

3945 – Advanced Machine Learning, Semester A 2025

Home Assignment 3: Time Series Forecasting using ARIMA model

Lecturer: Dr. Leon Anavy

In this assignment, you will implement and evaluate a simple ARIMA model for time series forecasting. You will demonstrate your understanding of time series data, preprocessing techniques, and ARIMA modeling through a structured set of tasks.

You should submit a single ipython notebook (.ipynb) file that includes all the code and outputs.

You are encouraged to use methods and techniques covered in class but are also welcome to use any other relevant method you find suitable for the task if you justify your decision. Make sure to use proper ML and DS methodologies when needed (Data preprocessing, Data splits, etc.).

Submission Guidelines

- Submit a single zip file containing the dataset, and a notebook (.ipynb) titled **HW3_ID1_ID2.ipynb** that includes:
 - Appropriately formatted cells (markdown) describing the design, implementation, and showing any calculations.
 - All code and outputs.
 - Explanations for each step.
 - Observations and conclusions for each section.
- Ensure your notebook runs from start to finish without errors.

You may use external libraries as needed. Specify all dependencies in the notebook.

Grading

- The work will be assessed based on methodology, execution, and presentation.
 - All figures and plots should be correctly labeled.
 - Grading will be based on correctness, elegance of solution, and style (comments, naming conventions, etc.).
-

Part 1: Design the Algorithm

- Describe the design you will use to implement ARIMA. Clearly describe the theoretical and practical considerations for the algorithm.
- Ensure that your design captures the main relationships and evaluates the residuals appropriately.

Part 2: Implement the Algorithm

- Based on your design, implement the algorithm.
- Write efficient, vectorized code whenever possible.
- You must explain and justify the design.
- Develop your implementation of ARIMA from scratch.

Part 3: Demonstrate the Algorithm on simulated data

- Test your ARIMA implementation on a simple, simulated dataset you generate (a "unit test") to validate correctness and debug any issues.
- Justify the parameters you set for the algorithm.

Part 4: Experiments and Analysis of a real world dataset

- Using the provided dataset, perform exploratory data analysis, and use any other methods discussed in class or in literature to evaluate the data.
- Prepare and split the data according to best-practices discussed in class.
- Generate plots and visualizations for all stages.
- After pre-processing, select parameters for a manual ARIMA and a library ARIMA, fit the models, and compare predictions.
- Ensure you evaluate ACF and PACF where relevant.
- Design and run experiments to test your ARIMA implementation on the provided dataset.
- Evaluate the performance and compare to the library implementation of ARIMA.

Part 5: Report

- Share your learnings about the difference between the library and manual ARIMA implementations, the performance of the manual ARIMA on a simulated and real dataset (if any), as well as what you learned about the data.

Dataset – Amazon Stock Prices

- In this assignment, you are using your technical skills in what is likely a new area for many of you, don't be discouraged, you do not need a background in finance to succeed. You are encouraged to analyze the data as you would any other dataset.
- Using the yfinance package (you can pip install it), download Amazon's stock data with the ticker AMZN. Make sure to take the **maximum** snapshot.
- The dataset contains historical stock prices for Amazon, including market open, close, high, low, adjusted close, and volume information. Open and Close are the first and last traded prices of the day, while High and Low mark the highest and lowest prices recorded within that same day. Volume indicates the total number of shares traded, reflecting how actively a stock is bought and sold. Adjusted Close modifies historical prices to account for corporate actions like splits and dividends, making it easier to compare a stock's performance over time.
- Make sure to perform EDA.
- If you are having difficulty, we have provided a snapshot with the assignment.

Experimentation Guidelines

- Clearly describe the experimental setup, including:
 - Dataset characteristics.
 - Data preprocessing steps.
 - Parameter selection.

- Justification for key design and implementation choices.
- Use appropriate plots to demonstrate results.
- Discuss the observations and insights derived from the results.

Important Note Regarding Generative AI: While you may use Generative AI models to assist you, ensure all submitted work is original and that you fully understand it. You may be asked to explain your work in an in-person review.

Pair Work: Submit in pairs, with one submission per pair.