MNRAS 000, 1–8 (2022) Preprint 16 April 2022 Compiled using MNRAS LaTeX style file v3.0

Time Series Forecasting at the two extremes: Predictable and Unpredictable

Alon Shapiro[1]★
[1]McGill University, 845 Sherbrooke Street W, Montréal, Quebec H3A 0G4, Canada

16 April 2022

ABSTRACT
Time series analysis and forecasting can provide vital information to an organization regarding its past performance as well as what is the best decision to take for the future. However, this is not a trivial task. A myriad of different time series models and statistical approaches have been developed, and each model might be appropriate for one type of time series, but inappropriate for another. Therefore, model selection must be given much consideration. Here, ARIMA, a popular time series model, is used to predict two very different data sets, one is highly predictable, while the other is extremely unpredictable. The model forecast is compared between the two data sets, and Bayesian Markov-Chain Monte Carlo approach is used to obtain a more statistically-conscious understanding of the model's parameters. A rolling forecast approach is discussed and compared to a longer forecast horizon approach.

Key words: Time series forecasting – ARIMA – Bayesian MCMC

1 INTRODUCTION

Time series (TS) are data sets which track a signal of some sort over time. This can be a record of the population size of a country over a decade, outside temperatures over the course of a day, how many students attend a certain university each year, or the price of a stock. Since TS can describe so many different processes, it would be advantageous if there was a way to predict its future behaviour. Imagine if an investor could accurately predict which stock prices would drop tomorrow and which ones would soar. They'd be able to invest with complete certainty. Or if a business could predict exactly the demand for each of their products, so they don't have to produce too much of some products (to avoid wasting resources), or too little of another (so they don't miss customers willing to buy their product). Therefore, predicting TS can provide huge benefits. However, not all TS are created equally. Depending on the underlying mechanism of a given TS, it can be relatively easy to make predictions, or extremely difficult. Consider outdoor temperatures in Montreal. Temperatures follow a very predictable pattern year-round which changes over very big timescales (Earth's temperature rises about 0.08 degrees C per decade, Lindsey 2021), therefore predicting a rough forecast for the next three months can be done with a relatively high accuracy. On the other hand, a TS such as the price of a stock has random fluctuations and is controlled by a lot of hidden processes. It is often very difficult to correctly predict stock prices over even short periods of a few weeks. In this work, two popular TS forecasting models (ARIMA and SARIMA, see sections 1.5) will be examined and used to forecast the temperature in Montreal as well as the stock price of the S&P 500 index to contrast the high predictability of the former against the lack of predictability of the latter, in the absence of more sophisticated forecasting methods. A Bayesian Markov-Chain Monte Carlo (MCMC) approach will be used to obtain posterior probability distributions for the stock TS model parameters, which will be used to examine the parameters produced by a model fitting which uses a non-Bayesian approach.

1.1 TS Decomposition

A time series can be decomposed into a number of underlying components which work additively (or multiplicatively) to produce the time series (see Fig. A1). The main components are the trend, seasonality, and residuals (ABS n.d.). The trend describes an overall upward or downward movement of the time series over time (over the whole lifetime of the TS or locally). It can be thought of as if the TS is 'smoothed out' to erase any local fluctuations, and only retain the bulk movement of the TS (similar to a moving average treatment). The seasonality describes the periodicity of the TS. An example of this is the periodic behaviour of outdoor temperatures throughout the year, which increase during the summer months, and drop during the winter months, with a period of 12 months. A TS might also display a complex seasonal component, that is multiple cyclical components. For example, outdoor temperatures cycle throughout the seasons of the year, but also throughout the hours of the day, with higher temperatures around noon, and lower temperatures during the night. Lastly, the residuals is the signal that remains from the TS after the trend and the seasonality have been removed. This component captures the noisiness (unpredictable part) of the TS. This can be the random variables affecting the decision of investors to sell or buy a specific stock (their mood on a particular day, for example), or something more predictable such as a peak in sales of a store during a black-Friday sale, which is highly irregular yet predictable. Depending on

★ E-mail: alon.shapiro@mail.mcgill.ca

the goal of the forecasting, the residuals of the TS can be attenuated by various smoothing methods.

1.2 Stationarity and Differencing

A stationary TS is a TS with a constant mean and variance, and no seasonality, and a constant ACF (see section 1.3, NIST, n.d., see Fig. B4 for an example of a stationary signal). Certain models (such as ARIMA, see section 1.5) require the TS to be stationary. Non-stationary TS can be made stationary by differencing. Differencing means to subtract the value of a previous time point from each given time point in order to produce a new TS, and can be described by the following equation:

$$y_t = Y_t - Y_{t-1}, \tag{1}$$

where y_t is the differenced time point, and Y_t is a time point from the original TS. Depending on the TS, more than one differencing might be required. Stationarity can be asserted by the absence of a unit root for a chosen TS model, which means that depending on the model, if some of its parameters are equal to one, the modeled series in non-stationary (Hurvich, n.d.). The presence of a unit root for a given TS can be confirmed with various statistical tests (see section 1.4).

1.3 The Autocorrelation Function and The Partial Autocorrelation Function

The autocorrelation function (ACF) is a tool to evaluate the self-similarity of a TS by comparing it to itself at different time lags (OTEXTS, n.d.). Consider a TS of monthly car sales over one year, with one value for each month, and 12 values in total. When the TS is compared to itself, each value of each month line-up perfectly, yielding a strong correlation. When the TS is compared at 1 time lag, the whole TS is 'shifted' one month over, so that now January is compare to February, February to March, March to April, and so on (y_t is compare to y_{t+1}). At 2 time lags, January is compared to March, February to April, and so on (y_t is compared to y_{t+2}). The way a given time point y_t is affected by past time points is thus quantified by the ACF, which includes indirect contributions of any $y_{\tau<t}$ through $y_\tau \rightarrow y_{\tau+1} \rightarrow ... \rightarrow y_{t-1} \rightarrow y_t$ on y_t, as well as direct contribution of y_{t-1} on y_t.

The partial ACF (PACF) similarly measures autocorrelation of a signal, but here only the direct effect of a time point $y_{\tau<t}$ on y_t is measured through $y_\tau \rightarrow y_t$ (PennState, n.d.).

Both the ACF and the PACF can be used in the analysis of a TS to estimate the type of model that would adequately fit it. These functions can also be used to analyze fitting residuals in order to estimate the goodness-of-fit (or the quality of a TS forecast) of a given model.

1.4 Augmented Dickey-Fuller test

The augmented Dickey-Fuller (ADF) test is a statistical test which can be used to assert the stationarity of a TS (Prabhakaran 2019). The null hypothesis is that there is a unit root in a TS, and thus a low p-value (<0.05) rejects the null hypothesis and concludes that the TS is stationary. Many programming languages offer easy to use ADF testing packages (such as adfuller from statsmodels for python).

1.5 The ARIMA model

The Autoregressive Integrated Moving Average (ARIMA) model is a general TS forecasting model which incorporates three different components: an autoregressive (AR) model, an integrated component (accounts for differencing required to make the TS stationary), and a moving average (MA) model (Boateng, n.d.). AR models allow for prediction of a TS based on previous points from the series itself. MA models allow for prediction of stationary TS by modelling the next data point as a combination of the mean with error terms from previous (given) points. By combining AR with MA and allowing for differencing of the TS, a much more general class of TS can be modelled (Nau, n,d,). The general form of an ARIMA model is given by (Alonso, n.d.):

$$(1 - \sum_{i=0}^{p} \phi_i B^i)(1-B)^d Y_t = (1 + \sum_{i=0}^{q} \theta_i B^i)\epsilon_t, \epsilon_t \sim N(0, \sigma^2), \tag{2}$$

where p, d, and q denote the order of the AR(p), I(d), and MA(q) models (can be denoted as ARIMA(p,d,q)), respectively, and ϵ_t is the error in time step t, which is normally-distributed with mean zero and standard deviation σ. B is the 'backward shift' operator, which acts on time point Y_t as follows:

$$B^n Y_t = Y_{t-n}. \tag{3}$$

For example, an ARIMA(1,0,1) model is given by:

$$\begin{aligned}(1 - \phi_1 B)Y_t &= (1 + \theta_1 B)\epsilon_t \\ Y_t &= \phi_1 Y_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t \\ \hat{Y}_t &= \phi_1 Y_{t-1} + \theta_1 \epsilon_{t-1},\end{aligned} \tag{4}$$

where $\hat{Y}_t$ is the model's estimation of a time point of a TS, and is given by the observed (true) time point minus the noise term $Y_t - \epsilon_t$.

1.6 ARIMA order selection via AIC

Akaike's Information Criterion (AIC) is a useful formula for selecting the order of the ARIMA model (OTEXTS, n.d.). It is given by the formula:

$$AIC = 2k - 2log(L) \tag{5}$$

where k is the number of parameters in the model and log(L) is the log-likelihood of the model, which is the probability of obtaining the data being modelled given a particular set of parameters. For example, for the ARIMA(1,0,1) model given by Eq. 4, there are 3 parameters, ϕ_1 for the AR(1) component of the model, θ_1 for the MA(1) component of the model, and σ, for the estimation of the variance of the noise terms ϵ_t. Therefore, its AIC would be $2(3) - 2log(L)$. In general, a more favourable model would contain less parameters (less complex model), and have a higher probability (lower log likelihood). Therefore, models with lower AIC values are said to be more favourable than models with higher AIC values. A 'brute force' approach for order selection can also be used. In this approach, many ARIMA models of various orders are generated and fitted to the data. The AIC values for all the models are recorded and then ranked by ascending AIC (lowest to highest). Once the order is chosen, the model can be fitted to the data, and the appropriate ARIMA mathematical formula (Eq. 2) can be used to forecast future values of the series. In the packages used in this work (statsmodel ARIMA and SARIMAX), the models are fitted to the data with a log-likelihood function, which is determined through a maximum likelihood estimation (MLE) with Kalman filter approach and is done under the hood.

1.7 SARIMA

When dealing with a TS that has a seasonal component, a seasonal ARIMA (SARIMA) model can be used. This model has all the ARIMA components, as well as another set of the same components for a seasonal-shift treatment (Pennstate, n.d.). The order of the SARIMA model can be written as: (p,d,q)x(P,D,Q,S) where (p,d,q) are the ARIMA orders, and (P,D,Q,S) are the seasonal ARIMA orders. SARIMA takes the following mathematical form (Alonso, n.d.):

$$\begin{aligned}(1-\sum_{i=0}^{p}\phi_i B^i)(1-\sum_{i=0}^{P}\Phi_i B^{iS})(1-B)^d(1-B^S)Y_t = \\ (1+\sum_{i=0}^{q}\theta_i B^i)(1+\sum_{i=0}^{Q}\Theta_i B^{iS})\epsilon_t, \\ \epsilon_t \sim N(0,\sigma^2).\end{aligned} \tag{6}$$

1.8 Forecast evaluation and the Shapiro-Wilk test

Once a model has been fitted to data and a forecast has been produced, the forecast can be evaluated by comparing it to the testing data, by looking at the residuals (difference between forecasted series and the testing data). The better the forecast is, the smaller the amplitude of the residuals will be, and the more the residuals will resemble Gaussian noise (OTEXTS, n.d.). To assess the resemblance of the residuals to a normally distributed signal, the Shapiro-Wilk (SW) test can be used (Shapiro, 1965). The null hypothesis of the SW test is that the provided distribution is normally distributed, meaning that a low p-value (<0.05) rejects the null hypothesis, and the conclusion is that the provided distribution is not normally distributed, and a high p-value (>0.05) accepts the null hypothesis, concluding the provided distribution is normally distributed. It is therefore desirable for the residuals of forecasted data to yield a high p-value for the SW test.

1.9 Rolling Forecast Approach

Stock market data will be predicted in this report for a period of 2 years. Due to the unpredictability inherent in stock prices, the ARIMA model prediction converges to the average trend found in the training data. In real-world applications of TS forecasting however, there is often interest in more reliable short-term forecast rather than long term trends. To this end, a rolling forecast approach is adopted (Hyndman 2014). In this approach, rather than predicting far into the future, a short period is predicted (typically a week up to a few months, depending on the specific TS), and then when the period passes and data is observed for this period, it is appended to the training data set, which then produces a forecast for the following period of time. A naive implementation of this approach is presented in the results.

2 DATA

2.1 Weather data

Daily weather data for Montreal between the years 2012 and 2021 was obtained from public data provided by the Government of Canada available at climare.weather.gc.ca. The data was taken from the Montreal/Pierre Elliott Trudeau INTL weather station data. The data was cleaned and treated to produce monthly averages with two separate python scripts. The final csv files were then loaded and combined in the main jupyter notebook (Time_Series_analysis_ARIMA_MCMC) to produce a Pandas Dataframe object, which was used for the TS analysis and forecasting. Only the mean temperatures were used for the analysis. This data was chosen because average temperatures remain fairly stable over the chosen period, and this TS has high predictability. This dataset is later transformed by adding a strong trend component to it, to also test forecasting for a TS with a strong trend. Original mean temperatures TS is plotted in top of Fig. 1, with training data in green, and testing data in orange. Transformed temperatures TS (by adding a linear trend) is plotted in bottom of Fig. 1.

2.2 Stock prices

Data set of daily stock prices for the S&P 500 Index for the years 1993 to 2022 was obtained using the open-source python package yfinance, which uses publicly available API's from the website Yahoo! finance. The data was loaded as a Pandas Series object. Any missing values were interpolated to assure there are no time jumps. This was necessary since the model fitting package used in this report requires no missing time points. The S&P 500 is an index which tracks the stock price of a collection of 500 of some of the largest companies in the United States. This dataset was chosen since on one hand, it is a very good indicator of the economy in the US and is very useful for economists, but on the other hand, it incorporates so many sources for noise and random fluctuations which makes it a very difficult and interesting TS to forecast. Stock prices raw data was plotted with training data in green and testing data in orange (see Fig. 2).

3 METHODS

When forecasting is done on a past data, the data set is split into to parts- the training set, which is used to choose the model and fit the model to the data, and the testing set, which the model then tries to forecast, and can be tested against to evaluate the performance of the model (how good it actually predicted the future of the TS). The scripts and jupyter notebooks used to process and analyze the data in this report are available in this GitHub repository.

3.1 Code used in this work

The entirety of the data processing, cleaning, and analysis was done in python 3, either as stand-alone python scripts, or in a Jupyter notebook. The Numpy and Pandas libraries were used to process and store data. The Statsmodels module was used extensively in all parts of the data analysis and forecasting. Data was plotted with the Matplotlib library. Bayesian MCMC technique was performed with the emcee module, and model parameters were plotted with the corner module.

3.2 Weather forecasting

The weather data was analyzed with a decomposition, ACF and PACF, and a brute force approach was used to determine its SARIMA order. The TS was split such that 2012-2019 was used as the training data for a SARIMA model, and 2019-2021 was used as a testing data to then evaluate the model fit. The residuals were examined and tested for normality with the SW test. The TS was then transformed by adding a pronounced linear trend to it to test the SARIMA model on a TS with a stronger trend component. The residuals were then analyzed to evaluate the forecasted data (see Fig. A2).

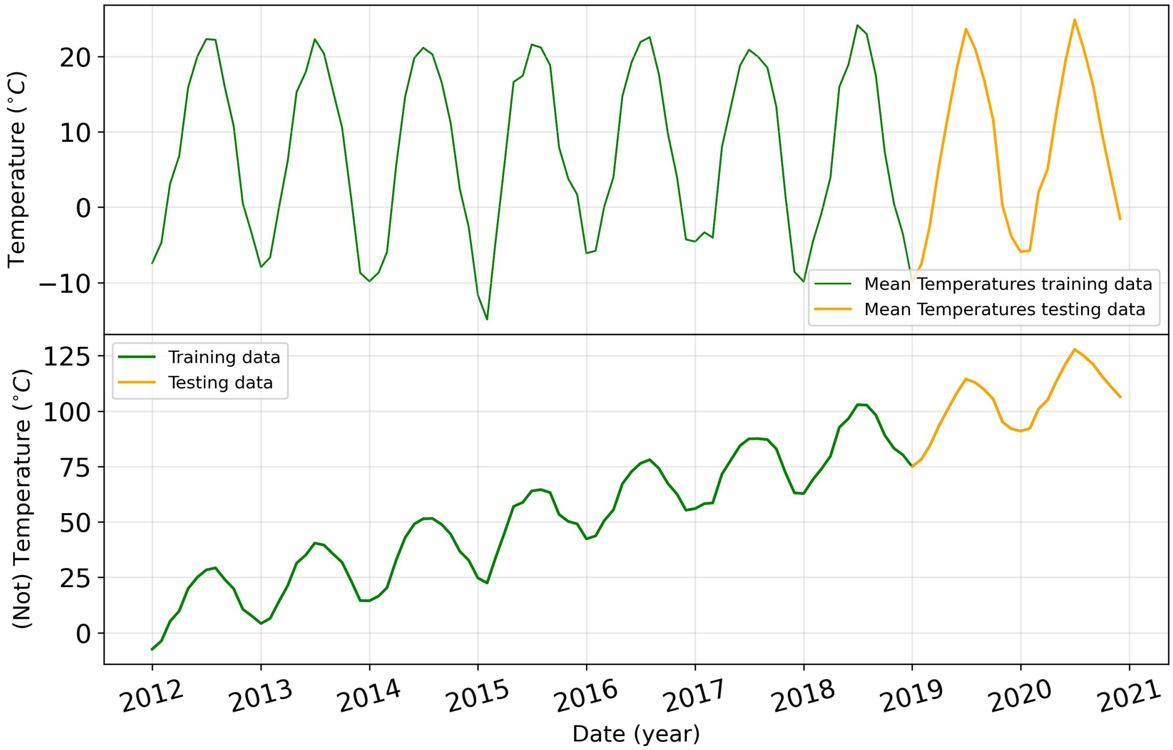

Figure 1. Top pane: Raw data of average monthly temperatures in Montreal between the years 2012 and 2021. Data is split into the training data in green from 2012 to 2019, and the testing data in orange from 2019 to 2021. Bottom pane: Transformed raw data from top pane by addition of a dominant linear trend. Data is split similarly to top pane description for training and testing.

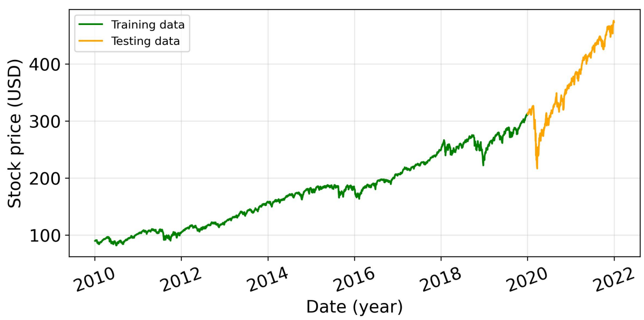

Figure 2. Raw data of S&P 500 index stock price between 2010 and 2022. Data is split for training in green from 2010 to 2020, and testing in orange from 2020 to 2022.

3.3 Stock price forecasting

The stock price data was then analyzed by similar methods. The volatility of the TS was also analyzed by looking at the residual component of the decomposition. A naive outlier detection method was devised (see Fig. B4). Only data from 2010 to 2022 was used for the model selection and fitting. Data from 2010 to 2020 was used as the training data, and data from 2020 to 2022 was used as the testing data. Brute force approach was used to determine both the ARIMA and SARIMA model orders. After investigation, an order (1,2,1) was chosen for ARIMA, and (0,1,1)x(0,1,1,7) for SARIMA. The two model forecasts and their residuals were analyzed and compared (see Fig. B6). ARIMA was chosen as the favourable model for the TS, and was then used along with the emcee python package to produce probability distributions for the model parameters using a Bayesian MCMC approach using a Metropolis-Hastings algorithm (see Fig. 6). One hundred MCMC calculated ARIMA parameter sets were then used to produce 100 forecasts, which are then compared against the non-Bayesian ARIMA forecast (see Fig. 5). Lastly, a rolling forecast procedure was used to produce forecasts using 3 rolling-window sizes of a day, a week, and a month (see Fig. B9, B10, B11). Residuals were plotted and analyzed for all 3 configurations.

4 RESULTS

4.1 Weather forecast

TS decomposition components (original series, trend, seasonal, and residual components) were plotted for original temperatures TS and trended temperatures TS (see Fig. 3, A1). Both temperature TS's were found to be non-stationary (p = 0.49 for original TS, p = 0.99 for trended TS), and required one differencing to establish stationarity($p = 2.4e-12$ after one differencing of original TS once, $p = 2.4e-12$ after once differencing of trended TS) . Seasonality for both original and trended TS's was found to be 12 using an ad hoc python function (see series_period() function in jupyter notebook, on the report's github repo). Original temperatures TS was fitted with a SARIMA(0,0,1) x (0,1,1,12) model (see Fig. 4, top pane). Trended temperatures TS was fitted with a SARIMA(1,1,1) x (0,1,1,12) model (see Fig. 4, bottom pane). Fit residuals for forecasts of both original and trended TS's were plotted (see Fig. A2, A3). The SW test was used to confirm the normality of both residual fits (p = 0.62 for residuals of original TS forecast, p = 0.74 for residuals of trended TS forecast).

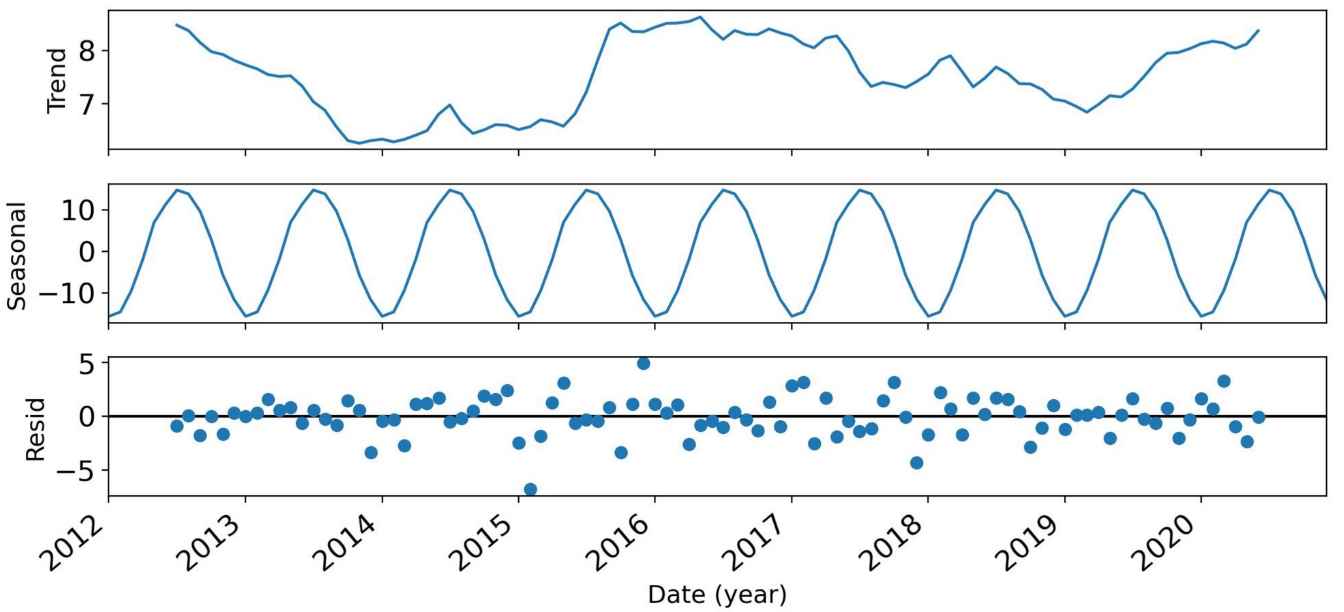

Figure 3. Time series decomposition of average monthly temperatures for Montreal between the years 2012 and 2021. Top pane shows the trend component, middle pane shows the seasonal component, and bottom pane shows the residual component.

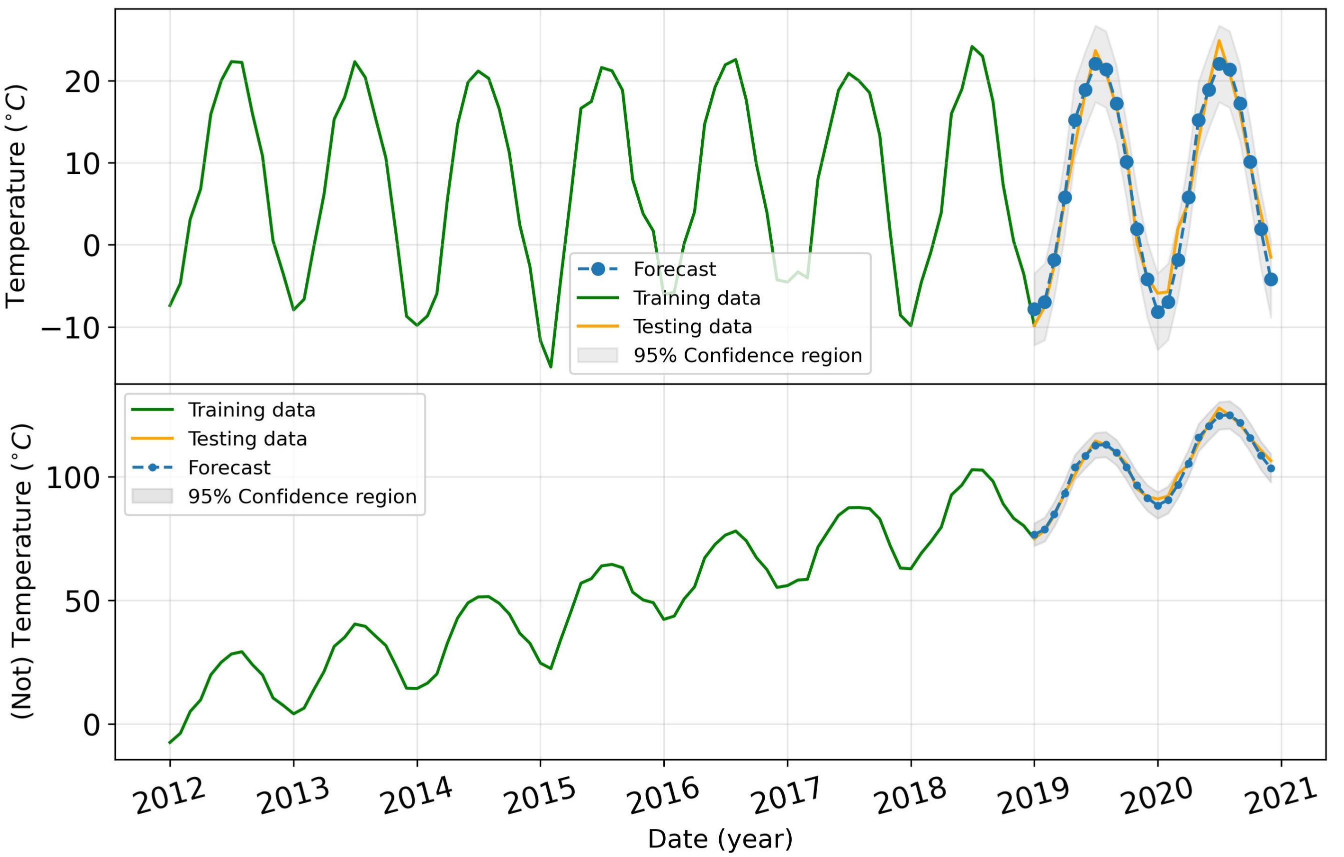

Figure 4. SARIMA model forecast for last two years. Top pane shows forecast for original monthly average temperatures. Bottom pane shows forecast for the transformed TS which includes a strong linear trend.

4.2 Stock price forecast

The TS was confirmed to be non-stationary with an ADF test (p = 1.0). One differencing step was enough to make the TS stationary by ADF test (p = 1.6e-26). The seasonality of the TS was determined by the ad hoc period finding function (series_period()) to be 7 time steps. ACF & PACF of the stock TS were plotted to determine a possible model order (See Fig. B1, B2). Decomposition of the stock TS was plotted (see Fig. B3), together with a decomposition of a subset of the TS to allow for better visualization of the seasonality component (see Fig. B5). The residual component of the TS was plotted together with the region of 3 standard deviations to visualize outliers (see Fig. B4). A brute force approach was used to determine the appropriate SARIMA order, and an automatic ARIMA order finder function was used to determine the ARIMA order. A SARIMA (0,1,1) x (0,1,1,7) model and an ARIMA (1,2,1) model were both used to forecast the stock TS. Fit residuals were plotted for both models (see Fig. B7, B8), and tested for stationarity with the SW test (p = 1.45e-11 for ARIMA model, p = 8.9e-12 for SARIMA model). Residual plots for both models were determined to be not normally distributed. A corner plot (see Fig. 6) was made to visualize the Bayesian MCMC-determined probability distributions for the three parameters of the ARIMA model, ϕ_1, θ_1, and σ, with mean values found to be 0.0565, −0.99, and 1.39, respectively. To compare the forecast of the ARIMA model with the MLE-determined parameters against the forecasts determined by the ARIMA models with the MCMC-determined parameters, a plot of the whole TS was made with all the forecasts overlaid (see Fig. 5). Lastly, a plot was made for each of the three rolling forecast windows along with their corresponding fit residuals (see Fig. B9, B10, B11).

5 DISCUSSION

Even after decomposing all the TS examined in this work and looking at the associated ACF's and PACF's, ARIMA order selection was not straight forward. Brute force approach to determine the best model based on AIC provided good estimates for the best fitting order for each TS, after which a short iterative trial-and-error approach of trying models with one order lower or higher proved to be the easiest way for determining model order.

5.1 Weather forecasting results

A strong seasonal component is present in the temperatures TS (see Fig. 3), as expected, which make the TS more predictable. An ADF test reveals that the TS is not stationary, so at least one order of seasonal differencing was used in the model. The chosen SARIMA model produces good forecast for the last 2 years (see Fig. 4), with residuals being normally distributed by the SW test with mean zero.

5.2 Trended weather forecasting results

One differencing step was enough to make the TS stationary (by ADF test), but another non-seasonal differencing yielded better forecasting results. This TS also has high predictability due to the strong trend and seasonal components. Residuals of forecast are normally distributed by SW test, which confirms the model performed well.

5.3 Stock price forecasting results

The S&P 500 Index can be used as a proxy to assess market state of the US since it includes so many big companies. High volatility due to the nature of the stock market was observed by examining the residual component of the TS (see Fig. B4). An even higher, anomalous volatility is present in the TS since the beginning of 2020, so it was expected the model will forecast the testing data very badly, since the knowledge of the COVID crash was not included in this model. However, the present model predicted the expected stock price growth in the scenario that there was no COVID crash. This forecast can be compared to how the market actually recovered from the crash.

5.4 Stock price decomposition

The decomposition (see Fig. B3, B5) revealed an unexpected seasonal component, and the residual component confirmed high volatility since the beginning of 2020. Looking at a subset of the series, it was revealed there is seasonality to the market. This can be explained by the fact that the market operates during the 5 week days and closes on the weekends, which introduces a type of cyclical forcing on the system. However, the seasonality amplitude is very small compared to the trend and the residual component, which means it might be insignificant for the modeling (i.e. not necessarily SARIMA). Looking at the residual component, outliers can be naively detected by looking at data points which lie more than 3 standard deviations away from the mean residual component value. Outliers coincide with historic economic crisis such as the Y2K panic in 2000, the 2008-2009 economic crash, and the COVID pandemic crash in the beginning of 2020.

5.5 ACF and PACF of first difference of stock data

Both peak at lag 0 (series matches itself perfectly), but all other time lags are small, insignificant, and don't follow any structure (see Fig. B1, B2). This makes ACF and PACF uninformative for this particular TS.

5.6 SARIMA vs. ARIMA for stock price modeling

Both methods produce very similar forecasts since the seasonal component of the TS is negligible compared to the overall trend and the residual component. Therefore, an ARIMA model, which includes less terms overall, is the favourable model in this case (as opposed to a SARIMA model). The forecast of both models converges on a straight line which has a slope of the average value of the differenced series. This happens since the model parameters are estimated based on the training data, but each forecasted time point relies on previous observations of the TS, be it the training data, or previously forecasted values. As the forecasting horizon increases, the forecast relies less on the training data and more on itself. With each forecasting step, the uncertainty increases, and so the forecasts converge on the mean slope of the training data.

5.7 Bayesian MCMC approach for parameter determination

Probability distributions for the three ARIMA parameters agreed with the MLE-determined values. However, those provide more knowledge about the parameters and a more informed understanding of the model. Comparing the ARIMA parameters with forecasts produced by the MCMC-generated parameters confirm this claim. Both approaches agree with each other, but the MCMC-generated forecasts reveal that some of the reasonable parameter choices produce higher forecast values compared to the ARIMA forecast, which

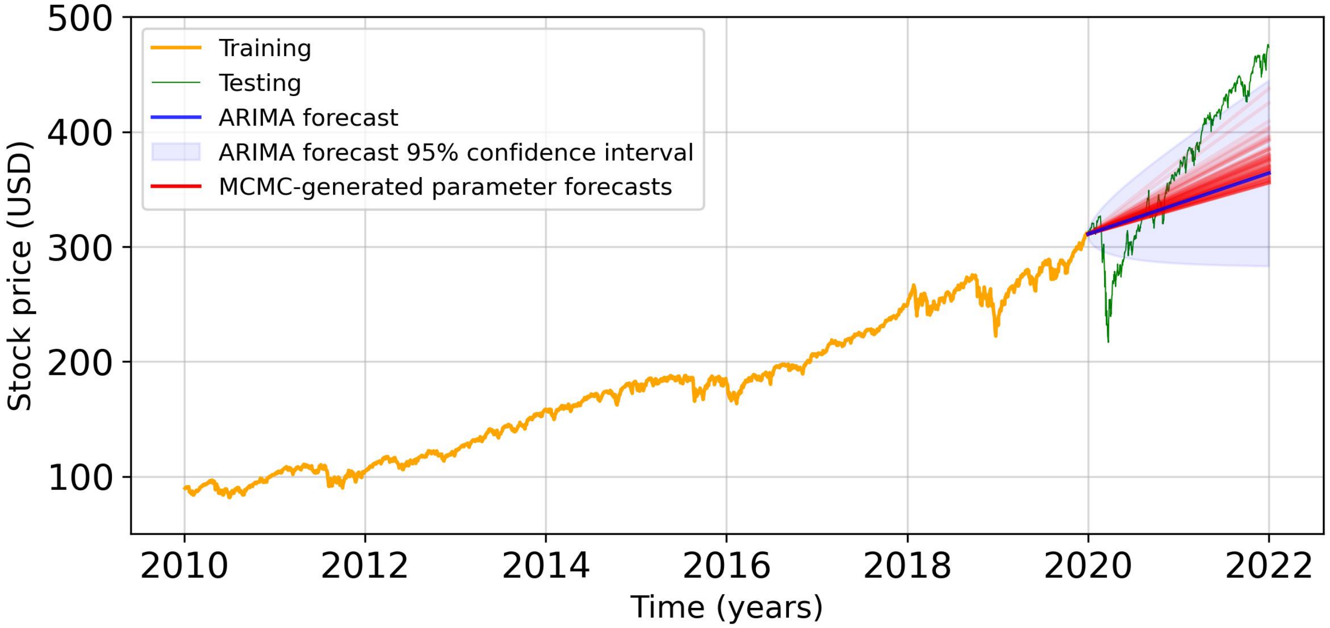

Figure 5. ARIMA model forecast shown in blue with 95% confidence interval in forecast. In red, 100 forecasts of ARIMA models which used parameters sampled from the parameter posterior distributions determined by a Bayesian MCMC method. Confidence intervals for those forecasts are not plotted. ARIMA model and models from the Bayesian MCMC method are in agreement, but Bayesian MCMC plotted forecasts provide more insights about the forecast statistics.

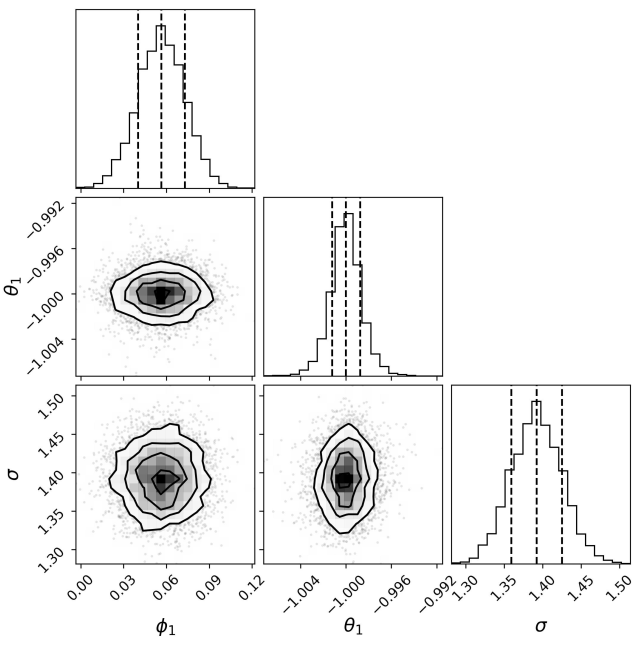

Figure 6. Posterior distributions for the three ARIMA parameters determined by a Bayesian MCMC method.

seems to prefer parameters which produce lower forecast values. Since the model is ignorant of the 2020 COVID crash, it can be used to estimate in which state the market would have been if the crash had not happened. Comparing the model forecasts to the testing data set, it appears the market recovered from the 2020 COVID crash to a better state than what it would have been predicted to be in the absence of the crash with the present (naive) model. Further insights based on these observations would require a deeper understanding of economics.

5.8 Rolling forecast method approach

Rolling forecasts appear to yield better results than one big forecasting horizon but provide a shorter-term forecast. The forecast window has to be adjusted for the specific requirements and goals involved with the forecasting of a TS. Looking at the monthly window forecasts, the forecast follows the same convergence to the mean trend which was present in the forecasting of the full horizon. A more sophisticated implementation of the RF method would include re-calibrating the forecasting model and its order at each iteration, which would yield more accurate forecasts.

6 CONCLUSION

Choosing time series model and their order is not simple nor definitive, and doesn't only depend on one parameter or another, but comes down to being the call of the person doing the forecasting, as well as to a visual assessment and prior knowledge of the system being modeled. The predictability of a given TS can be estimated with certain mathematical methods, however these were beyond the scope of this report. Nevertheless, I conclude that more predictable TS (such as the weather data used in this work) require less training data and allow for farther forecasting into the future, compared to less predictable TS (such as stock prices) which require more training data and allow for shorter reliable forecasting into the future, with uncertainty increasing rapidly. Some prior knowledge of the system is required to forecast it, but a lot more insights and knowledge about the system are acquired during the forecasting process. More informed models that would predict the stock price better would also include the opening price, and the high and low values provided by the API used in this report, but also require deeper knowledge of the dynamics of stock prices to be used correctly. Choosing a Bayesian approach provides more robust understanding of the model and its parameters and allows for better interpretation of the model parameters and the produced forecast.

DATA AVAILABILITY

The weather data used in this work is available at https://climate.weather.gc.ca/historical_data/search_historic_data_e.html. The stock prices data used in this work is available at https://ca.finance.yahoo.com/quote/%5EGSPC/history?p=%5EGSPC.

REFERENCES

ABS, Australian Bureau of Statistics, n.d., Time Series Analysis: The Basics. https://www.abs.gov.au/websitedbs/D3310114.nsf/home/Time+Series+Analysis:+The+Basics

Lindsey R., Dahlman, L. 2021, Climate Change: Global Temperature, Climate, Government of USA.https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature

NIST, US government, n.d. Stationarity. https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc442.htm

Hurvich, C., n.d. Differencing and Unit Root Tests. https://pages.stern.nyu.edu/~churvich/Forecasting/Handouts/UnitRoot.pdf

OTEXTS, n.d. Autocorrelation. https://otexts.com/fpp2/autocorrelation.html

PennState University, n.d. Partial Autocorrelation Function (PACF) . https://online.stat.psu.edu/stat510/lesson/2/2.2

Prabhakaran, S., November 2019. Augmented Dickey Fuller Test (ADF Test) – Must Read Guide. https://www.machinelearningplus.com/time-series/augmented-dickey-fuller-test/

Boateng, N., n.d.. Time Series Analysis Methods. https://rstudio-pubs-static.s3.amazonaws.com/303786_f1b99d6b7e9346c4b1488a174bab839a.html

Nau, R., n.d.. Introduction to ARIMA: nonseasonal models. https://people.duke.edu/~rnau/411arim.htm

Nau, R., 2014. The mathematical structure of ARIMA models. https://people.duke.edu/~rnau/Mathematical_structure_of_ARIMA_models--Robert_Nau.pdf

Alonso, A. M., n.d.. Seasonal ARIMA processes. http://halweb.uc3m.es/esp/Personal/personas/amalonso/esp/TSAtema6.pdf

OTEXTS, n.d.. Estimation and order selection. https://otexts.com/fpp2/arima-estimation.html

Prabhakaran, S., August 2021. ARIMA Model – Complete Guide to Time Series Forecasting in Python. https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/

Pennstate, n.d.. Seasonal ARIMA models. https://online.stat.psu.edu/stat510/lesson/4/4.1

Jones, W., December 2021. Forget ARIMA – Going Bayesian with Time Series Analysis. https://www.embecosm.com/2021/12/18/forget-arima-going-bayesian-with-time-series-analysis/

Nau, R., n.d. General seasonal ARIMA models: (0,1,1)x(0,1,1) etc.. https://people.duke.edu/~rnau/seasarim.htm

Shapiro, S. S.; Wilk, M. B. (1965). "An analysis of variance test for normality (complete samples)". Biometrika. 52 (3–4): 591–611. doi:10.1093/biomet/52.3-4.591. JSTOR 2333709. MR 0205384. p. 593

Hyndman, R. J., July 2014. Variations on rolling forecasts. https://robjhyndman.com/hyndsight/rolling-forecasts/

Larsen, K., April 2016. Sorry ARIMA, but I'm Going Bayesian. https://multithreaded.stitchfix.com/blog/2016/04/21/forget-arima/

OTEXTS, n.d., Evaluating forecast accuracy. https://otexts.com/fpp2/accuracy.html

APPENDIX A: WEATHER DATA EXTRA FIGURES

APPENDIX B: STOCK PRICE DATA EXTRA FIGURES

This paper has been typeset from a TEX/LATEX file prepared by the author.

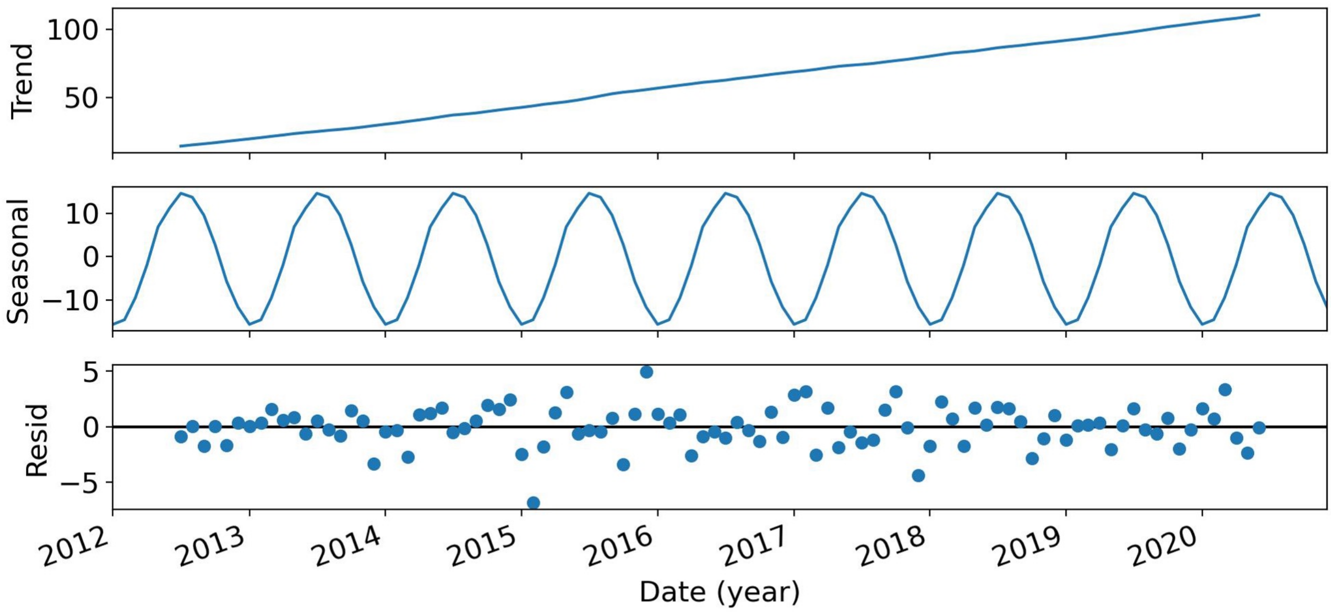

Figure A1. TS decomposition of the trended weather data. A strong trend and seasonal components compared to the residual component make this a highly predictable TS.

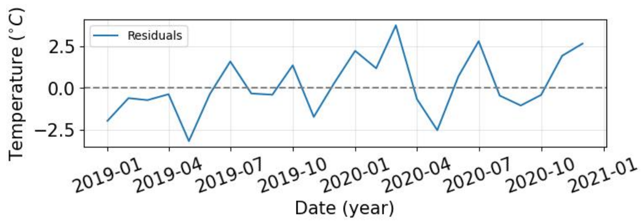

Figure A2. Forecast residuals for the monthly average temperature ARIMA model forecast. The residuals are normally distributed and have a low amplitude, indicating the model forecast fits the testing data well.

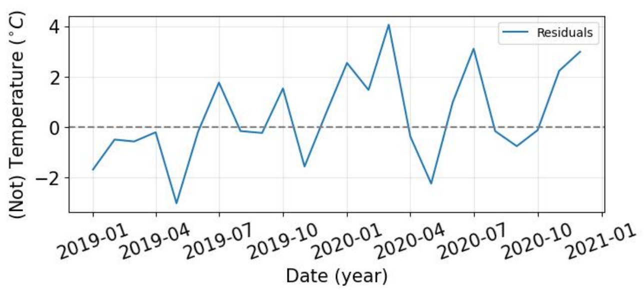

Figure A3. Forecast residuals for the transformed monthly average temperature TS. The residuals are normally distributed and have a low amplitude, indicating the model forecast fits the testing data well.

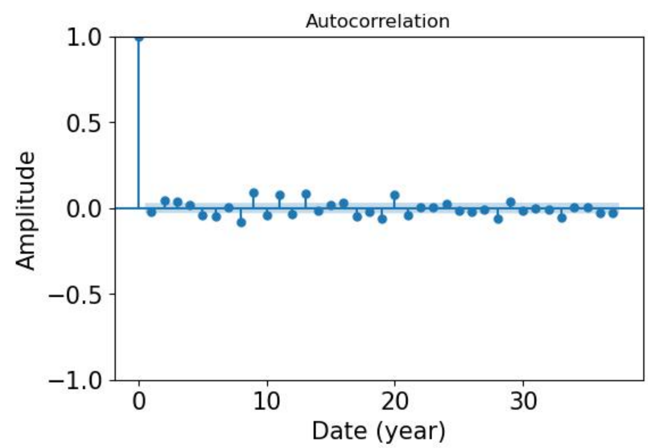

Figure B1. ACF of the first difference of the stock price TS. Strong peak at lag zero followed by unstructured insignificant lag amplitudes means the ACF is not informative.

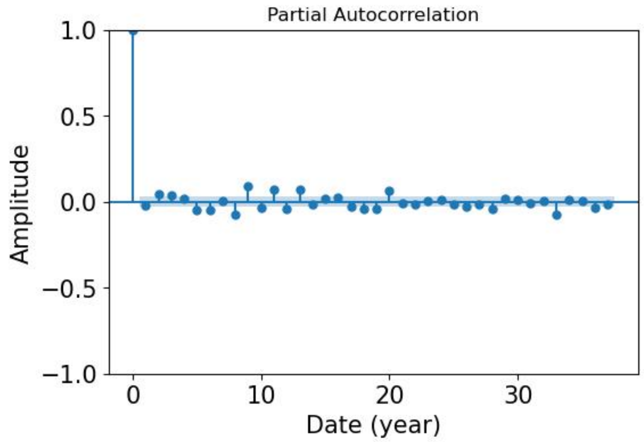

Figure B2. PACF of the first difference of the stock price TS. Strong peak at lag zero followed by unstructured insignificant lag amplitudes means the PACF is not very informative.

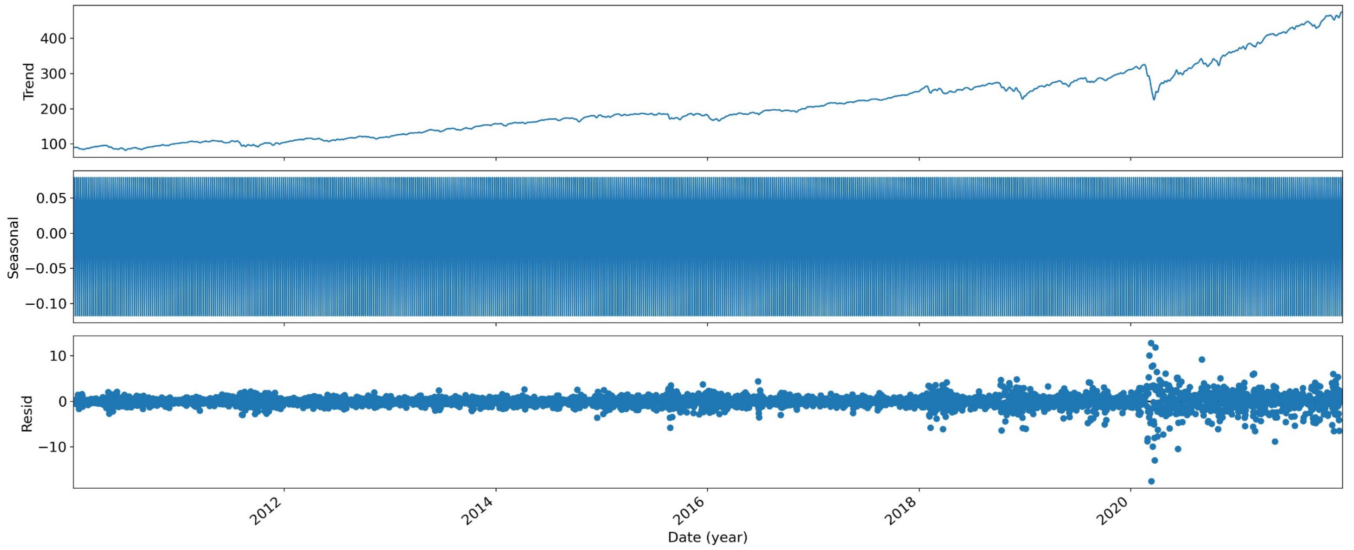

Figure B3. TS decomposition of the stock price TS. Seasonal component looks interesting, but is too difficult to be observed due to the scale of the figure.

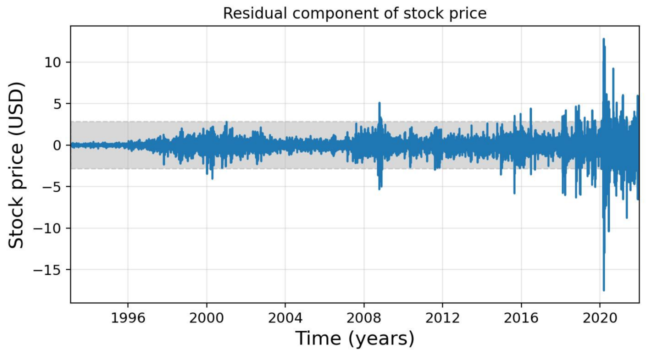

Figure B4. Residual component of the stock price TS, with 3-standard deviations from mean region shaded. Points falling outside the shaded region are outliers, and line up with historical economic events such as the Y2K panic, the 2008 economic crash, and the economic crash in early 2020 due to COVID.

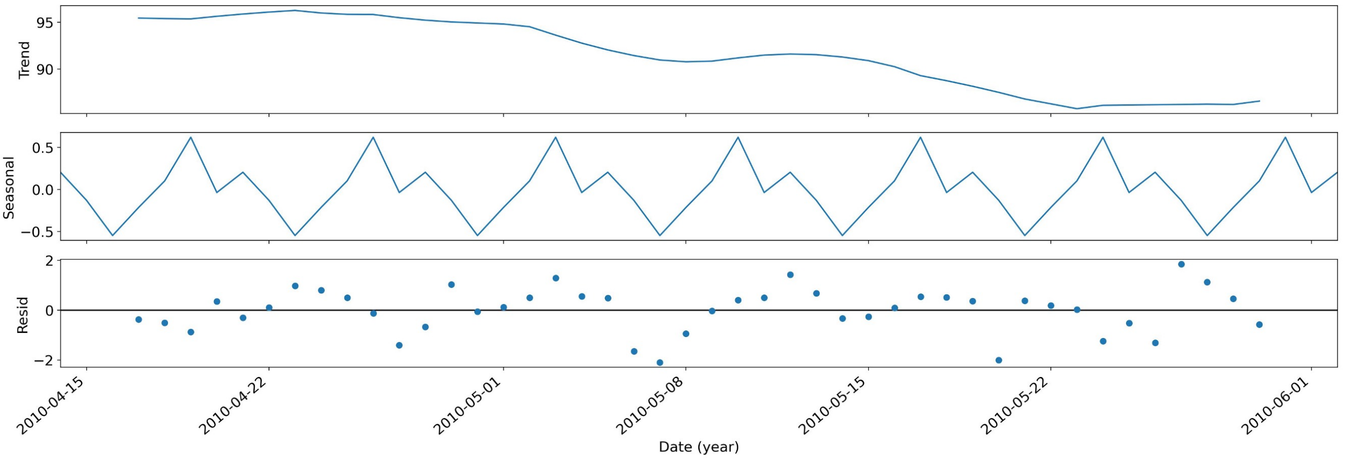

Figure B5. TS decomposition of a subset of the stock price TS. The features of the seasonal component can now be observed. The TS has a seasonality of 7 time points, which is the same as the number of days in a week.

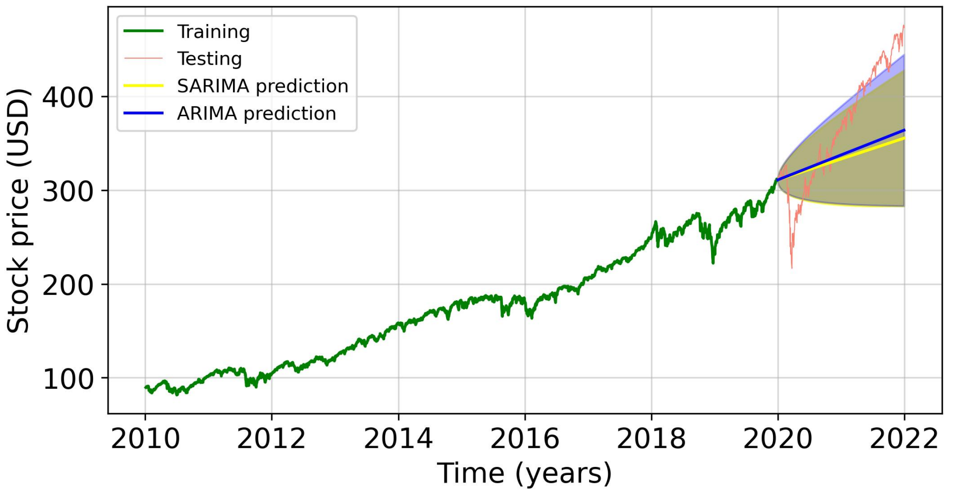

Figure B6. Comparison of the forecasts from the ARIMA and SARIMA models. Both appear to be in agreement with each other, with an almost complete overlap of their 95% confidence intervals. Since SARIMA contains more terms in its mathematical expression, ARIMA is the favoured model in this case.

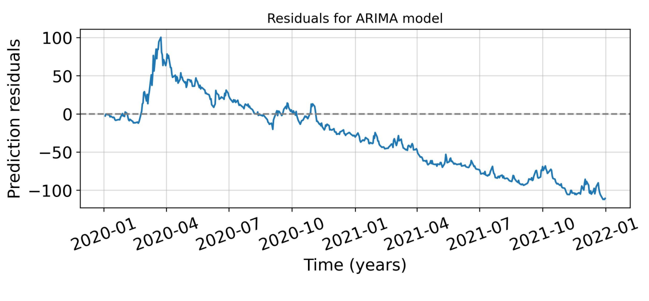

Figure B7. Fitting residuals of the ARIMA model forecast. Residuals have a structure and do not fluctuate around zero, indicating the model is not very accurate.

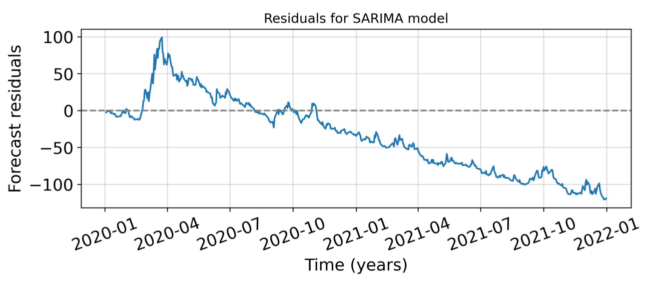

Figure B8. Fitting residuals of the SARIMA model forecast. Residuals have a structure and do not fluctuate around zero, indicating the model is not very accurate.

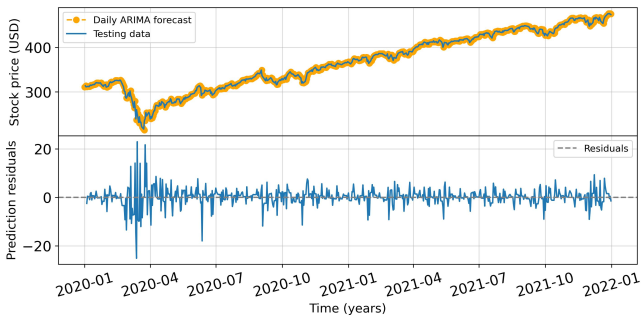

Figure B9. ARIMA forecast against testing data for the stock price TS with a daily rolling window. Due to the tiny forecast window and the large time scale included, the forecast appears to be performing very well.

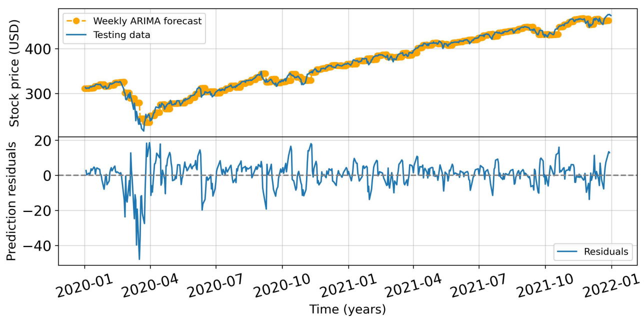

Figure B10. ARIMA forecast against testing data for the stock price TS with a weekly rolling window. Due to the small forecast window and the large time scale included, the forecast appears to be performing well. However, some drifts due to convergence on mean slope can be observed.

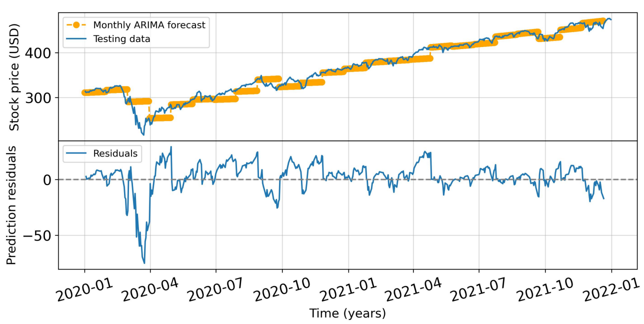

Figure B11. ARIMA forecast against testing data for the stock price TS with a monthly rolling window. Due to the big forecasting window, the data clearly seems to be performing bad. This is due in part to the fact that only one ARIMA model was used, whereas a more sophisticated approach would be to select a new ARIMA model at each forecasting step.