Московский физико-технический институт
Физтех-школа прикладной математики и информатики

ВВЕДЕНИЕ В АНАЛИЗ ДАННЫХ

IV СЕМЕСТР

Лектор:

h\nu

Автор: *Киселев Николай*
Репозиторий на Github

весна 2025

Содержание

1 Задача линейной регрессии

1.1 Прямой подход

Будем рассматривать следующие модели:

$$\mathcal{M} = \{y : \mathbb{R}^d \to \mathbb{R} | y(x) = x^T\theta, \theta \in \mathbb{R}^d\}$$

Наша цель — получить наилучшую модель, то есть оценить θ.
Пусть $\hat{\theta}$ — оценка θ. Тогда $\hat{y}(x) = x^T\hat{\theta}$ — предсказание для x.
Пусть $x_1, \dots x_n$ — объекты, $Y_1, \dots Y_n$ — таргеты. Пусть $\hat{Y}_i = \hat{y}(x_i)$
Введем функционал ошибки:

$$\mathcal{L}(y, z) = (y - z)^2$$

И тогда получаем, что

$$F(\theta) = \sum_{i=1}^{n} \mathcal{L}(Y_i, \hat{Y}_i) = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n} \left(Y_i - x_i^T\hat{\theta}\right) = \|Y - X\hat{\theta}\|^2$$

Где

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, X = \begin{pmatrix} x_{11} & \dots & x_{1d} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nd} \end{pmatrix}$$

Мы хотим минимизировать $F(\theta)$

Утверждение 1.1. *Если XX^T не вырождена, то* $\hat{\theta} = (X^TX)^{-1}X^TY$

Доказательство.

$$F(\theta) = \|Y - X\theta\|^2 = (Y - X\theta)^T(Y - X\theta) = Y^TY - 2Y^TX\theta + \theta^TX^TX\theta$$

Посчитаем градиент данной функции по θ:

$$\nabla F(\theta) = -2X^TY + 2X^TX\theta = 0$$

Домножим на $(X^TX)^{-1}$ слева:

$$\hat{\theta} = \underbrace{(X^TX)^{-1}X^T}_{\text{псевдообратная матрица}} Y$$

Т.к. функция $\|Y - X\theta\|^2$ квадратична, точка с нулевым градиентом является точкой минимума. □

1.2 Градиентный спуск (GD)

Пусть у нас есть задача $f(x) \to \min_x$.

Замечание. ∇f — направление наибольшего роста $f(x)$ в точке x.

Идея: будем идти в противоположную сторону. Пусть x_0 — начальное приближение. Будем действовать по следующему алгоритму: будем постепенно делать шаги, каждый новый шаг определяется формулой $x_{t+1} = x_t - \eta\nabla f$, где η — шаг метода.

Пример. Рассмотрим $f(x) = x^2 \Rightarrow \nabla f = 2x$. GD даст нам $x_{t+1} = x_t - 2\eta x_t$. При $\eta = 1$ мы получим $x_{t+1} = -x_t$.

Пример. Рассмотрим $f(x) = x^4 \Rightarrow \nabla f = 4x^3$. GD даст нам $x_{t+1} = x_t - 2\eta x_t$. При $\eta = 1$ мы получим $x_{t+1} = -4x_t$, т.е. наша последовательность не сходится.

Применим GD к задаче линейной регрессии: $F(\theta) = \|Y - X\theta\|^2 \to \min_\theta$

$$\nabla F(\theta) = -2X^TY + 2X^TX\theta = 2X^T(X\theta - Y)$$

Шаг GD (занесем константу 2 в η):

$$\theta_{t+1} = \theta_t - \eta X^T(X\theta_t - Y) = \theta_t - \eta\sum_{i=1}^{n} x_i(x_i^T\theta_t - Y_i)$$

Что мы можем сказать про градиентный спуск?

\+ Не надо обращать матрицу.

\- Если n велико, то каждый шаг выполняется долго

Возникает еще одна идея: а что если считать градиент не для каждого из n элементов, а для некоторого количества из них. Таким образом, мы приходим к идее стохастического градиентого спуска.

1.3 Стохастический градиентный спуск (SGD)

Возьмем индексы $I = \underbrace{\{i_1, \dots i_k\}}_{\text{батч}} \sim U\{1, 2, \dots n\}$ (отвечающие равномерному распределению), где k — размер батча. Тогда шаг стохастического градиентого спуска будет определяться по формуле:

$$\theta_{t+1} = \theta_t - \eta\frac{n}{k}\sum_{i\in I} x_i(x_i^T\theta_t - Y_i)$$

Здесь множитель $\frac{n}{k}$ добавлен для нормировки: т.к. мы взяли k объектов из n, то полученный градиент будет приблизительно в $\frac{n}{k}$ меньше исходного. Рассмотрим данные операции в матричном виде:

Пусть X_I — матрица из строк матрицы X с индексами $i_1, \dots i_k$, Y_I — вектор из элементов вектора Y с индексами $i_1, \dots i_k$. Таким образом, шаг SGD будет иметь следующий вид:

$$\theta_{t+1} = \theta_t - \eta\frac{n}{k}X_I^T(X_I\theta_t - Y_I)$$

Итого, процедура имеет следующий вид:

1. Сгенерировать набор I
2. Вычислить $\theta_{t+1} = \dots$

2 Линейные модели классификации

2.1 Случай двух классов

Пусть $x_1, \dots x_n \in \mathbb{R}^d$ — признаки, $y_1, \dots y_n \in \{0, 1\}$ — признаки. Мы предполагаем, что:

$$y_i = y_*(x_i, \varepsilon_i)$$

Где ε_i — неизвестные случайные величины.

Пример. $y_i = I\{x_i^T \theta_* + \varepsilon_i > 0\}$, где θ_* — неизвестна.

Мы будем предсказывать $P(y_i = 1) = \mu(x_i)$.

Замечание. Заметим, что $y_i \sim Bern(\mu(x_i))$.

Таким образом, мы свели задачу к задаче нахождения $\mu : \mathbb{R}^d \to [0, 1]$.

2.2 Логистическая регрессия

Мы будем предполагать, что $\mu_\theta(x) = \sigma(\theta^T x)$, где $\sigma(x) = \frac{1}{1+e^{-x}}$ — логистическая сигмоида.

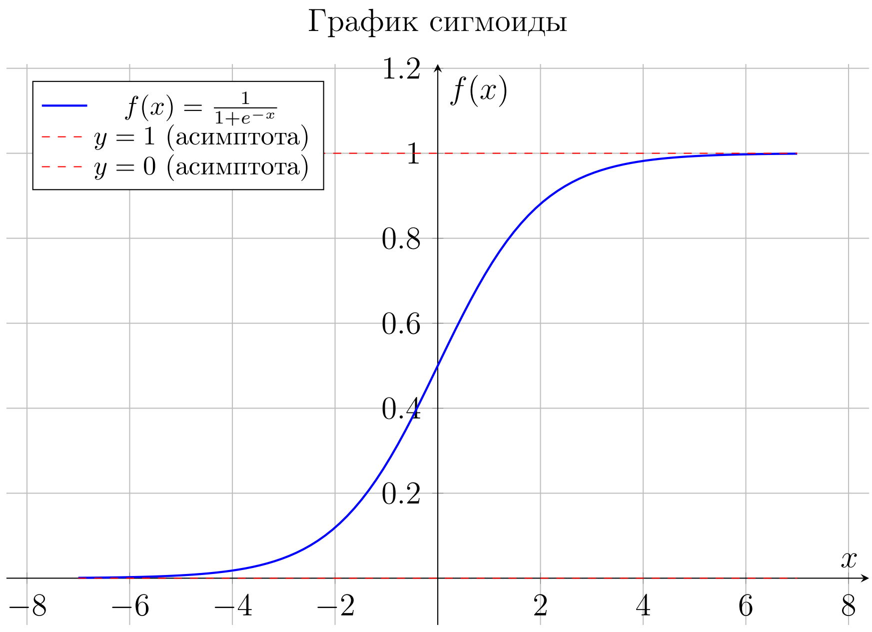

Рис. 1: Логистическая сигмоида

Замечание (Свойства $\sigma(z)$)**.** 1. $\sigma(-z) = 1 - \sigma(z)$

2. $\sigma'(z) = \sigma(z)(1 - \sigma(z))$

3. $\sigma^{-1}(s) = \ln \frac{s}{1-s}$ — логит-функция

Таким образом, если $s = P(y_i = 1)$, то $\frac{s}{1-s}$ — шанс, тогда $\ln \frac{s}{1-s}$ — логит-функция от шанса. Тогда наше предположение эквивалентно тому, что логит от шанса — линейная функция по x.

Соответственно, рассмотим множество моделей:

$$\mathcal{M} = \{\mu : \mathbb{R}^d \to [0, 1] \mid \mu(x) = \sigma(\theta^T x), \theta \in \mathbb{R}^d\}$$

2.3 Обучение логистической регрессии

Более подробно о кросс-энтропии и ее применении в теории кодирования — в презентации

Определение 2.1.

$$H(P, Q) = -\sum_{j=1}^{k} p_j \log_2 q_j$$

Для каждого объекта i рассмотрим кросс-энтропию $H(P_i, Q_i)$, где $Q_i = (1-\sigma(\theta^T x_i), \sigma(\theta^T x_i))$ — вероятность класса 0 и 1 соответственно. Это наше предполагаемое распределение для y_i. $P_i = (1 - y_i, y_i)$ — наблюдаемое распределение y_i. Будем минимизировать сумму кросс-энтропий по всем объектам, т.е.

$$F(\theta) = \sum_{i=1}^{n} H(P_i, Q_i) = -\sum_{i=1}^{n} (1 - y_i) \log_2(1 - \sigma(\theta^T x_i)) + y_i \log_2 \sigma(\theta^T x_i) \to \min_{\theta \in \mathbb{R}^d}$$

Для минимизации, посчитаем градиент:

$$\nabla F(\theta) = -\sum_{i=1}^{n} \left((1 - y_i) \frac{-\sigma(1 - \sigma)}{1 - \sigma} x_i + y_i \frac{\sigma(1 - \sigma)}{\sigma} x_i \right)$$

$$\nabla F(\theta) = -\sum_{i=1}^{n} \left((1 - y_i) \frac{-\sigma(1 - \sigma)}{1 - \sigma} x_i + y_i \frac{\sigma(1 - \sigma)}{\sigma} x_i \right) = -\sum_{i=1}^{n} (y_i - \sigma(\theta^T x_i)) x_i = X^T (S(\theta) - Y)$$

Где $S(\theta) = \begin{pmatrix} \sigma(\theta^T x_1) \\ \vdots \\ \sigma(\theta^T x_n) \end{pmatrix}$

Таким образом, формула для градиентного спуска будет иметь следующий вид:

Градиентный спуск (GD):

$$\theta_{t+1} = \theta_t - \eta X^T (S(\theta) - Y)$$

Стохастический градиентный спуск (SGD):

$$\theta_{t+1} = \theta_t - \eta \frac{n}{k} X_I^T (S(\theta)_I - Y_I)$$

Где $I = \{i_1, \dots i_k\} \sim U\{1, \dots n\}$ — батчи отвечают равномерному распределению.

2.4 Многоклассовый случай

Замечание.

$$\sigma(z) = \frac{1}{1+e^{-z}} = \frac{e^{z/2}}{e^{-z/2}+e^{z/2}} = \frac{e^{z_1}}{e^{z_1}+e^{z_2}}$$

Таким образом, сигмоиду можно обобщить для k классов, положив:

$$\sigma(z_1, \dots z_k) = \left(\frac{e^{z_1}}{\sum_{i=1}^n e^{z_i}}, \dots \frac{e^{z_k}}{\sum_{i=1}^n e^{z_i}} \right)$$

В качестве $z_1, \dots z_k$ мы будем подставлять $z_i = x_i^T \theta_i$, как и в двуклассовом случае. Таким образом, наше предположение имеет вид:

$$P(y_i = j) = \frac{e^{z_j}}{\sum_{i=1}^n e^{z_i}}$$