Обучение с учителем. Классификация. Дискриминантный анализ. Логистическая регрессия. Метод опорных векторов. Выбор модели с помощью кросс-валидации. Метод стохастического градиента.

Гриненко Ю., Хасанова К.

Санкт-Петербургский государственный университет
Кафедра статистического моделирования

2023

Классификация. Задача.

Дано:

- Множество объектов $X \in \mathbb{R}^p$;
- Множество ответов Y, $Y_i \in \mathcal{G}$.

Задача

- По выборке $X^N = (\mathbf{x}_i, y_i)_{i=1}^N$ построить классификатор $a : \mathbb{R}^p \to \mathcal{G}$, который по новому вектору признаков X предскажет отметку класса, т.е. $a(\mathbf{x}) \in \mathcal{G}$;
- На $a(\mathbf{x})$ должна достигаться минимальная ошибка классификации в некотором смысле;
- Хотим знать вероятности принадлежности объекта к тому или иному классу.

Классификация. Задача.

На генеральном языке:

- $\boldsymbol{\xi} \in \mathbb{R}^p$ – случайный вектор признаков;
- $\eta \in \mathcal{G} = \{G_k\}_{k=1}^K$ – дискретная случайная величина, класс;
- $P(\boldsymbol{\xi}, \eta)$ – их совместное распределение.

Дано: выборка $(\mathbf{x}_i, y_i)_{i=1}^N$ – N реализаций случайных векторов $(\boldsymbol{\xi}, \eta)$.

Строим классификатор

$$f : \mathbb{R}^p \to \mathcal{G}. \tag{1}$$

Запишем в форме

$$f(x_i, w) = sign(\langle x_i, w \rangle - w_o) = sign \sum_{i=1}^{n} w_i x_i - w_o. \tag{2}$$

Ключевые понятия

Вводим несколько ключевых понятий:

- Величина отступа (margin) — $M_i(w, w_0) = (\langle x_i, w \rangle - w_0) y_i$; чем меньше значение отступа, тем ближе объект к границе классов, соответственно;
- Функция потерь $\mathcal{L}(a, x)$ — неотрицательная функция, характеризующая величину ошибки предсказания на объекте x_i. Задачу классификации можно свести к минимизации функции потерь;
- Функционал качества алгоритма $Q(a, X^n) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(a, x_i)$ на выборке X^n, также называемый эмпирическим риском.

Margin-based loss function

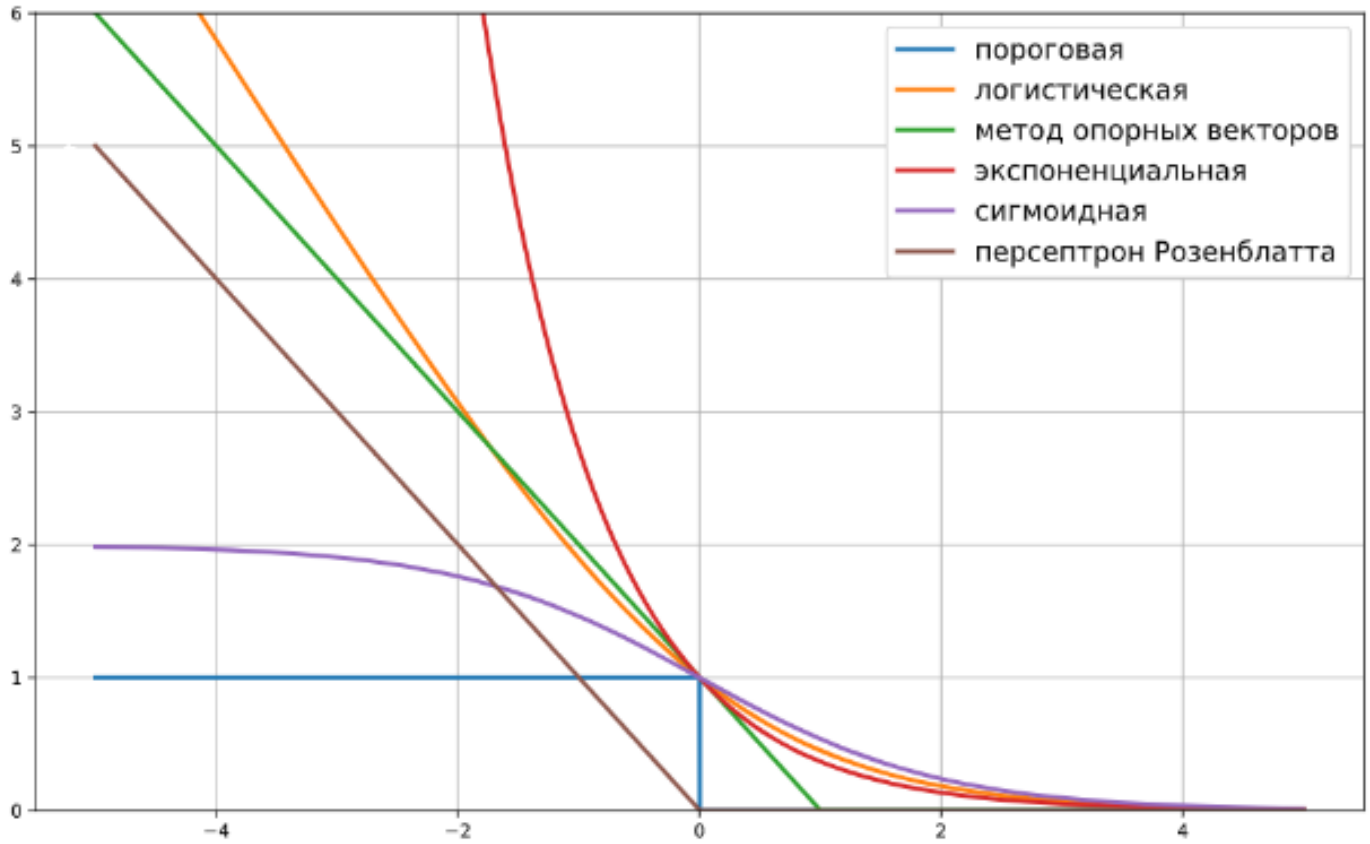


Рис.: Непрерывные аппроксимации пороговой функции потерь

Дискриминантный анализ

Строим классифицирующие функции f_i, индивид относится к классу с максимальным значением классифицирующей функции на нем.

$$f(x) = \arg\max_{Y \in \mathcal{G}} P(Y|\xi = x). \tag{3}$$

В качестве классифицирующих функций берем

$$f_i(x) = \frac{p_i(x)\pi_i}{\sum_{j=1}^{K} p_j(x)\pi_j}. \tag{4}$$

Априорные вероятности выбираем в зависимости от наших предположений/задачи.

LDA

В LDA предполагаем нормальное распределении классов с одинаковой ковариационной матрицей. Классифицирующие функции имеют вид

$$f_i(x) = \frac{\pi_i}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^{\mathsf{T}}\Sigma^{-1}(x-\mu_i)\right), \tag{5}$$

и приводятся к виду

$$h_i(x) = -\frac{1}{2}\mu_i^{\mathsf{T}}\Sigma^{-1}\mu_i + \mu_i^{\mathsf{T}}\Sigma^{-1}x + log\ \pi_i. \tag{6}$$

QDA

Применяем QDA, когда каждый класс имеет многомерное нормальное распрелеление и различные ковариационные матрицы $\Sigma_{i=1}^{K}$. Классифицирующие функции принимают вид

$$g_i(x) = log\ f_i(x) = log\ \pi_i - \frac{1}{2} log|\Sigma_i| - \frac{1}{2}(x^{\mathsf{T}} - \mu_i)\Sigma_i^{-1}(x - \mu_i). \quad (7)$$

Метод максимального правдоподобия

Параметры распределений классов нам не известны, используем оценки ОМП.

- Среднее $\widehat{\mu}_i = \frac{1}{n_i} \sum\limits_{j:y_j=G_i} \mathbf{x}_j$,
- Ковариационная матрица класса $\hat{\Sigma}_i = \frac{1}{n_i-1} \sum\limits_{j:y_j=G_i} (\mathbf{x}_j - \widehat{\mu}_i)^{\mathrm{T}}(\mathbf{x}_j - \widehat{\mu}_i)$,
- Pooled ковариационная матрица $\hat{\Sigma} = \sum\limits_{j=1}^{K} \frac{n_i-1}{n-K}\hat{\Sigma}_i$.

RDA

RDA представляет собой компромисс между LDA и QDA. Стягиваем ковариации каждого класса к общей матрице

$$\hat{\Sigma}_i(\alpha) = \alpha\hat{\Sigma}_i + (1-\alpha)\hat{\Sigma},$$

$\alpha \in [0;1]$ — настраиваемый параметр, определяющий, оценивать ли ковариации отдельно $\alpha = 1$ или совместно $\alpha = 0$ (pooled).

Канонические переменные

Задача состоит в нахождении линейного преобразования $\mathbf{Z} = A^T\mathbf{X}$.
В англоязычной литературе имеет название CDA (Canonical Discriminant Analysis)
Новые признаки $\mathbf{Z} = A^{\mathrm{T}}\mathbf{X}$. Выборочная ковариационная матрица новых признаков:

$$A^{\mathrm{T}}\mathbf{T}A = A^{\mathrm{T}}(\mathbf{E} + \mathbf{H})A = A^{\mathrm{T}}\mathbf{E}A + A^{\mathrm{T}}\mathbf{H}A, \tag{8}$$

$\mathbf{E}/\mathbf{H}$ - оценка внутригрупповых/межгрупповых отклонений соответственно, $\mathbf{T}$ — total covariance matrix.

Канонические переменные

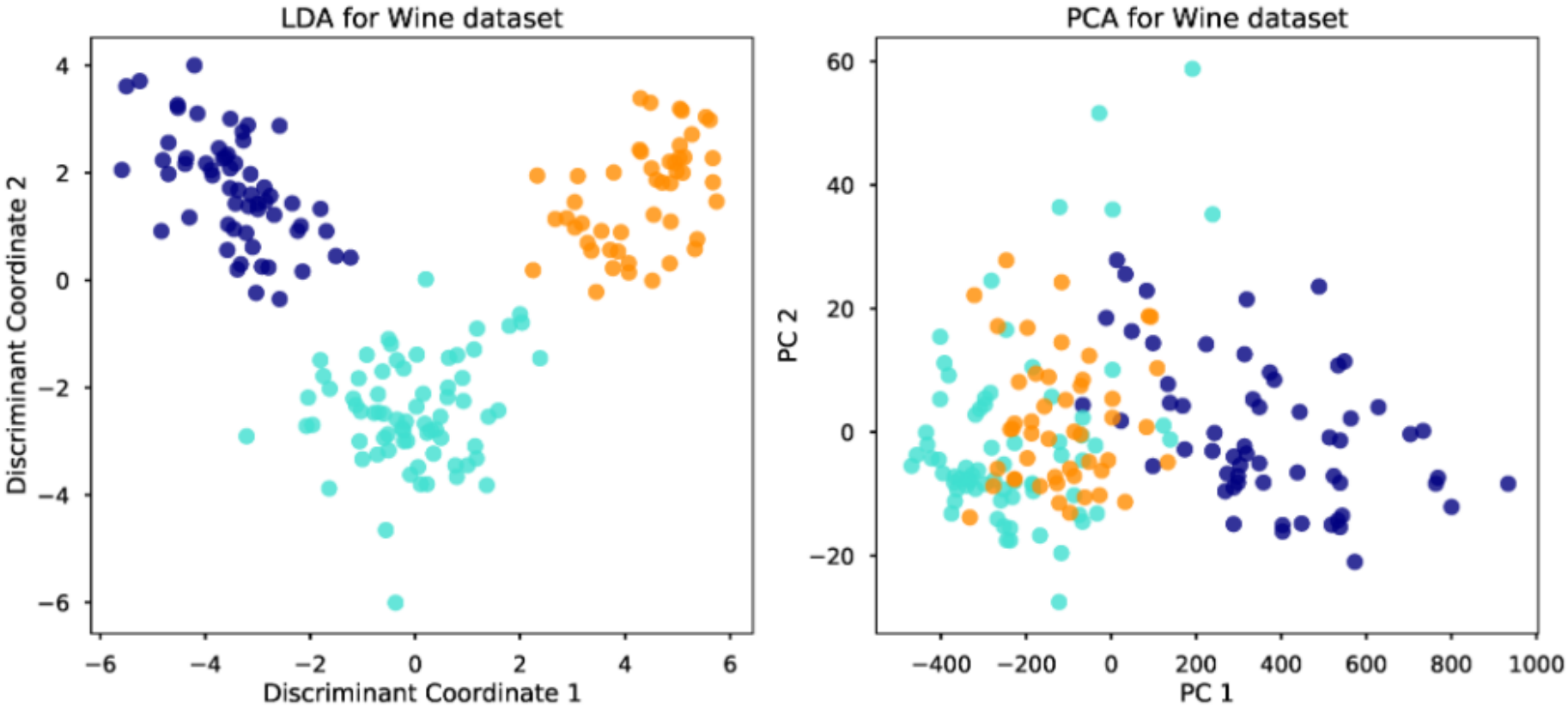


Рис.: Canonical Discriminant Analysis and PCA on data

Логистическая регрессия

Линейный классификатор. Минимизации эмпирического риска:

$$Q(w) = \sum_{i=1}^{n} ln(1 + exp(-y_i \langle x_i, w \rangle)) \to \min_w. \tag{9}$$

Когда решение w найдено, можем оценивать апостериорные вероятности принадлежности классам:

$$p = \sigma(\langle w, X_i \rangle),$$

где σ – сигмоидная функция:

$$\sigma(M) = \frac{1}{1 + e^{-M}}.$$

Логистическая регрессия

Отступ M отрицательный, когда алгоритм $a(x, w)$ допускает ошибку на объекте x_i. Число ошибок классификации через отступы:

$$Q(w) = \sum_{i=1}^{n} [M(x_i) < 0].$$

Идея в замене пороговой функции потерь непрерывной оценкой сверху:

$$[M(x_i) < 0] \leq log_2(1 + e^{-M}).$$

Так получаем функционал (9) с предыдущего слайда.

Логистическая регрессия

Используем logit преобразование, логарифм отношения вероятности положительного события к отрицательному $\log(\frac{p}{1-\mathrm{p}})$.

$$\langle w, x_i \rangle = \log(\frac{p}{1-p}); e^{\langle w, x_i \rangle} = \frac{p}{1-p},$$

$$\mathrm{p} = \frac{1}{1 + e^{-\langle w, x_i \}}}.$$

Справа сигмоида σ. Свойства:

- Монотонно возрастает;
- отображает всю числовую прямую на интервал (0; 1);
- $\sigma(-x) = 1 - \sigma(x)$.

Апостериорные вероятности необходимы для оценивания рисков, связанных с возможными ошибками классификации.

Логистическая регрессия

Как оптимизировать w так, чтобы модель как можно лучше предсказывала логиты? ММП для распределения Бернулли.

$$p(y|X,\ w) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}.$$

Приводим к логарифмическому виду:

$$\ell(w, X, y) = \sum_{i=1}^{n}(y_i \log(p_i) + (1-y_i)\log(1-p_i)) =$$

$$= \sum_{i=1}^{n}(y_i \log(\sigma(\langle w, x_i\rangle)) + (1-y_i)\log(1-\sigma(\langle w,\ x_i\rangle))).$$

Логистическая регрессия

$$\ell(w, X, y) = \sum_{i=1}^{n} (y_i \log(\sigma(\langle w, x_i \rangle)) + (1 - y_i) \log(\sigma(-\langle w, \ x_i \rangle))).$$

Максимизация правдоподобия эквивалентна минимизации функционала, гладко аппроксимирующего эмпирический риск. Чтобы получить функцию потерь, которую мы будем минимизировать, умножаем на -1 и получаем

$$\mathcal{L}(w, X, y) = -\sum_{i=1}^{n} (y_i \log(\sigma(\langle w, x_i \rangle)) + (1 - y_i) \log(\sigma(-\langle w, x_i \rangle))).$$

Методы решения задачи минимизации — метод стохастического градиента, метод Ньютона-Рафсона.

Метод опорных векторов

Линейный классификатор, задача классификации в рамках обучения с учителем. Имеется выборка $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, $x_i \in \mathbb{R}^p$, $y_i \in \{-1, 1\}$; задачей является построение классифицирующего правила $f : \mathbb{R}^p \to \{-1, 1\}$. Не накладывается ограничений на распределение.

Hard-margin SVM

Линейный классификатор:

$$f(x_i) = sign(\langle x_i, w\rangle - w_0), \tag{10}$$

Уравнение, описывающее гиперплоскость, разделяющую классы в пространстве $\mathbb{R}$:

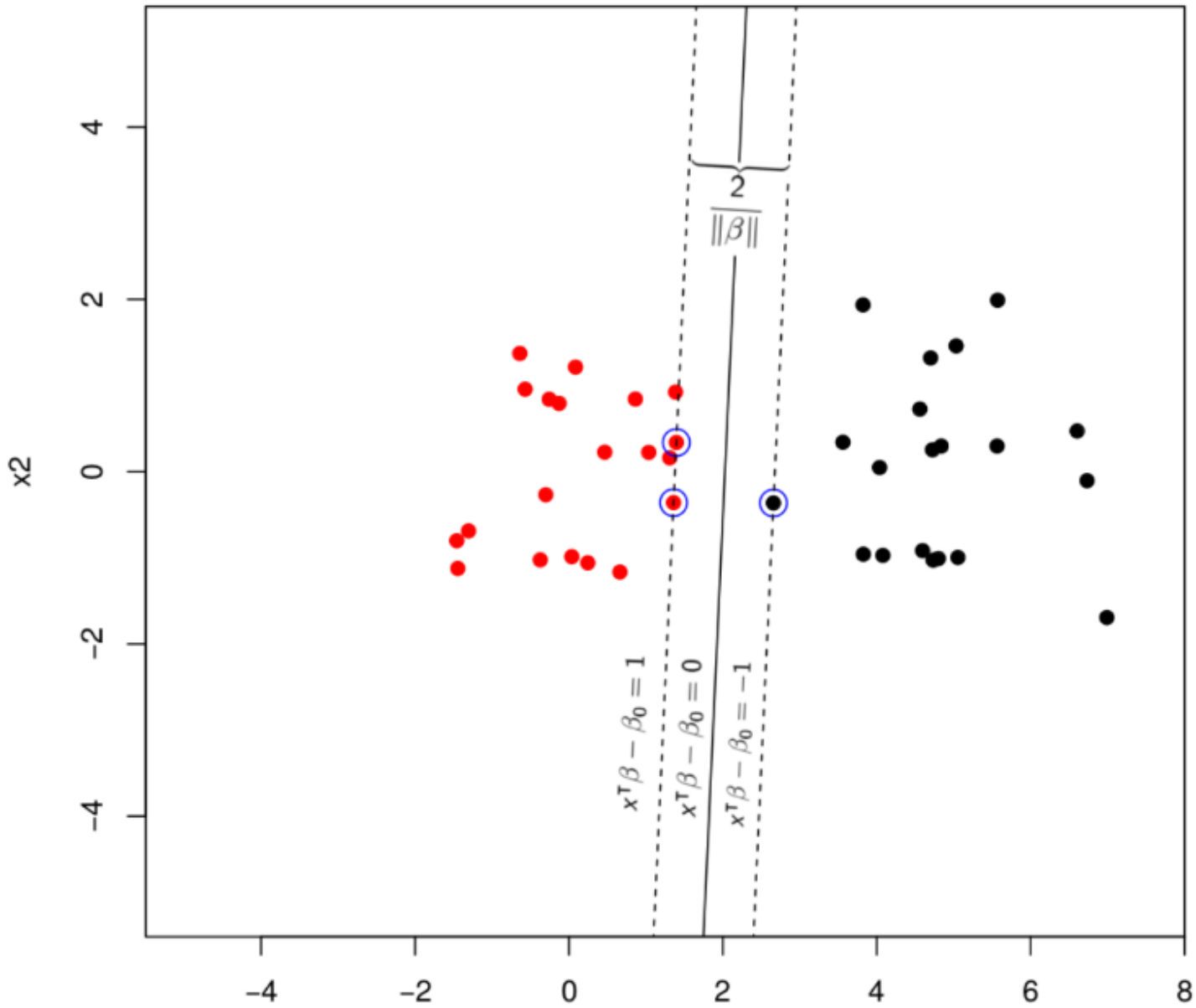
$$\langle w, x\rangle = w_0.$$

Предполагаем линейная разделимость классов (есть w, w_0, при которых функционал принимает нулевое значение):

$$Q(w,\ w_0) = \sum_{i=1}^{n} [y_i(\langle w,\ x_i\rangle - w_0) < 0] \tag{11}$$

Критерий оптимальности: максимальное расстояние между двумя гиперплоскостями, параллельных данной, без точек между ними. Точки, которые принадлежат одной из гиперплоскостей — опорные вектора.

Hard-margin SVM



Hard-margin SVM

Параметры линейного порогового классификатора определены с точностью до нормировки. Коэффициенты w, w_0 можем умножить на константу так, что

$$\langle w,\ x_i\rangle - w_0 = y_i.$$

$$\langle w, x_i\rangle - w_0 = \begin{cases} \leq -1, if\ y_i = -1; \\ \geq +1, if\ y_i = +1. \end{cases}$$

Условие $-1 < \langle w, x\rangle - w_0 < 1$ задаёт полосу, разделяющую классы. Ширина полосы

$$\langle (x_+ - x_-),\ \frac{w}{\|w\|}\rangle = \frac{\langle w, x_+\rangle - \langle w, x_-\rangle}{\|w\|} = \frac{(w_0+1)-(w_0-1)}{\|w\|} = \frac{2}{\|w\|}.$$

Ширина полосы максимальна, когда норма вектора w минимальна.

Обобщение задачи. Soft-Margin

Построение оптимальной разделяющей гиперплоскости сводится к минимизации квадратичной формы

$$\begin{cases} \frac{1}{2}\|w\|^2 \to \min\limits_{\mathbf{w}}; \\ (\langle x_i, w\rangle - w_0)\, y_i \geq 1. \end{cases}$$

Обобщение на случай линейно неразделимых выборок. Позволим алгоритму допускать ошибки на обучающих объектах.
Введём набор дополнительных переменных $\xi > 0$, характеризующих величину ошибки на объектах x_i, $i = 1, \ldots, n$ (C задает размер штрафа за ошибки.).

$$\begin{cases} \frac{1}{2}\langle w,\ w\rangle + C\sum_{i=1}^{n}\xi_i \to \min\limits_{w, w_0, \xi}; \\ y_i(\langle w,\ x_i\rangle - w_0) \geq 1 - \xi_i,\ \forall i; \\ \xi_i \geq 0,\ \forall i. \end{cases}$$

Обобщение задачи. Soft-Margin

Отступ объекта x_i от границы классов:

$$[M_i < 0] \leq (1 - M_i)_+.$$

Ошибка ξ_i выражается через отступ M_i. Т.к. минимизации суммы $\sum_{i=1}^{n} \xi_i$, то можем записать$\xi_i = (1 - M_i)_+$. Задача эквивалентной минимизации функционала Q, не зависящего от ξ_i:

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

Верхняя оценка эмпирического риска, к которому добавлен регуляризатор $\|w\|^2$, параметр регуляризаци $\frac{1}{2C}$.
($[M_i < 0] \leq (1 - M_i)_+$)

Обобщение задачи. Soft-Margin

Замена пороговой функции потерь $[M < 0]$ кусочно-линейной верхней оценкой $\mathcal{L}(M) = (1 - M)_+$ делает функцию потерь чувствительной к величине ошибки. Функция потерь $\mathcal{L}(M)$ штрафует объекты за приближение к границе классов.
Двойственная задача. Запишем функцию Лагранжа:

$$\mathcal{L}(w, w_0, \xi; \lambda, \eta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi - \sum_{i=1}^{n}\lambda_i(M_i(w, w_0) - 1 + \xi_i) - \sum_{i=1}^{n}\xi_i\eta_i =$$

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

где $\lambda = (\lambda_1, \ldots, \lambda_n)$ — вектор переменных, двойственных к w;
$\eta = (\eta_1, \ldots, \eta_n)$ — вектор переменных, двойственных к ξ.

Обобщение задачи. Soft-Margin

Согласно теореме Куна-Таккера, задача эквивалентна двойственной задаче поиска седловой точки функции Лагранжа:

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

Необходимым условием седловой точки функции Лагранжа является равенство нулю её производных. Отсюда получаются соотношения:

$$\begin{cases} \frac{\partial}{\partial w} : & w = \sum\limits_{i=1}^{n} \lambda_i y_i x_i \\ \frac{\partial}{\partial w_0} : & 0 = \sum\limits_{i=1}^{n} \lambda_i y_i \\ \frac{\partial}{\partial \xi_i} : & \lambda_i = C - \eta_i \end{cases}$$

Обобщение задачи. Soft-Margin

Согласно теореме Куна-Таккера, задача эквивалентна двойственной задаче поиска седловой точки функции Лагранжа:

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

Необходимым условием седловой точки функции Лагранжа является равенство нулю её производных. Отсюда получаются соотношения:

$$\begin{cases} \frac{\partial}{\partial w} : & w = \sum\limits_{i=1}^{n} \lambda_i y_i x_i \\ \frac{\partial}{\partial w_0} : & 0 = \sum\limits_{i=1}^{n} \lambda_i y_i \\ \frac{\partial}{\partial \xi_i} : & \lambda_i = C - \eta_i \end{cases}$$

Обобщение задачи. Soft-Margin

$$\left\{ \frac{\partial}{\partial w} : \quad w = \sum_{i=1}^{n} \lambda_i y_i x_i \right.$$

Искомый вектор весов w является линейной комбинацией векторов обучающей выборки x_i, причём только тех, для которых $\lambda_i > 0$.
Используя эти равенства, получаем

$$\begin{cases} -\mathcal{L}(\lambda) = -\sum\limits_{i=1}^{n} \lambda_i + \frac{1}{2} \sum\limits_{i=1}^{n} \sum\limits_{j=1}^{n} \lambda_i \lambda_j y_i y_k \langle x_i, x_j \rangle \to \min\limits_{\lambda} \\ 0 \leq \lambda_i \leq C, \forall i \\ \sum\limits_{i=1}^{n} \lambda_i y_i = 0 \end{cases}$$

Данная двойственная задача имеет единственное решение.
Константу C обычно выбирают по критерию скользящего контроля.

Обобщение задачи. Soft-Margin

Для определения порога w_0 достаточно взять произвольный опорный граничный вектор x_i и выразить w_0 из равенства $w_0 = \langle w, x_i \rangle - y_i$.
Алгоритм классификации представляется в следующем виде:

$$a(x) = \text{sign}\ (\sum_{i=1}^{n} \lambda_i y_i \langle x_i,\ x \rangle - w_0)$$

Суммирование только по опорным векторам, для которых $\lambda_i \neq 0$.
Но ненулевыми λ_I обладают не только опорные объекты, но и объекты-нарушители. Это недостаток SVM, т.к. ими часто оказываются шумовые выбросы, и построенное на них решающее правило опирается на шум.

Спрямляющее пространство

По построению, SVM применим лишь к (почти) линейно-разделимым данным.
Предположим, что известен набор отображений

$$\{\varphi_j(x)\}_{j=1}^{d}$$

$$\varphi_j : \mathbb{R}^p \to \mathbb{R}$$

таких, что набор данных

$$\left\{ \tilde{x}_i = \begin{pmatrix} \varphi_1(x_i) \\ \vdots \\ \varphi_d(x_i) \end{pmatrix}, y_i \right\}_{i=1}^{n}$$

линейно разделим; будем называть пространство, в котором наблюдается разделимость, спрямляющим пространством.

Двойственная задача в спрямляющем пространстве

Записывая двойственную задачу в спрямляющем пространстве,

$$\sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle \to \max_{\lambda_i}$$

можно заметить, что x_i входят в её формулировку лишь в виде скалярных произведений $\tilde{x}_i^{\mathsf{T}} \tilde{x}_j$.
Также, используя выражение

$$w = \sum_{i=1}^{n} \lambda_i y_i x_i$$

вычисление расстояния до разделяющей гиперплоскости (со знаком) также сводится к использованию скалярных произведений x_i.

The Kernel Trick

Скалярного произведения в спрямляющем пространстве достаточно для построения классифицирующего правила. Знание отображений φ_j не требуется и спрямляющее пространство может быть бесконечномерным.

Пусть $K(u, v) : X \times X \to \mathbb{R}$ – отображение:

- Симметричное: $K(u, v) = K(v, u)$,
- Положительно определённое:
$$\iint\limits_{X \times X} K(u, v)\, g(u)\, g(v)\, \mathrm{d}u\, \mathrm{d}v \geq 0, \forall g : X \to \mathbb{R}.$$

Тогда (и только тогда) существует пространство H и отображение $\varphi : X \to H : K(u, v) = \langle \varphi(u), \varphi(v) \rangle_H$.

После перехода в новое пространство (с помощью φ) исходные данные могут стать почти линейно-разделимыми.

The Kernel Trick

Способы построения ядер:

- Произвольное скалярное произведение $K(u,\ v) = \langle u, v \rangle$;
- Константа $K(u,\ v) = 1$ является ядром;
- Произведение ядер $K(u,\ v) = K_1(u,\ v)K_2(u,\ v)$ является ядром;
- Для любой функции $\psi : X \to \mathbb{R}$ произведение $K(u,\ v) = \psi(u)\psi(v)$ является ядром.
- Линейная комбинация ядер с неотрицательными коэффициентами $K(u,\ v) = \alpha_1 K_1(u,\ v) + \alpha_2 K_2(u,\ v)$ является ядром;
- Композиция произвольной функции $\varphi : X \to X$ и произвольного ядра K_0 является ядром: $K(u,\ v) = K_0(\varphi(u),\ \varphi(v))$.

Наиболее часто используемые ядра: линейное, полиномиальное, радиальное, сигмовидное.

The Kernel Trick

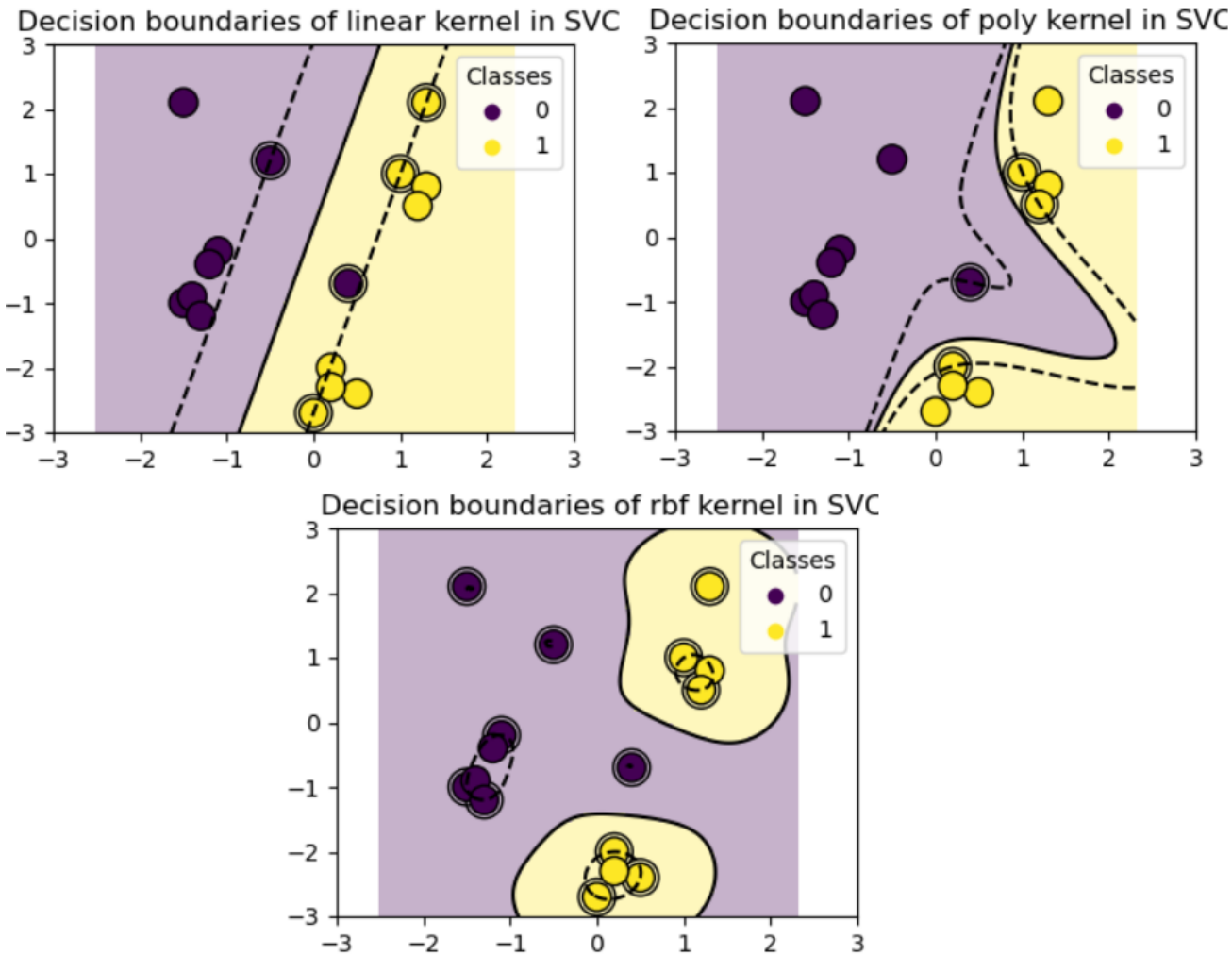


Рис.: Plot classification boundaries with different SVM Kernels

The Kernel Trick. Пример

Возьмём $X = \mathbb{R}^2$ и рассмотрим ядро $K(u,\ v) = \langle u, v\rangle^2$, где $u = (u_1,\ u_2)$, $v = (v_1,\ v_2)$. Какое спрямляющее пространство и преобразование ему соответствует? Разложим

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

Ядро K представляется в виде скалярного произведения в пространстве $H = \mathbb{R}^3$. Преобразование $\psi : \mathbb{R}^2 \to \mathbb{R}^3$ имеет вид $\psi : (u_1,\ u_2) \mapsto (u_1^2,\ u_2^2,\ \sqrt{2}u_1u_2)$. Линейной поверхности в пространстве H соответствует квадратичная поверхность в исходном пространстве X. Данное ядро позволяет разделить внутреннюю и наружную часть произвольного эллипса, что невозможно в исходном двумерном пространстве.

Метод стохастического градиента

Стохастический градиентный спуск - оптимизационный алгоритм, при котором градиент оптимизируемой функции считается на каждой итерации от случайно выбранного элемента выборки. Модификацией градиентного спуска, и в случае линейного классификатора искомый алгоритм имеет вид:

$$a(x, \omega) = \phi(\sum_{j=1}^{N} \omega_j x^j - \omega_0), \tag{12}$$

где $\phi(z)$ - функция активации.
Решаемая задача:

$$Q(\omega) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(a(x_i, \omega), y_i) \to \min_{\omega}.$$

Метод стохастического градиента

Обновление весов:

$$\omega^{(t+1)} = \omega^{(t)} - h\nabla Q(\omega^{(t)}). \tag{13}$$

Проблема GD — чтобы определить новое приближение вектора весов необходимо вычислить градиент от каждого элемента выборки, что может сильно замедлять алгоритм. Идея ускорения заключается в использовании либо одного элемента (SGD), либо некоторой подвыборки (Batch GD) для получения нового приближения весов. При подходе SGD веса обновляются:

$$\omega^{(t+1)} = \omega^{(t)} - h\nabla \mathcal{L}(a(x_i, \omega^{(t)}), y_i),$$

где i — случайно выбранный индекс.

Метод стохастического градиента. Начальные веса

Новый стохастический градиент должен быть несмещённой оценкой полного градиента (сходимость SGD обеспечивается несмещенностью). Т.к. направление изменения ω будет определяться за $O(1)$, подсчет Q на каждом шаге будет слишком дорогостоящим. Для ускорения оценки используются рекуррентные формулы:

- Среднее арифметическое:
  $\overline{Q}_m = \frac{1}{m}\mathcal{L}(a(x_{i_m}, \omega^{(m)}), y_{i_m}) + (1 - \frac{1}{m})\overline{Q}_{m-1}$;
- Экспоненциальное скользящее среднее:
  $\overline{Q}_m = \lambda\mathcal{L}(a(x_{i_m}, \omega^{(m)}), y_{i_m}) + (1 - \lambda)\overline{Q}_{m-1}$, где λ — темп забывания "предыстории" ряда.

SGD

Достоинства: метод легко реализуется, функция потерь и семейство алгоритмов могут быть любыми, подходит для задач с большими данными, иногда можно получить решение даже не обработав всю выборку;
Недостатки: проблемы с локальными экстремумами (для борьбы с этим используют технику "встряхивания коэффициентов"(jog of weights)); направление обновляется на основании получаемых в данный момент данных; алгоритм может не сходиться или сходиться слишком медленно.

Критерии выбора модели. Скользящий контроль

Критерий выбора — функционал $Q(\mu, X^n)$, характеризующий качество метода μ. Так, метод минимизации эмпирического риска строит алгоритм с минимальным значением критерия

$$\mu(X^n) = \arg\min_{a \in A} Q(a, X^n). \tag{14}$$

Скользящий контроль (cross-validation, CV) — процедура эмпирического оценивания, с помощью которой "эмулируется" наличие тестовой выборки, которая не участвует в обучении, но для которой имеются метки класса. Разбиение выборки: $X^L = X^n \cup X^k$.

$$CV(\mu, X^n) = \frac{1}{N} \sum_{n=1}^{N} Q(\mu(X_n^n), X_n^k). \tag{15}$$

Критерии выбора модели. CV (LOO)

Существует несколько вариантов скользящего контроля (Complete CV, leave-one-out CV, q-fold CV). Complete CV строится по всем $N = C_L^K$, вычислительно слишком сложен и применяется в теоретических исследования или если можно вывести удобную вычислительную формулу. На практике интересны другие 2 варианта. Leave-one-out CV:

$$LOO(\mu, X^L) = \frac{1}{L}\sum_{i=1}^{L} Q(\mu(X^L \backslash \{x_i\}), \{x_i\}). \tag{16}$$

Каждый объект участвует в контроле один раз, длина обучающих подвыборок на единицу меньше длины полной выборки. Обучать приходится L раз, что тоже трудоемко.

Критерии выбора модели. CV (q-fold)

Q-fold CV — случайное разбиение выборки на q непересекающихся блоков одинаковой длины $l_1, ..., l_n; X^L = X_1^{l_1} \cup \cdots \cup X_q^{l_q}$, $l_1 + \cdots + l_q = L$. Блок по очереди становится контрольной подвыборкой, обучение по остальным q-1 блокам. Критерий определяется как средняя по всем блокам ошибка.

$$Q_{ext}(\mu, X^L) = \frac{1}{q} \sum_{n=1}^{q} Q(\mu(X^L \backslash X_n^{l_n}), X_n^{l_n}). \tag{17}$$

Критерии выбора модели

Для выбора оптимального метода рекомендуется использовать несколько разные критериев. Например, отбираем несколько лучших методов по критерию, затем из них выбрать тот, для которого другой критерий принимает наименьшее значение.