

# Генеративное обучение

Денис Волхонский

# Программа занятия

- Зачем нужны генеративные сети и что они умеют
- Как работают Generative Adversarial Networks
- Советы по обучению GANs
- Открытые проблемы GANs

# Progressive Growing of GANs

- Video: <https://www.youtube.com/watch?v=G06dEcZ-QTg>

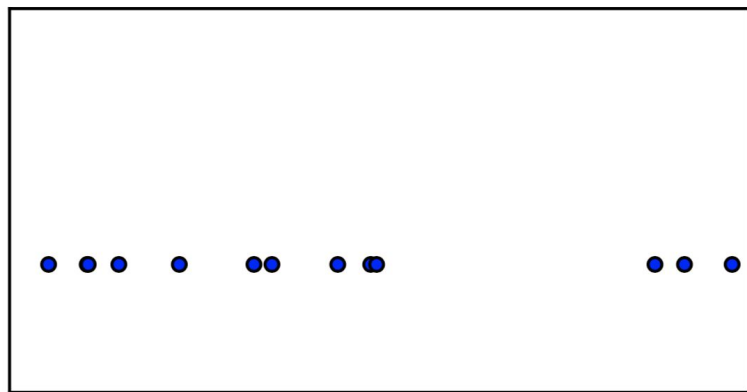


# Генеративное обучение

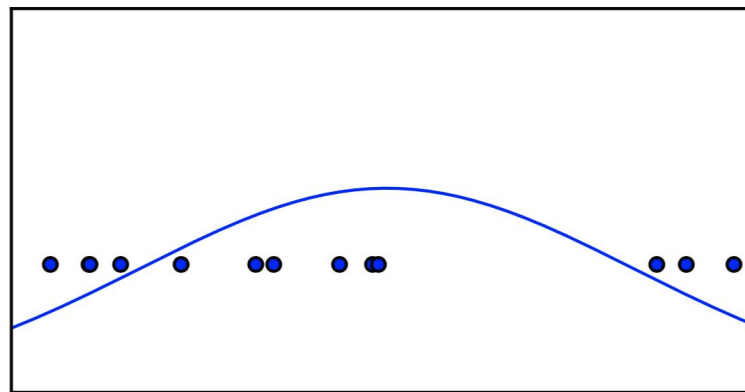
- Вход: набор данных  $X$  (например, изображений)
- Предположение: данные получены из распределения  $p(x)$
- Задача: научиться генерировать новые данные из  $p(x)$

**Вопрос: как?**

# Генеративное обучение: оценка плотности



Training Data

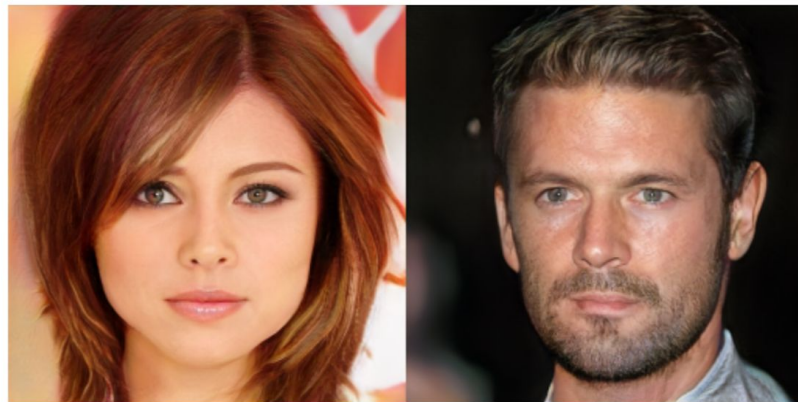


Density Function

# Генеративное обучение: генерация новых данных



Training Data  
(CelebA)

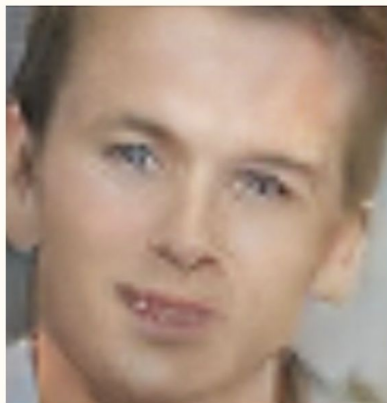


Sample Generator  
(Karras et al, 2017)

# История. Генерация лиц



2014



2015



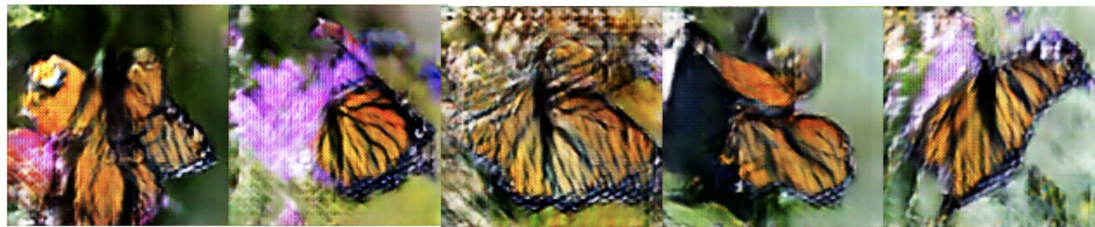
2016



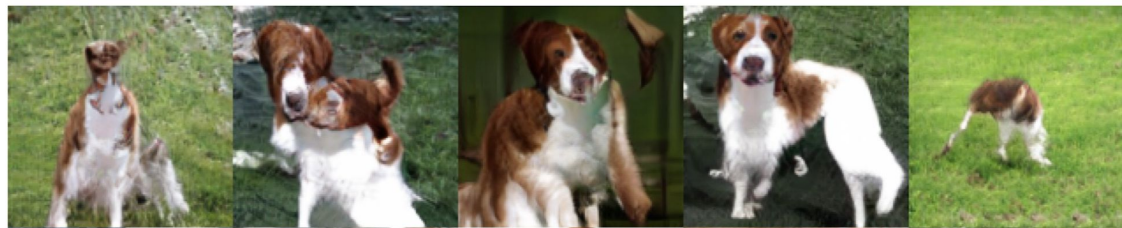
2017

# История. Генерация Image Net

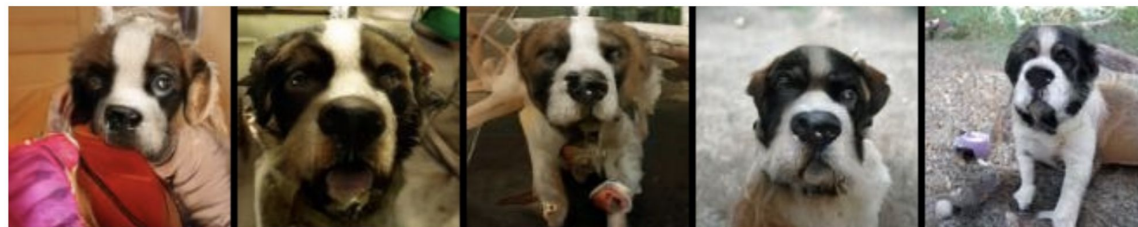
Odena et al  
2016



Miyato et al  
2017



Zhang et al  
2018



**Вопрос: зачем нужно генеративное обучение?**

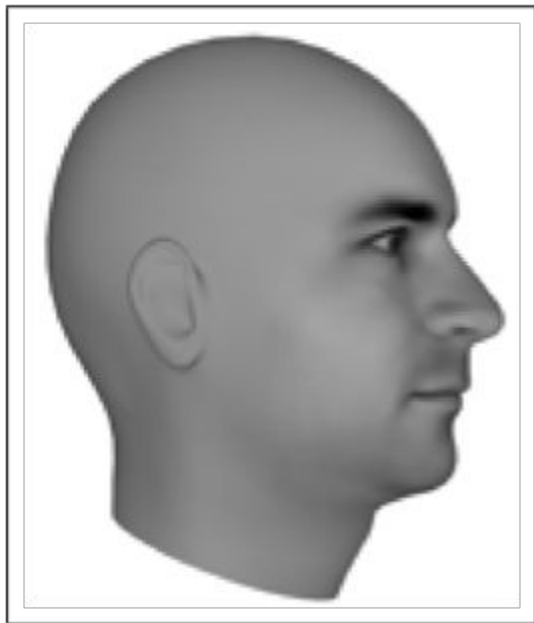


# Зачем нужно генеративное обучение?

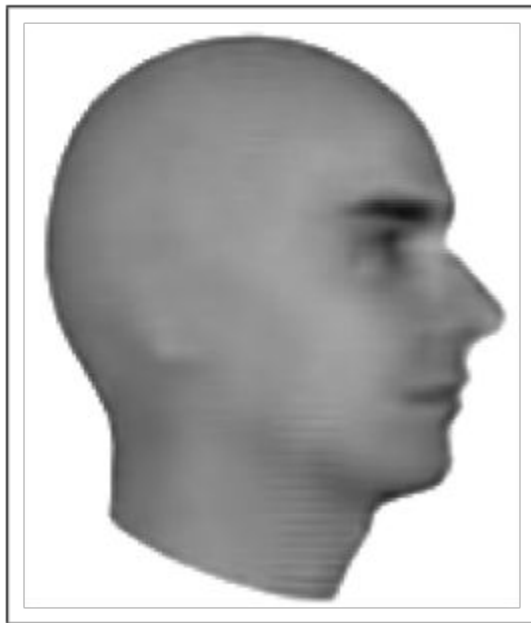
- Работа с высокоразмерными сложными распределениями
- Симулирование возможного будущего (планирование, RL)
- Работа с пропусками в данных
- Semi-supervised learning
- Мульти-модальный выход
- Генерация реалистичных семплов

# Предсказание следующего кадра видео

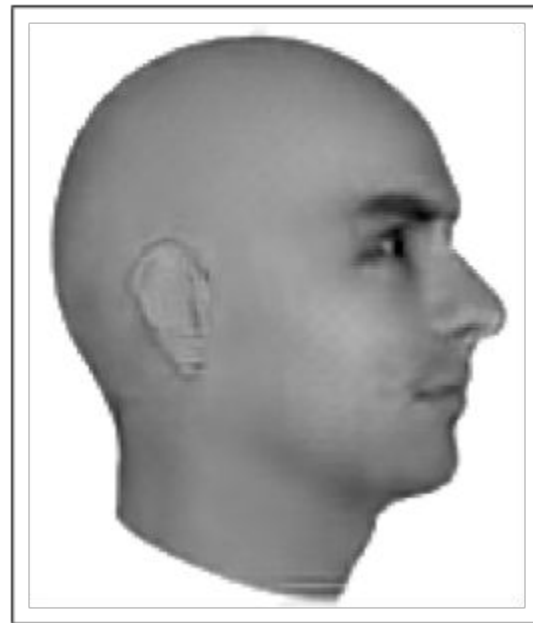
Ground Truth



MSE



Adversarial



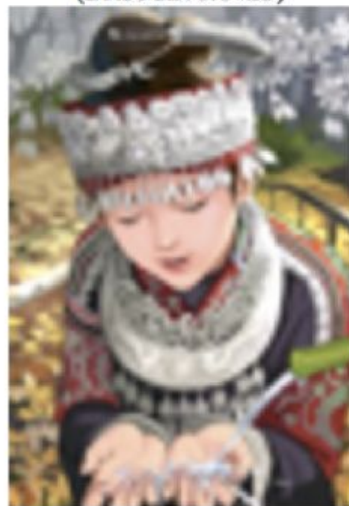
(Lotter et al 2016)

# Super-resolution

original



bicubic  
(21.59dB/0.6423)



SRResNet  
(23.44dB/0.7777)

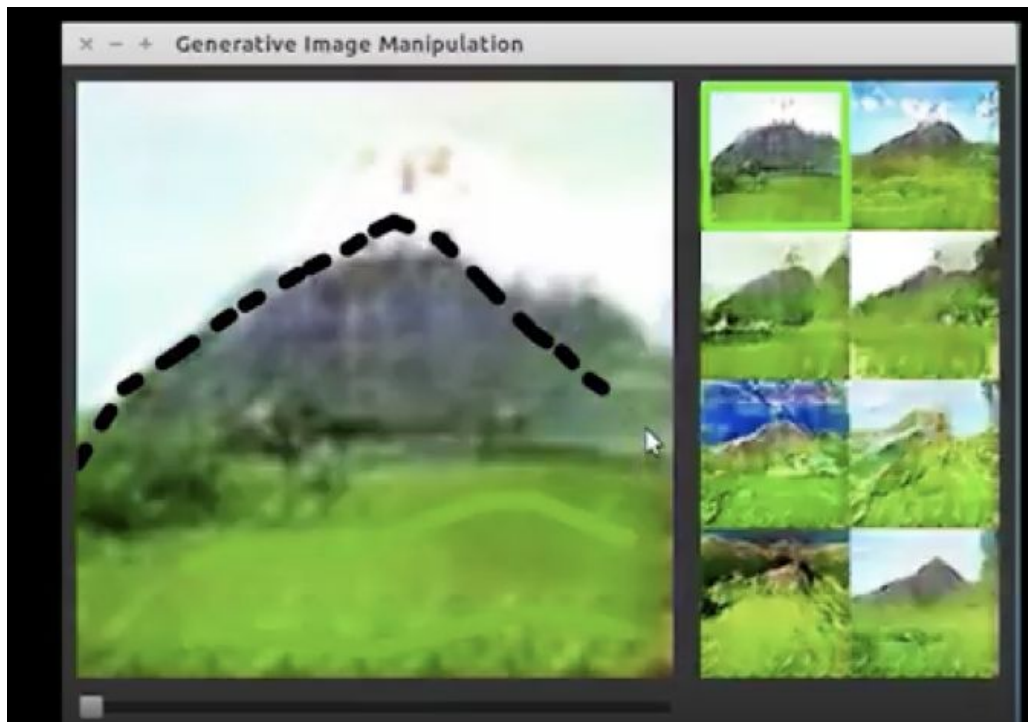


SRGAN  
(20.34dB/0.6562)



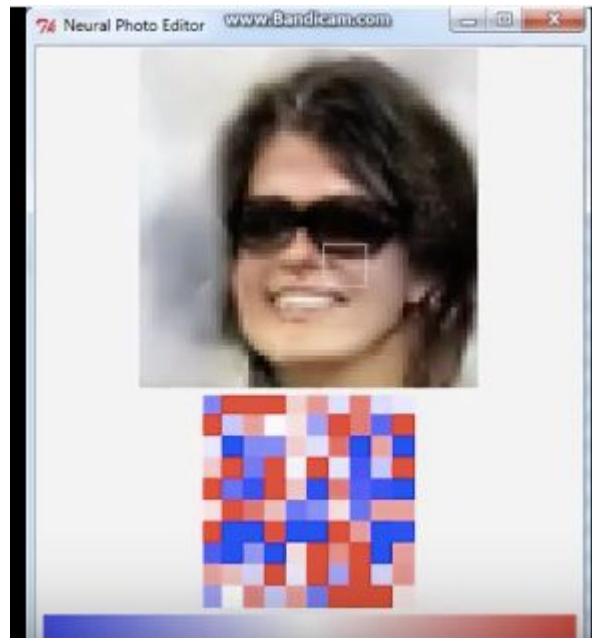
# iGAN

- (Zhu et al 2016)
- <https://www.youtube.com/watch?v=9c4z6YsBGQ0>



# Introspective Adversarial Networks

- (Brock et al 2016)
- <https://www.youtube.com/watch?v=FDELBFSegQs>

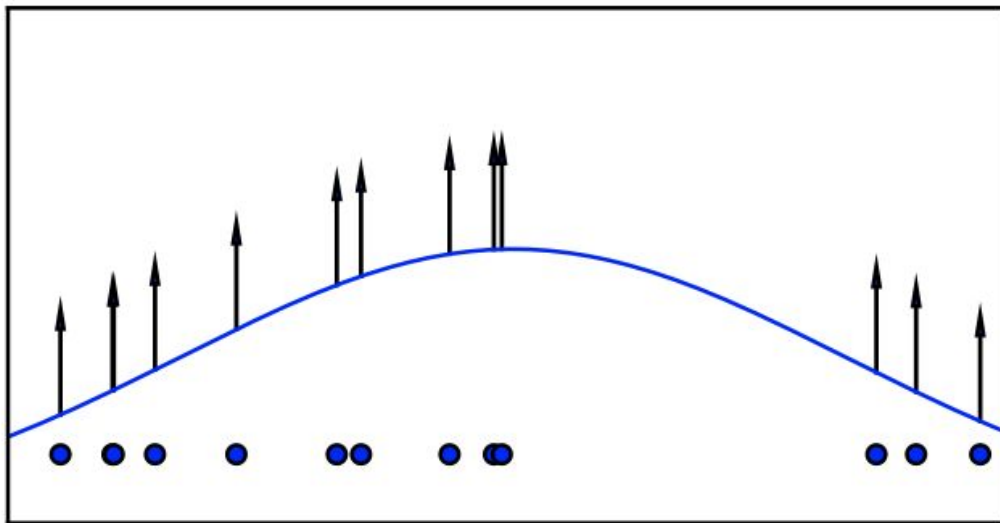


# Image to Image Translation



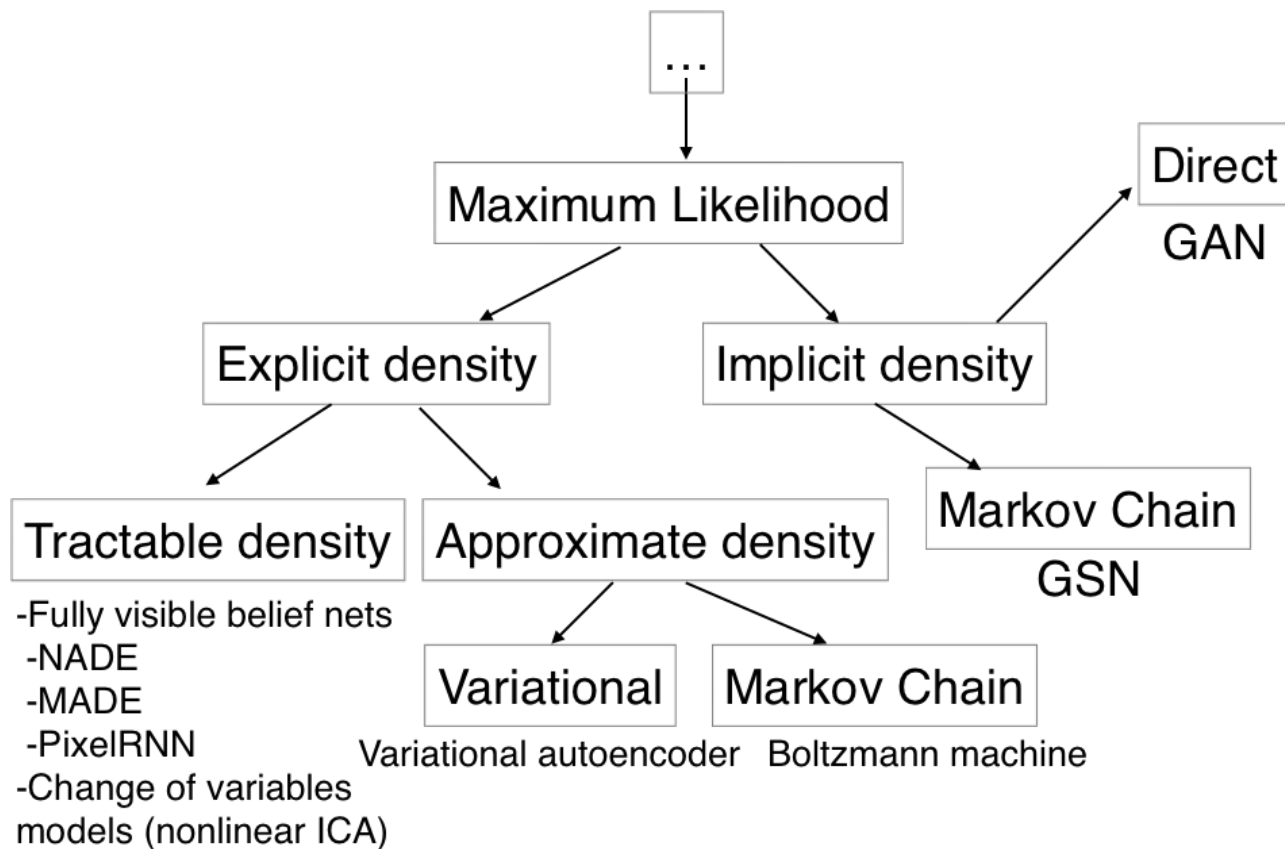
(Isola et al 2016)

# Принцип максимального правдоподобия



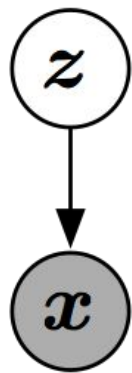
$$\theta^* = \arg \max_{\theta} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(x \mid \theta)$$

# Генеративные модели





# Variational Autoencoder



$$\begin{aligned}\log p(\mathbf{x}) &\geq \log p(\mathbf{x}) - D_{\text{KL}}(q(\mathbf{z}) \| p(\mathbf{z} | \mathbf{x})) \\ &= \mathbb{E}_{\mathbf{z} \sim q} \log p(\mathbf{x}, \mathbf{z}) + H(q)\end{aligned}$$



CIFAR-10 samples  
(Kingma et al 2016)

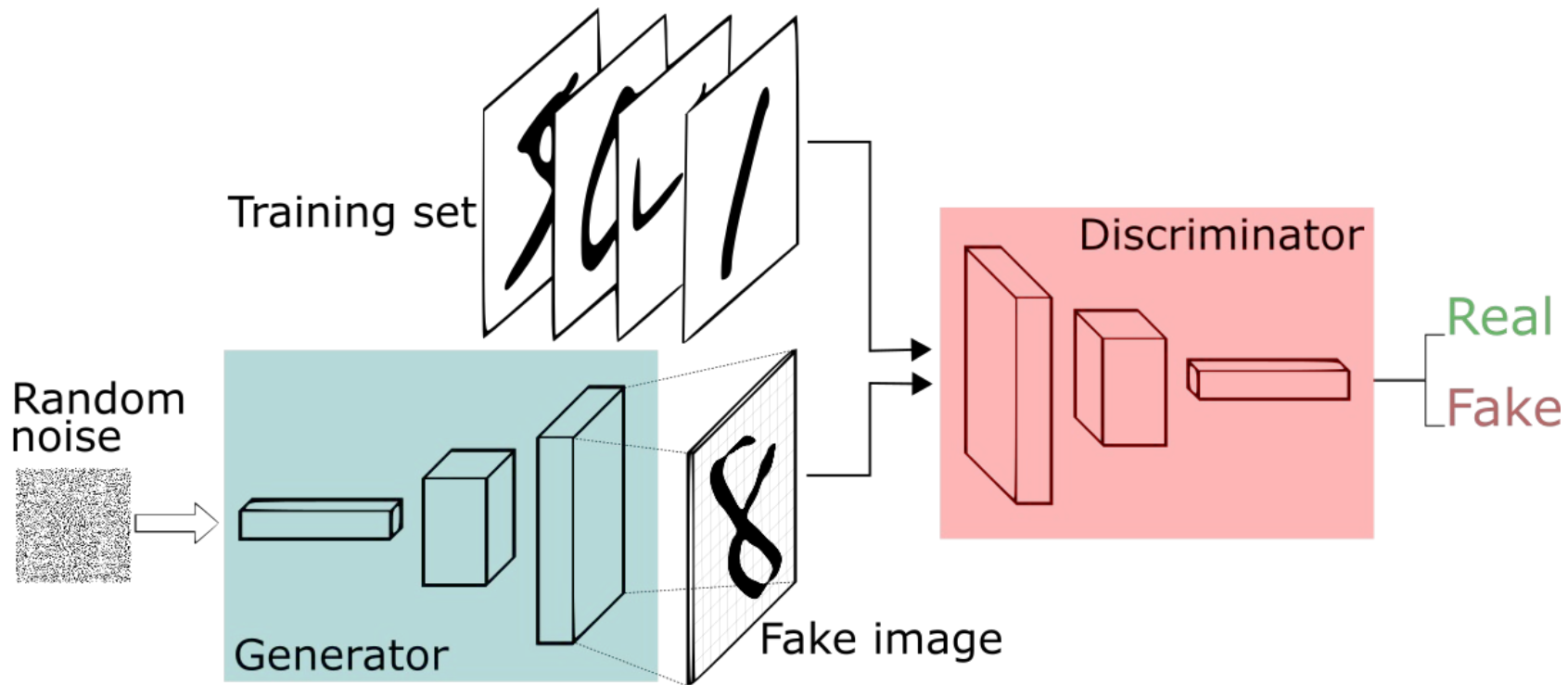
Недостаток: плохое качество генерации

# GANs

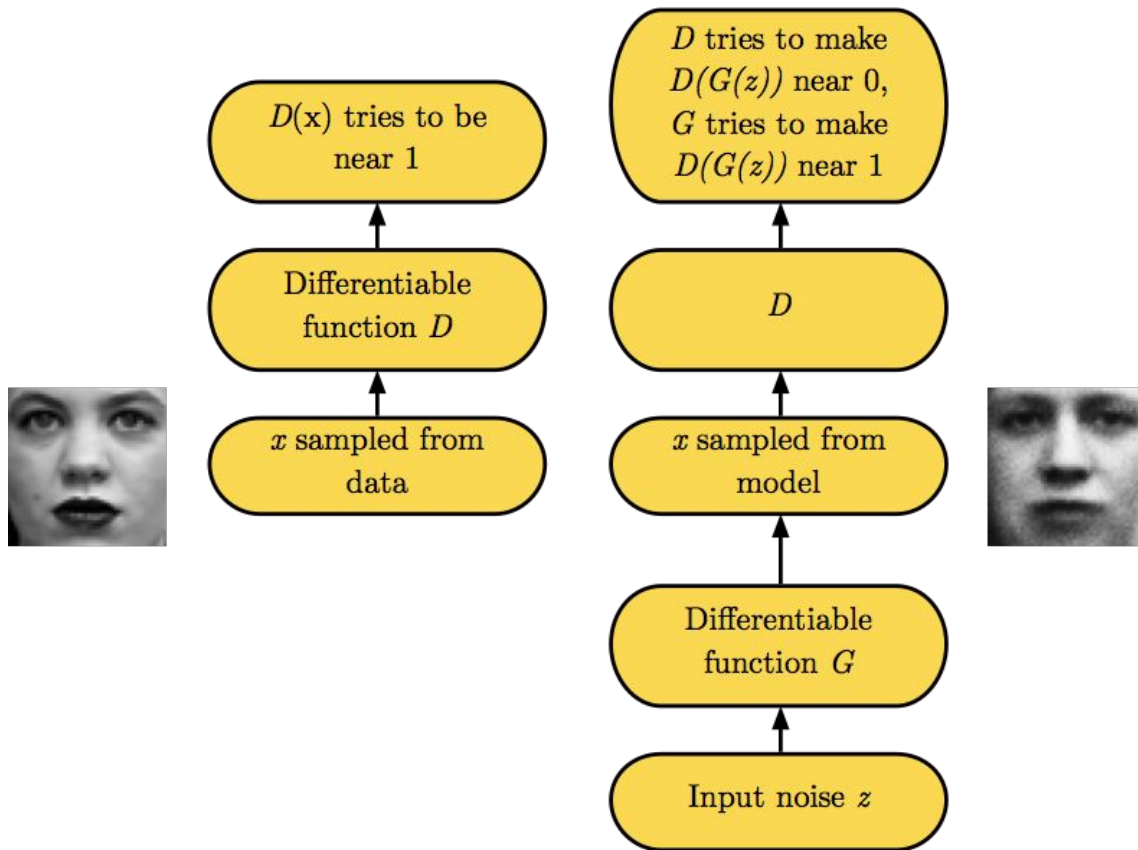
- Используют скрытое представление
- Не требуются Марковские цепи
- Самое лучшее качество генерации

Как работают GANs

# Как работают GANs



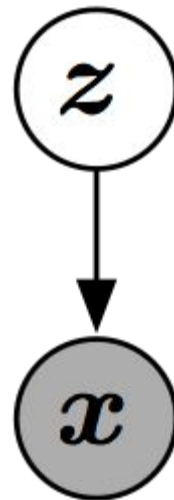
# Как работают GANs



# Генератор

- Дифференцируемая функция
- Не требуется инвертируемость
- Любой размер  $z$

$$\mathbf{x} = G(\mathbf{z}; \boldsymbol{\theta}^{(G)})$$



# Дискриминатор

- Обычный бинарный классификатор
- Дифференцируемая функция
- На вход принимает реальные объекты и сгенерированные
- Выход: fake/real

# Обучение GANs

- Использовать SGD-like алгоритм на двух минибатчах одновременно:
  - Минибатч тренировочного множества
  - Минибатч сгенерированных объектов
- Опционально:  $k$  шагов для одного игрока на один шаг другого



# Minimax Game

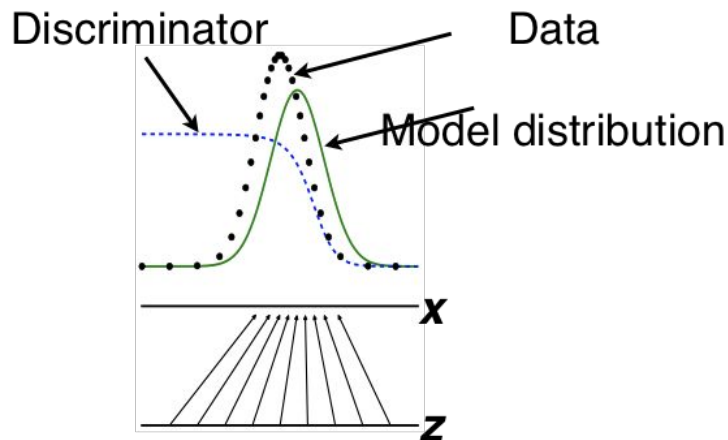
$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \log D(\mathbf{x}) - \frac{1}{2}\mathbb{E}_{\mathbf{z}} \log (1 - D(G(\mathbf{z})))$$

$$J^{(G)} = -\frac{1}{2}\mathbb{E}_{\mathbf{z}} \log D(G(\mathbf{z}))$$

- Генератор максимизирует логарифмическую вероятность ошибки дискриминатора
- Эвристическая мотивация; генератор все еще может учиться, даже когда дискриминатор успешно отвергает все образцы генератора

# Стратегия дискриминатора

- Для любых  $p_{\text{data}}(x)$  и  $p_{\text{model}}(x)$  оптимальная  $D(x)$  будет:

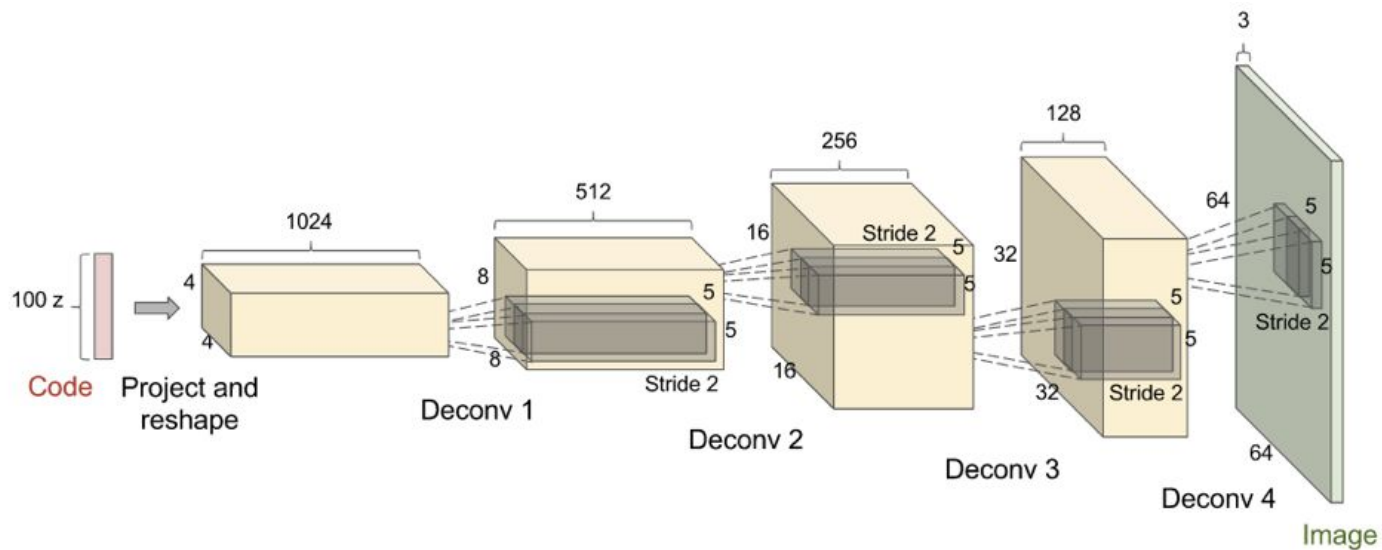


$$D(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_{\text{model}}(x)}$$

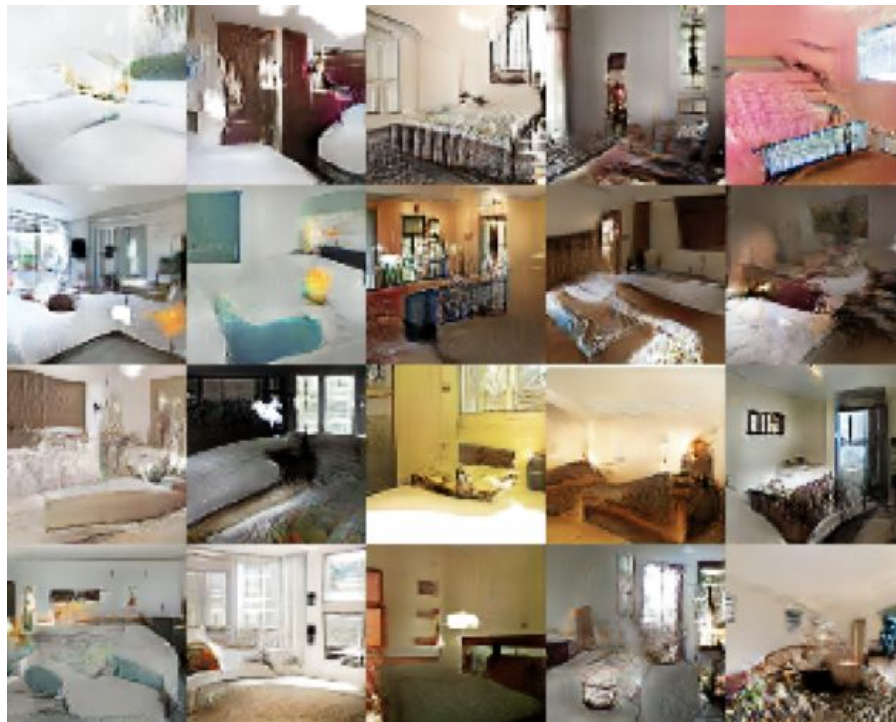
Оценка этой плотности в режиме обучения с учителем — ключевая аппроксимация, используемая в GANs

# DCGAN

- Deconvolution layers + BN

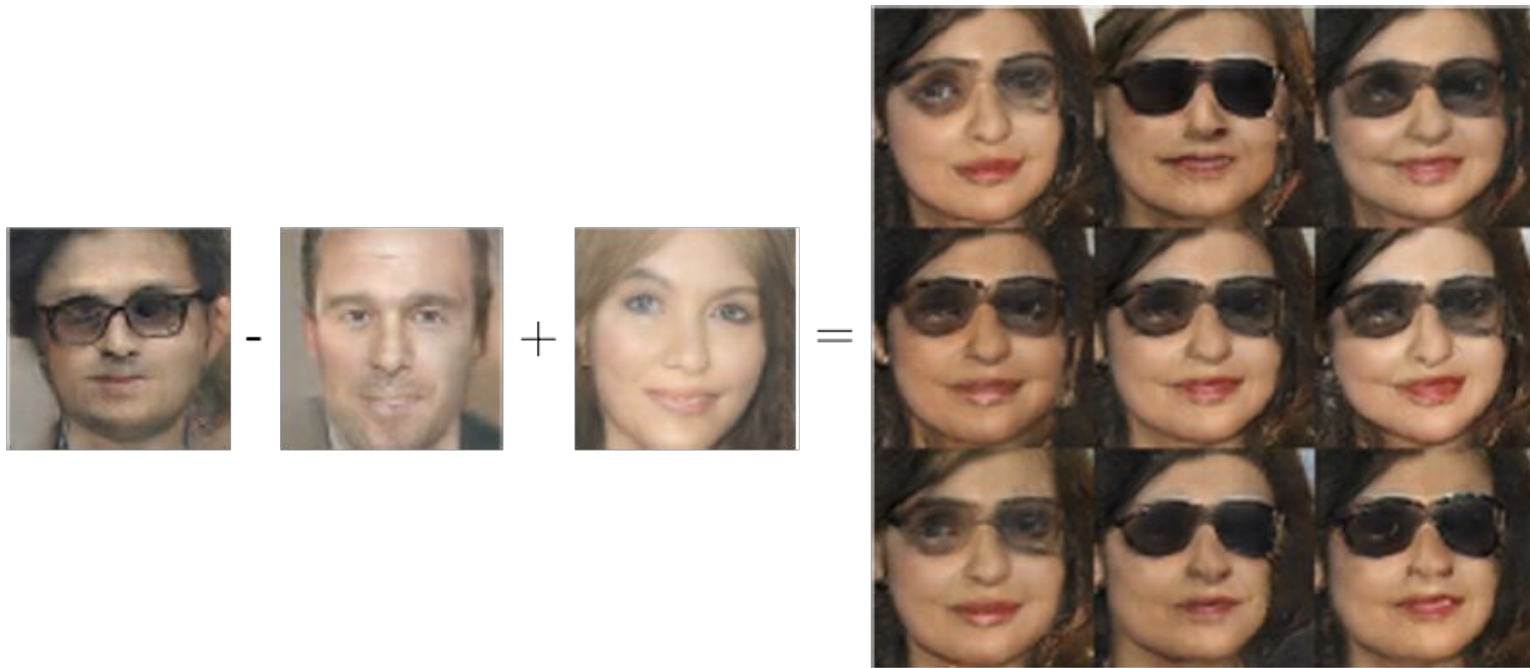


# DCGANs для LSUN Bedrooms



(Radford et al 2015)

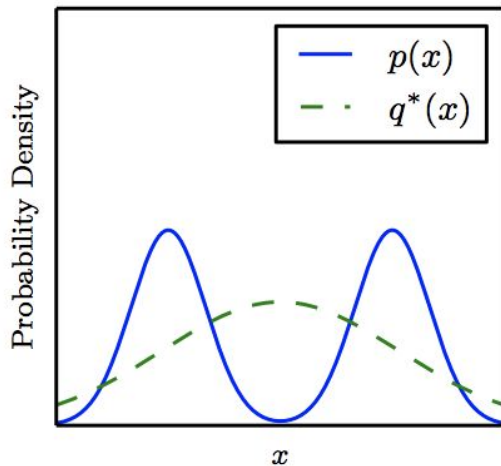
# Арифметика GANs



(Radford et al, 2015)

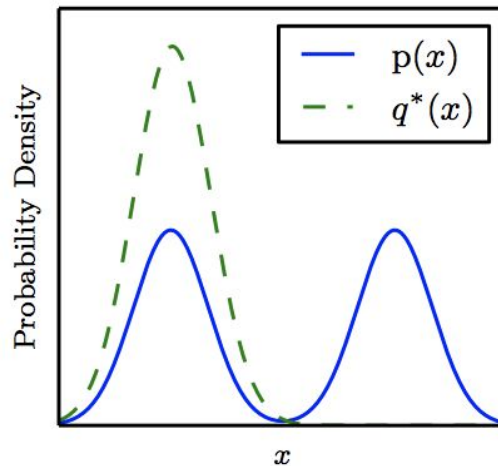
# Важна ли дивергенция?

$$q^* = \operatorname{argmin}_q D_{\text{KL}}(p \| q)$$



Maximum likelihood

$$q^* = \operatorname{argmin}_q D_{\text{KL}}(q \| p)$$



Reverse KL

(Goodfellow et al 2016)

# Модификация GANs к ММП

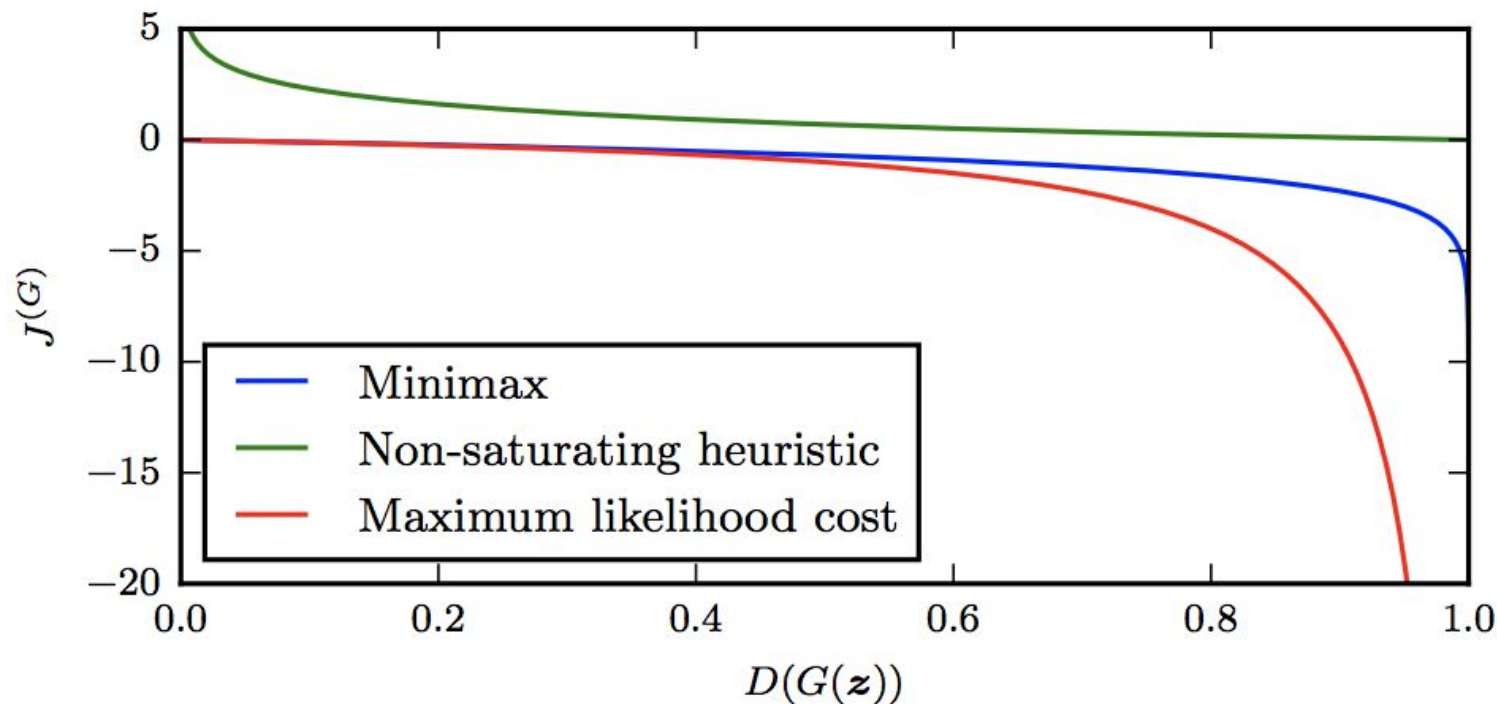
$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \log D(\mathbf{x}) - \frac{1}{2}\mathbb{E}_{\mathbf{z}} \log (1 - D(G(\mathbf{z})))$$

$$J^{(G)} = -\frac{1}{2}\mathbb{E}_{\mathbf{z}} \exp(\sigma^{-1}(D(G(\mathbf{z}))))$$

- Когда дискриминатор оптимален, градиент генератора совпадает с градиентом максимального правдоподобия

(“On Distinguishability Criteria for Estimating Generative Models”,  
Goodfellow 2014, pg 5)

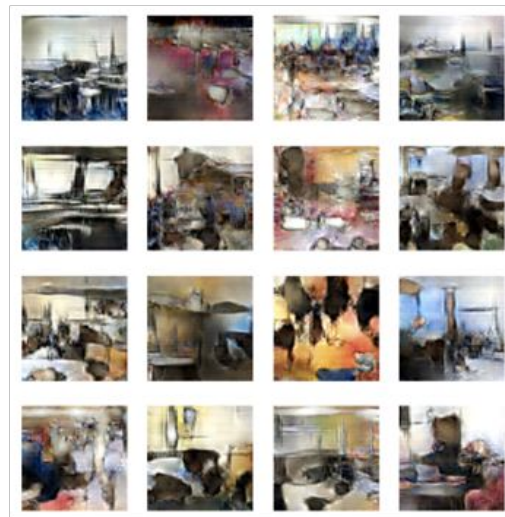
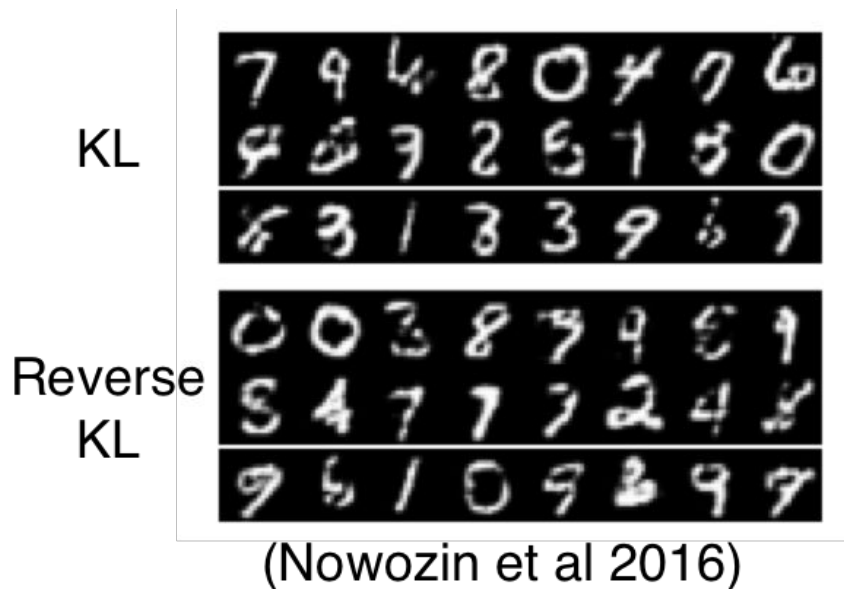
# Сравнение функций потерь Генератора



(Goodfellow 2014)



# Loss не объясняет резкость изображений



KL samples from LSUN

- Аппроксимация важнее, чем функция потерь

# Советы по обучению

# Лейблы улучшают субъективное качество изображений

- Обучение условной модели  $p(y|x)$  обычно даёт лучшие результаты для всех классов, чем обучение  $p(x)$  (Denton et al 2015)
- Даже обучение  $p(x,y)$  заставляет семплы из  $p(x)$  выглядеть гораздо лучше (Salimans et al 2016)

# Одностороннее сглаживание меток

- Loss дискриминатора по умолчанию

`cross_entropy(1., discriminator(data)) + cross_entropy(0., discriminator(samples))`

- Одностороннее сглаживание меток (Salimans et al 2016)

`cross_entropy(.9, discriminator(data)) + cross_entropy(0., discriminator(samples))`

# Преимущества сглаживания меток

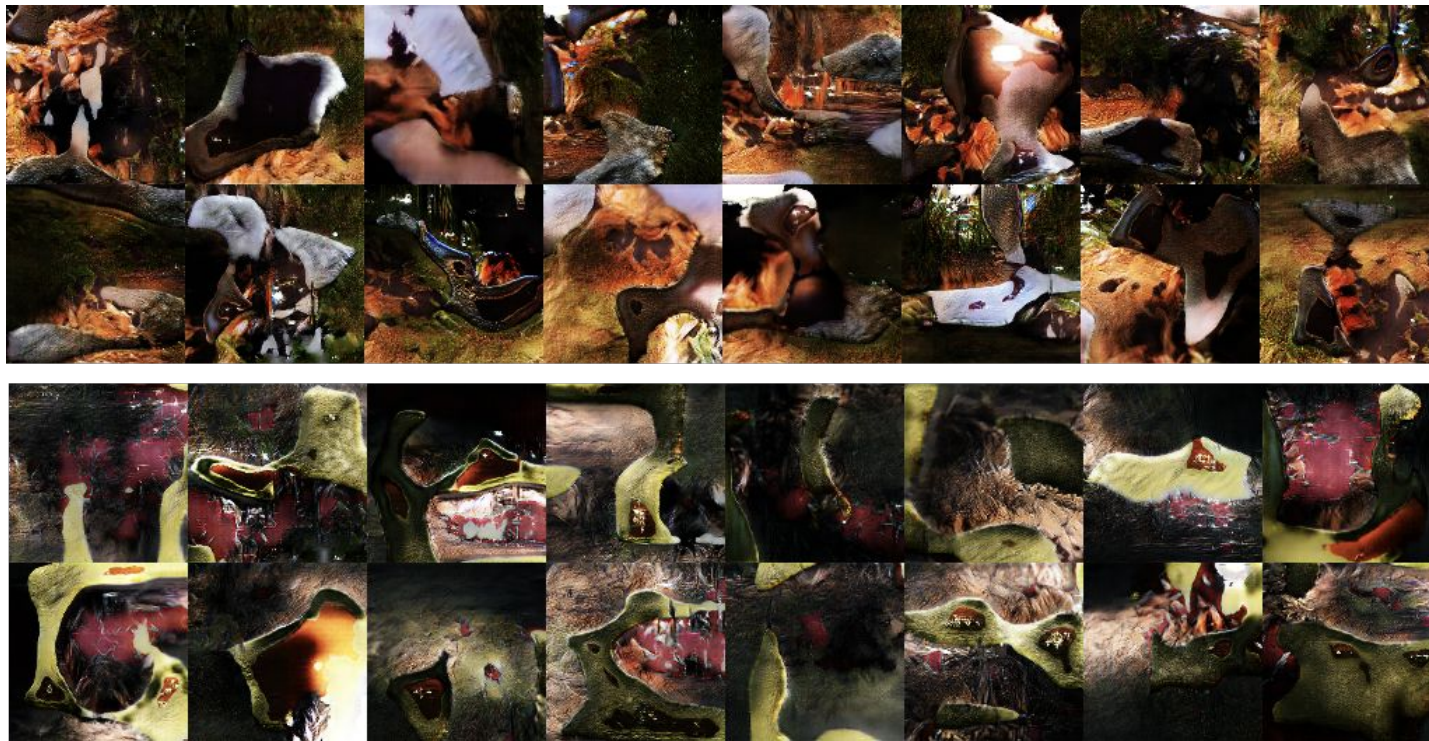
- Хорошая регуляризация (Szegedy et al 2015)
- Не снижает точность классификации
- Специфичные для GANs преимущества:
  - Предотвращает дискриминатор от передачи слишком большого градиента генератору
  - Мы не поощряем “экстремальные” сэмплы

# Batch Norm

- Вход:  $X = \{x(1), x(2), \dots, x(m)\}$
- Вычисляем среднее и стандартное отклонение признаков  $X$
- Нормализуем признаки (вычитаем среднее, делим на стандартное отклонение)
- Операция нормализации — часть графа
- Обратное распространение вычисляет градиент через нормализацию

**Вопрос: что плохого в таком подходе?**

# Корреляции внутри батча!



Вопрос: какое решение?

# Reference Batch Norm

- Фиксируем *reference batch*  $R = \{r^{(1)}, r^{(2)}, \dots, r^{(m)}\}$
- Для новых входных данных  $X = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- **Вычисляем среднее и стандартное отклонение признаков  $R$** 
  - Хотя  $R$  не изменяется, значения признаков изменяются, когда изменяются параметры модели
- Нормализуем признаки  $X$ , используя среднее и стандартное отклонение для  $R$
- **Каждый  $x^{(i)}$  всегда одинаково обрабатывается, независимо от того, какие семплы появляются в батче**
- Проблема: переобучение на  $R$ !!!

Вопрос: как будем решать?



# Virtual Batch Norm

- Фиксируем *reference batch*  $R=\{r^{(1)}, r^{(2)}, \dots, r^{(m)}\}$
- Для новых входных данных  $X=\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Для каждого  $x^{(i)}$  в  $X$ :
  - Конструируем виртуальный батч  $V$ , содержащий  $x^{(i)}$  и всё из  $R$
  - Вычисляем среднее и стандартное отклонение признаков  $V$
  - Нормализуем признаки  $x^{(i)}$ , используя среднее и стандартное отклонение признаков  $V$

# Балансировка $G$ и $D$

- Обычно дискриминатор “выигрывает”
- Это хорошо—теоретические выводы основаны на том, что  $D$  идеальный
- Обычно  $D$  больше и глубже, чем  $G$
- Иногда нужно  $D$  обновлять больше, чем  $G$
- Не пытайтесь ограничить  $D$  от того, чтобы он стал слишком умным
  - Используйте сглаживание меток класса

# Проблемы GANs

# Сходимость

- Алгоритмы оптимизации обычно сходятся в седловую точку или локальный минимум, но это нам обычно подходит
- Алгоритмы, оптимизирующие игровую постановку, могут вообще не достичь равновесия

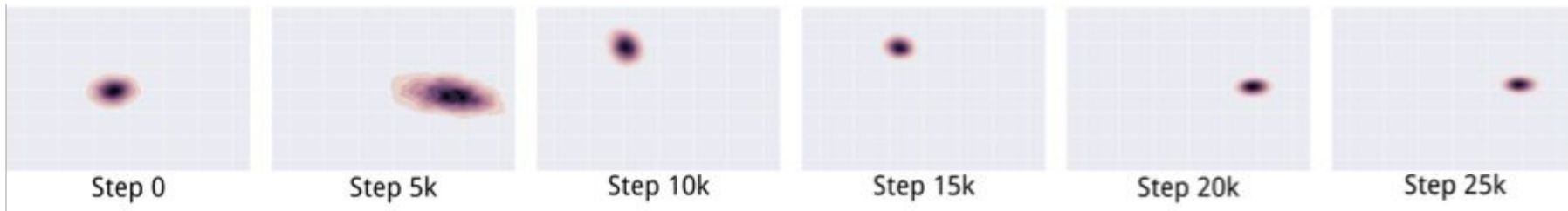
# Сходимость GANs

- Используя выпуклость в функциональном пространстве, теоретически гарантируется, что обучение GAN сходится, если мы можем напрямую изменять функции плотности, но:
  - Вместо этого мы модифицируем  $G$  и  $D$  (отношение плотностей)
  - Мы рассматриваем сложные высокоразмерные параметрические функции
- Мы можем тренироваться в течение очень долгого времени, генерируя много различных классов семплов без улучшения качества генерации

# Коллапс модели

$$\min_G \max_D V(G, D) \neq \max_D \min_G V(G, D)$$

- D внутри: сходимость к правильному распределению
- G внутри: все сгенерированные точки на самом вероятном месте



(Metz et al 2016)

# Коллапс модели

- Игры с функцией потерь не помогают
- GANs всегда коллапсируют к более слабым моделям, чем могли бы

# Коллапс модели: плохое разнообразие

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.

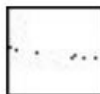


this white and yellow flower have thin white petals and a round yellow stamen



(Reed et al 2016)

**Key-points**



**GAN (Reed 2016b)**

A man in a orange jacket with sunglasses and a hat ski down a hill.



This guy is in black trunks and swimming underwater.



A tennis player in a blue polo shirt is looking down at the green court.



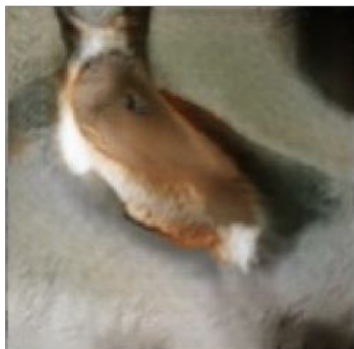
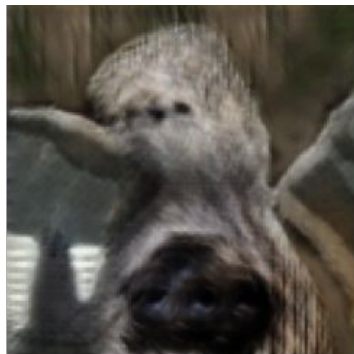
**This work**



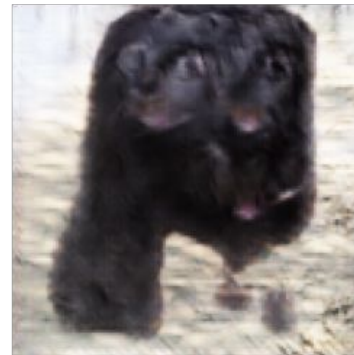
(Reed et al, ICLR 2017)



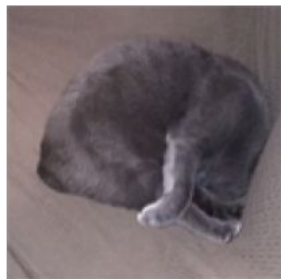
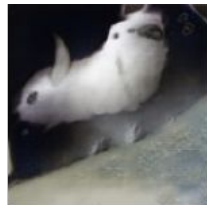
# Выборочные части объектов



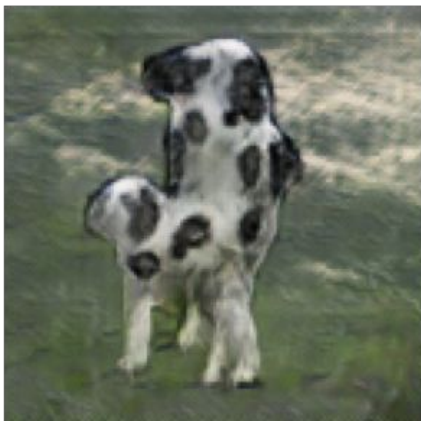
# Проблема с подсчётами



# Проблема с перспективой

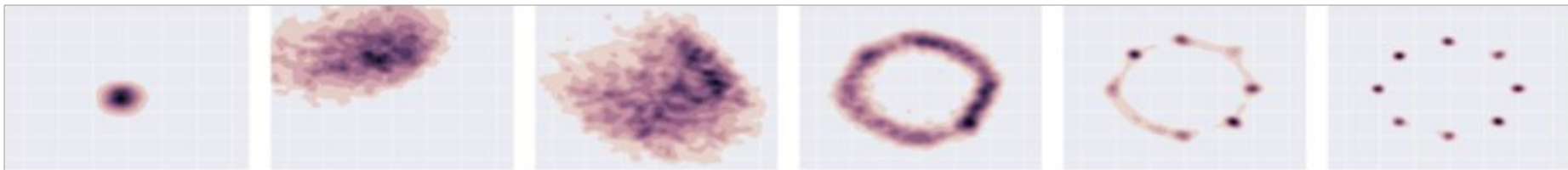


# Проблемы с глобальной структурой



# Unrolled GANs

- Backprop through  $k$  updates of the discriminator to prevent mode collapse:
- 

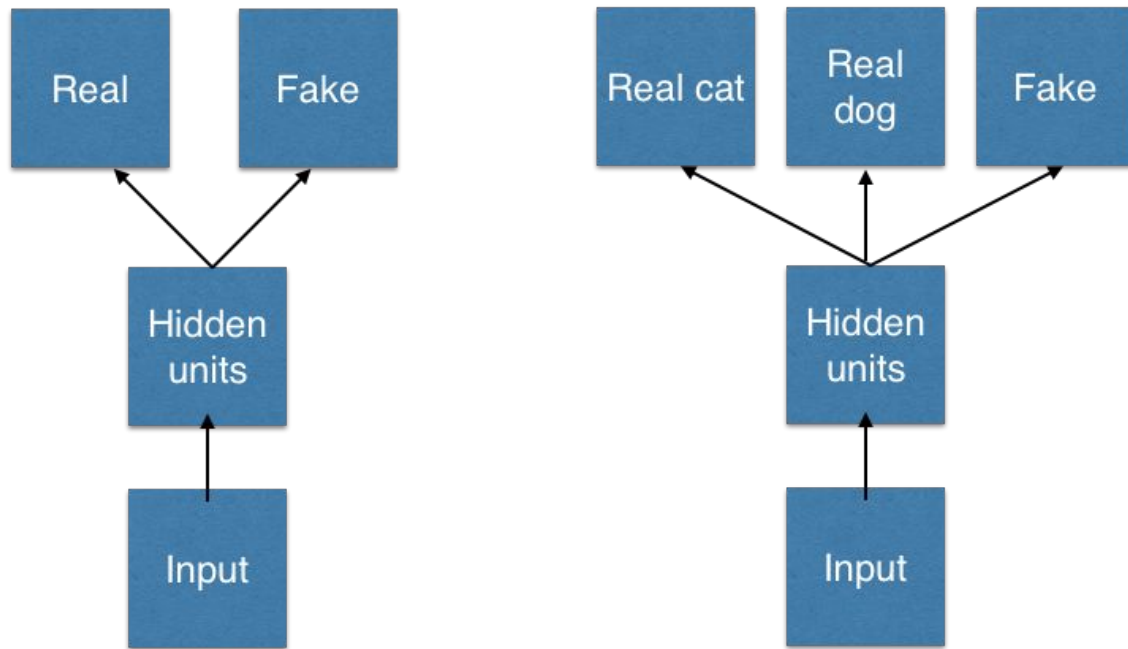


(Metz et al 2016)

# Evaluation

- Никто не понимает до конца, как обучать GANs
- Хорошее правдоподобие и плохие семплы
- Хорошие семплы и плохое правдоподобие
- Нету способа понять, насколько хороши семплы

# Supervised Discriminator



(Odena 2016, Salimans et al 2016)

# Интерпретируемое скрытое представление



(a) Azimuth (pose)

(b) Elevation



(c) Lighting

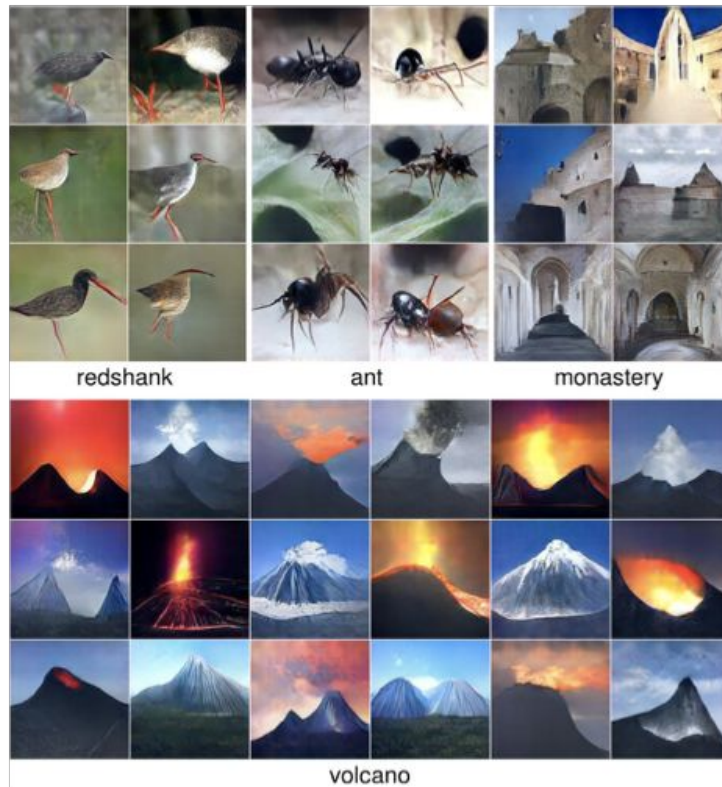
(d) Wide or Narrow

InfoGAN (Chen et al 2016)



State of the art

# PPGN Samples



(Nguyen et al 2016)

# PPGN for caption to image



oranges on a table next to a liquor bottle

(Nguyen et al 2016)

# Заключение

- GAN'ы — это сложно, но работает
- Много нерешённых задач (you are welcome!)