# Method of Least Squares

# OBJECTIVE

1. Selecting a suitable time series data with trend component and without seasonality 2.Discussing all the necessary steps to extract the stationary version of the series using method of least squares # 1. Importing the data set

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.1.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.1.3
```

```
POP <- read_csv("D:/MSTAT/SEM 3/Time Series Analysis/TS Lab/POP.csv")
```

```
## Rows: 816 Columns: 2
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): date
## dbl (1): value
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
View(POP)
```

# INTERPRETATION

This is a dataset from the U.S. Census Bureau hosted by the Federal Reserve Economic Database (FRED)

# 2. Converting the data into time series object

```
class(POP) #data is not in time series format
```

```
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

```
data = ts(POP$value, start = 1952, frequency = 12) #converting the data into time series object
data
```

```
##           Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
## 1952 156309.0 156527.0 156731.0 156943.0 157140.0 157343.0 157553.0 157798.0
## 1953 158973.0 159170.0 159349.0 159556.0 159745.0 159956.0 160184.0 160449.0
## 1954 161690.0 161912.0 162124.0 162350.0 162564.0 162790.0 163026.0 163290.0
## 1955 164588.0 164809.0 165018.0 165251.0 165463.0 165695.0 165931.0 166192.0
## 1956 167513.0 167746.0 167977.0 168221.0 168436.0 168659.0 168903.0 169191.0
## 1957 170571.0 170806.0 171029.0 171271.0 171501.0 171741.0 171984.0 172257.0
## 1958 173533.0 173746.0 173945.0 174176.0 174397.0 174639.0 174882.0 175143.0
## 1959 176447.0 176685.0 176905.0 177146.0 177365.0 177591.0 177830.0 178101.0
## 1960 179386.0 179597.0 179788.0 180007.0 180222.0 180444.0 180671.0 180945.0
## 1961 182287.0 182520.0 182742.0 182992.0 183217.0 183452.0 183691.0 183958.0
## 1962 185242.0 185452.0 185650.0 185874.0 186087.0 186314.0 186538.0 186790.0
## 1963 188013.0 188213.0 188387.0 188580.0 188790.0 189018.0 189242.0 189496.0
## 1964 190668.0 190858.0 191047.0 191245.0 191447.0 191666.0 191889.0 192131.0
## 1965 193223.0 193393.0 193540.0 193709.0 193888.0 194087.0 194303.0 194528.0
## 1966 195539.0 195688.0 195831.0 195999.0 196178.0 196372.0 196560.0 196762.0
## 1967 197736.0 197892.0 198037.0 198206.0 198363.0 198537.0 198712.0 198911.0
## 1968 199808.0 199920.0 200056.0 200208.0 200361.0 200536.0 200706.0 200898.0
## 1969 201760.0 201881.0 202023.0 202161.0 202331.0 202507.0 202677.0 202877.0
## 1970 203849.0 204008.0 204156.0 204401.0 204607.0 204830.0 205052.0 205295.0
## 1971 206466.0 206668.0 206855.0 207065.0 207260.0 207462.0 207661.0 207881.0
## 1972 208917.0 209061.0 209212.0 209386.0 209545.0 209725.0 209896.0 210075.0
## 1973 210985.0 211120.0 211254.0 211420.0 211577.0 211746.0 211909.0 212092.0
## 1974 212932.0 213074.0 213211.0 213361.0 213513.0 213686.0 213854.0 214042.0
## 1975 214931.0 215065.0 215198.0 215353.0 215523.0 215768.0 215973.0 216195.0
## 1976 217095.0 217249.0 217381.0 217528.0 217685.0 217861.0 218035.0 218233.0
## 1977 219179.0 219344.0 219504.0 219684.0 219859.0 220046.0 220239.0 220458.0
## 1978 221477.0 221629.0 221792.0 221991.0 222176.0 222379.0 222585.0 222805.0
## 1979 223865.0 224053.0 224235.0 224438.0 224632.0 224843.0 225055.0 225295.0
## 1980 226451.0 226656.0 226849.0 227061.0 227251.0 227522.0 227726.0 227953.0
## 1981 228937.0 229071.0 229224.0 229403.0 229575.0 229761.0 229966.0 230187.0
## 1982 231157.0 231313.0 231470.0 231645.0 231809.0 231992.0 232188.0 232392.0
## 1983 233322.0 233473.0 233613.0 233781.0 233922.0 234118.0 234307.0 234501.0
## 1984 235385.0 235527.0 235675.0 235839.0 235993.0 236160.0 236348.0 236549.0
## 1985 237468.0 237602.0 237732.0 237900.0 238074.0 238270.0 238466.0 238679.0
## 1986 239638.0 239788.0 239928.0 240094.0 240271.0 240459.0 240651.0 240854.0
## 1987 241784.0 241930.0 242079.0 242252.0 242423.0 242608.0 242804.0 243012.0
## 1988 243981.0 244131.0 244279.0 244445.0 244610.0 244806.0 245021.0 245240.0
## 1989 246224.0 246378.0 246530.0 246721.0 246906.0 247114.0 247342.0 247573.0
## 1990 248659.0 248827.0 249012.0 249306.0 249565.0 249849.0 250132.0 250439.0
## 1991 251889.0 252135.0 252372.0 252643.0 252913.0 253207.0 253493.0 253807.0
## 1992 255214.0 255448.0 255703.0 255992.0 256285.0 256589.0 256894.0 257232.0
## 1993 258679.0 258919.0 259152.0 259414.0 259680.0 259963.0 260255.0 260566.0
## 1994 261919.0 262123.0 262352.0 262631.0 262877.0 263152.0 263436.0 263724.0
## 1995 265044.0 265270.0 265495.0 265755.0 265998.0 266270.0 266557.0 266843.0
## 1996 268151.0 268364.0 268595.0 268853.0 269108.0 269386.0 269667.0 269976.0
## 1997 271360.0 271585.0 271821.0 272083.0 272342.0 272622.0 272912.0 273237.0
## 1998 274626.0 274838.0 275047.0 275304.0 275564.0 275836.0 276115.0 276418.0
## 1999 277790.0 277992.0 278198.0 278451.0 278717.0 279001.0 279295.0 279602.0
## 2000 280976.0 281190.0 281409.0 281653.0 281877.0 282126.0 282385.0 282653.0
## 2001 283920.0 284137.0 284350.0 284581.0 284810.0 285062.0 285309.0 285570.0
## 2002 286788.0 286994.0 287190.0 287397.0 287623.0 287864.0 288105.0 288360.0
```

```
## 2003 289518.0 289714.0 289911.0 290125.0 290346.0 290584.0 290820.0 291072.0
## 2004 292192.0 292368.0 292561.0 292779.0 292997.0 293223.0 293463.0 293719.0
## 2005 294914.0 295105.0 295287.0 295490.0 295704.0 295936.0 296186.0 296440.0
## 2006 297647.0 297854.0 298060.0 298281.0 298496.0 298739.0 298996.0 299263.0
## 2007 300574.0 300802.0 301021.0 301254.0 301483.0 301739.0 302004.0 302267.0
## 2008 303506.0 303711.0 303907.0 304117.0 304323.0 304556.0 304798.0 305045.0
## 2009 306208.0 306402.0 306588.0 306787.0 306984.0 307206.0 307439.0 307685.0
## 2010 308833.0 309027.0 309212.0 309191.2 309369.1 309548.5 309745.7 309957.8
## 2011 310960.7 311113.4 311265.4 311436.2 311607.1 311791.2 311997.0 312205.4
## 2012 313183.2 313339.0 313499.4 313667.1 313830.5 314017.6 314210.8 314422.3
## 2013 315389.6 315520.1 315662.2 315817.9 315983.7 316171.0 316358.8 316580.3
## 2014 317593.9 317753.9 317917.2 318089.2 318269.5 318464.2 318662.4 318893.8
## 2015 319928.6 320074.5 320230.8 320402.3 320584.0 320773.6 320978.2 321202.5
## 2016 322232.9 322398.1 322551.5 322721.2 322900.0 323088.5 323291.0 323501.4
## 2017 324438.2 324581.5 324714.0 324861.8 325019.2 325186.2 325367.6 325567.7
## 2018 326454.1 326600.8 326736.7 326887.9 327048.7 327219.1 327403.9 327600.2
## 2019 328467.8 328610.7 328742.8 328890.2 329047.3 329214.0 329395.0 329591.3
##           Sep      Oct      Nov      Dec
## 1952 158053.0 158306.0 158451.0 158757.0
## 1953 160718.0 160978.0 161223.0 161453.0
## 1954 163570.0 163847.0 164107.0 164349.0
## 1955 166473.0 166755.0 167023.0 167270.0
## 1956 169488.0 169780.0 170063.0 170315.0
## 1957 172538.0 172816.0 173070.0 173298.0
## 1958 175413.0 175697.0 175966.0 176207.0
## 1959 178376.0 178657.0 178921.0 179153.0
## 1960 181238.0 181528.0 181796.0 182042.0
## 1961 184243.0 184524.0 184783.0 185016.0
## 1962 187058.0 187323.0 187574.0 187796.0
## 1963 189761.0 190028.0 190265.0 190472.0
## 1964 192376.0 192631.0 192847.0 193039.0
## 1965 194761.0 194997.0 195195.0 195372.0
## 1966 196984.0 197207.0 197398.0 197572.0
## 1967 199113.0 199311.0 199498.0 199657.0
## 1968 201095.0 201290.0 201466.0 201621.0
## 1969 203090.0 203302.0 203500.0 203675.0
## 1970 205540.0 205788.0 206024.0 206238.0
## 1971 208114.0 208345.0 208555.0 208740.0
## 1972 210278.0 210479.0 210656.0 210821.0
## 1973 212289.0 212475.0 212634.0 212785.0
## 1974 214246.0 214451.0 214625.0 214782.0
## 1975 216393.0 216587.0 216771.0 216931.0
## 1976 218440.0 218644.0 218834.0 219006.0
## 1977 220688.0 220904.0 221109.0 221303.0
## 1978 223053.0 223271.0 223477.0 223670.0
## 1979 225547.0 225801.0 226027.0 226243.0
## 1980 228186.0 228417.0 228612.0 228779.0
## 1981 230412.0 230641.0 230822.0 230989.0
## 1982 232599.0 232816.0 232993.0 233160.0
## 1983 234701.0 234907.0 235078.0 235235.0
## 1984 236760.0 236976.0 237159.0 237316.0
## 1985 238898.0 239113.0 239307.0 239477.0
```

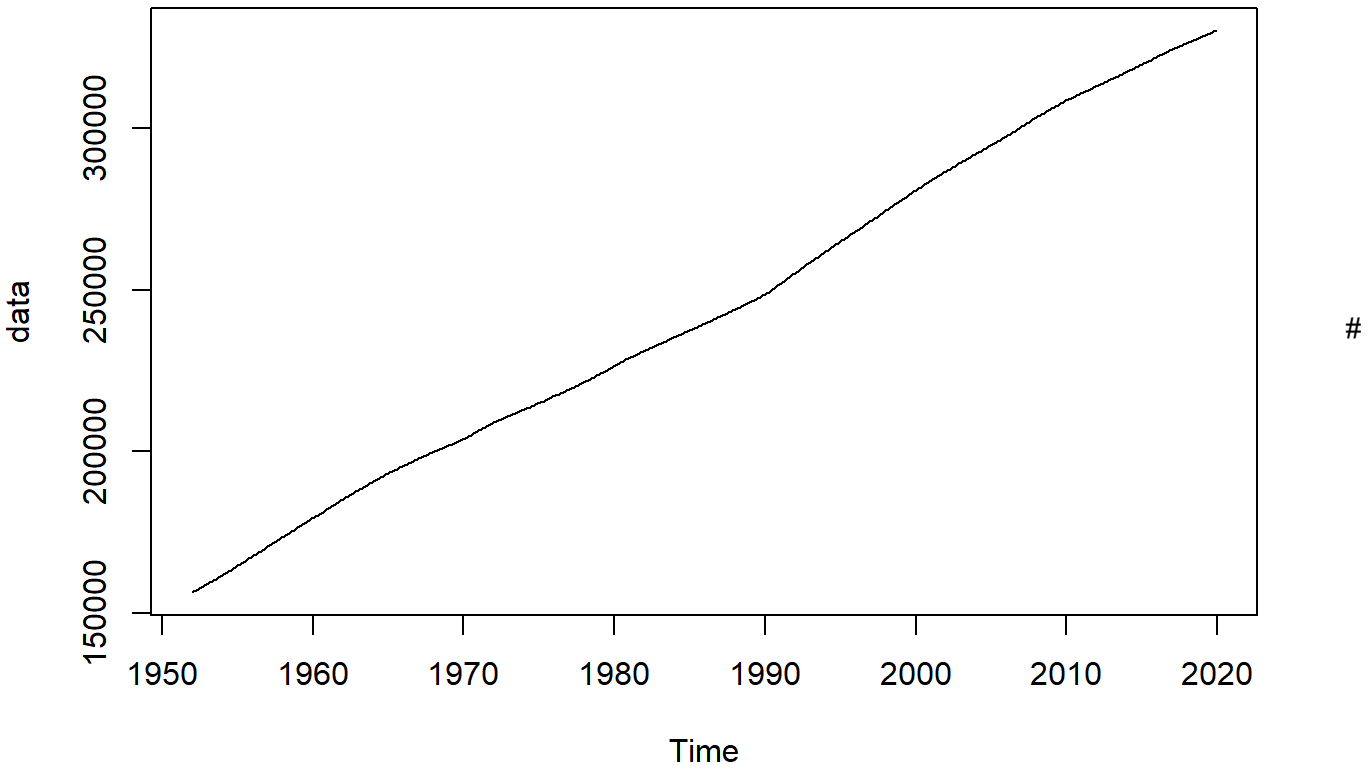
```
## 1986 241068.0 241274.0 241467.0 241620.0
## 1987 243223.0 243446.0 243639.0 243809.0
## 1988 245464.0 245693.0 245884.0 246056.0
## 1989 247816.0 248067.0 248281.0 248479.0
## 1990 250751.0 251057.0 251346.0 251626.0
## 1991 254126.0 254435.0 254718.0 254964.0
## 1992 257548.0 257861.0 258147.0 258413.0
## 1993 260867.0 261163.0 261425.0 261674.0
## 1994 264017.0 264301.0 264559.0 264804.0
## 1995 267152.0 267456.0 267715.0 267943.0
## 1996 270284.0 270581.0 270878.0 271125.0
## 1997 273553.0 273852.0 274126.0 274372.0
## 1998 276714.0 277003.0 277277.0 277526.0
## 1999 279903.0 280203.0 280471.0 280716.0
## 2000 282932.0 283201.0 283453.0 283696.0
## 2001 285843.0 286098.0 286341.0 286570.0
## 2002 288618.0 288870.0 289106.0 289313.0
## 2003 291321.0 291574.0 291807.0 292008.0
## 2004 293971.0 294230.0 294466.0 294694.0
## 2005 296707.0 296972.0 297207.0 297431.0
## 2006 299554.0 299835.0 300094.0 300340.0
## 2007 302546.0 302807.0 303054.0 303287.0
## 2008 305309.0 305554.0 305786.0 306004.0
## 2009 307946.0 308189.0 308418.0 308633.0
## 2010 310176.5 310400.0 310595.8 310781.7
## 2011 312429.1 312644.2 312829.5 313009.7
## 2012 314646.7 314854.0 315053.9 315232.8
## 2013 316806.1 317022.3 317228.0 317411.6
## 2014 319125.3 319353.7 319564.2 319746.2
## 2015 321427.6 321653.0 321856.3 322043.1
## 2016 323709.9 323919.7 324106.5 324274.9
## 2017 325766.0 325966.0 326142.6 326301.4
## 2018 327794.8 327991.0 328163.9 328318.9
## 2019 329785.9 329982.0 330154.9 330309.9
```

```
class(data)
```

```
## [1] "ts"
```

# 3. Time series plot

```
ts.plot(data)
```




#

INTERPRETATION The time series plot of the data is obtained and we can observe that the data consist of trend component (Constantly increasing nature) and random component and not the seasonal component (No predictive and repetitive nature). Hence the mathematical model of the data is additive in nature.

# 4. Checking the stationarity of the data - ADF Test

## H0 : the data set comes from a non-stationary process

## H1: the data set comes from a stationary process

```
adf.test(data)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  data
## Dickey-Fuller = -2.2811, Lag order = 9, p-value = 0.4593
## alternative hypothesis: stationary
```

# INTERPRETATION

ADF test is done for checking stationarity. p-value = 0.4593 > 0.05. Hence we are not rejecting H0. The data set is coming from a non-stationary process.

# 5. Steps invoved in extracting the stationary version of the data

```
#STEP 1 : Creating a time variable
t = seq(1:length(data))

#STEP 2 : Fitting a linear regression model
model = lm(data~t)
model
```

```
##
## Call:
## lm(formula = data ~ t)
##
## Coefficients:
## (Intercept)            t
##    156400.0        214.1
```

```
#STEP 3 : Extracting the residuals and checking for stationarity
res = resid(model)
adf.test(res)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res
## Dickey-Fuller = -2.2811, Lag order = 9, p-value = 0.4593
## alternative hypothesis: stationary
```

```
#STEP 4 : Creating a second order variable and checking for stationarity
t_2 = t^2
model2 = lm(data ~ t+t_2)
model2
```

```
##
## Call:
## lm(formula = data ~ t + t_2)
##
## Coefficients:
## (Intercept)            t          t_2
##   1.596e+05    1.907e+02    2.857e-02
```

```
res2 = resid(model2)
adf.test(res2)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res2
## Dickey-Fuller = -2.2686, Lag order = 9, p-value = 0.4646
## alternative hypothesis: stationary
```

```
#STEP 5 : Repeating the steps until stationarity is obtained
t_3 = t^3
model3 = lm(data ~ t+t_2+t_3)
model3
```

```
##
## Call:
## lm(formula = data ~ t + t_2 + t_3)
##
## Coefficients:
## (Intercept)            t          t_2          t_3
##   1.592e+05    1.965e+02    1.078e-02    1.451e-05
```

```
res3 = resid(model3)
adf.test(res3)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res3
## Dickey-Fuller = -2.1473, Lag order = 9, p-value = 0.516
## alternative hypothesis: stationary
```

```
#4th order
t_4 = t^4
model4 = lm(data ~ t+t_2+t_3+t_4)
model4
```

```
##
## Call:
## lm(formula = data ~ t + t_2 + t_3 + t_4)
##
## Coefficients:
## (Intercept)            t          t_2          t_3          t_4
##   1.534e+05    3.370e+02   -7.619e-01    1.485e-03   -9.000e-07
```

```
res4 = resid(model4)
adf.test(res4)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res4
## Dickey-Fuller = -3.3193, Lag order = 9, p-value = 0.06736
## alternative hypothesis: stationary
```

```
#5th order
t_5 = t^5
model5 = lm(data ~ t+t_2+t_3+t_4+t_5)
model5
```

```
##
## Call:
## lm(formula = data ~ t + t_2 + t_3 + t_4 + t_5)
##
## Coefficients:
## (Intercept)            t          t_2          t_3          t_4          t_5
##   1.540e+05    3.147e+02   -5.708e-01    8.619e-04   -4.200e-08   -4.201e-10
```

```
res5 = resid(model5)
adf.test(res5)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res5
## Dickey-Fuller = -3.0153, Lag order = 9, p-value = 0.1485
## alternative hypothesis: stationary
```

```
#6th order
t_6 = t^6
model6 = lm(data ~ t+t_2+t_3+t_4+t_5+t_6)
model6
```

```
##
## Call:
## lm(formula = data ~ t + t_2 + t_3 + t_4 + t_5 + t_6)
##
## Coefficients:
## (Intercept)            t          t_2          t_3          t_4          t_5
##   1.562e+05    2.050e+02    7.673e-01   -5.679e-03    1.496e-05   -1.657e-08
##         t_6
##   6.591e-12
```

```
res6 = resid(model6)
adf.test(res6)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  res6
## Dickey-Fuller = -3.6147, Lag order = 9, p-value = 0.03127
## alternative hypothesis: stationary
```

# INTERPRETATION

In order to transform the data into stationary series, method of least square is performed. For this in the first step a first order time variable is created and data is fitted into a linear regression model using this variable. Then residual of the model is extracted and checked for stationarity using ADF test. The data exhibited non-stationary behaviour. Hence a second order time variable is created and the same procedure was repeated until stationarity of the data is obtained. Here at the 6th level stationarity of the data is obtained (p-value = 0.0312 < 0.05).

# CONCLUSION

The chosen data was a time series data with trend component and it came from a non-stationary series. Method of least square was applied to extract the stationary version of the data and on the 6th step of model fitting the data attained stationarity with a p-value = 0.03127.