

exercise_1

Fanglin Liu

2020/3/8

exercice_1: lean R using data from R package

```
rm(list = ls())
```

cleaning the current environment

```
library(ade4)
data(package = "ade4")
data("doubts")
str(doubts)
```

```
## List of 4
## $ env      : 'data.frame': 30 obs. of 11 variables:
## ..$ dfs: num [1:30] 3 22 102 185 215 324 268 491 705 990 ...
## ..$ alt: num [1:30] 934 932 914 854 849 846 841 792 752 617 ...
## ..$ slo: num [1:30] 6.18 3.43 3.64 3.5 3.18 ...
## ..$ flo: num [1:30] 84 100 180 253 264 286 400 130 480 1000 ...
## ..$ pH : num [1:30] 79 80 83 80 81 79 81 81 80 77 ...
## ..$ har: num [1:30] 45 40 52 72 84 60 88 94 90 82 ...
## ..$ pho: num [1:30] 1 2 5 10 38 20 7 20 30 6 ...
## ..$ nit: num [1:30] 20 20 22 21 52 15 15 41 82 75 ...
## ..$ amm: num [1:30] 0 10 5 0 20 0 0 12 12 1 ...
## ..$ oxy: num [1:30] 122 103 105 110 80 102 111 70 72 100 ...
## ..$ bdo: num [1:30] 27 19 35 13 62 53 22 81 52 43 ...
## $ fish     : 'data.frame': 30 obs. of 27 variables:
## ..$ Cogo: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Satr: num [1:30] 3 5 5 4 2 3 5 0 0 1 ...
## ..$ Phph: num [1:30] 0 4 5 5 3 4 4 0 1 4 ...
## ..$ Neba: num [1:30] 0 3 5 5 2 5 5 0 3 4 ...
## ..$ Thth: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Teso: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Chna: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Chto: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Lele: num [1:30] 0 0 0 0 5 1 1 0 0 2 ...
## ..$ Lece: num [1:30] 0 0 0 1 2 2 1 0 5 2 ...
## ..$ Baba: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Spbi: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Gogo: num [1:30] 0 0 0 1 2 1 0 0 0 1 ...
## ..$ Eslu: num [1:30] 0 0 1 2 4 1 0 0 0 0 ...
## ..$ Pefl: num [1:30] 0 0 0 2 4 1 0 0 0 0 ...
## ..$ Rham: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Legi: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Scer: num [1:30] 0 0 0 0 2 0 0 0 0 0 ...
## ..$ Cyca: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
```

```
## ..$ Titi: num [1:30] 0 0 0 1 3 2 0 0 1 0 ...
## ..$ Abbr: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Icme: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Acce: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Ruru: num [1:30] 0 0 0 0 5 1 0 0 4 0 ...
## ..$ Blbj: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Alal: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Anan: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## $ xy      : 'data.frame': 30 obs. of 2 variables:
## ..$ x: num [1:30] 88 94 102 100 106 112 114 110 136 168 ...
## ..$ y: num [1:30] 7 14 18 28 39 51 61 76 100 112 ...
## $ species: 'data.frame': 27 obs. of 4 variables:
## ..$ Scientific: chr [1:27] "Cottus gobio" "Salmo trutta fario" "Phoxinus phoxinus" "Nemacheilus ba
## ..$ French      : chr [1:27] "chabot" "truite fario" "vairon" "loche franche" ...
## ..$ English     : chr [1:27] "european bullhead" "brown trout" "minnow" "stone loach" ...
## ..$ code        : Factor w/ 27 levels "Abbr","Acce",...: 9 22 19 17 26 25 7 8 16 14 ...
```

there are 4 list, each with several obs.

```
mydata <- doubs$env
head(mydata)
```

```
##   dfs alt   slo flo pH har pho nit amm oxy bdo
## 1   3 934 6.176 84 79 45   1 20   0 122 27
## 2  22 932 3.434 100 80 40   2 20  10 103 19
## 3 102 914 3.638 180 83 52   5 22   5 105 35
## 4 185 854 3.497 253 80 72  10 21   0 110 13
## 5 215 849 3.178 264 81 84  38 52  20 80 62
## 6 324 846 3.497 286 79 60  20 15   0 102 53
```

check the first 6 lines of the data

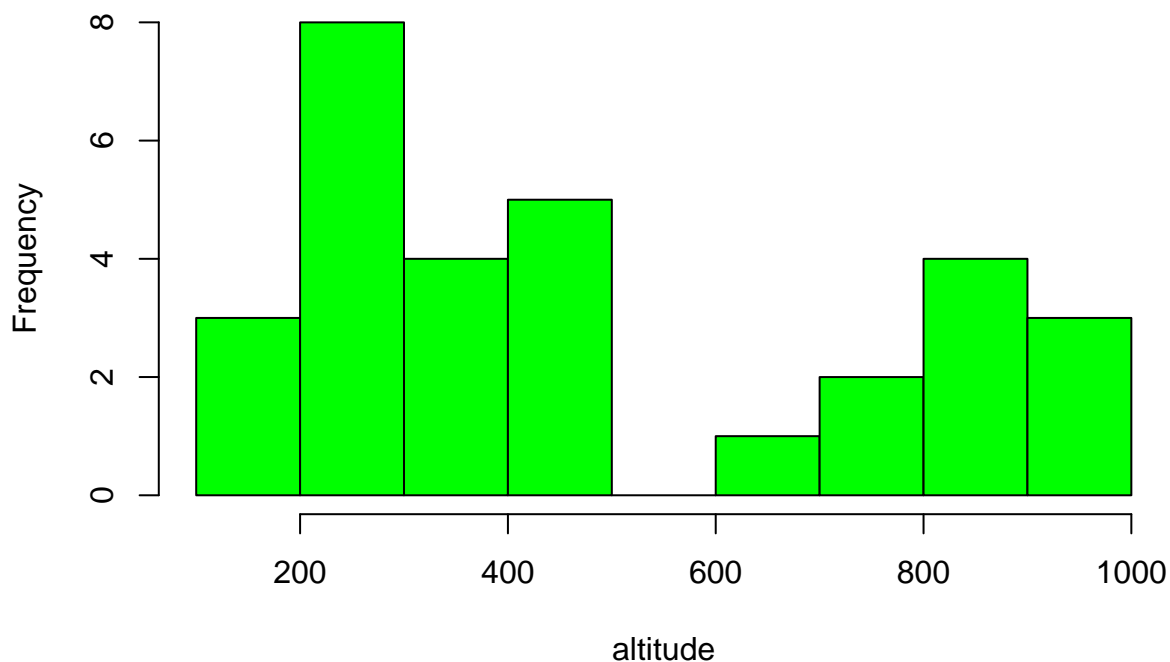
```
mydata <- mydata[,-1]
head(mydata)
```

```
##   alt   slo flo pH har pho nit amm oxy bdo
## 1 934 6.176 84 79 45   1 20   0 122 27
## 2 932 3.434 100 80 40   2 20  10 103 19
## 3 914 3.638 180 83 52   5 22   5 105 35
## 4 854 3.497 253 80 72  10 21   0 110 13
## 5 849 3.178 264 81 84  38 52  20 80 62
## 6 846 3.497 286 79 60  20 15   0 102 53
```

delete the first column of the data

```
hist(mydata$alt,
     col = "green",
     main = "altitude distribution",
     xlab = "altitude")
```

altitude distribution



check the distribution of the variable of altitude

```
library(corrplot)
```

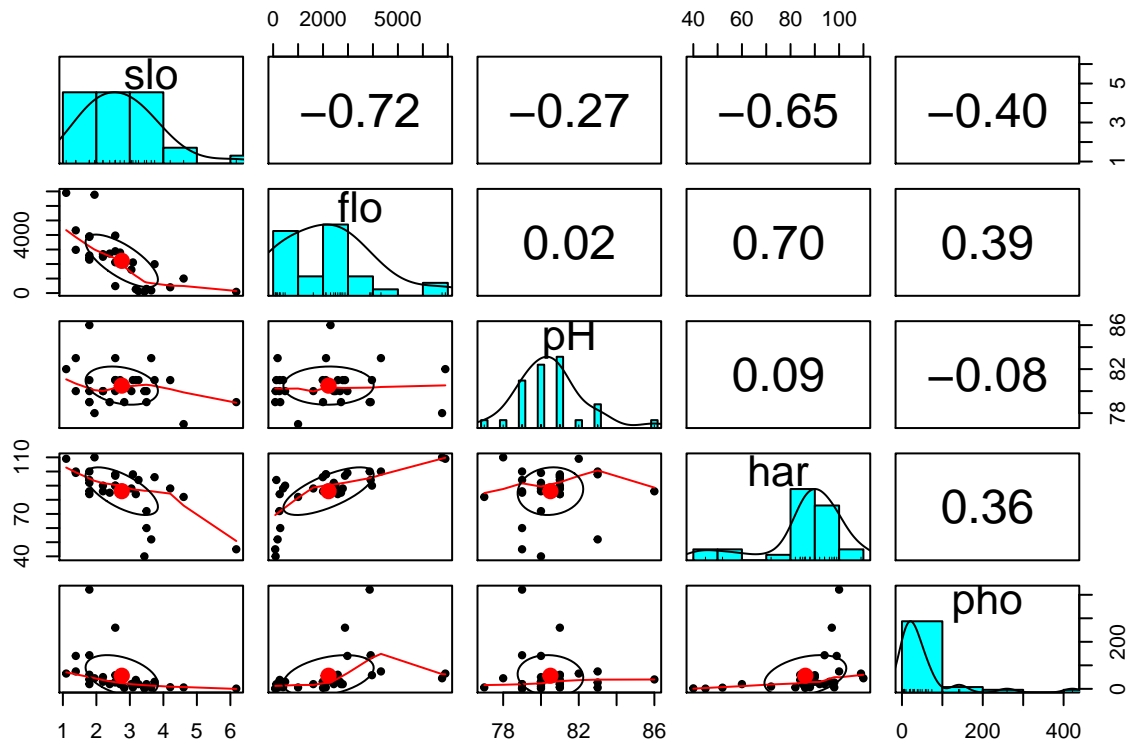
```
## corrplot 0.84 loaded
```

```
res1 <- cor(mydata)
res1
```

```
##          alt      slo      flo      pH      har      pho
## alt  1.0000000  0.7637673 -0.86926914 -0.03726938 -0.74481167 -0.4420491
## slo  0.76376732  1.0000000 -0.71571143 -0.27091451 -0.65375106 -0.4037680
## flo -0.86926914 -0.7157114  1.00000000  0.02042538  0.69678410  0.3852824
## pH  -0.03726938 -0.2709145  0.02042538  1.00000000  0.08886897 -0.0832395
## har -0.74481167 -0.6537511  0.69678410  0.08886897  1.00000000  0.3637981
## pho -0.44204914 -0.4037680  0.38528236 -0.08323950  0.36379811  1.0000000
## nit -0.76054593 -0.6108798  0.60707232 -0.04887849  0.51073526  0.8002507
## amm -0.38132330 -0.3514402  0.29490860 -0.12412055  0.29074449  0.9695215
## oxy  0.36190401  0.4637083 -0.35789468  0.17700293 -0.38239140 -0.7236924
## bdo -0.33784820 -0.3170900  0.25320534 -0.15181290  0.34496636  0.8855369
##          nit      amm      oxy      bdo
## alt -0.76054593 -0.3813233  0.3619040 -0.3378482
## slo -0.61087984 -0.3514402  0.4637083 -0.3170900
## flo  0.60707232  0.2949086 -0.3578947  0.2532053
## pH  -0.04887849 -0.1241205  0.1770029 -0.1518129
## har  0.51073526  0.2907445 -0.3823914  0.3449664
## pho  0.80025065  0.9695215 -0.7236924  0.8855369
## nit  1.00000000  0.7976855 -0.6290729  0.6422816
## amm  0.79768545  1.0000000 -0.7208146  0.8857985
## oxy -0.62907291 -0.7208146  1.0000000 -0.8431211
## bdo  0.64228156  0.8857985 -0.8431211  1.0000000
```

calculate the coefficients of these variables

```
library(psych)
pairs.panels(mydata[,2:6])
```



find which are strongly correlated