

Analyzing Survey Data in R

Importing a .csv file directly from the web

importing dataset

```
data <-
```

```
read.csv("https://www.consumerfinance.gov/documents/5614/NFWBS_PUF_2016_data.csv")
```

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for installing the `dplyr` package and importing the dataset from a web URL.
- Environment:** Shows the `data` object with 6394 observations and 217 variables.
- Console:** Displays the execution output, including an error message for `install.packages` and the successful installation of `dplyr`.

```
1 #Installs the package
2 install.packages("dplyr")
3
4 # importing dataset
5
6 data <- read.csv("https://www.consumerfinance.gov/documents/5614/NFWBS_PUF_2016_data.csv")
7
8
```

Environment

Object	Size
data	6394 obs. of 217 variables

Console

```
> data <- read.csv("https://www.consumerfinance.gov/documents/5614/NFWBS_PUF_2016_data.csv")
> #Installs the package
> install.packages("dplyr")
Error in install.packages : Updating loaded packages

Restarting R session...

> install.packages("dplyr")
trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.4/dplyr_1.1.4.tgz'
Content type 'application/x-gzip' length 1599250 bytes (1.5 MB)
downloaded 1.5 MB

The downloaded binary packages are in
/var/folders/kw/j_bzb8c138j7gx3bwplfyhr0000gn/T//Rtmpqxv5qT/downloaded_packages
> |
```

Creating a subset

#Gets the <\$50k income subset

```
income50k <- data %>% filter(PPINCIMP<=4)
```

The screenshot displays the RStudio environment with the following components:

- Script Editor:** Contains R code for installing and loading the `dplyr` package, reading a CSV file, and creating a subset of data.
- Environment:** Shows the global environment with two objects: `data` (6394 obs. of 217 variables) and `income50k` (2306 obs. of 217 variables).
- Console:** Displays the output of the R script, including package installation paths, package attachment, and masked objects.

```
1 #Installs the package
2 install.packages("dplyr")
3 library(dplyr) #Loads the package
4
5 # importing dataset
6
7 data <- read.csv("https://www.consumerfinance.gov/documents/5614/NFWS_PUF_2016_data.csv")
8
9
10 #Gets the <$50k income subset
11 income50k <- data %>% filter(PPINCIMP<=4)
12
13
```

Environment:

Object	Obs.	Variables
data	6394	217
income50k	2306	217

Console:

```
The downloaded binary packages are in
/var/folders/kw/j_bzb8c138j7gx3bwpdlfyhr0000gn/T//Rtmpqxv5qT/downloaded_packages
> library(dplyr) #Loads the package

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

> #Gets the <$50k income subset
> income50k <- data %>% filter(PPINCIMP<=4)
>
```

Selecting specific columns

```
sp_column <- income50k %>% select(PPGENDER,PPHHSIZE,PPINCIMP,FWBscore,finalwt)
```

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for installing the `dplyr` package, loading it, reading a CSV file from a URL, filtering for income less than or equal to \$50k, and selecting specific columns.
- Environment Pane:** Shows the current environment with three objects: `data` (6394 obs. of 217 variables), `income50k` (2306 obs. of 217 variables), and `sp_column` (2306 obs. of 5 variables).
- Console:** Displays the execution output, including warnings about masked objects and the results of the data manipulation steps.

```
1 #Installs the package
2 install.packages("dplyr")
3 library(dplyr) #Loads the package
4
5 # importing dataset
6
7
8 data <- read.csv("https://www.consumerfinance.gov/documents/5614/NFWBS_PUF_2016_data.csv")
9
10
11 #Gets the <$50k income subset
12 income50k <- data %>% filter(PPINCIMP<=4)
13 table(income50k$PPINCIMP)
14
15
16 #Selecting specific columns
17 sp_column <- income50k %>% select(PPGENDER,PPHHSIZE,PPINCIMP,FWBscore,finalwt)
18
```

Console Output:

```
R - R 4.4.2 - ~/
The following objects are masked from 'package:stats':
  filter, lag
The following objects are masked from 'package:base':
  intersect, setdiff, setequal, union
> #Gets the <$50k income subset
> income50k <- data %>% filter(PPINCIMP<=4)
> table(income50k$PPINCIMP)
 1  2  3  4
719 506 614 467
> #Selecting specific columns
> sp_column <- income50k %>% select(PPGENDER,PPHHSIZE,PPINCIMP,FWBscore,finalwt)
>
```

Creating a binary (a.k.a dummy) variable

```
table(data$PPEDUC)
```

```
#COLLEGE takes a value of 1 if PPEDUC >= 4; COLLEGE takes a value of 0 if PPEDUC < 4.  
data$COLLEGE<- ifelse(data$PPEDUC>=4,1,0)
```

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for importing a dataset, filtering for income50k, selecting specific columns, and creating a binary variable COLLEGE based on PPEDUC.
- Environment:** Shows the Global Environment with three objects: data (6394 obs. of 218 variables), income50k (2306 obs. of 217 variables), and sp_column (2306 obs. of 5 variables).
- Console:** Displays the execution of the code, including the output of the table function for income50k\$PPINCIMP and data\$PPEDUC.

```
# importing dataset  
data <- read.csv("https://www.consumerfinance.gov/documents/5614/NFWS_PUF_2016_data.csv")  
  
#Gets the <$50k income subset  
income50k <- data %>% filter(PPINCIMP<=4)  
table(income50k$PPINCIMP)  
  
#Selecting specific columns  
sp_column <- income50k %>% select(PPGENDER,PPHHSIZE,PPINCIMP,FWBscore,finalwt)  
  
#Creating a binary (a.k.a dummy) variable  
table(data$PPEDUC)  
  
#COLLEGE takes a value of 1 if PPEDUC >= 4; COLLEGE takes a value of 0 if PPEDUC < 4.  
data$COLLEGE<- ifelse(data$PPEDUC>=4,1,0)
```

Console Output:

```
The following objects are masked from 'package:base':  
intersect, setdiff, setequal, union  
  
> #Gets the <$50k income subset  
> income50k <- data %>% filter(PPINCIMP<=4)  
> table(income50k$PPINCIMP)  
  
 1  2  3  4  
719 506 614 467  
> #Selecting specific columns  
> sp_column <- income50k %>% select(PPGENDER,PPHHSIZE,PPINCIMP,FWBscore,finalwt)  
> View(sp_column)  
> #Creating a binary (a.k.a dummy) variable  
> table(data$PPEDUC)  
  
 1  2  3  4  5  
429 1622 1933 1312 1098
```

Renaming the levels of a categorical variable

#Renaming the levels of a categorical variable

```
data$PPINCIMP <- recode(data$PPINCIMP,"1"="Less than $20,000",
  "2"="$20,000 to $29,999",
  "3"="$30,000 to $39,999",
  "4"="$40,000 to $49,999",
  "5"="$50,000 to $59,999",
  "6"="$60,000 to $74,999",
  "7"="$75,000 to $99,999",
  "8"="$100,000 to $149,999",
  "9"="$150,000 or more")
table(data$PPINCIMP)
```

The screenshot shows an RStudio interface with a script editor and a console. The script editor contains R code for creating a binary variable, recoding income levels, and displaying a table. The console shows the execution of the code, including the recoding process and the resulting frequency table for the income variable.

```
20
21
22 #Creating a binary (a.k.a dummy) variable
23 table(data$PPEDUC)
24
25
26 #COLLEGE takes a value of 1 if PPEDUC >= 4; COLLEGE takes a value of 0 if PPEDUC < 4.
27 data$COLLEGE<- ifelse(data$PPEDUC>=4,1,0)
28
29
30
31 #Renaming the levels of a categorical variable
32 data$PPINCIMP <- recode(data$PPINCIMP,"1"="Less than $20,000",
33   "2"="$20,000 to $29,999",
34   "3"="$30,000 to $39,999",
35   "4"="$40,000 to $49,999",
36   "5"="$50,000 to $59,999",
37   "6"="$60,000 to $74,999",
38   "7"="$75,000 to $99,999",
39   "8"="$100,000 to $149,999",
40   "9"="$150,000 or more")
41 table(data$PPINCIMP)
42
43
44
```

36:51 (Top Level) R Script

```
R - R 4.4.2 ~ /
> data$PPINCIMP <- recode(data$PPINCIMP, 1 = "Less than $20,000",
+   "2"="$20,000 to $29,999",
+   "3"="$30,000 to $39,999",
+   "4"="$40,000 to $49,999",
+   "5"="$50,000 to $59,999",
+   "6"="$60,000 to $74,999",
+   "7"="$75,000 to $99,999",
+   "8"="$100,000 to $149,999",
+   "9"="$150,000 or more")
> table(data$PPINCIMP)

$100,000 to $149,999  $150,000 or more  $20,000 to $29,999  $30,000 to $39,999
1115                862                506                614
$40,000 to $49,999  $50,000 to $59,999  $60,000 to $74,999  $75,000 to $99,999
467                  505                  651                  955
Less than $20,000
719
```

The screenshot shows the RStudio Environment and Files panels. The Environment panel displays the global environment with three objects: 'data' (6394 obs. of 218 variables), 'income50k' (2306 obs. of 217 variables), and 'sp_column' (2306 obs. of 5 variables). The Files panel shows the current directory structure with subfolders for 'data', 'plots', 'packages', 'help', 'viewer', and 'presentation'.

Environment History Connections Tutorial

R - Global Environment

Data

data	6394 obs. of 218 variables
income50k	2306 obs. of 217 variables
sp_column	2306 obs. of 5 variables

Files Plots Packages Help Viewer Presentation

Zoom Export

Creating a new categorical variable

```
data$GENERATION.GENDER <- ifelse(data$PPGENDER==1 & data$generation==1, 'Male,
Pre-Boomer',
                                ifelse(data$PPGENDER==1 & data$generation==2, 'Male, Boomer',
                                ifelse(data$PPGENDER==1 & data$generation==3, 'Male, Gen X',
                                ifelse(data$PPGENDER==1 & data$generation==4, 'Male,
Millennial',
                                ifelse(data$PPGENDER==2 & data$generation==1, 'Female,
Pre-Boomer',
                                ifelse(data$PPGENDER==2 & data$generation==2,
'Female, Boomer',
                                ifelse(data$PPGENDER==2 & data$generation==3,
'Female, Gen X',
                                'Female, Millennial'))))))))
```

#I have data\$GENERATION.GENDER at the start of the code because I made a new variable called GENERATION.GENDER."

We write ifelse() conditions seven times since the intersection of the two variables (generation and PPGENDER) has eight categories.

The last category does not require a condition because a response will be placed in the final group ('Female, Millennial') if the seven conditions, which pertain to seven categories, do not match.

```
table(data$GENERATION.GENDER)
```

Untitled1*
Untitled2*
Untitled3*
Untitled4*
sp_column
iris

Source on Save

Run

Source

```
40 table(data$PPGENDER)
41
42
43 # Creating a new categorical variable
44 table(data$PPGENDER)
45 table(data$PPGENDER)
46
47
48
49
50
51 data$GENERATION.GENDER <- ifelse(data$PPGENDER==1 & data$generation==1, 'Male, Pre-Boomer',
52                                ifelse(data$PPGENDER==1 & data$generation==2, 'Male, Boomer',
53                                ifelse(data$PPGENDER==1 & data$generation==3, 'Male, Gen X',
54                                ifelse(data$PPGENDER==1 & data$generation==4, 'Male,
55                                ifelse(data$PPGENDER==2 & data$generation==1,
56                                ifelse(data$PPGENDER==2 & data$generation==2, 'Female, Millennial')))))))
57
58
59
60
61 table(data$GENERATION.GENDER)
62
63
64
```

62:1 (Top Level) R Script

Console

Terminal

Background Jobs

```
R - R 4.4.2 ~ /
+ ifelse(data$PPGENDER==2 & data$generation==2, 'Male, Boomer',
+ ifelse(data$PPGENDER==1 & data$generation==3, 'Male, Gen X',
+ ifelse(data$PPGENDER==1 & data$generation==4, 'Male, Mil
+ lennial',
+ ifelse(data$PPGENDER==2 & data$generation==1, 'Fe
+ male, Pre-Boomer',
+ ifelse(data$PPGENDER==2 & data$generation==
+ =2, 'Female, Boomer',
+ ifelse(data$PPGENDER==2 & data$gene
+ ration==3, 'Female, Gen X',
+ 'Female, Millennial')))))))
> table(data$GENERATION.GENDER)

      Female, Boomer      Female, Gen X Female, Millennial Female, Pre-Boomer      Male, Boomer
      1132             596             765             549             1121
      Male, Gen X      Male, Millennial      Male, Pre-Boomer
      834             834             563
```

Environment

History

Connections

Tutorial

Import Dataset

42 MiB

List

R - Global Environment

data

6394 obs. of 219 variables

income50k

2306 obs. of 217 variables

sp_column

2306 obs. of 5 variables

Files

Plots

Packages

Help

Viewer

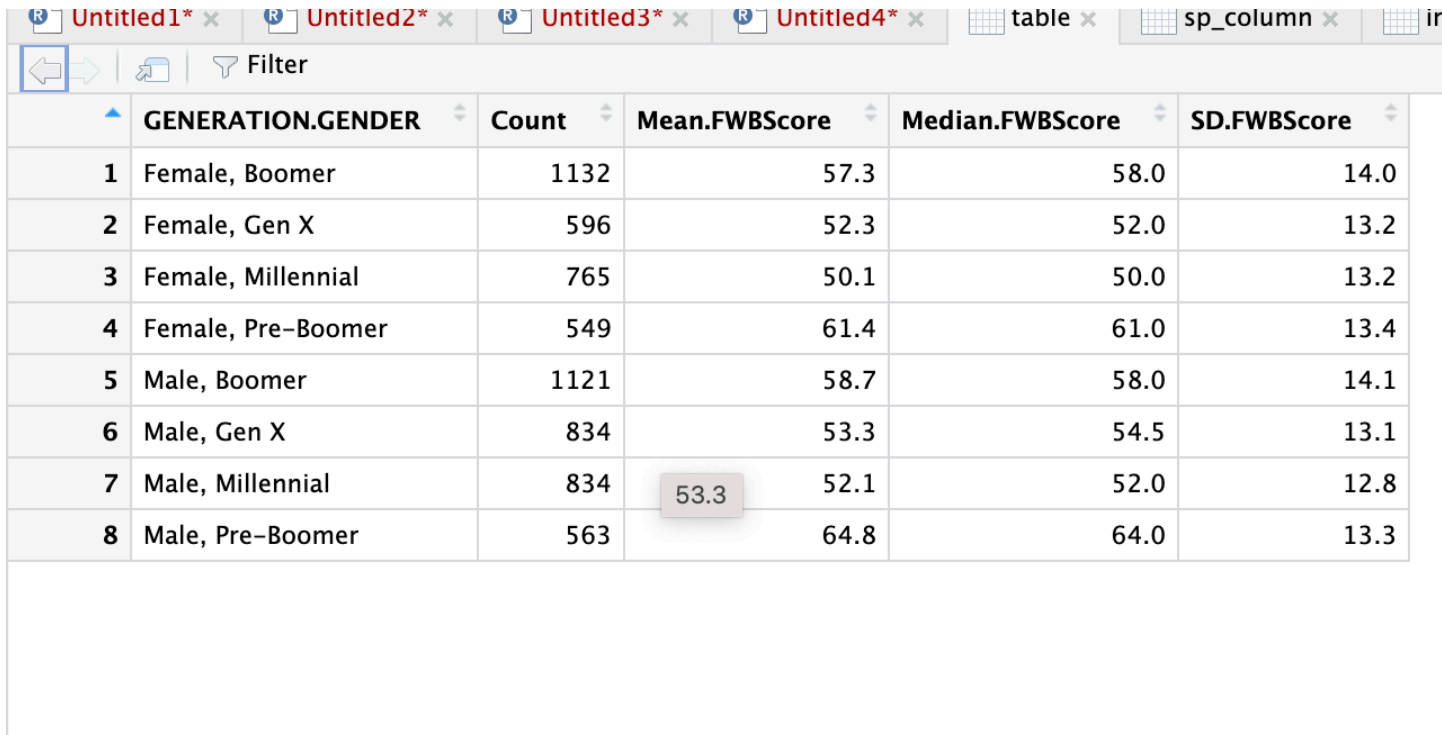
Presentation

Zoom

Export

Creating a summary statistics table

```
table <- data %>% group_by(GENERATION.GENDER) %>%  
  summarise(Count=n(),  
            Mean.FWBScore=round(mean(FWBScore),digits=1),  
            Median.FWBScore=round(median(FWBScore),digits=1),  
            SD.FWBScore=round(sd(FWBScore),digits=1)  
  )
```



The screenshot shows an RStudio window with four untitled R scripts. The active script displays the R code for creating a summary statistics table. Below the code, a table is shown with the following data:

	GENERATION.GENDER	Count	Mean.FWBScore	Median.FWBScore	SD.FWBScore
1	Female, Boomer	1132	57.3	58.0	14.0
2	Female, Gen X	596	52.3	52.0	13.2
3	Female, Millennial	765	50.1	50.0	13.2
4	Female, Pre-Boomer	549	61.4	61.0	13.4
5	Male, Boomer	1121	58.7	58.0	14.1
6	Male, Gen X	834	53.3	54.5	13.1
7	Male, Millennial	834	52.1	52.0	12.8
8	Male, Pre-Boomer	563	64.8	64.0	13.3

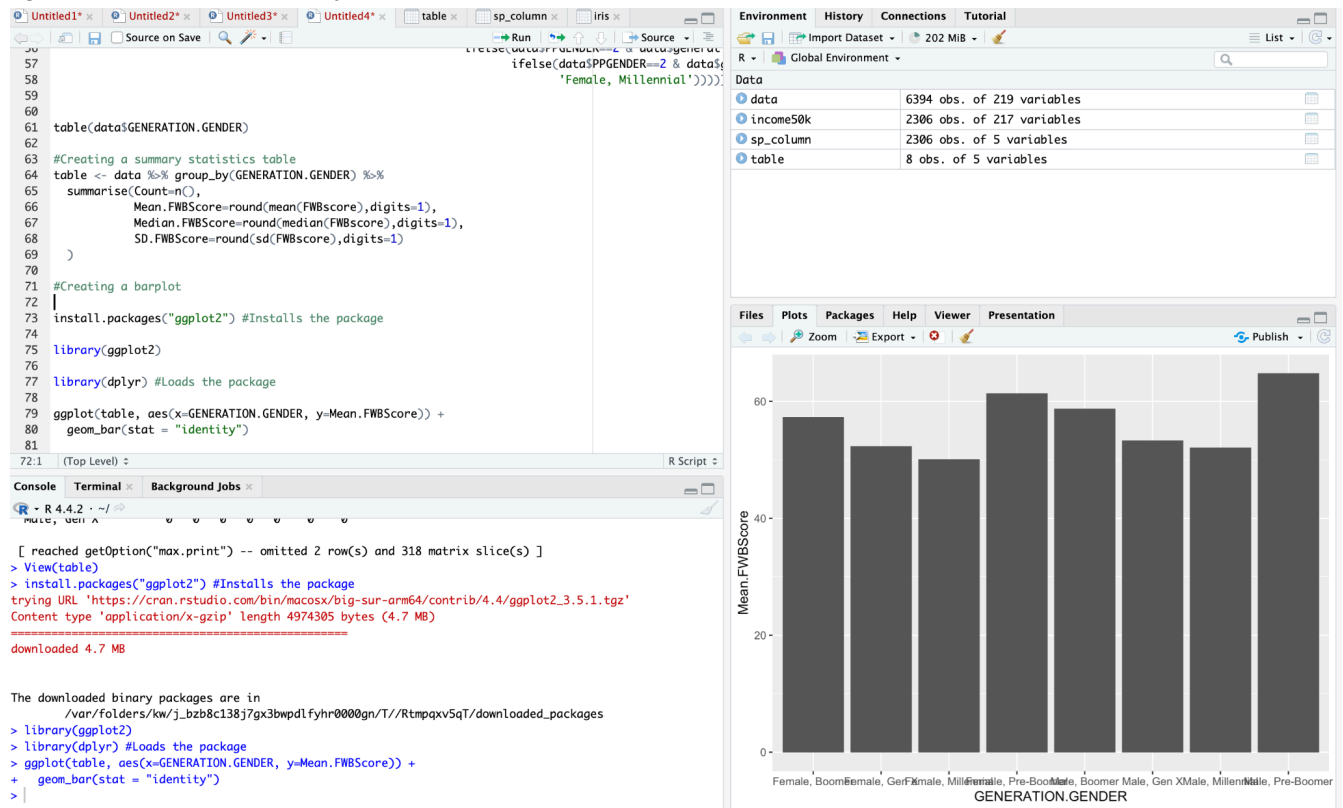
Creating a barplot

`install.packages("ggplot2")` #Installs the package

`library(ggplot2)`

`library(dplyr)` #Loads the package

`ggplot(table, aes(x=GENERATION.GENDER, y=Mean.FWBScore)) +
 geom_bar(stat = "identity")`



`ggplot(table, aes(x=GENERATION.GENDER, y=Mean.FWBScore)) +
 geom_bar(stat = "identity")+
 coord_flip()+
 theme_light()+
 labs(y="Average Financial Well-Being Score", x=" ")`

