STAT4601 Time-series analysis

Final Project

Monthly CO_2 concentrations in the globe

Based on Seasonal ARIMA model

1. Introduction

Global warming is the long-term warming of Earth's climate system that has been observed since the pre-industrial period, which is between 1850 and 1900, as a result of human activities and primarily fossil fuel combustion, which increases heat-trapping greenhouse gas (CO_2) levels in the atmosphere. The increase level of CO_2 may cause rise in sea level, rise in sea acidity and malnutrition of plants. With these severe consequences, the increase concentration on CO_2 is considered as one of the major threats in the 21_{st} century, as it contributes to the global warming issue and has had far-reaching effects on many aspects of human health in the last decades.

In this project, we will focus on the monthly CO_2 concentrations since 1958. Time series models will provide us informative mathematical patterns to describe and predict the increasing trend of CO_2 concentrations. As we all know that CO2 concentration should be attained below 460 ppm in order to consider it at a safe level, in the last part of this project, we will try to find out the probability that the CO_2 concentration will reach 460 ppm by 2050 (in the next 30 years).

2. Discussion of Stationarity

The data of monthly mean CO_2 is obtained from the website of Earth System Research Laboratories. We select the monthly mean CO_2 in the globe from 03-1958 to 03-2021, totally 757 records. For better comparison, the recent 5 years (2017 – 2021), total 51 records are used as the test data, which are the data obtained starting from 2017, but not the last 5 data as stated in the project outline.

We need to get an initial understanding about the dataset first, so we draw the time plot.

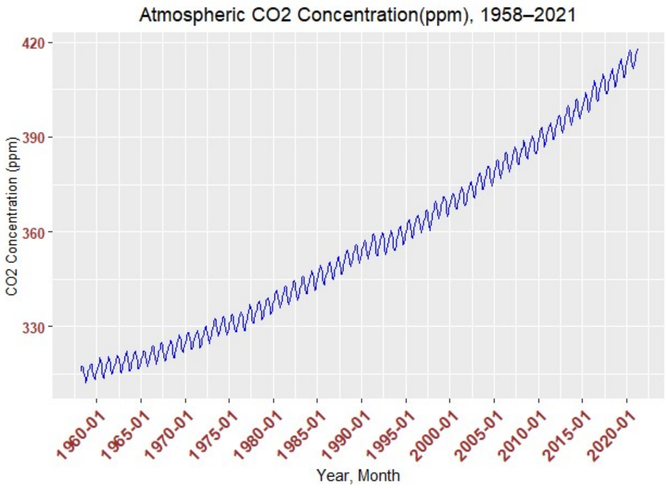
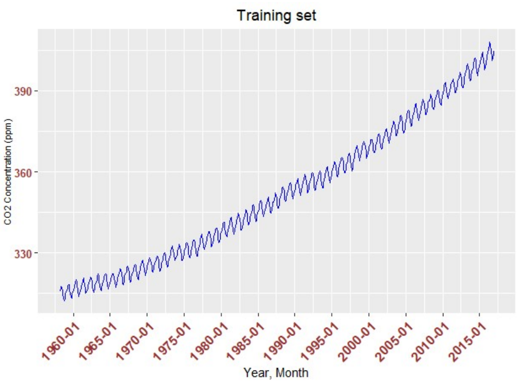


Figure 1a. Time plot of atmospheric CO_2 concentration



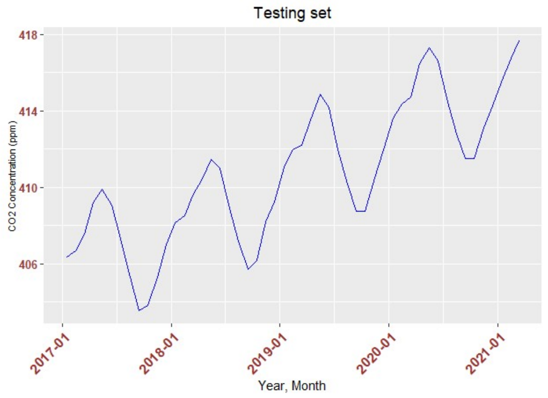


Figure 1b. Training data set

Figure 1c. Testing data set

On the other hand, we see the clear trend in the plot, so the properties of CO2 concentration depend on time, in other words, it is possible that the mean of the above data is time dependent, hence it is obvious that the data is non-stationary. As a result, we may need to take a difference.

Now we draw ACF and PACF of the original data to get further information.

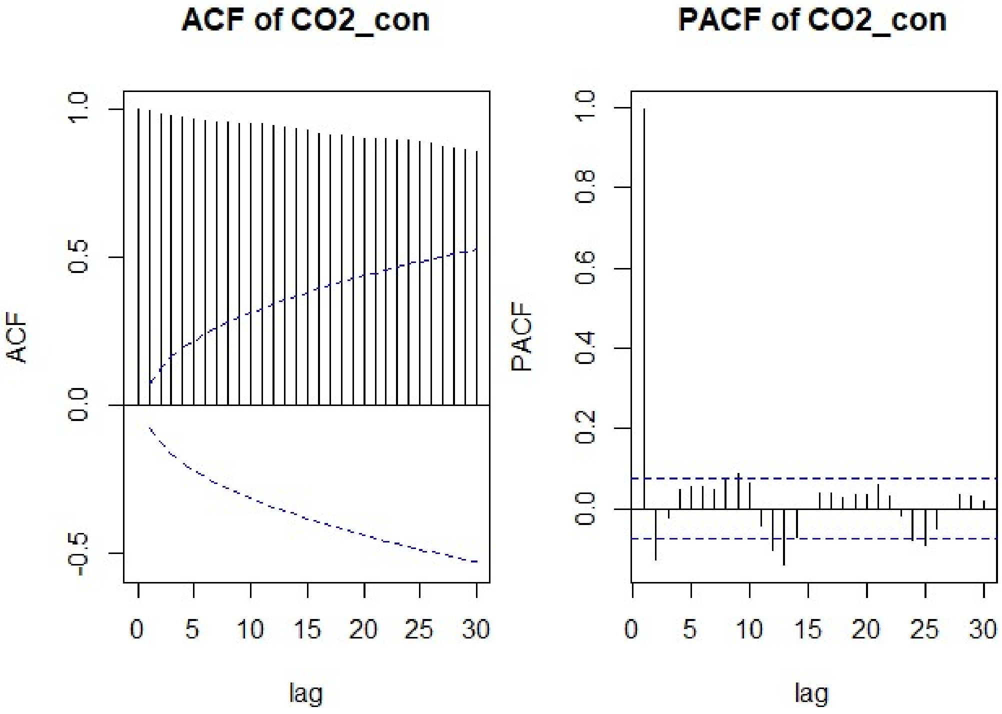


Figure 2. ACF and PACF of the original data

The ACF plot shows the sample ACF does not decay exponentially, so we need to take a difference first to see if it is stationary after transformation.

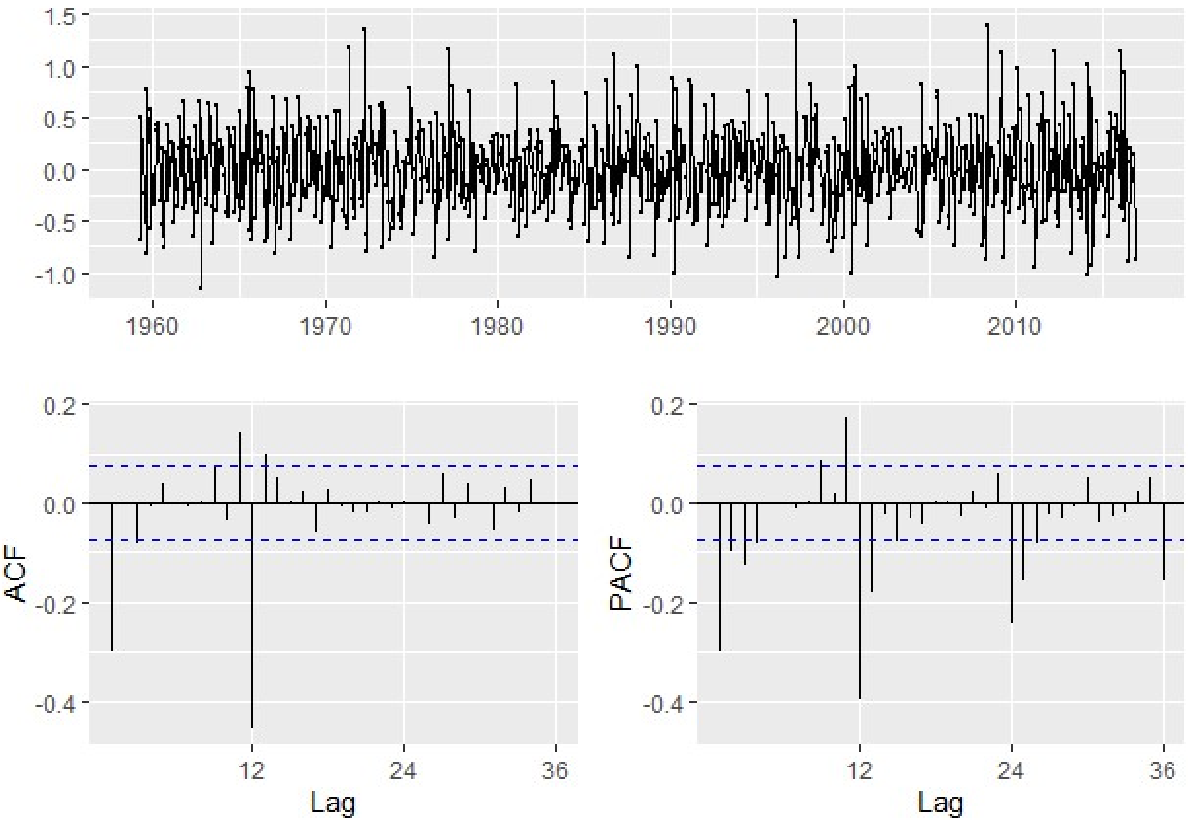


Figure 3. Time plot, ACF and PACF plot after taking difference once

We can see that after taking the difference once, the graph is tilted to the upper part of Figure 3, which means the CO_2 concentration is increasing year by year. Even so, the variation is still not significant. The time plot after difference is consistent with the real situation. Since from 20th century, the situation of global warming has become more serious, hence, the trend of the concentration is making sense to have an upward trend. In other word, the atmospheric CO_2 concentration is fluctuating around +0.5 ppm every month.

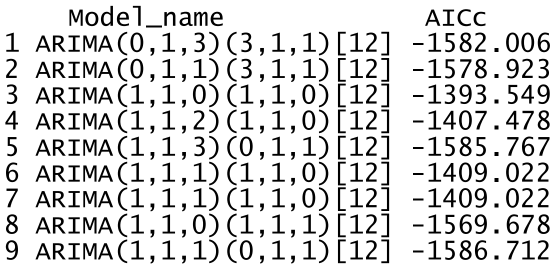
The lower part of Figure 3 is the ACF and PACF of the differenced data. By looking at the data in Figure 1, we can definitely see the seasonality, hence it is suggested to use seasonal differencing to model the data.

After differencing, we are glad to note that ACF is now decaying exponentially. We can justify its stationarity with high confidence by combining the ACF plot and the time plot after taking the difference once.

3. Model Specification and Fitting

Consider seasonal ARIMA(p,d,q)X(P,D,Q)$_S$ model. We start the model with the d=D = 1 with S=12, as we have differenced the model with lag = 12. By looking at the ACF and PACF in Figure 2, we can observe that the ACF shows one significant spike, indicating a possible MA(1) term. As a result, Q = 1 is the starting point. While there are three spikes at the ACF plot in the non-seasonal differenced data plots, this could indicate a seasonal MA(3) term, q=3.

Then, we begin to work with the ARIMA(0,1,3)(3,1,1)[12] and make changes to the AR and MA terms. We use the AICs values to judge the quality of models while keeping the order constant (d,D) and aiming to minimize the AICs.



```
             Model_name      AICc
1 ARIMA(0,1,3)(3,1,1)[12] -1582.006
2 ARIMA(0,1,1)(3,1,1)[12] -1578.923
3 ARIMA(1,1,0)(1,1,0)[12] -1393.549
4 ARIMA(1,1,2)(1,1,0)[12] -1407.478
5 ARIMA(1,1,3)(0,1,1)[12] -1585.767
6 ARIMA(1,1,1)(1,1,0)[12] -1409.022
7 ARIMA(1,1,1)(1,1,0)[12] -1409.022
8 ARIMA(1,1,0)(1,1,1)[12] -1569.678
9 ARIMA(1,1,1)(0,1,1)[12] -1586.712
```

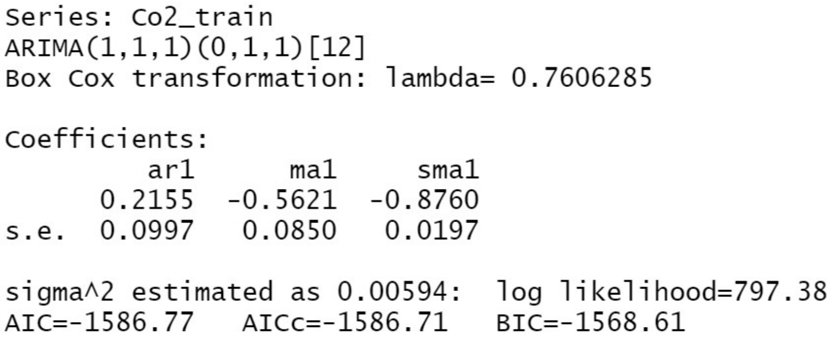
Table 1. Comparing different models with AICs

The ARIMA(1, 1, 1)(0, 1, 1)[12] will be chosen based on the above analysis, as it has the least AICs value with -1586.712.

4. Model Diagnosis

Residual Analysis

We have selected the best model among 9 in the above part, however, we must check the residual to ensure that there is no over- or underfitting, as well as to see if the residuals pass the Ljung-Box test and resemble white noise.

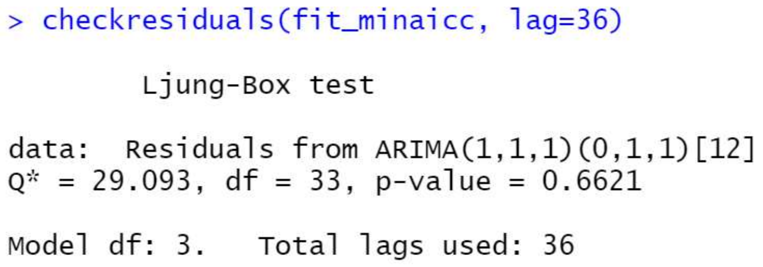


```
Series: Co2_train 
ARIMA(1,1,1)(0,1,1)[12] 
Box Cox transformation: lambda= 0.7606285 

Coefficients:
         ar1      ma1     sma1
      0.2155  -0.5621  -0.8760
s.e.  0.0997   0.0850   0.0197

sigma^2 estimated as 0.00594:  log likelihood=797.38
AIC=-1586.77   AICc=-1586.71   BIC=-1568.61
```

Table 2. Estimated parameters of ARIMA (1,1,1)(0,1,1)$_{12}$



```
> checkresiduals(fit_minaicc, lag=36)

	Ljung-Box test

data:  Residuals from ARIMA(1,1,1)(0,1,1)[12]
Q* = 29.093, df = 33, p-value = 0.6621

Model df: 3.   Total lags used: 36
```

Table 3. Ljung-Box test

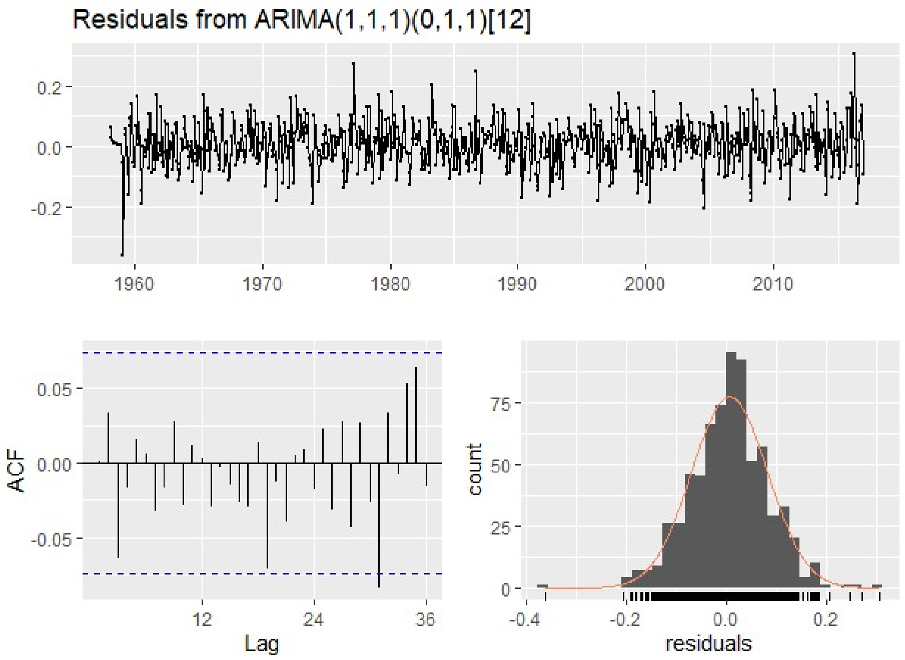
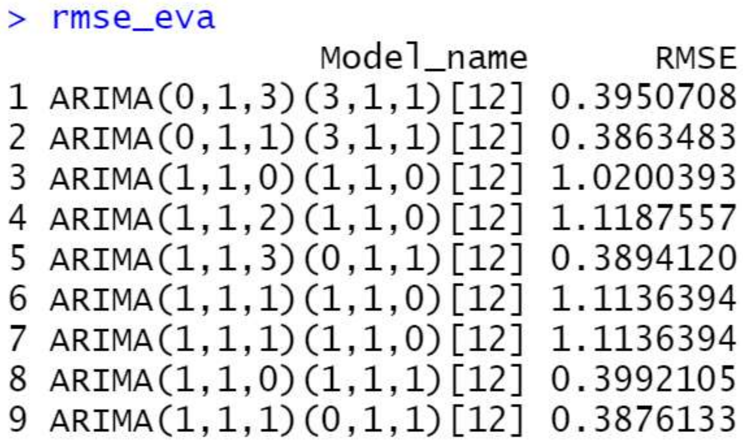


Figure 4. Residuals time plot

To begin, we create a residual time plot and a standardized residual time plot to see if there is any structured pattern that we have yet to find. Now we can see that the residual closely resembles white noise, and the p-value is high, indicating that the model passes the Ljong-Box test. However, with reference to Figure 4, the ACF at lag 31 is just reaching the blue line's boundary; however, I do not believe this will have a significant impact on the prediction — it is sometimes difficult to have a model pass all tests.

RMSE Analysis

After having the Ljong-Box test, we need to compare the performance of the model on the Test data. We seek to find the model which minimizes the root mean square error.



```
> rmse_eva
                Model_name      RMSE
1 ARIMA(0,1,3)(3,1,1)[12] 0.3950708
2 ARIMA(0,1,1)(3,1,1)[12] 0.3863483
3 ARIMA(1,1,0)(1,1,0)[12] 1.0200393
4 ARIMA(1,1,2)(1,1,0)[12] 1.1187557
5 ARIMA(1,1,3)(0,1,1)[12] 0.3894120
6 ARIMA(1,1,1)(1,1,0)[12] 1.1136394
7 ARIMA(1,1,1)(1,1,0)[12] 1.1136394
8 ARIMA(1,1,0)(1,1,1)[12] 0.3992105
9 ARIMA(1,1,1)(0,1,1)[12] 0.3876133
```

Figure 5. Checking the RMSE with the 9 models

The results show that the model ARIMA(1,1,1)(0,1,1)[12] does not have the lowest RMSE value, but it is very similar to the minimum; however, the AICc values show that it is the lowest. Finally, since the model residuals obey the white noise, the model ARIMA(1,1,1)(0,1,1)[12] is chosen to forecast the package because it has less parameters and keeps the AICs down.

5. Forecasting

As stated before, we are going to predict the CO_2 concentration till year 2050 using ARIMA(1,1,1)(0,1,1)[12]. The results are as follows:

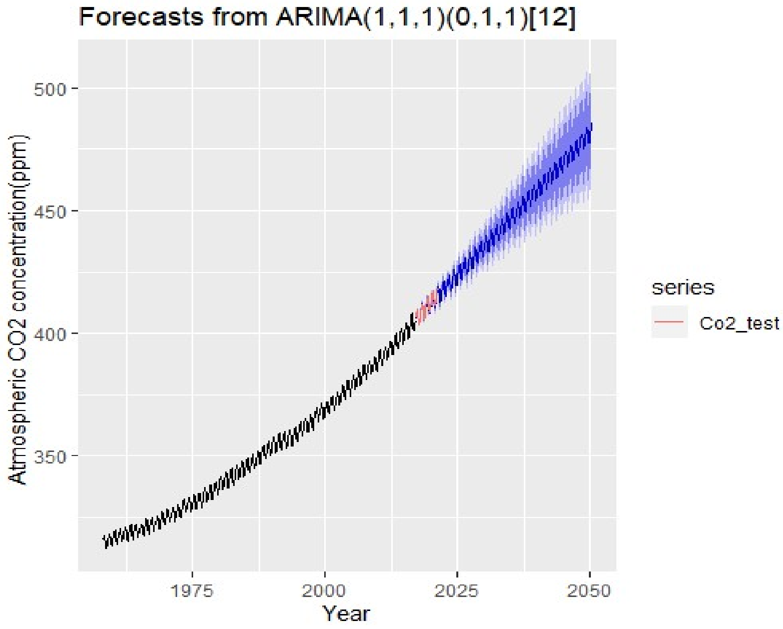


Figure 6. Forecasts from ARIMA (1,1,1)(0,1,1)[12]

The blue line is represented as the predicted value, the black line is the historical data, while the red line is the test data.

For better understanding, we will zoom in the model prediction and the test data (year 2017 – 2021) to see the model performance visually:

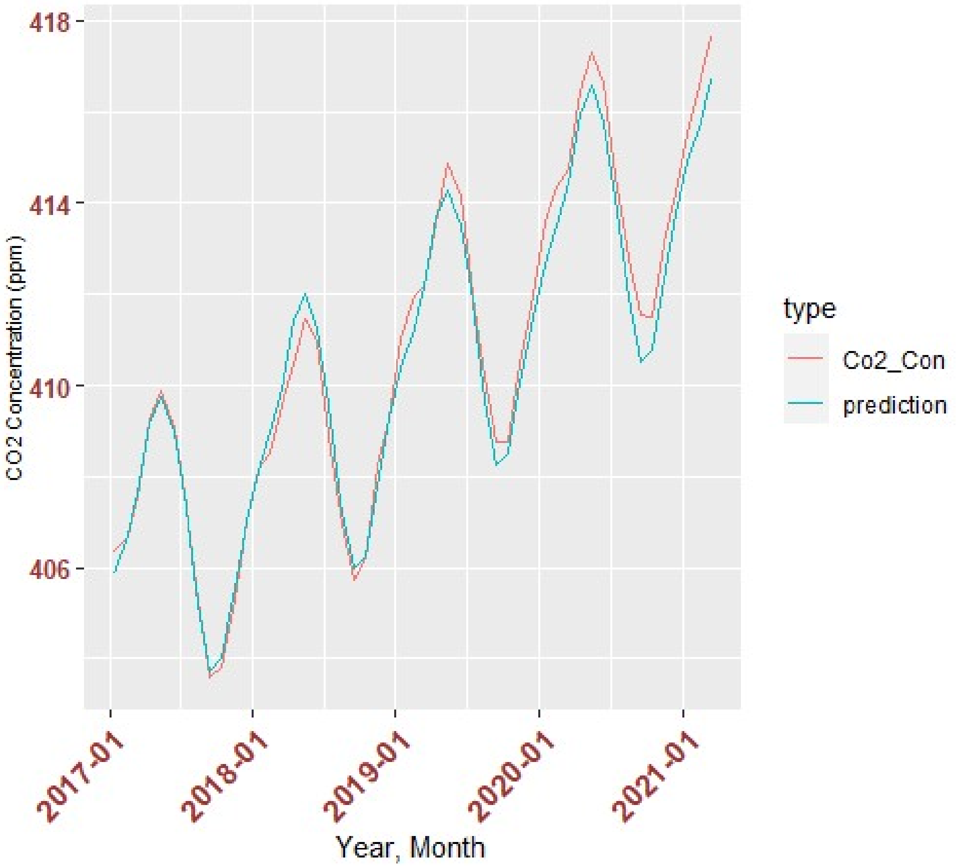


Figure 7. Model Prediction vs Test Data

With reference to Figure 7, we can see the model prediction is actually quite accurate by comparing with the test data, since the gap between the two sets of data is not immensely big.

6. Insight and Conclusion

As mentioned above, one objective of this project is to find out the probability that the atmospheric CO_2 concentration will stay within 460 ppm on 2050. So here is the result:

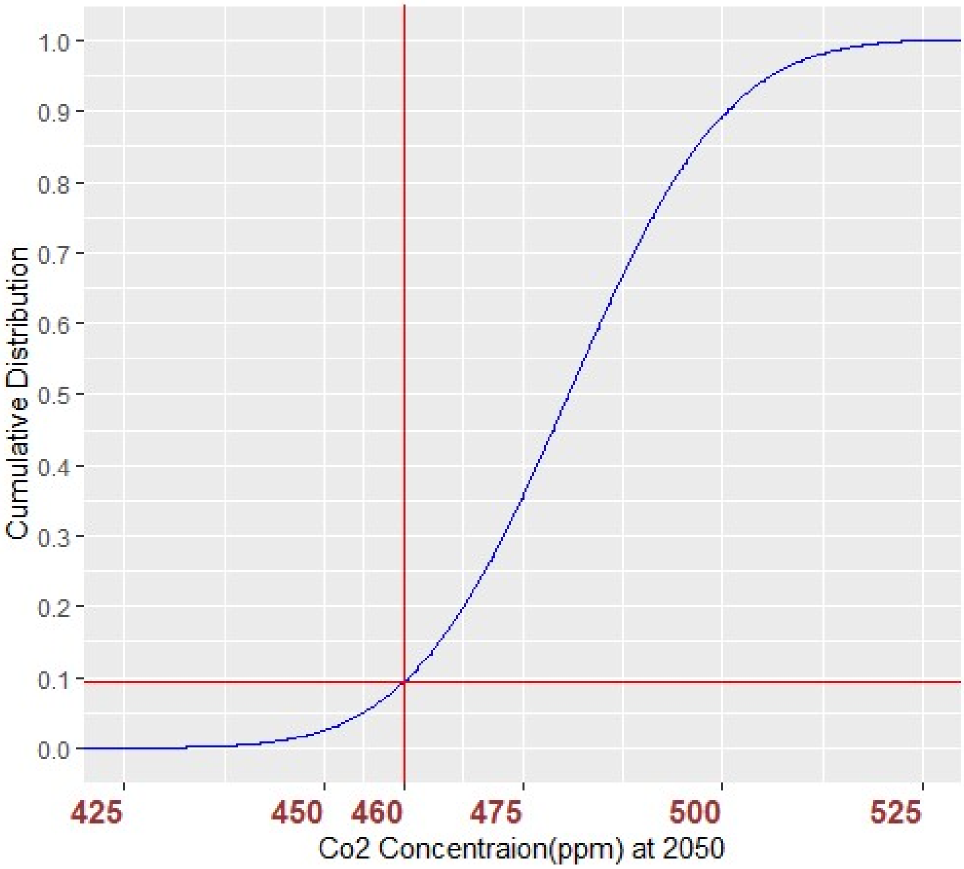


Figure 8. probability of CO_2 concentration at 2050

By calculation, the probability that the CO_2 concentration will stay below 460 by 2050 is 9.2238%, which is an extremely low percentage.

In conclusion, seasonal ARIMA $(1,1,1)(0,1,1)_{12}$ model is probably the best model in predicting the value in this dataset. However, there are many factors that will affect the prediction of the CO_2 concentration, like the occurrence of natural hazards and wars. We must place strict constraints on our model, such as unchanged weather, if we want it to forecast accurately. That means we can only choose a short period of time, such as a year, to develop our model and forecast several months. As each day, there are unforeseen incidents occurring all over the world.

Moreover, with the result in Figure 8, it brings a clear message that it is hard to keep the CO_2 concentration level below 460 ppm before 2050, however, with everyone's efforts in the world, for example, try to control and emit less greenhouse gases, and maintain a healthy, green lifestyle. I am sure we can make the small chance to be possible.

7. Appendices

R codes used in this project:

```
#initialization

library(lubridate)

library(ggplot2)

install.packages("forecast", dependencies = TRUE)

library(forecast)

library(dplyr)

library(tidyr)


#importing and processing the data

data <- read.delim('https://www.esrl.noaa.gov/gmd/webdata/ccgg/trends/co2/co2_mm_mlo.txt', comment.char = '#', header = FALSE, sep = '', col.names = c('Year','Month','Time','Co2_Concentration','Interpolated','#days','Trend','Days'))


which(is.na(data))


data_cc <- data %>%

  mutate(

    Co2_Con = case_when(

      Co2_Concentration == -99.99 ~ Interpolated,

      TRUE ~ Co2_Concentration

    )

  )


sapply(data_cc, class)


data_cc$Date <- ymd(paste0(data$Year, " ", data$Month, " ", "15"))

data_cc_sel <- data_cc %>%

  select(Year, Month, Date, Co2_Con )


data_cc_sel_test <- data_cc_sel %>%
  filter(Year > 2016)
data_cc_sel_train <- data_cc_sel %>%
  filter(Year <= 2016)

ggplot(data_cc_sel,aes(Date, Co2_Con)) +
  geom_line(color='blue') +
  theme(plot.title = element_text(hjust = 0.5))+
  ggtitle("Atmospheric CO2 Concentration(ppm), 1958-2021")+
  xlab("Year, Month") +
  scale_x_date(date_labels = "%Y-%m", date_breaks = "5 year") +
  theme(axis.text.x = element_text(face = "bold", color = "#993333",
                                   size = 12, angle = 45, hjust = 1)) +
  ylab("CO2 Concentration (ppm)") +
  #scale_x_continuous(breaks = trans_breaks(identity, identity, n = 10))
  scale_y_continuous() +
  theme(axis.text.y = element_text(face = "bold", color = "#993333",
                                   size = 10, hjust = 1),axis.title.y = element_text(size = 10))


p2 <- ggplot(data_cc_sel_train,aes(Date, Co2_Con)) +
  geom_line(color='blue') +
  theme(plot.title = element_text(hjust = 0.5))+
  ggtitle("Training set")+
  xlab("Year, Month") +
  scale_x_date(date_labels = "%Y-%m", date_breaks = "5 year") +
  theme(axis.text.x = element_text(face = "bold", color = "#993333",
                                   size = 12, angle = 45, hjust = 1)) +
  ylab("CO2 Concentration (ppm)") +
  #scale_x_continuous(breaks = trans_breaks(identity, identity, n = 10))
  scale_y_continuous() +
  theme(axis.text.y = element_text(face = "bold", color = "#993333",
                                   size = 10, hjust = 1), axis.title.y = element_text(size = 8))
```

```
p3 <- ggplot(data_cc_sel_test,aes(Date, Co2_Con)) +
  geom_line(color='blue') +
  theme(plot.title = element_text(hjust = 0.5))+
  ggtitle("Testing set")+
  xlab("Year, Month") +
  scale_x_date(date_labels = "%Y-%m", date_breaks = "1 year") +
  theme(axis.text.x = element_text(face = "bold", color = "#993333",
                                   size = 12, angle = 45, hjust = 1)) +
  ylab("CO2 Concentration (ppm)") +
  #scale_x_continuous(breaks = trans_breaks(identity, identity, n = 10))
  scale_y_continuous() +
  theme(axis.text.y = element_text(face = "bold", color = "#993333",
                                   size = 10, hjust = 1), axis.title.y = element_text(size = 8))
p2
p3

#draw acf and pacf of the original data
par(mfrow=c(1,2),mai=c(0.8,0.8,0.8,0.1))
acf(data_cc_sel_train$Co2_Con,ci.type='ma',lag=30,
    main = 'ACF of CO2_con', xlab= 'lag',ylab='ACF')
pacf(data_cc_sel_train$Co2_Con,lag=30,
     main = 'PACF of CO2_con',xlab= 'lag',ylab='PACF')
par(mfrow=c(1,1),mai=c(0.8,0.8,0.8,0.1))

# take the difference
Co2_train <- ts(data_cc_sel_train$Co2_Con, start = c(1958,3), frequency = 12)
Co2_train %>% diff(lag=12) %>% diff() %>% ggtsdisplay()

## fit seasonal ARIMA(0,1,1)*(0,1,1)_12 model
aicsvalue <- function(p,q,P,Q) {
  fit <- Arima(Co2_train, order=c(p,1,q),seasonal=list(order=c(P,1,Q),period=12),
               lambda = "auto"
  )
  return(fit$aicc)
}
model_arima <-
data.frame(Model_name=c("ARIMA(0,1,3)(3,1,1)[12]","ARIMA(0,1,1)(3,1,1)[12]","ARIMA(1,1,0)(1,1,0)[12]",
                              "ARIMA(1,1,2)(1,1,0)[12]","ARIMA(1,1,3)(0,1,1)[12]","ARIMA(1,1,1)(1,1,0)[12]",
                              "ARIMA(1,1,1)(1,1,0)[12]","ARIMA(1,1,0)(1,1,1)[12]","ARIMA(1,1,1)(0,1,1)[12]" ),
AICc=c(aicsvalue(0,3,3,1),aicsvalue(0,1,3,1),aicsvalue(1,0,1,0),aicsvalue(1,2,1,0),aicsvalue(1,3,0,1),aicsvalue(1,1,1,0)
,aicsvalue(1,1,1,0),aicsvalue(1,0,1,1), aicsvalue(1,1,0,1)))
model_arima


#residuals
(fit_minaicc <- Arima(Co2_train, order=c(1,1,1),seasonal=list(order=c(0,1,1),period=12),
                  lambda = "auto"
))
checkresiduals(fit_minaicc, lag=36)
fit_minaicc$aicc


#RMSE check
Co2_test <- ts(data_cc_sel_test$Co2_Con, start = c(2017,1), frequency = 12)
mm <- accuracy(forecast(fit_minaicc,h=35)$mean, Co2_test )


rmse_eva <- function(p,d,q,P,D,Q) {
  fit <- Arima(Co2_train, order=c(p,d,q),seasonal=list(order=c(P,D,Q),period=12),
            lambda = "auto"
  )
  mm <- accuracy(forecast(fit,h=35)$mean, Co2_test)
  return(mm[2])


}


rmse_eva <- data.frame(Model_name=c(
  "ARIMA(0,1,3)(3,1,1)[12]","ARIMA(0,1,1)(3,1,1)[12]","ARIMA(1,1,0)(1,1,0)[12]",
  "ARIMA(1,1,2)(1,1,0)[12]","ARIMA(1,1,3)(0,1,1)[12]","ARIMA(1,1,1)(1,1,0)[12]",

  "ARIMA(1,1,1)(1,1,0)[12]","ARIMA(1,1,0)(1,1,1)[12]","ARIMA(1,1,1)(0,1,1)[12]"

), RMSE=c(

  rmse_eva(0,1,3,3,1,1),rmse_eva(0,1,1,3,1,1),rmse_eva(1,1,0,1,1,0),
rmse_eva(1,1,2,1,1,0),rmse_eva(1,1,3,0,1,1),rmse_eva(1,1,1,1,1,0),
rmse_eva(1,1,1,1,1,0),rmse_eva(1,1,0,1,1,1),rmse_eva(1,1,1,0,1,1)))

print(rmse_eva)


#Forecast

Co2_train %>%

  Arima(order=c(1,1,1),seasonal=list(order=c(0,1,1),period=12),

      lambda = "auto"

  ) %>%

  forecast(h=400) %>%

  autoplot() +

  ylab("Atmospheric CO2 concentration(ppm) ") + xlab("Year") +

  autolayer(Co2_test)


#Forecast vs test data

prediction <- forecast(fit_minaicc,h=51)

data_cc_sel_test$prediction <- prediction$mean

data_test_pre_tidy <- gather(data_cc_sel_test, "type", "Co2", -Year,-Month,-Date)


ggplot(data_test_pre_tidy,aes(Date, Co2,color=type)) +

  geom_line() +

  xlab("Year, Month") +

  scale_x_date(date_labels = "%Y-%m", date_breaks = "1 year") +

  theme(axis.text.x = element_text(face = "bold", color = "#993333",

                           size = 12, angle = 45, hjust = 1)) +

  ylab("CO2 Concentration (ppm)") +

  #scale_x_continuous(breaks = trans_breaks(identity, identity, n = 10))

  scale_y_continuous() +

  theme(axis.text.y = element_text(face = "bold", color = "#993333",
                                  size = 10, hjust = 1), axis.title.y = element_text(size = 8))

#2050
prediction1 <- forecast(fit_minaicc,h=396, level = c(80,90))
p10 <- prediction1$upper[396,2]
p50 <- prediction1$mean[396]
sd_calc <- (p10-p50)/1.28
Co2_con_2050 <- rnorm(10^6,p50,sd_calc)
cdf_co2_con_2050 <- ecdf(Co2_con_2050)
cdf_co2_con_2050_data <- data.frame(Co2_con_2050)
ggplot(cdf_co2_con_2050_data, aes(Co2_con_2050)) + stat_ecdf(geom = "step", color='blue') +
  geom_vline(xintercept = 460, color='red') +
  geom_hline(yintercept = cdf_co2_con_2050(460), color='red') +
  theme(axis.text.x = element_text(face = "bold", color = "#993333",
                                  size = 12, angle = 0, hjust = 1)) +
  scale_x_continuous(breaks=c(400,425,450, 460,475,500,525, 550), limits = c(425,525)) +
  scale_y_continuous(breaks=c(seq(0,1,0.1)), limits = c(0,1)) +
  ylab('Cumulative Distribution') +
  xlab("Co2 Concentraion(ppm) at 2050")

cdf_co2_con_2050(460)
```