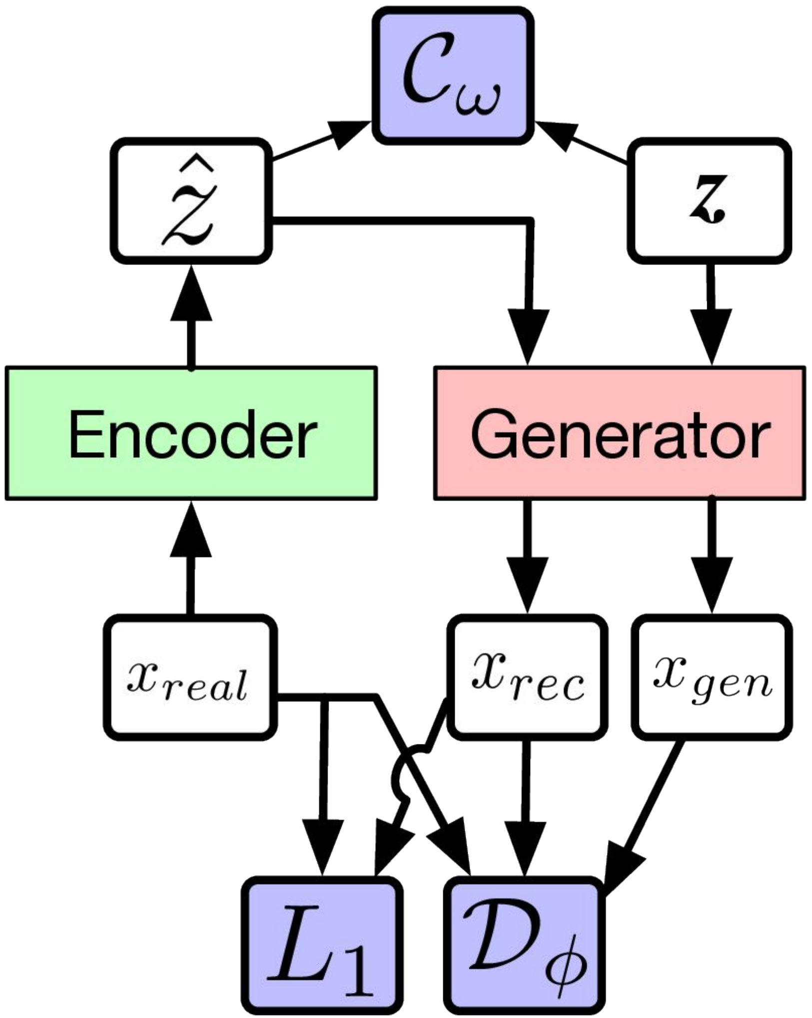
- the reconstruction and data distributions
  - likelihood term
- the code and prior distributions
  - the KL

Tools for matching distributions:

- (**GAN**) the density ratio trick
- (**VAE**) observer likelihoods
  - reconstructions losses

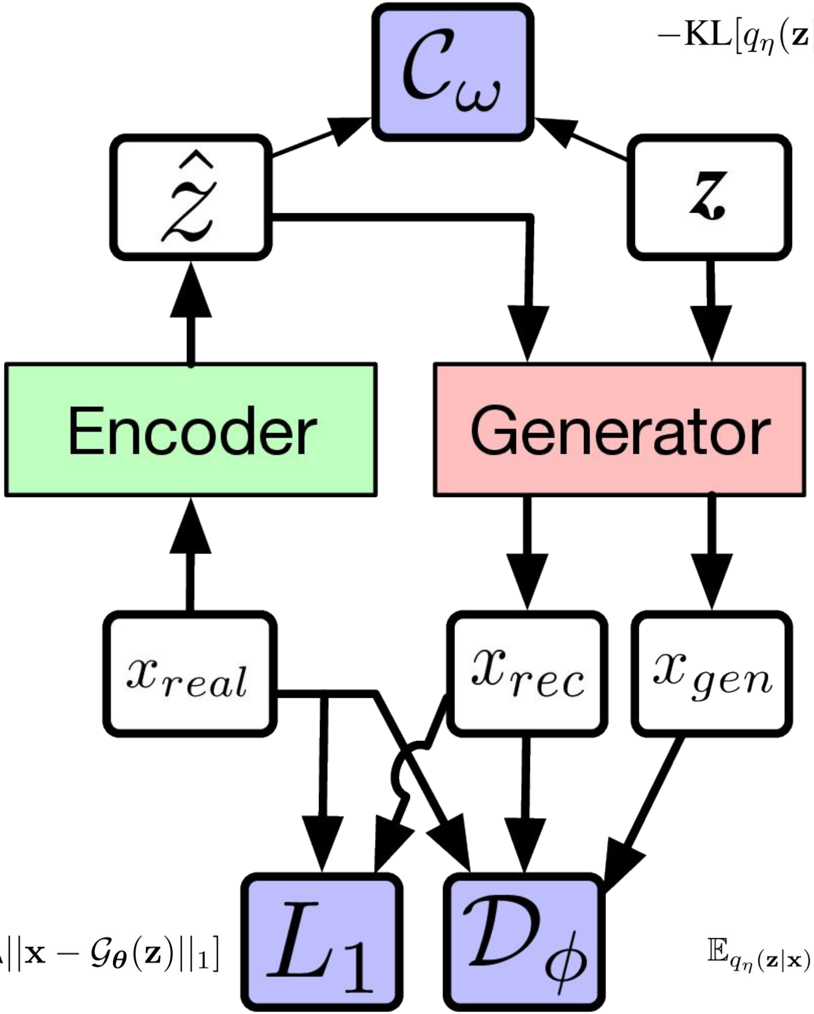
New loss functions - via the density ratio trick





$\mathcal{C}_\omega$
$\hat{z}$
z
Encoder
Generator
x_{real}
x_{rec}
x_{gen}
L_1
$\mathcal{D}_\phi$

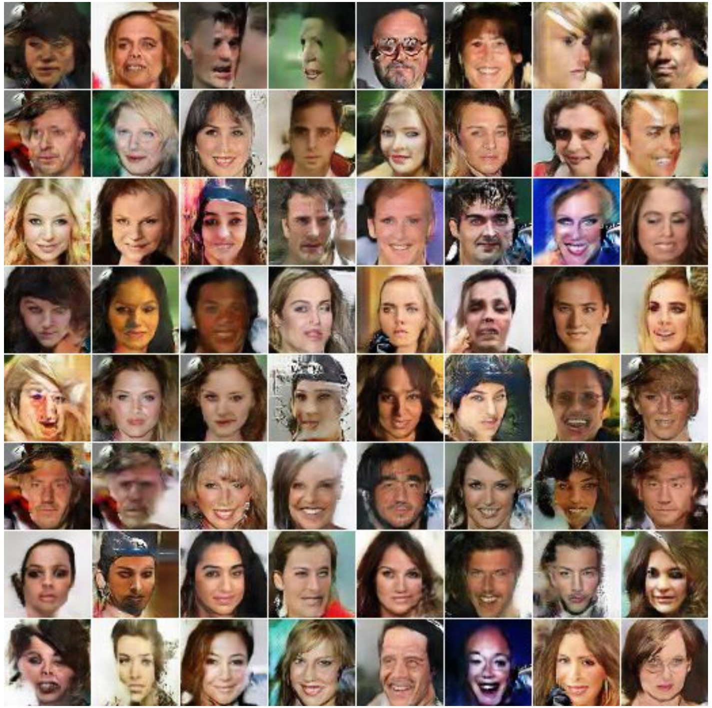
$$-\mathrm{KL}[q_\eta(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})] \approx \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}\left[\log \frac{\mathcal{C}_{\boldsymbol{\omega}}(\mathbf{z})}{1-\mathcal{C}_{\boldsymbol{\omega}}(\mathbf{z})}\right]$$



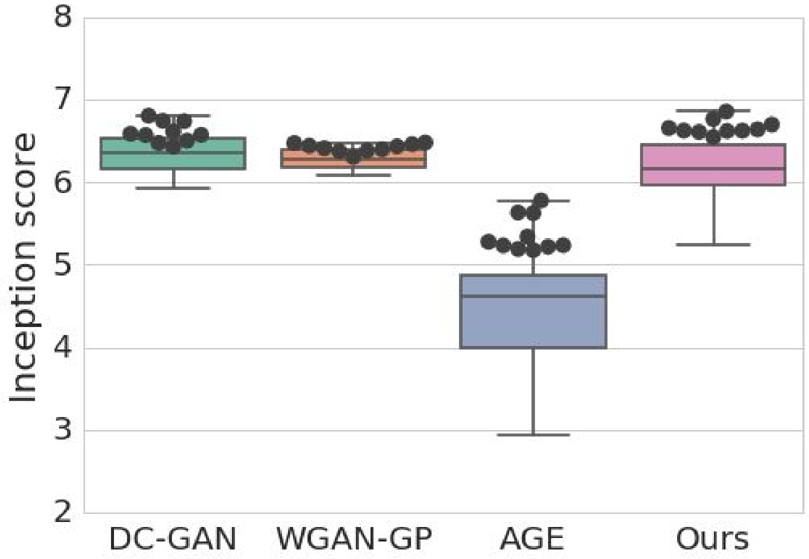
$$\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})\right] \approx \mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}\left[-\lambda||\mathbf{x}-\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z})||_1\right]$$

$$\mathbb{E}_{q_\eta(\mathbf{z}|\mathbf{x})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})\right] \approx \left[\log \frac{\mathcal{D}_{\boldsymbol{\phi}}(\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}))}{1-\mathcal{D}_{\boldsymbol{\phi}}(\mathcal{G}_{\boldsymbol{\theta}}(\mathbf{z}))}\right]$$

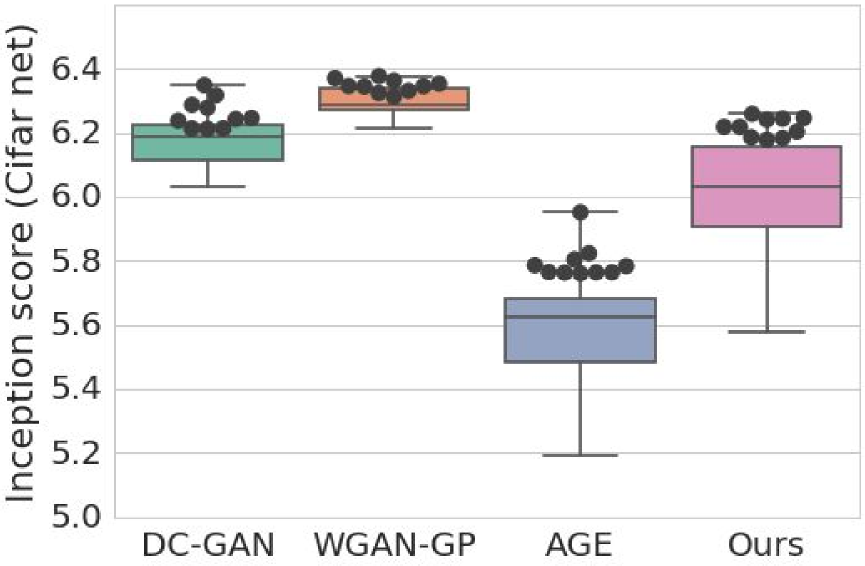
Samples



Cifar10 - Inception score



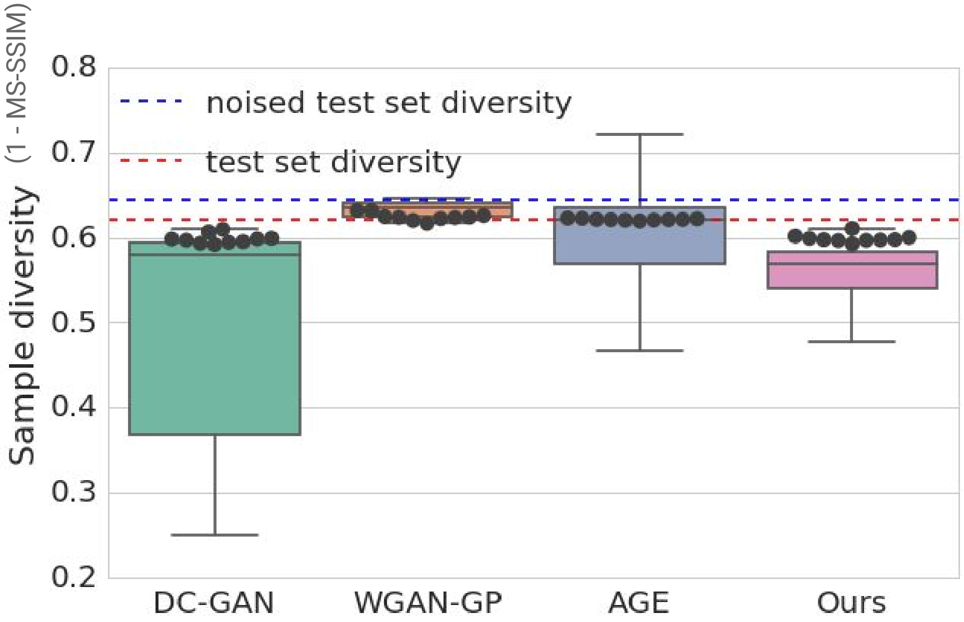
Classifier trained on Imagenet



Classifier trained on Cifar10

Improved Techniques for Training GANs
T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen

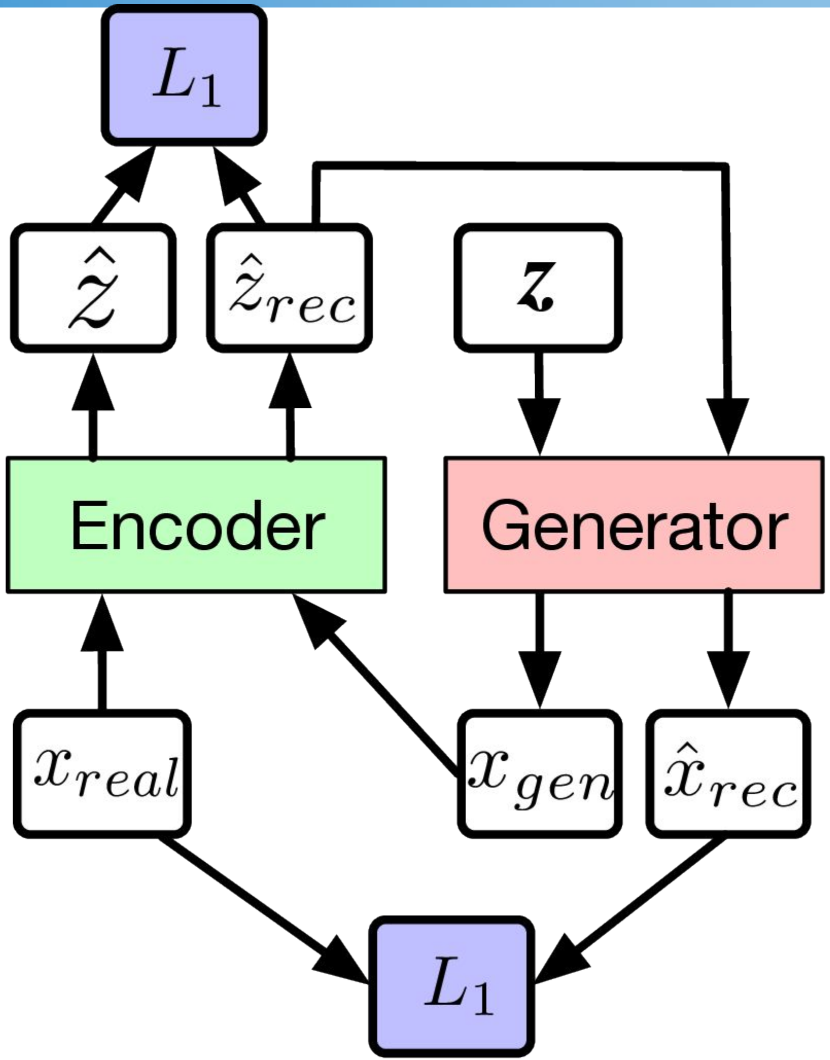
CelebA - sample diversity



Improved Techniques for Training GANs
T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen

Matching the prior - AGE

- the encoder is the discriminator
  - forces data codes to match the prior
  - sample codes to not match the prior



It Takes (Only) Two: Adversarial Generator-Encoder Networks
D. Ulyanov, A. Vedaldi, V. Lempitsky

Analysis of methods

Algorithm	Likelihood		Prior		
	Observer	Ratio estimator ("synthetic")	KL (analytic)	KL (approximate)	Ratio estimator
VAE	✓		✓		
DCGAN		✓			
VAE-GAN	✓	*	✓		
Adversarial-VB	✓				✓
AGE	✓			✓	
α-GAN (ours)	✓	✓			✓

Table 1: Comparison of different approaches for training generative latent variable models.

Summary

Try Autoencoder GANs if mode collapse is a problem.

Combining different learning principles results in a family of novel algorithms.

References

It Takes (Only) Two: Adversarial Generator-Encoder Networks
D. Ulyanov, A. Vedaldi, V. Lempitsky

Improved Techniques for Training GANs
T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen

Variational Approaches for Auto-Encoding Generative Adversarial Networks
M. Rosca, B. Lakshminarayanan, D. Warde-Farley, S. Mohamed

Plug and play generative models
A. Nguyen, J. Clune, Y, Bengio, A. Dosovitskiy, J. Yosinski

Adversarial Autoencoders
A. Makhzani, J. Shlens, N.Jaitly, I. Goodfellow, B. Frey

Generative Adversarial Networks
I. Goodfellow, J. Pouget-Abadie, M. Mirza, B.Xu,
D. Warde-Farley, S. Ozair, A.Courville, Y. Bengio

Adversarial Variational Bayes: Unifying Variational Autoencoders and Generative Adversarial Networks
L. Mescheder, S. Nowozin, A. Geiger

Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks
A. Radford, L. Metz, S. Chintala

Auto-Encoding Variational Bayes
D. P. Kingma, M. Welling