Прикладная статистика в машинном обучении
Лекция 12
Генеративные модели

И. К. Козлов
(*Мехмат МГУ*)

2022

Нельзя объять необъятное

Сегодня мы поговорим про генеративные модели.

Это гигантская тема. Мы буквально одним глазком посмотрим на некоторые модели.

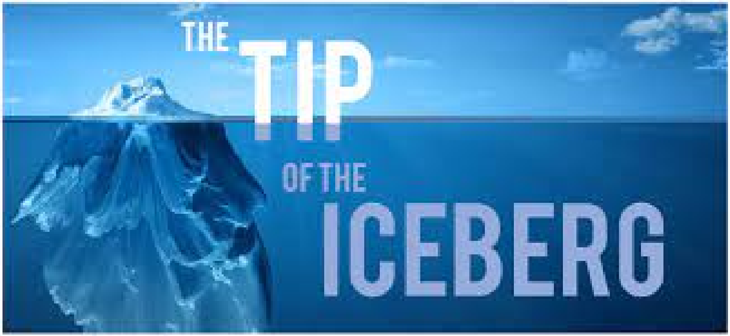


Рис.: Не надейтесь, даже столько не узнаем

NF и Diffusion Models

Мы обсудим Auto-Encoders и скажем пару слов про GAN.

Есть много других популярных моделей, например

- **Normalizing Flows**

  https://lilianweng.github.io/posts/2018-10-13-flow-models/

- **Diffusion Models**

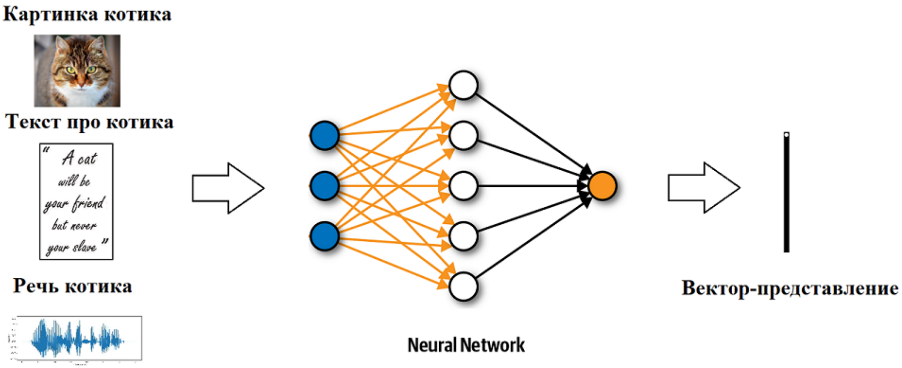
  https://lilianweng.github.io/lil-log/2021/07/11/diffusion-models.html

AE

Векторизация

Самые лучшие данные для компьютера — это векторы (или массивы).

Нейронки — по сути эффективный способ векторизации сложных (однородных) данных, таких как изображения, текст, речь.



Сжатие без потерь

Отображение в вектор — по сути “сжатие информации”.

По идее мы хотим сжать информацию как можно компактней, но при этом не потерять ничего нужного.

Q: А как это проверить — что мы “не потеряли ничего нужного”?

Сжатие без потерь

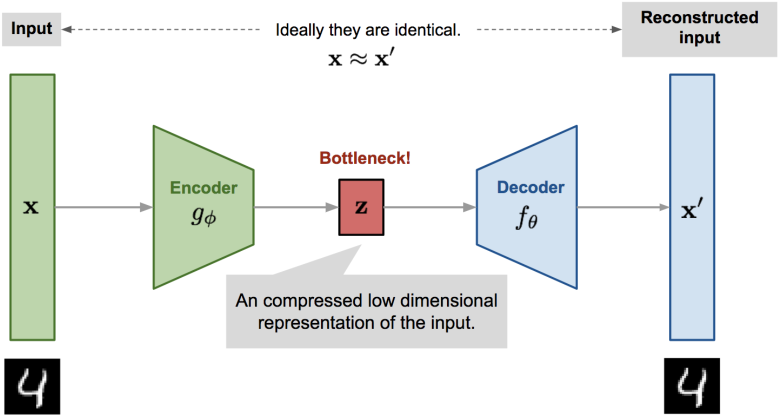
Отображение в вектор — по сути “сжатие информации”.

По идее мы хотим сжать информацию как можно компактней, но при этом не потерять ничего нужного.

Q: А как это проверить — что мы “не потеряли ничего нужного”?

A: Мы не потеряли информацию, если мы можем восстановить исходное изображение.

Так возникает идея автокодировщика (Autoencoder)



AutoEncoder

Как нейронная сеть автокодировщик — это пара нейронных сетей:

- Энкодер $f : x \to z$;
- Декодер $g : z \to x'$,

которые обучаются, минимизируя выбранный лосс

$$L(x, g(f(x))).$$

Пример AE

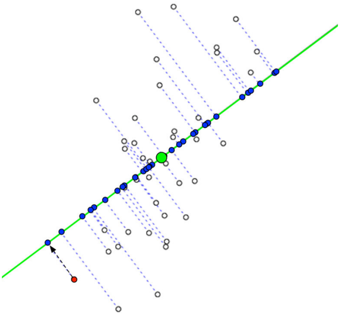
Q: Какое отображение выучит AE с

- MSE-лоссом и
- линейным декодером?

Пример AE

A: Это PCA:

- образ — линейное пространство т.,ч.
- сумма квадратов расстояний до образов точек минимальна.



Литература

Про AE см. Главу 14

Goodfellow C.M.
Pattern Recognition and Machine Learning.

Путь к BERT начинается с циферек в MNIST

Настало время реализовать нашу первую нейронку:
автоэнкодер для цифр из MNIST.

Откройте Jupyter-Notebook к семинару.

Рис.: Учимся считать. ☺

Модификации AE

Когда памяти слишком много

Возможная проблема с Auto-Encoder:

Q: Пусть $\dim z \geq \dim x$. Что может выучить авто-энкодер?

Когда памяти слишком много

Возможная проблема с Auto-Encoder:

Q: Пусть $\dim z \geq \dim x$. Что может выучить авто-энкодер?

A: Тождественное отображение

$$\mathrm{id}(x) = x.$$

Ключевые проблемы AE.

1. **У латентного представления z нет никаких хороших свойств.**
2. **Переобучение.** AE может заучить объекты x_i. Например, отображения
$$x_i \to i, \qquad i \to x_i$$
дадут идеальный, но абсолютно бесполезный AE
$$\mathrm{id}(x_i) = x_i.$$
Не знаем, как AE поведёт себя на других объектах $x_{new} \neq x_i$.
3. **Нельзя сэмплировать.**

Модификации AE

Стандартные модификации AE.

1. Регуляризации для улучшения свойств z (**Sparse AE**, **Contractive AE**).
2. Зашумление данных, чтобы избежать переобучения (**Denoising AE**)
3. Модификация для возможности сэмплирования (Variational AE).

Стандартные модификации AE.

1. Регуляризации для улучшения свойств z (**Sparse AE**, **Contractive AE**).
2. Зашумление данных, чтобы избежать переобучения (**Denoising AE**)
3. Модификация для возможности сэмплирования (Variational AE).

2ая и 3ья идея — более практичны, 1ая — скорее факт истории.

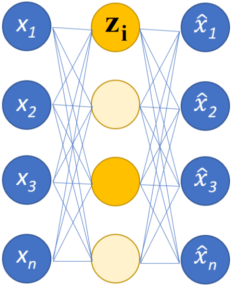
Sparse AutoEncoder.

Идея. *Чем разреженней представления z, чем дальше образы объектов друг от друга, тем лучше.*

Можно выучить много фичей $z_1, \dots, z_N$.

Но для каждого объекта x_j требуем большинство $z_i = 0$.

AE активирует лишь "важнейшие нейроны" z_i для каждого объекта x_j.



Sparse AutoEncoder.

На практике в лосс добавляется штраф на z:

$$L(x, x') + \Omega(z)$$

- Например, L_1-лосс $\Omega(z) = \lambda \sum_i |z_i|$;
- Или KL-дивергенция, чтобы сделать вероятность $z_i \neq 0$ близкой к малому p.

  https://lilianweng.github.io/posts/2018-08-12-vae/#sparse-autoencoder
- Или можно явно занулить все z_i кроме TOP-k.

Contractive AE

Contractive AutoEncoder.

Идея. *Требуем, чтобы z_i не сильно менялись при малом шевелении x.*

Добавляем в лосс регуляризацию:

$$L(x, x') + \lambda \sum_i ||\nabla_x z_i||^2.$$

Denoising AutoEncoder.

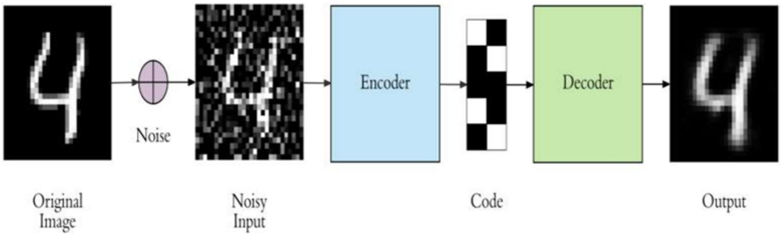
Идея. *Зашумляем x, чтобы модель не зазубривала тупо объекты.*

Добавляем к объектам шум $x \to \hat{x}$.

Требуем, чтобы AE правильно восстанавливал x по зашумлённому $\hat{x}$:

$$g(f(\hat{x})) \approx x.$$

Минимизируем лосс $L(x, g(f(\hat{x})))$.



VAE

Variational AutoEncoder

Обсудим вариационный автокодировщик (VAE).

Про VAE и его модификации можно почитать в:

- Глава VAE в https://ml-handbook.ru (доступно ШАД).
- https://lilianweng.github.io/posts/2018-08-12-vae/
- Главу 20.10.3

Goodfellow C.M.
Pattern Recognition and Machine Learning.

Хотим генерить новые объекты

Проблема. Из AE неудобно сэмплировать.

Q: Откуда мы умеем сэмплировать?

Хотим генерить новые объекты

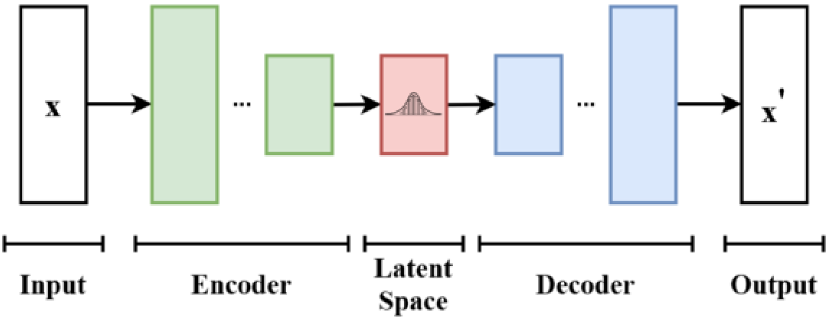
Проблема. Из AE неудобно сэмплировать.

Q: Откуда мы умеем сэмплировать?

A: Из распределения $p(x)$.

Вариационный автокодировщик (Variational Autoencoder)

Отличие от AE: на выходе энкодера и декодера — распределения.



VAE лишь внешне похож на AE

Несмотря на схожесть с AE, VAE — абсолютно “байесовский зверь” и скорее результат “параллельной эволюции”.

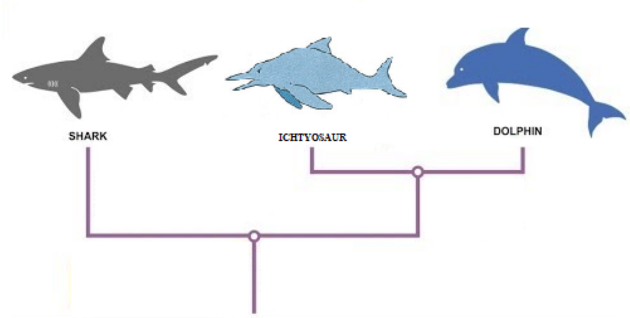


Рис.: Параллельная эволюция

Как байесиане дошли до жизни такой?

Вспомним модели с латентными переменными.

- x — наблюдаемые переменные;
- z — латентные переменные;
- θ — параметры модели.

Мы знаем x, не знаем z и ищем θ.

Совместное распределение обозначим через $p_\theta(x, z)$.

Процедура сэмплирования

Как сэмплировать объекты x? По правилу произведения

$$p_\theta(x, z) = p_\theta(x \mid z) p_\theta(z).$$

Процедура сэмплирования

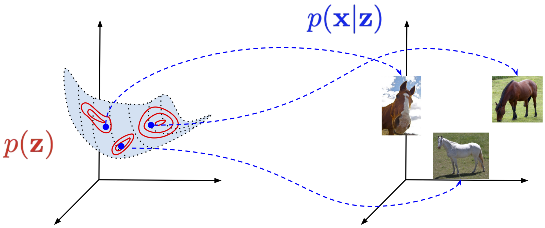
Как сэмплировать объекты x? По правилу произведения

$$p_\theta(x, z) = p_\theta(x \mid z) p_\theta(z).$$

- Сэмплируем **латентный вектор** z из $p_\theta(z)$.

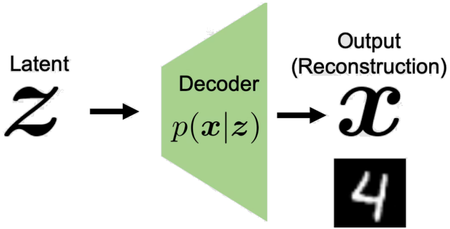
  Неформально — мы фиксируем "факторы для генерации" (цвет, размер и т.д.)

- Сэмплируем **объекты** x из $p_\theta(x \mid z)$.



Процедура сэмплирования

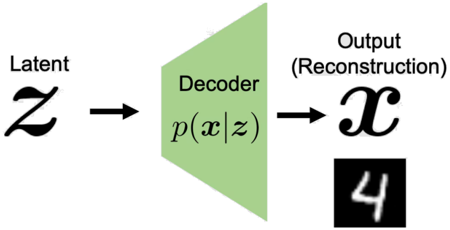
В VAE за сэмплирование отвечает декодер.



Q: Как найти параметр модели θ?

Процедура сэмплирования

В VAE за сэмплирование отвечает декодер.



Q: Как найти параметр модели θ?

A: Естественная оценка — максимум правдоподобия на наблюдаемых данных:

$$\theta^* = \arg\max_{\theta} p_\theta(x)$$

Маргинализация

Нам нужно вычислить $p_\theta(x)$. Это сложное распределение.

Q: Как перейти к более простому совместному распределению $p_\theta(x, z)$?

A: Маргинализуем по z:

$$p_\theta(x) = \int p_\theta(x, z) dz = \int\limits_{Z^M} p_\theta(x|z) p_\theta(z) dz.$$

Монте-Карло

Итак, нам нужно посчитать матожидание

$$p_\theta(x) = \int\limits_{Z^M} p_\theta(x|z)p_\theta(z)dz = \mathbb{E}_{z\sim p_\theta(z)}[p_\theta(x|z)].$$

Q: Как можно оценить матожидание?

A: Методом Монте-Карло:

$$\mathbb{E}_{z\sim p_\theta(z)}[p_\theta(x|z)] \approx \frac{1}{K}\sum_k p_\theta(x|z_k),$$

где z_k сэмплируются из $p_\theta(z)$.

Проклятие размерности

Q: В чём проблема с суммой

$$\mathbb{E}_{z \sim p_\theta(z)}[p_\theta(x|z)] \approx \frac{1}{K} \sum_k p_\theta(x|z_k)?$$

Проклятие размерности

Q: В чём проблема с суммой

$$\mathbb{E}_{z \sim p_\theta(z)}[p_\theta(x|z)] \approx \frac{1}{K} \sum_k p_\theta(x|z_k)?$$

A: Проклятие размерности.

Нужно много сэмплов, чтобы покрыть пространство z.

При этом вклад почти всех z_k в $p_\theta(x \mid z)$ — около 0.

Encoder

Идея. *Для каждого* x *брать те* z_k, *где велико апостериорное распределение*

$$p_\theta(z \mid x) \propto p_\theta(x|z)p_\theta(z).$$

Апостериорное распределение трудно вычислимо. Поэтому аппроксимируем его

$$q_\phi(z \mid x) \approx p_\theta(z \mid x).$$

Это делает энкодер.

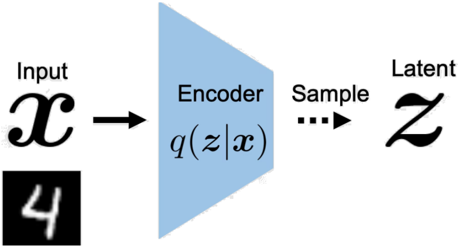
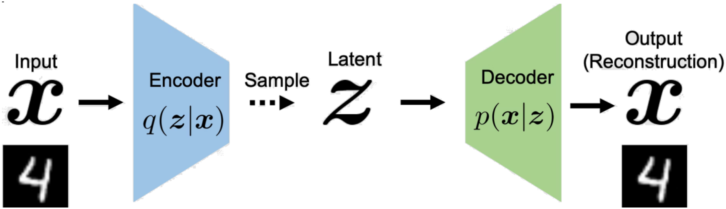


Схема VAE

Вот мы и получили схему VAE:



Обучение VAE

Как обучать VAE?

Энкодер и декодер — это нейронные сети. Нужно найти их параметры ϕ и θ.

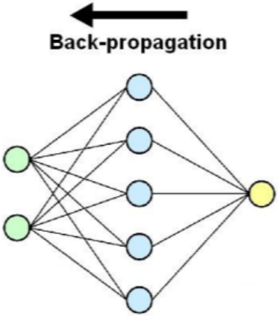
Q: Как находятся веса нейронных сетей?

Как обучать VAE?

Энкодер и декодер — это нейронные сети. Нужно найти их параметры ϕ и θ.

Q: Как находятся веса нейронных сетей?

A: Градиентным спуском, оптимизируя функцию потерь $\mathcal{L}$.



Функция потерь

Итак, нам нужно найти:

1. адекватную задаче функцию потерь $\mathcal{L}$,
2. которую можно дифференцировать по ϕ и θ,
3. и которую несложно посчитать.

ELBO

В качестве $\mathcal{L}$ возьмём вариационную нижнюю оценку (ELBO)

$$\mathcal{L}_{\theta,\phi}(x) = \log p_\theta(x) - \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x))$$

с прошлой лекции по EM-алгоритму.

Напомним необходимые определения.

KL-дивергенция

KL-дивергенция

Пусть P и Q — распределения с плотностями $p(x)$ и $q(x)$.

Дивергенция Кульбака–Лейблера суть

$$\mathrm{KL}(P \parallel Q) = \int p(x) \ln \frac{p(x)}{q(x)} \, \mathrm{d}x.$$

KL-дивергенция несимметрична и неотрицательна

$$\mathrm{KL}(P \parallel Q) \neq \mathrm{KL}(Q \parallel P), \qquad \mathrm{KL}(P \parallel Q) \geq 0.$$

Две KL-дивергенции

Пусть P — известное распределение, мы ищем близкое к нему Q.

Q: *А что оптимизировать* — $\mathrm{KL}(P \parallel Q)$ *или* $\mathrm{KL}(Q \parallel L)$?

Посмотрим на их свойства.

Forward KL

Стремимся минимизировать

$$\mathrm{KL}(P \parallel Q) = \int p(x) \ln \frac{p(x)}{q(x)} \, \mathrm{d}x$$

Q: Когда будет большое выражение под интегралом?

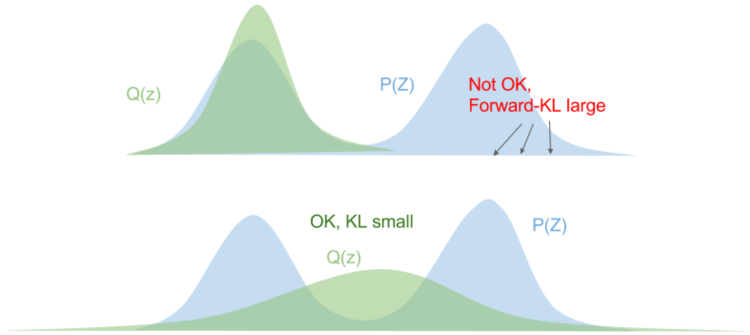
Forward KL

Стремимся минимизировать

$$\mathrm{KL}(P \parallel Q) = \int p(x) \ln \frac{p(x)}{q(x)} \, \mathrm{d}x$$

Q: Когда будет большое выражение под интегралом?

A: Если $p \neq 0, q = 0$. Значит, Q будет "размазываться" по всему носителю P.



Reversed KL

Стремимся минимизировать

$$\mathrm{KL}(Q \parallel P) = \int q(x) \ln \frac{q(x)}{p(x)} \, \mathrm{d}x$$

Q: Когда будет большое выражение под интегралом?

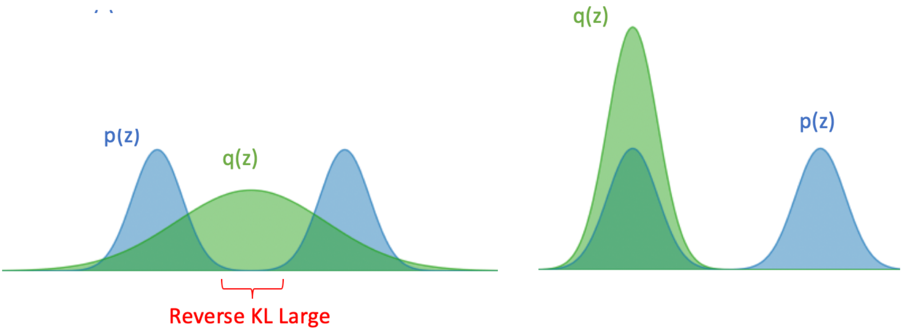
Reversed KL

Стремимся минимизировать

$$\mathrm{KL}(Q \parallel P) = \int q(x) \ln \frac{q(x)}{p(x)} \, dx$$

Q: Когда будет большое выражение под интегралом?

A: Если $p = 0, q \neq 0$. Значит, Q будет "локализовываться" близ моды P.



ELBO

ELBO

Вспомним факт, который мы доказали на прошлой лекции.

$$\ln p_\theta(X) = \mathcal{L}_{\theta,\phi}(x) + \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x)),$$

где 1ое слагаемое — это ELBO

$$\mathcal{L}_{\theta,\phi}(x) = \mathbb{E}_{q_\phi(z|x)} \ln \frac{p_\theta(z|x)}{q_\phi(z|x)}.$$

Вывод ELBO

Напомним доказательство. Для краткости $q_\phi(z|x) = q(z)$.

$$\log p_\theta(x) = \mathbb{E}_{q(z)}[\log p_\theta(x)] = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{p_\theta(z|x)}\right)\right]$$

Вывод ELBO

Напомним доказательство. Для краткости $q_\phi(z|x) = q(z)$.

$$\log p_\theta(x) = \mathbb{E}_{q(z)}[\log p_\theta(x)] = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{p_\theta(z|x)}\right)\right]$$

Единственный трюк — домножить числитель и знаменатель на $q(z)$:

$$\mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{q(z)}\frac{q(z)}{p_\theta(z|x)}\right)\right] = \underbrace{\mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{q(z)}\right)\right]}_{ELBO=\mathcal{L}_{\theta,\phi}(x)} + \underbrace{\mathbb{E}_{q(z)}\left[\log\left(\frac{q(z)}{p_\theta(z|x)}\right)\right]}_{\mathrm{KL}(q(z)\|p_\theta(z|x))}$$

Разложение ELBO

Посмотрим на ELBO повнимательней.

- Воспользуемся правилом произведения

$$ELBO = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{q(z)}\right)\right] = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x \mid z)p_\theta(z)}{q(z)}\right)\right]$$

Разложение ELBO

Посмотрим на ELBO повнимательней.

- Воспользуемся правилом произведения

$$ELBO = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x,z)}{q(z)}\right)\right] = \mathbb{E}_{q(z)}\left[\log\left(\frac{p_\theta(x \mid z)p_\theta(z)}{q(z)}\right)\right]$$

- выделим ещё KL-дивергенцию с априорным распределением:

$$ELBO = \underbrace{\mathbb{E}_{q(z)}\left[\log p_\theta(x|z)\right]}_{\text{Reconstruction Loss}} - \underbrace{\mathrm{KL}(q(z) \parallel p_\theta(z))}_{\text{Regularization Term}}$$

Функция потерь для VAE

Всё, мы доказали самую важную теорему про VAE.

Теорема

$$\ln p_\theta(X) - \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x)) = \mathcal{L}_{\theta,\phi}(x),$$

где справа стоит ELBO

$$\mathcal{L}_{\theta,\phi}(x) = \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]}_{\text{Reconstruction Loss}} - \underbrace{\mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z))}_{\text{Regularization Term}}.$$

Эту функцию мы и возьмём за функцию потерь для VAE.

Максимизируем правдоподобие и аппроксимируем апостериорное

Q: Какие слагаемые стоят слева?

$$\ln p_\theta(X) - \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x))$$

A: Мы максимизируем ELBO. Поэтому:

- Мы максимизируем правдоподобие данных $\ln p_\theta(X)$,

Максимизируем правдоподобие и аппроксимируем апостериорное

Q: Какие слагаемые стоят слева?

$$\ln p_\theta(X) - \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x))$$

A: Мы максимизируем ELBO. Поэтому:

- Мы максимизируем правдоподобие данных $\ln p_\theta(X)$,
- Мы минимизируем KL-дивергенцию $\mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z|x))$, т.е. приближаем $q_\phi(z|x)$ к апостериорному распределению $p_\theta(z|x)$.

Reconstruction and Regularization

Слагаемые справа тоже можно интерпретировать.

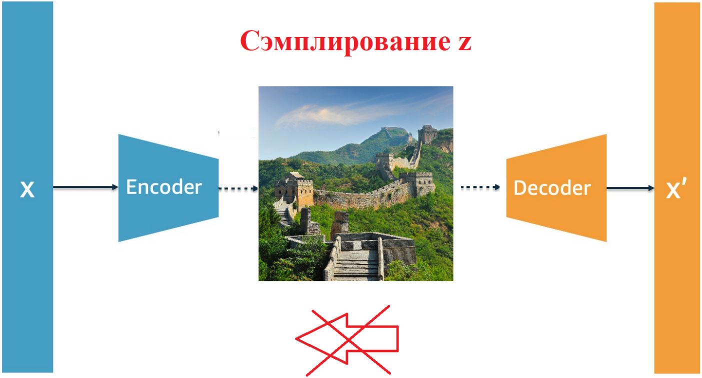
$$\mathcal{L}_{\theta,\phi}(x) = \underbrace{\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]}_{\text{Reconstruction Loss}} - \underbrace{\mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z))}_{\text{Regularization Term}}.$$

- Reconstruction Loss — ошибка при реконструкции объекта.
- Regularization Term — штраф за отклонение от априорного распределения.

Reparameterization Trick

Градиент не течёт

Есть проблема! У нас не течёт градиент!



Операция сэмплирования недифференцируема

В чём проблема? ELBO имеет вид

$$\mathcal{L} = \mathbb{E}_{q_\phi(z)} f_\theta(x, z)$$

Интегралы не считаются, поэтому они заменяются Монте-Карло оценкой:

$$\mathcal{L} \approx \frac{1}{K} \sum_{i=1}^{k} f_\theta(x, z_k), \qquad z_k \sim q_\phi(z).$$

Пропала производная по ϕ.

$$\boxed{\text{Операция сэмплирования недифференцируема.}}$$

Reparameterization trick

Идея. Отделим параметры и стохастичность.

Если $X \sim \mathcal{N}(\mu, \sigma)$, то $X = \mu + \sigma \cdot \varepsilon$, где $\varepsilon \sim \mathcal{N}(0, 1)$.

По ε дифференцировать нельзя — это случайная величина.

А по параметрам μ, σ — можно.

Reparameterization trick

В общем случае мы ищем замену

$$z = g(\varepsilon, \phi, x)$$

В матожиданиях в ELBO параметр “уйдёт из плотности в функцию”:

$$\mathbb{E}_{q_\phi(z)}[f_\theta(x, z)] = \mathbb{E}_{p(\epsilon)}[f_\theta(x, g(\varepsilon, \phi, x))].$$

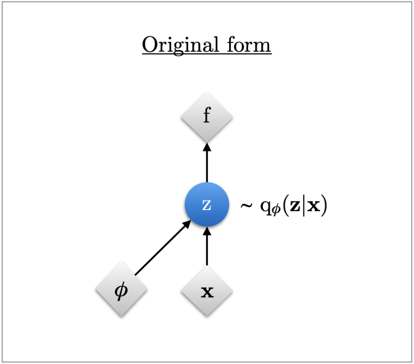
Интегралы можно заменить Монте-Карло оценкой

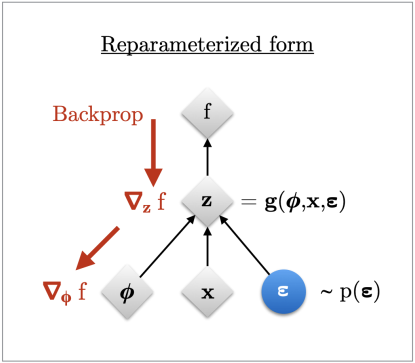
$$\frac{1}{K}\sum_{i=1}^{K} f_\theta(x, g(\varepsilon_k, \phi, x)),$$

которую можно диффенцировать по ϕ и θ.

Reparameterization trick

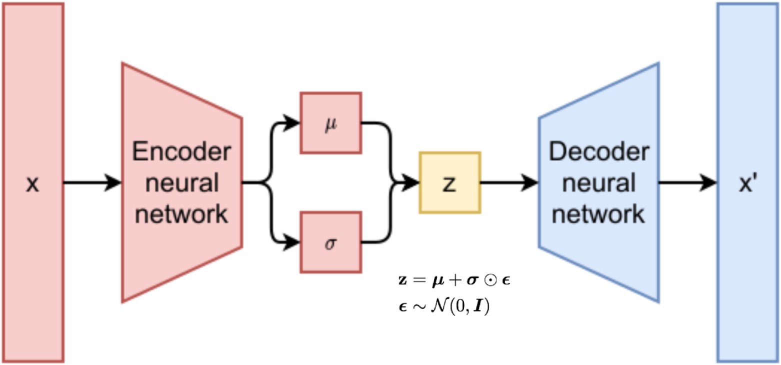
Этот трюк обычно иллюстрируется такой картинкой:





Variational AutoEncoder

Итак, VAE принимает следующий вид:



Пример VAE

Нормальные распределения

Рассмотрим простейший VAE, когда все распределения — нормальные.

- Априорное распределение — берём стандартное нормальное:

$$p_\theta(z) = \mathcal{N}(0, I)$$

- Правдоподобие. Ковариация — постоянная диагональная матрица.

$$p_\theta(x|z) = \mathcal{N}(f_\theta(z), \sigma^2)$$

- Апостериорное распределение. Матрица ковариации — диагональная:

$$q_\phi(z|x) = \mathcal{N}(\mu_\phi(x), \sigma^2_\phi(x))$$

Функция потерь

Посмотрим на ELBO:

$$\mathcal{L}_{\theta,\phi}(x) = \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - \mathrm{KL}(q_\phi(z|x) \parallel p_\theta(z))$$

2ое слагаемое вычисляется явно, 1ое — оценивается методом Монте-Карло:

$$\mathbb{E}_{\mathcal{N}(\mu_\phi(x),\sigma_\phi^2(x))}\left[\log p_\theta(x|z)\right] \approx \frac{1}{K}\sum_{k=1}^{K} \log p_\theta(x|z_k),$$

где $z_k \sim \mathcal{N}(\mu_\phi(x), \sigma_\phi^2(x))$.

Ошибка при реконструкции

Для гауссовской VAE 1ое слагаемое принимает вид

$$\log p_\theta(x|z) = \sum_{j=1}^{D} \log p_\theta(x_j|z) = \sum_{j=1}^{D} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_j - f_{\theta,j}(z))^2}{2\sigma^2}\right)\right) =$$

$$= -\frac{D}{2}\log 2\pi - D\log\sigma - \frac{1}{2\sigma^2}\sum_{j=1}^{D}(x_j - f_{\theta,j}(z))^2$$

Выделено единственное неконстантное слагаемое.

KL-дивергенция

2ое слагаемое — это KL-дивергенция нормальных распределений.

Формулу для него можно найти в https://mr-easy.github.io/2020-04-16-kl-divergence-between-2-gaussian-distributions/

В данном случае

$$\mathrm{KL}(q_\phi(z \| x) \parallel p_\theta(z)) = \mathrm{KL}(\mathcal{N}(\mu_\phi(x), \sigma^2_\phi(x)) \parallel \mathcal{N}(0, I)) =$$
$$= \frac{1}{2}\sum_{j=1}^{M}(\sigma_j^2 + \mu_j^2 - 1 - \ln \sigma_j^2)$$

Итоговый лосс для VAE

$$L_{VAE} = \frac{1}{K}\sum_{k=1}^{K}\left[\frac{1}{2\sigma^2}\sum_{j=1}^{D}(x_j - f_{\theta,j}(z_k))^2\right] + \frac{1}{2}\sum_{j=1}^{M}(\sigma_j^2 + \mu_j^2 - 1 - \ln\sigma_j^2),$$

где

$$z_k \sim \mathcal{N}(\mu_\phi(x), \sigma_\phi^2(x)).$$

Согласно https://arxiv.org/pdf/1312.6114.pdf (Раздел 2.3), если батч достаточно большой ($N \geq 100$), то можно положить $K = 1$.

Недостатки VAE

Размытость

Типичная проблема VAE — размытость изображений.

KL-дивергенция с распределением данных

Это ожидаемо. Максимизация правдоподобия влечёт минимизацию KL-дивергенции между реальными данными и генерируемыми:

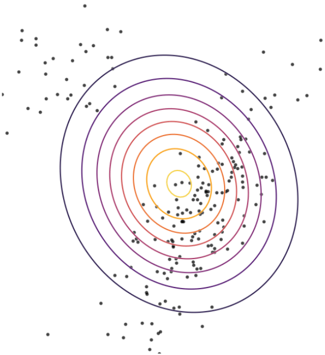
$$\theta = \arg\max_\theta p(x \mid \theta) = \arg\max_\theta \sum_{i=1}^{n} \log p(x_i \mid \theta) \to \arg\max_\theta \int p_{real}(x) \log p(x \mid \theta) dx =$$

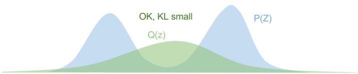
$$= \arg\max_\theta \int p_{real}(x) \left[\log p(x \mid \theta) - \log p_{real}(x)\right] dx = \mathrm{KL}(p_{real} \parallel p(x \mid \theta)).$$

KL-дивергенция с распределением данных

А при минимизации Forward KL
распределение “размазывается” по всей области.

Аппроксимируется всё, но “кое-как”.





VAE-GAN

Им несть числа

Мир генеративных моделей не ограничивается VAE.

Упомянем одну из возможных модификаций — VAE-GAN.

Заодно познакомимся с одной из важнейших генеративных моделей:
GAN (Generative Adversarial Networks).

Mortal Kombat

GAN in a nutshell

Лекция-12

GAN — большая тема, мы не успеем на ней остановиться. Для ознакомления:

https://lilianweng.github.io/posts/2017-08-20-gan/

Про “зоопарк различных GANов”:

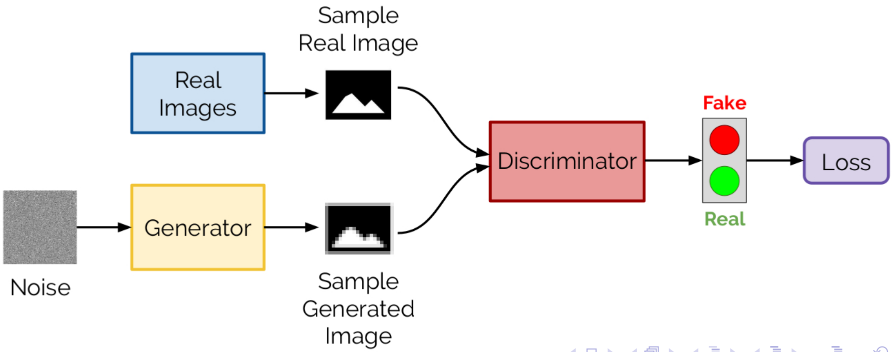
https://neptune.ai/blog/6-gan-architectures

Идея GAN

Как обучить хороший генератор?

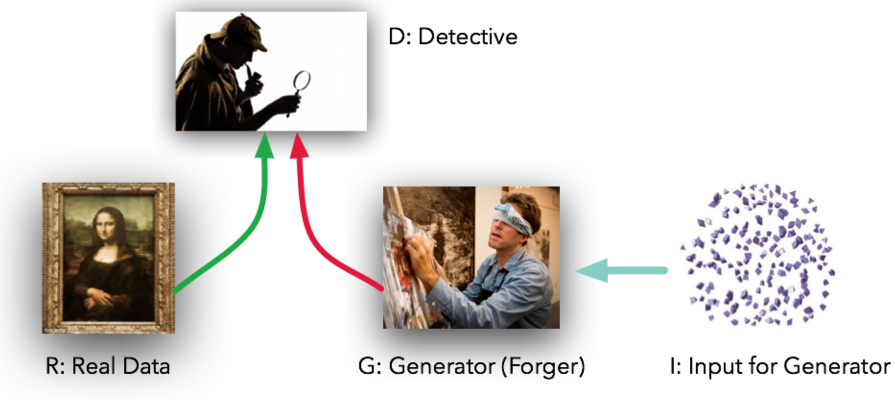
Идея. Одновременно обучать генератор и дискриминатор, отличающий “фейк от правды”.

Если мы обманем дискриминатор, то мы не можем отличить сгенерированные данные от истинных.



Казаки-разбойники

По сути — это противостояние фальшивомонетчиков и полицейских.



GAN loss

В отличие от большинства нейронных сетей,
обучение GAN — это **минимаксная игра**.

Формально, в качестве лосса берётся

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

- 1ое слагаемое требует, чтобы на реальных данных дискриминатор $D = 1$.
- 2ое слагаемое — что на данных генератора $G(x)$ дискриминатор $D = 0$.

JS Divergence

Пусть $p_r(x)$ — распределение реальных данных, а $p_g(x)$ — сгенерированных.

Можно показать (https://lilianweng.github.io/posts/2017-08-20-gan/), что

- идеальный дискриминатор имеет вид

$$D^*(x) = \frac{p_r(x)}{p_r(x) + p_g(x)};$$

- оптимизация лосса при идеальном дискриминаторе суть минимизация дивергенции Йенсена — Шеннона

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

где

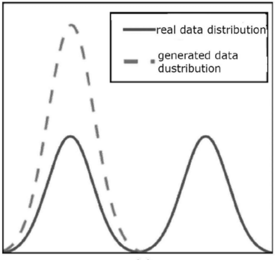
$$\mathrm{JS}(p\|q) = \frac{1}{2}\mathrm{KL}\left(p\Big\|\frac{p+q}{2}\right) + \frac{1}{2}\mathrm{KL}\left(q\Big\|\frac{p+q}{2}\right)$$

Mode Collapsing

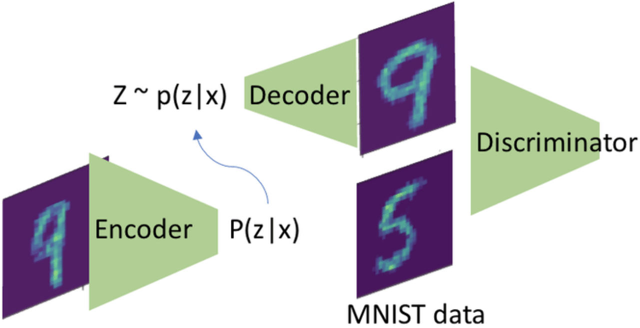
Обучение GAN — трудоёмкое и нетривиальное занятие.

У него есть склонность к mode collapsing.

В противоположность VAE, GANу гораздо выгоднее идеально подделывать небольшое подмножество элементов, чем пытаться сгенерировать всё.



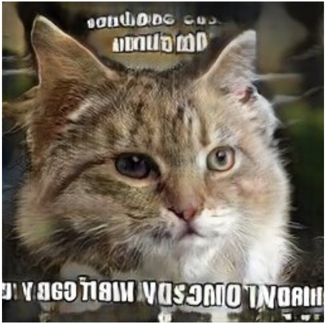
А вот и обещанная схема VAE-GAN:



Котик Goodfellow

Финальный вопрос курса — конечно будет про котика. ☺

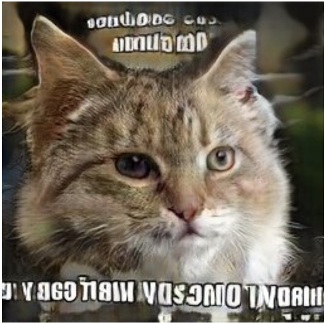
Q: Откуда вязь вокруг кота?



Котик Goodfellow

Финальный вопрос курса — конечно будет про котика. ☺

Q: Откуда вязь вокруг кота?



A: В обучающей выборке были мемасы с котиками.

Эпилог

Читайте книги — источник знаний

Наука не ограничивается университетскими курсами.

То, чему учат в ВУЗах — малая доля того, что знают учёные и написано в статьях и монографиях.

77. Могилевский А.
Если книг читать не будешь — скоро грамоту забудешь. 1925

IT динамично.

Существуют много разных способов освоения новой информации.

Теория не заменяет опыта и практику.

habr

kaggle

ШКОЛА АНАЛИЗА ДАННЫХ

The End

Поздравляю с окончанием курса!

Спасибо за внимание!