AAEC 4984/5984 – Applied Economic Forecasting

MASTER KEY

Homework #2 – Spring 2020

So far, we have largely been focusing on the datasets included in the text. This assignment will offer you a real world example of how to utilize the tools we have learnt so far.

Instructions: In all cases, please ensure that your graphs and visuals have properly titles and axes labels, where necessary. For your convenience, I have posted my R markdown file on our course website so that you can open and alter as you see fit. Refer to the output, whenever appropriate, when discussing the results. Lastly, remember that creativity (coupled with relevance) will be rewarded.

US Natural Gas Consumption

Last week, I sat in on a presentation about the US Natural Gas industry. I was very intrigued but unfortunately, I do not have the time nor the tools to explore the dynamics in that market. Fortunately, I remembered that you have been taking the Applied Economic Forecasting class and can therefore help me with this. Can you use the tools you have been introduced to so far to make sensible future consumption forecasts? Of course you can, I will guide you through the process below.

1. Using the code below, import the NG consumption data from the U.S. Energy Information Administration (EIA) into `R`.

```
tmp <- tempfile(fileext = ".xls") # Storing the file to your computer's temporary memory
#Pull data from EIA
download.file(url = "https://www.eia.gov/dnav/ng/xls/NG_CONS_SUM_DCU_NUS_M.xls",destfile = tmp,
              mode="wb")
```

2. Using the readxl command, read the temporary file into R. Be sure to skip the first 2 rows.

```
ngdata <- readxl::read_excel(tmp, sheet = 2, skip = 2)
```

3. Drop the original date column and convert `ngdata` to a time series object starting at January 1973. Be sure to specify the proper frequency. Save this as `tsng`.

```
tsng <- ts(ngdata[,-1],start = c(1973,1), frequency = 12)
```

4. Now keep only the first column of `tsng` ["U.S. Natural Gas Total Consumption (MMcf)"] and use the window command to drop all observations before January 2001. Save this as `conng`

```
conng <- window(tsng[,1],start = c(2001,1))
```

5. Convert the units of `conng` from MMcfs to Bcfs (it is ok to save this back into `conng`)

```
conng <- conng/1000
```

6. Present a time plot of `conng`.

```
autoplot(conng) + ggtitle("Total US Natural Gas Consumption") + labs(y="Bcf",x="")
```

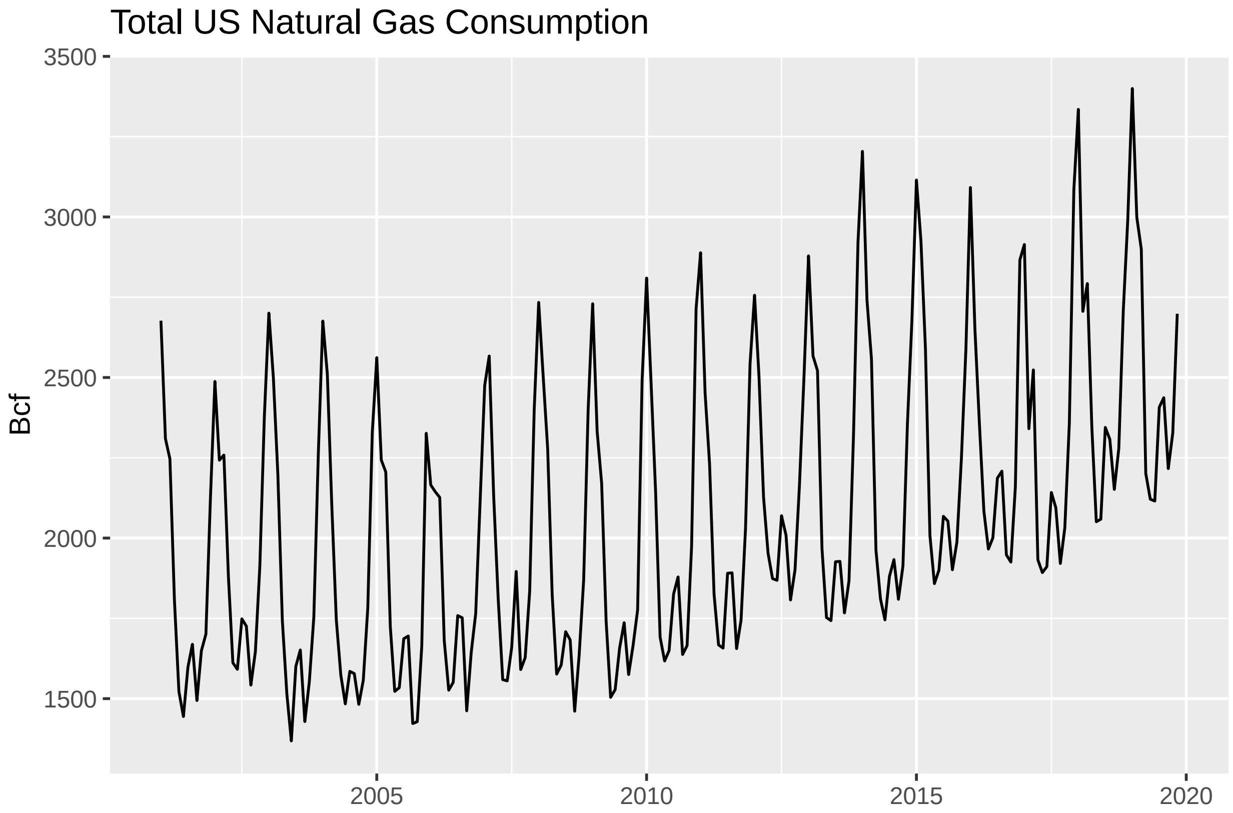


Figure 1: Total US NG Consumption

7. As we usually do, use the tools you have learnt so far to **comment** on possible seasonality and trends in the consumption data. Would you say that the series appears to be white noise? *I anticipate seeing at least 3 graphs here.*

```
g1 <- ggsubseriesplot(conng) + ggtitle("Subseries: US NG Consumption") + ylab ("Bcf")
g2 <- ggseasonplot(conng,year.labels = TRUE) + ggtitle("Seasonal plot: US NG Consumption")
g3 <- ggAcf(conng,lag.max = 36,col="blue") + ggtitle("ACF: US NG Consumption")

gridExtra::grid.arrange(g1,g2,g3, nrow=3,newpage = TRUE)
```

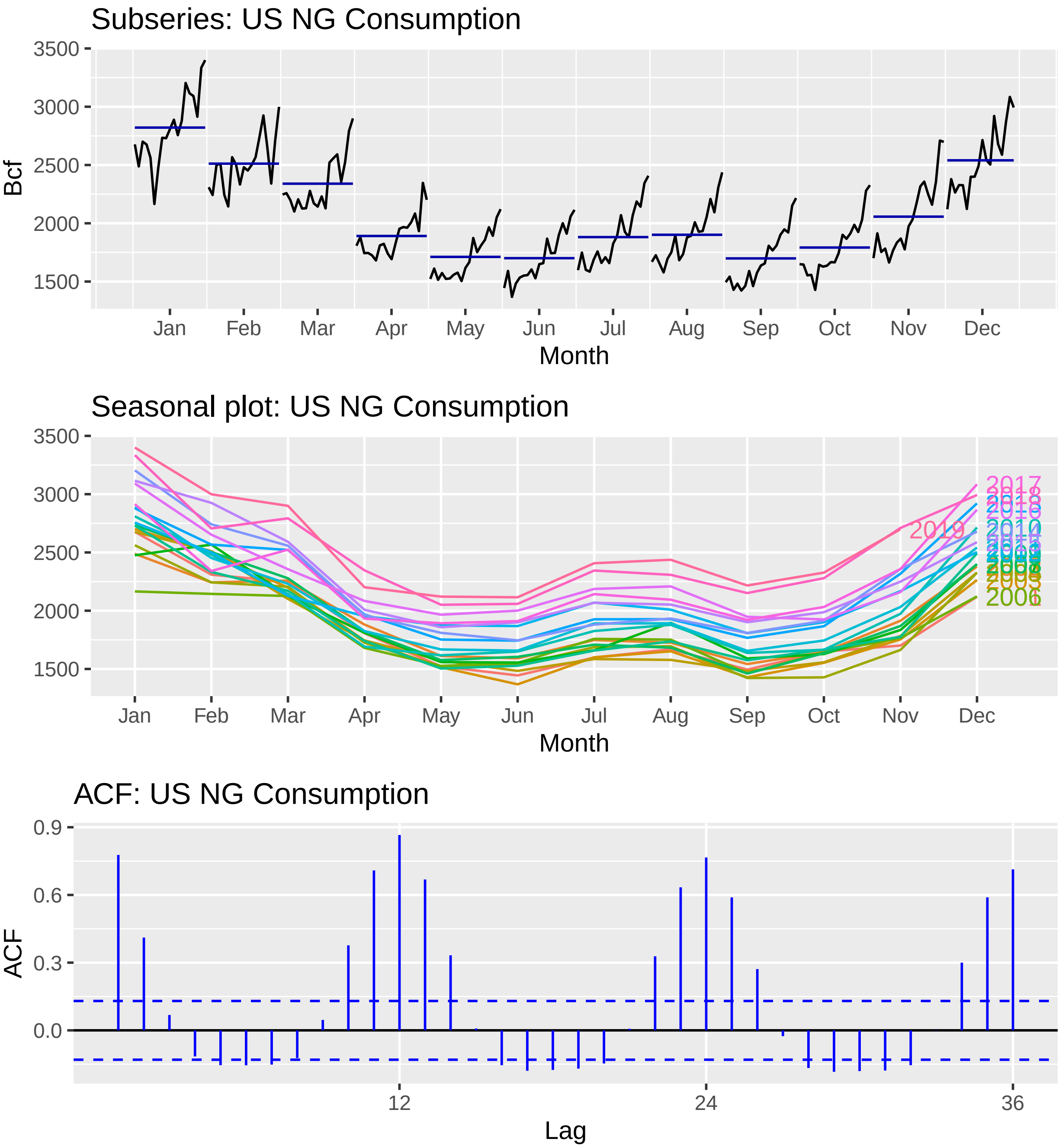


Figure 2: US NG Consumption: Seasonal Plots and ACF

**Solution: As expected the consumption of NG in the US follows a seasonal pattern. In Figure 2, for example, we notice that the consumption is highest in the colder months (November, December, January and February). In the summer months the consumption of NG falls drastically. This pattern is consistent across the years of the dataset. A quick look at the ACF confirms that there is indeed seasonality given the spike of the autocorrelations at multiples of the data frequency (12). Clearly, this series is not white noise. There is no evidence of a trend in the data.*

If you are more versed in the lingo of this market, you could have mentioned the number of heating/cooling degree days in the months as a possible factor for the heterogeneity across months. For a brief explanation, see (https://www.eia.gov/energyexplained/units-and-calculators/degree-days.php)

8. Split the data set into a training (ending December 2015) and testing set (starting January 2016). Name them

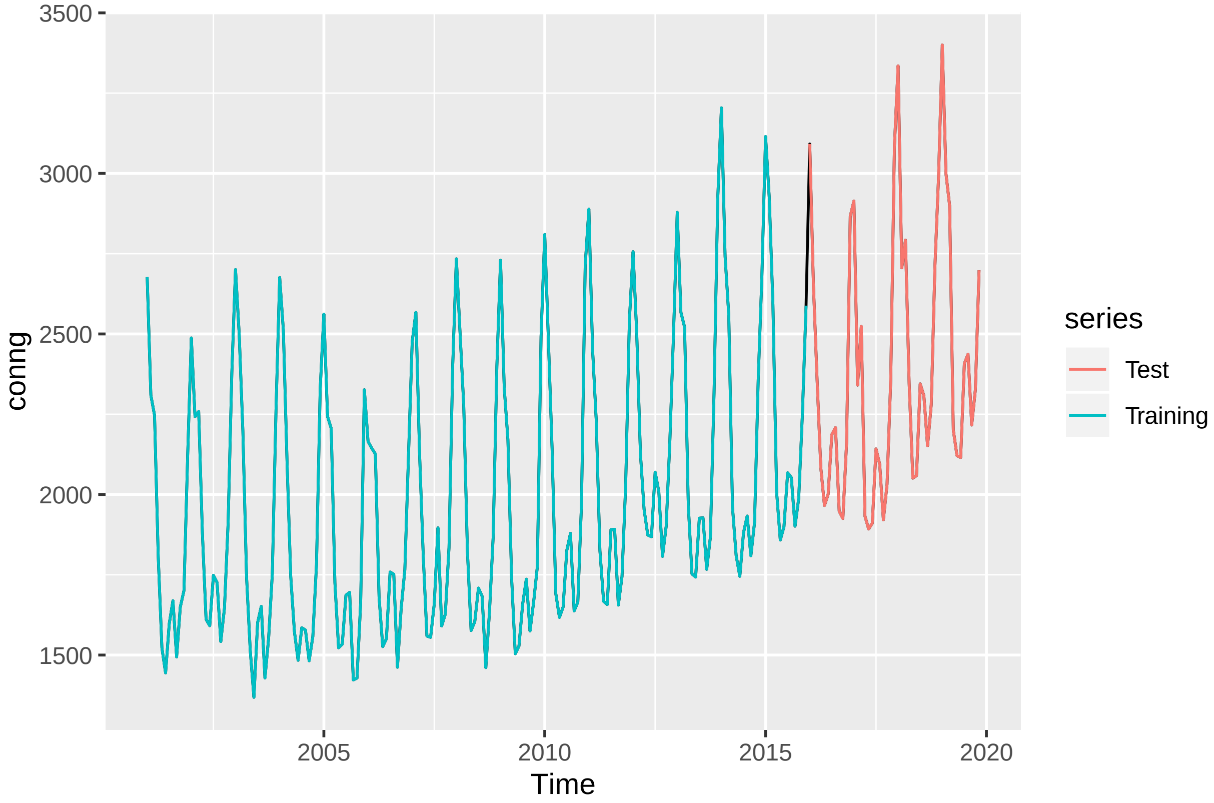


Figure 3: Confirming the split of the data

train.conng and test.conng, respectively.

```
train.conng <- window(conng,end = c(2015,12))
test.conng <- window(conng, start = c(2016,1))
```

9. Show that the data is properly split by using the autoplot and autolayer function. Be sure to include conng, train.conng and test.conng in this plot. A title is not necessary.

```
autoplot(conng) +
  autolayer(train.conng, series="Training") +
  autolayer(test.conng, series="Test")
```

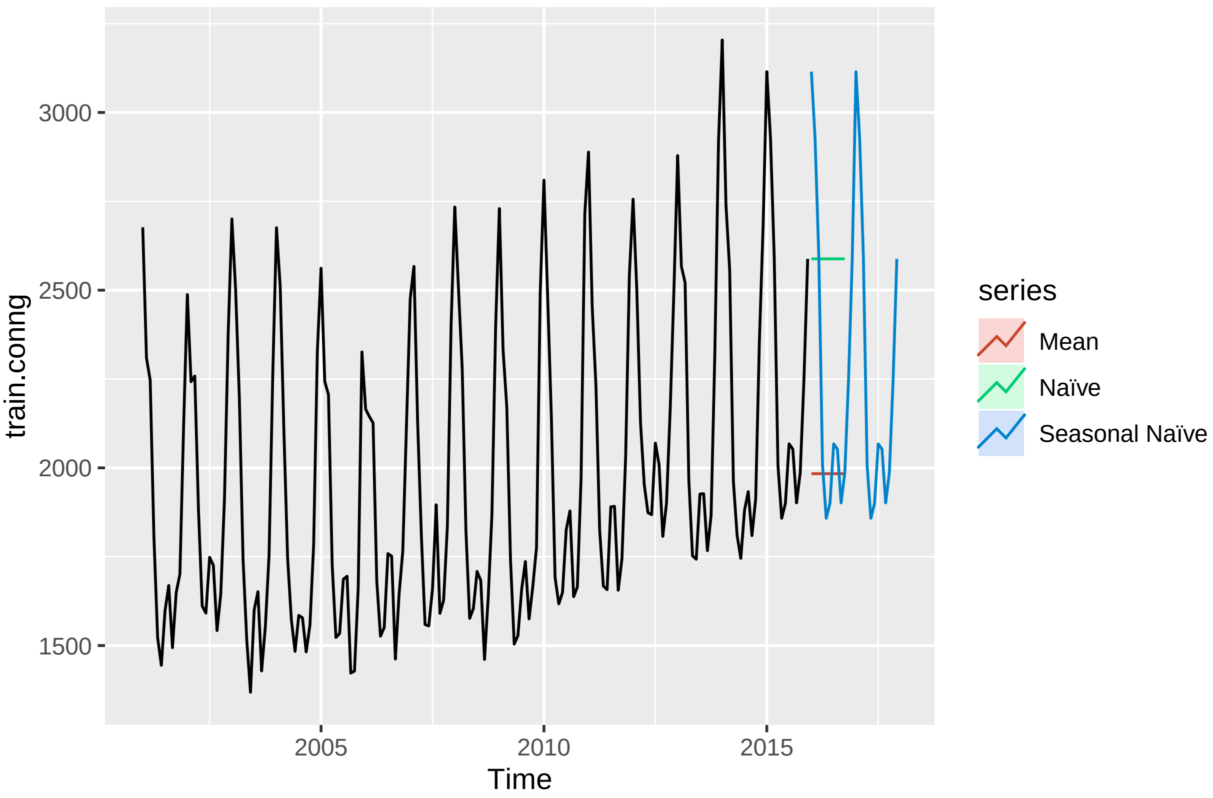
10. Given your conclusion in Part 7, use the benchmark models to forecast the training set.

```
fc1 <- meanf(train.conng)
fc2 <- naive(train.conng)
fc3 <- snaive(train.conng)
#fc4 <- rwf(train.conng,drift = TRUE)
```

Comment: There was no obvious trend in the data so I will not estimate a drift model. If you had ignored this fact and went ahead and estimated the rwf with drift anyways, you would have noticed that it lies directly on top of the naïve model.

11. In a single graph, plot the training data and forecasted series.

```
autoplot(train.conng) + autolayer(fc1,PI = FALSE, series = "Mean") +
  autolayer(fc2,PI = FALSE, series = "Naïve") + autolayer(fc3,PI = FALSE,
                                                          series = "Seasonal Naïve")
```



12. In each case, compare the accuracy of the fitted values against the actual values in the test set.

- Extract the `RMSE`, `MAE`, `MAPE`, and `MASE` statistics.
- Keep only the `Test set` row
- Round your answers to 3 dps.
- Place into a table (see my code below)
- Which model is preferred under the each method?

```
r1 <- round(accuracy(fc1,test.conng)[2,c("RMSE","MAE","MAPE","MASE")],3)
r2 <- round(accuracy(fc2,test.conng)[2,c("RMSE","MAE","MAPE","MASE")],3)
r3 <- round(accuracy(fc3,test.conng)[2,c("RMSE","MAE","MAPE","MASE")],3)
accuracy.tab <- as.table(rbind("Mean" = r1, "Naïve"= r2, "Seasonal Naïve" = r3))
accuracy.tab
```

```
##                   RMSE     MAE    MAPE   MASE
## Mean           438.302 280.668  10.778  2.729
## Naïve          495.319 459.842  21.665  4.472
## Seasonal Naïve 198.844 138.557   5.722  1.347
```

Solution: From the table above, we see that the Seasonal Naïve model is preferred across the four competing methods as it yields the smallest statistics in all the relevant categories. The MAPE, for example, indicates that we have about a 6% error rate using the seasonal naïve model versus 22% and 11%, respectively, using the Naïve and Mean forecasting models.

12. Comment on the residuals from each model. In particular,
    - Do they appear to be normally distributed?
    - Are the residuals uncorrelated? (Be sure to state the null and conclusion.)

```
checkresiduals(fc1, test = FALSE)
checkresiduals(fc1, plot = FALSE)
```

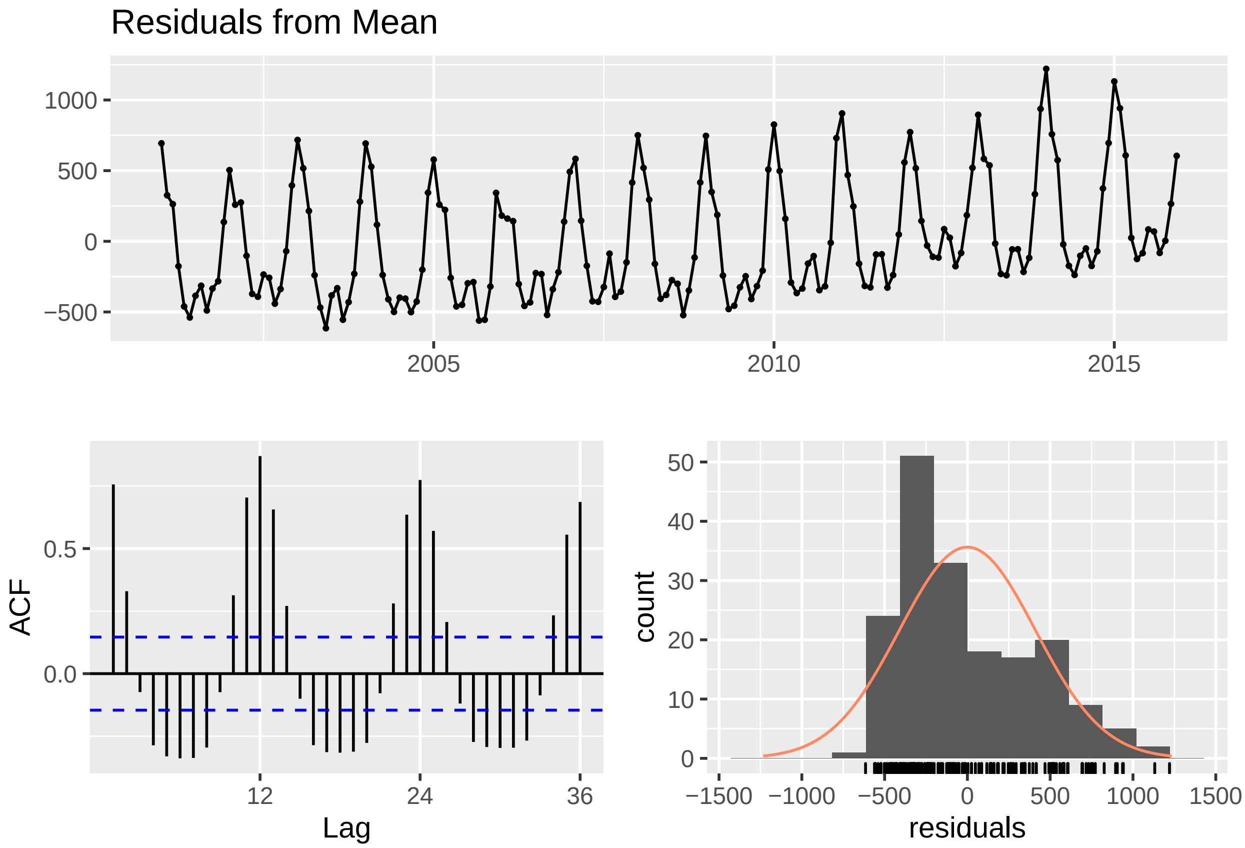


Figure 4: Residual Diagnostics: Mean Method

```
##
##  Ljung-Box test
##
## data:  Residuals from Mean
## Q* = 905.16, df = 23, p-value < 2.2e-16
##
## Model df: 1.   Total lags used: 24
```

Solution: In the Figure above, the residuals appear to be close to being normally distributed. The histogram however, doesn't exactly closely match the implied theoretical normal distribution. The time plot and ACF indicate that there is a lot of explanatory power left over in the residuals. The residuals and ACF plots both point to unexplained seasonality in the residuals. In sum, the mean forecast does not yield residuals that are white noise.

The Ljung-Box statistics yielded a pvalue < 0.01. This means that we reject the null hypothesis that there is no serial correlation and conclude that at the 1% level of significance, there is serial correlation in the residuals.

```
checkresiduals(fc2, test = FALSE)
checkresiduals(fc2, plot = FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from Naive method
## Q* = 612.1, df = 24, p-value < 2.2e-16
##
```

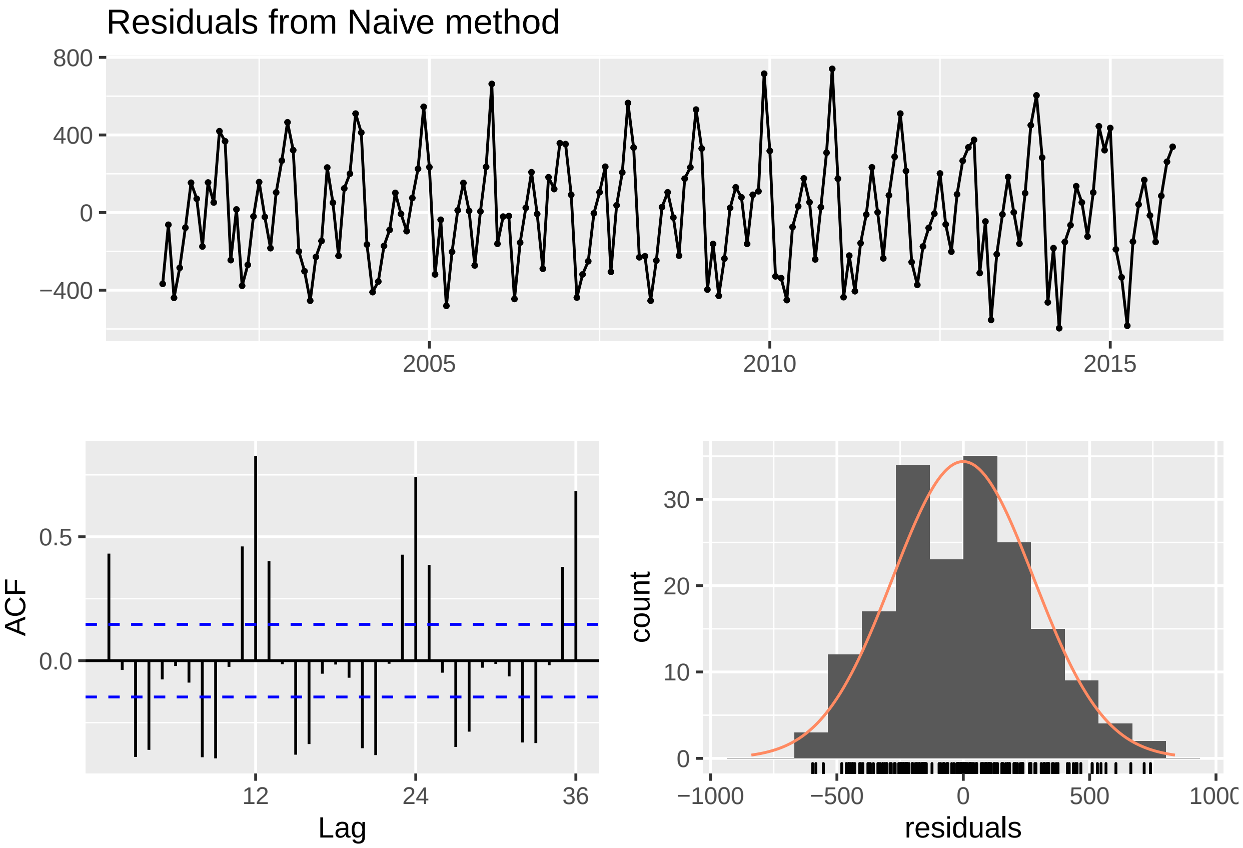


Figure 5: Residual Diagnostics: Naïve Method

```
## Model df: 0.   Total lags used: 24
```

Short Solution: The residuals appear to be normally distributed and serially correlated.

```
checkresiduals(fc3, test = FALSE)
checkresiduals(fc3, plot = FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from Seasonal naive method
## Q* = 82.711, df = 24, p-value = 2.237e-08
##
## Model df: 0.   Total lags used: 24
```

Short Solution: The residuals appear to be normally distributed and serially correlated.

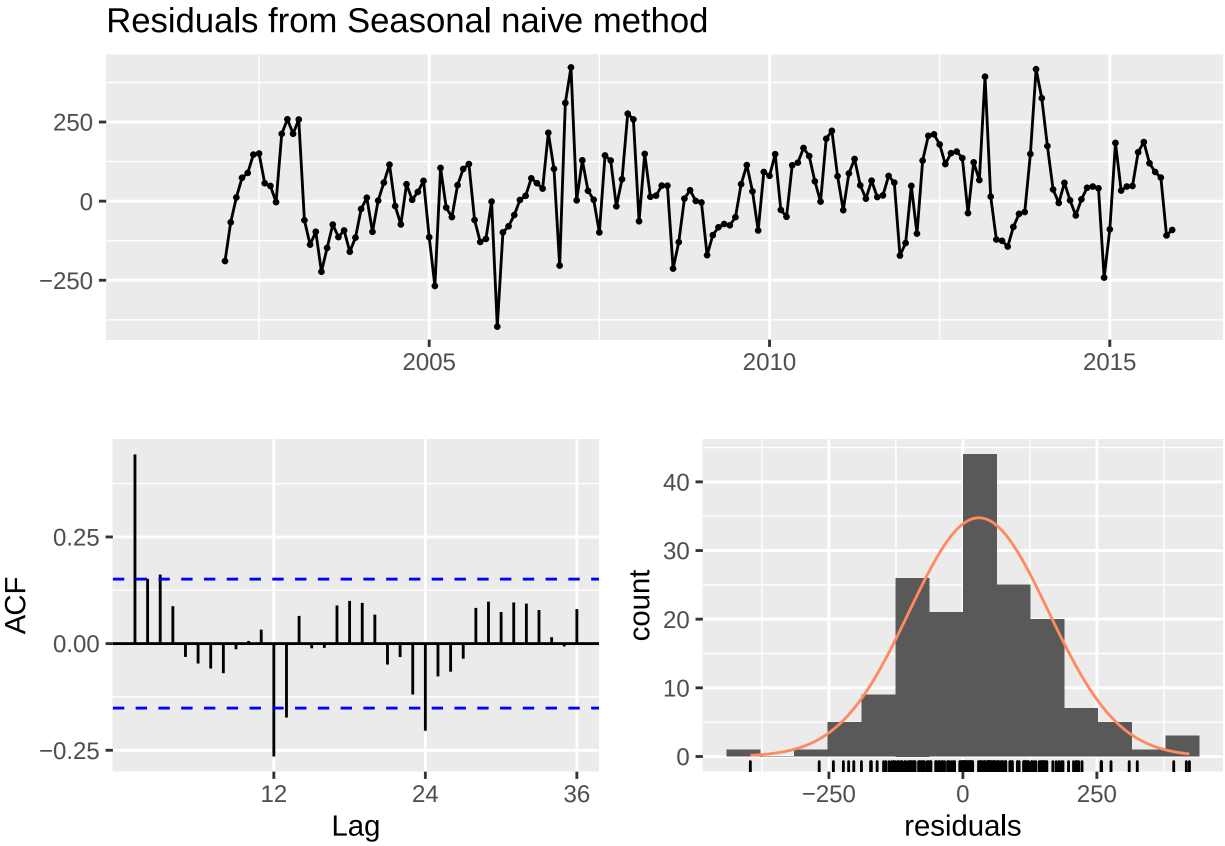


Figure 6: Residual Diagnostics: Seasonal Naïve Method