Unit Sales Heavy-Weight Trucks in US

Time Series Analysis

Derek Yan

March 13th, 2020

Abstract

The purpose of this report is to analyze monthly unit sales of heavy-weight cargo trucks in the US over a certain time period to forecast future sales. By creating such a model, consumers and producers can be better prepared for shocks towards manufacturing and related sectors of the economy. Demand for heavy-weight trucks are also correlated with the volume of products being ground-shipped: higher shipping demands indicate more need for trucks. The dataset was obtained online from the Federal Reserve Bank of St. Louis., and we employed techniques to fit the data to a time series model. We analyzed the ACF (autocorrelation function) and PACF (partial autocorrelation function) to construct a list of possible fits. It turns out that monthly truck sales follow a seasonal ARMA (AutoRegressive Moving Average) model: $(1-0.5159_{(0.0967)}B-0.1768_{(0.1106)}B^2+0.1451_{(0.0989)}B^3+0.2396_{(0.0878)}B^6)\nabla_1\nabla_{12}X_t=(1-0.8386_{(0.1813)}B^{12})Z_t$ where $\{Z_t\} \sim WN(0, 3.291)$. Diagnostics were checked for normality and independence between residuals, then the next eleven months were predicted within accurate intervals.

Introduction

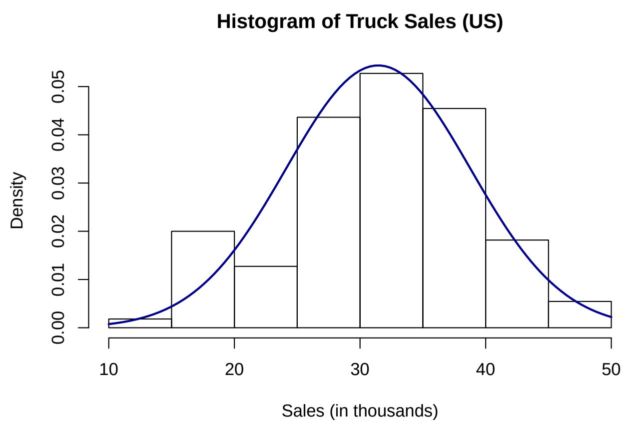
The data is from the FRED database of the Federal Reserve Bank of St. Louis. and is recorded on the first day of each month from January 2010 to January 2020,$\{X_t, t = 1, 2, ..., 121\}$. In order to create an adequate model, we first need to divide the dataset into a training set and a testing set. The training set will be monthly data from January 2010 to February 2019 and will be used to construct the model; the test set will be March 2019 to January 2020, and will be used to evaluate predictions.

Next, we need to see if any transformations are necessary in case unit sales are non-Gaussian. In order to obtain a stationary process, differencing for seasonality and trend may be required. We must analyze the ACF and PACF to determine this. We need to make sure each instance of differencing decreases variance. We will also use the ACF and PACF to determine a list of possible values for parameters of the seasonal ARMA model. We will test each possible model and select the best fitting model using the bias-corrected Aikaike Information Criterion (AICc). The best model turns out to be SARIMA $(6, 1, 0), (0, 1, 1)_{12}$. Further analysis is needed to determine if fixing any parameters continue to reduce AICc. Before prediction, we will make sure our final model is stationary and invertible. We will also have to check normality and indenpendence assumptions using residuals from our model. Finally, a prediction interval can be made to a set significance level.

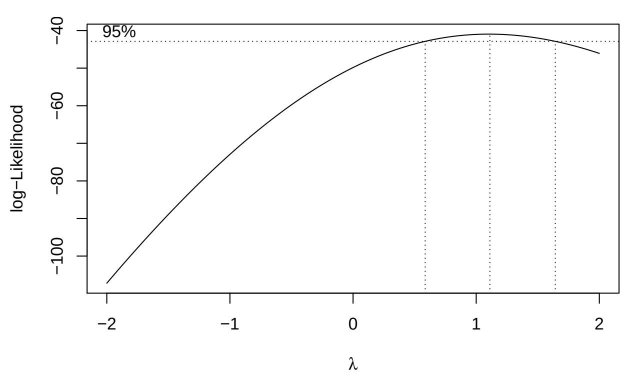
This report was created using R in RStudio and the following packages: `kableExtra`, `UnitCircle`, `MASS`, `qpcR`, `TSA`, and `dplyr`.

Time Series Analysis

Let us examine the histogram of unit sales for normality.

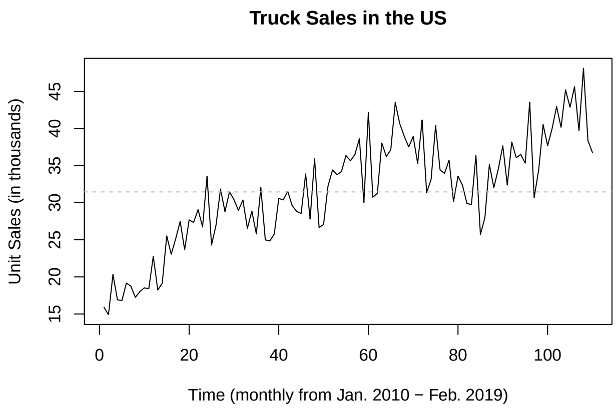


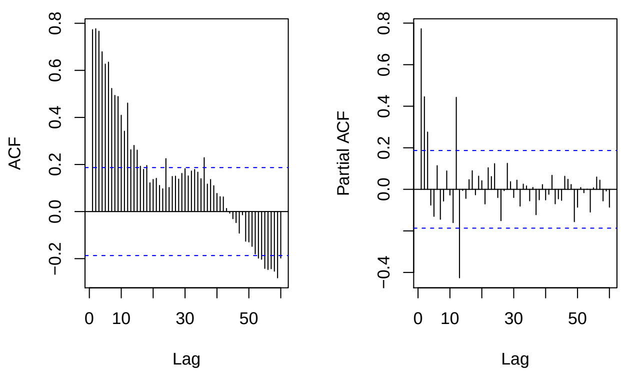
The histogram seems fairly normal, but it is necessary to cross reference with λ likelihood from the Box-Cox test.



It is apparent that $\lambda = 1$ is in the interval, thus transformation on unit sales data is unnecessary.

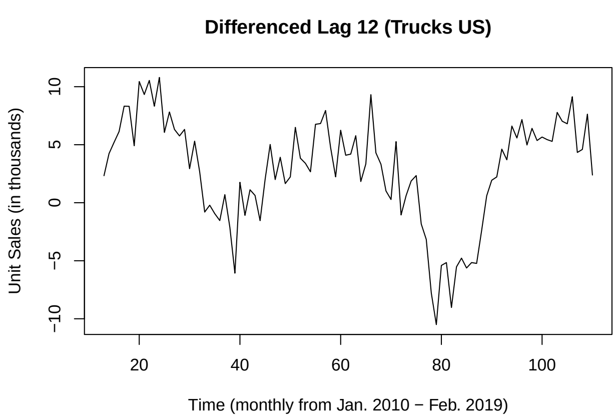
Let us examine the main features of the graph, checking, in particular, whether there exists seasonality, trend, or any sharp changes in behavior. Remember, each unit of time corresponds to one month.

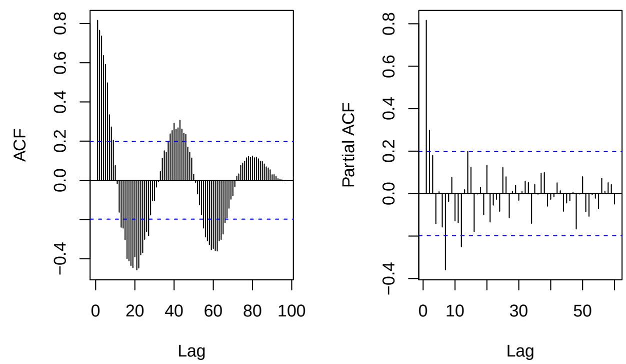




Variance
53.81802

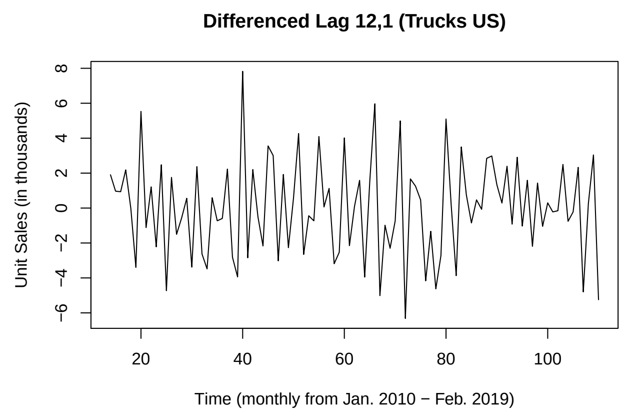
The ACF and PACF represent the autocorrelation function and partial autocorrelation function, respectively. We can see there may be seasonality at lags 12 and 24 resulting in a period of twelve months. We will difference at lag 12 and see if variance decreases.

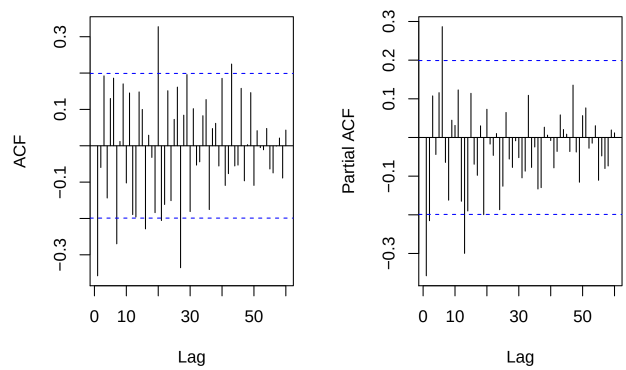




Variance
20.69142

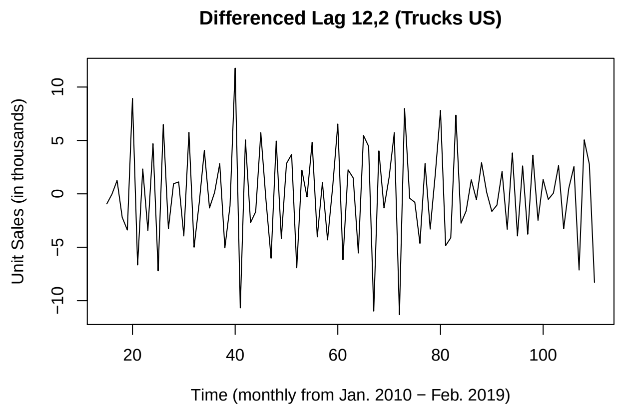
Differencing for seasonlity decreased variance, but the ACF of the seasonally differenced data still shows signs of non-stationarity; we will difference again but this time at lag 1 for linear trend, does the variance decrease?

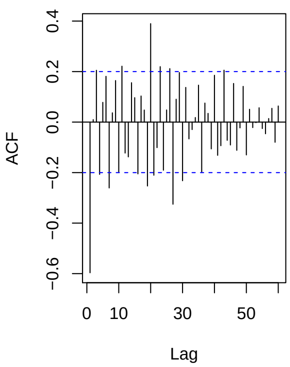


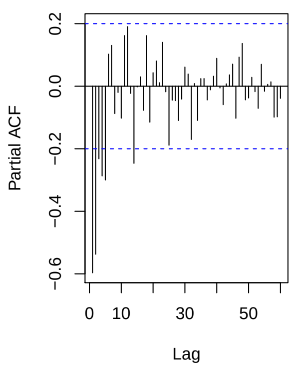


Variance
7.692066

Again, our variance decreased, and the ACF and PACF look like they can provide a list of possible parameters for an ARMA model. It is good practice to check if further differencing results in even lower variance.







Variance
20.77278

We have now over-differenced since the variance increased. Backtracking, our stationary series is $\nabla_1 \nabla_{12} X_t$.

Identifying Models

The ACF of our stationary series shows significant lags at 0, 1, 7, 16, 20, and will be possible parameter values for the moving average component. Likewise, the PACF shows possible parameter values for the autoregressive component: 0, 1, 2, 6, 13. The seasonal parameters P and Q seem like they can either be 0 or 1. Evaluating each permutation produces a total of 98 possible models (Note: The total number of permutations is actually 100 but two large models produced non-finite finite differencing errors and were discarded). Let us examine these models.

p	q	P	Q	AICc
6	0	0	1	433.4576
6	0	1	1	435.1202
2	7	0	1	435.2641
6	1	0	1	435.7574
2	1	0	1	436.4184

The table shows the best five models according to the AICc criterion. We will pick two models and check if they are stationary, invertible, and pass the required diagnostics to make predictions. Model A is SARIMA$(6, 1, 0), (0, 1, 1)_{12}$; we will fit this to our unit sales data and examine the specifications.

Paramter	Estimate	Std. Error
ar1	-0.5247	0.0991
ar2	-0.1803	0.1133
ar3	0.1759	0.1175
ar4	0.0812	0.1203
ar5	0.0814	0.118
ar6	0.2709	0.1001
sma1	-0.8309	0.1939

AICc
433.4576

Model A gives values $\hat{\phi}_1 = -0.5247_{(0.0991)}$, $\hat{\phi}_2 = -0.1803_{(0.1133)}$, $\hat{\phi}_3 = 0.1759_{(0.1175)}$, $\hat{\phi}_4 = 0.0812_{(0.1203)}$, $\hat{\phi}_5 = 0.0814_{(0.1180)}$, $\hat{\phi}_6 = 0.2709_{(0.1001)}$, $\hat{\Theta}_1 = -0.8309_{(0.1939)}$. Coefficients $\hat{\phi}_2$, $\hat{\phi}_3$, $\hat{\phi}_4$, and $\hat{\phi}_5$ are insignificant within the 95% confidence interval so we should test whether or not fixing them at zero gives us a better model. A coefficient $\hat{\phi}_p$ is considered insignificant at the $\alpha = 0.05$ level if $|\hat{\phi}_p| < |1.96\sqrt{var(\hat{\phi}_p)}|$. The AICc of our current model is 433.4576, we will test to see if any combinations of fixed values gives a lower AICc.

ar1	ar2	ar3	ar4	ar5	ar6	sma1	aicc
NA	NA	NA	0	0	NA	NA	430.0932
NA	NA	0	0	0	NA	NA	430.2129
NA	0	NA	0	0	NA	NA	430.6271
NA	NA	0	0	NA	NA	NA	431.7123
NA	NA	NA	0	NA	NA	NA	431.9162

The table above shows the model with the lowest AICc proposes that we fix parameters $\hat{\phi}_4$ and $\hat{\phi}_5$ at 0. This is the model we will use proceed with our diagnostics.

Paramter	Estimate	Std. Error
ar1	-0.5159	0.0967
ar2	-0.1768	0.1106
ar3	0.1451	0.0989
ar4	0	0
ar5	0	0
ar6	0.2396	0.0878
sma1	-0.8386	0.1813

AICc
430.0932

Model A is $(1 - 0.5159_{(0.0967)}B - 0.1768_{(0.1106)}B^2 + 0.1451_{(0.0989)}B^3 + 0.2396_{(0.0878)}B^6)\nabla_1\nabla_{12}X_t = (1 - 0.8386_{(0.1813)}B^{12})Z_t$.

Model B is SARIMA$(2,1,7),(0,1,1)_{12}$; we will fit this to our unit sales data and examine the specifications.

Paramter	Estimate	Std. Error
ar1	-1.1466	0.0409
ar2	-0.9649	0.0431
ma1	0.6867	0.1145
ma2	0.5634	0.1349
ma3	-0.2005	0.1547
ma4	0.2181	0.1742
ma5	0.1123	0.1337
ma6	0.1786	0.1278
ma7	0.1118	0.1131
sma1	-0.8187	0.2082

AICc
435.2641

Model B gives $\hat{\phi}_1 = -1.1466_{(0.0409)}$, $\hat{\phi}_2 = -0.9649_{(0.0431)}$, $\hat{\theta}_1 = 0.6867_{(0.1145)}$, $\hat{\theta}_2 = 0.5634_{(0.1349)}$, $\hat{\theta}_3 = -0.2005_{(0.1547)}$, $\hat{\theta}_4 = 0.2181_{(0.1742)}$, $\hat{\theta}_5 = 0.1123_{(0.1337)}$, $\hat{\theta}_6 = 0.1786_{(0.1278)}$, $\hat{\theta}_7 = 0.1118_{(0.1131)}$, and $\hat{\Theta}_1 = -0.8187_{(0.2082)}$. Coefficients $\hat{\theta}_3$, $\hat{\theta}_4$, $\hat{\theta}_5$, $\hat{\theta}_6$, $\hat{\theta}_7$ are insignificant. We will test if any combination of fixed parameters decreases the AICc.

ma3	ma4	ma5	ma6	ma7	aicc
NA	NA	0	0	0	431.2682
NA	NA	0	NA	0	432.5623
NA	0	0	0	0	432.8123
0	NA	0	0	0	432.8541
NA	0	0	NA	NA	432.9210

The table above is truncated to display relevant parameters only; Model B with the lowest AICc proposes that we fix parameters $\hat{\theta}_5$,$\hat{\theta}_6$, and $\hat{\theta}_7$ at 0. This is the model we will use proceed with our diagnostics.

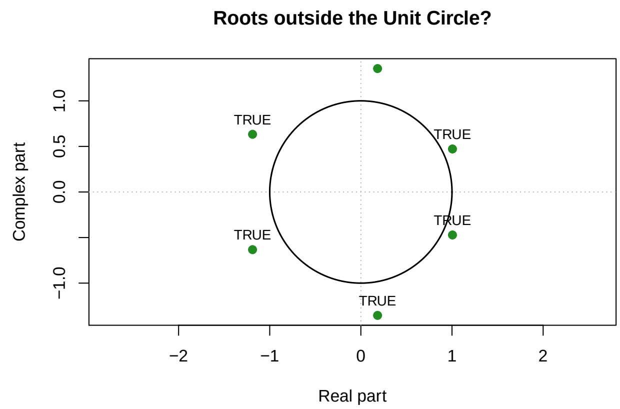
Paramter	Estimate	Std. Error
ar1	-1.1529	0.0335
ar2	-0.975	0.0363
ma1	0.6795	0.1134
ma2	0.5246	0.1281
ma3	-0.228	0.1192
ma4	0.1915	0.1042
ma5	0	0
ma6	0	0
ma7	0	0
sma1	-0.8296	0.2089

AICc
431.2682

Model B is $(1 - 1.1529_{(0.0335)}B - 0.9750_{(0.0363)}B^2)\nabla_1\nabla_{12}X_t = (1 + 0.6795_{(0.1134)}B + 0.5246_{(0.1281)}B^2 - 0.2280_{(0.1192)}B^3 + 0.1915_{(0.1042)}B^4)(1 - 0.8296_{(0.2089)}B^{12})Z_t$

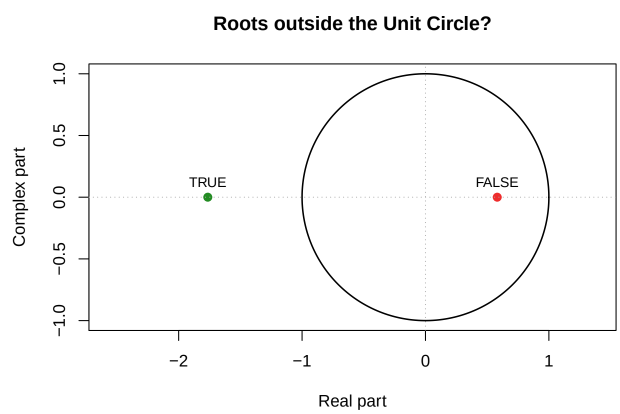
Model Diagnostics

We must make sure our models are stationary and invertible. Since Model A contains a MA(0) component, we know that by definition Model A is invertible. We will proceed to check for stationary in the autoregressive component by seeing if the roots to the polynomial lie outside the unit circle. The coefficient polynomial for Model A: $\phi(B) = 1 - 0.5159B - 0.1768B^2 + 0.1451B^3 + 0.2396B^6$

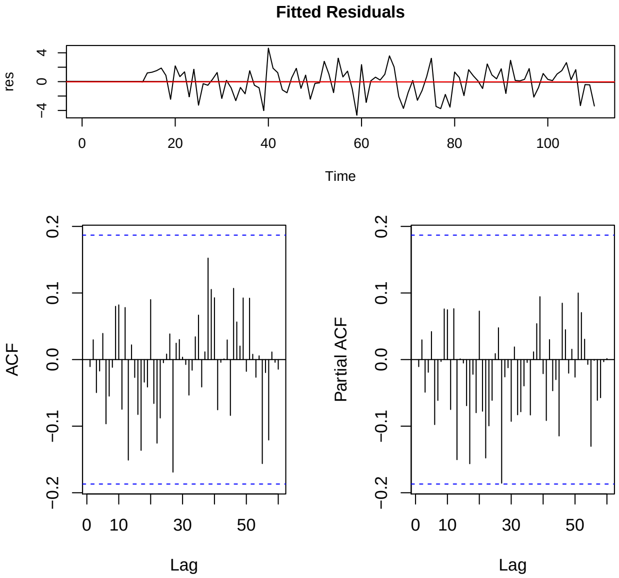


It is clear from the graph that all roots are outside of the unit circle, thus Model A is stationary and invertible.

To check for stationarity in Model B, we must examine the roots of the polynomial $\phi(B) = 1 - 1.1529B - 0.9750B^2$



One of the roots fails to be outside of the unit circle, thus this model does not have a moving average representation and is not stationary. This model will be discarded, and we will proceed to check residuals for normality and independence associated with Model A.



Mean	Variance
-0.0291903	3.291307

We can see that the residuals resemble random white noise; our mean is close to zero, and there is no significant correlation in the ACF and PACF plots. We can check normality with the Box-Pierce test. In this case, we wish to test H_0 : The residuals are independent vs. H_1 : The residuals are not independent; they exhibit correlation. The test statistic is

$$Q_W = n \sum_{j=1}^{h} \widehat{\rho_{\widehat{W}}}^2(j) \sim \chi^2_{(h-p-q)}$$

where $n = 110$, $h = \sqrt{n} \approx 10$, $p = 4$, and $q = 1$. The degree of freedom for the fit is five because there are five un-fixed free parameters.

```
	Box-Pierce test

data:  res
X-squared = 3.3927, df = 5, p-value = 0.6397
```

Because our p-value is greater than our significance level of $\alpha = 0.05$, we fail to reject the null hypothesis and do not have sufficient evidence to believe that the distribution of residuals are non-Gaussian.

The Box-Ljung test tests for the same null and alternate hypothesis as the Box-Pierce test but uses a slightly different statistic:

$$\tilde{Q}_W = n(n+2) \sum_{j=1}^{h} \frac{\widehat{\rho_{\widehat{W}}}^2(j)}{(n-j)} \sim \chi^2_{(h-p-q)}$$

```
	Box-Ljung test

data:  res
X-squared = 3.6969, df = 5, p-value = 0.5938
```

The results are the same as the Box-Pierce test.

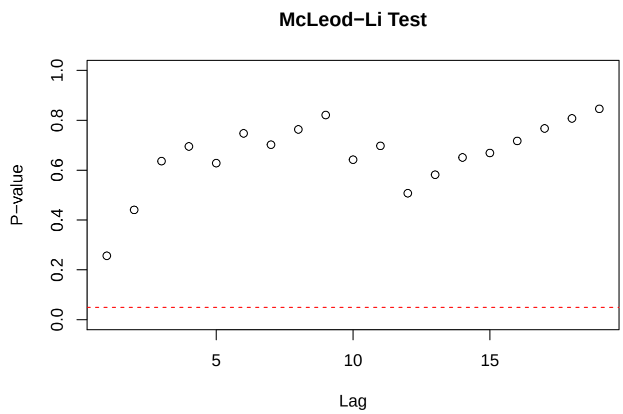
At times, there is a possibility where residuals are uncorrelated but squared residuals are. Thus we need to define the McLeod-Li test. We are interested in H_0 : The squared residuals are independent vs. H_1 : The squared residuals are not independent; they exhibit correlation. The test statistic in this case is

$$\tilde{Q}_{WW} = n(n+2) \sum_{j=1}^{h} \frac{\widehat{\rho_{\widehat{W}\widehat{W}}}^2(j)}{(n-j)} \sim \chi^2_h$$

```
	Box-Ljung test

data:  res^2
X-squared = 11.166, df = 10, p-value = 0.3447
```

Because the p-value is greater than the significance level $\alpha = 0.05$, we fail to reject the null hypothesis and do not have sufficient evidence to believe that the squared residuals are not independent.



In fact, the graph above clearly shows that the p-value is greater than the significance level for lags up to 20.

The Shapiro-Wilk test will test for normality in the residuals, H_0 : The residuals are approximately i.i.d. Gaussian vs. H_1 : The residuals are not i.i.d. Gaussian. The test statistic here is

$$W = \frac{(\sum_{i=1}^{n} a_i x_{(i)})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

where $x_{(i)}$ are the ordered residual values and a_i are constants generated from the means, variances, and covariances of the order statistics of a sample size $n = 110$ from a normal distribution. “Small” values of W are evidence of departure from normality and percentage points for the W statistic.

```
    Shapiro-Wilk normality test

data:  res
W = 0.98101, p-value = 0.1188
```

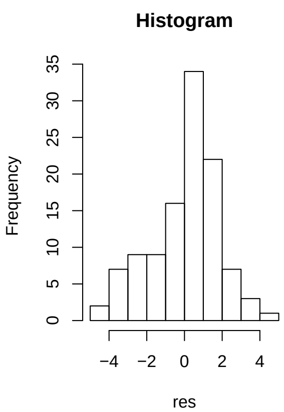
The information above shows that the p-value is greater than the significance level $\alpha = 0.05$, therefore we fail to reject the null hypothesis and do not have sufficient evidence to believe that the residuals are not identically and independently distributed (i.i.d.) Gaussian.

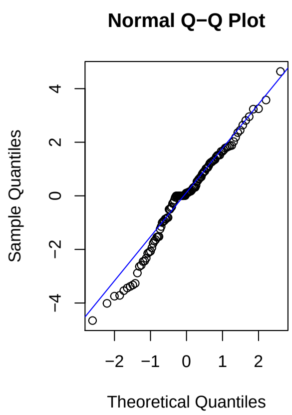
Next, we will use the `ar()` function to fit residual data to an autoregressive model and see if the outcome resembles white noise:

Order Selected	Estimated Variance
0	3.291

The residuals do, in fact, follow $WN \sim (0, 3.291)$.

Finally, let us look at the histogram and QQ plots of the residuals to confirm normality



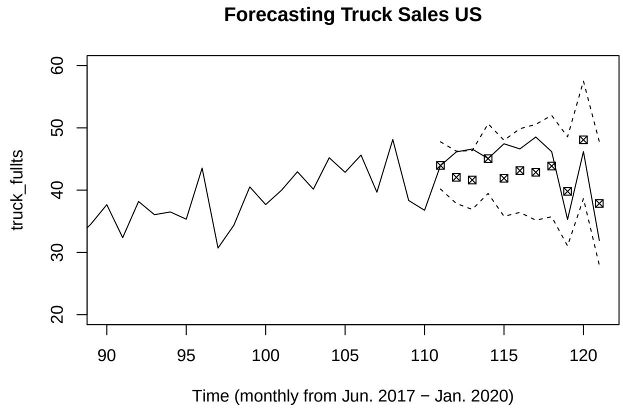


We can see that the histogram is fairly normal but note that the QQ plot shows slight trend below the theoretical quantity zero. Because the Shapiro-Wilk test above yieled a p-value greater than the significance level, it is still safe to assume the residuals are normal.

Forecasting

Our model passes all diagnostics allowing us to proceed with forecasting. This is where the test set will be used. We will create a prediction interval for the next eleven months of sales using the model we have created and compare them to the true values of the test set.

Our final model is $(1 - 0.5159_{(0.0967)}B - 0.1768_{(0.1106)}B^2 + 0.1451_{(0.0989)}B^3 + 0.2396_{(0.0878)}B^6)\nabla_1\nabla_{12}X_t = (1 - 0.8386_{(0.1813)}B^{12})Z_t$ where $\{Z_t\} \sim WN(0, 3.291)$.



The solid black line represents the true unit sales, the data points show predicted values, and the dashed lines provide a 95% prediction interval. We can see that two of the true values are on the cusp of the interval possibly due to variance, but overall, the model seems fairly accurate. We have created time series model to adequately analyze and predict unit sales of heavy-weight cargo trucks in the US.

Conclusion

The objective of this report was to construct an adequate time series model, using moving averages and autoregressive components, to predict future unit sales for heavy-weight cargo trucks in the US. Before creating the model, it was necessary to subset the data and check for transformations. We then chose two out of five possible models and further fine-tuned the parameters. Lastly, we checked normality diagnostics of residuals and were able to accurately forecast the next eleven months of sales. The final model: $(1 - 0.5159_{(0.0967)}B - 0.1768_{(0.1106)}B^2 + 0.1451_{(0.0989)}B^3 + 0.2396_{(0.0878)}B^6)\nabla_1\nabla_{12}X_t = (1 - 0.8386_{(0.1813)}B^{12})Z_t$ where $\{Z_t\} \sim WN(0, 3.291)$.

References

[1] Data obtained from the Federal Reserve Bank of St. Louis website FRED: https://fred.stlouisfed.org/series/HTRUCKSNSA

Appendix

```
# initializes necessary libraries
library(kableExtra)
library(UnitCircle)
library(MASS)
library(qpcR)
library(TSA)
library(dplyr)

trucks <- read.csv("trucks.csv") #read in dataset

truck_train <- trucks[1:110,] #separate into train and test set
truck_test <- trucks[111:120,]

for(i in 1:nrow(truck_train)){
  truck_train[i,"t"] <- i
} #adds time value t to dataframe

truck_ts <- ts(truck_train$HTRUCKSNSA) #construct time series object
g = truck_ts
m<-mean(g)
std<-sqrt(var(g))
hist(g, breaks=10, prob=TRUE,
     xlab="Sales (in thousands)",
     main="Histogram of Truck Sales (US)") #construct histogram with density curve
curve(dnorm(x, mean=m, sd=std),
      col="darkblue", lwd=2, add=TRUE, yaxt="n")

boxcox(truck_train$HTRUCKSNSA~t, data=truck_train) #check for box-cox transformations

plot(truck_ts, main="Truck Sales in the US", ylab="Unit Sales (in thousands)", xlab="Time
     (monthly from Jan. 2010 - Feb. 2019)") #examine main features

#examines acf, pacf, and variance
par(mfrow=c(1,2))
acf(truck_ts,lag.max = 60, main="  ", cex.main=1)
pacf(truck_ts,lag.max = 60, main="  ", cex.main=0.2)
kable(var(truck_ts), col.names = "Variance")

y12 <- diff(truck_ts, 12) #difference for seasonality of period 12
plot(y12,main="Differenced Lag 12 (Trucks US)", ylab="Unit Sales (in thousands)",
     xlab="Time (monthly from Jan. 2010 - Feb. 2019)")

#examines acf, pacf, and variance
par(mfrow=c(1,2))
acf(y12, lag.max = 100, main="  ")
pacf(y12, lag.max = 60, main="  ")
kable(var(y12), col.names = "Variance")

y12_1 <- diff(y12, 1) #difference once for trend
plot(y12_1,main="Differenced Lag 12,1 (Trucks US)", ylab="Unit Sales (in thousands)",
     xlab="Time (monthly from Jan. 2010 - Feb. 2019)")
```

```
#examines acf, pacf, and variance
par(mfrow=c(1,2))
acf(y12_1, lag.max = 60, main="  ")
pacf(y12_1, lag.max = 60, main="  ")
kable(var(y12_1), col.names = "Variance")

y12_2 <- diff(y12_1, 1) #difference twice for trend
plot(y12_2,main="Differenced Lag 12,2 (Trucks US)", ylab="Unit Sales (in thousands)",
     xlab="Time (monthly from Jan. 2010 - Feb. 2019)")

#examines acf, pacf, and variance
par(mfrow=c(1,2))
acf(y12_2, lag.max = 60, main="  ")
pacf(y12_2, lag.max = 60, main="  ")
kable(var(y12_2), col.names = "Variance")

#creates lists of possible parameter values to try
try_p <- c(0,1, 2, 6, 13)
try_q <- c(0,1, 7, 16, 20)
try_P <- c(0,1)
try_Q <- c(0,1)

models_all <- data.frame("p"=c(),
                     "q"=c(),
                     "P"=c(),
                     "Q"=c(),
                     "aicc"=c())
#creates possible models with permutatons of parameters
for(i in try_p){
  for (j in try_q) {
    for(k in try_P){
      for(l in try_Q){

    add <- data.frame("p"=i,
                     "q"=j,
                     "P"=k,
                     "Q"=l
                        )
    models_all <- rbind(models_all, add)
    }
    }
    }
}

#removes two values that causes non-finite finite difference error
##model parameter values were high and not close to be considered as a good fit.
models_all <- models_all[c(-75,-92),]


for(i in  1:nrow(models_all)){

  models_all[i,"aicc"] <- AICc(arima(truck_ts,order=c(models_all[i,1],1,models_all[i,2]),
                          seasonal=
```

```r
                              list(order=c(models_all[i,3],1,models_all[i,4]),period=12),
                            method="ML",optim.control=list(maxit=1000)))
} #tests each model and records AICc


#nicely outputs top five models and their AICc
k_p <- c(6,6,2,6,2)
k_q <- c(0,0,7,1,1)
k_P <- c(0,1,0,0,0)
k_Q <- c(1,1,1,1,1)
k_aicc <- c(433.4576,435.1202,435.2641,435.7574,436.4184)

kable(cbind(k_p,k_q,k_P,k_Q,k_aicc), col.names = c("p","q","P","Q","AICc"))

#create model
final_fit <- arima(truck_ts, order = c(6, 1, 0), seasonal=
                       list(order= c(0, 1, 1), period=12),
                     method = c("ML"),optim.control=list(maxit=1000))
#nicely outputs model
par <- c("ar1", "ar2", "ar3", "ar4", "ar5", "ar6","sma1")
est <- c(-0.5247,  -0.1803,  0.1759,  0.0812,  0.0814,  0.2709,  -0.8309)
se <- c(0.0991,   0.1133,  0.1175,  0.1203,  0.1180,  0.1001,   0.1939)

kable(cbind(par, est, se), col.names = c("Paramter", "Estimate", "Std. Error"))

kable(AICc(arima(truck_ts, order = c(6, 1, 0), seasonal=
                     list(order= c(0, 1, 1), period=12),
                   method = c("ML"),optim.control=list(maxit=1000))), col.names = "AICc")


#code to check to see if any parameter estimates should be fixed to increase model fit
#for loop tries every permutation of insignificant estimates and assigns AICc respectively
opt <- c(NA,0)
v <- 1:2
count1 <- 1

coef_model1 <- data.frame(
      "ar1"= c(),
      "ar2"= c(),
      "ar3"= c(),
      "ar4"= c(),
      "ar5"= c(),
      "ar6"= c(),
      "sma1"= c()
      )

for (v1 in v) {
  for (v2 in v) {
  for (v3 in v) {
  for (v4 in v) {

    aicc <-AICc( arima(truck_ts, order = c(6, 1, 0), seasonal=
                          list(order=c(0,1,1),period=12),
```

```
                        method =c("ML"),
        fixed = c(
          NA,
          opt[v1],
          opt[v2],
          opt[v3],
          opt[v4],
          NA,
          NA
          )
          )
    )

    add <- data.frame(
      "ar1"= "NA",
      "ar2"= opt[v1],
      "ar3"= opt[v2],
      "ar4"= opt[v3],
      "ar5"= opt[v4],
      "ar6"= "NA",
      "sma1"= "NA",
      "aicc"= aicc
      )

    coef_model1 <- rbind(coef_model1, add)
    count1=count1+1

}
}
}
}

coef_model1_sorted <- coef_model1[with(coef_model1,order(coef_model1$aicc)),]
kable(coef_model1_sorted[1:5,], row.names = FALSE) #nicely outputs


#apply fixed values to new model
final_fit2 <- arima(truck_ts, order = c(6, 1, 0), seasonal=
                      list(order= c(0, 1, 1), period=12),
                    method = c("ML"),optim.control=list(maxit=1000),
                    fixed = c(NA,   NA, NA, 0,  0,  NA, NA))

#nice output
par <- c("ar1", "ar2", "ar3", "ar4", "ar5", "ar6","sma1")
est <- c(-0.5159,  -0.1768,  0.1451,    0,    0,  0.2396,  -0.8386)
se <- c(0.0967,   0.1106,  0.0989,    0,    0,  0.0878,   0.1813)

kable(cbind(par, est, se), col.names = c("Paramter", "Estimate", "Std. Error"))


kable(AICc(arima(truck_ts, order = c(6, 1, 0), seasonal=
                   list(order= c(0, 1, 1), period=12),
                 method = c("ML"),optim.control=list(maxit=1000),
```

```r
                fixed=c(NA,    NA, NA, 0,  0,  NA, NA))), col.names = "AICc")


#check if roots are outside the unit circle
uc.check(c(1,-0.5159,  -0.1768,  0.1451 ,   0  ,  0 , 0.2396), print_output = FALSE)

#creates residual plots
res = residuals(final_fit)
layout(matrix(c(1,1,2,3),2,2,byrow=T))
ts.plot(res,main = "Fitted Residuals")
t = 1:length(res)
fit.res = lm(res~t) #trend line of residuals
abline(fit.res)
abline(h = mean(res), col = "red")

#examines acf, pacf, mean, and variance
par(mfrow=c(1,2))
acf(res, lag.max = 60, main="  ")
pacf(res, lag.max = 60, main="  ")

kable(cbind(mean(res),var(res)), col.names = c("Mean","Variance"))

#box-pierce test
Box.test(res, lag = 10, type = c("Box-Pierce"), fitdf = 5)

#box-ljung test
Box.test(res, lag = 10, type = c("Ljung-Box"), fitdf = 5)

#Mcleod-li test
Box.test(res^2, lag = 10, type = c("Ljung-Box"), fitdf = 0)

#mcleod-li test for multiple lags
McLeod.Li.test(final_fit, main="McLeod-Li Test")

#shapiro-wilk test for normality
shapiro.test(res)


#fits residual data to an ar model
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
kable(cbind(0,3.291), col.names = c("Order Selected","Estimated Variance"))


# Test for normality of residuals
par(mfrow=c(1,2))
hist(res,main = "Histogram")
qqnorm(res) # QQ Plot
qqline(res,col ="blue")


truck_fullts <- ts(trucks$HTRUCKSNSA) #true value
mypred = predict(final_fit, n.ahead=11)  # predicted values
plot(truck_fullts, xlim=c(90,121), ylim=c(20, 60),main="Forecasting Truck Sales US",
```

```
     xlab="Time (monthly from Jun. 2017 - Jan. 2020)")
points(111:121,mypred$pred, pch = 7)
lines(111:121,mypred$pred+1.96*mypred$se,lty=2)
lines(111:121,mypred$pred-1.96*mypred$se,lty=2)
```