Факультет
математики
и компьютерных
наук
СПбГУ

Машинное обучение

Лекция 8. Линейные методы классификации и регрессии: метод опорных векторов

29 октября 2021

Пятиминутка

1. Выпишите оптимизационную задачу метода наименьших квадратов
2. Назовите стратегии устранения мультиколлинеарности
3. Приведите примеры прикладных задач матричного разложения

Задача обучения линейного классификатора (напоминание)

Дано: Обучающая выборка $X^\ell = (x_i, y_i)_{i=1}^\ell$

x_i — объекты, векторы из множества $X = \mathbb{R}^n$

y_i — метки классов, элементы множества $Y = \{-1, +1\}$

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

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

Критерий — минимизация эмпирического риска:

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

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

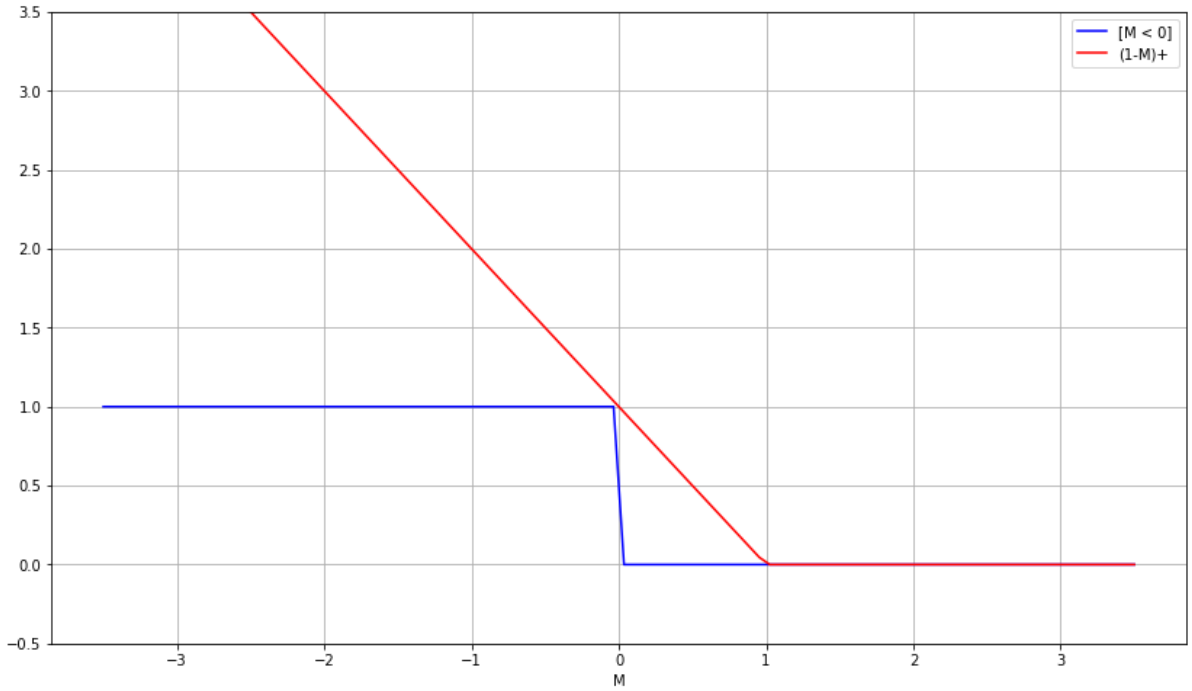
Аппроксимация и регуляризация эмпирического риска

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

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

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

In [1]:
```
import matplotlib.pyplot as plt
import numpy as np

fig, ax = plt.subplots()

ax.set_xlabel('M')

x = np.linspace(-3.5, 3.5, num=100)
acc_loss = x < 0
V_M = (1 - x) * ((1 - x) > 0)

ax.plot(x, acc_loss, 'b', label='[M < 0]')
ax.plot(x, V_M, 'r', label='(1-M)+')

ax.set_ylim(-0.5, 3.5)
ax.grid(True)

fig.set_size_inches(14, 8)
plt.legend(loc='best')
plt.show()
```



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:

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

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

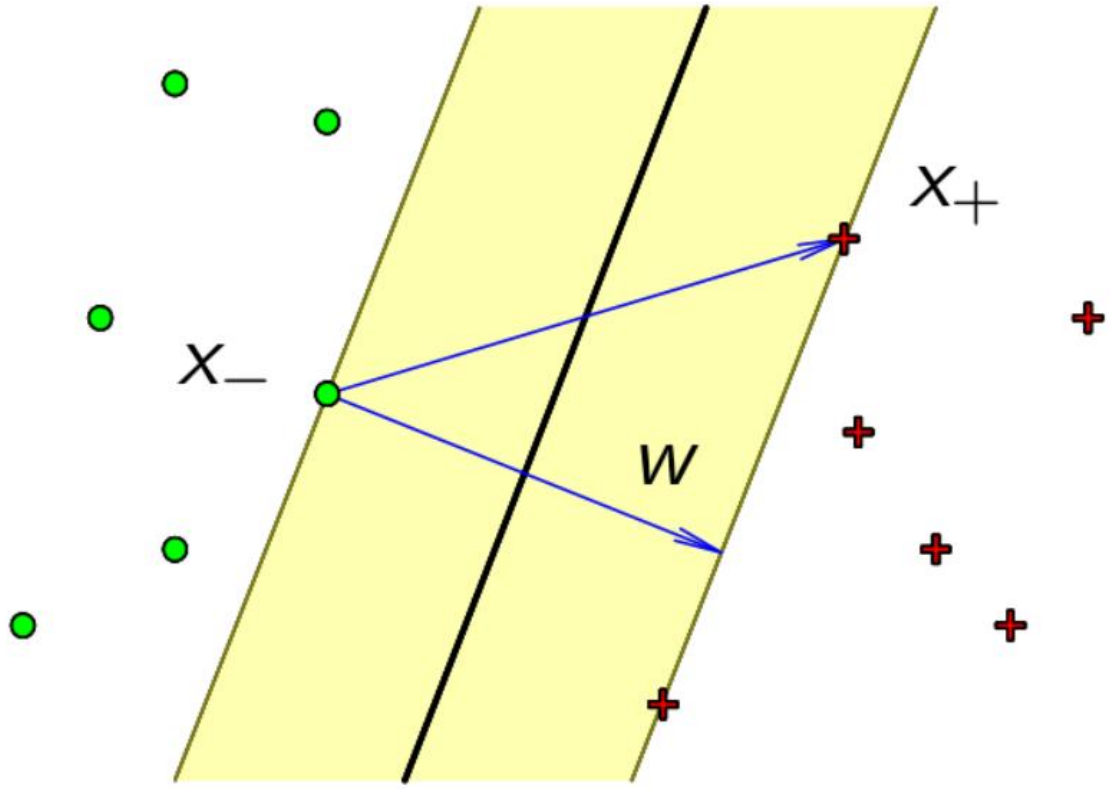
Разделяющая полоса (разделяющая гиперплоскость посередине): $\{x : -1 \leq \langle w, x \rangle - w_0 \leq 1\}$

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

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

Ширина полосы:

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



Обоснование кусочно-линейной функции потерь

Линейно разделимая выборка

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

Переход к линейно неразделимой выборке (**эвристика**)

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

Эквивалентная задача безусловной минимизации:

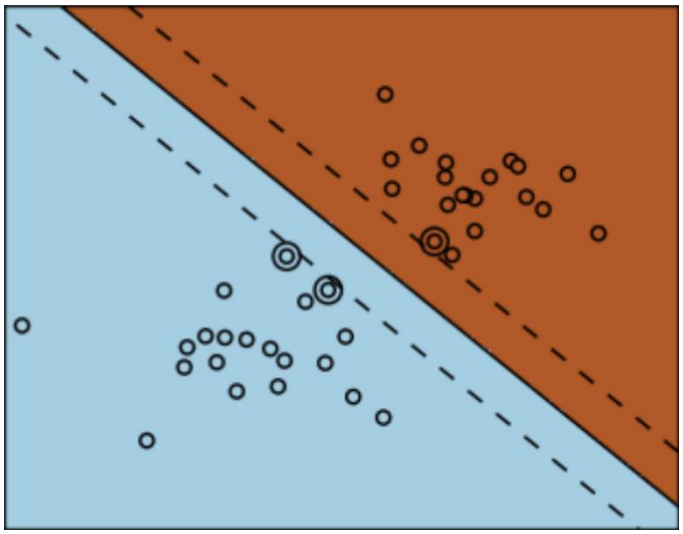
$$C\sum_{i=1}^{\ell}(1 - M_i(w,w_0))_+ + \frac{1}{2}\|w\|^2 \to \min_{w,w_0}$$

Влияние константы C на решение SVM

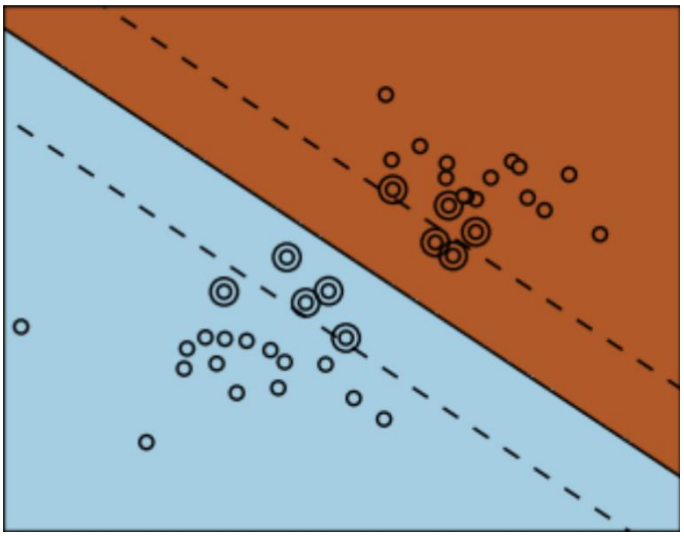
SVM — аппроксимация и регуляризация эмпирического риска:

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

большое C, слабая регуляризация



малое C, сильная регуляризация



Пример из Python SkLearn

Условия Каруша-Куна-Таккера (ККТ)

Задача математического программирования:

$$\begin{cases} f(x) \to \min\limits_x \\ g_i(x) \leq 0, i = 1, \ldots, m \\ h_j(x) = 0, j = 1, \ldots, k \end{cases}$$

Необходимые условия. Если x — точка локального минимума, то существуют множители $\mu_i, i = 1, \ldots, m, \lambda_j, j = 1, \ldots, k$:

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

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 $\xi_i \geq 0$

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

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

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

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

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

> **Вопрос 1:** Кто помнит доказательство?

Понятие опорного вектора

Типизация объектов:

1. $\lambda_i = 0, \eta_i = C, \xi_i = 0, M_i \geq 1$

— периферийные (неинформативные) объекты

1. $0 < \lambda_i < C, 0 < \eta_i < C, \xi_i = 0, M_i = 1$

— опорные граничные объекты

1. $\lambda_i = C, \eta_i = 0, \xi_i > 0, M_i < 1$

— опорные-нарушители

Определение

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

Двойственная задача

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

Решение прямой задачи выражается через решение двойственной:

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

Линейный классификатор с признаками $f_i(x) = \langle x_i, x\rangle$

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

Нелинейное обобщение SVM

Идея: заменить скалярное произведение $\langle x, x'\rangle$ нелинейной функцией $K(x, x')$. Переход к спрямляющему пространству, как правило, более высокой размерности: $\psi : X \to H$

Определение

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

Теорема

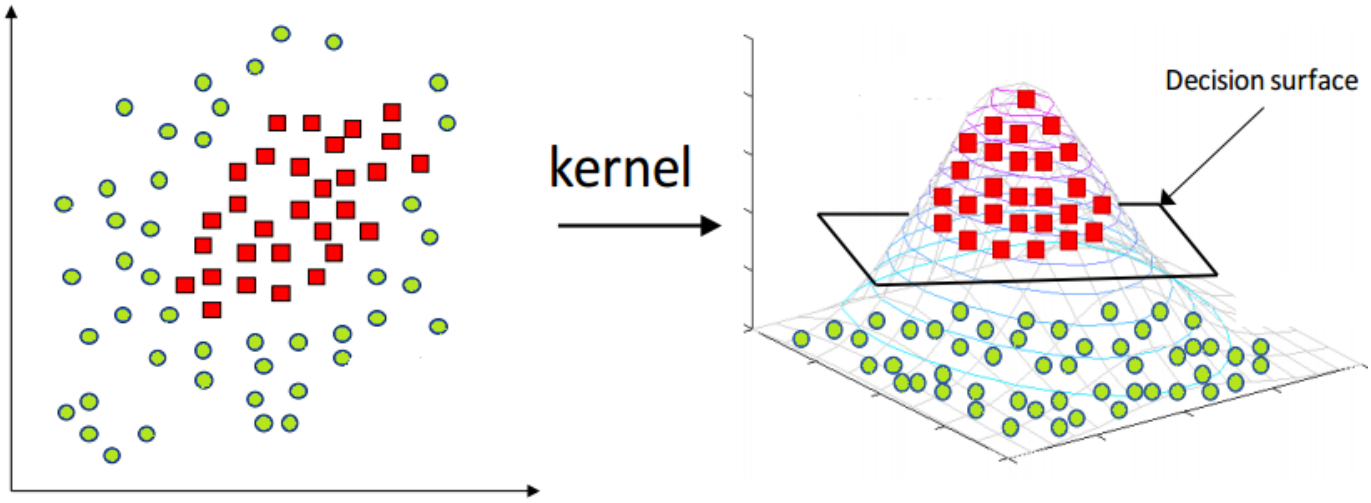
Функция $K(x, x')$ является ядром тогда и только тогда, когда она симметрична ($K(x, x') = K(x', x)$) и неотрицательно определена:

$\int_X \int_X K(x, x')g(x)g(x')dxdx' \ge 0$ для любой $g : X \to \mathbb{R}$

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. Линейная комбинация $K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x')$ при $\alpha_i > 0$ — ядро
6. $\forall \phi : X \to X$ если K_0 ядро, то $K(x, x') = K_0(\phi(x), \phi(x'))$ — ядро
7. если $s : X \times X \to \mathbb{R}$ — 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 $K(x, x') = f(K_0(x, x'))$ — ядро

Пример



Вопрос 2: Какое ядро подойдёт для такого?

Примеры ядер

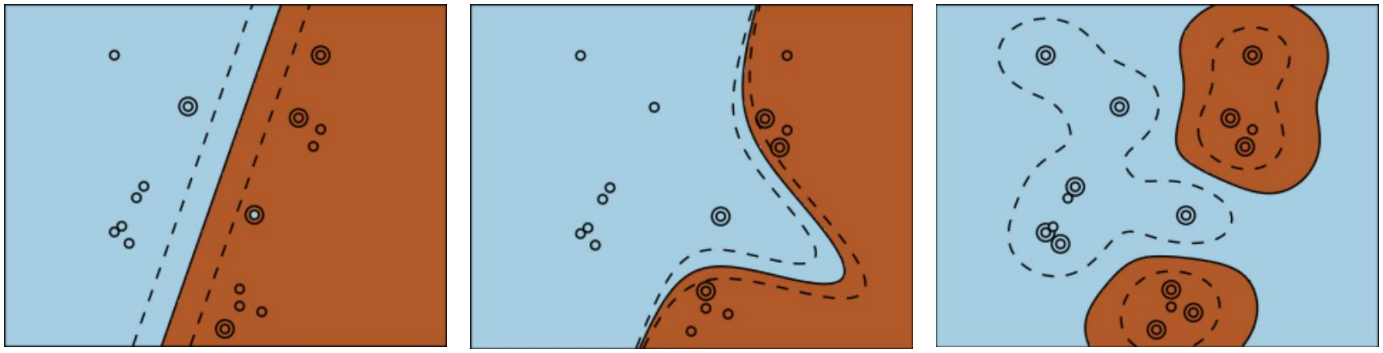
- $K(x, x') = \langle x, x' \rangle^2$ — квадратичное ядро
- $K(x, x') = \langle x, x' \rangle^d$ — полиномиальное ядро с мономами степени d
- $K(x, x') = (\langle x, x' \rangle + 1)^d$ — полиномиальное ядро с мономами степени $\leq d$
- $K(x, x') = \tanh(k_1 \langle x, x' \rangle - k_0), k_0, k_1 \geq 0$ — нейросеть с сигмоидными функциями активации
- $K(x, x') = \exp(-\gamma \|x - x'\|^2)$ — сеть радиальных базисных функций (RBF ядро)

Классификация с различными ядрами

Гиперплоскость в спрямляющем пространстве соответствует нелинейной разделяющей поверхности в исходном.

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

линейное, $\left$ полиномиальное, $(\left + 1)^d, d=3$ гауссовское, $\exp(-\gamma\|x - x'\|^2)$

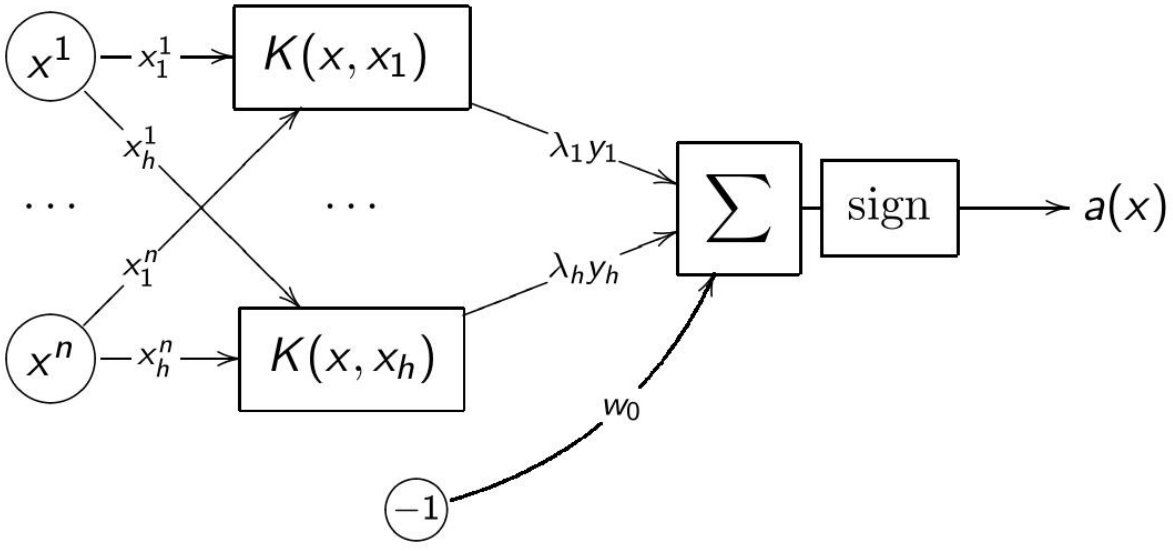


Пример из Python SkLearn

SVM как двухслойная нейронная сеть

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

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



Первый слой вместо скалярных произведений вычисляет ядра.

Преимущества и недостатки SVM

Преимущества SVM перед двухслойными нейронными сетями:

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

Недостатки классического SVM:

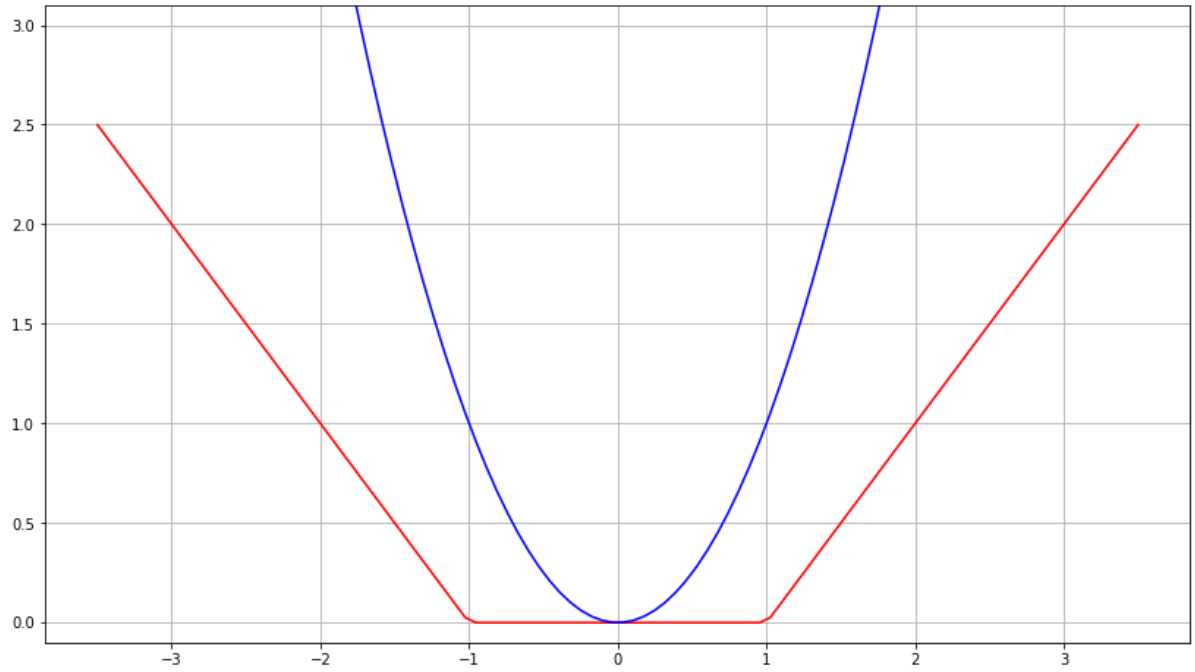
- Нет общих подходов к оптимизации $K(x, x')$ под задачу
- На больших данных SVM может обучаться медленно
- Нет «встроенного» отбора признаков
- Приходится подбирать константу C

SVM-регрессия

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

Функция потерь: $\mathcal{L}(|\varepsilon| - \delta)_+$ в сравнении с $\mathcal{L}(\varepsilon) = \varepsilon^2$:

In [2]:
```
fig, ax = plt.subplots()

x = np.linspace(-3.5, 3.5, num=100)
L_1 = (abs(x) - 1) * (abs(x) - 1 > 0)
L_2 = x**2

ax.plot(x, L_1, 'r')
ax.plot(x, L_2, 'b')

ax.set_ylim(-0.1, 3.1)
ax.grid(True)

fig.set_size_inches(14, 8)
plt.show()
```



Постановка задачи:

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

Задача решается путём замены переменных и сведения к задаче квадратичного программирования

SVM-регрессия

Замена переменных:

$$\varepsilon_i^+ = (\langle w, x_i \rangle - w_0 - y_i - \delta)_+$$

$$\varepsilon_i^- = (-\langle w, x_i \rangle + w_0 + y_i - \delta)_+$$

Постановка задачи SVM-регрессии:

$$\begin{cases} \frac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{\ell}(\varepsilon_i^+ + \varepsilon_i^-) \to \min\limits_{w, w_0, \varepsilon^+, \varepsilon^-} \\ y_i - \delta - \varepsilon_i^- \le \langle w, x_i \rangle - w_0 \le y_i + \delta + \varepsilon_i^+, i = 1, \ldots, \ell \\ \varepsilon_i^- \ge 0, \varepsilon_i^+ \ge 0, i = 1, \ldots, \ell \end{cases}$$

Это задача квадратичного программирования с линейными ограничениями-неравенствами, решается также сведением к двойственной задаче.

1-norm SVM (LASSO SVM)

LASSO — Least Absolute Shrinkage and Selection Operator. Аппроксимация эмпирического риска с L_1-регуляризацией:

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

- Отбор признаков с параметром *селективности* μ: чем больше μ, тем меньше признаков останется
- LASSO начинает отбрасывать значимые признаки, когда ещё не все шумовые отброшены
- Нет *эффекта группировки* (grouping effect): значимые зависимые признаки должны отбираться вместе и иметь примерно равные веса w_j

---

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

Вопрос 2: Почему L_1-регуляризатор приводит к отбору признаков?

1-norm SVM (LASSO SVM)

Аппроксимация эмпирического риска с L_1-регуляризацией:

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

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 $|w_j| = u_j + v_j$

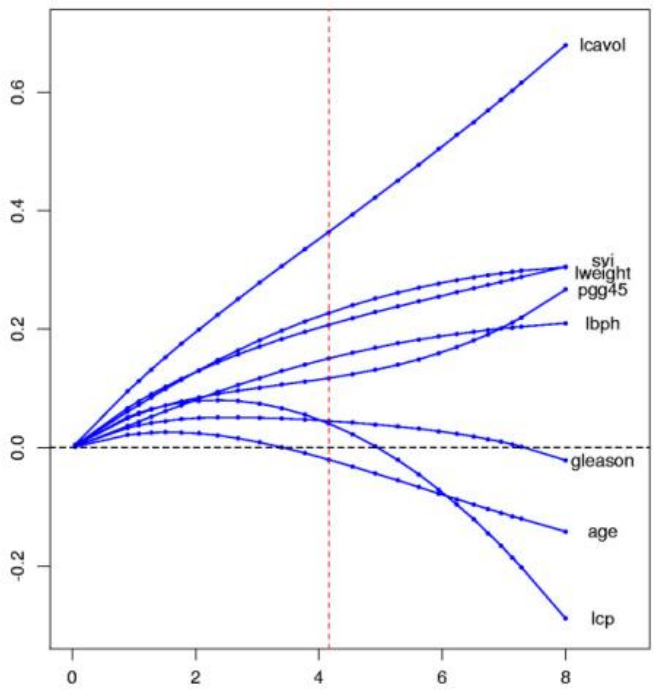
$$\begin{cases} \sum\limits_{i=1}^{\ell}(1 - M_i(u - v, w_0))_+ + \mu \sum\limits_{j=1}^{n}(u_j + v_j) \to \min\limits_{u,v} \\ u_j \geq 0, v_j \geq 0, j = 1, \ldots, n \end{cases}$$

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

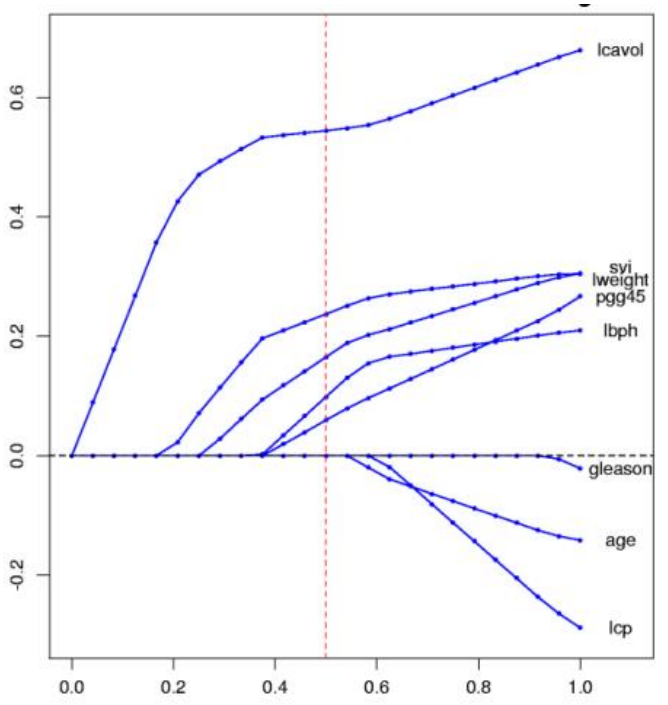
Сравнение L_2 и L_1 регуляризации

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)

T.Hastie, R. Tibshirani, J.Friedman. The Elements of Statistical Learning. 2001.

Doubly Regularized SVM (Elastic Net SVM)

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

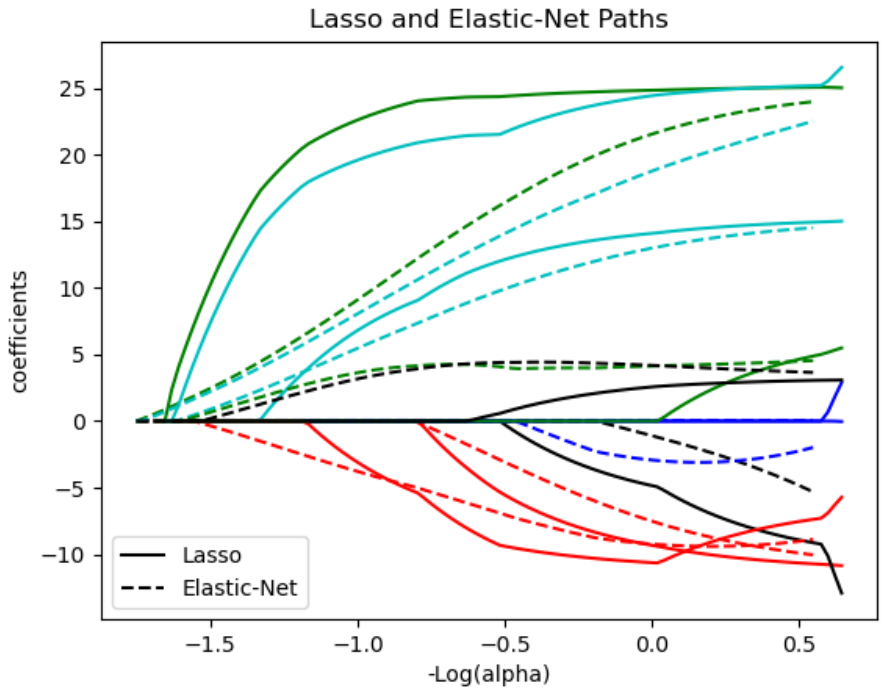
- Отбор признаков с параметром селективности μ: чем больше μ, тем меньше признаков останется
- Есть эффект группировки
- Шумовые признаки также группируются вместе, и группы значимых признаков могут отбрасываться, когда ещё не все шумовые отброшены

Li Wang, Ji Zhu, Hui Zou. The doubly regularized support vector machine. 2006

Doubly Regularized SVM (Elastic Net SVM)

Elastic Net менее жёстко отбирает признаки.

Зависимости весов w_j от коэффициента $\log \frac{1}{\mu} = -\log \alpha$

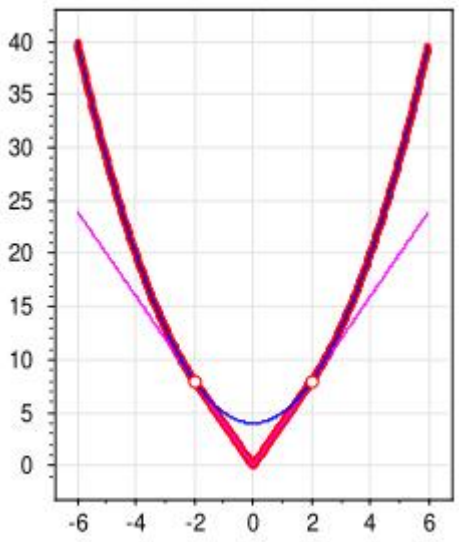


Пример из Python SkLearn: scikitlearn.org/0.5/auto_examples/glm/plot_lasso_coordinate_descent_path.html

Support Features Machine (SFM)

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

$$R_\mu(w_j) = \begin{cases} 2\mu|w_j|, |w_j| \le \mu \\ \mu^2 + w_j^2, \ge \mu \end{cases}$$



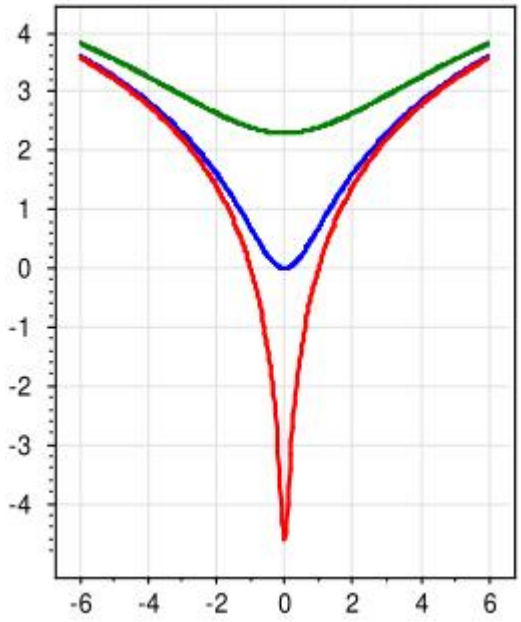
- Отбор признаков с параметром селективности μ
- Есть эффект группировки
- Значимые зависимые признаки ($|w_j| > \mu$) группируются и входят в решение совместно (как в Elastic Net)
- Шумовые признаки ($|w_j| < \mu$) подавляются независимо (как в LASSO)

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)

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

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



- Отбор признаков с параметром селективности μ: чем больше μ, тем меньше признаков останется
- 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.

Резюме по линейным классификаторам

- SVM — лучший метод линейной классификации
- SVM изящно обобщается для нелинейной классификации, для линейной и нелинейной регрессии
- *Аппроксимация пороговой функции потерь* $\mathcal{L}(M)$ увеличивает зазор и повышает качество классификации
- *Регуляризация* устраняет мультиколлинеарность и уменьшает переобучение, по сути она эквивалентна введению априорного распределения в пространстве коэффициентов
- Негладкость функции потерь приводит к отбору объектов
- Негладкость регуляризатора приводит к отбору признаков