

SPRAWOZDANIE

Zajęcia: Nauka o danych I

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium Nr 9

Data 02.03.2025

Temat: Wprowadzenie do analizy czasowych szeregów danych, projektowanie eksperymentów i test hipotez.

Wariant 14

Krzysztof Świerczek

Informatyka

II stopień, stacjonarne,
1semestr, gr. A

1. Polecenie: Wprowadzenie do analizy czasowych szeregów danych, projektowanie eksperymentów i test hipotez.

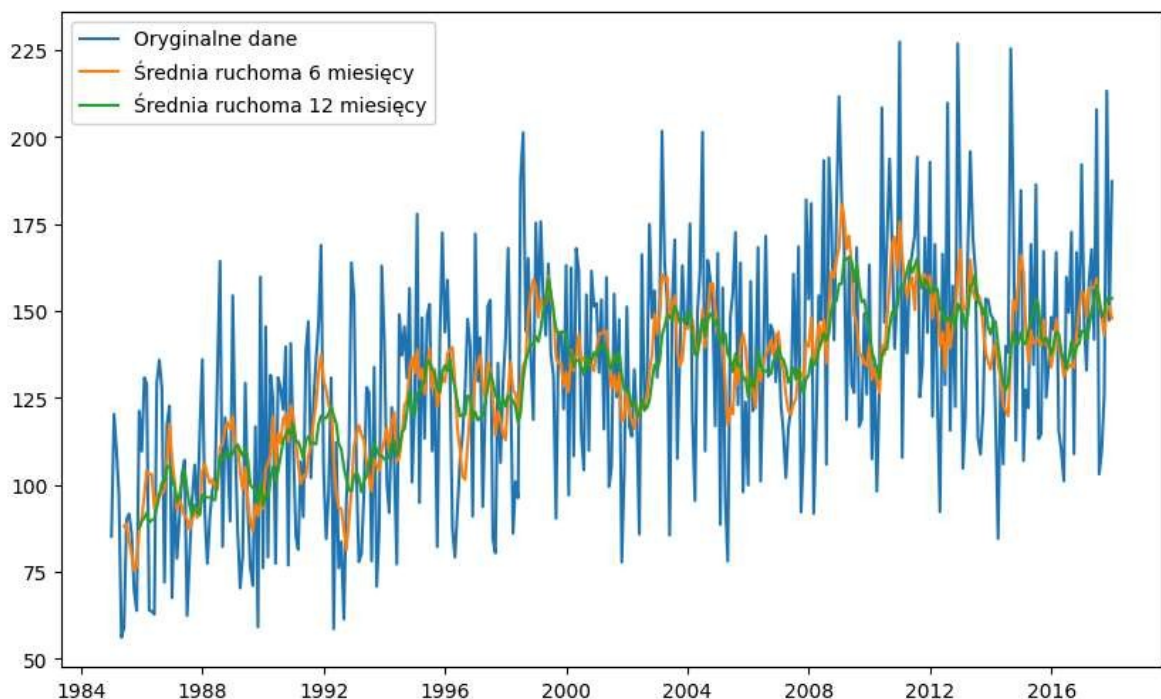
2. Github:

<https://github.com/Krzycho165/STUDIA>

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.seasonal import seasonal_decompose

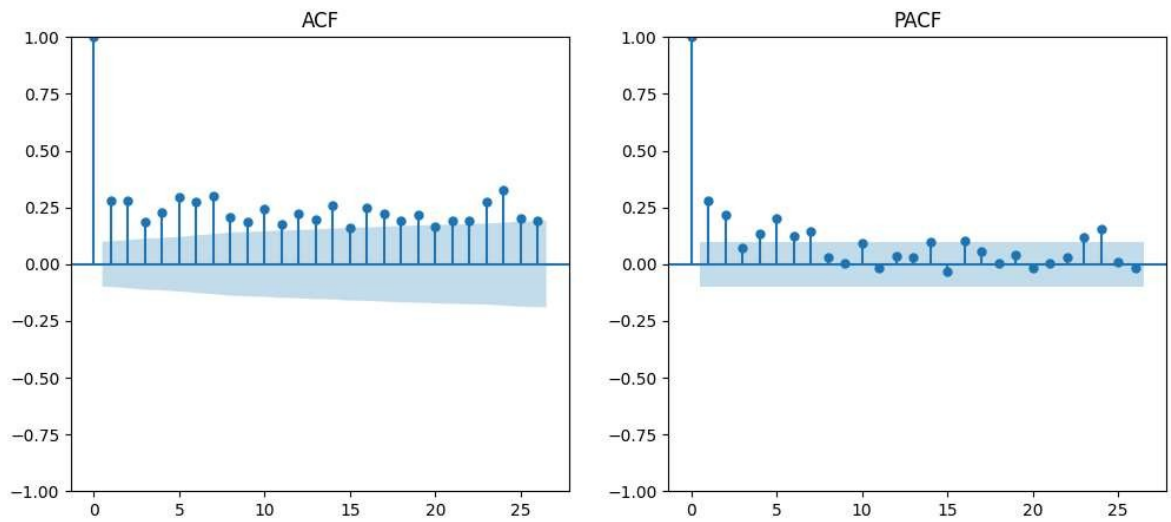
# Wczytanie danych
dane = pd.read_csv('science.csv', parse_dates=['DATE'], index_col='DATE')
dane.columns = ['wartosci']
# 1. Średnie ruchome
dane['srednia_6'] = dane['wartosci'].rolling(window=6).mean()
dane['srednia_12'] = dane['wartosci'].rolling(window=12).mean()

plt.figure(figsize=(10, 6))
plt.plot(dane['wartosci'], label='Oryginalne dane')
plt.plot(dane['srednia_6'], label='Średnia ruchoma 6 miesięcy')
plt.plot(dane['srednia_12'], label='Średnia ruchoma 12 miesięcy')
plt.legend()
plt.show()
```



```
In [2]: # 2. Analiza ACF i PACF
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plot_acf(dane['wartosci'].dropna(), ax=plt.gca())
plt.title('ACF')

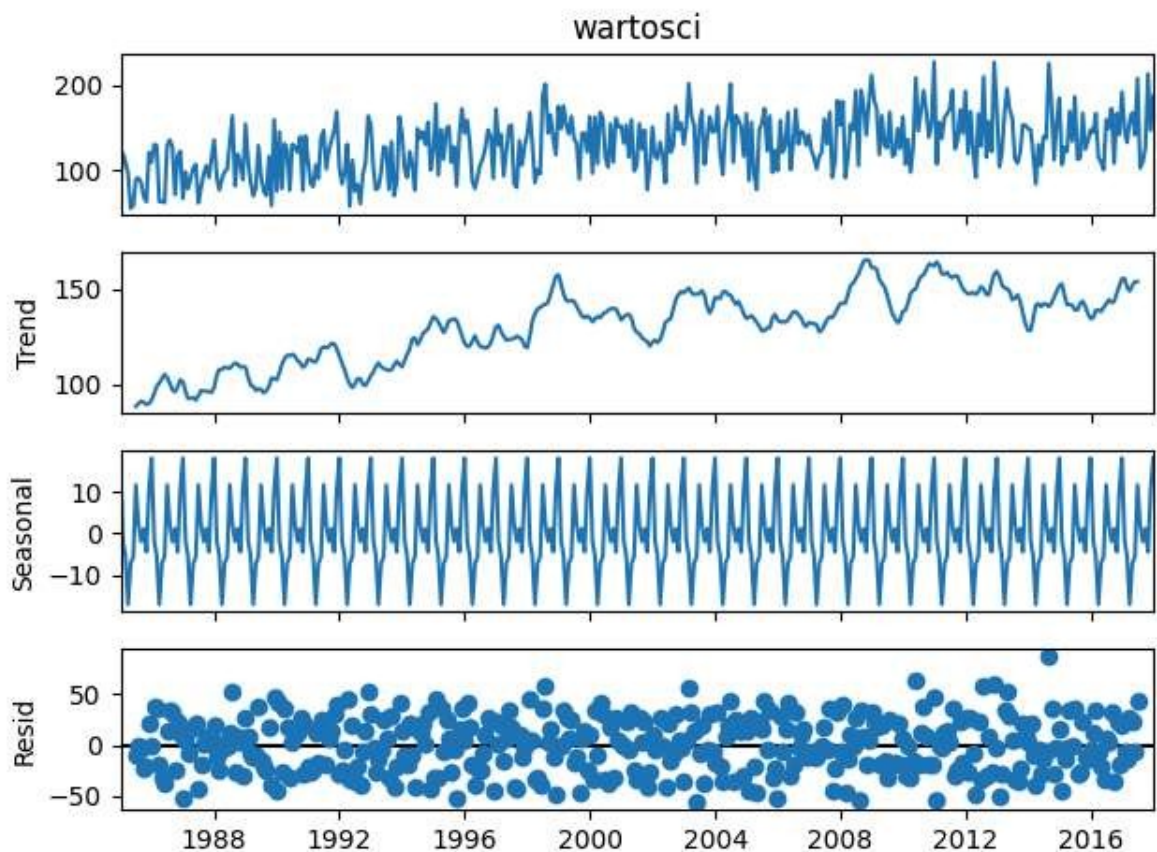
plt.subplot(1, 2, 2)
plot_pacf(dane['wartosci'].dropna(), ax=plt.gca(), method='ywm')
plt.title('PACF')
plt.show()
```



```
In [3]: # 3. Dekompozycja danych
dekompozycja = seasonal_decompose(dane['wartosci'], model='additive', period=12)

plt.figure(figsize=(10, 8))
dekompozycja.plot()
plt.show()
```

<Figure size 1000x800 with 0 Axes>



In []:

3. Wnioski z ćwiczenia

Kod zadania koncentruje się na analizie szeregów czasowych, wykorzystując różne techniki programistyczne do eksploracji i wizualizacji danych. Jego wykonanie pozwala na kilka kluczowych wniosków dotyczących użycia narzędzi programistycznych w analizie danych czasowych.

Pierwszym etapem jest **wczytanie danych z pliku CSV** i skonfigurowanie indeksu jako daty. Dzięki temu dane są odpowiednio sformatowane do analizy szeregów czasowych, co pozwala na łatwiejszą pracę z funkcjami biblioteki **pandas**. Dodatkowo, nazwa kolumny jest modyfikowana w celu ujednolicenia struktury danych. Kod następnie wykorzystuje **średnie ruchome** do wygładzenia danych. Implementacja tego podejścia poprzez metodę **rolling().mean()** w **pandas** pozwala na obserwację długoterminowych trendów i wyeliminowanie krótkoterminowych fluktuacji. Jest to prosty, ale skuteczny sposób na analizę trendów w danych.

Kolejnym krokiem jest **analiza autokorelacji i częściowej autokorelacji (ACF i PACF)**, które są wykorzystywane do oceny zależności między obserwacjami w szeregu czasowym. Kod wykorzystuje do tego bibliotekę **statsmodels**, co pokazuje, jak można efektywnie korzystać z gotowych narzędzi statystycznych do badania struktury czasowej danych.

Dodatkowo, kod zawiera **dekompozycję sezonową**, co pozwala na rozbicie szeregu czasowego na jego składowe: trend, sezonowość i składnik losowy. Wykorzystanie funkcji **seasonal_decompose()** z **statsmodels** jest dobrą praktyką w analizie danych czasowych, umożliwiającą lepsze zrozumienie struktury szeregu.

Podsumowując, wykonane ćwiczenie prezentuje kluczowe techniki stosowane w analizie szeregów czasowych, w tym przekształcanie i przeglądanie danych, wygładzanie średnią ruchomą, analizę autokorelacji oraz dekompozycję sezonową. Wykorzystanie bibliotek takich jak **pandas**, **matplotlib** i **statsmodels** pokazuje, jak zautomatyzować analizę danych czasowych i przygotować je do dalszego modelowania lub prognozowania.