Статистические методы обучения по прецедентам

К. В. Воронцов
vokov@forecsys.ru

Полный курс доступен на странице вики-ресурса
http://www.MachineLearning.ru/wiki
«Машинное обучение (курс лекций, К.В.Воронцов)»

26 сентября, 3 октября 2012

Содержание

Задача обучения по прецедентам

X — множество *объектов*;
Y — множество *ответов*;
$y^*: X \to Y$ — неизвестная зависимость (target function).

Дано:
$\{x_1, \ldots, x_\ell\} \subset X$ — *обучающая выборка* (training sample);
$y_i = y^*(x_i), \;\; i = 1, \ldots, \ell$ — известные ответы.

Найти:
$a: X \to Y$ — алгоритм, решающую функцию (decision function), приближающую y^* на всём множестве X.

- $Y = \{1, \ldots, M\}$ — задача *классификации* на M классов.
- $Y = \mathbb{R}$ — задача восстановления *регрессии*.
- Y — конечное упорядоченное множество — задача ранговой регрессии или ранжирования.

«Данные» в задачах обучения по прецедентам

$f_j\colon X \to D_j,\ \ j = 1,\ldots,n$ — признаки объектов.

Типы признаков:

- $D_j = \{0,1\}$ — *бинарный* признак f_j;
- $|D_j| < \infty$ — *номинальный* признак f_j;
- $|D_j| < \infty$, D_j упорядочено — *порядковый* признак f_j;
- $D_j = \mathbb{R}$ — *количественный* признак f_j.

Вектор $\big(f_1(x),\ldots,f_n(x)\big)$ — *признаковое описание* объекта x.

Данные — матрица «*объекты–признаки*» и вектор ответов

$$F = \big\|f_j(x_i)\big\|_{\ell\times n} = \begin{pmatrix} f_1(x_1) & \ldots & f_n(x_1) \\ \ldots & \ldots & \ldots \\ f_1(x_\ell) & \ldots & f_n(x_\ell) \end{pmatrix}, \quad y = \big\|y_i\big\|_{\ell\times 1} = \begin{pmatrix} y_1 \\ \ldots \\ y_\ell \end{pmatrix}.$$

Вероятностная постановка задачи классификации

X — объекты, Y — ответы, $|Y| < \infty$;
$X \times Y$ — вероятностное пространство с плотностью $p(x, y)$;
Дано:
$X^\ell = (x_i, y_i)_{i=1}^\ell$ — простая выборка из $p(x, y)$;
Найти:
классификатор $a\colon X \to Y$ с минимальной вероятностью ошибки.
Временное допущение: пусть известна совместная плотность

$$p(x, y) = p(x)\,\mathrm{P}(y|x) = \mathrm{P}(y)p(x|y).$$

$\mathrm{P}(y) \equiv P_y$ — *априорная вероятность* класса y;
$p(x|y) \equiv p_y(x)$ — *функция правдоподобия* класса y;
$\mathrm{P}(y|x)$ — *апостериорная вероятность* класса y;

Байесовский классификатор:

$$a(x) = \arg\max_{y \in Y} \mathrm{P}(y|x) = \arg\max_{y \in Y} P_y p_y(x).$$

Итак, есть две подзадачи, причём вторую мы уже решили!

1. **Дано:**
   $X^\ell = (x_i, y_i)_{i=1}^\ell$ — обучающая выборка.
   Найти:
   эмпирические оценки $\hat{P}_y$ и $\hat{p}_y(x)$, $y \in Y$
   (восстановить плотность распределения по выборке).
2. **Дано:**
   априорные вероятности P_y,
   функции правдоподобия $p_y(x)$, $y \in Y$.
   Найти:
   классификатор $a\colon X \times Y$, минимизирующий
   вероятность ошибочной классификации.

Ехидное замечание: Когда вместо P_y и $p_y(x)$ подставляются их эмпирические оценки, байесовский классификатор перестаёт быть оптимальным.

Задачи эмпирического оценивания

- **Оценивание априорных вероятностей** частотами

$$\hat{P}_y = \frac{\ell_y}{\ell}, \quad \ell_y = |X_y|, \quad X_y = \{x_i \in X : y_i = y\}, \quad y \in Y.$$

- **Оценивание функций правдоподобия:**
  Дано:
  $X^m = \{x_1, \ldots, x_m\}$ — простая выборка (X_y без ответов y_i).
  Найти:
  эмпирическую оценку плотности $\hat{p}(x)$,
  аппроксимирующую истинную плотность $p(x)$ на всём X:

$$\hat{p}(x) \to p(x) \text{ при } m \to \infty.$$

Оценка плотности Парзена–Розенблатта

Если на X задана функция расстояния $\rho(x, x')$:

$$\hat{p}_{y,h}(x) = \frac{1}{\ell_y V(h)} \sum_{i:\, y_i = y} K\left(\frac{\rho(x, x_i)}{h}\right),$$

где h — ширина окна;
$K(r)$ — *ядро*, невозрастающая неотрицательная функция;
$V(h) = \int_X K\left(\frac{\rho(x,x_i)}{h}\right) dx$ — нормирующий множитель,

Метод *парзеновского окна* (Parzen window):

$$a(x; X^\ell, h) = \arg\max_{y \in Y} \frac{P_y}{\ell_y} \sum_{i:\, y_i = y} K\left(\frac{\rho(x, x_i)}{h}\right).$$

Чтобы $V(h)$ сократился, он не должен зависеть от x_i и y.

Обоснование оценки Парзена-Розенблатта

Теорема (одномерный случай, $X = \mathbb{R}$)

Пусть выполнены следующие условия:

1) X^m — простая выборка из распределения $p(x)$;
2) ядро $K(z)$ непрерывно и ограничено: $\int_X K^2(z)\,dz < \infty$;
3) последовательность h_m: $\lim\limits_{m\to\infty} h_m = 0$ и $\lim\limits_{m\to\infty} mh_m = \infty$.

Тогда:

1) $\hat{p}_{h_m}(x) \to p(x)$ при $m \to \infty$ для почти всех $x \in X$;
2) скорость сходимости имеет порядок $O(m^{-2/5})$.

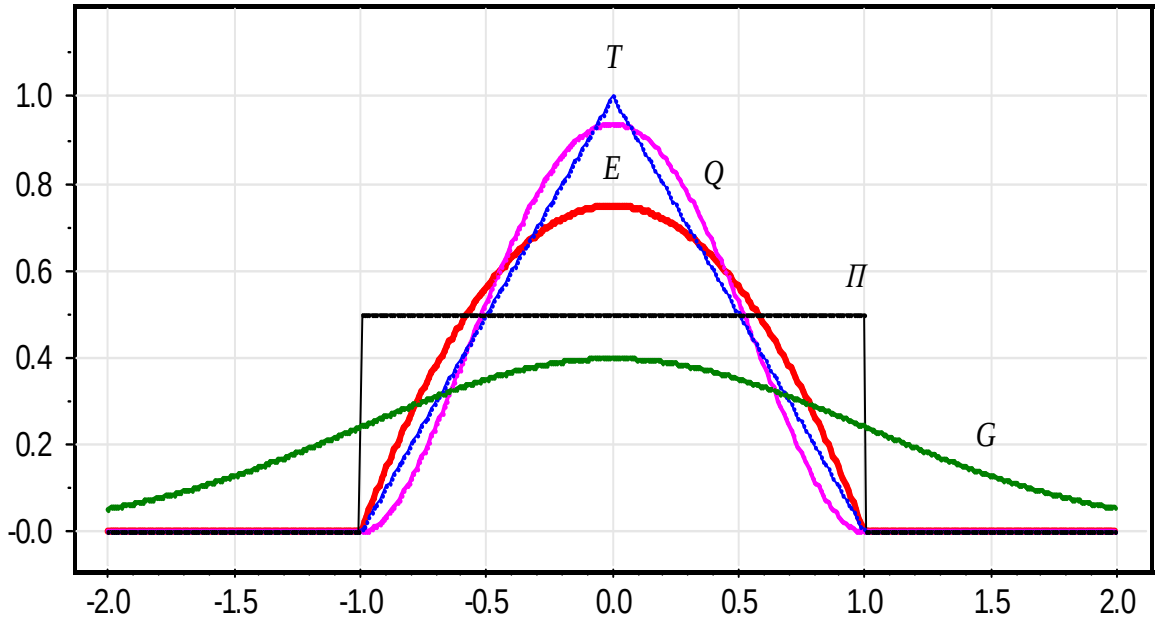
Выбор метрики, ядра, ширины окна

- Один из вариантов ρ — взвешенная метрика Минковского:

$$\rho(x, x') = \left(\sum_{j=1}^{n} w_j \left| f_j(x) - f_j(x') \right|^p \right)^{\frac{1}{p}},$$

  где w_j — неотрицательные веса признаков, $p > 0$.
- Ядро $K(r)$ влияет на гладкость границы.
- Ширина окна h влияет на качество классификации.

Часто используемые ядра



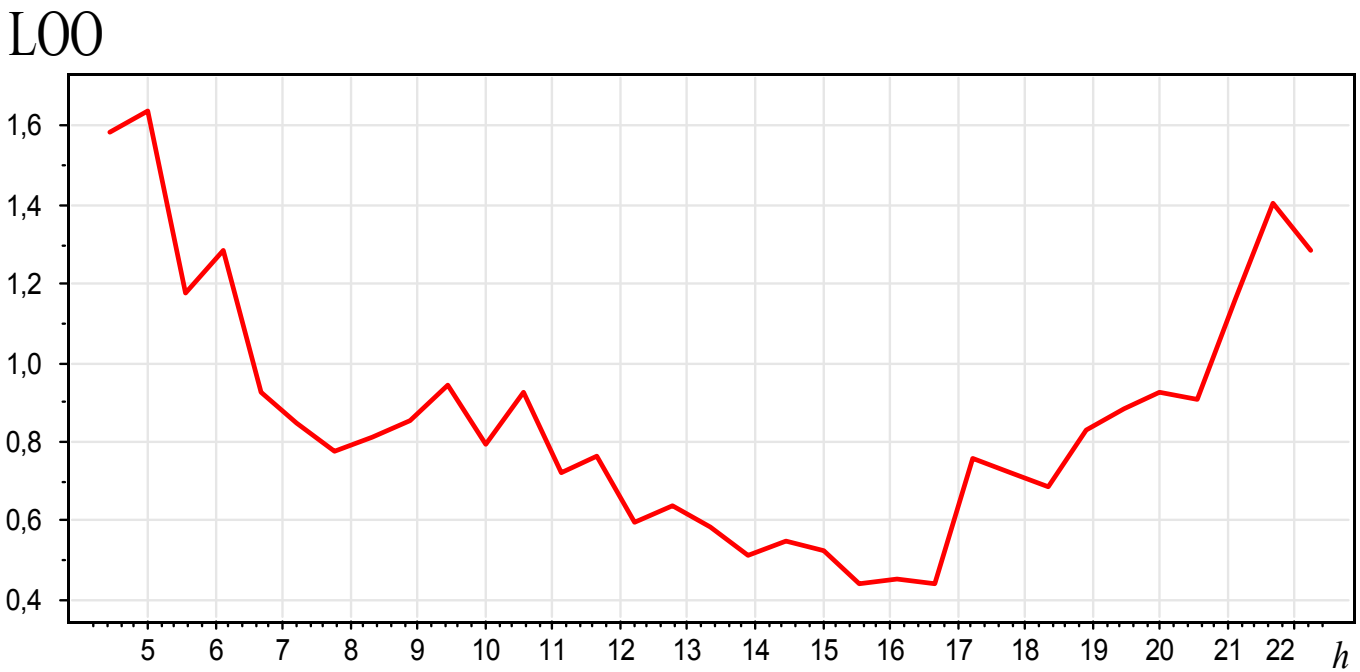
$E(r) = \frac{3}{4}(1-r^2)\big[|r| \leqslant 1\big]$ — оптимальное (Епанечникова);
$Q(r) = \frac{15}{16}(1-r^2)^2\big[|r| \leqslant 1\big]$ — квартическое;
$T(r) = \big(1-|r|\big)\big[|r| \leqslant 1\big]$ — треугольное;
$G(r) = (2\pi)^{-1/2}\exp(-\frac{1}{2}r^2)$ — гауссовское;
$\Pi(r) = \frac{1}{2}\big[|r| \leqslant 1\big]$ — прямоугольное.

Выбор ширины окна

Скользящий контроль *Leave One Out*:

$$\mathrm{LOO}(h, X^\ell) = \sum_{i=1}^{\ell} \Big[a\big(x_i; X^\ell \setminus x_i, h\big) \neq y_i \Big] \to \min_h,$$

Типичный вид зависимости LOO от h:



Окна переменной ширины (метод k ближайших соседей)

Проблема: при наличии локальных сгущений плотности $p(x)$ любое значение ширины окна h не оптимальна.

Идея: задавать не ширину окна h, а число соседей k:

$$h(x) = \rho\left(x, x^{(k+1)}\right),$$

где $x^{(i)}$ — i-й сосед объекта x при ранжировании выборки X^ℓ:

$$\rho(x, x^{(1)}) \leqslant \cdots \leqslant \rho(x, x^{(\ell)})$$

Замечание 1:

нормировка $V(k)$ не должна зависеть от y, поэтому выборка ранжируется целиком, а не по классам X_y.

Замечание 2:

Оптимизация k по LOO аналогична оптимизации h.

Замечание 3:

При $K(r) = \Pi(r)$ это обычный метод kNN.

Принцип максимума правдоподобия

Пусть известна параметрическая модель плотности

$$p(x) = \varphi(x;\theta),$$

где θ — параметр, φ — фиксированная функция.
Задача — найти оптимальное θ по простой выборке X^m.

Принцип максимума правдоподобия:

$$L(\theta; X^m) = \sum_{i=1}^{m} \ln \varphi(x_i;\theta) \to \max_{\theta}.$$

Необходимое условие оптимума:

$$\frac{\partial}{\partial\theta} L(\theta; X^m) = \sum_{i=1}^{m} \frac{\partial}{\partial\theta} \ln \varphi(x_i;\theta) = 0,$$

где функция $\varphi(x;\theta)$ достаточно гладкая по параметру θ.

Многомерное нормальное распределение

Пусть $X = \mathbb{R}^n$ — объекты описываются n числовыми признаками.

Гипотеза: классы имеют n-мерные гауссовские плотности:

$$p_y(x) = \mathcal{N}(x; \mu_y, \Sigma_y) = \frac{\exp\left(-\frac{1}{2}(x-\mu_y)^\intercal \Sigma_y^{-1}(x-\mu_y)\right)}{\sqrt{(2\pi)^n \det \Sigma_y}}, \quad y \in Y,$$

где $\mu_y \in \mathbb{R}^n$ — вектор матожидания (центр) класса $y \in Y$,
$\Sigma_y \in \mathbb{R}^{n \times n}$ — ковариационная матрица класса $y \in Y$
(симметричная, невырожденная, положительно определённая).

Теорема

1. Разделяющая поверхность $\left\{x \in X \;\middle|\; P_y p_y(x) = P_s p_s(x)\right\}$ квадратична для всех $y, s \in Y$, $y \neq s$.
2. Если $\Sigma_y = \Sigma_s$, то она вырождается в линейную.

Линейный дискриминант Фишера

Допущение:

ковариационные матрицы классов равны: $\Sigma_y = \Sigma, \ \ y \in Y$.

Оценки максимума правдоподобия параметров Σ, μ_y:

$$\hat{\mu}_y = \frac{1}{\ell_y} \sum_{i:\, y_i = y} x_i; \quad \hat{\Sigma} = \frac{1}{\ell} \sum_{i=1}^{\ell} (x_i - \hat{\mu}_{y_i})(x_i - \hat{\mu}_{y_i})^{\mathsf{T}},$$

Линейный дискриминант (подстановочный алгоритм):

$$\begin{aligned} a(x) &= \arg\max_{y \in Y} \ \hat{P}_y \hat{p}_y(x) = \\ &= \arg\max_{y \in Y} \Big(\underbrace{\ln \hat{P}_y - \tfrac{1}{2} \hat{\mu}_y^{\mathsf{T}} \hat{\Sigma}^{-1} \hat{\mu}_y}_{\beta_y} + x^{\mathsf{T}} \underbrace{\hat{\Sigma}^{-1} \hat{\mu}_y}_{\alpha_y} \Big) = \\ &= \arg\max_{y \in Y} \big(x^{\mathsf{T}} \alpha_y + \beta_y \big). \end{aligned}$$

Наивный байесовский классификатор

Допущение (в самом деле наивное):

Признаки $f_j : X \to D_j$ — независимые случайные величины с плотностями распределения, $p_{y,j}(\xi)$, $y \in Y$, $j = 1, \ldots, n$.

Тогда функции правдоподобия классов представимы в виде произведения одномерных плотностей по признакам:

$$p_y(x) = p_{y,1}(\xi_1) \cdots p_{y,n}(\xi_n), \quad x = (\xi_1, \ldots, \xi_n), \quad y \in Y.$$

Прологарифмируем (для удобства). Получим классификатор

$$a(x) = \arg\max_{y \in Y} \left(\ln \hat{P}_y + \sum_{j=1}^{n} \ln \hat{p}_{yj}(\xi_j) \right).$$

Восстановление n одномерных плотностей
— намного более простая задача, чем одной n-мерной.

Диагонализация ковариационной матрицы

Идея: пусть признаки некоррелированы: $\sigma_{ij} = 0, \quad i \neq j$.

Замечание: для нормального распределения
некоррелированность $\Longleftrightarrow$ независимость

Получаем **наивный байесовский классификатор:**

$$\hat{p}_{yj}(\xi) = \frac{1}{\sqrt{2\pi}\hat{\sigma}_{yj}} \exp\left(-\frac{(\xi - \hat{\mu}_{yj})^2}{2\hat{\sigma}_{yj}^2}\right), \quad y \in Y, \quad j = 1, \ldots, n;$$

$$a(x) = \arg\max_{y \in Y}\left(\ln \hat{P}_y - \sum_{j=1}^{n} \ln \hat{\sigma}_{yj} - \sum_{j=1}^{n} \frac{(\xi_j - \hat{\mu}_{yj})^2}{2\hat{\sigma}_{yj}^2}\right);$$

где $x \equiv (\xi_1, \ldots, \xi_n)$; $\hat{\mu}_{yj}$ и $\hat{\sigma}_{yj}$ — оценки среднего и дисперсии j-го признака, вычисленные по X_y — подвыборке класса y.

Задачи, в которых Naïve Bayes иногда неплохо работает

- распознавание спама;
- классификация текстов;
- категоризация текстов;

В этих задачах признаки $f_j(x)$ документа x — это слова:

- $f_j(x) = \big[\text{слово } j \text{ входит в текст } x\big]$;
- $f_j(x) = n_{xj}$ — число вхождений слова j в текст x;
- $f_j(x) = \text{TF-IDF}(x, j)$

Обычно используется предварительный отсев признаков, слабо коррелирующих с вектором ответов y.

TF-IDF(d, w) — мера релевантности слова w документу d

n_{dw} (term frequency) — число вхождений слова w в текст d;
N_w (document frequency) — число документов, содержащих w;
N — число документов в коллекции;
N_w/N — оценка вероятности встретить слово w в документе;
$(N_w/N)^{n_{dw}}$ — оценка вероятности встретить его n_{dw} раз;
$Q = \{w_1, \ldots, w_k\}$ — запрос;

$P(Q, d) = \prod_{w \in Q} (N_w/N)^{n_{dw}}$ — оценка вероятности встретить в документе d слова запроса Q *чисто случайно*;

оценка релевантности запроса Q документу d:

$$-\log P(Q, d) = \sum_{w \in Q} \underbrace{n_{dw}}_{TF} \underbrace{\log(N/N_w)}_{IDF} \to \max.$$

TF-IDF$(d, w) = n_{dw} \log(N/N_w)$,
IDF — inverted document frequency.

Задача построения разделяющей поверхности

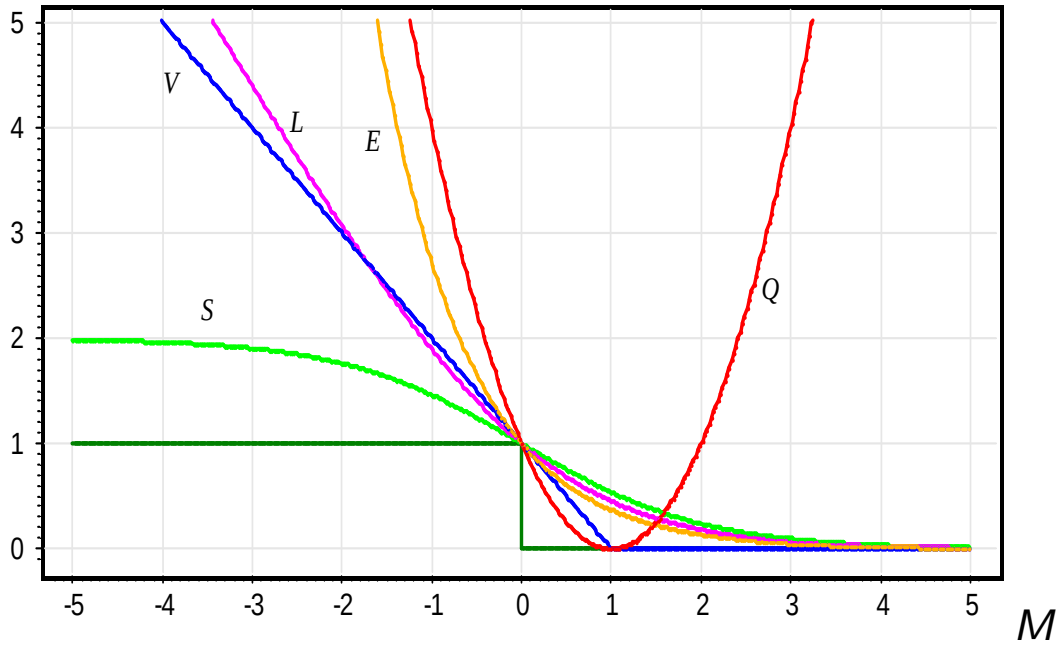
- Задача классификации с двумя классами, $Y = \{-1, +1\}$:
  по обучающей выборке $X^\ell = (x_i, y_i)_{i=1}^\ell$ построить
  алгоритм классификации $a(x, w) = \operatorname{sign} f(x, w)$, где
  $f(x, w)$ — *дискриминантная функция*,
  w — вектор параметров.
- $f(x, w) = 0$ — разделяющая поверхность;
  $M_i(w) = y_i f(x_i, w)$ — *отступ* (margin) объекта x_i;
  $M_i(w) < 0 \iff$ алгоритм $a(x, w)$ ошибается на x_i.
- *Минимизация эмпирического риска*:

$$Q(w) = \sum_{i=1}^{\ell} \big[M_i(w) < 0\big] \quad \leqslant \quad \widetilde{Q}(w) = \sum_{i=1}^{\ell} \mathscr{L}\big(M_i(w)\big) \to \min_w;$$

функция потерь $\mathscr{L}(M)$ невозрастающая, неотрицательная.

Непрерывные аппроксимации пороговой функции потерь

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



$Q(M) = (1 - M)^2$ — квадратичная (ЛДФ);
$V(M) = (1 - M)_+$ — кусочно-линейная (SVM);
$S(M) = 2(1 + e^M)^{-1}$ — сигмоидная (нейросети);
$L(M) = \log_2(1 + e^{-M})$ — логарифмическая (LR);
$E(M) = e^{-M}$ — экспоненциальная (AdaBoost).

Связь с принципом максимума правдоподобия

Пусть $X \times Y$ — в.п. с плотностью $p(x, y|w)$.
Пусть X^ℓ — простая выборка (i.i.d.)

- *Максимизация правдоподобия:*

$$L(w; X^\ell) = \ln \prod_{i=1}^{\ell} p(x_i, y_i|w) = \sum_{i=1}^{\ell} \ln p(x_i, y_i|w) \to \max_w.$$

- *Минимизация аппроксимированного эмпирического риска:*

$$\widetilde{Q}(w; X^\ell) = \sum_{i=1}^{\ell} \mathscr{L}\big(y_i f(x_i, w)\big) \to \min_w;$$

- Эти два принципа эквивалентны, если положить

$$-\ln p(x_i, y_i|w) = \mathscr{L}\big(y_i f(x_i, w)\big).$$

$$\boxed{\text{модель } p} \rightleftarrows \boxed{\text{модель } f \text{ и функция потерь } \mathscr{L}}.$$

Линейный классификатор

$f_j\colon X \to \mathbb{R},\ \ j = 1, \ldots, n$ — числовые признаки;
Линейный алгоритм классификации:

$$a(x, w) = \operatorname{sign}\left(\sum_{j=1}^{n} w_j f_j(x) - w_0\right),$$

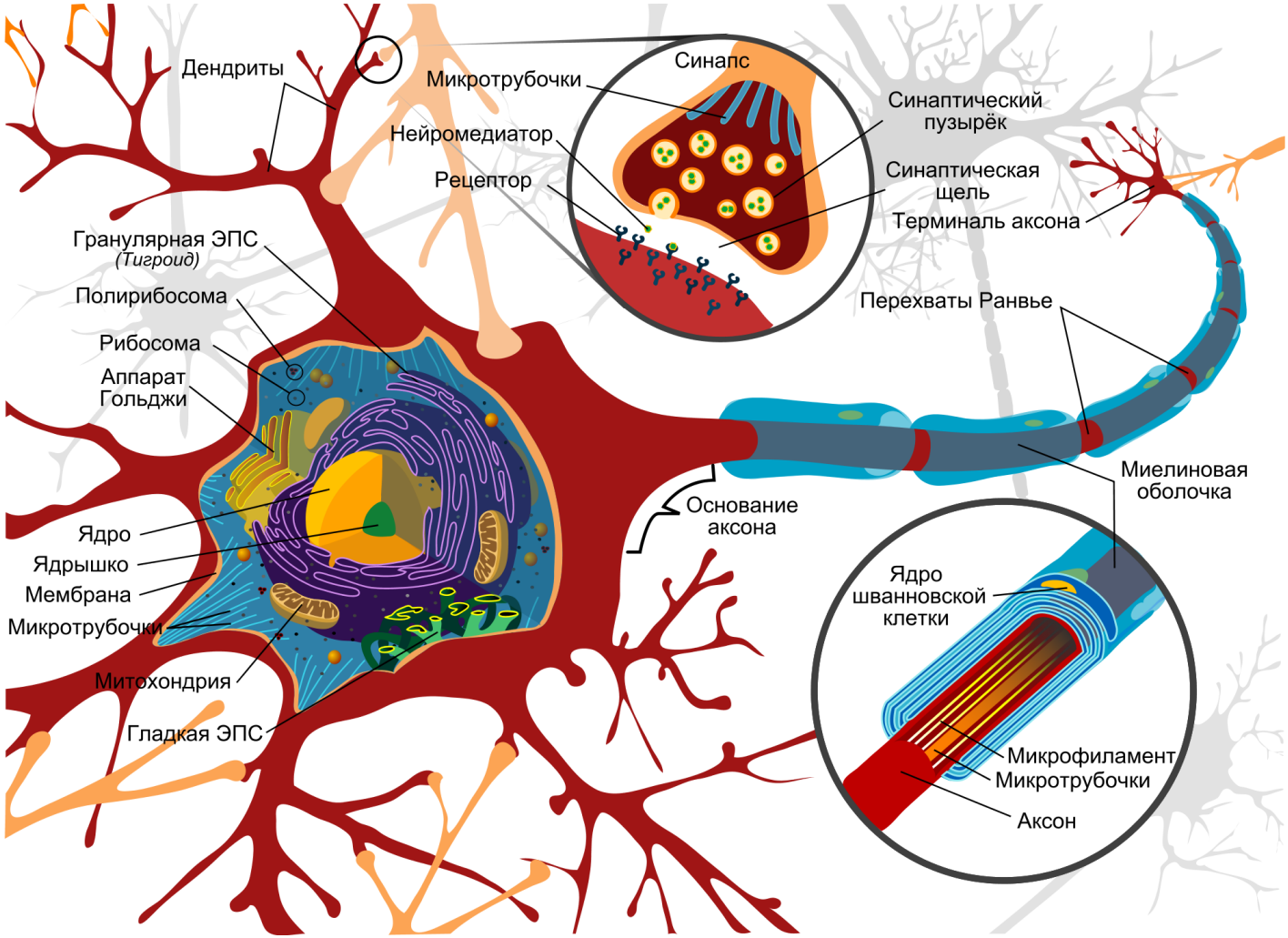
где $w_0, w_1, \ldots, w_n \in \mathbb{R}$ — коэффициенты (веса признаков);
Введём константный признак $f_0 \equiv -1$.
Векторная запись:

$$a(x, w) = \operatorname{sign}\big(\langle w, x\rangle\big).$$

Отступы объектов x_i:

$$M_i(w) = \langle w, x_i\rangle y_i.$$

Похож ли нейрон на линейный классификатор?

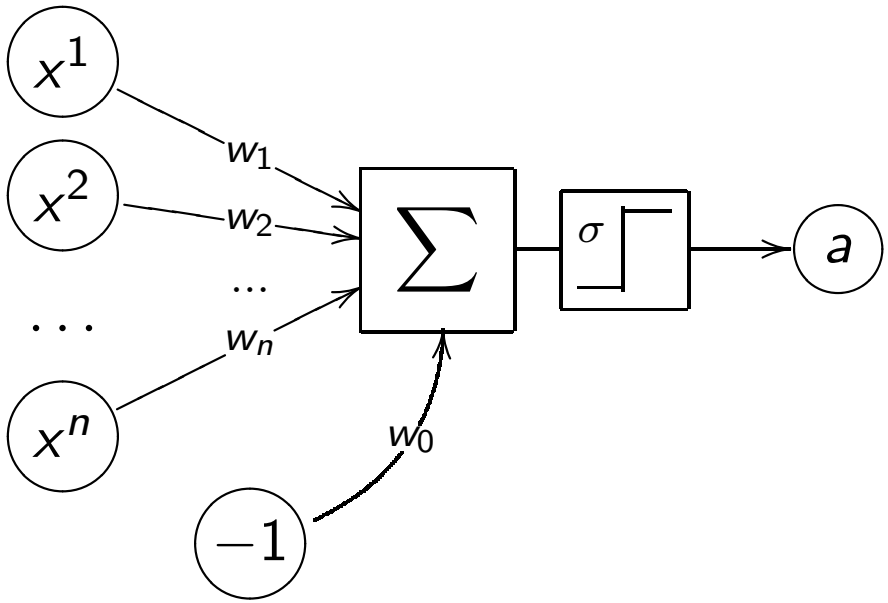


Математическая модель нейрона

Линейная модель нейрона МакКаллока-Питтса [1943]:

$$a(x, w) = \sigma\big(\langle w, x\rangle\big) = \sigma\left(\sum_{j=1}^{n} w_j f_j(x) - w_0\right),$$

где $\sigma(s)$ — функция активации (в частности, sign).



Градиентный метод численной минимизации

Минимизация аппроксимированного эмпирического риска:

$$Q(w; X^\ell) = \sum_{i=1}^{\ell} \mathscr{L}\big(\langle w, x_i\rangle y_i\big) \to \min_w .$$

Численная минимизация методом *градиентного спуска*:

$$w^{(0)} := \text{начальное приближение;}$$

$$w^{(t+1)} := w^{(t)} - \eta \cdot \nabla Q(w^{(t)}), \qquad \nabla Q(w) = \left(\frac{\partial Q(w)}{\partial w_j}\right)_{j=0}^{n},$$

где η — *градиентный шаг*, называемый также *темпом обучения*.

$$w^{(t+1)} := w^{(t)} - \eta \sum_{i=1}^{\ell} \mathscr{L}'\big(\langle w^{(t)}, x_i\rangle y_i\big)\, x_i\, y_i.$$

Идея ускорения сходимости:
брать (x_i, y_i) по одному и сразу обновлять вектор весов.

Алгоритм SG (Stochastic Gradient)

Вход:

выборка X^ℓ; темп обучения η; параметр λ;

Выход:

веса $w_0, w_1, \ldots, w_n$;

1: инициализировать веса w_j, $j = 0, \ldots, n$;
2: инициализировать текущую оценку функционала:
$Q := \sum_{i=1}^{\ell} \mathscr{L}\big(\langle w, x_i\rangle y_i\big)$;
3: **повторять**
4: выбрать объект x_i из X^ℓ (например, случайно);
5: вычислить потерю: $\varepsilon_i := \mathscr{L}\big(\langle w, x_i\rangle y_i\big)$;
6: градиентный шаг: $w := w - \eta \mathscr{L}'\big(\langle w, x_i\rangle y_i\big) x_i y_i$;
7: оценить значение функционала: $Q := (1-\lambda)Q + \lambda\varepsilon_i$;
8: **пока** значение Q и/или веса w не стабилизируются;

Логистическая регрессия: базовые предположения

- $X = \mathbb{R}^n$, $Y = \pm 1$, выборка $X^\ell = (x_i, y_i)_{i=1}^\ell$ i.i.d. из

$$p(x, y) = P_y p_y(x) = \mathsf{P}(y|x)p(x)$$

- Функции правдоподобия $p_y(x)$ *экспонентные*:

$$p_y(x) = \exp\big(c_y(\delta)\langle\theta_y, x\rangle + b_y(\delta, \theta_y) + d(x, \delta)\big),$$

где $\theta_y \in \mathbb{R}^n$ — параметр *сдвига*;
δ — параметр *разброса*;
b_y, c_y, d — произвольные числовые функции;
причём параметры $d(\cdot)$ и δ не зависят от y.

Класс экспонентных распределений очень широк:
равномерное, нормальное, гипергеометрическое, пуассоновское, биномиальное, Г-распределение, и др.

Пример: гауссовская плотность — экспонентная

Многомерное нормальное распределение, $\mu \in \mathbb{R}^n$, $\Sigma \in \mathbb{R}^{n \times n}$, является экспонентным:

параметр сдвига $\theta = \Sigma^{-1}\mu$;

параметр разброса $\delta = \Sigma$.

$$\mathcal{N}(x;\mu,\Sigma) = (2\pi)^{-\frac{n}{2}}|\Sigma|^{-\frac{1}{2}} \exp\Big(-\tfrac{1}{2}(x-\mu)^{\mathsf{T}}\Sigma^{-1}(x-\mu)\Big) =$$

$$= \exp\Big(\underbrace{\mu^{\mathsf{T}}\Sigma^{-1}x}_{\langle\theta,x\rangle} - \underbrace{\tfrac{1}{2}\mu^{\mathsf{T}}\Sigma^{-1}\Sigma\Sigma^{-1}\mu}_{b(\delta,\theta)} -$$

$$- \underbrace{\tfrac{1}{2}x^{\mathsf{T}}\Sigma^{-1}x - \tfrac{n}{2}\ln(2\pi) - \tfrac{1}{2}\ln|\Sigma|}_{d(x,\delta)}\Big).$$

Основная теорема

Оптимальный байесовский классификатор для двух классов:

$$a(x) = \text{sign}\big(\text{P}(+1|x) - \text{P}(-1|x)\big) = \text{sign}\left(\frac{p_+(x)}{p_-(x)} - \frac{P_-}{P_+}\right).$$

Теорема

Если p_y экспонентны, параметры $d(\cdot)$ и δ не зависят от y,
и среди признаков $f_1(x), \ldots, f_n(x)$ есть константа,
то байесовский классификатор линеен:

$$a(x) = \text{sign}\langle w, x\rangle,$$

апостериорные вероятности классов:

$$\text{P}(y|x) = \sigma\big(\langle w, x\rangle y\big),$$

где $\sigma(z) = \frac{1}{1+e^{-z}}$ — *логистическая* (*сигмоидная*) функция.

Обоснование логарифмической функции потерь

Максимизация логарифма правдоподобия выборки:

$$L(w, X^\ell) = \log \prod_{i=1}^{\ell} p(x_i, y_i) \to \max_w .$$

Подставим: $p(x, y) = \mathrm{P}(y|x) \cdot p(x) = \sigma\big(\langle w, x\rangle y\big) \cdot \mathrm{const}(w)$

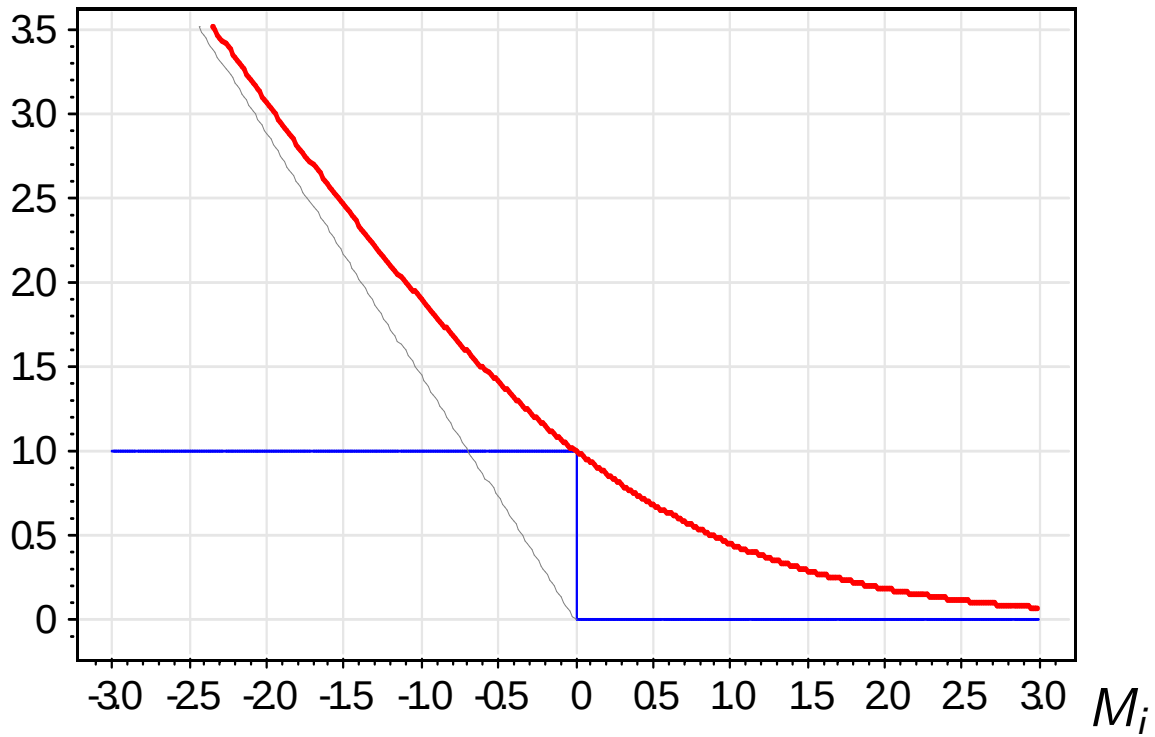
$$L(w, X^\ell) = \sum_{i=1}^{\ell} \log \sigma\big(\langle w, x_i\rangle y_i\big) + \mathrm{const}(w) \to \max_w .$$

Максимизация $L(w)$ эквивалентна минимизации $\widetilde{Q}(w)$:

$$\widetilde{Q}(w, X^\ell) = \sum_{i=1}^{\ell} \log\big(1 + \exp(-\underbrace{\langle w, x_i\rangle y_i}_{M_i(w)})\big) \to \min_w .$$

Логарифмическая функция потерь

Логарифмическая функция потерь $\mathscr{L}(M_i) = \log_2\left(1 + e^{-M_i}\right)$:



Градиентный метод

Производная сигмоидной функции: $\sigma'(z) = \sigma(z)\sigma(-z)$.
Вектор градиента функционала $\widetilde{Q}(w)$:

$$\nabla\widetilde{Q}(w) = -\sum_{i=1}^{\ell} y_i x_i \sigma\big(-\langle w, x_i\rangle y_i\big).$$

Градиентный шаг в методе стохастического градиента:

$$w := w + \eta\, y_i x_i\, \sigma\big(-\langle w, x_i\rangle y_i\big),$$

где (x_i, y_i) — предъявляемый прецедент, η — темп обучения.

Интерпретация:
чем меньше отступ, тем сильнее надо изменить w.

Бинаризация признаков и скоринговая карта

Возраст	до 25	5
	25 - 40	10
	40 - 50	15
	50 и больше	10
Собственность	владелец	20
	совладелец	15
	съемщик	10
	другое	5
Работа	руководитель	15
	менеджер среднего звена	10
	служащий	5
	другое	0
Стаж	1/безработный	0
	1..3	5
	3..10	10
	10 и больше	15
Работа_мужа /жены	нет/домохозяйка	0
	руководитель	10
	менеджер среднего звена	5
	служащий	1

Оценивание рисков

Оценка *риска* (математического ожидания) потерь объекта x:

$$R(x) = \sum_{y \in Y} D_{xy} \, \mathrm{P}(y|x) = \sum_{y \in Y} D_{xy} \sigma\big(\langle w, x \rangle y\big),$$

где D_{xy} — величина потери для (x, y).

Преимущество бинаризации признаков:

биномиальное распределение является экспонентным.

Методика VaR (Value at Risk):

— 1000 раз: для каждого x разыгрывается исход y с вероятностью $\mathrm{P}(y|x) = \sigma\big(\langle w, x \rangle y\big)$;

— строится эмпирическое распределение величины $V = \sum_{i=1}^{\ell} D_{xy}$;

— определяется α-квантиль распределения.

Принцип максимума ширины разделяющей полосы

Линейный классификатор:

$$a(x) = \operatorname{sign}\big(\langle w, x\rangle - w_0\big), \quad w, x \in \mathbb{R}^n,\ w_0 \in \mathbb{R}.$$

Пусть выборка $X^\ell = (x_i, y_i)_{i=1}^\ell$ линейно разделима:

$$\exists w, w_0: \quad M_i(w, w_0) = y_i\big(\langle w, x_i\rangle - w_0\big) > 0, \quad i = 1, \ldots, \ell$$

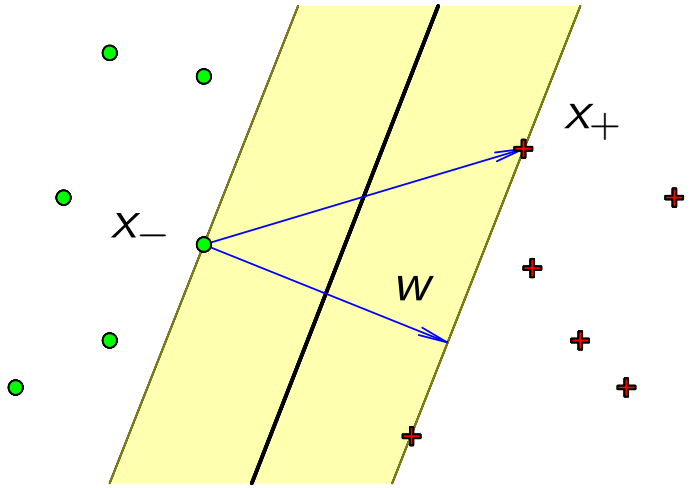
Нормировка: $\min\limits_{i=1,\ldots,\ell} M_i(w, w_0) = 1.$

Разделяющая полоса:

$$\big\{x:\ -1 \leqslant \langle w, x\rangle - w_0 \leqslant 1\big\}.$$

Ширина полосы:

$$\frac{\langle x_+ - x_-, w\rangle}{\|w\|} = \frac{2}{\|w\|} \to \max.$$



Обоснование кусочно-линейной функции потерь

Линейно разделимая выборка

$$\begin{cases} \|w\|^2 \to \min\limits_{w,w_0}; \\ M_i(w,w_0) \geqslant 1, \quad i = 1,\ldots,\ell. \end{cases}$$

Переход к линейно неразделимой выборке (эвристика)

$$\begin{cases} \frac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{\ell}\xi_i \to \min\limits_{w,w_0,\xi}; \\ M_i(w,w_0) \geqslant 1 - \xi_i, \quad i = 1,\ldots,\ell; \\ \xi_i \geqslant 0, \quad i = 1,\ldots,\ell. \end{cases}$$

Эквивалентная задача безусловной минимизации:

$$Q(w,w_0) = \sum_{i=1}^{\ell}\bigl(1 - M_i(w,w_0)\bigr)_+ + \frac{1}{2C}\|w\|^2 \to \min_{w,w_0}.$$

Задача поиска седловой точки функции Лагранжа

Функция Лагранжа: $\mathscr{L}(w, w_0, \xi; \lambda, \eta) =$

$$= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \lambda_i \big(M_i(w, w_0) - 1\big) - \sum_{i=1}^{\ell} \xi_i \big(\lambda_i + \eta_i - C\big),$$

λ_i — переменные, двойственные к ограничениям $M_i \geqslant 1 - \xi_i$;
η_i — переменные, двойственные к ограничениям $\xi_i \geqslant 0$.

$$\begin{cases} \mathscr{L}(w, w_0, \xi; \lambda, \eta) \;\to\; \min\limits_{w, w_0, \xi} \; \max\limits_{\lambda, \eta}; \\ \xi_i \geqslant 0, \quad \lambda_i \geqslant 0, \quad \eta_i \geqslant 0, \quad i = 1, \ldots, \ell; \\ \lambda_i = 0 \;\text{ либо }\; M_i(w, w_0) = 1 - \xi_i, \quad i = 1, \ldots, \ell; \\ \eta_i = 0 \;\text{ либо }\; \xi_i = 0, \quad i = 1, \ldots, \ell; \end{cases}$$

Необходимые условия седловой точки функции Лагранжа

Функция Лагранжа: $\mathscr{L}(w, w_0, \xi; \lambda, \eta) =$

$$= \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \lambda_i \big(M_i(w, w_0) - 1\big) - \sum_{i=1}^{\ell} \xi_i \big(\lambda_i + \eta_i - C\big),$$

Необходимые условия седловой точки функции Лагранжа:

$$\frac{\partial \mathscr{L}}{\partial w} = w - \sum_{i=1}^{\ell} \lambda_i y_i x_i = 0 \quad \Longrightarrow \quad w = \sum_{i=1}^{\ell} \lambda_i y_i x_i;$$

$$\frac{\partial \mathscr{L}}{\partial w_0} = -\sum_{i=1}^{\ell} \lambda_i y_i = 0 \quad \Longrightarrow \quad \sum_{i=1}^{\ell} \lambda_i y_i = 0;$$

$$\frac{\partial \mathscr{L}}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 \quad \Longrightarrow \quad \eta_i + \lambda_i = C, \quad i = 1, \ldots, \ell.$$

Понятие опорного вектора

Типизация объектов:

1. $\lambda_i = 0;\ \ \eta_i = C;\ \ \xi_i = 0;\ \ M_i \geqslant 1.$
   — периферийные (неинформативные) объекты.

2. $0 < \lambda_i < C;\ \ 0 < \eta_i < C;\ \ \xi_i = 0;\ \ M_i = 1.$
   — опорные граничные объекты.

3. $\lambda_i = C;\ \ \eta_i = 0;\ \ \xi_i > 0;\ \ M_i < 1.$
   — опорные-нарушители.

Определение

Объект x_i называется *опорным*, если $\lambda_i \neq 0$.

Двойственная задача

$$\begin{cases} -\mathscr{L}(\lambda) = -\sum\limits_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum\limits_{i=1}^{\ell} \sum\limits_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle \;\to\; \min\limits_{\lambda}; \\ 0 \leqslant \lambda_i \leqslant C, \quad i = 1, \ldots, \ell; \\ \sum\limits_{i=1}^{\ell} \lambda_i y_i = 0. \end{cases}$$

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} w = \sum\limits_{i=1}^{\ell} \lambda_i y_i x_i; \\ w_0 = \langle w, x_i \rangle - y_i, \qquad \text{для любого } i\text{: } \lambda_i > 0, \;\; M_i = 1. \end{cases}$$

Линейный классификатор:

$$a(x) = \operatorname{sign}\Big(\sum_{i=1}^{\ell} \lambda_i y_i \langle x_i, x \rangle - w_0 \Big).$$

Нелинейное обобщение SVM

Переход к спрямляющему пространству более высокой размерности: $\psi\colon X \to H$.

Определение

Функция $K\colon X \times X \to \mathbb{R}$ — *ядро*, если $K(x, x') = \langle \psi(x), \psi(x') \rangle$ при некотором $\psi\colon X \to H$, где H — гильбертово пространство.

Теорема

Функция $K(x, x')$ является ядром тогда и только тогда, когда
- она симметрична: $K(x, x') = K(x', x)$;
- и неотрицательно определена: $\int_X \int_X K(x, x') g(x) g(x') dx dx' \geqslant 0$ для любой $g\colon X \to \mathbb{R}$.

Конструктивные методы синтеза ядер

1. $K(x, x') = \langle x, x' \rangle$ — ядро;
2. константа $K(x, x') = 1$ — ядро;
3. произведение ядер $K(x, x') = K_1(x, x')K_2(x, x')$ — ядро;
4. $\forall \psi : X \to \mathbb{R}$ произведение $K(x, x') = \psi(x)\psi(x')$ — ядро;
5. $K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x')$ при $\alpha_1, \alpha_2 > 0$ — ядро;
6. $\forall \varphi : X \to X$ если K_0 ядро, то $K(x, x') = K_0(\varphi(x), \varphi(x'))$ — ядро;
7. если $s\colon X \times X \to \mathbb{R}$ — симметричная интегрируемая функция, то $K(x, x') = \int_X s(x, z)s(x', z)\, dz$ — ядро;
8. если K_0 — ядро и функция $f : \mathbb{R} \to \mathbb{R}$ представима в виде сходящегося степенного ряда с неотрицательными коэффициентами, то $K(x, x') = f\big(K_0(x, x')\big)$ — ядро;

Пример: спрямляющее пространство для квадратичного ядра

Пусть $X = \mathbb{R}^2$, $K(u,v) = \langle u,v\rangle^2$, где $u = (u_1,u_2)$, $v = (v_1,v_2)$.

Задача: найти пространство H и преобразование $\psi\colon X \to H$, при которых $K(x,x') = \langle \psi(x), \psi(x')\rangle_H$.

Разложим квадрат скалярного произведения:

$$\begin{aligned} K(u,v) = \langle u,v\rangle^2 &= \langle (u_1,u_2),(v_1,v_2)\rangle^2 = \\ &= (u_1v_1 + u_2v_2)^2 = u_1^2v_1^2 + u_2^2v_2^2 + 2u_1v_1u_2v_2 = \\ &= \langle (u_1^2, u_2^2, \sqrt{2}u_1u_2), (v_1^2, v_2^2, \sqrt{2}v_1v_2)\rangle. \end{aligned}$$

Таким образом,

$$H = \mathbb{R}^3, \quad \psi\colon (u_1,u_2) \mapsto \left(u_1^2, u_2^2, \sqrt{2}u_1u_2\right).$$

Линейной поверхности в пространстве H соответствует квадратичная поверхность в исходном пространстве X.

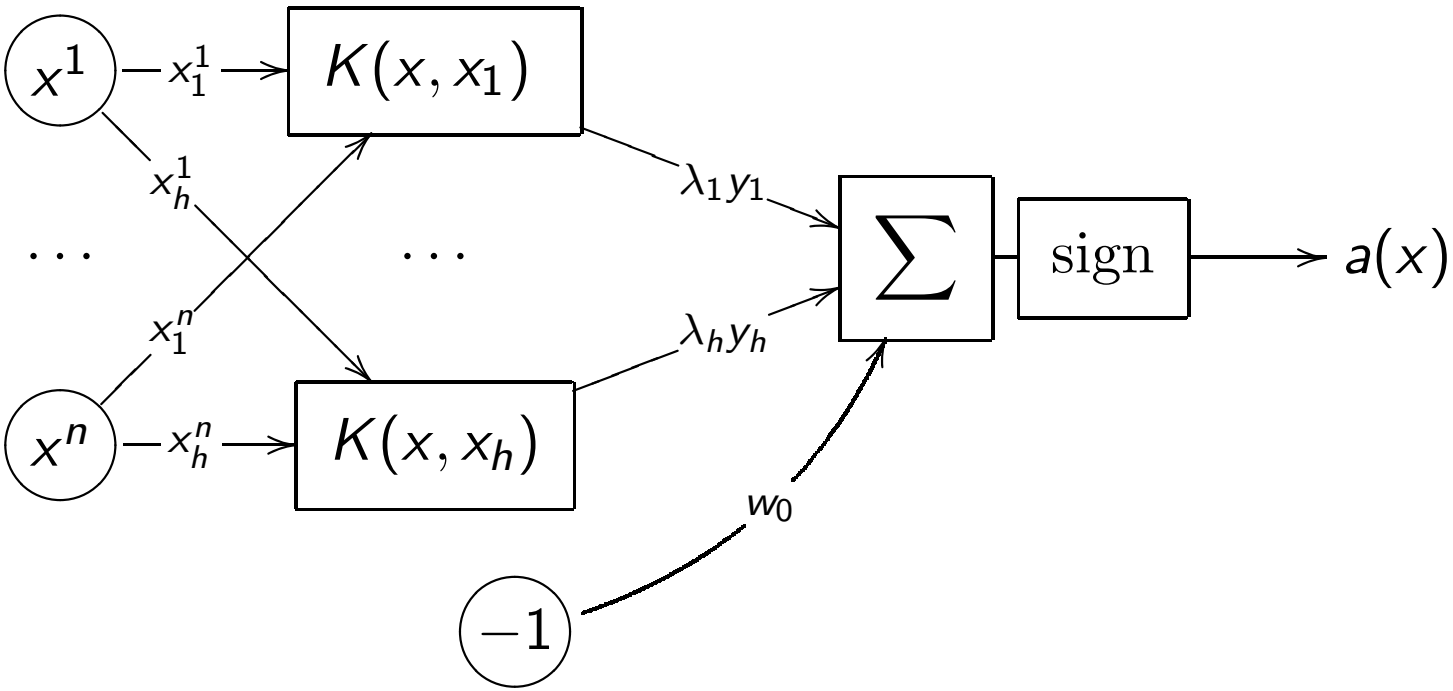
Примеры ядер

1. $K(x,x') = \langle x,x' \rangle^2$
   — квадратичное ядро;
2. $K(x,x') = \langle x,x' \rangle^d$
   — полиномиальное ядро с мономами степени d;
3. $K(x,x') = \big(\langle x,x' \rangle + 1\big)^d$
   — полиномиальное ядро с мономами степени $\leqslant d$;
4. $K(x,x') = \operatorname{th}\big(k_0 + k_1\langle x,x' \rangle\big)$
   — нейросеть с сигмоидными функциями активации;
5. $K(x,x') = \exp\big(-\beta\|x-x'\|^2\big)$
   — сеть радиальных базисных функций;

SVM как двухслойная нейронная сеть

Перенумеруем объекты так, чтобы $x_1, \ldots, x_h$ были опорными.

$$a(x) = \text{sign}\left(\sum_{i=1}^{h} \lambda_i y_i K(x, x_i) - w_0\right).$$



Преимущества и недостатки SVM

Преимущества SVM перед SG:

- Задача выпуклого квадратичного программирования имеет единственное решение.
- Число нейронов скрытого слоя определяется автоматически — это число опорных векторов.

Недостатки SVM:

- Неустойчивость к шуму.
- Нет общих подходов к оптимизации $K(x, x')$ под задачу.
- Приходится подбирать константу C.

Модель смеси распределений

Модель плотности:

$$p(x) = \sum_{j=1}^{k} w_j p_j(x), \qquad \sum_{j=1}^{k} w_j = 1, \qquad w_j \geqslant 0,$$

$p_j(x) = \varphi(x; \theta_j)$ — *функция правдоподобия* j-й компоненты смеси;
w_j — её *априорная вероятность*; k — число компонент смеси.

Задача: имея простую выборку $X^m \sim p(x)$,
зная число k и функцию φ, оценить вектор параметров
$\Theta = (w_1, \ldots, w_k, \theta_1, \ldots, \theta_k)$.

Общая схема EM-алгоритма

Проблема:
попытка применить принцип максимума правдоподобия «в лоб» приводит к очень сложной многоэкстремальной задаче оптимизации

Идея: вводятся *скрытые переменные* G.

Итерационный алгоритм Expectation–Maximization:

1: начальное приближение вектора параметров Θ;
2: **повторять**
3: $\quad G := \text{E-шаг}\,(\Theta)$;
4: $\quad \Theta := \text{M-шаг}\,(\Theta, G)$;
5: **пока** Θ и G не стабилизируются.

Задача E-шага

По формуле условной вероятности

$$p(x, \theta_j) = p(x)\,\mathrm{P}(\theta_j|x) = w_j p_j(x).$$

Скрытые переменные $G = (g_{ij})_{m\times k} = (g_1, \ldots, g_j)$:

$$g_{ij} \equiv \mathrm{P}(\theta_j|x_i), \quad i = 1, \ldots, m,\ j = 1, \ldots, k.$$

Зная параметры компонент w_j, θ_j, по формуле Байеса легко вычислить g_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, k$:

$$g_{ij} = \frac{w_j p_j(x_i)}{p(x_i)} = \frac{w_j p_j(x_i)}{\sum_{s=1}^{k} w_s p_s(x_i)}.$$

Очевидно, выполнено условие нормировки: $\sum_{j=1}^{k} g_{ij} = 1$.

Задача M-шага

Задача: максимизировать логарифм правдоподобия

$$Q(\Theta) = \ln \prod_{i=1}^{m} p(x_i) = \sum_{i=1}^{m} \ln \sum_{j=1}^{k} w_j p_j(x_i) \to \max_{\Theta}.$$

при ограничениях $\sum_{j=1}^{k} w_j = 1;\ \ w_j \geqslant 0.$

Если скрытые переменные известны, то задача максимизации $Q(\Theta)$ распадается на k независимых подзадач:

$$\theta_j := \arg\max_{\theta} \sum_{i=1}^{m} g_{ij} \ln \varphi(x_i; \theta), \quad j = 1, \ldots, k.$$

а оптимальные веса компонент вычисляются аналитически:

$$w_j := \frac{1}{m} \sum_{i=1}^{m} g_{ij}, \quad j = 1, \ldots, k.$$

Вывод формул M-шага (основные шаги)

Лагранжиан оптимизационной задачи «$Q(\Theta) \to \max$»:

$$L(\Theta; X^m) = \sum_{i=1}^{m} \ln \Bigg(\underbrace{\sum_{j=1}^{k} w_j p_j(x_i)}_{p(x_i)} \Bigg) - \lambda \Bigg(\sum_{j=1}^{k} w_j - 1 \Bigg).$$

Приравниваем нулю производные:

$$\frac{\partial L}{\partial w_j} p_j(x_i) = 0 \quad \Rightarrow \quad \lambda = m; \quad w_j = \frac{1}{m} \sum_{i=1}^{m} \underbrace{\frac{w_j p_j(x_i)}{p(x_i)}}_{g_{ij}} = \frac{1}{m} \sum_{i=1}^{m} g_{ij},$$

$$\frac{\partial L}{\partial \theta_j} = \sum_{i=1}^{m} \underbrace{\frac{w_j p_j(x_i)}{p(x_i)}}_{g_{ij}} \frac{\partial}{\partial \theta_j} \ln p_j(x_i) = \frac{\partial}{\partial \theta_j} \sum_{i=1}^{m} g_{ij} \ln p_j(x_i) = 0.$$

EM-алгоритм

Вход:

$X^m = \{x_1, \ldots, x_m\}$, k, δ — параметр критерия останова,

$\Theta = (w_j, \theta_j)_{j=1}^k$ — начальное приближение параметров;

Выход:

$\Theta = (w_j, \theta_j)_{j=1}^k$ — оптимизированный вектор параметров.

1: **повторять**

2: E-шаг (expectation):

$$g_{ij}^0 := g_{ij}; \quad g_{ij} := \frac{w_j \varphi(x_i; \theta_j)}{\sum_{s=1}^k w_s \varphi(x_i; \theta_s)}, \quad i = 1..m, \quad j = 1..k;$$

3: M-шаг (maximization):

$$\theta_j := \arg\max_\theta \sum_{i=1}^m g_{ij} \ln \varphi(x_i; \theta), \qquad w_j := \frac{1}{m} \sum_{i=1}^m g_{ij}, \qquad j = 1..k$$

4: **пока** $\max\limits_{i,j} |g_{ij} - g_{ij}^0| > \delta$;

5: **вернуть** $(w_j, \theta_j)_{j=1}^k$;

Гауссовская смесь с диагональными матрицами ковариации

Байесовский классификатор: $a(x) = \arg\max_{y \in Y} P_y p_y(x)$.

Допущения:

1. Функции правдоподобия классов $p_y(x)$ представимы в виде смесей k_y компонент, $y \in Y = \{1, \ldots, M\}$.
2. Компоненты имеют n-мерные гауссовские плотности с некоррелированными признаками:

$\mu_{yj} = (\mu_{yj1}, \ldots, \mu_{yjn})$, $\Sigma_{yj} = \operatorname{diag}(\sigma^2_{yj1}, \ldots, \sigma^2_{yjn})$, $j = 1, \ldots, k_y$:

$$p_y(x) = \sum_{j=1}^{k_y} w_{yj} p_{yj}(x), \quad p_{yj}(x) = \mathcal{N}(x; \mu_{yj}, \Sigma_{yj}),$$

$$\sum_{j=1}^{k_y} w_{yj} = 1, \quad w_{yj} \geqslant 0;$$

Эмпирические оценки средних и дисперсий

Числовые признаки: $f_d : X \to \mathbb{R}$, $d = 1, \ldots, n$.

Решение задачи M-шага:

для всех классов $y \in Y$ и всех компонент $j = 1, \ldots, k_y$,

$$w_{yj} = \frac{1}{\ell_y} \sum_{i:\, y_i = y} g_{yij}$$

для всех размерностей (признаков) $d = 1, \ldots, n$

$$\hat{\mu}_{yjd} = \frac{1}{\ell_y w_{yj}} \sum_{i:\, y_i = y} g_{yij} f_d(x_i);$$

$$\hat{\sigma}^2_{yjd} = \frac{1}{\ell_y w_{yj}} \sum_{i:\, y_i = y} g_{yij} \big(f_d(x_i) - \hat{\mu}_{yjd}\big)^2;$$

Замечание: компоненты «наивны», но смесь не «наивна».

Алгоритм классификации

Подставим гауссовскую смесь в байесовский классификатор:

$$a(x) = \arg\max_{y \in Y} P_y \sum_{j=1}^{k_y} w_{yj} \underbrace{\mathcal{N}_{yj} \exp\left(-\tfrac{1}{2}\rho^2_{yj}(x, \mu_{yj})\right)}_{p_{yj}(x)},$$

$\mathcal{N}_{yj} = (2\pi)^{-\frac{n}{2}}(\sigma_{yj1} \cdots \sigma_{yjn})^{-1}$ — нормировочные множители;
$\rho_{yj}(x, \mu_{yj})$ — *взвешенная евклидова метрика* в $X = \mathbb{R}^n$:

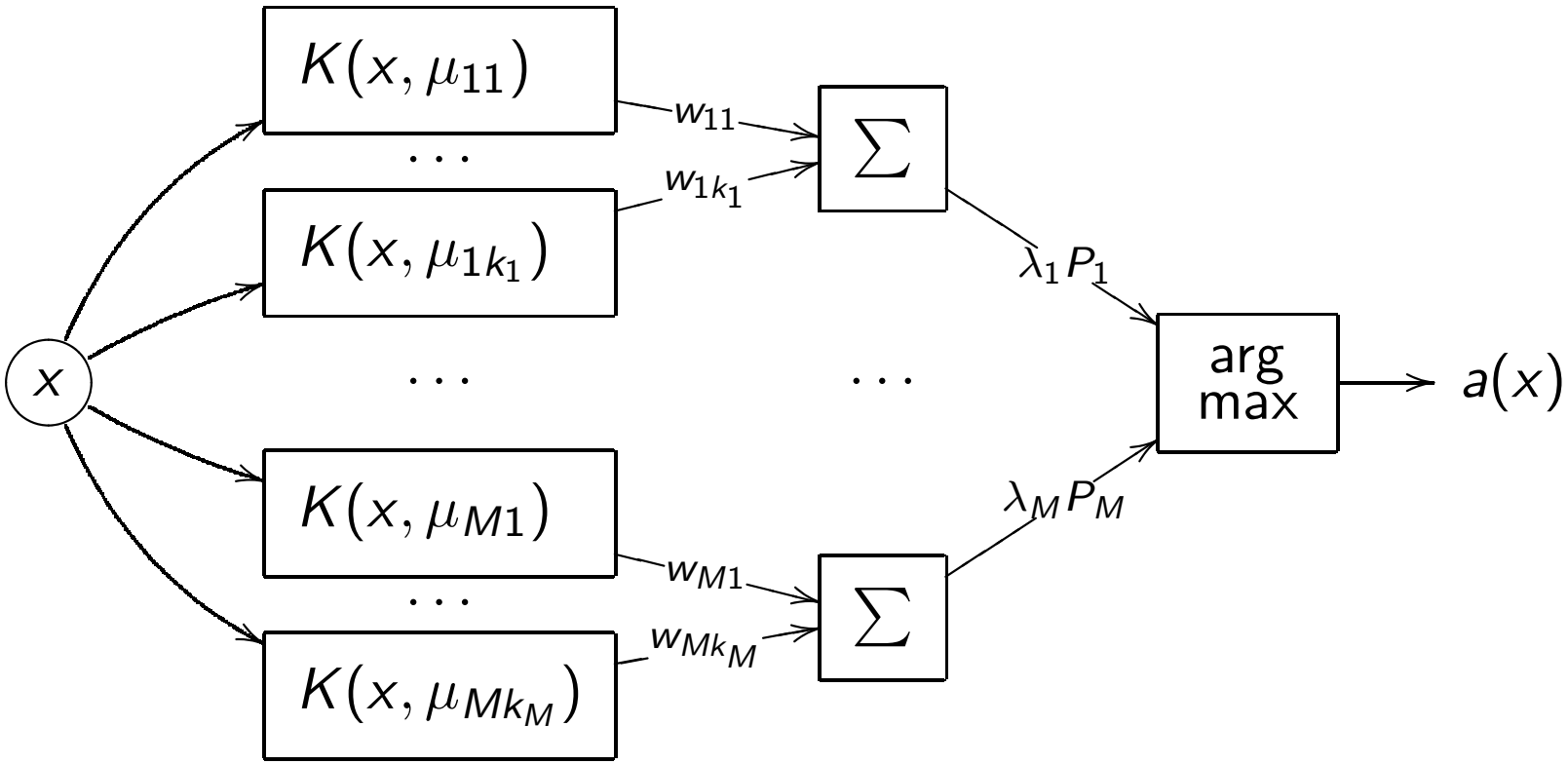
$$\rho^2_{yj}(x, \mu_{yj}) = \sum_{d=1}^{n} \frac{1}{\sigma^2_{yjd}} \big(f_d(x) - \mu_{yjd}\big)^2.$$

Плотности очень похожи на гауссовские ядра в SVM:

$$p_{yj}(x) \equiv K(x, \mu_{yj}).$$

Сеть радиальных базисных функций

Radial Basis Functions (RBF) — трёхуровневая суперпозиция:



Преимущества EM-RBF

EM — один из лучших алгоритмов обучения радиальных сетей.

Преимущества EM-алгоритма:

1. EM-алгоритм легко сделать устойчивым к шуму
2. EM-алгоритм довольно быстро сходится
3. автоматически строится *структурное описание* каждого класса в виде совокупности компонент — *кластеров*

Недостатки EM-алгоритма:

1. EM-алгоритм чувствителен к начальному приближению

Сравнение EM-RBF и SVM с гауссовским ядром:

1. в SVM объект x сравнивается с опорными объектами
2. в EM-RBF объект x сравнивается с центрами кластеров

Постановка задачи кластеризации

Дано:

X — пространство объектов;

$X^\ell = \{x_i\}_{i=1}^\ell$ — обучающая выборка;

$\rho\colon X \times X \to [0, \infty)$ — функция расстояния между объектами.

Найти:

Y — множество кластеров и

$a\colon X \to Y$ — алгоритм кластеризации, такие, что:

— каждый кластер состоит из близких объектов;

— объекты разных кластеров существенно различны.

Кластеризация — это *обучение без учителя*.

Метод k-средних (k-means)

$X = \mathbb{R}^n$. Упрощённый аналог EM-алгоритма:

1: начальное приближение центров μ_y, $y \in Y$;
2: **повторять**
3: аналог E-шага:
отнести каждый x_i к ближайшему центру:
$y_i := \arg\min\limits_{y \in Y} \rho(x_i, \mu_y)$, $i = 1, \ldots, \ell$;
4: аналог M-шага:
вычислить новые положения центров:
$$\mu_{yj} := \frac{\sum_{i=1}^{\ell} [y_i = y] f_j(x_i)}{\sum_{i=1}^{\ell} [y_i = y]}, \quad y \in Y, \quad j = 1, \ldots, n;$$
5: **пока** y_i не перестанут изменяться;

Модификации и обобщения

Варианты k-means:

- вариант Болла-Холла (на предыдущем слайде);
- вариант МакКина: при каждом переходе объекта из кластера в кластер их центры пересчитываются;

Основные отличия EM и k-means:

- EM: мягкая кластеризация: $g_{iy} = \mathsf{P}\{y_i = y\}$;
  k-m: жёсткая кластеризация: $g_{iy} = [y_i = y]$;
- EM: форма кластеров эллиптическая, настраиваемая;
  k-m: форма кластеров жёстко определяется метрикой ρ;

Гибридные варианты по пути упрощения EM:

- EM с жёсткой кластеризацией на E-шаге;
- EM без настройки дисперсий (сферические гауссианы);

Частичное обучение (SSL, semi-supervised learning)

Дано:

Y — множество кластеров;

$\{x_i\}_{i=1}^{\ell}$ — обучающая выборка;

$\{x_i, y_i\}_{i=\ell+1}^{\ell+m}$ — размеченная часть выборки, обычно $m \ll \ell$.

Найти:

$a\colon X \to Y$ — алгоритм кластеризации.

Как приспособить EM-алгоритм:

E-шаг: $g_{iy} := [y = y_i]$, $y \in Y$, $i = \ell{+}1, \ldots, \ell{+}m$;

Как приспособить k-means:

E-шаг: $y_i := \arg\min\limits_{y \in Y} \rho(x_i, \mu_y)$, $i = 1, \ldots, \ell$.

Задача тематического моделирования коллекции документов

Дано:

W — словарь, множество слов (терминов);

D — множество (коллекция, корпус) текстовых документов;

n_{dw} — сколько раз термин $w \in W$ встретился в документе $d \in D$.

Найти (задача би-кластеризации):

— к каким темам относится каждый документ

— какими терминами определяется каждая тема

— сколько тем содержится в коллекции

Гипотеза «мешка слов»:

порядок терминов не важен для определения тематики текста.

Гипотеза разреженности:

документ, как правило, относится к небольшому числу тем;

тема, как правило, определяется небольшим числом терминов.

Вероятностная формализация постановки задачи

Вероятностные предположения:

- каждое слово в документе связано с некоторой темой $t \in T$;
- коллекция D — это выборка независимых наблюдений (d, w) из дискретного распределения $p(d, w, t)$ на $D \times W \times T$;
- гипотеза условной независимости: $p(w|d, t) = p(w|t)$;

Вероятностная модель порождения документа d:

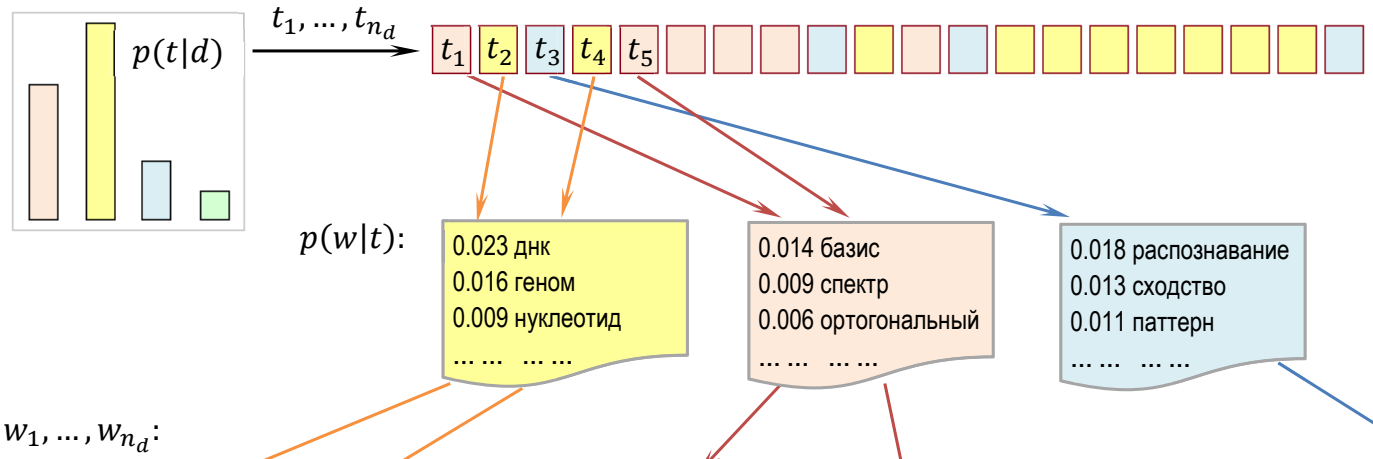
$$p(w|d) = \sum_{t \in T} p(w|d, t)\, p(t|d) = \boxed{\sum_{t \in T} p(w|t)\, p(t|d)}$$

Найти:

- $p(w|t)$ — распределение терминов в каждой теме $t \in T$;
- $p(t|d)$ — распределение тем в каждом документе $d \in D$.

Вероятностная тематическая модель порождения документа d

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d)$$



Разработан спектрально-аналитический подход к выявлению размытых протяженных повторов в геномных последовательностях. Метод основан на разномасштабном оценивании сходства нуклеотидных последовательностей в пространстве коэффициентов разложения фрагментов кривых GC- и GA-содержания по классическим ортогональным базисам. Найдены условия оптимальной аппроксимации, обеспечивающие автоматическое распознавание повторов различных видов (прямых и инвертированных, а также тандемных) на спектральной матрице сходства. Метод одинаково хорошо работает на разных масштабах данных. Он позволяет выявлять следы сегментных дупликаций и мегасателлитные участки в геноме, районы синтении при сравнении пары геномов. Его можно использовать для детального изучения фрагментов хромосом (поиска размытых участков с умеренной длиной повторяющегося паттерна).

Частотные оценки условных вероятностей

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d)$$

Если рассматривать коллекцию как выборку троек (d, w, t), то

$$\hat{p}(w|d) = \frac{n_{dw}}{n_d}, \qquad \hat{p}(w|t) = \frac{n_{wt}}{n_t}, \qquad \hat{p}(t|d) = \frac{n_{dt}}{n_d};$$

n_{dwt} — число троек (d, w, t) во всей коллекции;

$n_{dw} = \sum_{t \in T} n_{dwt}$ — число вхождений термина w в документ d;

$n_{dt} = \sum_{w \in d} n_{dwt}$; $\quad n_d = \sum_{w \in d} \sum_{t \in T} n_{dwt}$ — длина документа d;

$n_{wt} = \sum_{d \in D} n_{dwt}$; $\quad n_t = \sum_{d \in D} \sum_{w \in d} n_{dwt}$ — «длина темы» t;

$n = \sum_{d \in D} \sum_{w \in d} \sum_{t \in T} n_{dwt}$ — длина всей коллекции;

Цели тематического моделирования (topic modeling)

- Тематический поиск документов по тексту любой длины
- Категоризация, классификация, аннотирование, суммаризация текстовых документов
- Тематический поиск объектов, связанных с документами: рисунков, авторов, организаций, журналов, конференций
- Выявление трендов и фронта исследований

Типичные приложения:

- Поиск научной информации
- Анализ и агрегирование новостных потоков
- Рекомендательные сервисы (коллаборативная фильтрация)
- Рубрикация коллекций изображений, видео, музыки
- Аннотация генома и другие задачи биоинформатики

Вероятностный латентно-семантический анализ PLSA — Probabilistic Latent Semantic Analysis [Hofmann, 1999]

Максимизация правдоподобия по $\varphi_{wt} = p(w|t)$, $\theta_{td} = p(t|d)$:

$$\mathscr{L}(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln p(w|d) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \varphi_{wt} \theta_{td} \to \max_{\Phi, \Theta},$$

при ограничениях неотрицательности и нормировки

$$\varphi_{wt} \geqslant 0; \quad \sum_{w \in W} \varphi_{wt} = 1; \qquad \theta_{td} \geqslant 0; \quad \sum_{d \in D} \theta_{td} = 1.$$

Это задача неотрицательного матричного разложения $F \approx \Phi\Theta$,

$F = \big(\hat{p}(w|d)\big)_{W \times D}$ — известная матрица исходных данных;
$\Phi = \big(\varphi_{wt}\big)_{W \times T}$ — искомая матрица терминов тем $\varphi_{wt} = p(w|t)$;
$\Theta = \big(\theta_{td}\big)_{T \times D}$ — искомая матрица тем документов $\theta_{td} = p(t|d)$.

EM-алгоритм

E-шаг: если φ_{wt}, θ_{td} известны, то по формуле Байеса вычисляются условные вероятности тем $t \in T$ для всех (d, w):

$$H_{dwt} \equiv p(t|d,w) = \frac{p(w,t|d)}{p(w|d)} = \frac{\varphi_{wt}\theta_{td}}{\sum_s \varphi_{ws}\theta_{sd}}.$$

M-шаг: если H_{dwt} известны, то решение задачи максимизации правдоподобия аналитически выражается через H_{dwt}:
(по сути, это частотные оценки условных вероятностей)

$$\varphi_{wt} = \frac{\hat{n}_{wt}}{\hat{n}_t}, \qquad \hat{n}_{wt} = \sum_{d \in D} n_{dw} H_{dwt}, \qquad \hat{n}_t = \sum_{w \in W} \hat{n}_{wt};$$

$$\theta_{td} = \frac{\hat{n}_{dt}}{\hat{n}_d}, \qquad \hat{n}_{dt} = \sum_{w \in d} n_{dw} H_{dwt}, \qquad \hat{n}_d = \sum_{t \in T} \hat{n}_{dt}.$$

EM-алгоритм — это чередование E и M шагов до сходимости.

Рационализация EM-алгоритма: E-шаг встроен внутрь M-шага

Идея: не хранить H_{dwt}, а вычислять по мере необходимости.
Сложность алгоритма $O(|D| \cdot |W| \cdot |T|)$.

Вход: коллекция D, число тем $|T|$, начальные Φ и Θ;
Выход: распределения Φ и Θ;

повторять
  обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ для всех $d \in D$, $w \in W$, $t \in T$;
  для всех $d \in D$, $w \in d$
    $Z := \sum_t \varphi_{wt}\theta_{td}$;
    для всех $t \in T$ таких, что $\varphi_{wt}\theta_{td} > 0$
      увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на $n_{dw} \cdot \frac{1}{Z}\varphi_{wt}\theta_{td}$;
  $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $w \in W$, $t \in T$;
  $\theta_{td} := \hat{n}_{dt}/n_d$ для всех $d \in D$, $t \in T$;
пока Φ и Θ не стабилизируются.

Недостатки PLSA

- PLSA переобучается, т.к. параметров φ_{wt} и θ_{td} слишком много ($|D| \cdot |T| + |W| \cdot |T|$), и на них не накладывается никаких ограничений регуляризации.
- PLSA неверно оценивает вероятность новых слов:
  если $n_w = 0$, то $\hat{p}(w|t) = 0$ для всех $t \in T$.
- PLSA не позволяет управлять *разреженностью* Φ и Θ, т.к.
  (в начале $\varphi_{wt} = 0$) $\Leftrightarrow$ (в финале $\varphi_{wt} = 0$);
  (в начале $\theta_{td} = 0$) $\Leftrightarrow$ (в финале $\theta_{td} = 0$)

Гипотеза разреженности матриц Φ и Θ:
каждый документ относится к небольшому числу тем;
каждая тема описывается небольшим числом терминов...

т.е. на самом деле параметров должно быть намного меньше.

Латентное размещение Дирихле
LDA — Latent Dirichlet Allocation [David Blei, 2003]

Гипотеза об априорных распределениях:

- $\theta_d = (\theta_{td})_{t\in T} \in \mathbb{R}^{|T|}$ — случайные векторы из распределения Дирихле с параметром $\alpha \in \mathbb{R}^{|T|}$;
- $\varphi_t = (\varphi_{wt})_{w\in W} \in \mathbb{R}^{|W|}$ — случайные векторы из распределения Дирихле с параметром $\beta \in \mathbb{R}^{|W|}$

В результате недостатки PLSA устраняются благодаря сглаживанию частотных оценок условных вероятностей:

$$\varphi_{wt} := \frac{\hat{n}_{wt} + \beta_w}{\hat{n}_t + \sum_w \beta_w}; \qquad \theta_{td} := \frac{\hat{n}_{dt} + \alpha_t}{\hat{n}_d + \sum_t \alpha_t};$$

David Blei, Andrew Ng, Michael Jordan. Latent Dirichlet allocation // Journal of Machine Learning Research, 2003. — No. 3. — Pp. 993–1022.

Робастная вероятностная тематическая модель SWB — Special Words with Background [Steyvers et al. 2006]

Гипотеза: каждое употребление термина в документе объясняется либо темой, либо специфично для данного документа (шум), либо это общеупотребительный термин (фон).

Модель смеси тематической, шумовой и фоновой компонент:

$$p(w|d) = \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1 + \gamma + \varepsilon}; \qquad Z_{dw} = \sum_{t \in T} \varphi_{wt}\theta_{td},$$

$\pi_{dw} \equiv p_{\text{ш}}(w|d)$ — *шумовая* компонента, γ — параметр;
$\pi_w \equiv p_{\text{ф}}(w)$ — *фоновая* компонента, ε — параметр.

Требуется найти φ_{wt}, θ_{td}, π_{dw}, π_w для всех $d \in D$, $w \in W$, $t \in T$.

Chemudugunta C., Smyth P., Steyvers M. Modeling general and specific aspects of documents with a probabilistic topic model // Advances in Neural Information Processing Systems, MIT Press, 2006. — Vol. 19. — Pp. 241–248.

EM-алгоритм для робастной модели

E-шаг: вероятности тем, фона и шума для каждого (d, w):

$$H_{dwt} = \frac{\varphi_{wt}\theta_{td}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}, \quad t \in T;$$

$$H_{dw} = \frac{\gamma\pi_{dw}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}; \quad H'_{dw} = \frac{\varepsilon\pi_w}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}.$$

M-шаг— решение задачи максимизации правдоподобия:

$\varphi_{wt}, \theta_{td}$ — вычисляются по формулам PLSA;

$$\pi_w = \frac{\nu'_w}{\nu'}; \quad \nu'_w = \sum_{d\in D} n_{dw} H'_{dw}; \quad \nu' = \sum_{w\in W} \nu'_w;$$

$$\pi_{dw} = \left(\frac{n_{dw}}{\nu_d} - \frac{Z_{dw} + \varepsilon\pi_w}{\gamma}\right)_+.$$

Стандартная методика оценивания тематических моделей

Перплексия тестовой коллекции D' (hold-out perplexity):

$$\mathcal{P}(D') = \exp\left(-\frac{\sum\limits_{d\in D'}\sum\limits_{w\in d''} n_{dw}\ln p(w|d)}{\sum\limits_{d\in D'}\sum\limits_{w\in d''} n_{dw}}\right),$$

$d = d' \sqcup d''$ — случайное разбиение контрольного документа на две половины равной длины;

параметры φ_{wt} оцениваются по обучающей коллекции D;
параметры θ_{td} оцениваются по первой половине d';
перплексия вычисляется по второй половине d''.

Интерпретации перплексии:
1) $\mathcal{P}(D') \to |W|$ при $n \to \infty$, если слова равновероятны;
2) насколько хорошо мы можем предсказывать появление слов (чем меньше перплексия, тем лучше).

Другие методики оценивания тематических моделей

- Число ошибок классификации размеченной тестовой коллекции D'.
- Отклонение от *гипотезы условной независимости* $p(w|d,t) = p(w|t)$ на обучающей коллекции D для темы t:

$$\mathrm{KL}\Big(\hat{p}(d,w|t),\ \hat{p}(d|t)\cdot\hat{p}(w|t)\Big) = \sum_{d,w}\frac{n_{dwt}}{n_t}\log\frac{n_{dwt}\cdot n_t}{n_{td}\cdot n_{wt}}$$

D.Mimno, D.Blei. Bayesian checking for topic models // Empirical Methods in Natural Language Processing, 2011.

- Доля случаев, когда эксперт верно определяет:
  — лишнюю тему в списке главных тем документа;
  — лишний термин в списке главных терминов темы.

J.Chang, J.Boyd-Graber, S.Gerrish, C.Wang, D.Blei. Reading tea leaves: how humans interpret topic models // Advances in Neural Information Processing Systems 22, 2009, pp. 288–296.

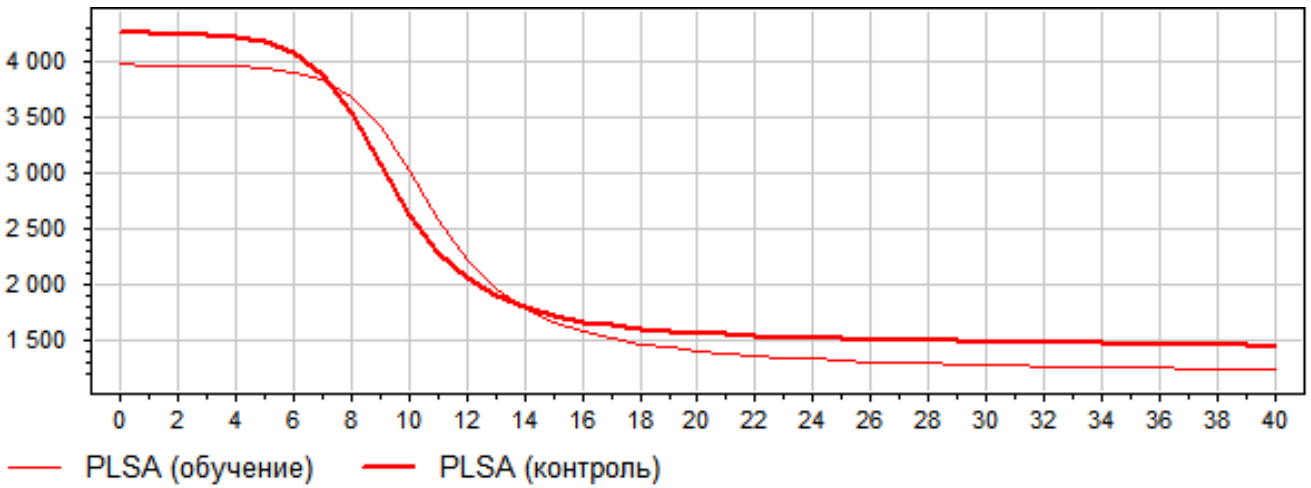
Методика эксперимента

D — коллекция 2000 авторефератов диссертаций на русском языке суммарной длины $n \approx 8.7 \cdot 10^6$, словарь $|W| \approx 3 \cdot 10^4$.

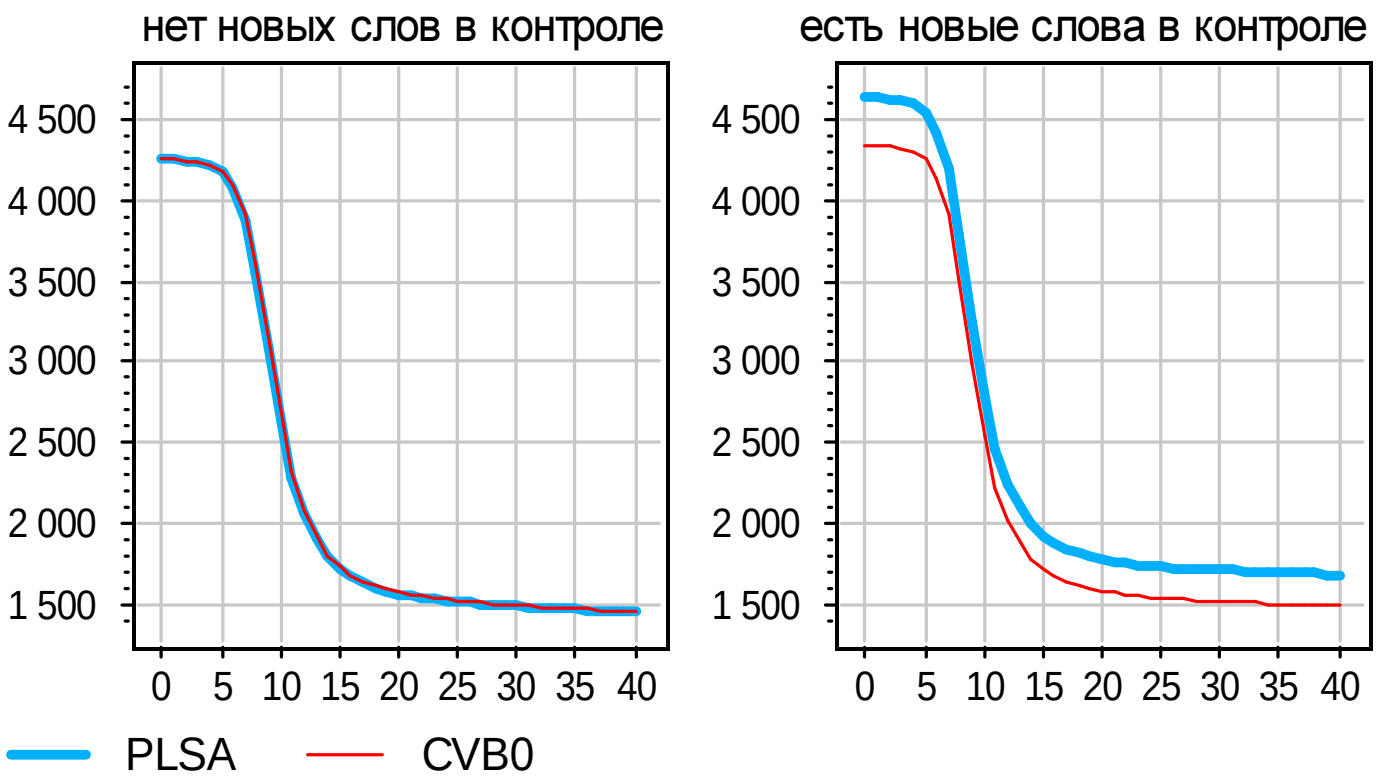
Предобработка: лемматизация, удаление стоп-слов.

D' — коллекция 200 авторефератов, не включённых в D.

Строятся графики зависимости перплексии от числа итераций (проходов коллекции); число итераций 40; число тем $|T| = 100$;



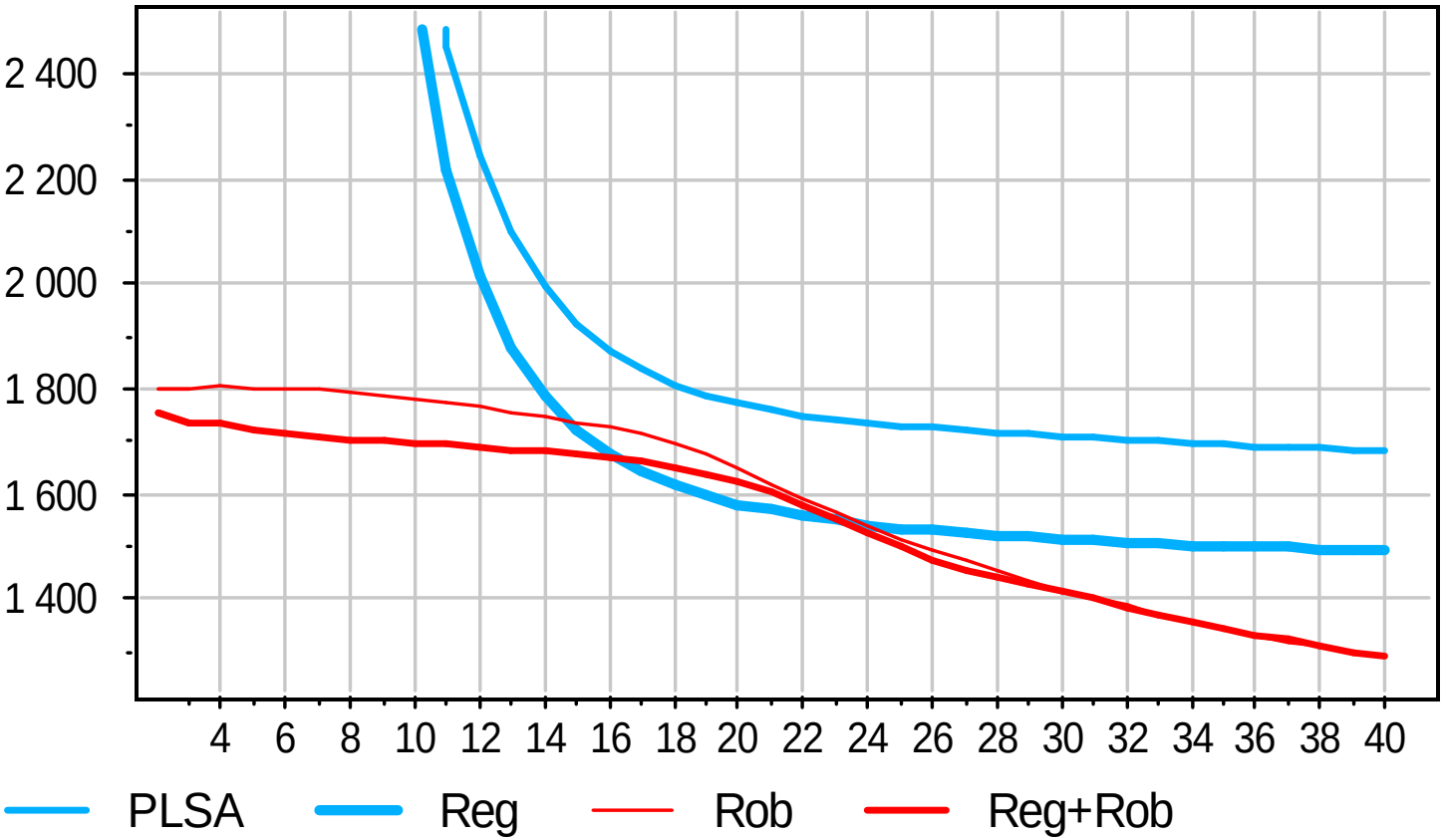
Регуляризация решает проблему новых слов, а не переобучения



PLSA без регуляризации, CVB0 с регуляризацией.

Вывод: регуляризация даёт преимущество только когда в контроле есть новые термины.

Робастная модель не нуждается в регуляризации



Робастность сильнее уменьшает перплексию PLSA, чем регуляризация. Регуляризация не улучает робастную модель.

Вопросы. . .

Воронцов Константин Вячеславович
vokov@forecsys.ru

Страницы на www.MachineLearning.ru:

- Участник:Vokov
- Машинное обучение (курс лекций, К.В.Воронцов)
- Тематическое моделирование

Воронцов К. В., Потапенко А. А. Робастные разреженные вероятностные тематические модели // Интеллектуализация обработки информации (ИОИ-2012): Докл. — Москва: Торус Пресс, 2012. С. 605–608.