Московский государственный технический университет им. Н. Э. Баумана

Курс «Технологии машинного обучения»

Отчёт по рубежному контролю №1

«Технологии разведочного анализа и обработки данных.»

Вариант № 18

Выполнил:
Швецов Д. Д.
группа ИУ5-63Б

Проверил:
Гапанюк Ю.Е.

Дата: 04.04.25

Дата:

Подпись:

Подпись:

Москва, 2025 г.

Задание:

Номер варианта: **18**

Номер задачи: **3**

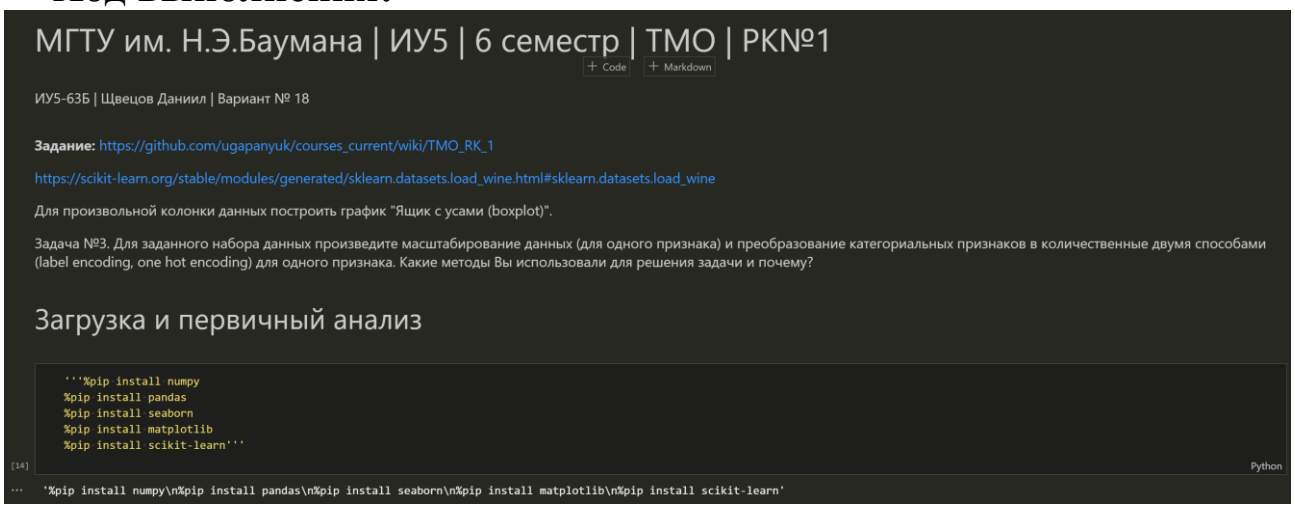
Номер набора данных, указанного в задаче: **2** (https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_wine.html#sklearn.datasets.load_wine)

Для студентов групп ИУ5-63Б, ИУ5Ц-83Б - для произвольной колонки данных построить график "Ящик с усами (boxplot)".

Задача №3.

Для заданного набора данных произведите масштабирование данных (для одного признака) и преобразование категориальных признаков в количественные двумя способами (label encoding, one hot encoding) для одного признака. Какие методы Вы использовали для решения задачи и почему?

Ход выполнения:



```python
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.datasets import load_wine
%matplotlib inline
sns.set_theme(style="whitegrid", palette="pastel")
```

[0] ✓ 2.1s Python

```python
wine = load_wine()
df0 = pd.DataFrame(wine.data, columns=wine.feature_names)
df0["class"] = wine.target
df0.info()
df0.head()
```

[1] ✓ 0.0s Python

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 178 entries, 0 to 177
Data columns (total 14 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   alcohol                       178 non-null    float64
 1   malic_acid                    178 non-null    float64
 2   ash                           178 non-null    float64
 3   alcalinity_of_ash             178 non-null    float64
 4   magnesium                     178 non-null    float64
 5   total_phenols                 178 non-null    float64
 6   flavanoids                    178 non-null    float64
 7   nonflavanoid_phenols          178 non-null    float64
 8   proanthocyanins               178 non-null    float64
 9   color_intensity               178 non-null    float64
 10  hue                           178 non-null    float64
 11  od280/od315_of_diluted_wines  178 non-null    float64
 12  proline                       178 non-null    float64
 13  class                         178 non-null    int64  
dtypes: float64(13), int64(1)
memory usage: 19.6 KB
```

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline	class
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0	0
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050.0	0
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185.0	0
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480.0	0
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735.0	0

"Ящик с усами" (boxplot)

Построим график "ящик с усами" (boxplot) для колонки "alcohol" с разбивкой по колонке "class"

```python
df0.boxplot(column='alcohol', by='class', grid=False)
```

[2] ✓ 0.1s Python

```
<Axes: title={'center': 'alcohol'}, xlabel='class'>
```

Масштабирование данных

Произведём масштабирование данных колонки "alcohol" при помощи двух методов: MinMax масштабирование (MinMaxScaler) и Масштабирование данных на основе Z-оценки (StandardScaler).

MinMax масштабирование:

$$x_{\text{новый}} = \frac{x_{\text{старый}} - min(X)}{max(X) - min(X)}$$

В этом случае значения лежат в диапазоне от 0 до 1.

Масштабирование данных на основе Z-оценки:

$$x_{\text{новый}} = \frac{x_{\text{старый}} - AVG(X)}{\sigma(X)}$$

В этом случае большинство значений попадает в диапазон от -2 до 2.

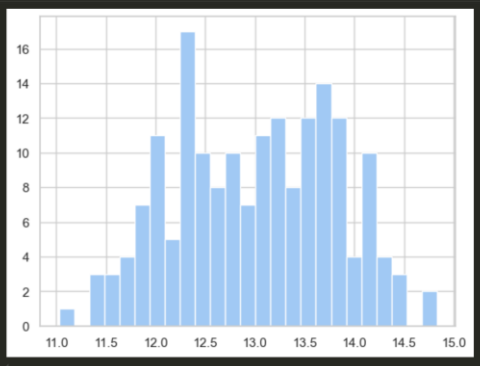
Стандартизированная оценка (z-оценка) - это мера относительного разброса наблюдаемого или измеренного значения, которая показывает, сколько стандартных отклонений составляет его разброс относительного среднего значения.

```
[3] ✓ 0.0s
from sklearn.preprocessing import MinMaxScaler, StandardScaler
```
Python

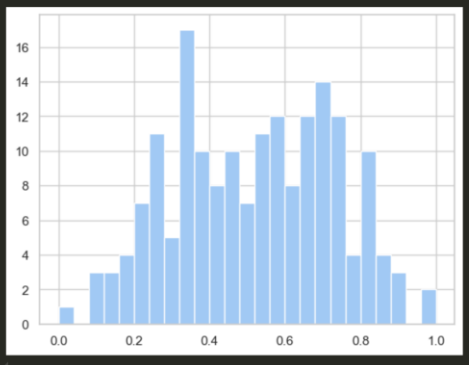
```
[4] ✓ 0.0s
sc1 = MinMaxScaler()
df0_sc1 = sc1.fit_transform(df0[['alcohol']])

sc2 = StandardScaler()
df0_sc2 = sc2.fit_transform(df0[['alcohol']])
```
Python

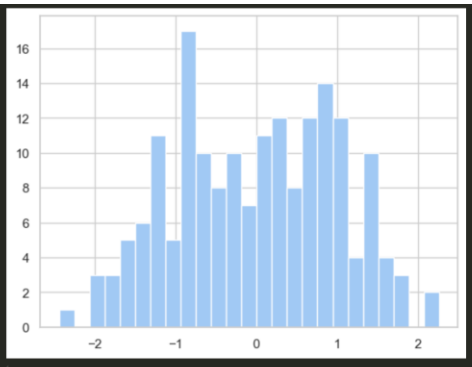
```
[5] ✓ 0.0s
plt.hist(df0['alcohol'], 25)
plt.show()
```
Python




```
[6] ✓ 0.0s
plt.hist(df0_sc1, 25)
plt.show()
```
Python




```
[7] ✓ 0.0s
plt.hist(df0_sc2, 25)
plt.show()
```
Python



Преобразование категориальных признаков в количественные

Преобразуем категориальный признак "Class" в количественный с использованием двух методов: "Label encoding" и "One hot encoding"

Label encoding

Ориентирован на применение к одному признаку. Предназначен для кодирования целевого признака, но может быть также использован для последовательного кодирования отдельных нецелевых признаков.

Сопоставляет значению категориального признака целое неотрицательное число

```
from sklearn.preprocessing import LabelEncoder
```

[8] ✓ 0.0s Python

Так как колонка "Class" в данный момент уже закодирована, раскодируем ее

```
mapping = {0: 'A', 1: 'B', 2: 'C'}

df0['class_decoded'] = df0['class'].apply(lambda x: mapping[x])
df0.head()
```

[9] ✓ 0.0s Python

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline	class	class_decoded
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0	0	A
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050.0	0	A
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185.0	0	A
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480.0	0	A
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735.0	0	A

```
LE = LabelEncoder()


df0['class_le'] = LE.fit_transform(df0['class_decoded'])
df0.head()
```

[10] ✓ 0.0s Python

	ol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline	class	class_decoded	class_le
23		1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0	0	A	0
20		1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050.0	0	A	0
16		2.36	2.67	18.6	101.0	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185.0	0	A	0
37		1.95	2.50	16.8	113.0	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480.0	0	A	0
24		2.59	2.87	21.0	118.0	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735.0	0	A	0

Уникализируем строки по значению колонки "class_decoded", чтобы наглядко продемонстрировать кодирование для всех возможных значений данного поля

```
unique_rows_LE = df0.drop_duplicates(subset='class_le')
unique_rows_LE
```

[11] ✓ 0.0s Python

	ol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline	class	class_decoded	class_le
23		1.71	2.43	15.6	127.0	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0	0	A	0
37		0.94	1.36	10.6	88.0	1.98	0.57	0.28	0.42	1.95	1.05	1.82	520.0	1	B	1
36		1.35	2.32	18.0	122.0	1.51	1.25	0.21	0.94	4.10	0.76	1.29	630.0	2	C	2

One hot encoding

Каждое уникальное значение признака становится новым отдельным признаком(столбцом 1-да, 0-нет).

```
from sklearn.preprocessing import OneHotEncoder
```

[12] ✓ 0.0s Python

```
OHE = OneHotEncoder()

encoded_classes = OHE.fit_transform(df0[['class_decoded']])

# Create a DataFrame with the encoded classes
df0[OHE.get_feature_names_out(['class_decoded'])] = encoded_classes.todense()
```

[13] ✓ 0.0s Python

```
df0.head()
```

[14] ✓ 0.0s Python

total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline	class	class_decoded	class_le	class_decoded_A	class_decoded_B	class_decoded_C
2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065.0	0	A	0	1.0	0.0	0.0
2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050.0	0	A	0	1.0	0.0	0.0
2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185.0	0	A	0	1.0	0.0	0.0
3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480.0	0	A	0	1.0	0.0	0.0
2.80	2.69	0.39	1.82	4.32	1.04	2.93	735.0	0	A	0	1.0	0.0	0.0