STA 137 Final Project

Zhuorui He, Shiyu Wu, Sulei Wang, Zhan Shi

2024-03-12

Contents

Introduction

This study analyzes the Central African Republic's annual exports as a percentage of GDP from 1960 to 2017, addressing the complexities of real-world data. It involves decomposing and transforming data, identifying optimal ARIMA parameters, and conducting residual diagnostics for model accuracy. The aim is to create an ARIMA model for accurately forecasting future exports.

Background

The Central African Republic (CAR), a landlocked nation at Africa's heart, ranked 183rd globally in exports in 2022 per the Observatory of Economic Complexity (OEC). Its economy, rich in natural resources, focuses on agriculture, services, and exporting minerals, oil, timber, and agricultural products, with gold and diamonds among its top exports. Despite agriculture being pivotal, its contribution to exports is lesser. From 1960 to 2017, export volumes rose, but their GDP percentage has been declining since 1968. Major importers include the UAE, Italy, Pakistan, China, and France.

However, CAR faces challenges like political instability, violence, and inadequate resource management, hindering sustainable growth and global market competitiveness. Reliance on international aid is critical due to slow growth and a rising population, with poverty affecting 65.7% of the populace by 2021.

Exploratory Data Analysis

Data Description

This dataset from the World Bank contains annual data on GDP, imports, exports (as a percentage of GDP), population, and GDP growth for the Central African Republic from 1960 to 2017. The main focus in this paper is on exports. Exports are presented as a percentage of GDP to reflect the value of exported goods and services relative to the GDP. The accompanying figure offers a concise view of the country's economic trends over these years.

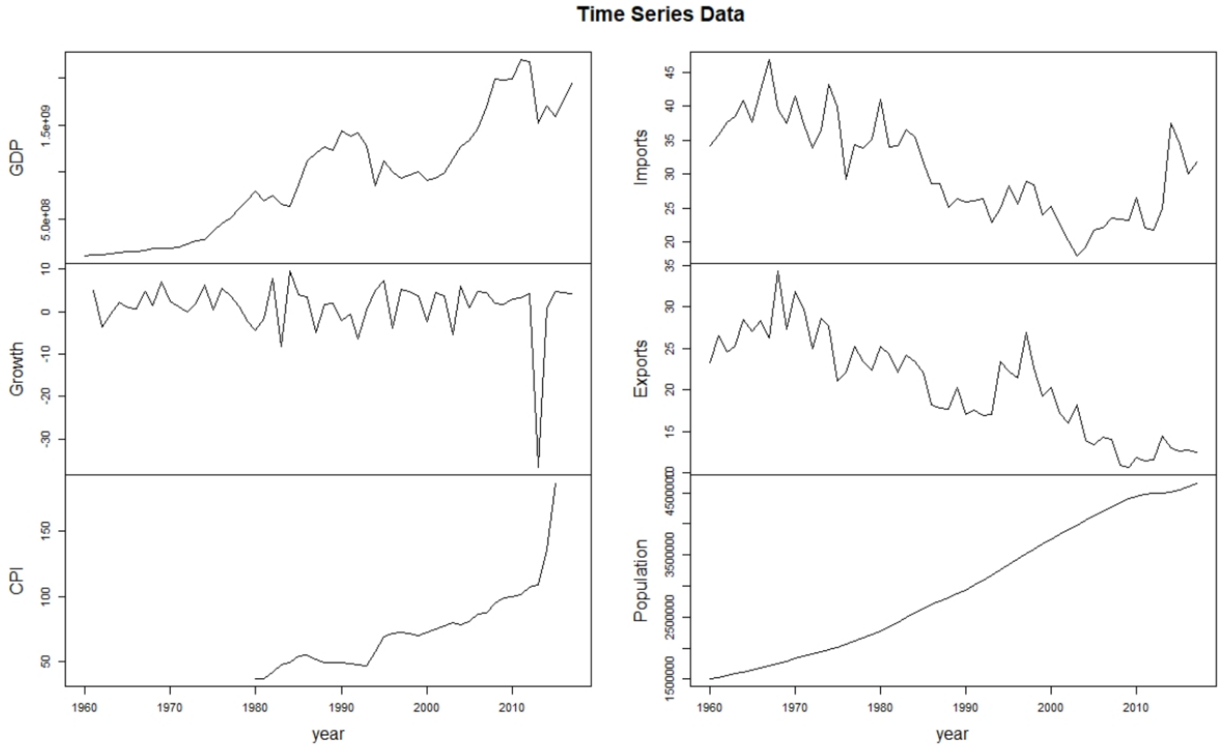
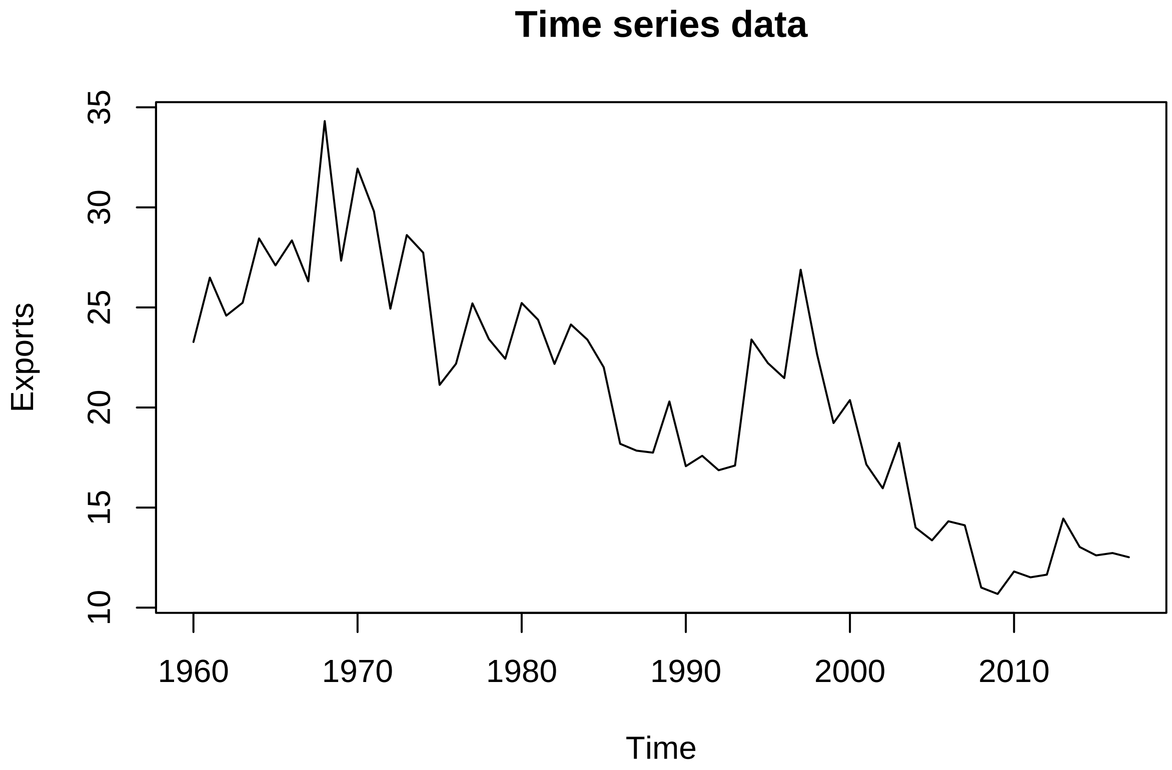
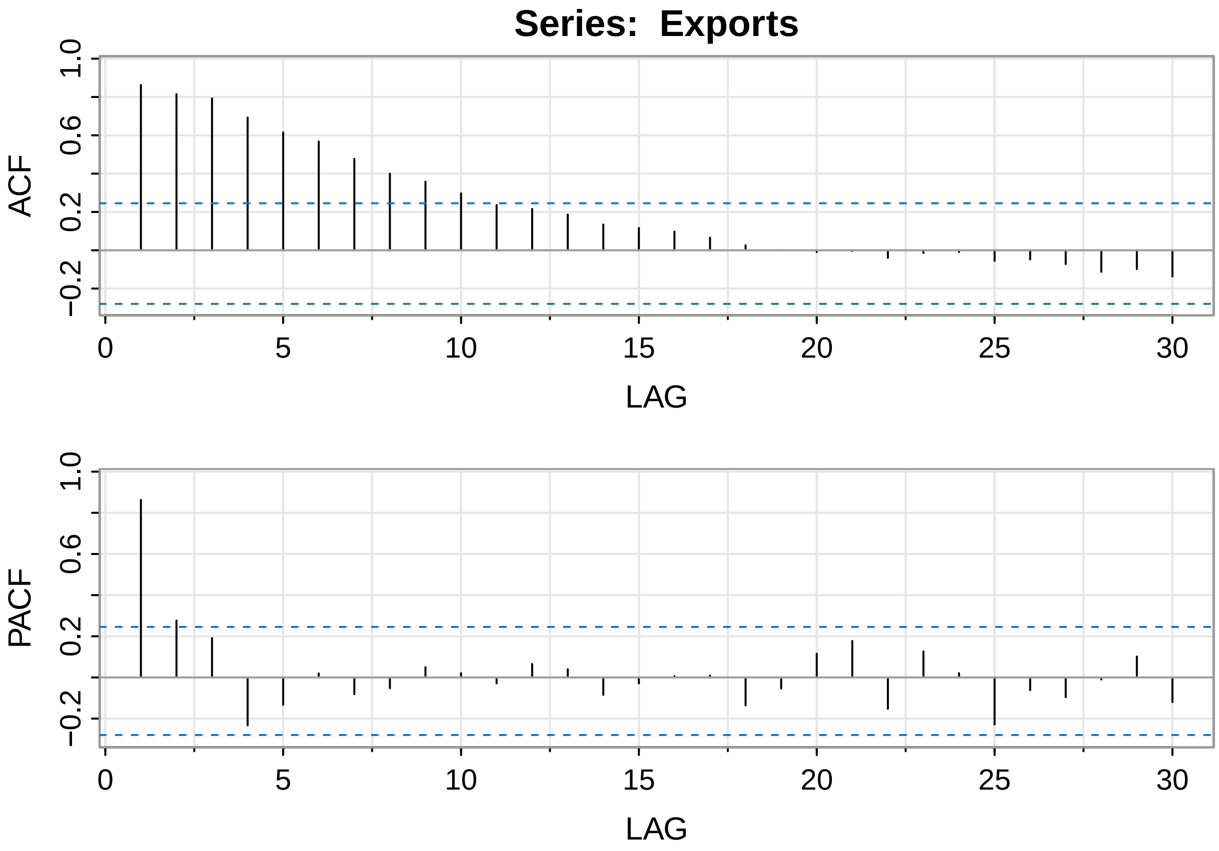


Figure 1: Time series of Gross Domestic Product (GDP), Growth rate, Consumer Price Index (CPI), Imports, Exports and Population for Central African Republic.

Visualization

The preliminary analysis commenced with an examination of the time series data for Exports. A subtle downward trend observed in exports, as illustrated in the figure below, suggests the presence of a non-constant mean, indicating that the time series is non-stationary. Further investigation was conducted on the dataset's Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). These analyses are crucial for understanding the direct influence of past data points on future values and for determining the appropriate order of the time series model.





Inferential Analysis

ADF and KPSS Tests for Stationary

To numerically assess the dataset's stationarity, we used the Augmented Dickey-Fuller (ADF) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests, both with a 0.05 significance level. The ADF test identifies stationarity with p-values under 0.05, while KPSS does so with p-values above 0.05. Both tests rejected stationarity, confirming the dataset's non-stationary nature.

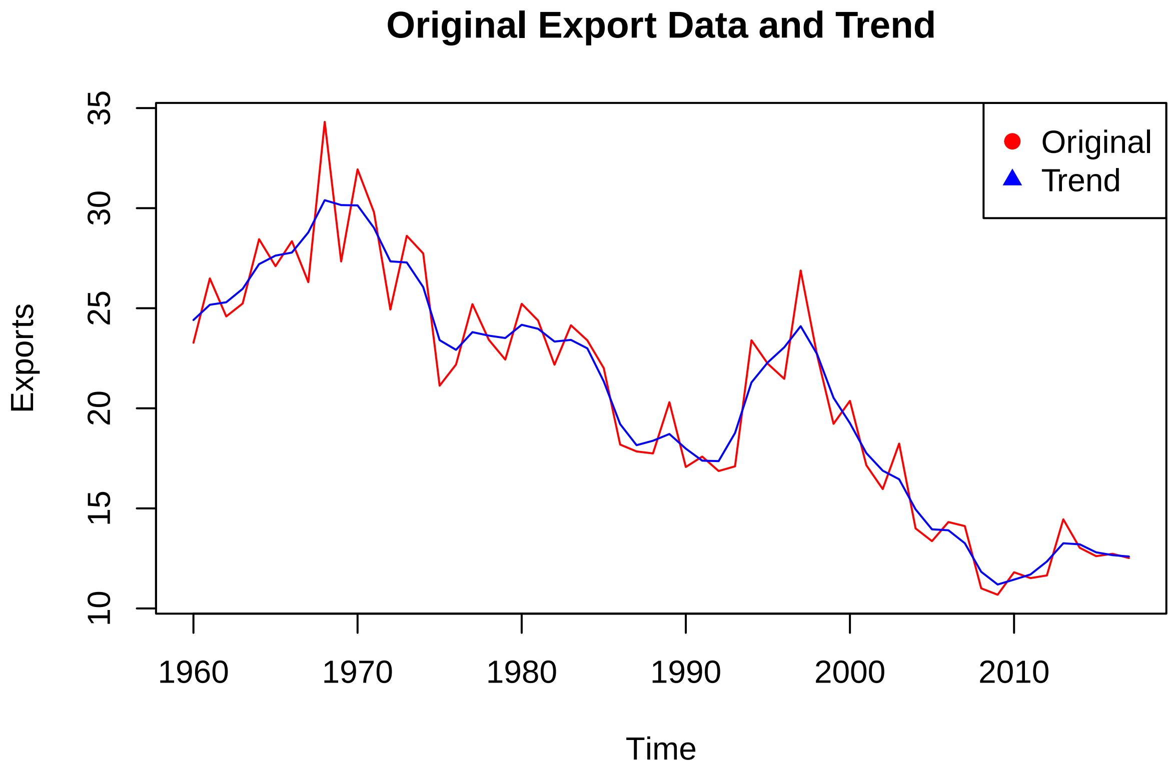
All our models are predicated on the foundation of stationary data, which imparts numerous characteristics and properties crucial for prediction. Consequently, transforming the data into a time-independent format is essential to harness these advantages effectively.

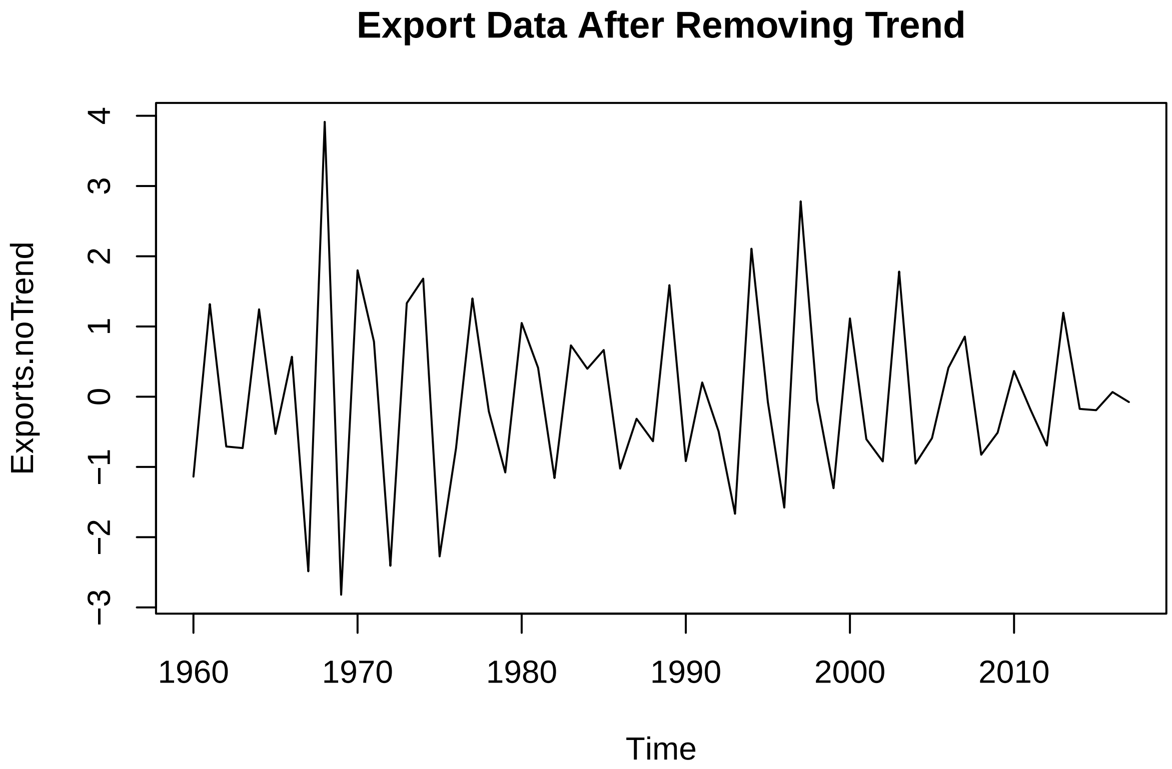
```
## [1] 0.1005534
```

```
## [1] 0.01
```

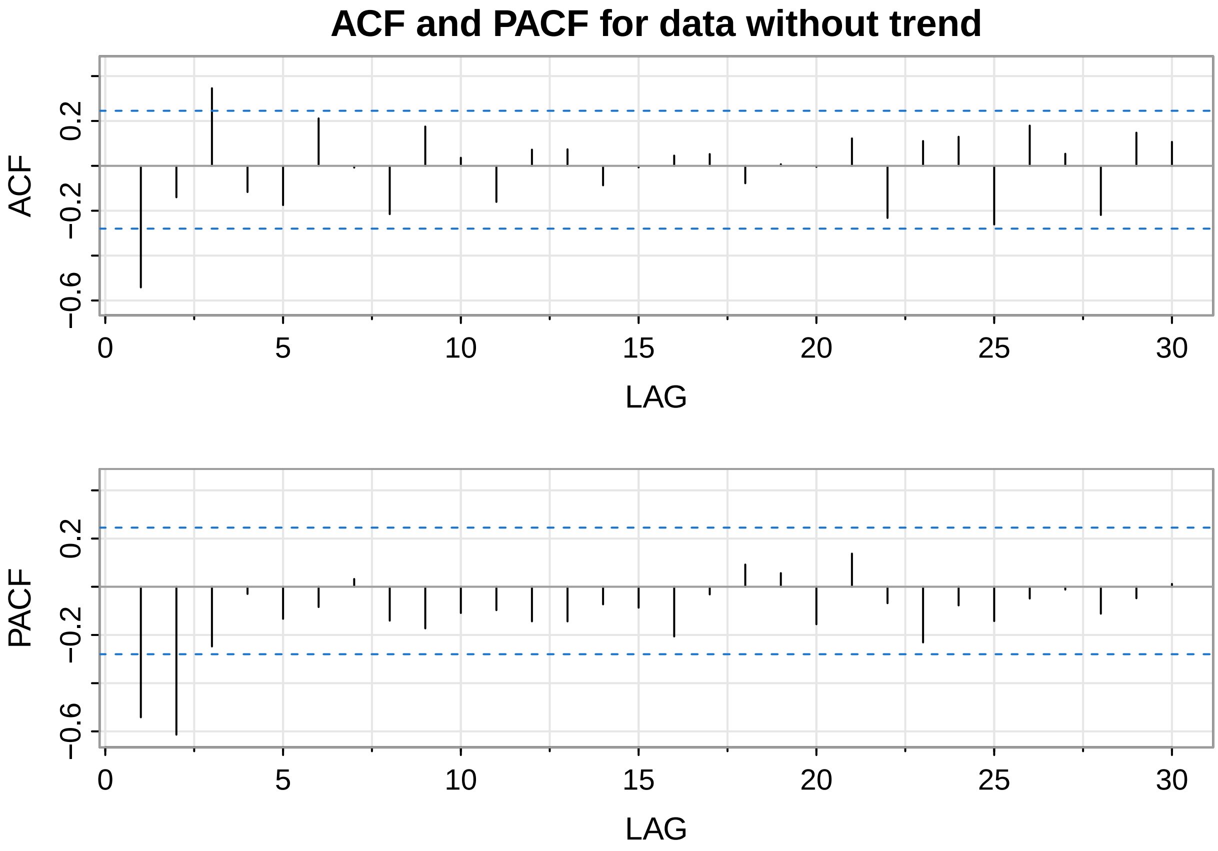
Decomposition

To tackle the dataset's non-stationarity, four detrending methods were explored. The first method, Kernel smoothing, was applied to remove the general decreasing trend, aiming to highlight the cyclical trend component, as shown in the figures below. This approach aimed to reduce the data to its cyclical variations, enabling a more targeted time series analysis.


Original Export Data and Trend
Exports
35
30
25
20
15
10
1960
1970
1980
1990
2000
2010
Time
● Original
▲ Trend



To delve deeper into the model exploration, calculations of the ACF and pacf were conducted. ACF and PACF are instrumental in identifying the lags with significant impact on the current value. They play a crucial role in delineating the pattern of the time series, thereby aiding in the selection of an appropriate model. The ACF revealed a very strong correlation at a lag of 1, while the PACF displayed two pronounced correlations at lags 1 and 2. These findings suggest the potential suitability of AR(2) and MA(1) models for the de-trended data.

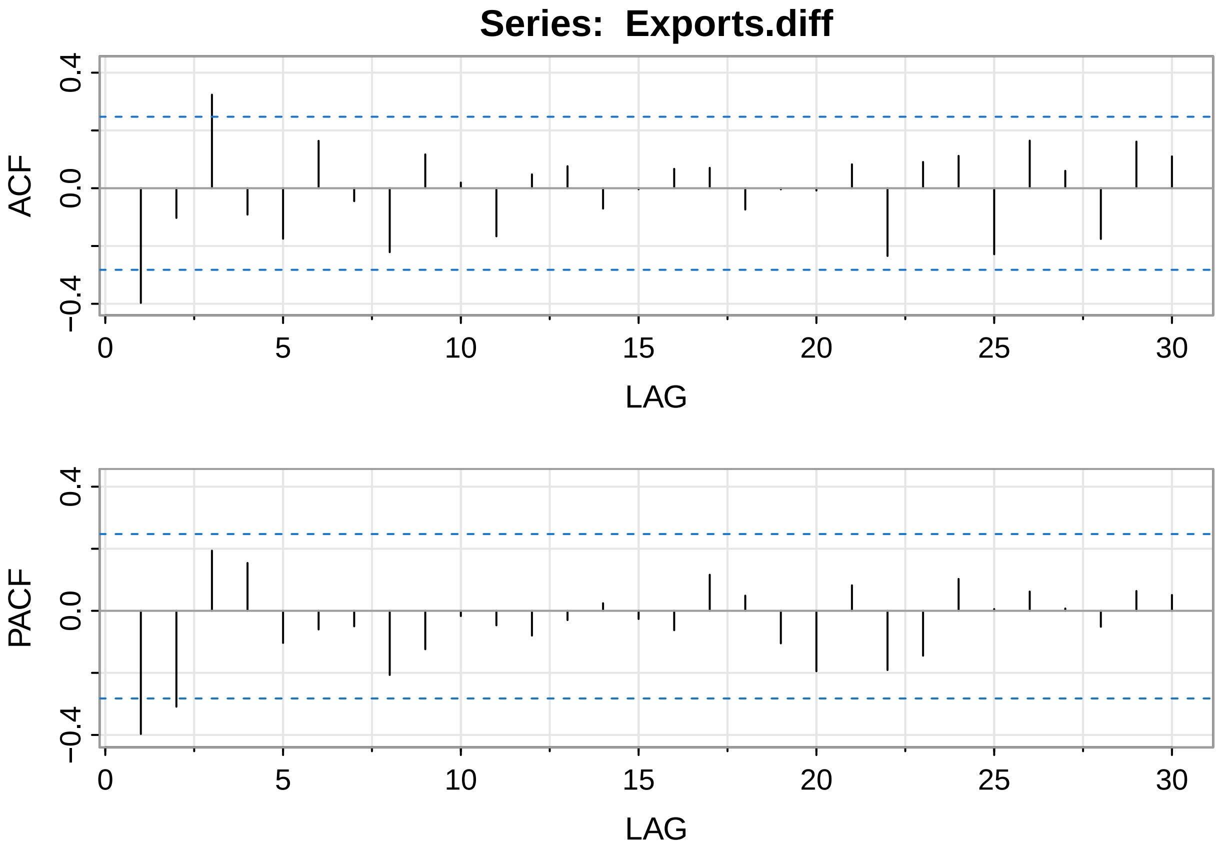


```
##
##  Augmented Dickey-Fuller Test
##
## data:  Exports.noTrend
## Dickey-Fuller = -5.4555, Lag order = 3, p-value = 0.01
## alternative hypothesis: stationary
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  Exports.noTrend
## KPSS Level = 0.032693, Truncation lag parameter = 3, p-value = 0.1
```

First-order Difference

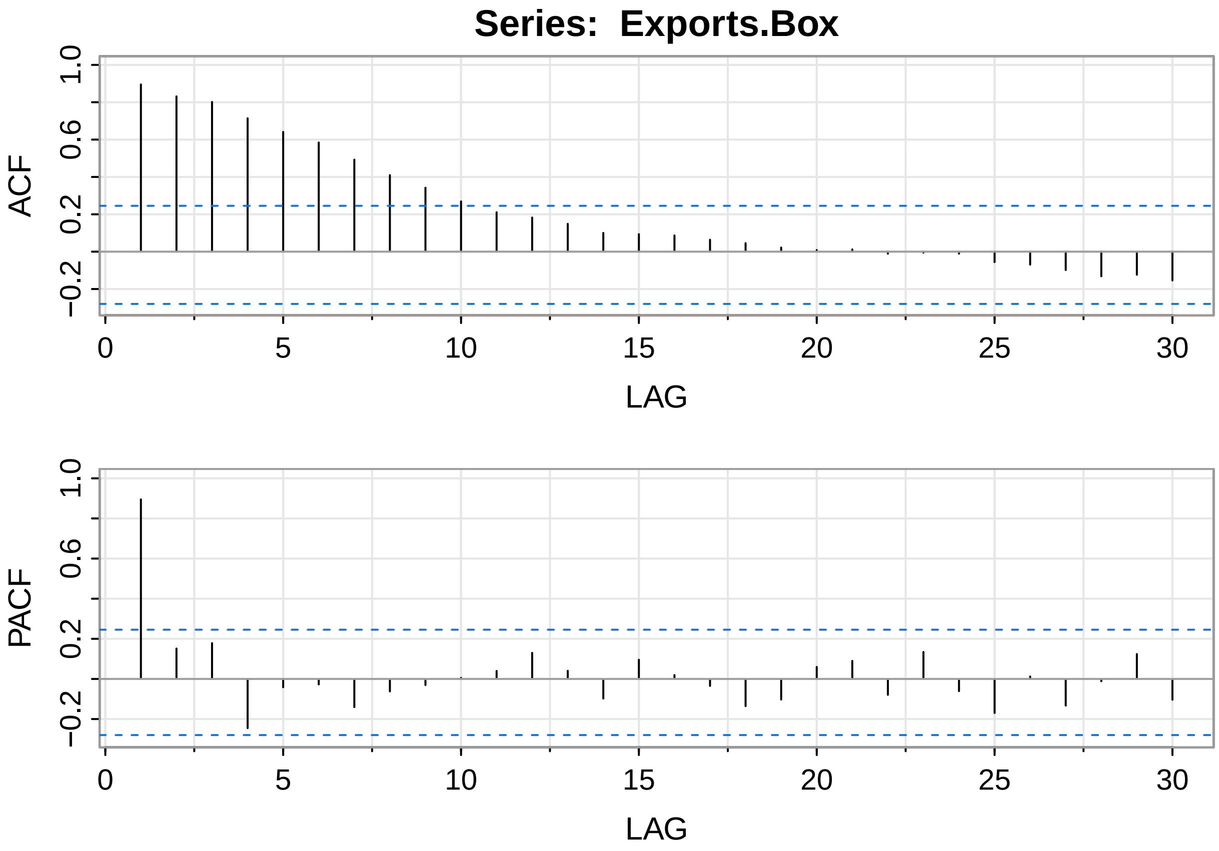
Another transformation attempt was made using the first-order difference, which similarly indicated the appropriateness of an AR(2), MA(1) model for the dataset. Despite this, the first-order difference transformation failed to pass the ADF test, indicating that it did not achieve stationarity. Given this outcome, the decision was made to abandon the first-order difference approach.

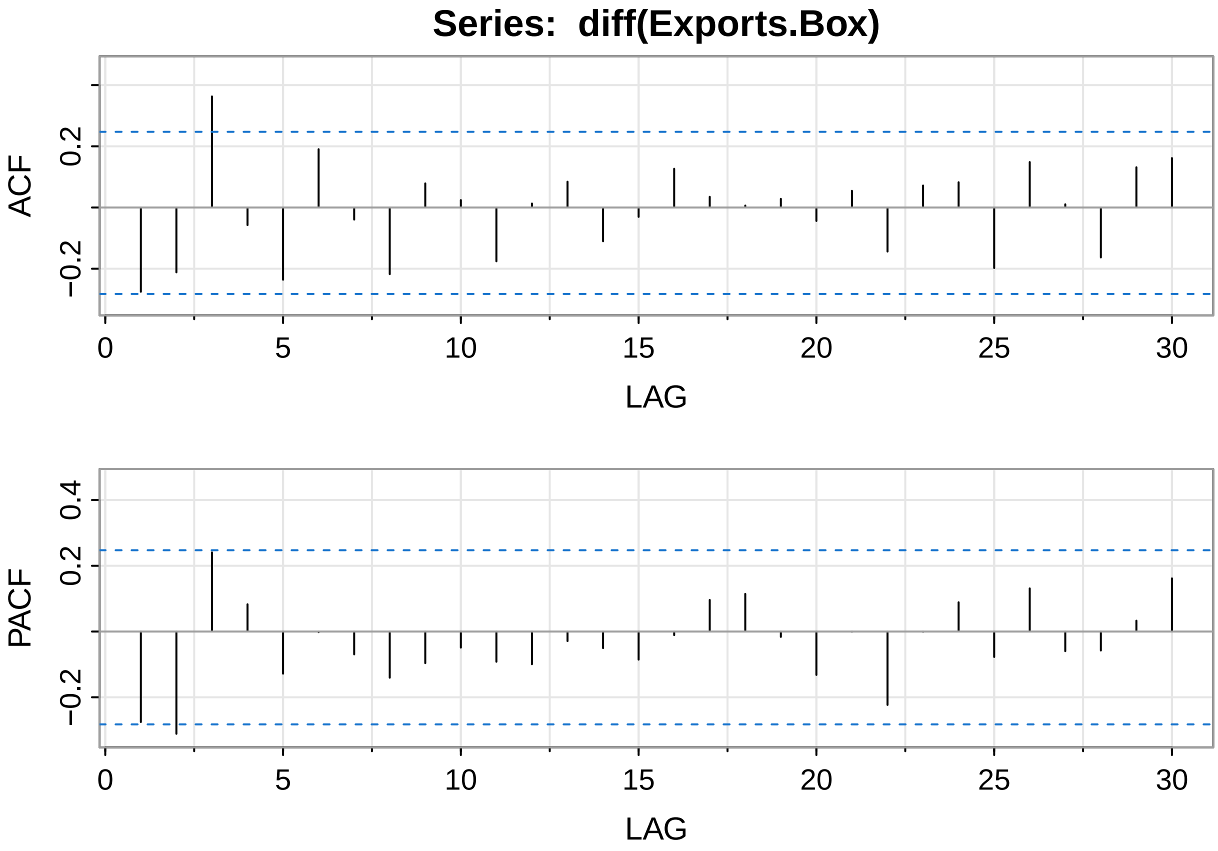


```
##
##  Augmented Dickey-Fuller Test
##
## data:  Exports.diff
## Dickey-Fuller = -3.2518, Lag order = 3, p-value = 0.08812
## alternative hypothesis: stationary

## Warning in kpss.test(Exports.diff): p-value greater than printed p-value

##
##  KPSS Test for Level Stationarity
##
## data:  Exports.diff
## KPSS Level = 0.092232, Truncation lag parameter = 3, p-value = 0.1
```

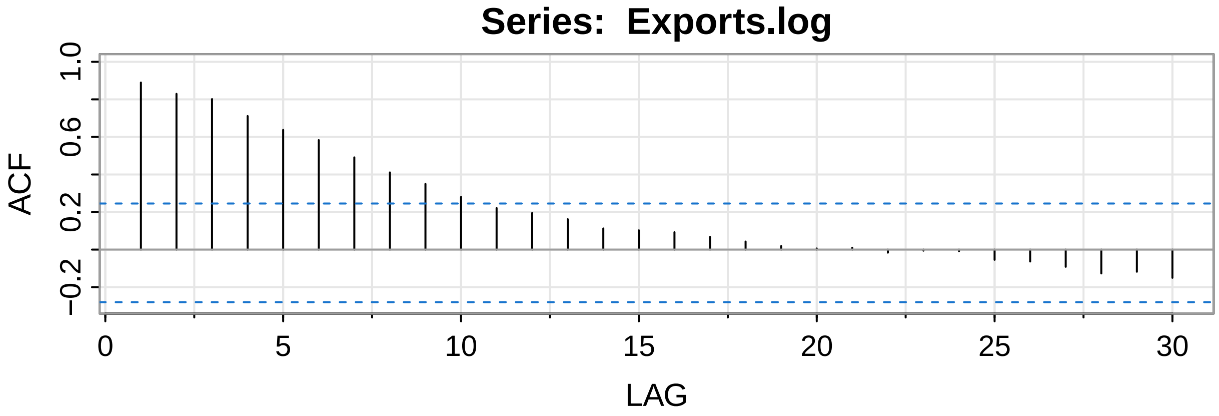
Box-Cox Transformation

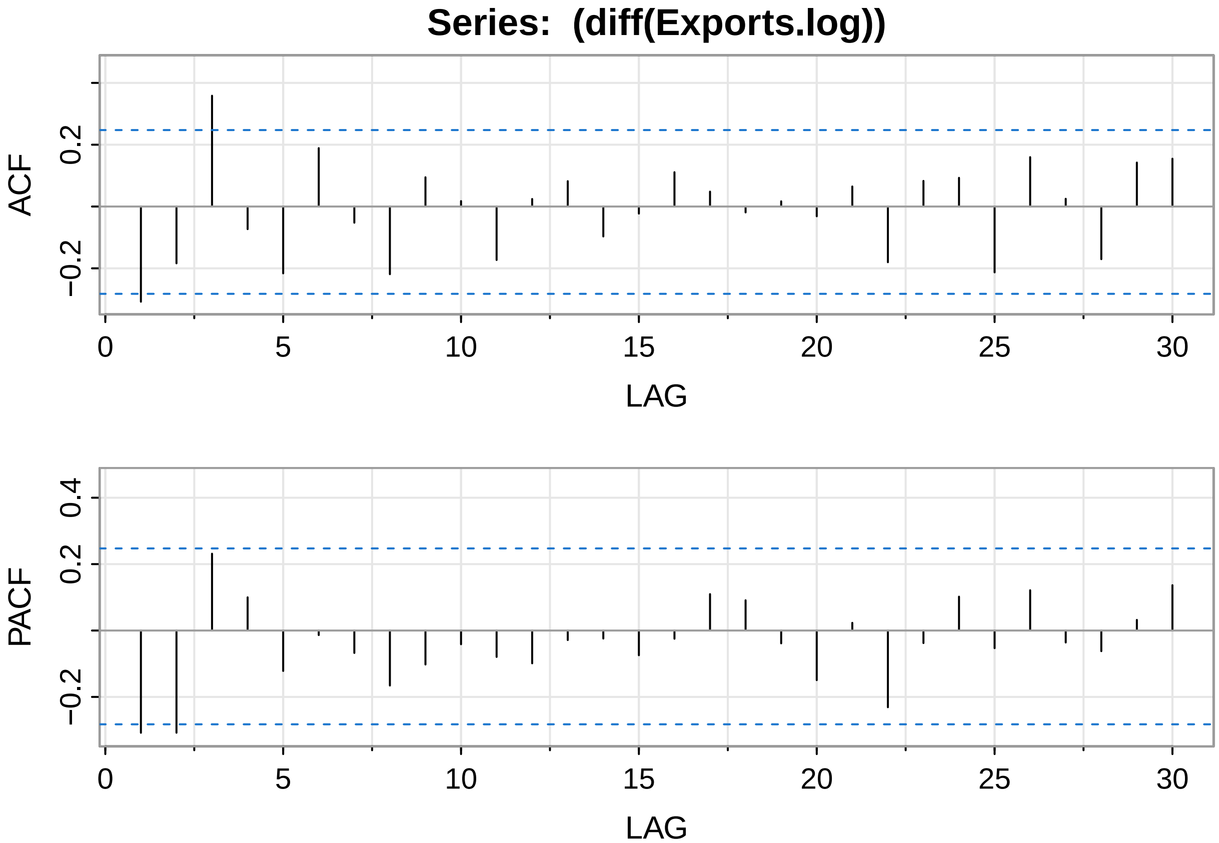
Exploring Box-Cox and Log Transformations to correct the time series' non-stationarity proved ineffective, as both methods led to persistently non-stationary series, shown in the figure below. Differencing these transformations only yielded ACF and PACF patterns resembling random, white noise series. Therefore, due to these findings, both transformations were excluded from further analysis.


Series: Exports.Box
ACF
−0.2
0.2
0.6
1.0
LAG
0
5
10
15
20
25
30
PACF
−0.2
0.2
0.6
1.0
LAG
0
5
10
15
20
25
30


Series: diff(Exports.Box)
ACF
0.2
−0.2
0
5
10
15
20
25
30
LAG
PACF
0.4
0.2
−0.2
0
5
10
15
20
25
30
LAG

Log Transformation





After thorough comparison and careful consideration, the decision was made to select the detrending method as the preferred approach for preparing our data for model fitting in the subsequent phase of analysis.

ARIMA Model

For the ARIMA(q,d,p) model

$$X'_t = \sum_{i=1}^{p} \alpha_p X'_{t-p} + \sum_{i=1}^{q} \theta_q \omega_{t-q}$$

where $X'_t = (1-B)^d X_t$

The model exhibits an order in the form of (p, d, q) where:

- p =The order of the Auto Regressive Model
- d = the order of differencing
- q = The order of the Moving Average

Following the detrending of the Exports dataset, the next step involved establishing the ARIMA model. To accurately determine the optimal parameters for the ARIMA(p,d,q) model, an automatic selection process was employed. This method suggested ARIMA(2,0,1) as the most fitting model for our data.

```
## Series: Exports.noTrend
## ARIMA(2,0,1) with zero mean
##
## Coefficients:
##           ar1      ar2      ma1
##       -0.5731  -0.4330  -0.5029
## s.e.   0.2122   0.1767   0.2619
##
## sigma^2 = 0.7146:  log likelihood = -71.79
## AIC=151.58   AICc=152.33   BIC=159.82
##
## Training set error measures:
##                      ME      RMSE       MAE      MPE     MAPE      MASE
## Training set 0.00429613 0.8231794 0.6252085 -15.1433 142.2494 0.3526863
##                     ACF1
## Training set 0.005845486
```

Model Comparison

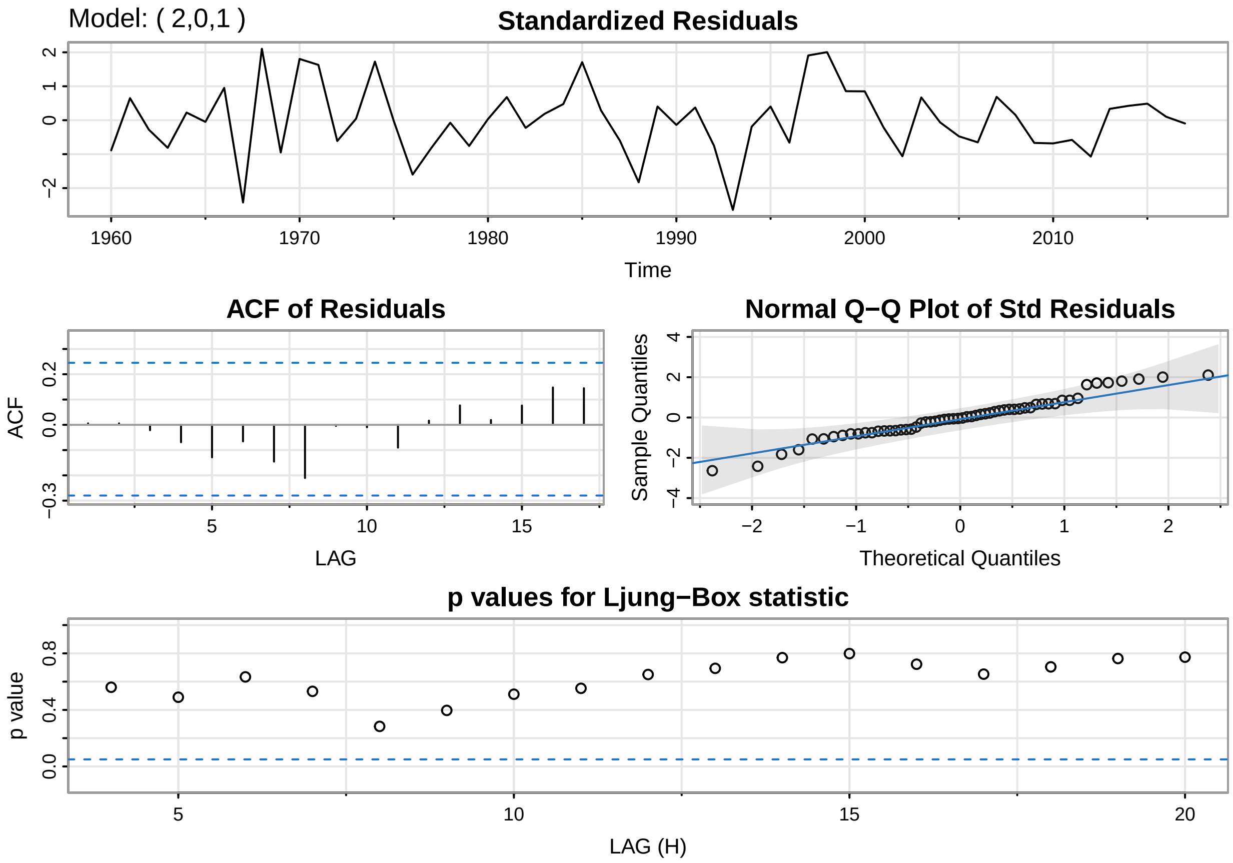
To discern the most effective model, Mean Squared Error (MSE) and Akaike Information Criterion (AIC) were employed as the primary metrics. MSE helps estimate the average of squared errors during model training, while AIC assesses both the prediction error and the extent of information loss during model fitting.

$$MSE = \frac{\sum(Residuals)^2}{n},\ AIC = n \times ln(\frac{\sum(Residuals)^2}{n}) + 2k$$

where n is the number of observations in the model, and k is the number of parameters.

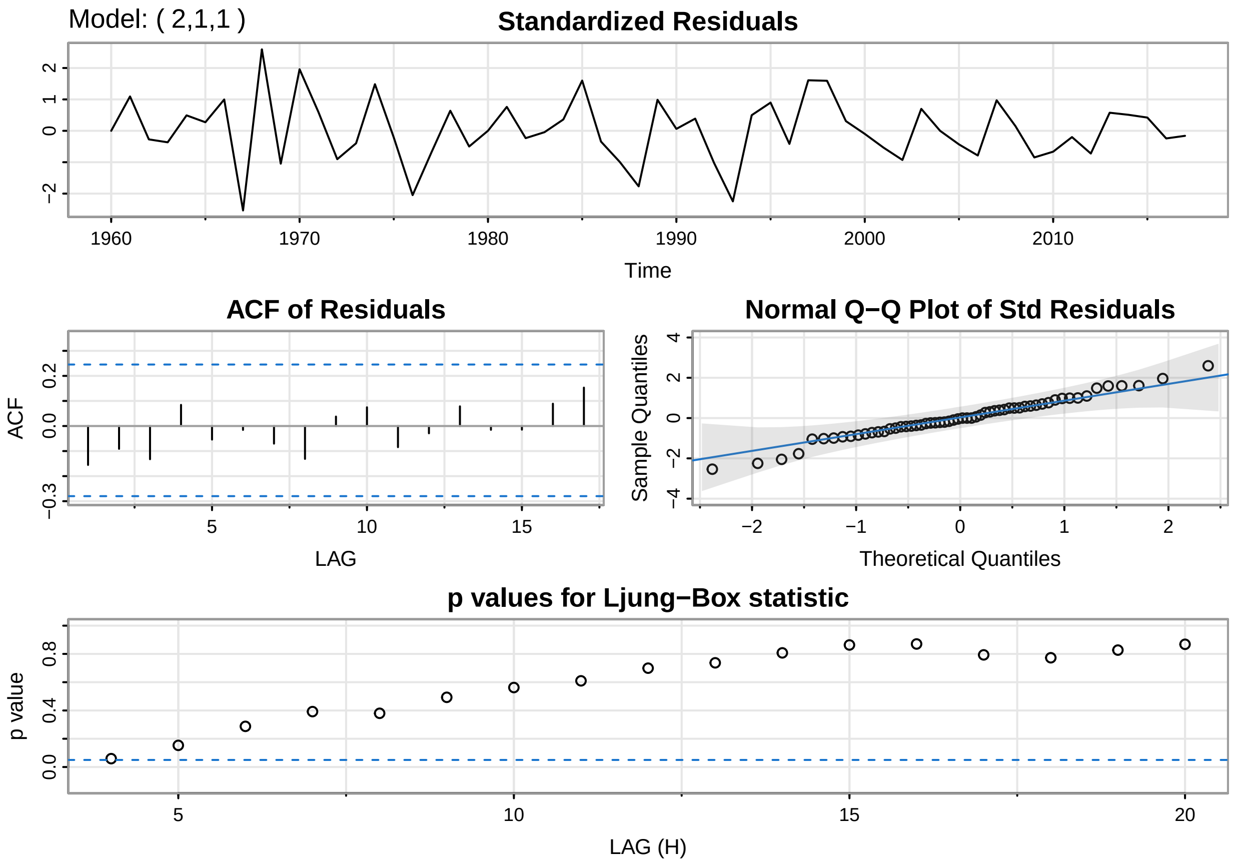
The SARIMA model was utilized for evaluation, showing that the ACF of residuals mostly fell within the expected confidence range, with the Q-Q plot revealing a good alignment of most points along the fit line, albeit with some evidence of heavy tails. Comparisons were made between ARIMA(2,1,1) and ARIMA(3,0,1), against the backdrop of ARIMA(2,0,1). While ARIMA(2,1,1) exhibited a lower MSE of 0.0128 compared to ARIMA(3,0,1)'s MSE of 0.0735 and ARIMA(2,0,1)'s MSE of 0.00430, it was disqualified due to its p-values falling below the significance line for lags less than 5, indicating autocorrelation and dependency within white noise, rendering it suboptimal.

Despite ARIMA(3,0,1) presenting a slightly better AIC of 2.544 against ARIMA(2,0,1)'s AIC of 2.613, the minimal difference and the preference for simplicity guided the choice towards ARIMA(2,0,1) for prediction purposes, with the specific equation provided in the following segment.



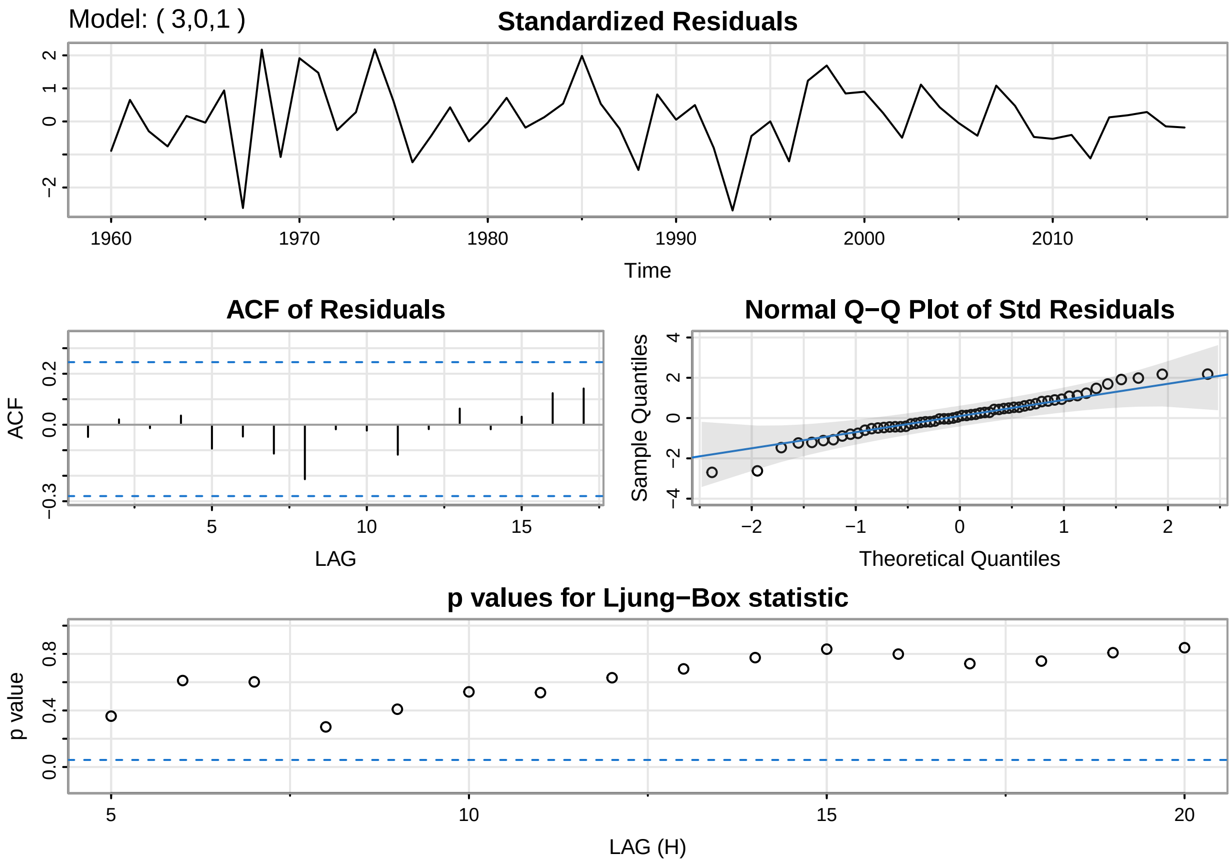
```
## [1] "AIC:"             "151.576013291681"
```

```
## [1] "Training MSE:"       "0.00429612956658058"
```



```
## [1] "AIC:"              "160.187603361086"
```

```
## [1] "Training MSE:"      "0.0128333615678964"
```



```
## [1] "AIC:"              "147.523620259954"
```

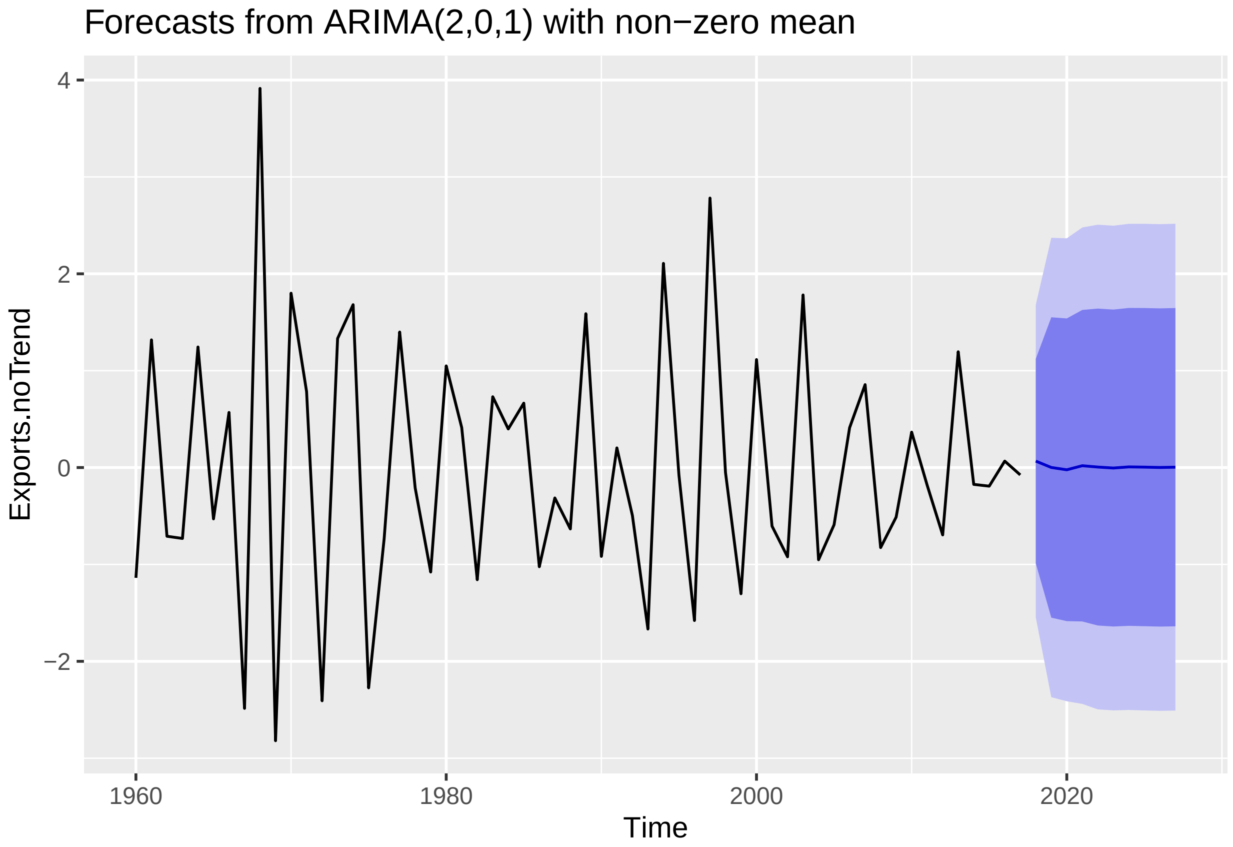
```
## [1] "Training MSE:"      "0.0735348695589228"
```

Forecast

After comparing models and conducting diagnostics, ARIMA(2,0,1) with detrended data was identified as the optimal choice. The model formula is

$$(1 - \phi_1 B - \phi_2 B^2)x_t = (1 + B)w_t$$

Utilizing this model, we forecasted the white noise in exports for the next six years with a significance level $\alpha = 0.05$.



The forecast depicted focuses solely on the random white noise component of exports over the next six years, not the actual future exports. To accurately forecast total exports, a linear fit and prediction of the trend using a separate linear model would be necessary. However, this paper concentrates primarily on analyzing the random noise within the time series, not on predicting the trend.

Conclusion

In this project, detrending the data was chosen as the necessary step to adhere to the stationary assumption and reduce errors, leading to the establishment of an ARIMA(2,0,1) model. However, the forecast process encounters a limitation due to its inability to predict future trends, offering only a 95% confidence interval for predictions that exclude any trend component. This constitutes a significant constraint of the study.

To overcome this and enhance the project, future efforts could include identifying methods to estimate trends for more inclusive confidence intervals. Introducing more data and alternative models like the Long Short-Term Memory (LSTM) model could also yield better results, including trend analysis, thus expanding the study's relevance and application.

Reference

(n.d.). Exports of goods and services (% of GDP) - Central African Republic. THE WORLD BANK. https://data.worldbank.org/indicator/NE.EXP.GNFS.ZS?locations=CF

Code Appendix

```r
# include=false will not performance our codes and results in the html file;
# echo=false will only show our outputs such as tables and plots
# import libraries and data here

library(ggplot2)
library(dplyr)
library(naniar)
library(astsa)
library(forecast)
library(TTR)
library(KernSmooth)
library(tseries)
load("finalproject.Rdata")
Exports<-ts(finalPro_data$Exports,start = 1960)
# Plot the Export as time series data
ts.plot(Exports,main = "Time series data")
acf2(Exports,max.lag=30)
sig_value <- 0.05 #significant value of test
# ADF: if p < sig, stationary
adf.test(Exports)$p.value
#KPSS: if p > sig, stationary
kpss.test(Exports)$p.value

# Both tests indicate non-stationary.
#Try kernal smoothing for trend
kernel.type <- "gaussian"
bandwidth <- dpill(time(Exports), Exports)
smoothed_values <- ksmooth(x=time(Exports),y=Exports,kernel='normal',
                           bandwidth = bandwidth,n.points = length(Exports))

plot(Exports,col='red',main="Original Export Data and Trend")
lines(smoothed_values$x, smoothed_values$y,col='blue')
legend("topright", legend=c("Original", "Trend"), col=c("red", "blue"), pch=c(19, 17))

# Remove the Trend-cycle effect
Exports.noTrend <- Exports - smoothed_values$y
plot(Exports.noTrend,main='Export Data After Removing Trend')
# Plot ACF and PACF
acf2(Exports.noTrend,max.lag= 30,main = "ACF and PACF for data without trend")
#adf and kpss tests
adf.test(Exports.noTrend)
kpss.test(Exports.noTrend)

# Stationary now, indicating a AR(2) MA(1) possibly.
# Already stationary, no need for differential.

# Take the 1st-order difference and test if the data is stationary
Exports.diff<-diff(Exports)
acf2(Exports.diff,max.lag= 30)
adf.test(Exports.diff)
kpss.test(Exports.diff)

#suggesting AR(2), MA(1), but does not pass ADF test.
# Box-Cox transformation
lambda<-BoxCox.lambda(Exports)
Exports.Box<-BoxCox(Exports,lambda)
acf2(Exports.Box,max.lag = 30)
acf2(diff(Exports.Box),max.lag = 30)

# suggesting white noise, so discard this transformation
Exports.log<-log(Exports)
acf2(Exports.log,max.lag= 30)
acf2((diff(Exports.log)),max.lag= 30)

#suggesting white noise, so discard this transformation
# Auto select the best p,d,q combination for ARIMA(p,d,q)
Exports.noTrend.auto <- auto.arima(Exports.noTrend)
summary(Exports.noTrend.auto)
# Use SARIMA to determent the performance, acf of residual all need to be in range,
#p-value needs to be greater than 0.1, QQ lines close to the line.
# ARIMA(2,0,1)
fit1<-sarima(Exports.noTrend, 2,0,1, no.constant=TRUE)
print(paste(c("AIC:",fit1$fit$aic)))

# MSE is a standard to evaluate the model. The smaller is the better.
MSE1<-sum(fit1$fit$residuals)/58
print(paste(c("Training MSE:",MSE1)))
# ARIMA(2,1,1)
fit2<-sarima(Exports.noTrend, 2,1,1, no.constant=TRUE)
print(paste(c("AIC:",fit2$fit$aic)))

# MSE
MSE2<-sum(fit2$fit$residuals)/58
print(paste(c("Training MSE:",MSE2)))
#ARIMA(3,0,1)
fit3<-sarima(Exports.noTrend, 3,0,1, no.constant=TRUE)
print(paste(c("AIC:",fit3$fit$aic)))
# MSE
MSE3<-sum(fit3$fit$residuals)/58
print(paste(c("Training MSE:",MSE3)))

forecast<-forecast(arima(Exports.noTrend,c(2,0,1)))
autoplot(forecast(arima(Exports.noTrend,c(2,0,1))))
```