Does ML algorithm performance change over time?

|  |  |  |  |
| --- | --- | --- | --- |
| Doireann Hanley Trinity College Dublin Computer Science and French dhanley@tcd.ie | | Eunice OreOluwa Fasan Trinity College Dublin Computer Science and French fasane@tcd.ie | Sloane Reeves Trinity College Dublin Computer Science and German sreeves@tcd.ie |
|  |

# Introduction

# EXPERIMENTAL AND COMPUTATIONAL DETAILS

# Methodology

The dataset we used, titled ‘Monthly milk production: pounds per cow. Jan 62-Dec 75’ was sourced from the Time Series Data Library (TDSL)[1]. The dataset consists of 168 fact values in 1 time series. The time granularity is Month and the units used are pound per cow. The time range is Jan 1962 – Dec 1975.

Our data frame was created by reading the data into a .csv file. We parsed and formatted the dates accordingly and set the attribute ‘Month’ to be our index. We then simply plotted the data to look for potential patterns, evidence of trend and seasonality. The data was then split into training and test sets in the 80:20 ratio.

We defined our problem and based on our time series data, proceeded to implement a range of quantitative forecasting methods. These included Moving Average, Exponential Smoothing, Double Exponential Smoothing, Holt-Winter’s model, ARIMA, Simple Lags/Linear regression, Ridge regression, Lasso regression and XGBoost.

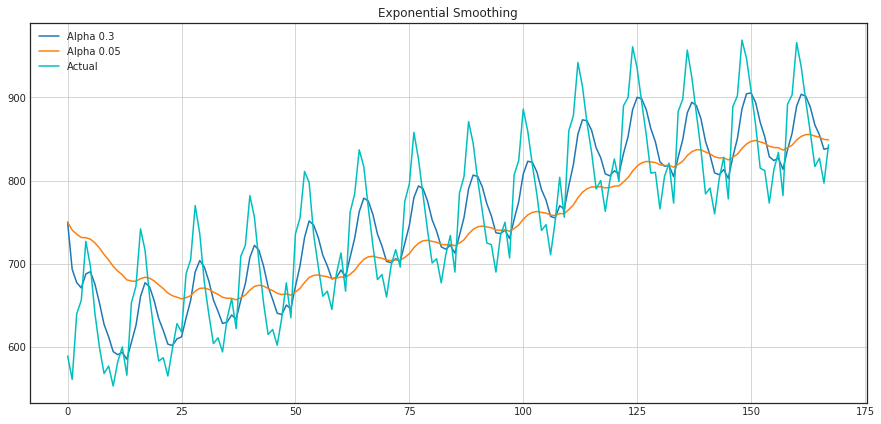
We chose to measure the quality of our forecasts using Mean Absolute Percentage Error (MAPE) as we consider it an interpretable metric that is particularly interesting because of its robustness to outliers.

We applied our baseline model, Moving Average, to smooth the original time series to identify trends. A close up of a map

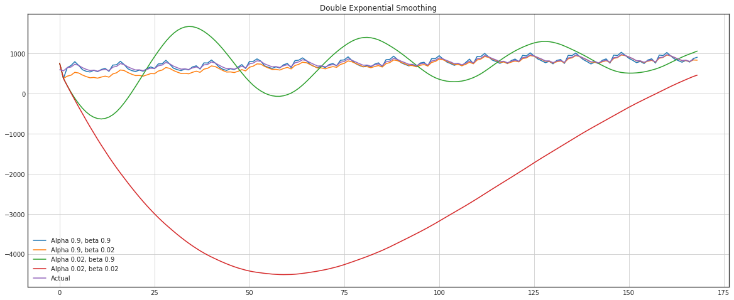
Description generated with high confidence

*Figure 1: Results from smoothing original time series with the previous 12 months and plotting the confidence intervals*

We then applied exponential smoothing to see what would happen if, instead of weighting the last k values of the data, we weighted all available observations while exponentially decreasing the weight the further back in time we moved. We experimented with various alpha values for our smoothing parameters.

*Exponential smoothing of time series with alpha values of 0.3 and 0.05*

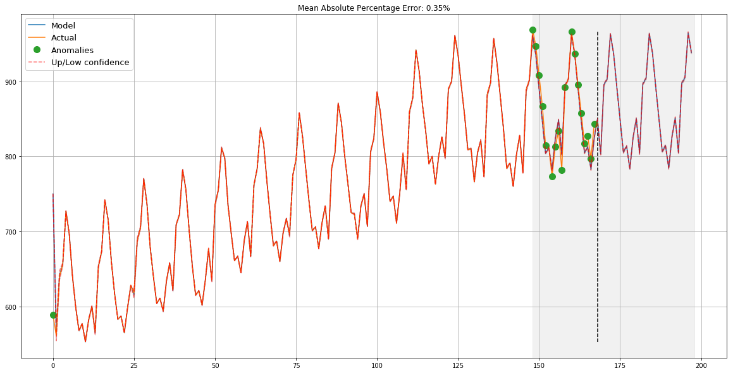
Both moving average and exponential smoothing up to this point were used for single future points prediction however this was insufficient. As a result, we decided to extend our exponential smoothing model to produce a double exponential smoothing capable of predicting two future points.



*Double exponential smoothing of data with alpha values of 0.9 and 0.02 and beta values of 0.9 and 0.02*

We tuned the two parameters, alpha-responsible for the series smoothing around itself, and beta-responsible for the smoothing of the trend itself.

We then decided to add a third component, seasonality, to our smoothing effectively applying a Holt-Winters model once we had successfully split our data into training and test sets. We chose an algorithm (the truncated Newton conjugate gradient) that supports constraints on the model parameters.



*Holt-Winter model approximating initial time series, capturing annual seasonality, overall upwards trend and even some anomalies*

# RESULTS AND DISCUSSION

# Limitations and outlook

One crucial step to help us better evaluate our chosen models and their performance is to plot the error per time interval i.e. each month, as opposed to an overall single number metric telling only of the average error over the previous 12 months. Errors in our code denied us this insight therefore we will aim correct our code and produce the desire result as part of our second submission.

# CONCLUSIONS

A HEADINGS IN APPENDICES

A.1 Introduction

A.2 Experimental and Computational Details

A.3 Results and Discussion

A.4 Conclusions

A.5 References