R Notebook

GUOTAI SUN

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

```
#install.packages("beepr")
library(beepr)
```

```
## Warning: package 'beepr' was built under R version 4.0.5
```

```
#Or library ("beepr"), function library is used to load the package bee
pr.

#Beep(x), put x to be any int from 1 to 11, it will play a sound

#beepr:: declares the beep() function belongs to the beepr package. You
 don't have to do that all the time especially when your code is pretty
 simple.

#Basic data types
x = 1:10
print(x)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
typeof(x)
```

```
## [1] "integer"
```

```
length(x)
```

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

```
characterData <- c("East","West","East","North","North","East","West","
West","West","East","North")
typeof(characterData)
```

```
## [1] "character"
```

```
print(characterData)
```

```
##  [1] "East"  "West"  "East"  "North" "North" "East"  "West"  "West"
 "West"
## [10] "East"  "North"
```

```r
factor_data <- factor(characterData)
typeof(factor_data)
```

```
## [1] "integer"
```

```r
print(factor_data)
```

```
##  [1] East  West  East  North North East  West  West  West  East  Nor
th
## Levels: East North West
```

```r
#Data structures
v <- c(TRUE, TRUE, FALSE, FALSE)
typeof(v)
```

```
## [1] "logical"
```

```r
print(v)
```

```
## [1]  TRUE  TRUE FALSE FALSE
```

```r
v <- c("1", 1, TRUE)
typeof(v)
```

```
## [1] "character"
```

```r
print(v)
```

```
## [1] "1"    "1"    "TRUE"
```

```r
l <- list("1", 1, TRUE)
typeof(l)
```

```
## [1] "list"
```

```r
print(1)
```

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

```r
#Matrix and data frame
m <- matrix(nrow = 3, ncol = 3)
show(m)
```

```
##      [,1] [,2] [,3]
## [1,]   NA   NA   NA
## [2,]   NA   NA   NA
## [3,]   NA   NA   NA
```

```r
FOURS <- matrix( # Or something like this. Tip: You don't have to write
 R script in one single line. Sometimes, breaking it will increase the
readability.
  c(2, 4, 5, 7), # This is all the data you want to put into the matrix
  nrow = 2, # This tells R the number of rows
```

```
  ncol = 2) # This tells R the number of columns
FOURS
```

```
##      [,1] [,2]
## [1,]    2    5
## [2,]    4    7
```

```
dim(FOURS)
```

```
## [1] 2 2
```

```
class(FOURS)
```

```
## [1] "matrix" "array"
```

```
typeof(FOURS)
```

```
## [1] "double"
```

```
id <- letters[1:10] #creating the first 10 alphabetic letters
score <- 1:10 # a numeric vector with 10 elements from 1 to 10.
enrolled <- logical(10) # a logical vector with 10 elements. All false

df <- data.frame(id, score, enrolled)
df
```

```
##    id score enrolled
## 1   a     1    FALSE
## 2   b     2    FALSE
## 3   c     3    FALSE
## 4   d     4    FALSE
## 5   e     5    FALSE
## 6   f     6    FALSE
## 7   g     7    FALSE
## 8   h     8    FALSE
## 9   i     9    FALSE
## 10  j    10    FALSE
```

```
data("mtcars")
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
data("iris")
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
#Tidy data
#install.packages("tidyverse")
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyve
rse 1.3.1 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_co
nflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
A = read_csv("ICE1_Data.csv")
```

```
##
## -- Column specification --------------------------------------------
------------
## cols(
##   DBN = col_character(),
##   Quality_Review_Score = col_character(),
##   `Progress_Rpt_10-11` = col_character(),
##   `Student_Progress_10-11` = col_character(),
##   `graduation 2010-11` = col_double(),
##   `college enroll 2010-11` = col_double()
## )
```

```
show(A)
```

```
## # A tibble: 422 x 6
##    DBN    Quality_Review_S~ `Progress_Rpt_10~ `Student_Progres~ `gra
duation 201~
```

```
##    <chr>  <chr>             <chr>             <chr>
       <dbl>
##  1 01M292 Developing        C                 C
       0.563
##  2 01M448 Developing        C                 B
       0.707
##  3 01M450 Well Developed    A                 B
       0.716
##  4 01M509 Proficient        C                 C
       0.564
##  5 01M539 Proficient        A                 A
       0.953
##  6 01M696 Well Developed    B                 C
       0.976
##  7 02M047 Proficient        C                 D
       0.696
##  8 02M288 Proficient        A                 B
       0.82
##  9 02M294 Well Developed    B                 B
       0.675
## 10 02M296 Proficient        A                 A
       0.793
## # ... with 412 more rows, and 1 more variable: college enroll 2010-1
1 <dbl>
```

```
summary(A)
```

```
##      DBN            Quality_Review_Score Progress_Rpt_10-11
##  Length:422         Length:422           Length:422
##  Class :character   Class :character     Class :character
##  Mode  :character   Mode  :character     Mode  :character
##
##
##
##
##  Student_Progress_10-11 graduation 2010-11 college enroll 2010-11
##  Length:422             Min.   :0.4120     Min.   :0.1410
##  Class :character       1st Qu.:0.6252     1st Qu.:0.3910
##  Mode  :character       Median :0.7260     Median :0.4890
##                         Mean   :0.7380     Mean   :0.5312
##                         3rd Qu.:0.8508     3rd Qu.:0.6560
##                         Max.   :1.0000     Max.   :1.0000
##                         NA's   :112        NA's   :131
```

```
A_factor = A %>%
  mutate_if(sapply(A, is.character), as.factor)
summary(A_factor)
```

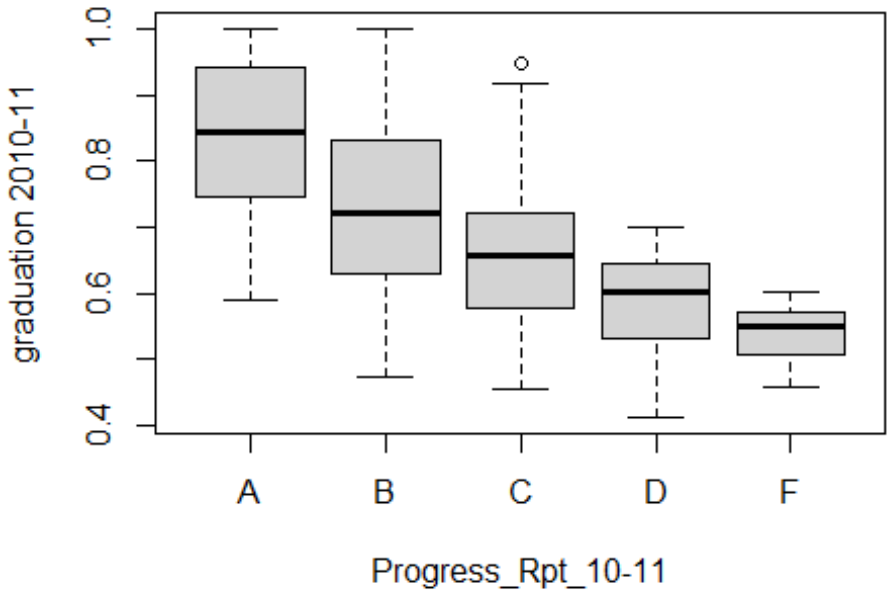
```
##      DBN
##  01M292 :  1
##  01M448 :  1
```

```
##  01M450 :  1
##  01M509 :  1
##  01M539 :  1
##  01M696 :  1
##  (Other):416
##
         Quality_Review_Score
##  Developing
                   : 64
##  Outstanding (only an option in 2007-8)
                   :  3
##  Proficient
                   :186
##  Underdeveloped
                   :  6
##  Underdeveloped with Proficient Features (only an option in 2007-8,
2008-9 and 2009-10):  1
##  Well Developed
                   :108
##  NA's
                   : 54
##  Progress_Rpt_10-11 Student_Progress_10-11 graduation 2010-11
##  A   :109           A   : 76               Min.   :0.4120
##  B   :102           B   : 91               1st Qu.:0.6252
##  C   : 72           C   : 93               Median :0.7260
##  D   : 22           D   : 35               Mean   :0.7380
##  F   :  5           F   : 15               3rd Qu.:0.8508
##  NA's:112           NA's:112               Max.   :1.0000
##                                            NA's   :112
##  college enroll 2010-11
##  Min.   :0.1410
##  1st Qu.:0.3910
##  Median :0.4890
##  Mean   :0.5312
##  3rd Qu.:0.6560
##  Max.   :1.0000
##  NA's   :131
```

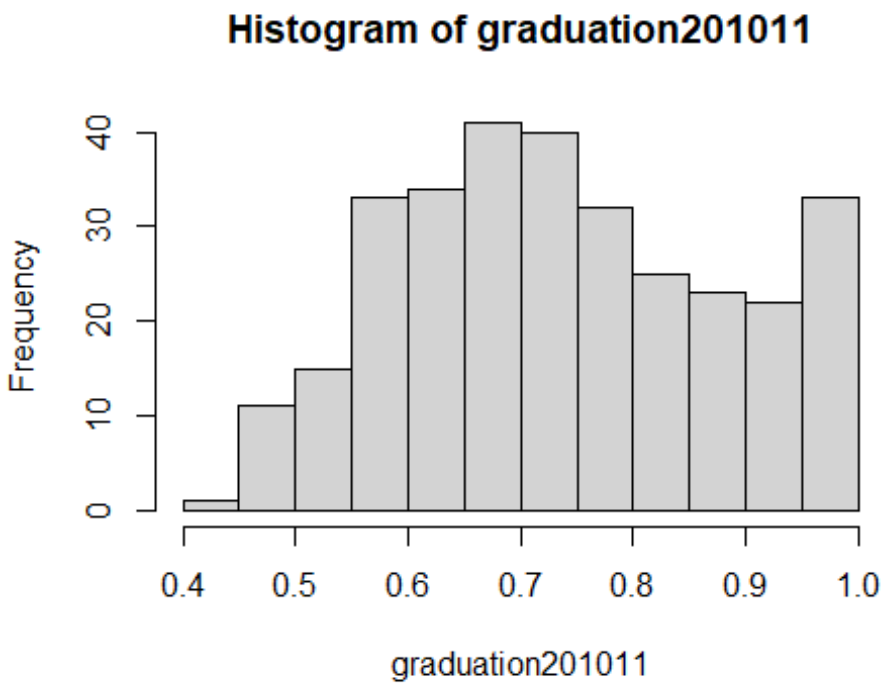
```
SPG = A_factor %>%
  select(`Progress_Rpt_10-11`, `graduation 2010-11`)
show(SPG)
```

```
## # A tibble: 422 x 2
##    `Progress_Rpt_10-11` `graduation 2010-11`
##    <fct>                                <dbl>
##  1 C                                    0.563
##  2 C                                    0.707
##  3 A                                    0.716
##  4 C                                    0.564
##  5 A                                    0.953
```

```
##  6 B                                          0.976
##  7 C                                          0.696
##  8 A                                          0.82
##  9 B                                          0.675
## 10 A                                          0.793
## # ... with 412 more rows

#This is a typical tidyverse syntax. tidyverse uses %>% to indicate pip
elines between commands.

graduation201011 <- A$`graduation 2010-11`
head(graduation201011)

## [1] 0.563 0.707 0.716 0.564 0.953 0.976

#Type ICEdata followed by a $, you will receive prompts from R to selec
t variable.
length(graduation201011)

## [1] 422

#plot something
plot(SPG)
```




```
hist(graduation201011)
```



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.