Stat 5810: Introduction to R	Fall 2020

Name: Niranjan Poudel

Homework Assignment 04 (Befor class starts)

70 Points — Due befor week 3

General Instructions

For this fourth homework assignment, you have to create your own RMarkdown (.Rmd) file, based on files from class and from Homework 1, copy the question numbers and the answer options into your .Rmd file, and knit that file into a pdf file. **Alternatively** (and much easier!!!), use this .Rnw file as a template, just fill in the answers into the provided spaces, and knit into a pdf file.

Only the final resulting pdf file (from .Rmd or .Rnw) has to be submitted via Canvas. As previously stated, I would like to encourage potential and current MS and PhD students to work with .Rnw and LaTeX instead of .Rmd.

You need to learn how to write R code that is easily readable for others. There exists *Google's R Style Guide* that summarizes rules for good R style. These rules are accessible at `https://google.github.io/styleguide/Rguide.xml`. In particular, make sure that you always have a space after a comma and that you consistently use the same type of assignment operator, ideally `<-`. Look at the examples on this web page and follow the style whenever you write your own R code from now on.

Do not forget to replace my name and include your name instead! We will print the homeworks, so a homework with no name/my name on it can't be graded!

In all question parts, show your R code and the results!

(i) (20 Points) **Family Data Revisited:**
In the following exercises, try to write your code to be as general as possible so that it would still work if the family had 27 members in it or if the variables were in a different order in the data frame.

Show your R code and the final results produced from within R for all question parts!

(a) (3 Points) Copy the family data set for this homework from Canvas into your local folder for this homework. Then load the `hw04_familyDF.rda` data set into R. Show the objects that have been loaded. Is the first object that is listed a data frame? Search for help if you don't recall how to check whether something is a data frame.

Answer:

```
> load(file = "hw04_familyDF.rda")
> print(load(file = "hw04_familyDF.rda"))

[1] "family"

> head(family)  # To show object is loaded

  firstName gender age height weight      bmi overWt
1       Tom      m  77     70    175 25.16239   TRUE
2       May      f  33     64    125 21.50106  FALSE
3       Joe      m  79     73    185 24.45884  FALSE
4       Bob      m  47     67    156 24.48414  FALSE
5       Sue      f  27     64    105 18.06089  FALSE
6       Liz      f  33     68    190 28.94981   TRUE

> options(width = 80)  # For pdf display
> is.data.frame(family)

[1] TRUE
```

(b) (4 Points) The NHANES survey used different cut-off values for men and women when classifying them as overweight. Suppose that a man is classified as obese if his bmi exceeds 26 and a woman is classified as obese if her bmi exceeds 25. Write a logical expression to create a logical vector, called OW.NHANES, that is TRUE if a member of the family is obese and FALSE otherwise. Display its content.

Answer:

```
> OW.NHANES <- as.logical((family$gender == 'm' & family$bmi > 26)
+                         | (family$gender == 'f' & family$bmi > 25))
> OW.NHANES

 [1] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
[13] FALSE FALSE
```

(c) (4 Points) Here is an alternative way to create the same vector that introduces some useful functions and ideas. We first create a numeric vector called OW.limit that is 26 for each male in the family and 25 for each female in the family. To do this, we create a vector of length 2, called OW.val, where the first element is 26 and second element is 25. Then we create the OW.limit vector by subsetting OW.val by position, where the positions are the numeric values in the gender variable (i.e., use as.numeric to coerce the factor vector to a numeric vector). Notice that we can "subset" a vector of length 2 by a much longer vector:

```
> OW.val <- 26:25
> OW.limit <- OW.val[as.numeric(family$gender)]
> OW.limit
```

Finally, use OW.limit and the bmi vector in family to create the desired logical vector, and call it OW.NHANES2. Display its content. Compare with your results from part (b) via the `any` function. Did you get the intended result? If not, check your R code again!

Answer:

```
> OW.val <- 26:25
> OW.limit <- OW.val[as.numeric(family$gender)]
> OW.limit

 [1] 26 25 26 26 25 25 26 25 26 26 25 26 26 25

> OW.NHANES2 <- as.logical(family$bmi > OW.limit)
> OW.NHANES2

 [1] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
[13] FALSE FALSE
```

(d) (4 Points) Use the vector OW.limit and each person's height to find the weight that they would have if their bmi was right at the limit (26 for men and 25 for women). Call this weight OW.weight and display its content. To do this, start with the formula

```
bmi = (weight / 2.2) / (2.54 / 100 * height)^2
```

and re-express it in terms of weight (i.e., `weight = ...`).

Answer:

```
> OW.weight <- OW.limit * (2.54 / 100 * family$height) ^ 2 * 2.2
> OW.weight

 [1] 180.8254 145.3416 196.6569 165.6582 145.3416 164.0771 170.6402 149.9191
 [9] 170.6402 186.0288 159.2868 160.7501 160.7501 136.3997
```

(e) (5 Points) Create the following plot of actual weight (on the vertical axis) against the weight at which they would be overweight (on the horizontal axis). If you get an error when you run this code, check whether you are using the correct variable names in your code earlier on.

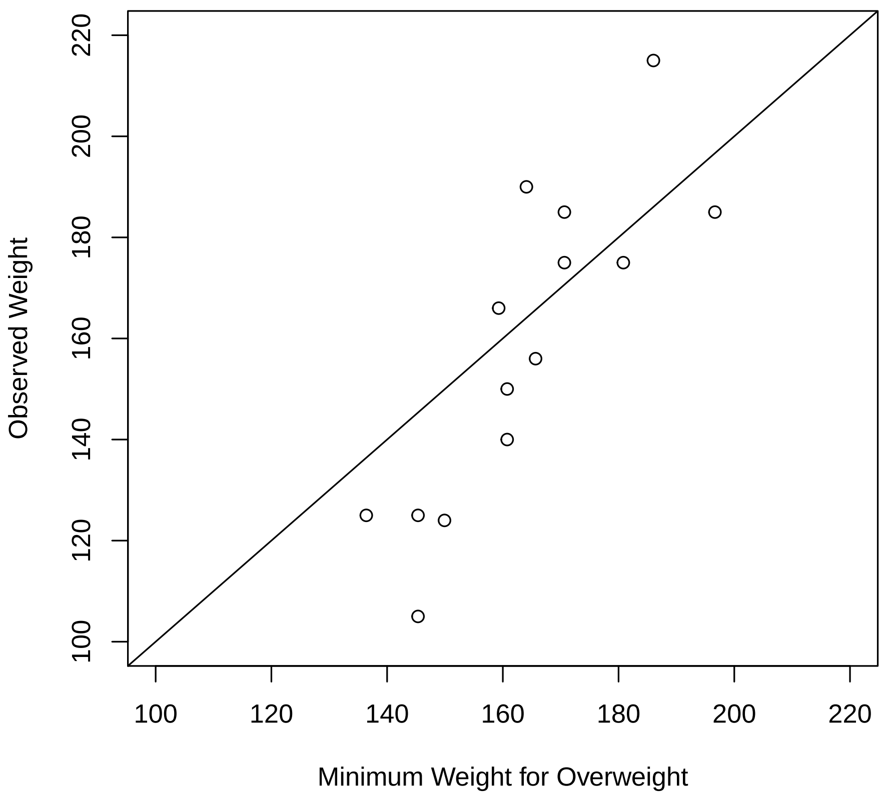
```
> plot(OW.weight, family$weight,
+       xlab = "Minimum Weight for Overweight",
+       xlim = c(100, 220), # !!!
+       ylab = "Observed Weight",
+       ylim = c(100, 220)) # !!!
> abline(a = 0, b = 1)
```

`abline` adds a straight line (here with y-intercept $a = 0$ and slope $b = 1$) to the plot. Note that this is not the regression line! Thus, points that fall exactly on the line belong to individuals where the observed weight exactly qualifies to be overweight. Points above the line represent individuals who are overweight, and points below the line represent individuals who are not overweight.

We can easily count in the plot how many points are above the line and how many points are below the line, but we want that R does this counting for us! So, write two R expressions that do this counting for us and display their results.

Answer:

```
> plot(OW.weight, family$weight,
+      xlab = "Minimum Weight for Overweight",
+      xlim = c(100, 220), # !!!
+      ylab = "Observed Weight",
+      ylim = c(100, 220)) # !!!
> abline(a = 0, b = 1)
```




```
> # Number of points above the line
> sum(OW.weight < family$weight)
[1] 5
> # Number of points below the line
> sum(OW.weight > family$weight)
[1] 9
```

(ii) (34 Points) **San Francisco Housing Data:**

In this question, you have to work with actual housing data from the San Francisco area.

Show your R code and the final results produced from within R for all question parts!

(a) (4 Points) Copy the San Francisco housing data set (`hw04_SFhousing.rda`) for this homework from Canvas into your local folder for this homework. Then load this data set into R. Show the objects that have been loaded. Are cities and housing both data frames? Let R answer this question! Search for help if you don't recall how to check whether something is a data frame.

Answer:

```
> load(file = "hw04_SFHousing.rda")
> print(load(file = "hw04_SFHousing.rda"))

[1] "cities"  "housing"

> head(cities)  # To show objects are loaded

                longitude latitude              county medianPrice medianSize
Alameda         -122.2485 37.75993      Alameda County      580000     1489.0
Alamo           -122.0205 37.85522 Contra Costa County     1250000     2723.5
Albany          -122.2940 37.89107      Alameda County      520250     1170.0
Almaden                NA       NA  Santa Clara County      835000     2139.0
American Canyon -122.2580 38.16664         Napa County      419000     1344.0
Angwin          -122.4499 38.57451         Napa County      662000     1822.0
                numHouses medianBR
Alameda              2339        3
Alamo                 760        4
Albany                640        2
Almaden              1705        4
American Canyon       463        3
Angwin                 79        3

> head(family)  # To show objects are loaded

  firstName gender age height weight      bmi overWt
1       Tom      m  77     70    175 25.16239   TRUE
2       May      f  33     64    125 21.50106  FALSE
3       Joe      m  79     73    185 24.45884  FALSE
4       Bob      m  47     67    156 24.48414  FALSE
5       Sue      f  27     64    105 18.06089  FALSE
6       Liz      f  33     68    190 28.94981   TRUE

> is.data.frame(cities)

[1] TRUE

> is.data.frame(housing)

[1] TRUE
```

(b) (2 Points) What are the names of the vectors in housing?

Answer:

```
> names(housing)

 [1] "county"  "city"    "zip"     "street"  "price"   "br"      "lsqft"
 [8] "bsqft"   "year"    "date"    "long"    "lat"     "quality" "match"
[15] "wk"
```

(c) (2 Points) How many observations are in housing?

Answer:

```
> nrow(housing)
[1] 281506
```

(d) (6 Points) Explore the housing data using the summary function. Describe in words at least three problems that you see with the data.

Answer:

```
> summary(housing)

                county                   city              zip
 Santa Clara County :70424   Oakland      : 14730   94565  :  4595
 Alameda County     :60410   Santa Rosa   :  9917   94509  :  4302
 Contra Costa County:59381   Fremont      :  9414   95123  :  4023
 Solano County      :23404   San Francisco:  8137   95687  :  3652
 San Mateo County   :22558   Evergreen    :  7947   94533  :  3472
 Sonoma County      :21676   Antioch      :  7726   (Other):261457
 (Other)            :23653   (Other)      :223635   NA's   :     5
    street              price                br             lsqft
 Length:281506      Min.   :   22000   Min.   :1.000   Min.   :       19
 Class :character   1st Qu.:  400000   1st Qu.:2.000   1st Qu.:     4000
 Mode  :character   Median :  530000   Median :3.000   Median :     5760
                    Mean   :  602000   Mean   :3.024   Mean   :    65939
                    3rd Qu.:  700000   3rd Qu.:4.000   3rd Qu.:     7701
                    Max.   :20000000   Max.   :8.000   Max.   :418611600
                                                       NA's   :21687
     bsqft              year            date
 Min.   :    122   Min.   :   0   Min.   :2003-04-27 01:00:00
 1st Qu.:   1121   1st Qu.:1954   1st Qu.:2004-02-08 01:00:00
 Median :   1430   Median :1971   Median :2004-10-24 01:00:00
 Mean   :   1624   Mean   :1966   Mean   :2004-11-01 17:06:12
 3rd Qu.:   1882   3rd Qu.:1985   3rd Qu.:2005-07-24 01:00:00
 Max.   :1868120   Max.   :3894   Max.   :2006-06-04 01:00:00
 NA's   :426       NA's   :9202
      long             lat
 Min.   :-123.6   Min.   :36.98
 1st Qu.:-122.3   1st Qu.:37.50
 Median :-122.1   Median :37.77
 Mean   :-122.1   Mean   :37.78
 3rd Qu.:-121.9   3rd Qu.:38.00
 Max.   :-121.5   Max.   :38.85
 NA's   :23316    NA's   :23316
                                     quality                    match
 QUALITY_ADDRESS_RANGE_INTERPOLATION      :170719   Exact           :197044
 gpsvisualizer                            : 31084   Relaxed         : 30570
 QUALITY_CITY_CENTROID                    : 20473   Relaxed; Soundex: 23338
 QUALITY_EXACT_PARCEL_CENTROID            : 17208   Soundex         :  2573
 QUALITY_ZIP_CODE_TABULATION_AREA_CENTROID: 14980   1               :  2244
 (Other)                                  :  3726   (Other)         :  2421
 NA's                                     : 23316   NA's            : 23316
       wk
 Min.   :2003-04-21
 1st Qu.:2004-02-01
 Median :2004-10-18
 Mean   :2004-10-26
 3rd Qu.:2005-07-18
 Max.   :2006-05-29
```

Problems:

i. Problem 1.High number of NA's present in some columns.

ii. Problem 2.The year column has years greater than 2020.

iii. Problem 3.The column lsqft seems to have a very large area (Max area)

(e) (4 Points) We will work with houses in Albany, Berkeley, Piedmont, and Emeryville only. Subset the data frame so that we have only houses in these cities, and keep only the variables city, zip, price, br, bsqft, and year. Call this new data frame BerkArea. This data frame should have 4059 observations and 6 variables (check it!).

Answer:

```
> BerkArea <- subset(housing, housing$city %in%
+                  c('Albany', 'Berkeley', 'Piedmont', 'Emeryville'),
+                  select = c('city', 'zip', 'price', 'br', 'bsqft',
+                                 'year'))
> dim(BerkArea)
```

```
[1] 4059    6
```

(f) (4 Points) We are interested in studying the relationship between price and size of house, but first we will further subset the data frame to remove the unusually large values. Use the quantile function to determine the 99th percentile of price and bsqft and eliminate all of those houses that are above either of these 99th percentiles. Call this new data frame BerkArea, as well. It should have 3999 observations (check it!). Write your code so that it is very general and does not depend on the actual numeric value for these quantiles.

Answer:

```
> BerkArea <- subset(BerkArea, (
+             (BerkArea$price <= quantile(BerkArea$price, 0.99))
+             & (BerkArea$bsqft <= quantile(BerkArea$bsqft, 0.99,
+             na.rm = T) | is.na(BerkArea$bsqft))))
> nrow(BerkArea)
```

```
[1] 3999
```

(g) (2 Points) Create a new vector that is called pricepsqft by dividing the sale price by the square footage of the house. Add this new variable to the BerkArea housing data frame.

Answer:

```
> BerkArea$pricepsqft <- BerkArea$price / BerkArea$bsqft
```

(h) (4 Points) Create a vector called br5 that contains the number of bedrooms in the house, except when this number is greater than 5, it is set to 5. That is, if a house has 5 or more bedrooms then br5 will be 5. Otherwise it will be the number of bedrooms in the house. Note that there is no need for any "if"-statements or loops to create this vector — just basic R expressions discussed so far will be sufficient! Recall how TRUE and FALSE are represented numerically or how to reassign a different value to a subset!

Answer:

```
> br5 <- BerkArea$br  # Makes it easy for to code as below
> br5[br5 >= 5] <- 5  # Felt more easy
```

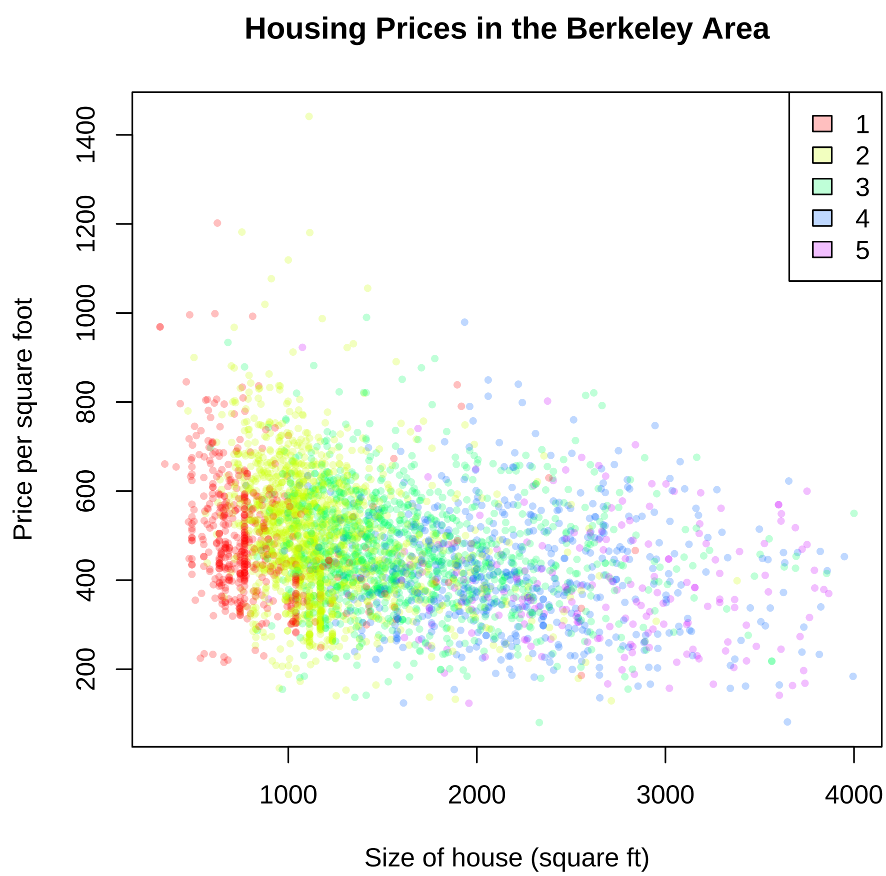
(i) (6 Points) Recreate the following plot on your side. Then answer the question below. If you get an error when you run this code, check whether you are using the correct variable names in your code earlier on.

```
> rCols <- rainbow(5, alpha = 0.25)
> brCols <- rCols[br5]
> plot(pricepsqft ~ bsqft, data = BerkArea,
+       main = "Housing Prices in the Berkeley Area",
+       xlab = "Size of house (square ft)",
+       ylab = "Price per square foot",
+       col = brCols, pch = 19, cex = 0.5)
> legend(legend = 1:5, fill = rCols, "topright")
```

What interesting feature do you see that you didn't know before making this plot? Numerically quantify (use only 3 decimal digits!) and interpret this feature!

Answer:

```
> rCols <- rainbow(5, alpha = 0.25)
> brCols <- rCols[br5]
> plot(pricepsqft ~ bsqft, data = BerkArea,
+       main = "Housing Prices in the Berkeley Area",
+       xlab = "Size of house (square ft)",
+       ylab = "Price per square foot",
+       col = brCols, pch = 19, cex = 0.5)
> legend(legend = 1:5, fill = rCols, "topright")
```



Before making this plot I was unaware of the housing prices based on the area of the houses (number of bedroom).Obviously as expected with large area there are more number of bedroom, i thought the houses with large area might be more expensive (Price per square feet) but it looks opposite as it seems like there is slight negative correlation (dipping downward trend). But obviously we have not accounted for other factors like the location (which seems to me as a major factor). **Frankly speaking i am not still sure in what context to explain numerically. (may be correlation or just the average price values per bedroom or per area)**. I will just calculate the correlation between pricesqft and bsqft.

```
> # I have used correlation (among many more possibility)
> cor(BerkArea$pricepsqft, BerkArea$bsqft, use = "complete.obs")

[1] -0.2929954
```

As expected negative correlation of 0.293 between the price per square feet and area of the house.

(iii) (16 Points) **Survival of Passengers on the Titanic:**
Work with the `Titanic` data set, a 4-dimensional array related to the survival of passengers and crew on board of the Titanic ocean liner. For further details, refer to the help page via `?Titanic`. Technically, the Titanic data set is a table, but we can access it similar to a multi-dimensional array.

Show your R code and the final results produced from within R for all question parts!

(a) (4 Points) Write an R expression that extracts the numbers of females in all three classes (but not crew) who survived the sinking of the Titanic. Provide data for children and adults. The result should look as follows:

```
      Age
Class Child Adult
  1st     1   140
  2nd    13    80
  3rd    14    76
```

Answer:

```
> # The Titanic data is pre-loaded in the packages(datasets)
> Titanic[, "Female", , "Yes"][c("1st", "2nd", "3rd"), ]

     Age
Class Child Adult
  1st     1   140
  2nd    13    80
  3rd    14    76
```

(b) (4 Points) Write an R expression that extracts the numbers of male crew members (adults only) who survived or did not survive the sinking of the Titanic. The result should be a vector of length 2.

Answer:

```
> Titanic["Crew", "Male", "Adult", ]

 No Yes
670 192
```

(c) (4 Points) Write an R expression that extracts the following matrix from the Titanic data set:

```
      Sex
Class  Female Male
  Crew     20  192
  1st     140   57
  2nd      80   14
  3rd      76   75
```

Describe what this matrix represents, i.e., which subgroup(s) from the Titanic passengers and crew.

Answer:

```
> row <- c("Crew", "1st", "2nd", "3rd")
> col <- c("Female", "Male")
> Titanic[, , "Adult", "Yes"][row, col]

      Sex
Class  Female Male
  Crew     20  192
  1st     140   57
  2nd      80   14
  3rd      76   75
```

This matrix represents the number of adults (male and female) who survived in all class categories.

(d) (4 Points) Write an R expression that extracts the following vector from the Titanic data set:

```
[1]  35  17 387  89
```

Describe what this vector represents, i.e., which subgroup(s) from the Titanic passengers and crew. Hint: I first extracted a matrix and then transformed this into a vector using `as.vector`.

Answer:

```
> as.vector(Titanic["3rd", , , "No"])
[1]  35  17 387  89
```

This vector represents the number of people who did not survive (Both child and adult) for male and female separately, the first two values are male and female for child group who did not survive and the later two for adults in similar category.

The end !!!!