University *of Ljubljana*
Faculty *of Mathematics and Physics*

Department of Physics

Time Series Analysis using Auto Regressive models

11. Task for Model Analysis I, 2023/24

Author: Marko Urbanč
Professor: Prof. dr. Simon Širca
Advisor: doc. dr. Miha Mihovilovič

Ljubljana, August 2024

Contents

1 Introduction **1**

2 Task **2**

2.1 Spectra of Signals 2

2.2 Forecasting Signals 2

3 Solution Overview **3**

4 Results **4**

5 Conclusion and Comments **4**

1 Introduction

In physics we often encounter time series data as we measure the evolution of a system in time. Today we'll be having a look at one of the plethora of methods in our time series analysis arsenal, called Auto Regressive (AR) models. Then we'll apply this method to a few different datasets to study its performance, applicability and behavior. AR models are a class of linear models that predict the future values of a time series based on its past values. Based on the coefficients of the model, we can infer the underlying dynamics of the system or find the signals spectrum.

An AR model of order p is defined as:

$$X_t = \sum_{i=1}^{p} a_i X_{t-i} + \epsilon_t \,, \tag{1}$$

where X_t is the time series at time t, a_i are the coefficients of the model and ϵ_t is the error term which we assume to be uncorrelated and with zero mean. In our idealized case, we'll only have a short look at how noise affects the model. For the coefficients a_i we can without loss of generality divide by the first coefficient, a_1, which yields a model with a leading coefficient of 1. The order of the model, p, is a hyperparameter that determines how many past values of the time series we use to predict the future value. The model is commonly used in the context of machine learning but today we'll be solving for the coefficients more hands-on, with out explicit use of optimization algorithms.

A popular method to determine the AR model coefficients are the Yule-Walker equations. These are a set of p linear equations that can be solved for the coefficients which are given by the autocorrelation of the time series $\{X_t\}$. The autocorrelation is defined as:

$$r_n = \sum_{i=1}^{p} a_i r_{n-i} \,, \tag{2}$$

where r_n is the autocorrelation at lag n. Essentially we need to solve a Toeplitz system of equations. The system of equations is given by:

$$\mathrm{R}\mathbf{a} = \mathbf{r} \,, \tag{3}$$

where R is the autocorrelation matrix, given as:

$$\mathrm{R} = \begin{bmatrix} r_0 & r_1 & \cdots & r_{p-1} \\ r_1 & r_0 & \cdots & r_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{p-1} & r_{p-2} & \cdots & r_0 \end{bmatrix} , \tag{4}$$

and $\mathbf{r}$ and $\mathbf{a}$ are the autocorrelation vector and coefficient vector respectively, given as:

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_p \end{bmatrix} \qquad \text{and} \qquad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} . \tag{5}$$

Stability of the model is ensured by the roots of the characteristic polynomial of the model which must lie inside the unit circle on the complex plane. This is something we'll have to keep in mind when analyzing the results, as any roots outside the unit circle will make the model unstable.

2 Task

2.1 Spectra of Signals

The instructions want us to plot the spectra of a few different signals that have been provided. The main comparison should be between the spectra given by the AR model and the spectra given by the Fourier transform. The FFT algorithm is truly a marvel of modern mathematics and computing so it will definitely be a tough competitor. The signals that have been provided are plotted in Figure 1 with their respective spectra computed using FFT.

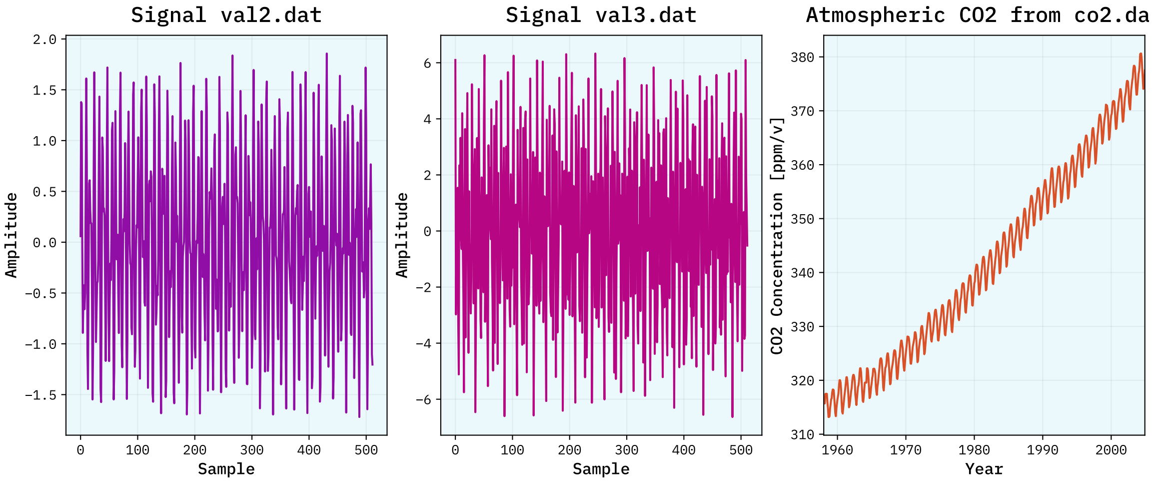

Figure 1: The signals that we'll be analyzing in the first subtask.

The instructions also want us to explore the various properties of the AR model and how it behaves when we change the order of the model as well as the number of points we use to evaluate the spectra. The spectra can be calculated using the following formula:

$$S(f) = \frac{1}{|1 - \sum_{i=1}^{p} a_i e^{-2\pi i f i}|^2} , \tag{6}$$

however I used the `scipy.signal.freqz` for ease of use and to avoid any mistakes.

2.2 Forecasting Signals

In the second subtask we've been provided some additional signals that we'll be using in combination with the signals from the first subtask to study the performance of the AR model in forecasting. The supplied signals are plotted in Figure 2.

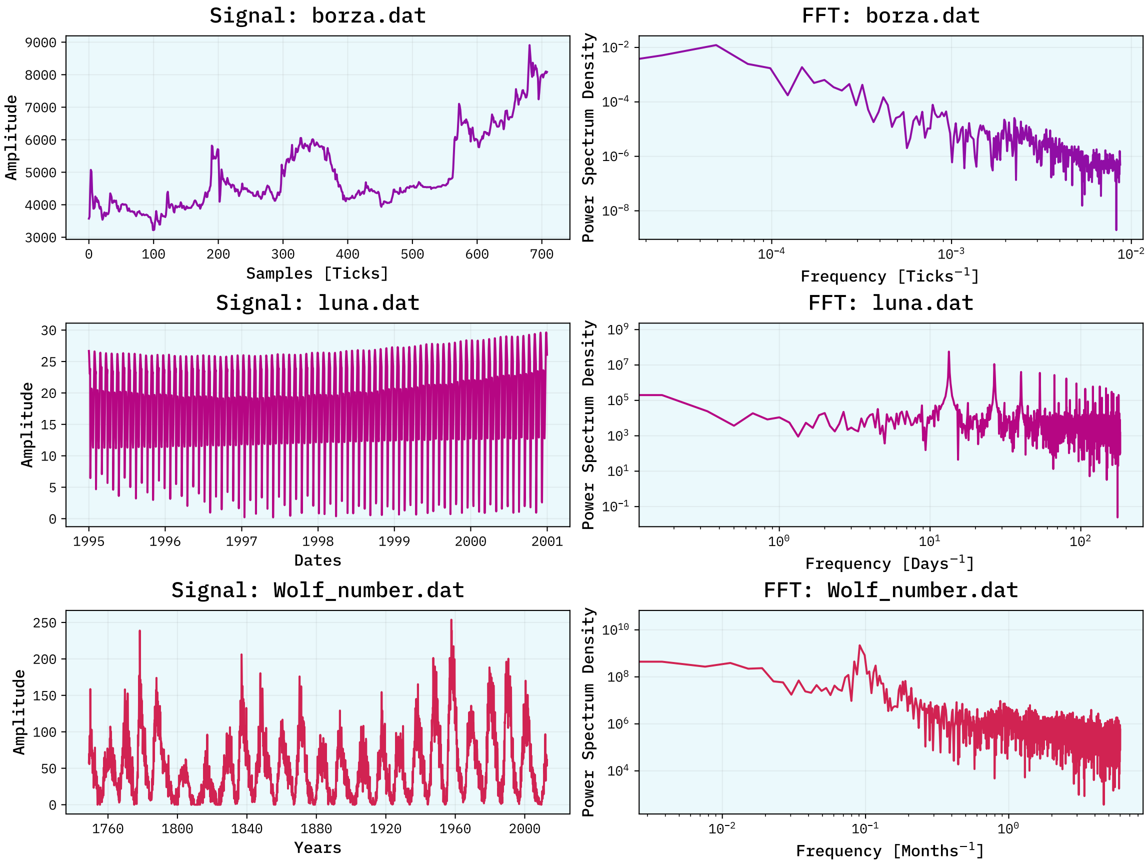

Figure 2: The signals that we'll be adding to our analysis in the second subtask.

They represent a stock's value, ephemeris data for the moon and Wolf sunspot numbers. The instructions want us to forecast the signals using the AR model and compare the results with the actual data. We can do so by providing half of the data to the model and then comparing the forecasted values with the actual values of the other half. We should also have a look at how noise affects the model and the forecasted values.

3 Solution Overview

Due to my speedrunning of these tasks this task lacks any extra bells and whistles. I implemented the Yule-Walker method using `numpy` and `scipy` by making a class `YuleWalker` that can be used to fit the model and evaluate PSDs. It takes inputs of the time series and the order of the model. I really like classes in Python. They really serve well as containers for variables and their associated functions. The class also has a method to forecast the time series using the AR model. The results are presented in the following sections. Before we proceed we need to detrend some of the signals. This was most commonly done by subtracting the mean of the signal, however I encountered two special cases. The first was the CO2 atmospheric concentration signal to which I fit a square polynomial and subtracted the fit from the signal. The second signal was the stock value signal which I detrended by the linear fit of the signal. The signals with their respective trends are plotted in Figures 3 and 4.

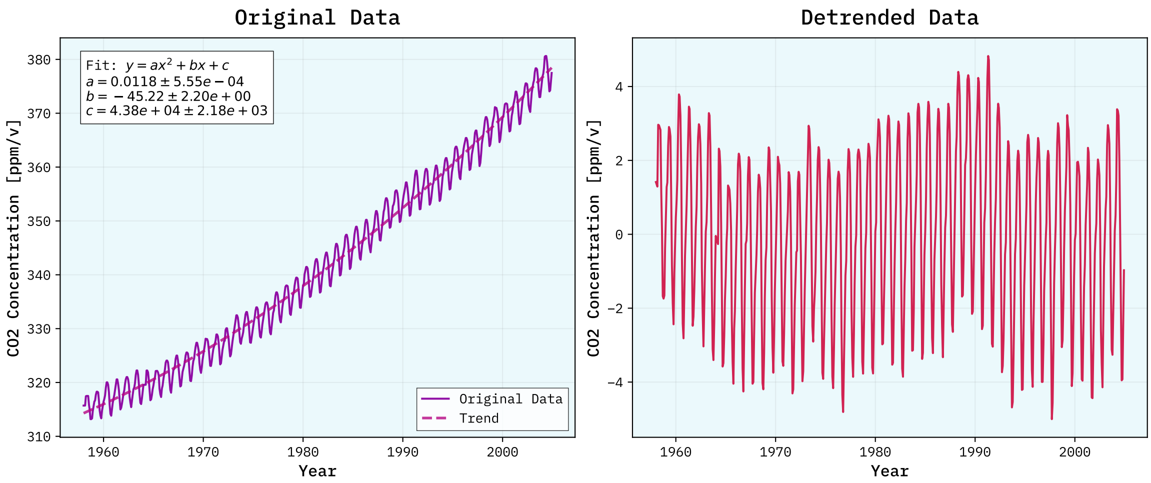

Figure 3: The CO2 atmospheric concentration signal with a square polynomial fit.

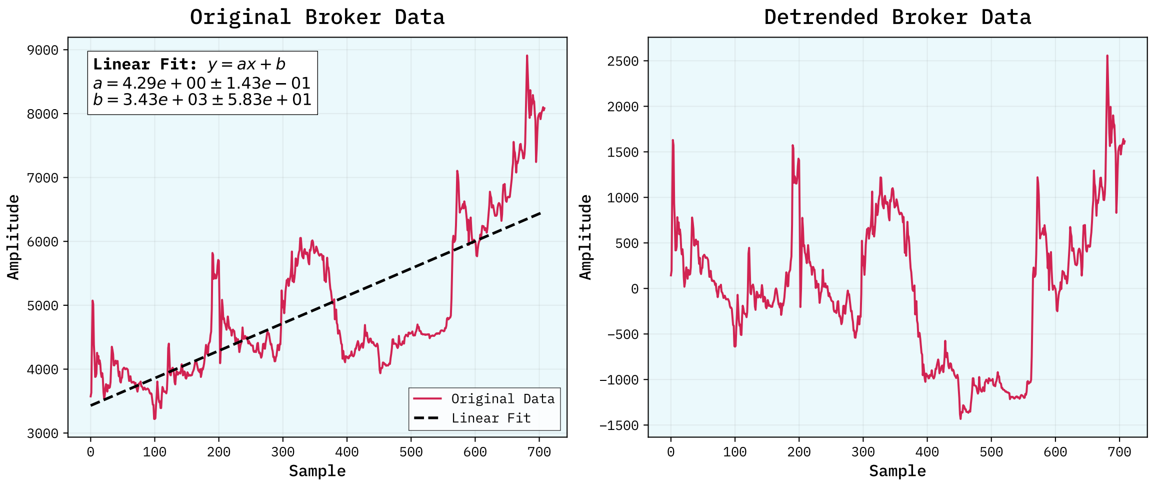

Figure 4: The stock value signal with a linear fit.

4 Results

5 Conclusion and Comments