thoughtfulbloke aka David Hood

Getting and Cleaning Week 1

A week with a bit of pain- mostly for those that need to configure there machine so that they are able to do various tasks- Read Excel files, download secure files etc.

- Part 1 : General Advice (https://thoughtfulbloke.wordpress.com/2015/08/31/hello-world/)
- Part 2 : Getting and Cleaning Week 1 (https://thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-week-1/)
- Part 3 : Getting and Cleaning Week 2 (https://thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-week-2/)
- Part 4 : Getting and Cleaning Week 3 (https://thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-week-3/)
- Part 5 : Getting and Cleaning Week 4 (https://thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-week-4/)
- Part 6 : Getting and Cleaning the Assignment (https://thoughtfulbloke.wordpress.com/2015/09/09/getting-and-cleaning-the-assignment/)

Quiz 1 advice

Question 1

This is a case of find the values in the codebook, and use that information to find the answer. However you may be trying to get the answer, and there are actually a lot of ways, the "checking your work step-by-step" advice in the thread linked to at the end of this post should be helpful for working out what step is causing the problem.

Because it is against the Coursera Honor Code for people to post quiz answers in the forum (and this includes explaining what the wrong answers are, or the one thing needed to make some not working code give the right answer). A good way to talk about this in the discussion forums is to

discuss how to approach a parallel problem. For example, it would be fine to discuss the steps to solving "How many households have Meals included in rent".

If you are having problems downloading the file, see the Downloading secure files in the Troubleshooting quizzes section of the general advice – getting started thread.

Question 2

This is based on the tidy data slide in the tidy data lecture, but also If you go to the site Jeff shows in the lecture, Hadley Wickham's tidy data paper is linked to. Everyone should read it. http://vita.had.co.nz/papers/tidy-data.pdf (http://vita.had.co.nz/papers/tidy-data.pdf) (This paper is actually directly referenced in the Week 3 revisiting and extending the idea of tidy data.)

As posting direct answers is against the honor code, we can't discuss the specific column and what its contents are. A better way to help is to list the options on a hypothetical "other" case, such as from the tidy data slide in the components of tidy data lecture we have

Each variable you measure should be in one column: This would be a problem if you are repeating variables in the data, or you are combining information about two different topics into one variable. Things in variables should be mutually exclusive of each other, not a bit from A and a bit from B.

Each different observation of that variable should be in a different row: This will be a problem if the data in the table is taking several lines to talk about the same observation.

There should be one table for each "kind" of variable: A scaled up version of the variable question, each table should be about a particular topic (with the variables being the attributes of that topic you are recording), so this could have problems if the data on one topic is split between several tables, or you are keeping information about different things in the one table.

If you have multiple tables, they should include a column in the table that allows them to be linked just like in databases for joint records together, each table should have a unique identifier for its records.

For those who want more practice, the Swirl tutorials on tidying up data are a great way to "level up" your understanding in a practical examples kind of way.

Question 3

This basically boils down to a question on configuring your computer to read in Excel files. To get the marks: download the file (as a binary file); read it in; run the line of code that gives the answer.

If using the xlsx package, with the read.xlsx command, you need to have Java installed. Because there are a lot of different operating systems I suggest you search the forum to see if there are relevant threads for yours and collaborate with fellow students.

But not everyone can install Java on their machines (it does need admin access). So here are a couple of alternatives for reading in Excel files:

- **gdata read.xls**

You can use the gdata package – read.xls command, however to use this depends on have Perl also being installed on the computer (and potentially gdata having the rights to update it) so people may be in the same place as with the Java problems. But a lot of linux distributions and Mac OS have Perl installed, so it might work for some people. You will need to vary the steps a bit from the question, so see below:

- **readxl read_excel**

so new, it doesn't have many features as I write this, so it needs similar steps to gdata

- **openxlsx read.xlsx**

You can use the openxlsx package – read.xlsx function to read in the Excel file just using R. You will need to vary the steps a tiny bit from the lecture: where you use the rowIndex setting in the xlsx package, you use the rows setting in openxlsx; where you use the colIndex setting in the xlsx package, you use the cols setting in openxlsx; When you use the sheet setting to specify which sheet, if you also have the xlsx package loaded, it may give an error about "multiple formal arguments", to get rid of the error specify you want read.xlsx from the openxlsx package by running the command as openxlsx::read.xlsx rather than read.xlsx (this avoids ambiguity).

- **Using gdata and readxl**

The xlsx package – read.xlsx command (the one that uses Java) and openxlsx have the row and column settings for only reading in the data from one part of a spreadsheet, the other methods don't, and while we could fix this by a lot of subsetting and column type conversion it is probably easier to do a two stage read- first we read in all the sheet, then we use part of the sheet to read in just the area we want. This is actually easier than fixing the type and name of columns individually after subsetting.

1. If using gdata read in the whole of sheet 1 using the function read.xls with setting header to FALSE and blank.lines.skip to FALSE, storing it in a variable called full. If using readxl read in the whole of sheet 1 using the function read_excel with setting col_names to FALSE, storing it in a variable called full.
2. Assuming you have rowIndex and colIndex defined as ranges as per the lecture slide, create a chunk with just the data you want with the line `chunk <- apply(full[rowIndex,colIndex],1, function(x){paste(x, collapse="\t")})`
3. Create a new variable called dat, using the read.table function with the three settings text=chunk, header=TRUE, and sep="\t".
4. Continue on with the question instructions.

Common errors in Question 3

Some kind of error about loading RJava means that you do not have the appropriate version of Java (32 or 64 bit) installed for you kind of R.

An error about being unable to open the file, and also being unable to open the file in other programs that understand Excel files, means that the file was not downloaded as a binary and has become corrupt. `mode = "wb"`

Question 4

This is a question where you really want to pay attention to the downloading files and checking your work, as described in the Getting started advice in the General forum.

You will learn a lot by checking step by step what is going on, and if you do the checking work things like reading the help for xmlParse gives "The name of the file containing the XML contents. This can contain ~ which is expanded to the user's home directory. It can also be a URL. See isURL."

and going to isURL further down the page gives

"indicates whether the file argument refers to a URL (accessible via ftp or http)"

which boils down to it does not support https connections directly, though you could download as a seperate file then read it in locally. So an https direct reading in is not going to be a happening thing. While you can use http, downloading it first means that you can look at the file in a text editor and go "yep, that is XML, so I know I am good up until this point", basically the general troubleshooting principle of checking your work at each step.

For those people struggling with what is going on with the XML package and the data, a good way of helping (which stays within the honor code) can be by providing a test case, that people can run through to understand things better

Assuming that xmlTreeParse has been used with useInternal=TRUE, you have got a fairly complex element that you then pull bits out of with xpathSapply, xmlTreeParse is the starting point of getting it into R. So as a small example of finding all the birds that are kiwis but not fruit that are kiwis:

```
xmldata <- "<xmldata>
<div><bird>kiwi</bird></div>
<div><bird>kiwi</bird></div>
<div><fruit>kiwi</fruit></div>
<div><bird>emu</bird></div>
</xmldata>"

writeLines(text = xmldata, con="things.xml")

file <- "things.xml"
dataXML <- xmlTreeParse(file, useInternal = TRUE)
color_vector <- xpathSApply(dataXML, "//div/bird", xmlValue)
length(color_vector[color_vector=='kiwi'])
```

(https://thoughtfulbloke.files.wordpress.com/2015/09/screen-shot-2015-10-08-at-5-06-10-pm.png)

But at the step where you pull data out with xpathSApply you could approach the problem in several different ways, depending on what tag information you are using to identify the areas.

Two fundamental questions that people struggling might want to consider:

What is my target? Am I looking for attribute or an element What is my target? XML general comes through as lists inside lists, which means you can explore by layers of subsetting, seeing what readInData[[1]] gives, reaching inside that to see what readInData[[1]][[1]] gives, and again with readInData[[1]][[1]][[1]], and similar by moving down the lists by swapping 1 for 2 etc.

Question 5

My interpretation of Quiz 1 Question 5 is that it is one of these programming assignment kind of questions where it doesn't matter exactly how your get to the end, so you are using this Hacking Skills mentioned in the Toolbox prerequisite course to look things up. Some Coursera courses have a mix of quiz questions and separate programming assignments, this one combines them in the quizzes.

Now, part of this question is just practicing using data tables, but part of it is practicing using the skills to solve a question, and there are a few standard ways people go wrong by trying to rush through the question or solve it without looking things up.

Because it uses data tables, you do actually need to install and load package before the fread command (if you get command not found error messages it either means you have made a typo or have not loaded the package with the library() command)

For example, the question asks *Which of the following is the fastest way to calculate the average value of the variable pwgtp15 broken down by sex*. I would suggest that it does not matter how blinding fast any of them are if it does not calculate the average value for each category of sex in the data. **CHECK THIS**, **CHECK YOUR ASSUMPTIONS ABOUT WHAT YOU ARE TESTING** As someone in an earlier session put it, we are in a Getting and Cleaning class so as a data cleaner need to check what the data is we are actually working with.

You might then (having ruled some entries out on logic grounds) be in a position of having a really fast computer (and in particular running Windows which gives less accurate timings) and find that one run of system time is giving you a tie. So you might read up on what system.time actually does, and decide the important bit is the user time, and because you have (hopefully) done the recommended background course of R Programming so know how to add things together and put things in a loop, you might decide to do the timing test 1000 times adding the results together and seeing if it gives a clear answer (it will). Alternatively, you looking things up might have lead you to a added library for R that basically does the same thing, an equally good solution. You are basically using the hacker problem solving skills talked about in the earlier courses to gather enough data from your own computer to get a reliable answer.

Now to help people get into the kind of mindset for approaching questions like this, while staying within the honor code, let's talk about hobbits:

We want to hold a race to determine the fastest hobbit, and the way we are measuring the race is we are holding it over a 1000 metres distance, and measuring the hobbits every metre. So we get together Frodo, Sam, Pippin, Merry, Gimli, and Legolas.

We now immediately disqualify Gimli and Legolas because they are not hobbits. And poor Sam is carrying all the packs so has problems in the race as well. We could help Sam by putting brackets around for suport {}, but we could just as easily leave him out of the race as even if he is set to race he is much, much slower and will be last.

Let's lets race each hobbit (I going to use a little pseudocode here because I feel if I just give the code in this section that basically gives away the question answer)

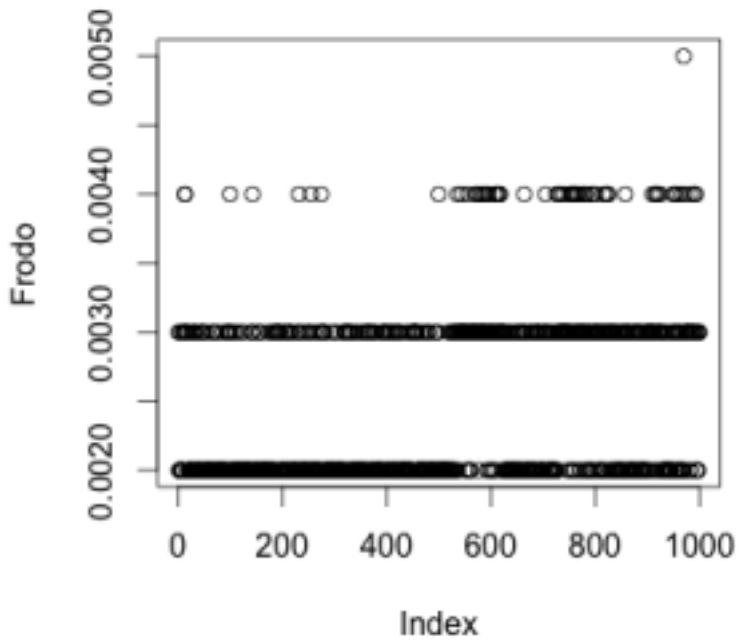
```
race = 1000
Frodo = replicate(race, function for measuring time(function being tested)[b
Pippin = replicate(race, function for measuring time(function being tested)[
Merry = replicate(race, function for measuring time(function beingtested)[bi
```

Now, starting off with R, you would have used a for loop, but once you get used to it there is a general rule of thumb that if you are using a loop in R there is probably a better way (I would expect most people doing this course would use a for loop, but you can use standard code without it or added libraries).

Now we have a 1000 measurements for each of the hobbits that were raced.

We can see how the measure times are based on the batches in the computers chip cycle if we just make a graph of individual results

```
plot(Frodo)
```



(https://thoughtfulbloke.files.wordpress.com/2015/09/frodo.png)

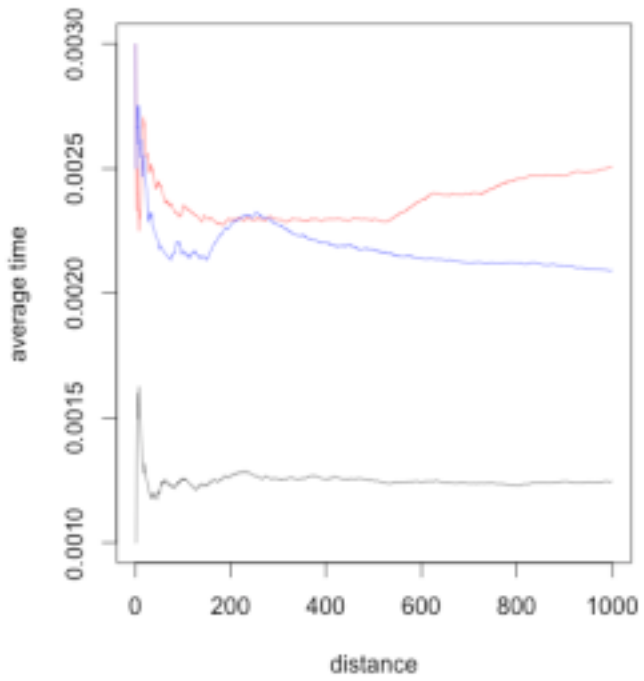
This shows use that when we measure speed on the computer, at a fine level of detail we are measuring a "statistical speed" based on how many cycles the instructions take.

But we are interested in the way cumulative average changes, so lets work out the way the cumulative average changes (disclaimer I just googled cumulative average in R for how to do this)

```
Frodo_av = cumsum(Frodo) / seq_along(Frodo)
Pippin_av = cumsum(Pippin) / seq_along(Pippin)
Merry_av = cumsum(Merry) / seq_along(Merry)
```

For making a graph, I wouldn't expect people to be taking it to this level until you are familiar with Exploratory Data Analysis. Also, this is going way, way, way beyond what is needed to answer the question.

```
topY = max(Frodo_av, Pippin_av, Merry_av) #making sure the y axis is the rig
lowY = min(Frodo_av, Pippin_av, Merry_av) #making sure the y axis is the rig
plot(Frodo_av, type="l", col="#FF000099", ylim=c(lowY,topY), xlab="distance"
lines(Pippin_av, col="#0000FF99")
lines(Merry_av, col="#00000099")
```



(https://thoughtfulbloke.files.wordpress.com/2015/09/hobbits.png)

As to which hobbit will be fastest if you race them, that will depend on the environment of the race, on foot Frodo is fastest (having done a lot more on foot travel) followed by Merry (who is bigger and stronger than Pippin) then Pippin. If you run the race in a horseback environment, Frodo has basically no experience of riding and comes last, so the winner is Merry (who spent more time with the Riders of Rohan) then Pippin second.

September 9, 2015 thoughtfulbloke Coursera, Getting and Cleaning, R

6 thoughts on “Getting and Cleaning Week 1”

1. *Luc L.* says:
   October 11, 2015 at 2:29 pm
   “If you are having problems downloading the file, see the Downloading secure files in the Troubleshooting quizzes section of the general advice – getting started thread.” That is from question 1 section at the end. I would like to have access to this information but cant find it. What is the url if there is one? Thanks.

   Reply
   - *thoughtfulbloke* says:
     October 12, 2015 at 1:42 am
     https://thoughtfulbloke.wordpress.com/2015/08/31/hello-world/ scroll down to the downloading secure files section (it is nearish the top)

     Reply
2. *Cj* says:
   December 2, 2015 at 2:09 am
   ohhh I was wondering why I couldn’t get this right… {slight edit from David Hood to avoid discussion of specific answer options}?

   Reply
   - *thoughtfulbloke* says:
     December 3, 2015 at 9:27 pm
     I’m not going to discuss specific answer options, but the general principle is that in order to be the fastest at calculating the mean for both genders, being part of the group of answers that calculate a mean for both genders (the hobbit set) is a requirement.

     Reply
     - *cj* says:
       December 3, 2015 at 9:52 pm
       Essentially, when I did the repetition thing, for all of the functions that actually calculated the mean properly … the one that was fastest, was not the correct answer.

     - *thoughtfulbloke* says:
       December 4, 2015 at 3:30 am
       The thing I suspect is it is not “the mean” the question asks for the mean by gender, so however many kinds of entries there are in the gender column you get a mean for each. That tends to be where people go wrong- it is not just that the code runs, it also needs to do the job it was supposed to.

       After that it is basically collecting enough data you have data rather than anecdote.