

Міністерство освіти і науки України
НТУУ «КПІ ім. Ігоря Сікорського»
Навчально-науковий інститут атомної та теплової енергетики
Кафедра цифрових технологій в енергетиці

Лабораторна робота №9
з дисципліни «Вступ до інтелектуального аналізу даних»
Тема «Візуалізація часових рядів, частина 2»
Варіант №19

Студента 3-го курсу НН ІАТЕ гр. ТР-12

Ковальова Олександра

Перевірив: д.т.н., проф. Путренко В. В.

Мета: Опрацювати приклад роботи з візуалізацією часових рядів, використовуючи Jupyter Notebook. Виконати поставлене завдання.

Хід роботи

Для початку підключаємо потрібні бібліотеки:

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import pandas as pd
Executed at 2024.04.14 17:51:53 in 740ms
```

У даному фрагменті коду використовується метод згладжування даних для аналізу мінімальних денних температур. Спершу, дані про температури завантажуються з CSV файлу за допомогою функції `np.genfromtxt()`, де параметр `delimiter` визначає символ-роздільник (в даному випадку - кома), а `skip_header` вказує кількість рядків заголовка, які необхідно пропустити.

Після завантаження даних, вони зберігаються у змінній `temps`. Наступною операцією є обчислення середнього значення та стандартного відхилення, що необхідно для подальшого аналізу.

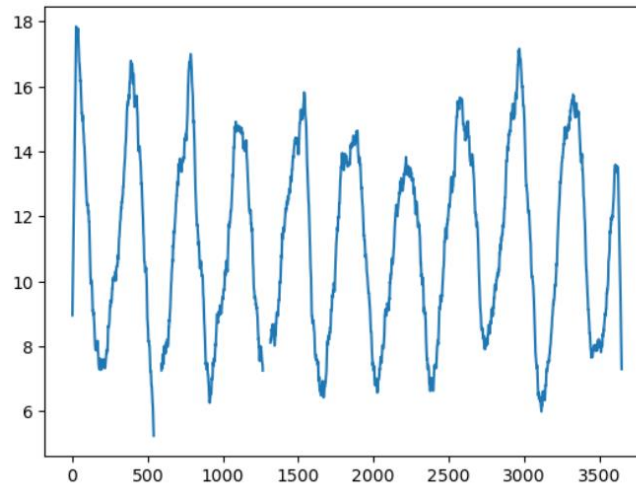
Для згладжування температурних даних використовується функція `np.convolve()`. Ця функція застосовує ваговий коефіцієнт, що задається як одиниця, до даних `temps` за допомогою конволюції. Ваговий коефіцієнт представлений як одиниця у вигляді масиву з розміром `window_size`, що вказує на ширину вікна згладжування. Опція `'same'` гарантує, що вихідний сигнал буде такого ж розміру, як і вхідний.

Отриманий результат підлягає візуалізації за допомогою функції `plt.plot()`, яка будує графік. Згладжені дані відображаються на графіку.

Загалом, цей фрагмент коду дозволяє згладити температурні дані з метою виявлення загальних тенденцій у зміні температури протягом періоду спостережень.

```
1 data = np.genfromtxt('./data/daily-minimum-temperatures-in-me.csv',
2                       delimiter = ",", skip_header = 1)
3 temps = data[:,1]
4
5 mean = np.nanmean(temps)
6 std = np.nanstd(temps)
7
8 window_size = 50
9
10 plt.plot(np.convolve(temps,
11                      np.ones(window_size,)/
12                      window_size, 'same'))
13 plt.show()
Executed at 2024.04.14 17:51:53 in 146ms
```

Результат:



У цьому фрагменті коду використовується метод експоненційного згладжування для аналізу мінімальних денних температур. Спершу, дані про температури завантажуються з CSV файлу за допомогою функції `np.genfromtxt()`, де параметр `delimiter` визначає символ-роздільник (в даному випадку - кома), а `skip_header` вказує кількість рядків заголовка, які необхідно пропустити.

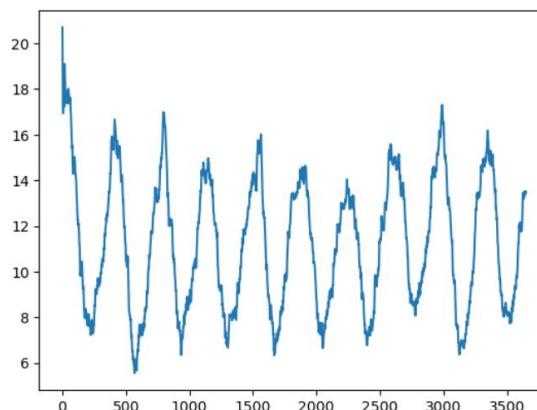
Після завантаження даних, вони зберігаються у змінній `temps`. Наступним кроком є створення об'єкту `DataFrame` з використанням бібліотеки `Pandas`, де колонка `temps` містить температурні дані.

Для експоненційного згладжування використовується метод `ewm()` (експоненційне зважене середнє) бібліотеки `Pandas`. Параметр `span` визначає ширину вікна згладжування, тобто кількість даних, які враховуються при обчисленні експоненційного середнього. Після цього, за допомогою методу `mean()` обчислюється експоненційне середнє для кожного значення вхідних даних.

Отриманий результат представлений на графіку за допомогою функції `plt.plot()`. Він відображає експоненційно згладжені температурні дані, які допомагають виявити загальні тенденції у зміні температури протягом періоду спостережень.

Цей підхід дозволяє аналізувати температурні дані з використанням експоненційного згладжування, що може бути корисним для виявлення трендів та зменшення шуму у даних.

Результат:



Висновок: Під час виконання лабораторної роботи були набуті практичні навички роботи з візуалізацією часових рядів. Було виконане поставлене завдання.

Програмний код

Notebook.ipynb:

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

data = np.genfromtxt('./data/daily-minimum-temperatures-in-me.csv',
                    delimiter = ",", skip_header = 1)
temps = data[:,1]

mean = np.nanmean(temps)
std = np.nanstd(temps)

window_size = 50

plt.plot(np.convolve(temps,
                    np.ones(window_size,)/
                    window_size, 'same'))

plt.show()

data = np.genfromtxt('./data/daily-minimum-temperatures-in-me.csv',
                    delimiter = ",", skip_header = 1)
temps = data[:,1]

span = 50
df = pd.DataFrame({'temps':temps})
df_exp = (df.ewm(span = 50)
        .mean())
plt.plot(df_exp)
plt.show()
```