Машинное обучение. Линейные модели

Алексей Колесов

Белорусский государственный университет

11 сентября 2019 г.

Линейные модели

- часто используются
- простой алгоритм обучения
- интерпретируемы

Линейные модели

Класс аффинных функций:

$$H_d = \{h_{w,b} : w \in \mathbb{R}^d, b \in \mathbb{R}\}$$

где

$$h_{w,b}(x) = \langle w, x \rangle + b = \left(\sum_{i=1}^{d} w_i x_i \right) + b$$

или в другой нотации:

$$H_d = \{x \mapsto \langle w, x \rangle + b : w \in \mathbb{R}^d, b \in \mathbb{R}\}$$

Линейные модели

Семейства гипотез обычно строятся как композиции функции $\phi : \mathbb{R} \to Y$ и H_d

- для бинарной классификации: $\phi = \mathrm{sign}$
- для регрессии: $\phi = \mathrm{id}$

Однородные линейные функции

- $h_w(x) = \langle w, x \rangle$ называется однородной (гомогенной) линейной функцией
- зачастую b (bias) вносят в w, и добавляют во все x_i константный признак (1)

$$w' = (b, w_1, \ldots, w_d) \in \mathbb{R}^{d+1}$$

$$x' = (1, x_1, \ldots, x_d) \in \mathbb{R}^{d+1}$$

тогда

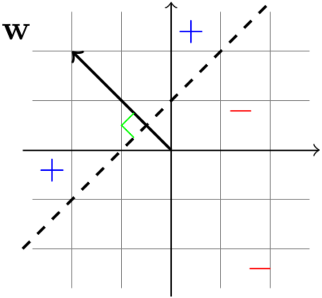
$$h_{w,b}(x) = \langle w, x \rangle + b = h_{w'}(x) = \langle w', x' \rangle$$

Содержание

Класс гипотез: полупространства

Решаем задачу бинарной классификации: $X = \mathbb{R}^d$, $Y = \{-1, 1\}$

$$\mathrm{HS}_d = \mathrm{sign} \circ H_d = \{x \mapsto \mathrm{sign}(h_{w,b}(x)) : h_{w,b} \in H_d\}$$



Гарантии

- изучаемый с помощью ERM-парадигмы, если размер выборки $\Omega\left(\frac{d+\log(1/\delta)}{\epsilon}\right)$ $\Rightarrow$ хотим алгоритм для нахождения ERM-гипотезы
- агностический случай — вычислительно сложен[1] $\Rightarrow$ обычно заменяют функцию потерь
- рассмотрим алгоритмы в случае предположения о реализуемости (**разделимая** выборка)

[1] Ben-David & Simon 2001

Задача линейного программирования

Задача **линейного программирования** — задача максимизации линейной функции при линейных ограничениях:

$$\max_{w \in \mathbb{R}^d} \langle u, w \rangle$$

$$Aw \geqslant v$$

- $w \in \mathbb{R}^d$ — искомый вектор переменных
- $A \in \mathbb{R}^{m \times d}$
- $v, u \in \mathbb{R}^d$

Задачи эффективно разрешимы (за полиномиальную от m, d и длины чисел сложность)

Сведение к ЛП: постановка

Цель: свести нахождение ERM-гипотезы для класса HS_d в случае разделимой выборки к задаче линейного программирования

Дано:

$$S = \{(x_i, y_i)\}_{i=1}^m \text{ — выборка}$$

Хотим: найти w (рассмотрим однородный случай), такой что:

$$\mathrm{sign}(\langle w, x_i \rangle) = y_i,\ \forall i = 1, \ldots, m$$

Сведение к ЛП

Ищем w, такой что:

$$\mathrm{sign}(\langle w, x_i \rangle) = y_i, \ \forall i = 1, \dots, m$$

По-другому:

$$y_i \langle w, x_i \rangle > 0, \ \forall i = 1, \dots, m$$

Обозначим за w^* вектор, который удовлетворяет предыдущему свойству (такой есть, так как выборка разделима)

Сведение к ЛП

Знаем, что есть w^*, такой что:

$$y_i \langle w^*, x_i \rangle > 0, \ \forall i = 1, \ldots, m$$

Пусть:

$$\gamma = \min_i (y_i \langle w^*, x_i \rangle)$$
$$\overline{w} = \frac{w^*}{\gamma}$$

Имеем:

$$y_i \langle \overline{w}, x_i \rangle = \frac{1}{\gamma} \langle w^*, x_i \rangle \geqslant 1, \ \forall i = 1, \ldots, m$$

Сведение к ЛП: итог

- доказали, что если выборка разделима, то $\exists w \in \mathbb{R}^d$, такой что $y_i\langle w, x_i\rangle \geqslant 1 \; \forall i = 1, \ldots, m$
- w, удовлетворяющий этим условиям, задаёт ERM-гипотезу в классе линейных моделей
- можно найти с помощью задачи ЛП

Сведение к ЛП: итог

- доказали, что если выборка разделима, то $\exists w \in \mathbb{R}^d$, такой что $y_i \langle w, x_i \rangle \geqslant 1 \ \forall i = 1, \ldots, m$
- w, удовлетворяющий этим условиям, задаёт ERM-гипотезу в классе линейных моделей
- можно найти с помощью задачи ЛП
    - A — матрица, в которой каждая строка — объект x_i, умноженный на y_i

Сведение к ЛП: итог

- доказали, что если выборка разделима, то $\exists w \in \mathbb{R}^d$, такой что $y_i \langle w, x_i \rangle \geqslant 1 \ \forall i = 1, \ldots, m$
- w, удовлетворяющий этим условиям, задаёт ERM-гипотезу в классе линейных моделей
- можно найти с помощью задачи ЛП
  - A — матрица, в которой каждая строка — объект x_i, умноженный на y_i
  - v — вектор из m единиц

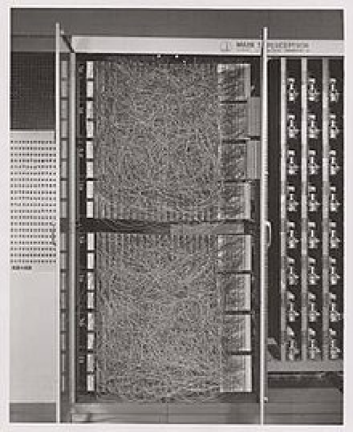
Сведение к ЛП: итог

- доказали, что если выборка разделима, то $\exists w \in \mathbb{R}^d$, такой что $y_i \langle w, x_i \rangle \geqslant 1 \; \forall i = 1, \ldots, m$
- w, удовлетворяющий этим условиям, задаёт ERM-гипотезу в классе линейных моделей
- можно найти с помощью задачи ЛП
  - A — матрица, в которой каждая строка — объект x_i, умноженный на y_i
  - v — вектор из m единиц
  - u — произвольный (например, нулевой) вектор

Алгоритм Perceptron: общее описание

- итеративный алгоритм, предложенный Розенблатом в 1958-м году
- начинает с $w^{(1)}$ — нулевого вектора
- на каждом шаге находит неправильно классифицируемый объект и «подправляет» w
- останавливается, когда найдена ERM-гипотеза

Mark I Perceptron



Алгоритм Perceptron: шаг

Пусть на шаге t имеем $w^{(t)}$ и объект x_i, который классифицируется неверно:

$$\mathrm{sign}(\langle w^{(t)}, x_i \rangle) \neq y_i$$

Положим:

$$w^{(t+1)} = w^{(t)} + y_i x_i$$

Алгоритм Perceptron: обоснование шага

Шаг алгоритма:

$$w^{(t+1)} = w^{(t)} + y_i x_i, \text{ при чём } \operatorname{sign}(\langle w^{(t)}, x_i \rangle) \neq y_i$$

Мы хотим, чтоб $\forall i = 1, \ldots, m$:

$$y_i \langle w, x_i \rangle > 0$$

Распишем:

$$y_i \langle w^{(t+1)}, x_i \rangle = y_i \langle w^{(t)} + y_i x_i, x_i \rangle = y_i \langle w^{(t)}, x_i \rangle + ||x_i||^2$$

Алгоритм Perceptron

Алгоритм 1 Batch perceptron

Вход: **Разделимая** тренировочная выборка $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$

Выход: w, такой что $y_i\langle w, x_i\rangle > 0\ \forall i = 1, \ldots, m$

1: $w^{(1)} = (0, \ldots, 0)$
2: **for** $t = 1, 2, \ldots$ **do**
3:     **if** $\exists i$, т.ч. $y_i\langle w^{(t)}, x_i\rangle \leqslant 0$ **then**
4:         $w^{(t+1)} = w^{(t)} + y_i x_i$
5:     **else**
6:         **return** $w^{(t)}$
7:     **end if**
8: **end for**

Теорема о сходимости алгоритма Perceptron

Теорема о сходимости алгоритма Perceptron

Пусть $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ — **линейно разделимая** выборка. Обозначим $B = \min\{||w|| : y_i\langle w, x_i\rangle \geqslant 1\ \forall i \in [m]\}$, $R = \max_i ||x_i||$. Тогда алгоритм 1 завершает свою работу не более, чем через $(RB)^2$ итераций, и итоговый вектор w удовлетворяет условию $y_i\langle w, x_i\rangle > 0$ для всех объектов выборки.

Доказательство теоремы

- докажем, что если алгоритм остановился на итерации T, то $T \leqslant (RB)^2$
- пусть w^* — вектор, доставляющий минимум в определении $B = \min\{||w|| : y_i\langle w, x_i\rangle \geqslant 1 \ \forall i \in [m]\}$
- докажем,

$$\frac{\langle w^*, w^{(t+1)}\rangle}{||w^*||||w^{(t+1)}||} \geqslant \frac{\sqrt{t}}{RB}$$

Доказательство теоремы

Хотим:

$$\frac{\langle w^*, w^{(t+1)}\rangle}{\|w^*\|\|w^{(t+1)}\|} \geqslant \frac{\sqrt{t}}{RB}$$

Имеем:

- $\|w^*\| = B$

Докажем:

- $\langle w^*, w^{(t+1)}\rangle \geqslant t$
- $\|w^{(t+1)}\| \leqslant \sqrt{t}R$

Доказательство теоремы

Хотим:

$$\langle w^*, w^{(t+1)} \rangle \geqslant t$$

Имеем: w^* — вектор, доставляющий минимум в определении $B = \min\{||w|| : y_i \langle w, x_i \rangle \geqslant 1 \ \forall i \in [m]\}$

Доказательство:

- $w^{(1)} = (0, \dots, 0) \Rightarrow \langle w^*, w^{(t+1)} \rangle = 0 \geqslant 0$
- пусть на итерации t мы выбрали (x_i, y_i):

$$\langle w^*, w^{(t+1)} \rangle - \langle w^*, w^{(t)} \rangle = \langle w^*, w^{(t+1)} - w^{(t)} \rangle$$
$$\langle w^*, w^{(t+1)} - w^{(t)} \rangle = \langle w^*, y_i x_i \rangle = y_i \langle w^*, x_i \rangle \geqslant 1$$

Доказательство теоремы

Хотим:

$$\|w^{(t+1)}\| \leqslant \sqrt{t}R$$

Имеем:

$$R = \max_i ||x_i||$$

Доказательство:

$$\|w^{(t+1)}\|^2 = \|w^{(t)} + y_i x_i\|^2 = \|w^{(t)}\| + 2y_i\langle w^{(t)}, x_i\rangle + y_i^2\|x_i\|^2$$
$$2y_i\langle w^{(t)}, x_i\rangle < 0 \text{ (по построению алгоритма)}$$
$$\|w^{(t+1)}\|^2 \leqslant \|w^{(t)}\|^2 + R^2$$
$$\|w^{(1)}\|^2 = 0 \Rightarrow \|w^{(t+1)}\|^2 \leqslant tR^2$$

Доказательство алгоритма

Имеем:

- $\langle w^*, w^{(t+1)} \rangle \geqslant t$
- $\|w^{(t+1)}\| \leqslant \sqrt{t}R$
- $\|w^*\| = B$

Таким образом:

$$\frac{\langle w^*, w^{(t+1)} \rangle}{\|w^*\| \|w^{(t+1)}\|} \geqslant \frac{\sqrt{t}}{RB}$$

Алгоритм 1 завершает свою работу не более, чем за $(RB)^2$ итераций

- алгоритм прост в реализации
- гарантировано находит решение
- B может быть экспоненциально большим относительно d
- на практике B невелико

Итоги

- в случае разделимой выборки решать задачу просто, в случае неразделимой — сложно
- что делать, если выборка неразделима?
- решений несколько, какое выбрать?

Содержание

Задачи восстановления регрессии

- задачи машинного обучения, где $Y = \mathbb{R}$ традиционно называют **задачами восстановления регрессии**[2]
- функция потерь должна отличаться от 0-1 loss
- зачастую разумным выбором является квадратичная функция потерь $L_S(h) = \sum_{i=1}^{m} (h(x_i) - y_i)^2$
- обучение с такой функцией потерь называется **метод наименьших квадратов (least squares)**

[2] Francis Galton «Regression towards mediocrity in heredity stature»

Метод наименьших квадратов

- Рассмотрим $H = \{h_\alpha(x)\}$, где $\alpha \in \mathbb{R}^d$ — **параметры модели**
- $\mathrm{RSS}_S(h) = \sum_{i=1}^{m} w_i (h_\alpha(x_i) - y_i)^2$, где w_i — степень важности объекта; такая функция потерь называется **остаточная сумма квадратов (residual sum of squares)**
- $\mathrm{MSE}_S(h) = \frac{1}{m} \sum_{i=1}^{m} (h_\alpha(x_i) - y_i)^2$ — **mean square error**
- обучение по правилу $A(S) = h_{\alpha^*}$, где $\alpha^* = \mathrm{argmin}_\alpha \, \mathrm{RSS}_S(h_\alpha)$ называется **методом наименьших квадратов (least sqaures)**

МНК: минимизация

Хотим минимизировать $\sum_{i=1}^{m} w_i(h_\alpha(x_i) - y_i)^2$.

Если h достаточно гладкая, то можно решить:

$$\frac{\partial \, \mathrm{RSS}_S}{\partial \alpha} = 2 \sum_{i=1}^{m} w_i \left(h_\alpha(x_i) - y_i\right) \frac{\partial h_\alpha}{\partial \alpha}(x_i) = 0$$

Метод максимального правдоподобия

Метод максимального правдоподобия (Maximum likelihood estimation)

Пусть дана выборка $(x_1, \dots, x_m)$ из распределения вероятности P_α, где α — неизвестный вектор параметров. Тогда оценка $\alpha^* = \operatorname*{argmax}_\alpha P((x_1, \dots, x_m)|\alpha)$ называется **оценкой максимального правдоподобия**

- состоятельная
- асимптотически эффективная
- асимптотически нормальная

МНК = ММП

Теорема о ММП оценке в случае гауссовского шума

Пусть разметочная функция f имеет вид:

$$f(x_i) = h_\alpha(x_i) + \epsilon_i$$

где ϵ_i — независимые нормальные случайные величины с нулевым средним и дисперсией σ_i^2. Тогда МНК-решение и ММП-оценка для α совпадает, в случае, если веса объектов w_i обратно пропорциональны дисперсии шума σ_i^2

Доказательство

Запишем функцию правдоподобия:

$$P(\epsilon_1, \dots, \epsilon_m | \alpha) = \prod_{i=1}^{m} \frac{1}{\sigma_i \sqrt{2\pi}} exp\left(-\frac{1}{2\sigma_i^2}(h_\alpha(x_i) - y_i)^2\right)$$

Прологарифмируем:

$$\ln P(\epsilon_1, \dots, \epsilon_m | \alpha) = \mathrm{const}(a) - \frac{1}{2} \sum_{i=1}^{m} \frac{1}{\sigma_i^2} (h_\alpha(x_i) - y_i)^2$$

Итоги

- w_i можно рассматривать как меру точности измерения признака i
- если предположение теоремы выполнено, то МНК-оценка имеет высокое качество
- если не выполнено, то сколько угодно плохое

Многомерная линейная регрессия

- пусть $X = \mathbb{R}^d$ и H — линейные функции: $h_w(x) = \langle w, x \rangle$
- $\mathrm{MSE}_S(h) = \dfrac{1}{m}\sum_{i=1}^{m}(\langle w, x_i \rangle - y_i)^2$
- МНК оценка выражается из уравнения:
  $\dfrac{2}{m}\sum_{i=1}^{m}(\langle w, x_i \rangle - y_i)x_i = 0$
  Или в матричном виде

Матричная постановка задачи

Имеем: $\frac{2}{m}\sum_{i=1}^{m}(\langle w, x_i\rangle - y_i)x_i = 0$

Положим: $A = \left(\sum_{i=1}^{m} x_i x_i^T\right)$, $b = \sum_{i=1}^{m} y_i x_i$, решаем $Aw = b$

$$A = XX^T = \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix} \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix}^T$$

$$b = Xy = \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

Полиномиальная регрессия

- пусть $X = \mathbb{R}$, положим $x_i' = (1, x_i, x_i^2, \ldots, x_i^d)$
- полученная модель называется моделью **полиномиальной регрессии**
- X^T в этом случае называется

Полиномиальная регрессия

- пусть $X = \mathbb{R}$, положим $x_i' = (1, x_i, x_i^2, \ldots, x_i^d)$
- полученная модель называется моделью **полиномиальной регрессии**
- X^T в этом случае называется матрицей Вандермонда

Криволинейная регрессия

- пусть $X = \mathbb{R}^d$, т.е. $x_i = (x_i^{(1)}, \ldots, x_i^{(d)})$
- пусть $X' = \mathbb{R}^{d'}$, $x_i' = (\phi_1(x_i), \ldots, \phi_{d'}(x_i))$
- линейная модель на пространством X' называется моделью **криволинейной регрессии**
- X'^T — обобщённая матрица Вандермонда

Нормальная система уравнений

Имеем:

$$A = XX^T = \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix} \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix}^T$$

$$b = Xy = \begin{pmatrix} \vdots & & \vdots \\ x_1 & \dots & x_m \\ \vdots & & \vdots \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$

Хотим решить задачу:

$$XX^T w = Xy$$

Нормальная система уравнений

Хотим решить задачу:

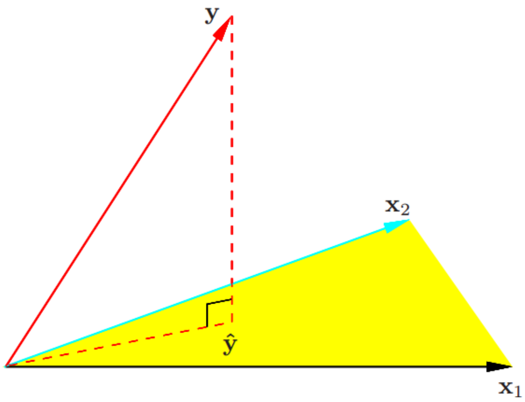
$$XX^T w = Xy$$

Если XX^T невырождена, то: $w = (XX^T)^{-1}Xy$
Матрица $(XX^T)^{-1}X$ называется **псевдообратной** для прямоугольной матрицы X^T.

$$\mathrm{MSE}_S(h_w) = \|X(XX^T)^{-1}Xy - y\|^2 = \|P_X y - y\|^2$$

где, $P_X = X^T(XX^T)^{-1}X$ — проекционная матрица на пространство строк матрицы X

Геометрическая интерпретация



Сингулярное разложение

SVD-разложение (singular value decomposition)

Произвольную матрицу X размера $d \times m$ можно разложить в произведение трёх матриц:

$$X = UDV^T$$

- $U = (u_1, \ldots, u_d)$ имеет размеры $d \times d$ $(U^T U = I)$, столбцы — собственные векторы XX^T
- D — диагональная размера $d \times m$, на главной диагонали корни собственных чисел XX^T, $X^T X$; число ненулевых таких значений равно рангу X, $X^T X$, XX^T (в случае $\mathbb{R}$)
- $V = (v_1, \ldots, v_m)$ имеет размер $m \times m$, $(V^T V = I)$; столбцы — собственные векторы $X^T X$.

Решение МНК через SVD

Решаем:

$$XX^T w = Xy$$

Имеем:

$$w = (XX^T)^{-1}Xy$$
$$(X^T)^+ = (XX^T)^{-1}X \text{ — псевдообратная для } X^T$$

SVD для X^T

$$X = UDV^T \Rightarrow X^T = VD^TU^T$$

Значит:

$$(X^T)^+ = (UDV^TVD^TU^T)^{-1}UDV^T$$

Замечание про D

$$D = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \end{pmatrix} \quad D^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad DD^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

$$D^+ = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (D^T)^+ = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 \end{pmatrix}$$

Псевдообратная матрица через SVD

$$(X^T)^+ = (UDV^TVD^TU^T)^{-1}UDV^T \tag{1}$$

$$(X^T)^+ = (U(DD^T)U^T)^{-1}UDV^T \tag{2}$$

$$(X^T)^+ = (U^T)^{-1}(DD^T)^{-1}U^{-1}UDV^T \tag{3}$$

$$(X^T)^+ = U(DD^T)^{-1}U^TUDV^T \tag{4}$$

$$(X^T)^+ = U(DD^T)^{-1}DV^T \tag{5}$$

$$(X^T)^+ = U(D^T)^+V^T \tag{6}$$

МНК-решение через SVD

Решение, через SVD:

$$w = (XX^T)^{-1}Xy = U(D^T)^+V^Ty = \sum_{i=1}^{d} \frac{1}{\sqrt{\lambda_i}} u_i(v_i^T y)$$

u_i — столбцы матрицы U v_i^T — строки матрицы V^T

Проекционный оператор через SVD:

$$P_X y = X^T(XX^T)^{-1}Xy = VD^TU^TU(D^T)^+V^T = \sum_{i=1}^{d} v_i(v_i^T y)$$

Норма вектора:

$$\|w\|^2 = \|(D^T)^+V^Ty\|^2 = \sum_{i=1}^{d} \frac{1}{\lambda_i}(v_i^T y)^2$$

Стандартизация данных

- разные признаки в разных масштабах (единицах измерения)
- проводят **стандартизацию** данных: $f_i = (f_i - \overline{f_i})/\sigma_i$
- такое же преобразование при применении

Проблема мультиколлинеарности

- если XX^T неполного ранга, её нельзя обратить
- на практике, чаще ранг полный, но матрица близка к матрице неполного ранга — проблема мультиколлинеарности (матрица *плохо обусловлена*, *неполоного псевдоранга*)
- случается, если объекты сконцентрированы около пространства меньшей размерности, чем d
- малые собственные значения

Проблемы с мультиколлинеарностью

- повышается разброс коэффициентов (см. формулу для w)
- повышается неустойчивость решения
- понижается обобщающая способность алгоритма

Методы борьбы с проблемой

- регуляризация
- преобразование признаков
- отбор признаков

Гребневая регрессия (ridge regression)

Изменим функцию потерь:

$$L_S(h_w) = \frac{1}{m}\sum_{i=1}^{m}(\langle w, x_i\rangle - y_i)^2 + \tau\|w\|^2$$

Тогда решение:

$$w = (XX^T + \tau I)^{-1}Xy$$

Через SVD:

$$\|w\|^2 = \|(D^T)^+ V^T y\|^2 = \sum_{i=1}^{d}\frac{1}{\lambda_i+\tau}(v_i^T y)^2$$

Происходит сжатие весов (weight decay)

Выбор τ

- слишком большой и слишком маленький параметр τ — плохо
- вопрос: как выбирать
- ответ: см. следующие лекции
- ответ 2: пытаются выбрать τ так, что получить обусловленность матрицы в заранее заданном диапазоне

Лассо Тибширани

$$\begin{cases} L_S(h_w) = \dfrac{1}{m} \sum\limits_{i=1}^{m} (\langle w, x_i \rangle - y_i)^2 \to \min \\ \sum\limits_{i=1}^{d} |w_i| \leqslant \psi \end{cases}$$

- $w_i = w_i^+ - w_i^-$
- $w_i^+ \geqslant 0$
- $w_i^- \geqslant 0$
- $\sum\limits_{i=1}^{d} w_i^+ + w_i^- \leqslant \psi$

Least absolute shrinkage and selection operator (LASSO)

Итоги

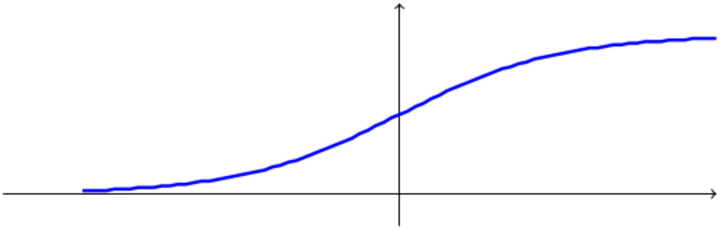
- МНК-оценка лучшая, если шумы гауссовы
- Метод хорошо изучен, допускает много обобщений
- Оценка неустойчива к шумным выбросам
- Для борьбы с мультиколлинеарностью зачастую используют ridge regression, иногда LASSO

Содержание

Логистическая регрессия

- вернёмся к задаче классификации
- эмпирический риск:
  $$L_S(h) = \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m} = \frac{|\{i \in [m] : y_i\langle w, x_i\rangle < 0\}|}{m}$$
- классификация слишком жёсткая
- будем считать, что $\phi(x) = \sigma(x) = \frac{1}{1 + exp(-x)}$, т.е.
  $$H_d = \{x \mapsto \sigma(\langle w, x\rangle + b) : w \in \mathbb{R}^d, b \in \mathbb{R}\}$$

Сигмоидальная функция



$$\sigma(x) = \frac{1}{1 + exp(-x)}$$

Экспоненциальные функции распределения

Экспоненциальные функции распределения

Плотность распределения $p(x)$, $x \in \mathbb{R}^d$ называется **экспоненциальной**, если:

$$p(x) = \exp\left(\theta^T x \cdot a(\delta) + b(\delta, \theta) + d(x, \delta)\right)$$

- $\theta \in \mathbb{R}^d$ — **сдвиг**
- δ — **разброс**
- a, b, d — произвольные числовые функции

Многие известные распределения экспоненциальны: равномерное, пуассоновское, биномиальное
Нормальное тоже является с $\theta = \Sigma^{-1}\mu$, $\delta = \Sigma$

Оптимальный байесовский классификатор

Оптимальный байесовский классификатор

Классификатор:

$$f_D(x) = \operatorname*{argmax}_y D(x, y) = \operatorname*{argmax}_y P(y|x)$$

называется **оптимальным байесовским классификатором**

Обоснование логистической регрессии

Теорема об оптимальности логистической регрессии

Пусть

- среди признаков есть константа
- плотности правдоподобия классов $P(x|y)$ являются экспоненциальными с равными разбросами

Тогда

- байесовское правило линейно $a(x) = \text{sign}(\langle w, x \rangle)$
- $P(y|x) = \sigma(y\langle w, x \rangle)$

Доказательство

$$\frac{P(+1|x)}{P(-1|x)} = \frac{P(x|y=1)P(y=1)P(x)}{P(x|y=-1)P(y=-1)P(x)}$$

Распишем

$$\exp\left(\alpha(\delta)(\theta_+ - \theta_-)^T x + b(\delta,\theta_+) - b(\delta,\theta_-) + \ln\frac{P(y=1)}{P(y=-1)}\right)$$

А значит

$$\frac{P(+1|x)}{P(-1|x)} = \exp\left(\langle w, x\rangle\right)$$
$$P(+1|x) = \sigma(\langle w, x\rangle)$$
$$P(-1|x) = \sigma(-\langle w, x\rangle)$$

Решение для w

- как найти вектор w?
- функция логарифма правдоподобия
$$L(w) = \sum_{i=1}^{m} \ln \sigma(y_i \langle w x_i \rangle) + C \to \max$$
- полученная функция выпуклая, мы научимся находить максимум подобных функций через несколько лекций

Итоги

- регрессия может помочь решить задачу классификации
- при определённых предположениях решение логистической регрессии ведёт себя оптимально
- даёт возможность вычислять вероятность классов
- позволяет решать задачу скоринга

Содержание

Итоги

- рассмотрели семейство линейных моделей
- разобрали два ERM-алгоритма для нахождения линейной модели в случае разделимой выборки: сведение к задаче линейного программирования и алгоритм Perceptron
- рассмотрели задачу линейной регрессии и предложили методы для борьбы с проблемой мультиколлинеарности: гребневая регрессия и Лассо
- ввели понятие логистической регрессии

Литература

- Shai Shalev-Shwartz and Shai Ben-David — Understanding Machine Learning: From theory to algorithms (глава 9)
- К.В.Воронцов — Лекции по алгоритмам восстановления регрессии — http://www.machinelearning.ru/wiki/images/a/aa/Voron-ML-Regression.pdf