Exercise 1.5

Data clean up

useful commands: `duplicated()`, `is.na()`, `complete.cases()`, `!duplicated()`

load the sleep data as in exercise 1.4

this data describes properties of sleep for different mammals

column names are:

- BodyWgt body weight (kg)
- BrainWgt brain weight (g)
- NonD slow wave (non-dreaming) sleep (hours/day)
- Dream paradoxical (dreaming) sleep (hours/day)
- Sleep total sleep (hours/day)
- Span maximum life span (years)
- Gest gestation time (days)
- Pred Predation index (1-5) 1 = minimum (least likely to be preyed upon), 5 = max
- Exp sleep exposure index (1-5) 1 = least exposed (sleeps in well-protected den), 5 = most exposed
- Danger overall danger index (1-5) 1 = least danger (from other animals), 5 = most danger

```
sleep = read.csv('sleep.csv')
```

* 01. How many duplicated rows are in the dataset?

```
sum(duplicated(sleep))
```

```
## [1] 6
```

02. Remove the duplicated rows from the datset

```
dim(sleep)
```

```
## [1] 68 10
```

```
sleep = sleep[!duplicated(sleep),]
dim(sleep)  ## note that there are six fewer rows now
```

```
## [1] 62 10
```

* 03. Subset the dataset to only include records with data on Gestation period

```
sleep[!is.na(sleep[,'Gest']),]  ## so we are only taking rows where Gest is not (!) NA
```

```
##     BodyWgt BrainWgt NonD Dream Sleep  Span  Gest Pred Exp Danger
## 1  6654.000  5712.00   NA    NA   3.3  38.6 645.0    3   5      3
## 2     1.000     6.60  6.3   2.0   8.3   4.5  42.0    3   1      3
## 3     3.385    44.50   NA    NA  12.5  14.0  60.0    1   1      1
## 4     0.920     5.70   NA    NA  16.5    NA  25.0    5   2      3
## 5  2547.000  4603.00  2.1   1.8   3.9  69.0 624.0    3   5      4
## 6    10.550   179.50  9.1   0.7   9.8  27.0 180.0    4   4      4
## 7     0.023     0.30 15.8   3.9  19.7  19.0  35.0    1   1      1
## 8   160.000   169.00  5.2   1.0   6.2  30.4 392.0    4   5      4
## 9     3.300    25.60 10.9   3.6  14.5  28.0  63.0    1   2      1
## 10   52.160   440.00  8.3   1.4   9.7  50.0 230.0    1   1      1
## 11    0.425     6.40 11.0   1.5  12.5   7.0 112.0    5   4      4
## 12  465.000   423.00  3.2   0.7   3.9  30.0 281.0    5   5      5
## 14  187.100   419.00   NA    NA   3.1  40.0 365.0    5   5      5
## 15    0.075     1.20  6.3   2.1   8.4   3.5  42.0    1   1      1
## 16    3.000    25.00  8.6   0.0   8.6  50.0  28.0    2   2      2
## 17    0.785     3.50  6.6   4.1  10.7   6.0  42.0    2   2      2
## 18    0.200     5.00  9.5   1.2  10.7  10.4 120.0    2   2      2
## 21  529.000   680.00   NA   0.3    NA  28.0 400.0    5   5      5
## 22   27.660   115.00  3.3   0.5   3.8  20.0 148.0    5   5      5
## 23    0.120     1.00 11.0   3.4  14.4   3.9  16.0    3   1      2
## 24  207.000   406.00   NA    NA  12.0  39.3 252.0    1   4      1
## 25   85.000   325.00  4.7   1.5   6.2  41.0 310.0    1   3      1
## 26   36.330   119.50   NA    NA  13.0  16.2  63.0    1   1      1
## 27    0.101     4.00 10.4   3.4  13.8   9.0  28.0    5   1      3
## 28    1.040     5.50  7.4   0.8   8.2   7.6  68.0    5   3      4
## 29  521.000   655.00  2.1   0.8   2.9  46.0 336.0    5   5      5
## 30  100.000   157.00   NA    NA  10.8  22.4 100.0    1   1      1
## 31   35.000    56.00   NA    NA    NA  16.3  33.0    3   5      4
## 32    0.005     0.14  7.7   1.4   9.1   2.6  21.5    5   2      4
## 33    0.010     0.25 17.9   2.0  19.9  24.0  50.0    1   1      1
## 34   62.000  1320.00  6.1   1.9   8.0 100.0 267.0    1   1      1
## 35    0.122     3.00  8.2   2.4  10.6    NA  30.0    2   1      1
## 36    1.350     8.10  8.4   2.8  11.2    NA  45.0    3   1      3
## 37    0.023     0.40 11.9   1.3  13.2   3.2  19.0    4   1      3
## 38    0.048     0.33 10.8   2.0  12.8   2.0  30.0    4   1      3
## 39    1.700     6.30 13.8   5.6  19.4   5.0  12.0    2   1      1
## 40    3.500    10.80 14.3   3.1  17.4   6.5 120.0    2   1      1
## 41  250.000   490.00   NA   1.0    NA  23.6 440.0    5   5      5
## 42    0.480    15.50 15.2   1.8  17.0  12.0 140.0    2   2      2
## 43   10.000   115.00 10.0   0.9  10.9  20.2 170.0    4   4      4
## 44    1.620    11.40 11.9   1.8  13.7  13.0  17.0    2   1      2
## 45  192.000   180.00  6.5   1.9   8.4  27.0 115.0    4   4      4
## 46    2.500    12.10  7.5   0.9   8.4  18.0  31.0    5   5      5
## 47    4.288    39.20   NA    NA  12.5  13.7  63.0    2   2      2
## 48    0.280     1.90 10.6   2.6  13.2   4.7  21.0    3   1      3
## 49    4.235    50.40  7.4   2.4   9.8   9.8  52.0    1   1      1
## 50    6.800   179.00  8.4   1.2   9.6  29.0 164.0    2   3      2
```

```
## 51    0.750    12.30  5.7   0.9   6.6   7.0 225.0    2   2      2
## 52    3.600    21.00  4.9   0.5   5.4   6.0 225.0    3   2      3
## 53   14.830    98.20   NA    NA   2.6  17.0 150.0    5   5      5
## 54   55.500   175.00  3.2   0.6   3.8  20.0 151.0    5   5      5
## 55    1.400    12.50   NA    NA  11.0  12.7  90.0    2   2      2
## 57    0.900     2.60 11.0   2.3  13.3   4.5  60.0    2   1      2
## 58    2.000    12.30  4.9   0.5   5.4   7.5 200.0    3   1      3
## 59    0.104     2.50 13.2   2.6  15.8   2.3  46.0    3   2      2
## 60    4.190    58.00  9.7   0.6  10.3  24.0 210.0    4   3      4
## 61    3.500     3.90 12.8   6.6  19.4   3.0  14.0    2   1      1
## 62    4.050    17.00   NA    NA    NA  13.0  38.0    3   1      1
```

04. Subset the dataset to only include records with data on Dreaming and non dreaming sleep

```
sleep[!is.na(sleep[,'Dream']) & !is.na(sleep[,'NonD']),]
```

```
##     BodyWgt BrainWgt NonD Dream Sleep  Span  Gest Pred Exp Danger
## 2     1.000     6.60  6.3   2.0   8.3   4.5  42.0    3   1      3
## 5  2547.000  4603.00  2.1   1.8   3.9  69.0 624.0    3   5      4
## 6    10.550   179.50  9.1   0.7   9.8  27.0 180.0    4   4      4
## 7     0.023     0.30 15.8   3.9  19.7  19.0  35.0    1   1      1
## 8   160.000   169.00  5.2   1.0   6.2  30.4 392.0    4   5      4
## 9     3.300    25.60 10.9   3.6  14.5  28.0  63.0    1   2      1
## 10   52.160   440.00  8.3   1.4   9.7  50.0 230.0    1   1      1
## 11    0.425     6.40 11.0   1.5  12.5   7.0 112.0    5   4      4
## 12  465.000   423.00  3.2   0.7   3.9  30.0 281.0    5   5      5
## 13    0.550     2.40  7.6   2.7  10.3    NA    NA    2   1      2
## 15    0.075     1.20  6.3   2.1   8.4   3.5  42.0    1   1      1
## 16    3.000    25.00  8.6   0.0   8.6  50.0  28.0    2   2      2
## 17    0.785     3.50  6.6   4.1  10.7   6.0  42.0    2   2      2
## 18    0.200     5.00  9.5   1.2  10.7  10.4 120.0    2   2      2
## 19    1.410    17.50  4.8   1.3   6.1  34.0    NA    1   2      1
## 20   60.000    81.00 12.0   6.1  18.1   7.0    NA    1   1      1
## 22   27.660   115.00  3.3   0.5   3.8  20.0 148.0    5   5      5
## 23    0.120     1.00 11.0   3.4  14.4   3.9  16.0    3   1      2
## 25   85.000   325.00  4.7   1.5   6.2  41.0 310.0    1   3      1
## 27    0.101     4.00 10.4   3.4  13.8   9.0  28.0    5   1      3
## 28    1.040     5.50  7.4   0.8   8.2   7.6  68.0    5   3      4
## 29  521.000   655.00  2.1   0.8   2.9  46.0 336.0    5   5      5
## 32    0.005     0.14  7.7   1.4   9.1   2.6  21.5    5   2      4
## 33    0.010     0.25 17.9   2.0  19.9  24.0  50.0    1   1      1
## 34   62.000  1320.00  6.1   1.9   8.0 100.0 267.0    1   1      1
## 35    0.122     3.00  8.2   2.4  10.6    NA  30.0    2   1      1
## 36    1.350     8.10  8.4   2.8  11.2    NA  45.0    3   1      3
## 37    0.023     0.40 11.9   1.3  13.2   3.2  19.0    4   1      3
## 38    0.048     0.33 10.8   2.0  12.8   2.0  30.0    4   1      3
## 39    1.700     6.30 13.8   5.6  19.4   5.0  12.0    2   1      1
## 40    3.500    10.80 14.3   3.1  17.4   6.5 120.0    2   1      1
## 42    0.480    15.50 15.2   1.8  17.0  12.0 140.0    2   2      2
## 43   10.000   115.00 10.0   0.9  10.9  20.2 170.0    4   4      4
## 44    1.620    11.40 11.9   1.8  13.7  13.0  17.0    2   1      2
```

```
## 45  192.000   180.00  6.5   1.9   8.4  27.0 115.0    4   4      4
## 46    2.500    12.10  7.5   0.9   8.4  18.0  31.0    5   5      5
## 48    0.280     1.90 10.6   2.6  13.2   4.7  21.0    3   1      3
## 49    4.235    50.40  7.4   2.4   9.8   9.8  52.0    1   1      1
## 50    6.800   179.00  8.4   1.2   9.6  29.0 164.0    2   3      2
## 51    0.750    12.30  5.7   0.9   6.6   7.0 225.0    2   2      2
## 52    3.600    21.00  4.9   0.5   5.4   6.0 225.0    3   2      3
## 54   55.500   175.00  3.2   0.6   3.8  20.0 151.0    5   5      5
## 56    0.060     1.00  8.1   2.2  10.3   3.5    NA    3   1      2
## 57    0.900     2.60 11.0   2.3  13.3   4.5  60.0    2   1      2
## 58    2.000    12.30  4.9   0.5   5.4   7.5 200.0    3   1      3
## 59    0.104     2.50 13.2   2.6  15.8   2.3  46.0    3   2      2
## 60    4.190    58.00  9.7   0.6  10.3  24.0 210.0    4   3      4
## 61    3.500     3.90 12.8   6.6  19.4   3.0  14.0    2   1      1
```

* 05. Subset the dataset to only include complete records

```
sleep[complete.cases(sleep),]  ## remember that complete.cases() returns indexes of rows
```

```
##     BodyWgt BrainWgt NonD Dream Sleep  Span  Gest Pred Exp Danger
## 2     1.000     6.60  6.3   2.0   8.3   4.5  42.0    3   1      3
## 5  2547.000  4603.00  2.1   1.8   3.9  69.0 624.0    3   5      4
## 6    10.550   179.50  9.1   0.7   9.8  27.0 180.0    4   4      4
## 7     0.023     0.30 15.8   3.9  19.7  19.0  35.0    1   1      1
## 8   160.000   169.00  5.2   1.0   6.2  30.4 392.0    4   5      4
## 9     3.300    25.60 10.9   3.6  14.5  28.0  63.0    1   2      1
## 10   52.160   440.00  8.3   1.4   9.7  50.0 230.0    1   1      1
## 11    0.425     6.40 11.0   1.5  12.5   7.0 112.0    5   4      4
## 12  465.000   423.00  3.2   0.7   3.9  30.0 281.0    5   5      5
## 15    0.075     1.20  6.3   2.1   8.4   3.5  42.0    1   1      1
## 16    3.000    25.00  8.6   0.0   8.6  50.0  28.0    2   2      2
## 17    0.785     3.50  6.6   4.1  10.7   6.0  42.0    2   2      2
## 18    0.200     5.00  9.5   1.2  10.7  10.4 120.0    2   2      2
## 22   27.660   115.00  3.3   0.5   3.8  20.0 148.0    5   5      5
## 23    0.120     1.00 11.0   3.4  14.4   3.9  16.0    3   1      2
## 25   85.000   325.00  4.7   1.5   6.2  41.0 310.0    1   3      1
## 27    0.101     4.00 10.4   3.4  13.8   9.0  28.0    5   1      3
## 28    1.040     5.50  7.4   0.8   8.2   7.6  68.0    5   3      4
## 29  521.000   655.00  2.1   0.8   2.9  46.0 336.0    5   5      5
## 32    0.005     0.14  7.7   1.4   9.1   2.6  21.5    5   2      4
## 33    0.010     0.25 17.9   2.0  19.9  24.0  50.0    1   1      1
## 34   62.000  1320.00  6.1   1.9   8.0 100.0 267.0    1   1      1
## 37    0.023     0.40 11.9   1.3  13.2   3.2  19.0    4   1      3
## 38    0.048     0.33 10.8   2.0  12.8   2.0  30.0    4   1      3
## 39    1.700     6.30 13.8   5.6  19.4   5.0  12.0    2   1      1
## 40    3.500    10.80 14.3   3.1  17.4   6.5 120.0    2   1      1
## 42    0.480    15.50 15.2   1.8  17.0  12.0 140.0    2   2      2
## 43   10.000   115.00 10.0   0.9  10.9  20.2 170.0    4   4      4
## 44    1.620    11.40 11.9   1.8  13.7  13.0  17.0    2   1      2
## 45  192.000   180.00  6.5   1.9   8.4  27.0 115.0    4   4      4
## 46    2.500    12.10  7.5   0.9   8.4  18.0  31.0    5   5      5
## 48    0.280     1.90 10.6   2.6  13.2   4.7  21.0    3   1      3
```

```
## 49    4.235    50.40  7.4   2.4   9.8   9.8  52.0    1   1       1
## 50    6.800   179.00  8.4   1.2   9.6  29.0 164.0    2   3       2
## 51    0.750    12.30  5.7   0.9   6.6   7.0 225.0    2   2       2
## 52    3.600    21.00  4.9   0.5   5.4   6.0 225.0    3   2       3
## 54   55.500   175.00  3.2   0.6   3.8  20.0 151.0    5   5       5
## 57    0.900     2.60 11.0   2.3  13.3   4.5  60.0    2   1       2
## 58    2.000    12.30  4.9   0.5   5.4   7.5 200.0    3   1       3
## 59    0.104     2.50 13.2   2.6  15.8   2.3  46.0    3   2       2
## 60    4.190    58.00  9.7   0.6  10.3  24.0 210.0    4   3       4
## 61    3.500     3.90 12.8   6.6  19.4   3.0  14.0    2   1       1
```

06. How heavy is the heaviest mammal in the dataset?

```
max(sleep[,'BodyWgt'])
```

```
## [1] 6654
```

* 07. What is the brain weight of the heaviest individual?

```
sleep[,'BrainWgt'][which.max(sleep[,'BodyWgt'])]
```

```
## [1] 5712
```

08. What are the weights of the animals that share the most common combination of Pred/Exp/Danger?

What is the mean weight of those animals? (This is hard. Don't worry if you don't get it.)

```
table(sleep[,c('Pred', 'Exp', 'Danger')]) ## this will give us a three way table
```

```
## , , Danger = 1
##
##     Exp
## Pred  1  2  3  4  5
##    1 10  2  1  1  0
##    2  4  0  0  0  0
##    3  1  0  0  0  0
##    4  0  0  0  0  0
##    5  0  0  0  0  0
##
## , , Danger = 2
##
##     Exp
## Pred  1  2  3  4  5
##    1  0  0  0  0  0
##    2  3  7  1  0  0
```

```
##    3  2  1  0  0  0
##    4  0  0  0  0  0
##    5  0  0  0  0  0
##
## , , Danger = 3
##
##     Exp
## Pred  1  2  3  4  5
##    1  0  0  0  0  0
##    2  0  0  0  0  0
##    3  4  1  0  0  1
##    4  2  0  0  0  0
##    5  1  1  0  0  0
##
## , , Danger = 4
##
##     Exp
## Pred  1  2  3  4  5
##    1  0  0  0  0  0
##    2  0  0  0  0  0
##    3  0  0  0  0  2
##    4  0  0  1  3  1
##    5  0  1  1  1  0
##
## , , Danger = 5
##
##     Exp
## Pred  1  2  3  4  5
##    1  0  0  0  0  0
##    2  0  0  0  0  0
##    3  0  0  0  0  0
##    4  0  0  0  0  0
##    5  0  0  0  0  9
```

```
max(table(sleep[,c('Pred', 'Exp', 'Danger')]))  ## this will tell us which is the max
```

```
## [1] 10
```

```
## look in the table above to see this value of 10 is observed when Danger==1, Pred==1, and Exp==1
sleep[,'BodyWgt'][sleep[,'Pred']==1 &   ## we can spread this command across lines when it gets unwieldy
                  sleep[,'Danger']==1 &
                  sleep[,'Exp']==1]
```

```
##  [1]   3.385   0.023  52.160   0.075  60.000  36.330 100.000   0.010
##  [9]  62.000   4.235
```

```
mean(sleep[,'BodyWgt'][sleep[,'Pred']==1 &
                  sleep[,'Danger']==1 &
                  sleep[,'Exp']==1]
)
```

```
## [1] 31.8218
```

This data came from a paper by Allison and Cicchetti, 1976, *Science*

For fun, these least preyed upon, least in danger, and least exposed when sleeping animals are:

- Arctic Fox

- Big brown bat
- Chimpanzee
- Eastern American mole
- Giant armadillo
- Gray wolf
- Jaguar
- Little brown bat
- Man
- Red fox