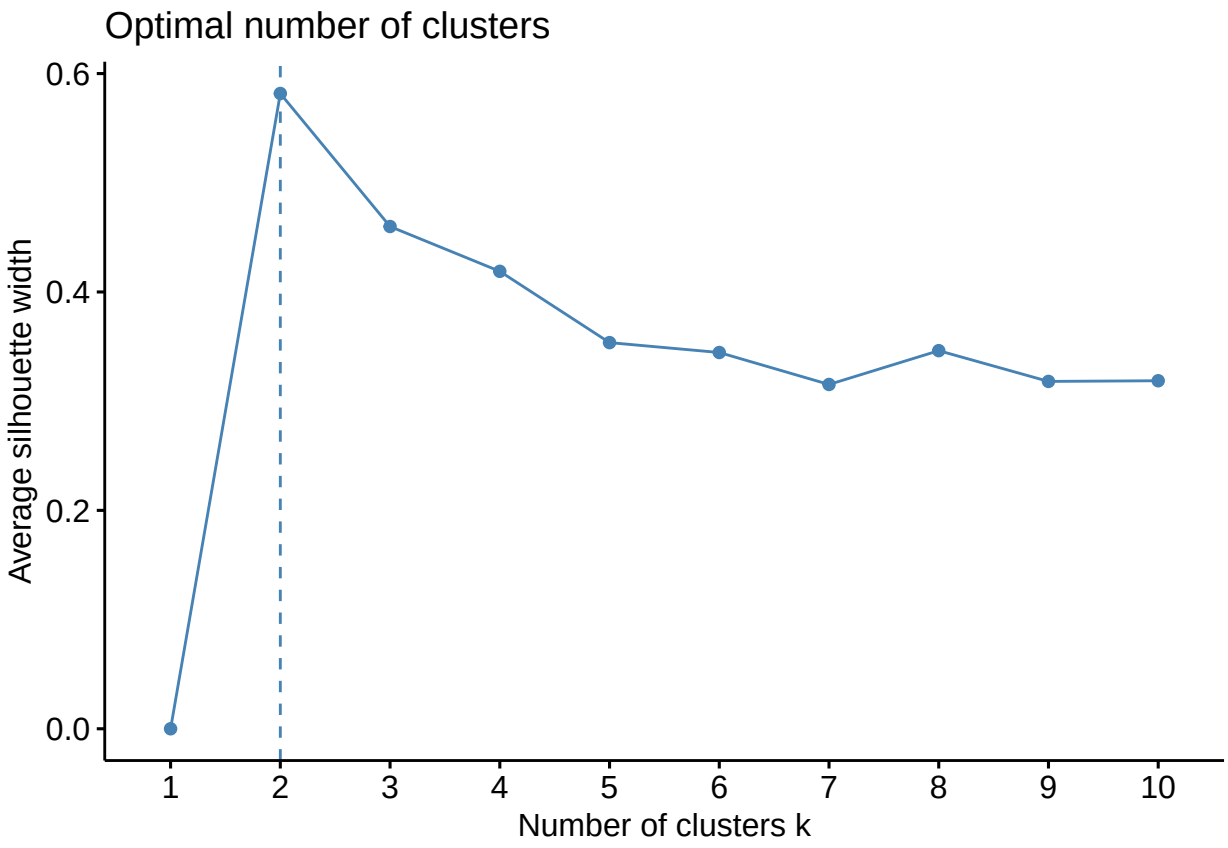
ex_clusters_Machine_Leraning

Identify clusters in the data

```
df <- iris[,1:4]
# find optimal number of clusters
  dff <- scale(df)                    # base R function
  factoextra::fviz_nbclust(dff, kmeans, method = "silhouette")
```



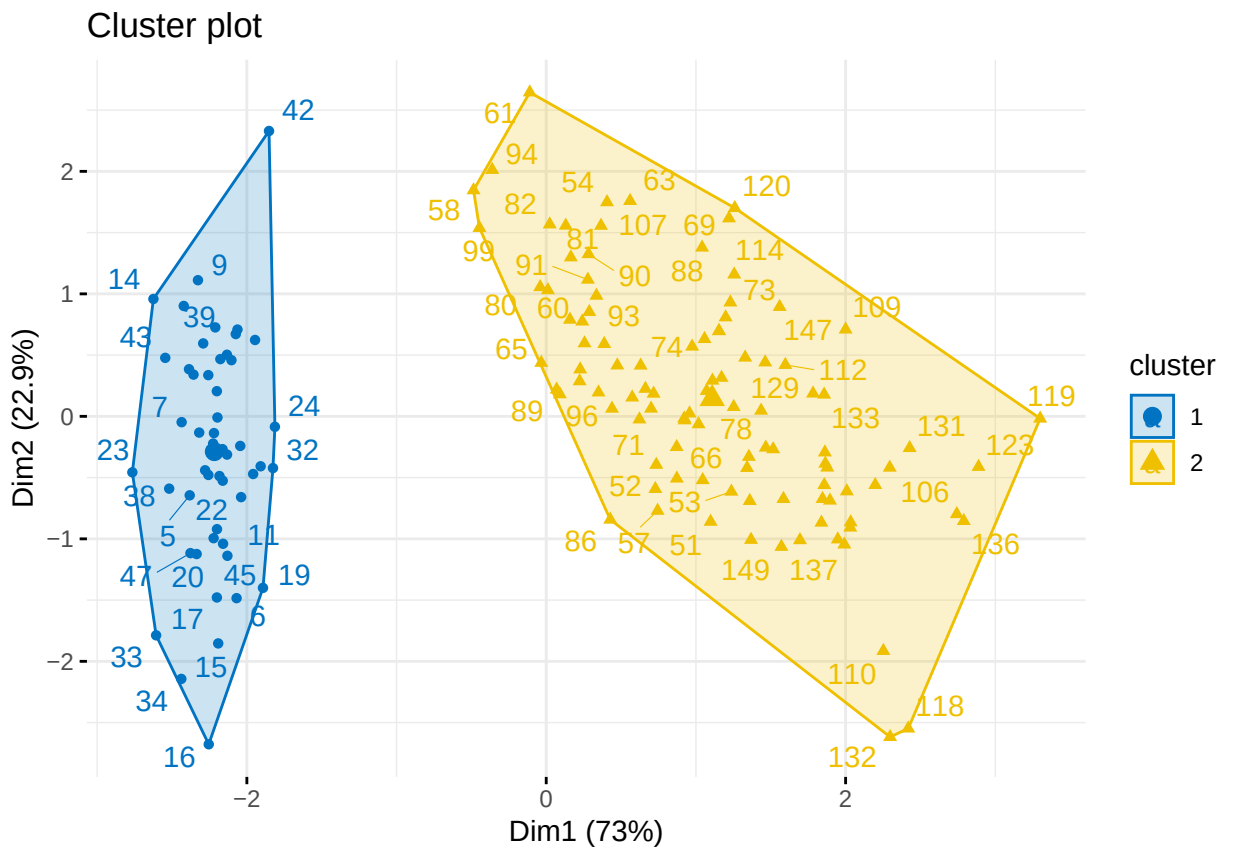

```
  # methods may be<. gap_stat, wss, silhouette
  #WHICH k? Where sudden "change decay slope"!
# compute and visualise
  set.seed(123)
  km.res <- kmeans(dff, 2, nstart = 25)
  km.res
```

```
## K-means clustering with 2 clusters of sizes 50, 100
##
```

```
## Cluster means:
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1   -1.0111914   0.8504137    -1.300630  -1.2507035
## 2    0.5055957  -0.4252069     0.650315   0.6253518
##
## Clustering vector:
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [149] 2 2
##
## Within cluster sum of squares by cluster:
## [1]  47.35062 173.52867
##  (between_SS / total_SS =  62.9 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```
# visualize
  factoextra::fviz_cluster(km.res, data = dff,
               ellipse.type = "convex",
                 # "norm",
                 # "convex",
               palette = "jco",
               repel = TRUE,
               ggtheme = ggplot2::theme_minimal())
```

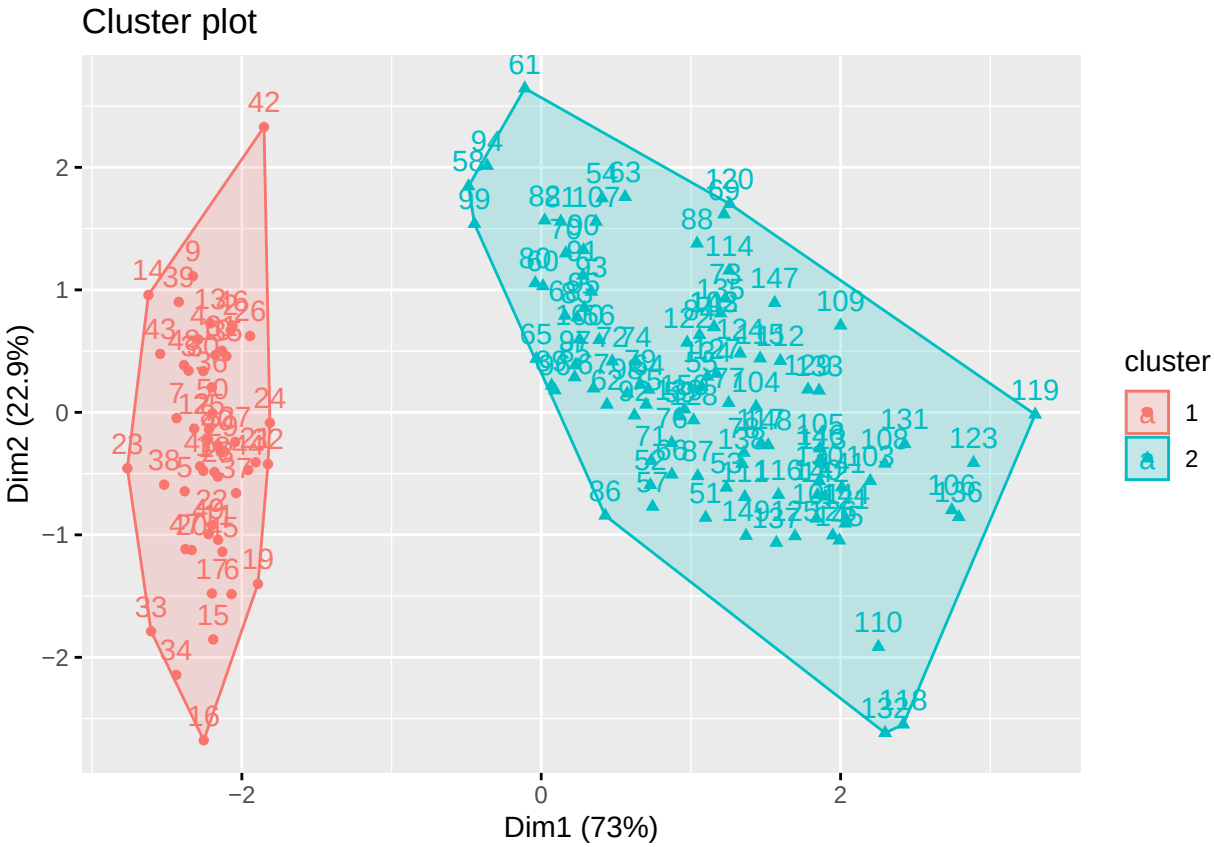
```
## Warning: ggrepel: 81 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```




```
# may adjust max.overlaps to show all labels on plots
```

. . . and continue

```
# compare with PAM clustering
  # Compute PAM
  pam.res <- cluster::pam(dff, 2)
  # Visualize
  factoextra::fviz_cluster(pam.res)
```


Cluster plot
Dim2 (22.9%)
Dim1 (73%)
cluster
1
2