DATA 624: Project 1 - Part B

Sang Yoon (Andy) Hwang & Vinicio Haro

October 22, 2019

Contents

Part B: Forecasting Power

Instructions: Part B consists of a simple dataset of residential power usage for January 1998 until December 2013. Your assignment is to model these data and a monthly forecast for 2014. The data is given in a single file. The variable 'KWH' is power consumption in Kilowatt hours, the rest is straight forward. Add these to your existing files above - clearly labeled.

Exploration

From our time series data (frequency = 12, monthly power_data) we observed there is a missing value in September 2008. We used imputation method called na.interpolation which performs a technique in numerical analysis which estimates a value from known data points. For our case, linear method using first order Taylor polynomial is used.

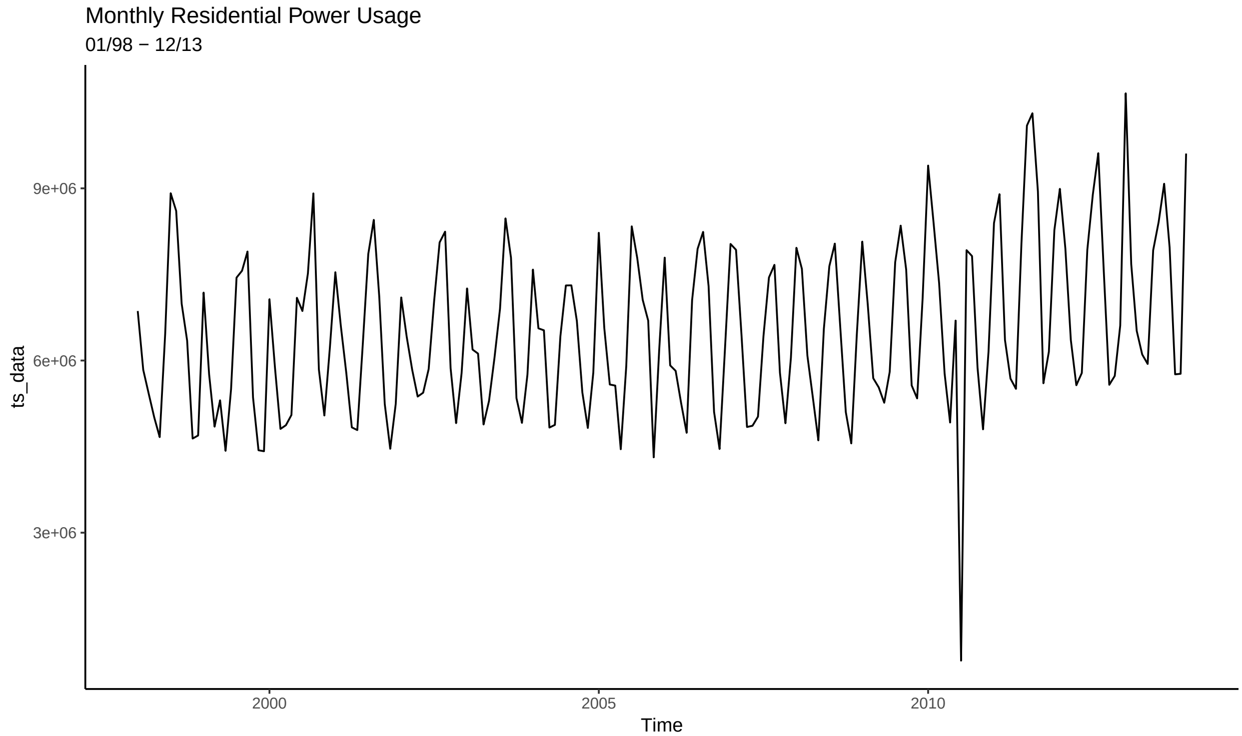
Our initial time series plot reveal annual seasonality within this time series. The box plot/seasonality plot actually reveals where power consumption fluctuations occur within each of the cycke positions. We can speculate that this could be due to there being no major Holidays that require power draining decor plus we assume minimal AC usage during the cold months.

We see power consumption increase between the months of June and August. This must be tied to AC usage during the warmer months of a year and finally power usage dips from September to Novemeber with a small spike in December. We speculate that thisis due to transitioning out of summer. The spike in December could be connected to the usage or Holiday lights being kept on.

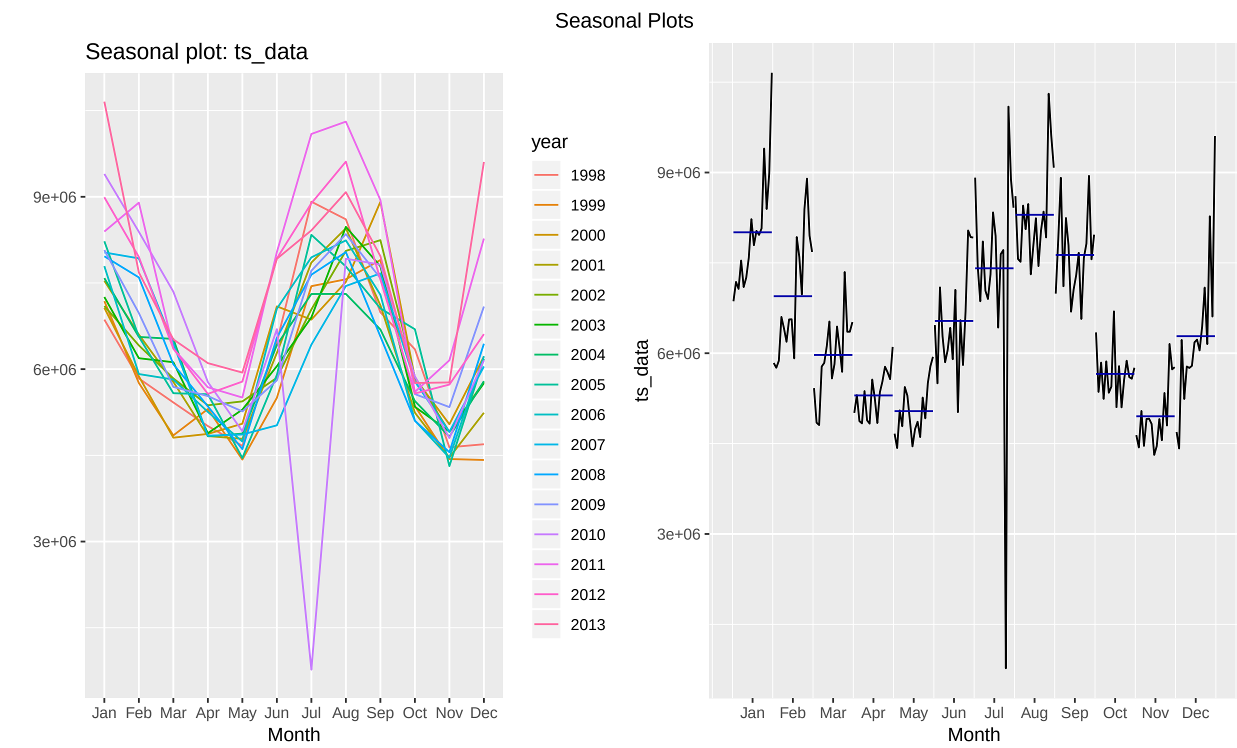
Within the overall TS plot, we see a dip in July 2010. This could be due to a power outtage during a hot summer month. This can certainly be considered to be an outlier within this TS. Using TSOutliers, we can actually identify the index where our outliers may be. TSoutliers also replaces the outlier using Box-Cox. If set lambda=auto, then TSoutliers will automatically perform Box-Cox transformation.

The ACF plot shows that autocorrelations are well outside the significant space indicating the series is not white noise, non-stationary.

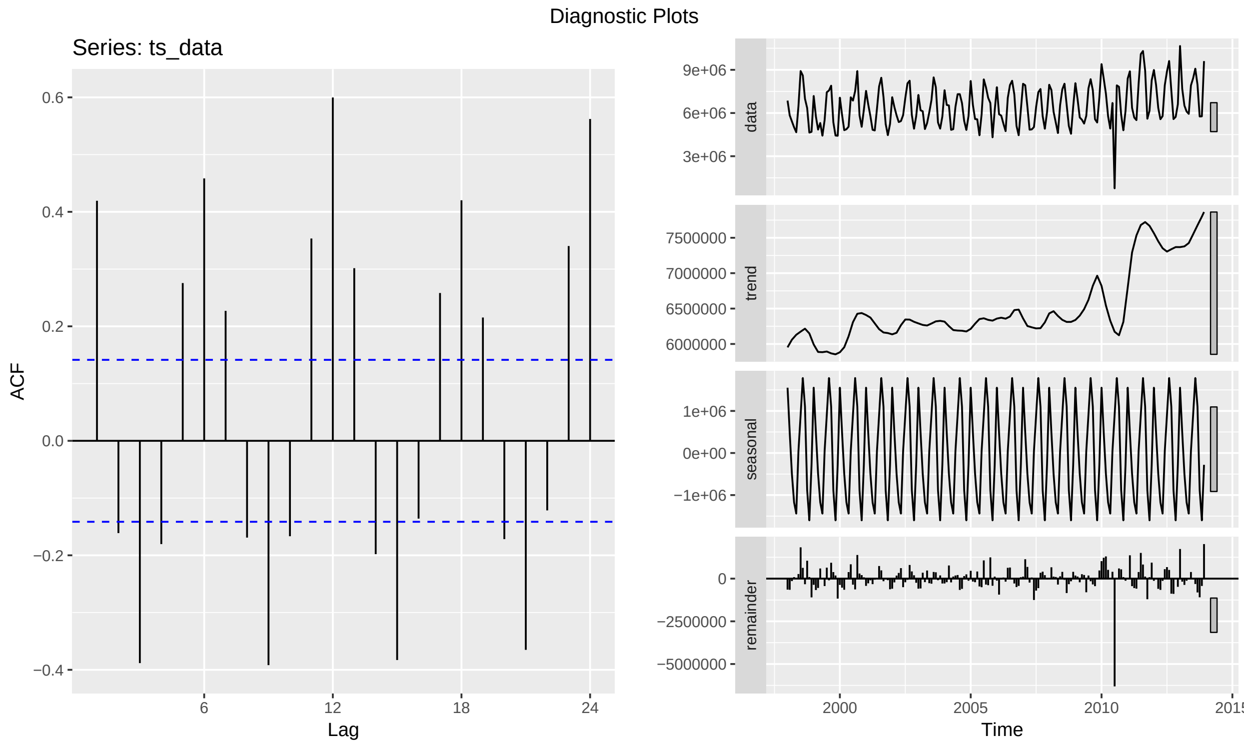
0.0.1 Series plot



0.0.2 Seasonal plots



0.0.3 Diagnostic plots



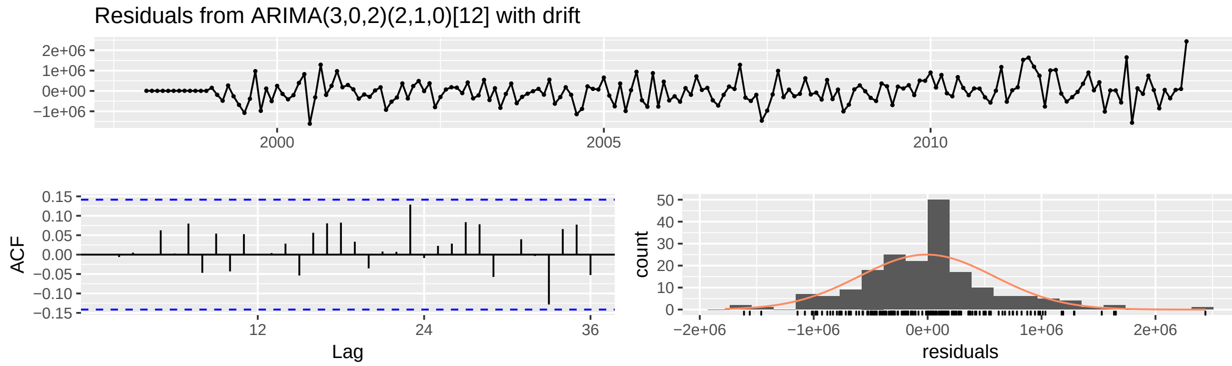
Data Model

Out of the models we built, we can make some preliminary observations. The residuals for each of our models does not have a major deviance from normality, however residuals of Model #1: ARIMA do not have an extended number of bins distorting the normality proximity but we can say it is still fairly normally distributed.

The residual ACF plots show residual autocorrelations for each of our models. Model #1: ARIMA has less autocorrelation than the other three models. Model 1 is well within the 95% limits indicated by the dotted blue lines.

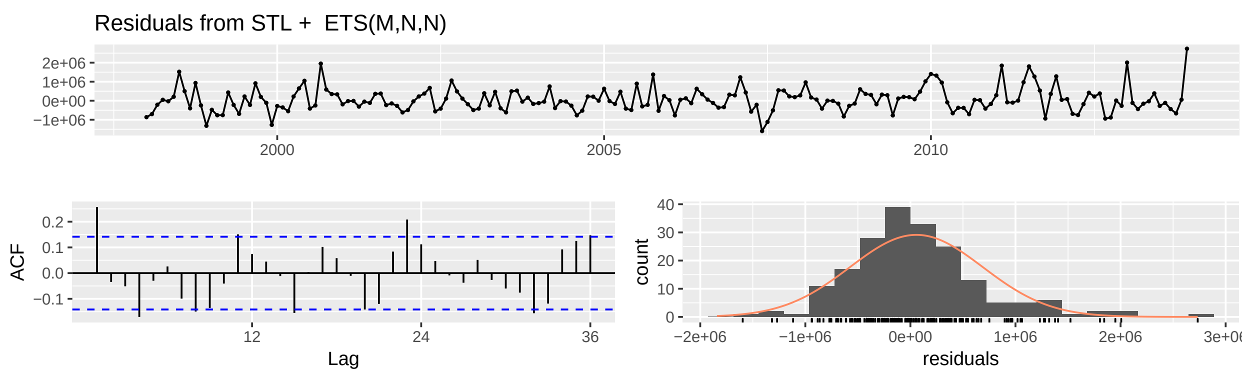
If we examine the Ljung-Box test results for our models, the only model with a p-value > 0.05 is Model #1: ARIMA. This implies that the residuals from other models are not independent, hence not white noise. The full model summary can be viewed in the appendix.

0.0.4 Model #1: ARIMA



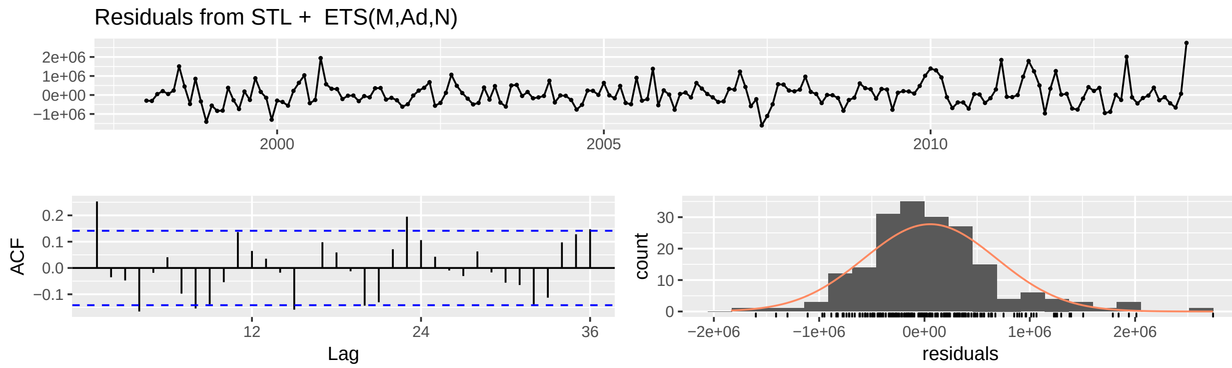

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from ARIMA(3,0,2)(2,1,0)[12] with drift
FALSE Q* = 12.555, df = 16, p-value = 0.705
FALSE
FALSE Model df: 8.   Total lags used: 24
```

0.0.5 Model #2: STL (no-demped) - MNN



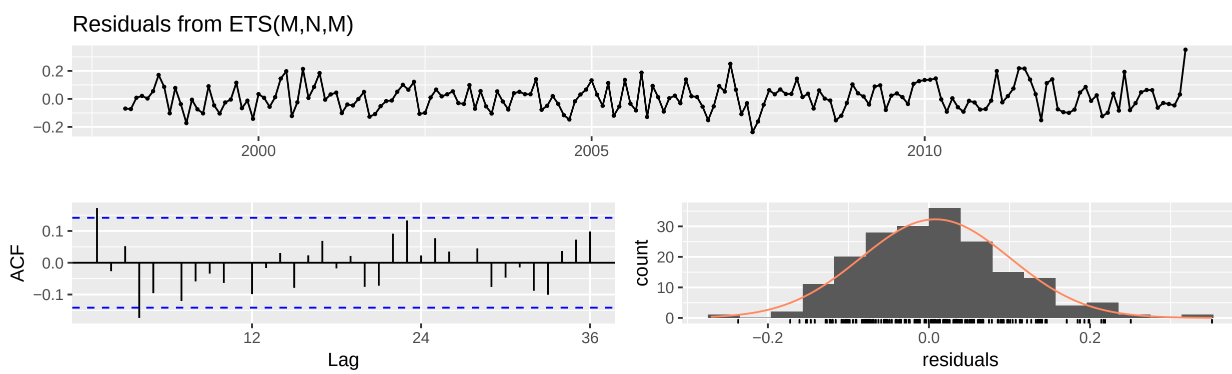

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from STL +  ETS(M,N,N)
FALSE Q* = 65.934, df = 22, p-value = 2.84e-06
FALSE
FALSE Model df: 2.   Total lags used: 24
```

0.0.6 Model #2-2: STL (demped) - MAdN




```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from STL +  ETS(M,Ad,N)
FALSE Q* = 63.375, df = 19, p-value = 1.119e-06
FALSE
FALSE Model df: 5.   Total lags used: 24
```

0.0.7 Model #3: ets - MNM



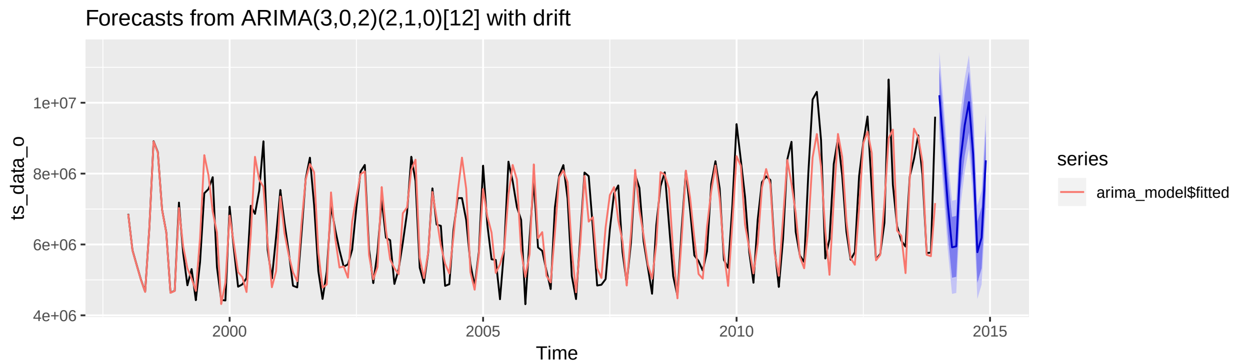

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from ETS(M,N,M)
FALSE Q* = 32.042, df = 10, p-value = 0.000394
FALSE
FALSE Model df: 14.   Total lags used: 24
```

Forecast

`auto.arima()` performs cross validation on hyperparameter tuning to find the best model with parameters of `order` and `seasonal` that minimize `AIC`. This gave us **arima_model**: ARIMA$(3,0,2)(2,1,0)12$ with drift resulting `AIC` = 5332.24.

Since ARIMA is the only reliable model, as other models failed Ljung test, we will plot forecasts of ARIMA only. The forecasted values can be viewed in the appendix.

0.0.8 Model #1: ARIMA



Discussion

We implemented a cross validation method of testing for h=12. The process randomly chooses 12 points to measure and take the average of RMSEs. By definition, a lower RMSE on test set is attributed with a better forecast on unseen data.

Using Time series cross-validation, we compute RMSE on testset (h=12). We would have to pick the model with the lowest RMSE on test set as our final model if we had more than 1 model to compare. In our case, since we only have 1 model left after Ljung test, we have no choice but to pick seasonal ARIMA model as our final choice. Cross-validation test shows that RMSE on test is around 720k when RMSE on training is around 589k. We can conclude the model is not necessarily overfitted. Given that MAPE on training is less than 7, it is not a suprising result.

```
FALSE [1] "RMSE - train: 589381.7"
```

```
FALSE [1] "RMSE - test: 725175"
```

Appendix

Part B

Model Summary

ARIMA:

```
FALSE
FALSE Forecast method: ARIMA(3,0,2)(2,1,0)[12] with drift
FALSE
FALSE Model Information:
FALSE Series: ts_data_o
FALSE ARIMA(3,0,2)(2,1,0)[12] with drift
FALSE
FALSE Coefficients:
FALSE           ar1      ar2     ar3     ma1     ma2     sar1     sar2     drift
FALSE       -0.5606  -0.2216  0.3284  0.8902  0.4827  -0.7249  -0.4152  9018.405
FALSE s.e.   0.3992   0.3382  0.0960  0.4120  0.4551   0.0797   0.0841  3027.685
FALSE
FALSE sigma^2 estimated as 3.878e+11:  log likelihood=-2657.12
FALSE AIC=5332.24   AICc=5333.3   BIC=5360.97
FALSE
FALSE Error measures:
FALSE                     ME     RMSE      MAE        MPE     MAPE      MASE
FALSE Training set -8455.077 589381.7 427752.5 -0.7944782 6.475365 0.6904053
FALSE                      ACF1
FALSE Training set 0.0006090194
FALSE
FALSE Forecasts:
FALSE          Point Forecast   Lo 80    Hi 80   Lo 95    Hi 95
FALSE Jan 2014       10210619 9412589 11008649 8990138 11431100
FALSE Feb 2014        8722658 7882412  9562903 7437613 10007702
FALSE Mar 2014        7137962 6295514  7980411 5849548  8426376
FALSE Apr 2014        5919874 5060514  6779234 4605596  7234152
FALSE May 2014        5946730 5087082  6806377 4632012  7261448
FALSE Jun 2014        8383812 7524148  9243475 7069070  9698553
FALSE Jul 2014        9362213 8500206 10224219 8043888 10680538
FALSE Aug 2014       10018953 9155935 10881971 8699080 11338826
FALSE Sep 2014        8547612 7684559  9410664 7227687  9867536
FALSE Oct 2014        5781906 4918467  6645344 4461391  7102421
FALSE Nov 2014        6193673 5329717  7057629 4872367  7514980
FALSE Dec 2014        8373767 7509705  9237829 7052298  9695236
```

STL - MNN:

```
FALSE
FALSE Forecast method: STL +  ETS(M,N,N)
```

```
FALSE
FALSE Model Information:
FALSE ETS(M,N,N)
FALSE
FALSE Call:
FALSE  ets(y = x, model = etsmodel, allow.multiplicative.trend = allow.multiplicative.trend)
FALSE
FALSE   Smoothing parameters:
FALSE     alpha = 0.1159
FALSE
FALSE   Initial states:
FALSE     l = 6317745.8917
FALSE
FALSE   sigma:  0.097
FALSE
FALSE      AIC     AICc      BIC
FALSE 6139.631 6139.758 6149.403
FALSE
FALSE Error measures:
FALSE                    ME     RMSE      MAE         MPE     MAPE      MASE
FALSE Training set 56926.03 633571.7 460713.4 -0.03288687 6.945185 0.7436052
FALSE                   ACF1
FALSE Training set 0.2570241
FALSE
FALSE Forecasts:
FALSE          Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
FALSE Jan 2014        7584737 6641718 8527755 6142515 9026959
FALSE Feb 2014        7584737 6635344 8534129 6132767 9036707
FALSE Mar 2014        7584737 6629012 8540461 6123082 9046391
FALSE Apr 2014        7584737 6622721 8546753 6113460 9056013
FALSE May 2014        7584737 6616469 8553004 6103899 9065574
FALSE Jun 2014        7584737 6610257 8559216 6094399 9075074
FALSE Jul 2014        7584737 6604084 8565390 6084957 9084516
FALSE Aug 2014        7584737 6597948 8571525 6075574 9093900
FALSE Sep 2014        7584737 6591850 8577624 6066247 9103226
FALSE Oct 2014        7584737 6585788 8583686 6056976 9112497
FALSE Nov 2014        7584737 6579761 8589712 6047760 9121714
FALSE Dec 2014        7584737 6573770 8595703 6038597 9130876
```

STL - MAdN:

```
FALSE
FALSE Forecast method: STL +  ETS(M,Ad,N)
FALSE
FALSE Model Information:
FALSE ETS(M,Ad,N)
FALSE
FALSE Call:
FALSE  ets(y = x, model = etsmodel, damped = TRUE, allow.multiplicative.trend = allow.multiplicative.tr
FALSE
FALSE   Smoothing parameters:
FALSE     alpha = 0.1233
FALSE     beta  = 1e-04
FALSE     phi   = 0.8
```

```
FALSE
FALSE   Initial states:
FALSE     l = 5615471.7851
FALSE     b = 173606.4508
FALSE
FALSE   sigma:  0.0972
FALSE
FALSE      AIC     AICc      BIC
FALSE 6143.452 6143.906 6162.997
FALSE
FALSE Error measures:
FALSE                    ME     RMSE      MAE         MPE     MAPE      MASE
FALSE Training set 54337.68 631081.9 458777.5 -0.07364717 6.937249 0.7404807
FALSE                   ACF1
FALSE Training set 0.2528558
FALSE
FALSE Forecasts:
FALSE          Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
FALSE Jan 2014        7599834 6653075 8546594 6151890 9047778
FALSE Feb 2014        7599969 6645945 8553992 6140916 9059022
FALSE Mar 2014        7600077 6638839 8561314 6129990 9070163
FALSE Apr 2014        7600163 6631760 8568566 6119118 9081207
FALSE May 2014        7600232 6624712 8575751 6108304 9092160
FALSE Jun 2014        7600287 6617700 8582874 6097549 9103024
FALSE Jul 2014        7600331 6610724 8589938 6086857 9113805
FALSE Aug 2014        7600366 6603786 8596947 6076228 9124505
FALSE Sep 2014        7600394 6596887 8603902 6065662 9135127
FALSE Oct 2014        7600417 6590028 8610806 6055161 9145673
FALSE Nov 2014        7600435 6583210 8617660 6044724 9156147
FALSE Dec 2014        7600450 6576432 8624467 6034350 9166549
```

ets - MNM:

```
FALSE
FALSE Forecast method: ETS(M,N,M)
FALSE
FALSE Model Information:
FALSE ETS(M,N,M)
FALSE
FALSE Call:
FALSE  ets(y = ts_data_o)
FALSE
FALSE   Smoothing parameters:
FALSE     alpha = 0.1428
FALSE     gamma = 0.2119
FALSE
FALSE   Initial states:
FALSE     l = 6189149.8743
FALSE     s = 0.8984 0.7596 0.938 1.2229 1.2597 1.2396
FALSE            1.0059 0.7638 0.8078 0.8864 1.0269 1.191
FALSE
FALSE   sigma:  0.0967
FALSE
FALSE      AIC     AICc      BIC
```

```
FALSE 6144.033 6146.760 6192.895
FALSE
FALSE Error measures:
FALSE                    ME     RMSE      MAE         MPE     MAPE      MASE
FALSE Training set 45241.77 628252.5 481520.9 -0.04000239 7.277118 0.7771892
FALSE                   ACF1
FALSE Training set 0.1927438
FALSE
FALSE Forecasts:
FALSE          Point Forecast   Lo 80    Hi 80   Lo 95    Hi 95
FALSE Jan 2014        9917654 8689211 11146096 8038913 11796394
FALSE Feb 2014        8522973 7456477  9589469 6891908 10154038
FALSE Mar 2014        7012478 6126191  7898765 5657019  8367937
FALSE Apr 2014        6208601 5416196  7001006 4996722  7420480
FALSE May 2014        5928833 5164834  6692832 4760398  7097269
FALSE Jun 2014        7840532 6820624  8860440 6280717  9400347
FALSE Jul 2014        9115823 7919004 10312642 7285446 10946200
FALSE Aug 2014        9648549 8370229 10926869 7693527 11603571
FALSE Sep 2014        8553364 7409986  9696742 6804718 10302010
FALSE Oct 2014        6266745 5421655  7111835 4974291  7559199
FALSE Nov 2014        5938289 5130560  6746017 4702975  7173603
FALSE Dec 2014        8020901 6920610  9121192 6338151  9703651
```

R Script

```
# Dependencies
## processing
library(readxl)
library(tinytex)
library(readr)

## graphs
library(ggplot2)
library(janitor)
library(gridExtra)
library(grid)

## formatting
library(default)
library(knitr)
library(kableExtra)
library(tidyverse)
library(scales)
library(readxl)
library(lubridate)

## forecasting packages
library(fpp2)
library(forecast)

## outlier & imputation
library(imputeTS)
library(tsoutliers)

# load data
power_data <- read_csv("https://raw.githubusercontent.com/vindication09/DATA-624/master/ResidentialCust

# Time Series
ts_data <- ts(power_data$KWH, frequency = 12, start = c(1998,1))

# Missing value imputation
ts_data <- na_interpolation(ts_data)

# STL decomposition
stl1 <- stl(ts_data, s.window = 'periodic')

# Handling outlier
outlier_func <- tsoutliers(ts_data, iterate = 2, lambda = "auto")

# Time Series - After outlier and imputation handeled
ts_data_o <- ts_data  # Let's treate outlier handled data seperatly for Modelling part.
ts_data_o[outlier_func$index] <- outlier_func$replacements

# Model#1: ARIMA
arima_auto <- auto.arima(ts_data_o)
arima_model <- forecast(arima_auto, h=12)
```

```
# Model #2: STL (no-demped) - MNN
stl_ndemp <- stlf(ts_data_o, s.window = "periodic", robust=TRUE, h = 12)

# Model #2-2: STL (demped) - MAdN
stl_demp <- stlf(ts_data_o, damped=TRUE, s.window = "periodic", robust=TRUE, h = 12)

# Model #3: ets - MNM
ets_auto <- ets(ts_data_o)
ets_model <- forecast(ets_auto, h=12)

# tsCv - ARIMA -> it takes so much time. I got the results and saved them
##arima_cv <- function(x, h){forecast(Arima(x, order = c(3, 0, 2), seasonal = c(2, 1, 0), include.drift
##e <- tsCV(ts_data_o, arima_cv, h=12)

# RMSEs -> tsCV takes lot of time to process so just saved the output
#rmse_train_arima <- arima_auto[2]
#rmse_test_arima <- sqrt(mean(e^2, na.rm=TRUE))

rmse_train_arima <- 589381.7
rmse_test_arima <- 725175

# Save output
write.csv(arima_model, file="forecasts/POWER_ARIMA_FC.csv")
```