Homework 1: Time Series Analysis, and Dimensionality Reduction (PCA) (100 points)

Prof. Amine Aboussalah

Due Date: Tuesday 13th October 2025.

Submission will be done on Brightspace.

Datasets

1. **Time Series Analysis**: U.S. Quarterly GDP data (1947-2023) - Available from the Federal Reserve Economic Data (FRED) repository (Gross Domestic Product (GDP) | FRED | St. Louis Fed)
2. **Principal Component Analysis**: UCI Machine Learning Repository - Wine Quality Dataset with 12 attributes (Wine Quality - UCI Machine Learning Repository)

Overview

This homework will develop your practical skills in two key machine learning areas: time series forecasting, and dimensionality reduction. You'll implement these techniques from scratch and apply them to real-world datasets.

Task 1: Time Series Analysis with Linear Models (50 points)

Objective

Analyze GDP time series data to identify patterns, determine appropriate linear model parameters, and compare model performance.

Instructions

1. **Data Exploration & Preprocessing** (4 points)
    - Load the provided quarterly GDP dataset (1947-2023) (2 points)
    - Create time series plots of the data (2 points)

2. **Model Determination** (22 points)
    - Decompose the series into trend, seasonal, and residual components. Justify whether you will be using an additive or multiplicative decomposition. (5 points)
    - Test for stationarity using the Augmented Dickey-Fuller test (2 points)
    - Document any transformations applied to achieve stationarity, and specify the chosen parameter d (5 Point)
    - Analyze ACF/PACF plots to determine appropriate p, q parameters (5 points)
    - Identify seasonal components and determine P, D, Q, s parameters if necessary (3 points)
    - Justify your choice of the linear model you are using and its parameters with clear explanations of your reasoning (2 points)
3. **Model Training & Evaluation** (12 points)
    - Split the data into training (80%) and test (20%) sets. Make sure to avoid the look-ahead bias. (2 points)
    - Train your selected linear model on the training data (3 points)
    - Generate forecasts for the test period (2 points)
    - Evaluate the model using multiple metrics: RMSE, MAE, MAPE (1 point)
    - Provide detailed interpretation of each metric (2 points)
4. **Comparative Analysis** (12 points)
    - Train a second model (either a different linear model specification or a different type of model) (3 points)
    - Compare performance with your first model (3 points)
    - Discuss strengths and weaknesses of each approach (3 points)

Task 2: Stationarity of Autoregressive Models (10 points)

Objective

Demonstrate that an autoregressive (AR) model is stationary if and only if all the roots of its characteristic polynomial lie outside the unit circle.

Instructions

- **Part 1 – Simple Case (2 points):** Prove this property for a first-order autoregressive model, AR(1).

- **Part 2 – General Case (8 points):** Extend the proof to an autoregressive model of arbitrary order, AR(p).

Task 3: Principal Component Analysis Implementation (40 points)

Objective

Implement PCA from scratch and analyze its effectiveness for dimensionality reduction.

Instructions

1. **Implementation** (25 points)
    - Code a complete PCA algorithm without using existing PCA libraries
    - Your implementation must include:
        - Data centering/standardization (5 points)
        - Covariance matrix computation (5 points)
        - Eigenvalue/eigenvector calculation (5 points)
        - Projection of data onto principal components (5 points)
        - Variance explained calculation (5 points)
2. **Visualization & Analysis** (10 points)
    - Apply your PCA implementation to the provided high-dimensional dataset (4 points)
    - Create scree plots showing explained variance by each principal component (3 points)
    - Visualize data in 2D and 3D using the top principal components (3 points)
3. **Parameter Analysis** (5 points)
    - Study how changing the number of components affects:
        - Performance of a simple downstream ML task (3 points)
        - Analysis of results and insights (2 points)

General Requirements (up to -10 points penalty for not meeting these)

- Submit a Jupyter notebook
- All code must be original, except where specifically noted
- Code should be well commented, explaining each step of your thought process
- Include error handling and validation in your implementations
- Provide clear documentation of all hyperparameters and design choices