

- Математические модели машинного обучения •

Метод опорных векторов

Воронцов Константин Вячеславович

`k.v.vorontsov@phystech.edu`

`http://www.MachineLearning.ru/wiki?title=User:Vokov`

Этот курс доступен на странице вики-ресурса

`http://www.MachineLearning.ru/wiki`

«Машинное обучение (курс лекций, К.В.Воронцов)»

МФТИ • 5 октября 2026

1 Метод опорных векторов SVM

- Принцип оптимальной разделяющей гиперплоскости
- Двойственная задача
- Понятие опорного вектора

2 Обобщения линейного SVM

- Ядра и спрямляющие пространства
- Вероятностная калибровка
- SVM-регрессия

3 Регуляризация

- Регуляризаторы для отбора признаков
- Методы SFM и RFM
- Метод релевантных векторов RVM

Задача обучения линейного бинарного классификатора

Дано:

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

Найти:

Параметры $w \in \mathbb{R}^n$, $w_0 \in \mathbb{R}$ линейной модели классификации

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

Критерий — минимум эмпирического риска:

$$\sum_{i=1}^{\ell} [a(x_i; w, w_0) \neq y_i] = \sum_{i=1}^{\ell} [M_i(w, w_0) < 0] \rightarrow \min_{w, w_0},$$

где $M_i(w, w_0) = (\langle x_i, w \rangle - w_0)y_i$ — отступ (margin) объекта x_i

В. Н. Вапник, А. Я. Лернер. Узнавание образов при помощи обобщенных портретов. 1963.

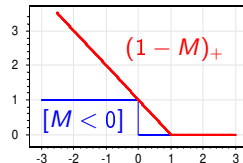
C. Cortes, V. Vapnik. Support vector networks. 1995.

Аппроксимация и регуляризация эмпирического риска

Эмпирический риск — это кусочно-постоянная функция
Заменим его оценкой сверху, непрерывной по параметрам:

$$Q(w, w_0) = \sum_{i=1}^{\ell} [M_i(w, w_0) < 0] \leq \sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

- Аппроксимация штрафует объекты за приближение к границе классов, увеличивая зазор между классами
- Регуляризация штрафует неустойчивые решения в случае мультиколлинеарности



В. Н. Вапник, А. Я. Лернер. Узнавание образов при помощи обобщенных портретов. 1963.

C. Cortes, V. Vapnik. Support vector networks. 1995.

Оптимальная разделяющая гиперплоскость

Линейный классификатор: $a(x, w) = \text{sign}(\langle w, x \rangle - w_0)$

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

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

Нормировка: $\min_{i=1, \dots, \ell} M_i(w, w_0) = 1$

Разделяющая полоса (разделяющая гиперплоскость посередине):

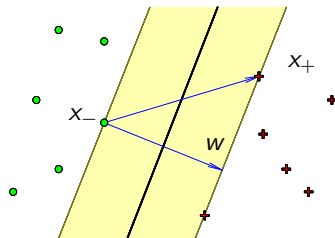
$$\{x : -1 \leq \langle w, x \rangle - w_0 \leq 1\}$$

$$\exists x_+ : \langle w, x_+ \rangle - w_0 = +1$$

$$\exists x_- : \langle w, x_- \rangle - w_0 = -1$$

Ширина полосы:

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



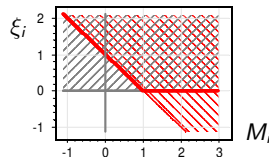
Обоснование кусочно-линейной функции потерь

Линейно разделимая выборка

$$\begin{cases} \frac{1}{2} \|w\|^2 \rightarrow \min_{w, w_0} \\ M_i(w, w_0) \geq 1, \quad i = 1, \dots, \ell \end{cases}$$

Переход к линейно неразделимой выборке (эвристика)

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{\ell} \xi_i \rightarrow \min_{w, w_0, \xi} \\ \xi_i \geq 1 - M_i(w, w_0), \quad i = 1, \dots, \ell \\ \xi_i \geq 0, \quad i = 1, \dots, \ell \end{cases}$$



Эквивалентная задача безусловной минимизации:

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

Напоминание. Условия Каруша–Куна–Таккера (ККТ)

Задача математического программирования:

$$\begin{cases} f(x) \rightarrow \min_x \\ g_i(x) \leq 0, & i = 1, \dots, m \\ h_j(x) = 0, & j = 1, \dots, k \end{cases}$$

Необходимые условия. Если x — точка локального минимума, то существуют множители μ_i , $i = 1, \dots, m$, λ_j , $j = 1, \dots, k$:

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial x} = 0, & \mathcal{L}(x; \mu, \lambda) = f(x) + \sum_{i=1}^m \mu_i g_i(x) + \sum_{j=1}^k \lambda_j h_j(x) \\ g_i(x) \leq 0; & h_j(x) = 0 \text{ (исходные ограничения)} \\ \mu_i \geq 0 & \text{(двойственные ограничения)} \\ \mu_i g_i(x) = 0 & \text{(условие дополняющей нежёсткости)} \end{cases}$$

Применение условий ККТ к задаче SVM

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

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

λ_i — переменные, двойственные к ограничениям $M_i \geq 1 - \xi_i$

η_i — переменные, двойственные к ограничениям $\xi_i \geq 0$

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial w} = 0, & \frac{\partial \mathcal{L}}{\partial w_0} = 0, & \frac{\partial \mathcal{L}}{\partial \xi} = 0 \\ \xi_i \geq 0, & \lambda_i \geq 0, & \eta_i \geq 0, & i = 1, \dots, \ell \\ \lambda_i = 0 \text{ либо } M_i(w, w_0) = 1 - \xi_i, & i = 1, \dots, \ell \\ \eta_i = 0 \text{ либо } \xi_i = 0, & i = 1, \dots, \ell \end{cases}$$

Необходимые условия седловой точки функции Лагранжа

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

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

Необходимые условия седловой точки функции Лагранжа:

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

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

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

Понятие опорного вектора и типизация объектов

Система условий ККТ:

$$\begin{cases} w = \sum_{i=1}^{\ell} \lambda_i y_i x_i; & \sum_{i=1}^{\ell} \lambda_i y_i = 0; & M_i(w, w_0) \geq 1 - \xi_i \\ \xi_i \geq 0, & \lambda_i \geq 0, & \eta_i \geq 0, & \eta_i + \lambda_i = C \\ \lambda_i = 0 & \text{либо} & M_i(w, w_0) = 1 - \xi_i \\ \eta_i = 0 & \text{либо} & \xi_i = 0 \end{cases}$$

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

Типизация объектов x_i , $i = 1, \dots, \ell$:

1. $\lambda_i = 0$; $\eta_i = C$; $\xi_i = 0$; $M_i \geq 1$ — периферийный
2. $0 < \lambda_i < C$; $0 < \eta_i < C$; $\xi_i = 0$; $M_i = 1$ — **опорный**-граничный
3. $\lambda_i = C$; $\eta_i = 0$; $\xi_i > 0$; $M_i < 1$ — **опорный**-нарушитель

Двойственная задача

$$\begin{cases} -\mathcal{L}(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \rightarrow \min_{\lambda} \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0; \quad 0 \leq \lambda_i \leq C, \quad i = 1, \dots, \ell \end{cases}$$

Решение прямой задачи выражается через решение двойственной:

$$\begin{cases} \mathbf{w} = \sum_{i=1}^{\ell} \lambda_i y_i \mathbf{x}_i \\ w_0 = \langle \mathbf{w}, \mathbf{x}_i \rangle - y_i, \quad \text{для любого } i: \lambda_i > 0, \quad M_i = 1 \end{cases}$$

Линейный классификатор с признаками $f_i(\mathbf{x}) = \langle \mathbf{x}, \mathbf{x}_i \rangle$:

$$a(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^{\ell} \lambda_i y_i \langle \mathbf{x}, \mathbf{x}_i \rangle - w_0 \right)$$

S.Fine, K.Scheinberg. INCAS: An incremental active set method for SVM. 2002.

J.Platt. Fast training support vector machines using sequential minimal optimization. 1999.

Двойственная задача. Нелинейное обобщение с ядром K

$$\begin{cases} -\mathcal{L}(\lambda) = -\sum_{i=1}^{\ell} \lambda_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j y_i y_j K(x_i, x_j) \rightarrow \min_{\lambda} \\ \sum_{i=1}^{\ell} \lambda_i y_i = 0; \quad 0 \leq \lambda_i \leq C, \quad i = 1, \dots, \ell \end{cases}$$

Решение прямой задачи выражается через решение двойственной:

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

Линейный классификатор с признаками $f_i(x) = K(x, x_i)$:

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

S.Fine, K.Scheinberg. INCAS: An incremental active set method for SVM. 2002.

J.Platt. Fast training support vector machines using sequential minimal optimization. 1999.

Нелинейное обобщение SVM

Идея: заменить $\langle x, x' \rangle$ нелинейной функцией $K(x, x')$

Переход к спрямляющему пространству,
как правило, более высокой размерности: $\psi: X \rightarrow H$

Определение

Функция $K: X \times X \rightarrow \mathbb{R}$ — *ядро*, если $K(x, x') = \langle \psi(x), \psi(x') \rangle$ при некотором $\psi: X \rightarrow H$, 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:

$$\int_X \int_X K(x, x') g(x) g(x') dx dx' \geq 0 \text{ для любой } g: X \rightarrow \mathbb{R}$$

Конструктивные методы синтеза ядер

- ❶ $K(x, x') = \langle x, x' \rangle$ — ядро
- ❷ константа $K(x, x') = 1$ — ядро
- ❸ произведение ядер $K(x, x') = K_1(x, x')K_2(x, x')$ — ядро
- ❹ $\forall \psi : X \rightarrow \mathbb{R}$ произведение $K(x, x') = \psi(x)\psi(x')$ — ядро
- ❺ $K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x')$ при $\alpha_1, \alpha_2 > 0$ — ядро
- ❻ $\forall \varphi : X \rightarrow X$ если K_0 ядро, то $K(x, x') = K_0(\varphi(x), \varphi(x'))$ — ядро
- ❼ если $s : X \times X \rightarrow \mathbb{R}$ — симметричная интегрируемая функция, то $K(x, x') = \int_X s(x, z)s(x', z) dz$ — ядро
- ❽ если K_0 — ядро и функция $f : \mathbb{R} \rightarrow \mathbb{R}$ представима в виде сходящегося степенного ряда с неотрицательными коэффициентами, то $K(x, x') = f(K_0(x, x'))$ — 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 $\psi: X \rightarrow H$,
при которых $K(x, x') = \langle \psi(x), \psi(x') \rangle_H$

Разложим квадрат скалярного произведения:

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

Таким образом,

$$H = \mathbb{R}^3, \quad \psi: (u_1, u_2) \mapsto (u_1^2, u_2^2, \sqrt{2}u_1 u_2),$$

линейной поверхности в пространстве H соответствует квадратичная поверхность в исходном пространстве X

Примеры ядер

- ❶ квадратичное ядро, $X = \mathbb{R}^n$, $\dim H = \frac{1}{2}n(n+1)$

$$K(x, x') = \langle x, x' \rangle^2$$

- ❷ полиномиальное с мономами степени d , $\dim H = C_{n+d-1}^d$

$$K(x, x') = \langle x, x' \rangle^d$$

- ❸ полиномиальное с мономами всех степеней от 0 до d

$$K(x, x') = (\langle x, x' \rangle + 1)^d$$

- ❹ линейный нейрон с сигмоидной функцией активации

$$K(x, x') = \text{th}(k_1 \langle x, x' \rangle - k_0), \quad k_0, k_1 \geq 0$$

- ❺ радиальная базисная функция (RBF ядро)

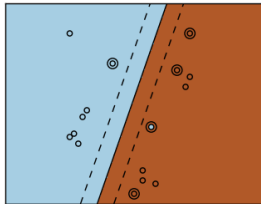
$$K(x, x') = \exp(-\gamma \|x - x'\|^2)$$

Классификация с различными ядрами

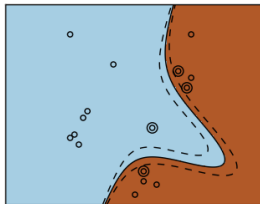
Гиперплоскость в спрямляющем пространстве соответствует нелинейной разделяющей поверхности в исходном

Примеры с различными ядрами $K(x, x')$:

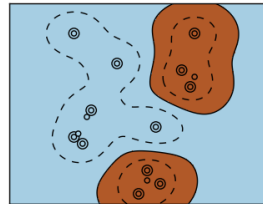
линейное
 $\langle x, x' \rangle$



полиномиальное
 $(\langle x, x' \rangle + 1)^d, \quad d=3$



гауссовское (RBF)
 $\exp(-\gamma \|x - x'\|^2)$

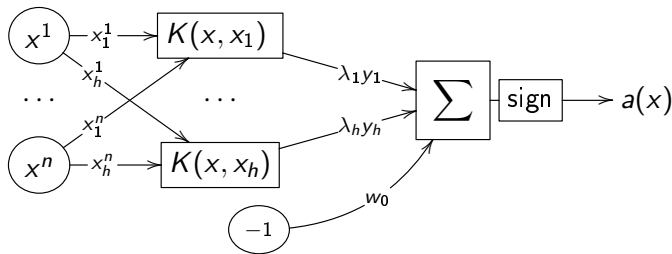


Пример из Python SkLearn: <http://scikit-learn.org/dev>

SVM — двухслойная нейросеть и метрический классификатор

Перенумеруем объекты так, чтобы $x_1, \dots, x_h$ были опорными

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



Первый слой вместо скалярных произведений вычисляет ядра

Веса первого слоя — это сами опорные объекты

Метрический классификатор, если K — функция близости

Преимущества и недостатки SVM

Преимущества перед двухслойными нейронными сетями:

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

Недостатки классического SVM (некоторые сейчас устраним):

- нет общих подходов к оптимизации $K(x, x')$ под задачу
- на больших данных SVM обучается медленнее SG
- нет «вероятностного выхода» $p(y|x)$ как в лог-регрессии
- приходится подбирать константу C
- нет «встроенного» отбора признаков
- шумовые объекты (выбросы) являются опорными

SVM и регуляризованная логистическая регрессия

$a(x; w, w_0) = \text{sign}(\langle x, w \rangle - w_0)$ — линейная модель классификации
 $M_i = (\langle x_i, w \rangle - w_0) y_i$ — отступ объекта x_i обучающей выборки

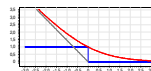
Критерий SVM (ERM с аппроксимацией и регуляризацией):

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



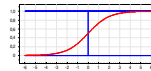
Критерий RLR (регуляризованная лог-регрессия):

$$\sum_{i=1}^{\ell} \ln(1 + e^{-M_i}) + \tau \|w\|^2 \rightarrow \min_{w, w_0}$$



Логистическая регрессия имеет probabilistic output:

$$P(y|x, w) = \sigma((\langle x, w \rangle - w_0) y), \quad \sigma(M) = \frac{1}{1 + e^{-M}}$$



Как в SVM оценивать апостериорные вероятности классов?

Калибровка Платта (classifier with probabilistic output)

Дано: выборка $(x_i, y_i)_{i=1}^{\ell}$, $y_i \in \{-1, +1\}$;
обученный разделяющий классификатор $a(x) = \text{sign } b(x, w)$,
исходно не имеющий вероятностной интерпретации.

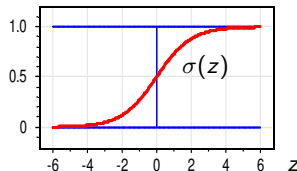
Найти: вероятностную модель классификации $P(y|x)$

Модель условной вероятности:

$$\pi(x; \kappa, \delta) = P(y=1|x) = \sigma(\kappa b(x, w) + \delta)$$

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

важное её свойство: $\sigma(-z) = 1 - \sigma(z)$



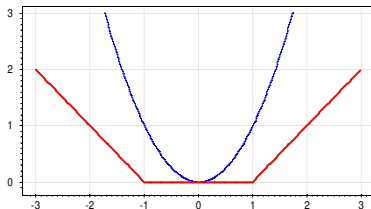
Критерий: максимум log-правдоподобия для калибровки
коэффициентов κ, δ по *контрольной* выборке (hold-out):

$$\sum_{y_i=-1} \ln(1 - \pi(x_i; \kappa, \delta)) + \sum_{y_i=+1} \ln \pi(x_i; \kappa, \delta) \rightarrow \max_{\kappa, \delta}$$

SVM-регрессия

Модель регрессии: $a(x, w) = \langle x, w \rangle - w_0$, $w \in \mathbb{R}^n$, $w_0 \in \mathbb{R}$

Функции потерь: $\mathcal{L}(\varepsilon) = \varepsilon^2$, $\mathcal{L}(\varepsilon) = (|\varepsilon| - \delta)_+$, $\varepsilon = a - y$



Постановка задачи:

$$\sum_{i=1}^{\ell} (|\langle w, x_i \rangle - w_0 - y_i| - \delta)_+ + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

Задача решается путём замены переменных
и сведения к задаче квадратичного программирования

SVM-регрессия

Замена переменных:

$$\begin{aligned}\xi_i^+ &= (\langle w, x_i \rangle - w_0 - y_i - \delta)_+ \\ \xi_i^- &= (-\langle w, x_i \rangle + w_0 + y_i - \delta)_+\end{aligned}$$

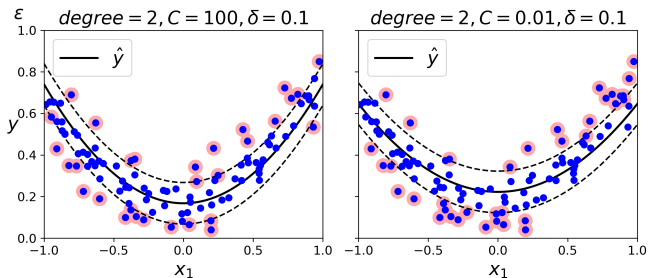
Постановка задачи SVM-регрессии:

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{\ell} (\xi_i^+ + \xi_i^-) \rightarrow \min_{w, w_0, \xi^+, \xi^-} \\ y_i - \delta - \xi_i^- \leq \langle w, x_i \rangle - w_0 \leq y_i + \delta + \xi_i^+, \quad i = 1, \dots, \ell \\ \xi_i^- \geq 0, \quad \xi_i^+ \geq 0, \quad i = 1, \dots, \ell \end{cases}$$

- выпуклая задача квадратичного программирования
- следовательно, решение единственное
- решение разреженное, выражается через опорные векторы
- возможна замена $\langle x, x_i \rangle$ ядром $K(x, x_i)$

SVM-регрессия. Пример 1

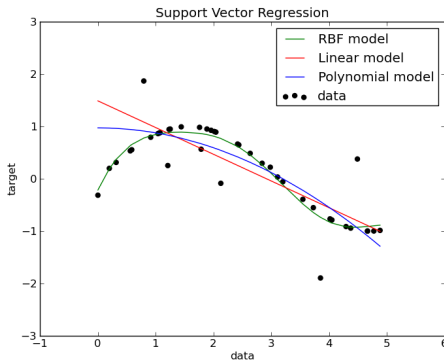
SVM-регрессия с полиномиальным ядром степени 2:



- Выделены опорные векторы
- Результат слабо зависит от константы C

SVM-регрессия. Пример 2

Сравнение SVM-регрессии с гауссовским (RBF) ядром, линейной и полиномиальной регрессией:



- Удачный выбор ядра имеет значение!

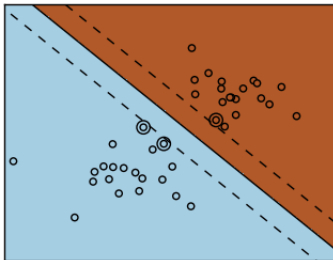
http://scikit-learn.org/0.5/auto_examples/svm/plot_svm_regression.html

Влияние константы C на решение SVM

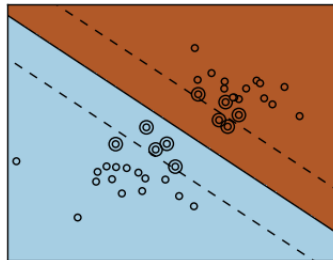
SVM — аппроксимация и регуляризация эмпирического риска:

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

большой C
слабая регуляризация



малый C
сильная регуляризация



Пример из Python SkLearn: <http://scikit-learn.org/dev>

1-norm SVM (LASSO SVM)

Аппроксимация эмпирического риска с L_1 -регуляризацией:

$$\sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \mu \sum_{j=1}^n |w_j| \rightarrow \min_{w, w_0}$$

- ⊕ Отбор признаков с параметром *селективности* μ :
чем больше μ , тем меньше признаков останется
- ⊖ Слишком агрессивный отбор признаков:
 - по мере увеличения μ признак может быть отброшен,
 - хотя у существенно зависит от него,
 - даже когда ещё не все шумовые признаки отброшены

Bradley P., Mangasarian O. Feature selection via concave minimization and support vector machines // ICML 1998.

1-norm SVM (LASSO SVM)

Аппроксимация эмпирического риска с L_1 -регуляризацией:

$$\sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \mu \sum_{j=1}^n |w_j| \rightarrow \min_{w, w_0}$$

Почему L_1 -регуляризатор приводит к отбору признаков?

Замена переменных: $u_j = \frac{1}{2}(|w_j| + w_j)$, $v_j = \frac{1}{2}(|w_j| - w_j)$.

Тогда $w_j = u_j - v_j$ и $|w_j| = u_j + v_j$;

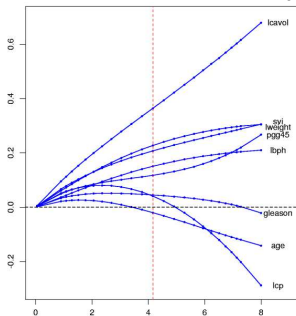
$$\begin{cases} \sum_{i=1}^{\ell} (1 - M_i(u - v, w_0))_+ + \mu \sum_{j=1}^n (u_j + v_j) \rightarrow \min_{u, v} \\ u_j \geq 0, \quad v_j \geq 0, \quad j = 1, \dots, n \end{cases}$$

чем больше μ , тем больше индексов j таких, что $u_j = v_j = 0$, но тогда $w_j = 0$, значит, **признак не учитывается**

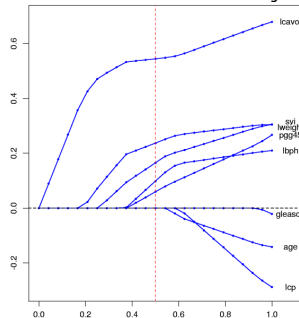
Сравнение L_2 и L_1 регуляризации

Зависимость весов w_j от коэффициента $\frac{1}{\mu}$

L_2 регуляризатор: $\mu \sum_j w_j^2$



L_1 регуляризатор: $\mu \sum_j |w_j|$



Задача из UCI: prostate cancer (диагностика рака)

Doubly Regularized SVM (Elastic Net SVM)

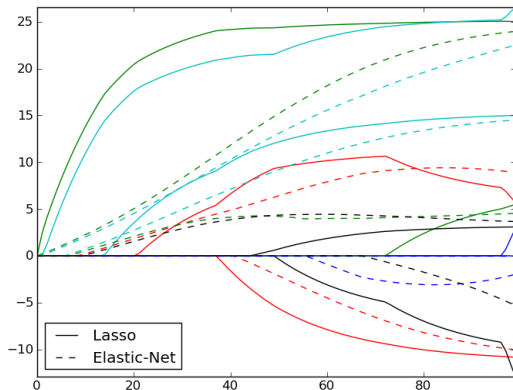
$$C \sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \mu \sum_{j=1}^n |w_j| + \frac{1}{2} \sum_{j=1}^n w_j^2 \rightarrow \min_{w, w_0}$$

- ⊕ Параметр селективности μ управляет отбором признаков: чем больше μ , тем меньше признаков останется
- ⊕ Есть эффект группировки (grouping effect): значимые зависимые признаки отбираются вместе
- ⊖ Шумовые признаки также группируются и могут вместе оставаться в модели
- ⊖ Приходится подбирать два параметра регуляризации μ, τ (есть специальные методы — regularization path)

Doubly Regularized SVM (Elastic Net SVM)

Elastic Net менее жёстко отбирает признаки.

Зависимости весов w_j от коэффициента $\log \frac{1}{\mu}$:

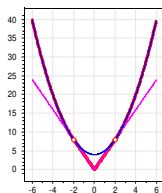


scikit-learn.org/0.5/auto_examples/glm/plot_lasso_coordinate_descent_path.html

Support Features Machine (SFM)

$$C \sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \sum_{j=1}^n R_{\mu}(w_j) \rightarrow \min_{w, w_0}$$

$$R_{\mu}(w_j) = \begin{cases} 2\mu|w_j|, & |w_j| \leq \mu \\ \mu^2 + w_j^2, & |w_j| \geq \mu \end{cases}$$



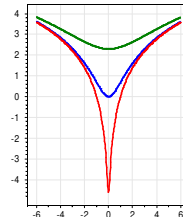
- ⊕ Только один параметр регуляризации μ
- ⊕ Отбор признаков с *параметром селективности* μ
- ⊕ *Эффект группировки*: значимые зависимые признаки ($|w_j| > \mu$) входят в решение совместно (как в Elastic Net)
- ⊕ Шумовые признаки ($|w_j| < \mu$) не группируются и подавляются независимо друг от друга (как в LASSO)

Tatarchuk A., Urlov E., Mottl V., Windridge D. A support kernel machine for supervised selective combining of diverse pattern-recognition modalities. 2010.

Relevance Features Machine (RFM)

$$C \sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \sum_{j=1}^n \ln(w_j^2 + \frac{1}{\mu}) \rightarrow \min_{w, w_0}$$

$$R(w) = \ln(w^2 + \frac{1}{\mu}) \text{ при } \mu = 0.1, 1, 100$$



- ⊕ Только один параметр регуляризации μ
- ⊕ Отбор признаков с параметром *селективности* μ
- ⊕ Есть эффект группировки
- ⊕ Лучше отбирает набор значимых признаков, когда они только совместно обеспечивают хорошее решение

Tatarchuk A., Mottl V., Eliseyev A., Windridge D. Selectivity supervision in combining pattern recognition modalities by feature- and kernel-selective Support Vector Machines. 2008.

Метод релевантных векторов RVM (Relevance Vector Machine)

Положим, как и в SVM, при некоторых $\lambda_i \geq 0$

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

причём опорным векторам x_i соответствуют $\lambda_i \neq 0$

Проблема: Какие из коэффициентов λ_i лучше обнулить?

Идея: пусть регуляризатор зависит не от w , а от λ_i ,
и пусть λ_i независимые, гауссовские, с дисперсиями α_i :

$$p(\lambda) = \frac{1}{(2\pi)^{\ell/2} \sqrt{\alpha_1 \cdots \alpha_\ell}} \exp \left(- \sum_{i=1}^{\ell} \frac{\lambda_i^2}{2\alpha_i} \right)$$

$$\sum_{i=1}^{\ell} (1 - M_i(w(\lambda), w_0))_+ + \frac{1}{2} \sum_{i=1}^{\ell} \left(\ln \alpha_i + \frac{\lambda_i^2}{\alpha_i} \right) \rightarrow \min_{\lambda, \alpha}$$

Преимущества и недостатки RVM

Преимущества:

- ⊕ Опорных векторов, как правило, меньше (более «разреженное» решение)
- ⊕ Шумовые выбросы уже не входят в число опорных
- ⊕ Не надо искать параметр регуляризации (вместо этого α ; оптимизируются в процессе обучения)
- ⊕ Аналогично SVM, можно использовать ядра

Недостатки:

- ⊖ RVM не гарантирует улучшения качества классификации

M. E. Tipping. The relevance vector machine. 2000.

C. M. Bishop, M. E. Tipping. Variational relevance vector machine. 2000

- *SVM* — лучший метод линейной классификации
- *SVM* легко обобщается и на задачи регрессии
- С помощью *ядер* (kernel trick) *SVM* изящно обобщается для нелинейной классификации и нелинейной регрессии
- *Регуляризация* увеличивает зазор, устраняет мультиколлинеарность и уменьшает переобучение
- *Аппроксимация пороговой функции потерь $\mathcal{L}(M)$* также увеличивает зазор и повышает надёжность классификации

Нетривиальный инсайт:

- *Негладкость функции потерь* приводит к отбору объектов
- *Негладкость регуляризатора* приводит к отбору признаков

В. Н. Вапник, А. Я. Лернер. Узнавание образов при помощи обобщенных портретов. 1963.

C. Cortes, V. Vapnik. Support vector networks. 1995.