QBIO490: Multi-omic Data Analysis
Intro to Clinical Data: Part II
Due: Friday 3/3 @ 11:59PM

Deliverables:

1. Submit your code by creating an R script (not a Notebook!) called `Intro_to_MAF_II_yourname.R` and adding it to your GitHub.
    a. Because this is a script, you will need to charge your working directory via `setwd("/PATH/TO/analysis_data")`
2. Include any graphs or figures created in this script in the folder with your R script with descriptive file names.

Since this is an optional partner activity, it is okay if your answers are the same as your partner's as long as everyone understands it and could explain it in their own words if asked. Each person must individually push their code to Github. At the top of your R Script, write the name of you and your partner(s) as a comment.

Before you start:

1. Read in the clinical data csv you saved in Part I.
2. Initialize your maf_object with the clinical annotations.
    - (You should not need to redownload the MAF TCGA files if you completed the TCGA Download Activity so the download function is commented out.)

```
maf_query <- GDCquery(
    project = "TCGA-BRCA",
    data.category = "Simple Nucleotide Variation",
    access = "open",
    data.type = "Masked Somatic Mutation",
    workflow.type = "Aliquot Ensemble Somatic Variant Merging
and Masking")

#GDCdownload(maf_query)

maf <- GDCprepare(maf_query)

maf_object <- read.maf(maf = maf,
                       clinicalData = clinical,
                       isTCGA = TRUE)
```

Complete the following coding activity and answer the following questions as comments in your R script:

In Intro to MAF: Part I, you looked at how to analyze mutation data in terms of clinical variables as well as mutation status. In this assignment, you will need to combine the two skills to demonstrate your understanding of categorical variables and R data structures.

1. Choose a clinical variable (or from clin_rad or clin_drug) to separate your populations into two different groups and rewrite the column or create a new column with that variable as a factor. Do not use age or vital_status as your clinical variable. *Hint: if your variable is continuous, you will need to determine your own cutoffs for the different levels of the factor. If your variable is categorical and has more than two possible values, choose the two that are the most common.

2. Create a co-oncoplot with the top 10-20 (you choose) most mutated genes for the two groups. Pick one that has a large discrepancy in % mutated or type of mutations between the groups and research it.
    - Research it. What is the gene used for? Can you think of any reason for the discrepancy?

3. Create a contingency table with your variable and chosen gene. Run a Fisher's Exact Test between presence of mutations for that gene and your clinical variable. Create and save a mosaic plot.
    - Interpret the output of the Fisher's Exact Test in terms of the odds ratio and p-value.

4. Subset your maf_object based on your chosen clinical variable and create a co-lollipop plot of your chosen gene divided between the two different clinical variable possibilities. Include descriptive names on your plot.
    - Do you notice any difference in terms of mutations (e.g. sites, types, number) between the two populations?

5. Create your Overall_Survival_Status column and create a mafSurvival KM plot based on mutations in your chosen gene.
    - Does there seem to be a difference? Hypothesize why or not based on the other analysis you did with the gene above.

NOTE: This assignment may be more difficult than previous assignments and requires a beginner-intermediate understanding of how to program in R. Please start it early and come to office hours/post on Piazza if you encounter any problems/bugs/questions. The TAs and Instructors are here to help with any troubles :)

Check before submitting:

You **must** include informative comments throughout your code.

```
str(clinical) # view structure of clinical data frame
head(clinical) # view first few rows of clinical data frame
```

You **must** install and load all necessary packages at the top of your coding fall.

```
if (!require(package)){
install.packages("package")
}

library(package)
```

You **must** change your working directory at the top of your coding file.

```
setwd("/Users/nicoleblack/Desktop/QBIO/qbio_nicole/analysis_data")
```

You **must** be able to run your script from top to bottom (with a clean environment) without any issues.

- Before turning it in, hit the broom in the top right corner of Environment to clear all values and data. Then run the entire script by hitting the run button in the top right of your source panel. Your code should run all the way through with no errors.