Microsoft Stock Price in the Year 2022

Edward Bickerton, rw19842@bristol.ac.uk
Department of Computer Science, University of Bristol

***Abstract*—This report was completed as coursework for the module Advanced Financial Technology (COMSM0090) and explains how an autoregressive model can be used to predict time series data.**

***Index Terms*—Time Series, Stationarity, Partial Autocorrelation Function, Autoregressive Model.**

I. Introduction

Fig. 1 shows the close price of Microsoft stock for the year 2022, for which raw data can be found at [1]. We can see that 2022 was a bad year for holders of Microsoft stock as there is a clear downtrend, the close price decreased from \$335 on the 3rd of January 2022 by 28% to \$240 on the 30th of December 2022.

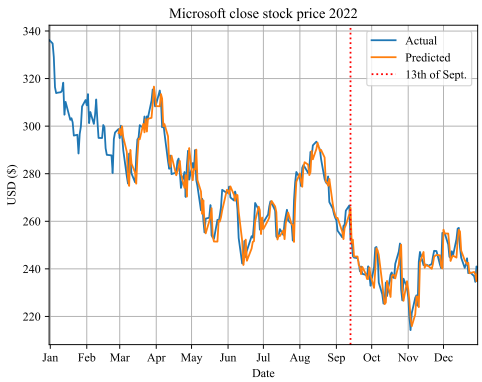

Fig. 1. Microsoft close stock price in dollars (USD) from the 31st of December 2021 to the 30th of December 2022 (blue line), predicted values are shown in the orange line.

Notably, the close price saw a 14% increase from its March low of \$276 to \$315 on the 29th of March, and during the months from June to August there was a 21% rise from \$242 to \$293 before continuing its decline.

Such price movements pose as lucrative buying and selling opportunities for traders and serves as the motivation for fitting an autoregressive (AR) model to this data.

II. Stationarity

Before attempting to fit the AR model it is worth understanding what it means for time series data to be stationary. A stationary process is one in which for all possible lags, k, the distribution of $y_t,\ y_{t+1},\ \ldots,\ y_{t+k}$, doesn't depend on t [2]. That is for any subsection of a stationary time series the statistical properties such as mean and standard deviation are constant with respect to t. Hence, a time series exhibiting either an upward or downward trend is excluded from stationarity on account of it having a changing mean, similarly time series with seasonal or cyclic variations are not stationary. For an example of stationary data see fig. 2.

Therefore stock price data is rarely stationary and the close price of Microsoft is no exception to the rule due to its downward trend, likewise, stock prices often experience periods of high and low volatility – also excluding them from stationarity (varying standard deviation). As expected, performing an Augmented Dickey-Fuller (ADF) test on the Microsoft data gives a p-value of 0.10 so there is insignificant evidence to suggest that the data is stationary.

III. Autoregressive Model

Put simply autoregression uses past values to predict future values [3]. It does this by modelling, y_t, the value of the time series at time t as:

$$y_t = c + \sum_{i=1}^{p} \phi_i \cdot y_{t-i} + \varepsilon_t,$$

where c is a constant, the ϕ_i are regression constants and ε_t is an error term which is assumed to have a constant variance and mean of 0. Such a model is known as AR(p) since it depends on the p most recent previous values. The constant c and regression constants, ϕ_i, can be learned via linear regression where $(y_{t-1},\ y_{t-2},\ \ldots,\ y_{t-p}) \in \mathbb{R}^p$ is the feature vector and y_t is the target.

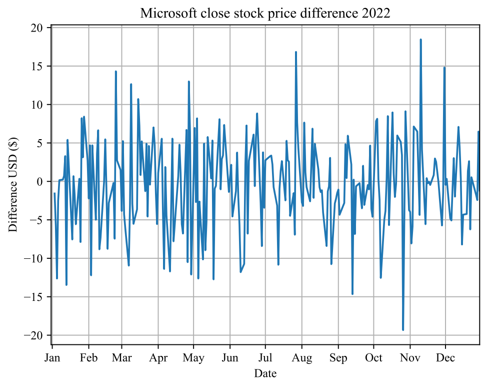

Fig. 2. Difference between Microsoft close stock price and that of the previous close price, in dollars (USD) from the 3rd of January 2022 to the 30th of December 2022.

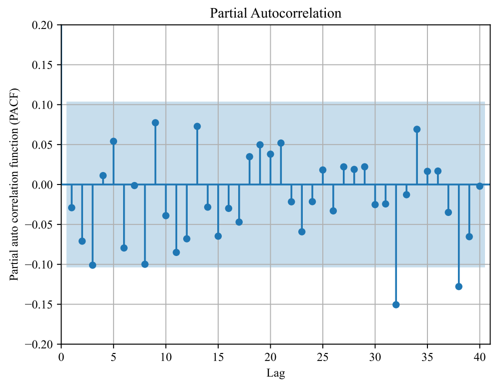

Fig. 3. Partial autocorrelation function off the differenced data, the shaded area represents the 10% significance threshold.

A constraint of autoregression is that it assumes stationarity of the time series. However, raw non-stationary data can often be transformed into stationary data as is the case for the Microsoft close stock price. Fig. 2 shows the stock price after differencing; we can see that the trend in the data has been removed by this transformation and an ADF test gives a p-value of 2.7e-19 suggesting that the transformed data is stationary.

Thus, I will fit an AR(p) model to the transformed Microsoft data shown in fig. 2 and undo the transformation to predict the close price of the next day by $y_t \approx y_{t-1} + \Delta_t$, where Δ_t is the difference predicted by the AR(p) model. I will use the close price from the 31st of December 2021 to the 13th of September 2022 as training data (roughly 70%) and the remaining will be used as test data.

One question remains, what value of p should be used? To determine this, we use the partial autocorrelation function (PACF), the plot of which can be seen in fig. 3.

From fig. 3 we see that there are significant values of the PACF for lags: 3, 8, 32 and 38, thus I set $p = 38$. However, a model with 39 parameters (38 regression constants and c) would be overly complex and likely perform poorly on unseen data, so I set $\phi_i = 0$ for $i \notin \{3, 38\}$.

The model achieved a root mean squared error (RMSE) of \$5.57 using values of c, ϕ_3 and ϕ_{38} learned via linear regression. Predictions of this model can be seen in fig. 1.

While this sounds promising, and the predicted (orange) line is indeed very close to the (blue) actual line in fig. 1; it's worth noting that a "dumb" model which predicts the future close price will be equal to that of todays, achieves an RMSE of \$5.91 and predicting values just one time step in advance leaves little opportunity to make money using this model.

In general AR models are best for making short-term predictions, although these models can predict many time steps into the future, these models lose predictive power for large future horizons. For example, the AR model in this report can be used to predict the close price in two days using $y_t \approx y_{t-2} + \Delta_{t-1} + \Delta_t$, however, the RMSE increases to \$8.18 only slightly beating the \$8.28 RMSE the "dumb" model achieves.

Another limitation of AR models is that, while performing well on small data sets, fail to extract additional performance from larger data sets due to their simplicity.

References

[1] Microsoft Corporation (MSFT) stock historical prices & data (2023) Yahoo! Finance. Yahoo! Available at: https://finance.yahoo.com/quote/MSFT/history.

[2] Nielsen, A. (2020) "Understanding Stationarity," in Practical time series analysis: Prediction with statistics and machine learning. Beijing: O'Reilly.

[3] Nielsen, A. (2020) "Autoregressive Models," in Practical time series analysis: Prediction with statistics and machine learning. Beijing: O'Reilly.