

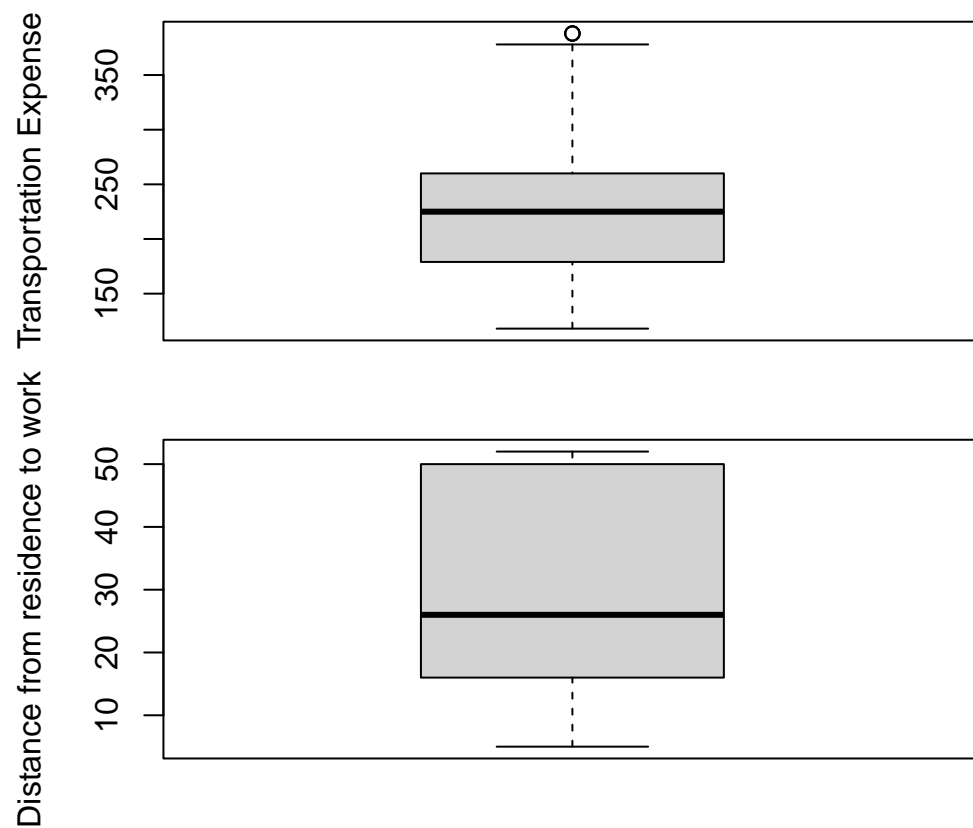
# **STATS/CSE 780**

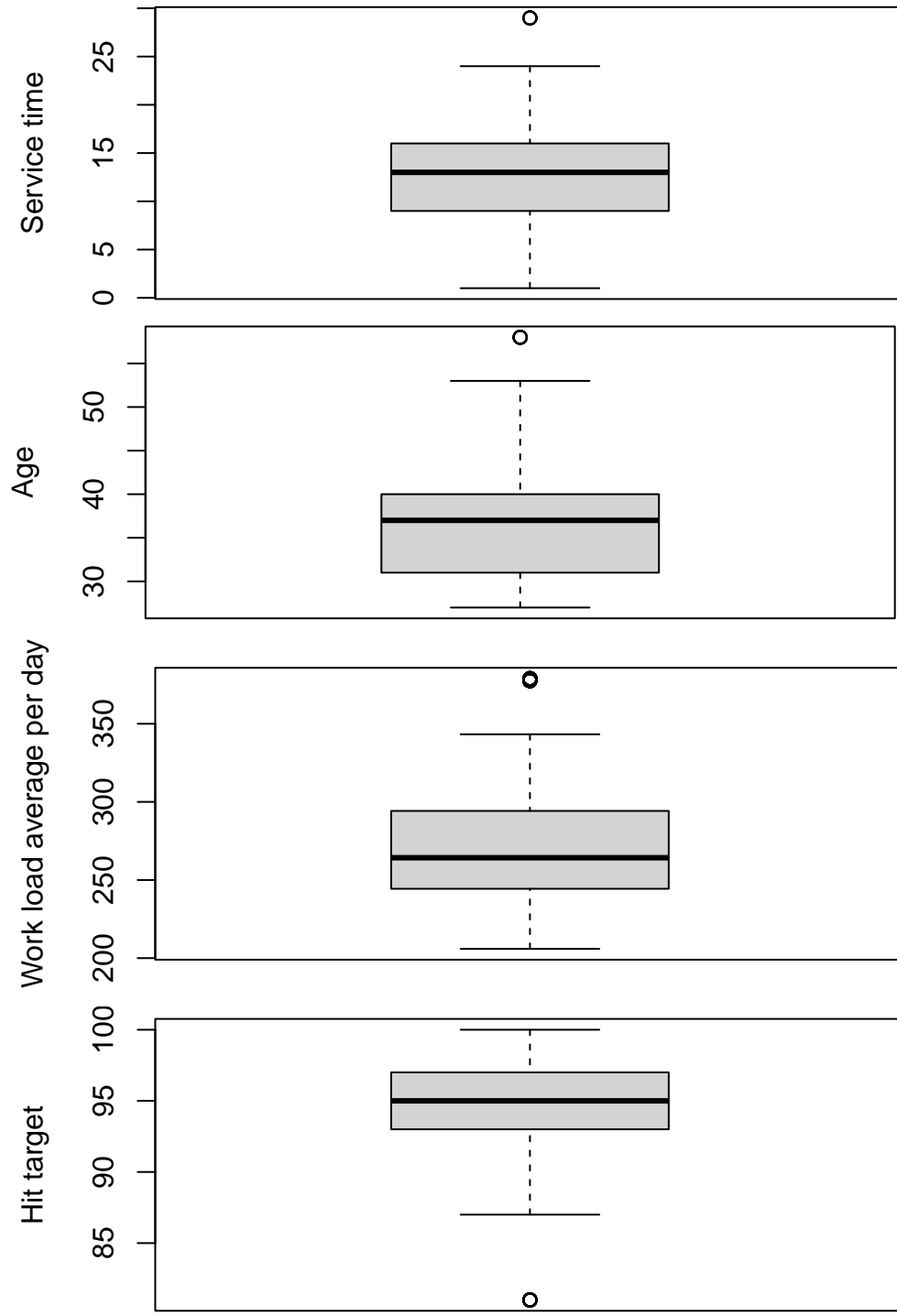
## **Assignment 3**

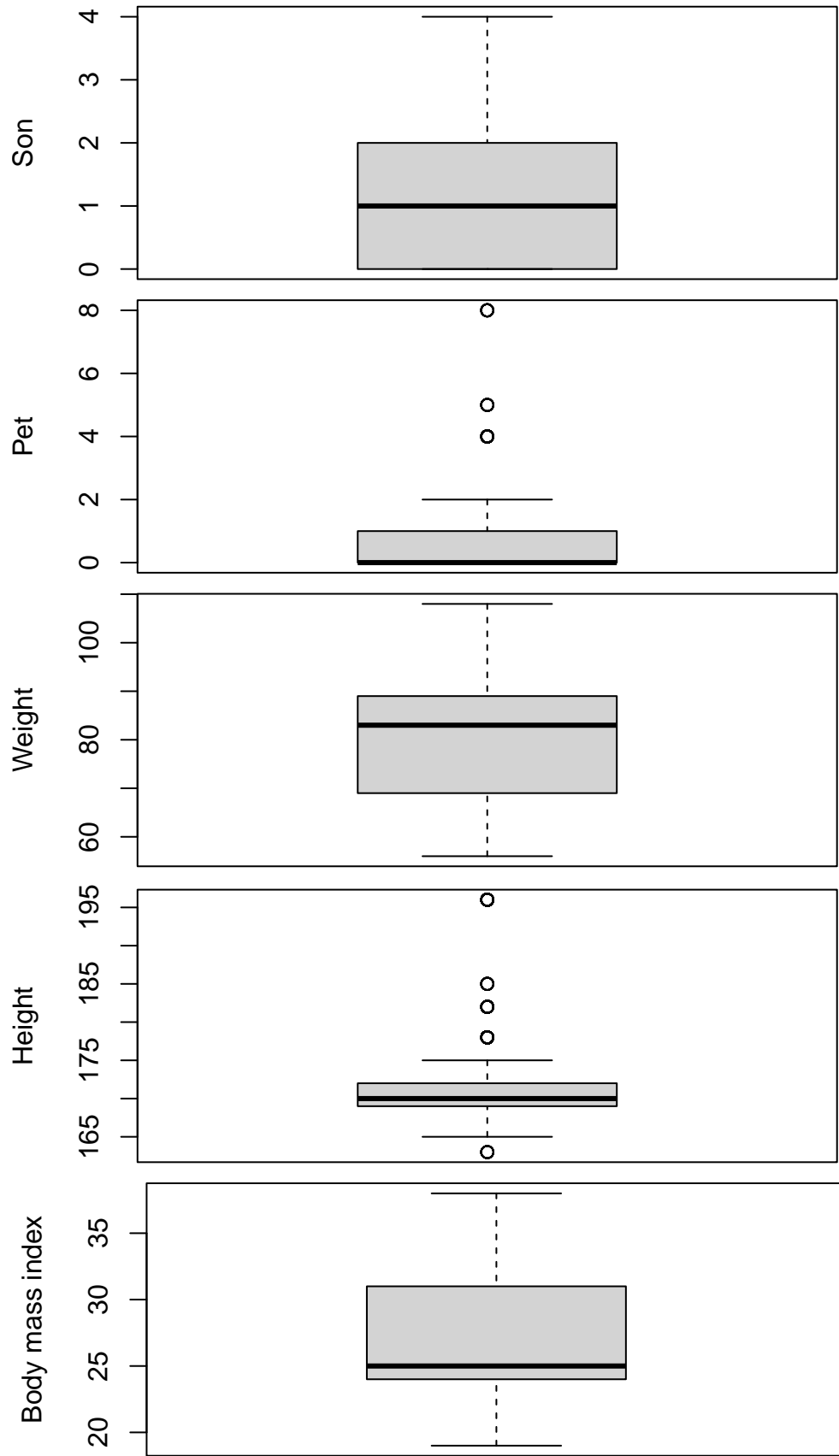
Pao Zhu Vivian Hsu (Student Number: 400547994)

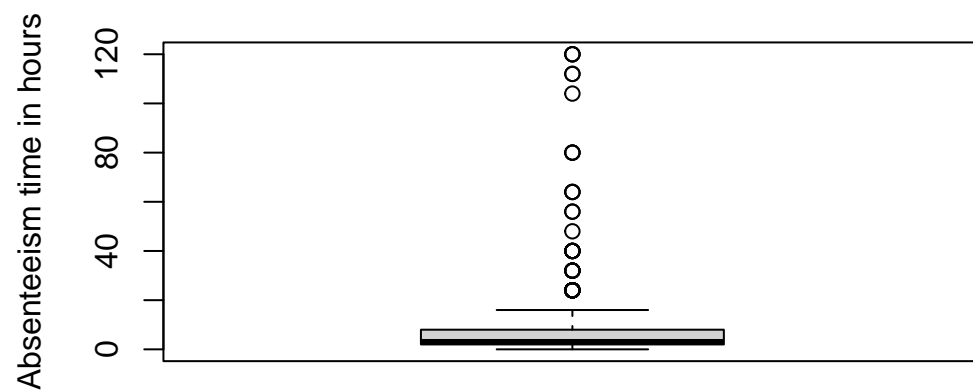
2023-11-07

	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

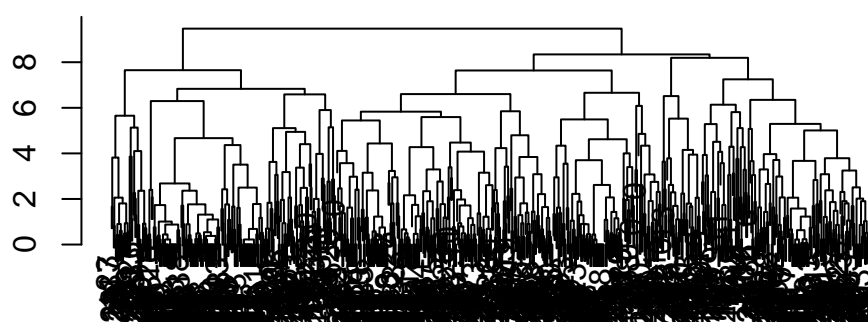




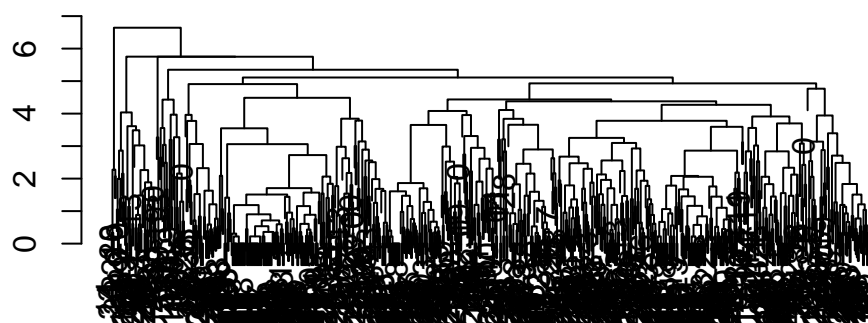




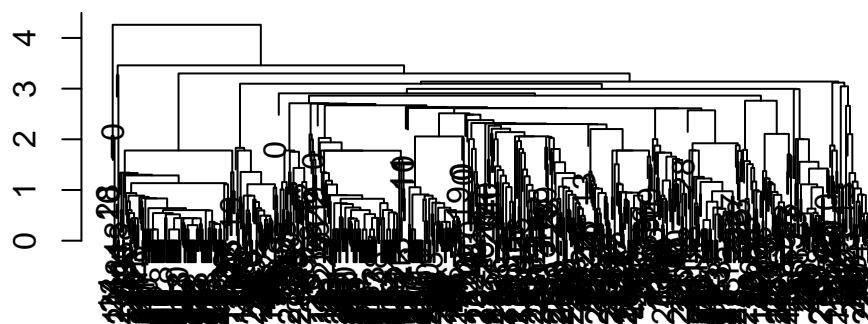
**Complete Linkage**

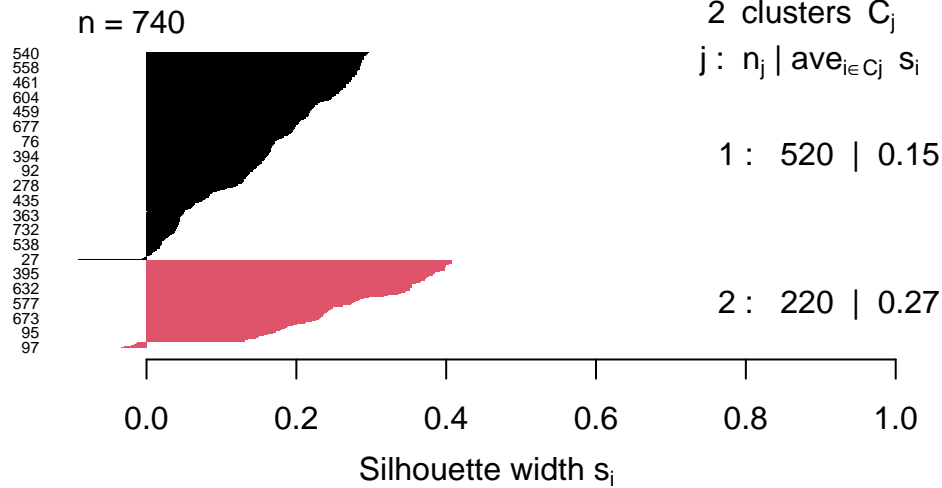


**Average Linkage**

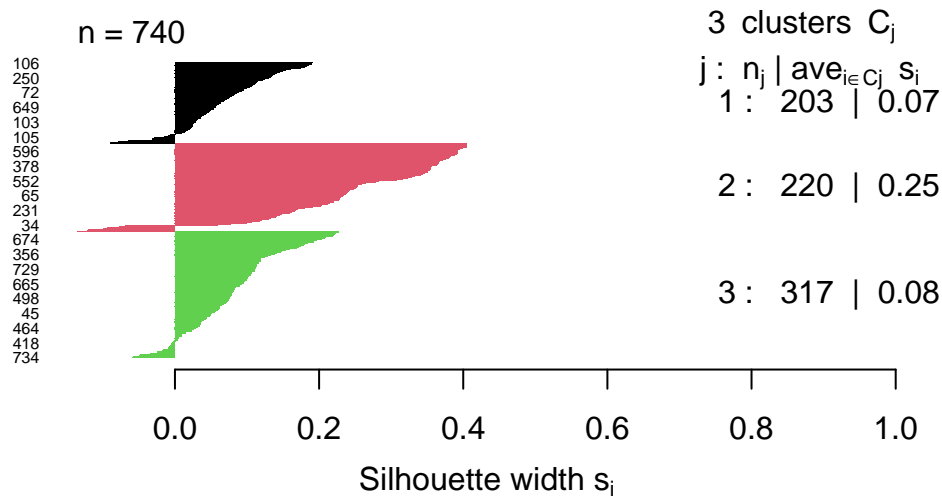


**Single Linkage**

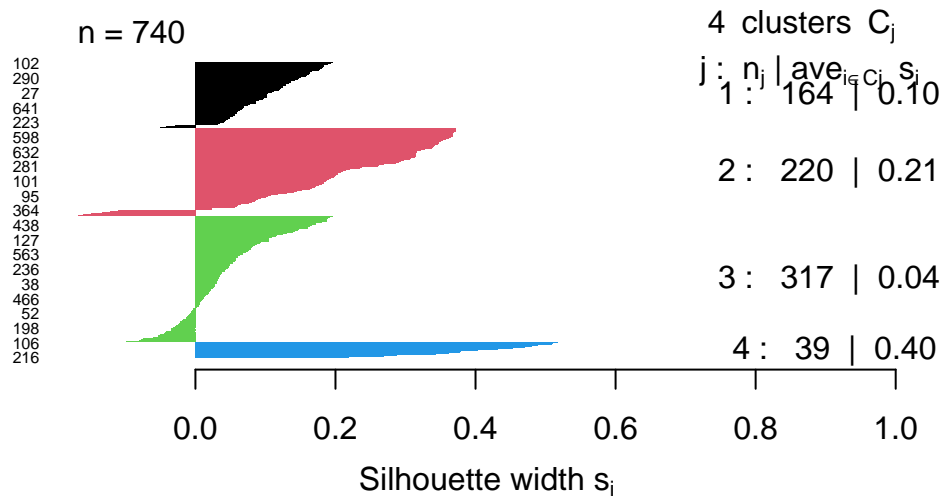




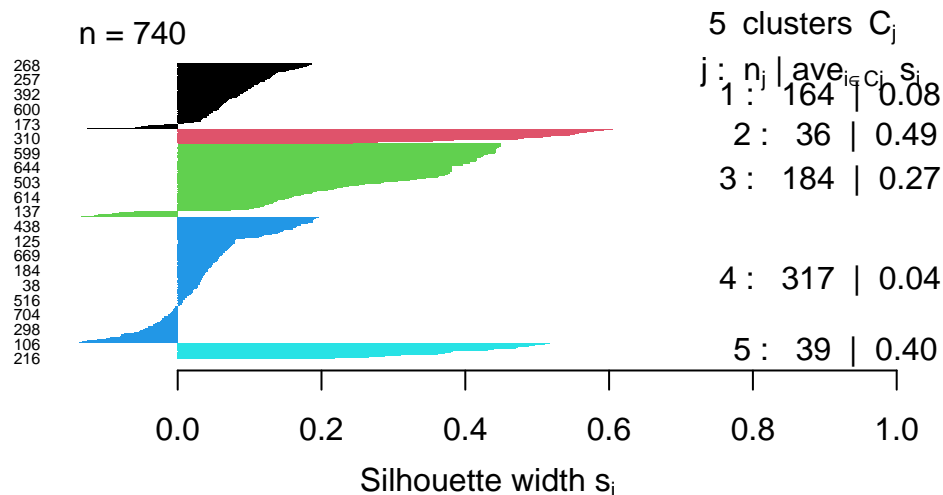
Average silhouette width : 0.19



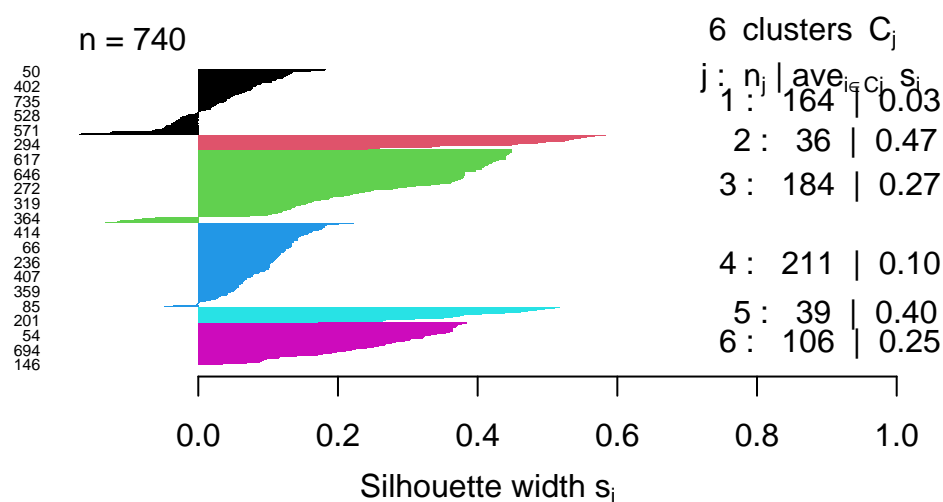
Average silhouette width : 0.13



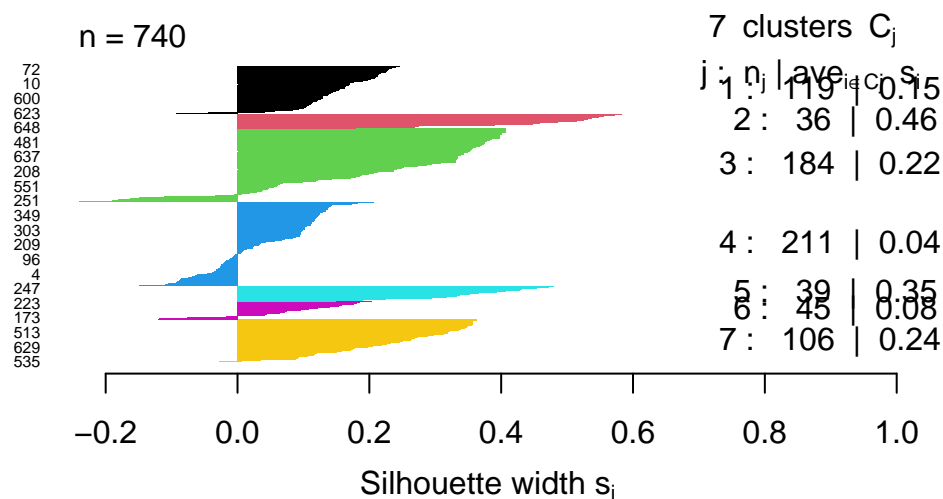
Average silhouette width : 0.12



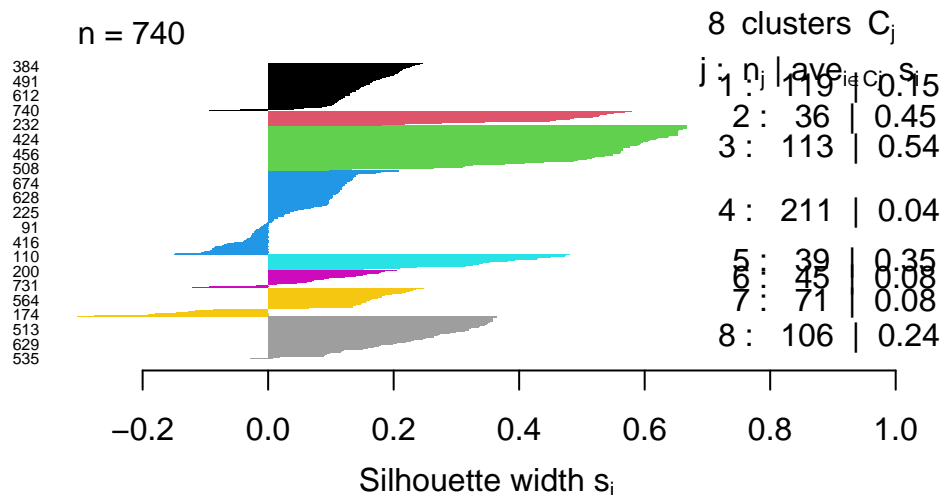
Average silhouette width : 0.15



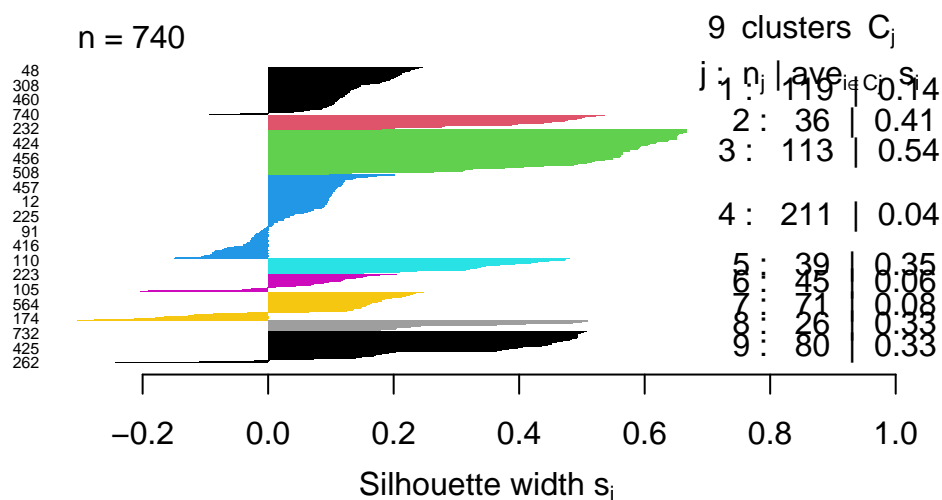
Average silhouette width : 0.18



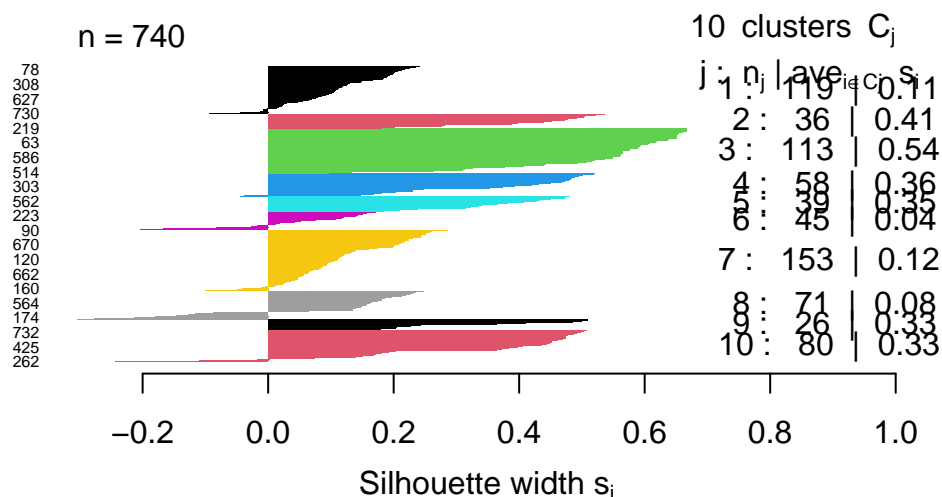
Average silhouette width : 0.17



Average silhouette width : 0.21



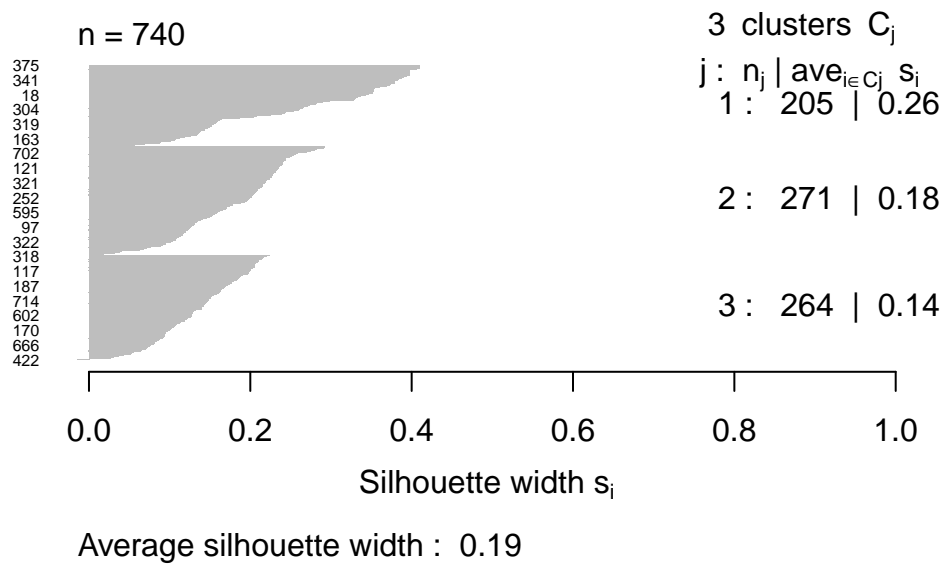
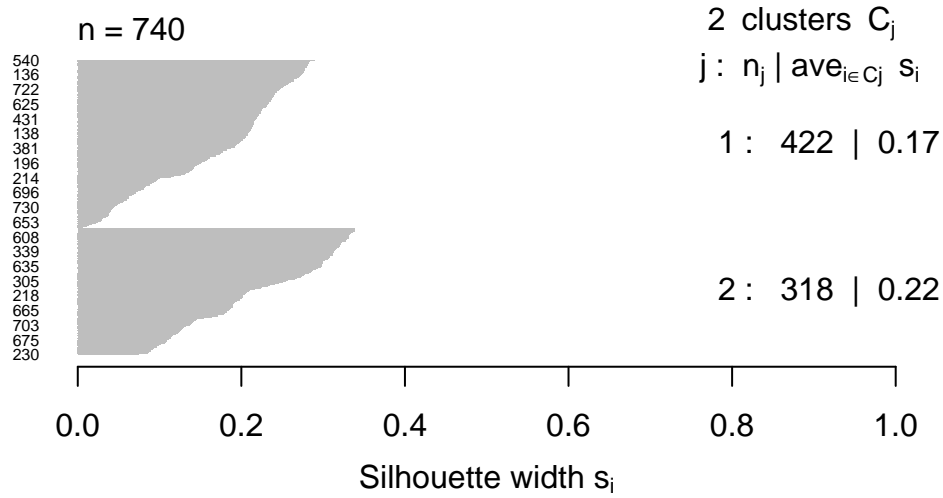
Average silhouette width : 0.21

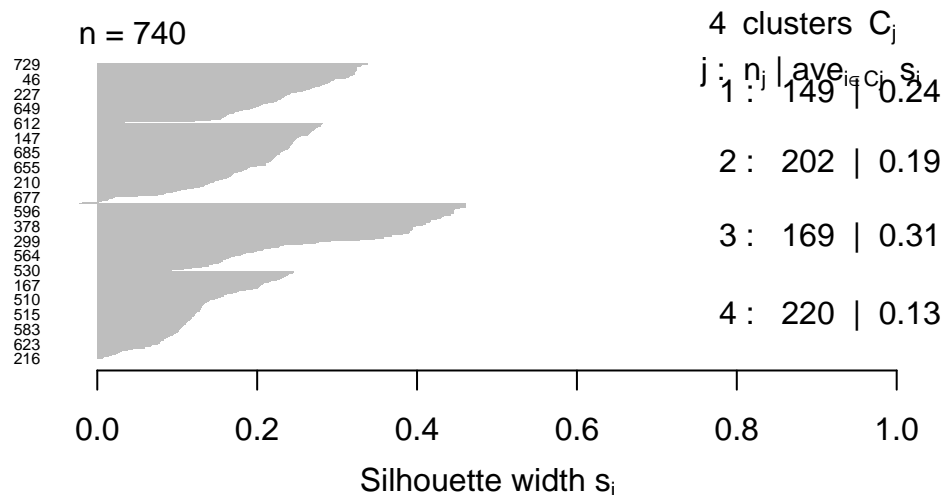


Average silhouette width : 0.25

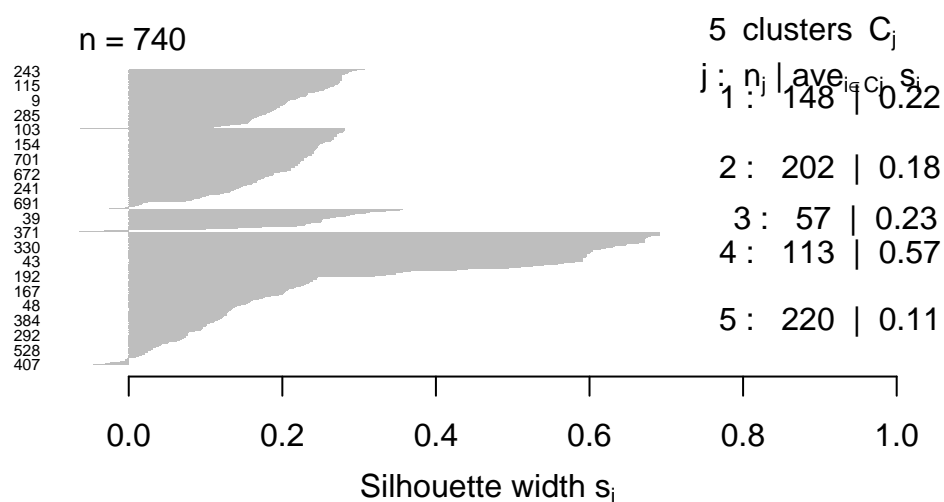


[1] 0.556248

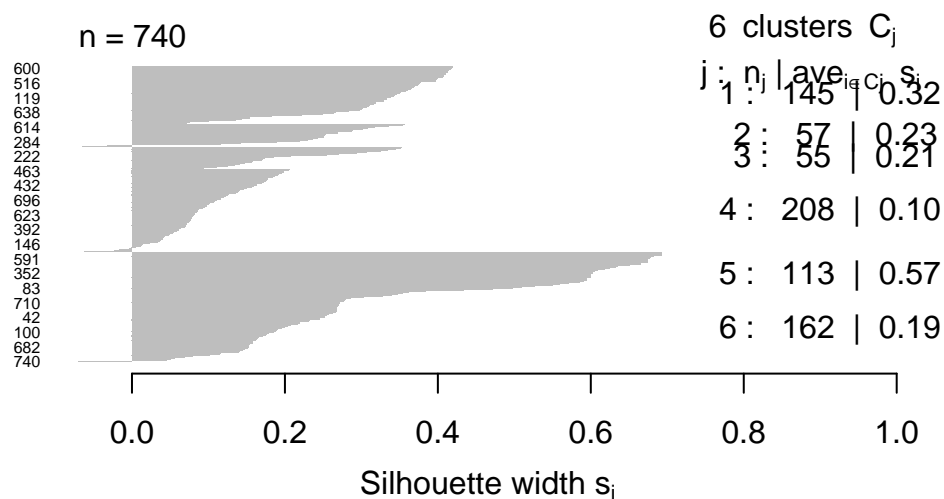




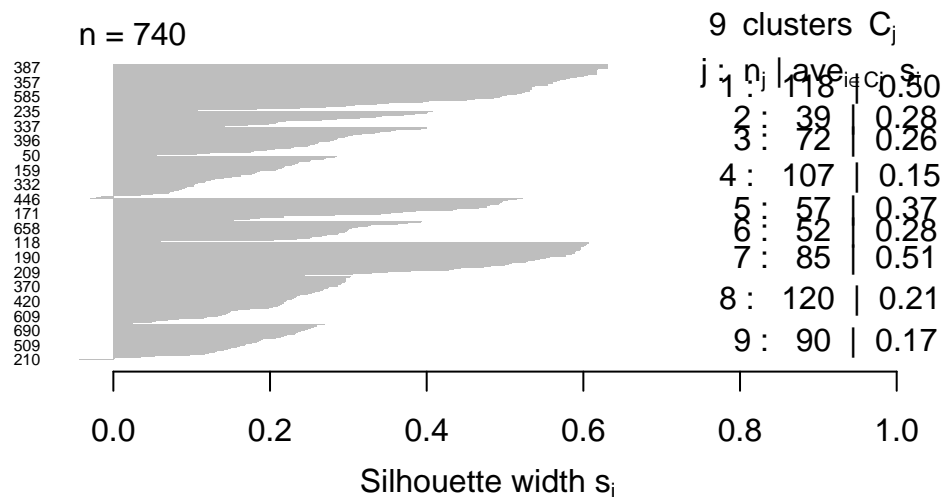
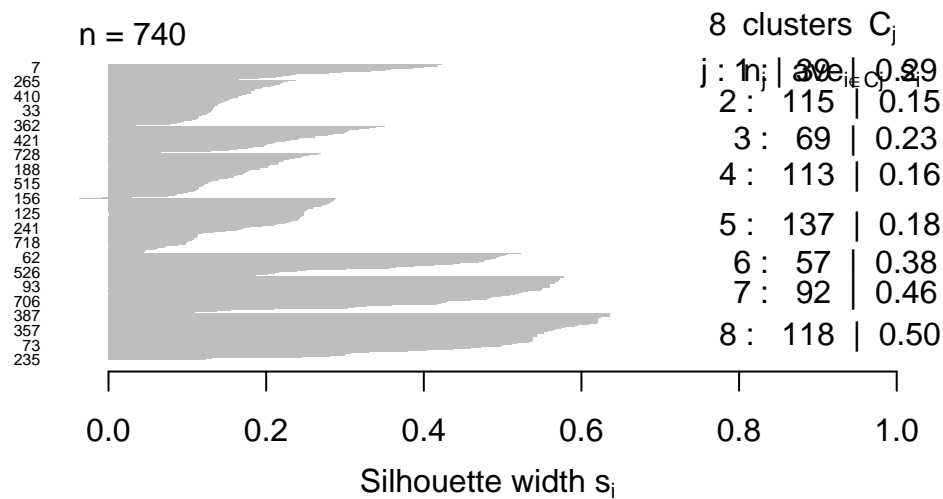
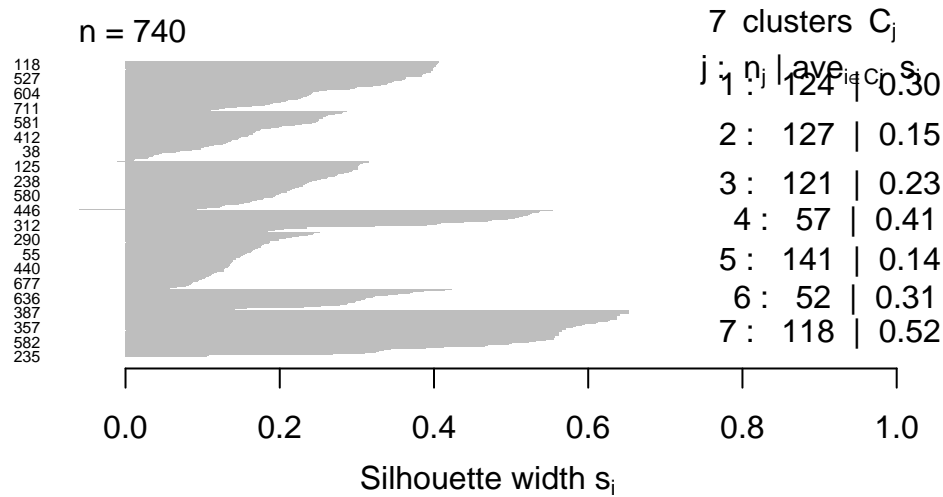
Average silhouette width : 0.21

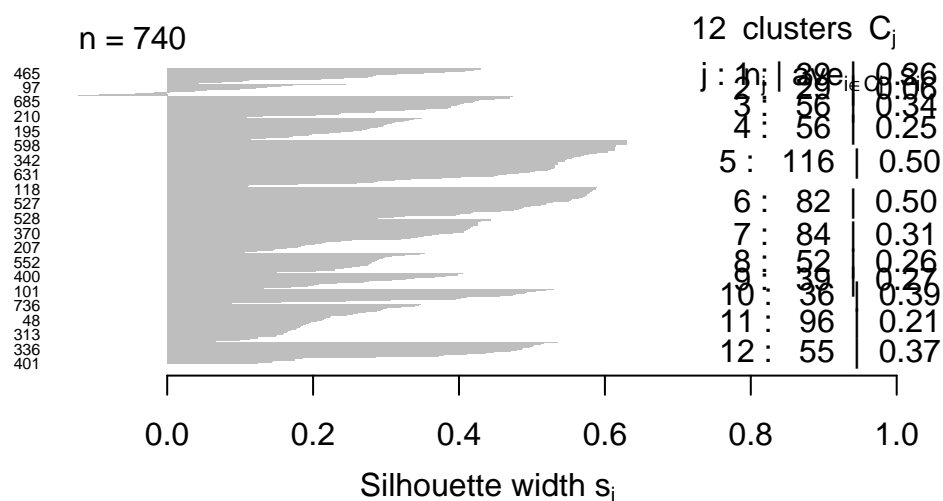
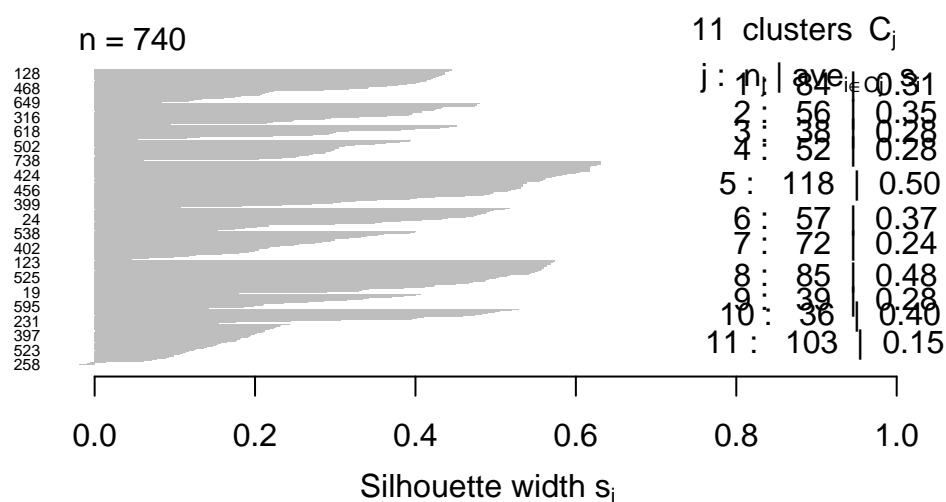
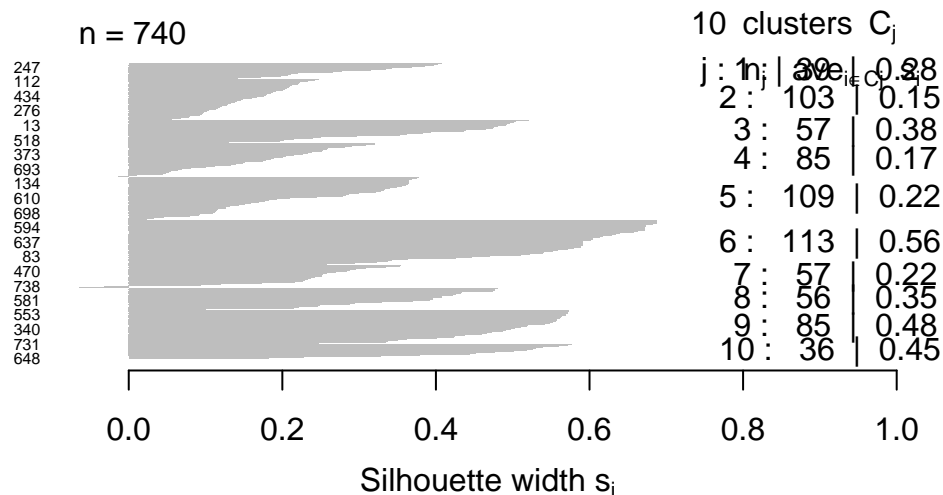


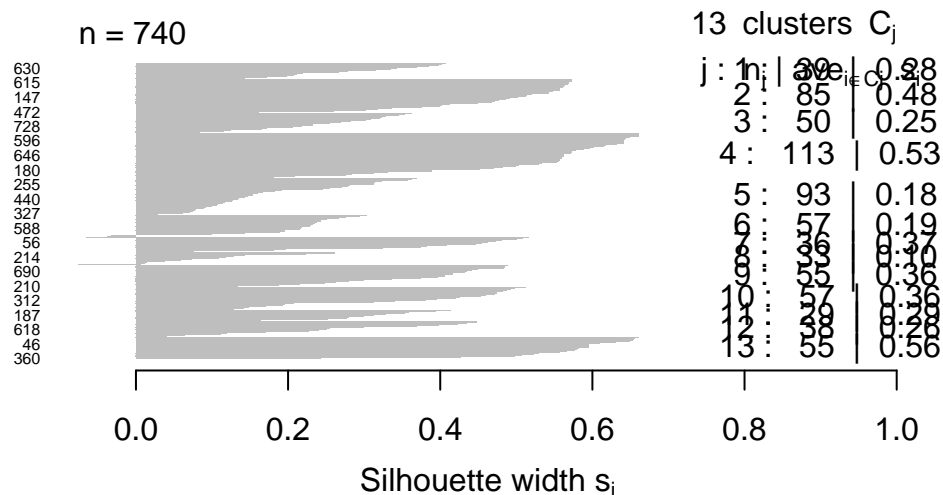
Average silhouette width : 0.23



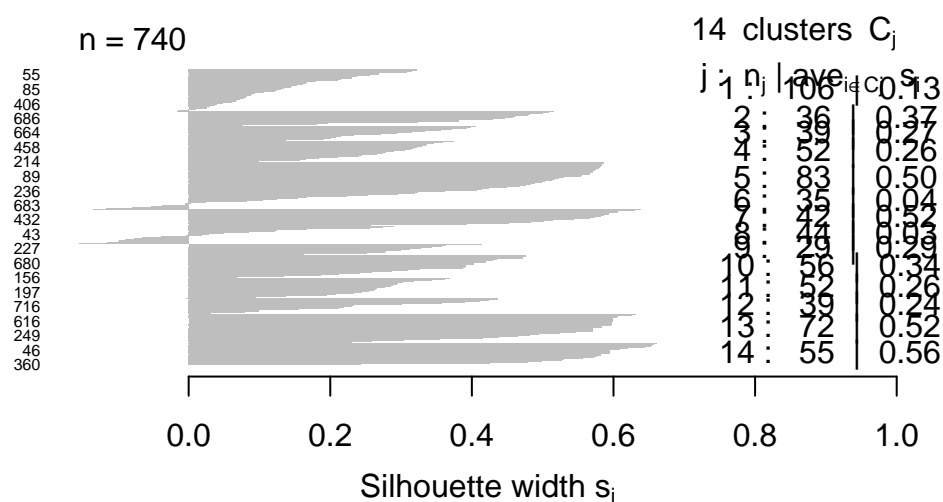
Average silhouette width : 0.25



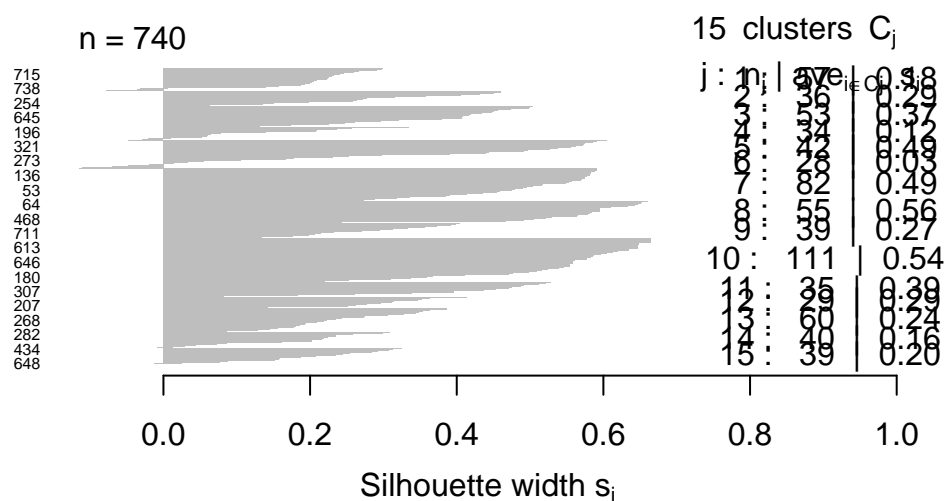




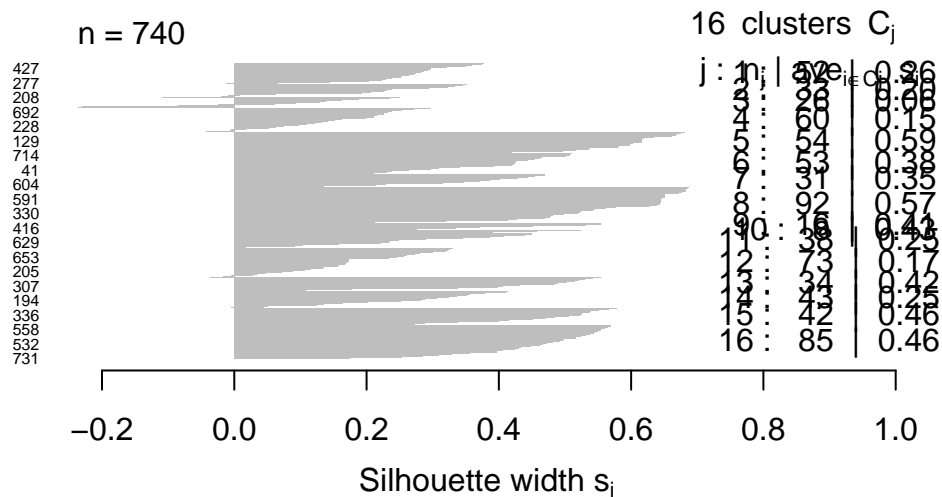
Average silhouette width : 0.35



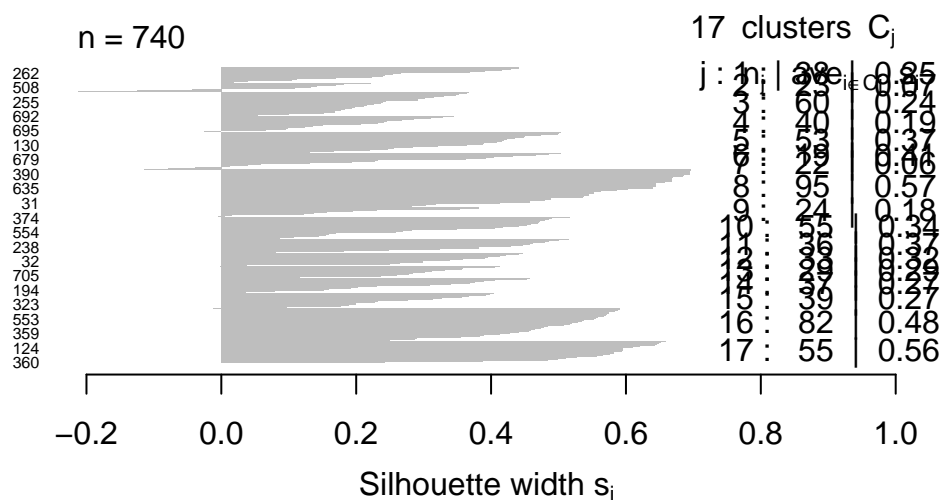
Average silhouette width : 0.32



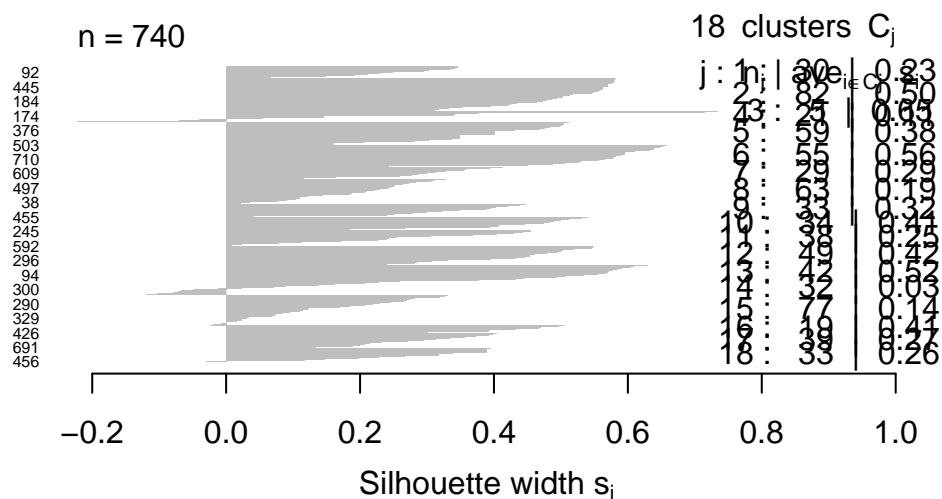
Average silhouette width : 0.35



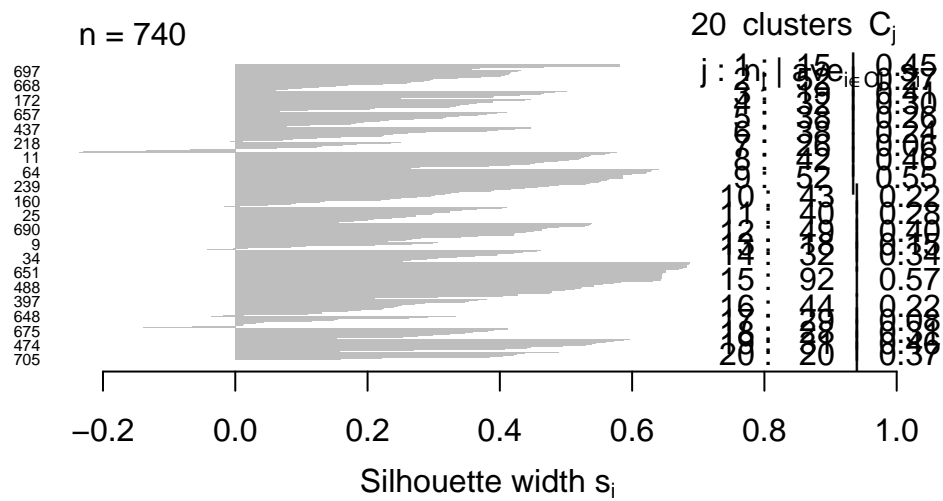
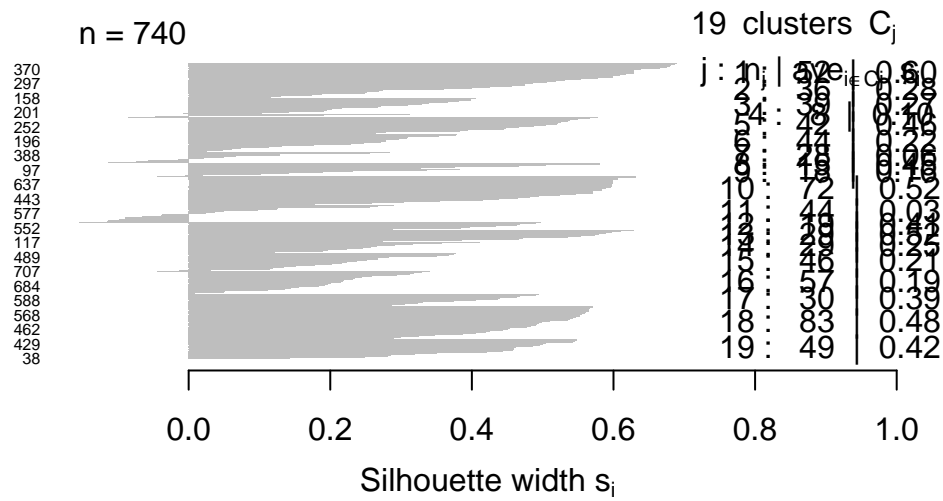
Average silhouette width : 0.35



Average silhouette width : 0.35

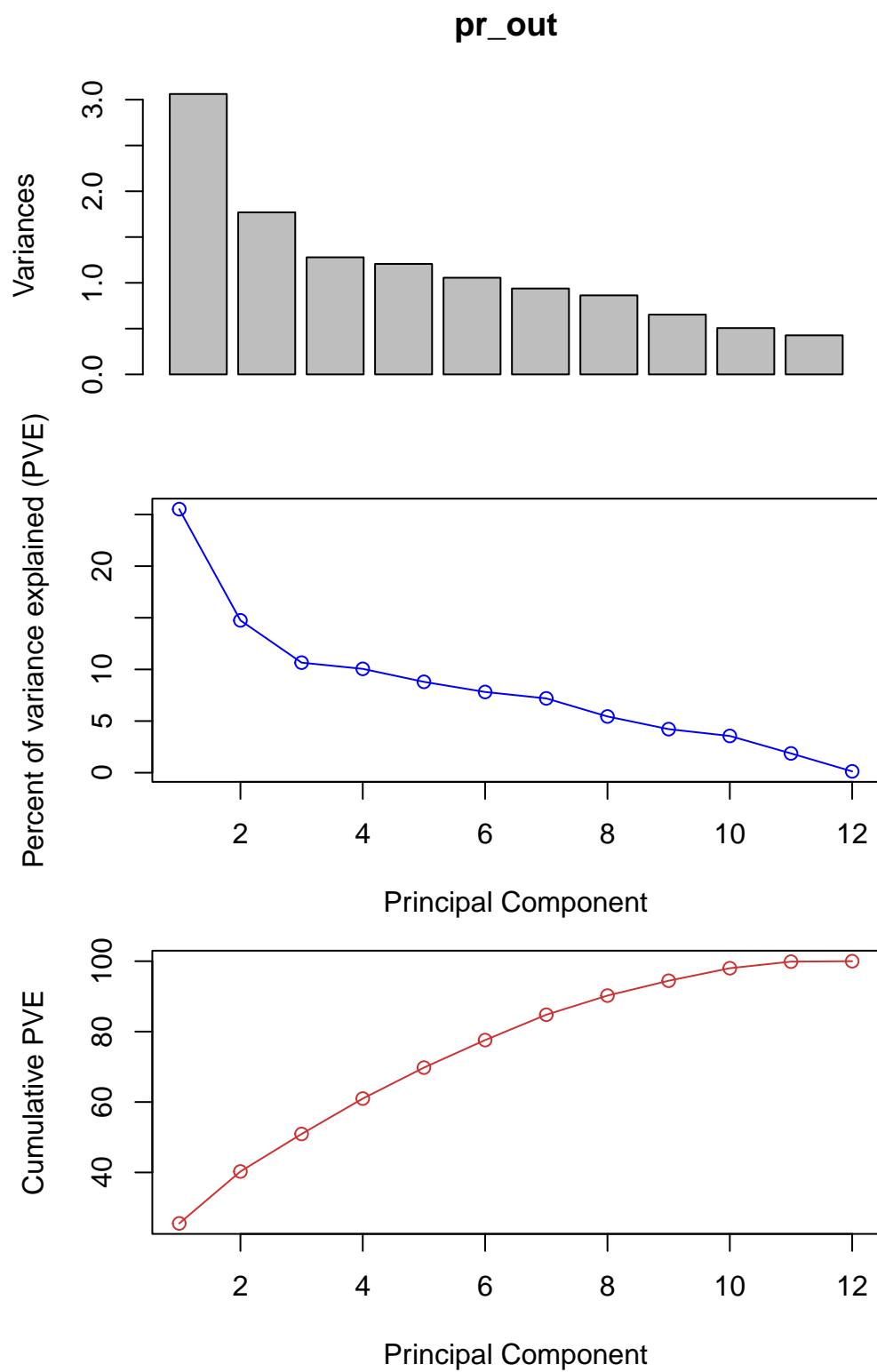


Average silhouette width : 0.33

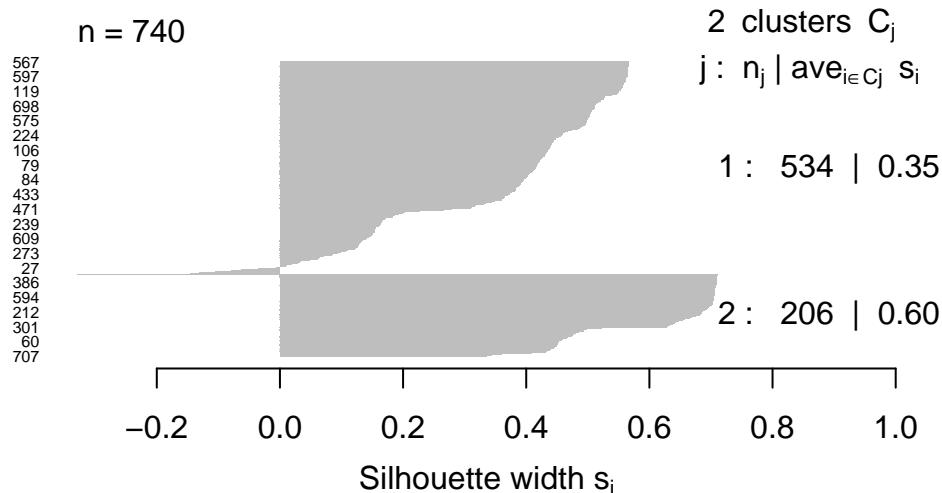


[1] 0.8415865

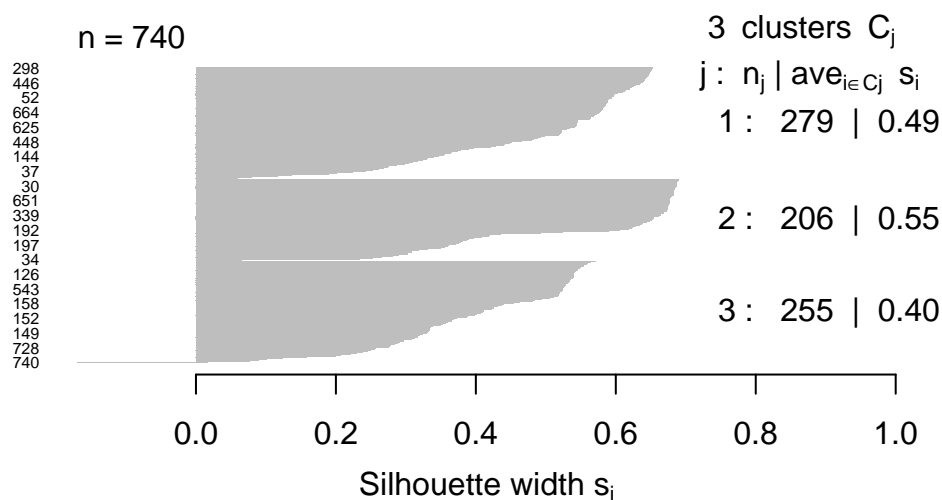
[1] 0.09915955



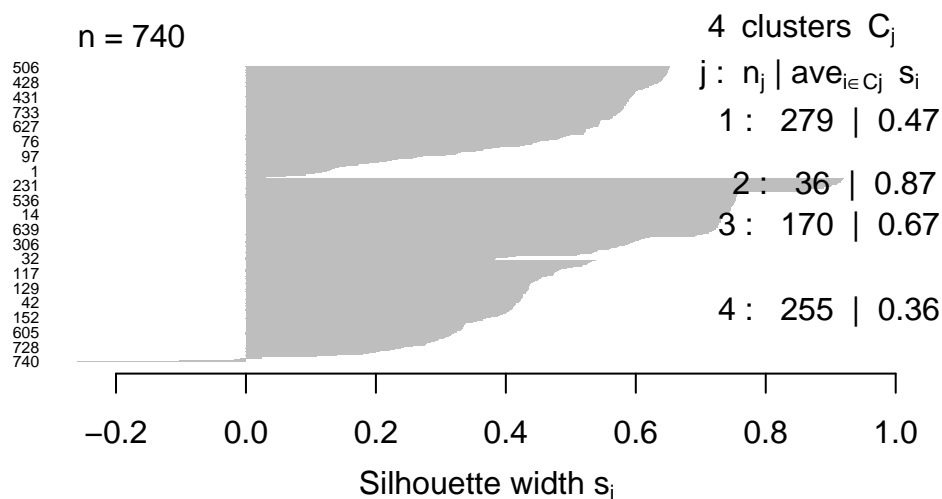




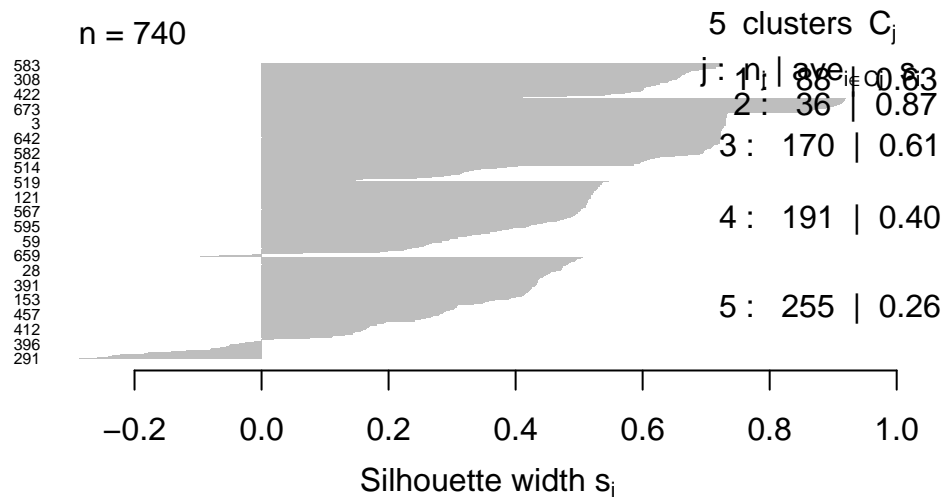
Average silhouette width : 0.42



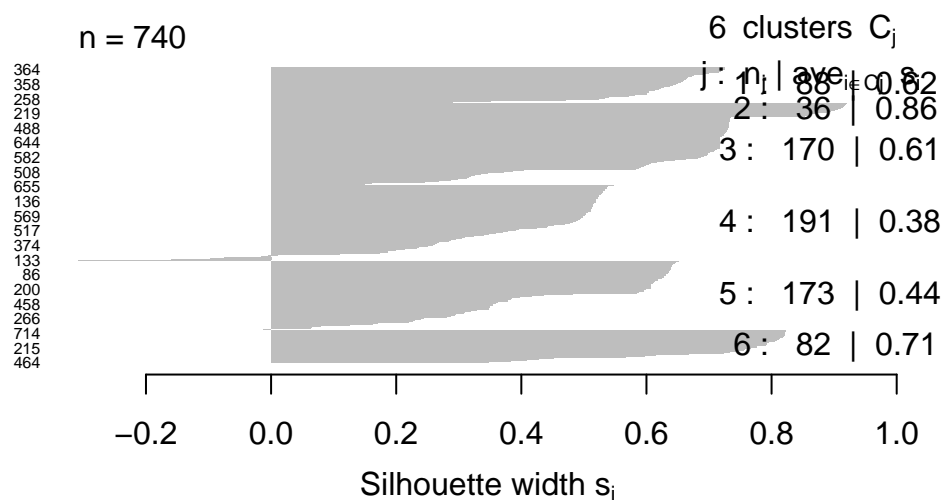
Average silhouette width : 0.48



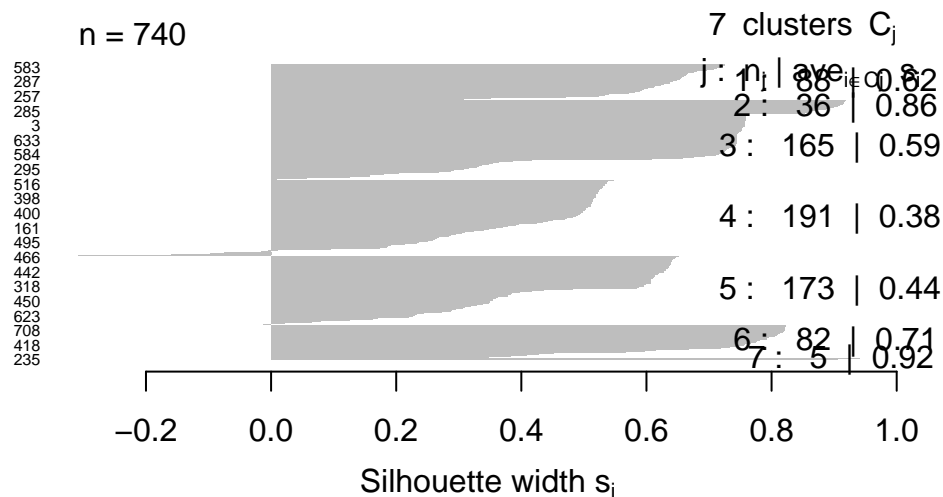
Average silhouette width : 0.5



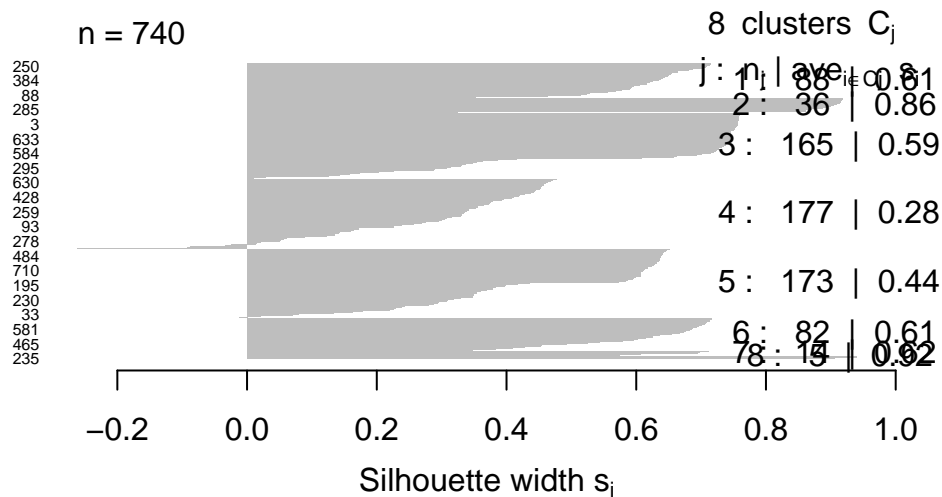
Average silhouette width : 0.45



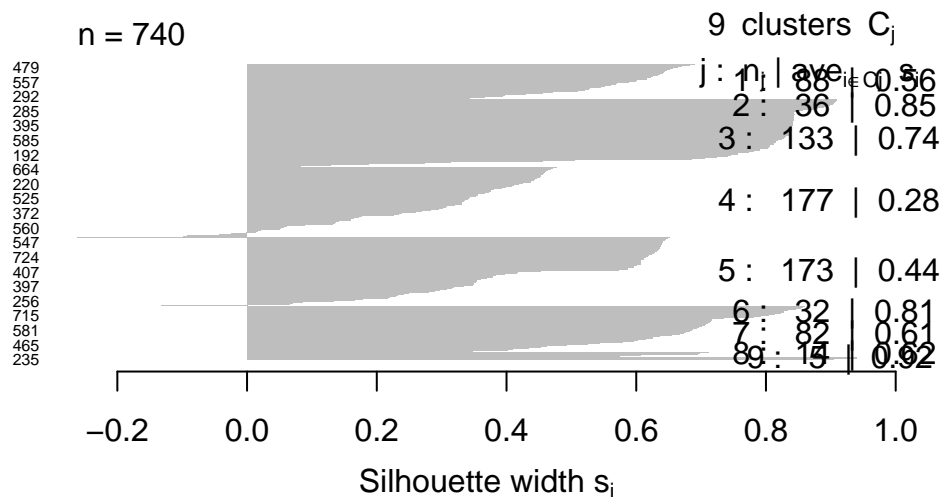
Average silhouette width : 0.54



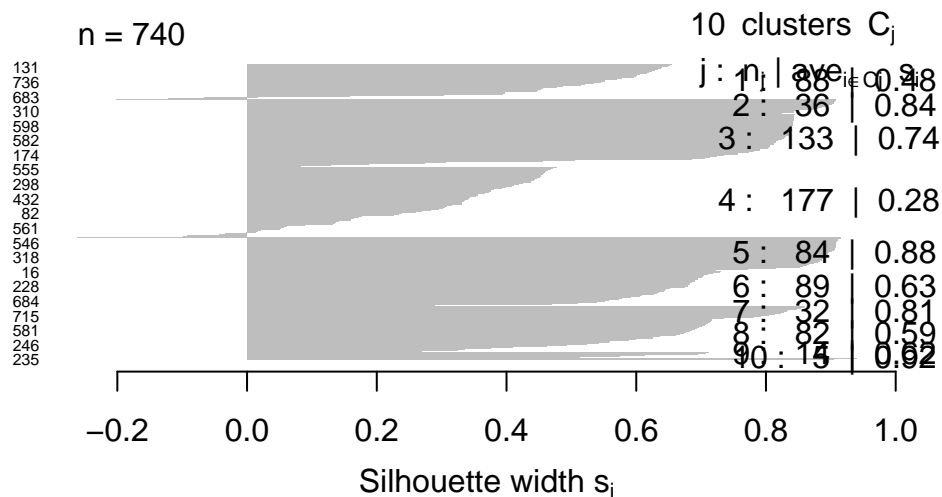
Average silhouette width : 0.53



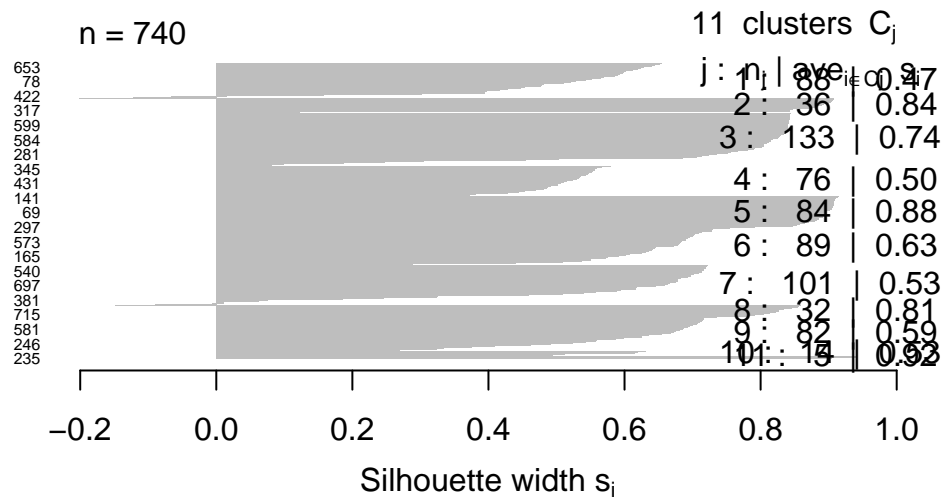
Average silhouette width : 0.5



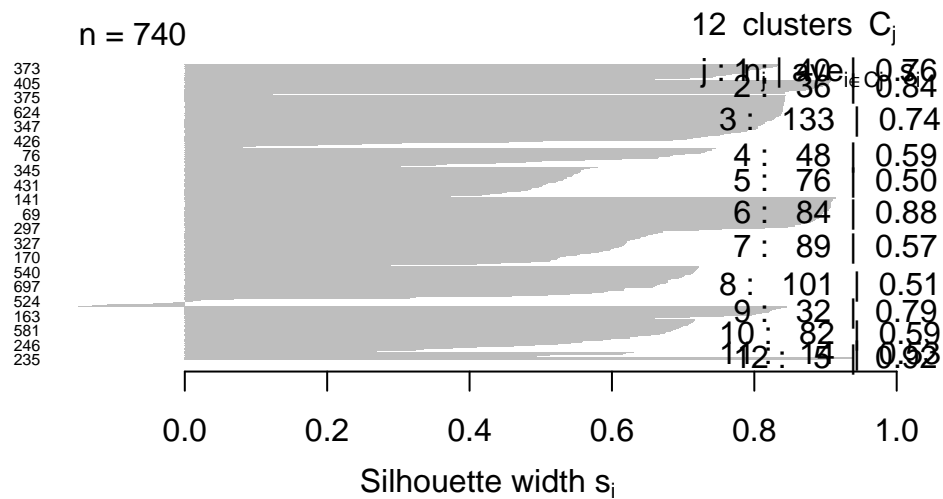
Average silhouette width : 0.53



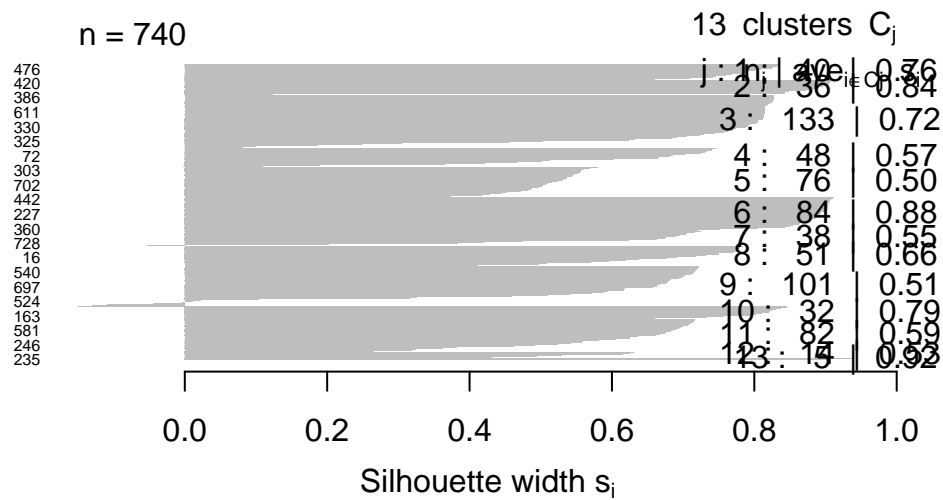
Average silhouette width : 0.59



Average silhouette width : 0.65



Average silhouette width : 0.66



Average silhouette width : 0.66

[1] 0.171699

**Introduction**

**Methods**

**Discussion**

## Supplementary material

```
# ----- SETUP ----- #
# Load packages
packages <- c("knitr", "tidyverse", "ggplot2", "cluster", "fossil")
lapply(packages, library, character.only = TRUE)

# Read data, extract labels, and keep only quantitative data
absentData_raw <- read.csv("Absenteeism_at_work.csv", sep = ";")
absentData_lab <- absentData_raw$`Reason.for.absence`
absentData <- absentData_raw %>%
  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, class)
min <- sapply(absentData, function(col){min(col, na.rm=TRUE)})
max <- sapply(absentData, function(col){max(col, na.rm=TRUE)})
nulls <- sapply(absentData, function(col){sum(is.na(col))})
blanks <- sapply(absentData,
  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$Transportation.expense, ylab = "Transportation Expense")
b02 <- boxplot(absentData$Distance.from.Residence.to.Work,
  ylab = "Distance from residence to work")
b03 <- boxplot(absentData$Service.time, ylab = "Service time")
b04 <- boxplot(absentData$Age, ylab = "Age")
b05 <- boxplot(absentData$Work.load.Average.day, ylab = "Work load average per day")
```

```

b06 <- boxplot(absentData$Hit.target, ylab = "Hit target")
b07 <- boxplot(absentData$Son, ylab = "Son")
b08 <- boxplot(absentData$Pet, ylab = "Pet")
b09 <- boxplot(absentData$Weight, ylab = "Weight")
b10 <- boxplot(absentData$Height, ylab = "Height")
b11 <- boxplot(absentData$Body.mass.index, ylab = "Body mass index")
b12 <- boxplot(absentData$Absenteeism.time.in.hours, ylab = "Absenteeism time in hours")

# ----- 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 %>%
  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))
}

# Get silhouette for hierarchical clustering
hcSilK <- function(k, data){
  hc_out <- hclust(dist(data))
  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) {
  sil_k <- lapply(k, silFun, data=data)
  sil_score <- sapply(sil_k, function(x) {mean(x[, "sil_width"])}))
  sil_max <- max(sil_score)
  sil_max_k <- match(sil_max, sil_score)+min(k)-1
  plot(x=k, y=sil_score, col=ifelse(sil_score==sil_max, "red", "black"))
  text(x=sil_max_k, y=sil_max, pos=1, col="red",
       labels= c(paste0("Max: (",sil_max_k,", ",round(sil_max,2), ")")))
}

```



```

#plotSil(kmcSilK(2, absentData_sd)) # k-means clustering
#plotSil(hcSilK(2, absentData_sd)) # hierarchical clustering
#plotSil(hcSilK(2, pr_out$x[, 1:2])) # hierarchical clustering after PCA
#chooseK(2:30, kmcSilK, absentData_sd)
#chooseK(2:30, hcSilK, absentData_sd)
#chooseK(2:30, hcSilK, pr_out$x[, 1:2])

# ----- AGGLOMERATIVE HIERARCHICAL CLUSTERING ----- #
# Compare linkage types
plot(hclust(dist(absentData_sd)), xlab = "", sub = "", ylab = "",
     labels = absentData_lab, main = "Complete Linkage")
plot(hclust(dist(absentData_sd), method = "average"),
     labels = absentData_lab, main = "Average Linkage",
     xlab = "", sub = "", ylab = "")
plot(hclust(dist(absentData_sd), method = "single"),
     labels = absentData_lab, main = "Single Linkage",
     xlab = "", sub = "", ylab = "")

# Choose k using goodness-of-clustering
set.seed(780)
plotHeirSilK <- function(k){
  hc_out <- hclust(dist(absentData_sd))
  hc_clusters <- cutree(hc_out, k)
  sil <- silhouette(hc_clusters, dist(absentData_sd))
  plot(sil, nmax= 800, cex.names=0.5, main = "", col=1:k, border=NA)
}
plotHeirSilK(2)
plotHeirSilK(3)
plotHeirSilK(4)
plotHeirSilK(5)

```

```

plotHeirSilK(6)
plotHeirSilK(7)
plotHeirSilK(8)
plotHeirSilK(9)
plotHeirSilK(10)

# Perform hierarchical clustering using k=2
set.seed(780)
hc_out <- hclust(dist(absentData_sd))
adj.rand.index(cutree(hc_out, k = 2), as.numeric(as.factor(absentData_lab)))
# ----- K-MEANS CLUSTERING ----- #
# Choose k using goodness-of-clustering
set.seed(780)
plotSilK <- function(k){
  x_k <- kmeans(absentData_sd, k, nstart = 20)
  sil <- silhouette(x_k$cluster, dist(absentData_sd))
  plot(sil, nmax= 800, cex.names=0.5, main = "", border=NA)
}
plotSilK(2)
plotSilK(3)
plotSilK(4)
plotSilK(5)
plotSilK(6)
plotSilK(7)
plotSilK(8)
plotSilK(9)
plotSilK(10)
plotSilK(11)
plotSilK(12)
plotSilK(13)
plotSilK(14)

```

```

plotSilK(15)
plotSilK(16)
plotSilK(17)
plotSilK(18)
plotSilK(19)
plotSilK(20)

# Perform k-means clustering with best k=15
set.seed(780)
km_out <- kmeans(absentData, 15, nstart = 20)
km_clusters <- km_out$cluster

# Compare the k-means clusters with the given labels.
# Compute the rand index between the labels and k-means clustering.
rand.index(km_clusters, as.numeric(as.factor(absentData_lab)))
adj.rand.index(km_clusters, as.numeric(as.factor(absentData_lab)))

# ----- HIERARCHICAL CLUSTERING AFTER PCA ----- #
# Principal component analysis (PCA)
pr_out <- prcomp(absentData, scale = TRUE)

# Proportion of variance explained
plot(pr_out)

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)")

plot(cumsum(pve), type = "o", ylab = "Cumulative PVE",
     xlab = "Principal Component", col = "brown3")

```

```

# Choose k using goodness-of-clustering
set.seed(2023780)
plotPCAHeirSilK <- function(k){
  hc_out <- hclust(dist(pr_out$x[, 1:2]))
  hc_clusters <- cutree(hc_out, k)
  sil <- silhouette(hc_clusters, dist(pr_out$x[, 1:2]))
  plot(sil, nmax= 800, cex.names=0.5, main = "", border=NA)
}

plotPCAHeirSilK(2)
plotPCAHeirSilK(3)
plotPCAHeirSilK(4)
plotPCAHeirSilK(5)
plotPCAHeirSilK(6)
plotPCAHeirSilK(7)
plotPCAHeirSilK(8)
plotPCAHeirSilK(9)
plotPCAHeirSilK(10)
plotPCAHeirSilK(11)
plotPCAHeirSilK(12)
plotPCAHeirSilK(13)

hc_out <- hclust(dist(dist(pr_out$x[, 1:2])))
hc_clusters <- cutree(hc_out, 11)
adj.rand.index(hc_clusters, as.numeric(as.factor(absentData_lab)))

```

## References

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