Занятие 4
Линейные модели классификации.

Елена Кантонистова

elena.kantonistova@yandex.ru

ВШЭ, 2023

ПЛАН ЗАНЯТИЯ

- Градиентный спуск в python (окончание)
- Задача классификации и метрики
- Задача классификации в python

ОБУЧЕНИЕ ЛИНЕЙНОЙ РЕГРЕССИИ (НАПОМИНАНИЕ)

Обучающая выборка:

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

Модель линейной регрессии:

$$a(\boldsymbol{x}) = w_0 + w_1 x_1 + \cdots = \sum_{i=1}^{l} w_j x_j$$

ОБУЧЕНИЕ ЛИНЕЙНОЙ РЕГРЕССИИ (НАПОМИНАНИЕ)

Обучающая выборка:

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

Модель линейной регрессии:

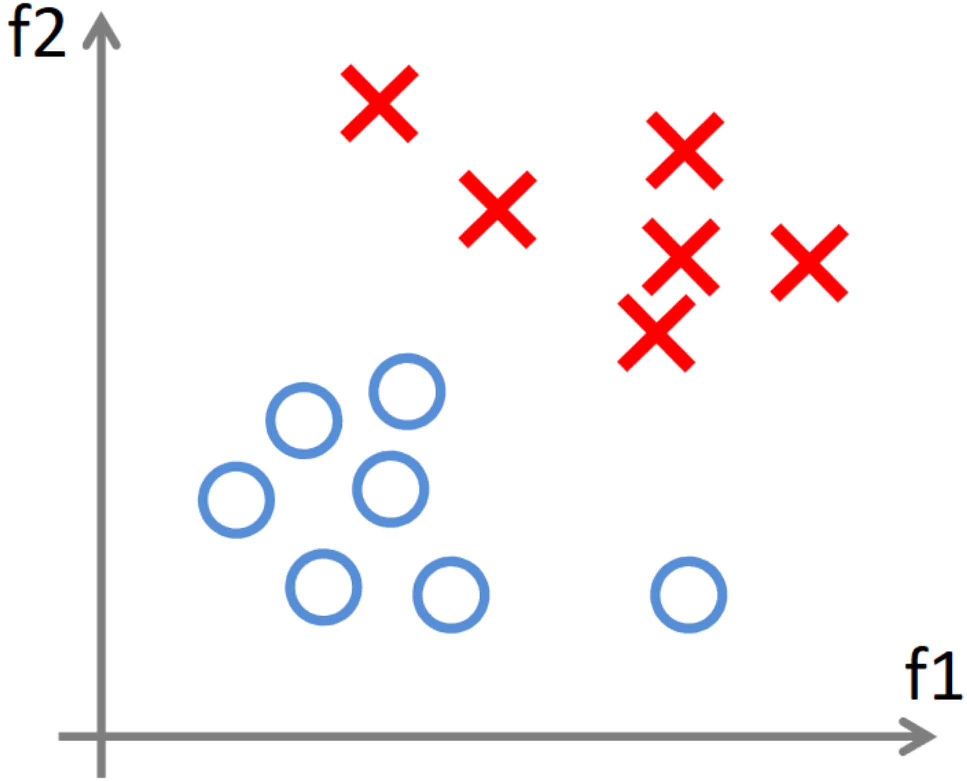
$$a(\boldsymbol{x}) = w_0 + w_1 x_1 + \cdots = \sum_{i=1}^{l} w_j x_j$$

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

$$Q(w) = \sum_{i=1}^{n} (a(\boldsymbol{x}_i) - y_i)^2 \to \min_{w}$$

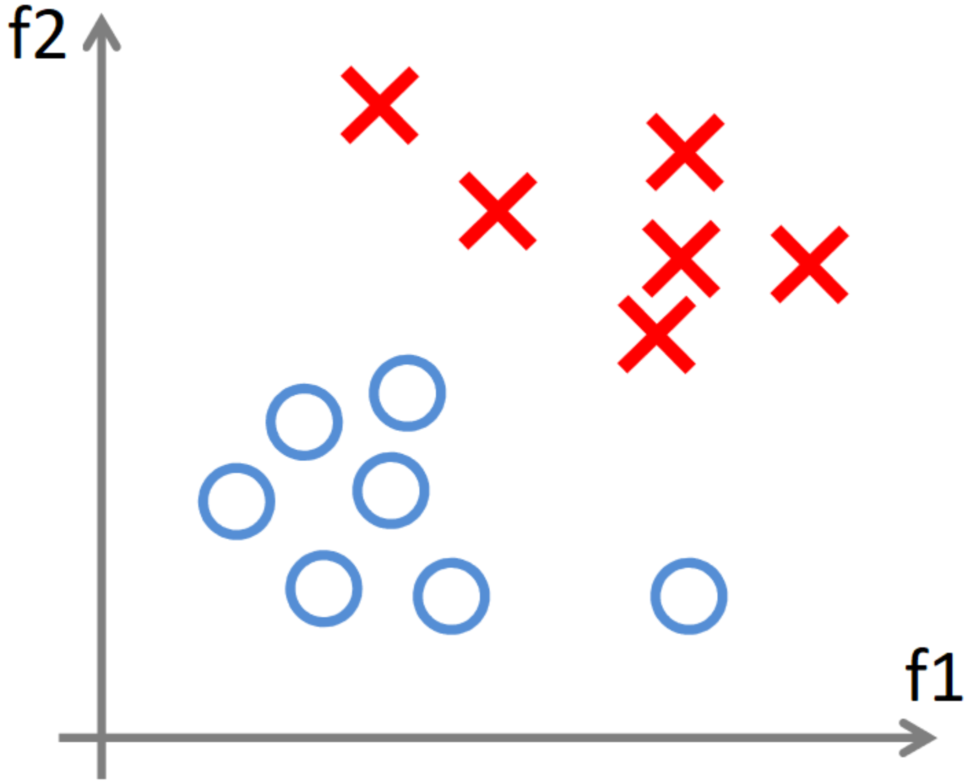
БИНАРНАЯ КЛАССИФИКАЦИЯ

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



БИНАРНАЯ КЛАССИФИКАЦИЯ

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



Как выглядит модель линейного классификатора: $\boldsymbol{a}(\boldsymbol{x}, \boldsymbol{w}) = ?$

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

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

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

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

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

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

$$\boldsymbol{a}(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{sign}\Big(\sum_{j=1}^{l} \boldsymbol{w_j x_j}\Big)$$

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

Почему такой классификатор будет линейным?

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

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

- если $\sum_{j=1}^{l} w_j x_j > 0$, то $sign(\sum_{j=1}^{l} w_j x_j) = +1$, то есть объект отнесён к положительному классу
- если $\sum_{j=1}^{l} w_j x_j < 0$, то $sign(\sum_{j=1}^{l} w_j x_j) = -1$, то есть объект отнесён к отрицательному классу
- значит, $\sum_{j=1}^{l} w_j x_j = w_1 x_1 + w_2 x_2 + \cdots = 0$ – *уравнение разделяющей границы* между классами. *Это уравнение плоскости* (или прямой в двумерном случае), поэтому *классификатор является линейным*.

БИНАРНАЯ КЛАССИФИКАЦИЯ

Модель линейного классификатора:

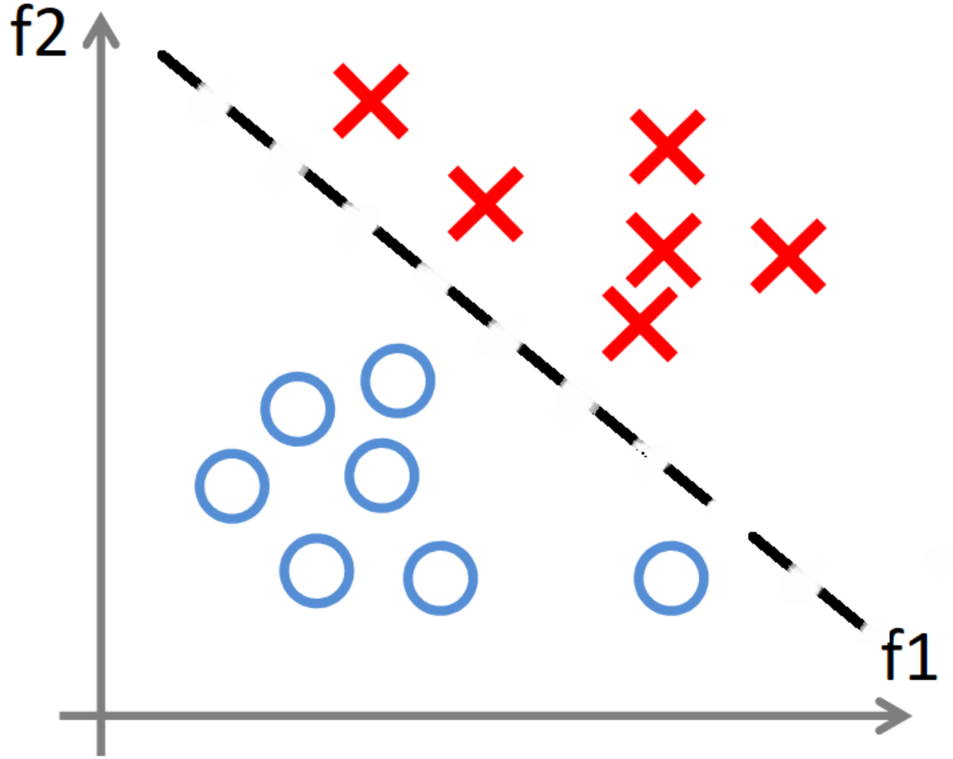
$$\boldsymbol{a(x, w) = sign(\sum_{j=1}^{l} w_j x_j)}$$

Уравнение

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

– уравнение плоскости

(или прямой).



ОБУЧЕНИЕ КЛАССИФИКАТОРА

Как обучить линейный классификатор?

ОБУЧЕНИЕ КЛАССИФИКАТОРА

- Обучение - минимизация доли ошибок классификатора:

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

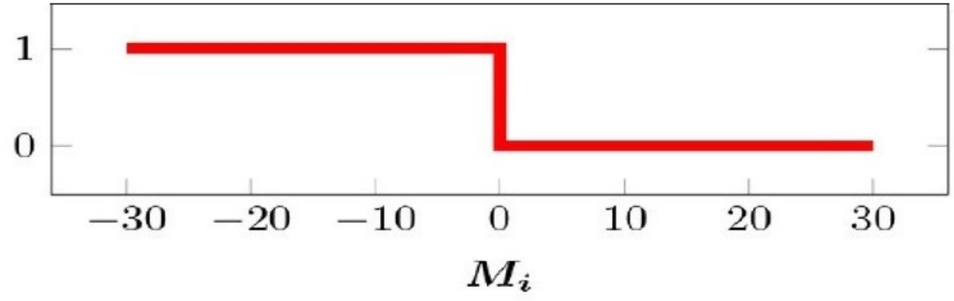
где $[a(\boldsymbol{x_i}) \neq y_i] = 1$, если предсказание на объекте неверное, то есть $a(\boldsymbol{x_i}) \neq y_i$, и 0 иначе.

ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

Обучение классификатора – это минимизация ***пороговой функции потерь***:

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

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

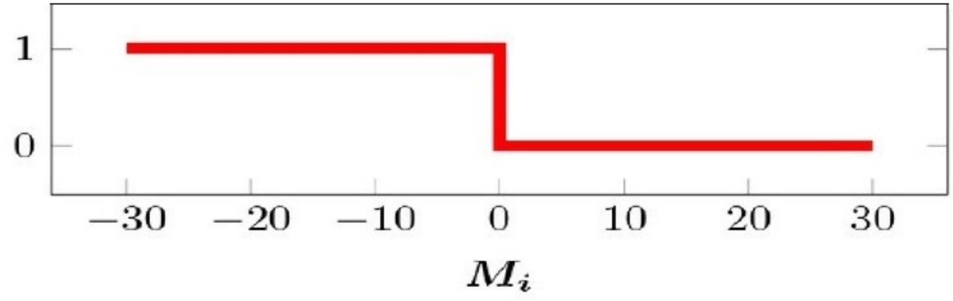


ПОРОГОВАЯ ФУНКЦИЯ ПОТЕРЬ

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

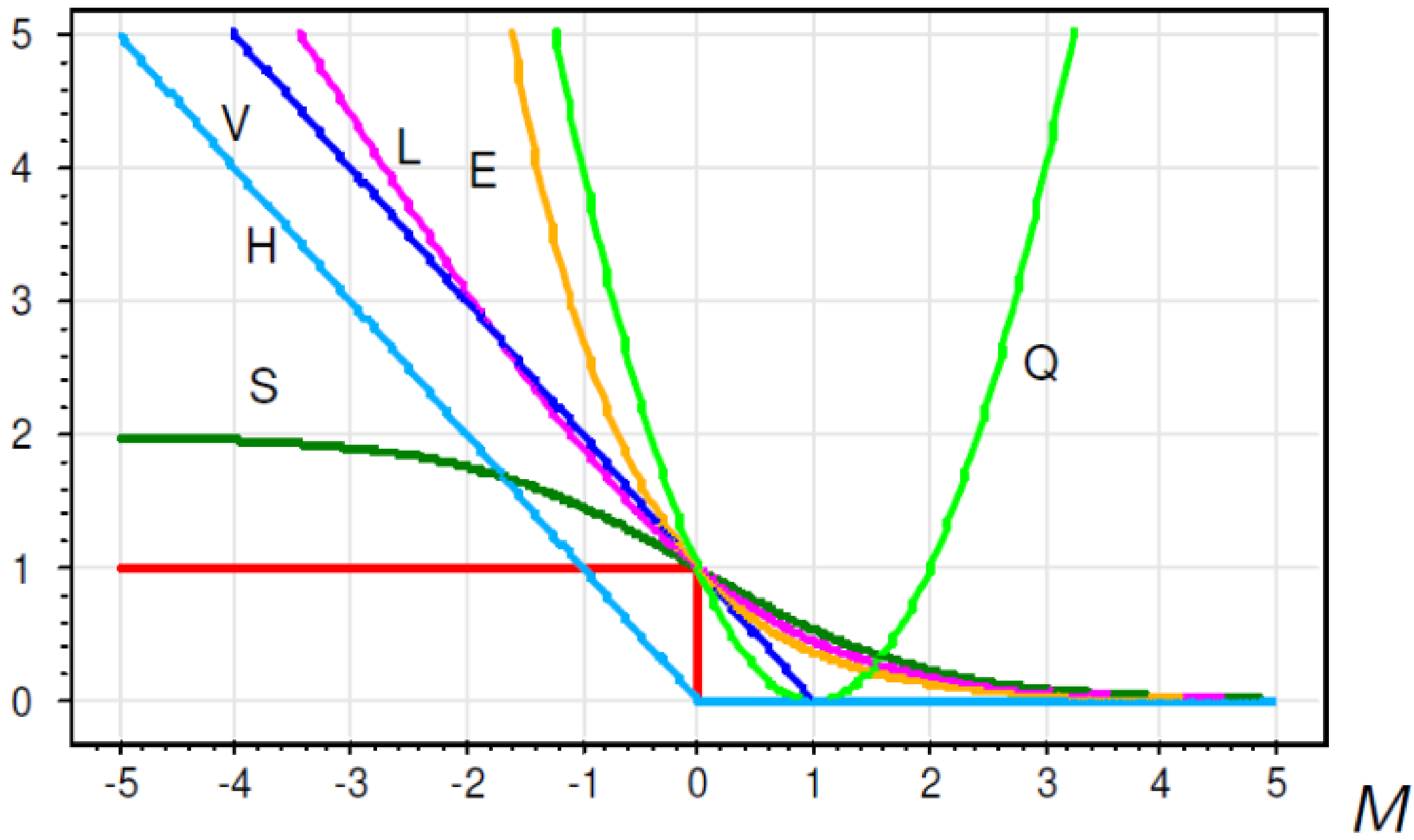
$$Q(a, X) = \frac{1}{n}\sum_{i=1}^{n}[\boldsymbol{a}(\boldsymbol{x}_i) \neq \boldsymbol{y}_i] \to min\,,$$

- 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***.

ФУНКЦИИ ПОТЕРЬ



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
- $\mathrm{V}(M) = (1 - \mathrm{M})_{+} = \max(0{,}1 - M)$ – кусочно-линейная функция потерь (метод опорных векторов)
- $H(M) = (-\mathrm{M})_{+} = \max(0, -M)$ – кусочно-линейная функция потерь (персептрон)
- $E(M) = e^{-M}$ - экспоненциальная функция потерь
- $S(M) = \frac{2}{1+e^{-M}}$ - сигмоидная функция потерь
- $[M < 0]$ – пороговая функция потерь

ОПТИМИЗАЦИЯ ФУНКЦИОНАЛА ПОТЕРЬ

- Нахождение минимума функции потерь $\boldsymbol{Q}$ происходит с помощью метода градиентного спуска:

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

МЕТРИКИ КАЧЕСТВА

МЕТРИКИ КАЧЕСТВА КЛАССИФИКАЦИИ

- Accuracy – доля правильных ответов:

$$\boldsymbol{accuracy}(\boldsymbol{a},\boldsymbol{X})=\frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}}[\boldsymbol{a}(\boldsymbol{x}_{\boldsymbol{i}})=\boldsymbol{y}_{\boldsymbol{i}}]$$

МЕТРИКИ КАЧЕСТВА КЛАССИФИКАЦИИ

- Accuracy – доля правильных ответов:

$$\boldsymbol{accuracy(a, X) = \frac{1}{n}\sum_{i=1}^{n}[a(x_i) = y_i]}$$

Недостаток: при сильно несбалансированной выборке не отражает качество работы алгоритма

МАТРИЦА ОШИБОК

Матрица ошибок (confusion matrix):

		Actual Value	
		positives	negatives
Predicted Value	positives	**TP** True Positive	**FP** False Positive
	negatives	**FN** False Negative	**TN** True Negative

МЕТРИКИ КАЧЕСТВА КЛАССИФИКАЦИИ: PRECISION, RECALL

- **Precision (точность):**

$$\boldsymbol{Precision(a, X) = \frac{TP}{TP + FP}}$$

Показывает, насколько можно доверять классификатору при $a(x) = +1$.

PRECISION: ПРИМЕР

Модель $a_1(x)$:

$\text{precision}(a_1, X) = 0.8$

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$\mathbf{a(x) = 1}$ Получили кредит	80	20
$\mathbf{a(x) = -1}$ Не получили кредит	20	80

Модель $a_2(x)$:

$\text{precision}(a_2, X) = 0.96$

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$\mathbf{a(x) = 1}$ Получили кредит	48	2
$\mathbf{a(x) = -1}$ Не получили кредит	52	98

МЕТРИКИ КАЧЕСТВА КЛАССИФИКАЦИИ: PRECISION, RECALL

- Precision (точность):

$$\boldsymbol{Precision(a,X) = \frac{TP}{TP+FP}}$$

Показывает, насколько можно доверять классификатору при $a(x) = +1$.

- **Recall (полнота)**:

$$\boldsymbol{Recall(a,X) = \frac{TP}{TP+FN}}$$

Показывает, как много объектов положительного класса находит классификатор.

RECALL: ПРИМЕР

Модель $a_1(x)$:

$\mathrm{rec}all(a_1, X) = 0.8$

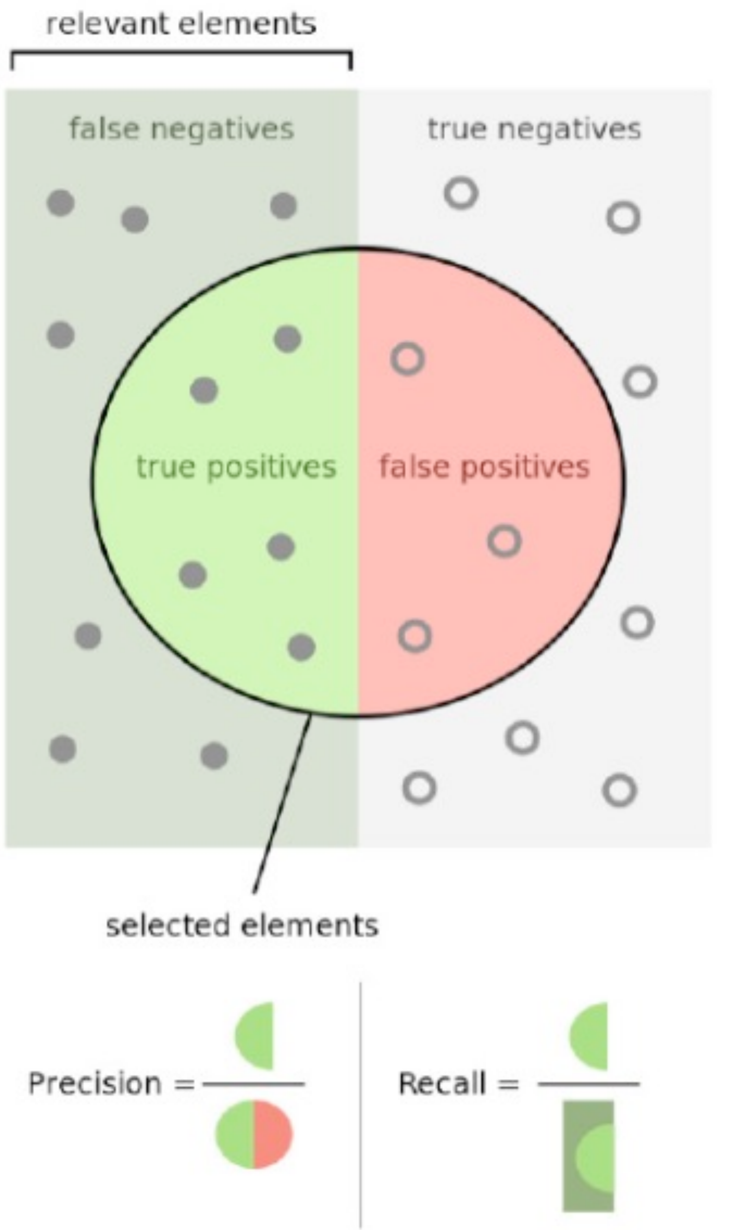
	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$\mathbf{a}(\mathbf{x}) = \mathbf{1}$ Получили кредит	80	20
$\mathbf{a}(\mathbf{x}) = -\mathbf{1}$ Не получили кредит	20	80

Модель $a_2(x)$:

$\mathrm{recall}(a_2, X) = 0.48$

	$y = 1$ Могут вернуть	$y = -1$ Не могут вернуть
$\mathbf{a}(\mathbf{x}) = \mathbf{1}$ Получили кредит	48	2
$\mathbf{a}(\mathbf{x}) = -\mathbf{1}$ Не получили кредит	52	98

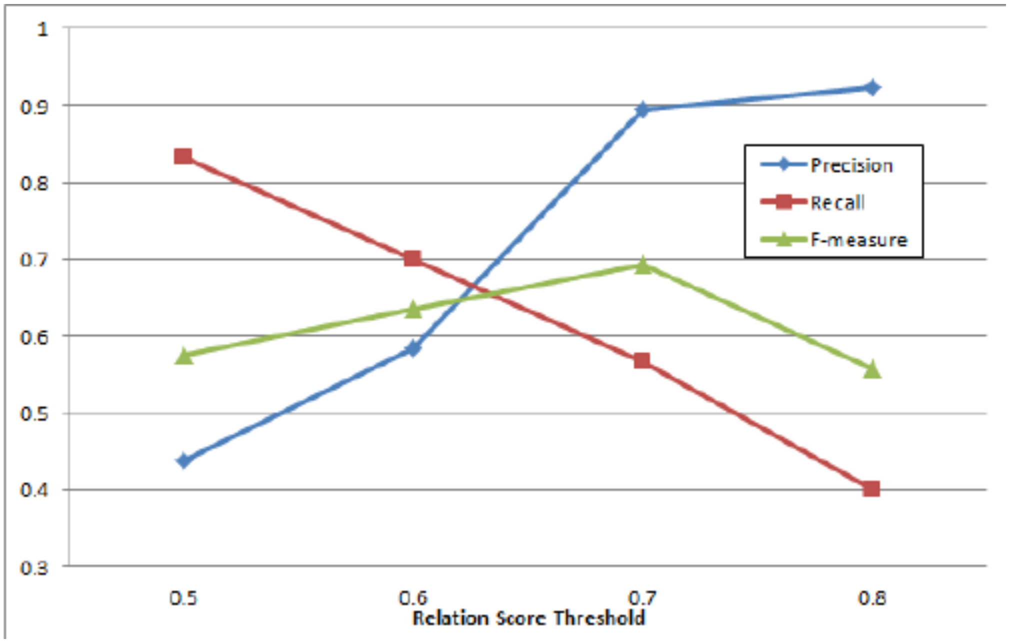
ТОЧНОСТЬ И ПОЛНОТА



F-МЕРА

F-мера – это метрика качества, учитывающая и точность, и полноту

$$F(a, X) = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$



РЕГУЛИРУЕМ ТОЧНОСТЬ И ПОЛНОТУ

Пусть $p(x)$ - уверенность классификатора в том, что объект x относится к классу $+1$, $p(x) \in [0; 1]$.

Обычно если $p(x) > 0.5$, то мы относим объект к положительному классу, а иначе – к отрицательному.

РЕГУЛИРУЕМ ТОЧНОСТЬ И ПОЛНОТУ

Пусть $p(x)$ - уверенность классификатора в том, что объект x относится к классу $+1$, $\quad p(x) \in [0; 1]$.

Обычно если $p(x) > 0.5$, то мы относим объект к положительному классу, а иначе – к отрицательному.

Можно изменять этот порог, то есть вместо 0.5 брать другое число из отрезка $[0; 1]$.

РЕГУЛИРУЕМ ТОЧНОСТЬ И ПОЛНОТУ

Пусть $p(x)$ - уверенность классификатора в том, что объект x относится к классу $+1$, $p(x) \in [0; 1]$.

Обычно если $p(x) > 0.5$, то мы относим объект к положительному классу, а иначе – к отрицательному.

Можно изменять этот порог, то есть вместо 0.5 брать другое число из отрезка $[0; 1]$.

Путем изменения порога t можно регулировать точность и полноту:

- Чему будут равны точность и полнота при $t = 0$?

РЕГУЛИРУЕМ ТОЧНОСТЬ И ПОЛНОТУ

Пусть $p(x)$ - уверенность классификатора в том, что объект x относится к классу $+1$, $p(x) \in [0; 1]$.

Обычно если $p(x) > 0.5$, то мы относим объект к положительному классу, а иначе – к отрицательному.

Можно изменять этот порог, то есть вместо 0.5 брать другое число из отрезка $[0; 1]$.

Путем изменения порога t можно регулировать точность и полноту:

- при $t = 0$ мы все объекты относим к положительному классу, то есть ***полнота = 1, а точность маленькая***.

РЕГУЛИРУЕМ ТОЧНОСТЬ И ПОЛНОТУ

Пусть $p(x)$ - уверенность классификатора в том, что объект x относится к классу $+1$, $p(x) \in [0; 1]$.

Обычно если $p(x) > 0.5$, то мы относим объект к положительному классу, а иначе – к отрицательному.

Можно изменять этот порог, то есть вместо 0.5 брать другое число из отрезка $[0; 1]$.

Путем изменения порога t можно регулировать точность и полноту:

- при $t = 0$ мы все объекты относим к положительному классу, то есть полнота = 1, а точность маленькая.
- ***При увеличении t полнота уменьшается*** (могут появиться объекты положительного класса, которые мы не нашли), ***а точность возрастает*** (появляются объекты положительного класса).

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Хотим предсказывать не только классы, но и ***вероятности классов***.

- Линейная регрессия:

$$a(x, w) = w_0 + w_1 x_1 + w_2 x_2 + \ \dots$$

- Линейный классификатор (любой):

$$a(x, w) = sign(w_0 + w_1 x_1 + w_2 x_2 + \ \dots)$$

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Хотим предсказывать не только классы, но и ***вероятности классов***.

- Линейная регрессия:

$$a(x, w) = w_0 + w_1 x_1 + w_2 x_2 + \ \dots$$

- Линейный классификатор (любой):

$$a(x, w) = sign(w_0 + w_1 x_1 + w_2 x_2 + \ \dots)$$

- Логистическая регрессия:

$$\boldsymbol{a(x, w) = \sigma(w_0 + w_1 x_1 + w_2 x_2 + \ \dots) = \sigma(w, x)},$$

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

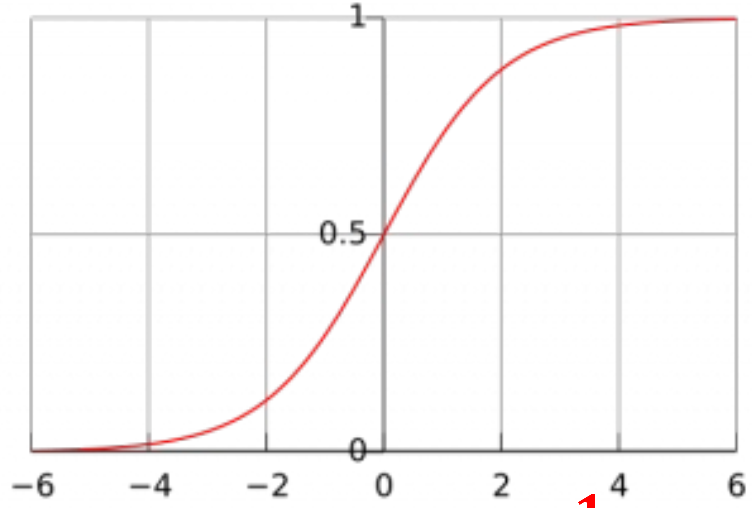
ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Хотим предсказывать не классы, а вероятности классов.

- Логистическая регрессия: $\boldsymbol{a(x, w) = \sigma(w, x)}$,

где $\sigma(z) = \frac{1}{1+e^{-z}}$ - сигмоида (логистическая функция),

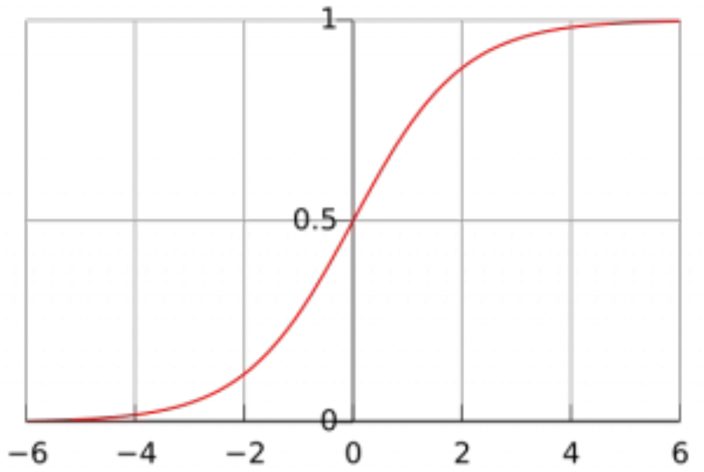
$\sigma(z) \in (0; 1)$.



Логистическая регрессия: $\boldsymbol{a(x, w) = \frac{1}{1+e^{-(w,x)}}}$

РАЗДЕЛЯЮЩАЯ ГРАНИЦА

Предсказываем $y = +1$, если $a(x, w) \geq 0.5$.



$a(x, w) = \sigma(w, x) \geq 0.5$, если $(w, x) \geq 0$.

Получаем, что

- $y = +1 \text{ при } (w, x) \geq 0$
- $y = -1 \text{ при } (w, x) < 0,$

т.е. $(w, x) = w_1 x_1 + w_2 x_2 + \cdots = \ 0$ – разделяющая гиперплоскость.

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Логистическая регрессия - это линейный классификатор!