МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ
«Санкт-Петербургский национальный исследовательский
университет информационных технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И
КОМПЬЮТЕРНОЙ ТЕХНИКИ

Отчёт по лабораторной работе №3

по дисциплине

«Системы искусственного интеллекта»

Выполнил ***:*** Студент группы
P3334 Баянов Равиль
Динарович
Преподаватель:
Авдюшина А. Е.

Санкт-Петербург, 2024

Оглавление

Задание

Вариант: набор данных о жилье в Калифорнии, так как номер в группе 4.

- Получите и визуализируйте (графически) 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 (для использования коэффициентов использовать библиотеки тоже нельзя). 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

Статистика по датасету

	longitude	latitude	housing_median_age	total_rooms
count	17000.000000	17000.000000	17000.000000	17000.000000
mean	-119.562108	35.625225	28.589353	2643.664412
std	2.005166	2.137340	12.586937	2179.947071
min	-124.350000	32.540000	1.000000	2.000000
25%	-121.790000	33.930000	18.000000	1462.000000
50%	-118.490000	34.250000	29.000000	2127.000000
75%	-118.000000	37.720000	37.000000	3151.250000
max	-114.310000	41.950000	52.000000	37937.000000

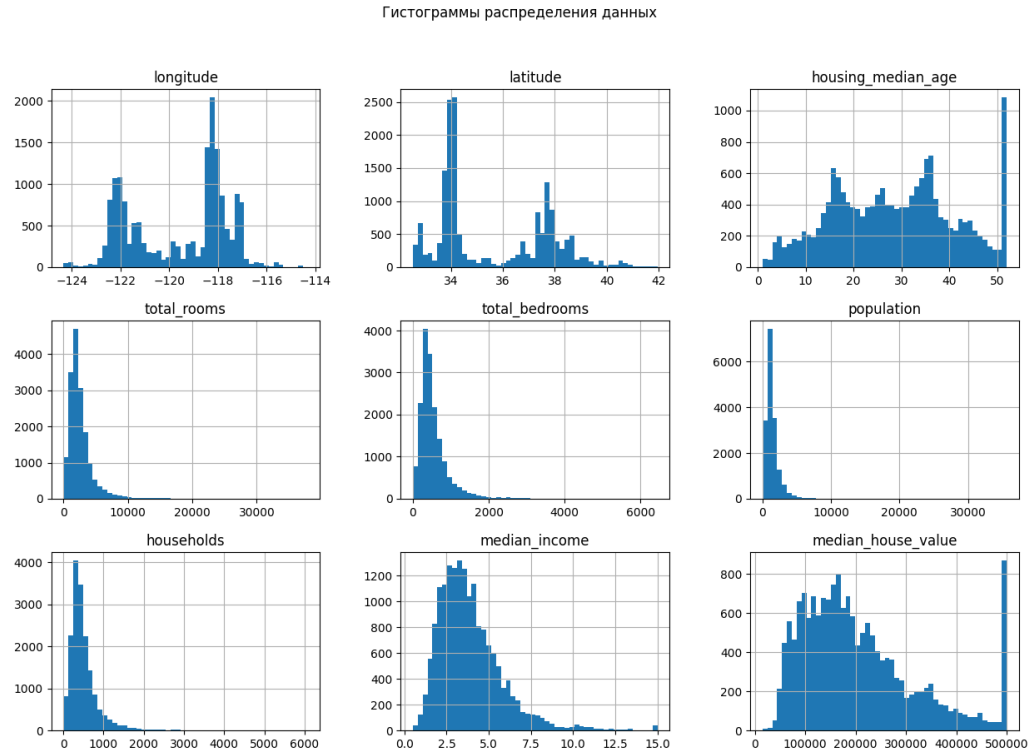
	total_bedrooms	population	households	median_income
count	17000.000000	17000.000000	17000.000000	17000.000000
mean	539.410824	1429.573941	501.221941	3.883578
std	421.499452	1147.852959	384.520841	1.908157
min	1.000000	3.000000	1.000000	0.499900
25%	297.000000	790.000000	282.000000	2.566375
50%	434.000000	1167.000000	409.000000	3.544600
75%	648.250000	1721.000000	605.250000	4.767000
max	6445.000000	35682.000000	6082.000000	15.000100

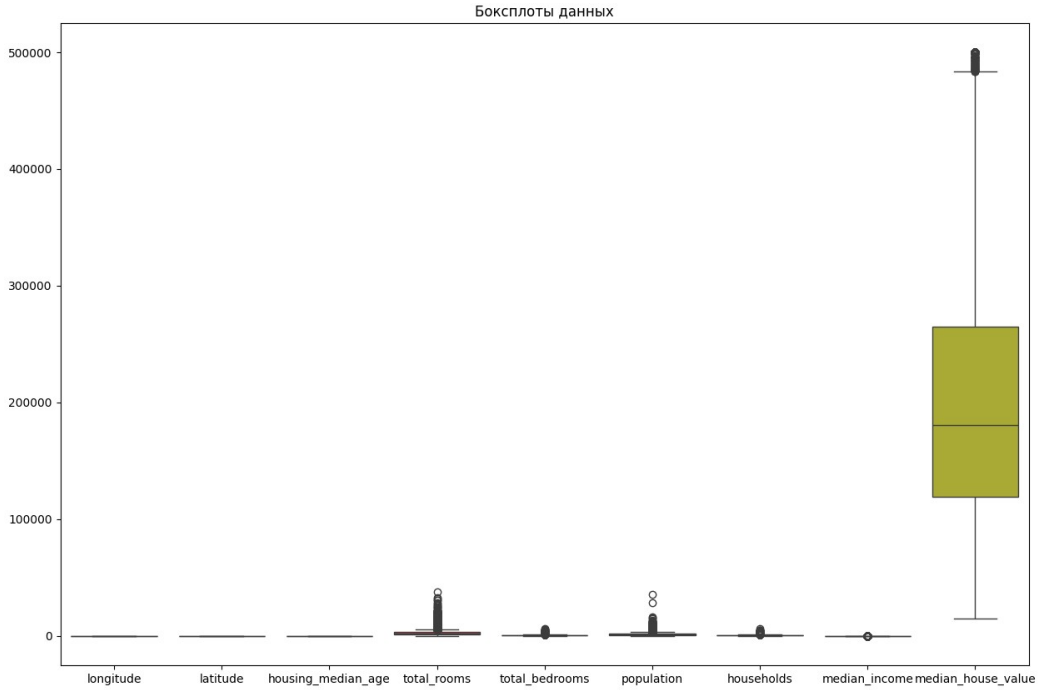
	median_house_value
count	17000.000000
mean	207300.912353
std	115983.764387
min	14999.000000
25%	119400.000000
50%	180400.000000
75%	265000.000000
max	500001.000000

Где:

- count – кол-во значений
- mean – среднее
- std – стандартное отклонение (среднеквадратическое отклонение от мат. ожидания)
- min – минимум
- max – максимум
- Значения процентов - квантили столбцов

Графики, описывающие датасет:





Код на Python реализации модели

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn.model_selection import train_test_split


# Функция расчёта линейной регрессии
def linear_regression(X, y):
    X_tran = X.T
    return np.linalg.inv(X_tran.dot(X)).dot(X_tran).dot(y)


# Функция расчёта коэфициента детерминации для модели
def coefficient_of_determination(X, y, odds):
    y_pred = X.dot(odds)
    sse = np.sum((y - y_pred) ** 2)
    sst = np.sum((y - np.mean(y)) ** 2)
    return 1 - sse / sst


data = pd.read_csv('data/california_housing_train.csv')
pd.set_option('display.max_columns', None)
stats = data.describe()
print(stats)

# Отрисовка диаграмм распределений каждого столбца из датасета
data.hist(bins=50, figsize=(15, 10))
plt.suptitle('Гистограммы распределения данных')
plt.show()

# Отрисовка диаграммы "ящик с усами" для каждого столбца из датасета
plt.figure(figsize=(15, 10))
sns.boxplot(data=data)
plt.title('Боксплоты данных')
plt.show()

# Удаление отсутствующих значений
data.dropna()

# Нормализация данных
data = (data - data.mean()) / data.std()

# Добавление синтетического признака
data['total_rooms_without_bedrooms'] = data['total_rooms'] -
data['total_bedrooms']

# Поделим даные на тестовый и обучающий наборы данных
train_data, test_data = train_test_split(data, test_size=0.2,
random_state=30)

# Создадим первую модель с небольшим набором признаков
X_train_1 = train_data[['total_rooms', 'median_income']].values
y_train_1 = train_data['median_house_value'].values

# Добавление столбца для смещения (добавляем столбец с единицами)
X_train_1 = np.c_[np.ones(X_train_1.shape[0]), X_train_1]

# Получение коэффициентов
odds_1 = linear_regression(X_train_1, y_train_1)

# Получение коэффициента детерминации для оценки модели 1
r_2_1 = coefficient_of_determination(X_train_1, y_train_1, odds_1)
print(f'R^2 для первой модели: {r_2_1}')

# Создадим вторую модель с чуть большим набором признаков
```

```
X_train_2 = train_data[['total_rooms', 'median_income',
'housing_median_age']].values
y_train_2 = train_data['median_house_value'].values

# Добавление столбца для смещения (добавляем столбец с единицами)
X_train_2 = np.c_[np.ones(X_train_2.shape[0]), X_train_2]

# Получение коэффициентов
odds_2 = linear_regression(X_train_2, y_train_2)

# Получение коэффициента детерминации для оценки модели 2
r_2_2 = coefficient_of_determination(X_train_2, y_train_2, odds_2)
print(f'R^2 для второй модели: {r_2_2}')

# Создадим третью модель с этими же коэффициентами + синтетический признак
X_train_3 = train_data[['total_rooms', 'median_income',
'housing_median_age', 'total_rooms_without_bedrooms']].values
y_train_3 = train_data['median_house_value'].values

# Добавление столбца для смещения (добавляем столбец с единицами)
X_train_3 = np.c_[np.ones(X_train_3.shape[0]), X_train_3]

# Получение коэффициентов
odds_3 = linear_regression(X_train_3, y_train_3)

# Получение коэффициентов детерминации для оценки модели 3
r_2_3 = coefficient_of_determination(X_train_3, y_train_3, odds_3)
print(f'R^2 для третьей модели: {r_2_3}')

print(
    f"Сравнение моделей:\nМодель 1 R^2: {r_2_1}\nМодель 2 R^2: {r_2_2}\
nМодель 3 R^2: {r_2_3} (с синтетическим признаком)")

# А сейчас протестируем наши модели, которые мы построили
# 1
X_test_1 = test_data[['total_rooms', 'median_income']].values
y_test_1 = test_data['median_house_value'].values
X_test_1 = np.c_[np.ones(X_test_1.shape[0]), X_test_1]

r_2_1_test = coefficient_of_determination(X_test_1, y_test_1, odds_1)
print(f'R^2 для тестовых данных первой модели: {r_2_1_test}')

# 2
X_test_2 = test_data[['total_rooms', 'median_income',
'housing_median_age']].values
y_test_2 = test_data['median_house_value'].values
X_test_2 = np.c_[np.ones(X_test_2.shape[0]), X_test_2]

r_2_2_test = coefficient_of_determination(X_test_2, y_test_2, odds_2)
print(f'R^2 для тестовых данных второй модели: {r_2_2_test}')

# 3
X_test_3 = test_data[['total_rooms', 'median_income', 'housing_median_age',
'total_rooms_without_bedrooms']].values
y_test_3 = test_data['median_house_value'].values
X_test_3 = np.c_[np.ones(X_test_3.shape[0]), X_test_3]

r_2_3_test = coefficient_of_determination(X_test_3, y_test_3, odds_3)
print(f'R^2 для тестовых данных третьей модели: {r_2_3_test}')

print(
    f"Сравнение моделей на тестовых данных:\nМодель 1 R^2: {r_2_1_test}\
nМодель 2 R^2: {r_2_2_test}\nМодель 3 R^2: {r_2_3_test} (с синтетическим
признаком)")
```

Показатели детерминации модели

```
R^2 для первой модели: 0.47842815120849735
R^2 для второй модели: 0.5184622405521854
R^2 для третьей модели: 0.5522503959624603
Сравнение моделей:
Модель 1 R^2: 0.47842815120849735
Модель 2 R^2: 0.5184622405521854
Модель 3 R^2: 0.5522503959624603 (с синтетическим признаком)
R^2 для тестовых данных первой модели: 0.47971026264117556
R^2 для тестовых данных второй модели: 0.517897215988669
R^2 для тестовых данных третьей модели: 0.5533400004490683
Сравнение моделей на тестовых данных:
Модель 1 R^2: 0.47971026264117556
Модель 2 R^2: 0.517897215988669
Модель 3 R^2: 0.5533400004490683 (с синтетическим признаком)
```

Вывод

В этой лабораторной работе мы рассмотрели метод машинного обучения – линейная регрессия. Узнали, как с помощью тестового набора данных можно спрогнозировать последующие данные и оценили качество, построенных моделей с помощью коэффициента детерминации.