

Ford Ka Clustering Analysis

FRE518: Survey Design and Data Analysis

1 Getting started

1a. Create a folder called FRE518 Assignment 1 in your computer (you can name it whatever you like). Then create two subfolders called **Data** and **Code**, respectively.

Open up RStudio and start a new R Project. This project should be associated with this assignment's folder (File -> New Project -> Existing Directory -> FRE518 Assignment 1 folder). You will submit this zip folder in Canvas as part of your assignment.

Start a new RMarkdown file (File -> New file -> R Markdown) and save this file in the **Code** folder.

1b. Load the {pacman} package. Now use the p_load() function to load the following packages: here, dplyr, tidyverse, readxl, janitor, modelsummary, gtsummary, cluster, factoextra, and kableExtra.

```
pacman::p_load(dplyr, tidyverse, ggplot2, here, readxl, janitor, modelsummary,
               kableExtra, cluster, factoextra, gtsummary)
```

1c. Download the dataset from Canvas. Save it in the Data subfolder of your FRE518 Assignment 1 folder.

1d. Read the the demographic and psychographic sheets into R using the read_excel() function. *Hint: You can use the arguments sheet = sheet_name and skip = #_of_rows_to_skip to load these files without having to make any changes to the Excel file uploaded on Canvas. Then merge these two dataframes by respondent_id using the left_join function.*

You will notice that some of the variable names are more than one word, and some are capitalized. While this is okay, it might just be more time consuming to call these variables later on. So you can use the clean_names() function from the {janitor} package to “clean” the variable names.

```
# Read in the data
dg <- read_excel(here("data", "Ford Ka (Student).xls"),
                 sheet = "Demographic Data",
                 skip = 6) %>%
  clean_names()

psyc <- read_excel(here("data", "Ford Ka (Student).xls"),
                  sheet = "Psychographic Data",
                  skip = 6) %>%
  clean_names()

# Merge the two files together
ford <- left_join(dg, psyc, by = c("respondent_number"))
```

1e. All variables are stored as numeric (you can check with the glimpse() function), which is in the format that we need to do the clustering analysis. It may be helpful to label the demographic variables based on

their definition. One way to do this in R is to convert the variables to factors because R stores these variables as numeric but can display text when printed.

Use the `recode_factor()` to convert the following variables to factors: `preference_group`, `gender`, `x1st_time_purchase`, `age_category`, `children_category`, `income_category`.

```
# manually factor variables
ford$preference_group <- recode_factor(factor(ford$preference_group),
                                       `1` = "Ka Chooser (top 3)",
                                       `2` = "Ka Non-Chooser (bottom 3)",
                                       `3` = "Middle (middle 4)")

ford$gender <- recode_factor(factor(ford$gender),
                              `1` = "Male",
                              `2` = "Female")

ford$marital_status <- recode_factor(factor(ford$marital_status),
                                       `1` = "Married",
                                       `2` = "Living Together",
                                       `3` = "Single")

ford$x1st_time_purchase <- recode_factor(factor(ford$x1st_time_purchase),
                                           `1` = "Yes",
                                           `2` = "No")

ford$age_category <- recode_factor(factor(ford$age_category),
                                    `1` = "<25",
                                    `2` = "25-29",
                                    `3` = "30-34",
                                    `4` = "35-39",
                                    `5` = "40-44",
                                    `6` = ">44")

ford$children_category <- recode_factor(factor(ford$children_category),
                                          `0` = "0 child",
                                          `1` = "1 child",
                                          `2` = ">1 child")

ford$income_category <- recode_factor(factor(ford$income_category),
                                          `1` = "<100K",
                                          `2` = "100K-150K",
                                          `3` = "150K-200K",
                                          `4` = "200K-250K",
                                          `5` = "250K-300K",
                                          `6` = ">300K")
```

2 Crosstabs Analysis

Run a cross-tab analysis to check whether different demographic variables separate “Ka Choosers” from “Ka Non-Choosers”. You can use either the `datasummary_crosstab()` or `tbl_cross()` functions or another function of your choice. Make sure that you show the row percentages.

```
# using {datasummary} package
ct_gender <- datasummary_crosstab(preference_group ~ gender,
                                statistic = 1 ~ Percent("row"),
                                data = ford,
                                title = 'Crosstab of Preference Group and Gender',
                                output = 'kableExtra')

ct_marital <- datasummary_crosstab(preference_group ~ marital_status,
                                statistic = 1 ~ Percent("row"),
                                data = ford,
                                title = 'Crosstab of Preference Group and Marital Status',
                                output = 'latex')

ct_firstcar <- datasummary_crosstab(preference_group ~ x1st_time_purchase,
                                statistic = 1 ~ Percent("row"),
                                title = 'Crosstab of Preference Group and First Car Purchase',
                                data = ford)

ct_kids <- datasummary_crosstab(preference_group ~ children_category,
                                statistic = 1 ~ Percent("row"),
                                title = 'Crosstab of Preference Group and Number of Kids',
                                data = ford)

ct_income <- datasummary_crosstab(preference_group ~ income_category,
                                statistic = 1 ~ Percent("row"),
                                title = 'Crosstab of Preference Group and Income Category',
                                data = ford)

ct_age <- datasummary_crosstab(preference_group ~ age_category,
                                statistic = 1 ~ Percent("row"),
                                title = 'Crosstab of Preference Group and Age Category',
                                data = ford)
```

The results are in line with Student number 76174135.

```
ct_gender %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 1: Crosstab of Preference Group and Gender

| preference_group | | Male | Female |
|---------------------------|-------|------|--------|
| Ka Chooser (top 3) | % row | 46.6 | 53.4 |
| Ka Non-Chooser (bottom 3) | % row | 50.0 | 50.0 |
| Middle (middle 4) | % row | 64.5 | 35.5 |
| All | % row | 52.0 | 48.0 |

```
ct_marital %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 2: Crosstab of Preference Group and Marital Status

| preference_group | | Married | Living Together | Single |
|---------------------------|-------|---------|-----------------|--------|
| Ka Chooser (top 3) | % row | 56.9 | 12.1 | 31.0 |
| Ka Non-Chooser (bottom 3) | % row | 47.2 | 8.3 | 44.4 |
| Middle (middle 4) | % row | 43.5 | 12.9 | 43.5 |
| All | % row | 50.8 | 11.2 | 38.0 |

```
ct_firstcar %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 3: Crosstab of Preference Group and First Car Purchase

| preference_group | | Yes | No |
|---------------------------|-------|------|------|
| Ka Chooser (top 3) | % row | 11.2 | 88.8 |
| Ka Non-Chooser (bottom 3) | % row | 11.1 | 88.9 |
| Middle (middle 4) | % row | 25.8 | 74.2 |
| All | % row | 14.8 | 85.2 |

```
ct_kids %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 4: Crosstab of Preference Group and Number of Kids

| preference_group | | 0 child | 1 child | >1 child |
|---------------------------|-------|---------|---------|----------|
| Ka Chooser (top 3) | % row | 53.4 | 25.0 | 21.6 |
| Ka Non-Chooser (bottom 3) | % row | 62.5 | 16.7 | 20.8 |
| Middle (middle 4) | % row | 66.1 | 11.3 | 22.6 |
| All | % row | 59.2 | 19.2 | 21.6 |

```
ct_income %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 5: Crosstab of Preference Group and Income Category

| preference_group | | <100K | 100K-150K | 150K-200K | 200K-250K | 250K-300K | >300K |
|---------------------------|-------|-------|-----------|-----------|-----------|-----------|-------|
| Ka Chooser (top 3) | % row | 9.5 | 16.4 | 15.5 | 16.4 | 24.1 | 18.1 |
| Ka Non-Chooser (bottom 3) | % row | 6.9 | 20.8 | 22.2 | 22.2 | 16.7 | 11.1 |
| Middle (middle 4) | % row | 11.3 | 19.4 | 19.4 | 17.7 | 17.7 | 14.5 |
| All | % row | 9.2 | 18.4 | 18.4 | 18.4 | 20.4 | 15.2 |

```
ct_age %>% kable_styling(latex_options = c("striped", "hold_position"))
```

Table 6: Crosstab of Preference Group and Age Category

| preference_group | | <25 | 25-29 | 30-34 | 35-39 | 40-44 | >44 |
|---------------------------|-------|------|-------|-------|-------|-------|------|
| Ka Chooser (top 3) | % row | 8.6 | 15.5 | 19.8 | 9.5 | 31.0 | 15.5 |
| Ka Non-Chooser (bottom 3) | % row | 4.2 | 18.1 | 16.7 | 15.3 | 20.8 | 25.0 |
| Middle (middle 4) | % row | 17.7 | 19.4 | 19.4 | 14.5 | 19.4 | 9.7 |
| All | % row | 9.6 | 17.2 | 18.8 | 12.4 | 25.2 | 16.8 |

3 Clustering Analysis

3a. Create a new dataframe called `ford_psync` that contains only the `ford` and `q1:q62` variables only.

3b. Use the `kmeans()` function to run the clustering analysis.

3c. Make sure you use the `set.seed(insert_random_number)` function to ensure I can replicate your answers.

```
ford_psync <- select(ford, q1:q62)

set.seed(2022)
k3 <- kmeans(ford_psync, centers = 3, nstart = 25)
set.seed(2022)
k4 <- kmeans(ford_psync, centers = 4, nstart = 25)
set.seed(2022)
k5 <- kmeans(ford_psync, centers = 5, nstart = 25)
```

The results of the clustering are presented in the Appendix.

- The three clusters have the following sizes: **107, 65, 78**
- The four clusters have the following sizes: **32, 65, 78, 75** These results replicate the numbers in the file [Ford Ka 4-Cluster Results - no variables missing](#).
- The five clusters have the following sizes: **32, 37, 38, 65, 78** I don't get the same small cluster as noted by the students

4 Further Analysis - Using the output from the 4 cluster analysis

4a. Using the `cbind()` function, join the `ford` dataframe and the `cluster` variable from your `k4` object.

```
ford_cluster <- cbind(ford, k4["cluster"])
```

4b. Run a cross-tab analysis on the 4 segments to identify choice preferences and demographic characteristics of each segment.

The sample results below show the analysis for Cluster 1 (their cluster 3) with size 32 or 13% of the sample.

- **Gender:** I also get an even 50/50 split for this whole cluster, but not an even split if we look at Ka Chooser only.
- **Income:** I'm getting that they are high (>300K) income and between 100-150K, not <100K
- **Age:** I also get majority <44 years old, but as noted above, this number is for the whole cluster and not just Ka Chooser

```

c1_gender <- datasummary_crosstab(preference_group ~ gender,
  statistic = 1 ~ Percent("row"),
  data = ford_cluster %>% filter(cluster == 1),
  title = 'Crosstab of Preference Group and Gender (Cluster 1)')

c1_income <- datasummary_crosstab(preference_group ~ income_category,
  statistic = 1 ~ Percent("row"),
  data = ford_cluster %>% filter(cluster == 1),
  title = 'Crosstab of Preference Group and Income (Cluster 1)')

c1_age <- datasummary_crosstab(preference_group ~ age_category,
  statistic = 1 ~ Percent("row"),
  data = ford_cluster %>% filter(cluster == 1),
  title = 'Crosstab of Preference Group and Age (Cluster 1)')

c1_gender %>% kable_styling(latex_options = c("striped", "hold_position"))

```

Table 7: Crosstab of Preference Group and Gender (Cluster 1)

| preference_group | | Male | Female |
|---------------------------|-------|------|--------|
| Ka Chooser (top 3) | % row | 55.6 | 44.4 |
| Ka Non-Chooser (bottom 3) | % row | 25.0 | 75.0 |
| Middle (middle 4) | % row | 50.0 | 50.0 |
| All | % row | 50.0 | 50.0 |

```

c1_income %>% kable_styling(latex_options = c("striped", "hold_position"))

```

Table 8: Crosstab of Preference Group and Income (Cluster 1)

| preference_group | | <100K | 100K-150K | 150K-200K | 200K-250K | 250K-300K | >300K |
|---------------------------|-------|-------|-----------|-----------|-----------|-----------|-------|
| Ka Chooser (top 3) | % row | 22.2 | 27.8 | 0.0 | 16.7 | 16.7 | 16.7 |
| Ka Non-Chooser (bottom 3) | % row | 0.0 | 0.0 | 50.0 | 0.0 | 25.0 | 25.0 |
| Middle (middle 4) | % row | 10.0 | 20.0 | 10.0 | 20.0 | 10.0 | 30.0 |
| All | % row | 15.6 | 21.9 | 9.4 | 15.6 | 15.6 | 21.9 |

```

c1_age %>% kable_styling(latex_options = c("striped", "hold_position"))

```

Table 9: Crosstab of Preference Group and Age (Cluster 1)

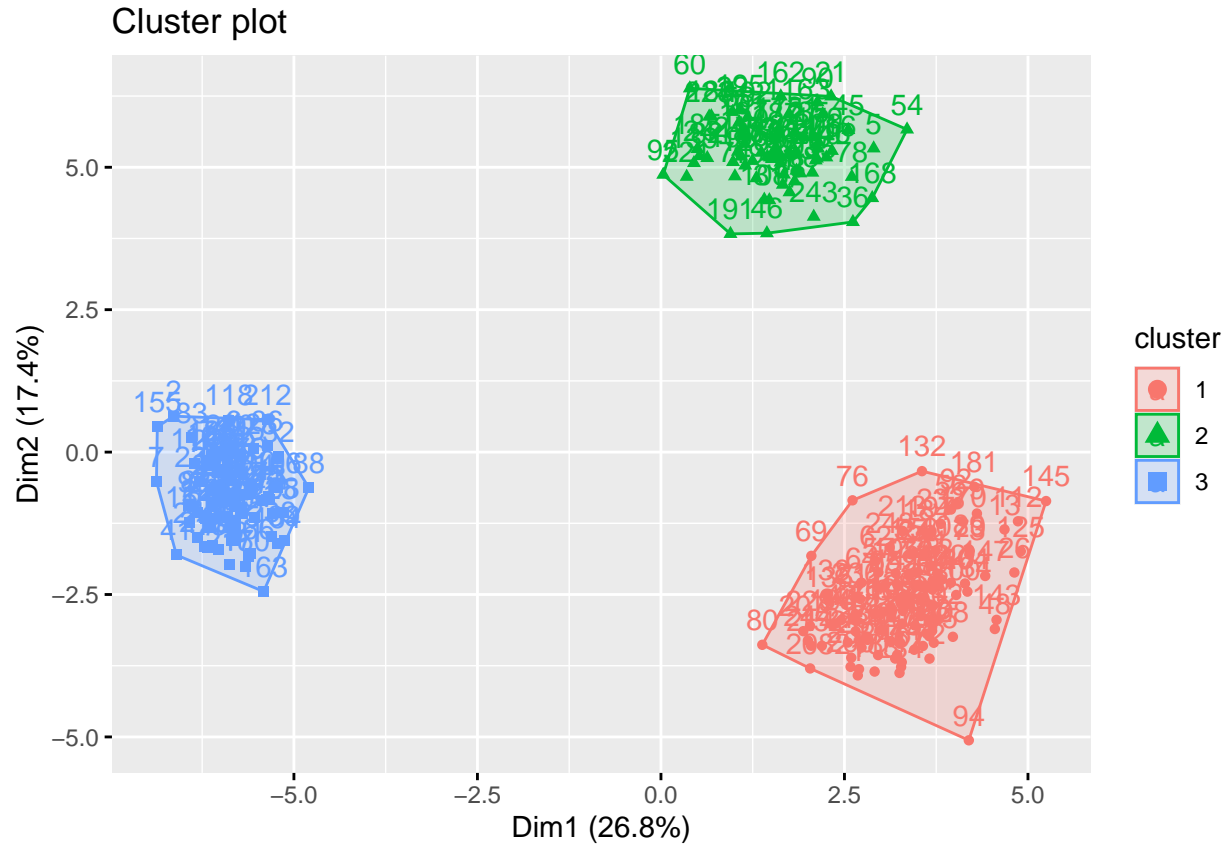
| preference_group | | <25 | 25-29 | 30-34 | 35-39 | 40-44 | >44 |
|---------------------------|-------|------|-------|-------|-------|-------|------|
| Ka Chooser (top 3) | % row | 5.6 | 16.7 | 16.7 | 5.6 | 38.9 | 16.7 |
| Ka Non-Chooser (bottom 3) | % row | 25.0 | 50.0 | 25.0 | 0.0 | 0.0 | 0.0 |
| Middle (middle 4) | % row | 20.0 | 20.0 | 0.0 | 30.0 | 30.0 | 0.0 |
| All | % row | 12.5 | 21.9 | 12.5 | 12.5 | 31.2 | 9.4 |

5 Appendix

```
# 3 cluster analysis
k3$centers
```

```
##          q1          q2          q3          q4          q5          q6          q7          q8
## 1 4.728972 2.448598 3.943925 3.308411 2.411215 4.018692 3.822430 3.915888
## 2 4.015385 3.769231 5.938462 6.015385 6.015385 3.969231 3.800000 4.107692
## 3 6.512821 6.512821 3.884615 4.025641 4.012821 3.974359 4.025641 3.756410
##          q9          q10          q11          q12          q13          q14          q15          q16
## 1 3.859813 3.878505 3.971963 4.084112 3.934579 6.130841 6.224299 5.308411
## 2 3.769231 3.815385 3.876923 4.138462 4.169231 3.984615 4.169231 3.923077
## 3 4.076923 4.051282 4.089744 4.000000 3.910256 1.512821 3.923077 3.910256
##          q17          q18          q19          q20          q21          q22          q23          q24
## 1 6.205607 5.336449 5.485981 5.383178 6.121495 6.186916 2.542056 2.345794
## 2 2.000000 4.015385 4.215385 4.061538 4.200000 4.107692 3.876923 3.692308
## 3 4.064103 3.858974 3.987179 1.512821 3.846154 4.089744 6.487179 1.320513
##          q25          q26          q27          q28          q29          q30          q31          q32
## 1 1.934579 1.822430 2.691589 1.897196 2.700935 2.598131 4.672897 4.112150
## 2 4.169231 3.800000 4.169231 4.030769 4.107692 3.892308 6.030769 6.046154
## 3 3.961538 3.987179 3.923077 4.038462 3.923077 3.910256 1.564103 4.076923
##          q33          q34          q35          q36          q37          q38          q39          q40
## 1 4.018692 4.018692 4.028037 3.925234 4.813084 4.140187 3.971963 3.841121
## 2 5.969231 5.938462 6.076923 5.969231 6.092308 6.046154 1.861538 2.046154
## 3 4.141026 3.948718 4.102564 3.910256 3.820513 4.076923 4.038462 3.820513
##          q41          q42          q43          q44          q45          q46          q47          q48
## 1 3.205607 3.252336 3.981308 3.990654 4.009346 4.046729 3.803738 4.037383
## 2 1.969231 2.030769 1.892308 2.000000 3.923077 3.800000 4.184615 3.846154
## 3 6.500000 3.935897 3.833333 6.512821 6.435897 6.512821 6.576923 6.564103
##          q49          q50          q51          q52          q53          q54          q55          q56
## 1 4.056075 4.102804 4.158879 4.803738 4.906542 4.009346 4.000000 3.943925
## 2 3.923077 3.969231 4.000000 3.953846 3.984615 3.800000 3.861538 3.953846
## 3 6.474359 6.487179 1.564103 1.435897 1.538462 1.461538 1.346154 1.384615
##          q57          q58          q59          q60          q61          q62
## 1 4.822430 4.785047 4.785047 3.420561 3.102804 3.196262
## 2 3.861538 4.092308 3.830769 3.907692 3.892308 3.969231
## 3 4.000000 4.076923 4.089744 4.141026 4.294872 4.076923
```

```
fviz_cluster(k3, data = ford_psync)
```

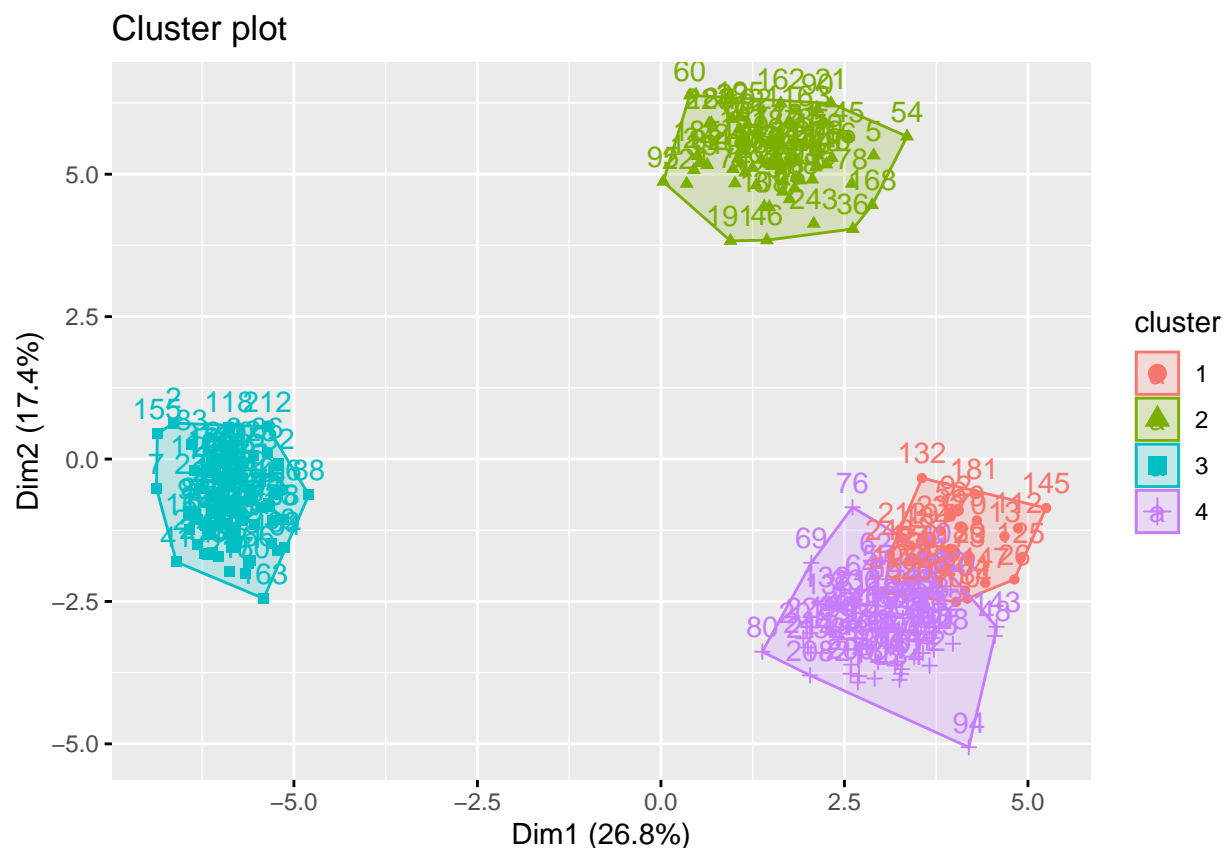


```
# 4 cluster analysis
k4$centers
```

| ## | q1 | q2 | q3 | q4 | q5 | q6 | q7 | q8 |
|------|----------|----------|----------|----------|----------|----------|----------|----------|
| ## 1 | 6.500000 | 3.562500 | 3.781250 | 1.500000 | 3.937500 | 4.218750 | 3.562500 | 3.843750 |
| ## 2 | 4.015385 | 3.769231 | 5.938462 | 6.015385 | 6.015385 | 3.969231 | 3.800000 | 4.107692 |
| ## 3 | 6.512821 | 6.512821 | 3.884615 | 4.025641 | 4.012821 | 3.974359 | 4.025641 | 3.756410 |
| ## 4 | 3.973333 | 1.973333 | 4.013333 | 4.080000 | 1.760000 | 3.933333 | 3.933333 | 3.946667 |
| ## | q9 | q10 | q11 | q12 | q13 | q14 | q15 | q16 |
| ## 1 | 3.718750 | 3.750000 | 4.000000 | 3.875000 | 3.843750 | 6.468750 | 6.562500 | 3.843750 |
| ## 2 | 3.769231 | 3.815385 | 3.876923 | 4.138462 | 4.169231 | 3.984615 | 4.169231 | 3.923077 |
| ## 3 | 4.076923 | 4.051282 | 4.089744 | 4.000000 | 3.910256 | 1.512821 | 3.923077 | 3.910256 |
| ## 4 | 3.920000 | 3.933333 | 3.960000 | 4.173333 | 3.973333 | 5.986667 | 6.080000 | 5.933333 |
| ## | q17 | q18 | q19 | q20 | q21 | q22 | q23 | q24 |
| ## 1 | 6.468750 | 3.843750 | 4.281250 | 4.218750 | 6.437500 | 6.562500 | 3.968750 | 3.625000 |
| ## 2 | 2.000000 | 4.015385 | 4.215385 | 4.061538 | 4.200000 | 4.107692 | 3.876923 | 3.692308 |
| ## 3 | 4.064103 | 3.858974 | 3.987179 | 1.512821 | 3.846154 | 4.089744 | 6.487179 | 1.320513 |
| ## 4 | 6.093333 | 5.973333 | 6.000000 | 5.880000 | 5.986667 | 6.026667 | 1.933333 | 1.800000 |
| ## | q25 | q26 | q27 | q28 | q29 | q30 | q31 | q32 |
| ## 1 | 1.625000 | 1.468750 | 4.187500 | 1.656250 | 4.062500 | 4.125000 | 6.562500 | 3.937500 |
| ## 2 | 4.169231 | 3.800000 | 4.169231 | 4.030769 | 4.107692 | 3.892308 | 6.030769 | 6.046154 |
| ## 3 | 3.961538 | 3.987179 | 3.923077 | 4.038462 | 3.923077 | 3.910256 | 1.564103 | 4.076923 |
| ## 4 | 2.066667 | 1.973333 | 2.053333 | 2.000000 | 2.120000 | 1.946667 | 3.866667 | 4.186667 |
| ## | q33 | q34 | q35 | q36 | q37 | q38 | q39 | q40 |
| ## 1 | 4.062500 | 4.000000 | 4.218750 | 3.875000 | 6.656250 | 4.312500 | 4.093750 | 3.750000 |
| ## 2 | 5.969231 | 5.938462 | 6.076923 | 5.969231 | 6.092308 | 6.046154 | 1.861538 | 2.046154 |


```
## 3 4.141026 3.948718 4.102564 3.910256 3.820513 4.076923 4.038462 3.820513
## 4 4.000000 4.026667 3.946667 3.946667 4.026667 4.066667 3.920000 3.880000
##      q41      q42      q43      q44      q45      q46      q47      q48
## 1 1.437500 1.468750 4.187500 4.375000 4.062500 4.406250 3.718750 4.000000
## 2 1.969231 2.030769 1.892308 2.000000 3.923077 3.800000 4.184615 3.846154
## 3 6.500000 3.935897 3.833333 6.512821 6.435897 6.512821 6.576923 6.564103
## 4 3.960000 4.013333 3.893333 3.826667 3.986667 3.893333 3.840000 4.053333
##      q49      q50      q51      q52      q53      q54      q55      q56
## 1 4.093750 4.187500 4.343750 6.375000 6.468750 3.843750 3.656250 4.125000
## 2 3.923077 3.969231 4.000000 3.953846 3.984615 3.800000 3.861538 3.953846
## 3 6.474359 6.487179 1.564103 1.435897 1.538462 1.461538 1.346154 1.384615
## 4 4.040000 4.066667 4.080000 4.133333 4.240000 4.080000 4.146667 3.866667
##      q57      q58      q59      q60      q61      q62
## 1 6.468750 6.562500 6.406250 1.500000 1.406250 1.593750
## 2 3.861538 4.092308 3.830769 3.907692 3.892308 3.969231
## 3 4.000000 4.076923 4.089744 4.141026 4.294872 4.076923
## 4 4.120000 4.026667 4.093333 4.240000 3.826667 3.880000
```

```
fviz_cluster(k4, data = ford_psync)
```



```
# 5 cluster analysis
k5$centers
```

```
##      q1      q2      q3      q4      q5      q6      q7      q8
## 1 6.500000 3.562500 3.781250 1.500000 3.937500 4.218750 3.562500 3.843750
```

```

## 2 4.216216 1.972973 4.162162 3.918919 1.783784 3.864865 4.081081 3.513514
## 3 3.736842 1.973684 3.868421 4.236842 1.736842 4.000000 3.789474 4.368421
## 4 4.015385 3.769231 5.938462 6.015385 6.015385 3.969231 3.800000 4.107692
## 5 6.512821 6.512821 3.884615 4.025641 4.012821 3.974359 4.025641 3.756410
##      q9      q10      q11      q12      q13      q14      q15      q16
## 1 3.718750 3.750000 4.000000 3.875000 3.843750 6.468750 6.562500 3.843750
## 2 3.405405 4.351351 3.918919 4.675676 4.567568 6.027027 6.054054 5.945946
## 3 4.421053 3.526316 4.000000 3.684211 3.394737 5.947368 6.105263 5.921053
## 4 3.769231 3.815385 3.876923 4.138462 4.169231 3.984615 4.169231 3.923077
## 5 4.076923 4.051282 4.089744 4.000000 3.910256 1.512821 3.923077 3.910256
##      q17      q18      q19      q20      q21      q22      q23      q24
## 1 6.468750 3.843750 4.281250 4.218750 6.437500 6.562500 3.968750 3.625000
## 2 5.891892 5.864865 5.810811 5.918919 5.918919 5.891892 1.918919 1.837838
## 3 6.289474 6.078947 6.184211 5.842105 6.052632 6.157895 1.947368 1.763158
## 4 2.000000 4.015385 4.215385 4.061538 4.200000 4.107692 3.876923 3.692308
## 5 4.064103 3.858974 3.987179 1.512821 3.846154 4.089744 6.487179 1.320513
##      q25      q26      q27      q28      q29      q30      q31      q32
## 1 1.625000 1.468750 4.187500 1.656250 4.062500 4.125000 6.562500 3.937500
## 2 2.162162 1.972973 2.135135 2.000000 2.189189 1.864865 3.864865 4.297297
## 3 1.973684 1.973684 1.973684 2.000000 2.052632 2.026316 3.868421 4.078947
## 4 4.169231 3.800000 4.169231 4.030769 4.107692 3.892308 6.030769 6.046154
## 5 3.961538 3.987179 3.923077 4.038462 3.923077 3.910256 1.564103 4.076923
##      q33      q34      q35      q36      q37      q38      q39      q40
## 1 4.062500 4.000000 4.218750 3.875000 6.656250 4.312500 4.093750 3.750000
## 2 4.216216 3.837838 3.648649 4.027027 4.162162 4.027027 4.189189 4.027027
## 3 3.789474 4.210526 4.236842 3.868421 3.894737 4.105263 3.657895 3.736842
## 4 5.969231 5.938462 6.076923 5.969231 6.092308 6.046154 1.861538 2.046154
## 5 4.141026 3.948718 4.102564 3.910256 3.820513 4.076923 4.038462 3.820513
##      q41      q42      q43      q44      q45      q46      q47      q48
## 1 1.437500 1.468750 4.187500 4.375000 4.062500 4.406250 3.718750 4.000000
## 2 4.270270 4.135135 3.837838 3.810811 3.918919 3.972973 3.675676 3.756757
## 3 3.657895 3.894737 3.947368 3.842105 4.052632 3.815789 4.000000 4.342105
## 4 1.969231 2.030769 1.892308 2.000000 3.923077 3.800000 4.184615 3.846154
## 5 6.500000 3.935897 3.833333 6.512821 6.435897 6.512821 6.576923 6.564103
##      q49      q50      q51      q52      q53      q54      q55      q56
## 1 4.093750 4.187500 4.343750 6.375000 6.468750 3.843750 3.656250 4.125000
## 2 4.081081 3.972973 3.972973 3.891892 4.054054 4.027027 4.000000 3.837838
## 3 4.000000 4.157895 4.184211 4.368421 4.421053 4.131579 4.289474 3.894737
## 4 3.923077 3.969231 4.000000 3.953846 3.984615 3.800000 3.861538 3.953846
## 5 6.474359 6.487179 1.564103 1.435897 1.538462 1.461538 1.346154 1.384615
##      q57      q58      q59      q60      q61      q62
## 1 6.468750 6.562500 6.406250 1.500000 1.406250 1.593750
## 2 4.189189 4.270270 3.783784 4.324324 4.270270 3.540541
## 3 4.052632 3.789474 4.394737 4.157895 3.394737 4.210526
## 4 3.861538 4.092308 3.830769 3.907692 3.892308 3.969231
## 5 4.000000 4.076923 4.089744 4.141026 4.294872 4.076923

```

```
fviz_cluster(k5, data = ford_psync)
```

