Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №2
по дисциплине
«Методы машинного обучения»
на тему

«Обработка признаков часть 1»

Выполнил:
студент группы ИУ5И-23М
Лю Цзычжан

Москва-2025 г.

Цель лабораторной работы:

изучение продвинутых способов предварительной обработки данных для дальнейшего формирования моделей.

Задание:

1. Выбрать набор данных (датасет), содержащий категориальные и числовые 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.) Просьба не использовать датасет, на котором данная задача решалась в лекции.
2. Для выбранного датасета (датасетов) на основе материалов лекций решить следующие задачи:
   i. устранение пропусков в данных;
   ii. кодирование категориальных признаков;
   iii. нормализация числовых признаков.

```
!pip install pandas scikit-learn
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.0.3)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.1)
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.10/dist-packages (from pandas) (1.25.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

```
[25] import pandas as pd
     import numpy as np
     from sklearn.impute import SimpleImputer
     from sklearn.preprocessing import StandardScaler, OneHotEncoder
     from sklearn.compose import ColumnTransformer
     from sklearn.pipeline import Pipeline
```

```python
# 读取数据集
df = pd.read_csv('E:\BMSTU\一下\jqxx\heart.csv')
```

✓ 0.0s

```python
# 查看数据集基本信息
print("Основная информация о наборе данных: ")
print(df.info())
```

✓ 0.0s

```
Основная информация о наборе данных:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1025 non-null   int64
 1   sex       1025 non-null   int64
 2   cp        1025 non-null   int64
 3   trestbps  1025 non-null   int64
 4   chol      1025 non-null   int64
 5   fbs       1025 non-null   int64
 6   restecg   1025 non-null   int64
 7   thalach   1025 non-null   int64
 8   exang     1025 non-null   int64
 9   oldpeak   1025 non-null   float64
 10  slope     1025 non-null   int64
 11  ca        1025 non-null   int64
 12  thal      1025 non-null   int64
 13  target    1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
None
```

```python
# 1. Устранение пропусков в данных (null表示缺失值)
# 将值为null的部分替换为np.nan
df[['age','sex','cp','trestbps','chol','fbs','restecg','thalach','exang','oldpeak','slope','ca','thal','target']] = df[['age','sex','cp','trestbps','chol','fbs','restecg','thalach','exang','oldpeak','slope','ca','thal','target']].replace('null', np.nan)
```

✓ 0.0s

```python
# 使用SimpleImputer来处理缺失值
imputer = SimpleImputer(strategy='mean')
df[['age','sex','cp','trestbps','chol','fbs','restecg','thalach','exang','oldpeak','slope','ca','thal','target']] = imputer.fit_transform(df[['age','sex','cp','trestbps','chol','fbs','restecg','thalach','exang','oldpeak','slope','ca','thal','target']])
```

0.0s

```python
# 输出处理后的数据集
print("Обработанный набор данных: ")
print(df.head())
```

✓ 0.0s

```
Обработанный набор данных:
        age       sex        cp  trestbps      chol       fbs   restecg  \
0 -0.268437  0.661504 -0.915755 -0.377636 -0.659332 -0.418878  0.891255
1 -0.158157  0.661504 -0.915755  0.479107 -0.833861  2.387330 -1.004049
2  1.716595  0.661504 -0.915755  0.764688 -1.396233 -0.418878  0.891255
3  0.724079  0.661504 -0.915755  0.936037 -0.833861 -0.418878  0.891255
4  0.834359 -1.511706 -0.915755  0.364875  0.930822  2.387330  0.891255

    thalach     exang   oldpeak     slope        ca      thal    target
0  0.821321 -0.712287 -0.060888  0.995433  1.209221  1.089852 -1.026698
1  0.255968  1.403928  1.727137 -2.243675 -0.731971  1.089852 -1.026698
2 -1.048692  1.403928  1.301417 -2.243675 -0.731971  1.089852 -1.026698
3  0.516900 -0.712287 -0.912329  0.995433  0.238625  1.089852 -1.026698
4 -1.874977 -0.712287  0.705408 -0.624121  2.179817 -0.522122 -1.026698
```