Generative models I

Nikita Kazeev

Generative models

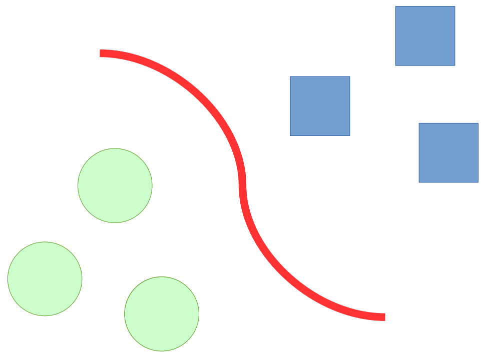
Dataset: (x, y) pairs

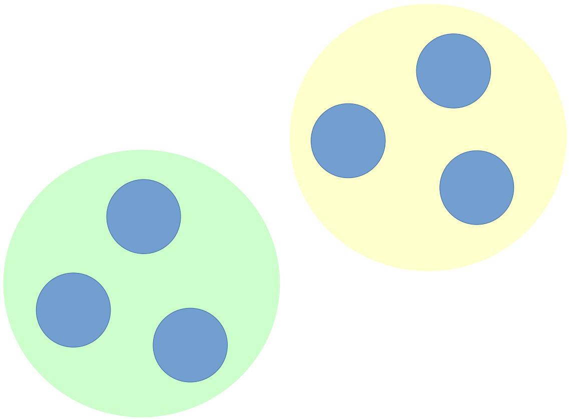
Regression & classification

- Learn mapping $x \to y$

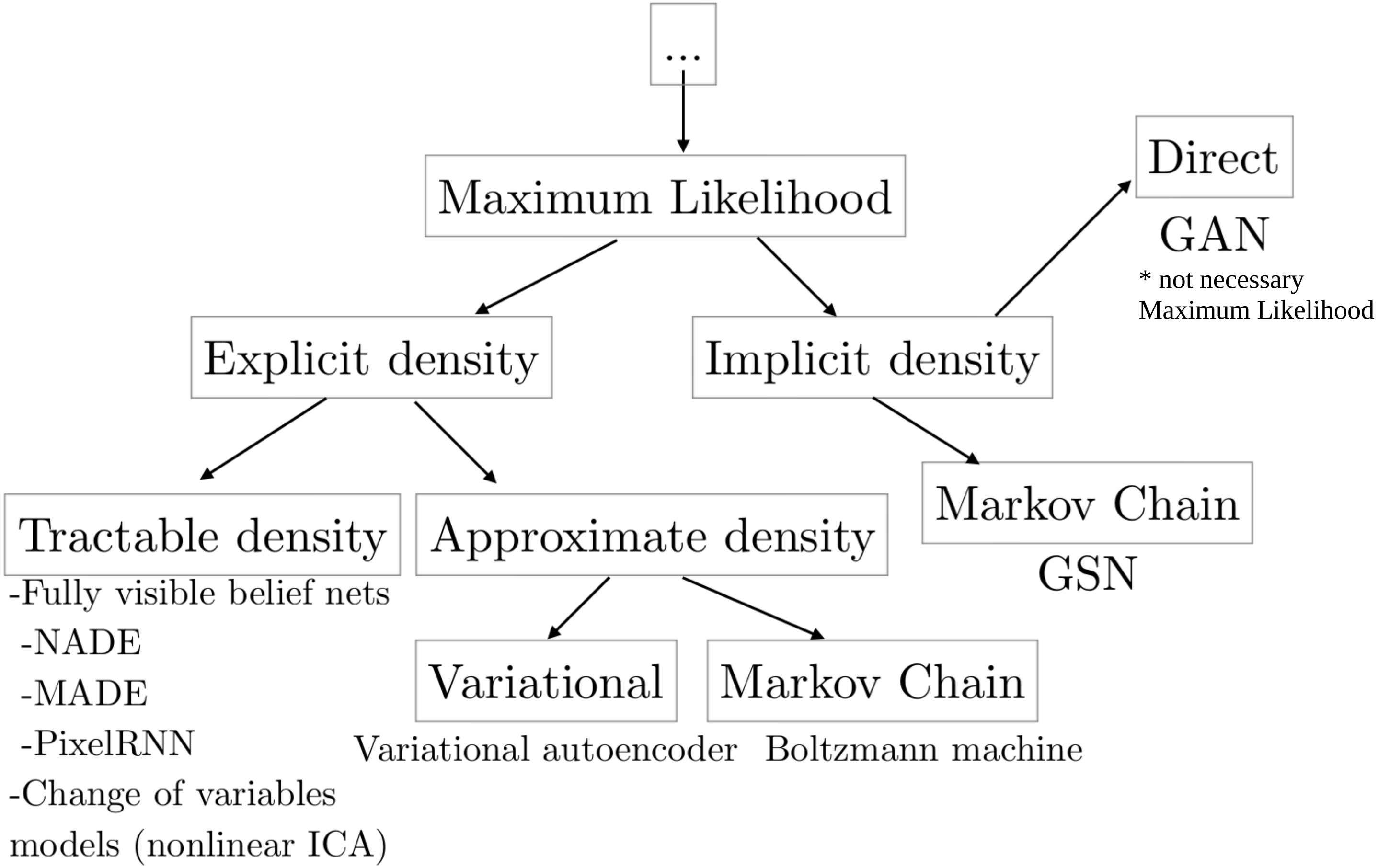
Generative models

- Learn to sample $P(y|x)$
- X can be null, then it's unsupervised learning



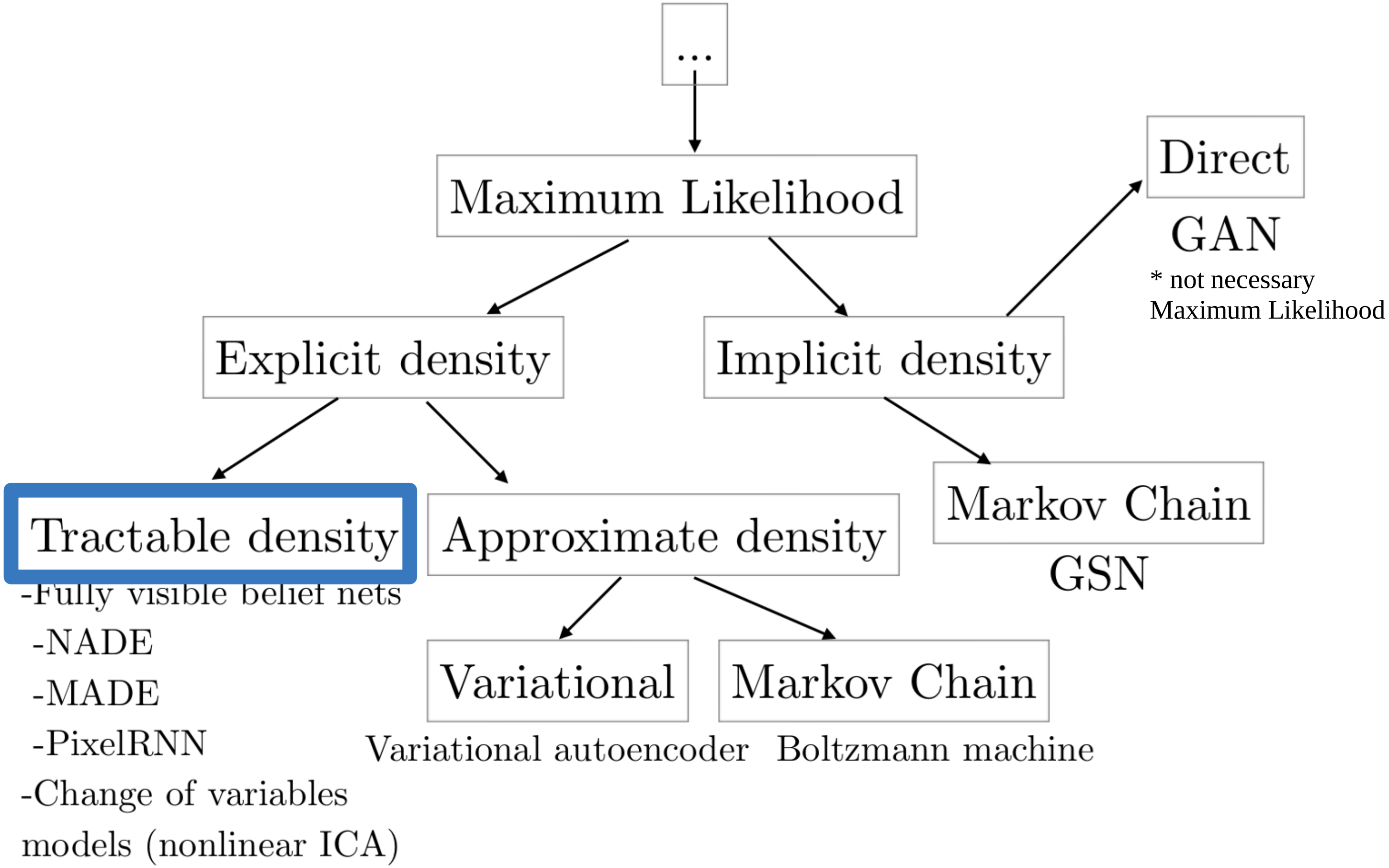


Generative Model Taxonomy



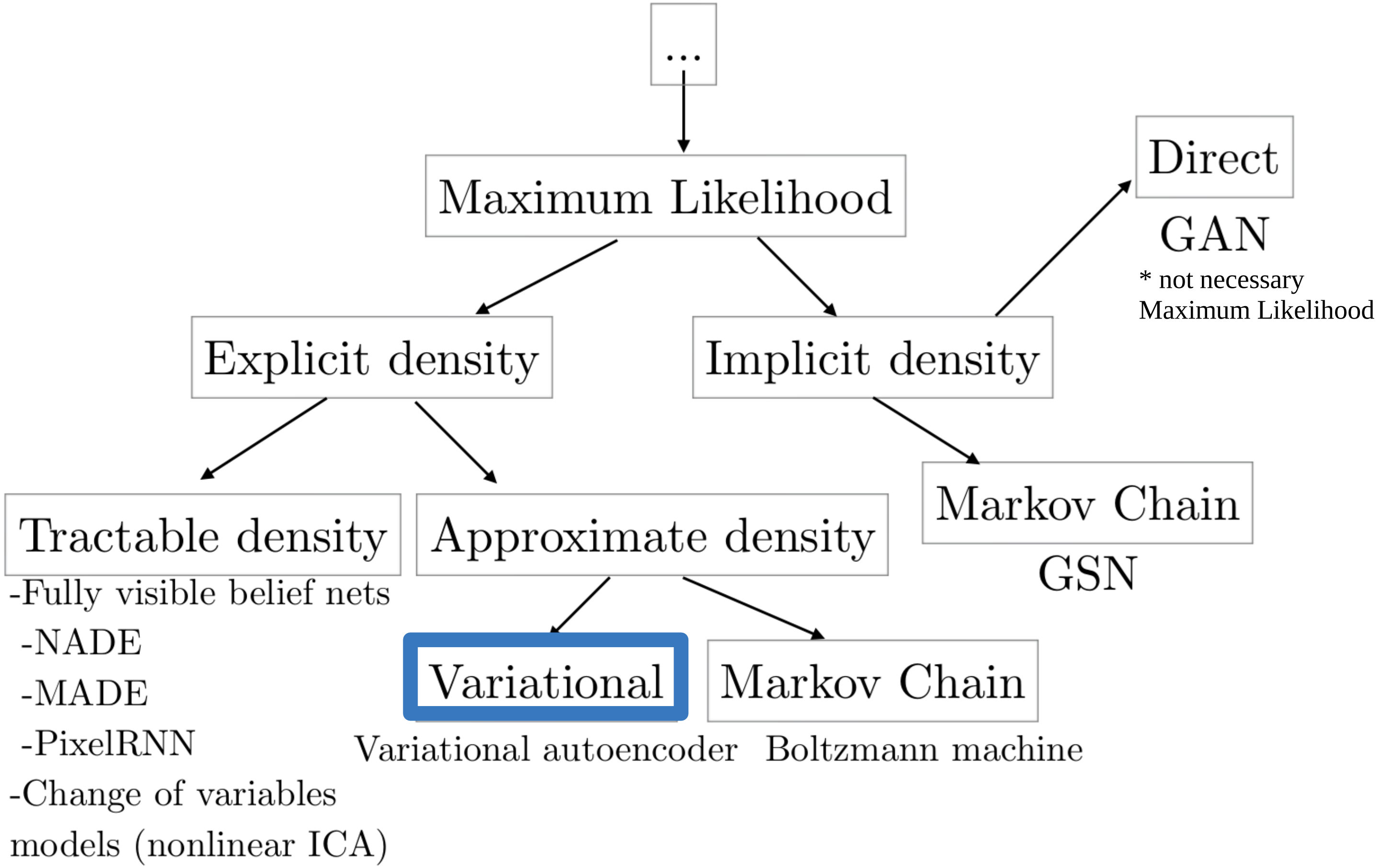
https://arxiv.org/pdf/1701.00160.pdf

Generative Model Taxonomy



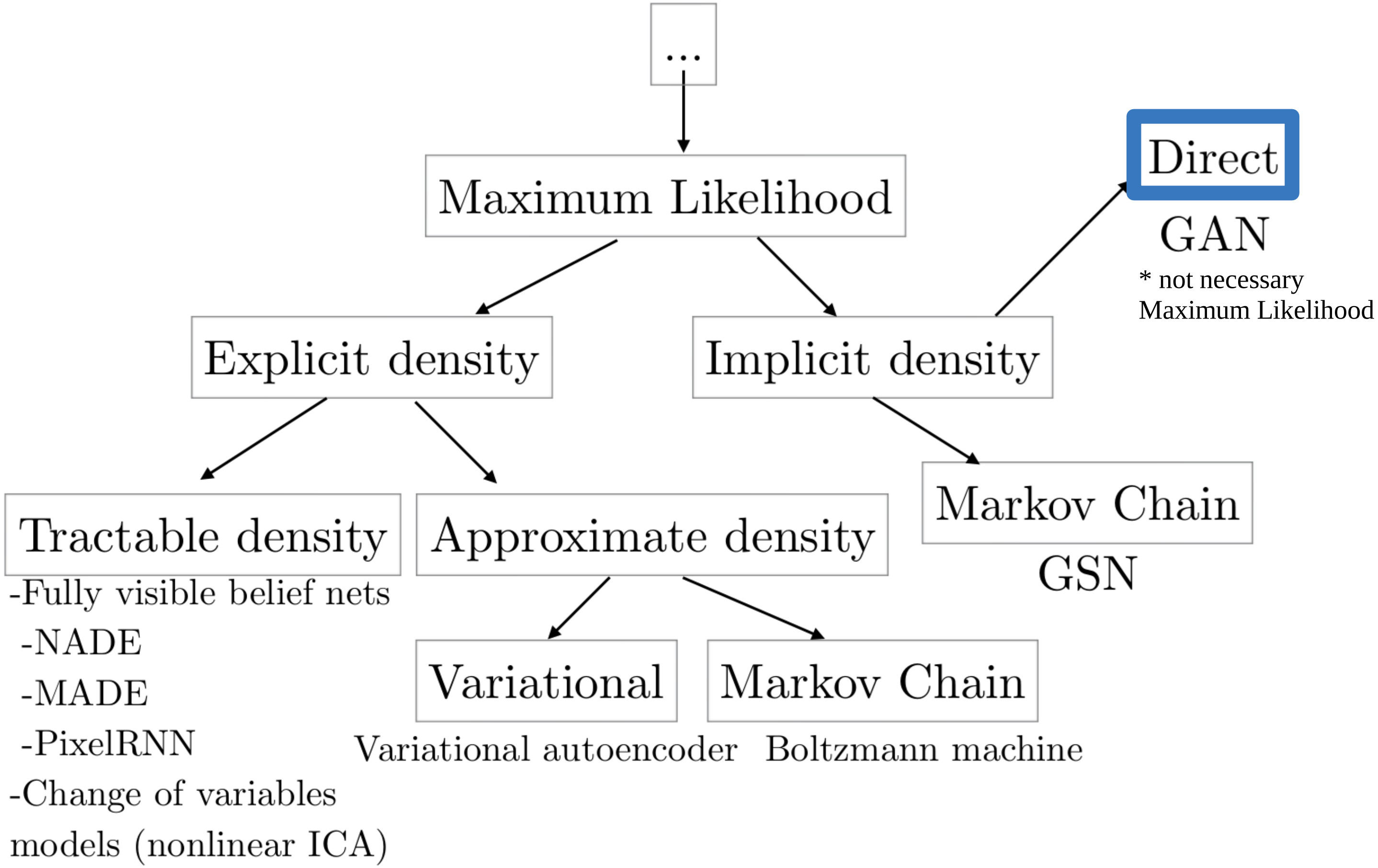
https://arxiv.org/pdf/1701.00160.pdf

Generative Model Taxonomy



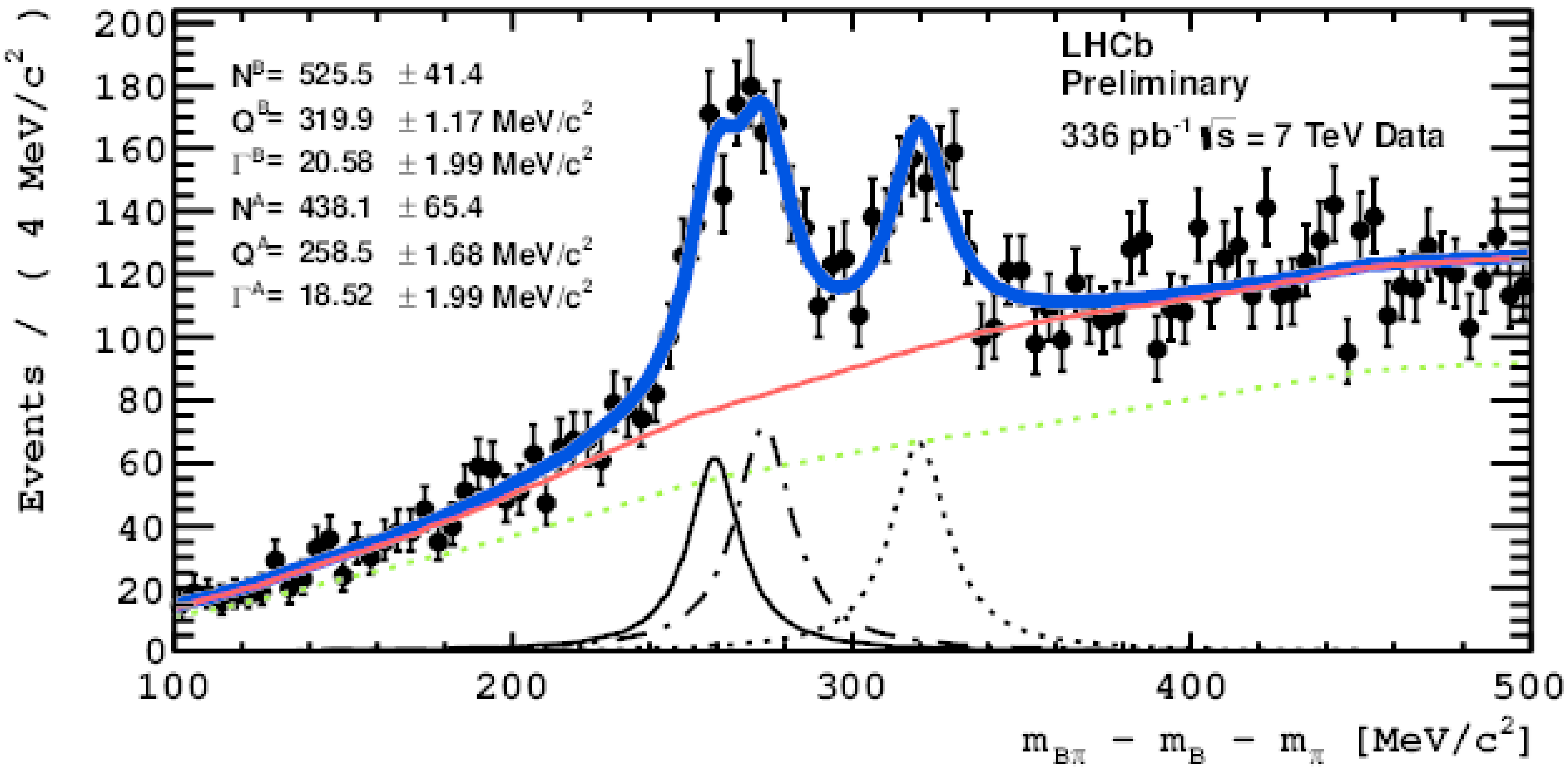
https://arxiv.org/pdf/1701.00160.pdf

Generative Model Taxonomy



https://arxiv.org/pdf/1701.00160.pdf

Good ol' distribution fitting



LHCb collaboration

Good ol' distribution fitting

Given

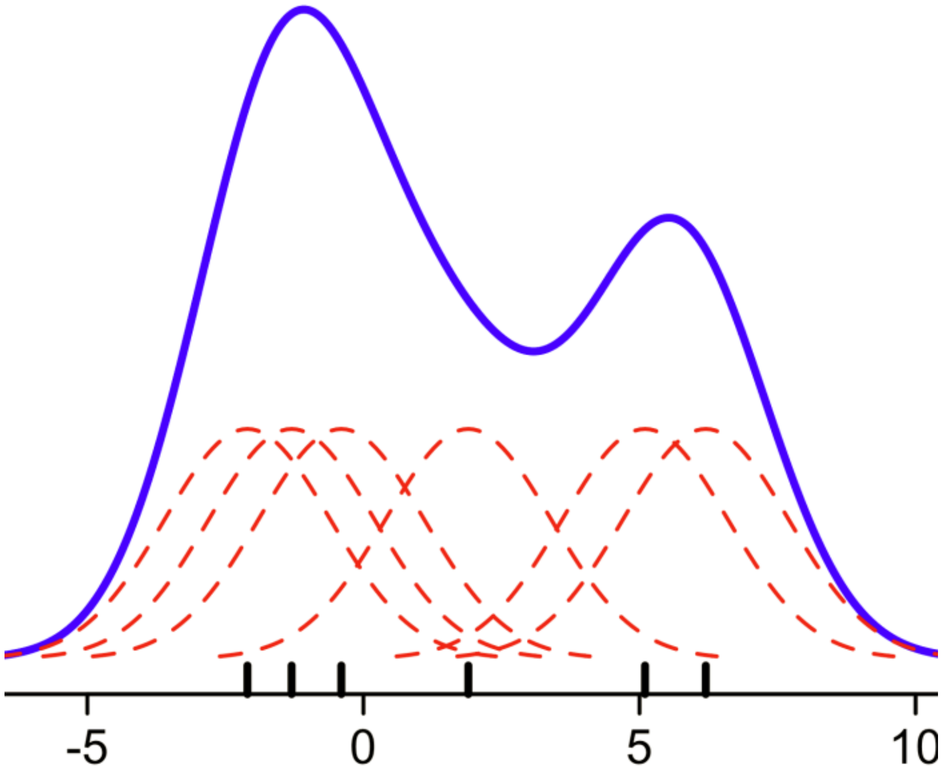
- data points $x_1, \dots, x_n \in \mathbb{R}_\mathrm{m}$
- a parametrized distribution it's supposed to come from $P(x|\theta)$

Find a set of parameters to maximize the empirical likelihood:

- $\max\limits_{\theta} L(\theta|x) = \max\limits_{\theta} \prod_{i=0}^{n} P(x_i|\theta)$

Kernel density

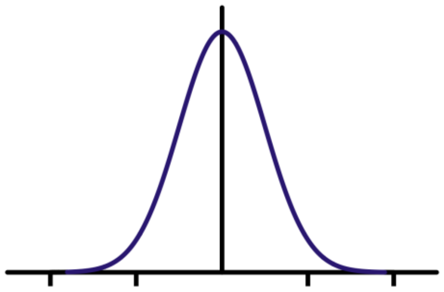
What if we place many Gaussians on the data points and call their sum a PDF?



http://www.spiderfinancial.com/support/documentation/numxl/users-guide/descriptive-statistics/kernel-density-estimation-kde-tutorial

Kernel density

Kernel function, usually looks like this:

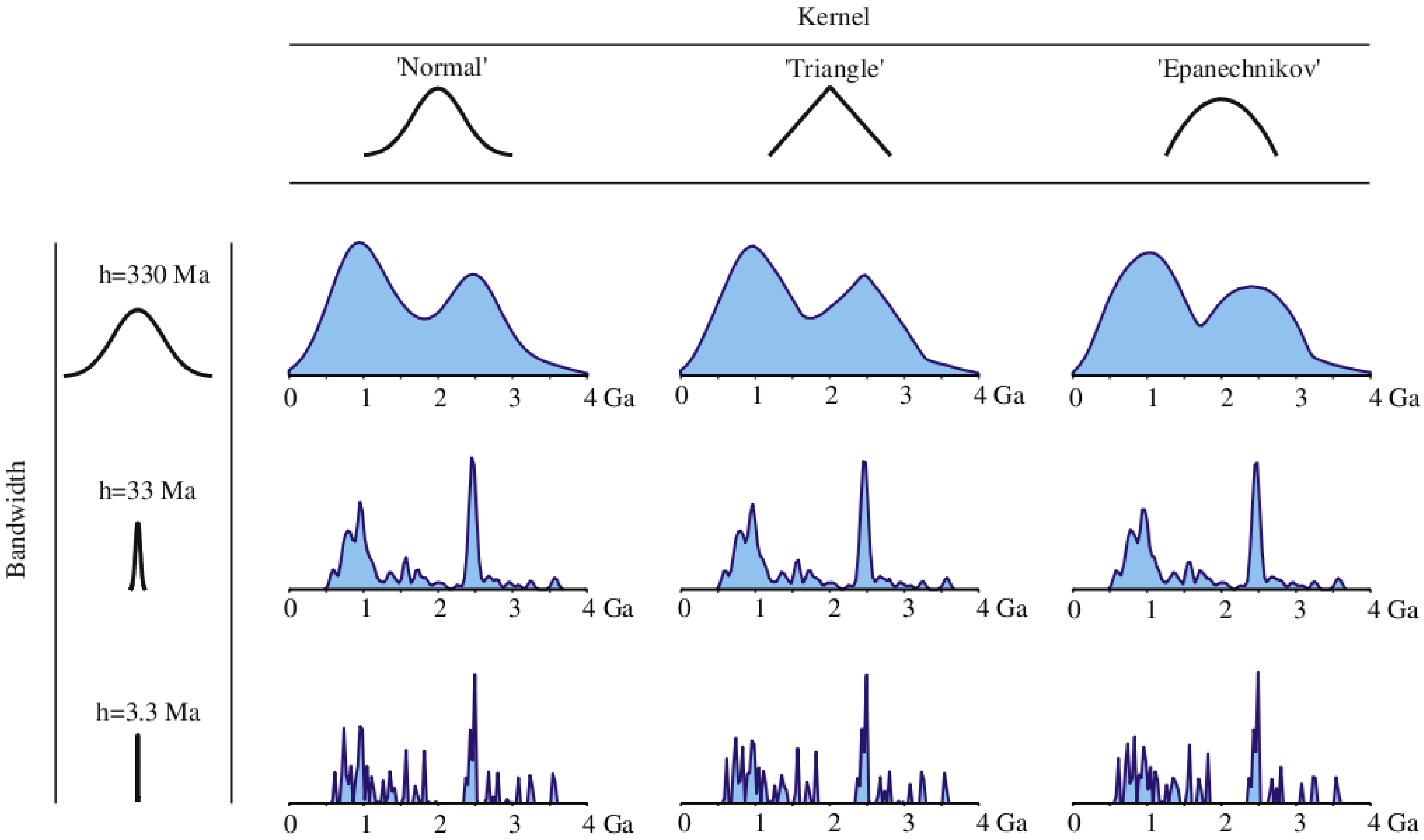


$$P_{\mathrm{KDE}}(x) = \sum_i K\left(x - x_i\right)/N$$

Number of points in the training dataset

Sum over all points in the training dataset

Image: Wikipedia



http://www.ucl.ac.uk/~ucfbpve/papers/VermeeschChemGeol2012/

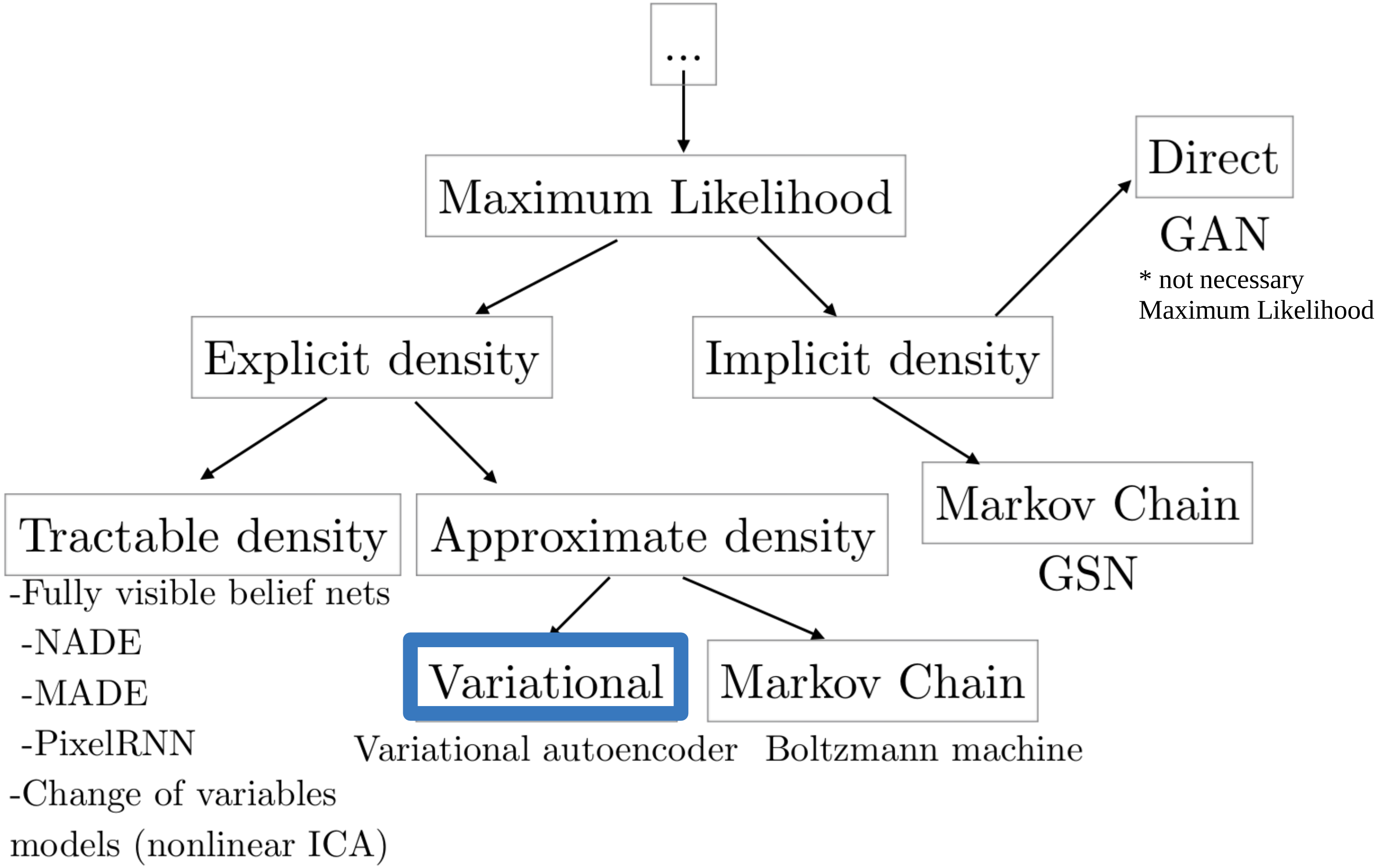
Kernel Density vs Histogram

Kernel Density	Histogram
Smooth PDF	Discrete binned PDF
With number of data points approaching infinity, the value in a point approaches the convolution of the PDF with the kernel function	With number of data points approaching infinity, the value in a bin approaches the unbiased mean PDF in that bin
No easy way to estimate the uncertainty	Straightforward uncertainty estimation of bin values
User-defined parameter: kernel shape and width	User-defined parameter: bins
Requires storing the full training dataset Finite support kernels allow for KDTree optimization	Fast, memory efficient

Kernel Density: summary

- Go-to way for an easy 1-2D PDF approximation
- Applicable in the same cases as histograms
- Has heuristic parameters: kernel shape
- Memory expensive
- Doesn’t scale for high dimensions
- Nice demo: https://mathisonian.github.io/kde/

Generative Model Taxonomy



https://arxiv.org/pdf/1701.00160.pdf

Variational autoencoders (VAE)

$P(x)$ is the distribution of the data (e. g. images)

$P(z)$ is some easily samplable distribution (e. g. Gaussian)

What if we could find some mapping $F(z)$ so that $P\big(F(z)\big) = P(x)$?

VAE

Given

- data points $x_1, \dots, x_n \in \mathbb{R}_\mathrm{m}$
- a distribution $P(z)$
- a parametrized mapping $P(x|z, \theta)$

Find a set of parameters θ to maximize the empirical likelihood:

- $L(\theta) = \prod_i P(x_i|\theta) = \prod_i \int P_\theta(x_i|z)P(z)dz$

VAE

$$L(\theta) = \prod_i \mathbb{E}_z\left[P_\theta(x_i|z)\right]$$

How to optimize?

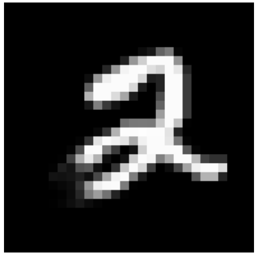
Let x be any training example
Problem: for early stages

$$P_\theta(x|z)$$

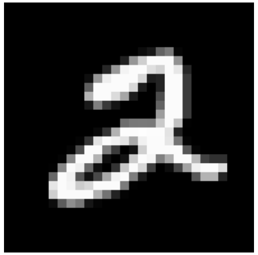
is very sparse, most pixel
combinations are
not like training images

https://arxiv.org/abs/1606.05908

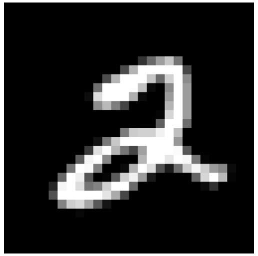
Semantic difference is not like pixel difference. To make them aligned, we need to be strict.



MSE=0.0387



original



MSE=0.2693

VAE: training

Let's add $Q_\theta(z|x)$ – distribution of z values likely produce x

If we were able to sample $Q_\theta(z|x)$, computation of $\mathbb{E}_{z\sim Q} P_\theta(x|z)$ is easy

At the same time, we are interested in the $P_\theta(x)$

Kullback–Leibler divergence

$$D_{KL}(P \parallel Q) = \int \log\left(\frac{P(x)}{Q(x)}\right) P(x)dx$$

Asymmetric

Possibly infinite

Has roots in information theory

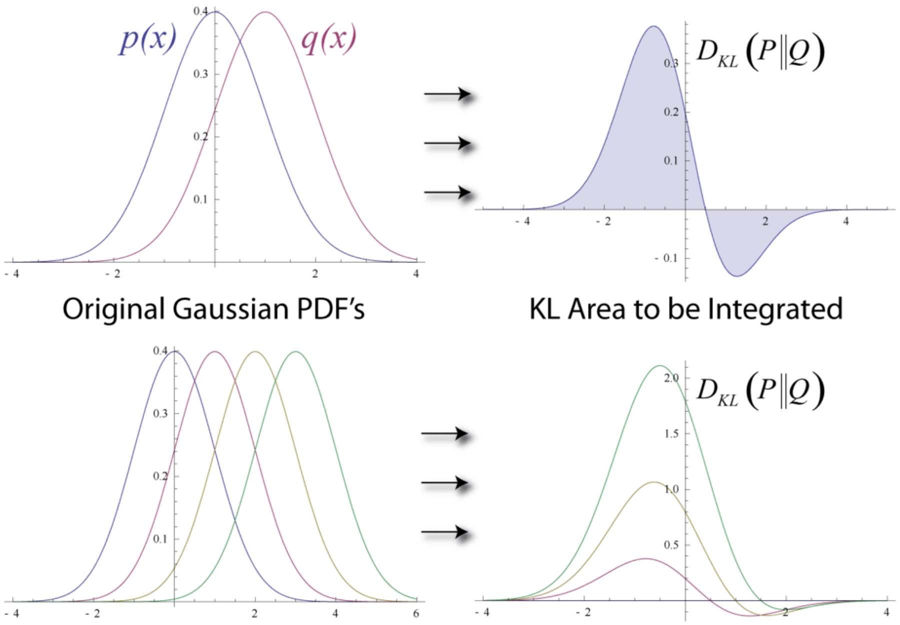


Image: Wikipedia

VAE training: KL divergence

$$D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z|x)] = \mathbb{E}_{z\sim Q}[\log Q_\theta(z|x) - \log P_\theta(z|x)]$$

applying Bayes rule to $P(z|x)$:

$$D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z|x)] = \mathbb{E}_{z\sim Q}[\log Q_\theta(z|x) - \log P_\theta(x|z) - \log P(z)] + \log P_\theta(x)$$

$$\log P_\theta(x) - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z|x)] = \mathbb{E}_{z\sim Q}[\log P_\theta(x|z)] - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z)]$$

We want to maximize the likelihood of the model

Minimize the error Q approximation

This can be optimized via SGD!

VAE: Gaussian parametrization

Let's use Gaussian Q:

$$Q_\theta(z|x) = N\big(z|\mu_\theta(x), \sigma_\theta(x)\big)$$

This will be an NN

This too

VAE: Gaussian parametrization

Let's use Gaussian Q:

$$Q_\theta(z|x) = N\big(z|\mu_\theta(x), \sigma_\theta(x)\big)$$

This will be an NN

This too

And Gaussian P(z):

$$P(z) = N(z|0,1)$$

This way D is analytically computable:

$$-D_{\mathrm{KL}}\big(Q_\theta(z|x) \parallel P_\theta(z)\big) = \frac{1}{2}\sum\big(1 + \log\big(\sigma_\theta^2(x)\big) - \mu_\theta^2(x) - \sigma_\theta^2(x)\big)$$

VAE training

$$\log P_\theta(x) - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z|x)] = \mathbb{E}_{z\sim Q}[\log P_\theta(x|z)] - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z)]$$

$$-D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z)] = \frac{1}{2}\sum\left(1 + \log\left(\sigma_\theta^2(x)\right) - \mu_\theta^2(x) - \sigma_\theta^2(x)\right)$$

$$\mathbb{E}_{z\sim Q}[\log P_\theta(x|z)] = ?$$

VAE training

$$\log P_\theta(x) - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z|x)] = \mathbb{E}_{z\sim Q}[\log P_\theta(x|z)] - D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z)]$$

$$-D_{\mathrm{KL}}[Q_\theta(z|x) \parallel P_\theta(z)] = \frac{1}{2}\sum\left(1 + \log\left(\sigma_\theta^2(x)\right) - \mu_\theta^2(x) - \sigma_\theta^2(x)\right)$$

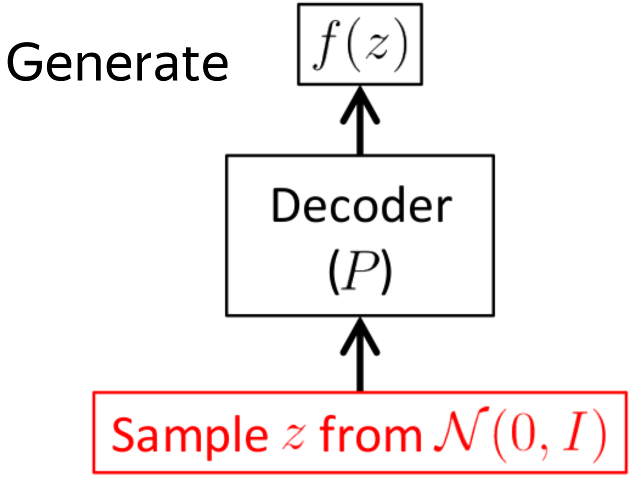
Gaussian?

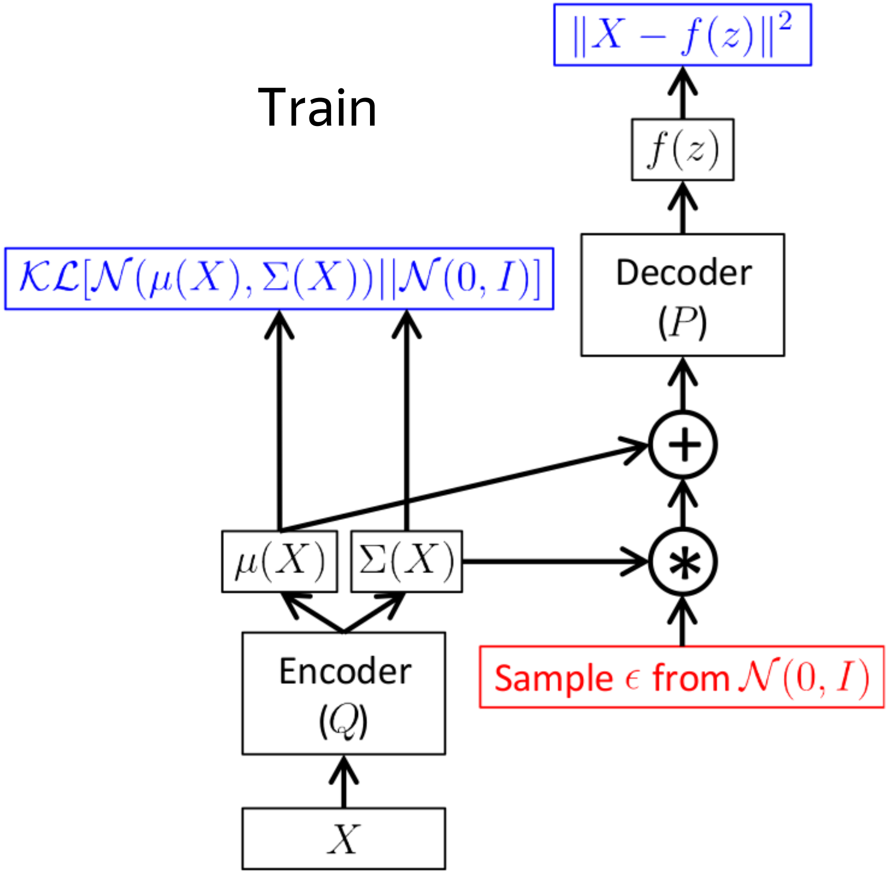
$$\mathbb{E}_{z\sim Q}[\log P_\theta(x|z)] = \mathbb{E}_{z\sim Q}[\log N(x|f_\theta(z), h^2)] = C - \frac{1}{2}\|x - f_\theta(z)\|^2/h^2$$

Constant, doesn't depend on f_θ

Parameter

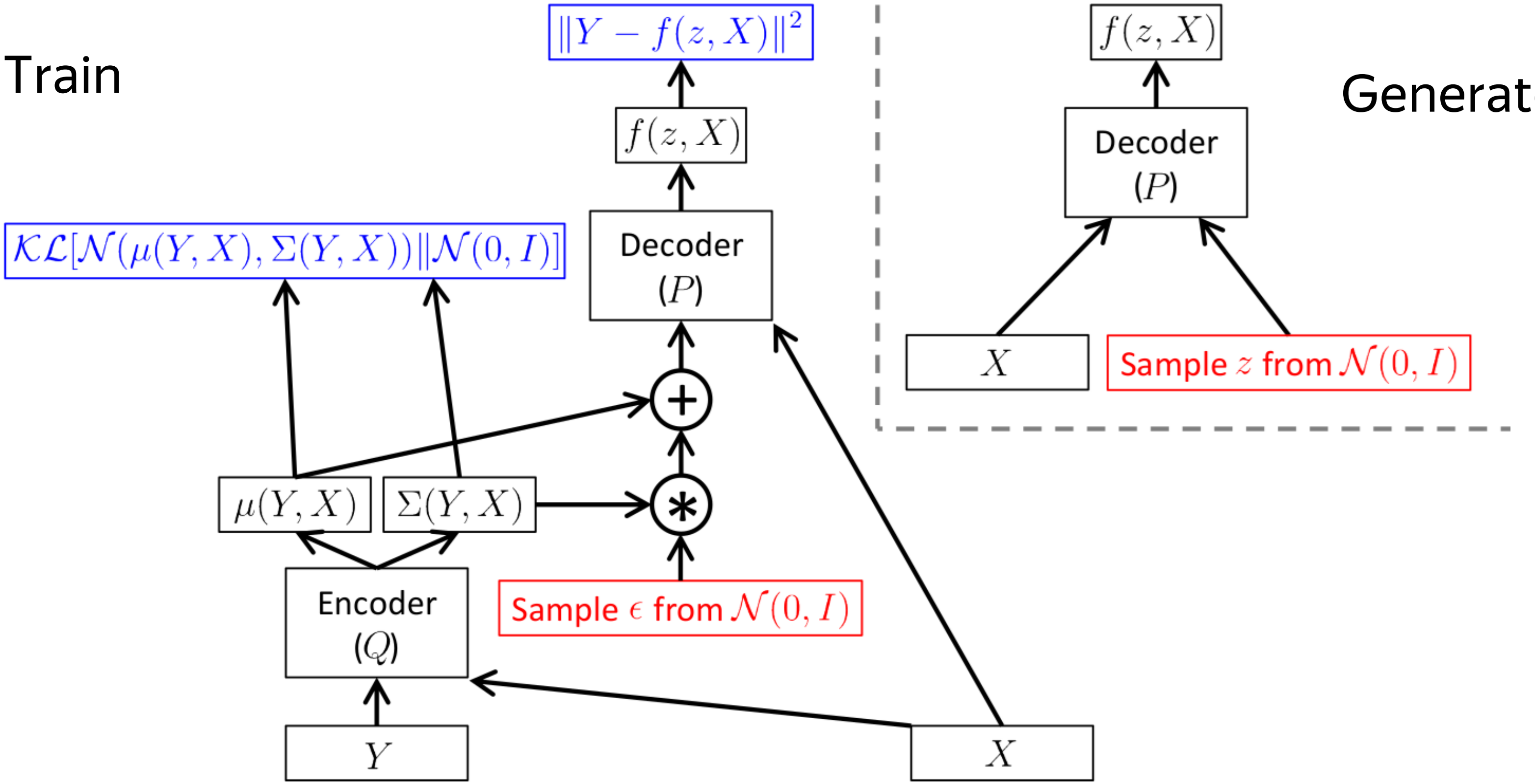
VAE as feedforward NN





https://arxiv.org/abs/1606.05908

Conditional VAE



https://arxiv.org/abs/1606.05908

VAE: summary

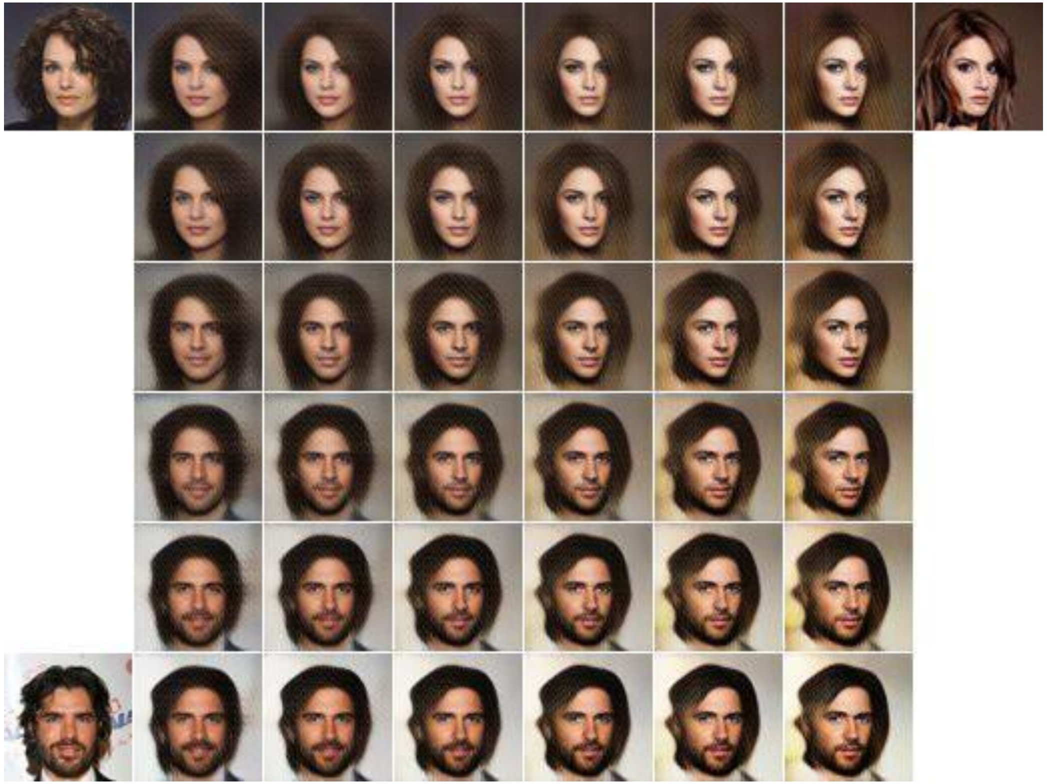
General-purpose generative model

Lots of image generation fun

Latent space fun

The Gaussian approximations may be to rigid

Subjectively worse sample image quality than GAN



https://hackernoon.com/latent-space-visualization-deep-learning-bits-2-bd09a46920df

Feel free to drop a line

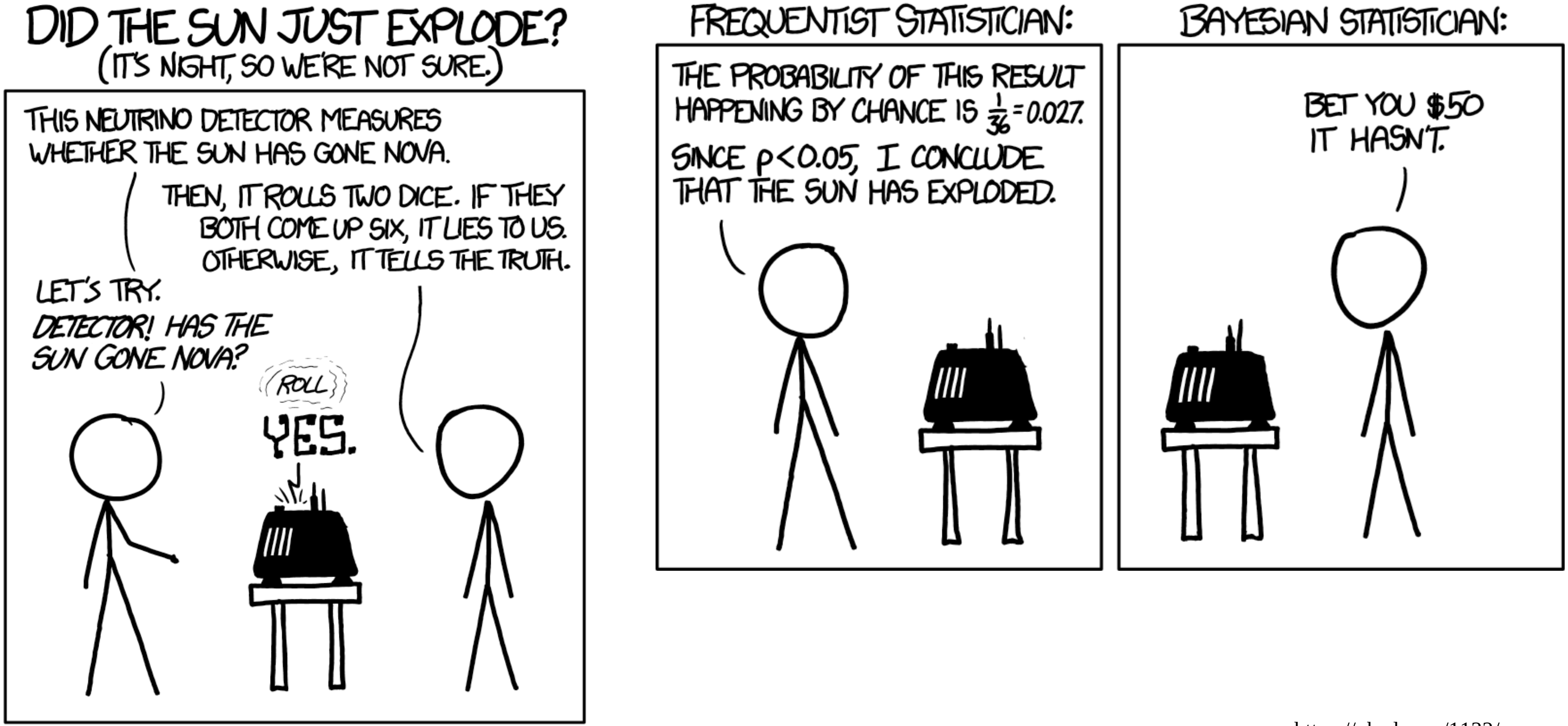
Nikita Kazeev
HSE, Rome Sapienza, YSDA, trace amounts of Yandex proper

kazeevn@yandex-team.ru

telegram.me/kazeevn

Backup

What is probability?



https://xkcd.com/1132/