Assignment: Clustering Employees

Dunja Novaković

2021-08-02

Instructions

This assignment reviews the *Unsupervised Learning* analytical lecture. You will use the *unsupervised_learning.Rmd* file I reviewed in the video lectures to complete this assignment. You will *copy and paste* relevant code from that file and update it to answer the questions in this assignment. You will respond to questions in each section after executing relevant code to answer a question. You will submit this assignment to its *Submissions* folder on *D2L*. You will submit this *(1)* completed **R Markdown** script and *(2)* a *PDF*, *Word*, or *HTML* rendered version of it to *D2L* by the due date and time. As a first option, if you installed `TinyTeX` successfully, then I prefer a *PDF* version. As a second option, if you have *Microsoft Word*, then I prefer a *Word* version. As a third option, you can knit to *HTML*. The first two options work better with *D2L*.

To start:

For any analytical project, you want to create a clear project directory structure.
All materials from this course should exist in one folder on your computer. Inside of that main course folder, you should create folders to store course documentation, lecture analytical projects, assignments analytical projects, etc. Inside of your folder for assignments analytical projects, you should create folder for this assignment named *unsupervised_learning*.

Any analytical project folder should contain inside it at least three additional folders named *scripts*, *data*, and *plots*. Store this script in the *scripts* folder, the data for this assignment in the *data* folder, and any requested plots in the *plots* folder. Each analytical project should also contain a **.Rproj** file in its top-level directory. Go to the *File* menu in *RStudio*, select *New Project…*, choose *Existing Directory*, go to the folder you created to contain this analytical project. Select it as the top-level directory for this **RStudio Project**.

Global Settings

The first code chunk sets the global settings for the remaining code chunks in the document. Do *not* change anything in this code chunk.

Load Packages

In this code chunk, we load packages we need for this assignment:

1. **here**,
2. **tidyverse**,
3. **skimr**,
4. **cluster**,
5. **dendextend**,
6. **factoextra**, and

7. **Rtsne**.

We will use functions from these packages to import the data, examine the data, calculate summaries on the data, build logistic regression models, and create visualizations from the data. Do *not* change anything in this code chunk.

```
### load libaries for use in current working session
## here for workflow
library(here)
```

```
## here() starts at C:/Users/novak/OneDrive/Desktop/MGT 591/Assignments/unsupervised_learning
```

```
## tidyverse for data manipulation and plotting
# loads eight different libraries simultaneously
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     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
```

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

```
## skimr for summary statistics
library(skimr)

## cluster for partitioning around medoids
library(cluster)

## dendextend for visualizing dendrograms
library(dendextend)
```

```
##
## ---------------------
## Welcome to dendextend version 1.15.1
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## Or contact: <tal.galili@gmail.com>
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## ---------------------
```

```
##
## Attaching package: 'dendextend'
```

```
## The following object is masked from 'package:stats':
##
##     cutree
```

```
## factoextra for clustering visualizations
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
## Rtsne for dimensionality reduction
library(Rtsne)
```

Task 1: Load, Clean, and Explore Data

Load the **emp_att.tsv** data file with the correct functions, skipping the first line, setting column names to false, and save the data as **emp_att_data**. Name and mutate the columns just like in the lecture script.

Use **skim_without_charts()** on **emp_att_data** while grouping by **mentor_type**.

Question 1.1: What is the average **email_overload** for individuals with an internal manager mentor (i.e., **Mgr. Mentor Internal**)?

Response 1.1: *2.08*.

Produce density plots for **job_stress** filling in by **married**. Use a facet grid for **useNowFlextime** by **gender**. Label the axes and fill appropriately.

Question 1.2: For which combination of **useNowFlextime** and **gender** are the density distributions for single and married individuals essentially the same?

Response 1.2: *Gender: Female, UseNowFlextime:No*.

Produce a scatterplot of the relationship between **mgr_burnout** on the x-axis and **burnout** on the y-axis. Facet wrap by **mentor_type**. Fit a **lm** smooth geometry. Label the axes appropriately.

Question 1.3: For which **mentor_type** is the linear relationship between manager (**mgr_burnout**) and employee (**burnout**) burnout negative?

Response 1.3: *For the "No Mentor" type.*

```
#### Q1.1
### load data via the read_tsv and here functions
emp_att_data <- read_tsv(here("data", "emp_att.tsv"),
                         ## do not create column names
                         col_names = FALSE,
                         ## skip the first row in the data file
                         skip = 1)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   X1 = col_character(),
##   X14 = col_character(),
##   X15 = col_character(),
##   X16 = col_character(),
```

```
##   X17 = col_character(),
##   X18 = col_character(),
##   X19 = col_character(),
##   X20 = col_character(),
##   X21 = col_character(),
##   X22 = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
### clean data
## name the columns
names(emp_att_data) <- c("gender", "mgr_perf", "mgr_help_beh",
  "mgr_burnout", "email_overload", "depression", "anxiety",
  "cog_failure", "workaholism", "burnout", "job_stress",
  "perfectionism", "engagement", "useNowFlextime", "useNowShortWk",
  "useNowPaidFMLA", "useNowFlexAcct", "useNowEaseBack", "career_interrupt",
  "mentor_type", "partner_employment", "married", "sum_social_interrupt")

## convert character columns to factors
# overwrite data
emp_att_data <- emp_att_data %>%
  # mutate relevant variables
  mutate_at(vars(gender, starts_with("use"), married,
                 career_interrupt, mentor_type, partner_employment),
            as_factor) %>%
  # change labels for gender
  mutate(gender = fct_recode(gender, `Female` = "female", `Male` = "male")) %>%
  # round numeric variables to two decimal places
  mutate_if(is.numeric, ~ round(., digits = 2))

### explore data
## call data
emp_att_data %>%
  ## group by variables
  group_by(mentor_type) %>%
  ## summarize
  skim_without_charts()
```

Table 1: Data summary

Name	Piped data
Number of rows	457
Number of columns	23
Column type frequency:	
factor	9
numeric	13
Group variables	mentor_type

Variable type: factor

skim_variable	mentor_type	n_missing	complete_rate	ordered	n_unique	top_counts
gender	Coach Mentor	0	1	FALSE	2	Fem: 75, Mal: 41
gender	Mgr. Mentor External	0	1	FALSE	2	Fem: 34, Mal: 1
gender	Mgr. Mentor Internal	0	1	FALSE	2	Fem: 80, Mal: 9
gender	Peer Mentor Internal	0	1	FALSE	2	Fem: 70, Mal: 47
gender	No Mentor	0	1	FALSE	2	Mal: 57, Fem: 17
gender	Peer Mentor External	0	1	FALSE	2	Fem: 23, Mal: 3
useNowFlextime	Coach Mentor	0	1	FALSE	2	No: 64, Yes: 52
useNowFlextime	Mgr. Mentor External	0	1	FALSE	2	Yes: 32, No: 3
useNowFlextime	Mgr. Mentor Internal	0	1	FALSE	2	Yes: 70, No: 19
useNowFlextime	Peer Mentor Internal	0	1	FALSE	2	No: 63, Yes: 54
useNowFlextime	No Mentor	0	1	FALSE	2	No: 59, Yes: 15
useNowFlextime	Peer Mentor External	0	1	FALSE	2	Yes: 20, No: 6
useNowShortWk	Coach Mentor	0	1	FALSE	2	No: 112, Yes: 4
useNowShortWk	Mgr. Mentor External	0	1	FALSE	1	No: 35, Yes: 0
useNowShortWk	Mgr. Mentor Internal	0	1	FALSE	2	No: 86, Yes: 3
useNowShortWk	Peer Mentor Internal	0	1	FALSE	2	No: 109, Yes: 8
useNowShortWk	No Mentor	0	1	FALSE	2	No: 68, Yes: 6
useNowShortWk	Peer Mentor External	0	1	FALSE	1	No: 26, Yes: 0
useNowPaidFMLA	Coach Mentor	0	1	FALSE	2	No: 112, Yes: 4
useNowPaidFMLA	Mgr. Mentor External	0	1	FALSE	1	No: 35, Yes: 0
useNowPaidFMLA	Mgr. Mentor Internal	0	1	FALSE	2	No: 81, Yes: 8
useNowPaidFMLA	Peer Mentor Internal	0	1	FALSE	2	No: 110, Yes: 7
useNowPaidFMLA	No Mentor	0	1	FALSE	2	No: 68, Yes: 6
useNowPaidFMLA	Peer Mentor External	0	1	FALSE	1	No: 26, Yes: 0
useNowFlexAcct	Coach Mentor	0	1	FALSE	2	No: 107, Yes: 9
useNowFlexAcct	Mgr. Mentor External	0	1	FALSE	2	No: 34, Yes: 1
useNowFlexAcct	Mgr. Mentor Internal	0	1	FALSE	2	No: 74, Yes: 15
useNowFlexAcct	Peer Mentor Internal	0	1	FALSE	2	No: 107, Yes: 10
useNowFlexAcct	No Mentor	0	1	FALSE	2	No: 66, Yes: 8
useNowFlexAcct	Peer Mentor External	0	1	FALSE	1	No: 26, Yes: 0
useNowEaseBack	Coach Mentor	0	1	FALSE	2	No: 115, Yes: 1

skim_variable	mentor_type	n_missing	complete_rate	ordered	n_unique	top_counts
useNowEaseBack	Mgr. Mentor External	0	1	FALSE	1	No: 35, Yes: 0
useNowEaseBack	Mgr. Mentor Internal	0	1	FALSE	2	No: 87, Yes: 2
useNowEaseBack	Peer Mentor Internal	0	1	FALSE	2	No: 116, Yes: 1
useNowEaseBack	No Mentor	0	1	FALSE	2	No: 72, Yes: 2
useNowEaseBack	Peer Mentor External	0	1	FALSE	1	No: 26, Yes: 0
career_interrupt	Coach Mentor	0	1	FALSE	3	Hav: 109, Edu: 6, Eld: 1, Par: 0
career_interrupt	Mgr. Mentor External	0	1	FALSE	3	Hav: 29, Edu: 5, Par: 1, Eld: 0
career_interrupt	Mgr. Mentor Internal	0	1	FALSE	4	Hav: 78, Eld: 7, Edu: 2, Par: 2
career_interrupt	Peer Mentor Internal	0	1	FALSE	4	Hav: 99, Edu: 9, Eld: 7, Par: 2
career_interrupt	No Mentor	0	1	FALSE	4	Hav: 65, Edu: 7, Par: 1, Eld: 1
career_interrupt	Peer Mentor External	0	1	FALSE	2	Edu: 14, Hav: 12, Par: 0, Eld: 0
partner_employment	Coach Mentor	0	1	FALSE	2	Yes: 109, No: 7
partner_employment	Mgr. Mentor External	0	1	FALSE	1	Yes: 35, No: 0
partner_employment	Mgr. Mentor Internal	0	1	FALSE	2	Yes: 76, No: 13
partner_employment	Peer Mentor Internal	0	1	FALSE	1	Yes: 117, No: 0
partner_employment	No Mentor	0	1	FALSE	1	Yes: 74, No: 0
partner_employment	Peer Mentor External	0	1	FALSE	2	Yes: 25, No: 1
married	Coach Mentor	0	1	FALSE	2	Yes: 87, No: 29
married	Mgr. Mentor External	0	1	FALSE	2	Yes: 34, No: 1
married	Mgr. Mentor Internal	0	1	FALSE	2	Yes: 64, No: 25
married	Peer Mentor Internal	0	1	FALSE	2	Yes: 80, No: 37
married	No Mentor	0	1	FALSE	2	Yes: 63, No: 11
married	Peer Mentor External	0	1	FALSE	2	Yes: 20, No: 6

Variable type: numeric

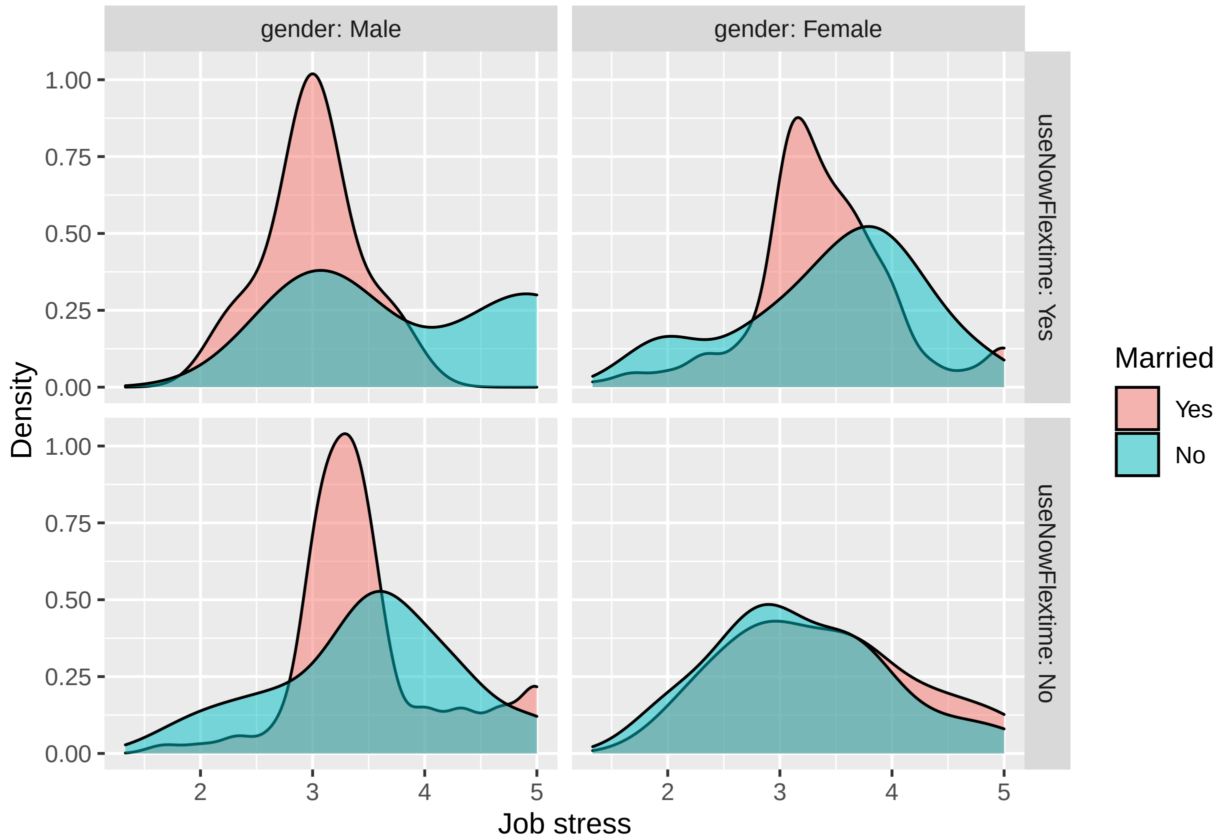
skim_variable	mentor_type	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
mgr_perf	Coach Mentor	0	1	4.11	0.73	1.00	4.00	4.07	4.50	5.00
mgr_perf	Mgr. Mentor External	0	1	4.04	0.32	3.00	4.00	4.00	4.00	5.00
mgr_perf	Mgr. Mentor Internal	0	1	4.19	0.41	3.25	4.00	4.00	4.39	5.00

skim_variable	mentor_type	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
mgr_perf	Peer Mentor Internal	0	1	4.20	0.54	2.50	4.00	4.16	4.50	5.00
mgr_perf	No Mentor	0	1	3.42	0.58	1.00	3.00	3.50	3.91	4.75
mgr_perf	Peer Mentor External	0	1	4.51	0.33	4.00	4.25	4.38	4.75	5.00
mgr_help_beh	Coach Mentor	0	1	3.88	0.67	1.00	3.50	4.00	4.17	5.00
mgr_help_beh	Mgr. Mentor External	0	1	4.06	0.31	3.00	4.00	4.00	4.08	5.00
mgr_help_beh	Mgr. Mentor Internal	0	1	4.08	0.46	2.17	3.83	4.00	4.33	5.00
mgr_help_beh	Peer Mentor Internal	0	1	3.98	0.55	2.17	3.67	4.00	4.19	5.00
mgr_help_beh	No Mentor	0	1	3.42	0.64	1.17	3.04	3.50	3.83	5.00
mgr_help_beh	Peer Mentor External	0	1	4.39	0.46	3.67	3.87	4.50	4.79	5.00
mgr_burnout	Coach Mentor	0	1	1.96	0.90	0.00	1.25	1.77	2.42	4.33
mgr_burnout	Mgr. Mentor External	0	1	2.21	0.46	1.67	2.00	2.00	2.33	4.00
mgr_burnout	Mgr. Mentor Internal	0	1	2.40	1.12	0.00	1.53	2.01	3.33	5.00
mgr_burnout	Peer Mentor Internal	0	1	2.44	0.98	1.00	2.00	2.33	3.01	4.67
mgr_burnout	No Mentor	0	1	3.16	0.68	1.00	3.00	3.16	3.67	4.67
mgr_burnout	Peer Mentor External	0	1	2.88	1.14	1.00	2.03	3.00	3.59	5.00
email_overload	Coach Mentor	0	1	2.00	0.67	1.00	1.50	1.95	2.42	3.75
email_overload	Mgr. Mentor External	0	1	2.29	0.35	1.50	2.19	2.19	2.42	3.50
email_overload	Mgr. Mentor Internal	0	1	2.08	0.75	1.00	1.50	2.00	2.74	4.00
email_overload	Peer Mentor Internal	0	1	2.17	0.81	1.00	1.50	2.00	2.83	4.50
email_overload	No Mentor	0	1	2.94	0.50	1.75	2.50	3.03	3.16	4.00
email_overload	Peer Mentor External	0	1	2.03	0.52	1.00	2.07	2.24	2.30	2.75
depression	Coach Mentor	0	1	1.38	0.45	1.00	1.00	1.20	1.60	3.40
depression	Mgr. Mentor External	0	1	1.49	0.28	1.20	1.38	1.38	1.44	2.54
depression	Mgr. Mentor Internal	0	1	1.53	0.43	1.00	1.23	1.41	1.80	3.00
depression	Peer Mentor Internal	0	1	1.65	0.74	1.00	1.00	1.40	2.20	5.00
depression	No Mentor	0	1	2.66	0.75	1.00	2.20	2.86	3.02	4.60
depression	Peer Mentor External	0	1	1.48	0.42	1.00	1.37	1.40	1.47	3.00
anxiety	Coach Mentor	0	1	1.45	0.43	1.00	1.00	1.35	1.65	3.00
anxiety	Mgr. Mentor External	0	1	1.70	0.39	1.56	1.57	1.59	1.61	3.75
anxiety	Mgr. Mentor Internal	0	1	1.62	0.40	1.00	1.37	1.62	1.83	2.50
anxiety	Peer Mentor Internal	0	1	1.65	0.67	1.00	1.00	1.50	2.00	5.00

skim_variable	mentor_type	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
anxiety	No Mentor	0	1	2.57	0.77	1.00	2.00	2.75	2.99	4.50
anxiety	Peer Mentor External	0	1	1.61	0.41	1.00	1.62	1.64	1.67	3.00
cog_failure	Coach Mentor	0	1	2.03	0.51	1.00	1.66	2.01	2.40	3.40
cog_failure	Mgr. Mentor External	0	1	2.35	0.27	1.80	2.26	2.27	2.46	3.40
cog_failure	Mgr. Mentor Internal	0	1	2.14	0.57	1.00	1.82	2.30	2.51	4.00
cog_failure	Peer Mentor Internal	0	1	2.32	0.69	1.00	2.00	2.40	2.80	4.00
cog_failure	No Mentor	0	1	2.56	0.57	1.00	2.25	2.70	2.88	4.00
cog_failure	Peer Mentor External	0	1	2.17	0.52	1.00	2.20	2.40	2.43	2.48
workaholism	Coach Mentor	0	1	2.64	0.35	1.54	2.46	2.68	2.77	3.62
workaholism	Mgr. Mentor External	0	1	2.94	0.28	1.77	2.96	2.98	2.98	3.54
workaholism	Mgr. Mentor Internal	0	1	2.78	0.47	1.00	2.67	2.79	2.88	4.38
workaholism	Peer Mentor Internal	0	1	2.57	0.61	1.00	2.31	2.63	2.85	4.62
workaholism	No Mentor	0	1	2.95	0.40	1.92	2.85	2.96	3.02	4.00
workaholism	Peer Mentor External	0	1	3.05	0.54	1.54	2.85	3.05	3.17	4.31
burnout	Coach Mentor	0	1	2.48	0.72	1.00	2.00	2.47	3.00	3.67
burnout	Mgr. Mentor External	0	1	2.65	0.45	1.67	2.54	2.56	2.65	4.33
burnout	Mgr. Mentor Internal	0	1	3.04	1.41	1.00	2.00	2.85	3.77	7.00
burnout	Peer Mentor Internal	0	1	3.05	1.36	1.00	2.00	3.28	3.77	6.00
burnout	No Mentor	0	1	4.58	0.65	3.67	4.10	4.41	5.00	7.00
burnout	Peer Mentor External	0	1	3.01	0.62	1.00	2.84	3.04	3.28	4.33
job_stress	Coach Mentor	0	1	3.46	0.64	1.67	3.27	3.47	3.73	5.00
job_stress	Mgr. Mentor External	0	1	3.09	0.29	2.33	3.06	3.06	3.12	4.33
job_stress	Mgr. Mentor Internal	0	1	3.77	0.63	2.00	3.46	3.71	4.00	5.00
job_stress	Peer Mentor Internal	0	1	3.12	0.82	1.33	2.67	3.00	3.54	5.00
job_stress	No Mentor	0	1	3.55	0.72	1.67	3.11	3.32	3.92	5.00
job_stress	Peer Mentor External	0	1	3.54	0.52	2.00	3.22	3.66	3.92	4.33
perfectionism	Coach Mentor	0	1	4.32	0.44	3.00	4.00	4.26	4.66	5.00
perfectionism	Mgr. Mentor External	0	1	3.95	0.35	2.00	3.99	4.00	4.02	4.12
perfectionism	Mgr. Mentor Internal	0	1	4.45	0.33	3.40	4.24	4.40	4.69	5.00
perfectionism	Peer Mentor Internal	0	1	3.90	0.54	1.60	3.60	4.00	4.04	5.00
perfectionism	No Mentor	0	1	3.48	0.66	2.20	3.02	3.20	4.00	5.00

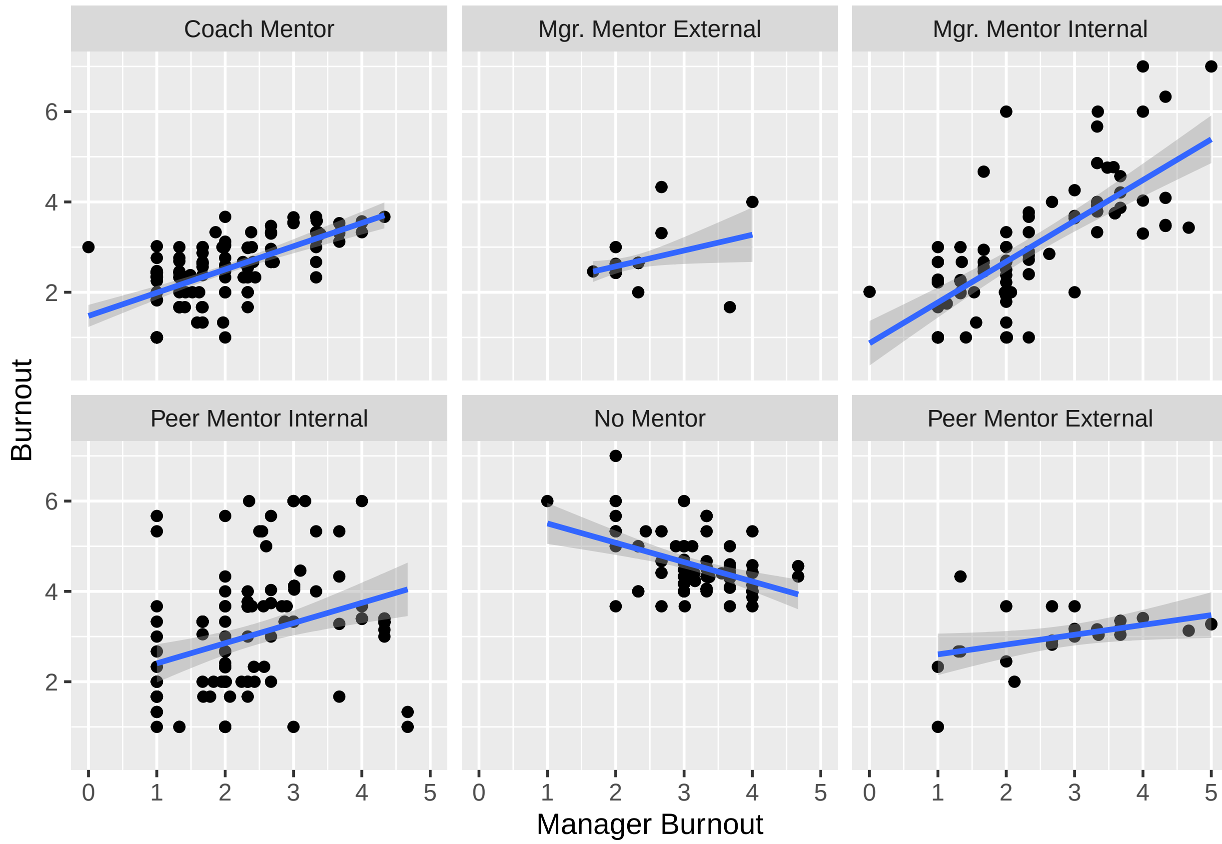
skim_variable	mentor_type	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
perfectionism	Peer Mentor External	0	1	4.30	0.30	3.80	4.20	4.24	4.36	5.00
engagement	Coach Mentor	0	1	3.85	0.40	2.78	3.56	3.89	4.11	5.00
engagement	Mgr. Mentor External	0	1	3.77	0.29	2.67	3.82	3.82	3.83	4.67
engagement	Mgr. Mentor Internal	0	1	3.85	0.44	2.56	3.59	3.82	4.00	5.00
engagement	Peer Mentor Internal	0	1	3.58	0.56	1.89	3.13	3.63	4.00	5.00
engagement	No Mentor	0	1	3.23	0.56	1.00	3.04	3.13	3.44	4.89
engagement	Peer Mentor External	0	1	3.92	0.55	2.44	3.65	3.88	3.95	5.00
sum_social_interact	Coach Mentor	0	1	77.66	43.67	0.00	56.00	71.52	96.08	285.00
sum_social_interact	Mgr. Mentor External	0	1	87.53	26.08	29.00	79.23	80.20	84.75	155.82
sum_social_interact	Mgr. Mentor Internal	0	1	130.78	66.99	25.00	98.13	121.52	148.00	370.00
sum_social_interact	Peer Mentor Internal	0	1	109.34	157.29	0.00	50.00	74.32	115.75	1507.00
sum_social_interact	No Mentor	0	1	132.17	125.57	13.00	72.77	84.74	165.51	944.00
sum_social_interact	Peer Mentor External	0	1	126.09	53.25	10.00	96.84	119.90	146.16	248.39

```r
#### Q1.2
### plot data
## density distributions
# call data and set aesthetics
ggplot(emp_att_data, aes(x = job_stress, fill = married)) +
  # density geometry
  geom_density(alpha = 0.5) +
  # facet by flex time and short week
  facet_grid(useNowFlextime ~ gender, labeller = label_both) +
  # aesthetic labels
  labs(x = "Job stress", y = "Density", fill = "Married")
```




```
#### Q1.3
## hexagonal count plots
# call data and set aesthetics
ggplot(emp_att_data, aes(x = mgr_burnout, y = burnout)) +
  # hexagonal geometry
  geom_point() +
  # facet by flex time and short week
  facet_wrap(~ mentor_type) +
  # smooth geometry
  geom_smooth(method = "lm") +
  # aesthetic labels
  labs(x = "Manager Burnout", y = "Burnout")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Task 2: Agglomerative Hierarchical Clustering

Create a new data object named **emp_att_num** consisting of the following numeric variables: **burnout**, **job_stress**, **workaholism**, **anxiety**, **perfectionism**, **engagement**, and **depression**. Compute the Euclidean distance matrix based on **emp_att_num** and name the result **emp_dist_num**. Make sure to apply **scale** to **emp_att_num**. Apply **fviz_dist()** to **emp_dist_num** to visualize the distance matrix.

Question 2.1: Relatively speaking, are the individuals in the top-right quadrant more dissimilar or similar to each other?

Response 2.1: *More similar to each other.*

Apply the agglomerative hierarchical clustering algorithm to the distance matrix saving the result as **emp_hclust** using the **complete** method. Apply **head()** to the *merge* sequence from **emp_hclust** and set **n = 20** inside **head()**.

Question 2.2: Which two individuals merged to form the first cluster? How many individuals subsequently joined the first cluster? Which two individuals merged to form the second cluster?

Response 2.2: *Individuals 2 and 110 merged to form the first cluster. Ten more individuals subsequently joined the first cluster. Individuals 8 and 67 merged to form the second cluster.*

Produce a tree and radial dendrogram plots. First, create a new object named **dend_emp_hclust** from applying **as.dendrogram()** to **emp_hclust**. Set attributes of **dend_emp_hclust** by using code from the lecture. Set **branches_k_color** to *10* clusters. Remove labels of dendrogram leaves by setting **labels_cex** to *0*. Convert **dend_emp_hclust** to a ggplot dendrogram. Produce a tree dendrogram. Produce a radial dendrogram.

Question 2.3: Looking at the plots, is there an approximately equal number of individuals in each of the *10* clusters?

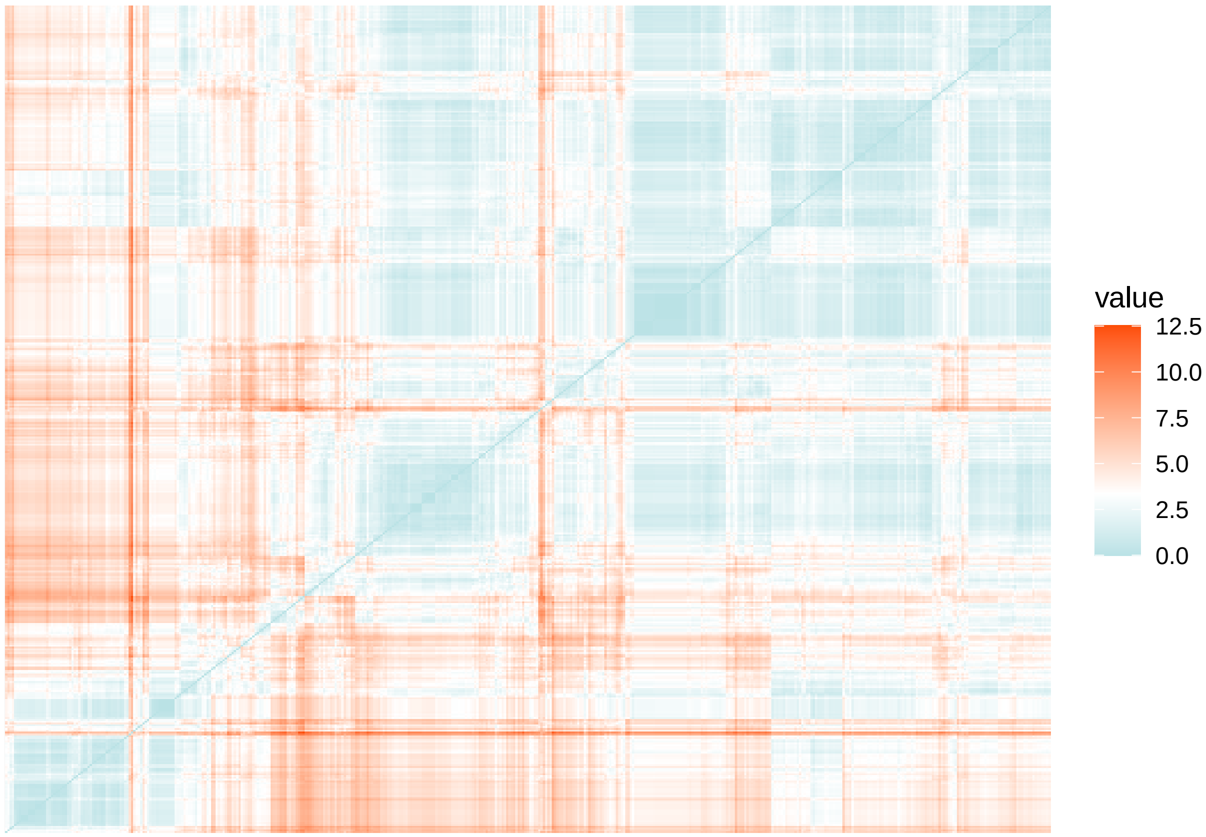
Response 2.3: *No.*

First, use **cutree()** to count the number of individuals when the results from **emp_hclust** are divided into *10* clusters. Second, plot the average values on the original variables of the first *6* clusters after applying **cutree()** to divide **emp_hclust** into *10* clusters. The plot should be a bar plot of each cluster on the x-axis. The height of the bar should represent the average value on each of the original variables. Apply a facet wrap using the original variables. See the lecture script.

Question 2.4: How many individuals are in the fifth cluster? Which cluster has the highest average **anxiety**? Which cluster has the highest average **engagement**?

Response 2.4: *There are 32 individuals in the fifth cluster. Cluster 6 has the highest average anxiety. Cluster 3 has the highest average engagement.*

```
#### Q2.1
### numeric variables data object
## create data object
emp_att_num <- emp_att_data %>%
  ## select variables of choice
  select(burnout, job_stress, workaholism, anxiety, perfectionism, engagement, depression)

### distance between employees on set of variables
## calculate Euclidean distance
emp_dist_num <- dist(scale(emp_att_num), method = "euclidean")

## visualize distances
fviz_dist(emp_dist_num,
          gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07"),
          show_labels = FALSE)
```



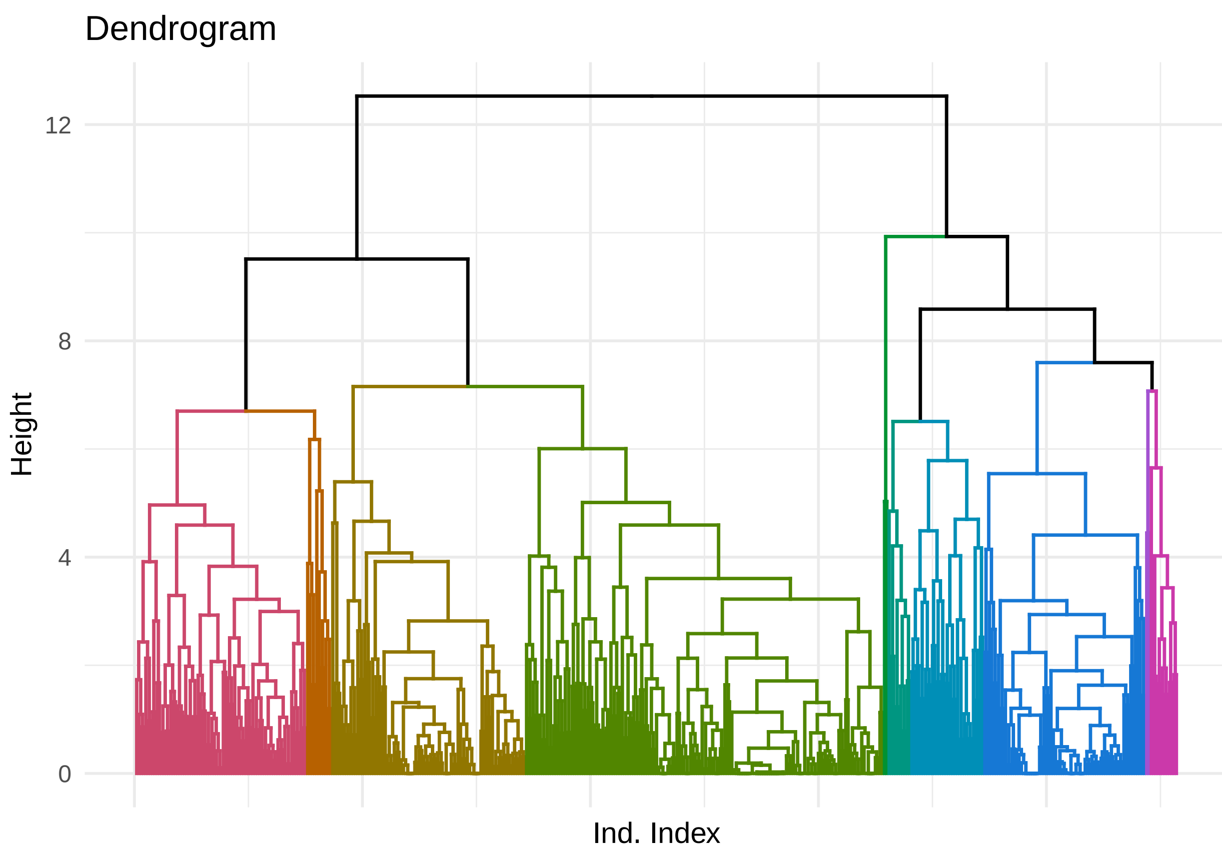

```
#### Q2.2
### agglomerative hierarchical clustering with complete linkage
## run clustering
emp_hclust <- hclust(emp_dist_num, method = "complete")

##head
# merge sequence
head(emp_hclust$merge, n=20)
```

```
##       [,1] [,2]
##  [1,]   -2 -110
##  [2,] -127    1
##  [3,] -144    2
##  [4,] -149    3
##  [5,] -175    4
##  [6,] -219    5
##  [7,] -234    6
##  [8,] -283    7
##  [9,] -338    8
## [10,] -355    9
## [11,] -371   10
## [12,]   -8  -67
## [13,] -115   12
## [14,]   -9 -238
## [15,] -276   14
## [16,]  -13 -155
```

```
## [17,]  -21 -229
## [18,] -308   17
## [19,]  -22 -212
## [20,] -264   19
```

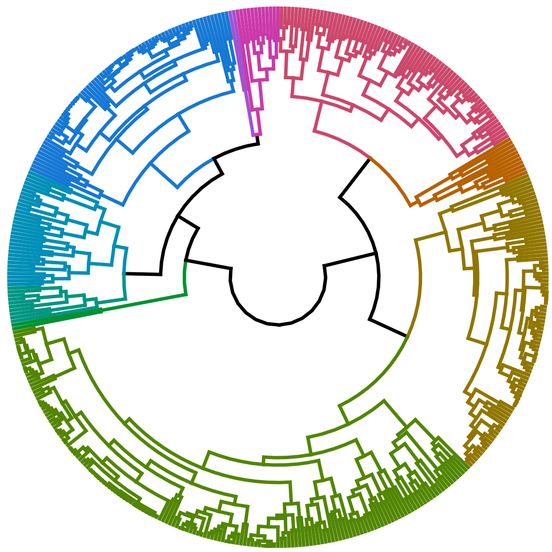
```r
#### Q2.3
### visualize
## create dendrogram object
dend_emp_hclust <- as.dendrogram(emp_hclust)

## set attributes of dendrogram
# overwrite dendrogram
dend_emp_hclust <- dend_emp_hclust %>%
  # set colors of branches and number of cuts
  set("branches_k_color", k = 10) %>%
  # set width of branches
  set("branches_lwd", 0.6) %>%
  # set color of labels
  set("labels_colors",
      value = c("darkslategray")) %>%
  # set size of labels
  set("labels_cex", 0)

## convert to ggplot object
dend_emp_hclust <- as.ggdend(dend_emp_hclust)

## traditional dendrogram plot
# call plot
ggplot(dend_emp_hclust) +
  # minimal theme
  theme_minimal() +
  # remove x-axis labels
  theme(axis.text.x = element_blank()) +
  # labels
  labs(x = "Ind. Index", y = "Height", title = "Dendrogram")
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```




```r
## radial dendrogram plot
# call plot
ggplot(dend_emp_hclust) +
  # minimal theme
  scale_y_reverse(expand = c(0.2, 0.2)) +
  # polar coordinates
  coord_polar(theta = "x")
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.

## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

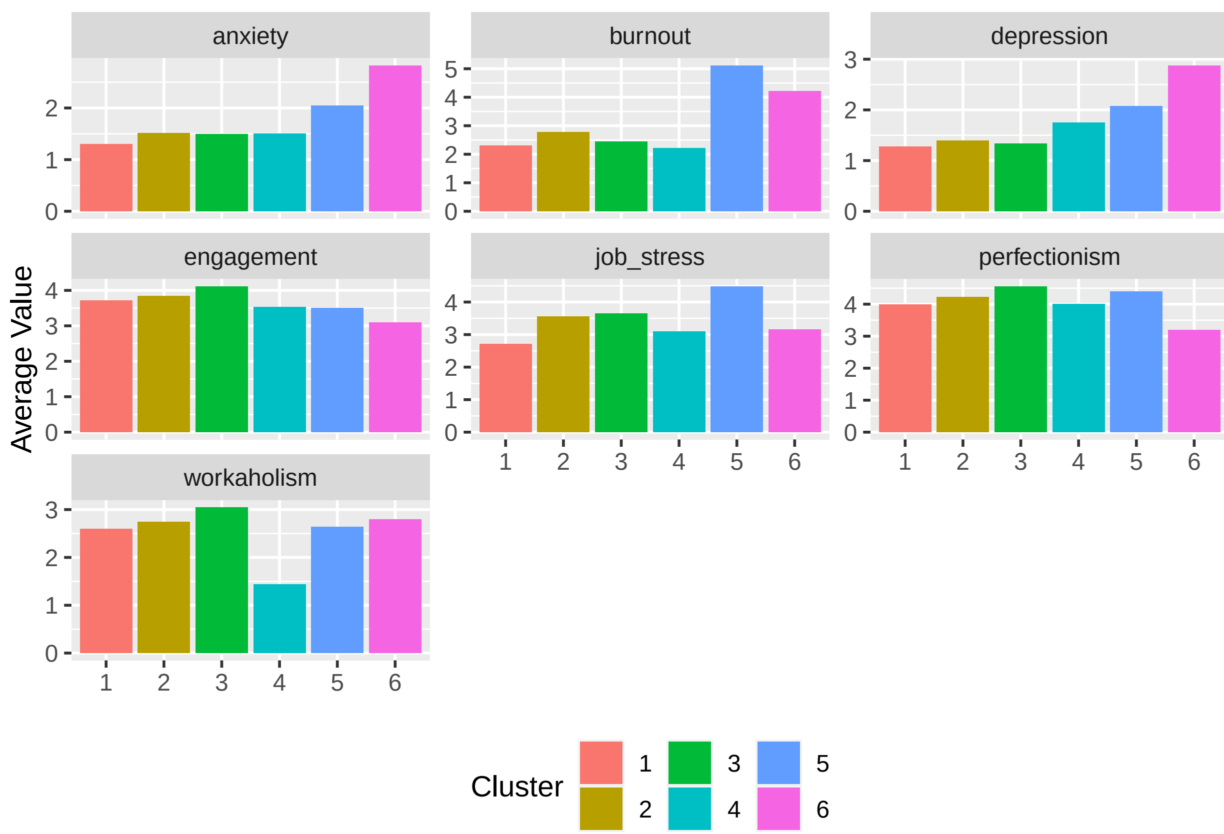


```r
#### Q2.4
### compute cluster statistics
## call data
emp_att_num %>%
  ## add cluster variable
  mutate(hier_clust = cutree(emp_hclust, k = 10)) %>%
  ## count individuals
  count(hier_clust)
```

```
## # A tibble: 10 x 2
##    hier_clust     n
##         <int> <int>
##  1          1    75
##  2          2   157
##  3          3    85
##  4          4    11
##  5          5    32
##  6          6    71
##  7          7    12
##  8          8    10
##  9          9     2
## 10         10     2
```

```r
## call data
emp_att_num %>%
```

```r
  ## add cluster variable
  mutate(hier_clust = cutree(emp_hclust, k = 10)) %>%
  ## filter
  filter(hier_clust %in% 1:6) %>%
  ## group by cluster
  group_by(hier_clust) %>%
  ## summarize
  summarize_all(list(~mean(.))) %>%
  ## pivot longer
  pivot_longer(cols = -hier_clust, names_to = "var", values_to = "value") %>%
  ## mutate
  mutate(hier_clust = as_factor(hier_clust)) %>%
  ## ggplot
  ggplot(aes(x = hier_clust, y = value, fill = hier_clust)) +
    ## bar plot
    geom_col() +
    ## facet wrap
    facet_wrap(~var, scales = "free_y") +
    ## labels
    labs(y = "Average Value", fill = "Cluster") +
    ## change legend position and remove x-axis label
    theme(legend.position = "bottom",
          axis.title.x = element_blank())
```



Task 3: K-means Clustering

Set the random seed to *27* with **set.seed(27)**. Then, apply the K-means clustering algorithm on **emp_att_num** with *k* (i.e., number of centers) set to *8* and number of starts set to *25*. Name the result **emp_kmeans**. Examine the centroids and size of each resulting cluster. Apply **fviz_cluster()** to visualize the solution on the first two principal components.

Question 3.1: What is the centroid for the *fourth* cluster on **workaholism**? What is the size of the *seventh* cluster? Examining the plot, to which cluster does observation *414* belong?

Response 3.1: *Centroid for the fourth cluster: 2.674516. Size of the seventh cluster: 101. Observation 414 belongs to cluster 8.*

Use **fviz_nbclust** and set method to **wss** to determine the optimal number of clusters. Use **fviz_nbclust** and set method to **silhouette** to determine the optimal number of clusters. Use **clusGap()** on **emp_att_num** setting **K.max** to *15* and **B** to *100* and naming the result of **emp_kmeans_gap**. Ignore any warning messages. Use **fviz_gap_stat** on **emp_kmeans_gap** to determine the optimal number of clusters.

Question 3.2: What is the optimal number of clusters when examining the total within sum of square? What is the optimal number of clusters when examining the average silhouette width? What is the optimal number of clusters when examining the gap statistic?

Response 3.2: *WSS: 4. Average silhouette: 2. Gap: 6.*

Apply the K-means clustering algorithm again on **emp_num_att** this time using *6* clusters while keeping the number of starts to *25*. Overwrite the previous **emp_kmeans** result with the new result. Plot the average values on the original variables of the *6* clusters referencing the correct part of the output of **emp_kmeans**. The plot should be a bar plot of each cluster on the x-axis. The height of the bars should represent the average value on each of the original variables. Apply a facet wrap using the original variables. See the lecture script.

Question 3.3: Which cluster has the highest average **burnout**? Which cluster has the highest average **depression**?

Response 3.3: *Cluster 6 has the highest average burnout. Cluster 1 has the highest average depression.*

```r
#### Q3.1
### k-means clustering
## set seed
set.seed(27)
## run clustering
emp_kmeans <- kmeans(emp_att_num,
                     # number of clusters
                     centers = 8,
                     # number of random sets
                     nstart = 25)


## examine centroids
emp_kmeans$centers
```

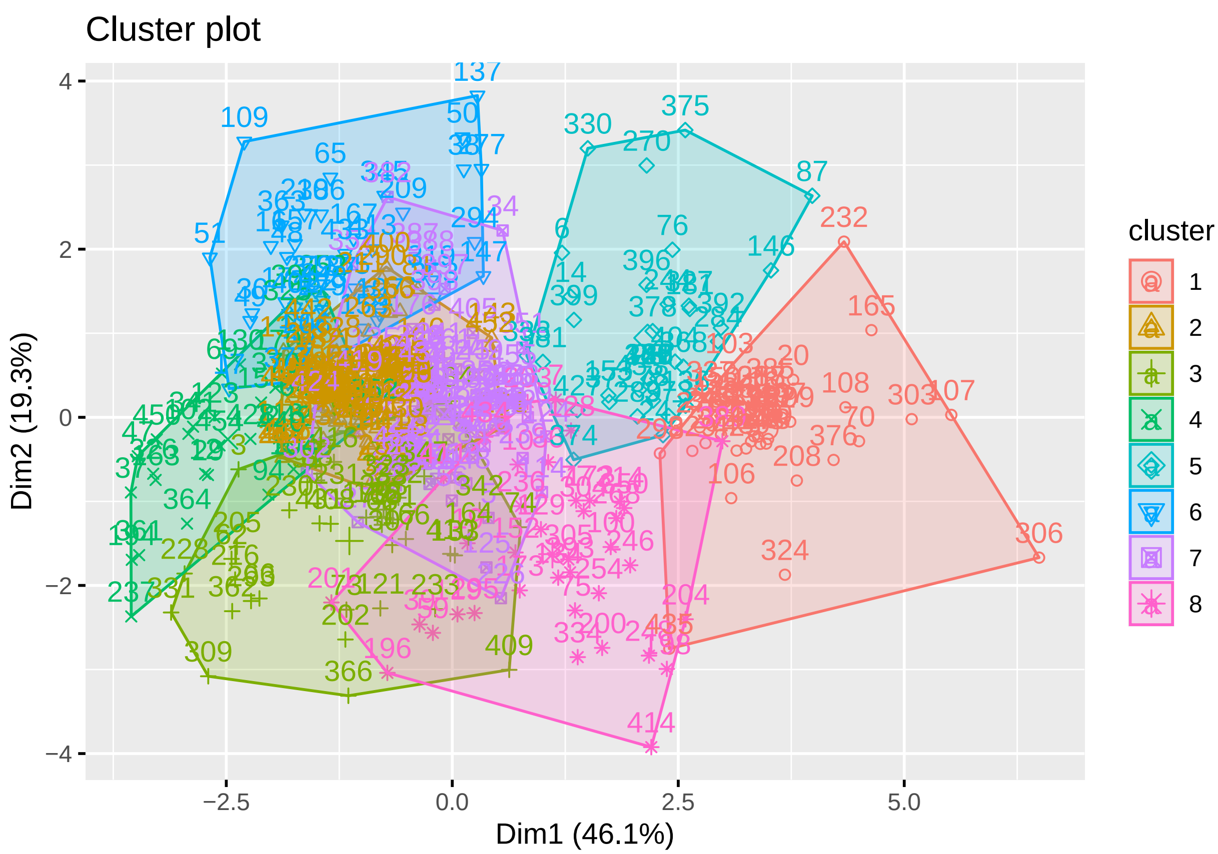
```
##    burnout job_stress workaholism  anxiety perfectionism engagement depression
## 1 4.404800   3.238400    2.868200 3.079000      3.155800   3.061200   3.093600
## 2 2.389052   3.323534    2.758621 1.431638      4.256552   3.861207   1.313362
## 3 2.756744   4.349535    3.087442 1.580930      4.509070   4.118605   1.501628
## 4 1.174516   3.967419    2.674516 1.159355      4.584516   4.387097   1.051613
## 5 3.840789   2.905526    2.409211 2.280000      3.524737   2.998947   2.482105
```

```
## 6 1.642250   2.350500    2.567750 1.150000      4.030000   3.788750   1.170000
## 7 3.475446   3.364356    2.724356 1.658416      4.144554   3.645545   1.521584
## 8 5.525263   4.323421    2.827105 1.928421      4.190000   3.496053   2.022632
```

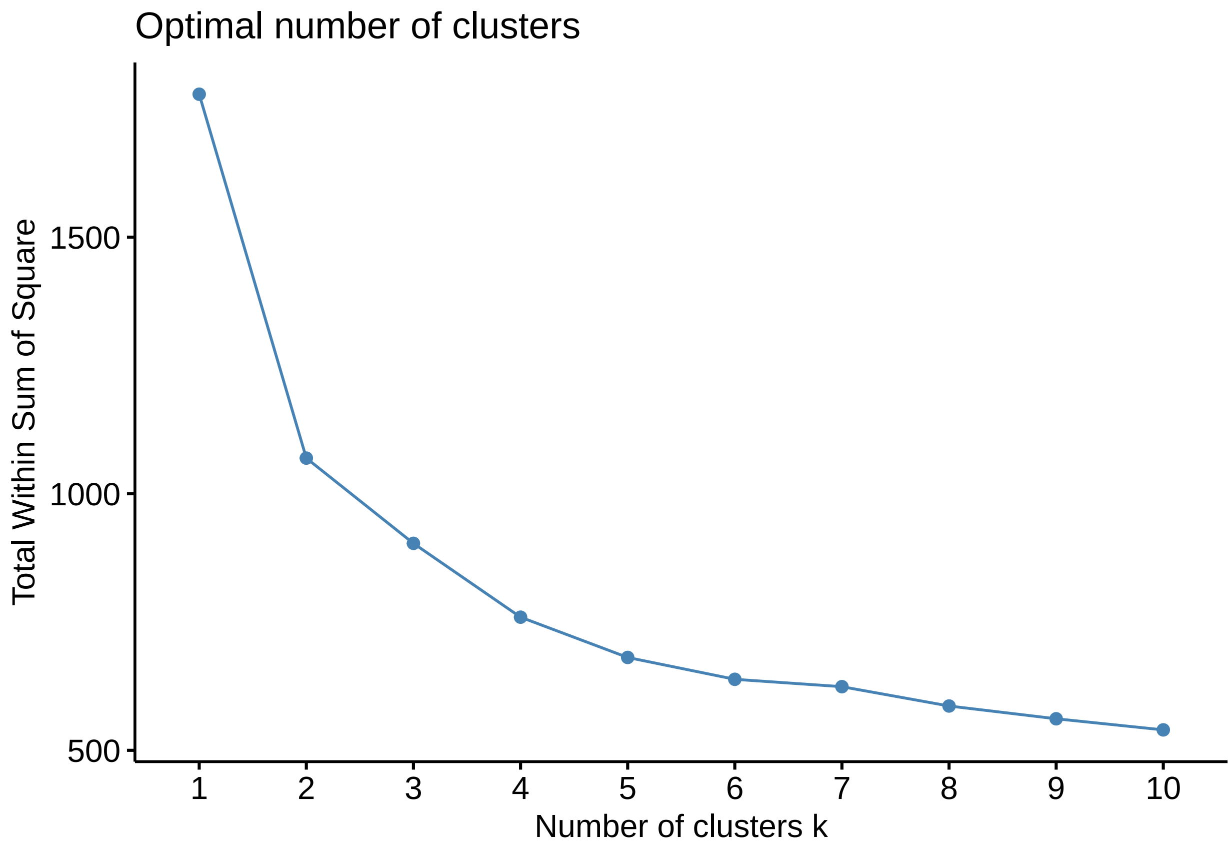
```
## cluster size
emp_kmeans$size
```

```
## [1]  50 116  43  31  38  40 101  38
```

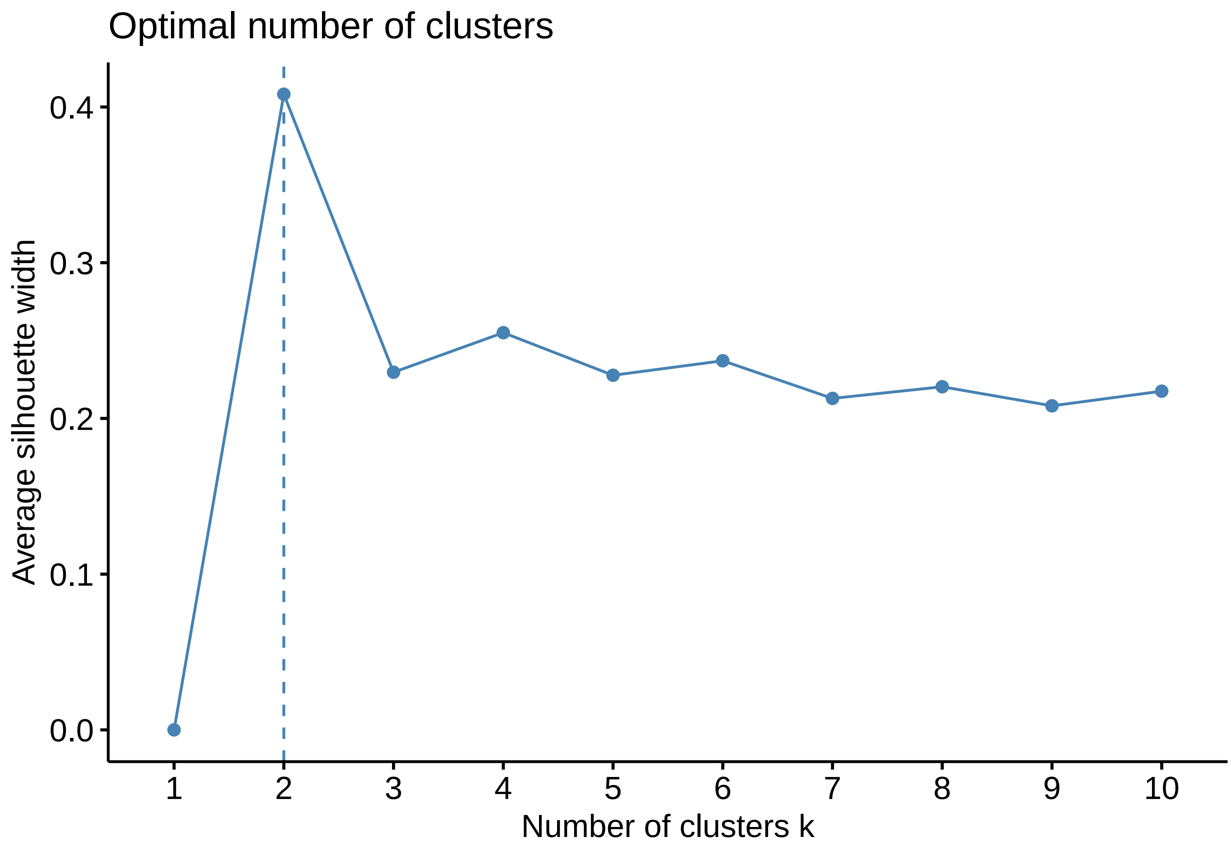
```
### visualize results
## use first two principal components of original variables
fviz_cluster(emp_kmeans, data = emp_att_num)
```




```
#### Q3.2
### choosing the number of clusters
## total within-cluster sum of squares
fviz_nbclust(emp_att_num, kmeans, method = "wss")
```



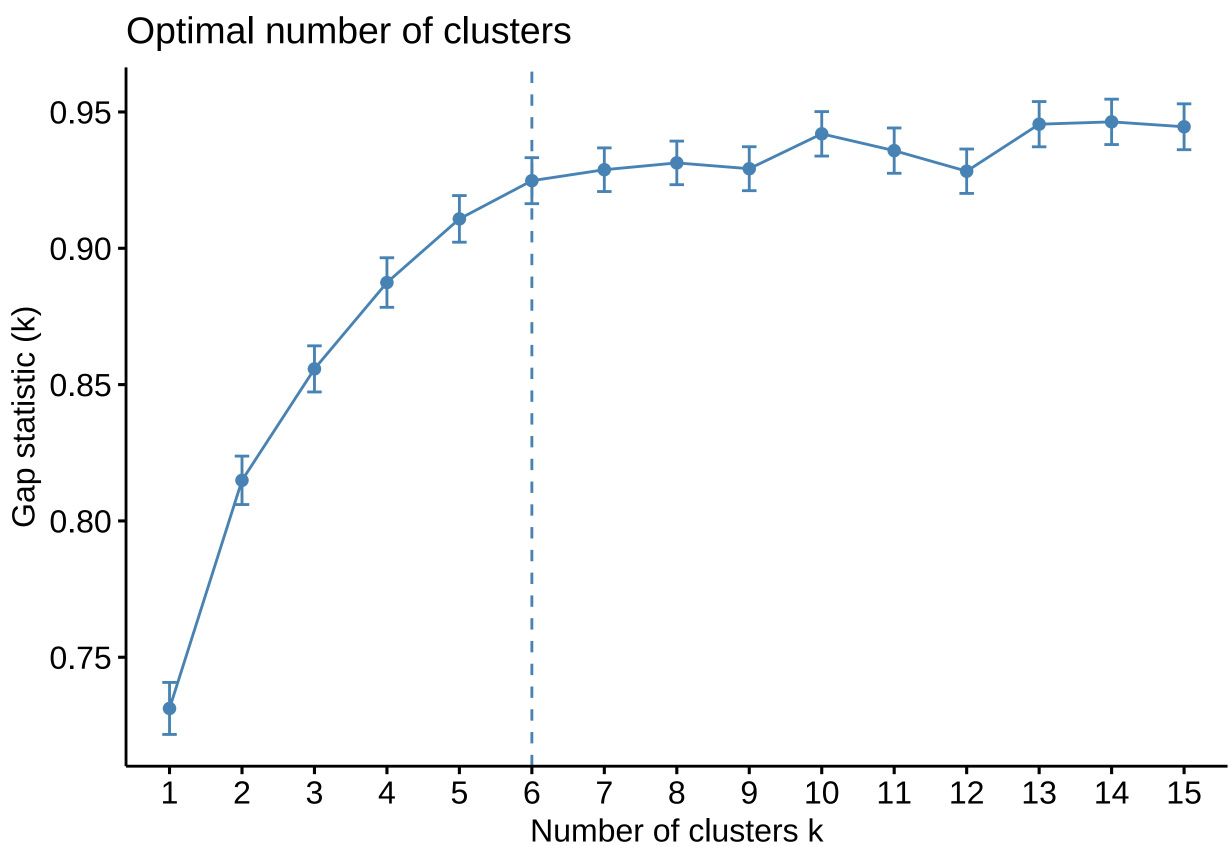

```
## average silhouette method
fviz_nbclust(emp_att_num, kmeans, method = "silhouette")
```




```r
## gap statistic
# estimate statistic
emp_kmeans_gap <- clusGap(emp_att_num, kmeans, nstart = 25,
                          K.max = 15, B = 100)
```

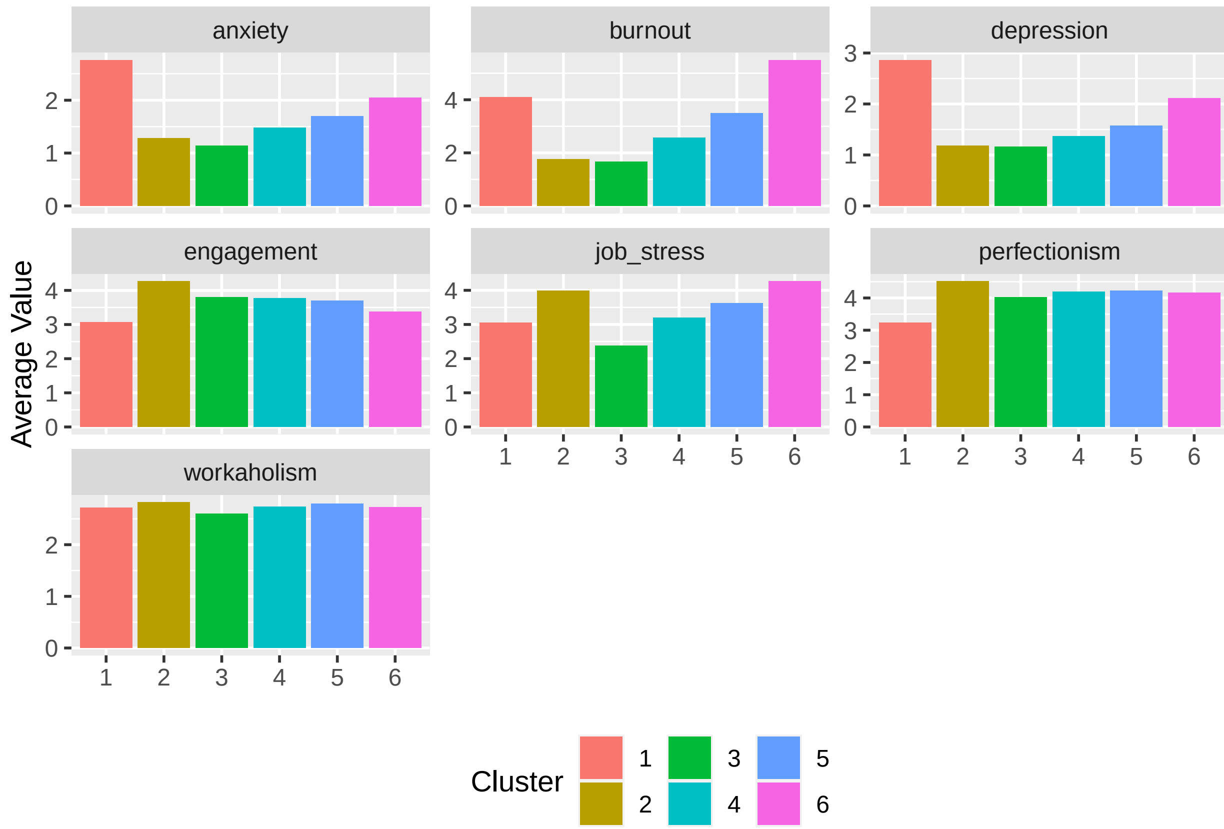
```
## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations
```

```r
# plot
fviz_gap_stat(emp_kmeans_gap)
```




```
#### Q3.3
## run clustering
emp_kmeans <- kmeans(emp_att_num,
                     # number of clusters
                     centers = 6,
                     # number of random sets
                     nstart = 25)
## call data
emp_kmeans$centers %>%
  ## convert to tibble
  as_tibble() %>%
  ## add cluster variable
  rowid_to_column(var = "kmeans_clust") %>%
  ## pivot longer
  pivot_longer(cols = -kmeans_clust, names_to = "var", values_to = "value") %>%
  ## mutate
  mutate(kmeans_clust = as_factor(kmeans_clust)) %>%
  ## ggplot
  ggplot(aes(x = kmeans_clust, y = value, fill = kmeans_clust)) +
    ## bar plot
    geom_col() +
    ## facet wrap
    facet_wrap(~var, scales = "free_y") +
    ## labels
    labs(y = "Average Value", fill = "Cluster") +
    ## change legend position and remove x-axis label
```

```
  theme(legend.position = "bottom",
        axis.title.x = element_blank())
```



Task 4: Partitioning Around Medoids

Create a new data object named **emp_att_mix** consisting of the following mixed variables: **burnout**, **job_stress**, **workaholism**, **anxiety**, **perfectionism**, **engagement**, **depression**, **useNowFlextime**, **married**, **partner_employment**, and **gender**. Compute the Gower distance matrix based on **emp_att_mix** and name the result **emp_dist_mix**. Apply **summary()** to **emp_dist_mix**.

Question 4.1: What is the median dissimilarity?

Response 4.1: *0.2491*.

Apply **pam()** to **emp_dist_mix** to determine the optimal number of clusters from *2* to *20* by calculating the average silhouette width for each cluster quantity. Name the result **emp_pam_sil**. Plot the average silhouette widths for the cluster quantities.

Question 4.2: How many clusters is optimal based on this plot?

Response 4.2: *7*.

Apply **pam()** to **emp_dist_mix** with *5* clusters. Name the result **emp_pam**. Then, create two plots. First, for the numeric variables, plot the average values on the original variables of the *5* clusters referencing the correct part of the output of **emp_pam**. The plot should be a bar plot of each cluster on the x-axis. The height of the bars should represent the average value on each of the original variables. Apply a facet wrap using the original variables. Second, for the factor variables, plot the percentage values on the original variables of the *5* clusters referencing the correct part of the output of **emp_pam**. Use a facet grid with the factor variables in the columns and clusters in the rows. The x-axis should represent the levels of the factors. The y-axis should represent percentage of individuals in each level of the factor variable for a particular cluster. See the lecture script.

Question 4.3: Which cluster has the lowest average **perfectionism**? Which two clusters consisted of *100%* unmarried employees? Which cluster consisted of *100%* of employees with employed partners?

Response 4.3: *Cluster 4 has the lowest average perfectionism. Clusters 3 and 5 consist of 100% of unmarried employees. Clusters 2 and 4 consist of 100% of employees with employed partners.*

Set the random seed to *57* with **set.seed(57)**. Then, apply **Rtsne()** to **emp_dist_mix** and save the result as **tsne_mix**. Plot the clusters from **emp_pam** on the resulting two-dimensional solution in **tsne_mix**.

Question 4.4: Overall, do the clusters look separated in the two-dimensional space? Which clusters mix data points?

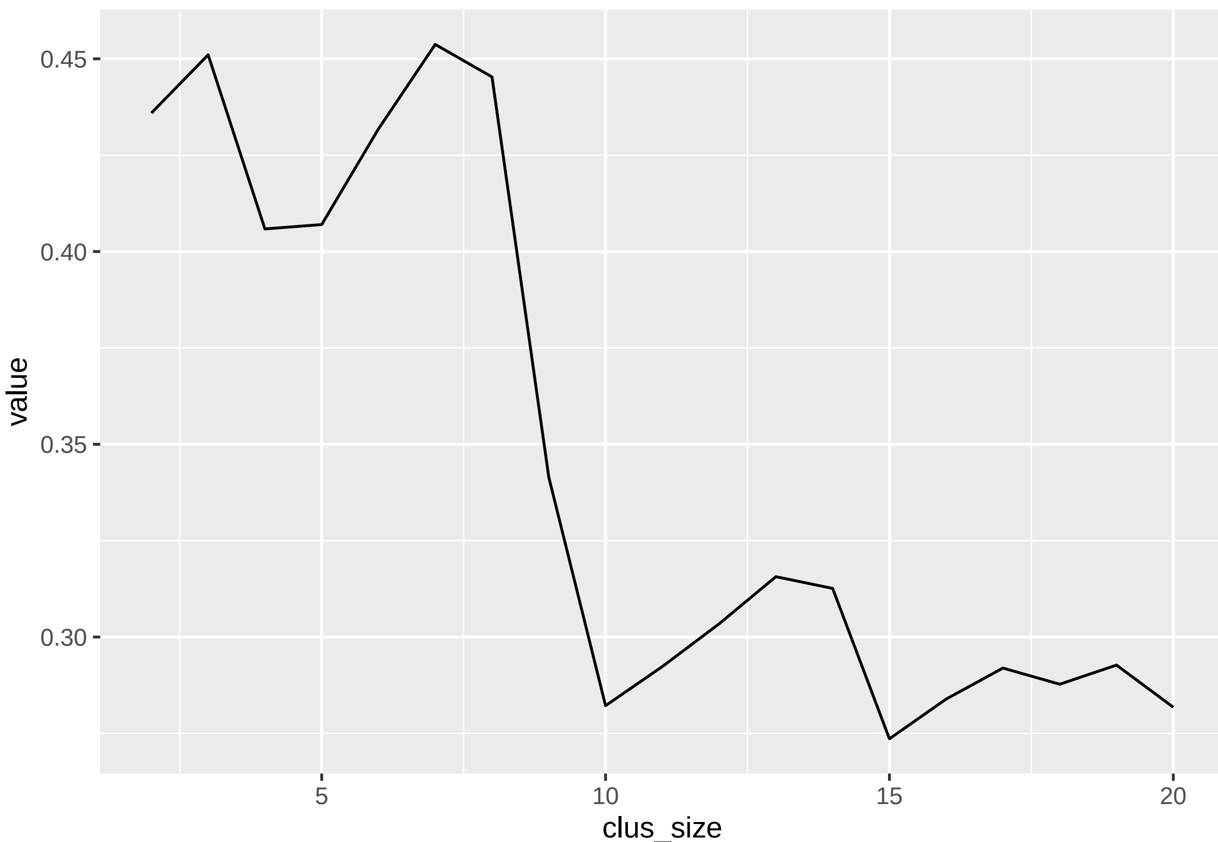
Response 4.4: *The clusters look separated. Clusters 4 and 5 mix data points, as well as the following clusters: 3 and 5, 1 and 2, 1 and 4, 2 and 4.*

```
#### Q4.1
### mixed variables data object
## create data object
emp_att_mix <- emp_att_data %>%
  ## select variables of choice
         # numeric variables
  select(burnout, job_stress, workaholism, anxiety, perfectionism, engagement, depression,
         # factor variables
         useNowFlextime, married, partner_employment, gender)

### distance between employees on set of variables
## calculate Gower distance
emp_dist_mix <- daisy(emp_att_mix, metric = "gower")

# summary
summary(emp_dist_mix)
```

```
## 104196 dissimilarities, summarized :
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.1584  0.2491  0.2452  0.3306  0.6634
## Metric :  mixed ;  Types = I, I, I, I, I, I, I, N, N, N, N
## Number of objects : 457
```

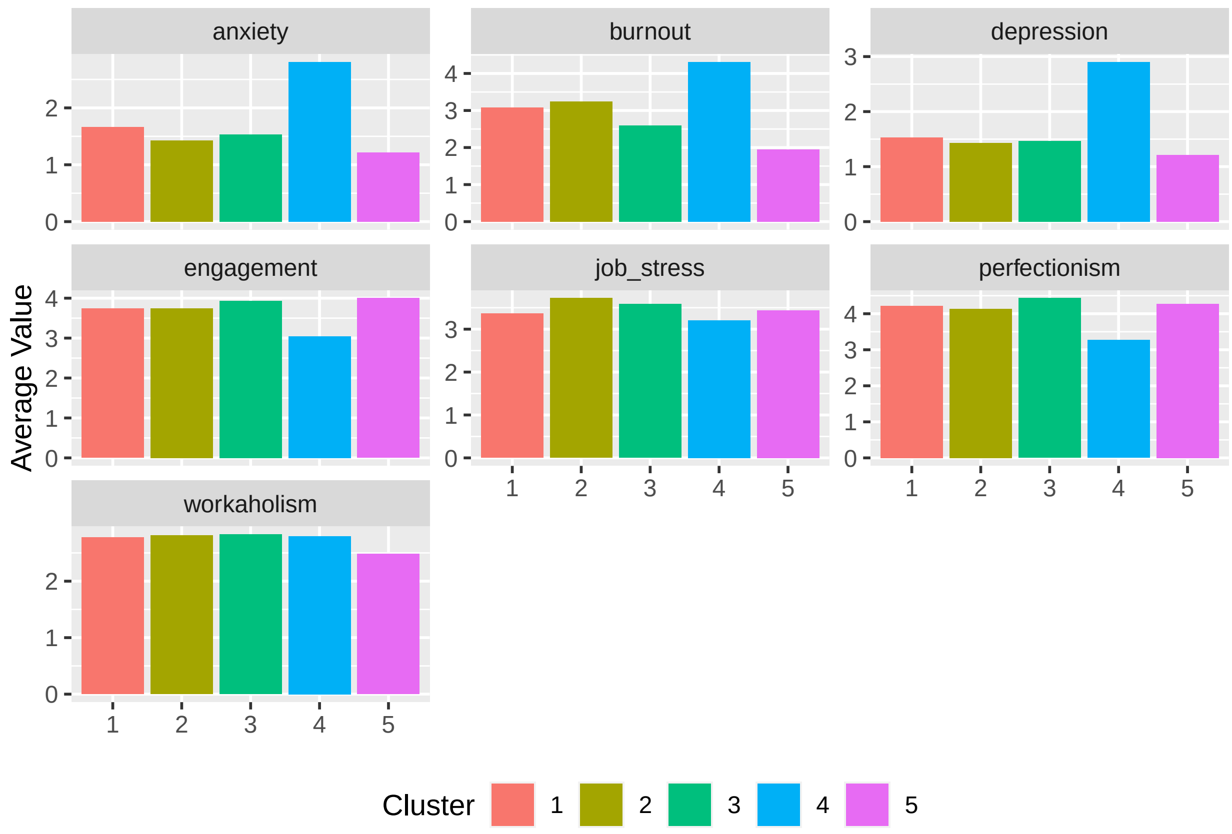
```
#### Q4.2
### pam clustering
### choosing the number of clusters
## iterate over different number of clusters
emp_pam_sil <- map_dbl(2:20, function(.x) {
    # run pam for each cluster size
    fit <- pam(emp_dist_mix, diss = TRUE, k = .x)
    # extract average silhouette width
    fit$silinfo$avg.width
  })

## call data
emp_pam_sil %>%
```

```
  ## convert to tibble
  as_tibble() %>%
  ## add number of clusters
  mutate(clus_size = 2:20) %>%
  ## plot
  ggplot(aes(x = clus_size, y = value)) +
    ## line geometry
    geom_line()
```




```
#### Q4.3
### pam clustering
## run clustering
emp_pam <- pam(emp_dist_mix,
               # use dissimilarity matrix
               diss = TRUE,
               # number of clusters
               k = 5)

### plot numeric variables against clusters
## call data
emp_att_mix %>%
  ## select numeric
  select_if(is.numeric) %>%
  ## add cluster variable
  mutate(pam_clust = as_factor(emp_pam$clustering)) %>%
```

```
## group by cluster
group_by(pam_clust) %>%
## summarize
summarize_all(list(~mean(.))) %>%
## pivot longer
pivot_longer(cols = -pam_clust, names_to = "var", values_to = "value") %>%
## ggplot
ggplot(aes(x = pam_clust, y = value, fill = pam_clust)) +
  ## bar plot
  geom_col() +
  ## facet wrap
  facet_wrap(~var, scales = "free_y") +
  ## labels
  labs(y = "Average Value", fill = "Cluster") +
  ## change legend position and remove x-axis label
  theme(legend.position = "bottom",
        axis.title.x = element_blank())
```




```
### plot factors against clusters
## call data
emp_att_mix %>%
  ## select factors
  select_if(is.factor) %>%
  ## add cluster variable
  mutate(pam_clust = as_factor(emp_pam$clustering)) %>%
```

```
## pivot longer
pivot_longer(cols = -pam_clust, names_to = "var", values_to = "value") %>%
## count
count(pam_clust, var, value) %>%
## group by
group_by(pam_clust, var) %>%
## mutate
mutate(pct = n/sum(n)) %>%
## ggplot
ggplot(aes(x = value, y = pct,
           fill = pam_clust)) +
  ## bar plot
  geom_col() +
  ## facet wrap
  facet_grid(pam_clust ~ var, scales = "free_x") +
  ## y-axis
  scale_y_continuous(labels = scales::percent_format()) +
  ## labels
  labs(y = "Count", fill = "Cluster") +
  ## change legend position and remove x-axis label
  theme(legend.position = "none",
        axis.title.x = element_blank(),
        axis.text.x = element_text(angle = 45, hjust = 1))
```

```r
## set seed
set.seed(57)
### visualize results
## use t-distributed stochastic neighborhood embedding
tsne_mix <- Rtsne(emp_dist_mix, is_distance = TRUE)

## extract locations
tsne_mix$Y %>%
  ## convert to tibble
  as_tibble(.name_repair = "minimal") %>%
  ## set names
  setNames(c("X", "Y")) %>%
  ## mutate
  mutate(pam_clust = as_factor(emp_pam$clustering)) %>%
  ## plot
  ggplot(aes(x = X, y = Y, color = pam_clust)) +
    ## point geometry
    geom_point()
```