

Генеративные модели в машинальном обучении

Лекция 4

Вариационные автокодировщики. Нормализационные потоки.

Михаил Гущин

mhushchyn@hse.ru

НИУ ВШЭ, 2024



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

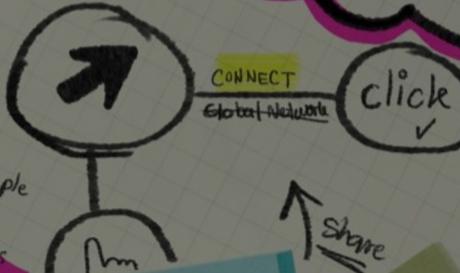
Приложения

CHECK LIST:

- chat

share

SOCIAL MEDIA



people

Friends

click ✓

Share

cloud

↑↓

good!

hello!

...

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

“ ”

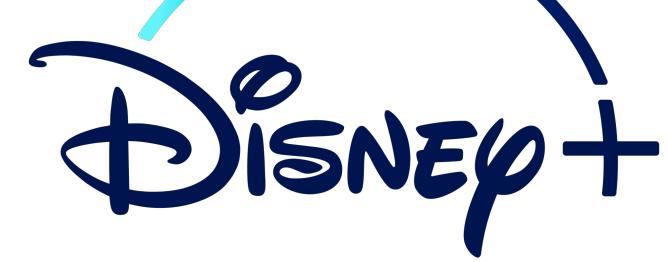
“ ”

“ ”

“ ”

“ ”

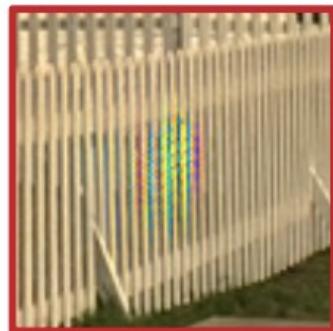
Пример



Пример

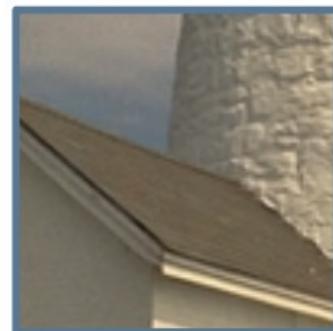


Autoencoder based
(Minnen et al. 2018)



$N = 1$

$N = 15$



Ours



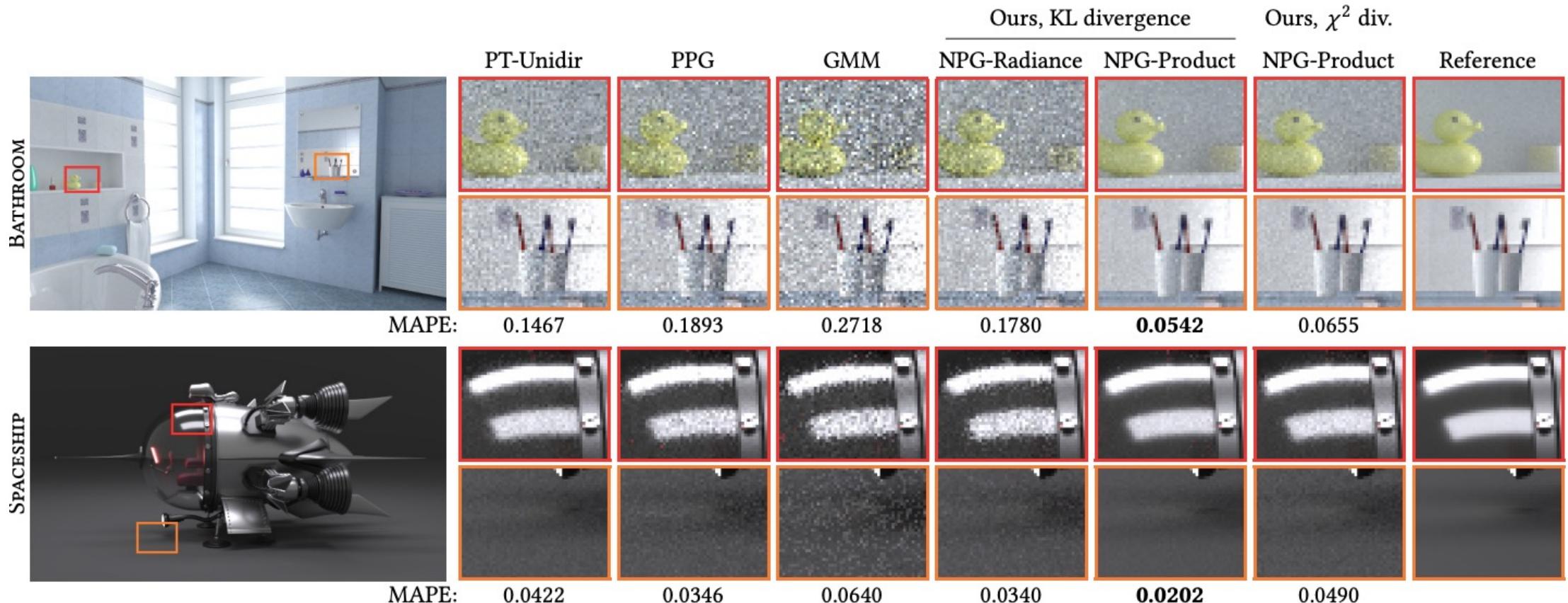
$N = 1$

$N = 15$

Results after N compression/decompression operations

Lossy Image Compression with Normalizing Flows, ICLR 2021

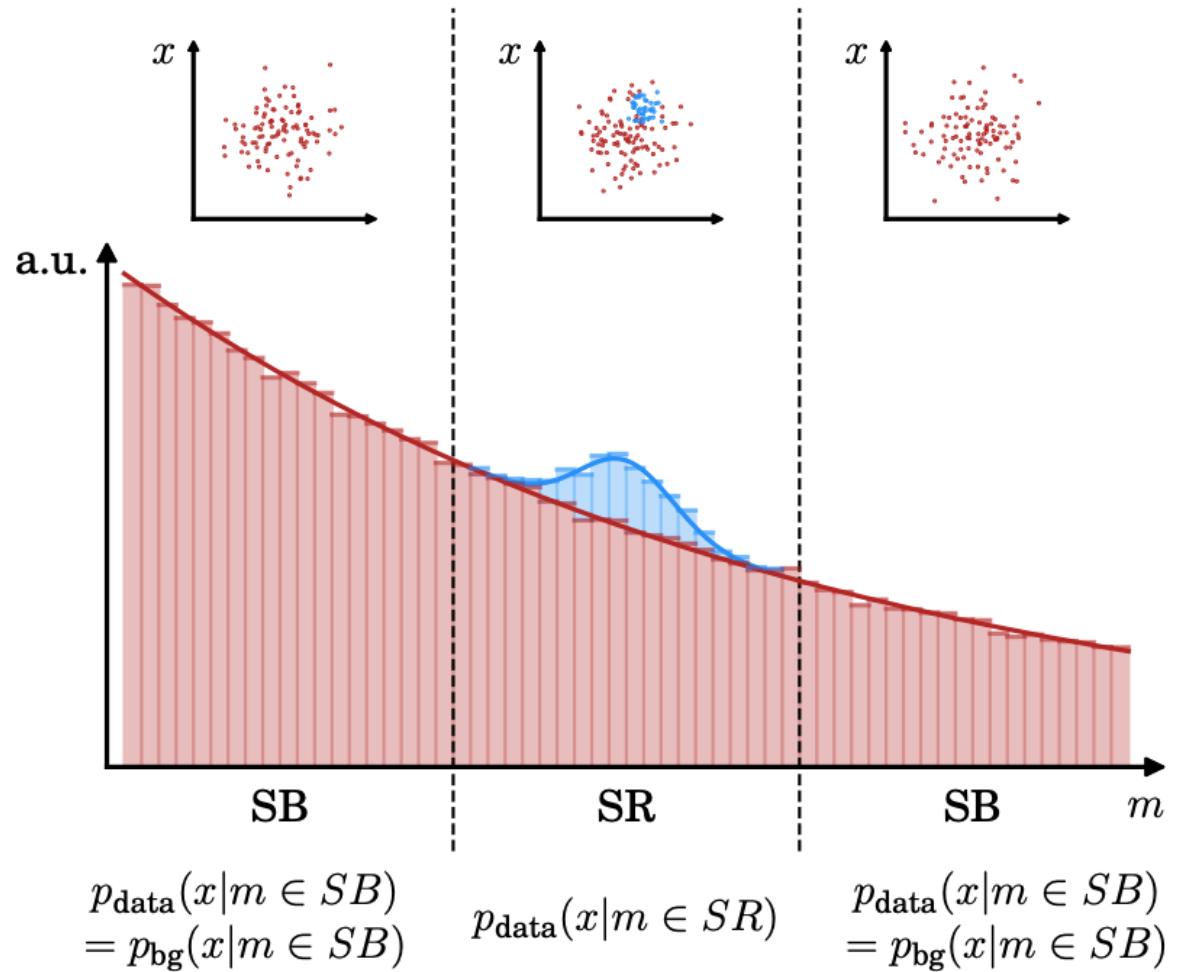
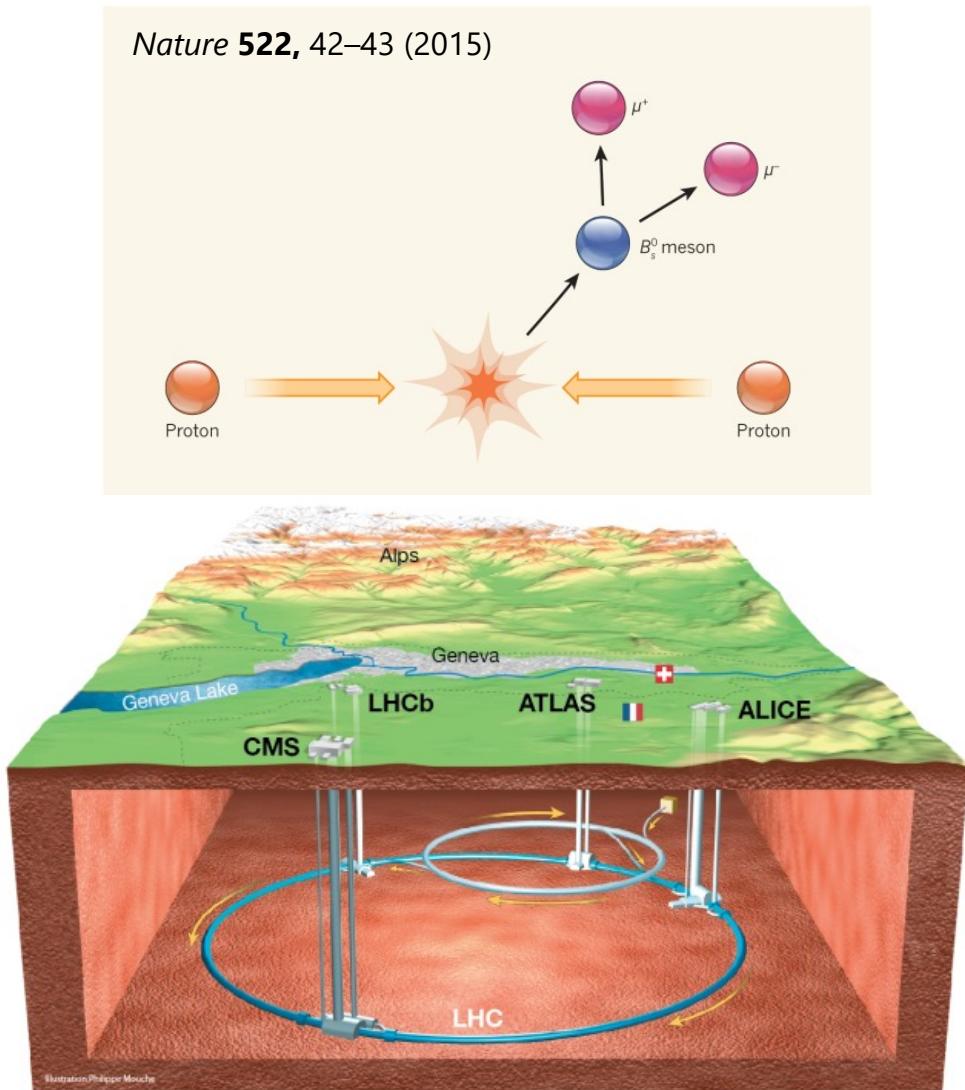
Пример



Neural Importance Sampling

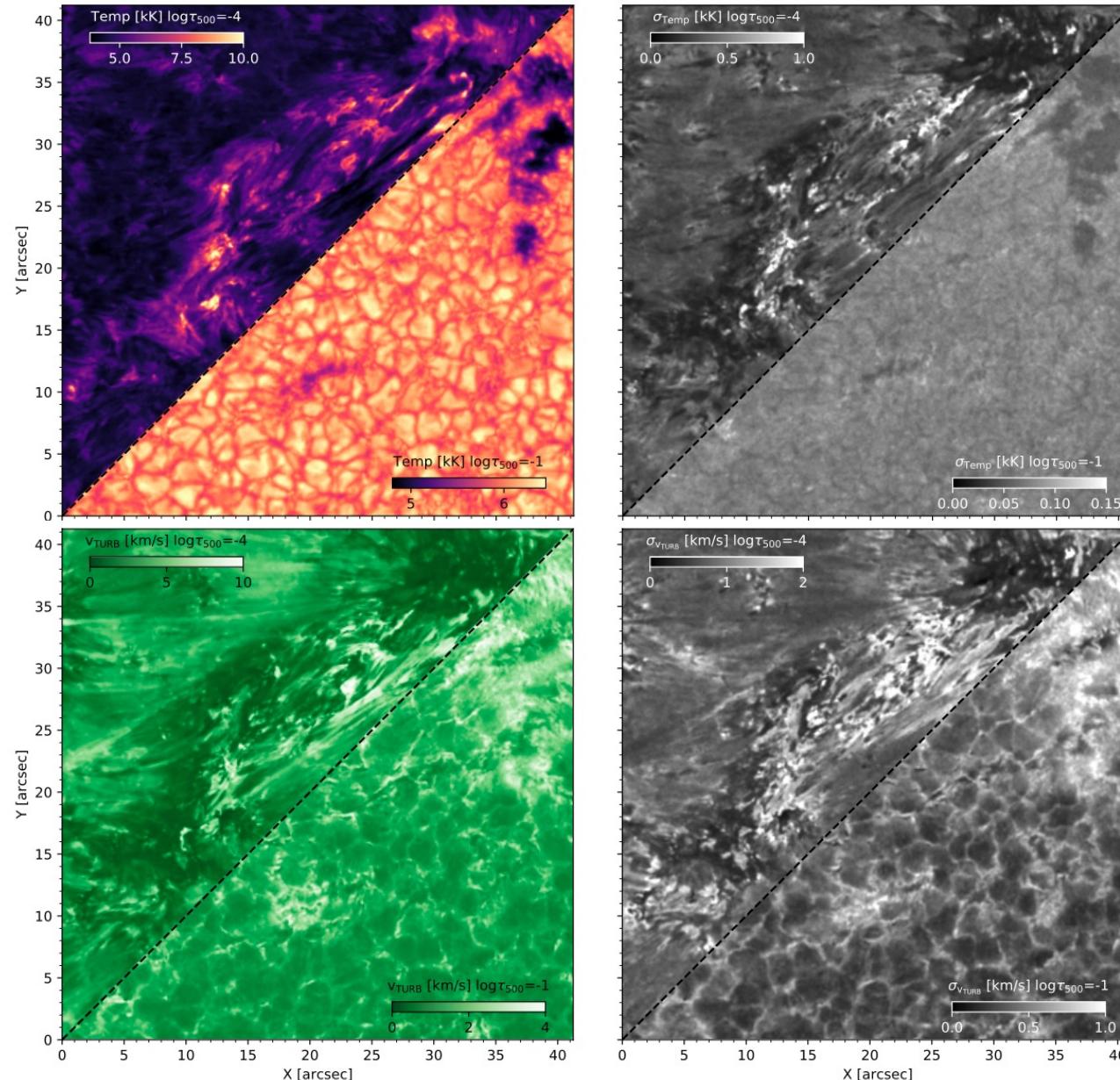
Поиск новой физики

Nature 522, 42–43 (2015)



Реконструкция солнечной атмосферы

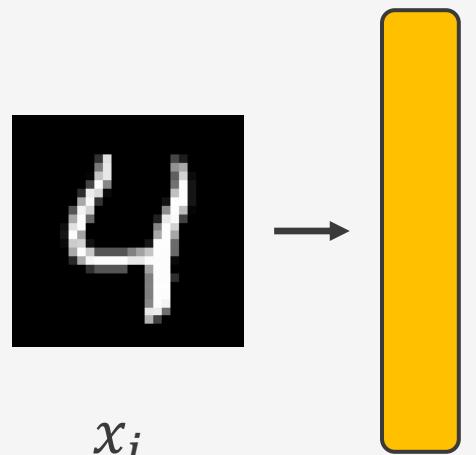
Bayesian Stokes inversion
with normalizing flows,
A&A 659, A165 (2022)



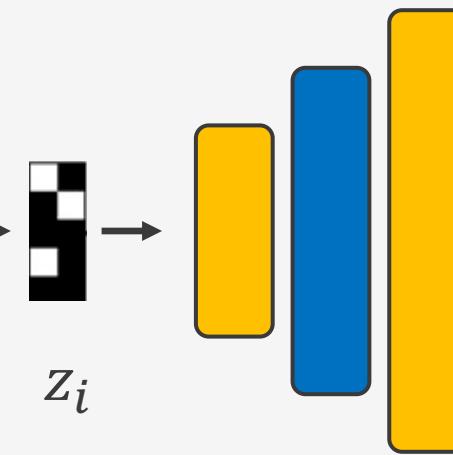
Автокодировщики

Автокодировщик

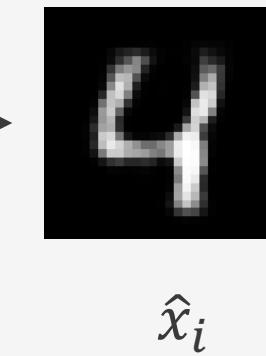
Входное
изображение



Скрытое
представление



Восстановленно
е изображение



\hat{x}_i

Кодировщик

Декодировщик

Функция потерь

MSE функция потерь:

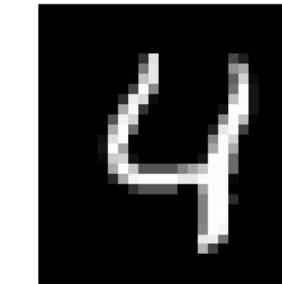
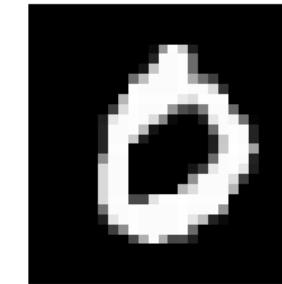
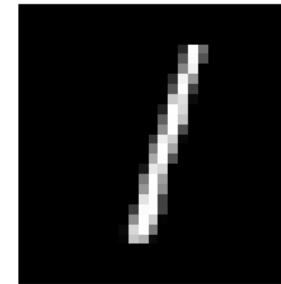
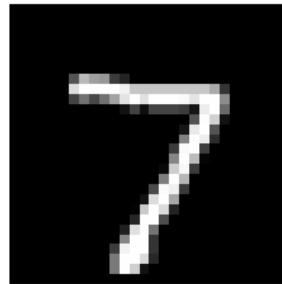
$$L = \frac{1}{n} \sum_{i=1}^n \sum_k (\hat{x}_{ik} - x_{ik})^2$$

x_{ik} - k -ый пиксель i -го входного изображения;

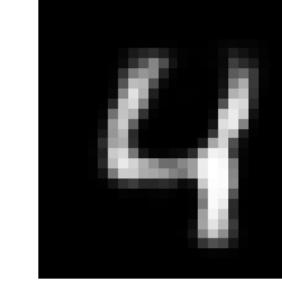
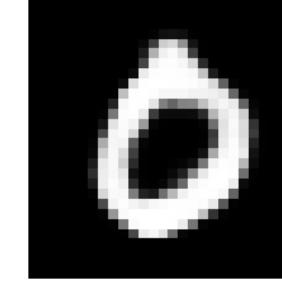
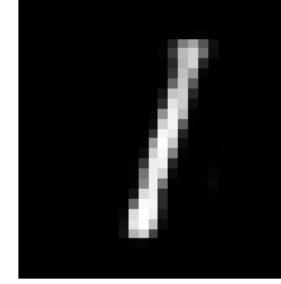
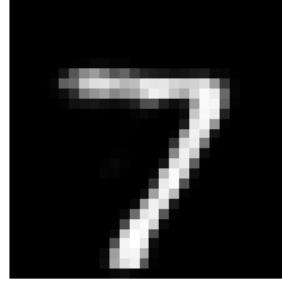
\hat{x}_{ik} - k -ый пиксель i -го восстановленного изображения

Пример

Первоначальные изображения



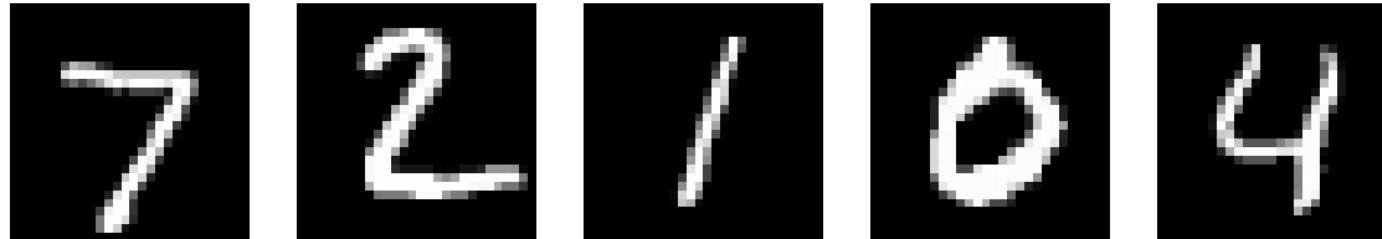
Восстановленные изображения



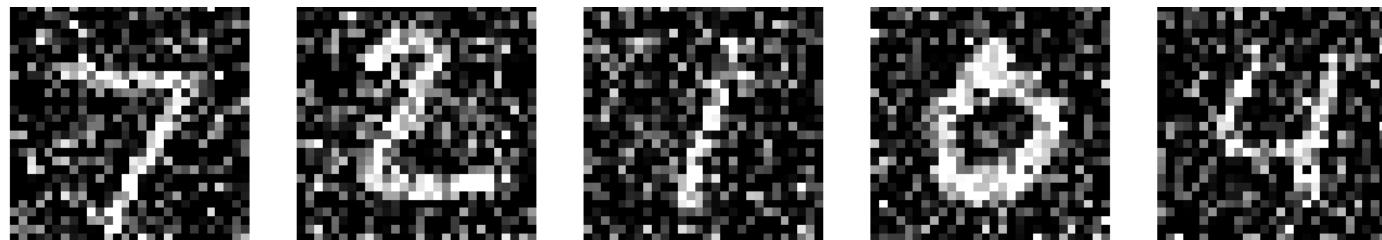
<https://www.machinelearningmastery.ru/applied-deep-learning-part-3-autoencoders-1c083af4d798/>

Пример: подавление шума

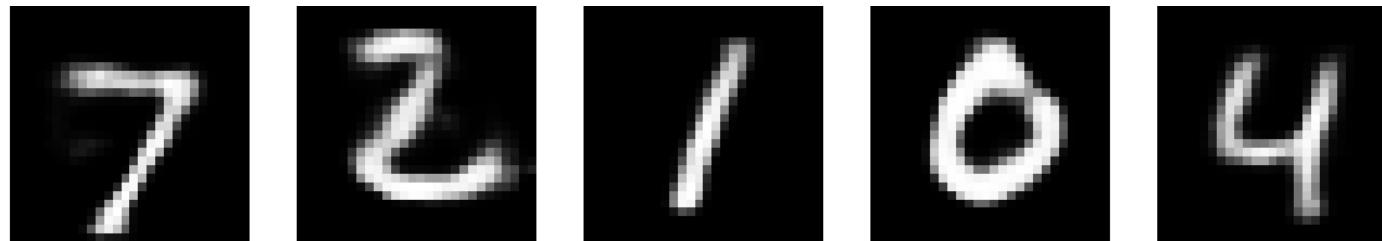
Первоначальные изображения



Зашумление



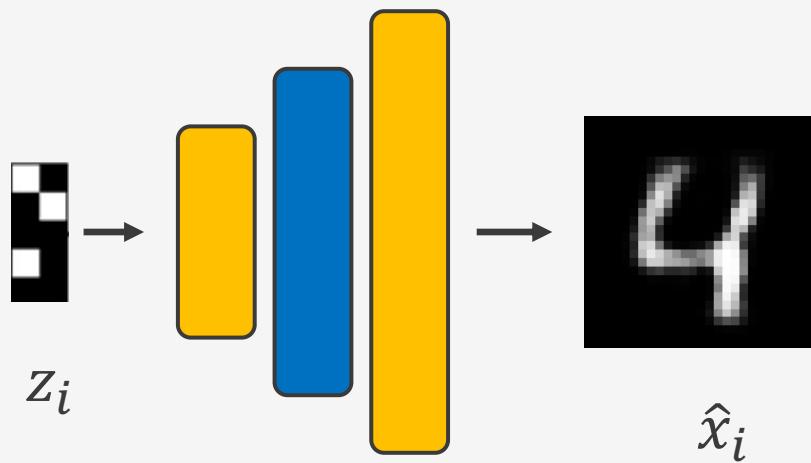
Выход автокодировщика



Генерация изображений

Генерация
случайного z_i

Восстановленно
е изображение

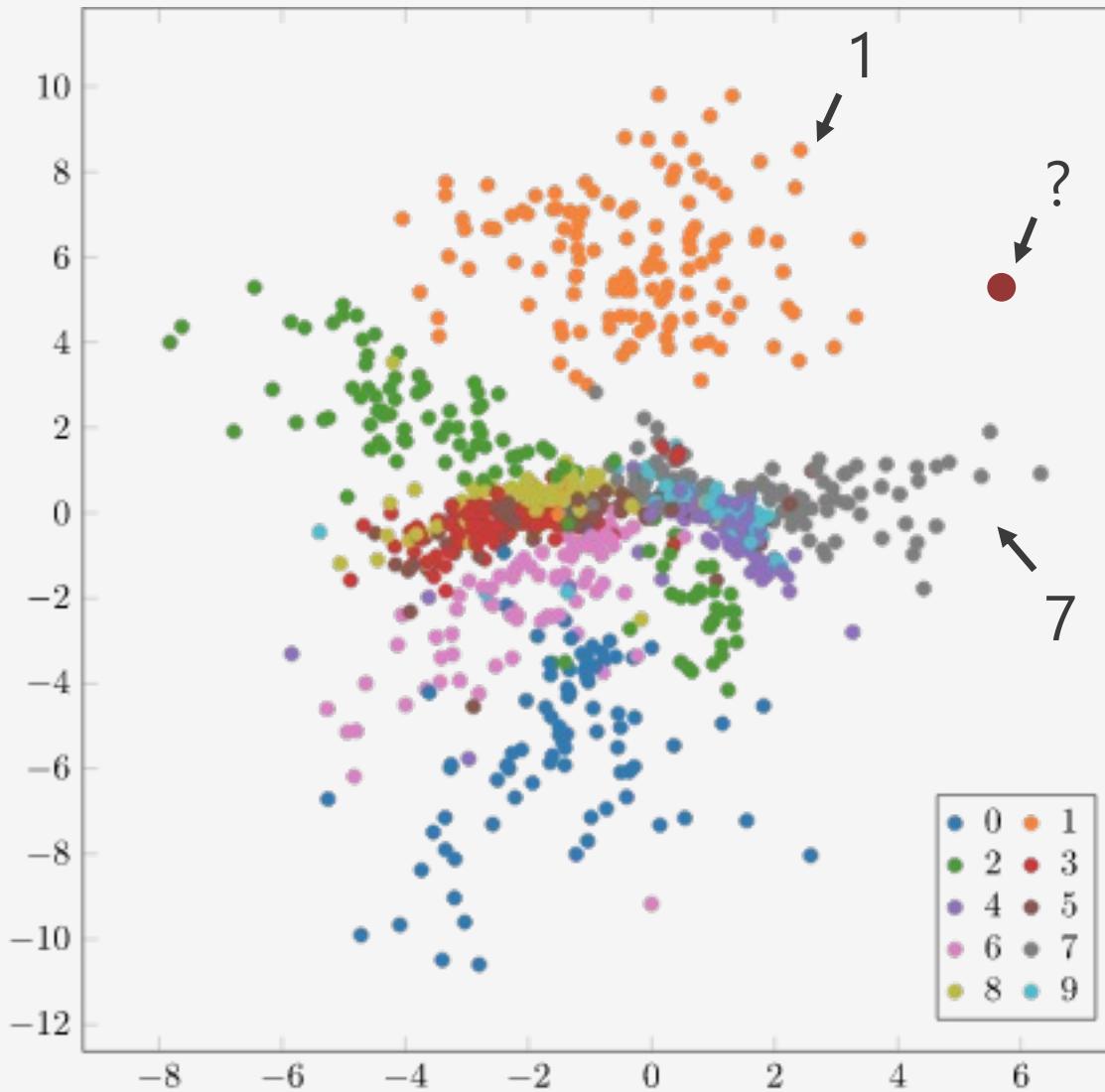


Декодировщик

Пример

3	9	2	1	3	9	0	2
5	0	3	6	9	9	0	3
4	0	7	8	4	5	3	9
9	8	1	5	9	1	6	1
0	1	8	3	5	4	7	8
2	0	8	7	9	9	8	3
1	9	3	6	5	6	0	8
3	3	0	0	9	7	4	0

Скрытое пространство



Выводы

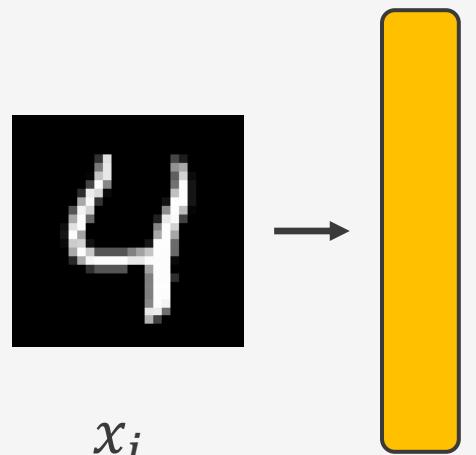
- Автокодировщики хорошо сжимают изображения
- Хорошо подавляют шум
- Плохо генерируют новые изображения из-за пропусков в скрытом пространстве

Вариационные автокодировщики

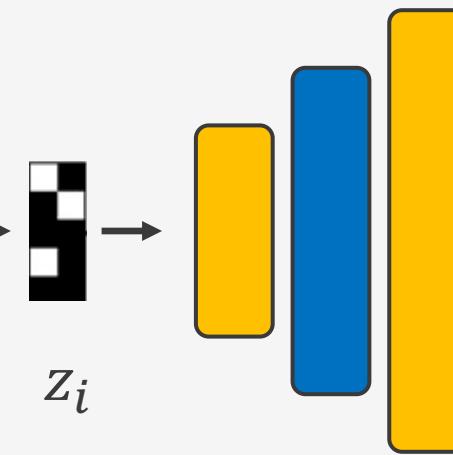


Автокодировщик

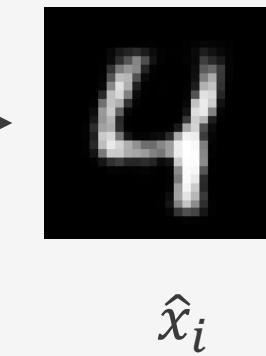
Входное
изображение



Скрытое
представление



Восстановленно
е изображение

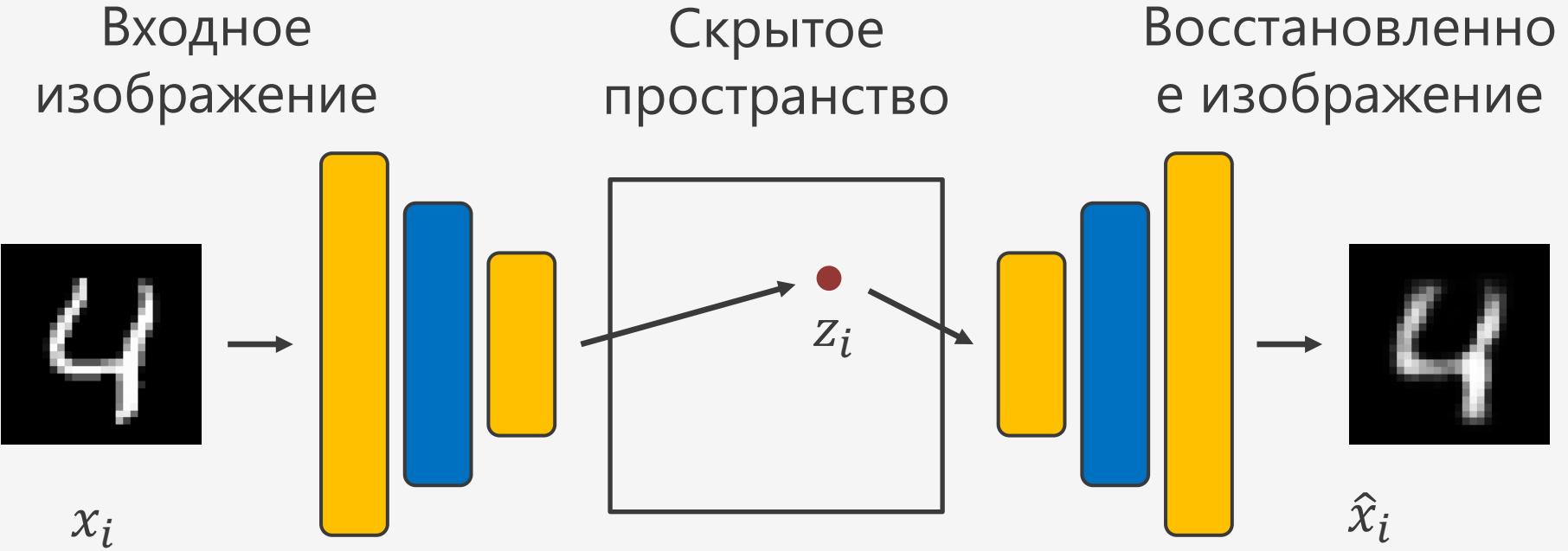


\hat{x}_i

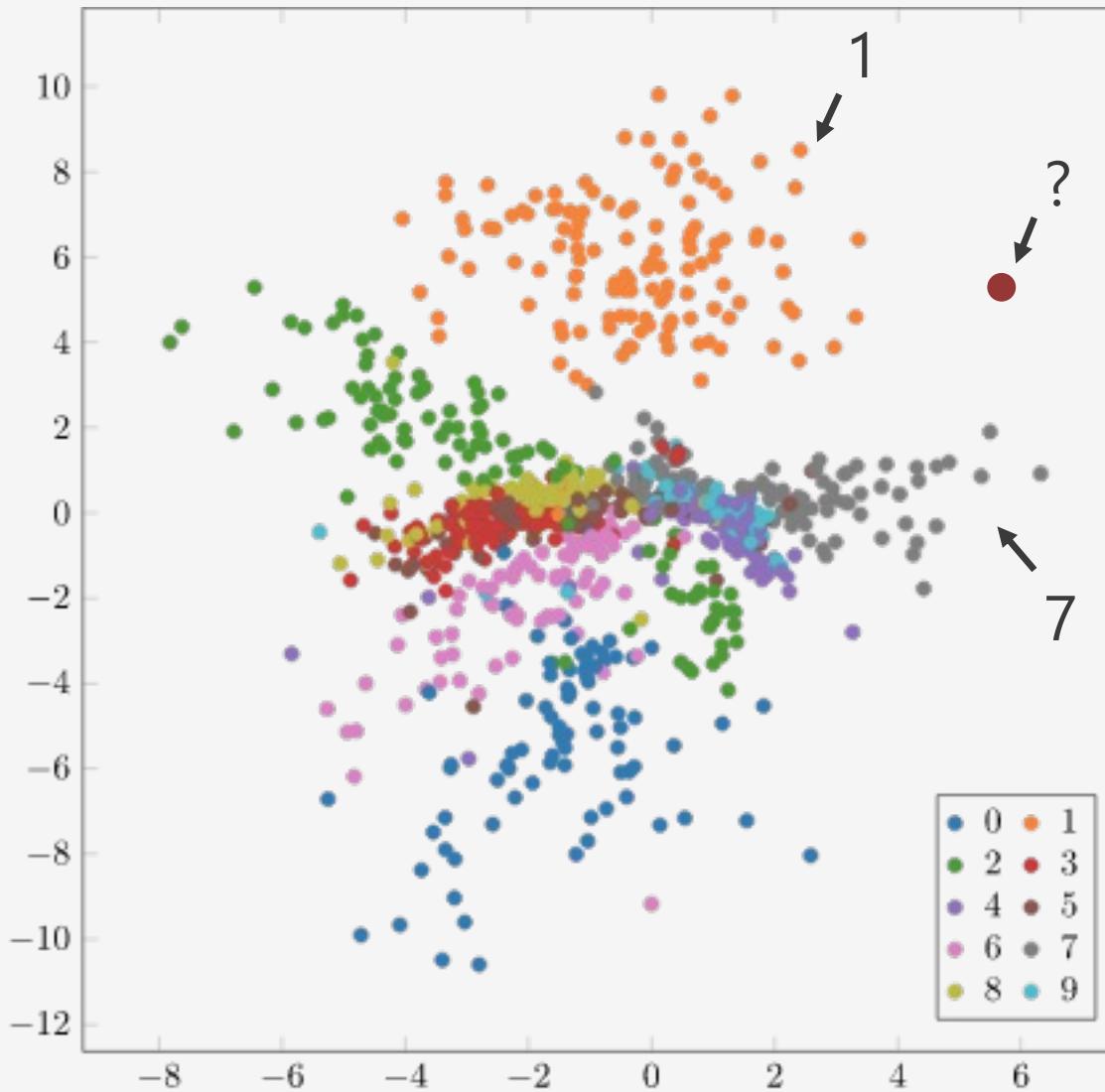
Кодировщик

Декодировщик

Автокодировщик



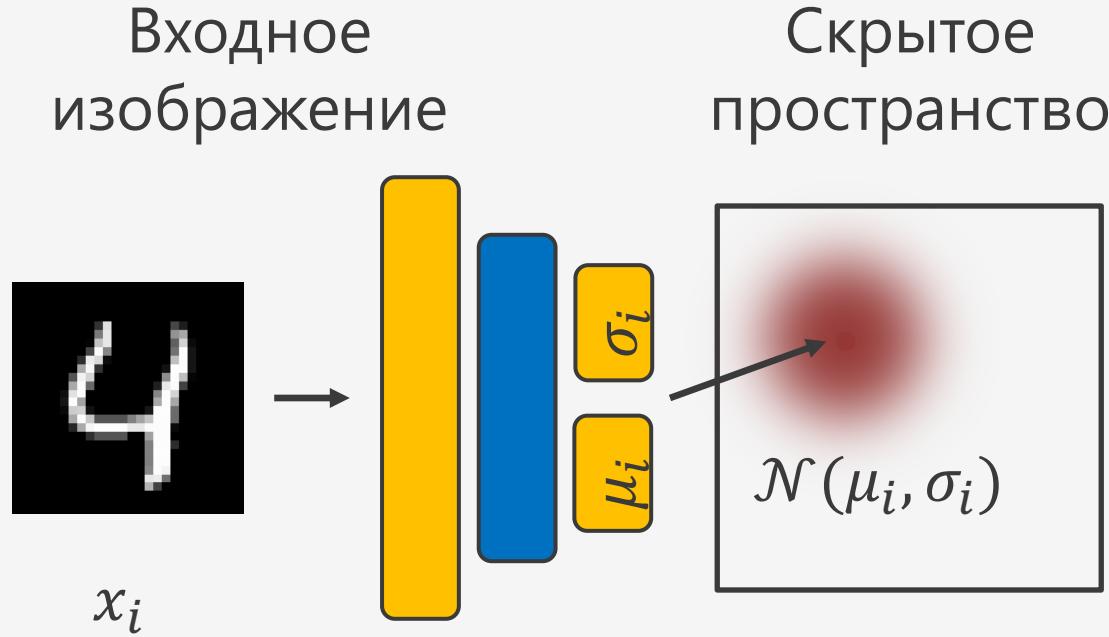
Скрытое пространство



Скрытое пространство

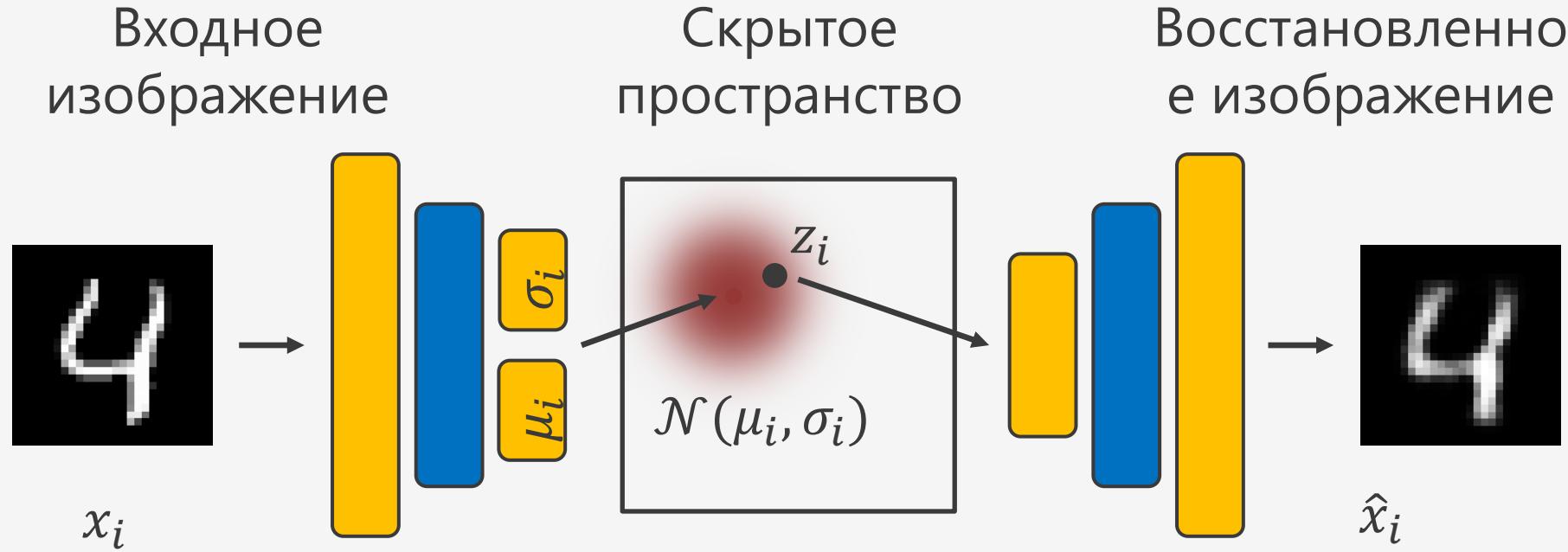
- Автокодировщик переводит изображение в **одну** точку в скрытом пространстве
- Будем кодировать изображение в целое **распределение** в скрытом пространстве

Вариационный автокодировщик



Кодируем изображение нормальным распределением $\mathcal{N}(\mu_i, \sigma_i)$

Вариационный автокодировщик



Для любого $z_i \sim \mathcal{N}(\mu_i, \sigma_i)$ хотим восстановить исходное изображение

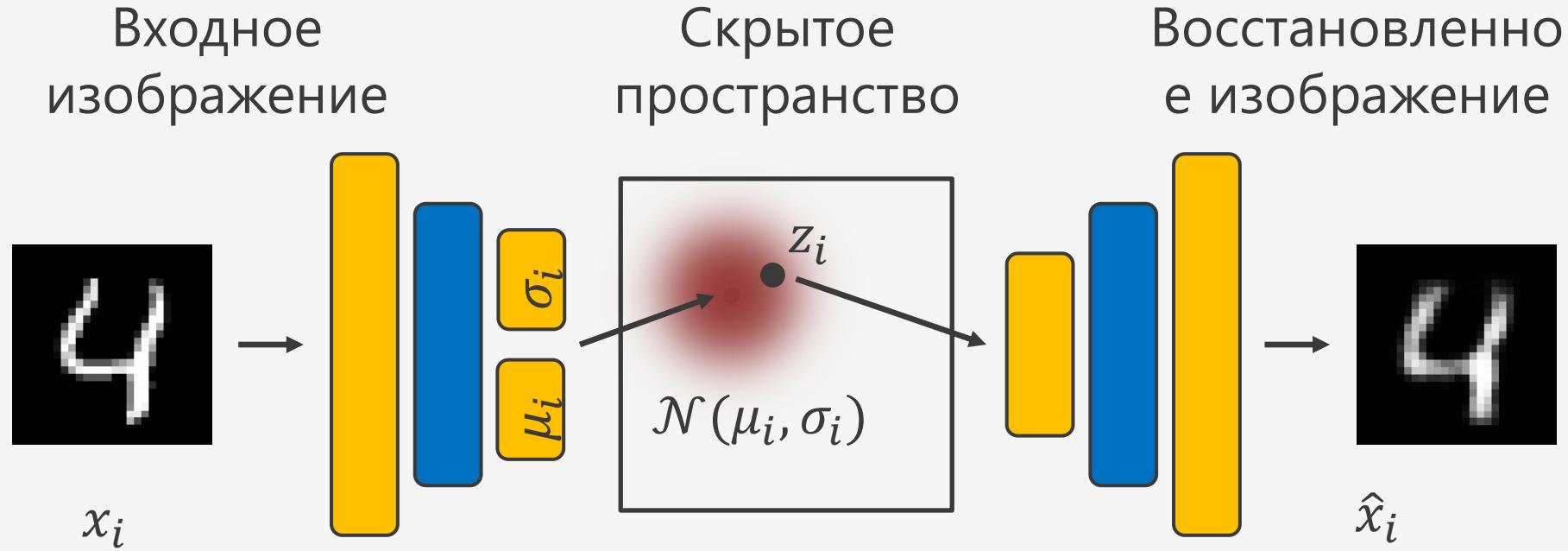
Выводы

- Вариационный автокодировщик переводит изображение в нормальное распределение в скрытом пространстве
- Каждый вектор из этого распределения хотим восстановить в исходное изображение
- Как обучать вариационный кодировщик?

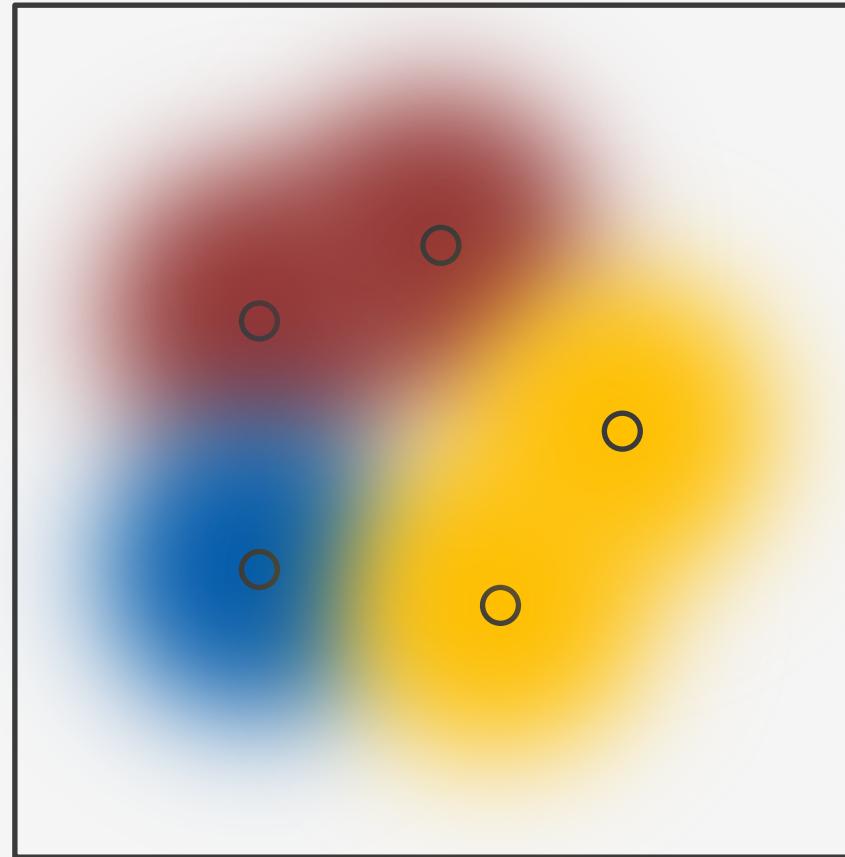
Обучение вариационных автокодировщиков



Вариационный автокодировщик



Скрытое пространство



Хотим полностью заполнить
пространство без пробелов

Функция потерь

$$L = \frac{1}{n} \sum_{i=1}^n \alpha(\hat{x}_i - x_i)^2 + \text{KL}(\mathcal{N}(\mu_i, \sigma_i) \parallel \mathcal{N}(0, 1)) \rightarrow \min$$



Качество
реконструкции
изображения



«Регуляризатор» на
размер и положение
облачка

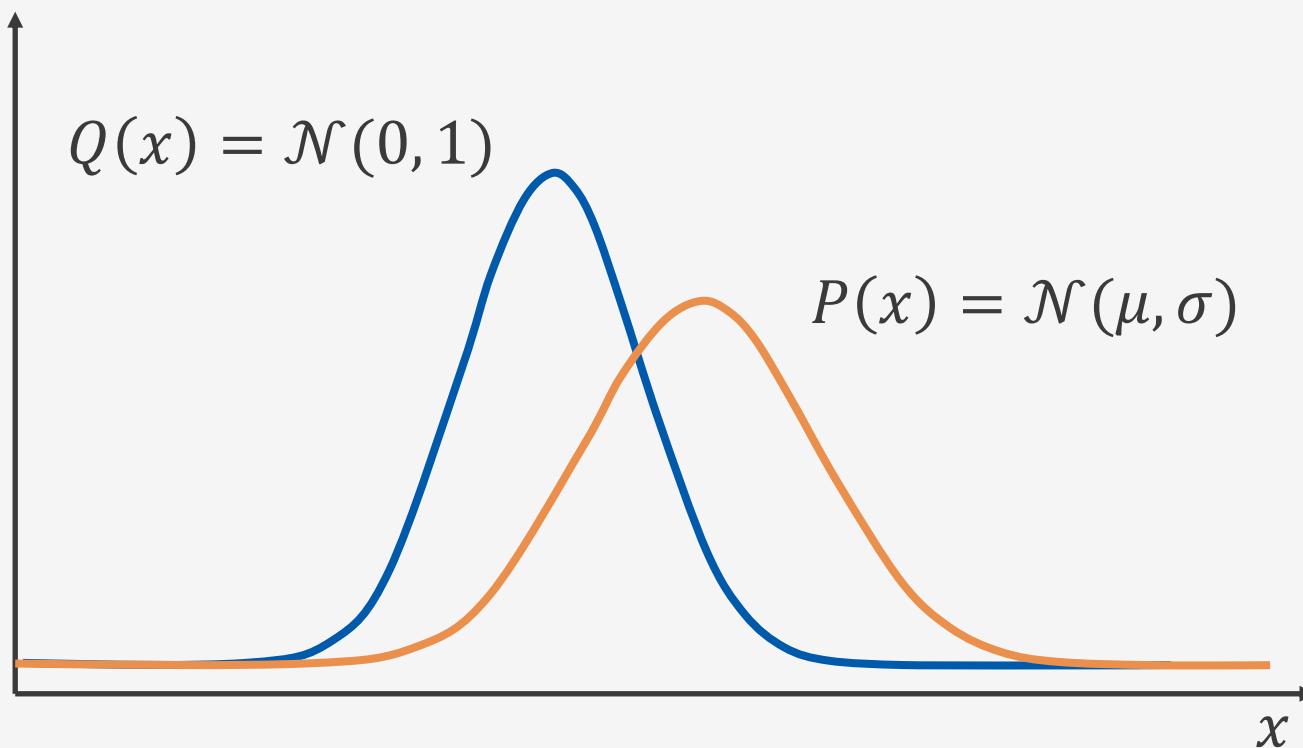
Функция потерь

- Дивергенция Кульбака-Лейблера:

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

- Является мерой расхождения двух распределений P и Q

Функция потерь



С ростом μ, σ распределения отличаются сильнее,
а $KL(P \parallel Q)$ увеличивается

Функция потерь

$$KL(\mathcal{N}(\mu_i, \sigma_i) \parallel \mathcal{N}(0, 1)) = \frac{1}{2} (\sigma_i^2 + \mu_i^2 - 1 - \ln \sigma_i^2)$$

Функция потерь

$$L = \frac{1}{n} \sum_{i=1}^n \alpha (\hat{x}_i - x_i)^2 + \frac{1}{2} (\sigma_i^2 + \mu_i^2 - 1 - \ln \sigma_i^2) \rightarrow \min$$

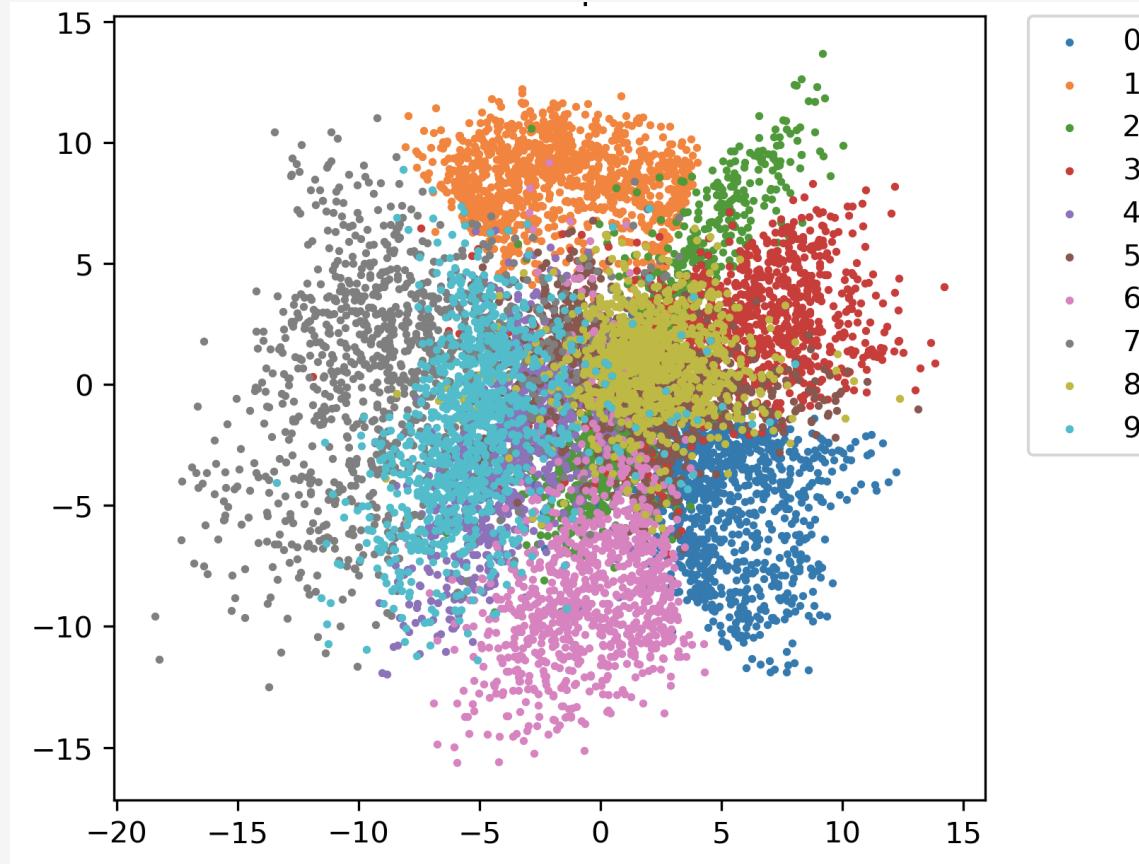


Качество
реконструкции
изображения



«Регуляризатор» на
размер и положение
облачка

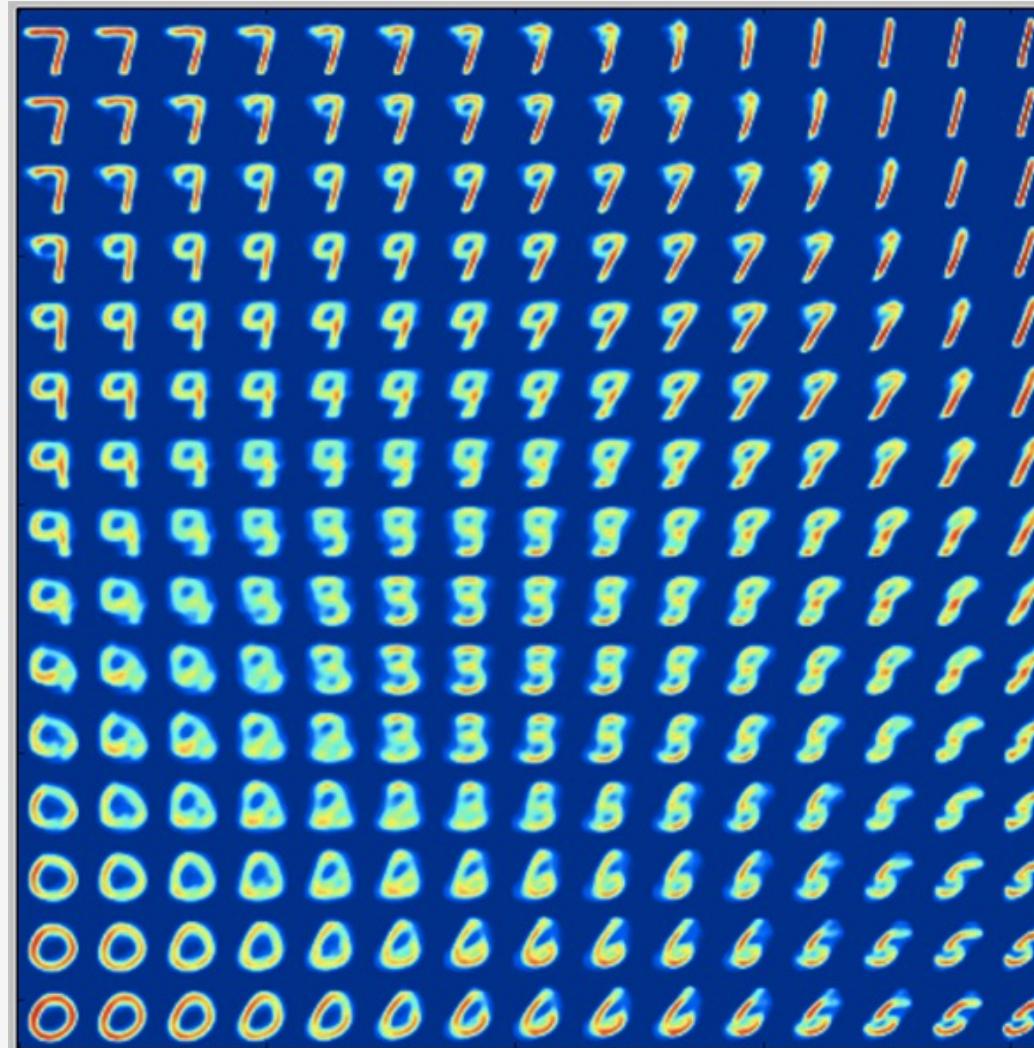
Скрытое пространство



<https://github.com/greentfrapp/keras-aae>

Пространство заполняется без пропусков

Интерполяция в скрытом пространстве



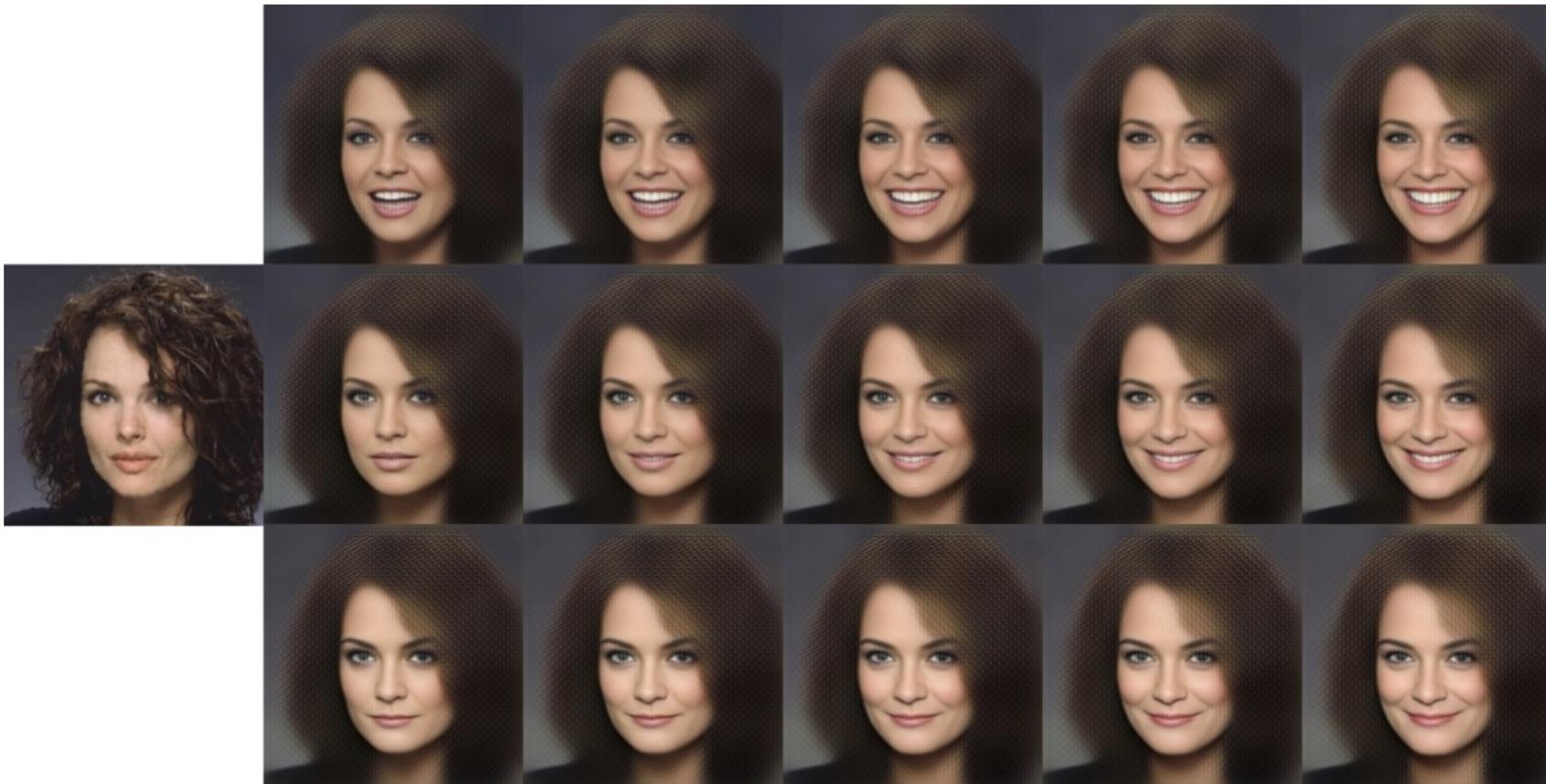
<https://blog.keras.io/building-autoencoders-in-keras.html>

Пример



<https://neurohive.io/ru/osnovy-data-science/variacionnyj-avtojenkoder-vae/>

Пример



<https://arxiv.org/pdf/1609.04468.pdf>

Интерполяция в скрытом пространстве для изменения
улыбки на фотографии

Выводы

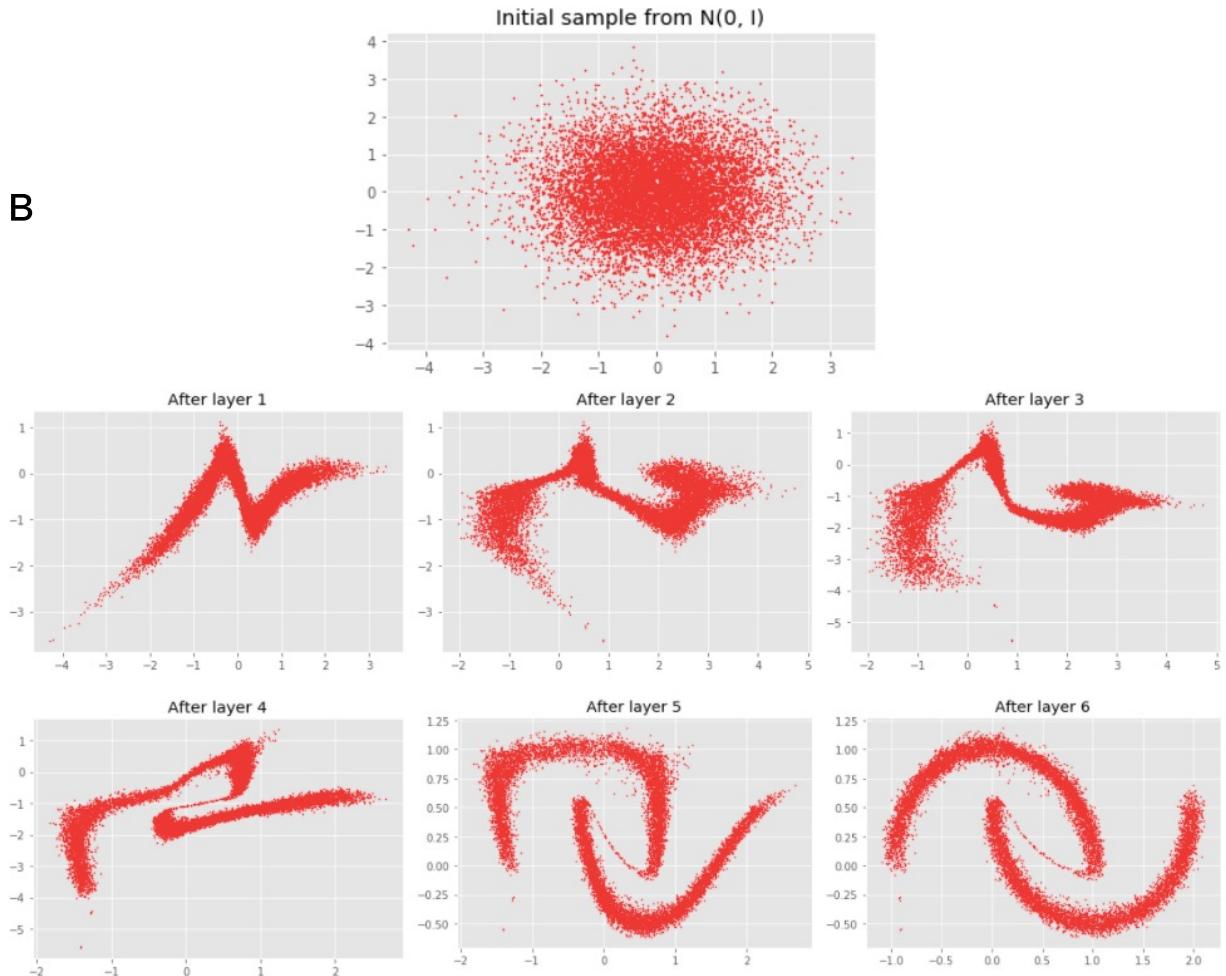
- Вариационный автокодировщик переводит изображение в нормальное распределение в скрытом пространстве
- Нет пропусков в скрытом пространстве
- Хорошо генерирует новые изображения
- Хорошо интерполирует изображения
- Замыливание изображений

The background image shows a wide, arid valley or canyon floor. The terrain is covered in light-colored, scrubby vegetation. Several distinct, reddish-brown streaks of soil or sediment run down the slopes of the surrounding hills, indicating erosion patterns. A narrow, more densely vegetated area of green trees and bushes is visible in the center of the valley.

Нормализационные потоки

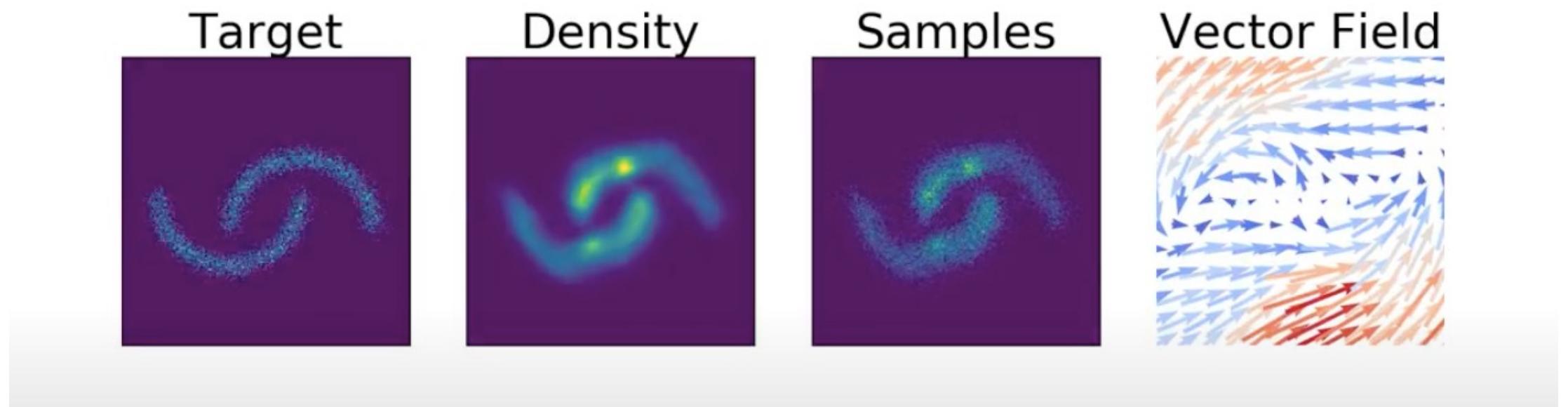
Интуиция

Нормализационный поток –
последовательность обратимых
преобразований одного распределения в
другое



<https://engineering.paperclip.com/posts/normalizing-flows-part-2/>

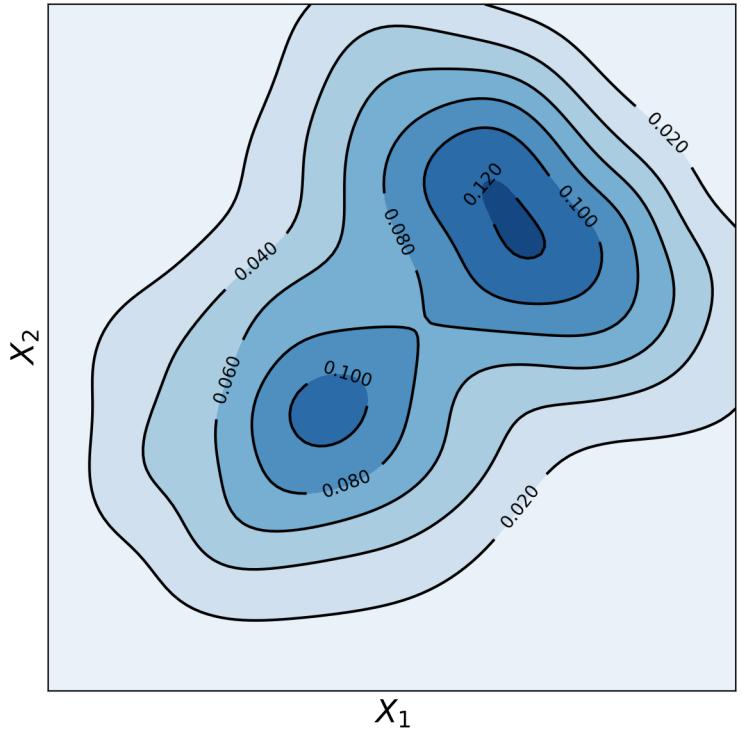
Интуиция



<https://youtu.be/QQpz7V3OgFc>

Теорема о замене переменных

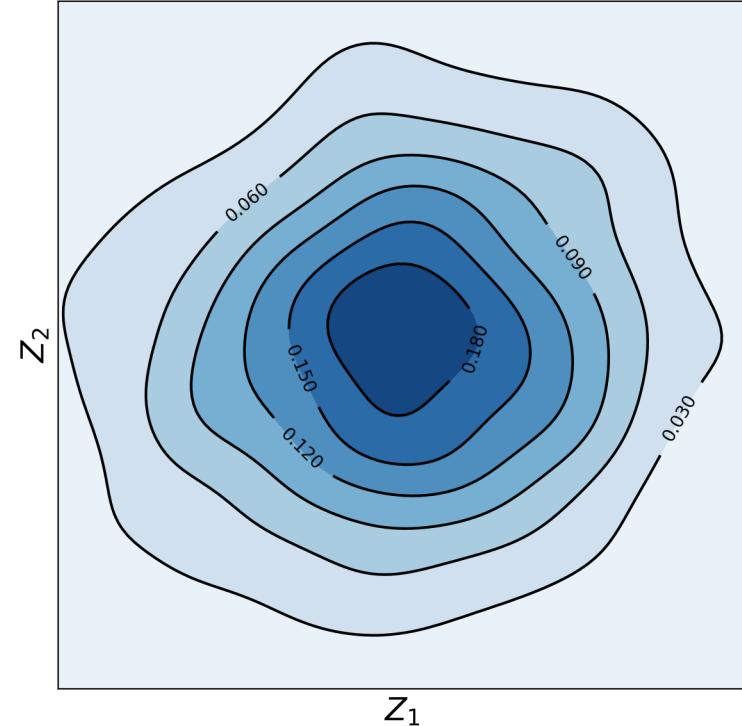
Замена переменных



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$

$$z = f(x)$$



$$z_i \sim p_z(z)$$

$$p_z(z) - \text{известно}$$

Теорема о замене переменных

Пусть даны $p_z(z)$ и $z = \mathbf{f}(x)$, тогда $p_x(x)$ находим так:

$$p_x(x_i) = p_z(\mathbf{f}(x_i)) \left| \det \frac{\partial \mathbf{f}(x_i)}{\partial x_i} \right|,$$

Отношение объема
 ∂z к новому объему
 ∂x

где матрица первых производных определяется так:

$$\frac{\partial \mathbf{f}(x_i)}{\partial x_i} = \begin{pmatrix} \frac{\partial \mathbf{f}(x_i)_1}{\partial x_{i1}} & \dots & \frac{\partial \mathbf{f}(x_i)_1}{\partial x_{in}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{f}(x_i)_m}{\partial x_{i1}} & \dots & \frac{\partial \mathbf{f}(x_i)_m}{\partial x_{in}} \end{pmatrix}.$$

Пример 1

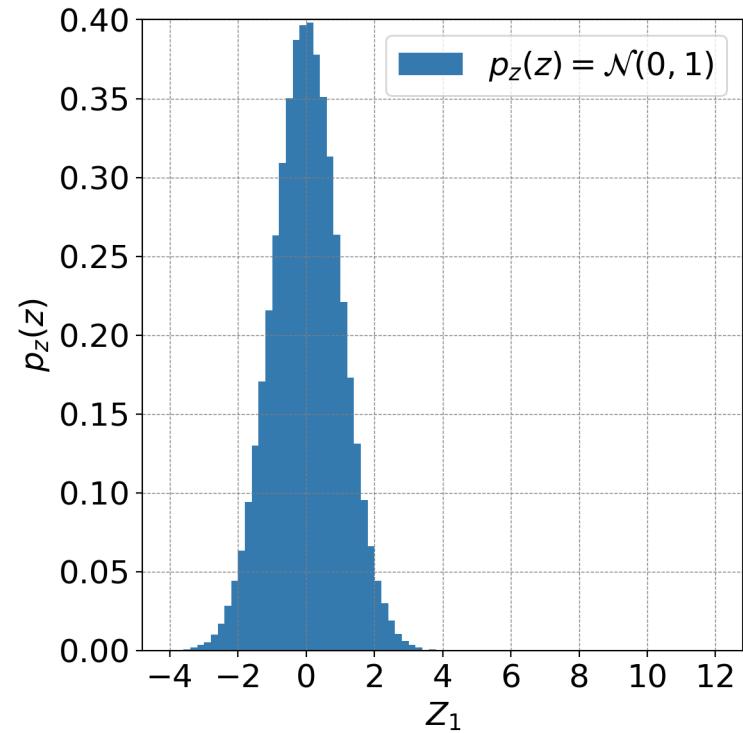


$$z = f(x) = 0.5x - 2.5$$



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$



$$z_i \sim p_z(z)$$

$$p_z(z) = \mathcal{N}(0, 1)$$

Пример 1

Итак, дана функция $\mathbf{f}(x)$:

$$z = \mathbf{f}(x) = 0.5x - 2.5$$

Тогда, матрица первых производных:

$$\frac{\partial \mathbf{f}(x_i)}{\partial x_i} = (0.5)$$

И значение Якобиана:

$$\left| \det \frac{\partial \mathbf{f}(x_i)}{\partial x_i} \right| = 0.5$$

Пример 1

Формула замены переменных:

$$p_x(x_i) = \mathbf{p_z}(\mathbf{f}(x_i)) \left| \det \frac{\partial \mathbf{f}(x_i)}{\partial x_i} \right|,$$

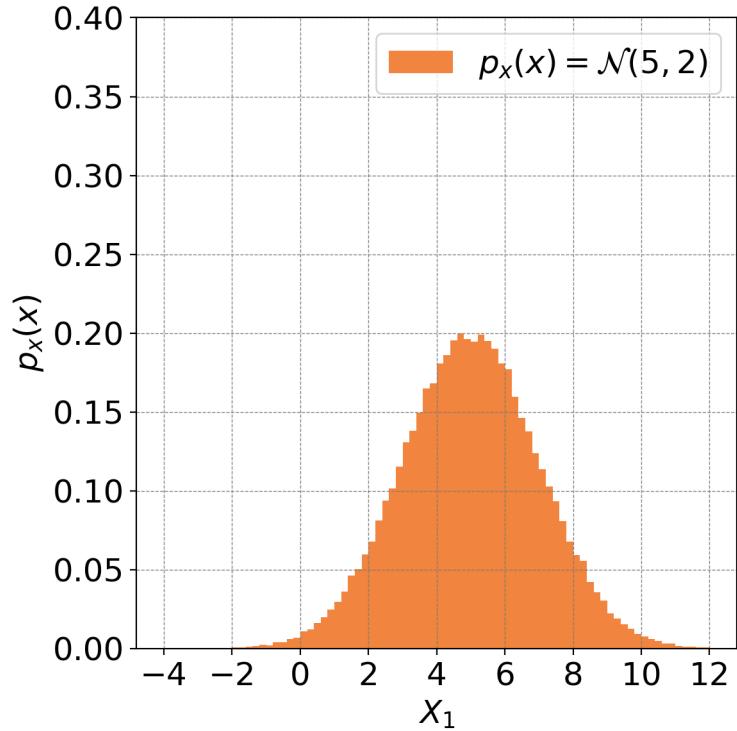
Подставим известные выражения:

$$\mathbf{p_z}(z_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(z_i)^2}{2}}$$

$$p_x(x_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(0.5x_i - 2.5)^2}{2}} * 0.5$$

$$= \frac{1}{2\sqrt{2\pi}} e^{-\frac{(x_i - 5)^2}{2*2^2}} = \mathcal{N}(5, 2)$$

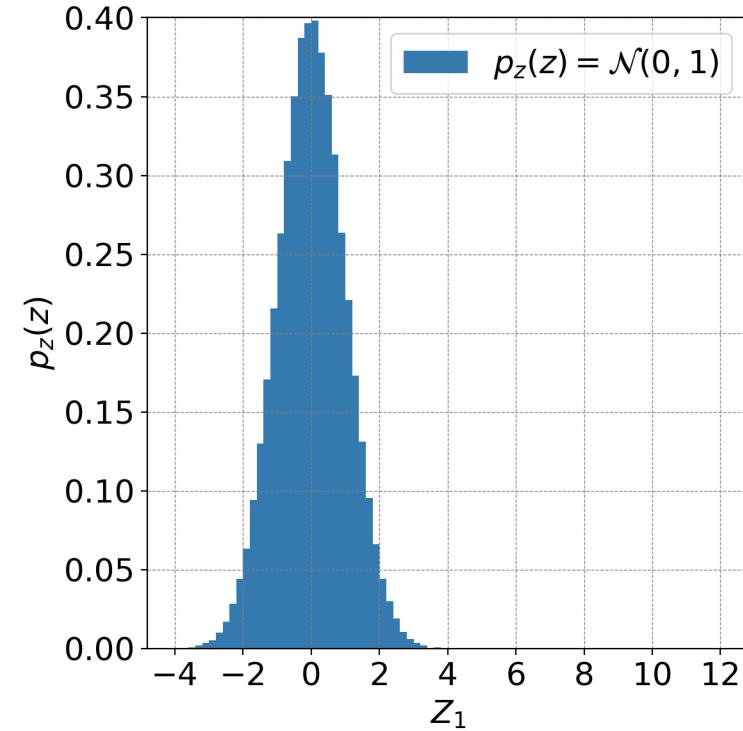
Пример 1



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$

$$z = f(x) = 0.5x - 2.5$$



$$z_i \sim p_z(z)$$

$$p_z(z) = \mathcal{N}(0, 1)$$

Теорема о замене переменных

Пусть даны $p_z(z)$ и $z = \mathbf{f}(x)$, тогда $p_x(x)$ находим так:

$$p_x(x_i) = p_z(\mathbf{f}(x_i)) \left| \det \frac{\partial \mathbf{f}(x_i)}{\partial x_i} \right|,$$

Отношение объема
 ∂z к новому
объему ∂x

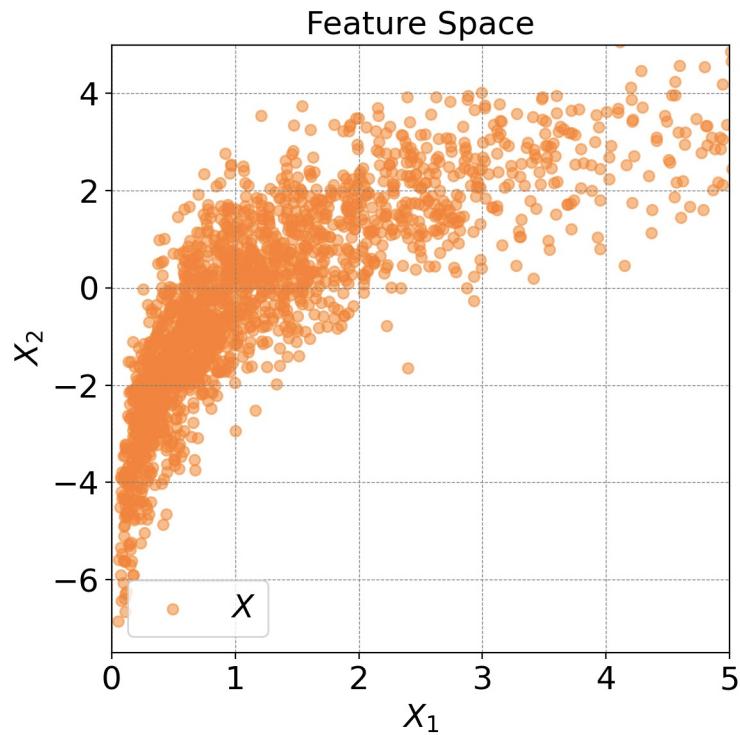
Обратная замена переменных:

$$p_z(z_i) = p_x(\mathbf{f}^{-1}(z_i)) \left| \det \frac{\partial \mathbf{f}^{-1}(z_i)}{\partial z_i} \right|$$

The background consists of a dense, abstract pattern of wavy, horizontal lines in shades of blue and white. The lines are thin and dark blue, creating a sense of depth and motion against a lighter blue background. They form a complex, organic texture that resembles flowing water or a microscopic view of a biological structure.

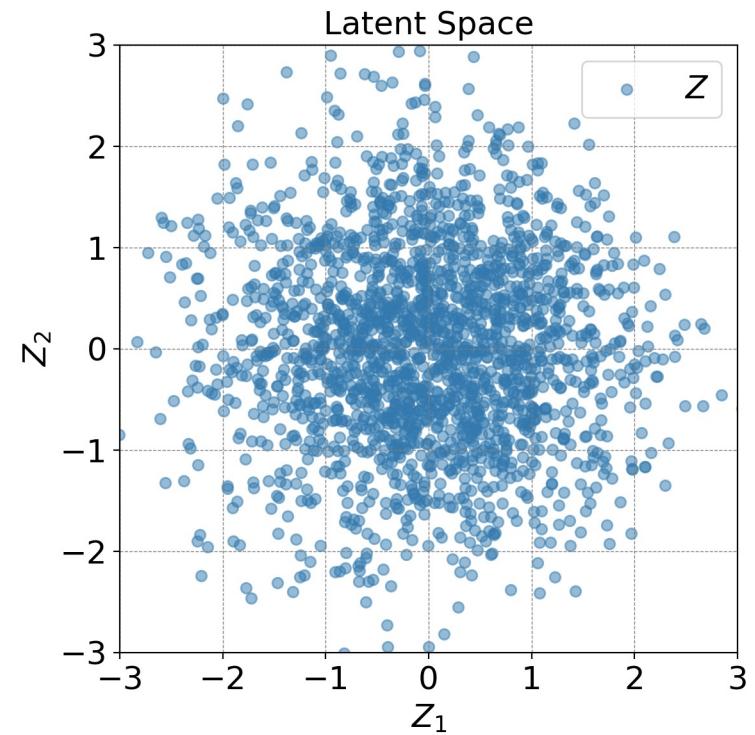
Нормализационные потоки

Постановка задачи



$$z = f(x) - ?$$

A large black arrow pointing to the right, indicating the transformation from the feature space to the latent space.



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$

$$z_i \sim p_z(z)$$

$p_z(z)$ - известно

Постановка задачи

- ▶ **Дано:**
 - матрица реальных объектов X
- ▶ **Задача:**
 - найти такую $z_i = f(x_i)$, чтобы $z_i \sim p_z(z)$
 - при этом, $p_z(z)$ известно и задано

Решение

- ▶ Как будем находить $z_i = \mathbf{f}(x_i)$?
 - ▶ **Ответ:** методом градиентного спуска!
-
- ▶ Какую функцию потерь будем оптимизировать?
 - ▶ **Ответ:** логарифм правдоподобия:

$$L = -\frac{1}{n} \sum_{i=1}^n \log p_x(x_i)$$

Функция потерь

Функция потерь:

$$L = -\frac{1}{n} \sum_{i=1}^n \log p_x(x_i)$$

Замена переменных:

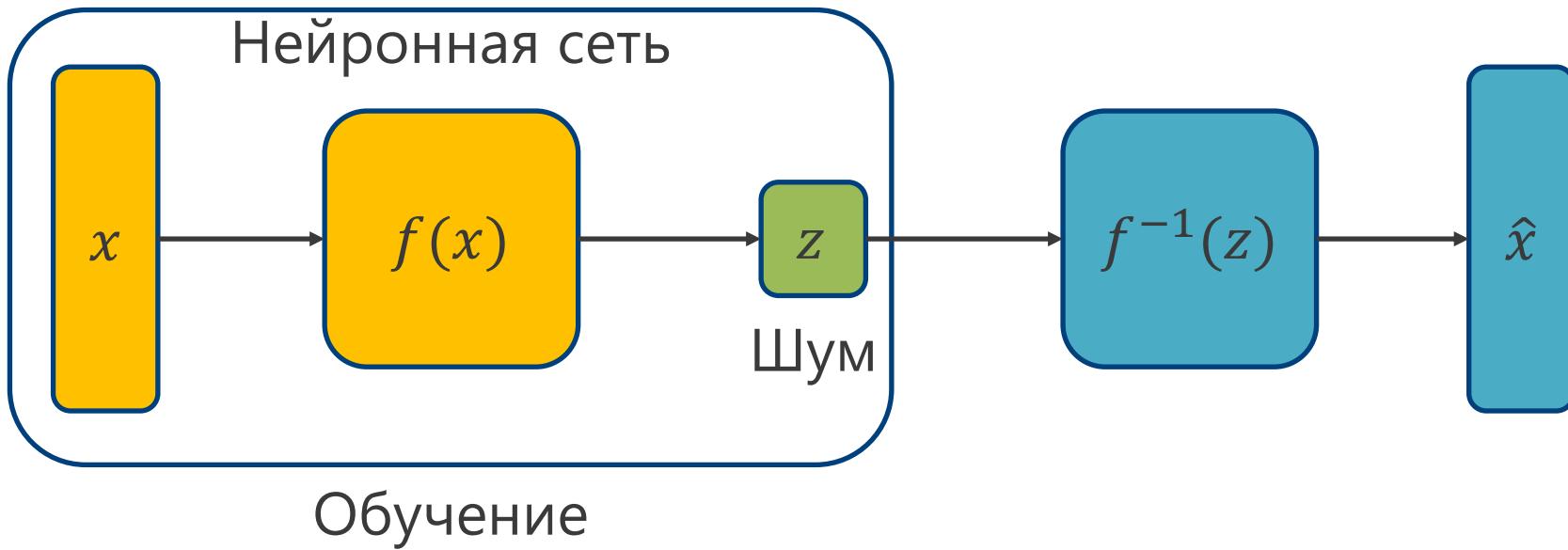
$$p_x(x_i) = p_z(f(x_i)) \left| \det \frac{\partial f(x_i)}{\partial x_i} \right|$$

Подставим в функцию потерь:

$$L = -\frac{1}{n} \sum_{i=1}^n \left(\log p_z(f(x_i)) + \log \left| \det \frac{\partial f(x_i)}{\partial x_i} \right| \right)$$

Общий алгоритм

Алгоритм обучения



Алгоритм обучения

Цикл обучения:

- ▶ Берем m реальных объектов $\{x_1, x_2, \dots, x_m\}$
- ▶ Считаем функцию потерь:

$$L = -\frac{1}{m} \sum_{i=1}^m \left(\log p_z(f(x_i)) + \log \left| \det \frac{\partial f(x_i)}{\partial x_i} \right| \right)$$

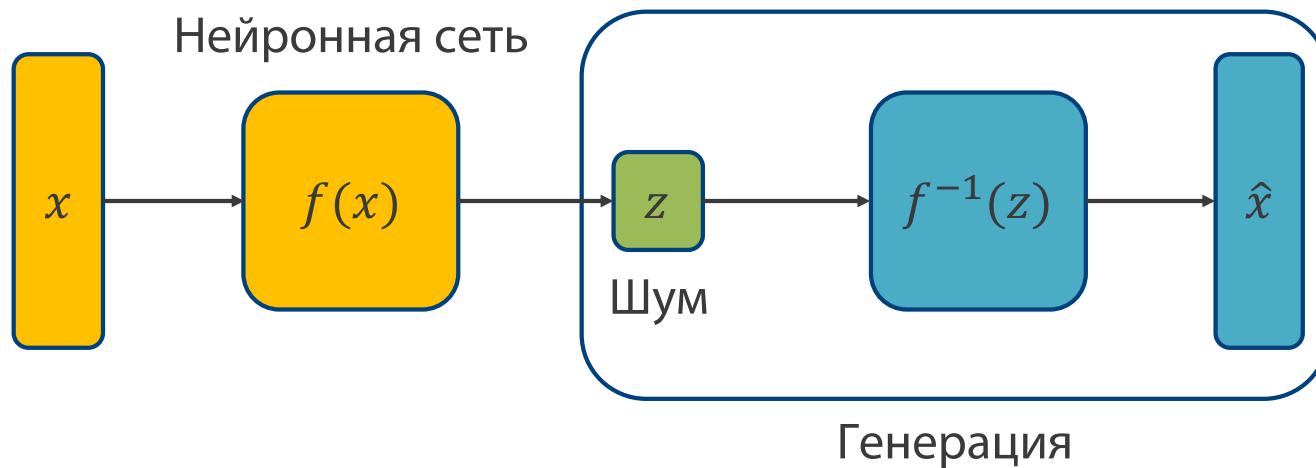
- ▶ Обновляем параметры θ_f функции $z_i = f(x_i)$:

$$\theta_f = \theta_f - \nabla_{\theta_f} L$$

Алгоритм генерации

- ▶ Генерируем случайный шум $\{z_1, z_2, \dots, z_m\}$
- ▶ Генерируем новые объекты $\{x_1, x_2, \dots, x_m\}$ по формуле:

$$x_i = f^{-1}(z_i)$$



Выбор функции

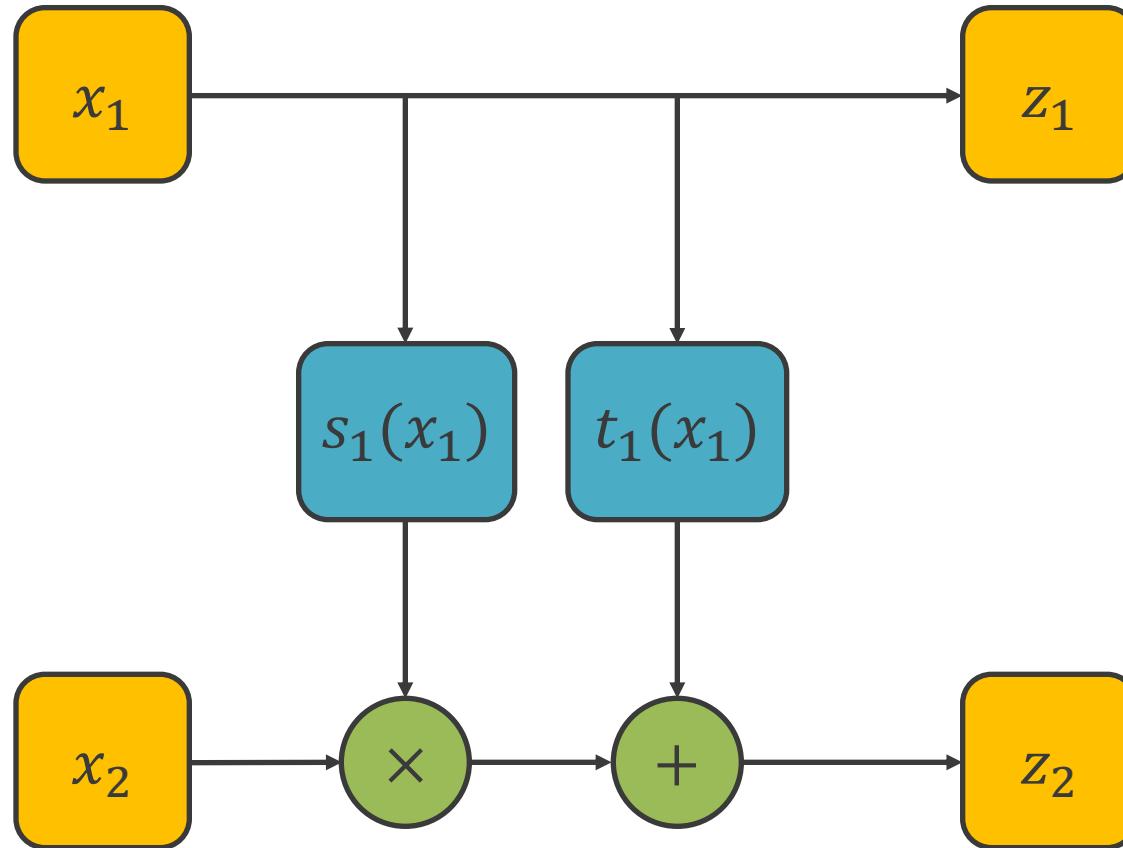
Как выбрать такую функцию $z_i = \mathbf{f}(x_i)$, чтобы она:

- ▶ была дифференцируемой,
- ▶ была обратимой?



Real-NVP

Real-NVP



ФУНКЦИЯ

$$z = \mathbf{f}(x) = \begin{cases} z_{1:d} = x_{1:d} \\ z_{d+1:D} = x_{d+1:D} \odot \exp(\mathbf{s}(x_{1:d})) + \mathbf{t}(x_{1:d}) \end{cases}$$

где:

- ▶ $z_{1:d}$ - первые d компонент вектора z ;
- ▶ $\mathbf{s}(x_{1:d})$ и $\mathbf{t}(x_{1:d})$ – **нейронные сети** с d входами и $D - d$ выходами;
- ▶ \odot – поэлементное умножение.

<https://arxiv.org/abs/1605.08803>

Якобиан

- Матрица первых производных:

$$\frac{\partial \mathbf{f}(x)}{\partial x} = \begin{pmatrix} \mathbb{I}_d & 0 \\ \frac{\partial z_{1:d}}{\partial x_{1:d}} & diag(\exp(\mathbf{s}(x_{1:d}))) \end{pmatrix}$$

- Значение Якобиана:

$$\left| \det \frac{\partial \mathbf{f}(x)}{\partial x} \right| = \exp\left(\sum_{j=d+1}^D \mathbf{s}(x_{1:d})_j \right)$$

<https://arxiv.org/abs/1605.08803>

Обратная функция

$$x = \mathbf{f}^{-1}(z) = \begin{cases} x_{1:d} = z_{1:d} \\ x_{d+1:D} = (z_{d+1:D} - \mathbf{t}(x_{1:d})) \odot \exp(-\mathbf{s}(x_{1:d})) \end{cases}$$

<https://arxiv.org/abs/1605.08803>

Пример



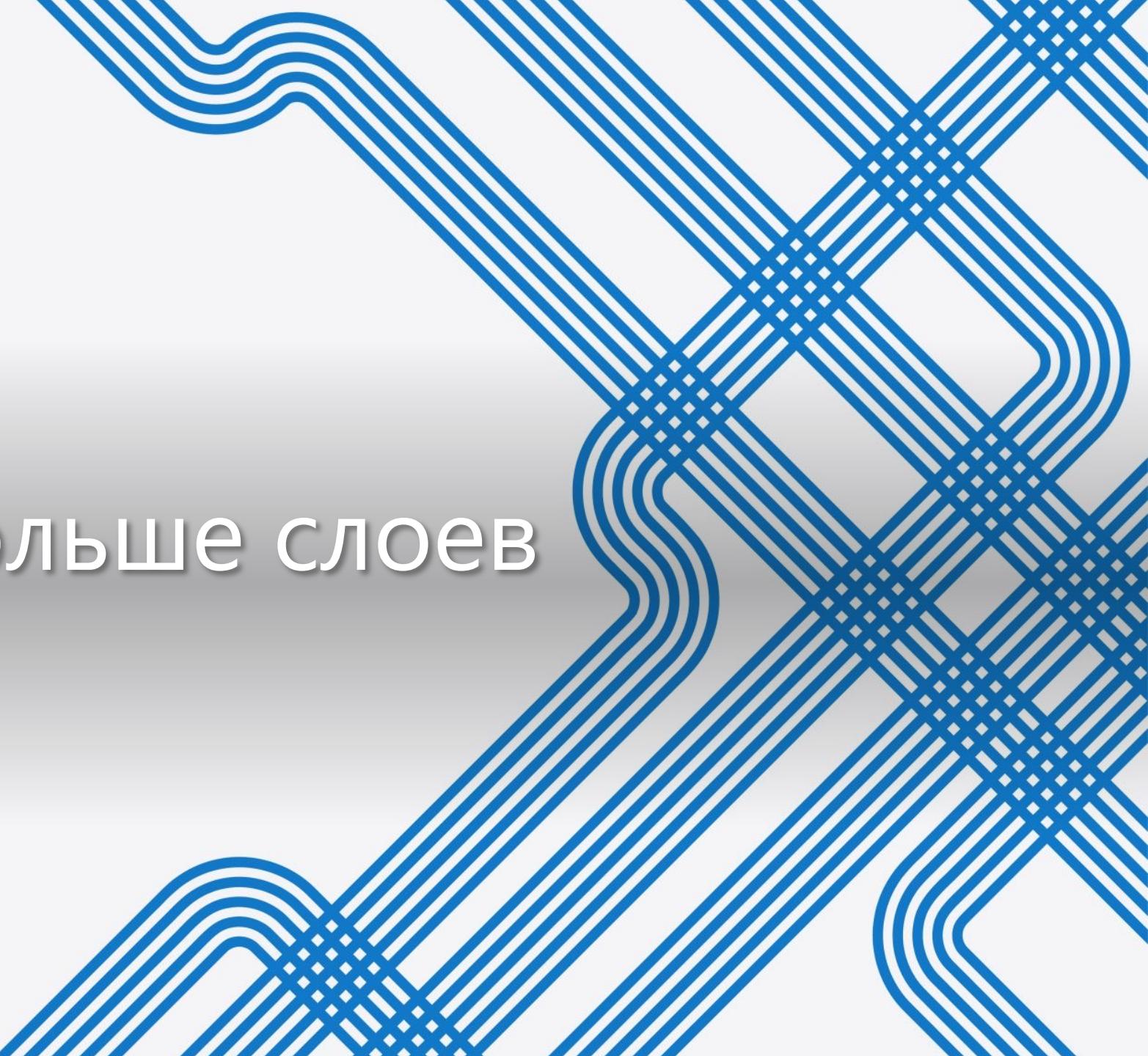
Рис.: https://github.com/laurent-dinh/laurent-dinh.github.io/blob/master/img/real_nvp_fig/celeba_samples.png

Пример Glow

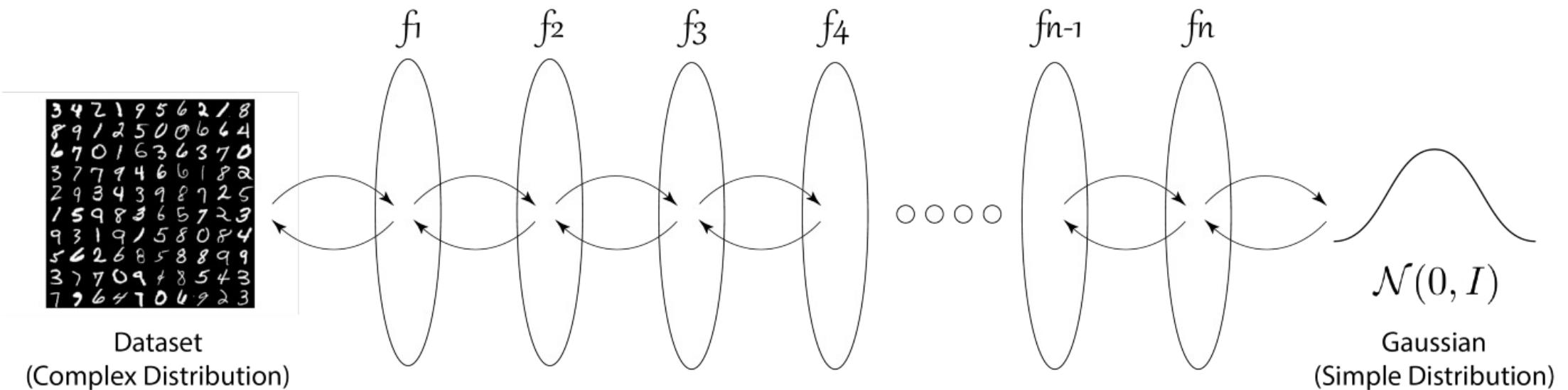


<https://arxiv.org/pdf/1807.03039.pdf>

Больше слоев



Больше слоев!



<https://towardsdatascience.com/introduction-to-normalizing-flows-d002af262a4b>

Больше слоев!

Пусть $z_i = \mathbf{f}_2(y_i)$, $y_i = \mathbf{f}_1(x_i)$.

Тогда:

$$p_x(x_i) = p_y(f_1(x_i)) \left| \det \frac{\partial f_1(x_i)}{\partial x_i} \right|$$

$$p_y(y_i) = p_z(f_2(y_i)) \left| \det \frac{\partial f_2(y_i)}{\partial y_i} \right|$$

В итоге получим:

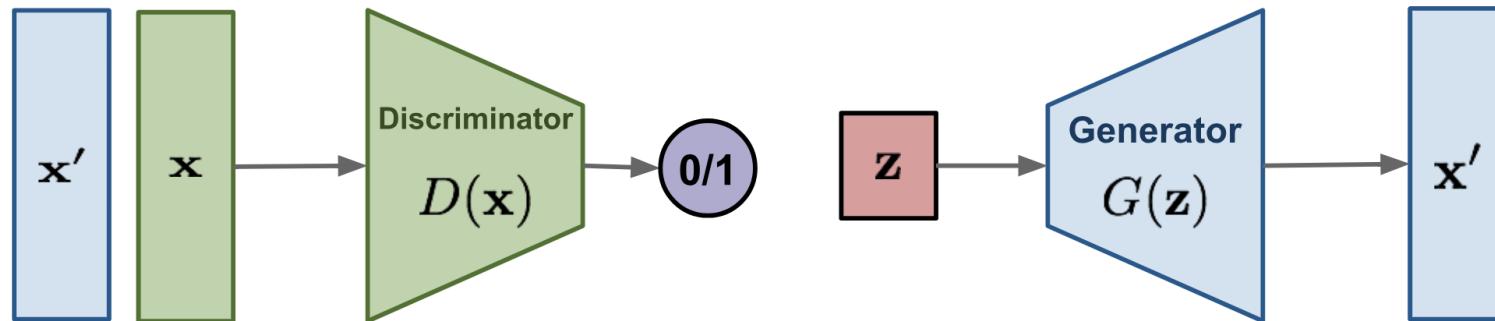
$$p_x(x_i) = p_z(f_2(f_1(x_i))) \left| \det \frac{\partial f_2(y_i)}{\partial y_i} \right| \left| \det \frac{\partial f_1(x_i)}{\partial x_i} \right|$$

Другие модели

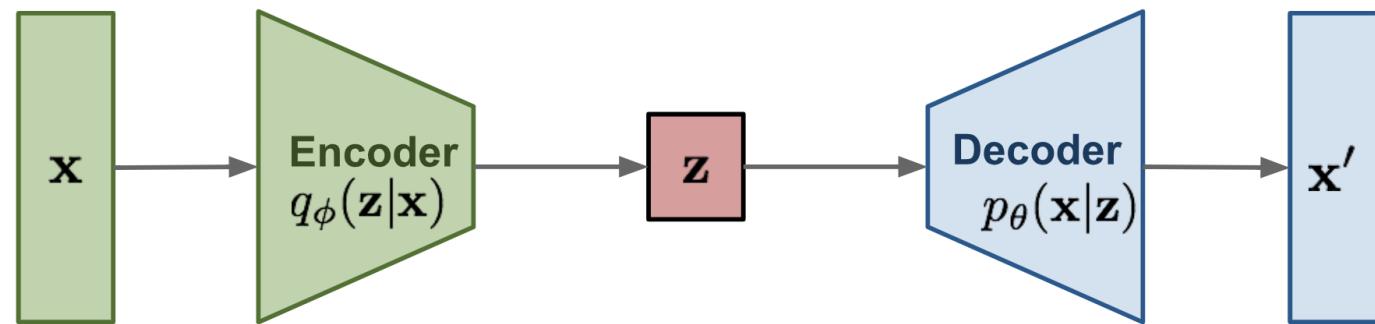
Architecture	Coupling function	Flow name
Coupling, 3.4.1	Affine, 3.4.4.1	RealNVP Glow
	Mixture CDF, 3.4.4.3	Flow++
	Splines, 3.4.4.4	quadratic (C) cubic RQ-NSF(C)
	Piecewise Bijective, 3.4.4.7	RAD
Autoregressive, 3.4.2	Affine	MAF
	Polynomial, 3.4.4.6	SOS
	Neural Network, 3.4.4.5	NAF UMNN
	Splines	quadratic (AR) RQ-NSF(AR)
Residual, 3.5		iResNet
		Residual flow
ODE, 3.6.1		FFJORD

Генеративные модели

GAN: minimax the classification error loss.

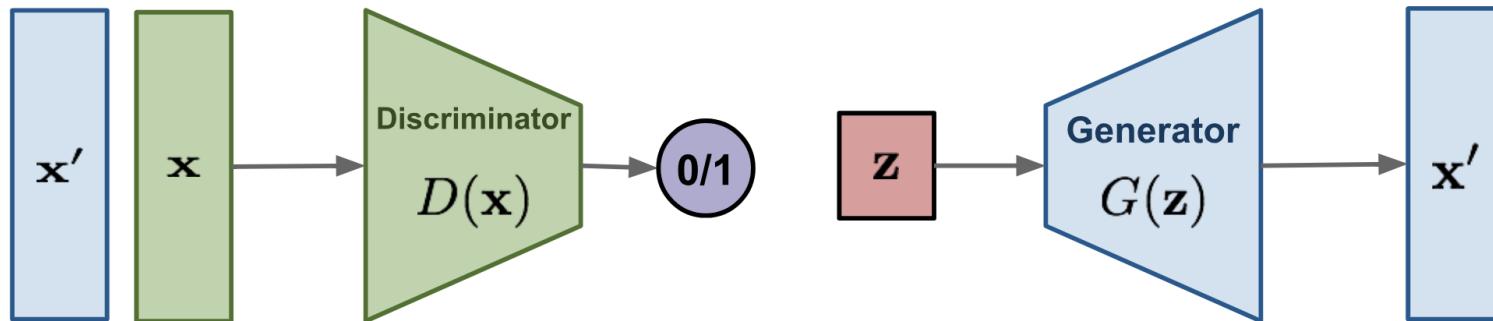


VAE: maximize ELBO.

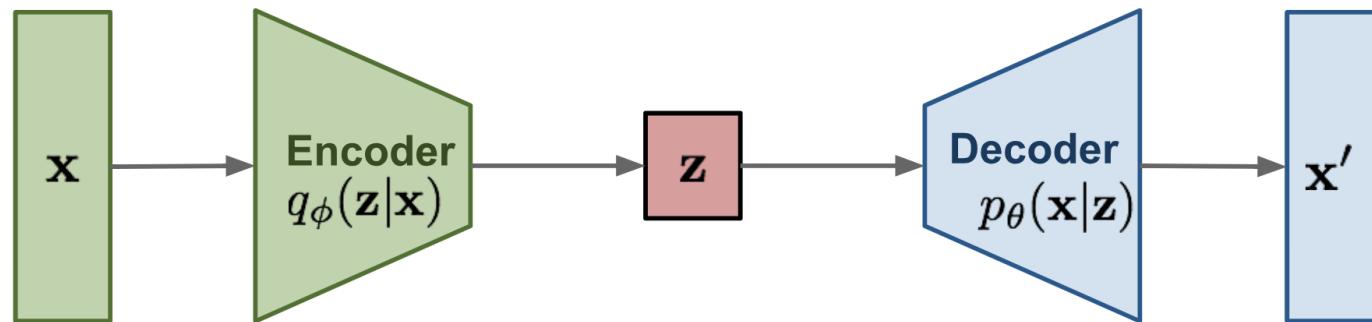


Генеративные модели

GAN: minimax the classification error loss.



VAE: maximize ELBO.



Flow-based generative models: minimize the negative log-likelihood

