IE-360 Project

Group 20:
Murat Öztürk & Yasin Seçal

Instructor:
M. Gökçe Baydoğan

June 2022

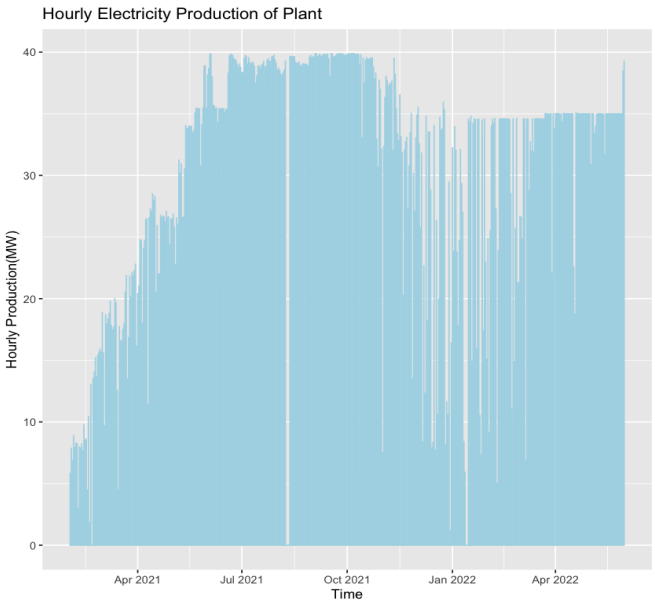
Table of Content

1. Introduction:

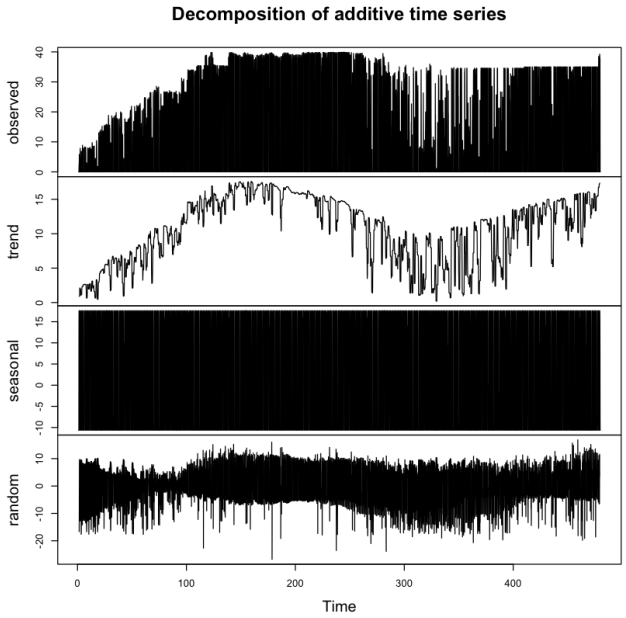
In this project, the main goal is forecasting hourly electricity production of KIVANC 2 Solar Power Plant on dates between 25 May and 03 June. Every day, prediction of the next 24 hours production is made by using production data from 2 days ago.

Beside past production data, weather predictions of 9 different coordinates nearby the power plant are available. Basically, the weather measurements are hourly temperature, relative humidity, downward shortwave radiation flux and total cloud cover for low-level type of cloud. However, these 9 different coordinates do not reflect the nearby coordinates of the power plant and may significantly change the prediction of the models. Therefore, based on map coordinates of the power plant, the closest 4 coordinates are selected among the 9 ones.

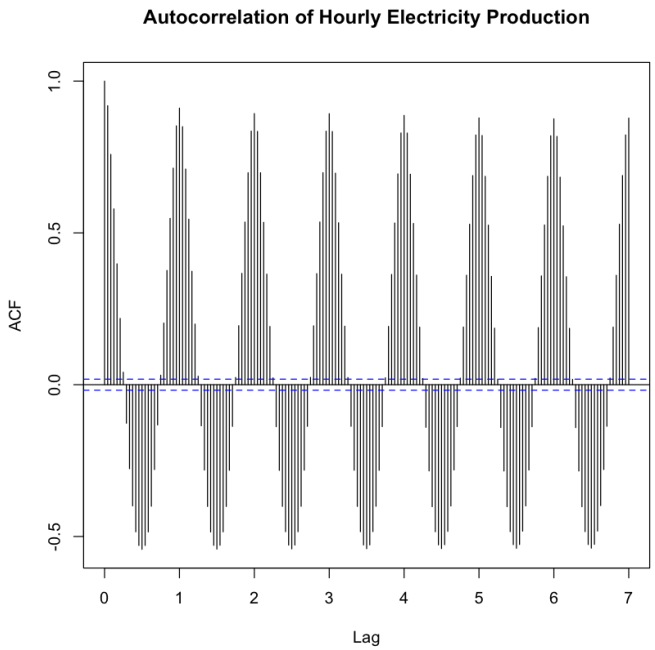
When hourly production data is plotted respect to time,



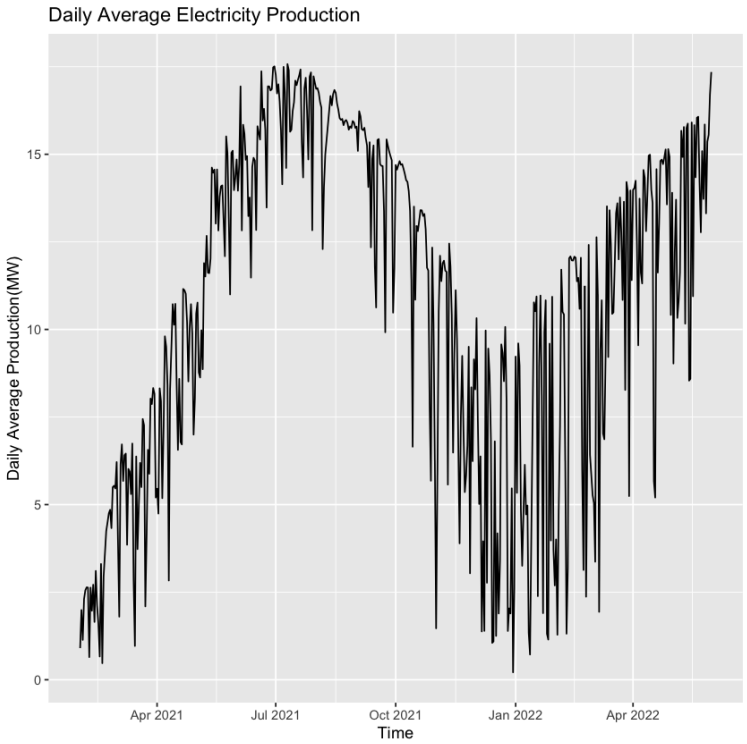
From this plot production data is not smooth and it varies a lot hour to hour and day to day. It is obvious that there is an increasing trend in Spring,2021. The reason here is that the capacity of the power plant increased in the spring period. Also, variability of daily production significantly decreases in summer months because weather conditions become more stable in summers. However, this plot is not obvious enough to understand seasonality and trend,so it is needed to have a look at decomposition of production data.



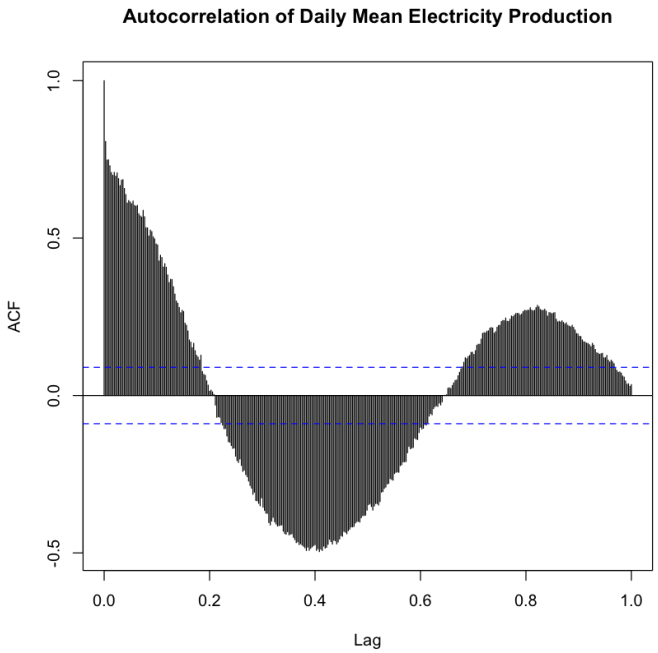
Now, trend is more understandable, yet it is not the same for seasonality, because variability in the data restricts to see seasonal patterns obviously. If having a look at the trend again, it is more clear that variability in weather in winters makes electricity production more unstable. Therefore, predicting any day from winter will be more difficult.
Before moving on, it is also important to analyze the autocorrelation and partial autocorrelation graphs.



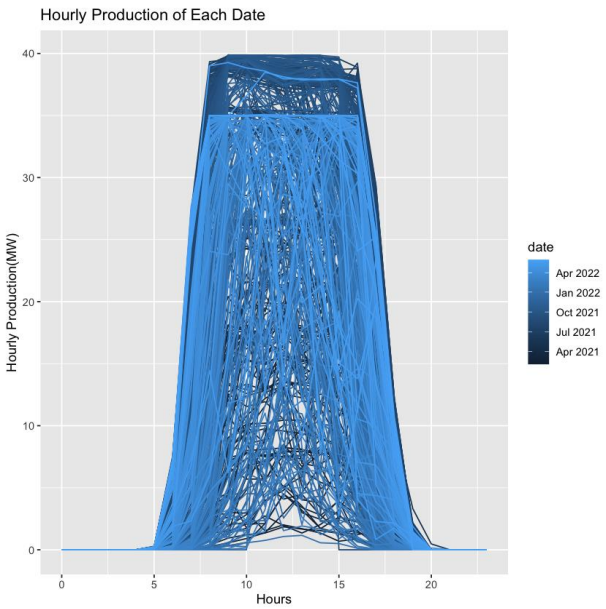
As seen above, the autocorrelation graph shows that there is a high correlation between hours and days. As expected, production is correlated with the same hours of the day. These autocorrelations can be visible at lag 24.



The plot above is a better representation of electricity production. The trend and variability are more obvious there.



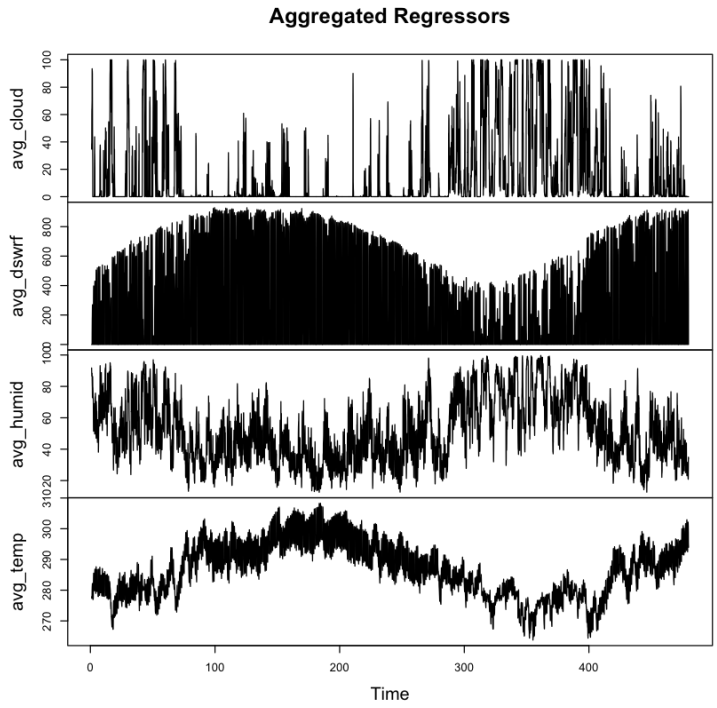
After removing effects of hours, the autocorrelation function shows there is a seasonal correlation between daily mean production. The lag here is 365 days. That means it is needed to use new input like month or season in the models.



When having a look at production of each day they have similar patterns. From this perspective it is possible to say that daily electricity production has a similar pattern and it means that each consecutive day is correlated. In overall production reaches its peak level at noon, then it tends to decrease towards evening.

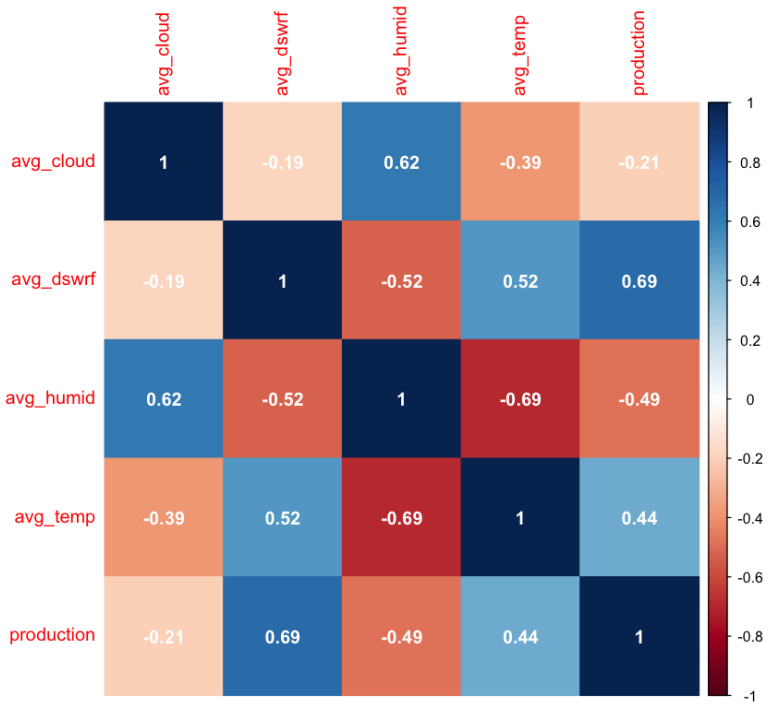
Up to now, electricity production is discussed in terms of autocorrelation, trend and seasonality. Before moving on the model and approaches part it is needed to have a look and analyze regressors.As mentioned in previous paragraphs some coordinates are irrelevant with the power plant's coordinates and these irrelevant points may cause mislead in the models . Therefore, 5 of them are removed.

Firstly, to analyze the regressors easily the average is taken for each of them. Now it may be better to plot each of the averages.



As seen above, average temperature and dswrf have cyclic behaviors. On the other hand, average dswrf and cloud have high variability. It is believed that these weather measurements may help explain the unexplained and unusual parts in the residuals in the models. Therefore, these cyclic and unusual patterns are not bad for now.

It may be good to see the relationship of regressor and output variables.



As seen above regressors have significant correlation with the production. Dswrf and temperature are regressors that affect the output positively. On the other hand, humidity and clouds are regressors that affect the output negatively. Also, they have correlation with each other, yet it is not important to discuss for modeling now.

Lastly, to forecast the electricity production there are various domains. In the next parts time series analysis and statistical modeling are used. As well as methods from the time series perspective there are other methods in data mining and machine learning domain such as decision trees, boosting. However, in this project only Autoregressive models and multiple linear regression are used because as discussed previously the dataset contains highly correlated inputs and outputs, then these models give better forecasts with these types of data.

2. Approaches

While comparing models we will split the data into train data and test data, from the end by 15 days. Then we will train the model with train data. After obtaining a model WMAPE (weighted mean absolute percentage error) value is calculated for the test data. The model which has the lowest value for the test data will be treated as the better model.

2.1. Linear Regression Approach

2.1.1. Weather Regressors

First approach used to create a model is linear regression with just weather forecast data. This model does not have any predictors for trend or seasonality.

```
Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                   -9.094e+01  5.064e+00 -17.957  < 2e-16 ***
`36.25_33_CLOUD_LOW_LAYER`     6.059e-03  8.061e-03   0.752  0.45227
`36.25_33_DSWRF`              -4.004e-05  4.029e-03  -0.010  0.99207
`36.25_33_REL_HUMIDITY`       -1.778e-01  2.345e-02  -7.580 3.72e-14 ***
`36.25_33_TEMP`                3.466e-01  1.756e-01   1.974  0.04844 *
`36.25_33.25_CLOUD_LOW_LAYER`  1.563e-02  8.414e-03   1.857  0.06327 .
`36.25_33.25_DSWRF`            4.348e-03  4.090e-03   1.063  0.28781
`36.25_33.25_REL_HUMIDITY`     1.389e-01  2.465e-02   5.634 1.81e-08 ***
`36.25_33.25_TEMP`             2.008e+00  1.745e-01  11.508  < 2e-16 ***
`36.5_33_CLOUD_LOW_LAYER`     -2.974e-02  7.033e-03  -4.228 2.37e-05 ***
`36.5_33_DSWRF`                2.392e-03  3.585e-03   0.667  0.50465
`36.5_33_REL_HUMIDITY`         5.077e-02  1.968e-02   2.580  0.00989 **
`36.5_33_TEMP`                 1.118e+00  1.293e-01   8.642  < 2e-16 ***
`36.5_33.25_CLOUD_LOW_LAYER`  -3.371e-03  7.968e-03  -0.423  0.67229
`36.5_33.25_DSWRF`             2.739e-02  3.671e-03   7.463 9.07e-14 ***
`36.5_33.25_REL_HUMIDITY`     -3.090e-02  2.456e-02  -1.258  0.20839
`36.5_33.25_TEMP`             -3.140e+00  1.482e-01 -21.191  < 2e-16 ***
```

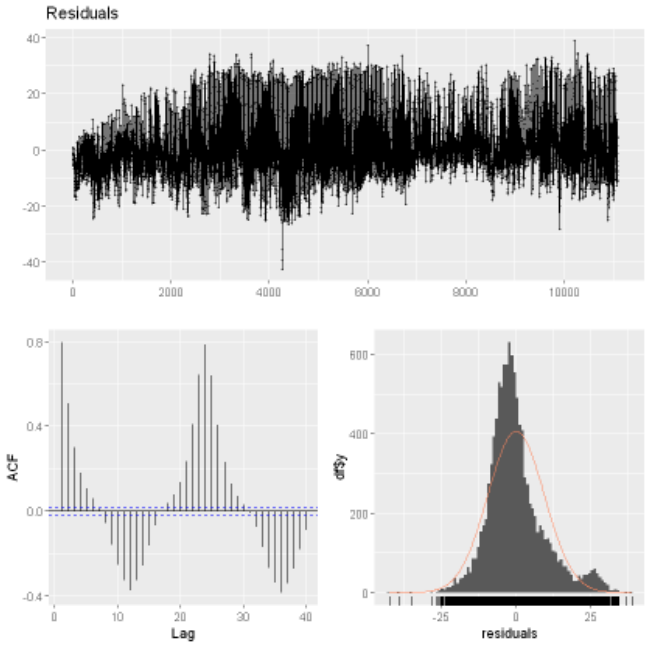
From the summary it is obvious that there are many insignificant predictors, this is due to the high correlation with some of the predictors.

```
Residual standard error: 9.439 on 11071 degrees of freedom
Multiple R-squared:  0.5635,    Adjusted R-squared:  0.5629
F-statistic: 893.4 on 16 and 11071 DF,  p-value: < 2.2e-16
```

```
"train WMAPE: 0.66306321488786"
"test WMAPE: 0.578318559011905"
```

The adjusted R-squared is 0.5629. The WMAPE value is 0.5783.

```
        Breusch-Godfrey test for serial correlation of order up to 20

data:  Residuals
LM test = 8151.2, df = 20, p-value < 2.2e-16
```



Residuals do not look like white noise. There is a very strong autocorrelation. Therefore it doesn't pass the breusch-godfrey test.

2.1.2. Weather Regressors and Seasonality (hour + month)

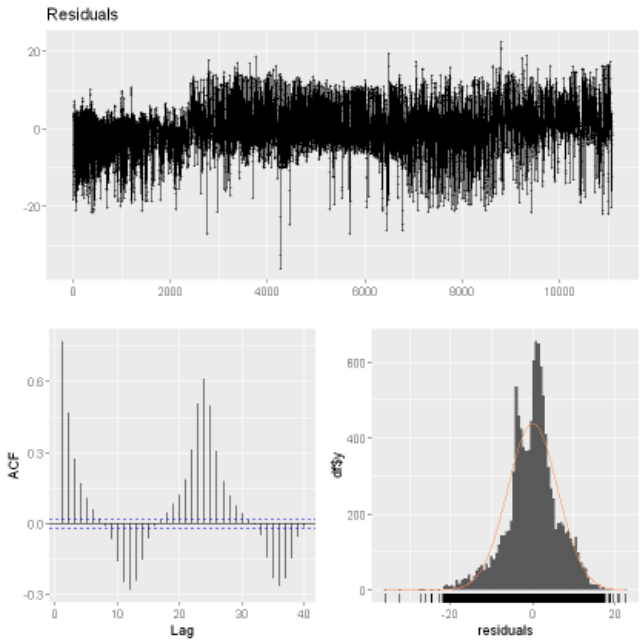
After adding seasonality the model seems to improve. 16 of the hours coefficients (23 at total) and 8 of the month coefficients (11 at total) are significant.

```
Residual standard error: 6.106 on 11037 degrees of freedom
Multiple R-squared:  0.8179,    Adjusted R-squared:  0.8171
F-statistic: 991.5 on 50 and 11037 DF,  p-value: < 2.2e-16
```

```
"train WMAPE: 0.436510709266955"
"test WMAPE: 0.26413425005615"
```

Adjusted square and WMAPE values have improved significantly.

```
        Breusch-Godfrey test for serial correlation of order up to 54

data:  Residuals
LM test = 8279.1, df = 54, p-value < 2.2e-16
```



The autocorrelation in the residuals still exists.

2.1.3. Weather Regressors and Seasonality (hour + month) + New Features

For the final model with linear regression, we tried adding more features.
First feature added is cloud_sqr, this feature is equal to the sum of squares of the cloud values of different locations (4 locations).
Second feature added is clear_sky, this feature is equal to 1 when cloud_sqr<2800 & hour<19 & hour>8. The intuition behind this was to determine when a clear sky with none to little clouds is obtained. The decision of the particular numbers, 2800, 19, 8 were based on intuition and observations made with the data.

Third and fourth features were added after seeing cloud_sqr has performed well.

```
cloud_sqr                4.125e-04  1.086e-04   3.799 0.000146 ***
clear_sky                5.607e+00  3.009e-01  18.630  < 2e-16 ***
dswrf_sqr               -1.118e-06  1.326e-06  -0.843 0.399263
temp_sqr                 2.911e-03  7.961e-04   3.657 0.000256 ***
```

After adding these, to solve the autocorrelation problem in the residuals, it is needed to add a lag variable. Since the dataset has only the data from 2 days ago to predict a day ahead, a lag variable with 3 days lag is added, equivalent to 72 hours.

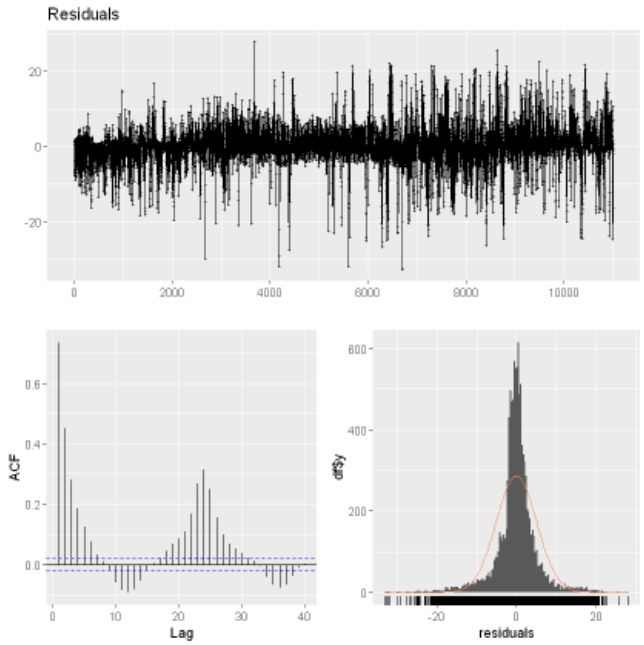
```
lag72                    4.832e-01  7.454e-03  64.817  < 2e-16 ***
```

It has a t value of 64.817 which is the highest in this model. It is significant.

```
Residual standard error: 5.078 on 10960 degrees of freedom
Multiple R-squared:  0.8746,    Adjusted R-squared:  0.874
F-statistic:  1390 on 55 and 10960 DF,  p-value: < 2.2e-16
                "train WMAPE: 0.306287603640692"
                "test WMAPE: 0.224963559576275"
```

Adjusted R-squared and WMAPE have improved.

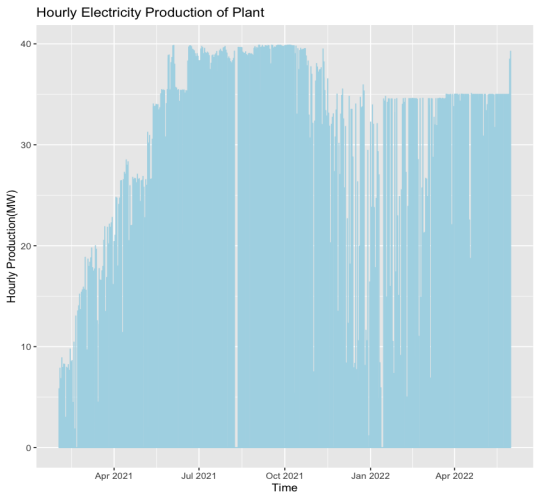
```
        Breusch-Godfrey test for serial correlation of order up to 59

data:  Residuals
LM test = 6648.7, df = 59, p-value < 2.2e-16
```

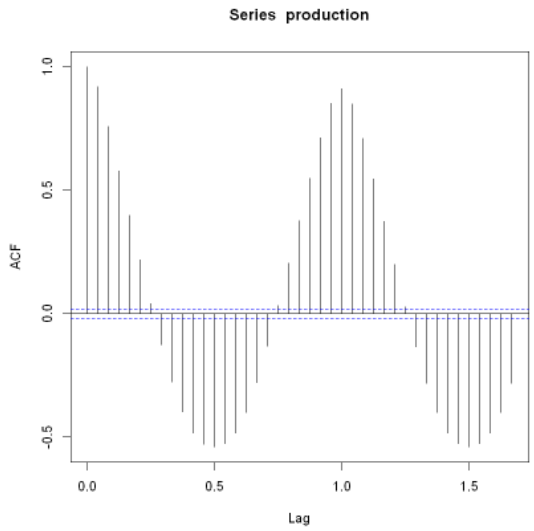


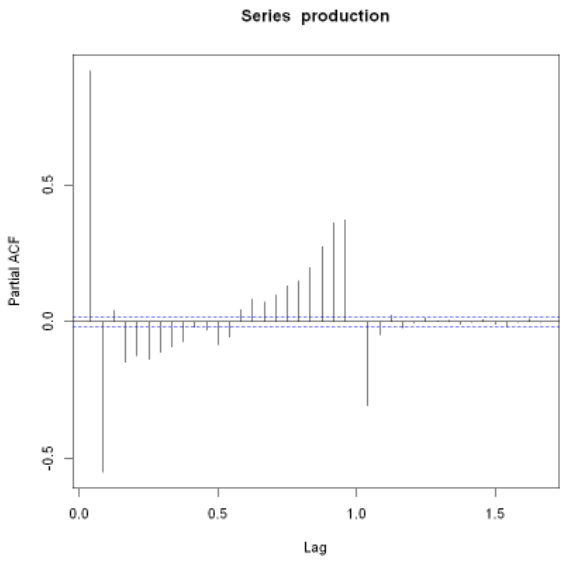
The residuals histogram looks better in terms of normality but the acf plot still looks bad. However, since adding lag as 1 hour is a problem the case cannot be solved.

2.2. Arima Approach

To construct an arima model the dataset is analyzed from the time series domain.







As it can be seen from the residuals of the linear models, there is a significant autocorrelation on the data.

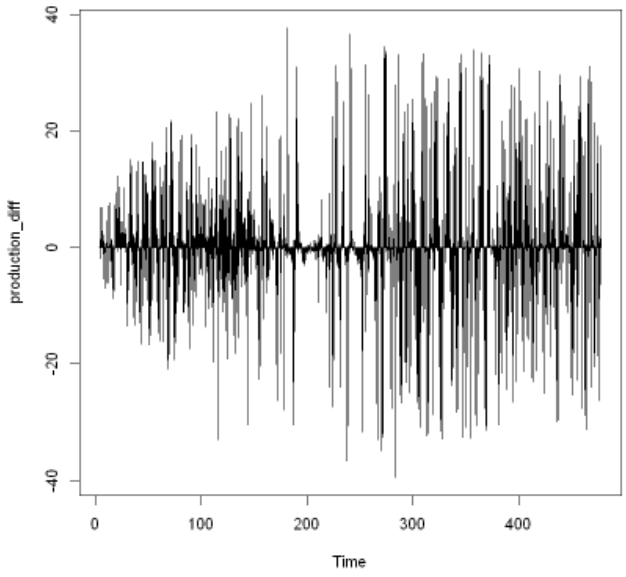
```
"p-value smaller than printed p-value"
```

```
        KPSS Test for Level Stationarity

data:  production
KPSS Level = 2.8187, Truncation lag parameter = 13, p-value = 0.01
```

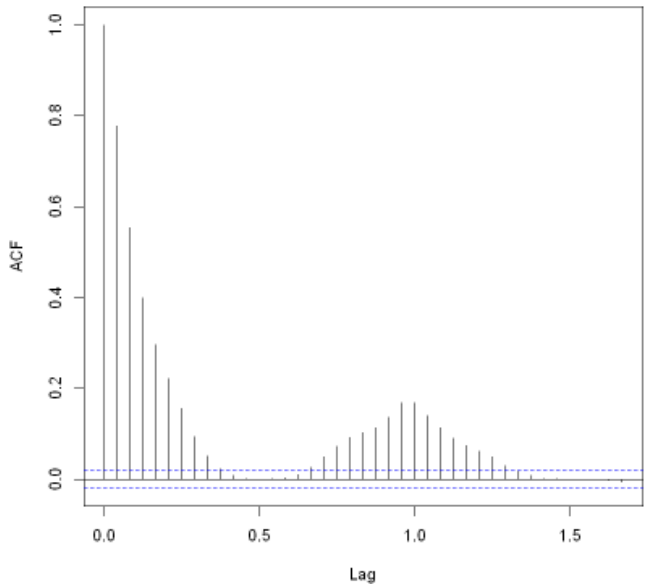
Non-Stationary

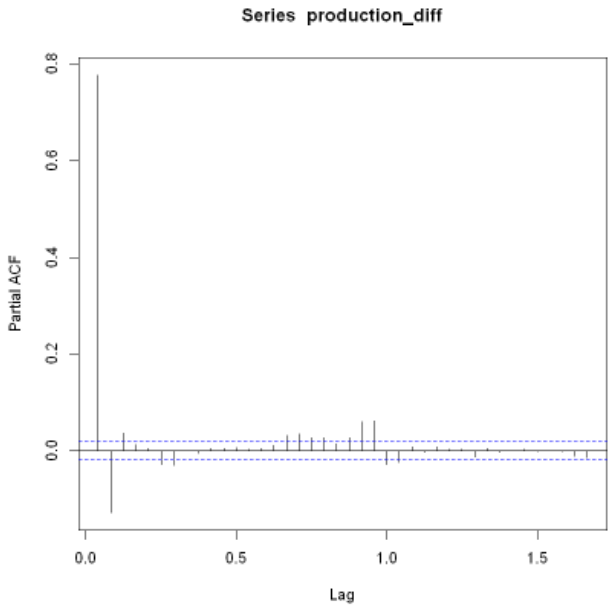
The kpss test rejects the null hypothesis, therefore the difference of the time series should be taken.

Seasonal difference is taken, lag is taken as 72 rather than 24 since the model is required to forecast points 72 hours ahead in time.



Series production_diff





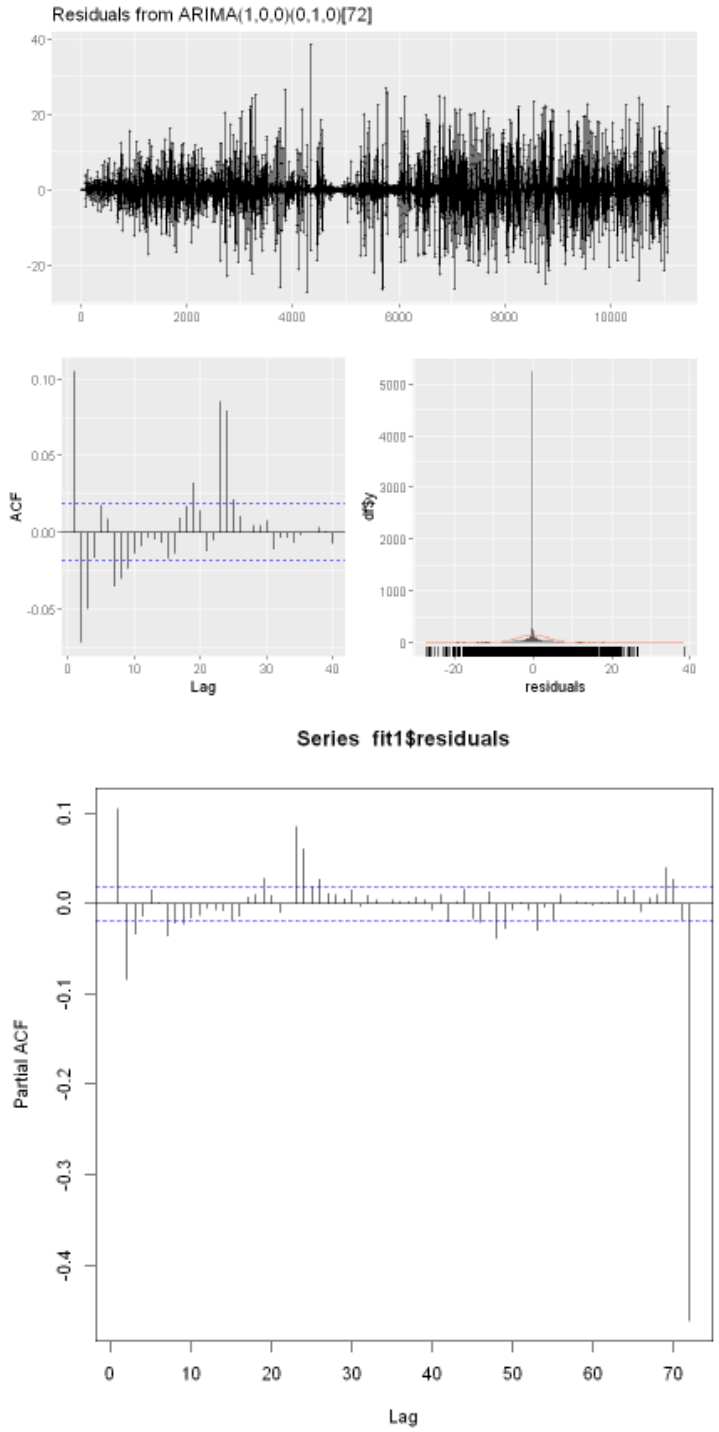

```
"p-value greater than printed p-value"

        KPSS Test for Level Stationarity

data:  production_diff
KPSS Level = 0.079093, Truncation lag parameter = 13, p-value = 0.1

Stationary
```

Null hypothesis can not be rejected, no need to take more differences.

2.2.1. Order (1,0,0) | SOrder (0,1,0)_72

By looking at the pacf at the differenced production, add 1 AR to the arima model. The model will have order = (1,0,0), seasonal order = (0,1,0)_72

```
Call:
arima(x = train, order = c(1, 0, 0), seasonal = list(order = c(0, 1, 0), period = 72))

Coefficients:
         ar1
      0.7828
s.e.  0.0059

sigma^2 estimated as 16.08:  log likelihood = -30930.12,  aic = 61864.24

Training set error measures:
                    ME     RMSE      MAE MPE MAPE      MASE      ACF1
Training set 0.01699801 3.996849 1.783461 NaN  Inf 0.6438037 0.1048959
```



A spike at lag 72 can be seen. That would suggest adding seasonal AR 1.

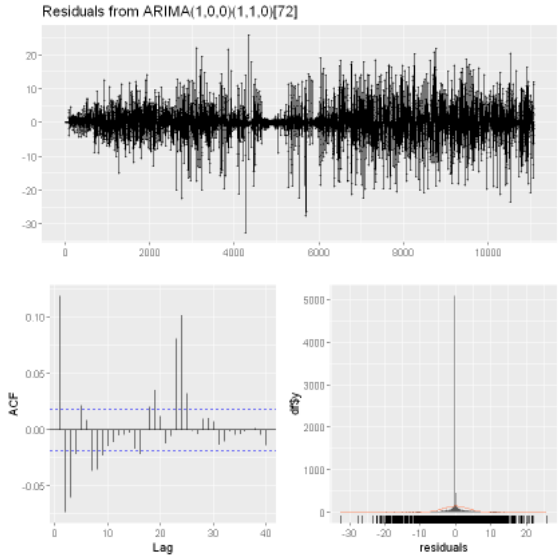
```
"train WMAPE: 0.16990133575405"
"test WMAPE: 0.236141508955498"
```

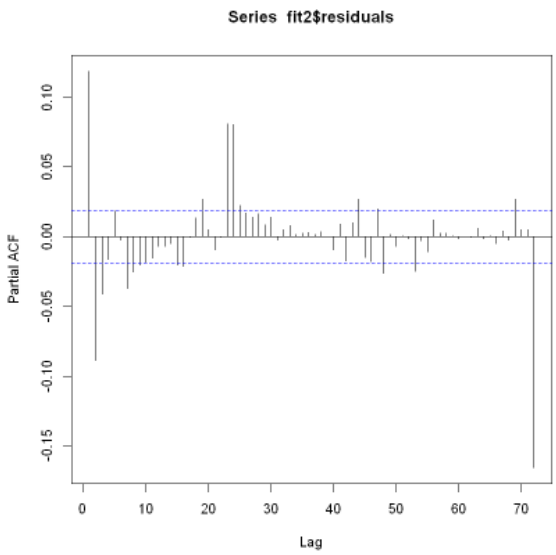
The WMAPE value is almost as good as the best linear regression model that has been done in 2.1.3.

2.2.2. Order (1,0,0) | SOrder (1,1,0)_72

By the observations of the residuals of the previous model add Seasonal AR 1.

```
arima(x = train, order = c(1, 0, 0), seasonal = list(order = c(1, 1, 0), period = 72))

Coefficients:
         ar1     sar1
      0.7921  -0.4688
s.e.  0.0058   0.0084

sigma^2 estimated as 12.54:  log likelihood = -29567.6,  aic = 59141.2

Training set error measures:
                     ME     RMSE      MAE MPE MAPE      MASE      ACF1
Training set 0.02386221 3.528976 1.653441 NaN  Inf 0.5968682 0.1187119
```





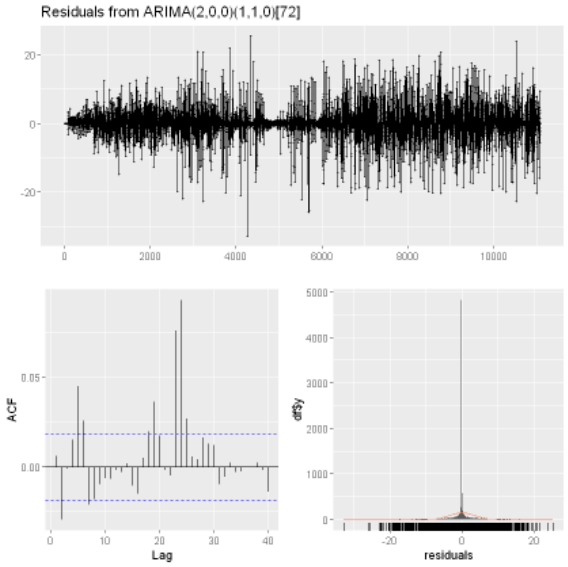

```
"train WMAPE: 0.157514956959684"
"test WMAPE: 0.216382850722771"
```

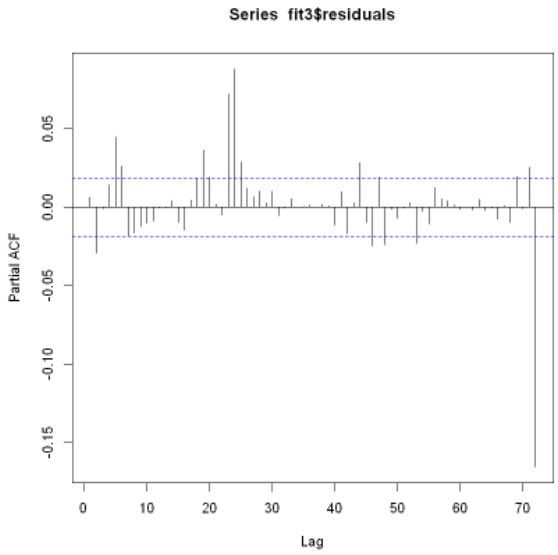
Pacf plot is still not fixed but the spike has decreased in value and test WMAPE has improved so continue with seasonal AR1.

2.2.3. Order (2,0,0) | SOrder (1,1,0)_72

There is also a spike at pacf lag 2. Try adding AR 2 to get rid of that spike.

```
arima(x = train, order = c(2, 0, 0), seasonal = list(order = c(1, 1, 0), period = 72))

Coefficients:
         ar1      ar2     sar1
      0.9110  -0.1499  -0.4727
s.e.  0.0094   0.0094   0.0084

sigma^2 estimated as 12.25:  log likelihood = -29442.43,  aic = 58892.86

Training set error measures:
                     ME     RMSE      MAE MPE MAPE      MASE        ACF1
Training set 0.02749758 3.489043 1.635536 NaN  Inf 0.5904048 0.006160518
```






```
"train WMAPE: 0.155809246739833"
"test WMAPE: 0.216272792969796"
```

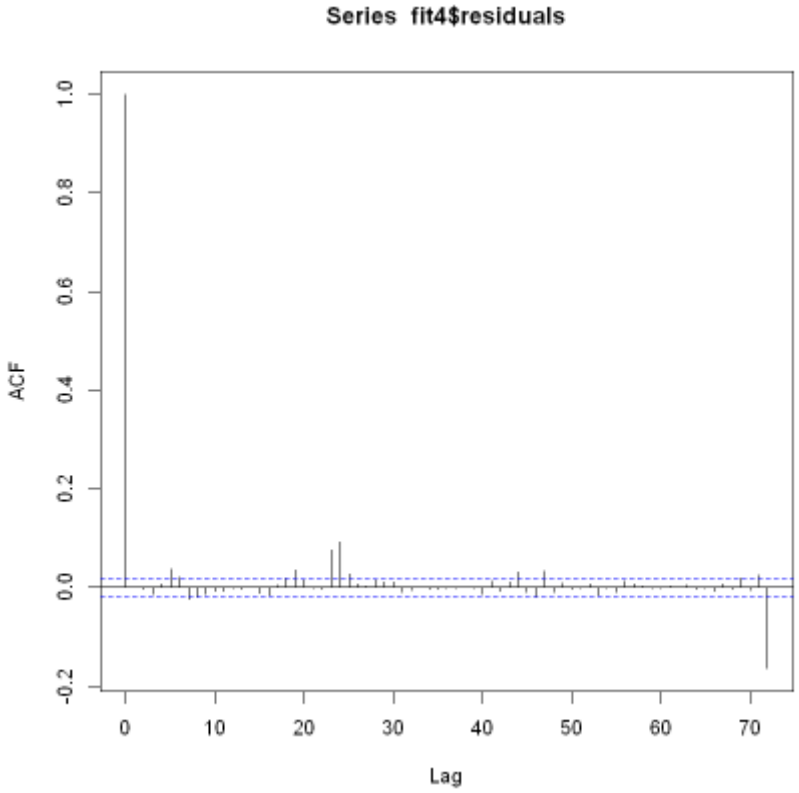
Very small improvement, but the pacf is better for the first 2 lags.

2.2.4. Order (2,0,1) | SOrder (1,1,0)_72

Add MA 1 to see if it makes the model better.

```
"train WMAPE: 0.155711625800991"
"test WMAPE: 0.216277624132058"
```

It does not improve WMAPE.

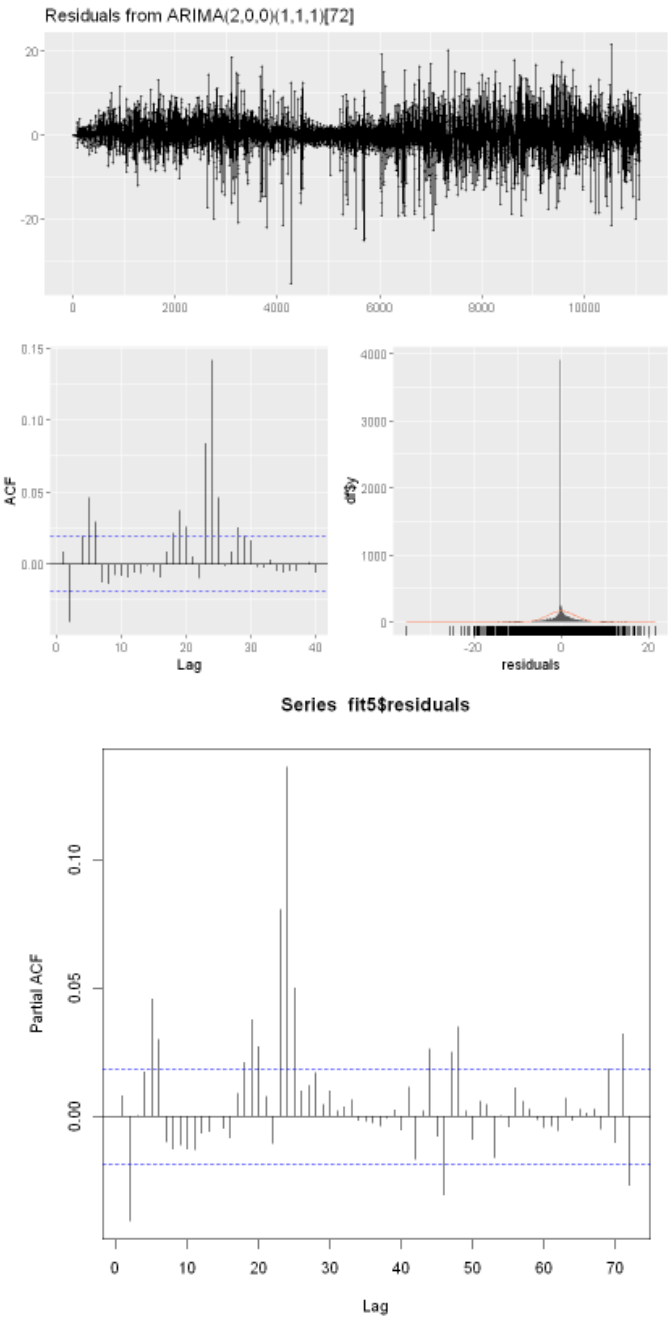


ACF is high at lag 72.

2.2.5. Order (2,0,0) | SOrder (1,1,1)_72

Removing MA 1. Looking at the acf plot, add seasonal MA 1 to see if it makes the model better.

```
Coefficients:
         ar1      ar2     sar1     sma1
      0.9264  -0.1582  -0.0231  -0.7771
s.e.  0.0095   0.0094   0.0115   0.0067

sigma^2 estimated as 9.621:  log likelihood = -28135.93,  aic = 56281.87

Training set error measures:
                     ME     RMSE      MAE MPE MAPE     MASE        ACF1
Training set 0.07707041 3.091661 1.511119 NaN  Inf 0.545492 0.008133973
```




```
"train WMAPE: 0.143956657314241"
"test WMAPE: 0.213353249893948"
```

The aic value and WMAPE value have decreased which makes this model better. However, the spikes were not eliminated entirely. It may not be possible to get rid of all the spikes. For now the model with arima is better than the linear regression models that have been tried

2.2.6. ARIMAX Order (2,0,0) | SOrder (1,1,1)_72 + cloud_sqr

Adding a regression matrix to the arima model can improve the model, however, adding even 1 variable to the regression matrix would cause it to not compute a solution under 30 minutes.

2.3. Modeling For Each Hour Separately

While analyzing the models and their predictions, it is clear that there were predictions below 0 production, also the predictions for some hours were really bad.

For example at hour 2, there is no recorded data that has production other than 0. However, the models that have to generalize for all hours, can not predict precisely 0 for that hour.

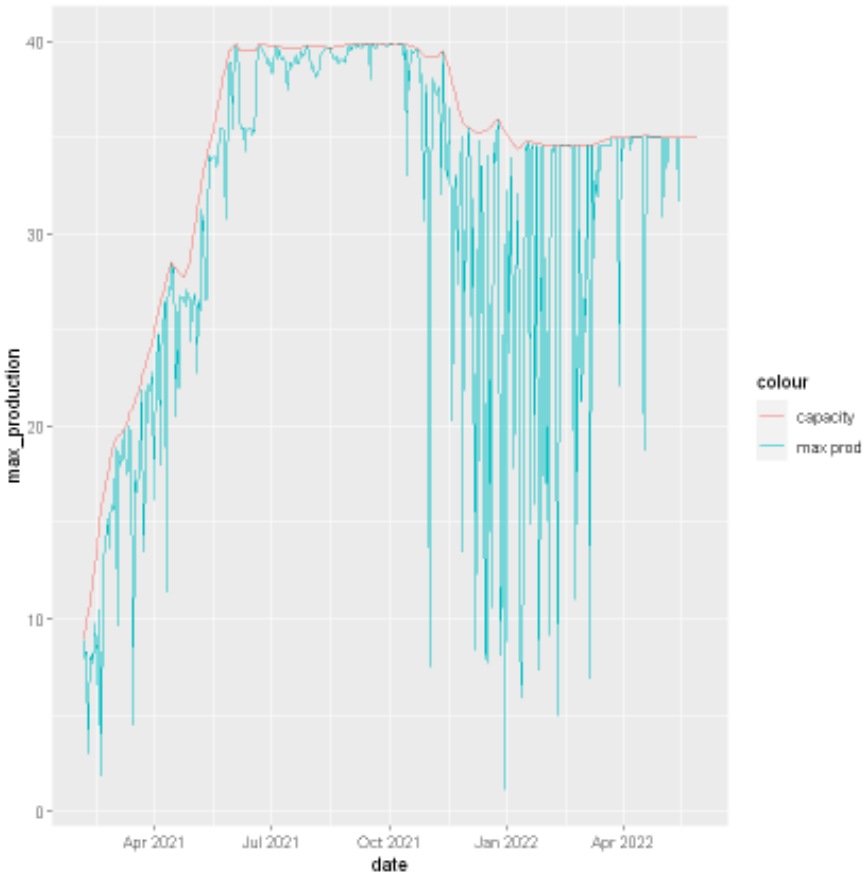
Therefore, it is decided to set those hours 0 by hand, but that wouldn't be enough as those hours could affect the predictions of the other hours in a bad way.

After some thinking, we wanted to try creating a model for each hour separately. This would reduce the data for each model by a lot, this is a big disadvantage. However, if that reduced data is still enough, these models may outperform the models that use all the hours.

We have realized from the beginning that the capacity was not constant, and capacity is affecting the production values. Thus, making use of that would improve the models.

To use that, get the maximum production value of the hours in days. A function called rolled max with width 15, then a function called roll mean were used to smooth the maximum values.

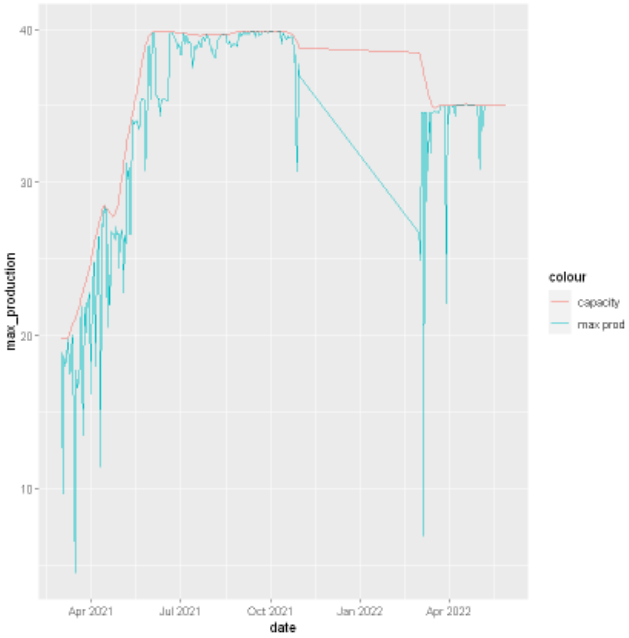
The process of determining the value of the width was mostly comparing the capacity and the value's of maximum production in the days and determining if the capacity line was fitting well.



After finding the capacities, utilization (production/capacity) values were calculated.
Then linear models were fit to predict utilization values, the values below 0 were assigned 0, values above 1 were assigned 1. After finding the utilization values, multiplying the values with capacity gave the predictions for the production.

```
"train WMAPE: 0.151151536203116"
"test WMAPE: 0.194208060010904"
```

Better test WMAPE value is obtained.
After getting this result predictions were analyzed, and there was some data that did not make sense. Some data had the same values for the predictors but the production values were different. It seemed that the values for the weather data were not 100% accurate. For that reason a decision is taken to get rid of some outliers.
The dates that had a prediction value with 5 standard deviations away from the real production value would be removed.
While doing that, a model is wanted to predict May and June, it is decided to get rid of the datas in winter which would affect the predictions for summer in a bad way.



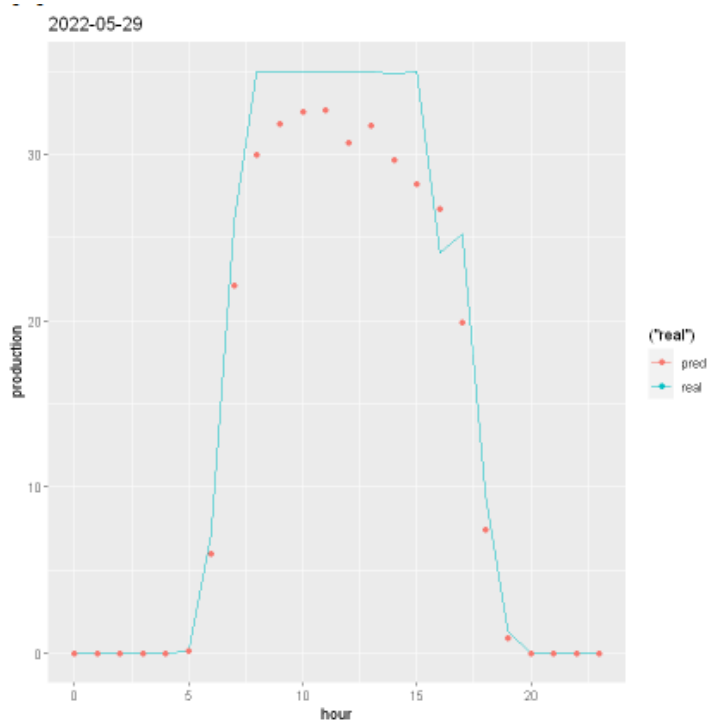

```
"train WMAPE: 0.0992710258572334"
"test WMAPE: 0.180071990417073"
```

This is the WMAPE values for test data with 15days.

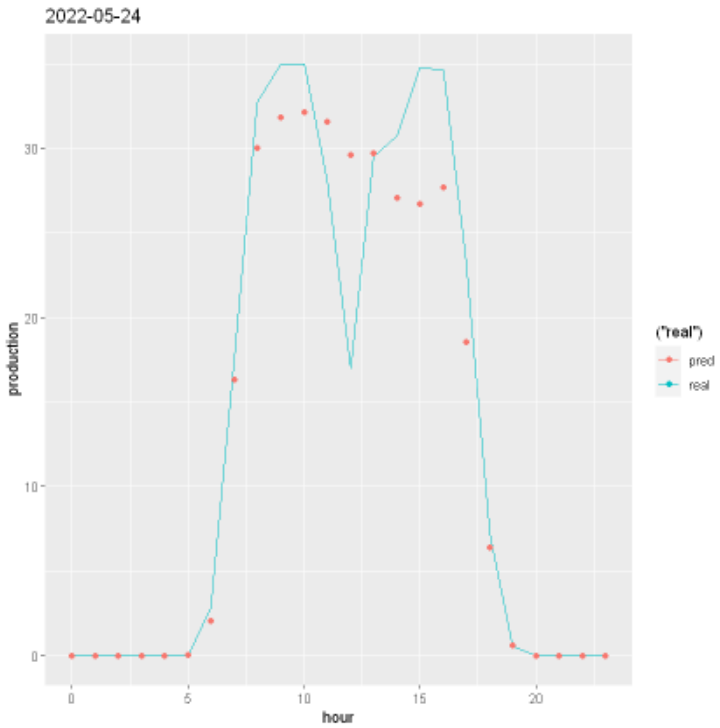
```
"train WMAPE: 0.100945195173881"
"test WMAPE: 0.153285904394444"
```

This is the WMAPE values for test data with 10 days.

Prediction and real data for date '2022-05-29'.



The problem with this model was that it still had flaws. It couldn’t predict 35, but rather stayed near 32,33. This was probably due to some of the days when production would fall drastically without a warning, thus it would try to stay in the middle in case that happened. e.g.



3. Results

To determine which model gives more reliable results, Weighted Mean Absolute Percentage Error is used. In the project, the dataset is separated into a train and test set. Last 15 rows are used as a test set.

WMAPE takes the weight of each error, so it is not affected by low and high actual values as much as MAPE.

The WMAPE metric is available for all candidate models above. When possible models are compared it is obvious that the models for each separate hour are the best.

Comparison:

2.1.3. Weather Regressors and Seasonality (hour + month) + New Features

```
"train WMAPE: 0.306287603640692"
"test WMAPE: 0.224963559576275"
```

2.2.5. Order (2,0,0) | SOrder (1,1,1)_72

```
"train WMAPE: 0.143956657314241"
"test WMAPE: 0.213353249893948"
```

2.3. Modeling For Each Hour Separately

```
"train WMAPE: 0.0992710258572334"
"test WMAPE: 0.180071990417073"
```

The best model's WMAPE value:

```
"train WMAPE: 0.0992710258572334"
"test WMAPE: 0.180071990417073"
```

4. Conclusion & Future Ideas

Even though the final model gives lower WMAPE error for the train set It still gives some unsatisfying results for the test set. That means there are still some points to improve. On the other hand, improving the model may not work because the electricity production data may highly deviate, so the model can give abnormal estimation. Moreover, to predict the future production another forecast(weather forecast) is used. For example, it is very normal to see intensive cloud layers in the sky although the weather forecast says a clear sky. These types of problems make it almost impossible to construct perfect models.

Even though it is impossible to have a perfect model there are some other ways to obtain better results with different approaches and domains. Also, some extensions such as defining new attributes is reasonable.

Firstly, the simplest way is to remove more data that are outliers from the training set. However, it should be discussed when training models because removing some data may cause overfit. Even if there is riskiness of overfit, removing this data works well in standard

weather conditions. It should not be forgotten that the model may collapse in unusual conditions.

Another simplest idea is identifying new attributes for models. It requires more exploratory analysis on the data and finding exogenous attributes by searching related literature for solar power plants. For example in the final model the “clear_sky” attribute is defined after realizing the significant effects of no cloudy hours. In addition to cloud information, there can be found such critical conditions, then a new combined attribute can be added to the model.

Last but not least, the model approach can be changed or improved. At that point, the most powerful model can be hybrid models. For instance, incorporating the regression and time series analysis approach hybrid models can be obtained. After training the model with multiple linear regression residuals can be modeled with an autoregressive approach. Then, the autoregression model can train error terms and overall predictions can be improved, too. Rather than training the error terms, it is possible to construct 24 different hourly models with an autoregressive approach. However, it is not guaranteed that ARIMA gives more satisfactory results. Therefore, these hourly models should be constructed with the ARIMAX approach, because it is obvious that ARIMA will work better with any weather forecast regressor.

Last words, all these considerations will take the models one more step away.

5. Code

The code available here link