Введение в машинное обучение

Лазар В. И. и Козлова Е. Р.

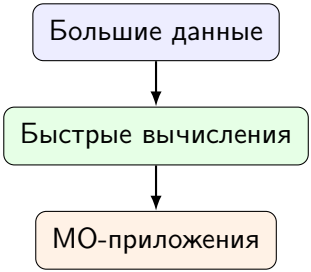
6 сентября 2025 г.

Что такое машинное обучение (МО)

- Компьютеры учатся на примерах, чтобы делать предсказания и решения.
- Не кодируем все правила вручную — модель выводит их из данных.
- Примеры вокруг: рекомендации, фильтр спама, распознавание речи.

Почему МО важно сегодня

- Много данных + быстрые компьютеры.
- Где правил слишком много — МО эффективнее.
- Помогает автоматизировать рутину и поддерживать решения.

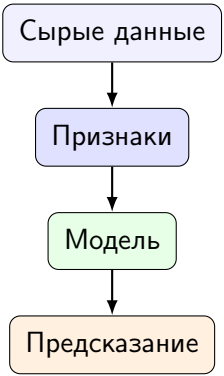


Постановка задачи МО

- Имеем обучающую выборку $D = \{(x_i, y_i)\}_{i=1}^{n}$.
- Ищем модель $f(x)$, минимизирующую ошибку $L(f(x), y)$.
- Важно: не выучить наизусть, а обобщать на новые данные.

Основные понятия МО

- Признаки (features)
- Метка / целевая переменная (label/target)
- Модель и её параметры
- Обучение и обобщающая способность

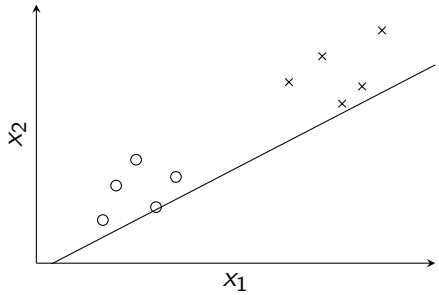


Типы задач МО

- Обучение с учителем: есть ответы y (классификация, регрессия).
- Без учителя: меток нет — ищем структуру (кластеризация, понижение размерности).
- С подкреплением: агент учится по наградам во взаимодействии со средой.

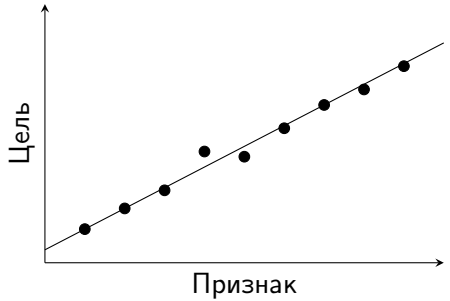
Классификация: пример границы

- Цель: предсказать категорию (спам/не спам и т.п.).
- Метрики: Accuracy, Precision/Recall, F1.



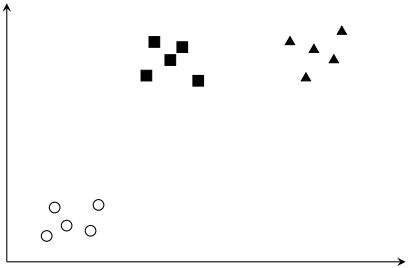
Регрессия: линия тренда

- Цель: предсказать число (например, цену).
- Метрики: MAE, RMSE, R^2.



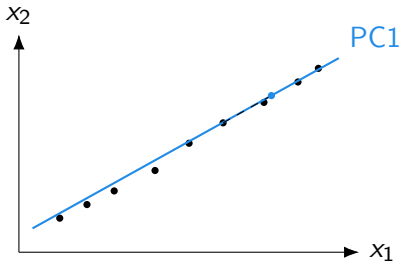
Обучение без учителя: кластеризация

- Группируем похожие объекты без меток.
- Пример: сегменты покупателей по поведению.
- Алгоритмы: k-means, иерархическая, DBSCAN.

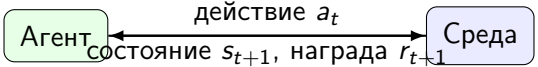


Понижение размерности (PCA) — идея

- Сжать данные, сохраняя главное.
- Визуализация высоких размерностей.



Обучение с подкреплением (RL)

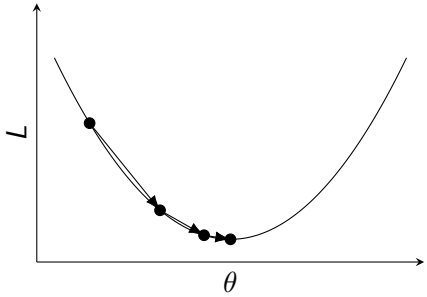


- Цель: максимизировать суммарную награду.
- Примеры: игры, роботы, рекомендации.

Как учится модель: градиентный спуск

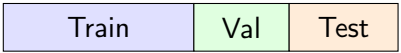
$$\theta \leftarrow \theta - \eta \nabla_\theta L(\theta)$$

где $L(\theta)$ — функция потерь, η — шаг обучения.

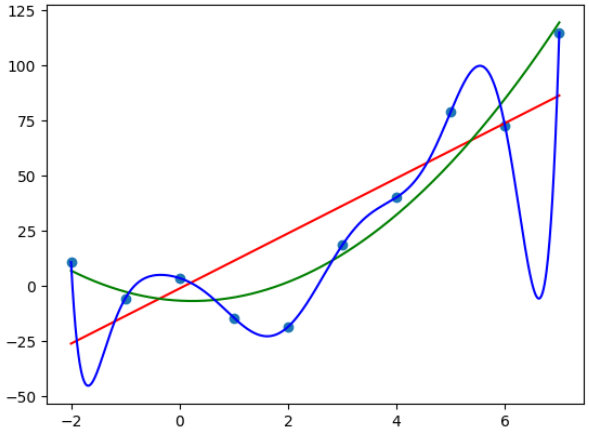


Обучение, валидация, тест

- Train — настраиваем параметры.
- Val — подбираем гиперпараметры.
- Test — финальная независимая проверка.

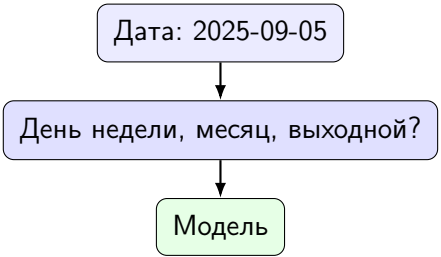


Недообучение и переобучение: шумные данные и три модели



Признаки и предобработка

- Очистка: пропуски, выбросы, опечатки.
- Кодирование категорий, нормализация чисел.
- Feature engineering: новые информативные признаки.



k-Nearest Neighbors (k-NN): идея

Регрессия.

Одинаковые веса:

$$\hat{y} = \frac{1}{k} \sum_{i \in N_k(x)} y_i \,.$$

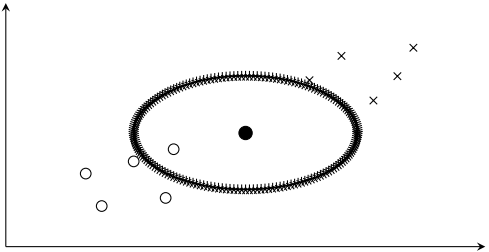
Веса по расстоянию:

$$\hat{y} = \frac{\sum_{i \in N_k(x)} w_i \, y_i}{\sum_{i \in N_k(x)} w_i} \,.$$

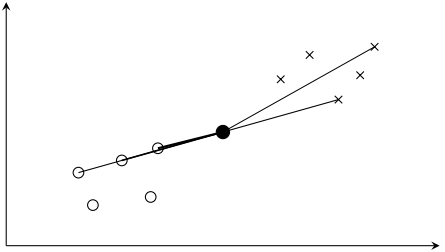
Замечание: метрика d обычно евклидова после масштабирования признаков; k и параметры весов (p, h, ε) подбираются по кросс-валидации.

k-NN: как понимать веса (физическая интерпретация)

Одинаковые веса (uniform). Все соседи внутри «окна» влияют одинаково — как равномерно идущий дождь в радиусе r: каждая капля даёт одинаковый вклад. Это соответствует «top-hat» ядру (равномерному по диску).



Веса по расстоянию. Ближние «тянут» сильнее, дальние — слабее: как интенсивность света или гравитация ($\propto 1/r^2$). Пример: $w_i = 1/(d_i + \varepsilon)^p$ (обычно $p \in [1, 2]$) или гауссово ядро.



Выбор k и типа весов — гиперпараметры: подбираются по кросс-валидации; масштабируйте признаки перед поиском соседей.