

R Outputs for Tutorial 1 Activity: Home Ownership and Education Level

```
> # Hello :)
> # This data set that we will explore is named Home_SCF2013
> # It is a csv file.
> # However, upon importing it into R, we will rename it as Home
> Home <- read.csv("Home_SCF2013.csv")
> # Get the variable names in the data frame
> names(Home)
[1] "Education_Level" "Home_Ownership"
> # Get the structure of the variables in the data frame
> str(Home)
'data.frame': 6015 obs. of 2 variables:
 $ Education_Level: int  2 4 2 4 4 2 4 4 3 2 ...
 $ Home_Ownership : int  2 2 1 1 1 1 1 2 2 2 ...
> # Attach the data frame
> # Reason for doing so is to use each variable name as they are named
> attach(Home)

> # Open the tidyverse library
> ## We need the tidyverse package for several reasons.
> ## We need to change the levels of each variable from numeric to text format.
> ## Why do this?
> ## We need to know what each numeric value represents in a variable.
> ## This helps to display results clearly.
> library(tidyverse)
-- Attaching packages ----- tidyverse 1.3.1 --
v ggplot2 3.3.5      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   2.1.1      v forcats 0.5.1
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
Warning messages:
1: package 'tidyverse' was built under R version 4.0.5
2: package 'ggplot2' was built under R version 4.0.5
3: package 'tibble' was built under R version 4.0.5
4: package 'tidyr' was built under R version 4.0.5
5: package 'readr' was built under R version 4.0.5
6: package 'dplyr' was built under R version 4.0.5
7: package 'forcats' was built under R version 4.0.5
> # Let's obtain contingency table of Education_Level and Home_Ownership
> # Save the contingency table in "Home.Table"
> Home.Table <- table(Education_Level, Home_Ownership)
> Home.Table
```

	Home_Ownership	
Education_Level	1	2
1	252	294
2	953	646
3	567	463
4	2227	613

```

> # Let's make some changes to our data frame.
> # We will use the pipe function %>% to apply the changes.
> # Let's create a new data frame to store our changes.
> # Like this: Home2 <- Home %>%
> # We will use the mutate function to make change to structure of our variables.
> # We use the argument as_factor inside the mutate function.
> # We are changing our variables' integer formats to factors.
> Home2 <- Home %>%
+   mutate(Education_Level = as_factor(Education_Level),
+          Home_Ownership = as_factor(Home_Ownership))
> # We will now recode the levels of our factors for the purpose of clarity.
> # We will use the argument fct_recode inside the mutate function.
> Home2 <- Home2 %>%
+   mutate(Education_Level = fct_recode(Education_Level,
+                                       "No High School" = "1",
+                                       "High School" = "2",
+                                       "Some College" = "3",
+                                       "College Degree" = "4"),
+          Home_Ownership = fct_recode(Home_Ownership,
+                                       "Yes" = "1",
+                                       "No" = "2"))
> # View few lines of Home2 data frame to see the changes made.
> head(Home2)
  Education_Level Home_Ownership
1    High School           No
2 College Degree           No
3    High School           Yes
4 College Degree           Yes
5 College Degree           Yes
6    High School           Yes

> # Let's re-order the levels of our factor levels.
> # This is for the purpose of displaying results in order.
> # Otherwise, R will display results in alphabetical order.
> Home2 <- Home2 %>%
+   mutate(Education_Level = fct_relevel(Education_Level,
+                                       c("No High School",
+                                       "High School",
+                                       "Some College",
+                                       "College Degree")),
+          Home_Ownership = fct_relevel(Home_Ownership,
+                                       c("Yes", "No")))
> # Detach the data frame Home
> detach(Home)
> # Obtain contingency table of Education_Level and Home_Ownership
> # Save the the contingency table in "Home.Table"
> Table <- table(Education_Level, Home_Ownership)
Error in table(Education_Level, Home_Ownership) :
  object 'Education_Level' not found
> Table
Error: object 'Table' not found

```

```
> # Attach Home2 data frame
> attach(Home2)
> # Obtain contingency table of Education_Level and Home_Ownership
> # Save the the contingency table in "Table"
> Table <- table(Education_Level, Home_Ownership)
> Table
```

	Home_Ownership	
Education_Level	Yes	No
No High School	252	294
High School	953	646
Some College	567	463
College Degree	2227	613

```
> # Add Margins to the table
> addmargins(Table)
```

	Home_Ownership		
Education_Level	Yes	No	Sum
No High School	252	294	546
High School	953	646	1599
Some College	567	463	1030
College Degree	2227	613	2840
Sum	3999	2016	6015

```
> # Calculate Marginal Proportions for Education_Level
> Margin.Prop.Edu <- prop.table(margin.table(Table, 1))
> Margin.Prop.Edu
```

Education_Level	High School	Some College	College Degree
No High School	0.09077307	0.26583541	0.17123857
			0.47215295

```
> # Add margins to the marginal distribution table
> addmargins(Margin.Prop.Edu)
```

Education_Level	High School	Some College	College Degree	Sum
No High School	0.09077307	0.26583541	0.17123857	0.47215295
				1.00000000

```
> # Calculate Marginal Proportions for Home_Ownership
> Margin.Prop.Home <- prop.table(margin.table(Table, 2))
> Margin.Prop.Home
```

Home_Ownership	Yes	No
	0.6648379	0.3351621

```
> # Add margins to the marginal distribution table
> addmargins(Margin.Prop.Home)
```

Home_Ownership	Yes	No	Sum
	0.6648379	0.3351621	1.0000000

```
> Joint.Prop <- prop.table(Table)
```

```
> Joint.Prop
```

	Home_Ownership	
Education_Level	Yes	No
No High School	0.04189526	0.04887781
High School	0.15843724	0.10739817
Some College	0.09426434	0.07697423
College Degree	0.37024106	0.10191189

```
> # Add margins to the joint distribution table
```

```
> addmargins(Joint.Prop)
```

	Home_Ownership		
Education_Level	Yes	No	Sum
No High School	0.04189526	0.04887781	0.09077307
High School	0.15843724	0.10739817	0.26583541
Some College	0.09426434	0.07697423	0.17123857
College Degree	0.37024106	0.10191189	0.47215295
Sum	0.66483791	0.33516209	1.00000000

```
> # Calculate Row Proportions
```

```
> Row.Prop <- prop.table(Table, 1)
```

```
> Row.Prop
```

	Home_Ownership	
Education_Level	Yes	No
No High School	0.4615385	0.5384615
High School	0.5959975	0.4040025
Some College	0.5504854	0.4495146
College Degree	0.7841549	0.2158451

```
> # Calculate Column Proportions
```

```
> Col.Prop <- prop.table(Table, 2)
```

```
> Col.Prop
```

	Home_Ownership	
Education_Level	Yes	No
No High School	0.06301575	0.14583333
High School	0.23830958	0.32043651
Some College	0.14178545	0.22966270
College Degree	0.55688922	0.30406746

```
> # To plot the data, we will use the package ggplot2
> # ggplot2: grammar of graphics - Data Visualization
> # Generally, we need to make sure we have the package installed,
> # before we call for its functions.
> # Since we have loaded the tidyvers library,
> # ggplot2 library is included in it and it is already loaded.
> # That is, we can use the functions in ggplot2
> # But, you can open the ggplot2 library again :)
> library(ggplot2)
```

```

> # Exercise.
> # We will construct a side-by-side (clustered) bar chart of the data
> # bar.plot is a name where we want to save the plot and its features
> # ggplot function will make a canvas,
> # and will make the plot ready using the data set and its variables of interest
> bar.plot = ggplot(Home2, aes(x = Education_Level, fill = Home_Ownership))
> # We will add the bars to the plot of the data
> # As well, we will add the legends and position it to the right-hand side
> bar.plot = bar.plot + geom_bar(position = "dodge")
> # We will add a label to the x-axis,
> # We will differentiate the bars by filling in the levels of the response variable
> # We will add a title and a subtitle to the plot
> # And, we will centre the position of both the title and the subtitle
> # Modify line 140 with your last-name in the subtitle
> bar.plot = bar.plot + labs(xlab = "Education Level", fill = "Home Ownership",
+                             title = "Bar Plot of Home Ownership and Education Level",
+                             subtitle = "Constructed by You")
> bar.plot = bar.plot + theme(plot.title=element_text(hjust=0.5),
+                             plot.subtitle = element_text(hjust=0.5))
> bar.plot

```

