

Ćwiczenie: Interaktywna EDA

Stworzona przy użyciu biblioteki Streamlit aplikacja, pozwala przeprowadzić eksploracyjną analizę danych w sposób interaktywny. Dzięki możliwości stworzenia interfejsu użytkownika, dodanie dynamicznych wykresów czy przycisków, zwiększa się przejrzystość – można wyświetlić wybrane wyniki przetwarzania danych, a cały kod pozostaje ukryty. Aplikacja stworzona z pomocą streamlit może zostać zaprezentowana osobom nietechnicznym, które bez konieczności znajomości programowania mogą zobaczyć rezultaty.

Aplikacja wykonana na potrzeby ćwiczenia jest podzielona na sekcje:

1. Podgląd danych.
2. Rozkład zmiennych.
3. Macierz korelacji.
4. Rozrzut zmiennych.

Ad 1. Podgląd danych to sekcja, którą można rozwinąć lub ukryć i zawiera statystyki opisowe, próbkę zbioru danych oraz informacje o nim.

Statystyki zbioru danych

Podgląd danych

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude	target
0	8.3252	41	6.9841	1.0238	322	2.5556	37.88	-122.23	4.526
1	8.3014	21	6.2381	0.9719	2,401	2.1098	37.86	-122.22	3.585
2	7.2574	52	8.2881	1.0734	496	2.8023	37.85	-122.24	3.521
3	5.6431	52	5.8174	1.0731	558	2.5479	37.85	-122.25	3.413
4	3.8462	52	6.2819	1.0811	565	2.1815	37.85	-122.25	3.422
5	4.0368	52	4.7617	1.1036	413	2.1399	37.85	-122.25	2.697
6	3.6591	52	4.9319	0.9514	1,094	2.1284	37.84	-122.25	2.992
7	3.12	52	4.7975	1.0618	1,157	1.7883	37.84	-122.25	2.414
8	2.0804	42	4.2941	1.1176	1,206	2.0269	37.84	-122.26	2.267
9	3.6912	52	4.9706	0.9902	1,551	2.1723	37.84	-122.25	2.611

Typy kolumn

	0
MedInc	float64
HouseAge	float64
AveRooms	float64
AveBedrms	float64
Population	float64
AveOccup	float64
Latitude	float64
Longitude	float64
target	float64

Podstawowe statystyki opisowe

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude	target
count	20,640	20,640	20,640	20,640	20,640	20,640	20,640	20,640	20,640
mean	3.8707	28.6395	5.429	1.0967	1,425.4767	3.0707	35.6319	-119.5697	2.0686

Statystyki zbioru danych

Statystyki są schowane

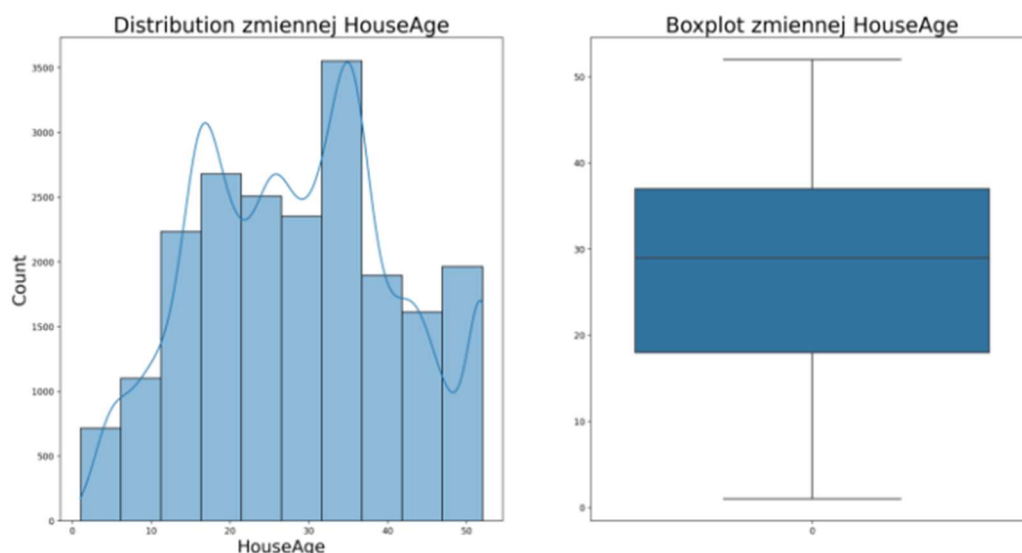
Statystyki

Ad 2. Sekcja rozkład zmiennych zawiera drop-down, na którym można wybrać wyświetlaną zmienną. Wykresy to histogram oraz boxplot.

Rozkład zmiennych

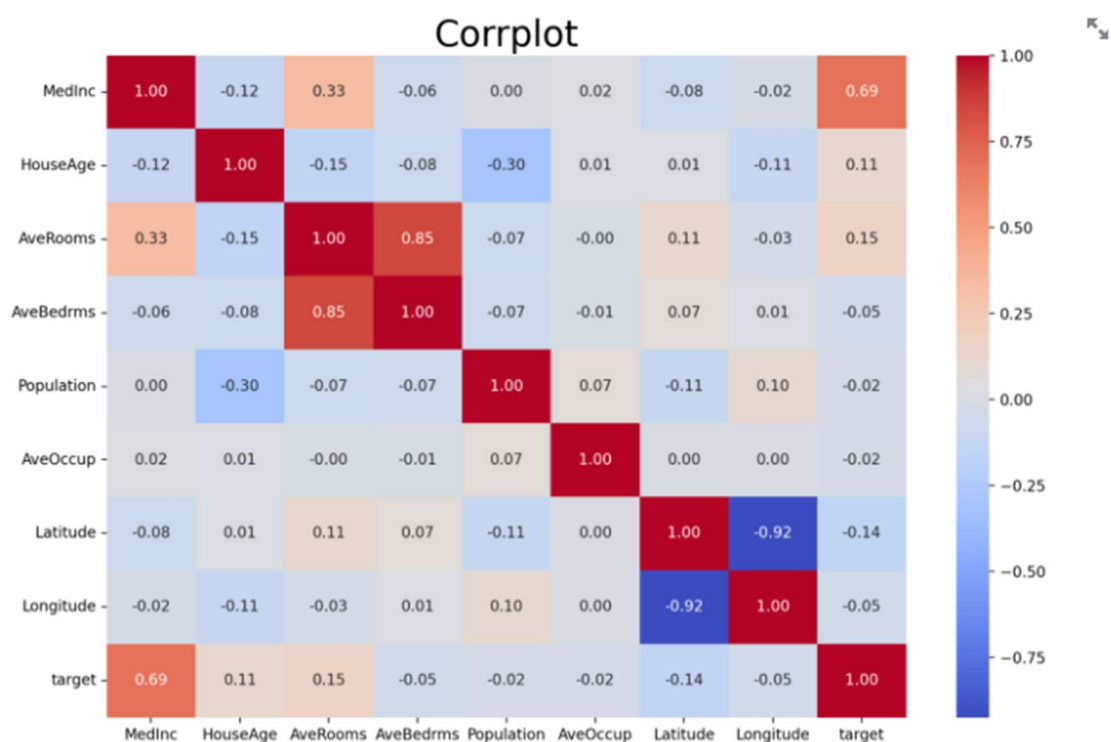
Wybierz zmienną

HouseAge



Ad 3. Sekcja Macierz korelacji to wykres prezentujący jak wygląda wzajemne skorelowanie między zmiennymi.

Macierz korelacji



Ad 4. Sekcja Rozrzut zmiennych zawiera wykres rozrzutu z możliwością wyboru kolumny na osi X oraz Y

Rozrzut zmiennych

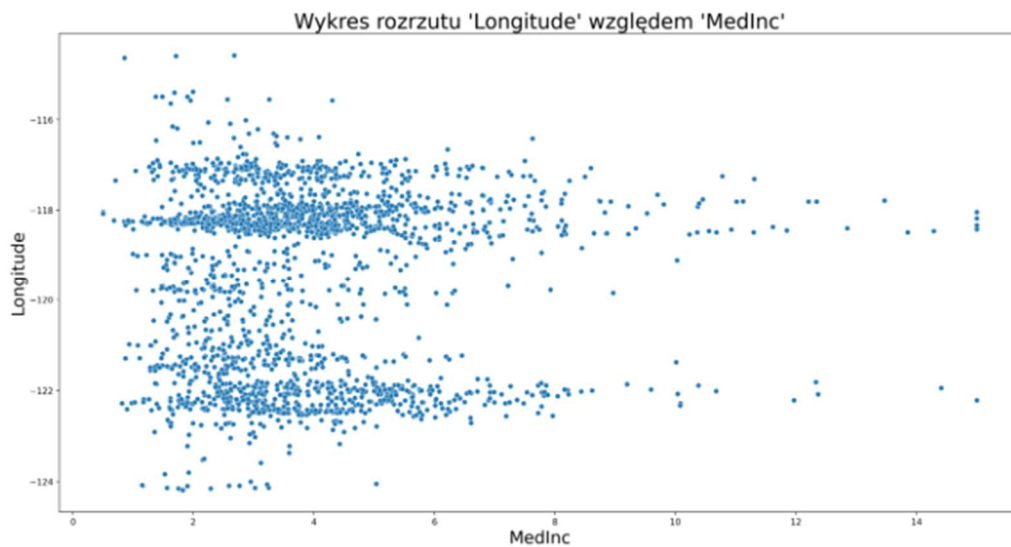
Wybierz oś X

MedInc



Wybierz oś Y

Longitude



Kod który posłużył do wykonania aplikacji:

```
import streamlit as st
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.datasets import load_iris
from sklearn.datasets import fetch_california_housing
import matplotlib as mpl

@st.cache
def load_housing_data():
    housing = fetch_california_housing()
    df = pd.DataFrame(data=housing.data, columns=housing.feature_names)

    df['target'] = housing.target
    return df
```

```

def show_basic_stats(data):
    st.subheader("Podgląd danych")
    st.write(data.head(10))
    st.subheader("Typy kolumn")
    st.write(data.dtypes)
    st.subheader("Podstawowe statystyki opisowe")
    st.write(data.describe())
    st.subheader("Wartości brakujące")
    st.write(data.isnull().sum())

def handle_stats_button(data):
    st.subheader("Statystyki zbioru danych")
    if 'button_clicked' not in st.session_state:
        st.session_state.button_clicked = False

    # Render DataFrame if button is clicked
    if st.session_state.button_clicked:
        show_basic_stats(data)
    else:
        st.markdown("Statystyki są schowane")

    # Button to toggle DataFrame visibility
    if st.button("Statystyki"):
        st.session_state.button_clicked = not st.session_state.button_clicked

# Funkcja do wizualizacji rozkładu zmiennych
def show_distribution_plots(data):
    st.subheader("Rozkład zmiennych")
    mpl.rcParams['axes.labelsize'] = 20
    mpl.rcParams['axes.titlesize'] = 24

    selected_col = st.selectbox("Wybierz zmienną", data.columns)

    if selected_col:
        fig, ax = plt.subplots(1,2,figsize=(20,10))

        col = data[data[selected_col] <= data[selected_col].mean() + 3 *
data[selected_col].std()][selected_col]
        sns.histplot(x=col, bins=10, kde=True, ax=ax[0])
        sns.boxplot(data=col, ax=ax[1])

        ax[0].set_title(f"Distribution zmiennej {selected_col}")
        ax[1].set_title(f"Boxplot zmiennej {selected_col}")

        st.pyplot(fig)

def show_corrplot(data):
    st.subheader("Macierz korelacji")

```

```

correlation_matrix = data.corr()

fig, ax = plt.subplots(figsize=(12,8))
# Create correlation plot using Seaborn
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", ax=ax)
ax.set_title("Corrplot")
st.pyplot(fig)

def show_scatterplot(data):
    st.subheader("Rozrzut zmiennych")
    data_frac = data.sample(frac=0.1, random_state=42)
    selected_col_x = st.selectbox("Wybierz oś X", data_frac.columns)
    selected_col_y = st.selectbox("Wybierz oś Y", data_frac.columns)
    # Plot chart based on selected column
    if selected_col_x and selected_col_y:
        fig, ax = plt.subplots(figsize=(20,10))
        sns.scatterplot(x=data_frac[selected_col_x], y=data_frac[selected_col_y], ax=ax)

        ax.set_title(f"Wykres rozrzutu '{selected_col_y}' względem '{selected_col_x}'")

        st.pyplot(fig)

def main():

    housing_data = load_housing_data()

    handle_stats_button(housing_data)

    show_distribution_plots(housing_data)

    show_corrplot(housing_data)

    show_scatterplot(housing_data)

if __name__ == "__main__":
    main()

```