Basic R usage

- R is command-line driven. It requires you to type or copy-and-paste commands after a command prompt (>) that appears when you open R. After typing a command in the R console and pressing **Enter** on your keyboard, the command will run. If your command is not complete, R issues a continuation prompt (signified by a plus sign: +). Alternatively you can write a script in the script window, and select a command, and click the **Run** button.

- R is case sensitive. Make sure your spelling and capitalization are correct.

- Commands in R are also called functions. The basic format of a function in R is:

```
object <- function.name(argument_1 = data, argument_2 = TRUE)
```

- The **up arrow** on your keyboard can be used to recall previous commands that you've typed in the R console. This often works even across different sessions.

- The $ symbol is used to select a particular column within the table (e.g., `table$column`).

- Any text that you do not want R to act on (such as comments, notes, or instructions) needs to be preceded by the # symbol. R ignores the remainder of the line following #. For example:

```
plot(x, y) # This text will not affect the plot function
# Neither will this text
```

Set a working directory (not required, but useful)

```
setwd("/Users/tpd0001/Desktop")


# what is your working directory?
getwd()
```

Writing to output

```
"Hello world"
```
or
```
print("I said, HELLO WORLD!")
```

Basic arithmetic

```
1+1
log10(100)
log2(8)
```

Revisiting 0.1 + 0.2

```
0.1+0.2
print(0.1+0.2, digits=18)
```

Assigning (storing) to a variable

```
x = 84
```

or

```
x <- 42
```

Working with data (basic)

```
# import comma-separated data and store it as "genes"
genes <- read.csv("~/Desktop/genes.r_demo.txt")
# more explicitly
genes <- read.csv("~/Desktop/genes.r_demo.txt",header=TRUE,sep=",")

# export comma-separated data
write.csv(genes, "~/Desktop/genes.r_demo.OUTPUT.txt")

# view the structure of your data
str(genes)

# view a summary of your data
summary(genes)

# view all of your data
print(genes)

# view the first 6 lines of your data
head(genes)
head(genes, n=6)

# view the column names of your data
names(genes)

# remove your data object
rm(genes)
```

Visualizing data (basic)

```
# view the values for liver genes as a barplot
barplot(genes$liver)

# view liver values as a histogram
hist(genes$liver)

# view liver values as a scatter plot
plot(genes$liver)

# plot liver vs brain
```

```
plot(genes$liver, genes$brain)

# boxplot counts for each tissue
boxplot(genes[0:4])

# plot a correlation matrix of all columns in our data
plot(genes)
```

Working with packages

```
# what packages are installed on my machine?
library()

# load an installed package
library("ggplot2")

# get help/documentation for a package
help(package = "ggplot2")

# installing ballgown specifically
if (!requireNamespace("BiocManager", quietly=TRUE))
    install.packages("BiocManager")
BiocManager::install("ballgown")
```