



# Многоклассовая и нелинейная классификация. Отбор признаков.

# МНОГОКЛАССОВАЯ КЛАССИФИКАЦИЯ

# ПОДХОД ONE-VS-ALL

Решаем задачу классификации на  $K$  классов.

- Обучим  $K$  бинарных классификаторов  $b_1(x), \dots, b_K(x)$ , каждый из которых решает задачу: **принадлежит объект  $x$  к классу  $k_i$  или не принадлежит?**

Например, линейные классификаторы будут иметь вид

$$b_k(x) = \text{sign}((w_k, x))$$

# ПОДХОД ONE-VS-ALL

Решаем задачу классификации на  $K$  классов.

- Обучим  $K$  бинарных классификаторов  $b_1(x), \dots, b_K(x)$ , каждый из которых решает задачу: *принадлежит объект  $x$  к классу  $k_i$  или не принадлежит?*

Например, линейные классификаторы будут иметь вид

$$b_k(x) = \text{sign}((w_k, x))$$

- Тогда в качестве итогового предсказания будем выдавать **класс самого уверенного классификатора**:

$$a(x) = \underset{k \in \{1, \dots, K\}}{\operatorname{argmax}} ((w_k, x))$$

# ПОДХОД ONE-VS-ALL

Решаем задачу классификации на  $K$  классов.

- Обучим  $K$  бинарных классификаторов  $b_1(x), \dots, b_k(x)$ , каждый из которых решает задачу: *принадлежит объект  $x$  к классу  $k_i$  или не принадлежит?*

Например, линейные классификаторы будут иметь вид

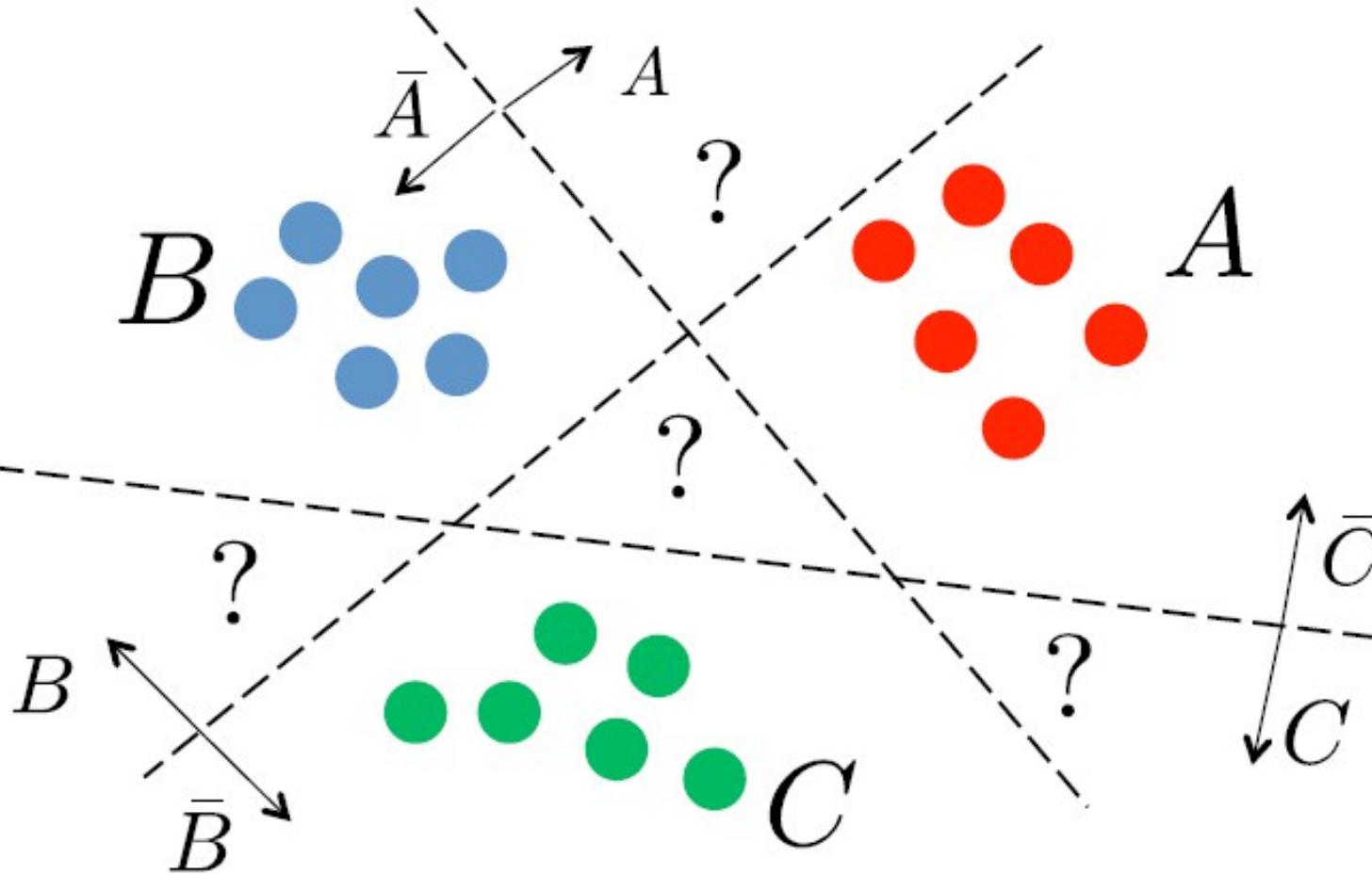
$$b_k(x) = \text{sign}((w_k, x))$$

- Тогда в качестве итогового предсказания будем выдавать класс самого уверенного классификатора:

$$a(x) = \underset{k \in \{1, \dots, K\}}{\operatorname{argmax}} ((w_k, x))$$

- **Предсказания классификаторов могут иметь разные масштабы, поэтому сравнивать их некорректно.**

# ПОДХОД ONE-VS-ALL



## ПОДХОД ALL-VS-ALL

- Для каждой пары классов  $i$  и  $j$  обучим бинарный классификатор  $a_{ij}(x)$ , который будет предсказывать класс  $i$  или  $j$

(если всего  $K$  классов, то получим  $C_K^2$  классификаторов).

Каждый такой классификатор будем обучать только на объектах классов  $i$  и  $j$ .

## ПОДХОД ALL-VS-ALL

- Для каждой пары классов  $i$  и  $j$  обучим бинарный классификатор  $a_{ij}(x)$ , который будет предсказывать класс  $i$  или  $j$

(если всего  $K$  классов, то получим  $C_K^2$  классификаторов).

Каждый такой классификатор будем обучать только на объектах классов  $i$  и  $j$ .

- **В качестве итогового предсказания выдадим класс, который предсказало наибольшее число алгоритмов:**

$$a(x) = \operatorname{argmax}_{k \in \{1, \dots, K\}} \sum_{i=1}^K \sum_{j \neq i} [a_{ij}(x) = k]$$

## ПОДХОД ALL-VS-ALL

- Для каждой пары классов  $i$  и  $j$  обучим бинарный классификатор  $a_{ij}(x)$ , который будет предсказывать класс  $i$  или  $j$

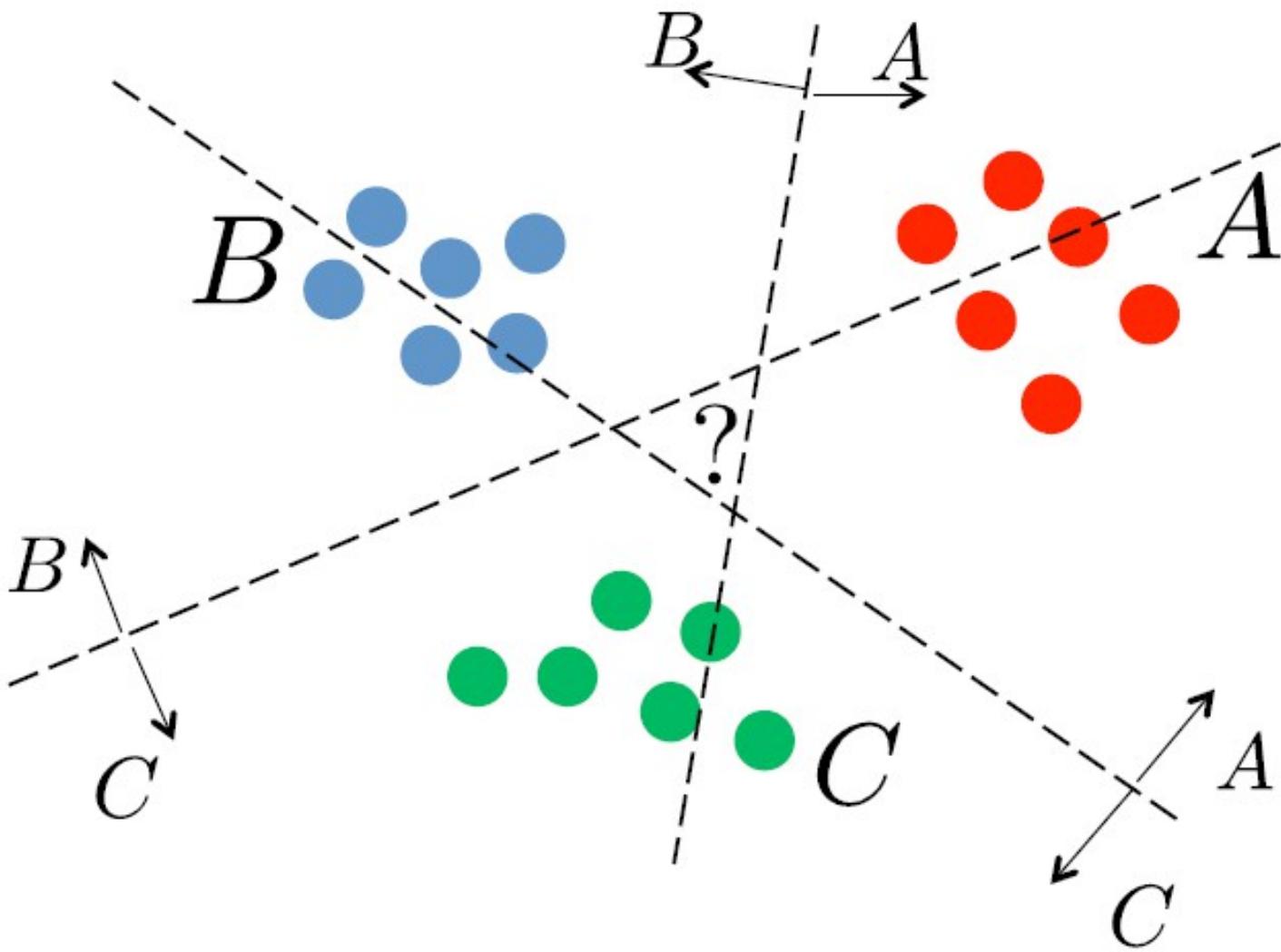
(если всего  $K$  классов, то получим  $C_K^2$  классификаторов).

Каждый такой классификатор будем обучать только на объектах классов  $i$  и  $j$ .

- В качестве итогового предсказания выдадим класс, который предсказало наибольшее число алгоритмов:

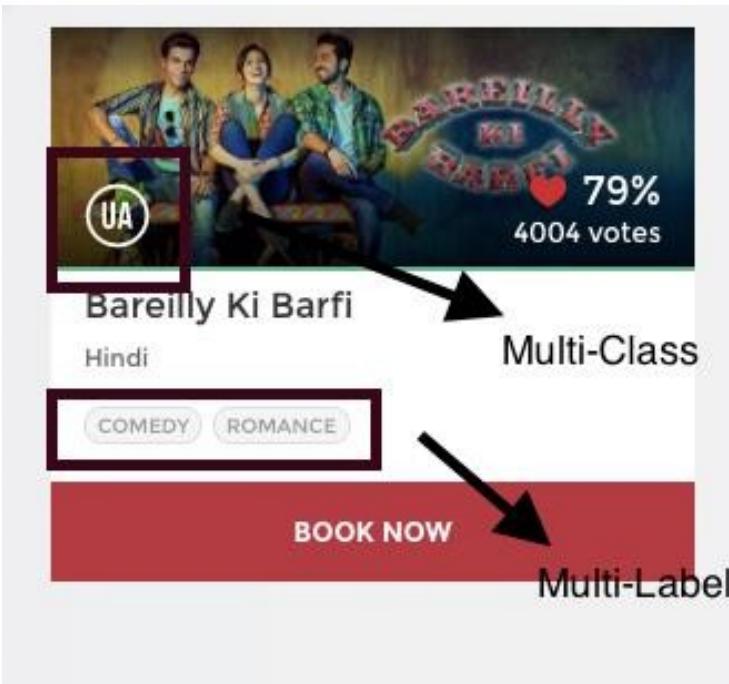
$$a(x) = \operatorname{argmax}_{k \in \{1, \dots, K\}} \sum_{i=1}^K \sum_{j \neq i} [a_{ij}(x) = k]$$

# ПОДХОД ALL-VS-ALL



# MULTICLASS AND MULTILABEL CLASSIFICATION

- Если каждый объект может принадлежать только одному классу, то решаем задачу **multiclass** классификации
- Если каждый объект может принадлежать нескольким классам (задача классификации с пересекающимися классами), то решаем задачу **multi-label** классификации.



# МЕТРИКИ КАЧЕСТВА

Идея: сводим подсчет метрик к бинарному случаю

Подход 1 (микроусреднение, micro average):

- Вычислим для каждого двухклассового классификатора

- Усредним каждую характеристику по всем классам,  
например,  $TP = \frac{1}{K} \sum_{k=1}^K TP_k$ .

Тогда точность в многоклассовом случае:

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

# МЕТРИКИ КАЧЕСТВА

Идея: сводим подсчет метрик к бинарному случаю

Подход 2 (макроусреднение, macro average):

- Вычислим для каждого двухклассового классификатора

$$a^k(x) = [a(x) = k] \text{ метрики } TP_k, FP_k, FN_k, TN_k$$

- Вычислим итоговую метрику для каждого класса в

отдельности:  $precision_k(a, X) = \frac{TP_k}{TP_k + FP_k}$

Тогда точность в многоклассовом случае:

$$precision(a, X) = \frac{1}{K} \sum_{k=1}^K precision_k(a, X)$$

# МНОГОКЛАССОВАЯ ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Бинарная лог.регрессия предсказывает вероятность класса

1:

$$(w, x) \rightarrow a(x) = \frac{1}{1 + e^{-(w, x)}} = \frac{e^{(w, x)}}{1 + e^{(w, x)}}$$

# МНОГОКЛАССОВАЯ ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Бинарная лог.регрессия предсказывает вероятность класса

1:

$$(w, x) \rightarrow a(x) = \frac{1}{1 + e^{-(w, x)}} = \frac{e^{(w, x)}}{1 + e^{(w, x)}}$$

- Предположим, у нас есть  $K$  моделей, каждая из которых дает оценку принадлежности выбранному классу:  $b_k(x) = (w_k, x)$ .
- Преобразуем вектор предсказаний в вектор вероятностей (softmax-преобразование):

$$\text{softmax}(\mathbf{b}_1, \dots, \mathbf{b}_K) = \left( \frac{\exp(b_1)}{\sum_{i=1}^K \exp(b_i)}, \frac{\exp(b_2)}{\sum_{i=1}^K \exp(b_i)}, \dots, \frac{\exp(b_K)}{\sum_{i=1}^K \exp(b_i)} \right)$$

# МНОГОКЛАССОВАЯ ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Бинарная лог.регрессия предсказывает вероятность класса 1:

$$(w, x) \rightarrow a(x) = \frac{1}{1 + e^{-(w, x)}} = \frac{e^{(w, x)}}{1 + e^{(w, x)}}$$

- Предположим, у нас есть  $K$  моделей, каждая из которых дает оценку принадлежности выбранному классу:  $b_k(x) = (w_k, x)$ .
- Преобразуем вектор предсказаний в вектор вероятностей (softmax-преобразование):

$$\text{softmax}(\mathbf{b}_1, \dots, \mathbf{b}_K) = \left( \frac{\exp(\mathbf{b}_1)}{\sum_{i=1}^K \exp(\mathbf{b}_i)}, \frac{\exp(\mathbf{b}_2)}{\sum_{i=1}^K \exp(\mathbf{b}_i)}, \dots, \frac{\exp(\mathbf{b}_K)}{\sum_{i=1}^K \exp(\mathbf{b}_i)} \right)$$

Тогда вероятность класса  $k$ :

$$P(y = k | x, w) = \frac{\exp((w_k, x))}{\sum_{i=1}^K \exp((w_i, x))}$$

# ОБУЧЕНИЕ ВЕСОВ МОДЕЛИ

$$a_j(x) = P(y = j|x, w) = \frac{\exp(b_j(x))}{\sum_{i=1}^K \exp(b_i(x))}$$

**Обучение – по методу максимального правдоподобия  
(аналогично бинарной классификации):**

$$\Pi = \prod_{i=1}^n a_1(x_i)^{[y_i=1]} \cdot a_2(x_i)^{[y_i=2]} \cdot \dots a_K(x_i)^{[y_i=K]} =$$

$$= \prod_{i=1}^n \prod_{j=1}^K a_j(x_i)^{[y_i=j]} \rightarrow \max_{w_1, \dots, w_K}$$

$$- \sum_{i=1}^n \sum_{j=1}^K [y_i = j] \log P(y = j|x_i, w) \rightarrow \min_{w_1, \dots, w_K}$$

# ОБУЧЕНИЕ ВЕСОВ МОДЕЛИ

$$a_j(x) = P(y = j|x, w) = \frac{\exp(b_j(x))}{\sum_{i=1}^K \exp(b_i(x))}$$

*Обучение – по методу максимального правдоподобия  
(аналогично бинарной классификации):*

$$\Pi = \prod_{i=1}^n a_1(x_i)^{[y_i=1]} \cdot a_2(x_i)^{[y_i=2]} \cdot \dots a_K(x_i)^{[y_i=K]} =$$

$$= \prod_{i=1}^n \prod_{j=1}^K a_j(x_i)^{[y_i=j]} \rightarrow \max_{w_1, \dots, w_K}$$

**С помощью ММП находим сразу все веса, а не обучаем  
отдельно каждую модель.**

$$-\sum_{i=1}^n \sum_{j=1}^K [y_i = j] \log P(y = j|x_i, w) \rightarrow \min_{w_1, \dots, w_K}$$

## 2. ПРОСТЫЕ НЕЛИНЕЙНЫЕ КЛАССИФИКАТОРЫ

# НАИВНЫЙ БАЙЕСОВСКИЙ КЛАССИФИКАТОР

# НАИВНЫЙ БАЙЕСОВСКИЙ КЛАССИФИКАТОР

**Наивный байесовский классификатор** – это алгоритм классификации, основанный на теореме Байеса с допущением о независимости признаков.

Пример: фрукт может считаться яблоком,

если:

- 1) он красный
- 2) круглый
- 3) его диаметр составляет порядка 8 см

Предполагаем, что признаки вносят независимый вклад в вероятность того, что фрукт является яблоком.



# ТЕОРЕМА БАЙЕСА

Теорема Байеса:

$$P(c|x) = \frac{P(x|c) \cdot P(c)}{P(x)}$$

- $P(c|x)$  - вероятность того,

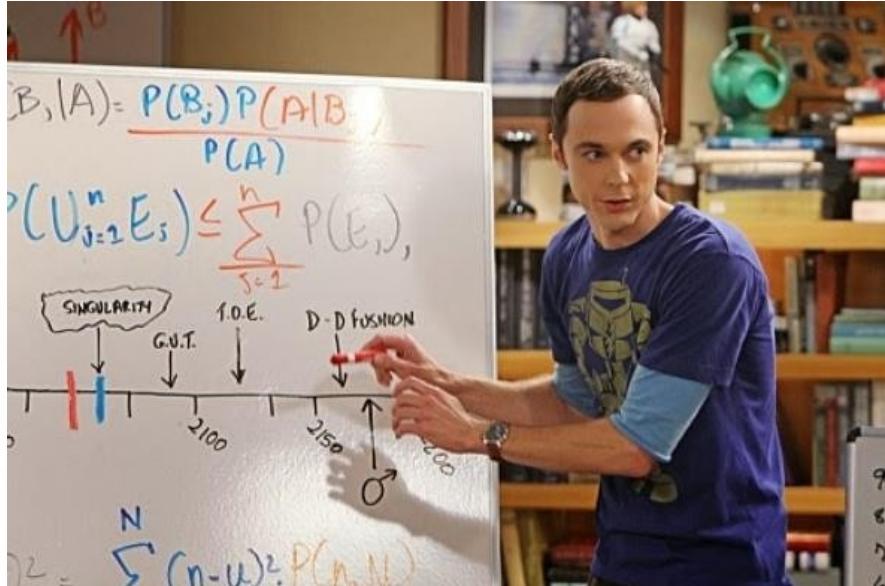
что объект со значением признака  $x$

принадлежит классу  $c$ .

- $P(c)$  – априорная вероятность класса  $c$ .

- $P(x|c)$  - вероятность того, что значение признака равно  $x$  при условии, что объект принадлежит классу  $c$ .

- $P(x)$  – априорная вероятность значения признака  $x$ .



# ПРИМЕР РАБОТЫ БАЙЕСОВСКОГО АЛГОРИТМА

Пример: на основе данных о погодных условиях необходимо определить, состоится ли матч.

- Преобразуем набор данных в следующую таблицу:

Weather	No	Yes
Overcast	0	4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

# ПРИМЕР РАБОТЫ БАЙЕСОВСКОГО АЛГОРИТМА

Решим задачу с помощью теоремы Байеса:

$$P(Yes|Sunny) = P(Sunny|Yes) \cdot P(Yes)/P(Sunny)$$

Таблица частот			
Weather	No	Yes	
Overcast	0	4	=4/14 0.29
Rainy	3	2	=5/14 0.36
Sunny	2	3	=5/14 0.36
Grand Total	5	9	
	=5/14	=9/14	
	0.36	0.64	

- $P(Sunny|Yes) = \frac{3}{9}, P(Sunny) = \frac{5}{14}, P(Yes) = \frac{9}{14}.$
- $P(Yes|Sunny) = \frac{3}{9} \cdot \frac{9}{14} : \frac{5}{14} = \frac{3}{5} = 0,6 \Rightarrow 60\%.$

# В СЛУЧАЕ НЕСКОЛЬКИХ ПРИЗНАКОВ

Пусть  $x_1, \dots, x_n$  – признаки объекта,  $y$  – целевая переменная.

Тогда теорема Байеса записывается в виде

$$P(y|x_1, \dots, x_n) = \frac{P(x_1|y)P(x_2|y)\dots P(x_n|y)P(y)}{P(x_1)P(x_2)\dots P(x_n)}.$$

Вероятности в правой части формулы вычисляются с помощью частотных таблиц, как и в одномерном случае.

# БАЙЕСОВСКИЙ АЛГОРИТМ ДЛЯ КЛАССИФИКАЦИИ

Плюсы и минусы:

- + классификация быстрая и простая
- + в случае, если выполняется предположение о независимости, классификатор показывает очень высокое качество
- если в тестовых данных присутствует категория, не встречавшаяся в данных для обучения, модель присвоит ей нулевую вероятность

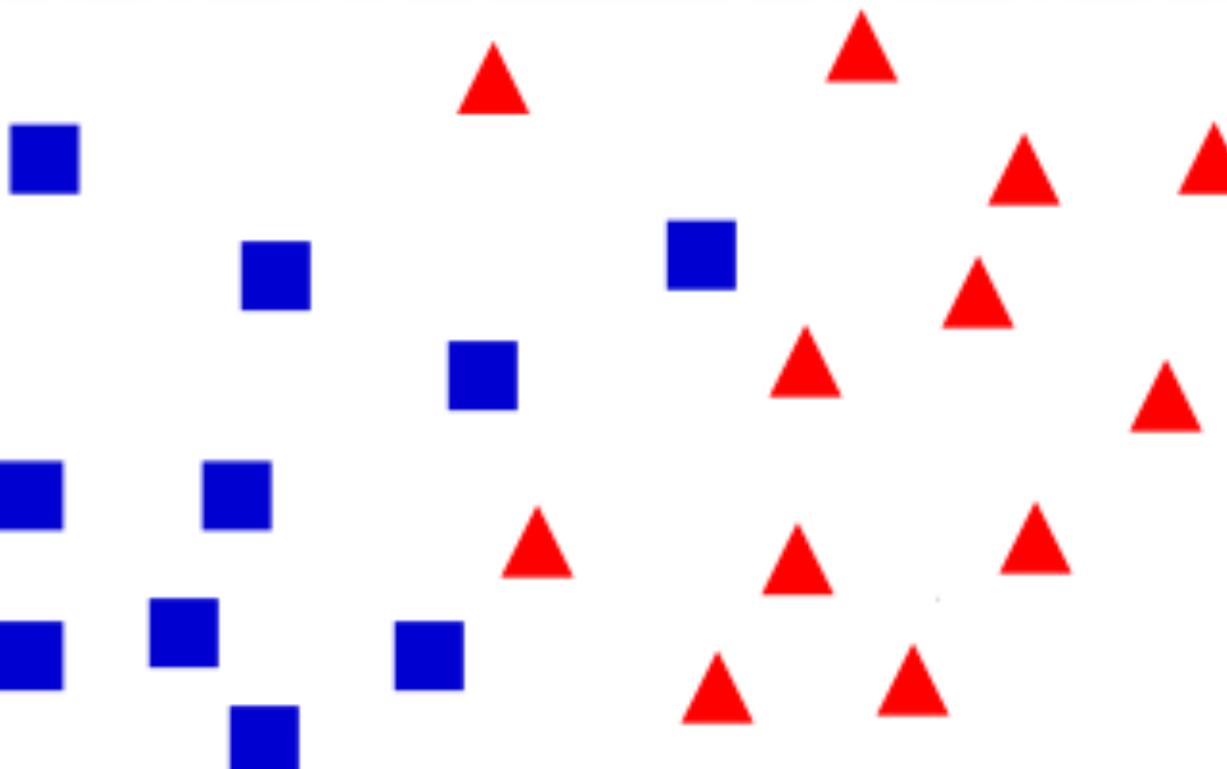
# НАИВНЫЙ БАЙЕСОВСКИЙ АЛГОРИТМ

[https://scikit-learn.org/stable/modules/naive\\_bayes.html](https://scikit-learn.org/stable/modules/naive_bayes.html)

# МЕТОД БЛИЖАЙШИХ СОСЕДЕЙ

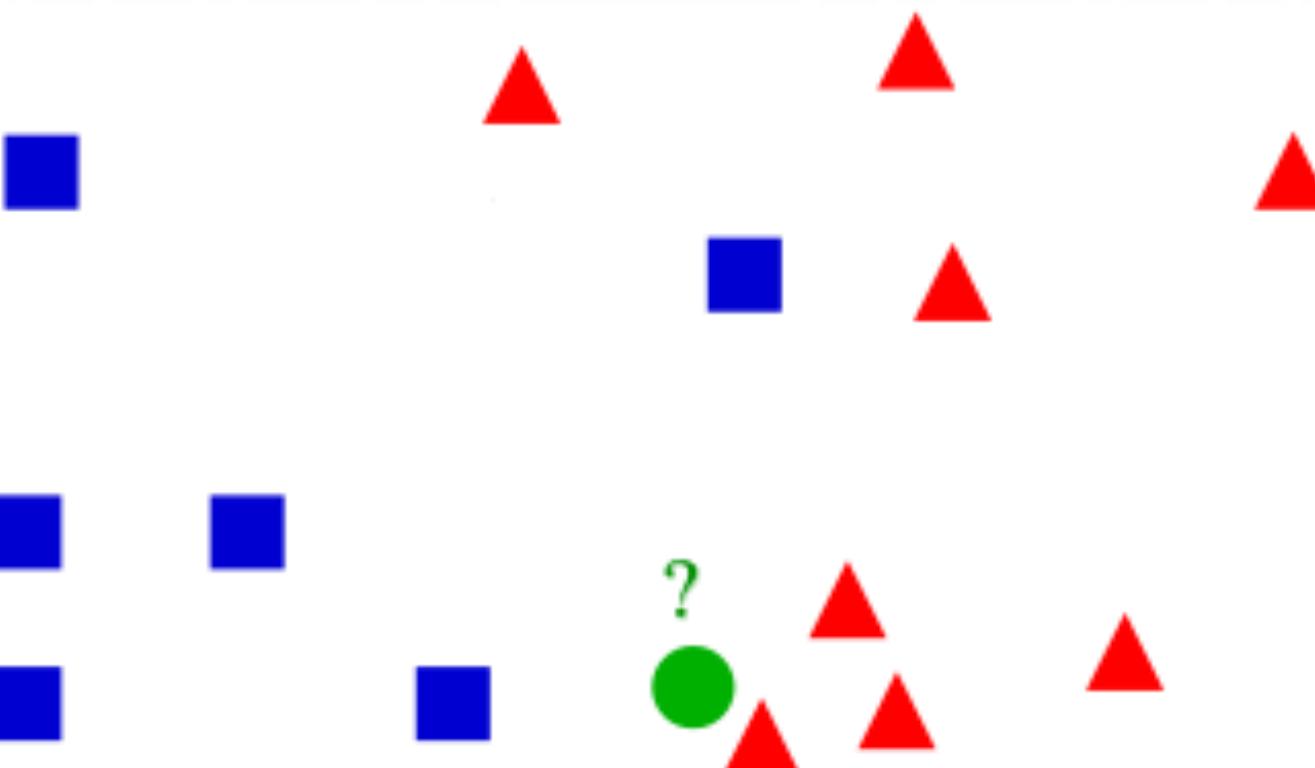
# МЕТОД БЛИЖАЙШИХ СОСЕДЕЙ

**Идея:** схожие объекты находятся близко друг к другу в пространстве признаков.



# МЕТОД БЛИЖАЙШИХ СОСЕДЕЙ

*Как классифицировать новый объект?*



# МЕТОД БЛИЖАЙШИХ СОСЕДЕЙ

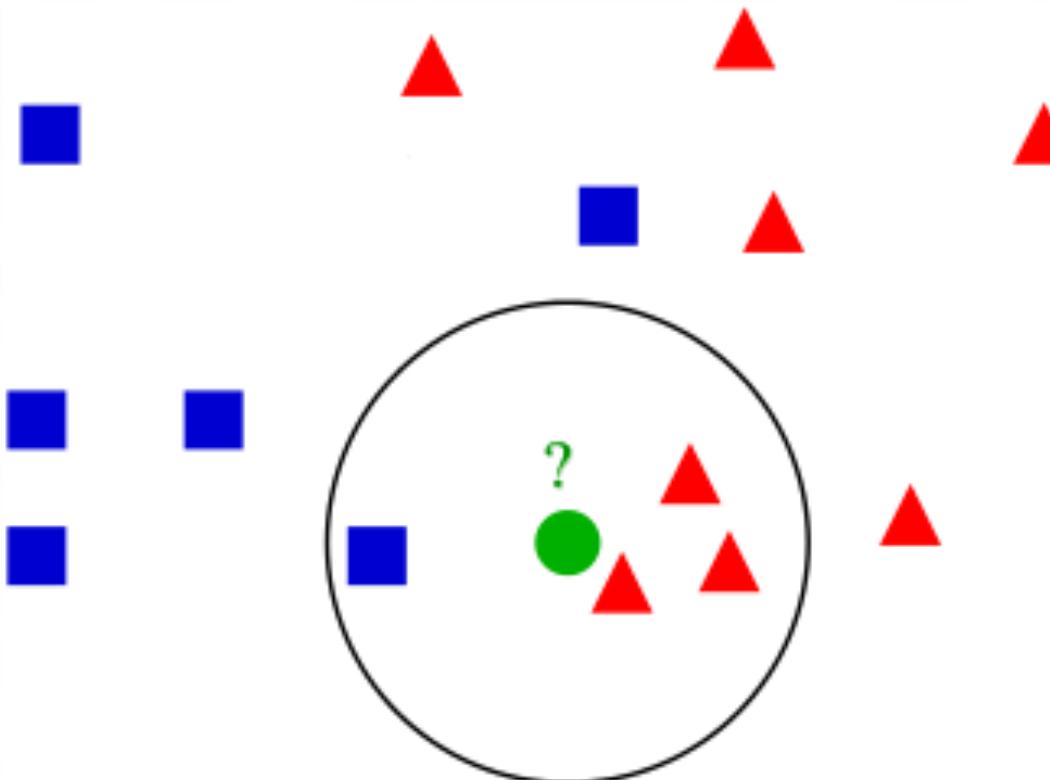
Чтобы классифицировать новый объект, нужно:

- Вычислить расстояние до каждого из объектов обучающей выборки.
- Выбрать  $k$  объектов обучающей выборки, расстояние до которых минимально.
- Класс классифицируемого объекта — это класс, наиболее часто встречающийся среди  $k$  ближайших соседей.

# МЕТОД БЛИЖАЙШИХ СОСДЕЙ

Число ближайших соседей  $k$  – гиперпараметр метода.

Например, для  $k = 4$  получим:



То есть объект будет отнесён к классу *треугольников*.

# ФОРМАЛИЗАЦИЯ МЕТОДА

Пусть  $k$  – количество соседей. Для каждого объекта  $u$  возьмём  $k$  ближайших к нему объектов из тренировочной выборки:

$$x_{(1;u)}, x_{(2;u)}, \dots, x_{(k;u)}.$$

Тогда класс объекта  $u$  определяется следующим образом:

$$a(u) = \operatorname{argmax}_{y \in Y} \sum_{i=1}^k [y(x_{(i;u)}) = y].$$

# ФОРМАЛИЗАЦИЯ МЕТОДА

Пусть  $k$  – количество соседей. Для каждого объекта  $u$  возьмём  $k$  ближайших к нему объектов из тренировочной выборки:

$$x_{(1;u)}, x_{(2;u)}, \dots, x_{(k;u)}.$$

Тогда класс объекта  $u$  определяется следующим образом:

$$a(u) = \operatorname{argmax}_{y \in Y} \sum_{i=1}^k [y(x_{(i;u)}) = y].$$

*Ближайшие объекты* – это объекты, расстояние от которых до данного объекта наименьшее по некоторой метрике  $\rho$ .

# ФОРМАЛИЗАЦИЯ МЕТОДА

Пусть  $k$  – количество соседей. Для каждого объекта  $u$  возьмём  $k$  ближайших к нему объектов из тренировочной выборки:

$$x_{(1;u)}, x_{(2;u)}, \dots, x_{(k;u)}.$$

Тогда класс объекта  $u$  определяется следующим образом:

$$a(u) = \operatorname{argmax}_{y \in Y} \sum_{i=1}^k [y(x_{(i;u)}) = y].$$

*Ближайшие объекты* – это объекты, расстояние от которых до данного объекта наименьшее по некоторой метрике  $\rho$ .

- В качестве метрики  $\rho$  как правило используют евклидово расстояние, но можно использовать и другие метрики.
- Перед использованием метода необходимо масштабировать данные, иначе признаки с большими числовыми значениями будут доминировать при вычислении расстояний.

### 3. ОТБОР ПРИЗНАКОВ

# VARIANCE THRESHOLD

- Можем удалить признаки, которые имеют очень маленькую дисперсию, т.е. практически константы.

# ОТБОР ПРИЗНАКОВ ПО КОРРЕЛЯЦИИ С ЦЕЛЕВОЙ ПЕРЕМЕННОЙ

- Для каждого признака вычислим его корреляцию с целевой переменной. Будем выкидывать признаки, имеющие маленькую корреляцию.

# БОЛЕЕ СЛОЖНЫЕ МЕТОДЫ

- **Filtration methods** (фильтрационные методы)
- **Wrapping methods** (оберточные методы)
- **Model selection** (встроенный в модель отбор признаков)

# 1. ФИЛЬТРАЦИОННЫЕ МЕТОДЫ

- **Фильтрационные методы - это отбор признаков по различным статистическим тестам.** Идея метода состоит в вычислении влияния каждого признака в отдельности на целевую переменную (с помощью вычисления некоторой статистики).

Очевидный плюс метода: скорость, так как мы вычисляем значения  $N$  статистик, где  $N$  - количество признаков.

# 1. ФИЛЬТРАЦИОННЫЕ МЕТОДЫ

В `sklearn` есть сразу несколько методов, использующих отбор по статистическим критериям. Среди них выделим следующие:

- **SelectKBest** - оставляет  $k$  признаков с наибольшим значением выбранной статистики
- **SelectPercentile** - оставляет признаки со значениями выбранной статистики, попавшими в заданную пользователем квантиль

# 1. СТАТИСТИЧЕСКИЕ ТЕСТЫ ДЛЯ ОТБОРА ПРИЗНАКОВ (ПРИМЕР)

- Тест  $\chi^2$  используется в статистике для проверки независимости двух событий.
- Поскольку  $\chi^2$  проверяет степень независимости между двумя переменными, а мы хотим сохранить только признаки, наиболее зависимые от метки, то будем вычислять  $\chi^2$  между каждым признаком и меткой, сохраняя только признаки с наибольшими значениями.
- Критерий  $\chi^2$  можем применять только для бинарных или порядковых признаков.

# 1. СТАТИСТИЧЕСКИЕ ТЕСТЫ ДЛЯ ОТБОРА ПРИЗНАКОВ (ПРИМЕР)

- Статистика  $\chi^2$  вычисляется по формуле

$$\chi^2(X; Y) = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}},$$

где  $O_{ij}$  - наблюдаемая частота,  $E_{ij}$  - ожидаемая частота.

Пример: хотим выявить влияние курения на гипертонию:

	Артериальная гипертония есть (1)	Артериальной гипертонии нет (0)	Всего
Курящие (1)	40	30	70
Некурящие (0)	32	48	80
Всего	72	78	150

Вычисляем  $\chi^2$ :  $\chi^2 = (40-33.6)^2/33.6 + (30-36.4)^2/36.4 + (32-38.4)^2/38.4 + (48-41.6)^2/41.6 = 4.396$ .

# 1. СТАТИСТИЧЕСКИЕ ТЕСТЫ ДЛЯ ОТБОРА ПРИЗНАКОВ

- Статистика  $\chi^2$  вычисляется по формуле

$$\chi^2(X; Y) = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}},$$

где  $O_{ij}$  - наблюдаемая частота,  $E_{ij}$  - ожидаемая частота.

Пример: хотим выявить влияние курения на гипертонию:

	Артериальная гипертония есть (1)	Артериальной гипертонии нет (0)	Всего
Курящие (1)	40	30	70
Некурящие (0)	32	48	80
Всего	72	78	150

Вычисляем  $\chi^2$ :  $\chi^2 = (40-33.6)^2/33.6 + (30-36.4)^2/36.4 + (32-38.4)^2/38.4 + (48-41.6)^2/41.6 = 4.396$ .

При отборе признаков оставляем  $k$  (или заданную квантиль) признаков с наибольшим значением  $\chi^2$ .

# 1. СТАТИСТИЧЕСКИЕ ТЕСТЫ ДЛЯ ОТБОРА ПРИЗНАКОВ

- mutual information:

для векторов  $X$  и  $Y$  статистика вычисляется по формуле

$$I(X; Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \log\left(\frac{p(x, y)}{p(x)p(y)}\right)$$

- хи-квадрат:

$$\chi^2(X; Y) = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}},$$

где  $O_{ij}$  - наблюдаемая частота,  $E_{ij}$  - ожидаемая частота.

## 2. ОБЕРТОЧНЫЕ МЕТОДЫ

Оберточные методы используют **жадный отбор признаков**, т.е. последовательно выкидывают наименее подходящие по мнению методов признаки.

В `sklearn` есть оберточный метод - **Recursive Feature Elimination (RFE)**.

Параметры метода:

- a) алгоритм, используемый для отбора признаков (например, `RandomForest`)
- b) число признаков, которое мы хотим оставить.

## 2. ЖАДНЫЙ ОТБОР ПРИЗНАКОВ

1 шаг: Перебираем все признаки и убираем тот, удаление которого сильнее всего уменьшает ошибку

2 шаг: Из оставшихся признаков убираем тот, удаление которого сильнее всего уменьшает ошибку

И т.д.

### 3. ВСТРОЕННЫЕ В МОДЕЛЬ МЕТОДЫ

*Напоминание:*  $L_1$ -регуляризация умеет отбирать признаки.

$$Q(w) + \alpha \sum_{i=1}^d |w_j| \rightarrow \min_w$$

### 3. ВСТРОЕННЫЕ В МОДЕЛЬ МЕТОДЫ

*Напоминание:*  $L_1$ -регуляризация умеет отбирать признаки.

$$Q(w) + \alpha \sum_{i=1}^d |w_j| \rightarrow \min_w$$

Рассмотрим другой вариант регуляризации, которая тоже умеет отбирать признаки ( $L_0$ -регуляризация):

$$Q(w) + \alpha \sum_{i=1}^d [w_j \neq 0] \rightarrow \min_w$$

### 3. ИНФОРМАЦИОННЫЕ КРИТЕРИИ

- **Информационный критерий** - мера качества модели, учитывающая степень «подгонки» модели под данные с корректировкой (штрафом) на используемое количество параметров.
- Информационные критерии основаны на **компромиссе между точностью и сложностью модели**. Критерии различаются тем, как они обеспечивают этот баланс.

### 3. КРИТЕРИЙ AIC

#### Критерий Акаике (AIC, Akaike Information Criterion)

- Дополнительно предполагаем, что модель  $a$  – линейная.

$$AIC(a, X) = Q(a, X) + \frac{2\hat{\sigma}^2}{l}n \rightarrow \min$$

$Q$  – функционал ошибки

$\hat{\sigma}^2$  - оценка дисперсии ошибки  $D(y_i - a(x_i))$

$n$  – количество используемых признаков

$l$  – число объектов

### 3. КРИТЕРИЙ AIC

#### Критерий Акаике (AIC, Akaike Information Criterion)

- Дополнительно предполагаем, что модель  $a$  – линейная.

$$AIC(a, X) = Q(a, X) + \frac{2\hat{\sigma}^2}{l}n \rightarrow \min$$

$Q$  – функционал ошибки

$\hat{\sigma}^2$  - оценка дисперсии ошибки  $D(y_i - a(x_i))$

$n$  – количество используемых признаков

$l$  – число объектов

- Если  $Q$  – среднеквадратичная ошибка для линейной регрессии, и шумы нормально распределены, то

$$AIC = -\ln P + n$$

### 3. КРИТЕРИЙ BIC

Критерий Шварца (BIC, Bayesian Information Criterion)

$$BIC(a, X) = \frac{l}{\hat{\sigma}^2} (Q(a, X) + \frac{\hat{\sigma}^2 \ln l}{l} n) \rightarrow \min$$

- Если  $Q$  – среднеквадратичная ошибка для линейной регрессии, и шумы нормально распределены, то

$$BIC = -\ln P + \frac{n}{2} \ln l$$

### 3. ОТБОР ПРИЗНАКОВ С ПОМОЩЬЮ ИНФОРМАЦИОННЫХ КРИТЕРИЕВ

- Если в модели  $k$  признаков (регрессоров), то существует  $2^k$  всевозможных моделей
- В идеале необходимо построить все  $2^k$  моделей, для каждой посчитать значение критерия качества (AIC, BIC) и выбрать модель, лучшую по этому критерию
- При большом количестве регрессоров используют метод включений-исключений (жадный отбор признаков)

### 3. ПРИМЕР

Задача предсказания уровня преступности в разных штатах по следующим признакам:

Регрессор

Нулевой коэффициент

Возраст

Южный штат(да/нет)

Образование

Расходы

Труд

Количество мужчин

Численность населения

Безработные (14-24)

Безработные (25-39)

Доход

### 3. ПРИМЕР: ОТБОР ПРИЗНАКОВ ПО AIC

- Мы решаем задачу линейной регрессии с предположением, что ошибки нормально распределены, поэтому  $AIC = \ln P(a, X) - n \rightarrow \max$ .

В модели с полным набором регрессоров  $AIC = -310.37$ . В порядке убывания AIC при удалении каждой из переменных равен:

Численность населения ( $AIC = -308$ ), Труд ( $AIC = -309$ ), Южный штат ( $AIC = -309$ ), Доход ( $AIC = -309$ ), Количество мужчин ( $AIC = -310$ ), Безработные I ( $AIC = -310$ ), Образование ( $AIC = -312$ ), Безработные II ( $AIC = -314$ ), Возраст ( $AIC = -315$ ), Расходы ( $AIC = -324$ ).

Таким образом, имеет смысл удалить переменную “Население”.

### 3. ПРИМЕР: ОТБОР ПРИЗНАКОВ ПО AIC

Южный штат (AIC = -308), Труд (AIC = -308), Доход (AIC = -308), Количество мужчин (AIC = -309), Безработные I (AIC = -309), Образование (AIC = -310), Безработные II (AIC = -313), Возраст (AIC = -313), Расходы (AIC = -329).

Удаляем переменные до тех пор, пока не удастся больше получить увеличения AIC.

Уровень преступности = 1.2 Возраст + 0.75 Образование + 0.87  
Расходы + 0.34 Количество мужчин – 0.86 Безработные I + 2.31  
Безработные II.