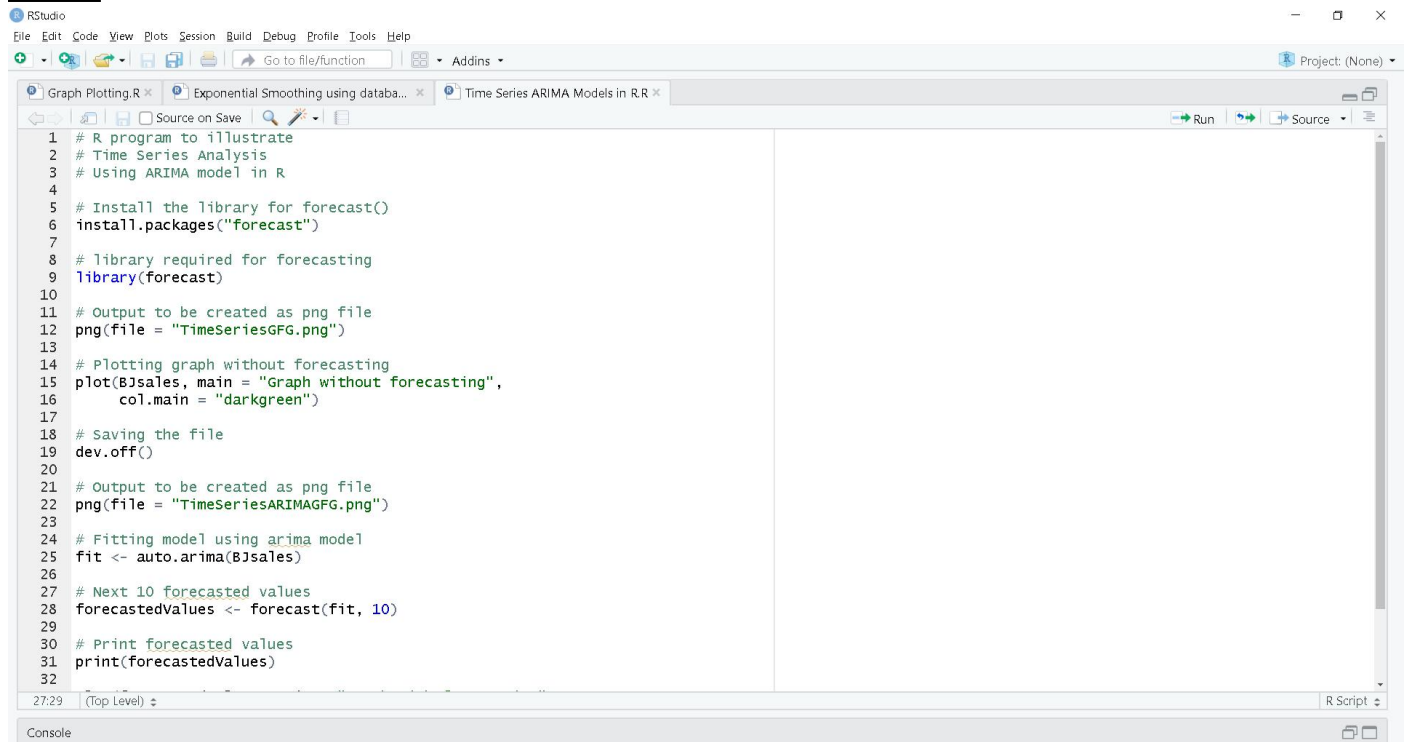


# ARIMA using Data set in R

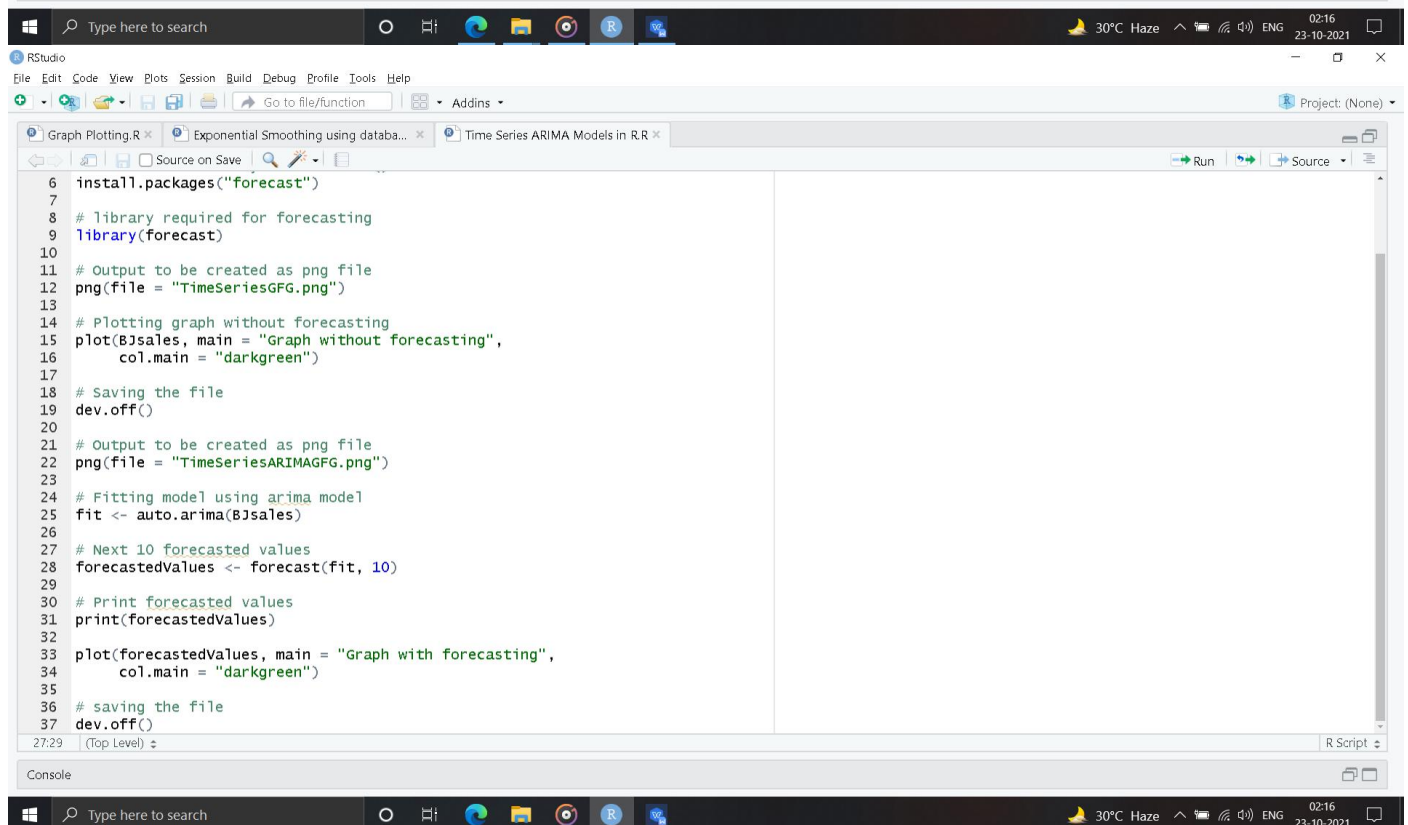
-- Jay Rathod

## Code:



This screenshot shows the RStudio interface with a script editor containing the first 32 lines of an R program. The script is titled 'Time Series ARIMA Models in R.R'. It includes comments for each step: installing the 'forecast' package, loading the library, setting the output file to 'TimeSeriesGFG.png', plotting 'BJSales' without forecasting, saving the file, and fitting an ARIMA model using 'auto.arima(BJSales)'. It also calculates the next 10 forecasted values and prints them.

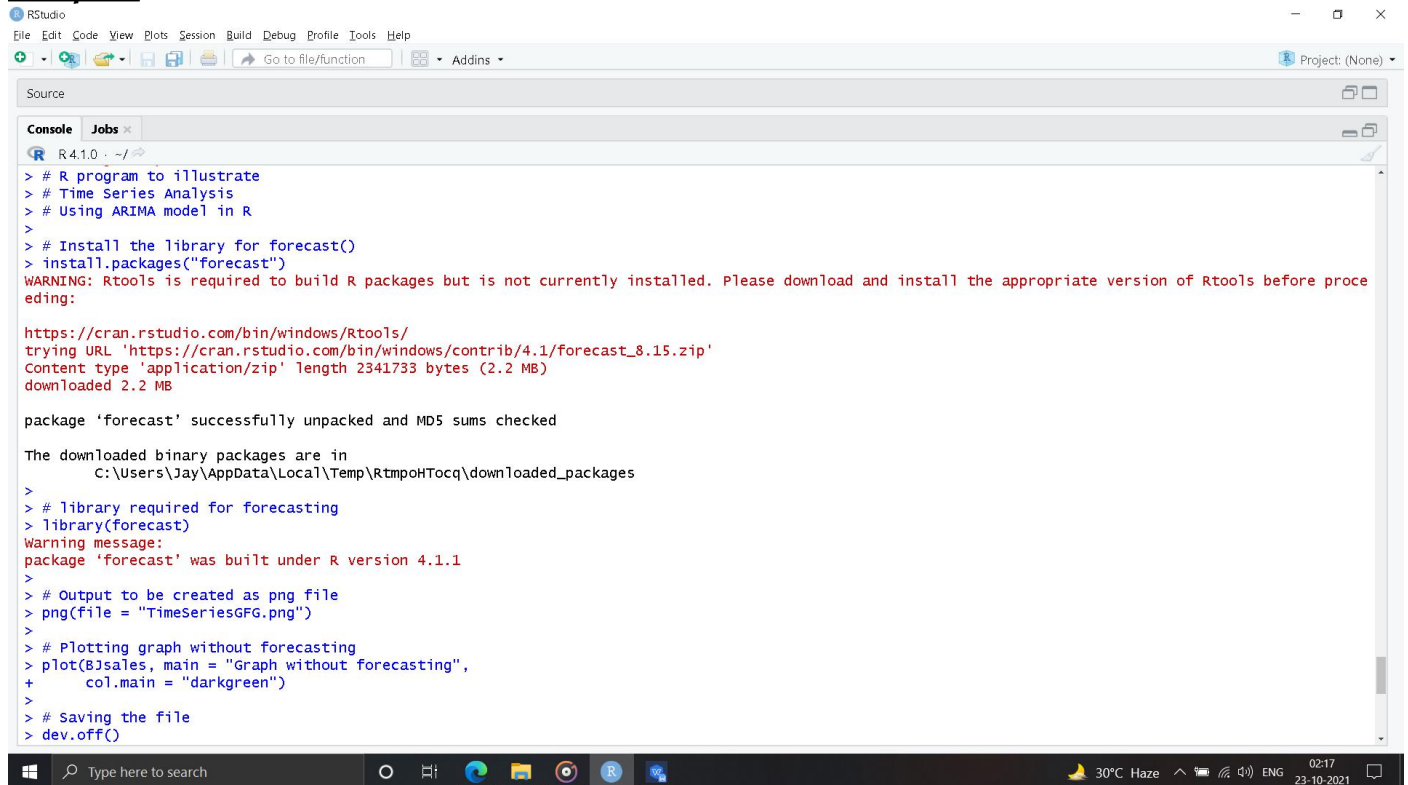
```
1 # R program to illustrate
2 # Time Series Analysis
3 # Using ARIMA model in R
4
5 # Install the library for forecast()
6 install.packages("forecast")
7
8 # library required for forecasting
9 library(forecast)
10
11 # Output to be created as png file
12 png(file = "TimeSeriesGFG.png")
13
14 # Plotting graph without forecasting
15 plot(BJSales, main = "Graph without forecasting",
16      col.main = "darkgreen")
17
18 # Saving the file
19 dev.off()
20
21 # Output to be created as png file
22 png(file = "TimeSeriesARIMAGFG.png")
23
24 # Fitting model using arima model
25 fit <- auto.arima(BJSales)
26
27 # Next 10 forecasted values
28 forecastedValues <- forecast(fit, 10)
29
30 # Print forecasted values
31 print(forecastedValues)
32
```



This screenshot shows the RStudio interface with a script editor containing the second part of the R program, starting from line 6. It continues from the previous script by installing the 'forecast' package, loading the library, setting the output file to 'TimeSeriesARIMAGFG.png', plotting the forecasted values, saving the file, and printing the forecasted values.

```
6 install.packages("forecast")
7
8 # library required for forecasting
9 library(forecast)
10
11 # Output to be created as png file
12 png(file = "TimeSeriesARIMAGFG.png")
13
14 # Plotting graph without forecasting
15 plot(BJSales, main = "Graph without forecasting",
16      col.main = "darkgreen")
17
18 # Saving the file
19 dev.off()
20
21 # Output to be created as png file
22 png(file = "TimeSeriesARIMAGFG.png")
23
24 # Fitting model using arima model
25 fit <- auto.arima(BJSales)
26
27 # Next 10 forecasted values
28 forecastedValues <- forecast(fit, 10)
29
30 # Print forecasted values
31 print(forecastedValues)
32
33 plot(forecastedValues, main = "Graph with forecasting",
34      col.main = "darkgreen")
35
36 # saving the file
37 dev.off()
```

# Output:



```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins Project: (None)

Source

Console Jobs x
R 4.1.0 ~ /
> # R program to illustrate
> # Time Series Analysis
> # Using ARIMA model in R
>
> # Install the library for forecast()
> install.packages("forecast")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/forecast_8.15.zip'
Content type 'application/zip' length 2341733 bytes (2.2 MB)
downloaded 2.2 MB

package 'forecast' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\Jay\AppData\Local\Temp\RtmpoHTocq\downloaded_packages
>
> # library required for forecasting
> library(forecast)
Warning message:
package 'forecast' was built under R version 4.1.1
>
> # Output to be created as png file
> png(file = "TimeSeriesGFG.png")
>
> # Plotting graph without forecasting
> plot(Bjsales, main = "Graph without forecasting",
+      col.main = "darkgreen")
>
> # Saving the file
> dev.off()
```

Windows taskbar: 30°C Haze, 02:17, 23-10-2021

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins Project: (None)

Source

Console Jobs
R 4.1.0 ~ /
> # Output to be created as png file
> png(file = "TimeSeriesGFG.png")
>
> # Plotting graph without forecasting
> plot(BJsales, main = "Graph without forecasting",
+      col.main = "darkgreen")
>
> # saving the file
> dev.off()
RStudioGD
2

>
> # Output to be created as png file
> png(file = "TimeSeriesARIMAGFG.png")
>
> # Fitting model using arima model
> fit <- auto.arima(BJsales)
>
> # Next 10 forecasted values
> forecastedValues <- forecast(fit, 10)
>
> # Print forecasted values
> print(forecastedValues)
  Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
151    262.8620 261.1427 264.5814 260.2325 265.4915
152    263.0046 260.2677 265.7415 258.8189 267.1903
153    263.1301 259.4297 266.8304 257.4709 268.7893
154    263.2405 258.5953 267.8857 256.1363 270.3447
155    263.3377 257.7600 268.9153 254.8074 271.8680
156    263.4232 256.9253 269.9211 253.4855 273.3608
157    263.4984 256.0941 270.9028 252.1744 274.8224
158    263.5647 255.2691 271.8602 250.8778 276.2516
159    263.6229 254.4529 272.7930 249.5986 277.6473
```

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins Project: (None)

Source

Console Jobs
R 4.1.0 ~ /
2

>
> # Output to be created as png file
> png(file = "TimeSeriesARIMAGFG.png")
>
> # Fitting model using arima model
> fit <- auto.arima(BJsales)
>
> # Next 10 forecasted values
> forecastedValues <- forecast(fit, 10)
>
> # Print forecasted values
> print(forecastedValues)
  Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
151    262.8620 261.1427 264.5814 260.2325 265.4915
152    263.0046 260.2677 265.7415 258.8189 267.1903
153    263.1301 259.4297 266.8304 257.4709 268.7893
154    263.2405 258.5953 267.8857 256.1363 270.3447
155    263.3377 257.7600 268.9153 254.8074 271.8680
156    263.4232 256.9253 269.9211 253.4855 273.3608
157    263.4984 256.0941 270.9028 252.1744 274.8224
158    263.5647 255.2691 271.8602 250.8778 276.2516
159    263.6229 254.4529 272.7930 249.5986 277.6473
160    263.6742 253.6474 273.7011 248.3395 279.0089
>
> plot(forecastedValues, main = "Graph with forecasting",
+      col.main = "darkgreen")
>
> # saving the file
> dev.off()
RStudioGD
2

>
```