ACTIVIDAD 1 UD1 SOLUCIONES

Actividad de tratamiento de datos com e carca de R

Apartado A

Criar un dataframe apartir del vectores

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
x<- c(1,2,3,1,4,5,2)
y<-c(0,3,2,0,5,9,3)
dt= data.frame(x,y)
print(dt)
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
## 4 1 0
## 5 4 5
## 6 5 9
## 7 2 3
```

Elimina los duplicados usando una función de R adecuada help(unique)

```r
help(unique)
```

```
## starting httpd help server ... done
```

```r
dt.uniq<- unique(dt)
dt.uniq
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
## 5 4 5
## 6 5 9
```

Crea una nueva columna Z que sea la suma de las dos primeras dividida por la primera

```
mutate(dt.uniq, z = (dt.uniq$x+dt.uniq$y)/dt.uniq$x) #transformamos una columna
```

```
##   x y        z
## 1 1 0 1.000000
## 2 2 3 2.500000
## 3 3 2 1.666667
## 4 4 5 2.250000
## 5 5 9 2.800000
```

```
dt1.uniq<-dt.uniq
#ou
dt1.uniq$w <- (dt1.uniq$x+dt1.uniq$y)/dt1.uniq$x
```

Cambia el valor X de la provincia BA por 2

```
dt.uniq[4,1:1]<- 2
dt.uniq
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
## 5 2 5
## 6 5 9
```

Selecciona aquellas provincias cuyo valor y sea menor que 4

```
filter(dt.uniq, y <4)
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
```

```
#ou
dt.uniq[dt.uniq$y<4,]
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
```

```r
#ou
subset(dt.uniq,subset=c(y<4))
```

```
##   x y
## 1 1 0
## 2 2 3
## 3 3 2
```

Apartado B

```r
url<-"http://archive.ics.uci.edu/ml/machine-learning-databases/autos/imports-85.data"
dat <- read.csv(url,na.strings = "?", header=FALSE)
View(dat)
```

Haz un sumario del dataframe

```r
summary(dat)
```

```
##       V1                V2              V3              V4          V5
##  Min.   :-2.0000   Min.   : 65    toyota    : 32   diesel: 20   std  :168
##  1st Qu.: 0.0000   1st Qu.: 94    nissan    : 18   gas   :185   turbo: 37
##  Median : 1.0000   Median :115    mazda     : 17
##  Mean   : 0.8341   Mean   :122    honda     : 13
##  3rd Qu.: 2.0000   3rd Qu.:150    mitsubishi: 13
##  Max.   : 3.0000   Max.   :256    subaru    : 12
##                    NA's   :41     (Other)   :100
##     V6               V7          V8          V9            V10
##  four:114    convertible: 6   4wd:  9   front:202    Min.   : 86.60
##  two : 89    hardtop    : 8   fwd:120   rear :  3    1st Qu.: 94.50
##  NA's:  2    hatchback  :70   rwd: 76                Median : 97.00
##              sedan      :96                          Mean   : 98.76
##              wagon      :25                          3rd Qu.:102.40
##                                                      Max.   :120.90
##
##       V11              V12              V13              V14          V15
##  Min.   :141.1    Min.   :60.30    Min.   :47.80    Min.   :1488    dohc : 12
##  1st Qu.:166.3    1st Qu.:64.10    1st Qu.:52.00    1st Qu.:2145    dohcv:  1
##  Median :173.2    Median :65.50    Median :54.10    Median :2414    l    : 12
##  Mean   :174.0    Mean   :65.91    Mean   :53.72    Mean   :2556    ohc  :148
##  3rd Qu.:183.1    3rd Qu.:66.90    3rd Qu.:55.50    3rd Qu.:2935    ohcf : 15
##  Max.   :208.1    Max.   :72.30    Max.   :59.80    Max.   :4066    ohcv : 13
##                                                                     rotor:  4
##      V16             V17            V18            V19             V20
##  eight :  5    Min.   : 61.0   mpfi  :94    Min.   :2.54    Min.   :2.070
##  five  : 11    1st Qu.: 97.0   2bbl  :66    1st Qu.:3.15    1st Qu.:3.110
##  four  :159    Median :120.0   idi   :20    Median :3.31    Median :3.290
##  six   : 24    Mean   :126.9   1bbl  :11    Mean   :3.33    Mean   :3.255
##  three :  1    3rd Qu.:141.0   spdi  : 9    3rd Qu.:3.59    3rd Qu.:3.410
##  twelve:  1    Max.   :326.0   4bbl  : 3    Max.   :3.94    Max.   :4.170
##  two   :  4                    (Other): 2   NA's   :4       NA's   :4
##      V21             V22              V23             V24             V25
##  Min.   : 7.00   Min.   : 48.0    Min.   :4150    Min.   :13.00   Min.   :16.00
```

3

```
##   1st Qu.: 8.60    1st Qu.: 70.0    1st Qu.:4800    1st Qu.:19.00    1st Qu.:25.00
##   Median : 9.00    Median : 95.0    Median :5200    Median :24.00    Median :30.00
##   Mean   :10.14    Mean   :104.3    Mean   :5125    Mean   :25.22    Mean   :30.75
##   3rd Qu.: 9.40    3rd Qu.:116.0    3rd Qu.:5500    3rd Qu.:30.00    3rd Qu.:34.00
##   Max.   :23.00    Max.   :288.0    Max.   :6600    Max.   :49.00    Max.   :54.00
##                    NA's   :2        NA's   :2
##        V26
##   Min.   : 5118
##   1st Qu.: 7775
##   Median :10295
##   Mean   :13207
##   3rd Qu.:16500
##   Max.   :45400
##   NA's   :4
```

¿Qué dimensión tiene el dataframe?

```
lapply(dat,class)
```

```
## $V1
## [1] "integer"
##
## $V2
## [1] "integer"
##
## $V3
## [1] "factor"
##
## $V4
## [1] "factor"
##
## $V5
## [1] "factor"
##
## $V6
## [1] "factor"
##
## $V7
## [1] "factor"
##
## $V8
## [1] "factor"
##
## $V9
## [1] "factor"
##
## $V10
## [1] "numeric"
##
## $V11
## [1] "numeric"
##
## $V12
```

```
## [1] "numeric"
##
## $V13
## [1] "numeric"
##
## $V14
## [1] "integer"
##
## $V15
## [1] "factor"
##
## $V16
## [1] "factor"
##
## $V17
## [1] "integer"
##
## $V18
## [1] "factor"
##
## $V19
## [1] "numeric"
##
## $V20
## [1] "numeric"
##
## $V21
## [1] "numeric"
##
## $V22
## [1] "integer"
##
## $V23
## [1] "integer"
##
## $V24
## [1] "integer"
##
## $V25
## [1] "integer"
##
## $V26
## [1] "integer"
```

```r
#class(dat)
#str(dat)
factores<- which(sapply(dat,class)=="factor")
factores
```

```
##  V3  V4  V5  V6  V7  V8  V9 V15 V16 V18
##   3   4   5   6   7   8   9  15  16  18
```

```r
for (i in 1:length(factores)){
  print(unique(dat[,i]))
}
```

```
## [1]  3  1  2  0 -1 -2
##  [1]   NA 164 158 192 188 121  98  81 118 148 110 145 137 101  78 106  85 107 104
## [20] 113 150 129 115  93 142 161 153 125 128 122 103 168 108 194 231 119 154  74
## [39] 186  83 102  89  87  77  91 134  65 197  90  94 256  95
##  [1] alfa-romero   audi          bmw           chevrolet     dodge
##  [6] honda         isuzu         jaguar        mazda         mercedes-benz
## [11] mercury       mitsubishi    nissan        peugot        plymouth
## [16] porsche       renault       saab          subaru        toyota
## [21] volkswagen    volvo
## 22 Levels: alfa-romero audi bmw chevrolet dodge honda isuzu jaguar ... volvo
## [1] gas    diesel
## Levels: diesel gas
## [1] std    turbo
## Levels: std turbo
## [1] two  four <NA>
## Levels: four two
## [1] convertible hatchback   sedan       wagon       hardtop
## Levels: convertible hardtop hatchback sedan wagon
## [1] rwd fwd 4wd
## Levels: 4wd fwd rwd
## [1] front rear
## Levels: front rear
##  [1]  88.6  94.5  99.8  99.4 105.8  99.5 101.2 103.5 110.0  88.4  93.7 103.3
## [13]  95.9  86.6  96.5  94.3  96.0 113.0 102.0  93.1  95.3  98.8 104.9 106.7
## [25] 115.6  96.6 120.9 112.0 102.7  93.0  96.3  95.1  97.2 100.4  91.3  99.2
## [37] 107.9 114.2 108.0  89.5  98.4  96.1  99.1  93.3  97.0  96.9  95.7 102.4
## [49] 102.9 104.5  97.3 104.3 109.1
```

Indica la proporción de valores faltantes por columna usando sum e is.na y un apply (usa nrow)

```r
apply(is.na(dat),2,sum)/nrow(dat)*100
```

```
##         V1         V2         V3         V4         V5         V6         V7
##  0.0000000 20.0000000  0.0000000  0.0000000  0.0000000  0.9756098  0.0000000
##         V8         V9        V10        V11        V12        V13        V14
##  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
##        V15        V16        V17        V18        V19        V20        V21
##  0.0000000  0.0000000  0.0000000  0.0000000  1.9512195  1.9512195  0.0000000
##        V22        V23        V24        V25        V26
##  0.9756098  0.9756098  0.0000000  0.0000000  1.9512195
```

Inspecciona los datos, ¿de qué clase es cada tipo? Extrae los distintos valores que toman los factores

```r
dim(dat)
```

```
## [1] 205  26
```

Da una media de los valores de la primera columna respecto a la columna de número de puertas del coche

```
tapply(dat$V1, dat$V6, mean)
```

```
##      four      two
## 0.1052632 1.7752809
```

Extrae los coches audi que son cuatro puertas con valor V13 menor que 55

```
dtf<-filter(dat,V3=='audi'& V6=="four" & V13 < 55)
View(dtf)
# Ou
dat[dat$V3=='audi' & dat$V6=='four'& dat$V13< 55,]
```

```
##   V1  V2   V3  V4  V5   V6    V7    V8    V9  V10   V11   V12  V13  V14 V15  V16
## 4  2 164 audi gas std four sedan fwd front 99.8 176.6 66.2 54.3 2337 ohc four
## 5  2 164 audi gas std four sedan 4wd front 99.4 176.6 66.4 54.3 2824 ohc five
##   V17  V18  V19 V20 V21 V22  V23 V24 V25   V26
## 4 109 mpfi 3.19 3.4  10 102 5500  24  30 13950
## 5 136 mpfi 3.19 3.4   8 115 5500  18  22 17450
```

FIM

Por: Adilson Varela