Министерство образования Республики Беларусь
Учреждение образования
«Брестский Государственный технический университет»
Кафедра ИИТ

Лабораторная работа №1
По дисциплине: «Основы машинного обучения»
Тема: «Знакомство с анализом данных: предварительная обработка и визуализация»

Выполнила:
Студентка 3 курса
Группы АС-66
Прокурат В. Д.
Проверил:
Крощенко А. А.

Брест 2025

Цель работы: получить практические навыки работы с данными с использованием библиотек Pandas для манипуляции и Matplotlib для визуализации. Научиться выполнять основные шаги предварительной обработки данных, такие как очистка, нормализация и работа с различными типами признаков.

Вариант 9

Выборка Melbourne Housing Market. Содержит данные о продажах домов в Мельбурне, включая цену, количество комнат, район и т.д.

Задачи:

1. Загрузите данные. Найдите столбец с наибольшим количеством пропущенных значений и удалите его.
2. Удалите все строки, где отсутствует значение цены (Price).
3. Постройте гистограмму распределения цен на недвижимость.
4. Рассчитайте среднюю цену за дом для 5 самых популярных пригородов (Suburb).
5. Создайте новый признак PropertyAge на основе года постройки (YearBuilt).
6. Преобразуйте признак Type (тип недвижимости) в числовой формат с помощью One-Hot Encoding.

Код программы:

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

# 1.
df = pd.read_csv('Melbourne_housing.csv', na_values=['missing', 'inf'],)

missing_counts = df.isna().sum()
print("Количество пропусков по столбцам:")
print(missing_counts.sort_values(ascending=False).head(5))

most_missing = missing_counts.idxmax()
print(f"\nУдаляем столбец с наибольшим количеством пропусков: {most_missing}
({missing_counts[most_missing]} пропусков)")
df.drop(columns=[most_missing], inplace=True)

print(f"Оставшиеся столбцы: {list(df.columns)}")



# 2
before_rows = len(df)
df.dropna(subset=['Price'], inplace=True)
```

```
after_rows = len(df)

print(f"\nУдалено строк без цены: {before_rows - after_rows}")
print(f"Оставшиеся строки: {after_rows}")



# 3
plt.figure(figsize=(10, 6))
plt.hist(df['Price'], bins=40, color='skyblue', edgecolor='black')
plt.title('Распределение цен на недвижимость в Мельбурне')
plt.xlabel('Цена (AUD)')
plt.ylabel('Количество объектов')
plt.grid(True)
plt.tight_layout()
plt.show()



# 4
top_suburbs = df['Suburb'].value_counts().nlargest(5).index
avg_prices = df[df['Suburb'].isin(top_suburbs)].groupby('Suburb')['Price'].mean()
print("\nСредняя цена по 5 самым популярным пригородам:")
print(avg_prices)



# 5
current_year = pd.Timestamp.now().year
df['PropertyAge'] = current_year - pd.to_numeric(df['YearBuilt'])

print("\nШапка DataFrame с PropertyAge:")
print(df[['Suburb', 'YearBuilt', 'PropertyAge']].head())



# 6
df = pd.get_dummies(df, columns=['Type'], prefix='Type')

print("\nНовые столбцы после One-Hot Encoding:")
print([col for col in df.columns if col.startswith('Type_')])

print("\nШапка итогового DataFrame:")
print(df.head())
```

Результат работы программы:

#1

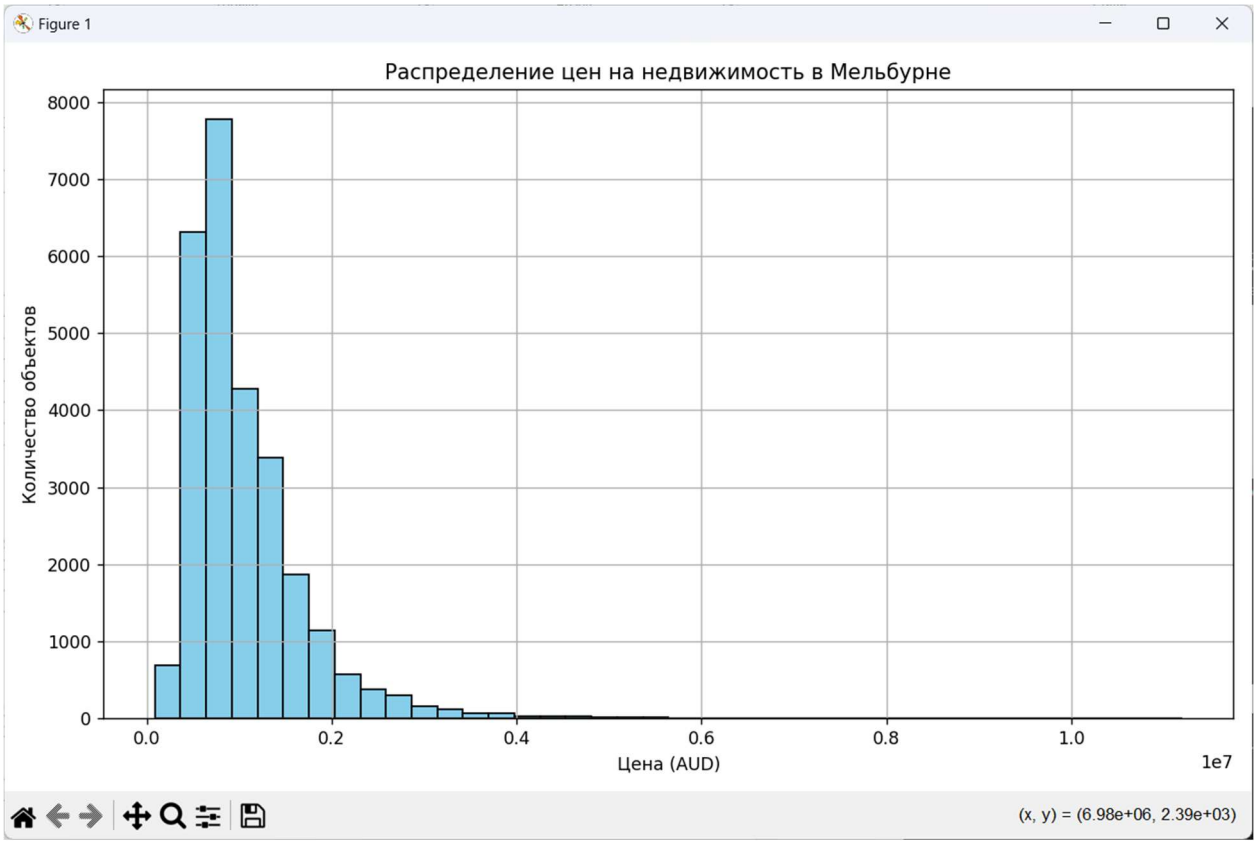
```
Количество пропусков по столбцам:
BuildingArea    21115
YearBuilt       19306
Landsize        11810
Car              8728
Bathroom         8226
dtype: int64

Удаляем столбец с наибольшим количеством пропусков: BuildingArea (21115 пропусков)
Оставшиеся столбцы: ['Suburb', 'Address', 'Rooms', 'Type', 'Method', 'SellerG', 'Date', 'Distance'
uncilArea', 'Latitude', 'Longtitude', 'Regionname', 'Propertycount', 'ParkingArea', 'Price']
```

#2

```
Удалено строк без цены: 7610
Оставшиеся строки: 27247
```

#3



#4

```
Средняя цена по 5 самым популярным пригородам:
Suburb
Bentleigh East    1.131418e+06
Brunswick         9.779888e+05
Preston           8.778699e+05
Reservoir         6.911045e+05
Richmond          1.067585e+06
Name: Price, dtype: float64
```

#5

```
Шапка DataFrame с PropertyAge:
          Suburb  YearBuilt  PropertyAge
1   Airport West     2016.0          9.0
2    Albert Park     1900.0        125.0
3    Albert Park        NaN          NaN
5     Alphington     1930.0         95.0
6     Alphington     2013.0         12.0
```

#6

```
Новые столбцы после One-Hot Encoding:
['Type_h', 'Type_t', 'Type_u']
```

```
Шапка итогового DataFrame:
      Suburb          Address  Rooms Method       SellerG     Date  Distance  ...  Propertycount     ParkingArea     Price  PropertyAge  Type_h  Type_t  Type_u
1  Airport West     154 Halsey Rd      3     PI        Nelson  3/9/2016      13.5  ...         3464.0  Detached Garage  840000.0          9.0   False    True   False
2   Albert Park     105 Kerferd Rd      2      S  hockingstuart  3/9/2016       3.3  ...         3280.0  Attached Garage 1275000.0        125.0    True   False   False
3   Albert Park  85 Richardson St      2      S       Thomson  3/9/2016       3.3  ...         3280.0           Indoor 1455000.0          NaN    True   False   False
5    Alphington       6 Smith St      4      S         Brace  3/9/2016       6.4  ...         2211.0      Underground 2000000.0         95.0    True   False   False
6    Alphington     5/6 Yarralea St      3      S        Jellis  3/9/2016       6.4  ...         2211.0     Outdoor Stall 1110000.0         12.0    True   False   False
```

Вывод: получила практические навыки работы с данными с использованием библиотек Pandas для манипуляции и Matplotlib для визуализации. Научилась выполнять основные шаги предварительной обработки данных, такие как очистка, нормализация и работа с различными типами признаков.