Обзор оптимизационных задач машинного обучения

Воронцов Константин Вячеславович

СМСМ 2021 • МФТИ • 27–29 октября 2021

Содержание

Общая оптимизационная задача машинного обучения

Дано: обучающая выборка объектов $\{x_i\}_{i=1}^{\ell}$

Найти: вектор параметров w модели $a(x, w)$

Критерий: минимум эмпирического риска

$$\sum_{i=1}^{\ell} L_i(w) \to \min_w$$

где $L_i(w)$ — функция потерь модели $a(x, w)$ на объекте x_i,

или минимум регуляризованного эмпирического риска

$$\sum_{i=1}^{\ell} L_i(w) + \sum_{j=1}^{r} \tau_j R_j(w) \to \min_w$$

где R_j — регуляризаторы, τ_j — коэффициенты регуляризации

Оптимизационная задача восстановления регрессии

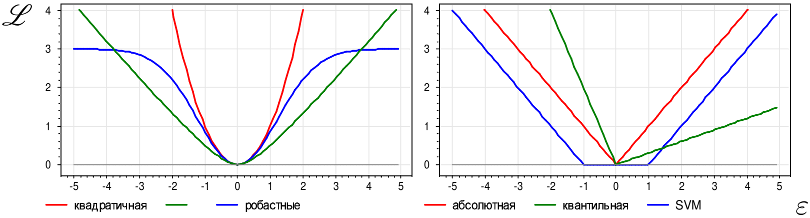
Дано: обучающая выборка $(x_i, y_i)_{i=1}^{\ell}$, $y_i \in \mathbb{R}$

Найти: вектор параметров w модели регрессии $a(x, w)$

Критерий: *минимизация эмпирического риска*

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w) - y_i\bigr) \to \min_{w}$$

Унимодальные функции потерь $\mathscr{L}(\varepsilon)$ от невязки $\varepsilon = a(x, w) - y$:



Оптимизационная задача обучения классификация

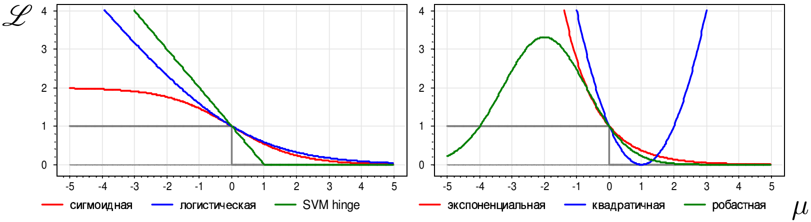
Дано: обучающая выборка $(x_i, y_i)_{i=1}^{\ell}$, $y_i \in \{-1, +1\}$

Найти: вектор w модели классификации $a(x, w) = \operatorname{sign} g(x, w)$

Критерий: *аппроксимация эмпирического риска*

$$\sum_{i=1}^{\ell} \bigl[g(x_i, w) y_i < 0\bigr] \leqslant \sum_{i=1}^{\ell} \mathscr{L}\bigl(g(x_i, w) y_i\bigr) \to \min_{w}$$

Убывающие функции потерь $\mathscr{L}(\mu)$ от отступа $\mu = g(x, w) y$:



Многоклассовая классификация, логистическая регрессия

Дано: обучающая выборка $(x_i, y_i)_{i=1}^{\ell}$, $y_i \in Y$, $|Y| < \infty$

Найти: модель классификации: $a(x, w) = \arg\max\limits_{y \in Y} g(x_i, w_y)$

модель вероятности того, что объект x относится к классу y:

$$P(y|x, w) = \frac{\exp g(x, w_y)}{\sum\limits_{z \in Y} \exp g(x, w_z)} = \operatorname*{SoftMax}_{y \in Y} g(x, w_y),$$

где SoftMax: $\mathbb{R}^Y \to \mathbb{R}^Y$ — гладкое преобразование произвольного вектора в нормированный вектор дискретного распределения.

Критерий: максимум правдоподобия (log-loss):

$$-\sum_{i=1}^{\ell} \ln P(y_i|x_i, w) \;\to\; \min_w$$

Регуляризаторы, штрафующие сложность линейных моделей

Регуляризатор — аддитивная добавка к основному критерию:

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(\langle x_i, w\rangle, y_i\bigr) + \tau\, \text{штраф}(w) \;\to\; \min_w$$

где $\mathscr{L}(a, y)$ — функция потерь, τ — коэффициент регуляризации

L_2-регуляризация (гребневая регрессия, SVM):

$$\text{штраф}(w) = \|w\|_2^2 = \sum_{j=1}^{n} w_j^2.$$

L_1-регуляризация (LASSO, ElasticNet — для отбора признаков):

$$\text{штраф}(w) = \|w\|_1 = \sum_{j=1}^{n} |w_j|.$$

L_0-регуляризация (критерии Акаике AIC, байесовский BIC):

$$\text{штраф}(w) = \|w\|_0 = \sum_{j=1}^{n} \bigl[w_j \neq 0\bigr].$$

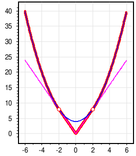
Негладкие регуляризаторы для отбора признаков

Общий вид регуляризаторов (μ — параметр селективности):

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(\langle x_i, w\rangle, y_i\bigr) + \tau \sum_{j=1}^{n} R_\mu(w_j) \;\to\; \min_w .$$

Регуляризаторы с эффектом группировки зависимых признаков:

Elastic Net: $R_\mu(w) = \mu|w| + w^2$

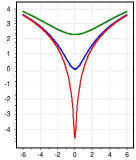


Support Features Machine (SFM):

$$R_\mu(w) = \begin{cases} 2\mu|w|, & |w| \leqslant \mu; \\ \mu^2 + w^2, & |w| \geqslant \mu; \end{cases}$$

Relevance Features Machine (RFM):

$$R_\mu(w) = \ln\bigl(\mu w^2 + 1\bigr)$$



Задачи обучения ранжированию (Learning to Rank)

Дано: обучающая выборка объектов $\{x_i\}_{i=1}^{\ell}$
$i \prec j$ — отношение частичного порядка на парах (x_i, x_j)

Найти: модель ранжирования $a\colon X \to \mathbb{R}$ такую, что

$$i \prec j \quad \Rightarrow \quad a(x_i, w) < a(x_j, w)$$

Критерий: число неверно упорядоченных пар (x_i, x_j)
или аппроксимированный попарный эмпирический риск:

$$\sum_{i \prec j} \bigl[a(x_j, w) < a(x_i, w)\bigr] \leqslant \sum_{i \prec j} \mathscr{L}\bigl(\underbrace{a(x_j, w) - a(x_i, w)}_{\mu_{ij}(w)}\bigr) \to \min_w$$

где $\mathscr{L}(\mu)$ — убывающая функция *попарного отступа* $\mu_{ij}(w)$

Задача восстановления плотности распределения

Дано: обучающая выборка объектов $\{x_i\}_{i=1}^{\ell}$

Найти: вектор параметров θ в модели $p(x|\theta)$

Критерий: максимум правдоподобия

$$\sum_{i=1}^{\ell} \ln p(x_i|\theta) \to \max_{\theta}$$

или максимум апостериорной вероятности

$$\sum_{i=1}^{\ell} \ln p(x_i|\theta) + \ln p(\theta|\gamma) \to \max_{\theta}$$

где γ — вектор гиперпараметров априорного распределения

Задача восстановления смеси плотностей распределения

Дано: обучающая выборка объектов $\{x_i\}_{i=1}^{\ell}$

Найти: параметры w_j, θ_j в модели $p(x|\theta, w) = \sum_{j=1}^{K} w_j p(x|\theta_j)$

Критерий: максимум правдоподобия

$$\sum_{i=1}^{\ell} \ln p(x_i|\theta, w) \;\to\; \max_{\theta, w}$$

или максимум апостериорной вероятности

$$\sum_{i=1}^{\ell} \ln p(x_i|\theta, w) + \ln p(\theta, w|\gamma) \;\to\; \max_{\theta, w}$$

где γ — вектор гиперпараметров априорного распределения

Задача кластеризации (clustering)

Дано: обучающая выборка объектов $\{x_i \in \mathbb{R}^n\colon i = 1, \dots, \ell\}$

Найти:
— центры кластеров $\mu_j \in \mathbb{R}^n$, $j = 1, \dots, K$
— какому кластеру принадлежит каждый объект $a_i \in \{1, \dots, K\}$

Критерий: минимум внутрикластерных расстояний

$$\sum_{i=1}^{\ell} \|x_i - \mu_{a_i}\|^2 \to \min_{\{a_i\}, \{\mu_j\}}$$

в случае евклидовой метрики

$$\|x_i - \mu_j\|^2 = \sum_{d=1}^{n} \bigl(x_{id} - \mu_{jd}\bigr)^2$$

Задача частичного обучения (semi-supervised learning, SSL)

Данные: размеченные $(x_i, y_i)_{i=1}^{k}$, неразмеченные $(x_i)_{i=k+1}^{\ell}$

Найти: классификации $(a_i)_{i=k+1}^{\ell}$ неразмеченных объектов

Критерий и кластеризации, и классификации:

- без модели классификации (transductive learning):

$$\sum_{i=1}^{\ell} \|x_i - \mu_{a_i}\|^2 + \lambda \sum_{i=1}^{k} [a_i \neq y_i] \;\to\; \min_{\{a_i\}, \{\mu_j\}}$$

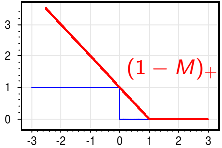
- при построении модели классификации, $a_i = a(x_i, w)$:

$$\sum_{i=1}^{\ell} \|x_i - \mu_{a_i}\|^2 + \lambda \sum_{i=1}^{k} \mathscr{L}(a(x_i, w), y_i) \;\to\; \min_{\{a_i\}, \{\mu_j\}, w}$$

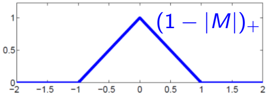
Трансдуктивное обучение модели классификации

$M_i(w) = g(x_i, w)y_i$ — отступ объекта x_i

- Функция потерь $\mathscr{L}(M) = (1 - M)_+$ штрафует размеченные объекты за уменьшение отступа



- Функция потерь $\mathscr{L}(M) = (1 - |M|)_+$ штрафует размеченные объекты за попадание в зазор между классами



Обучение весов w по частично размеченной выборке:

$$\sum_{i=1}^{k} (1 - M_i(w))_+ + \gamma \sum_{i=k+1}^{\ell} (1 - |M_i(w)|)_+ \to \min_w$$

Задачи низкорангового матричного разложения

- Формирование векторных представлений объектов
- Восстановление пропущенных значений в матрице

Дано: матрица $Z = \|z_{ij}\|_{n\times m}$, $(i,j) \in \Omega \subseteq \{1..n\} \times \{1..m\}$

Найти: матрицы $X = \|x_{it}\|_{n\times k}$ и $Y = \|y_{tj}\|_{k\times m}$

Критерий:

$$\|Z - XY\| = \sum_{(i,j)\in\Omega} \mathscr{L}\Bigl(z_{ij} - \sum_t x_{it} y_{tj}\Bigr) \to \min_{X,Y}$$

Почему на практике отказываются от классического SVD:

- неквадратичная функция потерь $\mathscr{L}$
- неотрицательное матричное разложение: $x_{it} \geqslant 0$, $y_{tj} \geqslant 0$
- разреженные данные: $|\Omega| \ll nm$
- ортогональность не нужна или не интерпретируема

Задача построения автокодировщика (обучение без учителя)

Дано: обучающая выборка объектов $\{x_i\}_{i=1}^{\ell}$

Найти:

$f\colon X \to Z$ — кодировщик (encoder), кодовый вектор $z = f(x, \alpha)$

$g\colon Z \to X$ — декодировщик (decoder), реконструкция $\hat{x} = g(z, \beta)$

Критерий: качество реконструкции исходных объектов

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(g(f(x_i, \alpha), \beta), x_i\bigr) \;\to\; \min_{\alpha,\beta}$$

Квадратичная функция потерь: $\mathscr{L}(\hat{x}, x) = \|\hat{x} - x\|^2$

Примеры автокодировщиков:

$f(x, A) = \underset{m\times n}{A}\, x, \quad g(z, B) = \underset{n\times m}{B}\, z$ — линейный

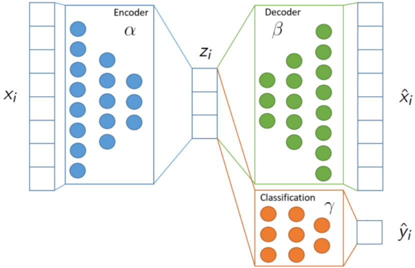
$f(x, A) = \sigma(Ax), \;\; g(z, B) = \sigma(Bz)$ — нейросевой

Автокодировщики для обучения с учителем

Данные: размеченные $(x_i, y_i)_{i=1}^{k}$, неразмеченные $(x_i)_{i=k+1}^{\ell}$

Найти:

$z_i = f(x_i, \alpha)$ — кодировщик
$\hat{x}_i = g(z_i, \beta)$ — декодировщик
$\hat{y}_i = \hat{y}(z_i, \gamma)$ — предиктор



Функции потерь:
$\mathscr{L}(\hat{x}_i, x_i)$ — реконструкция
$\tilde{\mathscr{L}}(\hat{y}_i, y_i)$ — предсказание

Критерий: совместное обучение автокодировщика и предсказательной модели (классификации, регрессии или др.):

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(g(f(x_i, \alpha), \beta), x_i\bigr) + \lambda \sum_{i=1}^{k} \tilde{\mathscr{L}}\bigl(\hat{y}(f(x_i, \alpha), \gamma), y_i\bigr) \to \min_{\alpha, \beta, \gamma}$$

Dor Bank, Noam Koenigstein, Raja Giryes. Autoencoders. 2020

Графовые (матричные) разложения (graph factorization)

Дано: $(i,j) \in E$ — выборка рёбер графа $\langle V, E\rangle$,
S_{ij} — близость между вершинами ребра (i,j)
Например, $S_{ij} = [(i,j) \in E]$ — матрица смежности вершин

Найти: векторные представления вершин, так, чтобы близкие (по графу) вершины имели близкие векторы

Критерий:

- для неориентированного графа (S симметрична):

$$\sum_{(i,j)\in E} \bigl(\langle z_i, z_j\rangle - S_{ij}\bigr)^2 \to \min_{Z}, \quad Z \in \mathbb{R}^{V\times d}$$

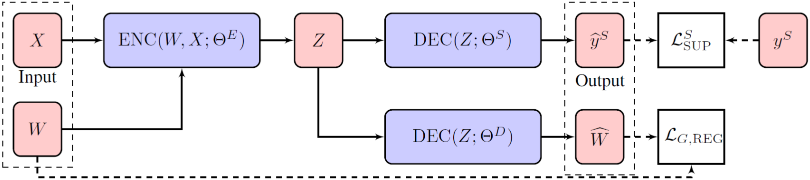
- для ориентированного графа (S несимметрична):

$$\sum_{(i,j)\in E} \bigl(\langle \varphi_i, \theta_j\rangle - S_{ij}\bigr)^2 \to \min_{\Phi,\Theta}, \quad \Phi, \Theta \in \mathbb{R}^{V\times d}$$

I.Chami et al. Machine learning on graphs: a model and comprehensive taxonomy. 2020.

GraphEDM: обобщённый автокодировщик на графах

Graph Encoder Decoder Model — обобщает более 30 моделей:



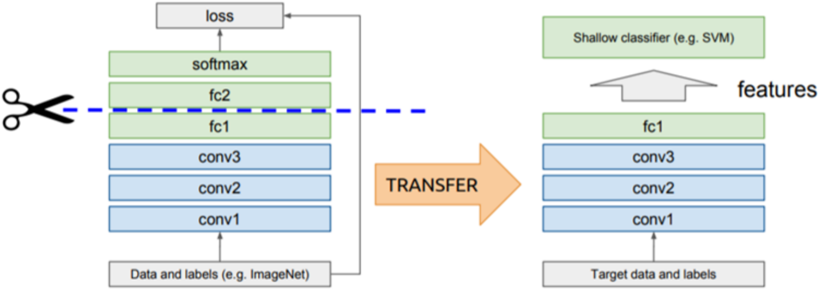
$W \in \mathbb{R}^{V\times V}$ — входные данные о рёбрах
$X \in \mathbb{R}^{V\times n}$ — входные данные о вершинах, признаковые описания
$Z \in \mathbb{R}^{V\times d}$ — векторные представления вершин графа
$\mathrm{DEC}(Z;\Theta^D)$ — декодер, реконструирующий данные о рёбрах
$\mathrm{DEC}(Z;\Theta^S)$ — декодер, решающий supervised-задачу
y^S — (semi-)supervised данные о вершинах или рёбрах
$\mathcal{L}$ — функции потерь

I.Chami et al. Machine learning on graphs: a model and comprehensive taxonomy. 2020.

Пред-обучение нейронных сетей (pre-training)

Свёрточная сеть для обработки изображений:

- $z = f(x, \alpha)$ — свёрточные слои для векторизации объектов
- $y = g(z, \beta)$ — полносвязные слои под конкретную задачу



Jason Yosinski, Jeff Clune, Yoshua Bengio, Hod Lipson. How transferable are features in deep neural networks? 2014.

Перенос обучения (transfer learning)

$f(x,\alpha)$ — универсальная часть модели (векторизация)
$g(x,\beta)$ — специфичная для задачи часть модели

Базовая задача на выборке $\{x_i\}_{i=1}^{\ell}$ с функцией потерь $\mathscr{L}_i$:

$$\sum_{i=1}^{\ell} \mathscr{L}_i\bigl(f(x_i,\alpha), g(x_i,\beta)\bigr) \;\to\; \min_{\alpha,\beta}$$

Целевая задача на другой выборке $\{x'_i\}_{i=1}^{m}$, с другими $\mathscr{L}'_i$, g':

$$\sum_{i=1}^{m} \mathscr{L}'_i\bigl(f(x'_i,\alpha), g'(x'_i,\beta')\bigr) \;\to\; \min_{\beta'}$$

при $m \ll \ell$ это может быть намного лучше, чем

$$\sum_{i=1}^{m} \mathscr{L}'_i\bigl(f(x'_i,\alpha), g'(x'_i,\beta')\bigr) \;\to\; \min_{\alpha,\beta'}$$

Sinno Jialin Pan, Qiang Yang. A Survey on Transfer Learning. 2009

Многозадачное обучение (multi-task learning)

$f(x, \alpha)$ — универсальная часть модели (векторизация)
$g_t(x, \beta)$ — специфичная часть модели для задачи $t \in T$

Совместное обучение модели f по задачам X_t, $t \in T$:

$$\sum_{t \in T} \sum_{i \in X_t} \mathscr{L}_{ti}\bigl(f(x_{ti}, \alpha), g_t(x_{ti}, \beta_t)\bigr) \to \min_{\alpha, \{\beta_t\}}$$

Обучаемость (learnability): качество решения отдельной задачи $\langle X_t, \mathscr{L}_t, g_t \rangle$ улучшается с ростом объёма выборки $\ell_t = |X_t|$.

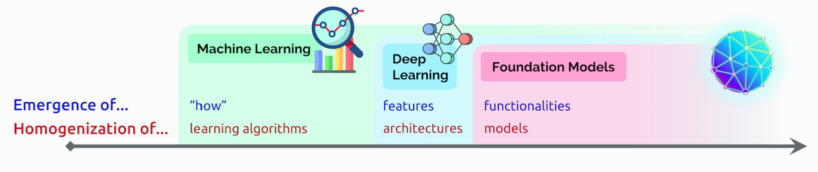
Learning to learn: качество решения каждой из задач $t \in T$ улучшается с ростом как ℓ_t, так и общего числа задач $|T|$.

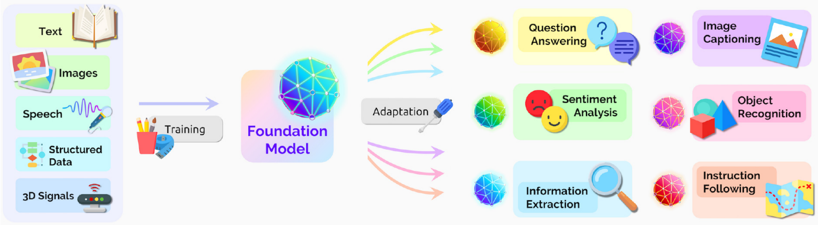
Few-shot learning: для решения задачи t достаточно небольшого числа примеров, иногда даже одного.

M.Crawshaw. Multi-task learning with deep neural networks: a survey. 2020
Y.Wang et al. Generalizing from a few examples: a survey on few-shot learning. 2020

Концепция фундаментальных моделей (Foundation Models)

Обучаемая векторизация данных — глобальный тренд AI/ML





R.Bommasani et al. (Center for Research on Foundation Models, Stanford University)
On the opportunities and risks of foundation models // CoRR, 20 August 2021.

Дистилляция моделей или суррогатное моделирование

Обучение сложной модели $a(x, w)$ «долго, дорого»:

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w), y_i\bigr) \;\to\; \min_{w}$$

Обучение простой модели $b(x, w')$, возможно, на других данных:

$$\sum_{i=1}^{k} \mathscr{L}\bigl(b(x'_i, w'), a(x'_i, w)\bigr) \;\to\; \min_{w'}$$

Примеры задач:

- замена сложной модели (климат, аэродинамика и др.), которая вычисляется на суперкомпьютере месяцами, «лёгкой» аппроксимирующей суррогатной моделью
- замена сложной нейросети, которая обучается неделями на больших данных, «лёгкой» аппроксимирующей нейросетью с минимизацией числа нейронов и связей

Задача обучения с привилегированной информацией

x_i^* — информация об объекте x_i, доступная только на обучении

Раздельное обучение модели-ученика и модели-учителя:

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w), y_i\bigr) \to \min_w \qquad \sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i^*, w^*), y_i\bigr) \to \min_w$$

Модель-ученик обучается повторять ошибки модели-учителя:

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w), y_i\bigr) + \mu \mathscr{L}\bigl(a(x_i, w), a(x_i^*, w^*)\bigr) \;\to\; \min_w$$

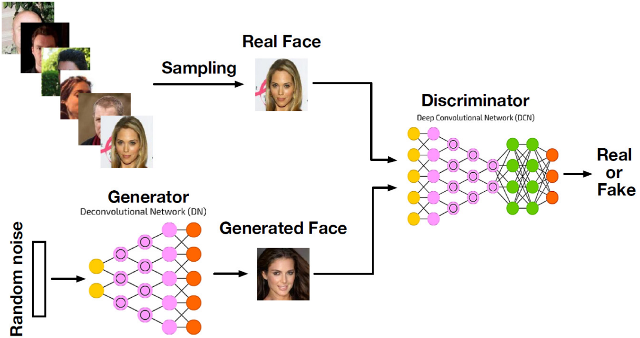
Совместное обучение модели-ученика и модели-учителя:

$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w), y_i\bigr) + \lambda \mathscr{L}\bigl(a(x_i^*, w^*), y_i\bigr) + \\ + \mu \mathscr{L}\bigl(a(x_i, w), a(x_i^*, w^*)\bigr) \;\to\; \min_{w, w*}$$

D.Lopez-Paz, L.Bottou, B.Scholkopf, V.Vapnik. Unifying distillation and privileged information. 2016.

Генеративная состязательная сеть (Generative Adversarial Net)

Генератор $G(z)$ учится порождать объекты x из шума z
Дискриминатор $D(x)$ учится отличать их от реальных объектов



Antonia Creswell et al. Generative Adversarial Networks: an overview. 2017.
Zhengwei Wang et al. Generative Adversarial Networks: a survey and taxonomy. 2019.
Chris Nicholson. A Beginner's Guide to Generative Adversarial Networks. https://pathmind.com/wiki/generative-adversarial-network-gan. 2019.

Постановка задачи GAN

Дано: выборка объектов $\{x_i\}_{i=1}^{\ell}$

Найти две вероятностные модели:

- модель $x = G(z, \alpha)$ генерации $x \sim p(x|z, \alpha)$ из шума z
- дискриминативная модель $D(x, \beta) = p(1|x, \beta)$

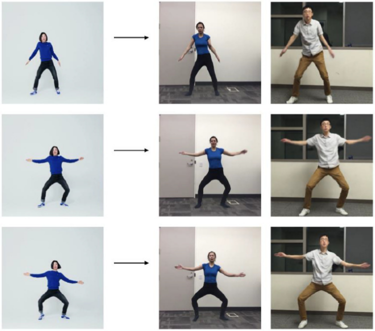
Критерий: log правдоподобия дискриминативной модели; *генератор* $G(z)$ учится порождать объекты x из шума z, *дискриминатор* $D(x)$ учится отличать их от реальных объектов, в антагонистической игре генератора против дискриминатора:

$$\sum_{i=1}^{\ell} \ln D(x_i, \beta) + \ln\bigl(1 - D(G(z_i, \alpha), \beta)\bigr) \;\to\; \max_{\beta} \min_{\alpha}$$

Ian Goodfellow et al. Generative Adversarial Nets. 2014

Примеры GAN для синтеза изображений и видео

(d) input image (e) output 3d face (f) textured 3d face



Source Subject Target Subject 1 Target Subject 2

Chuan Li, Michael Wand. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. 2016.

Xiaoxing Zeng, Xiaojiang Peng, Yu Qiao. DF2Net: A Dense Fine Finer Network for Detailed 3D Face Reconstruction. ICCV-2019.

Caroline Chan, Shiry Ginosar, Tinghui Zhou, Alexei A. Efros. Everybody Dance Now. ICCV-2019.

Вместо резюме. Типология задач машинного обучения

1. Предварительная обработка (data preparation)
   - извлечение признаков (feature extraction)
   - отбор признаков (feature selection)
   - восстановление пропусков (missing values)
   - фильтрация выбросов (outlier detection)
2. Обучение с учителем (supervised learning)
   - классификация (classification)
   - регрессия (regression)
   - ранжирование (learning to rank)
   - прогнозирование (forecasting)
3. Обучение без учителя (unsupervised learning)
   - кластеризация (clustering)
   - восстановление плотности (density estimation)
   - поиск ассоциативных правил (association rule learning)
   - одноклассовая классификация (anomaly detection)
4. Частичное обучение (semi-supervised learning)
   - трансдуктивное обучение (transductive learning)
   - обучение с положительными примерами (PU-learning)

Вместо резюме. Типология задач машинного обучения

5. Обучение представлений (representation learning)
   - обучение признаков (feature learning)
   - матричные разложения (matrix factorization)
   - обучение многообразий (manifold learning)
6. Глубокое обучение (deep learning)
7. Обучение близости/связей (similarity/relational learning)
8. Перенос обучения (transfer learning)
9. Многозадачное обучение (multitask learning)
10. Привилегированное обучение (privileged learning, distilling)
11. Состязательное обучение (adversarial learning)
12. Обучение структуры модели (structure learning)
13. Динамическое обучение (online/incremental learning)
14. Активное обучение (active learning)
15. Обучение с подкреплением (reinforcement learning)
16. Мета-обучение (meta-learning, AutoML)