STAT 210
Applied Statistics and Data Analysis:
Homework 2 - Solution

Due on Sept. 18/2022

Question 1

You will need the file `Human_data.txt`. Place this file on your working environment.

(a) Read the file `Human_data.txt` and store this in an object called `human`. Before reading the data, check whether the file has a header. If it does, use the appropriate argument in the read function to include the header. Look at the structure of `human` using the function `str`.

```
human <- read.table('Human_data.txt', header = T)
str(human)
```

```
## 'data.frame':    500 obs. of  10 variables:
##  $ Index         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Gender        : chr  "M" "F" "M" "F" ...
##  $ age           : int  22 33 46 24 37 31 38 38 21 31 ...
##  $ Ocupation     : chr  "Nothing" "Nothing" "Work" "student" ...
##  $ Head_size     : num  34.4 28 27 24.8 30.1 26.6 25.6 25.6 27.6 23.6 ...
##  $ Height_cm     : num  206 163 162 156 173 ...
##  $ Weight_kg     : num  105.3 71.3 94.7 56 103.3 ...
##  $ Salary        : int  0 0 19268 2034 14829 10586 11272 13048 2068 12326 ...
##  $ blood_type    : int  4 4 4 3 2 3 4 2 1 3 ...
##  $ Sugar_in_blood: num  95.2 83.5 92.7 95.8 114.1 ...
```

We see that the file has 500 observations of 10 variables, two of which are character variables while the rest are numeric.

(b) The body mass index (BMI) is defined as a person's weight in kilograms divided by the square of height in meters. Add a column named `bmi` to the data frame with the value of this index for each subject. Count how many subjects have BMI above 30.

There are several ways of adding a variable `bmi` to the data frame `human`. One is to use the function `within` to create the new variable

```
human <- within(human, {bmi = Weight_kg/(Height_cm/100)^2})
str(human)
```

```
## 'data.frame':    500 obs. of  11 variables:
##  $ Index         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Gender        : chr  "M" "F" "M" "F" ...
##  $ age           : int  22 33 46 24 37 31 38 38 21 31 ...
##  $ Ocupation     : chr  "Nothing" "Nothing" "Work" "student" ...
##  $ Head_size     : num  34.4 28 27 24.8 30.1 26.6 25.6 25.6 27.6 23.6 ...
##  $ Height_cm     : num  206 163 162 156 173 ...
##  $ Weight_kg     : num  105.3 71.3 94.7 56 103.3 ...
##  $ Salary        : int  0 0 19268 2034 14829 10586 11272 13048 2068 12326 ...
```

```
##  $ blood_type    : int  4 4 4 3 2 3 4 2 1 3 ...
##  $ Sugar_in_blood: num  95.2 83.5 92.7 95.8 114.1 ...
##  $ bmi           : num  24.9 26.9 35.9 23 34.6 ...
```

Another is to define a new variable inside the file **human** using the $ notation:

```
human$bmi <- human$Weight_kg/(human$Height_cm/100)^2
```

A third way is to use the function **transform**

```
human <- transform(human, bmi = Weight_kg/(Height_cm/100)^2)
```

Now we count how many subjects have bmi greater than 30.

```
sum(human$bmi>30)
```

```
## [1] 108
```

There are 108 subjects out of 500 with bmi above 30.

(c) Calculate mean and standard deviation for **bmi** according to **Gender**. Compare these results and comment. Boxplot **bmi** against **Gender** and comment.

A convenient function for calculating mean and standard deviation is **tapply**

```
tapply(human$bmi, human$Gender, mean)
```

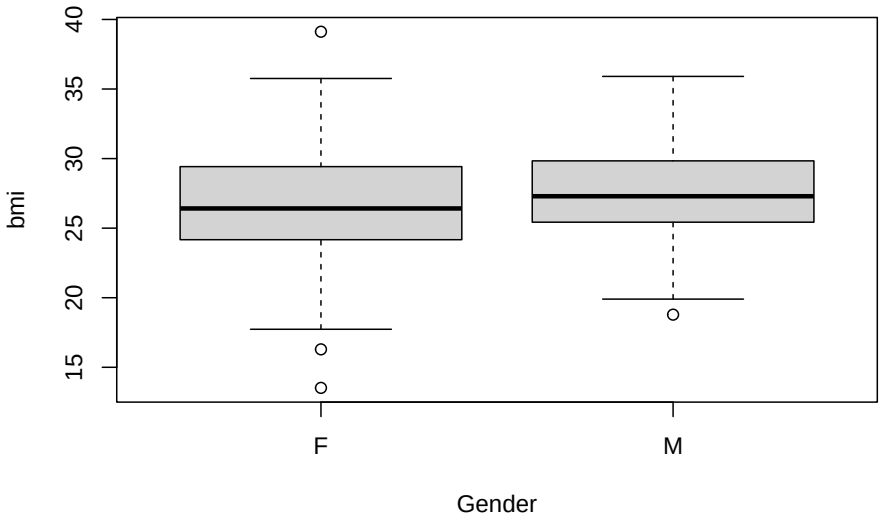
```
##        F        M
## 26.60612 27.68282
```

```
tapply(human$bmi, human$Gender, sd)
```

```
##        F        M
## 3.888583 3.253710
```

We see that females have a lower mean value for BMI but the standard deviation is bigger, so the distribution is more spread. This is also apparent form the boxplots:

```
boxplot(bmi ~ Gender, data = human)
```



The median for males is higher and the range of values and the box width are shorter than for females.

(d) Using **subset**, create a new data frame from **human** with the variables **Head_size, Height_cm, Weight_kg** for subjects with age between 30 and 50 (both inclusive) and head size bigger than 26. Call this new data frame **human1**.

```
human1 <- subset(human, age >= 30 & age <= 50 & Head_size > 26,
                 select = c(Head_size, Height_cm, Weight_kg))
str(human1)
```

```
## 'data.frame':    203 obs. of  3 variables:
##  $ Head_size: num  28 27 30.1 26.6 27.9 30.1 28.7 26.3 26.5 30.1 ...
##  $ Height_cm: num  163 162 173 158 165 ...
##  $ Weight_kg: num  71.3 94.7 103.3 47 98 ...
```

(e) Use the function `apply` twice to calculate the mean and standard deviation for each of the three variables in `human1`. Call the vectors you obtain `human.mean` and `human.sd`.

```
(human.mean <- apply(human1, 2, mean))
```

```
## Head_size Height_cm Weight_kg
##  29.06995 173.72069  83.62611
```

```
(human.sd <- apply(human1, 2, sd))
```

```
## Head_size Height_cm Weight_kg
##  1.844911 11.261780 15.266648
```

(f) Use the function `sweep` twice, first to subtract the mean for each variable to the values in `human1` and then to divide by the standard deviation. Store the result in a data frame named `human.std`.

```
human.cen <- sweep(human1, 2, human.mean)
human.std <- sweep(human.cen,2,human.sd,'/')
```

(g) The previous procedure is known as *standardization*. The resulting columns in the `human.std` should now have mean zero and variance equal to one. Verify this using `apply`.

```
apply(human.std, 2, mean)
```

```
##    Head_size    Height_cm    Weight_kg
## 6.606647e-16 6.125368e-16 3.787337e-16
```

Observe that the values we obtain are not exactly zero. This is due to rounding off errors in the calculations. To get a neater result, we can use the function round that rounds off the results to a given accuracy level, in our case 10^{-14}:

```
round(apply(human.std, 2, mean),14)
```

```
## Head_size Height_cm Weight_kg
##         0         0         0
```

```
apply(human.std, 2, sd)
```

```
## Head_size Height_cm Weight_kg
##         1         1         1
```

Question 2

For this question you will use again the file `human` that you created in the first question.

(a) Use the function `split` on the file `human` with second argument `Gender` and store the result in an object called `human2`. Describe this object.

```
human2 <- split(human,human$Gender)
str(human2)
```

```
## List of 2
##  $ F:'data.frame':   272 obs. of  11 variables:
##   ..$ Index         : int [1:272] 2 4 5 6 7 8 9 10 13 15 ...
##   ..$ Gender        : chr [1:272] "F" "F" "F" "F" ...
##   ..$ age           : int [1:272] 33 24 37 31 38 38 21 31 49 33 ...
##   ..$ Ocupation     : chr [1:272] "Nothing" "student" "Work" "Work" ...
##   ..$ Head_size     : num [1:272] 28 24.8 30.1 26.6 25.6 25.6 27.6 23.6 30.1 25.4 ...
##   ..$ Height_cm     : num [1:272] 163 156 173 158 152 ...
##   ..$ Weight_kg     : num [1:272] 71.3 56 103.3 47 46.5 ...
##   ..$ Salary        : int [1:272] 0 2034 14829 10586 11272 13048 2068 12326 0 17270 ...
##   ..$ blood_type    : int [1:272] 4 3 2 3 4 2 1 3 2 1 ...
##   ..$ Sugar_in_blood: num [1:272] 83.5 95.8 114.1 95.1 82.7 ...
##   ..$ bmi           : num [1:272] 26.9 23 34.6 18.9 20.1 ...
##  $ M:'data.frame':   228 obs. of  11 variables:
##   ..$ Index         : int [1:228] 1 3 11 12 14 16 17 18 19 20 ...
##   ..$ Gender        : chr [1:228] "M" "M" "M" "M" ...
##   ..$ age           : int [1:228] 22 46 19 38 57 35 29 52 54 41 ...
##   ..$ Ocupation     : chr [1:228] "Nothing" "Work" "student" "Work" ...
##   ..$ Head_size     : num [1:228] 34.4 27 32.3 27.9 27.7 28.7 30.8 29.5 28 26.3 ...
##   ..$ Height_cm     : num [1:228] 206 162 190 165 167 ...
##   ..$ Weight_kg     : num [1:228] 105.3 94.7 109.6 98 75.4 ...
##   ..$ Salary        : int [1:228] 0 19268 2493 10900 19709 14521 16198 10961 0 12904 ...
##   ..$ blood_type    : int [1:228] 4 4 2 4 1 4 2 1 3 4 ...
##   ..$ Sugar_in_blood: num [1:228] 95.2 92.7 95.7 106.2 96 ...
##   ..$ bmi           : num [1:228] 24.9 35.9 30.3 35.9 27.1 ...
```

The function `split` creates a list whose components are obtained by dividing the original file according to the value of a factor. In our case the factor is `Gender`, with two values. Therefore, the list has two components, one for `F` and one for `M`. Each component is a data frame with the same variables as the original file.

(b) Using the data in `human2` obtain a numerical summary (`summary`) for the variable `Salary` for males and females and compare.

```
summary(human2$F$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    2063   10660    8849   13364   44232
```

```
summary(human2$M$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    2152   11178    9836   14404   27354
```

We observe that all the values in the summary are higher for males, except for the maximum, which is much higher for females.

(c) Use again the function `split` on `human` but now you want to use two variables for splitting the data, `Gender` and `Work`. Look at the help for this function to find out how to do this. Call the resulting object `human3`. Describe the file `human3`.

```
human3 <- split(human,list(human$Ocupation, human$Gender))
str(human3)
```

```
## List of 6
##  $ Nothing.F:'data.frame':   61 obs. of  11 variables:
##   ..$ Index         : int [1:61] 2 13 22 26 27 32 33 38 44 61 ...
##   ..$ Gender        : chr [1:61] "F" "F" "F" "F" ...
##   ..$ age           : int [1:61] 33 49 20 24 21 26 24 38 26 31 ...
```

```
##   ..$ Ocupation     : chr [1:61] "Nothing" "Nothing" "Nothing" "Nothing" ...
##   ..$ Head_size     : num [1:61] 28 30.1 27.8 25.2 24.6 29.6 26.8 28.7 28.4 27.6 ...
##   ..$ Height_cm     : num [1:61] 163 178 160 156 156 ...
##   ..$ Weight_kg     : num [1:61] 71.3 109 83.9 64.6 58.7 96.9 74.8 91.1 88.3 84.9 ...
##   ..$ Salary        : int [1:61] 0 0 0 0 0 0 0 0 0 0 ...
##   ..$ blood_type    : int [1:61] 4 2 3 3 4 3 4 3 1 4 ...
##   ..$ Sugar_in_blood: num [1:61] 83.5 81.1 83 98.1 83.2 ...
##   ..$ bmi           : num [1:61] 26.9 34.3 32.9 26.4 24.2 ...
##  $ student.F:'data.frame':   47 obs. of  11 variables:
##   ..$ Index         : int [1:47] 4 9 47 55 62 103 122 126 129 141 ...
##   ..$ Gender        : chr [1:47] "F" "F" "F" "F" ...
##   ..$ age           : int [1:47] 24 21 23 22 21 20 22 20 22 24 ...
##   ..$ Ocupation     : chr [1:47] "student" "student" "student" "student" ...
##   ..$ Head_size     : num [1:47] 24.8 27.6 23.5 26.5 24.9 27.9 29.9 22.7 26.8 29.6 ...
##   ..$ Height_cm     : num [1:47] 156 164 145 159 155 ...
##   ..$ Weight_kg     : num [1:47] 56 82.2 82.7 75.3 51.7 75.1 92.7 40.1 64.7 73.5 ...
##   ..$ Salary        : int [1:47] 2034 2068 2271 2157 2415 2539 2345 2008 2064 2443 ...
##   ..$ blood_type    : int [1:47] 3 1 3 1 4 3 1 4 2 2 ...
##   ..$ Sugar_in_blood: num [1:47] 95.8 80.5 91.2 95.5 86.7 87.1 94.5 99 93.1 81.2 ...
##   ..$ bmi           : num [1:47] 23 30.7 39.1 29.9 21.6 ...
##  $ Work.F   :'data.frame':   164 obs. of  11 variables:
##   ..$ Index         : int [1:164] 5 6 7 8 10 15 21 24 31 36 ...
##   ..$ Gender        : chr [1:164] "F" "F" "F" "F" ...
##   ..$ age           : int [1:164] 37 31 38 38 31 33 49 42 44 50 ...
##   ..$ Ocupation     : chr [1:164] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:164] 30.1 26.6 25.6 25.6 23.6 25.4 26.5 23.7 24.6 26.6 ...
##   ..$ Height_cm     : num [1:164] 173 158 152 153 149 ...
##   ..$ Weight_kg     : num [1:164] 103.3 47 46.5 64.6 52.1 ...
##   ..$ Salary        : int [1:164] 14829 10586 11272 13048 12326 17270 11992 11996 12646 10412 ...
##   ..$ blood_type    : int [1:164] 2 3 4 2 3 1 1 4 3 2 ...
##   ..$ Sugar_in_blood: num [1:164] 114.1 95.1 82.7 112 100.6 ...
##   ..$ bmi           : num [1:164] 34.6 18.9 20.1 27.7 23.5 ...
##  $ Nothing.M:'data.frame':   49 obs. of  11 variables:
##   ..$ Index         : int [1:49] 1 19 23 34 42 43 46 52 60 106 ...
##   ..$ Gender        : chr [1:49] "M" "M" "M" "M" ...
##   ..$ age           : int [1:49] 22 54 37 20 45 36 34 52 59 45 ...
##   ..$ Ocupation     : chr [1:49] "Nothing" "Nothing" "Nothing" "Nothing" ...
##   ..$ Head_size     : num [1:49] 34.4 28 30.1 28.6 28.2 32.9 28.9 28.7 27.5 30.5 ...
##   ..$ Height_cm     : num [1:49] 206 173 182 177 163 ...
##   ..$ Weight_kg     : num [1:49] 105.3 69.1 92.4 74.9 56.6 ...
##   ..$ Salary        : int [1:49] 0 0 0 0 0 0 0 0 0 0 ...
##   ..$ blood_type    : int [1:49] 4 3 1 3 1 2 1 3 3 3 ...
##   ..$ Sugar_in_blood: num [1:49] 95.2 99.4 93.3 88 90.9 ...
##   ..$ bmi           : num [1:49] 24.9 23.2 28 23.8 21.3 ...
##  $ student.M:'data.frame':   28 obs. of  11 variables:
##   ..$ Index         : int [1:28] 11 63 94 108 110 111 124 128 155 163 ...
##   ..$ Gender        : chr [1:28] "M" "M" "M" "M" ...
##   ..$ age           : int [1:28] 19 18 24 24 21 20 19 22 22 21 ...
##   ..$ Ocupation     : chr [1:28] "student" "student" "student" "student" ...
##   ..$ Head_size     : num [1:28] 32.3 31.6 29.4 28 26 28.9 30.8 31.5 31.3 31.3 ...
##   ..$ Height_cm     : num [1:28] 190 190 174 174 165 ...
##   ..$ Weight_kg     : num [1:28] 109.6 110.2 81.5 80.4 89.3 ...
##   ..$ Salary        : int [1:28] 2493 2739 2028 2083 2063 2263 2427 3402 2152 2269 ...
##   ..$ blood_type    : int [1:28] 2 4 2 1 2 3 3 4 1 1 ...
```

```
##   ..$ Sugar_in_blood: num [1:28] 95.7 80.9 108.7 108.8 91.3 ...
##   ..$ bmi           : num [1:28] 30.3 30.6 27 26.7 32.8 ...
##  $ Work.M   :'data.frame':   151 obs. of  11 variables:
##   ..$ Index         : int [1:151] 3 12 14 16 17 18 20 25 28 29 ...
##   ..$ Gender        : chr [1:151] "M" "M" "M" "M" ...
##   ..$ age           : int [1:151] 46 38 57 35 29 52 41 45 55 50 ...
##   ..$ Ocupation     : chr [1:151] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:151] 27 27.9 27.7 28.7 30.8 29.5 26.3 30.2 28.1 30.4 ...
##   ..$ Height_cm     : num [1:151] 162 165 167 172 182 ...
##   ..$ Weight_kg     : num [1:151] 94.7 98 75.4 55.7 90.1 95.9 58.8 96.2 90.1 88.6 ...
##   ..$ Salary        : int [1:151] 19268 10900 19709 14521 16198 10961 12904 17219 10406 16740 ...
##   ..$ blood_type    : int [1:151] 4 4 1 4 2 1 4 3 4 2 ...
##   ..$ Sugar_in_blood: num [1:151] 92.7 106.2 96 92.3 101 ...
##   ..$ bmi           : num [1:151] 35.9 35.9 27.1 18.8 27.1 ...
```

The file **human3** is a list with six components, each of which is a data frame. The components correspond to the combination of two genders and three ocupation status.

(d) Using the data in **human3** obtain numerical summaries for the variable **Salary** for males and females that work and compare.

```
summary(human3$Work.F$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10011   11130   12772   13960   15696   44232
```

```
summary(human3$Work.M$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10012   11245   12904   14386   16163   27354
```

Except for the maxima which have the same values as before, the values for the other parameters are closer than when we looked at the complete data set.

(e) The function **cut** divides the range of values of a continuous variable into intervals and creates a factor according to which interval they fall. You have to use this function to divide the range of salaries in the file **human** into three intervals, according to the following scheme: between 2000 and 8000 it is **low**, between 8000 and 18000 it is **medium**, and more than 18000 it is **high**. Call the resulting factor **sal**. Use the function **table** to count how many subjects fall in each category.

```
sal <- cut(human$Salary,c(2000,8000, 18000, 45000), labels = c('low','medium','high'))
table(sal)
```

```
## sal
##    low medium   high
##     75    272     43
```

(f) Using the factor **sal** and the variable **Gender**, split the file **human** and call the resulting file **human4**. Using this file, obtain numerical summaries for the variable **Salary** for males and females that have a high salary and compare.

```
human4 <- split(human, list(sal, human$Gender))
str(human4)
```

```
## List of 6
##  $ low.F   :'data.frame':    47 obs. of  11 variables:
##   ..$ Index         : int [1:47] 4 9 47 55 62 103 122 126 129 141 ...
##   ..$ Gender        : chr [1:47] "F" "F" "F" "F" ...
##   ..$ age           : int [1:47] 24 21 23 22 21 20 22 20 22 24 ...
##   ..$ Ocupation     : chr [1:47] "student" "student" "student" "student" ...
```

```
##   ..$ Head_size     : num [1:47] 24.8 27.6 23.5 26.5 24.9 27.9 29.9 22.7 26.8 29.6 ...
##   ..$ Height_cm     : num [1:47] 156 164 145 159 155 ...
##   ..$ Weight_kg     : num [1:47] 56 82.2 82.7 75.3 51.7 75.1 92.7 40.1 64.7 73.5 ...
##   ..$ Salary        : int [1:47] 2034 2068 2271 2157 2415 2539 2345 2008 2064 2443 ...
##   ..$ blood_type    : int [1:47] 3 1 3 1 4 3 1 4 2 2 ...
##   ..$ Sugar_in_blood: num [1:47] 95.8 80.5 91.2 95.5 86.7 87.1 94.5 99 93.1 81.2 ...
##   ..$ bmi           : num [1:47] 23 30.7 39.1 29.9 21.6 ...
##  $ medium.F:'data.frame':    147 obs. of  11 variables:
##   ..$ Index         : int [1:147] 5 6 7 8 10 15 21 24 31 36 ...
##   ..$ Gender        : chr [1:147] "F" "F" "F" "F" ...
##   ..$ age           : int [1:147] 37 31 38 38 31 33 49 42 44 50 ...
##   ..$ Ocupation     : chr [1:147] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:147] 30.1 26.6 25.6 25.6 23.6 25.4 26.5 23.7 24.6 26.6 ...
##   ..$ Height_cm     : num [1:147] 173 158 152 153 149 ...
##   ..$ Weight_kg     : num [1:147] 103.3 47 46.5 64.6 52.1 ...
##   ..$ Salary        : int [1:147] 14829 10586 11272 13048 12326 17270 11992 11996 12646 10412 ...
##   ..$ blood_type    : int [1:147] 2 3 4 2 3 1 1 4 3 2 ...
##   ..$ Sugar_in_blood: num [1:147] 114.1 95.1 82.7 112 100.6 ...
##   ..$ bmi           : num [1:147] 34.6 18.9 20.1 27.7 23.5 ...
##  $ high.F  :'data.frame':    17 obs. of  11 variables:
##   ..$ Index         : int [1:17] 50 65 85 88 95 116 168 221 245 299 ...
##   ..$ Gender        : chr [1:17] "F" "F" "F" "F" ...
##   ..$ age           : int [1:17] 48 36 53 54 43 31 46 40 26 45 ...
##   ..$ Ocupation     : chr [1:17] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:17] 24.9 26.9 30.3 28.1 25.9 30.3 26.6 30.7 24.5 24.1 ...
##   ..$ Height_cm     : num [1:17] 145 158 186 160 158 ...
##   ..$ Weight_kg     : num [1:17] 56.6 61.4 85.5 66.2 69.1 75.1 62.7 87.2 71.4 48.1 ...
##   ..$ Salary        : int [1:17] 20360 23772 18025 21671 26658 18568 18372 20351 20065 21314 ...
##   ..$ blood_type    : int [1:17] 1 2 4 3 3 2 3 4 2 1 ...
##   ..$ Sugar_in_blood: num [1:17] 101.1 81.6 94.5 96.8 97 ...
##   ..$ bmi           : num [1:17] 26.9 24.6 24.6 25.8 27.6 ...
##  $ low.M   :'data.frame':    28 obs. of  11 variables:
##   ..$ Index         : int [1:28] 11 63 94 108 110 111 124 128 155 163 ...
##   ..$ Gender        : chr [1:28] "M" "M" "M" "M" ...
##   ..$ age           : int [1:28] 19 18 24 24 21 20 19 22 22 21 ...
##   ..$ Ocupation     : chr [1:28] "student" "student" "student" "student" ...
##   ..$ Head_size     : num [1:28] 32.3 31.6 29.4 28 26 28.9 30.8 31.5 31.3 31.3 ...
##   ..$ Height_cm     : num [1:28] 190 190 174 174 165 ...
##   ..$ Weight_kg     : num [1:28] 109.6 110.2 81.5 80.4 89.3 ...
##   ..$ Salary        : int [1:28] 2493 2739 2028 2083 2063 2263 2427 3402 2152 2269 ...
##   ..$ blood_type    : int [1:28] 2 4 2 1 2 3 3 4 1 1 ...
##   ..$ Sugar_in_blood: num [1:28] 95.7 80.9 108.7 108.8 91.3 ...
##   ..$ bmi           : num [1:28] 30.3 30.6 27 26.7 32.8 ...
##  $ medium.M:'data.frame':    125 obs. of  11 variables:
##   ..$ Index         : int [1:125] 12 16 17 18 20 25 28 29 35 37 ...
##   ..$ Gender        : chr [1:125] "M" "M" "M" "M" ...
##   ..$ age           : int [1:125] 38 35 29 52 41 45 55 50 52 28 ...
##   ..$ Ocupation     : chr [1:125] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:125] 27.9 28.7 30.8 29.5 26.3 30.2 28.1 30.4 31.5 32.5 ...
##   ..$ Height_cm     : num [1:125] 165 172 182 180 163 ...
##   ..$ Weight_kg     : num [1:125] 98 55.7 90.1 95.9 58.8 ...
##   ..$ Salary        : int [1:125] 10900 14521 16198 10961 12904 17219 10406 16740 10170 10540 ...
##   ..$ blood_type    : int [1:125] 4 4 2 1 4 3 4 2 1 1 ...
##   ..$ Sugar_in_blood: num [1:125] 106.2 92.3 101 95.9 85.9 ...
```

```
##   ..$ bmi           : num [1:125] 35.9 18.8 27.1 29.4 22.2 ...
##  $ high.M  :'data.frame':    26 obs. of  11 variables:
##   ..$ Index         : int [1:26] 3 14 30 57 70 87 93 98 158 165 ...
##   ..$ Gender        : chr [1:26] "M" "M" "M" "M" ...
##   ..$ age           : int [1:26] 46 57 37 44 34 50 60 48 58 52 ...
##   ..$ Ocupation     : chr [1:26] "Work" "Work" "Work" "Work" ...
##   ..$ Head_size     : num [1:26] 27 27.7 32 27.7 27.9 27.7 31.5 25.7 30 27.9 ...
##   ..$ Height_cm     : num [1:26] 162 167 189 171 174 ...
##   ..$ Weight_kg     : num [1:26] 94.7 75.4 100.8 92 92.3 ...
##   ..$ Salary        : int [1:26] 19268 19709 23514 18945 22760 21741 18905 20455 26258 21811 ...
##   ..$ blood_type    : int [1:26] 4 1 2 1 1 1 1 1 4 4 ...
##   ..$ Sugar_in_blood: num [1:26] 92.7 96 81.5 83.9 81 ...
##   ..$ bmi           : num [1:26] 35.9 27.1 28.3 31.6 30.5 ...
```

```
summary(human4$high.F$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18025   18678   20351   22980   21671   44232
```

```
summary(human4$high.M$Salary)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18905   19930   21776   22272   24252   27354
```

In this final comparison, the only important change is that the average value for female salary is now above the male salary while the other parameters remain below, except for the maximum value which we already know corresponds to a female. The reason for this change is the effect of two large female salaries on a set of reduced size. See the boxplots below.

```
boxplot(Salary ~ Gender, data = human[sal == 'high',])
```