

# R\_Assignment - Group1

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Using the data provided by one of my professors in the analytics program. I do not have a link but only csv file.

This code works perfectly on my machine, as i have this file in a same folder but if it does not work on your end,you could simply import it.

```
Cereals <- read.csv("Cereals.csv", header=TRUE)
```

**Q1. Provide a introduction of your analysis in the .RMD file so it can be produced in the output.**

### Introduction to dataset:

##The following dataset contains the different nutritional values for various cereal brands. Using this data, we can find out that which cereals are best healthwise. Moreover, looking at different variables like sodium, fat, protein, carbohydrates, we can presumably recommend the best suited cereal brands to the customer according to their needs.

**Q2. Print the structure of your dataset.**

```
Cereals
```

```
##              name mfr type calories protein fat sodium
## 1      100%_Bran   N    C       70        4   1    130
## 2  100%_Natural_Bran Q    C      120        3   5     15
## 3      All-Bran    K    C       70        4   1    260
## 4 All-Bran_with_Extra_Fiber K    C       50        4   0    140
## 5      Almond_Delight R    C      110        2   2    200
## 6  Apple_Cinnamon_Cheerios G    C      110        2   2    180
## 7      Apple_Jacks   K    C      110        2   0    125
## 8      Basic_4      G    C      130        3   2    210
## 9      Bran_Chex    R    C       90        2   1    200
```

Q2. Print the structure of your dataset.

---

```
## 10          Bran_Flakes      P      C      90      3      0      210
## 11          Cap'n'Crunch    Q      C     120      1      2      220
## 12          Cheerios        G      C     110      6      2      290
## 13          Cinnamon_Toast_Crunch G      C     120      1      3      210
## 14          Clusters        G      C     110      3      2      140
## 15          Cocoa_Puffs     G      C     110      1      1      180
## 16          Corn_Chex       R      C     110      2      0      280
## 17          Corn_Flakes     K      C     100      2      0      290
## 18          Corn_Pops       K      C     110      1      0       90
## 19          Count_Chocula   G      C     110      1      1      180
## 20          Cracklin'_Oat_Bran K      C     110      3      3      140
## 21          Cream_of_Wheat_(Quick) N      H     100      3      0       80
## 22          Crispix         K      C     110      2      0      220
## 23          Crispy_Wheat_&_Raisins G      C     100      2      1      140
## 24          Double_Chex     R      C     100      2      0      190
## 25          Froot_Loops     K      C     110      2      1      125
## 26          Frosted_Flakes  K      C     110      1      0      200
## 27          Frosted_Mini-Wheats K      C     100      3      0         0
## 28 Fruit_&_Fibre_Dates,_Walnuts,_and_Oats P      C     120      3      2      160
## 29          Fruitful_Bran   K      C     120      3      0      240
## 30          Fruity_Pebbles   P      C     110      1      1      135
## 31          Golden_Crisp     P      C     100      2      0       45
## 32          Golden_Grahams   G      C     110      1      1      280
## 33          Grape_Nuts_Flakes P      C     100      3      1      140
## 34          Grape-Nuts      P      C     110      3      0      170
## 35          Great_Grains_Pecan P      C     120      3      3       75
## 36          Honey_Graham_Ohs Q      C     120      1      2      220
## 37          Honey_Nut_Cheerios G      C     110      3      1      250
## 38          Honey-comb      P      C     110      1      0      180
## 39          Just_Right_Crunchy__Nuggets K      C     110      2      1      170
## 40          Just_Right_Fruit_&_Nut K      C     140      3      1      170
## 41          Kix             G      C     110      2      1      260
## 42          Life            Q      C     100      4      2      150
## 43          Lucky_Charms     G      C     110      2      1      180
## 44          Maypo           A      H     100      4      1         0
## 45          Muesli_Raisins,_Dates,_&_Almonds R      C     150      4      3       95
## 46          Muesli_Raisins,_Peaches,_&_Pecans R      C     150      4      3      150
## 47          Mueslix_Crispy_Blend K      C     160      3      2      150
## 48          Multi-Grain_Cheerios G      C     100      2      1      220
## 49          Nut&Honey_Crunch K      C     120      2      1      190
## 50          Nutri-Grain_Almond-Raisin K      C     140      3      2      220
## 51          Nutri-grain_Wheat K      C      90      3      0      170
## 52          Oatmeal_Raisin_Crisp G      C     130      3      2      170
## 53          Post_Nat._Raisin_Bran P      C     120      3      1      200
## 54          Product_19      K      C     100      3      0      320
## 55          Puffed_Rice      Q      C      50      1      0         0
## 56          Puffed_Wheat     Q      C      50      2      0         0
## 57          Quaker_Oat_Squares Q      C     100      4      1      135
## 58          Quaker_Oatmeal   Q      H     100      5      2         0
## 59          Raisin_Bran      K      C     120      3      1      210
## 60          Raisin_Nut_Bran   G      C     100      3      2      140
## 61          Raisin_Squares    K      C      90      2      0         0
## 62          Rice_Chex        R      C     110      1      0      240
## 63          Rice_Krispies    K      C     110      2      0      290
```

Q2. Print the structure of your dataset.

---

```
## 64          Shredded_Wheat      N      C      80      2      0      0
## 65      Shredded_Wheat_ 'n' Bran      N      C      90      3      0      0
## 66      Shredded_Wheat_spoon_size      N      C      90      3      0      0
## 67          Smacks      K      C      110      2      1      70
## 68          Special_K      K      C      110      6      0      230
## 69      Strawberry_Fruit_Wheats      N      C      90      2      0      15
## 70          Total_Corn_Flakes      G      C      110      2      1      200
## 71          Total_Raisin_Bran      G      C      140      3      1      190
## 72          Total_Whole_Grain      G      C      100      3      1      200
## 73          Triples      G      C      110      2      1      250
## 74          Trix      G      C      110      1      1      140
## 75          Wheat_Chex      R      C      100      3      1      230
## 76          Wheaties      G      C      100      3      1      200
## 77      Wheaties_Honey_Gold      G      C      110      2      1      200
##      fiber carbo sugars potass vitamins shelf weight cups rating
## 1      10.0   5.0      6      280      25      3      1.00 0.33 68.40297
## 2       2.0   8.0      8      135       0      3      1.00 1.00 33.98368
## 3       9.0   7.0      5      320      25      3      1.00 0.33 59.42551
## 4      14.0   8.0      0      330      25      3      1.00 0.50 93.70491
## 5       1.0  14.0      8      NA      25      3      1.00 0.75 34.38484
## 6       1.5  10.5     10       70      25      1      1.00 0.75 29.50954
## 7       1.0  11.0     14       30      25      2      1.00 1.00 33.17409
## 8       2.0  18.0      8      100      25      3      1.33 0.75 37.03856
## 9       4.0  15.0      6      125      25      1      1.00 0.67 49.12025
## 10      5.0  13.0      5      190      25      3      1.00 0.67 53.31381
## 11      0.0  12.0     12       35      25      2      1.00 0.75 18.04285
## 12      2.0  17.0      1      105      25      1      1.00 1.25 50.76500
## 13      0.0  13.0      9       45      25      2      1.00 0.75 19.82357
## 14      2.0  13.0      7      105      25      3      1.00 0.50 40.40021
## 15      0.0  12.0     13       55      25      2      1.00 1.00 22.73645
## 16      0.0  22.0      3       25      25      1      1.00 1.00 41.44502
## 17      1.0  21.0      2       35      25      1      1.00 1.00 45.86332
## 18      1.0  13.0     12       20      25      2      1.00 1.00 35.78279
## 19      0.0  12.0     13       65      25      2      1.00 1.00 22.39651
## 20      4.0  10.0      7      160      25      3      1.00 0.50 40.44877
## 21      1.0  21.0      0      NA       0      2      1.00 1.00 64.53382
## 22      1.0  21.0      3       30      25      3      1.00 1.00 46.89564
## 23      2.0  11.0     10      120      25      3      1.00 0.75 36.17620
## 24      1.0  18.0      5       80      25      3      1.00 0.75 44.33086
## 25      1.0  11.0     13       30      25      2      1.00 1.00 32.20758
## 26      1.0  14.0     11       25      25      1      1.00 0.75 31.43597
## 27      3.0  14.0      7      100      25      2      1.00 0.80 58.34514
## 28      5.0  12.0     10      200      25      3      1.25 0.67 40.91705
## 29      5.0  14.0     12      190      25      3      1.33 0.67 41.01549
## 30      0.0  13.0     12       25      25      2      1.00 0.75 28.02576
## 31      0.0  11.0     15       40      25      1      1.00 0.88 35.25244
## 32      0.0  15.0      9       45      25      2      1.00 0.75 23.80404
## 33      3.0  15.0      5       85      25      3      1.00 0.88 52.07690
## 34      3.0  17.0      3       90      25      3      1.00 0.25 53.37101
## 35      3.0  13.0      4      100      25      3      1.00 0.33 45.81172
## 36      1.0  12.0     11       45      25      2      1.00 1.00 21.87129
## 37      1.5  11.5     10       90      25      1      1.00 0.75 31.07222
## 38      0.0  14.0     11       35      25      1      1.00 1.33 28.74241
## 39      1.0  17.0      6       60     100      3      1.00 1.00 36.52368
```

Q3. List the variables in your dataset.

---

```
## 40  2.0 20.0      9    95    100    3  1.30 0.75 36.47151
## 41  0.0 21.0      3    40     25    2  1.00 1.50 39.24111
## 42  2.0 12.0      6    95     25    2  1.00 0.67 45.32807
## 43  0.0 12.0     12    55     25    2  1.00 1.00 26.73451
## 44  0.0 16.0      3    95     25    2  1.00 1.00 54.85092
## 45  3.0 16.0     11   170     25    3  1.00 1.00 37.13686
## 46  3.0 16.0     11   170     25    3  1.00 1.00 34.13976
## 47  3.0 17.0     13   160     25    3  1.50 0.67 30.31335
## 48  2.0 15.0      6    90     25    1  1.00 1.00 40.10596
## 49  0.0 15.0      9    40     25    2  1.00 0.67 29.92429
## 50  3.0 21.0      7   130     25    3  1.33 0.67 40.69232
## 51  3.0 18.0      2    90     25    3  1.00 1.00 59.64284
## 52  1.5 13.5     10   120     25    3  1.25 0.50 30.45084
## 53  6.0 11.0     14   260     25    3  1.33 0.67 37.84059
## 54  1.0 20.0      3    45    100    3  1.00 1.00 41.50354
## 55  0.0 13.0      0    15      0    3  0.50 1.00 60.75611
## 56  1.0 10.0      0    50      0    3  0.50 1.00 63.00565
## 57  2.0 14.0      6   110     25    3  1.00 0.50 49.51187
## 58  2.7  NA     NA   110      0    1  1.00 0.67 50.82839
## 59  5.0 14.0     12   240     25    2  1.33 0.75 39.25920
## 60  2.5 10.5      8   140     25    3  1.00 0.50 39.70340
## 61  2.0 15.0      6   110     25    3  1.00 0.50 55.33314
## 62  0.0 23.0      2    30     25    1  1.00 1.13 41.99893
## 63  0.0 22.0      3    35     25    1  1.00 1.00 40.56016
## 64  3.0 16.0      0    95      0    1  0.83 1.00 68.23588
## 65  4.0 19.0      0   140      0    1  1.00 0.67 74.47295
## 66  3.0 20.0      0   120      0    1  1.00 0.67 72.80179
## 67  1.0  9.0     15    40     25    2  1.00 0.75 31.23005
## 68  1.0 16.0      3    55     25    1  1.00 1.00 53.13132
## 69  3.0 15.0      5    90     25    2  1.00 1.00 59.36399
## 70  0.0 21.0      3    35    100    3  1.00 1.00 38.83975
## 71  4.0 15.0     14   230    100    3  1.50 1.00 28.59278
## 72  3.0 16.0      3   110    100    3  1.00 1.00 46.65884
## 73  0.0 21.0      3    60     25    3  1.00 0.75 39.10617
## 74  0.0 13.0     12    25     25    2  1.00 1.00 27.75330
## 75  3.0 17.0      3   115     25    1  1.00 0.67 49.78744
## 76  3.0 17.0      3   110     25    1  1.00 1.00 51.59219
## 77  1.0 16.0      8    60     25    1  1.00 0.75 36.18756
```

Q3. List the variables in your dataset.

```
names(Cereals)
```

```
## [1] "name"      "mfr"      "type"      "calories" "protein"  "fat"
## [7] "sodium"    "fiber"    "carbo"     "sugars"   "potass"   "vitamins"
## [13] "shelf"     "weight"   "cups"      "rating"
```

Q4. Print the top 15 rows of your dataset.

Q5. Write a user defined function using any of the variables from the data set.

```
head(Cereals,n=15)
```

```
##           name mfr type calories protein fat sodium fiber carbo
## 1      100%_Bran  N   C       70      4  1   130  10.0  5.0
## 2    100%_Natural_Bran  Q   C      120      3  5    15   2.0  8.0
## 3         All-Bran  K   C       70      4  1   260   9.0  7.0
## 4 All-Bran_with_Extra_Fiber  K   C       50      4  0   140  14.0  8.0
## 5        Almond_Delight  R   C      110      2  2   200   1.0 14.0
## 6   Apple_Cinnamon_Cheerios  G   C      110      2  2   180   1.5 10.5
## 7         Apple_Jacks  K   C      110      2  0   125   1.0 11.0
## 8           Basic_4  G   C      130      3  2   210   2.0 18.0
## 9          Bran_Chex  R   C       90      2  1   200   4.0 15.0
## 10         Bran_Flakes  P   C       90      3  0   210   5.0 13.0
## 11         Cap'n'Crunch  Q   C      120      1  2   220   0.0 12.0
## 12          Cheerios  G   C      110      6  2   290   2.0 17.0
## 13   Cinnamon_Toast_Crunch  G   C      120      1  3   210   0.0 13.0
## 14          Clusters  G   C      110      3  2   140   2.0 13.0
## 15        Cocoa_Puffs  G   C      110      1  1   180   0.0 12.0
##      sugars potass vitamins shelf weight cups  rating
## 1         6    280      25     3    1.00 0.33 68.40297
## 2         8    135       0     3    1.00 1.00 33.98368
## 3         5    320      25     3    1.00 0.33 59.42551
## 4         0    330      25     3    1.00 0.50 93.70491
## 5         8     NA      25     3    1.00 0.75 34.38484
## 6        10     70      25     1    1.00 0.75 29.50954
## 7        14     30      25     2    1.00 1.00 33.17409
## 8         8    100      25     3    1.33 0.75 37.03856
## 9         6    125      25     1    1.00 0.67 49.12025
## 10        5    190      25     3    1.00 0.67 53.31381
## 11       12     35      25     2    1.00 0.75 18.04285
## 12        1    105      25     1    1.00 1.25 50.76500
## 13        9     45      25     2    1.00 0.75 19.82357
## 14        7    105      25     3    1.00 0.50 40.40021
## 15       13     55      25     2    1.00 1.00 22.73645
```

Q5. Write a user defined function using any of the variables from the data set.

Here we are using sodium variable from our dataset, and doing it square.

```
sodium_square<-function(x)
{
  result=x*x
  result
}
sodium_square(Cereals$sodium)
```

```
## [1] 16900    225  67600 19600 40000 32400 15625 44100 40000 44100
## [11] 48400 84100 44100 19600 32400 78400 84100  8100 32400 19600
## [21]  6400 48400 19600 36100 15625 40000      0 25600 57600 18225
## [31]  2025 78400 19600 28900  5625 48400 62500 32400 28900 28900
```

Q6. Use data manipulation techniques and filter rows based on any logical criteria that exist in your dataset.

```
## [41] 67600 22500 32400      0  9025 22500 22500 48400 36100 48400
## [51] 28900 28900 40000 102400      0      0 18225      0 44100 19600
## [61]      0 57600 84100      0      0      0  4900 52900      225 40000
## [71] 36100 40000 62500 19600 52900 40000 40000
```

Q6. Use data manipulation techniques and filter rows based on any logical criteria that exist in your dataset.

Here we are filtering our data, we are just considering rows which has sodium more than 130.

```
library(dplyr)
sample1 = filter(Cereals, sodium > 100)
sample1
```

```
##               name mfr type calories protein fat sodium
## 1          100%_Bran   N   C        70         4   1   130
## 2           All-Bran   K   C        70         4   1   260
## 3 All-Bran_with_Extra_Fiber K   C        50         4   0   140
## 4       Almond_Delight   R   C       110         2   2   200
## 5   Apple_Cinnamon_Cheerios G   C       110         2   2   180
## 6         Apple_Jacks   K   C       110         2   0   125
## 7           Basic_4   G   C       130         3   2   210
## 8          Bran_Chex   R   C        90         2   1   200
## 9          Bran_Flakes   P   C        90         3   0   210
## 10         Cap'n'_Crunch   Q   C       120         1   2   220
## 11           Cheerios   G   C       110         6   2   290
## 12   Cinnamon_Toast_Crunch G   C       120         1   3   210
## 13           Clusters   G   C       110         3   2   140
## 14         Cocoa_Puffs   G   C       110         1   1   180
## 15          Corn_Chex   R   C       110         2   0   280
## 16         Corn_Flakes   K   C       100         2   0   290
## 17        Count_Chocula   G   C       110         1   1   180
## 18   Cracklin'_Oat_Bran   K   C       110         3   3   140
## 19          Crispix   K   C       110         2   0   220
## 20   Crispy_Wheat_&_Raisins G   C       100         2   1   140
## 21         Double_Chex   R   C       100         2   0   190
## 22         Froot_Loops   K   C       110         2   1   125
## 23       Frosted_Flakes   K   C       110         1   0   200
## 24 Fruit_&_Fibre_Dates,_Walnuts,_and_Oats P   C       120         3   2   160
## 25         Fruitful_Bran   K   C       120         3   0   240
## 26         Fruity_Pebbles   P   C       110         1   1   135
## 27         Golden_Grahams   G   C       110         1   1   280
## 28         Grape_Nuts_Flakes P   C       100         3   1   140
## 29           Grape-Nuts   P   C       110         3   0   170
## 30         Honey_Graham_Ohs Q   C       120         1   2   220
## 31         Honey_Nut_Cheerios G   C       110         3   1   250
## 32           Honey-comb   P   C       110         1   0   180
## 33   Just_Right_Crunchy__Nuggets K   C       110         2   1   170
## 34         Just_Right_Fruit_&_Nut K   C       140         3   1   170
## 35              Kix   G   C       110         2   1   260
```

Here we are filtering our data, we are just considering rows which has sodium more than 130.

## 36	Life	Q	C	100	4	2	150		
## 37	Lucky_Charm	G	C	110	2	1	180		
## 38	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3	150		
## 39	Mueslix_Crispy_Blend	K	C	160	3	2	150		
## 40	Multi-Grain_Cheerios	G	C	100	2	1	220		
## 41	Nut&Honey_Crunch	K	C	120	2	1	190		
## 42	Nutri-Grain_Almond-Raisin	K	C	140	3	2	220		
## 43	Nutri-grain_Wheat	K	C	90	3	0	170		
## 44	Oatmeal_Raisin_Crisp	G	C	130	3	2	170		
## 45	Post_Nat._Raisin_Bran	P	C	120	3	1	200		
## 46	Product_19	K	C	100	3	0	320		
## 47	Quaker_Oat_Squares	Q	C	100	4	1	135		
## 48	Raisin_Bran	K	C	120	3	1	210		
## 49	Raisin_Nut_Bran	G	C	100	3	2	140		
## 50	Rice_Chex	R	C	110	1	0	240		
## 51	Rice_Krispies	K	C	110	2	0	290		
## 52	Special_K	K	C	110	6	0	230		
## 53	Total_Corn_Flakes	G	C	110	2	1	200		
## 54	Total_Raisin_Bran	G	C	140	3	1	190		
## 55	Total_Whole_Grain	G	C	100	3	1	200		
## 56	Triples	G	C	110	2	1	250		
## 57	Trix	G	C	110	1	1	140		
## 58	Wheat_Chex	R	C	100	3	1	230		
## 59	Wheaties	G	C	100	3	1	200		
## 60	Wheaties_Honey_Gold	G	C	110	2	1	200		
##	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups	rating
## 1	10.0	5.0	6	280	25	3	1.00	0.33	68.40297
## 2	9.0	7.0	5	320	25	3	1.00	0.33	59.42551
## 3	14.0	8.0	0	330	25	3	1.00	0.50	93.70491
## 4	1.0	14.0	8	NA	25	3	1.00	0.75	34.38484
## 5	1.5	10.5	10	70	25	1	1.00	0.75	29.50954
## 6	1.0	11.0	14	30	25	2	1.00	1.00	33.17409
## 7	2.0	18.0	8	100	25	3	1.33	0.75	37.03856
## 8	4.0	15.0	6	125	25	1	1.00	0.67	49.12025
## 9	5.0	13.0	5	190	25	3	1.00	0.67	53.31381
## 10	0.0	12.0	12	35	25	2	1.00	0.75	18.04285
## 11	2.0	17.0	1	105	25	1	1.00	1.25	50.76500
## 12	0.0	13.0	9	45	25	2	1.00	0.75	19.82357
## 13	2.0	13.0	7	105	25	3	1.00	0.50	40.40021
## 14	0.0	12.0	13	55	25	2	1.00	1.00	22.73645
## 15	0.0	22.0	3	25	25	1	1.00	1.00	41.44502
## 16	1.0	21.0	2	35	25	1	1.00	1.00	45.86332
## 17	0.0	12.0	13	65	25	2	1.00	1.00	22.39651
## 18	4.0	10.0	7	160	25	3	1.00	0.50	40.44877
## 19	1.0	21.0	3	30	25	3	1.00	1.00	46.89564
## 20	2.0	11.0	10	120	25	3	1.00	0.75	36.17620
## 21	1.0	18.0	5	80	25	3	1.00	0.75	44.33086
## 22	1.0	11.0	13	30	25	2	1.00	1.00	32.20758
## 23	1.0	14.0	11	25	25	1	1.00	0.75	31.43597
## 24	5.0	12.0	10	200	25	3	1.25	0.67	40.91705
## 25	5.0	14.0	12	190	25	3	1.33	0.67	41.01549
## 26	0.0	13.0	12	25	25	2	1.00	0.75	28.02576
## 27	0.0	15.0	9	45	25	2	1.00	0.75	23.80404
## 28	3.0	15.0	5	85	25	3	1.00	0.88	52.07690

Q7. Identify the dependent & independent variables and use reshaping techniques and create a new data frame by joining those variables from your dataset

---

```
## 29  3.0  17.0    3    90    25    3  1.00 0.25 53.37101
## 30  1.0  12.0   11    45    25    2  1.00 1.00 21.87129
## 31  1.5  11.5   10    90    25    1  1.00 0.75 31.07222
## 32  0.0  14.0   11    35    25    1  1.00 1.33 28.74241
## 33  1.0  17.0    6    60   100    3  1.00 1.00 36.52368
## 34  2.0  20.0    9    95   100    3  1.30 0.75 36.47151
## 35  0.0  21.0    3    40    25    2  1.00 1.50 39.24111
## 36  2.0  12.0    6    95    25    2  1.00 0.67 45.32807
## 37  0.0  12.0   12    55    25    2  1.00 1.00 26.73451
## 38  3.0  16.0   11   170    25    3  1.00 1.00 34.13976
## 39  3.0  17.0   13   160    25    3  1.50 0.67 30.31335
## 40  2.0  15.0    6    90    25    1  1.00 1.00 40.10596
## 41  0.0  15.0    9    40    25    2  1.00 0.67 29.92429
## 42  3.0  21.0    7   130    25    3  1.33 0.67 40.69232
## 43  3.0  18.0    2    90    25    3  1.00 1.00 59.64284
## 44  1.5  13.5   10   120    25    3  1.25 0.50 30.45084
## 45  6.0  11.0   14   260    25    3  1.33 0.67 37.84059
## 46  1.0  20.0    3    45   100    3  1.00 1.00 41.50354
## 47  2.0  14.0    6   110    25    3  1.00 0.50 49.51187
## 48  5.0  14.0   12   240    25    2  1.33 0.75 39.25920
## 49  2.5  10.5    8   140    25    3  1.00 0.50 39.70340
## 50  0.0  23.0    2    30    25    1  1.00 1.13 41.99893
## 51  0.0  22.0    3    35    25    1  1.00 1.00 40.56016
## 52  1.0  16.0    3    55    25    1  1.00 1.00 53.13132
## 53  0.0  21.0    3    35   100    3  1.00 1.00 38.83975
## 54  4.0  15.0   14   230   100    3  1.50 1.00 28.59278
## 55  3.0  16.0    3   110   100    3  1.00 1.00 46.65884
## 56  0.0  21.0    3    60    25    3  1.00 0.75 39.10617
## 57  0.0  13.0   12    25    25    2  1.00 1.00 27.75330
## 58  3.0  17.0    3   115    25    1  1.00 0.67 49.78744
## 59  3.0  17.0    3   110    25    1  1.00 1.00 51.59219
## 60  1.0  16.0    8    60    25    1  1.00 0.75 36.18756
```

Q7. Identify the dependent & independent variables and use reshaping techniques and create a new data frame by joining those variables from your dataset

Looking at data, rating (variable) is dependent on all the other variables and all the variables are independent to each other.

Here, we created a new dataframe with total 6 columns.

```
df = data.frame(Cereals %>% select(1:6))
df
```

```
##              name mfr type calories protein fat
## 1          100%_Bran   N   C         70         4  1
## 2    100%_Natural_Bran   Q   C        120         3  5
## 3           All-Bran   K   C         70         4  1
## 4 All-Bran_with_Extra_Fiber K   C         50         4  0
## 5         Almond_Delight   R   C        110         2  2
## 6   Apple_Cinnamon_Cheerios G   C        110         2  2
```



Here, we created a new dataframe with total 6 columns.

---

## 7	Apple_Jacks	K	C	110	2	0
## 8	Basic_4	G	C	130	3	2
## 9	Bran_Chex	R	C	90	2	1
## 10	Bran_Flakes	P	C	90	3	0
## 11	Cap'n'Crunch	Q	C	120	1	2
## 12	Cheerios	G	C	110	6	2
## 13	Cinnamon_Toast_Crunch	G	C	120	1	3
## 14	Clusters	G	C	110	3	2
## 15	Cocoa_Puffs	G	C	110	1	1
## 16	Corn_Chex	R	C	110	2	0
## 17	Corn_Flakes	K	C	100	2	0
## 18	Corn_Pops	K	C	110	1	0
## 19	Count_Chocula	G	C	110	1	1
## 20	Cracklin'_Oat_Bran	K	C	110	3	3
## 21	Cream_of_Wheat_(Quick)	N	H	100	3	0
## 22	Crispix	K	C	110	2	0
## 23	Crispy_Wheat_&_Raisins	G	C	100	2	1
## 24	Double_Chex	R	C	100	2	0
## 25	Froot_Loops	K	C	110	2	1
## 26	Frosted_Flakes	K	C	110	1	0
## 27	Frosted_Mini-Wheats	K	C	100	3	0
## 28	Fruit_&_Fibre_Dates,_Walnuts,_and_Oats	P	C	120	3	2
## 29	Fruitful_Bran	K	C	120	3	0
## 30	Fruity_Pebbles	P	C	110	1	1
## 31	Golden_Crisp	P	C	100	2	0
## 32	Golden_Grahams	G	C	110	1	1
## 33	Grape_Nuts_Flakes	P	C	100	3	1
## 34	Grape-Nuts	P	C	110	3	0
## 35	Great_Grains_Pecan	P	C	120	3	3
## 36	Honey_Graham_Ohs	Q	C	120	1	2
## 37	Honey_Nut_Cheerios	G	C	110	3	1
## 38	Honey-comb	P	C	110	1	0
## 39	Just_Right_Crunchy__Nuggets	K	C	110	2	1
## 40	Just_Right_Fruit_&_Nut	K	C	140	3	1
## 41	Kix	G	C	110	2	1
## 42	Life	Q	C	100	4	2
## 43	Lucky_Charms	G	C	110	2	1
## 44	Maypo	A	H	100	4	1
## 45	Muesli_Raisins,_Dates,_&_Almonds	R	C	150	4	3
## 46	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3
## 47	Mueslix_Crispy_Blend	K	C	160	3	2
## 48	Multi-Grain_Cheerios	G	C	100	2	1
## 49	Nut&Honey_Crunch	K	C	120	2	1
## 50	Nutri-Grain_Almond-Raisin	K	C	140	3	2
## 51	Nutri-grain_Wheat	K	C	90	3	0
## 52	Oatmeal_Raisin_Crisp	G	C	130	3	2
## 53	Post_Nat._Raisin_Bran	P	C	120	3	1
## 54	Product_19	K	C	100	3	0
## 55	Puffed_Rice	Q	C	50	1	0
## 56	Puffed_Wheat	Q	C	50	2	0
## 57	Quaker_Oat_Squares	Q	C	100	4	1
## 58	Quaker_Oatmeal	Q	H	100	5	2
## 59	Raisin_Bran	K	C	120	3	1
## 60	Raisin_Nut_Bran	G	C	100	3	2

Q8.Remove missing values in your dataset.

---

```
## 61          Raisin_Squares K C 90 2 0
## 62          Rice_Chex R C 110 1 0
## 63          Rice_Krispies K C 110 2 0
## 64          Shredded_Wheat N C 80 2 0
## 65          Shredded_Wheat_ 'n' Bran N C 90 3 0
## 66          Shredded_Wheat_spoon_size N C 90 3 0
## 67          Smacks K C 110 2 1
## 68          Special_K K C 110 6 0
## 69          Strawberry_Fruit_Wheats N C 90 2 0
## 70          Total_Corn_Flakes G C 110 2 1
## 71          Total_Raisin_Bran G C 140 3 1
## 72          Total_Whole_Grain G C 100 3 1
## 73          Triples G C 110 2 1
## 74          Trix G C 110 1 1
## 75          Wheat_Chex R C 100 3 1
## 76          Wheaties G C 100 3 1
## 77          Wheaties_Honey_Gold G C 110 2 1
```

Q8.Remove missing values in your dataset.

```
na_vec<-which(!complete.cases(Cereals))
Cereals <- Cereals[-na_vec,]
Cereals
```

```
##          name mfr type calories protein fat sodium
## 1          100%_Bran N C 70 4 1 130
## 2          100%_Natural_Bran Q C 120 3 5 15
## 3          All-Bran K C 70 4 1 260
## 4          All-Bran_with_Extra_Fiber K C 50 4 0 140
## 6          Apple_Cinnamon_Cheerios G C 110 2 2 180
## 7          Apple_Jacks K C 110 2 0 125
## 8          Basic_4 G C 130 3 2 210
## 9          Bran_Chex R C 90 2 1 200
## 10          Bran_Flakes P C 90 3 0 210
## 11          Cap'n'Crunch Q C 120 1 2 220
## 12          Cheerios G C 110 6 2 290
## 13          Cinnamon_Toast_Crunch G C 120 1 3 210
## 14          Clusters G C 110 3 2 140
## 15          Cocoa_Puffs G C 110 1 1 180
## 16          Corn_Chex R C 110 2 0 280
## 17          Corn_Flakes K C 100 2 0 290
## 18          Corn_Pops K C 110 1 0 90
## 19          Count_Chocula G C 110 1 1 180
## 20          Cracklin' _Oat_Bran K C 110 3 3 140
## 22          Crispix K C 110 2 0 220
## 23          Crispy_Wheat_&_Raisins G C 100 2 1 140
## 24          Double_Chex R C 100 2 0 190
## 25          Froot_Loops K C 110 2 1 125
## 26          Frosted_Flakes K C 110 1 0 200
## 27          Frosted_Mini-Wheats K C 100 3 0 0
## 28 Fruit_&_Fibre_Dates,_Walnuts,_and_Oats P C 120 3 2 160
## 29          Fruitful_Bran K C 120 3 0 240
```

Q8.Remove missing values in your dataset.

## 30	Fruity_Pebbles	P	C	110	1	1	135
## 31	Golden_Crisp	P	C	100	2	0	45
## 32	Golden_Grahams	G	C	110	1	1	280
## 33	Grape_Nuts_Flakes	P	C	100	3	1	140
## 34	Grape-Nuts	P	C	110	3	0	170
## 35	Great_Grains_Pecan	P	C	120	3	3	75
## 36	Honey_Graham_Ohs	Q	C	120	1	2	220
## 37	Honey_Nut_Cheerios	G	C	110	3	1	250
## 38	Honey-comb	P	C	110	1	0	180
## 39	Just_Right_Crunchy__Nuggets	K	C	110	2	1	170
## 40	Just_Right_Fruit_&_Nut	K	C	140	3	1	170
## 41	Kix	G	C	110	2	1	260
## 42	Life	Q	C	100	4	2	150
## 43	Lucky_Charms	G	C	110	2	1	180
## 44	Maypo	A	H	100	4	1	0
## 45	Muesli_Raisins,_Dates,_&_Almonds	R	C	150	4	3	95
## 46	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3	150
## 47	Mueslix_Crispy_Blend	K	C	160	3	2	150
## 48	Multi-Grain_Cheerios	G	C	100	2	1	220
## 49	Nut&Honey_Crunch	K	C	120	2	1	190
## 50	Nutri-Grain_Almond-Raisin	K	C	140	3	2	220
## 51	Nutri-grain_Wheat	K	C	90	3	0	170
## 52	Oatmeal_Raisin_Crisp	G	C	130	3	2	170
## 53	Post_Nat._Raisin_Bran	P	C	120	3	1	200
## 54	Product_19	K	C	100	3	0	320
## 55	Puffed_Rice	Q	C	50	1	0	0
## 56	Puffed_Wheat	Q	C	50	2	0	0
## 57	Quaker_Oat_Squares	Q	C	100	4	1	135
## 59	Raisin_Bran	K	C	120	3	1	210
## 60	Raisin_Nut_Bran	G	C	100	3	2	140
## 61	Raisin_Squares	K	C	90	2	0	0
## 62	Rice_Chex	R	C	110	1	0	240
## 63	Rice_Krispies	K	C	110	2	0	290
## 64	Shredded_Wheat	N	C	80	2	0	0
## 65	Shredded_Wheat_'n'Bran	N	C	90	3	0	0
## 66	Shredded_Wheat_spoon_size	N	C	90	3	0	0
## 67	Smacks	K	C	110	2	1	70
## 68	Special_K	K	C	110	6	0	230
## 69	Strawberry_Fruit_Wheats	N	C	90	2	0	15
## 70	Total_Corn_Flakes	G	C	110	2	1	200
## 71	Total_Raisin_Bran	G	C	140	3	1	190
## 72	Total_Whole_Grain	G	C	100	3	1	200
## 73	Triples	G	C	110	2	1	250
## 74	Trix	G	C	110	1	1	140
## 75	Wheat_Chex	R	C	100	3	1	230
## 76	Wheaties	G	C	100	3	1	200
## 77	Wheaties_Honey_Gold	G	C	110	2	1	200
##	fiber carbo sugars potass vitamins shelf weight cups rating						
## 1	10.0	5.0	6	280	25	3	1.00 0.33 68.40297
## 2	2.0	8.0	8	135	0	3	1.00 1.00 33.98368
## 3	9.0	7.0	5	320	25	3	1.00 0.33 59.42551
## 4	14.0	8.0	0	330	25	3	1.00 0.50 93.70491
## 6	1.5	10.5	10	70	25	1	1.00 0.75 29.50954
## 7	1.0	11.0	14	30	25	2	1.00 1.00 33.17409

```
## 8    2.0  18.0      8   100      25    3    1.33 0.75 37.03856
## 9    4.0  15.0      6   125      25    1    1.00 0.67 49.12025
## 10   5.0  13.0      5   190      25    3    1.00 0.67 53.31381
## 11   0.0  12.0     12    35      25    2    1.00 0.75 18.04285
## 12   2.0  17.0      1   105      25    1    1.00 1.25 50.76500
## 13   0.0  13.0      9    45      25    2    1.00 0.75 19.82357
## 14   2.0  13.0      7   105      25    3    1.00 0.50 40.40021
## 15   0.0  12.0     13    55      25    2    1.00 1.00 22.73645
## 16   0.0  22.0      3    25      25    1    1.00 1.00 41.44502
## 17   1.0  21.0      2    35      25    1    1.00 1.00 45.86332
## 18   1.0  13.0     12    20      25    2    1.00 1.00 35.78279
## 19   0.0  12.0     13    65      25    2    1.00 1.00 22.39651
## 20   4.0  10.0      7   160      25    3    1.00 0.50 40.44877
## 22   1.0  21.0      3    30      25    3    1.00 1.00 46.89564
## 23   2.0  11.0     10   120      25    3    1.00 0.75 36.17620
## 24   1.0  18.0      5    80      25    3    1.00 0.75 44.33086
## 25   1.0  11.0     13    30      25    2    1.00 1.00 32.20758
## 26   1.0  14.0     11    25      25    1    1.00 0.75 31.43597
## 27   3.0  14.0      7   100      25    2    1.00 0.80 58.34514
## 28   5.0  12.0     10   200      25    3    1.25 0.67 40.91705
## 29   5.0  14.0     12   190      25    3    1.33 0.67 41.01549
## 30   0.0  13.0     12    25      25    2    1.00 0.75 28.02576
## 31   0.0  11.0     15    40      25    1    1.00 0.88 35.25244
## 32   0.0  15.0      9    45      25    2    1.00 0.75 23.80404
## 33   3.0  15.0      5    85      25    3    1.00 0.88 52.07690
## 34   3.0  17.0      3    90      25    3    1.00 0.25 53.37101
## 35   3.0  13.0      4   100      25    3    1.00 0.33 45.81172
## 36   1.0  12.0     11    45      25    2    1.00 1.00 21.87129
## 37   1.5  11.5     10    90      25    1    1.00 0.75 31.07222
## 38   0.0  14.0     11    35      25    1    1.00 1.33 28.74241
## 39   1.0  17.0      6    60     100    3    1.00 1.00 36.52368
## 40   2.0  20.0      9    95     100    3    1.30 0.75 36.47151
## 41   0.0  21.0      3    40      25    2    1.00 1.50 39.24111
## 42   2.0  12.0      6    95      25    2    1.00 0.67 45.32807
## 43   0.0  12.0     12    55      25    2    1.00 1.00 26.73451
## 44   0.0  16.0      3    95      25    2    1.00 1.00 54.85092
## 45   3.0  16.0     11   170      25    3    1.00 1.00 37.13686
## 46   3.0  16.0     11   170      25    3    1.00 1.00 34.13976
## 47   3.0  17.0     13   160      25    3    1.50 0.67 30.31335
## 48   2.0  15.0      6    90      25    1    1.00 1.00 40.10596
## 49   0.0  15.0      9    40      25    2    1.00 0.67 29.92429
## 50   3.0  21.0      7   130      25    3    1.33 0.67 40.69232
## 51   3.0  18.0      2    90      25    3    1.00 1.00 59.64284
## 52   1.5  13.5     10   120      25    3    1.25 0.50 30.45084
## 53   6.0  11.0     14   260      25    3    1.33 0.67 37.84059
## 54   1.0  20.0      3    45     100    3    1.00 1.00 41.50354
## 55   0.0  13.0      0    15      0     3    0.50 1.00 60.75611
## 56   1.0  10.0      0    50      0     3    0.50 1.00 63.00565
## 57   2.0  14.0      6   110      25    3    1.00 0.50 49.51187
## 59   5.0  14.0     12   240      25    2    1.33 0.75 39.25920
## 60   2.5  10.5      8   140      25    3    1.00 0.50 39.70340
## 61   2.0  15.0      6   110      25    3    1.00 0.50 55.33314
## 62   0.0  23.0      2    30      25    1    1.00 1.13 41.99893
## 63   0.0  22.0      3    35      25    1    1.00 1.00 40.56016
```

You can see from the data that now we have 3 less rows in our dataset after removing N/A values.

```
## 64  3.0  16.0    0   95    0    1  0.83 1.00 68.23588
## 65  4.0  19.0    0  140    0    1  1.00 0.67 74.47295
## 66  3.0  20.0    0  120    0    1  1.00 0.67 72.80179
## 67  1.0   9.0   15   40   25    2  1.00 0.75 31.23005
## 68  1.0  16.0    3   55   25    1  1.00 1.00 53.13132
## 69  3.0  15.0    5   90   25    2  1.00 1.00 59.36399
## 70  0.0  21.0    3   35  100    3  1.00 1.00 38.83975
## 71  4.0  15.0   14  230  100    3  1.50 1.00 28.59278
## 72  3.0  16.0    3  110  100    3  1.00 1.00 46.65884
## 73  0.0  21.0    3   60   25    3  1.00 0.75 39.10617
## 74  0.0  13.0   12   25   25    2  1.00 1.00 27.75330
## 75  3.0  17.0    3  115   25    1  1.00 0.67 49.78744
## 76  3.0  17.0    3  110   25    1  1.00 1.00 51.59219
## 77  1.0  16.0    8   60   25    1  1.00 0.75 36.18756
```

You can see from the data that now we have 3 less rows in our dataset after removing N/A values.

**Q9. Identify and remove duplicated data in your dataset.**

```
Cereals<-Cereals[!duplicated(Cereals),]
Cereals
```

```
##              name mfr type calories protein fat sodium
## 1          100%_Bran  N   C        70         4   1   130
## 2      100%_Natural_Bran  Q   C       120         3   5    15
## 3             All-Bran  K   C        70         4   1   260
## 4 All-Bran_with_Extra_Fiber  K   C        50         4   0   140
## 6      Apple_Cinnamon_Cheerios  G   C       110         2   2   180
## 7             Apple_Jacks  K   C       110         2   0   125
## 8              Basic_4  G   C       130         3   2   210
## 9              Bran_Chex  R   C        90         2   1   200
## 10             Bran_Flakes  P   C        90         3   0   210
## 11             Cap'n'Crunch  Q   C       120         1   2   220
## 12              Cheerios  G   C       110         6   2   290
## 13      Cinnamon_Toast_Crunch  G   C       120         1   3   210
## 14              Clusters  G   C       110         3   2   140
## 15             Cocoa_Puffs  G   C       110         1   1   180
## 16             Corn_Chex  R   C       110         2   0   280
## 17             Corn_Flakes  K   C       100         2   0   290
## 18             Corn_Pops  K   C       110         1   0    90
## 19             Count_Chocula  G   C       110         1   1   180
## 20      Cracklin'_Oat_Bran  K   C       110         3   3   140
## 22              Crispix  K   C       110         2   0   220
## 23      Crispy_Wheat_&_Raisins  G   C       100         2   1   140
## 24              Double_Chex  R   C       100         2   0   190
## 25              Froot_Loops  K   C       110         2   1   125
## 26              Frosted_Flakes  K   C       110         1   0   200
## 27      Frosted_Mini-Wheats  K   C       100         3   0     0
## 28 Fruit_&_Fibre_Dates,_Walnuts,_and_Oats  P   C       120         3   2   160
## 29              Fruitful_Bran  K   C       120         3   0   240
```

Q9. Identify and remove duplicated data in your dataset.

## 30	Fruity_Pebbles	P	C	110	1	1	135
## 31	Golden_Crisp	P	C	100	2	0	45
## 32	Golden_Grahams	G	C	110	1	1	280
## 33	Grape_Nuts_Flakes	P	C	100	3	1	140
## 34	Grape-Nuts	P	C	110	3	0	170
## 35	Great_Grains_Pecan	P	C	120	3	3	75
## 36	Honey_Graham_Ohs	Q	C	120	1	2	220
## 37	Honey_Nut_Cheerios	G	C	110	3	1	250
## 38	Honey-comb	P	C	110	1	0	180
## 39	Just_Right_Crunchy__Nuggets	K	C	110	2	1	170
## 40	Just_Right_Fruit_&_Nut	K	C	140	3	1	170
## 41	Kix	G	C	110	2	1	260
## 42	Life	Q	C	100	4	2	150
## 43	Lucky_Charms	G	C	110	2	1	180
## 44	Maypo	A	H	100	4	1	0
## 45	Muesli_Raisins,_Dates,_&_Almonds	R	C	150	4	3	95
## 46	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3	150
## 47	Mueslix_Crispy_Blend	K	C	160	3	2	150
## 48	Multi-Grain_Cheerios	G	C	100	2	1	220
## 49	Nut&Honey_Crunch	K	C	120	2	1	190
## 50	Nutri-Grain_Almond-Raisin	K	C	140	3	2	220
## 51	Nutri-grain_Wheat	K	C	90	3	0	170
## 52	Oatmeal_Raisin_Crisp	G	C	130	3	2	170
## 53	Post_Nat._Raisin_Bran	P	C	120	3	1	200
## 54	Product_19	K	C	100	3	0	320
## 55	Puffed_Rice	Q	C	50	1	0	0
## 56	Puffed_Wheat	Q	C	50	2	0	0
## 57	Quaker_Oat_Squares	Q	C	100	4	1	135
## 59	Raisin_Bran	K	C	120	3	1	210
## 60	Raisin_Nut_Bran	G	C	100	3	2	140
## 61	Raisin_Squares	K	C	90	2	0	0
## 62	Rice_Chex	R	C	110	1	0	240
## 63	Rice_Krispies	K	C	110	2	0	290
## 64	Shredded_Wheat	N	C	80	2	0	0
## 65	Shredded_Wheat_'n'Bran	N	C	90	3	0	0
## 66	Shredded_Wheat_spoon_size	N	C	90	3	0	0
## 67	Smacks	K	C	110	2	1	70
## 68	Special_K	K	C	110	6	0	230
## 69	Strawberry_Fruit_Wheats	N	C	90	2	0	15
## 70	Total_Corn_Flakes	G	C	110	2	1	200
## 71	Total_Raisin_Bran	G	C	140	3	1	190
## 72	Total_Whole_Grain	G	C	100	3	1	200
## 73	Triples	G	C	110	2	1	250
## 74	Trix	G	C	110	1	1	140
## 75	Wheat_Chex	R	C	100	3	1	230
## 76	Wheaties	G	C	100	3	1	200
## 77	Wheaties_Honey_Gold	G	C	110	2	1	200
##	fiber carbo sugars potass vitamins shelf weight cups rating						
## 1	10.0	5.0	6	280	25	3	1.00 0.33 68.40297
## 2	2.0	8.0	8	135	0	3	1.00 1.00 33.98368
## 3	9.0	7.0	5	320	25	3	1.00 0.33 59.42551
## 4	14.0	8.0	0	330	25	3	1.00 0.50 93.70491
## 6	1.5	10.5	10	70	25	1	1.00 0.75 29.50954
## 7	1.0	11.0	14	30	25	2	1.00 1.00 33.17409

Q9. Identify and remove duplicated data in your dataset.

---

```
## 8    2.0 18.0      8   100      25    3    1.33 0.75 37.03856
## 9    4.0 15.0      6   125      25    1    1.00 0.67 49.12025
## 10   5.0 13.0      5   190      25    3    1.00 0.67 53.31381
## 11   0.0 12.0     12    35      25    2    1.00 0.75 18.04285
## 12   2.0 17.0      1   105      25    1    1.00 1.25 50.76500
## 13   0.0 13.0      9    45      25    2    1.00 0.75 19.82357
## 14   2.0 13.0      7   105      25    3    1.00 0.50 40.40021
## 15   0.0 12.0     13    55      25    2    1.00 1.00 22.73645
## 16   0.0 22.0      3    25      25    1    1.00 1.00 41.44502
## 17   1.0 21.0      2    35      25    1    1.00 1.00 45.86332
## 18   1.0 13.0     12    20      25    2    1.00 1.00 35.78279
## 19   0.0 12.0     13    65      25    2    1.00 1.00 22.39651
## 20   4.0 10.0      7   160      25    3    1.00 0.50 40.44877
## 22   1.0 21.0      3    30      25    3    1.00 1.00 46.89564
## 23   2.0 11.0     10   120      25    3    1.00 0.75 36.17620
## 24   1.0 18.0      5    80      25    3    1.00 0.75 44.33086
## 25   1.0 11.0     13    30      25    2    1.00 1.00 32.20758
## 26   1.0 14.0     11    25      25    1    1.00 0.75 31.43597
## 27   3.0 14.0      7   100      25    2    1.00 0.80 58.34514
## 28   5.0 12.0     10   200      25    3    1.25 0.67 40.91705
## 29   5.0 14.0     12   190      25    3    1.33 0.67 41.01549
## 30   0.0 13.0     12    25      25    2    1.00 0.75 28.02576
## 31   0.0 11.0     15    40      25    1    1.00 0.88 35.25244
## 32   0.0 15.0      9    45      25    2    1.00 0.75 23.80404
## 33   3.0 15.0      5    85      25    3    1.00 0.88 52.07690
## 34   3.0 17.0      3    90      25    3    1.00 0.25 53.37101
## 35   3.0 13.0      4   100      25    3    1.00 0.33 45.81172
## 36   1.0 12.0     11    45      25    2    1.00 1.00 21.87129
## 37   1.5 11.5     10    90      25    1    1.00 0.75 31.07222
## 38   0.0 14.0     11    35      25    1    1.00 1.33 28.74241
## 39   1.0 17.0      6    60     100    3    1.00 1.00 36.52368
## 40   2.0 20.0      9    95     100    3    1.30 0.75 36.47151
## 41   0.0 21.0      3    40      25    2    1.00 1.50 39.24111
## 42   2.0 12.0      6    95      25    2    1.00 0.67 45.32807
## 43   0.0 12.0     12    55      25    2    1.00 1.00 26.73451
## 44   0.0 16.0      3    95      25    2    1.00 1.00 54.85092
## 45   3.0 16.0     11   170      25    3    1.00 1.00 37.13686
## 46   3.0 16.0     11   170      25    3    1.00 1.00 34.13976
## 47   3.0 17.0     13   160      25    3    1.50 0.67 30.31335
## 48   2.0 15.0      6    90      25    1    1.00 1.00 40.10596
## 49   0.0 15.0      9    40      25    2    1.00 0.67 29.92429
## 50   3.0 21.0      7   130      25    3    1.33 0.67 40.69232
## 51   3.0 18.0      2    90      25    3    1.00 1.00 59.64284
## 52   1.5 13.5     10   120      25    3    1.25 0.50 30.45084
## 53   6.0 11.0     14   260      25    3    1.33 0.67 37.84059
## 54   1.0 20.0      3    45     100    3    1.00 1.00 41.50354
## 55   0.0 13.0      0    15      0     3    0.50 1.00 60.75611
## 56   1.0 10.0      0    50      0     3    0.50 1.00 63.00565
## 57   2.0 14.0      6   110      25    3    1.00 0.50 49.51187
## 59   5.0 14.0     12   240      25    2    1.33 0.75 39.25920
## 60   2.5 10.5      8   140      25    3    1.00 0.50 39.70340
## 61   2.0 15.0      6   110      25    3    1.00 0.50 55.33314
## 62   0.0 23.0      2    30      25    1    1.00 1.13 41.99893
## 63   0.0 22.0      3    35      25    1    1.00 1.00 40.56016
```

It shows that we do not have any duplicate values in our dataset.

---

```
## 64  3.0 16.0    0    95      0    1  0.83 1.00 68.23588
## 65  4.0 19.0    0   140      0    1  1.00 0.67 74.47295
## 66  3.0 20.0    0   120      0    1  1.00 0.67 72.80179
## 67  1.0  9.0   15    40     25    2  1.00 0.75 31.23005
## 68  1.0 16.0    3    55     25    1  1.00 1.00 53.13132
## 69  3.0 15.0    5    90     25    2  1.00 1.00 59.36399
## 70  0.0 21.0    3    35    100    3  1.00 1.00 38.83975
## 71  4.0 15.0   14   230    100    3  1.50 1.00 28.59278
## 72  3.0 16.0    3   110    100    3  1.00 1.00 46.65884
## 73  0.0 21.0    3    60     25    3  1.00 0.75 39.10617
## 74  0.0 13.0   12    25     25    2  1.00 1.00 27.75330
## 75  3.0 17.0    3   115     25    1  1.00 0.67 49.78744
## 76  3.0 17.0    3   110     25    1  1.00 1.00 51.59219
## 77  1.0 16.0    8    60     25    1  1.00 0.75 36.18756
```

It shows that we do not have any duplicate values in our dataset.

Q10.Reorder multiple rows in descending order.

Here, we are sorting the rows based on ratings in descending order.

```
Cereals <- arrange(Cereals,desc(Cereals$rating))
Cereals
```

```
##              name mfr type calories protein fat sodium
## 1 All-Bran_with_Extra_Fiber K   C      50      4  0    140
## 2 Shredded_Wheat_'n'Bran N   C      90      3  0      0
## 3 Shredded_Wheat_spoon_size N   C      90      3  0      0
## 4 100%_Bran N   C      70      4  1    130
## 5 Shredded_Wheat N   C      80      2  0      0
## 6 Puffed_Wheat Q   C      50      2  0      0
## 7 Puffed_Rice Q   C      50      1  0      0
## 8 Nutri-grain_Wheat K   C      90      3  0    170
## 9 All-Bran K   C      70      4  1    260
## 10 Strawberry_Fruit_Wheats N   C      90      2  0     15
## 11 Frosted_Mini-Wheats K   C     100      3  0      0
## 12 Raisin_Squares K   C      90      2  0      0
## 13 Maypo A   H     100      4  1      0
## 14 Grape-Nuts P   C     110      3  0    170
## 15 Bran_Flakes P   C      90      3  0    210
## 16 Special_K K   C     110      6  0    230
## 17 Grape_Nuts_Flakes P   C     100      3  1    140
## 18 Wheaties G   C     100      3  1    200
## 19 Cheerios G   C     110      6  2    290
## 20 Wheat_Chex R   C     100      3  1    230
## 21 Quaker_Oat_Squares Q   C     100      4  1    135
## 22 Bran_Chex R   C      90      2  1    200
## 23 Crispix K   C     110      2  0    220
## 24 Total_Whole_Grain G   C     100      3  1    200
## 25 Corn_Flakes K   C     100      2  0    290
## 26 Great_Grains_Pecan P   C     120      3  3     75
```



Here, we are sorting the rows based on ratings in descending order.

---

##	27		Life	Q	C	100	4	2	150	
##	28		Double_Chex	R	C	100	2	0	190	
##	29		Rice_Chex	R	C	110	1	0	240	
##	30		Product_19	K	C	100	3	0	320	
##	31		Corn_Chex	R	C	110	2	0	280	
##	32		Fruitful_Bran	K	C	120	3	0	240	
##	33	Fruit_&_Fibre_Dates,_Walnuts,_and_Oats	P	C	120	3	2	160		
##	34	Nutri-Grain_Almond-Raisin	K	C	140	3	2	220		
##	35	Rice_Krispies	K	C	110	2	0	290		
##	36	Cracklin'_Oat_Bran	K	C	110	3	3	140		
##	37	Clusters	G	C	110	3	2	140		
##	38	Multi-Grain_Cheerios	G	C	100	2	1	220		
##	39	Raisin_Nut_Bran	G	C	100	3	2	140		
##	40	Raisin_Bran	K	C	120	3	1	210		
##	41	Kix	G	C	110	2	1	260		
##	42	Triples	G	C	110	2	1	250		
##	43	Total_Corn_Flakes	G	C	110	2	1	200		
##	44	Post_Nat._Raisin_Bran	P	C	120	3	1	200		
##	45	Muesli_Raisins,_Dates,_&_Almonds	R	C	150	4	3	95		
##	46	Basic_4	G	C	130	3	2	210		
##	47	Just_Right_Crunchy__Nuggets	K	C	110	2	1	170		
##	48	Just_Right_Fruit_&_Nut	K	C	140	3	1	170		
##	49	Wheaties_Honey_Gold	G	C	110	2	1	200		
##	50	Crispy_Wheat_&_Raisins	G	C	100	2	1	140		
##	51	Corn_Pops	K	C	110	1	0	90		
##	52	Golden_Crisp	P	C	100	2	0	45		
##	53	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3	150		
##	54	100%_Natural_Bran	Q	C	120	3	5	15		
##	55	Apple_Jacks	K	C	110	2	0	125		
##	56	Froot_Loops	K	C	110	2	1	125		
##	57	Frosted_Flakes	K	C	110	1	0	200		
##	58	Smacks	K	C	110	2	1	70		
##	59	Honey_Nut_Cheerios	G	C	110	3	1	250		
##	60	Oatmeal_Raisin_Crisp	G	C	130	3	2	170		
##	61	Mueslix_Crispy_Blend	K	C	160	3	2	150		
##	62	Nut&Honey_Crunch	K	C	120	2	1	190		
##	63	Apple_Cinnamon_Cheerios	G	C	110	2	2	180		
##	64	Honey-comb	P	C	110	1	0	180		
##	65	Total_Raisin_Bran	G	C	140	3	1	190		
##	66	Fruity_Pebbles	P	C	110	1	1	135		
##	67	Trix	G	C	110	1	1	140		
##	68	Lucky_Charms	G	C	110	2	1	180		
##	69	Golden_Grahams	G	C	110	1	1	280		
##	70	Cocoa_Puffs	G	C	110	1	1	180		
##	71	Count_Chocula	G	C	110	1	1	180		
##	72	Honey_Graham_Ohs	Q	C	120	1	2	220		
##	73	Cinnamon_Toast_Crunch	G	C	120	1	3	210		
##	74	Cap'n'Crunch	Q	C	120	1	2	220		
##		fiber	carbo	sugars	potass	vitamins	shelf	weight	cups	rating
##	1	14.0	8.0	0	330	25	3	1.00	0.50	93.70491
##	2	4.0	19.0	0	140	0	1	1.00	0.67	74.47295
##	3	3.0	20.0	0	120	0	1	1.00	0.67	72.80179
##	4	10.0	5.0	6	280	25	3	1.00	0.33	68.40297
##	5	3.0	16.0	0	95	0	1	0.83	1.00	68.23588

Here, we are sorting the rows based on ratings in descending order.

---

## 6	1.0	10.0	0	50	0	3	0.50	1.00	63.00565
## 7	0.0	13.0	0	15	0	3	0.50	1.00	60.75611
## 8	3.0	18.0	2	90	25	3	1.00	1.00	59.64284
## 9	9.0	7.0	5	320	25	3	1.00	0.33	59.42551
## 10	3.0	15.0	5	90	25	2	1.00	1.00	59.36399
## 11	3.0	14.0	7	100	25	2	1.00	0.80	58.34514
## 12	2.0	15.0	6	110	25	3	1.00	0.50	55.33314
## 13	0.0	16.0	3	95	25	2	1.00	1.00	54.85092
## 14	3.0	17.0	3	90	25	3	1.00	0.25	53.37101
## 15	5.0	13.0	5	190	25	3	1.00	0.67	53.31381
## 16	1.0	16.0	3	55	25	1	1.00	1.00	53.13132
## 17	3.0	15.0	5	85	25	3	1.00	0.88	52.07690
## 18	3.0	17.0	3	110	25	1	1.00	1.00	51.59219
## 19	2.0	17.0	1	105	25	1	1.00	1.25	50.76500
## 20	3.0	17.0	3	115	25	1	1.00	0.67	49.78744
## 21	2.0	14.0	6	110	25	3	1.00	0.50	49.51187
## 22	4.0	15.0	6	125	25	1	1.00	0.67	49.12025
## 23	1.0	21.0	3	30	25	3	1.00	1.00	46.89564
## 24	3.0	16.0	3	110	100	3	1.00	1.00	46.65884
## 25	1.0	21.0	2	35	25	1	1.00	1.00	45.86332
## 26	3.0	13.0	4	100	25	3	1.00	0.33	45.81172
## 27	2.0	12.0	6	95	25	2	1.00	0.67	45.32807
## 28	1.0	18.0	5	80	25	3	1.00	0.75	44.33086
## 29	0.0	23.0	2	30	25	1	1.00	1.13	41.99893
## 30	1.0	20.0	3	45	100	3	1.00	1.00	41.50354
## 31	0.0	22.0	3	25	25	1	1.00	1.00	41.44502
## 32	5.0	14.0	12	190	25	3	1.33	0.67	41.01549
## 33	5.0	12.0	10	200	25	3	1.25	0.67	40.91705
## 34	3.0	21.0	7	130	25	3	1.33	0.67	40.69232
## 35	0.0	22.0	3	35	25	1	1.00	1.00	40.56016
## 36	4.0	10.0	7	160	25	3	1.00	0.50	40.44877
## 37	2.0	13.0	7	105	25	3	1.00	0.50	40.40021
## 38	2.0	15.0	6	90	25	1	1.00	1.00	40.10596
## 39	2.5	10.5	8	140	25	3	1.00	0.50	39.70340
## 40	5.0	14.0	12	240	25	2	1.33	0.75	39.25920
## 41	0.0	21.0	3	40	25	2	1.00	1.50	39.24111
## 42	0.0	21.0	3	60	25	3	1.00	0.75	39.10617
## 43	0.0	21.0	3	35	100	3	1.00	1.00	38.83975
## 44	6.0	11.0	14	260	25	3	1.33	0.67	37.84059
## 45	3.0	16.0	11	170	25	3	1.00	1.00	37.13686
## 46	2.0	18.0	8	100	25	3	1.33	0.75	37.03856
## 47	1.0	17.0	6	60	100	3	1.00	1.00	36.52368
## 48	2.0	20.0	9	95	100	3	1.30	0.75	36.47151
## 49	1.0	16.0	8	60	25	1	1.00	0.75	36.18756
## 50	2.0	11.0	10	120	25	3	1.00	0.75	36.17620
## 51	1.0	13.0	12	20	25	2	1.00	1.00	35.78279
## 52	0.0	11.0	15	40	25	1	1.00	0.88	35.25244
## 53	3.0	16.0	11	170	25	3	1.00	1.00	34.13976
## 54	2.0	8.0	8	135	0	3	1.00	1.00	33.98368
## 55	1.0	11.0	14	30	25	2	1.00	1.00	33.17409
## 56	1.0	11.0	13	30	25	2	1.00	1.00	32.20758
## 57	1.0	14.0	11	25	25	1	1.00	0.75	31.43597
## 58	1.0	9.0	15	40	25	2	1.00	0.75	31.23005
## 59	1.5	11.5	10	90	25	1	1.00	0.75	31.07222

Q11. Rename some of the column names in your dataset.

---

```
## 60  1.5  13.5    10   120    25    3  1.25 0.50 30.45084
## 61  3.0  17.0    13   160    25    3  1.50 0.67 30.31335
## 62  0.0  15.0     9    40    25    2  1.00 0.67 29.92429
## 63  1.5  10.5    10    70    25    1  1.00 0.75 29.50954
## 64  0.0  14.0    11    35    25    1  1.00 1.33 28.74241
## 65  4.0  15.0    14   230   100    3  1.50 1.00 28.59278
## 66  0.0  13.0    12    25    25    2  1.00 0.75 28.02576
## 67  0.0  13.0    12    25    25    2  1.00 1.00 27.75330
## 68  0.0  12.0    12    55    25    2  1.00 1.00 26.73451
## 69  0.0  15.0     9    45    25    2  1.00 0.75 23.80404
## 70  0.0  12.0    13    55    25    2  1.00 1.00 22.73645
## 71  0.0  12.0    13    65    25    2  1.00 1.00 22.39651
## 72  1.0  12.0    11    45    25    2  1.00 1.00 21.87129
## 73  0.0  13.0     9    45    25    2  1.00 0.75 19.82357
## 74  0.0  12.0    12    35    25    2  1.00 0.75 18.04285
```

Q11. Rename some of the column names in your dataset.

Here, We have renamed the two variables, 1: name to Cereals Names, and 2: carbo to carbohydrates.

```
names(Cereals)[names(Cereals) == "name"] <- "Cereals Names"
names(Cereals)[names(Cereals) == "carbo"] <- "carbohydrates"
Cereals
```

```
##           Cereals Names mfr type calories protein fat sodium
## 1      All-Bran_with_Extra_Fiber K   C      50      4  0    140
## 2      Shredded_Wheat_ 'n' Bran N   C      90      3  0     0
## 3      Shredded_Wheat_spoon_size N   C      90      3  0     0
## 4           100%_Bran N   C      70      4  1    130
## 5      Shredded_Wheat N   C      80      2  0     0
## 6      Puffed_Wheat Q   C      50      2  0     0
## 7      Puffed_Rice Q   C      50      1  0     0
## 8      Nutri-grain_Wheat K   C      90      3  0    170
## 9      All-Bran K   C      70      4  1    260
## 10     Strawberry_Fruit_Wheats N   C      90      2  0     15
## 11     Frosted_Mini-Wheats K   C     100      3  0     0
## 12     Raisin_Squares K   C      90      2  0     0
## 13           Maypo A   H     100      4  1     0
## 14      Grape-Nuts P   C     110      3  0    170
## 15      Bran_Flakes P   C      90      3  0    210
## 16      Special_K K   C     110      6  0    230
## 17      Grape_Nuts_Flakes P   C     100      3  1    140
## 18      Wheaties G   C     100      3  1    200
## 19      Cheerios G   C     110      6  2    290
## 20      Wheat_Chex R   C     100      3  1    230
## 21      Quaker_Oat_Squares Q   C     100      4  1    135
## 22      Bran_Chex R   C      90      2  1    200
## 23      Crispix K   C     110      2  0    220
## 24      Total_Whole_Grain G   C     100      3  1    200
## 25      Corn_Flakes K   C     100      2  0    290
```

Here, We have renamed the two variables, 1: name to Cereals Names, and 2: carbo to carbohydrates.

## 26	Great_Grains_Pecan	P	C	120	3	3	75
## 27	Life	Q	C	100	4	2	150
## 28	Double_Chex	R	C	100	2	0	190
## 29	Rice_Chex	R	C	110	1	0	240
## 30	Product_19	K	C	100	3	0	320
## 31	Corn_Chex	R	C	110	2	0	280
## 32	Fruitful_Bran	K	C	120	3	0	240
## 33	Fruit_&_Fibre_Dates,_Walnuts,_and_Oats	P	C	120	3	2	160
## 34	Nutri-Grain_Almond-Raisin	K	C	140	3	2	220
## 35	Rice_Krispies	K	C	110	2	0	290
## 36	Cracklin'_Oat_Bran	K	C	110	3	3	140
## 37	Clusters	G	C	110	3	2	140
## 38	Multi-Grain_Cheerios	G	C	100	2	1	220
## 39	Raisin_Nut_Bran	G	C	100	3	2	140
## 40	Raisin_Bran	K	C	120	3	1	210
## 41	Kix	G	C	110	2	1	260
## 42	Triples	G	C	110	2	1	250
## 43	Total_Corn_Flakes	G	C	110	2	1	200
## 44	Post_Nat._Raisin_Bran	P	C	120	3	1	200
## 45	Muesli_Raisins,_Dates,_&_Almonds	R	C	150	4	3	95
## 46	Basic_4	G	C	130	3	2	210
## 47	Just_Right_Crunchy__Nuggets	K	C	110	2	1	170
## 48	Just_Right_Fruit_&_Nut	K	C	140	3	1	170
## 49	Wheaties_Honey_Gold	G	C	110	2	1	200
## 50	Crispy_Wheat_&_Raisins	G	C	100	2	1	140
## 51	Corn_Pops	K	C	110	1	0	90
## 52	Golden_Crisp	P	C	100	2	0	45
## 53	Muesli_Raisins,_Peaches,_&_Pecans	R	C	150	4	3	150
## 54	100%_Natural_Bran	Q	C	120	3	5	15
## 55	Apple_Jacks	K	C	110	2	0	125
## 56	Froot_Loops	K	C	110	2	1	125
## 57	Frosted_Flakes	K	C	110	1	0	200
## 58	Smacks	K	C	110	2	1	70
## 59	Honey_Nut_Cheerios	G	C	110	3	1	250
## 60	Oatmeal_Raisin_Crisp	G	C	130	3	2	170
## 61	Mueslix_Crispy_Blend	K	C	160	3	2	150
## 62	Nut&Honey_Crunch	K	C	120	2	1	190
## 63	Apple_Cinnamon_Cheerios	G	C	110	2	2	180
## 64	Honey-comb	P	C	110	1	0	180
## 65	Total_Raisin_Bran	G	C	140	3	1	190
## 66	Fruity_Pebbles	P	C	110	1	1	135
## 67	Trix	G	C	110	1	1	140
## 68	Lucky_Charm	G	C	110	2	1	180
## 69	Golden_Grahams	G	C	110	1	1	280
## 70	Cocoa_Puffs	G	C	110	1	1	180
## 71	Count_Chocula	G	C	110	1	1	180
## 72	Honey_Graham_Ohs	Q	C	120	1	2	220
## 73	Cinnamon_Toast_Crunch	G	C	120	1	3	210
## 74	Cap'n'Crunch	Q	C	120	1	2	220
##	fiber carbohydrates sugars potass vitamins shelf weight cups rating						
## 1	14.0	8.0	0	330	25	3	1.00 0.50 93.70491
## 2	4.0	19.0	0	140	0	1	1.00 0.67 74.47295
## 3	3.0	20.0	0	120	0	1	1.00 0.67 72.80179
## 4	10.0	5.0	6	280	25	3	1.00 0.33 68.40297

Here, We have renamed the two variables, 1: name to Cereals Names, and 2: carbo to carbohydrates.

---

## 5	3.0	16.0	0	95	0	1	0.83	1.00	68.23588
## 6	1.0	10.0	0	50	0	3	0.50	1.00	63.00565
## 7	0.0	13.0	0	15	0	3	0.50	1.00	60.75611
## 8	3.0	18.0	2	90	25	3	1.00	1.00	59.64284
## 9	9.0	7.0	5	320	25	3	1.00	0.33	59.42551
## 10	3.0	15.0	5	90	25	2	1.00	1.00	59.36399
## 11	3.0	14.0	7	100	25	2	1.00	0.80	58.34514
## 12	2.0	15.0	6	110	25	3	1.00	0.50	55.33314
## 13	0.0	16.0	3	95	25	2	1.00	1.00	54.85092
## 14	3.0	17.0	3	90	25	3	1.00	0.25	53.37101
## 15	5.0	13.0	5	190	25	3	1.00	0.67	53.31381
## 16	1.0	16.0	3	55	25	1	1.00	1.00	53.13132
## 17	3.0	15.0	5	85	25	3	1.00	0.88	52.07690
## 18	3.0	17.0	3	110	25	1	1.00	1.00	51.59219
## 19	2.0	17.0	1	105	25	1	1.00	1.25	50.76500
## 20	3.0	17.0	3	115	25	1	1.00	0.67	49.78744
## 21	2.0	14.0	6	110	25	3	1.00	0.50	49.51187
## 22	4.0	15.0	6	125	25	1	1.00	0.67	49.12025
## 23	1.0	21.0	3	30	25	3	1.00	1.00	46.89564
## 24	3.0	16.0	3	110	100	3	1.00	1.00	46.65884
## 25	1.0	21.0	2	35	25	1	1.00	1.00	45.86332
## 26	3.0	13.0	4	100	25	3	1.00	0.33	45.81172
## 27	2.0	12.0	6	95	25	2	1.00	0.67	45.32807
## 28	1.0	18.0	5	80	25	3	1.00	0.75	44.33086
## 29	0.0	23.0	2	30	25	1	1.00	1.13	41.99893
## 30	1.0	20.0	3	45	100	3	1.00	1.00	41.50354
## 31	0.0	22.0	3	25	25	1	1.00	1.00	41.44502
## 32	5.0	14.0	12	190	25	3	1.33	0.67	41.01549
## 33	5.0	12.0	10	200	25	3	1.25	0.67	40.91705
## 34	3.0	21.0	7	130	25	3	1.33	0.67	40.69232
## 35	0.0	22.0	3	35	25	1	1.00	1.00	40.56016
## 36	4.0	10.0	7	160	25	3	1.00	0.50	40.44877
## 37	2.0	13.0	7	105	25	3	1.00	0.50	40.40021
## 38	2.0	15.0	6	90	25	1	1.00	1.00	40.10596
## 39	2.5	10.5	8	140	25	3	1.00	0.50	39.70340
## 40	5.0	14.0	12	240	25	2	1.33	0.75	39.25920
## 41	0.0	21.0	3	40	25	2	1.00	1.50	39.24111
## 42	0.0	21.0	3	60	25	3	1.00	0.75	39.10617
## 43	0.0	21.0	3	35	100	3	1.00	1.00	38.83975
## 44	6.0	11.0	14	260	25	3	1.33	0.67	37.84059
## 45	3.0	16.0	11	170	25	3	1.00	1.00	37.13686
## 46	2.0	18.0	8	100	25	3	1.33	0.75	37.03856
## 47	1.0	17.0	6	60	100	3	1.00	1.00	36.52368
## 48	2.0	20.0	9	95	100	3	1.30	0.75	36.47151
## 49	1.0	16.0	8	60	25	1	1.00	0.75	36.18756
## 50	2.0	11.0	10	120	25	3	1.00	0.75	36.17620
## 51	1.0	13.0	12	20	25	2	1.00	1.00	35.78279
## 52	0.0	11.0	15	40	25	1	1.00	0.88	35.25244
## 53	3.0	16.0	11	170	25	3	1.00	1.00	34.13976
## 54	2.0	8.0	8	135	0	3	1.00	1.00	33.98368
## 55	1.0	11.0	14	30	25	2	1.00	1.00	33.17409
## 56	1.0	11.0	13	30	25	2	1.00	1.00	32.20758
## 57	1.0	14.0	11	25	25	1	1.00	0.75	31.43597
## 58	1.0	9.0	15	40	25	2	1.00	0.75	31.23005

Q12. Add new variables in your data frame by using a mathematical function (for e.g. – multiply an existing column by 2 and add it as a new variable to your data frame).

```
## 59  1.5      11.5    10    90      25    1    1.00 0.75 31.07222
## 60  1.5      13.5    10   120      25    3    1.25 0.50 30.45084
## 61  3.0      17.0    13   160      25    3    1.50 0.67 30.31335
## 62  0.0      15.0     9    40      25    2    1.00 0.67 29.92429
## 63  1.5      10.5    10    70      25    1    1.00 0.75 29.50954
## 64  0.0      14.0    11    35      25    1    1.00 1.33 28.74241
## 65  4.0      15.0    14   230     100    3    1.50 1.00 28.59278
## 66  0.0      13.0    12    25      25    2    1.00 0.75 28.02576
## 67  0.0      13.0    12    25      25    2    1.00 1.00 27.75330
## 68  0.0      12.0    12    55      25    2    1.00 1.00 26.73451
## 69  0.0      15.0     9    45      25    2    1.00 0.75 23.80404
## 70  0.0      12.0    13    55      25    2    1.00 1.00 22.73645
## 71  0.0      12.0    13    65      25    2    1.00 1.00 22.39651
## 72  1.0      12.0    11    45      25    2    1.00 1.00 21.87129
## 73  0.0      13.0     9    45      25    2    1.00 0.75 19.82357
## 74  0.0      12.0    12    35      25    2    1.00 0.75 18.04285
```

Q12. Add new variables in your data frame by using a mathematical function (for e.g. – multiply an existing column by 2 and add it as a new variable to your data frame).

Here, we have created a new variable called `double_calories` by using `calories` variable, we multiplied `calories` by 2 and stored in `double_calories`.

```
Cereals %>% mutate(double_calories = calories*2)
```

```
##           Cereals Names mfr type calories protein fat sodium
## 1 All-Bran_with_Extra_Fiber K   C      50         4  0    140
## 2 Shredded_Wheat_ 'n' Bran N   C      90         3  0     0
## 3 Shredded_Wheat_spoon_size N   C      90         3  0     0
## 4 100%_Bran N   C      70         4  1    130
## 5 Shredded_Wheat N   C      80         2  0     0
## 6 Puffed_Wheat Q   C      50         2  0     0
## 7 Puffed_Rice Q   C      50         1  0     0
## 8 Nutri-grain_Wheat K   C      90         3  0    170
## 9 All-Bran K   C      70         4  1    260
## 10 Strawberry_Fruit_Wheats N   C      90         2  0     15
## 11 Frosted_Mini-Wheats K   C     100         3  0     0
## 12 Raisin_Squares K   C      90         2  0     0
## 13 Maypo A   H     100         4  1     0
## 14 Grape-Nuts P   C     110         3  0    170
## 15 Bran_Flakes P   C      90         3  0    210
## 16 Special_K K   C     110         6  0    230
## 17 Grape_Nuts_Flakes P   C     100         3  1    140
## 18 Wheaties G   C     100         3  1    200
## 19 Cheerios G   C     110         6  2    290
## 20 Wheat_Chex R   C     100         3  1    230
## 21 Quaker_Oat_Squares Q   C     100         4  1    135
## 22 Bran_Chex R   C      90         2  1    200
## 23 Crispix K   C     110         2  0    220
## 24 Total_Whole_Grain G   C     100         3  1    200
```

Here, we have created a new variable called `double_calories` by using `calories` variable, we multiplied `calories` by 2 and stored in `double_calories`.

---

```
## 25          Corn_Flakes      K    C    100    2    0    290
## 26      Great_Grains_Pecan    P    C    120    3    3     75
## 27              Life        Q    C    100    4    2    150
## 28      Double_Chex          R    C    100    2    0    190
## 29      Rice_Chex           R    C    110    1    0    240
## 30      Product_19          K    C    100    3    0    320
## 31      Corn_Chex           R    C    110    2    0    280
## 32      Fruitful_Bran        K    C    120    3    0    240
## 33 Fruit_&_Fibre_Dates,_Walnuts,_and_Oats P    C    120    3    2    160
## 34      Nutri-Grain_Almond-Raisin K    C    140    3    2    220
## 35      Rice_Krispies        K    C    110    2    0    290
## 36      Cracklin'_Oat_Bran    K    C    110    3    3    140
## 37      Clusters             G    C    110    3    2    140
## 38      Multi-Grain_Cheerios  G    C    100    2    1    220
## 39      Raisin_Nut_Bran       G    C    100    3    2    140
## 40      Raisin_Bran           K    C    120    3    1    210
## 41      Kix                  G    C    110    2    1    260
## 42      Triples              G    C    110    2    1    250
## 43      Total_Corn_Flakes     G    C    110    2    1    200
## 44      Post_Nat._Raisin_Bran P    C    120    3    1    200
## 45      Muesli_Raisins,_Dates,_&_Almonds R    C    150    4    3     95
## 46      Basic_4              G    C    130    3    2    210
## 47      Just_Right_Crunchy__Nuggets K    C    110    2    1    170
## 48      Just_Right_Fruit_&_Nut K    C    140    3    1    170
## 49      Wheaties_Honey_Gold   G    C    110    2    1    200
## 50      Crispy_Wheat_&_Raisins G    C    100    2    1    140
## 51      Corn_Pops            K    C    110    1    0     90
## 52      Golden_Crisp          P    C    100    2    0     45
## 53      Muesli_Raisins,_Peaches,_&_Pecans R    C    150    4    3    150
## 54      100%_Natural_Bran     Q    C    120    3    5     15
## 55      Apple_Jacks           K    C    110    2    0    125
## 56      Froot_Loops          K    C    110    2    1    125
## 57      Frosted_Flakes        K    C    110    1    0    200
## 58      Smacks               K    C    110    2    1     70
## 59      Honey_Nut_Cheerios    G    C    110    3    1    250
## 60      Oatmeal_Raisin_Crisp  G    C    130    3    2    170
## 61      Mueslix_Crispy_Blend  K    C    160    3    2    150
## 62      Nut&Honey_Crunch      K    C    120    2    1    190
## 63      Apple_Cinnamon_Cheerios G    C    110    2    2    180
## 64      Honey-comb            P    C    110    1    0    180
## 65      Total_Raisin_Bran     G    C    140    3    1    190
## 66      Fruity_Pebbles        P    C    110    1    1    135
## 67      Trix                  G    C    110    1    1    140
## 68      Lucky_Charms          G    C    110    2    1    180
## 69      Golden_Grahams        G    C    110    1    1    280
## 70      Cocoa_Puffs           G    C    110    1    1    180
## 71      Count_Chocula         G    C    110    1    1    180
## 72      Honey_Graham_Ohs      Q    C    120    1    2    220
## 73      Cinnamon_Toast_Crunch G    C    120    1    3    210
## 74      Cap'n'Crunch          Q    C    120    1    2    220
##      fiber carbohydrates sugars potass vitamins shelf weight cups  rating
## 1      14.0           8.0      0      330      25      3      1.00 0.50 93.70491
## 2       4.0          19.0      0      140       0      1      1.00 0.67 74.47295
## 3       3.0          20.0      0      120       0      1      1.00 0.67 72.80179
```

Here, we have created a new variable called `double_calories` by using `calories` variable, we multiplied `calories` by 2 and stored in `double_calories`.

---

```
## 4  10.0      5.0      6  280      25      3  1.00 0.33 68.40297
## 5   3.0     16.0      0   95       0      1  0.83 1.00 68.23588
## 6   1.0     10.0      0   50       0      3  0.50 1.00 63.00565
## 7   0.0     13.0      0   15       0      3  0.50 1.00 60.75611
## 8   3.0     18.0      2   90      25      3  1.00 1.00 59.64284
## 9   9.0      7.0      5  320      25      3  1.00 0.33 59.42551
## 10  3.0     15.0      5   90      25      2  1.00 1.00 59.36399
## 11  3.0     14.0      7  100      25      2  1.00 0.80 58.34514
## 12  2.0     15.0      6  110      25      3  1.00 0.50 55.33314
## 13  0.0     16.0      3   95      25      2  1.00 1.00 54.85092
## 14  3.0     17.0      3   90      25      3  1.00 0.25 53.37101
## 15  5.0     13.0      5  190      25      3  1.00 0.67 53.31381
## 16  1.0     16.0      3   55      25      1  1.00 1.00 53.13132
## 17  3.0     15.0      5   85      25      3  1.00 0.88 52.07690
## 18  3.0     17.0      3  110      25      1  1.00 1.00 51.59219
## 19  2.0     17.0      1  105      25      1  1.00 1.25 50.76500
## 20  3.0     17.0      3  115      25      1  1.00 0.67 49.78744
## 21  2.0     14.0      6  110      25      3  1.00 0.50 49.51187
## 22  4.0     15.0      6  125      25      1  1.00 0.67 49.12025
## 23  1.0     21.0      3   30      25      3  1.00 1.00 46.89564
## 24  3.0     16.0      3  110     100      3  1.00 1.00 46.65884
## 25  1.0     21.0      2   35      25      1  1.00 1.00 45.86332
## 26  3.0     13.0      4  100      25      3  1.00 0.33 45.81172
## 27  2.0     12.0      6   95      25      2  1.00 0.67 45.32807
## 28  1.0     18.0      5   80      25      3  1.00 0.75 44.33086
## 29  0.0     23.0      2   30      25      1  1.00 1.13 41.99893
## 30  1.0     20.0      3   45     100      3  1.00 1.00 41.50354
## 31  0.0     22.0      3   25      25      1  1.00 1.00 41.44502
## 32  5.0     14.0     12  190      25      3  1.33 0.67 41.01549
## 33  5.0     12.0     10  200      25      3  1.25 0.67 40.91705
## 34  3.0     21.0      7  130      25      3  1.33 0.67 40.69232
## 35  0.0     22.0      3   35      25      1  1.00 1.00 40.56016
## 36  4.0     10.0      7  160      25      3  1.00 0.50 40.44877
## 37  2.0     13.0      7  105      25      3  1.00 0.50 40.40021
## 38  2.0     15.0      6   90      25      1  1.00 1.00 40.10596
## 39  2.5     10.5      8  140      25      3  1.00 0.50 39.70340
## 40  5.0     14.0     12  240      25      2  1.33 0.75 39.25920
## 41  0.0     21.0      3   40      25      2  1.00 1.50 39.24111
## 42  0.0     21.0      3   60      25      3  1.00 0.75 39.10617
## 43  0.0     21.0      3   35     100      3  1.00 1.00 38.83975
## 44  6.0     11.0     14  260      25      3  1.33 0.67 37.84059
## 45  3.0     16.0     11  170      25      3  1.00 1.00 37.13686
## 46  2.0     18.0      8  100      25      3  1.33 0.75 37.03856
## 47  1.0     17.0      6   60     100      3  1.00 1.00 36.52368
## 48  2.0     20.0      9   95     100      3  1.30 0.75 36.47151
## 49  1.0     16.0      8   60      25      1  1.00 0.75 36.18756
## 50  2.0     11.0     10  120      25      3  1.00 0.75 36.17620
## 51  1.0     13.0     12   20      25      2  1.00 1.00 35.78279
## 52  0.0     11.0     15   40      25      1  1.00 0.88 35.25244
## 53  3.0     16.0     11  170      25      3  1.00 1.00 34.13976
## 54  2.0      8.0      8  135       0      3  1.00 1.00 33.98368
## 55  1.0     11.0     14   30      25      2  1.00 1.00 33.17409
## 56  1.0     11.0     13   30      25      2  1.00 1.00 32.20758
## 57  1.0     14.0     11   25      25      1  1.00 0.75 31.43597
```



Here, we have created a new variable called `double_calories` by using `calories` variable, we multiplied `calories` by 2 and stored in `double_calories`.

---

```
## 58  1.0      9.0    15    40      25    2  1.00 0.75 31.23005
## 59  1.5     11.5    10    90      25    1  1.00 0.75 31.07222
## 60  1.5     13.5    10   120      25    3  1.25 0.50 30.45084
## 61  3.0     17.0    13   160      25    3  1.50 0.67 30.31335
## 62  0.0     15.0     9    40      25    2  1.00 0.67 29.92429
## 63  1.5     10.5    10    70      25    1  1.00 0.75 29.50954
## 64  0.0     14.0    11    35      25    1  1.00 1.33 28.74241
## 65  4.0     15.0    14   230     100    3  1.50 1.00 28.59278
## 66  0.0     13.0    12    25      25    2  1.00 0.75 28.02576
## 67  0.0     13.0    12    25      25    2  1.00 1.00 27.75330
## 68  0.0     12.0    12    55      25    2  1.00 1.00 26.73451
## 69  0.0     15.0     9    45      25    2  1.00 0.75 23.80404
## 70  0.0     12.0    13    55      25    2  1.00 1.00 22.73645
## 71  0.0     12.0    13    65      25    2  1.00 1.00 22.39651
## 72  1.0     12.0    11    45      25    2  1.00 1.00 21.87129
## 73  0.0     13.0     9    45      25    2  1.00 0.75 19.82357
## 74  0.0     12.0    12    35      25    2  1.00 0.75 18.04285
##      double_calories
## 1             100
## 2             180
## 3             180
## 4             140
## 5             160
## 6             100
## 7             100
## 8             180
## 9             140
## 10            180
## 11            200
## 12            180
## 13            200
## 14            220
## 15            180
## 16            220
## 17            200
## 18            200
## 19            220
## 20            200
## 21            200
## 22            180
## 23            220
## 24            200
## 25            200
## 26            240
## 27            200
## 28            200
## 29            220
## 30            200
## 31            220
## 32            240
## 33            240
## 34            280
## 35            220
## 36            220
```

Q13. Create a training set using random number generator engine.

---

```
## 37      220
## 38      200
## 39      200
## 40      240
## 41      220
## 42      220
## 43      220
## 44      240
## 45      300
## 46      260
## 47      220
## 48      280
## 49      220
## 50      200
## 51      220
## 52      200
## 53      300
## 54      240
## 55      220
## 56      220
## 57      220
## 58      220
## 59      220
## 60      260
## 61      320
## 62      240
## 63      220
## 64      220
## 65      280
## 66      220
## 67      220
## 68      220
## 69      220
## 70      220
## 71      220
## 72      240
## 73      240
## 74      240
```

Q13. Create a training set using random number generator engine.

created training and test dataset using random number engine.

```
set.seed(74)
h <- runif(nrow(Cereals))
c <- Cereals[order(h), ]
train <- c[1:45, ]
test <- c[46:74, ]
str(train)
```

```
## 'data.frame': 45 obs. of 16 variables:
## $ Cereals Names: Factor w/ 77 levels "100%_Bran","100%_Natural_Bran",...: 55 77 10 32 12 36 28 74 72
```

Q14.Print the summary statistics of your dataset.

```
## $ mfr      : Factor w/ 7 levels "A","G","K","N",...: 6 2 5 2 2 5 5 2 2 3 ...
## $ type     : Factor w/ 2 levels "C","H": 1 1 1 1 1 1 1 1 1 1 ...
## $ calories : int  50 110 90 110 110 110 120 110 100 110 ...
## $ protein  : int   1 2 3 1 6 1 3 1 3 2 ...
## $ fat      : int   0 1 0 1 2 0 2 1 1 1 ...
## $ sodium   : int   0 200 210 280 290 180 160 140 200 70 ...
## $ fiber    : num   0 1 5 0 2 0 5 0 3 1 ...
## $ carbohydrates: num  13 16 13 15 17 14 12 13 16 9 ...
## $ sugars   : int   0 8 5 9 1 11 10 12 3 15 ...
## $ potass   : int  15 60 190 45 105 35 200 25 110 40 ...
## $ vitamins : int   0 25 25 25 25 25 25 25 100 25 ...
## $ shelf    : int   3 1 3 2 1 1 3 2 3 2 ...
## $ weight   : num   0.5 1 1 1 1 1 1.25 1 1 1 ...
## $ cups     : num   1 0.75 0.67 0.75 1.25 1.33 0.67 1 1 0.75 ...
## $ rating   : num  60.8 36.2 53.3 23.8 50.8 ...
```

```
str(test)
```

```
## 'data.frame': 29 obs. of 16 variables:
## $ Cereals Names: Factor w/ 77 levels "100%_Bran","100%_Natural_Bran",...: 41 71 3 4 37 22 18 39 60 3 ...
## $ mfr          : Factor w/ 7 levels "A","G","K","N",...: 2 2 3 3 6 3 3 3 2 5 ...
## $ type         : Factor w/ 2 levels "C","H": 1 1 1 1 1 1 1 1 1 1 ...
## $ calories     : int  110 140 70 50 120 110 110 110 100 110 ...
## $ protein      : int   2 3 4 4 1 2 1 2 3 3 ...
## $ fat          : int   1 1 1 0 2 0 0 1 2 0 ...
## $ sodium       : int  260 190 260 140 220 220 90 170 140 170 ...
## $ fiber        : num   0 4 9 14 1 1 1 1 2.5 3 ...
## $ carbohydrates: num  21 15 7 8 12 21 13 17 10.5 17 ...
## $ sugars       : int   3 14 5 0 11 3 12 6 8 3 ...
## $ potass       : int  40 230 320 330 45 30 20 60 140 90 ...
## $ vitamins     : int  25 100 25 25 25 25 25 100 25 25 ...
## $ shelf        : int   2 3 3 3 2 3 2 3 3 3 ...
## $ weight       : num   1 1.5 1 1 1 1 1 1 1 1 ...
## $ cups         : num   1.5 1 0.33 0.5 1 1 1 1 0.5 0.25 ...
## $ rating       : num  39.2 28.6 59.4 93.7 21.9 ...
```

Q14.Print the summary statistics of your dataset.

Summary Statistics of dataset.

```
summary(Cereals)
```

```
##           Cereals Names mfr      type      calories      protein
## 100%_Bran           : 1    A: 1    C:73    Min.      : 50    Min.      :1.000
## 100%_Natural_Bran   : 1    G:22    H: 1    1st Qu.:100    1st Qu.:2.000
## All-Bran            : 1    K:23           Median :110    Median :2.500
## All-Bran_with_Extra_Fiber: 1    N: 5           Mean   :107    Mean   :2.514
## Apple_Cinnamon_Cheerios : 1    P: 9           3rd Qu.:110    3rd Qu.:3.000
## Apple_Jacks         : 1    Q: 7           Max.    :160    Max.    :6.000
## (Other)             :68    R: 7
##           fat      sodium      fiber      carbohydrates      sugars
```

Q15. Use any of the numerical variables from the dataset and perform the following statistical functions

---

```
## Min. :0 Min. : 0.0 Min. : 0.000 Min. : 5.00 Min. : 0.000
## 1st Qu.:0 1st Qu.:135.0 1st Qu.: 0.250 1st Qu.:12.00 1st Qu.: 3.000
## Median :1 Median :180.0 Median : 2.000 Median :14.50 Median : 7.000
## Mean :1 Mean :162.4 Mean : 2.176 Mean :14.73 Mean : 7.108
## 3rd Qu.:1 3rd Qu.:217.5 3rd Qu.: 3.000 3rd Qu.:17.00 3rd Qu.:11.000
## Max. :5 Max. :320.0 Max. :14.000 Max. :23.00 Max. :15.000
##
## potass vitamins shelf weight
## Min. : 15.00 Min. : 0.00 Min. :1.000 Min. :0.500
## 1st Qu.: 41.25 1st Qu.: 25.00 1st Qu.:1.250 1st Qu.:1.000
## Median : 90.00 Median : 25.00 Median :2.000 Median :1.000
## Mean : 98.51 Mean : 29.05 Mean :2.216 Mean :1.031
## 3rd Qu.:120.00 3rd Qu.: 25.00 3rd Qu.:3.000 3rd Qu.:1.000
## Max. :330.00 Max. :100.00 Max. :3.000 Max. :1.500
##
## cups rating
## Min. :0.2500 Min. :18.04
## 1st Qu.:0.6700 1st Qu.:32.45
## Median :0.7500 Median :40.25
## Mean :0.8216 Mean :42.37
## 3rd Qu.:1.0000 3rd Qu.:50.52
## Max. :1.5000 Max. :93.70
##
```

Q15. Use any of the numerical variables from the dataset and perform the following statistical functions

we are using protein for our statistical analysis

```
# 1.Mean
cat("The mean value for protein is:", mean(Cereals$protein), "\n");
```

```
## The mean value for protein is: 2.513514
```

```
# 2. Median
cat("The median value for protein is:", median(Cereals$protein), "\n");
```

```
## The median value for protein is: 2.5
```

```
# 3.Mode
# R does not have any in-built function for mode, so we had to create a function for it.
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
v <- Cereals$protein
result <- getmode(v)

cat("The mode value for protein is:", result, "\n");
```

```
## The mode value for protein is: 3
```

Q16. Plot a scatter plot for any 2 variables in your dataset.

---

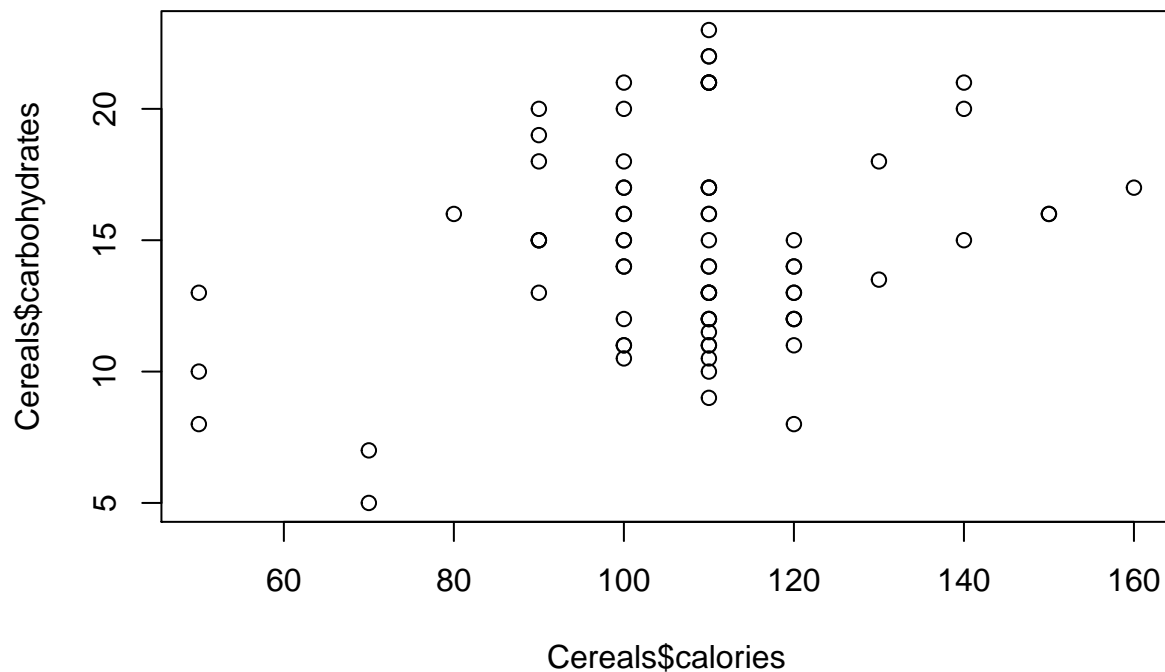
```
# 4.Range  
cat("The range values for protein is:", range(Cereals$protein))
```

```
## The range values for protein is: 1 6
```

Q16. Plot a scatter plot for any 2 variables in your dataset.

The scatterplot for Calories and carbohydrates.

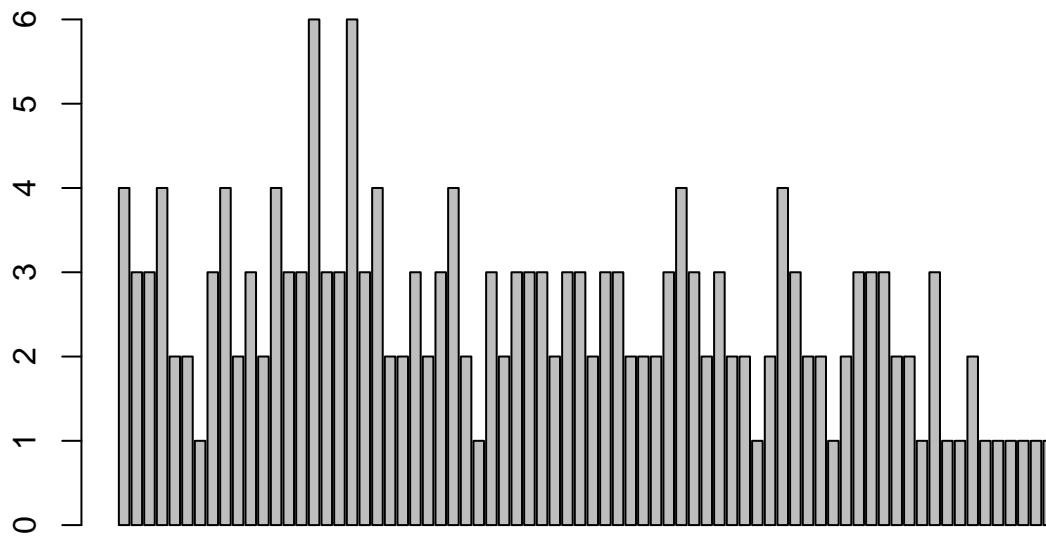
```
plot(Cereals$calories,Cereals$carbohydrates)
```



Q17. Plot a bar plot for any 2 variables in your dataset.

Created 3 different barplots with 3 different variables.

```
barplot(Cereals$protein)
```

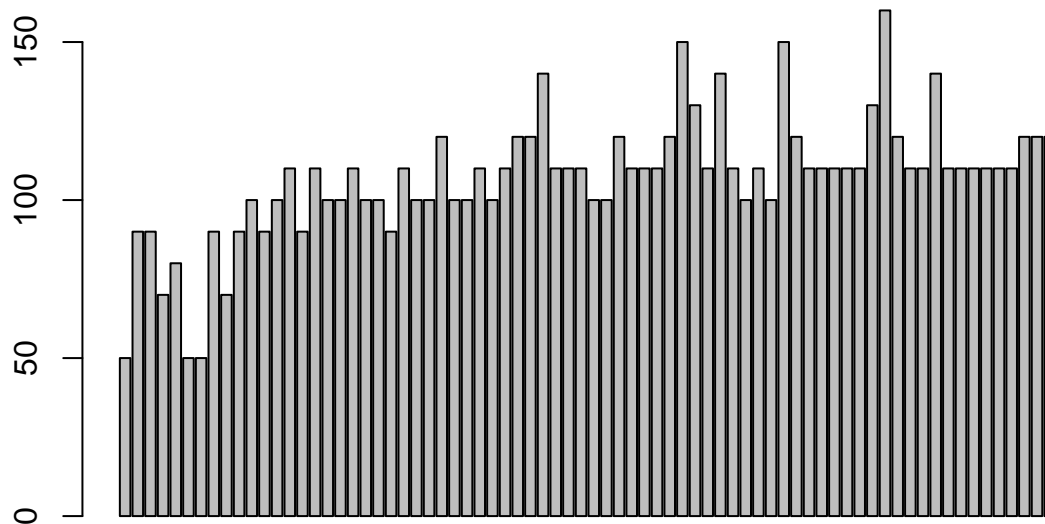


```
barplot(Cereals$fat)
```

```
barplot(Cereals$calories)
```

Q18. Find the correlation between any 2 variables by applying least square linear regression model.

---



Q18. Find the correlation between any 2 variables by applying least square linear regression model.

```
X = Cereals[,"carbohydrates"]
Y = Cereals[,"calories"]
CORRELATION = cor(Y,X,method = "pearson")
CORRELATION
```

```
## [1] 0.270606
```

Q19. Provide a conclusion of your analysis.

Well, the complete analysis has to be done, which will require us to look at all the variables, finding correlations of variables with each other. plotting more graphs. But, with the given analysis, we can say that carbohydrates and calories are somewhat proportional with a positive correlation.