# Wasserstein Generative Adversarial Networks

Цыпин Артем Андреевич
ВМК МГУ

2018/05/24

# Содержание

# Постановка задачи

Одной из задач в обучении без учителя является восстановление вероятностного распределения.

Зачастую восстанавливают плотность распределения, определяя параметрическое семейство плотностей и максимизируя логарифм правдоподобия.

$$\max_{\theta \in \mathbb{R}^d} \frac{1}{m} \sum_{i=1}^{m} \log P_\theta(x^{(i)})$$

# Проблемы в традиционном подходе

Если предположить, что реальное распределение на данных $\mathbb{P}_r$ имеет плотность, а $\mathbb{P}_\theta$ – распределение, отвечающее плотности $P_\theta$, то эта задача сводится к минимизации KL-дивергенции между двумя распределениями.

$$\min_{\mathbb{P}_\theta} KL(\mathbb{P}_r || \mathbb{P}_\theta)$$

В пространствах высокой размерности маловероятно, что носитель реального распределения данных имеет не пустое пересечение с множеством значений, принимаемых моделью. Это ведет к тому, что KL-дивергенция может быть не определенной (или бесконечной).

## Решение проблемы

Для решения описанной проблемы в модель можно добавлять шум (ухудшает качество).

Гораздо удобнее определить некую случайную величину $Z$ с фиксированным распределением $p(z)$ и подавать её на вход параметризованной функции $g_\theta$, которая будет генерировать примеры из распределения $\mathbb{P}_\theta$.

Такой подход используется в VAE и GAN моделях.

# GAN

Итак, пусть $\mathcal{X}$ – компакт в метрическом пространстве (например пространство изображений $[0,1]^d$), а $x$ – случайная величина определенная на $\mathcal{X}$.

Пусть задана функция $D : \mathcal{X} \to (0,1)$, называемая дискриминатором. Она принимает на вход объект из $\mathcal{X}$ и возвращает вероятность того, что объект из $\mathbb{P}_r$.

Пусть также задана функция $G : \mathcal{Z} \times \mathbb{R}^d \to \mathcal{X}$, называемая генератором, которая параметризована $\theta \in \mathbb{R}^d$. Она принимает на вход случайную величину $Z$ и возвращает объект из $\mathcal{X}$.

# Обучение GAN

Будем обучать GAN итеративно.
На каждой итерации сначала обучим дискриминатор отличать настоящие объекты от сгенерированных:

$$D_k = \arg\max_D \mathbb{E}_{x\sim p(x)} \log D(x_i) + \mathbb{E}_{z\sim p(z)} \log\left(1 - D(G_{k-1}(z_i))\right)$$

А затем обучим генератор «обманывать» дискриминатор:

$$G_k = \arg\max_G \mathbb{E}_{z\sim p(z)} \log D_k(G(z))$$

## Равновесие Нэша

Задачу обучения GAN можно переформулировать следующим образом:

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

Процесс сходится, когда $D(x) = \frac{1}{2}$ для любого $x$. Точка равновесия называется равновесием Нэша.

# Проблемы при обучении GAN

- Нужно выдерживать баланс между обучением генератора и дискриминатора.
- Обучение не устойчиво и сильно зависит от архитектуры нейросетей для $D$ и $G$.
- Коллапс моды.
- Отсутсвие функции потерь, значение которой коррелирует с качеством.

## Расстояние между распределениями

Посмотрим на проблему восстановления распределения с другой стороны и попробуем минимизировать расстояние между $\mathbb{P}_r$ и $\mathbb{P}_\theta$.

Последовательность распределений $(\mathbb{P}_t)_{t\in\mathbb{N}}$ сходится тогда и только тогда, когда существует распределение $\mathbb{P}_\infty$, такое что $\rho(\mathbb{P}_t, \mathbb{P}_\infty)$ стремится к нулю.

Понятно, что сходимость распределений сильно зависит от функции $\rho$, определяющей расстояние между распределениями.

## Оптимизационная задача

Хотим оптимизировать параметр $\theta$, исходя из некоторой функции потерь:

$$\theta \mapsto \rho(\mathbb{P}_\theta, \mathbb{P}_r)$$

Это эквивалентно условию, что $\mathbb{P}_\theta$ непрерывно зависит от параметра $\theta$, то есть если $\theta_n \to \theta$, то $\mathbb{P}_{\theta_n} \to \mathbb{P}_\theta$.

# Различные расстояния

- *Kullback-Leibler* (KL) дивергенция

$$KL(\mathbb{P}_r \| \mathbb{P}_g) = \int \log \left( \frac{P_r(x)}{P_g(x)} \right) P_r(x) d\mu(x)$$

- *Jensen-Shannon* (JS) дивергенция

$$JS(\mathbb{P}_r, \mathbb{P}_g) = KL(\mathbb{P}_r \| \mathbb{P}_m) + KL(\mathbb{P}_g \| \mathbb{P}_m)$$

где $\mathbb{P}_m = (\mathbb{P}_r + \mathbb{P}_g)/2$

- *Earth-Mover* (EM) или Wasserstein-1 расстояние

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma} \left[ \|x - y\| \right] \qquad (1)$$

# Теорема 1

## Theorem

*Пусть $\mathbb{P}_r$ – распределение на $\mathcal{X}$. Пусть $Z$ – случайная величина определенная на другом пространстве $\mathcal{Z}$. Пусть $g : \mathcal{Z} \times \mathbb{R}^d \to \mathcal{X}$ – функция, обозначаемая $g_\theta(z)$ с $z$ в качестве первого аргумента и $\theta$ в качестве второго. Пусть $\mathbb{P}_\theta$ – распределение $g_\theta(Z)$. Тогда,*

1. *Если $g$ непрерывна по $\theta$, то и $W(\mathbb{P}_r, \mathbb{P}_\theta)$ непрерывно по $\theta$.*
2. *Если $g$ – локально Липшицева и удовлетворяет предположению о регулярности, то $W(\mathbb{P}_r, \mathbb{P}_\theta)$ везде непрерывна, и дифференцируема почти всюду.*
3. *Пункты 1-2 не выполняются для Jensen-Shannon дивергенции $JS(\mathbb{P}_r, \mathbb{P}_\theta)$ и для KL дивергенции.*

# Теорема 2

## Theorem

*Пусть $\mathbb{P}$ – распределение в $\mathcal{X}$ и $(\mathbb{P}_n)_{n\in\mathbb{N}}$ – последовательность распределений на $\mathcal{X}$. Тогда,*

1. *Следующие утверждения эквивалентны*
   - $\delta(\mathbb{P}_n, \mathbb{P}) \to 0$ *где* $\delta$ *– расстояние полной вариации (total variation).*
   - $JS(\mathbb{P}_n, \mathbb{P}) \to 0$ *где JS – Jensen-Shannon дивергенция.*
2. *Следующие утверждения эквивалентны*
   - $W(\mathbb{P}_n, \mathbb{P}) \to 0$.
   - $\mathbb{P}_n \xrightarrow{\mathcal{D}} \mathbb{P}$ *где* $\xrightarrow{\mathcal{D}}$ *обозначает сходимость по распределению для случайных величин.*
3. *Из* $KL(\mathbb{P}_n \| \mathbb{P}) \to 0$ *или* $KL(\mathbb{P} \| \mathbb{P}_n) \to 0$ *следует пункт (1).*
4. *Из пункта (1) следует пункт (2).*

## Расстояние Вассерштейна

В силу того, что в классическом определении расстояние Вассерштейна сложно подсчитать, введем для него альтернативное определение.

$$W(\mathbb{P}_r, \mathbb{P}_\theta) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x \sim \mathbb{P}_r}[f(x)] - \mathbb{E}_{x \sim \mathbb{P}_\theta}[f(x)]$$

Изменение класса функций на K-Липшицивые оставляет оценку расстояния Вассерштейна одинаковым с точностью до мультипликативной константы.

## Расстояние Вассерштейна

Итак, для нахождения функции $f_w$, играющей роль дискриминатора (критика), предлагается решить следующую задачу.

$$\max_{w \in \mathcal{W}} \mathbb{E}_{x \sim \mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z \sim p(z)}[f_w(g_\theta(z)]$$

Если функция-генератор удовлетворяет предположению о регулярности, то решение существует и выполнено следующее равенство.

$$\nabla_\theta W(\mathbb{P}_r, \mathbb{P}_\theta) = -\mathbb{E}_{z \sim p(z)}[\nabla_\theta f(g_\theta(z))]$$

.

# Обучение критика

Будем обучать критика аналогично дискриминатору в модели GAN. Пусть есть нейросеть с параметрами $w \in \mathcal{W}$, где $\mathcal{W}$ – компакт.

Этот факт позволяет утверждать, что $f_w$ является K-Липшицевой.

Для того, чтобы выполнить условие компактности будем «обрезать» веса до некоторого значения $\mathcal{W} = [-0.01, 0.01]^l$.

## Алгоритм

*Require*: $\alpha$, learning rate. $c$, параметр клиппинга. $m$, размер батча. $n_{\text{critic}}$, количество итераций критика на одну итерацию генератора, $w_0$, начальные параметры критика. $\theta_0$, начальные параметры генератора.

**while** $\theta$ **не сошлось**:

  **for** $t = 0, ..., n_{\text{critic}}$:

    1: Сэмплируем $x^{(i)}{}_{i=1}^{m} \sim \mathbb{P}_r$ батч из реальных данных.

    2: Сэмплируем $z^{(i)}{}_{i=1}^{m} \sim p(z)$ батч случайных величин.

    3: $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right]$

    4: $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$

    5: $w \leftarrow \text{clip}(w, -c, c)$

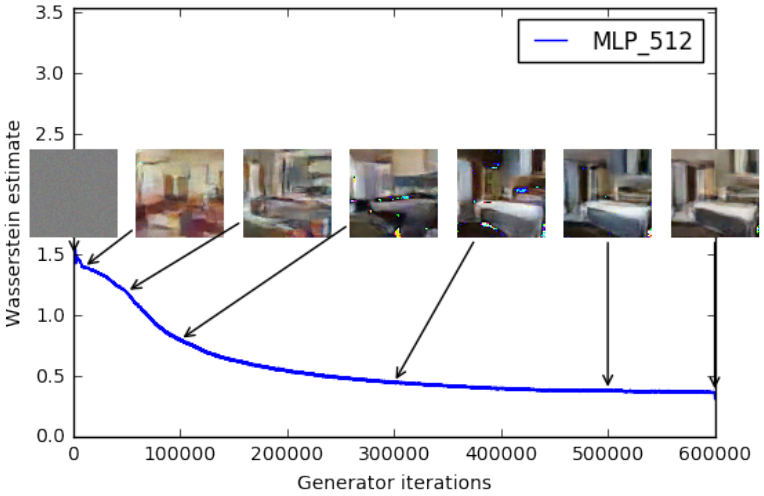
  **end for**

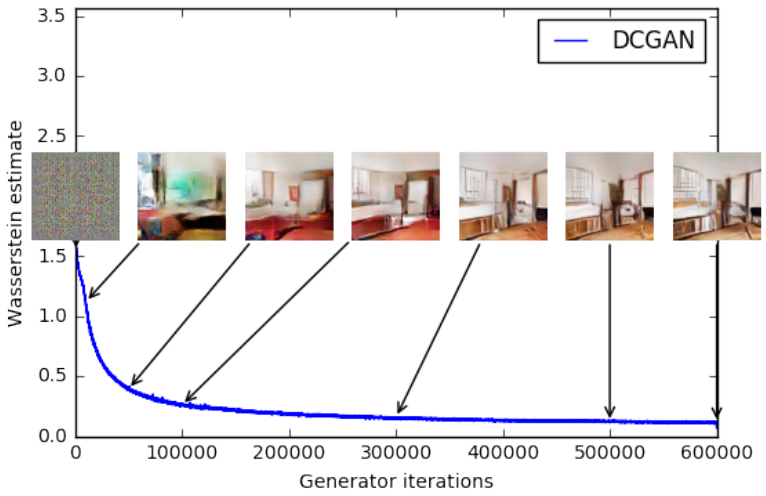
  6: Сэмплируем $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ батч случайных величин.

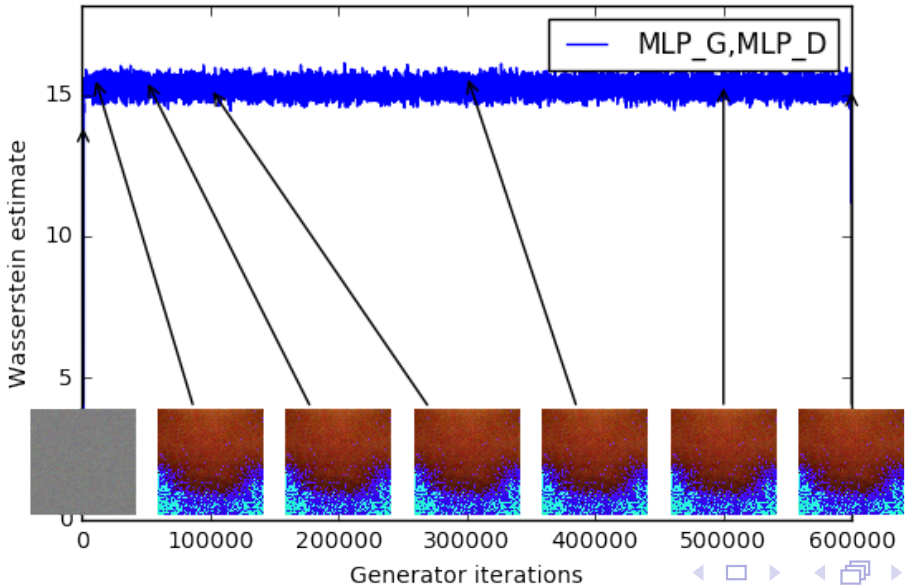
  7: $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))$

  8: $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$

# Корреляция функции потерь с качеством



## Сравнение WGAN и GAN

Функция, оптимизируемая при обучении дискриминатора:

$$L(D, g_\theta) = \mathbb{E}_{x\sim\mathbb{P}_r}[\log D(x)] + \mathbb{E}_{x\sim\mathbb{P}_\theta}[\log(1 - D(x))]$$

Эта функция является нижней оценкой следующей функции расстояния между распределениями:

$$2JS(\mathbb{P}_r, \mathbb{P}_\theta) - 2\log 2$$

При этом значение функции потерь дискриминатора никак не коррелирует с качеством.

# Сравнение WGAN и GAN

Тот факт, что критик в модели WGAN тренируется до оптимальности позволяет избежать сразу нескольких недостатков модели GAN:

- Не нужно соблюдать баланс между обучением критика и генератора.
- Не нужно аккуратно настраивать архитектуру генератора и критика.
- Коллапса моды удается избежать.

# Пример

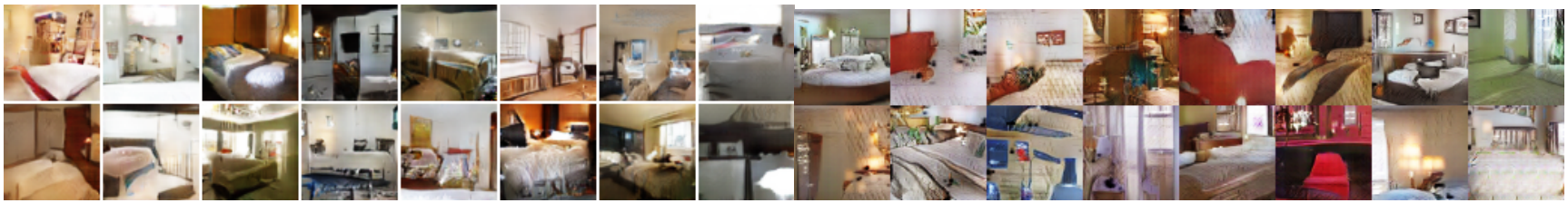


**Рис.:** Алгоритмы с DCGAN генератором. Слева: WGAN. Справа: обычный GAN.

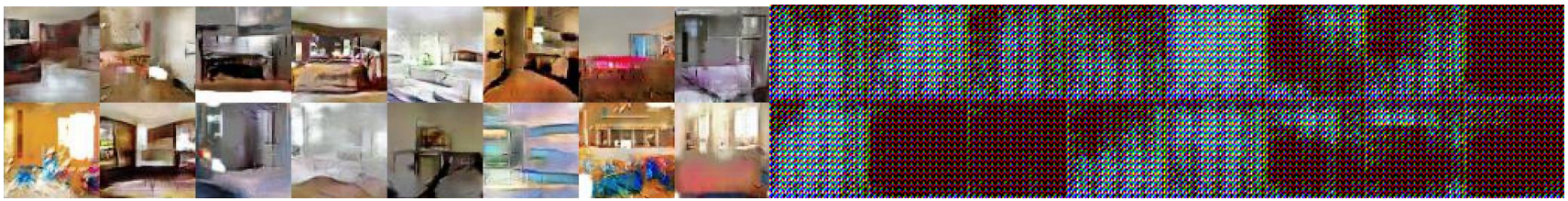


**Рис.:** Генератор без батч-нормализации с постоянным количеством фильтров. Слева: WGAN. Справа: обычный GAN. .

## Предположение регулярности и ссылки

Пусть $g : \mathcal{Z} \times \mathbb{R}^d \to \mathcal{X}$ – локально Липшицева. Будем говорить, что $g_\theta(z)$ удовлетворяет предположению о регулярности для некоего распределения $\mathcal{Z}$, если существует Липшицева константа $L(\theta, z)$, такая что:

$$\mathbb{E}_{z \sim p}[L(\theta, z)] < +\infty$$

Ссылки:

https://arxiv.org/pdf/1701.07875.pdf

https://habr.com/post/352794/