# PSTAT 174 Final Project:
# Time Series Analysis of Monthly Global Methane Data

Cameron Joe

6/4/2021

# 1. Abstract

Currently, climate change poses one of the greatest challenges humanity has ever faced. Although there are multiple greenhouse gases that contribute to this phenomenon, methane ($CH_4$) stands out as one of the more integral compounds with its ability to trap heat over an 100-year time period being 28 times more potent than $CO_2$ (Stein). The objective of this paper is to analyze average monthly global methane recordings, propose a SARIMA model of the data, test the model for validity, and assess the model's predictive validity by forecasting. After transforming and differencing the data, ACF/PACF analysis and maximum likelihood parameter estimation resulted in the following model:

$$\begin{aligned}&(1 + 0.6250_{(0.1499)}B + 0.5395_{(0.0745)}B^2)\nabla_1\nabla_{12}bc(U_t) = \\ &\quad (1 + 1.8660_{(0.1721)}B + 1.2115_{(0.3040)}B^2 + 0.2719_{(0.1465)}B^3)(1 - 0.8747_{(0.0449)}B^{12})Z_t; \sigma_Z^2 = 1684069 \quad (1)\end{aligned}$$

Analysis of residuals and portmanteau testing was conducted to confirm the validity of the model. After forecasting for the next year and untransforming data, it was found that the some of the observed test values were not within predictive confidence intervals. This failure in the model could be due to the unexpected surge in methane during 2020 (Stein). Although test values were not in the confidence interval, predicted values were still relatively close to observed values.

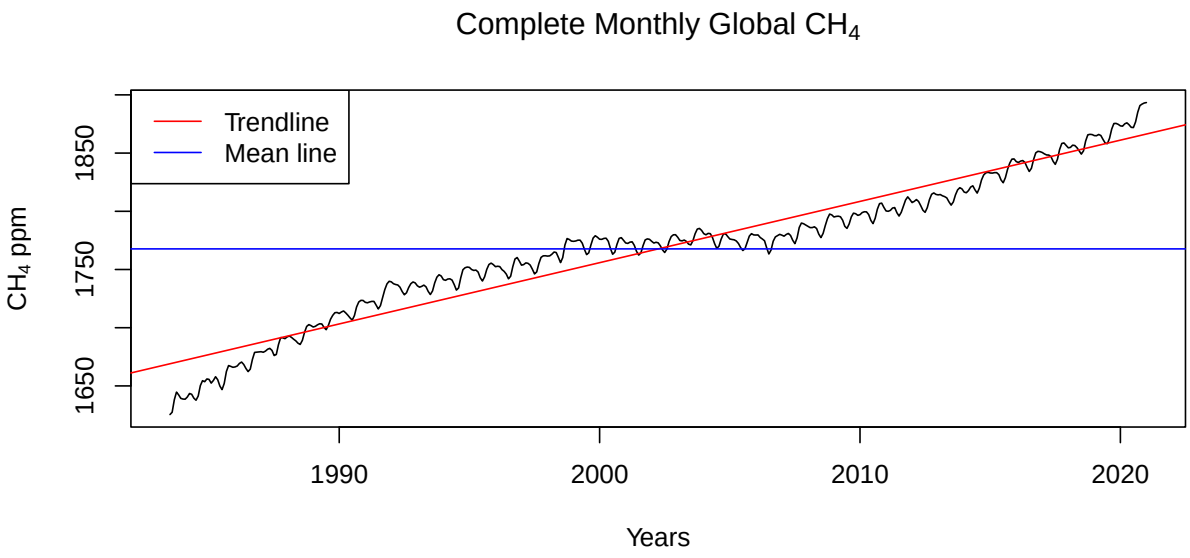
# 2. Main Body

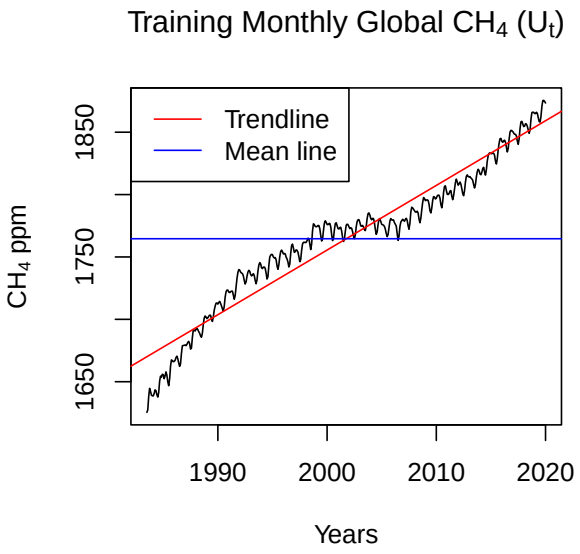
## Introduction

There are three main goals underlying our analysis: to propose an accurate SARIMA model of the data, test and analyze the model for validity, and utilize the model to forecast (and comparing these forecasts to test data). All analysis was conducted using R statistical software (R Core Team). The data utilized for analysis was the "globally averaged marine surface monthly mean data" collected by the Global Monitoring Division of NOAA's Earth System Research Laboratory (Dlugokencky). Recorded data was measured in parts per million and ranged in time from July of 1983 up to January of 2021 - 451 observations in total. As was mentioned previously, methane is a key player in driving climate change. As time passes, understanding climate change and its driving factors is becoming increasingly more valuable as humans attempt to predict the trajectory of our uncertain future and guide our endeavors to avoid the worst possible outcomes.

The overarching plan of analysis follows the Box-Jenkins methodology. First, a Box-Cox transformation was performed on the data to bring it closer to a Gaussian. After differencing the transformed data at lags 12 and 1, ACF/PACF analysis allowed us to propose multiple models. The two most promising models were selected based on AICc and parsimony. Then, model parameters were found through maximum likelihood estimation. To ascertain the validity of these models, the roots for the moving average and autoregressive portions were plotted to ensure stationary, causality, and invertibility. Model residuals underwent ACF/PACF analysis and portmanteau testing and a single model ,$SARIMA(2,1,3) \times (0,1,1)_{12}$, was selected. Lastly, forecasting was conducted on the transformed data using the selected model and these results were then untransformed. It was found that seven out of the twelve test data points were not within predicted confidence intervals, but predicted estimates were still close to test values. This discrepancy in forecasted results may be due to an uncharacterstically large increase in both methane and carbon-dioxide during the 2020 year (Stein).

## Plotting and Analyzing Data





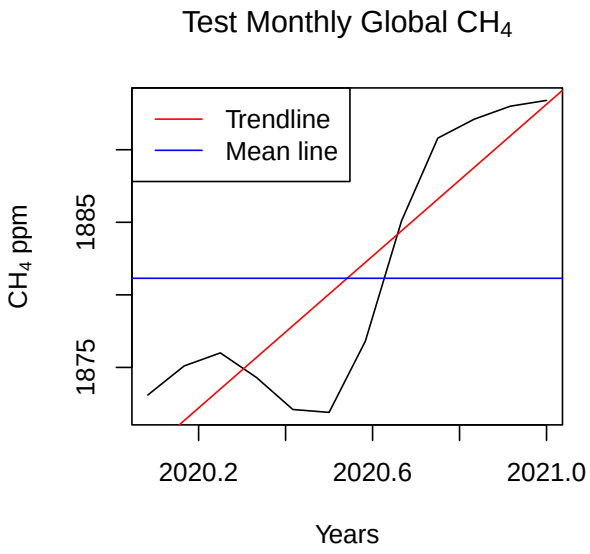
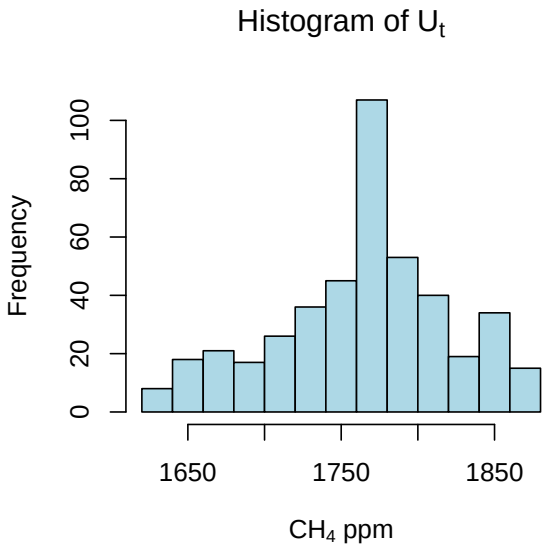



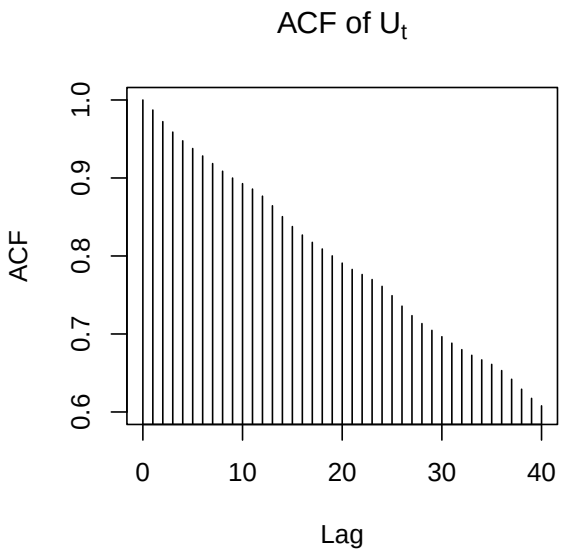

Table 1: Table of Trendline Slopes

| | Complete fit | Training fit | Test fit |
|---|---|---|---|
| Slope | 5.2625 | 5.1829 | 26.1608 |

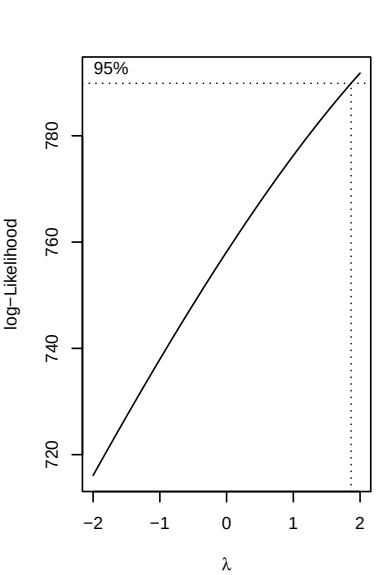
Looking at the time series plots, there is an evident increasing trend in global $CH_4$; the linear regression model fit to the entire data outputs a slope of 5.262. The trend does not appear to follow a strictly linear trend, but something like a third-order polynomial. The slope of the data appears to be greater at the beginning and end of the data, while there appears to be almost no trend towards the middle of the plot. This is evident as the linear fit of the test data has a slope of 26.16, which is significantly greater than 5.262 - the slope from the complete data fit. In addition, there is a yearly (s = 12) seasonality in the data. One full cycle of the data can observed in the test data plot which consists of the 12 most recent $CH_4$ observations ranging from February 2020 to January 2021. The training partition will be referred to as $U_t$ for the remainder of the report. Visually, there appears to be no changes in variance or significant sudden changes in behavior.

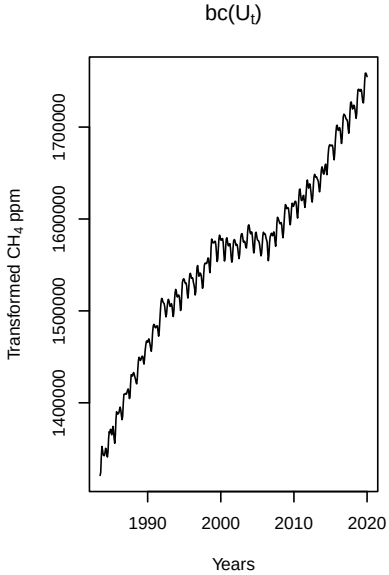
## Transformations and Differencing

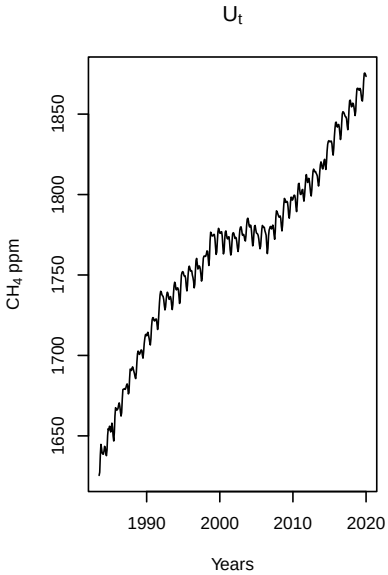


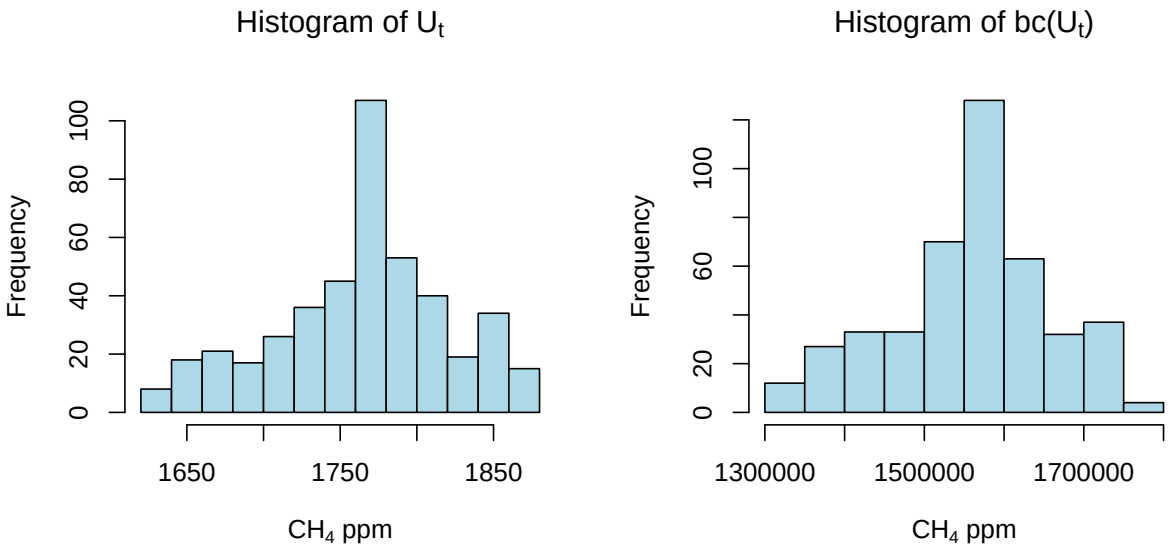




Looking at the histogram of the data, the $CH_4$ data clearly does not conform to a normal distribution well; the tails of the histogram are too large while the mode at ~1775 ppm is extremely high in frequency. So, a Box-Cox transformation will be performed to bring the distribution of the data closer to normal. The ACF values of the data remain relatively large and decrease very slowly with lag. There is also extremely slight periodicity in ACFs with peaks at multiples of 12. Differencing the data at a lag of 1 can help decrease the ACF values and differencing at a lag of 12 can aid in removing periodicity.

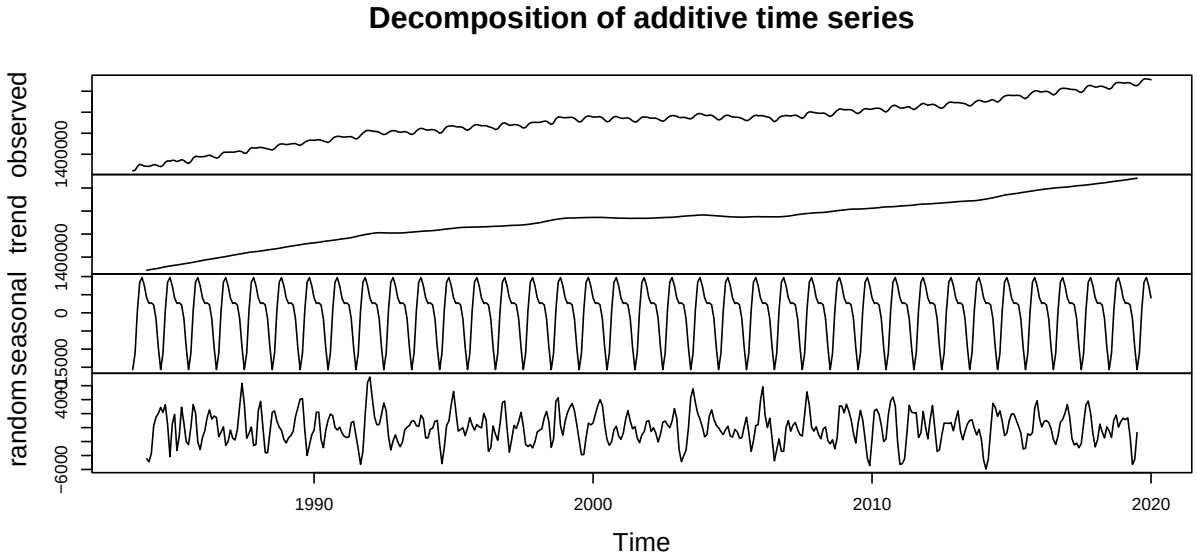





The Box-Cox transformation yielded a $\lambda = 2$ -the maximum possible value-, so a Box-Cox transformation was conducted over a log transformation. The time series plot of our transformed data maintains a similar shape, but our transformed data fits a normal distribution better with tail lengths being reduced and the size of the mode being reduced relative to other value frequencies.




Looking at a decomposition of $bc(U_t)$, we can observe that there is a clear yearly seasonality and a roughly linear trend. Therefore, it makes sense to difference $bc(U_t)$ at lags of 12 and 1.

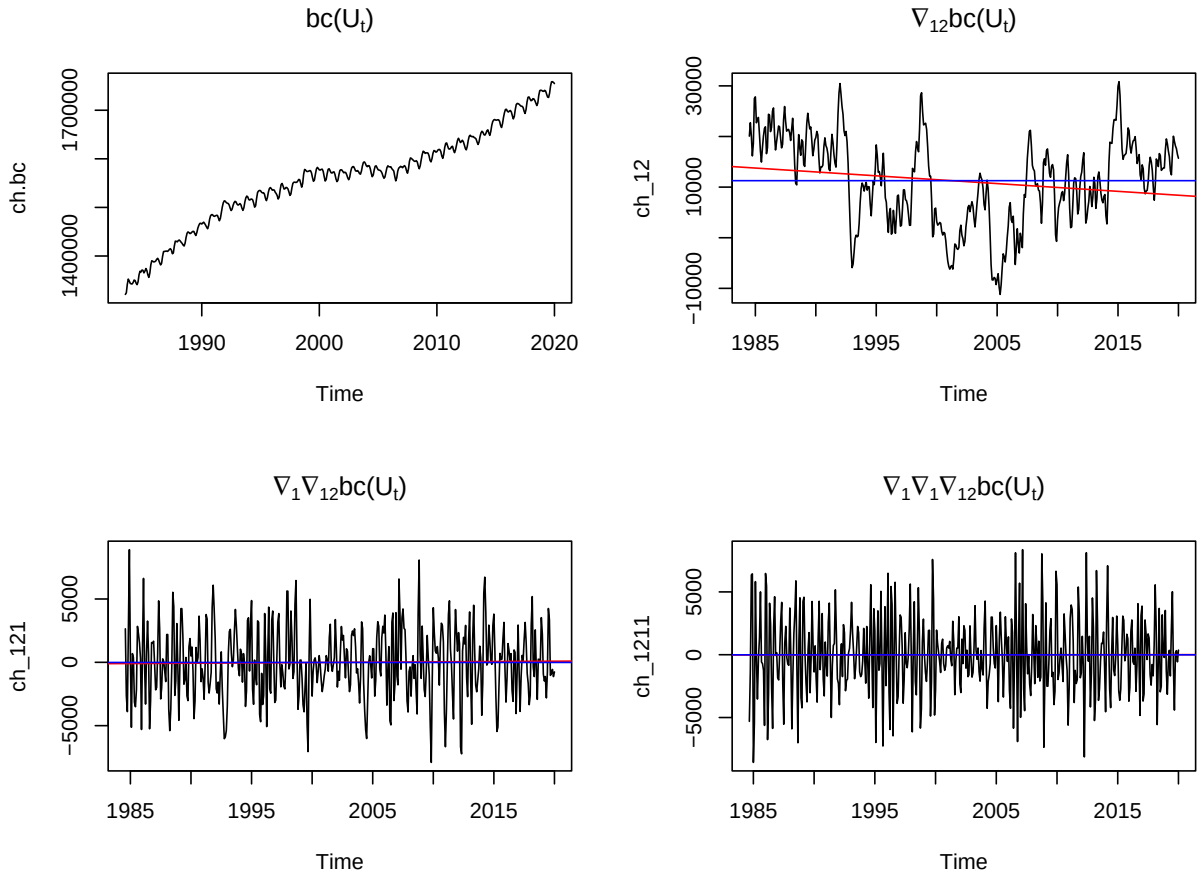
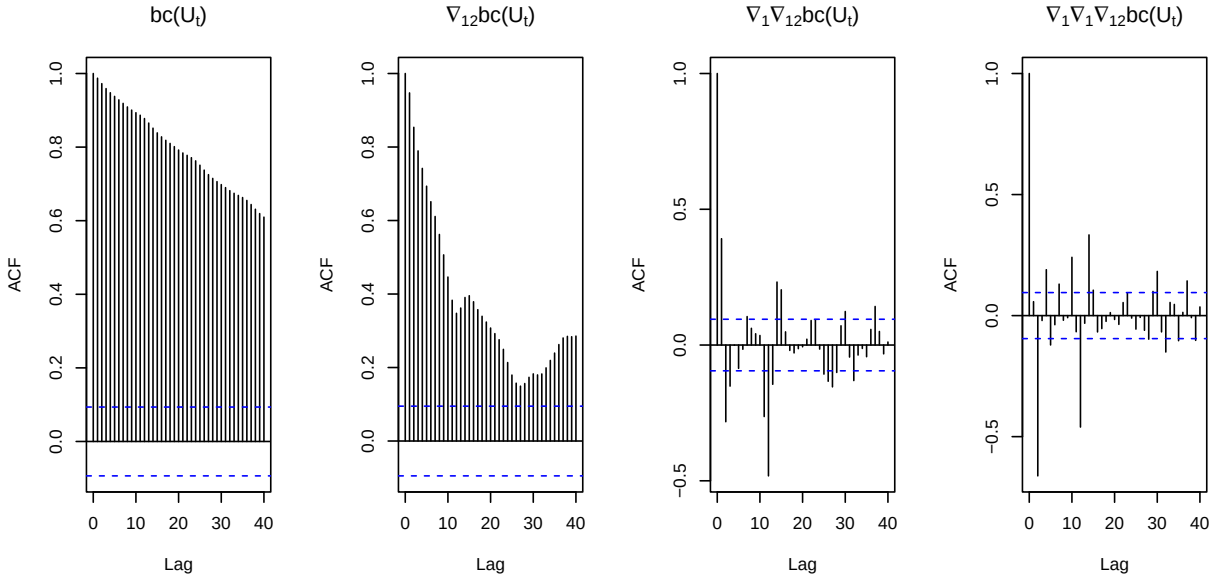


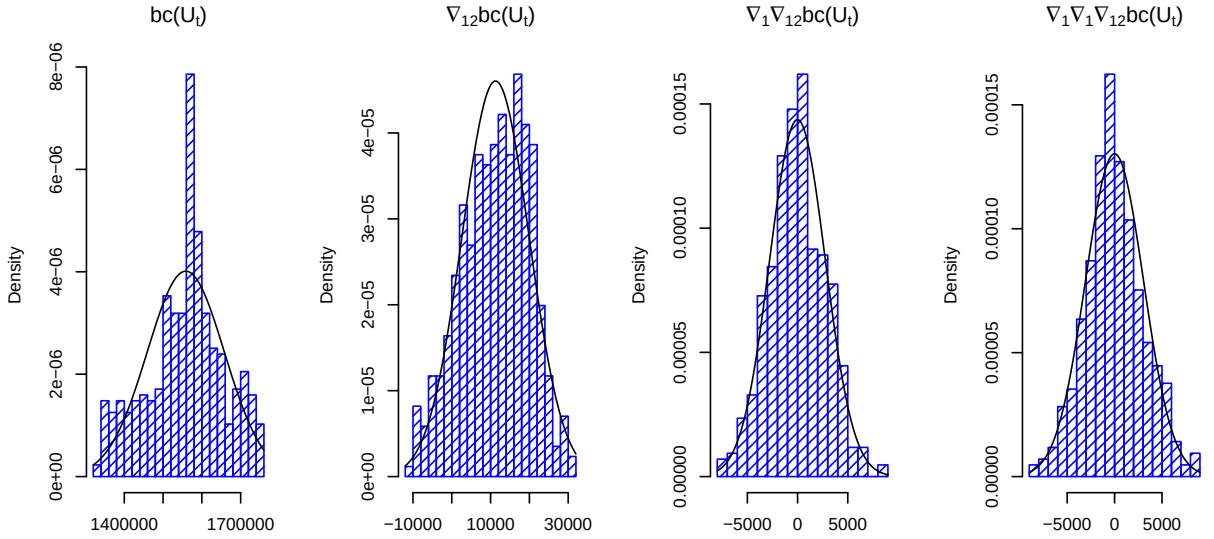

Table 2: Variance of Differenced Data

| | $BC(U_t)$ | $\nabla_{12}bc(U_t)$ | $\nabla_1\nabla_{12}bc(U_t)$ | $\nabla_1\nabla_1\nabla_{12}bc(U_t)$ |
|---|---|---|---|---|
| Variance | 9887027800 | 75036548 | 7702777 | 9376896 |

Looking at the variances, differencing at lag 12 significantly reduced variance from 9.887e+09 to 75,036,548 and visually removed the seasonality of the plot. Additional differencing at lag 1 further reduced variance from 75,036,548 to 7,702,777 and reduced the trend in the data. As there still remained a slight trend in the data, differencing at lag 1 was conducted again. This removed almost all trend, but caused variance to increase from 7,702,777 to 9,376,896. In addition, the graph of the data shows an increase in overall noise. Therefore, $\nabla_1\nabla_{12}bc(U_t)$ was selected for use in further analysis as seasonality and trend was sufficiently removed and the variance of the data was the lowest.

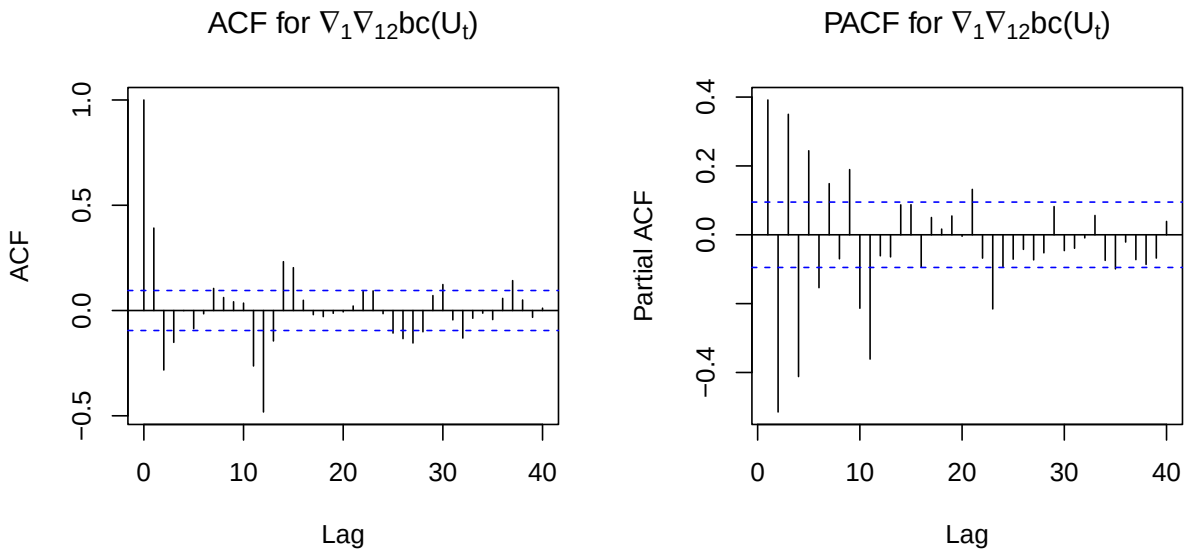



ACF plots allow another form of visual analysis on the transformations. As we had expected, differencing at a lag of 12 removed the seasonality of the data and differencing at lag of 1 causes ACF to decay more rapidly which indicates that $\nabla_1\nabla_{12}BC(U_t)$ is stationary. Differencing at the lag of 1 increases some ACF values which is another indicator to use $\nabla_1\nabla_{12}BC(U_t)$ over $\nabla_1\nabla_1\nabla_{12}BC(U_t)$.




Examining the distribution of the transformed data, it can be seen that $\nabla_1\nabla_{12}BC(U_t)$ is both symmetrical and fits the Gaussian distribution well. Therefore, the following analysis of ACF and PACF will be conducted on $\nabla_1\nabla_{12}BC(U_t)$ to identify potential models.

## Identifying Potential Models through ACF and PACF




First, as we have already differenced at lag of 12 and 1 once, we know that $s = 12$, $d = 1$, and $D = 1$. Looking at the ACF plot, we observe a significant ACF values at lag 12. As this is a multiple of $s = 12$, we are led to believe that $Q = 1$. (Our hypothesis that $Q \neq 0$ is supported by the fact that there are significant slowly-decreasing PACF values at multiples of 12.) In addition, ACF values are significant at lags 1, 2, and 3 in addition to 11, 12, 13, 14, and 15 - values surrounding lag 12. So, we can hypothesize that $q = 2$ *or* 3. The absence of significant slowly-decreasing ACF values at multiples of 12 hints that $P = 0$. Looking at the PACF plot, we see oscillating, slowly-decreasing PACF values starting from 1 which indicates that $q \neq 0$. In addition, as this pattern only occurs once at the beginning of the plot, there is converging evidence that $P = 0$. We also observe significant PACF values within $\pm 2$ *or* 3 of multiples of 12 (such as lags 9, 10, and 11 which are neighbors to 12). So, we can hypothesize that $p = 2$ *or* 3. Therefore the models to be fitted will include $p = 2, 3$, $d = 1$, $q = 2, 3$, $P = 0$, $D = 1$, $Q = 1$, and $s = 12$.

## Model Fitting and Diagnostic Checking

```
# Model fitting
mod1 <- arima(ch.bc, order = c(2, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod1
```

```
##
## Call:
## arima(x = ch.bc, order = c(2, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12),
##     method = "ML")
##
## Coefficients:
##           ar1      ar2     ma1     ma2     ma3     sma1
##       -0.6250  -0.5395  1.8660  1.2115  0.2719  -0.8747
## s.e.   0.1499   0.0745  0.1721  0.3040  0.1465   0.0449
##
## sigma^2 estimated as 1684069:  log likelihood = -3668.7,  aic = 7351.4
```

```
mod2 <- arima(ch.bc, order = c(3, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod2
```

```
##
## Call:
## arima(x = ch.bc, order = c(3, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12),
##     method = "ML")
##
## Coefficients:
```

```
## Warning in sqrt(diag(x$var.coef)): NaNs produced
```

```
##          ar1      ar2     ar3     ma1      ma2      ma3     sma1
##       0.3374  -0.1504  0.2813  0.8621  -0.4486  -0.4718  -0.8489
## s.e.     NaN      NaN     NaN     NaN      NaN      NaN      NaN
##
## sigma^2 estimated as 1696920:  log likelihood = -3669.11,  aic = 7354.23
```

```
mod3 <- arima(ch.bc, order = c(3, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod3
```

```
##
## Call:
## arima(x = ch.bc, order = c(3, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12),
##     method = "ML")
##
## Coefficients:
##           ar1      ar2     ar3     ma1     ma2     sma1
##       -0.2052  -0.4185  0.0855  1.4041  0.4928  -0.8434
## s.e.   0.1241   0.0472  0.0708  0.1290  0.1275   0.0485
##
## sigma^2 estimated as 1715306:  log likelihood = -3671.38,  aic = 7356.76
```

```
mod4 <- arima(ch.bc, order = c(2, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod4
```

```
##
## Call:
## arima(x = ch.bc, order = c(2, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12),
##     method = "ML")
##
## Coefficients:
##           ar1      ar2     ma1     ma2     sma1
##       -0.3668  -0.4103  1.5687  0.6633  -0.8522
## s.e.   0.0599   0.0488  0.0508  0.0502   0.0450
##
## sigma^2 estimated as 1696092:  log likelihood = -3669.14,  aic = 7350.28
```

Table 3: Model Estimates and AICs

| | $\phi_1$ | $\phi_2$ | $\phi_3$ | $\theta_1$ | $\theta_2$ | $\theta_3$ | $\Theta_1$ | AIC |
|---|---|---|---|---|---|---|---|---|
| SARIMA(2, 1, 3) x (0, 1, 1)_12 | -0.63 | -0.54 | NA | 1.87 | 1.21 | 0.27 | -0.87 | 7351.40 |
| SARIMA(3, 1, 3) x (0, 1, 1)_12 | 0.34 | -0.15 | 0.28 | 0.86 | -0.45 | -0.47 | -0.85 | 7354.23 |
| SARIMA(3, 1, 2) x (0, 1, 1)_12 | -0.21 | -0.42 | 0.09 | 1.40 | 0.49 | NA | -0.84 | 7356.76 |
| SARIMA(2, 1, 2) x (0, 1, 1)_12 | -0.37 | -0.41 | NA | 1.57 | 0.66 | NA | -0.85 | 7350.28 |

Looking through our various models, the $SARIMA(2,1,3) \times (0,1,1)_{12}$ and $SARIMA(2,1,2) \times (0,1,1)_{12}$ have the lowest AICs of 7351.40 and 7350.28. By parsimony (having the fewest estimated parameters) and the fact that it has the lowest AIC, the $SARIMA(2,1,2) \times (0,1,1)_{12}$ model has been chosen as the primary

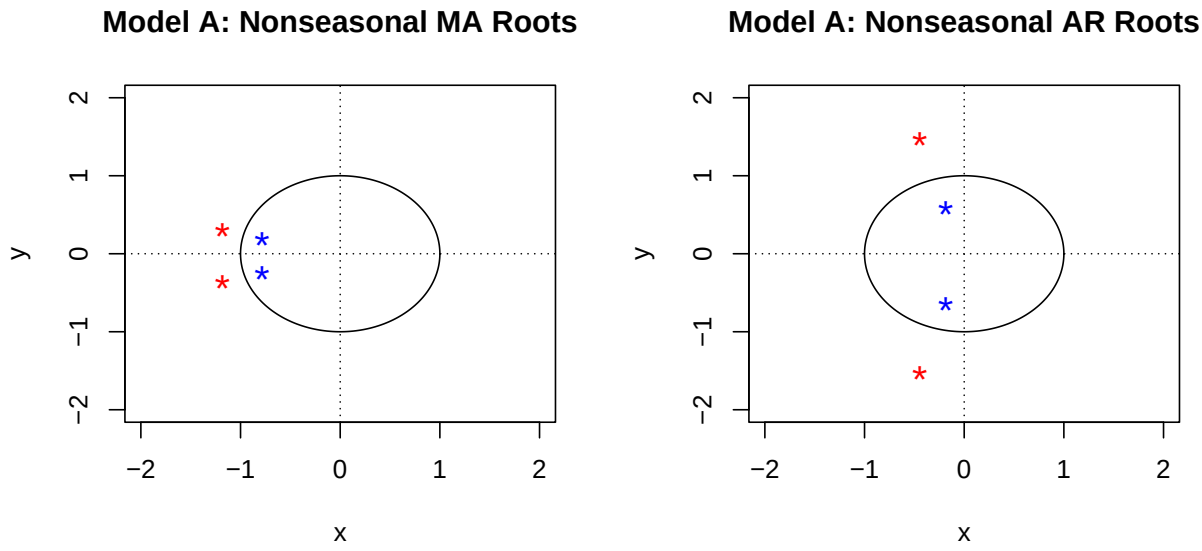
candidate model. Based off of prior ACF/PACF analysis, this model did appear to be the one of the most probable. This model will be referred to as model A and can be written algebraically as:

$$(1 + 0.3668_{(0.0599)}B + 0.4103_{(0.0488)}B^2)\nabla_1\nabla_{12}bc(U_t) = (1 + 1.5687_{(0.0508)}B + 0.6633_{(0.0502)}B^2)(1 - 0.8522_{(0.0450)}B^{12})Z_t; \sigma_Z^2 = 1696092 \quad (2)$$

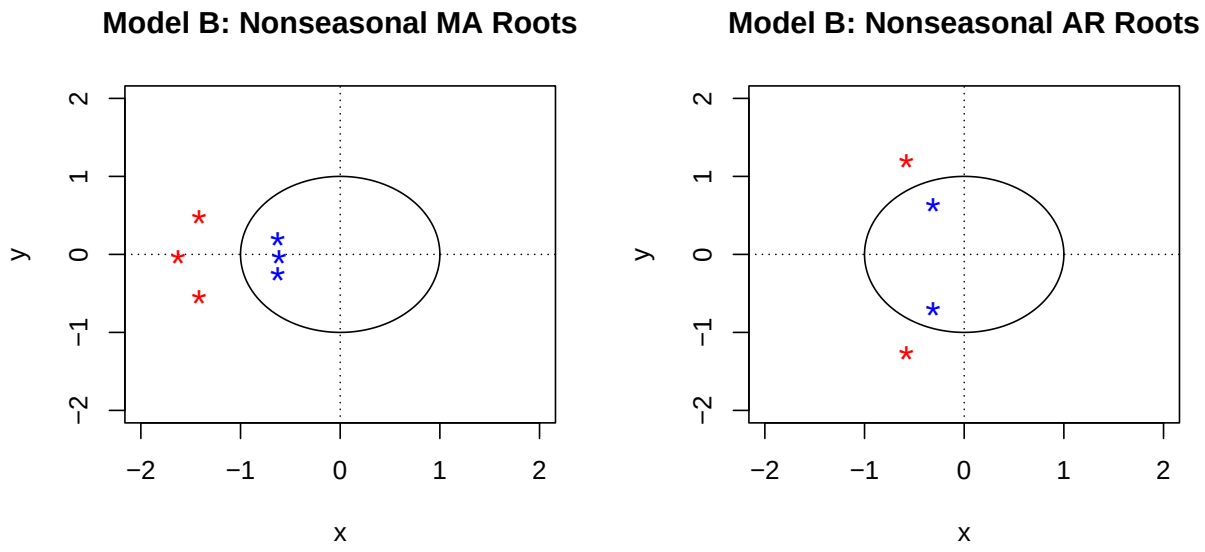
Our second model will be referred to as model B and can be written as:

$$(1 + 0.6250_{(0.1499)}B + 0.5395_{(0.0745)}B^2)\nabla_1\nabla_{12}bc(U_t) = (1 + 1.8660_{(0.1721)}B + 1.2115_{(0.3040)}B^2 + 0.2719_{(0.1465)}B^3)(1 - 0.8747_{(0.0449)}B^{12})Z_t; \sigma_Z^2 = 1684069 \quad (3)$$

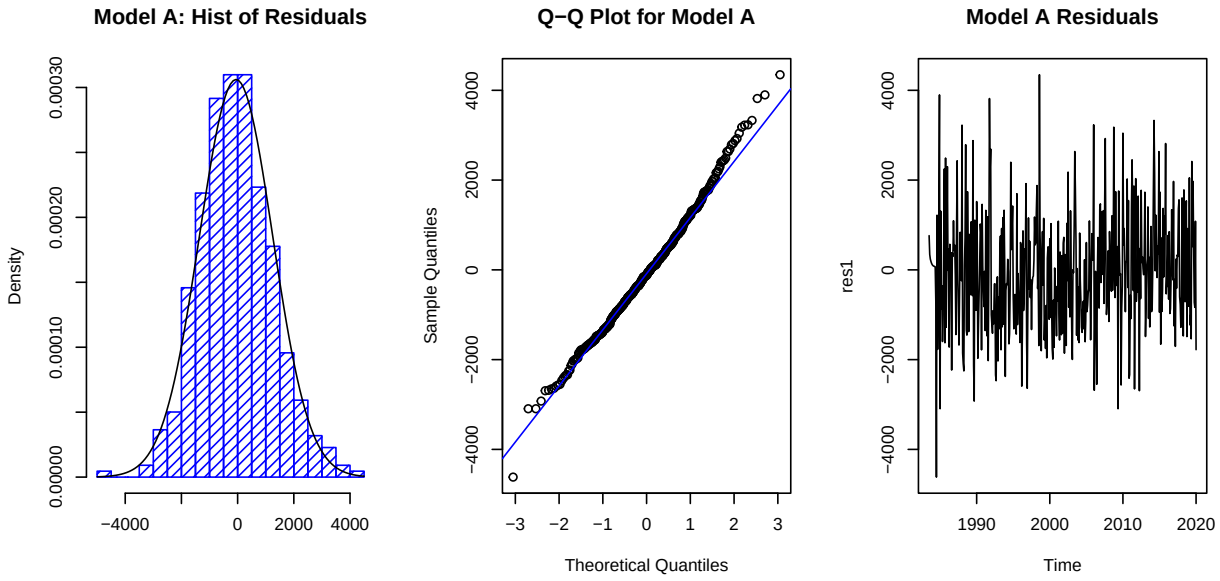
```
# plot model A roots
source("plot.roots.R")
par(mfrow = c(1, 2))
plot.roots(NULL, polyroot(c(1, 1.5687, 0.6633)), main = "Model A: Nonseasonal MA Roots")
plot.roots(NULL, polyroot(c(1, 0.3668, 0.4103)), main = "Model A: Nonseasonal AR Roots")
```



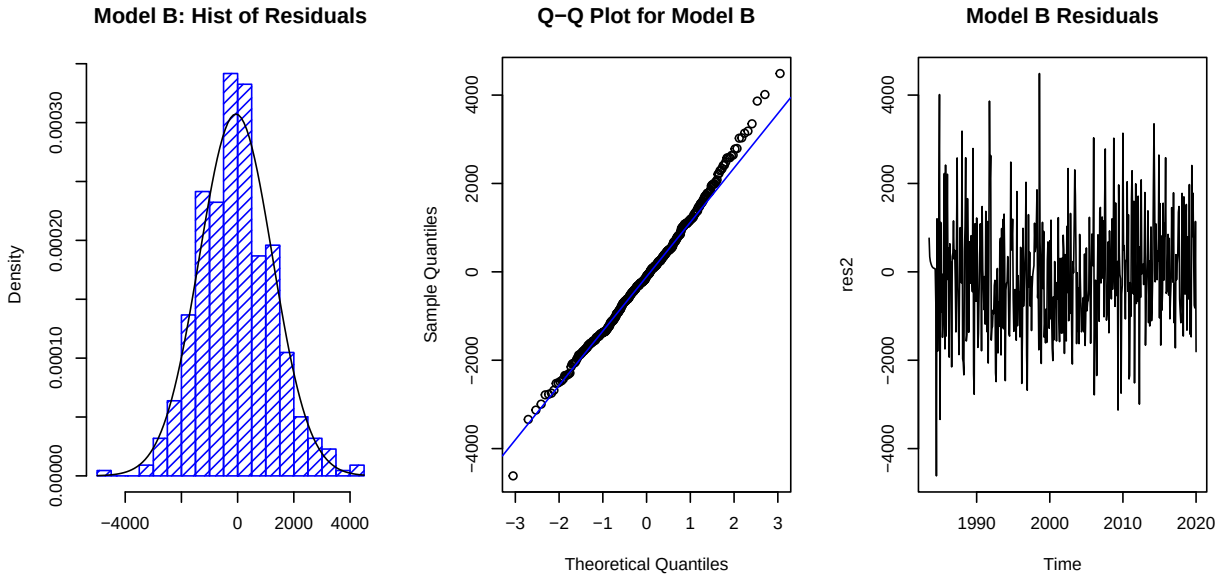

```
# plot model B roots
par(mfrow = c(1, 2))
plot.roots(NULL, polyroot(c(1, 1.8660,  1.2115,  0.2719)), main = "Model B: Nonseasonal MA Roots")
plot.roots(NULL, polyroot(c(1, 0.6250,  0.5395)), main = "Model B: Nonseasonal AR Roots")
```




For both model A and B, $|\Theta_1| < 1$, so both have seasonal moving average parts with roots outside of the unit circle. In addition, for both models, unit roots for the MA portion lie outside of the unti circle. Therefore, both models have met the conditions necessary for invertibility. As depicted above, the AR portion of both models have roots outside of the unit circle which implies that both models are stationary and causal.




For model A, the histogram of the residuals appears to follow a Gaussian distribution well. Looking at a Q-Q plot, residuals follow the Q-Q line relatively well, but there is some deviation at extreme values. The plot of residuals has no trend, seasonality, or changes in variance. Therefore, initial inspection of model residuals looks acceptable.




The distribution of model B residuals roughly follows a Gaussian distribution, although there appears to be more deviation in comparison to model A residuals. The Q-Q plot for model B appears to follow the Q-Q line well with only residuals at the upper quantiles deviating. Lastly, a time series plot of the residuals reveals that there is no trend, change in variance, or seasonality. Overall, the residuals for model B appear to be acceptable as well.

```
# Model A testing
# length(res2) ^ 0.5 = 20.95233 which is rounded to 21
shapiro.test(res1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res1
## W = 0.9934, p-value = 0.05181
```

```
Box.test(res1, lag = 21, type = c("Box-Pierce"), fitdf = 5)
```

```
##
##  Box-Pierce test
##
## data:  res1
## X-squared = 22.749, df = 16, p-value = 0.1206
```

```
Box.test(res1, lag =21, type = c("Ljung-Box"), fitdf = 5)
```

```
##
##  Box-Ljung test
##
## data:  res1
## X-squared = 23.448, df = 16, p-value = 0.1023
```

```
Box.test((res1)^ 2, lag = 21, type = c("Ljung-Box"), fitdf = 0)
```

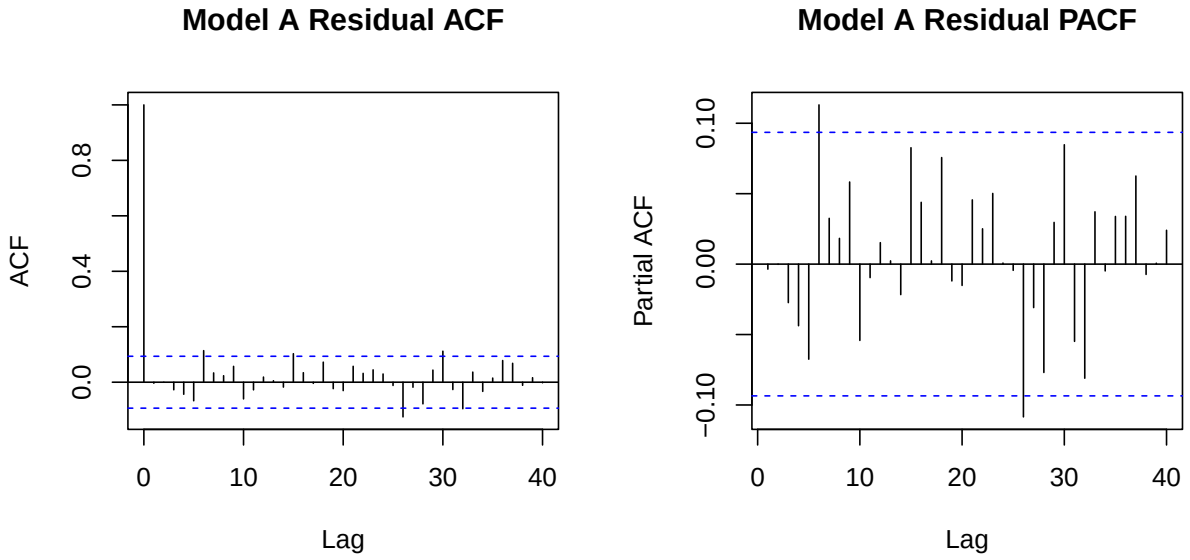
```
##
##  Box-Ljung test
##
```

```
## data:  (res1)^2
## X-squared = 12.797, df = 21, p-value = 0.9156
```

```
ar(res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```
##
## Call:
## ar(x = res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1698342
```

```
par(mfrow = c(1, 2))
acf(res1[c(1:length(res1))], lag.max = 40, main = "Model A Residual ACF")
pacf(res1[c(1:length(res1))], lag.max = 40, main = "Model A Residual PACF")
```




Model A passes all portmanteau tests, with p-values for the Shapiro-Wilk, Box-Pierce, Box-Ljung, and McLeod-Li tests being 0.05181, 0.1206, 0.1023, and 0.9156 respectively. Attempting to fit the residuals to an auto-regressive model, we receive an AR(0) fit indicating that the proposed model is sufficient. Looking at residual ACF and PACF, there are a few values that exceed the confidence interval. As R outputs conservative estimates for the confidence intervals, ACF and PACF values that barely exceed the confidence interval may be ignored.

```
# Model B testing
shapiro.test(res2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res2
## W = 0.99442, p-value = 0.1102
```

```
Box.test(res2, lag = 21, type = c("Box-Pierce"), fitdf = 6)
```

```
##
##  Box-Pierce test
##
```

```
## data:  res2
## X-squared = 17.256, df = 15, p-value = 0.3038
```

```
Box.test(res2, lag =21, type = c("Ljung-Box"), fitdf = 6)
```

```
##
##  Box-Ljung test
##
## data:  res2
## X-squared = 17.773, df = 15, p-value = 0.2748
```

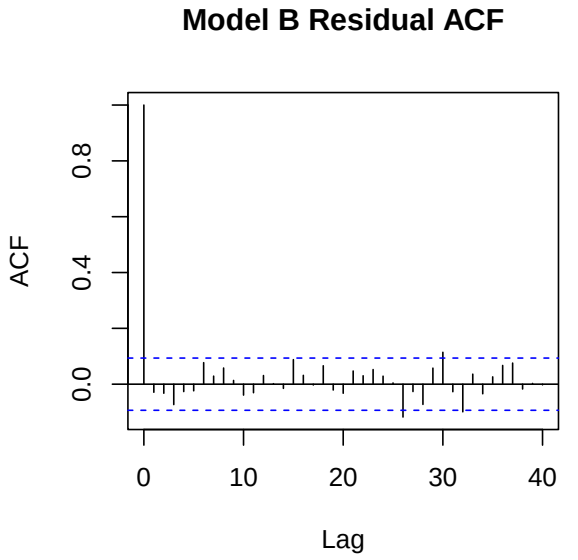
```
Box.test((res2)^ 2, lag = 21, type = c("Ljung-Box"), fitdf = 0)
```

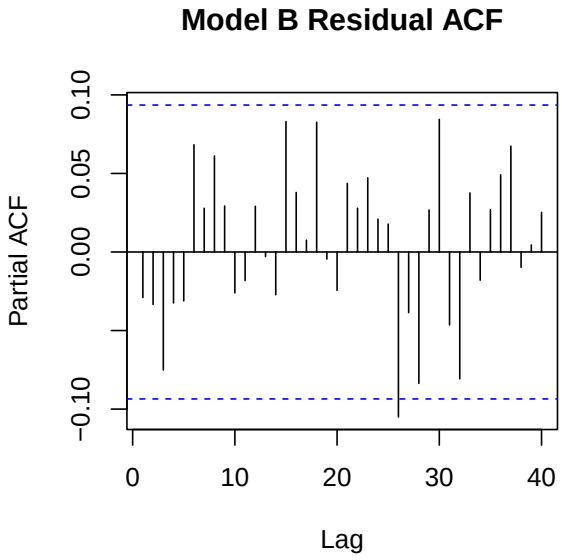
```
##
##  Box-Ljung test
##
## data:  (res2)^2
## X-squared = 12.481, df = 21, p-value = 0.9257
```

```
ar(res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```
##
## Call:
## ar(x = res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
##
## Order selected 0  sigma^2 estimated as  1686100
```

```
par(mfrow = c(1, 2))
acf(res2[c(1:length(res2))], lag.max = 40, main = "Model B Residual ACF")
pacf(res2[c(1:length(res2))], lag.max = 40, main = "Model B Residual ACF")
```




Model B passes all portmanteau tests, with p-values for the Shapiro-Wilk, Box-Pierce, Box-Ljung, and McLeod-Li tests being 0.1102, 0.3038, 0.2748, and 0.9257 respectively. When attempting to fit the residuals to an auto-regressive model through Yule-Walker estimation, we receive an AR(0) fit again. Looking at residual ACF and PACF, there are a few values that exceed the confidence interval. Once again, as R outputs conservative estimates for the confidence intervals, ACF and PACF values that barely exceed these limits can

be ignored. As model B has higher p-values for all portmanteau tests and has fewer significant ACF/PACF values in comparison to model A, model B will be used for forecasting.

## Forecasting Using Selected Model

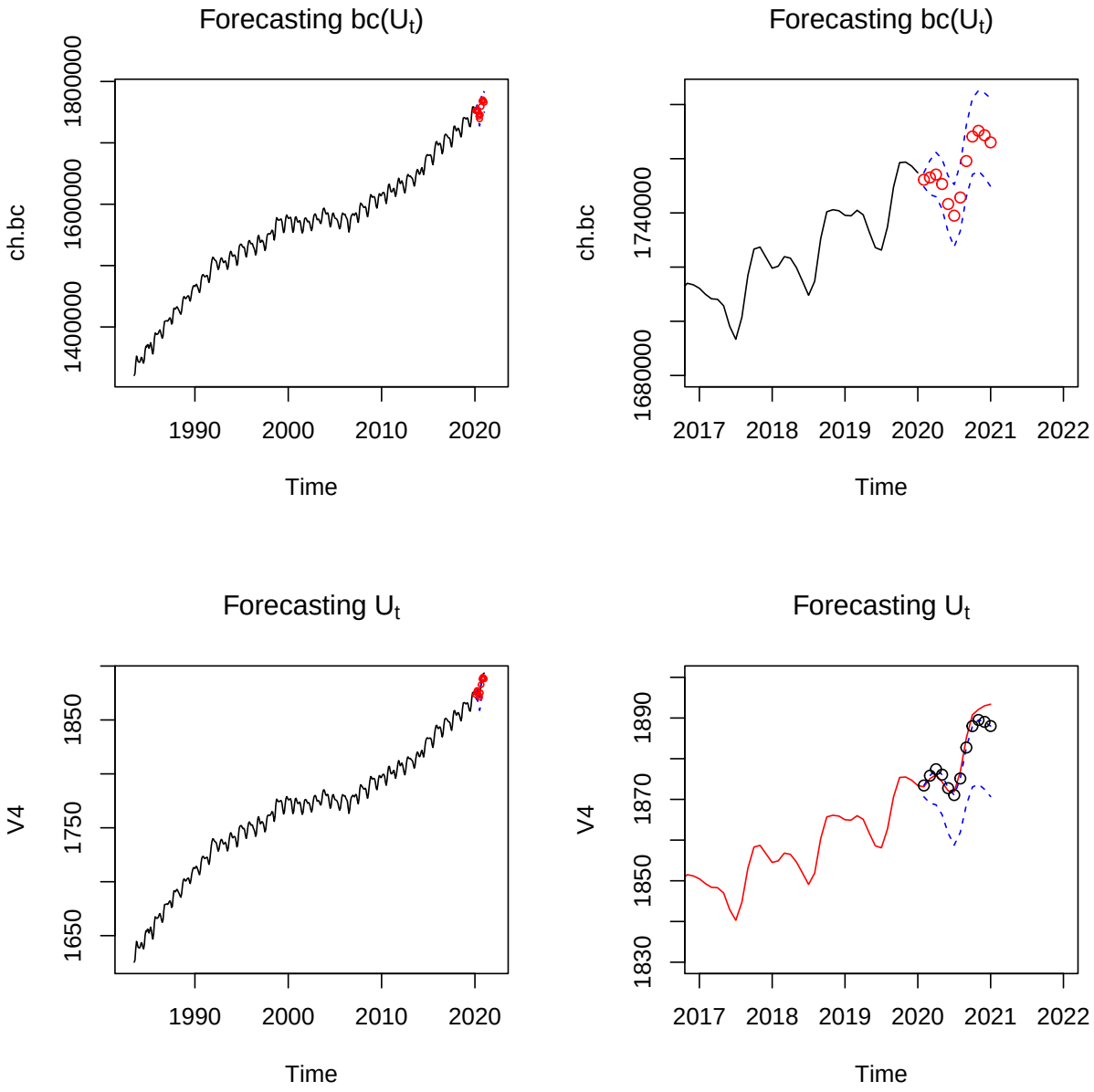



Table 4: Forecasts and Confidence Intervals

| | Feb 2020 | Mar 2020 | Apr 2020 | May 2020 | June 2020 | Jul 2020 | Aug 2020 | Sep 2020 | Oct 2020 | Nov 2020 | Dec 2020 | Jan 2021 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Predicted | 1873.4 | 1875.8 | 1877.4 | 1876.1 | 1872.8 | 1871.1 | 1875.1 | 1882.8 | 1888.0 | 1889.5 | 1889.0 | 1888.0 |
| Observed | 1873.1 | 1875.1 | 1876.0 | 1874.3 | 1872.1 | 1871.9 | 1876.8 | 1885.1 | 1890.8 | 1892.1 | 1893.0 | 1893.4 |
| Lower Bound | 1870.6 | 1869.1 | 1868.7 | 1866.3 | 1861.7 | 1858.8 | 1861.9 | 1868.6 | 1873.0 | 1873.7 | 1872.5 | 1870.7 |

| | Feb 2020 | Mar 2020 | Apr 2020 | May 2020 | June 2020 | Jul 2020 | Aug 2020 | Sep 2020 | Oct 2020 | Nov 2020 | Dec 2020 | Jan 2021 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Upper Bound | 1873.4 | 1875.8 | 1877.4 | 1876.1 | 1872.8 | 1871.1 | 1875.1 | 1882.8 | 1888.0 | 1889.5 | 1889.0 | 1888.0 |

Looking at our table, it appears that observations starting at July 2020 exceed the upper bound of the confidence interval. Yet, predicted values were relatively close to the upper bound of the confidence interval and therefore remained relatively close to observed values.

# 3. Conclusion

Monthly global atmospheric methane data collected by the NOAA was used for project analysis (Dlugokencky). The objectives of this project included three parts: proposing a SARIMA model, evaluating the model through diagnostic testing, and forecasting based off the proposed model to ascertain predictive validity. After conducting a Box-Cox transformation, ACF/PACF analysis resulted in a $SARIMA(2,1,3) \times (0,1,1)_{12}$ being selected. Maximum likelihood parameter estimation resulted in the following formula:

$$(1 + 0.6250_{(0.1499)}B + 0.5395_{(0.0745)}B^2)\nabla_1\nabla_{12}bc(U_t) =$$
$$(1 + 1.8660_{(0.1721)}B + 1.2115_{(0.3040)}B^2 + 0.2719_{(0.1465)}B^3)(1 - 0.8747_{(0.0449)}B^{12})Z_t; \sigma_Z^2 = 1684069 \quad (4)$$

Through examining polynomial roots, the model was deemed to be stationary, invertible, and causal. Analysis and portmanteau testing of model residuals all concluded that the model was acceptable. Lastly, forecasting based off of the model was conducted and these forecasted values were then untransformed. For seven out of the twelve values, observations were outside of the predicted confidence interval. Yet, these observed values remained relatively close to the forecasted values. The closeness between the upper bound of the confidence interval and forecasted values may be due to the extremeness of the Box-Cox transformation -a $\lambda = 2$ is the largest value a Box-Cox transformation will accept. In addition, the failure of the confidence interval to encapsulate all observed values may be due to an unexpected "surge" methane which was reported by the NOAA to be the largest in measurement history (Stein). Special thanks to Professor Raisa Feldman for instruction and guidance.

# 4. Works Cited

Dlugokencky, Ed. “Global Monitoring Laboratory - Carbon Cycle Greenhouse Gases.” NOAA ESRL Global Monitoring Laboratory, NOAA/GML, 7 June 2021, gml.noaa.gov/ccgg/trends_ch4/.

R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Stein, Theo. "Despite Pandemic Shutdowns, Carbon Dioxide and Methane Surged in 2020 - Welcome to NOAA Research." Welcome to NOAA Research, NOAA, 7 Apr. 2021, research.noaa.gov/article/ArtMID/587/ArticleID/2742/Despite-pandemic-shutdowns-carbon-dioxide-and-methane-surged-in-2020.

# 5. Appendix

```
knitr::include_graphics("/Users/Alienware R5/Desktop/PSTAT 174/PSTAT 174 Final Project/UC-Santa-Barbara-
knitr::opts_chunk$set(cache = T,
                      fig.align = 'center',
                      warnings= F,
                      fig.height = 4,
                      fig.width = 8)

library(dplyr)
library(forecast)
library(ggplot2)
library(ggfortify)
library(MASS)
library(pander)
library(tsdl)
library(latex2exp)
library(MuMIn)
# Load data
ch4 <- read.table("ch4_mm_gl.txt", skip = 52, sep = "")

ch4.ts <- ch4 %>%
  dplyr::select(V4) %>%
  ts(start = c(1983, 7), frequency = 12)

# Partition data
ch <- ts(ch4.ts[c(1:(length(ch4.ts) - 12))], start = c(1983, 7), frequency = 12)
ch.test <- ts(ch4.ts[c((length(ch4.ts) - 11):length(ch4.ts))], start = c(2020, 2), frequency = 12)

# Plot entire data
n <- length(ch4.ts)
fstart <- attributes(ch4.ts)$tsp[1]
fstop <- attributes(ch4.ts)$tsp[2]
fit <- lm(ch4.ts ~ seq(fstart, fstop, by = 1/12))

plot.ts(ch4.ts, main = expression("Complete Monthly Global CH"[4]),
        ylab = expression("CH"[4]*" ppm"),
        xlab = "Years")
abline(fit, col = "red")
abline(h = mean(ch4.ts), col = "blue")
legend("topleft", legend = c( "Trendline", "Mean line"),
       col = c("red", "blue"), lty = 1)

par(mfrow = c(1, 2))
# Plot train data
n2 <- length(ch)
fstart2 <- attributes(ch)$tsp[1]
fstop2 <- attributes(ch)$tsp[2]
fit2 <- lm(ch ~ seq(fstart2, fstop2, by = 1/12))

plot.ts(ch, main = expression("Training Monthly Global CH"[4]*" (U"[t]*")"),
        ylab = expression("CH"[4]*" ppm"),
        xlab = "Years")
```

```r
abline(fit2, col = "red")
abline(h = mean(ch), col = "blue")
legend("topleft", legend = c( "Trendline", "Mean line"),
       col = c("red", "blue"), lty = 1)

# Plot test data
n3 <- length(ch.test)
fstart3 <- attributes(ch.test)$tsp[1]
fstop3 <- attributes(ch.test)$tsp[2]
fit3 <- lm(ch.test ~ seq(fstart3, fstop3, by = 1/12))

plot.ts(ch.test, main = expression("Test Monthly Global CH"[4]),
        ylab = expression("CH"[4]*" ppm"),
        xlab = "Years")
abline(fit3, col = "red")
abline(h = mean(ch.test), col = "blue")
legend("topleft", legend = c( "Trendline", "Mean line"),
       col = c("red", "blue"), lty = 1)

# Table of trendline slopes
slopes <- list(fit$coefficients[2], fit2$coefficients[2], fit3$coefficients[2]) %>%
  as.data.frame()

colnames(slopes) <- c("Complete fit", "Training fit", "Test fit")
rownames(slopes) <- c("Slope")

knitr::kable(slopes, caption = "Table of Trendline Slopes", digits = 4)

# Histogram
ch1 <- ch4.ts[c(1:(length(ch4.ts) - 12))]
par(mfrow = c(1, 2))
hist(ch1, col = "light blue", main = expression("Histogram of U"[t]),
     xlab = expression("CH"[4]*" ppm"))
acf(ch1, lag.max = 40, main = expression("ACF of U"[t]), ylim = c(0.6, 1))

par(mfrow = c(1, 3))

# Box-Cox Transformation
bcTransform <- boxcox(ch ~ as.numeric(1:length(ch)))
lambda <- bcTransform$x[which(bcTransform$y == max(bcTransform$y))]

ch.bc <- ((ch ^ lambda) - 1) / lambda

# Paired histograms and ts plots
plot.ts(ch.bc, main = expression("bc(U"[t]*")"),
        ylab = expression("Transformed CH"[4]*" ppm"),
        xlab = "Years")
plot.ts(ch, main = expression("U"[t]),
        ylab = expression("CH"[4]*" ppm"),
        xlab = "Years")

# Comparing transformed hist
par(mfrow = c(1, 2))
```

```r
hist(ch1, col = "light blue", main = expression("Histogram of U"[t]),
     xlab = expression("CH"[4]*" ppm"))
hist(ch.bc, col = "light blue", main = expression("Histogram of bc(U"[t]*")"),
     xlab = expression("CH"[4]*" ppm"))

# Decomposing and plotting
decomp <- decompose(ch.bc)
plot(decomp)

# Comparing ts plots
par(mfrow = c(2, 2))

# Original Plot
plot.ts(ch.bc, main = TeX("$bc(U_t)$"))

# Differencing at lag 12
ch_12 <- diff(ch.bc, lag = 12)
fit_12 <- lm(ch_12 ~ seq(fstart2, fstop2 - 1, by = 1/12))

plot.ts(ch_12, main = TeX("$\\nabla_{12} bc(U_t)$"))
abline(fit_12, col = "red")
abline(h = mean(ch_12), col = "blue")

# Differencing at lag 1
ch_121 <- diff(ch_12, lag = 1)
fit_121 <- lm(ch_121 ~ seq(fstart2, fstop2 - 13/12, by = 1/12))

plot.ts(ch_121, main = TeX("$\\nabla_1 \\nabla_{12} bc(U_t)$"))
abline(fit_121, col = "red")
abline(h = mean(ch_121), col = "blue")

# Differencing at lag 1 again
ch_1211 <- diff(ch_121, lag = 1)
fit_1211 <- lm(ch_1211 ~ seq(fstart2, fstop2 - 14/12, by = 1/12))

plot.ts(ch_1211, main = TeX("$\\nabla_1 \\nabla_1 \\nabla_{12} bc(U_t)$"))
abline(fit_1211, col = "red")
abline(h = mean(ch_1211), col = "blue")

variances <- list(var(ch.bc), var(ch_12), var(ch_121), var(ch_1211)) %>%
  as.data.frame()
rownames(variances) <- c("Variance")

knitr::kable(variances, caption = "Variance of Differenced Data",
             col.names = c("$BC(U_t)$", "$\\nabla_{12} bc(U_t)$", "$\\nabla_1 \\nabla_{12} bc(U_t)$", "$
             digits = 4)

# Comparing ACF plots
par(mfrow = c(1, 4))
n <- length(ch.bc)
acf(ch.bc[c(1:n)], lag.max = 40, main = TeX("$bc(U_t)$"))
acf(ch_12[c(1:(n-12))], lag.max = 40, main = TeX("$ \\nabla_{12} bc(U_t)$"))
acf(ch_121[c(1:(n-13))], lag.max = 40, main = TeX("$\\nabla_1 \\nabla_{12} bc(U_t)$"))
```

```r
acf(ch_1211[c(1:(n - 14))], lag.max = 40, main = TeX("$\\nabla_1 \\nabla_1 \\nabla_{12} bc(U_t)$"))

# Comparing histograms
curve_gen <- function(data) {
  m <- mean(data)
  std <- sqrt(var(data))
  curve(dnorm(x, m, std), add= TRUE)
}

par(mfrow = c(1, 4))
hist(ch.bc, density = 20, breaks = 20, col = "blue", xlab = "", main = TeX("$bc(U_t)$"), prob = TRUE)
curve_gen(ch.bc)
hist(ch_12, density = 20, breaks = 20, col = "blue", xlab = "", main = TeX("$\\nabla_{12} bc(U_t)$"), pr
curve_gen(ch_12)
hist(ch_121, density = 20, breaks = 20, col = "blue", xlab = "", main = TeX("$\\nabla_1 \\nabla_{12} bc
curve_gen(ch_121)
hist(ch_1211, density = 20, breaks = 20, col = "blue", xlab = "", main = TeX("$\\nabla_1 \\nabla_1 \\nab
curve_gen(ch_1211)

# Plotting acf and pacf
par(mfrow = c(1, 2))
acf(ch_121[c(1:(n-13))], lag.max = 40, main = TeX("ACF for $\\nabla_1 \\nabla_{12} bc(U_t)$"))
pacf(ch_121[c(1:(n-13))], lag.max = 40, main = TeX("PACF for $\\nabla_1 \\nabla_{12} bc(U_t)$"))

# Model fitting
mod1 <- arima(ch.bc, order = c(2, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod1
mod2 <- arima(ch.bc, order = c(3, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod2
mod3 <- arima(ch.bc, order = c(3, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod3
mod4 <- arima(ch.bc, order = c(2, 1, 2), seasonal = list(order = c(0, 1, 1), period = 12), method = "ML"
mod4
# Generate table of models
aics <- c(mod1$aic, mod2$aic, mod3$aic, mod4$aic)
coef1 <- c(mod1$coef)
coef1 <- append(coef1, NA, 2)
coef1 <- list(coef1[1], coef1[2], coef1[3], coef1[4], coef1[5], coef1[6], coef1[7])

coef2 <- c(mod2$coef)

coef3 <- c(mod3$coef)
coef3 <- append(coef3, NA, 5)

coef4 <- c(mod4$coef)
coef4 <- append(coef4, NA, 2)
coef4 <- append(coef4, NA, 5)

models <- as.data.frame(coef1)
models <- models %>%
  rbind(coef2, coef3, coef4) %>%
  cbind(aics)
rownames(models) <- c("SARIMA(2, 1, 3) x (0, 1, 1)_12", "SARIMA(3, 1, 3) x (0, 1, 1)_12",
```

```
                      "SARIMA(3, 1, 2) x (0, 1, 1)_12", "SARIMA(2, 1, 2) x (0, 1, 1)_12")

options(knitr.table.NA = "")

knitr::kable(models, caption = "Model Estimates and AICs",
             col.names = c("$\\phi_1$", "$\\phi_2$", "$\\phi_3$", "$\\theta_1$", "$\\theta_2$", "$\\the

# plot model A roots
source("plot.roots.R")
par(mfrow = c(1, 2))
plot.roots(NULL, polyroot(c(1, 1.5687, 0.6633)), main = "Model A: Nonseasonal MA Roots")
plot.roots(NULL, polyroot(c(1, 0.3668, 0.4103)), main = "Model A: Nonseasonal AR Roots")

# plot model B roots
par(mfrow = c(1, 2))
plot.roots(NULL, polyroot(c(1, 1.8660,  1.2115,  0.2719)), main = "Model B: Nonseasonal MA Roots")
plot.roots(NULL, polyroot(c(1, 0.6250,  0.5395)), main = "Model B: Nonseasonal AR Roots")

# model A residual plots
par(mfrow = c(1, 3))
res1 <- residuals(mod4)
res_m <- mean(res1)
res_std <- sqrt(var(res1))

hist(res1, density = 20, breaks = 20, col = "blue", xlab = "", prob = TRUE, main = "Model A: Hist of Res
curve(dnorm(x, res_m, res_std), add = TRUE)

qqnorm(res1, main = "Q-Q Plot for Model A")
qqline(res1, col = "blue")

plot.ts(res1, main= "Model A Residuals")

# model B residual plots
par(mfrow = c(1, 3))
res2 <- residuals(mod1)
res_m <- mean(res2)
res_std <- sqrt(var(res2))

hist(res2, density = 20, breaks = 20, col = "blue", xlab = "", prob = TRUE, main = "Model B: Hist of Res
curve(dnorm(x, res_m, res_std), add = TRUE)

qqnorm(res2, main = "Q-Q Plot for Model B")
qqline(res2, col = "blue")

plot.ts(res2, main= "Model B Residuals")

# Model A testing
# length(res2) ^ 0.5 = 20.95233 which is rounded to 21
shapiro.test(res1)
Box.test(res1, lag = 21, type = c("Box-Pierce"), fitdf = 5)
Box.test(res1, lag =21, type = c("Ljung-Box"), fitdf = 5)
Box.test((res1)^ 2, lag = 21, type = c("Ljung-Box"), fitdf = 0)
ar(res1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
```

```r
par(mfrow = c(1, 2))
acf(res1[c(1:length(res1))], lag.max = 40, main = "Model A Residual ACF")
pacf(res1[c(1:length(res1))], lag.max = 40, main = "Model A Residual PACF")

# Model B testing
shapiro.test(res2)
Box.test(res2, lag = 21, type = c("Box-Pierce"), fitdf = 6)
Box.test(res2, lag =21, type = c("Ljung-Box"), fitdf = 6)
Box.test((res2)^ 2, lag = 21, type = c("Ljung-Box"), fitdf = 0)
ar(res2, aic = TRUE, order.max = NULL, method = c("yule-walker"))

par(mfrow = c(1, 2))
acf(res2[c(1:length(res2))], lag.max = 40, main = "Model B Residual ACF")
pacf(res2[c(1:length(res2))], lag.max = 40, main = "Model B Residual ACF")

# Forecasting and plotting transformed data
par(mfrow = c(1, 2))
pred.tr <- predict(mod4, n.ahead = 12)
U.tr <- pred.tr$pred + 2 * pred.tr$se
L.tr <- pred.tr$pred - 2 * pred.tr$se
plot.ts(ch.bc, xlim = c(1983, 2022), ylim = c(min(ch.bc), max(U.tr)), main = TeX("Forecasting $bc(U_t)$
lines(U.tr, col = "blue", lty = "dashed")
lines(L.tr, col = "blue", lty = "dashed")
points(seq(2020 + 1/12, 2020 + 1, by = 1/12), pred.tr$pred, col = "red", cex = 0.5)

plot.ts(ch.bc, xlim = c(2017, 2022), ylim = c(1680000, max(U.tr)), main = TeX("Forecasting $bc(U_t)$"))
lines(U.tr, col = "blue", lty = "dashed")
lines(L.tr, col = "blue", lty = "dashed")
points(seq(2020 + 1/12, 2020 + 1, by = 1/12), pred.tr$pred, col = "red")

# Forecasting and plotting untransformed data
par(mfrow = c(1, 2))

pred.orig <- (pred.tr$pred * lambda + 1) ^ (1/lambda)
U <- (U.tr * lambda + 1) ^ (1/lambda)
L <- (L.tr * lambda + 1) ^ (1/lambda)

plot.ts(ch4.ts, xlim = c(1983, 2022), ylim = c(min(ch4.ts), max(U)), col = "black", main = TeX("Forecas
lines(U, col = "blue", lty = "dashed")
lines(L, col = "blue", lty = "dashed")
points(seq(2020 + 1/12, 2020 + 1, by = 1/12), U, col = "red", cex = 0.5)

plot.ts(ch4.ts, xlim = c(2017, 2022), ylim = c(1830, 1900), col = "red", main = TeX("Forecasting $U_t$"
lines(U, col = "blue", lty = "dashed")
lines(L, col = "blue", lty = "dashed")
points(seq(2020 + 1/12, 2020 + 1, by = 1/12), U, col = "black")

# Generating table of forecasts and CI
paste <- as.numeric()
fore <- list(U[1], U[2], U[3], U[4], U[5], U[6], U[7], U[8], U[9], U[10], U[11], U[12])
fore <- fore %>%
  as.data.frame()
fore <- fore %>%
```

```
  rbind(ch.test, L, U)
rownames(fore) <- c("Predicted", "Observed", "Lower Bound", "Upper Bound")
colnames(fore) <- c("Feb 2020", "Mar 2020", "Apr 2020", "May 2020", "June 2020", "Jul 2020", "Aug 2020"

knitr::kable(fore, caption = "Forecasts and Confidence Intervals",
             digits = 1)
```