Машинное обучение в гидрологии

Классификация

Содержание лекции

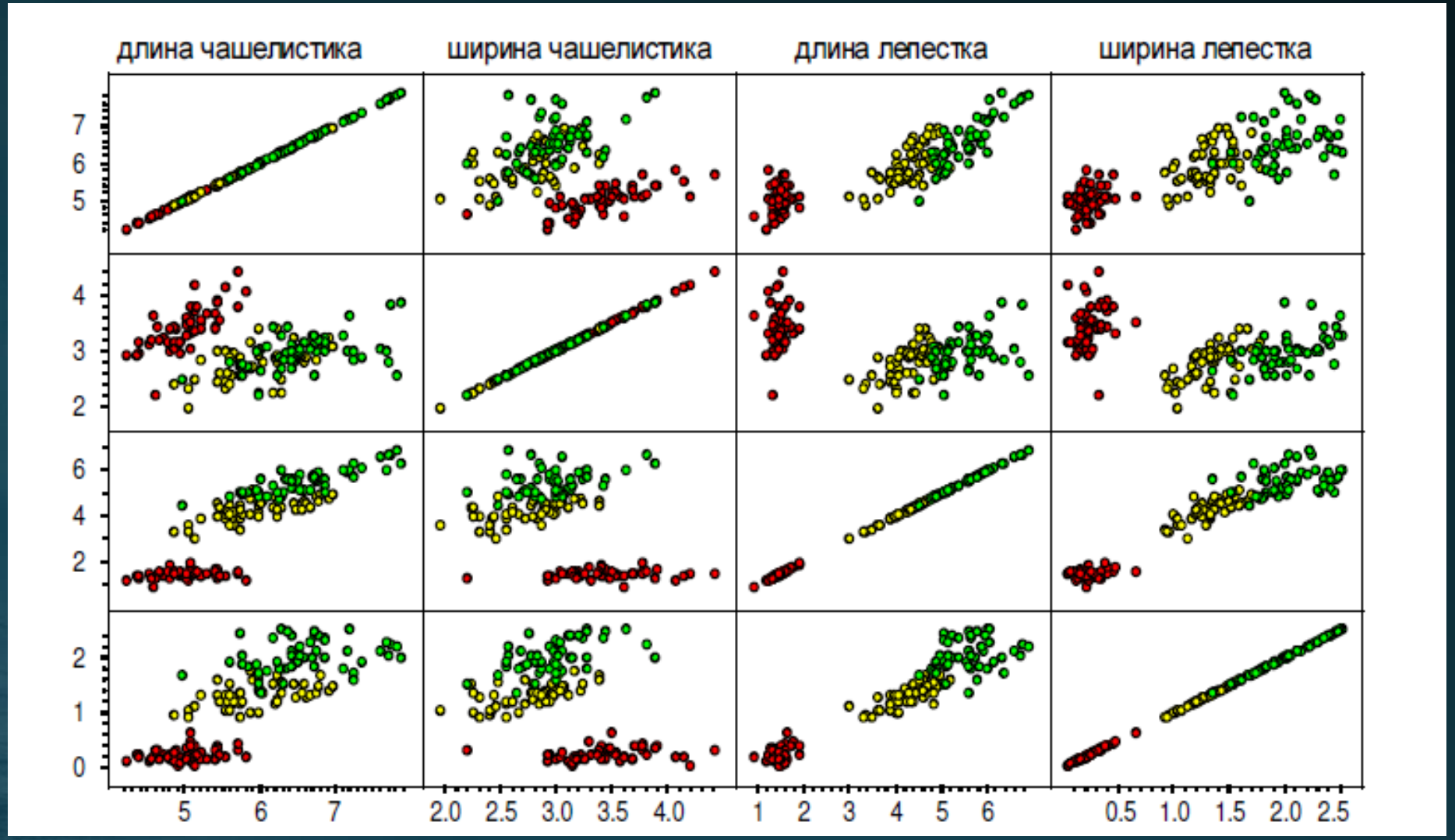
1. Метрические методы классификации
   - Метод ближайшего соседа
   - KNN – k ближайших соседей
   - Алгоритм k взвешенных ближайших соседей
2. Критерии качества
3. Логические методы классификации
   - Решающие деревья
   - Критерии качества
   - Алгоритмы построения дерева

Метрические классификаторы

- Алгоритмы, основанные на оценке сходства между объектами
- «Гипотеза компактности» - близкие объекты как правило лежат в одном классе

- Для формализации понятия сходства вводится функция расстояния между объектами (метрика) - $\rho(x, x')$

Задача классификации цветов ириса (1936г.)

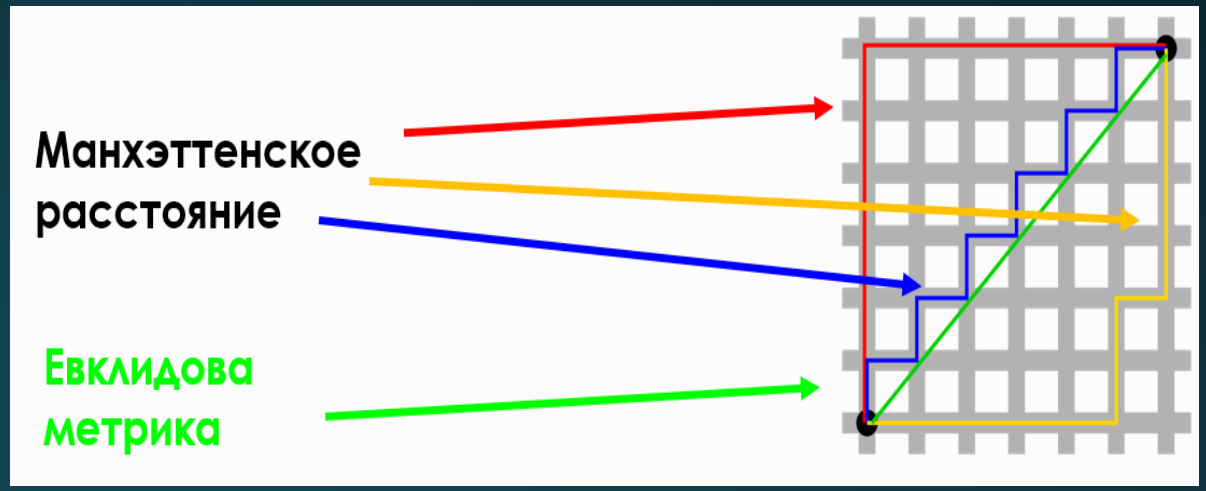


Задание метрики

- Примеры метрик
  - Евклидова метрика $\rho(x, y) = \sqrt{\sum(x_i - y_i)^2}$
  - Манхэттенское расстояние $\rho(x, y) = \sum |x_i - y_i|$
  - Расстояние Чебышева $\rho(x, y) = \max_i |x_i - y_i|$

$(x_1, \dots, x_n)$ – вектор признаков объекта x

$(y_1, \dots, y_n)$ – вектор признаков объекта y



Масштабирование признаков

Что это такое и зачем нужно?

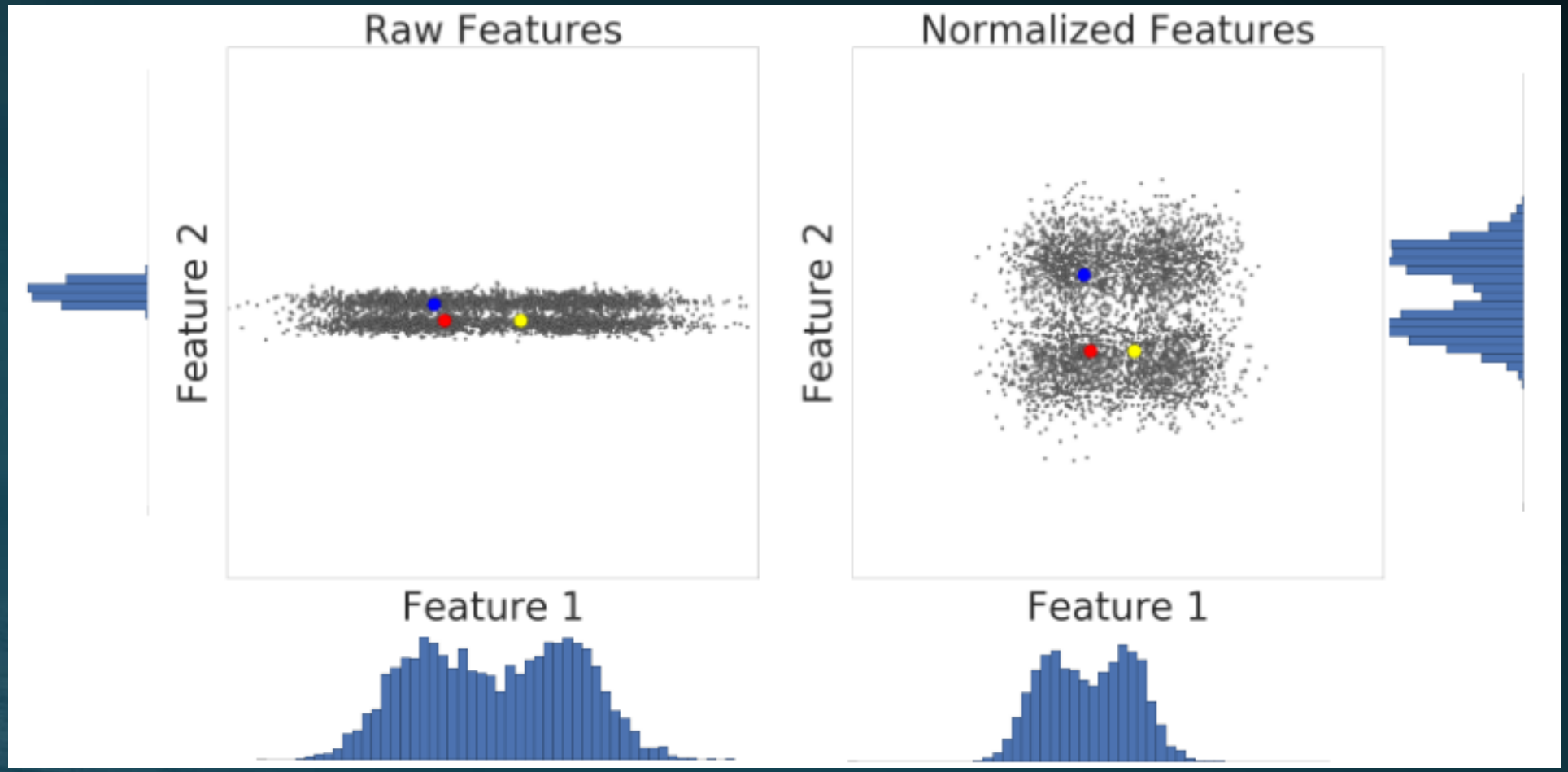
Масштабирование признаков

Метрические алгоритмы чувствительны к масштабированию данных.

Чтобы все признаки вносили одинаковый вклад в метрику их необходимо масштабировать.

- Нормализация – замена признаков так, чтобы каждый лежал в диапазоне от 0 до 1
- Стандартизация – предобработка данных, после которой признак имеет среднее 0 и дисперсию 1

Масштабирование признаков

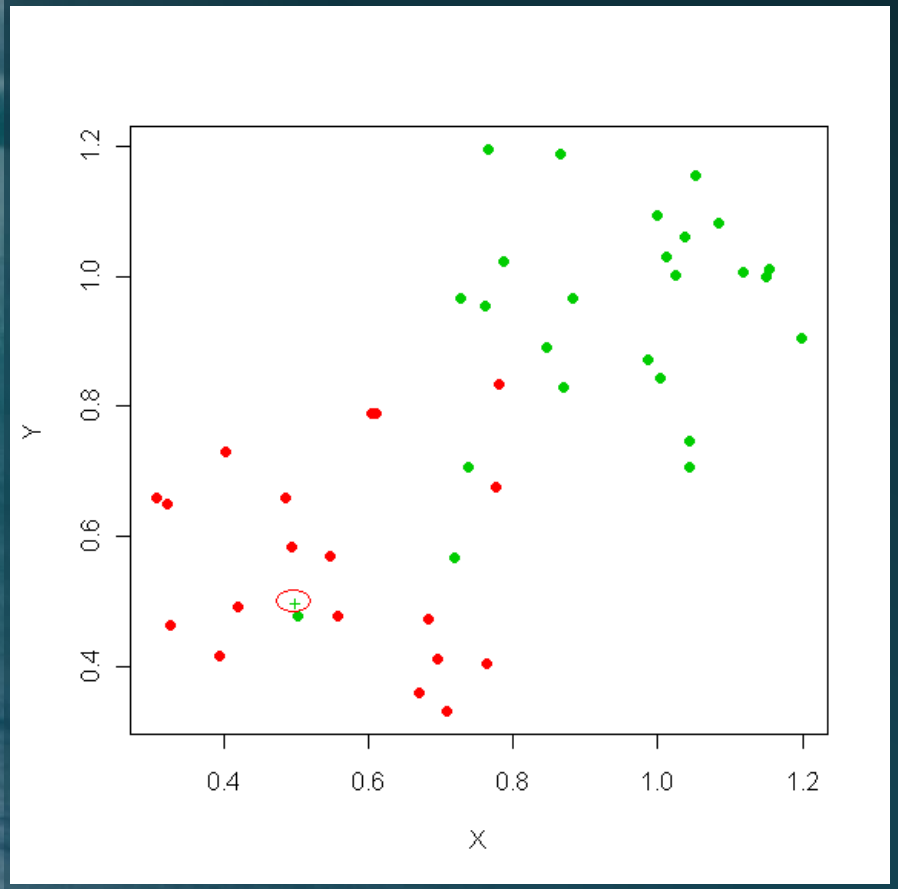


Метод ближайшего соседа

- Один из самых простейших алгоритмов классификации
- 1 параметр модели: метрика
- Алгоритм следующий:

По заданной метрике ищем ближайший объект в обучающей выборке и классифицируем объект точно так же

Метод ближайшего соседа



Преимущества:

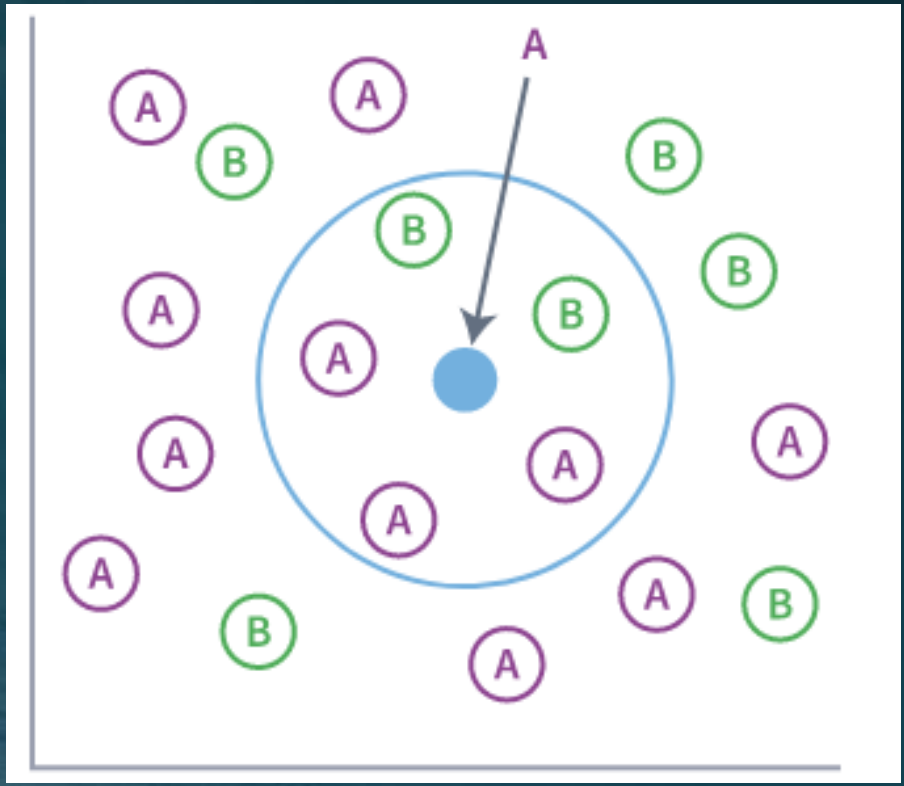
- Простая реализация
- Хорошая интерпретируемость

Недостатки:

- Неустойчивость к выбросам
- Неоднозначность классификации при равных расстояниях до двух ближайших объектов
- Необходимо хранить всю выборку
- Алгоритм поиска может быть вычислительно сложен
- Не учитывается значение расстояния

Что же делать?

Метод k ближайших соседей (k Nearest Neighbors, KNN)

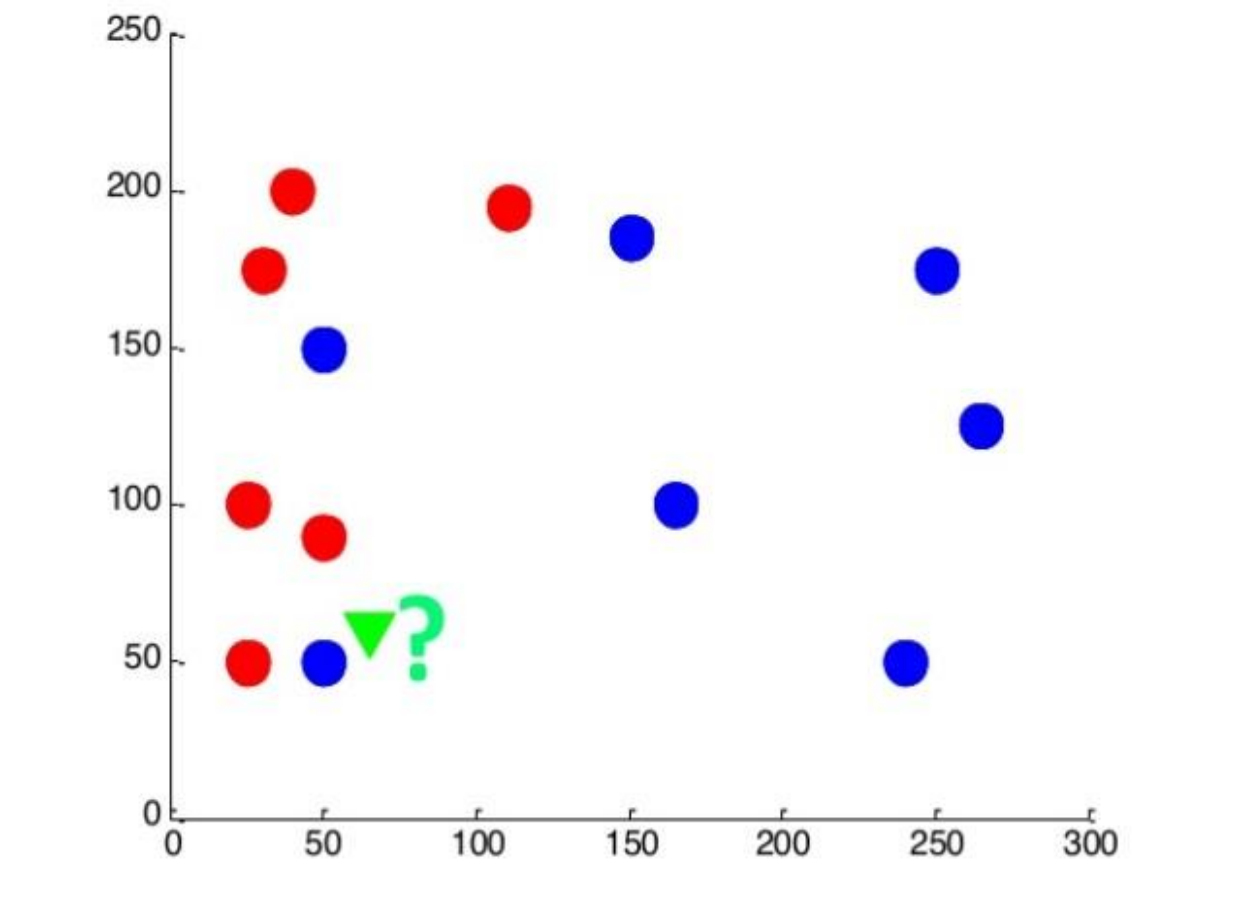


- Параметры модели:
  - Метрика
  - Количество соседей k

- Алгоритм следующий:

По заданной метрике ищем k ближайших «соседей» в обучающей выборке и классифицируем объект как большинство из k соседей

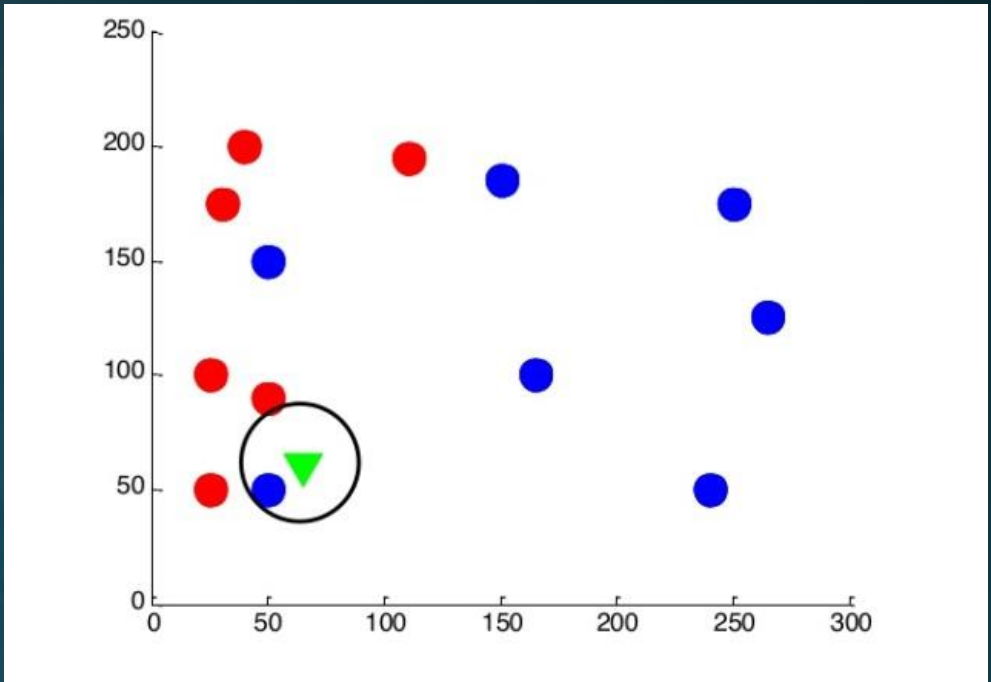
Сравнение двух методов



Сделаем прогноз для каждой точки на плоскости.

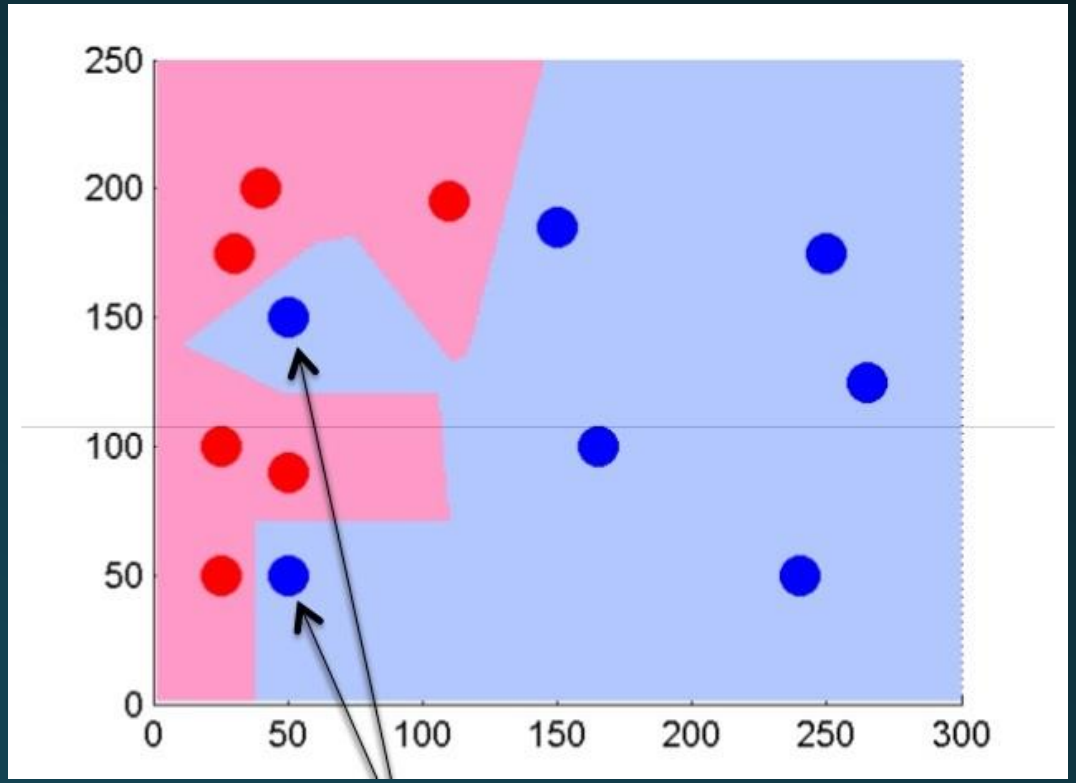
Как будет работать метод ближайшего соседа?

Сравнение двух методов

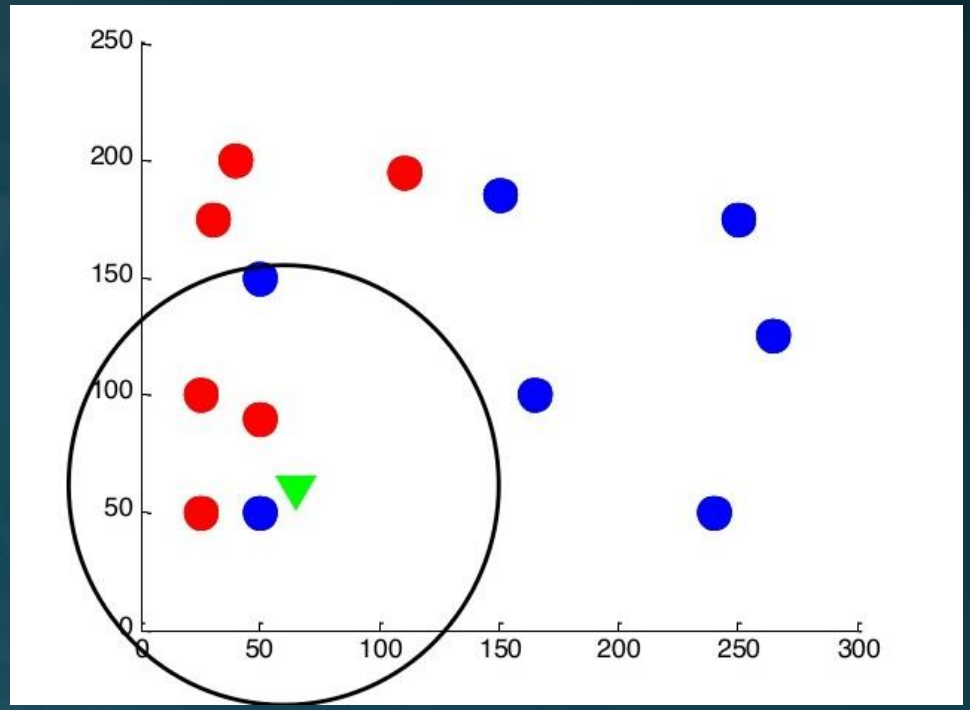


Получается две области: в одной велика вероятность появления красных точек, а в другой синих.

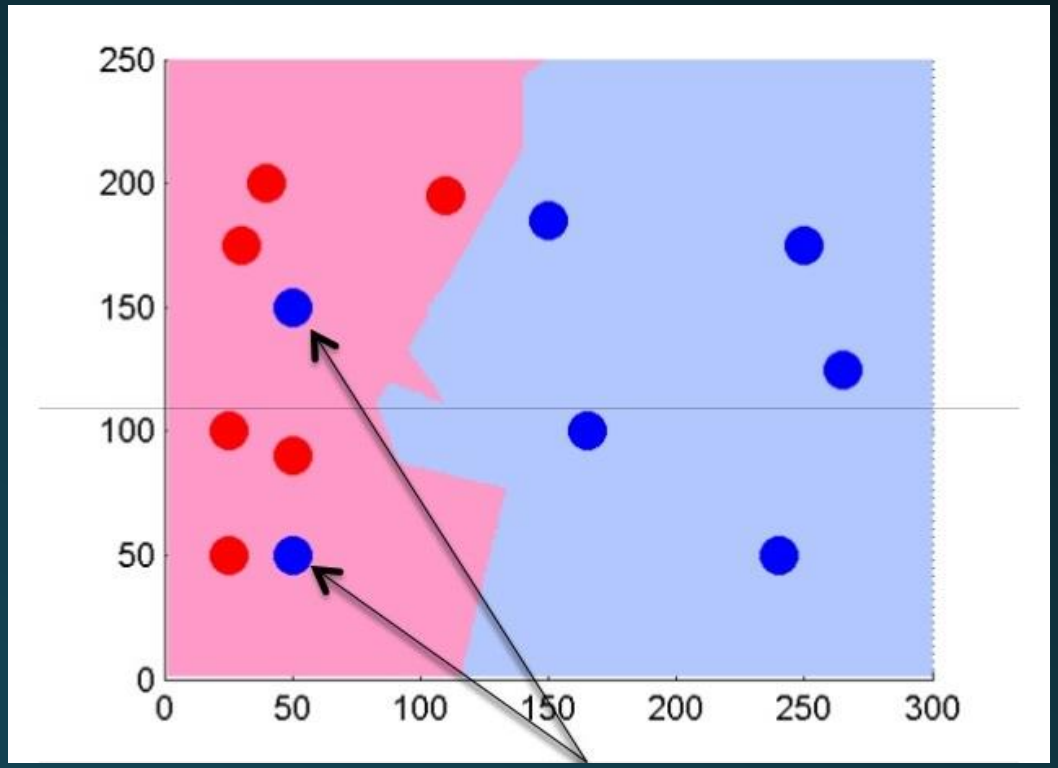
Чувствительность к выбросам



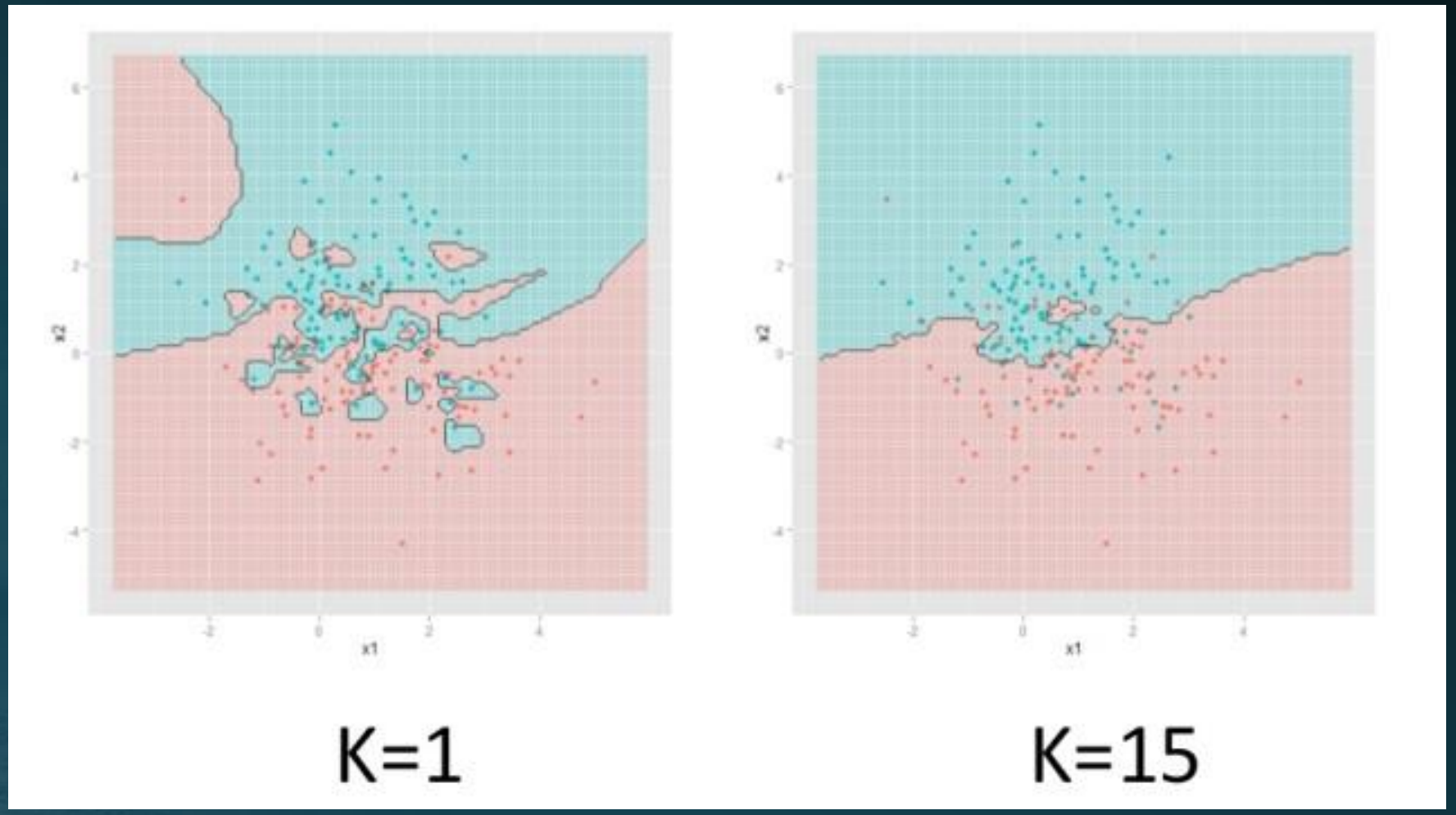
Сравнение двух методов



В случае пяти соседей области отсекаются наиболее шумовые объекты и получается более ровная граница разделения классов

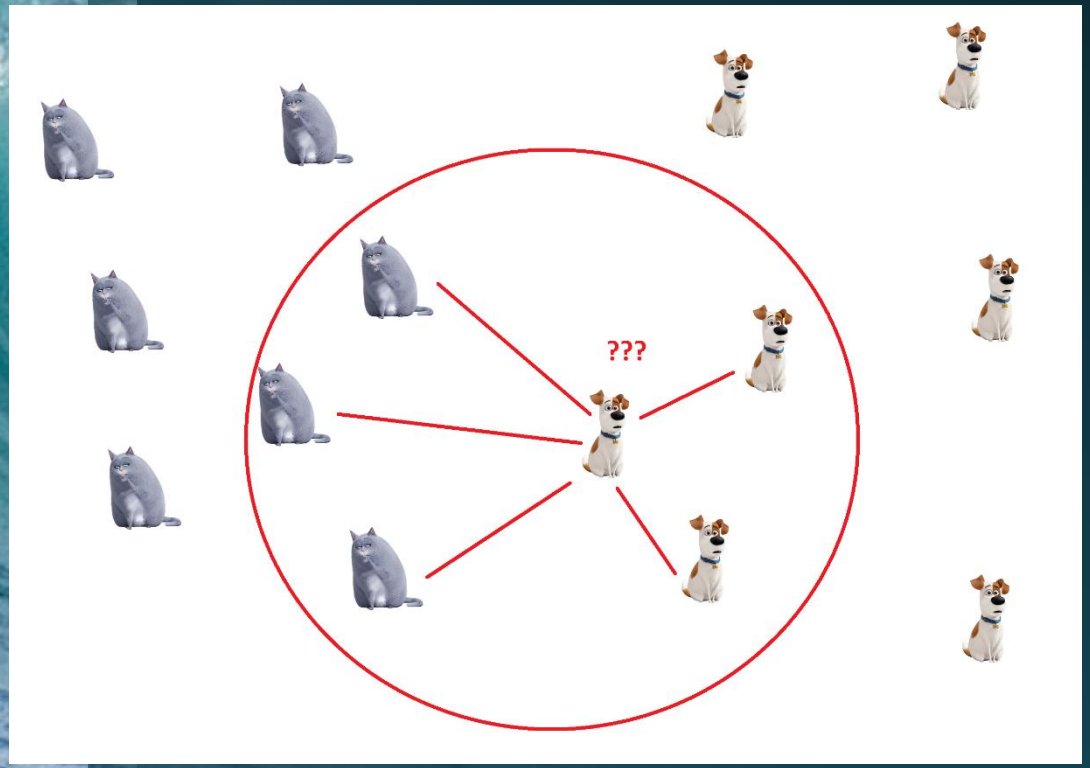


Сравнение двух методов



Вот так будет выглядеть разделение на большем количестве объектов, полученных путем нормального распределения

Метод k ближайших соседей



Преимущества:

- Простая реализация
- Хорошая интерпретируемость
- Возможность оптимизации параметра k

Недостатки:

- ~~Неустойчивость к выбросам~~
- ~~Неоднозначность классификации при равных расстояниях до двух ближайших объектов~~
- Необходимо хранить всю выборку
- Алгоритм поиска может быть вычислительно сложен
- **Не учитывается значение расстояния**

Метод k ближайших взвешенных соседей

- Параметры модели:
  - Метрика
  - Количество соседей k
  - **Веса**

- Алгоритм следующий:

По заданной метрике ищем k ближайших «соседей» в обучающей выборке и классифицируем объект **взвешенным голосованием** (например, сосед, находящийся ближе всего имеет больший вес)

Как выбирать веса?

- Каждый объект имеет фиксированный вес
- Веса в зависимости от порядкового номера соседа:
  - Линейно убывающие веса
  - Экспоненциально убывающие веса
  - Любая невозрастающая функция от порядкового номера
- Веса зависят от расстояния:
  - Любая невозрастающая функция от расстояния до соседа

Критерии качества. Как оценивать качество модели?

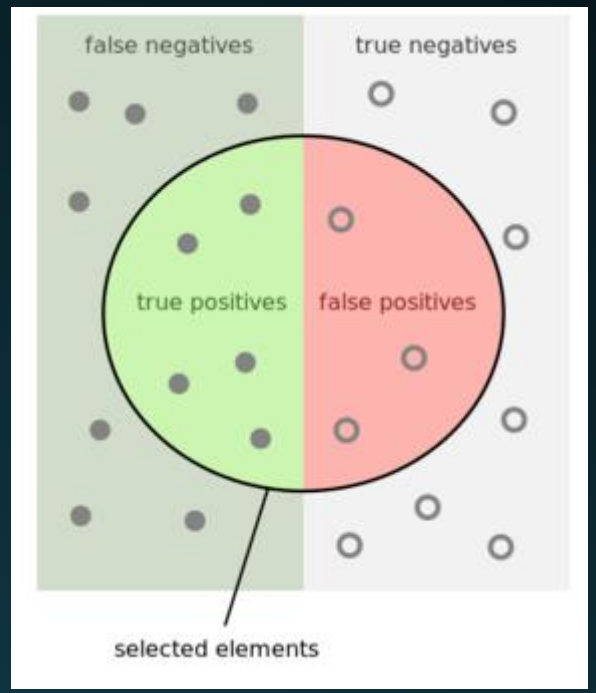
- Какие бывают метрики?
- Что важно при выборе метрики?
- Что может пойти не так?

Критерии качества. Как оценивать качество модели?

Бинарная модель. Y = {+1, -1}

Класс с “+1” – positive
Класс с “-1” – negative

Составляется матрица ошибок.



	Y = 1	Y = -1
$\hat{Y} = 1$	True Positive (TP)	False Positive (FP)
$\hat{Y} = -1$	False Negative (FN)	True Negative (TN)

Ошибка 2 рода
«пропуск цели»

Ошибка 1 рода
«ложная тревога»

Матрица ошибок

Простейшая метрика качества

Простейшая метрика качества – доля правильных ответов

$$accuracy = \frac{\text{правильные ответы}}{\text{все ответы}} = \frac{TP + TN}{TP + FP + FN + TN}$$

Простейшая метрика качества

Простейшая метрика качества – доля правильных ответов

$$accuracy = \frac{\text{правильные ответы}}{\text{все ответы}} = \frac{TP + TN}{TP + FP + FN + TN}$$

Не учитывается дисбаланс классов.

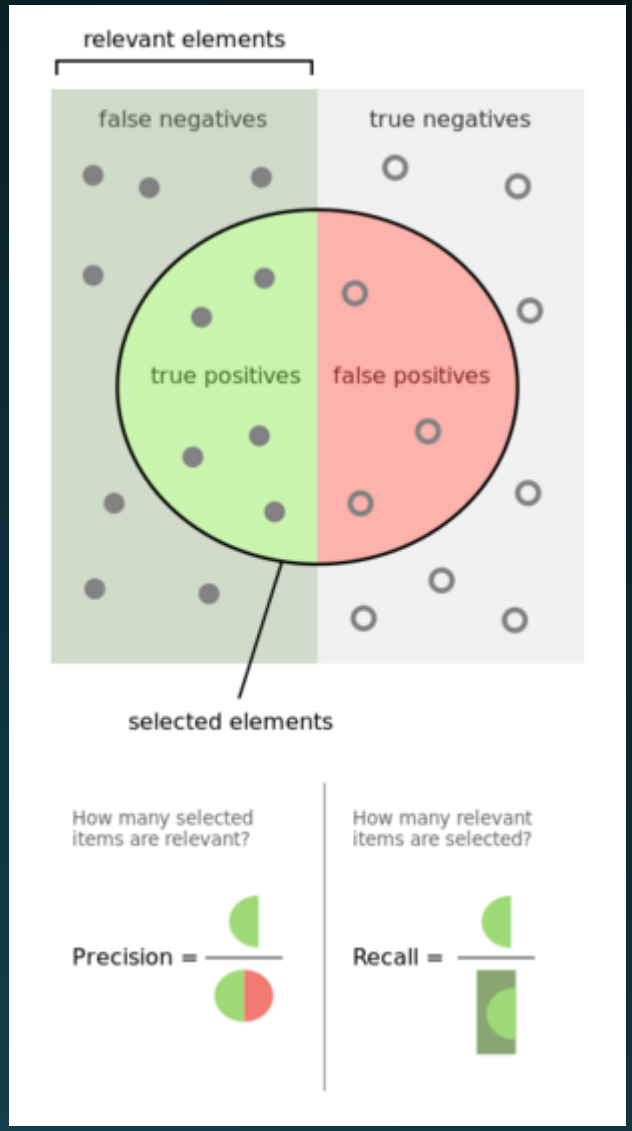
Лучше перейти от общего для всех классов метрики к отдельным показателям качества классов

Метрики по отклику алгоритма

$$\text{Precision} = \frac{TP}{TP + FP} = PPV - positive\ predictive\ value$$

$$Recall = \frac{TP}{TP + FN} = TPR - true\ positive\ rate$$

- Recall (полнота) позволяет следить за тем, чтобы было мало пропусков, однако ничего не говорит о ложных тревогах (можно использовать, если высока цена пропуска, а ложной тревоги маленькая - маленькая)
- Precision(точность) следит за тем, чтобы было мало ложных срабатываний, но не дает информации о пропусках (можно использовать, если высока цена ложной тревоги, а пропуска - маленькая)

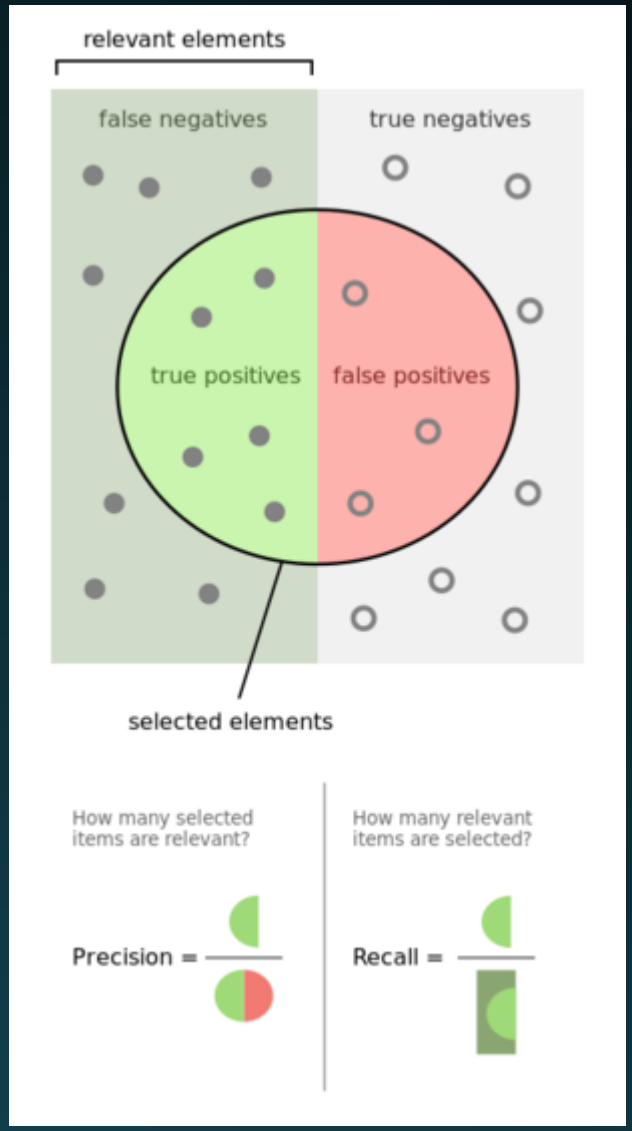


Метрики по отклику алгоритма

$$\text{Precision} = \frac{TP}{TP + FP} = PPV - positive\ predictive\ value$$

$$Recall = \frac{TP}{TP + FN} = TPR - true\ positive\ rate$$

Зачастую ставят задачу минимизации одной метрики при фиксированной второй.



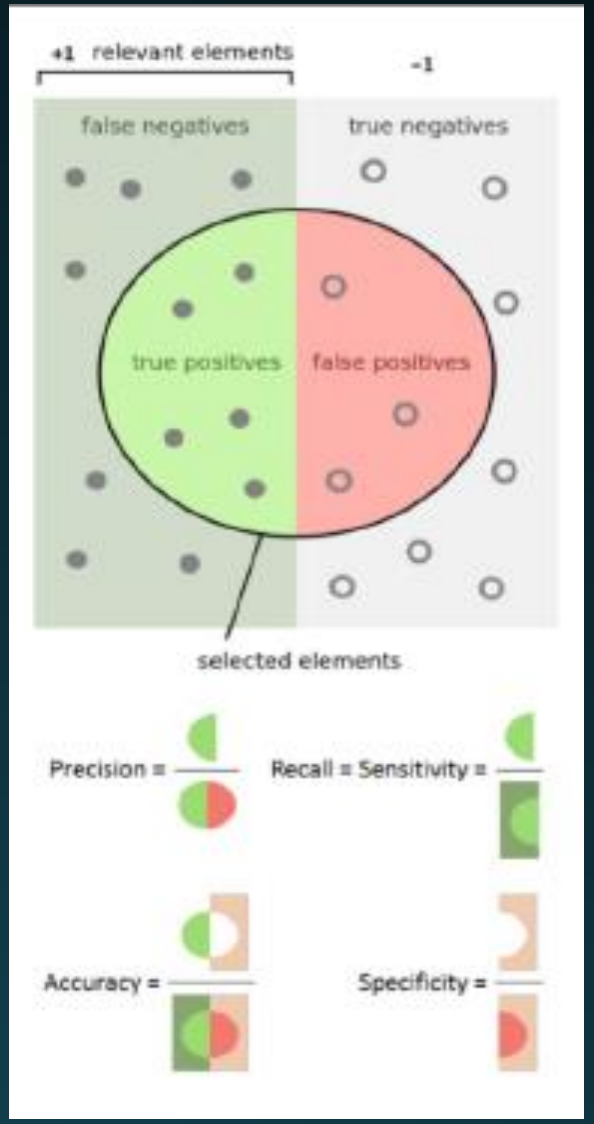
Метрики по отклику алгоритма

Еще одна важная метрика

$$Specificity = \frac{TN}{TN + FN} = TNR - true\ negative\ rate$$

Позволяет максимизировать количество верных отрицательных «диагнозов» (в случае, если стоимость лечения высокая, а цена пропуска низкая).

Чаще используются Precision и Recall

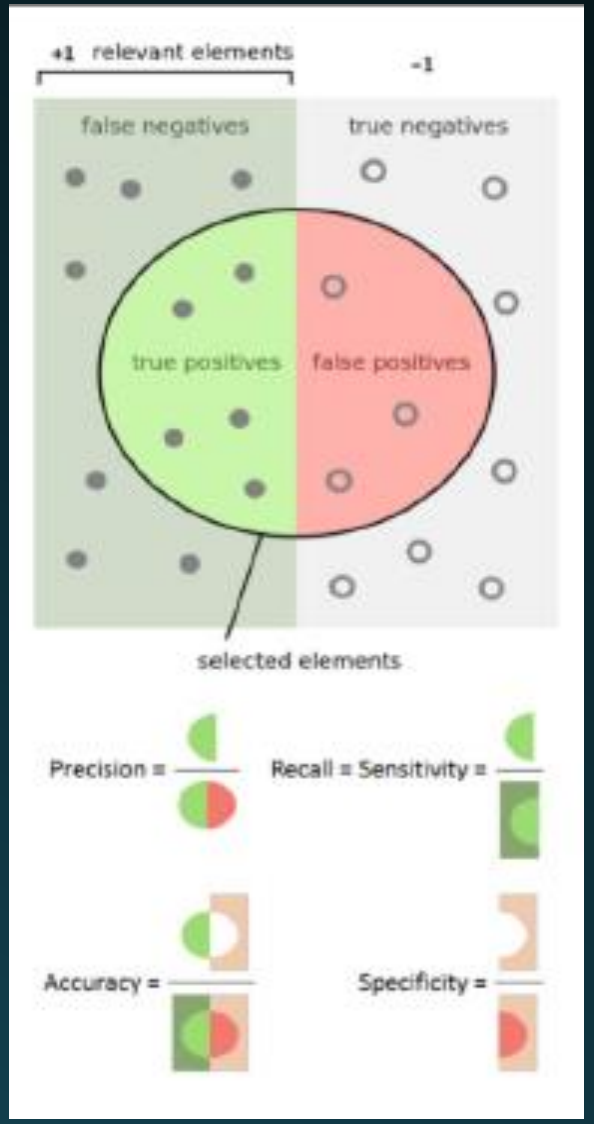


Метрики по отклику алгоритма

Чтобы как-то учесть и Precision и Recall вводится агрегированная метрика качества:

$$F_1 = \frac{2}{\frac{1}{precision}+\frac{1}{recall}} = \frac{2*recall*precision}{precision+recall}$$

Это гармоническое среднее, которое стремится к 0, когда хотя бы одно из значений стремится к 0.



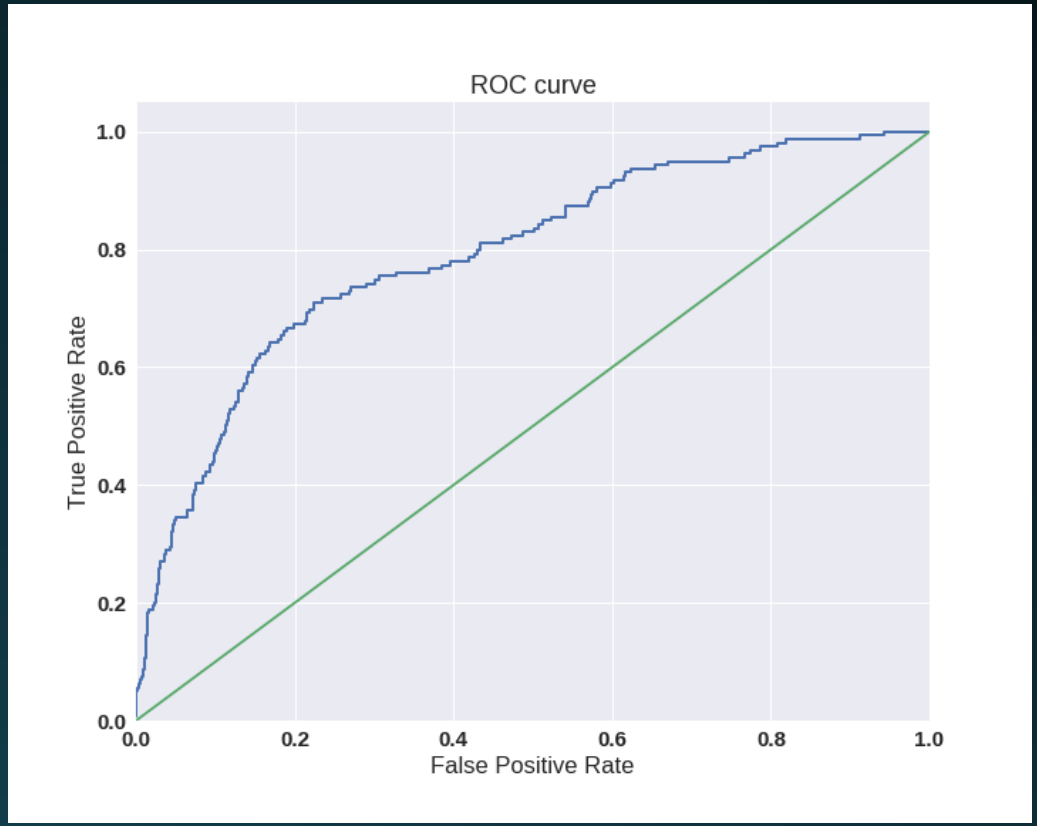
ROC кривая. ROC-AUC

При конвертации вещественного ответа алгоритма (как правило вероятности принадлежности к классу) в бинарную метрику, мы должны выбрать порог, при котором 0 становится 1.

Один из способов оценить модель не привязываясь к конкретному порогу является AUC-ROC (или ROC-AUC) – площадь (Area Under Curve) под кривой ошибок (Receiver Operating Characteristic curve)

ROC кривая

ROC-кривая – линия от (0,0) до (1,1) в координатах True Positive Rate и False Positive Rate



- *Доля верных положительных классификаций*

$$TPR = \frac{TP}{TP + FN}$$

- *Доля ложных положительных классификаций*

$$\mathrm{FPR} = \frac{FP}{FP + TN}$$

Каждая точка на кривой соответствует некоторому порогу

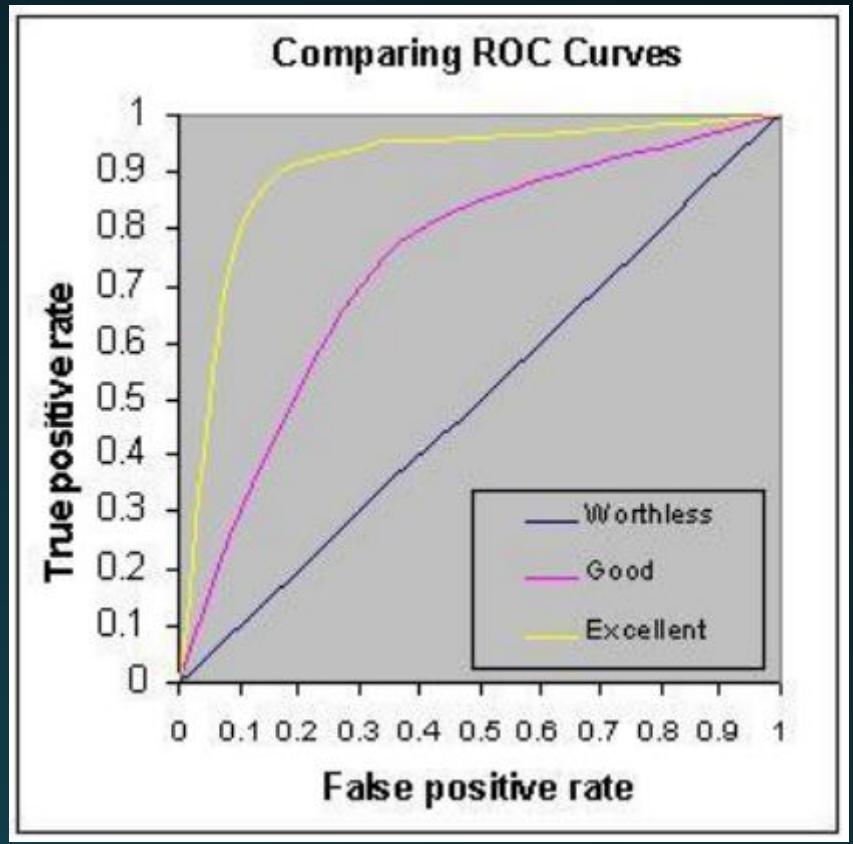
Идеальный случай – TPR = 1, FPR = 0. И площадь под кривой будет 1.

AUC-ROC

Разные кривые можно сравнивать по площади, которую они ограничивают (AUC-ROC).

- 0.5 -- случайный классификатор
- 1.0 -- идеальный классификатор

В промежутке между ними реальные модели

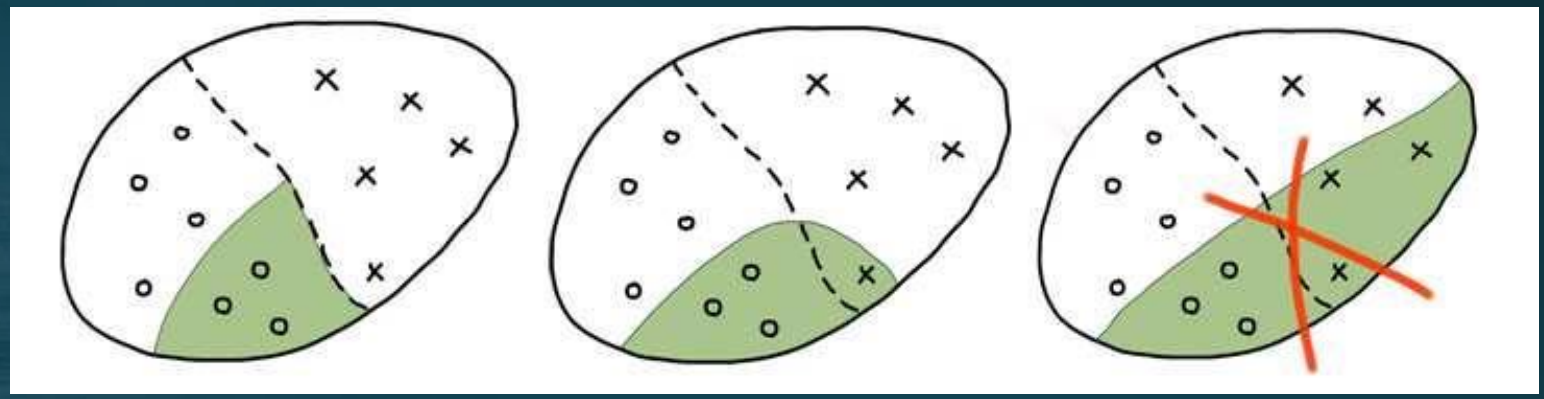


Логические классификаторы

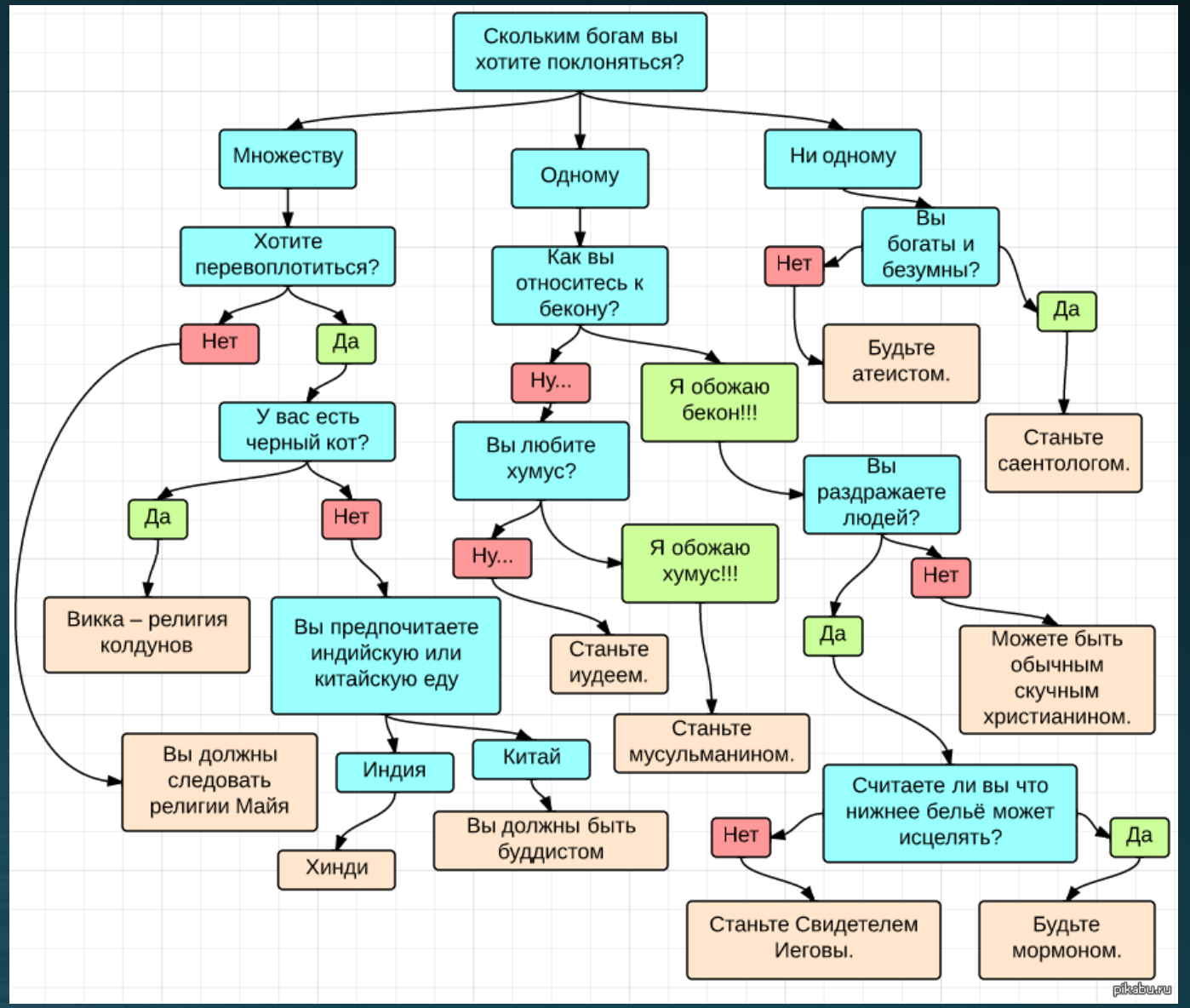
Основная идея – смоделировать логику, по которой человек принимает решения.

Логические классификаторы достаточно понятные и легко интерпретируемые.

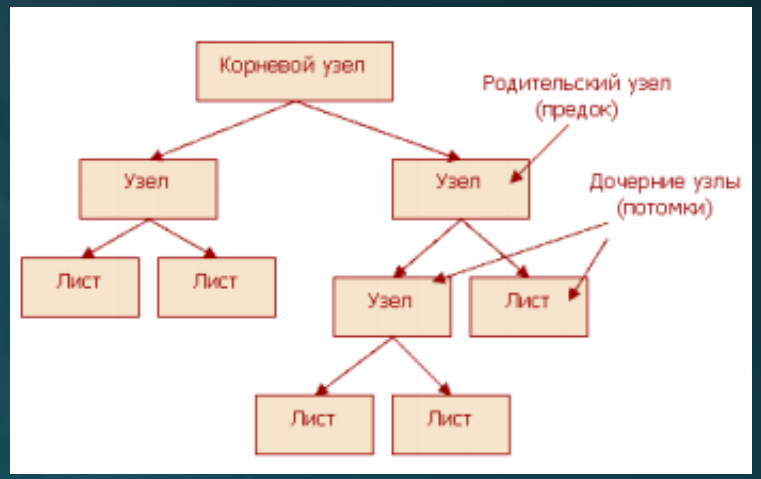
Нужно получить не только решающее правило, но и понять насколько оно разумно и какие логические закономерности были сформированы



Решающие деревья

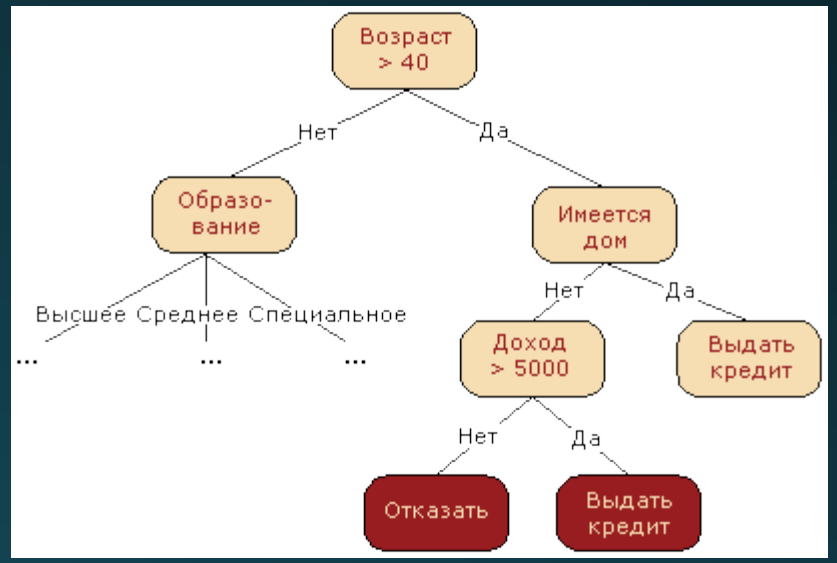


Решающие деревья



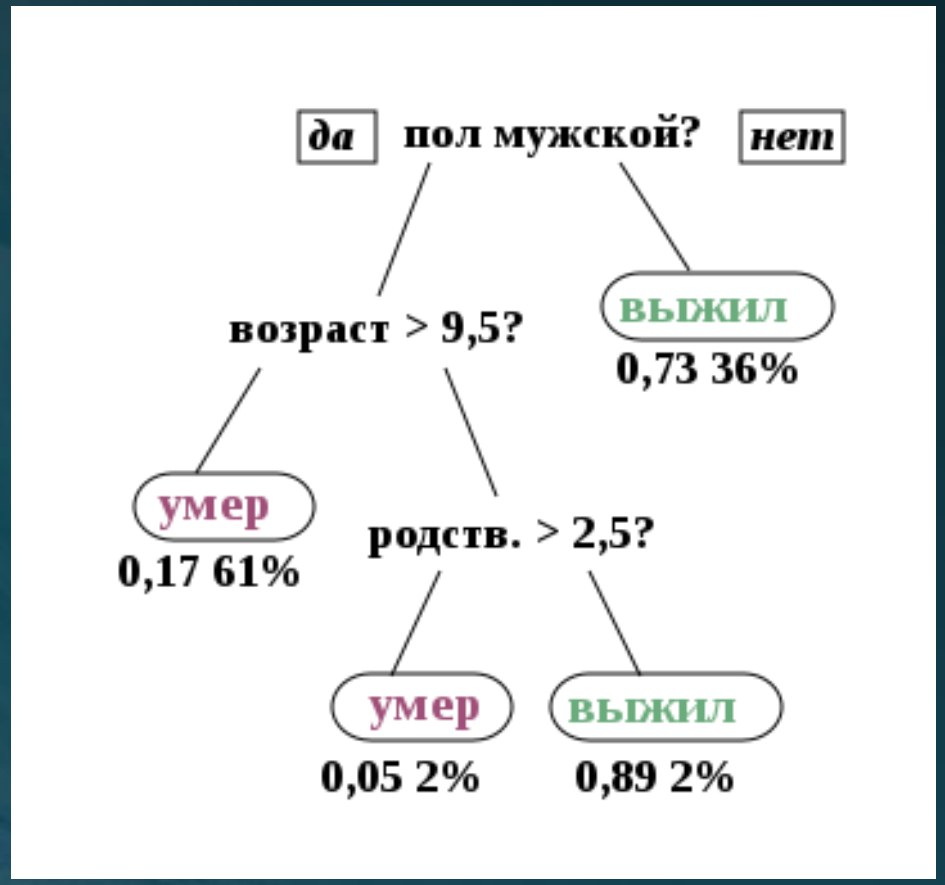
Строение дерева

- Решающие деревья – наглядная инструкция, что делать в той или иной ситуации
- Достаточно ясно отражают процесс принятия решения



Выдача кредита

Решающие деревья

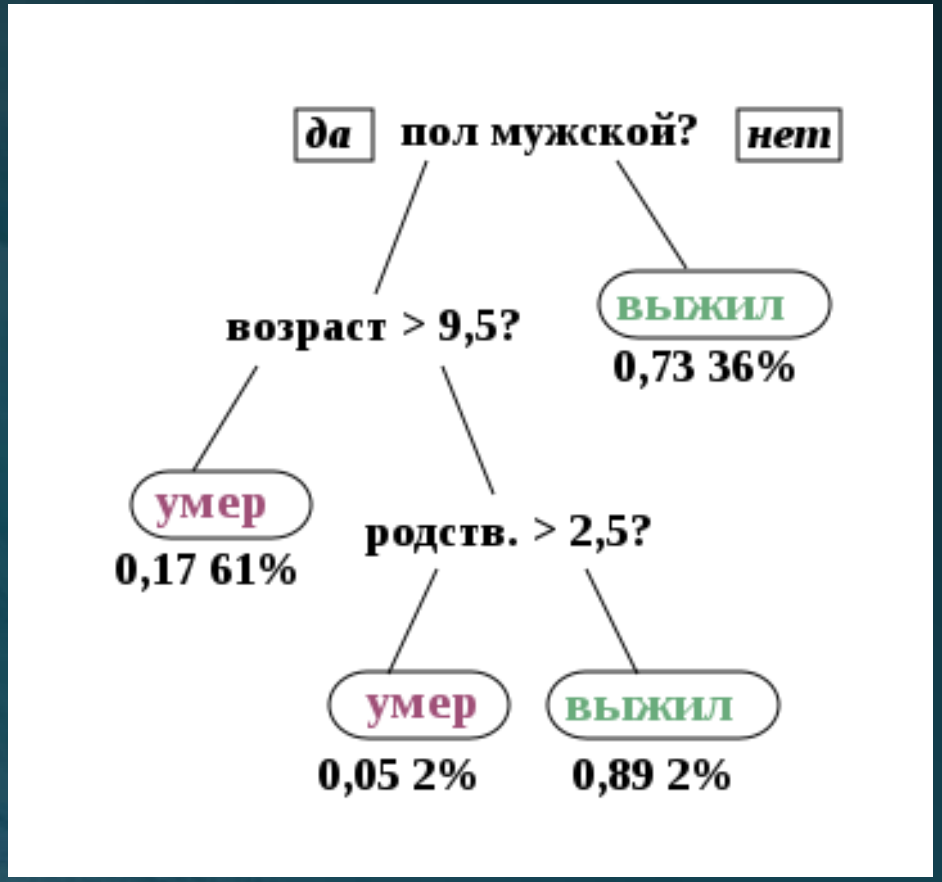


Пассажиры титаника

- Бинарное дерево (да/нет)
- В каждой внутренней вершине записано условие
- В каждом листе записан прогноз

Мы берем классифицируемый объект и начинаем двигаться по дереву либо влево либо вправо в зависимости от ответа. В итоге попадаем в лист, который дает прогноз, который и выдается в качестве ответа

Решающие деревья



Пассажиры титаника

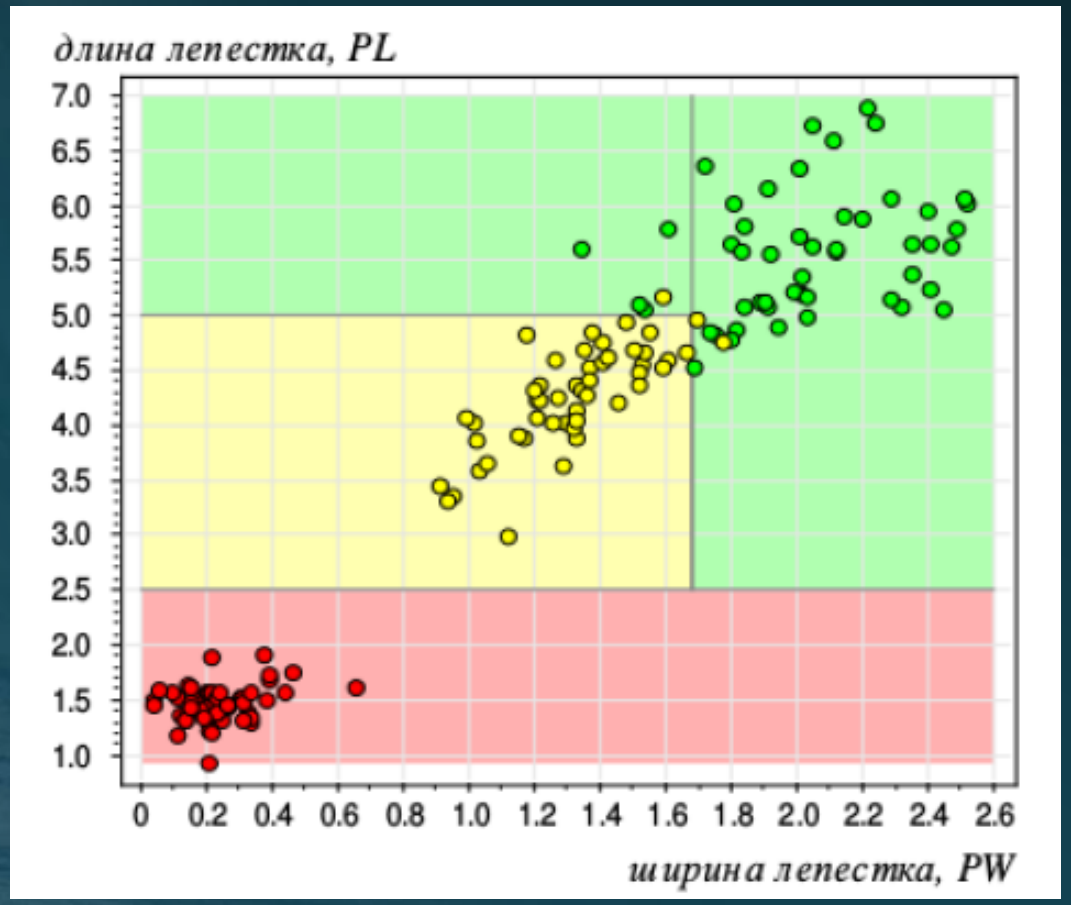
Условия:

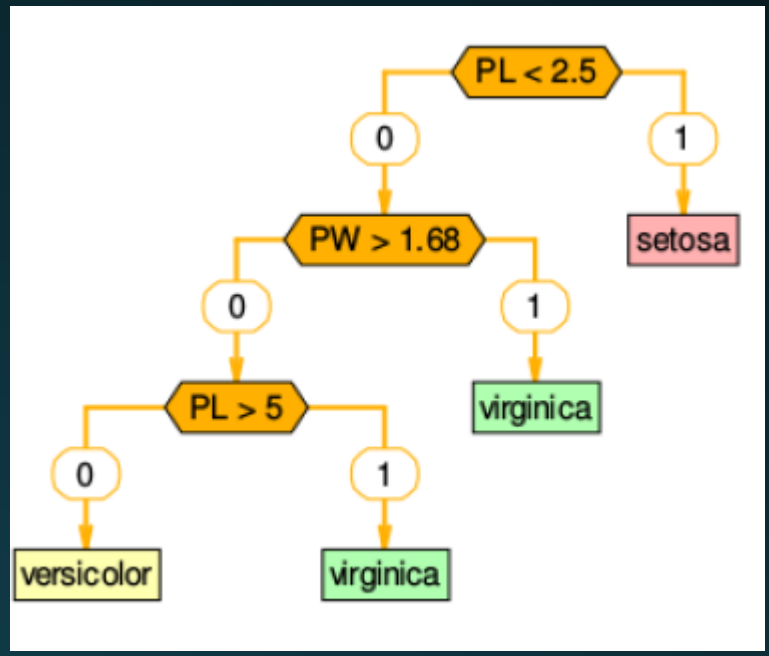
Самый популярный вариант:
$[x_j \leq t]$ – условие зависит всего от одного признака

В качестве результат лист может выдавать как принадлежность к тому или иному классу, так и вероятность принадлежности к классам.

Пример решающего дерева

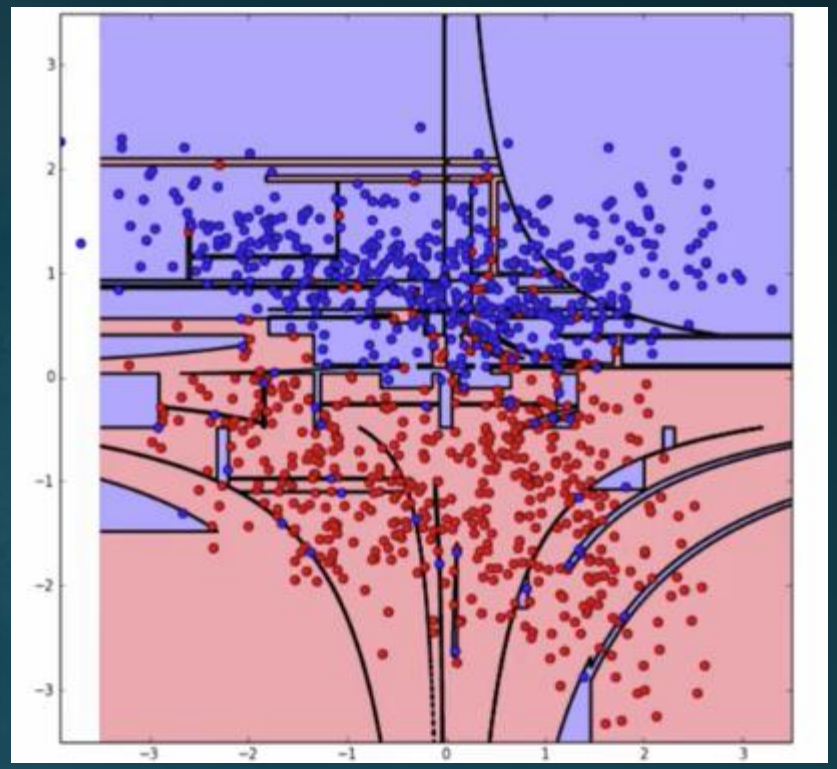
Задача Фишера о классификации листков ириса

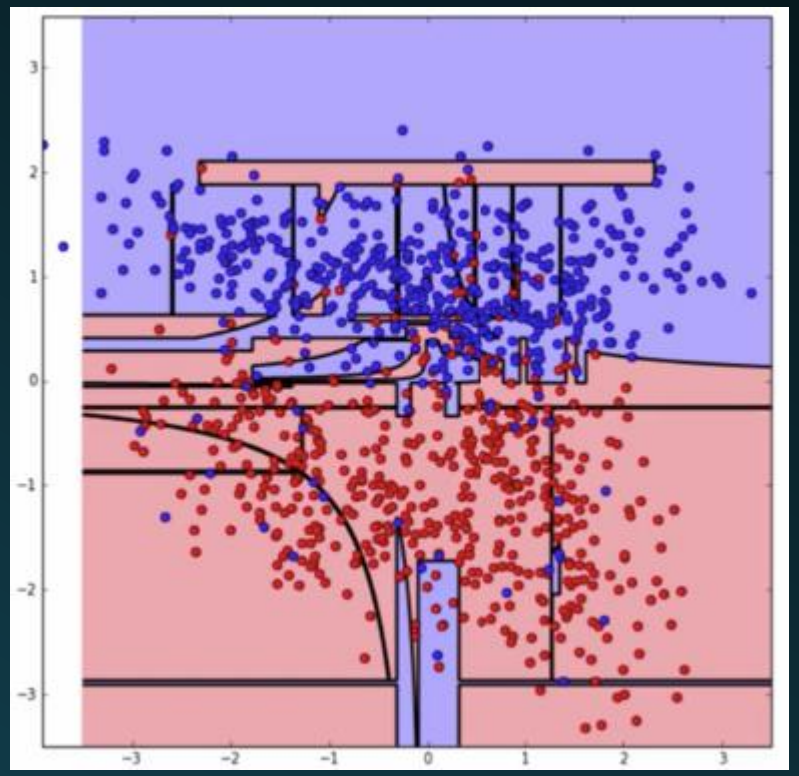




Главное не переборщить с деревом!!

Переобучение на решающих деревьях





Так как дерево может быть очень глубоким, то оно пытается уловить тончайшие закономерности, что может привести к переобучению. Алгоритм подстраивается под обучающие данные, а не улавливает закономерности

Как строятся решающие деревья?

- Какой вопрос лучше всего задать первым?

Как строятся решающие деревья?

- Какой вопрос лучше всего задать первым?
- Тот, который сильнее всего уменьшает количество оставшихся вариантов.

Как строятся решающие деревья?

- Какой вопрос лучше всего задать первым?
- Тот, который сильнее всего уменьшает количество оставшихся вариантов.
- Это интуитивно соответствует понятию прироста информации, связанному с энтропией.

Критерий качества. Энтропия

Энтропия Шеннона для системы с N различными состояниями задается следующим образом:

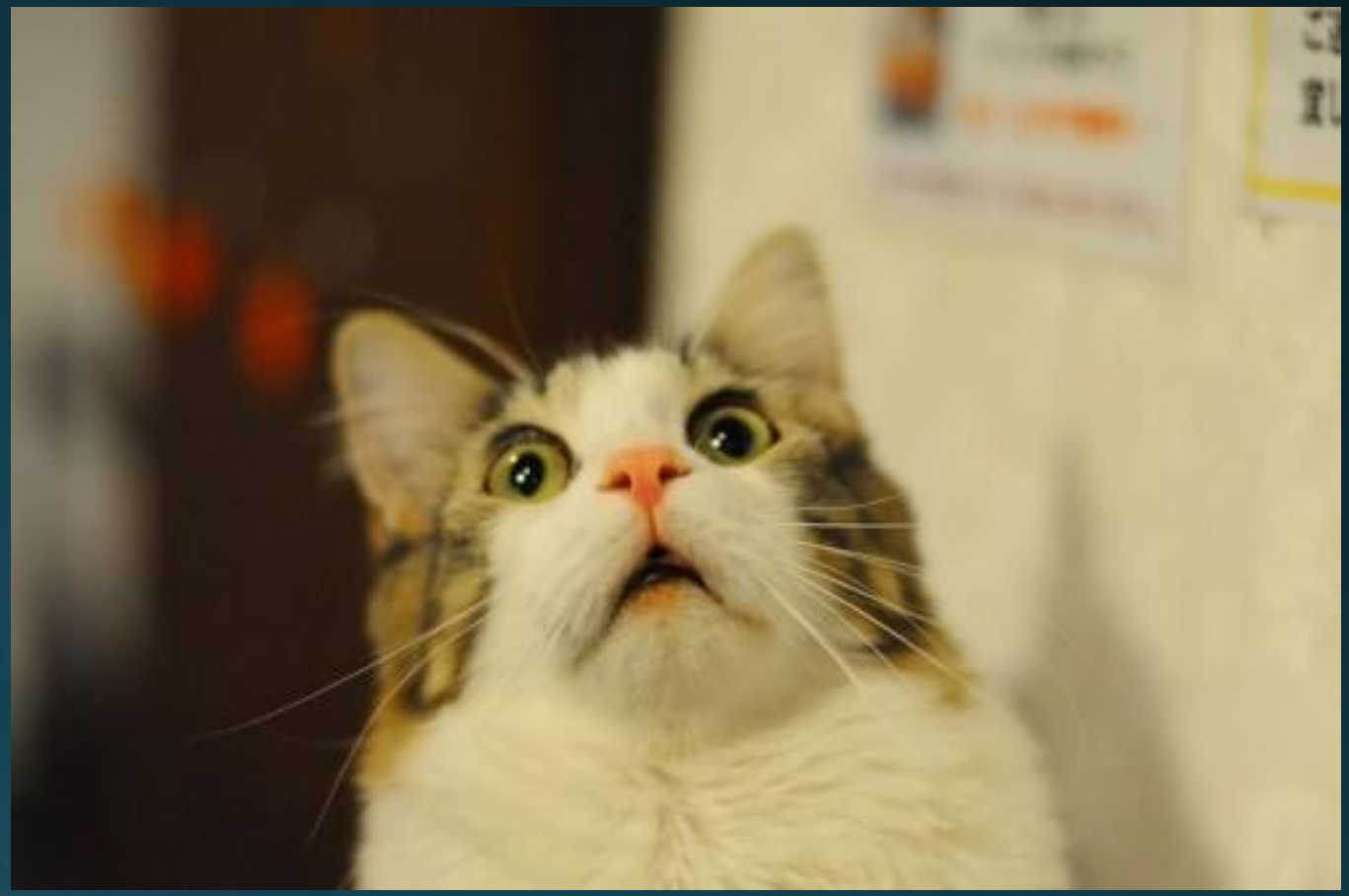
$$S = -\sum_{i=1}^{N} p_i \log_2 p_i ,$$

где p_i - вероятность системы находиться в состоянии i (многоклассовые критерии).

Энтропия – мера неопределенности системы.

Интуитивно, энтропия определяет степень хаоса в системе. Чем выше энтропия, тем менее упорядочена система.

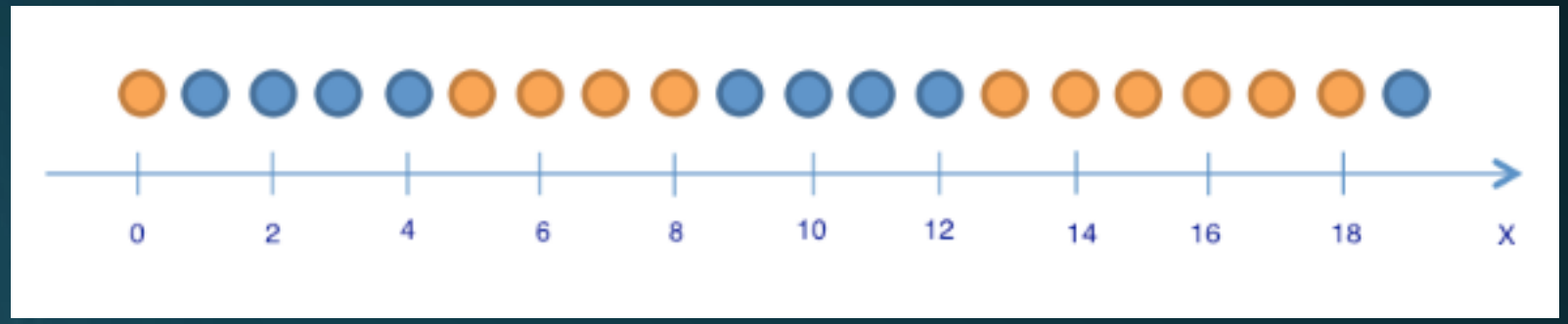
И как это работает?



Как строятся решающие деревья? Энтропия

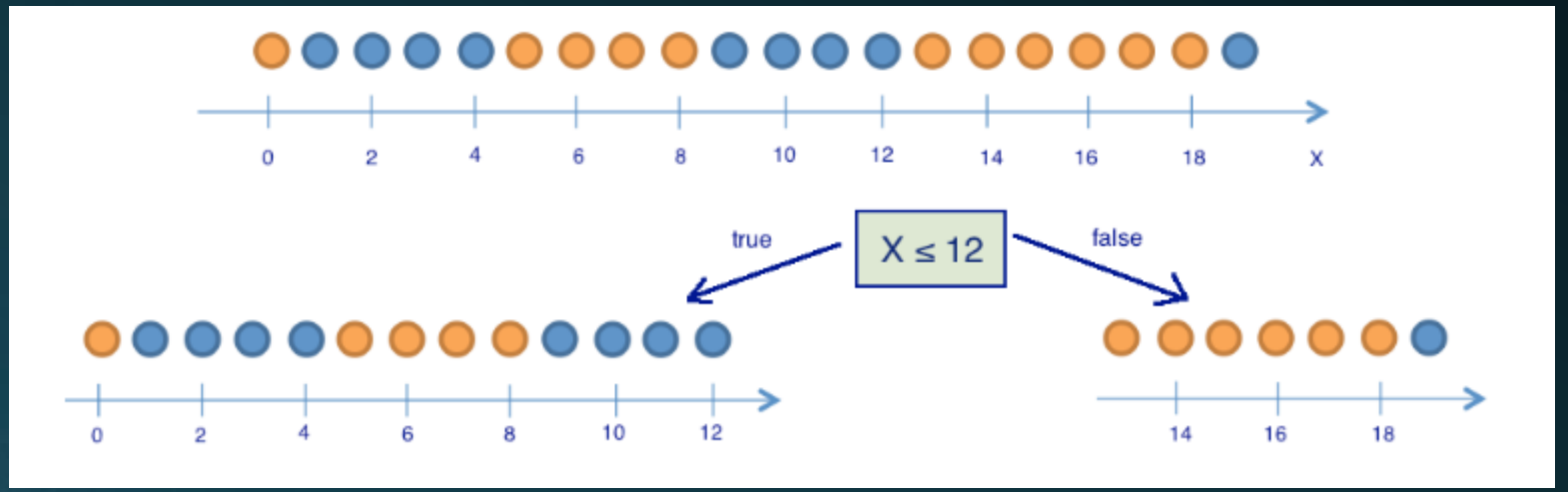
Игрушечный пример: определяем цвет шарика по его координате

Энтропия системы: $S = -\frac{9}{20}\log_2\frac{9}{20} - \frac{11}{20}\log_2\frac{11}{20} \approx 1$



Что будет с системой при разделении шариков на две группы: $x \leq 12$ и $x > 12$??

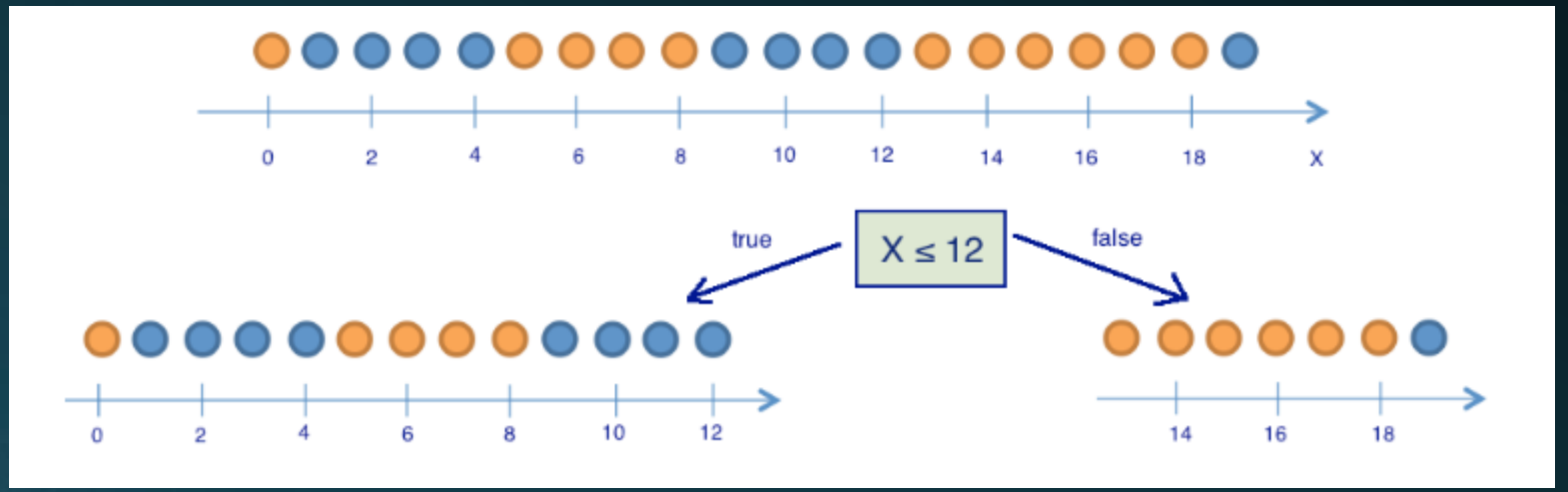
Как строятся решающие деревья? Энтропия



Энтропия левой группы: $S_1 = -\frac{8}{13}\log_2\frac{8}{13} - \frac{5}{13}\log_2\frac{5}{13} \approx 0{,}96$

Энтропия правой группы: $S_2 = -\frac{1}{7}\log_2\frac{1}{7} - \frac{6}{7}\log_2\frac{6}{7} \approx 0{,}6$

Как строятся решающие деревья? Энтропия



Энтропия левой группы: $S_1 = -\frac{8}{13}\log_2\frac{8}{13} - \frac{5}{13}\log_2\frac{5}{13} \approx 0{,}96$

Энтропия правой группы: $S_2 = -\frac{1}{7}\log_2\frac{1}{7} - \frac{6}{7}\log_2\frac{6}{7} \approx 0{,}6$

Что дальше?

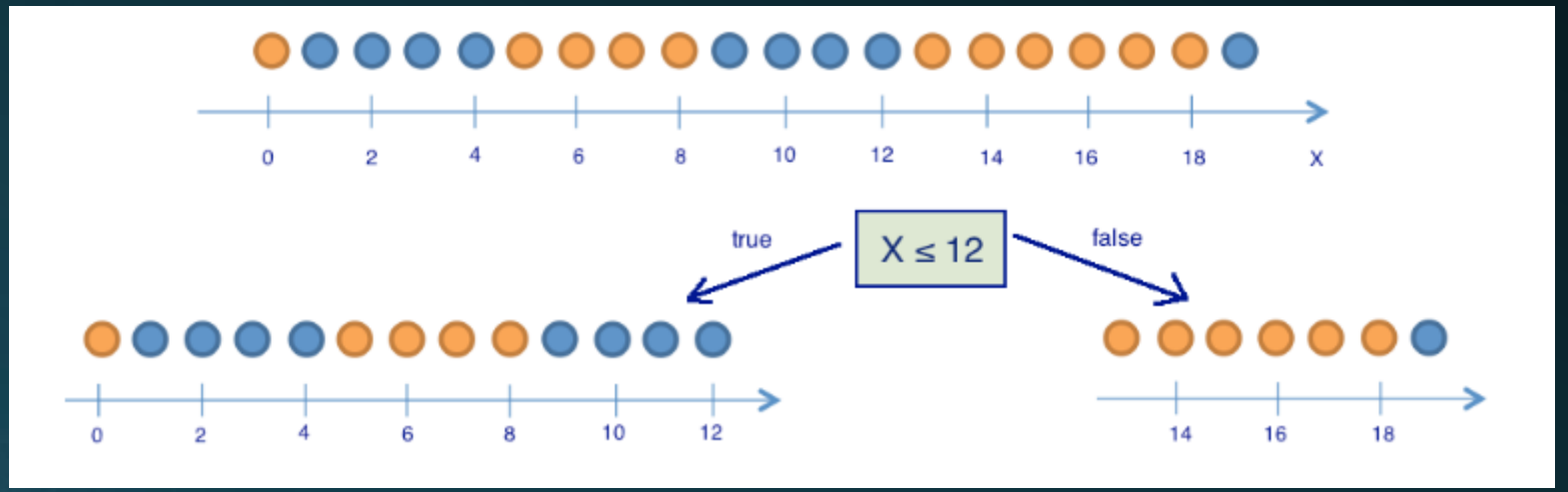
Как строятся решающие деревья? Энтропия

Так как энтропия – степень хаоса системы, то уменьшение энтропии называют приростом информации (information gain) при разбиении выборки по признаку Q:

$$IG(Q) = S - \sum_{i=1}^{q} \frac{N_i}{N} S_i.$$

где q – количество групп после разбиения, N_i - количество элементов в i-ой группе, N – общее количество элементов

Как строятся решающие деревья? Энтропия

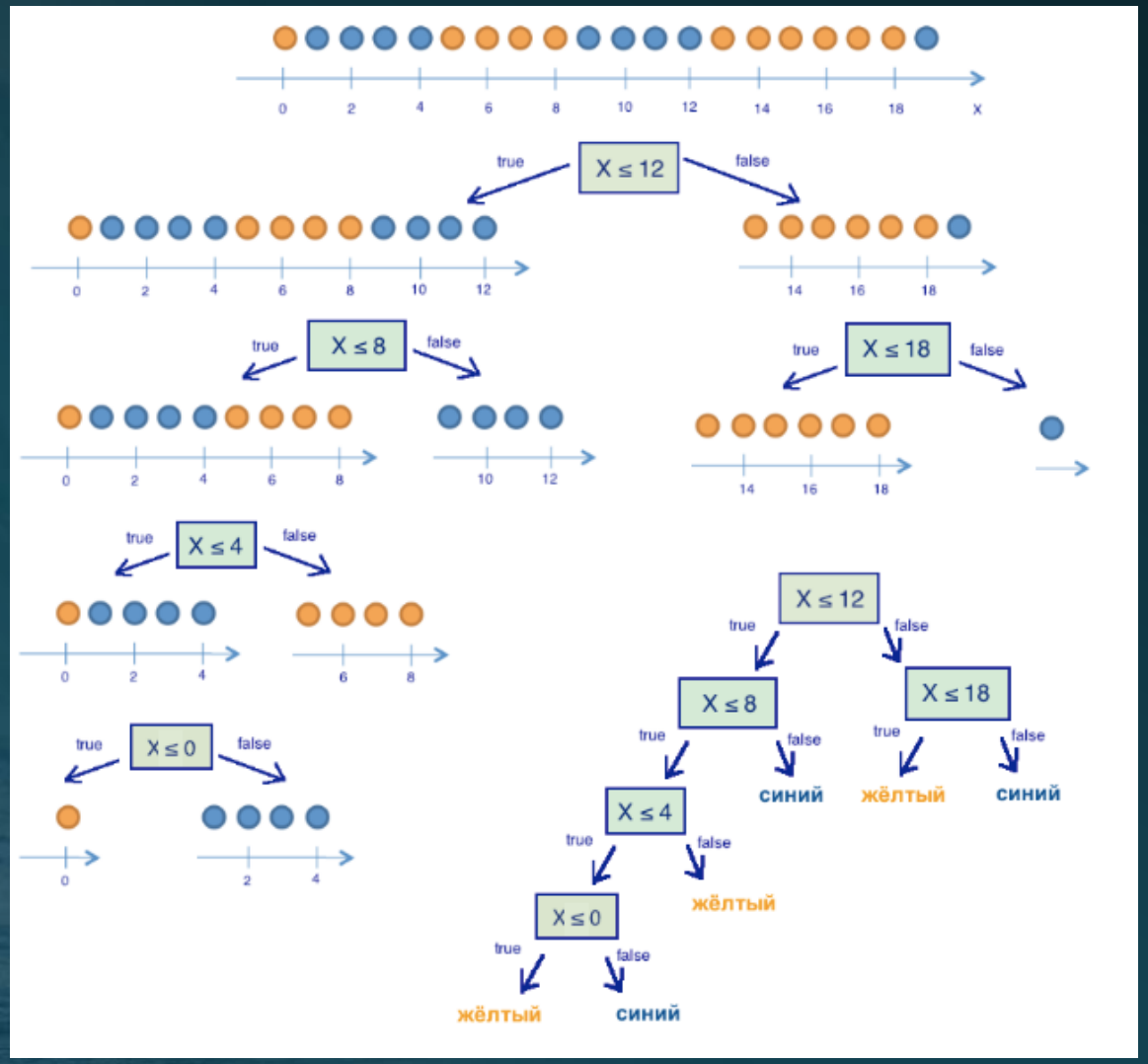


Таким образом:

$$IG(''x \leq 12'') = S - \frac{13}{20}S_1 - \frac{7}{20}S_2 \approx 0.16$$

В итоге мы получаем более упорядоченную систему, чем в начале.

Как строятся решающие деревья? Энтропия



Продолжим эту операцию до тех пор, пока в каждой группе шарики не будут одного цвета.

Энтропия в каждой группе будет равна 0.

В итоге получилось дерево, которое определяет цвет шарика по его координате.

Главное не переобучиться!

Как именно делить выборку на группы?

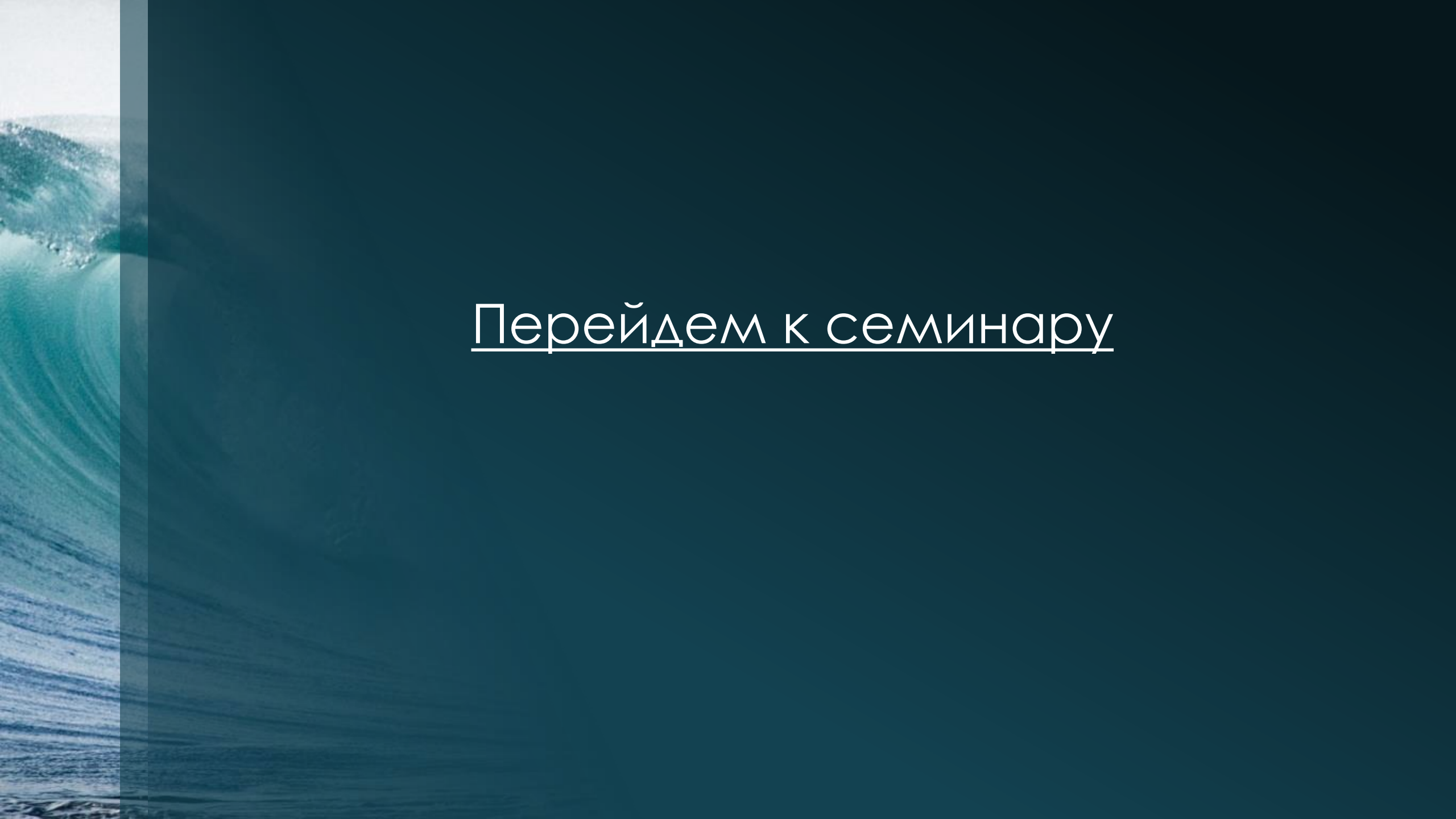
Алгоритм построения дерева ID3.

В основе лежит жадный алгоритм – на каждом шаге выбирается тот признак, при разделении по которому прирост информации будет наибольшим.
Дальше это процедура повторяется рекурсивно для каждой из подгрупп.

Чтобы не переобучиться обычно на алгоритм накладываются различные ограничения:

- Ограничение на прирост информации
- Ограничения на структуру дерева
  - Глубина дерева
  - Число наблюдений для ветвления
  - Количество признаков для деления

И так далее…



Перейдем к семинару