01_initialization_kmeans

Frances Lin

11/11/2020

Description

This .Rmd file calculates the initial estimates for the first EM iteration using K-means clustering. All related results can be found in the results folder.

Load packages

```
library(here)
library(tidyverse)
library(tibble)
library(stats)
```

The author of this article suggests that it is common to use K-means clustering (hard labelling) to obtain the initial estimates for EM algorithm (the soft labelling).

Load data

```
data <- readRDS(here("results", "data.rds"))
head(data)
```

```
## # A tibble: 6 x 2
##   component  value
##   <chr>      <dbl>
## 1 A         -1.21
## 2 A          0.277
## 3 A          1.08
## 4 A         -2.35
## 5 A          0.429
## 6 A          0.506
```

Perform k-mean clustering

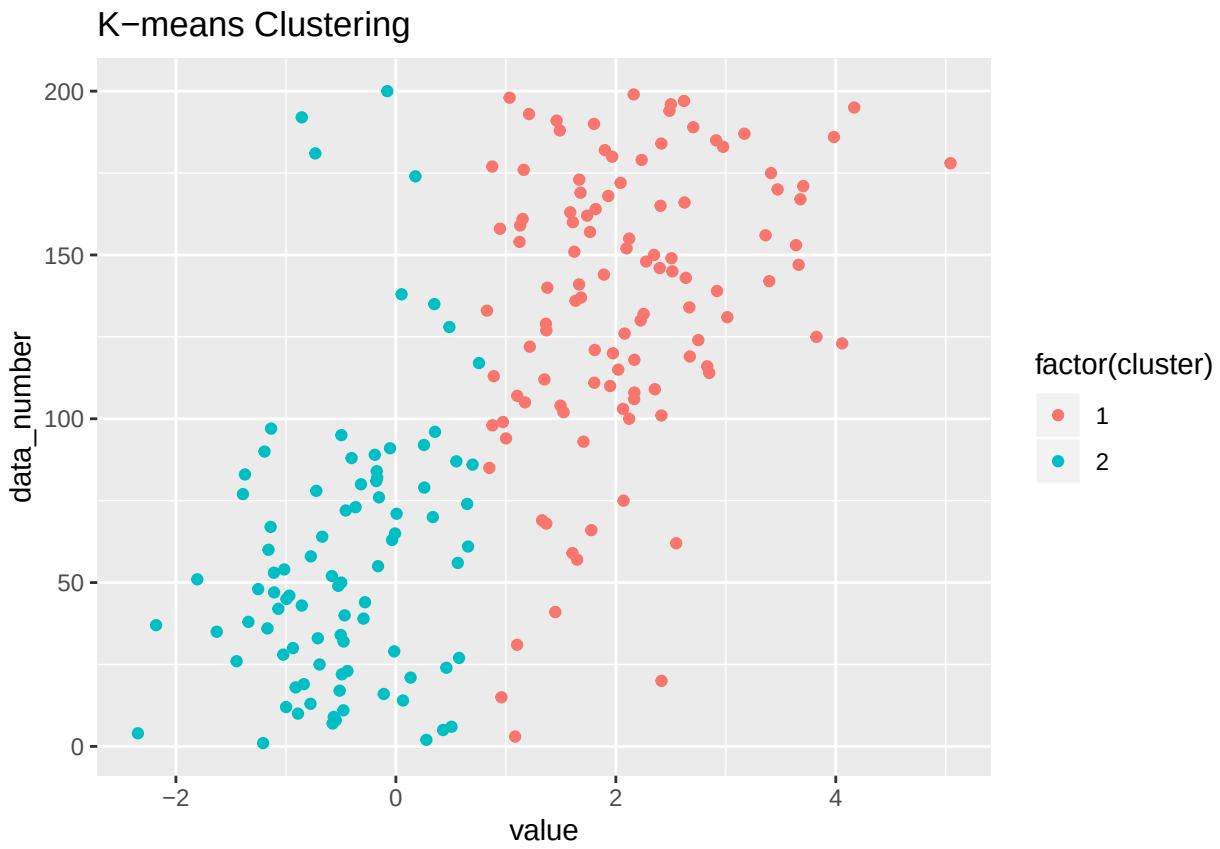
```
# K-means clustering is "hard" because we assign k = 2
# (i.e. a data point is either 1 or 2)
value <- data$value
kmeans <- kmeans(x = value, centers = 2)
kmeans_cluster <- kmeans$cluster
```

Plot k-mean clustering

```
df_kmeans <- tibble(
  value = value,
  cluster = kmeans_cluster
)
head(df_kmeans)
```

```
## # A tibble: 6 x 2
##    value cluster
##    <dbl>   <int>
## 1 -1.21        2
## 2  0.277       2
## 3  1.08        1
## 4 -2.35        2
## 5  0.429       2
## 6  0.506       2
```

```
df_plot_kmeans <- df_kmeans %>%
  mutate(data_number = row_number()) %>%
  ggplot(aes(x = value, y = data_number, color = factor(cluster))) +
  geom_point() +
  ggtitle("K-means Clustering")
df_plot_kmeans
```



Store the values from K-means clustering, which then become initial estimates for EM algorithm

Estimate mean (μ_1, μ_2) and sd (σ_1, σ_2)

```
df_summary_kmeans <- df_kmeans %>%
  group_by(cluster) %>%
  summarize(
    mean = mean(value),
    var = var(value),
    sd = sd(value),
    size = length(value)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
df_summary_kmeans
```

```
## # A tibble: 2 x 5
##   cluster   mean   var    sd  size
##     <int>  <dbl> <dbl> <dbl> <int>
## 1       1  2.10  0.737 0.859   110
## 2       2 -0.471 0.445 0.667    90
```

Estimate weighting probability π

```
df_summary_kmeans <- df_summary_kmeans %>%
  mutate(
    pi = size / sum(size))
df_summary_kmeans
```

```
## # A tibble: 2 x 6
##   cluster   mean   var    sd  size    pi
##     <int>  <dbl> <dbl> <dbl> <int> <dbl>
## 1       1  2.10  0.737 0.859   110  0.55
## 2       2 -0.471 0.445 0.667    90  0.45
```

Now we can pass the values to EM algorithm

Save out results

```
write_rds(df_kmeans, here("results", "df_kmeans.rds"))
write_rds(df_plot_kmeans, here("results", "df_plot_kmeans.rds"))
write_rds(df_summary_kmeans, here("results", "df_summary_kmeans.rds"))
```