

IE360 PROJECT

Buse Erduran, Baran Kırkgöz, Derya Özşeker

2022-06-07

There is a solar power plant named “Kıvanç 2 Güneş Enerji Santrali” in Mersin, Turkey which is located between 36-37° north latitude and 33-35° east longitude. The aim of this project is to analyze the behavior of that power plant’s hourly solar electric production by looking at the past data and choosing an approach to predict the future productions. With respect to the ‘persistence approach’, the forecasts will be with lag 48, which corresponds to 2 days.

There are some variables that might affect the production rates:

TEMP: This is the temperature variable for this location. There are two impacts of temperature. The first one is, temperature can represent the seasonality. In a season, hourly temperature values will be similar. The second one is, the efficiency of solar plants decreases with higher temperatures due to the fact that high temperatures affect the solar panels.

REL_HUMIDITY: This value stands for relative humidity at the provided location. One can reach the information about the rainy or cloudy times by looking at this value. Rainy or cloudy times, which means relative humidity potentially decrease the production.

DSWRF: This is the short version of downward shortwave radiation flux which is known to be highly important for the production level.

CLOUD_LOW_LAYER: This is total cloud cover data (in terms of percentage) for low-level type of clouds which is also expected to affect the production rate.

By looking at the paired correlations between these variables and the production data, one can see the possible relations between them and that will give an idea for choosing the regressors when founding prediction models.

Uploading Necessary Libraries:

```
library(xlsx)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##   date, intersect, setdiff, union

library(zoo)

##
## Attaching package: 'zoo'
```

```

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

library(ggplot2)
library(RcppRoll)
library(GGally)

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

library(skimr)
library(forecast)

## Registered S3 method overwritten by 'quantmod':
##   method           from
##   as.zoo.data.frame zoo

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(data.table)

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year

library(reshape)

##
## Attaching package: 'reshape'

```

```

## The following object is masked from 'package:data.table':
##
##     melt

## The following object is masked from 'package:dplyr':
##
##     rename

## The following object is masked from 'package:lubridate':
##
##     stamp

library(reshape2)

##
## Attaching package: 'reshape2'

## The following objects are masked from 'package:reshape':
##
##     colsplit, melt, recast

## The following objects are masked from 'package:data.table':
##
##     dcast, melt

library(readr)
library(caTools)

```

Data Manipulation:

```

long_weather <- data.table(read_csv("~/Desktop/project_data/long_weather.csv"))

## Rows: 403488 Columns: 6
## -- Column specification -----
## Delimiter: ","
## chr (1): variable
## dbl (4): hour, lat, lon, value
## date (1): date
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

production <- data.table(read_csv("~/Desktop/project_data/production.csv"))

## Rows: 10896 Columns: 3
## -- Column specification -----
## Delimiter: ","
## dbl (2): hour, production
## date (1): date
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

```

```

str(long_weather)

## Classes 'data.table' and 'data.frame':  403488 obs. of  6 variables:
## $ date      : Date, format: "2021-02-01" "2021-02-01" ...
## $ hour      : num  0 1 2 3 4 5 6 7 8 9 ...
## $ lat       : num  36.2 36.2 36.2 36.2 36.2 ...
## $ lon       : num  33 33 33 33 33 33 33 33 33 33 ...
## $ variable: chr  "DSWRF" "DSWRF" "DSWRF" "DSWRF" ...
## $ value     : num  0 0 0 0 0 0 0 0 0 3 ...
## - attr(*, ".internal.selfref")=<externalptr>

wide_weather= dcast(long_weather, date+hour~lat+lon+variable)
data <- data.table(merge(wide_weather,production))
data[,AverageTEMP:=rowMeans(data[,c("36.75_33.5_TEMP","36.75_33.25_TEMP","36.75_33_TEMP","36.5_33.5_TEMP")])
data[,AverageREL_HUMIDITY:=rowMeans(data[,c("36.75_33.5_REL_HUMIDITY","36.75_33.25_REL_HUMIDITY","36.75_33_REL_HUMIDITY")])
data[,AverageDSWRF:=rowMeans(data[,c("36.75_33.5_DSWRF","36.75_33.25_DSWRF","36.75_33_DSWRF","36.5_33.5_DSWRF")])
data[,AverageCLOUD_LOW_LAYER:=rowMeans(data[,c("36.75_33.5_CLOUD_LOW_LAYER","36.75_33.25_CLOUD_LOW_LAYER")])
data <- data[order(hour,decreasing = F)]
data <- data[order(date,decreasing = F)]
data[, Year:=as.factor(year(date))]
data[,Month := as.factor(month(date))]
data[,Hour_factor := as.factor(hour)]
data[,max_in_month:=runmax(x=production, k=720, align = "left")]
data[,max_in_week:=runmax(x=production, k=168, align ="left")]
data[hour<=5|hour>=21,night:=1]
data[hour<21&hour>5,night:=0]
data$night <- as.factor(data$night)
data[,Lag1:=c(NA, production[1:(.N-1)])]
data[,Lag_week:=c(rep(NA,168), production[1:(.N-24*7)])]
data[,Lag_day:=c(rep(NA,24), production[1:(.N-24)])]
data[, Trend:=(1:.N)]
colnames(data) <- c("date","hour","CLOUD_LOW_LAYER_36.25_33","DSWRF_36.25_33","REL_HUMIDITY_36.25_33","Production")
View(data)

```

Data Analysis:

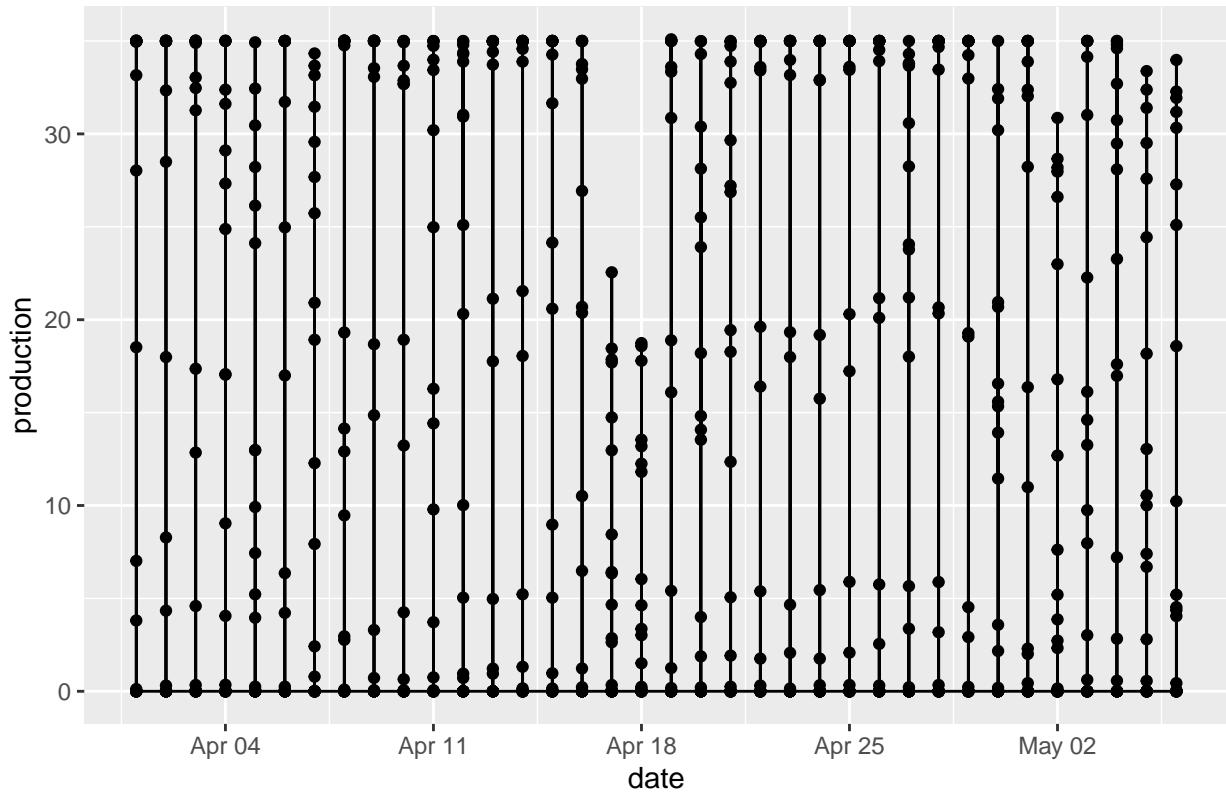
Before going into models, the data is analyzed firstly. Since the data includes a very long time, only April and May 2022 are examined. As there can be seen below, production is 0 at night times and it reaches its maximum during midday hours. It does makes sense because electricity production is affected directly by sunlight.

```

ggplot(subset(production,date >= "2022-04-01"),aes(x=date, y=production))+
  geom_line() + geom_point() + ggtitle("Electricity Production April-May 2022")

```

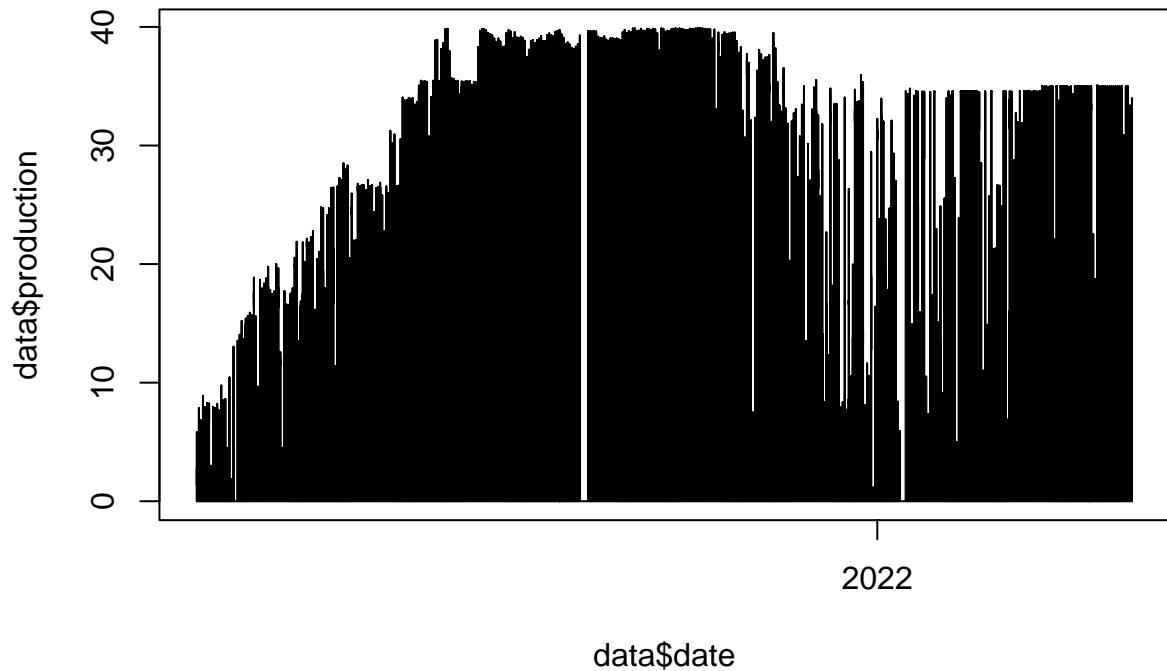
Electricity Production April–May 2022



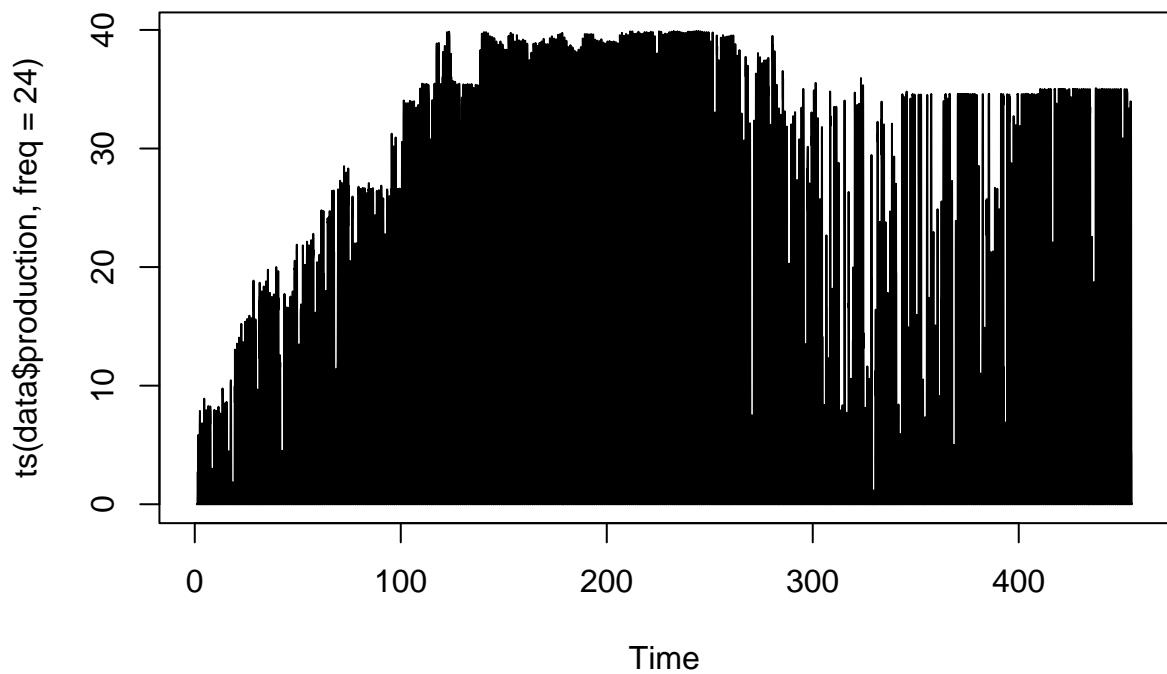
Then, the data is plotted to see if there is a part of the data which should be removed before creating the model.

```
plot(data$date, data$production, type="line", main="Plot of the Data")
```

Plot of the Data

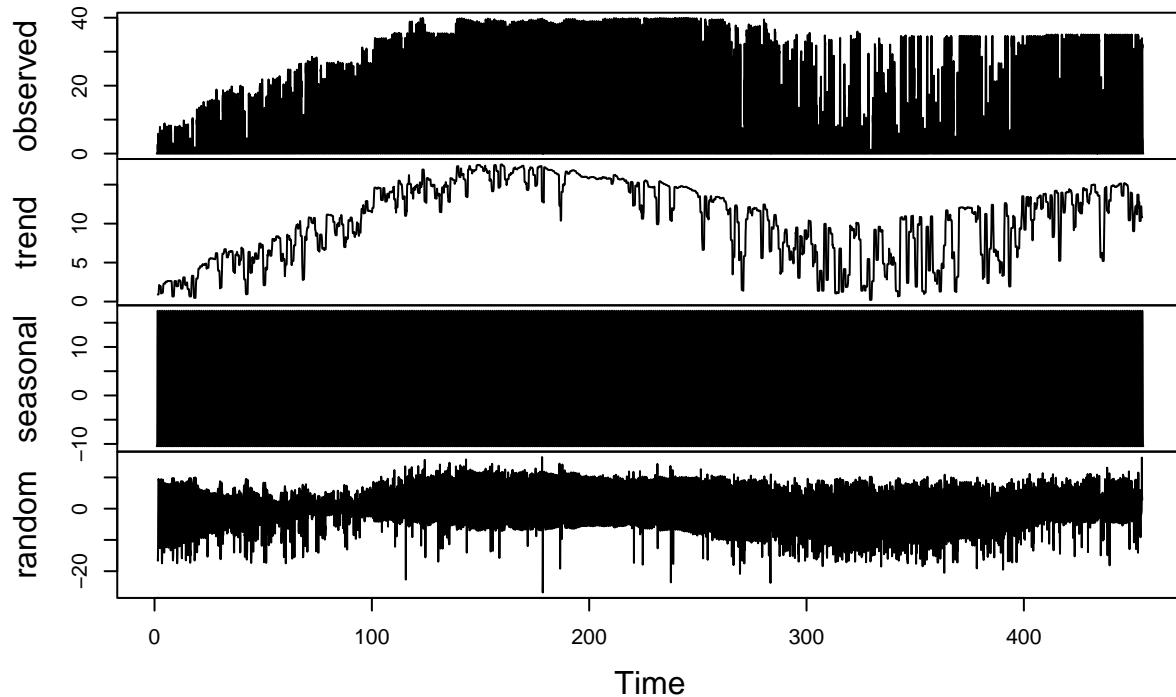


```
plot(ts(data$production,freq=24))
```



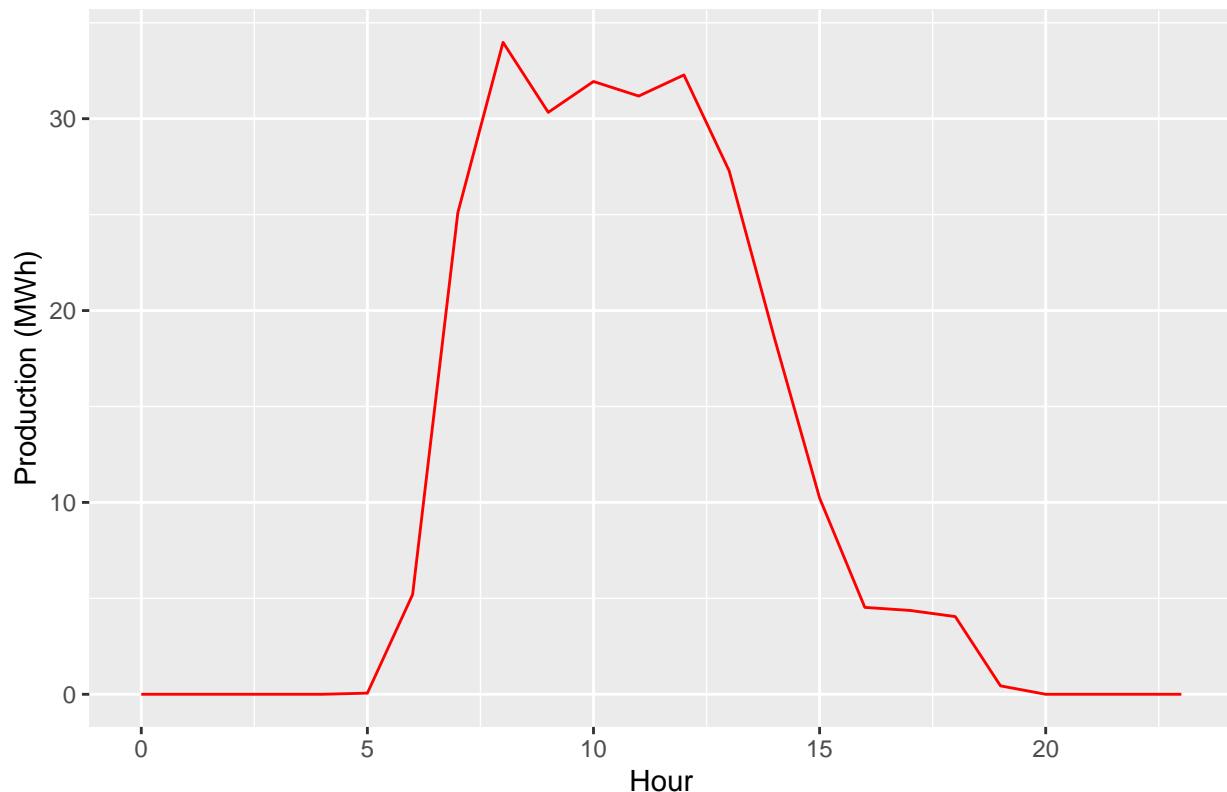
```
decomposed = decompose(ts(data$production,freq=24))
plot(decomposed)
```

Decomposition of additive time series



```
ggplot(data[date=="2022-05-06"], aes(x=hour, y= production)) +  
  geom_line(color= "red") +  
  labs(title = "Hourly Electricity Production Data in 06/05/22 ",  
       x = "Hour",  
       y= "Production (MWh)")
```

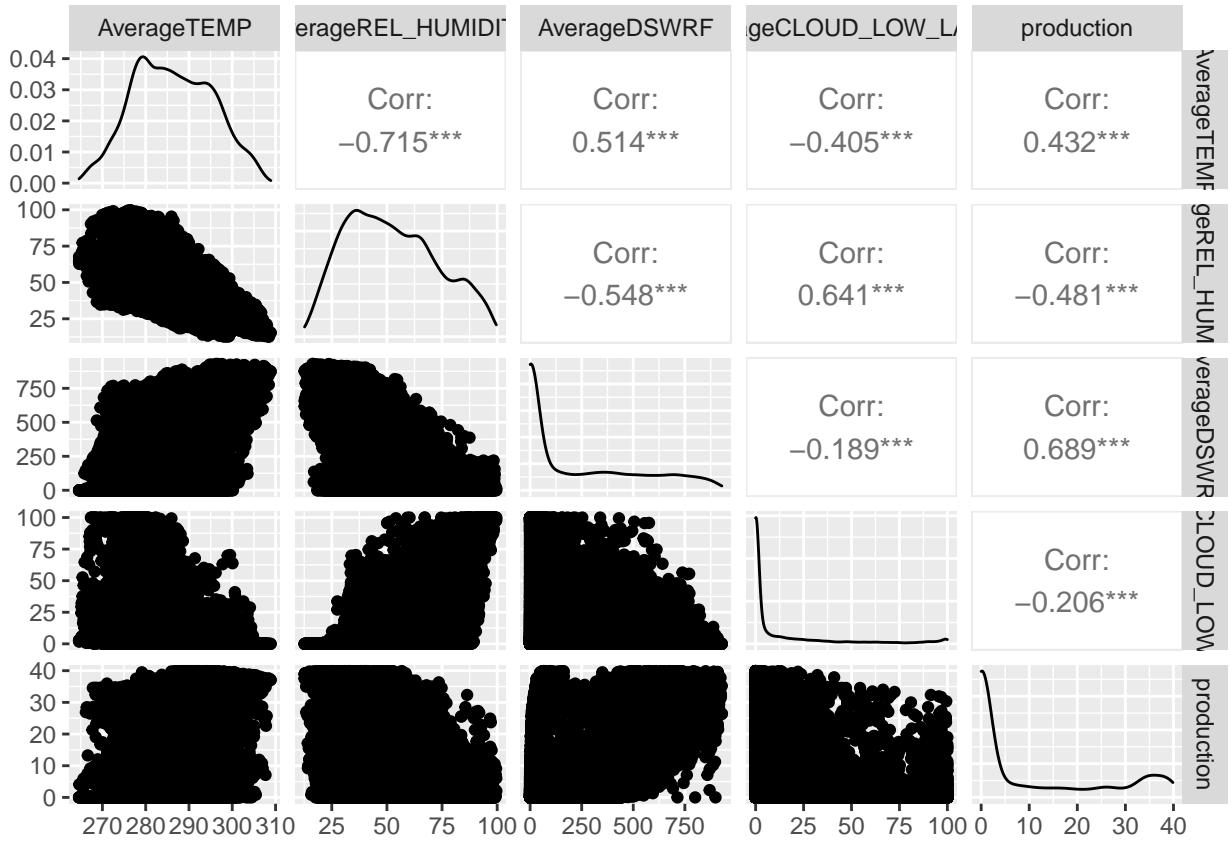
Hourly Electricity Production Data in 06/05/22



The additive decomposition plots show that the data is not stationary currently, which show there's some information in the data in order to create good predictions with it. The plots show clear hourly, yearly and dayly seasonality and trend; they should be dealt with.

Lastly, to investigate relations between averages of variables with production, correlation plot is created.

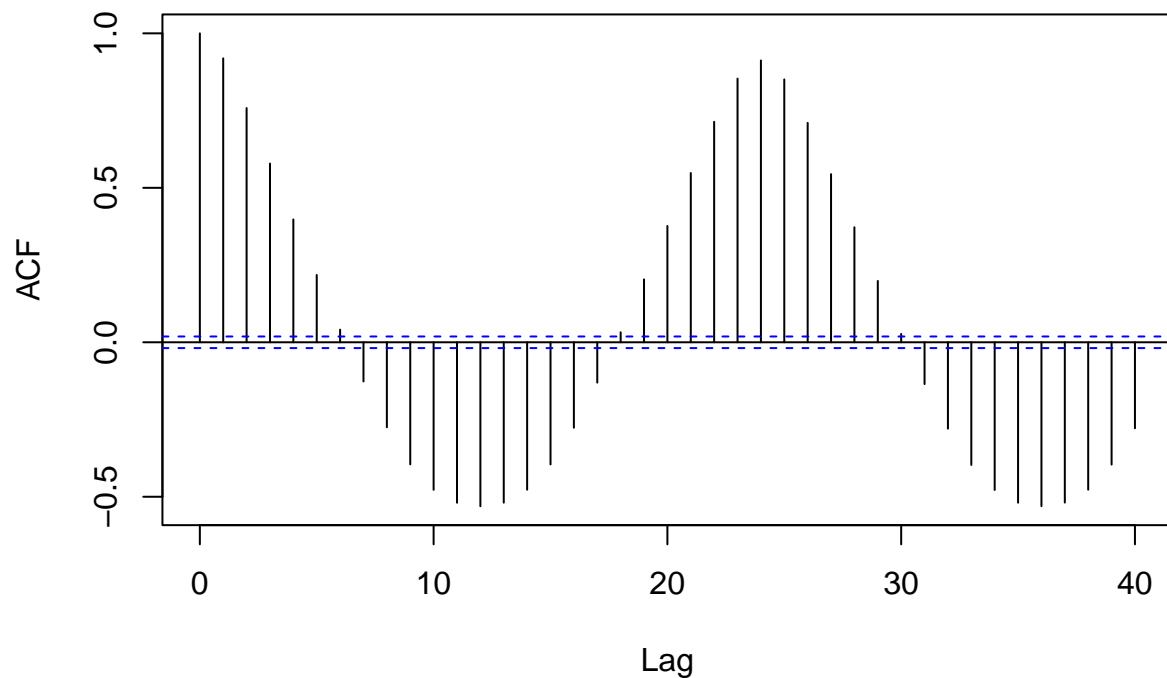
```
ggpairs(data,columns = c("AverageTEMP","AverageREL_HUMIDITY","AverageDSWRF","AverageCLOUD_LOW_LAYER","p
```



As there can be seen, there is a high and positive correlation between average DSWRF and production. After DSWRF, temperature has also a positive correlation with production.

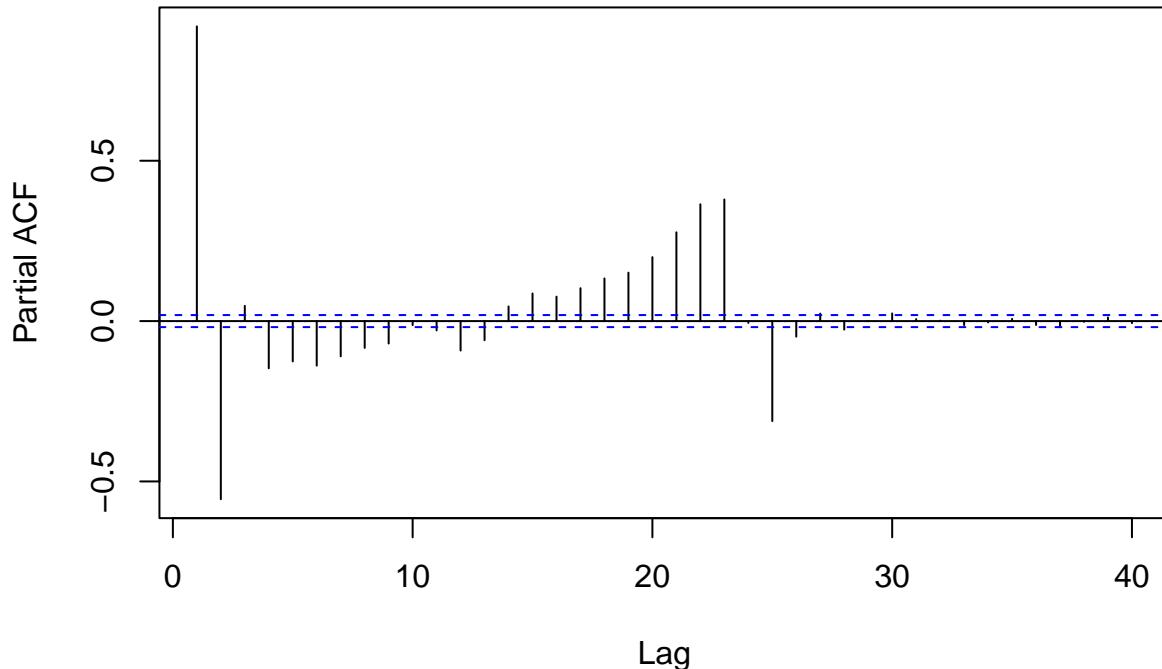
```
acf(data$production)
```

Series data\$production



```
pacf(data$production)
```

Series data\$production



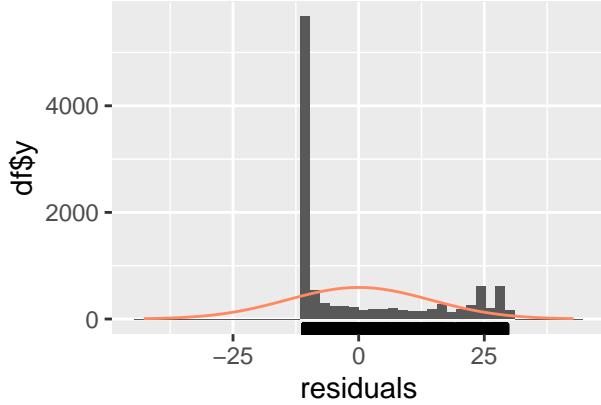
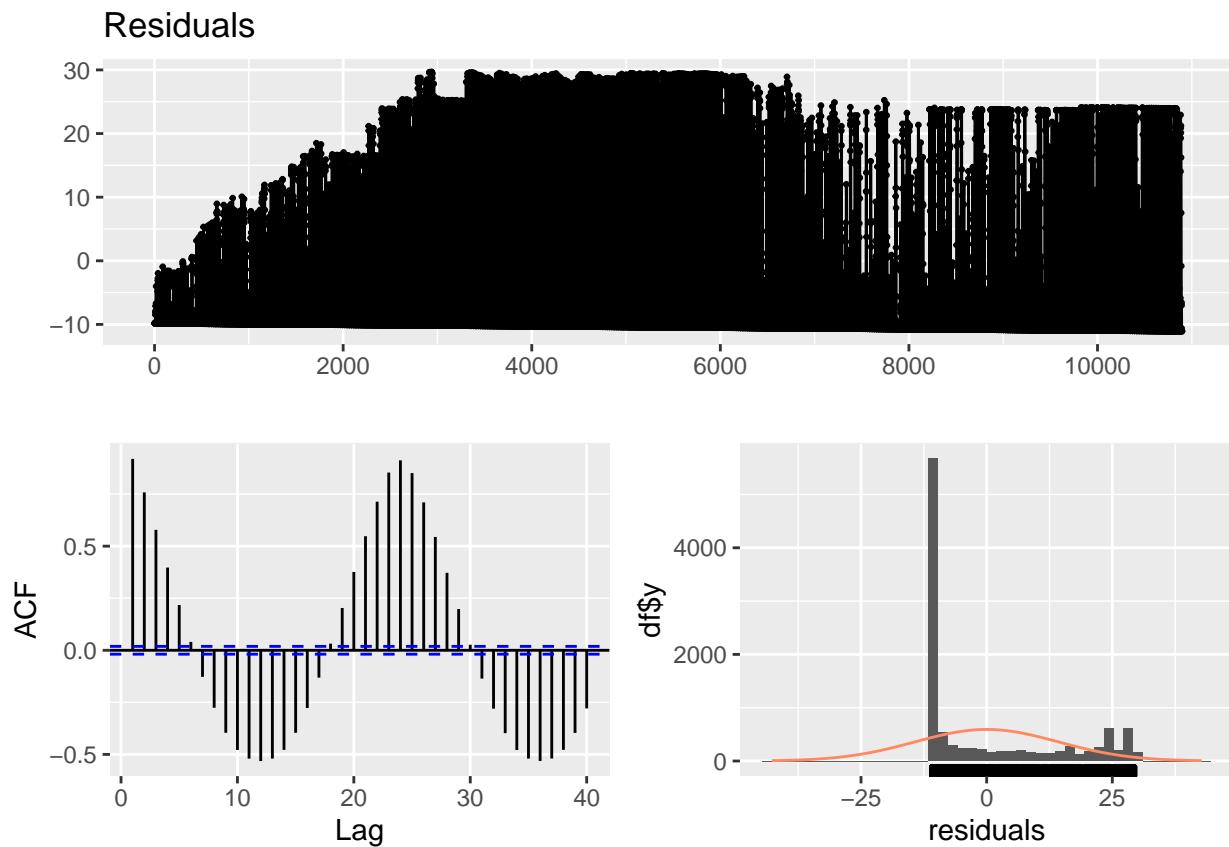
The autocorrelation function shows sinusoidal behavior while partial autocorrelation function show significance at the first two lags.

Models:

```
model1 <- lm(production~Trend, data)
summary(model1)

##
## Call:
## lm(formula = production ~ Trend, data = data)
##
## Residuals:
##      Min    1Q   Median    3Q    Max 
## -11.056 -10.489 -9.863  11.636  29.687 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 9.831e+00  2.731e-01  35.998 < 2e-16 ***
## Trend       1.125e-04  4.341e-05   2.591  0.00959 ** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 14.25 on 10894 degrees of freedom
## Multiple R-squared:  0.0006158, Adjusted R-squared:  0.000524 
## F-statistic: 6.712 on 1 and 10894 DF, p-value: 0.009587
```

```
checkresiduals(model1)
```



```
##  
## Breusch-Godfrey test for serial correlation of order up to 10  
##  
## data: Residuals  
## LM test = 9819.8, df = 10, p-value < 2.2e-16
```

```
AIC(model1)
```

```
## [1] 88824.38
```

```
BIC(model1)
```

```
## [1] 88846.26
```

```
model2 <- lm(production~Hour_factor, data)  
summary(model2)
```

```
##  
## Call:  
## lm(formula = production ~ Hour_factor, data = data)  
##
```

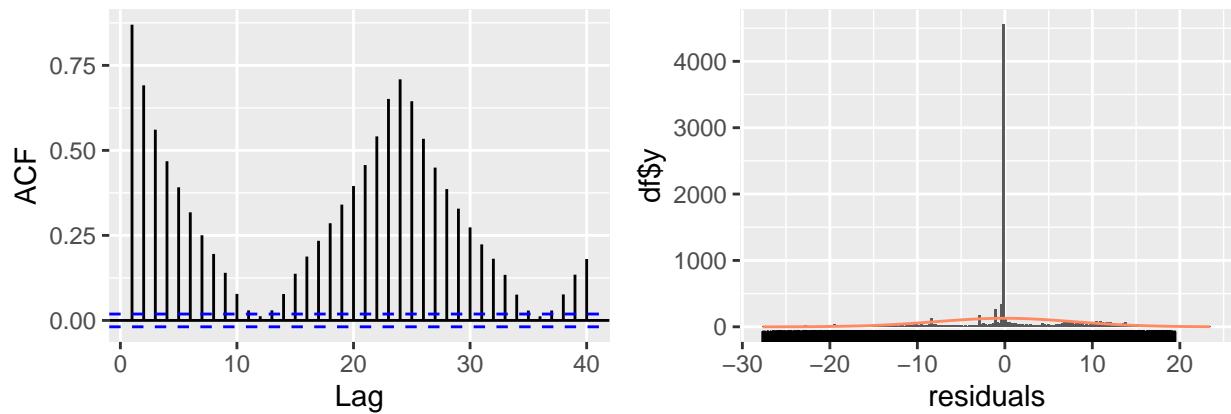
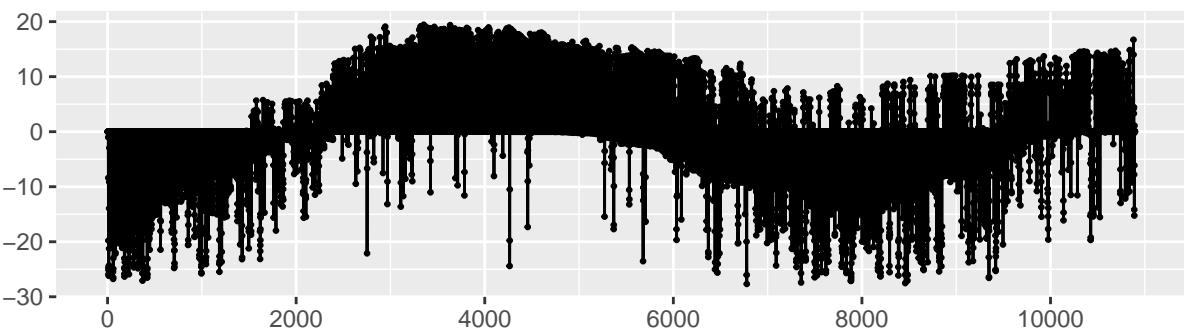
```

## Residuals:
##      Min     1Q Median     3Q    Max
## -27.6673 -0.6648  0.0000  1.9038 19.4437
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.163e-14 3.676e-01  0.000  1.0000
## Hour_factor1 3.126e-12 5.199e-01  0.000  1.0000
## Hour_factor2 7.462e-13 5.199e-01  0.000  1.0000
## Hour_factor3 3.537e-13 5.199e-01  0.000  1.0000
## Hour_factor4 -1.472e-12 5.199e-01  0.000  1.0000
## Hour_factor5 2.510e-02 5.199e-01  0.048  0.9615
## Hour_factor6 1.011e+00 5.199e-01  1.944  0.0519 .
## Hour_factor7 8.393e+00 5.199e-01 16.145 < 2e-16 ***
## Hour_factor8 1.997e+01 5.199e-01 38.423 < 2e-16 ***
## Hour_factor9 2.603e+01 5.199e-01 50.079 < 2e-16 ***
## Hour_factor10 2.767e+01 5.199e-01 53.219 < 2e-16 ***
## Hour_factor11 2.788e+01 5.199e-01 53.623 < 2e-16 ***
## Hour_factor12 2.783e+01 5.199e-01 53.541 < 2e-16 ***
## Hour_factor13 2.755e+01 5.199e-01 52.998 < 2e-16 ***
## Hour_factor14 2.636e+01 5.199e-01 50.712 < 2e-16 ***
## Hour_factor15 2.441e+01 5.199e-01 46.945 < 2e-16 ***
## Hour_factor16 1.977e+01 5.199e-01 38.035 < 2e-16 ***
## Hour_factor17 1.049e+01 5.199e-01 20.179 < 2e-16 ***
## Hour_factor18 2.940e+00 5.199e-01 5.654  1.6e-08 ***
## Hour_factor19 2.964e-01 5.199e-01 0.570  0.5687
## Hour_factor20 1.370e-03 5.199e-01 0.003  0.9979
## Hour_factor21 6.235e-15 5.199e-01 0.000  1.0000
## Hour_factor22 5.342e-15 5.199e-01 0.000  1.0000
## Hour_factor23 3.869e-15 5.199e-01 0.000  1.0000
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.833 on 10872 degrees of freedom
## Multiple R-squared:  0.6987, Adjusted R-squared:  0.6981
## F-statistic: 1096 on 23 and 10872 DF, p-value: < 2.2e-16

```

```
checkresiduals(model2)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 27  
##  
## data: Residuals  
## LM test = 9084.3, df = 27, p-value < 2.2e-16
```

```
AIC(model2)
```

```
## [1] 75802.03
```

```
BIC(model2)
```

```
## [1] 75984.44
```

```
model3 <- lm(production~Hour_factor+Month, data)  
summary(model3)
```

```
##  
## Call:  
## lm(formula = production ~ Hour_factor + Month, data = data)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -100000 -100000 -100000 -100000 -100000
```

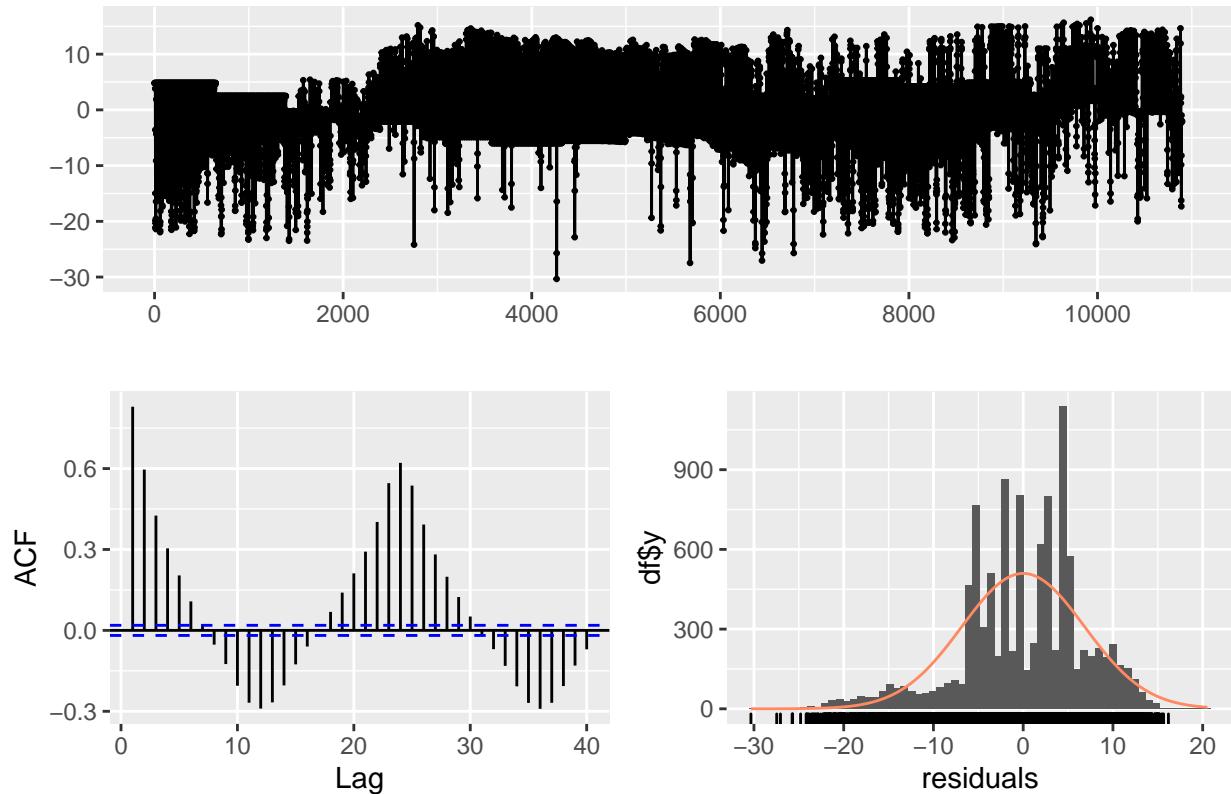
```

## -30.3370 -4.2010 0.3091 4.8116 16.1838
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.146e+00 4.082e-01 -10.155 < 2e-16 ***
## Hour_factor1 3.100e-12 4.549e-01 0.000 1.00000
## Hour_factor2 7.624e-13 4.549e-01 0.000 1.00000
## Hour_factor3 3.225e-13 4.549e-01 0.000 1.00000
## Hour_factor4 -1.562e-12 4.549e-01 0.000 1.00000
## Hour_factor5 2.510e-02 4.549e-01 0.055 0.95600
## Hour_factor6 1.011e+00 4.549e-01 2.222 0.02629 *
## Hour_factor7 8.393e+00 4.549e-01 18.451 < 2e-16 ***
## Hour_factor8 1.997e+01 4.549e-01 43.910 < 2e-16 ***
## Hour_factor9 2.603e+01 4.549e-01 57.230 < 2e-16 ***
## Hour_factor10 2.767e+01 4.549e-01 60.819 < 2e-16 ***
## Hour_factor11 2.788e+01 4.549e-01 61.281 < 2e-16 ***
## Hour_factor12 2.783e+01 4.549e-01 61.187 < 2e-16 ***
## Hour_factor13 2.755e+01 4.549e-01 60.566 < 2e-16 ***
## Hour_factor14 2.636e+01 4.549e-01 57.954 < 2e-16 ***
## Hour_factor15 2.441e+01 4.549e-01 53.649 < 2e-16 ***
## Hour_factor16 1.977e+01 4.549e-01 43.466 < 2e-16 ***
## Hour_factor17 1.049e+01 4.549e-01 23.061 < 2e-16 ***
## Hour_factor18 2.940e+00 4.549e-01 6.462 1.08e-10 ***
## Hour_factor19 2.964e-01 4.549e-01 0.651 0.51477
## Hour_factor20 1.370e-03 4.549e-01 0.003 0.99760
## Hour_factor21 -4.633e-14 4.549e-01 0.000 1.00000
## Hour_factor22 -4.054e-14 4.549e-01 0.000 1.00000
## Hour_factor23 -3.691e-14 4.549e-01 0.000 1.00000
## Month2      -6.661e-01 3.211e-01 -2.075 0.03804 *
## Month3      1.668e+00 3.147e-01 5.301 1.18e-07 ***
## Month4      4.470e+00 3.164e-01 14.128 < 2e-16 ***
## Month5      6.221e+00 3.470e-01 17.928 < 2e-16 ***
## Month6      8.983e+00 3.643e-01 24.655 < 2e-16 ***
## Month7      1.008e+01 3.614e-01 27.880 < 2e-16 ***
## Month8      9.679e+00 3.707e-01 26.111 < 2e-16 ***
## Month9      8.080e+00 3.643e-01 22.176 < 2e-16 ***
## Month10     6.163e+00 3.614e-01 17.051 < 2e-16 ***
## Month11     2.172e+00 3.643e-01 5.963 2.56e-09 ***
## Month12     -1.116e+00 3.614e-01 -3.089 0.00201 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.854 on 10861 degrees of freedom
## Multiple R-squared: 0.7696, Adjusted R-squared: 0.7688
## F-statistic: 1067 on 34 and 10861 DF, p-value: < 2.2e-16

```

```
checkresiduals(model3)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 38  
##  
## data: Residuals  
## LM test = 8553.8, df = 38, p-value < 2.2e-16
```

```
AIC(model3)
```

```
## [1] 72904.12
```

```
BIC(model3)
```

```
## [1] 73166.78
```

```
model4 <- lm(production~Hour_factor+Month+Year, data)  
summary(model4)
```

```
##  
## Call:  
## lm(formula = production ~ Hour_factor + Month + Year, data = data)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max
```

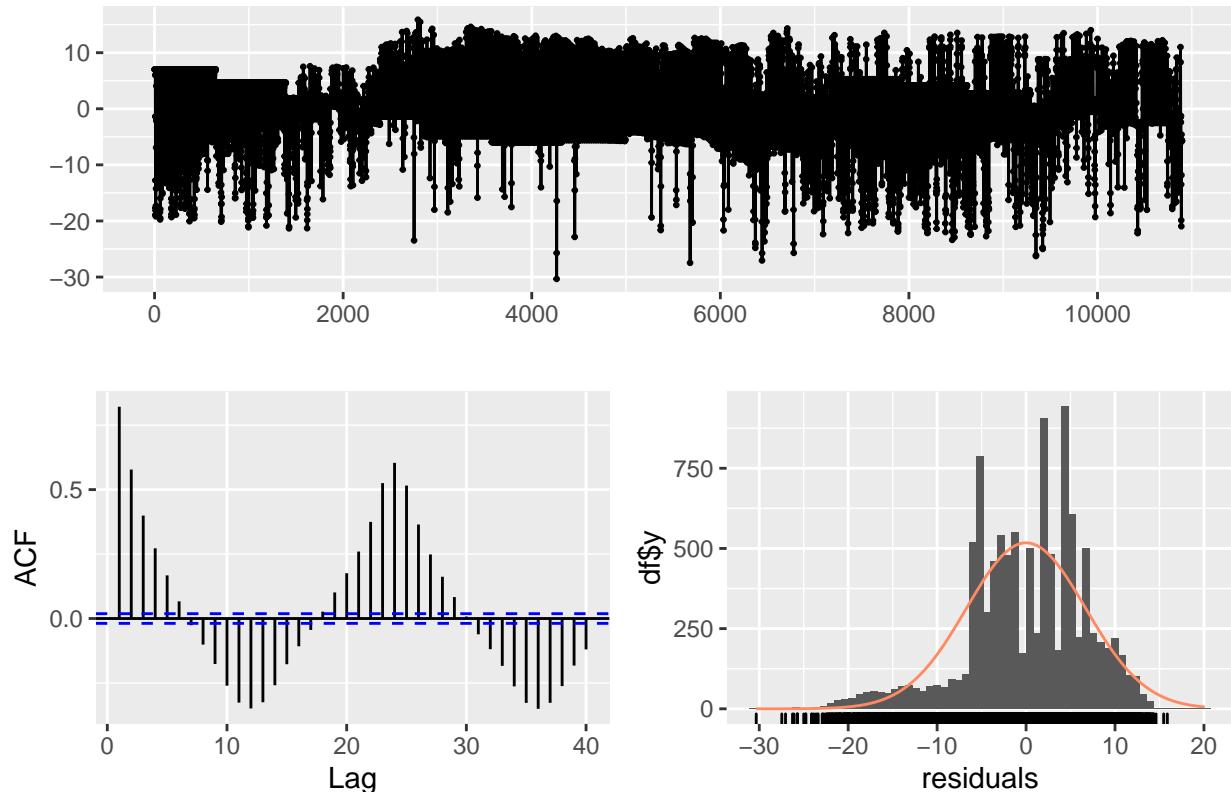
```

## -30.3370 -4.2304 0.4865 4.6429 15.8822
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.477e+00 4.443e-01 -19.079 < 2e-16 ***
## Hour_factor1 3.001e-12 4.449e-01 0.000 1.0000
## Hour_factor2 7.498e-13 4.449e-01 0.000 1.0000
## Hour_factor3 3.503e-13 4.449e-01 0.000 1.0000
## Hour_factor4 -1.583e-12 4.449e-01 0.000 1.0000
## Hour_factor5 2.510e-02 4.449e-01 0.056 0.9550
## Hour_factor6 1.011e+00 4.449e-01 2.272 0.0231 *
## Hour_factor7 8.393e+00 4.449e-01 18.865 < 2e-16 ***
## Hour_factor8 1.997e+01 4.449e-01 44.895 < 2e-16 ***
## Hour_factor9 2.603e+01 4.449e-01 58.514 < 2e-16 ***
## Hour_factor10 2.767e+01 4.449e-01 62.184 < 2e-16 ***
## Hour_factor11 2.788e+01 4.449e-01 62.655 < 2e-16 ***
## Hour_factor12 2.783e+01 4.449e-01 62.560 < 2e-16 ***
## Hour_factor13 2.755e+01 4.449e-01 61.925 < 2e-16 ***
## Hour_factor14 2.636e+01 4.449e-01 59.254 < 2e-16 ***
## Hour_factor15 2.441e+01 4.449e-01 54.853 < 2e-16 ***
## Hour_factor16 1.977e+01 4.449e-01 44.441 < 2e-16 ***
## Hour_factor17 1.049e+01 4.449e-01 23.578 < 2e-16 ***
## Hour_factor18 2.940e+00 4.449e-01 6.607 4.11e-11 ***
## Hour_factor19 2.964e-01 4.449e-01 0.666 0.5054
## Hour_factor20 1.370e-03 4.449e-01 0.003 0.9975
## Hour_factor21 -6.926e-14 4.449e-01 0.000 1.0000
## Hour_factor22 -6.902e-14 4.449e-01 0.000 1.0000
## Hour_factor23 -4.490e-14 4.449e-01 0.000 1.0000
## Month2      1.460e+00 3.283e-01 4.448 8.73e-06 ***
## Month3      3.834e+00 3.229e-01 11.874 < 2e-16 ***
## Month4      6.636e+00 3.245e-01 20.453 < 2e-16 ***
## Month5      9.850e+00 3.766e-01 26.153 < 2e-16 ***
## Month6      1.331e+01 4.062e-01 32.779 < 2e-16 ***
## Month7      1.441e+01 4.037e-01 35.692 < 2e-16 ***
## Month8      1.401e+01 4.116e-01 34.037 < 2e-16 ***
## Month9      1.241e+01 4.062e-01 30.556 < 2e-16 ***
## Month10     1.049e+01 4.037e-01 25.996 < 2e-16 ***
## Month11     6.504e+00 4.062e-01 16.013 < 2e-16 ***
## Month12     3.215e+00 4.037e-01 7.964 1.83e-15 ***
## Year2022    4.332e+00 1.949e-01 22.221 < 2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ',' 1
##
## Residual standard error: 6.704 on 10860 degrees of freedom
## Multiple R-squared: 0.7796, Adjusted R-squared: 0.7789
## F-statistic: 1097 on 35 and 10860 DF, p-value: < 2.2e-16

```

```
checkresiduals(model4)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 39  
##  
## data: Residuals  
## LM test = 8458.9, df = 39, p-value < 2.2e-16
```

```
AIC(model4)
```

```
## [1] 72421.66
```

```
BIC(model4)
```

```
## [1] 72691.62
```

```
model5 <- lm(production~Hour_factor+Month+Year+Trend, data)  
summary(model5)
```

```
##  
## Call:  
## lm(formula = production ~ Hour_factor + Month + Year + Trend,  
##      data = data)  
##  
## Residuals:
```

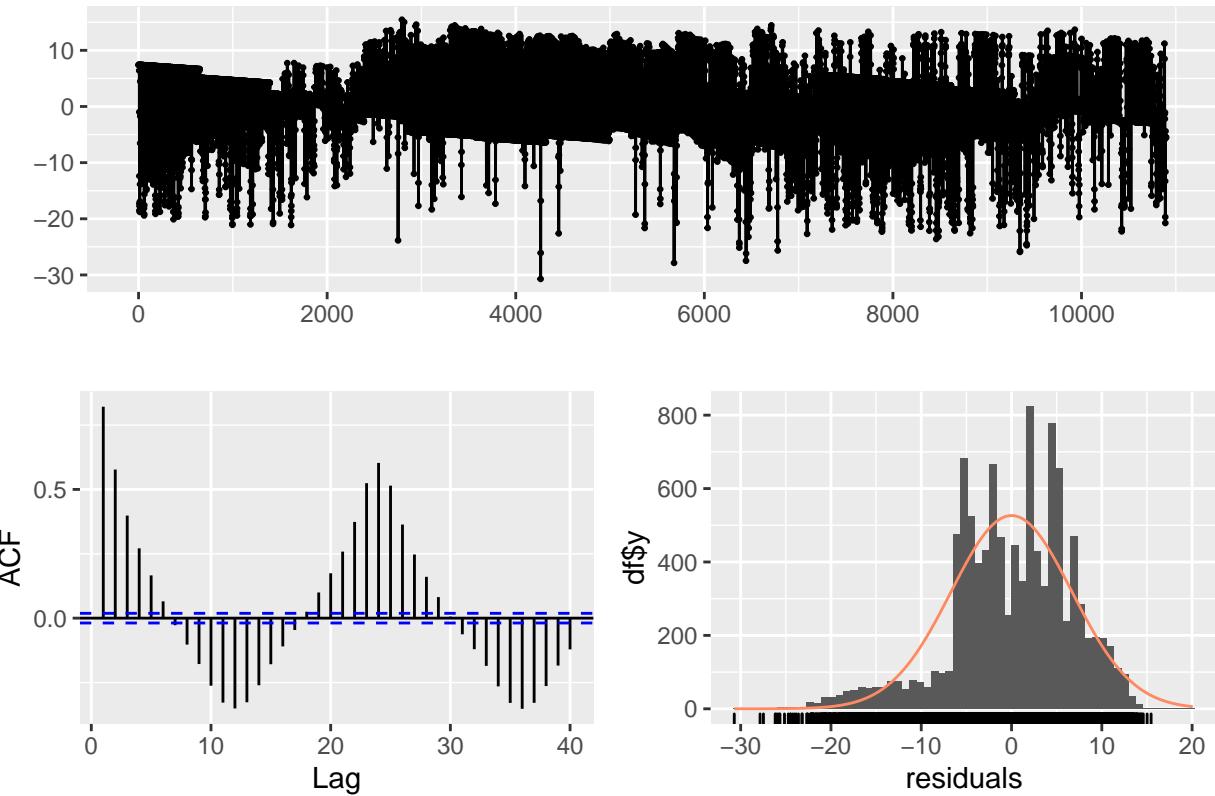
```

##      Min     1Q   Median     3Q    Max
## -30.7113 -4.2507  0.6076  4.7898 15.4562
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.0584033  0.4568902 -17.638 < 2e-16 ***
## Hour_factor1 -0.0011996  0.4446398 -0.003 0.997847
## Hour_factor2 -0.0023992  0.4446401 -0.005 0.995695
## Hour_factor3 -0.0035989  0.4446406 -0.008 0.993542
## Hour_factor4 -0.0047985  0.4446414 -0.011 0.991390
## Hour_factor5  0.0191031  0.4446423  0.043 0.965732
## Hour_factor6  1.0036926  0.4446435  2.257 0.024009 *
## Hour_factor7  8.3850900  0.4446449 18.858 < 2e-16 ***
## Hour_factor8 19.9653318  0.4446465 44.902 < 2e-16 ***
## Hour_factor9 26.0236963  0.4446483 58.526 < 2e-16 ***
## Hour_factor10 27.6553305  0.4446504 62.196 < 2e-16 ***
## Hour_factor11 27.8640225  0.4446526 62.665 < 2e-16 ***
## Hour_factor12 27.8203512  0.4446551 62.566 < 2e-16 ***
## Hour_factor13 27.5364941  0.4446577 61.927 < 2e-16 ***
## Hour_factor14 26.3471388  0.4446606 59.252 < 2e-16 ***
## Hour_factor15 24.3876844  0.4446637 54.845 < 2e-16 ***
## Hour_factor16 19.7540949  0.4446670 44.424 < 2e-16 ***
## Hour_factor17 10.4703401  0.4446706 23.546 < 2e-16 ***
## Hour_factor18  2.9179594  0.4446743  6.562 5.55e-11 ***
## Hour_factor19  0.2735587  0.4446783  0.615 0.538447
## Hour_factor20 -0.0226223  0.4446824 -0.051 0.959428
## Hour_factor21 -0.0251920  0.4446868 -0.057 0.954824
## Hour_factor22 -0.0263916  0.4446914 -0.059 0.952676
## Hour_factor23 -0.0275912  0.4446962 -0.062 0.950528
## Month2        0.6527270  0.3882210  1.681 0.092728 .
## Month3        2.1846903  0.5328047  4.100 4.16e-05 ***
## Month4        4.1085771  0.7260929  5.658 1.57e-08 ***
## Month5        6.5163564  0.9358607  6.963 3.52e-12 ***
## Month6        9.0506697  1.1686062  7.745 1.04e-14 ***
## Month7        9.2668571  1.3817534  6.707 2.09e-11 ***
## Month8        8.0194716  1.5938497  5.032 4.94e-07 ***
## Month9        5.5853457  1.8007975  3.102 0.001930 **
## Month10       2.7904970  2.0208349  1.381 0.167349
## Month11       -2.0781295  2.2428956 -0.927 0.354187
## Month12       -6.2450931  2.4648022 -2.534 0.011300 *
## Year2022      -5.9923372  2.6607068 -2.252 0.024332 *
## Trend         0.0011996  0.0003083  3.891 0.000101 ***
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.699 on 10859 degrees of freedom
## Multiple R-squared:  0.7799, Adjusted R-squared:  0.7792
## F-statistic: 1069 on 36 and 10859 DF, p-value: < 2.2e-16

```

```
checkresiduals(model15)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 40  
##  
## data: Residuals  
## LM test = 8457.4, df = 40, p-value < 2.2e-16
```

```
AIC(model5)
```

```
## [1] 72408.48
```

```
BIC(model5)
```

```
## [1] 72685.73
```

Dummy variables for month, year, hour and trend component each make a contribution to the model, making residuals closer to normal with zero mean and constant variable assumptions and each increase adjusted R_squared.

```
model16 <- lm(production~Hour_factor+Month+Year+Trend+Lag1, data)  
summary(model16)
```

```
##  
## Call:
```

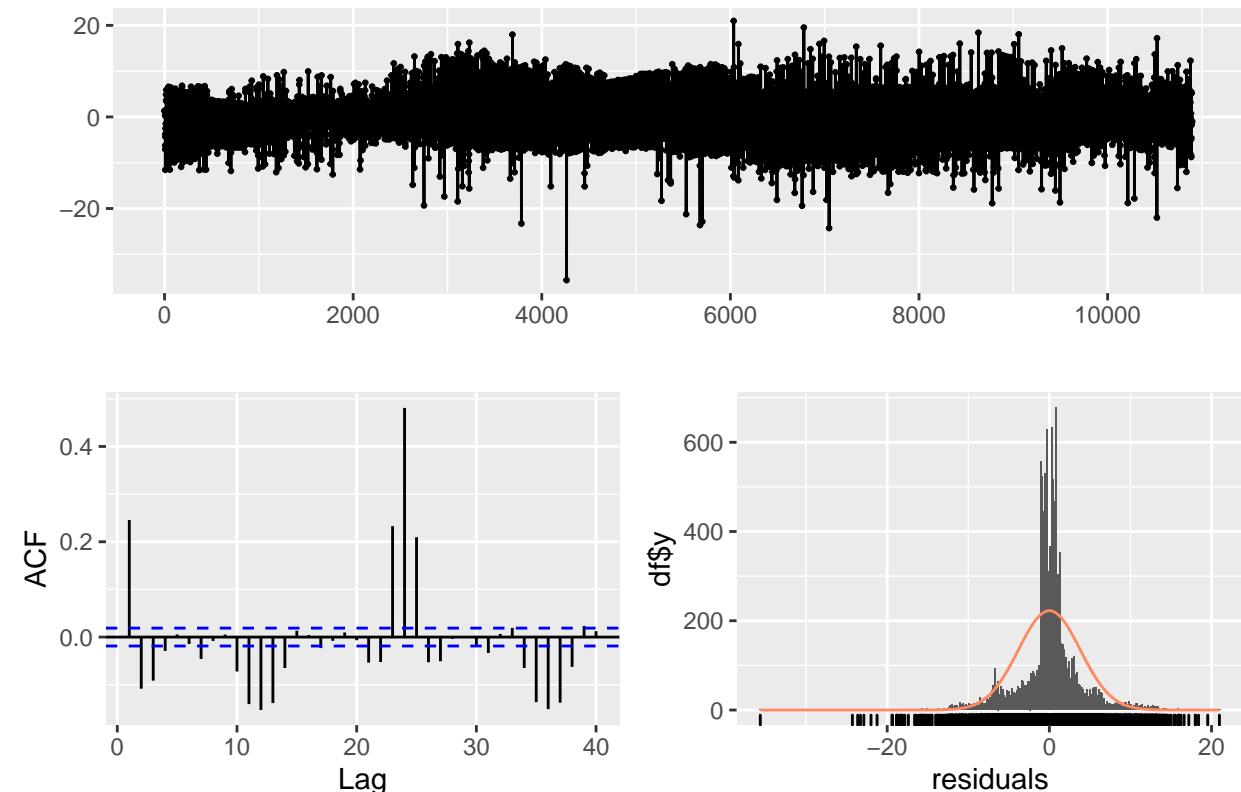
```

## lm(formula = production ~ Hour_factor + Month + Year + Trend +
##     Lag1, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -35.651  -0.984   0.103   1.186  20.969
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.4417670  0.2644242 -5.452 5.08e-08 ***
## Hour_factor1  0.0027064  0.2537093  0.011  0.9915
## Hour_factor2  0.0024913  0.2537094  0.010  0.9922
## Hour_factor3  0.0022762  0.2537096  0.009  0.9928
## Hour_factor4  0.0020612  0.2537100  0.008  0.9935
## Hour_factor5  0.0269472  0.2537104  0.106  0.9154
## Hour_factor6  0.9919016  0.2537111  3.910 9.30e-05 ***
## Hour_factor7  7.5644930  0.2537730 29.808 < 2e-16 ***
## Hour_factor8 13.0811799  0.2578442 50.733 < 2e-16 ***
## Hour_factor9  9.6267890  0.2762756 34.845 < 2e-16 ***
## Hour_factor10 6.2816926  0.2909976 21.587 < 2e-16 ***
## Hour_factor11 5.1500545  0.2954743 17.430 < 2e-16 ***
## Hour_factor12 4.9349493  0.2960624 16.669 < 2e-16 ***
## Hour_factor13 4.6869658  0.2959405 15.838 < 2e-16 ***
## Hour_factor14 3.7307879  0.2951452 12.641 < 2e-16 ***
## Hour_factor15 2.7483455  0.2918747  9.416 < 2e-16 ***
## Hour_factor16 -0.2756233  0.2867269 -0.961  0.3364
## Hour_factor17 -5.7530509  0.2758310 -20.857 < 2e-16 ***
## Hour_factor18 -5.6791599  0.2601361 -21.831 < 2e-16 ***
## Hour_factor19 -2.1195510  0.2542369 -8.337 < 2e-16 ***
## Hour_factor20 -0.2434522  0.2537369 -0.959  0.3373
## Hour_factor21 -0.0027207  0.2537344 -0.011  0.9914
## Hour_factor22 -0.0018103  0.2537369 -0.007  0.9943
## Hour_factor23 -0.0020254  0.2537396 -0.008  0.9936
## Month2        0.1151993  0.2214565  0.520  0.6029
## Month3        0.3890823  0.3040947  1.279  0.2008
## Month4        0.7319002  0.4147105  1.765  0.0776 .
## Month5        1.1613577  0.5349216  2.171  0.0299 *
## Month6        1.6132120  0.6683094  2.414  0.0158 *
## Month7        1.6511445  0.7896689  2.091  0.0366 *
## Month8        1.4277987  0.9100663  1.569  0.1167
## Month9        0.9925881  1.0274957  0.966  0.3341
## Month10       0.4929430  1.1526415  0.428  0.6689
## Month11       -0.3769457  1.2792457 -0.295  0.7683
## Month12       -1.1215624  1.4061773 -0.798  0.4251
## Year2022      -1.0770902  1.5178519 -0.710  0.4780
## Trend         0.0002151  0.0001760  1.222  0.2217
## Lag1          0.8214642  0.0054729 150.097 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.82 on 10857 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.9284, Adjusted R-squared:  0.9282
## F-statistic: 3806 on 37 and 10857 DF, p-value: < 2.2e-16

```

```
checkresiduals(model6)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 41  
##  
## data: Residuals  
## LM test = 3401.6, df = 41, p-value < 2.2e-16
```

```
AIC(model6)
```

```
## [1] 60164.98
```

```
BIC(model6)
```

```
## [1] 60449.52
```

```
model7 <- lm(production~Hour_factor+Month+Year+Trend+Lag1+Lag_day, data)  
summary(model7)
```

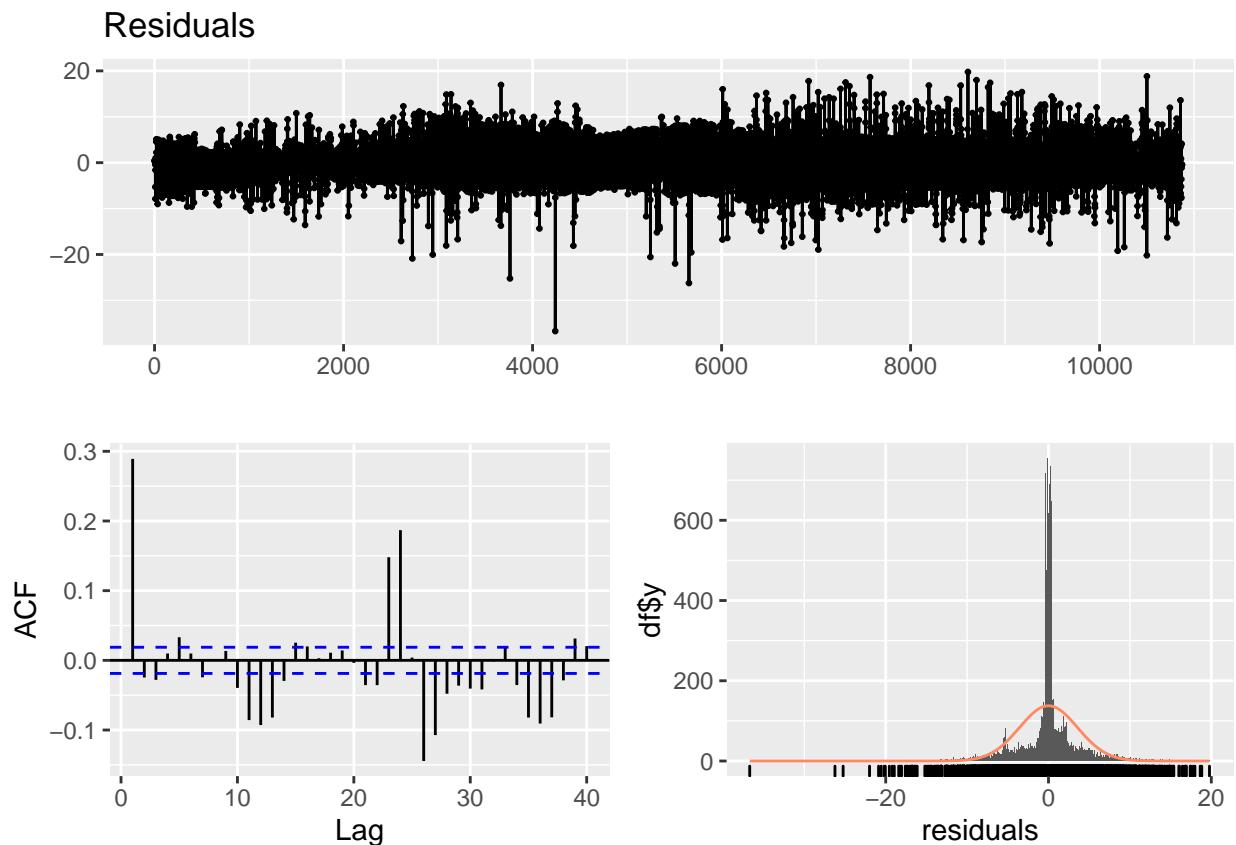
```
##  
## Call:  
## lm(formula = production ~ Hour_factor + Month + Year + Trend +  
##     Lag1 + Lag_day, data = data)
```

```

##
## Residuals:
##      Min     1Q Median     3Q    Max
## -36.705 -0.427  0.032  0.641 19.776
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.846e-01 2.479e-01 -2.359 0.01836 *
## Hour_factor1 7.149e-05 2.370e-01  0.000 0.99976
## Hour_factor2 1.430e-04 2.370e-01  0.001 0.99952
## Hour_factor3 2.145e-04 2.370e-01  0.001 0.99928
## Hour_factor4 2.860e-04 2.370e-01  0.001 0.99904
## Hour_factor5 1.950e-02 2.370e-01  0.082 0.93442
## Hour_factor6 7.555e-01 2.371e-01  3.186 0.00144 **
## Hour_factor7 5.702e+00 2.417e-01 23.591 < 2e-16 ***
## Hour_factor8 9.391e+00 2.583e-01 36.351 < 2e-16 ***
## Hour_factor9 5.947e+00 2.745e-01 21.664 < 2e-16 ***
## Hour_factor10 2.978e+00 2.846e-01 10.462 < 2e-16 ***
## Hour_factor11 2.000e+00 2.876e-01  6.956 3.71e-12 ***
## Hour_factor12 1.826e+00 2.878e-01  6.343 2.34e-10 ***
## Hour_factor13 1.638e+00 2.873e-01  5.700 1.23e-08 ***
## Hour_factor14 9.225e-01 2.850e-01  3.237 0.00121 **
## Hour_factor15 2.520e-01 2.801e-01  0.899 0.36842
## Hour_factor16 -1.921e+00 2.713e-01 -7.081 1.52e-12 ***
## Hour_factor17 -5.753e+00 2.578e-01 -22.316 < 2e-16 ***
## Hour_factor18 -5.052e+00 2.436e-01 -20.740 < 2e-16 ***
## Hour_factor19 -1.817e+00 2.376e-01 -7.646 2.24e-14 ***
## Hour_factor20 -2.036e-01 2.370e-01 -0.859 0.39036
## Hour_factor21 5.485e-04 2.370e-01  0.002 0.99815
## Hour_factor22 1.573e-03 2.370e-01  0.007 0.99471
## Hour_factor23 1.644e-03 2.370e-01  0.007 0.99447
## Month2        1.792e-01 2.075e-01  0.863 0.38797
## Month3        3.683e-01 2.842e-01  1.296 0.19500
## Month4        5.920e-01 3.878e-01  1.526 0.12692
## Month5        8.864e-01 5.003e-01  1.772 0.07645 .
## Month6        1.151e+00 6.251e-01  1.842 0.06557 .
## Month7        1.247e+00 7.387e-01  1.687 0.09156 .
## Month8        1.263e+00 8.514e-01  1.484 0.13783
## Month9        1.187e+00 9.613e-01  1.235 0.21703
## Month10       1.103e+00 1.079e+00  1.022 0.30665
## Month11       9.576e-01 1.197e+00  0.800 0.42385
## Month12       7.195e-01 1.316e+00  0.547 0.58472
## Year2022      8.960e-01 1.421e+00  0.630 0.52841
## Trend         -7.149e-05 1.649e-04 -0.434 0.66460
## Lag1          6.939e-01 6.032e-03 115.043 < 2e-16 ***
## Lag_day        2.402e-01 6.022e-03 39.891 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.567 on 10833 degrees of freedom
##   (24 observations deleted due to missingness)
## Multiple R-squared:  0.9377, Adjusted R-squared:  0.9375
## F-statistic:  4290 on 38 and 10833 DF,  p-value: < 2.2e-16

```

```
checkresiduals(model17)
```



```
##  
## Breusch-Godfrey test for serial correlation of order up to 42  
##  
## data: Residuals  
## LM test = 2201.5, df = 42, p-value < 2.2e-16
```

```
AIC(model17)
```

```
## [1] 58546.93
```

```
BIC(model17)
```

```
## [1] 58838.68
```

```
model18 <- lm(production~Hour_factor+Month+Year+Trend+Lag1+Lag_week+Lag_day, data)  
summary(model18)
```

```
##  
## Call:  
## lm(formula = production ~ Hour_factor + Month + Year + Trend +  
##     Lag1 + Lag_week + Lag_day, data = data)
```

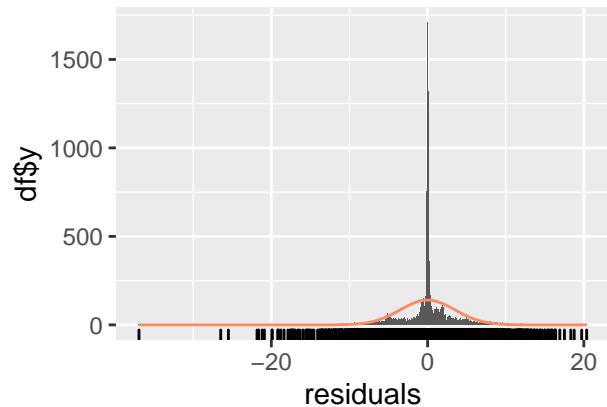
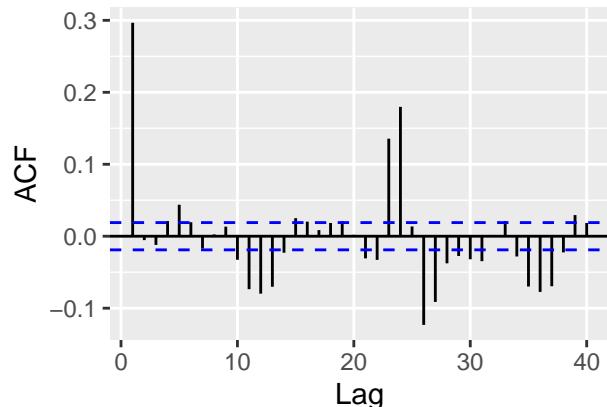
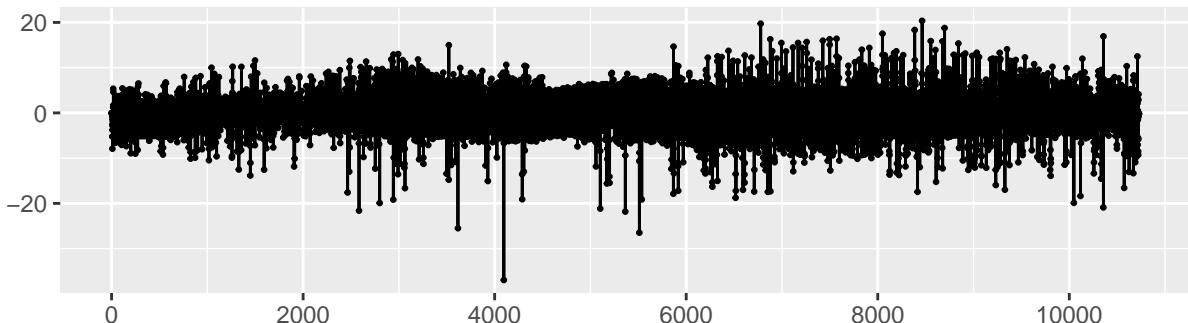
```

##
## Residuals:
##      Min     1Q Median     3Q    Max
## -36.939 -0.398  0.001  0.745 20.361
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.1752621  0.2459746 -0.713  0.47616
## Hour_factor1  0.0002648  0.2347595  0.001  0.99910
## Hour_factor2  0.0005297  0.2347597  0.002  0.99820
## Hour_factor3  0.0007945  0.2347600  0.003  0.99730
## Hour_factor4  0.0010594  0.2347604  0.005  0.99640
## Hour_factor5  0.0187726  0.2347609  0.080  0.93627
## Hour_factor6  0.6936325  0.2348648  2.953  0.00315 ***
## Hour_factor7  5.1798796  0.2414369 21.454 < 2e-16 ***
## Hour_factor8  8.2417698  0.2643670 31.175 < 2e-16 ***
## Hour_factor9  4.6136925  0.2819794 16.362 < 2e-16 ***
## Hour_factor10 1.6873894  0.2914684  5.789 7.27e-09 ***
## Hour_factor11 0.7371912  0.2940021  2.507  0.01218 *
## Hour_factor12 0.5507862  0.2942277  1.872  0.06124 .
## Hour_factor13 0.3934491  0.2934229  1.341  0.17998
## Hour_factor14 -0.2633223  0.2902959 -0.907  0.36438
## Hour_factor15 -0.8156499  0.2842695 -2.869  0.00412 **
## Hour_factor16 -2.7035864  0.2727440 -9.913 < 2e-16 ***
## Hour_factor17 -5.9976207  0.2562296 -23.407 < 2e-16 ***
## Hour_factor18 -5.0067617  0.2414939 -20.732 < 2e-16 ***
## Hour_factor19 -1.7705045  0.2354164 -7.521 5.88e-14 ***
## Hour_factor20 -0.1935733  0.2347899 -0.824  0.40970
## Hour_factor21  0.0046380  0.2347848  0.020  0.98424
## Hour_factor22  0.0058266  0.2347873  0.025  0.98020
## Hour_factor23  0.0060914  0.2347899  0.026  0.97930
## Month2        0.2632169  0.2085440  1.262  0.20692
## Month3        0.4717311  0.2807094  1.680  0.09289 .
## Month4        0.6799372  0.3839411  1.771  0.07660 .
## Month5        0.9130761  0.4949861  1.845  0.06512 .
## Month6        1.1872906  0.6188888  1.918  0.05508 .
## Month7        1.2894638  0.7319553  1.762  0.07815 .
## Month8        1.4678309  0.8440602  1.739  0.08206 .
## Month9        1.5600370  0.9535314  1.636  0.10186
## Month10       1.6357000  1.0702943  1.528  0.12647
## Month11       1.8647378  1.1889689  1.568  0.11683
## Month12       1.8984348  1.3079124  1.451  0.14667
## Year2022      2.3104517  1.4127153  1.635  0.10198
## Trend         -0.0002648  0.0001643 -1.612  0.10702
## Lag1          0.6638716  0.0061568 107.827 < 2e-16 ***
## Lag_week       0.1235608  0.0059751 20.679 < 2e-16 ***
## Lag_day        0.1937060  0.0063178 30.660 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.51 on 10688 degrees of freedom
##   (168 observations deleted due to missingness)
## Multiple R-squared:  0.9402, Adjusted R-squared:  0.94
## F-statistic: 4307 on 39 and 10688 DF, p-value: < 2.2e-16

```

```
checkresiduals(model18)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 43  
##  
## data: Residuals  
## LM test = 2133.4, df = 43, p-value < 2.2e-16
```

```
AIC(model18)
```

```
## [1] 57424.93
```

```
BIC(model18)
```

```
## [1] 57723.44
```

The autoregressive lags of one hour, one week and one day each make a contribution to the model and increase the adjusted R-squared value.

```
model19 <- lm(production ~ . - AverageTEMP - AverageREL_HUMIDITY - AverageDSWRF - AverageCLOUD_LOW_LAYER, data)  
summary(model19)
```

```

## 
## Call:
## lm(formula = production ~ . - AverageTEMP - AverageREL_HUMIDITY -
##      AverageDSWRF - AverageCLOUD_LOW_LAYER, data = data)
##
## Residuals:
##    Min      1Q  Median      3Q     Max
## -37.304  -0.708 -0.061   1.113  20.389
##
## Coefficients: (2 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                 9.365e+02  1.554e+03  0.602  0.546855  
## date                      -4.993e-02  8.326e-02 -0.600  0.548711  
## hour                       4.026e-04  1.063e-02  0.038  0.969792  
## CLOUD_LOW_LAYER_36.25_33   -2.478e-03  3.100e-03 -0.800  0.424016  
## DSWRF_36.25_33            -2.122e-03  1.603e-03 -1.324  0.185628  
## REL_HUMIDITY_36.25_33     1.134e-02  9.506e-03  1.193  0.233052  
## TEMP_36.25_33              1.431e-01  7.327e-02  1.953  0.050800 .  
## CLOUD_LOW_LAYER_36.25_33.25 -1.114e-03  4.212e-03 -0.264  0.791439  
## DSWRF_36.25_33.25         1.745e-03  2.146e-03  0.813  0.416046  
## REL_HUMIDITY_36.25_33.25  -1.880e-02  1.414e-02 -1.329  0.183842  
## TEMP_36.25_33.25          1.352e-01  1.049e-01  1.289  0.197430  
## CLOUD_LOW_LAYER_36.25_33.5 -1.616e-03  3.470e-03 -0.466  0.641576  
## DSWRF_36.25_33.5          -1.301e-03  1.716e-03 -0.758  0.448369  
## REL_HUMIDITY_36.25_33.5   8.999e-03  9.861e-03  0.913  0.361458  
## TEMP_36.25_33.5           -1.098e-01  7.020e-02 -1.564  0.117846  
## CLOUD_LOW_LAYER_36.5_33   -4.014e-03  3.089e-03 -1.299  0.193899  
## DSWRF_36.5_33              7.490e-03  1.539e-03  4.869  1.14e-06 ***  
## REL_HUMIDITY_36.5_33      -3.216e-04  9.048e-03 -0.036  0.971649  
## TEMP_36.5_33                -1.502e-01  5.770e-02 -2.603  0.009264 **  
## CLOUD_LOW_LAYER_36.5_33.25 -2.968e-03  3.997e-03 -0.743  0.457717  
## DSWRF_36.5_33.25          -5.485e-03  1.926e-03 -2.847  0.004421 **  
## REL_HUMIDITY_36.5_33.25   -1.939e-02  1.365e-02 -1.420  0.155684  
## TEMP_36.5_33.25            -6.633e-02  9.590e-02 -0.692  0.489194  
## CLOUD_LOW_LAYER_36.5_33.5 -9.666e-03  3.804e-03 -2.541  0.011064 *  
## DSWRF_36.5_33.5            2.852e-03  1.813e-03  1.574  0.115622  
## REL_HUMIDITY_36.5_33.5    6.588e-03  1.132e-02  0.582  0.560635  
## TEMP_36.5_33.5             -3.248e-02  6.162e-02 -0.527  0.598164  
## CLOUD_LOW_LAYER_36.75_33   1.174e-03  2.637e-03  0.445  0.656200  
## DSWRF_36.75_33              6.920e-05  1.306e-03  0.053  0.957752  
## REL_HUMIDITY_36.75_33     1.938e-02  8.456e-03  2.292  0.021919 *  
## TEMP_36.75_33                1.888e-01  6.374e-02  2.963  0.003058 **  
## CLOUD_LOW_LAYER_36.75_33.25 -2.256e-03  4.249e-03 -0.531  0.595360  
## DSWRF_36.75_33.25          -7.001e-04  1.993e-03 -0.351  0.725370  
## REL_HUMIDITY_36.75_33.25   -2.921e-02  1.182e-02 -2.472  0.013459 *  
## TEMP_36.75_33.25            -2.784e-01  8.457e-02 -3.292  0.000998 ***  
## CLOUD_LOW_LAYER_36.75_33.5  3.943e-04  3.406e-03  0.116  0.907844  
## DSWRF_36.75_33.5            -3.354e-03  1.566e-03 -2.142  0.032217 *  
## REL_HUMIDITY_36.75_33.5    2.688e-02  7.922e-03  3.393  0.000693 ***  
## TEMP_36.75_33.5              1.507e-01  6.747e-02  2.233  0.025544 *  
## Year2022                    3.951e+00  1.419e+00  2.784  0.005381 **  
## Month2                       4.089e-01  2.256e-01  1.812  0.069958 .  
## Month3                       1.010e+00  2.913e-01  3.467  0.000529 ***  
## Month4                       1.476e+00  4.017e-01  3.674  0.000240 ***

```

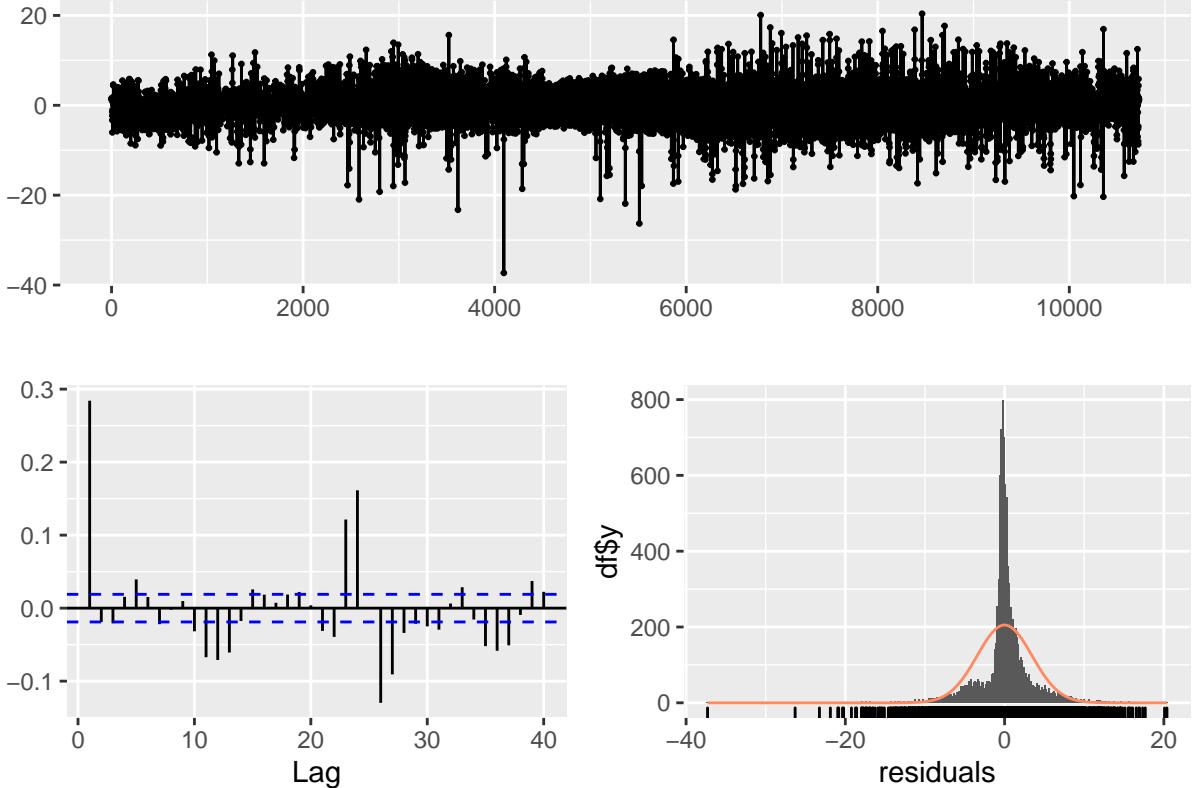
```

## Month5          1.605e+00  5.193e-01  3.090  0.002007 **
## Month6          1.931e+00  6.516e-01  2.963  0.003053 **
## Month7          2.210e+00  7.745e-01  2.853  0.004334 **
## Month8          2.454e+00  8.758e-01  2.803  0.005078 **
## Month9          2.613e+00  9.739e-01  2.683  0.007305 **
## Month10         2.569e+00  1.078e+00  2.382  0.017227 *
## Month11         3.000e+00  1.195e+00  2.511  0.012042 *
## Month12         3.521e+00  1.313e+00  2.681  0.007346 **
## Hour_factor1    -3.421e-02  2.262e-01 -0.151  0.879784
## Hour_factor2    -7.924e-02  2.219e-01 -0.357  0.721068
## Hour_factor3    -1.054e-01  2.182e-01 -0.483  0.629014
## Hour_factor4    -1.359e-01  2.152e-01 -0.631  0.527730
## Hour_factor5    -1.512e-01  2.126e-01 -0.711  0.476857
## Hour_factor6    4.846e-01  2.105e-01  2.302  0.021363 *
## Hour_factor7    4.897e+00  2.151e-01  22.763 < 2e-16 ***
## Hour_factor8    7.933e+00  2.394e-01  33.130 < 2e-16 ***
## Hour_factor9    4.475e+00  2.615e-01  17.115 < 2e-16 ***
## Hour_factor10   1.999e+00  2.961e-01  6.750  1.56e-11 ***
## Hour_factor11   1.246e+00  3.082e-01  4.043  5.32e-05 ***
## Hour_factor12   1.281e+00  3.176e-01  4.035  5.51e-05 ***
## Hour_factor13   1.350e+00  3.235e-01  4.172  3.05e-05 ***
## Hour_factor14   8.837e-01  3.249e-01  2.720  0.006544 **
## Hour_factor15   4.529e-01  3.218e-01  1.407  0.159374
## Hour_factor16   -1.367e+00 2.938e-01 -4.654  3.29e-06 ***
## Hour_factor17   -4.685e+00 2.716e-01 -17.247 < 2e-16 ***
## Hour_factor18   -3.869e+00 2.555e-01 -15.145 < 2e-16 ***
## Hour_factor19   -9.002e-01 2.461e-01 -3.658  0.000256 ***
## Hour_factor20   4.363e-01 2.400e-01  1.818  0.069042 .
## Hour_factor21   4.127e-01 2.367e-01  1.744  0.081253 .
## Hour_factor22   6.445e-02 2.264e-01  0.285  0.775859
## Hour_factor23          NA      NA      NA      NA
## max_in_month    -3.439e-03 2.795e-02 -0.123  0.902069
## max_in_week     4.343e-02 2.547e-02  1.705  0.088189 .
## night1          NA      NA      NA      NA
## Lag1            6.489e-01 7.012e-03  92.543 < 2e-16 ***
## Lag_week         1.373e-01 6.033e-03  22.764 < 2e-16 ***
## Lag_day          1.958e-01 6.304e-03  31.059 < 2e-16 ***
## Trend           1.585e-03 3.519e-03  0.450  0.652511
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.451 on 10649 degrees of freedom
##   (168 observations deleted due to missingness)
## Multiple R-squared:  0.9424, Adjusted R-squared:  0.9419
## F-statistic:  2232 on 78 and 10649 DF,  p-value: < 2.2e-16

```

```
checkresiduals(model9)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 84  
##  
## data: Residuals  
## LM test = 2541.7, df = 84, p-value < 2.2e-16
```

```
AIC(model19)
```

```
## [1] 57104.58
```

```
BIC(model19)
```

```
## [1] 57687.02
```

```
model10 <- lm(production ~ . - AverageTEMP - AverageREL_HUMIDITY - AverageDSWRF - AverageCLOUD_LOW_LAYER - DSWRF_36  
summary(model10)
```

```
##  
## Call:  
## lm(formula = production ~ . - AverageTEMP - AverageREL_HUMIDITY -  
##     AverageDSWRF - AverageCLOUD_LOW_LAYER - DSWRF_36.25_33.5 -  
##     CLOUD_LOW_LAYER_36.25_33 - CLOUD_LOW_LAYER_36.25_33.25 -  
##     CLOUD_LOW_LAYER_36.25_33.5 - REL_HUMIDITY_36.5_33 - REL_HUMIDITY_36.5_33.5 -
```

```

##      TEMP_36.5_33.5 - CLOUD_LOW_LAYER_36.75_33 - DSWRF_36.75_33 -
##      CLOUD_LOW_LAYER_36.75_33.25 - DSWRF_36.75_33.25 - CLOUD_LOW_LAYER_36.75_33.5,
##      data = data)
##
## Residuals:
##      Min     1Q   Median     3Q    Max
## -37.300 -0.711 -0.061  1.105 20.720
##
## Coefficients: (2 not defined because of singularities)
##                                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   9.658e+02  1.550e+03   0.623  0.533315
## date                      -5.146e-02  8.305e-02  -0.620  0.535461
## hour                       2.397e-04  1.062e-02   0.023  0.981996
## DSWRF_36.25_33            -1.402e-03  1.406e-03  -0.997  0.318964
## REL_HUMIDITY_36.25_33     8.654e-03  8.842e-03   0.979  0.327772
## TEMP_36.25_33              1.339e-01  6.979e-02   1.919  0.055055 .
## DSWRF_36.25_33.25         9.683e-04  1.455e-03   0.666  0.505744
## REL_HUMIDITY_36.25_33.25 -2.035e-02  1.382e-02  -1.473  0.140892
## TEMP_36.25_33.25          1.362e-01  1.026e-01   1.327  0.184640
## REL_HUMIDITY_36.25_33.5   1.158e-02  8.814e-03   1.313  0.189108
## TEMP_36.25_33.5           -1.183e-01  6.602e-02  -1.793  0.073042 .
## CLOUD_LOW_LAYER_36.5_33   -5.289e-03  2.541e-03  -2.082  0.037380 *
## DSWRF_36.5_33              6.965e-03  1.401e-03   4.972  6.73e-07 ***
## TEMP_36.5_33                -1.430e-01  4.650e-02  -3.076  0.002104 **
## CLOUD_LOW_LAYER_36.5_33.25 -4.107e-03  3.836e-03  -1.071  0.284395
## DSWRF_36.5_33.25           -5.504e-03  1.863e-03  -2.955  0.003131 **
## REL_HUMIDITY_36.5_33.25   -1.414e-02  9.243e-03  -1.530  0.126062
## TEMP_36.5_33.25             -8.931e-02  7.965e-02  -1.121  0.262204
## CLOUD_LOW_LAYER_36.5_33.5  -1.197e-02  3.367e-03  -3.554  0.000381 ***
## DSWRF_36.5_33.5             1.726e-03  1.659e-03   1.041  0.298107
## REL_HUMIDITY_36.75_33      2.003e-02  6.956e-03   2.879  0.003994 **
## TEMP_36.75_33                2.077e-01  5.789e-02   3.587  0.000335 ***
## REL_HUMIDITY_36.75_33.25   -3.002e-02  1.013e-02  -2.964  0.003045 **
## TEMP_36.75_33.25             -3.030e-01  7.859e-02  -3.855  0.000116 ***
## DSWRF_36.75_33.5             -3.586e-03  1.118e-03  -3.208  0.001339 **
## REL_HUMIDITY_36.75_33.5     2.709e-02  7.661e-03   3.536  0.000407 ***
## TEMP_36.75_33.5               1.545e-01  6.614e-02   2.335  0.019542 *
## Year2022                     4.078e+00  1.412e+00   2.889  0.003877 **
## Month2                        4.466e-01  2.232e-01   2.001  0.045369 *
## Month3                        1.044e+00  2.896e-01   3.604  0.000315 ***
## Month4                        1.524e+00  3.994e-01   3.815  0.000137 ***
## Month5                        1.669e+00  5.158e-01   3.235  0.001220 **
## Month6                        2.015e+00  6.475e-01   3.112  0.001862 **
## Month7                        2.294e+00  7.703e-01   2.979  0.002901 **
## Month8                        2.549e+00  8.707e-01   2.927  0.003425 **
## Month9                        2.738e+00  9.683e-01   2.828  0.004699 **
## Month10                       2.708e+00  1.072e+00   2.526  0.011557 *
## Month11                       3.129e+00  1.188e+00   2.633  0.008468 **
## Month12                       3.640e+00  1.306e+00   2.787  0.005331 **
## Hour_factor1                  -3.137e-02  2.260e-01  -0.139  0.889641
## Hour_factor2                  -7.446e-02  2.218e-01  -0.336  0.737050
## Hour_factor3                  -1.014e-01  2.180e-01  -0.465  0.641873
## Hour_factor4                  -1.312e-01  2.149e-01  -0.610  0.541694
## Hour_factor5                  -1.472e-01  2.122e-01  -0.694  0.487868

```

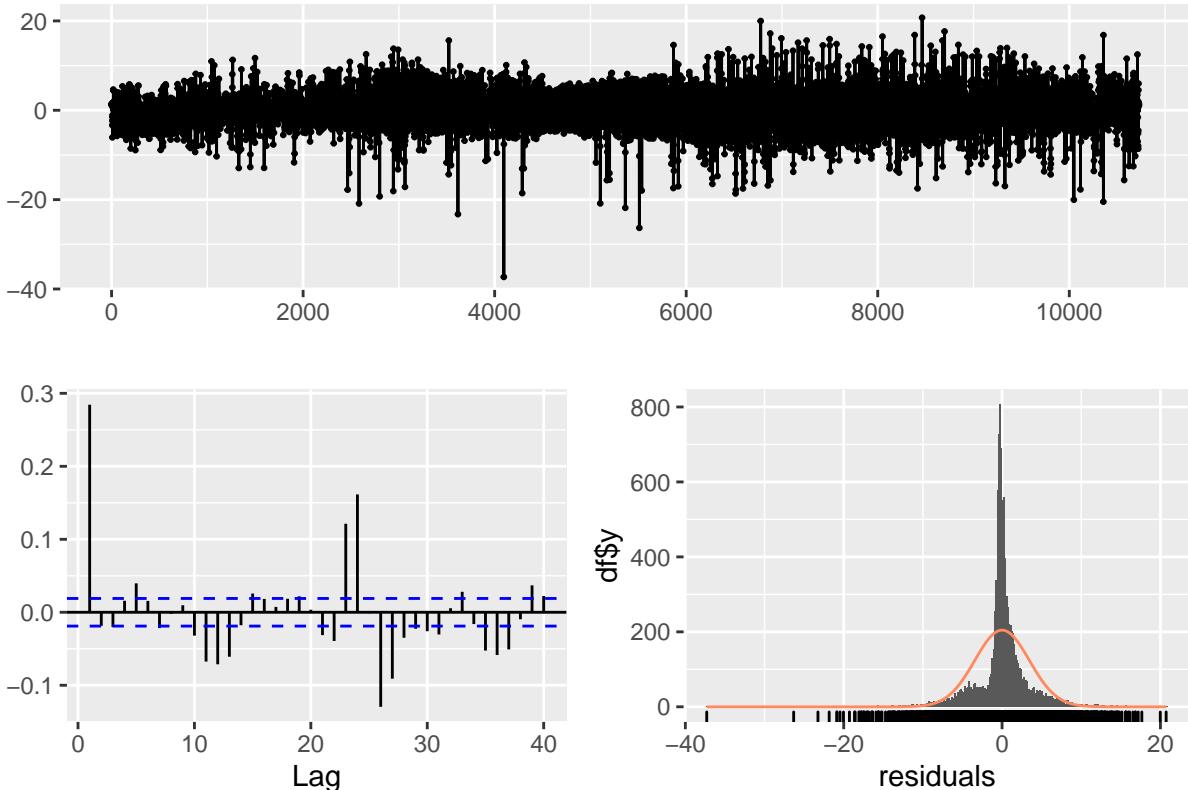
```

## Hour_factor6      4.902e-01  2.101e-01   2.334  0.019634 *
## Hour_factor7      4.901e+00  2.148e-01   22.813 < 2e-16 ***
## Hour_factor8      7.931e+00  2.391e-01   33.172 < 2e-16 ***
## Hour_factor9      4.461e+00  2.604e-01   17.130 < 2e-16 ***
## Hour_factor10     1.996e+00  2.931e-01   6.810  1.03e-11 ***
## Hour_factor11     1.234e+00  3.050e-01   4.047  5.22e-05 ***
## Hour_factor12     1.263e+00  3.144e-01   4.017  5.94e-05 ***
## Hour_factor13     1.327e+00  3.204e-01   4.142  3.47e-05 ***
## Hour_factor14     8.576e-01  3.219e-01   2.665  0.007722 **
## Hour_factor15     4.251e-01  3.189e-01   1.333  0.182454
## Hour_factor16     -1.401e+00 2.910e-01  -4.814  1.50e-06 ***
## Hour_factor17     -4.715e+00 2.695e-01  -17.495 < 2e-16 ***
## Hour_factor18     -3.892e+00 2.535e-01  -15.352 < 2e-16 ***
## Hour_factor19     -9.123e-01 2.446e-01  -3.729  0.000193 ***
## Hour_factor20     4.320e-01  2.390e-01   1.808  0.070698 .
## Hour_factor21     4.170e-01  2.362e-01   1.766  0.077483 .
## Hour_factor22     6.336e-02  2.262e-01   0.280  0.779426
## Hour_factor23      NA        NA        NA        NA
## max_in_month     -6.034e-03 2.788e-02  -0.216  0.828674
## max_in_week      4.564e-02  2.541e-02   1.796  0.072479 .
## night1          NA        NA        NA        NA
## Lag1            6.492e-01  6.993e-03  92.833 < 2e-16 ***
## Lag_week         1.375e-01  5.998e-03  22.920 < 2e-16 ***
## Lag_day          1.950e-01  6.271e-03  31.096 < 2e-16 ***
## Trend           1.633e-03  3.510e-03   0.465  0.641693
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.45 on 10661 degrees of freedom
##   (168 observations deleted due to missingness)
## Multiple R-squared:  0.9423, Adjusted R-squared:  0.942
## F-statistic:  2639 on 66 and 10661 DF,  p-value: < 2.2e-16

```

```
checkresiduals(model10)
```

Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 72  
##  
## data: Residuals  
## LM test = 2521, df = 72, p-value < 2.2e-16
```

```
AIC(model10)
```

```
## [1] 57086.41
```

```
BIC(model10)
```

```
## [1] 57581.5
```

All the exogenous variables are added to the linear regression model, however some of the insignificant ones are removed to get a more compact model.

```
model11 <- auto.arima(data[, "production"], seasonal =TRUE, trace=T)
```

```
##  
## Fitting models using approximations to speed things up...  
##  
## ARIMA(2,1,2) with drift : 65640.88
```

```

##  ARIMA(0,1,0) with drift      : 68977.85
##  ARIMA(1,1,0) with drift      : 65949.94
##  ARIMA(0,1,1) with drift      : 65852.52
##  ARIMA(0,1,0)                  : 68975.84
##  ARIMA(1,1,2) with drift      : 65638.35
##  ARIMA(0,1,2) with drift      : 65639.46
##  ARIMA(1,1,1) with drift      : 65641.52
##  ARIMA(1,1,3) with drift      : 65640.25
##  ARIMA(0,1,3) with drift      : 65636.85
##  ARIMA(0,1,4) with drift      : 65638.17
##  ARIMA(1,1,4) with drift      : Inf
##  ARIMA(0,1,3)                  : 65634.85
##  ARIMA(0,1,2)                  : 65637.46
##  ARIMA(1,1,3)                  : 65637.77
##  ARIMA(0,1,4)                  : 65636.16
##  ARIMA(1,1,2)                  : 65636.35
##  ARIMA(1,1,4)                  : Inf
##
## Now re-fitting the best model(s) without approximations...
##
##  ARIMA(0,1,3)                  : 65638.68
##
##  Best model: ARIMA(0,1,3)

```

```
summary(model11)
```

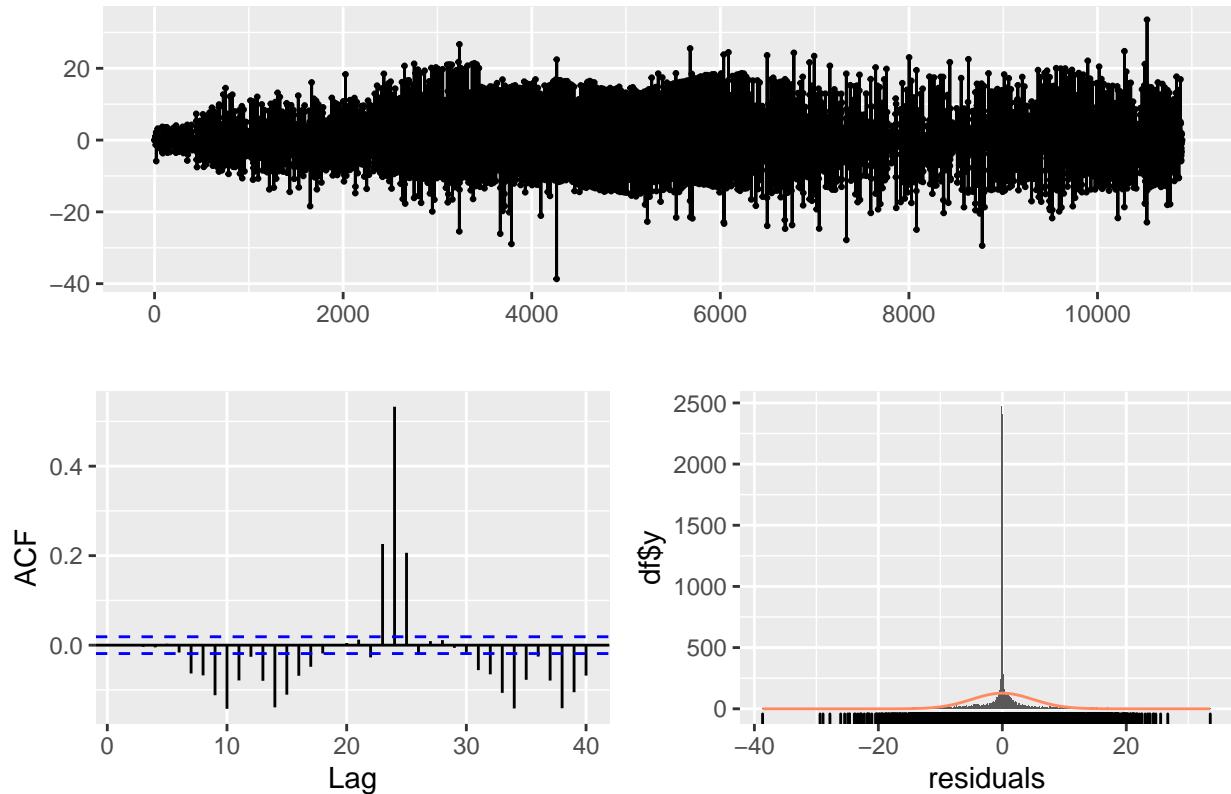
```

## Series: data[, "production"]
## ARIMA(0,1,3)
##
## Coefficients:
##          ma1     ma2     ma3
##        0.5803  0.1522  0.0203
##  s.e.  0.0096  0.0111  0.0094
##
## sigma^2 = 24.2: log likelihood = -32815.34
## AIC=65638.68  AICc=65638.68  BIC=65667.86
##
## Training set error measures:
##           ME      RMSE      MAE MPE MAPE      MASE      ACF1
## Training set -7.927164e-08 4.918302 2.431535 NaN  Inf 0.8797759 -0.0001083999

```

```
checkresiduals(model11)
```

Residuals from ARIMA(0,1,3)



```
##  
## Ljung-Box test  
##  
## data: Residuals from ARIMA(0,1,3)  
## Q* = 454.1, df = 7, p-value < 2.2e-16  
##  
## Model df: 3. Total lags used: 10
```

```
AIC(model11)
```

```
## [1] 65638.68
```

```
BIC(model11)
```

```
## [1] 65667.86
```

```
model12 <- auto.arima(data[, "production"], seasonal = FALSE, trace=T)
```

```
##  
## Fitting models using approximations to speed things up...  
##  
## ARIMA(2,1,2) with drift : 65640.88  
## ARIMA(0,1,0) with drift : 68977.85
```

```

##  ARIMA(1,1,0) with drift      : 65949.94
##  ARIMA(0,1,1) with drift      : 65852.52
##  ARIMA(0,1,0)                  : 68975.84
##  ARIMA(1,1,2) with drift      : 65638.35
##  ARIMA(0,1,2) with drift      : 65639.46
##  ARIMA(1,1,1) with drift      : 65641.52
##  ARIMA(1,1,3) with drift      : 65640.25
##  ARIMA(0,1,3) with drift      : 65636.85
##  ARIMA(0,1,4) with drift      : 65638.17
##  ARIMA(1,1,4) with drift      : Inf
##  ARIMA(0,1,3)                  : 65634.85
##  ARIMA(0,1,2)                  : 65637.46
##  ARIMA(1,1,3)                  : 65637.77
##  ARIMA(0,1,4)                  : 65636.16
##  ARIMA(1,1,2)                  : 65636.35
##  ARIMA(1,1,4)                  : Inf
##
## Now re-fitting the best model(s) without approximations...
##
##  ARIMA(0,1,3)                  : 65638.68
##
## Best model: ARIMA(0,1,3)

```

```
summary(model12)
```

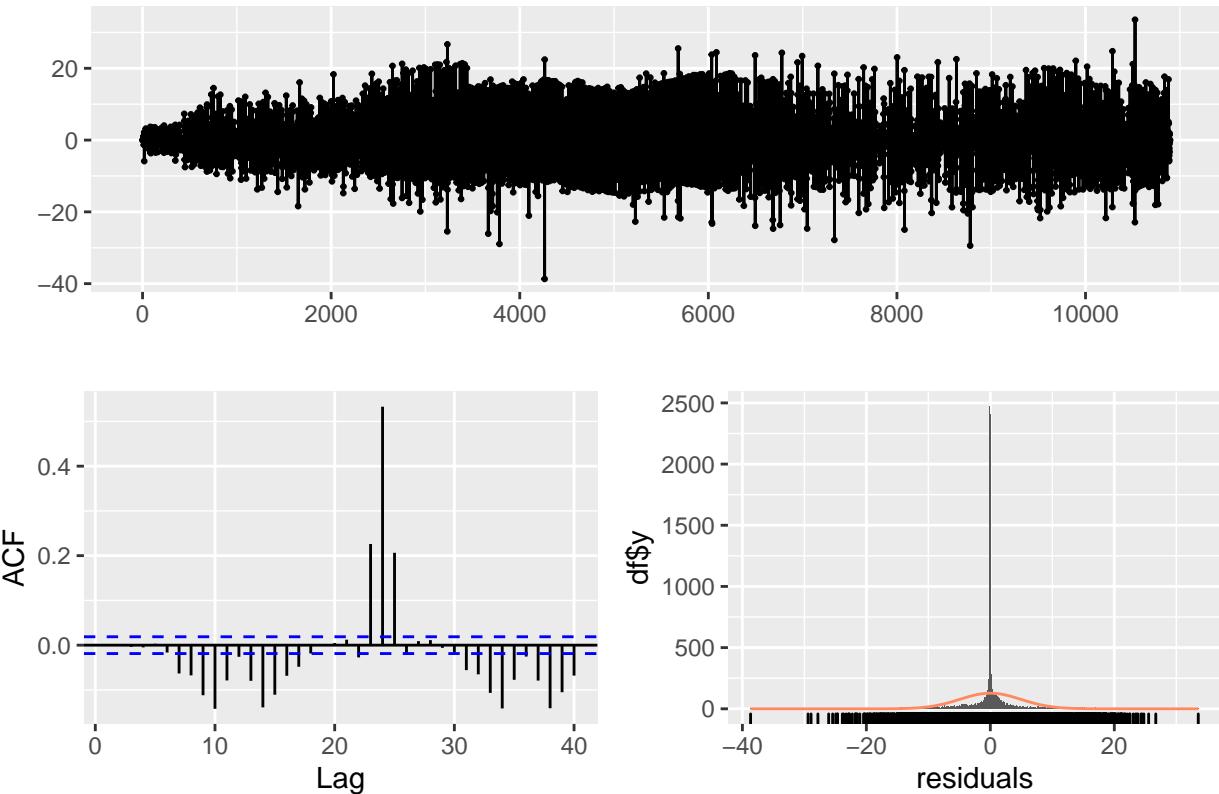
```

## Series: data[, "production"]
## ARIMA(0,1,3)
##
## Coefficients:
##             ma1      ma2      ma3
##             0.5803  0.1522  0.0203
## s.e.  0.0096  0.0111  0.0094
##
## sigma^2 = 24.2: log likelihood = -32815.34
## AIC=65638.68   AICc=65638.68   BIC=65667.86
##
## Training set error measures:
##                   ME      RMSE      MAE MPE MAPE      MASE          ACF1
## Training set -7.927164e-08 4.918302 2.431535 NaN  Inf 0.8797759 -0.0001083999

```

```
checkresiduals(model12)
```

Residuals from ARIMA(0,1,3)



```
##  
## Ljung-Box test  
##  
## data: Residuals from ARIMA(0,1,3)  
## Q* = 454.1, df = 7, p-value < 2.2e-16  
##  
## Model df: 3. Total lags used: 10
```

```
AIC(model12)
```

```
## [1] 65638.68
```

```
BIC(model12)
```

```
## [1] 65667.86
```

```
AvgDSWRF <- as.numeric(data[,AverageDSWRF])  
model13 <- auto.arima(data[, "production"], xreg= AvgDSWRF, seasonal =FALSE, trace=T)
```

```
##  
## Fitting models using approximations to speed things up...  
##  
## Regression with ARIMA(2,1,2) errors : Inf
```

```

## Regression with ARIMA(0,1,0) errors : 68398.07
## Regression with ARIMA(1,1,0) errors : 65934.24
## Regression with ARIMA(0,1,1) errors : 65679.35
## ARIMA(0,1,0) : 68396.07
## Regression with ARIMA(1,1,1) errors : 65564.14
## Regression with ARIMA(2,1,1) errors : 65544.59
## Regression with ARIMA(2,1,0) errors : 65550.69
## Regression with ARIMA(3,1,1) errors : 65547.36
## Regression with ARIMA(1,1,2) errors : 65550.36
## Regression with ARIMA(3,1,0) errors : 65546.36
## Regression with ARIMA(3,1,2) errors : Inf
## ARIMA(2,1,1) : 65542.59
## ARIMA(1,1,1) : 65562.11
## ARIMA(2,1,0) : 65548.63
## ARIMA(3,1,1) : 65545.36
## ARIMA(2,1,2) : Inf
## ARIMA(1,1,0) : 65932.23
## ARIMA(1,1,2) : 65548.35
## ARIMA(3,1,0) : 65544.31
## ARIMA(3,1,2) : Inf
##
## Now re-fitting the best model(s) without approximations...
##
## ARIMA(2,1,1) : 65544.35
##
## Best model: Regression with ARIMA(2,1,1) errors

```

```
summary(model13)
```

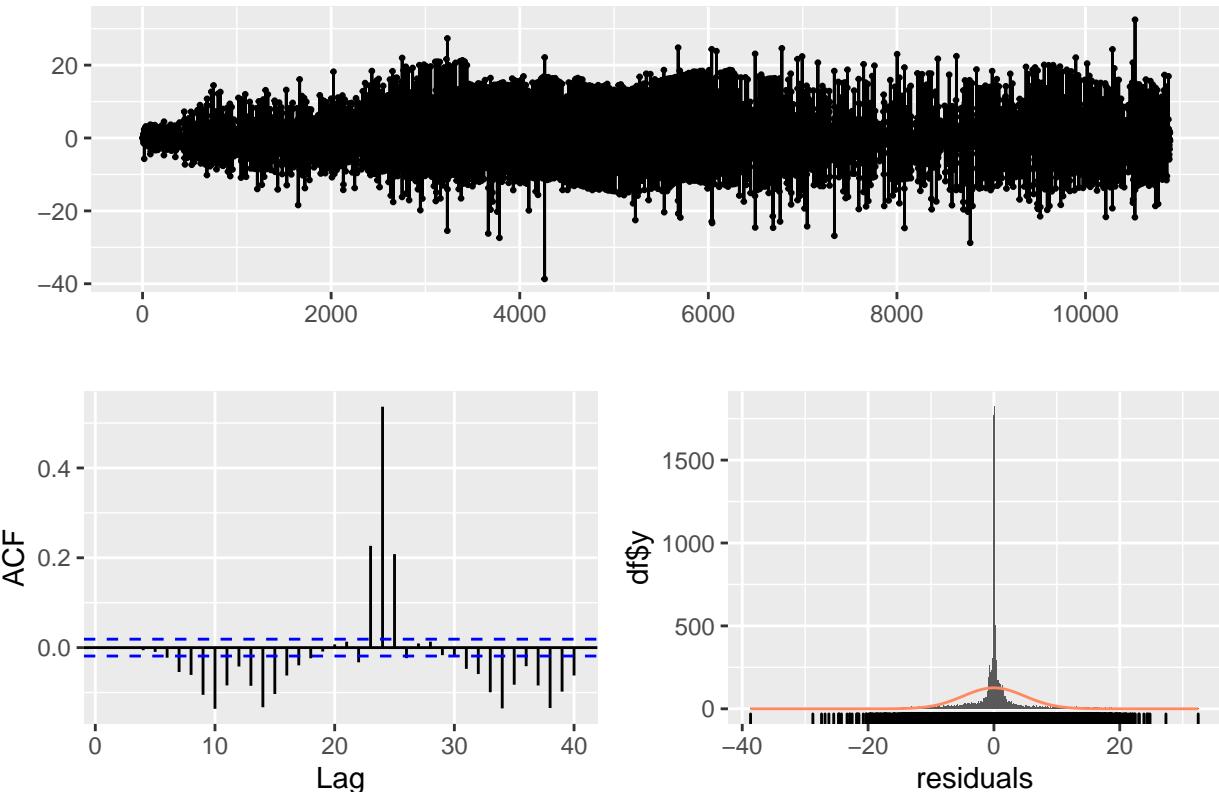
```

## Series: data[, "production"]
## Regression with ARIMA(2,1,1) errors
##
## Coefficients:
##          ar1      ar2      ma1     xreg
##        0.4152 -0.1289  0.1448  0.0049
##  s.e.  0.0489  0.0258  0.0494  0.0005
##
## sigma^2 = 23.99: log likelihood = -32767.17
## AIC=65544.35  AICc=65544.35  BIC=65580.83
##
## Training set error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
## Training set -2.406619e-05 4.89661 2.500338 NaN  Inf  0.9046698 -6.247474e-05

```

```
checkresiduals(model13)
```

Residuals from Regression with ARIMA(2,1,1) errors



```
##  
## Ljung-Box test  
##  
## data: Residuals from Regression with ARIMA(2,1,1) errors  
## Q* = 402.21, df = 6, p-value < 2.2e-16  
##  
## Model df: 4. Total lags used: 10
```

```
AIC(model13)
```

```
## [1] 65544.35
```

```
BIC(model13)
```

```
## [1] 65580.83
```

```
AvgTEMP <- as.numeric(data[,AverageTEMP])  
model14 <- auto.arima(data[,"production"], xreg= AvgTEMP, seasonal =FALSE, trace=T)
```

```
##  
## Fitting models using approximations to speed things up...  
##  
## Regression with ARIMA(2,1,2) errors : 64134.79
```

```

## Regression with ARIMA(0,1,0) errors : 66078.88
## Regression with ARIMA(1,1,0) errors : 64565.76
## Regression with ARIMA(0,1,1) errors : 64202.59
## ARIMA(0,1,0) : 66076.88
## Regression with ARIMA(1,1,2) errors : 64184.01
## Regression with ARIMA(2,1,1) errors : 64132.86
## Regression with ARIMA(1,1,1) errors : 64200.95
## Regression with ARIMA(2,1,0) errors : 64131.04
## Regression with ARIMA(3,1,0) errors : 64133.85
## Regression with ARIMA(3,1,1) errors : 64135.83
## ARIMA(2,1,0) : 64129.06
## ARIMA(1,1,0) : 64563.76
## ARIMA(3,1,0) : 64131.85
## ARIMA(2,1,1) : 64130.88
## ARIMA(1,1,1) : 64198.94
## ARIMA(3,1,1) : 64133.83
##
## Now re-fitting the best model(s) without approximations...
##
## ARIMA(2,1,0) : 64130.51
##
## Best model: Regression with ARIMA(2,1,0) errors

```

```
summary(model14)
```

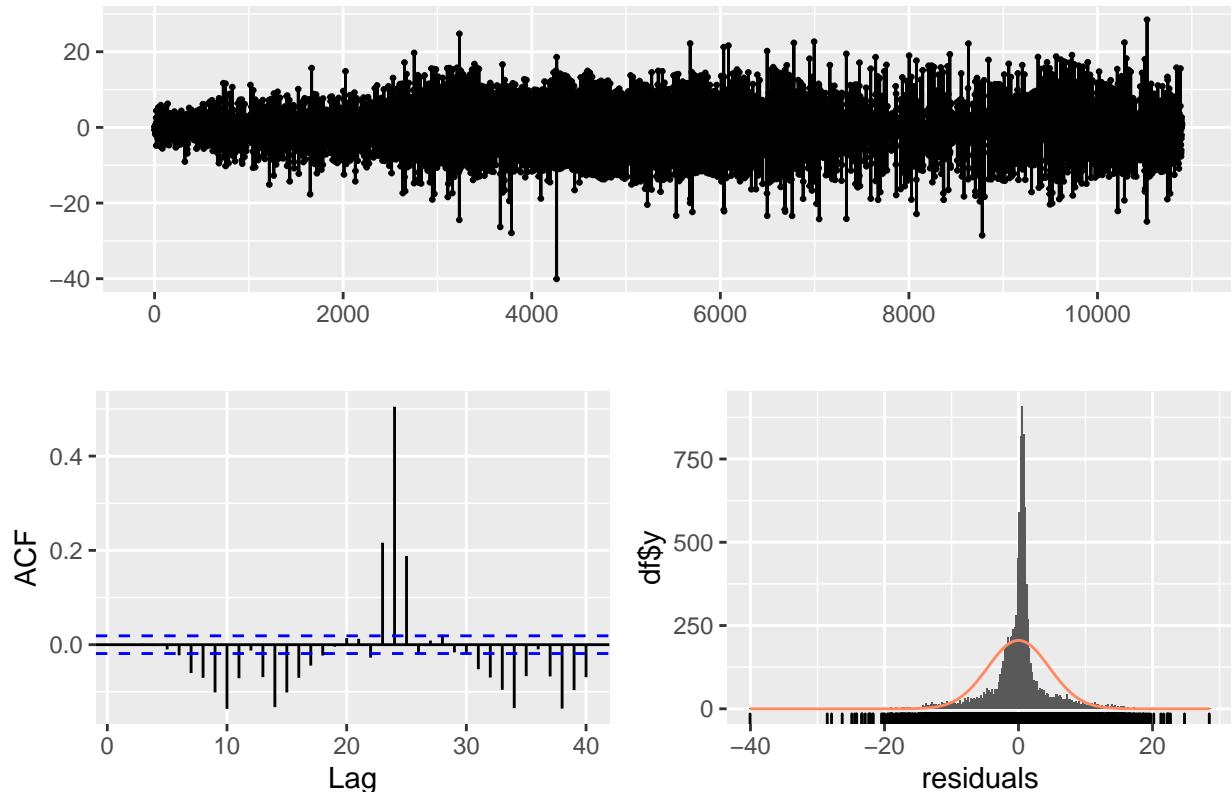
```

## Series: data[, "production"]
## Regression with ARIMA(2,1,0) errors
##
## Coefficients:
##             ar1      ar2      xreg
##             0.4367 -0.1985  2.4070
## s.e.  0.0096  0.0094  0.0544
##
## sigma^2 = 21.07: log likelihood = -32061.25
## AIC=64130.5   AICc=64130.51   BIC=64159.69
##
## Training set error measures:
##                   ME      RMSE      MAE MPE MAPE      MASE       ACF1
## Training set -0.0009995426 4.589428 2.753393 NaN  Inf 0.99623 0.0008446895

```

```
checkresiduals(model14)
```

Residuals from Regression with ARIMA(2,1,0) errors



```
##  
## Ljung-Box test  
##  
## data: Residuals from Regression with ARIMA(2,1,0) errors  
## Q* = 413.5, df = 7, p-value < 2.2e-16  
##  
## Model df: 3. Total lags used: 10
```

```
AIC(model14)
```

```
## [1] 64130.5
```

```
BIC(model14)
```

```
## [1] 64159.69
```

```
model15 <- arima(data[, "production"], c(2, 0, 0))  
summary(model15)
```

```
##  
## Call:  
## arima(x = data[, "production"], order = c(2, 0, 0))  
##
```

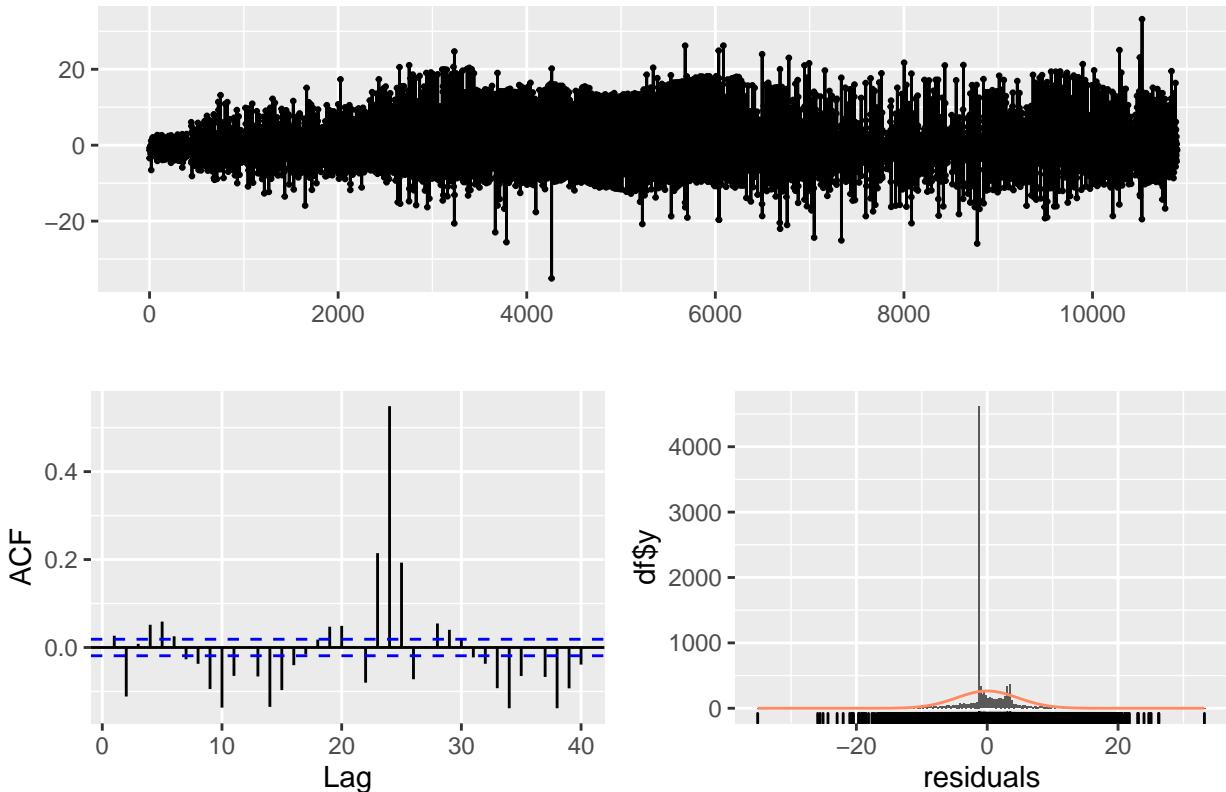
```

## Coefficients:
##      ar1      ar2  intercept
##      1.4298 -0.5557   10.4434
##  s.e.  0.0080  0.0080    0.3552
##
## sigma^2 estimated as 21.81:  log likelihood = -32255.29,  aic = 64518.57
##
## Training set error measures:
##          ME      RMSE      MAE MPE MAPE      MASE      ACF1
## Training set -0.0001431244 4.670327 3.01387 NaN  Inf 1.090475 0.02674318

```

```
checkresiduals(model15)
```

Residuals from ARIMA(2,0,0) with non-zero mean



```

##
## Ljung-Box test
##
## data: Residuals from ARIMA(2,0,0) with non-zero mean
## Q* = 542.14, df = 7, p-value < 2.2e-16
##
## Model df: 3. Total lags used: 10

```

```
AIC(model15)
```

```
## [1] 64518.57
```

```
BIC(model15)
```

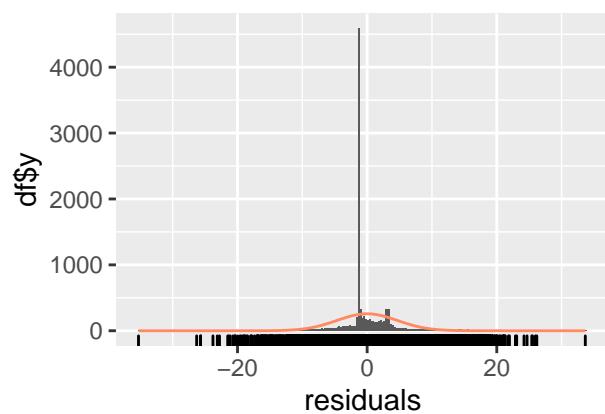
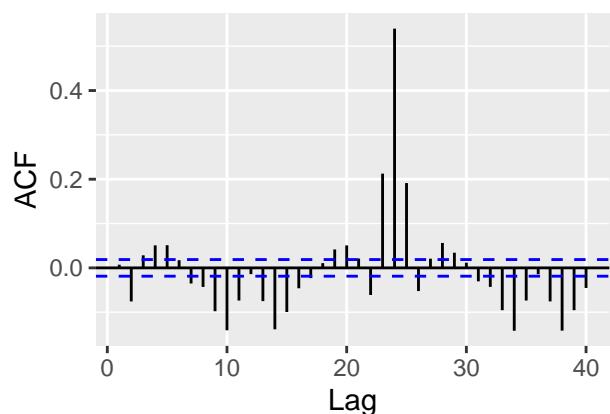
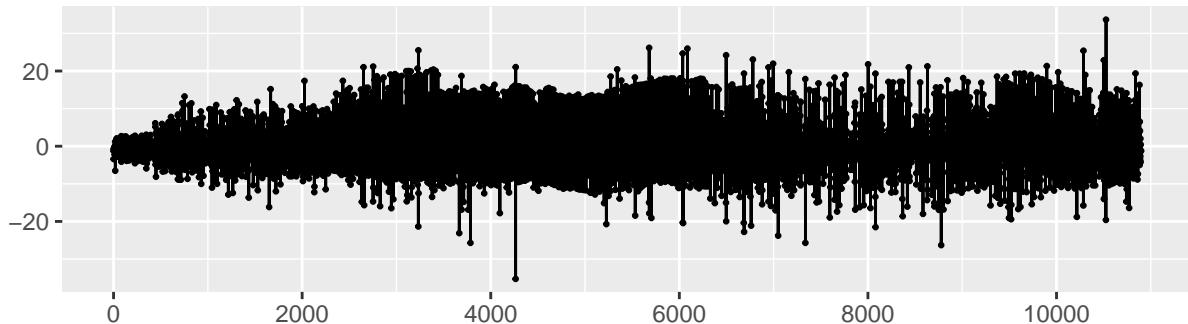
```
## [1] 64547.76
```

```
model16 <- arima(data[, "production"], c(3, 0, 0))
summary(model16)
```

```
##
## Call:
## arima(x = data[, "production"], order = c(3, 0, 0))
##
## Coefficients:
##          ar1      ar2      ar3  intercept
##        1.4565 -0.6243  0.0480     10.4437
##  s.e.   0.0096  0.0158  0.0096     0.3727
##
## sigma^2 estimated as 21.76:  log likelihood = -32242.72,  aic = 64495.44
##
## Training set error measures:
##           ME      RMSE      MAE MPE MAPE      MASE      ACF1
## Training set -0.0001934271 4.664941 2.983898 NaN  Inf 1.079631 0.007148439
```

```
checkresiduals(model16)
```

Residuals from ARIMA(3,0,0) with non-zero mean



```

##  

## Ljung-Box test  

##  

## data: Residuals from ARIMA(3,0,0) with non-zero mean  

## Q* = 485.75, df = 6, p-value < 2.2e-16  

##  

## Model df: 4. Total lags used: 10

AIC(model16)

## [1] 64495.44

BIC(model16)

## [1] 64531.92

model17 <- arima(data[, "production"], c(4, 0, 1))
summary(model17)

##  

## Call:  

## arima(x = data[, "production"], order = c(4, 0, 1))  

##  

## Coefficients:  

##          ar1      ar2      ar3      ar4      ma1  intercept  

##        2.1593 -1.7501  0.7496 -0.2175 -0.7465    10.4458  

##  s.e.  0.0130  0.0245  0.0216  0.0096  0.0099     0.1859  

##  

## sigma^2 estimated as 20.23: log likelihood = -31845.09, aic = 63704.19  

##  

## Training set error measures:  

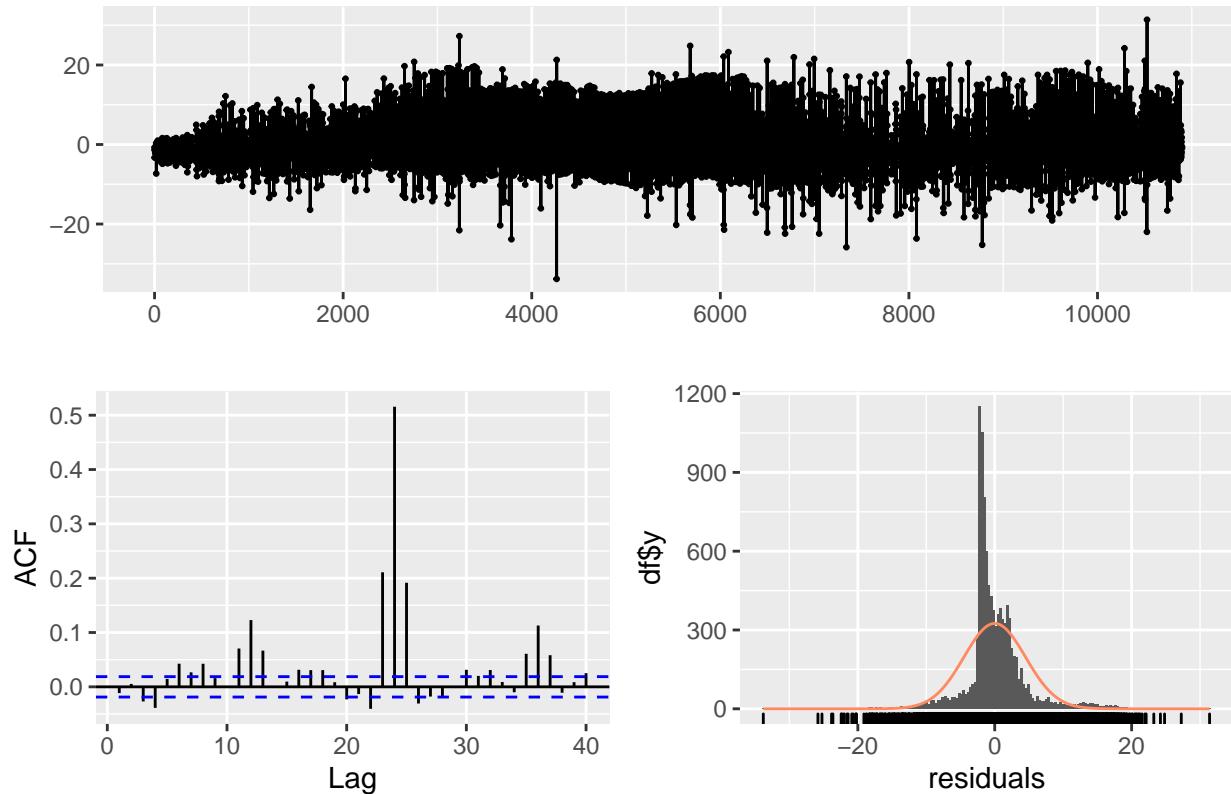
##           ME      RMSE      MAE      MPE      MAPE      MASE      ACF1  

## Training set -0.000289889 4.497696 2.962509 NaN  Inf  1.071892 -0.01150789

checkresiduals(model17)

```

Residuals from ARIMA(4,0,1) with non-zero mean



```
##  
## Ljung-Box test  
##  
## data: Residuals from ARIMA(4,0,1) with non-zero mean  
## Q* = 78.318, df = 4, p-value = 4.441e-16  
##  
## Model df: 6. Total lags used: 10
```

```
AIC(model17)
```

```
## [1] 63704.19
```

```
BIC(model17)
```

```
## [1] 63755.26
```

Both auto.arima is used to develop different models and self-developed models are formed from the information given by partial auto correlation and autocorrelation plots. Arima models explain a large portion of the data, however the best linear regression model seems to be better than the arima models considering their AIC and BIC measures.

Model Selection & Chosen Approach:

```
final_model=model10
```

Among different models developed, the Akaike and Bayesian information criteria give the smallest result with model 10, which is a linear regression model with autoregressive components, moving average components, dummy variables for yearly, monthly and hourly seasonality, trend component and variables given except the ones that are really insignificant. So, model10 is used as the final model.

Prediction & Results:

Data is manipulated before making predictions in order to use the real weather forecasts that is not developed with the model created but given.

```
long_weather1 =
  long_weather %>%
  arrange(long_weather) %>%
  mutate(value= shift(value,-2592))

long_weather1=long_weather1[-c((.N-6047):.N),]
wide_weather1= dcast(long_weather1, date+hour~lat+lon+variable)
data1 <- data.table(merge(wide_weather1,production))
data1[,AverageTEMP:=rowMeans(data1[,c("36.75_33.5_TEMP","36.75_33.25_TEMP","36.75_33_TEMP","36.5_33.5_TEMP")])
data1[,AverageREL_HUMIDITY:=rowMeans(data1[,c("36.75_33.5_REL_HUMIDITY","36.75_33.25_REL_HUMIDITY","36.75_33_REL_HUMIDITY","36.5_33_REL_HUMIDITY")])
data1[,AverageDSWRF:=rowMeans(data1[,c("36.75_33.5_DSWRF","36.75_33.25_DSWRF","36.75_33_DSWRF","36.5_33_DSWRF")])
data1[,AverageCLOUD_LOW_LAYER:=rowMeans(data1[,c("36.75_33.5_CLOUD_LOW_LAYER","36.75_33.25_CLOUD_LOW_LAYER","36.75_33_CLOUD_LOW_LAYER","36.5_33_CLOUD_LOW_LAYER")])
data1 <- data1[order(hour,decreasing = F)]
data1 <- data1[order(date,decreasing = F)]
data1[, Year:=as.factor(year(date))]
data1[,Month := as.factor(month(date))]
data1[,Hour_factor := as.factor(hour)]
data1[,max_in_month:=runmax(x=data$production, k=720, align = "left")]
data1[,max_in_week:=runmax(x=data$production, k=168, align ="left")]
data1[hour<=5|hour>=21,night:=1]
data1[hour<21&hour>5,night:=0]
data1$night <- as.factor(data1$night)
data1[,Lag1:=c(NA, data1$production[1:(.N-1))]]
data1[,Lag_week:=c(rep(NA,168), data1$production[1:(.N-24*7))]]
data1[,Lag_day:=c(rep(NA,24), data1$production[1:(.N-24))]]
data1[, Trend:=(1:.N)]
colnames(data1) <- c("date", "hour", "CLOUD_LOW_LAYER_36.25_33", "DSWRF_36.25_33", "REL_HUMIDITY_36.25_33", "Year", "Month", "Hour_factor", "max_in_month", "max_in_week", "night", "Lag1", "Lag_week", "Lag_day", "Trend")

tmp= data1[(.N-71):( .N)]
```

```
predictions = rep(0,72)
for(i in 1:72) {
  predictions[i] = predict(final_model,newdata = tmp[i,])
  tmp[i+1,"Lag1"] = predictions[i]
  if(predictions[i]<0){predictions[i]=0}
}
predictions[49:72]
```

```
## [1] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
## [7] 1.2256838 9.9448617 21.7291328 28.8898355 29.9850460 30.6944920
## [13] 30.6114639 30.6471145 27.9944763 24.3377290 18.0918973 8.5723744
## [19] 0.7482614 0.0000000 0.0000000 0.0000000 0.0000000
```

Conclusion:

For understanding the behavior of the data, autoregressive, moving average and arima methods are used. Trend and seasonality are added for analyzing time dependency and autocorrelation of the data. After that process, temperature, relative humidity, downward shortwave radiation flux and cloud cover data are added and subtracted to see the correlation and relations between them and the production rate. Also, for the arima part of model trials, different p (for autoregression part), q (for moving average part) and d (for the number of differencing) values are tried in addition to the auto.arima trials.

Model10 is a linear regression model with trend and seasonality components that also contains autoregressive components with lags 1,24,168 to account for hourly, dayly and weekly seasonality and trend. This model contains dummy variables for hour, day, month and year and moving average component that gives maximum production level within the week and within the month. Model10 is used as the final model because the AIC and BIC performance measures as well as the adjusted R_squared value gives the best result among the models developed.

The residuals of the final model is not perfect white noise as it should have been, however there is no visible seasonality left in the autocorrelation function and even though there is autocorrelation among some variables, they are small. The residuals seem to be normally distributed and around zero mean with constant variation. There is no visible trend and/or seasonality left in the residuals, which show that the predictions that this model come up with can be used. The predictions are made for two days ahead using the variables related to weather forecast information that belong to the day that is being predicted and using model10 created with the data from two days before.