1.15 Exercise: Import data into R

(R version of Exercise 1.15)

Note: *Copying and pasting text (e.g. R code) from a pdf is not reliable. For that reason we have also provided this file in Word format (.docx) and also the code in a text file*

From Exercise 1.10 (R version) you have already seen how to make data sets in the FutureLearnData package available for analysis but we will reiterate the general pattern soon.

The # character in R: If you type or paste a line into the R Console window, R will ignore everything that comes after a "#" character. So # tells R that what follows is a comment left for human readers, not an instruction for R itself.

We will use this in the following as we talk about the pattern for making the data in a package, in our case the FutureLearnData package available for analysis.

library(FutureLearnData) # Load the package FutureLearnData

data(package= "FutureLearnData") # give me info about the data in the package FutureLearnData

I can copy and paste from this to get the names of data sets exactly right

data(olympics100m) # data(*dataset name*) makes it available for use

olympics100m # *saying the name of something causes it to display*

*OK to do here as this particular dataset is small*

*Otherwise use commands from Exercise 1.10 for displaying small parts of the data set*

Olympics100m # *this name is wrong because of the capital "O" so will give an error*

data(package= "FutureLearnData") # *curly quotation marks " from Word, not straight ones, ", so error*

***This whole block*** *of lines* ***can be copied and pasted as code. Try it***

Reading csv and tab-separated text files into R

It is simple to read rectangular data sets in csv or tab-separated text file formats into R. We will do it now.

1. **Download** the file ***Census at School-500.csv*** from https://www.stat.auckland.ac.nz/~wild/d2i/FutureLearn/

2. **Download** the file ***olympics100m.txt*** from https://www.stat.auckland.ac.nz/~wild/d2i/FutureLearn_TabTxt/

3. **Now try the following:** (Paste lines of code, or even several lines of code at a time, into the **R Console** window. See what they do.

# R CODE	**COMMENTARY**
# Import the file *Census at School-500.csv*	**read.csv** *is asking R to read a csv file* **file.choose()** *is telling R to throw up a browser window that will allow you to navigate to wherever you have stored* ***Census at School-500.csv*** *and open the file* **header = TRUE** *tells R that this file has a header line containing the names of the variables* **cas_500 =** *tells R to store the result as* ***cas_500***
cas_500 = read.csv(file.choose(), header = TRUE)	
cas_500[1:5, 1:9]	*Show me the* ***first 5 rows*** *and* ***9 columns*** *of* ***cas_500***
names(cas_500)	*Give me the* ***names of*** *all of* ***the variables*** *in* ***cas_500***
library(iNZightPlots)	*Need to load iNZightPlots package if not already done this session*
iNZightPlot(armspan, data= cas_500)	***Plot*** *the variable named* ***armspan*** *in* ***cas_500***
# Now import the file *olympics100m.txt*	*As above* but to read the tab-separated text file we use read**.table**, not read**.csv**. We include **sep="\t"** to tell R to look for tab characters as the separators between data fields *We store the result as store it as* ***Olymp_imp***
Olymp_imp = read.table(file.choose(), header = TRUE, sep="\t")	
names(Olymp_imp)	*Give me the* ***names of*** *all of* ***the variables*** *in*

iNZightPlot(YEAR, TIME, data= Olymp_imp)	***Olymp_imp*** ***Plot YEAR, TIME*** *in* ***Olymp_imp (gives a scatter plot of*** *y=**TIME** versus x=**YEAR**)*
?read.table	*Show me the* ***help file*** *for the function* ***read.table***
?read.csv	*Show me the* ***help file*** *for the function* ***read.csv.*** ***In this case the same help file covers both of these closely related functions***

[Note: Most actions in R are invoked by calling an R function. Function calls in R are of the form:

function.name(*list of function parameters separated by commas***)**

When you look at help files you will note in the "Usage" paragraph that a function will often have a large number of parameters. You do not need to include any parameters in your call to a function if that parameter is set equal to a value in this paragraph. That assigned value is the ***default value***. You do not need to include any parameter that has a default in your call unless you want to change its value from the default to something else.**]**

4. **Try some variations of the above,** e.g. plotting new variables, reading another data file.

5. **When you have finished, close R**. When it asks "***Save Workspace image?***", click, "**No**".

To discuss issues related to this Exercise,

go to **https://gitter.im/iNZightVIT/d2i-R-discussion**

To be able to post to the list you will have to set up a (free) account on ***Github***
https://github.com/login

If your question relates to an Exercise, say which one you are talking about!