SF2943 - Project Report

Dan Vicente (danvi@kth.se) **Erik Lindé** (elinde2@kth.se)
Sebastian Hauger (hauger@kth.se) **Alexander Gutell** (agutell@kth.se)

May 2024

1 Datasets

We have examined two different datasets, the first dataset is from the SMHI open-data archive [SMH] and the second dataset is the exchange rate between the USD and EUR currencies during some time period. To generate test-sets for prediction we omit a portion on the end of the time series and use these remaining points as ground truth for making rolling forecasts of a specified length, the specifics of this procedure are outlined in the methods section.

1.1 Dataset A: Temperature data from Stockholm-Bromma airport

SMHI collects massive amounts of data from meteorological obseration stations across Sweden every day. Since we live in Stockholm, we have chosen to consider the meteorologial observation station located at Stockholm-Bromma airport (BMA), which measures variables such as humidity, air pressure, cloud altitude and temperature. Each of these variables can be downloaded via the API as a time-series, of which we have chosen to consider temperature. Temperature is a complex physical quantity with many innate cyclical structures, for instance; it is usually colder in the winter than in the summer and colder in the evening than during the day etc.

1.2 Dataset B: EUR/USD Currency pair

The foreign exchange market is an old financial market where companies, banks, central banks, people etc exchange currencies or buy currency pairs. The EURUSD pair is the most traded currency [inv] and is very highly linked to the real economy and global interest rates. We will not consider any of the macroeconomic implications of the currency pair, we are simply interested if we can make accurate forecasts of the future price of the EUR/USD currency pair using methods from this course. [BD02]

2 Software

We have used `python3` for our analysis. In particular, we have used `pandas, matplotlib` and `numpy` for the analysis of the data. For the Timeseries specific analysis, we have used the `statsmodels.tsa` API for defining our ARMA(p,q) processes and computing/visualizing ACFs etc. [sta]

3 Methods

3.1 Cleaning the data

3.1.1 Dataset A: Making temperature stationary

One way to remove the daily cyclicality is by averaging. New temperature data is added every hour at BMA and via the API we can in fact get a daily average (of all the 24 datapoints), effectively removing the daily cyclicality. To remove the yearly cycles, we consider the transformed timeseries $Y(t) = \mathcal{B}^{365}T(t)$, where t denotes time (daily), $\mathcal{B}$ is the lag operator and T is the daily average temperature at BMA. A seasonal decomposition (`seasonal_decomposition()`) with period set to 365, was used to retrieve the residual timeseries without seasonality or trend (see fig 1).

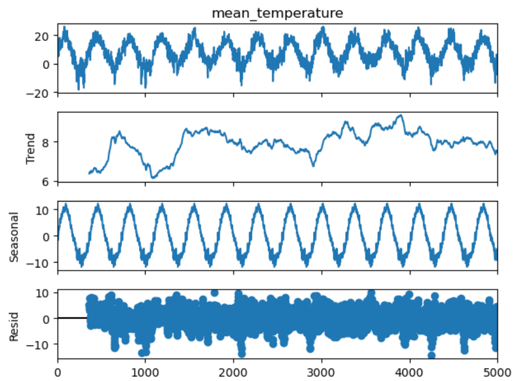


Figure 1: A seasonal decomposition of the data splits the data into a seasonal component, a trend component and a residual component

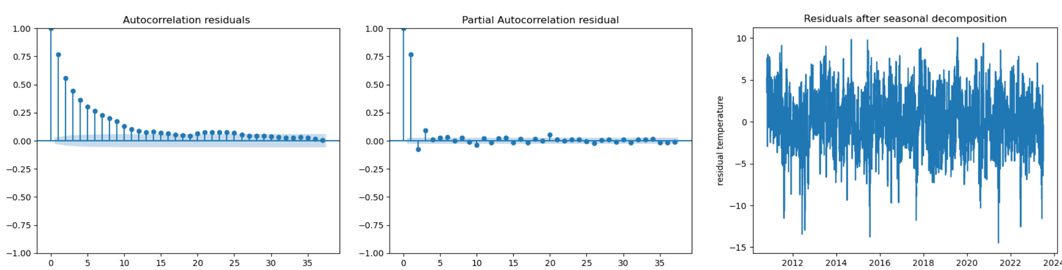


Figure 2: The ACF shows that the correlation decreases geometrically meaning that we can apply `AR` models. The PACF shows that each value is highly correlated with the previous value.

We perform an augmented dickey fuller test (see [BD02] 6.3.1), which gives us the ADF test statistic $= -11.7$ and p-value$=\ 2.1 \cdot 10^{-21}$, so we can conclude that the time series is stationary with a high statistical significance.

3.1.2 Dataset B: Making EUR/USD stationary

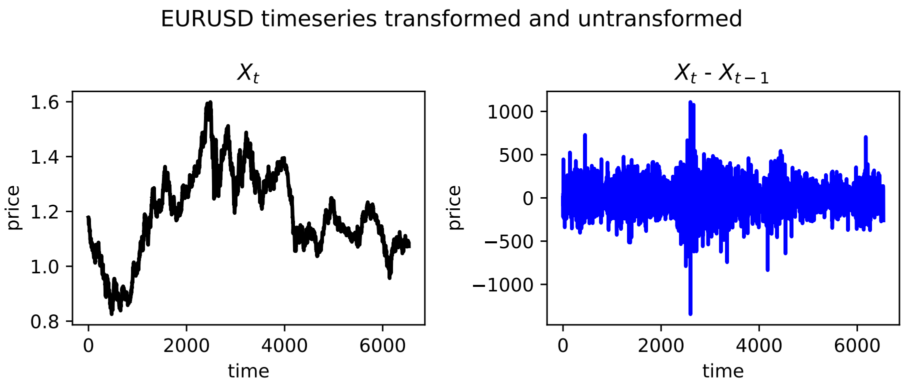


Figure 3: EURUSD timeseries X_t over some time period.

Due to the nature of financial markets, one might expect the previous price measurement to be the dominant component in the next price measurement. With our daily data, we thus take the daily differences in a lag one model. Upon this change, and normalization, we have the following results for ACF and PACF in figure 4.

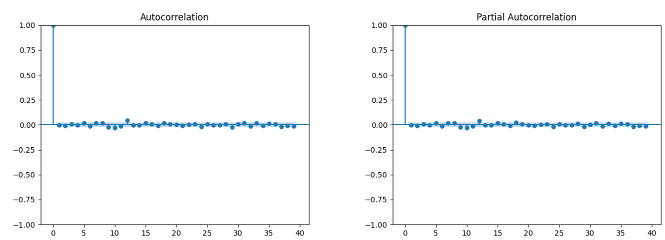


Figure 4: The ACF shows little to no autocorrelation for $h > 0$. The PACF shows similar results.

The results exhibit stationarity. Furthermore, an augmented dickey fuller test is conducted and the obtained p-value is numerically given as 0. Thus, the data with lag 1 seems to be stationary. From here on, the results with financial data will be conducted on this stationary data.

3.2 Order selection

3.2.1 Dataset A: Choosing a model

In order to fit an ARMA(p,q) model to the residual component, the optimal parameters p and q must be estimated. We implemented a grid search over p, q where we searched for the minimum of the different AIC values when we fit the ARMA(p, q) models to the data [BD02]. The grid was set to $[0, \dots, 4] \times [0, \dots 4]$. The AIC values are summarized in Table 1.

p \ q	0	1	2	3	4
0	24025	21217	20443	20157	20004
1	19889	19857	19826	19823	19823
2	19865	19843	19820	19822	19824
3	19826	19825	19822	19824	19825
4	419827	19829	19825	19821	19818

Table 1: AIC on the grid

constant term	0.062496053736981326
$\hat{\phi}_1$	1.1871469503536396
$\hat{\phi}_2$	-0.28643514932010966
$\hat{\theta}_1$	-0.35307591086302303
$\hat{\theta}_2$	-0.16717947032508915
$\hat{\sigma}^2$	4.152376501446936

Table 2

Note that not all these simulations were able to converge, thereby we get some very large outliers. We notice that there is a local minima at $p = q = 2$, this is also the global minima of all the solutions that were able to converge (as $p = q = 4$ is not a solution that converged). After fitting the model we get the parameters seen in table 2

3.2.2 Dataset B: Choosing a model

We fit an ARMA(p,q) model here as well. We implemented a grid search over p, q for values $[0, \dots, 4] \times [0, \dots 4]$ where we searched for the minimum of the different AIC values when we fit the ARMA(p, q). The AIC values are summarized in Table 3.

p \ q	0	1	2	3	4
0	82432.3	82434.1	82435.3	82436.9	82438.8
1	82434.1	82436.2	82435.7	82439.0	82439.3
2	82435.3	82435.6	82439.3	82436.4	82429.2
3	82437.0	82439.0	82439.4	82438.5	82430.3
4	82438.9	82440.9	82441.5	82443.6	82445.8

Table 3: Grid search of financial data AIC minimization.

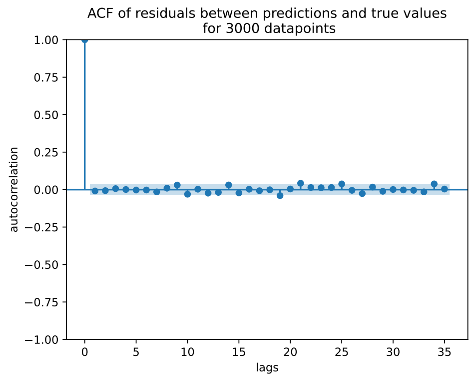
We see that the minimum is attained for $p = 2$ and $q = 4$. However, the mle estimate in the computation of the AIC-score did not converge and there are several warning flags from the algorithm fitting the ARMA models. It is thus ambiguous if this ARMA$(2, 4)$ model really is most appropriate.

3.3 Forecasts

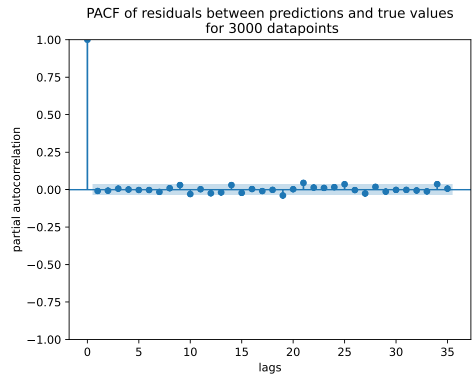
3.3.1 Dataset A: Forecasting temperature

`statsmodels.tsa` package offers a h-step forecast function, which uses linear estimators to predict the value at step h. A forecast was done on the final part of the data set, see fig.5c, where the yellow graph displays h = 1 step predictions estimated from included ground truth data. The green line displays a forecast of h = 30 without ground truth data.

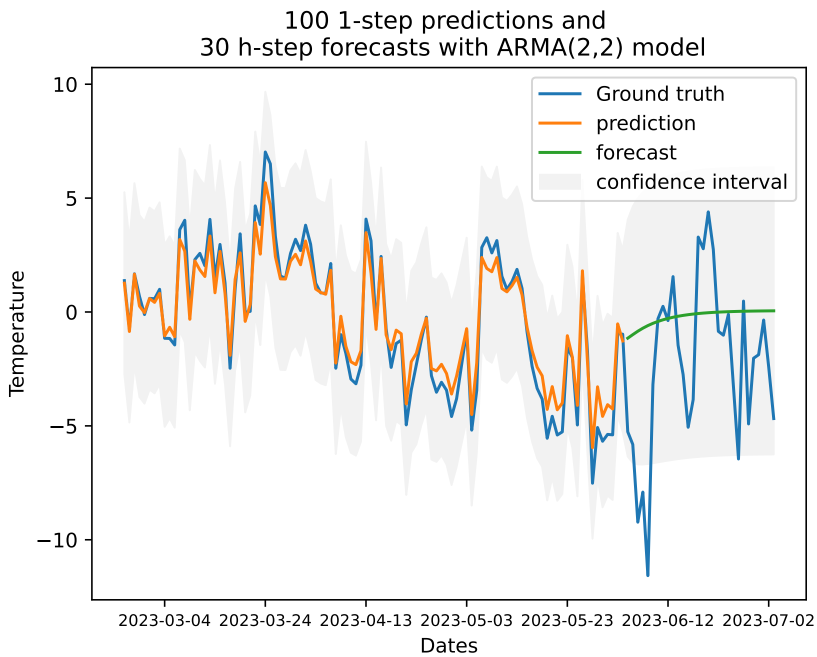
In order to check if the model truly fits the data, the errors between the predictions on the training data was calculated, see fig.5c. The ACF and PACF of these errors was then plotted, see fig.5.



(a) ACF prediction errors



(b) PACF of prediction errors



(c) Forecasted timeseries

Figure 5: ACF and PACFs of residuals in the predictions made with an ARMA(2,2) model. And forecasts

3.3.2 Dataset B:Forecasting EUR/USD

As with dataset A, `statsmodels.tsa` was used to forecast the financial data. A forecast was once again done on the final part of the data set as is shown in figure 6.

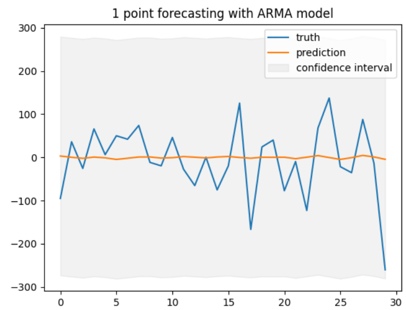


Figure 6: One-day predictions of 1-day differenced EURUSD.

The forecast is very flat, which means that the model prediction is not impacted much by the previous values. Instead, the next prediction even for the ARMA$(2,4)$ model tends to be close to zero. We can thus compare this model to just a white-noise model, and compare the predicitons.

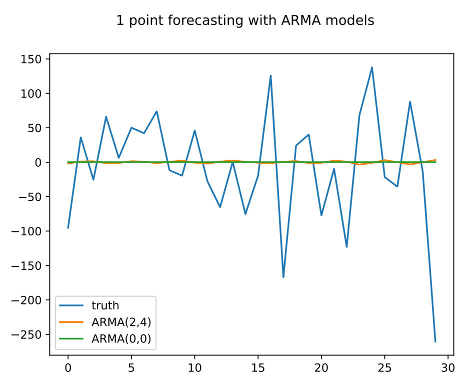


Figure 7: 1-step predictions of 1-day differenced USDEUR with ARMA(2,4) and ARMA(0,0) models

We see that they are nearly identical, and the curves lie very close to each other. Due to the lack of convergence, we prefer the simple white-noise model to predict the stationary series. This can also be inferred from the ACF and PACF plots, as they are almost zero for $h > 0$. The plot seen in Figure 7 confirms this heuristic argument.

4 Results

The predictions displayed in fig.5c as the yellow graph, show that our fitted ARMA(2,2) fits the training data well. Furthermore, the prediction errors are white noise, which also indicates a good fit, see fig.5.

Regarding the forecast, we got the expected behavior from an ARMA(2,2) model. The green line in fig.5c indicates that a good forecast above h = 2 was not possible, and that subsequent forecasts converged to mean zero.
For the financial data, after finding a stationary series by differencing the previous day's trading price for USD/EUR, the optimal model for the stationary series was ARMA$(2,4)$, at least according to the AIC

criterion. However, due to lack of convergence and the similarity of the ACF and PACF-curves to a simple white-noise ARMA(0,0) model, the latter was chosen as a better model due to simplicity, in accordance with Occam's razor. In other words, the best prediction of the value of the currency pair EURUSD at any given day is simply the price of the trading pair the previous day.

5 Discussion

One of the difficulties in the project has come from the fact that the REST API of SMHI can be inconsistent; not all data is supplied in the same format, which means that the process of parsing and cleaning the data is hard to automate completely, thus making the initial data exploration somewhat cumbersome. Another difficulty arises in the documentation of `statsmodels`, compared to many other modern packages for data science and engineering, the documentation is quite terse and in many cases non-existant. The analysis is quite robust according to the theory we have went through in the course so far. We used all the available theory regarding ARMA processes, and utilized this in our methodology. However, we did not try to fine tune the transformation to stationarity, this is something that could be improved as an extension. Moreover, it could be interesting to consider more complex/nonlinear time series models such as ARCH, GARCH etc, since the seasonal decomposition seems to reveal a nonlinear trend in the data.

References

[BD02] Peter J Brockwell and Richard A Davis. *Introduction to time series and forecasting*. Springer, 2002.

[inv] investopedia. *EUR/USD*. URL: `https://www.investopedia.com/terms/forex/e/eur-usd-euro-us-dollar-currency-pair.asp`.

[SMH] SMHI. *Meteorological Observations Open data API*. URL: `https://opendata.smhi.se/apidocs/metobs`.

[sta] statsmodels. *statsmodels.tsa API docs*. URL: `https://www.statsmodels.org/stable/tsa.html`.