STAT-S675

Homework 9

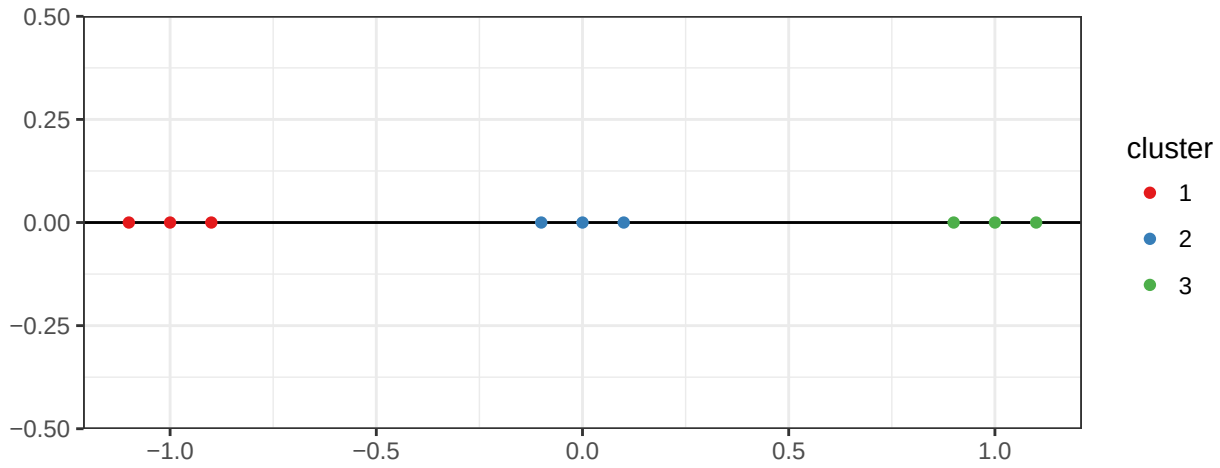
John Koo

Link to assignment

Exercise 8.5.1

Let the points $\{-1.1, -1, -.9, -.1, 0, .1, .9, 1, 1.1)\} \subset \mathbb{R}$ be grouped into three clusters. Then a natural clustering of these nine points might be to have the first three points in one cluster, the middle three points in another cluster, and the last three points in the last cluster.

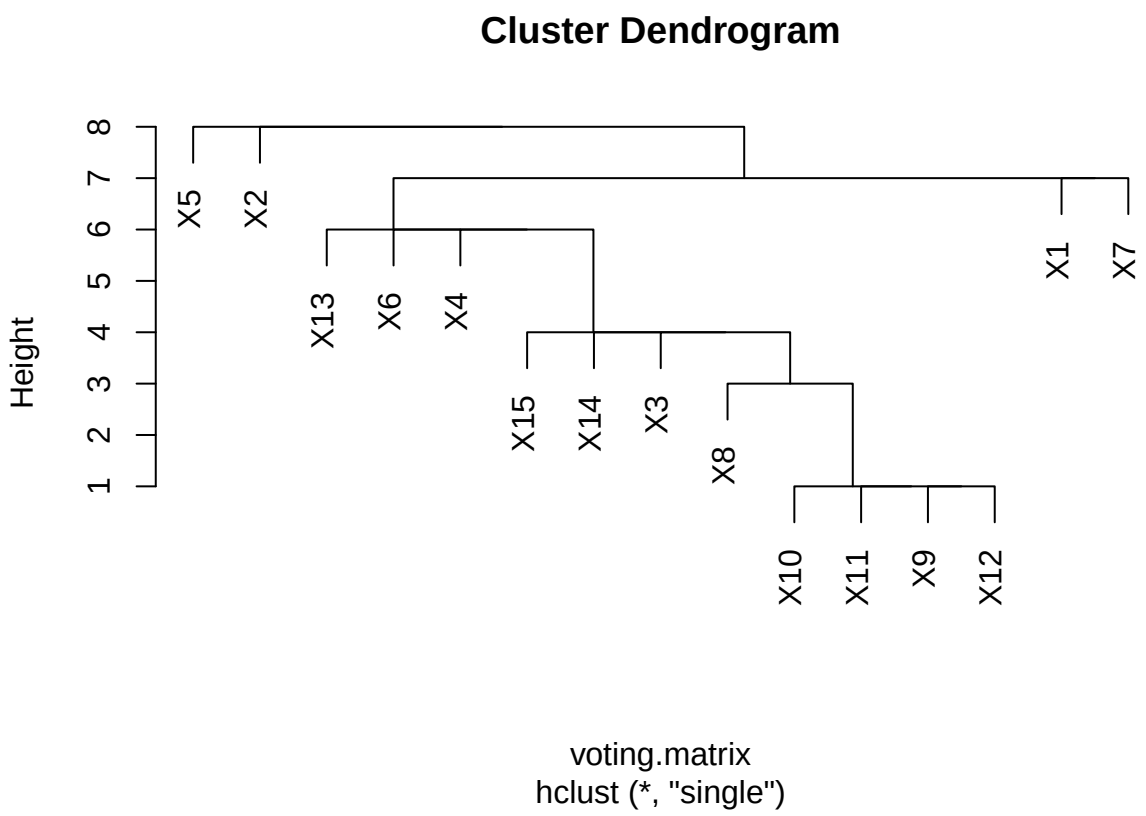
```
library(ggplot2)
theme_set(theme_bw())

cluster.df <- dplyr::data_frame(
  x = c(-1.1, -1, -.9, -.1, 0, .1, .9, 1, 1.1),
  cluster = rep(c('1', '2', '3'), each = 3)
)

ggplot(cluster.df) +
  coord_fixed() +
  geom_hline(yintercept = 0) +
  scale_colour_brewer(palette = 'Set1') +
  geom_point(aes(x = x, y = 0, colour = cluster)) +
  labs(x = NULL, y = NULL)
```



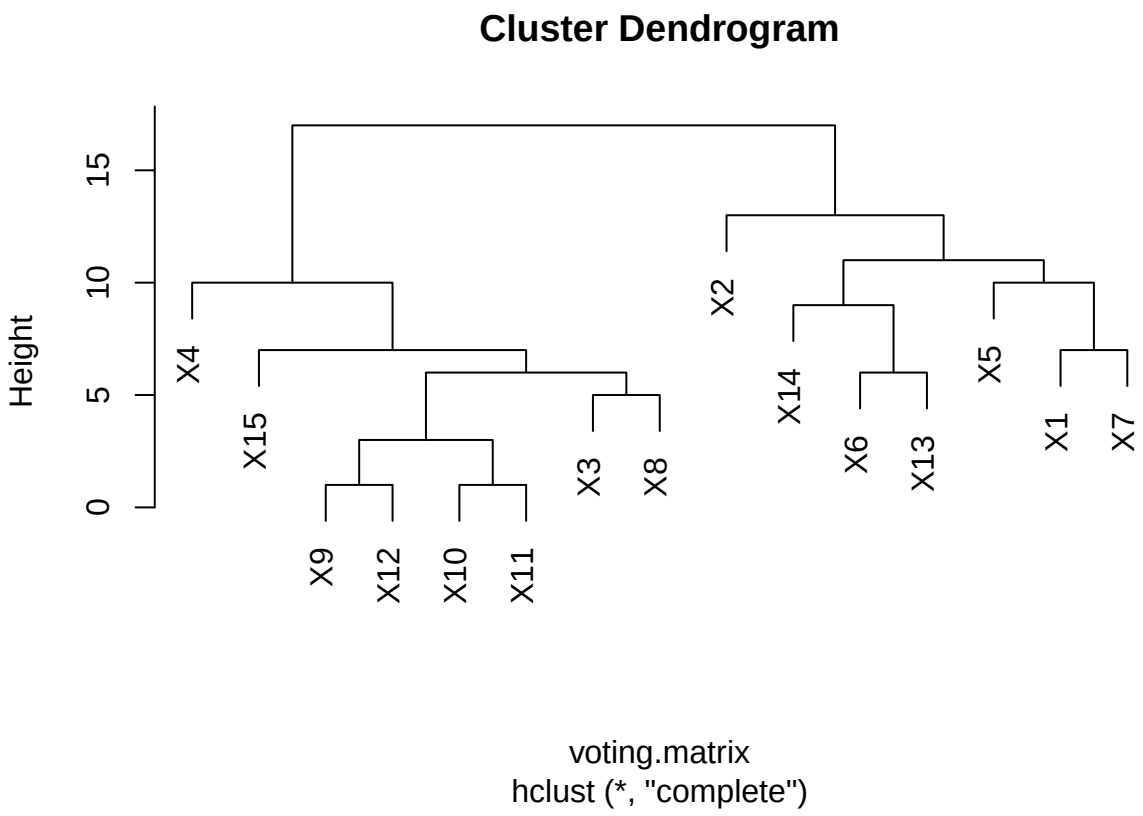
Then $ES_{12} = .8 = ES_{23}$ and $ES_{13} = 1.8$. But $1.8 > .8 + .8$. Therefore, single-linkage doesn't satisfy the triangle inequality and so is not EDM-1.

Exercise 8.5.2

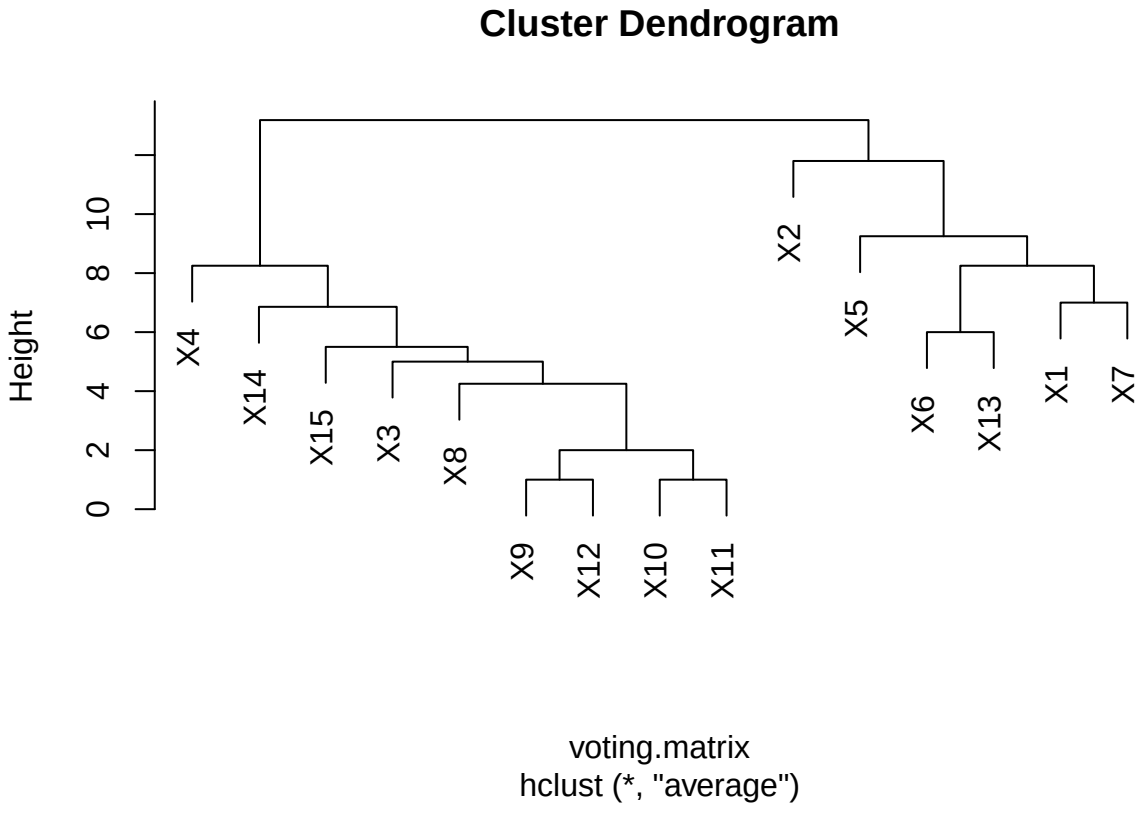
```
import::from(readr, read_table2)
import::from(magrittr, `%>%`)

voting.url <- 'http://pages.iu.edu/~mtrosset/Courses/675/congress.dat'
voting.matrix <- read_table2(voting.url,
                             col_names = FALSE) %>%
  as.dist()

single.link <- hclust(voting.matrix, 'single')
plot(single.link)
```




```
complete.link <- hclust(voting.matrix, 'complete')
plot(complete.link)
```



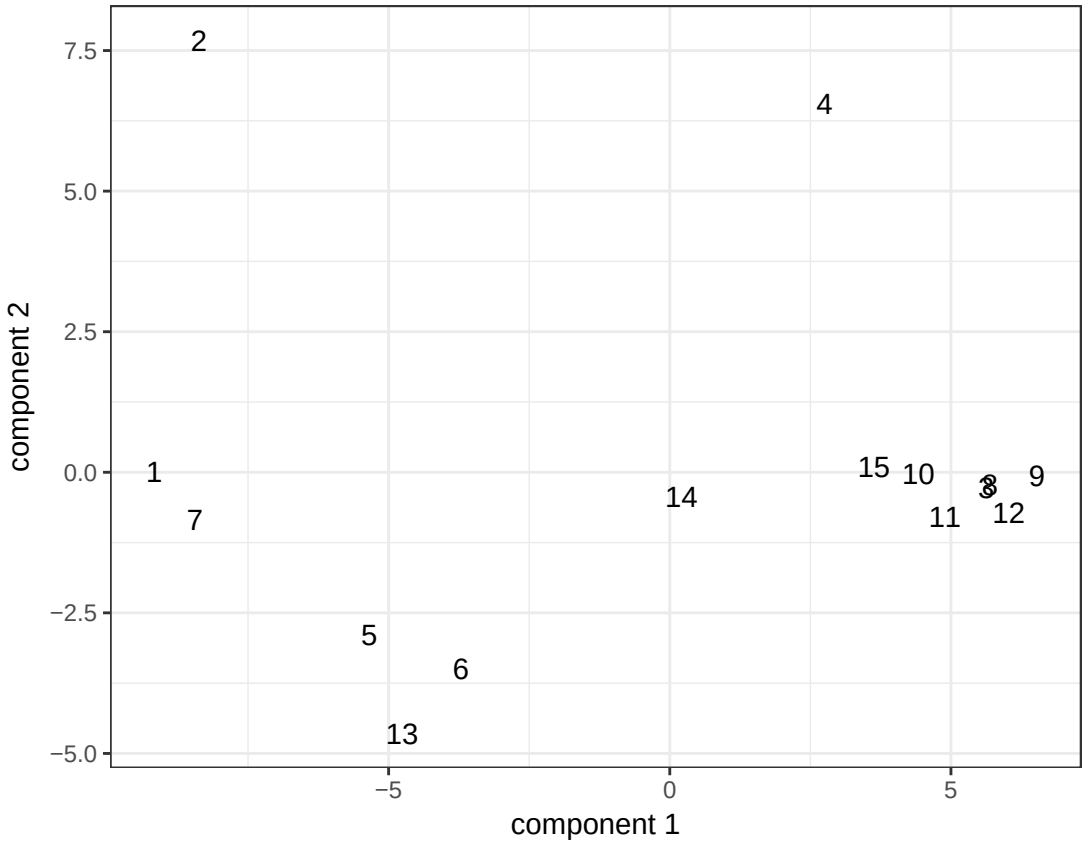

```
average.link <- hclust(voting.matrix, 'average')
plot(average.link)
```



The complete and average linkage methods had similar results compared to the single linkage method.

We can also try looking at an embedding of the dissimilarity matrix:

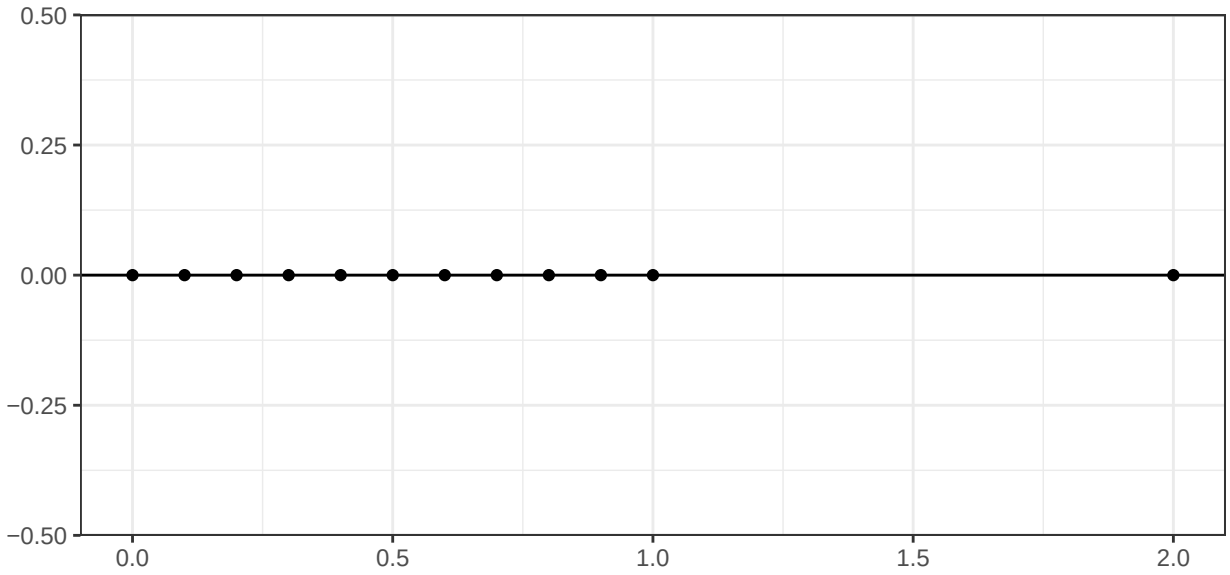
```
cmdscale(voting.matrix) %>%
  as.data.frame() %>%
  dplyr::mutate(id = seq(n())) %>%
  ggplot() +
  coord_fixed() +
  geom_text(aes(x = V1, y = V2, label = id)) +
  labs(x = 'component 1', y = 'component 2')
```



Based on an (imperfect) 2-dimensional embedding, the complete and average linkage methods were able to create two clusters based on the first component. It appears that 3 and 8 would naturally fall within a cluster, and this is seen in the average linkage dendrogram, we can see this.

Exercise 8.5.3

```
r <- seq(11)
x <- c((r - 1) / 10, 2)

ggplot() +
  geom_hline(yintercept = 0) +
  coord_fixed() +
  geom_point(aes(x = x, y = 0)) +
  labs(x = NULL, y = NULL)
```



If we put the 11 leftmost points in one cluster and the point at $x = 2$ in another:

```
sum(as.matrix(dist(x[seq(11)])))
```

```
[1] 44
```

On the other hand, if we put the two outermost points in one cluster and the 10 points between them in another cluster:

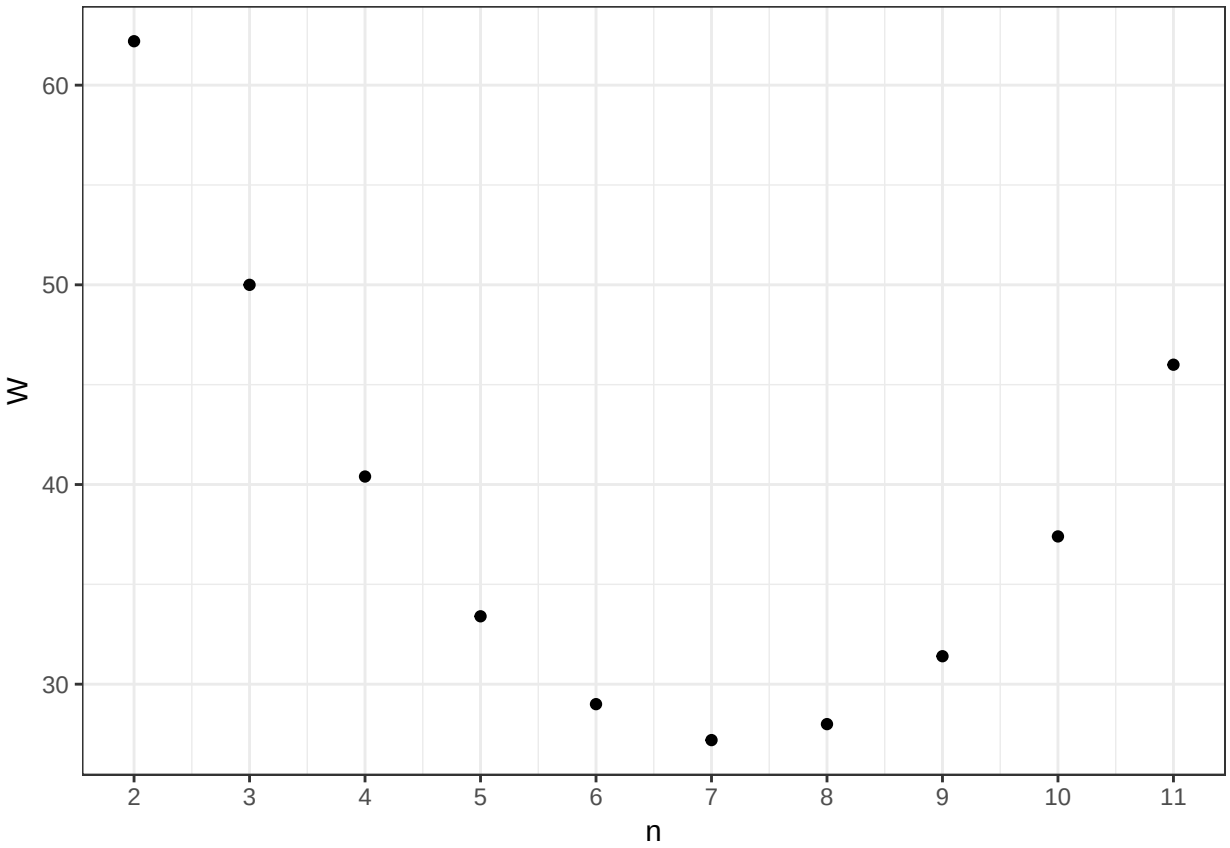
```
sum(as.matrix(dist(x[seq(2, 11)]))) + sum(as.matrix(dist(x[c(1, 12)])))
```

```
[1] 37
```

Optimal clustering with $K = 2$

We can see that the optimal clustering would divide the left n points into one cluster and the right $12 - n$ points into another cluster. So now we just have to find n.

```
W.vector <- sapply(seq(2, 11), function(n) {
  sum(as.matrix(dist(x[seq(n)]))) + sum(as.matrix(dist(x[seq(n, 12)])))
})

ggplot() +
  geom_point(aes(x = seq(2, 11), y = W.vector)) +
  labs(x = 'n', y = 'W') +
  scale_x_continuous(breaks = seq(12))
```



So W is minimized when $n = 7$, i.e., when the seven left-most points are in one cluster and the 5 right-most points are in the other cluster.