AI

ITAM
AI Lab MISIS

Лекция 2

Метрические алгоритмы, метрики задачи классификации, линейная и логистическая регрессия, SVM

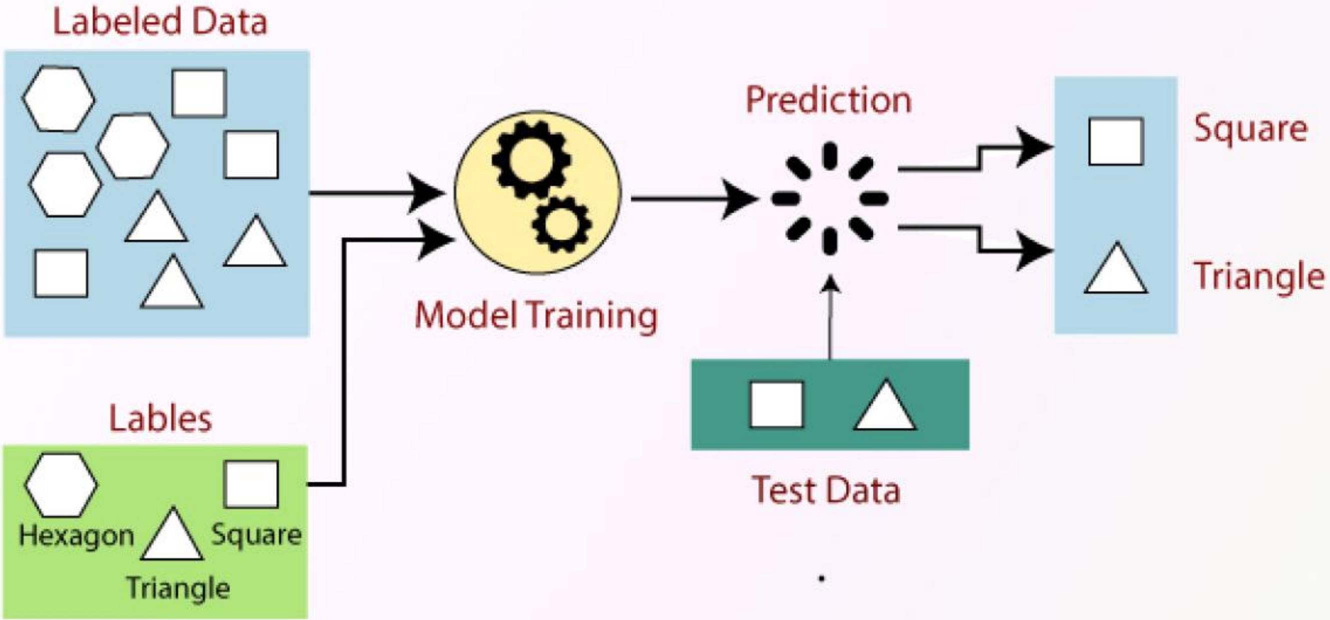
План лекции

- Обучение с учителем. Повторение
- Метрические алгоритмы
- Метод k ближайших соседей
- Метрики классификации
- Линейная регрессия
- Метрики регрессии
- Логистическая регрессия
- Метод опорных векторов

Supervised Learning

AI Lab MISIS

Пусть у нас есть датасет Xtrain={(x1, y1), ... (xn, yn). Где xi - объекты, yi- таргеты. Как уже было сказано, мы хотим построить отображение f: X → Y, где X - пространство объектов, а Y - пространство таргетов.



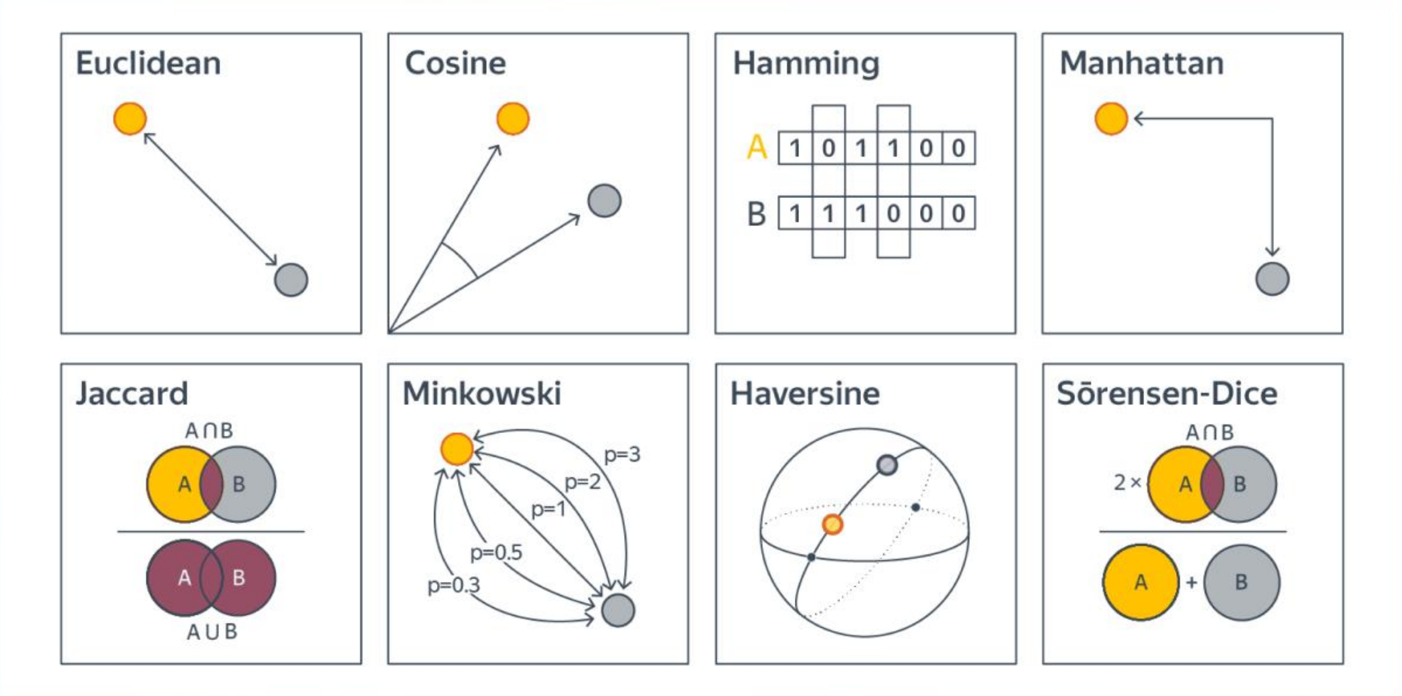
У нас есть объекты, мы хотим восстановить целевую зависимость, чтобы уметь выдавать ответы и на новые объекты тоже.

Метрические алгоритмы

Такими алгоритмами являются алгоритмы, основанные на расстоянии.

Для какого-то x из X мы анализируем расстояние до других точек из X.

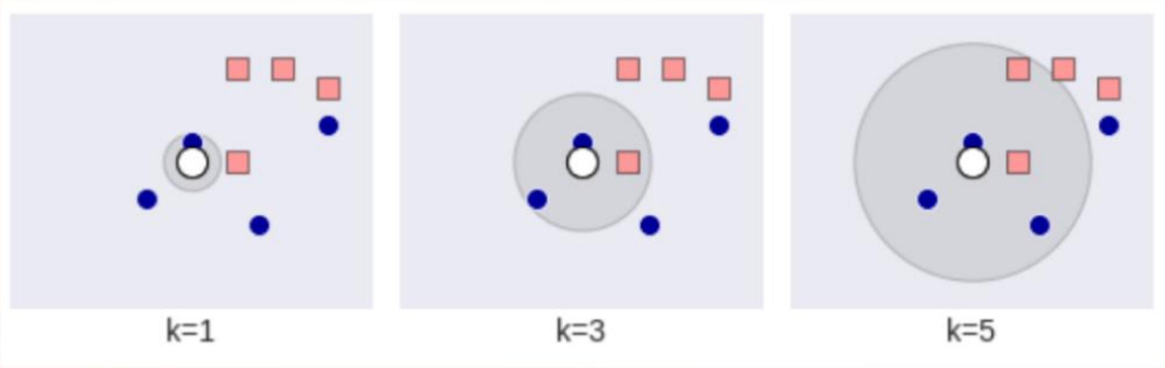
Их называют «distance-based» или «memory-based»



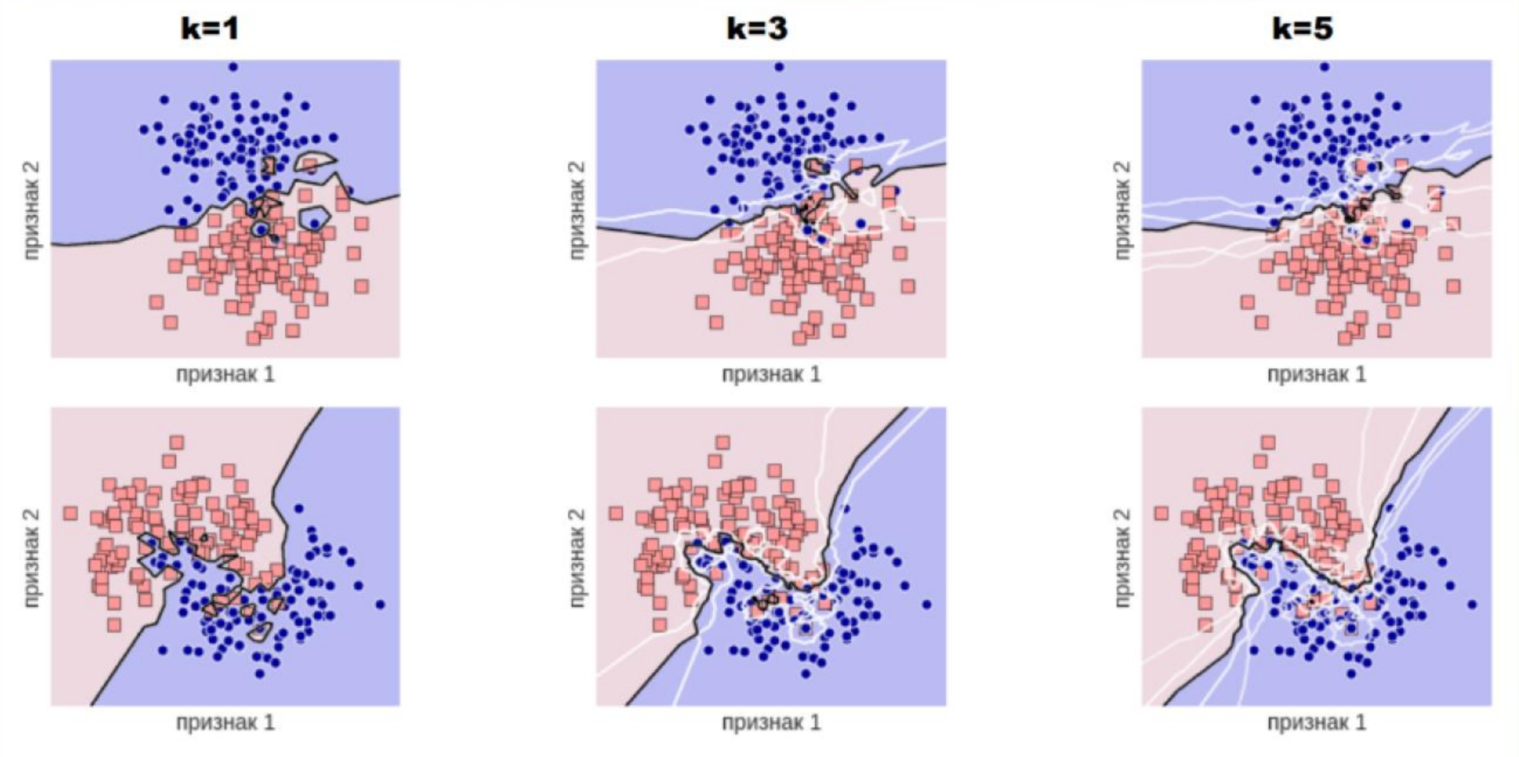
K-ближайших соседей (KNN)

Пусть **N(x)** - это окрестность объекта **x**. Введем расстояние от объекта x до остальных объектов датасета и упорядочим. В **N(x)** будут выходить **k** ближайших точек.

Задача классификации: a(x) = mode(y_i | x_i из N(x))
Задача регрессии: a(x) = mean(y_i | x_i из N(x))



- **k** - гиперпараметр. Метрика (способ считать расстояние) - тоже! Пытаемся как-то подобрать наилучшие.
- Нет обучения - мы просто запоминаем всю выборку, а потом просматриваем ее. Это называется **ленивым алгоритмом.**



Итог по метрическим алгоритмам

+ достаточно просто уметь измерять расстояние между точками
+ легко реализовать
+ легко интерпретировать
+ не надо обучать
+ мало гиперпараметров
- медленно
- надо много памяти для хранения
- проклятие размерности

Метрики классификации

Accuracy (верность) = (TP + TN) / (TP + FP + FN + TN)

Precision (точность) = TP/(TP + FP)

Recall (полнота) = TP / (TP + FN)

F1 = 2*Precision*Recall / (Precision + Recall)

	$y = 1$	$y = -1$
$a(x) = 1$	True Positive (TP)	False Positive (FP)
$a(x) = -1$	False negative (FN)	True Negative (TN)

Линейная регрессия

Предполагаем, что есть решение в виде: a(x1, x2, ..., xn) = w0 + w1×1 + w2×2 + ... + ... + wnxn, где x_i - признак (не объект!)

Для одного признака: a(x1) = w0 + w1×1

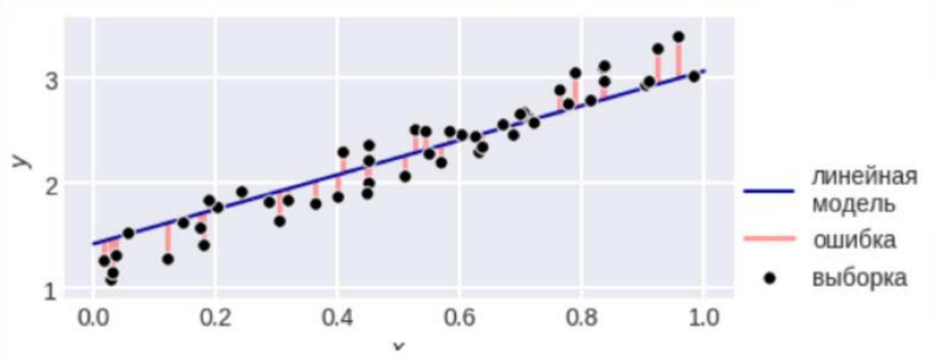
Хотим:

w0 + w1×1_1 = y1

w0 + w1×1_2 = y2

...

Матричная запись: w.T*X + b



Метрики регрессии

MSE - Mean Squared Error, Средняя квадратичная ошибка, Средний квадрат отклонения.

$$\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m} | a_i - y_i |^2$$

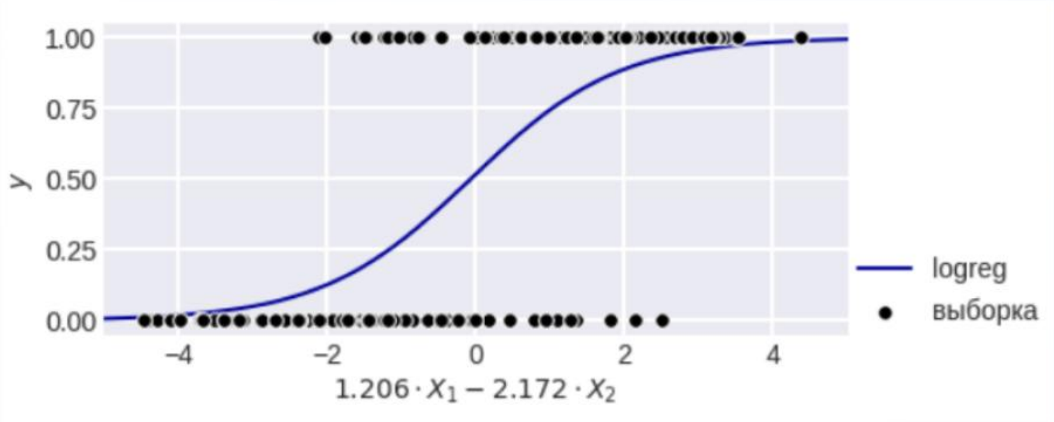
MAE - Mean Absolute Error, Средняя абсолютная ошибка, Средний модуль отклонения.

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m} | a_i - y_i |$$

RMSE - корень из средней квадратичной ошибки, корень из среднего квадрата отклонения.

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m} | a_i - y_i |^2}$$

Логистическая регрессия



У нас есть линейная регрессия: w.T*X + b
f: X → R
Чтобы решить задачу классификации: f: X → {0, 1}.
Давайте сделаем: s: R → [0, 1]!

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

SVM

Идея: строим разделяющую гиперплоскость, которая одновременно далеко до всех классов

Из плюсов: существование уникального решения и минимальная склонность к переобучению