

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

Курс «Технологии машинного обучения»

Отчёт по рубежному контролю №1

«Технологии разведочного анализа и обработки данных.»

Вариант № 16

Выполнил:
Пронченко Т.А.
группа ИУ5-62Б

Проверил:
Гапанюк Ю.Е.

Дата: 14.04.25

Дата:

Подпись:

Подпись:

Москва, 2025 г.

Полученное задание

Номер варианта: 16

Номер задачи: 2

1. Номер набора данных, указанного в задаче: 8

<https://www.kaggle.com/datasets/altavish/boston-housing-dataset>

Задача №2.

Для заданного набора данных проведите обработку пропусков в данных для одного категориального и одного количественного признака. Какие способы обработки пропусков в данных для категориальных и количественных признаков Вы использовали? Какие признаки Вы будете использовать для дальнейшего построения моделей машинного обучения и почему?

Ход выполнения

Импортируем библиотеки

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

[1] ✓ 2.0s

Python

Импортируем данные

- **CRIM** – уровень преступности на душу населения по городам.
- **ZN** – доля жилых зон, отведённых под участки площадью более 25 000 кв. футов.
- **INDUS** – доля земель коммерческого назначения (не розничная торговля) в городе.
- **CHAS** – фиктивная переменная реки Чарльз (= 1, если участок прилегает к реке; 0 в противном случае).
- **NOX** – концентрация оксидов азота (частей на 10 миллионов).
- **RM** – среднее количество комнат в жилом помещении.
- **AGE** – доля домов, построенных до 1940 года и занятых владельцами.
- **DIS** – средневзвешенное расстояние до пяти рабочих центров Бостона.
- **RAD** – индекс доступности радиальных магистралей.
- **TAX** – ставка налога на недвижимость (полная стоимость на \$10 000).
- **PTRATIO** – соотношение учеников к учителям по городам.
- **B** – расчётный показатель: $1000(Bk - 0.63)^2$, где **Bk** – доля чернокожего населения в городе.
- **LSTAT** – процент населения с низким социально-экономическим статусом.
- **MEDV** – медианная стоимость домов, занимаемых владельцами (в тысячах долларов).

```
df = pd.read_csv('HousingData.csv')
```

[12] ✓ 0.0s

Python

Основные характеристики датасета

```
df.head()
```

[13] ✓ 0.0s Python

...

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	NaN	36.2

```
# Размер датасета - 506 строк, 14 столбцов
df.shape
```

[14] ✓ 0.0s Python

...

(506, 14)

В качестве категориального признака выберем бинарный признак «CHAS» - переменная реки Чарльз, указывающая прилегает ли участок к реке. В качестве количественного признака выберем признак «LSTAT» - процент населения с низким социально-экономическим статусом.

Проверим признаки на пропуски

```
print("Пропуски до обработки:")
print(df[['CHAS', 'LSTAT']].isnull().sum())
```

[21] ✓ 0.0s Python

...

Пропуски до обработки:

CHAS	20
LSTAT	20

dtype: int64

Заполним пропуски

```
# Заполнение пропусков в CHAS (категориальный)
if df['CHAS'].isnull().sum() > 0:
    df.fillna({'CHAS': df['CHAS'].mode()[0]}, inplace=True)
    print("Пропуски в CHAS заполнены модой.")
else:
    print("Пропусков в CHAS нет.")

# Заполнение пропусков в LSTAT (количественный)
if df['LSTAT'].isnull().sum() > 0:
    df.fillna({'LSTAT': df['LSTAT'].median()}, inplace=True)
    print("Пропуски в LSTAT заполнены медианой.")
else:
    print("Пропусков в LSTAT нет.")
```

[23] ✓ 0.0s Python

...

Пропуски в CHAS заполнены модой.
Пропуски в LSTAT заполнены медианой.

```
# Проверка после обработки
print("\nПропуски после обработки:")
print(df[['CHAS', 'LSTAT']].isnull().sum())
```

[24] ✓ 0.0s Python

...

Пропуски после обработки:

CHAS	0
LSTAT	0

dtype: int64

Какие способы обработки пропусков в данных для категориальных и количественных признаков вы использовали?

Какие признаки Вы будете использовать для дальнейшего построения моделей машинного обучения и почему?

1) Признак CHAS является бинарной переменной, принимающей значения 0 или 1, где 1 обозначает расположение недвижимости вблизи реки Чарльз, а 0 - отсутствие такого соседства. Для обработки пропущенных значений в данном признаке могут быть применены следующие методы:

1. Замена на моду (наиболее частое значение)

Данный подход целесообразен при небольшом количестве пропусков и значительном преобладании одного из значений.

2. Создание новой категории "Unknown"

При существенном количестве пропусков рекомендуется введение дополнительной категории.

3. Предсказание пропущенных значений

При наличии значительного числа пропусков возможно применение простых моделей для прогнозирования отсутствующих значений на основе других признаков.

Т.к. в нашем случае пропуски составили менее 5% и значение 0 преобладает замена на моду будет самым оптимальным способом.

2) Признак LSTAT отражает процент населения с низким социально-экономическим статусом. Методы обработки пропусков:

1. Замена на медиану

Медиана является устойчивой к выбросам мерой центральной тенденции

2. Замена на среднее значение

При нормальном распределении данных и отсутствии выбросов допустимо использование среднего

3. Замена на константу

В некоторых случаях применяется заполнение фиксированным значением (0 или -1), однако это может исказить данные.

4. Использование KNN Imputer

Для большого числа пропусков эффективен алгоритм k-ближайших соседей:

Т.к. в нашем случае пропусков немного, и можно заметить выбросы на диаграмме, наиболее оптимальным способом будет замена на медиану.

3) Выбор признаков для построения модели машинного обучения

Для датасета Boston Housing (CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B, LSTAT, MEDV) отбор наиболее информативных признаков осуществлялся на основе корреляции с целевой переменной (MEDV - стоимость жилья).

Для нахождения наиболее важных признаков для построения модели посмотрим корреляцию с целевой переменной

```
# Строим корреляционную матрицу
corr_matrix = df.corr()

print("\nКорреляция всех признаков с MEDV:")
print(corr_matrix['MEDV'].sort_values(ascending=False)[1:])
```

```
[39] ✓ 0.0s Python
...
Корреляция всех признаков с MEDV:
RM      0.695360
ZN      0.373136
B       0.333461
DIS     0.249929
CHAS    0.183844
RAD     -0.381626
CRIM    -0.391363
AGE     -0.394656
NOX     -0.427321
TAX     -0.468536
INDUS   -0.481772
PTRATIO -0.507787
LSTAT   -0.723093
Name: MEDV, dtype: float64
```

Наиболее значимые признаки:

- **LSTAT** (% населения с низким статусом): обратная зависимость от стоимости
- **RM** (среднее число комнат): сильная положительная корреляция с ценой
- **PTRATIO** (соотношение учеников и учителей): влияет на привлекательность района
- **INDUS** (доля нежилых площадей): обратная зависимость от стоимости - чем больше промышленных зон, тем ниже цены на жилье
- **TAX** (налог на имущество): высокая налоговая нагрузка снижает стоимость недвижимости

Несмотря на то, что некоторые признаки выделяются на фоне других, в нашем случае для построения моделей машинного лучше использовать все признаки, т.к. каждый из них все равно оказывает значительное влияние на стоимость участка.