**Analysis**

**Data Wrangling**

A computer screen with text on it

Description automatically generated

A blue screen with white text

Description automatically generated

First, I imported the data and skipped the first row of the csv.

Then , I dropped the non-required columns.

I used tidyr library to covert the df into long format with temperature as the value and month as the key.

I then check the summary of the dataset and found that the temperature is stored as a character datatype.

I converted this to numeric. Further, I checked for missing values and found 5 Nas which I removed.

A computer screen shot of a program code

Description automatically generated

Encoding months, creating date and lags for the time series.

A blue screen with white text

Description automatically generated

Creating yearly dummies for every 12th month with the logic that if month number is divided by 12 , remainder is 0 , then assign 1.

**Plot the Resulting Time Series:**

Plotting the time series using ggplot.

A screen shot of a computer program

Description automatically generated

A graph of red lines

Description automatically generated

**Insights:**

**Trend:** there is an upward trend in the temperatures especially form the 20th century onwards . The average temperature seem to be increases globally

**Seasonality:** there is definite seasonality as within each year. There seems to be regular fluctuations that suggest seasonality. These regular ups and downs can be attributed to changing seasons with temperature variations.

**Cyclical components** : From the plot, it’s a bit challenging to confidently identify any Cyclical components because the visible trend could overshadow them. A detailed decomposition of the plot would perhaps provide more clarity on that. However, there is potential for such components to be present given the temperature data over time.

**Time Series Model:**

**A computer screen shot of text

Description automatically generated**

A screenshot of a computer

Description automatically generated

Since, I had added 2 order lags I am fitting 2 models to compare if the inclusion of lag2 improves our model.

From the above comparison, the modelling on the training data with lag 2 seems to increase the R2 and further the RMSE for model 2 is lower than model 1 when compared to the test set.

A screen shot of a computer

Description automatically generated

A lower **Rmse** means that the model2 is better at predicting the temperature and the inclusion of lag 2 has made the model more accurate. Further, except the month, all other inputs are statistically significant.

***We can also say that for a yearly increase the temperature rises by 0.001 degrees Celsius.***

**Durbin Watson Test:**

**A screenshot of a computer program

Description automatically generated**

From the durbin Watson test ,

H0: No auto correlation

Model 1: has a negative auto correlation with lag 1 and a t statistic greater than 2. It is also significant given the p-value. Thus the null is rejected.

Model 2: has a lesser intensity of negative autocorrelation and a t statistic closer to 2 . It also has a p-value > 0.05 , thus we fail to reject the null.

Thus , Model 2 definitely is a better model than model1 which has first order lag.

We could further add third order lag to make the model better.

**Summary of Findings:**

From the analyses, we can make pretty accurate predictions. Given we have an R2 of 86% and an RMSE of 12%. As pointed before , we can also say that for a yearly increase the temperature rises by 0.001 degrees Celsius.

Including a second order lag produced a better model and lowered the auto correlation. We could further optimize the model by adding some more factors if needed or adding a third order lag.

A screen shot of a computer screen

Description automatically generated

There clearly is an upward trend in predicted temperatures and thus we can predict global warming with good confidence based on our model.

**APPENDIX:**

library(tidyr)

library(dplyr)

library(ggplot2)

library(car)

library(readr)

df <- read\_csv("C:/Users/mudas/OneDrive/Desktop/QMB/GLB.Ts+dSST.csv", skip = 1)

df = df[,-c(14:19)]

df\_clean = gather(df,key='Month',value='Temperature', 2:13)

summary(df\_clean)

str(df\_clean)

df\_clean$Temperature = as.numeric(df\_clean$Temperature)

#check for missing values

sum(is.na(df\_clean))

#remove missing values

df\_clean = na.omit(df\_clean)

#Encoding months as numericals

df\_clean$month\_num = match(df\_clean$Month, month.abb)

#Create a date column as required in the task

df\_clean$date = as.Date(paste(df\_clean$Year,df\_clean$month\_num, 01, sep = "-"),"%Y-%m-%d")

#Arranging the data as per the date for better visibility

df\_clean = arrange(df\_clean,date)

#Creating lags

df\_clean$Temperature\_lag1 = c(NA,head(df\_clean$Temperature,-1))

df\_clean$Temperature\_lag2 = c(NA,NA,head(df\_clean$Temperature,-2))

#create yearly dummies as temperatures tend to have yearly seasonallity

df\_clean$yearly\_dummies = ifelse(df\_clean$month\_num %% 12 == 0,1,0)

#Plotting the time series

ggplot(df\_clean, aes(date,Temperature)) +

geom\_line(color = 'red') +

labs(title = "Global Surface Temperatures ",

x="Years",

y="Temperatures")

#Time series modelling

#Train and test split

df\_train = subset(df\_clean, Year>=1880 & Year<=2009)

df\_test = subset (df\_clean, Year>= 2010 & Year <=2019)

#fit a Linear model

lm1.out = lm(Temperature ~ Temperature\_lag1 + Year + month\_num +yearly\_dummies, data = df\_train )

lm2.out = lm(Temperature ~ Temperature\_lag1 + Temperature\_lag2 + Year + month\_num +yearly\_dummies, data = df\_train )

library(stargazer)

stargazer(lm1.out,lm2.out, type="text")

#predicitng on the test set

pred = predict(lm1.out, df\_test)

pred2 = predict(lm2.out, df\_test)

#calculating rmse using Metrics library

library(Metrics)

rmse(df\_test$Temperature, pred)

rmse(df\_test$Temperature, pred2)

#Durbin Watson Test for Auto Correlation.

durbinWatsonTest(lm1.out)

durbinWatsonTest(lm2.out)