PSTAT174 Final Project - Theo Lee

1. Abstract

The objective of this project was to utilize key time series techniques in a real-world application: CO2 data collected from the Mauna Loa Observatory in Hawaii. To forecast this CO2 data, we broke the data into training and test sets. Working on the training set, we first aimed to make it stationary and identify a model. We did this by performing a Box-Cox transformation and then taking a difference at 12 & 1 to eliminate seasonality and trend respectively. After a model was identified from several candidates via diagnostics, forecasting was performed using the training set to predict CO2 levels. It was then compared to the true levels from the test set. We found that our model, SARIMA$(1,1,1)x(0,1,1)_{12}$, plotted similarly to the true levels in addition to the true levels largely falling between the 95% confidence intervals appropriately.

2. Introduction

Global warming is one of the most prevalent threats to our planet. Normally, sunlight should reflect off of the earth's surface and back into space when it strikes the earth. But due to an excess of gasses in the earth's atmosphere, these rays and radiation from the sun end up trapped, thus generating heat on the surface of the earth. Carbon dioxide, in particular, is one of the greenhouse gasses that is often deemed most responsible for trapping heat.

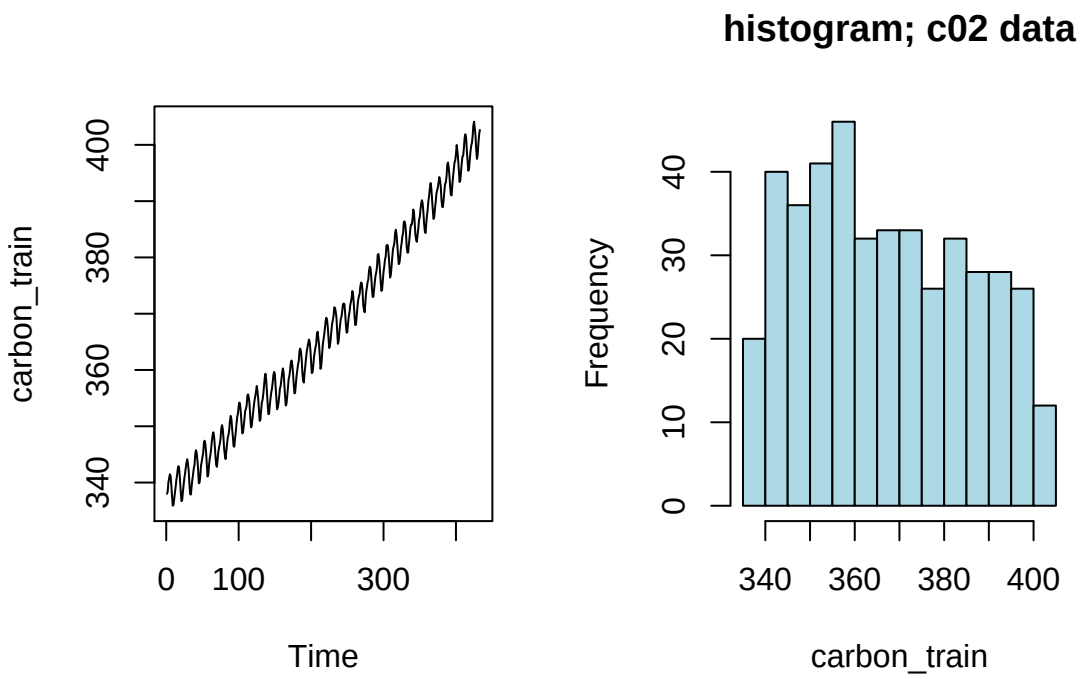
This project utilized a dataset that was sourced from Kaggle and was collected and published by the University of California's Scripps Institution of Oceanography. It contains monthly observations of C02 concentrations from the Mauna Loa Observatory in Hawaii. This dataset was chosen because this laboratory has the world's longest record of unbroken C02 observations, thus ensuring that missing values would not be a consideration. However, due to a spike of outlier values in earlier years, the dataset was cleaned to reflect only the years 1980-2017. The columns containing the date and monthly C02 concentrations in PPM were kept from the original dataset. Google Sheets was used to clean the original dataset and store it as a .csv file.

The aim of this project was to develop a model from the dataset for forecasting future CO2 levels. Forecasting future levels may prove crucial in setting appropriate targets for countries to meet if we wish to curb global warming before it is too late. We started by reading in our data and splitting it into training and test sets. A Box-Cox Transformation was performed to stabilize variance and to make the data appear (slightly) more Gaussian. Analyzing the decomposition of the transformed data depicted clear seasonality and a nearly linear trend, so we differenced at 12 and 1, thus making our data stationary. Possible models were generated by analyzing the ACF/PACF of stationary data and then selected depending on performance in diagnostics. Moving to forecasting, our final model closely matched the true values from the test set with the test values largely falling within the 95% confidence intervals. This suggests that we created a model that, for the most part, forecasts future global warming levels - at least within the context of the data we used. R Studio was used for all computations and analysis.

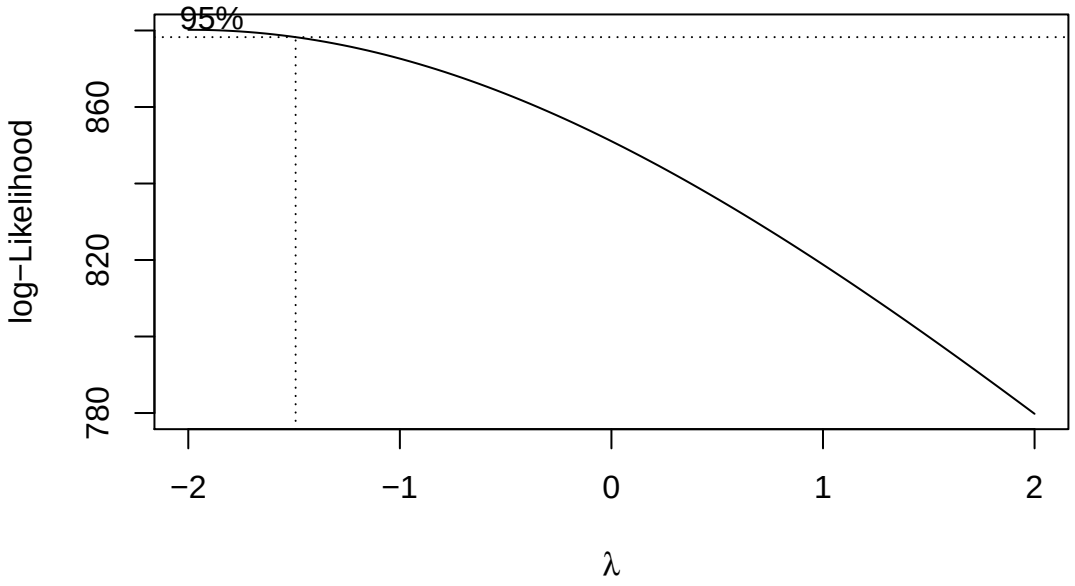
3. Plotting and Analyzing the Time Series

First, we plot the original data to confirm non-stationarity. We see from the plot that the data follows a positive linear trend. There are visible, consistent spikes and dips to this upwards trend, which further

suggests that there is a seasonal component to this data. There are no apparent sharp changes to the behavior of the data. Examination of the histogram reveals no signs of skewed tailing; however, it is clearly non-Gaussian in nature.



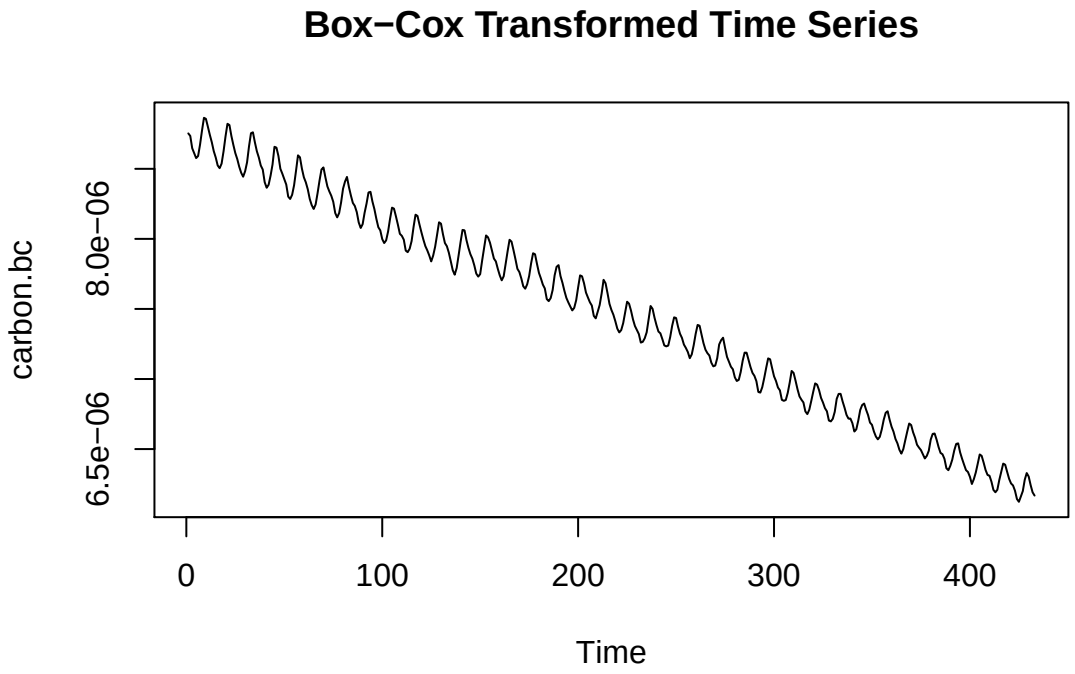
In an effort to stabilize variance which is currently 341 and to make our histogram appear more Gaussian, we perform the Box-Cox transformation.

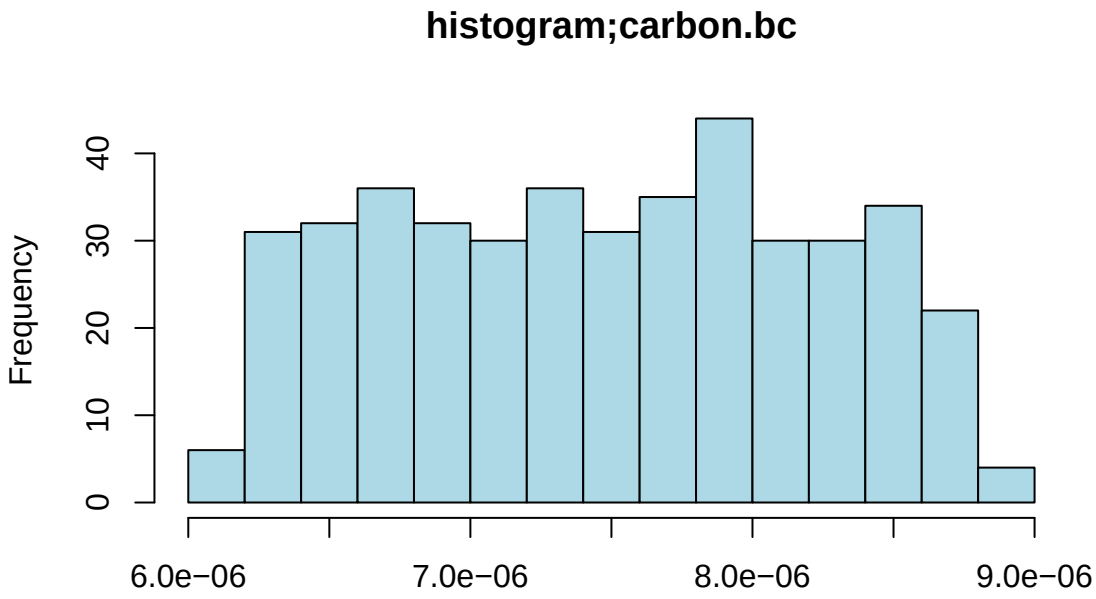


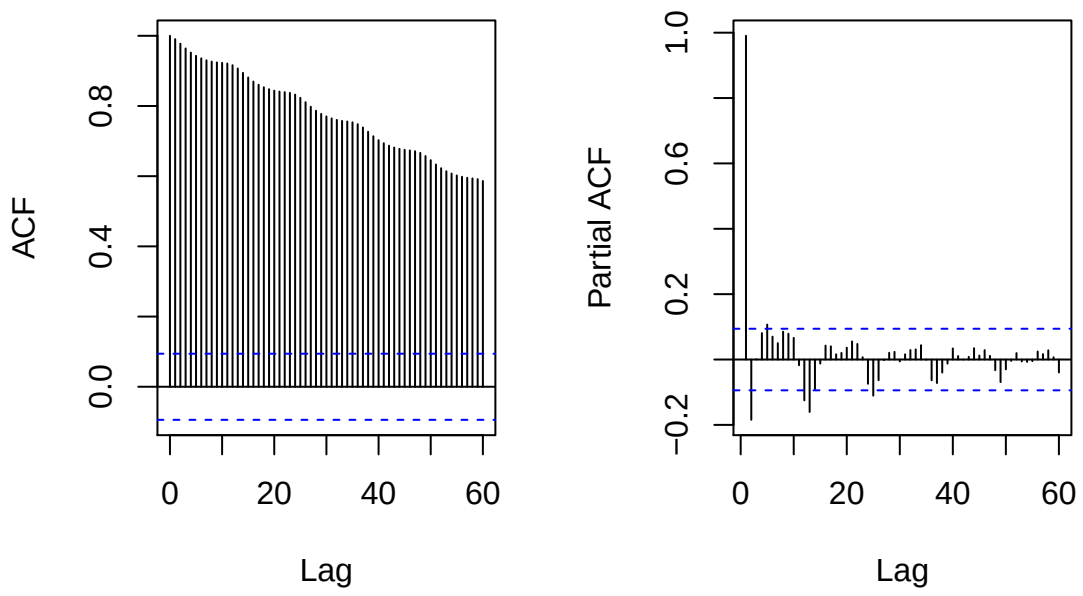

```
## [1] "lambda"
```

```
## [1] -2
```

Lambda is calculated as -2, so we perform the transformation 1/Yˆ2 on the training set which results in the following:


Box−Cox Transformed Time Series
carbon.bc
6.5e−06
8.0e−06
0
100
200
300
400
Time


histogram;carbon.bc
Frequency
0
10
20
30
40
6.0e−06
7.0e−06
8.0e−06
9.0e−06



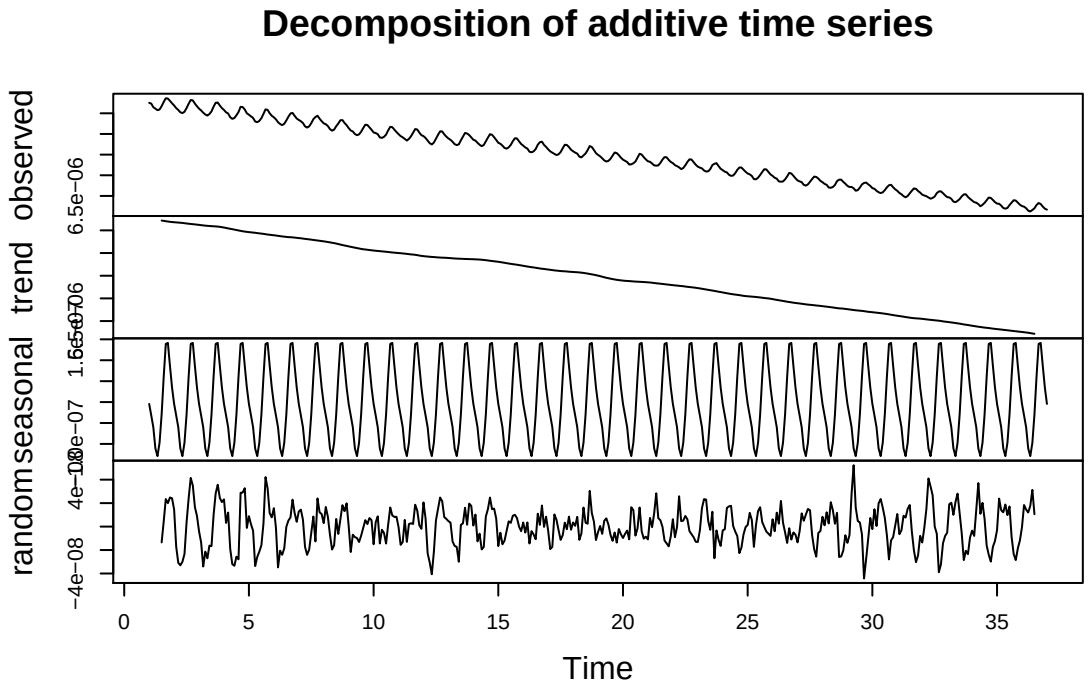

```
## [1] "variance before"
```

```
## [1] 341.8832
```

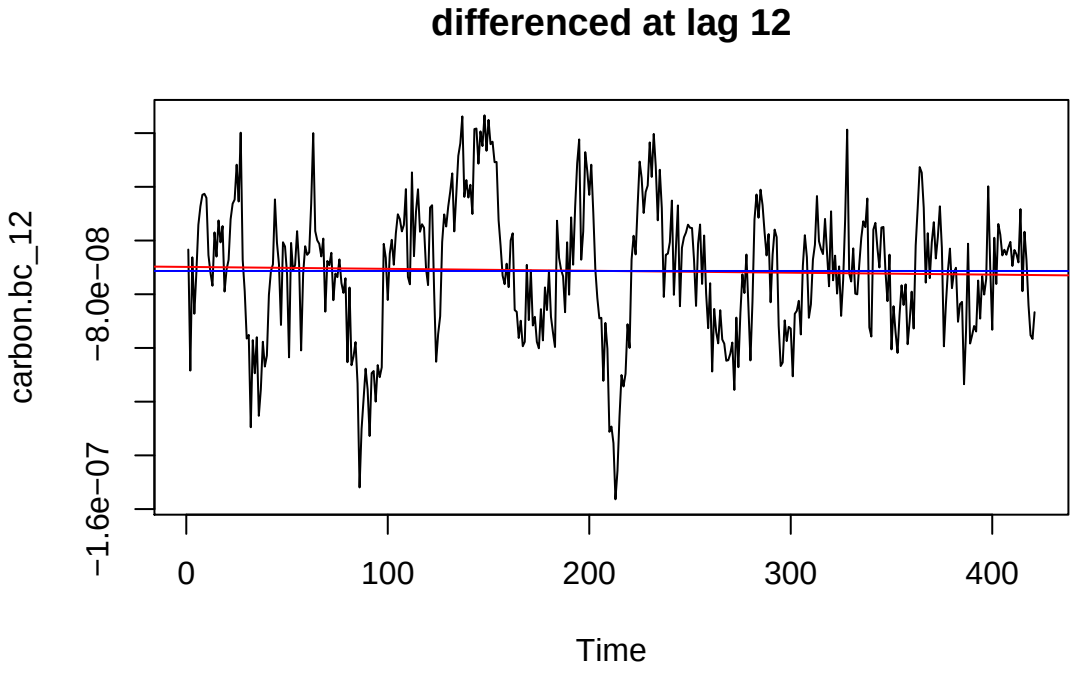
```
## [1] "variance after"
```

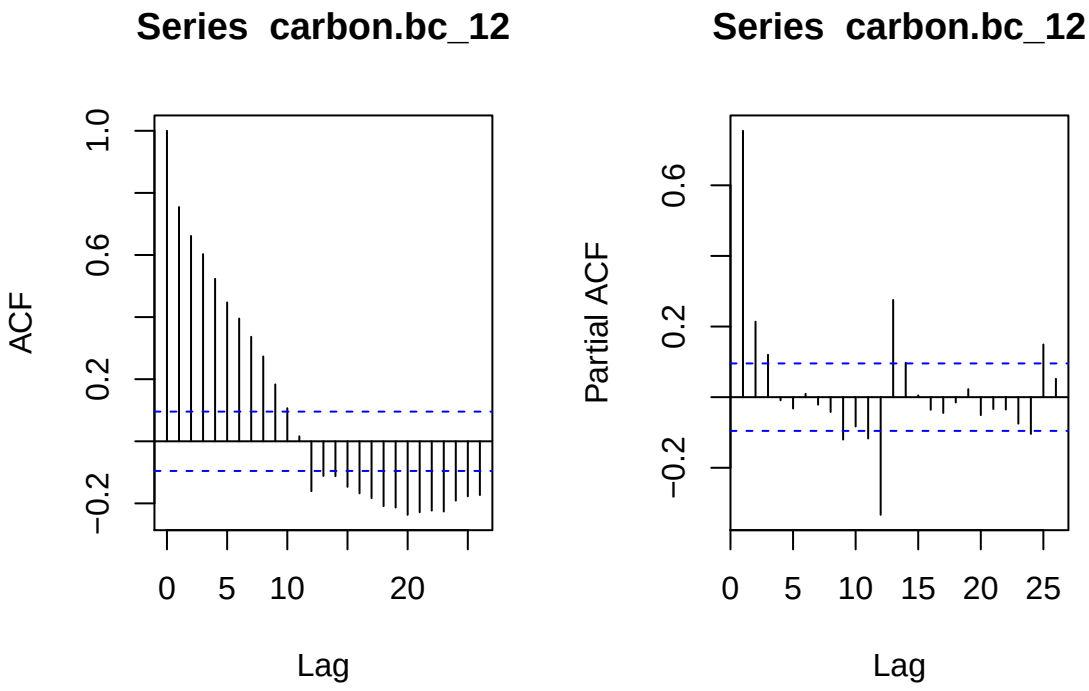
```
## [1] 5.544553e-13
```

Examining the ACF of the Box-Cox transformed data depicts slow decay which indicates non-stationarity and seasonality. The histogram is still clearly non-Gaussian even though it does look slightly better than the original data. Producing the decomposition for the Box-Cox transformed data reinforces the notion that we should expect to take differences to achieve stationarity: the trend line is roughly linear which suggests a difference of 1 to eliminate trend, and there is a clear seasonal pattern which points to a difference of 12 to eliminate seasonality.

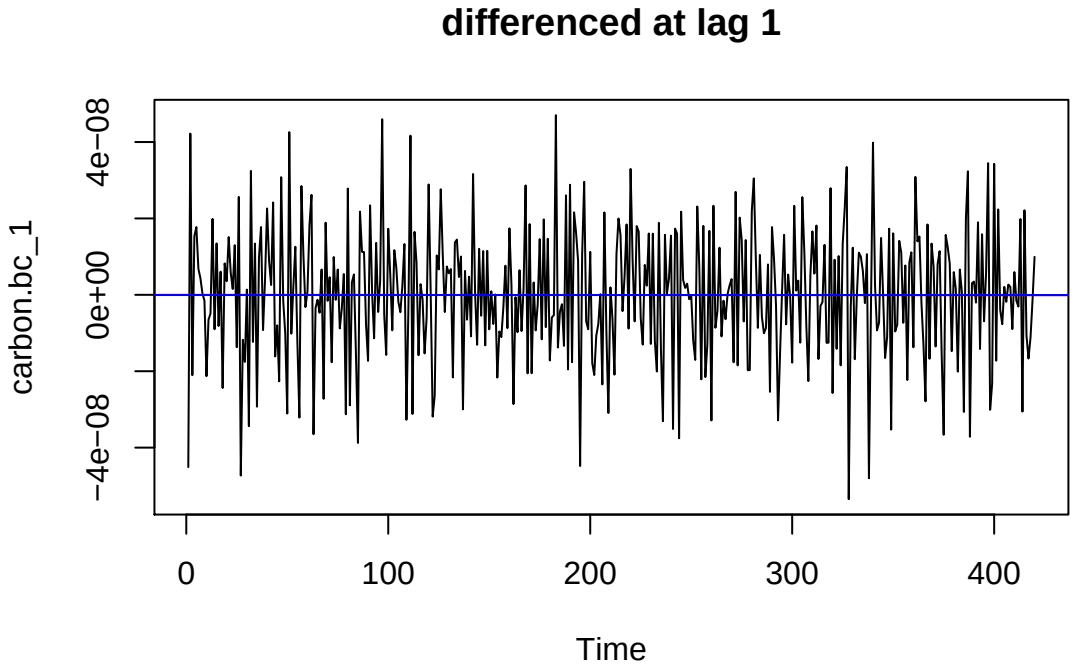


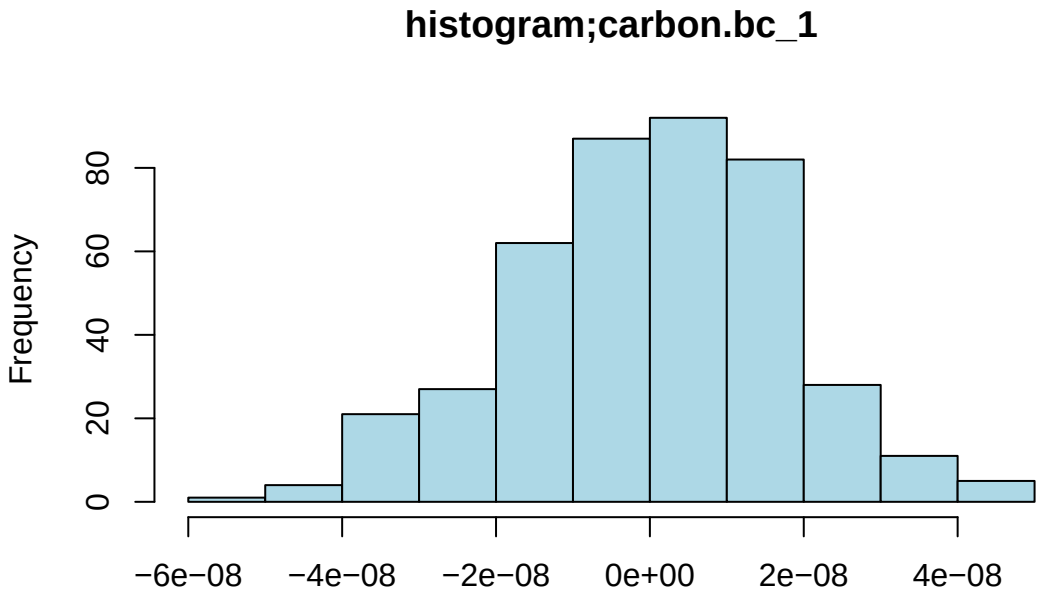
Proceeding to differencing at lag 12 yields the following below. ACF still exhibits slow decay which depicts non-stationarity; however, there is no longer a visible seasonal component.

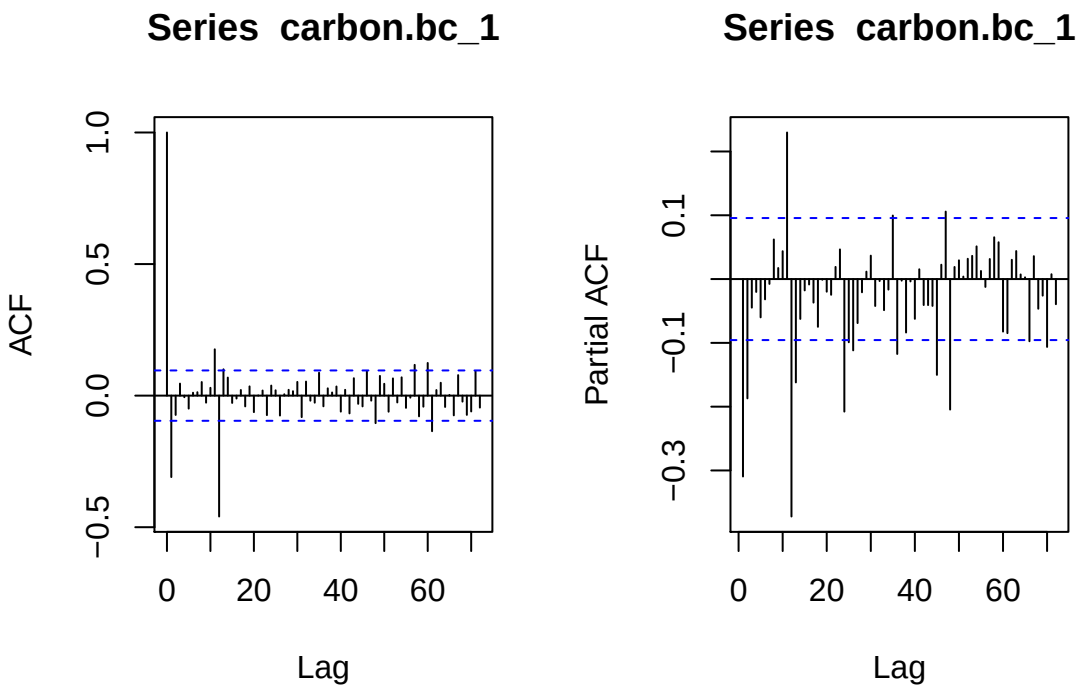




Next, we difference at 1 to eliminate the trend which yields the following below. We see that the plot is no longer dependent on either the mean or the variance. The histogram is roughly Gaussian. The ACF and PACF are also stationary.







We consider taking another difference, verifying this by checking the variances before and after. We see that variance *increases* so we choose to stop at one difference. This also indicates that we can proceed to model identification.

```
## [1] "Variance before another difference:"
```

```
## [1] 3.046744e-16
```

```
## [1] "Variance after another difference:"
```

```
## [1] 7.947347e-16
```

4. Model Identification

We took a difference to eliminate seasonality and another difference to eliminate trend; therefore, we know that D=1 and d=1. Similarly, we know that this is monthly data, so s=12. Looking at the ACF above, we see that the last lag to break past the confidence interval boundaries is at lag 12, which suggests that Q=1. We observe one lag past the CI boundaries inside of lag 12 (lag 11 is an artifact from lag 12), so we take this to be q=0 or q=1. Examination of the PACF shows that lags taper off of lag 12 and cut off abruptly from lag 48. We can take this to be P=0. For these possibilities, we have p=0-4 respectively. We search for possible values of p and q, generating the three models with the lowest AICc scores:

```
##      p q      AICc
## [1,] 3 1 -14038.96
## [2,] 4 1 -14037.14
## [3,] 0 1 -14036.11
```

Fitting these values for p and q into our established d=1,P=0, D=1, and Q=1 results in the following models:

Model A: SARIMA $(3,1,1)x(0,1,1)_{12}$

$(1-0.6323_{0.0506}B-0.1535_{0.0578}B^2-0.1275_{0.0498}B^3)Y_t=(1-1_{0.0095}B^)(1-0.7367_{0.0347}B^{12})Z_t$

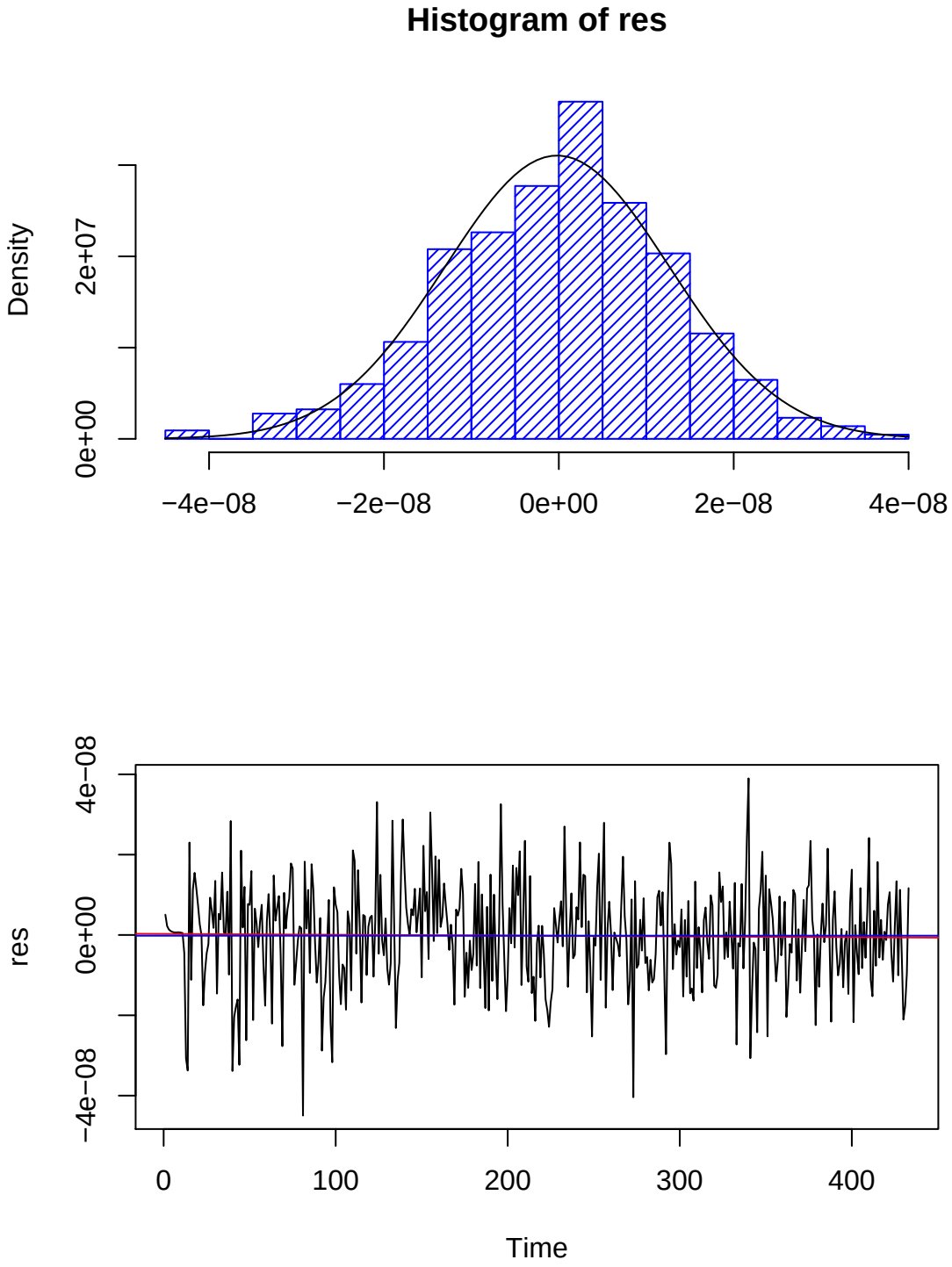
Model B: SARIMA $(4,1,1)x(0,1,1)_{12}$

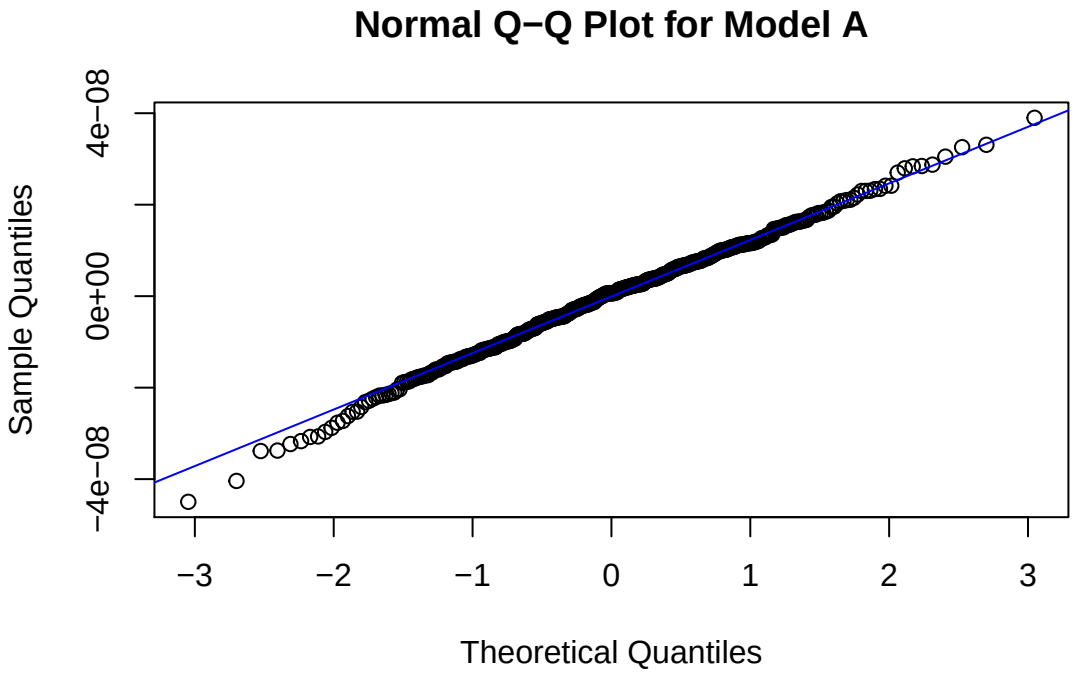
$(1-0.6281_{0.0506}B-0.1501_{0.0581}B^2-0.1116_{0.0591}B^3-0.0250_{0.0496}B^4)Y_t=(1-1_{0.0094}B^)(1-0.7356_{0.0346}B^{12})Z_t$

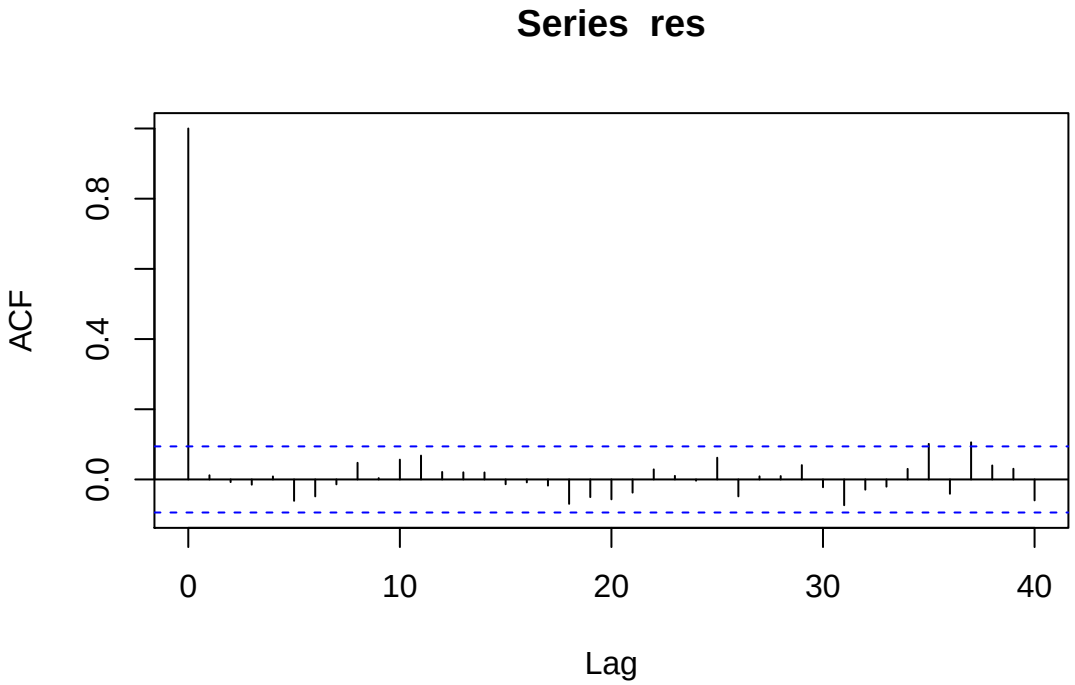
Model C: SARIMA $(1,1,1)x(0,1,1)_{12}$

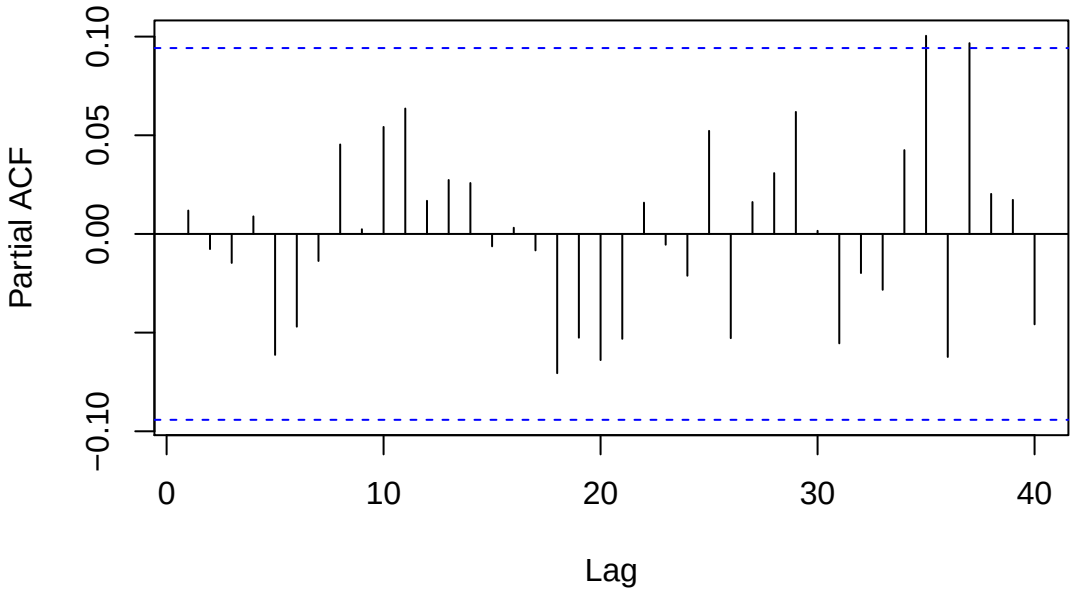
$(1-0.2095_{0.1524}B)Y_t=(1-0.5504_{0.1312}B)(1-0.7441_{0.0332}B^{12})Z_t$

Model A Diagnostics:

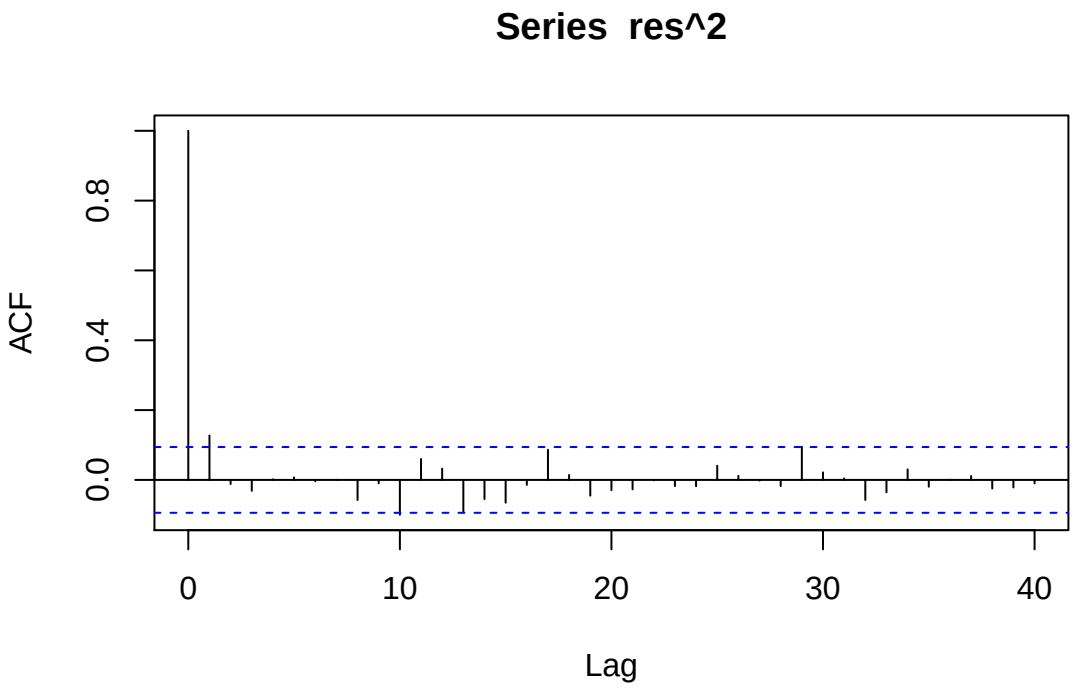

Histogram of res
Density
0e+00
2e+07
−4e−08
−2e−08
0e+00
2e−08
4e−08
res
4e−08
0e+00
−4e−08
0
100
200
300
400
Time


Normal Q−Q Plot for Model A
Sample Quantiles
Theoretical Quantiles
4e−08
0e+00
−4e−08
−3
−2
−1
0
1
2
3


Series res
ACF
Lag
0.0
0.4
0.8
0
10
20
30
40

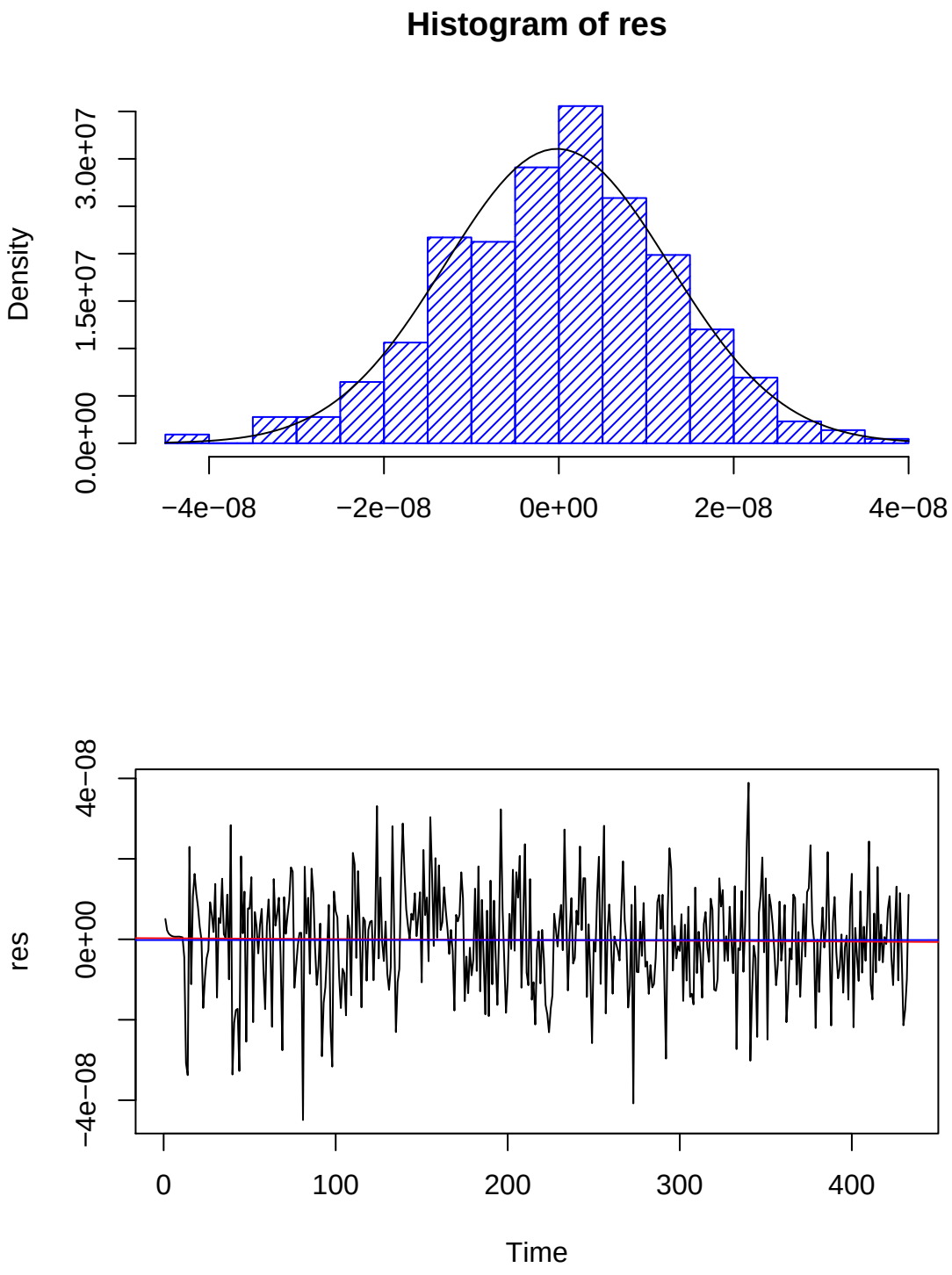


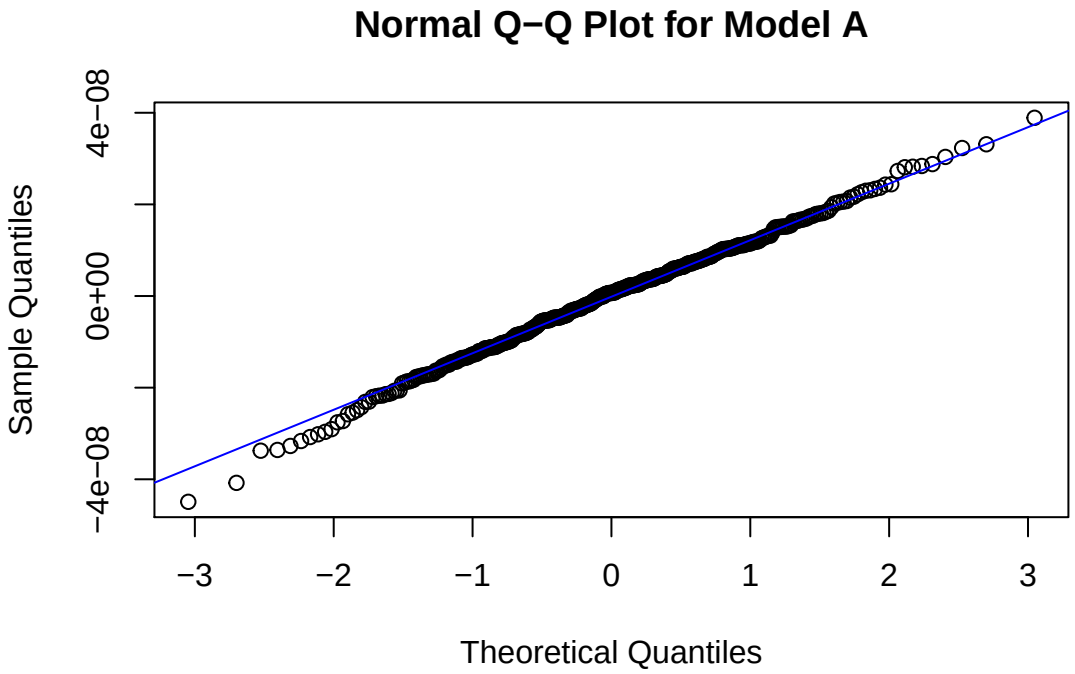

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.99632, p-value = 0.4205

##
##  Box-Pierce test
##
## data:  res
## X-squared = 7.4624, df = 7, p-value = 0.3824

##
##  Box-Ljung test
##
## data:  res
## X-squared = 7.6418, df = 7, p-value = 0.3652

##
##  Box-Ljung test
##
## data:  res^2
## X-squared = 15.632, df = 12, p-value = 0.2087
```

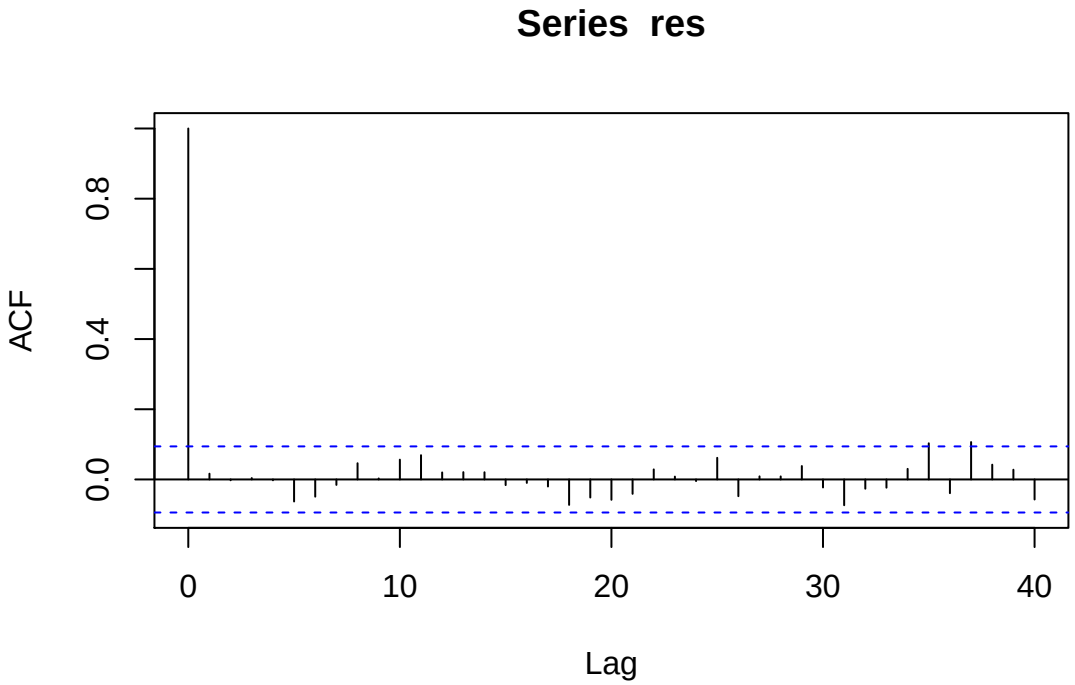


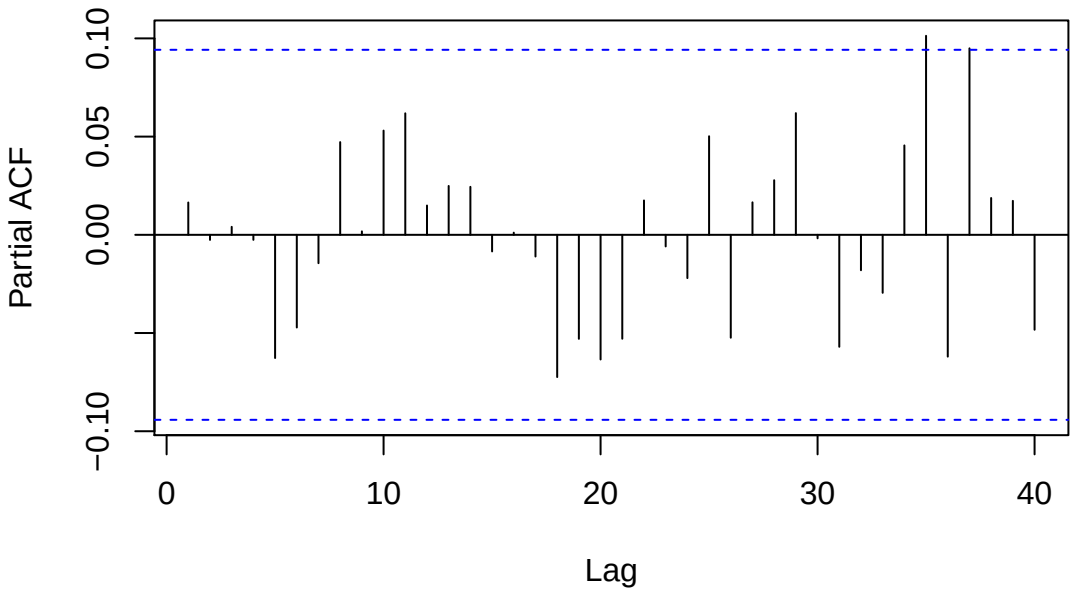

```
##
## Call:
## ar(x = res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.651e-16
```

Model B diagnostics:

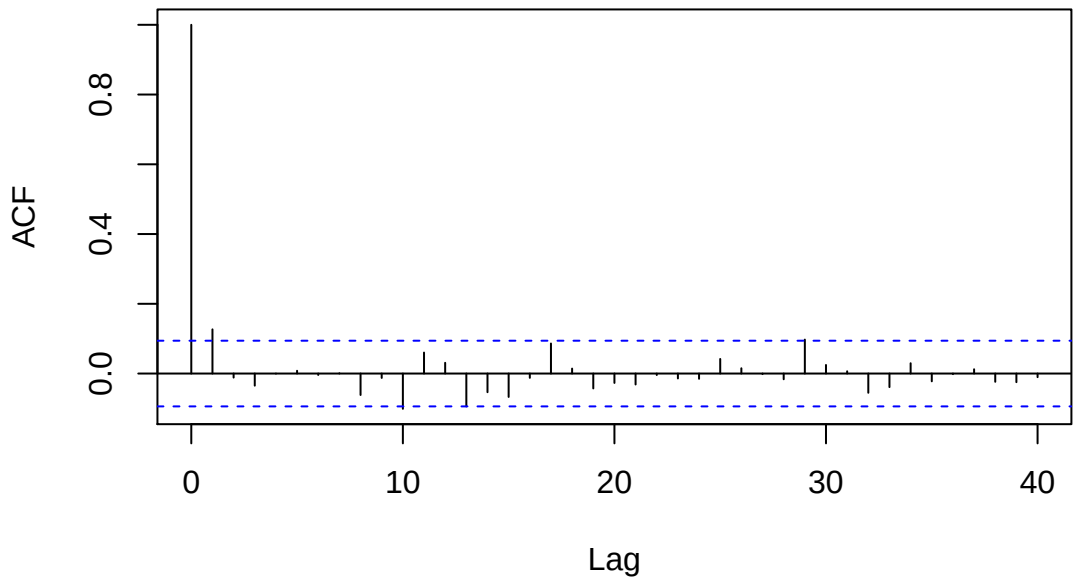

Histogram of res
Density
0.0e+00
1.5e+07
3.0e+07
−4e−08
−2e−08
0e+00
2e−08
4e−08
res
−4e−08
0e+00
4e−08
0
100
200
300
400
Time


Normal Q−Q Plot for Model A
Sample Quantiles
Theoretical Quantiles
4e−08
0e+00
−4e−08
−3
−2
−1
0
1
2
3

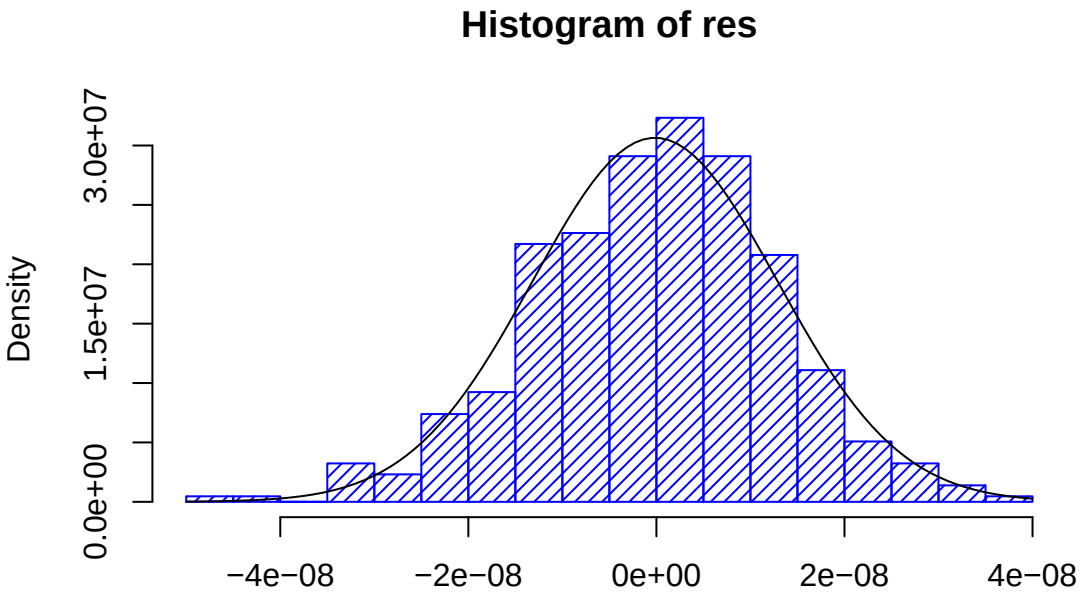

Series res
ACF
Lag
0.0
0.4
0.8
0
10
20
30
40



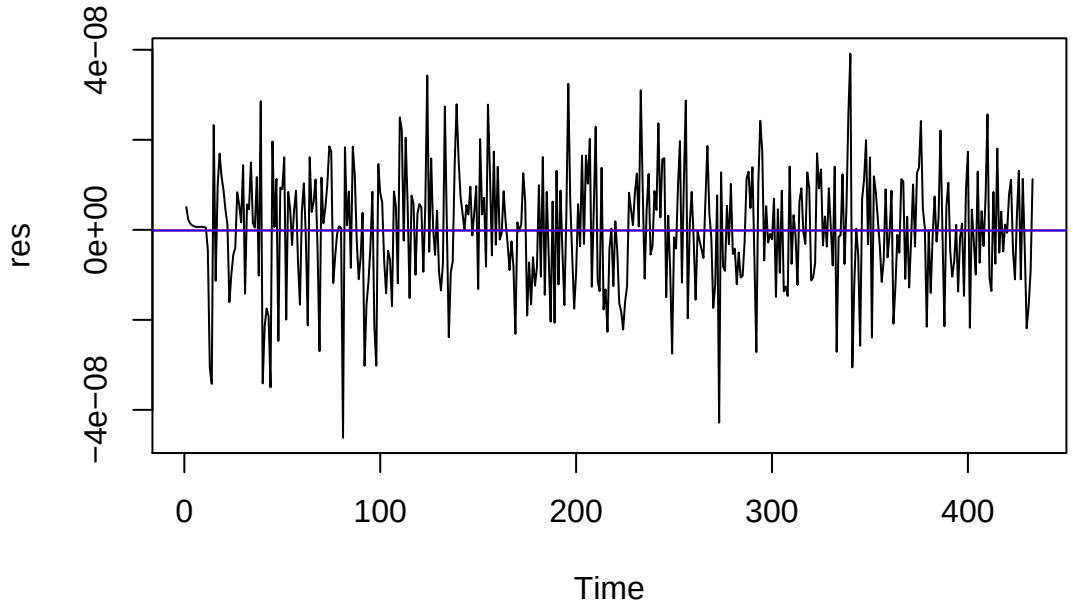

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.99624, p-value = 0.4013

##
##  Box-Pierce test
##
## data:  res
## X-squared = 7.5346, df = 6, p-value = 0.2742

##
##  Box-Ljung test
##
## data:  res
## X-squared = 7.7158, df = 6, p-value = 0.2597

##
##  Box-Ljung test
##
## data:  res^2
## X-squared = 15.975, df = 12, p-value = 0.1924
```

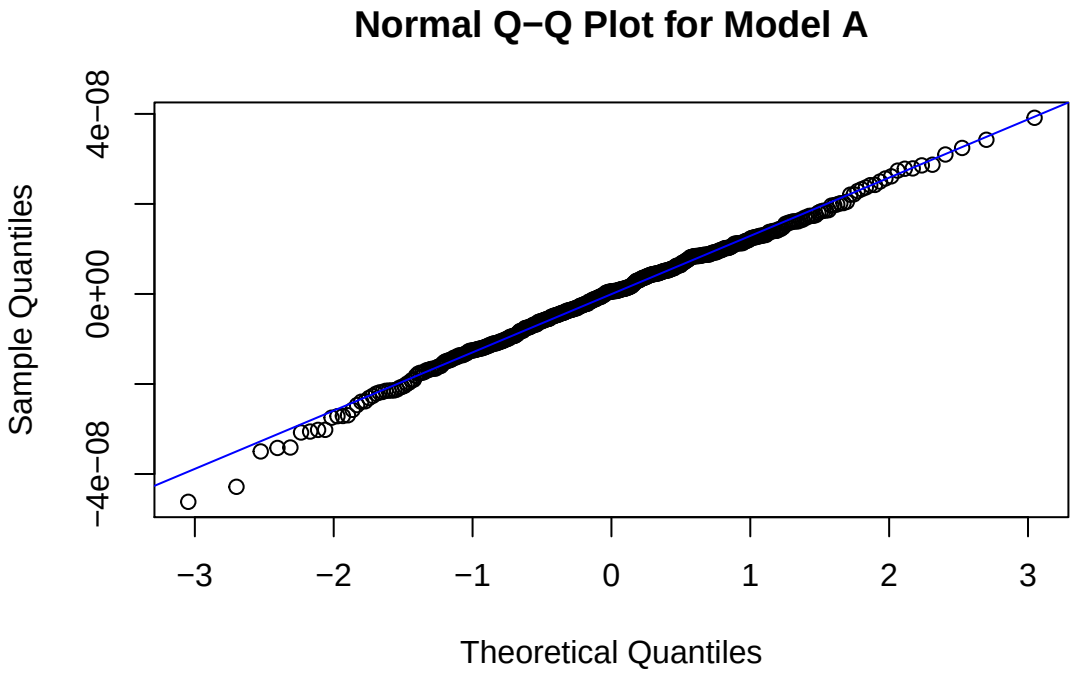


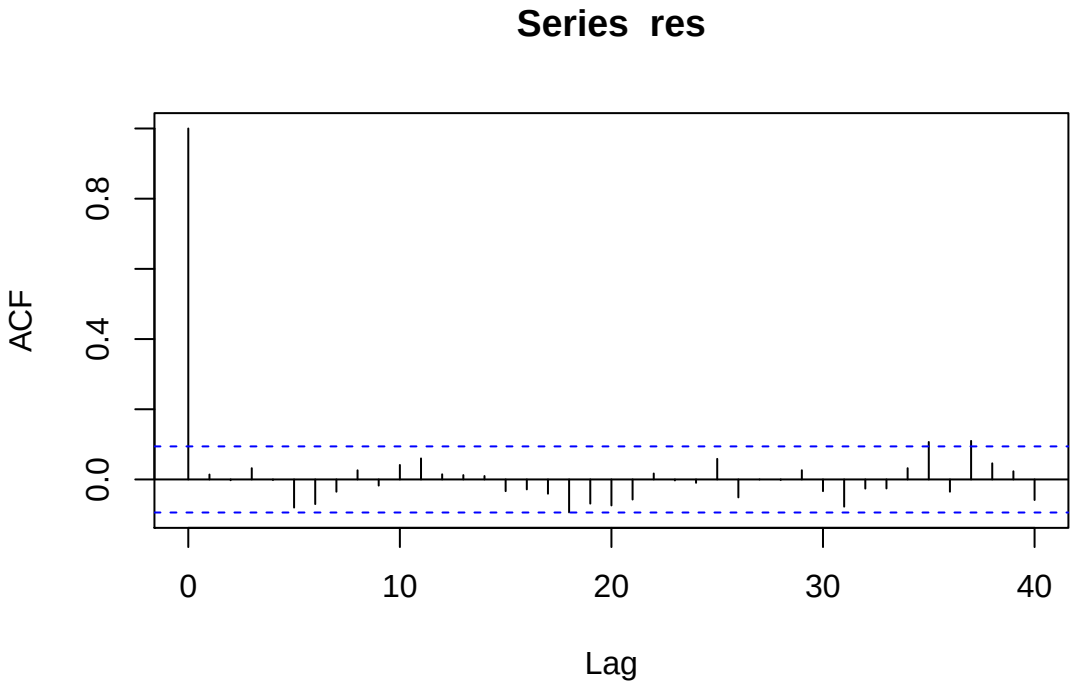

```
##
## Call:
## ar(x = res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.65e-16
```

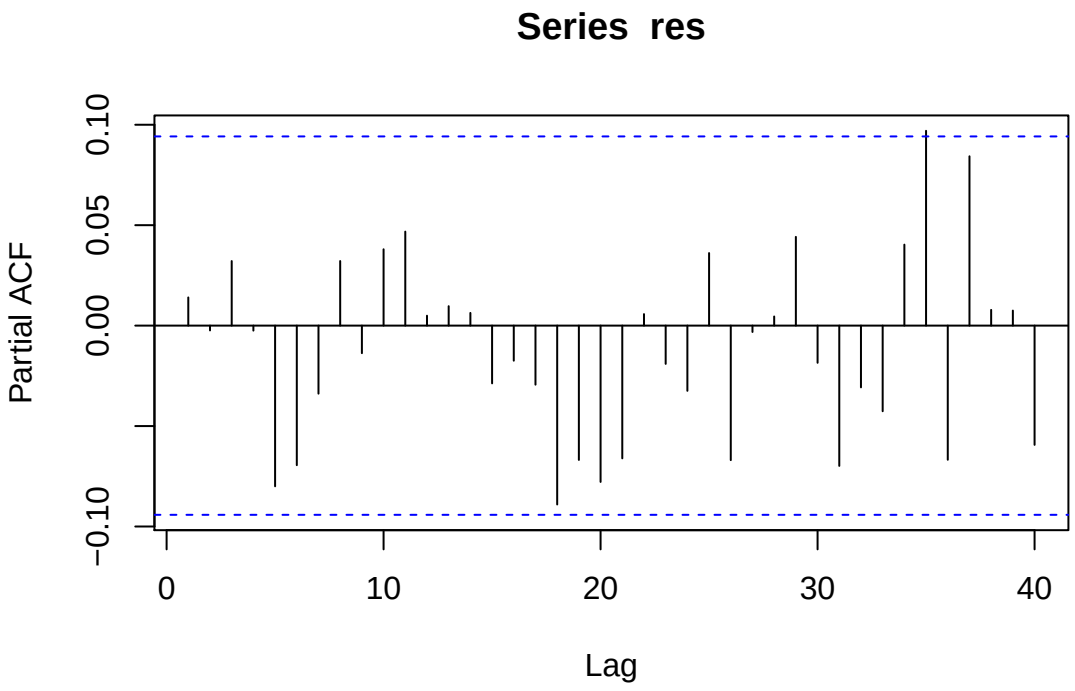


Model C diagostics:

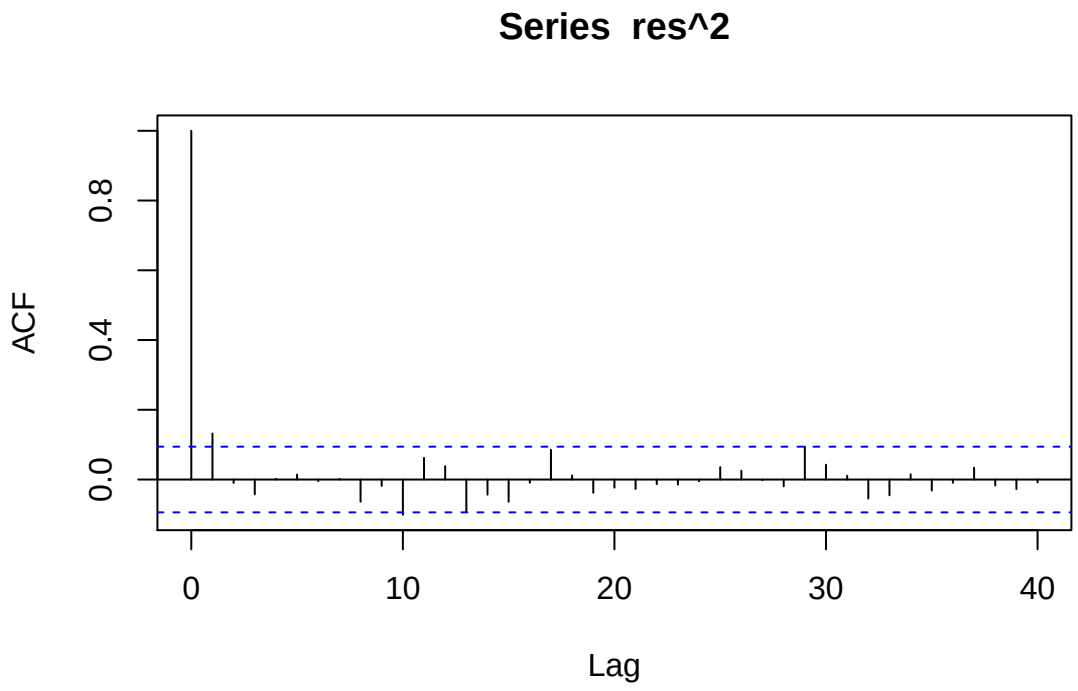



Normal Q−Q Plot for Model A
Sample Quantiles
Theoretical Quantiles
4e−08
0e+00
−4e−08
−3
−2
−1
0
1
2
3


Series res
ACF
Lag
0.8
0.4
0.0
0
10
20
30
40




```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.99602, p-value = 0.3498

##
##  Box-Pierce test
##
## data:  res
## X-squared = 8.7864, df = 9, p-value = 0.4572

##
##  Box-Ljung test
##
## data:  res
## X-squared = 8.9711, df = 9, p-value = 0.4399

##
##  Box-Ljung test
##
## data:  res^2
## X-squared = 17.392, df = 12, p-value = 0.1354
```




```
##
## Call:
## ar(x = res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1.694e-16
```

All three models pass the required tests with p-values higher than 0.05. Residuals appear random. Checking the ACF and PACF, we see that all lags are contained within the confidence intervals appropriately. The normal QQ plot for all three is approximately normal. Thus, we are left with the dilemma of which model to choose. We can safely eliminate Model B because the added complexity of p=4 vs. p=3 (in Model A) for worse diagnostic results makes Model B redundant. Between Model A and Model C, Model C actually outperforms Model A on some diagnostic checks despite having only a slightly lower AICc score. It is also considerably less complex, so we proceed to forecasting with Model C: SARIMA $(1,1,1)x(0,1,1)_{12}$.

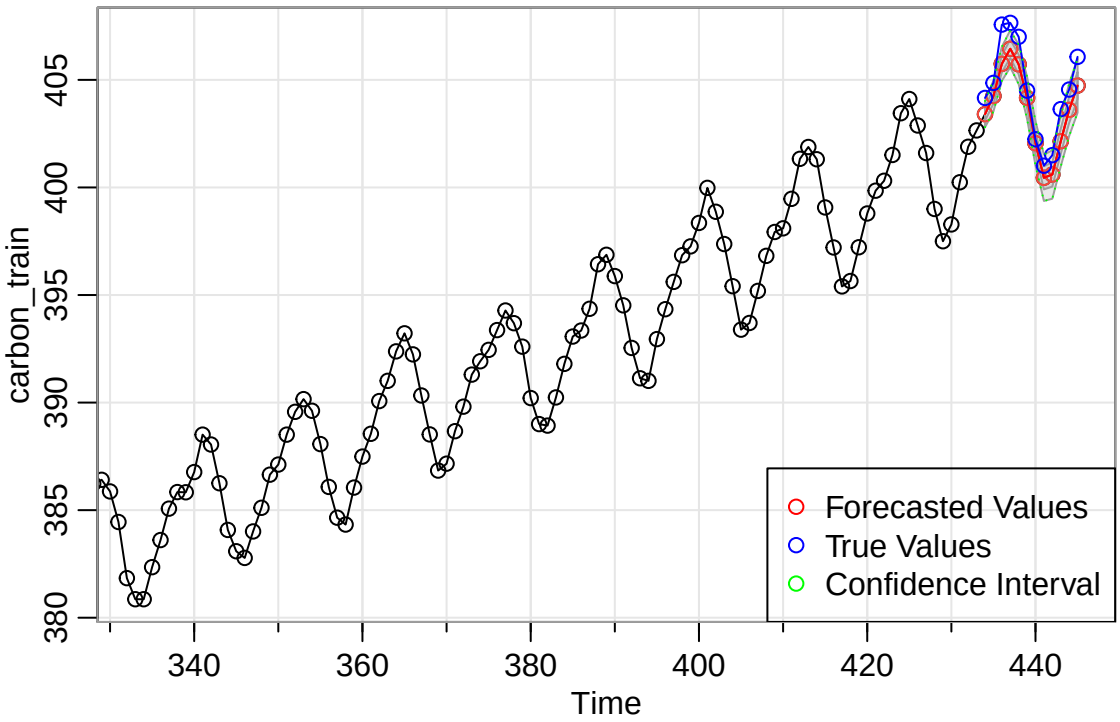
5. Forecasting using Model C

First, we create a table of our forecasts with prediction bounds:

```
##     Point Forecast        Lo 80        Hi 80        Lo 95        Hi 95
## 434   6.144476e-06 6.127674e-06 6.161279e-06 6.118779e-06 6.170173e-06
## 435   6.113200e-06 6.093076e-06 6.133324e-06 6.082423e-06 6.143977e-06
## 436   6.061304e-06 6.038888e-06 6.083720e-06 6.027021e-06 6.095587e-06
## 437   6.036104e-06 6.011709e-06 6.060499e-06 5.998795e-06 6.073413e-06
## 438   6.063950e-06 6.037745e-06 6.090156e-06 6.023873e-06 6.104028e-06
## 439   6.114822e-06 6.086927e-06 6.142717e-06 6.072161e-06 6.157483e-06
## 440   6.185394e-06 6.155907e-06 6.214880e-06 6.140298e-06 6.230489e-06
```

```
## 441   6.238290e-06 6.207294e-06 6.269287e-06 6.190885e-06 6.285696e-06
## 442   6.229310e-06 6.196873e-06 6.261746e-06 6.179702e-06 6.278917e-06
## 443   6.174644e-06 6.140829e-06 6.208460e-06 6.122928e-06 6.226360e-06
## 444   6.124923e-06 6.089783e-06 6.160063e-06 6.071181e-06 6.178665e-06
## 445   6.089700e-06 6.053283e-06 6.126116e-06 6.034006e-06 6.145394e-06
## 446   6.068293e-06 6.029330e-06 6.107256e-06 6.008704e-06 6.127882e-06
## 447   6.037461e-06 5.996575e-06 6.078347e-06 5.974931e-06 6.099991e-06
## 448   5.985658e-06 5.943024e-06 6.028292e-06 5.920455e-06 6.050862e-06
## 449   5.960478e-06 5.916181e-06 6.004774e-06 5.892732e-06 6.028223e-06
## 450   5.988328e-06 5.942434e-06 6.034222e-06 5.918139e-06 6.058517e-06
## 451   6.039200e-06 5.991762e-06 6.086638e-06 5.966650e-06 6.111751e-06
## 452   6.109772e-06 6.060839e-06 6.158705e-06 6.034936e-06 6.184609e-06
## 453   6.162669e-06 6.112286e-06 6.213053e-06 6.085614e-06 6.239724e-06
## 454   6.153688e-06 6.101895e-06 6.205481e-06 6.074477e-06 6.232899e-06
## 455   6.099023e-06 6.045857e-06 6.152189e-06 6.017713e-06 6.180333e-06
## 456   6.049302e-06 5.994798e-06 6.103806e-06 5.965946e-06 6.132658e-06
## 457   6.014078e-06 5.958269e-06 6.069888e-06 5.928725e-06 6.099432e-06
```

Now we can forecast our training data, using Model C, relative to the test data which contains the true values:



6. Conclusion

The goal of this project was to forecast the next 12 months in the CO2 training data. We see that our forecasted values are very close to the true values even though there appears to be a couple of true values that fall slightly outside the 95% confidence intervals. We conclude that our model, SARIMA$(1,1,1)x(0,1,1)_{12}$ is adequate. Much appreciation goes to our TAs Sunpeng and Jasmine for their help in identifying the model characteristics from the ACF/PACF graphs of the stationary data and for being available for general project

questions. And thank to you Professor Raya for providing the lecture slides, especially the example project in Lecture 15, and lab material which proved invaluable in completing this project.

7. References

Climate.gov Climate Change: Atmospheric Carbon Dioxide

PSTAT174 Lab 4

PSTAT174 Lab 5

PSTAT174 Lab 6

PSTAT174 Lab 7 - Solution

PSTAT174 Lecture 15 - Let's Do a Time Series Project!

7. Appendix of Code

```
# reading in data and checking
co2 <- read.csv("/Users/theolee/Desktop/co2.csv")
head(co2)
dim(co2)

# convert data to time series
carbon <- ts(co2[,2],start=c(1980),frequency=12)
head(carbon)

# take training and test data sets
carbon_train <- carbon[1:433]
carbon_test <- carbon[434:445]
length(carbon_test)

# plot training set to confirm non-stationarity
op <- par(mfrow = c(1,2))
ts.plot(carbon_train)
hist(carbon_train,col="light blue",main="histogram; c02 data")
op <- par(mfrow = c(1,2))
acf(carbon_train)
pacf(carbon_train)

# perform box-cox transformation
library(MASS)
fit = lm(carbon_train~as.numeric(1:length(carbon_train)))
bcTransform = boxcox(carbon_train~as.numeric(1:length(carbon_train)),plotit=TRUE)
# find lambda and perform transformation using method from Lab 4
lambda <- bcTransform$x[which(bcTransform$y==max(bcTransform$y))]
lambda
carbon.bc <- 1/(carbon_train^2)

# plot box-cox transformed data
plot.ts(carbon.bc,main="Box-Cox Transformed Time Series")
```

```r
hist(carbon.bc,col="light blue",xlab="",main="histogram;carbon.bc")
op = par(mfrow = c(1,2))
acf(carbon.bc,lag.max = 60,main = "")
pacf(carbon.bc,lag.max = 60,main = "")
title("Box-Cox Transformed Time Series", line = -1, outer=TRUE)
var(carbon_train);var(carbon.bc) # check variance from before and after

# check decomposition of box-cox transformed
library(ggplot2)
library(ggfortify)
y <- ts(as.ts(carbon.bc),frequency=12)
decomp <- decompose(y)
plot(decomp)

# difference at 12 to eliminate seasonality
carbon.bc_12 <- diff(carbon.bc,12)
plot.ts(carbon.bc_12,main="differenced at lag 12")
var(carbon.bc_12)
fit <- lm(carbon.bc_12~as.numeric(1:length(carbon.bc_12))); abline(fit,col="red")
mean(carbon.bc_12)
abline(h=mean(carbon.bc_12),col="blue")
op <- par(mfrow = c(1,2))
acf(carbon.bc_12)
pacf(carbon.bc_12)

# difference at 1 to eliminate trend
carbon.bc_1 <- diff(carbon.bc_12,1)
plot.ts(carbon.bc_1,main="differenced at lag 1")
var(carbon.bc_1)
fit <- lm(carbon.bc_1~as.numeric(1:length(carbon.bc_1))); abline(fit,col="red")
mean(carbon.bc_1)
abline(h=mean(carbon.bc_1),col="blue")
hist(carbon.bc_1,col="light blue",xlab="",main="histogram;carbon.bc_1")
par(mfrow=c(1,2))
acf(carbon.bc_1,lag.max=72)
pacf(carbon.bc_1,lag.max=72)

# checking variance to see if we should take another difference
paste("Before:")
var(carbon.bc_1)
paste("After another difference:")
carbon.bc_2 <- diff(carbon.bc_1,1)
var(carbon.bc_2)

# generating possible models using lab 6 example
library(MuMIn)
aiccs = matrix(NA, nr = 36, nc = 3)
colnames(aiccs) = c("p", "q", "AICc")
i=0
for(p in 0:4){
  for(q in 0:1){
    aiccs[i+1, 1] = p
    aiccs[i+1, 2] = q
```

```
    aiccs[i+1, 3] = AICc(arima(carbon.bc, order=c(p,1,q),seasonal=list(order=c(0,1,1),period=12), metho
    i = i+1
} }
## Models with the first three smallest AICc
aiccs[order(aiccs[,3])[1:3],]

# model A test diagnostics
modelA <- arima(carbon.bc,order=c(3,1,1),seasonal=list(order=c(0,1,1),period=12),method="ML")
res <- residuals(modelA)
hist(res,density=20,breaks=20,col="blue",xlab="",prob=TRUE)
m <- mean(res)
std <- sqrt(var(res))
curve(dnorm(x,m,std),add=TRUE)
plot.ts(res)
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")
qqnorm(res,main= "Normal Q-Q Plot for Model A")
qqline(res,col="blue")
acf(res, lag.max=40)
pacf(res, lag.max=40)
shapiro.test(res)
Box.test(res, lag = 12, type = c("Box-Pierce"), fitdf = 5)
Box.test(res, lag = 12, type = c("Ljung-Box"), fitdf = 5)
Box.test(res^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
acf(res^2, lag.max=40)
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# model B test diagnostics
modelB <- arima(carbon.bc,order=c(4,1,1),seasonal=list(order=c(0,1,1),period=12),method="ML")
res <- residuals(modelB)
hist(res,density=20,breaks=20,col="blue",xlab="",prob=TRUE)
m <- mean(res)
std <- sqrt(var(res))
curve(dnorm(x,m,std),add=TRUE)
plot.ts(res)
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")
qqnorm(res,main= "Normal Q-Q Plot for Model A")
qqline(res,col="blue")
acf(res, lag.max=40)
pacf(res, lag.max=40)
shapiro.test(res)
Box.test(res, lag = 12, type = c("Box-Pierce"), fitdf = 6)
Box.test(res, lag = 12, type = c("Ljung-Box"), fitdf = 6)
Box.test(res^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
acf(res^2, lag.max=40)
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# model C test diagnostics
modelC <- arima(carbon.bc,order=c(1,1,1),seasonal=list(order=c(0,1,1),period=12),method="ML")
res <- residuals(modelC)
hist(res,density=20,breaks=20,col="blue",xlab="",prob=TRUE)
m <- mean(res)
```

```r
std <- sqrt(var(res))
curve(dnorm(x,m,std),add=TRUE)
plot.ts(res)
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")
qqnorm(res,main= "Normal Q-Q Plot for Model A")
qqline(res,col="blue")
acf(res, lag.max=40)
pacf(res, lag.max=40)
shapiro.test(res)
Box.test(res, lag = 12, type = c("Box-Pierce"), fitdf = 3)
Box.test(res, lag = 12, type = c("Ljung-Box"), fitdf = 3)
Box.test(res^2, lag = 12, type = c("Ljung-Box"), fitdf = 0)
acf(res^2, lag.max=40)
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))

# forecasting
library(forecast)
forecast(modelC)

# plotting training + test sets using solution to lab 7
library(astsa)
mypred <- sarima.for(carbon_train, n.ahead=12, plot.all=F, p=1, d=1, q=1, P=0, D=1, Q=1, S=12)
lines(434:445, mypred$pred,col="red")
lines(434:445, carbon_test, col="blue")
points(434:445, carbon_test, col="blue")
lines(mypred$pred + 2*mypred$se, col="green", lty="dotted")
lines(mypred$pred - 2*mypred$se, col="green", lty="dotted")
legend("bottomright", pch=1, col=c("red", "blue","green"), legend=c("Forecasted Values", "True Values",
```