US Candy Production Data Analysis

Ruoxin Wang

03/11/2022

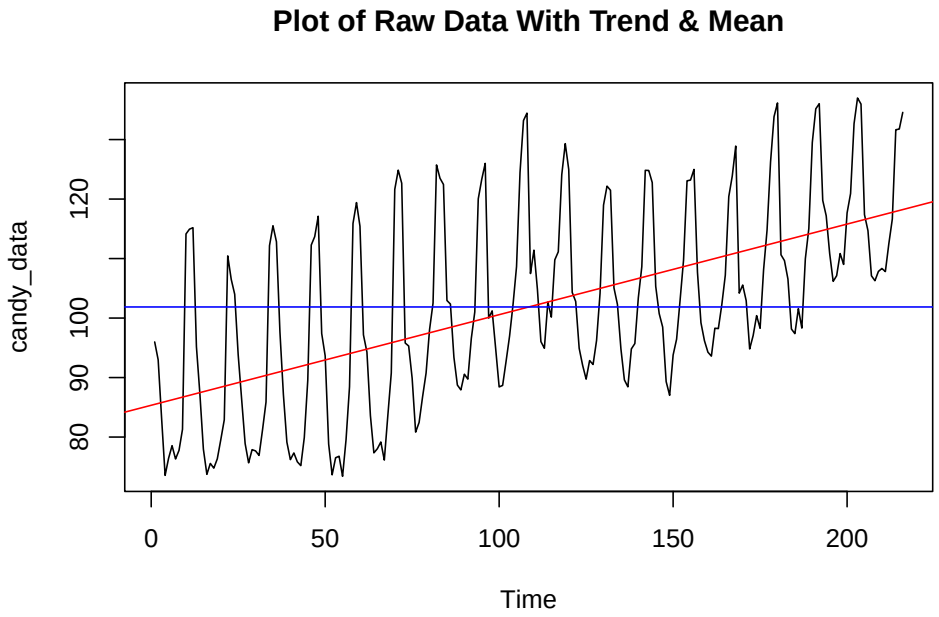
Abstract

The purpose of this project was to apply time series analysis techniques from PSTAT 174 to analyze a real-world time series data set "US Candy Production, 1982 - 1999" and predict future data points via forecasting. My questions addressed in this project is whether the candy production in the US has seasonality and how it going to develop in the following years. To analyze this time series data set, I first plot out the raw data and utilize box-cox transformation to stable the variance and differenced data to make it stationary. Then, plotting out the sample ACF and PACF graph allows me to choose candidate models needed to be further fitted, which are **SARIMA(1, 1, 1)(0, 1, 1)[12], SARIMA(0, 1, 2)(0, 1, 1)[12], SARIMA(2, 1, 2)(0, 1, 1)[12]**. Selecting the best model according to the AICc scores and use diagnostic checking with p-value of three additional tests greater than 0.05 to confirm that the model residual ACF, PACF, normaltiy, independence, fitting AR(0) work well to behave like a white noise. Finally, I forecast and make prediction with a 95% confidence interval for the following years. According to the forecasting result, all the prediction points are inside the 95% confidence interval and show a seasonal pattern with a increasing trend, which can make a further conclusion that US candy production was affected by a seasonal factor and the amount of production would increase in the following years.

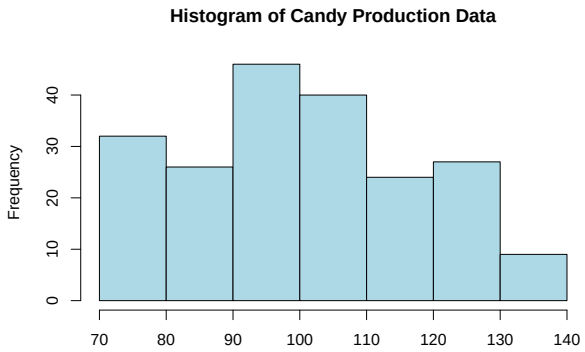
Introduction

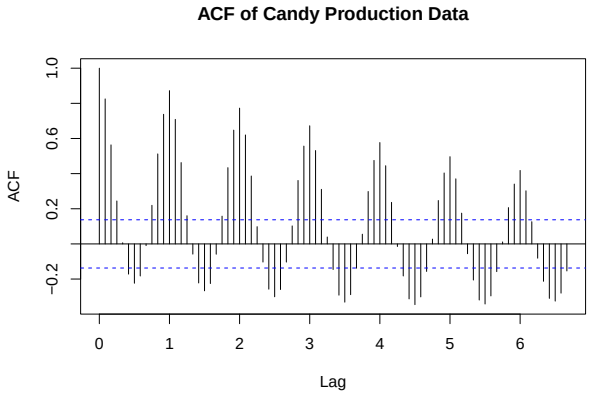
The problem I want to solve is whether the candy production in the US affected by a seasonal factor and what the possible trend, increasing or decreasing, it would have in the following years.The data set was collected from the website: https://www.kaggle.com/rtatman/us-candy-production-by-month. The US Candy Production data set includes monthly data that tracks the industrial production of candy every month in the United States from January 1972 to August 2017 with totally 500 data points. However, I only use the data from January 1982 to December 1999 due to the reason of controlling the number of observations. The reason that I want to choose this data set is that candy seems to be an important part in Americans' daily life, which people consumes a lot on holidays like Halloween and Christmas especially. I was wondering whether this kind of high demanding pushes the candy production in the US as time goes. The software I use to analyze this time series is R. In my analysis, I use box-cox transformation and differencing at lag 12 and lag 1 to make the time series stationary for further model fitting. I choose **SARIMA(1, 1, 1)(0, 1, 1)[12]** as my final model after comparing its AICc score with my other options like **SARIMA(0, 1, 2)(0, 1, 1)[12], SARIMA(2, 1, 2)(0, 1, 1)[12]**, and going over diagnostic checking on the model residual ACF, PACF, normaltiy, independence, fitting AR(0) with Box-Pierce test, Box-Ljung test, and McLeod-Li test to see if it behaves like a white noise. After forecasting 60 prediction points ahead, my data implies that there was a strong seasonality on candy production in the US from 1982 to 1999 and the amount of candy production in the following years follows a possible increasing trend, which means that the high consumers' demand would push the candy production to become more.

Data Plot And Analysis



From the plot of time series above, it shows that the original data is highly nonstationry because of its significant trend and seasonality. It may suffer from the non-constant variance and mean due to this. But there is no apparent sharp changes in the overall pattern of the data.



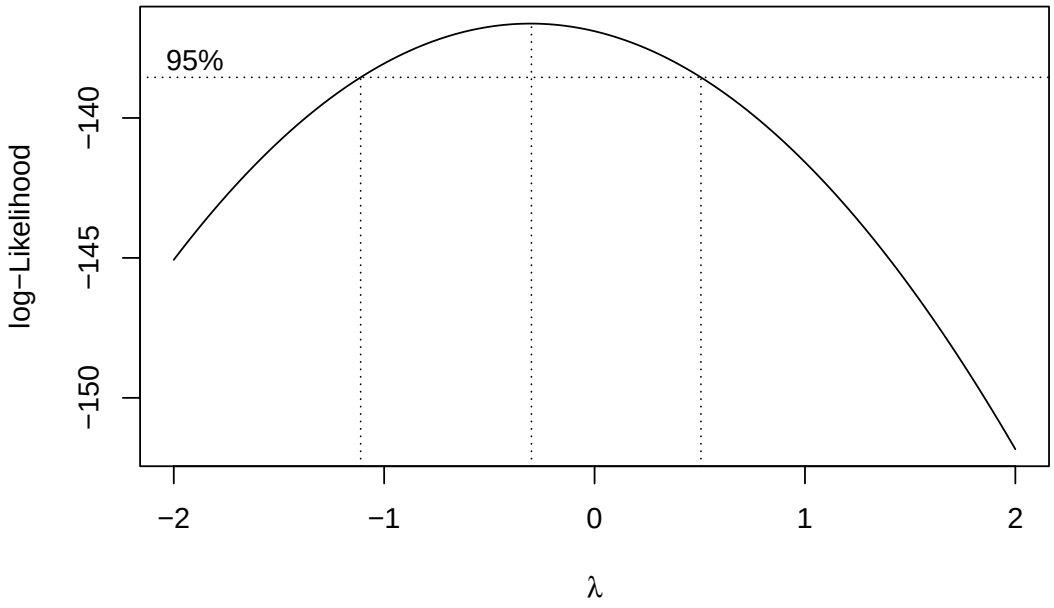


Before doing the transformation, I double check with the histogram and ACF of the time series. They show that the histogram is skewed and ACF remains large and periodic. Therefore, I plan to do the transformation and differencing for the next step to make it stationary.

Transfromations

1) Box-Cox Transformation

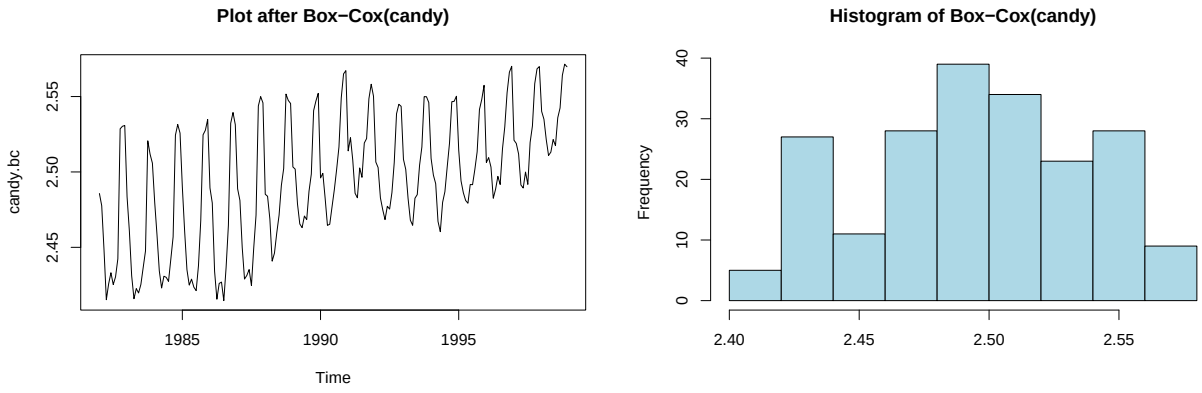
```
## [1] "The value of lambda is -0.3"
```



According to the Log-Likelihood plot above, the 95% confidence interval of λ contains 0. The value of λ is -0.3, which is also close to 0. Therefore, I choose to use ***Box-Cox transformation*** instead of other transformations for my data.

```
## [1] "Variance of candy data before transformation is 278.343739457128"
```

```
## [1] "Variance of candy data after transformation is 0.00174927057442286"
```

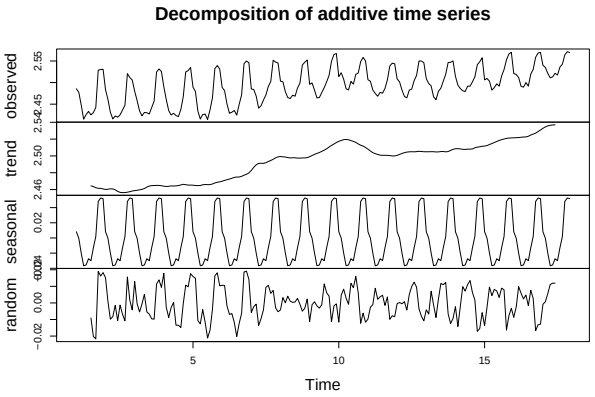


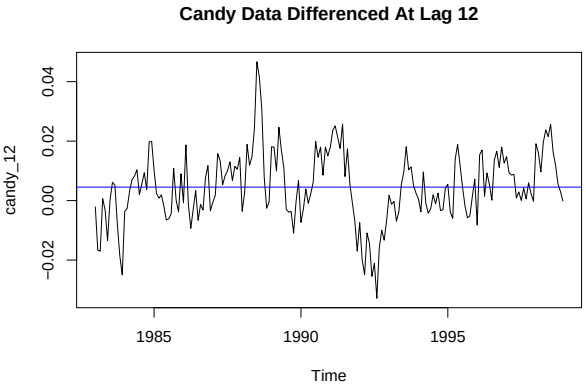
After applying the box-cox transformation, it can be seen from the plot above that the variance of the data is more stable and the transformed data also gives a more symmetric histogram with more evenly spread variance. Comparing to the variance before box-cox transformation 278.34, the variance after is 0.0017 and is close to 0, which suggest it as the stable variance and the transformation is needed.

2) Differencing At Lag 12

```
## [1] "Variance of candy data before difference at lag 12 is 0.00174927057442286"
```

```
## [1] "Variance of candy data after difference at lag 12 is 0.000139767124383988"
```





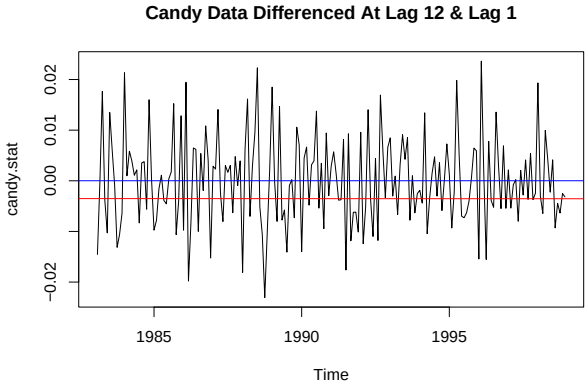
Applying a decomposition on the time series after box-cox transformation, I see that there is still seasonality and trend in this data. As a result, it needed to be further differenced. I firstly utilize the ***differencing at lag 12*** to remove the seasonality. After the differencing, it is easy to see from the plot that the seasonality is no longer apparent. Double checking with the variances before and after the differencing, which is 0.00174 and 0.00013 respectively, suggests that the variance decreased and the differencing is necessary. However, since there is still trend in the plot, the series is not stationary yet and I am going to further difference it at lag 1.

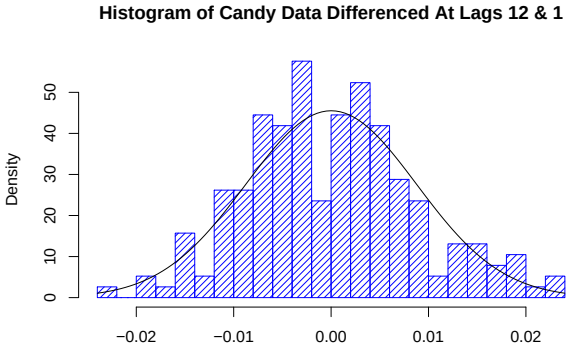
3) Differencing At Lag 1

```
## [1] "Mean of candy data after difference at lag 12 & lag 1 is 1.05526702891196e-05"
```

```
## [1] "Variance of candy data before difference at lag 1 is 0.000139767124383988"
```

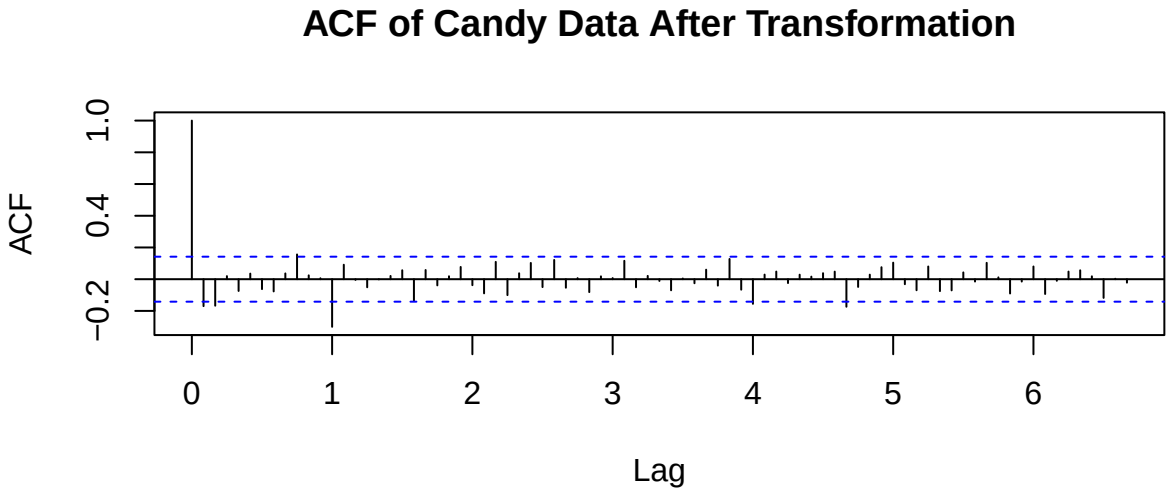
```
## [1] "Variance of candy data after difference at lag 1 is 7.68215842292494e-05"
```

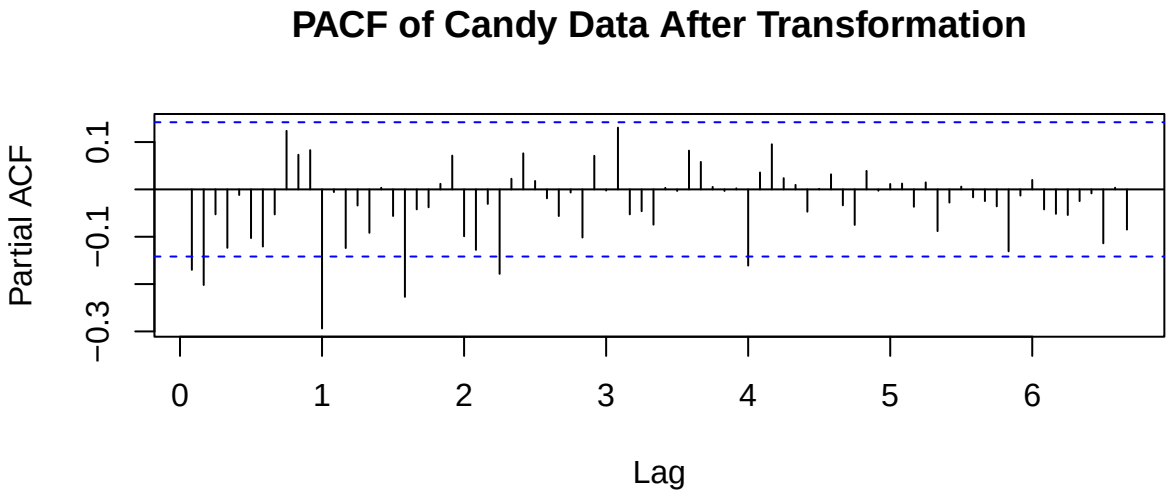




Here, I apply the ***differencing at lag 1*** to remove the trend for the data. After the differencing, it is easy to see from the plot that the trend is not obvious anymore. Double checking with the variances before and after the differencing, which is 0.00013 and 7.682e-05 respectively, suggests that the variance decreased and the differencing is necessary. The mean of the series is 1.055e-05, which is also close to 0. For now, the variance of the data is stable and there is no seasonality or trend. Therefore, the series is stationary and ready to fit SARIMA models.

ACF And PACF Analysis





As the ACF and PACF plots above, I analyze the possible values for p, q, P, and Q. For ACF graph, it goes out of the confidence interval at lag 1, lag 2, lag 9, lag 12, which suggests that nonseasonal part could contains lag 1, lag 2, lag 9 and have one significant spikes for seasonal part. For PACF graph, it goes out of the confidence interval at lag 1, lag 2, lag 12, which suggests that nonseasonal part could contains lag 1, lag 2, lag 12 and have one significant spikes for seasonal part. As a result, the possible values for p, q, P, Q here should be following:

-> **p = 0 or 1 or 2**
-> **q = 0 or 1 or 2 or 9**
-> **P = 0 or 1**
-> **Q = 0 or 1**

And the model should be in the form: **SARIMA(p, 1, q)(P, 1, Q)[12] with d = 1, D = 1, and s = 12**

Model Selection

1) Find Possible Models

```
##    p q P Q      AICc
## 29 1 1 0 1 -1308.743
```

```
##    p q P Q      AICc
## 31 0 2 0 1 -1306.806
```

```
##    p q P Q      AICc
## 33 2 2 0 1 -1307.987
```

For choosing the best model, I would like to compare the AICc scores of the candidate models and choose the ones with the smallest score. From the AICc Score table of the possible models above, I find out that **SARIMA(1, 1, 1)(0, 1, 1)[12]** has the smallest AICc score -1308.743. But in order to get better result, I will also compare with models **SARIMA(2, 1, 2)(0, 1, 1)[12]** with AICc score -1307.987 and **SARIMA(0, 1, 2)(0, 1, 1)[12]** with AICc score -1306.806.

2) Fit models

```
##
## Call:
## arima(x = candy.bc, order = c(1, 1, 1), seasonal = list(order = c(0, 1, 1),
##     period = 12), method = "ML")
##
## Coefficients:
##          ar1      ma1     sma1
##       0.6371  -0.9044  -0.4438
## s.e.  0.0991   0.0606   0.0758
##
## sigma^2 estimated as 5.827e-05:  log likelihood = 658.43,  aic = -1308.86
```

```
## [1] "SARIMA(1, 1, 1)(0, 1, 1)[12] AICc score is -1308.74277930975"
```

```
##
## Call:
## arima(x = candy.bc, order = c(2, 1, 2), seasonal = list(order = c(0, 1, 1),
##     period = 12), method = "ML")
##
## Coefficients:
##           ar1     ar2      ma1      ma2     sma1
##       -0.2048  0.4615  -0.0093  -0.7787  -0.4433
## s.e.   0.1609  0.1554   0.1288   0.1334   0.0779
##
## sigma^2 estimated as 5.72e-05:  log likelihood = 660.14,  aic = -1308.29
```

```
## [1] "SARIMA(2, 1, 2)(0, 1, 1)[12] AICc score is -1307.9867420761"
```

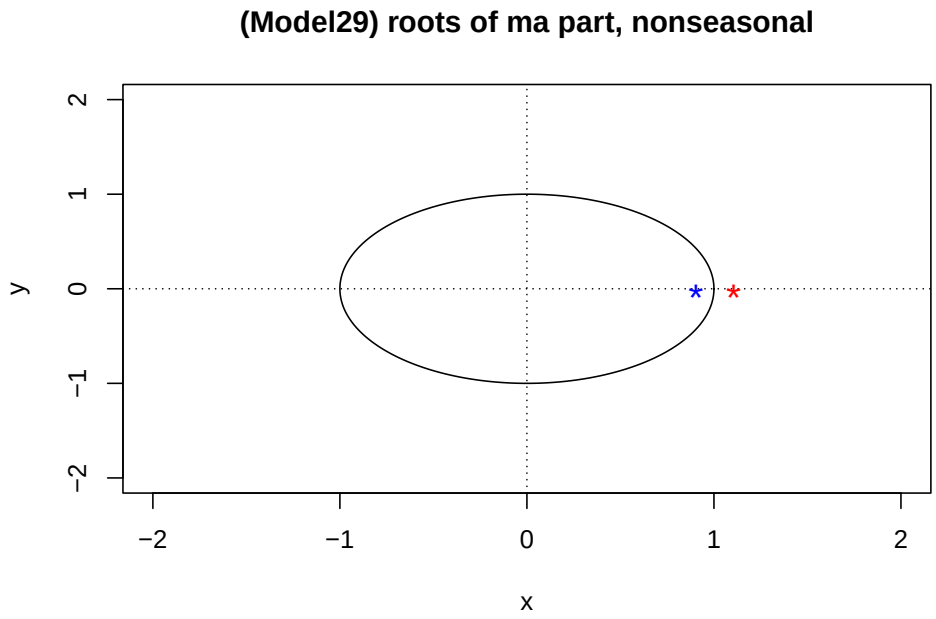
```
##
## Call:
```

```
## arima(x = candy.bc, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 1),
##     period = 12), method = "ML")
##
## Coefficients:
##           ma1      ma2     sma1
##       -0.2335  -0.2886  -0.4476
## s.e.   0.0730   0.0858   0.0718
##
## sigma^2 estimated as 5.9e-05:  log likelihood = 657.46,  aic = -1306.93

## [1] "SARIMA(0, 1, 2)(0, 1, 1)[12] AICc score is -1306.80632415714"
```

After fitting the three models separately, I choose to use model29, which is **SARIMA(1, 1, 1)(0, 1, 1)[12]**. Because it not only has the smallest AICc but also has a simpler form than the other two models. And this model is also the same as one of the models suggested by ACF/PACF.

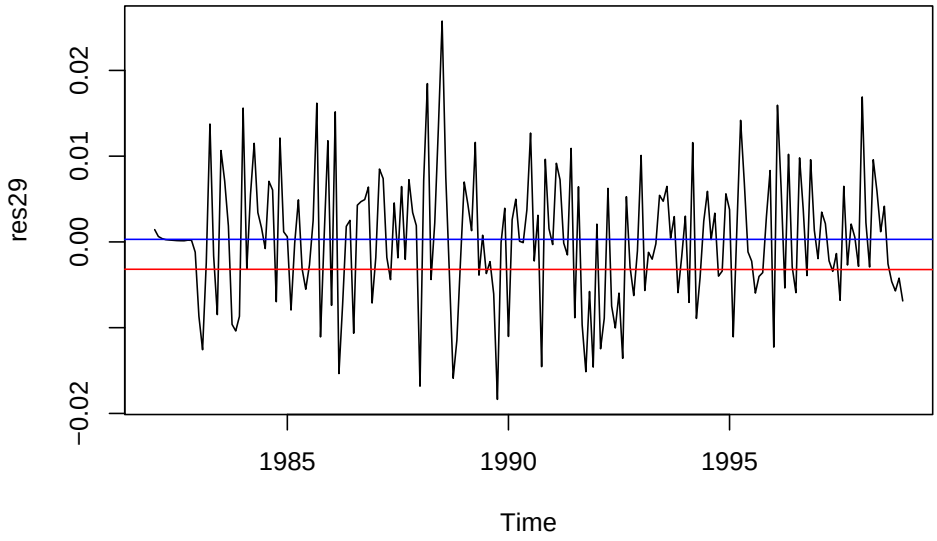
3) Check Invertibility With Unit Circle

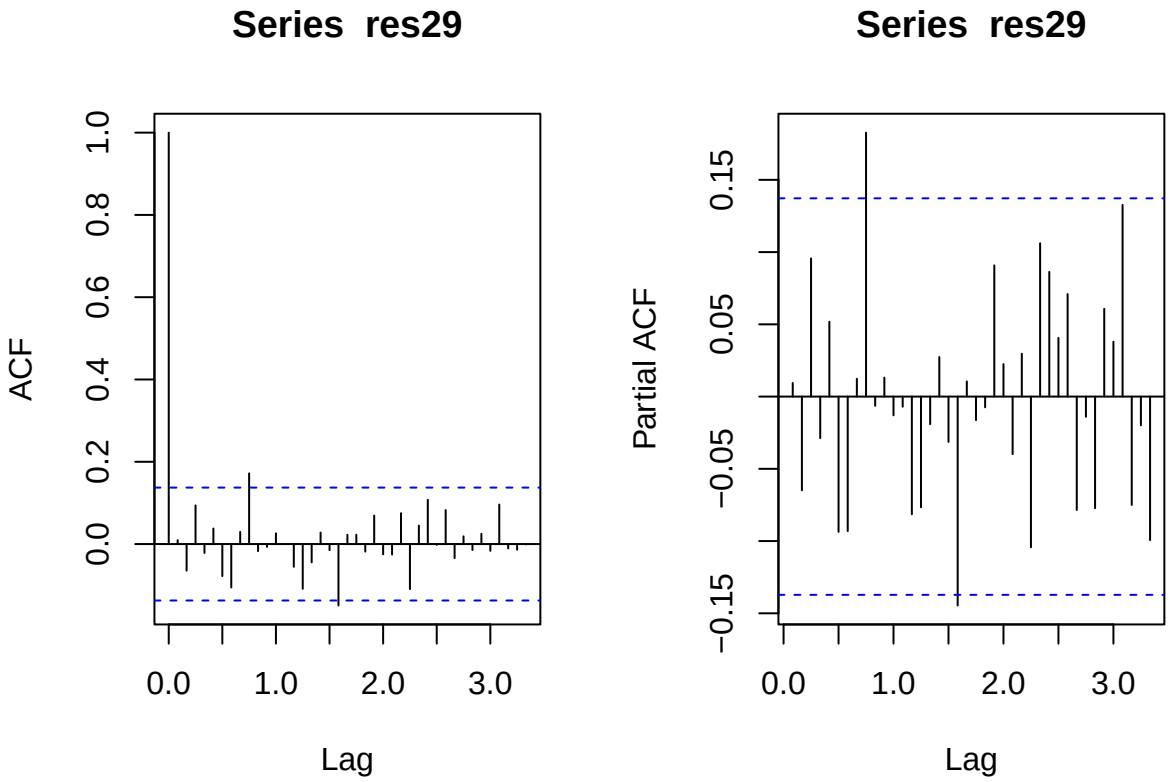


For model29, since the MA part of the model has only one coefficient and is -0.9044, which absolute value is less than one. Also, from the plot above, the root represented by the red star is outside the unit circle. As a result, this model is invertible and can be used to do the diagnostic checking.

4) Diagnostic Checking

4-1: Check the residual time series plot, ACF and PACF

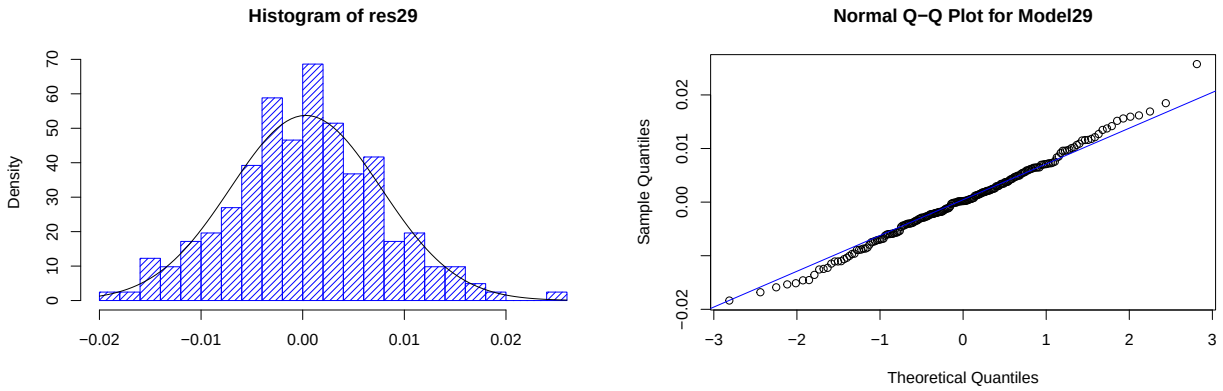




According to the residual plot of the model, it seems to be stationary and similar to white noise with no seasonality or trend and mean close to 0. Also, all ACF and PACF of residuals are roughly within confidence intervals and can be counted as zeros. Therefore, this step of checking passed. Then, I move forward to check the residual normality.

4-2: Check the residual normality with histogram, qqplot and Shapiro test

```
##
##  Shapiro-Wilk normality test
##
## data:  res29
## W = 0.9952, p-value = 0.7674
```



According to the residual histogram and qqplot, it seems to be normal. And it is further proved to be normal since it passes the shapiro-test with p-value equals to 0.7674, which is way larger than 0.05. As a result, the normality checking passed. Then, I will check residual independence.

4-3: Check the residual independence with Box-Pierce test, Ljung-Box test, and Mcleod-Li test

```
##
##  Box-Pierce test
##
## data:  res29
## X-squared = 16.027, df = 12, p-value = 0.19

##
##  Box-Ljung test
##
## data:  res29
## X-squared = 16.889, df = 12, p-value = 0.1538

##
##  Box-Ljung test
##
## data:  res29^2
## X-squared = 24.58, df = 15, p-value = 0.05588
```

According to the results provided above, the p-value for Box-Pierce test is 0.19, which is greater than 0.05; the p-value for Box-Ljung test is 0.1538, which is greater than 0.05; the p-value for McLeod-Li test is 0.05588, which is greater than 0.05. This means that the model pass all three test with all p-value greater than 0.05 and it is valid for further forecasting.

4-4: AR(0) Model Fitting

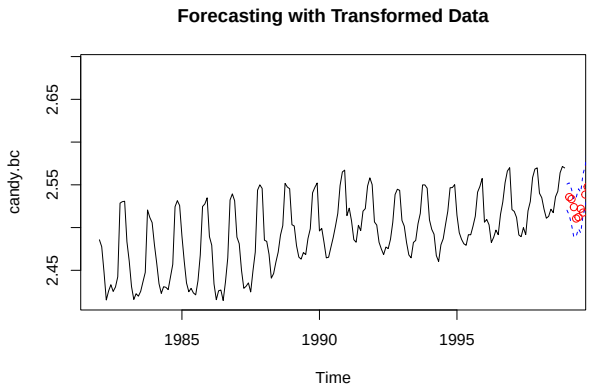
```
##
## Call:
## ar(x = res29, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  5.513e-05
```

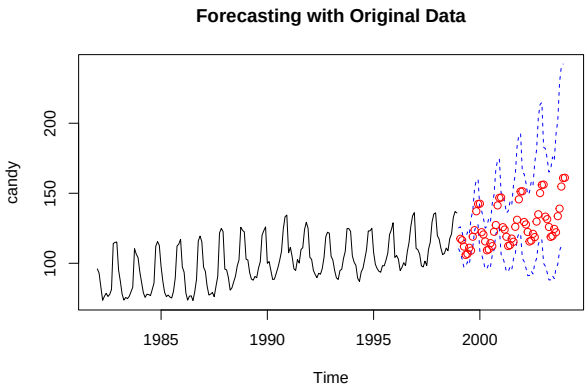
Since the order selected is 0, then the residual fits well in AR(0) model. *For now, the model passed all diagnostic checking steps, which means that the model is satisfactory based on the analysis of residuals. And it is ready to be used for further forecasting.*

5) Final Model Decision & Algebraic Form

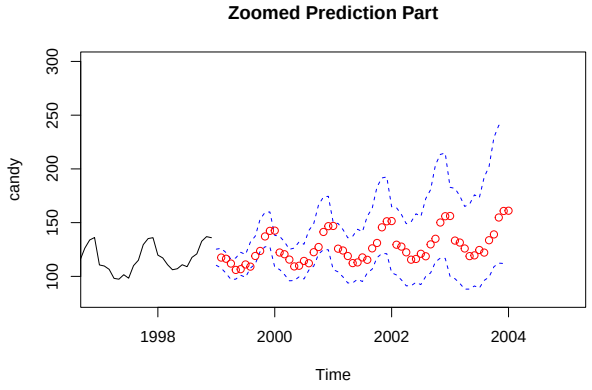
The final model is **SARIMA(1, 1, 1)(0, 1, 1)[12]**.
The fitted model in algebraic form is:
$(1-B)(1-B^{12})(1-0.6371B)X_t = (1-0.9044B)(1-0.4438B^{12})Z_t$
where $Z_t \overset{i.i.d}{\sim} WN(0, 5.827e-05)$

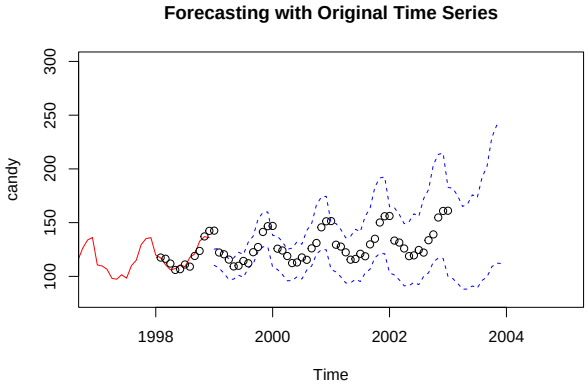
Forecasting and Prediction





By using the model SARIMA(1, 1, 1)(0, 1, 1)[12], I first forecast 60 prediction points on transformed data. Since I used the box-cox transformation for the time series, then I revert the data back to the original one and also apply the 60 prediction points on the original data.





After zoom in the forecasting part from the previous forecasting plot, it is obvious that all the prediction points (black points) are staying in the 95% confidence interval(the blue dashed lines). Also, the test set(the last part of the red line) overlaps with the first few prediction points, which means that the test set is within the prediction intervals. As a result, the prediction is successful and reasonable.

Conclusion

For this project, I tend to explore the candy production development in the US by analyzing the time series data "US Candy Production, 1982 - 1999" and forecasting 60 prediction points ahead with the model **SARIMA(1, 1, 1)(0, 1, 1)[12]** or the algebraic form as:
$(1-B)(1-B^{12})(1-0.6371B)X_t = (1-0.9044B)(1-0.4438B^{12})Z_t$, where $Z_t \overset{i.i.d}{\sim} WN(0, 5.827e-05)$

Based on my analysis and forecasting, the model is well fitted, which passes all the diagnostic checking steps; all my prediction points lie inside the 95% confidence interval and my test data set is also well contained in the prediction intervals. This means that the results of my analysis would be reasonable.

As a result, my final conclusion for the candy production development is that there is a significant seasonal factor influence the candy production in the US to have a strong seasonality and the forecasting suggests a possible increasing trend for it in the following years, which means that the high demanding of candy would enable the candy production in the US to increase.

Finally, I would like to say thank you to **Prof. Raya Feldman** and **Sunpeng Duan**, who gave me a lot of helps on my project with constructive suggestions.

References

1. https://www.kaggle.com/rtatman/us-candy-production-by-month
2. Prof. Raya Feldmen "PSTAT 174 Winter2022 Lecture slides, Labs"
3. Special helps from Prof. Raya Feldman and Sunpeng Duan

Code Appendix

```
knitr::opts_chunk$set(echo = F,
                      results = 'hide',
                      message = F,
                      warning = F)
library(ggplot2)
library(MASS)
library(ggfortify)
library(forecast)
library(qpcR)
library(tseries)
# Read in the data set file of 'candy_production1999.csv'
candy_data <- read.table(file = 'candy_production1999.csv', sep = ',', header = TRUE)

plot.ts(candy_data[,2], ylab = 'candy_data', main = 'Plot of Raw Data With Trend & Mean')
# add trend to data plot
nt = length(candy_data[,2])
fit <- lm(candy_data[,2] ~ as.numeric(1:nt)); abline(fit, col = 'red')
# add mean to data plot
abline(h = mean(candy_data[,2]), col = 'blue')
# Create a training data set
candy.traning <- candy_data[c(1:204),2]

# Create a test data set
candy.test <- candy_data[c(205:216),2]

# Create a time series data
candy <- ts(candy.traning, start = c(1982,1), frequency = 12)
# plot histogram of time series
hist(candy, col = 'light blue',
            xlab = '',
            main = 'Histogram of Candy Production Data')

# plot acf of time series
acf(candy, lag.max = 80, main = 'ACF of Candy Production Data')
# Box-Cox transformation for data
## plot the graph
bcTransform <- boxcox(candy ~ as.numeric(1:length(candy)),
                      lambda = seq(-2,2, 0.01))

#Give the value of $lambda$ = -0.3
lambda <- bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
paste('The value of lambda is', lambda)
# Perform box-cox transformation
candy.bc = (1/lambda)*(candy^lambda-1)

# Plot transfromed data, histogram
plot.ts(candy.bc, main = 'Plot after Box-Cox(candy)')
hist(candy.bc, col = 'light blue', xlab = '',
     main = 'Histogram of Box-Cox(candy)')

# Check variance to see if it is more stable
```

```r
paste('Variance of candy data before transformation is', var(candy))
paste('Variance of candy data after transformation is', var(candy.bc))
# Decomposition of raw data
y <- ts(as.ts(candy.bc), frequency = 12)
decomp <- decompose(y); plot(decomp)

# Remove seasonality with differencing at lag 12 and plot
candy_12 <- diff(candy.bc, lag = 12)
plot.ts(candy_12, main = 'Candy Data Differenced At Lag 12')
fit <- lm(candy_12 ~ as.numeric(1:length(candy_12))); abline(fit, col="red")
abline(h=mean(candy_12), col="blue")

# check variance to see if there is overdifferencing
paste('Variance of candy data before difference at lag 12 is', var(candy.bc))
paste('Variance of candy data after difference at lag 12 is', var(candy_12))
# Remove trend with differencing at lag1 and plot
candy.stat <- diff(candy_12, lag=1)
plot.ts(candy.stat, main="Candy Data Differenced At Lag 12 & Lag 1")
fit <- lm(candy.stat ~ as.numeric(1:length(candy.stat))); abline(fit, col="red")
abline(h=mean(candy.stat), col="blue")

hist(candy.stat, density=20,breaks=20, col="blue", xlab="", prob=TRUE,
     main = 'Histogram of Candy Data Differenced At Lags 12 & 1' )
m<-mean(candy.stat); std<- sqrt(var(candy.stat))
curve( dnorm(x,m,std), add=TRUE )

# check mean & variance to see if there is overdifferencing
paste('Mean of candy data after difference at lag 12 & lag 1 is', mean(candy.stat))
paste('Variance of candy data before difference at lag 1 is', var(candy_12))
paste('Variance of candy data after difference at lag 1 is', var(candy.stat))
# Plot ACF
acf(candy.stat, lag.max=80, main="ACF of Candy Data After Transformation")
pacf(candy.stat, lag.max=80, main="PACF of Candy Data After Transformation")
df <- expand.grid(p=0:2, q=c(0:2,9), P=0:1, Q=0:1)
df <- cbind(df, AICc=NA)

for (i in 1:nrow(df)) {
  sarima.obj <- NULL
  try(arima.obj <- arima(candy.bc, order=c(df$p[i], 1, df$q[i]),
  seasonal=list(order=c(df$P[i], 1, df$Q[i]), period=12),
  method="ML"))
  if (!is.null(arima.obj)) { df$AICc[i] <- AICc(arima.obj) }
  #print(df[i, ])
}

df[which.min(df$AICc), ] # model29: SARIMA(1, 1, 1)(0, 1, 1)[12]
df[31, ] # model31: SARIMA(0, 1, 2)(0, 1, 1)[12]
df[33, ] # model33: SARIMA(2, 1, 2)(0, 1, 1)[12]
# model29: SARIMA(1, 1, 1)(0, 1, 1)[12]
model29 <- arima(candy.bc, order=c(1,1,1),
                 seasonal = list(order = c(0,1,1), period = 12), method="ML")
model29
```

```r
paste('SARIMA(1, 1, 1)(0, 1, 1)[12] AICc score is',AICc(model29))
# model33: SARIMA(2, 1, 2)(0, 1, 1)[12]
model33 <- arima(candy.bc, order=c(2,1,2),
                 seasonal = list(order = c(0,1,1), period = 12), method="ML")
model33

paste('SARIMA(2, 1, 2)(0, 1, 1)[12] AICc score is',AICc(model33))
# model31: SARIMA(0, 1, 2)(0, 1, 1)[12]
model31 <- arima(candy.bc, order=c(0,1,2),
                 seasonal = list(order = c(0,1,1), period = 12), method="ML")
model31

paste('SARIMA(0, 1, 2)(0, 1, 1)[12] AICc score is',AICc(model31))
source('plot.roots.R.txt')
# Check invertibility for MA part of model29
plot.roots(NULL,polyroot(c(1,-0.9044)), main="(Model29) roots of ma part, nonseasonal")
# Diagonostic Checking
res29 <- residuals(model29)
# Check the residual time series plot
plot.ts(res29)
fitt29 <- lm(res29 ~ as.numeric(1:length(res29))); abline(fitt29, col="red")
abline(h=mean(res29), col="blue")
# Check the residual ACF and PACF
op = par(mfrow = c(1,2))
acf(res29, lag.max=40)
pacf(res29, lag.max=40)
# Check the residual normality with histogram, qqplot and Shapiro test
hist(res29,density=20,breaks=20, col="blue", xlab="", prob=TRUE)
m <- mean(res29)
std <- sqrt(var(res29))
curve( dnorm(x,m,std), add=TRUE )

qqnorm(res29,main= "Normal Q-Q Plot for Model29")
qqline(res29,col="blue")

shapiro.test(res29)
# Check the residual independence
Box.test(res29, lag = 15, type = c("Box-Pierce"), fitdf = 3)
Box.test(res29, lag = 15, type = c("Ljung-Box"), fitdf = 3)
Box.test(res29^2, lag = 15, type = c("Ljung-Box"), fitdf = 0)
# Check the characteristic root
ar(res29, aic = TRUE, order.max = NULL, method = c("yule-walker"))
#install.packages("forecast")
library(forecast)
forecast(model29)
# Predict with the fitted model and data after transformation
pred.tr <- predict(model29, n.ahead = 60)

U.tr = pred.tr$pred + 2*pred.tr$se
L.tr = pred.tr$pred - 2*pred.tr$se

ts.plot(candy.bc, xlim=c(1982,1999), ylim = c(min(candy.bc), max(U.tr)),
        main = 'Forecasting with Transformed Data')
```

```
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points(1999+(1:60)/12, pred.tr$pred, col="red")


# Use the original data to do the prediction again
invbox = function(d,lambda){
  return(exp(log(d * lambda + 1)/lambda))
}
pred.orig <- invbox(pred.tr$pred, lambda)

U = invbox(U.tr, lambda)
L = invbox(L.tr, lambda)

ts.plot(candy, xlim=c(1982,2004), ylim = c(min(candy),max(U)),
        main = 'Forecasting with Original Data')
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points(1999+(1:60)/12, pred.orig, col="red")


# Zoom in the prediction part
ts.plot(candy, xlim = c(1997,2005), ylim = c(80, 300),
        main = 'Zoomed Prediction Part')
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points(1999+(1:60)/12, pred.orig, col="red")

# plot the prediction part with line and prediction points
ts.plot(candy, xlim = c(1997,2005), ylim = c(80, 300), col="red",
        main = 'Forecasting with Original Time Series')
lines(U, col="blue", lty="dashed")
lines(L, col="blue", lty="dashed")
points(1998+(1:60)/12, pred.orig, col="black")
```