

# Прикладные задачи анализа данных

Лекция 4  
Нормализационные потоки

Михаил Гущин

[mhushchyn@hse.ru](mailto:mhushchyn@hse.ru)

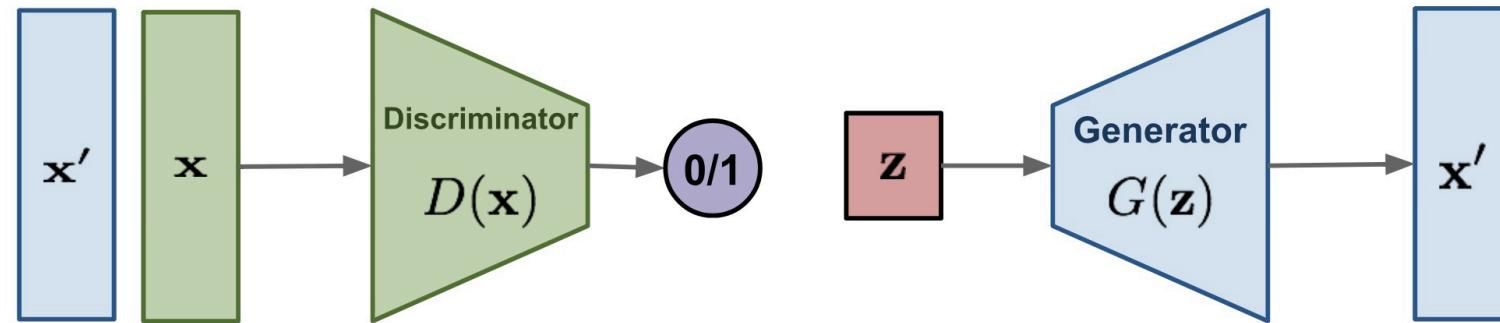
НИУ ВШЭ, 2022



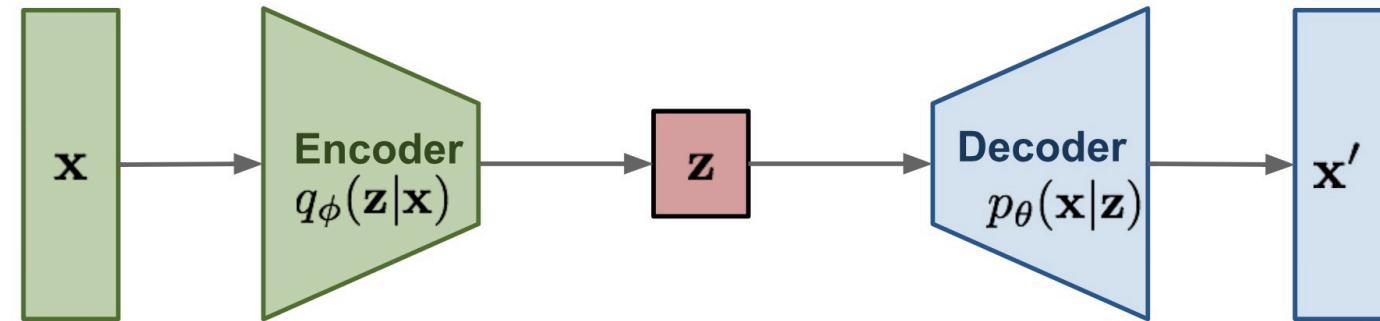
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ

# Генеративные модели

**GAN:** minimax the classification error loss.

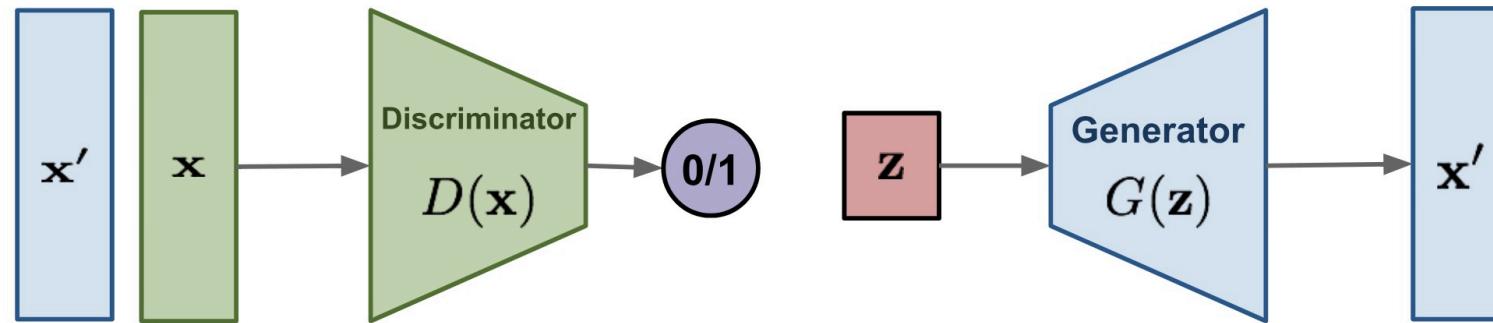


**VAE:** maximize ELBO.

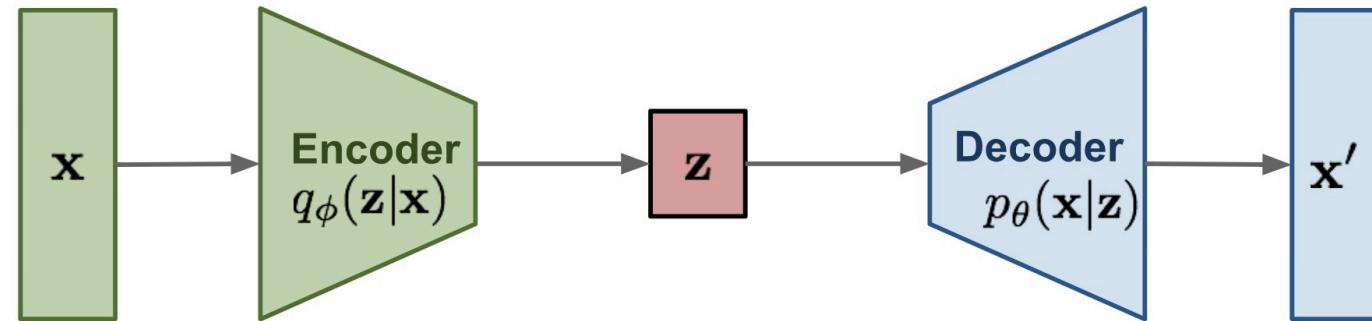


# Генеративные модели

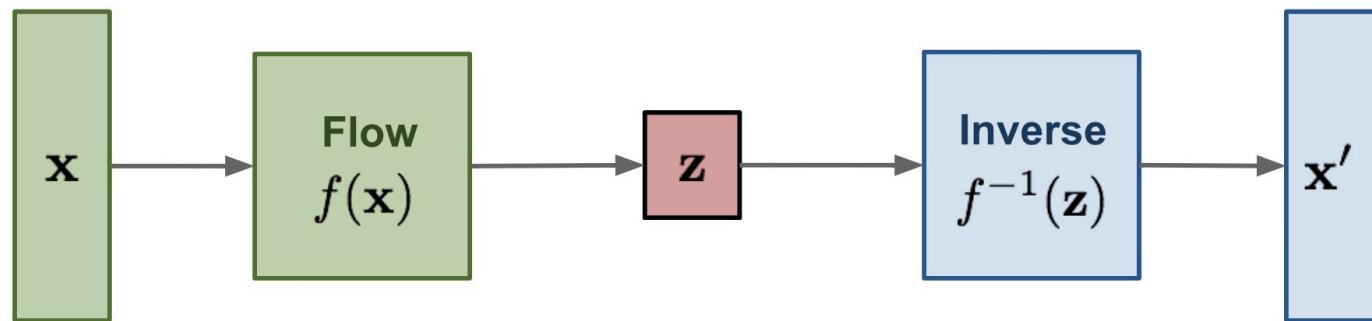
**GAN:** minimax the classification error loss.



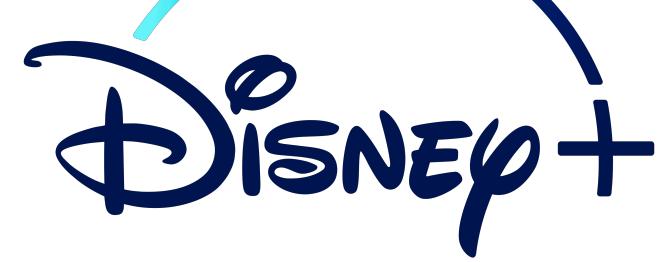
**VAE:** maximize ELBO.



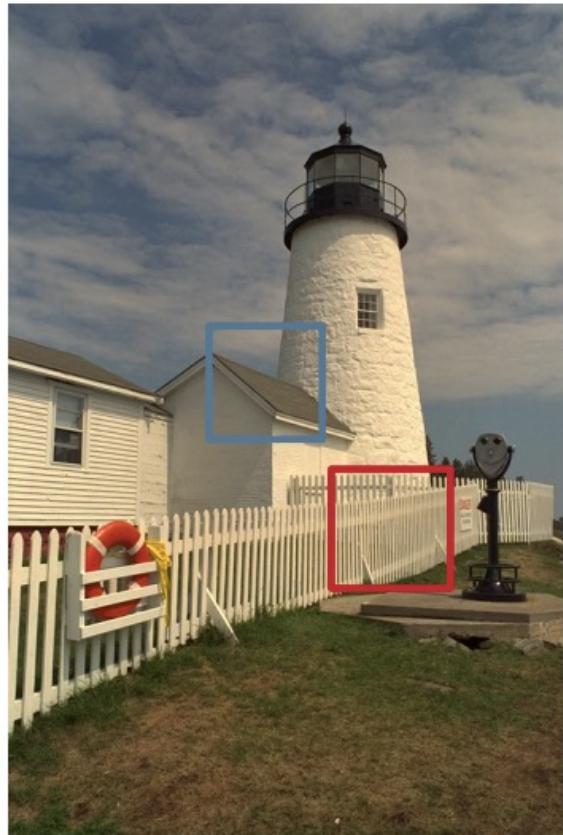
**Flow-based generative models:** minimize the negative log-likelihood



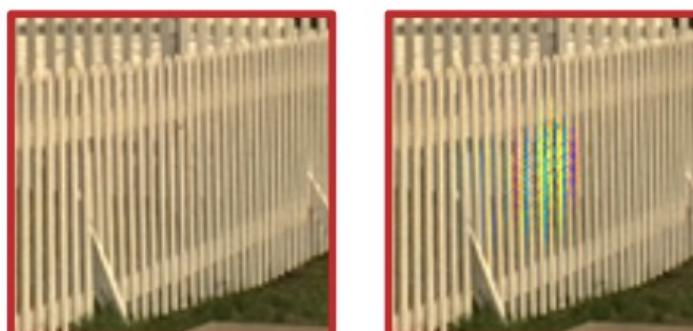
Пример



# Пример



Autoencoder based  
(Minnen et al. 2018)



$N = 1$

$N = 15$

Ours

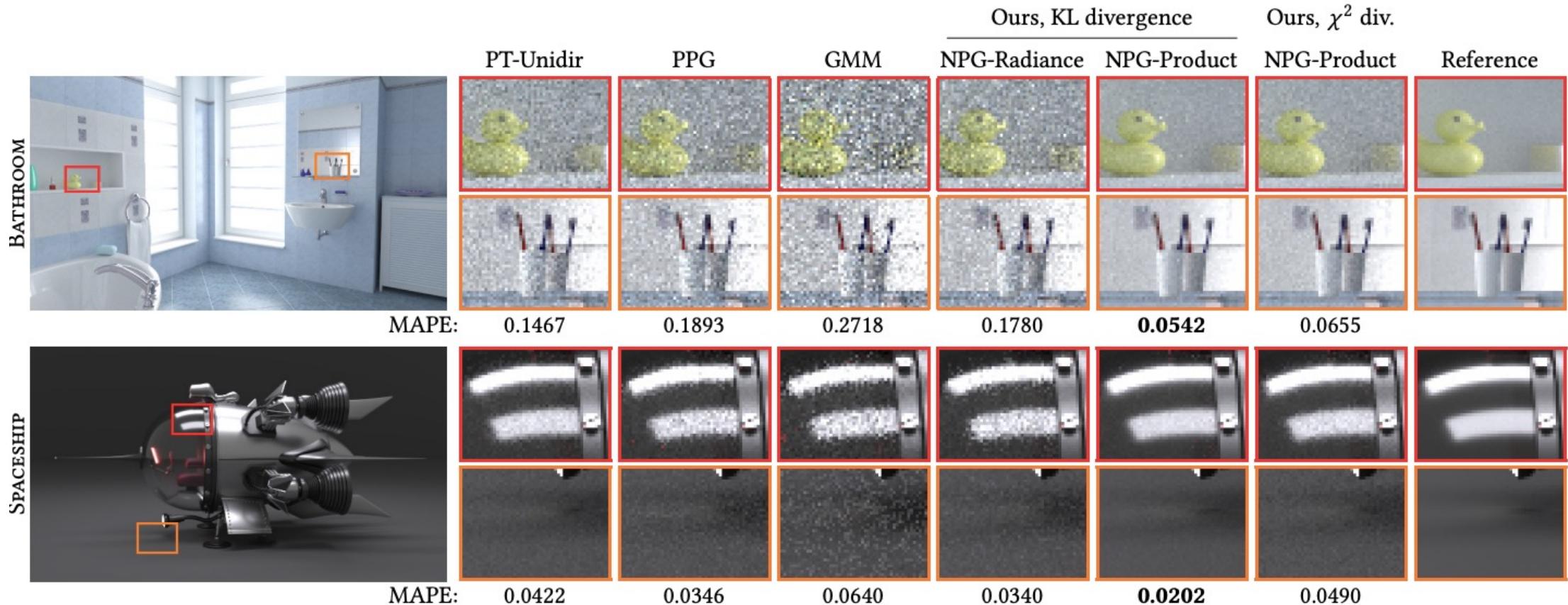


$N = 1$

$N = 15$

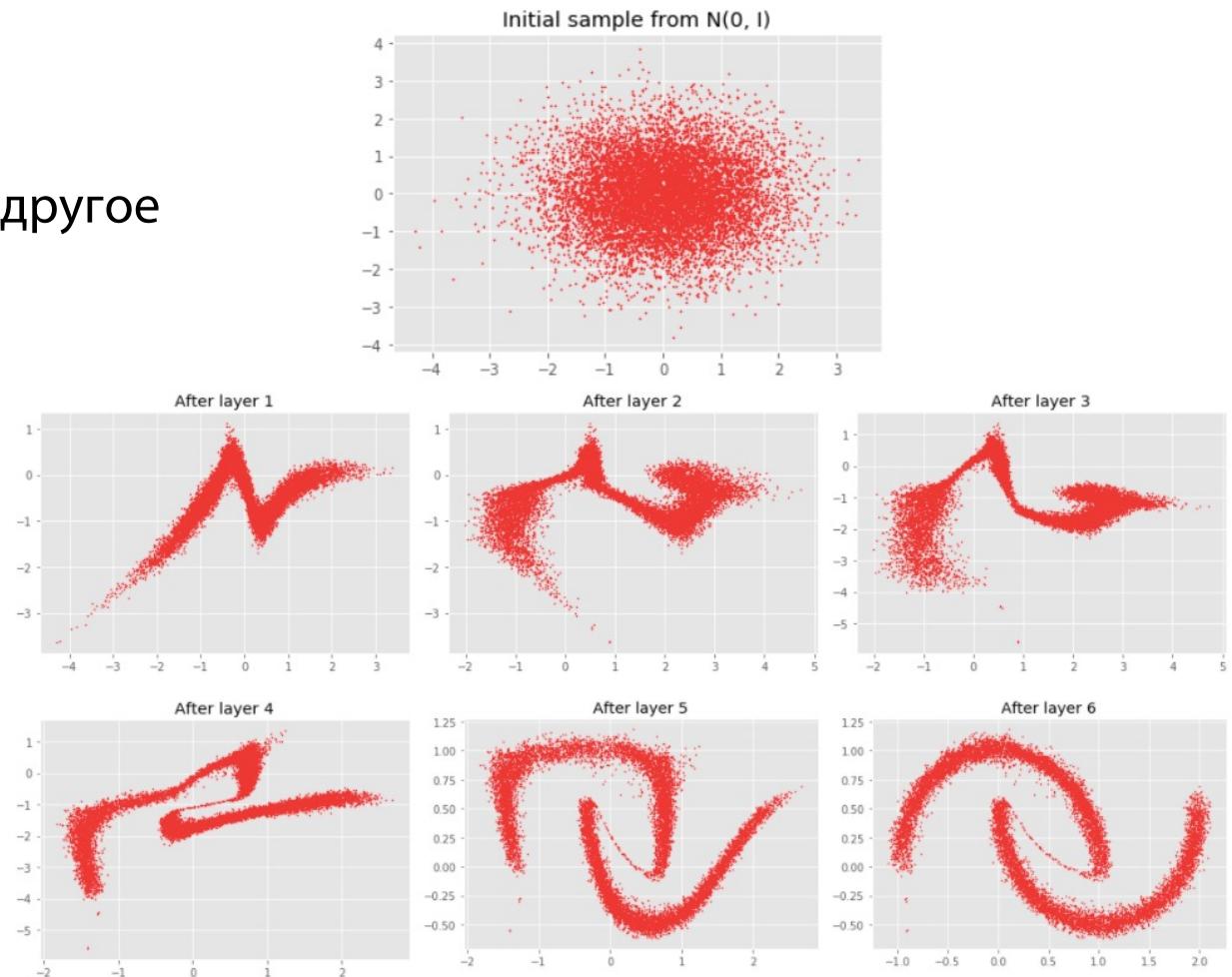
Results after  $N$  compression/decompression operations

# Пример



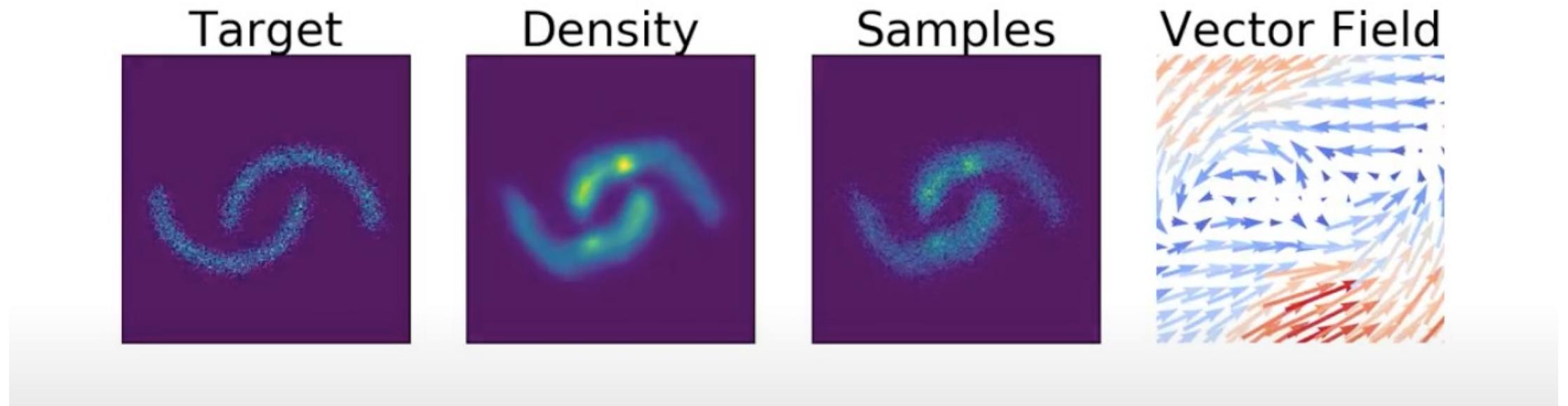
# Интуиция

Нормализационный поток –  
**последовательность обратимых**  
преобразований одного распределения в другое



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

# Интуиция

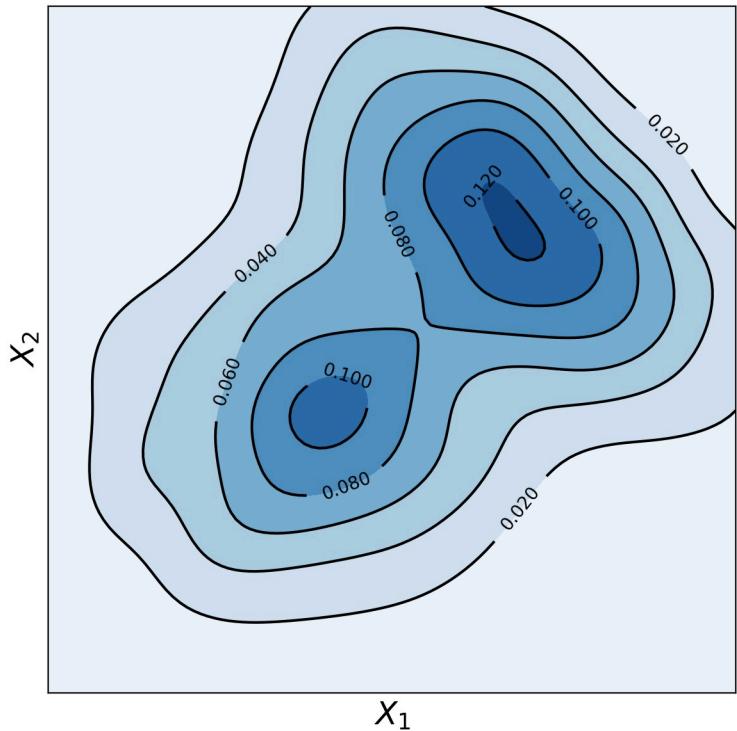


<https://youtu.be/QQpz7V3OgFc>

# Теорема о замене переменных



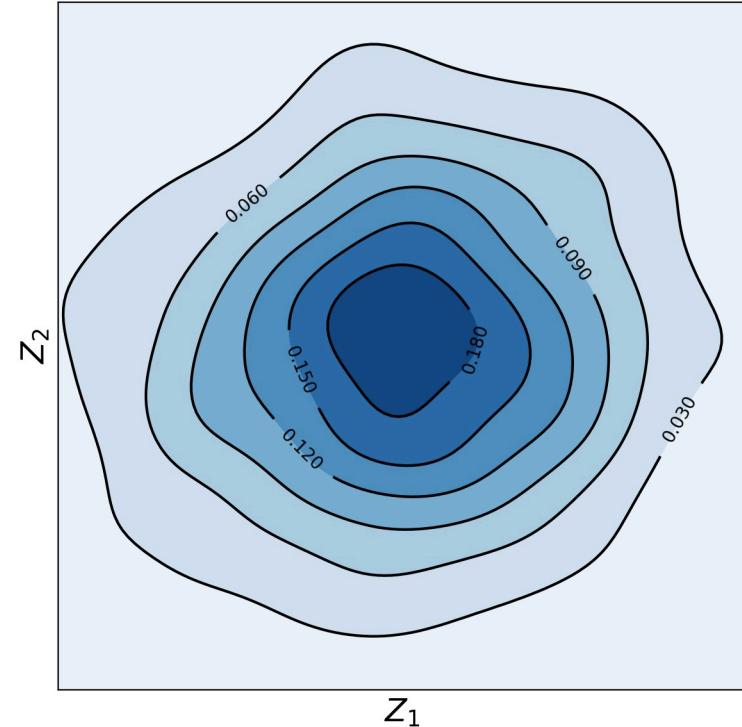
# Замена переменных



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$

$$z = f(x)$$

$$z_i \sim p_z(z)$$

$p_z(z)$  - 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  $\partial z$   
к новому объему  $\partial x$

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

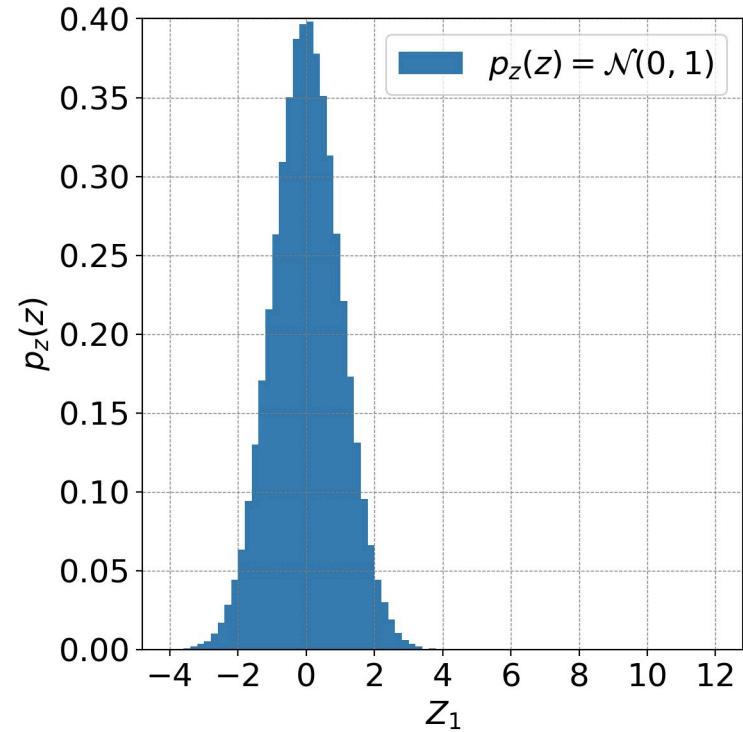


$$z = f(x) = 0.5x - 2.5$$



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$



$$z_i \sim p_z(z)$$

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

# 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:

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

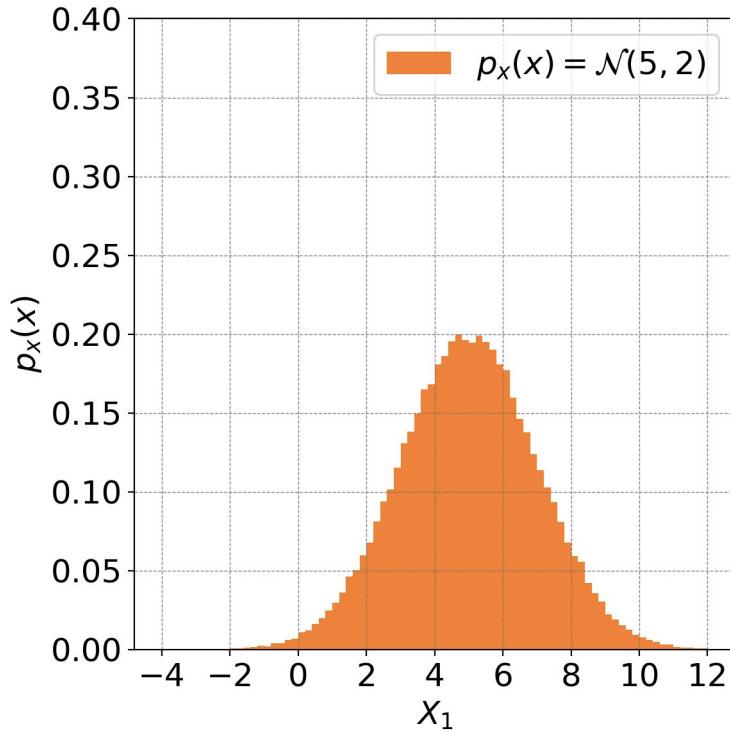
Подставим известные выражения:

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

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

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

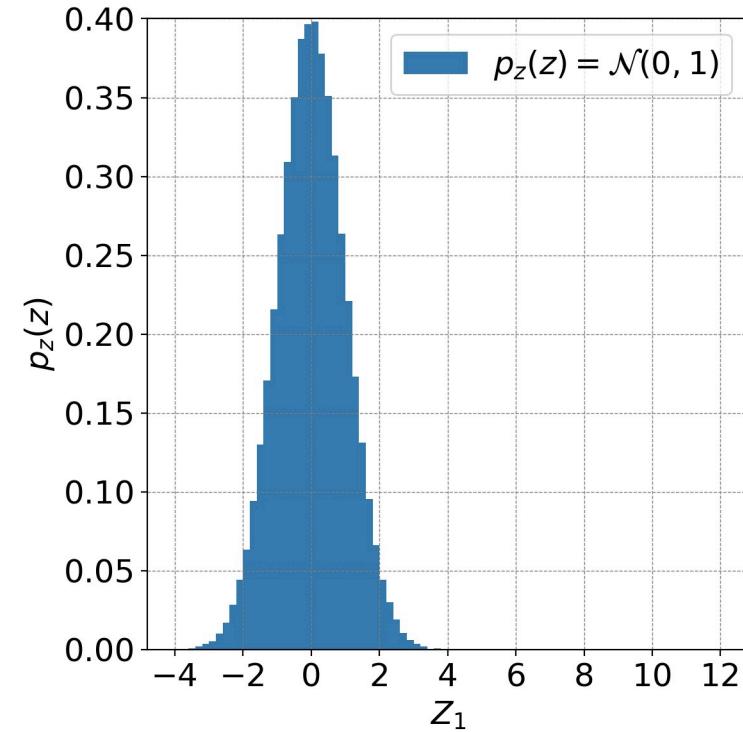
# Пример 1



$$x_i \sim p_x(x)$$

$$p_x(x) - ?$$

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



$$z_i \sim p_z(z)$$

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

# 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  $\partial z$   
к новому объему  $\partial x$

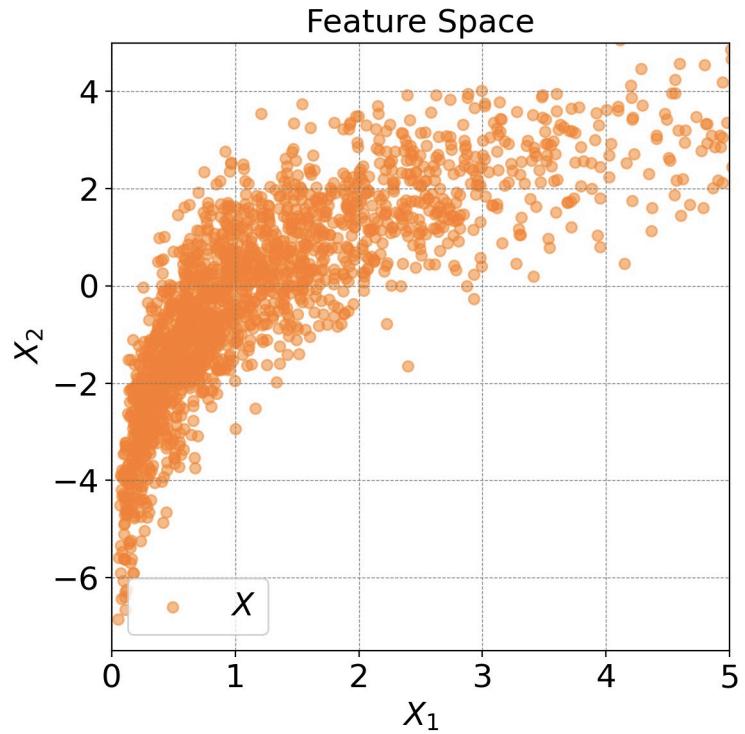
Обратная замена переменных:

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

# Нормализационные потоки

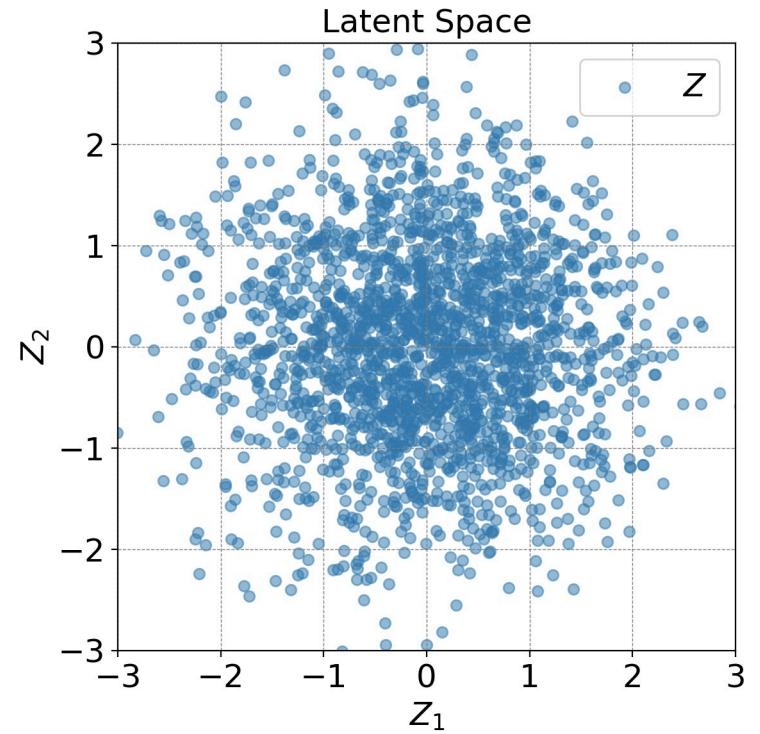


# Постановка задачи



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

A large black arrow pointing to the right, indicating a transformation or mapping 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:

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

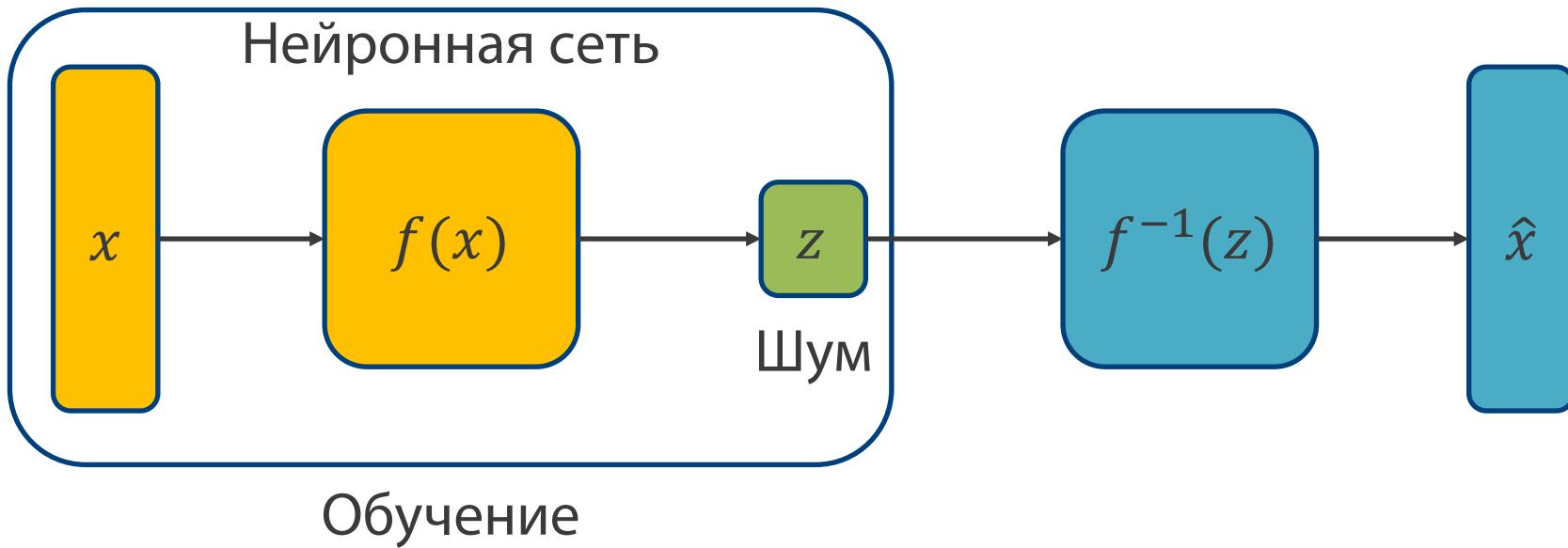
Подставим в функцию потерь:

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

# Общий алгоритм



# Алгоритм обучения



# Алгоритм обучения

Цикл обучения:

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

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

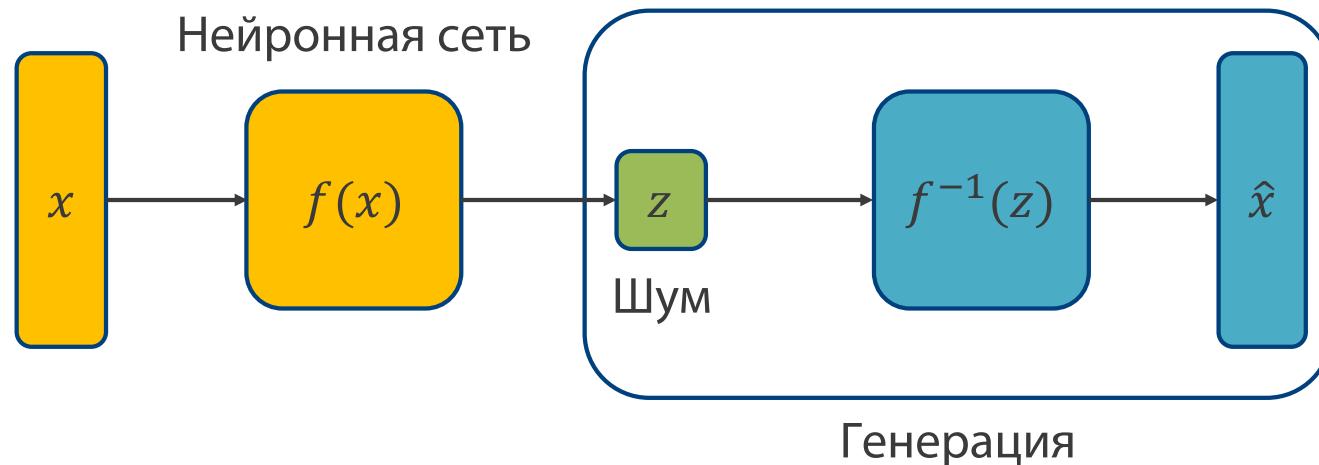
- ▶ Обновляем параметры  $\theta_f$  функции  $z_i = f(x_i)$ :

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

# 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

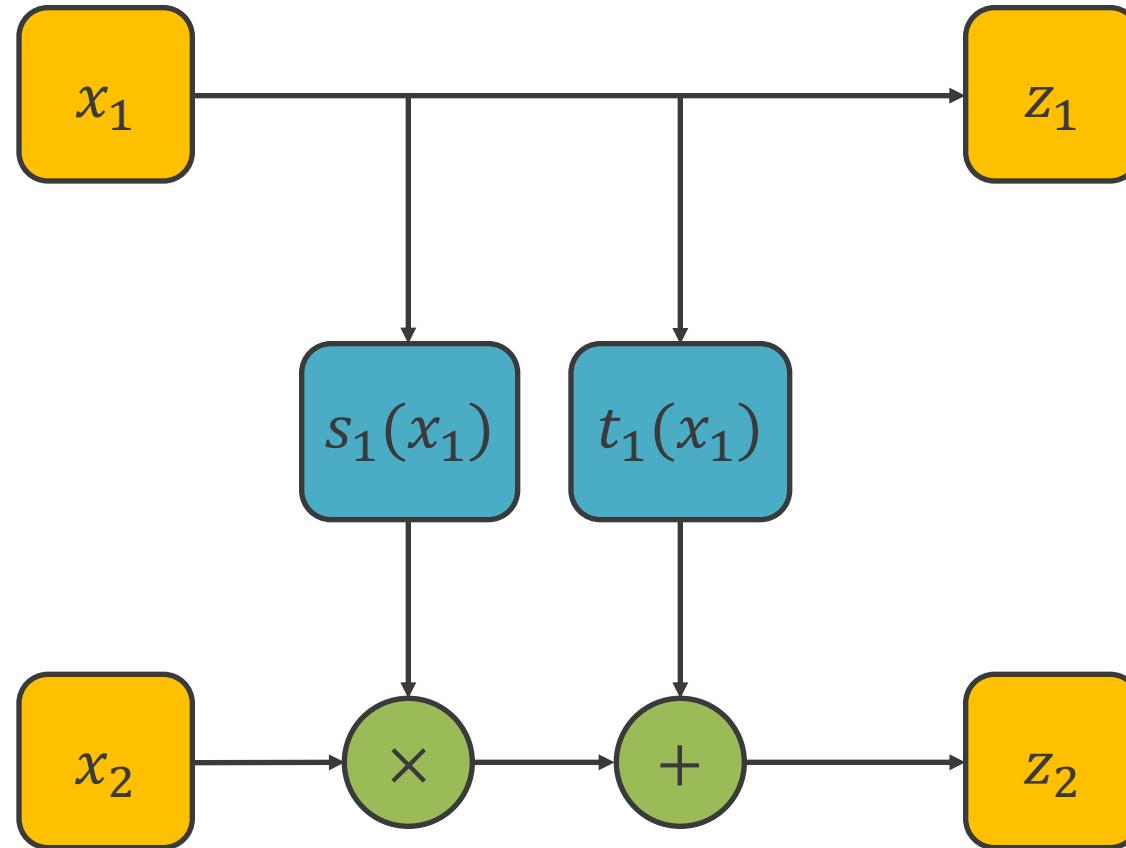
Как выбрать такую функцию  $z_i = \mathbf{f}(x_i)$ , чтобы она:

- ▶ 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



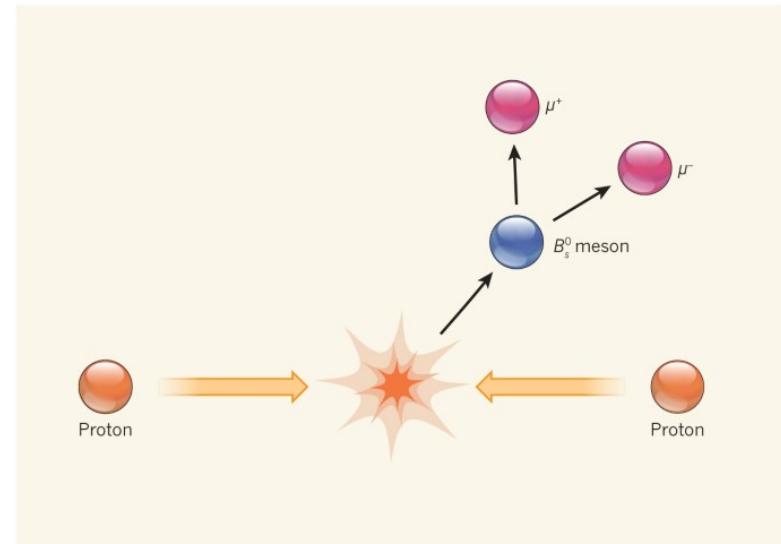
Рис.: [https://github.com/laurent-dinh/laurent-dinh.github.io/blob/master/img/real\\_nvp\\_fig/celeba\\_samples.png](https://github.com/laurent-dinh/laurent-dinh.github.io/blob/master/img/real_nvp_fig/celeba_samples.png)

# Пример Glow

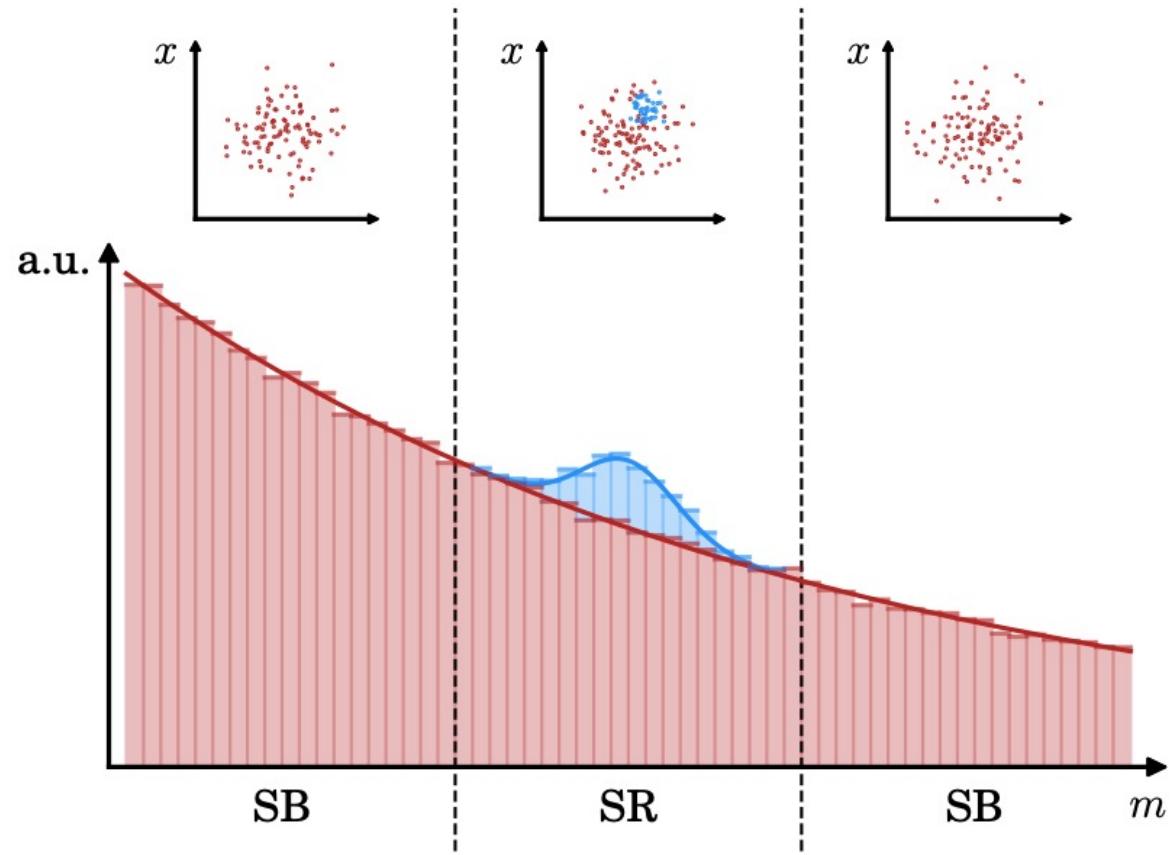


<https://arxiv.org/pdf/1807.03039.pdf>

# Поиск новой физики



Nature 522, 42–43 (2015)



$$p_{\text{data}}(x|m \in SB) = p_{\text{bg}}(x|m \in SB)$$

$$p_{\text{data}}(x|m \in SR)$$

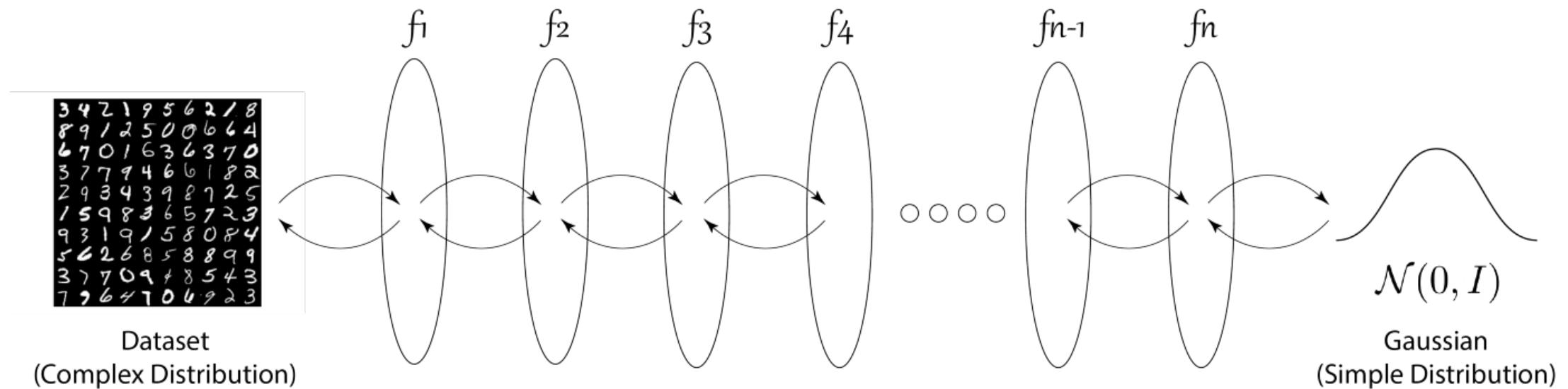
$$p_{\text{data}}(x|m \in SB) = p_{\text{bg}}(x|m \in SB)$$

FERMILAB-PUB-21-389-T

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)
Autoregressive, 3.4.2	Piecewise Bijective, 3.4.4.7	RAD
	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