Bike Sharing Analysis
Time Series Project

Mathieu Pont and Lucas Rodrigues Pereira

March 2020

Abstract

1 Introduction

In the context of the Master's Degree on Machine Learning for Data Science, at the University of Paris (Descartes), we have been given the task to analyze data from new online bike sharing systems, as part of the "Time Series" course.

Bike sharing systems are a new generation of traditional bike rentals where the whole process from membership, rental and return back has become automatic.

The dataset contains two csv files. Both hour.csv and day.csv have the following fields, except hr which is not available in day.csv

- **instant**: record index
- **dteday** : date
- **season** : season (1:winter, 2:spring, 3:summer, 4:fall)[1]
- **yr** : year (0: 2011, 1:2012)
- **mnth** : month (1 to 12)
- **hr** : hour (0 to 23)
- **holiday** : weather day is holiday or not (extracted from http://dchr.dc.gov/page/holiday-schedule)
- **weekday** : day of the week
- **workingday** : if day is neither weekend nor holiday is 1, otherwise is 0.
- **weathersit** :
  - 1: Clear, Few clouds, Partly cloudy, Partly cloudy
  - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
  - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
  - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- **temp** : Normalized temperature in Celsius. The values are divided to 41 (max)
- **atemp**: Normalized feeling temperature in Celsius. The values are divided to 50 (max)
- **hum**: Normalized humidity. The values are divided to 100 (max)
- **windspeed**: Normalized wind speed. The values are divided to 67 (max)
- **casual**: count of casual users
- **registered**: count of registered users
- **cnt**: count of total rental bikes including both casual and registered

[1] https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

2 Experiments and Results

2.1 Examining data

The main goal of the study will be to forecast the number of bike rentals through time. In a preliminary analysis, we can see that the temperature has a correlation with the number of bike rentals. Indirectly the seasons play a role in the number of bike rentals.

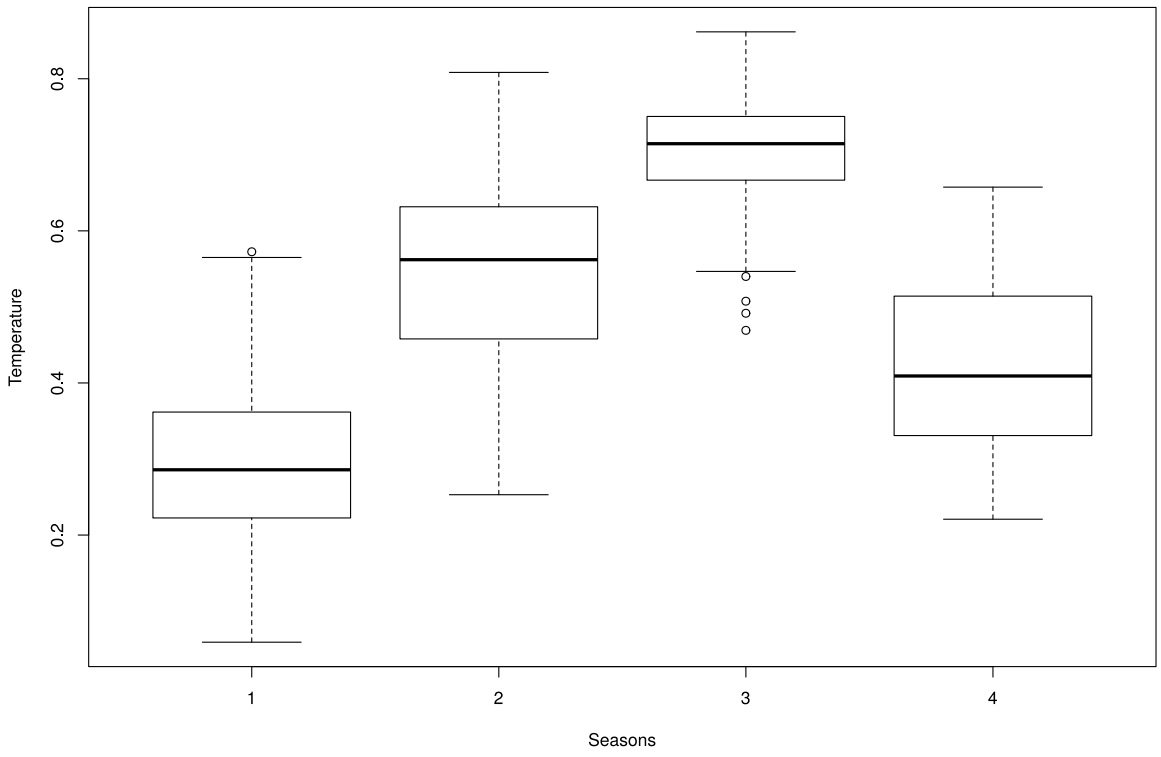

Figure 1: How temperature changes accross seasons.

	temp	**atemp**	**cnt**	**casual**	**registered**
temp	1.00	0.99	**0.63**	0.54	0.54
atemp	-	1.00	**0.63**	0.54	0.54
cnt	-	-	1.00	0.67	0.94
casual	-	-	-	1.00	0.39
registered	-	-	-	-	1.00

Table 1: Correlation between temperature and the number of rentals.

Like expected *temp* and *atemp* are highly correlated since one is the normalized temperature and the other one the normalized feeling temperature. Regarding the relation between these two variables and *cnt* we see that there is a correlation of 0.63, it is not significant but still quite high. Moreover when

we compute the correlation between *cnt* and the mean of *temp* and *atemp* we also found a correlation of 0.63.

When we look at *casual* and *registered* separately we see that there is a small correlation of 0.54 for all relation between them and the two variables related to temperature.

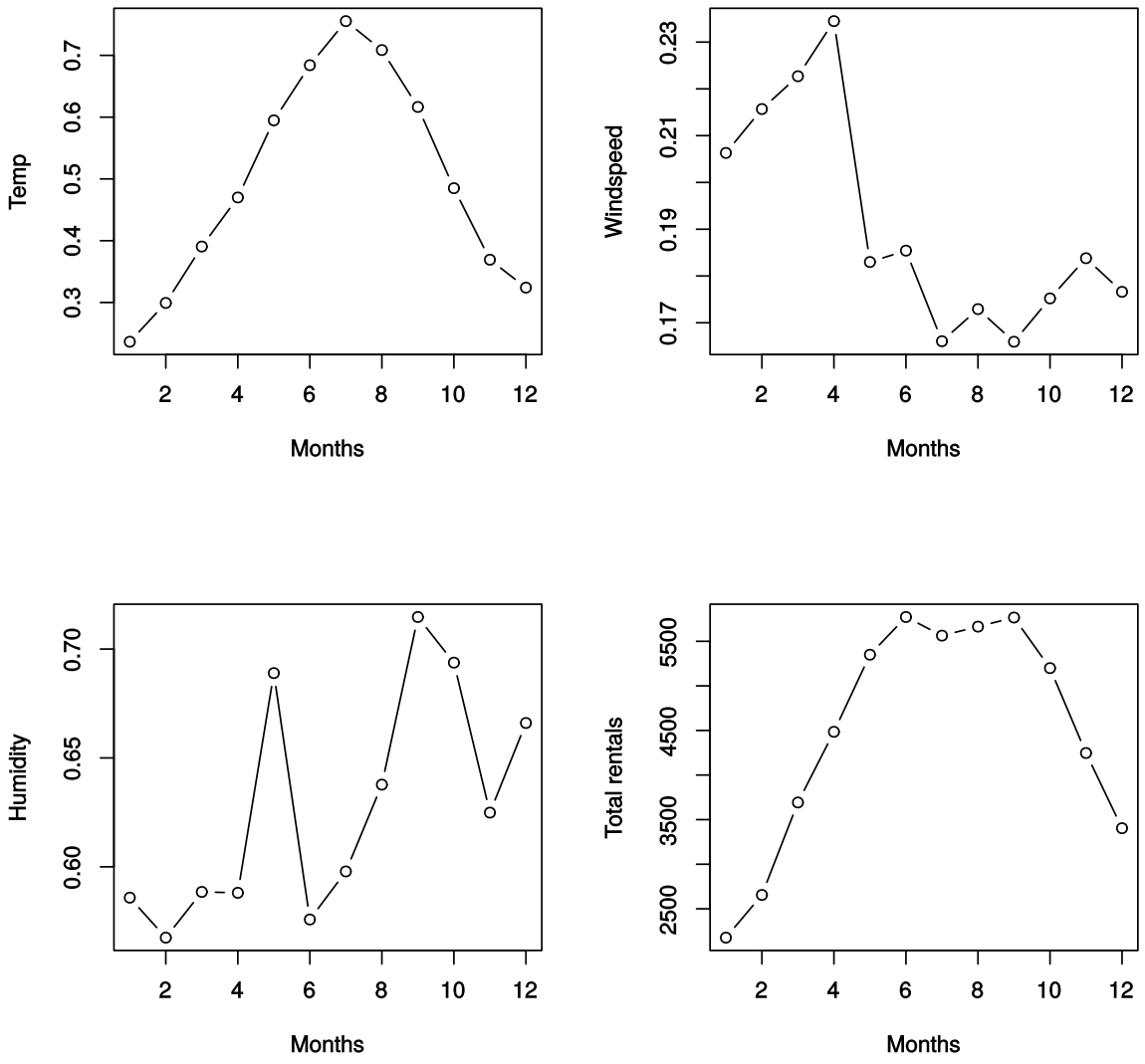

Figure 2: Mean temperature, humidity, windspeed and total rentals per months.

Like it was already noticed, we see that when the temperature reach high values the total rentals also increases. Moreover, we see that when the windspeed decreases the total rentals increases. As well for the humidity, when the latter increases at the 9^{th} month the total rentals begins to decrease.

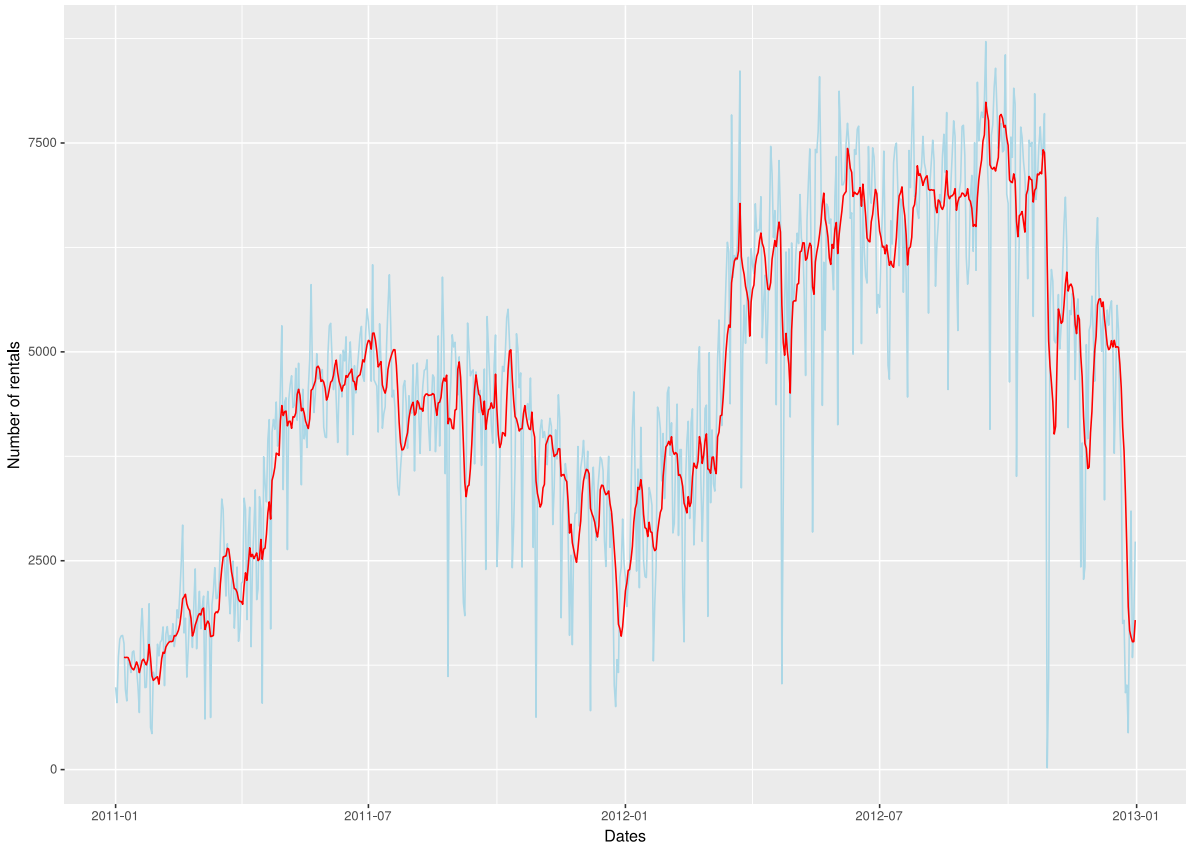

Figure 3: Moving Average smoothing.

The smoothed moving average time series helps us understand the behaviour of bike rentals over the period of two years. The smoothed curve is in red while the original in blue. Like expected the behaviour of both curves are more or less the same but the smoothed one is easier to interpret.

2.2 Decomposing data

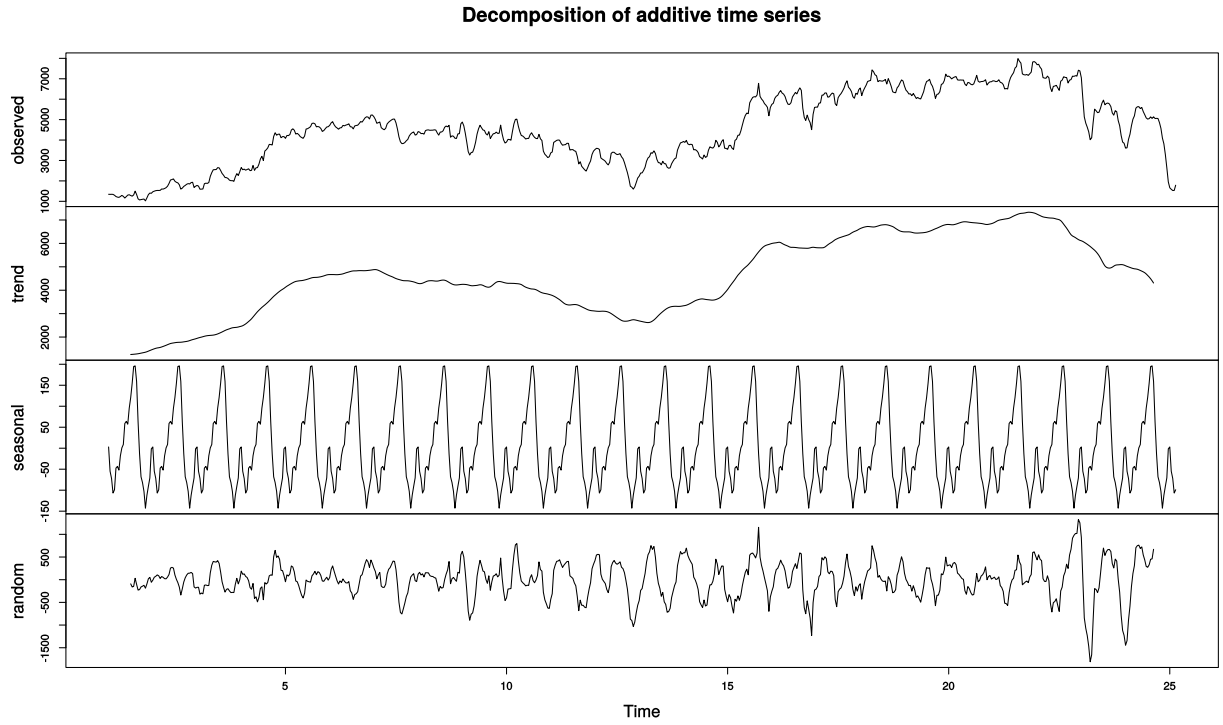

Figure 4: Decomposed smoothed time series.

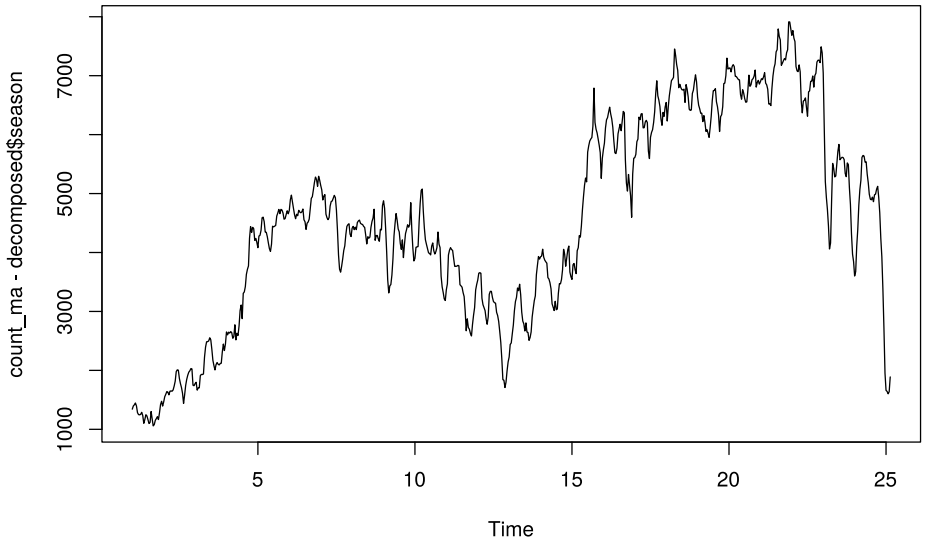

Figure 5: Time series without the seasonal component.

2.3 Stationarity

Dickey-Fuller	-0.64299
Lag order	8
p-value	0.9753
alternative hypothesis	stationary

Table 2: Augmented Dickey-Fuller Test.

For this test, the null hypothesis is the non-stationarity of the series whereas the alternative hypothesis is its stationarity as written in Table 2. Regarding the p-value it seems that the test does not reject the null hypothesis, we can therefore think that the series is non-stationary. We can make stationary the serie by transforming the variable and/or differentiating with ARIMA or SARIMA.

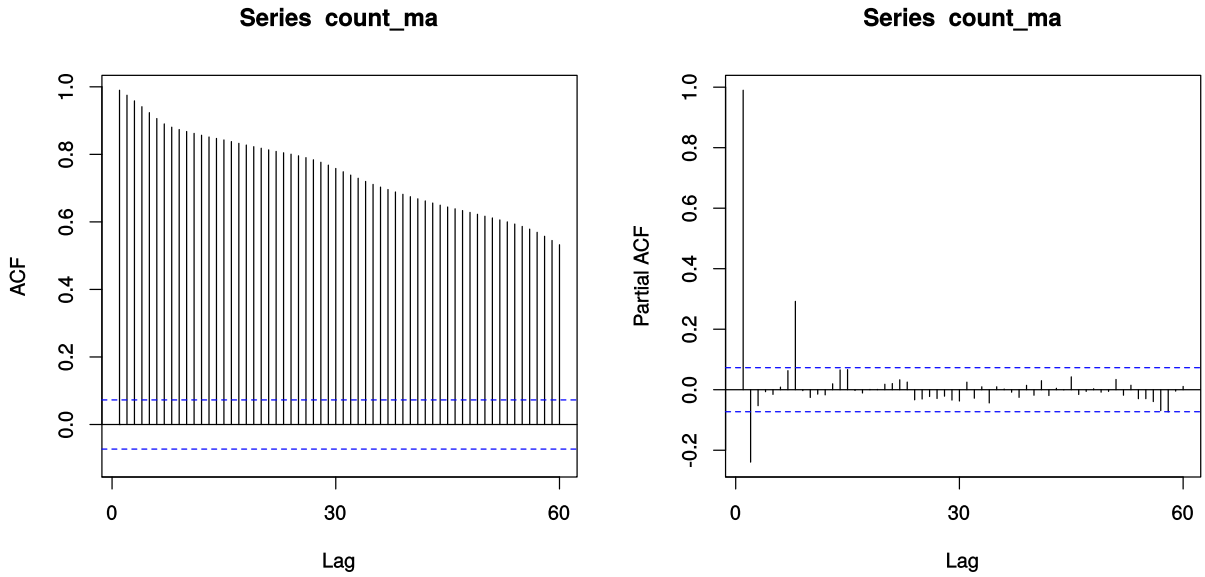

Figure 6: ACF (left) and PACF (right).

The ACF plot confirms that the time series is non-stationary, because there is a decreasing tendency in the plot. The PACF helps us identify the correct number of differences for the ARIMA model.

There are two spikes with significant auto correlations at lags 1, 2 and 7, which makes us believe that we might want to test models with AR or MA components of order 1, 2, or 7. A spike at lag 7 might suggest that there is a seasonal pattern present, perhaps as day of the week.

2.4 Forecasting with ARIMA Models

2.4.1 Fitting ARIMA model

Fitting an ARIMA model with no differenciation, gives us an AIC of 12877.9, much higher than the BIC using the component orders proposed by auto ARIMA (as demonstrated below).

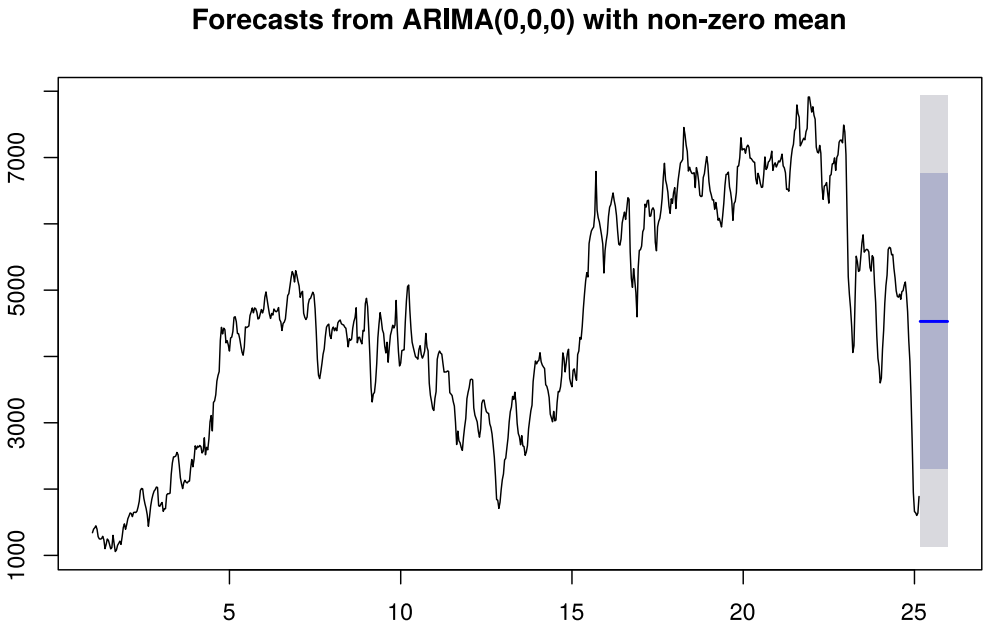

Figure 7: ARIMA model fitted with parameters 0,0,0.

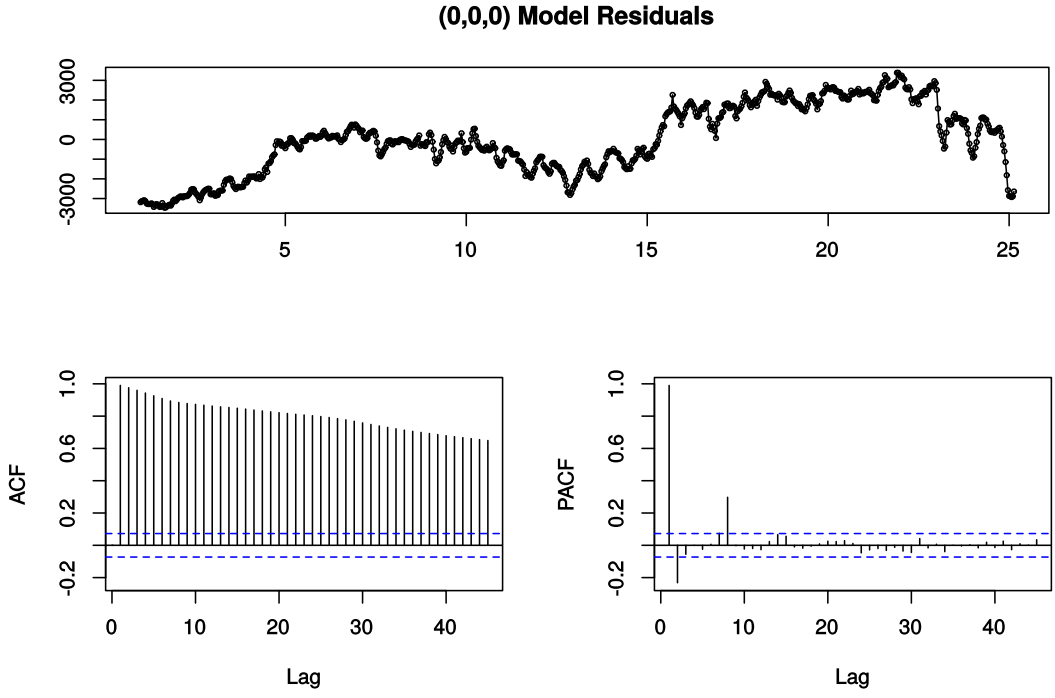

Figure 8: ARIMA model fitted with parameters 0,0,0.

We can see a pattern present in the ACF plot. Based on the PACF and model residuals plots, we can infer a repetition at lag 7, which leads us to a different specification, for example p or q equals 7.

2.4.2 Fit an ARIMA with Auto-ARIMA

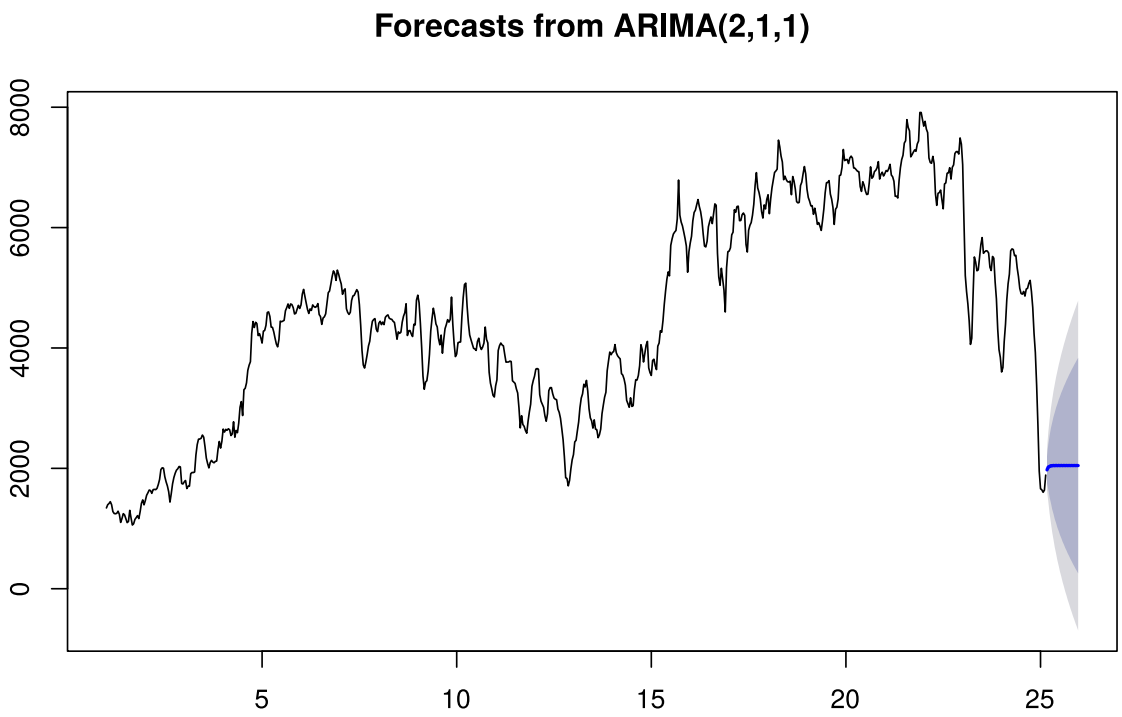

Figure 9: Auto ARIMA model.

Auto ARIMA proposed parameters ARIMA(2,1,1). Using these parameters, we have reduced the AIC parameter to 9537.1, which indicates the model has increased in quality. The residuals have no pattern and are normally distributed. We see that the prediction has not behave correctly, better than the default ARIMA, but it makes a plateau at the last value without really predicting anything.

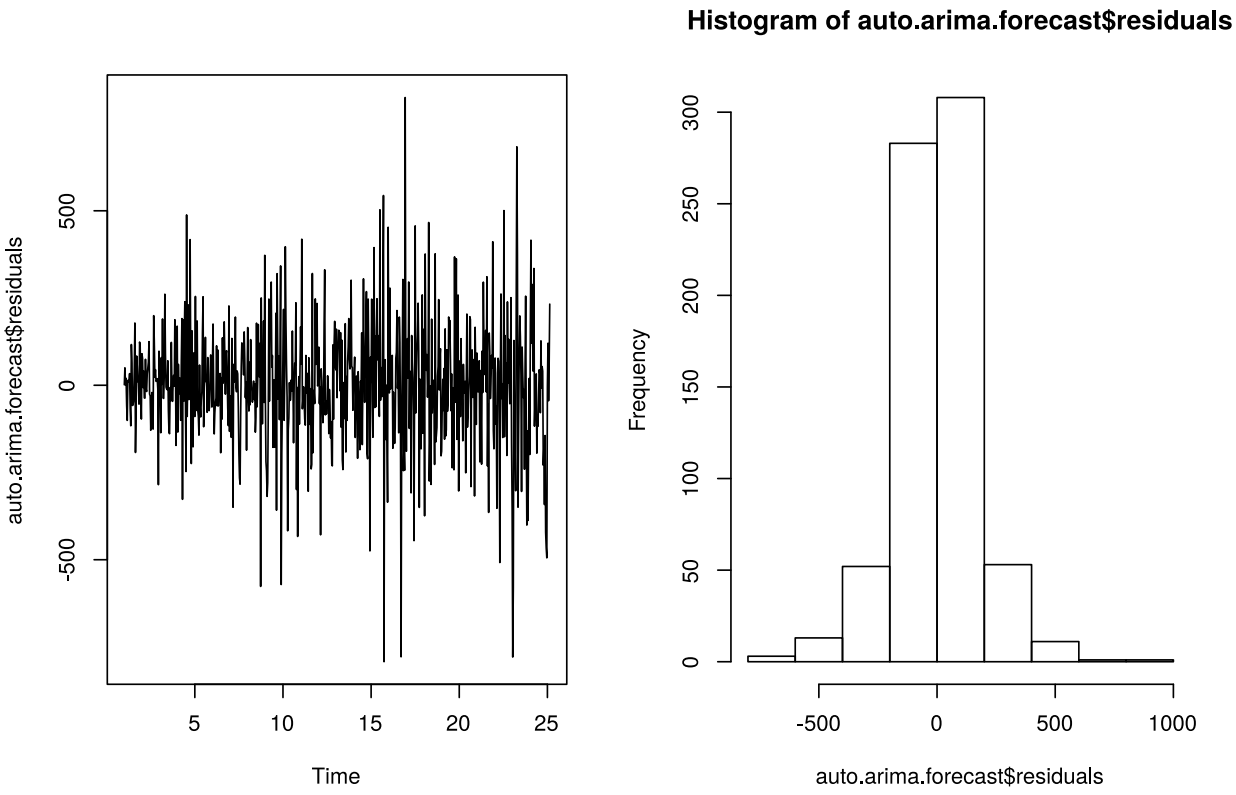

Figure 10: Auto ARIMA model residuals.

2.4.3 Evaluate and Iterate

The next step is to analyse the ACF and PACF plots for model residuals. If model order parameters are correctly chosen, no significant auto-correlations are expected. Thus, we can repeat the fitting process chosing better parameters to increase the model quality. If we use structure (1, 1, 7), as compared to the auto ARIMA structure, we get smaller AIC of 9134.2, which is the goal. We can even go further by choosing (1,2,8) which gives a quite smaller AIC of 9123.72 but will gives more meaningful results as we will see.

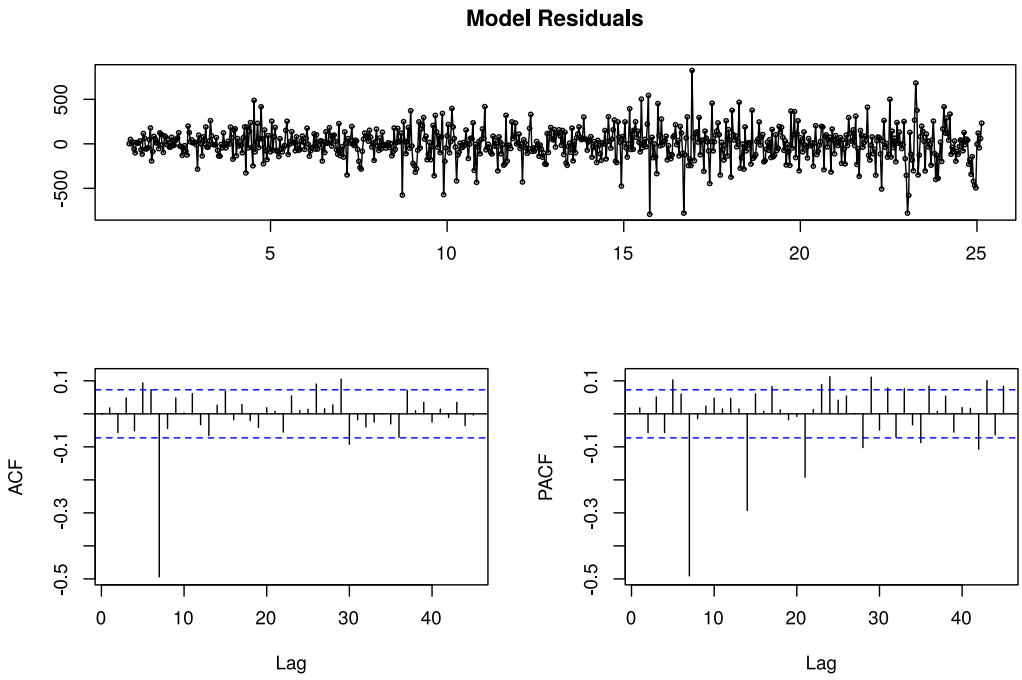

Figure 11: Auto ARIMA model residuals.

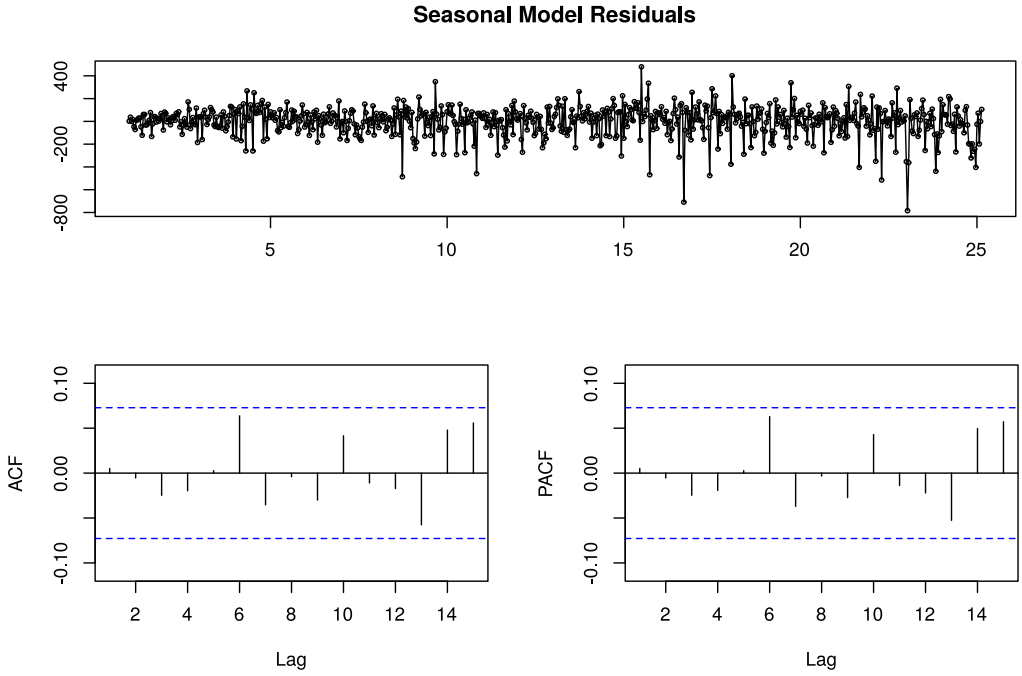

Figure 12: ARIMA model residuals (1,1,7).

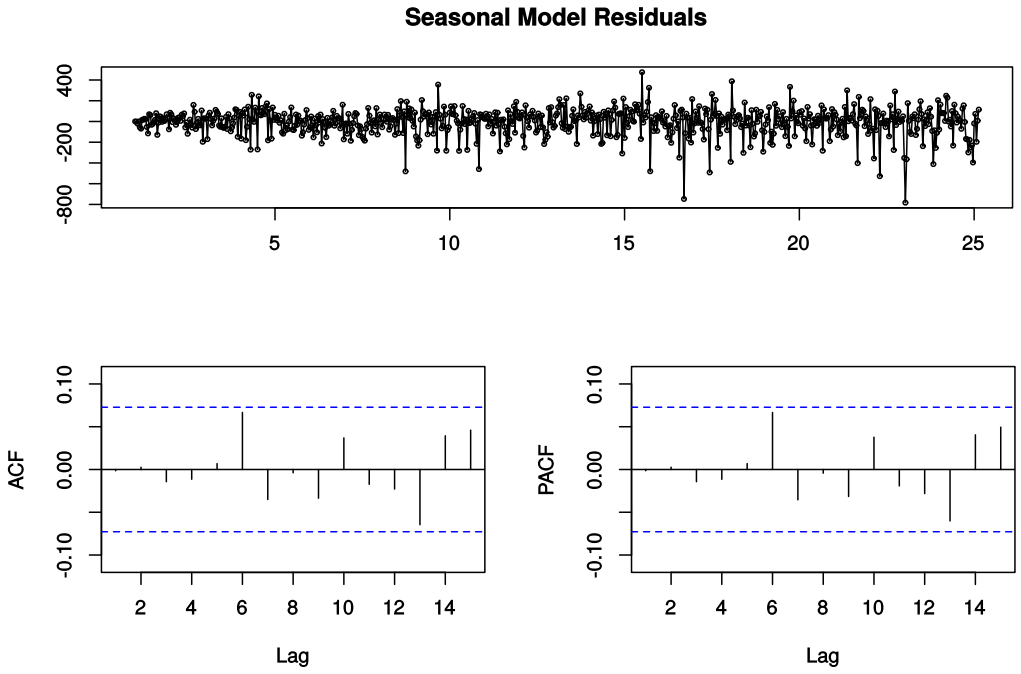

Figure 13: ARIMA model residuals (1,2,8).

As the (1,1,7) and (1,2,8) structures increased the model quality, we have chosen them to forecast. We must notive that these both models gives more or less the same results when we compare the two figures above representing their residuals, ACF and PACF. However, we will see later that (1,2,8) has a better predicting power.

2.4.4 Forecasting

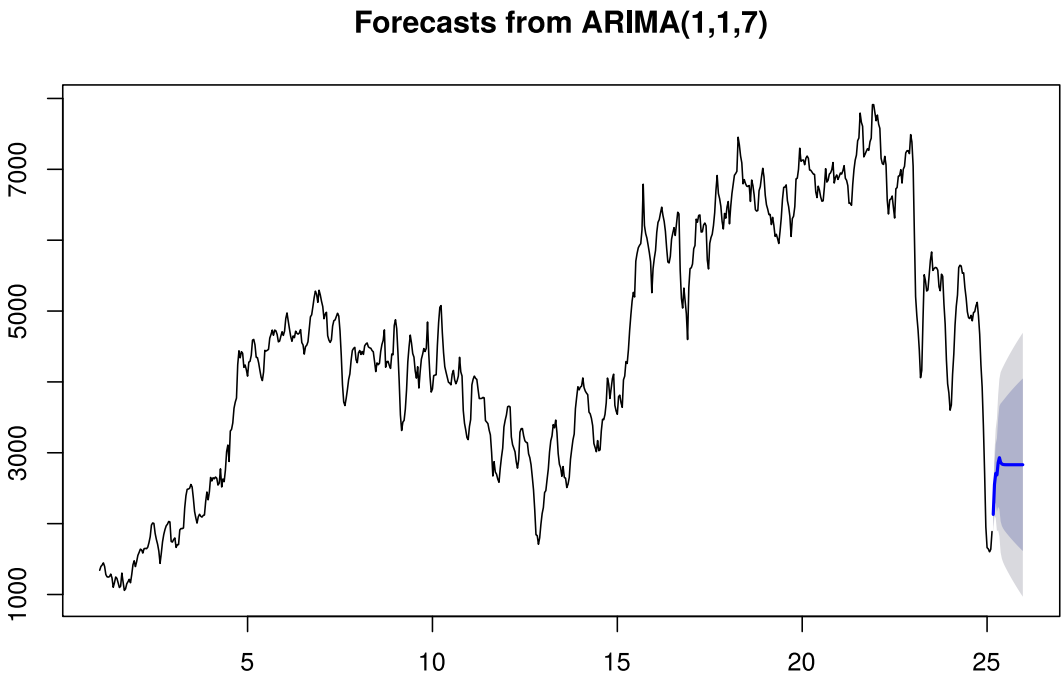

Figure 14: ARIMA forecasting using (1,1,7) structure.

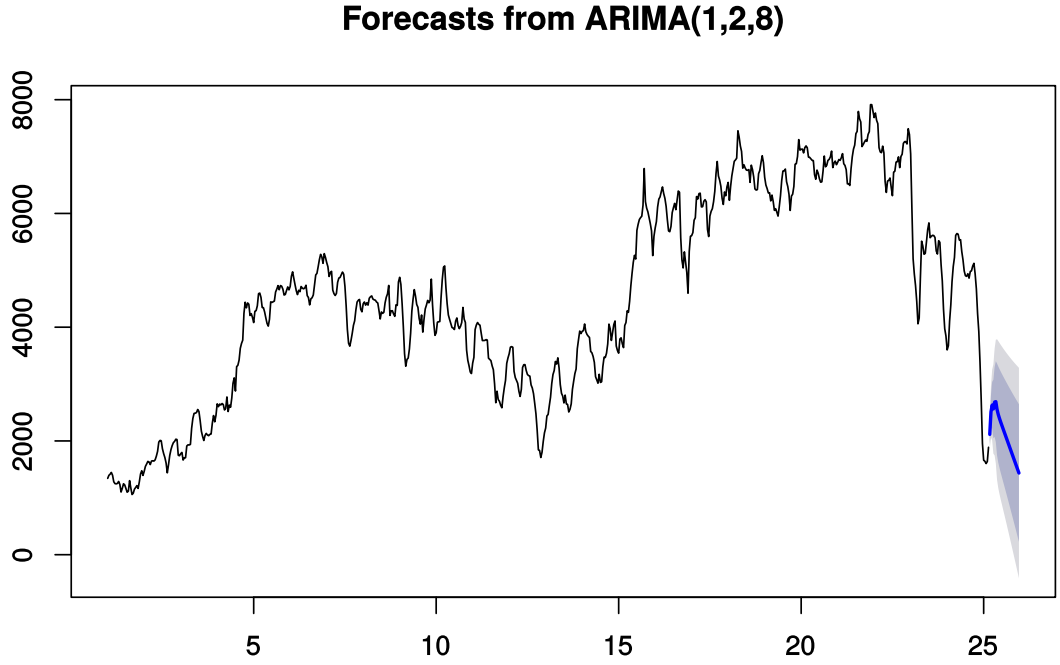

Figure 15: ARIMA forecasting using (1,2,8) structure.

Finally, we have split the time series into two parts: training, which will be used to train the ARIMA and auto ARIMA models, and the test part used to evaluate the forecasts.

As can be seen below, the auto ARIMA created a nearly constant prediction around the moving average. The optimized ARIMA model, based on the evaluation and iteration over the ACF/PACF plots and the structure of the ARIMA model parameters have produced better forecasts, even though neither of them have managed to predict the plummet in bike rentals.

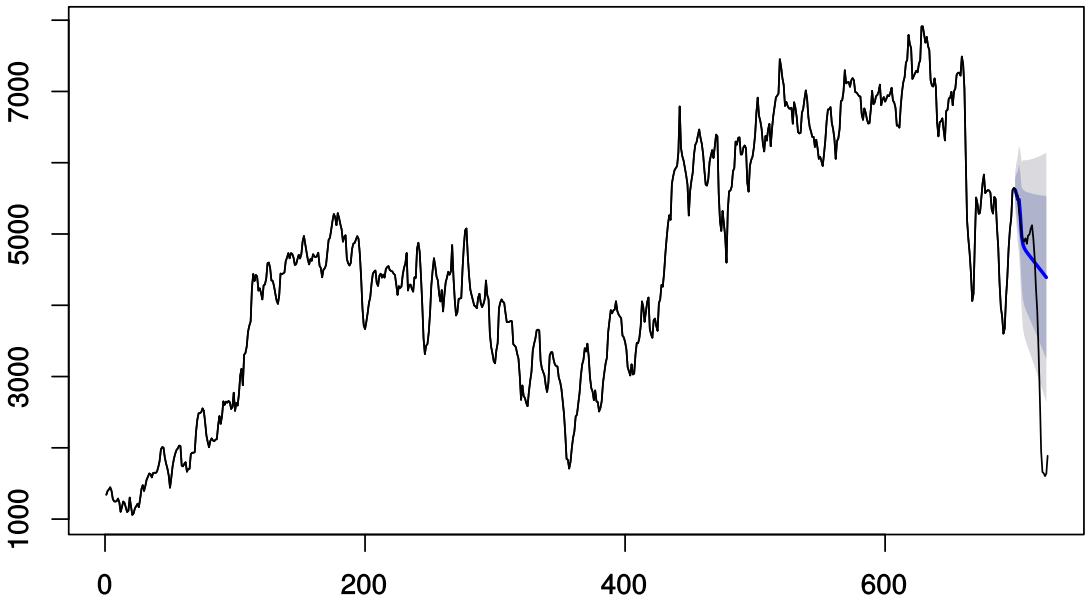

Figure 16: ARIMA forecasting using (1,2,8) structure on training dataset.

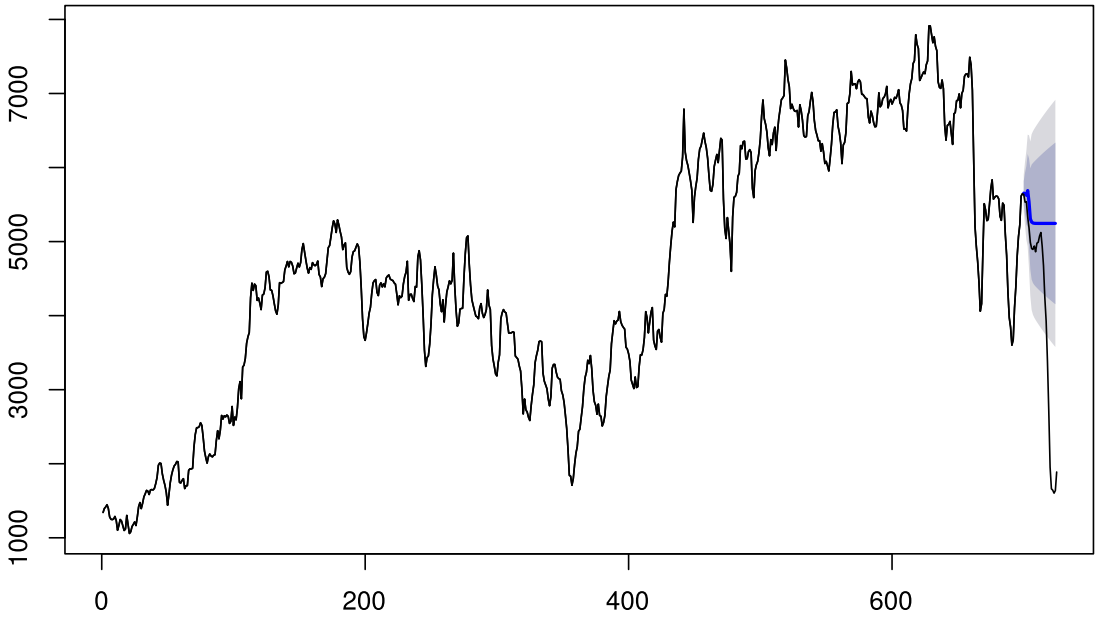

Figure 17: ARIMA forecasting using (1,1,7) structure on training dataset.

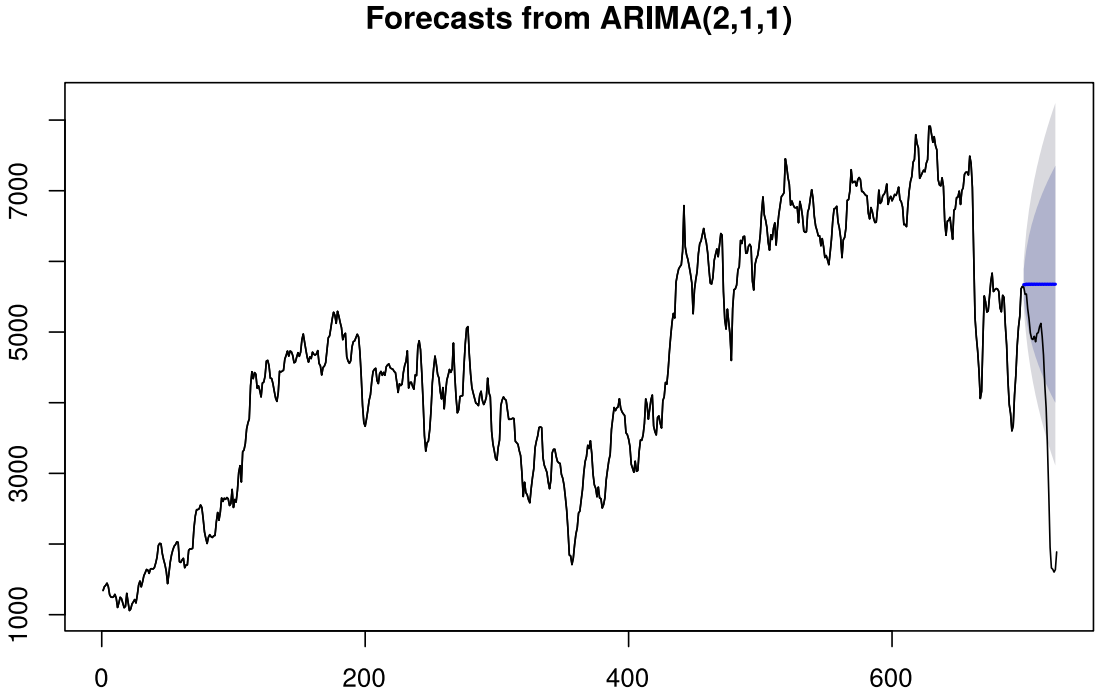

Figure 18: AUTO ARIMA forecasting on training dataset.

When we compare between the predicted data and the original data we see that both ARIMA(2,1,1) and (1,1,7) suffer from the problem of predicting a plateau even if (1,1,7) predict correctly the first values where (2,1,1) makes a plateau at the last known value. ARIMA(1,2,8) seems to be more pertinent by predicting correctly more values than (1,1,7) and by having a behaviour more accurate to the true behaviour, with a decreasing slope.

2.4.5 Models comparison

Model	AIC	Log-likelihood
ARIMA(0,0,0)	12877.9	-6436.95
AUTO ARIMA(2,1,1)	9537.1	-4764.55
ARIMA(1,1,7)	9134.2	-4558.1
ARIMA(1,2,8)	9123.72	-4551.86

Table 3: Models comparison.

3 Conclusion

From this project we have learned how important it is to analyze and get to know the time series before jumping to conclusions. As we demonstrated, the auto ARIMA will not always find the optimal model to fit. Through evaluating and iterating it is possible to find better model structures even if we found that ARIMA struggles to have a good predicting power.

Finally, our github repository is open source and free to use[2].

[2]https://github.com/lucarp/bikeSharing_Project