Spatio-Temporal Data Analysis Project

2020-04-29

Patterns in foreign sims connected to OpenWiFi-Milan

Author: Bernardi Riccardo - 864018

Contents

1 Introduction & Motivation

The dataset that I've chosen is about the presence of foreign smartphone's sims to the OpenWifi of the Municipality of Milan. This data is open and available on the website data.gov.it. The reasons why I would like to go further with this project is that I strongly believe that are present seasonalities that can be interesting to be analysed but also can be more interesting to relate the outliers to some events that happened in the past with a certain mediatic relevance. In practice I would like to both analyse trend and seasonalities to know in which months there are more foreign people and if the trend is increasing in time and both search for outlier peaks to be related to important happenings in the Milan city. Finally I would like to forecast the possible presences in the new year in the city of Milan.

2 The Data

The dataset comes from the open data provided by all the municipalities of Milan. This repository is available at dati.gov.it. From this repository I selected the data going from January of 2018 to October of the 2019.

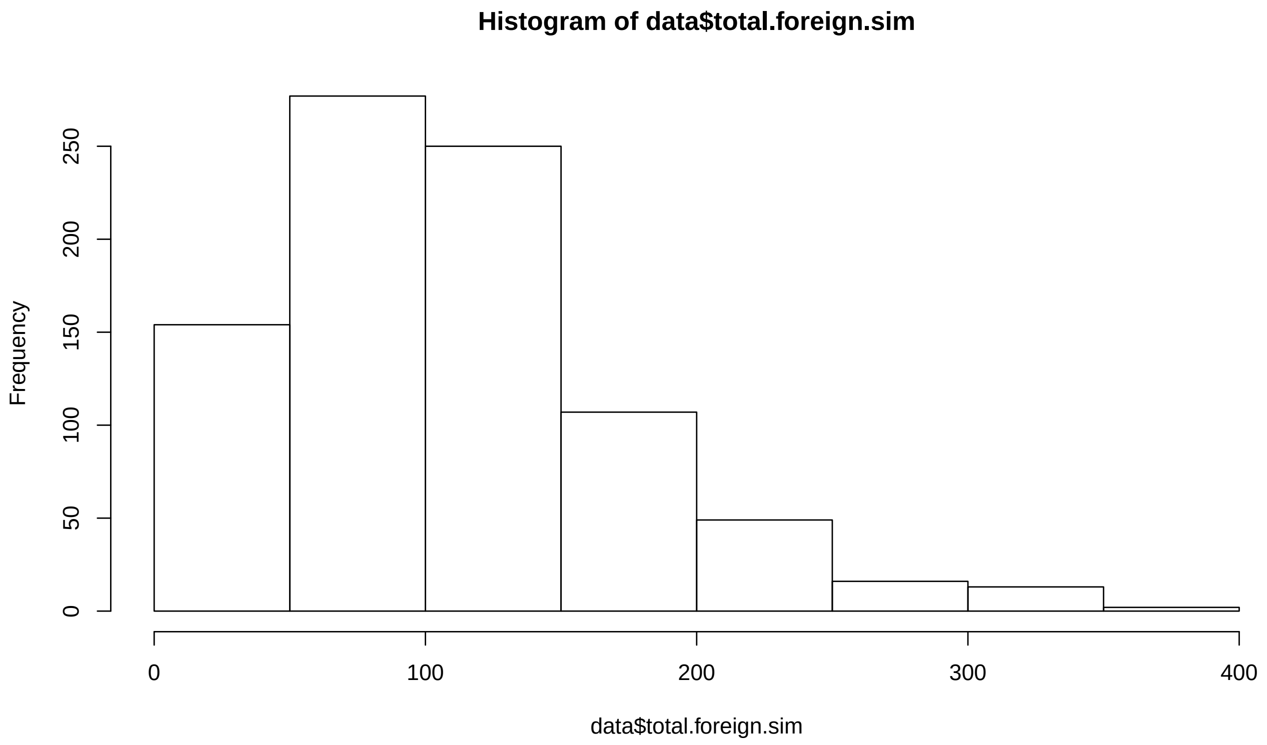
Characteristics of the DataSet:

- the dataset contains 2 columns "Date, Number_of_Foreign_Sims"
- has 658 rows
- Dates goes from from 01/01/18 to 30/10/19 (~2 years)
- the datasets have no NA
- no lacking days
- the "Number_of_Foreign_Sims" is a discrete variable about total number of foreign sims in a certain Date connected to the OpenWifi of Milan

3 Exploration of the Data

```
## [1] "minimum, lower-hinge, median, upper-hinge, maximum)"
```

```
## [1]   1.0  61.5 101.0 141.0 378.0
```



4 Preprocessing

Checking Nans

```
## [1] 0
```

```
## [1] 0
```

Checking limit values

```
## [1] 1
```

```
## [1] 378
```

```
## [1] 109.9228
```

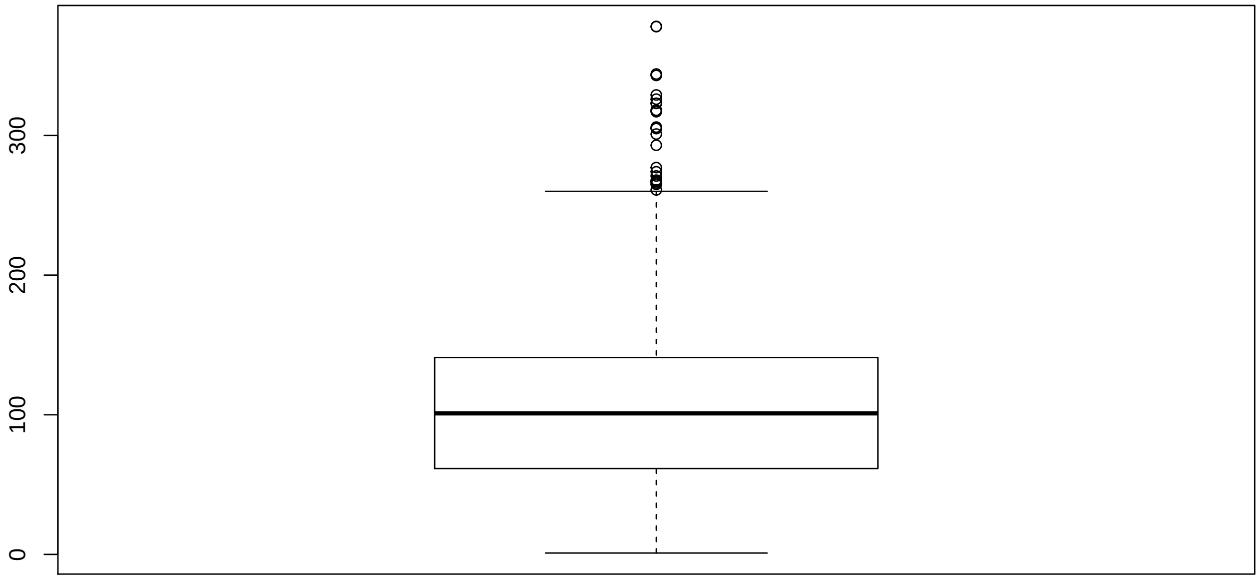
```
## [1] 63.63468
```

Elements that are good in our ts stand between mean±std

```
## [1] 173.5575
```

```
## [1] 46.28813
```

boxplot to check outliers



```
##      0%     25%     50%     75%    100%
##   1.00  61.75 101.00 141.00 378.00
```

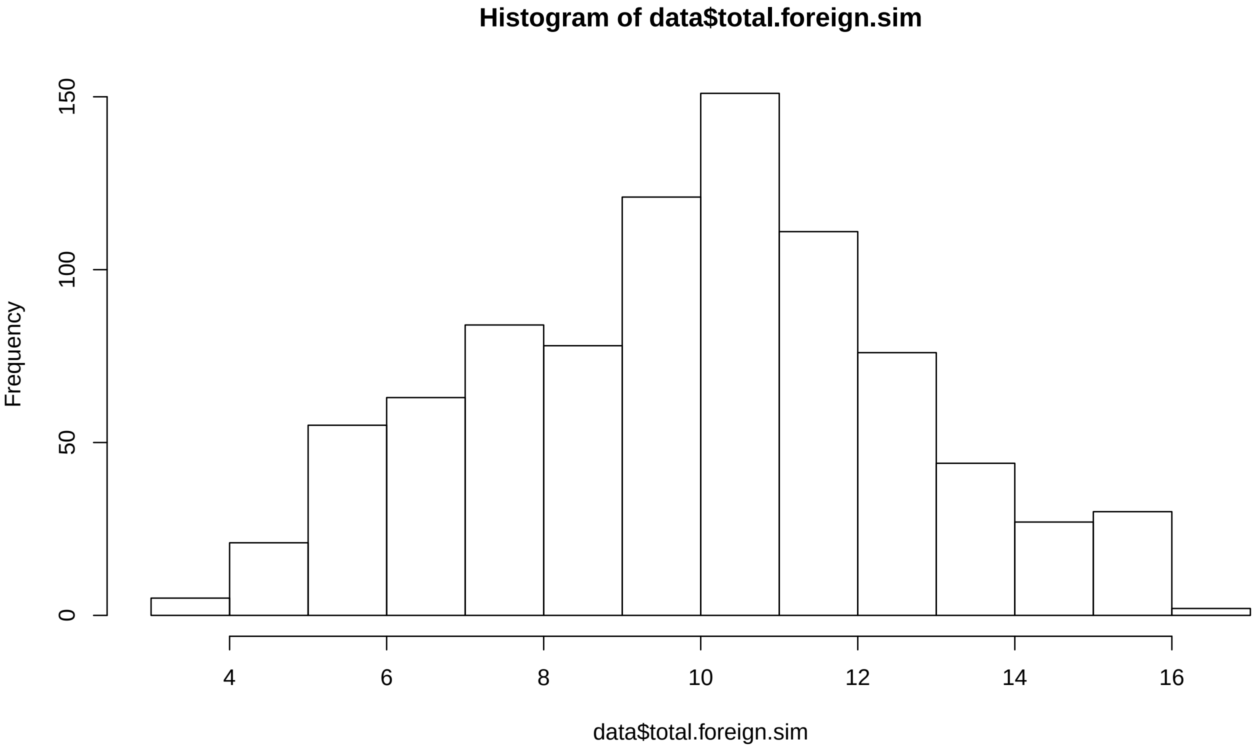
```
##     25%
## -57.125
```

```
##     75%
## 259.875
```

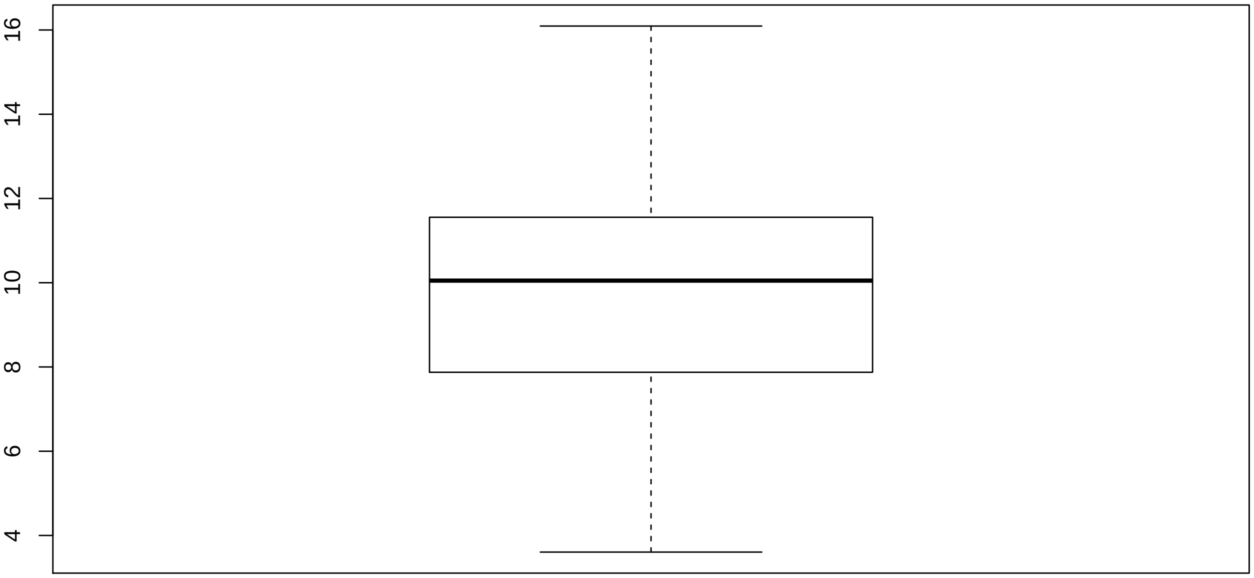
Checking last elements of the serie

5 Using a boxCox transform

6 Hist after the transformation



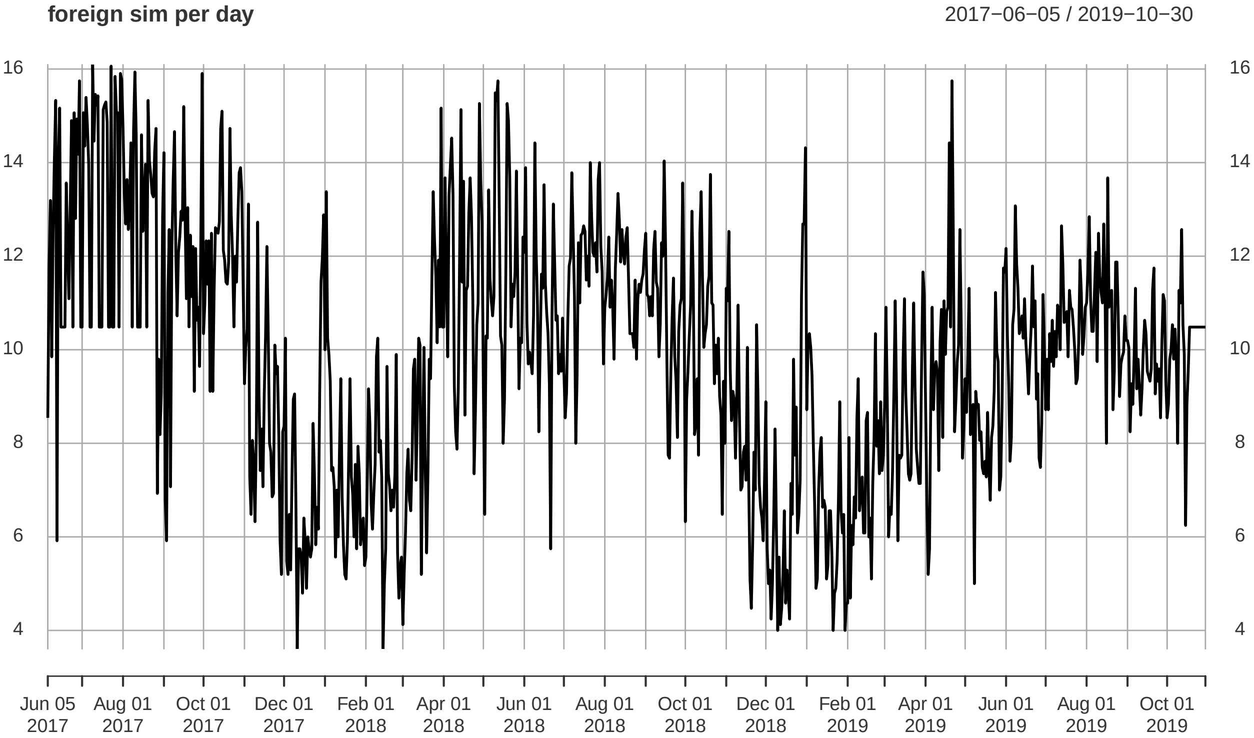
7 Boxplot after the transformation

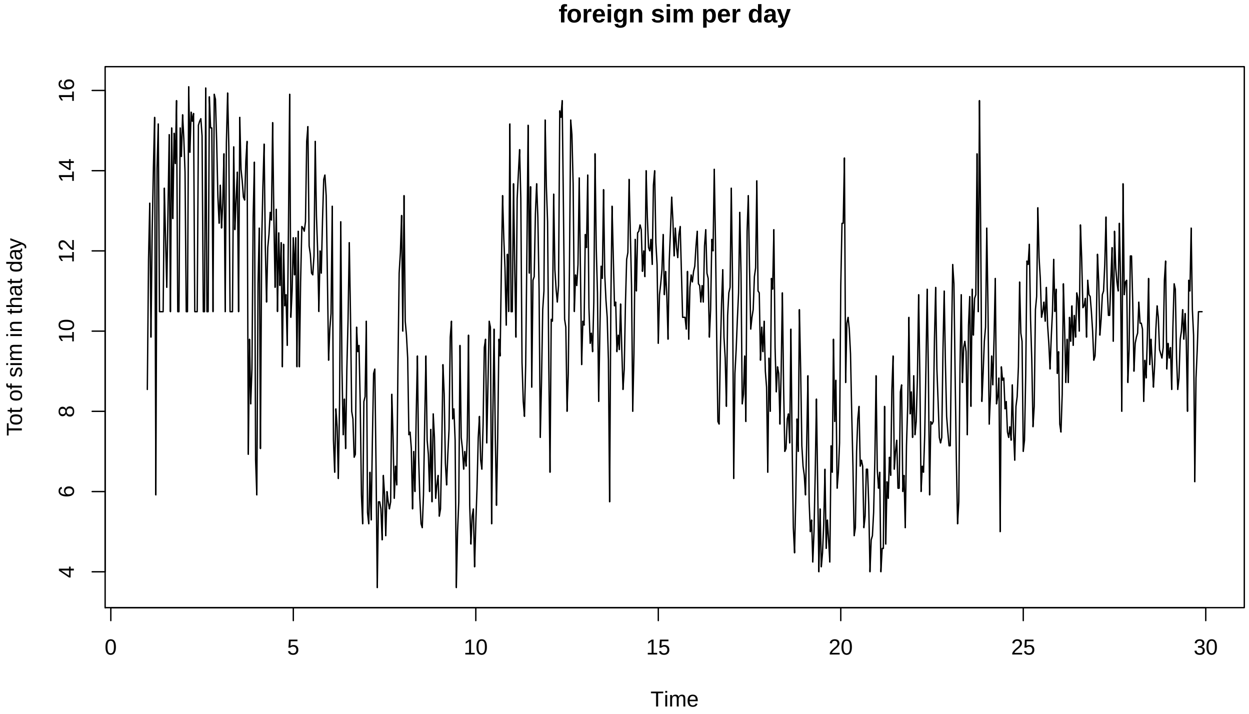


8 Time serie is built

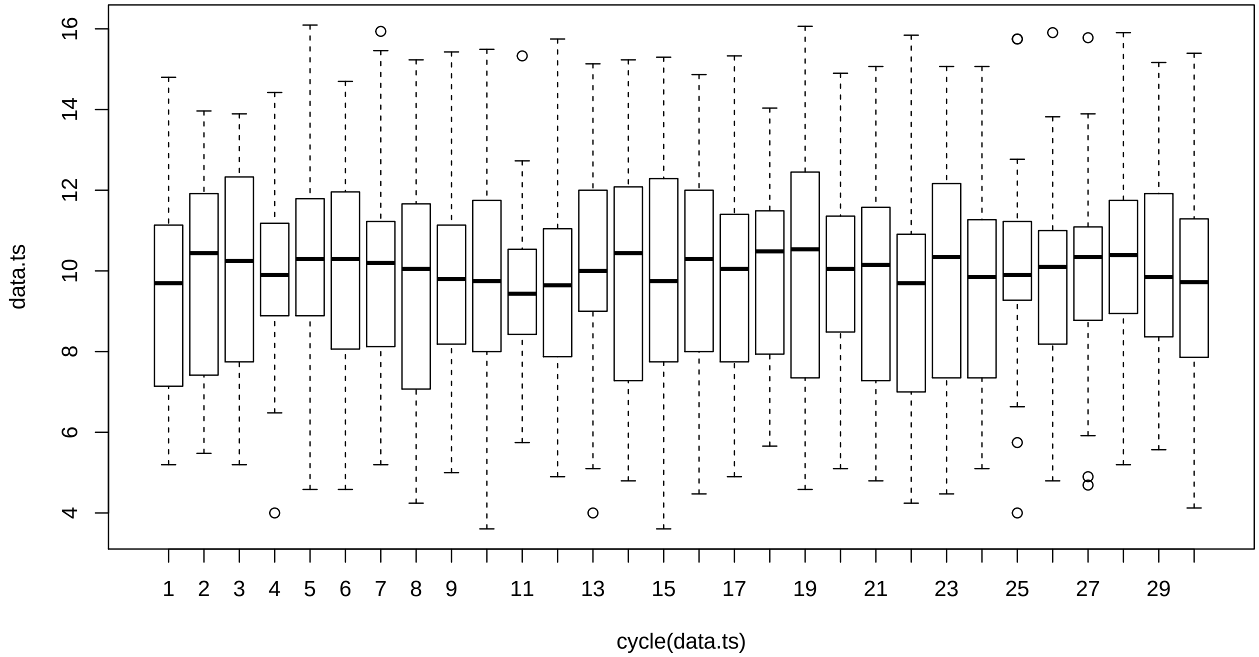
Here the time serie is built

We loaded the dataset from the various datasets aggregating into only one dataset with 655 rows representing 2 years of data gathered. Starting from 05.06.2017 to 30.10.2019. Data is here:





9 Month by month plot



10 Peaks Explanation

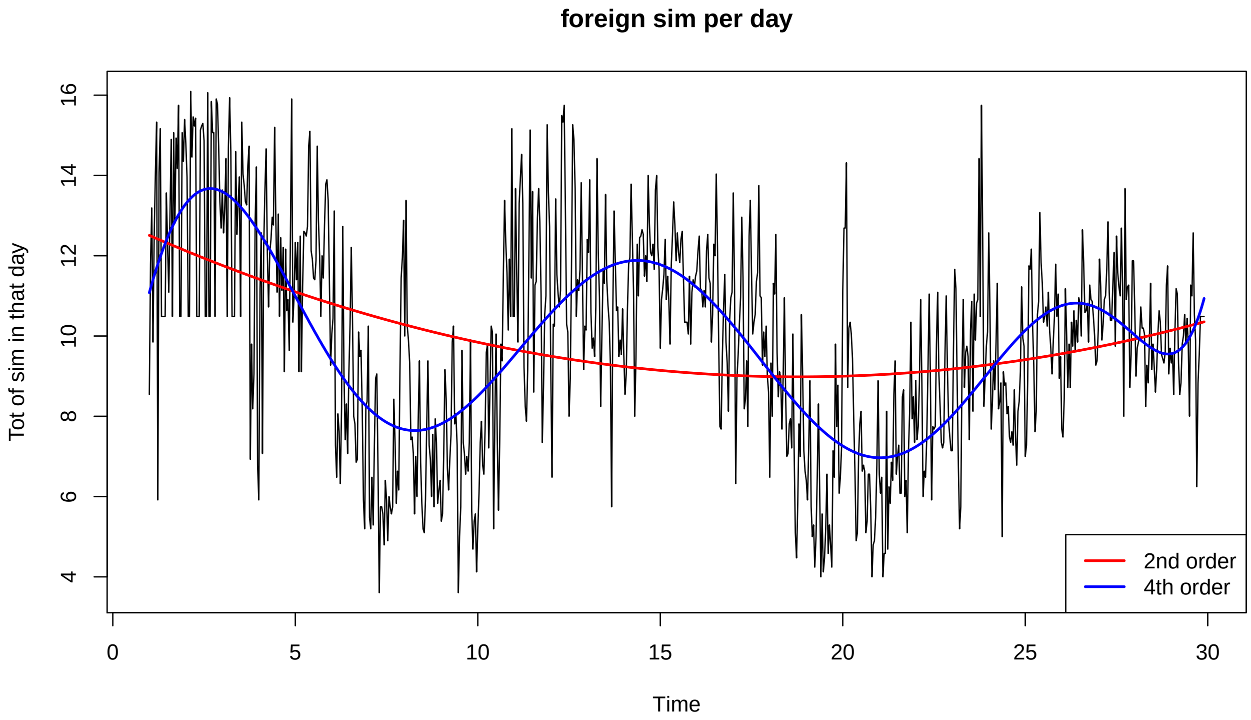
Many peaks are present we would like to exaplin them and to cut them out to be able to predict with a simple arima

- automatic roaming [https://www.mobileworld.it/2017/08/07/roaming-gratis-europa-condizioni-fair-us
- fashion week [https://www.cameramoda.it/it/milano-moda-donna/] february
- fashion week 2017 [https://www.milanoweekend.it/articoli/milano-fashion-week-2017-eventi-programm february

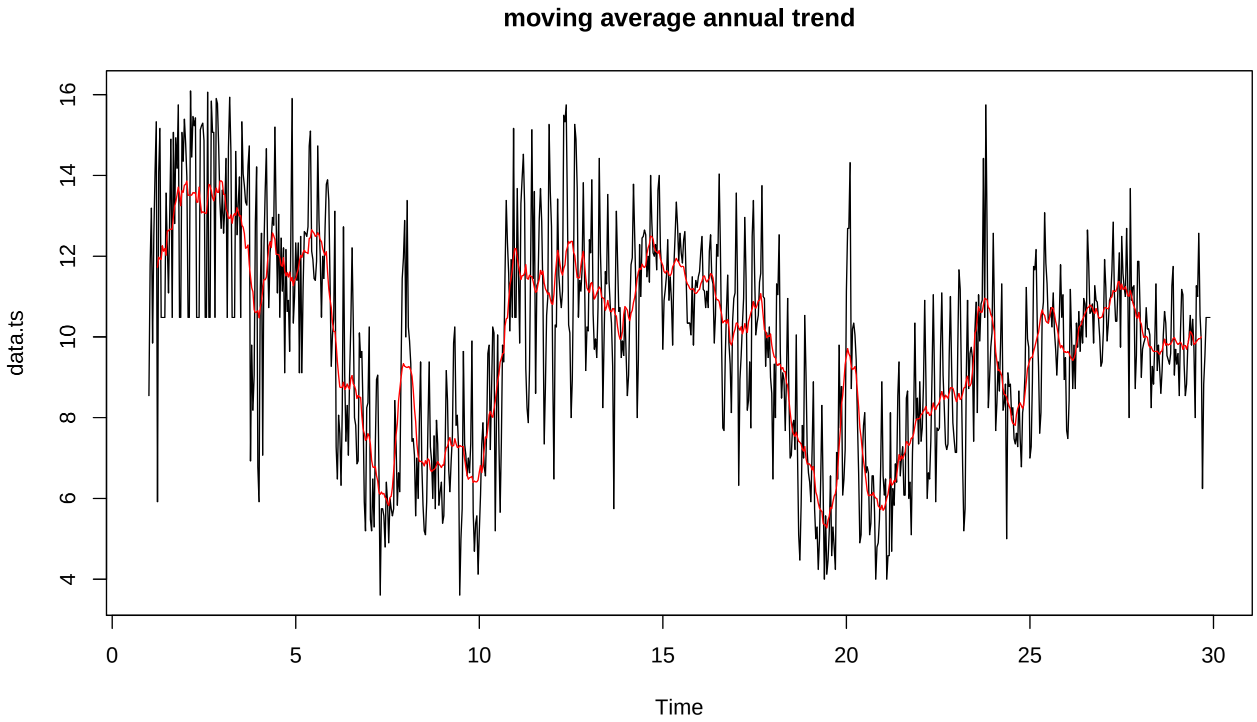
```
## [1] "2017-07-09"
```

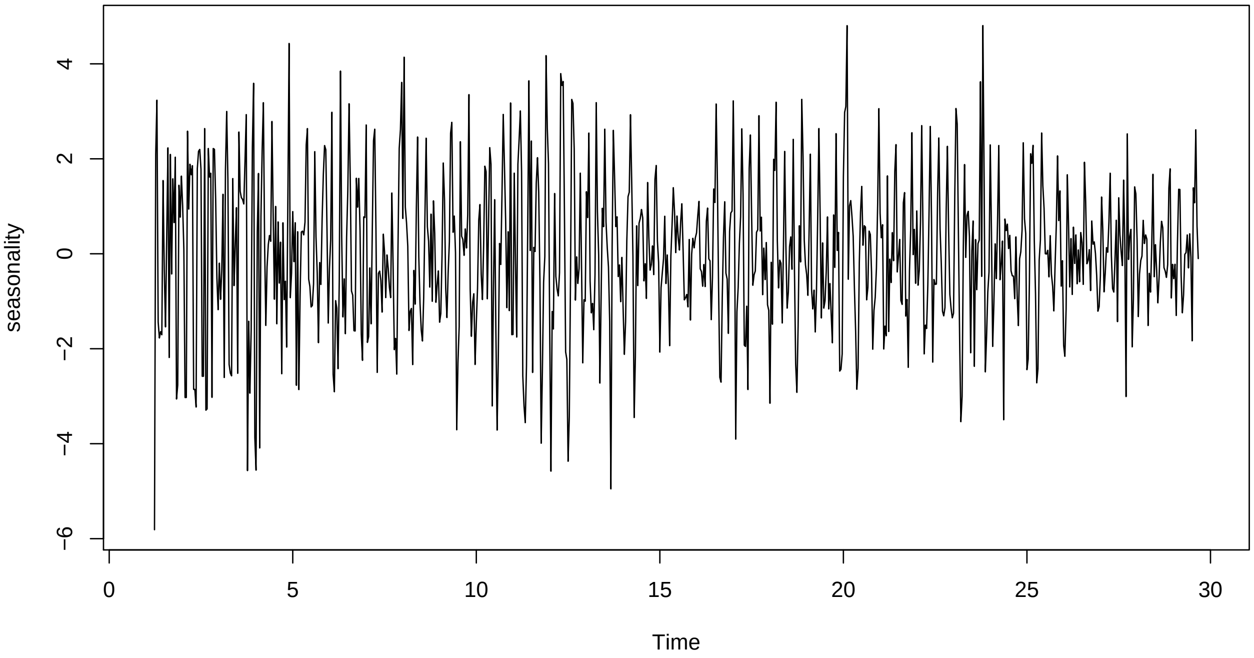
- arch week [https://www.lastampa.it/milano/2017/06/17/news/milano-smart-city-del-futuro-se-ne-par 34584894?refresh_ce]
- it was a saturday!!
- it was the orient festival [https://www.wikieventi.it/milano/index.php?data_selezionata=2017-06-17]
- many mucis events, samsara of papetee and others, folk's festivals, discounts [https://www.wikieventi.it/milano/index.php?data_selezionata=2017-07-22]

11 Trend recognition



12 Smoothing





13 Detrending using LM

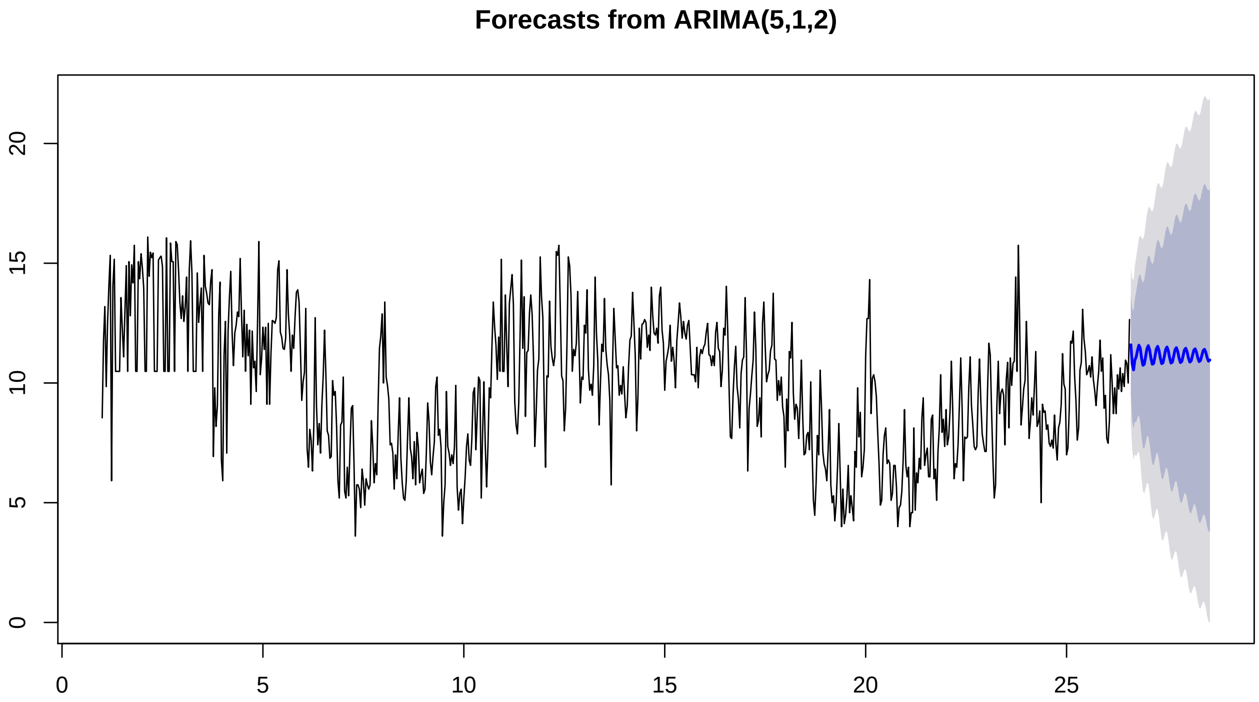
14 Derivative to avoid stationarity

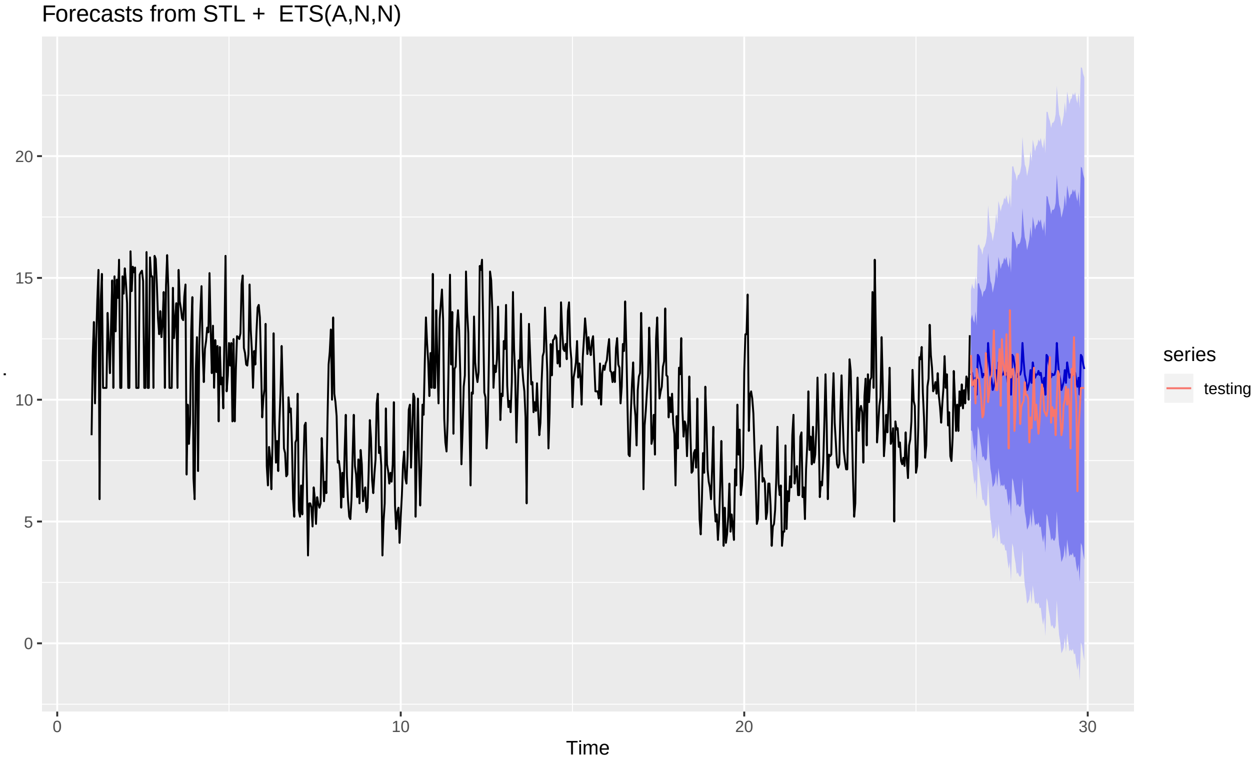
15 Checking stationarity before entering arima

16 Splitting

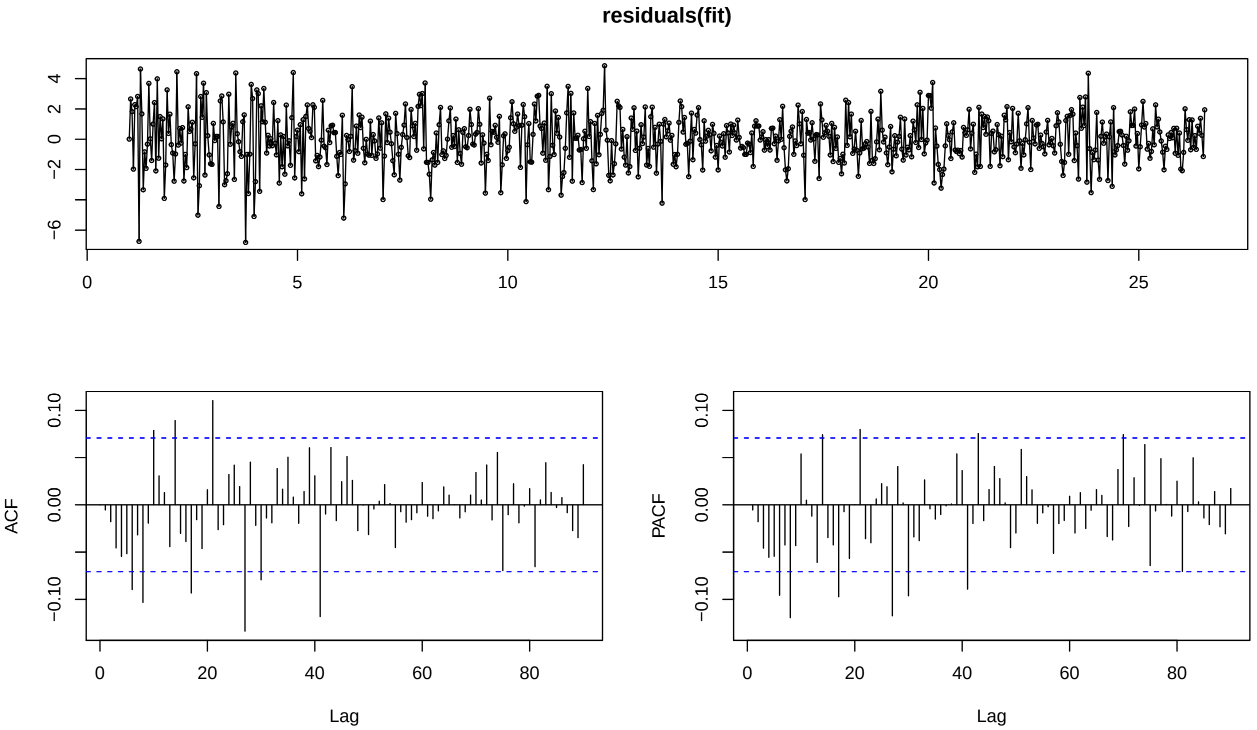
17 Auto Arima

```
##
##  ARIMA(2,1,2)(1,0,1)[30] with drift         : 3080.248
##  ARIMA(0,1,0)            with drift         : 3256.764
##  ARIMA(1,1,0)(1,0,0)[30] with drift         : 3219.467
##  ARIMA(0,1,1)(0,0,1)[30] with drift         : 3157.311
##  ARIMA(0,1,0)                               : 3250.127
##  ARIMA(2,1,2)(0,0,1)[30] with drift         : 3074.433
##  ARIMA(2,1,2)            with drift         : 3078.275
##  ARIMA(2,1,2)(0,0,2)[30] with drift         : 3078.718
##  ARIMA(2,1,2)(1,0,0)[30] with drift         : 3075.845
##  ARIMA(2,1,2)(1,0,2)[30] with drift         : Inf
##  ARIMA(1,1,2)(0,0,1)[30] with drift         : 3079.515
##  ARIMA(2,1,1)(0,0,1)[30] with drift         : 3077.71
##  ARIMA(3,1,2)(0,0,1)[30] with drift         : 3084.969
##  ARIMA(2,1,3)(0,0,1)[30] with drift         : 3088.12
##  ARIMA(1,1,1)(0,0,1)[30] with drift         : 3076.012
##  ARIMA(1,1,3)(0,0,1)[30] with drift         : 3081.478
##  ARIMA(3,1,1)(0,0,1)[30] with drift         : 3073.748
##  ARIMA(3,1,1)            with drift         : 3079.351
##  ARIMA(3,1,1)(1,0,1)[30] with drift         : 3079.623
##  ARIMA(3,1,1)(0,0,2)[30] with drift         : 3078.335
##  ARIMA(3,1,1)(1,0,0)[30] with drift         : 3075.38
##  ARIMA(3,1,1)(1,0,2)[30] with drift         : Inf
##  ARIMA(3,1,0)(0,0,1)[30] with drift         : 3165.755
##  ARIMA(4,1,1)(0,0,1)[30] with drift         : 3073.161
##  ARIMA(4,1,1)            with drift         : 3077.643
##  ARIMA(4,1,1)(1,0,1)[30] with drift         : 3079.436
##  ARIMA(4,1,1)(0,0,2)[30] with drift         : Inf
##  ARIMA(4,1,1)(1,0,0)[30] with drift         : 3074.272
##  ARIMA(4,1,1)(1,0,2)[30] with drift         : Inf
##  ARIMA(4,1,0)(0,0,1)[30] with drift         : 3143.605
##  ARIMA(5,1,1)(0,0,1)[30] with drift         : 3052.699
##  ARIMA(5,1,1)            with drift         : 3054.381
##  ARIMA(5,1,1)(1,0,1)[30] with drift         : 3059.198
##  ARIMA(5,1,1)(0,0,2)[30] with drift         : 3058.938
##  ARIMA(5,1,1)(1,0,0)[30] with drift         : 3053.283
##  ARIMA(5,1,1)(1,0,2)[30] with drift         : Inf
##  ARIMA(5,1,0)(0,0,1)[30] with drift         : 3075.742
##  ARIMA(5,1,2)(0,0,1)[30] with drift         : 2992.022
##  ARIMA(5,1,2)            with drift         : 2990.584
##  ARIMA(5,1,2)(1,0,0)[30] with drift         : 2991.7
##  ARIMA(5,1,2)(1,0,1)[30] with drift         : 2998.189
##  ARIMA(4,1,2)            with drift         : Inf
##  ARIMA(5,1,3)            with drift         : 2992.598
##  ARIMA(4,1,3)            with drift         : Inf
##  ARIMA(5,1,2)                               : 2983.944
##  ARIMA(5,1,2)(1,0,0)[30]                    : 2985.059
##  ARIMA(5,1,2)(0,0,1)[30]                    : 2985.381
##  ARIMA(5,1,2)(1,0,1)[30]                    : 2991.548
##  ARIMA(4,1,2)                               : Inf
##  ARIMA(5,1,1)                               : 3047.739
##  ARIMA(5,1,3)                               : 2985.956
##  ARIMA(4,1,1)                               : 3071.002
##  ARIMA(4,1,3)                               : Inf
##
##  Best model: ARIMA(5,1,2)
```



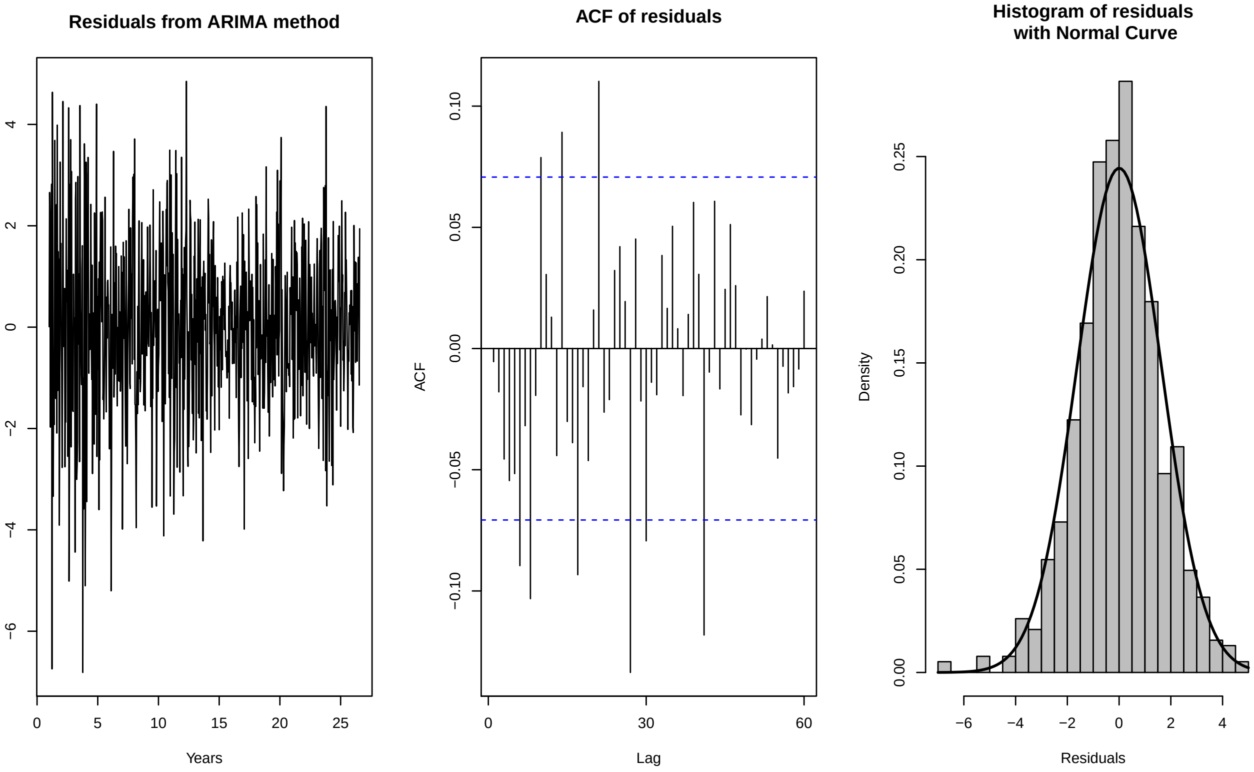


```
##                            ME     RMSE      MAE       MPE     MAPE      MASE
## Training set  0.005344561 1.631300 1.258333 -2.382954 13.86273 0.5049961
## Test set     -0.859054565 1.393915 1.111943 -9.739800 11.78224 0.4462465
##                      ACF1 Theil's U
## Training set -0.005417519        NA
## Test set      0.275986523  1.007212
```



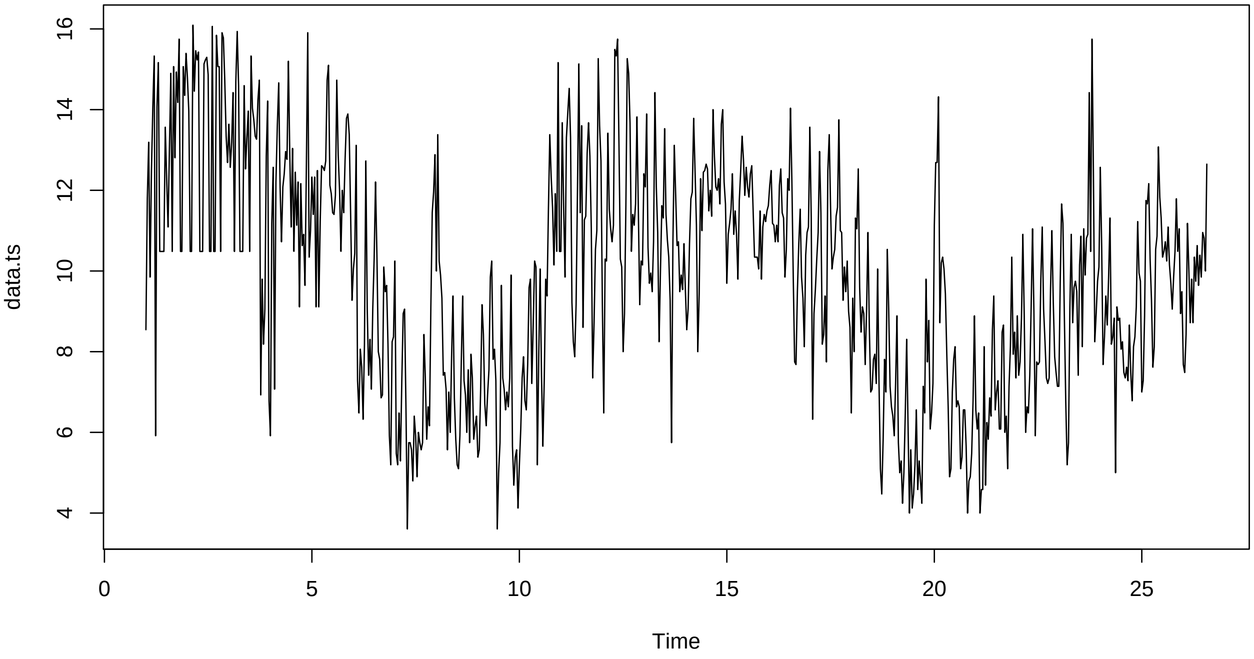
```
##
##  Box-Ljung test
##
## data:  residuals(fit)
## X-squared = 57.473, df = 24, p-value = 0.0001439
```

```
##
##  Box-Pierce test
##
## data:  residuals(fit)
## X-squared = 56.376, df = 24, p-value = 0.0002036
```



```
##
##  KPSS Test for Level Stationarity
##
## data:  diff(data.ts)
## KPSS Level = 0.020484, Truncation lag parameter = 6, p-value = 0.1
```

```
## [1] 1
```

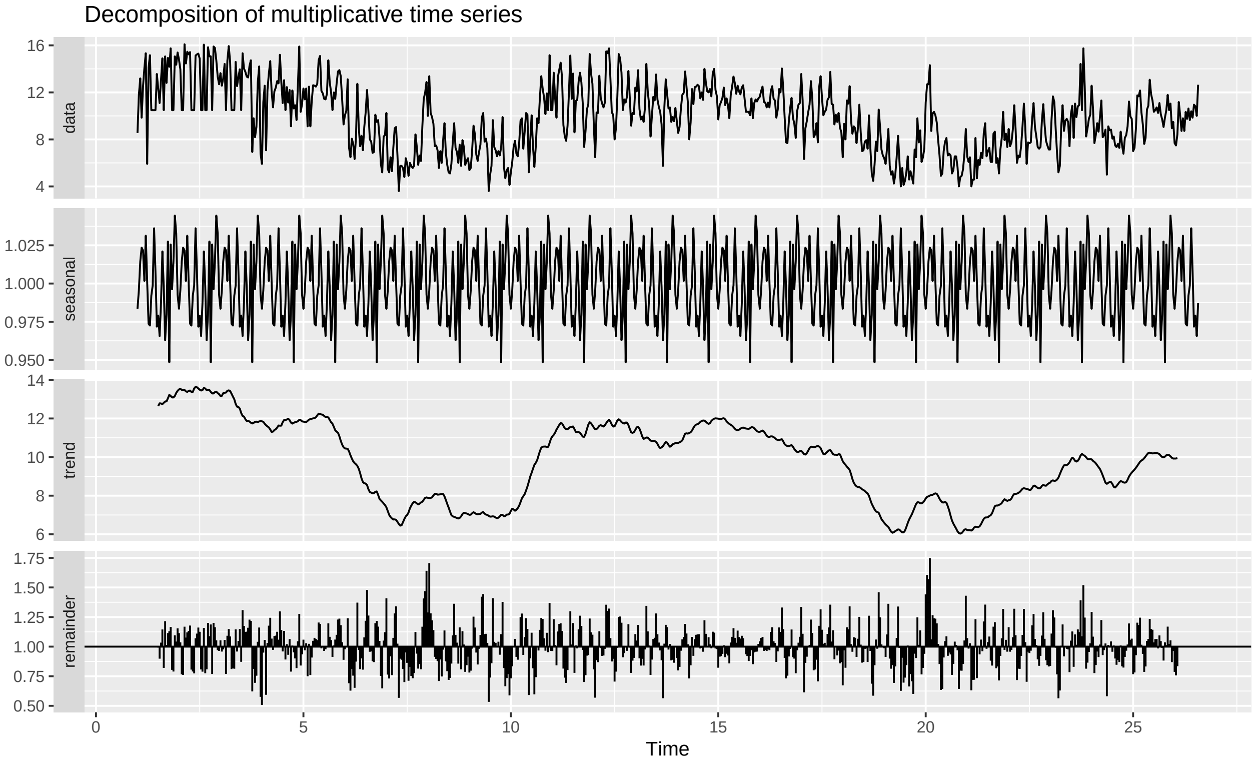


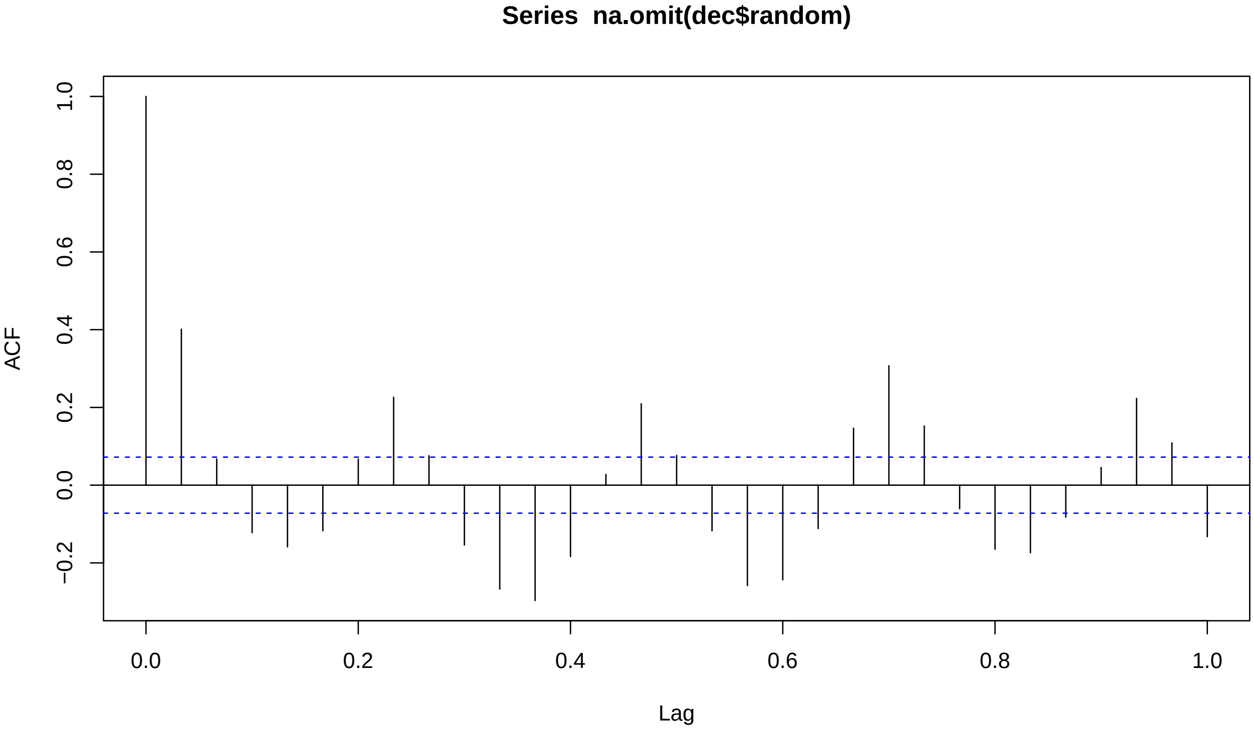
The plot is not good but AIC and BIC are very high, we should try with a multi seasonal decomposition

```
## [1] 30
```

18 Searching for multi seasonalities

without differentiation residuals looks pretty bad





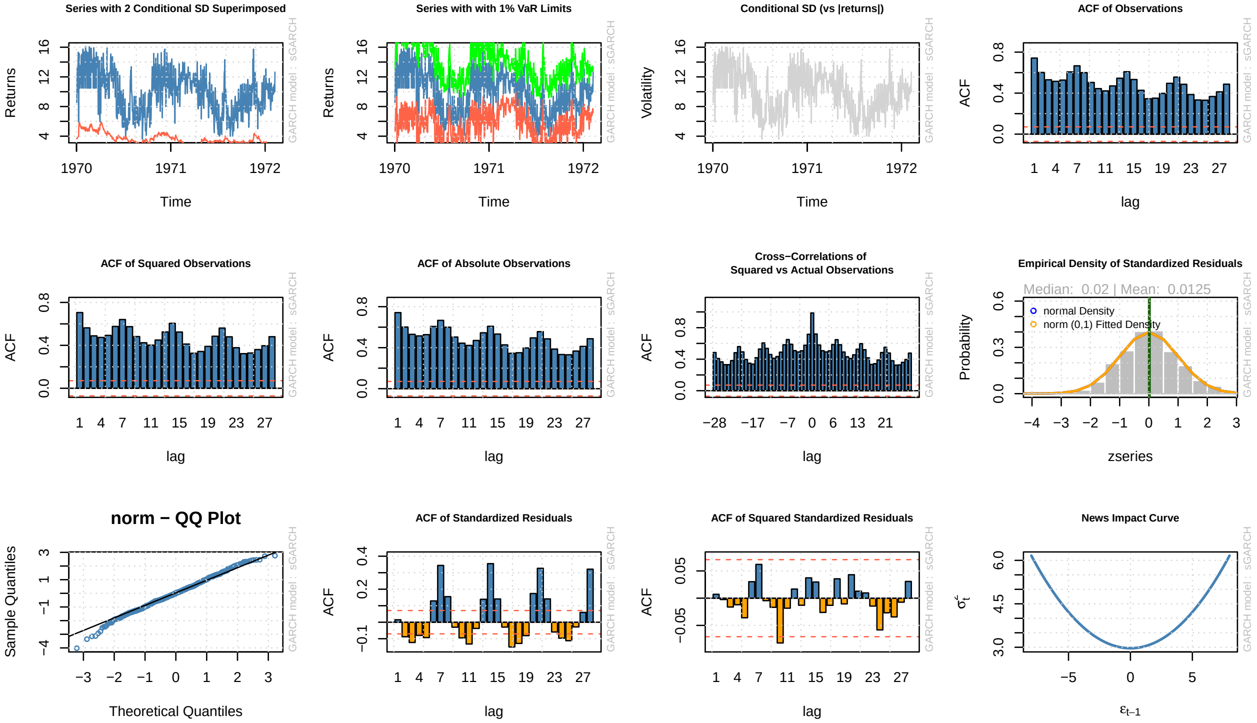
Looks better than before but we can still see every 5(*7) a seasonality/trend left. 5*7 is about a month, probably there is a monthly seasonality

19 RUGARCH

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(1,0,1)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      9.603943    0.311757  30.8059 0.000000
## ar1     0.833914    0.031045  26.8611 0.000000
## ma1    -0.175875    0.065358  -2.6909 0.007125
## omega   0.037851    0.024524   1.5434 0.122727
## alpha1  0.049945    0.014435   3.4601 0.000540
## beta1   0.937934    0.016150  58.0752 0.000000
```

```
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      9.603943    0.362385  26.5020 0.000000
## ar1     0.833914    0.044022  18.9431 0.000000
## ma1    -0.175875    0.096566  -1.8213 0.068563
## omega   0.037851    0.025578   1.4798 0.138918
## alpha1  0.049945    0.014798   3.3752 0.000738
## beta1   0.937934    0.014235  65.8876 0.000000
##
## LogLikelihood : -1535.579
##
## Information Criteria
## ------------------------------------
##
## Akaike       4.0145
## Bayes        4.0508
## Shibata      4.0144
## Hannan-Quinn 4.0285
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.1678   0.682
## Lag[2*(p+q)+(p+q)-1][5]   15.2984   0.000
## Lag[4*(p+q)+(p+q)-1][9]   62.2552   0.000
## d.o.f=2
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                    0.03906  0.8433
## Lag[2*(p+q)+(p+q)-1][5]   0.41044  0.9706
## Lag[4*(p+q)+(p+q)-1][9]   2.17081  0.8837
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]    0.2026 0.500 2.000  0.6526
## ARCH Lag[5]    0.8870 1.440 1.667  0.7667
## ARCH Lag[7]    2.6715 2.315 1.543  0.5776
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  1.5236
## Individual Statistics:
## mu     0.3404
## ar1    0.1234
## ma1    0.5853
## omega  0.1550
## alpha1 0.1763
## beta1  0.2121
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.49 1.68 2.12
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           1.0043 0.3155
## Negative Sign Bias  0.2479 0.8043
## Positive Sign Bias  0.1968 0.8440
## Joint Effect        1.5370 0.6738
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     13.56      0.80857
## 2    30     26.69      0.58856
## 3    40     43.04      0.30235
## 4    50     62.08      0.09939
##
##
## Elapsed time : 0.1395361

##
## please wait...calculating quantiles...
```



20 Conclusions

It was a hard work! I tried a lot of methods to fit the data and obtain good forecasting and good residuals. I tried searching for multi seasonalities, difference some times to reach stationarity, detrending with lm and ma, smoothing. I tried by hand many arima models. I tried to decompose the time serie in many ways. I tried all the possible frequencies that can be thought as valid. Eventually it was really interesting! I experimented a lot. The dataset is brand new, never touched by other data scientists for what i know.

21 TODO

prima diff, poi prima diff seasonal, check acf pacf, check no trend(trend se con decadono a 0 velocemente) identificare i picchi identificare l estate doppia seasonality una settimanale e una annuale ARCH GARCH VAR<—- stabilizzare con trasformazioni