Activity-Import and examine data in R

JB

updated 04/01/2021

[Download this file as an R markdown doc](import-examine-instructions-w-code.Rmd)

*working format*

1. *JB gives brief intro (<10min) to activity (how to locate materials, what data are we working with, any new packages/tools/functions)*
2. *Instructors then walk the room to help with problems, provide explanation, etc.*
   * *students don’t turn in anything for activities, so solutions are provided in the instructions*
3. *Brief sum-up (what was learned, common problems encountered) led by JB (10minutes?)*

## Learning Objectives

* Set up a project directory and create an R markdown file (\*.Rmd) to document your work
* Import data from csv (comma separated values)
* Get descriptives and characterize distributions using simple visuals:
  + histogram
  + quantile-quantile plot
  + box plot
* Run a normality test (shapiro-wilk)
* Transform a variable

## Step 1 - Get organized.

[Video resource (Prof Andy Field) - Working in RStudio](http://milton-the-cat.rocks/learnr/r/r_getting_started/#section-working-in-rstudio) - refer to the video if you want a review

#### 1.1 Set up a typical project work flow consisting of a project folder containing:

* \*.Rproj file
* “r\_docs” folder to store your lab notes in R markdown
* “data” folder to store data files for this activity
* {“images” folder to store plots} - *not needed for most activities*

#### 1.2 Start a new R markdown document to save your work on this activity

* name it something sensible
* in the “setup” code chunk load these packages with the library() function:
  + tidyverse, readr, ggplot2
* also in the “setup” chunk, paste in the line below (sets the base folder for code run from your markdown file):
  + knitr::opts\_knit$set(root.dir = rprojroot::find\_rstudio\_root\_file())
* run the setup chunk (click the play button in the top right of the chunk) - this action will load the libraries that will be used below
  + if you get an error that “there is no package called blahblahblah” then you either have a typo (remember R is case-sensitive), or you need to install the package (in the console type, e.g., install.packages("tidyverse") )

Show/Hide Solution



setup code chunk image

## Step 2 - Import datasets for this activity

#### 2.1 download these files:

* [“nhanes\_selectvars\_n500.csv”](../data/nhanes_selectvars_n500.csv)
* [“cort-hypothetical.txt”](../data/cort-hypothetical.txt)  
  (right-click, save as) and move each to the “data” folder in your project

[NHANES is a large public health dataset](https://www.rdocumentation.org/packages/NHANES/versions/2.1.0/topics/NHANES) - you will work with a small subset of cases and variables.  
The data in “cort-hypothetical.txt” is hypothetical salivary cortisol values (nmol/L) generated for this activity. It is in a tab-delimited format with each line containing ID and cortisol value.

#### 2.2 Import the data into R

* put the code to read the files in a new code chunk (name it “*Step2-load-datasets*” within your markdown file
* Use readr::read\_csv() to import “data/nhanes\_selectvars\_n500.csv”
* Workflow : Insert a “code chunk” into your R markdown doc and put the code there, then **use the “play” button to run chunks of code as you go along**. Don’t “knit” the file until you want to see all your work in one html/pdf doc.
* **Don’t use the View() function in any of your markdown code chunks**- it will make “knitr” stall when you try to knit your final doc. Instead, just **click on the variable** in your environment window to look at the data.

Show/Hide Solution

nhanes\_tib <- readr::read\_csv("data/nhanes\_selectvars\_n500.csv")

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## ID = col\_double(),  
## Gender = col\_character(),  
## Age = col\_double(),  
## Weight = col\_double(),  
## Height = col\_double(),  
## AlcoholDay = col\_double(),  
## Depressed = col\_character(),  
## BMI = col\_double(),  
## SleepHrsNight = col\_double(),  
## SleepTrouble = col\_character()  
## )

## Step 3 - Get descriptives and view the distribution

#### 3.1 Get mean, median, and sd for the variable “Height”

* use mean(), median(), and sd() functions and select the “Height” column using “$” like this mean(nhanes\_tib$Height, na.rm = True) (notice what happens if you leave out “na.rm = True”). It’s a good idea to also count cases - use n() and sum(is.na())
* Put the code in a code chunk called “*Step3.1-nhanes-height-all*”.
* To organize the stats in a neater way call the same functions from within the dplyr::summarise() function (see the solution code below)

Show/Hide Solution

nhanes\_tib %>% dplyr::summarise(  
 median = median(Height,na.rm = TRUE),  
 mean = mean(Height,na.rm = TRUE),  
 sd = sd(Height,na.rm=TRUE),  
 cases = n() - sum(is.na(Height))  
 ) %>%   
 knitr::kable(caption = "Height Descriptives- all individuals", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)

Height Descriptives- all individuals

median

mean

sd

cases

165.4

159.859

21.849

477

Bonus! Code to add normal 95% confidence interval around mean

add 95% confidence intervals using ggplot2::mean\_cl\_normal()

nhanes\_tib %>% dplyr::summarise(  
 median = median(Height,na.rm = TRUE),  
 mean = mean(Height,na.rm = TRUE),  
 ci.low = ggplot2::mean\_cl\_normal(Height)$ymin,  
 ci.upp = ggplot2::mean\_cl\_normal(Height)$ymax,  
 sd = sd(Height,na.rm = TRUE),  
 cases = n() - sum(is.na(Height))  
 ) %>%   
 knitr::kable(caption = "Height Descriptives- all individuals", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)

Height Descriptives- all individuals

median

mean

ci.low

ci.upp

sd

cases

165.4

159.859

157.894

161.825

21.849

477

#### 3.2 Look at the distribution of Height

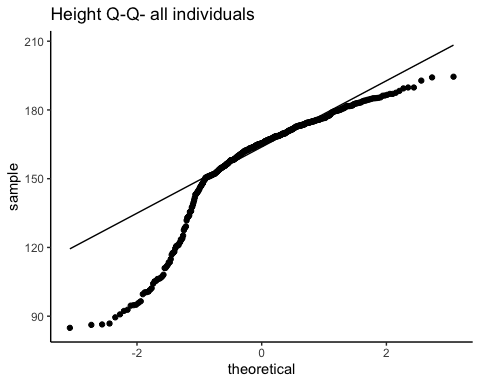
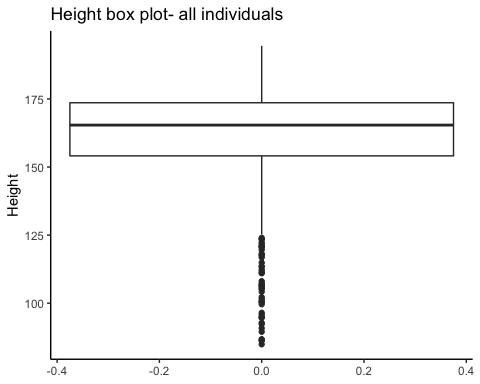
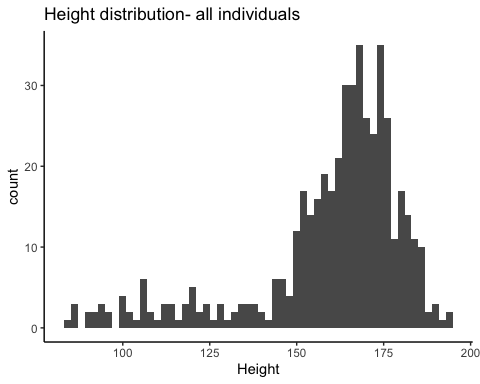
*Put your code in the same code chunk that you used for descriptive stats.*

* Plot a histogram and a boxplot of “Height” using nhanes\_tib %>% drop\_na(Height) %>% ggplot( aes(x=Height)) + geom\_histogram(binwidth=2). Try different values for the binwidth argument to see how it changes the plot.
* Make a boxplot of “Height” using nhanes\_tib %>% drop\_na(Height) %>% ggplot( aes(y=Height)) + geom\_boxplot()
* Make a quantile-quantile plot of Height. Use nhanes\_tib %>% drop\_na(Height) %>% ggplot( aes(sample=Height)) + geom\_qq() + geom\_qq\_line() Is Height normally distributed in this sample? Does this Q-Q plot look different from the SPSS one (notice the flipped axis labels)?
* Describe the distribution shape in your own words. Why does the boxplot show so many outliers at low values?
* Calculate the Shapiro-Wilk statistic for deviation from a normal distribution (use the function shapiro.test() on the Height data). shapiro.test() is an function that doesn’t accept a “data” argument, so to use the usual piping syntax, the code is nhanes\_tib %>% {shapiro.test(.$Height)} - or alternatively you can use shapiro.test(nhanes\_tib$Height). Although the first example looks unnecessarily complicated, the syntax can be very useful when there are additional piping steps before calling a function.
* What does the statistic tell you?

Show/Hide Solution

p1 <- nhanes\_tib %>% drop\_na(Height) %>%  
 ggplot( aes(x=Height)) + geom\_histogram(binwidth=2) + theme\_classic() +  
 labs (title = "Height distribution- all individuals")  
p2 <- nhanes\_tib %>% drop\_na(Height) %>%  
 ggplot( aes(y=Height)) + geom\_boxplot() + theme\_classic() +   
 labs (title = "Height box plot- all individuals")  
p3 <- nhanes\_tib %>% drop\_na(Height) %>%  
 ggplot( aes(sample=Height)) + geom\_qq() + geom\_qq\_line() + theme\_classic() +  
 labs (title = "Height Q-Q- all individuals")  
p1  
p2  
p3  
nhanes\_tib %>% drop\_na(Height) %>% {shapiro.test(.$Height)}

##   
## Shapiro-Wilk normality test  
##   
## data: .$Height  
## W = 0.84446, p-value < 2.2e-16



#### 3.3 Filter by Age

* Insert a new code chunk (call it *“Step3.3-nhanes-height-adults”*)
* Now let’s restrict the plots to individuals age 18 or older, make a new descriptives table, histogram, box plot, and q-q plot of height using just those individuals (use the filter() function to select cases, like this: nhanes\_tib %>% filter(Age>=18)).
* Re-calculate the Shapiro-Wilk statistic. What do the plots and shapiro test tell you?

Show/Hide Solution

nhanes\_tib %>% filter(Age>=18) %>% dplyr::summarise(  
 median = median(Height,na.rm = TRUE),  
 mean = mean(Height,na.rm = TRUE),  
 sd = sd(Height,na.rm=TRUE),  
 cases = n() - sum(is.na(Height))  
 ) %>%   
 knitr::kable(caption = "Height Descriptives (Age 18 or older)", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)  
p1 <- nhanes\_tib %>% drop\_na(Height) %>% filter(Age>=18) %>%  
 ggplot( aes(x=Height)) + geom\_histogram(binwidth=2) + theme\_classic() +  
 labs (title = "Height distribution- (Age 18 or older)")  
p2 <- nhanes\_tib %>% drop\_na(Height) %>% filter(Age>=18) %>%  
 ggplot( aes(y=Height)) + geom\_boxplot() + theme\_classic() +   
 labs (title = "Height box plot- (Age 18 or older)")  
p3 <- nhanes\_tib %>% drop\_na(Height) %>% filter(Age>=18) %>%  
 ggplot( aes(sample=Height)) + geom\_qq() + geom\_qq\_line() +   
 theme\_classic() + labs (title = "Height Q-Q- (Age 18 or older)")  
p1  
p2  
p3  
nhanes\_tib %>% drop\_na(Height) %>% filter(Age>=18) %>% {shapiro.test(.$Height)}

Height Descriptives (Age 18 or older)

median

mean

sd

cases

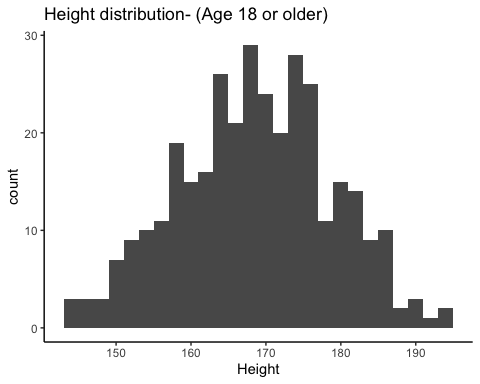
168.75

168.629

10.219

336

##   
## Shapiro-Wilk normality test  
##   
## data: .$Height  
## W = 0.99544, p-value = 0.4318



## Step 4 - Applying a mathematical transformation

#### Imagine a subset (N=400) of the NHANES participants gave saliva samples The lab sent you a file “cort-hypothetical.txt” containing an ID and cortisol measurement on each line, separated by a tab (“\t”).

4.1 Import the text file into a new dataframe/tibble (insert a new code chunk called *“Step4-cort”* ) - note that some values are “qns” which stands for “quantity not sufficient” and should be treated as missing values. Use readr::read\_delim() and set the argument “delim” to “ (for tab-delimited) and”na" to “qns”.

Show/Hide Solution

cort\_tib <- readr::read\_delim("data/cort-hypothetical.txt", delim = "\t",na = "qns")

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## ID = col\_double(),  
## cortisol\_baseline = col\_double()  
## )

4.2 Now check out the distribution of “cortisol\_baseline” using what you’ve learned so far.

* Describe the distribution of cortisol\_baseline in your notes.

Show/Hide Solution

cort\_tib <- readr::read\_delim("data/cort-hypothetical.txt", delim = "\t",na = "qns")

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## ID = col\_double(),  
## cortisol\_baseline = col\_double()  
## )

cort\_tib %>% dplyr::summarise(  
 median = median(cortisol\_baseline,na.rm = TRUE),  
 mean = mean(cortisol\_baseline,na.rm = TRUE),  
 sd = sd(cortisol\_baseline,na.rm=TRUE),  
 cases = n() - sum(is.na(cortisol\_baseline))  
 ) %>%   
 knitr::kable(caption = "Cort Descriptives", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)  
p1 <- cort\_tib %>% drop\_na(cortisol\_baseline) %>%  
 ggplot( aes(x=cortisol\_baseline)) +  
 geom\_histogram(bins=30) + theme\_classic() +  
 labs (title = "Cort distribution")  
p2 <- cort\_tib %>% drop\_na(cortisol\_baseline) %>%  
 ggplot( aes(y=cortisol\_baseline)) + geom\_boxplot() +   
 theme\_classic() + labs(title = "Cort Box Plot")  
p3 <- cort\_tib %>% drop\_na(cortisol\_baseline) %>%  
 ggplot( aes(sample=cortisol\_baseline)) + geom\_qq() +  
 geom\_qq\_line() + theme\_classic() +  
 labs(title = "Cort Q-Q Plot")  
p1  
p2  
p3  
cort\_tib %>% drop\_na(cortisol\_baseline) %>% {shapiro.test(.$cortisol\_baseline)}

Cort Descriptives

median

mean

sd

cases

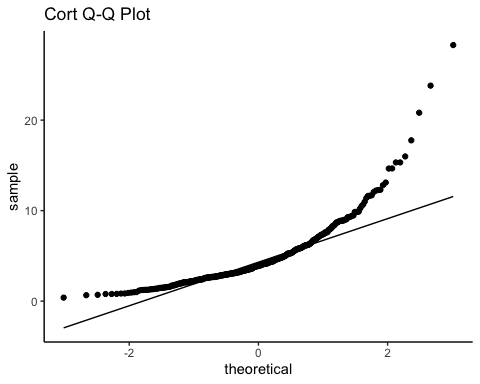
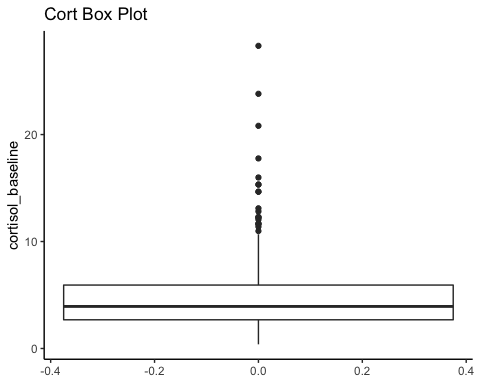
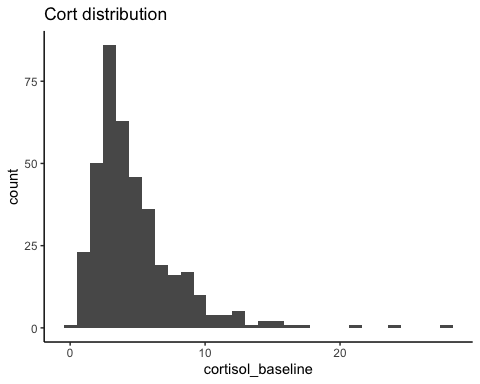
3.931

4.796

3.364

390

##   
## Shapiro-Wilk normality test  
##   
## data: .$cortisol\_baseline  
## W = 0.80483, p-value < 2.2e-16



4.3 Log transformation of a variable can be useful for some measurements with positively skewed distributions, and it is a common practice with salivary cortisol.

* Use the function “log” to create a new column in your cortisol dataframe, containing the natural logarithm of each cortisol\_baseline measurement, like this `. Then, describe the distribution of the log-transformed cortisol\_baseline.

Show/Hide Solution

cort\_tib <- cort\_tib %>%   
 mutate(ln\_cortisol\_baseline=log(cortisol\_baseline))  
cort\_tib %>% dplyr::summarise(  
 median = median(ln\_cortisol\_baseline,na.rm = TRUE),  
 mean = mean(ln\_cortisol\_baseline,na.rm = TRUE),  
 sd = sd(ln\_cortisol\_baseline,na.rm=TRUE),  
 cases = n() - sum(is.na(ln\_cortisol\_baseline))  
 ) %>%   
 knitr::kable(caption = "Log Cort Descriptives", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)  
p1 <- cort\_tib %>% drop\_na(ln\_cortisol\_baseline) %>%  
 ggplot( aes(x=ln\_cortisol\_baseline)) +  
 geom\_histogram(bins=30) + theme\_classic() +  
 labs (title = "Log Cort distribution")  
p2 <- cort\_tib %>% drop\_na(ln\_cortisol\_baseline) %>%  
 ggplot( aes(y=ln\_cortisol\_baseline)) +   
 geom\_boxplot() + theme\_classic() +   
 labs (title = "Log Cort box plot")  
p3 <- cort\_tib %>% drop\_na(ln\_cortisol\_baseline) %>%  
 ggplot( aes(sample=ln\_cortisol\_baseline)) +   
 geom\_qq() + geom\_qq\_line() + theme\_classic()  
p1  
p2  
p3  
cort\_tib %>% drop\_na(ln\_cortisol\_baseline) %>% {shapiro.test(.$ln\_cortisol\_baseline)}

Log Cort Descriptives

median

mean

sd

cases

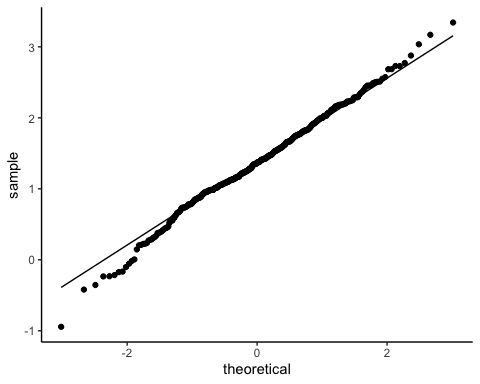
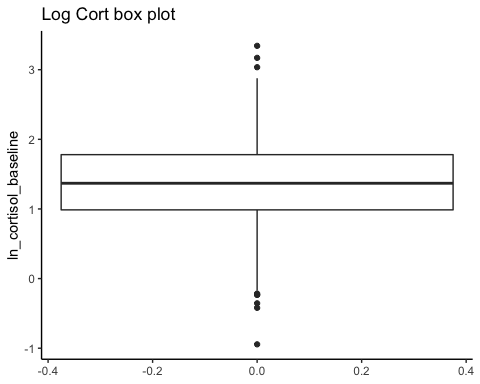
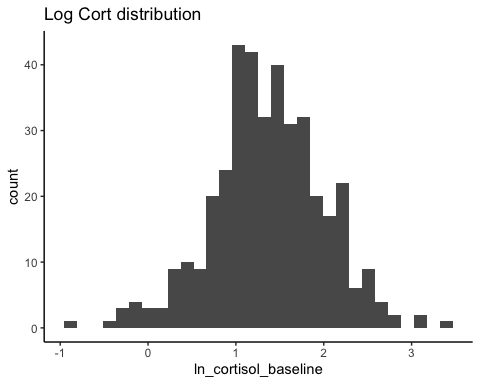
1.369

1.369

0.637

390

##   
## Shapiro-Wilk normality test  
##   
## data: .$ln\_cortisol\_baseline  
## W = 0.99414, p-value = 0.1394



## 5.0 Mini-challenge (if you have more time)

* Now re-do descriptives and a histogram and boxplot of Height from the NHANES data set, but group by Gender, so that you have 1 histogram plot (where different Genders get color-coded) and 1 boxplot (with Gender on the x-axis). Keep it restricted to Age 18+ only. Do you see any outliers (according to the boxplot threshold), when the data are split by Gender?
* Hints:
  + you can use the “fill” argument in aes()to map a variable onto the fill color chart component (called an “aesthetic” in ggplot language) - use position=“identity” and alpha=.5 so that the histogram bars for different Gender are overlayed (rather than stacked)
  + you can use the “x” axis argument in aes() to map a variable onto the x-axis (for the boxplot)
* **Things to pay attention to:** did you see a warning like “Removed {X} rows containing non-finite values” above the plots? What does that mean? How can you adjust your code so that you don’t get the warning?
* Experiment with different settings (position, alpha, theme) if you finish early

Show/Hide Solution

# nhanes\_tib$Gender <- factor(nhanes\_tib$Gender) # its a good idea to make sure Gender is treated correctly, but not actually needed here  
nhanes\_tib %>% drop\_na(Height,Gender) %>% filter(Age>=18) %>%   
 group\_by(Gender) %>%   
 dplyr::summarise(  
 median = median(Height,na.rm = TRUE),  
 mean = mean(Height,na.rm = TRUE),  
 sd = sd(Height,na.rm=TRUE),  
 cases = n()  
 ) %>%   
 knitr::kable(caption = "Height Descriptives by Gender (Age 18+)", digits = 3) %>%   
 kableExtra::kable\_styling(full\_width = FALSE)  
p1 <- nhanes\_tib %>% drop\_na(Height,Gender) %>% filter(Age>=18) %>%  
 ggplot( aes(x=Height, fill=Gender)) +   
 geom\_histogram(position="identity",alpha=.5,binwidth=2) +   
 theme\_classic() + labs (title = "Height Distribution by Gender (Age 18+")  
p2 <- nhanes\_tib %>% drop\_na(Height,Gender) %>% filter(Age>=18) %>%  
 ggplot( aes(y=Height, x=Gender)) + geom\_boxplot() + theme\_classic() +   
 labs (title = "Height Distribution by Gender")  
p1  
p2

Height Descriptives by Gender (Age 18+)

Gender

median

mean

sd

cases

female

162.3

161.745

7.649

167

male

175.4

175.433

7.533

169

