Time Series Analysis of Household Electric Consumption

Time Series Analysis & Forecasting Methods 99106

Haim Elbaz-311141162
Asaf Levi-200967677

Intro and Motivation

Household electricity consumption is a critical aspect of energy management, sustainability, and cost optimization. Analyzing and understanding the patterns and trends of electricity consumption data can provide valuable insights for effective energy planning, load forecasting, and demand management. Time series analysis enables us to explore the temporal characteristics of household electricity consumption and develop models that capture the underlying dynamics. This project aims to use time series analysis techniques to gain a deeper understanding of household electricity consumption patterns and try to achieve a good forecasting for the near future.

Data Description

The dataset was collected from a single household in Seaux, France between December 2006 and November 2010. Each value is the average of the past minute's recorded values.

- Number of lines: 2,075,259
- Number of columns: 9

Columns Info:

1) Date- Date of measurement dd/mm/yyyy

2) Time – time of measurement hh:mm:ss

3) Global_active_power – house overall consumption in kW

4) Global_reactive_power – house overall reactive power in kW

5) Voltage – global voltage(V)

6) Global_intensity – current intensity in Amp

7) Sub_metering_1 - Kitchen power consumption in kW

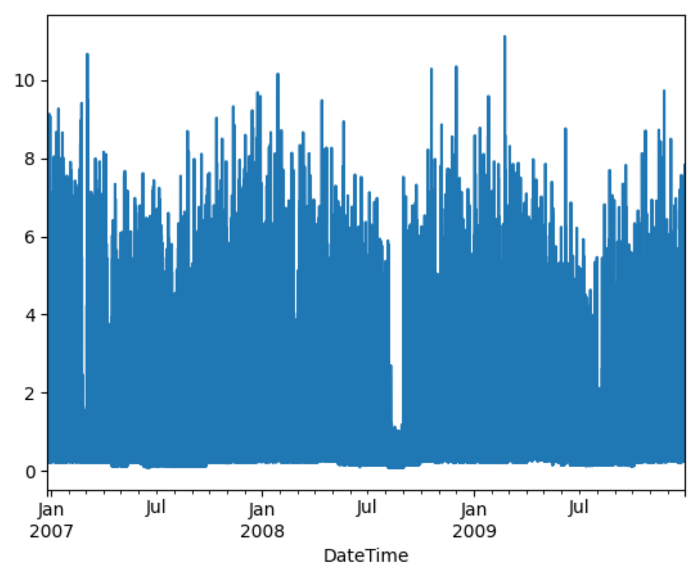
8) Sub_metering_1 – laundry room power consumption in kW

9) Sub_metering_1 – Air conditioner & water heater consumption in kW

Following the article, we will focus on the column "Global_activate_power". Unaggregated and uncleaned, this timeseries is not very informative.

Statistic	Value
Count	1585439
Mean	1.09
Standard Deviation	1.08
Min	0.076
Q1	0.3
Q2	0.56
Q3	1.53
Max	11.12

<u>Global active power time series plot</u>



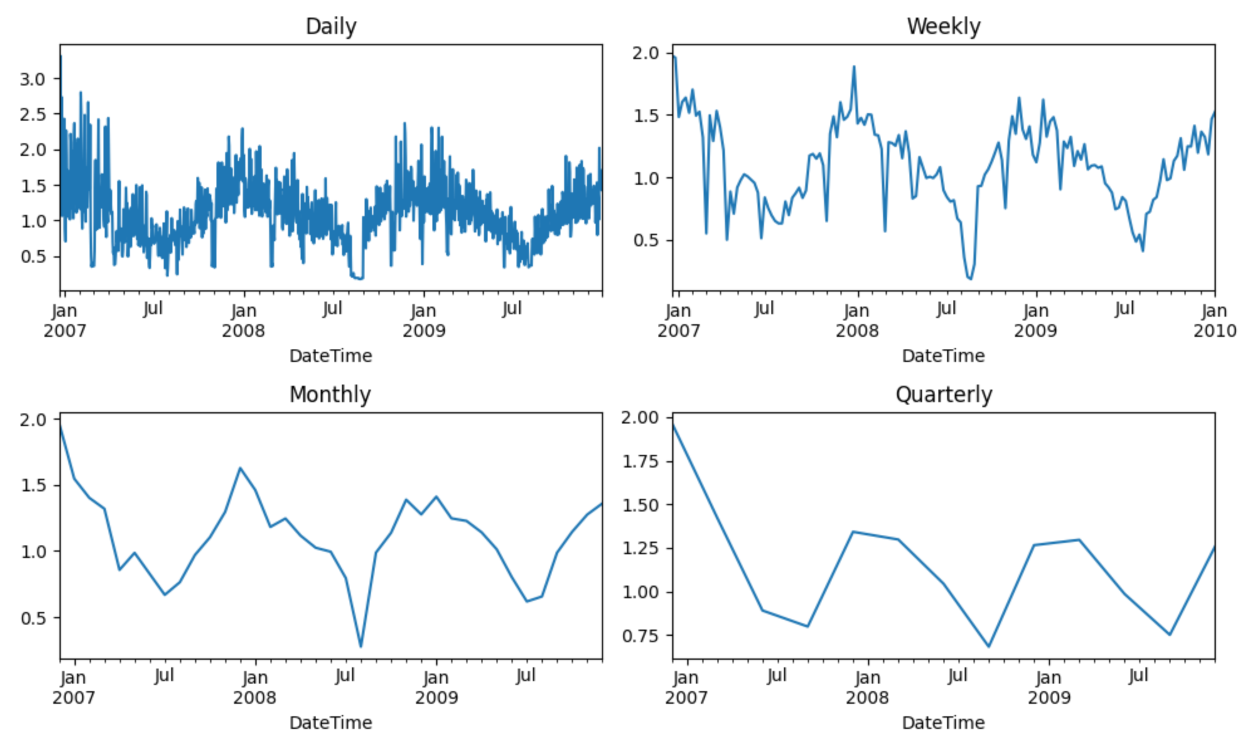
Data Preprocessing & Preparation

- The datafile is originally a text file containing semicolon-separated values. The data-loading step required parsing the date & time data into timestamp format, and indexing by our new, combined, “datetime” column.

- The timeseries contains a considerable amount of NaN values, signified as “?” or “nan” in the original text file. We replaced these values with the forward-fill method – filling a NaN value with the previous value.

- We have scoped our time series between 2006-12-26 and 2009-12-30, as the authors did, and removed the unnecessary columns (all columns besides global consumption). The rest of the data, January to November of 2010, is kept as a test dataset for the models’ prediction.

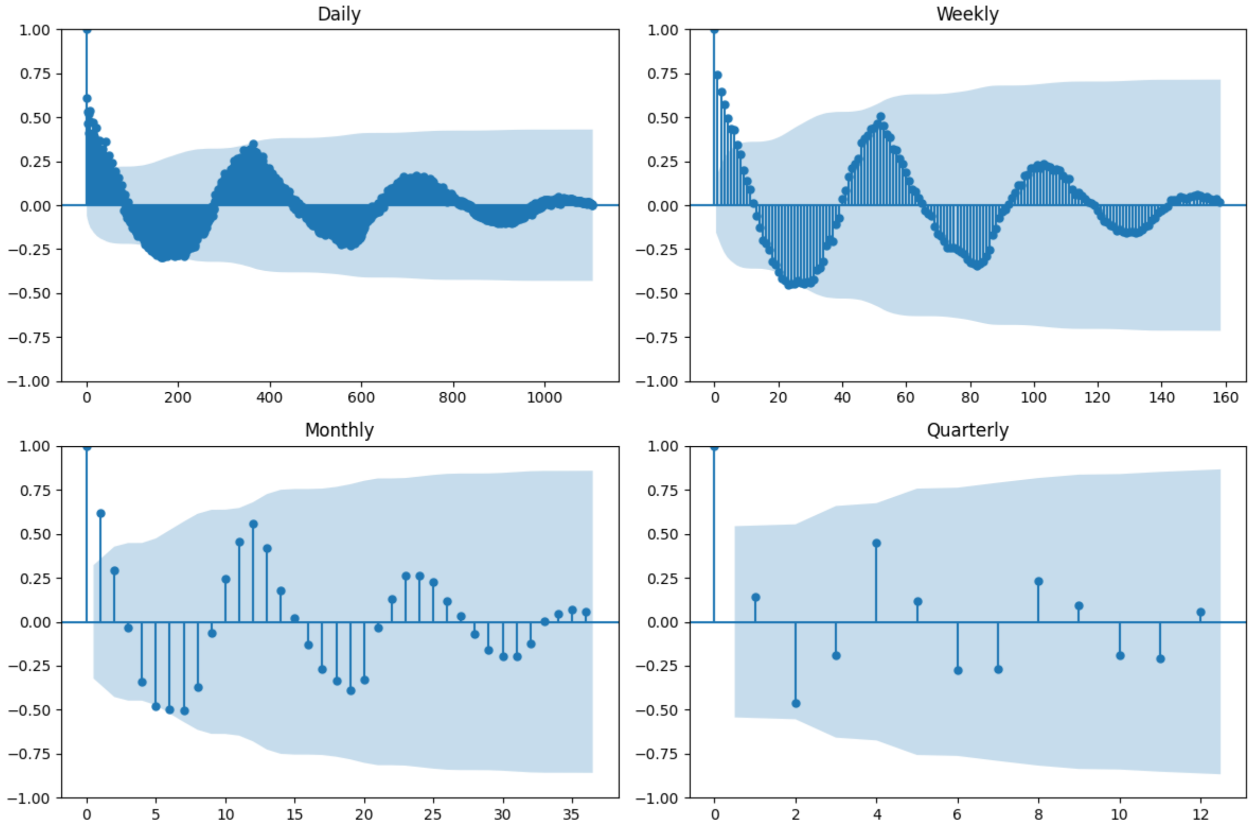
- We then changed the data format to the appropriate time intervals, and aggregated our data into four different datasets, by daily, weekly, monthly and quarterly means.



Methods & Results

Auto-Correlation

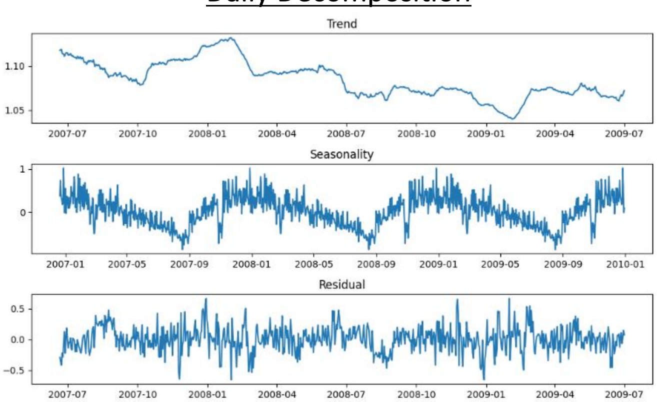
We calculated ACVF and ACF for each of the time series (pre-decomposition). ACVF & ACF are very similar and show a clear periodical/seasonal pattern of around 1 year, therefore seasonal decomposition was chosen with the respective frequencies of time-scope/year - 365, 53, 12, 4 for daily, weekly, monthly, and quarterly time scopes accordingly.



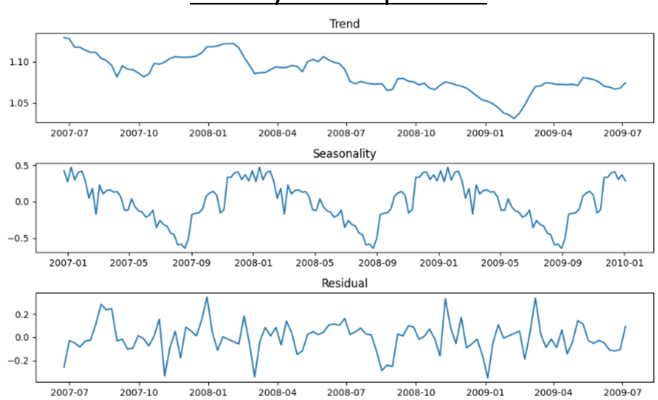
Timeseries Decomposition

We decomposed each of the time series using python's statsmodels seasonal decompose function. The chosen model as additive, not only because this is what the authors used, but also because after checking the multiplicative model we saw that it is much more accurate. The periods for the time scopes were the frequency of the time scope per year, assuming, as the authors, that there is a yearly cycle.

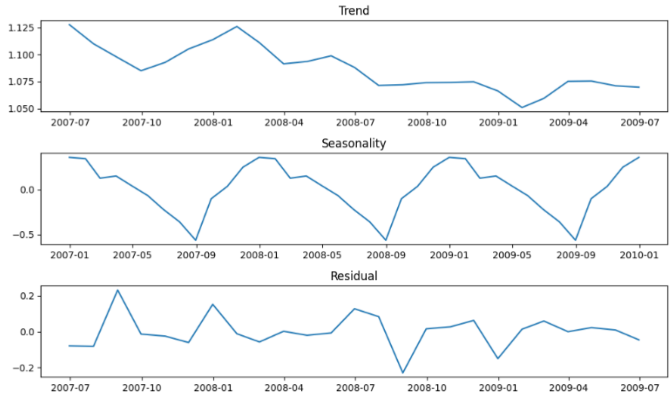
Daily Decomposition



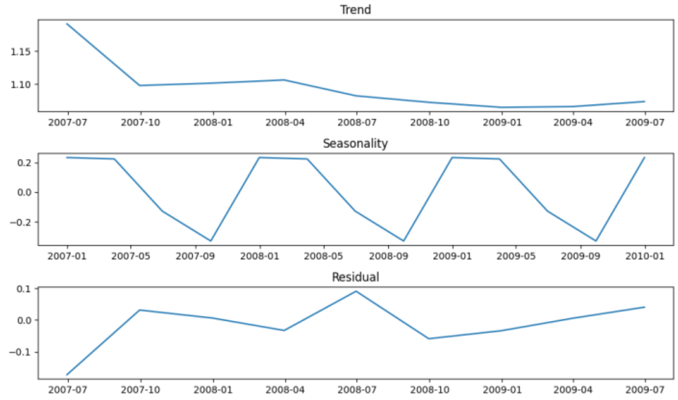
Weekly Decomposition



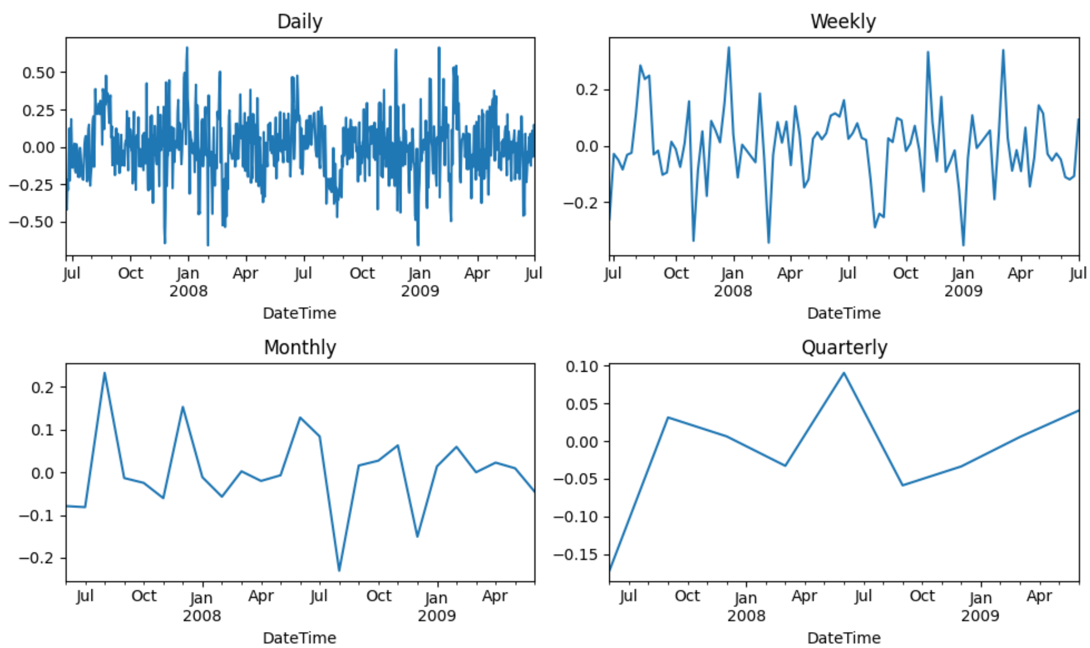
Monthly Decomposition



Quarterly Decomposition



After removing the trend and seasonal components, we are left with the residuals:



- The next step was to define if each series was indeed stationarity or not.
- We use the ADF test to test this question.

```
   Augmented Dickey-Fuller Results
=====================================
Test Statistic                 -5.177
P-value                         0.000
Lags                               18
-------------------------------------
```

```
   Augmented Dickey-Fuller Results
=====================================
Test Statistic                 -5.095
P-value                         0.000
Lags                               11
-------------------------------------
```

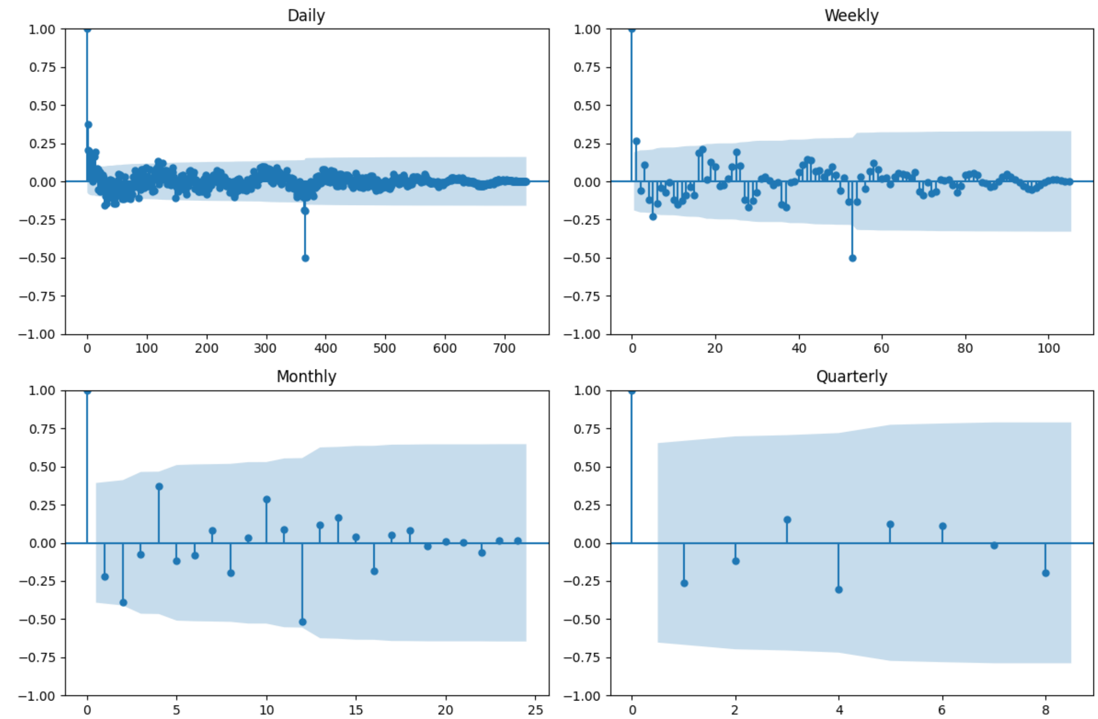
```
   Augmented Dickey-Fuller Results
=====================================
Test Statistic                 -5.572
P-value                         0.000
Lags                                2
-------------------------------------
```

```
   Augmented Dickey-Fuller Results
=====================================
Test Statistic                 -5.572
P-value                         0.000
Lags                                2
-------------------------------------
```

The results of the test show that each series is stationarity meaning we can continue in the search of the right ARMA parameters

For the residual timeseries, we could not find any clear models that fit the data from the ACF function by eye – it seems that the timeseries are too long and complex to analyse p, d, q values from these plots alone. Therefore, we continued to search those parameters according to the AIC criterion.

<u>ACF of the residuals data</u>



ARIMA Model Fitting

For each of the timeseries, we used statsmodel's ARIMA model to fit an ARIMA model for all combinations of p, d, q values, up to 3 and calculate their AIC score. The chosen models are those with the lower AIC score in each time scope.

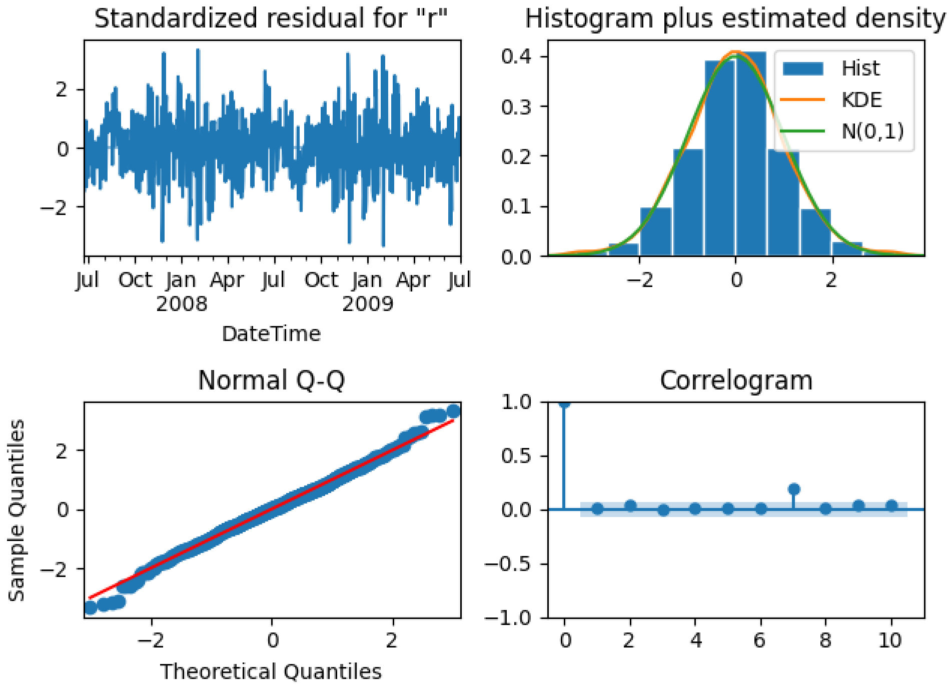
<u>AIC table summarization:</u>

Time Scope	p	d	q	AIC
Daily	2	0	3	-386.13
Weekly	3	0	2	-143.49
Monthly	3	0	0	-51.48
Quarterly	0	0	2	-20.72

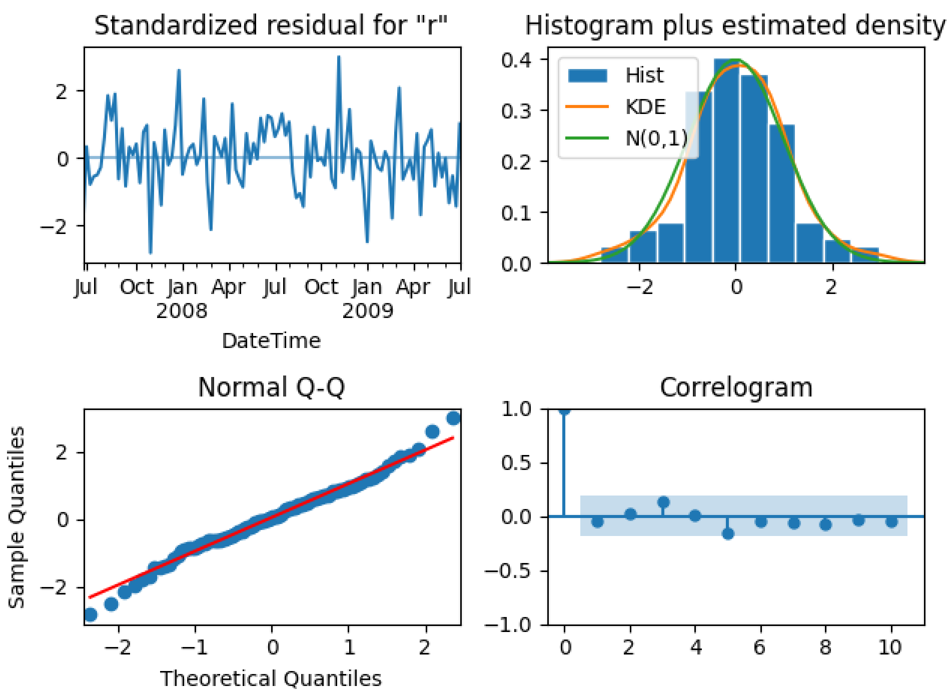
Model Diagnostics

Once we received the values for (p, d, q) from the parameters search we ran a diagnostic for each of the models to identify if each one of them is a result of a normal distribution. For that, we ran the diagnostics both visually and statistically (Shapiro test). We've also checked that all the data's models are also IID by conducting Ljung-Box test.

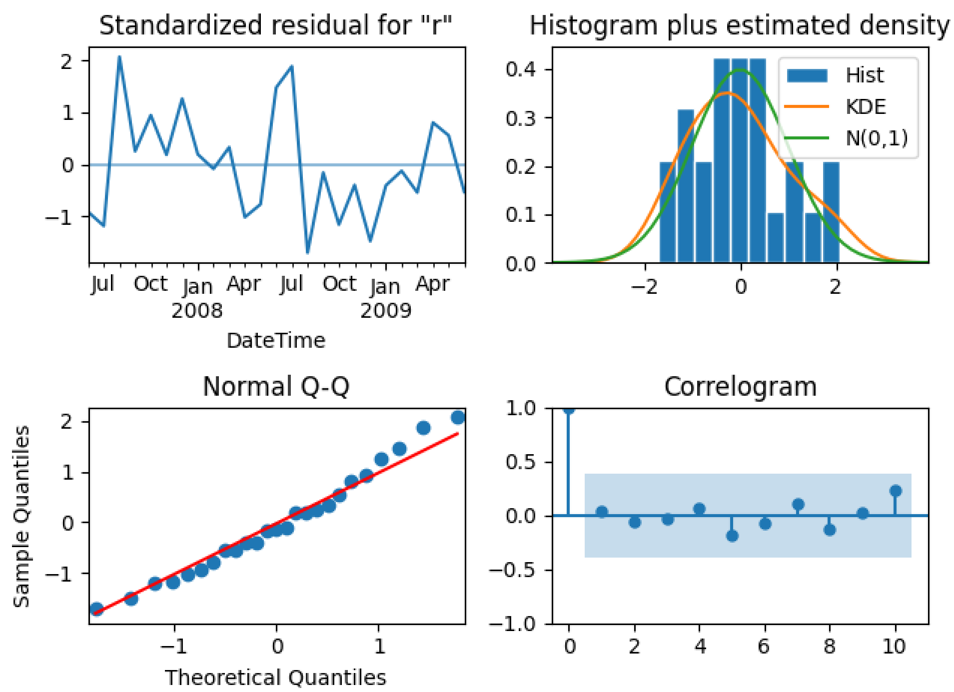
Daily



Weekly



Monthly



<u>Time series statistical tests summarization</u>

Time Scope	Ljung-Box values*	Shapiro-Wilk values*
Daily	0.67	0.12
Weekly	0.66	0.71
Monthly	0.86	0.67
Quarterly	0.2	0.78

*p-value=0.05

In all the cases, the null hypothesis is accepted, meaning, these residuals are normally distributed and are result of an IID. Both visually and statistically.

Model Predictions & Evaluations

We evaluated our derived models' predictive ability by forecasting the rest of the timeseries, 01/01/2010 and 26/11/2010. We then we compared our results of RMSE across smaller time periods to the author's RMSE results. The presented forecasting periods were chosen so that they could compare for the author's best forecasting time for each of the time periods.

<u>RMSE results comparison table</u>

Time Scope	Forecasting Period	Authors Models RMSE	Our Models RMSE
Daily	28	0.29	0.21
Weekly	20	0.18	0.17
Monthly	10	0.09	0.10
Quarterly	3	0.38	0.07

The table shows a good RMSE values for our models even comparing to the authors RMSE values.

Discussion & Conclusions

Using the same methods, albeit a different programming language and 10 years later in which the computational backend has probably improved, we got different results from the authors. Our results show that for all time scopes, an ARIMA model is not the best fit, but an ARMA (2, 0, 3) model for daily, ARMA (3, 0, 2) for weekly, AR (3) model for monthly and MA (2) model for quarterly.

Furthermore, we achieved better (lower) RMSE results for our models than the authors – but very close to those.

Since the daily model has a very similar predictive power, but our model is simpler, we believe that model is generally better performing than the authors' model. For the other time scopes, we reached these scores using slightly more complex models.

In general, the monthly and quarterly timeseries are different because they have much less data points than daily and weekly. Daily, weekly, monthly and quarterly time periods have 1106, 159, 37 & 13 datapoints respectively. Having so few datapoints, we think that the monthly and quarterly timeseries do not have enough datapoints for a meaningful analysis. We did get the same model trend as the authors - in which daily & weekly fit on different models than monthly & quarterly, which are fitted to simpler models.

We therefore conclude that the daily and weekly time scopes are better for prediction; The RMSE values might be higher in general, but are based on a much larger dataset, and the lower RMSE scores are retained in longer forecasting periods. For monthly and quarterly predictions, it might be that the trend and seasonal components are more meaningful for prediction. These components are clearly presented in the time series, as we can see in the series plot and the series ACF plots.