Deep Learning School

Лекция

Машинное обучение.
Введение.

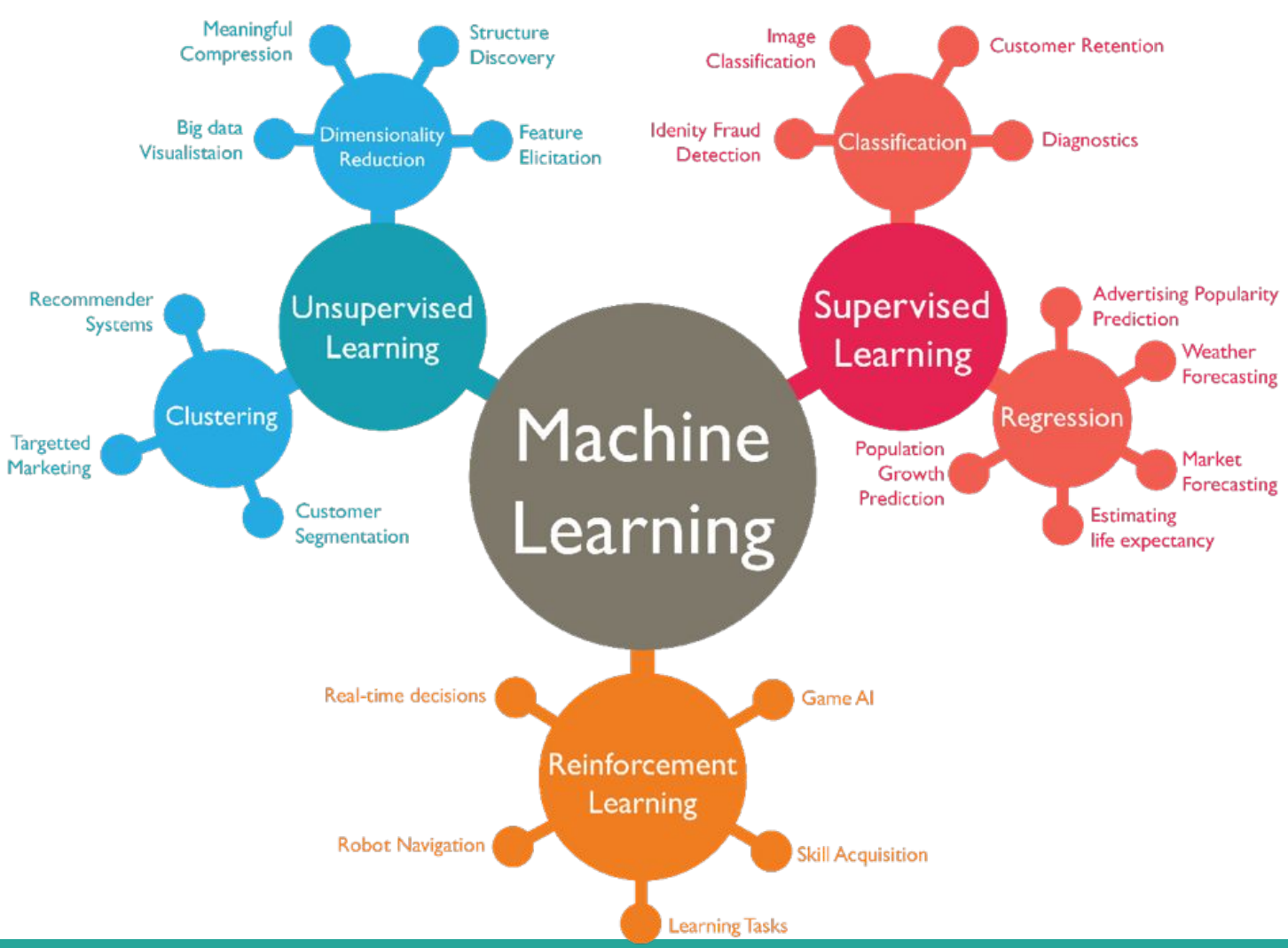
План курса

1. Машинное обучение (3 занятия)
2. Нейронные сети (2 занятия)
3. Сверточные нейронные сети (2 занятия)
4. Детекция (1 занятие)
5. Сегментация (1 занятие)
6. GAN (1 занятие)
7. Гостевая лекция (1 занятие)
8. Практическое занятие по CNN (1 занятие)
9. Kaggle (1 занятие)

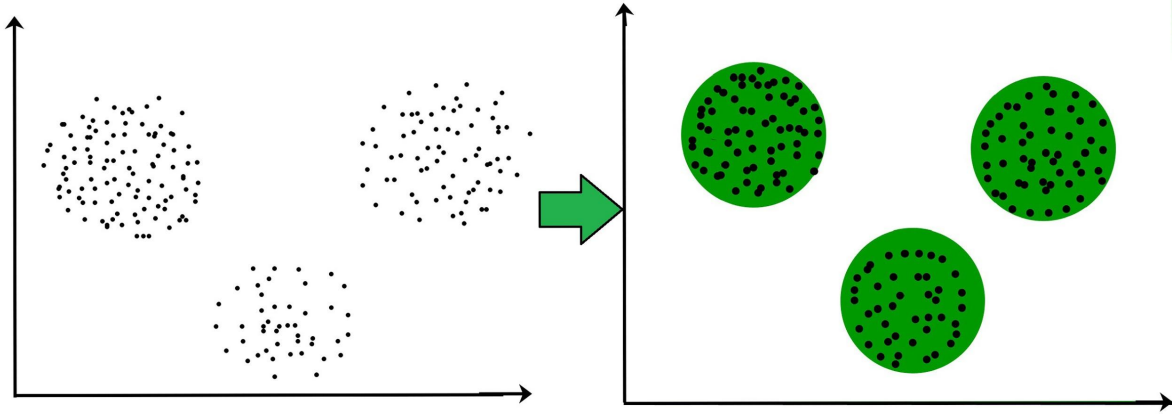
План лекции

1. Типы задач машинного обучения
2. Обучение с учителем
3. KNN
4. Обучение моделей
5. Линейная регрессия и переобучение
6. Алгоритм применения ML к задачам

Типы задач машинного обучения


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Unsupervised learning



Кластеризация. Применение на практике: разделение аудитории на группы с общими интересами для эффективной рекламы.

Reinforcement learning

Reinforcement Learning основан на том, что алгоритм за каждое свое действие получает награду или наказание.

AlphaGo сыграл сам с собой миллионы партий.

Обучение с учителем

Обучение с учителем (supervised learning)

X - множество объектов
Y - множество ответов

$y : X \to Y$ - истинная зависимость.

Обучающий датасет - множество наборов из фичей и значений целевой переменной. Мы обозначим его $X_{train} \subset X$.

$$X_{train} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ & \dots & \dots & \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{pmatrix} \qquad y_{train} = \begin{pmatrix} y_1 \\ \dots \\ y_n \end{pmatrix}$$

Обучение с учителем (supervised learning)

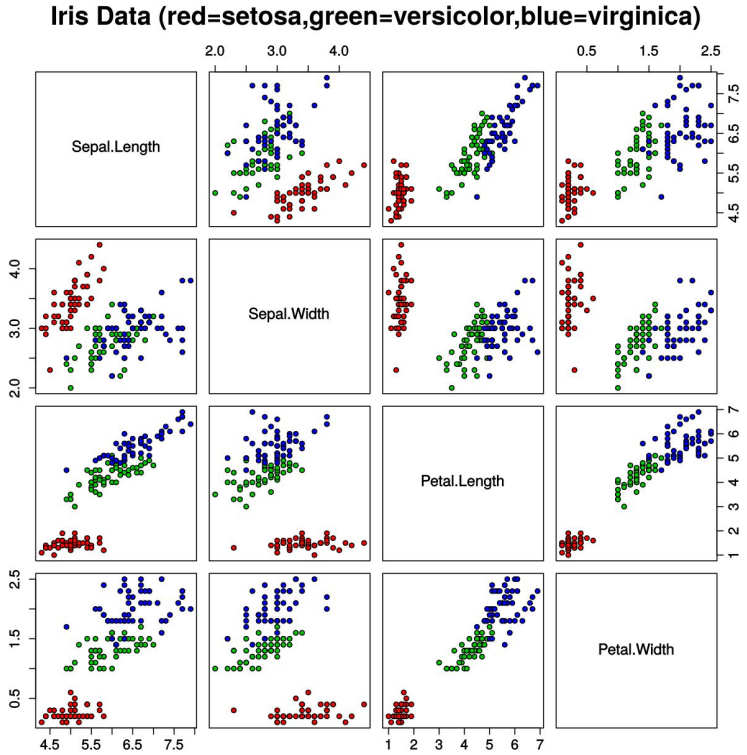
Типы признаков (features):

- Числовые (Numerical)
- Категориальные (Categorical)
- Порядковые (Ordinal)

Типы задач:

- Классификация (Classification)
  $Y = \{0, 1\}, Y = \{1, 2, \ldots, n\}, Y = \{0, 1\}^n$
- Регрессия (Regression)
  $Y = \mathbb{R}$
- Ранжирование (Ranking)
  $Y = \{1, 2, \ldots, n\}$ (числа упорядочены)

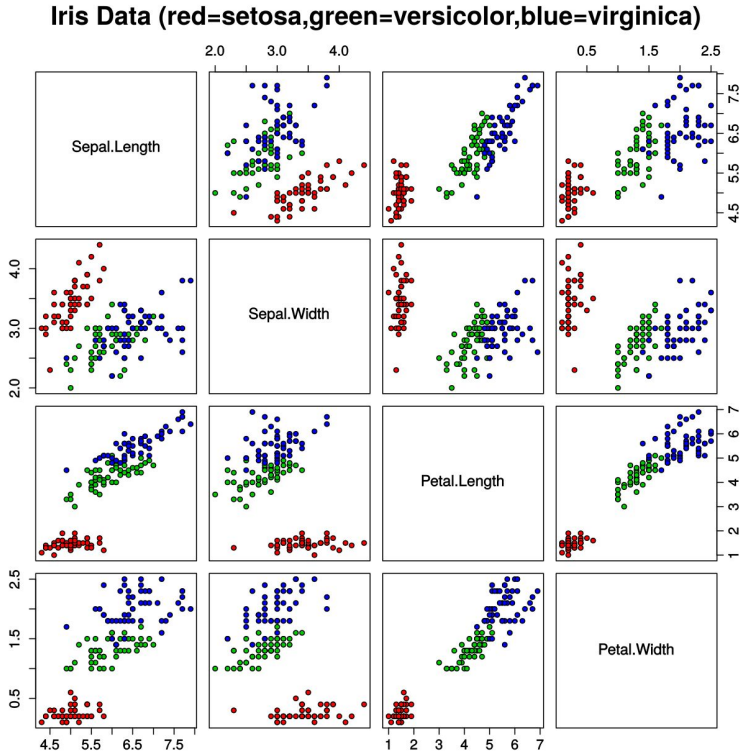
Примеры задач (Ирисы Фишера)



Какая это задача?

Какие есть признаки?

Примеры задач (Ирисы Фишера)



Какая это задача?

$Y = \{1, 2, 3\}$.

Задача классификации.

Какие есть признаки?

$X = \mathbb{R}^4$.

Есть только числовые признаки.

Примеры задач (Цена дома)

Нужно предсказать стоимость дома. Есть обучающий датасет со следующими признаками:

- Удаленность от метро.
- Оценка состояния дома (плохое, среднее, хорошее, отличное).
- Количество комнат.
- Площадь.
- Год строительства.
- Название района, в котором находится дом.

Какая это задача?

Какие есть признаки?

Примеры задач (Цена дома)

Нужно предсказать стоимость дома. Есть обучающий датасет со следующими признаками:

- Удаленность от метро.
- Оценка состояния дома (плохое, среднее, хорошее, отличное).
- Количество комнат.
- Площадь.
- Год строительства.
- Название района, в котором находится дом.

Какая это задача?
$Y = \mathbb{R}$.
Задача регрессии.

Какие есть признаки?
Числовые, порядковые, категориальные.

Примеры задач (Поисковая выдача)

Какая это задача?

Получив запрос от пользователя нужно найти наиболее полезные документы из некоторой базы.
Что нам известно:

- Запрос пользователя.
- Текст документа.
- Какие ключевые слова есть в каждом документе.
- Насколько каждый документ популярен.
- итд.

Какие есть признаки?

Примеры задач (Поисковая выдача)

Получив запрос от пользователя нужно найти наиболее полезные документы из некоторой базы.
Что нам известно:

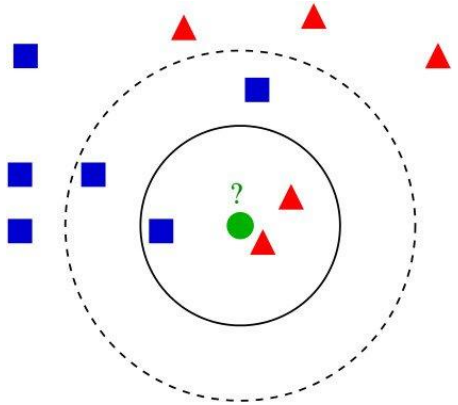
- Запрос пользователя.
- Текст документа.
- Какие ключевые слова есть в каждом документе.
- Насколько каждый документ популярен.
- итд.

Какая это задача?
$Y = \{1, 2, \dots, n\}$ (числа упорядочены)
Задача ранжирования.

Какие есть признаки?
Данные намного сложнее и требуют предобработки.

KNN

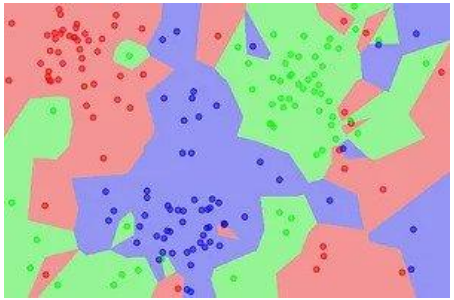
K-Nearest Neighbors



Решение задачи классификации:
Обучение: Просто запоминаем обучающую выборку.

Предсказание:

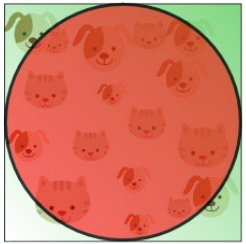
- Получаем точку **x**, в которой надо сделать предсказание.
- Ищем k ближайших соседей.
- В качестве ответа возвращаем класс, которого больше всего среди соседей.



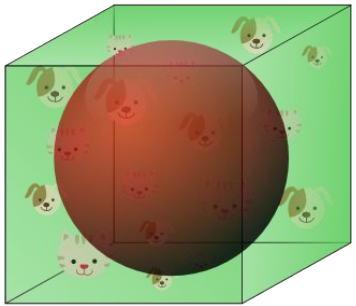
Images from https://www.unite.ai/what-is-k-nearest-neighbors

Curse of Dimensionality

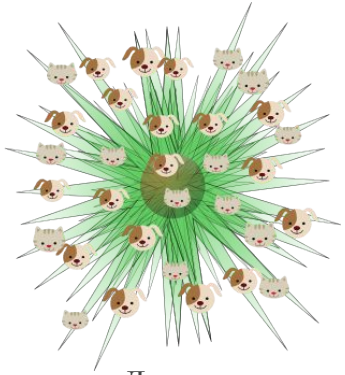
В KNN мы делаем очень слабое предположение: близкие точки будут иметь близкие ответы.
При большой размерности данных в близкую область попадет мало объектов.



Два признака



Три признака



Десятки признаков

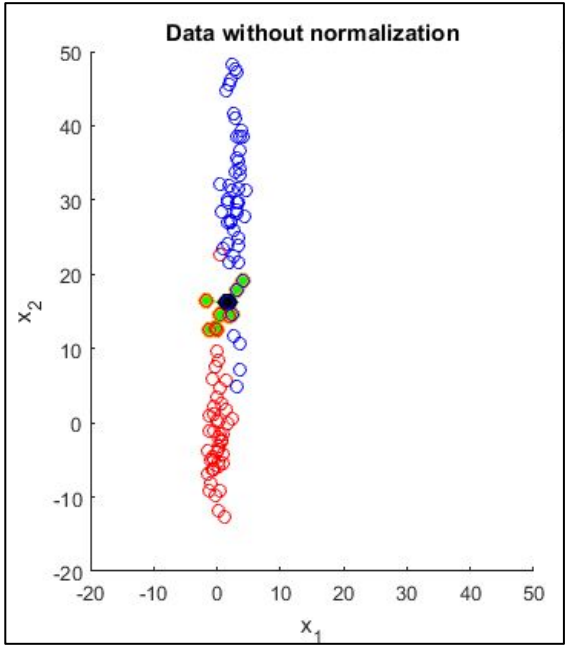
Feature Scale

Если в качестве метрики взять обычное расстояние между векторами, то возникает проблема масштаба признаков.

Пример:
Задача определения стоимости дома по признакам:

- Расстояние до метро в метрах
- Количество комнат

Количество комнат почти не будет влиять на предсказание



Обучение моделей

Обучение с учителем (supervised learning)

Наша задача - найти функцию хорошо приближающую реальную зависимость $y(x)$.
Назовем такое решение $\hat{y} : X \to Y$ (эта функция должна быть вычислима на компьютере).

Обычно мы выбираем решение из некоторого параметризованного семейства.

$$\mathcal{F} = \{\hat{y}_\theta \mid \theta \in \Theta\}, \Theta - \textit{множество параметров.}$$

Обучение с учителем (supervised learning)

Обучение -- процесс выбора параметра θ, которому соответствует наиболее подходящее нам решение задачи $\hat{y}_\theta(x_1, x_2)$.

Пример семейства моделей (функции порога)

Задача: определить, можно ли ребенку пройти на аттракцион? Причем мы знаем его рост и возраст.

Множество, в котором мы будем искать решения состоит из функций вида:

$$\hat{y}_{(a,b)}(x_1, x_2) = \begin{cases} 1 & x_1 \geq a, x_2 \geq b \\ 0 & otherwise \end{cases}$$

Параметр в данном случае $\theta = (a, b)$. А множество возможных значений параметра $\Theta = \mathbb{R}$.

Обучение с учителем (supervised learning)

Как обучать алгоритм (подбирать оптимальные параметры)?

Обучение с учителем (supervised learning)

Функция потерь (loss):
Определим функцию $L(y, \hat{y}(x))$, ее значение показывает насколько сильно наше предсказание отличается от реального значения.

Пример:
Задача предсказания цены дома из предыдущих примеров.
Возможные функции потерь:

$$L(y_{\text{true}}, \hat{y}(x)) = (y_{\text{true}} - \hat{y}(x))^2$$ --- квадратичная функция потерь

$$L(y_{\text{true}}, \hat{y}(x)) = |y_{\text{true}} - \hat{y}(x)|$$ --- абсолютная функция потерь

$$L(y_{\text{true}}, \hat{y}(x)) = (y_{\text{true}} - \hat{y}(x))^2 + 7$$

Обучение с учителем (supervised learning)

Эмпирический риск:
Определим эмпирический риск как среднее значение функции потерь на обучающем датасете.
Часто функцию эмпирического риска также называют лоссом.

Обучение:

$$\theta_{\text{best}} = \underset{\theta \in \Theta}{\operatorname{argmin}} \frac{1}{\text{dataset size}} \sum_i L(y^i_{\text{true}}, \hat{y}_\theta(x^i))$$

(Это просто математическое определение. Конкретный алгоритм получения лучшего параметра для каждой модели свой.)

Линейная регрессия и переобучение

Переобучение для линейной регрессии

Вспомним как выглядит линейная регрессия:

$$\hat{y}(x_1, x_2, \ldots, x_n) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

Обучение линейной регрессии:

Классически в качестве лосса берут Mean Squared Error (среднее квадратов ошибок)

$$\arg\min_{\theta_0,\ldots,\theta_n} \sum_i (y^i_{\text{true}} - \theta_0 - \theta_1 x^i_1 - \ldots - \theta_n x^i_n)^2$$

Переобучение для линейной регрессии

Вспомним как выглядит линейная регрессия:

$$\hat{y}(x_1, x_2, \ldots, x_n) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

Polynomial Regression:

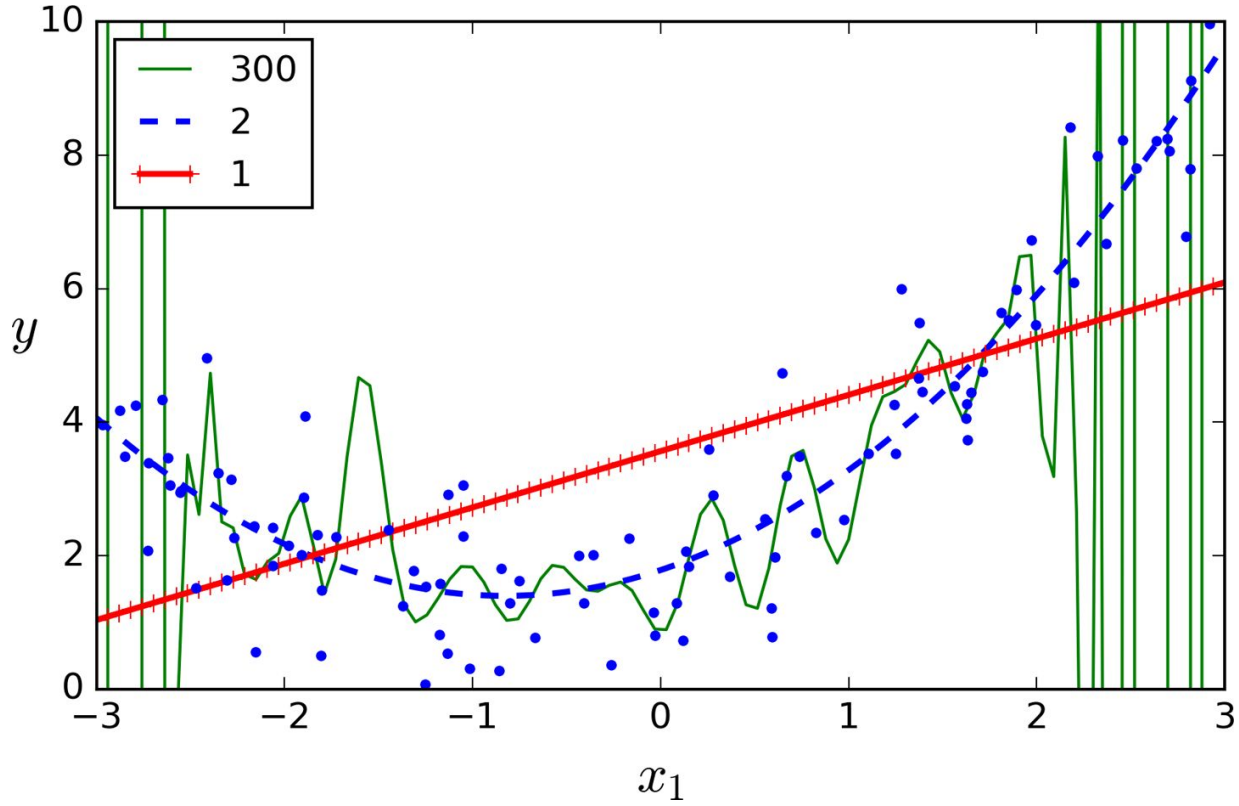
Пусть у нас изначально есть только один признак x. Создадим новые:

$x_1 = x, x_2 = x^2, \ldots, x_n = x^n$

Тогда линейная регрессия от таких признаков называется полиномиальной:

$$\hat{y}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \ldots + \theta_n x^n$$

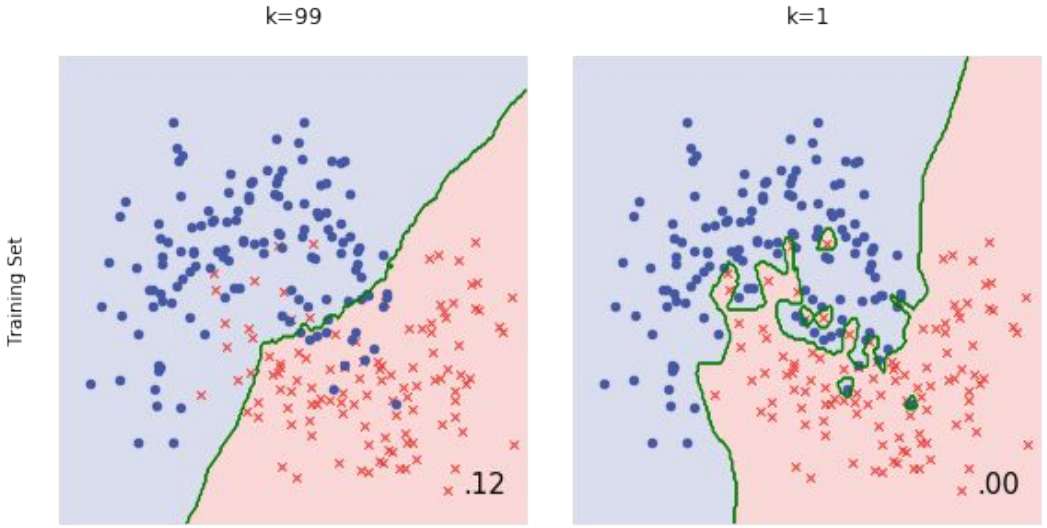
Переобучение для линейной регрессии



Переобучение (KNN)

Если в алгоритме KNN мы возьмем k = 1, то получим идеальные предсказания на всем обучающем датасете и эмпирический риск (средний лосс) будет равен 0.

Но такие предсказания могут быть очень плохими.

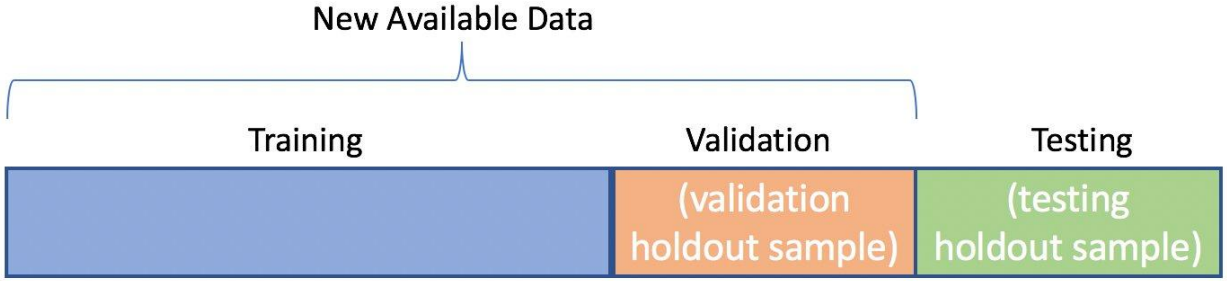


Как определить термин переобучение и потом находить его?

Разделение на Train/Validation/Test

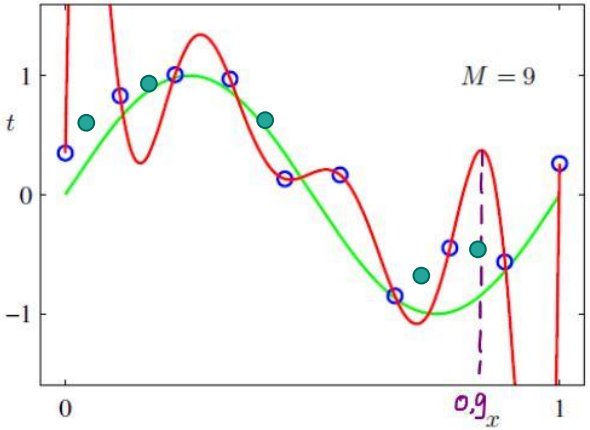
- Train - данные для обучения.
- Validation - данные для итеративной оценки качества.
- Test - данные для финальной оценки качества.

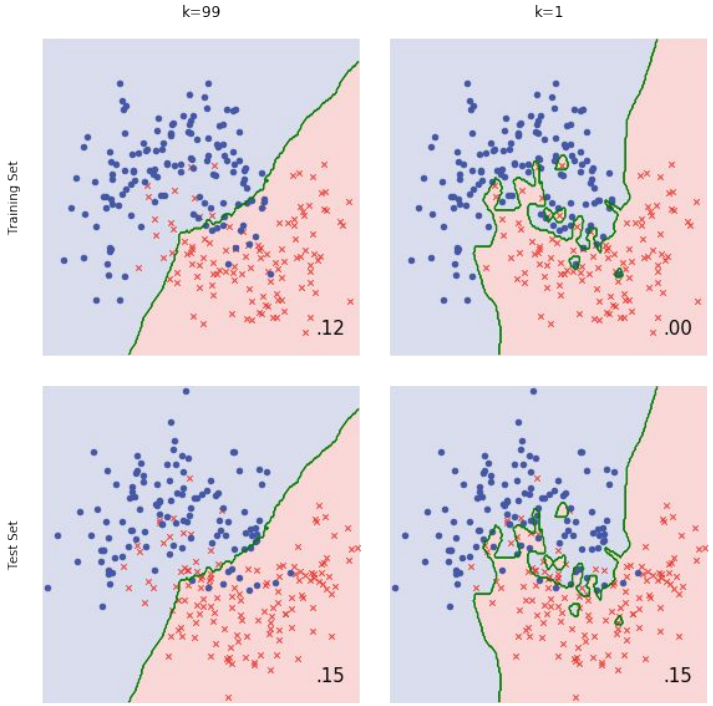
Часто можно опустить test часть. В этом случае название validation dataset и test dataset значат одно и то же.



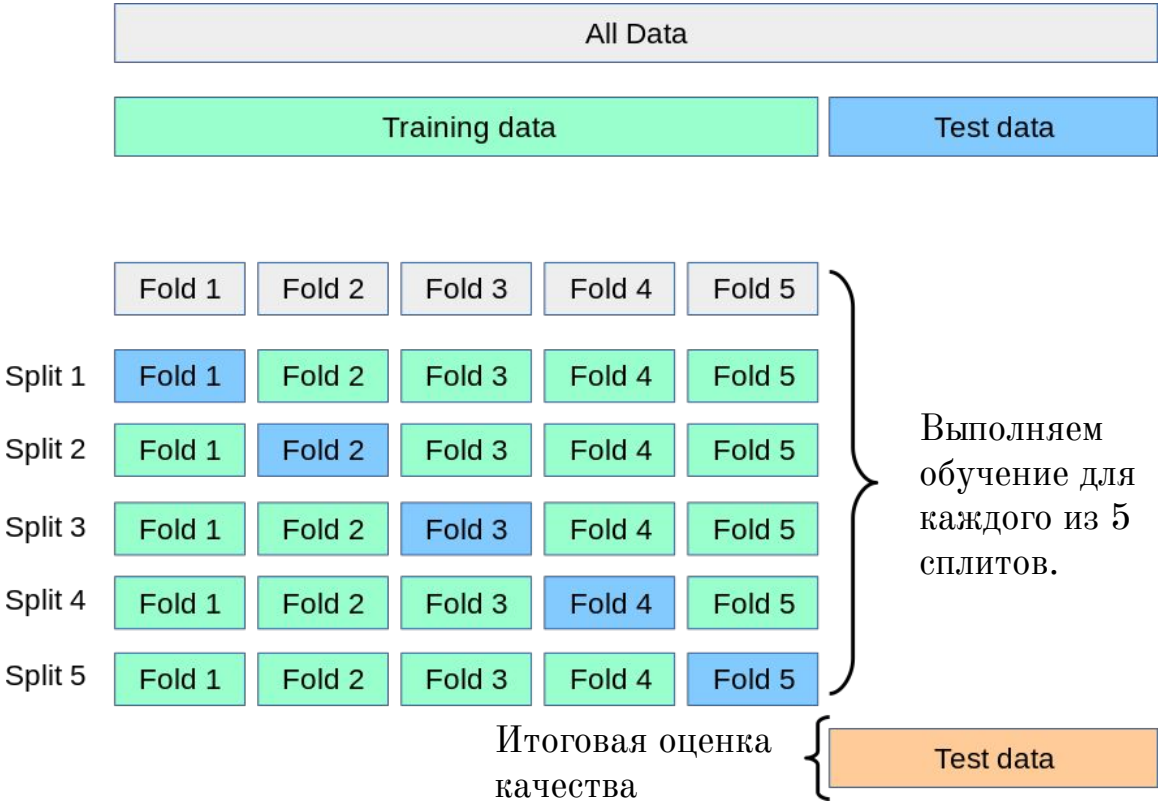
Разделение на Train/Validation/Test

Переобучение - ситуация, когда качество модели на train данных значительно лучше, чем на validation/test.





Cross-validation



Алгоритм применения ML

Решение задачи

Algorithm 1: Алгоритм решение ML задачи

Result: Модель "закона природы"$\hat{y}_\theta(x)$

Разделяем датасет на train, validation, test часть.;

Обрабатываем данные. ;

while *Качество на validation не достаточно высокое* **do**

  Выбираем семество моделей $\mathcal{F}$, выбрав тип модели и гиперпараметры.;

  Обучаем, решая задачу $\theta_{\text{best}} = \arg\min_\theta \sum_i L(y^i_{\text{true}}, \hat{y}_\theta(x^i))$;

  Проверяем качество (лосс и метрики) на validation датасете.;

end

Проверяем качество на test датасете;

The End