**R AND RSTUDIO INSTALLATION AND TUTORIAL**

TEMPLATE: Please copy content/questions to your own document.

Go to File → Make a Copy → Save to your own Drive.

Your name:

**INSTALLATION – BEFORE COMING TO CLASS**

**EXPLORING R – IN CLASS**

(Slide 5 - *Helpful Commands #1*)  
Test your console by typing print("Hello World!") in the console and pressing Enter..

(Slide 6 - *Writing and Executing Code*)  
We will be saving and executing code in an R script file – within the *File Window*.

Open a new R script file.

Save your file as “R\_intro\_<first name>\_<last name>.R in a spot on your computer that you will remember.

(Slide 7 - *Loading Data*)  
Type in the R script file:

air\_quality\_data\_full <- airquality

and press **Cmd + Enter** to execute the line.

(Slide 8 - *Set Working Directory*)  
Type the command getwd() in your command line to see what your current working directory is. What folder is R currently looking at (the last folder in the filepath)?

In your Documents directory (using Finder or Files Viewer, not RStudio), create a folder called “RStudio”. Within the “RStudio” folder, create another folder called “ENGS 37 Air Quality”.

In RStudio Console, run the command setwd() to set your working directory to end at the “ENGS 37 Air Quality” folder. Example on a macOS:

setwd("/Users/Andrew/Documents/RStudio/ENGS 37 Air Quality")

Folders are separated by forward slashes, and the order starts at a top level folder.

(Slide 10 - *Viewing Data*)  
Learn a little about the data with these useful commands:  
  
air\_quality\_data\_full – lists the entire dataframe for you.  
How many entries are there in total? Find line 43 and paste it below, what day and month is it? What is the wind speed on that day (with units)?

|  |
| --- |

head(air\_quality\_data\_full).  
How many rows does the head function show? Paste your output below.

|  |
| --- |

colnames(air\_quality\_data\_full)  
Paste your output below. What character surrounds the column names?

|  |
| --- |

str(air\_quality\_data\_full)  
Paste your output below. There is an additional column here that shows the words “int” or “num” for each variable, what do “int” and “num” mean?

|  |
| --- |

length(air\_quality\_data\_full$Ozone) – $ notation denotes columns.  
Paste your output, what does this number correspond to?

|  |
| --- |

(Slide 11 - *Cleaning Data*)  
We will only be using 2 variables (Ozone and Month). Thus, we will take a subset of this dataset:  
  
Execute this line:  
  
air\_quality\_data <- subset(air\_quality\_data\_full, select=c(Ozone, Month))

(Slide 12 - *Verifying Data*)  
Make sure you have **2** variables/columns (**Ozone** and **Month**) and **153** lines of data. Sometimes a line gets lost due to programming or file formatting errors; always double check that your data are correct before proceeding.  
  
In the *Environment* tab/window/panel, double-click on air\_quality\_data.  
  
This will open up the data sheet in the Viewer panel. In addition, what command(s) that you already learned can you use to show that you only have Ozone and Month now?

|  |
| --- |

(Slide 14 - *Plot 1: Histogram*)  
Let’s check what hist can do for us. Execute this line:  
  
help(hist)  
  
hist can take many different arguments, but at least one is required: *x*. The other arguments that you should know are *breaks*, *main*, *xlab*, *ylab*, *xlim*, and *ylim*. From the help function, summarize what these arguments represent and used for:  
x:   
breaks:   
main, xlab, ylab:

xlim, ylim:

|  |
| --- |

(Slide 14 - *Plot 1: Histogram*)  
Create a histogram for ALL of the months data on ozone, using the default options:  
  
hist(air\_quality\_data$Ozone)  
  
Your plot will appear in the Plots panel. Paste your plot → oh! Next slide or next step in the full assignment will explain how to save/copy plots.

|  |
| --- |

(Slide 16 - *Plot 1: Histogram (con’t)*)  
This code includes an x-axis label, but deletes the graph’s title because it won’t be needed once you write a good caption.  
  
hist(air\_quality\_data$Ozone, xlab="ppb", main="")  
hist(air\_quality\_data$Ozone, xlab="ug/L", main="")  
  
Run both commands and paste your plots. Which argument deleted the title?

|  |
| --- |

(Slide 16 - *Plot 1: Histogram (con’t)*)  
To make the notation prettier you can correspond the expression (value) to a variable name, and use it later.  
  
hist(air\_quality\_data$Ozone, xlab=expression(paste(mu,"g/",L)), main="")  
  
Run the command and paste your plot. The expression command creates an expression object and assigns it to xlab, so what does the paste function do?

|  |
| --- |

(Slide 17 - *Plot 1: Histogram (con’t)*)  
Change the number of bins in a histogram. R determines the number of bins (or breaks) to put in a histogram that makes it look "prettiest" using something called the Sturges method. Alternatively, you can tell R how many bins you would like it to create:  
  
hist(air\_quality\_data$Ozone, xlab="ppb", main="", breaks=22)  
hist(air\_quality\_data$Ozone, xlab="ppb", main="", breaks=5)  
  
Choose a number of bins/breaks you think is most appropriate, paste that plot, and explain why that number is most appropriate.

|  |
| --- |

(Slide 18 - *Plot 1: Histogram (con’t)*)  
Notice that you can change the look of the histogram by changing the range of values shown on the x-axis:  
  
hist(air\_quality\_data$Ozone, xlab="ppb", main="", breaks=22, xlim=c(0, 200))  
  
Choose an xlim that you think is best, explain why, and paste your plot.

|  |
| --- |

(Slide 18 - *Plot 1: Histogram (con’t)*)  
Play around with the other options (and more that you find in the help documentation for hist()): create a plot that you think is best, paste it below, and describe the distribution you see (i.e., shape, center, spread)**.**

Hint: hist() has lots of options, so look at its help page for details.

|  |
| --- |

(Slide 19 - *Refining and Segmenting Data*)  
Let’s create a histogram on ozone for each month.  
  
Brute-force method:  
hist(air\_quality\_data$Ozone[1:31], xlab="ppb", main="May")  
hist(air\_quality\_data$Ozone[32:61], xlab="ppb", main="June")  
hist(air\_quality\_data$Ozone[62:92], xlab="ppb", main="July")  
hist(air\_quality\_data$Ozone[93:123], xlab="ppb", main="August")  
hist(air\_quality\_data$Ozone[124:153], xlab="ppb", main="September")  
  
Logical selection way:  
hist(air\_quality\_data$Ozone[air\_quality\_data$Month==5], xlab="ppb", main="May")

What goes inside the square brackets after air\_quality\_data$Ozone (looking for a general answer)? Think about what the code in the brackets does.

|  |
| --- |

Run air\_quality\_data$Month==5 alone, what does this return (describe the output)?

|  |
| --- |

Now, explain how the logical selection method works, and why this might be better than a brute-force approach.

|  |
| --- |

Paste a plot for each month and describe the distribution (i.e., shape, spread, outliers) of each plot.

|  |
| --- |

(Slide 22 - *Installing and Loading lattice*)  
Execute the following lines:  
  
install.packages("lattice") **library**(lattice)  
  
*IMPORTANT:* After installing lattice, you will not have to install it again – but you will need to call it in your script via the **library** function to use its commands.

(Slide 23 - *Segmenting with lattice*)  
Describe at least one similarity and one difference between the new histogram function and the old hist function.  
  
You may use the following function call to help out.  
? histogram

|  |
| --- |

(Slide 23 - *Segmenting with lattice*)  
The following arguments will be needed for histogram plots: *x*, *data*, and *type*.

From the previous step’s help type function, summarize what these arguments represent and used for:  
  
*x*:   
*data*:   
*type*:

|  |
| --- |

(Slide 23 - *Segmenting with lattice*)  
Execute the following lines *one at a time*:  
  
histogram(~ Ozone | Month, data=air\_quality\_data)  
histogram(~ Ozone | Month, data=air\_quality\_data, type="count")  
  
Notice the difference between the two histograms? Paste your plots and describe the difference.

|  |
| --- |

(Slide 24 - *Custom Graphics*)  
We will be using the par function (feel free to check it out: ? par ).  
  
Now we will set some graphical parameters ourselves. Build the figure we want:  
  
par(mfrow=c(5,1), mar=c(4,2,2,2)+0.01, oma=c(2,2,0,0))  
hist(air\_quality\_data$Ozone[1:31], xlab="", main="May", xlim=c(0, 200))  
hist(air\_quality\_data$Ozone[32:61], xlab="", main="June", xlim=c(0, 200))  
hist(air\_quality\_data$Ozone[62:92], xlab="", main="July", xlim=c(0, 200))  
hist(air\_quality\_data$Ozone[93:123], xlab="", main="August", xlim=c(0, 200))  
hist(air\_quality\_data$Ozone[124:153], xlab="", main="September", xlim=c(0, 200))  
mtext("ppb", 1, outer=TRUE, cex=1.5)  
mtext("Frequency", 2, outer=TRUE, cex=1.5)  
  
Paste your final plot and explain at least two pros and two cons of this plot compared to the earlier ones.

|  |
| --- |

(Slides 25-26 - *Box Plots*)  
Create box plots for the whole dataset and for each month.  
  
boxplot(air\_quality\_data$Ozone, xlab="ppb", main="")  
boxplot(air\_quality\_data$Ozone ~ air\_quality\_data$Month, xlab="ppb", ylab="Month", main="")  
  
*Trouble?* If you are having trouble plotting the boxplot due to the previous par window formatting, run this before the boxplots:  
par(mfrow=c(1,1), mar=c(4,2,2,2)+0.01, oma=c(2,2,0,0))

Paste your plots. Describe what you see in the box plots, and how this is similar or different from the histograms.

|  |
| --- |

Which months are captured well by the histogram? By the boxplot? Which are captured poorly by either or both of these months? Is there perhaps a third option worth pursuing?

|  |
| --- |

(Slide 27 - *Numeric Summaries*)  
Numeric summaries – mean & SD vs. 5-number summary – can be important for discussing the distribution of each variable, if appropriate.

To calculate all of the numeric summaries by month:  
  
by(air\_quality\_data$Ozone, air\_quality\_data$Month, summary)

Write the output in the table below. Also, highlight the row (month) that has the greatest potential error. (Hint: what do the NAs in the data mean? Look at your data in a way that shows you NAs to learn more about this.)

| **Month** | **Min.** | **1st Qu.** | **Median** | **Mean** | **3rd Qu.** | **Max.** | **NA’s** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

(Slide 28 - *Removing NAs*)  
At times, calls/functions might not work properly with your dataset as it includes NA’s. To clean or remove NA’s from datasets:

nas <- is.na(air\_quality\_data) # Returns a logical array of TRUEs where NAs are in the original set. New array is same size as data  
  
air\_quality\_data\_cleaned <- air\_quality\_data[complete.cases(air\_quality\_data), ] # Note: we need the comma and space in the brackets to select all rows and columns to check  
  
Check this cleaned up dataset – what is the number of the last row? Are there that many rows?

|  |
| --- |

(Slide 29)  
Tidyverse package and examples.  
  
To install and load:  
  
install.packages("tidyverse")  
install.packages("ggplot2")  
**library**(tidyverse)  
**library**(ggplot2)  
  
Go through each example, paste your plot, and in one sentence describe the possible purpose of designing such a plot.

1. Histograms
   1. (Slide 30) Ex. 1:  
      qplot(air\_quality\_data$Ozone,  
       geom = "histogram", # graph type  
       binwidth = 3, # bin size  
       main = "Histogram of Ozone", # figure title  
       xlab = "ppb", # label for x axis  
       fill = I("green"), # fill color  
       col = I("black"), # outline color  
       alpha = I(.5), # fill transparency (between 0-see through and 1-opaque)  
       xlim = c(0, 200)) # x axis range

|  |
| --- |

* 1. (Slide 31) Ex. 2:  
     ggplot(data = air\_quality\_data, aes(air\_quality\_data$Ozone)) +  
      geom\_histogram(breaks = seq(0, 200, by = 3),  
      col = "black",  
      aes(fill = ..count..)) +  
      scale\_fill\_gradient("Count", low = "green", high = "red") +  
      labs(title = "Histogram of Ozone", x = "ppb", y = "Count")

|  |
| --- |

* 1. (Slide 32) Ex. 3:  
     multi <- air\_quality\_data %>% #tidyverse's "piping" functionality with "%>%"  
      ggplot(aes(x = Ozone, color = Month, fill = Month)) +  
      geom\_histogram(alpha = 0.5, binwidth = 3) +  
      theme(legend.position = "none") +  
      xlab("") +  
      ylab("ppb") +  
      facet\_wrap(~Month)  
     multi

|  |
| --- |

1. Boxplots
   1. (Slide 33) Ex. 1:  
      outliboxplot <- air\_quality\_data %>%  
       ggplot(aes(x = Month, y = Ozone, fill = Month)) +  
       geom\_boxplot(outlier.color = "black", outlier.shape = 8, outlier.size = 2) + # outlier boxplot  
       stat\_summary(fun = mean, geom = "point", shape = 23, size = 4) # add mean diamond  
      outliboxplot

|  |
| --- |

* 1. (Slide 34) Ex. 2:  
     dotnboxplot <- air\_quality\_data %>%  
      ggplot(aes(x = Month, y = Ozone)) +   
      geom\_boxplot() +   
      geom\_jitter(color = "black", size = 0.9, alpha = 0.8) +  
      theme(panel.border = element\_blank(),  
      panel.background = element\_blank(),  
      panel.grid.major = element\_blank(),  
      panel.grid.minor = element\_blank(),  
      axis.line = element\_line(color = "black"))  
     dotnboxplot

|  |
| --- |

1. Numeric Summaries  
     
   ? summarise  
     
   The following calls/functions will not perform correctly if there are NA’s in your dataset. Please remove them beforehand:  
     
   air\_quality\_data\_cleaned <- air\_quality\_data[complete.cases(air\_quality\_data), ]
   1. (Slide 35) Ex. 1:  
      air\_quality\_data\_cleaned %>%  
       group\_by(Month) %>%  
       summarise(count = n(),  
       df = count-1,  
       mean = mean(Ozone),  
       sd = sd(Ozone))

|  |
| --- |

* 1. (Slide 36) Ex. 2:  
     air\_quality\_data\_cleaned %>%  
      group\_by(Month) %>%  
      summarise(count = n(),  
      df = count-1,  
      min = min(Ozone),  
      Q1 = quantile(Ozone, 0.25),  
      M = median(Ozone),  
      Q3 = quantile(Ozone, 0.75),  
      max = max(Ozone))

|  |
| --- |

(Slide 37 - *Final Plot: Your Choice!*)  
Finally, see if you can create a figure (possibly with multiple panels) comparing the distributions, this time, of Wind or Temperature. The exact nature of the figure is up to you; if you have multiple panels, just be sure to label the separate panels. You should also be sure to include descriptions of the distribution (i.e., shape, spread, outliers) of each month or note any observations of the data that you had.

|  |
| --- |

Ask your TA for help if needed.

**R RESOURCES**

* [R Resources maintained by UCLA](https://stats.oarc.ucla.edu/r/)
* [R Bootcamp](https://www.jaredknowles.com/r-bootcamp/)
* [Quick-R](https://www.statmethods.net/) (*reference guide*)
* [RStudio Cheatsheets](https://www.rstudio.com/resources/cheatsheets/)
* [R Graph Gallery](https://www.r-graph-gallery.com/index.html) *(Tidyverse and ggplot2 help*)
* [GGPlot Color Tricks](https://www.datanovia.com/en/blog/ggplot-colors-best-tricks-you-will-love/) (*for color-blind-friendly graph-making*)
* [Stack Overflow](https://stackoverflow.com/questions/tagged/r) (*public forum for script-writing questions*)
* [R for Data Science](https://r4ds.had.co.nz/) (*by Hadley Wickham*)
* [Kickstarter guide](https://paulvanderlaken.com/2017/10/18/learn-r/) (*for new R users*)
* [Curated list of packages and tools](https://github.com/qinwf/awesome-R/blob/master/README.md) (*on Github*)