

# Generative Modeling

How to Use Deep Neural Networks

Denis Derkach, Artem Ryzhikov, Maxim Artemiev

Laboratory for methods of big data analysis

Spring 2021



# In this Lecture

- ▶ What's a generative model.
- ▶ What it does.
- ▶ What are the main components.

# Results of Generative Modeling



# This X Does Not Exist!



## This Person Does Not Exist

The site that started it all, with the name that says it all. Created using a style-based generative adversarial network (StyleGAN), this website had the tech community buzzing with excitement and intrigue and inspired many more sites.

Created by Phillip Wang.



## This Cat Does Not Exist

These purr-fect GAN-made cats will freshen your feeline-gs and make you wish you could reach through your screen and cuddle them. Once in a while the cats have visual deformities due to imperfections in the model – beware, they can cause nightmares.

Created by Ryan Hoover.



## This Rental Does Not Exist

Why bother trying to look for the perfect home when you can create one instead? Just find a listing you like, buy some land, build it, and then enjoy the rest of your life.

Created by Christopher Schmidt.

<https://thisxdoesnotexist.com/>

# Video Modifications

We can **automatically** remove snow in video



<https://incrussia.ru/news/ii-nauchilsya-poddelyvat-video/>

# More Tricks for Your Brain

- ▶ Text generation.

Two men happily working on a plastic computer.  
The toilet in the bathroom is filled with a bunch of ice.  
A bottle of wine near stacks of dishes and food.  
A large airplane is taking off from a runway.  
Little girl wearing blue clothing carrying purple bag sit

---

## SeqGAN (Baseline)

---

A baked mother cake sits on a street with a rear of it.  
A tennis player who is in the ocean.  
A highly many fried scissors sits next to the older.  
A person that is sitting next to a desk.  
Child jumped next to each other.

---

## RankGAN (Ours)

---

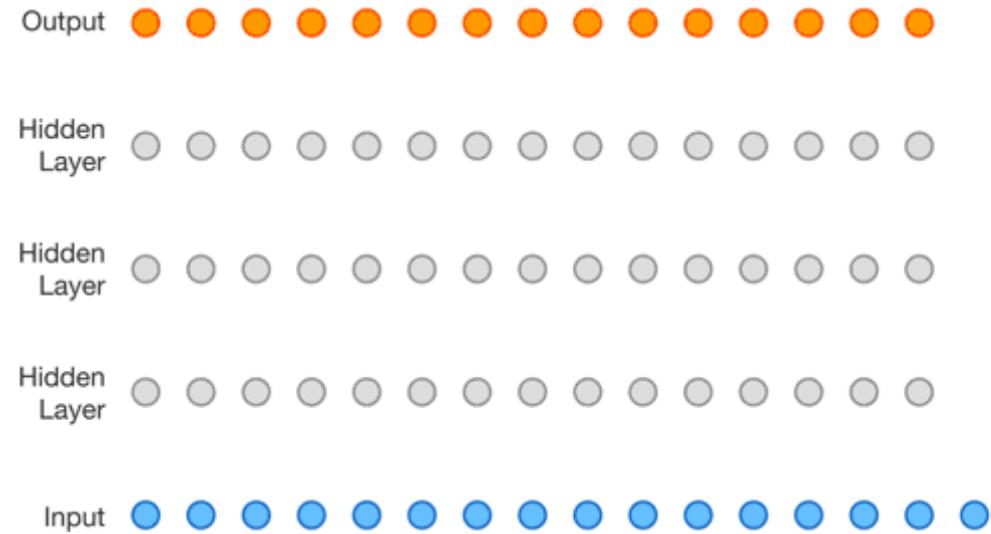
Three people standing in front of some kind of boats.  
A bedroom has silver photograph desk.  
The bears standing in front of a palm state park.  
This bathroom has brown bench.  
Three bus in a road in front of a ramp.

# More Tricks for Your Brain

- ▶ Text generation.



- ▶ Voice from text generation.



# More Tricks for Your Brain

- ▶ Text generation.
- ▶ Voice from text generation.
- ▶ Style transfer.



# More Tricks for Your Brain: Links

- ▶ Text generation.
  - [https://www.tensorflow.org/tutorials/text/text generation](https://www.tensorflow.org/tutorials/text/text_generation)
- ▶ Voice from text generation.
  - <https://deepmind.com/blog/article/wavenet-generative-model-raw-audio>
- ▶ Style transfer.
  - <https://towardsdatascience.com/style-transfer-with-gans-on-hd-images-88e8efcf3716>

# Generative Models Progress

The news are well motivated.



- ▶ Enormous progress in recent years.
- ▶ Technology is ready for new tasks.

[https://twitter.com/goodfellow\\_ian/status/1084973596236144640](https://twitter.com/goodfellow_ian/status/1084973596236144640)

# Generative Models Failures

FAST COMPANY

02-08-19

## This AI dreams about cats—and they'll haunt your nightmares

Nvidia's new AI is capable of generating everything from human faces to kittens. But the development process left behind plenty of...errors.

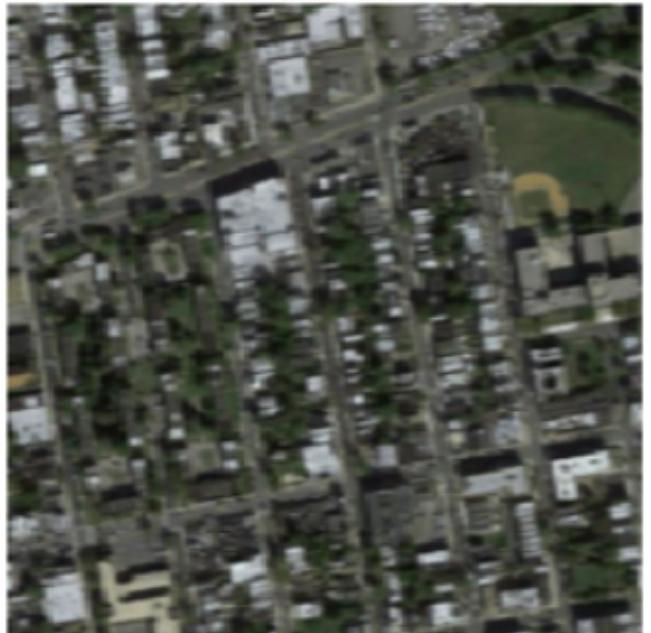
- ▶ Image is created as **interpolation** between existing ones.



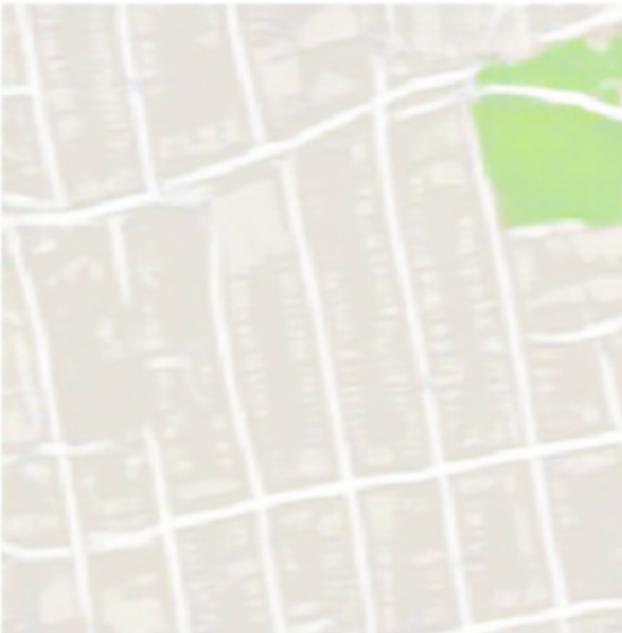
<https://www.fastcompany.com/90303908/this-ai-dreams-about-cats-and-theyll-haunt-your-nightmares>

# Dealing with Maps: generating map

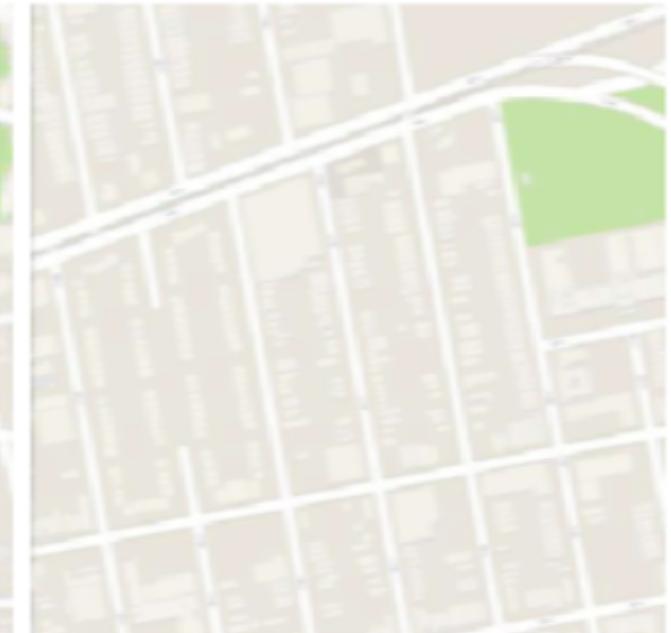
- ▶ Image-to-image style transfer.
- ▶ Creates map on-the-fly from satellite image.



**Input**



**Generated**

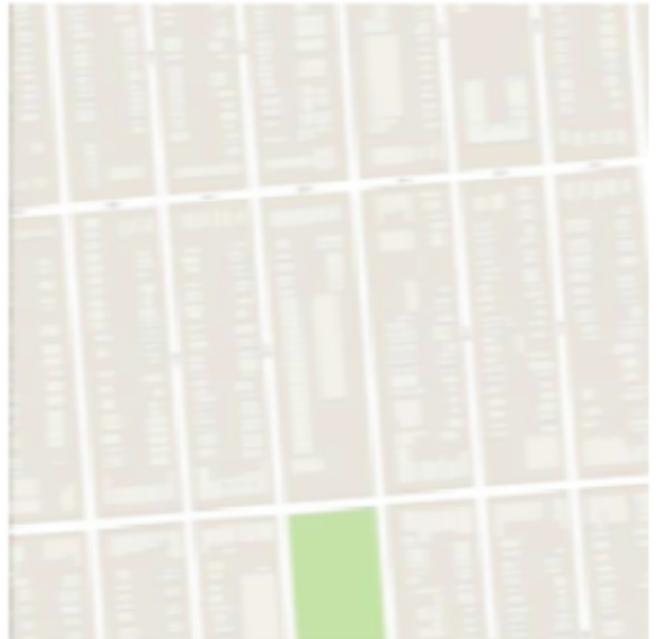


**True**

<https://github.com/ChengBinJin/pix2pix-tensorflow>

# Dealing with Maps: generating satellite image

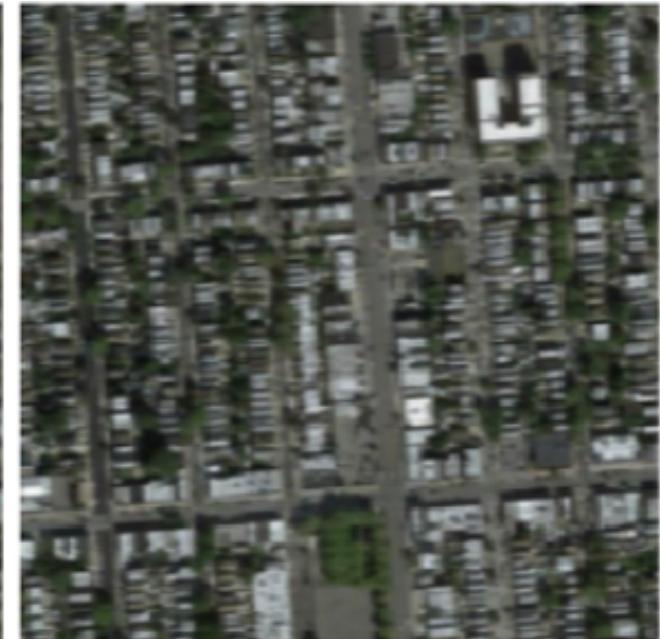
- ▶ Image-to-image style transfer
- ▶ Creates map on-the-fly from satellite image and vice versa.



**Input**



**Generated**

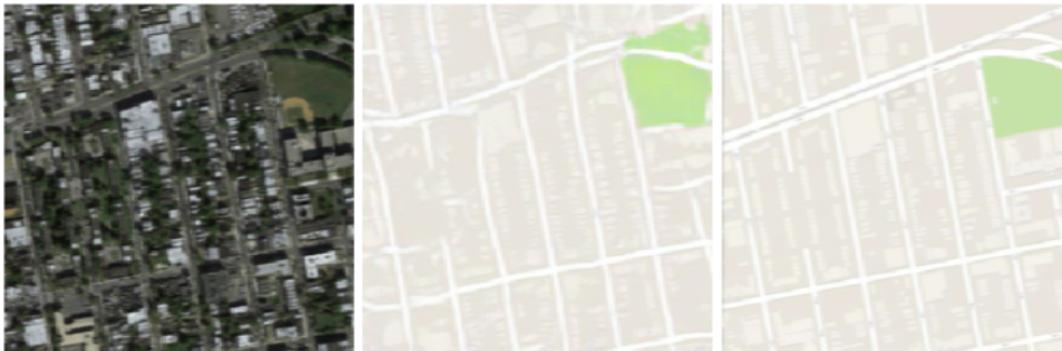


**True**

<https://github.com/ChengBinJin/pix2pix-tensorflow>

# Dealing with Maps: generating satellite image

- ▶ Image-to-image style transfer
- ▶ Creates map on-the-fly from satellite image and vice versa.
- ▶ The technology is the same as for “Monet” painting. Just need good representation.



=



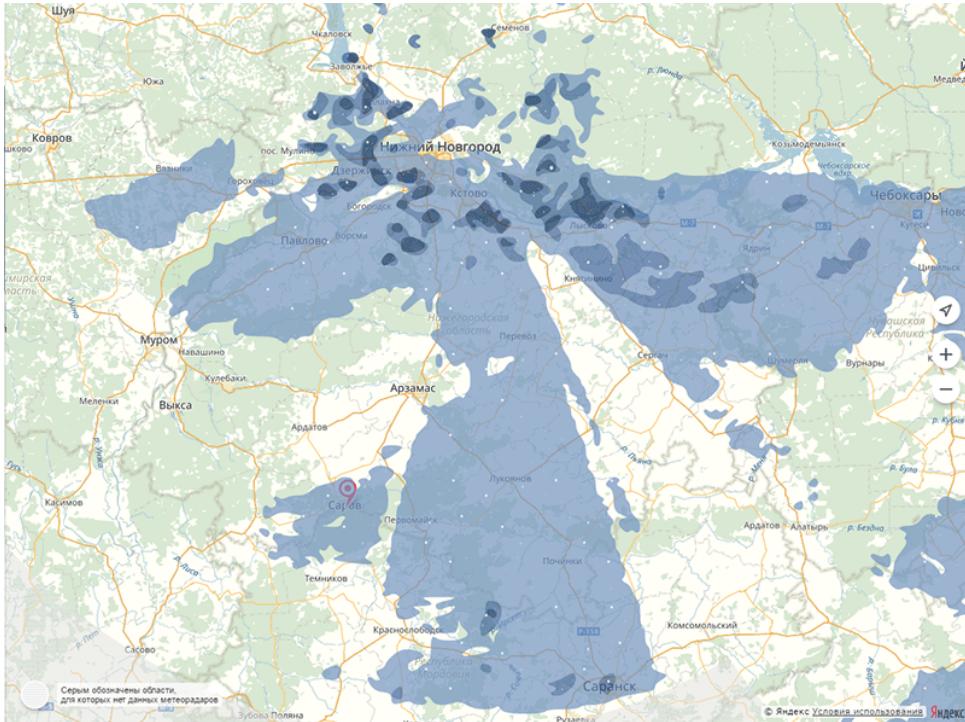
# Dealing with Satellite Images: Super-resolution

- ▶ We can “create” a more appropriate map quality.
- ▶ This later can be used in segmentation task.



<https://omdena.com/blog/super-resolution/>

# Weather prediction: nowcast



- ▶ Video prediction for precipitation.
- ▶ Generation of future state, based on the previous one.

<https://www.kdd.org/kdd2019/accepted-papers/view/precipitation-nowcasting-with-satellite-imagery>

# Dirty Road Signs Generation



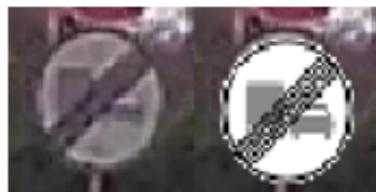
Class 0



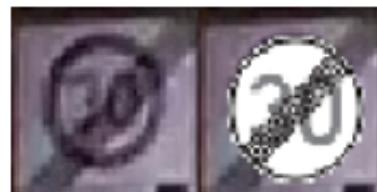
Class 1



Class 2



Class 6



Class 7



Class 8

- ▶ Road signs from the book are too clean.
- ▶ Need to put mud and shadows on the signs.

<https://arxiv.org/abs/1907.12902>  
<https://www.hse.ru/sci/diss/426009543>

# What Generative Models **Do not** Produce

- ▶ No new information is created.
- ▶ All interpolations are done in some representation space.

# Chapter outcome

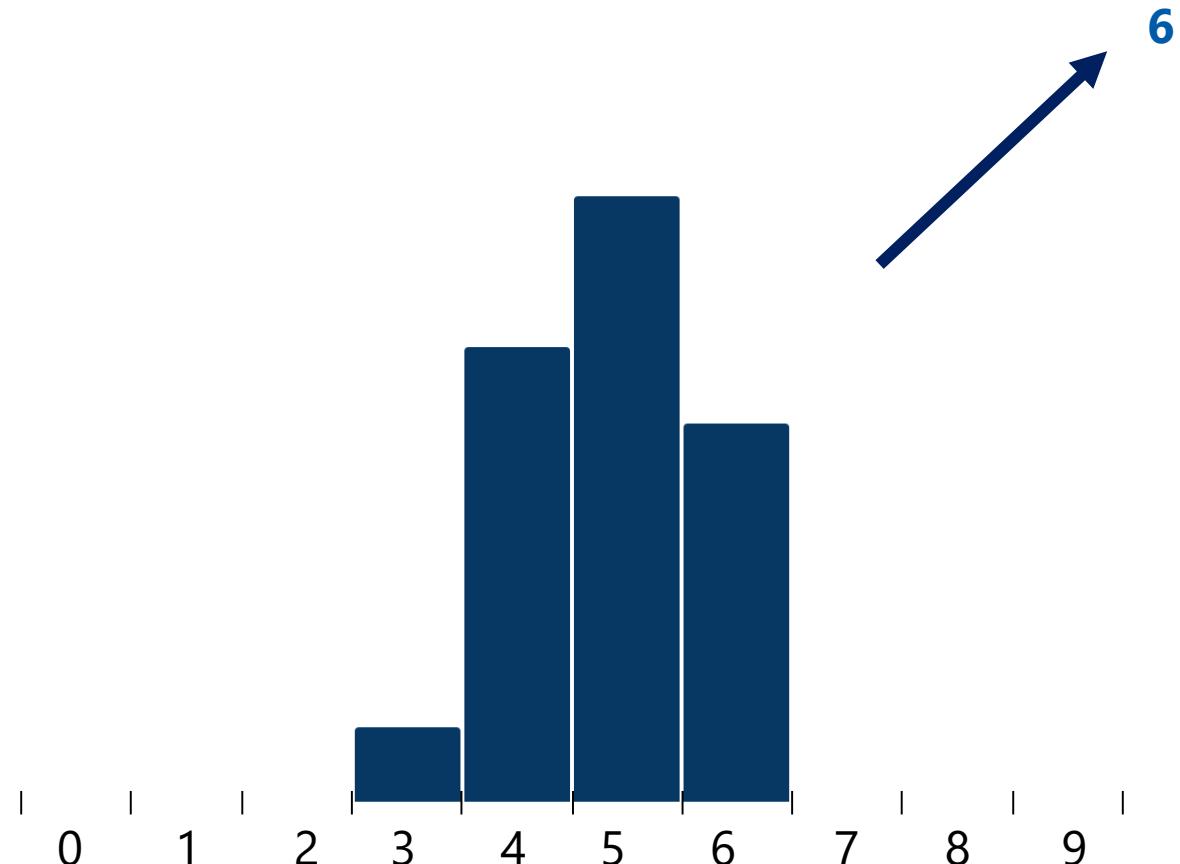
- ▶ Generative models in machine learning were developing quickly in the last 6 years.
- ▶ Current state-of-the-art allows to implement generative models in more serious tasks than deceiving non-expert human.

# What is Generative Modeling



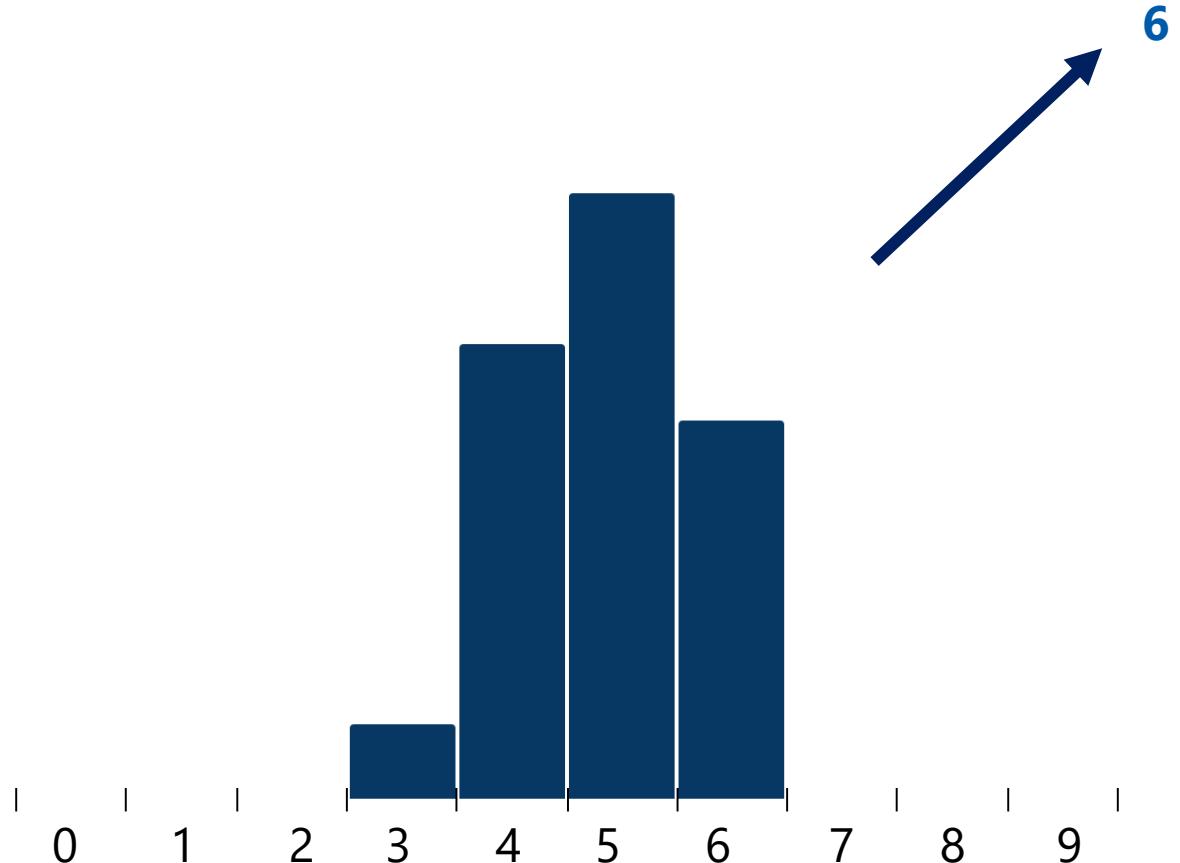
# Random Number Generation

- ▶ We have sample with numbers:  
**3; 5; 4; 4; 4; 5 ; 6 ; 5 ; 4 ; 5;  
4; 5; 6; 5; 6; 5; 5; 6; 6**
- ▶ Want to create a new number alike.



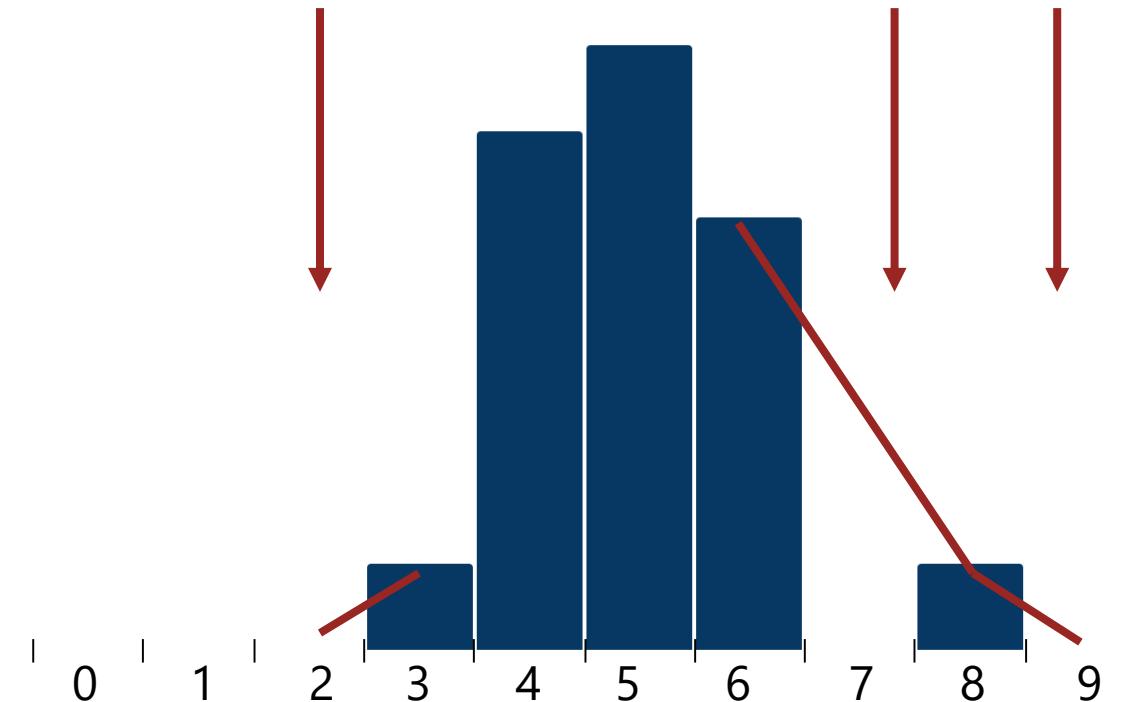
# How we did it?

- ▶ Assume there is a probability density  $p_{\text{true}}(x)$ .
- ▶ Try to estimate  $p_{\text{true}}(x)$  using data and obtain  $p_{\text{data}}(x)$ .
- ▶ Sample from  $p_{\text{data}}(x)$ .



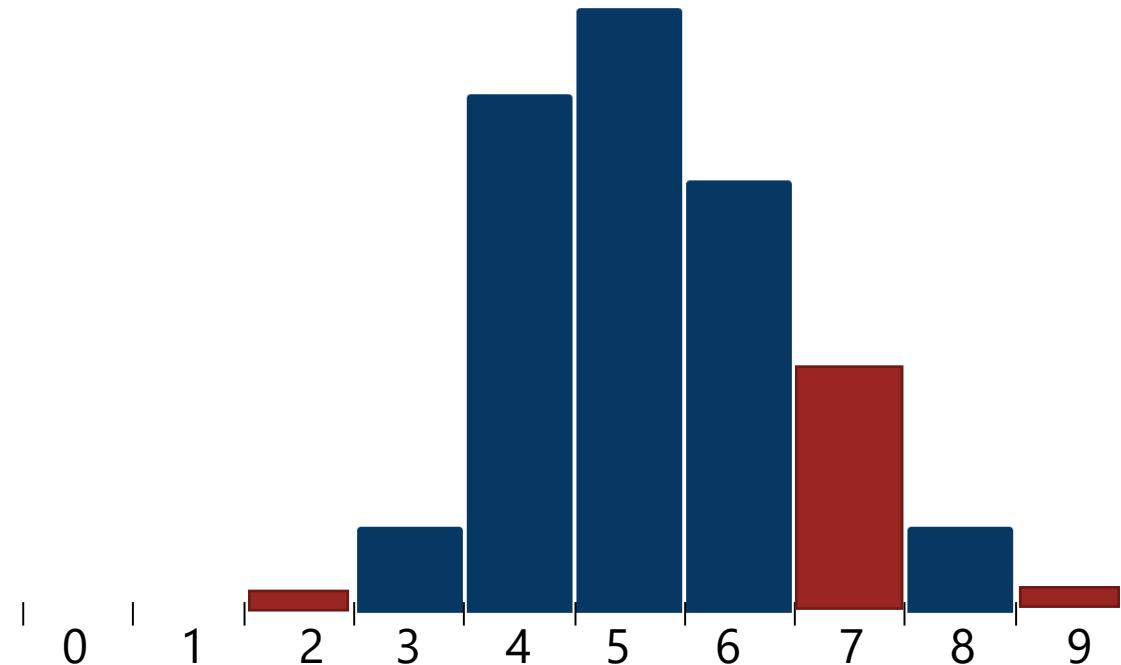
# Random Number Generation

- ▶ We have **different** sample with numbers:  
**3; 5; 4; 4; 4; 4; 5 ; 6 ; 8 ; 4 ; 5;  
4; 5; 6; 5; 6; 5; 5; 6; 5**
- ▶ Want to create a new number alike.



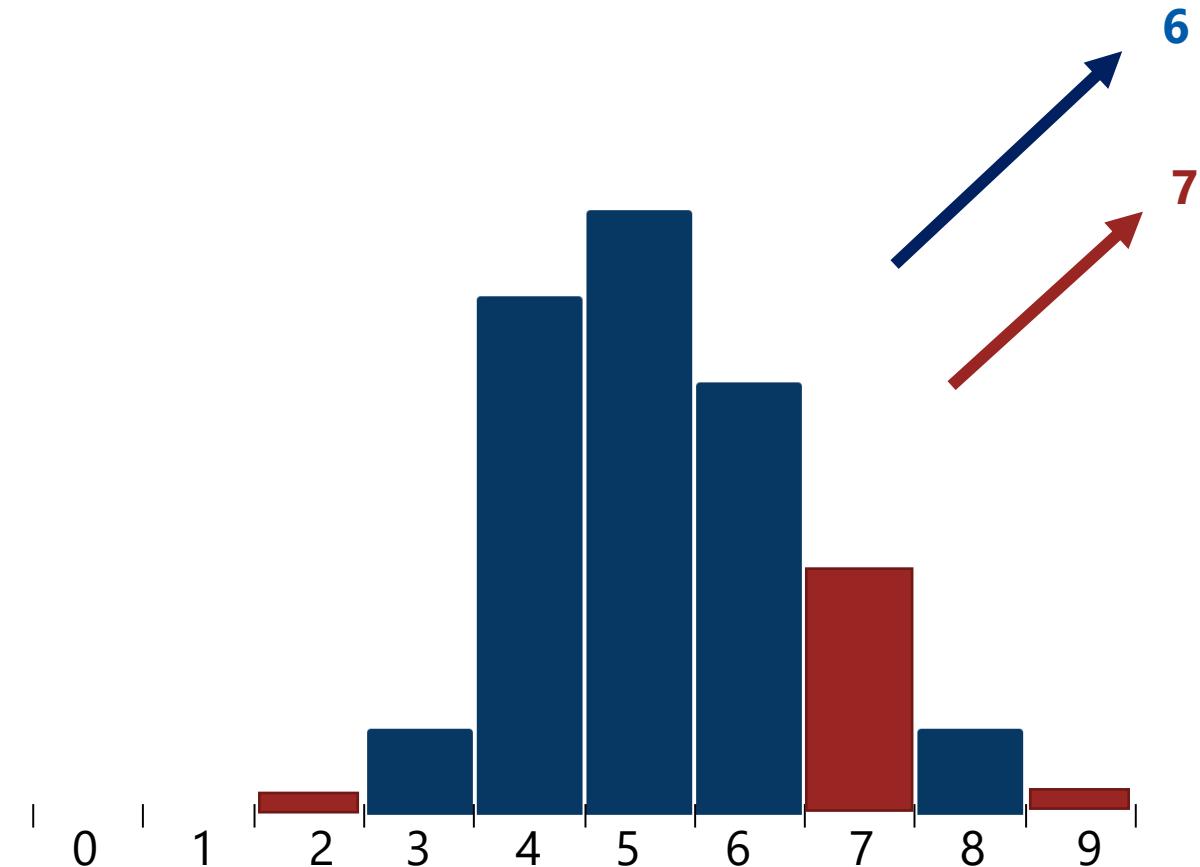
# Random Number Generation

- ▶ We have **different** sample with numbers:  
**3; 5; 4; 4; 4; 4; 5 ; 6 ; 8 ; 4 ; 5;**  
**4; 5; 6; 5; 6; 5; 5; 6; 5**
- ▶ Want to create a new number alike.



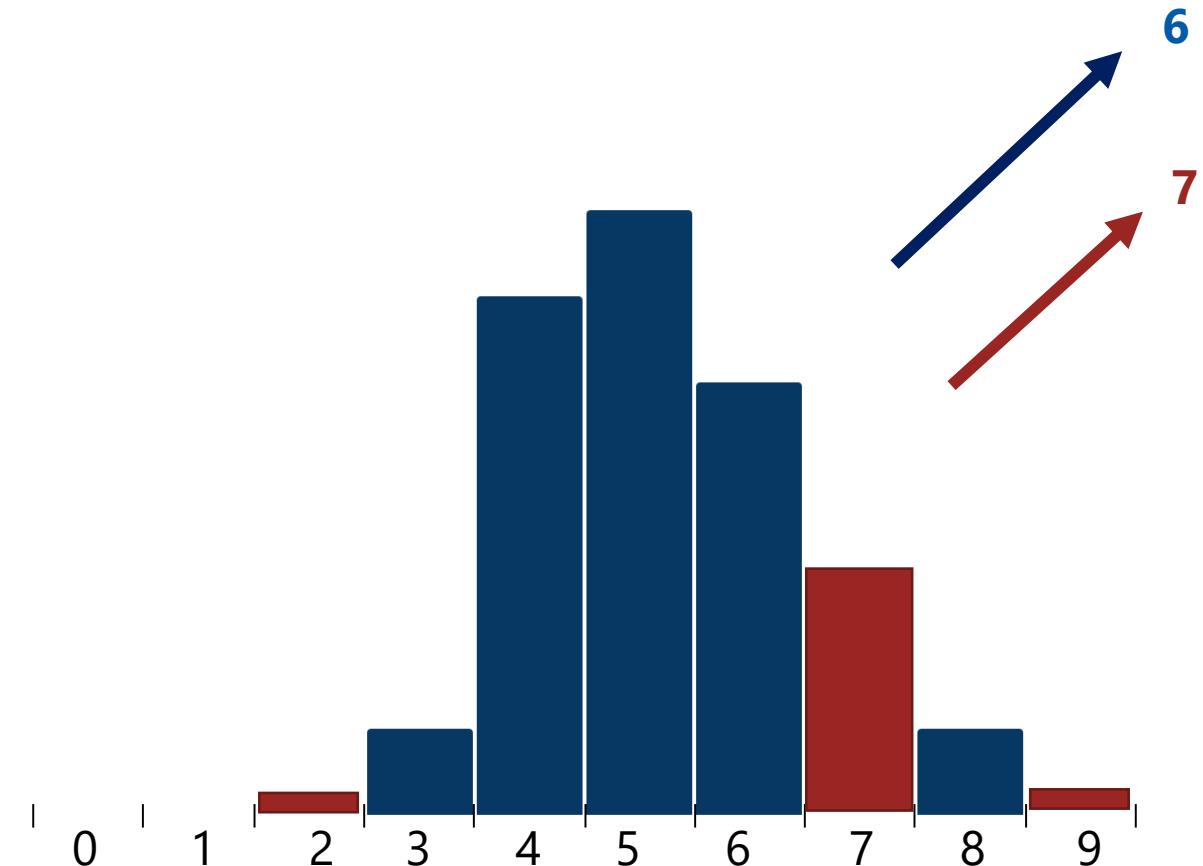
# Random Number Generation

- ▶ We have **different** sample with numbers:  
**3; 5; 4; 4; 4; 4; 5 ; 6 ; 8 ; 4 ; 5;  
4; 5; 6; 5; 6; 5; 5; 6; 5**
- ▶ Want to create a new number alike.



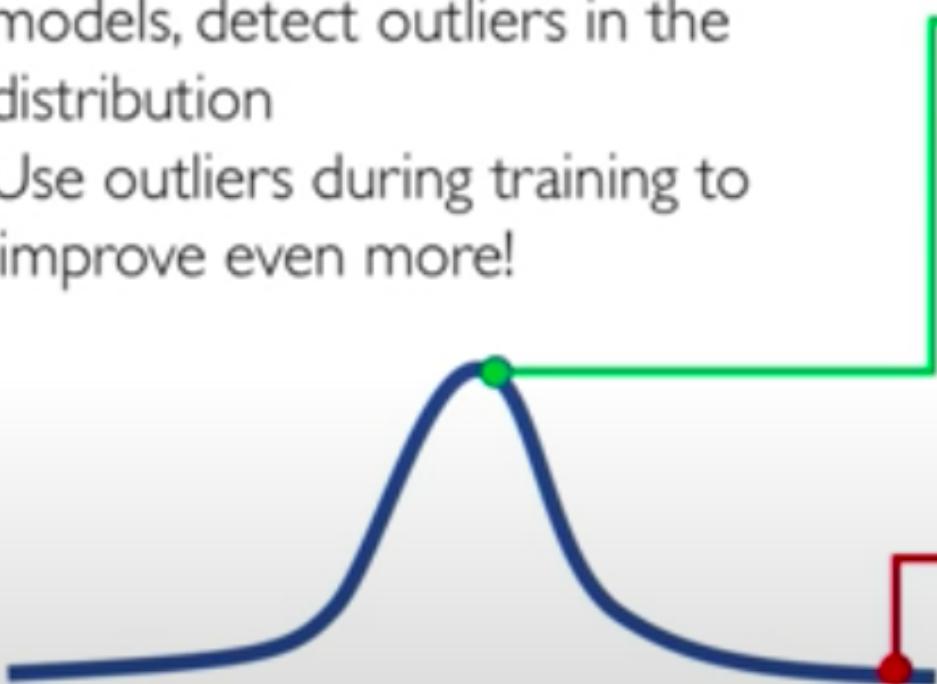
# Random Number Generation

- ▶ Assume there is a probability density  $p_{\text{true}}(x)$ .
- ▶ **Choose interpolation model.**
- ▶ Try to estimate  $p_{\text{true}}(x)$  using data and obtain  $p_{\text{data}}(x)$ .
- ▶ Sample from  $p_{\text{data}}(x)$ .



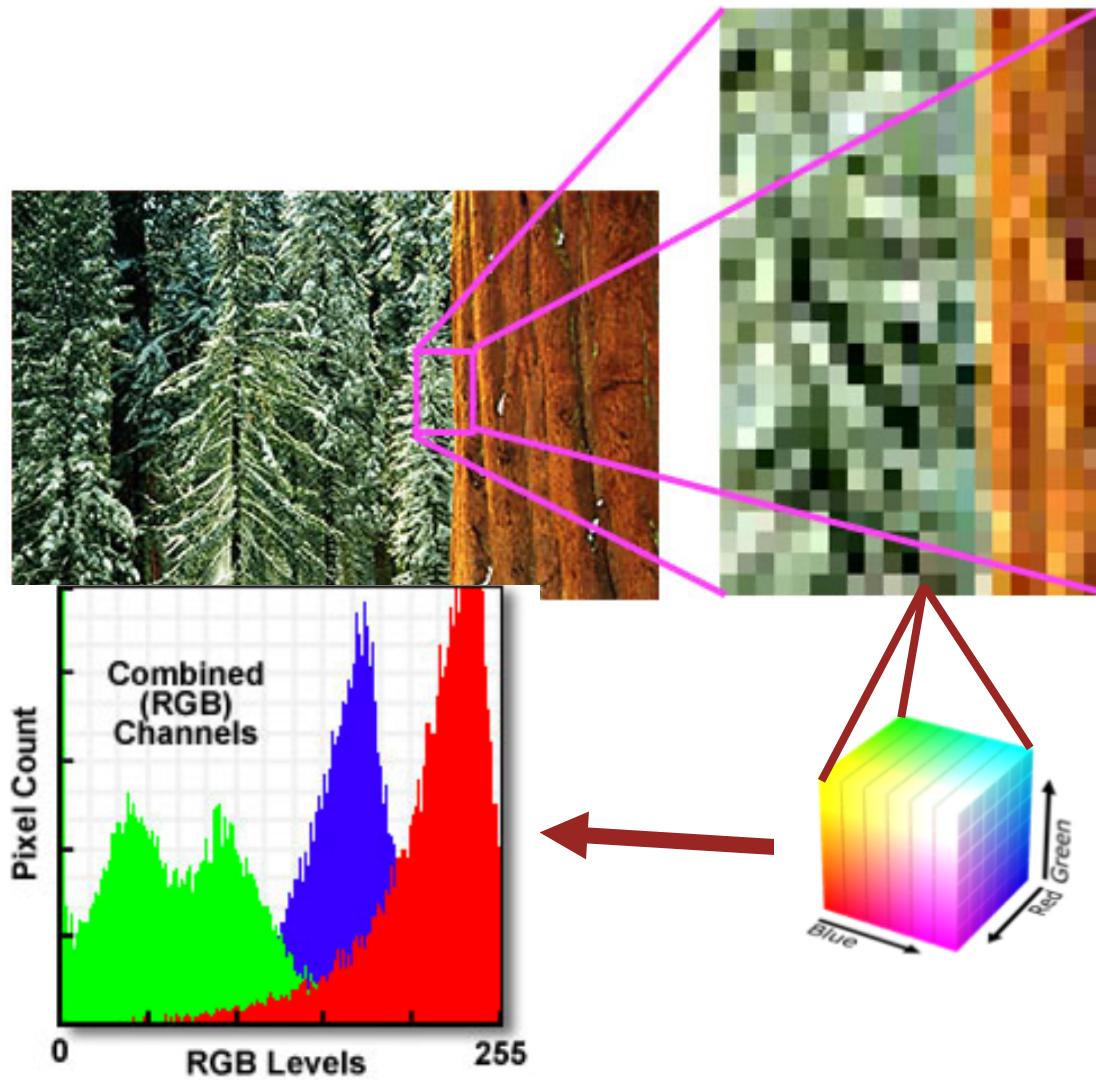
# Case Study: Anomaly Detection

- **Problem:** How can we detect when we encounter something new or rare?
- **Strategy:** Leverage generative models, detect outliers in the distribution
- Use outliers during training to improve even more!



<http://introtodeeplearning.com/>

# More Complicated Case: Figures



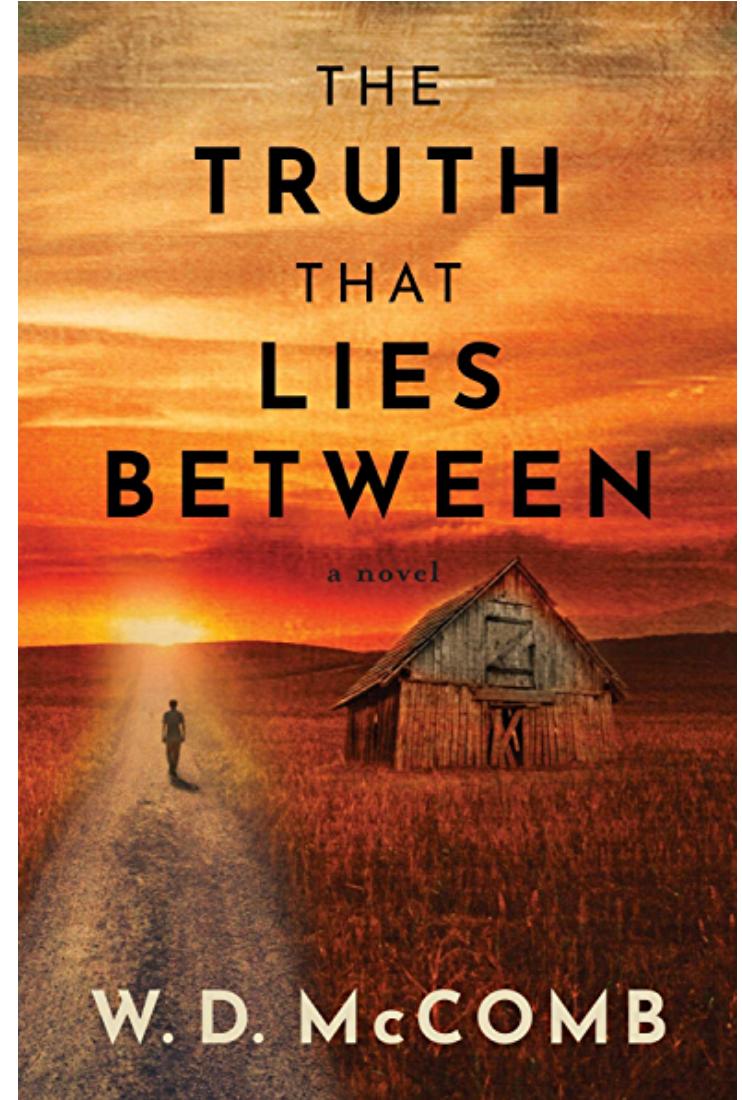
- ▶ Figure consists of pixels.
- ▶ One can use this representation.
- ▶ Each pixel is encoded by 3 colours.
- ▶ **Multi-modal distribution.**
- ▶ **Multidimensional problem.**

# Number of Parameters

- ▶ Handwritten digits dataset.
- ▶ Only black and white pixels.
- ▶ Number of pixels 28X28.
- ▶ Number of possible states:

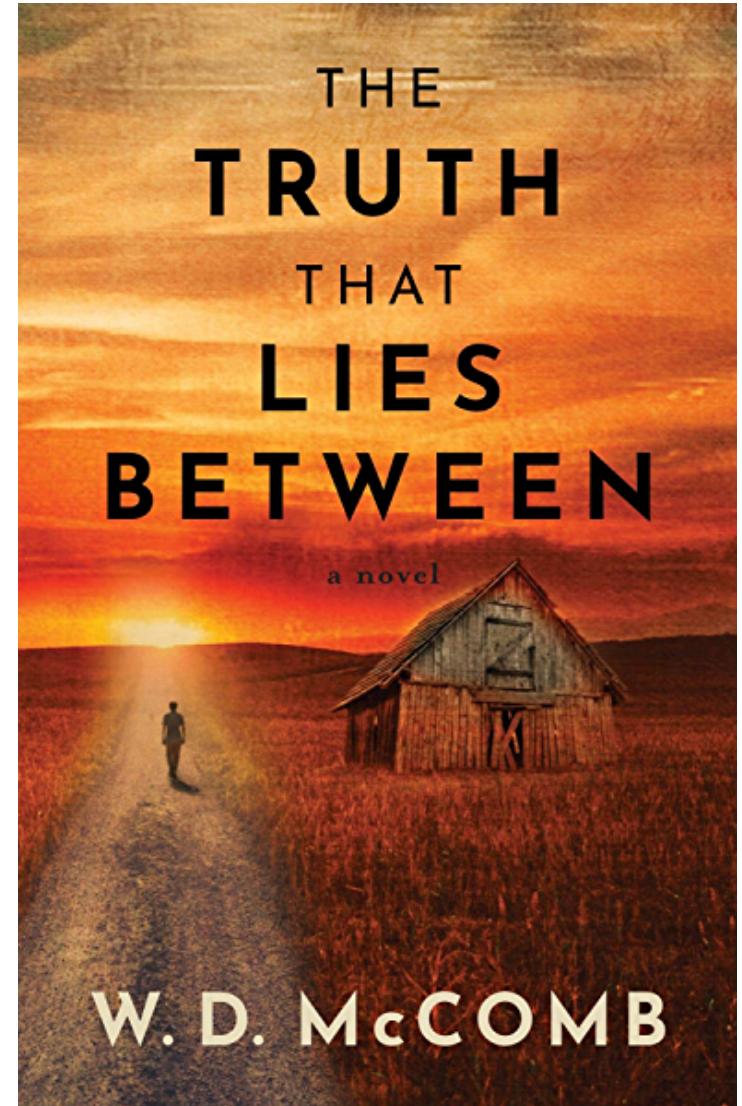
$$2 \times 2 \times 2 \times \dots \times 2 = 2^n.$$

- ▶ **Number of parameters:**  
 $2^n - 1.$
- ▶ **For Independent pixels:**  
**n.**



# Generative model: Final Touch

- ▶ Assume there is a probability density  $p_{\text{true}}(x)$ .
- ▶ Choose interpolation model.
- ▶ **Reduce number of dimensions.**
- ▶ Try to estimate  $p_{\text{true}}(x)$  using data and obtain  $p_{\text{data}}(x)$ .
- ▶ Sample from  $p_{\text{data}}(x)$ .



[Amazon](#)

# Generative model: Problem Statement

Three major tasks, given a generative model  $f$  from a class of models  $\mathcal{F}$  :

- ▶ **Estimation**: find the  $f$  in  $\mathcal{F}$  that best matches observed data.
- ▶ **Evaluate Likelihood**: compute  $f(z)$  for a given  $z$ .
- ▶ **Sampling**: drawing from  $f$ .

[S. Nowozin et al. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization](#)

# Generative model vs Discriminative model

## Discriminative models

- › learn  $\mathbb{P}(y|x)$
- › Directly characterizes the decision boundary between classes only
- › Examples: Logistic Regression, SVM, etc

## Generative models

- › learn  $\mathbb{P}(x|y)$  (and eventually  $\mathbb{P}(y, x)$ )
- › Characterize how data is generated (distribution of individual class)
- › Examples: Naive Bayes, HMM, etc.

<https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf>

# Chapter outcome

- ▶ Generative modeling is a distinct task in machine learning.
- ▶ Mathematically, it aims to reconstruct the probability density, from which the given dataset was sampled.

# Early Generative Models



# First ideas

For parametric model.

- ▶ **Inversion sampling.** For  $x$  with CDF  $F_X(x)$  :

$$z \sim \text{Unif}(0; 1); x = F_X^{-1}(z).$$

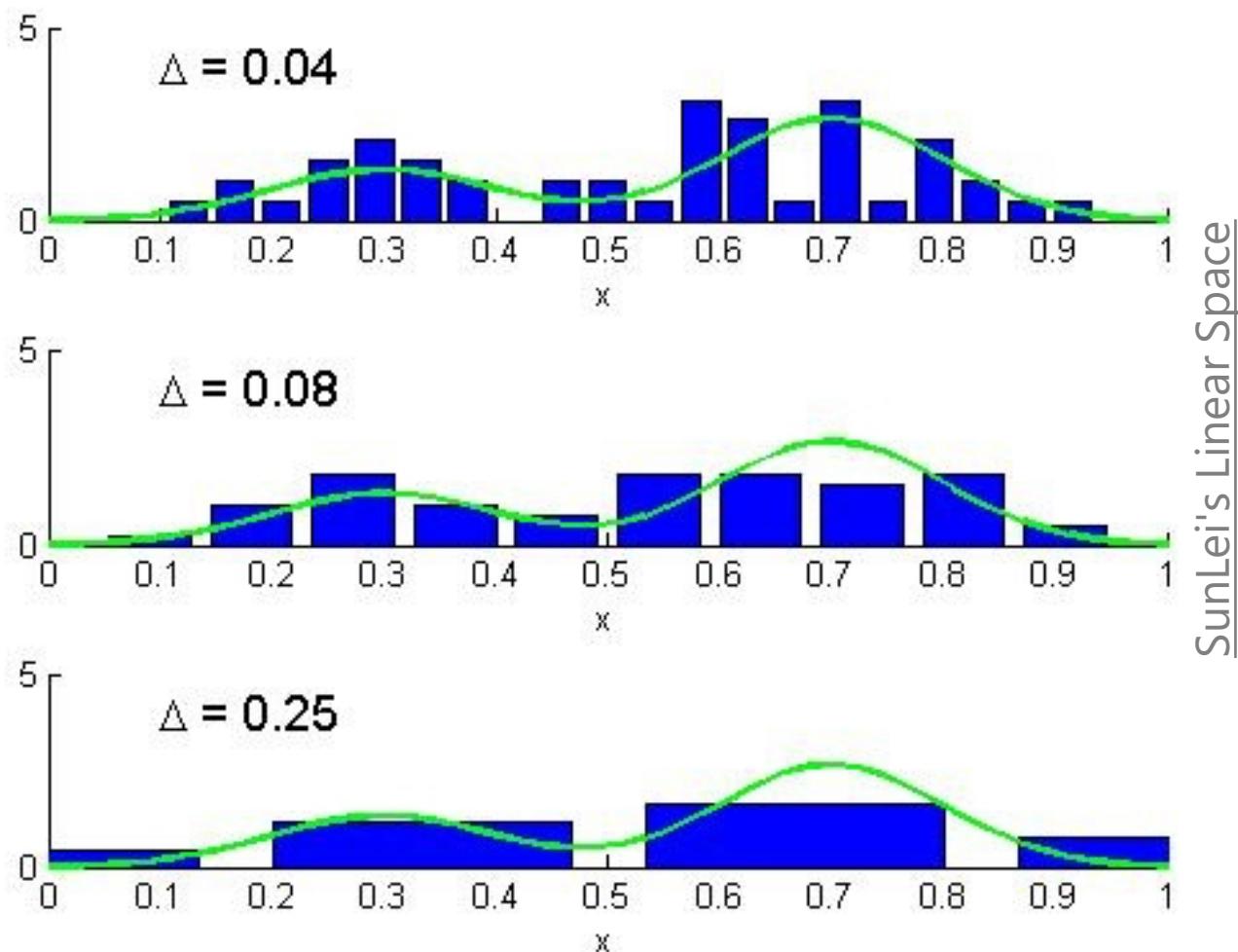
- ▶ **Works in multidimensions.** Sample successively.

- Generate  $X$  from the marginal  $p_X(x) = \int p_{X,Y}(x, y) dy$ .
- Generate  $Y$  given  $X = x$  from the conditional  $p_{Y|X}(y|x) = \frac{p_{x,y}(x,y)}{p_X(x)}$ .

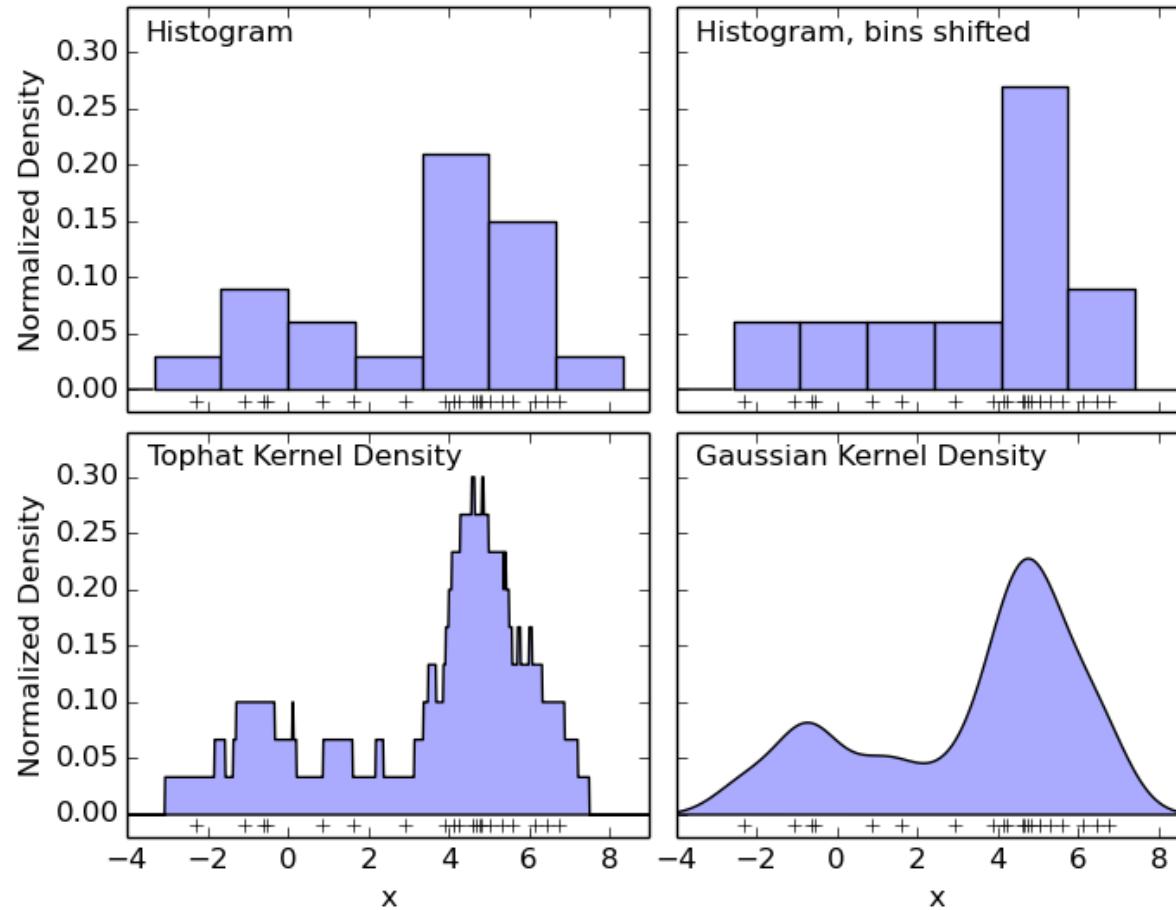
For 1D Gaussian model, the convergence is  $\mathcal{O}\left(\frac{1}{\sqrt{n}}\right)$ .

# “Non-parametric” Approaches

- ▶ Histograms can be used.
- ▶ Need to choose optimal bin size.
- ▶ Smaller bins for approximate constant estimate.
- ▶ Larger bins for less fluctuations.
- ▶ Can be chosen using empirical risk.



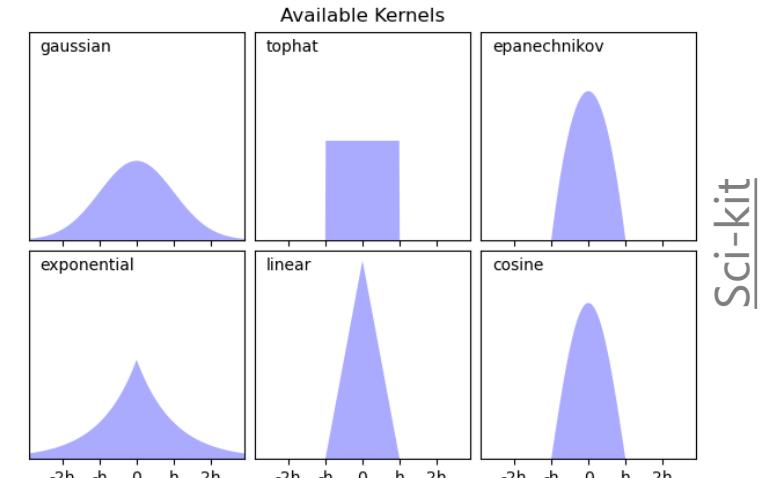
# Kernel-density estimation



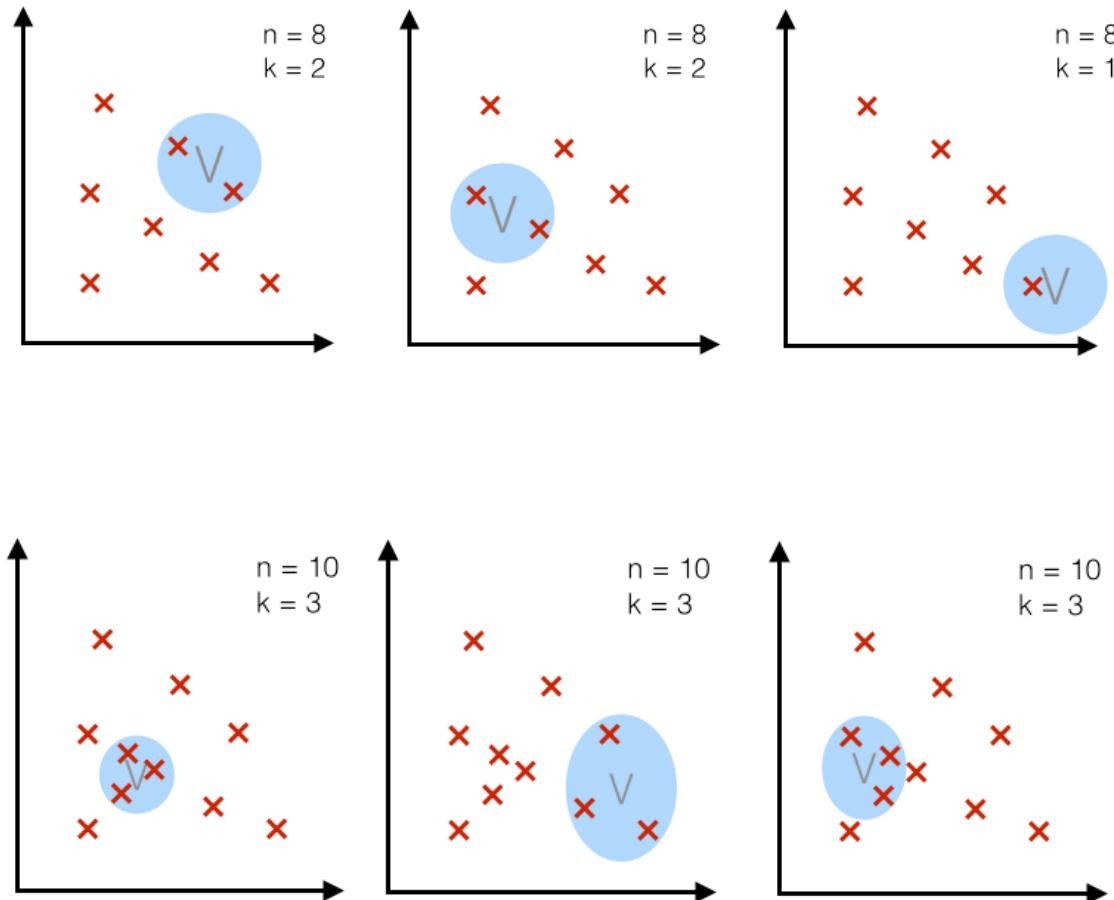
- ▶ Assign **every** event a weight.
- ▶ Smooth between events.
- ▶ Kernel Density Estimation:

$$\hat{p}_n(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x-x_i}{h}\right),$$

$K$  – some kernel,  $h$  – bandwidth.



# KDE2KNN



- ▶ With fixed volume kernel outliers can lead to fluctuations in  $\hat{p}(x)$ .
- ▶ Vary kernel volume to cover k nearest neighbors.
- ▶ Better coverage of tails.

[S. Raschka's blog](#)

# KDE and kNN Optimal Parameter Choice

Minimize integral MSE (or L2 risk function) to determine optimal parameter and convergence:

$$MSE(\hat{p}_n(x_0)) = \text{bias}^2(\hat{p}_n(x_0)) + Var(\hat{p}_n(x_0)).$$

$$MISE(\hat{p}_n) = \int MSE(\hat{p}_n(x)) dx$$

This is not a straightforward task (need cross-validation selector) but can be solved under some conditions.

$$MISE_{opt}(\hat{p}_n) = \mathcal{O}(n^{-\frac{4}{4+d}}).$$

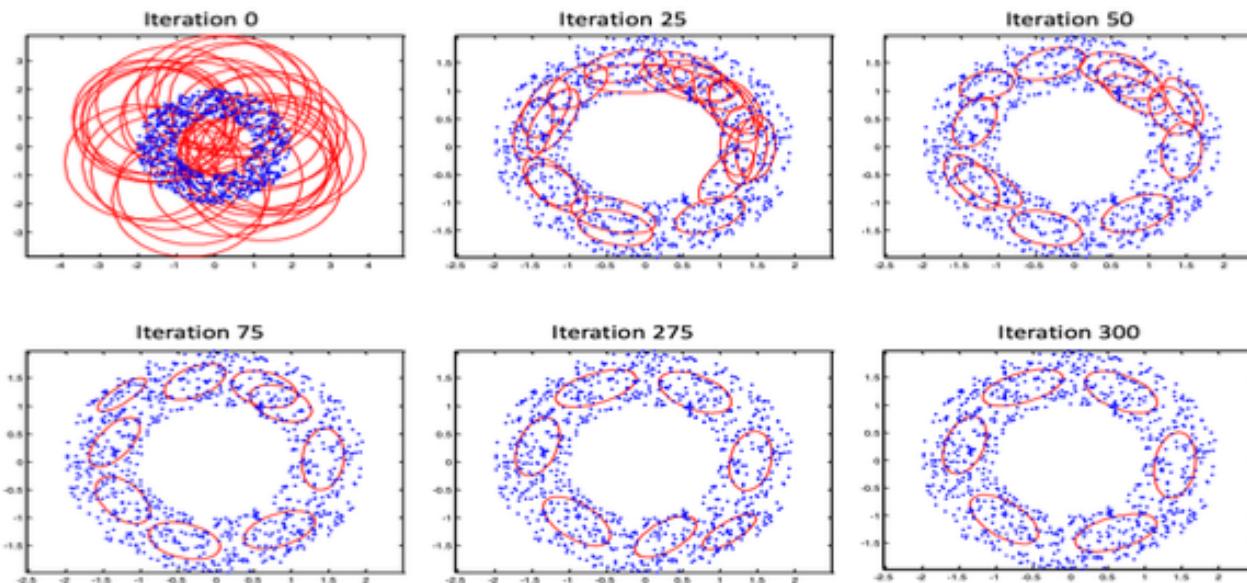
when  $d$  is large, the optimal convergence rate is very very slow.

# KDE and kNN Summary

- ▶ Efficient in low dimensional estimation.
- ▶ Controllable convergence rate for bias or variance but the overall rate is similar.
- ▶ Mixture of KDE and kNN are available.
- ▶ To speed up the convergence, one can attempt to find manifolds in the  $d$ -dimension.
- ▶ Fairly hard to sample and keep the model in memory.

# Gaussian Mixture Model

Training set:  $n = 900$  examples from a uniform pdf inside an annulus,  
model: GMM with  $K = 30$  Gaussian components



- ▶ Reduce number of Gaussians.
- ▶ Infer Gaussian parameters from data.
- ▶ Estimate density:

$$\hat{p}_n(x) = \sum_{l=1}^K \pi_l \phi(x; \mu_l, \sigma_l)$$

$K$  Gaussian distribution  $\phi$  is used.

- ▶ **Need EM-algorithm**

# GMM Summary

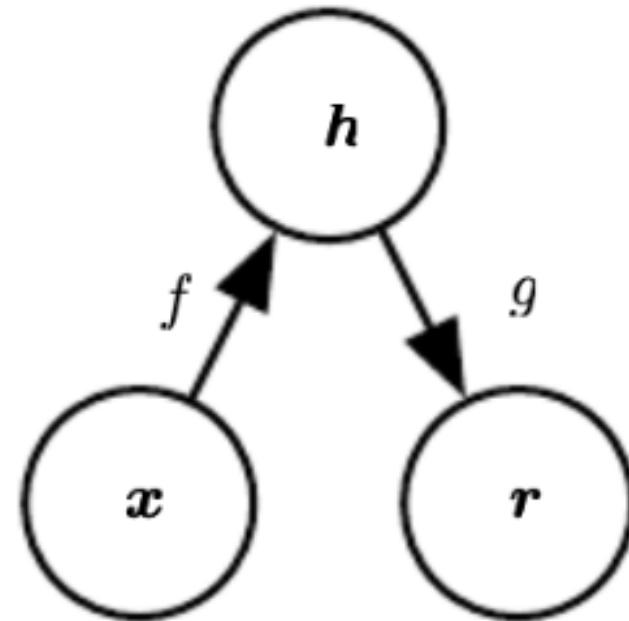
- ▶ Better convergence rate than non parametric.
- ▶ Identifiability problem. We cannot distinguish between two exchanged solutions. .
- ▶ Computation problem. We need to use EM algorithm to find solution.
- ▶ Choice of K. A very difficult task, one may use a model selection technique to choose it, however, no simple rule exists.
- ▶ Does not really converge to a true PDF.

# Autoencoders

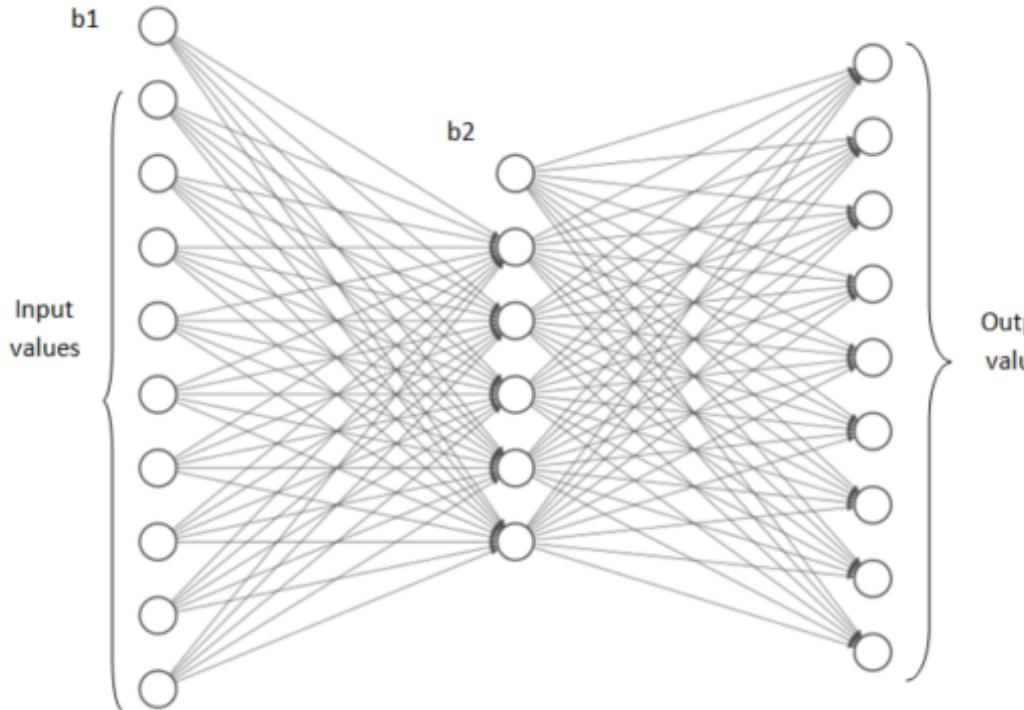


# Rationale

- ▶ Need a trick to reduce number of dimensions.
- ▶ Simplest idea: have a network with narrow hidden layer –  
**Autoencoder!**
  - Encoder:  $h = f(x)$ ;
  - Decoder:  $r = g(h)$ .
- ▶ Want to find transformation  
$$g(f(x)) = x.$$



# Vanilla Autoencoder



- ▶ For binary data  $x \in R^D$  and a one-layer network:

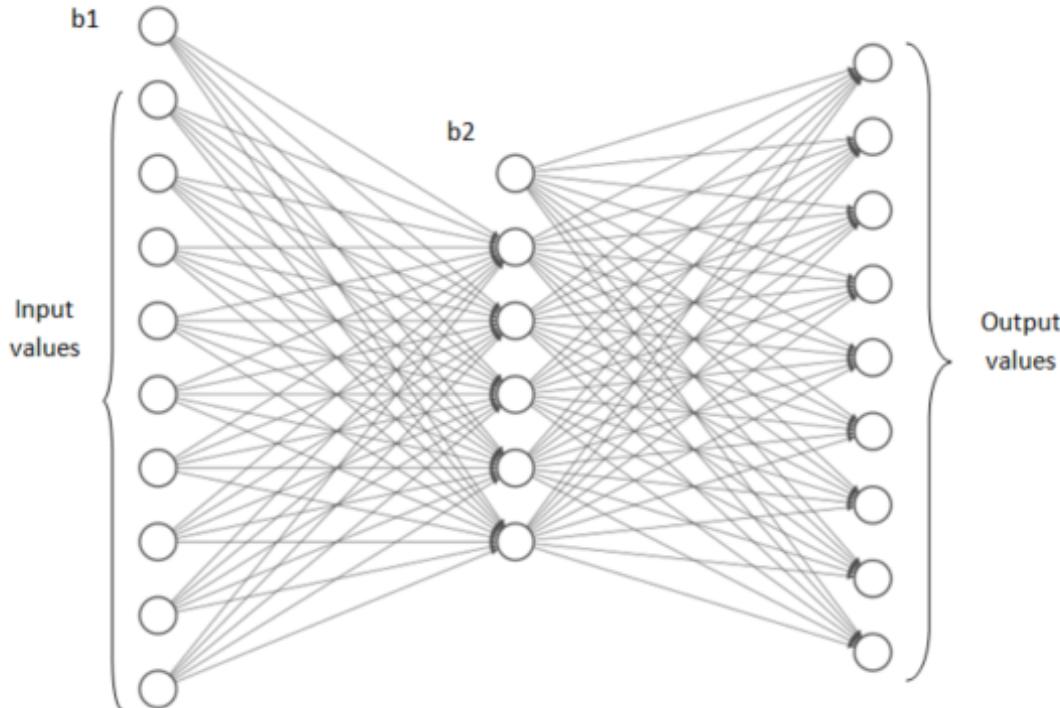
$$h(x) = g(b + Wx);$$

$$\hat{x} = \sigma(c + Vh(x)).$$

- ▶ A typical loss is cross-entropy:

$$\mathcal{L}(x) = \sum -x_d \log(\hat{x}_d) - (1 - x_d) \log(1 - \hat{x}_d)$$

# Vanilla Autoencoder



- ▶ A typical loss is cross-entropy:
$$\mathcal{L}(x) = \sum -x_d \log(\hat{x}_d) - (1 - x_d) \log(1 - \hat{x}_d)$$
- ▶ Interpret  $\mathcal{P}(x_d = 1) = \hat{x}_d$ , and cross-entropy as negative log-probability.
- ▶ However,  $e^{-\mathcal{L}(x)}$  does not sum to 1.
- ▶ **The proposed generative model fails to reconstruct probability.**

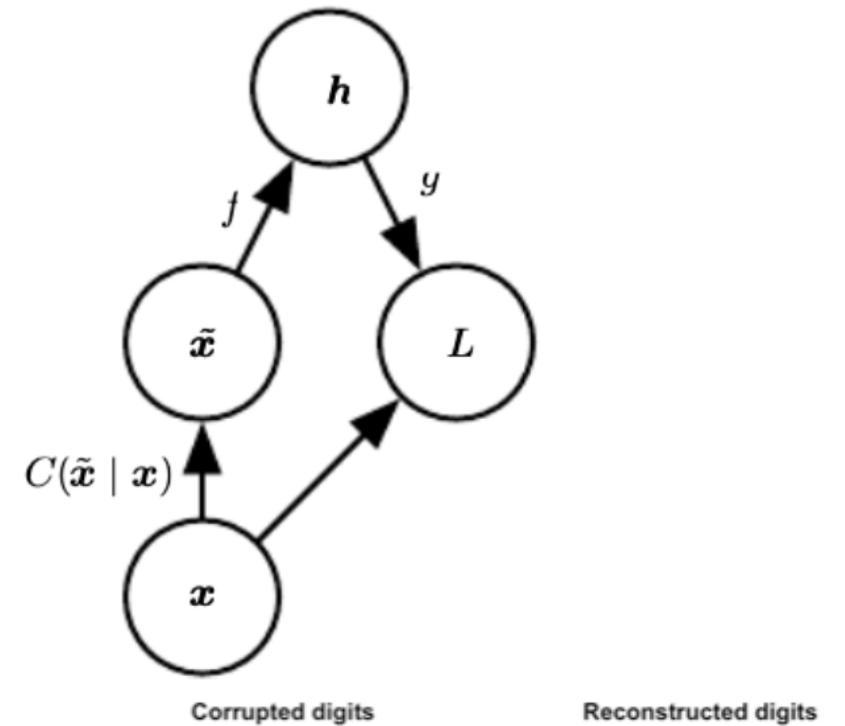
# Denoising Autoencoders as Generative model

- ▶ Artificially add noise  $\tilde{x} \sim C(\tilde{x}|x)$ .
- ▶ Use

$$\mathcal{L}(\theta) = -E[\log P_\theta(X|\tilde{X})]$$

where expectation is taken over the joint data-generating distribution:

$$P(X, \tilde{X}) = P(X)C(\tilde{X}|X).$$



Original digits



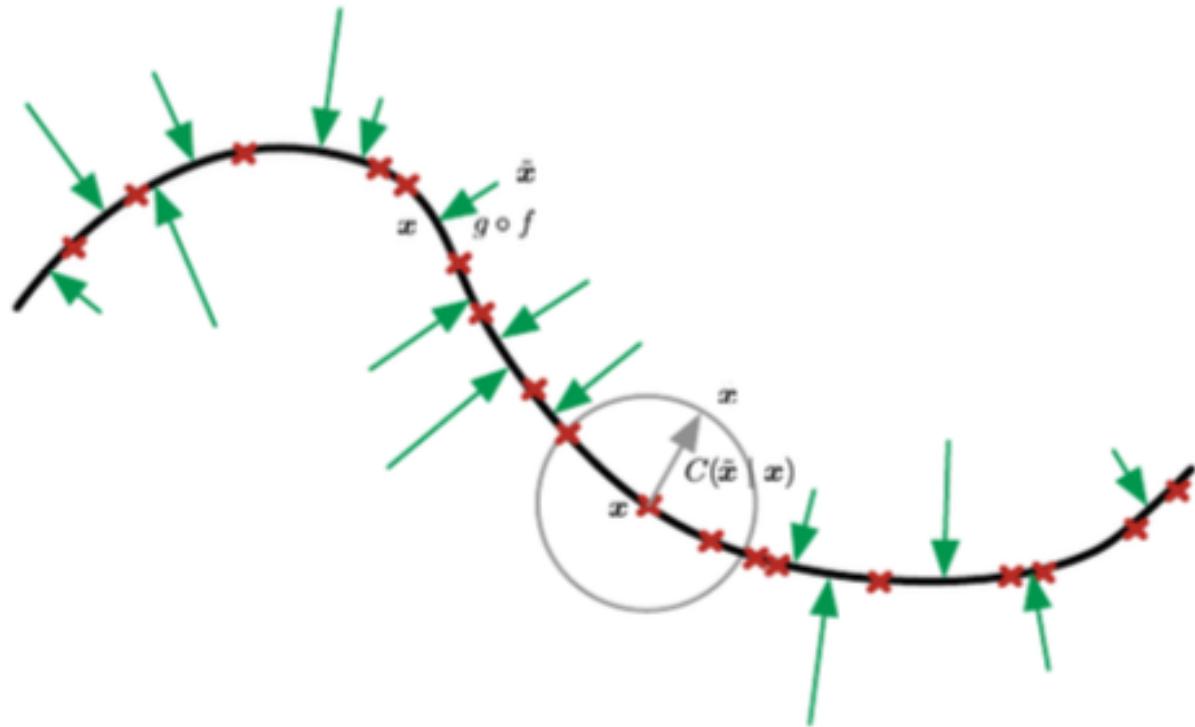
Corrupted digits



Reconstructed digits

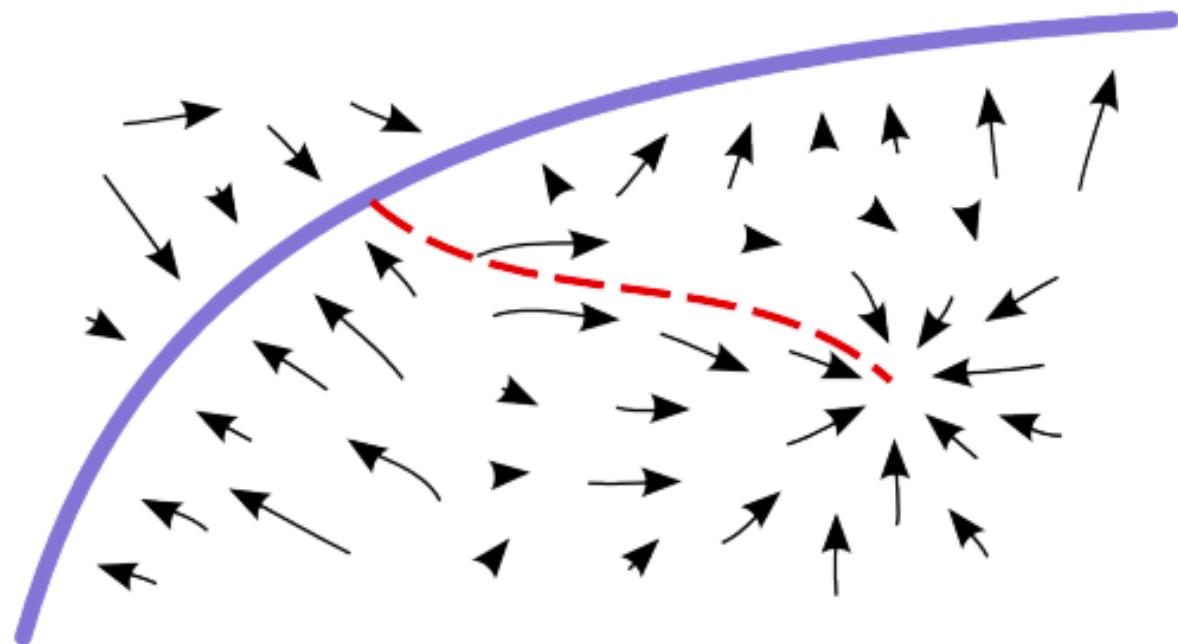


# Signal Manifold



- ▶ The true signal is always situated on a manifold inside the  $R^D$  space.
- ▶ Denoising autoencoder is trained to map a corrupted data point  $\tilde{x}$  back to the original data point  $x$ .

# Constructing Probabilistic Model



- ▶ Construct a Markov Chain:  
$$X_t \sim P_\theta(X|\tilde{X}_{t-1});$$
  
$$X_t \sim C(\tilde{X} | X_t).$$
- ▶ Start with random point  $X_0$  and sample new data points from it.
- ▶ Walk-back return algorithm
- ▶ For infinitesimal noise equivalent to KDE.

[Y. Bengio et al., Generalized Denoising Auto-Encoders as Generative Models, NIPS-13](#)

# MNIST results



Figure 4: Successive samples generated by Markov chain associated with the trained DAEs according to the plain sampling scheme (left) and walkback sampling scheme (right). There are less “spurious” samples with the walkback algorithm.

[Y. Bengio et al., Generalized Denoising Auto-Encoders as Generative Models, NIPS-13](#)

# AE Summary

- ▶ Autoencoders are able to find hidden representations of the observables.
- ▶ Vanilla Autoencoders are not suitable for generative modeling.
- ▶ Autoencoders can produce a sampling model with implicit PDF reconstruction.

# Final Summary

- ▶ Generative modeling is a distinct task of machine learning.
- ▶ Several pre-deep learning algorithms can produce reasonable results in the low dimensional data.
- ▶ Denoising Autoencoder is one of the first pseudo-generative models.