IONIAN UNIVERSITY
DEPARTMENT OF INFORMATICS

STOCHASTIC DATA ANALYSIS

Labs Report

Markou Dimitra

During the laboratory lessons, we saw the following terms and their application to different data sets.

Mean Value

The mean value, is a measure of central tendency and represents the central point of the data.

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Variance

Variance measures the dispersion of a set of data points around their mean value. High variance indicates that the data points are far from the mean, while low variance indicates that they are close to the mean.

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$$

Standard Deviation

The standard deviation is the square root of the variance and provides a measure of the average distance of each data point from the mean. It is expressed in the same units as the data, making it more interpretable than the variance.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

Covariance

The covariance shows the direction of the linear relationship between two variables. If the covariance is positive, as one variable increases, the other tends to increase as well. If it is negative, one variable tends to decrease as the other increases. The magnitude of the covariance depends on the units of the variables, which makes it difficult to interpret directly.

$$\text{Cov}(X, Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$$

Correlation

Correlation is a standard measure of the strength and direction of the linear relationship between two variables. It ranges from -1 to 1, where 1 indicates a perfect positive linear relationship, -1 indicates a perfect negative linear relationship and 0 indicates a zero linear relationship. Unlike covariance, correlation is dimensionless.

$$\rho_{X,Y} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}$$

Variance Function

A variance function describes how the variance of a variable or process changes over time or in relation to another variable.

$$\gamma(T) = \frac{\sigma_T^2}{\sigma_0^2}$$

Moving Average

The moving average is used to smooth out short-term fluctuations and highlight long-term trends in time series data.

$$MA_k = \frac{1}{k}\sum_{i=0}^{k-1} x_{t-i}$$

Autocovariance

Autocovariance measures the covariance of a time series with a later version of itself. It helps to identify the linear dependence of current values on past values. High autocovariance indicates strong similarity between the time series and its subsequent values.

$$\mathrm{cov}[X_{t_1}, X_{t_2}] = \mathbb{E}[(X_{t_1} - \mu_{t_1})(X_{t_2} - \mu_{t_2})] = \mathbb{E}[X_{t_1} X_{t_2}] - \mu_{t_1}\mu_{t_2}$$

Autocorrelation

Autocorrelation measures the correlation of a time series with its own past and future values. It helps to identify recurring patterns or trends in time series data. Autocorrelation ranges from -1 to 1, similar to correlation.

$$\rho_k = \frac{\gamma_k}{\gamma_0}$$

Convolution

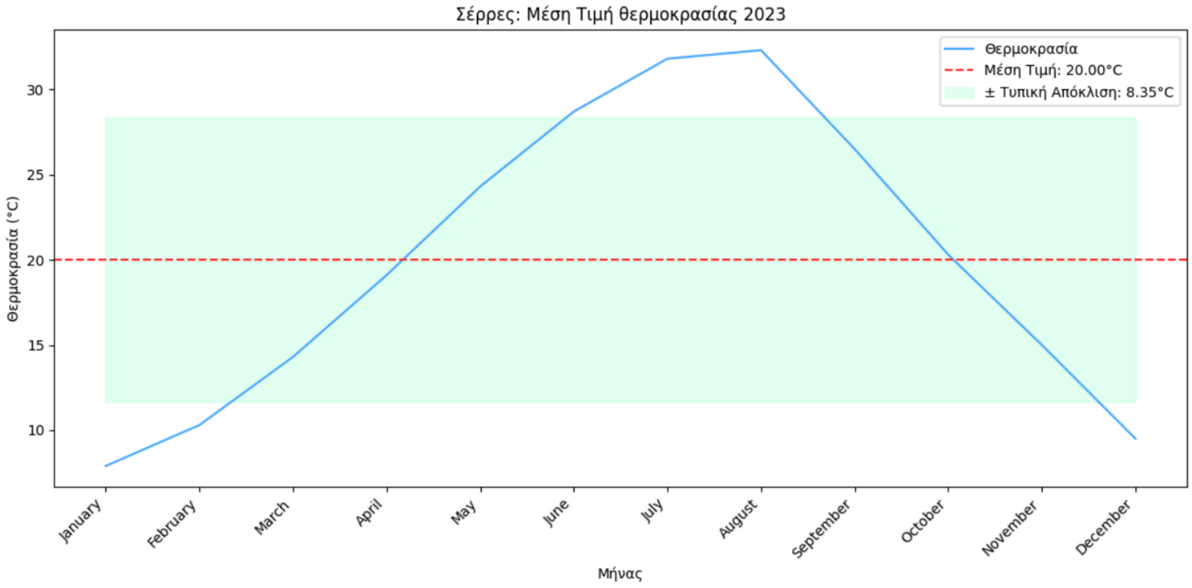
Convolution is used to analyse the effect of one function on another, which is often applied in time series analysis and probability theory. It combines two functions to produce a third function that represents how the form of one is modified by the other.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t - \tau) \, d\tau$$

Applied to a Data Set: serres-temp.csv

For the dataset serres-temp.csv, which contains the average temperature value for each month of the year 2023 for the city of Serres, I calculated the mean value, the variance and the standard deviation.

Mean Value: 20.0
Variance: 69.675
Standard Deviation: 8.34715520402011



The following code, using the Python programming language libraries, numpy, pandas and matplotlib, calculates the mean, variance and standard deviation and creates a graph that reflects them. In the graph, the blue solid line shows the temperatures of the entire dataset, the red dashed line shows the mean, and the faint green box shows the ± standard deviation around the mean.

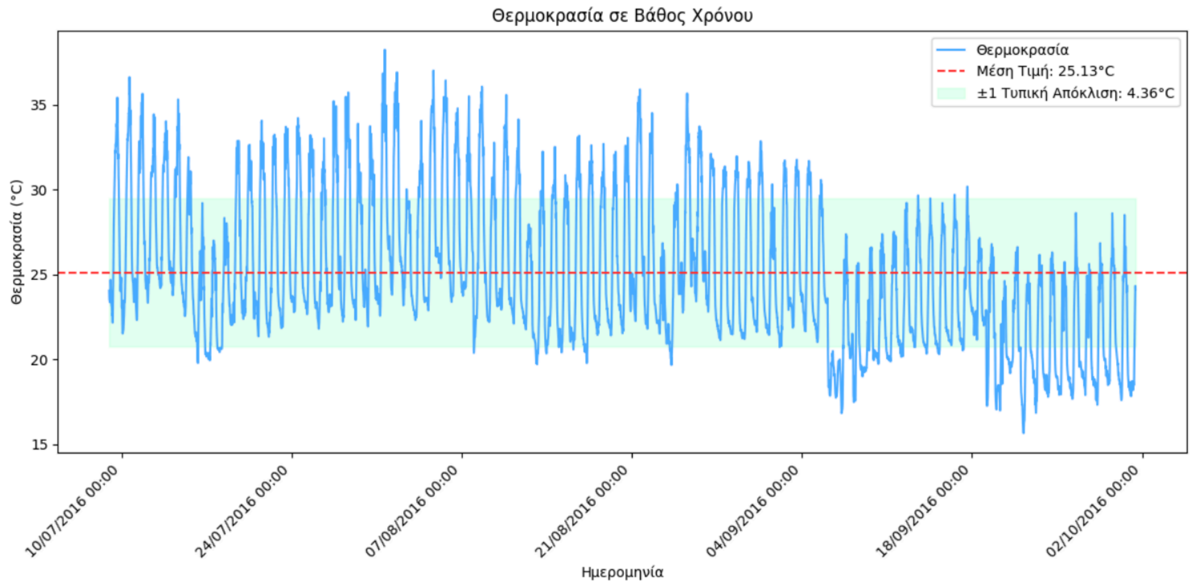
Code

```
# Σύνολο Δεδομένων - serres-temp.csv

file_path = '/kaggle/input/serres-temperatures-2023/serres-temp.csv'
serresTemps = pd.read_csv(file_path)
temperatures = serresTemps['Temperature (°C)']

# Υπολογισμός Μέσης τιμής, Διακύμανσης και Τυπικής απόκλισης
mean_temp = np.mean(temperatures)
variance_temp = np.var(temperatures)
std_dev_temp = np.std(temperatures)

print("Δεδομένα μέσης τιμής θερμοκρασίας για κάθε μήνα του έτους 2023 της πόλης των
Σερρών,")
print("Μέση Τιμή (Mean):", mean_temp)
print("Διακύμανση (Variance):", variance_temp)
print("Τυπική Απόκλιση (Standard Deviation):", std_dev_temp)

# Γράφημα
fig, ax = plt.subplots(figsize=(12, 6))

ax.plot(serresTemps['Month'], temperatures, label='Θερμοκρασία',
color='#4AABFF')
ax.axhline(mean_temp, color='#FF3333', linestyle='--', label=f'Μέση Τιμή:
{mean_temp:.2f}°C')
ax.fill_between(serresTemps['Month'], mean_temp - std_dev_temp, mean_temp +
std_dev_temp, color='#00FF80', alpha=0.1, label=f'± Τυπική Απόκλιση:
{std_dev_temp:.2f}°C')
ax.set_title('Σέρρες: Μέση Τιμή θερμοκρασίας 2023')
ax.set_xlabel('Μήνας')
plt.setp(ax.xaxis.get_majorticklabels(), rotation=45, ha='right')
ax.set_ylabel('Θερμοκρασία (°C)')
ax.legend()

plt.tight_layout()
plt.show()
```

Applied to a Data Set: temp-dataset.csv

For the dataset temp-dataset.csv, which contains data from the area of Agios Markos Corfu, collected over a period of about 3 months, with a sampling frequency of every 15 minutes, I calculated the mean, variance and standard deviation.

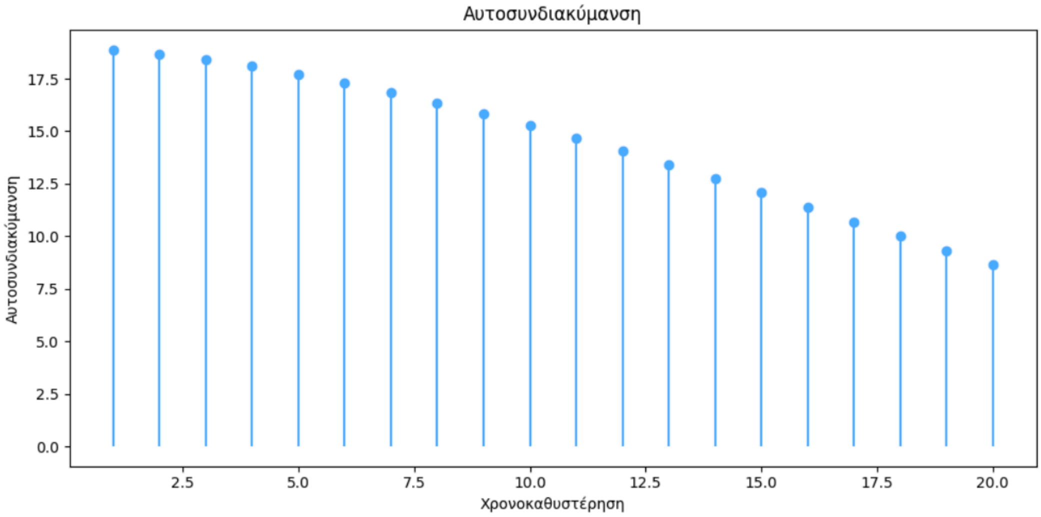
Mean Value: 25.131947342458997
Variance: 18.993359515850425
Standard Deviation: 4.358137161202069



The average temperature during this period was about 25.13°C. The calculated variation of 18.99°C² and the standard deviation of 4.36°C reflect the variability of the temperature data. In the graph the blue solid line shows the temperatures of the entire data set, the red dashed line shows the mean value and the faint green box shows the ± standard deviation around the mean value.

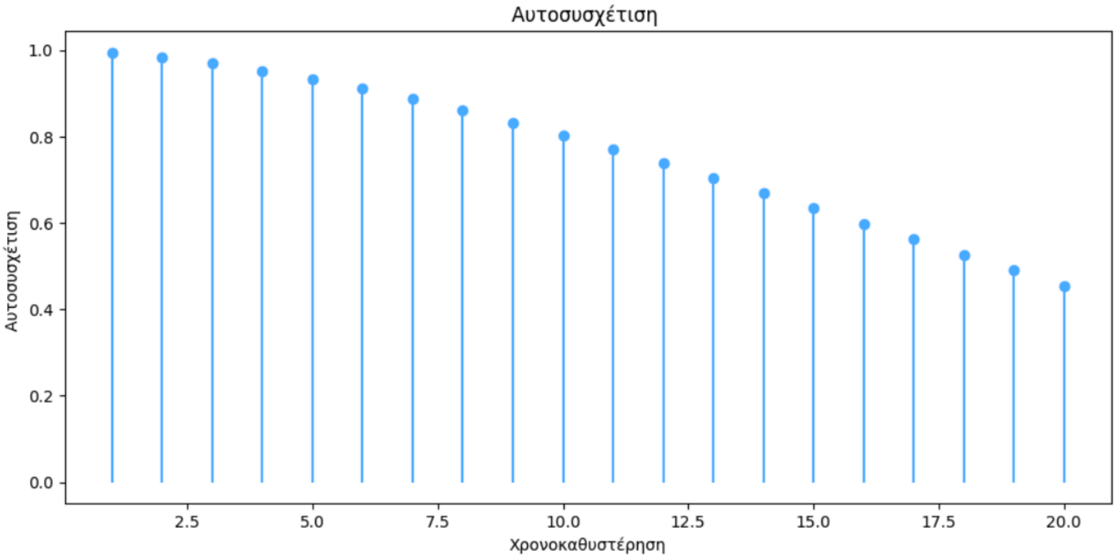
The covariance measures the linear relationship between two different variables. In this case, we have only one variable of interest: temperature. The autocovariance measures the covariance of a time series with a later version of itself. For the analysis of this data set, autocovariance is more important than covariance.

Autocovariance



The autocovariance diagram illustrates the extent to which the temperature values are linearly related to the corresponding time-delayed values at various time intervals, from 1 to 20. High autocovariance values at shorter time lags indicate strong time dependence in the temperature data, with periodic spikes indicating a seasonal or recurring pattern.

Autocorrelation



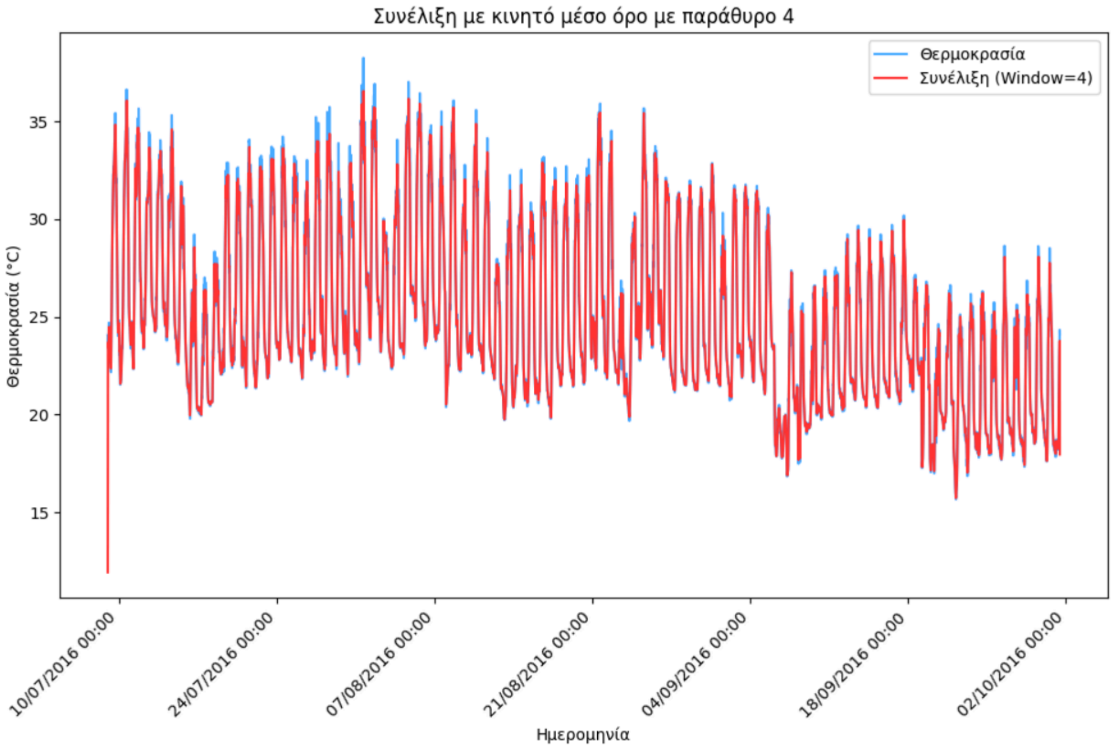
The autocorrelation plot illustrates the correlation between temperature values and their posterior versions, providing a normalized measure of the linear relationship. Time scales range from 1 to 20 and autocorrelation values range between -1 and 1. Values close to 1 indicate a strong positive correlation, while values close to -1 indicate a strong negative correlation. Our graph reveals that the temperature data have high autocorrelation at small lags, confirming a strong temporal relationship.

Convolution With Moving Average

To calculate the moving average, I have chosen 3 different window sizes. The dataset we have is sampled every 15 minutes, so window 4 denotes an hour, window 96 a day and 672 a week.

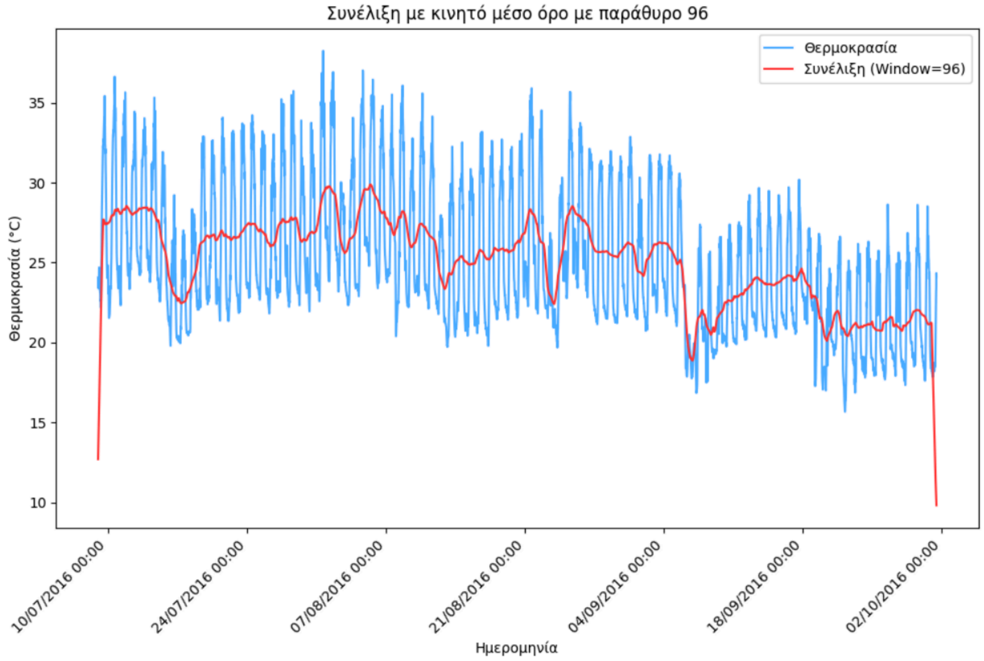
The convolution plots of the temperature data with a moving average kernel of different window sizes: 4, 96 and 672, illustrate the smoothing effect of the different window sizes on the temperature data.

Convolution with a window size of 4 (1 hour)



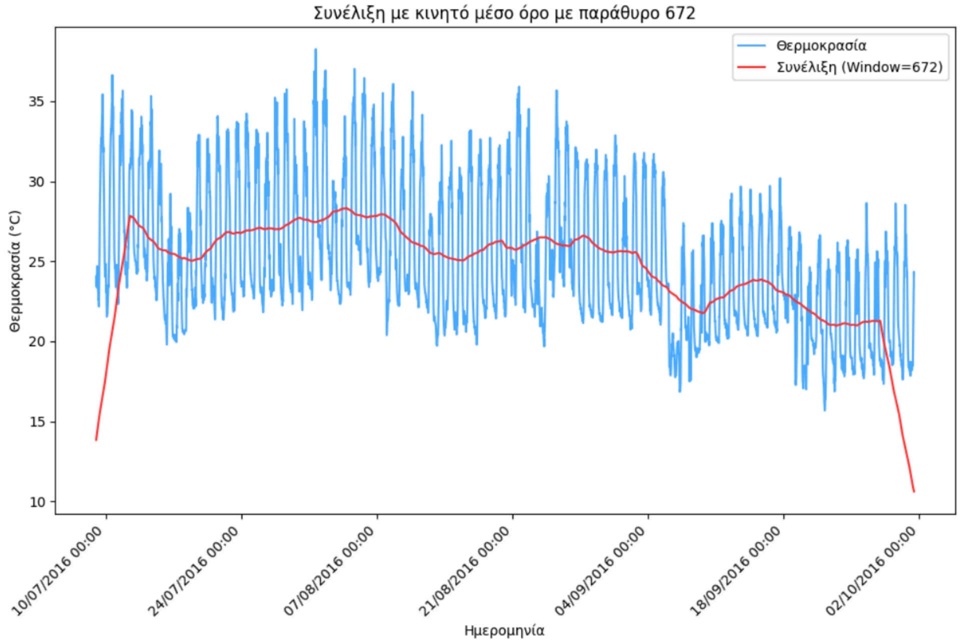
This chart smooths the temperature data in a short 1-hour window, capturing rapid changes and reducing noise. The resulting line is slightly smoothed, but still shows short-term fluctuations.

Convolution with a window size of 96 (1 day)



Using a 1-day window size provides a smoother curve that highlights daily trends and further reduces short-term fluctuations.

Convolution with a window size of 672 (1 week)



With a 1-week window size, the graph emphasizes long-term trends, smoothing out the weekly patterns and capturing the overall temperature trend more clearly.

Code

```
# Βιβλιοθήκες
import numpy as np
import pandas as pd
from scipy.signal import convolve
import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# Σύνολο Δεδομένων - temp-dataset.csv

file_path = '/kaggle/input/temperatures/temp-dataset-st-analysis.csv'
tempDataset = pd.read_csv(file_path)
tempDataset['Datetime'] = pd.to_datetime(tempDataset['Datetime'],
format='%d/%m/%Y %H:%M', errors='coerce')
tempDataset['Temperature'] = tempDataset['Temperature'].str.replace(',',
'.').astype(float)

# Υπολογισμός Μέσης τιμής, Διακύμανσης και Τυπικής απόκλισης
mean = tempDataset['Temperature'].mean()
variance = tempDataset['Temperature'].var()
standardDeviation = tempDataset['Temperature'].std()

print("Δεδομένα από την περιοχή του Αγίου Μάρκου Κέρκυρας, συλλεγμένα σε περίοδο
περίπου 3 μηνών, με συχνότητα δειγματοληψίας 'τιμή θερμοκρασίας/15 λεπτά'.")
print(f"Μέση Τιμή (Mean): {mean}")
print(f"Διακύμανση (Variance): {variance}")
print(f"Τυπική Απόκλιση (Standard Deviation): {standardDeviation}")

# Γράφημα
fig, ax = plt.subplots(figsize=(12, 6))

ax.plot(tempDataset['Datetime'], tempDataset['Temperature'],
label='Θερμοκρασία',color='#4AABFF')
ax.axhline(mean, color='#FF3333', linestyle='--', label=f'Μέση Τιμή:
{mean:.2f}°C')
ax.fill_between(tempDataset['Datetime'], mean - standardDeviation, mean +
standardDeviation, color='#00FF80', alpha=0.1, label=f'±1 Τυπική Απόκλιση:
{standardDeviation:.2f}°C')
ax.set_title('Θερμοκρασία σε Βάθος Χρόνου')
ax.set_xlabel('Ημερομηνία')
ax.xaxis.set_major_locator(mdates.DayLocator(interval=14))  # Two-week interval
ax.xaxis.set_major_formatter(mdates.DateFormatter('%d/%m/%Y %H:%M'))
plt.setp(ax.xaxis.get_majorticklabels(), rotation=45, ha='right')
ax.set_ylabel('Θερμοκρασία (°C)')
ax.legend()

plt.tight_layout()
plt.show()
```

```
# Αυτοσυνδιακύμανση και Αυτοσυσχέτιση
def autocovariance(series, lag):
        return series.cov(series.shift(lag))

lags = np.arange(1, 21)

autocovariances = [autocovariance(tempDataset['Temperature'], lag) for lag in lags]
autocorrelations = [tempDataset['Temperature'].autocorr(lag) for lag in lags]

# Γράφημα
plt.figure(figsize=(10, 10))

# Αυτοσυνδιακύμανση
plt.subplot(2, 1, 1)
plt.stem(lags, autocovariances, linefmt='#4AABFF', markerfmt='o', basefmt=' ')
plt.title('Αυτοσυνδιακύμανση')
plt.xlabel('Χρονοκαθυστέρηση')
plt.ylabel('Αυτοσυνδιακύμανση')

# Αυτοσυσχέτιση
plt.subplot(2, 1, 2)
plt.stem(lags, autocorrelations, linefmt='#4AABFF', markerfmt='o', basefmt=' ')
plt.title('Αυτοσυσχέτιση ')
plt.xlabel('Χρονοκαθυστέρηση')
plt.ylabel('Αυτοσυσχέτιση ')

plt.tight_layout()
plt.show()
```

```
# Κινητός Μέσος Όρος με Διαφορετικά Παράθυρα και Συνέλιξη

window_sizes = [4, 96, 672] # 4= 1 ώρα, 96= 1 ημέρα, 672= 1 εβδομάδα

# Γράφημα Συνέλιξης με Κινητό Μέσο Όρο
fig, ax = plt.subplots(3, 1, figsize=(10, 20))

for i, window_size in enumerate(window_sizes):
        kernel = np.ones(window_size) / window_size
        moving_average_convolution = np.convolve(tempDataset['Temperature'],
kernel, mode='same')

        ax[i].plot(tempDataset['Datetime'], tempDataset['Temperature'],
label='Θερμοκρασία',color='#4AABFF')
        ax[i].plot(tempDataset['Datetime'], moving_average_convolution,
label=f'Συνέλιξη (Window={window_size})', color='#FF3333')
        ax[i].set_title(f'Συνέλιξη με κινητό μέσο όρο με παράθυρο {window_size}')
        ax[i].set_xlabel('Ημερομηνία')
        ax[i].xaxis.set_major_locator(mdates.DayLocator(interval=14))
        ax[i].xaxis.set_major_formatter(mdates.DateFormatter('%d/%m/%Y %H:%M'))
        plt.setp(ax[i].xaxis.get_majorticklabels(), rotation=45, ha='right')
        ax[i].set_ylabel('Θερμοκρασία (°C)')
        ax[i].legend()

plt.tight_layout()
plt.show()
```