HPU2 Journal of Sciences:
Natural Sciences and Technology

journal homepage: https://sj.hpu2.edu.vn

Article type: Research article

Time series analysis and applications in data analysis, forecasting and prediction

Le-Hang Le*

University of Economics - Technology for Industries (UNETI), Hanoi, Vietnam

Abstract

Time series analysis is an essential field in data analysis, particularly within forecasting and prediction domains. Researching and building time series models play a crucial role in understanding and predicting the temporal dynamics of various phenomena. In mathematics, time series data is defined as data points indexed in chronological order and have a consistent time interval between consecutive observations. This can include data such as daily stock prices, annual national income, quarterly company revenue, and more. The advantage of time series data is that it can capture the state of a variable over time. In contrast, the world is constantly changing, and phenomena rarely remain static they typically exhibit variations over time. Therefore, time series data has highly practical applications and is used in various fields, including statistics, econometrics, financial mathematics, weather forecasting, earthquake prediction, electroencephalography, control engineering, astronomy, telecommunications, and signal processing. ARIMA, which stands for Auto Regressive Integrated Moving Average, is a widely used time series forecasting method in data science. It is a popular model for analyzing and predicting time-dependent data points. ARIMA combines autoregression, differencing, and moving averages to capture different aspects of time series data. In this paper, we study ARIMA, which is a significant model for analyzing and predicting time series data.

Keywords: Time series, data analysis, forecasting, prediction, arima

1. Introduction

A time series is a collection of values recorded at different points in time and can be used to describe changes over time. Examples of time series include monthly sales volume, daily stock prices, hourly temperatures, and daily COVID-19 infection counts.

* Corresponding author, E-mail: lehang1102@gmail.com

https://doi.org/10.56764/hpu2.jos.2024.3.1.20-29

Received date: 18-10-2023 ; Revised date: 22-11-2023 ; Accepted date: 01-12-2023

A time series is a sequence of values recorded over time, where each value in the time series is associated with a specific timestamp [1], [2]. Time series data is commonly used to model and forecast variables that change over time, such as stock prices, temperature, sales, and many other variables. It involves using historical data to predict future values [3]–[7]. This can be applied in various fields, including finance, weather forecasting, and energy consumption. Time series allows you to examine trends and changes over time, identify factors causing variations, and extract useful information. It can be used to validate assumptions about data, such as correlation and seasonality [8]–[13].

2. Meterials and Methods

2.1. Time series

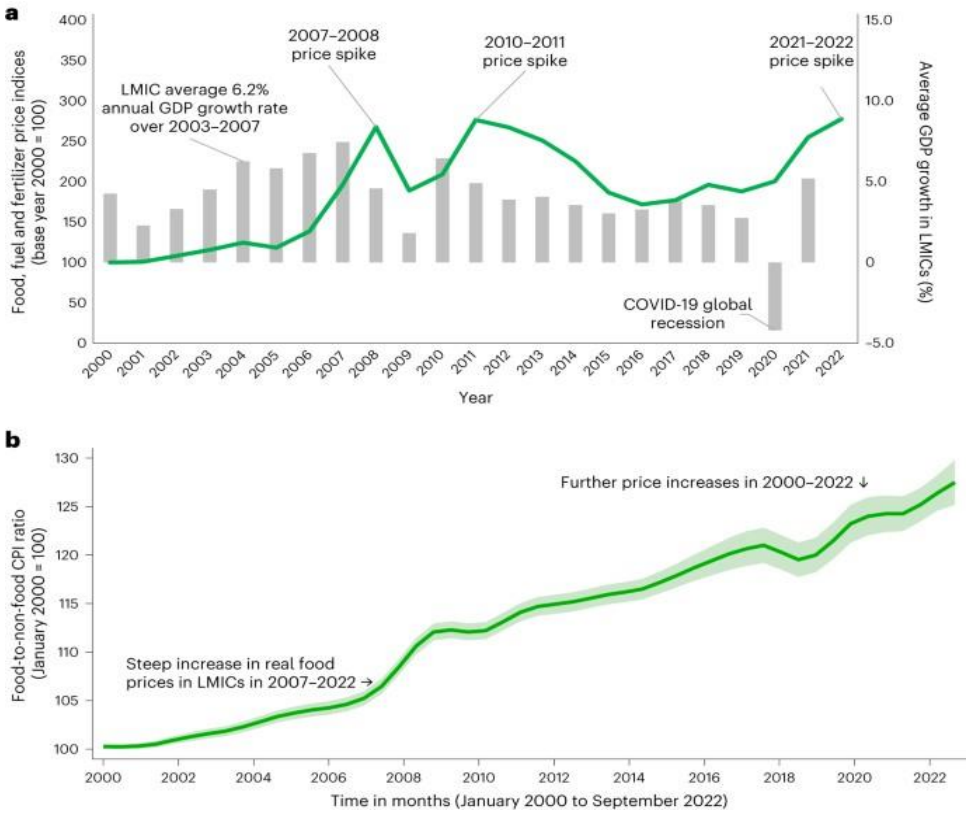

Figure 1. The trend component. (a) The trend of data reduction of food, fuel and fertilizer indices. (b) The trend of data increase of food.

Time series data can also be used to predict sudden events or unexpected changes in data, such as anomalies or outliers. To work with time series data, appropriate methods and tools are needed, such as the ARIMA (Autoregressive Integrated Moving Average) model, recurrent neural networks (RNNs), Long Short-Term Memory (LSTM) networks, or programming languages and libraries like Python with pandas and scikit-learn [1], [2].

Singular Value Decomposition (SVD) is an important method in linear algebra and data processing. It allows the decomposition of a not necessarily square matrix into the product of three special matrices: a unitary matrix U, a diagonal matrix Σ, and another unitary matrix, the transpose of U (U*). Below is the definition, properties, and a specific example of the SVD method [1]–[3].

Trend Component: It signifies the upward or downward direction of data points in a time series. The trend component is often depicted on a graph as a straight line or a smooth curve, Figure 1. A time series data without a trend component (meaning it doesn't exhibit an apparent increase or decrease) is considered stationary around its mean value [4]–[6].

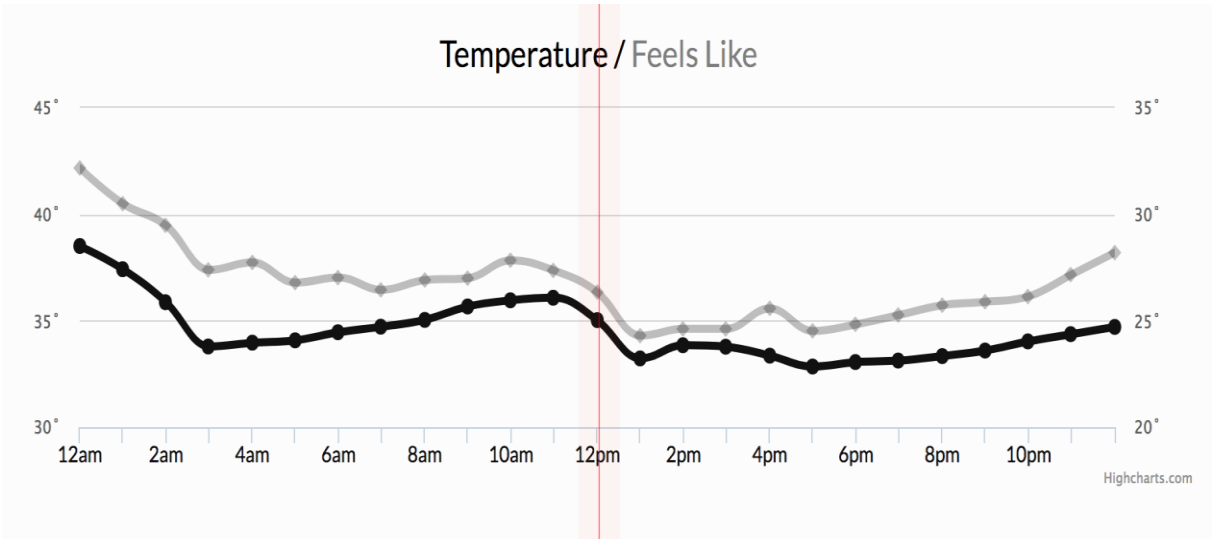

Figure 2. Representing changes in a time series over intervals.

Seasonal Component: This component represents the cyclic variation in the values of y calculated over short time periods. For example, the number of children with respiratory illnesses tends to increase during peak cold seasons in our country [7]–[9]. Figure 2 provides a representation of changes in a time series over intervals.

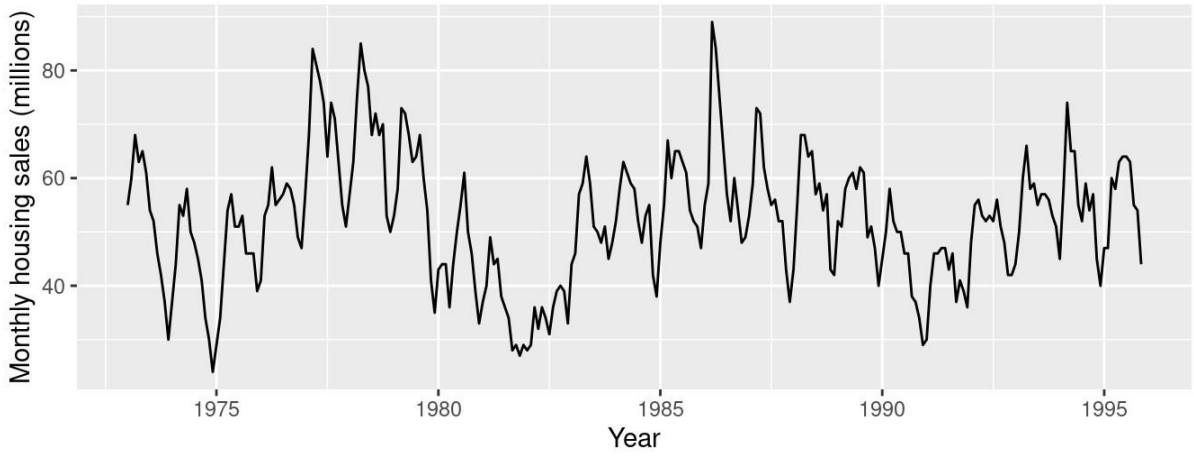

Figure 3. Representing the cyclical pattern in a time series.

Cyclic Component (Long-term): It reflects the long-term increase or decrease in the time series data revolving around the trend. Identifying cyclic components in long-term time series data can be challenging. Figure 3 provides a representation of the cyclical pattern in a time series. Figure 4 illustrates the time series in statistics.

Random Component: This is the opposite of cyclic components. The random component accounts for irregular fluctuations in the time series data and is often unpredictable. These fluctuations are typically caused by external factors [10]–[15].

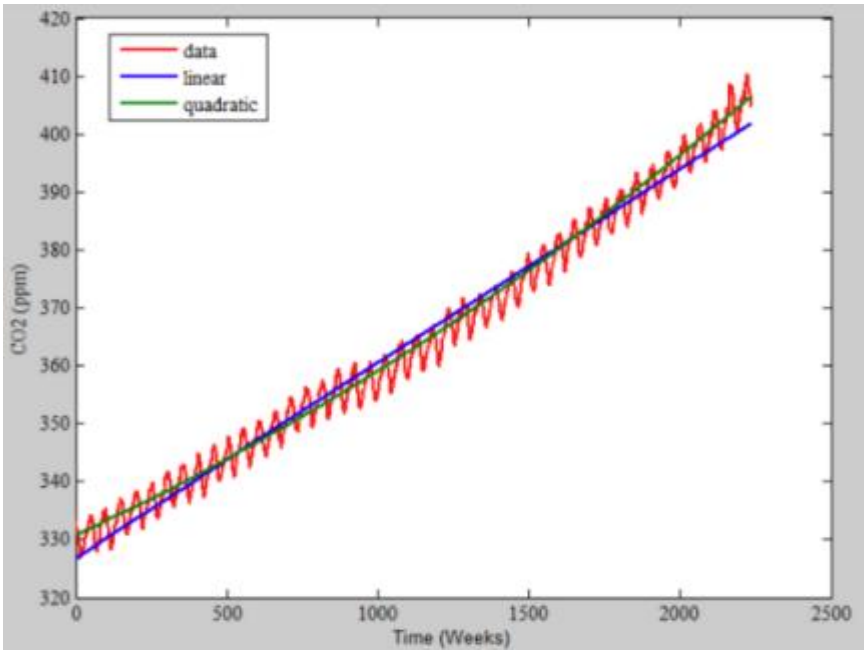

Figure 4. Time series in statistics.

According to time series diagram Y_t represents the quantitative value over time calculated at time t, through which we can determine the following models:

Additive model: $Y_t = T_t + S_t + C_t + I_t$.

Multiplicative model: $Y_t = T_t + S_t + C_t + I_t$.

Where: T is the Trend component; S is the Seasonality component; C is the Cyclical component; I is the Irregular component.

If the cyclical and seasonal components do not affect the overall level of the time series, it is advisable to use the additive model. Conversely, the multiplicative model is used if the seasonal component depends on the trend and cycle.

2.2. Characteristics of Time Series Data

The characteristics of time series data can be better understood by examining real-world examples from various fields (for example, Figure 4). Below is an example of quarterly profits for Johnson & Johnson.

Figure 5 depicts the quarterly profit chart for each share of Johnson & Johnson, provided by Professor Paul Griffin from the University of California's School of Management. This data includes 84 quarters (equivalent to 21 years) from the first quarter of 1960 to the last quarter of 1980. The goal is to build a time series model by observing key patterns in the past. In this case, we can observe a general upward trend and regular fluctuations added to the trend, seemingly repeating over the quarters.

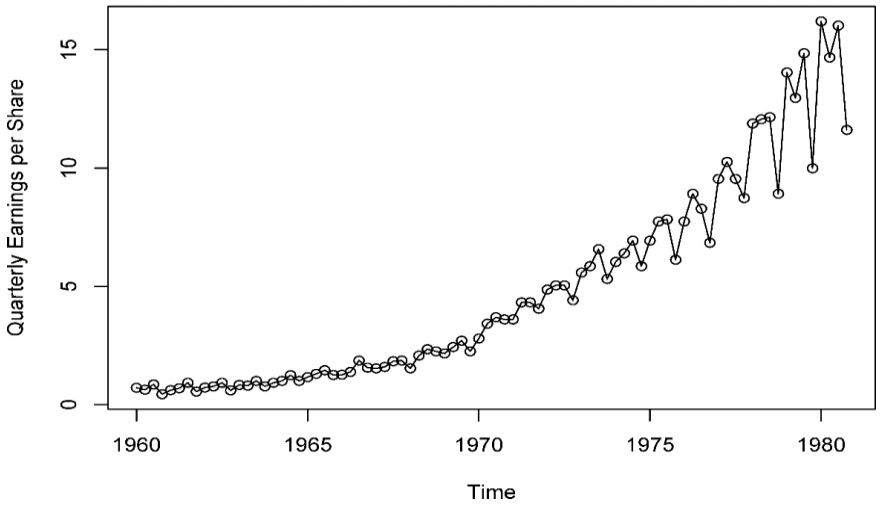

Figure 5. Johnson & Johnson's Quarterly Profits.

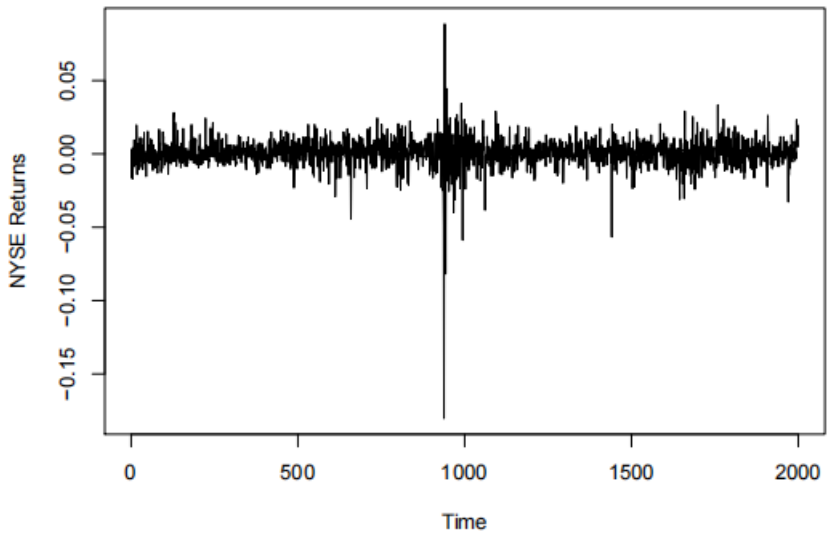

Figure 6. Financial Time Series Data.

Figure 6 provides an example of financial time series data, depicting the daily changes (or percentage changes) of the New York Stock Exchange (NYSE) from February 2, 1984, to December 31, 1991. In the graph, we can easily observe the market crash that occurred on October 19, 1987. The data in Figure 6 is a typical illustration of financial data. The time series average is stable, with an average return approximately equal to zero. However, the data's volatility (or standard deviation) varies over time. In fact, the data exhibits clustering of volatility cycles, meaning that periods of high volatility tend to cluster together. An important issue in the analysis of such financial data is forecasting the future volatility of returns. To address this issue, models like the ARCH and GARCH models were developed by Engle Bollerslev, as well as the stochastic volatility models of Harvey, Ruiz, and Shephard. Differencing is a crucial step in the ARIMA model. It is used to remove non-linearity and trends in time series data for analysis. The differencing process transforms the original data into a new time series with the aim of minimizing data dependency on previous time points [10], [12], [15], [16].

The theory behind differencing involves using the differences between consecutive values in a time series. Typically, if a time series has an increasing trend, the differences between consecutive values will also increase over time. When differencing is applied once to this series, the increasing trend in differences will decrease, and there will no longer be a long-term trend. Similarly, if a time series has a decreasing trend, the differences between consecutive values will decrease over time. When differencing is applied to such a series, the decreasing trend in differences will decrease, and there will be no long-term trend [17]–[20].

The mathematical formulation of differencing involves using the difference operator denoted as "d" to calculate the differences between values in the time series. The difference operator is represented as "B," where "B" is the backward shift operator defined as: $B * Y_t = Y_{t-1}$

The differencing formula can be expressed as the difference between the current value and the value at the previous time point. In the ARIMA model, differencing is typically used to reduce non-linearity and trends in time series data. After applying differencing to a time series, we check if the new series has become more predictable. The "AR" in ARIMA stands for "autoregressive." An AR model uses past values to predict the current value, assuming that the time series data exhibits autocorrelation, meaning that adjacent values in the series are correlated. The order "p" specifies how many past values are considered in the model, and it is one of the parameters that need to be determined when fitting an ARIMA model to a specific time series dataset. For example, Amazon's stock price today might depend on the cost from yesterday to earlier days (for example, Figure 7).

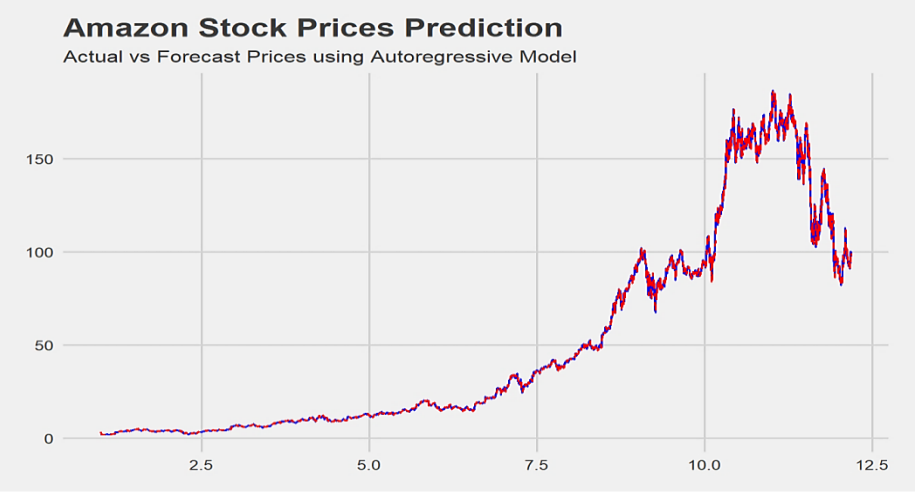

Figure 7. Amazon Stock Price.

The AR model idea is to regress its data in the past cycles.

In this context:

Y_t represents the current observation.

$Y_{t-1}, Y_{t-2}, \ldots$ are past observations.

$a_0, a_1, a_2, \ldots$ are regression parameters.

u_t is the random forecasting error at the current time, with an expected value of 0.

The linear function Y_t is a function of the past observations: Y_{t-1}, Y_{t-2}, etc.

When regressing Y_t on values in the time series with a lag, the delayed time series is used, resulting in an AR model. The number of past observations used in the autoregression model is called the order p of the AR model. For example:

AR(1) Model: $Y_t = a_0 + a_1 Y_{t-1} + u_t$

AR(2) Model: $Y_t = a_0 + a_1 Y_{t-1} + a_2 Y_{t-2} + u_t$

The parameters of the AR model are determined through linear regression methods. In more complex cases, when the time series data exhibits a more intricate pattern, alternative methods like ARIMA models may be used.

It's crucial to strike a balance between the model's complexity and its predictive ability. A model that is too simple might miss important data patterns, while an overly complex model can lead to overfitting and poor predictions. Notably, AR(p) models are most suitable for stationary time series data.

3. Results and Discussion

The value of p in the AR(p) model will affect the number of AR coefficients that need to be estimated. A higher p-value will require estimating more AR coefficients, potentially leading to a better model fit for the data. The AR model can be used for time series analysis and forecasting future values. However, it is suitable only for time series data with linear autocorrelation properties. If a time series lacks these properties, the AR model may not be appropriate. The AR model can be extended to include other components, such as the MA (ARMA model), differencing (I), or seasonal components (SARIMA model).

AR models need to be evaluated based on the accuracy of predictions and the precision of parameter estimates.

The process of analyzing time series data and forecasting is a method that utilizes historical values of factors such as prices, production, inflation, profits, etc., to predict the current value or forecast the change in the current value. Time series analysis falls under the category of quantitative forecasting as model's outcome is a quantitative value. It is commonly used in economic research for variables like GDP, inflation, growth rates, or market price studies. Some basic forecasting principles in this category include AR (Auto Regressive), MA (Moving Average), etc.

The Box-Jenkins method is considered one of the highly effective techniques for producing accurate and reliable forecasts. Its strength lies in providing information to analysts to select an appropriate model for the observed data. In contrast to other methods, where analysts assume a specificmodel and then estimate its parameters, Box-Jenkins methodology identifies a tentative model initially by comparing the sample autocorrelation and partial autocorrelation functions of the stationary time series data with the theoretical autocorrelation and partial autocorrelation functions of ARMA models.

ARIMA is a common and versatile forecasting model that uses historical data to make forecasts. This type of model serves as a basic forecasting technique that can serve as a foundation for more complex models. Based on these characteristics, the trainee decided to use the ARIMA model to experiment with time series data in the practical experiment.

The main steps in the Box-Jenkins methodology include:

Step 1: Model Identification

Historical data is used to tentatively identify a suitable ARIMA model.

Step 2: Model Estimation

Historical data is used to estimate the parameters of the tentative model.

Step 3: Model Checking for Adequacy

Various assessments are used to check the suitability of the tentative model, and if necessary, suggest a better model, which then becomes a new tentative one.

This methodology is essential for time series analysis and forecasting as it allows analysts to systematically identify, estimate, and validate models for making accurate predictions based on historical data.

Step 4: Forecasting

Once the final model has been selected, it is used to forecast future values of the time series.

In summary, time series data analysis typically involves the following steps:

Data Collection: Gather time series data from the respective source, such as databases, APIs, or direct data sources. For example, the following example will use stock price data from Yahoo Finance.

Exploratory Data Analysis (EDA): Before delving into detailed analysis, examine the data to understand basic characteristics such as line plots, percentage plots, descriptive statistics, and identify issues such as noise or missing values.

Data Preprocessing: Remove noise, impute missing values (if any), and normalize data if necessary.

Modeling: Build an appropriate model for the time series data. For example, the example below will use the ARIMA model.

Model Evaluation: Assess the performance of the model using evaluation methods like Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), or posterior checks for Bayesian models.

Prediction and Evaluation: Utilize the model to forecast future values and evaluate the accuracy of the predictions.

We give a specific example of time series data analysis using Python and the pandas, numpy, and statsmodels libraries:

```
# Import the necessary libraries:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima_model import ARIMA
# Step 1: Read data from a CSV file or another data source:
data = pd.read_csv('stock_price.csv')
data['Date'] = pd.to_datetime(data['Date'])
data.set_index('Date', inplace=True)
# Step 2: Exploratory Data Analysis (EDA) - Plotting:
plt.figure(figsize=(12, 6))
plt.plot(data['Close'], label='Closing Price')
plt.title('Stock Price Over Time')
plt.xlabel('Time')
plt.ylabel('Stock Price')
plt.legend()
```

```
plt.show()
# Step 3: Data Preprocessing (if needed)
# Example: Handling missing values
data['Close'].fillna(method='ffill', inplace=True)
# Step 4: ARIMA Modeling
model = ARIMA(data['Close'], order=(1, 1, 1))
model_fit = model.fit(disp=0)
# Step 5: Model Evaluation
mse = ((model_fit.fittedvalues - data['Close']) ** 2).mean()
print(f'Mean Squared Error: {mse}')
# Step 6: Prediction and Evaluation
predicted = model_fit.predict(start=len(data), end=len(data)+10, typ='levels')
plt.figure(figsize=(12, 6))
plt.plot(data['Close'], label='Closing Price')
plt.plot(pd.date_range(start=data.index[-1], periods=11, closed='right'), predicted,
label='Predicted')
plt.title('Stock Price Prediction')
plt.xlabel('Time')
plt.ylabel('Stock Price')
plt.legend()
plt.show()
```

In this problem, we perform time series data analysis for stock price data using Python. We provide the steps as follows.

Import the necessary libraries.

Read the data from a CSV file and set the 'Date' column as the index.

Plot the stock price data for exploratory data analysis.

Preprocess the data if needed (e.g., handling missing values).

Create an ARIMA model and fit it to the data.

Evaluate the model by calculating the Mean Squared Error (MSE).

Make predictions for future time periods and visualize the results.

This example provides a clear and structured approach to time series analysis using Python and ARIMA modeling.

4. Conclusion

Time series analysis is a valuable technique for anyone dealing with temporal data. It provides the tools and methodologies needed to understand historical patterns, make informed predictions, and ultimately make better decisions. As technology advances, the importance of time series analysis will continue to grow, making it an essential skill for data scientists, economists, and analysts across various industries. In this article, we've only scratched the surface of time series analysis. More advanced models, techniques, and considerations exist for dealing with complex temporal data. Nonetheless, this overview should serve as a solid foundation for diving deeper into the fascinating world of time series analysis. Whether forecasting stock prices, analyzing climate data, or studying customer behavior, timeseries analysis offers a powerful toolkit for extracting meaningful insights from your data.

References

[1] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," *Int. J. Forecast.*, vol. 36, no. 3, pp. 1181–1191, Jul. 2020, doi: 10.1016/j.ijforecast.2019.07.001.

[2] A. Carriero, T. Clark, and M. Marcellino, "Large vector autoregressions with stochastic volatility and flexible priors," *Working paper (Federal Reserve Bank of Cleveland),* Jun. 2016, doi: 10.26509/frbc-wp-201617.

[3] H. Hewamalage, C. Bergmeir, K. Bandara, "Recurrent neural networks for time series forecasting: current status and future directions," *Int. J. Forecast.*, vol. 37, no. 1, pp. 388–427, Jan. 2021, doi:10.1016/j.ijforecast.2020.06.008.

[4] L. Qu, W. Li, W. Li, D. Ma, and Y. Wang, "Daily long-term traffic flow forecasting based on a deep neural network," *Expert Syst. Appl.*, vol. 121, pp. 304–312, May 2019, doi: 10.1016/j.eswa.2018.12.031.

[5] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renew. Sustain. Energy Rev.*, vol. 74, pp. 902–924, Jul. 2017, doi:10.1016/j.rser.2017.02.085.

[6] X. Yang, F. Yu, and W. Pedrycz, "Long-term forecasting of time series based on linear fuzzy information granules and fuzzy inference system," *Int. J. Approx. Reason*, vol. 81, pp. 1–27, Feb. 2017, doi: 10.1016/j.ijar.2016.10.010.

[7] Z. Chen, M. Ma, T. Li, H. Wang, C. Li, "Long sequence time-series forecasting with deep learning: A survey," *Inf. Fusion*, vol. 97, p.101819, Sep. 2023, doi: 10.1016/j.inffus.2023.101819.

[8] H. Zhou, J. Li , S. Zhang, S. Zhang, M. Yan, and H. Xiong, "Expanding the prediction capacity in long sequence time-series forecasting," *Artif. Intell.*, vol. 318, p. 103866, May 2023, doi: 10.1016/j.artint.2023.103886.

[9] K. Bandara, C. Bergmeir, and S. Smyl, "Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach," *Expert Syst. Appl.*, vol. 140, p. 112896, Feb. 2020, doi:10.1016/j.eswa.2019.112896.

[10] G. Athanasopoulos, R. J. Hyndman, H. Song, and D. C. Wu, "The tourism forecasting competition," *Int. J. Forecast.*, vol. 27, no. 3, pp. 822–844, Jul. 2011, doi: 10.1016/j.ijforecast.2010.04.009.

[11] M. Assaad, and H. Cardot, "A new boosting algorithm for improved time-series forecasting with recurrent neural networks," *Inf. Fusion*, vol. 9, no.1, pp. 41–55, Jan. 2008, doi: 10.1016/j.inffus.2006.10.009.

[12] S. Ben Taieb, G. Bontempi, A. F. Atiya, and A. Sorjamaa, "A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition," *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7067–7083, Jun. 2012, doi:10.1016/j.eswa.2012.01.039.

[13] C. Bergmeir, R. J. Hyndman, and B. Koo, "A note on the validity of cross-validation for evaluating autoregressive time series prediction," *Comput. Stat. Data Anal.*, vol. 120, pp. 70–83, Apr. 2018, doi: 10.1016/j.csda.2017.11.003.

[14] M. Mudelsee, "Trend analysis of climate time series: A review of methods," *Earth-Science Rev.*, vol. 190, no. 8, pp. 310–322, Mar. 2019, doi: 10.1016/j.earscirev.2018.12.005.

[15] D.S. Stoffer, and H. Ombao, "Editorial: special issue on time series analysis in the biological sciences," *J. Time Ser. Anal.*, vol. 33, no. 5, pp. 701–703, Sep. 2012, doi: 10.1111/j.1467-9892.2012.00805.x.

[16] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nat. Med.*, vol. 25, no. 1, pp. 44–56, Jan. 2019, doi: 10.1038/s41591-018-0300-7.

[17] J.H. Böse *et al.*, "Probabilistic demand forecasting at scale," *Proc. VLDB Endow.*, vol. 10, no. 12, pp. 1694–1705, Aug. 2017, doi: 10.14778/3137765.3137775.

[18] T. G. Andersen, T. Bollerslev, and N. Meddahi, "Correcting the errors: Volatility forecast evaluation using high-frequency data and realized volatilities," *Econometrica*, vol. 73, no. 1, pp. 279–296, Jan. 2005, doi: 10.1111/j.1468-0262.2005.00572.x.

[19] M. Knott, M. Hollander, and D. A. Wolfe, "Nonparametric statistical methods," *J. R. Stat. Soc. Ser. A*, vol. 137, no. 2, p. 264, Aug. 1974, doi: 10.2307/2344557.

[20] R. J. Hyndman, and A. B. Koehler, "Another look at measures of forecast accuracy," *Int. J. Forecast.*, vol. 22, no. 4, pp. 679–688, Oct. 2006, doi: 10.1016/j.ijforecast.2006.03.001.