UNIVERSITY OF ILLINOIS AT CHICAGO
IE 594

**Time Series Analysis and Forecasting**

---

**Francesco Donato & Niko Stepniak**

# Final Project

---

Prof. Lin Li
T.A. Muyue Han

CHICAGO, ILLINOIS, AUGUST 2023

# Contents

**1 Introduction** **3**

1.1 The Data: U.S. Dollars to Euro Spot Exchange Rate . . . 3

1.2 Modeling Procedure . . . 3

1.3 Paper Outline . . . 3

**2 Raw Data Analysis** **4**

**3 Modeling Procedure** **5**

3.1 Deterministic Trend: 1st Attempt . . . 5

3.2 Deterministic Trend: 2nd Attempt . . . 6

3.3 Stochastic Trend . . . 9

3.4 Joint Optimization and Characteristic Roots Evaluation . . . 9

**4 Forecasting** **11**

**5 Conclusion** **13**

# 1 Introduction

## 1.1 The Data: U.S. Dollars to Euro Spot Exchange Rate

This report contains our findings from the fitting of U.S. Dollars to Euro Spot Exchange Rate (DEXUSEU) monthly data into a jointly optimized ARMA model. This data was obtained from the Federal Reserve Bank of St. Louis and spans the time period of January 1999 to October 2019, for a total of 250 data points. 90% of this data (225 points) was treated as a training set while the remainder was saved to compare to our forecast for the same period. We considered this data to be ideal for our modeling procedure due to its assumed presence of deterministic trends, which we aimed to identify and remove before fitting the stationary data to an ARMA model.

## 1.2 Modeling Procedure

Our ARMA modeling procedure is based on the information provided in *Time Series and System Analysis With Applications* by S.M. Pandit, as well as knowledge gained from the course. In short, we remove seasonal and deterministic trends from the data, calculate residuals from the de-trended data, identify the correct ARMA model for the stationary data based on the procedure in Fig. 1, and jointly optimize the integrated model defined by the ARMA, deterministic, and seasonal trends. Lastly, we check the effectiveness of our model by comparing the forecast of our jointly optimized model to the final 10% of the data set.

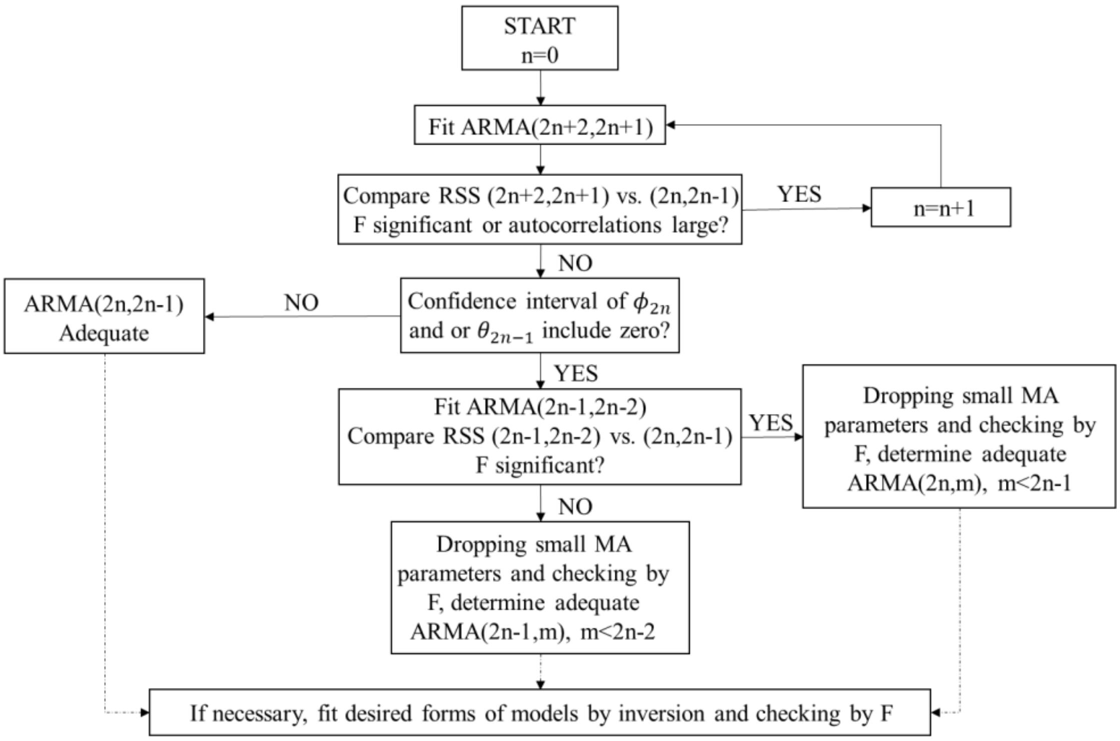



Figure 1: The ARMA modeling procedure.

## 1.3 Paper Outline

There are a total of five sections in this report, including the Introduction. Section 2 provides a brief analysis of the raw data through conversion into a log of the data and STL decomposition of it. Section 3 details the entire modeling procedure of removing deterministic and seasonal trends, fitting the stochastic trend, and joint optimization of the model. In section 4 we discuss about the forecasting results, and section 5 is a conclusion of our findings.

## 2 Raw Data Analysis

In order to give an initial visualization of our data we decided to plot the training part of our raw data as well as the whole data set. For both cases, also the log of the data is illustrated. As mentioned before the data represents the monthly U.S. Dollars to Euro Spot Exchange Rate (DEXUSEU) from January 1999 to October 2019.

Fig. 2 shows the plots described.

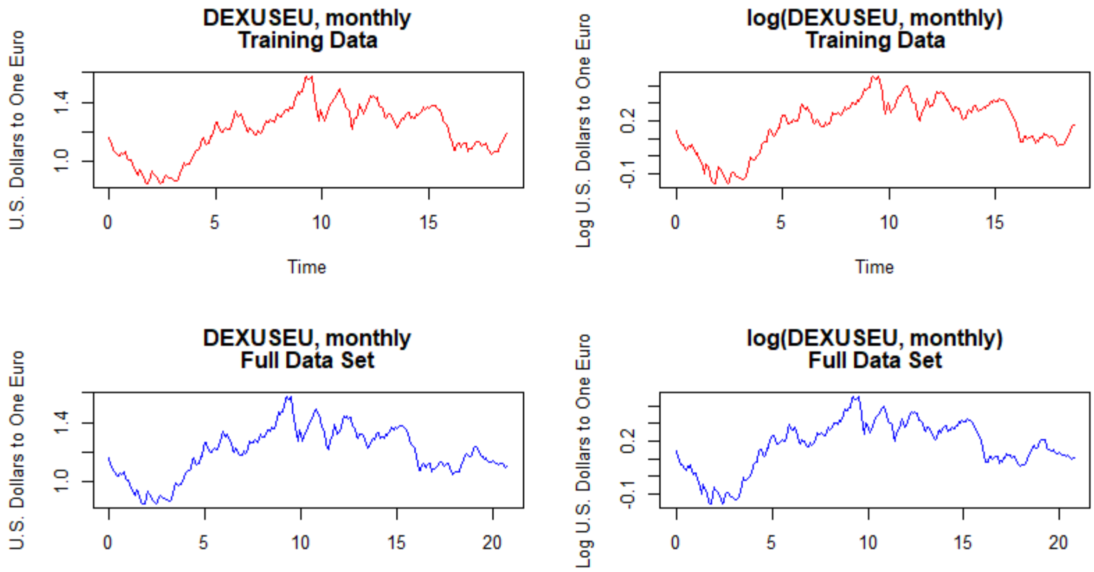



Figure 2: Raw data plots.

We then decided to perform an STL (Seasonal and Trend decomposition using Loess) decomposition as a method for estimating non-linear relationships. It uses locally fitted regression models to decompose a time series into trend, seasonal, and remainder components.

Fig. 3 shows the results obtained, where the seasonal component is calculated first and removed to calculate the trend component, while the remainder is calculated by subtracting the seasonal and trend components from the time series.

The percentage of variance in each (seasonal, trend and remainder) is respectively 3.8%, 93.3% and 15.8%.

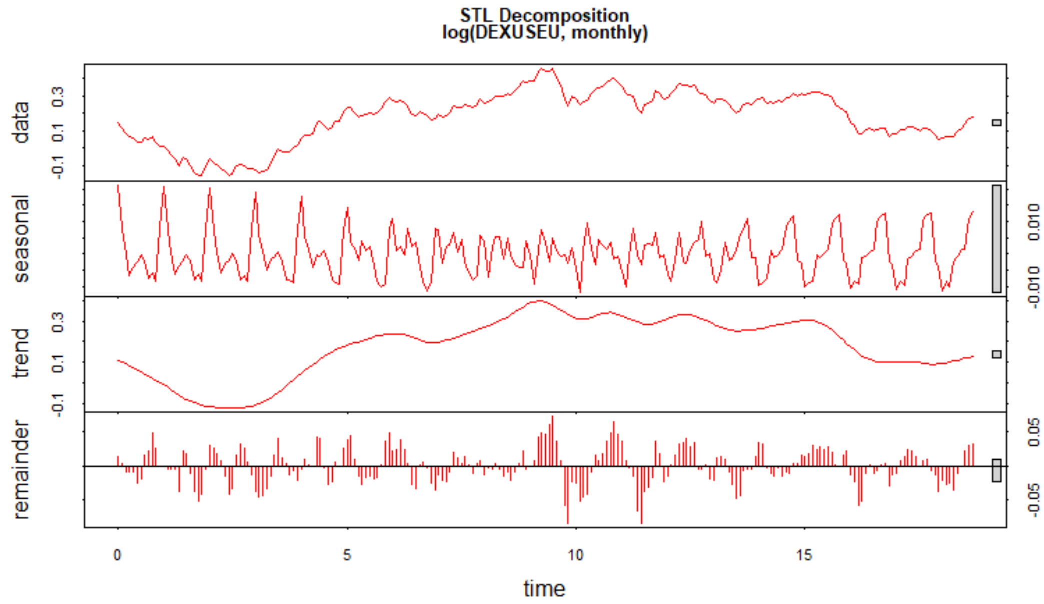



Figure 3: STL decomposition.

# 3 Modeling Procedure

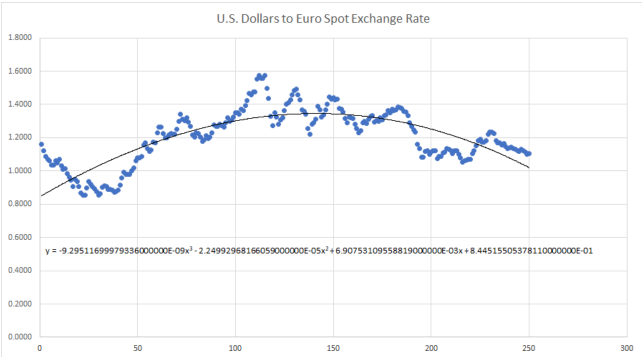
In order to build our model and further perform the forecasting and prediction, it is necessary to follow these steps:

1. Identification and removal of the deterministic trend (representing the mean of the series) as well as periodicity;
2. Evaluation of the de-trended data residuals;
3. Identification of the adequate ARMA model through the modeling procedure learned during the course;
4. Joint optimization of the parameters present in the integrated model defined through the previous steps.

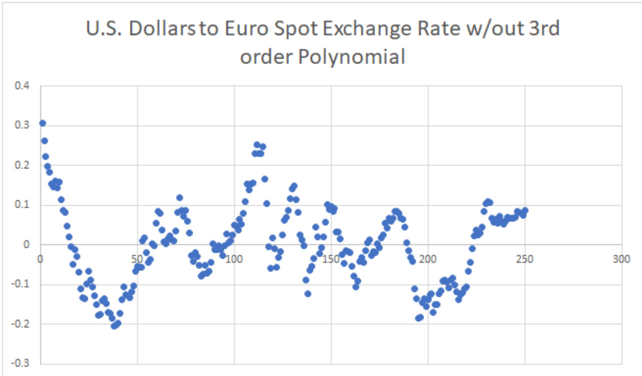
## 3.1 Deterministic Trend: 1st Attempt

In order to find the best deterministic trend fitting our data, we first used Excel to fit the data to exponential, linear, logarithmic, power, and polynomial trends, of which only the latter trend was observed in the data. We then fit the data to polynomials of orders 1 to 6 using Excel's fitting function and performed the F-test on these polynomial fits. From this, we found that the best fit for the data was a 3rd order polynomial, which can be seen in Fig. 4a. We removed this trend from our data by subtracting values obtained from this fit from our original data into a new Excel sheet. This new de-trended data is seen in Fig. 4b.

The new data we obtained in Fig. 4b appears to have a periodic trend. To remove this seasonality, we first attempted to fit this model to a sinusoidal equation using the built-in capabilities of Desmos graphing calculator by fitting the data to a function modeled by the equation $y_1 = a \cdot sin(b(x_1 - c)) + d$. The program automatically fits this model to the function with values that result in the lowest $R^2$ value, findings of which are summarized in Fig. 5a and the trend itself is observed in Fig. 5b. We subtracted this trend from the data and graphed the results in Fig. 5c, which we considered to be our stationary data.

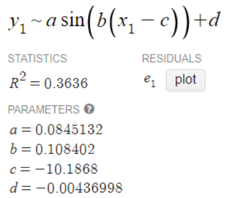



(a) Fitting of the 3rd order polynomial to the data in Excel.

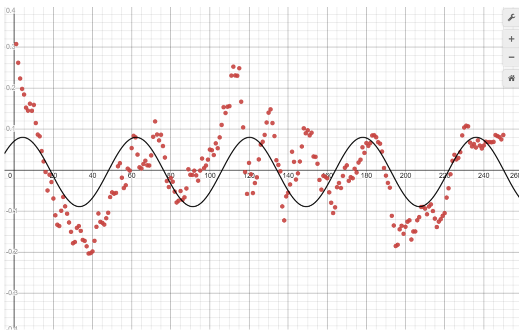



(b) Data with Excel-fitted 3rd order polynomial removed.

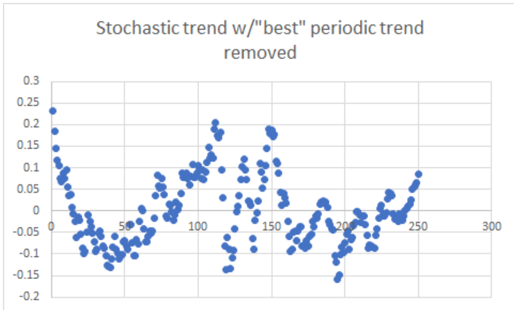
Figure 4: Plots of the removal of the polynomial trend with Excel.




(a) Values of the sinusoidal fitting in Desmos.



(b) Graph of the Desmos sinusoidal fitting.




(c) Stationary data obtained by removing the deterministic and seasonal trends obtained through Excel and Desmos.

Figure 5: Plots of the removal of the seasonal trend with Desmos and Excel.

However, we observed some issues with our stationary data and process, which led us to pursue a different procedure based on the R code and report provided to us from Alex King and Brandon Van Hecke's project. First, we believed that the polynomial fit we found with Excel could be improved. Second, our sinusoidal model has a periodicity of roughly 57 months, which we considered to be unfit for our data as we expected seasonal models based on 3,4,6, and 12 month trends. We also considered the possibility that this periodicity is close to 60 months (5 years), but decided that it was unlikely this correlation was more than a coincidence. Additionally, we decided to pursue a different procedure because we realized that our current one lacks certain components of the process which we learned in the course.

## 3.2 Deterministic Trend: 2nd Attempt

Despite the shortcomings of our initial approach, we understood that a polynomial trend would be the ideal fit for our data. Using the R code, we fit the data and performed the F-test procedure by starting from the lowest order polynomial and comparing it with the next higher order one. We iterated this process by computing at each step the F-static value and the p-value coming from each comparison (see Tab. 1). Due to the periodic nature characteristic of this data set as well as its known seasonality, we also included sine and cosine functions in our trend fitting analysis. The seasonality in our data set will be later confirmed by the calculation of the characteristic roots which, as we will show, are complex (see end of section 3.4).

As shown in Tab. 1, we stopped increasing the order of the polynomial when the p-value of the F-test was

| Polynomial Order | F-Static | p-Value |
|---|---|---|
| 1 | - | - |
| 2 | 243.10 | 0.000 |
| 3 | 89.80 | 0.000 |
| 4 | 122.19 | 0.000 |
| 5 | 82.66 | 0.000 |
| 6 | 7.34 | 0.007 |
| 7 | 7.92 | 0.005 |
| 8 | 15.45 | 0.000 |
| **9** | 11.83 | 0.000 |
| 10 | 0.38 | **0.534** |

Table 1: F-test.

higher than 0.05, hence 5%. In our case the F-test resulted to be significant up to the comparison of the 9th order polynomial with the 10th order one, which gave a p-value equal to 0.534, identifying the 9th order polynomial to be the best fit in terms of deterministic trend.

The polynomial trend follows the formula:

$$y_t = \sum_{l=0}^{N} \alpha_l * \mathrm{t}^{\,l} + \sum_{\mathrm{j}=1}^{4} \left\{ \delta_{0,\mathrm{j}} \sin\left(\frac{2\pi * t}{12} * j\right) + \delta_{1,\mathrm{j}} \cos\left(\frac{2\pi * t}{12} * j\right) \right\},$$

where t represents time in the data.

In order to attempt a parsimonious approach, we additionally tried to remove each order of the polynomial included in the fitting individually in order to check if a smaller model was adequate in defining the trend. By performing this elimination procedure and evaluating the p-value after each removal, we finally found significant to eliminate the 3rd order term from the polynomial.

Fig. 6 shows the results obtained with the final fitting results for our trend.

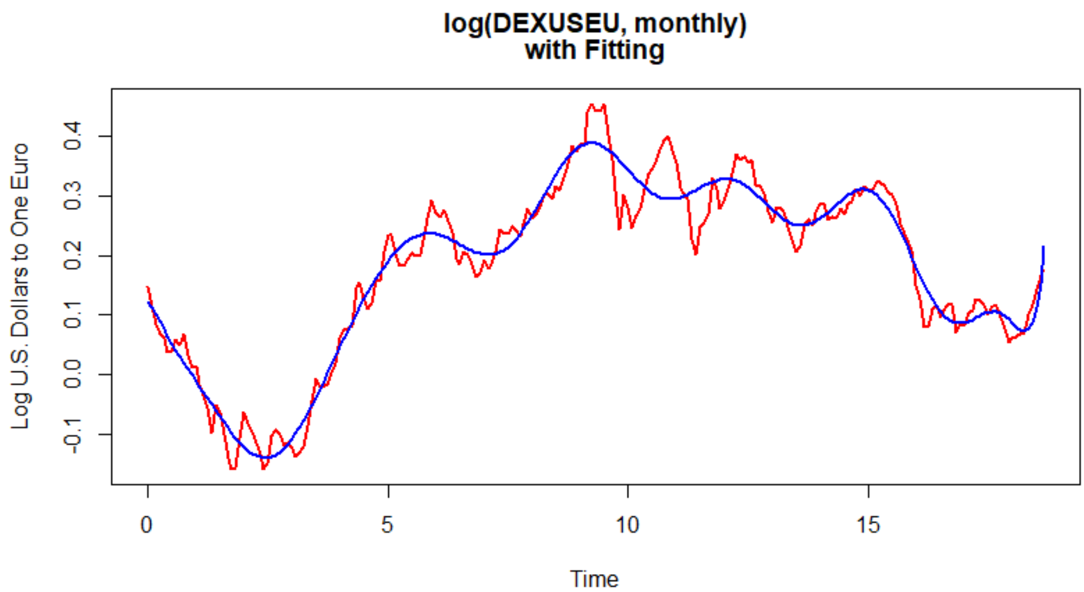



Figure 6: Curve fitting.

After defining the best polynomial fit, the next step consisted in the subtraction of this trend from our data in order to obtain stationary data representing the trend adjusted residuals (see Fig. 7).

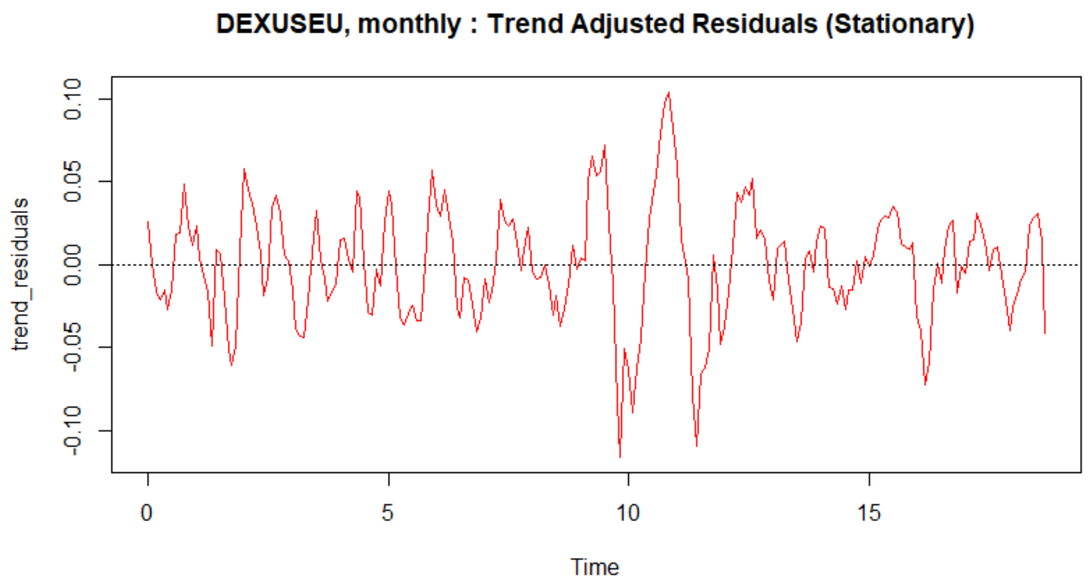



Figure 7: Stationary data.

Through the stationary data obtained it is possible to compute the autocorrelation and partial autocorrelation of the residuals. We decided to take this step in order to have a general understanding of how complex the ARMA model would be. These graphs are shown in Fig. 8, displaying an evident autocorrelation. By observing the partial autocorrelation it appears that the ARMA model will be quite small as only the first two lags look to be significant compared to the remaining ones.

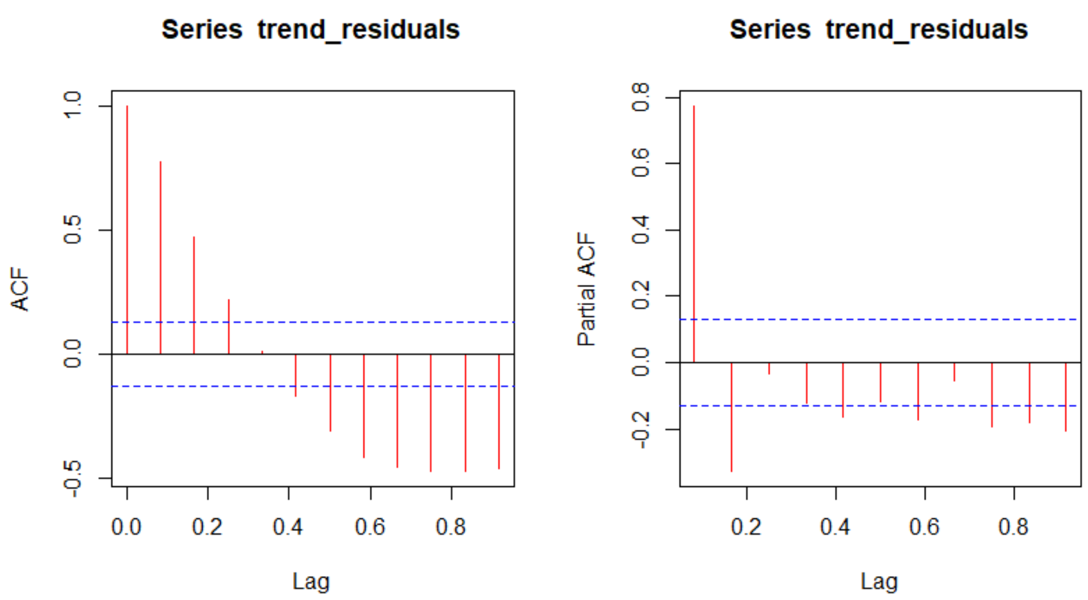



Figure 8: Autocorrelation and partial autocorrelation functions.

### 3.3 Stochastic Trend

After having identified the deterministic trend, we worked on the stationary data in order to define the stochastic ARMA part of our model. We started with an ARMA(2n, 2n-1) model with n=1 and followed the modeling procedure learned during the course to get the results shown in Tab. 2.

Before performing this procedure we compared the ARMA(2,1) model with the AR(1) and obtained a p-value equal to 0, showing significance in the comparison, hence defining the ARMA(2,1) as the better model.

Following the modeling procedure we then compared the ARMA(2,1) with the ARMA(4,3) and obtained a p-value equal to 100%, meaning that in the comparison the F-test was not significant. As a consequence we proceeded in the diagram shown in Fig. 1 and checked if the confidence interval of $\phi_{2n}$ and/or $\theta_{2n-1}$ included zero. As the first of the two intervals included it we then proceeded in comparing the ARMA(2,1) with the AR(1), knowing that the first one would win and finally compared the ARMA(2,1) with the AR(2) model as we dropped the MA parameter. Through this final comparison the F-test proved to be significant, hence the ARMA(2,1) would be our final adequate model.

| **Parameters** | **ARMA(2,1)** | **ARMA(4,3)** | **AR(2)** |
|---|---|---|---|
| Intercept | 0.00 (0.0002) | 0.0002 (0.0003) | -0.0002 (0.0045) |
| $\phi_1$ | 1.7116 (0.0353) | 0.3807 (0.1984) | 1.0618 (0.0631) |
| $\phi_2$ | -0.8391 (0.0352) | 1.0347 (0.1553) | -0.3594 (0.0630) |
| $\phi_3$ | - | -0.5072 (0.1673) | - |
| $\phi_4$ | - | -0.2553 (0.1691) | - |
| $\theta_1$ | -1.000 (0.0113) | 0.5297 (0.1806) | - |
| $\theta_2$ | - | -0.9274 (0.0288) | - |
| $\theta_3$ | - | -0.5320 (0.1653) | - |
| RSS | 0.0732 | 0.0743 | 0.0902 |
| N | 225 | 225 | 225 |
| r | 4 | 8 | 3 |
| F-static | 45.46 | 0.807 | 51.24 |
| p-value | 0.00% | 100% | 0.00% |

Table 2: ARMA modeling procedure.

Values in the columns are structured in this way: Coefficient Estimate (Standard Error).

### 3.4 Joint Optimization and Characteristic Roots Evaluation

After finding the best deterministic trend and the appropriate ARMA model it was possible to jointly optimize the integrated model, which is the following:

$$y_t = \sum_{l=0}^{9} \alpha_l * \mathrm{t}^{\,l} + \sum_{\mathrm{j}=1}^{4} \left\{ \delta_{0,\mathrm{j}} \sin\left(\frac{2\pi * t}{12} * j\right) + \delta_{1,\mathrm{j}} \cos\left(\frac{2\pi * t}{12} * j\right) \right\} + \phi_1 x_{t-1} + \phi_2 x_{t-2} - \theta_1 a_{t-1} + a_t,$$

$$x_t = y_t - \sum_{l=0}^{9} \alpha_l * \mathrm{t}^{\,l} + \sum_{\mathrm{j}=1}^{4} \left\{ \delta_{0,\mathrm{j}} \sin\left(\frac{2\pi * t}{12} * j\right) + \delta_{1,\mathrm{j}} \cos\left(\frac{2\pi * t}{12} * j\right) \right\}$$

Through the joint optimization procedure we were able to obtain the following optimized parameters, seen in Tabs. 3 and 4:

| Parameters | Joint Estimate | Std. Error | p-value |
|---|---|---|---|
| $\phi_1$ | 1.7564 | 0.0303 | 0.0000 |
| $\phi_2$ | -0.8624 | 0.0259 | 0.0000 |
| $\theta_1$ | -0.9999 | 0.0112 | 0.0000 |
| $\alpha_0$ | 5.1905 | 0.4261 | 0.0000 |
| $\alpha_1$ | -0.1940 | 0.0530 | 0.0006 |
| $\alpha_2$ | -0.8307 | 0.0816 | 0.0000 |
| $\alpha_4$ | 0.0385 | 0.0036 | 0.0000 |
| $\alpha_5$ | -0.0050 | 0.0004 | 0.0000 |
| $\alpha_6$ | 0.0002 | 0.0000 | 0.0000 |
| $\alpha_7$ | 0.0000 | 0.0000 | 0.0000 |
| $\delta_{0,1}$ | 0.6786 | 0.0762 | 0.0000 |
| $\delta_{1,1}$ | -5.1019 | 0.4417 | 0.0000 |
| $\delta_{0,2}$ | -0.1347 | 0.0085 | 0.0000 |
| $\delta_{1,2}$ | -0.0260 | 0.0035 | 0.0000 |
| $\delta_{0,3}$ | 0.0267 | 0.0029 | 0.0000 |
| $\delta_{1,3}$ | 0.0115 | 0.0038 | 0.0021 |
| $\delta_{0,4}$ | -0.0036 | 0.0038 | 0.2575 |
| $\delta_{1,4}$ | 0.0269 | 0.0039 | 0.0000 |

Table 3: Jointly optimized parameters from integrated model with ARMA(2,1).

As it will be later explained in the forecasting section, we decided to compare the results obtained from the integrated models of the ARMA(2,1) and AR(2).

In order to do this, we ran two different instances of the same code, by just modifying the line related to the fitting of the chosen ARMA model in the integrated one.

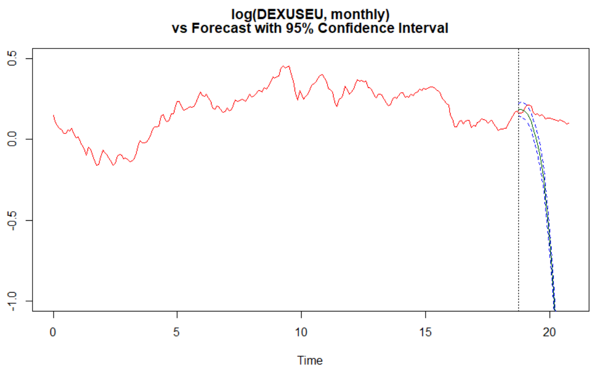
| Parameters | Joint Estimate | Std. Error | p-value |
|---|---|---|---|
| $\phi_1$ | 1.091 | 0.055 | 0.0000 |
| $\phi_2$ | -0.347 | 0.0293 | 0.0000 |
| $\alpha_0$ | 3.737 | 1.1568 | 0.0024 |
| $\alpha_1$ | -0.165 | 0.1196 | 0.1528 |
| $\alpha_2$ | -0.575 | 0.2032 | 0.0077 |
| $\alpha_4$ | 0.0265 | 0.0092 | 0.0068 |
| $\alpha_5$ | -0.0034 | 0.0012 | 0.0075 |
| $\alpha_6$ | 0.0001 | 0.0000 | 0.0087 |
| $\alpha_7$ | 0.0000 | 0.0000 | 0.0091 |
| $\delta_{0,1}$ | 0.398 | 0.223 | 0.0816 |
| $\delta_{1,1}$ | -3.623 | 1.166 | 0.0035 |
| $\delta_{0,2}$ | -0.109 | 0.025 | 0.0000 |
| $\delta_{1,2}$ | -0.019 | 0.013 | 0.1361 |
| $\delta_{0,3}$ | 0.028 | 0.008 | 0.0032 |
| $\delta_{1,3}$ | 0.005 | 0.009 | 0.3850 |
| $\delta_{0,4}$ | -0.004 | 0.008 | 0.3390 |
| $\delta_{1,4}$ | 0.025 | 0.008 | 0.0024 |

Table 4: Jointly optimized parameters from integrated model with AR(2).

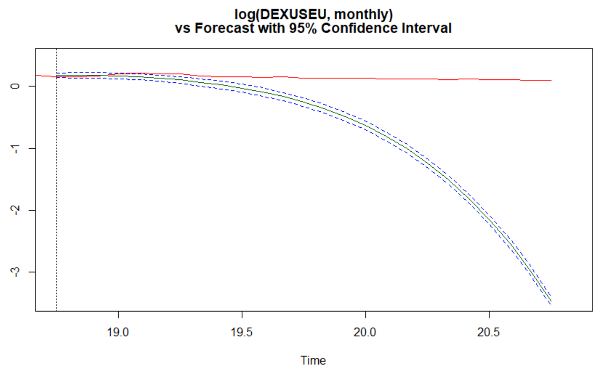
Finally, by exploiting an R function we managed to calculate the characteristic roots from the parameters of our ARMA(2,1) model after the joint optimization, $\phi_1$ and $\phi_2$, obtaining the result $1.018 \pm 0.350i$. These complex roots indicate that in our model we have stochastic seasonality.

# 4 Forecasting

Our model performed very well in recreating the trend of the original training data, as can be seen from Fig. 9c. Despite this, the forecast of the last 10% of the data failed at effectively predicting the data past a relatively short period. As can be seen in Fig. 9a, the forecast (blue) predicts a steep downward descent of the data which does not occur in reality. The dashed vertical line represents the cutoff between training and test data, and the dashed blue lines are the 95% confidence interval for the forecasted values based on the RSS value of the joint estimate. A plot of this data solely focused on the forecast portion is available in Fig. 9b.




(a) Log of test data vs the forecast.

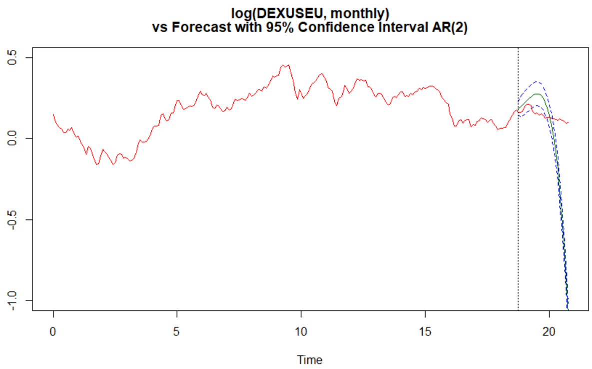



(b) Results from Fig. 9a zoomed in on the test data only.

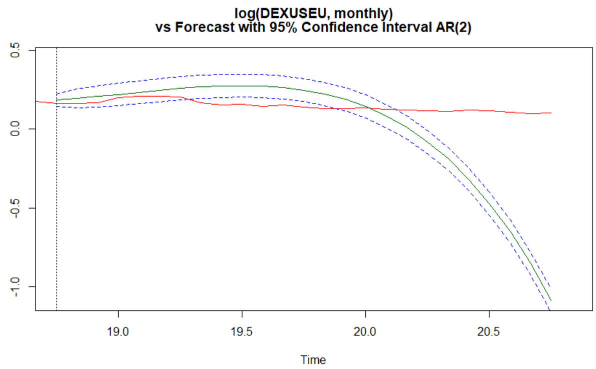

(c) Original training data (red) vs our model (blue).

Figure 9: Forecasting plots of the ARMA(2,1) model.

Since our integrated model fit works well to recreate the training data (Fig. 9c), it would have been fair to make the assumption that our model can predict the DEXUSEU rate fairly accurately a short period in the future. However, our model appears to provide a reasonable prediction of the exchange rate for less than a quarter of the test data. Because of these lackluster results in the forecast, we decided to test the forecast with an AR(2) model rather than the ARMA(2,1). In our modeling procedure we found that the ARMA(2,1) model is most significant via the F-test, but the results of the AR(2) model looked better based on an initial assessment, which is why we are including them as a comparison. The results from the AR(2) forecast were more encouraging as the model reasonably predicts the test data for over half the test period (Fig. 10).




(a) Log of test data vs the AR(2) forecast.




(b) Fig. 10a zoomed in.


(c) AR(2) fitting of the training data.

Figure 10: Forecasting plots of the AR(2) model.

Nevertheless, it is important to note that the test period of 10% of the data set corresponds to a time period of 25 months, meaning that based on our findings the ARMA(2,1) and AR(2) models could be suitable as a means of predicting the DEXUSEU for about a half year to a year in advance, respectively. To ensure the most accurate predictions of the exchange rate it would likely be most effective to update the model on a monthly basis and accept the predictions of a few months in advance as the most legitimate. Forecasting could be based on results from both models as well, since both the ARMA(2,1) and AR(2) provide accurate forecasts for the initial period after the training data.

In order to improve and further analyze our forecast, we also attempted to increase the training data percentage. However, the results of this were so abysmal that they were not included, and this can most likely be explained by the outbreak of COVID-19 and its effects on international markets. This leads us to the conclusion that it is possible the assessment of the accuracy of our forecast might have been tainted by global events which led to outliers in the test data.

# 5 Conclusion

We were correct in our initial assessment of the data in believing that the spot exchange data would have polynomial and seasonal components. This makes sense because the US dollar and Euro are two currencies closely dependent on each other and on recurring political and global events, which tend to have a seasonality based on certain times of the year. As expected, we were able to remove these trends from the data and obtain a stationary model.

While the forecast was not as accurate as we originally had hoped for, if the model can in fact be proven to correctly predict the DEXUSEU even a few months in advance, it would be considered a success. Currency exchange rates are extremely volatile and subject to change rapidly as a result of unforeseen circumstances, making it difficult to create a model which can forecast them too far in the future.

Both the AR(2) and ARMA(2,1) exhibited accuracy in forecasting the data immediately after the training set, which lead us to believe that increasing the training set size would significantly increase the accuracy of the forecast. Even though this did not work for us as previously mentioned, it could be worth it to try test this hypothesis by applying it to different parts of the historical data set where we would be confident that outlier events would not affect the test set. From there, we could run different trials to optimize the value of the training percentage in order to maximize the accuracy of the forecast.

Something else which could be also optimized for DEXUSEU spot exchange prediction models is the frequency of the data collection. We initially believed daily data would be too frequent and chaotic, which is why we chose monthly data, but it might be worth testing the former in addition to weekly and biweekly samples. From this and a series of trials, we could define the most accurate sampling frequency.

Generally speaking, we discovered that it is possible to model the DEXUSEU rate using an ARMA model. It is also possible to forecast this rate a short time period in the future. Future work related to this project could focus on optimizing parameters such as the training percentage, data set length, and data collection frequency, in order to define a framework for obtaining a desired accuracy for a desired forecast period.