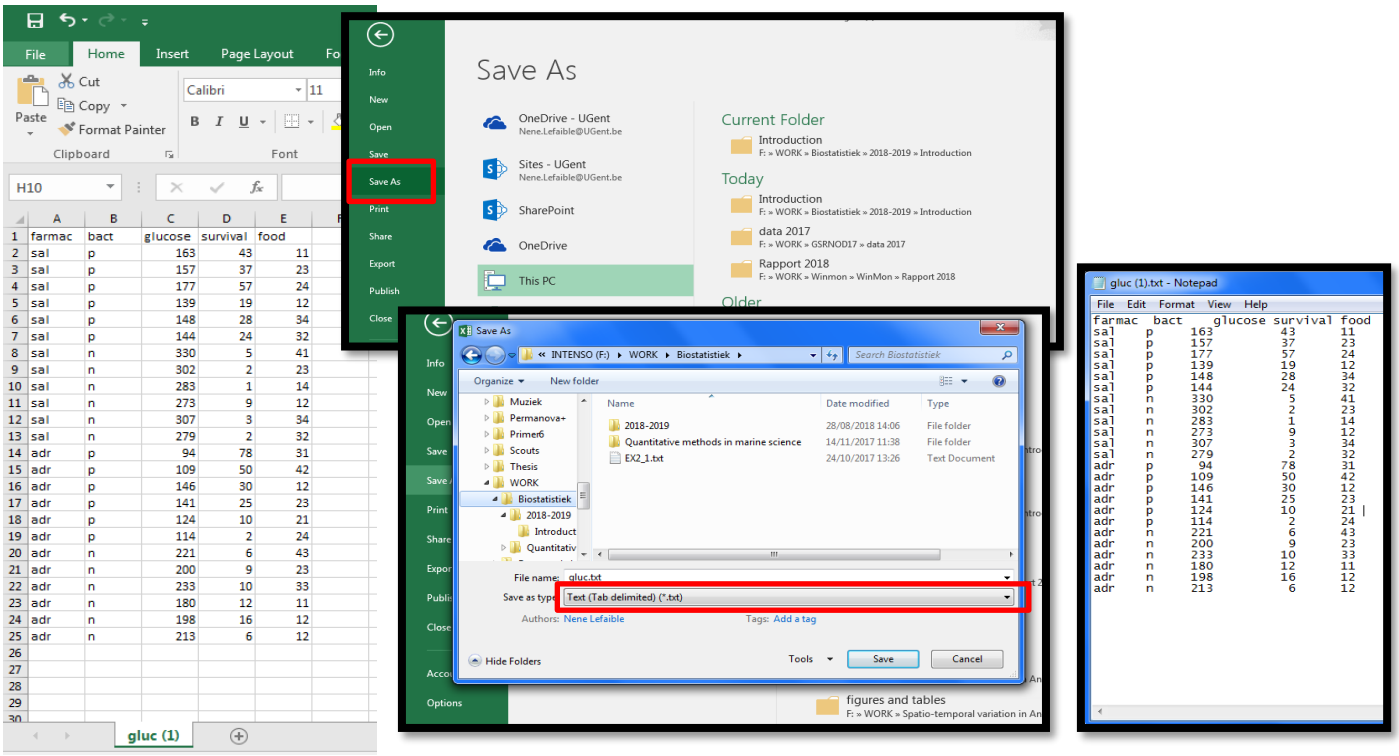
R Introduction to data frames and packages

Working with data frames

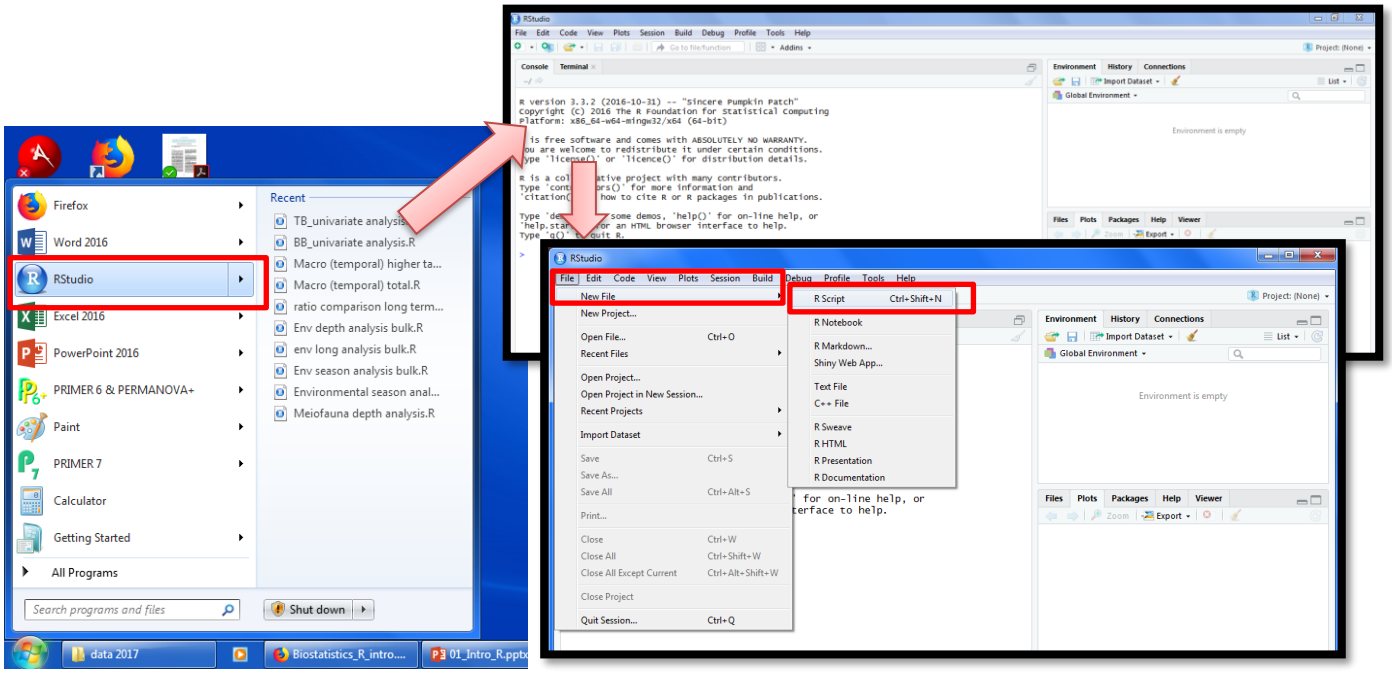
Key Points:

- R can only open datasets as ".txt" or ".csv" files
- R only recognizes "." As a decimal sign
- The first row usually contains the column names/ names of the variables that you are investigating and testing
- Your column names cannot contain white spaces eg. "glucose_level" instead of "glucose level"

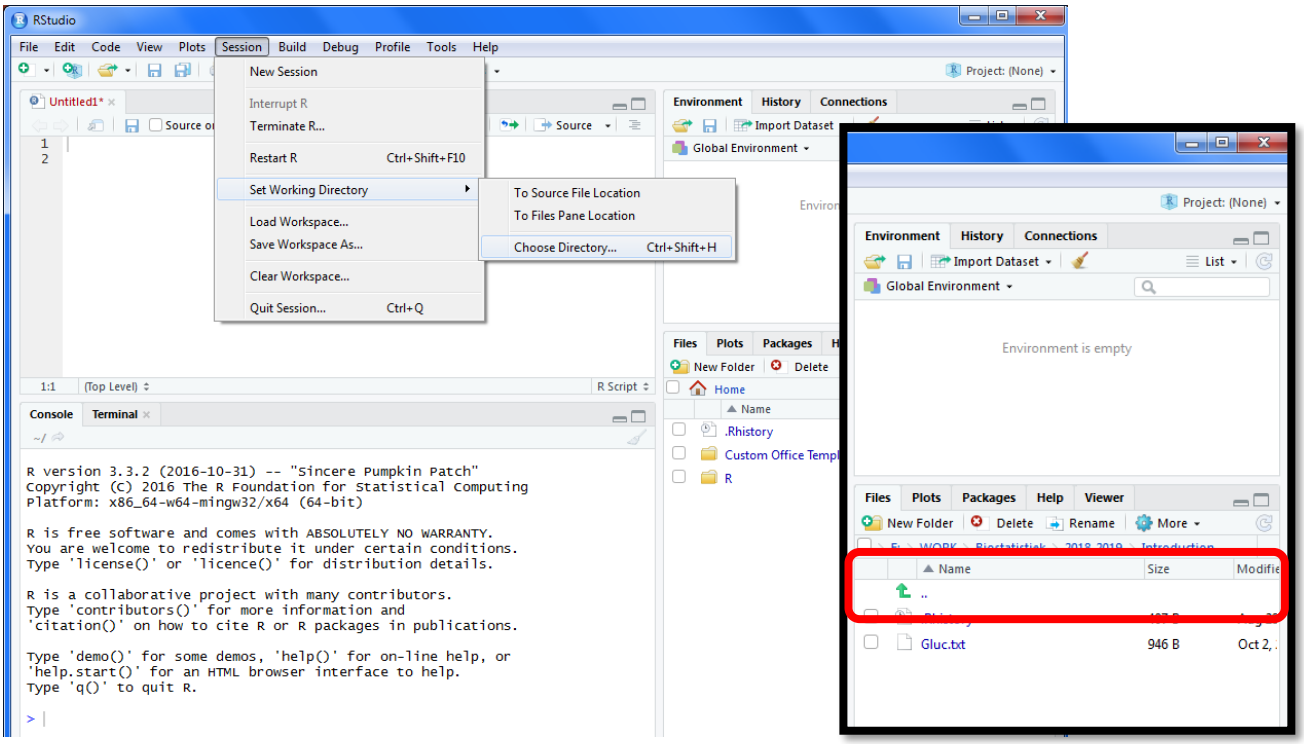
1. If you are working in excel, you can save your data into a text file as shown here:



2. To read a data frame, first create a new R-script within RStudio as shown here:

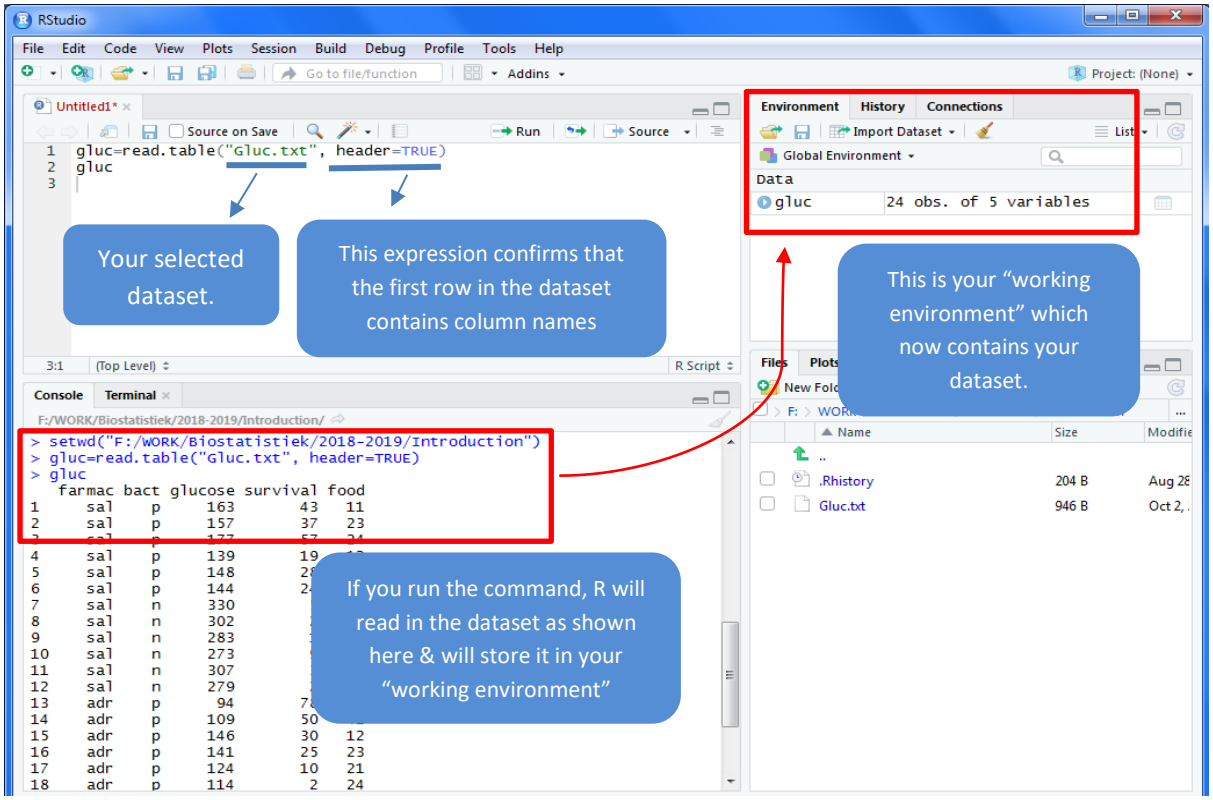


3. You can tell R where to find the data on your computer by setting the correct “working directory” as shown here and then browsing for the directory where you saved your files on your computer.



After choosing the directory, you will see your dataset on the right side within the “Files” folder.

4. Write the command line in the script to “read” your selected dataset as shown here. Make sure that your selected dataset is written exactly the same as the file you stored in the working directory including capitalization. So, if the file is stored as “Gluc.txt”, “gluc.txt” will give you an error message.



5. Once you have read in your file, you can start writing commands (“code”) to perform several actions on your data.

   Here are some examples of things you might want to do with your data

   a. You can select specific parts of your dataset = subsets
   b. You can perform calculations such as finding the mean, the standard deviation or the standard error
   c. You can perform statistical analyses such as t-tests, ANOVA or regression analyses
   d. You can create a variety of graphs such as X-Y plots, barplots, boxplots or many others…

Packages

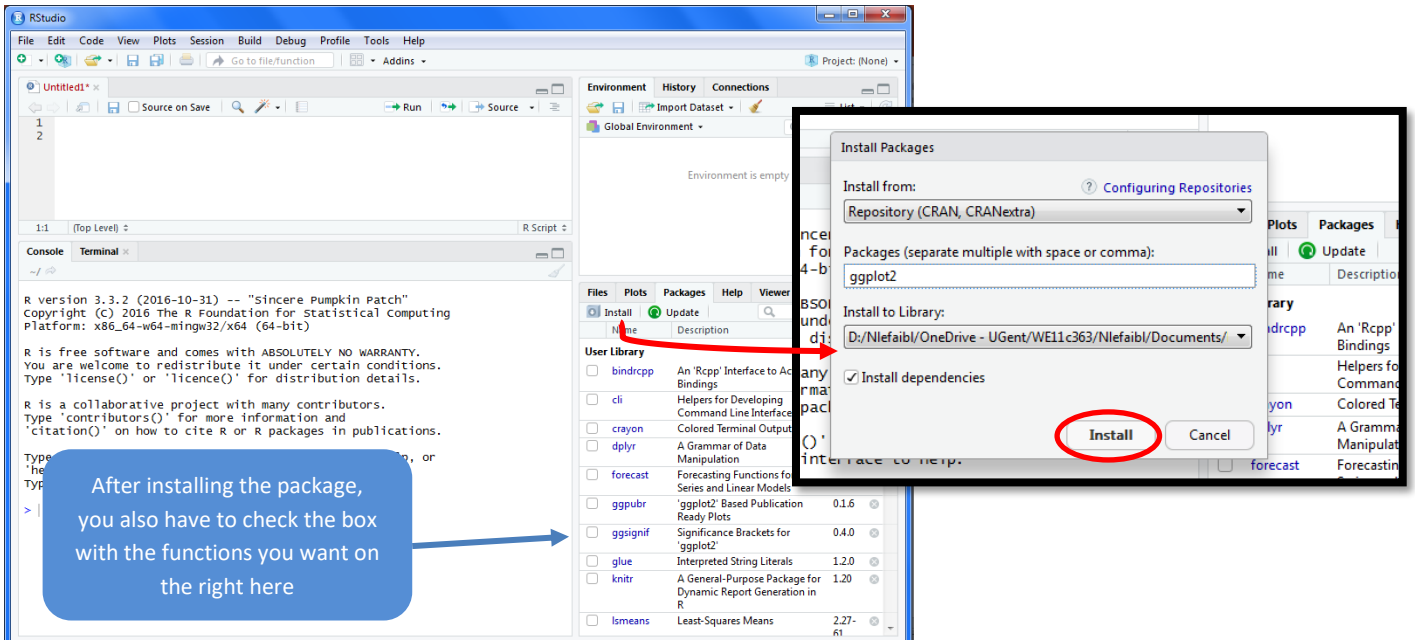
Many useful R function come in packages, free libraries of code written by R's active user community. To install an R package, open an R session and type at the command line

```
install.packages("<the package's name>")
```

R will download the package from CRAN, so you'll need to be connected to the internet. Once you have a package installed, you can make its contents available to use in your current R session by running

```
library("<the package's name>")
```

You can also install packages through the "packages panel on the right side of your interface as shown here:



Here are some examples of useful packages for different purposes:

To manipulate data

dplyr - Essential shortcuts for subsetting, summarizing, rearranging, and joining together data sets. dplyr is our go to package for fast data manipulation.

tidyr - Tools for changing the layout of your data sets. Use the gather and spread functions to convert your data into the tidy format, the layout R likes best.

stringr - Easy to learn tools for regular expressions and character strings.

lubridate - Tools that make working with dates and times easier.

To visualize data

ggplot2 - R's famous package for making beautiful graphics. ggplot2 lets you use the grammar of graphics to build layered, customizable plots.

ggvis - Interactive, web based graphics built with the grammar of graphics.

rgl - Interactive 3D visualizations with R

htmlwidgets - A fast way to build interactive (javascript based) visualizations with R. Packages that implement htmlwidgets include:

- leaflet (maps)
- dygraphs (time series)
- DT (tables)
- diagrammeR (diagrams)
- network3D (network graphs)
- threeJS (3D scatterplots and globes).

googleVis - Let's you use Google Chart tools to visualize data in R. Google Chart tools used to be called Gapminder, the graphing software Hans Rosling made famous in hie TED talk.

To model data

car - car's Anova function is popular for making type II and type III Anova tables.

mgcv - Generalized Additive Models

lme4/nlme - Linear and Non-linear mixed effects models

randomForest - Random forest methods from machine learning

multcomp - Tools for multiple comparison testing

vcd - Visualization tools and tests for categorical data

glmnet - Lasso and elastic-net regression methods with cross validation

survival - Tools for survival analysis

caret - Tools for training regression and classification models