

Generative Adversarial Networks

Генеративно-состязательные сети

Работу подготовил: Нуриев Айнур Зуфарович

Generative Adversarial Networks

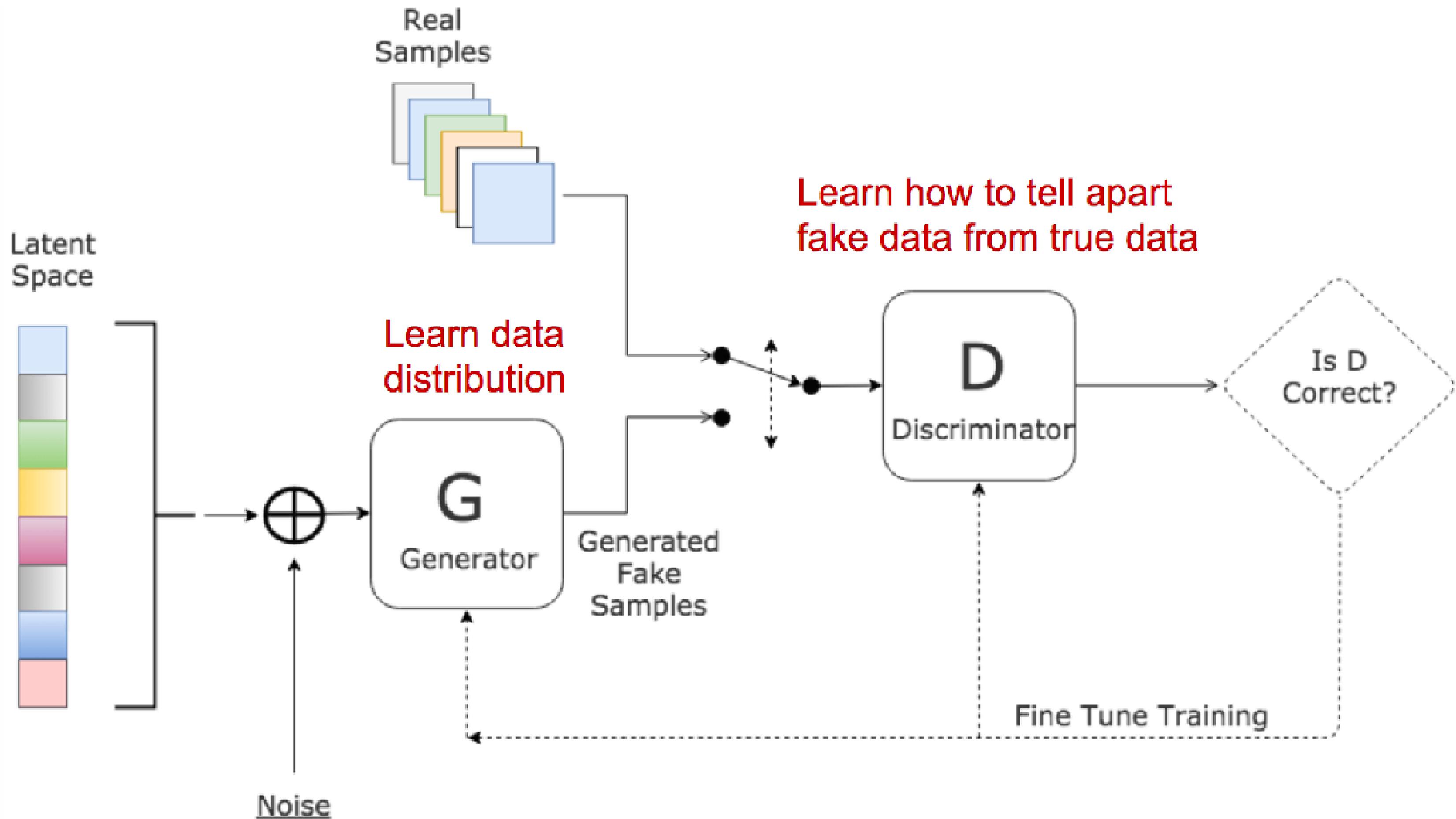
Генеративно-состязательные сети

Ian Goodfellow

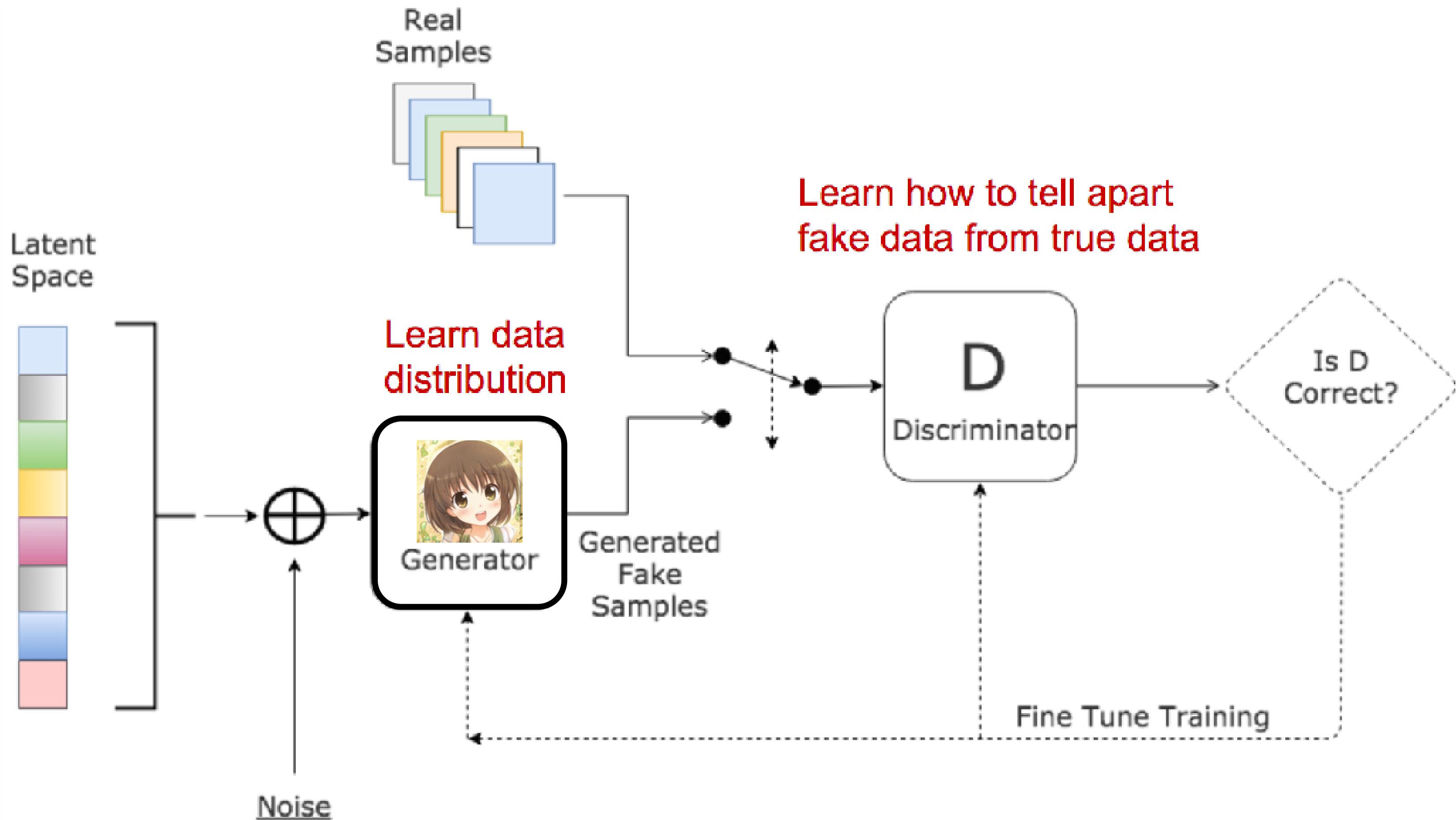
I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde- Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in neural information processing systems, 2014, pp. 2672–2680.



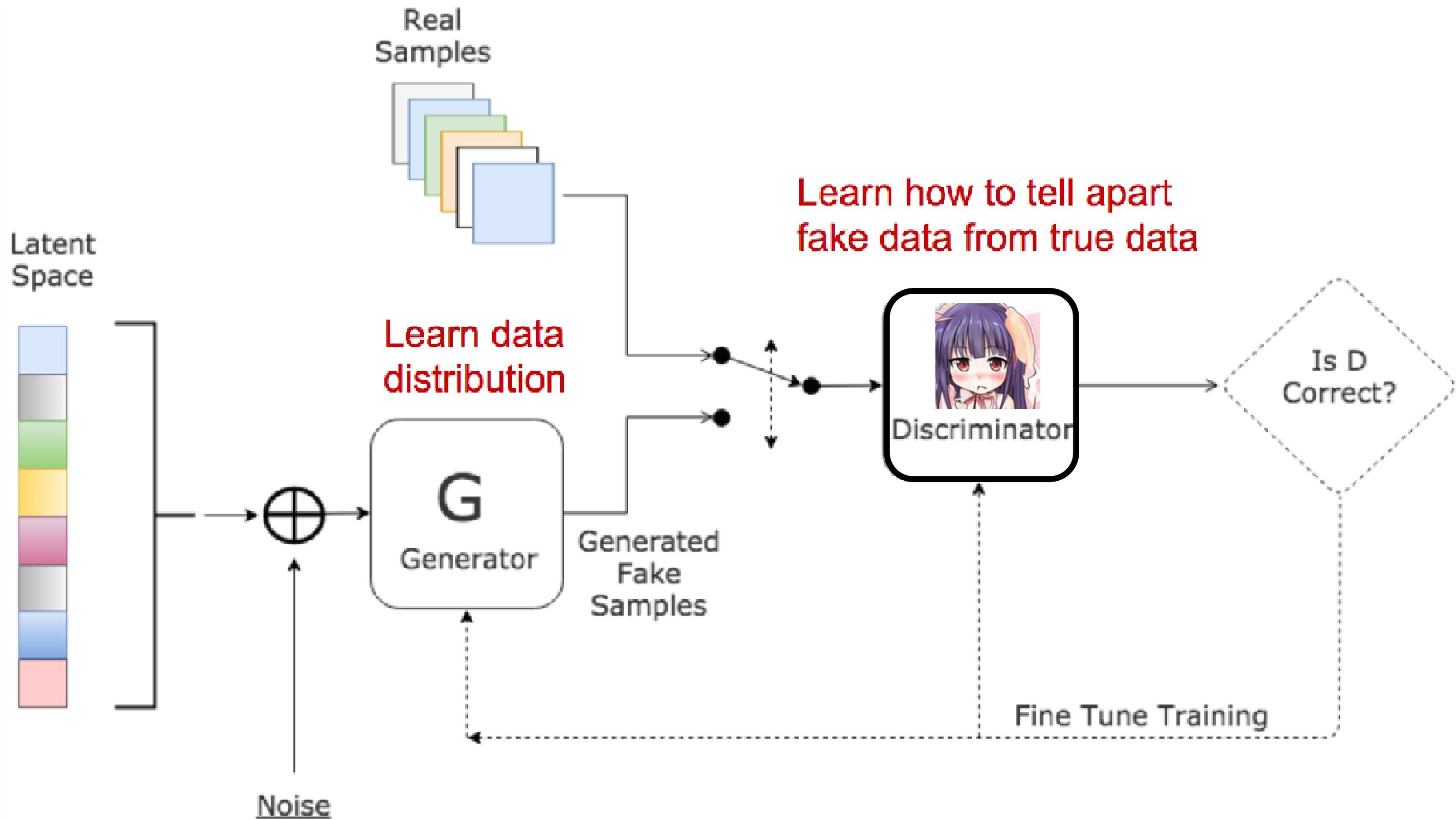
/Архитектура



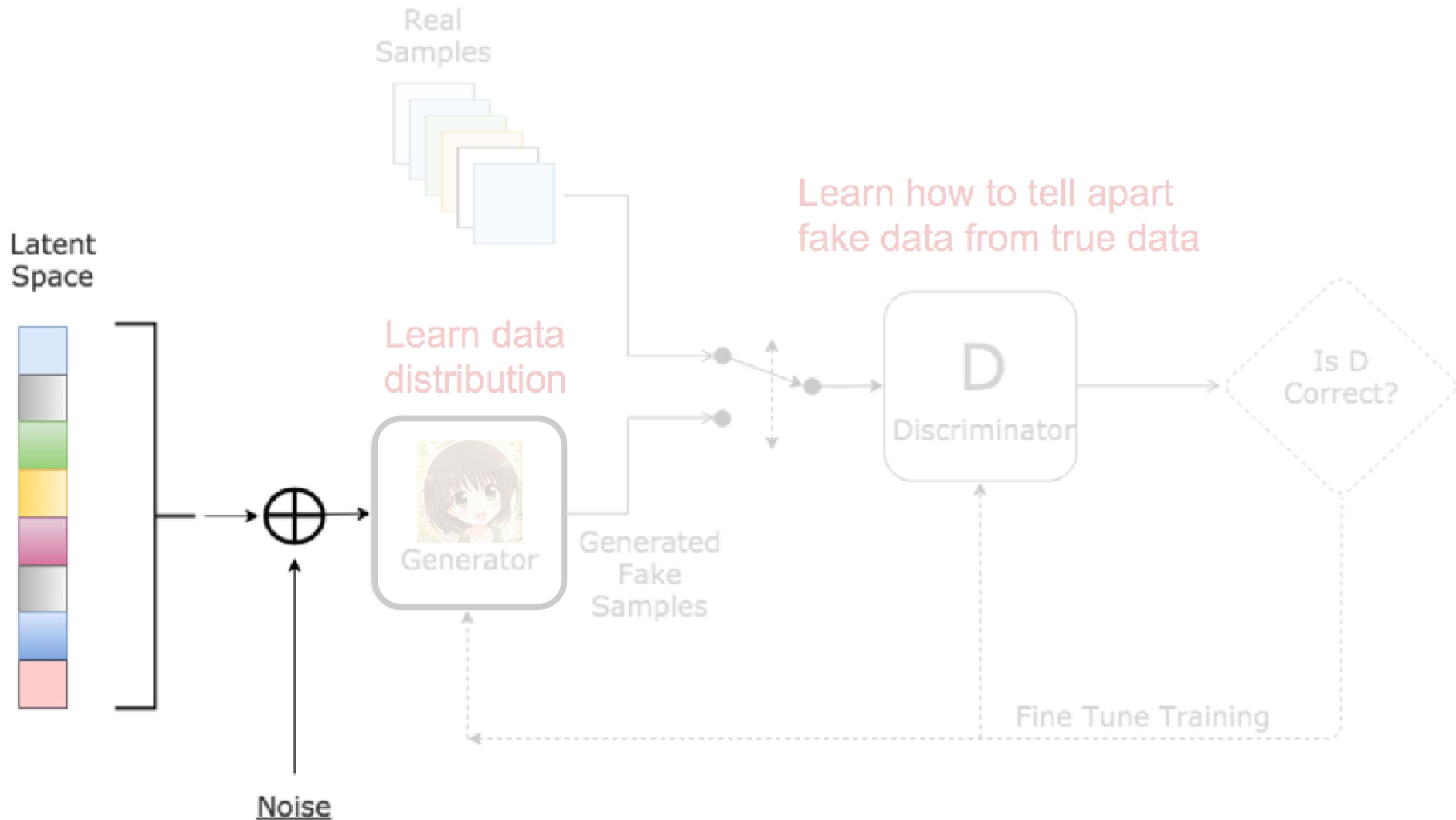
/Архитектура



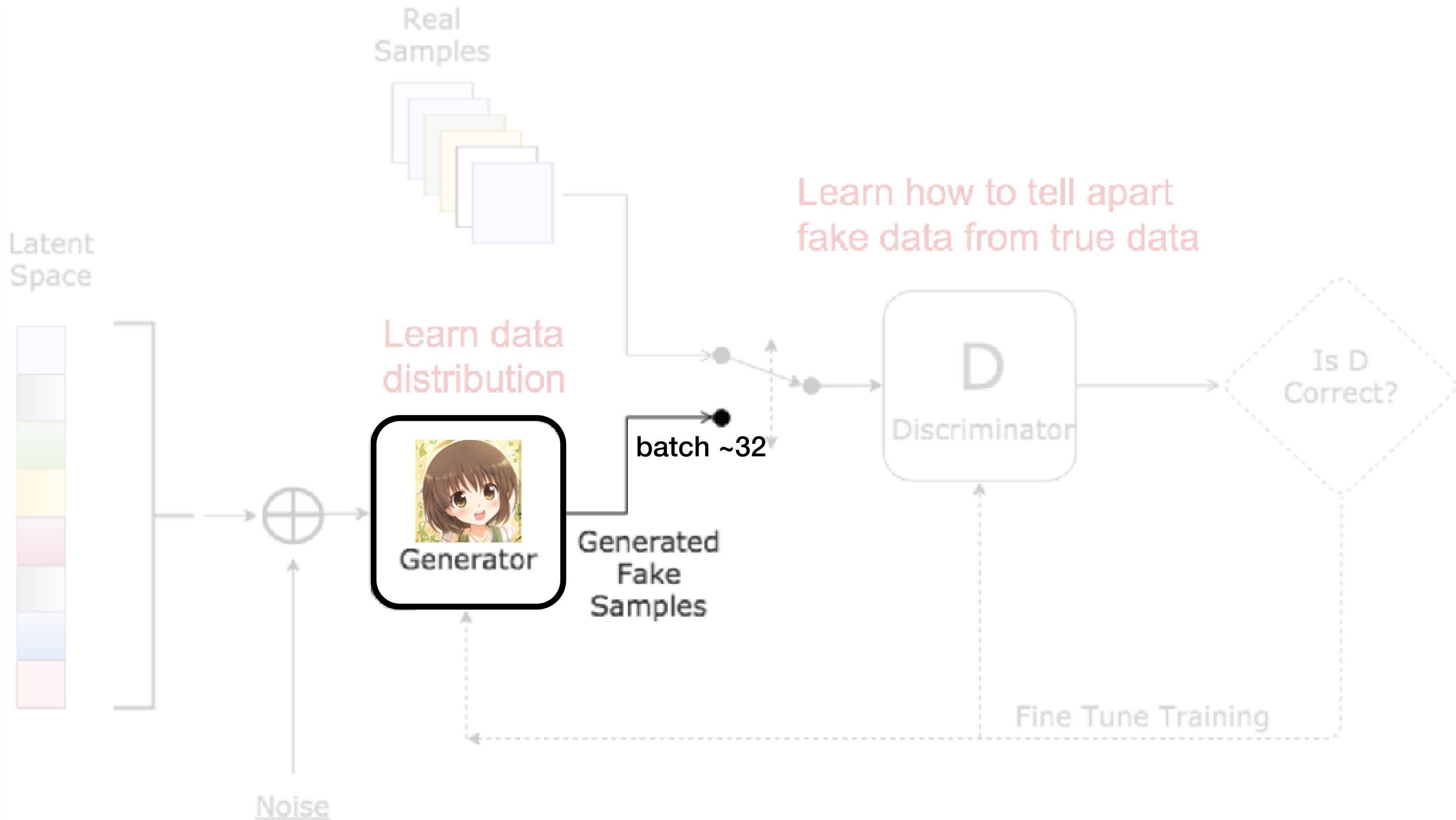
/Архитектура



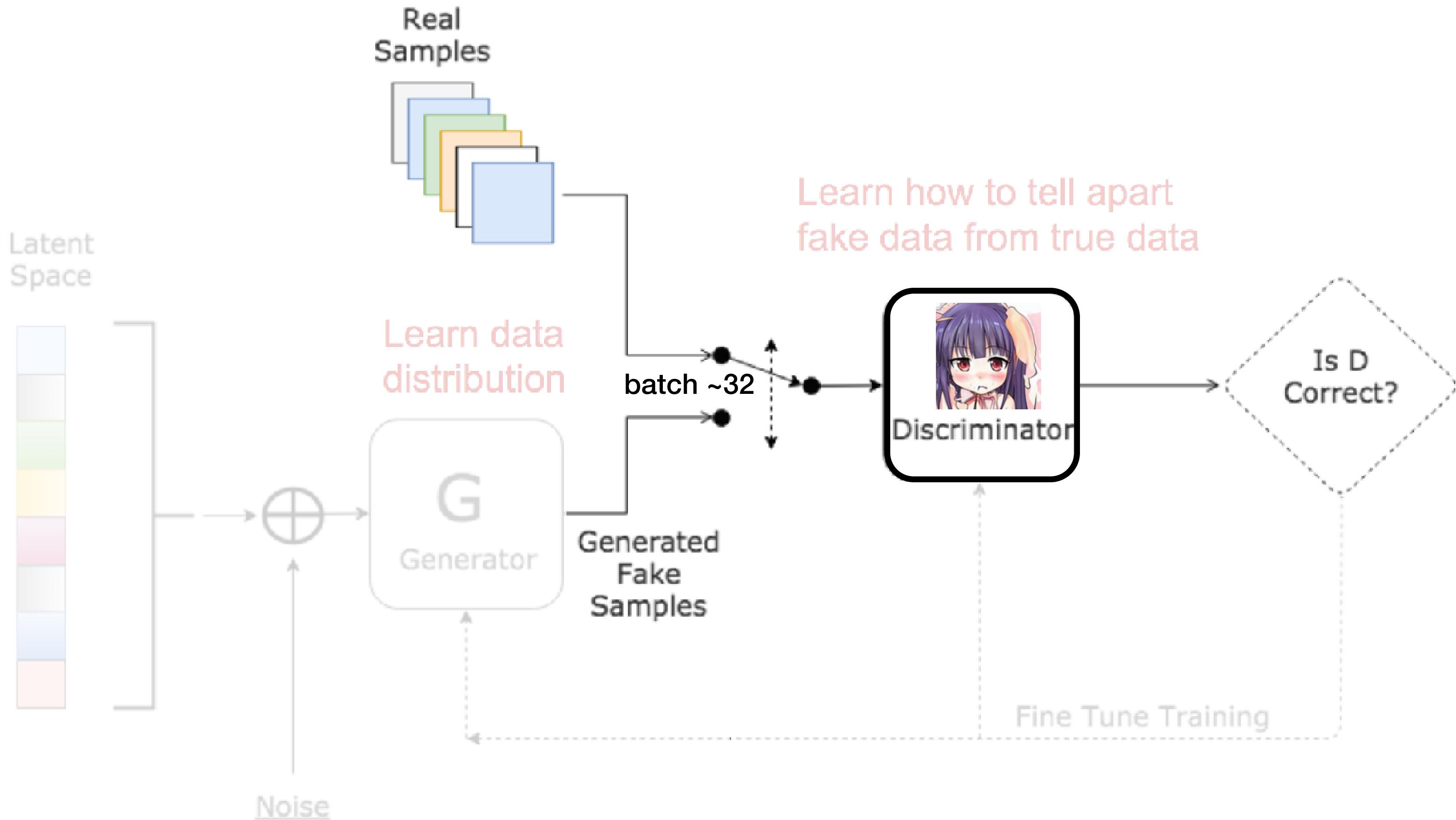
/Архитектура



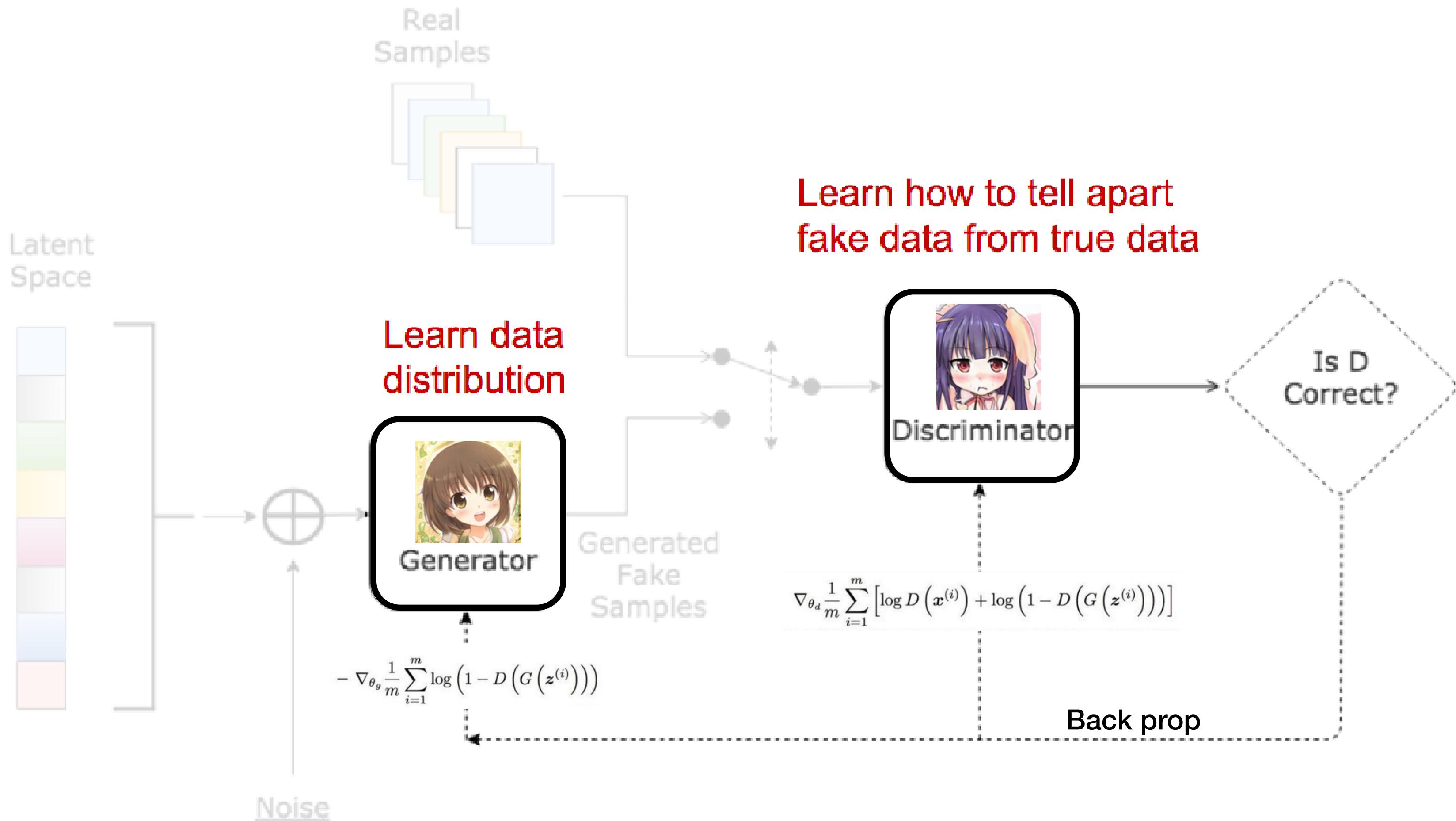
/Архитектура



/Архитектура



/Архитектура



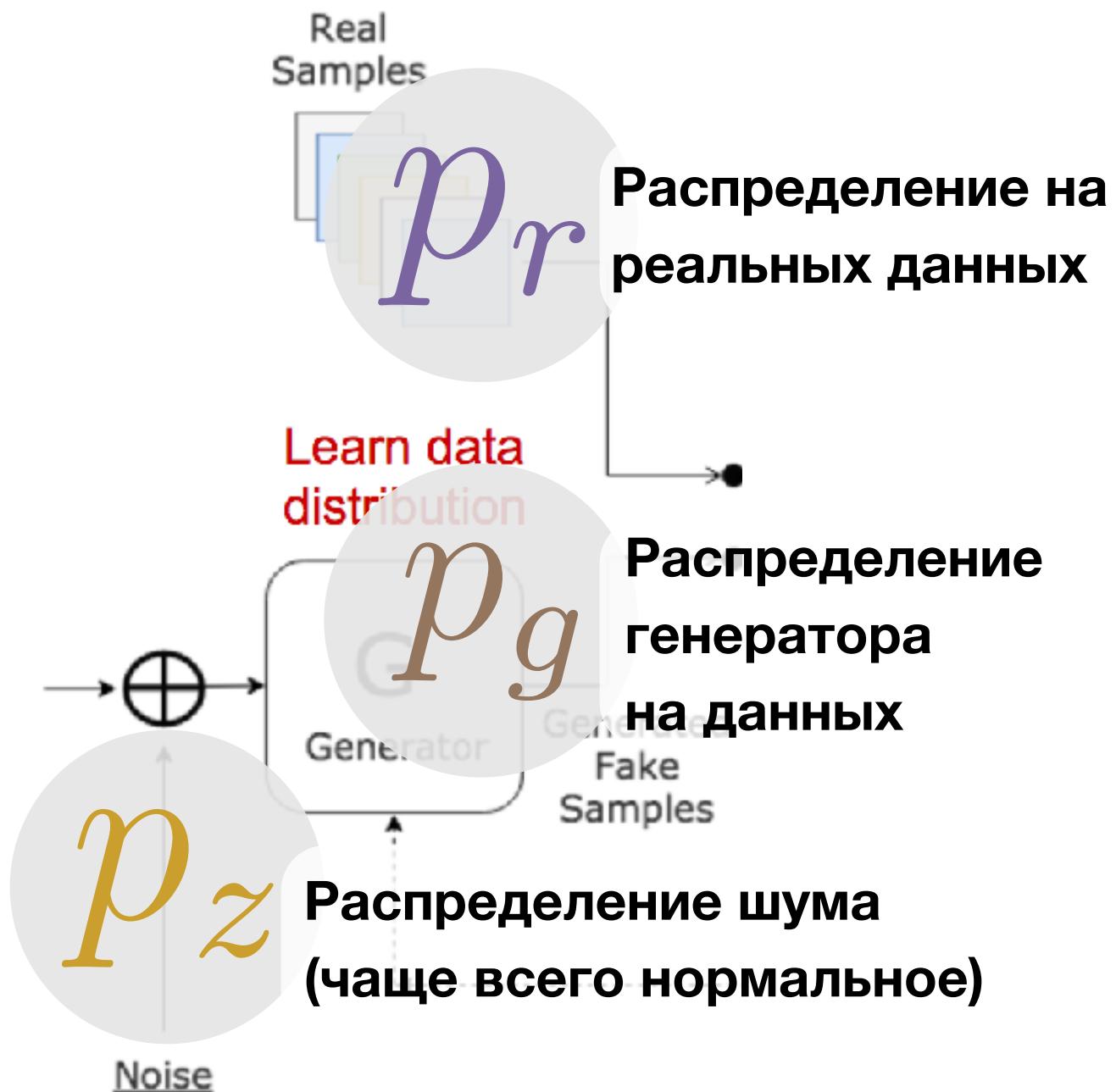
Знаем модели бинарной классификации

Знаем генеративные модели

Остается узнать Loss-функцию

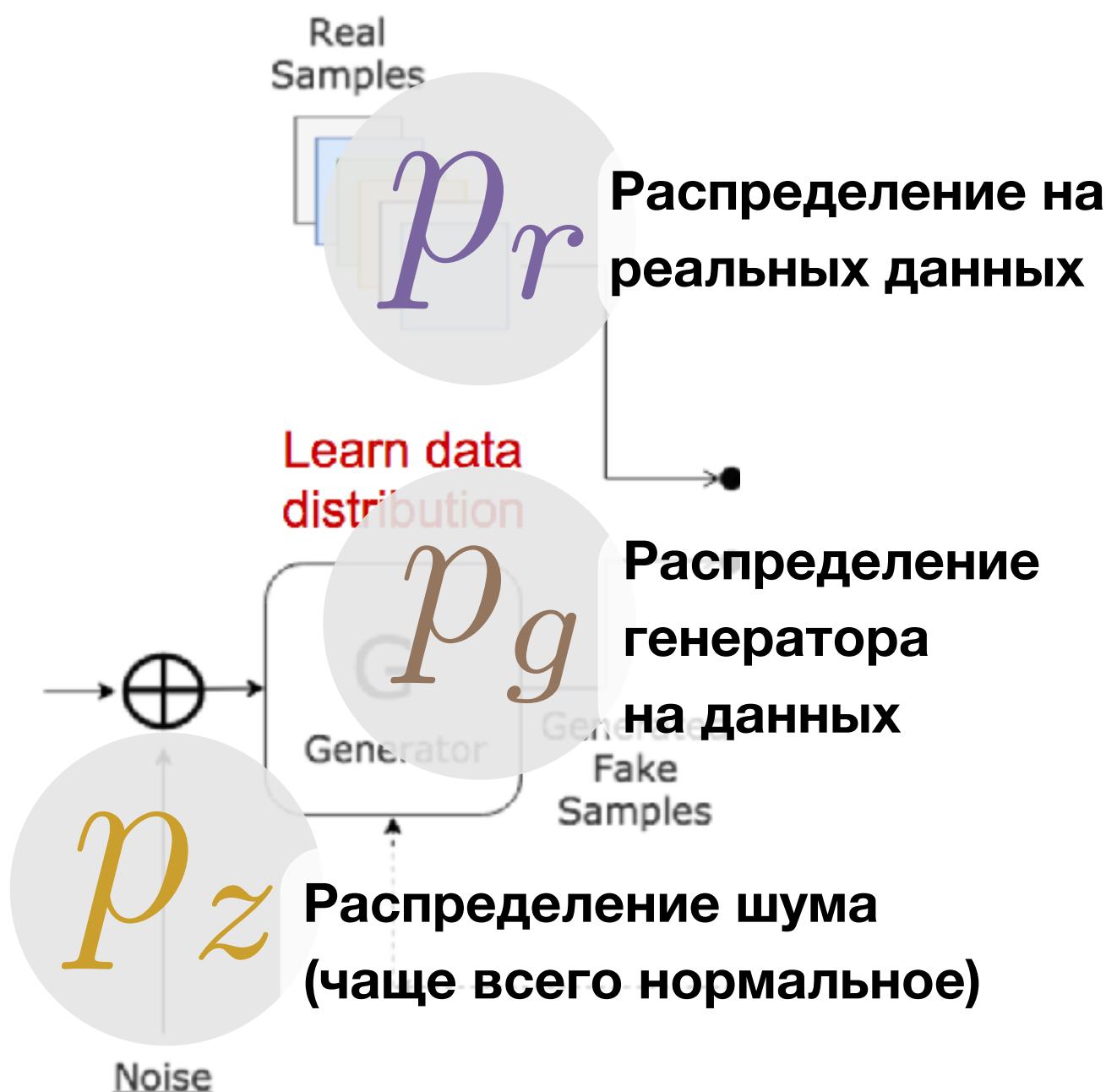
/Loss-функция

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

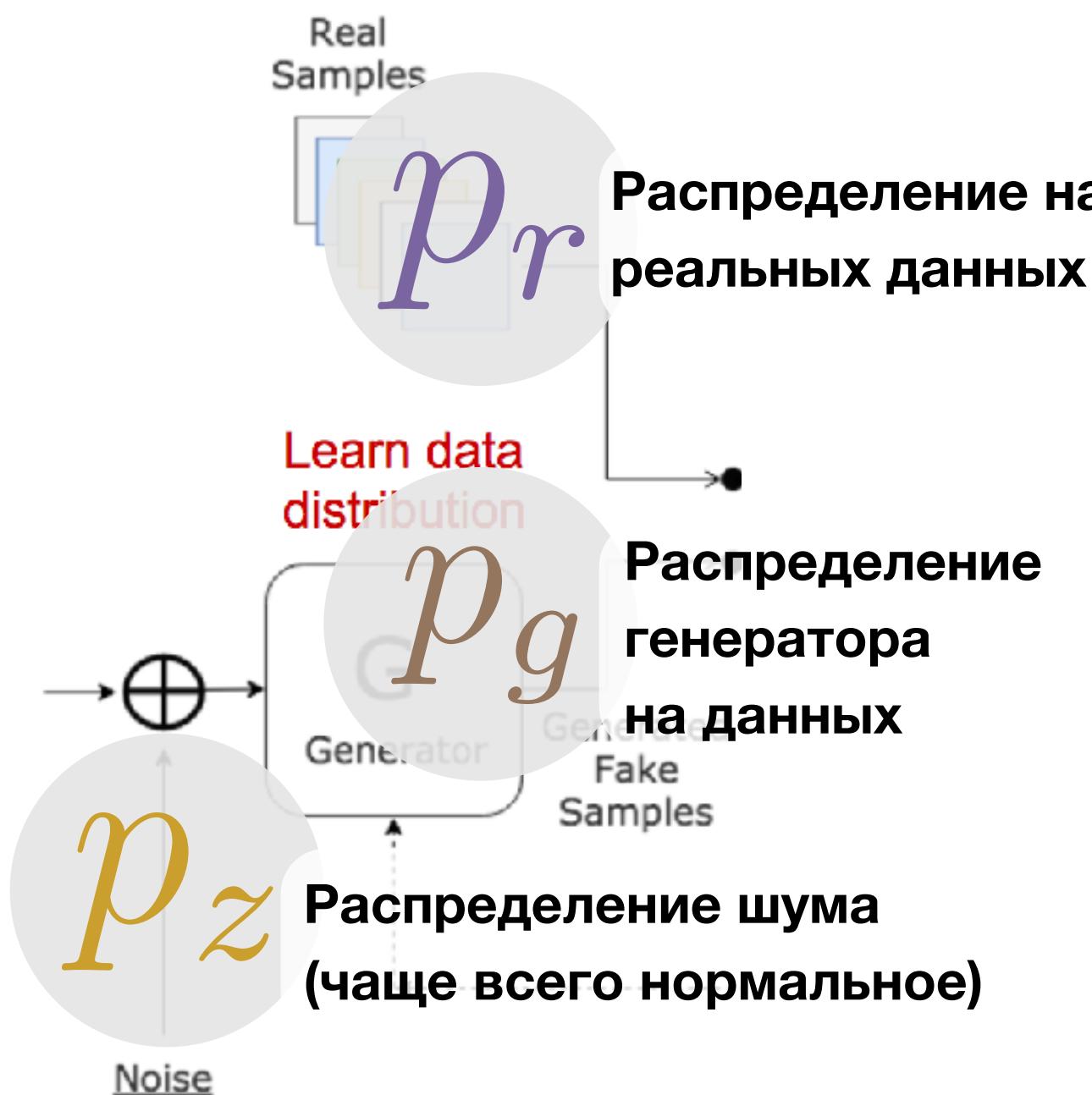


/Loss-функция

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



/Loss-функция



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

$$\mathbb{E}_{x \sim p_g(x)} [\log(1 - D(x))]$$

Никак не трогает G во время спуска по градиенту!

/Loss-функция/Оптимальное значение для D

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

Распишем матожидание через интеграл:

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

/Loss-функция/Оптимальное значение для D

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

Распишем матожидание через интеграл:

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

Интегрируем по всей области определения, $\tilde{x} = D(x)$,
поэтому можем убрать интеграл: $A = p_r(x)$,

$$f(\tilde{x}) = A \log \tilde{x} + B \log(1 - \tilde{x}) \quad B = p_g(x)$$

$$\frac{df(\tilde{x})}{d\tilde{x}} = A \frac{1}{\ln 10} \frac{1}{\tilde{x}} - B \frac{1}{\ln 10} \frac{1}{1 - \tilde{x}} = \frac{1}{\ln 10} \left(\frac{A}{\tilde{x}} - \frac{B}{1 - \tilde{x}} \right) = \frac{1}{\ln 10} \frac{A - (A + B)\tilde{x}}{\tilde{x}(1 - \tilde{x})}$$

Приравняв производную нулю сможем найти лучшее значение D:

$$D^*(x) = \tilde{x}^* = \frac{A}{A + B} = \frac{p_r(x)}{p_r(x) + p_g(x)} \in [0, 1]$$

/Loss-функция/Глобальный оптимум

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

$$p_g = p_r \Rightarrow D^*(x) = \frac{1}{2}$$

$$L(G^*, D^*) = \log \frac{1}{2} \int_x p_r(x) dx + \log \frac{1}{2} \int_x p_g(x) dx = -2 \log 2$$

/Loss-функция/Глобальный оптимум

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

$$p_g = p_r \Rightarrow D^*(x) = \frac{1}{2}$$

$$L(G^*, D^*) = \log \frac{1}{2} \int_x p_r(x) dx + \log \frac{1}{2} \int_x p_g(x) dx = -2 \log 2$$

$$L(G^*, D^*) = -2 \log 2$$

/Loss-функция/Оптимальное значение для D

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

Распишем матожидание через интеграл:

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

Интегрируем по всей области определения, $\tilde{x} = D(x)$,
поэтому можем убрать интеграл: $A = p_r(x)$,

$$f(\tilde{x}) = A \log \tilde{x} + B \log(1 - \tilde{x}) \quad B = p_g(x)$$

$$\frac{df(\tilde{x})}{d\tilde{x}} = A \frac{1}{\ln 10} \frac{1}{\tilde{x}} - B \frac{1}{\ln 10} \frac{1}{1 - \tilde{x}} = \frac{1}{\ln 10} \left(\frac{A}{\tilde{x}} - \frac{B}{1 - \tilde{x}} \right) = \frac{1}{\ln 10} \frac{A - (A + B)\tilde{x}}{\tilde{x}(1 - \tilde{x})}$$

Приравняв производную нулю сможем найти лучшее значение D:

$$D^*(x) = \tilde{x}^* = \frac{A}{A + B} = \frac{p_r(x)}{p_r(x) + p_g(x)} \in [0, 1]$$

Когда G натренирован до оптимума $p_g \approx p_r$

/Loss-функция/Откуда она взялась?

$$D_{JS}(p_r \| p_g) = \frac{1}{2} D_{KL}(p_r \| \frac{p_r + p_g}{2}) + \frac{1}{2} D_{KL}(p_g \| \frac{p_r + p_g}{2})$$

- по определению

/Loss-функция/Откуда она взялась?

$$D_{JS}(p_r \| p_g) = \frac{1}{2} D_{KL}(p_r || \frac{p_r + p_g}{2}) + \frac{1}{2} D_{KL}(p_g || \frac{p_r + p_g}{2}) \quad - \text{ по определению}$$

$$\frac{1}{2} \left(\log 2 + \int_x p_r(x) \log \frac{p_r(x)}{p_r + p_g(x)} dx \right) + \frac{1}{2} \left(\log 2 + \int_x p_g(x) \log \frac{p_g(x)}{p_r + p_g(x)} dx \right)$$

/Loss-функция/Откуда она взялась?

$$D_{JS}(p_r \| p_g) = \frac{1}{2} D_{KL}(p_r \| \frac{p_r + p_g}{2}) + \frac{1}{2} D_{KL}(p_g \| \frac{p_r + p_g}{2})$$

- по определению

$$\frac{1}{2} \left(\log 2 + \int_x p_r(x) \log \frac{p_r(x)}{p_r + p_g(x)} dx \right) + \frac{1}{2} \left(\log 2 + \int_x p_g(x) \log \frac{p_g(x)}{p_r + p_g(x)} dx \right)$$
$$\frac{1}{2} \left(2 \log 2 + L(G, D^*) \right)$$

Следовательно: $L(G, D^*) = 2D_{JS}(p_r \| p_g) - 2 \log 2$

/Loss-функция/Откуда она взялась?

$$D_{JS}(p_r \| p_g) = \frac{1}{2} D_{KL}(p_r \| \frac{p_r + p_g}{2}) + \frac{1}{2} D_{KL}(p_g \| \frac{p_r + p_g}{2})$$
$$\frac{1}{2} \left(\log 2 + \int_x p_r(x) \log \frac{p_r(x)}{p_r + p_g(x)} dx \right) + \frac{1}{2} \left(\log 2 + \int_x p_g(x) \log \frac{p_g(x)}{p_r + p_g(x)} dx \right)$$
$$\frac{1}{2} \left(2 \log 2 + L(G, D^*) \right)$$

Следовательно: $L(G, D^*) = 2D_{JS}(p_r \| p_g) - 2 \log 2$

Т.е. наша Loss-функция это расстояние Йенсена-Шеннона между распределениями реальных и сгенерированных данных. И как мы уже высказали, распределения совпадают, когда Loss-функция достигает значения $-2 \log 2$.

/Проблемы GAN/Отсутствие сходимости

Пример: $f_1(x) = xy \rightarrow \min_x$

$f_2(y) = -xy \rightarrow \min_y$

/Проблемы GAN/Отсутствие сходимости

Пример: $f_1(x) = xy \rightarrow \min_x$

$f_2(y) = -xy \rightarrow \min_y$

$$\begin{aligned} \frac{\partial f_1}{\partial x} &= y && \text{learning rate} \\ \frac{\partial f_2}{\partial y} &= -x && \Rightarrow \quad \begin{aligned} x &= x - \eta \cdot y \\ y &= y + \eta \cdot x \end{aligned} \end{aligned}$$

/Проблемы GAN/Отсутствие сходимости

Пример: $f_1(x) = xy \rightarrow \min_x$

$f_2(y) = -xy \rightarrow \min_y$

WGAN

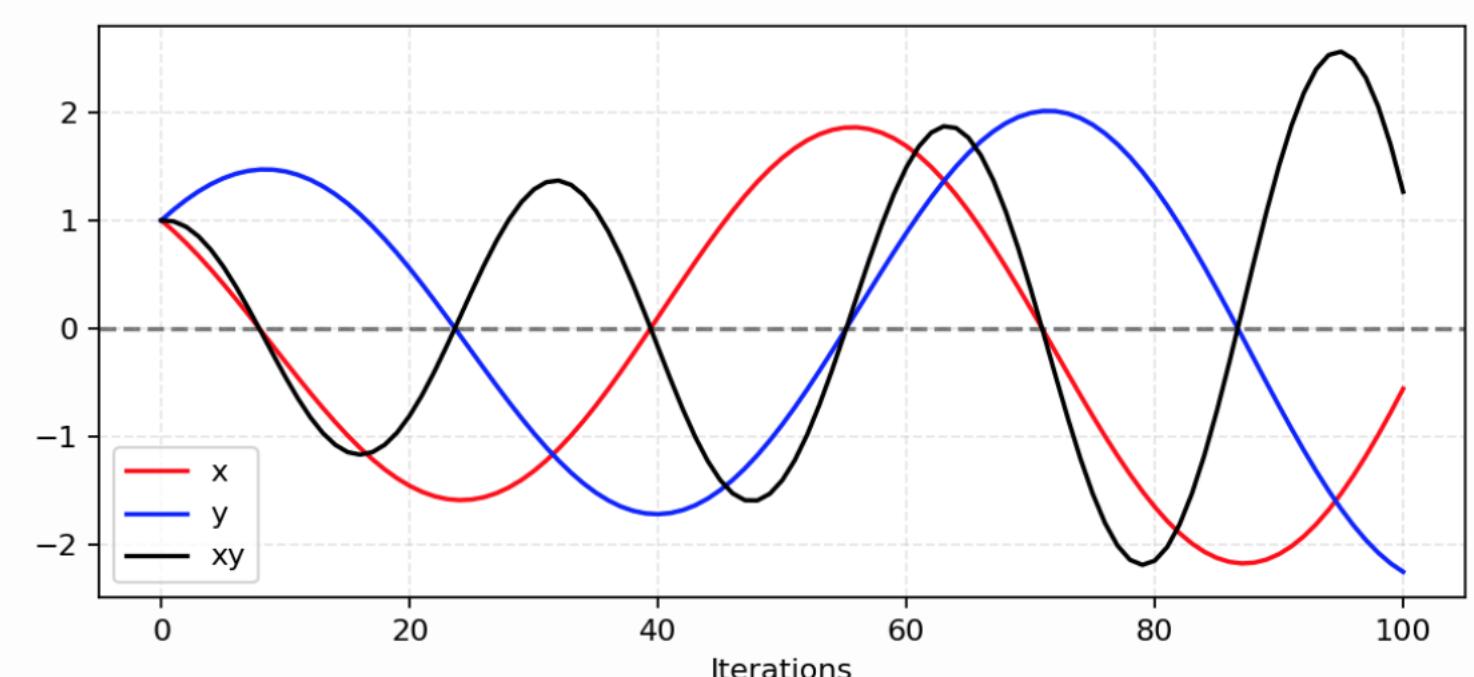
$$\begin{aligned} \frac{\partial f_1}{\partial x} &= y \\ \frac{\partial f_2}{\partial y} &= -x \end{aligned} \Rightarrow \begin{array}{c} x = x - \eta \cdot y \\ y = y + \eta \cdot x \end{array}$$

learning rate
↓

Как только x и y становятся противоположных знаков, с каждым последующим шагом градиентного спуска колебания становятся все больше и больше:

Подробнее:

T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans,” in Advances in neural information processing systems, 2016, pp. 2234–2242.



/Проблемы GAN/Исчезающий градиент

Пусть мы имеем идеальную модель D, тогда:

$$\begin{aligned}D(x) &= 1, \forall x \in p_r \\D(x) &= 0, \forall x \in p_g\end{aligned}$$

/Проблемы GAN/Исчезающий градиент

Пусть мы имеем идеальную модель D, тогда:

$$\begin{aligned}D(x) &= 1, \forall x \in p_r \\D(x) &= 0, \forall x \in p_g\end{aligned}$$

Но из этого следует, что Loss-функция зануляется

Значит, нет градиента

/Проблемы GAN/Исчезающий градиент

Пусть мы имеем идеальную модель D , тогда:

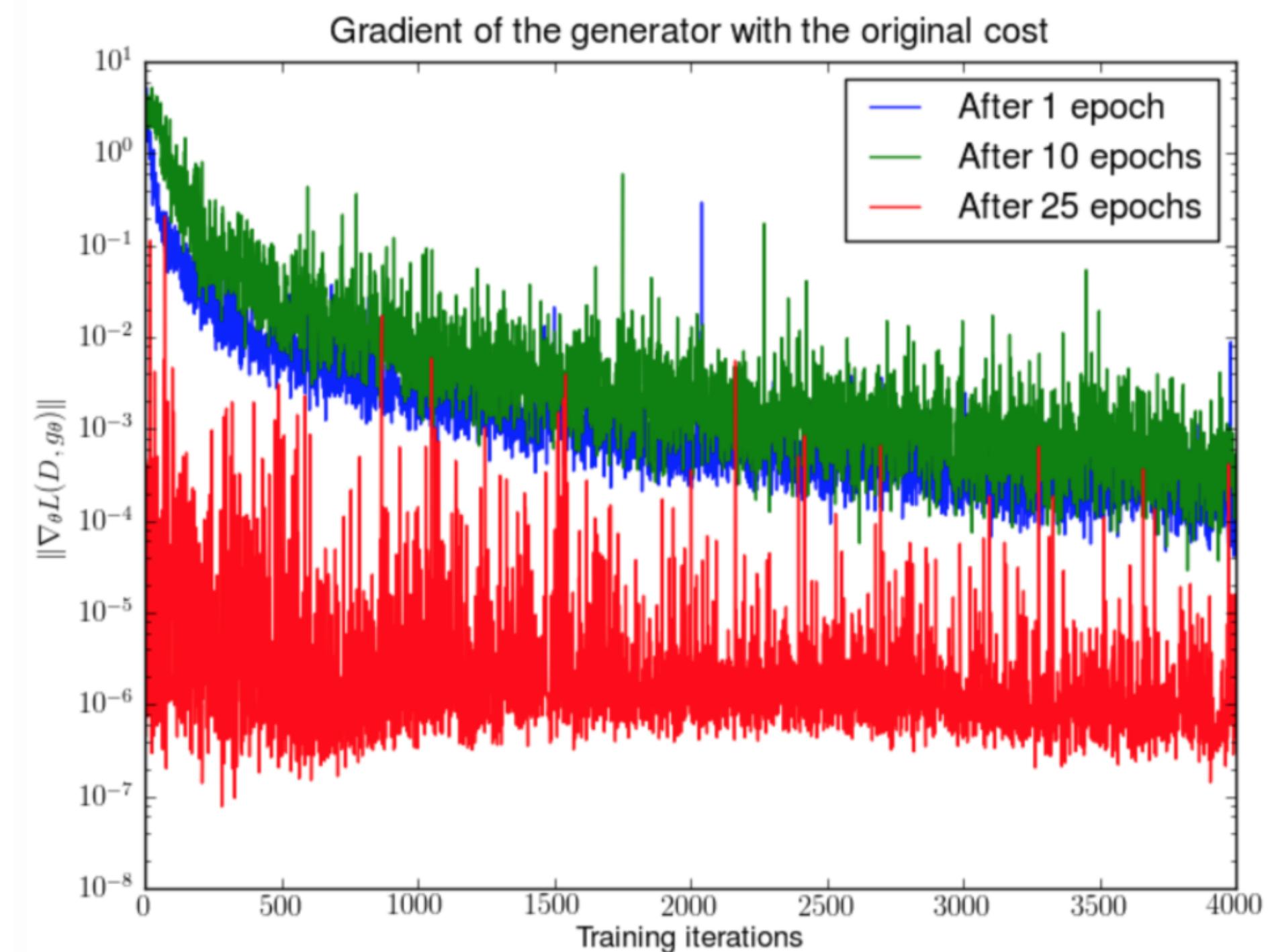
$$D(x) = 1, \forall x \in p_r$$
$$D(x) = 0, \forall x \in p_g$$

Но из этого следует, что Loss-функция зануляется

Значит, нет градиента

Пример того, как из-за быстрой
скорости обучения
дискриминатора, градиент
стремительно затухает.

На картинке представлен
эксперимент с обучением
дискриминатора с нуля при
фиксированном генераторе.



/Проблемы GAN/Исчезающий градиент

При обучении GAN возникает следующая дилемма:

- Если дискриминатор предсказывает плохо, то loss-функция не отражает реальность и генератор не сможет выдать хорошие результаты
- Если дискриминатор предсказывает хорошо, то градиент loss-функции зануляется и обучение сильно замедляется, либо же совсем останавливается

Именно из-за этого очень сложно обучать GANы!



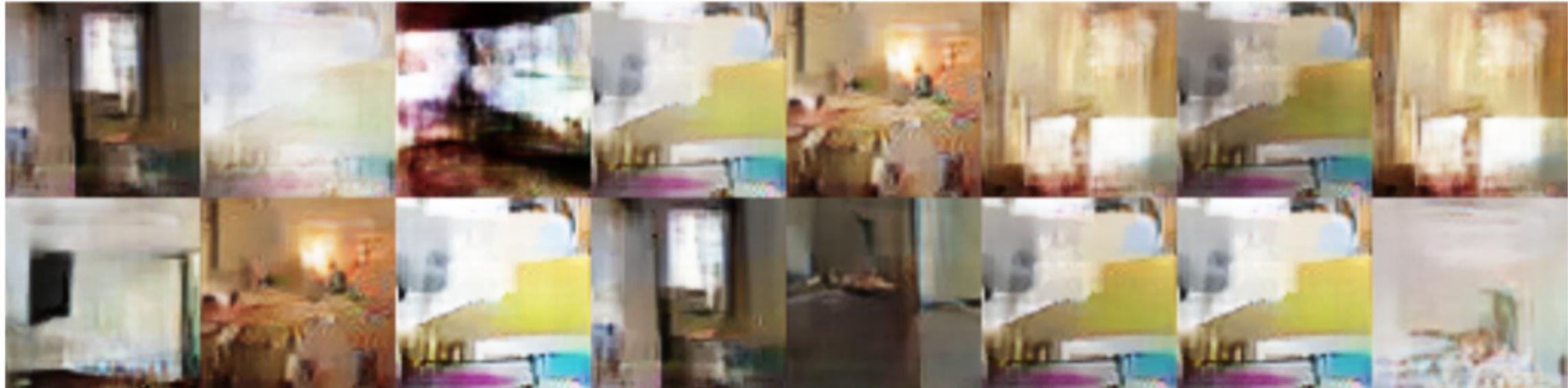
Подробнее:

Arjovsky, Martin, and Léon Bottou. "Towards principled methods for training generative adversarial networks." arXiv preprint arXiv:1701.04862 (2017).

/Проблемы GAN/Mode Collapse

Генератор начинает выдавать одинаковые объекты

WGAN



При этом он может эффективно обманывать дискриминатор, но выданные им объекты не будут отображать реального распределения (будет учитываться лишь какая-то маленькая их вариация)

Подробнее:

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein gan.
arXiv preprint arXiv:1701.07875.

/Проблемы GAN/Метрика качества

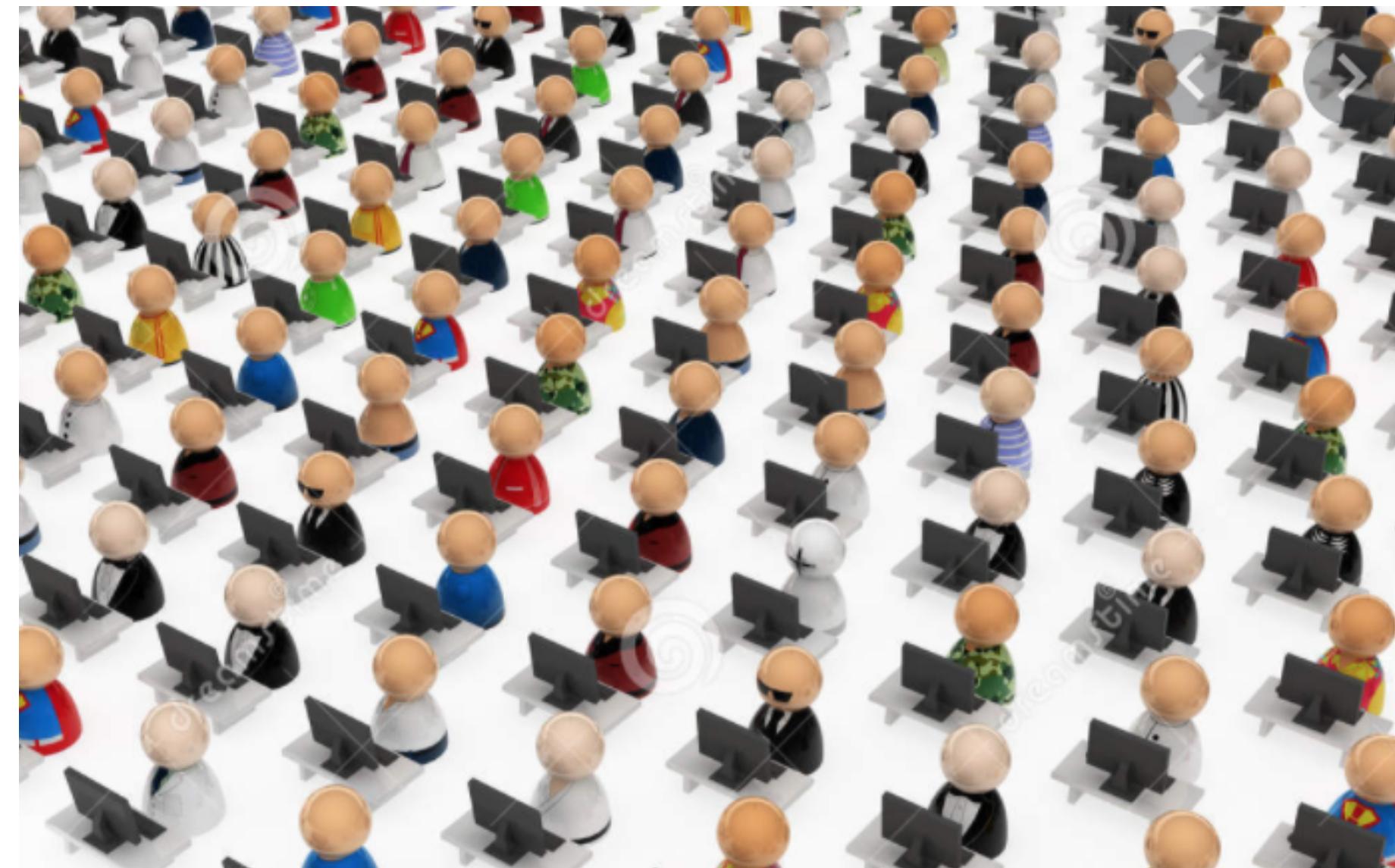
... а еще для GANов нет адекватной метрики качества.

- Когда останавливаться?
- Как сравнивать различные модели?

/Проблемы GAN/Метрика качества

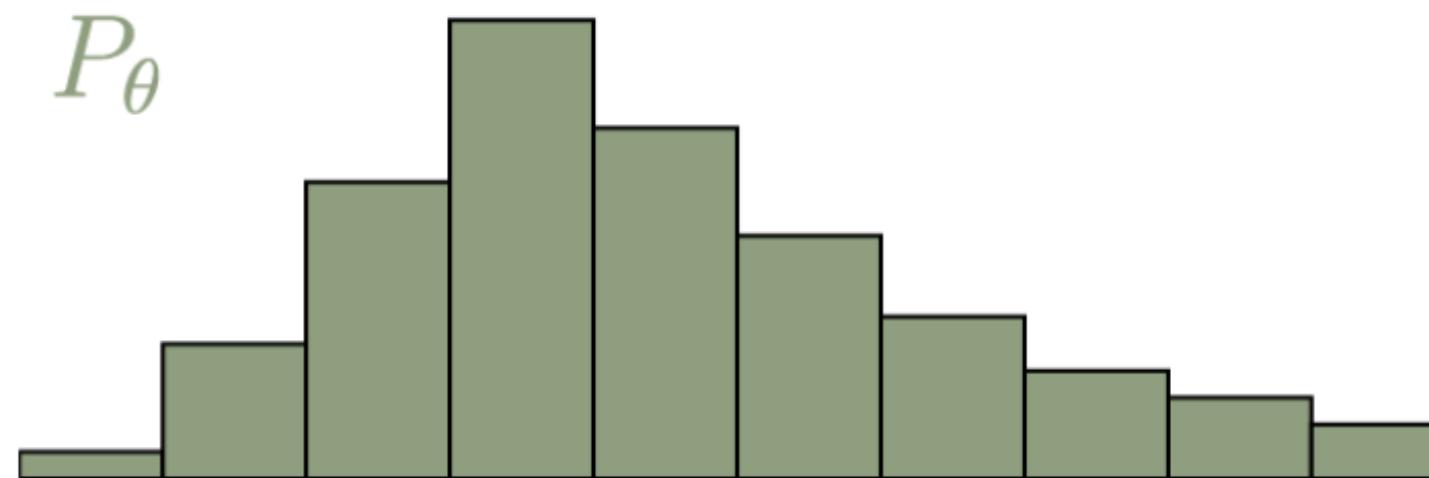
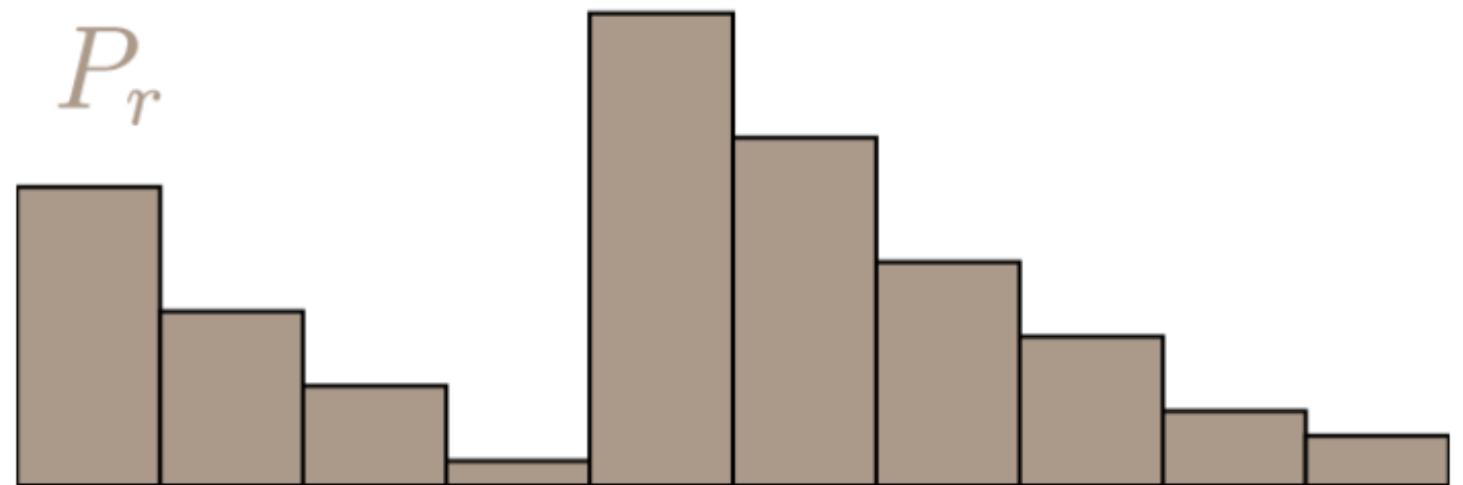
... а еще для GANов нет адекватной метрики качества.

- Когда останавливаться?
- Как сравнивать различные модели?

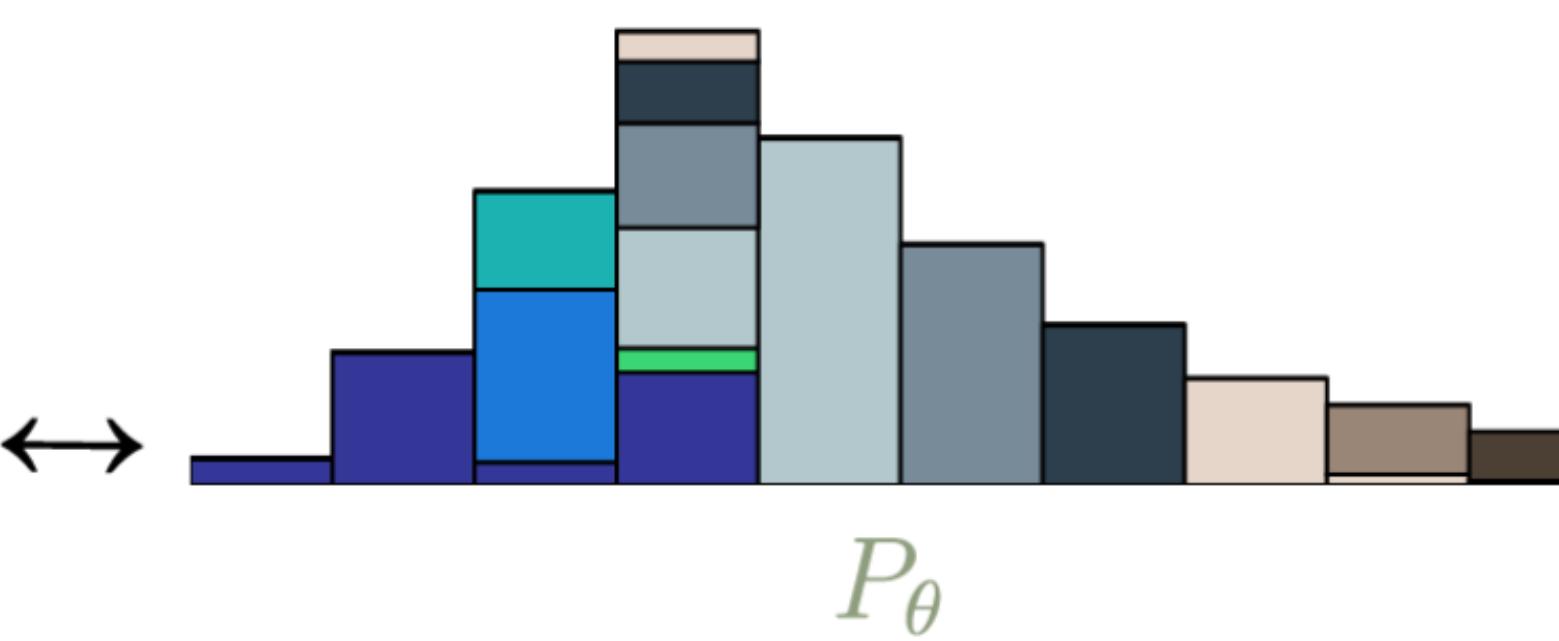
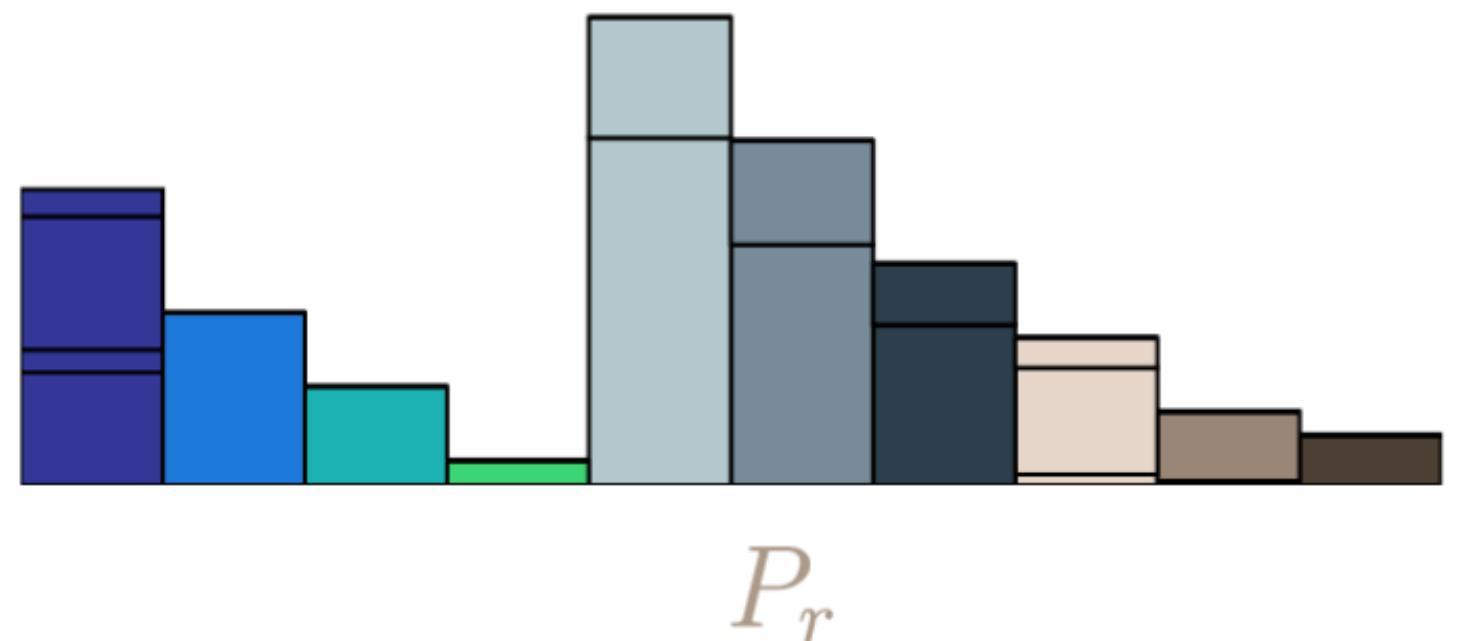


/Расстояние Вассерштейна

Earth Mover's Distance



Минимальная стоимость, сделать из одного дискретного распределения другое. Прямо пропорционально количеству перемещаемых кусочков и расстояния, на которое эти кусочки перемещают.



/Расстояние Вассерштейна

Расстояние Вассерштейна:

$$W(p_r, p_g) = \inf_{\gamma \sim \Pi(p_r, p_g)} \mathbb{E}_{(x,y) \sim \gamma} \|x - y\|$$

↑
множество всех возможных совместных
вероятностных распределений между p_r и p_g

/Расстояние Вассерштейна

Расстояние Вассерштейна:

$$W(p_r, p_g) = \inf_{\gamma \sim \Pi(p_r, p_g)} \mathbb{E}_{(x,y) \sim \gamma} \|x - y\|$$

↑
множество всех возможных совместных
вероятностных распределений между p_r и p_g

Вычислить можно по формуле, основанной на двойственности
Канторовича-Рубинштейна:

$$W(p_r, p_g) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{x \sim p_r} [f(x)] - \mathbb{E}_{x \sim p_g} [f(x)]$$

/Расстояние Вассерштейна

Расстояние Вассерштейна:

$$W(p_r, p_g) = \inf_{\gamma \sim \Pi(p_r, p_g)} \mathbb{E}_{(x,y) \sim \gamma} \|x - y\|$$

↑
множество всех возможных совместных
вероятностных распределений между p_r и p_g

Вычислить можно по формуле, основанной на двойственности
Канторовича-Рубинштейна:

$$W(p_r, p_g) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{x \sim p_r}[f(x)] - \mathbb{E}_{x \sim p_g}[f(x)]$$

↑

функция должна иметь константу Липшица равную K
(должна увеличивать расстояния не более, чем в K раз) $\forall x_1, x_2 \in \mathbb{R} : |f(x_1) - f(x_2)| \leq K|x_1 - x_2|$

/Wasserstein GAN

Положим, функция f , параметризуемая w ,
имеет константу Липшица K : $\{f_w\}_{w \in W}$

Во WGAN, в отличие от GAN, дискриминатор обучается на w , чтобы найти подходящее f_w

$$L(p_r, p_g) = W(p_r, p_g) = \max_{w \in W} \mathbb{E}_{x \sim p_r}[f_w(x)] - \mathbb{E}_{z \sim p_r(z)}[f_w(g_\theta(z))]$$

Таким образом, дискриминатор больше не является прямым критиком, отсеивающим поддельные образцы от настоящих. Вместо этого он обучается использовать Липшицевы отображения, чтобы вычислить расстояние Вассерштейна.

/Wasserstein GAN

Положим, функция f , параметризуемая w ,
имеет константу Липшица K : $\{f_w\}_{w \in W}$

Во WGAN, в отличие от GAN, дискриминатор обучается на w , чтобы найти подходящее f_w

$$L(p_r, p_g) = W(p_r, p_g) = \max_{w \in W} \mathbb{E}_{x \sim p_r}[f_w(x)] - \mathbb{E}_{z \sim p_r(z)}[f_w(g_\theta(z))]$$

Таким образом, дискриминатор больше не является прямым критиком, отсеивающим поддельные образцы от настоящих. Вместо этого он обучается использовать отображение Липшица, чтобы вычислить расстояние Вассерштейна.

Как поддерживать константу Липшица по мере обучения?

- После каждого обновления градиента ограничить веса w в маленьком промежутке, например $[-0.01, 0.01]$

/Wasserstein GAN/Отличия от классического GAN

- 1) После каждого шага спуска по градиенту веса зажимаются в маленьком фиксированном промежутке $[-c, c]$.
- 2) Дискриминатор больше не отсеивает поддельные образцы от настоящих напрямую, а оценивает расстояние Вассерштейна между реальным и генерируемым распределением данных. В новой функции потерь отсутствует логарифм.
- 3) На практике рекомендуется использовать RMSProp в качестве оптимизатора для дискриминатора, нежели классический Adam, с которым модель может обучаться нестабильно.

/Wasserstein GAN/Недостатки

- 1) Неустойчивое обучение
- 2) Медленная сходимость после отсечения по вему (когда границы были выбраны слишком большими)
- 3) Обращающиеся в ноль градиенты (когда границы были выбраны слишком маленькими)

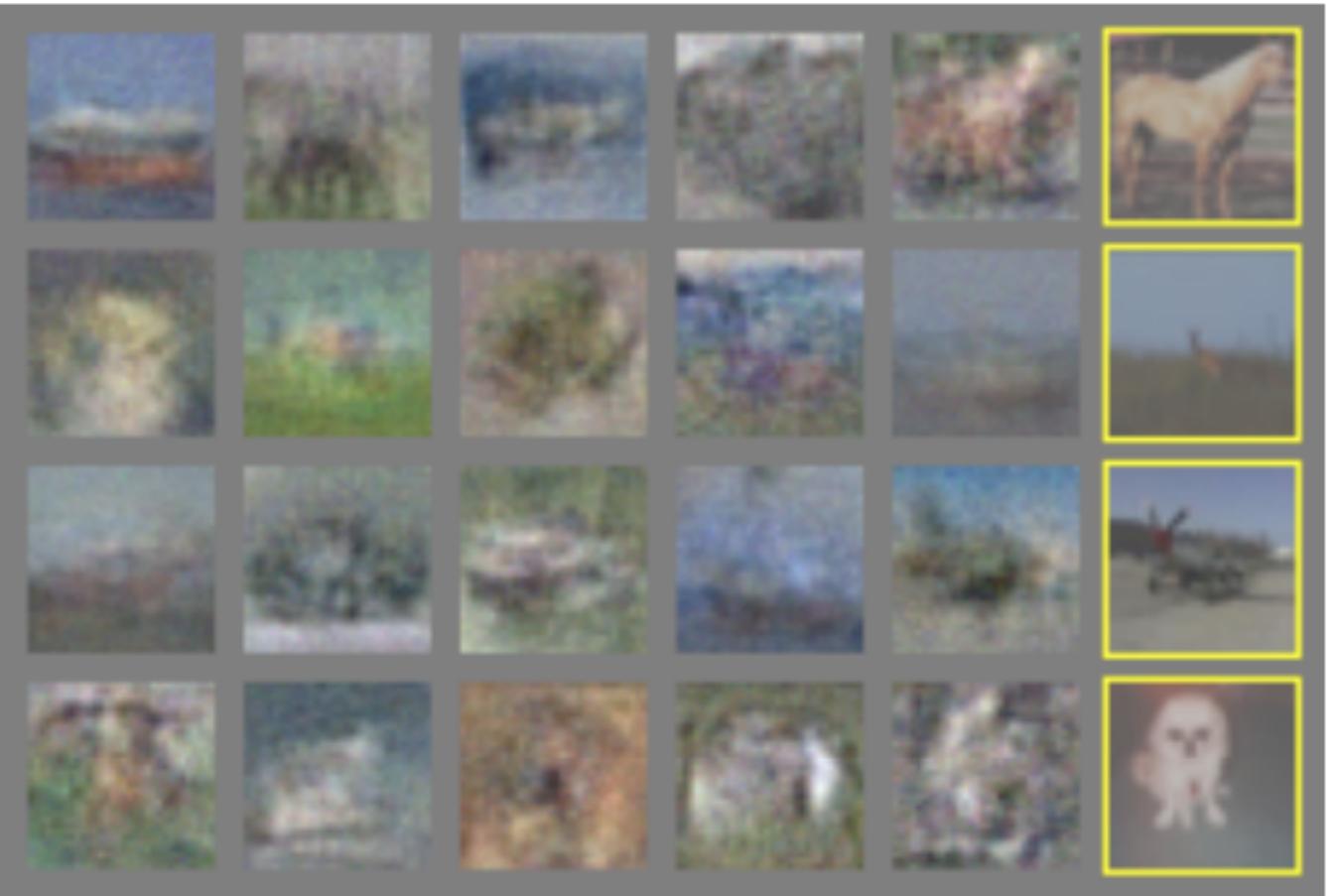
/Примеры использования



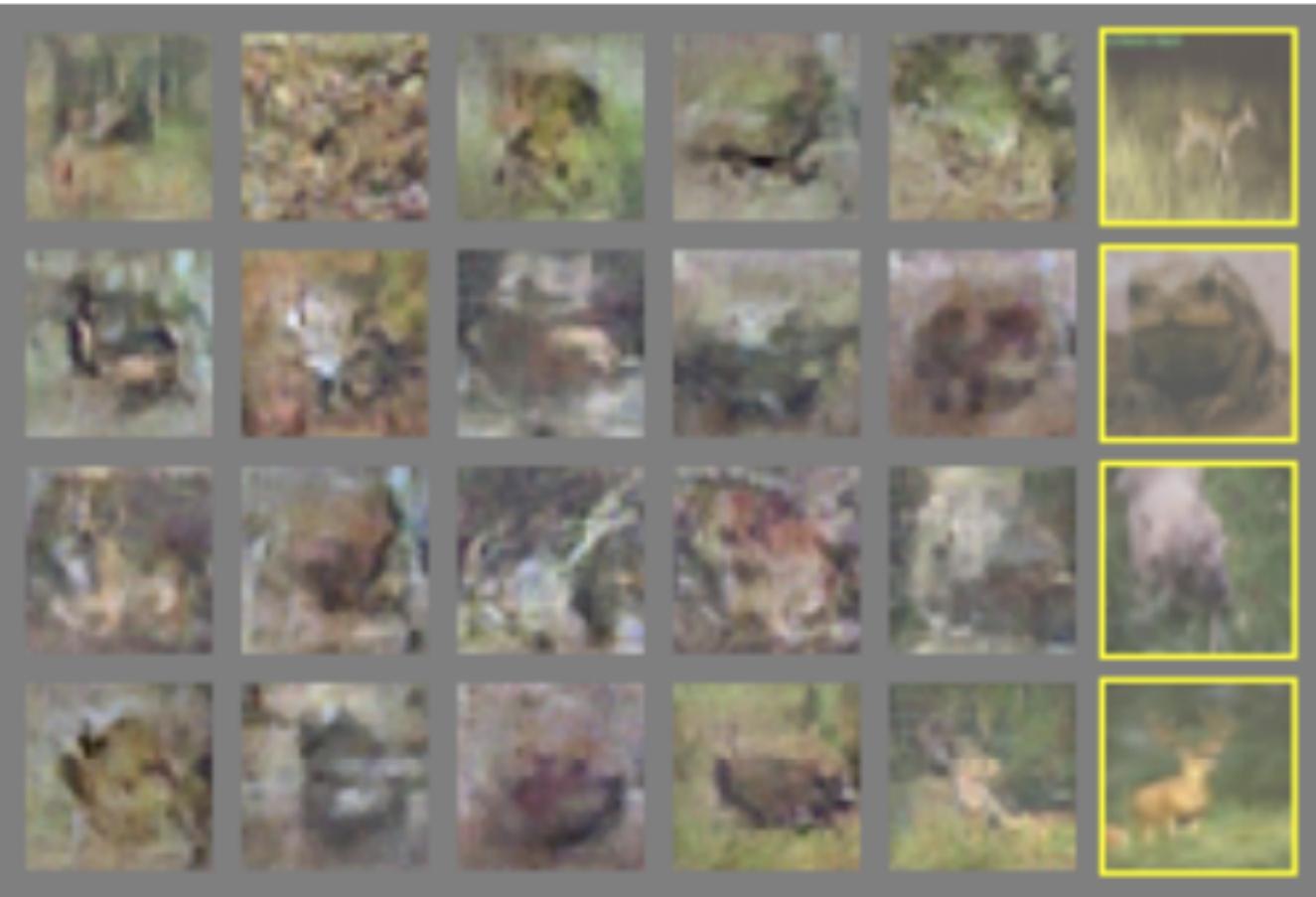
a)



b)



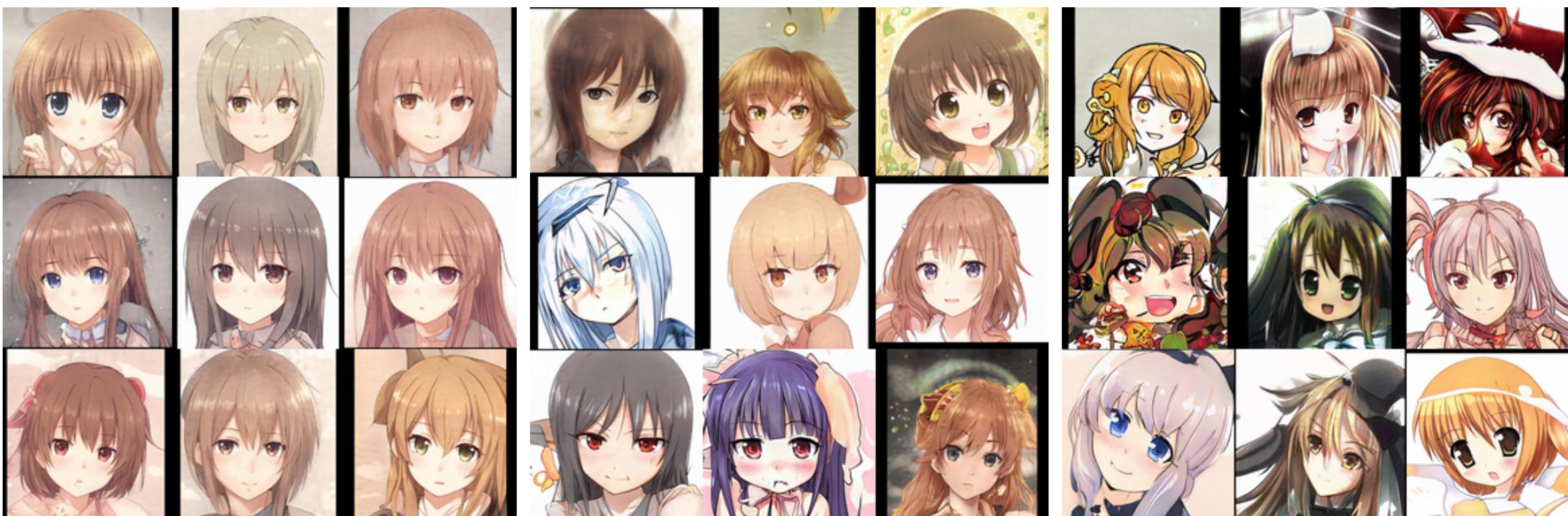
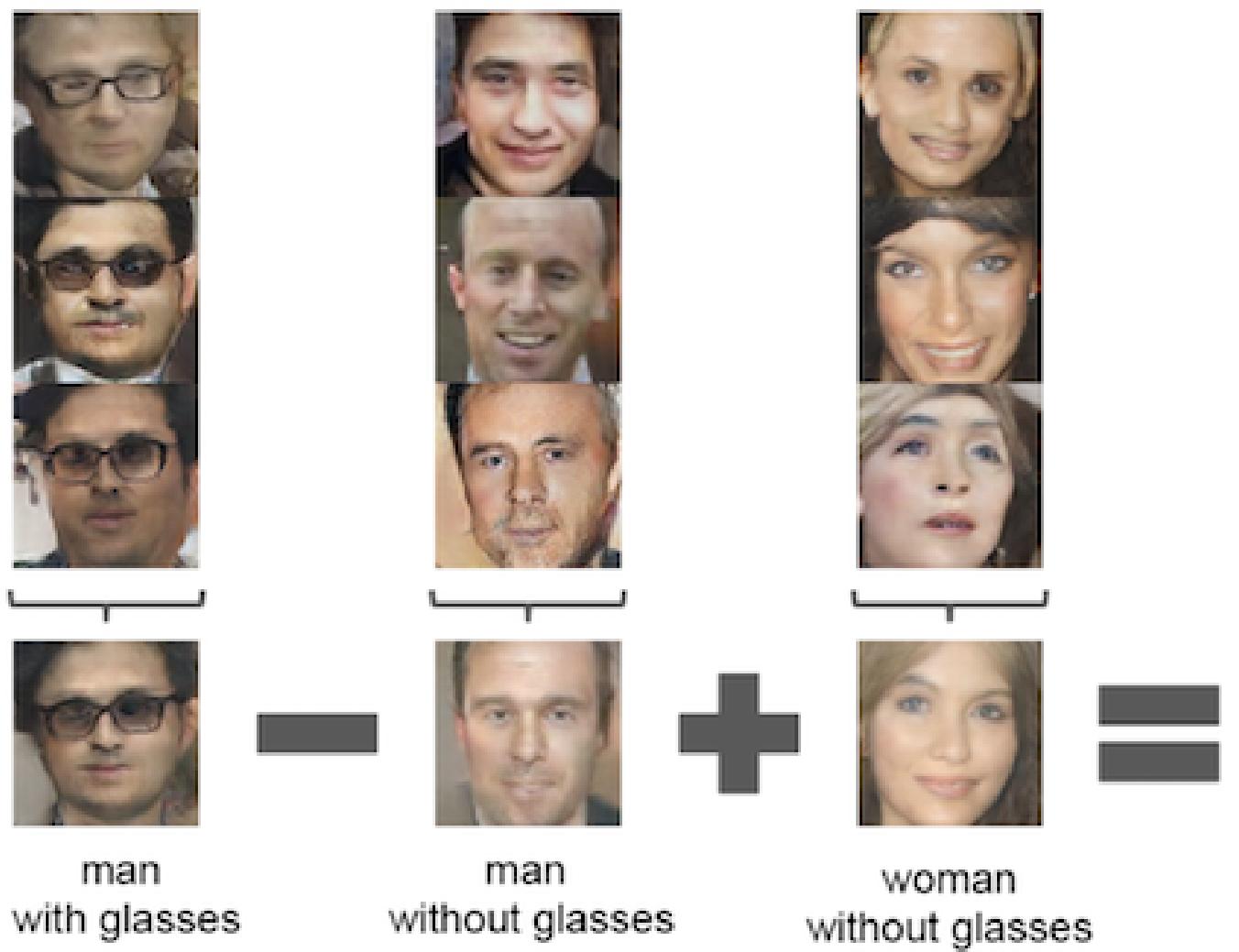
c)



d)

Генерация дополнительных изображений
для датасетов

/Примеры использования

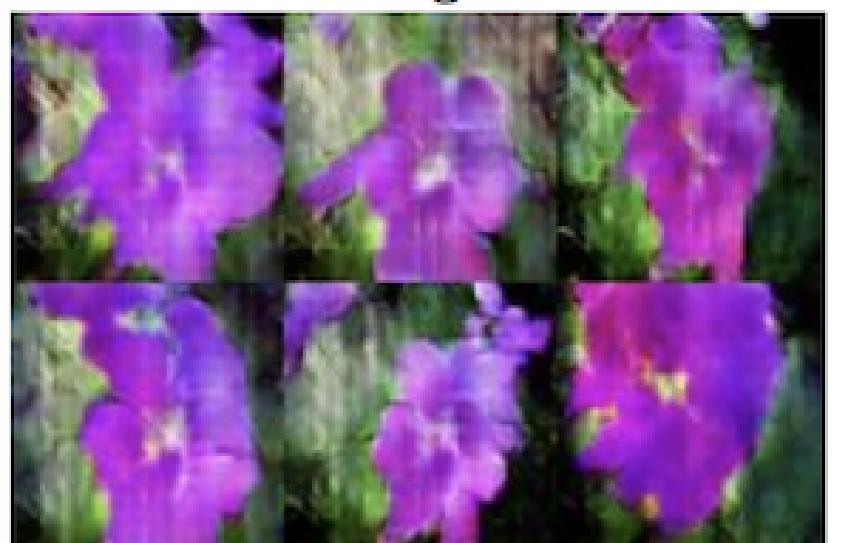


/Примеры использования

this small bird has a pink breast and crown, and black primaries and secondaries.



the flower has petals that are bright pinkish purple with white stigma



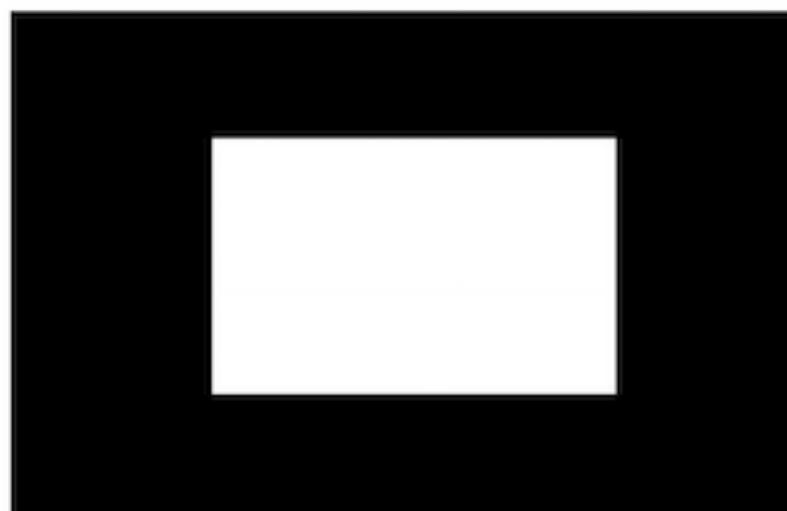
this magnificent fellow is almost all black with a red crest, and white cheek patch.



(a)



(b)



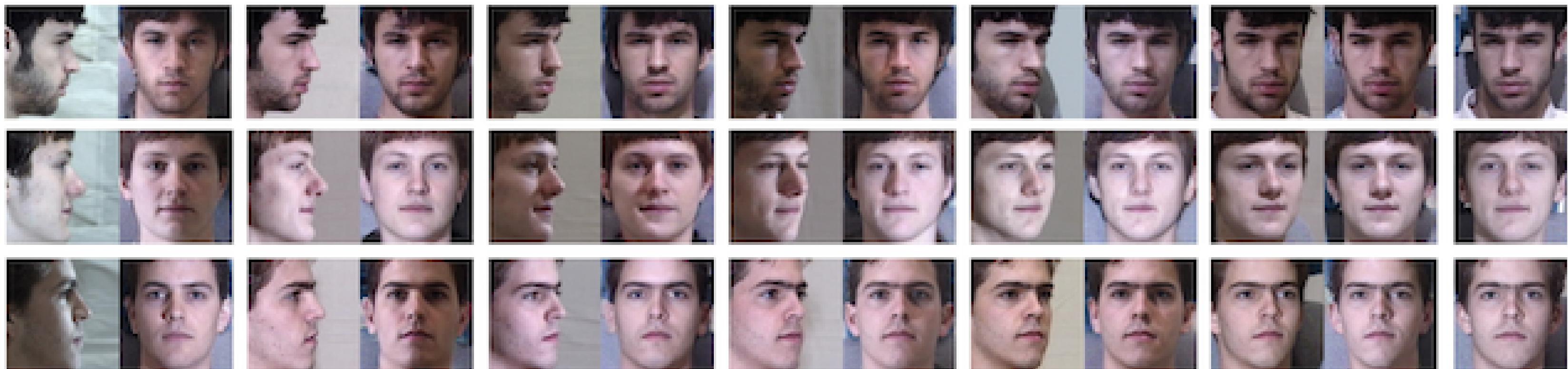
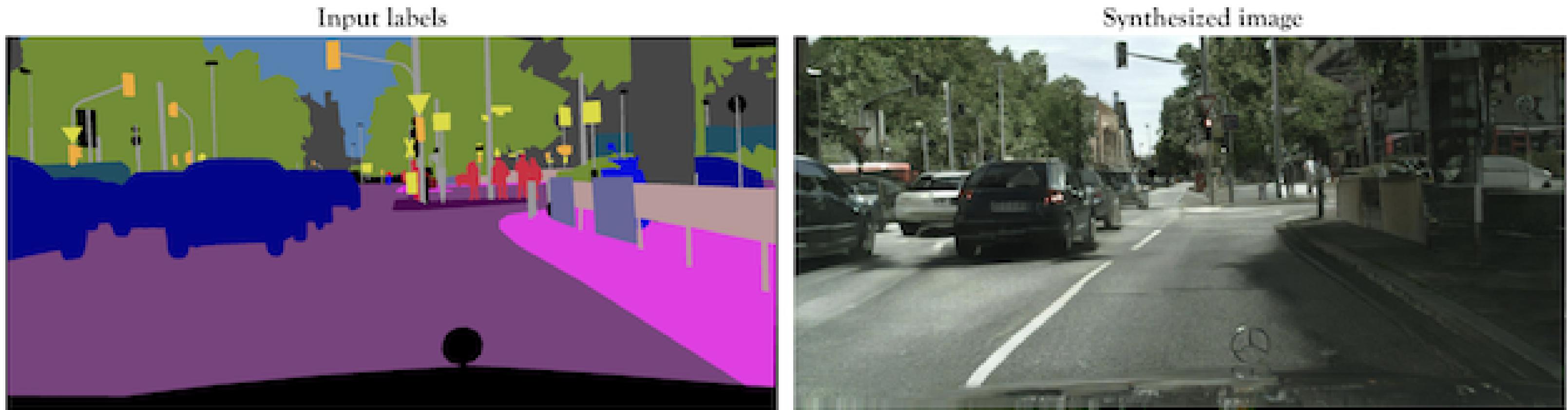
(c)



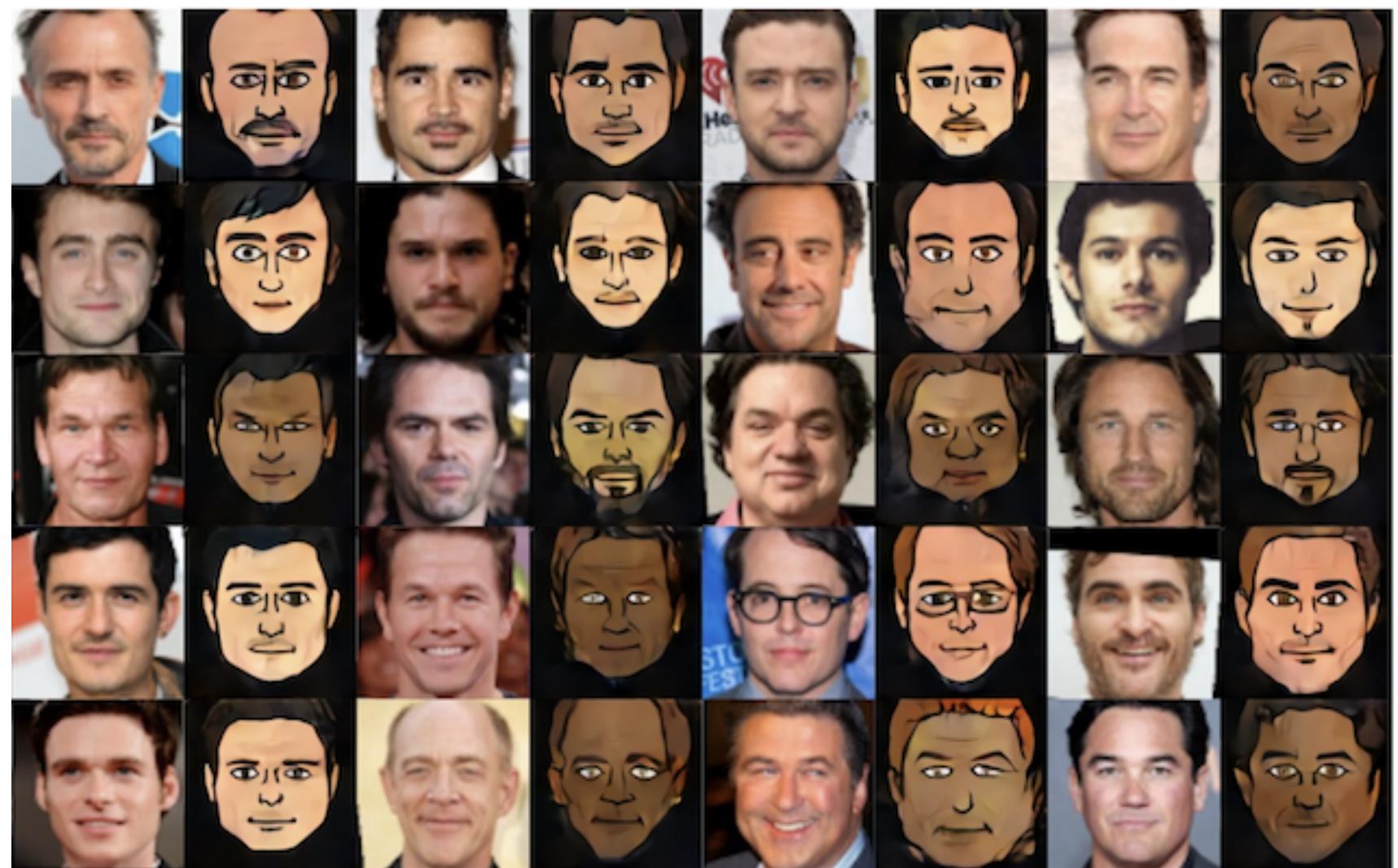
(d)



/Примеры использования



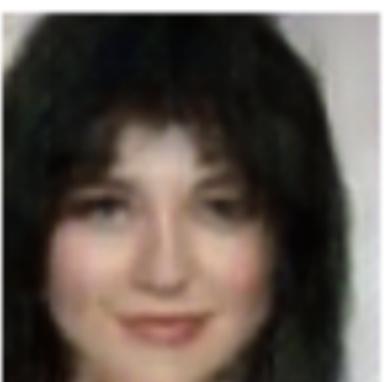
/Примеры использования



Real image



Reconstructed images



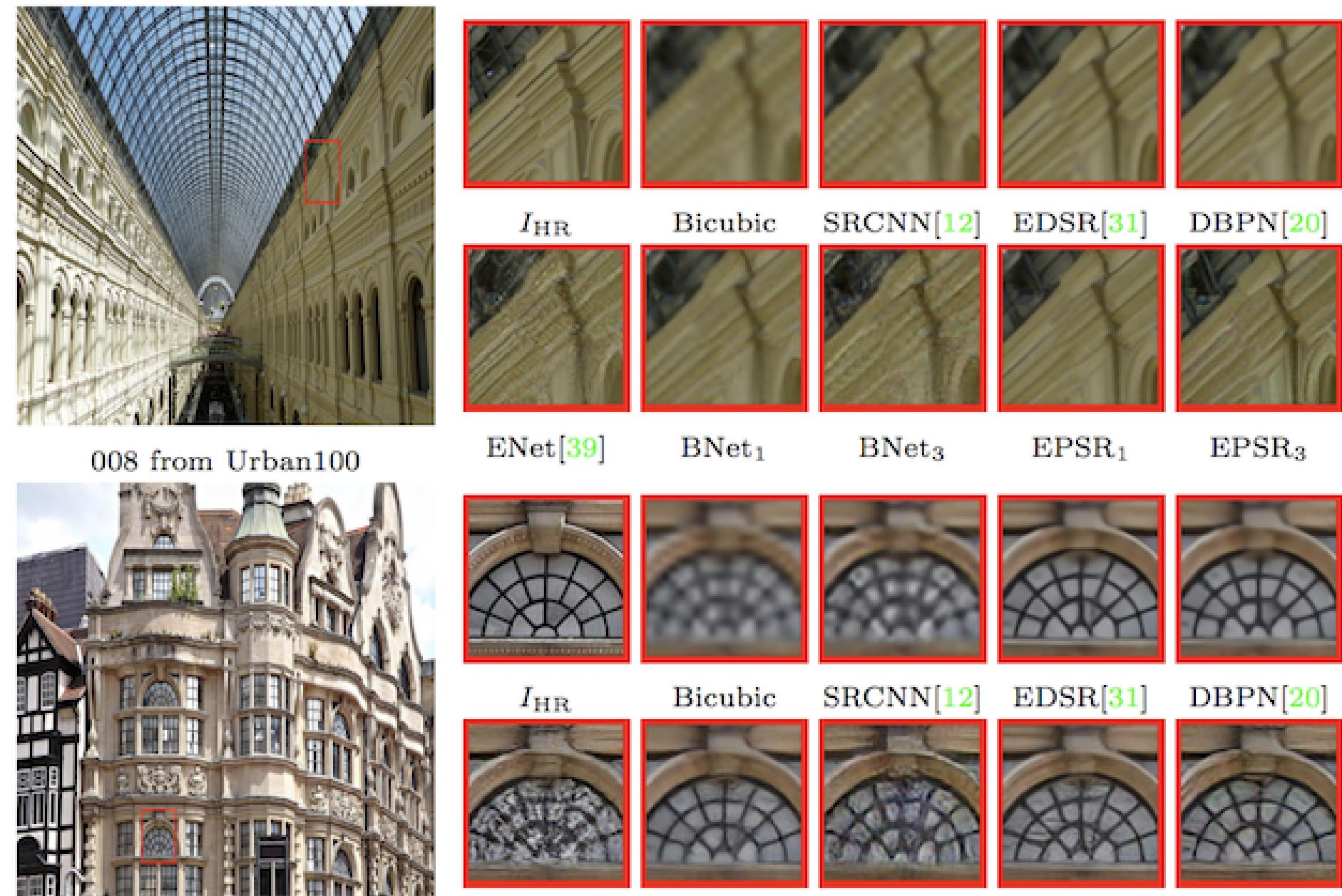
Blonde ↑

Bangs ↑

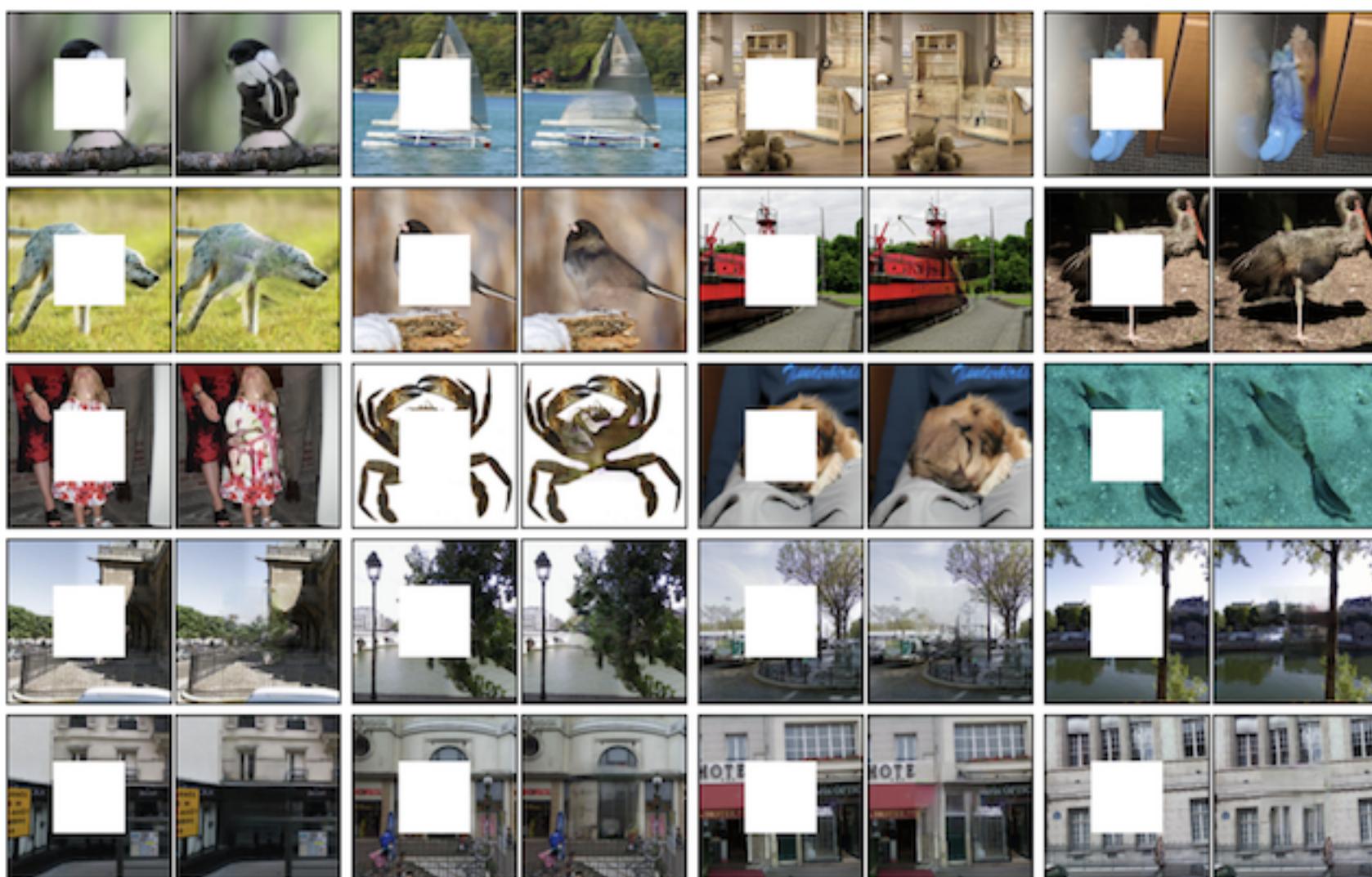
Smile ↑

Male ↑

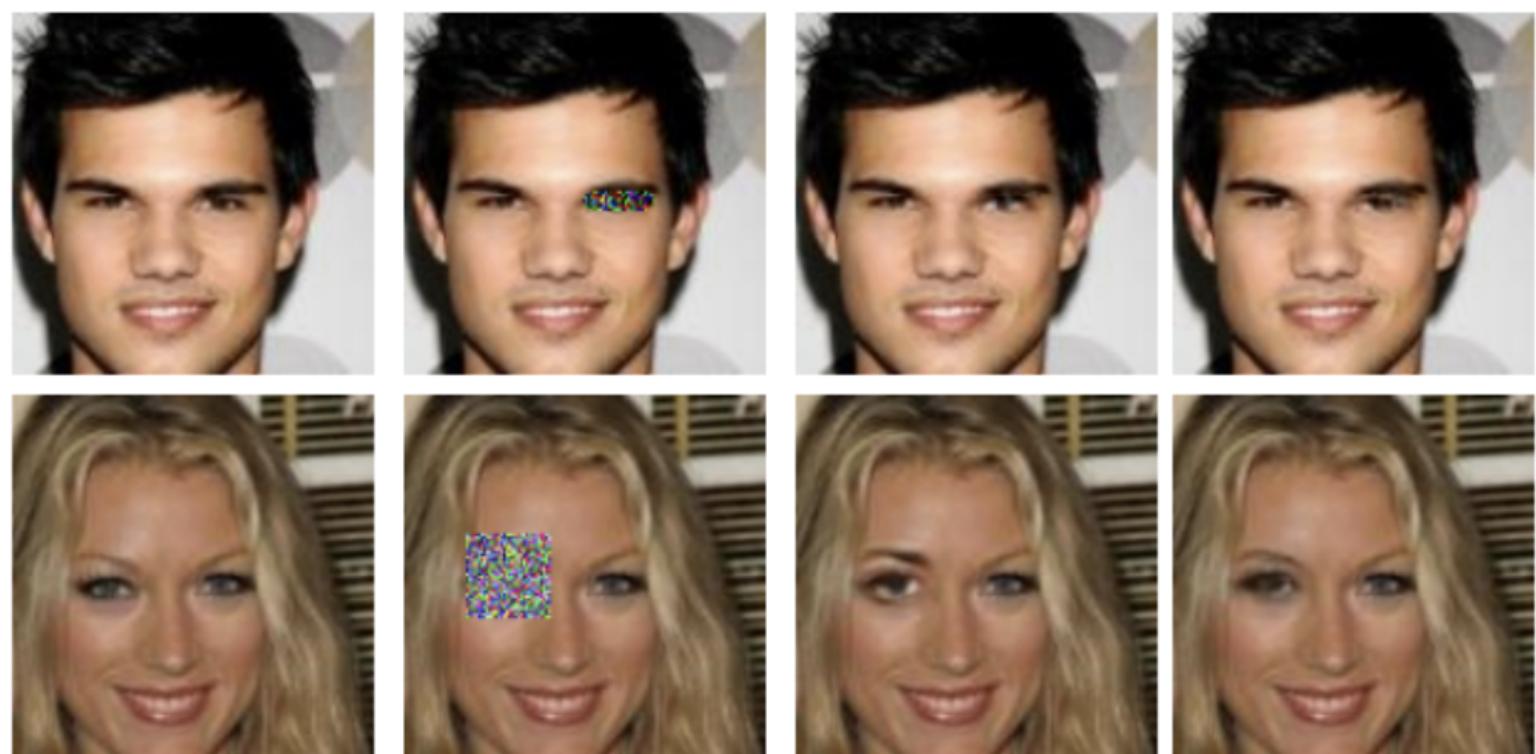
/Примеры использования



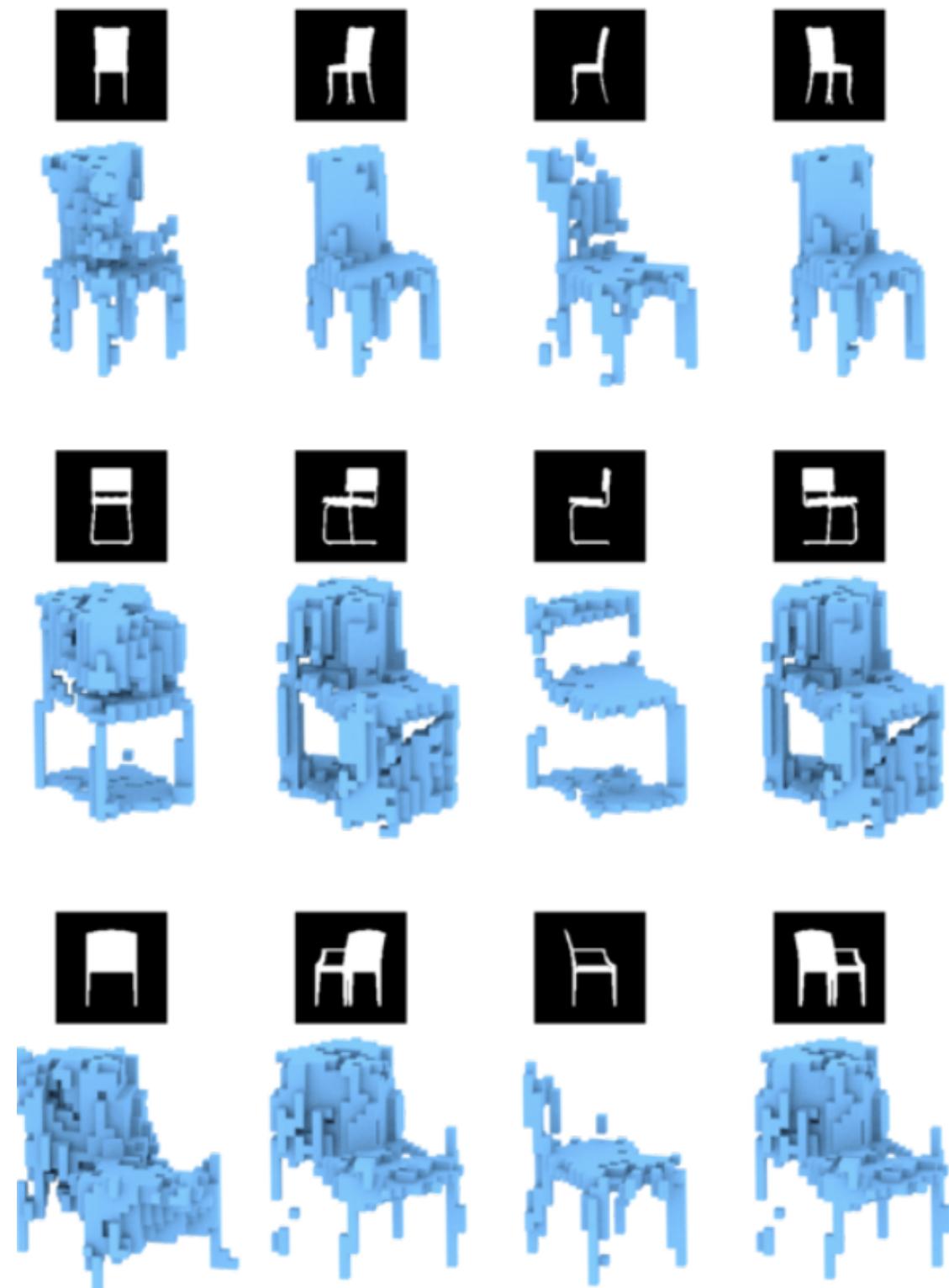
/Примеры использования



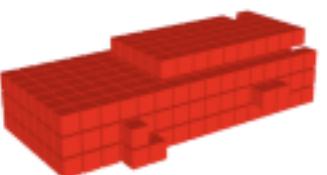
Real Input Ours NN



/Примеры использования



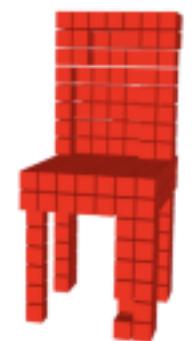
High-res



Low-res



High-res



Low-res



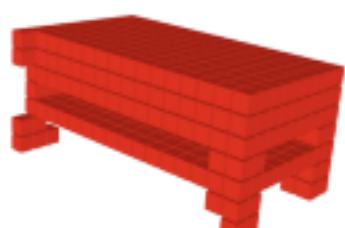
High-res



Low-res



High-res



Low-res

/Ссылки

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde- Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in neural information processing systems, 2014, pp. 2672–2680.

T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans,” in Advances in neural information processing systems, 2016, pp. 2234–2242.

Arjovsky, Martin, and Léon Bottou. "Towards principled methods for training generative adversarial networks." arXiv preprint arXiv:1701.04862 (2017).

<https://vincentherrmann.github.io/blog/wasserstein>

<https://yandex.ru/lab/ganart>

<https://www.nvidia.com/en-us/research/ai-playground/>

<https://poloclub.github.io/ganlab/>

Спасибо за внимание!