Graphical user interface, application

Description automatically generated 2022

**Insurance Economics**

**Computer Assignments**

For the installation of R and RStudio see:

Torfs, P. and Brauer, C. “A (very) short introduction to R”, 3.3.2014.

(<https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf>)

R includes a line editor but RStudio is more convenient to use. RStudio splits the computer screen into four windows:

|  |  |
| --- | --- |
| **Editor window**   * This window provides a text editor. * Click Run or press CTRL+ENTER to send a line to the console window below where it is executed. * You may edit and save the programs that you write in this window. | **Workspace/history window**   * The workspace window shows the objects that R holds in its memory. You can edit their values by clicking on them. * The history window provides a protocol of your session. |
| **Console (command) window**   * Commands are executed in this window. * You may write directly into this window using the line editor. | **Plots/help/files/packages window**   * R displays plots in this window. * You may also use the help function, view files and install packages in this window. |

A more detailed introduction to R can be found in:

Venables W.N., Smith D.M. and the R Core Team, “An Introduction to R”, Version 4.2.0 (2022-04-22). (<https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>)

Some useful R commands:

getwd() Shows the current work directory.

setwd() Can be used to set the path to the work directory.

help() Use if you need information on some R command. For example, write help(plot) if you need information on the options of the plot command. ?plot also works and ??plot provides even more information.

R is case sensitive; that is, x and X or gdp and Gdp indicate different objects.

**Computer Assignment 1. South African Stock Market Index**

This assignment shows how to read data into R from a database, use time series data, and transform nominal variables into real ones.

All assignments should be handed in electronically. Use copy and paste to transfer R output into a word processor**.**

**Reading Data from the FRED Database**

The Federal Reserve Bank of St. Louis, which is a branch of the United States central bank, maintains the FRED (Federal Reserve Bank Economic Data) database.

Go to the website of FRED (<https://fred.stlouisfed.org/>) and use the search box to get the two time series:

* Total Share Prices for All Shares for South Africa, monthly

Change the format of the series to ‘Monthly, Index, 2015 = 100, not seasonally adjusted’.

* Consumer Price Index: All Items for South Africa, monthly

Give the time series convenient names (for example SmiZA and CpiZA) and then download them into a CSV (comma separated values) file. Name the data file DATAZA.csv. Make sure that the dates of both time series are correctly aligned, starting in January 1960. The top of your CSV data file should look as follows:

|  |  |  |
| --- | --- | --- |
| DATE | SmiZA | CpiZA |
| 1/01/1960 | 0.367696 | 1.345897 |
| 1/02/1960 | 0.362329 | 1.345897 |
| 1/03/1960 | 0.336768 | 1.350413 |
| 1/04/1960 | 0.306209 | 1.35493 |
| 1/05/1960 | 0.293459 | 1.359446 |
| 1/06/1960 | 0.285586 | 1.363963 |

**Plotting the Stock Price Index: Nominal Versus Real**

Start RStudio and read the data file into R:

DataZA <- read.csv("K:/Insurance/AIMS 2017/DATAZA.csv",header=TRUE)

Adjust the path of directories for the data file as needed. The header option is required because the first row of the data file includes the names of the time series.

Check whether the top of your data file is correct:

head(DataZA)

The attach command makes it possible to directly access data in the program code:

attach(DataZA)

The stock market index (SmiZA) and consumer price index (CpiZA) are stored as vectors in the data file. Convert those data vectors into time series objects with monthly frequency and a starting date of January 1960:

SmitsZA <- ts(SmiZA,frequency=12,start=c(1960,1))

CpitsZA <- ts(CpiZA,frequency=12,start=c(1960,1))

You are free to use other names for the time series than SmitsZA and CpitsZA.

Create a new time series by dividing the stock market index by the CPI for each date:

SmirtsZA <- SmitsZA/CpitsZA

Again, you may use another name than SmirtsZA for the new series.

Plot the real (inflation adjusted) stock price index. The x axis of the graph shows the dates because the data vector has been converted into a time series object.

plot.ts(SmirtsZA)

Compare with the nominal stock price index:

plot.ts(SmitsZA)

Finally, draw semi-logrithmic graphs:

plot.ts(log(SmirtsZA))

plot.ts(log(SmitsZA))

Copy and paste the output from the console window into a word processor and hand it in together with all four graphs.

**Summary of R commands**:

DataZA <- read.csv("K:/Insurance/AIMS 2017/DATAZA.csv",header=TRUE)

head(DataZA)

attach(DataZA)

SmitsZA <- ts(SmiZA,frequency=12,start=c(1960,1))

CpitsZA <- ts(CpiZA,frequency=12,start=c(1960,1))

SmirtsZA <- SmitsZA/CpitsZA

plot.ts(SmirtsZA)

plot.ts(SmitsZA)

plot.ts(log(SmirtsZA))

plot.ts(log(SmitsZA))