Genes-Proteins

The Genes Dataset

Given is a table of information about genes. There is one row for each gene. This row contains also information about one transcript of the certain gene. Even if genes can have multiple transcripts, the table contains only one gene-transcript association (it could be the most important one within a certain biological context).

!) Read in the genes table ("genes.csv") and assign it to the variable "genes". Show the head of this table.

```
##           gene.ID   transcript.ID gene.chromosome gene.start  gene.end
## 1 ENSG00000268991 ENST00000600179               1   16539066  16539575
## 2 ENSG00000058673 ENST00000488411               1  203795654 203854999
## 3 ENSG00000133055 ENST00000621380               1  203167811 203175813
## 4 ENSG00000174059 ENST00000356522               1  207880972 207911402
## 5 ENSG00000163431 ENST00000616739               1  201896452 201946588
## 6 ENSG00000117834 ENST00000471020               1   48222685  48248644
##   gene.name gene.transcript.count gene.GC.content      gene.type
## 1   FAM231B                     1           56.86 protein_coding
## 2   ZC3H11A                    15           40.73 protein_coding
## 3     MYBPH                     2           57.62 protein_coding
## 4      CD34                     4           45.46 protein_coding
## 5     LMOD1                     2           48.52 protein_coding
## 6    SLC5A9                    10           48.99 protein_coding
##   transcript.start transcript.end transcription.start.site
## 1         16539066       16539575                 16539066
## 2        203850174      203851122                203850174
## 3        203167811      203175783                203175783
## 4        207886539      207911402                207911402
## 5        201896456      201946588                201946588
## 6         48240384       48248638                 48240384
##        transcript.type uniprot.accession
## 1       protein_coding            A6NCW3
## 2 processed_transcript            O75152
## 3       protein_coding            Q13203
## 4       protein_coding            P28906
## 5       protein_coding            P29536
## 6 processed_transcript            Q2M3M2
```

The first two columns are the IDs, one for the gene, one for the transcript. Furthermore, the table contains several columns describing the position of the gene (chromosome, start, end) and also the start and end position of the transcript. As already mentioned there is only one gene-transcript association per gene in the table, but the column "gene.transcript.count" tells us the total number of transcripts of a gene. There is also information about the gene name, type and GC content and the transcript type. For protein coding genes the last column provides the UniProt accession that can be linked to entries in the UniProt database.

Exercise 1

a) Show the number of rows (number of genes) and then the column names in the given dataset.

```
## [1] 29503
```

```
##  [1] "gene.ID"                  "transcript.ID"
##  [3] "gene.chromosome"          "gene.start"
```

```
##  [5] "gene.end"                  "gene.name"
##  [7] "gene.transcript.count"     "gene.GC.content"
##  [9] "gene.type"                 "transcript.start"
## [11] "transcript.end"            "transcription.start.site"
## [13] "transcript.type"           "uniprot.accession"
```

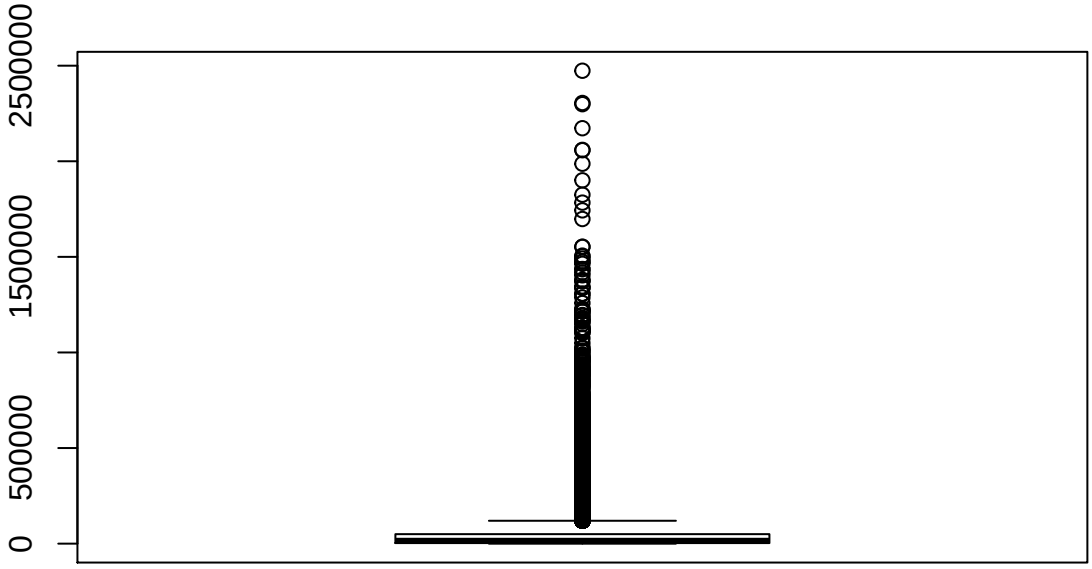
b) Show the maximum transcript count

```
## [1] 193
```

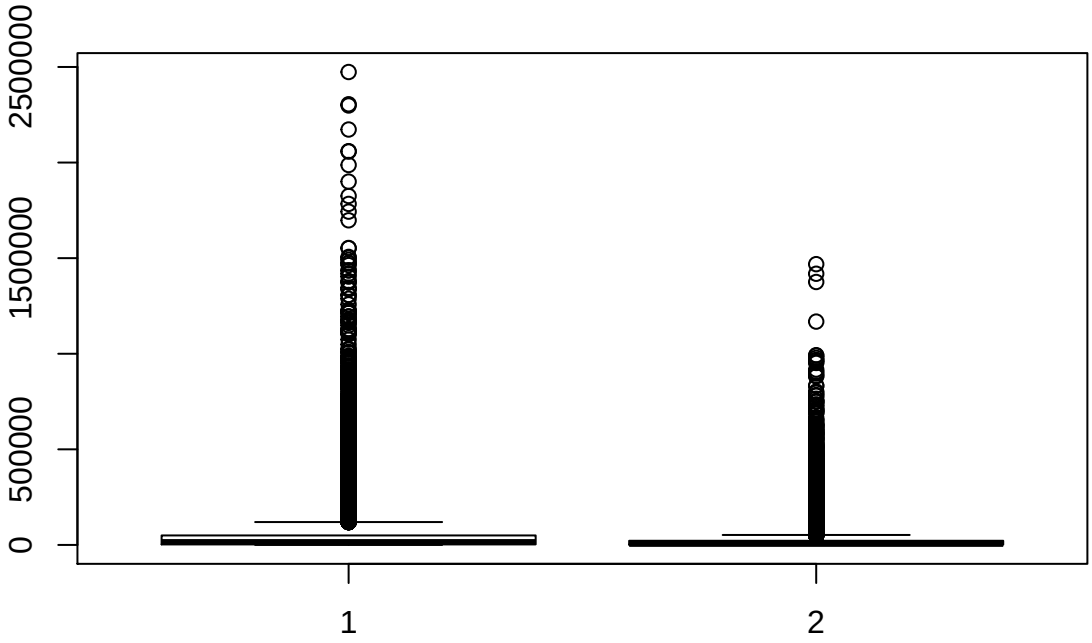
Exercise 2

a) Now we would like to analyze some details about the gene length and transcript length. The start and end positions can be used to calculate the gene length and transcript length for each row in the table. Calculate 2 new vectors and add them as 2 new columns to "genes" for this purpose (name the columns gene.len and transcript.len).

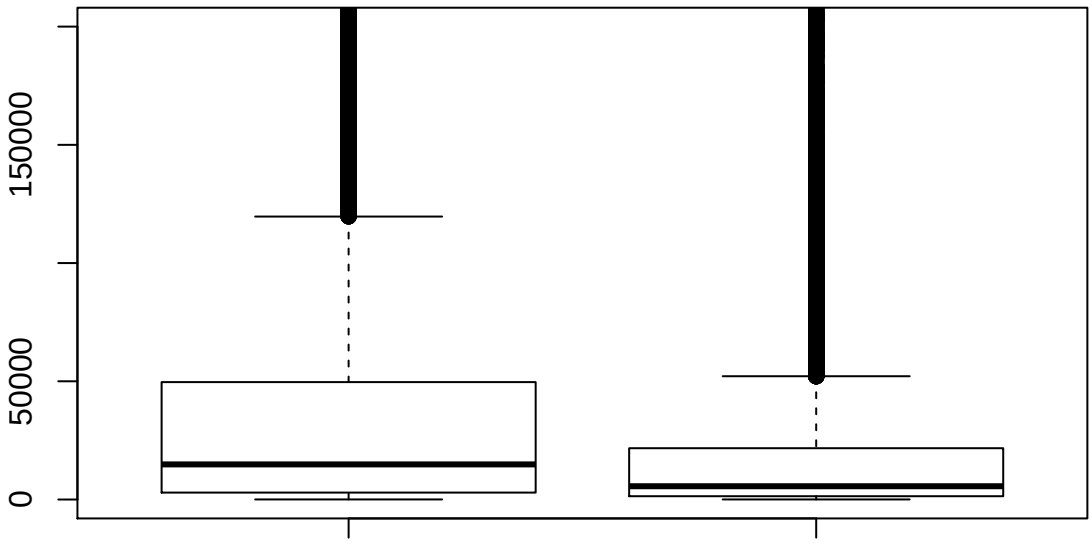
b) Use the *boxplot()* function to show the boxplot for the gene length.



c) Simply add the column for the transcript length within parameters of the *boxplot()* function (parameters are separated by commas). In this way two boxplots are created and you can compare them.



d) As you can see, the maximum outliers strongly influence the boxplot and it is hard to compare the medians and quartiles. Within the *boxplot()* function you can use the *xlim* and *ylim* variable to set the limits for the x-axis and y-axis, respectively. Change the limits of the y-axis to 0 and 200000. To do so, you must assign a vector to *ylim* within the *boxplot()* function. (add *ylim=c(0,200000)* within the brackets separated by a comma)



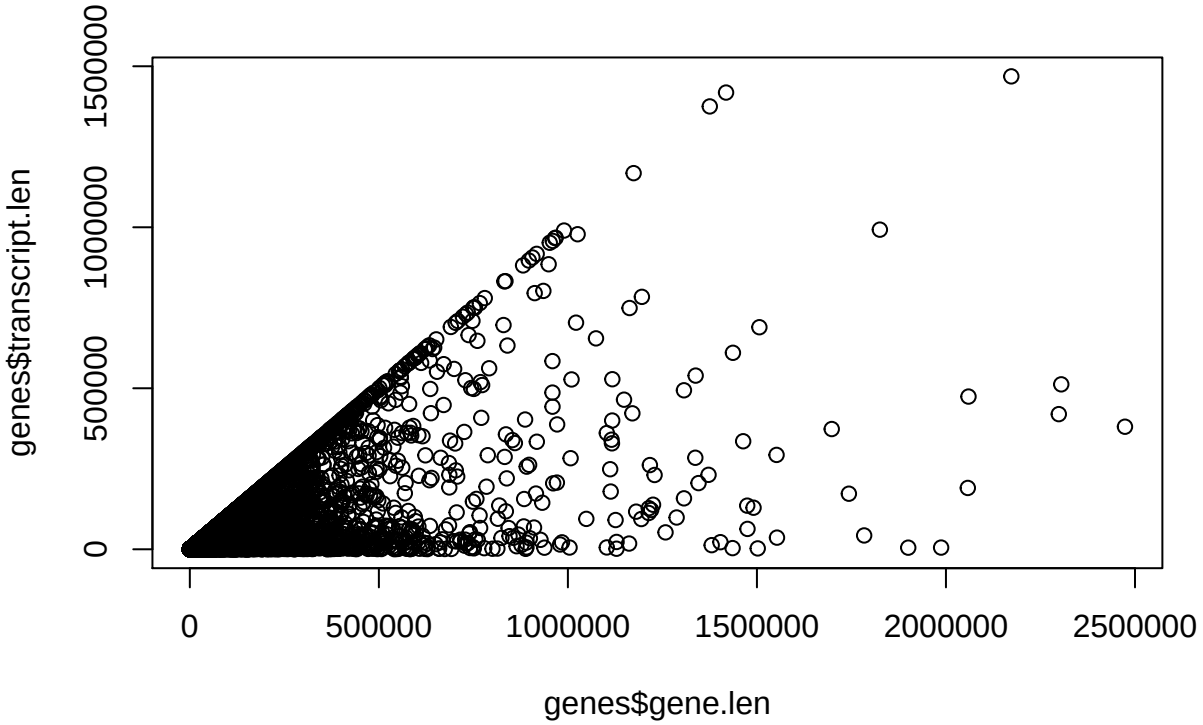
What is your observation? Does your observation fit to the biological context?

e) A paired t-test to compare vectors x and y can be written as follows: *t.test(x, y, paired=TRUE)*. Perform a paired t-test in order to assess the statistical significance of the difference in length for genes and corresponding transcripts.

```
##
##  Paired t-test
##
## data:  genes$gene.len and genes$transcript.len
## t = 51.203, df = 29502, p-value < 2.2e-16
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  24084.59 26001.89
## sample estimates:
## mean of the differences
##                25043.24
```

f) Create a simple scatter plot with the gene length on the x-axis and the transcript length on the y-axis.



Is there something interesting to observe in this scatter plot? Think about the previous findings within this exercise.

Exercise 3

a) Let's have a detailed look at the chromosome and the gene types, especially protein coding genes. For each chromosome show the mean value of the gene length, then the mean value of the transcript length, and finally the mean value of the CG content.

```
##        1        2        3        4        5        6        7        8
## 43366.29 64162.78 71707.78 68441.19 61297.25 53326.99 65373.51 62618.93
##        9       10       11       12       13       14       15       16
## 50845.91 61719.29 42862.35 49277.57 62087.69 49082.84 51412.93 37273.70
##       17       18       19       20       21       22
## 30879.05 65345.43 22224.09 38329.70 41899.20 33990.01
```

```
##        1        2        3        4        5        6        7        8
## 23096.85 32603.33 34021.57 35729.23 29853.71 27716.54 31883.77 29692.37
##        9       10       11       12       13       14       15       16
## 30665.46 31571.85 21621.89 23729.29 34333.74 24953.61 25069.05 17805.83
##       17       18       19       20       21       22
## 13479.04 33442.39 11439.47 20288.62 24957.36 20046.50
```

```
##        1        2        3        4        5        6        7        8
## 46.01839 43.91479 43.39250 40.97645 43.07182 43.99544 45.03877 43.65128
##        9       10       11       12       13       14       15       16
## 46.52766 44.66863 46.82821 44.19282 41.81781 44.97824 44.39175 50.22671
##       17       18       19       20       21       22
## 49.69819 42.20131 51.58970 47.97453 46.28135 50.99586
```

(in this way you can define the chromosome with the shortest genes in average or the highest amount of GCs or . . .)

b) Show the number of genes from type "protein_coding".

```
## [1] 18920
```

c) In percent, how many genes are protein coding genes?

```
## [1] 64.12907
```

d) Show the number of genes for all chromosome-gene.type combinations. Which chromosome is the one with the most protein coding genes? Does this chromosome have the most lincRNAs or snoRNAs, too?

```
##
##      lincRNA protein_coding pseudogene rRNA scaRNA snoRNA snRNA sRNA
##   1      602           2040          5   69     14     68   219    0
##   2      573           1249          1   41      7     64   157    0
##   3      346           1072          0   29      2     56   133    1
##   4      400            747          1   24      1     30   116    0
##   5      489            882          0   25      0     40   103    0
##   6      356           1035          0   27      1     39   107    0
##   7      279            902          3   24      0     39    84    1
##   8      408            668          0   29      0     33    84    0
##   9      272            769          1   19      1     31    65    0
##   10     277            728          1   29      0     28    86    0
##   11     299           1279          0   24      3     52    71    2
##   12     437           1033          0   27      2     37    99    0
##   13     239            323          0   15      0     17    41    0
##   14     293            611          1   10      2     72    64    0
##   15     296            590          1   12      3    114    64    0
##   16     353            858          1   32      1     33    52    0
##   17     335           1185          0   17      5     51    86    0
##   18     247            268          0   13      2     17    46    1
##   19     235           1468          0   13      0     22    29    0
##   20     213            540          0   20      1     25    46    0
##   21     193            234          1    9      0     15    21    0
##   22     165            439          0    6      3     11    26    0
```

Exercise 4

a) For the GC content, show the minimum, maximum, median, mean, standard deviation and variation.

```
## [1] 23.82
```

```
## [1] 76.96
```

```
## [1] 44.15
```

```
## [1] 45.64717
```

```
## [1] 7.814933
```

```
## [1] 61.07318
```

b) For a subset of genes having a lower GC content, calculate the percentage of protein coding genes. First, create a subset containing the genes with lower GC content than the median GC content. Then calculate the percentage of protein coding genes based on this subset.

```
## [1] 57.36371
```

c) The same as in b) but for genes with a GC content greater than or equal to the median GC content. Based on another subset of genes having a higher GC content than the median GC content, calculate the percentage of protein coding genes of this subset.

```
## [1] 70.89122
```

Obviously, the proportion of protein coding genes is higher in the set of genes having a higher GC content.

d) Show the median GC content for each type of gene.

```
##        lincRNA protein_coding     pseudogene           rRNA         scaRNA
##         42.430         45.390         56.095         49.570         49.940
##         snoRNA          snRNA           sRNA
##         43.670         40.190         43.330
```

(e) NOT mandatory) To practice, calculate the percentages for other gene types or other values for the GC content. Or furthermore, create another own scenario for some small exercises to get used to the different functions.

Exercise 5 - Inlcude protein information

We found out that about 65 % of the genes are protein coding genes. So let's include protein data.

!) Read in the file from UniProt ("uniprot.csv") and show the head. Use a variable with the name "uniprot" for this table.

```
##   uniprot.accession  entry.name   status   gene.names protein.mass.Da
## 1            Q96EK9 KTI12_HUMAN reviewed KTI12 SBBI81           38616
## 2        A0A087WSZ0 KVD08_HUMAN reviewed     IGKV1D-8           12837
## 3            P01611 KVD12_HUMAN reviewed    IGKV1D-12           12620
## 4        A0A0C4DH25 KVD20_HUMAN reviewed    IGKV3D-20           12515
## 5            P01615 KVD28_HUMAN reviewed    IGKV2D-28           12957
## 6            P04432 KVD39_HUMAN reviewed    IGKV1D-39           12737
##   protein.length RNA.editing
## 1            354          no
## 2            117          no
## 3            117          no
## 4            116          no
## 5            120          no
## 6            117          no
```

As you see, the first column is the "UniProt accession". This table provides some information about some entries in UniProt. Notice, this is a comparatively small table with only a few columns, there is way more information available in such databases. The last three columns are important for this exercise. Two columns describe the mass and length of the protein. The last column tells us if the protein is a RNA editing protein.

a) Merge the two tables "genes" and "uniprot" and assign this new table to the variable "gene.prot".

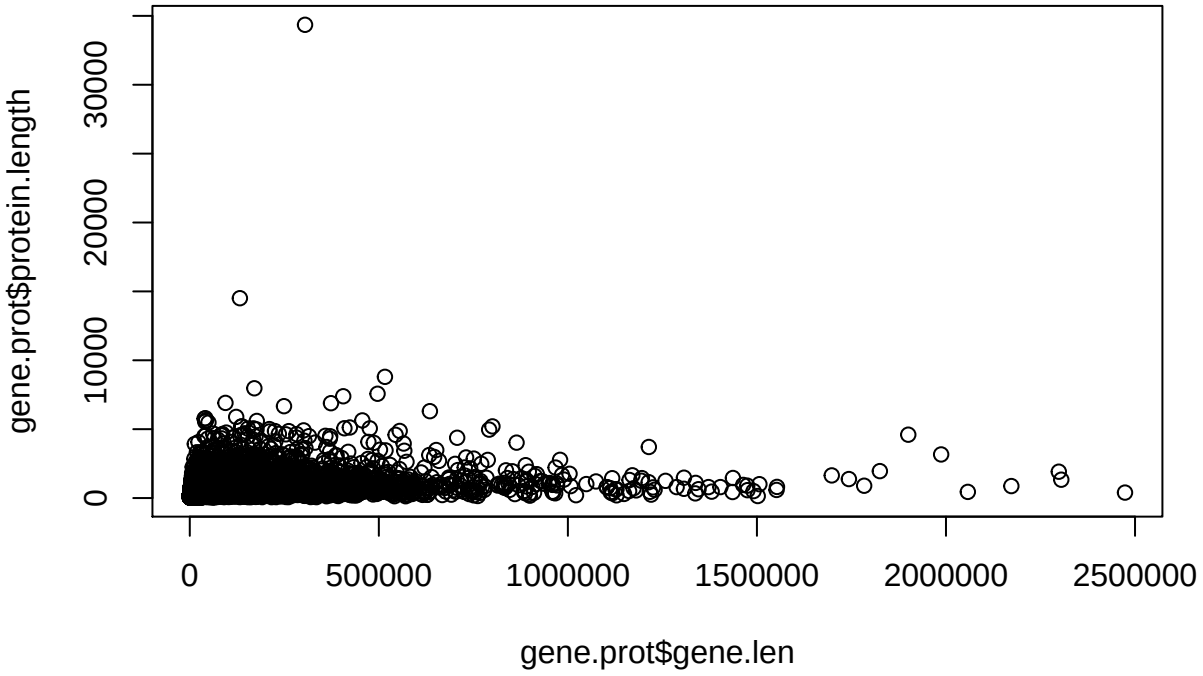
b) Again, calculate the percentage of protein coding genes in this new table. Are you surprised by this percentage?

```
## [1] 100
```

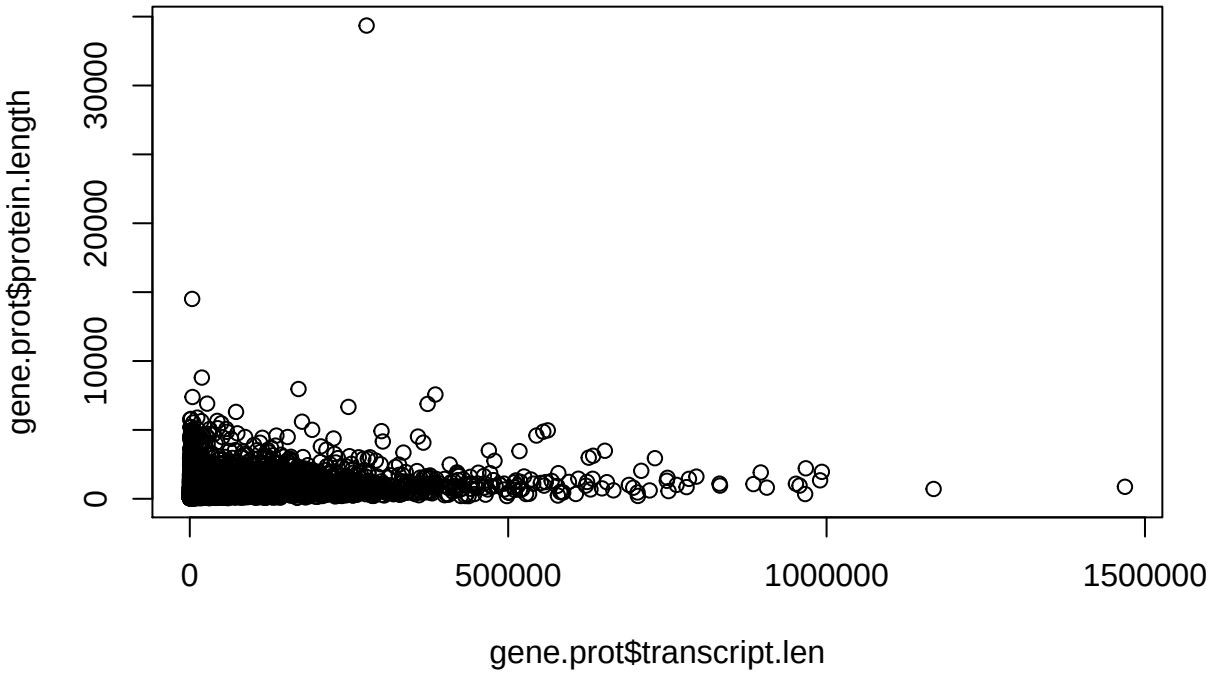
Exercise 6 - use the new table "gene.prot"

Finally, the set only contains protein coding genes and some basic information about the proteins UniProt entry. Now, that we know the length of the proteins which are linked to the genes, we would like to know if the gene length is in any statistical relation to the protein length. As we know that the transcript of a protein coding gene is later translated into a protein we might expect that protein sequences are longer if it's gene is also longer.

a) Create a scatter plot with the gene length on the x-axis and the protein length on the y-axis.



b) Change the x-axis to the length of the transcript.



c) It seems to be that the scatter plots are not very helpful to see if longer proteins are coded by longer genes. Calculate the correlation coefficient between the gene length and the protein length. And also calculate the correlation coefficient between the transcript length and the protein length.

```
## [1] 0.2910942
```

```
## [1] 0.2292858
```

Is there a difference? Why don't we observe a higher statistical relationship between the gene respectively transcript length and protein length? Which biological process could influence this?

d) A simple question about the proteins themselves: How many percent of the proteins are RNA editing proteins?

```
## [1] 0.1052107
```

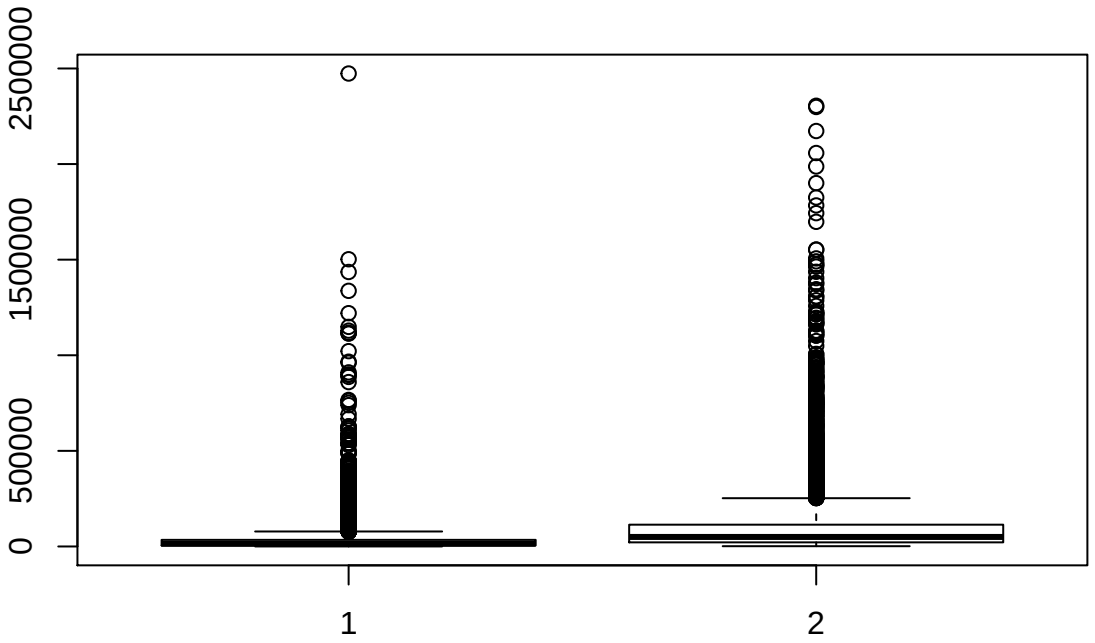
Exercise 7

a) Show the maximum mass of a protein for each chromosome where the protein coding gene is located.

```
##       1       2       3       4       5       6       7       8       9
##  868484 3816030  493953  555482  693069 1011086  576159  531791  541978
##      10      11      12      13      14      15      16      17      18
##  480410  629101  593389  521126  796442  552042  575892  591407  366649
##      19      20      21      22
## 1519175  399737  378037  329486
```

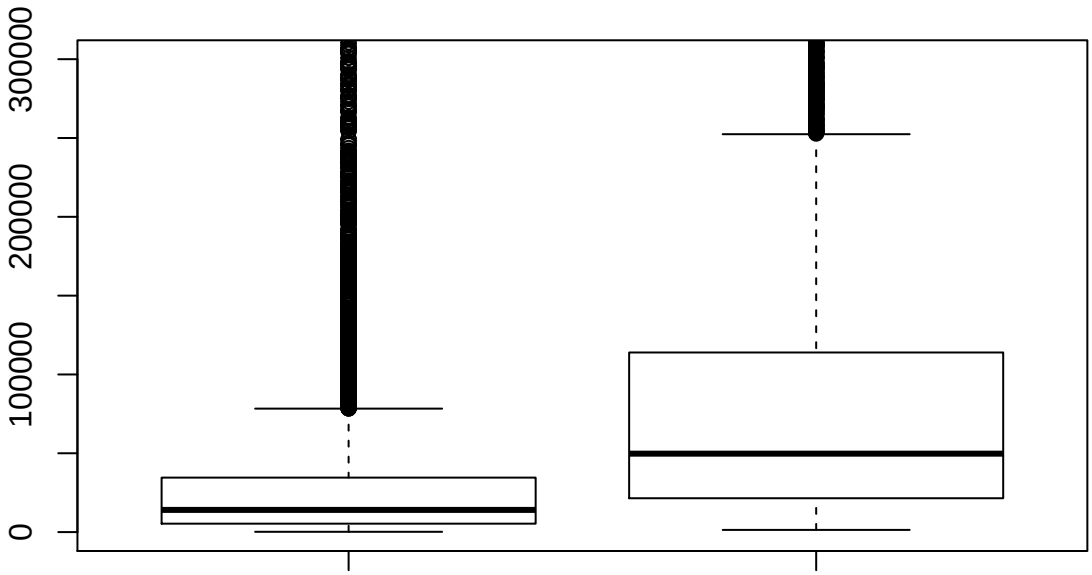
b) Create two subsets. One containing all rows having a protein mass smaller than the median of the protein mass. The other subset with a protein mass greater or equal than its median. In this way you create two subsets of gene-protein associations. One for "light" proteins with a lower mass and one for "heavy" protein with a higher mass. Both subsets should contain all columns of the gene.prot data frame.

c) Use the *boxplot()* function with two vectors, one for the length of genes coding "light" proteins and one for the length of genes coding "heavy" proteins.



Again, the maximum outliers have a strong impact and it is hard to see a trend with respect to the median or the quartiles.

d) Plot the boxplot again, but now set the limits of the y-axis to *0* and *300000*.



Would you say, that there is a significant difference between "light" and "heavy" proteins with respect to the length of their protein coding genes?

e) Perform an unpaired t-test to the two vectors, gene length for "light" proteins and gene length for "heavy" proteins.

```
##
##  Welch Two Sample t-test
##
## data:  light$gene.len and heavy$gene.len
## t = -34.409, df = 13316, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -68941.77 -61510.53
## sample estimates:
## mean of x mean of y
##  35787.53 101013.68
```

f) The same as in e), do it for the transcript length.

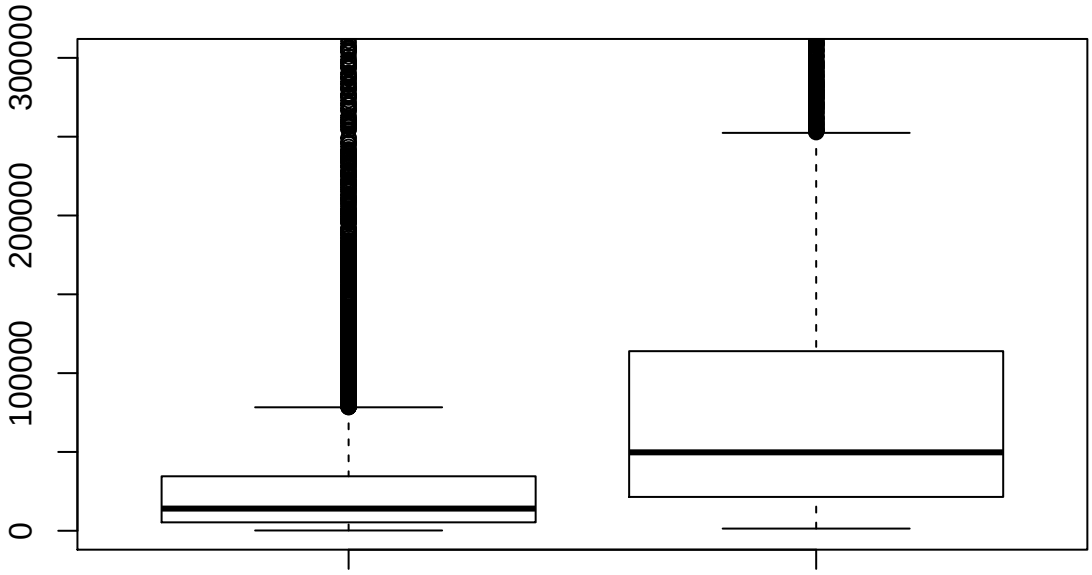
```
##
##  Welch Two Sample t-test
##
## data:  light$transcript.len and heavy$transcript.len
## t = -27.904, df = 12224, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -29322.50 -25473.27
## sample estimates:
## mean of x mean of y
##  16078.95  43476.84
```

Exercise 8

So far we always used all columns when we created particular subsets. For instance, in exercise 7 we created 2 subsets divided by the protein mass and both subsets contained all columns. However, for 7c-f) we only used the length of the genes and transcripts, respectively.

a) Repeat exercise 7b), but now select only the two columns that are needed in c), d), e) and f).

b) Based on the subsets containing only the two columns needed, repeat 7 d), e) and f)




```
##
##  Welch Two Sample t-test
##
## data:  light$gene.len and heavy$gene.len
## t = -34.409, df = 13316, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -68941.77 -61510.53
## sample estimates:
## mean of x mean of y
##  35787.53 101013.68
```

```
##
##  Welch Two Sample t-test
##
## data:  light$transcript.len and heavy$transcript.len
## t = -27.904, df = 12224, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -29322.50 -25473.27
## sample estimates:
## mean of x mean of y
##  16078.95  43476.84
```

In case you did it in the right way, the results from 8b) should be the same as in 7.