

Лекция 4

Линейные модели

классификации

Юлия Конюшенко

ТГ: [@ko_iulia](https://t.me/ko_iulia)

koniushenko.iun@phystech.edu

Лекция 1.

ML

→ обучение с учителем

обучение без учителя

- категоризация
- снижение размерн.
- визуализация

- классифр.
- регрессия
- ранжирование

Лекция 2.

1) линейная регрессия

Обучение \equiv минимизация MSE

почему именно MSE? → есть вероятностная интерпретация

2) градиентного спуск

$$a(x) = (w, x)$$

$$MSE$$

$$w^{(k)} = w^{(k-1)} - \eta \triangleright Q(w^{(k-1)})$$

стochastic \rightarrow по 1 объекту
mini-batch \rightarrow по батчу

Лекция 3. Метрики качества и функционалы

$$MSE \rightarrow RMSE \rightarrow R^2$$

ошибки

$$MAE \rightarrow MSLE \rightarrow MAPE \rightarrow SMAPE$$

квантильная регрессия

онлайн / офлайн / бизнес метрики

Признаки переобучения, регуляризация

- разница качества на train/test
- большие веса

+ оптимумы MSE и

MAE

среднее медиана

$$\begin{aligned} \cdot L_2: & + \sum w_i^2 \\ \cdot L_1: & + \sum |w_i| \end{aligned}$$

ОЦЕНИВАНИЕ КАЧЕСТВА МОДЕЛИ

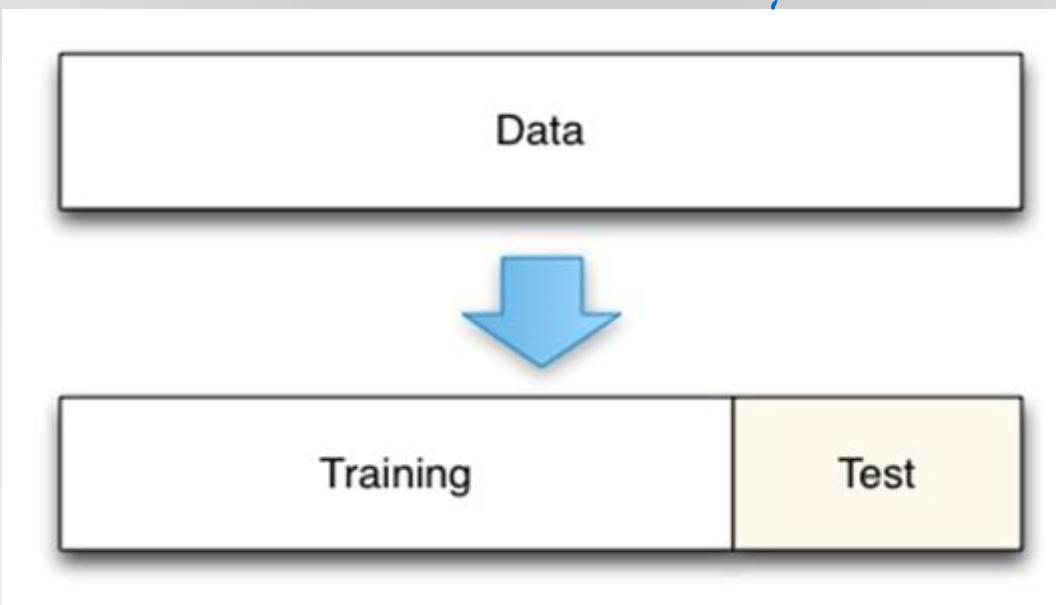
- Отложенная выборка
- Кросс-валидация

ОТЛОЖЕННАЯ ВЫБОРКА

Делим тренировочную выборку на две части:

- По первой части обучаем модель (*train*)
- По оставшимся данным – оцениваем качество (*test*)

Какие проблемы могут быть при такой оценке качества модели?



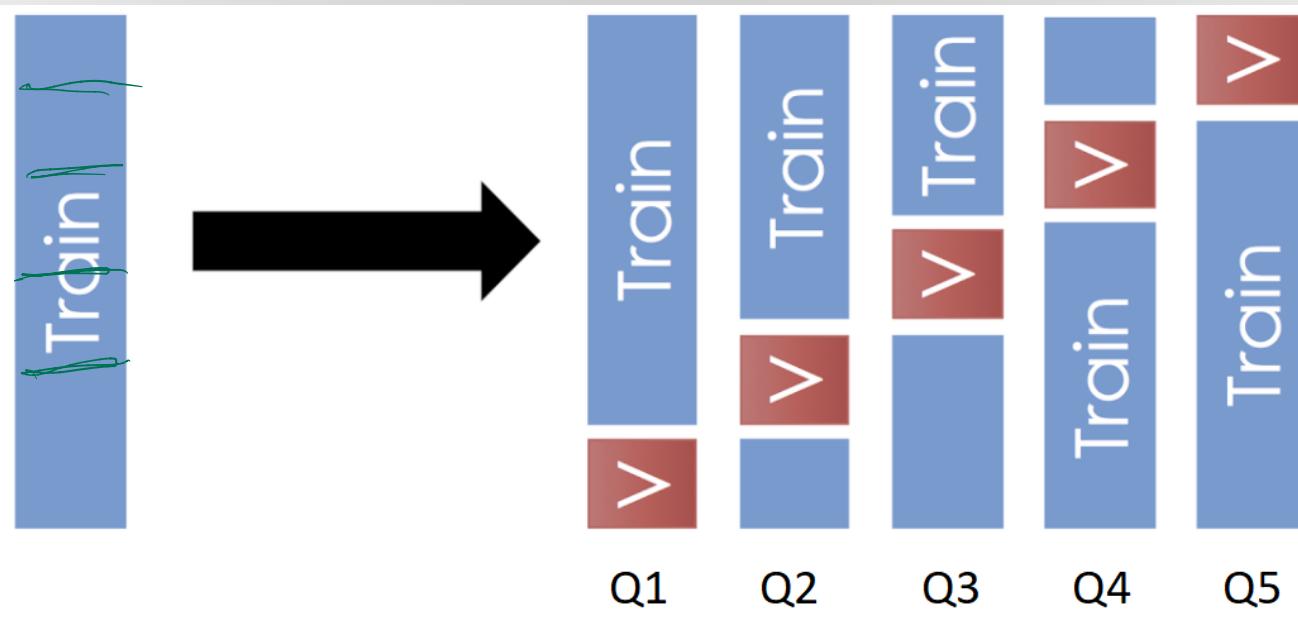
Недостаток:

- Результат сильно зависит от разбиения на *train* и *test*

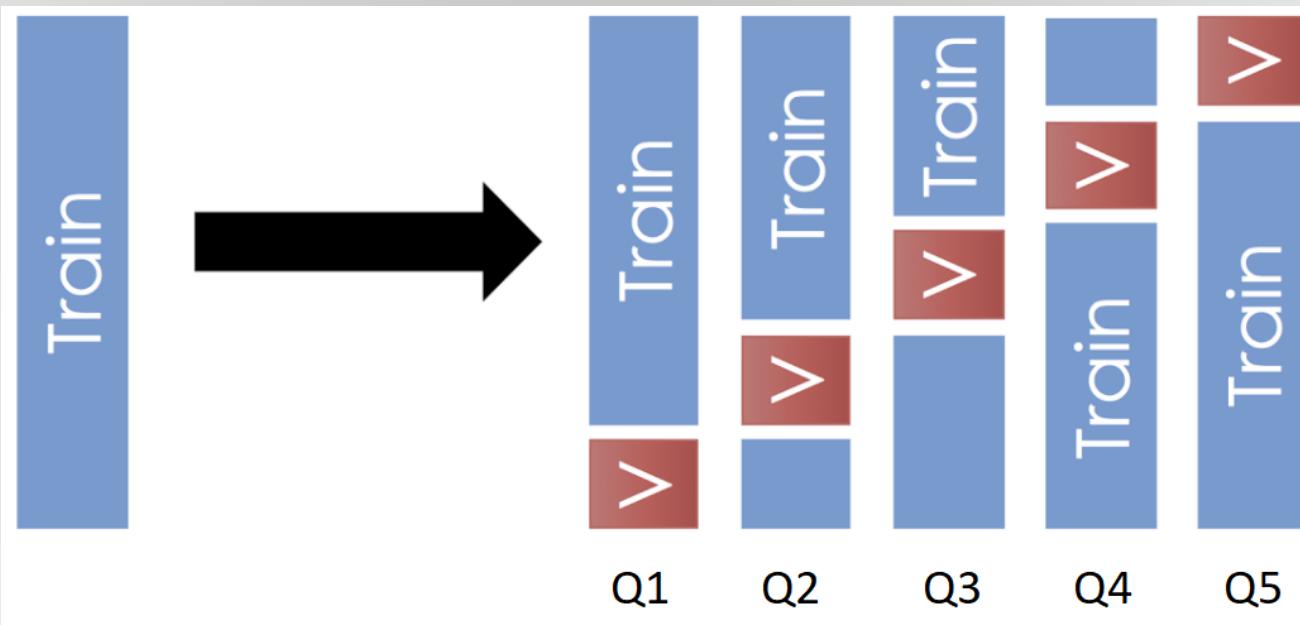
КРОСС-ВАЛИДАЦИЯ

$K = 5$

- Разбиваем объекты на тренировку (train) и валидацию (validation) несколько раз (при разбиении k раз получаем k-fold кросс-валидацию)
- Для каждого разбиения вычисляем качество на валидационной части
- Усредняем полученные результаты



КРОСС-ВАЛИДАЦИЯ



$$CV = \frac{1}{k} \sum_{i=1}^k Q(a_i(x), X_i) = \frac{1}{k} \sum_{i=1}^k Q_i$$

ВИДЫ КРОСС-ВАЛИДАЦИИ

- **k-fold cross-validation** – разбиваем данные на k блоков, каждый из которых по очереди становится контрольным (валидационным) $n = \frac{N}{k}$
- **Complete cross-validation** – перебираем ВСЕ разбиения
- **Leave-one-out cross-validation** – каждый блок состоит из одного объекта (число блоков = числу объектов) $n = 1$

какой из типов кросс валидации
даёт наилучшую оценку качества
модели?

ВЫБОР КОЛИЧЕСТВА БЛОКОВ В K-FOLD КРОСС-ВАЛИДАЦИИ



• Проблемы при маленьком k ?

• Проблемы при большом k ?

ВЫБОР КОЛИЧЕСТВА БЛОКОВ В K-FOLD КРОСС-ВАЛИДАЦИИ



- Маленькое k – оценка может быть пессимистично занижена из-за маленького размера тренировочной части
- Большое k – оценка может иметь большую дисперсию из-за маленького размера валидационной части

СПОСОБЫ КОДИРОВАНИЯ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ

КОДИРОВАНИЕ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ: ONE-HOT ENCODING

- Предположим, категориальный признак $f_j(x)$ принимает m различных значений: C_1, C_2, \dots, C_m .

Пример: еда может быть *горькой, сладкой, солёной или кислой* (4 возможных значения признака).

• какие способы знаете?

КОДИРОВАНИЕ КАТЕГОРИАЛЬНЫХ ПРИЗНАКОВ: ONE-HOT ENCODING

- Предположим, категориальный признак $f_j(x)$ принимает m различных значений: C_1, C_2, \dots, C_m .

Пример: еда может быть *горькой, сладкой, солёной или кислой* (4 возможных значения признака).

- Заменим категориальный признак на m бинарных признаков: $b_i(x) = [f_j(x) = C_i]$ (индикатор события).

Тогда One-Hot кодировка для нашего примера будет следующей:

горький = $(1, 0, 0, 0)$, *сладкий* = $(0, 1, 0, 0)$,

солёный = $(0, 0, 1, 0)$, *кислый* = $(0, 0, 0, 1)$.

какие проблемы у этого подхода?
1 → $(n - 1)$

- Col. → 1
- con → 2
- top. → 3
- Kue → 4

— на самом деле
нет определенного порядка
~~2 > 3~~

СЧЁТЧИКИ

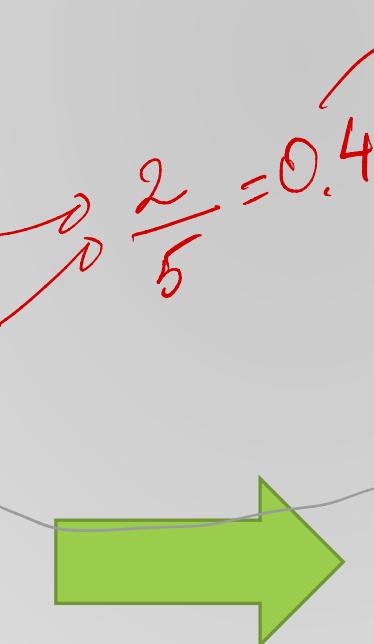
Счётчик (*mean target encoding*) – это вероятность получить значение целевой переменной для данного значения категориального признака.

СЧЁТЧИКИ (ПРИМЕР)

	feature	target
0	Moscow	0
1	Moscow	1
2	Moscow	1
3	Moscow	0
4	Moscow	0
5	Tver	1
6	Tver	1
7	Tver	1
8	Tver	0
9	Klin	0
10	Klin	0
11	Tver	1

СЧЁТЧИКИ (ПРИМЕР)

	feature	target
0	Moscow	0 5
1	Moscow	1 10
2	Moscow	1 15
3	Moscow	0 70
4	Moscow	0 23
5	Tver	1
6	Tver	1
7	Tver	1
8	Tver	0
9	Klin	0
10	Klin	0
11	Tver	1



	feature	feature_mean	target
0	Moscow	0.4	0
1	Moscow	0.4	1
2	Moscow	0.4	1
3	Moscow	0.4	0
4	Moscow	0.4	0
5	Tver	0.8	1
6	Tver	0.8	1
7	Tver	0.8	1
8	Tver	0.8	0
9	Klin	0.0	0
10	Klin	0.0	0
11	Tver	0.8	1

а это

будем делать в задаче регрессии? многоклассовый классификатор?

СЧЁТЧИКИ: ПРИМЕР

city	target	0	1	2
Moscow	1	1/4	1/2	1/4
London	0	1/2	0	1/2
London	2	1/2	0	1/2
Kiev	1	1/2	1/2	0
Moscow	1	1/4	1/2	1/4
Moscow	0	1/4	1/2	1/4
Kiev	0	1/2	1/2	0
Moscow	2	1/4	1/2	1/4

СЧЁТЧИКИ В ЗАДАЧЕ БИНАРНОЙ КЛАССИФИКАЦИИ

В случае бинарной классификации счётчики можно задать формулой:

$$Likelihood = \frac{Goods}{Goods + Bads} = mean(target),$$

где *Goods* – число единиц в столбце *target*,
Bads – число нулей в столбце *target*.

СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

- Пусть целевая переменная y принимает значения от 1 до K .
- Закодируем категориальную переменную $f(x)$ следующим способом:

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)} \approx p(y = k | f(x))$$

СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)}$$

Недостаток? Когда такой способ кодирования
переобучит наш алгоритм?

Вычисляя счётчики мы закладываем в них
информацию об отвёде!

Как будем бороться с этим?

СЧЁТЧИКИ (ОБЩАЯ ФОРМУЛА)

$$counts(u, X) = \sum_{(x,y) \in X} [f(x) = u]$$

$$successes_k(u, X) = \sum_{(x,y) \in X} [f(x) = u][y = k], k = 1, \dots, K$$

Тогда кодировка:

$$mean_target_k(x, X) = \frac{successes_k(f(x), X)}{counts(f(x), X)}$$

*Недостаток? Когда такой способ кодирования
переобучит наш алгоритм?*

Ответ: если в данных много редких категорий.

СЧЁТЧИКИ + СГЛАЖИВАНИЕ

Используем счётчики (mean target encoding) со
сглаживанием:

$$\frac{\text{mean}(\text{target}) \cdot n_{\text{rows}} + \text{global mean} \cdot \alpha}{n_{\text{rows}} + \alpha},$$

средний
по
всем
данным

n_{rows} - количество строк в категории,

α – параметр регуляризации.

если объектов какого-то класса мало, то
кодируем средним значением таргета по всей выборке



СЧЁТЧИКИ: КАК ВЫЧИСЛЯТЬ

- Можно вычислять счётчики так:

city	target	
Moscow	1	Вычисляем счетчики по этой части
London	0	
London	2	
Kiev	1	
Moscow	1	
Moscow	0	Кодируем признак вычисленными счётчиками и обучаемся по этой части
Kiev	0	
Moscow	2	

СЧЁТЧИКИ: КАК ВЫЧИСЛЯТЬ

Более продвинутый способ (по кросс-валидации):

1) Разбиваем выборку

на m частей X_1, \dots, X_m

2) На каждой части X_i

значения признаков

вычисляются по

оставшимся частям:

$$x \in X_i \Rightarrow g_k(x) = g_k(x, X \setminus X_i)$$



БОРЬБА С ПЕРЕОБУЧЕНИЕМ В СЧЁТЧИКАХ

- Вычисление счётчиков по кросс-валидации
- Сглаживание
- Добавление случайных шумов
- *Expanding mean*

<https://necromuralist.github.io/kaggle-competitions/posts/mean-encoding/#org01e0376>

ЛИНЕЙНЫЕ МОДЕЛИ КЛАССИФИКАЦИИ

ОБУЧЕНИЕ ЛИНЕЙНОЙ РЕГРЕССИИ (НАПОМИНАНИЕ)

Обучающая выборка:

пусть x – объект (x_1, x_2, \dots, x_l - его признаки), а y – ответ на объекте (произвольное число), n – количество объектов.

Модель линейной регрессии:

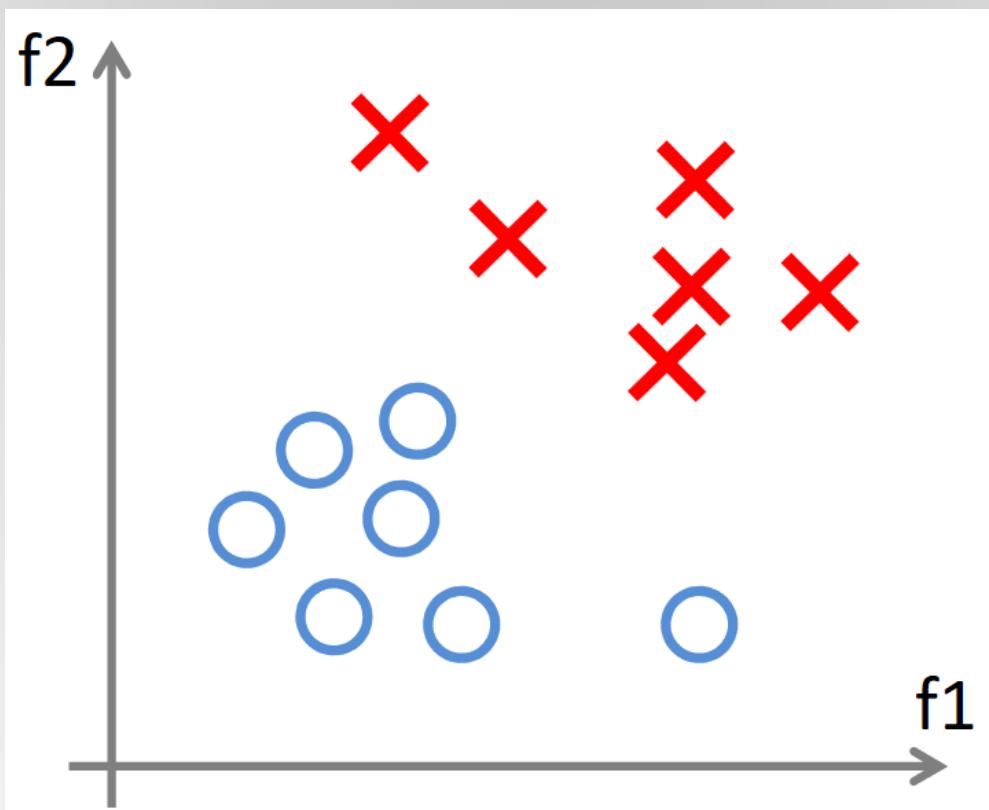
$$a(x, w) = \sum_{i=1}^l w_j x_j$$

- Метод обучения – метод наименьших квадратов
(минимизируем разность между предсказанием и правильным ответом):

$$Q(w) = \sum_{i=1}^n (a(x_i, w) - y_i)^2 \rightarrow \min_w$$

БИНАРНАЯ КЛАССИФИКАЦИЯ

y_1, y_2, \dots, y_n - ответы (+1 или -1).



Как выглядит модель линейного классификатора: $a(x, w) = ?$

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

$$a(x, w) = \textcolor{red}{sign}\left(\sum_{j=1}^l w_j x_j\right)$$

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

$$a(x, w) = \text{sign}\left(\sum_{j=1}^l w_j x_j\right)$$

- если $\sum_{j=1}^l w_j x_j > 0$, то $\text{sign}(\sum_{j=1}^l w_j x_j) = +1$, то есть объект отнесён к положительному классу
- если $\sum_{j=1}^l w_j x_j < 0$, то $\text{sign}(\sum_{j=1}^l w_j x_j) = -1$, то есть объект отнесён к отрицательному классу

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

$$a(x, w) = \text{sign} \left(\sum_{j=1}^l w_j x_j \right)$$

- если $\sum_{j=1}^l w_j x_j > 0$, то $\text{sign}(\sum_{j=1}^l w_j x_j) = +1$, то есть объект отнесён к положительному классу
- если $\sum_{j=1}^l w_j x_j < 0$, то $\text{sign}(\sum_{j=1}^l w_j x_j) = -1$, то есть объект отнесён к отрицательному классу
- значит, $\sum_{j=1}^l w_j x_j = 0$ – уравнение разделяющей границы между классами. Это уравнение плоскости (или прямой в двумерном случае), поэтому классификатор является линейным.

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

$$a(x, w) = \text{sign}\left(\sum_{j=1}^l w_j x_j\right)$$

надо знать:
- $a(x, w)$?
- $Q(w)$ - ?

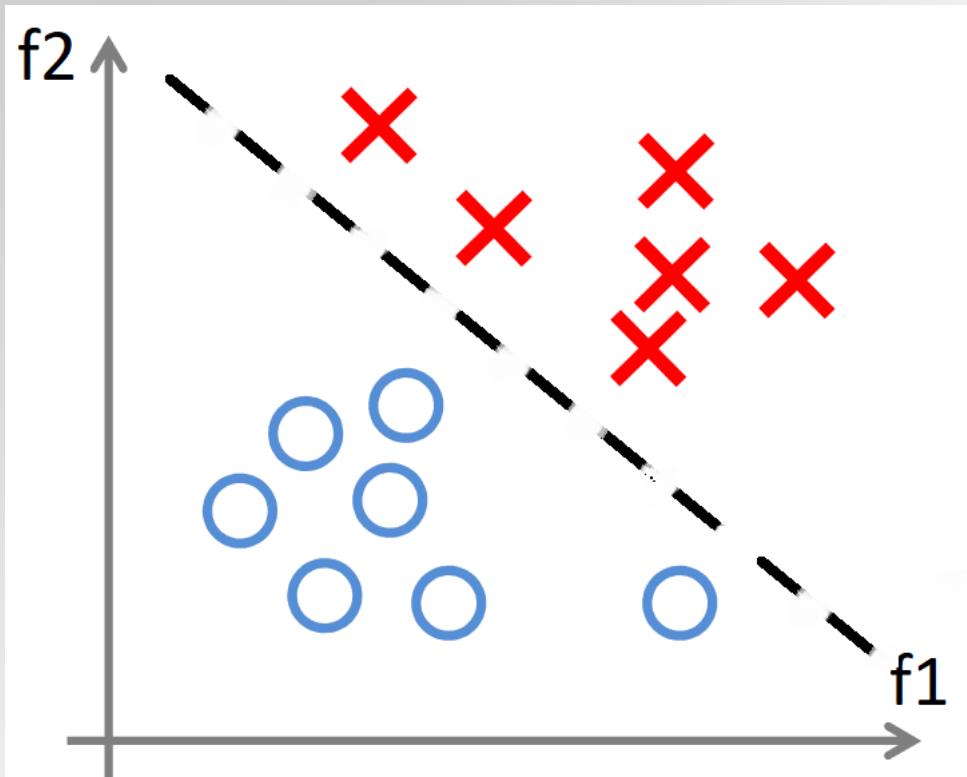
Уравнение

$$\sum_{j=1}^l w_j x_j = 0$$

– уравнение плоскости

(или прямой).

А как обугать?
Какая функция номер?



ОБУЧЕНИЕ КЛАССИФИКАТОРА

Как обучить линейный классификатор?

ОБУЧЕНИЕ КЛАССИФИКАТОРА

- Обучение - минимизация доли ошибок классификатора:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [a(x_i) \neq y_i] \rightarrow \min,$$

индикатор события

где $[a(x_i) \neq y_i] = 1$, если предсказание на объекте неверное, то есть $a(x_i) \neq y_i$, и 0 иначе.

$$0 + 0 + \dots + 0 + 1 + \dots \xrightarrow{n} \min$$

ОБУЧЕНИЕ КЛАССИФИКАТОРА

- Обучение - минимизация доли ошибок классификатора:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [a(x_i) \neq y_i] \rightarrow \min (*),$$

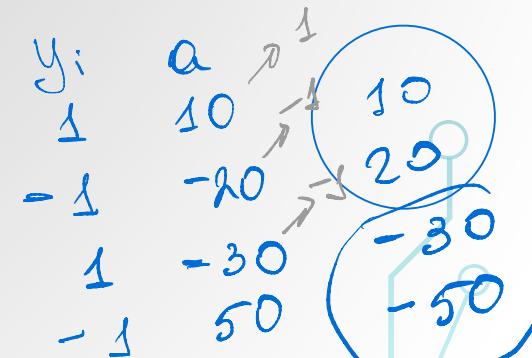
где $[a(x_i) \neq y_i] = 1$, если предсказание на объекте неверное, то есть $a(x_i) \neq y_i$, и 0 иначе.

- Обозначим $M_i = y_i \cdot (w, x_i)$ - **отступ** на i -м объекте.

$$1 \cdot 5 = 5$$

если угадали, то $M_i > 0$

если не угадали, то $M_i < 0$
будет ли $M_i = 0$? когда?



ОБУЧЕНИЕ КЛАССИФИКАТОРА

- Обучение - минимизация доли ошибок классификатора:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [a(x_i) \neq y_i] \rightarrow \min (*),$$

где $[a(x_i) \neq y_i] = 1$, если предсказание на объекте неверное, то есть $a(x_i) \neq y_i$, и 0 иначе.

- Обозначим $M_i = y_i \cdot (w, x_i)$ - **отступ** на i -м объекте.

Утверждение. Решение задачи (*) эквивалентно решению задачи

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

ДОКАЗАТЕЛЬСТВО УТВЕРЖДЕНИЯ

- Обучение - минимизация доли ошибок классификатора:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [a(x_i) \neq y_i] = \frac{1}{n} \sum_{i=1}^n [\text{sign}(w, x_i) \neq y_i] \rightarrow \min$$

Функционал Q можно переписать в виде:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [y_i \cdot (w, x_i) < 0] = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

- $M_i = y_i \cdot (w, x_i)$ - **отступ**

ОТСТУП (MARGIN)

Знак отступа $M = y \cdot (w, x)$ говорит о корректности классификации на объекте:

ОТСТУП (MARGIN)

Знак отступа $M = y \cdot (w, x)$ говорит о корректности классификации на объекте:

Случай неверной классификации (предсказание не совпадает с правильным ответом):

- Если $(w, x) > 0$ (то есть объект отнесён к классу +1), а $y = -1$, то $M = y \cdot (w, x) < 0$.

ОТСТУП (MARGIN)

Знак отступа $M = y \cdot (w, x)$ говорит о корректности классификации на объекте:

Случай неверной классификации (предсказание не совпадает с правильным ответом):

- Если $(w, x) > 0$ (то есть объект отнесён к классу +1), а $y = -1$, то $M = y \cdot (w, x) < 0$.
- Аналогично, если $(w, x) < 0$, а $y = +1$, то $M = y \cdot (w, x) < 0$.

ОТСТУП (MARGIN)

Знак отступа $M = y \cdot (w, x)$ говорит о корректности классификации на объекте:

Случай неверной классификации (предсказание не совпадает с правильным ответом):

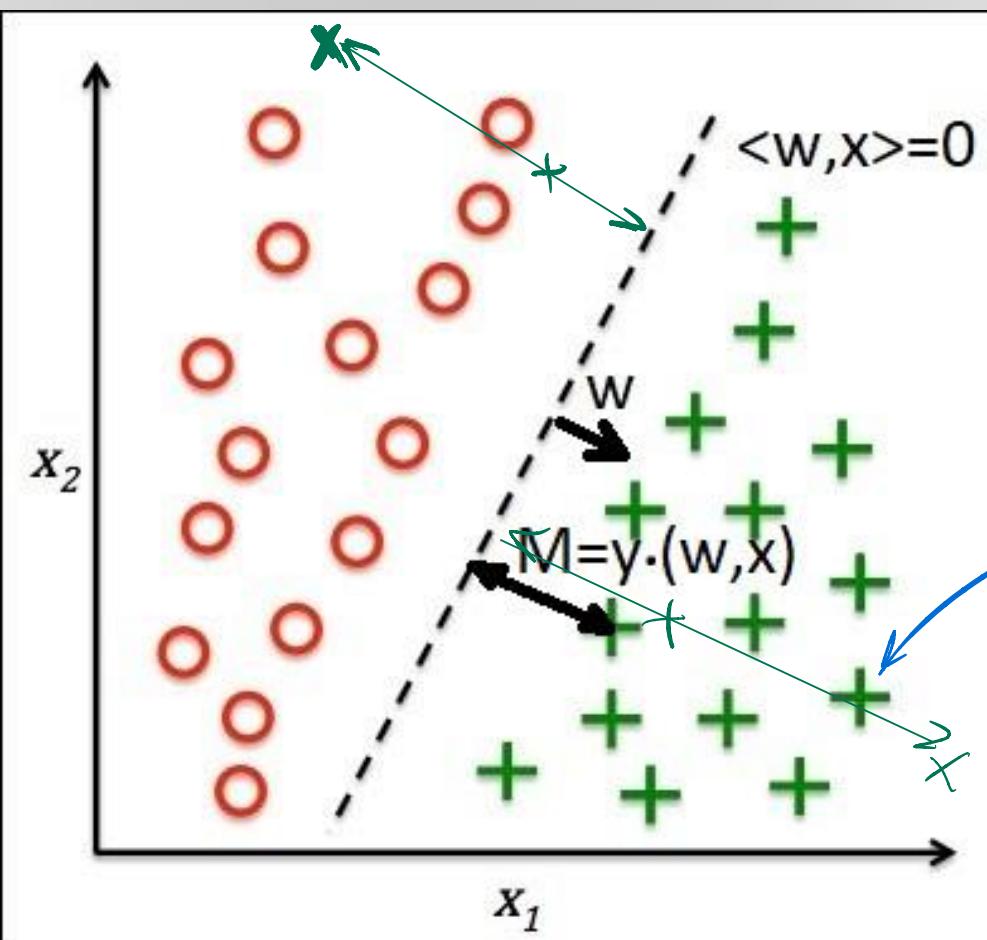
- Если $(w, x) > 0$ (то есть объект отнесён к классу +1), а $y = -1$, то $M = y \cdot (w, x) < 0$.
- Аналогично, если $(w, x) < 0$, а $y = +1$, то $M = y \cdot (w, x) < 0$.

Случай верной классификации:

- Если $(w, x) > 0$ и $y = +1$ или $(w, x) < 0$ и $y = -1$ получаем $M = y \cdot (w, x) > 0$.

ОТСТУП (MARGIN)

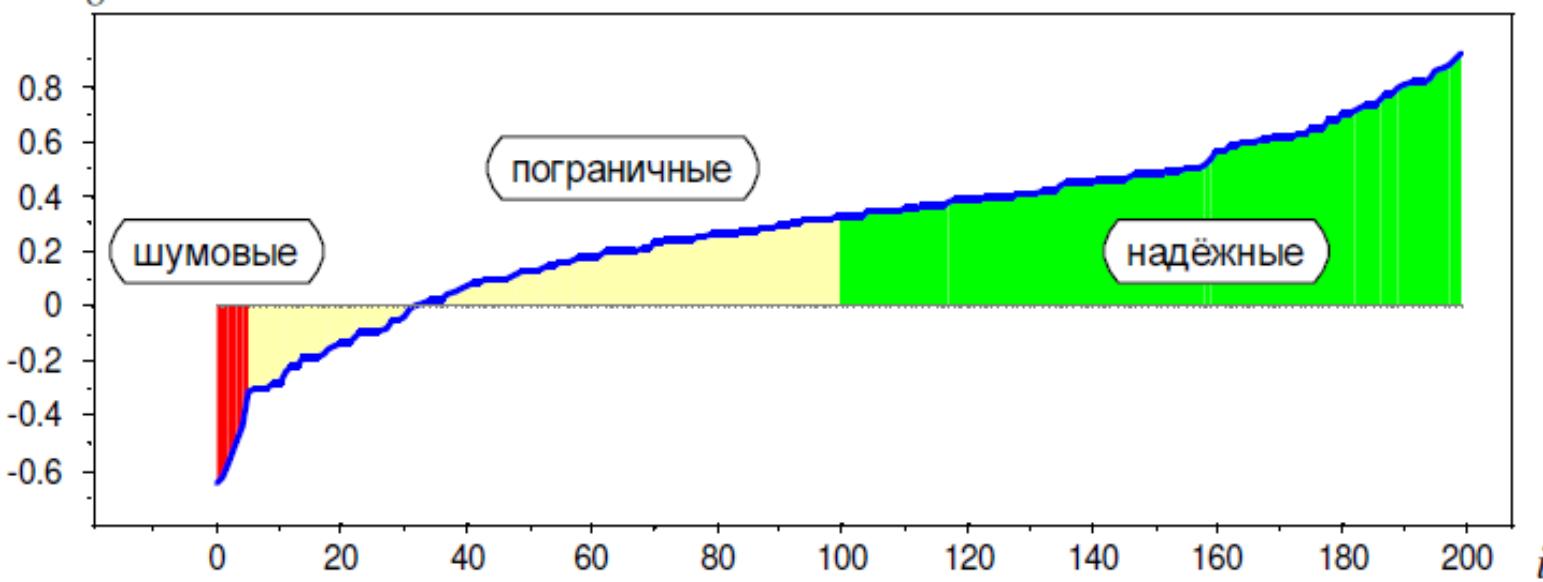
Абсолютная величина отступа M обозначает степень уверенности классификатора в ответе (чем ближе M к нулю, тем меньше уверенность в ответе)



ОТСТУП (MARGIN)

Ранжирование объектов по возрастанию отступа:

Margin



ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

Ранее мы показали, что обучение классификатора – это минимизация *пороговой функции потерь*:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

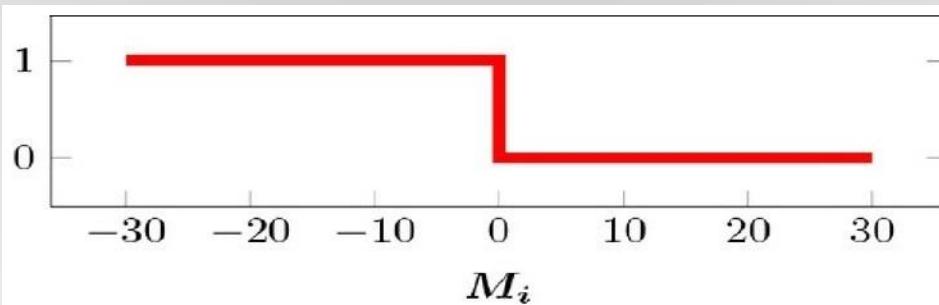
Как будем оптимизировать?

ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

Ранее мы показали, что обучение классификатора – это минимизация *пороговой функции потерь*:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

- Пороговая функция потерь *разрывна*, и этот факт сильно затрудняет процесс минимизации.

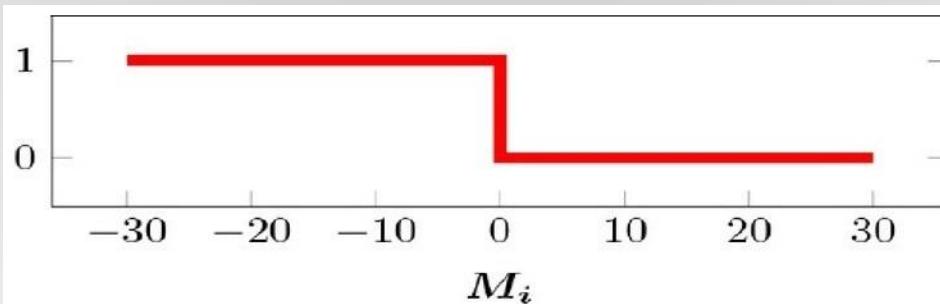


ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

Ранее мы показали, что обучение классификатора – это минимизация *пороговой функции потерь*:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

- Пороговая функция потерь разрывна, и этот факт сильно затрудняет процесс минимизации.



- Для решения этой проблемы используют *другие функции потерь – непрерывные или гладкие, как правило, являющиеся верхними оценками пороговой функции*.

ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

Ранее мы показали, что обучение классификатора – это минимизация *пороговой функции потерь*:

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n [M_i < 0] \rightarrow \min$$

- Пороговая функция потерь разрывна, и этот факт сильно затрудняет процесс минимизации.
- Для решения этой проблемы используют другие функции потерь – непрерывные или гладкие, как правило, являющиеся верхними оценками пороговой функции.
- Задача минимизации некоторой функции потерь называется *минимизацией эмпирического риска* (сама функция потерь – эмпирический риск).

ВЕРХНИЕ ОЦЕНКИ ЭМПИРИЧЕСКОГО РИСКА

- $L(a, y) = L(M) = [M < 0]$ – разрывная функция потерь

Оценим

$L(M) \leq \tilde{L}(M)$, где $\tilde{L}(M)$ - непрерывная или гладкая функция потерь.

- Тогда

$$Q(a, X) = \frac{1}{n} \sum_{i=1}^n L(y_i \cdot (w, x_i)) \leq \frac{1}{n} \sum_{i=1}^n \tilde{L}(y_i \cdot (w, x_i)) \rightarrow \min$$

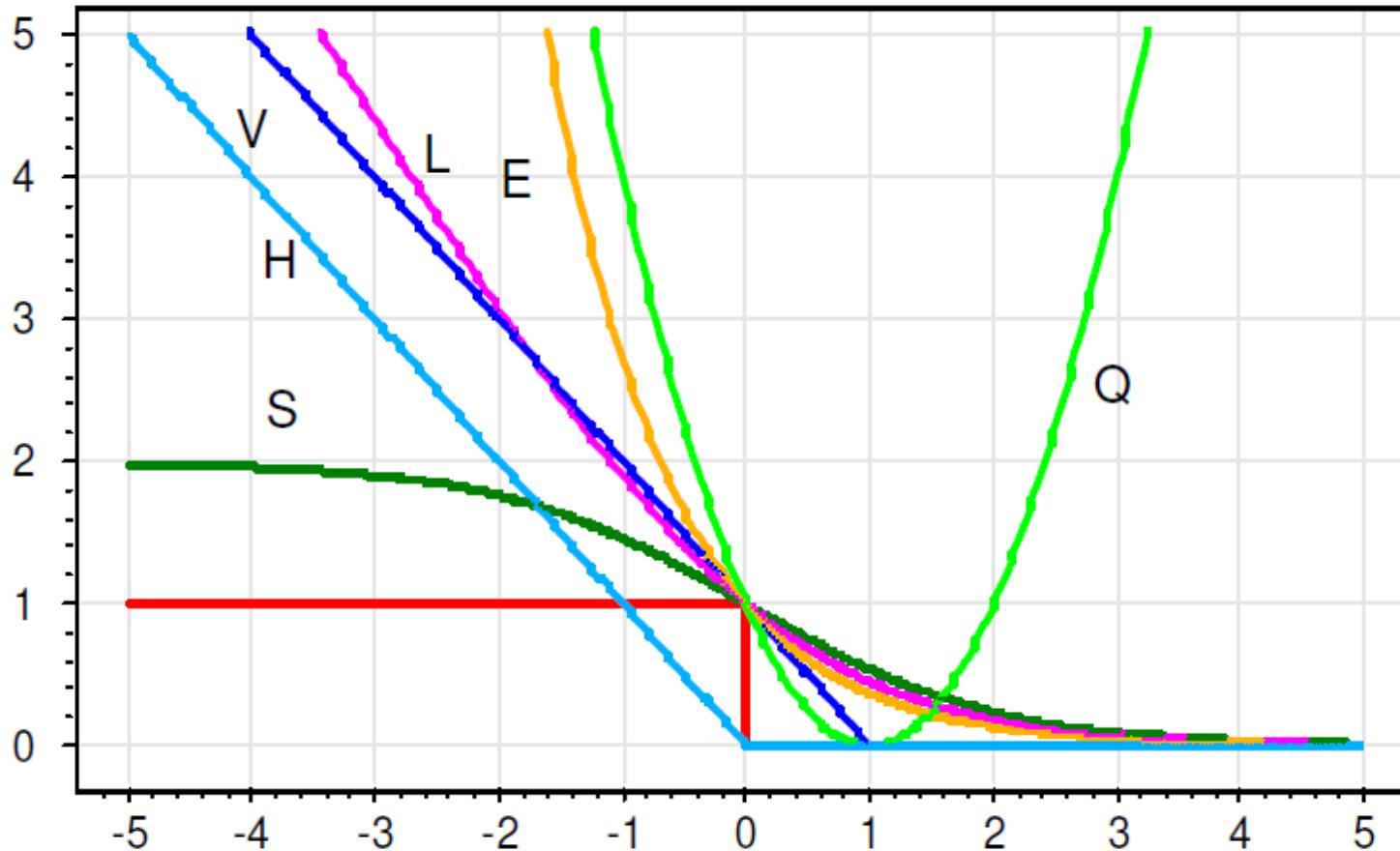
ФУНКЦИИ ПОТЕРЬ

Минимизируя различные функции потерь, получаем разные результаты. Поэтому разные функции потерь определяют различные классификаторы.

- $L(M) = \log(1 + e^{-M})$ – логистическая функция потерь } лог. рег
- $V(M) = (1 - M)_+ = \max(0, 1 - M)$ – кусочно-линейная } SVM
- $H(M) = (-M)_+ = \max(0, -M)$ – кусочно-линейная } перцептрон
- $E(M) = e^{-M}$ - экспоненциальная функция потерь
- $S(M) = \frac{2}{1+e^{-M}}$ - сигмоидная функция потерь
- $[M < 0]$ – пороговая функция потерь

далее в
курсе поговорим
про каждую
из них
они задают
разные модели

ФУНКЦИИ ПОТЕРЬ



M

ОПТИМИЗАЦИЯ ФУНКЦИОНАЛА ПОТЕРЬ

- Нахождение минимума функции потерь Q происходит с помощью метода градиентного спуска:

$$\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} - \eta \cdot \nabla Q(\mathbf{w}^{(k-1)})$$