

## Домашнее задание №8: «Недвижимость и регрессия»

Дедлайн 1 (20 баллов): 7 мая, 23:59

Дедлайн 2 (10 баллов): 14 мая, 23:59

Домашнее задание нужно написать на Python и сдать в виде одного файла. Правило именования файла: `name_surname_8.py`. Например, если вас зовут Иван Петров, то имя файла должно быть: `ivan_petrov_8.py`.

---



Рис. 1: Newbury Street, Boston. Источник: <https://flickr.com/rudiriet/119512463>

Рынок недвижимости в любой стране — явление непредсказуемое. Цены на жильё зависят от миллиона факторов, учесть все из которых невозможно, но мы всё-таки попробуем.

По ссылке<sup>1</sup> находятся данные о стоимости недвижимости в пригородах Бостона. Последняя колонка каждой строки — стоимость объекта в долларах. Значения остальных колонок указаны в заголовке файла.

<sup>1</sup><https://gist.github.com/superbobry/1529fb37998d74c6679a>

1 Реализуйте обучение коэффициентов линейной регрессии с помощью нормальной системы уравнений. Структура класса приведена ниже:

```
class NormalLR:
    def fit(self, X, y):
        # ...
        self.weights = weights
        return self

    def predict(self, X):
        # ...
        return y
```

Дополните реализацию методом predict, который применяет линейную регрессию к переданной матрице X.

Для отладки алгоритма можно воспользоваться функцией sample, порождающей случайную выборку с указанными коэффициентами регрессии.

```
import numpy as np
```

```
def sample(size, *, weights):
    X = np.ones((size, 2))
    X[:, 1] = np.random.gamma(4., 2., size)
    y = X.dot(np.asarray(weights))
    y += np.random.normal(0, 1, size)
    return X[:, 1:], y
```

С помощью функции sample и библиотеки matplotlib можно визуализировать результаты работы алгоритма:

```
from matplotlib import pyplot as plt
```

```
X, y_true = sample(size, weights=[24., 42.])
lr.fit(X, y_true)
plt.scatter(X, y_true)
plt.plot(X, lr.predict(X), color="red")
plt.show()
```

2 Реализуйте метод градиентного спуска для обучения параметров линейной регрессии. Шаг градиентного спуска:

$$w_j^{(t+1)} = w_j^{(t)} - \frac{\alpha}{l} \sum_{i=1}^l (\langle x_i, w^{(t)} \rangle - y_i) x_{ij}$$

Убедиться в корректности шага можно, вычислив частные производные по  $w_j$  от функции потерь

$$Q(w, X^l) = \frac{1}{l} \sum_{i=1}^l (\langle x_i, w \rangle - y_i)^2.$$

Обратите внимание, что в качестве функции потерь используется среднее значение квадрата ошибки, а не сумма. Такая форма функции потерь удобнее для реализации.

Структура класса приведена ниже:

```

class GradientLR(NormalLR):
    def __init__(self, *, alpha):
        if alpha <= 0:
            raise ValueError("alpha should be positive")
        self.alpha = alpha
        self.threshold = alpha / 100

    def fit(self, X, y):
        # ...
        self.weights = weights
        return self

```

3 Реализуйте функцию `mse`, принимающую вектор истинных значений `y_true` и вектор предсказаний линейной регрессии `y_pred`. Результатом функции является среднее значение квадрата разности между компонентами векторов.

4 Примените две реализации линейной регрессии к данным, полученным с помощью функции `sample`. Попробуйте использовать выборки разных размеров, например, 128, 256, 512, 1024.

Ответьте на вопросы.

- Какой из подходов имеет меньшее значение средней ошибки?
- Как ведут себя алгоритмы в зависимости от размера выборки?
- Что можно сказать о времени работы каждого из алгоритмов?

5 Примените две реализации линейной регрессии к данным о стоимости недвижимости.

Для обучения и оценки следует использовать две различные выборки. Вам может быть полезна функция `train_test_split` из предыдущего домашнего задания.

Ответьте на вопросы.

- Какой из подходов имеет меньшее значение средней ошибки? Сопласуется ли результат с полученным на симулированных данных?
- Как вы считаете, требуется ли нормировка признаков в случае данных о стоимости недвижимости? Объясните, почему.
- Интерпретируйте коэффициенты регрессии, полученные одним из алгоритмов. Какой из признаков даёт наибольший вклад в стоимость недвижимости?
- Какой из алгоритмов лучше подходит для задачи предсказания стоимости? Почему?

6\* Модифицируйте оба алгоритма так, чтобы они использовали регуляризацию весов линейной регрессии с помощью  $L^2$ -нормы. Опишите влияние регуляризации на значение среднего квадрата ошибки. Как писал классик: «To regularize or not to regularize, that is the question [...]».

Это задание не обязательно для выполнения, но, если вы его сделаете, то получите дополнительные 10 баллов.