Методы машинного обучения. Типология оптимизационных задач в машинном обучении

Воронцов Константин Вячеславович
www.MachineLearning.ru/wiki?title=User:Vokov
вопросы к лектору: voron@forecsys.ru

материалы курса:
github.com/MSU-ML-COURSE/ML-COURSE-22-23
орг.вопросы по курсу: ml.cmc@mail.ru

ВМК МГУ • 27 сентября 2022

Содержание

1. **Обучение с учителем**
   - Классификация и регрессия
   - Регуляризация
   - Обучение ранжированию
2. **Обучение без учителя**
   - Восстановление плотности распределения
   - Кластеризация и частичное обучение
   - Обучаемая векторизация объектов
3. **Неклассические парадигмы обучения**
   - Перенос обучения и многозадачное обучение
   - Обучение с привилегированной информацией
   - Типология задач машинного обучения

Общая оптимизационная задача машинного обучения

Дано: 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

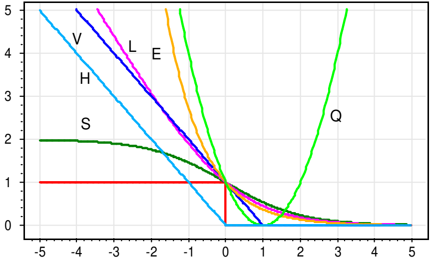
Оптимизационная задача обучения классификации

Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell$, $x_i \in \mathbb{R}^n$, $y_i \in \{-1, +1\}$

1. Фиксируется модель классификации, например, *линейная*:
$$a(x, w) = \operatorname{sign}\langle x, w\rangle = \operatorname{sign} \sum_{j=1}^{n} w_j f_j(x)$$
2. Функция потерь — убывающая функция $\mathscr{L}$ от отступа:
$$L_i(w) = \bigl[a(x_i, w) y_i < 0\bigr] = \bigl[\langle x_i, w\rangle y_i < 0\bigr] \leqslant \mathscr{L}\bigl(\langle x_i, w\rangle y_i\bigr)$$
3. Метод обучения — *минимизация эмпирического риска*:
$$\sum_{i=1}^{\ell} \bigl[\langle x_i, w\rangle y_i < 0\bigr] \leqslant \sum_{i=1}^{\ell} \mathscr{L}\bigl(\langle x_i, w\rangle y_i\bigr) \to \min_w$$
4. Проверка по тестовой выборке $X^k = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$:
$$Q(X^k) = \frac{1}{k} \sum_{i=1}^{k} \bigl[\langle \tilde{x}_i, w\rangle \tilde{y}_i < 0\bigr]$$

Непрерывные аппроксимации пороговой функции потерь

Часто используемые непрерывные функции потерь $\mathscr{L}(M)$:



$[M < 0]$ — пороговая функция потерь

$V(M) = (1 - M)_+$ — кусочно-линейная (SVM)

$H(M) = (-M)_+$ — кусочно-линейная (Hebb's rule)

$L(M) = \log_2(1 + e^{-M})$ — логарифмическая (LR)

$Q(M) = (1 - M)^2$ — квадратичная (FLD)

$S(M) = 2(1 + e^M)^{-1}$ — сигмоидная (ANN)

$E(M) = e^{-M}$ — экспоненциальная (AdaBoost)

Напоминание. Двухклассовая логистическая регрессия

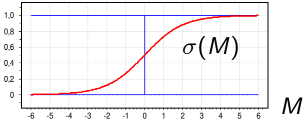
Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell$, $x_i \in \mathbb{R}^n$, $y_i \in \{-1, +1\}$

Линейная модель классификации: $a(x, w) = \operatorname{sign}\langle x, w\rangle$

Вероятностная модель классификации:

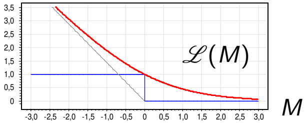
$$P(y|x, w) = \sigma(\langle w, x\rangle y) = \frac{1}{1+e^{-\langle w,x\rangle y}}$$

где $\sigma(M) = \frac{1}{1+e^{-M}}$ — сигмоида



Функция потерь log-loss:

$$\mathscr{L}(M) = \ln(1 + e^{-M}) = -\ln\sigma(M)$$



Минимизация эмпирического риска
эквивалентна максимизации правдоподобия (+ регуляризация):

$$-\sum_{i=1}^{\ell} \ln P(y_i|x_i, w) + \frac{\tau}{2}\|w\|^2 \to \min_w$$

Напоминание. Многоклассовая логистическая регрессия

Линейная модель многоклассовой классификации, $|Y| \geqslant 2$:

$$a(x, w) = \arg\max_{y \in Y} \langle w_y, x \rangle, \quad x, w_y \in \mathbb{R}^n$$

Вероятностная модель классификации:

$$P(y|x, w) = \frac{\exp\langle w_y, x\rangle}{\sum\limits_{z \in Y} \exp\langle w_z, x\rangle} = \mathop{\mathrm{SoftMax}}_{y \in Y} \langle w_y, x\rangle,$$

где $\mathrm{SoftMax}\colon \mathbb{R}^Y \to \mathbb{R}^Y$ переводит произвольный вектор в нормированный вектор дискретного распределения.

Максимизация правдоподобия (+ регуляризация):

$$-\sum_{i=1}^{\ell} \ln P(y_i|x_i, w) + \frac{\tau}{2} \sum_{y \in Y} \|w_y\|^2 \to \min_w$$

Оптимизационная задача восстановления регрессии

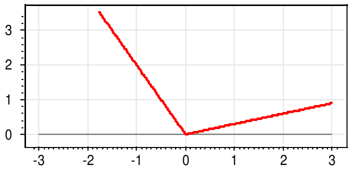
Обучающая выборка: $X^\ell = (x_i, y_i)_{i=1}^\ell, \;\; x_i \in \mathbb{R}^n, \;\; y_i \in \mathbb{R}$

1. Фиксируется модель регрессии, например, *линейная*:
$$a(x, w) = \langle x, w \rangle = \sum_{j=1}^n w_j f_j(x), \qquad w \in \mathbb{R}^n$$
2. Фиксируется функция потерь, например, *квадратичная*:
$$L_i(w) = \bigl(a(x_i, w) - y_i\bigr)^2$$
3. Метод обучения — *метод наименьших квадратов*:
$$\sum_{i=1}^\ell \bigl(a(x_i, w) - y_i\bigr)^2 \to \min_w$$
4. Проверка по тестовой выборке $X^k = (\tilde{x}_i, \tilde{y}_i)_{i=1}^k$:
$$Q(X^k) = \frac{1}{k} \sum_{i=1}^k \bigl(a(\tilde{x}_i, w) - \tilde{y}_i\bigr)^2$$

Квантильная регрессия

Функция потерь, $\varepsilon = a(x_i, w) - y_i$:

$$\mathscr{L}(\varepsilon) = \begin{cases} C_+|\varepsilon|, & \varepsilon > 0 \\ C_-|\varepsilon|, & \varepsilon < 0; \end{cases}$$



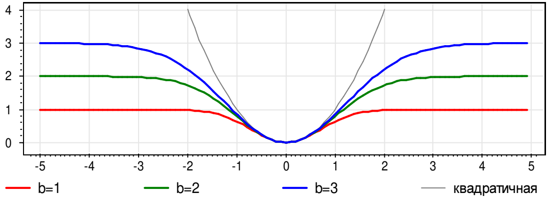
Модель $a(x, w) = w$: решением является q-кванти́ль $w = y^{(\ell q)}$, где $y^{(1)}, \dots, y^{(\ell)}$ — вариационный ряд, $q = \frac{C_-}{C_- + C_+}$.

Модель $a(x, w) = \langle x, w\rangle$: задача линейного программирования после замены переменных $\varepsilon_i^+ = \bigl(a(x_i) - y_i\bigr)_+$, $\varepsilon_i^- = \bigl(y_i - a(x_i)\bigr)_+$:

$$\begin{cases} \sum\limits_{i=1}^{\ell} \bigl(C_+\varepsilon_i^+ + C_-\varepsilon_i^-\bigr) \to \min\limits_{w}; \\ \langle x_i, w\rangle - y_i = \varepsilon_i^+ - \varepsilon_i^-; \quad \varepsilon_i^+ \geqslant 0; \quad \varepsilon_i^- \geqslant 0. \end{cases}$$

Робастная (помехоустойчивая) регрессия

Функция Мешалкина: $\mathscr{L}(\varepsilon) = b\bigl(1 - \exp\bigl(-\frac{1}{b}\varepsilon^2\bigr)\bigr)$, $\varepsilon = a - y$



Модель регрессии: не обязательно линейная $a(x, w)$

Постановка оптимизационной задачи:

$$\sum_{i=1}^{\ell} \exp\Bigl(-\frac{1}{b}(a(x_i, w) - y_i)^2\Bigr) \to \max_w$$

Численное решение — методом Ньютона-Рафсона

Задачи прогнозирования временных рядов

Дано: $y_0, y_1, \ldots, y_t, \ldots$ — временной ряд, $y_i \in \mathbb{R}$
Найти: $\hat{y}_{t+d}(w) = f_{t,d}(y_1, \ldots, y_t; w)$ — модель временного ряда, где $d = 1, \ldots, D$, D — горизонт прогнозирования,
w — вектор параметров модели.

Критерий: минимум среднеквадратичной ошибки прогнозов:

$$\sum_{t=T_0}^{T} \bigl(\hat{y}_{t+d}(w) - y_{t+d}\bigr)^2 \;\to\; \min_w$$

Пример: линейная модель авторегрессии.
В роли признаков выступают n предыдущих наблюдений ряда:

$$\hat{y}_{t+1}(w) = \sum_{j=1}^{n} w_j y_{t-j+1}, \quad w \in \mathbb{R}^n$$

В роли объектов $\ell = t - n + 1$ моментов истории ряда.

Регуляризаторы, штрафующие сложность линейной модели

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:

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

L_1-регуляризация (приводит к отбору признаков):

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

L_0-регуляризация (приводит к отбору признаков):

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

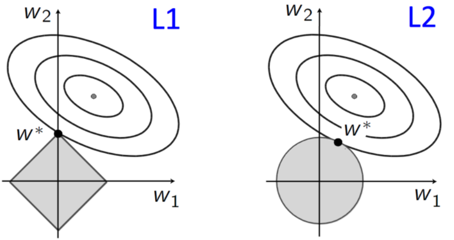
L1-регуляризация приводит к отбору признаков

Эквивалентная постановка задачи $\sum\limits_{i=1}^{\ell} \mathscr{L}\bigl(\langle x_i, w\rangle, y_i\bigr) \to \min\limits_{w}$

с регуляризатором в виде ограничения-неравенства:

$$\text{L1:}\ \sum_{j=1}^{n} |w_j| \leqslant \varkappa \qquad \text{L2:}\ \sum_{j=1}^{n} w_j^2 \leqslant \varkappa$$

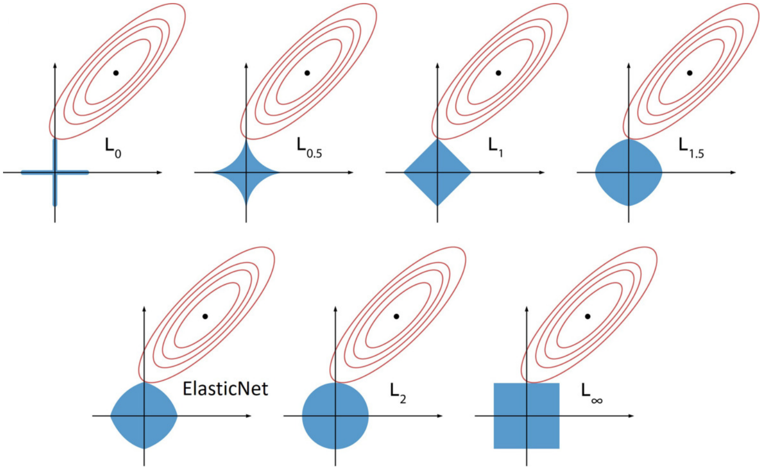
L1 — это метод LASSO (Least Absolute Shrinkage and Selection Operator) приводит к обнулению некоторых w_j, то есть к *отбору признаков*



T.Hastie, R.Tibshirani, J.Friedman. The Elements of Statistical Learning. 2017.

Геометрическая интерпретация L_p-регуляризаторов

Отбор признаков происходит благодаря негладкости нормы:



Задачи ранжирования (Learning to Rank, LtR, L2R, LETOR)

Дано: $X^\ell = \{x_1, \ldots, x_\ell\}$ — обучающая выборка
$i \prec j$ — правильный порядок на парах (x_i, x_j)

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

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

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

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

где $\mathscr{L}(M)$ — убывающая функция *парного отступа* $M_{ij}(w)$

Задача восстановления плотности распределения

Дано: обучающая выборка $\{x_i\colon i = 1, \ldots, \ell\}$

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 $\{x_i\colon i = 1, \ldots, \ell\}$

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

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 $\{x_i \in \mathbb{R}^n \colon i = 1, \ldots, \ell\}$

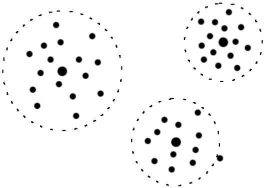
Найти:

— центры кластеров $\mu_j \in \mathbb{R}^n$, $j = 1, \ldots, K$

— какому кластеру принадлежит каждый объект $a_i \in \{1, \ldots, K\}$

Критерий: минимум суммы внутрикластерных расстояний

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



Метрика, как правило, евклидова (но может быть и другая):

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

Одноклассовая классификация (one-class classification)

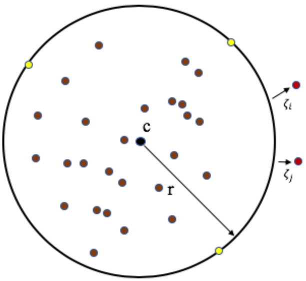
Задачи детекции выбросов / аномалий / новизны
(outlier / anomaly / novelty detection)

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

Найти: центр $c \in \mathbb{R}^n$ и радиус r шара, охватывающего всю выборку кроме, быть может, небольшого числа аномальных объектов-выбросов

Критерий: минимизация радиуса шара и суммы штрафов за выход из шара:

$$\tau r^2 + \sum_{i=1}^{\ell} \mathscr{L}\bigl(\underbrace{r^2 - \|x_i - c\|^2}_{\zeta_i = \text{margin}(c,r)}\bigr) \to \min_{c,r}$$



Задача частичного обучения (semi-supervised learning, SSL)

Дано:

$X^k = \{x_1, \ldots, x_k\}$ — размеченные объекты (labeled data);
$\{y_1, \ldots, y_k\}$

$U = \{x_{k+1}, \ldots, x_\ell\}$ — неразмеченные объекты (unlabeled data).

Найти: классификации $\{a_{k+1}, \ldots, a_\ell\}$ неразмеченных объектов

Критерий 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, $a_i = a(x_i, w)$:

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

Частный случай SSL: PU-learning (Positive and Unlabeled)

Примеры задач, когда известны объекты только одного класса:

- обнаружение мошеннических транзакций
- рекомендательные системы, персонализация рекламы
- медицинская диагностика при неизвестном анамнезе
- автоматическое пополнение базы знаний фактами

Модель двухклассовой классификации $a(x_i, w)$.
Неразмеченные трактуются как негативные с весом $\lambda \ll 1$:

$$\sum_{i=1}^{k} \mathscr{L}(a(x_i, w), +1) + \lambda \sum_{i=k+1}^{\ell} \mathscr{L}(a(x_i, w), -1) \to \min_{w}$$

Неразмеченные могут выбираться случайно (Negative Sampling)

Gang Li. A Survey on Positive and Unlabelled Learning. 2013.
J.Bekker, J.Davis. Learning From Positive and Unlabeled Data: A Survey. 2020.

Задача обучения автокодировщика (AutoEncoder)

Дано: $X^\ell = \{x_1, \ldots, x_\ell\}$ — обучающая выборка

Найти модель векторизации, сохраняющую информацию:
$f\colon X \to Z$ — кодировщик (encoder), кодовый вектор $z = f(x, \alpha)$
$g\colon Z \to X$ — декодировщик (decoder), реконструкция $\hat{x} = g(z, \beta)$

Критерий: точность восстановления объектов $g(f(x_i)) = \hat{x}_i \approx x_i$

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

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

Пример. Линейный автокодировщик: $x \in \mathbb{R}^n$, $z \in \mathbb{R}^m$

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

При $m \ll n$ происходит сжатие данных об объектах

Автокодировщик, частично обучаемый с учителем

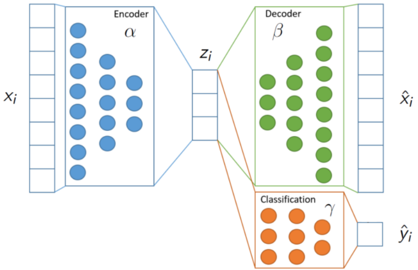
Данные: размеченные $(x_i, y_i)_{i=1}^k$, неразмеченные $(x_i)_{i=k+1}^\ell$

Найти: кодировщик f, декодировщик g и предиктор $\hat{y}$ (предсказательную модель классификации, регрессии или др.):

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

$z_i = f(x_i, \alpha)$ — 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



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

Задачи низкорангового матричного разложения

Дано: матрица $X = \|x_{ij}\|_{\ell\times n}$, $(i,j) \in \Omega \subseteq \{1..\ell\} \times \{1..n\}$

Найти: матрицы $Z = \|z_{it}\|_{\ell\times m}$ и $B = \|b_{tj}\|_{m\times n}$, $m \ll \ell, n$

Критерий: точность восстановления X произведением ZB:

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

Применения матричных разложений:

- для восстановления пустых ячеек (missing values) $x_{ij} \notin \Omega$
- для генерации сжатых векторных представлений $x_i \mapsto z_i$
- для векторизации объектов по транзакционным данным
- в рекомендательных системах
- в тематическом моделировании

Графовые (матричные) разложения (graph factorization)

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

Найти: векторные представления вершин $z_i \in \mathbb{R}^d$, чтобы близкие (по графу) вершины имели близкие векторы.

Критерий для неориентированного графа (X симметрична):

$$\left\|X - ZZ^{\mathsf{T}}\right\|_E = \sum_{(i,j)\in E} \bigl(x_{ij} - \langle z_i, z_j\rangle\bigr)^2 \to \min_Z, \quad Z \in \mathbb{R}^{V\times d}$$

Критерий для ориентированного графа (X несимметрична):

$$\left\|X - ZB\right\|_E = \sum_{(i,j)\in E} \bigl(x_{ij} - \langle z_i, b_j\rangle\bigr)^2 \to \min_{Z,B}, \quad Z, B \in \mathbb{R}^{V\times d}$$

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

Многомерное шкалирование (multidimensional scaling, MDS)

Дано: $(i,j) \in E$ — выборка рёбер графа $\langle V, E \rangle$,
x_{ij} — расстояния (distance) между вершинами ребра (i,j)

Найти: векторные представления вершин $z_i \in \mathbb{R}^d$, чтобы близкие (по графу) вершины имели близкие векторы

Критерий стресса (stress):

$$\sum_{(i,j)\in E} w_{ij} \bigl(\rho(z_i, z_j) - x_{ij}\bigr)^2 \to \min_Z, \quad Z \in \mathbb{R}^{V \times d},$$

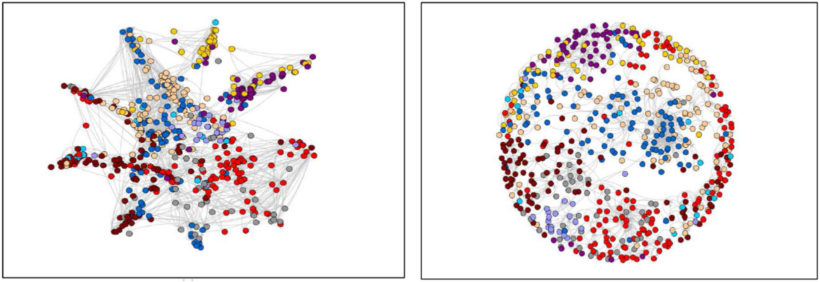
где $\rho(z_i, z_j) = \|z_i - z_j\|$ — обычно евклидово расстояние,
w_{ij} — веса (какие расстояния важнее, большие или малые)

Обычно решается методом стохастического градиента (SG)

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

Многомерное шкалирование для визуализации данных

При $d = 2$ осуществляется проекция выборки на плоскость



- Используется для визуализации кластерных структур
- Форму облака точек можно настраивать весами и метрикой
- Недостаток — искажения неизбежны
- Наиболее популярный метод для визуализации — t-SNE

Laurens van der Maaten, Geoffrey Hinton. Visualizing data using t-SNE. 2008

Перенос обучения (transfer learning)

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

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

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

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

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

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

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

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

Многозадачное обучение (multi-task learning)

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

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

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

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 новой 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

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 у 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:

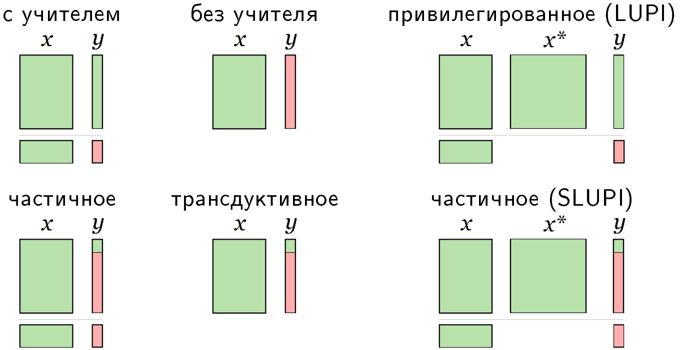
$$\sum_{i=1}^{\ell} \mathscr{L}\bigl(a(x_i, w), y_i\bigr) + \lambda \mathscr{L}\bigl(a(x_i^*, w^*), y_i\bigr) + \\ + \mu \mathscr{L}\bigl(a(x_i, w), a(x_i^*, w^*)\bigr) \;\to\; \min_{w, w*}$$

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

Обучение с использованием привилегированной информации

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

Варианты LUPI (Learning Using Priveleged Information):



V. Vapnik, A. Vashist. A new learning paradigm: Learning Using Privileged Information // Neural Networks. 2009.

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)
   - восстановление плотности (density estimation)
   - кластеризация (clustering)
   - одноклассовая классификация (anomaly detection)
   - поиск ассоциативных правил (association rule learning)
4. Частичное обучение (semi-supervised learning)
   - трансдуктивное обучение (transductive learning)
   - обучение с положительными примерами (PU-learning)

Типология задач машинного обучения

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