Московский государственный технический университет им. Н. Э. Баумана

Курс «Технологии машинного обучения»

Отчёт по лабораторной работе №1

Выполнил:
Лупарев С. В.
группа ИУ5-63Б

Проверил:
Гапанюк Ю.Е.

Дата:

Подпись:

Дата:

Подпись:

Москва, 2025 г.

Цель лабораторной работы

Цель: изучение различных методов визуализация данных.

Краткое описание

Построение основных графиков, входящих в этап разведочного анализа данных.

Рекомендуемые инструментальные средства можно посмотреть здесь.

Задание

- Выбрать набор данных (датасет). Вы можете найти список свободно распространяемых датасетов здесь.
- Для первой лабораторной работы рекомендуется использовать датасет без пропусков в данных, например из Scikit-learn.
- Пример преобразования датасетов Scikit-learn в Pandas Dataframe можно посмотреть здесь.

Для лабораторных работ не рекомендуется выбирать датасеты большого размера.

- Создать ноутбук, который содержит следующие разделы:

1. Текстовое описание выбранного Вами набора данных.
2. Основные характеристики датасета.
3. Визуальное исследование датасета.
4. Информация о корреляции признаков.

- Сформировать отчет и разместить его в своем репозитории на github.

[ ]:
```
Data Set Characteristics:

Number of Instances: 20640

Number of Attributes: 8 numeric, predictive attributes and the target

Attribute Information:
    1) MedInc median income in block group
    2) HouseAge median house age in block group
    3) AveRooms average number of rooms per household
    4) AveBedrms average number of bedrooms per household
    5) Population block group population
    6) AveOccup average number of household members
    7) Latitude block group latitude
    8) Longitude block group longitude

Missing Attribute Values: None

This dataset was obtained from the StatLib repository. https://www.dcc.fc.up.pt/~ltorgo/Regression/cal_housing.html

The target variable is the median house value for California districts, expressed in hundreds of thousands of dollars ($100,000).
```

[14]:
```
import pandas as pd
import numpy as np
import sklearn
import matplotlib
```

[15]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

[9]:
```
from sklearn.datasets import fetch_california_housing
housing = fetch_california_housing()
df = pd.DataFrame(data= np.c_[housing['data'], housing['target']],
                  columns= housing['feature_names'] + ['target'])
df
```

[9]:

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude	target
0	8.3252	41.0	6.984127	1.023810	322.0	2.555556	37.88	-122.23	4.526
1	8.3014	21.0	6.238137	0.971880	2401.0	2.109842	37.86	-122.22	3.585
2	7.2574	52.0	8.288136	1.073446	496.0	2.802260	37.85	-122.24	3.521
3	5.6431	52.0	5.817352	1.073059	558.0	2.547945	37.85	-122.25	3.413
4	3.8462	52.0	6.281853	1.081081	565.0	2.181467	37.85	-122.25	3.422
...	...	...	...	...	...	...	...	...	...
20635	1.5603	25.0	5.045455	1.133333	845.0	2.560606	39.48	-121.09	0.781
20636	2.5568	18.0	6.114035	1.315789	356.0	3.122807	39.49	-121.21	0.771
20637	1.7000	17.0	5.205543	1.120092	1007.0	2.325635	39.43	-121.22	0.923
20638	1.8672	18.0	5.329513	1.171920	741.0	2.123209	39.43	-121.32	0.847
20639	2.3886	16.0	5.254717	1.162264	1387.0	2.616981	39.37	-121.24	0.894

20640 rows × 9 columns

28]:
```
df = df.rename(columns={'target':'MedHouseVal'})
```

29]:
```
df.describe().T
```

[29]:
```
df.describe().T
```

[29]:

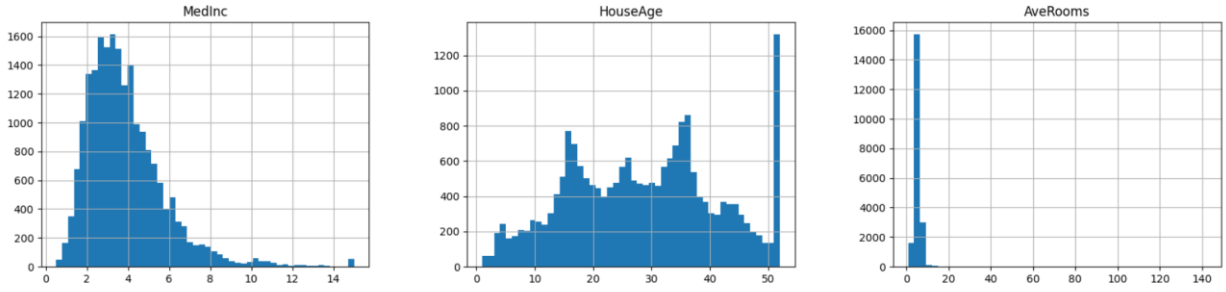
	count	mean	std	min	25%	50%	75%	max
MedInc	20640.0	3.870671	1.899822	0.499900	2.563400	3.534800	4.743250	15.000100
HouseAge	20640.0	28.639486	12.585558	1.000000	18.000000	29.000000	37.000000	52.000000
AveRooms	20640.0	5.429000	2.474173	0.846154	4.440716	5.229129	6.052381	141.909091
AveBedrms	20640.0	1.096675	0.473911	0.333333	1.006079	1.048780	1.099526	34.066667
Population	20640.0	1425.476744	1132.462122	3.000000	787.000000	1166.000000	1725.000000	35682.000000
AveOccup	20640.0	3.070655	10.386050	0.692308	2.429741	2.818116	3.282261	1243.333333
Latitude	20640.0	35.631861	2.135952	32.540000	33.930000	34.260000	37.710000	41.950000
Longitude	20640.0	-119.569704	2.003532	-124.350000	-121.800000	-118.490000	-118.010000	-114.310000
MedHouseVal	20640.0	2.068558	1.153956	0.149990	1.196000	1.797000	2.647250	5.000010

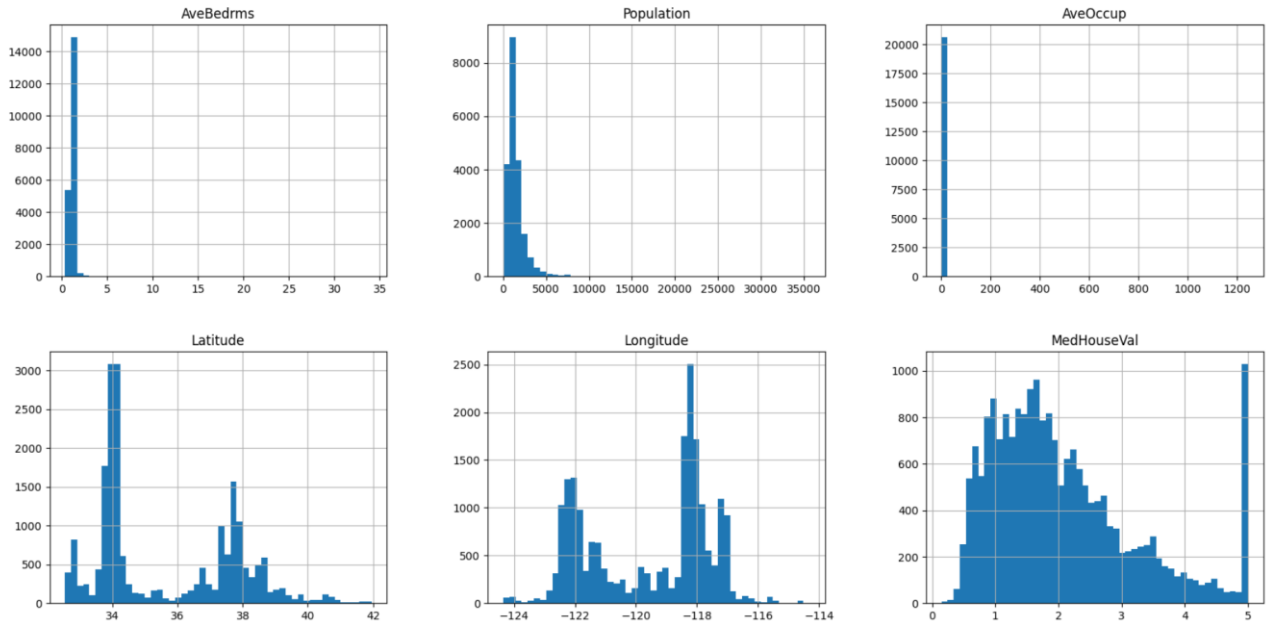
[30]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 9 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   MedInc       20640 non-null  float64
 1   HouseAge     20640 non-null  float64
 2   AveRooms     20640 non-null  float64
 3   AveBedrms    20640 non-null  float64
 4   Population   20640 non-null  float64
 5   AveOccup     20640 non-null  float64
 6   Latitude     20640 non-null  float64
 7   Longitude    20640 non-null  float64
 8   MedHouseVal  20640 non-null  float64
dtypes: float64(9)
memory usage: 1.4 MB
```

[31]:
```
df.hist(bins=50, figsize=(20, 15))
```

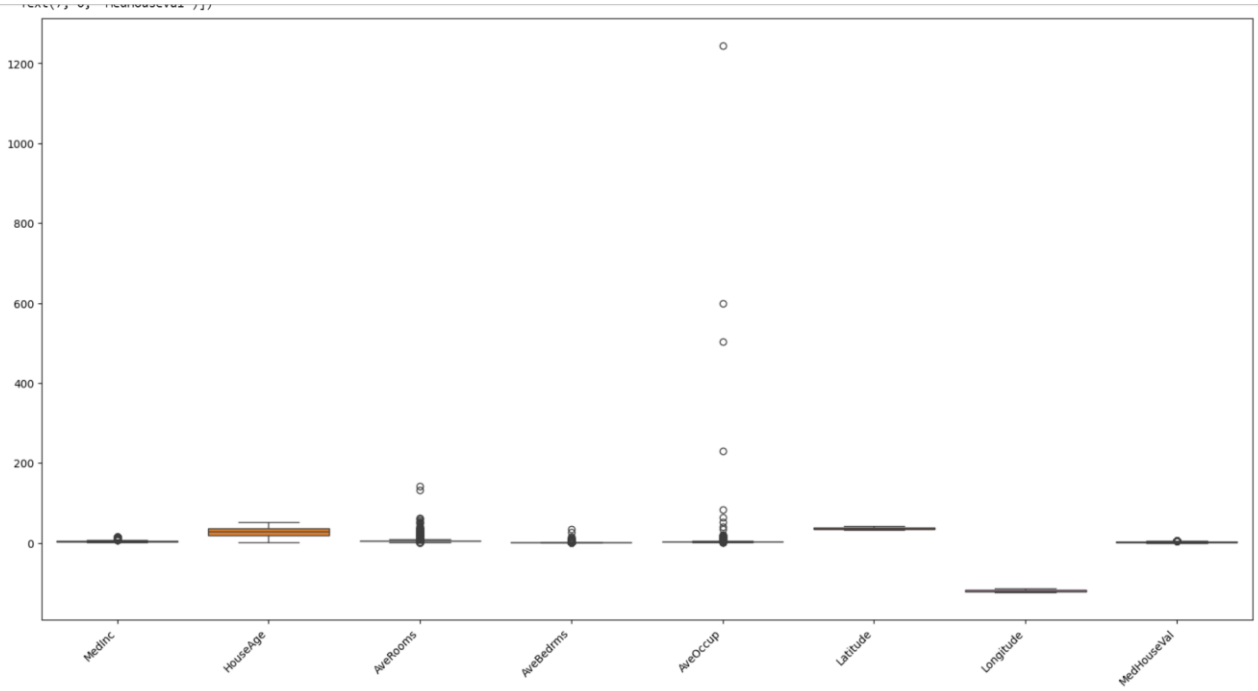
[31]:
```
array([[<Axes: title={'center': 'MedInc'}>,
        <Axes: title={'center': 'HouseAge'}>,
        <Axes: title={'center': 'AveRooms'}>],
       [<Axes: title={'center': 'AveBedrms'}>,
        <Axes: title={'center': 'Population'}>,
        <Axes: title={'center': 'AveOccup'}>],
       [<Axes: title={'center': 'Latitude'}>,
        <Axes: title={'center': 'Longitude'}>,
        <Axes: title={'center': 'MedHouseVal'}>]], dtype=object)
```



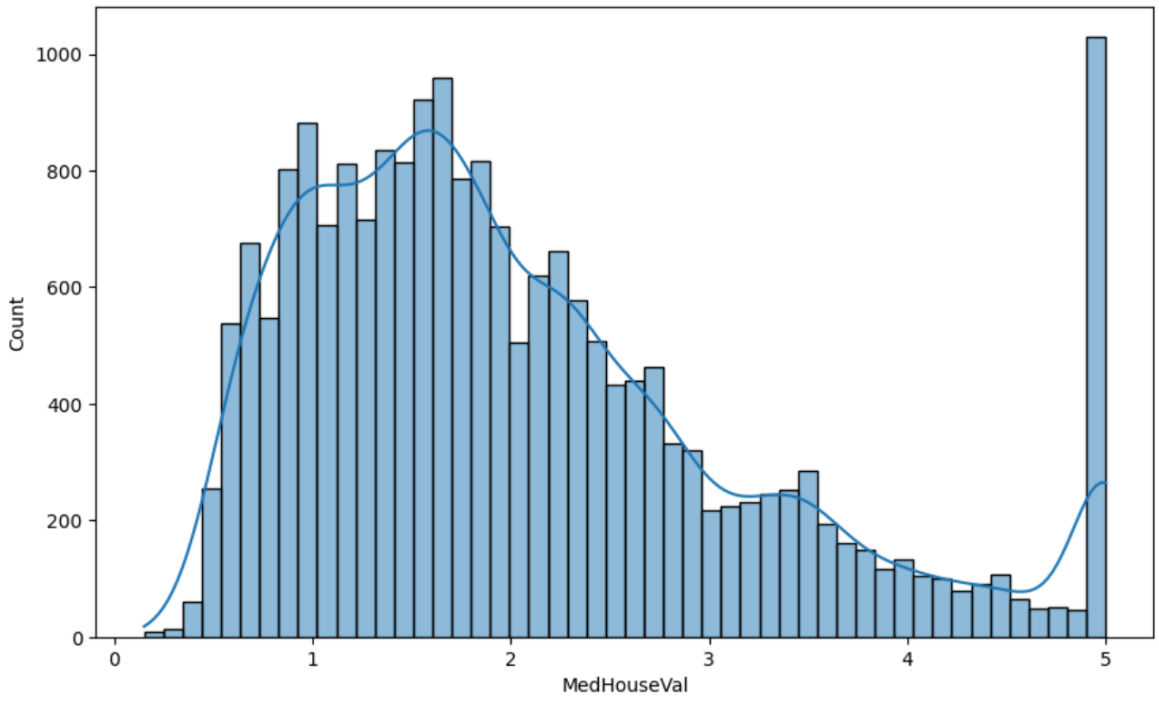


[32]:
```
temp_df = df.drop('Population', axis=1)
plt.figure(figsize=(20, 10))
sns.boxplot(data=temp_df)
plt.xticks(rotation=45, ha="right")
```

[32]:
```
([0, 1, 2, 3, 4, 5, 6, 7],
 [Text(0, 0, 'MedInc'),
  Text(1, 0, 'HouseAge'),
  Text(2, 0, 'AveRooms'),
  Text(3, 0, 'AveBedrms'),
  Text(4, 0, 'AveOccup'),
  Text(5, 0, 'Latitude'),
  Text(6, 0, 'Longitude'),
  Text(7, 0, 'MedHouseVal')])
```

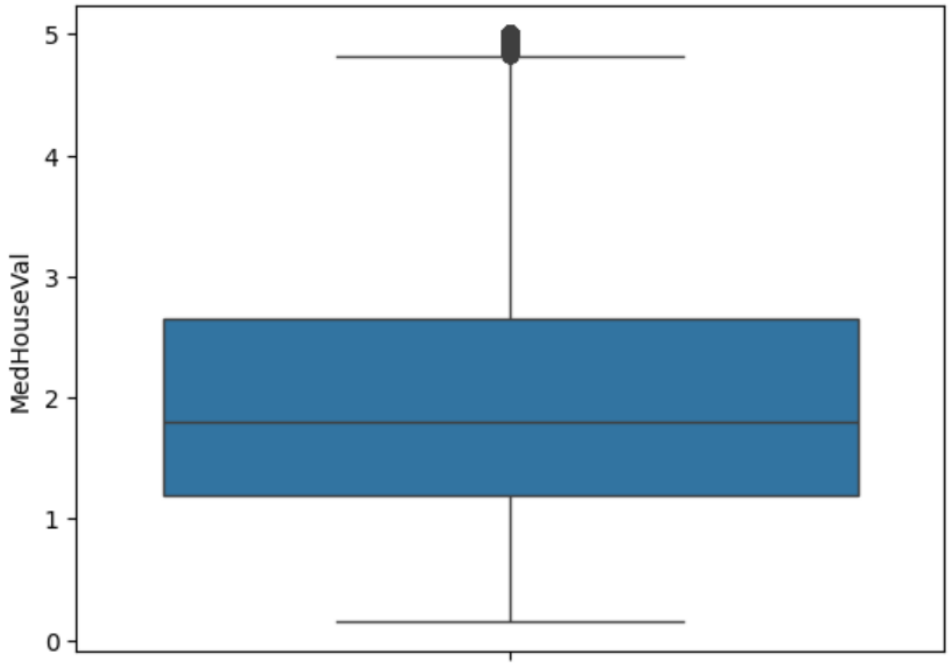


[34]:
```
plt.figure(figsize=(10, 6))
sns.histplot(df['MedHouseVal'], kde=True, bins=50)
plt.show()
```



[38]:
```
sns.boxplot(y=df['MedHouseVal'])
```

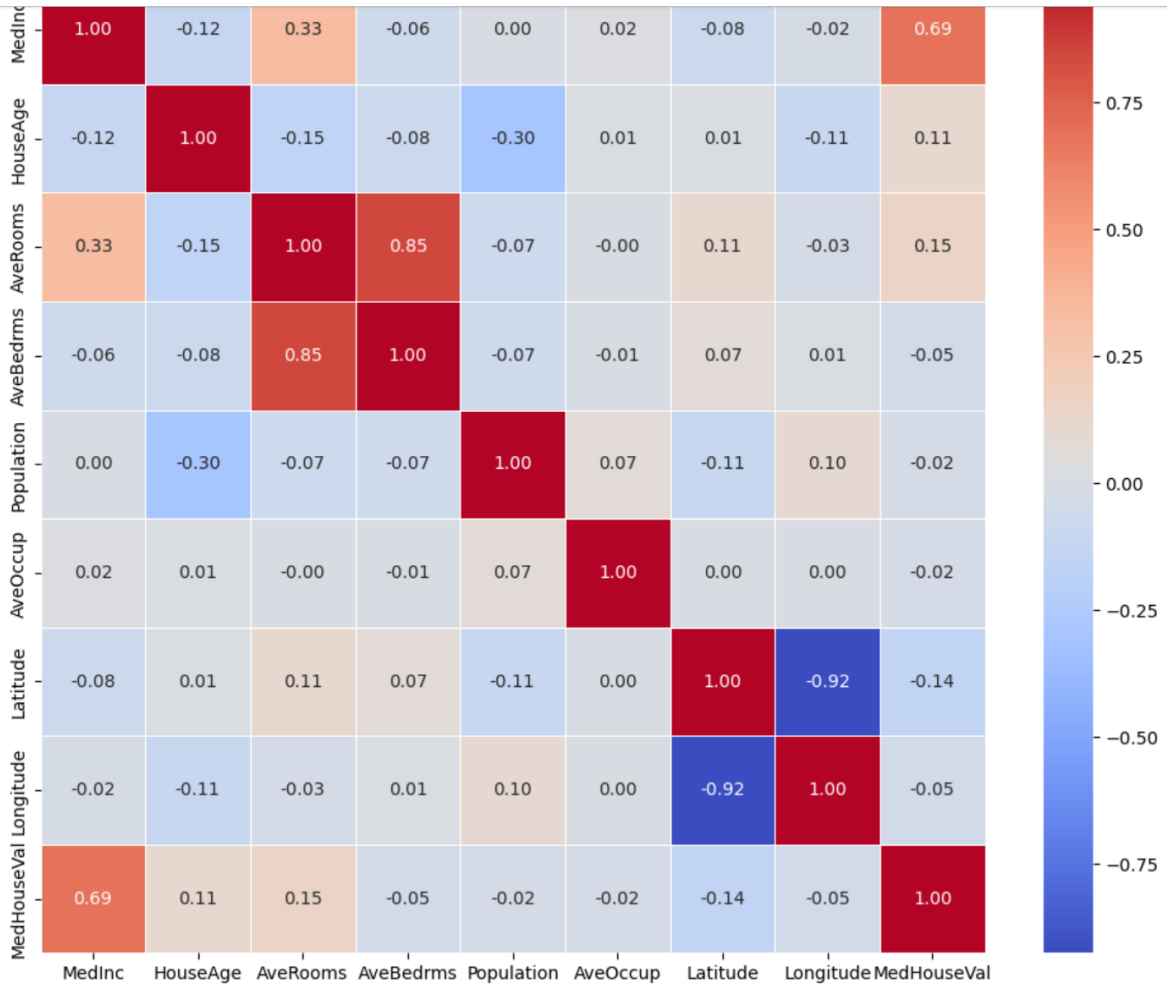
[38]:
```
<Axes: ylabel='MedHouseVal'>
```



[39]:
```
cm = df.corr()
print(cm)

plt.figure(figsize=(12, 10))
sns.heatmap(cm, annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.show()
```

```
                MedInc  HouseAge  AveRooms  AveBedrms  Population  AveOccup  \
MedInc        1.000000 -0.119034  0.326895  -0.062040    0.004834  0.018766
HouseAge     -0.119034  1.000000 -0.153277  -0.077747   -0.296244  0.013191
AveRooms      0.326895 -0.153277  1.000000   0.847621   -0.072213 -0.004852
AveBedrms    -0.062040 -0.077747  0.847621   1.000000   -0.066197 -0.006181
Population    0.004834 -0.296244 -0.072213  -0.066197    1.000000  0.069863
AveOccup      0.018766  0.013191 -0.004852  -0.006181    0.069863  1.000000
Latitude     -0.079809  0.011173  0.106389   0.069721   -0.108785  0.002366
Longitude    -0.015176 -0.108197 -0.027540   0.013344    0.099773  0.002476
MedHouseVal   0.688075  0.105623  0.151948  -0.046701   -0.024650 -0.023737
```



[43]:
```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='MedInc', y='MedHouseVal', data=df, alpha=0.5)
plt.show()
```