МАШИННОЕ ОБУЧЕНИЕ

Машинное обучение

Arthur Samuel (1959), Machine learning -

“Поле исследования, которое дает возможность компьютеру обучаться не будучи явно запрограммированным”.

Tom Mitchell (1998):

“ Компьютерная программа перенимает опыт E в отношении задачи T и некоторого показателя измерения P, если ее производительность на T, измеренная по P улучшается с опытом E”.

Область искусственного интеллекта, использующая самообучающиеся алгоритмы, имитирующие обучение мозга.

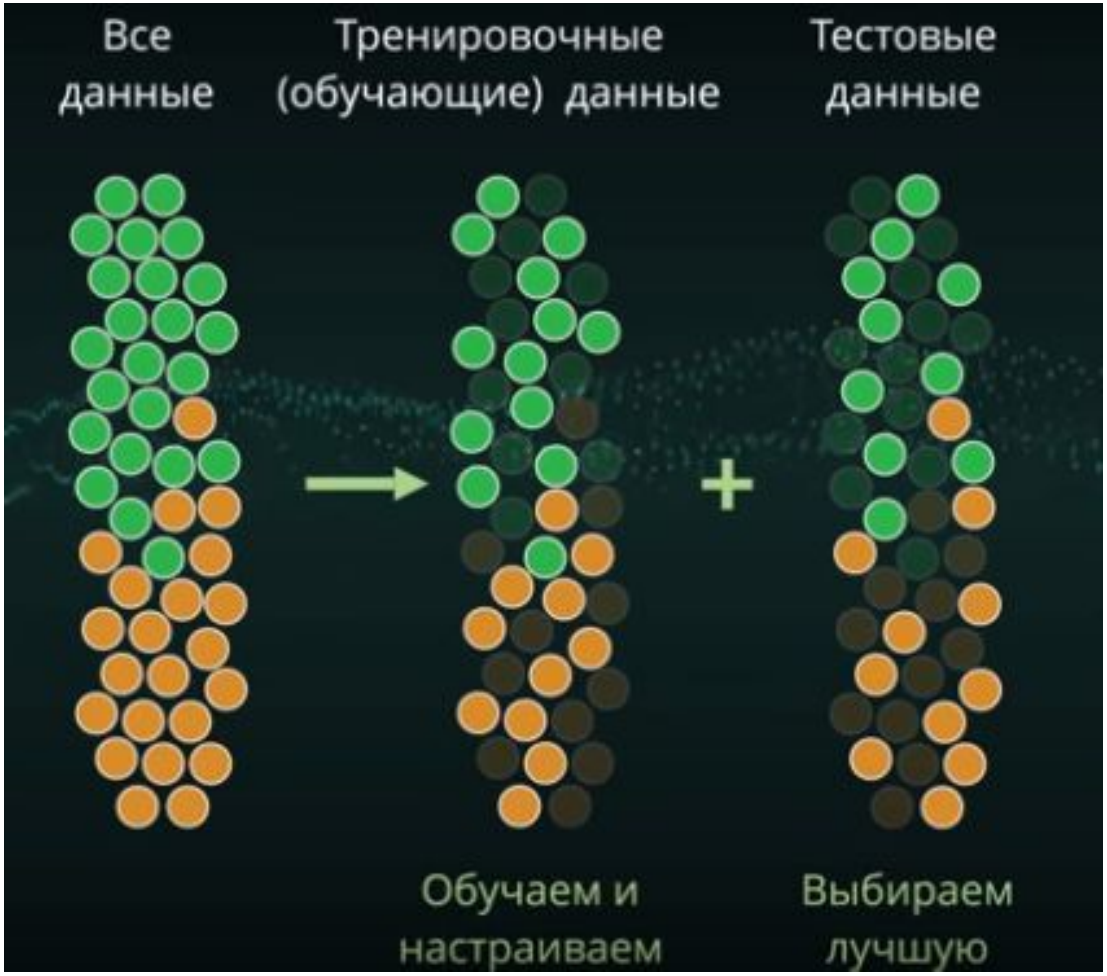
Применение машинного обучения

- Обработка запросов поисков систем
- Распознавание фотографий
- Распознавание рукописного текста
- Фильтры почтового спама
- Интеллектуальный анализ баз данных (лучшее обслуживание потребителей, вычислительная биология, машиностроение,…)
- Автономное управление машинами (автомобили, вертолеты,…)
- Выдавать индивидуальные рекомендации - подстраиваться под ваши предпочтения

Типы машинного обучения

- **Обучение с учителем**
  - ✔ Помеченные данные
  - ✔ Непосредственная обратная связь
  - ✔ Прогнозирование исхода
- **Обучение без учителя**
  - ✔ Метки/цели отсутствуют
  - ✔ Отсутствует обратная связь
  - ✔ Поиск скрытой структуры в данных
- **Обучение с подкреплением**
  - ✔ Процесс принятия решений
  - ✔ Система вознаграждений
  - ✔ Изучение последовательности действий

Обучение с учителем



Цель –
обучить модель
на помеченных
обучающих
данных для
получения
прогнозов
на новых данных.

Обучение с учителем. Категории задач

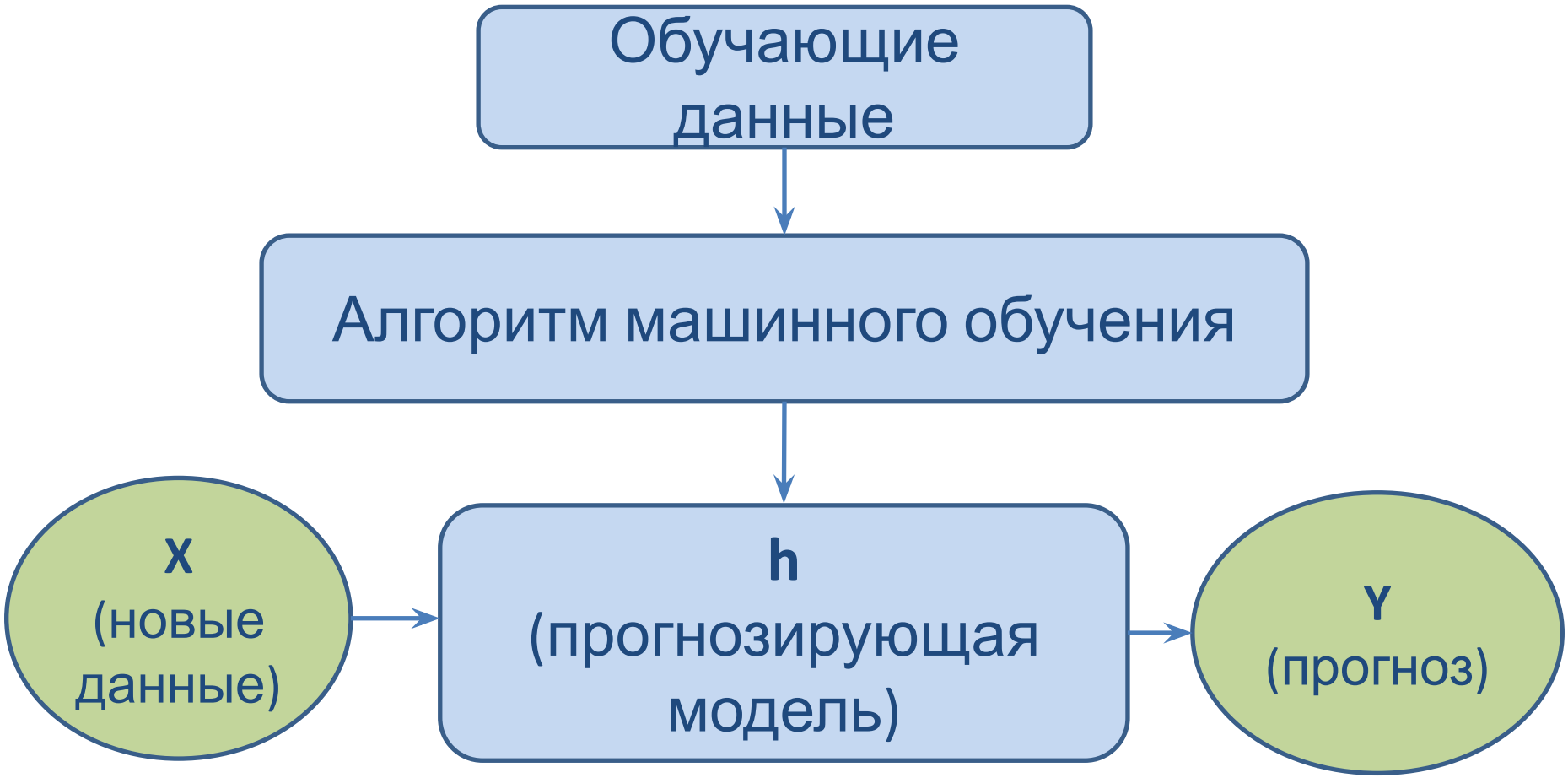
Регрессия – пытаемся предсказать результат, являющийся непрерывной функцией.

- Предсказать цену на жилье по результатам прошлых сделок.
- По фотографии предсказать возраст.

Классификация – пытаемся предсказать класс образца - категориальная метка (дискретный выход).

- Распознавание рукописных символов.
- Спам-фильтр.
- Определение типа опухоли.

Линейная регрессия



Одномерная линейная регрессия

Задача:

обучить компьютер переводить мили в км. Предположим, что формула неизвестна, но знаем, что зависимость линейная.

Тренировочные данные

Номер примера	мили	км
1	1	1.609344]
2	2	3.218688
3	3	4.828032
4	4	6.437376
5	6	9.656064
6	9	14.484096

Одномерная линейная регрессия

- Введем обозначения:

  X - входные данные,

  Y – выходные,

  m – количество тренировочных наборов,

  $x^{(i)}$ и $y^{(i)}$ - i -ый набор.

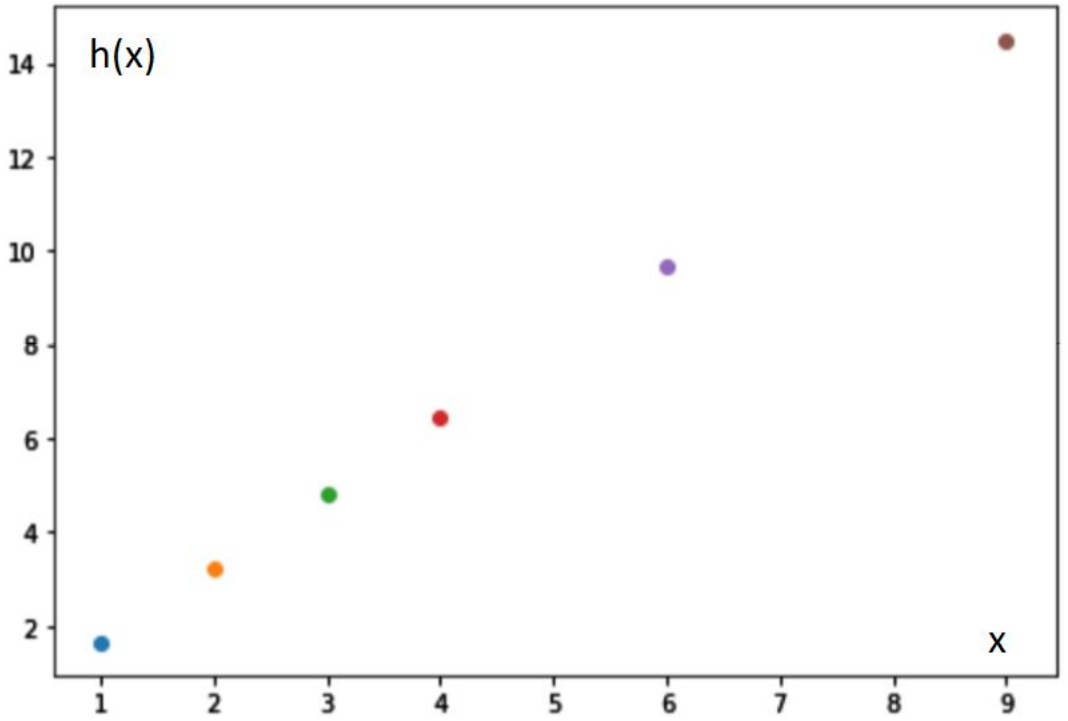
Функция перевода из миль в км должна выглядеть так:

h(X) = w X

Хотим подобрать такое значение **w**, чтобы разница **h(X)** и **Y** была минимальной (в среднем по всем тренировочным наборам).

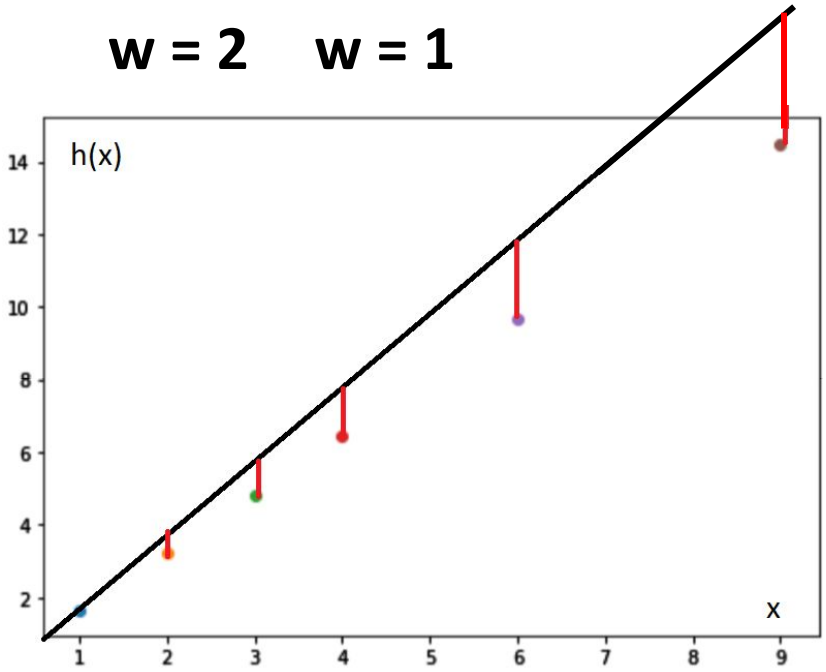
Тренировочный набор

X	Y
[[1.	1.609344]
[2.	3.218688]
[3.	4.828032]
[4.	6.437376]
[6.	9.656064]
[9.	14.484096]]

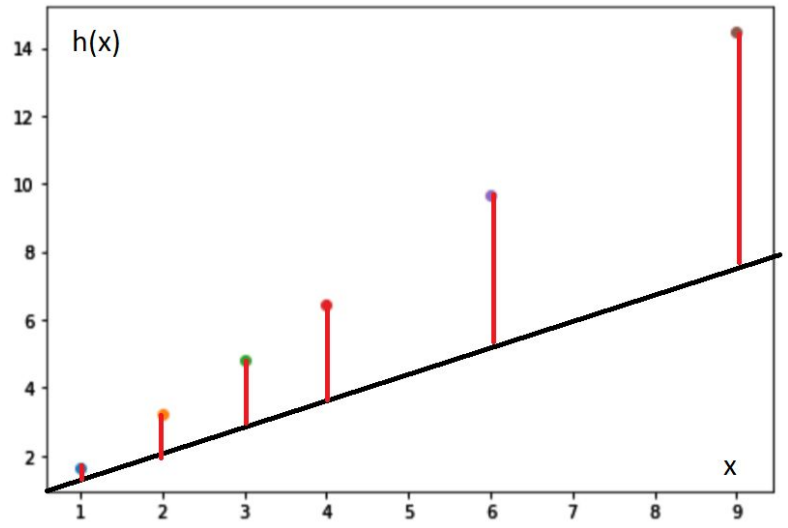


Подбираем $\mathbf{w_1}$

w = 2



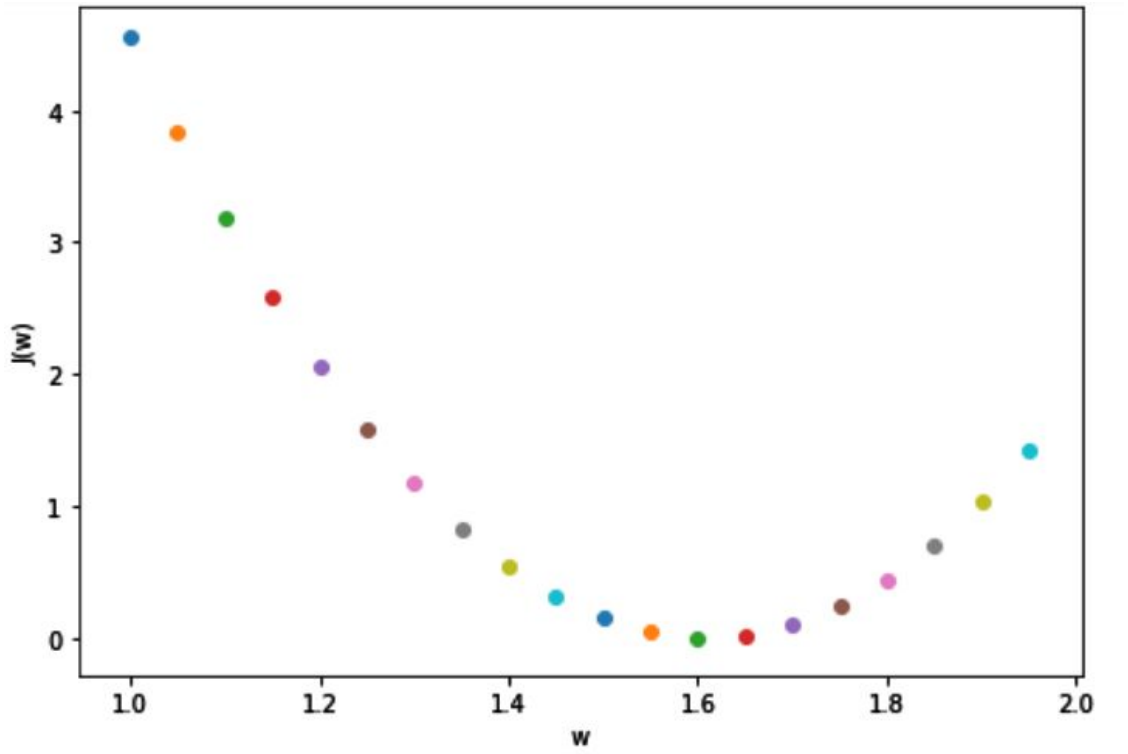
w = 1



Результат принято оценивать по квадратичной функции стоимости ошибки:

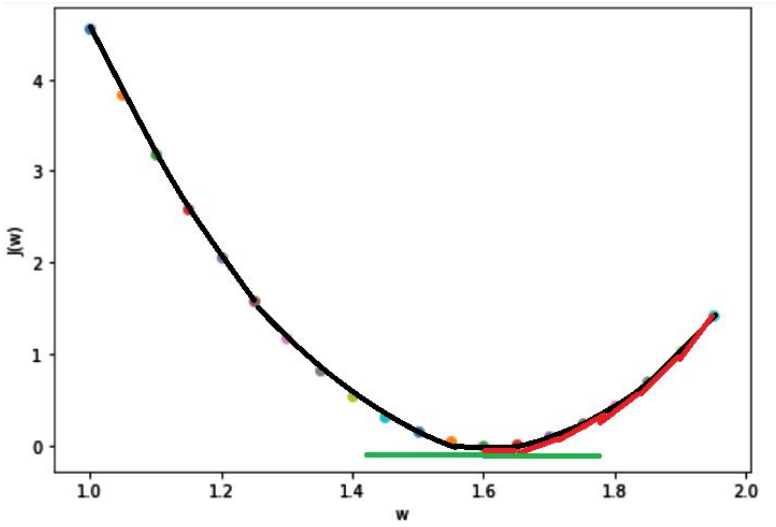
$$J(w) = \frac{1}{2m}\sum_{i=1}^{m}\left(h\left(x^{(i)}\right) - y^{(i)}\right)^2$$

Точечная диаграмма **J(w)**



$$J(w) = \frac{1}{2m}\sum_{i=1}^{m}(h(x^{(i)}) - y^{(i)})^2$$

Градиентный спуск



$$\frac{\partial}{\partial w} \mathbf{J(w)} = \frac{\partial}{\partial w} \frac{1}{2m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2$$

Будем обновлять параметр w до тех пор, пока градиент не станет равным нулю (с некоторой погрешностью ε).

$$J(w) = \frac{1}{2m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2$$

$$w = w - \alpha \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)}) x^{(i)}$$

h(x)= *w* x

Здесь α - скорость обучения.

Градиентный спуск

Если рассмотрим общий случай:
$\mathbf{h(x)= w_0 + w_1 x}$

$$\frac{\partial}{\partial w_j} \mathbf{J(w_0, w_1)} = \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2 =$$

$$= \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^{m} (\mathbf{w_0} + \mathbf{w_1} x^{(i)} - y^{(i)})^2$$

Получим частные производные:

$$\mathbf{j = 0:} \quad \frac{\partial}{\partial w_0} \mathbf{J(w_0, w_1)} = \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})$$

$$\mathbf{j = 1:} \quad \frac{\partial}{\partial w_1} \mathbf{J(w_0, w_1)} = \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)}) x^{(i)}$$

Формулы для пересчета параметров:

$$w_0 = w_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})$$

$$w_1 = w_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)}) x^{(i)}$$

Как прочитать данные из файла

```
import numpy as np
# тренировочные данные - таблица значений: мили, км
data = np.loadtxt('data.txt')
X = data[:, 0].copy().reshape((-1, 1))   # нулевой столбец
Y = data[:, 1].copy().reshape((-1, 1))   # первый столбец
# начальное значение параметров –любое
W = np.array([0, 0.5])            # можно без массива
alpha = 0.05                      # скорость обучения
eps = 1e-10      # при таком изменении J останавливаемся
```

Задание

Постройте модель с регулируемыми параметрами для решения рассмотренной задачи.

Входные данные находятся в файле data.txt на сервере в общей папке (можете создать свои данные с тренировочными примерами) .

Используйте матричные операции вместо циклов.

Выведите полученные параметры w0 и w1.