STATS/CSE 780
Assignment 3

Pao Zhu Vivian Hsu (Student Number: 400547994)

2023-11-09

Introduction

Methods

The data in this study was sourced from the UC Irvine Machine Learning Repository (Martiniano & Ferreira, 2018). It contains 21 variables and 740 records of absenteeism at work from 2007 to 2010 at a courier company in Brazil (Martiniano & Ferreira, 2018). The data set includes a categorical variable indicating the reason for absenteeism and various columns describing the employee and their working conditions. The categorical variable for absenteeism combined with quantitative variables makes this a suitable data set for hierarchical and k-means clustering. Since we are only interested in clustering using quantitative variables, all categorical variables apart from the reason for absenteeism were removed from the data set, resulting in a total of 12 variables.

Prior to clustering, we first explored the data to check if data transformation was required. There were no missing values, unexpected data types, or unreasonable data types detected in the data as outlined in Supp. Materials Table 2. Of the 12 variables, 8 of them had outliers through box plots as shown in Supp. Materials Figure 3. These were capped by the lower and upper fences computed with the interquartile range. We also scaled and centered the data.

Once data transformation was done, we then applied clustering methods to the data. We first performed hierarchical clustering to the data. Silhouette plots for cluster sizes 2 to 20 were created and the average silhouette widths were plotted as shown in the left plot in Figure 1. The maximum average silhouette width was for a cluster size of 20, however we decided to use 10 clusters instead because the average widths were not very different between a cluster size of 10 versus 20 as shown in Figure 1, and there were many negative silhouette widths (i.e. samples placed into the wrong cluster) for cluster size of 20.

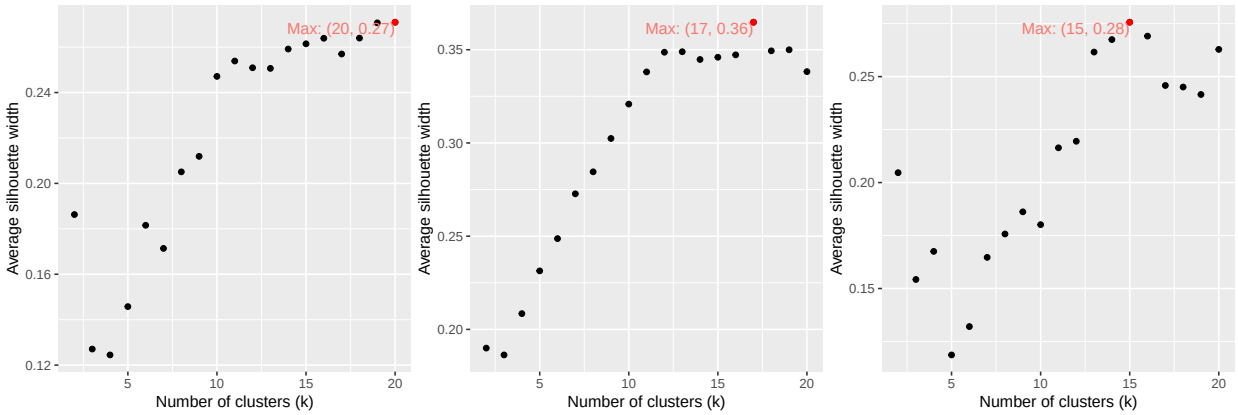

Figure 1: Average silhouette widths for cluster sizes 1 to 20; Left chart is for hierarchical clustering, centre chart is for k-means clustering, and right chart is for PCA followed by hierarchical clustering.

Next, we performed k-means clustering. Similar to the previous method, we created silhouette plots for cluster sizes 2 to 20 and looked for the maximum average silhouette width. In this case, a cluster size of 17 produced the highest average as displayed in the center plot of Figure 1. We decided to use 17 clusters since other cluster sizes did not show strong improvements in terms of negative silhouette widths as shown in Supp. Materials Figure 4.

Finally, we performed principal component analysis (PCA) followed by hierarchical clustering. 8 principal components (PCs) and 14 clusters were used. We decided the PCs using Figure 2 and selected the size where approximately 85% of the variation could be explained. We decided in a similar way as the previous two clustering methods. We plotted average silouette widths for sizes of 2 to 20. We then selected the cluster with the maximum silhouette score as illustrated in the right plot of Figure 1, and ensured that the negative silhouette width was minimal compared to other cluster sizes that produced a similar average silhouette width. This shown in Supp. Materials Figure 5.

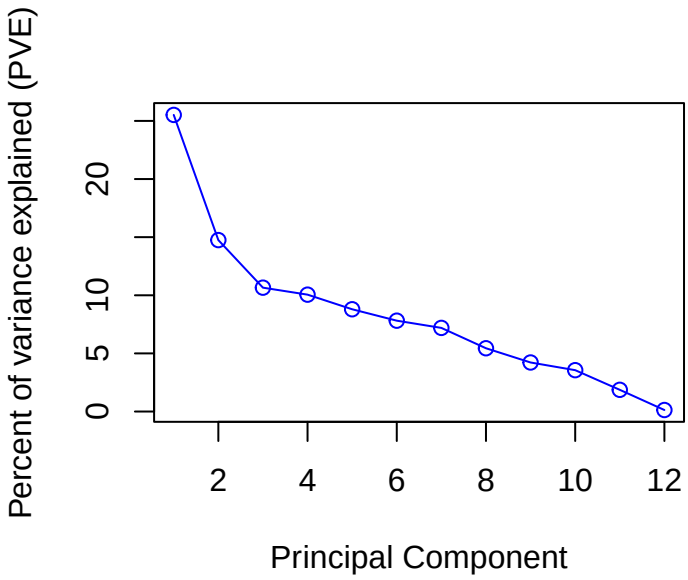

Figure 2: Scree plot

Results

Table 1 shows the results for the three clustering methods using Rand index and adjusted Rand index. All three methods produced a Rand index between 0.81 and 0.86. K-means clustering had the highest Rand index, followed closely by hierarchical clustering after PCA, and finally hierarchical clustering. While this suggests that k-means clustering produces the most similar data clustering compared to the given data labels, the adjusted Rand index is quite low compared to the other two methods. Hierarchical clustering produced the highest adjusted Rand index. All three methods produced a low adjusted Rand index value ranging from about 0.09 to 0.17.

Thus, out of the three methods, the hierarchical methods performed the best. While Rand indices were high, adjusted Rand indices were low. This indicates that these results may not be reliable and the clustering methods we used may not explain the absenteeism at work very well. Further work is required to determine if another method, such as PCA followed by k-means clustering, would better describe absenteeism.

Table 1: Comparison of clustering methods using Rand index and adjusted Rand index

clustering_method	rand_index	adj_rand_index
hierarchical	0.812	0.176
k-means	0.853	0.092

clustering_method	rand_index	adj_rand_index
pca followed by hierarchical	0.847	0.111

Supplementary material

Tables and figures

Table 2: Summary of data prior to cleansing

	data_type	min	max	nulls_blanks
Transportation.expense	integer	118.000	388.000	0
Distance.from.Residence.to.Work	integer	5.000	52.000	0
Service.time	integer	1.000	29.000	0
Age	integer	27.000	58.000	0
Work.load.Average.day	numeric	205.917	378.884	0
Hit.target	integer	81.000	100.000	0
Son	integer	0.000	4.000	0
Pet	integer	0.000	8.000	0
Weight	integer	56.000	108.000	0
Height	integer	163.000	196.000	0
Body.mass.index	integer	19.000	38.000	0
Absenteeism.time.in.hours	integer	0.000	120.000	0

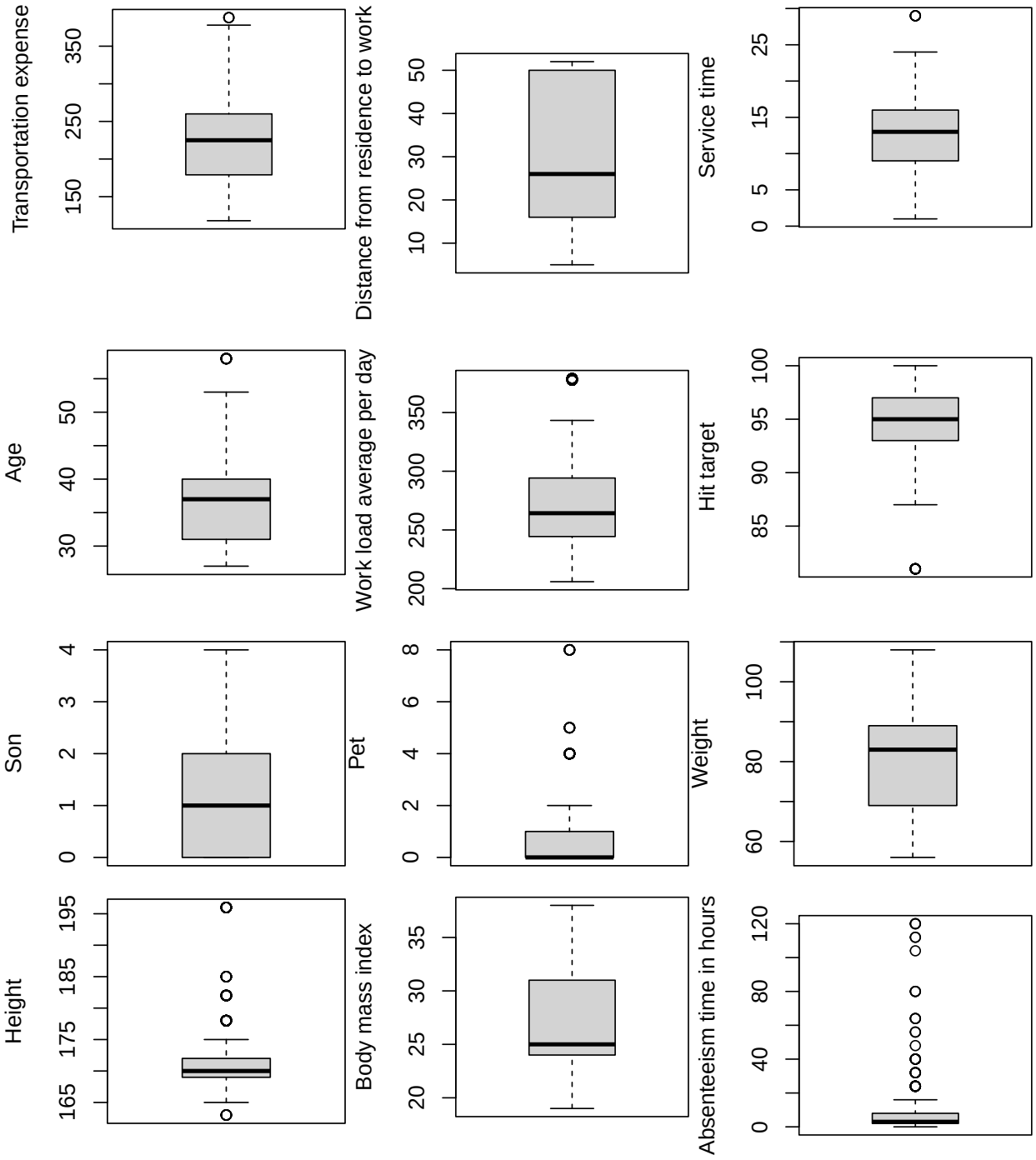

Figure 3: Outliers were detected from the box plots of some variables

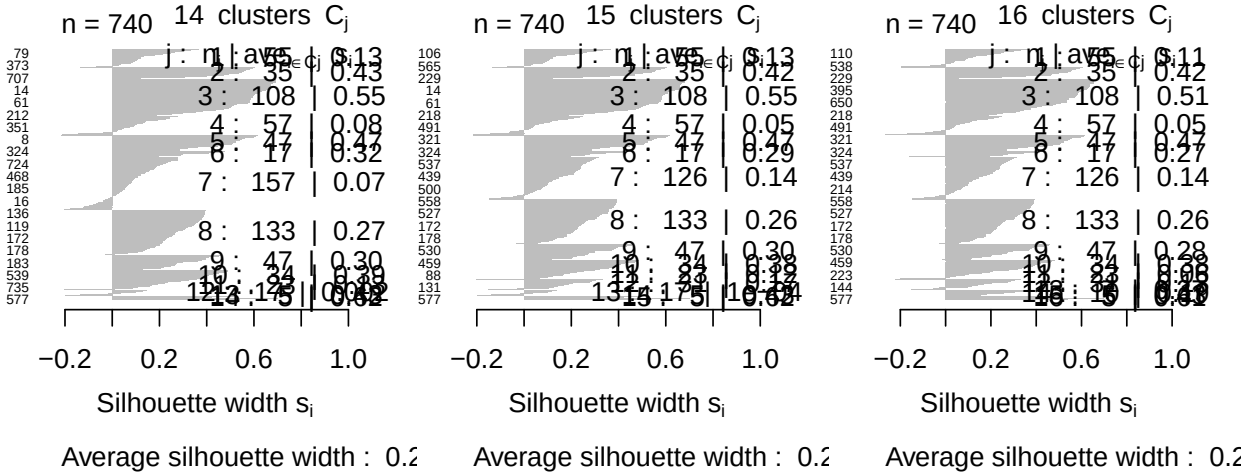

Figure 4: Silhouette plots for cluster sizes k = 16, 17, and 18. k = 17 returned the maximum silhouette width values while k = 16 and 18 are neighbouring cluster sizes with a similar silhouette widths.

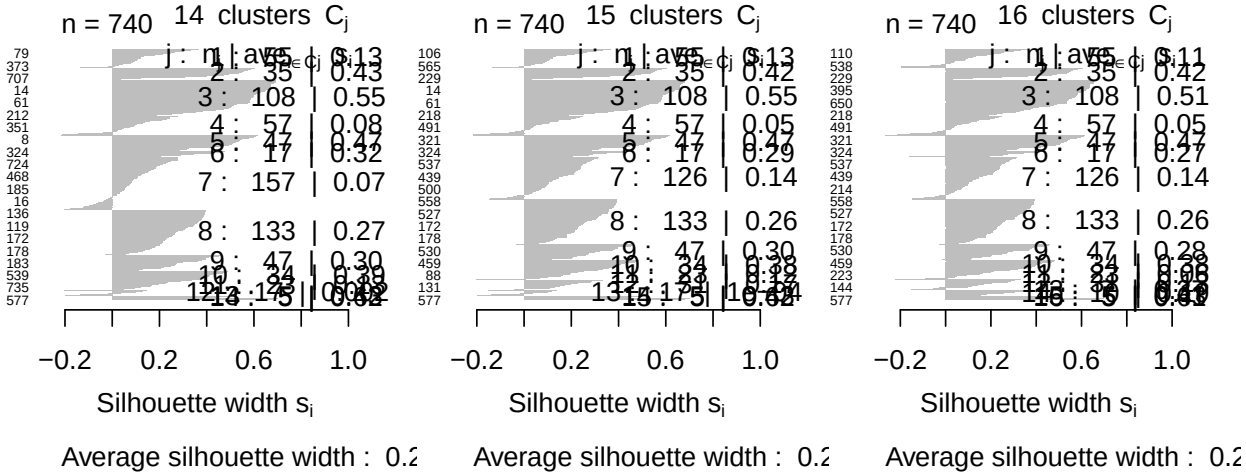

Figure 5: Silhouette plots for cluster sizes k = 14, 15, and 16. k = 15 returned the maximum silhouette width values while k = 14 and 16 are neighbouring cluster sizes with a similar silhouette widths.

Code

```
# ----- SETUP ----- #

packages <- c("knitr", "tidyverse", "ggplot2", "cluster", "fossil")

lapply(packages, library, character.only = TRUE)
```

```r
# Read data, extract labels, and keep only quantitative data
absentData_full <- read.csv("Absenteeism_at_work.csv", sep = ";")
absentData_lab <- absentData_full$`Reason.for.absence`
absentData_notclean <- absentData_full %>%
  select(-c("Reason.for.absence","ID","Month.of.absence","Day.of.the.week","Seasons",
            "Disciplinary.failure","Education","Social.drinker","Social.smoker"))

# ----- DATA EXPLORATION ----- #
# Check data types, min, max, and missing data
data_type <- sapply(absentData_notclean,class)
min <- sapply(absentData_notclean, function(col){min(col,na.rm=TRUE)})
max <- sapply(absentData_notclean, function(col){max(col,na.rm=TRUE)})
nulls <- sapply(absentData_notclean, function(col){sum(is.na(col))})
blanks <- sapply(absentData_notclean,
                 function(col){ifelse(is.na(sum(col == "")), 0, sum(col == ""))})
data_summary <- data.frame(row.names = names(nulls), data_type=data_type,
                           min=min, max=max, nulls_blanks=nulls+blanks)
kable(data_summary)

# Create box plots to check for outliers
b01 <- boxplot(absentData_notclean$Transportation.expense, ylab = "Transportation expense")
b02 <- boxplot(absentData_notclean$Distance.from.Residence.to.Work,
               ylab = "Distance from residence to work")
b03 <- boxplot(absentData_notclean$Service.time, ylab = "Service time")
b04 <- boxplot(absentData_notclean$Age, ylab = "Age")
b05 <- boxplot(absentData_notclean$Work.load.Average.day, ylab = "Work load average per day"
b06 <- boxplot(absentData_notclean$Hit.target, ylab = "Hit target")
b07 <- boxplot(absentData_notclean$Son, ylab = "Son")
b08 <- boxplot(absentData_notclean$Pet, ylab = "Pet")
b09 <- boxplot(absentData_notclean$Weight, ylab = "Weight")
b10 <- boxplot(absentData_notclean$Height, ylab = "Height")
```

```r
b11 <- boxplot(absentData_notclean$Body.mass.index, ylab = "Body mass index")
b12 <- boxplot(absentData_notclean$Absenteeism.time.in.hours, ylab = "Absenteeism time in ho

# ----- DATA CLEANSING -----
# Handle outliers by capping them using interquartile range
cap <- function(val, bplot) {
  lower_fence <- bplot$stats[2]-(1.5*(bplot$stats[4]-bplot$stats[2])) #Q1-1.5*IQR
  upper_fence <- bplot$stats[4]+(1.5*(bplot$stats[4]-bplot$stats[2])) #Q3+1.5*IQR
  val <- ifelse(val < lower_fence, lower_fence, val)
  val <- ifelse(val > upper_fence, upper_fence, val)
  val
}
absentData <- absentData_notclean %>%
  mutate(Transportation.expense = cap(val=Transportation.expense, bplot=b01),
         Service.time = cap(val=Service.time, bplot=b03),
         Age = cap(val=Age, bplot=b04),
         Work.load.Average.day = cap(val=Work.load.Average.day, bplot=b05),
         Hit.target = cap(val=Hit.target, bplot=b06),
         Pet = cap(val=Pet, bplot=b08),
         Height = cap(val=Height, bplot=b10),
         Absenteeism.time.in.hours = cap(val=Absenteeism.time.in.hours, bplot=b12))

# Scale the data
absentData_sd <- scale(absentData)

# ----- CLUSTERING FUNCTIONS ----- #
# Get silhouette for k-means clustering
kmcSilK <- function(k, data){
  x_k <- kmeans(data, k, nstart = 20)
  silhouette(x_k$cluster, dist(data))
}
```

```r
# Get silhouette for hierarchical clustering
hcSilK <- function(k, data, method = "complete"){
  hc_out <- hclust(dist(data), method = method)
  hc_clusters <- cutree(hc_out, k)
  silhouette(hc_clusters, dist(data))
}

# Plot silhouette
plotSil <- function(sil){
  plot(sil, nmax= 800, cex.names=0.5, main = "", border=NA)
}

# Choose k using goodness-of-clustering
# k = the k values to test
# silFun = the silhouette function
# data = the data used in silFun
chooseK <- function(k, silFun, data) {

  # Get silhouettes and their widths
  sil_k <- lapply(k, silFun, data=data)
  sil_score <- sapply(sil_k, function(x) {mean(x[,"sil_width"])})

  # Find the k with the max width
  sil_max <- max(sil_score)
  sil_max_k <- match(sil_max, sil_score)+min(k)-1

  # Plot the silhouette widths and label the maximum
  silData <- tibble(k, sil_score)
  max_point <-tibble(k=sil_max_k,sil_score=sil_max)
  max_lab <- paste0("Max: (",sil_max_k,", ",round(sil_max,2), ")")
```

```
  plot <- ggplot(silData, aes(x=k, y=sil_score)) + geom_point() +
    labs(x="Number of clusters (k)", y="Average silhouette width") +
    geom_point(data=max_point, colour="red") +
    geom_text(data=max_point, aes(label=ifelse(k==sil_max_k,max_lab,""), color="red"),hjust=
    theme(legend.position="none")

  # Return plot, silhouettes, and max k
  list(plot=plot, sil_k=sil_k, max_k=sil_max_k)
}

# ----- AGGLOMERATIVE HIERARCHICAL CLUSTERING ----- #
# Get silhouette scores for multiple k values and plot them
set.seed(3)
k <- c(2:20)
good_of_cluster <- chooseK(k, hcSilK, absentData_sd) # uses complete linkage
good_of_cluster$plot
plotSil(good_of_cluster$sil_k[[9]])
plotSil(good_of_cluster$sil_k[[10]])
plotSil(good_of_cluster$sil_k[[11]])
plotSil(good_of_cluster$sil_k[[12]])
plotSil(good_of_cluster$sil_k[[14]])
plotSil(good_of_cluster$sil_k[[19]])
k <- 10

# Perform hierarchical clustering using best k
set.seed(3)
hc_out <- hclust(dist(absentData_sd))
ri_hc <- rand.index(cutree(hc_out, k), as.numeric(as.factor(absentData_lab)))
ari_hc <- adj.rand.index(cutree(hc_out, k), as.numeric(as.factor(absentData_lab)))

# ----- K-MEANS CLUSTERING ----- #
```

```r
# Get silhouette scores for multiple k values and plot them
set.seed(3)
k <- c(2:20)
good_of_cluster <- chooseK(k, kmcSilK, absentData_sd)
good_of_cluster$plot

# Based off silhouette plots, choose the best k
plotSil(good_of_cluster$sil_k[[15]])
plotSil(good_of_cluster$sil_k[[16]])
plotSil(good_of_cluster$sil_k[[17]])
k <- good_of_cluster$max_k
# Perform k-means clustering with best k and compute the rand indices
set.seed(3)
km_out <- kmeans(absentData, k, nstart = 20)
km_clusters <- km_out$cluster
ri_kmc <- rand.index(km_clusters, as.numeric(as.factor(absentData_lab)))
ari_kmc <- adj.rand.index(km_clusters, as.numeric(as.factor(absentData_lab)))

# ----- HIERARCHICAL CLUSTERING AFTER PCA ----- #
# Proportion of variance explained
set.seed(3)
pr_out <- prcomp(absentData, scale = TRUE)

# Scree plot
pve <- 100 * pr_out$sdev^2 / sum(pr_out$sdev^2)
plot(pve, type = "o",
xlab = "Principal Component", col = "blue", ylab = "Percent of variance explained (PVE)")

# Get silhouette scores for multiple k values and plot them
set.seed(3)
k <- c(2:20)
```

```r
good_of_cluster <- chooseK(k, hcSilK, pr_out$x[, 1:8])
good_of_cluster$plot

# Based off silhouette plots, choose the best k
plotSil(good_of_cluster$sil_k[[13]])
plotSil(good_of_cluster$sil_k[[14]])
plotSil(good_of_cluster$sil_k[[15]])
k <- good_of_cluster$max_k
set.seed(3)
hc_out <- hclust(dist(dist(pr_out$x[, 1:8])))
hc_clusters <- cutree(hc_out, k)
ri_pcahc <- rand.index(hc_clusters, as.numeric(as.factor(absentData_lab)))
ari_pcahc <- adj.rand.index(hc_clusters, as.numeric(as.factor(absentData_lab)))
# ----- COMPARISON ----- #
clustering_method <- c("hierarchical", "k-means", "pca followed by hierarchical")
rand_index <- round(c(ri_hc, ri_kmc, ri_pcahc),3)
adj_rand_index <- round(c(ari_hc, ari_kmc, ari_pcahc),3)
comparison <- data.frame(clustering_method = clustering_method,
                         rand_index=rand_index, adj_rand_index=adj_rand_index)
kable(comparison)
```

References

Martiniano, A., & Ferreira, R. (2018). *Absenteeism at work*. UC Irvine Machine Learning Repository. https://doi.org/10.24432/C5X882

R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/