Assignment 3

You are asked to submit both the R Markdown file and its pdf output.

Q1. Write an if-else statement:

1. If the number is greater than 0 and less than 10, print: "This number is between 0 and 10"
2. If the number is greater than 10, print: "This number is greater than 10"
3. If the number is less than 0, print: "This number is a negative number"
4. Otherwise print: "This number is either 0 or 10"

Q2. Write a function that gets a vector as its input and returns the mean and the standard deviation of the vector using the formula below:

$$\left(\frac{\sum(x_i - \bar{x})^2}{n-1}\right)^{1/2}$$

where $\bar{x}$ is the mean and n is the length of the vector.

Q3.

(a) Write a function that returns Euclidian Distance between two k-dimensional vectors:

$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_k - y_k)^2}$$

(b) Write a function that will input the vectors x, y and p and will return the distance between two k-dimensional vectors:

$$d_p(x, y) = ((x_1 - y_1)^p + (x_2 - y_2)^p + ... + (x_k - y_k)^p)^{1/p}$$

Pick the default value for p as 2.

Q4. Create a function altman_plot that takes two arguments, `x` and `y`, and plots the difference against the sum.

Q5. Write a function `compute_s_n_2` that for any given n, computes the sum:

$$1^2 + 2^2 + 3^2 + 4^2 + ... + (n-1)^2 + n^2$$

(a) Find `compute_s_n_2(30)`. (b) Confirm that the formula for this sum is $= \frac{n(n+1)(2n+1)}{6}$.

Q6. Which of the following built-in datasets is tidy (you can pick more than one):

(a) BJsales
(b) EuStockMarkets
(c) DNase
(d) Formaldehyde
(e) Orange
(f) UCBAdmissions

Q7. Load the `dplyr` package and the `murders` dataset.

```
#library(dplyr)
#library(dslabs)
#data(murders)
```

(a) By using dplyr's `mutate` function, add a new column:

```
population_in_millions = population / 10^6
```

(b) If `rank(x)` gives you the ranks of `x` from lowest to highest, `rank(-x)` gives you the ranks from highest to lowest. Use the function mutate to add a column rank containing the rank, from highest to lowest murder rate. Make sure you redefine murders so we can keep using this variable.

(c) Select the columns `state`, `population` and give it a name `new_df`.

(d) Filter the observations with only `state== 'New York'` and call it as `murders_ny`.

(e) Remove all the observations with `state=='Florida'` and call that dataframe as `murders_no_fl`.

(f) Filter the murders dataset using `%in%` to filter the observations with `state=='New York'` or `state=='Texas'`.

(g) Suppose you want to live in the Northeast or West and want the murder rate to be less than 1. How many options do you have?

Q8. Use a pipe to create a new data frame called my_states that considers only states in the Northeast or West which have a murder rate lower than 1, and contains only the state, rate and rank columns.

```
#my_states <- murders %>%
 # mutate SOMETHING %>%
  #filter SOMETHING %>%
  #select SOMETHING
```

Q9. Install the NHANES package, load the data `NHANES`.

```
#library(NHANES)
#data(NHANES)
```

Observe that `NHANES` data has many missing values.

(a) Find the mean and the standard deviation of the variable `Age`. Remember the exclude the missing values. Hint: Add `na.rm = TRUE` inside the mean and the sd functions.

(b) First select the group as 20-to-29-year-old females. `AgeDecade` is a categorical variable with these ages. Note that the category is coded like " 20-29", with a space in front! What is the average and standard deviation of systolic blood pressure as saved in the `BPSysAve` variable? Save it to a variable called `ref`.

Hint: Use `filter` and `summarize` and use the `na.rm = TRUE` argument when computing the average and standard deviation. You can also `filter` the NA values using `filter`.

(c) Using a pipe, assign the average to a numeric variable `ref_avg`. Hint: Use the code similar to above and then `pull`.

(d) Compute the average and standard deviation for females, but for each age group separately rather than a selected decade as in the earlier question. Note that the age groups are defined by AgeDecade. Hint: rather than filtering by age and gender, `filter` by *Gender* and then use `group_by`.

(e) Repeat exercise (d) for males.

(f) We can actually combine both summaries for exercises 4 and 5 into one line of code. This is because `group_by` permits us to group by more than one variable. Obtain one big summary table using `group_by(AgeDecade, Gender)`.

(g) For males between the ages of 40-49, compare systolic blood pressure `BPSysAve` across race as reported in the Race1 variable. Order the resulting table from lowest to highest average systolic blood pressure.