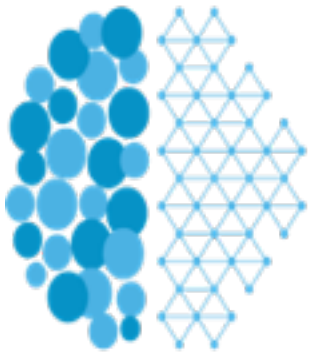
GANs and Unsupervised Representation Learning

Yoshua Bengio

March 19th, 2018

NYU, ECE Seminar Series on Modern AI



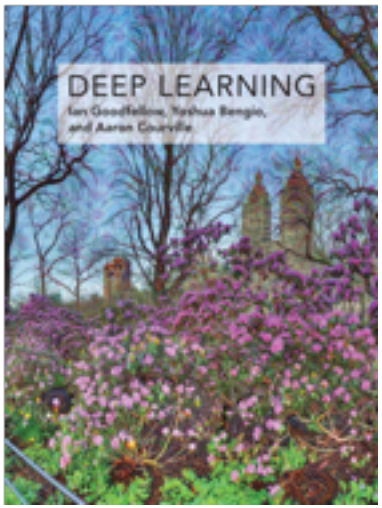
MILA

CIFAR
CANADIAN INSTITUTE FOR ADVANCED RESEARCH

Université de Montréal

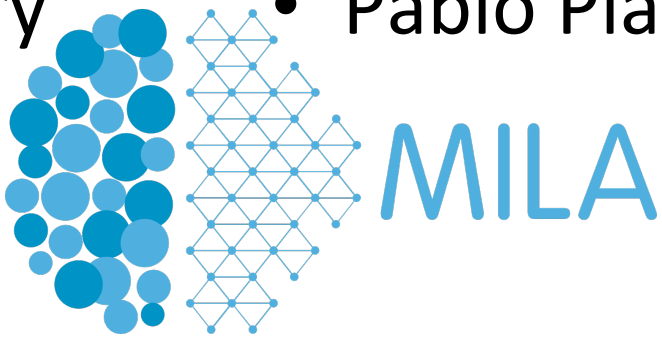
IVADO

PLUG: **Deep Learning**, MIT Press book is out, chapters will remain online



Thanks

- Devon Hjelm
- Philemon Brakel
- Aaron Courville
- Ishmael Belghazi
- Aristide Baratin
- Sai Rajeswar
- Clément Feutry
- Sherjil Ozair
- Ian Goodfellow
- Athul Paul Jacob
- Gerry Che
- Adam Trischler
- Kyunghyun Cho
- Pablo Piantanida



Still Far from Human-Level AI

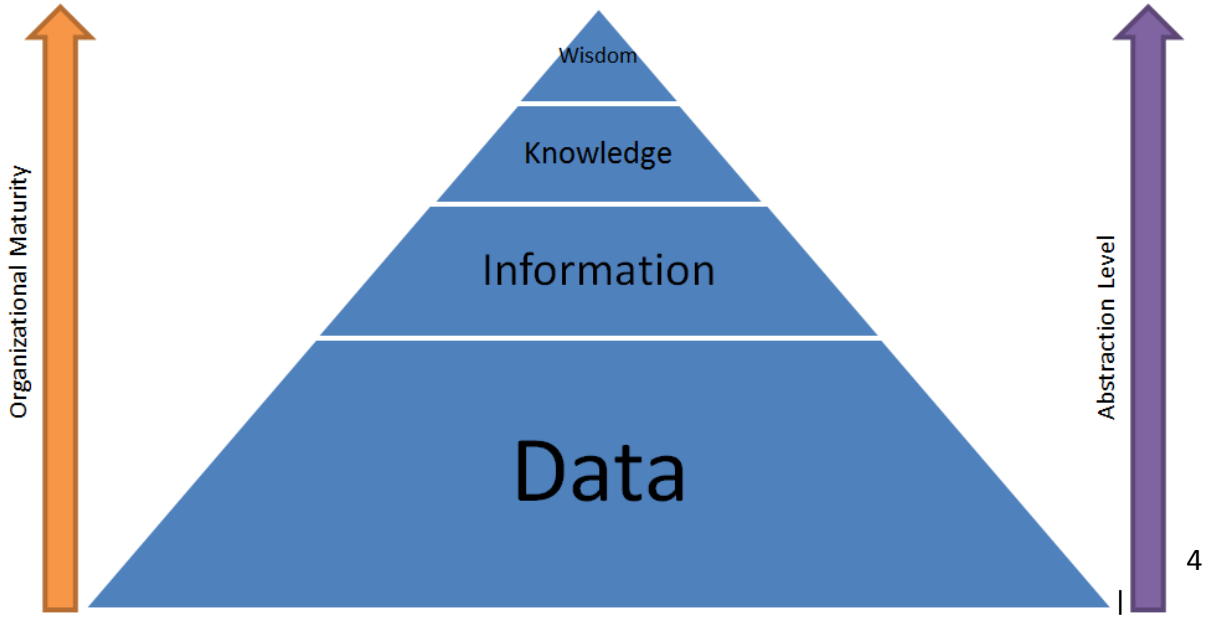
- Industrial successes mostly based on **supervised** learning

- Learning superficial clues, not generalizing well enough outside of training contexts, easy to fool trained networks:
  - Current models cheat by picking on surface regularities

Learning Multiple Levels of Abstraction

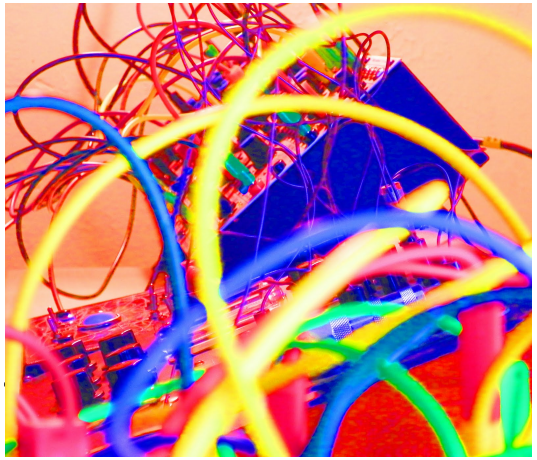
(Bengio & LeCun 2007)

- The big payoff of deep learning is to allow learning higher levels of abstraction
- Higher-level abstractions **disentangle the factors of variation**, which allows much easier generalization and transfer



Invariance and Disentangling

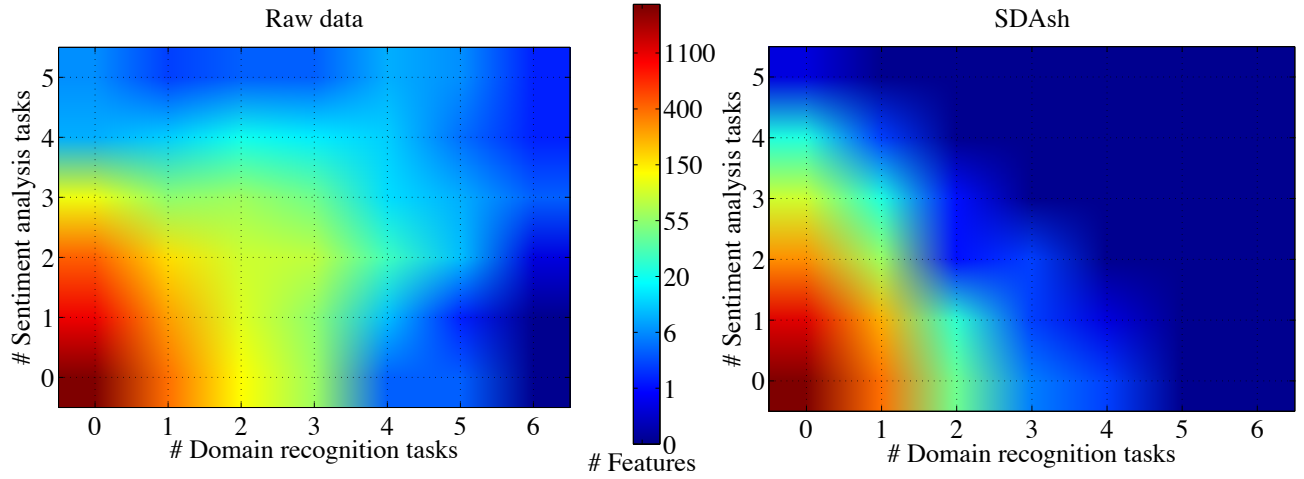
- Invariant features
- Which invariances?
- Alternative: learning to disentangle
- Good disentangling →
  avoid the curse of dimensionality:
  Dependencies are "simple" when the data is projected in the right abstract space



Disentangling from denoising objective

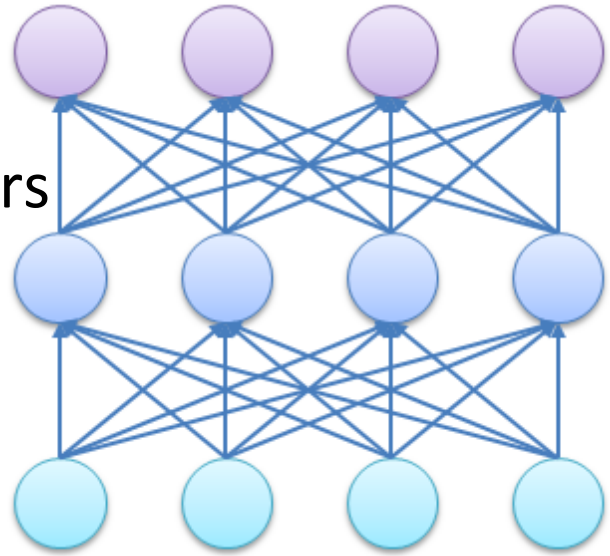
(Glorot, Bordes & Bengio ICML 2011)

- Early deep learning research already is looking for possible disentangling arising from unsupervised learning of representations
- Experiments on stacked denoising auto-encoders with ReLUs, on BoW text classification
- Features tend to specialize to either sentiment or domain



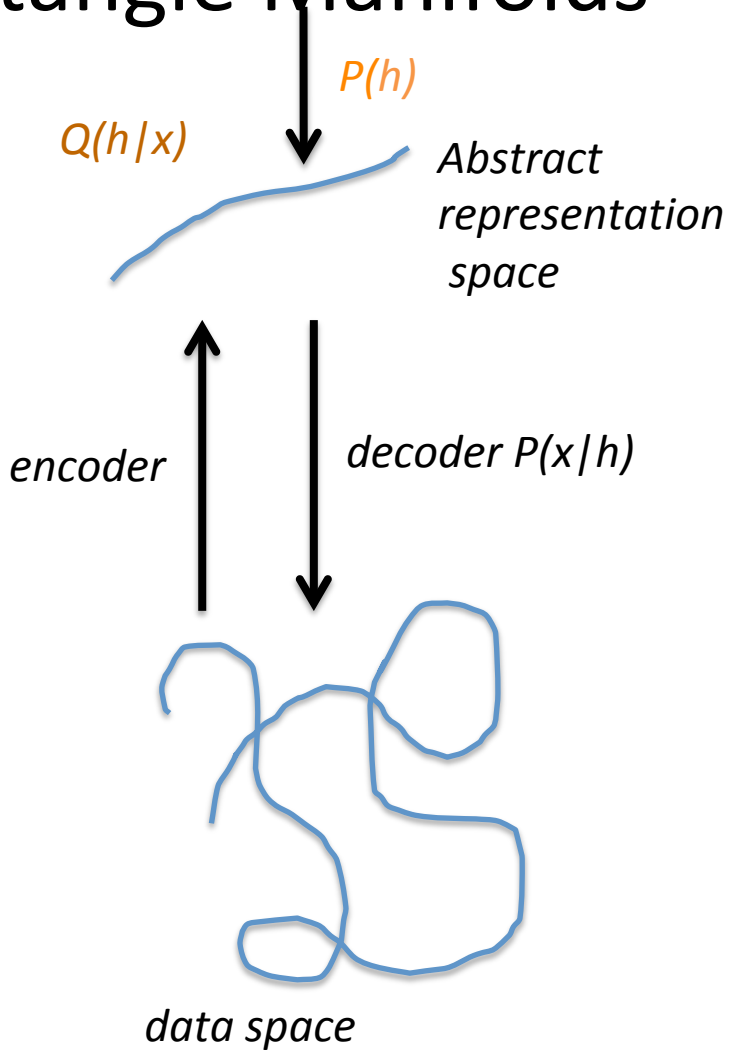
How to Discover Good Disentangled Representations

- How to discover abstractions?
- What is a good representation? ***(Bengio et al 2013)***
- Need clues (= priors) to help **disentangle** the underlying factors, such as
  - Spatial & temporal scales
  - Marginal independence
  - Simple dependencies between factors
    - *Consciousness prior*
  - Causal / mechanism independence
    - *Controllable factors*



Latent Variables and Abstract Representations to Disentangle Manifolds

- Encoder/decoder view: maps between low & high-levels
- Encoder does inference: interpret the data at the abstract level
- Decoder can generate new configurations
- Encoder flattens and disentangles the data manifold



Why Generative Models?

Generation

- **Conditional generation**
- Style transfer
- De-noising / image completion (inpainting)
- Super-resolution

Conditioning variable

Generated face

[Antipov et. al., 2017]

Given text

This bird has a yellow belly and tarsus, grey back, wings, and brown throat, nape with a black face

This bird is white with some black on its head and wings, and has a long orange beak

This flower has overlapping pink pointed petals surrounding a ring of short yellow filaments

Generated image

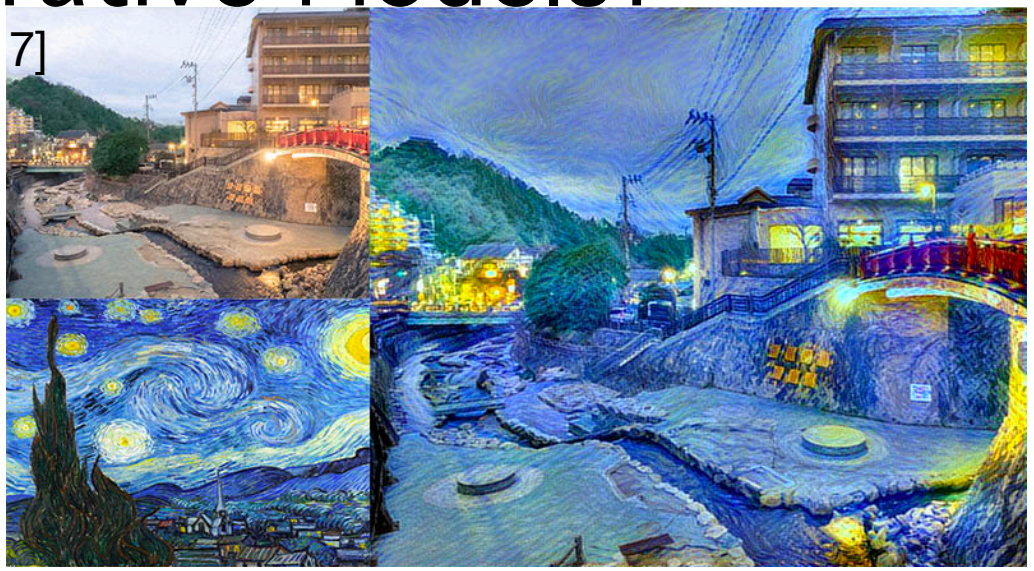
[Zhang et. al., 2016]

Why Generative Models?

Generation

- Conditional generation
- **Style transfer**
- De-noising / image completion (inpainting)
- Super-resolution

[à la Zhu et. al., 2017]



Why Generative Models?

Generation

- Conditional generation
- Style transfer
- **De-noising / image completion (inpainting)**
- Super-resolution

Missing pixels

Inpainted image

Why Generative Models?

Generation

- Conditional generation
- Style transfer
- De-noising / image completion (inpainting)
- **Super-resolution**

[Ledig et. al., 2016

Low res **High res**

Why Generative Models?

SOTA Generation

- Conditional generation
- Style transfer
- De-noising / image completion (inpainting)
- Super-resolution
- Drug discovery
- Speech synthesis
- Domain transfer
- And much much more

Those were all GANs

Figure 5: 1024 × 1024 images generated using the CELEBA-HQ dataset

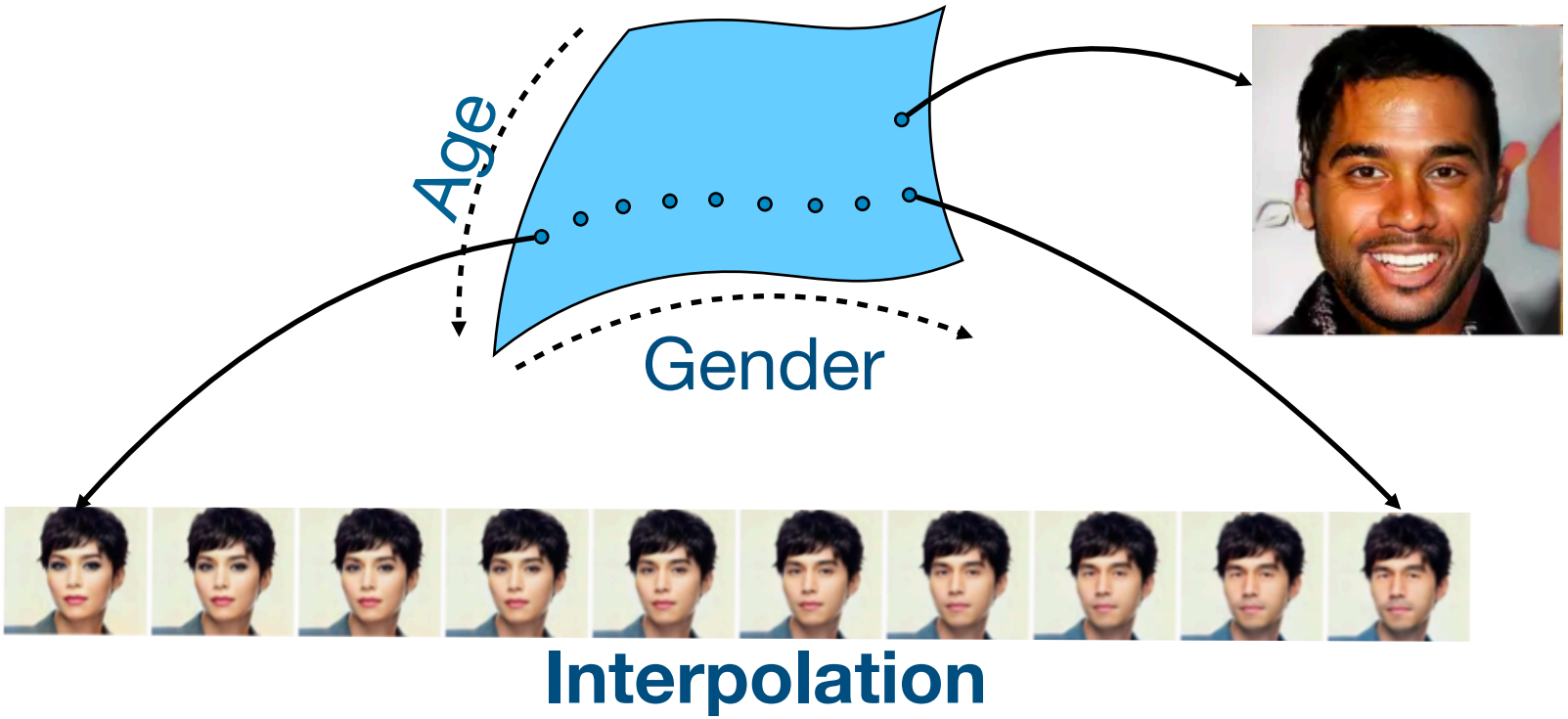
[Karras et. al., 2017]

Why Generative Models?

Discovery

- Learn relevant factors

"What I cannot create, I do not understand"
-Richard Feynman

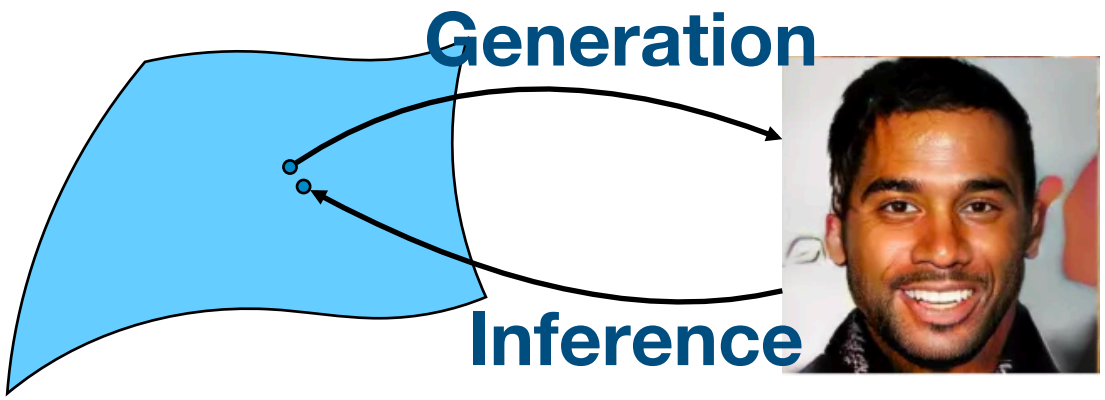


Why Generative Models?

Discovery

- Learn relevant factors
- Inference

"What I cannot create, I do not understand"
-Richard Feynman

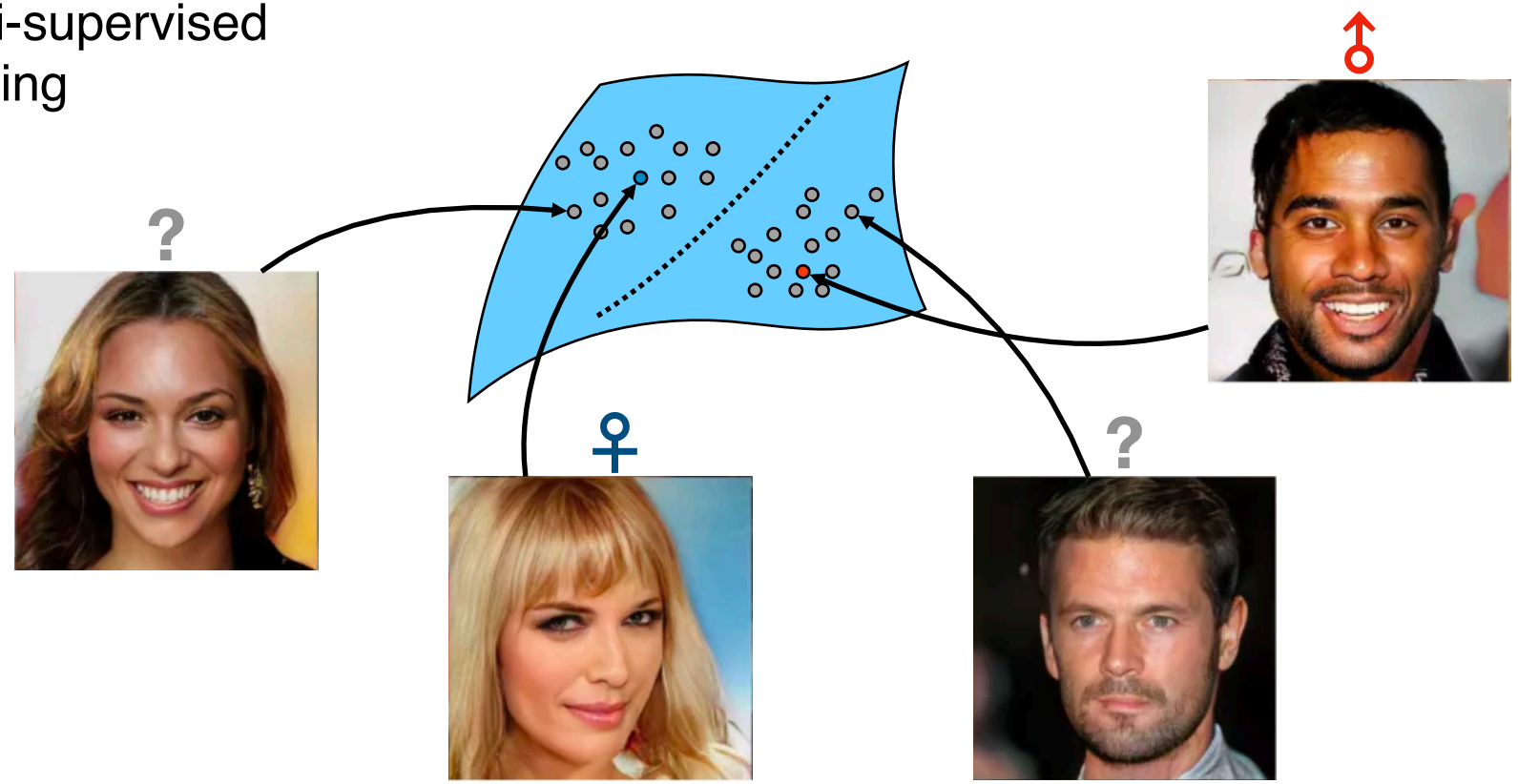


Why Generative Models?

Discovery

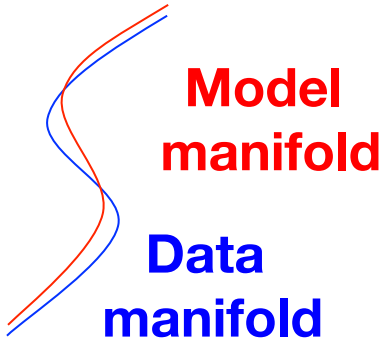
- Learn relevant factors
- Inference
- Semi-supervised learning

"What I cannot create, I do not understand"
-Richard Feynman



What's wrong with standard maximum likelihood?

- Pay a huge price for not putting probability mass at even a single training example, even if the data manifold and model manifold are very close.

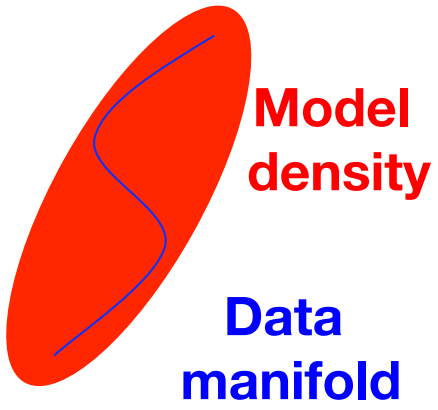


What's wrong with standard maximum likelihood?

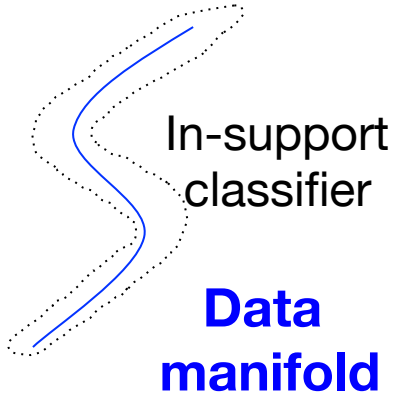
1. Pay a huge price for not putting probability mass at even a single training example, even if the data manifold and model manifold are very close.

   - So MLE makes the model distribution very fat and conservative

2. Another problem is that MLE measures error bits in pixel space whereas humans really care about errors in abstract space, so we would like loss measured in learned latent space



Classifiers for modeling distributions



- We were inspired by the work of Gutmann & Hyvarinen using probabilistic classifiers to estimate energy functions

  Gutmann & Hyvarinen 2012, Noise-Contrastive Estimation

- In high dimension, more relevant then density is whether you are in-support vs out-of-support

- A classifier of in-support vs out-of-support pays a *constant* price (rather than huge) for not putting support at a training example

Generative adversarial networks (GANs): a two player game with neural networks

Givens:

Samples from a **target distribution** $\mathbb{P}$

(Simple) prior $\mathbb{Q}_z$

$\mathbb{P}$

$\mathbb{Q}_z$

[Goodfellow et. al., 2014]

Generative adversarial networks (GANs): a two player game with neural networks

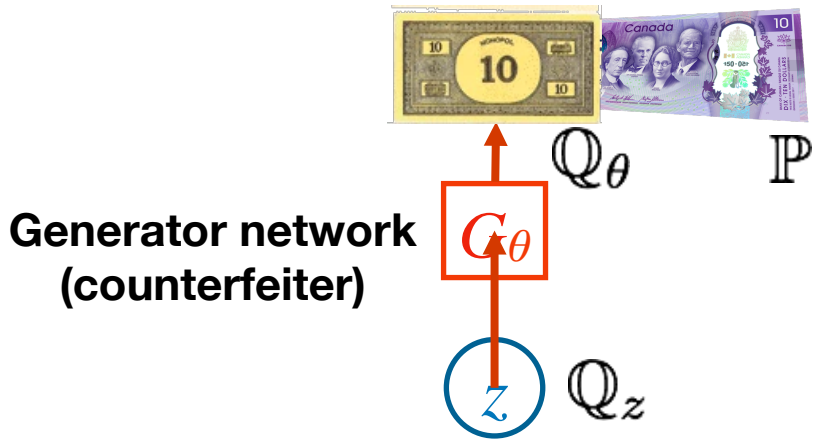
Givens:

Samples from a **target distribution** $\mathbb{P}$

(Simple) prior $\mathbb{Q}_z$

Player 1: Generator

A neural network with parameters, θ, whose samples **fool the discriminator**



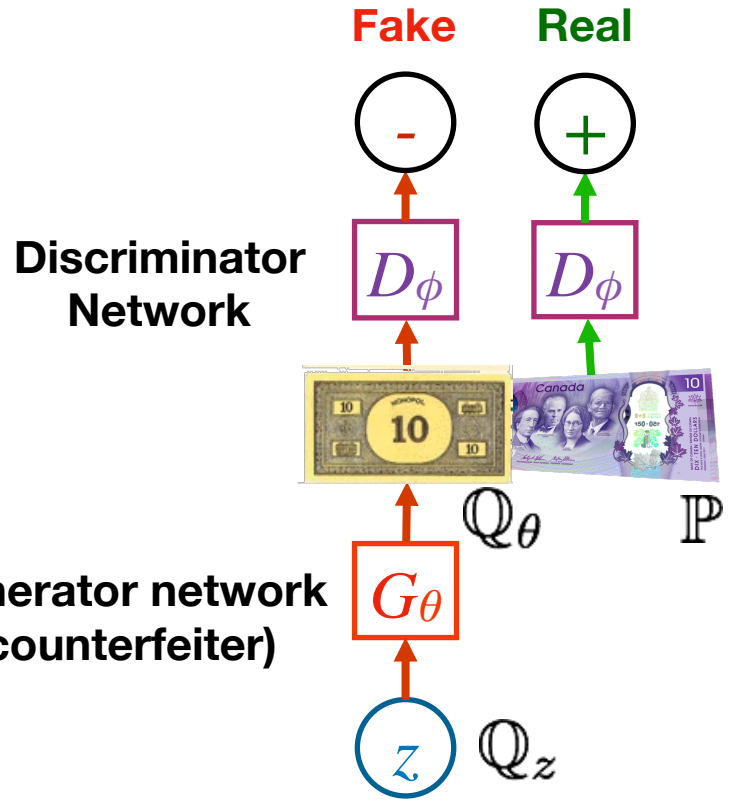
[Goodfellow et. al., 2014]

Generative adversarial networks (GANs): a two player game with neural networks

Givens:
Samples from a **target distribution** $\mathbb{P}$
(Simple) prior $\mathbb{Q}_z$

Player 1: Generator
A neural network with parameters, θ, whose samples **fool the discriminator**

Player 2: Discriminator
Distinguish (classify) real and fake correctly



[Goodfellow et. al., 2014]

Generative adversarial networks (GANs): a two player game with neural networks

Givens:
Samples from a **target distribution** $\mathbb{P}$
(Simple) prior $\mathbb{Q}_z$

Player 1: Generator
A neural network with parameters, θ, whose samples **fool the discriminator**

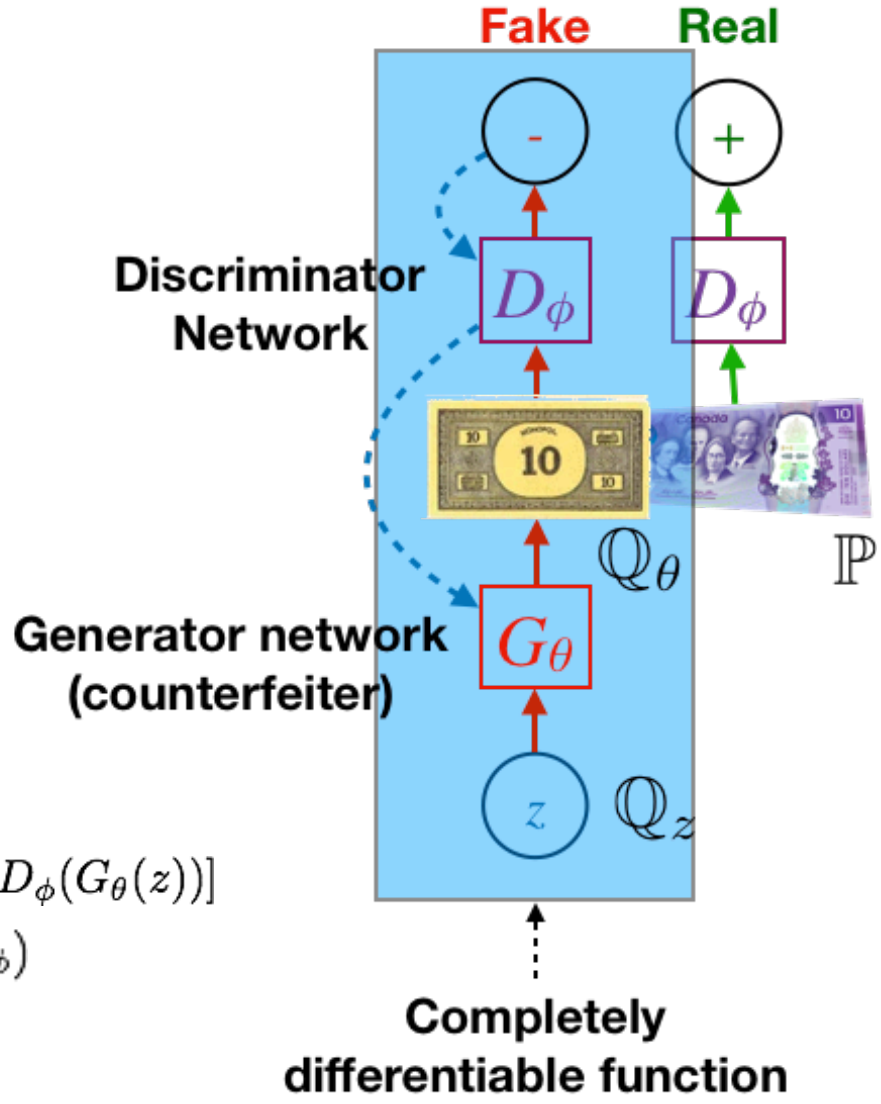
Player 2: Discriminator
Distinguish (classify) real and fake correctly

Minimax on *value function*

$$\mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta, D_\phi) = \mathbb{E}_{\mathbb{P}}\left[\log D_\phi(x)\right] + \mathbb{E}_{\mathbb{Q}_z}\left[\log(1 - D_\phi(G_\theta(z))\right]$$

$$(\hat{\theta}, \hat{\phi}) = \arg\min_\theta \arg\max_\phi \mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta, D_\phi)$$

Fine print: Continuous data only

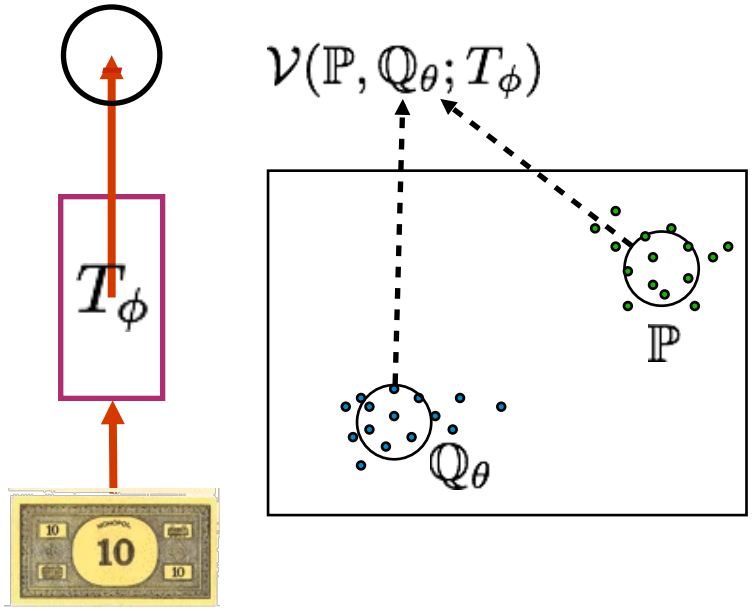


[Goodfellow et. al., 2014]

A closer look at the discriminator

- The discriminator defines a lower-bound

$$2 * \mathcal{D}_{JSD}(\mathbb{P}||\mathbb{Q}_\theta) - \log 4 \geq \mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)$$



[*f-GAN*. Nowozin et. al., 2017]

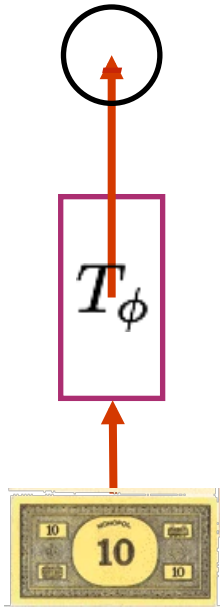
A closer look at the discriminator

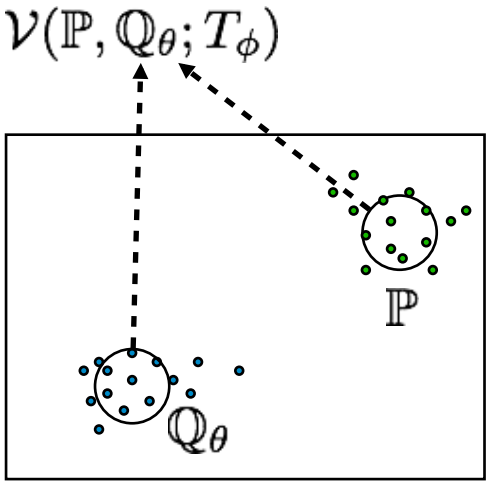
- The discriminator defines a lower-bound

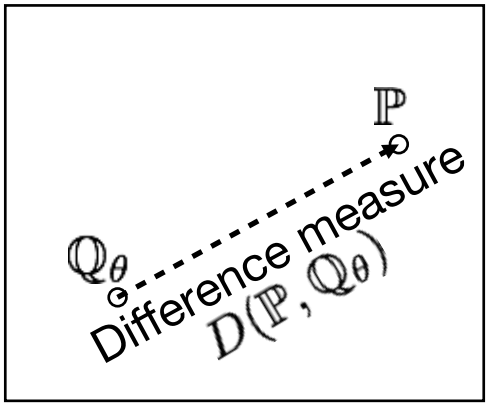
$$2 * \mathcal{D}_{JSD}(\mathbb{P}||\mathbb{Q}_\theta) - \log 4 \geq \mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)$$

- f-divergence

$$\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) = \mathbb{E}_{\mathbb{Q}_\theta}\left[f\left(\frac{p(x)}{q_\theta(x)}\right)\right]$$







Primal

[*f-GAN*. Nowozin et. al., 2017]

A closer look at the discriminator

- The discriminator defines a lower-bound

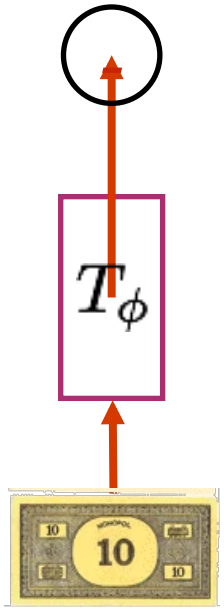
$$2 * \mathcal{D}_{JSD}(\mathbb{P}||\mathbb{Q}_\theta) - \log 4 \geq \mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)$$

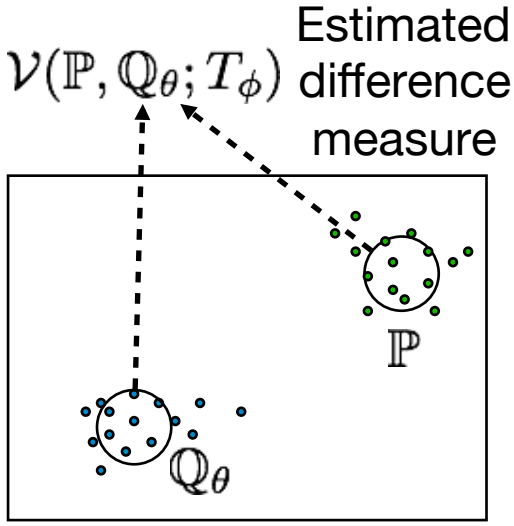
- f-divergence

$$\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) = \mathbb{E}_{\mathbb{Q}_\theta}\left[f\left(\frac{p(x)}{q_\theta(x)}\right)\right]$$

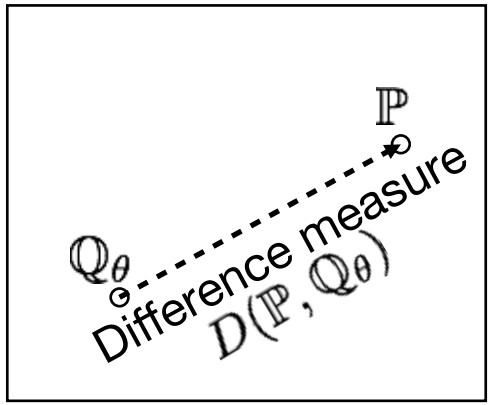
- Convex dual using neural networks

$$\begin{aligned}\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) &\geq \mathbb{E}_{\mathbb{P}}[T_\phi(x)] - \mathbb{E}_{\mathbb{Q}_\theta}[f^\star(T_\phi(x))] \\ &= \mathcal{V}_f(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)\end{aligned}$$





Dual using a classifier



Primal

[*f-GAN*. Nowozin et. al., 2017]

A closer look at the discriminator

- The discriminator defines a lower-bound

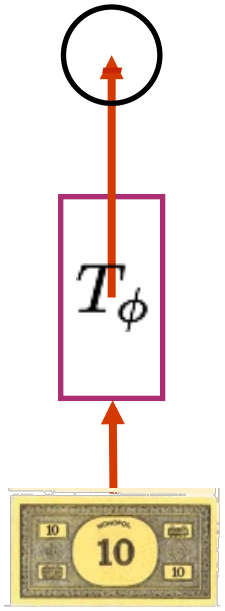
$$2 * \mathcal{D}_{JSD}(\mathbb{P}||\mathbb{Q}_\theta) - \log 4 \geq \mathcal{V}(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)$$

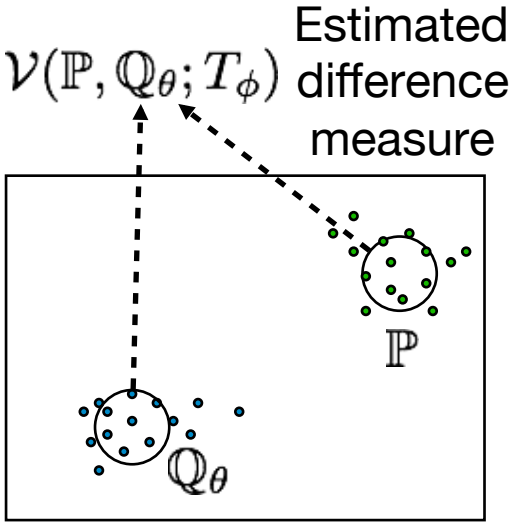
- f-divergence

$$\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) = \mathbb{E}_{\mathbb{Q}_\theta}\left[f\left(\frac{p(x)}{q_\theta(x)}\right)\right]$$

- Convex dual using neural networks

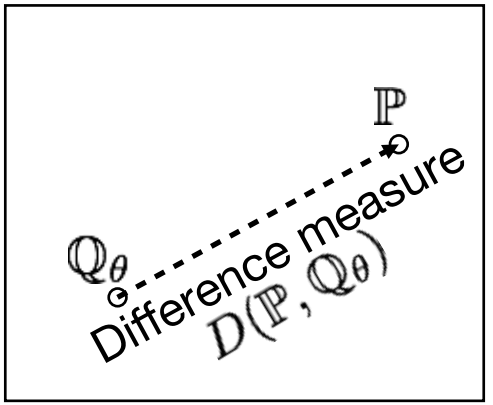
$$\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) \geq \mathbb{E}_{\mathbb{P}}[T_\phi(x)] - \mathbb{E}_{\mathbb{Q}_\theta}[f^\star(T_\phi(x))]$$
$$= \mathcal{V}_f(\mathbb{P}, \mathbb{Q}_\theta; T_\phi)$$





Dual using a classifier

- Estimate using samples
- **Other Examples**
  KL, Jensen-Shannon, Squared Hellinger, Pearson χ^2
- **GANS are a convex dual optimization with a classifier**



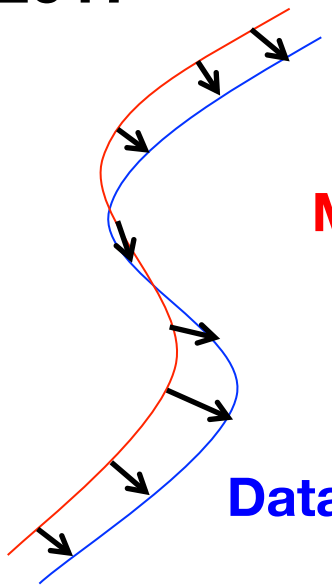
Primal

[*f-GAN*. Nowozin et. al., 2017]

WGAN

Arjowski et al 2017

- Penalize the Earth-Mover's distance between the generated and data distribution: pay a small price if the two manifolds do not overlap but are close in data space.



Model manifold

Data manifold

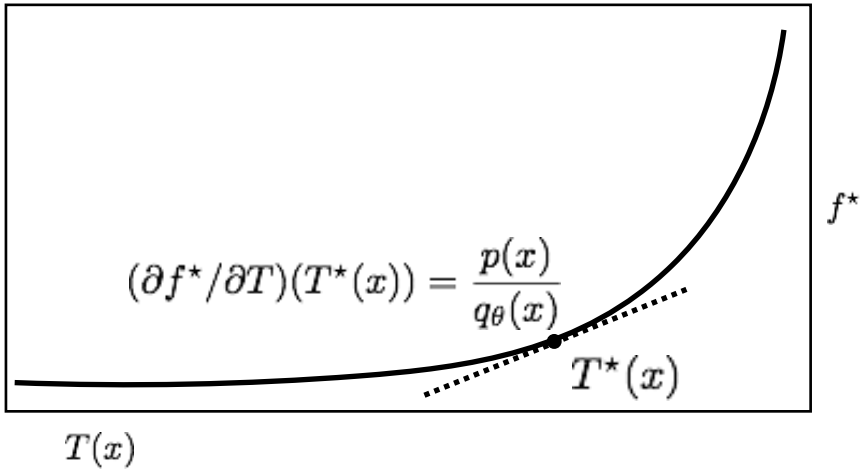
$$D_W(\mathbb{P}||\mathbb{Q}_\theta) = \sup_{||T_\phi||_L<1} \mathbb{E}_\mathbb{P}[T_\phi(x)] - \mathbb{E}_{\mathbb{Q}_\theta}[T_\phi(x)]$$

Estimating the likelihood ratio

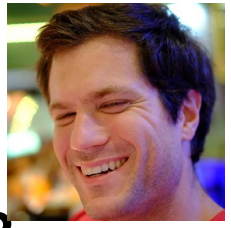
- Recall the **convex dual form**: $\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) \geq \mathbb{E}_{\mathbb{P}}[T_\phi(x)] - \mathbb{E}_{\mathbb{Q}_\theta}[f^\star(T_\phi(x))]$

- For **perfect discriminator** $T^\star$:

$$p(x) = (\partial f^\star / \partial T)(T^\star(x))q_\theta(x)$$



Boundary-Seeking GANs



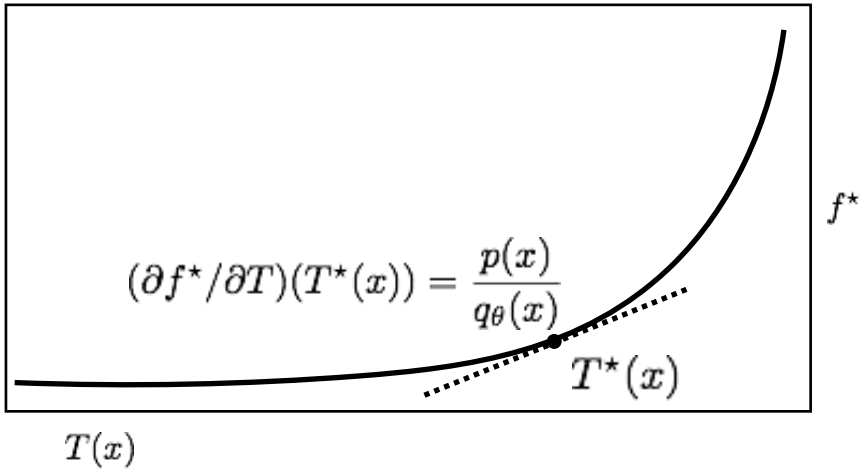
Hjelm, Jacob, Che, Trischler, Cho & Bengio ICLR 2018

- Recall the **convex dual form**:

$$\mathcal{D}_f(\mathbb{P}||\mathbb{Q}_\theta) \geq \mathbb{E}_{\mathbb{P}}[T_\phi(x)] - \mathbb{E}_{\mathbb{Q}_\theta}[f^\star(T_\phi(x))]$$

- For **perfect discriminator** $T^\star$:

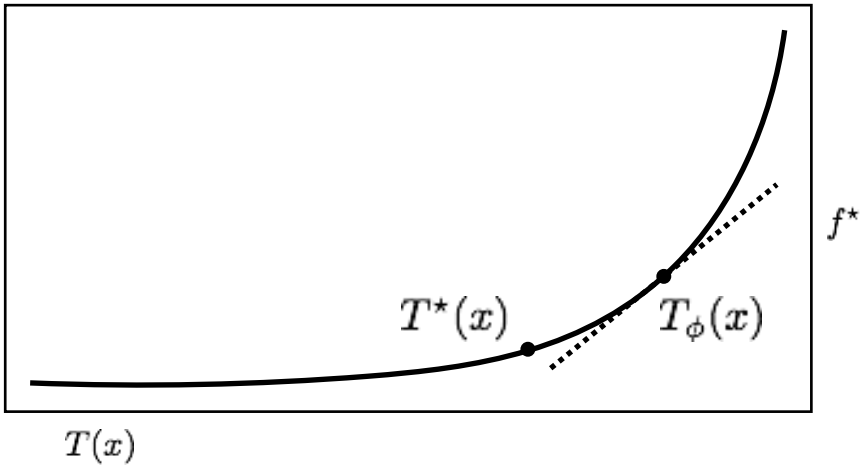
$$p(x) = (\partial f^\star / \partial T)(T^\star(x))q_\theta(x)$$



- Given a **neural network** T_ϕ:

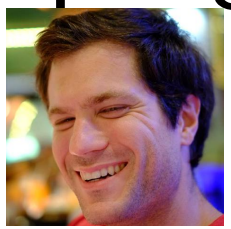
$$\tilde{p}(x) = \frac{w(x)}{\beta} q_\theta(x)$$

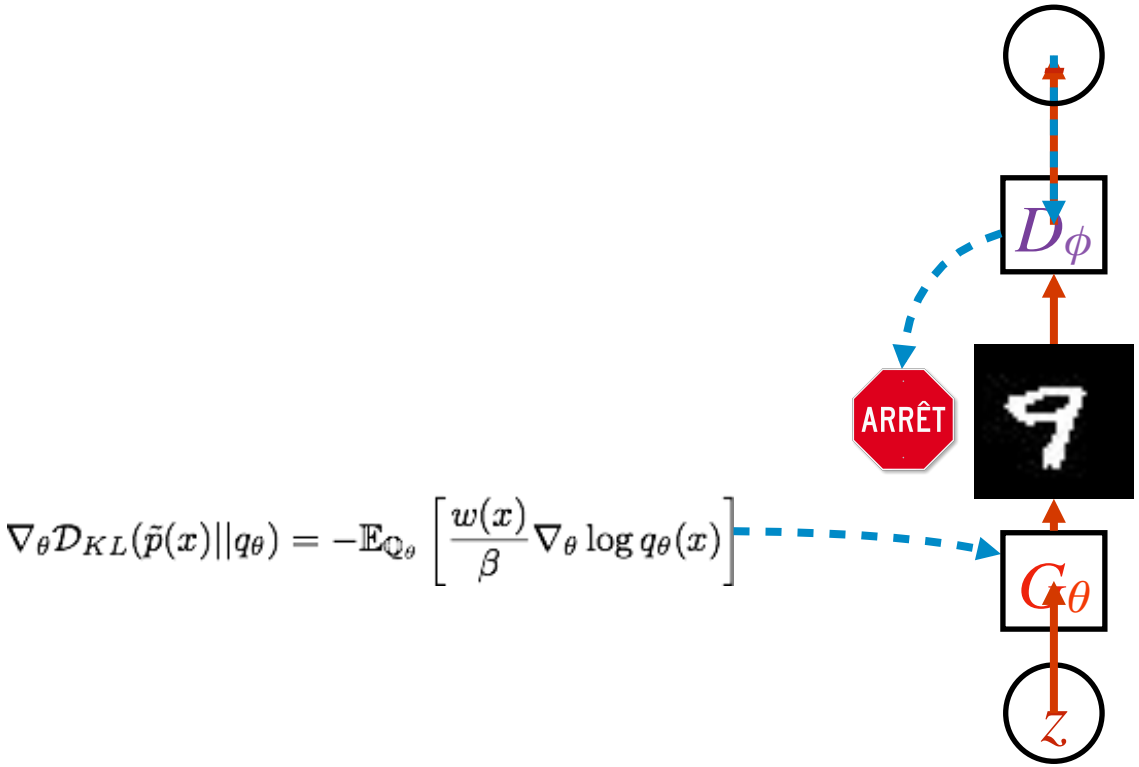
- Importance weights: $w(x) = (\partial f^\star / \partial T)(T_\phi(x))$
- Partition function: $\beta = \mathbb{E}_{\mathbb{Q}_\phi}[w(x)]$



BGAN: Importance sampling

Hjelm et al ICLR 2018





$$\nabla_\theta \mathcal{D}_{KL}(\tilde{p}(x)||q_\theta) = -\mathbb{E}_{\mathbb{Q}_\theta}\left[\frac{w(x)}{\beta}\nabla_\theta \log q_\theta(x)\right]$$

Gradient becomes 0 when p=q

[Hjelm et. al., 2017]

Qualitative discrete results [sic]

Discrete MNIST

Ground Truth

Generated

Quantized CelebA

- And it ’s miant a quert could he
- He weirst placed produces hopesi
- What ’s word your changerg bette
- ” We pait of condels of money wi
- Sance Jory Chorotic , Sen doesin
- In Lep Edger ’s begins of a find”,
- Lankard Avaloma was Mr. Palin ,
- What was like one of the July 2
- ” I stroke like we all call on a
- Thene says the sounded Sunday in
- The BBC nothing overton and slea
- With there was a passes ipposing
- About dose and warthestrinds fro
- College is out in contesting rev
- And tear he jumped by even a roy

Character-level 1-billion word (convnet) [Hjelm et. al., 2017]

BGAN: boundary-seeking
→ increased stability by pushing discriminator to the boundary

- The BGAN objective ends up pushing the discriminator output towards the decision surface whereas all the other GAN objectives can actually push it well *beyond* it

- Continuous case objective on generator with f-divergence (KL):

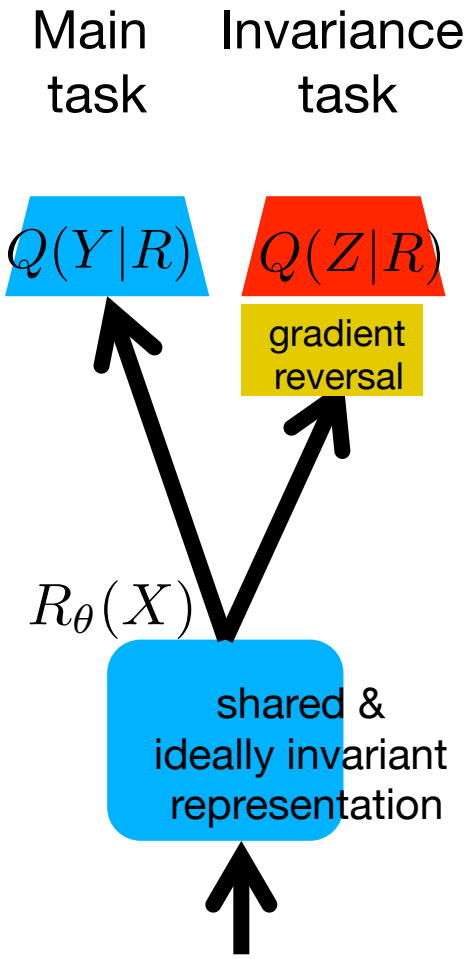
$$\min_{\theta} F_{\phi}(G_{\theta}(z))^2$$

Adversarial Domain Adaptation

Yaroslav Ganin

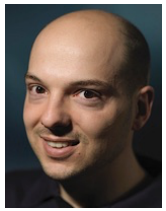
- Adversarial domain adaptation: Ganin & Lempitsky ICML 2015

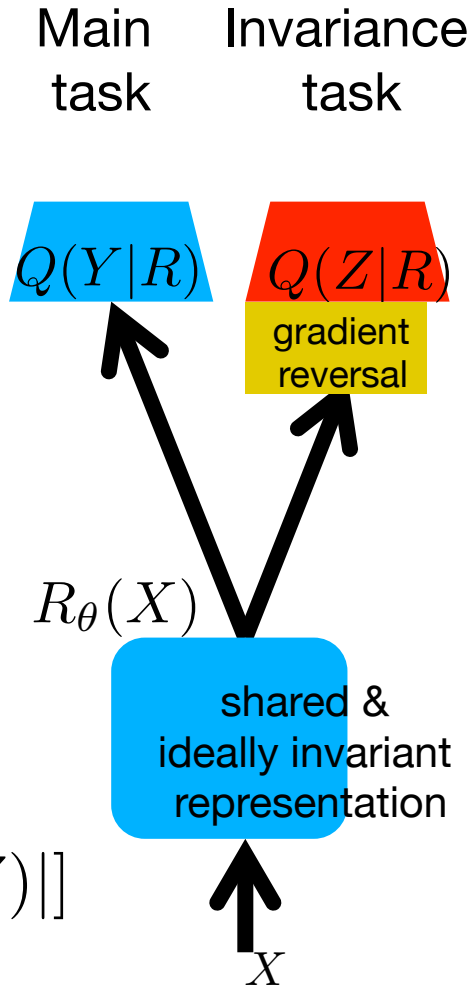
- Shared representation R(x) is trained to optimize main task Y|R(x) and worsen the prediction of the ‘domain’ variables Z wrt which we want the representation R(x) to be invariant.



Stability Trick in Adversarial Domain Adaptation

- Learning Anonymized Representations with Adversarial Neural Networks: Feutry et al arXiv:1802.09386

Clément Feutry   Pablo Piantanida



- Instead of *maximizing* cross-entropy on the invariance variables prediction Q(Z|R), bring it to the cross-entropy of the marginal distribution *Q(Z)*

$$\min_\theta \mathbb{E}_{X,Y,Z}[-\log Q(Y|R_\theta(X)) + \lambda|\log Q(Z|R_\theta(X)) - \log Q(Z)|]$$

Using a discriminator to optimize independence, mutual information or entropy

- The GAN discriminator is trained to estimate a similarity function between two distributions
- Two independent r-v A & B have the property that P(A,B)=P(A)P(B)
  - Given samples from P(A,B) you can obtain samples from P(A)P(B), e.g. by shuffling A values within a minibatch

Train a discriminator to separate between pairs (A,B) coming from P(A,B) and pairs coming from P(A) P(B)

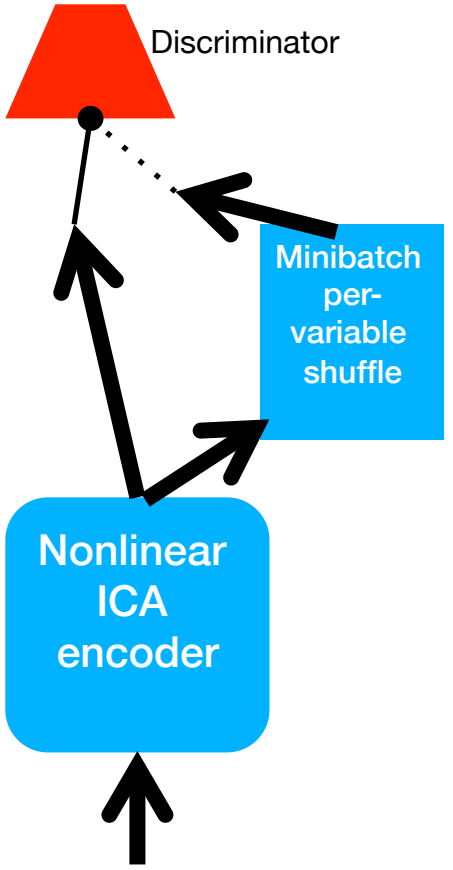
Brakel & Bengio ArXiv:1710.05050

Using a discriminator to optimize **independence**, mutual information or entropy

Brakel & Bengio ArXiv:1710.05050

- Train a discriminator to separate between pairs (A,B) coming from P(A,B) and pairs coming from P(A) P(B)

- Generalize this to measuring **independence** of all the outputs of a representation function (encoder). Maximize independence by backprop independence score into encoder → NON-LINEAR ICA.

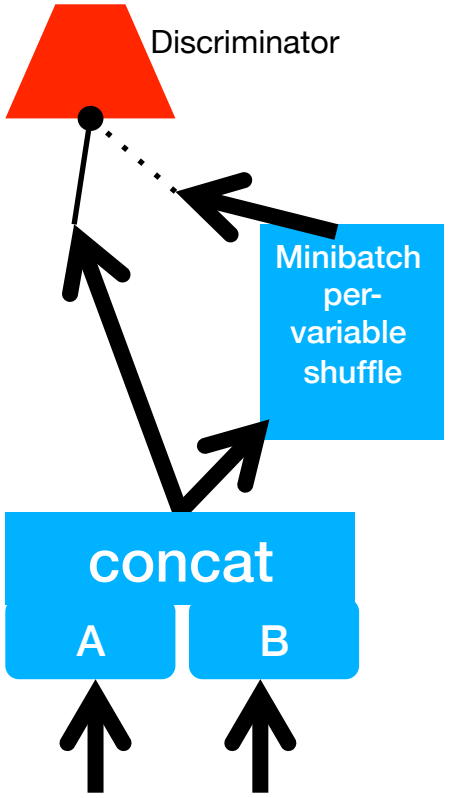


Using a discriminator to optimize independence, mutual information or entropy

MINE: Mutual Information Neural Estimator

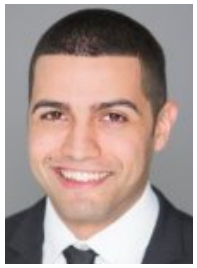
Belghazi et al ArXiv:1801.04062

Same architecture, but with a twist in the training objective which provides an asymptotically correct estimator of mutual independence



- Note that

$$MI(A,B) = H[A] - H[B|A]$$



Mutual information neural estimator (MINE)

Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, R Devon Hjelm

Mutual information: measure of dependence between two variables

$$I(X;Z) = \mathcal{D}_{KL}(\mathbb{P}_{X,Z}||\mathbb{P}_X \otimes \mathbb{P}_Z) = \mathbb{E}_{\mathbb{P}_{X,Z}}\left[\log\left(\frac{p(x,z)}{p(x)p(z)}\right)\right]$$

Fenchel convex dual (f-GAN): MINE-f

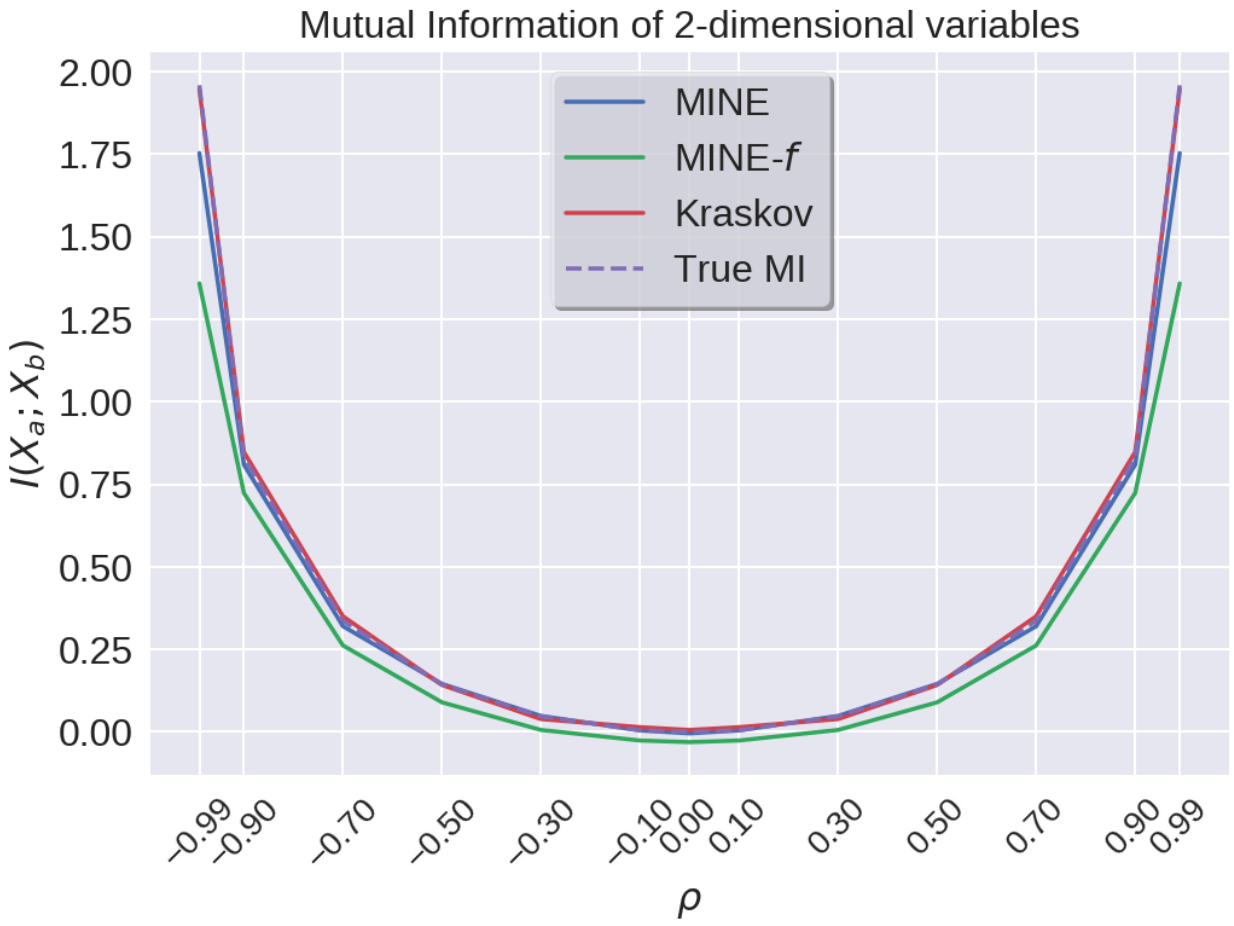
$$\mathcal{D}_{KL}(\mathbb{P}_{X,Z}||\mathbb{P}_X \otimes \mathbb{P}_Z) \geq \mathbb{E}_{\mathbb{P}_{X,Z}}[T_\phi(x)] - \mathbb{E}_{\mathbb{P}_X \otimes \mathbb{P}_Z}[e^{T_\phi(x)-1}]$$

Donsker-Varadhan (tighter): MINE

$$\mathcal{D}_{KL}(\mathbb{P}_{X,Z}||\mathbb{P}_X \otimes \mathbb{P}_Z) \geq \mathbb{E}_{\mathbb{P}_{X,Z}}[T_\phi(x)] - \log \mathbb{E}_{\mathbb{P}_X \otimes \mathbb{P}_Z}[e^{T_\phi(x)}]$$

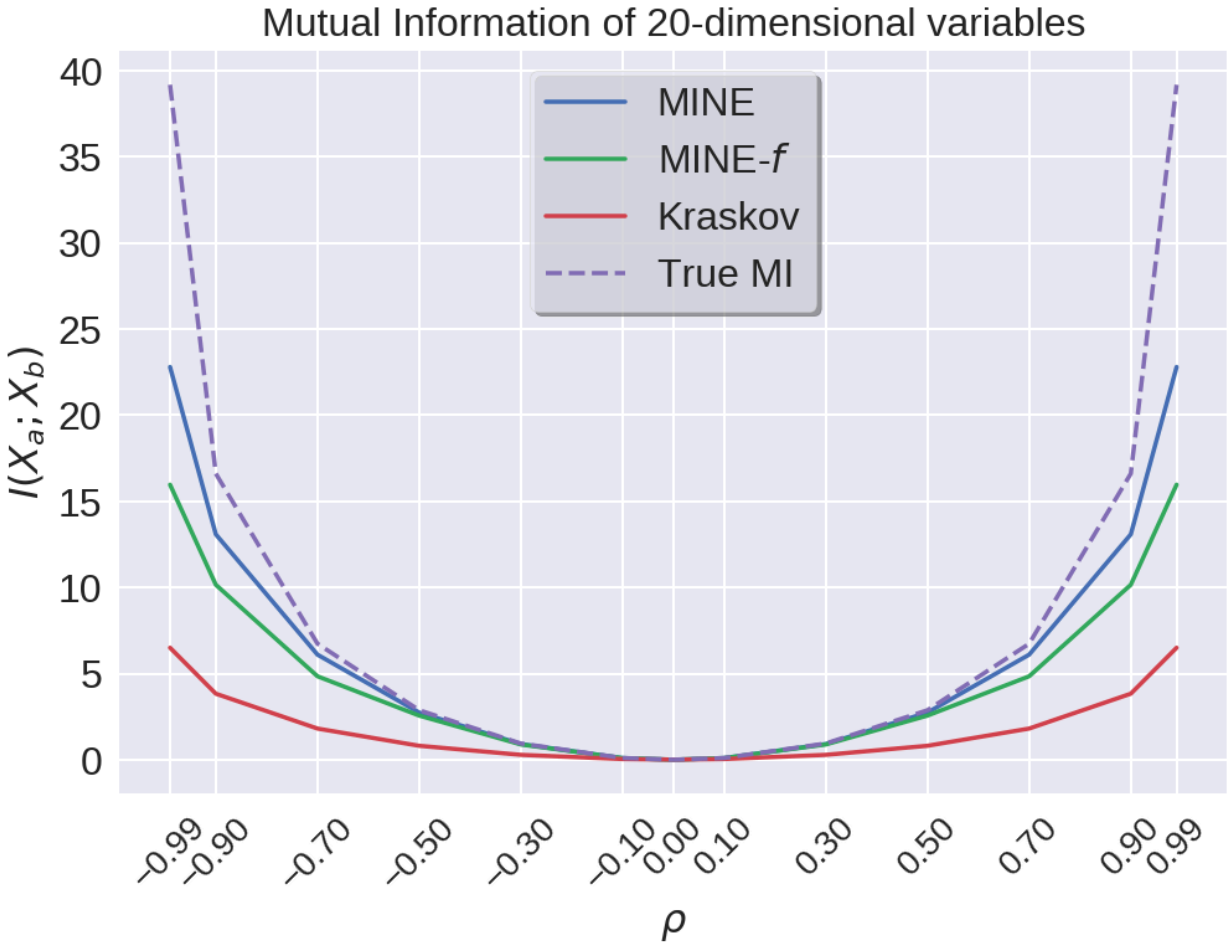
[Belghazi et. al., 2018]

Demonstration of estimation



[Belghazi et. al., 2018]

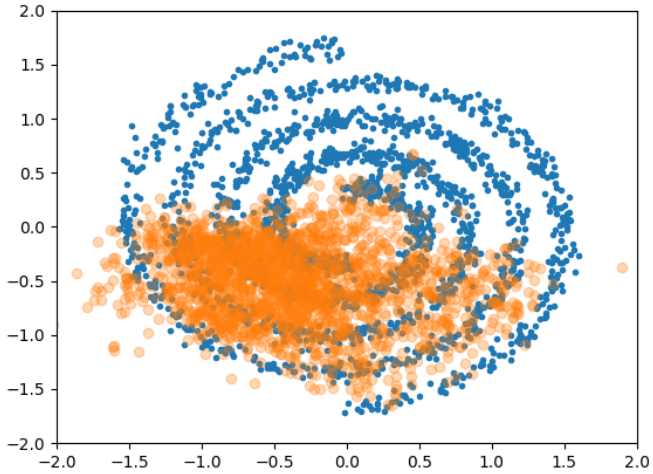
Demonstration of estimation



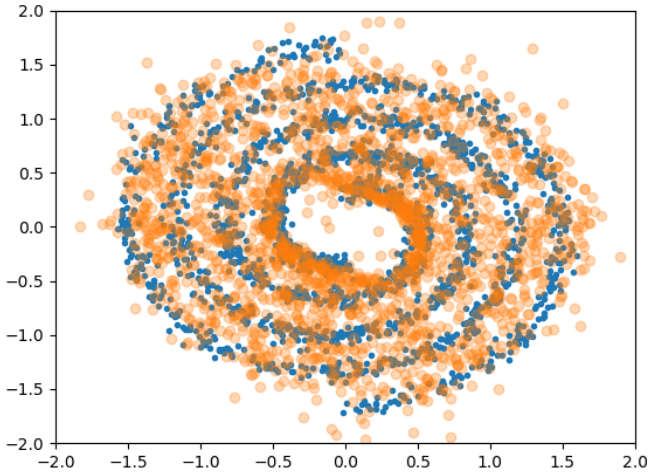
[Belghazi et. al., 2018]

Maximizing mutual information: avoid GAN mode dropping by max MI(X,Z)

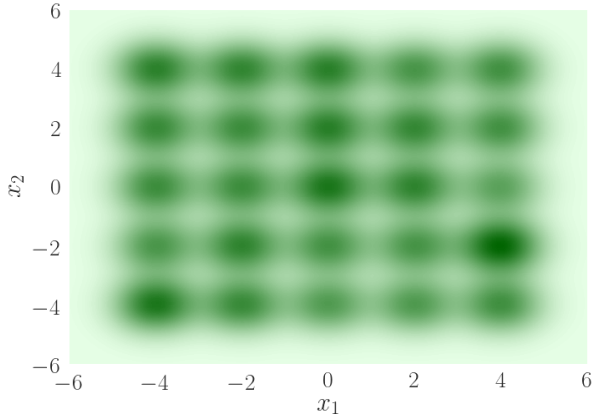
GAN



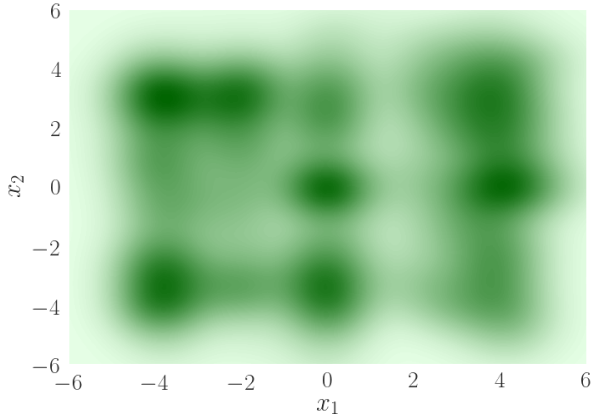
GAN+MINE



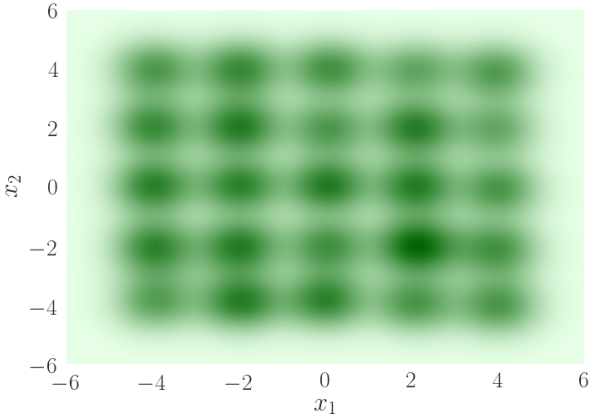
Ground Truth



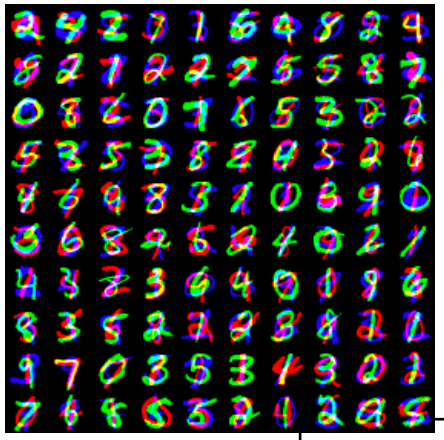
GAN



GAN+MINE



[Belghazi et. al., 2018]



Maximizing mutual information (stacked MNIST)

	Modes (max 1000)	$\mathcal{D}_{KL}(\mathbb{P}_Y \| \mathbb{Q}_Y)$
DCGAN	99	3,4
ALI	16	5,4
Unrolled GAN	48,7	4,32
VEEGAN	150	2,96
PacGAN	1000	0,6
DCGAN+MINE	1000	0,5

[Belghazi et. al., 2018]

Montreal Institute for Learning Algorithms

MILA

Université de Montréal