Data loading

For this exercise you need to import the Alzheimer ROSMAP dataset (file RM_xsect.csv).

```
RM = read_csv("./datasets/Rosmap/RM_xsect.csv")

Parsed with column specification:
cols(
  ranid = col_double(),
  age_death = col_double(),
  educ = col_double(),
  msex = col_double(),
  time_in_study = col_double(),
  gpath = col_double(),
  global_bl = col_double(),
  global_lv = col_double(),
  pathoAD = col_double(),
  apoe4d = col_double(),
  ceradsc = col_double(),
  braaksc = col_double(),
  cad = col_double(),
  cad_year = col_double(),
  globcog_slope = col_double(),
  pmad = col_double(),
  np = col_double(),
  nft = col_double()
)
```

Part A: Subsetting data

1. Let's consider the following sequence: x <- 1:100, what will the following statements produce?

(a) x[50:55]

```
[1] 50 51 52 53 54 55
```

(b) x[c(1,5,99)]

```
[1]  1  5 99
```

2. Subset the ROSMAP data to only males that have an *APOE* variant (column apoe4d is equal to 1) and save the result to RM_males_apoe

```
which_males_with_apoe = RM$msex==1 & RM$apoe4d == 1
RM_males_apoe <- RM[which_males_with_apoe,]
```

3. How many males with an *APOE* variant are there in the data? 127

4. We now want to restrict our analysis to the following columns: ranid, age_death, global_bl and global_lv. Subset the RM_males_apoe data frame to just these columns. How can you do it?

```
RM_males_apoe <- RM_males_apoe[, c('ranid', 'age_death', 'global_bl', 'global_lv')]
```

Part B: Combining data

1. A colleague has found 4 additional Alzheimer male patients with an APOE variant that you can load into the new_patients variable using:

```
new_patients <- data.frame(
  ranid=c(2500, 2501, 2502, 2503),
  age_death=c(85.1, 88.4, 78.9, 77.8),
  global_bl=c(0.134, -0.4, 0.055, 0.343),
  global_lv=c(0.233, -0.43, -1.1, 0.451)
)
```

How can you add these additional patients to your data (save the results in RM_males_apoe)

```
RM_males_apoe <- rbind(RM_males_apoe, new_patients)
```

2. You become aware that the sample ids in your data (column ranid) are identifiable and that you need to use new identifiers. Specifically, your given the following code that will generate a vector of random ids:

```
new_ids <- sample(nrow(RM_males_apoe), replace = FALSE)
```

You now want to replace the values in the ranid column with the values from new_ids.

(a) By removing the existing the ranid column and using the cbind function to add the new one.

```
RM_males_apoe <- RM_males_apoe[, which(colnames(RM_males_apoe)!='ranid')]
RM_males_apoe <- cbind(RM_males_apoe, ranid=new_ids)
```

(b) By using the $ operator

```
RM_males_apoe$ranid = new_ids
```

Part C: Joining data

1. Consider the following data frames:

df1:

x	y
1	a
2	b
3	c

df2:

xx	y
4	a
5	b
6	d

(a) Which column(s) will merge(df1, df2) join on? Column y

(b) What type of join would produce the following result? Outer join

y	x	xx
a	1	4
b	2	5
c	3	NA
d	NA	6

(c) What type of join would produce the following result? Left join

y	x	xx
a	1	4
b	2	5
d	NA	6

(d) What type of join would produce the following result? Inner join

y	x	xx
a	1	4
b	2	5

2. Another collaborator has given you new data specifying whether 10 of the sample’s fathers was diagnosed with Alzheimer (column father_az):

```
father_data  <- data.frame(
 id=125:135,
 father_az=c(F,T,T,T,T,F,F,T,F,F,T)
)
```

(a) Join your data (RM_males_apoe) with the father_data data all rows present in the RM_males_apoe (but not those specific to father_data, if any). Note that the sample id in father_data is named id and not ranid!

```
RM_males_apoe  <- merge(
 RM_males_apoe,
 father_data,
 by.x = c('ranid'),
 by.y = c('id'),
 all.x = TRUE
   )
```

(b) How many of the 10 samples you got father data for were in RM_males_apoe?

```
length(which(!is.na(RM_males_apoe$father_az)))

[1] 7
```