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.

Data Exploration and Processing

Explore data. Process as needed.

```
library(tidyverse)
library(scales)
library(readxl)
library(forecast)
library(lubridate)
library(fpp2)
library(ggplot2)
library(forecast)
library(tseries)
library(imputeTS)
library(tsoutliers)
#install.packages('tsoutliers')
```

```
#power_data <- read_excel("data/ResidentialCustomerForecastLoad-624.xlsx")
library (readr)

power="https://raw.githubusercontent.com/vindication09/DATA-624/master/ResidentialCustomerForecastLoad-6

partb_data<-read_csv(url(power))

head(partb_data)
```

```
FALSE # A tibble: 6 x 3
FALSE   CaseSequence `YYYY-MMM`     KWH
FALSE          <dbl> <chr>        <dbl>
FALSE 1          733 1998-Jan   6862583
FALSE 2          734 1998-Feb   5838198
FALSE 3          735 1998-Mar   5420658
FALSE 4          736 1998-Apr   5010364
FALSE 5          737 1998-May   4665377
FALSE 6          738 1998-Jun   6467147
```

Transformed data into time-series with freq - 12.

```
ts_data <- ts(partb_data$KWH, frequency = 12, start = c(1998,1))
```

Missing value check

```r
sum(is.na(ts_data))
```

```
FALSE [1] 1
```

Impute Missing Data using TSImpute

```r
ts_data<-na.interpolation(ts_data)
```

Review the cycle of the time series to get an idea of the positions within the cycle.

```r
cycle(ts_data)
```

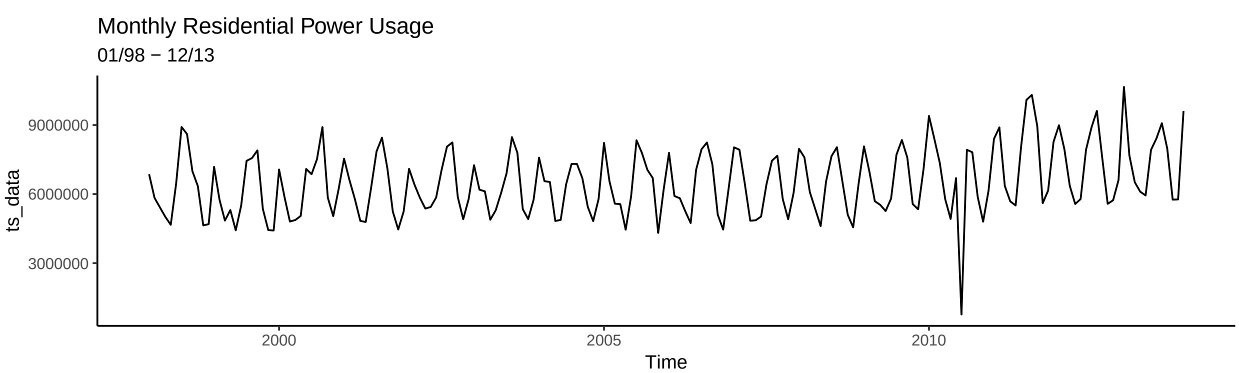
```
FALSE      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
FALSE 1998   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 1999   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2000   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2001   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2002   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2003   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2004   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2005   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2006   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2007   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2008   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2009   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2010   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2011   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2012   1   2   3   4   5   6   7   8   9  10  11  12
FALSE 2013   1   2   3   4   5   6   7   8   9  10  11  12
```

Summary Statistics

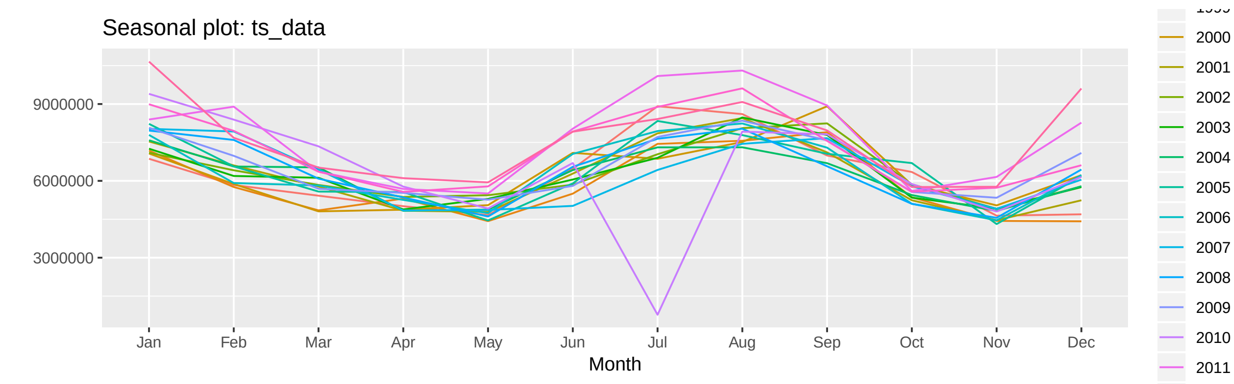
```r
summary(ts_data)
```

```
FALSE     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
FALSE   770523  5434539  6314472  6502824  7608792 10655730
```

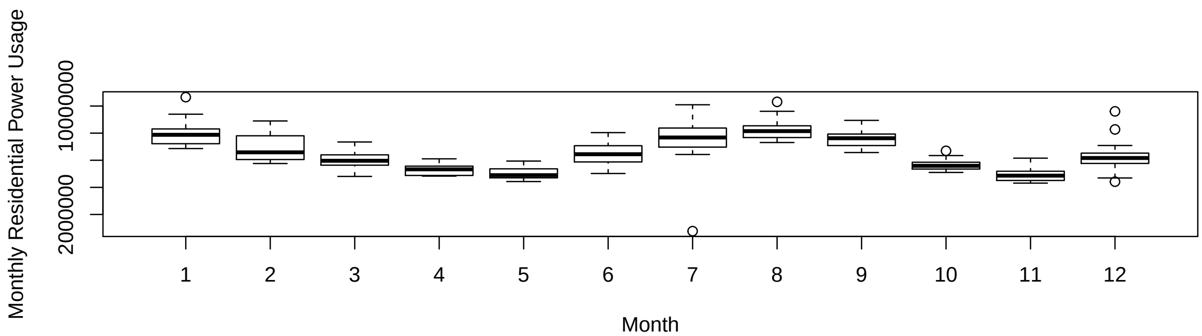
```r
#disable scientific notation (ONLY RUN ONCE)
options(scipen = 99999)

autoplot(ts_data) +
labs(title = "Monthly Residential Power Usage", subtitle = "01/98 - 12/13")+
theme_classic();
```



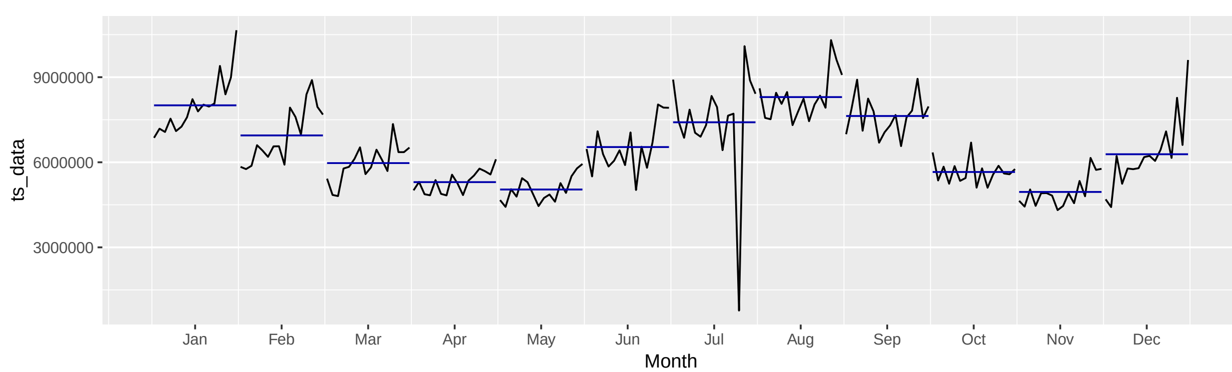
```r
ggseasonplot(ts_data);
```



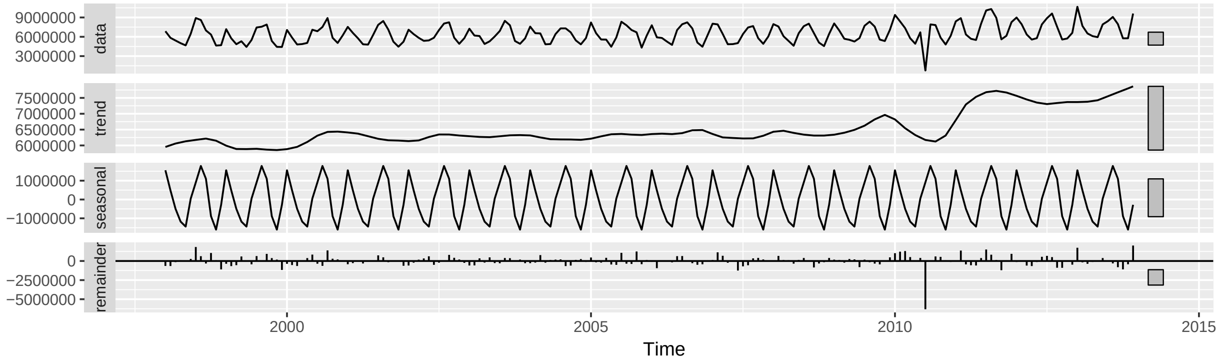

```r
boxplot(ts_data~cycle(ts_data),xlab="Month", ylab = "Monthly Residential Power Usage");
```



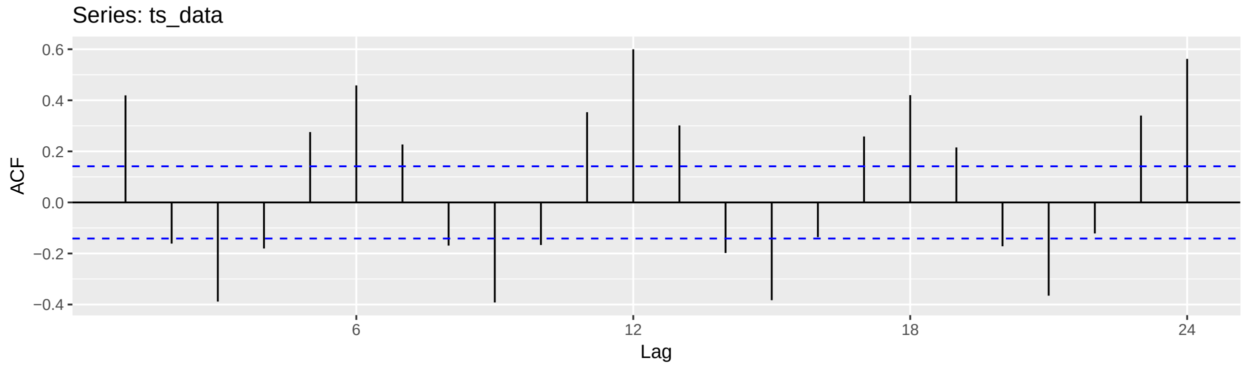

```r
ggsubseriesplot(ts_data);
```




```r
stl(ts_data, s.window = 'periodic') %>% autoplot();
```



```
ggAcf(ts_data);
```




```
tsoutliers(ts_data, iterate = 2, lambda = "auto")
```

```
FALSE $index
FALSE [1] 151
FALSE
FALSE $replacements
FALSE [1] 7757226
```

Our initial plots 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.

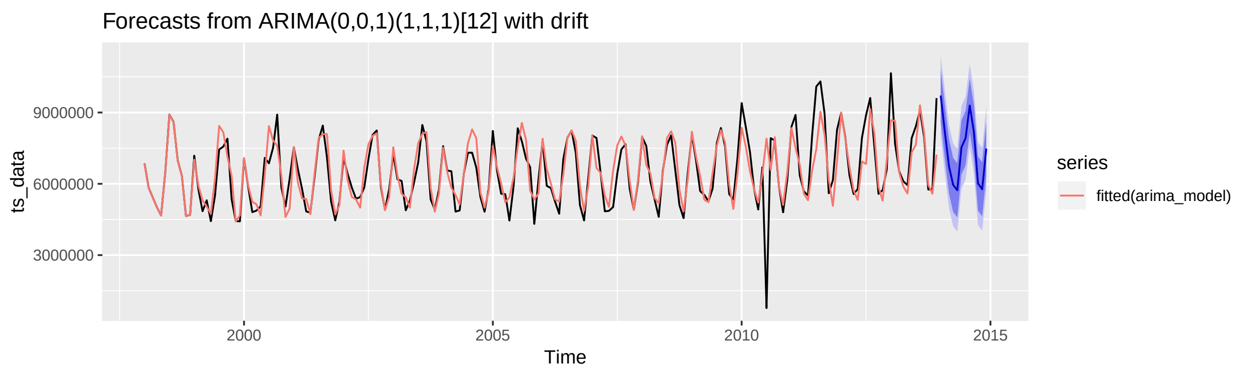
The ACF plot shows that autocorrelations are well outside the significant space indicating white noise.

Data Model

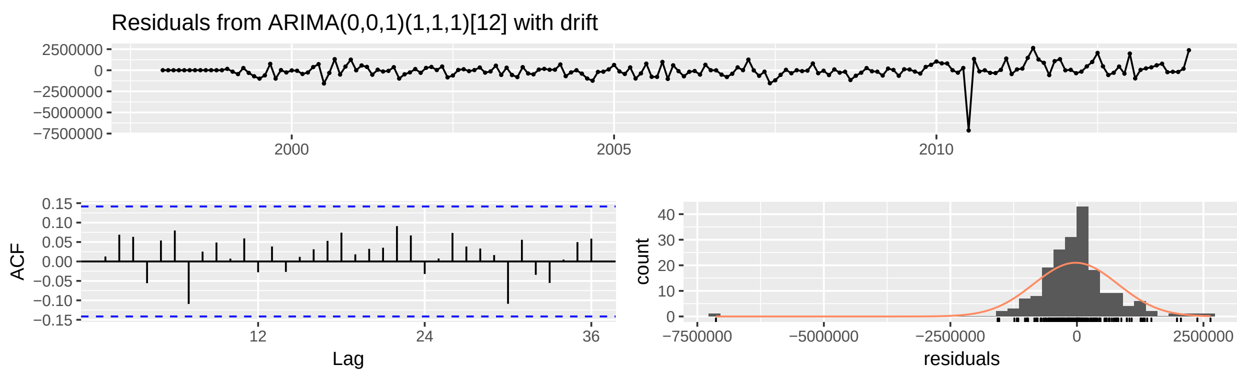
0.0.1 Model #1: ARIMA

```
arima_model <- auto.arima(ts_data)
```

```r
arima_model <- forecast(arima_model, h=12)

autoplot(arima_model) + autolayer(fitted(arima_model))
```



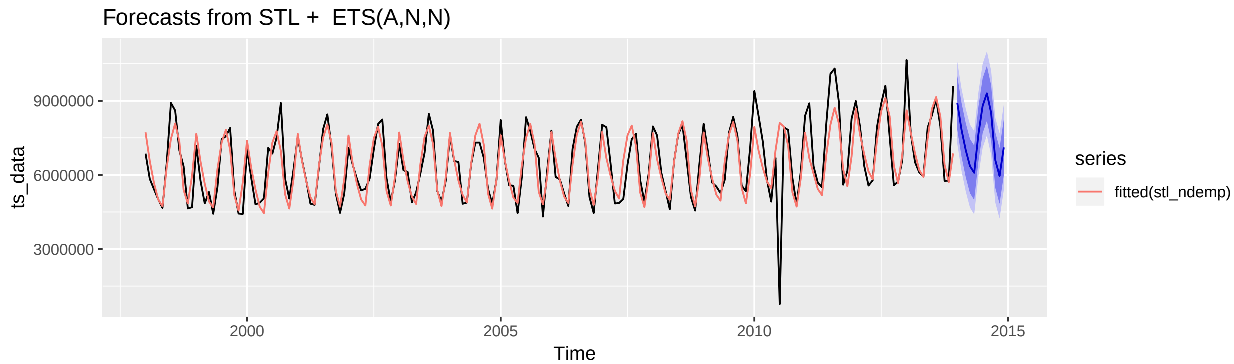

```r
checkresiduals(arima_model)
```



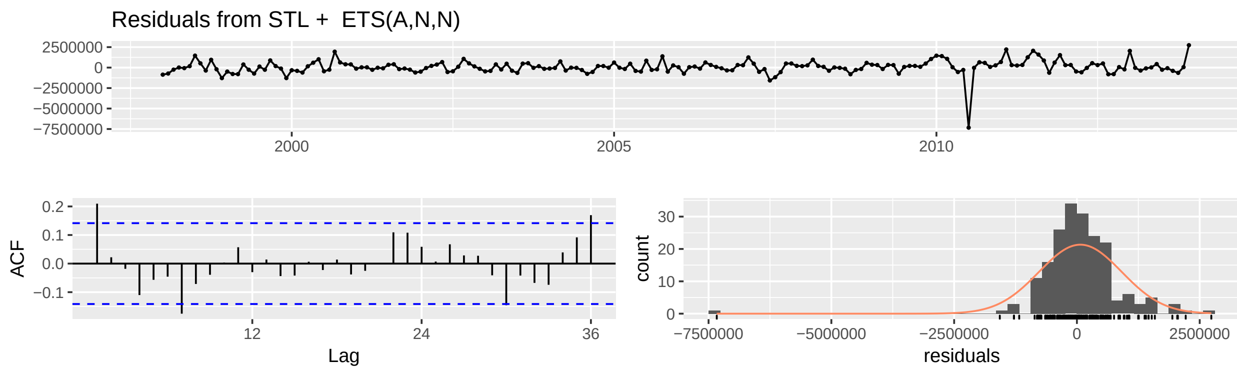

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from ARIMA(0,0,1)(1,1,1)[12] with drift
FALSE Q* = 14.209, df = 20, p-value = 0.8197
FALSE
FALSE Model df: 4.   Total lags used: 24
```

0.0.2 Model #2: STL (no-demped) - ANN

```r
#stlf - etsmodel estimation --- A,N,N is chosen.
stl_ndemp <- stlf(ts_data, damped=FALSE, s.window = "periodic", robust=TRUE, h = 12)

# forecast plot
autoplot(stl_ndemp) + autolayer(fitted(stl_ndemp))
```



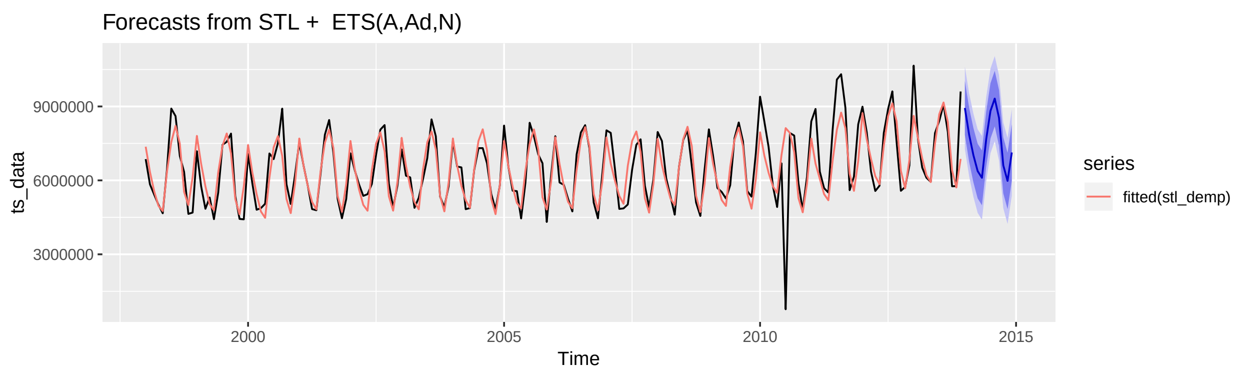

```r
checkresiduals(stl_ndemp)
```



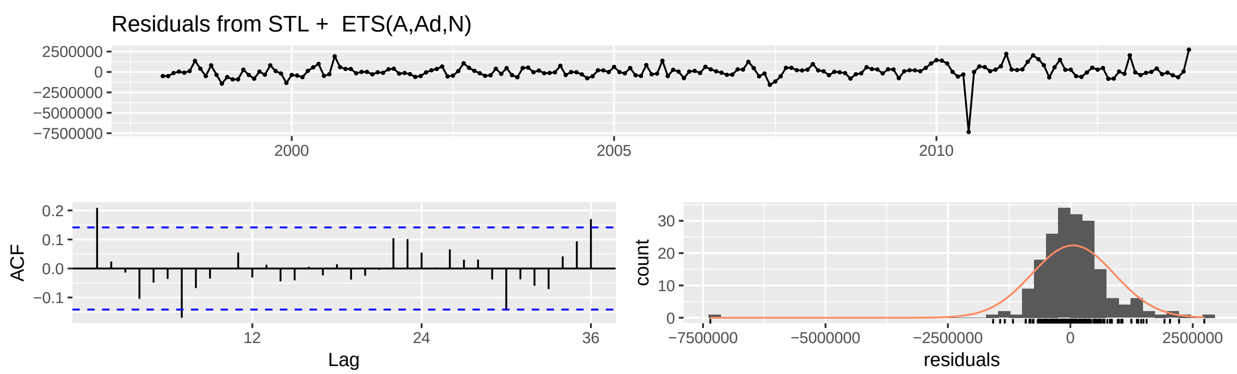

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from STL +  ETS(A,N,N)
FALSE Q* = 27.948, df = 22, p-value = 0.1774
FALSE
FALSE Model df: 2.   Total lags used: 24
```

1 Model #2-2: STL (demped) - AAdN

```r
#stlf - etsmodel estimation --- M, Ad, N is chosen.
stl_demp <- stlf(ts_data, damped=TRUE, s.window = "periodic", robust=TRUE, h = 12)

# forecast plot
autoplot(stl_demp) + autolayer(fitted(stl_demp))
```



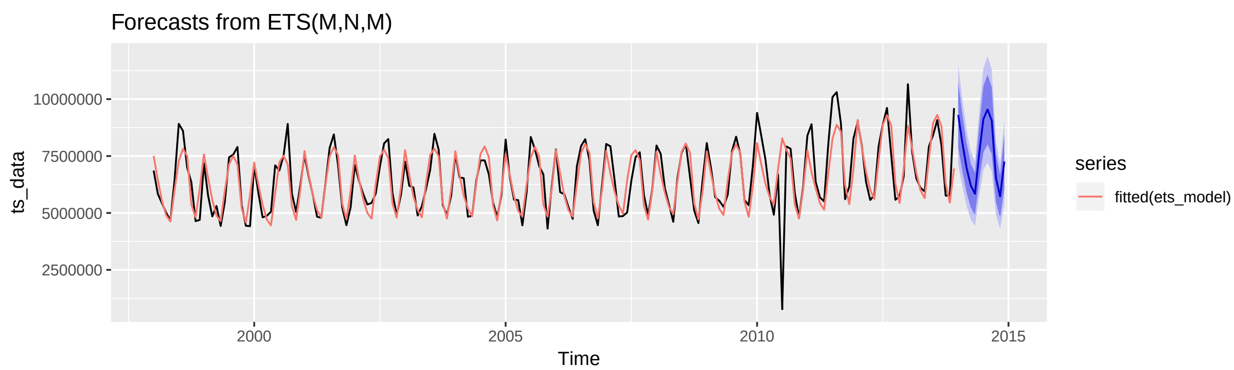

```r
checkresiduals(stl_demp)
```



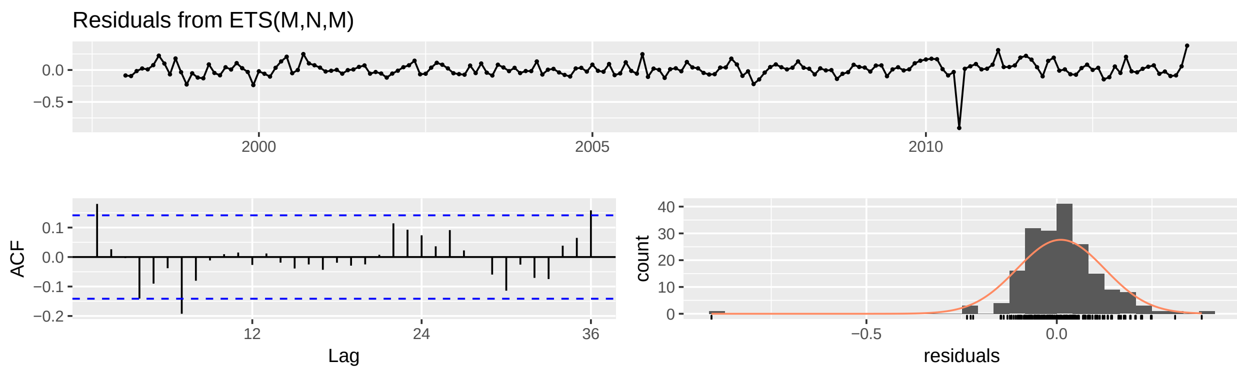

```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from STL +  ETS(A,Ad,N)
FALSE Q* = 26.06, df = 19, p-value = 0.1285
FALSE
FALSE Model df: 5.   Total lags used: 24
```

2 Model #3: ets - MNM

```r
# ETS models - MNM
ets_model <- ets(ts_data)

# forecast plot
autoplot(forecast(ets_model, h=12)) + autolayer(fitted(ets_model))
```




```r
checkresiduals(ets_model)
```




```
FALSE
FALSE   Ljung-Box test
FALSE
FALSE data:  Residuals from ETS(M,N,M)
FALSE Q* = 28.615, df = 10, p-value = 0.001438
FALSE
FALSE Model df: 14.   Total lags used: 24
```

Accuracy of Models

```r
accuracy(arima_model);
```

```
FALSE                     ME     RMSE      MAE       MPE   MAPE      MASE
FALSE Training set -25089.69 827254.2 493308.5 -5.511184 11.685 0.7080556
FALSE                    ACF1
FALSE Training set 0.01283694
```

```r
accuracy(stl_ndemp);
```

```
FALSE                        ME     RMSE      MAE      MPE     MAPE      MASE
FALSE Training set 70019.52 841778.6 510068.7 -4.24069 12.00083 0.7321119
FALSE                     ACF1
FALSE Training set 0.2096288
```

```r
accuracy(stl_demp);
```

```
FALSE                        ME     RMSE      MAE       MPE     MAPE      MASE
FALSE Training set 55479.88 841315.3 509435.8 -4.493116 12.03609 0.7312034
FALSE                     ACF1
FALSE Training set 0.2087849
```

```r
accuracy(ets_model)
```

```
FALSE                        ME   RMSE      MAE      MPE     MAPE      MASE
FALSE Training set 61009.73 835107 503972.9 -4.39013 12.04006 0.7233624
FALSE                     ACF1
FALSE Training set 0.1698584
```

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 Model #1: ARIMA residuals do not have an extended number of bins distorting the normality proximity.

The ACF plots show autocorrelations for each of our 4 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 pvalue < 0.05 is Model #3: ets - MNM. This implies that the residuals are not independent.

Forecast

We will impliment a cross validation method of testing for h=12. The process randomly chooses 12 points to measure and generate the RMSE.By definition, a lower RMSE is attributed with a better fit.

3 Model #1: ARIMA

```
arima_cv <- function(x, h){forecast(Arima(ts_data, order = c(0, 0, 1), seasonal = c(1, 1, 1),  include.c
e <- tsCV(ts_data, arima_cv, h=12)

sqrt(mean(e^2, na.rm=TRUE))
```

```
FALSE [1] 2135370
```

4 Model #2: STL (no-demped) - ANN

```
e <- tsCV(ts_data, stlf, damped=FALSE, s.window = "periodic", robust=TRUE, h=12)

sqrt(mean(e^2, na.rm=TRUE))
```

```
FALSE [1] 1014297
```

5 Model #2-2: STL (demped) - AAdN

```
e <- tsCV(ts_data, stlf, damped=TRUE, s.window = "periodic", robust=TRUE, h=12)

sqrt(mean(e^2, na.rm=TRUE))
```

```
FALSE [1] 1018495
```

Using Time series cross-validation, we compute RMSE on testset (h=12). We will pick the model with the lowest RMSE on testset as our final model. ARIMA is the worst predictor in terms of RMSE on test set (the highest RMSE) which shows that the model is seriously overfitted - low bias but very high variance. Surprisingly, STL (no-demped) - ANN, which was the worst predictor in terms of RMSE on training set, has the lowest RMSE on test set among all models. Since this is yearly forecast, tsCV(h = 12) would make sense.

Discussion

Given that 4 models we created did not vary much in terms of RMSE on training, while STL - ANN has significantly lower RMSE on test set than ARIMA, we will choose STL - ANN as our final model.

We found that ARIMA is the worst predictor and STL - AAN is the best model as RMSE on test set is the lowest, contradicting to its' RMSE on train set. It comes down to the discussion of bias-variance trade off; overfitted model cannot generalize the outcome of predictions on unseen data well.