NAME: Penugonda Abhignanendra **REGNO**:19BCE1677
FACULTY: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

Essential of Data Analytics
Tasks for Week-3: Multi linear regress and Time-series Forecasting

Understand time-series operations/functions and forecast the annual gdp growth rate of India based on given instructions.

Aim: To develop a multi linear regression and time series forecasting model for the given data using R programming and to predict the future data

Algorithm:

➔ Multi linear regression:

- Import the dplyr library.
- Store the weatherHistory2016.csv data into variables data2.
- Take a sample of 100 data2 using sample_n() function and store it in train.
- Using cor.test we get the correlation between temperature and apparent temperature, humidity, windspeed, wind bearning degree, visibility, pressure millibars.
- Create a linear regression model using lm() function for temperature and apparent temperature, humidity.
- Find the summary of the lm model created using summary () function.

➔ Time series function:

- Set the working directory and the read the respective csv file using read.csv() function.
- Import the forecast and tseries libraries.
- Using ts() function we convert normal numerical data int R time series object. In ts the previous data which use for predicting is written first later the start will the start data of year, month or day and end will be the end data of year, month or day.
- By using class we can check the class of the object.
- By plotting we can the variation in data.

NAME: Penugonda Abhignanendra **REGNO**:19BCE1677
FACULTY: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

Essential of Data Analytics
Tasks for Week-3: Multi linear regress and Time-series Forecasting

- By using the aut.arima() we can make the non-stationary data into stationary by using some models. the pdq values in arima means p value is the auto regression d value is integrated and q is the moving average.
- By using the best model, we use forecast to future data.
- By plotting the graph, we can see the range of change in data.
- By using accuracy data, we can we can find how best our model is.

Statistic:

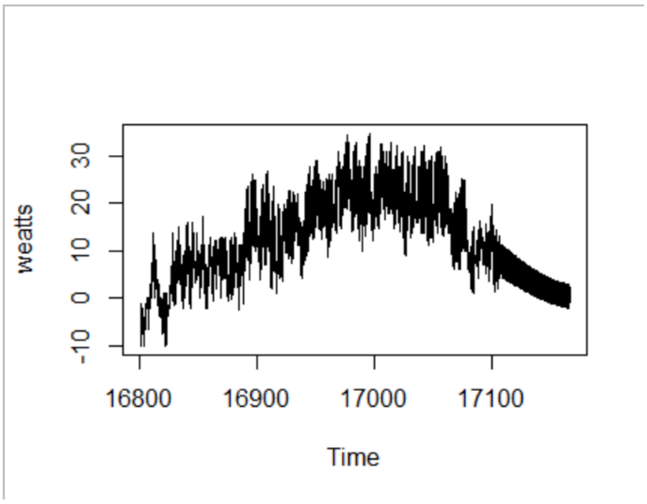
	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	5.84697	0.71670	8.158	1.31e-12 ***
Apparent.Temperature..C.	0.85342	0.01308	65.261	< 2e-16 ***
Humidity	-3.70090	0.70158	-5.275	8.20e-07 ***

Residual standard error: 0.8627
R2: 0.9915 **Adj-R2: 0.9912**
F-statistic: 3726 **p-value: <2.2e-16**

Inference:

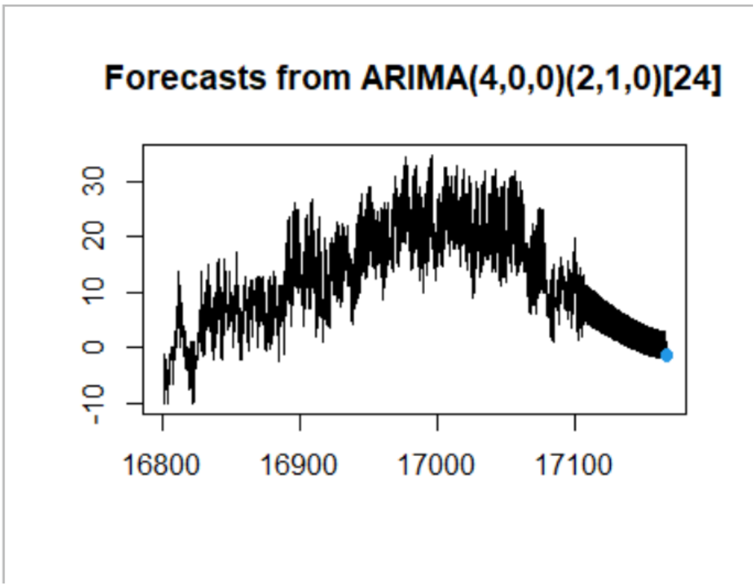
The p-value is more than the critical p value that means the data is non stationary data. We convert it into stationary and the arima values are p=4, d=0, q=0.

Result:



NAME: Penugonda Abhignanendra **REGNO**:19BCE1677
FACULTY: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

Essential of Data Analytics
Tasks for Week-3: Multi linear regress and Time-series Forecasting



	Point Forecast	Lo 95	Hi 95
17166.04	-1.152319	-2.838037	0.5333991
17166.08	-1.478033	-3.955125	0.9990587

Program:

```
#multi linear regression
rm(list=ls())
setwd("C:/Abhi notes/class3-2/eda/lab/Lab 3")
library(dplyr)
data2<-read.csv("weatherHistory2016.csv")
data2
train=sample_n(data2,100)
cor.test(train$Temperature..C.,train$Apparent.Temperature..C.)
cor.test(train$Temperature..C.,train$Humidity)
cor.test(train$Temperature..C.,train$Wind.Speed..km.h.)
cor.test(train$Temperature..C.,train$Wind.Bearing..degrees.)
cor.test(train$Temperature..C.,train$Visibility..km.)
```

Essential of Data Analytics
Tasks for Week-3: Multi linear regress and Time-series Forecasting

```
cor.test(train$Temperature..C.,train$Pressure..millibars.)

lmodel<-
lm(train$Temperature..C.~(train$Apparent.Temperature..C.+train$Humidity))

summary(lmodel)

print("by this inference the p value of b2,b1 and b0 are of less than 0.05 that means
the model is0. significant")


#time series function

library(forecast)

library(tseries)

weat<-read.csv("weatherHistory2016.csv")

weatts <- ts(weat$Temperature..C., start=as.Date("2016-01-01"),
end=as.Date("2016-12-31"), frequency=24)

class(weatts)

plot(weatts)

gdpmodel=auto.arima(weatts,ic="aic",trace=TRUE)

gdpf=forecast(gdpmodel,level=c(95),h=2)

gdpf

plot(gdpf)

accuracy(gdpf)
```