Московский государственный технический университет им. Н. Э. Баумана

Курс «Технологии машинного обучения»

Отчёт по лабораторной работе №2

Выполнил:
Сайфутдинов Р.И.
группа ИУ5-62Б

Проверил:
Гапанюк Ю.Е.

Дата: 09.04.25

Дата:

Подпись:

Подпись:

Москва, 2025 г.

Цель лабораторной работы: изучение способов предварительной обработки данных для дальнейшего формирования моделей.

Задание:

1. Выбрать набор данных (датасет), содержащий категориальные признаки и пропуски в данных. Для выполнения следующих пунктов можно использовать несколько различных наборов данных (один для обработки пропусков, другой для категориальных признаков и т.д.)
2. Для выбранного датасета (датасетов) на основе материалов лекции решить следующие задачи:
    a. обработку пропусков в данных;
    b. кодирование категориальных признаков;
    c. масштабирование данных.

Ход выполнения:

```
[35]
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
```

```
[49]
path = 'student_admission_record_dirty.csv'
df = pd.read_csv(path)
```

```
[50]
df.isna().sum()
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 157 entries, 0 to 156
Data columns (total 7 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Name                    147 non-null    object
 1   Age                     147 non-null    float64
 2   Gender                  147 non-null    object
 3   Admission Test Score    146 non-null    float64
 4   High School Percentage  146 non-null    float64
 5   City                    147 non-null    object
 6   Admission Status        147 non-null    object
dtypes: float64(3), object(4)
memory usage: 8.7+ KB
```

[51]
```
df.dropna(subset=['Gender', 'City'], inplace=True)
df['Age'].fillna(df['Age'].mean(), inplace=True)
df['High School Percentage'].fillna(df['High School Percentage'].mean(), inplace=True)
df['Admission Test Score'].fillna(df['Admission Test Score'].mean(), inplace=True)
```

[52]
```
df.isnull().sum()
```

```
Name                     8
Age                      0
Gender                   0
Admission Test Score     0
High School Percentage   0
City                     0
Admission Status         6
dtype: int64
```

[53]
```
df.drop(columns='Name', inplace=True)
```

[54]
```
df.head()
```

	Age	Gender	Admission Test Score	High School Percentage	City	Admission Status
0	24.0	Female	50.0	68.90	Quetta	Rejected
1	21.0	Female	99.0	60.73	Karachi	NaN
2	17.0	Male	89.0	76.30	Islamabad	Accepted
3	17.0	Male	55.0	85.29	Karachi	Rejected
4	20.0	Male	65.0	61.13	Lahore	NaN

```
numerical_features = ['Age', 'Admission Test Score', 'High School Percentage']
scaler = StandardScaler()
df[numerical_features] = scaler.fit_transform(df[numerical_features])
```

```
df.head()
```

	Age	Gender	Admission Test Score	High School Percentage	City	Admission Status
0	1.009103	Female	-1.722736	-4.357777e-01	Quetta	Rejected
1	0.297768	Female	1.440320	-9.168999e-01	Karachi	NaN
2	-0.650678	Male	0.794798	-1.673723e-15	Islamabad	Accepted
3	-0.650678	Male	-1.399975	5.294110e-01	Karachi	Rejected
4	0.060656	Male	-0.754453	-8.933443e-01	Lahore	NaN

```
categorical_features = df[['Gender', 'City']]
categorical_data_encoded = pd.get_dummies(categorical_features, drop_first=True)
```

```
categorical_data_encoded.head()
```

	Gender_Male	City_Karachi	City_Lahore	City_Multan	City_Peshawar	City_Quetta	City_Rawalpindi
0	False	False	False	False	False	True	False
1	False	True	False	False	False	False	False
2	True	False	False	False	False	False	False
3	True	True	False	False	False	False	False
4	True	False	True	False	False	False	False

```
numerical_data = ['Age', 'Admission Test Score', 'High School Percentage']
categorical_data = ['Gender', 'City']
categorical_data_encoded = pd.get_dummies(df[categorical_data], drop_first=True)
encoded_data = categorical_data_encoded
numerical_data_encoded = scaler.fit_transform(df[numerical_data])
encoded_data[numerical_data] = numerical_data_encoded
encoded_data
```

	Gender_Male	City_Karachi	City_Lahore	City_Multan	City_Peshawar	City_Quetta	City_Rawalpindi	Age	Admission Test Score	High School Percentage
0	False	False	False	False	False	True	False	1.009103	-1.722736	-4.357777e-01
1	False	True	False	False	False	False	False	0.297768	1.440320	-9.168999e-01
2	True	False	False	False	False	False	False	-0.650678	0.794798	-9.142658e-16
3	True	True	False	False	False	False	False	-0.650678	-1.399975	5.294110e-01
4	True	False	True	False	False	False	False	0.060656	-0.754453	-8.933443e-01
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
152	False	False	False	False	False	True	False	-0.176455	0.536590	1.054111e-01
153	False	False	False	False	False	False	False	-0.650678	0.278381	4.769999e-01
154	False	False	False	True	False	False	False	0.297768	1.375768	-1.498133e+00
155	True	False	False	False	False	True	False	-4.918688	0.923903	2.249555e-01
156	True	False	True	False	False	False	False	-0.650678	0.730246	6.212777e-01

138 rows × 10 columns

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
le.fit_transform(df["Gender"])
```

[55]

```
array([0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1,
       1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0,
       0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0,
       1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0,
       0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0,
       1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1,
       0, 0, 0, 0, 1, 1])
```