

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика с системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Отчёт по рубежному контролю №1
Вариант №8

Выполнил:

студент группы РТ5-61Б
Кузнецов А.В.

Подпись и дата:

Проверил:

преподаватель каф. ИУ5
Гапанюк Ю.Е.

Подпись и дата:

Москва, 2023 г.

```
from google.colab import drive
drive.mount('/content/drive', force_remount=True)
```

Mounted at /content/drive

Вариант 8 (Задача №1, Датасет №8)

Для заданного набора данных проведите корреляционный анализ. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Сделайте выводы о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

Описание датасета

Датасет содержит информацию, собранную Службой переписи населения США о жилье в районе Бостона, штат Массачусетс. Были выбраны 506 различных жилых помещений и для каждого из них указаны следующие значения (соответственно, колонки таблицы):

- CRIM - уровень преступности на душу населения по городам
- ZN - доля земель под жилую застройку зонирована под участки площадью более 25 000 кв. футов.
- INDUS - доля акров неторгового бизнеса на город.
- CHAS - Фиктивная переменная реки Чарльз (1, если участок граничит с рекой; 0 в противном случае)
- NOX - концентрация оксидов азота (частиц на 10 миллионов)
- RM - среднее количество комнат в квартире
- AGE - доля жилых единиц, построенных до 1940 г.
- DIS - взвешенные расстояния до пяти центров занятости Бостона
- RAD - индекс доступности к радиальным магистралям
- TAX - полная ставка налога на имущество на 10 000 долларов США
- PTRATIO - соотношение учеников и учителей по городам
- B - $1000 (B_k - 0,63)^2$, где B_k - доля чернокожих
- LSTAT - % населения низкого статуса
- MEDV - Средняя стоимость домов, занимаемых владельцами (тысяч долларов)

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
filename = '/content/drive/My Drive/МГТУ/6
семестр/TMO/PK1/HousingData.csv'
ds = pd.read_csv(filename)
```

```
pd.set_option('display.max_colwidth', None)
pd.set_option('display.float_format', '{:.2f}'.format)
ds = pd.DataFrame(ds)
display(ds)
```

B \	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO
0 396.90	0.01	18.00	2.31	0.00	0.54	6.58	65.20	4.09	1	296	15.30
1 396.90	0.03	0.00	7.07	0.00	0.47	6.42	78.90	4.97	2	242	17.80
2 392.83	0.03	0.00	7.07	0.00	0.47	7.18	61.10	4.97	2	242	17.80
3 394.63	0.03	0.00	2.18	0.00	0.46	7.00	45.80	6.06	3	222	18.70
4 396.90	0.07	0.00	2.18	0.00	0.46	7.15	54.20	6.06	3	222	18.70
..
501 391.99	0.06	0.00	11.93	0.00	0.57	6.59	69.10	2.48	1	273	21.00
502 396.90	0.05	0.00	11.93	0.00	0.57	6.12	76.70	2.29	1	273	21.00
503 396.90	0.06	0.00	11.93	0.00	0.57	6.98	91.00	2.17	1	273	21.00
504 393.45	0.11	0.00	11.93	0.00	0.57	6.79	89.30	2.39	1	273	21.00
505 396.90	0.05	0.00	11.93	0.00	0.57	6.03	NaN	2.50	1	273	21.00

	LSTAT	MEDV
0	4.98	24.00
1	9.14	21.60
2	4.03	34.70
3	2.94	33.40
4	NaN	36.20
...
501	NaN	22.40
502	9.08	20.60
503	5.64	23.90
504	6.48	22.00
505	7.88	11.90

```
[506 rows x 14 columns]
```

```
ds.dropna()
```

[illegible]

```

1      0.03  0.00   7.07  0.00  0.47  6.42  78.90  4.97    2  242    17.80
396.90
2      0.03  0.00   7.07  0.00  0.47  7.18  61.10  4.97    2  242    17.80
392.83
3      0.03  0.00   2.18  0.00  0.46  7.00  45.80  6.06    3  222    18.70
394.63
5      0.03  0.00   2.18  0.00  0.46  6.43  58.70  6.06    3  222    18.70
394.12
..      ...    ...    ...    ...    ...    ...    ...    ...    ...
...
499  0.18  0.00   9.69  0.00  0.58  5.57  73.50  2.40    6  391    19.20
395.77
500  0.22  0.00   9.69  0.00  0.58  6.03  79.70  2.50    6  391    19.20
396.90
502  0.05  0.00  11.93  0.00  0.57  6.12  76.70  2.29    1  273    21.00
396.90
503  0.06  0.00  11.93  0.00  0.57  6.98  91.00  2.17    1  273    21.00
396.90
504  0.11  0.00  11.93  0.00  0.57  6.79  89.30  2.39    1  273    21.00
393.45

```

```

      LSTAT  MEDV
0      4.98 24.00
1      9.14 21.60
2      4.03 34.70
3      2.94 33.40
5      5.21 28.70
..      ...  ...
499  15.10 17.50
500  14.33 16.80
502   9.08 20.60
503   5.64 23.90
504   6.48 22.00

```

[394 rows x 14 columns]

```
ds.describe()
```

```

      CRIM      ZN  INDUS  CHAS    NOX     RM   AGE     DIS  RAD
TAX \
count 486.00 486.00 486.00 486.00 506.00 506.00 486.00 506.00 506.00
506.00
mean   3.61  11.21  11.08   0.07   0.55   6.28  68.52   3.80   9.55
408.24
std    8.72  23.39   6.84   0.26   0.12   0.70  28.00   2.11   8.71
168.54
min    0.01   0.00   0.46   0.00   0.39   3.56   2.90   1.13   1.00
187.00
25%    0.08   0.00   5.19   0.00   0.45   5.89  45.17   2.10   4.00
279.00

```

50%	0.25	0.00	9.69	0.00	0.54	6.21	76.80	3.21	5.00
330.00									
75%	3.56	12.50	18.10	0.00	0.62	6.62	93.97	5.19	24.00
666.00									
max	88.98	100.00	27.74	1.00	0.87	8.78	100.00	12.13	24.00
711.00									

	PTRATIO	B	LSTAT	MEDV
count	506.00	506.00	486.00	506.00
mean	18.46	356.67	12.72	22.53
std	2.16	91.29	7.16	9.20
min	12.60	0.32	1.73	5.00
25%	17.40	375.38	7.12	17.02
50%	19.05	391.44	11.43	21.20
75%	20.20	396.23	16.96	25.00
max	22.00	396.90	37.97	50.00

уникальные значения целевого признака
ds['MEDV'].unique()

```
array([24. , 21.6, 34.7, 33.4, 36.2, 28.7, 22.9, 27.1, 16.5, 18.9, 15.
,
      21.7, 20.4, 18.2, 19.9, 23.1, 17.5, 20.2, 13.6, 19.6, 15.2,
14.5,
      15.6, 13.9, 16.6, 14.8, 18.4, 21. , 12.7, 13.2, 13.1, 13.5, 20.
,
      24.7, 30.8, 34.9, 26.6, 25.3, 21.2, 19.3, 14.4, 19.4, 19.7,
20.5,
      25. , 23.4, 35.4, 31.6, 23.3, 18.7, 16. , 22.2, 33. , 23.5, 22.
,
      17.4, 20.9, 24.2, 22.8, 24.1, 21.4, 20.8, 20.3, 28. , 23.9,
24.8,
      22.5, 23.6, 22.6, 20.6, 28.4, 38.7, 43.8, 33.2, 27.5, 26.5,
18.6,
      20.1, 19.5, 19.8, 18.8, 18.5, 18.3, 19.2, 17.3, 15.7, 16.2, 18.
,
      14.3, 23. , 18.1, 17.1, 13.3, 17.8, 14. , 13.4, 11.8, 13.8,
14.6,
      15.4, 21.5, 15.3, 17. , 41.3, 24.3, 27. , 50. , 22.7, 23.8,
22.3,
      19.1, 29.4, 23.2, 24.6, 29.9, 37.2, 39.8, 37.9, 32.5, 26.4,
29.6,
      32. , 29.8, 37. , 30.5, 36.4, 31.1, 29.1, 33.3, 30.3, 34.6,
32.9,
      42.3, 48.5, 24.4, 22.4, 28.1, 23.7, 26.7, 30.1, 44.8, 37.6,
46.7,
      31.5, 31.7, 41.7, 48.3, 29. , 25.1, 17.6, 24.5, 26.2, 42.8,
21.9,
      44. , 36. , 33.8, 43.1, 48.8, 31. , 36.5, 30.7, 43.5, 20.7,
21.1,
```

```

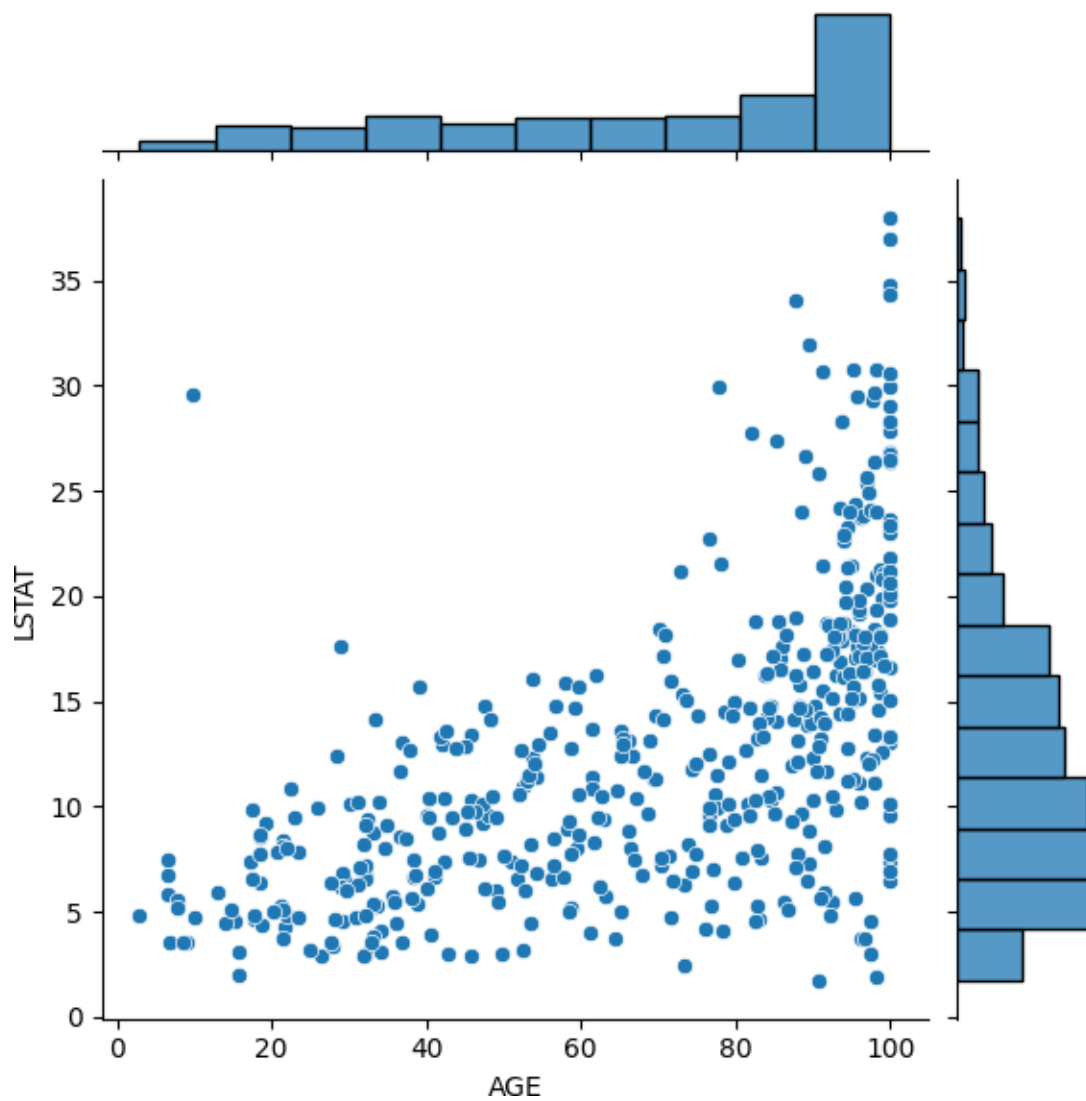
25.2, 35.2, 32.4, 33.1, 35.1, 45.4, 46. , 32.2, 28.5, 37.3,
27.9, 28.6, 36.1, 28.2, 16.1, 22.1, 19. , 32.7, 31.2, 17.2, 16.8,
10.2, 10.4, 10.9, 11.3, 12.3, 8.8, 7.2, 10.5, 7.4, 11.5, 15.1,
9.7, 12.5, 8.5, 5. , 6.3, 5.6, 12.1, 8.3, 11.9, 17.9, 16.3, 7.
',
7.5, 8.4, 16.7, 14.2, 11.7, 11. , 9.5, 14.1, 9.6, 8.7,
12.8, 10.8, 14.9, 12.6, 13. , 16.4, 17.7, 12. , 21.8, 8.1])

```

Jointplot

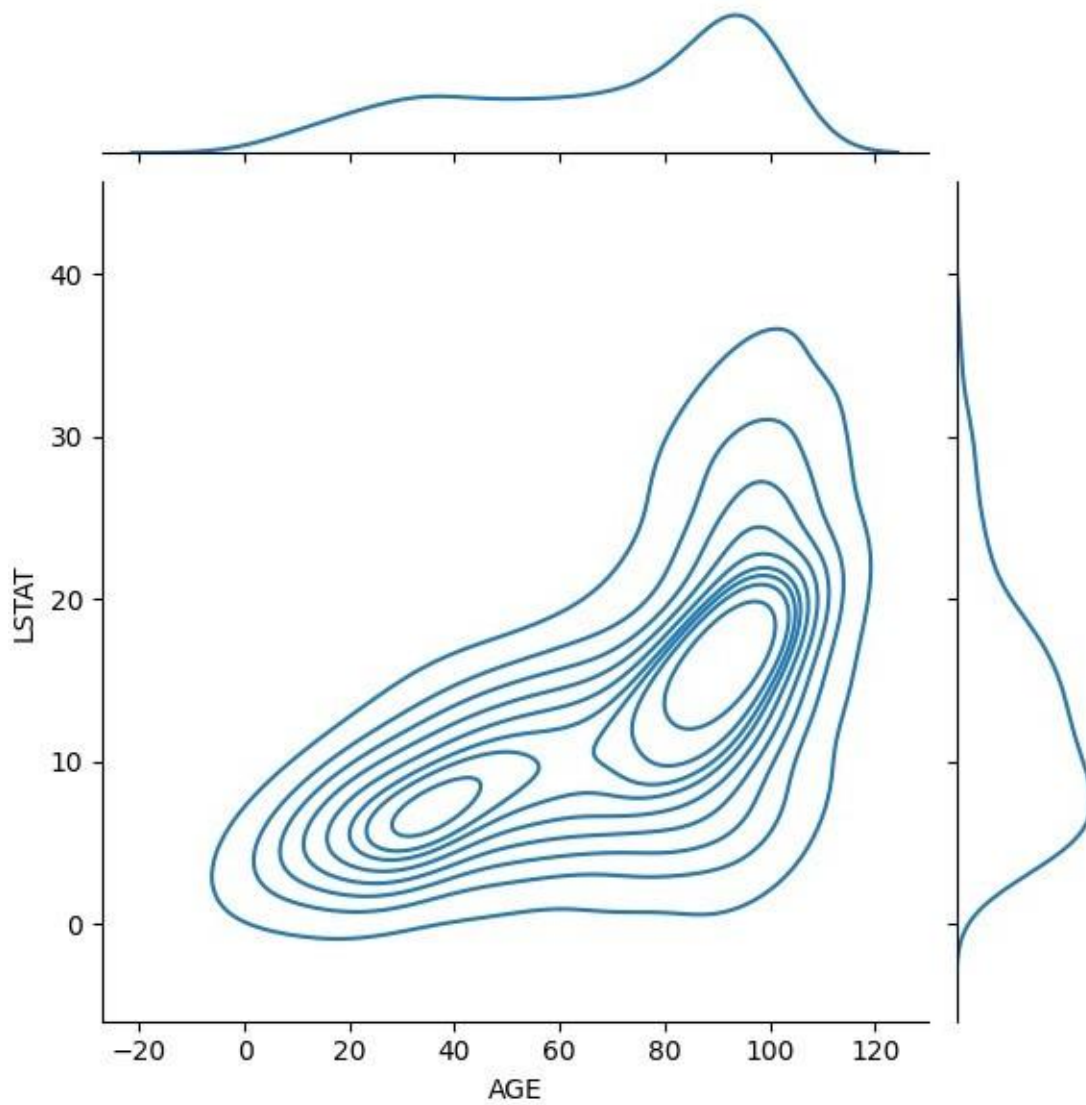
```
sns.jointplot(x='AGE', y='LSTAT', data=ds)
```

```
<seaborn.axisgrid.JointGrid at 0x7f0a20de32e0>
```

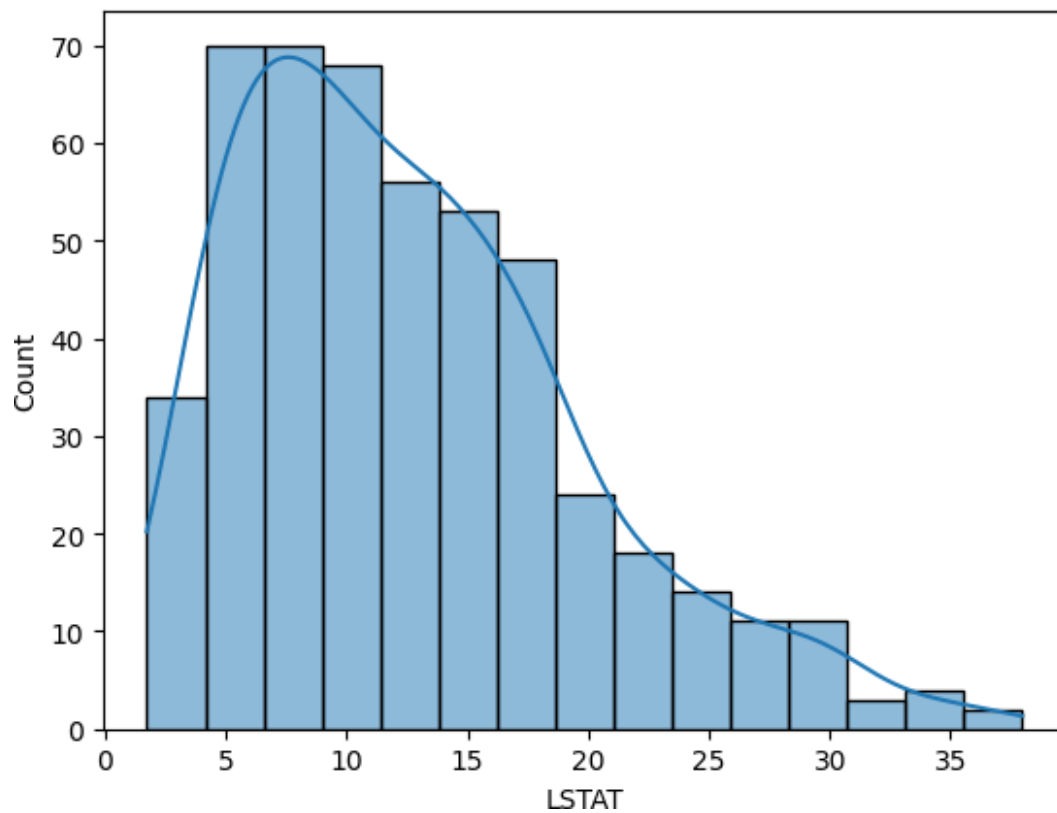


```
sns.jointplot(x='AGE', y='LSTAT', data=ds, kind="kde")
```

```
<seaborn.axisgrid.JointGrid at 0x7f0a2070c760>
```

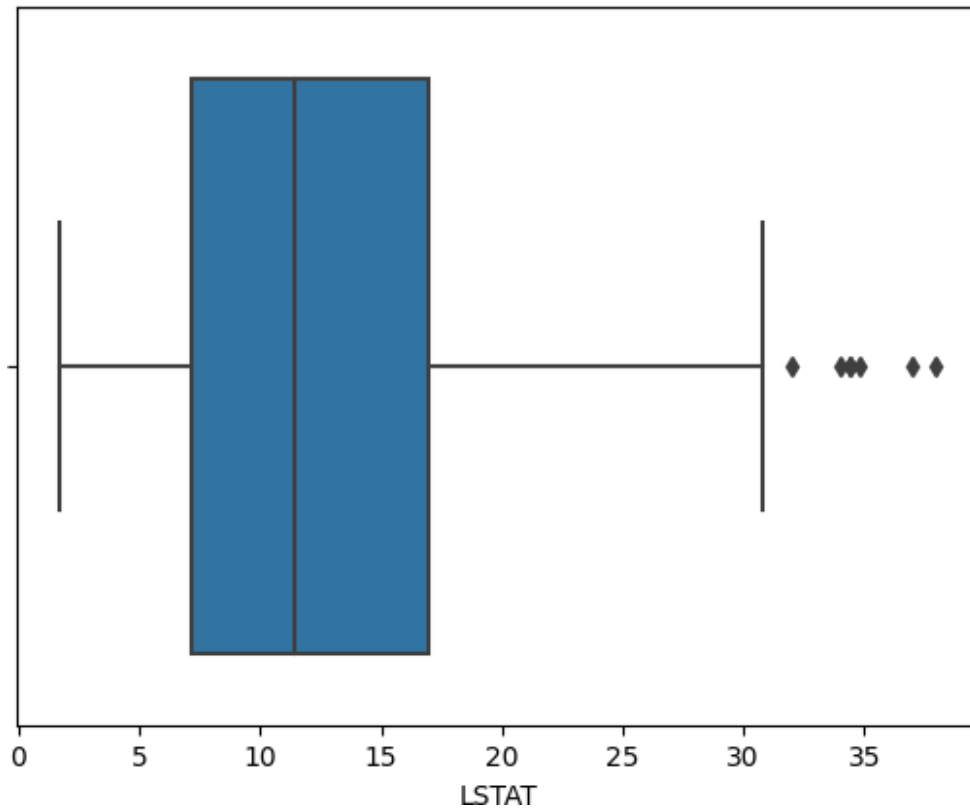


```
sns.histplot(ds['LSTAT'], kde=True)
<Axes: xlabel='LSTAT', ylabel='Count'>
```

```
sns.boxplot(x=ds['LSTAT'])
```

```
<Axes: xlabel='LSTAT'>
```



```
sns.jointplot(x='AGE', y='LSTAT', data=ds, kind="hex")  
<seaborn.axisgrid.JointGrid at 0x7f0a1dd19930>
```


RM	-0.22	0.32	-0.39	0.10	-0.30	1.00	-0.25	0.21	-0.21	-0.29
-0.36										
AGE	0.35	-0.56	0.64	0.08	0.73	-0.25	1.00	-0.74	0.46	0.51
0.27										
DIS	-0.37	0.66	-0.71	-0.09	-0.77	0.21	-0.74	1.00	-0.49	-0.53
-0.23										
RAD	0.62	-0.31	0.60	0.00	0.61	-0.21	0.46	-0.49	1.00	0.91
0.46										
TAX	0.58	-0.31	0.73	-0.03	0.67	-0.29	0.51	-0.53	0.91	1.00
0.46										
PTRATIO	0.28	-0.41	0.39	-0.11	0.19	-0.36	0.27	-0.23	0.46	0.46
1.00										
B	-0.38	0.17	-0.36	0.05	-0.38	0.13	-0.28	0.29	-0.44	-0.44
-0.18										
LSTAT	0.44	-0.41	0.59	-0.05	0.58	-0.61	0.60	-0.49	0.48	0.54
0.38										
MEDV	-0.39	0.37	-0.48	0.18	-0.43	0.70	-0.39	0.25	-0.38	-0.47
-0.51										

	B	LSTAT	MEDV
CRIM	-0.38	0.44	-0.39
ZN	0.17	-0.41	0.37
INDUS	-0.36	0.59	-0.48
CHAS	0.05	-0.05	0.18
NOX	-0.38	0.58	-0.43
RM	0.13	-0.61	0.70
AGE	-0.28	0.60	-0.39
DIS	0.29	-0.49	0.25
RAD	-0.44	0.48	-0.38
TAX	-0.44	0.54	-0.47
PTRATIO	-0.18	0.38	-0.51
B	1.00	-0.37	0.33
LSTAT	-0.37	1.00	-0.74
MEDV	0.33	-0.74	1.00

На основе корреляционной матрицы можно сделать следующие выводы:

1. Целевой признак наиболее сильно коррелирует с признаками RM(0,70), ZN(0,37). Эти признаки должны остаться в модели.
2. Целевой признак наименее коррелирует с признаком LSTAT(-0,74) (доля низкообеспеченного населения), его можно убрать из модели.
3. Многие признаки (TAX, PTRATIO, INDUS) плохо коррелируют с целевым, поэтому их тоже можно убрать из модели, так как они почти не будут оказывать влияния.