Санкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет программной инженерии и компьютерной техники

# Отчет по лабораторной работы 4

По дисциплине «Системы искусственного интеллекта»

Выполнила:
Мозговая Лариса Андреевна
Группа P33311

Преподаватель:
Кугаевских Александр Владимирович

Санкт-Петербург, 2023

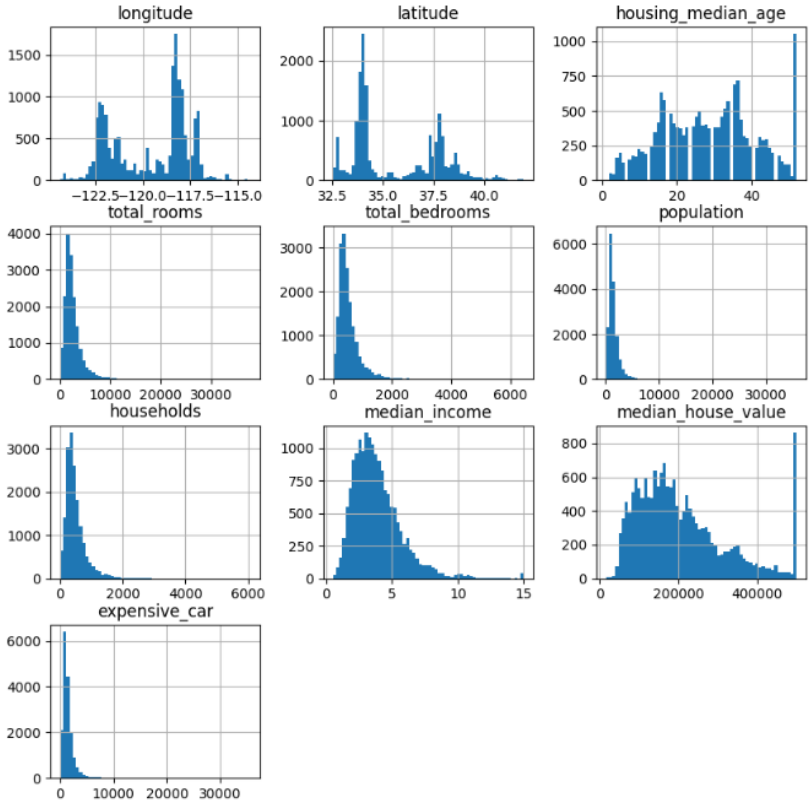
## Задание

- Использовать набор данных о жилье в Калифорнии Скачать тут
- Получите и визуализируйте статистику по датасету (включая количество, среднее значение, стандартное отклонение, минимум, максимум и различные квантили).
- Проведите предварительную обработку данных, включая обработку отсутствующих значений, кодирование категориальных признаков и нормировка.
- Разделите данные на обучающий и тестовый наборы данных.
- Реализуйте линейную регрессию с использованием метода наименьших квадратов без использования сторонних библиотек, кроме NumPy и Pandas. Использовать минимизацию суммы квадратов разностей между фактическими и предсказанными значениями для нахождения оптимальных коэффициентов.
- Постройте **три модели** с различными наборами признаков.
- Для каждой модели проведите оценку производительности, используя метрику коэффициент детерминации, чтобы измерить, насколько хорошо модель соответствует данным.
- Сравните результаты трех моделей и сделайте выводы о том, какие признаки работают лучше всего для каждой модели.
- Бонусное задание
    - Ввести синтетический признак при построении модели

## Этапы реализации и пояснения:

Сначала я импортировала нужные библиотеки, далее создала синтаксический признак и вывела статистику по датасету.

```
data['expensive_car'] = data['population'] +
data['housing_median_age'] + data['median_income']
```



Дальше я сделала предварительную обработку и нормировку данных

```
# Проверка наличия отсутствующих значений
missing_values = data.isnull().sum()
print(missing_values)

# Заполнение отсутствующих значений средними значениями
data.fillna(data.mean(), inplace=True)

# Нормировка данных
for column in ['longitude', 'latitude', 'housing_median_age',
'total_rooms', 'total_bedrooms', 'population', 'households',
'median_income']:
    mean = data[column].mean()
    std = data[column].std()
    data[column] = (data[column] - mean) / std
```

Функцией isnull, я проверила что в датасете нет отсутствующих значений, но на всякий случай заполнила их средними значениями.

Для нормировки сначала мы вычисляем среднее значение (mean) и стандартное отклонение (std) для каждого признака. Затем мы вычитаем среднее значение из каждой точки данных и делим на стандартное отклонение, чтобы получить нормированные данные.

Далее я разделила данные на обучающий и тестовый набор

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

Таким образом, мы получаем два набора данных: X_train, y_train - обучающий набор, и X_test, y_test - тестовый набор.

Далее я создаю линейную регрессию, для этого я добавляю столбец с единицами и вычисляю коэффициент линейной регрессии методом наименьших квадратов.

```
def linear_regression(X_train, y_train):
    # Добавим столбец с единицами для учёта свободного члена
    X = np.column_stack((np.ones(X_train.shape[0]), X_train))

    # Вычислим коэффициентов линейной регрессии методом
наименьших квадратов
    coefficients =
np.linalg.inv(X.T.dot(X)).dot(X.T).dot(y_train)

    return coefficients
```

Получаю предсказания для тестового набора данных

```
def predict(X_test, coefficients):
    # Добавим столбец с единицами для учёта свободного члена
    X = np.column_stack((np.ones(X_test.shape[0]), X_test))

    # Предскажем значения
    y_pred = X.dot(coefficients)
    return y_pred
```

Чтобы оценить производительность, я использую коэффициент детерминации (R^2)

```
def r2_score(y_test, y_pred):
    total_variance = np.sum((y_test - np.mean(y_test)) ** 2)
    residual_variance = np.sum((y_test - y_pred) ** 2)
    r2 = 1 - (residual_variance / total_variance)
    return r2
```

И в конце я реализовала несколько моделей по разным признакам

## Вывод:

Можно сделать вывод, что коэффициент детерминации сильно повысился за счёт median_income, значит цена дома сильно зависит от среднего заработка.