Cluster for microbiome count data via K-means

Weijia Xiong

6/25/2020

```
load("data/DiGiulio.RData")
otu_data = as.data.frame(DiGiulio$OTU)  # 927 samples, 1271 OTU
taxonomy = DiGiulio$Taxonomy  # 1271
sampledata = DiGiulio$SampleData  #  927 samples, other covariates
```

```
otu_data_all=
  cbind(sampledata, otu_data) %>%
  mutate(
    Preg = as.factor(Preg),
    Subject = as.factor(Subject)
  ) %>%
  na.omit()
rownames(otu_data_all) = sampledata$SampleID
```

```
term =
  otu_data_all %>%
  filter(preterm == "Term")

preterm =
  otu_data_all %>%
  filter(preterm != "Term")
```

Term

```
term_count =
  term %>%
  select(-SampleID,-Subject,-weeks,-Race,-NumReads,-Preg,-preterm,-CST)
ncol(term_count)
```

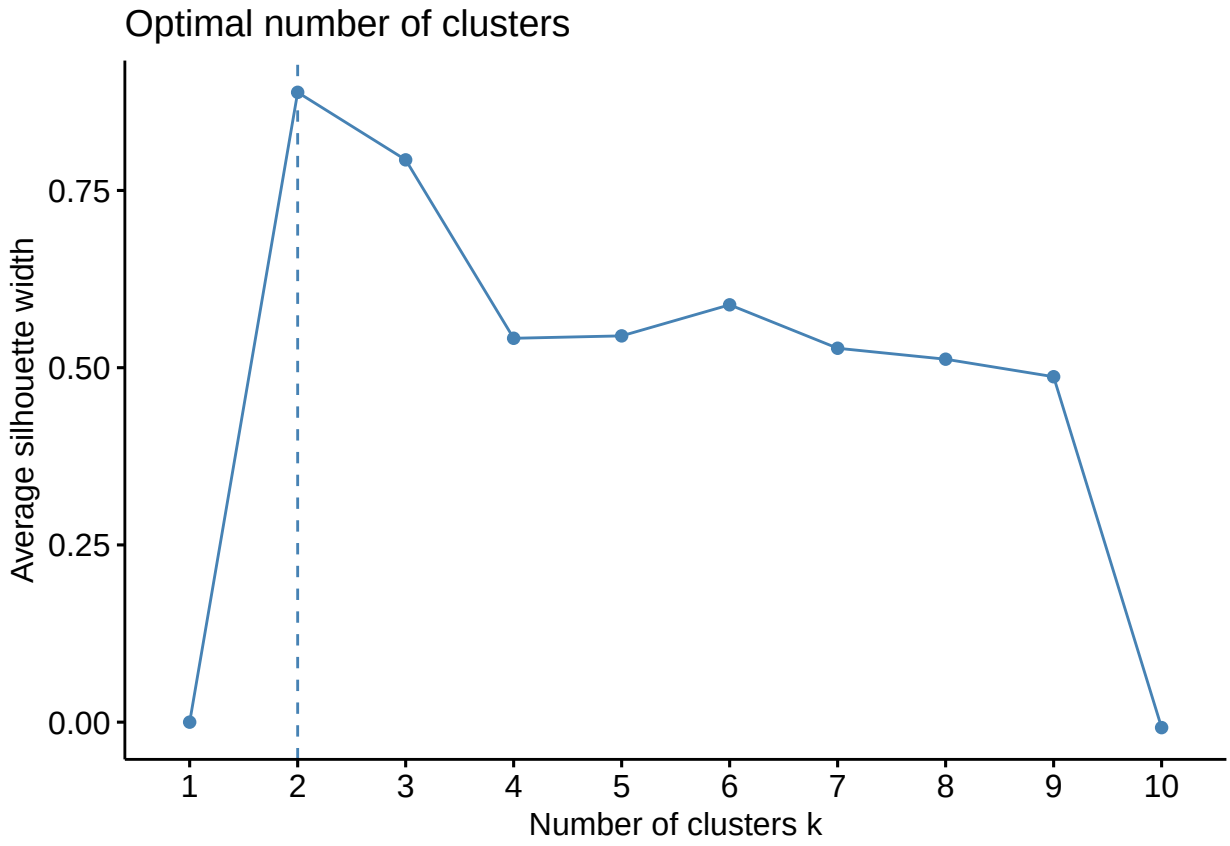
```
## [1] 1271
```

```
term_filter = term_count[,colSums(term_count) > 0] %>% scale()
ncol(term_filter)
```

```
## [1] 623
```

K-means cluster

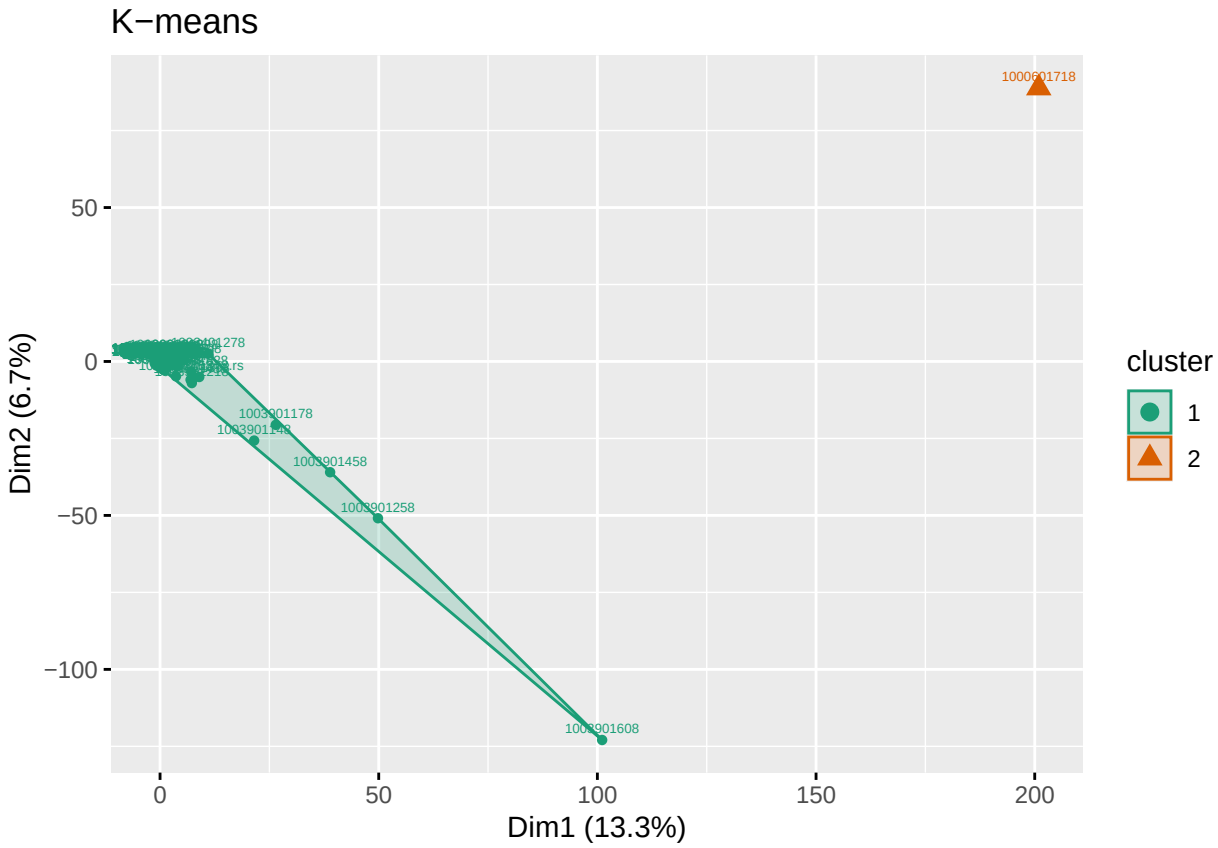
```
fviz_nbclust(term_filter,
             FUNcluster = kmeans,
             method = "silhouette")
```




```
set.seed(1)
km_term <- kmeans(term_filter, centers = 2, nstart = 20)
```

```
km_vis_term <- fviz_cluster(list(data = term_filter, cluster = km_term$cluster),
                            ellipse.type = "convex",
                            geom = c("point","text"),
                            labelsize = 5,
                            palette = "Dark2") + labs(title = "K-means")

km_vis_term
```




```
term[km_term$cluster == 2, 1:10]
```

```
##              SampleID Subject weeks  Race NumReads  Preg preterm CST 4330849
## 1000601718 1000601718   10006    71 White    10165 FALSE    Term   0       0
##            4400869
## 1000601718       0
```

Hierarchical clustering

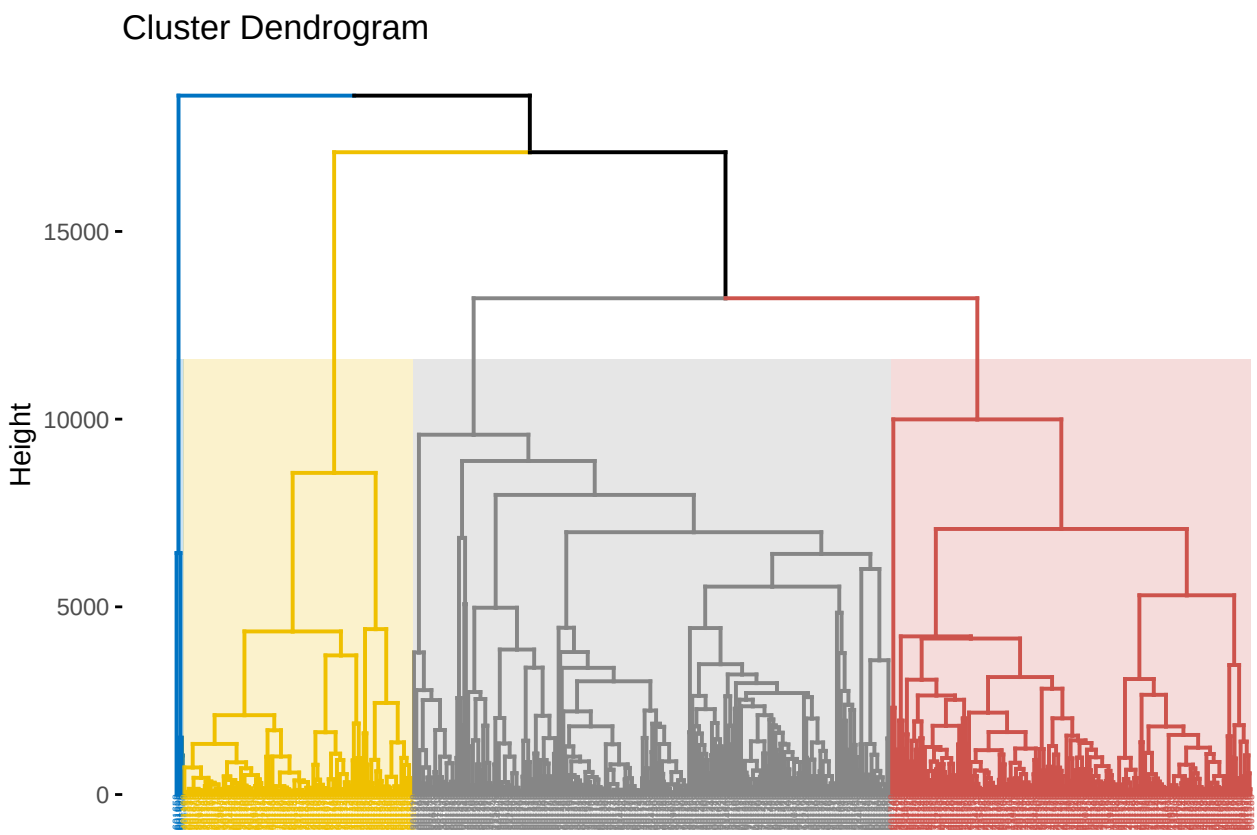
We can also apply hierarchical clustering on this data. Here we use the Euclidean distance and different types of linkage.

```
dat1 = term_count
hc.complete <- hclust(dist(dat1), method = "complete")

# distance.bray<-vegdist(dat1,method="bray",na.rm=TRUE)
# hc.bray<- hclust(distance.bray,method="complete")
```

The function `fviz_dend()` can be applied to visualize the dendrogram.

```
fviz_dend(hc.complete, k = 4,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```

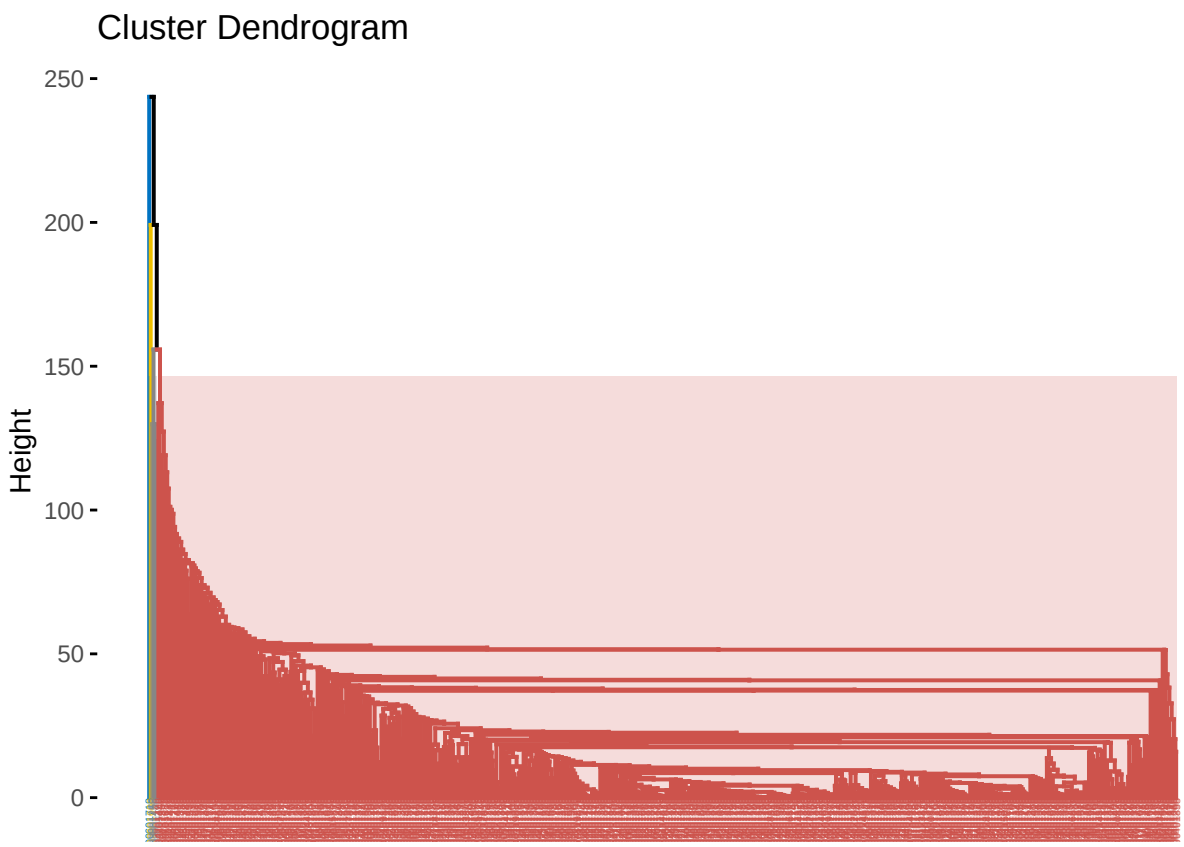



```
# Who are in the fourth cluster?
complete <- cutree(hc.complete, 4)
term[complete == 4,1:10]
```

```
##              SampleID Subject weeks  Race NumReads  Preg preterm CST 4330849
## 1000601528 1000601528   10006    53 White     7981 FALSE    Term   0       0
## 1000601608 1000601608   10006    60 White     8472 FALSE    Term   0       0
## 1000601658 1000601658   10006    66 White    13533 FALSE    Term   0       0
## 1000601718 1000601718   10006    71 White    10165 FALSE    Term   0       0
## 1004501308 1004501308   10045    31 White     7152  TRUE    Term   0       0
##            4400869
## 1000601528       0
## 1000601608       0
## 1000601658       0
## 1000601718       0
## 1004501308       0
```

After scaling and filtering

```
dat1 = term_filter
hc.complete <- hclust(dist(dat1), method = "complete")
fviz_dend(hc.complete, k = 4,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```




```
complete <- cutree(hc.complete, 4)
preterm[complete == 4,1:10]
```

```
##    SampleID Subject weeks Race NumReads Preg preterm CST 4330849 4400869
## NA     <NA>    <NA>    NA <NA>       NA <NA>    <NA>  NA       NA       NA
```

Preterm

```
preterm_count =
  preterm %>%
  select(-SampleID,-Subject,-weeks,-Race,-NumReads,-Preg,-preterm,-CST)
ncol(preterm_count)
```

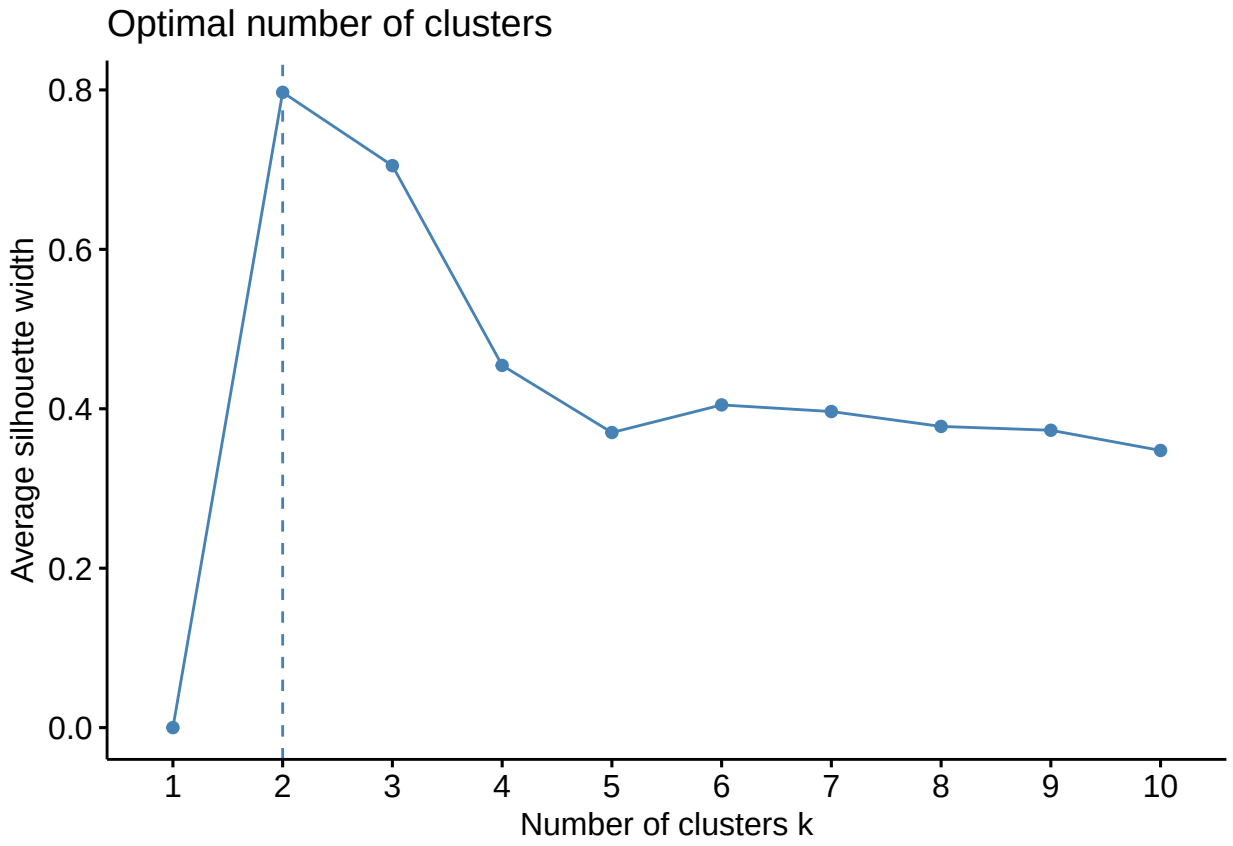
```
## [1] 1271
```

```
preterm_filter = preterm_count[,colSums(preterm_count) > 0] %>% scale()
ncol(preterm_filter)
```

```
## [1] 514
```

K-means cluster

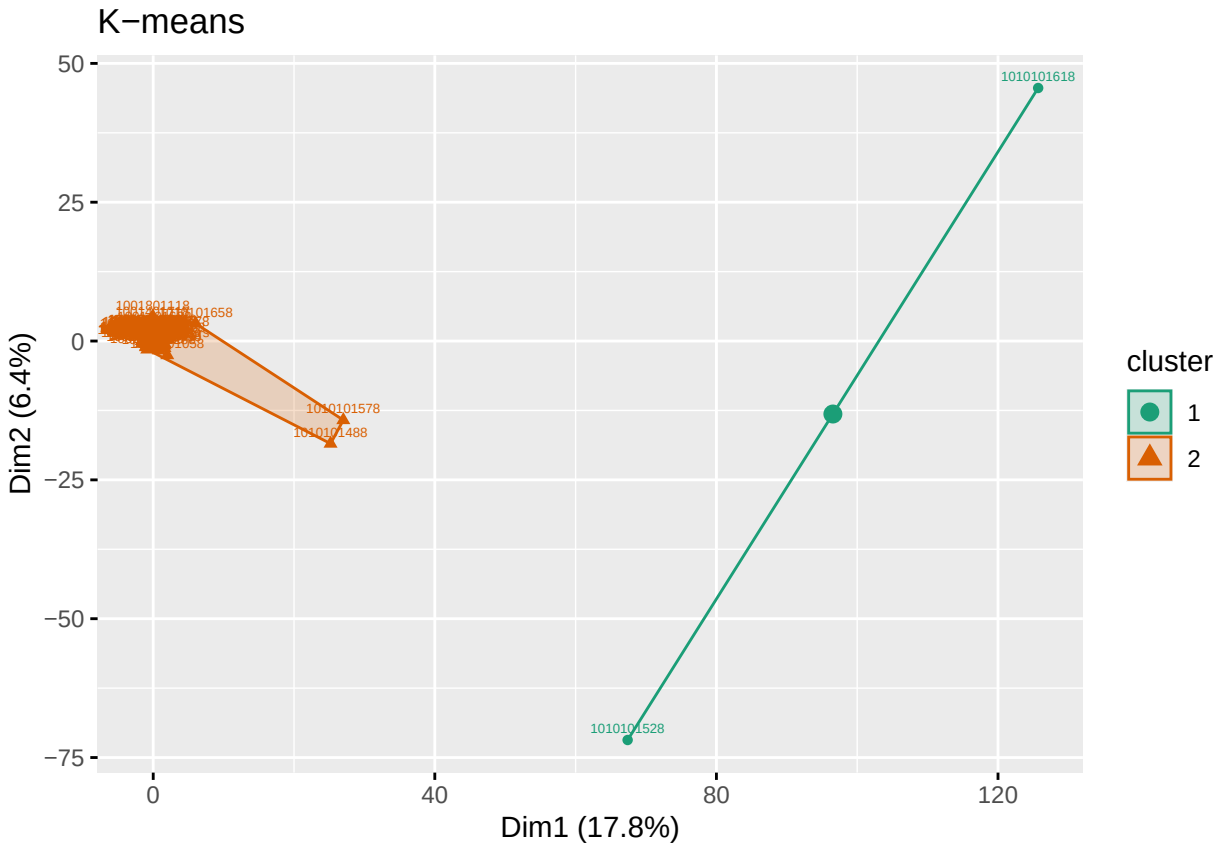
```
fviz_nbclust(preterm_filter,
             FUNcluster = kmeans,
             method = "silhouette")
```




```
set.seed(1)
km_preterm <- kmeans(preterm_filter, centers = 2, nstart = 20)
```

```
km_vis_preterm <- fviz_cluster(list(data = preterm_filter, cluster = km_preterm$cluster),
                               ellipse.type = "convex",
                               geom = c("point","text"),
                               labelsize = 5,
                               palette = "Dark2") + labs(title = "K-means")

km_vis_preterm
```




```
preterm[km_preterm$cluster == 1, 1:10]
```

```
##              SampleID Subject weeks  Race NumReads  Preg  preterm CST 4330849
## 1010101528 1010101528   10101    48 White     4078 FALSE Marginal   0       0
## 1010101618 1010101618   10101    58 White     9103 FALSE Marginal   0       0
##            4400869
## 1010101528       0
## 1010101618       0
```

Hierarchical clustering

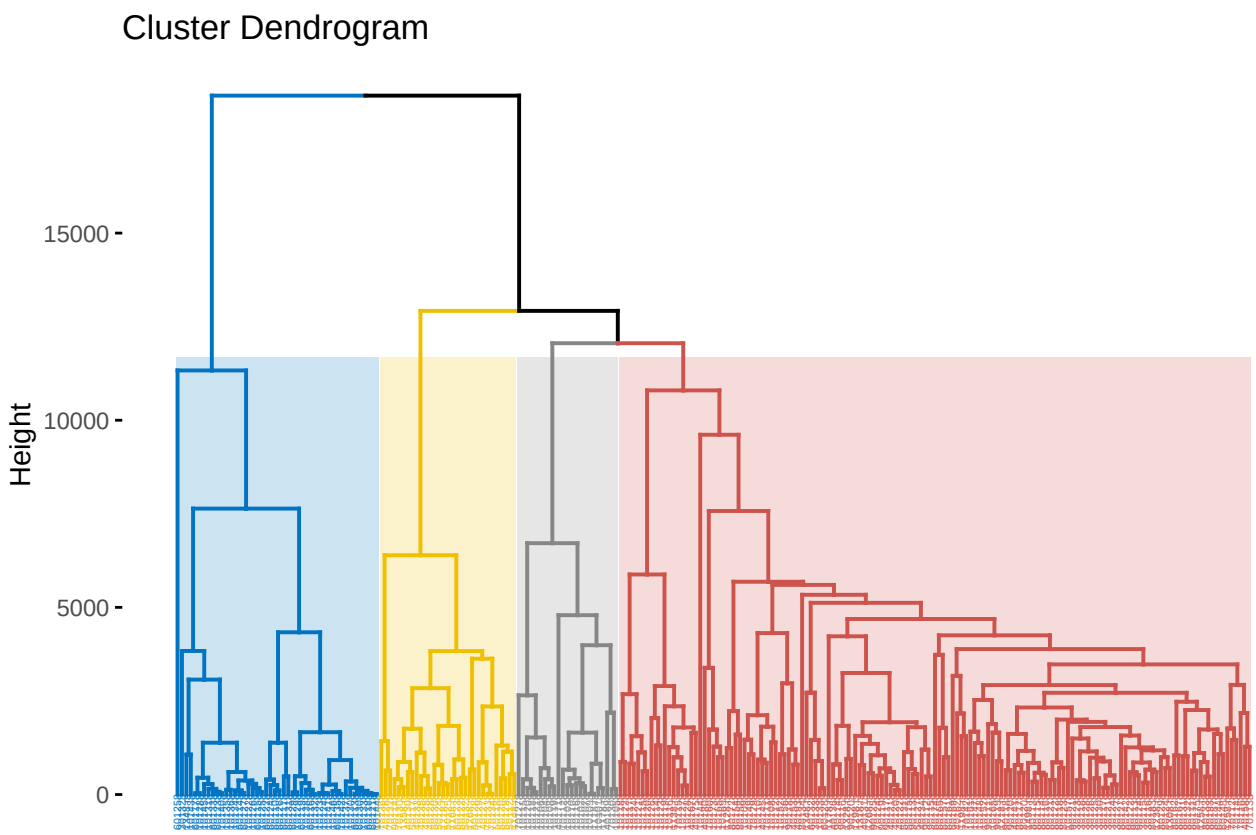
We can also apply hierarchical clustering on this data. Here we use the Euclidean distance and different types of linkage.

```
dat1 = preterm_count
hc.complete <- hclust(dist(dat1), method = "complete")

# distance.bray<-vegdist(dat1,method="bray",na.rm=TRUE)
# hc.bray<- hclust(distance.bray,method="complete")
```

The function `fviz_dend()` can be applied to visualize the dendrogram.

```
fviz_dend(hc.complete, k = 4,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```



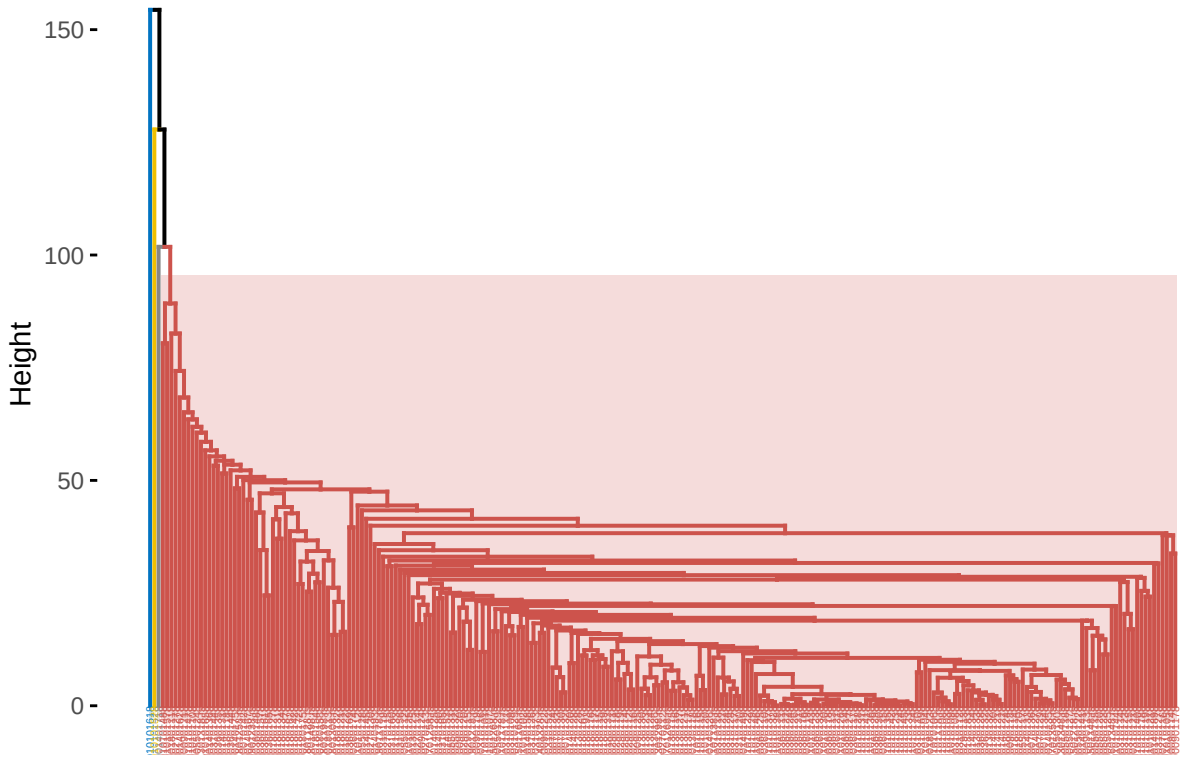

```
# Who are in the fourth cluster?
complete <- cutree(hc.complete, 4)
preterm[complete == 4,1:10]
```

```
##                   SampleID Subject weeks                  Race NumReads  Preg
## 1001401718       1001401718   10014    68     Asian-Unspecified    10311 FALSE
## 1001401898       1001401898   10014    86     Asian-Unspecified     5369 FALSE
## 1002701278       1002701278   10027    28 Other (Specify below)    11515  TRUE
## 1002701308       1002701308   10027    30 Other (Specify below)     3218  TRUE
## 1010101018       1010101018   10101    -7                 White     4580 FALSE
## 1010101028       1010101028   10101    -7                 White     6229 FALSE
## 1010101038       1010101038   10101    -5                 White     5697 FALSE
## 1010101048       1010101048   10101    -5                 White     6938 FALSE
## 1010101058       1010101058   10101    -3                 White     3182 FALSE
## 1010101068       1010101068   10101    -2                 White     7660 FALSE
## 1010101078       1010101078   10101    -2                 White     6293 FALSE
## 1010101088       1010101088   10101    -1                 White     7165 FALSE
## 1010101098       1010101098   10101     0                 White     5386 FALSE
## 1010101108       1010101108   10101     1                 White     6356  TRUE
## 1010101118.rs 1010101118.rs   10101     1                 White     5389  TRUE
## 1010101128       1010101128   10101     2                 White     6854  TRUE
## 1010101158       1010101158   10101     6                 White     8581  TRUE
## 1010101168       1010101168   10101     7                 White     6283  TRUE
## 1010101178.rs 1010101178.rs   10101     7                 White     4826  TRUE
## 1010101188       1010101188   10101     8                 White     8354  TRUE
## 1010101198       1010101198   10101    10                 White     7579  TRUE
## 1010101208       1010101208   10101    11                 White     8262  TRUE
## 1010101218       1010101218   10101    11                 White     8370  TRUE
```

```
##                preterm CST 4330849 4400869
## 1001401718    Marginal   0       0       0
## 1001401898    Marginal   0       0       0
## 1002701278    Marginal   0       0       0
## 1002701308    Marginal   0       0       0
## 1010101018    Marginal   0       0       0
## 1010101028    Marginal   0       0       0
## 1010101038    Marginal   0       0       0
## 1010101048    Marginal   0       0       0
## 1010101058    Marginal   0       0       0
## 1010101068    Marginal   0       0       0
## 1010101078    Marginal   0       0       0
## 1010101088    Marginal   0       0       0
## 1010101098    Marginal   0       0       0
## 1010101108    Marginal   0       0       0
## 1010101118.rs Marginal   0       0       0
## 1010101128    Marginal   0       0       0
## 1010101158    Marginal   0       0       0
## 1010101168    Marginal   0       0       0
## 1010101178.rs Marginal   0       0       0
## 1010101188    Marginal   0       0       0
## 1010101198    Marginal   0       0       0
## 1010101208    Marginal   0       0       0
## 1010101218    Marginal   0       0       0
```

After scaling and filtering

```
dat1 = preterm_filter
hc.complete <- hclust(dist(dat1), method = "complete")
fviz_dend(hc.complete, k = 4,
          cex = 0.3,
          palette = "jco",
          color_labels_by_k = TRUE,
          rect = TRUE, rect_fill = TRUE, rect_border = "jco",
          labels_track_height = 2.5)
```




```
complete <- cutree(hc.complete, 4)
preterm[complete == 4,1:10]
```

```
##              SampleID Subject weeks  Race NumReads  Preg  preterm CST 4330849
## 1010101618 1010101618   10101    58 White     9103 FALSE Marginal   0       0
##            4400869
## 1010101618       0
```