hamburger.r: Explanation of code

Goals of code:

- Read an excel sheet (.xls or .xlsx file)
- Wilcoxon rank sum test

Reading an excel sheet:
When you have a .xls or .xlsx file, there are two ways you could read this into R. One is to store a specific spreadsheet as a comma delimited file (.csv format). The other is to read the Excel workbook (.xlsx format file). The ability to read an Excel workbook is not part of base R. It is added to R by functions in an add-on package (also called library). There are many (nearly 10,000) add-on packages that do all sorts of different analyses. There are multiple packages that read .xls or .xlsx files. I demonstrate the `read_excel` function in the readxl package. This is the one I find most useful. You are welcome to use a different one.

The R archive site, called CRAN: comprehensive R archive network, is so active, it is distributed over 10's of mirror sites. I suggest you use the mirror site closest to you. For us, that is the one in IA.

Downloading a package: `install.packages('readxl')`
Before you can use any add-on library, you need to download it from CRAN, the Comprehensive R Archive Network. Since there are now so many packages, I find it easiest to download it by name. `install.packages()` installs the package you name. You will have to specify a mirror site (I recommend IA). The package is now downloaded and unzipped.

Note: the package name is in quotes.

If you want to look for a package in a menu, from the main menu: Packages / Install Package. Choose a mirror site (I recommend the IA one), then choose the readxl library.

Downloading the library (either by name or by menu) only needs to be done once.

Importing a .xlsx file: `library(readxl); read_excel()`
Before you can use a package, you have to enable it. That's what `library(readxl)` does.

Note: Here the filename is not in quotes.

Library() has to be done once each session before you can use the read.excel() function. If you get the error message: Error: could not find function "read.excel", you forgot to enable the library. The required argument is the file name, or replace the filename with choose.files() to use the mouse to select the file. The default is to read Sheet1, but you can specify the name of another sheet.

`read_excel()` also allows you to designate a specific range of cells to read. Default is to read all rows and all columns in the specified worksheet.

`read_excel()` will read both xlsx and xls format files. Just specify the appropriate filename.

The result from `read_excel()` is a tibble. This is an extension of the data.frame that we use to store data. You can treat the tibble just like a data frame.

Wilcoxon rank sum test: `wilcox.test(cfu~treatment, data=hb)`
The syntax is exactly like t.test: a formula with the response variable (here, cfu) $\sim$ the treatment variable (here, treatment), and the name of the data frame.

A couple of details: I prefer to not use a continuity correction, so I add correct=F. Lecture will discuss this. If the data set is small, I prefer the permutation test p-value, which you get by adding exact=T, as in the second line.

One slightly irritating limitation: the algorithm R uses to get the exact p-value does not work when there are tied values (two observations with the same response value). R will print a warning when it can't compute the exact p value. The exact p-value is well defined, even when there are ties. It's just that R won't calculate it.