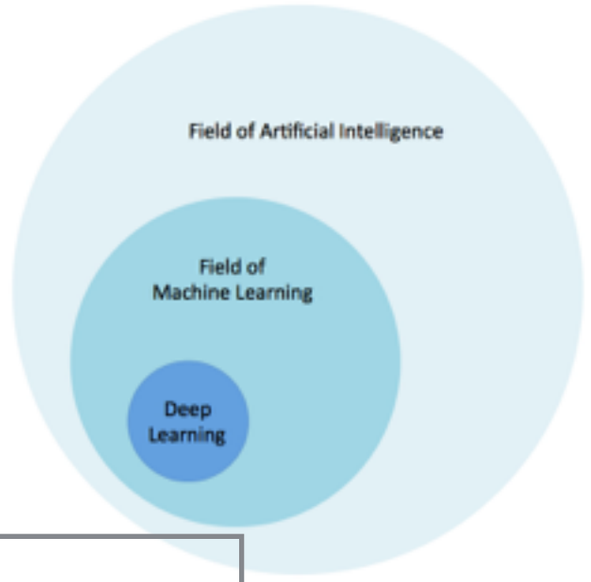
~~Light~~
Dark side of Deep Learning

Santiago VELASCO-FORERO
Centre for Mathematical Morphology
MINES ParisTech
PSL Research University
http://cmm.ensmp.fr/~velasco/

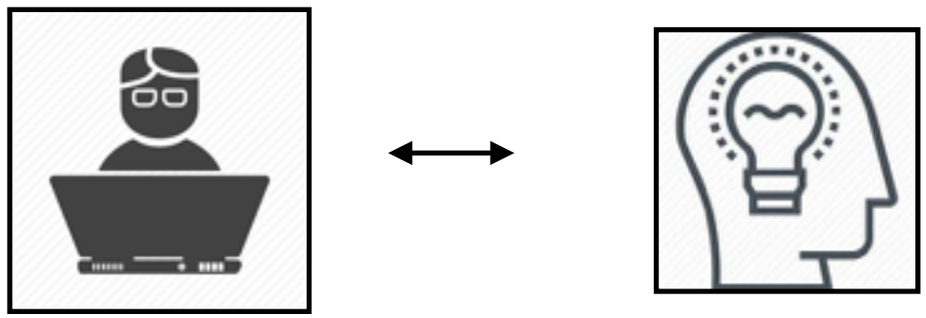
Machine Learning vs Deep Learning



Historic

Batch

Machine Learning



Input → Feature Extraction → Classification → Output

Deep Learning

Input → Feature Extraction+ Classification → Output

Excellent Performance on Supervised Learning

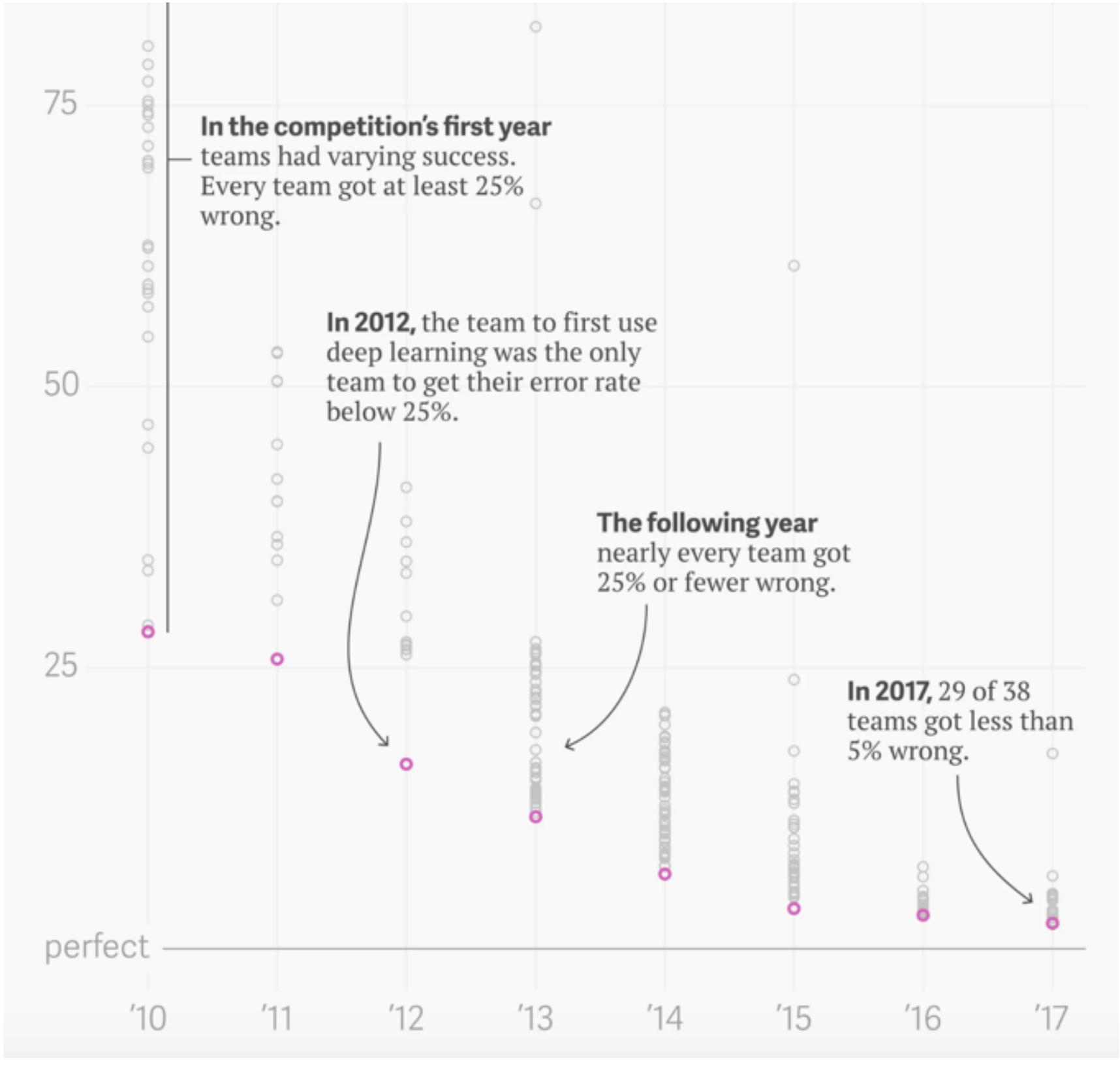
Machine Learning

MNIST
60000 images
10 classes

Deep Learning

ImageNet
1281167 images
1000 classes

Performance on Imagenet



David Yanofsky (Quartz)

What to do on Deep Learning?

Deep Learning

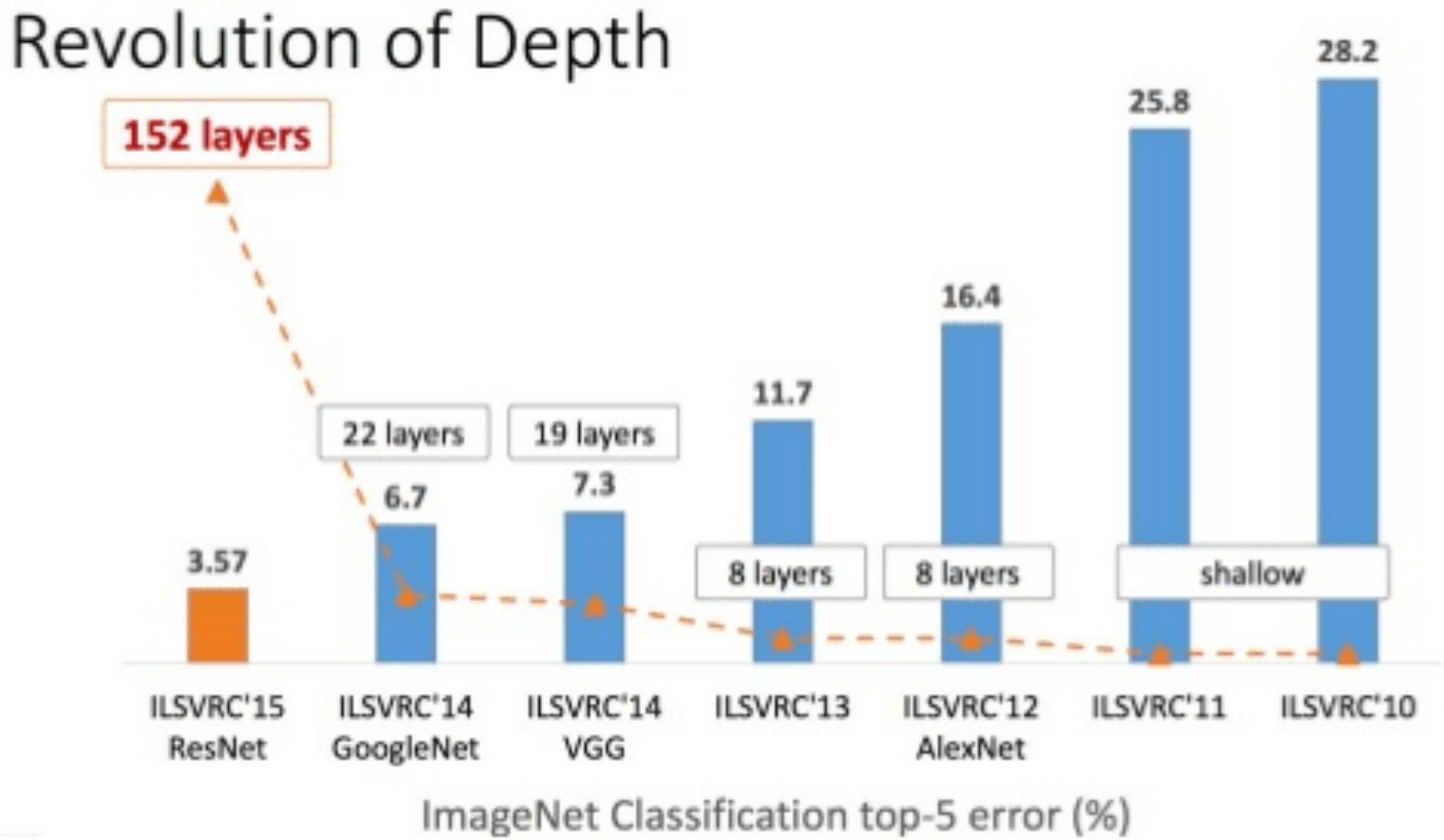
Input → Feature Extraction+ Classification → Output

Training Samples

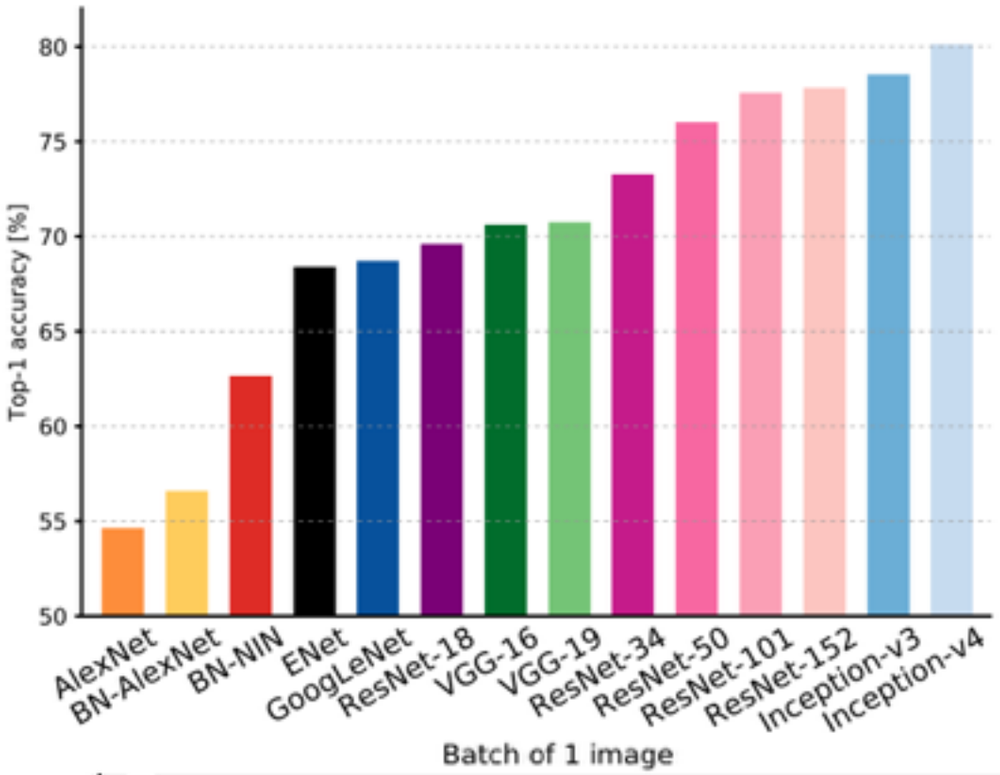
Optimization Algorithm

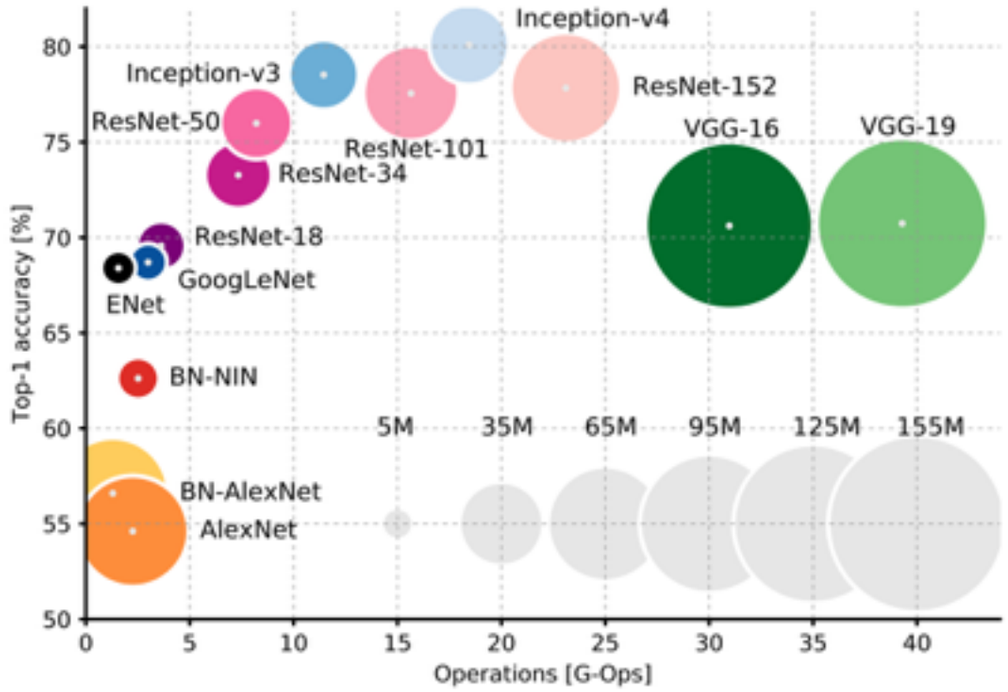
Loss Function

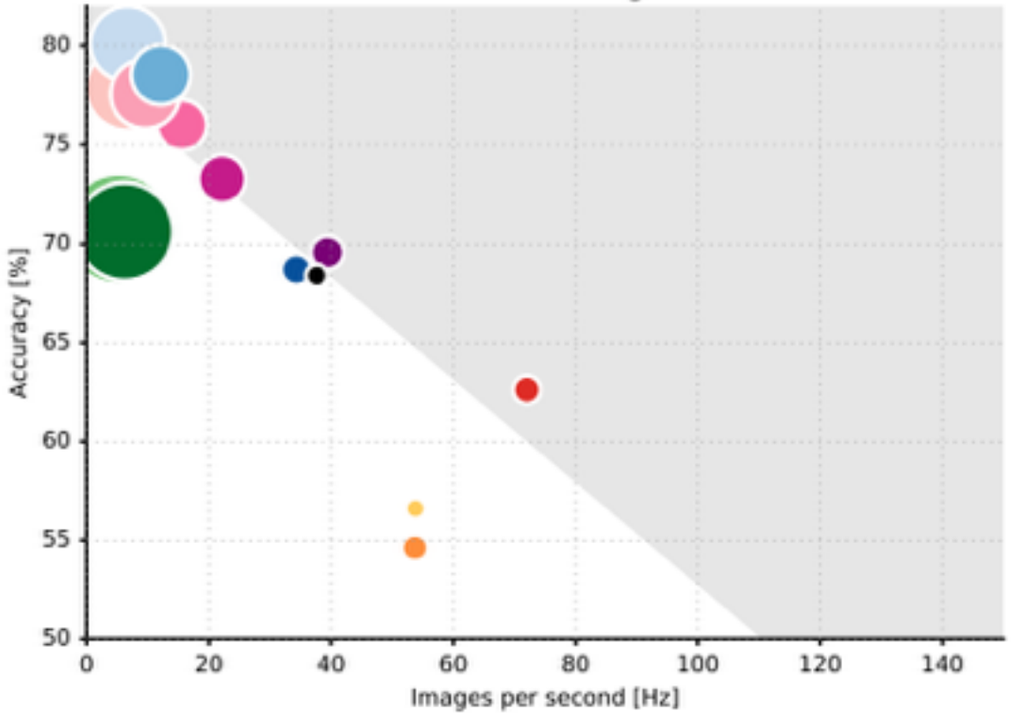
Architecture

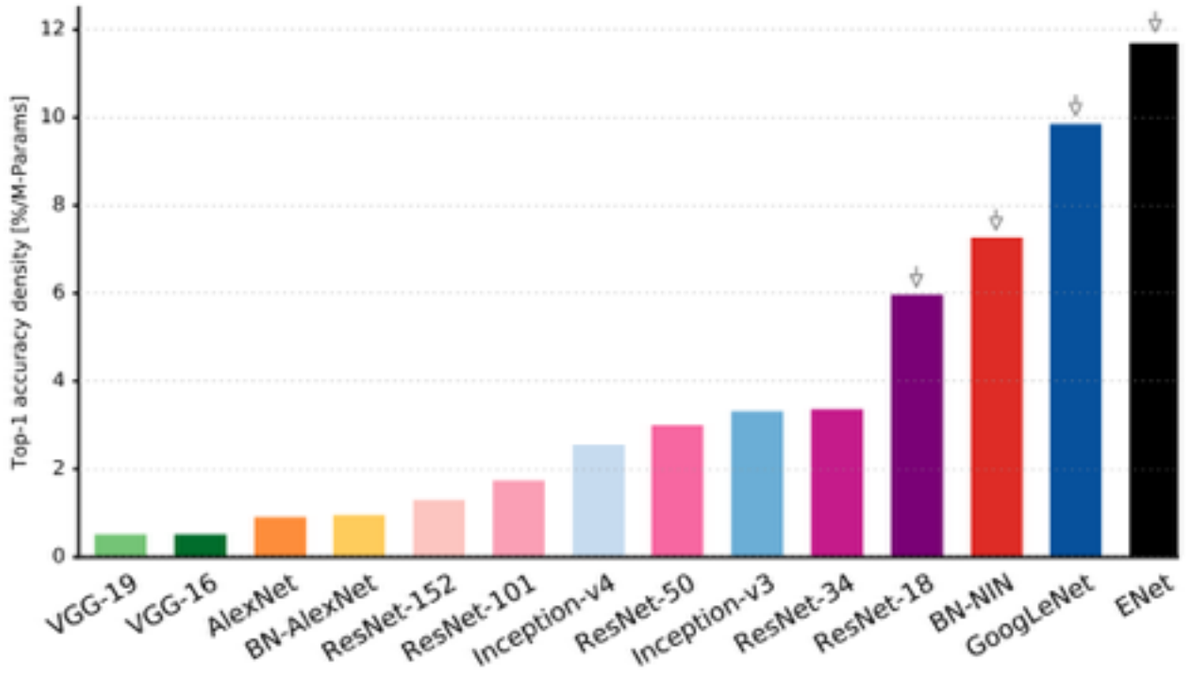

Revolution of Depth
152 layers
22 layers
19 layers
8 layers
8 layers
shallow
3.57
6.7
7.3
11.7
16.4
25.8
28.2
ILSVRC'15 ResNet
ILSVRC'14 GoogleNet
ILSVRC'14 VGG
ILSVRC'13
ILSVRC'12 AlexNet
ILSVRC'11
ILSVRC'10
ImageNet Classification top-5 error (%)

Architecture design









- Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello. "An analysis of deep neural network models for practical applications." arXiv preprint arXiv:1605.07678 (2016).

What to do on Deep Learning?

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

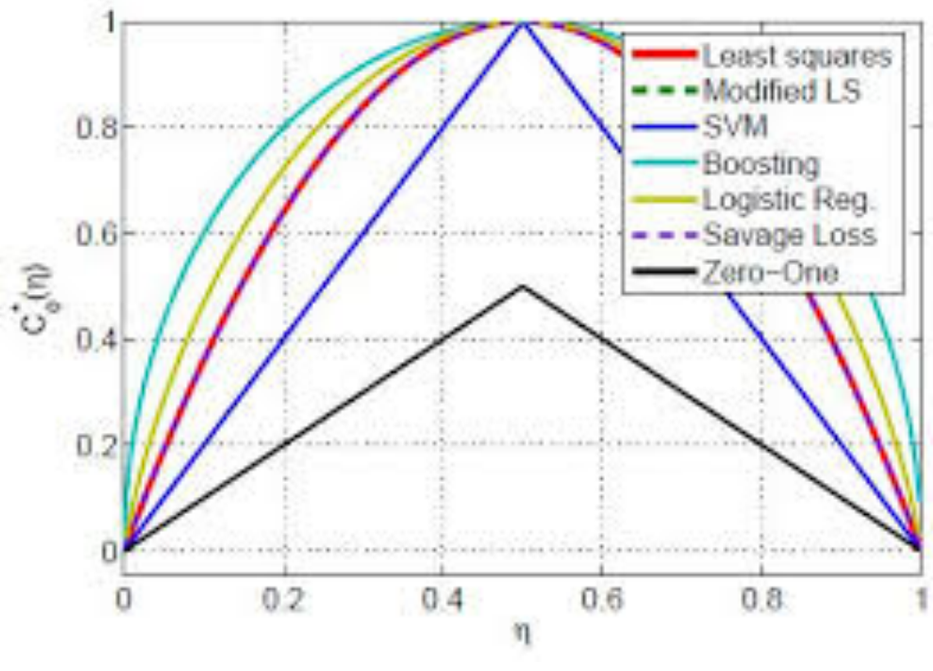
Optimization Algorithm

Loss Function

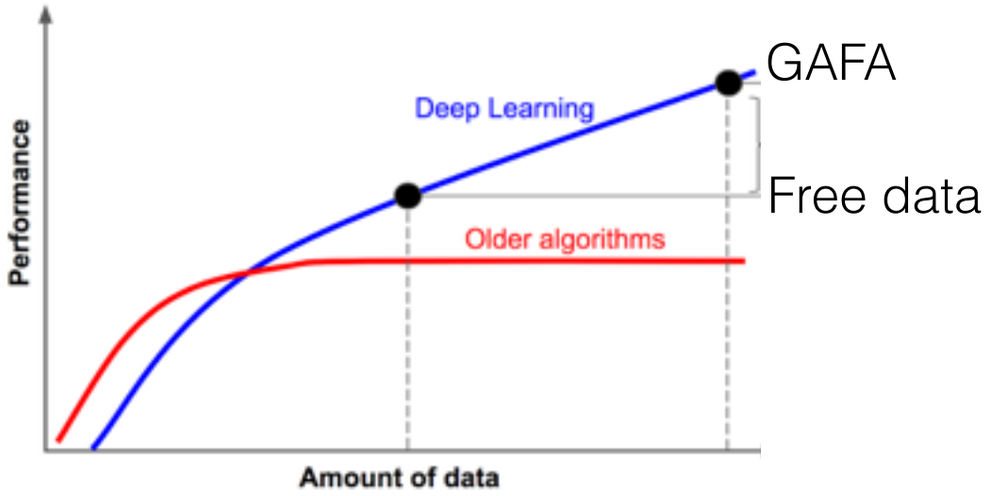
Architecture

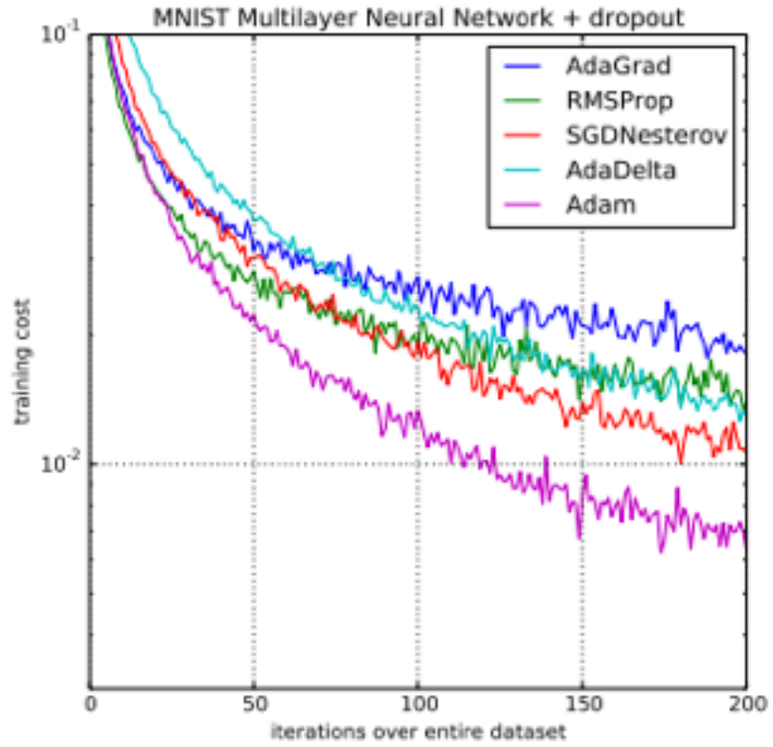
Loss function on manifolds

Penalization L1/L2



MINES ParisTech



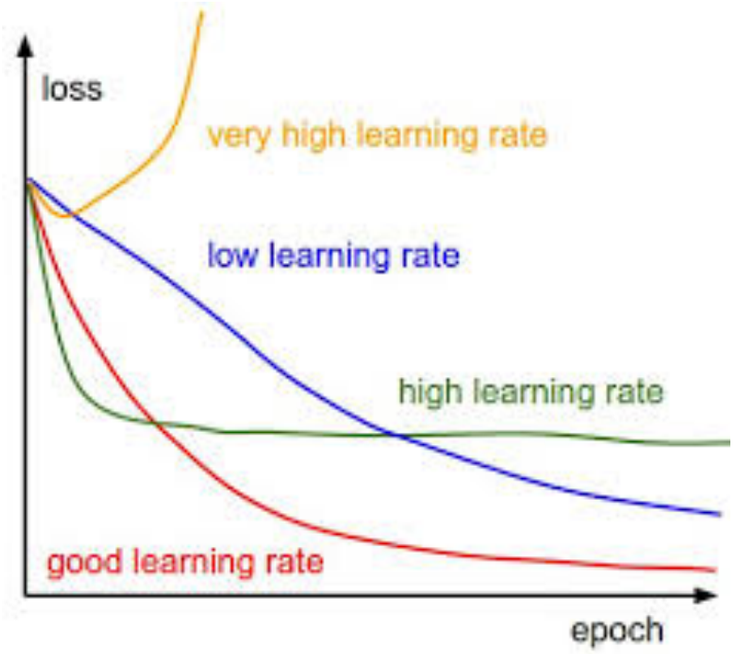


Training Samples

Optimization Algorithm

Loss Function

Architecture



$$\texttt{model}(\theta) := \theta^T x$$

$$\hat{y} := \texttt{model}(\hat{\theta})$$

$$\texttt{loss}(\texttt{model}(\hat{\theta})) = \frac{1}{2n}\sum_{i=1}^{n}(\hat{y}^i - y^i)^2$$

Vanilla Gradient Descent:

$$\begin{aligned}
\hat{\theta}^{(i+1)} &:= \hat{\theta}^{(i)} - \eta\frac{\partial \texttt{loss}(\hat{\theta})}{\partial\hat{\theta}} \\
&= \hat{\theta}^{(i)} - \eta\frac{\partial \texttt{loss}(\hat{\theta})}{\partial \texttt{model}(\hat{\theta})}\frac{\partial \texttt{model}(\hat{\theta})}{\partial\hat{\theta}} \\
&= \hat{\theta}^{(i)} - \eta\frac{\partial \texttt{loss}(\hat{\theta})}{\partial\hat{y}}\frac{\partial \texttt{model}(\hat{\theta})}{\partial\hat{\theta}} \\
&= \hat{\theta}^{(i)} - \eta\frac{1}{n}\sum_{i=1}^{n}(\hat{y}^i - y^i)x
\end{aligned}$$

Stochastic Gradient Descent:

```
for i in range(n):
```

$$\hat{\theta}^{(i+1)} = \hat{\theta}^{(i)} - \eta(\hat{y}^i - y^i)x$$

Minibatch Stochastic Gradient Descent: (Batch of size $|B|$)

```
for i in range( n / |B|):
```

$$\hat{\theta}^{(i+1)} = \hat{\theta}^{(i)} - \eta\frac{1}{|B|}\sum_{i=1}^{|B|}(\hat{y}^i - y^i)x$$

Basic math! (?)

Process as many data as you have (can?)

Memory requirement depend on batch size

GPU (Graphics processing unit) / TPU (Tensor processing unit)

New methods new challenges…

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

Optimization Algorithm

Loss Function

Architecture

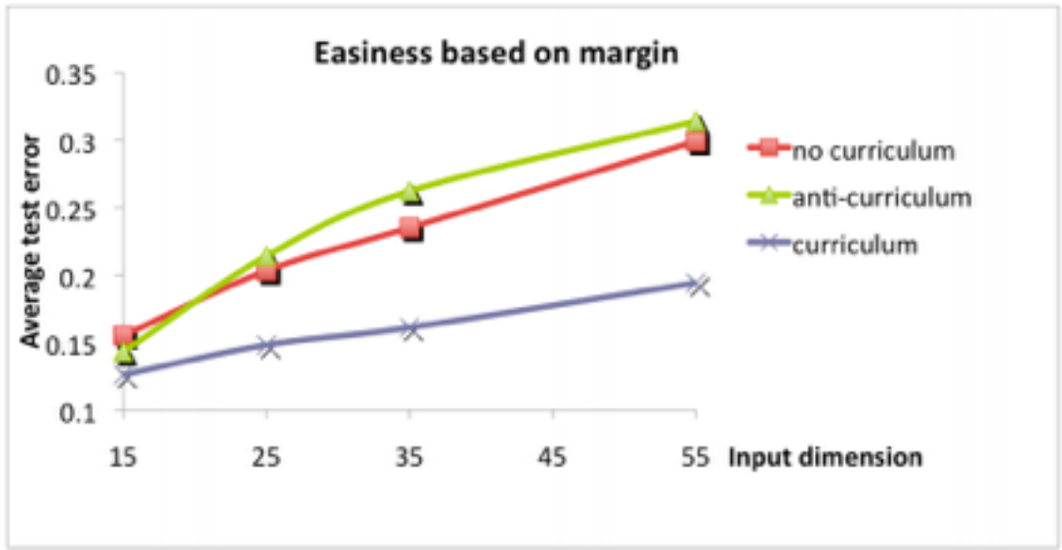
Order matters

More and more!

Excellent Interpolation but bad extrapolations

Curriculum learning

1. Cleaner examples may yield better generalization faster
2. Introducing gradually more difficult examples speeds-up training



Training Samples

Optimization Algorithm

Loss Function

Architecture

order matters

Bengio, Yoshua, et al. "Curriculum learning." Proceedings of the 26th annual international conference on machine learning. ACM, 2009.

Curriculum Learning

Data Augmentation

Transfer Learning

Transfer from Simulation

MINES ParisTech

Data augmentation

	Top-1 Accuracy	Top-5 Accuracy
Baseline	48.13 ± 0.42%	64.50 ± 0.65%
Flipping	49.73 ± 1.13%	67.36 ± 1.38%
Rotating	50.80 ± 0.63%	69.41 ± 0.48%
Cropping	**61.95 ± 1.01%**	**79.10 ± 0.80%**
Color Jittering	**49.57 ± 0.53%**	67.18 ± 0.42%
Edge Enhancement	49.29 ± 1.16%	66.49 ± 0.84%
Fancy PCA	49.41 ± 0.84%	**67.54 ± 1.01%**

Table 3: CNN training results for each DA method.

Caltech 101

Training Samples

Optimization Algorithm

Loss Function

Architecture

More and more!

Improving Deep Learning using Generic Data Augmentation

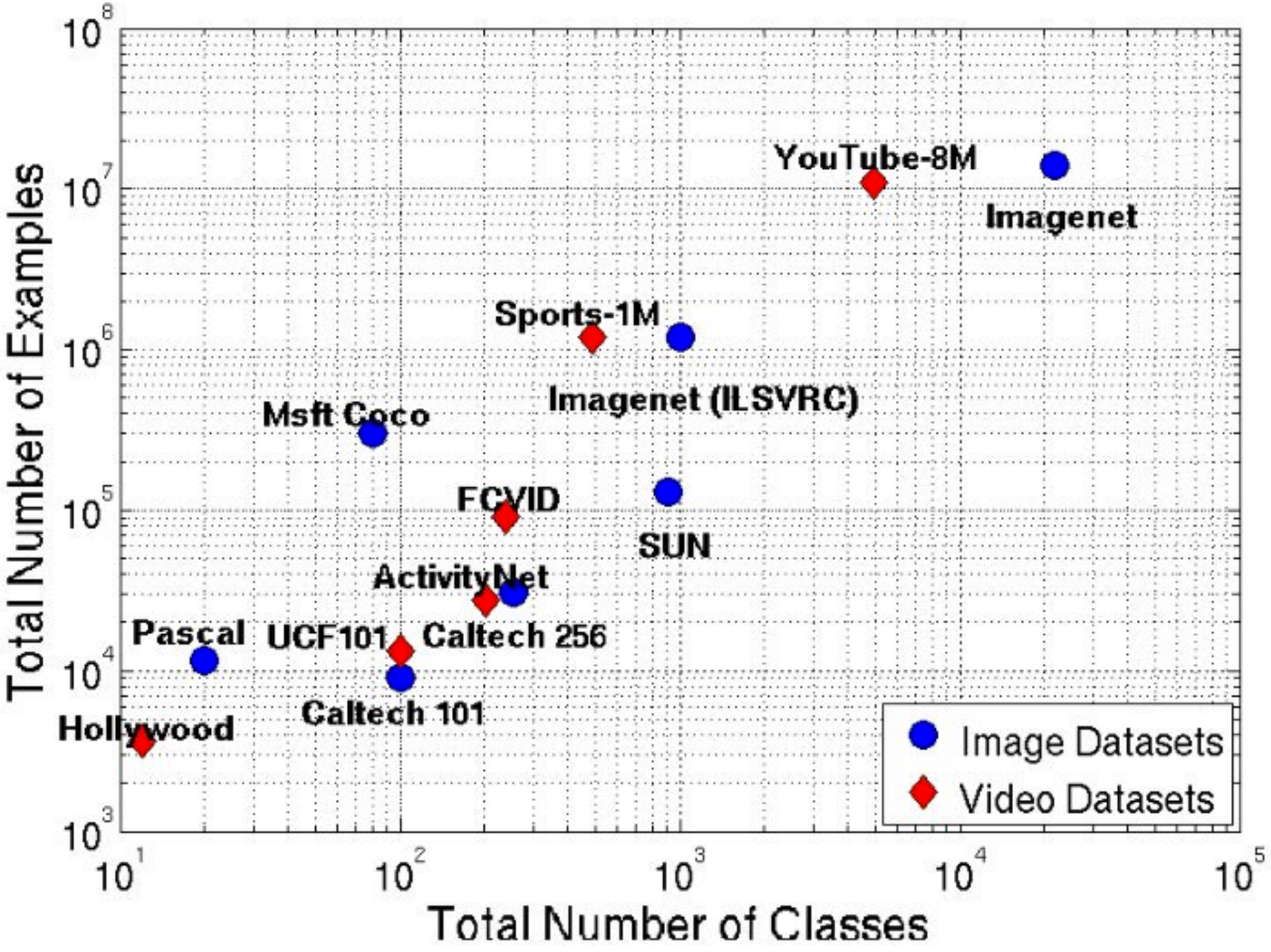
Curriculum Learning

Data Augmentation

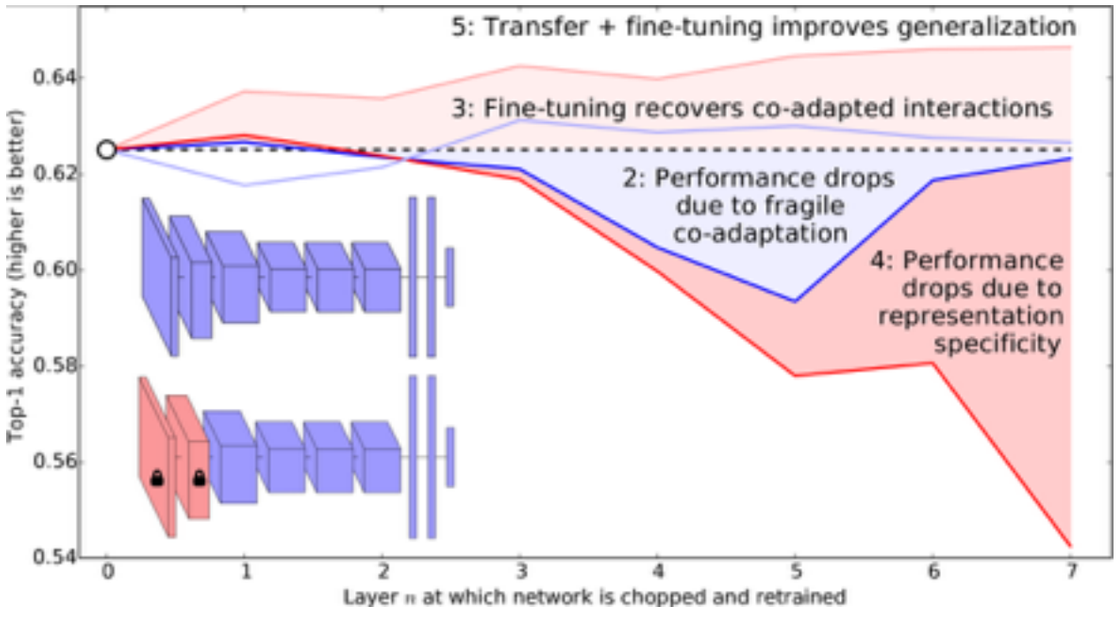
Transfer Learning

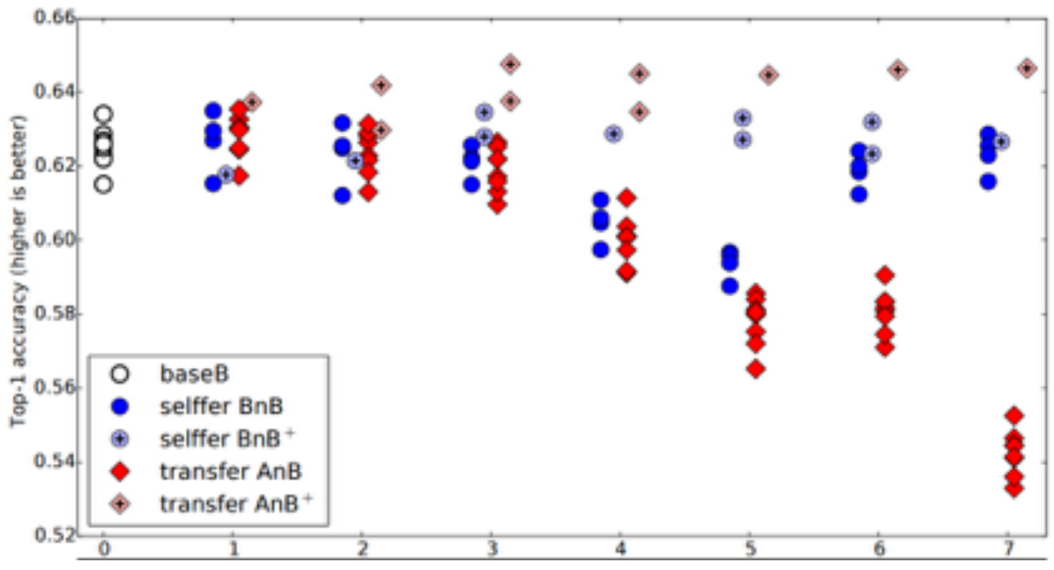
Transfer from Simulation

MINES ParisTech


Total Number of Examples
Total Number of Classes
10^8
10^7
10^6
10^5
10^4
10^3
10^1
10^2
10^3
10^4
10^5
YouTube-8M
Imagenet
Sports-1M
Imagenet (ILSVRC)
Msft Coco
FCVID
SUN
ActivityNet
Pascal
UCF101
Caltech 256
Caltech 101
Hollywood
Image Datasets
Video Datasets

Transfer Learning





Imagenet

Training Samples

Optimization Algorithm

Loss Function

Architecture

# of layers matter

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. In Advances in neural information processing systems (pp. 3320-3328).

Curriculum Learning

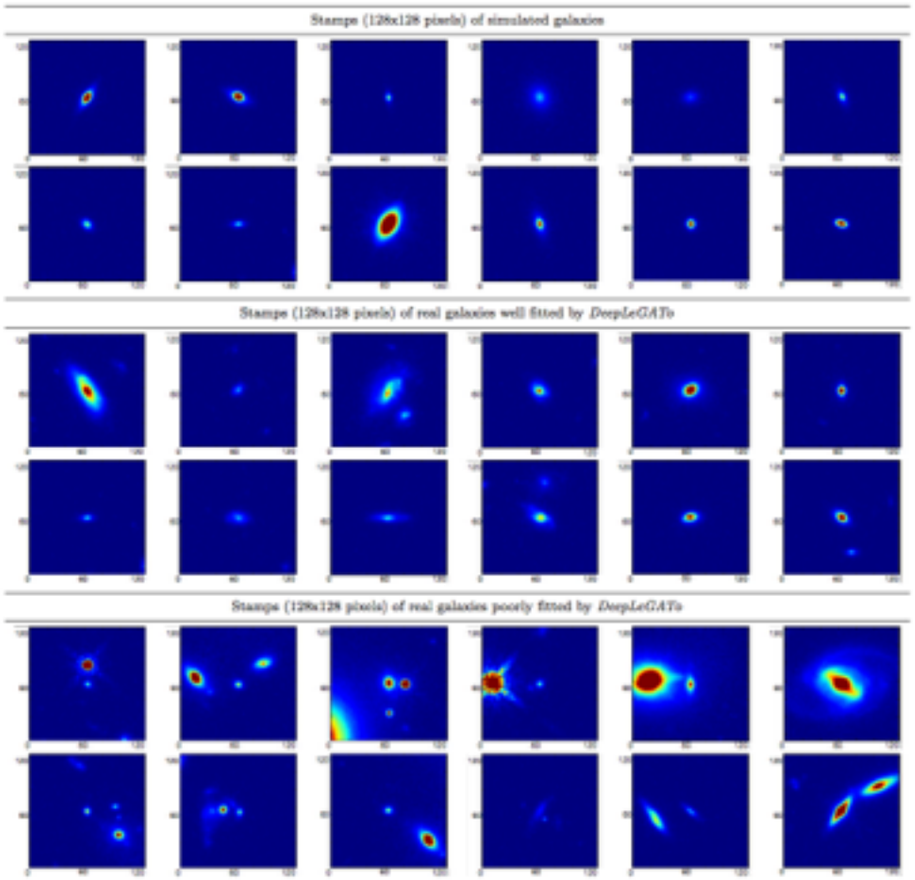
Data Augmentation

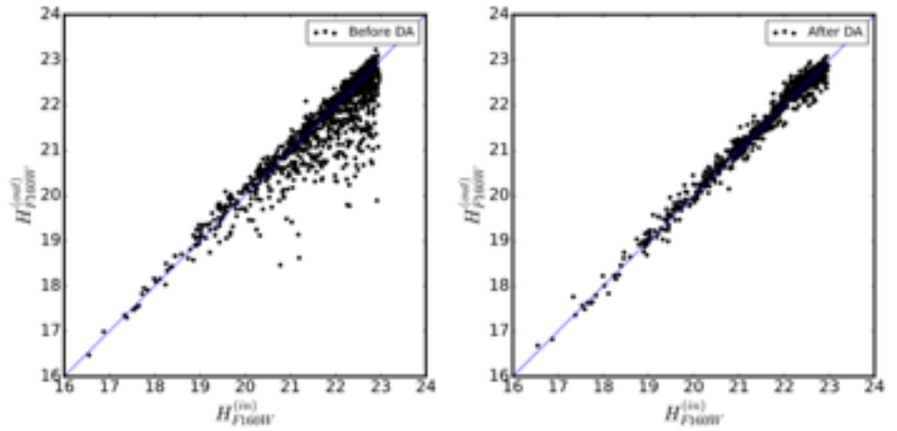
Transfer Learning

Transfer from Simulation

MINES ParisTech

Transfer from Simulations





(a) Before domain adaptation (BDA)

(b) After domain adaptation

R^2 Real data				
Parameter	Before TL	BTL isolated	After TL	2 GALFIT
Magnitude	0.795	0.979	0.980	0.984
Radius	-0.431	0.630	0.813	0.860
Sérsic index	-0.331	0.516	0.813	0.819
Axis ratio	0.773	0.915	0.934	0.914

Training Samples

Optimization Algorithm

Loss Function

Architecture

Domain adaptation

Curriculum Learning

Data Augmentation

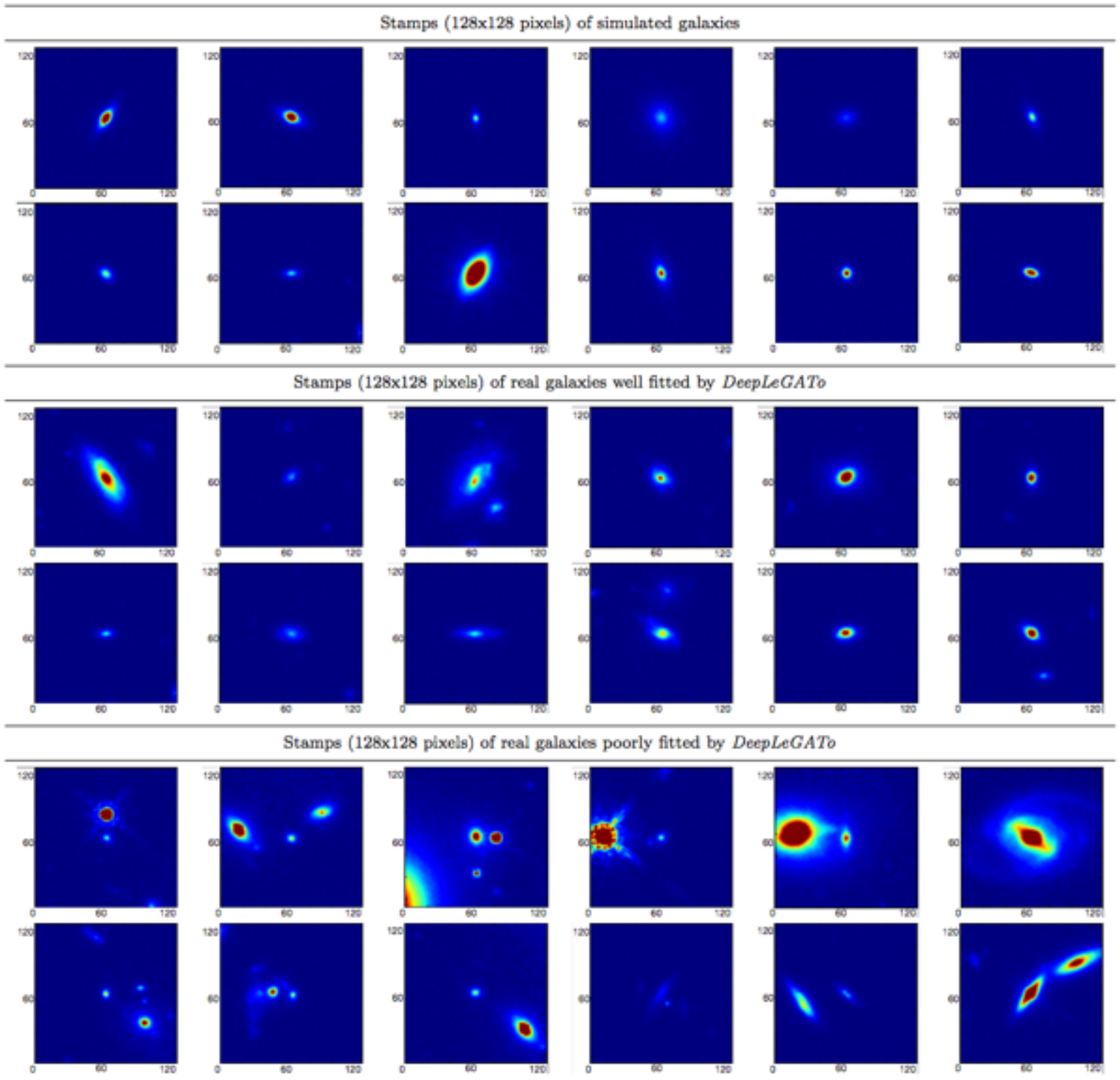
Transfer Learning

Transfer from Simulation

Tuccillo, D., et al. "Deep learning for galaxy surface brightness profile fitting." Monthly Notices of the Royal Astronomical Society 475.1 (2017): 894-909.

MINES ParisTech

Transfer from Simulations

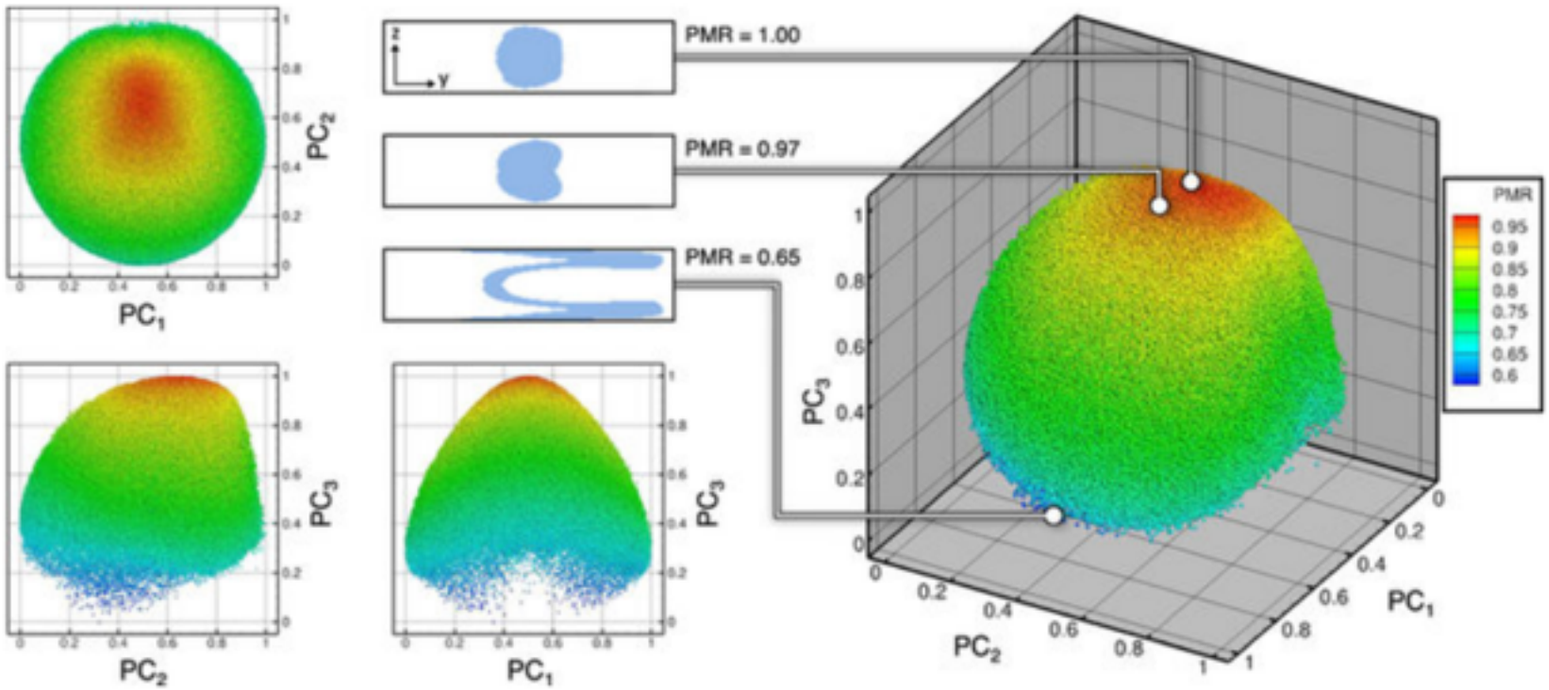


Tuccillo, D., et al. "Deep learning for galaxy surface brightness profile fitting." Monthly Notices of the Royal Astronomical Society 475.1 (2017): 894-909.

MINES ParisTech

Manifold Interpolation vs Extrapolation

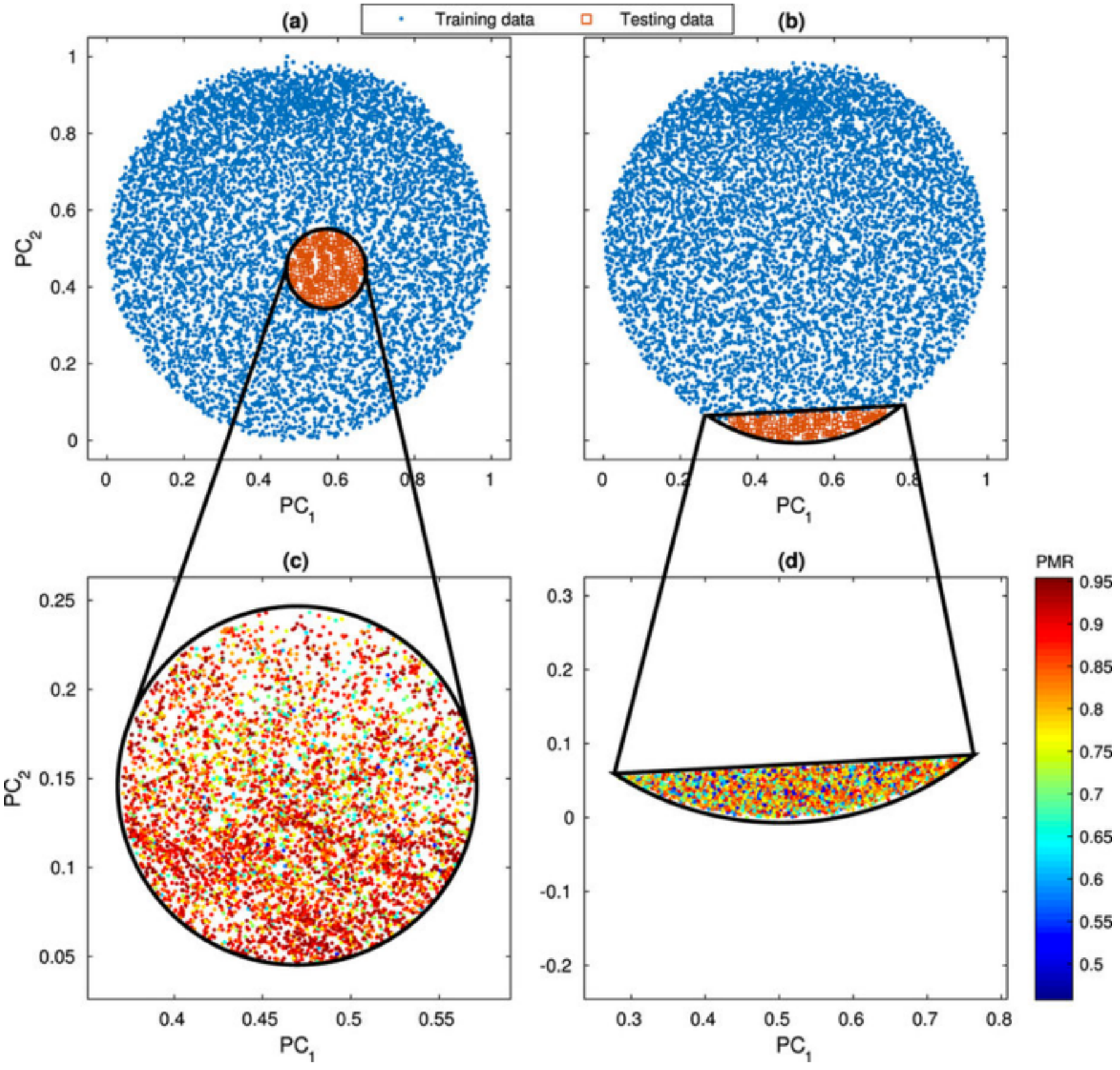
Figure 3: Visualization of the space $\mathcal{O}$ via PCA.



This illustration shows the projection of a training set of 150,000 images (chosen by random sampling of $\mathcal{S}$) onto its PCA-space, with the first three principal components as the axes PC_1, PC_2, and PC_3, and every point corresponds to an image in the dataset. Each point (image) is colored with a measure of similarity, the Pixel Match Rate (PMR), to a single image in the set (arbitrarily chosen as the image with the largest PC_3 coordinate). A high PMR (red) means the flow shape is more similar to the selected image, while a low PMR (blue) is less similar. Inserted are the selected target flow shape image (top) and two other sample images, each with their respective PMR value to the target image and indicated location in the PCA-space.

D. Stoecklein et al., Deep Learning for Flow Sculpting: Insights into Efficient Learning using Scientific Simulation Data, Scientific Reports volume 7, (2017)

Manifold Interpolation vs Extrapolation




DARK SIDE
LIGHTSIDE

Image Colorization

Grey Images

Color Images

Image-to-Image Translation with Conditional Adversarial Nets, Isola et al. 2017

Image-to-Image Translation

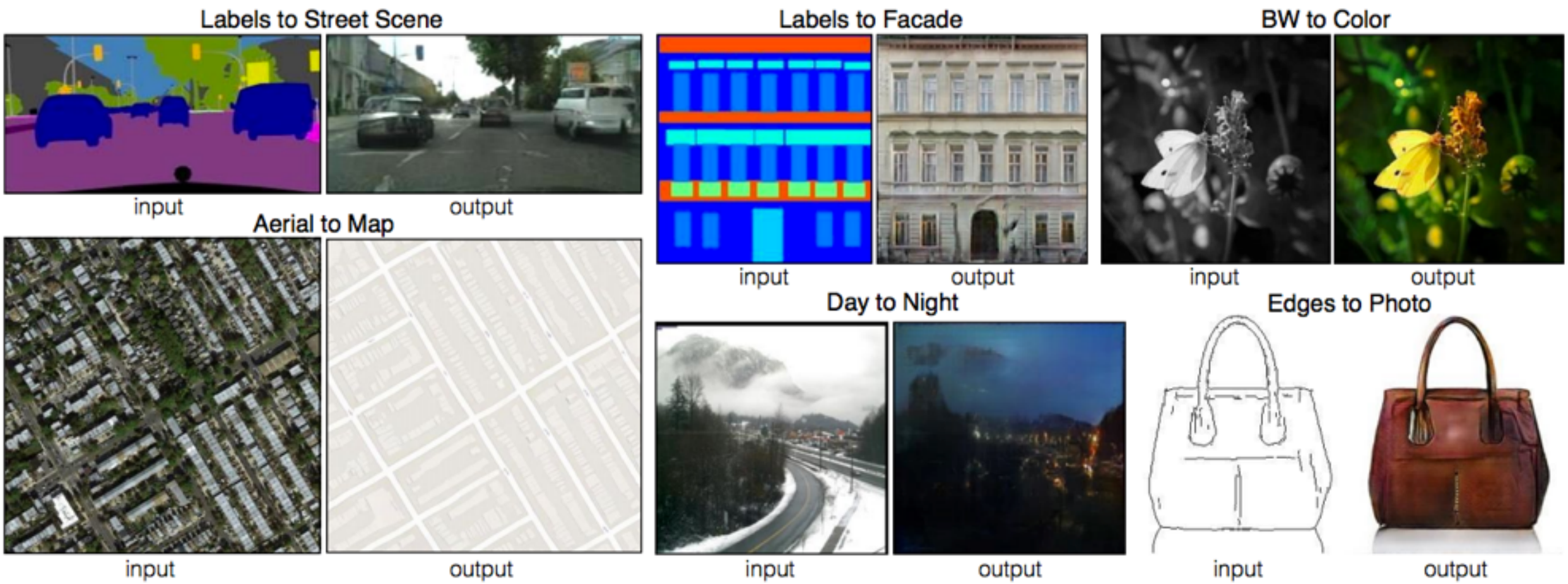
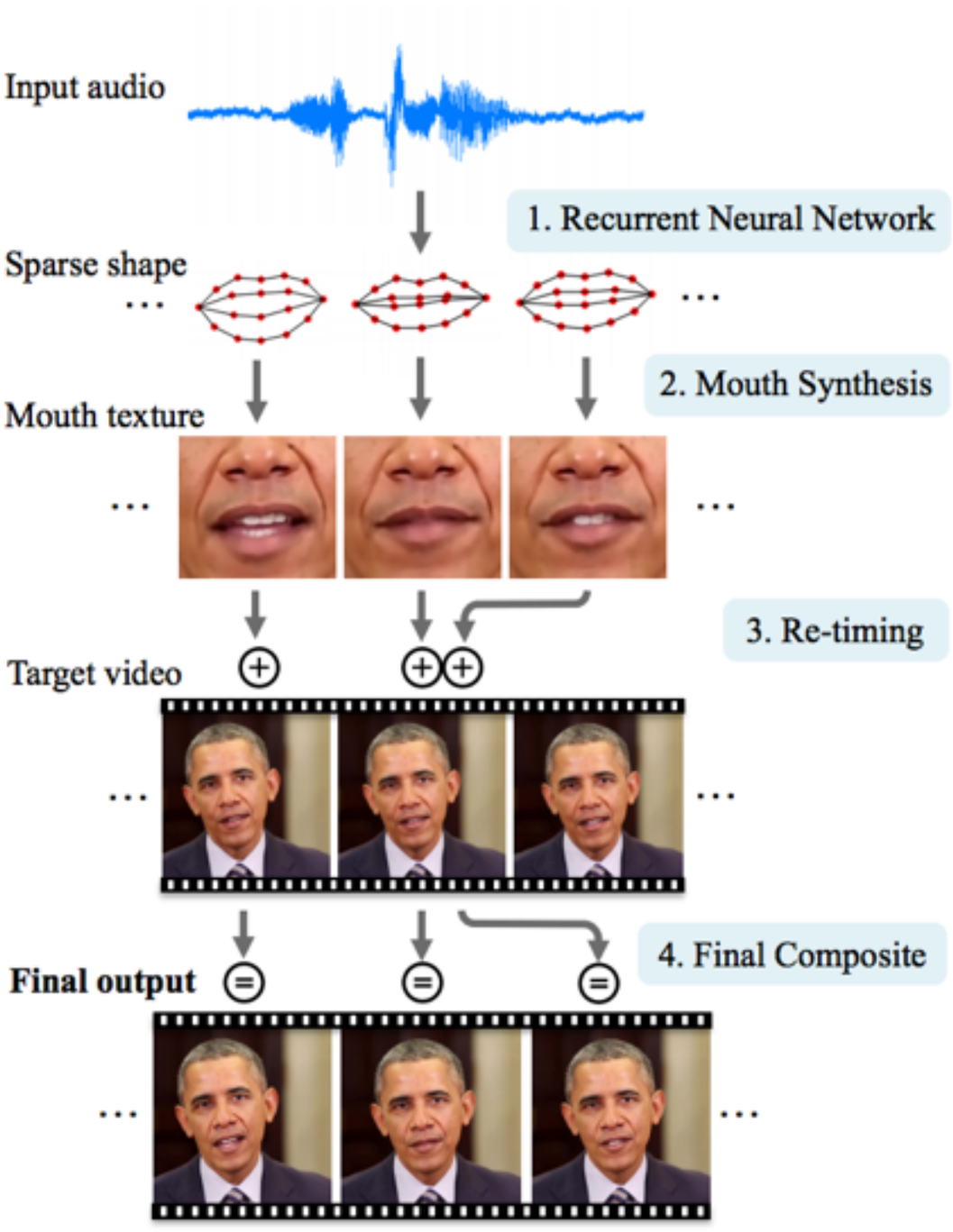


Image-to-Image Translation with Conditional Adversarial Nets, Isola et al. 2017

Ming-Yu Liu, Thomas Breuel, Jan Kautz,
UNIT: UNsupervised Image-to-image Translation Networks, NIPS 2017 Spotlight,

Learning Lip Sync from Audio



Input

Postprocessing

Output

https://www.youtube.com/watch?v=9Yq67CjDqvw

Can DL blackboxes be ~~creative~~ generative?

Deep Learning

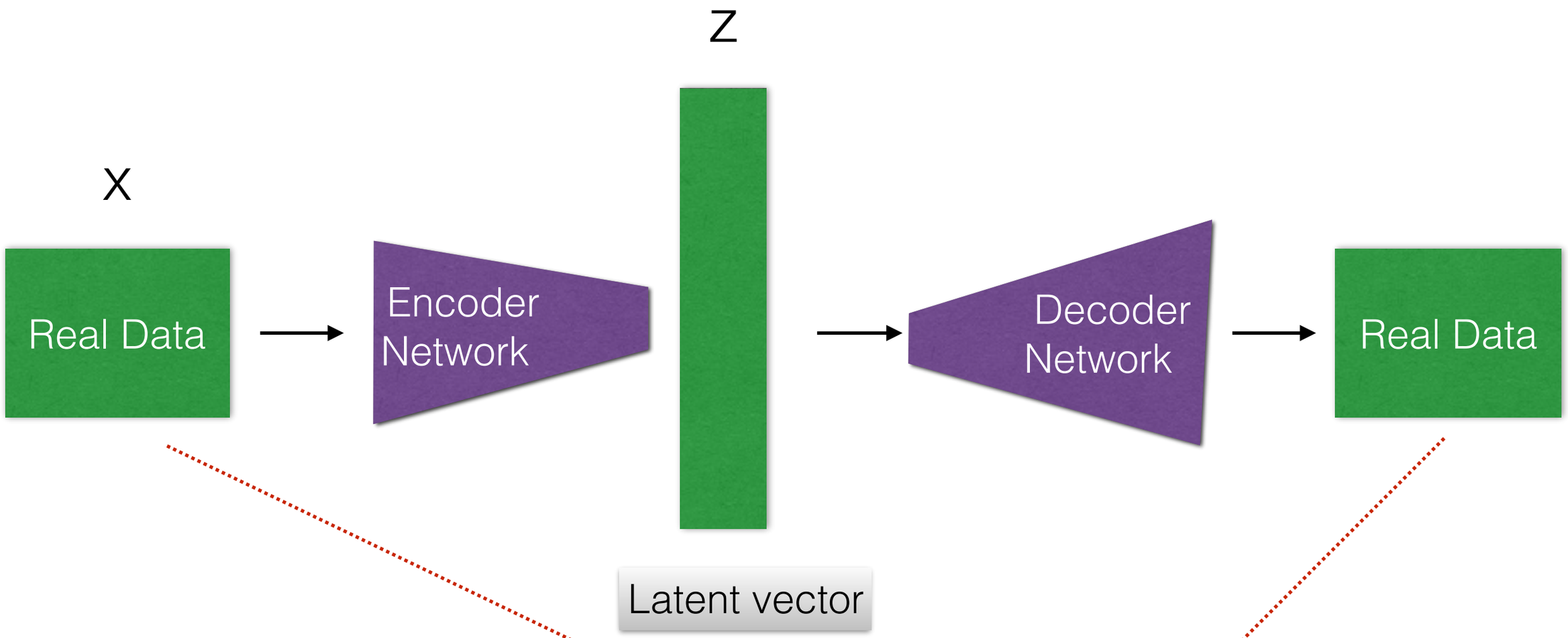
Input ⟶ Learn to Generate ⟶ New Data

Deep graphical models

Variational Autoencoder

Generative Adversarial Networks (GANs)

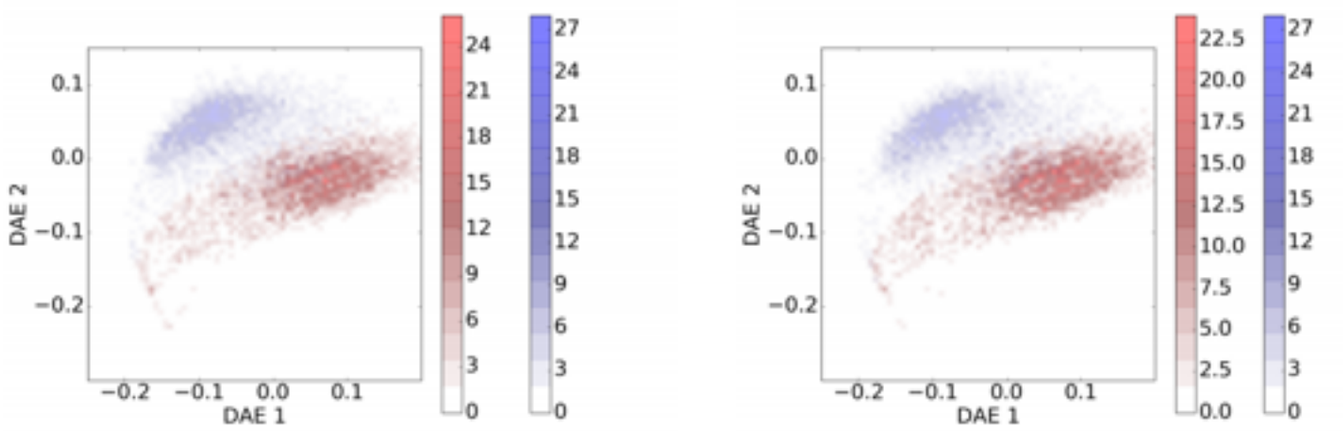
Autoencoder



Loss Function = Reconstruction Error

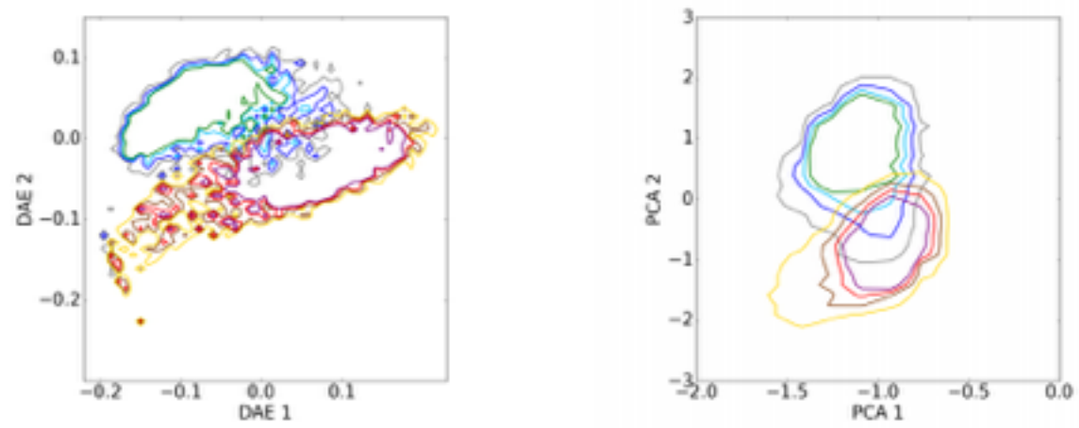
Deterministic Coding

Example of Autoencoders



(a) Training set (b) Test set

Fig. 1. Histogram of the training set and the test set on the DAE diagram. In the new plane, galaxies classified as star-forming galaxies in the COSMOS2015 catalog are depicted in blue, while the quiescent galaxies are shown in red.

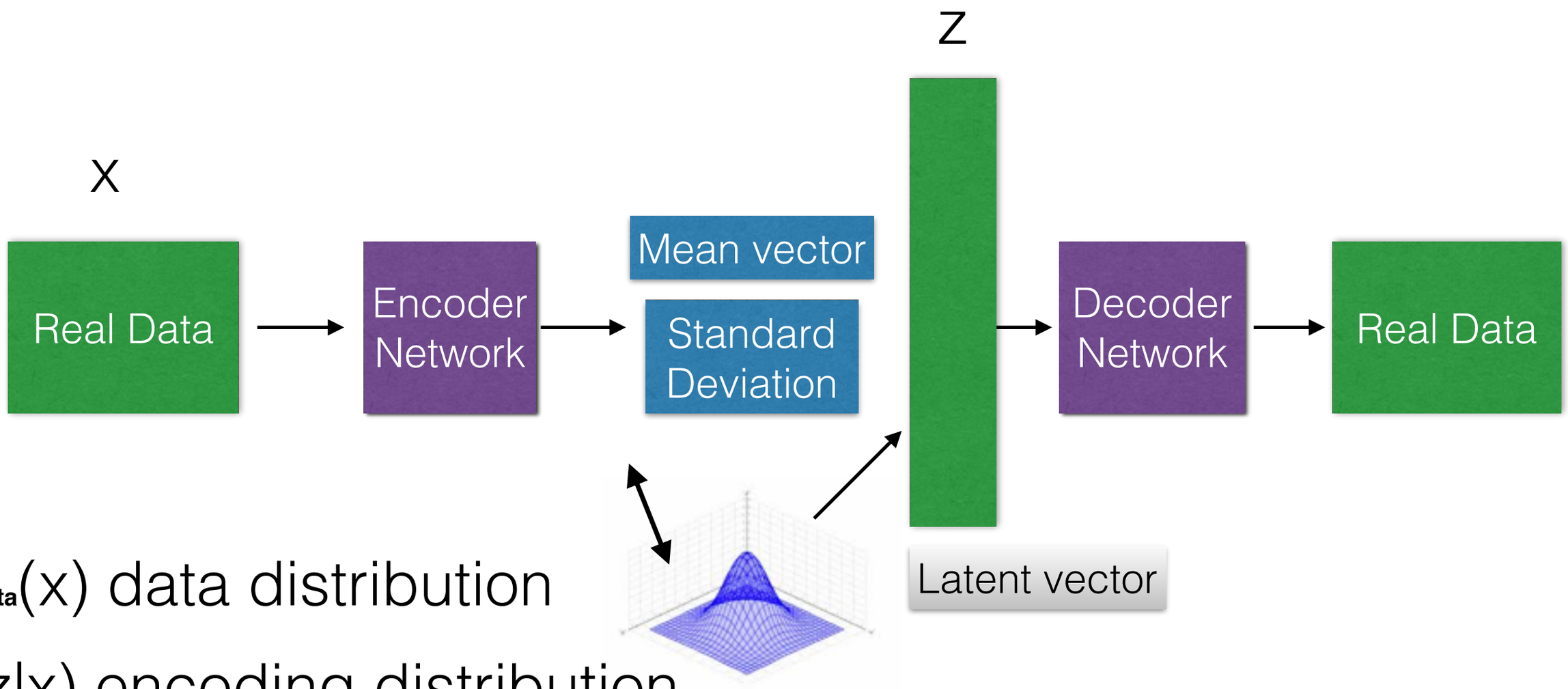


(a) DAE diagram (b) PCA diagram

Fig. 3. Contour levels for the test set on the DAE diagram (a) and on the PCA diagram (b). The curves are depicted from 95% of the population in grey to 70% in green for star-forming galaxies, and from yellow to purple for quiescent sources. These figures illustrate that the separation between the two classes of populations is clearer on the DAE diagram compared to the PCA diagram.

J. Frontera-Pons, F. Sureau, J. Bodin, E. Le Floc'h, Unsupervised feature-learning for galaxy SEDs with denoising autoencoders, Astronomy & Astrophysics, 2017

Variational Autoencoder



$p_{data}(x)$ data distribution

$q(z|x)$ encoding distribution

$p(x|z)$ decoding distribution

$p(x)$ model distribution $p(z)$ prior distribution

Doersch, Carl. "Tutorial on variational autoencoders." arXiv preprint arXiv:1606.05908 (2016).

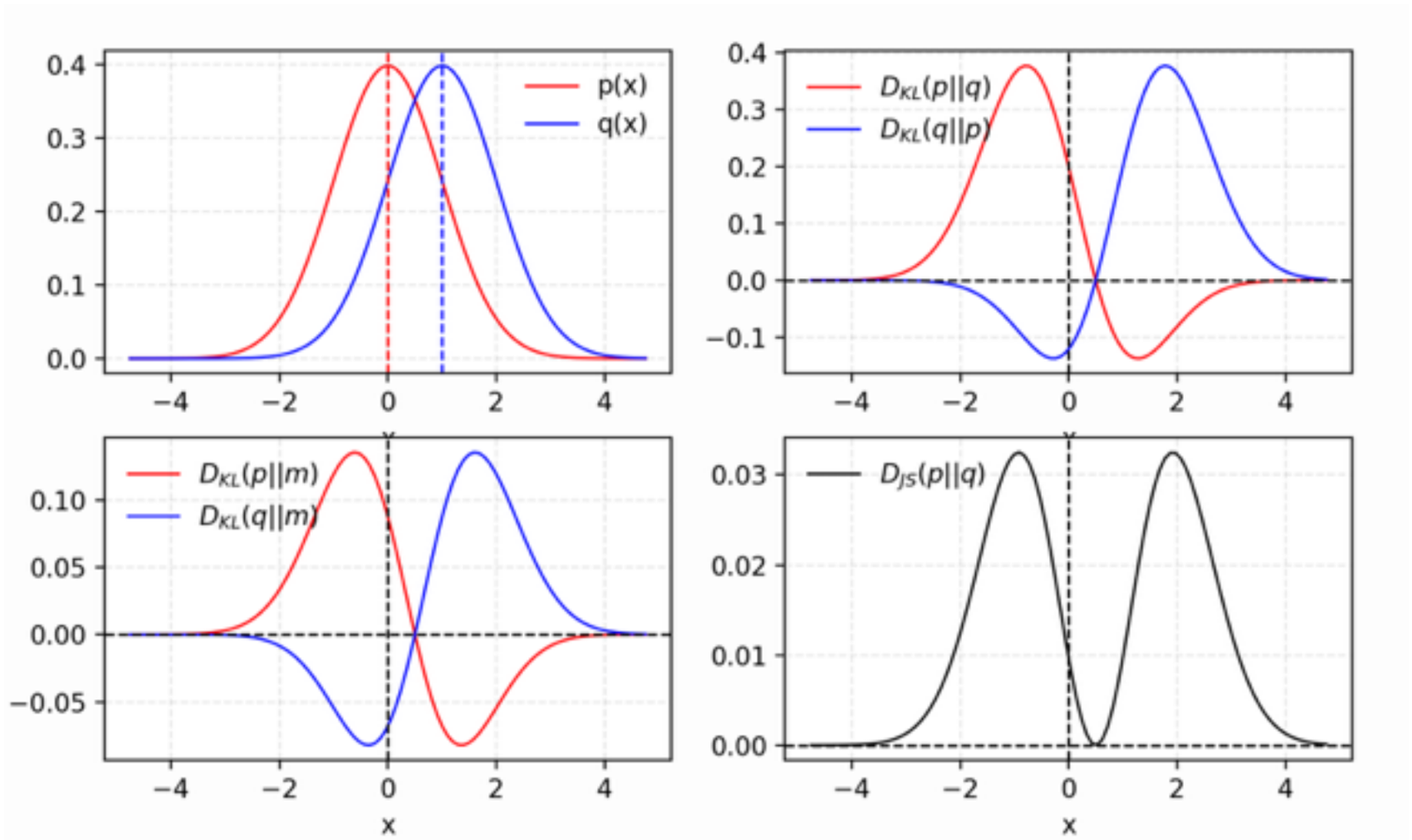
Divergences

KL (Kullback–Leibler) divergence

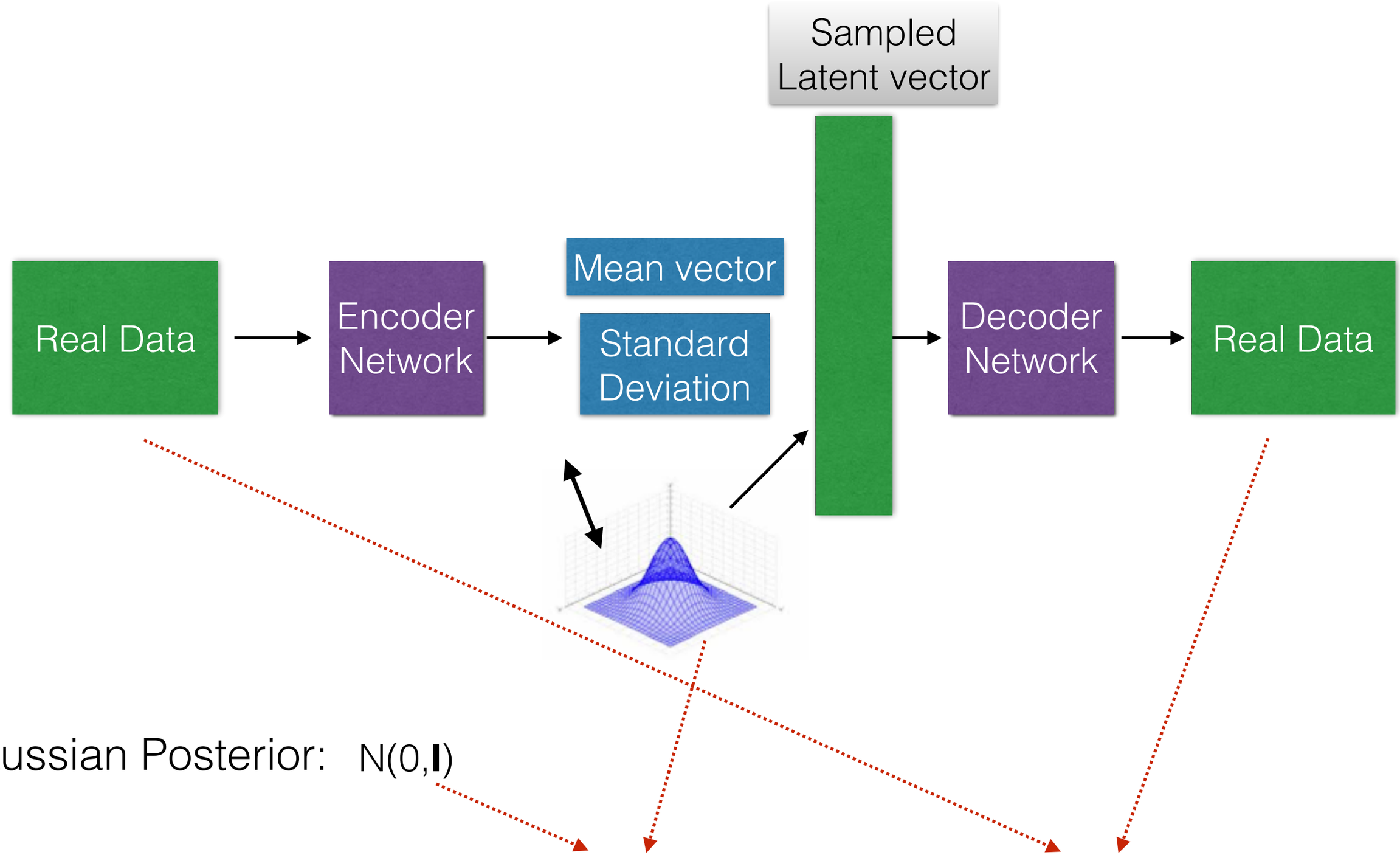
$$D_{KL}(p,q) = \int_x p(x) \log \frac{p(x)}{q(x)} dx$$

Jensen–Shannon Divergence

$$D_{JS}(p,q) = \frac{1}{2} D_{KL}\left(p, \frac{p+q}{2}\right) + \frac{1}{2} D_{KL}\left(q, \frac{p+q}{2}\right)$$

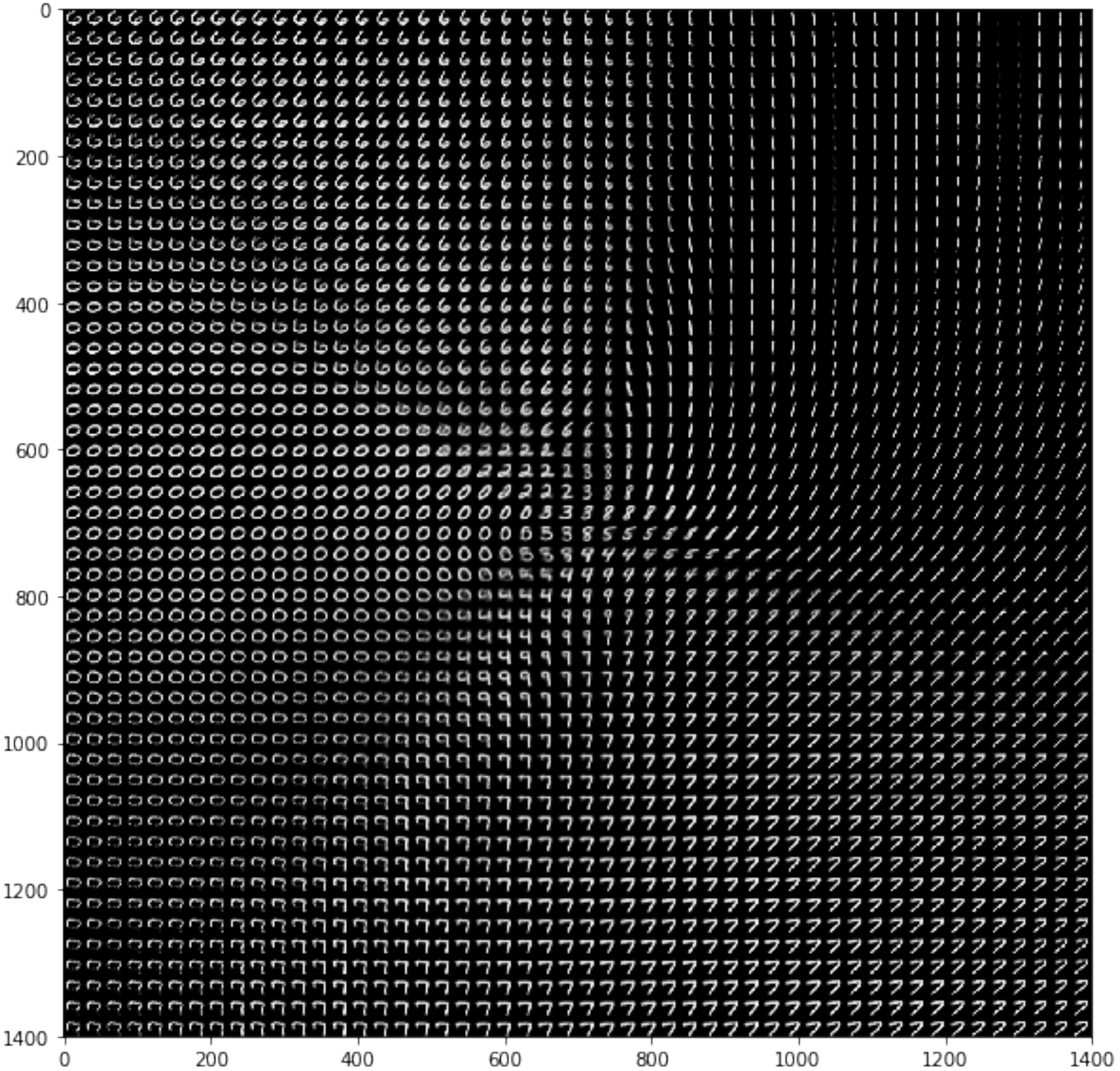


Variational Autoencoder

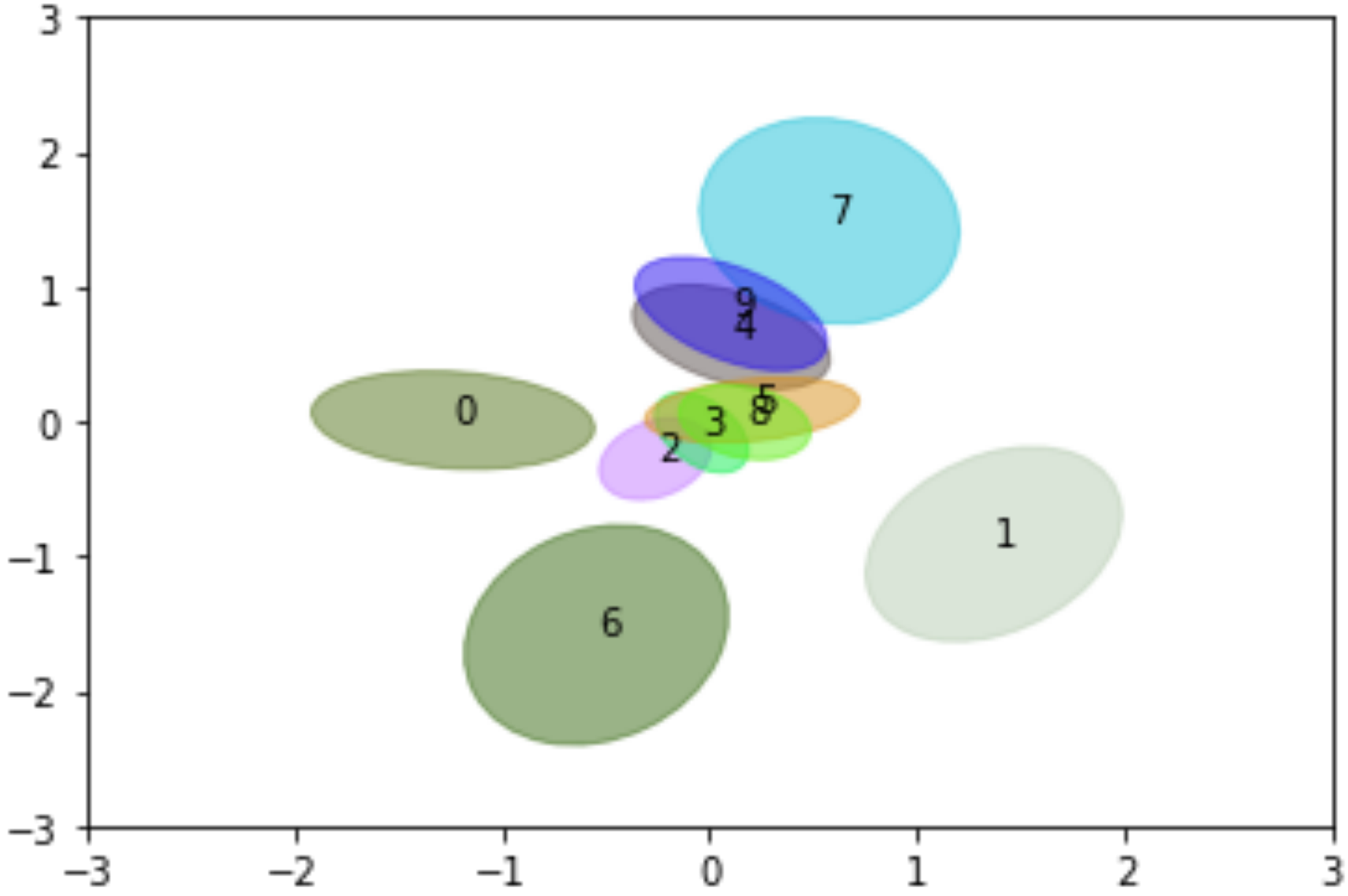


Doersch, Carl. "Tutorial on variational autoencoders." arXiv preprint arXiv:1606.05908 (2016).

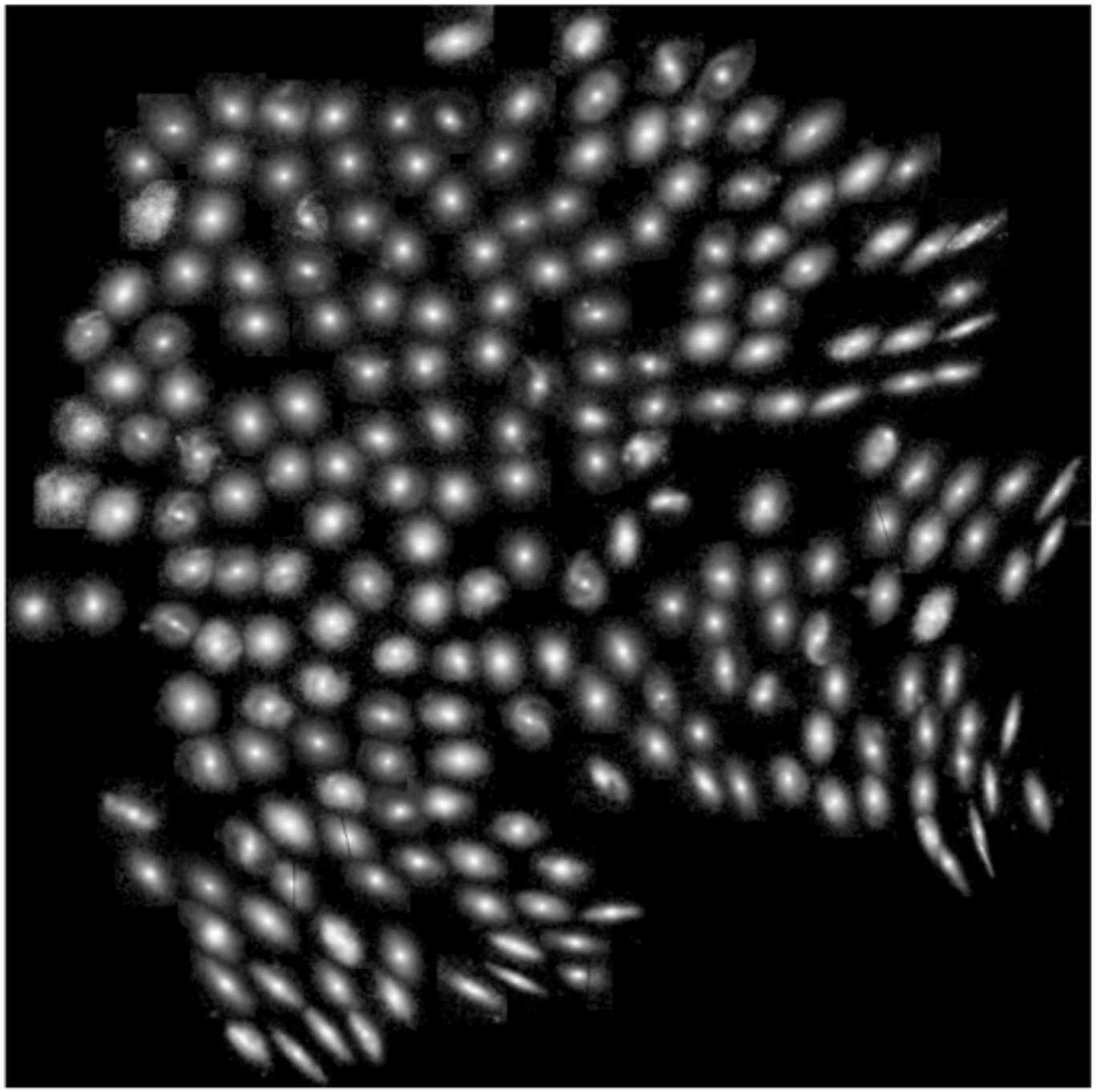
Visualization of Latent Space



Interpretable Latent Space



Variational Autoencoder



Galaxies embedded in two dimensions based on the means of their variational distributions, $f_\mu(x)$.

43,444 galaxy images for training from Sloan Digital Sky
Survey

Each image is cropped around one prominent galaxy.

Each image is downscaled to 69 × 69 pixels.

Jeffrey Regier et al, A deep generative model for astronomical images of galaxies

Variational Autoencoder

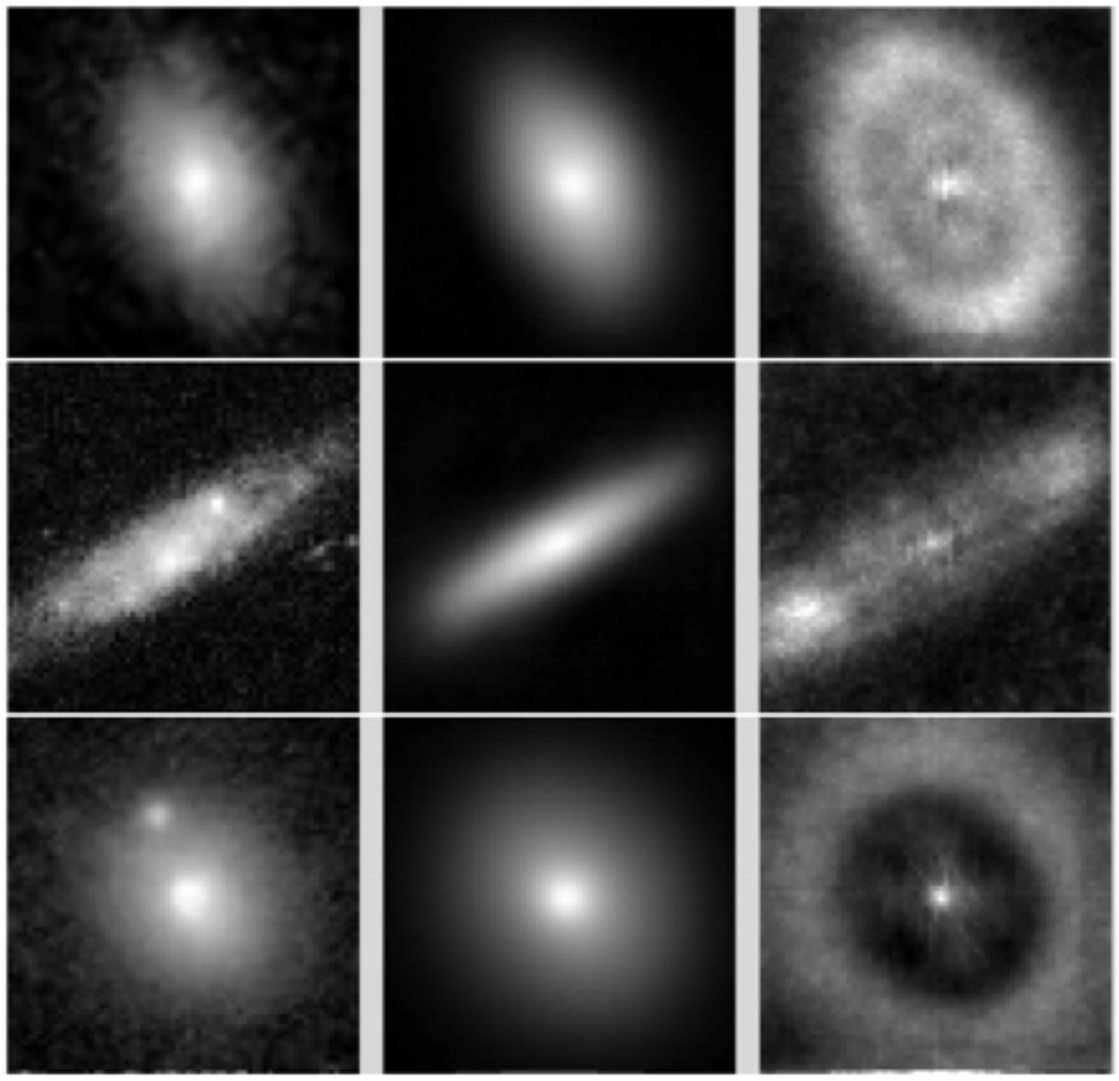
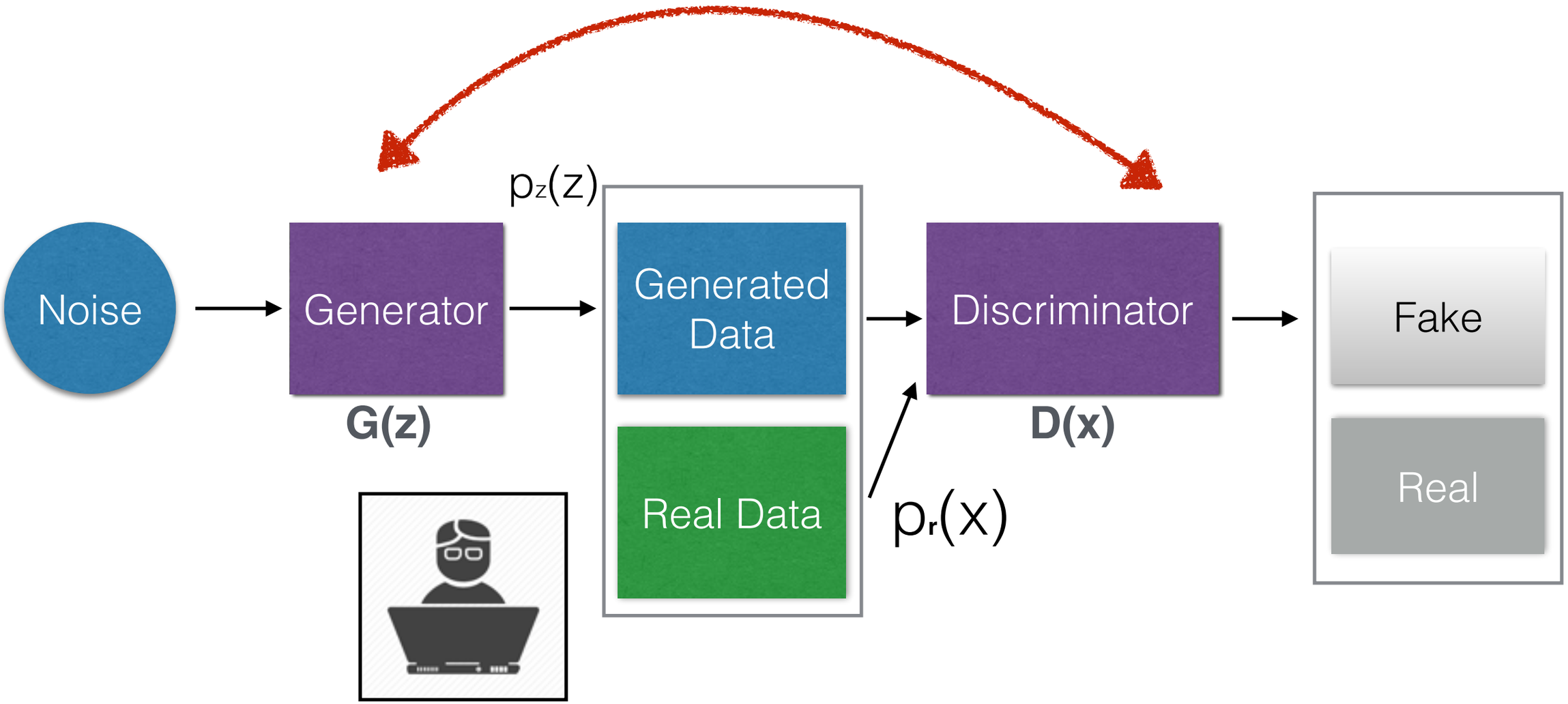


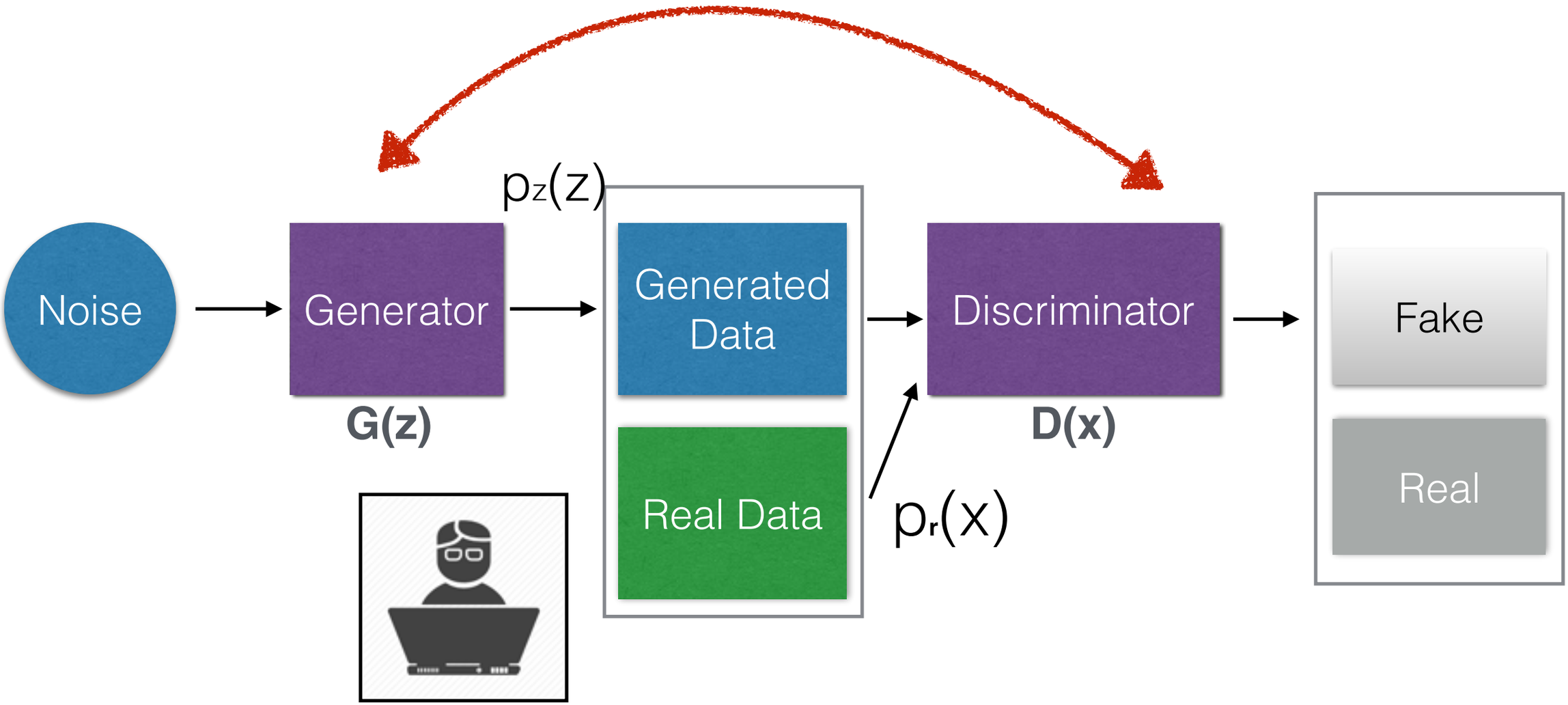
Figure 3: Each row corresponds to a different example from a test set. The left column shows the input x. The center column shows the output $f_\mu(z)$ for a z sampled from $\mathcal{N}\left(g_\mu(x), g_\Sigma(x)\right)$. The right column shows the output $f_\Sigma(z)$ for the same z.

Jeffrey Regier et al, A deep generative model for astronomical images of galaxies

Generative Adversarial Networks (GANs)



Generative Adversarial Networks (GANs)



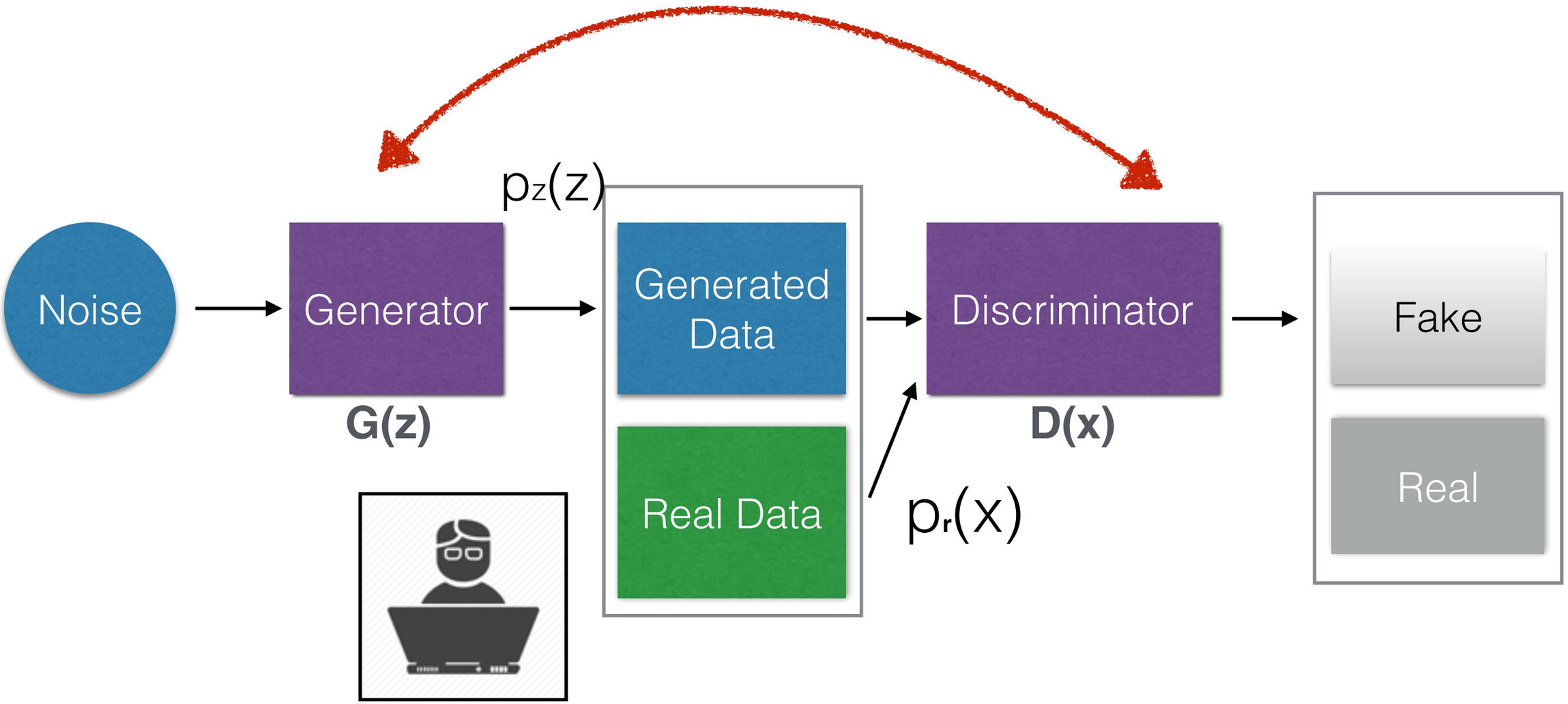
$$\mathbb{E}_{x \sim p_r(x)}[\log D(x)]$$

Decision over real data

$$\mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

Decision over fake data

Generative Adversarial Networks (GANs)



$$\mathbb{E}_{x \sim p_r(x)}[\log D(x)]$$

Decision over real data

$$\mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

Decision over fake data

$$\min_G \max_D L(D, G) = \mathbb{E}_{x \sim p_r(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$
$$= \mathbb{E}_{x \sim p_r(x)}[\log D(x)] + \mathbb{E}_{x \sim p_g(x)}[\log(1 - D(x)]$$

Minimax Game

Generative Adversarial Networks (GANs)

Optimal value of Generator?

Generative Adversarial Networks (GANs)

Optimal value of Generator?

$$p_g(x) = p_r(x)$$

Optimal value of Discriminator?

Generative Adversarial Networks (GANs)

Optimal value of Generator?

$$p_g(x) = p_r(x)$$

Optimal value of Discriminator?

$$\begin{aligned} \frac{\partial L(G,D)}{\partial D(x)} &= p_r(x)\frac{1}{D(x)} + p_g(x)\frac{1}{1-D(x)} \\ &= \frac{p_r(x) - (p_r(x) + p_g(x)D(x)}{D(x)(1-D(x))} \end{aligned}$$

Setting $\frac{\partial L(G,D)}{\partial D(x)} = 0$ then $D^{opt}(x) = \frac{p_r(x)}{p_r(x)+p_g(x)}$

Generative Adversarial Networks (GANs)

Global Optimal?

$$L(G, D) = \int_x p_r(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx$$

$$\begin{aligned} L(G^{opt}, D^{opt}) &= \int_x p_r(x) \log(D^{opt}(x)) + p_g(x) \log(1 - D^{opt}(x)) dx \\ &= \int_x p_r(x) \log(1/2) + p_r(x) \log(1/2) dx \\ &= 2 \log(1/2) \end{aligned}$$

Generative Adversarial Networks (GANs)

Bad news: GANs are hard to train

Some difficulties:

Discriminator collapse close to 1. (Change Initialization)
Problems with counting (Use conditional)
Perspective Problem (Data augmentation)
….

How to Train a GAN? Tips and tricks to make GANs work

https://github.com/soumith/ganhacks

Training simultaneously to find a Nash equilibrium to a two-player non-cooperative game

Updating the gradient of both models do not guarantee a convergence.

Player 1: Minimize f(x)= xy

f’(x)=y

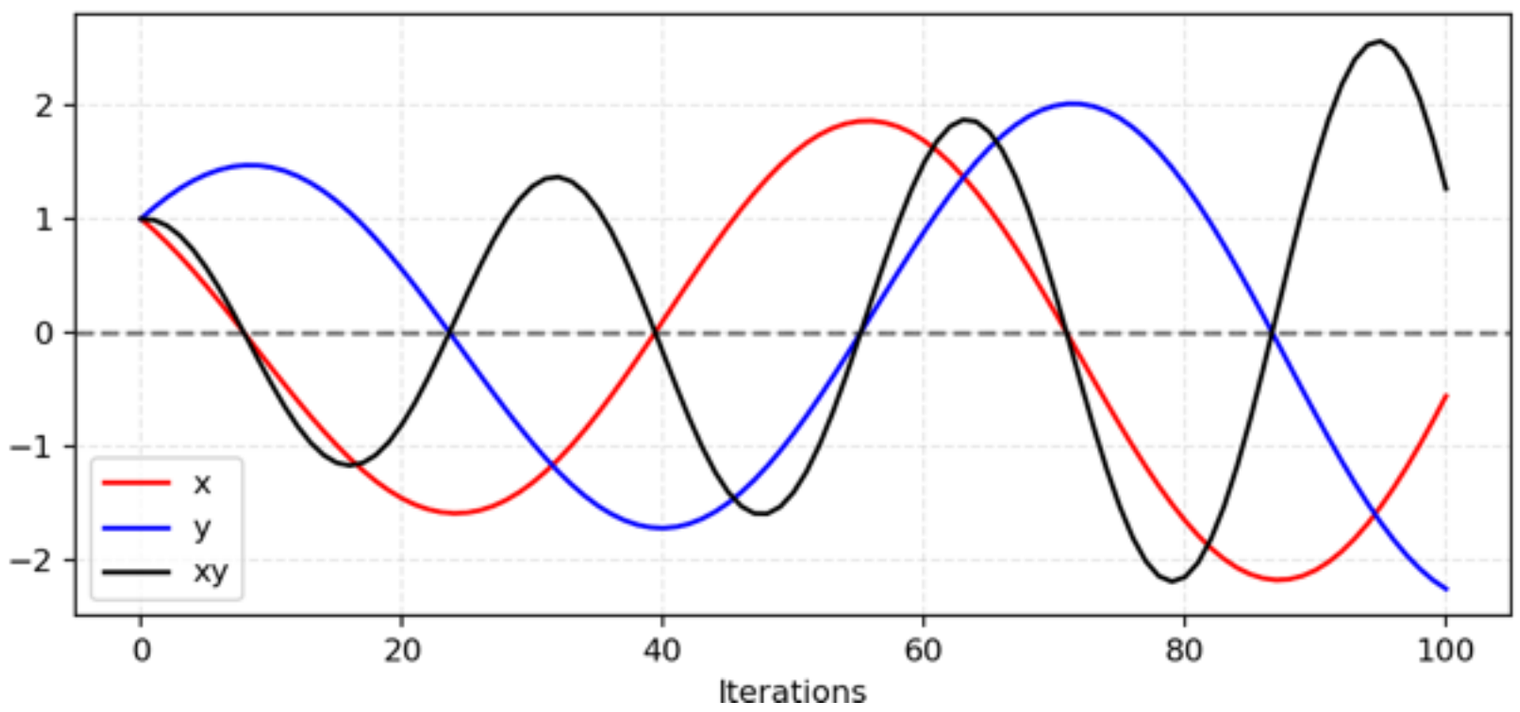
Player 2: Minimize g(y)= -xy

g’(y)=-x

Gradient Update:

x_{new}= x - \tao y

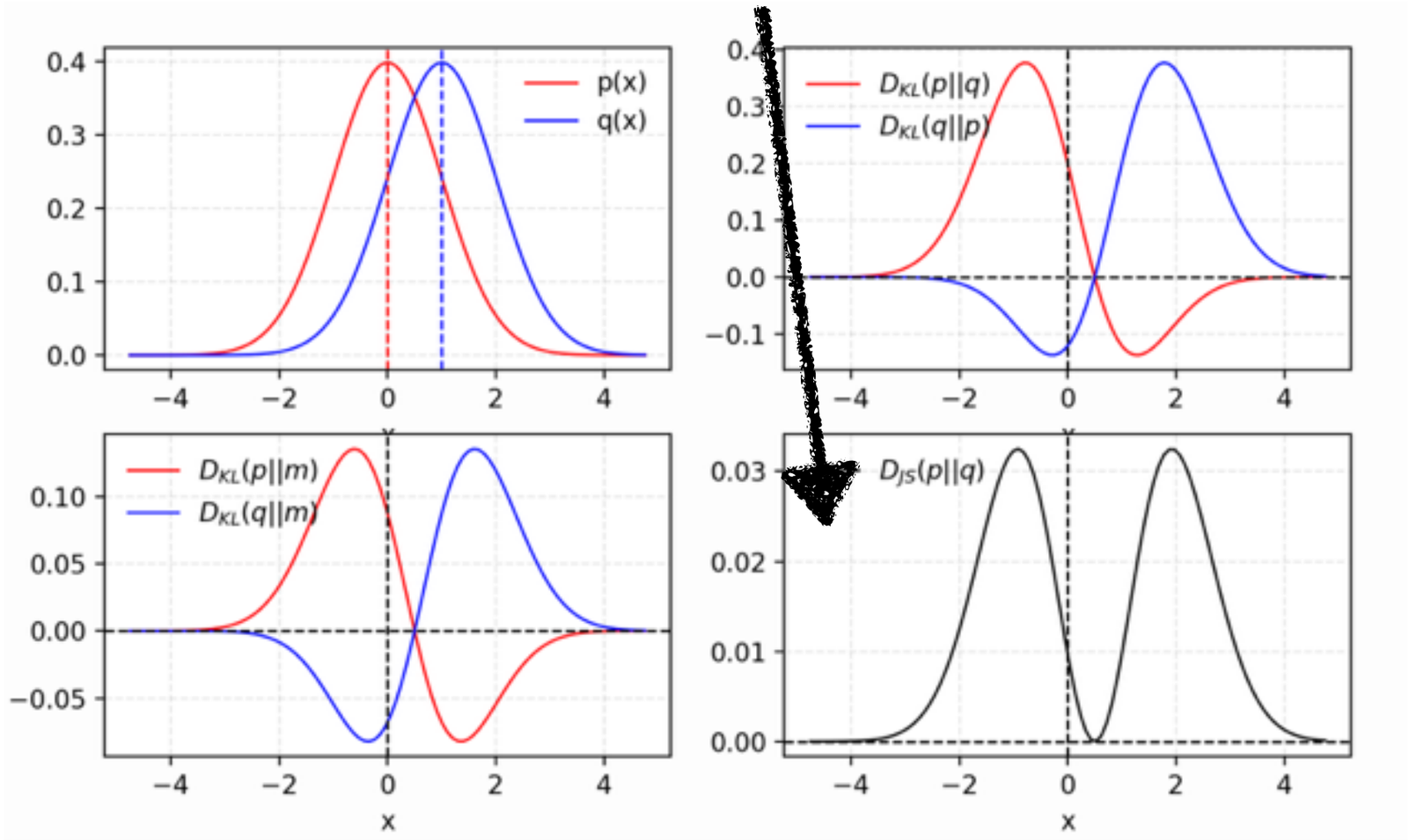
y_{new}(x)= y + \tao x

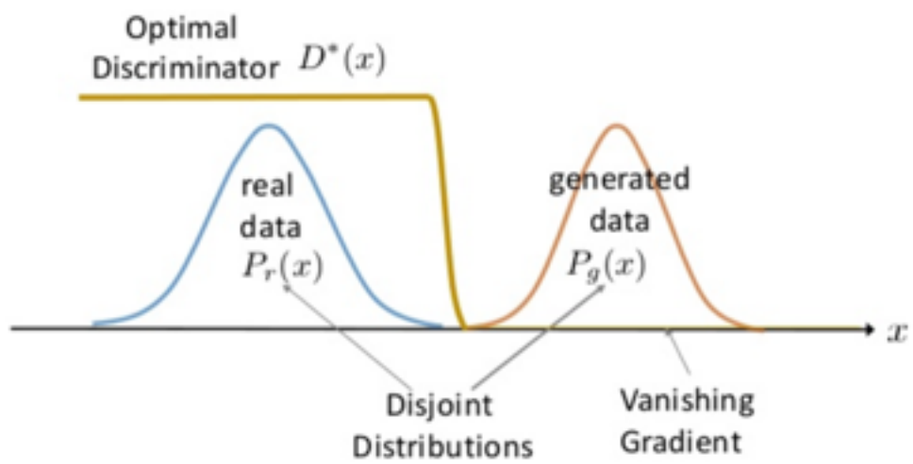


Salimans et al, Improved Techniques for Training GANs, NIPS 2016

Generative Adversarial Networks (GANs)

$$L(G, D^{opt}) = 2D_{JS}(p_r, p_g) - 2\log(2)$$



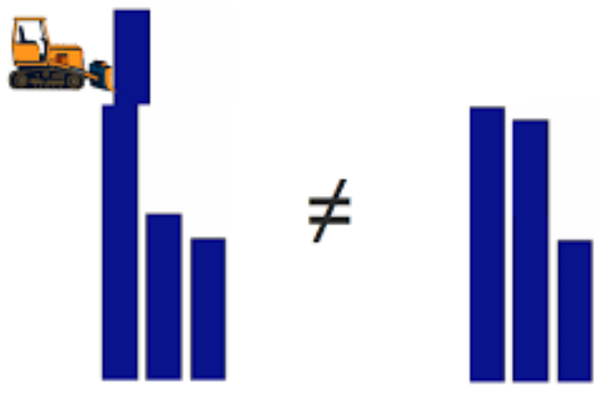


Wasserstein GAN

Wasserstein distance between two probability measures

$$W_p(\mu,\nu) := \left(\inf_{\gamma\in\Gamma(\mu,\nu)}\int_{M\times M} d(x,y)^p \,\mathrm{d}\gamma(x,y)\right)^{1/p},$$

p=1 Earth Mover's Distance

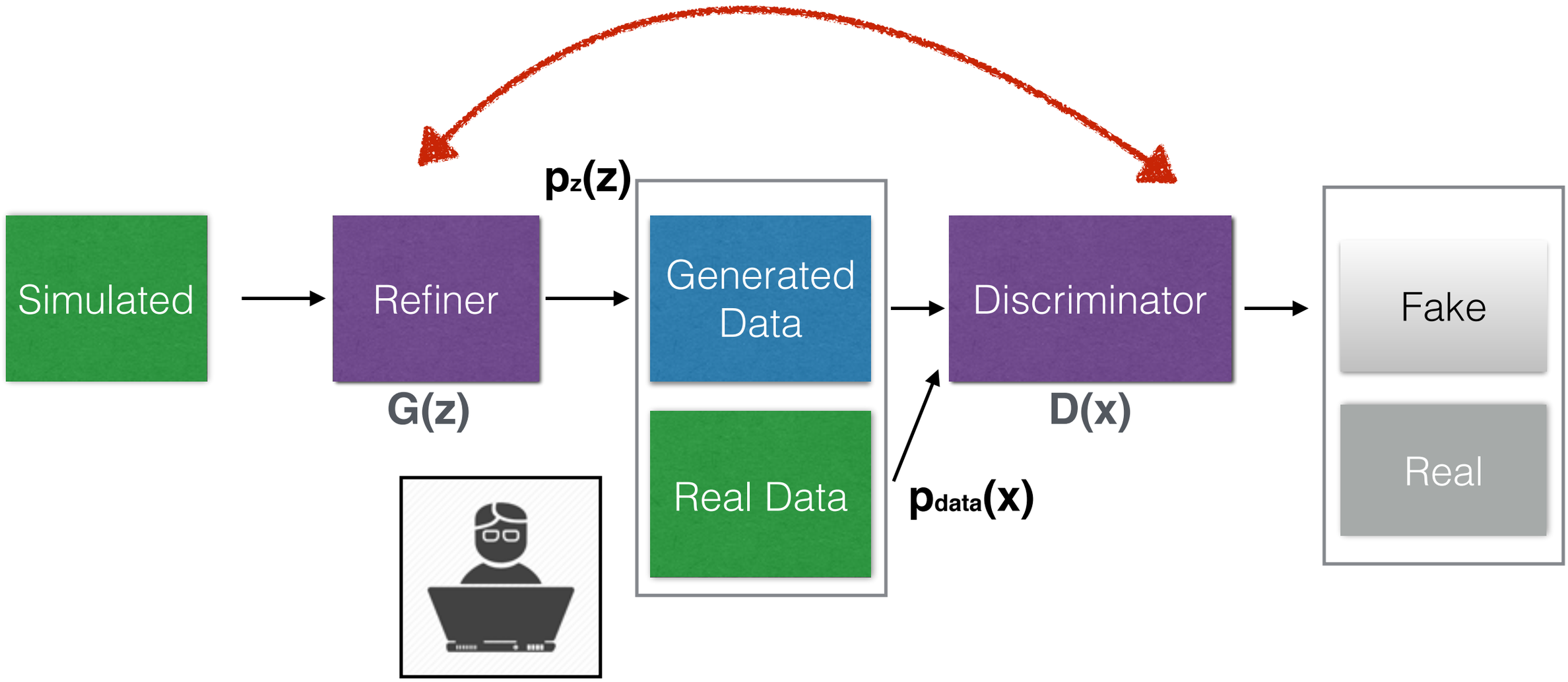


```
def d_loss(y_true, y_pred):
    return K.mean(y_true * y_pred)
```

Binary Case

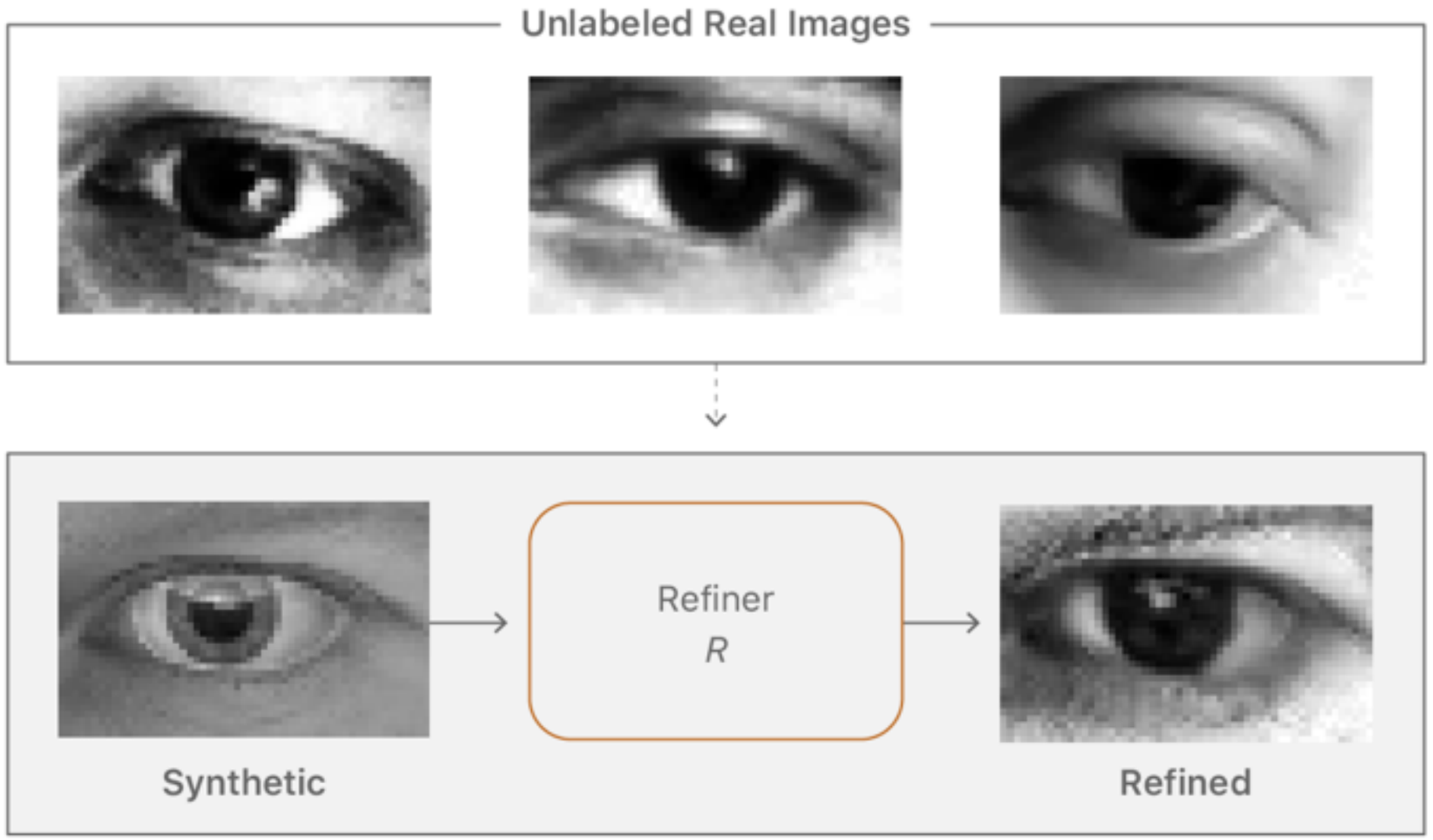
Salimans et al, Improved Techniques for Training GANs, NIPS 2016

GANs to improve simulations



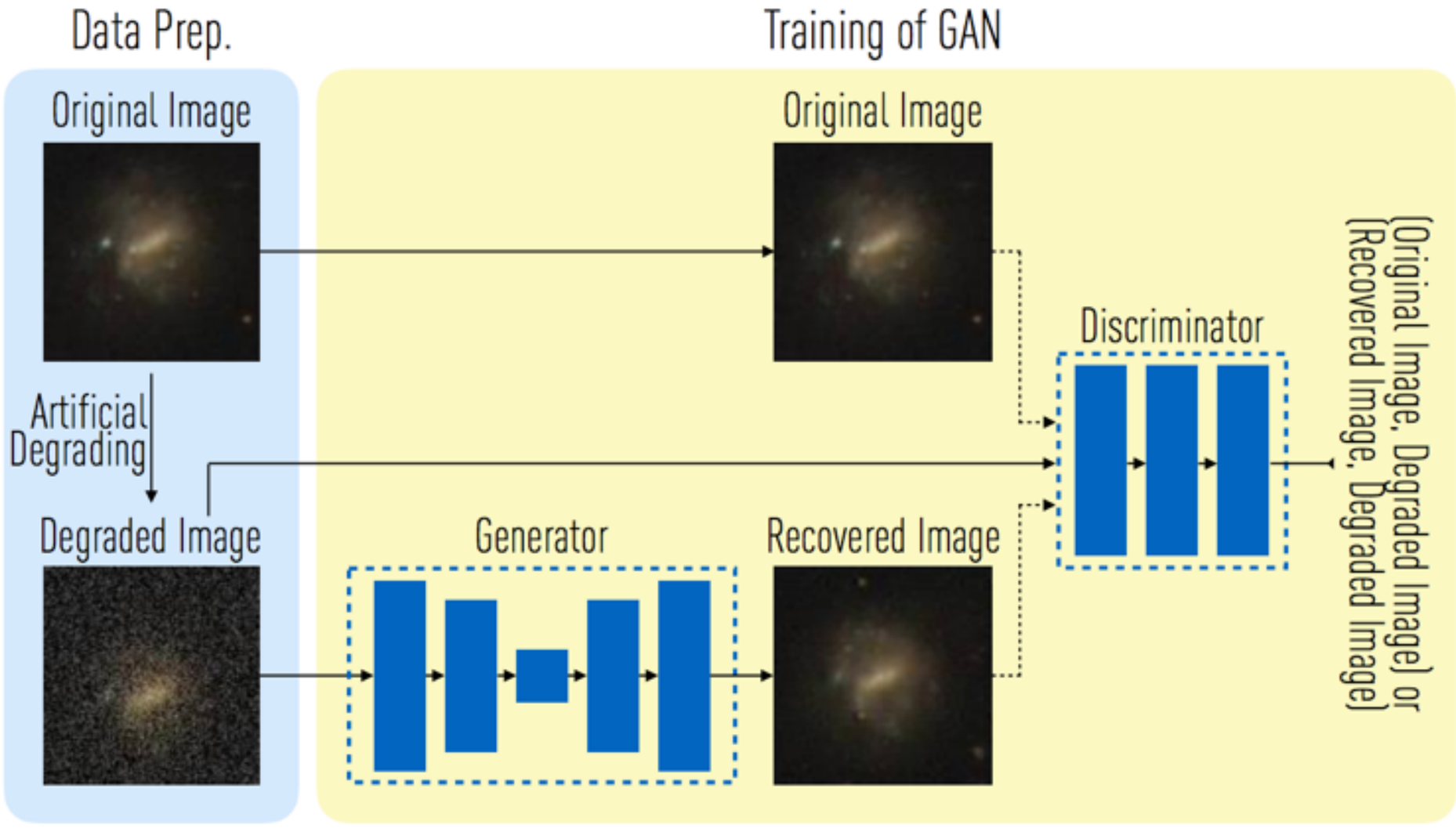
Towards Adversarial Retinal Image Synthesis

GANs to improve simulations



Zhang, Xucong, et al. "Appearance-based gaze estimation in the wild." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

GANs to recover …



https://demo.ds3.ethz.ch/zarastro/

Generative Adversarial Networks recover features in astrophysical images of galaxies beyond the deconvolution limit, Kevin Schawinski et al. MNRAS 2017

GANs in Astronomy

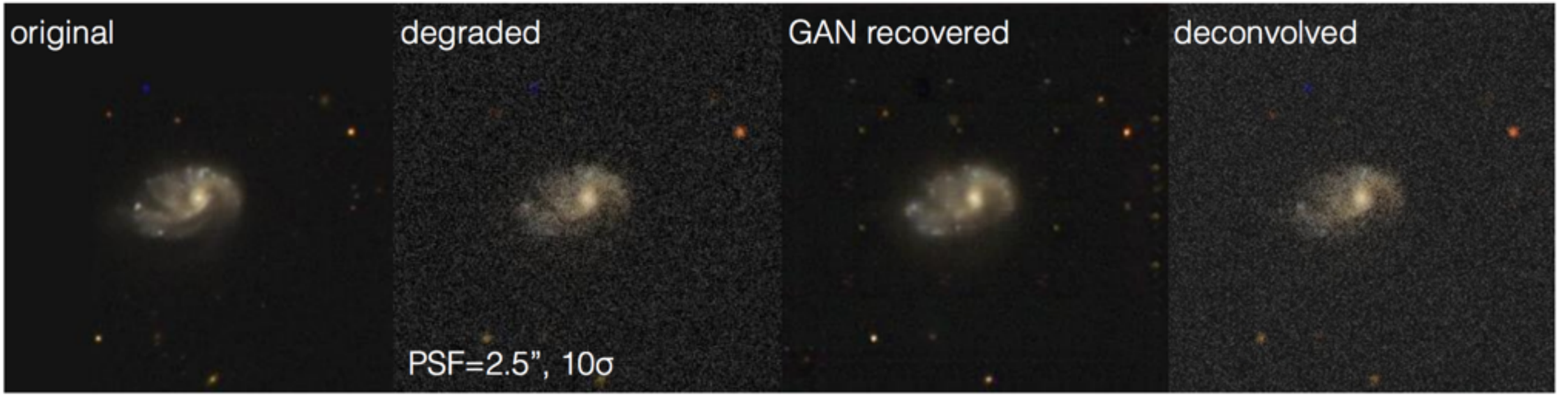


Figure 2. We show the results obtained for one example galaxy. From left to right: the original SDSS image, the degraded image with a worse PSF and higher noise level (indicating the PSF and noise level used), the image as recovered by the GAN, and for comparison, the result of a deconvolution. This figure visually illustrates the GAN’s ability to recover features which conventional deconvolutions cannot.

https://demo.ds3.ethz.ch/zarastro/

Generative Adversarial Networks recover features in astrophysical images of galaxies beyond the deconvolution limit, Kevin Schawinski et al. MNRAS 2017

GANs for separating quasar point sources and host galaxy light

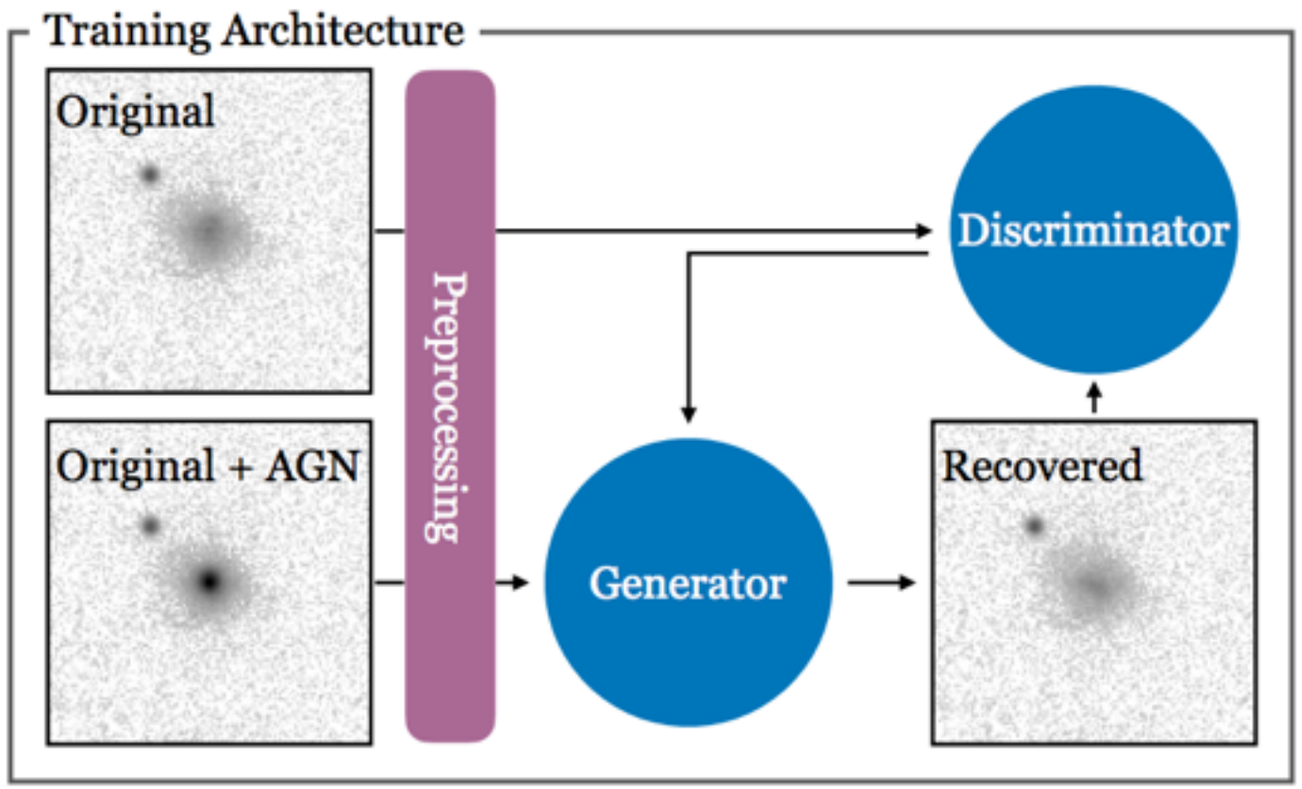
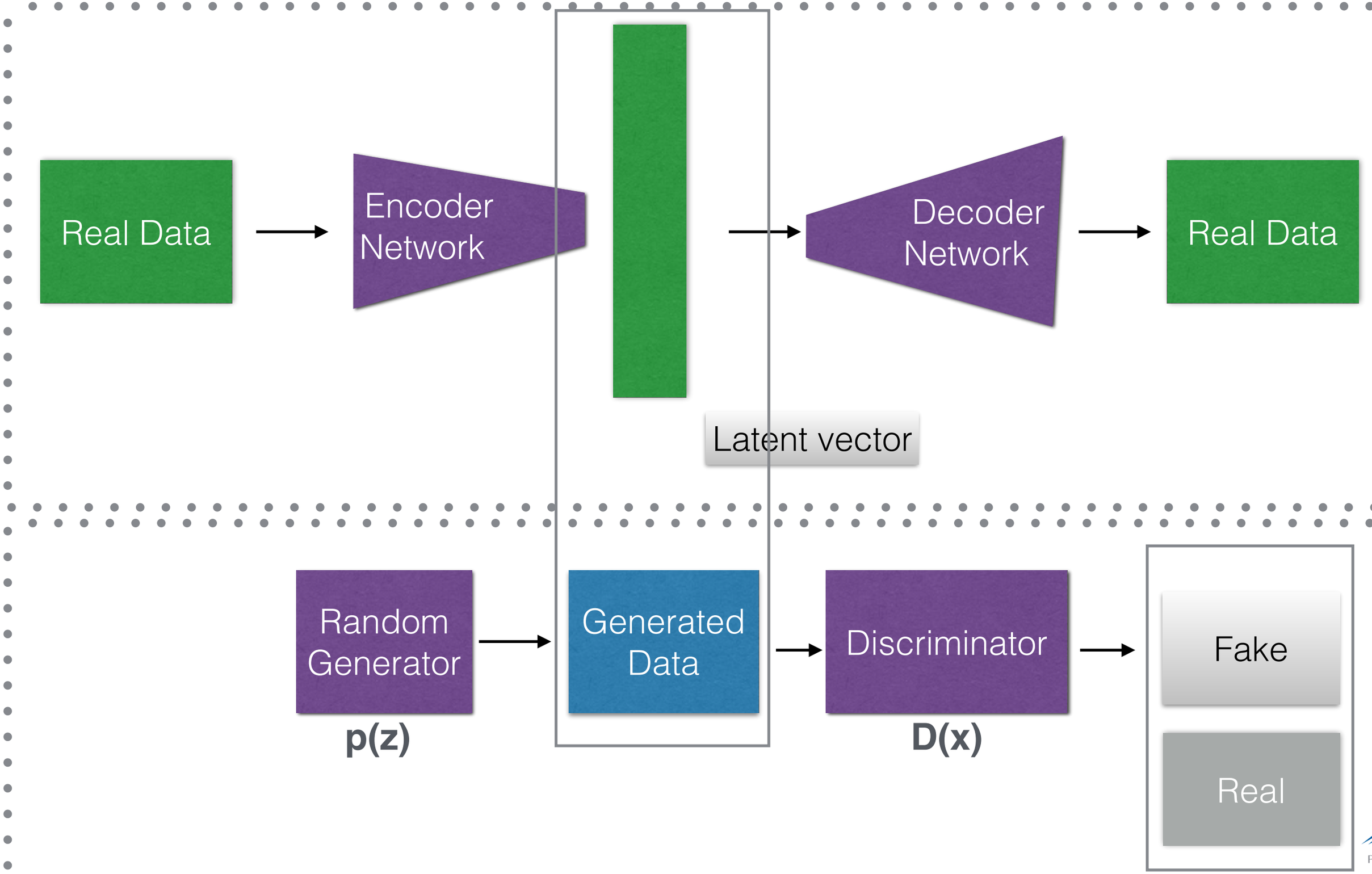


Figure 1. Scheme of the architecture used in this work. The generator takes as input the modified image (original galaxy image with a simulated PS in its center) and tries to recover the original galaxy image. The discriminator distinguishes the original from the recovered image. Before feeding the images to the GAN they are normalized to have values in [0, 1] and transformed by an invertible stretch function.

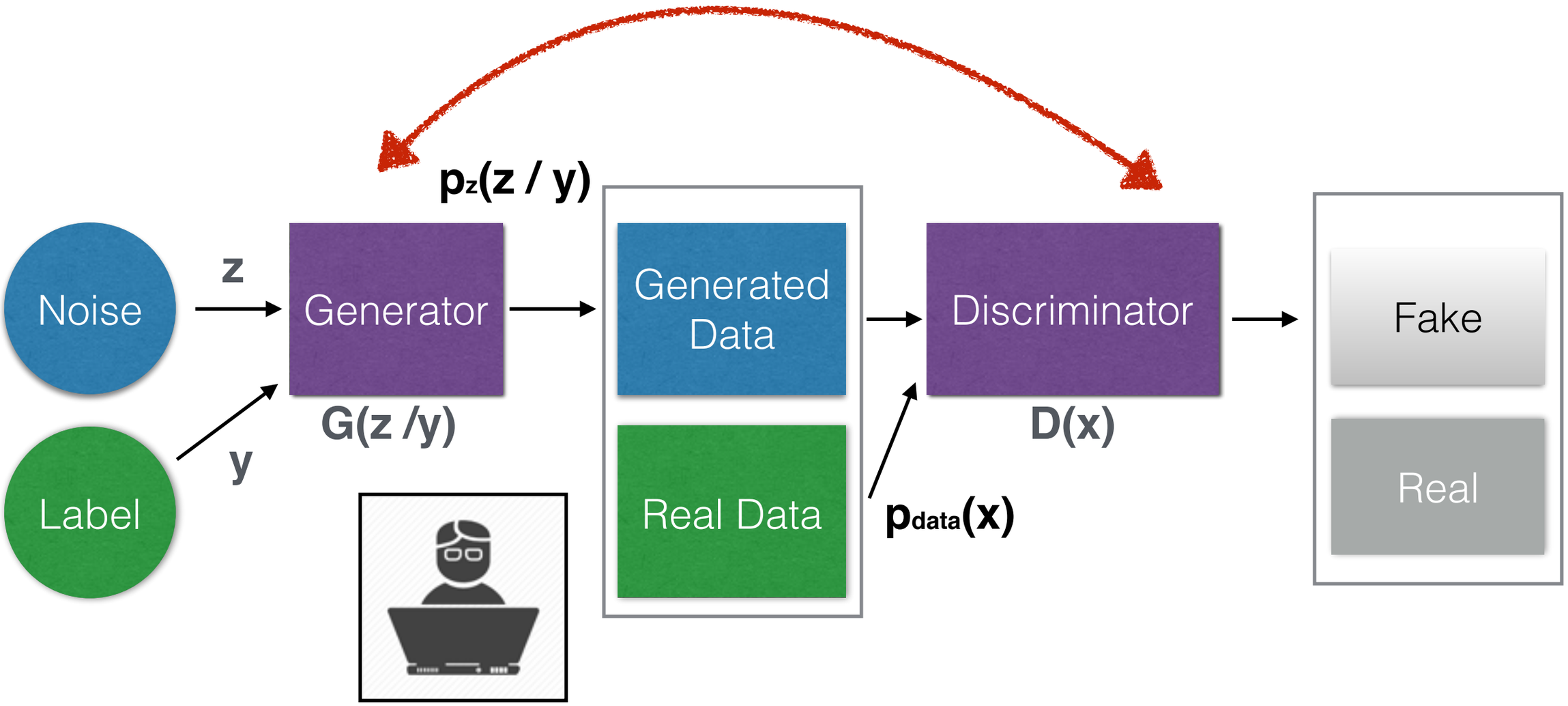
Dominic Stark, et al. PSFGAN: a generative adversarial network system for separating quasar point sources and host galaxy light, MNRAS 2018

Adversarial Autoencoder

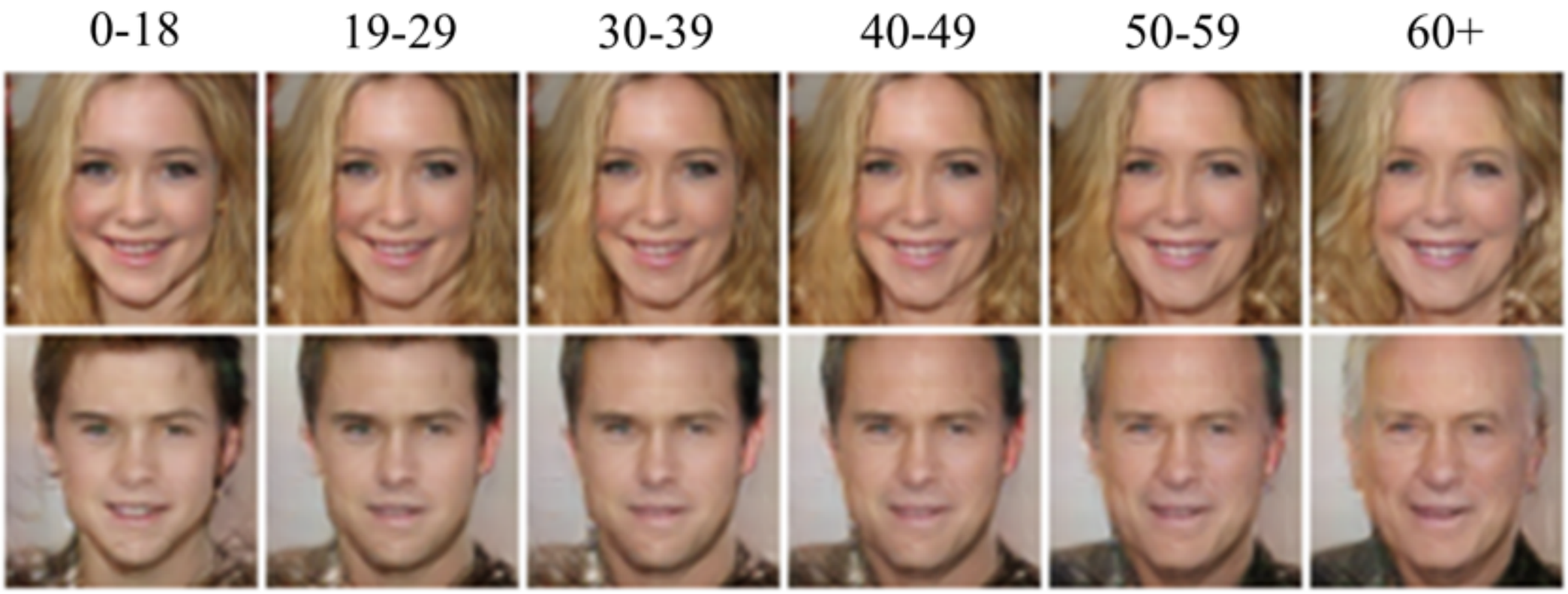


A. Makhzani et al. (2016). Adversarial Autoencoder, ICLR, 2016

Conditional Generative Adversarial Networks

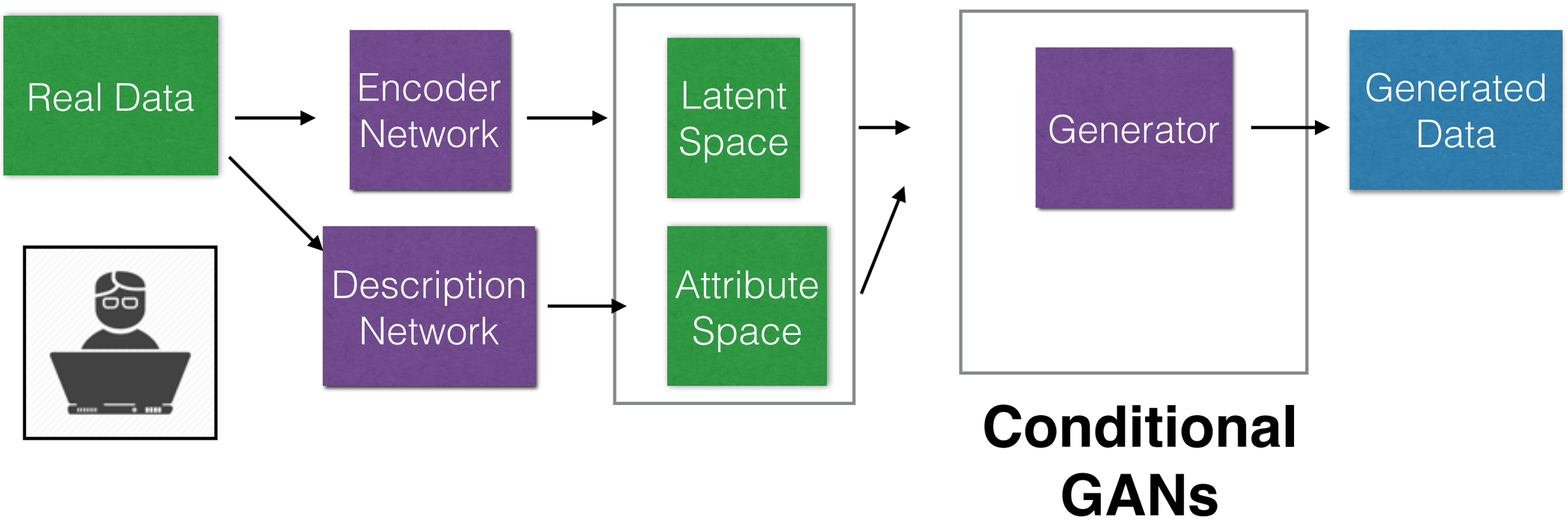


Conditional Generative Adversarial Networks



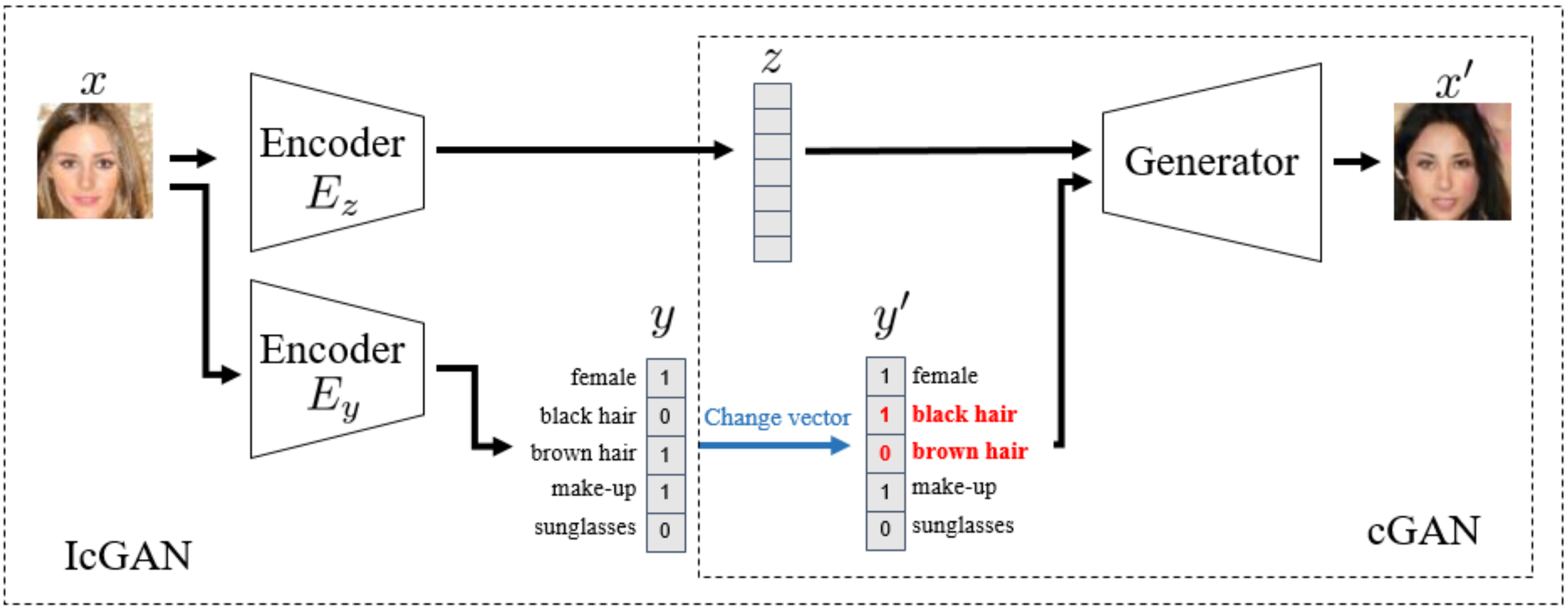
Antipov, Grigory, Moez Baccouche, and Jean-Luc Dugelay. "Face aging with conditional generative adversarial networks." arXiv preprint arXiv:1702.01983 (2017).

Invertible Conditional Generative Adversarial Networks



G. Perarnau et al, Invertible Conditional GANs for image editing. NIPS Workshop on Adversarial Training, 2016.

Invertible Conditional Generative Adversarial Networks

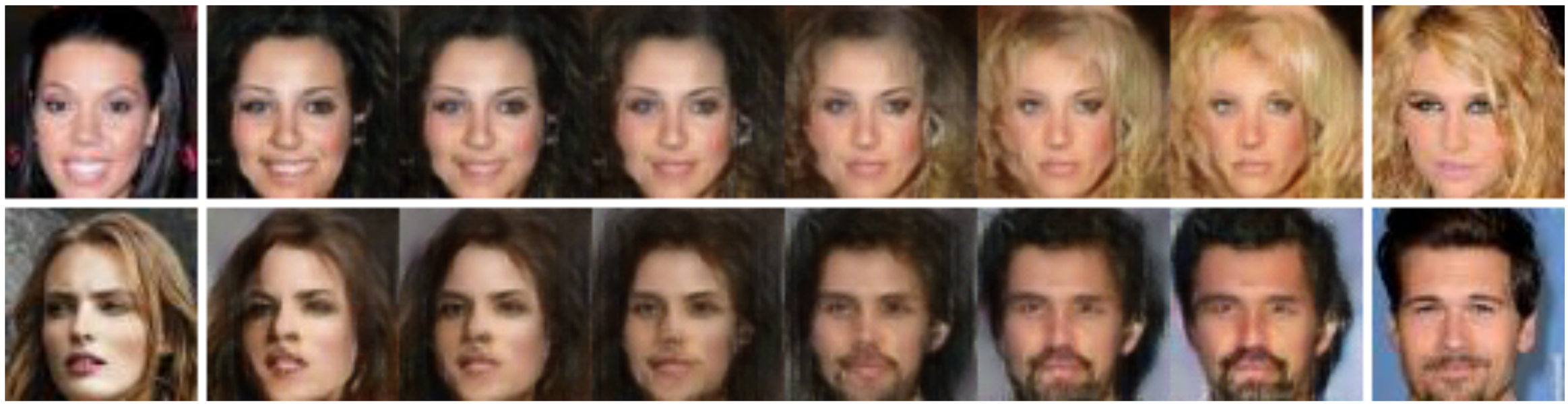


G. Perarnau et al, Invertible Conditional GANs for image editing. NIPS Workshop on Adversarial Training, 2016.

Invertible Conditional Generative Adversarial Networks

G. Perarnau Invertible Conditional GANs for image editing. November 2016.

Invertible Conditional Generative Adversarial Networks



G. Perarnau et al, Invertible Conditional GANs for image editing. NIPS Workshop on Adversarial Training, 2016.

Conditional GANs in Astronomy?

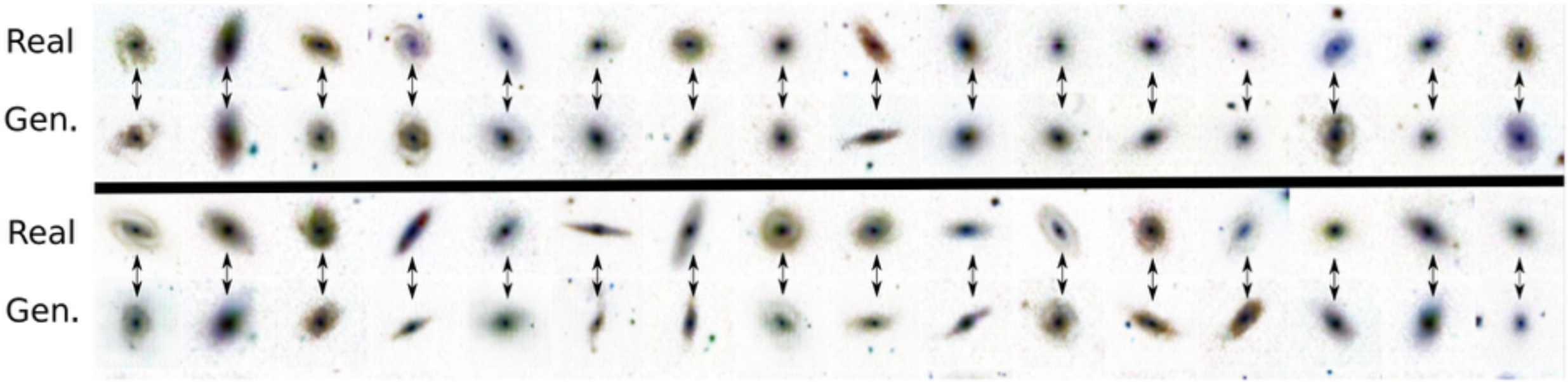


Fig. 2: Samples from the GALAXY-ZOO dataset versus generated samples using conditional generative adversarial network of Section III. Each synthetic image is a 128×128 colored image (here inverted) produced by conditioning on a set of features $y \in [0, 1]^{37}$. The pair of observed and generated images in each column correspond to the same y value. For details on these crowd-sourced y features see Willett et al. (2013). These instances are selected from the test-set and were unavailable to the model during the training.

Ravanbakhsh, S., Lanusse, F., Mandelbaum, R., Schneider, J. G., & Poczos, B. (2017). Enabling Dark Energy Science with Deep Generative Models of Galaxy Images. In AAAI (pp. 1488-1494).


DARKSIDE
LIGHTSIDE

Toward an explainable AI (Deep Learning)

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

Optimization Algorithm

Loss Function

Architecture

Why did you do that?
When do you succeed?
When do you fail?
When can I trust you?
How do I correct an error?

Toward an explainable AI (Deep Learning)

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

Optimization Algorithm

Loss Function

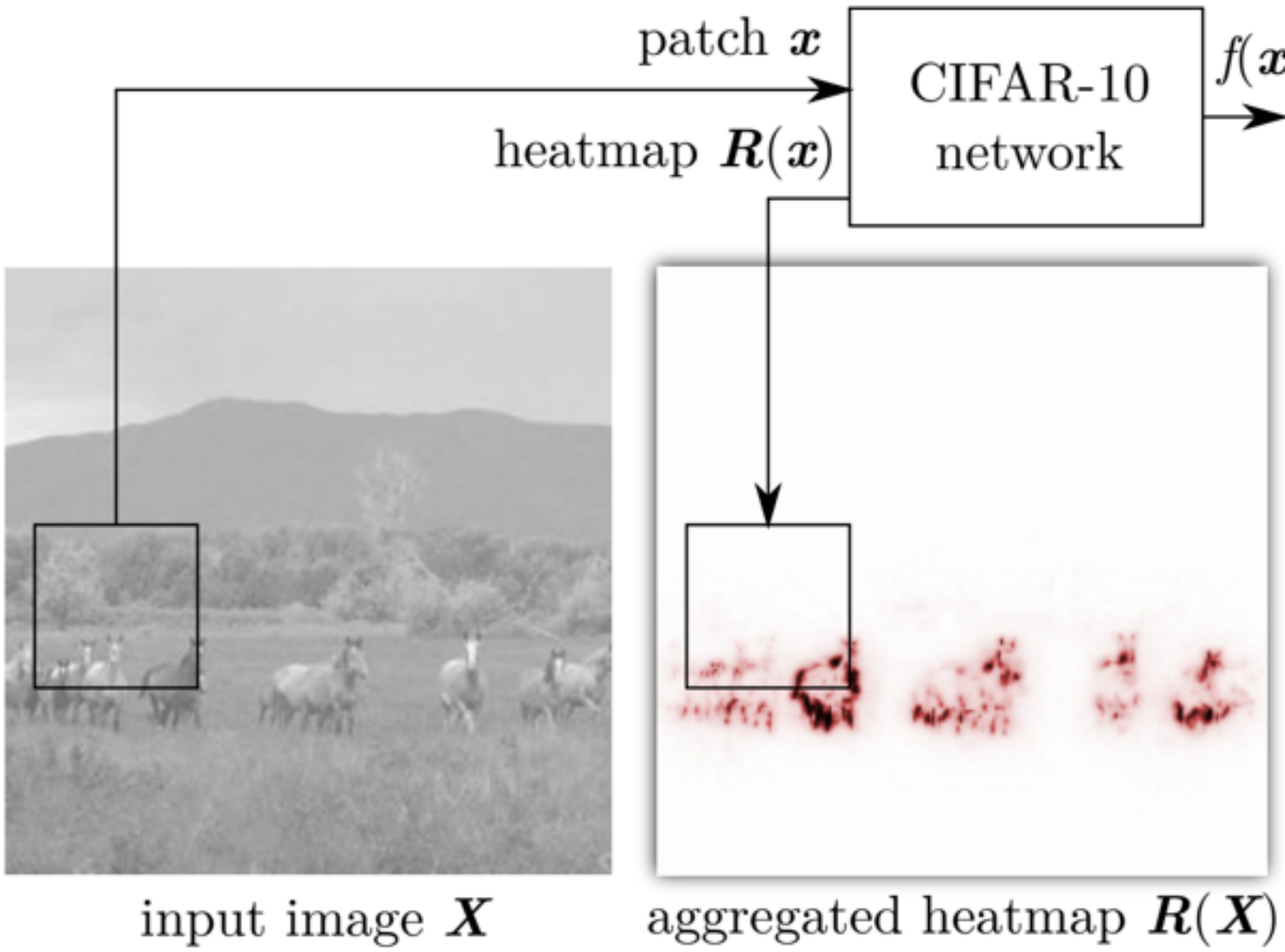
Architecture

Why did you do that?
When do you succeed?
When do you fail?
When can I trust you?
How do I correct an error?
Can someone fool you?

Adversarial Examples
Adversarial Training
(Model,Data) Confidence
Network Simplification
Network Understanding

MINES ParisTech

Network Understanding



Heatmap / Aggregated

Toward an explainable AI (Deep Learning)

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

Optimization Algorithm

Loss Function

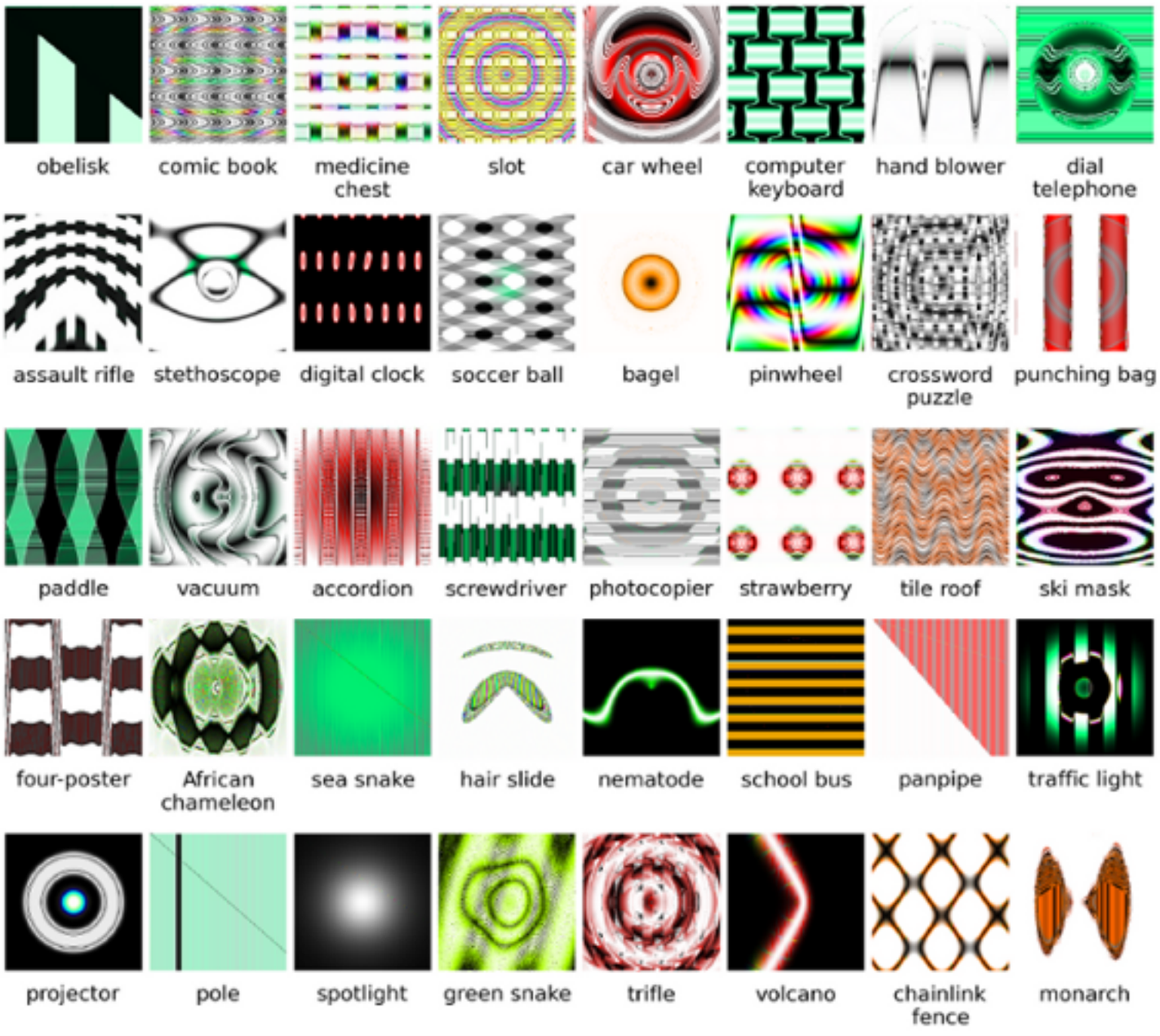
Architecture

Why did you do that?
When do you succeed?
When do you fail?
When can I trust you?
How do I correct an error?
Can someone fool you?

Adversarial Examples
Adversarial Training
(Model,Data) Confidence
Network Simplification
Network Understanding

MINES ParisTech

Deep neural networks are easily fooled: High confidence predictions for unrecognizable images

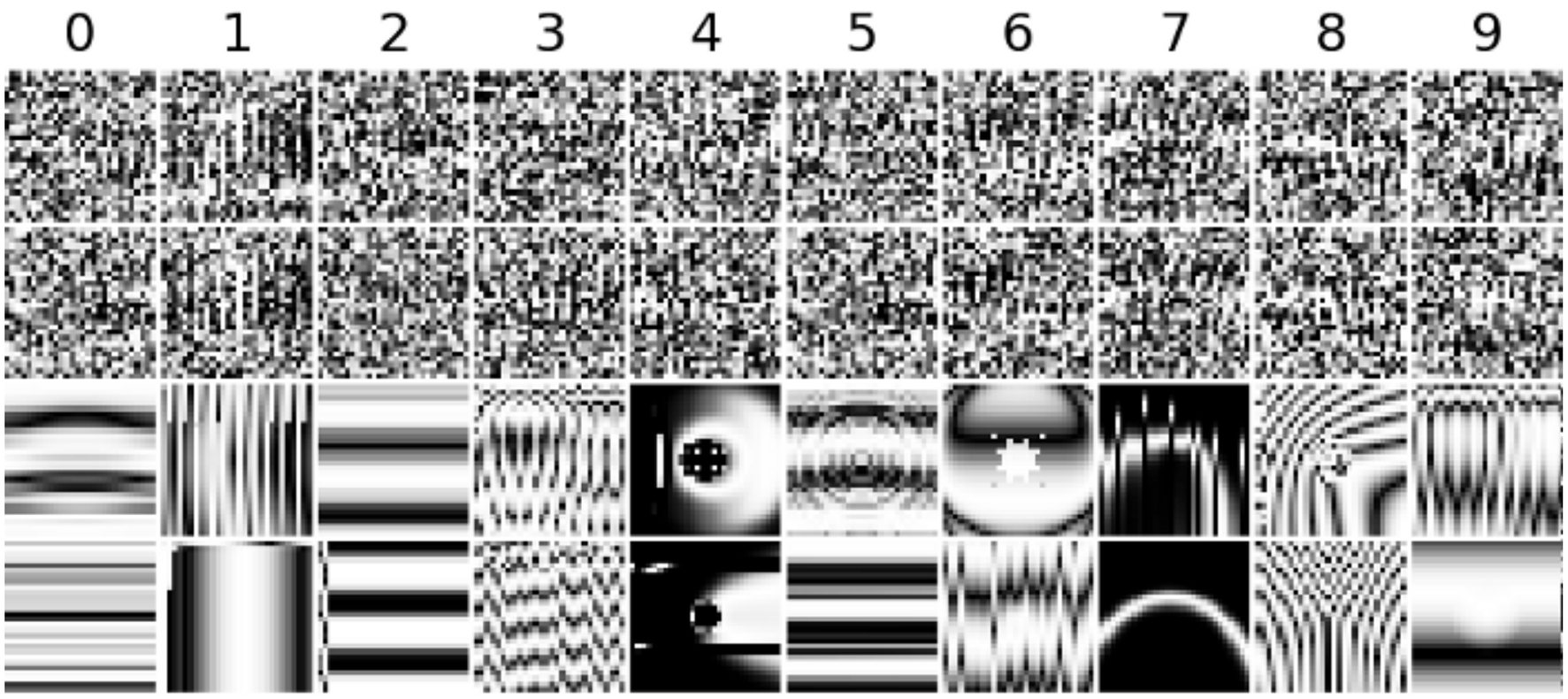


The mean DNN confidence scores for these images is 99.12%

Nguyen A, Yosinski J, Clune J. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In Computer Vision and Pattern Recognition (CVPR '15), IEEE, 2015.

MINES ParisTech

Deep neural networks are easily fooled: High confidence predictions for unrecognizable images



Directly encoded, thus irregular, images that a cutting edge deep neural networ
(LeNet) believes with 99.99% confidence are digits 0-9.

MNIST

Toward an explainable AI (Deep Learning)

Deep Learning

Input → Feature Extraction+ Classification → Output

Training Samples

Optimization Algorithm

Loss Function

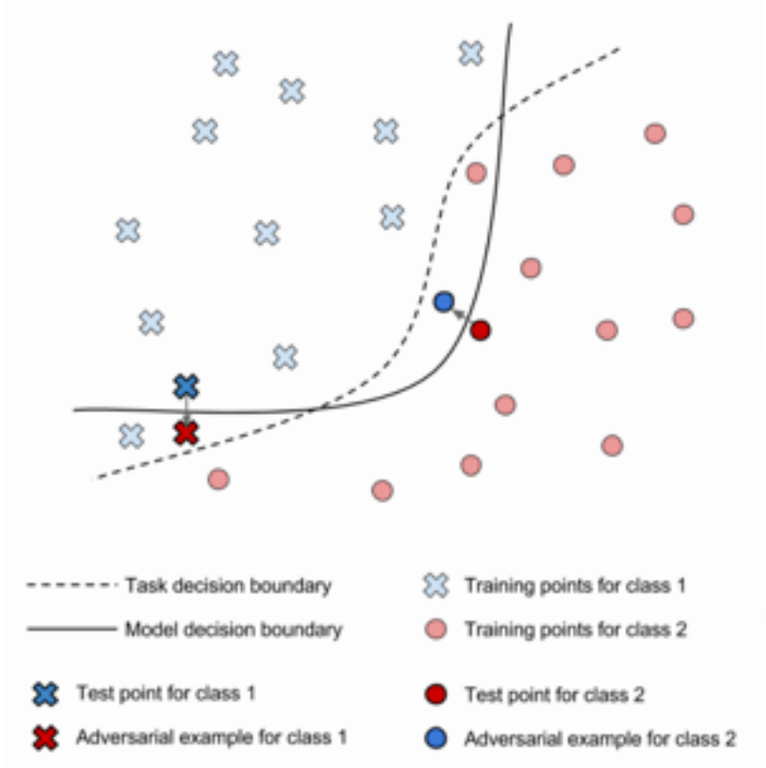
Architecture

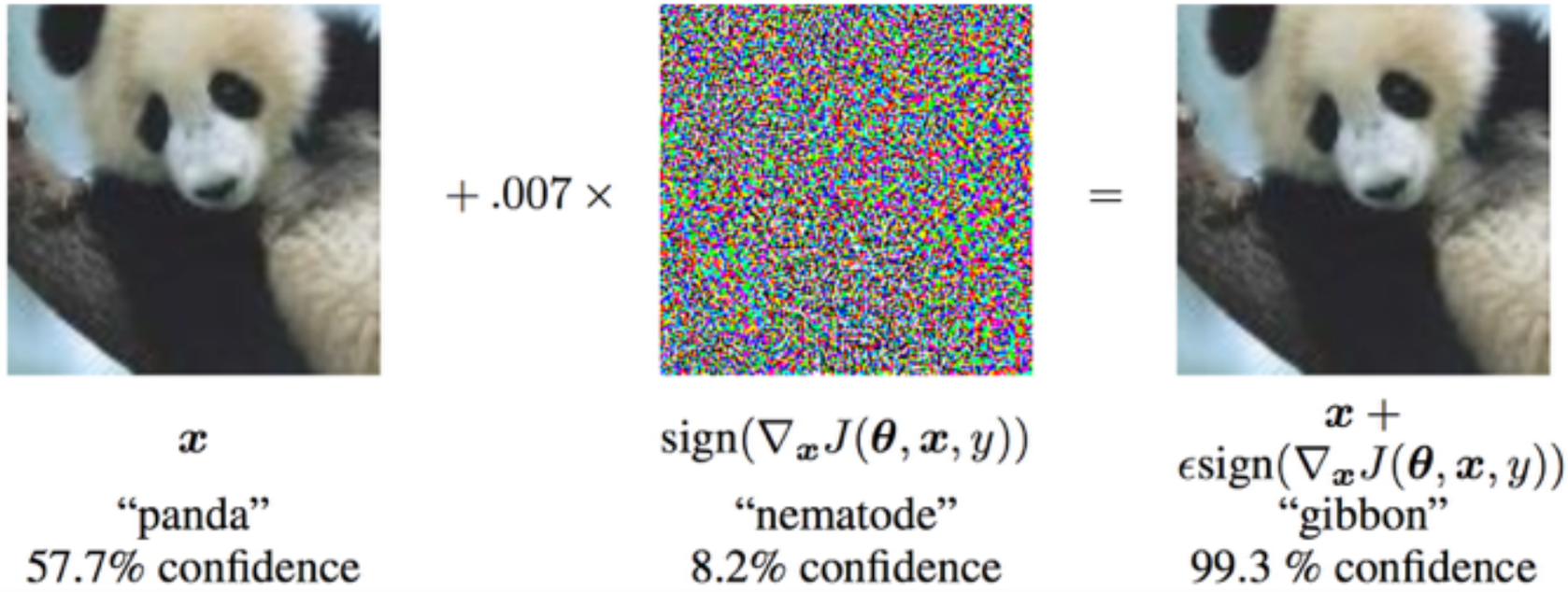
Why did you do that?
When do you succeed?
When do you fail?
When can I trust you?
How do I correct an error?
Can someone fool you?

Adversarial Examples
Adversarial Training
(Model,Data) Confidence
Network Simplification
Network Understanding

MINES ParisTech

Black-Box Adversarial Attacks

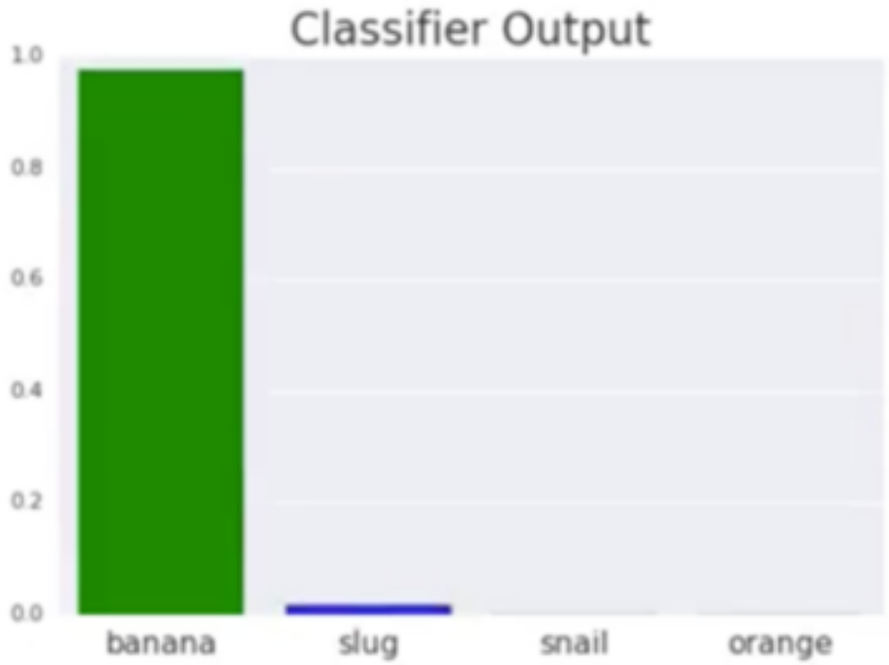


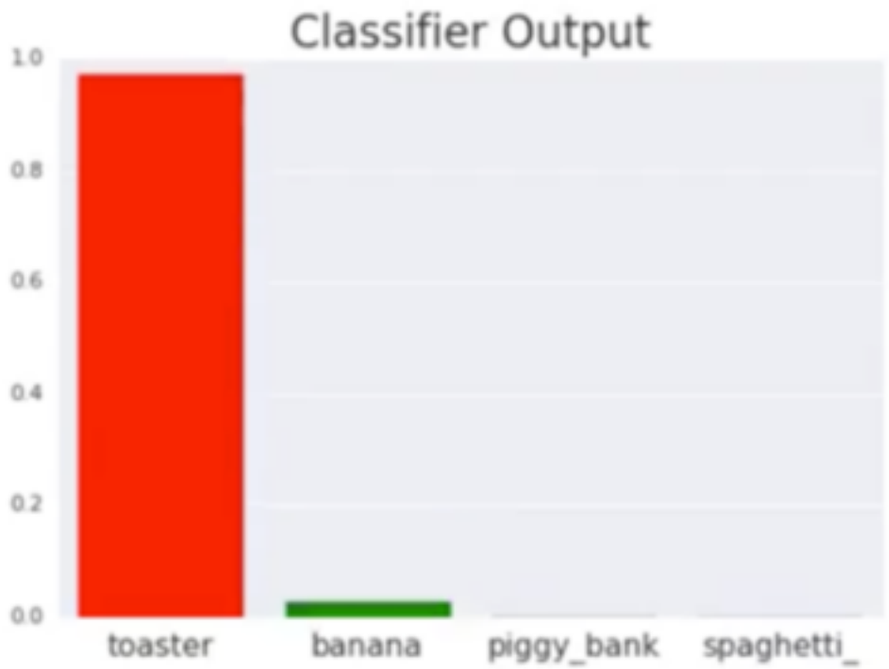


Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Berkay Celik, Z., & Swami, A. (2016). Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples.

MINES ParisTech

White-Box Adversarial Attacks





Brown, Tom B., et al. "Adversarial patch." arXiv preprint arXiv:1712.09665 (2017)

MINES ParisTech

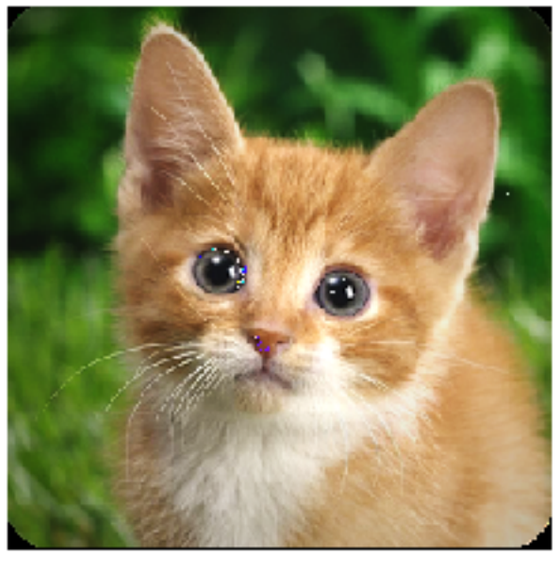
White-Box Adversarial Attacks

'Egyptian_cat', 0.3396838

'jay', 0.96423554

VGG19

'plastic_bag', 0.54238236



'lynx', 0.47152225

'electric_ray', 0.8287997

MINES ParisTech

Conclusions

A deep learning machine is a powerful tool for *extracting regularities from data according a given objective*

A deep learning machine will be just as smart as:

- the data it gets
- the objective it optimizes