

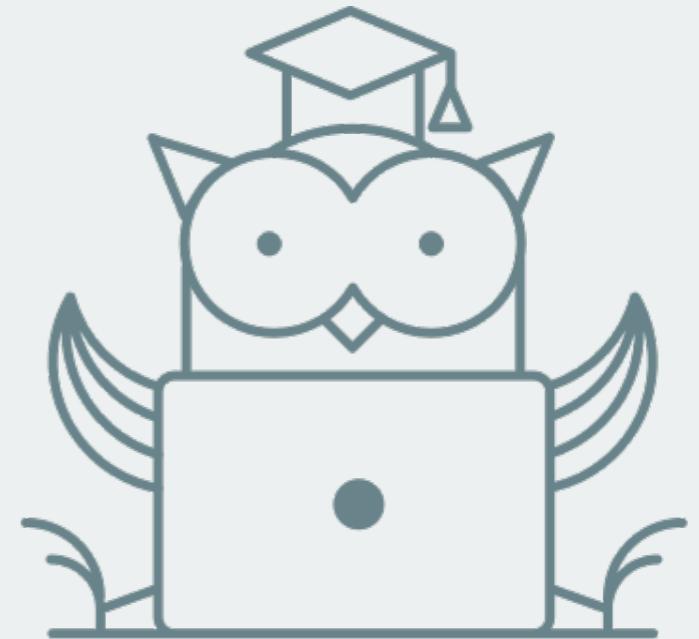


ОНЛАЙН-ОБРАЗОВАНИЕ

# Генеративные Состязательные Сети

Я тебя породил, я тебя и... отличу!

Артур Кадурин  
Insilico Medicine



1. Формула Байеса
2. Генеративные Состязательные Сети
3. DCGAN
4. Практика



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Теорема Байеса позволяет перейти от априорного распределения к апостериорному, но при этом еще позволяет связать между собой условные распределения.



# Формула Байеса



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

$p(y)$  — априорное распределение над классами:

Какова вероятность выпадения орла/решки

Какая доля людей болеет раком груди

$p(x)$  — априорное распределение над объектами:

Вероятность выбора монеты достоинством 1р./2р./5р. ...

Вероятность выбора конкретного человека для анализа



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

$p(x|y)$  – условное распределение над объектами:

Если выпала решка, то какова вероятность что выбрали 5р.

Если рак груди обнаружен, то какова вероятность что это Иван  
Иванович



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

$p(y|x)$  – условное распределение над классами:

Если выбрали 5р., то какова вероятность орла/решки

Если обследуем Ивана Ивановича, то какова вероятность обнаружить рак груди



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

**Дискриминативная модель** – используя обучающую выборку  
аппроксимируем  $p(y|x)$  с помощью нейросети  $q_\theta(y|x)$



$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

**Дискриминативная модель** – используя обучающую выборку  
аппроксимируем  $p(y|x)$  с помощью нейросети  $q_\theta(y|x)$

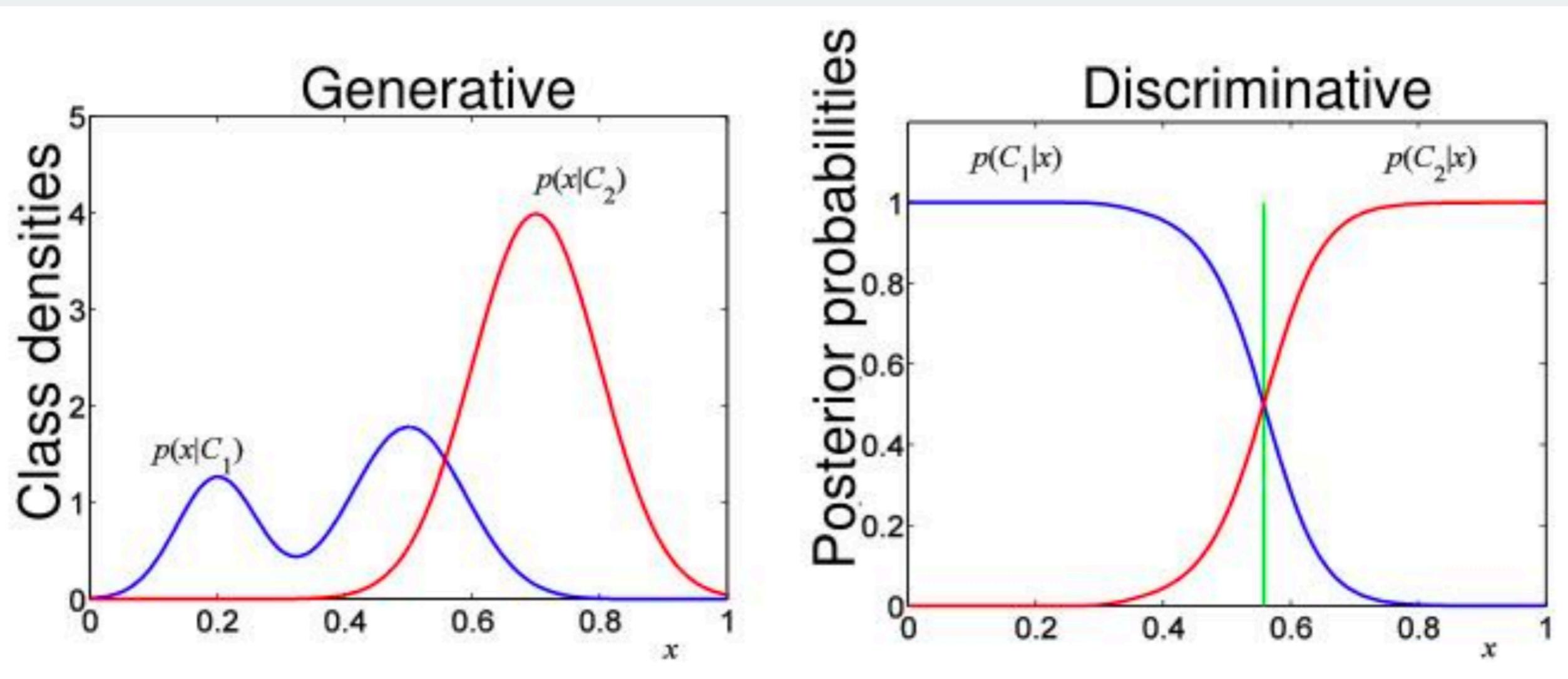
$$p(y|x) \sim p(x|y)p(y)$$

**Генеративная модель** – используя обучающую выборку оцениваем  
 $p(y)$  и аппроксимируем  $p(y|x)$  с помощью нейросети  $q_\theta(x|y)$



# Формула Байеса

O T U S



1. Формула Байеса
2. Генеративные Состязательные Сети
3. DCGAN
4. Практика



Генеративные Состязательные Сети называются так потому, что они позволяют генерировать новые объекты. Нашей целью будет обучить нейросеть  $G_\theta(y): Y \rightarrow X$ , которая по заданному «условию» порождает новый объект.



Генеративные Состязательные Сети называются так потому, что они позволяют генерировать новые объекты. Нашей целью будет обучить нейросеть  $G_\theta(y): Y \rightarrow X$ , которая по заданному «условию» порождает новый объект.

В качестве обучающей выборки мы воспользуемся каким-нибудь существующим датасетом и будем учить сеть порождать объекты «похожие» на те что есть в тренировочном множестве.



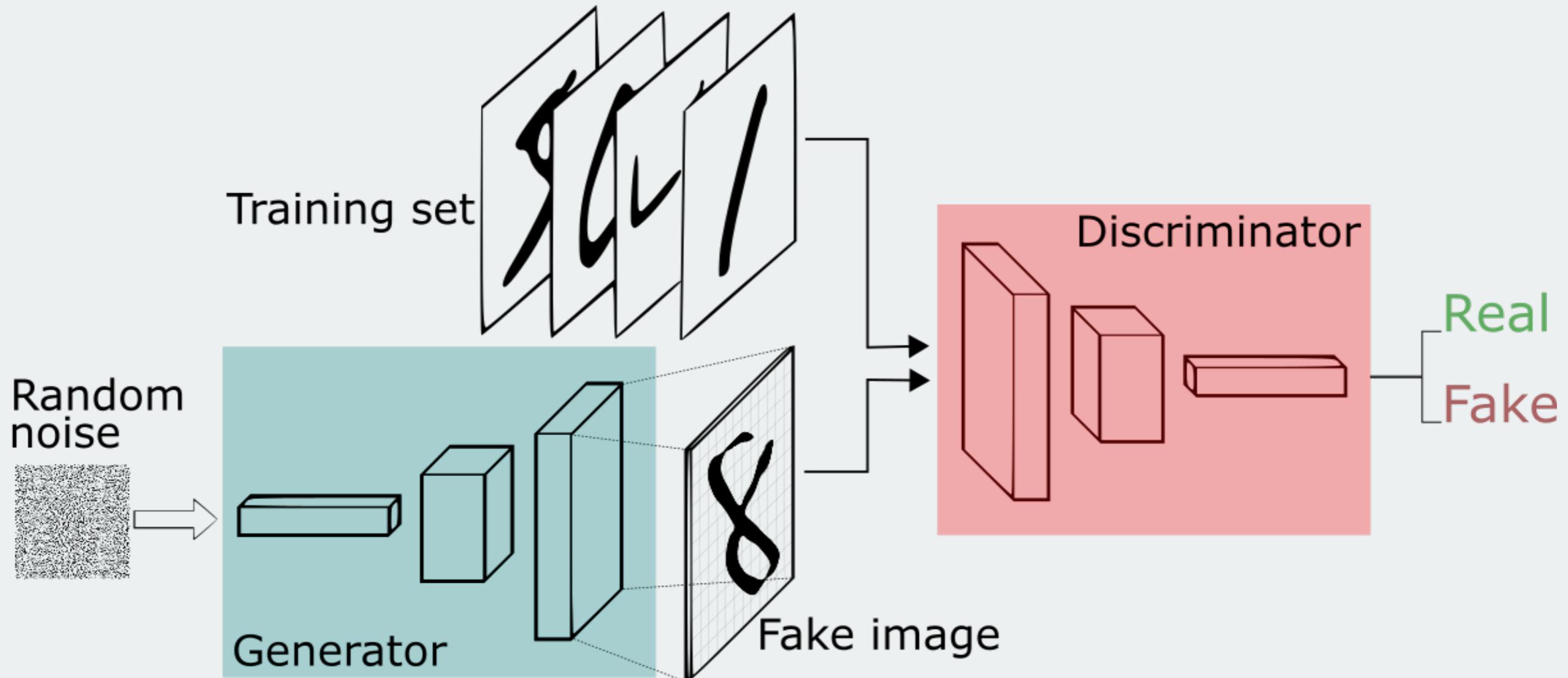
Генеративные Состязательные Сети называются так потому, что они позволяют генерировать новые объекты. Нашей целью будет обучить нейросеть  $G_\theta(y): Y \rightarrow X$ , которая по заданному «условию» порождает новый объект.

В качестве обучающей выборки мы воспользуемся каким-нибудь существующим датасетом и будем учить сеть порождать объекты «похожие» на те что есть в тренировочном множестве.

В качестве условия, для начала, мы будем использовать семплы из нормального распределения. Осталось только понять как будет выглядеть функция ошибки.

## Как можно задавать функции ошибки?





$$\mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x \sim p_{gen}(x)}[\log(1 - D(x))]$$



$$\mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x \sim p_{gen}(x)}[\log(1 - D(x))]$$

$x \sim p_{gen}(x)$  – это то же самое, что и  $G(z), z \sim p_z$



$$\mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{x \sim p_{gen}(x)}[\log(1 - D(x))]$$

$x \sim p_{gen}(x)$  – это то же самое, что и  $G(z), z \sim p_z$

$$\min_G \max_D \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))]$$

С одной стороны мы хотим максимизировать точность

Дискриминатора, с другой – максимизировать шанс Генератора  
«обмануть» Дискриминатор.



Устойчивость: минимаксная игра — поиск седловой точки

Mode collapse: часто Генератору оказывается достаточно генерировать только часть датасета

Насыщение Дискриминатора: градиенты от хорошего Дискриминатора могут быть нулевыми



Подбор гиперпараметров: баланс глубины, количества шагов обучения, оптимизаторов и т.д.

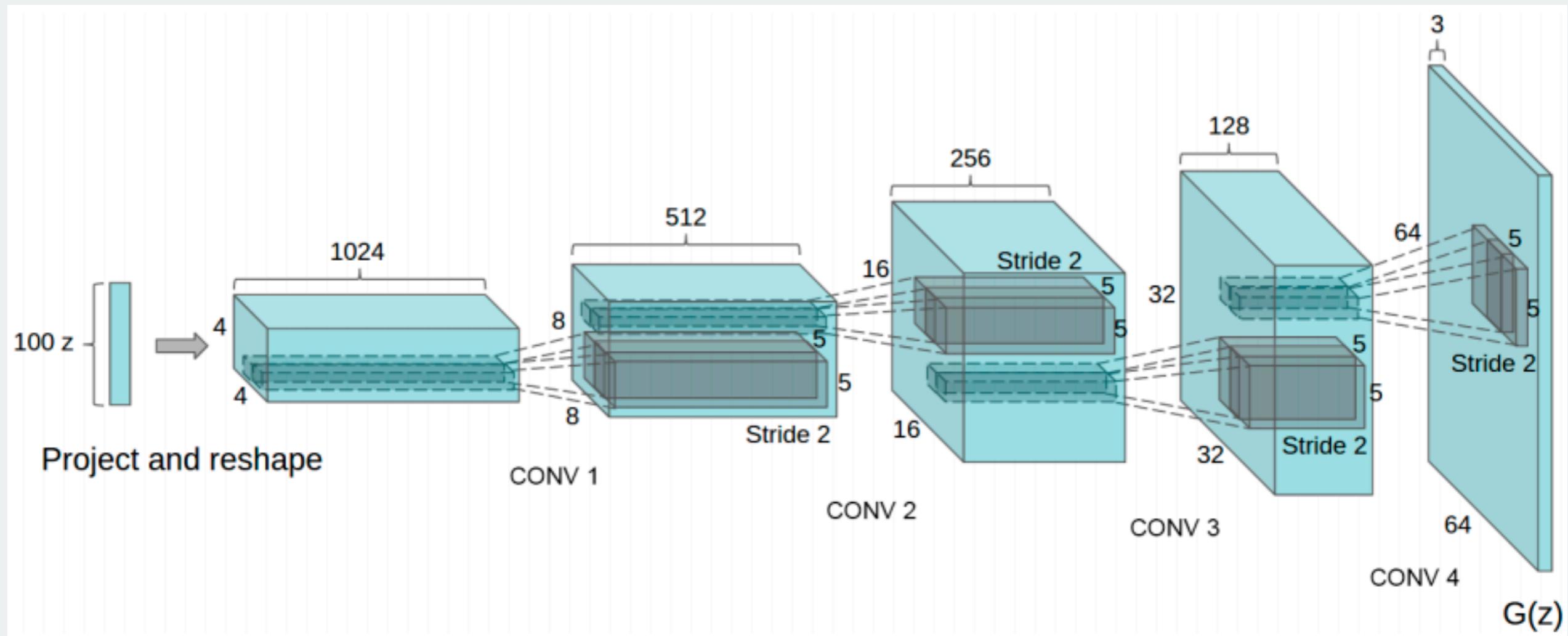
Другие функции потерь: Wasserstein loss, statistics matching

Слои: дропаут, батч нормализация и т.д.



1. Формула Байеса
2. Генеративные Состязательные Сети
3. DCGAN
4. Практика





<https://arxiv.org/abs/1511.06434>



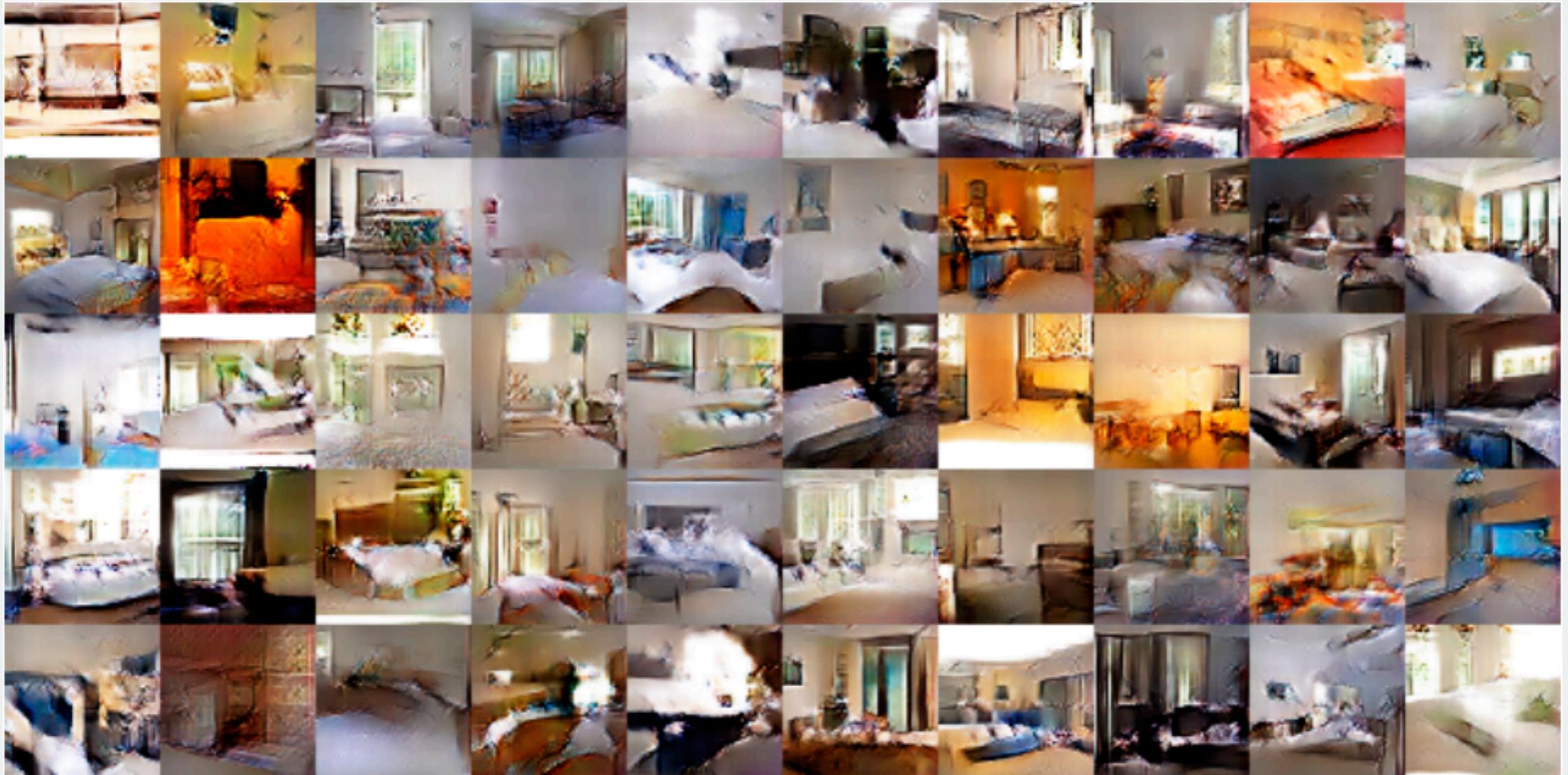
1. Ни каких пуллинг слоев как в Генераторе, так и в Дискриминаторе
2. Нет полносвязных слоев
3. Batch Normalization на всех слоях кроме выхода G и входа D
4. ReLU в Генераторе и LeakyReLU в Дискриминаторе

<https://arxiv.org/abs/1511.06434>



# DCGAN: Примеры

O T U S

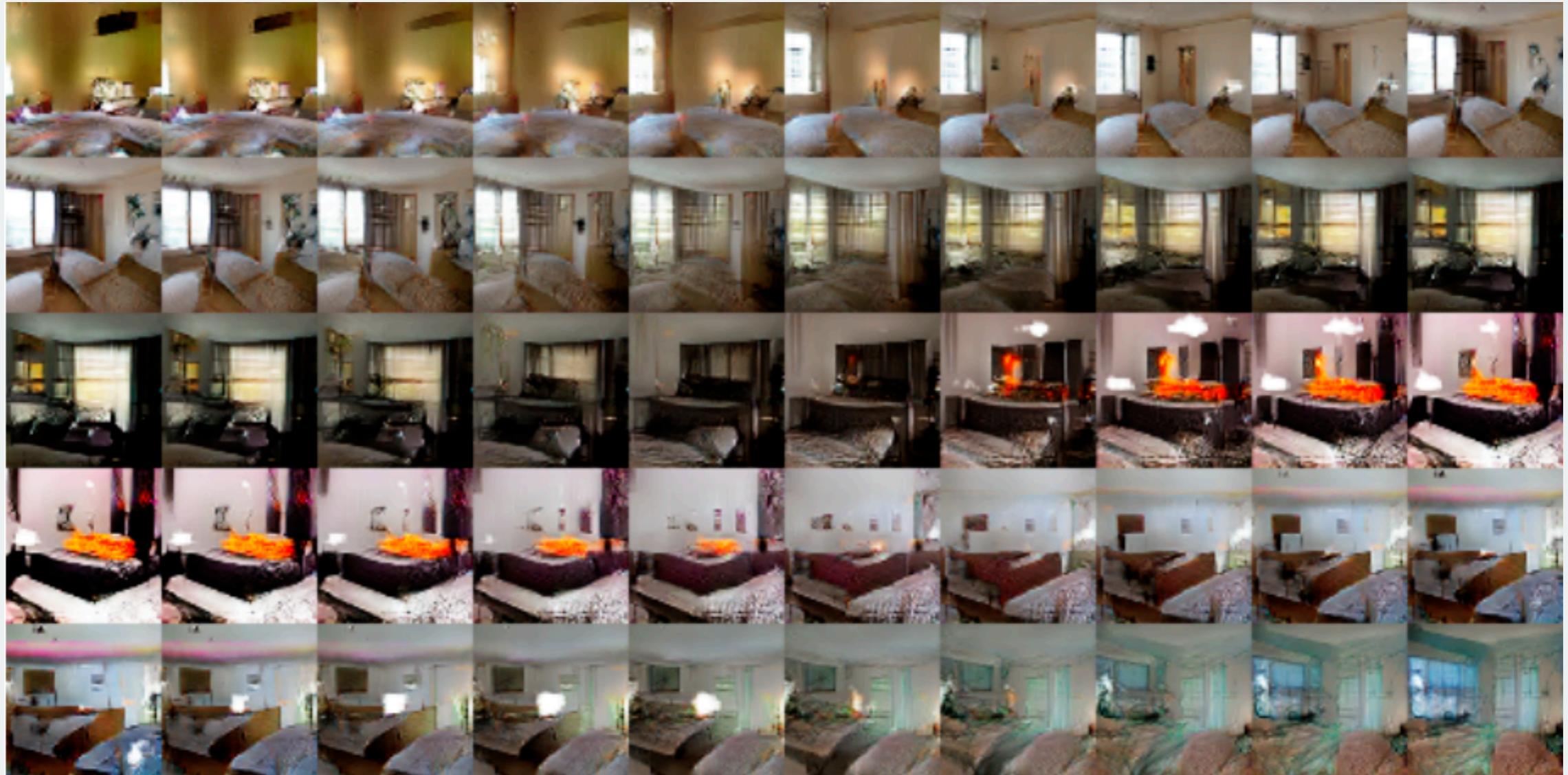


<https://arxiv.org/abs/1511.06434>



# DCGAN: Примеры

O T U S

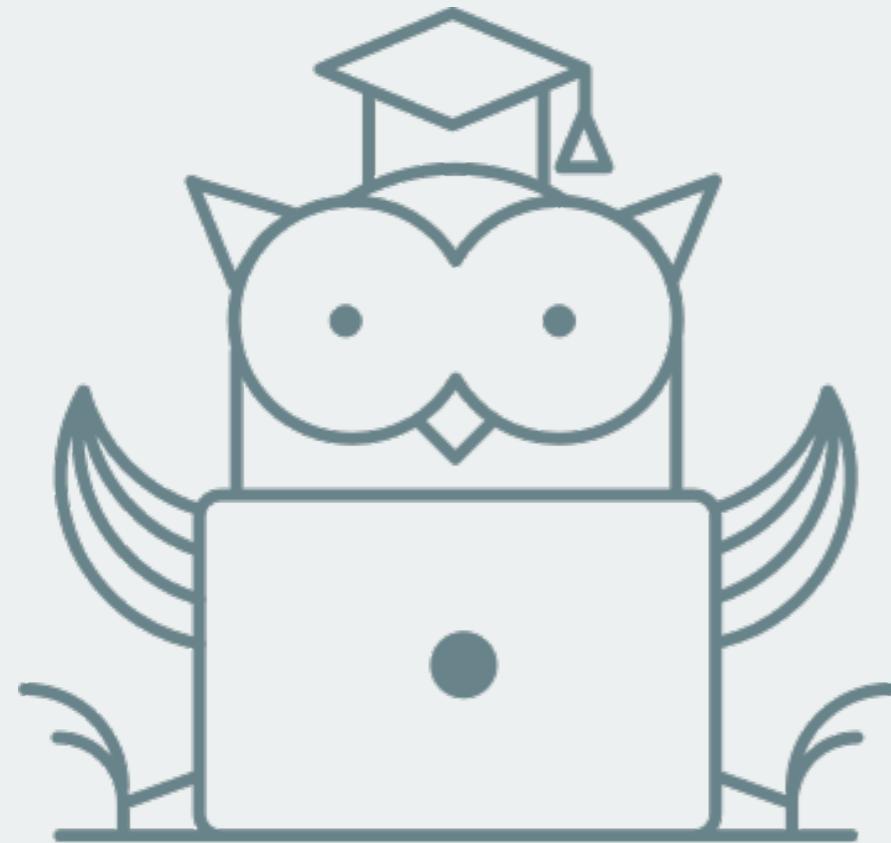


<https://arxiv.org/abs/1511.06434>



1. Формула Байеса
2. Генеративные Состязательные Сети
3. DCGAN
4. Практика





Спасибо  
за внимание!