Bitcoin and Ethereum cryptocoins´ forecasting using Exponential Smoothing

Author: Lúcia Moreira

Date: 17/12/2019

Summary

Exponential smoothing methods were used for 10 days forecasting on two crypto-coins regarding its open, volume and spread values for Bitcoin and market capitalization for Ethereum. The 2019 daily historical data ranges from January 1st to December 5th and forecasts were compared with real market values extracted from December 6th to December 15th, 2019. Fitting to the 340 days in year 2019 was very good as well as the first-day ahead forecast comparison with the market values observed to that specific day. For longer times, prediction was not so precise.

Introduction [1]

When performing forecasting using naïve methods, all forecasts for the future are equal to the last observed value of the series. While using the averaged method, all future forecasts are equal to a simple average of the observed data. Hence, the average method assumes that all observations are of equal importance, and gives them equal weights when generating forecasts. But what is desired often is something between these two extremes. For example, it may be sensible to attach larger weights to more recent observations than to observations from the distant past. This is exactly the concept behind simple exponential smoothing. Forecasts using exponential smoothing methods are weighted averages of past observations, where the weights decay exponentially as the observations get older (the more recent the observation the higher its weight).

The simple exponential smoothing (SES), is suitable for forecasting data with no clear trend or seasonal pattern.

The simple exponential smoothing was extended by the Holt's linear method to allow forecasting with trend. The method involves a forecast equation and two smoothing equations (one for the level and one for the trend, with α and β smoothing parameters). These forecasts present a constant trend indefinitely into the future and because of that tend to over-forecast, especially for longer forecast horizons.

To prevent such over-forecasting, a parameter that "dampens" the trend to a flat line sometime in the future was introduced. So, in conjunction with the smoothing parameters α and β, this method also includes a damping parameter $0 < \phi < 1$. Such modified methods that include a damped trend have proven to be very successful, and are quite popular methods when forecasts are required automatically for many series.

Later, the extended Holt's method was modified to also capture seasonality. The Holt-Winters seasonal method involves the forecast equation and three smoothing equations —

one for the level, one for the trend, and one for the seasonal component, with the corresponding smoothing parameters α, β and γ.

As each of these models consist of a measurement equation that describes the observed data, and some state equations that describe how the unobserved components or states (level, trend, seasonal) change over time, they are usually referred as state space models.

In such state space models there are two approaches: one with additive errors and one with multiplicative errors. The point forecasts produced by both models are identical if they use the same smoothing parameter values, however, they generate different prediction intervals.

To distinguish between a model with additive errors and one with multiplicative errors, a third letter to the model classification system was added. They have labelled each state space model as ETS(·,·,··,·,·) for (Error, Trend, Seasonal). This label can also be thought of as ExponenTial Smoothing. A great advantage of the ETS statistical framework is that information criteria can be used for model selection.

The models can be estimated in R using the `ets()` function in the forecast package using the three-letter code. The possible inputs are “N” for none, “A” for additive, “M” for multiplicative, or “Z” for automatic selection. So, the possibilities for each component are: Error ={={A,M}}, Trend ={={N,A,Ad}d} and Seasonal ={={N,A,M}}. The following parameters can also be tuned:

`damped`

If `damped=TRUE`, then a damped trend will be used (either A or M). If `damped=FALSE`, then a non-damped trend will be used. If `damped=NULL` (the default), then either a damped or a non-damped trend will be selected, depending on which model has the smallest value for the information criterion.

`alpha, beta, gamma, phi`

The values of the smoothing parameters can be specified using these arguments. If they are set to `NULL` (the default setting for each of them), the parameters are estimated.

`lambda`

Box-Cox transformation parameter. It will be ignored if `lambda=NULL` (the default value). Otherwise, the time series will be transformed before the model is estimated. When `lambda` is not `NULL`, `additive.only` is set to `TRUE`.

The dataset

The historic crypto currency market daily data for the 2019 year from January 1st up to December 5th was obtained using the *crypto_history()* function from the crypto package available in R. The function returns among other variables the following ones:

date	Market date
open	Market open
high	Market high
low	Market low
close	Market close
volume	Volume 24 hours
market	USD Market cap
spread	Volatility premium, high minus low for that day

The crypto coins considered were both Bitcoin and Ethereum.

Forecasts were performed for 10 days ahead and are compared with the values extracted with the *crypto_history()* function from December 6th to December 15th, 2019.

Fig 1 and 2 show the daily history of the variables: open, high, low, close, volume and market capitalization for the 340 selected days in 2009 for Bitcoin and Ethereum, respectively.

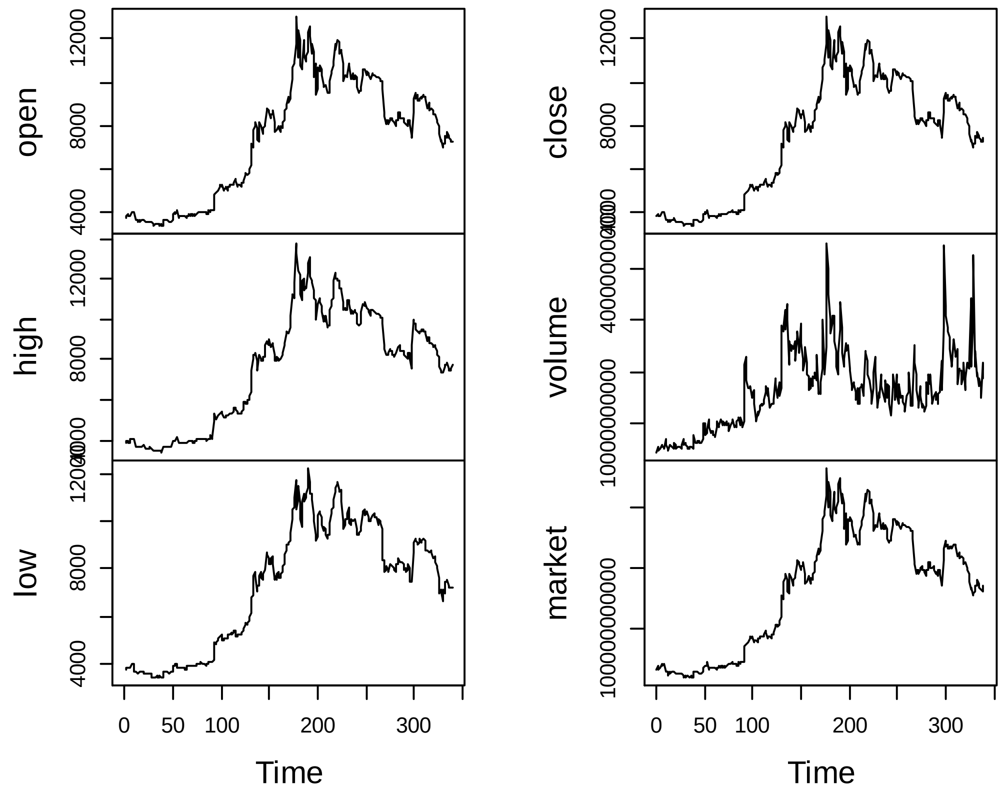

Fig. 1- Daily historical data for Bitcoin from January 1st, up to December 5th, 2019 (340 days).

Fig. 3 (left) shows that variables open, high, low and close majorly overcome for each of the crypto coins, so variable open was chosen forecasting for the Bitcoin together with variable volume and spread. Regarding Ethereum, only forecasting for variable market capitalization was considered. By comparing both crypto coins, one can see that market value of Bitcoin is substantially larger (2 orders of magnitude) than Ethereum, being the major crypto coin in the market nowadays. Also one can see that both coins follow nearly the same daily trend.

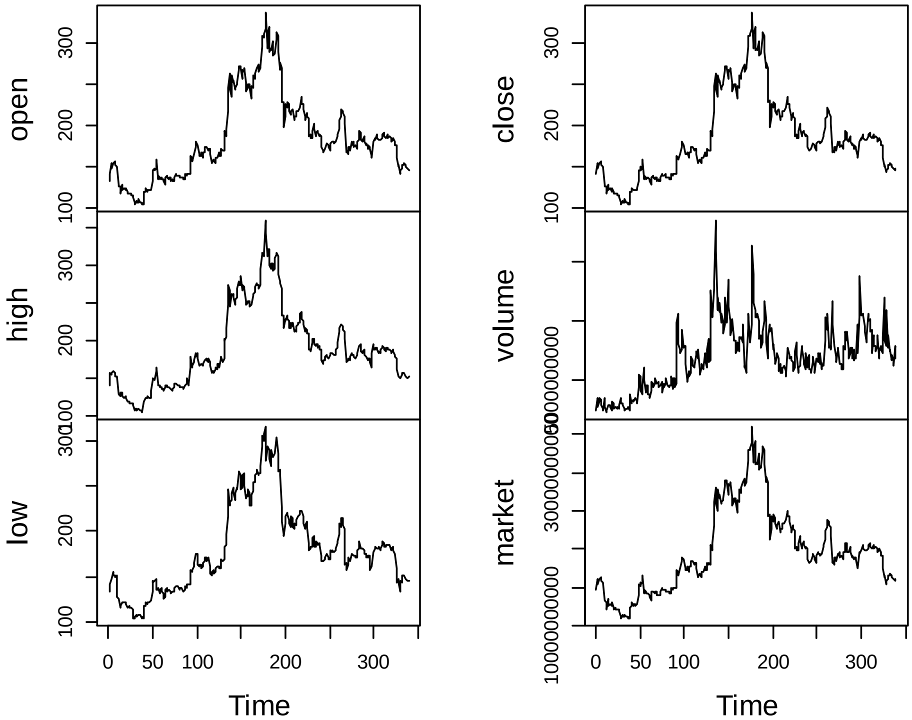

Fig. 2 - Daily historical data for Ethereum from January 1st, up to December 5th, 2019 (340 days).

Forecasting

Forecasting was performed for ten days ahead, from December 6th to December 15th, regarding the open value for the crypto coin Bitcoin. Fig.4 left shows the 10 days forecast including the 95 % confidence interval. Fig. 4 (right) shows the fitted model (red) to the real data), non-parametric fitting with the ets() function was very good.

The outcome of this fitting is:

```
Smoothing parameters:
    alpha = 0.9738

  Initial states:
    l = 3744.253

  sigma:  0.0365

     AIC     AICc      BIC
5736.852 5736.924 5748.339
```

And indicating that a ETS(M,N,N) was fitted to the open Bitcoin values.

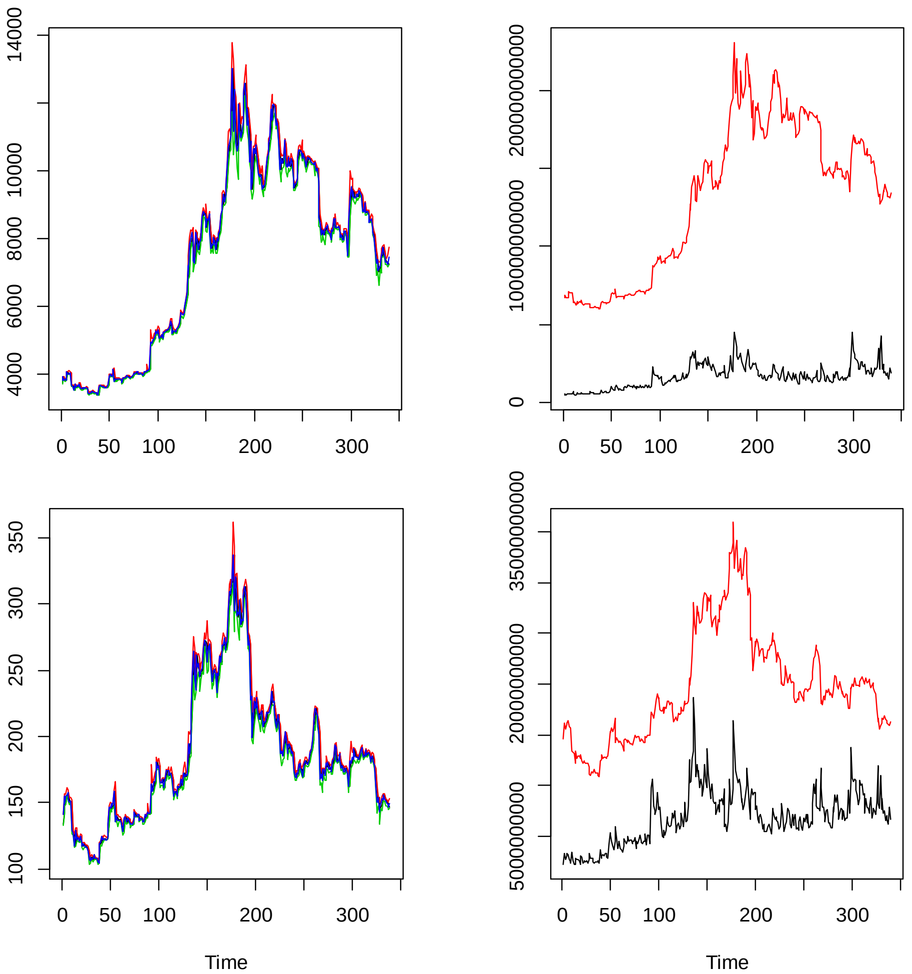

Fig. 3 – Combined daily historical plots for both crypto-coin for the first 340 days of year 2009 (top: Bitcoin, bottom: Ethereum).

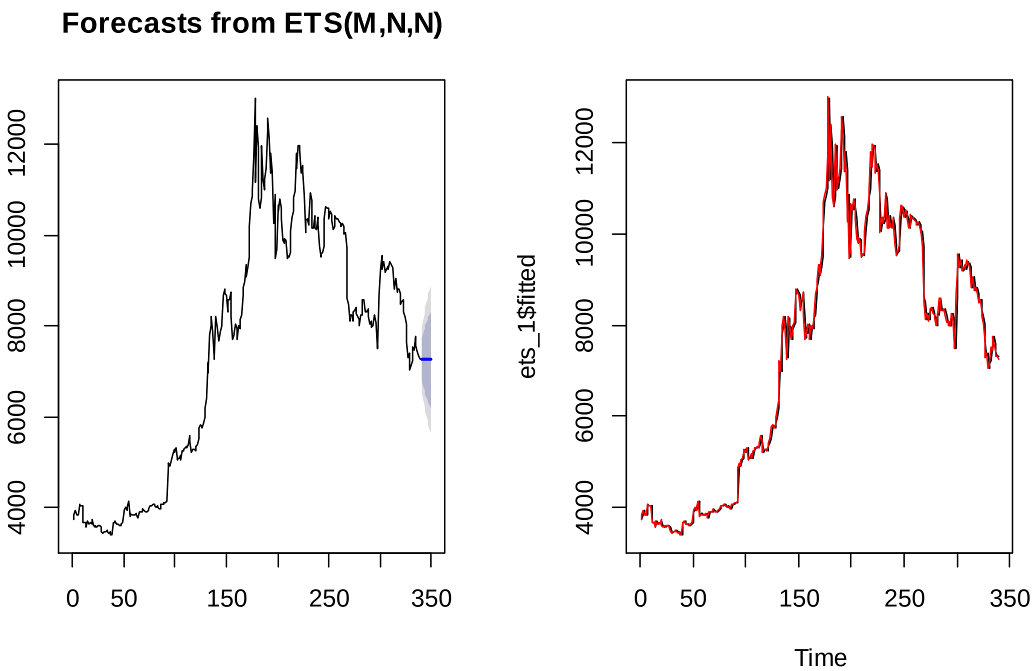

Fig. 4 - ETS(M,N,N) fitting (right, in red) and 10-days forecast (left) to the open Bitcoin values for the first 340 days of 2019.

Fig. 5 shows the 10 days history of the Bitcoin values for the open, high, low and close values as well as for the 24 hours volume traded and the respective market capitalization. Fig. 6 shows the comparison between the forecasting and the real values observed in the real market for crypto coin Bitcoin for the same days considered in forecasting. Fig. 6 shows that the point forecast (red line) matches the one-day ahead while deviating for a more distant future, however, the real data observed is within the prediction interval.

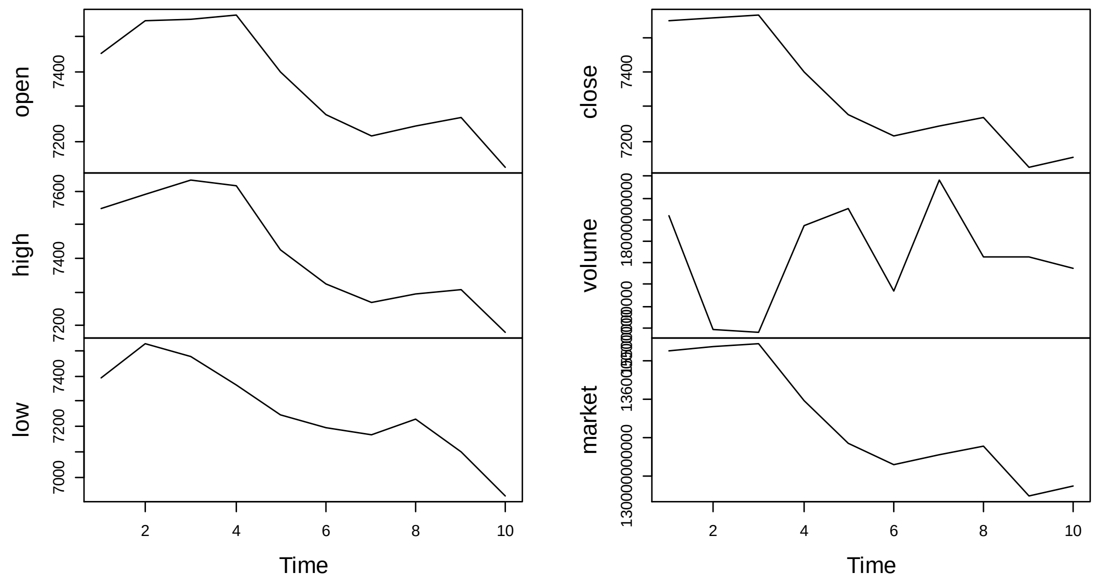

Fig. 5 – Daily history for the Bitcoin crypto coin from December 5th to 15th 2019 used to compare with the 10-days ahead forecast performed.

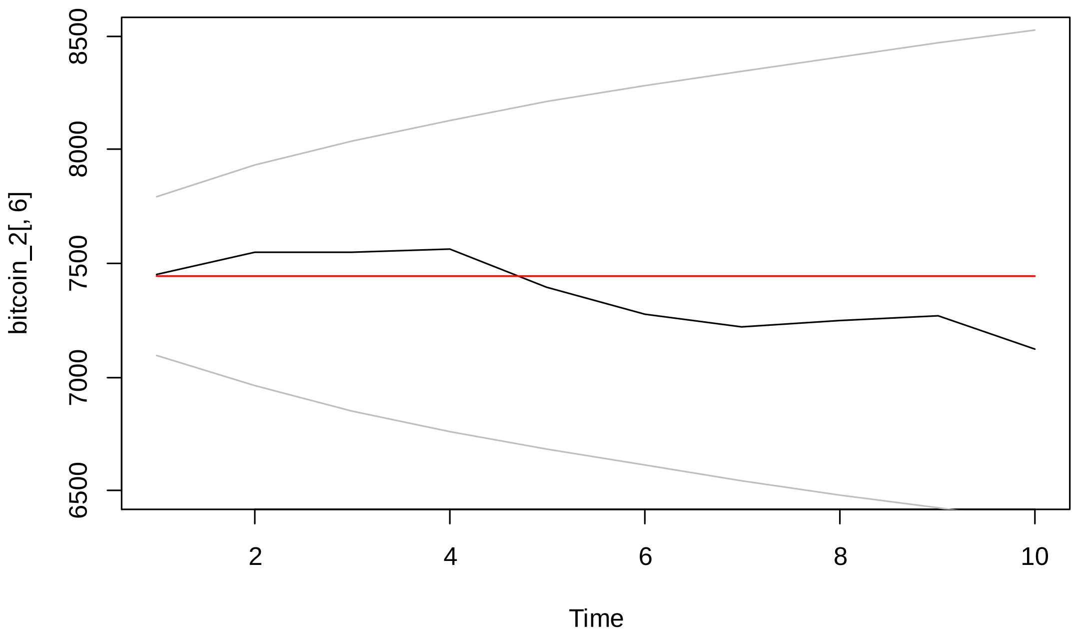

Fig. 6 – Open values daily history for the Bitcoin crypto coin from December 5th to 15th 2019 comparison with the 10-days ahead forecast.

The same approach was considered but now for the 24h volume traded for the crypto coin Bitcoin. Fig. 7 (right) shows the fitted model (red) to the real data; non-parametric fitting with the ets() function was very good but prediction interval was quite large. The outcome of this fitting is:

```
Smoothing parameters:
    alpha = 0.9106
    beta  = 0.0001

  Initial states:
    l = 4650374718.6885
    b = 166849166.6465

  sigma:  0.1795

     AIC     AICc      BIC
16740.80 16740.98 16759.94
```

And indicating that a ETS(M,A,N) was fitted to the 24 volume Bitcoin values.

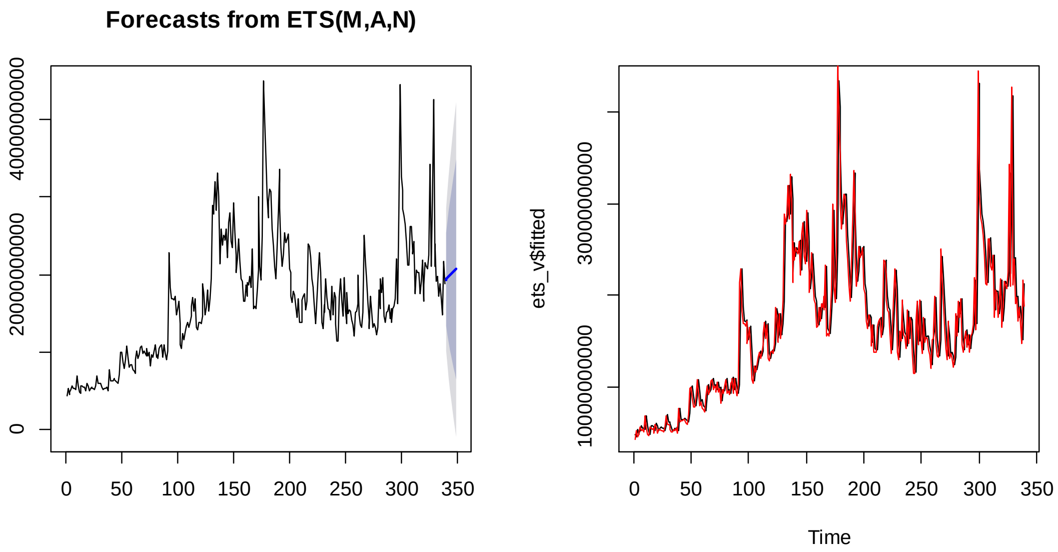

Fig. 7 - ETS(M,A,N) fitting (right, in red) and 10-days forecast (left) to the 24 h volume Bitcoin values for the first 340 days of 2019.

Fig. 8 shows the comparison between the forecasting and the real values observed in the real market for crypto coin Bitcoin for the same days considered in forecasting. Fig. 8 shows that the point forecast (red line) matches once the first-day ahead while deviating for a more distant future, however, the real data observed is within the prediction interval. Values are in log scale due to the very high amounts involved.

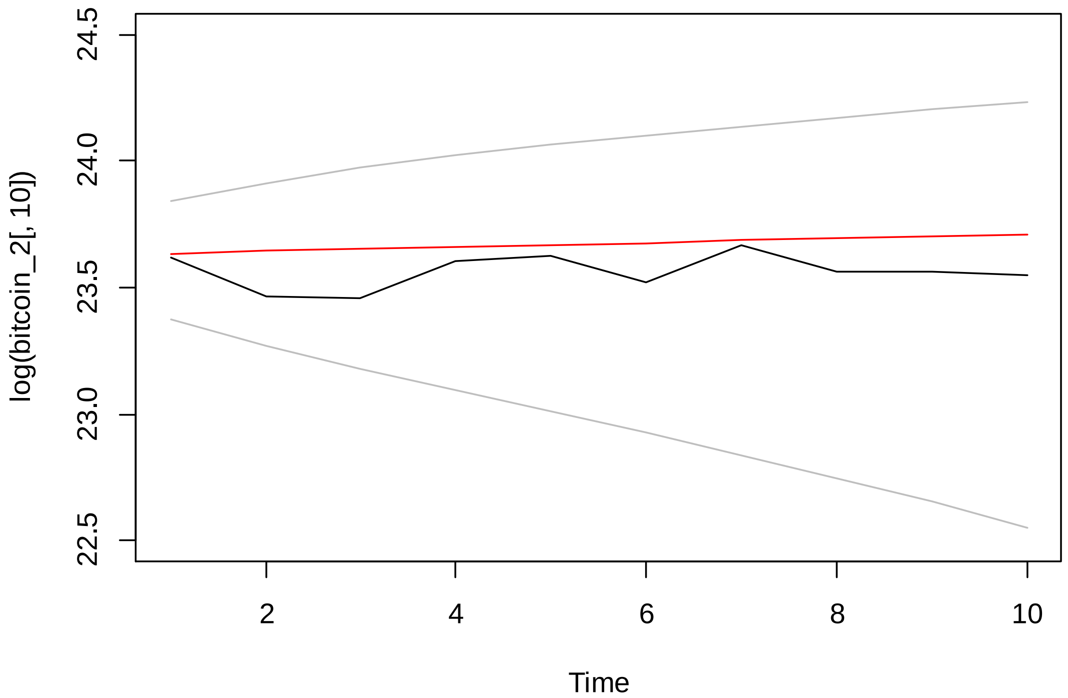

Fig. 8 – 24 h volume log values daily history for the Bitcoin crypto coin from December 5th to 15th 2019 comparison with the 10-days ahead forecast.

To conclude the analysis on Bitcoin, the forecast for the volatility premium (high minus low for that day – spread) was considered. The same conclusions as before are obtained from a ETS (M,A, N) fitting to the real data (Fig. 9). Fig. 10 compares the predictions with the observed values, and in this case even the first-day forecast was not predicted as for the other variables.

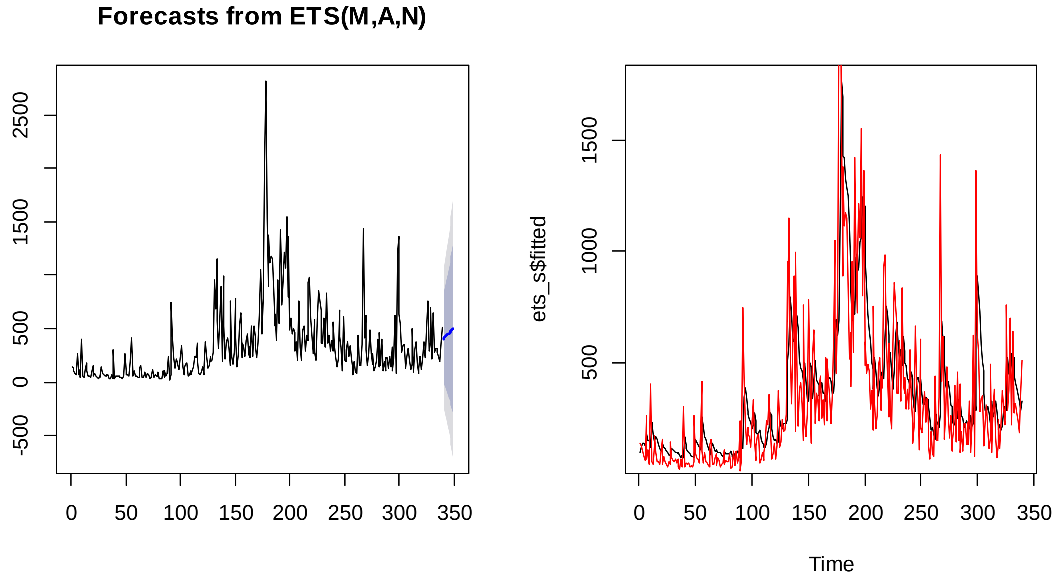

Fig. 9 - ETS(M,A,N) fitting (right, in red) and 10-days forecast (left) to the spread Bitcoin values for the first 340 days of 2019.

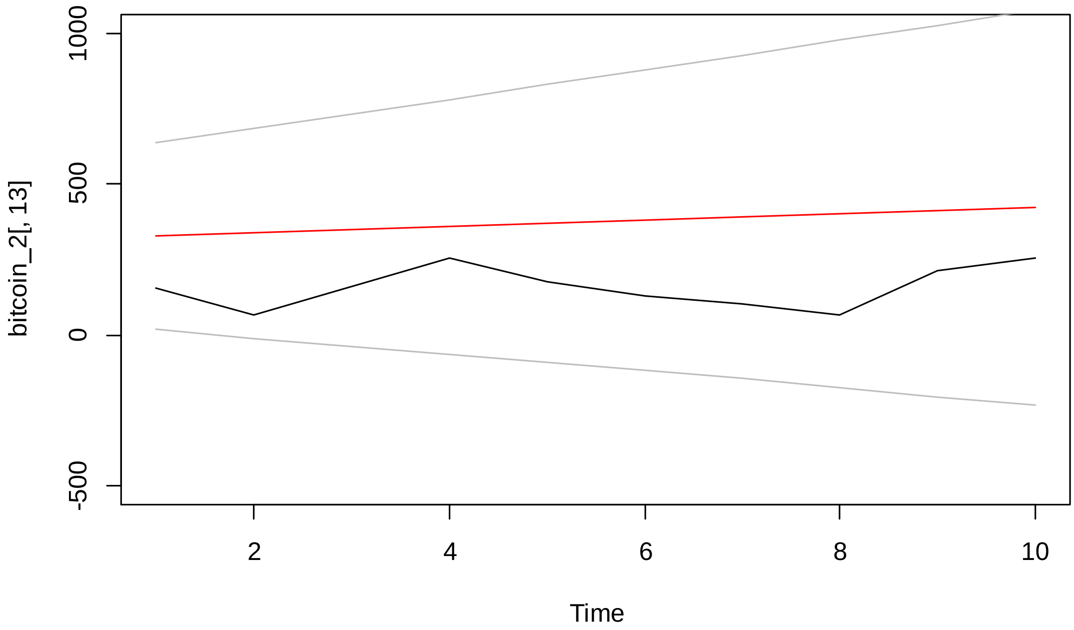

Fig. 10 – Spread values daily history for the Bitcoin crypto coin from December 5th to 15th 2019 comparison with the 10-days ahead forecast.

Finally, the market capitalization 10 days-ahead forecasting the Ethereum crypto coin is considered. Fig. 11 shows the 10 days history of the Ethereum values for the open, high, low and close values as well as for the 24 hours volume traded and the respective market capitalization.

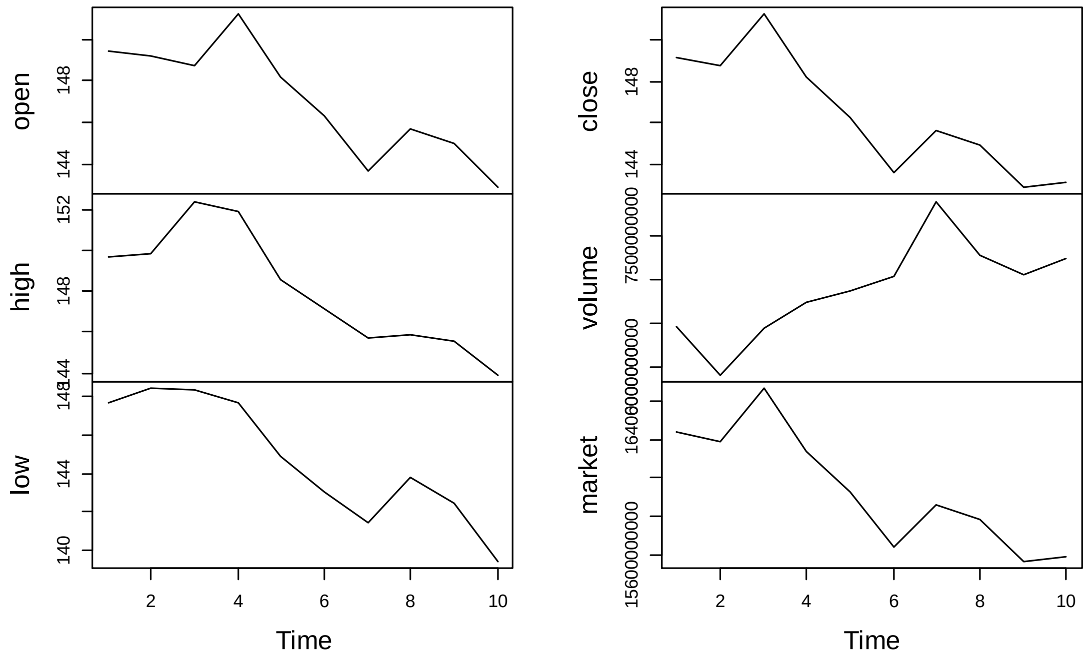

Fig. 11 – Daily history for the Ethereum crypto coin from December 5th to 15th 2019 used to compare with the 10-days ahead forecast performed.

Fig. 12 (right) shows the fitted model (red) to the real data; the non-parametric fitting with the ets() function was very good and the prediction interval was not very large. The outcome of this fitting is:

```
Smoothing parameters:
    alpha = 0.936

  Initial states:
    l = 15575755835.8814

  sigma:  0.0417

     AIC     AICc      BIC
15920.54 15920.62 15932.03
```

And indicating that a ETS(M,N,N) was fitted to the market capitalization Ethereum values.

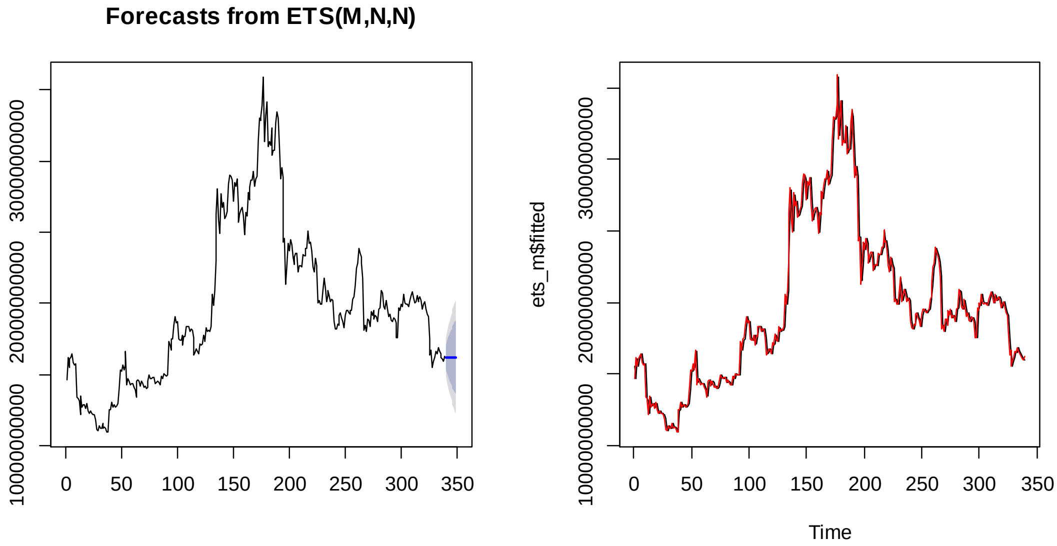

Fig. 12 - ETS(M,N,N) fitting (right, in red) and 10-days forecast (left) to the market capitalization Ethereum values for the first 340 days of 2019.

Fig. 13 compares the predictions with the observed values, and in this case both the first and second-day logarithm forecast of the market capitalization for the crypto coin Ethereum was well predicted.

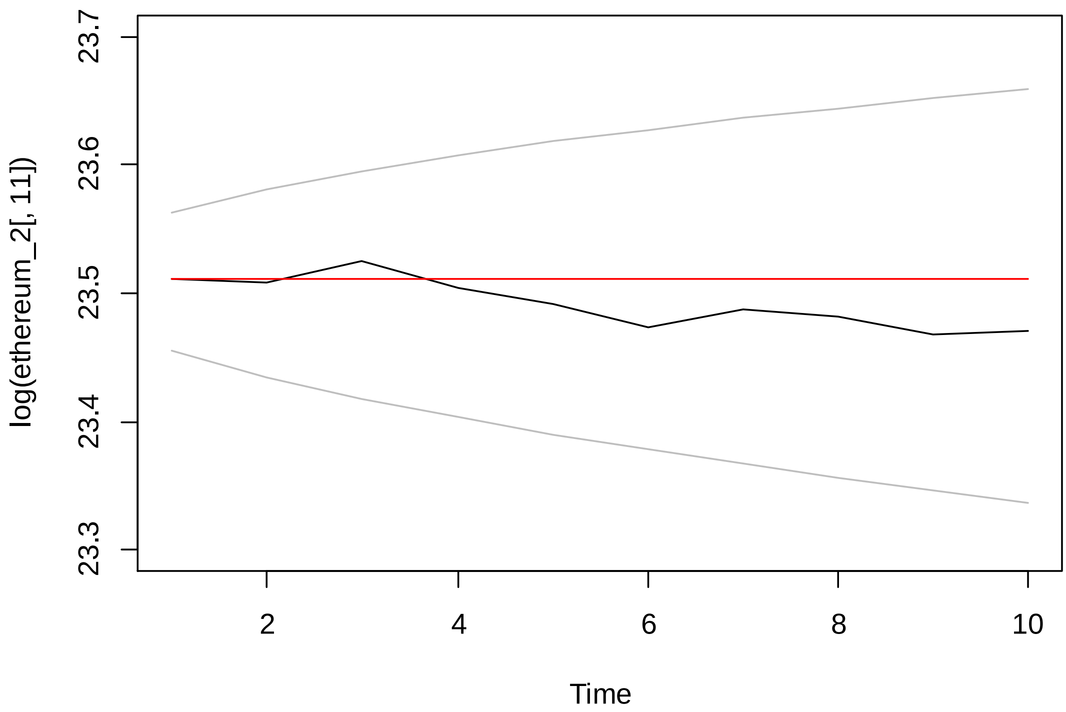

Fig. 13 – Market capitalization log values daily history for the Ethereum crypto coin from December 5th to 15th 2019 comparison with the 10-days ahead forecast.

Conclusions

Forecasting of real crypto coin market data was performed for the Bitcoin and Ethereum coins by using the non-parametric approaching exponential smoothing. The est() function with automatic detection of the type of errors was considered. Fitting to the 340 days of year 2019 was very good as well as the first-day ahead forecast comparison with the market values observed to that specific day. For longer time prediction was not so precise.

Reference

[1] – Very short summary by adapting sentences from Chapter 7 from the book “Forecasting Principles”, by Rob Hyndman.