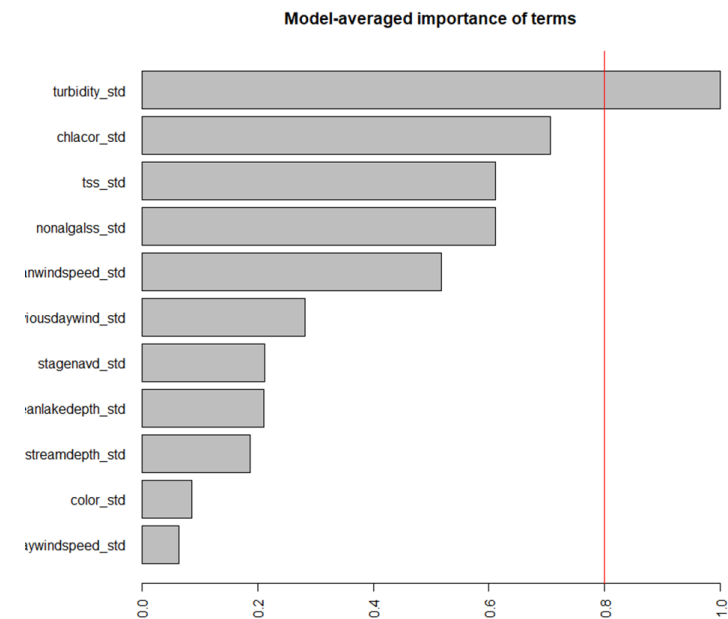
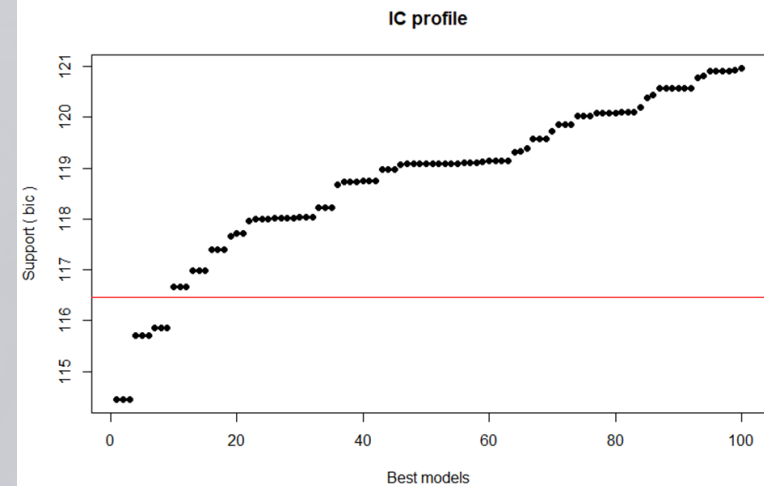


Advanced R: Statistical Machine Learning

Dan Schmutz, MS
Chief Environmental Scientist

Zoom Workshop for SJRWMD
September 24, 2020



Overview of Unsupervised Learning Algorithms

Unsupervised Learning

- Clustering – segment observations (i.e., rows of dataframe) into similar groups based on the observed variables, e.g.:
 - K-means clustering
 - Hierarchical clustering
- Dimension reduction – reducing the number of variables in a data set (i.e., columns of dataframe), e.g.,:
 - PCA (Principal Components Analysis)

Iris dataset

- Fisher's iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper The use of multiple measurements in taxonomic problems as an example of linear discriminant analysis in Annals of Eugenics* 7 (2): 179–188.

https://en.wikipedia.org/wiki/Iris_flower_data_set



Iris setosa



Iris versicolor



Iris virginica

*Journal renamed in 1954 to Journal of Human Genetics reflecting changing perceptions on eugenics



Parts of an iris



Petal

Sepal

Photo Credit: Dan Schmutz



Hierarchical Clustering

- Can unsupervised learning using the sepal and petal measurements match the actual species classifications?
- First calculate a distance matrix (Euclidian in this case)

```

irisb<-iris
# hierarchical clustering based on https://cran.r-project.org/web/packages/dendextend/vignette:
d_iris <- dist(irisb) # method="man" # is a bit better
hc_iris <- hclust(d_iris, method = "average")
iris_species <- rev(levels(iris[,5]))

library(dendextend)
dend <- as.dendrogram(hc_iris)
# order it the closest we can to the order of the observations:
dend <- rotate(dend, 1:150)

# Color the branches based on the clusters:
dend <- color_branches(dend, k=3) #, groupLabels=iris_species)

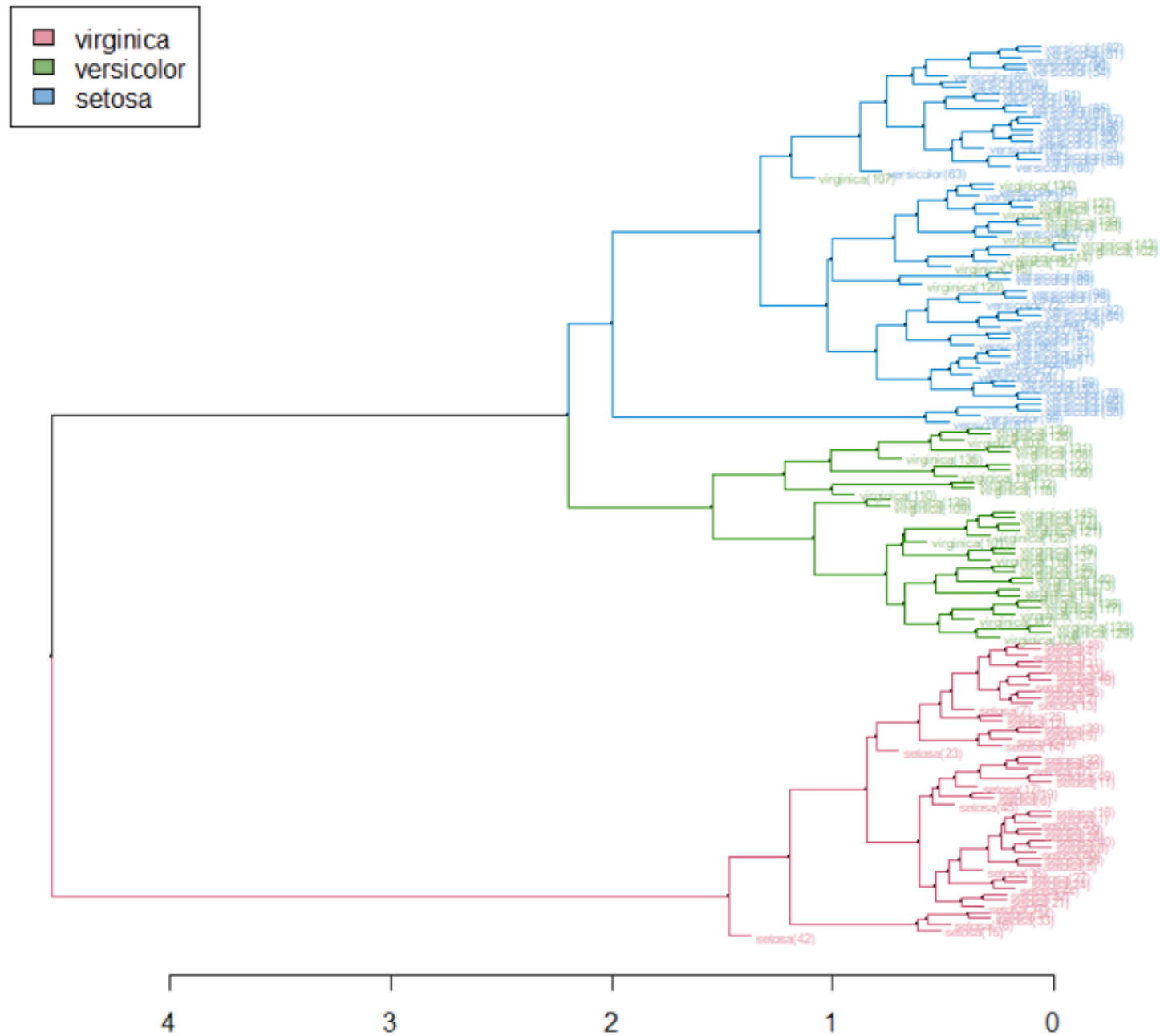
# Manually match the labels, as much as possible, to the real classification of the flowers:
labels_colors(dend) <-
  rainbow_hcl(3)[sort_levels_values(
    as.numeric(iris[,5])[order.dendrogram(dend)]
  )]

# We shall add the flower type to the labels:
labels(dend) <- paste(as.character(iris[,5])[order.dendrogram(dend)],
  "(", labels(dend), ")",
  sep = "")

# We hang the dendrogram a bit:
dend <- hang.dendrogram(dend, hang_height=0.1)
# reduce the size of the labels:
# dend <- assign_values_to_leaves_nodePar(dend, 0.5, "lab.cex")
dend <- set(dend, "labels_cex", 0.5)
# And plot:
par(mar = c(3,3,3,7))
plot(dend,
  main = "Clustered Iris data set
  (the labels give the true flower species)",
  horiz = TRUE, nodePar = list(cex = .007))
legend("topleft", legend = iris_species, fill = rainbow_hcl(3))

```


GFI



K-means clustering pseudocode

- Specify the number of clusters (K) to be created (by the analyst)
- Select randomly k objects from the dataset as the initial cluster centers or means
- Assigns each observation to their closest centroid, based on the Euclidean distance between the object and the centroid
- For each of the k clusters update the cluster centroid by calculating the new mean values of all the data points in the cluster. The centroid of a Kth cluster is a vector of length p containing the means of all variables for the observations in the kth cluster; p is the number of variables.
- Iteratively minimize the total within sum of square. That is, iterate steps 3 and 4 until the cluster assignments stop changing or the maximum number of iterations is reached. By default, the R software uses 10 as the default value for the maximum number of iterations.

Determining k

```
# k-means clustering from factoextra package
library(factoextra)
# Remove species column (5) and scale the data
iris.scaled <- scale(iris[, -5])

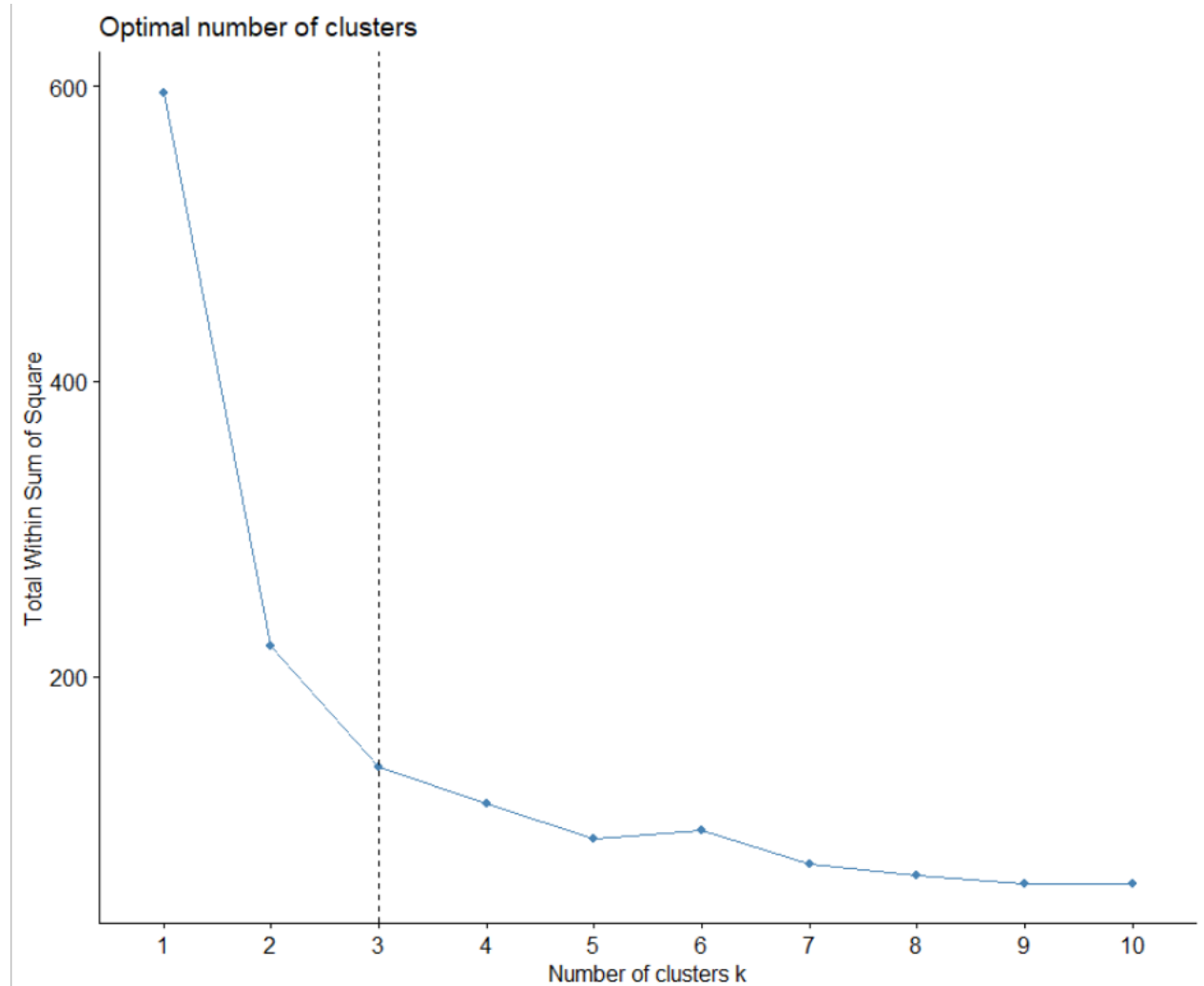
# Optimal number of clusters in the data
# ++++++
# Examples are provided only for kmeans, but
# you can also use cluster::pam (for pam) or
# hcut (for hierarchical clustering)

### Elbow method (look at the knee)
# Elbow method for kmeans
fviz_nbclust(iris.scaled, kmeans, method = "wss") +
  geom_vline(xintercept = 3, linetype = 2)

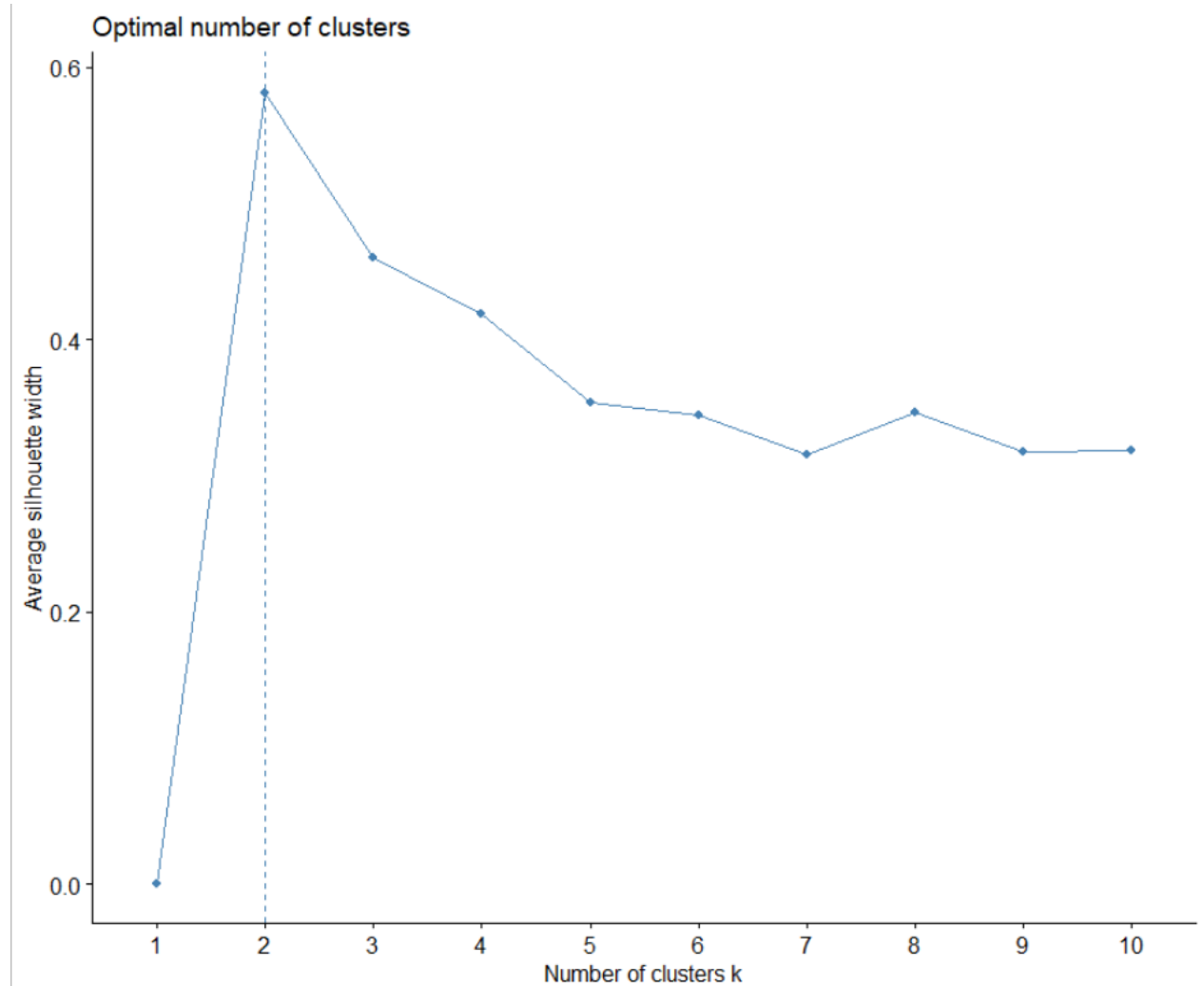
# Average silhouette for kmeans
fviz_nbclust(iris.scaled, kmeans, method = "silhouette")

### Gap statistic
library(cluster)
set.seed(123)
# Compute gap statistic for kmeans
# we used B = 10 for demo. Recommended value is ~500
gap_stat <- clusGap(iris.scaled, FUN = kmeans, nstart = 25,
                    K.max = 10, B = 10)
print(gap_stat, method = "firstmax")
fviz_gap_stat(gap_stat)
```

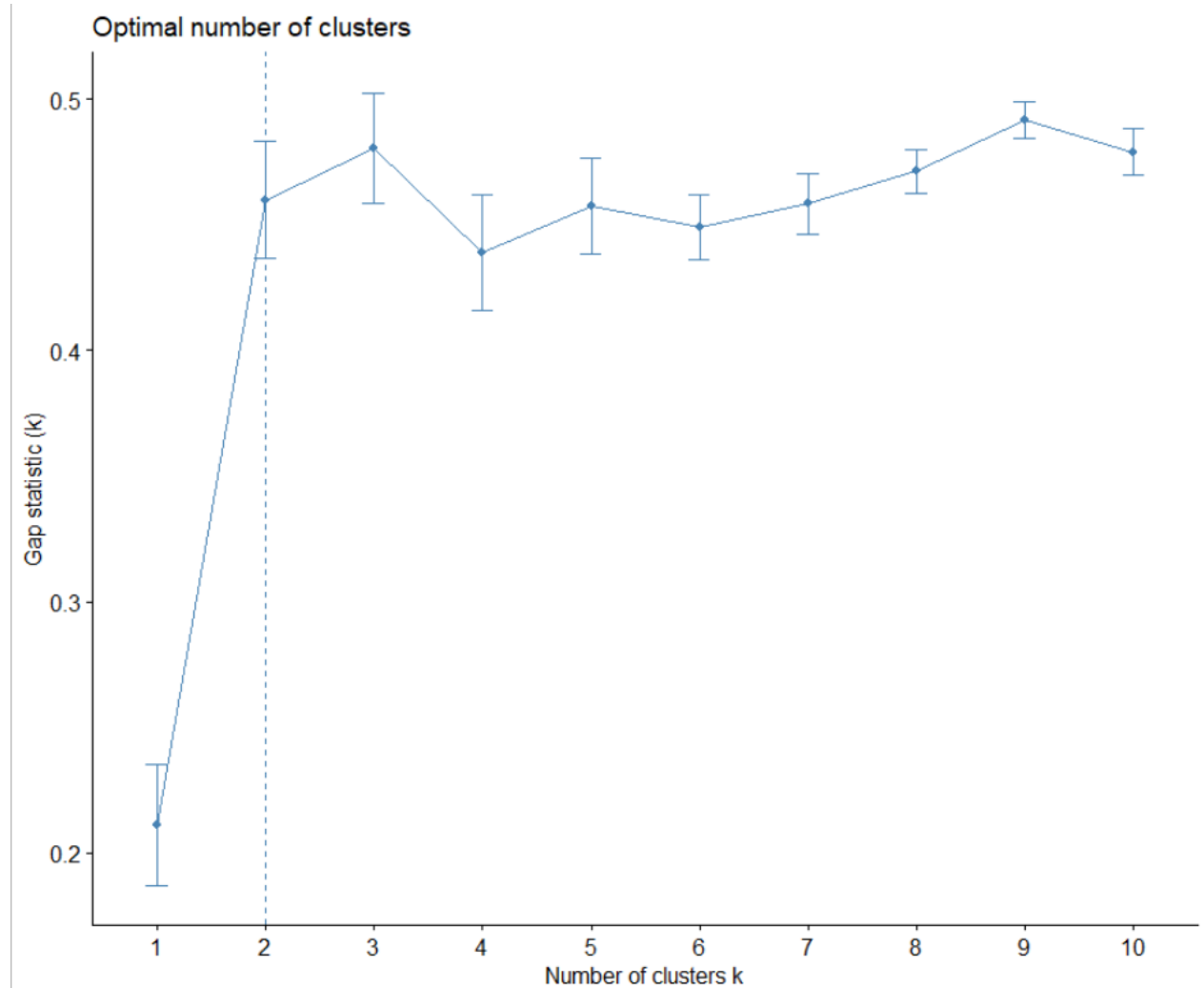
Optimal number of clusters using knee method



Optimal number of clusters using silhouette method



Optimal number of clusters using gap statistic method

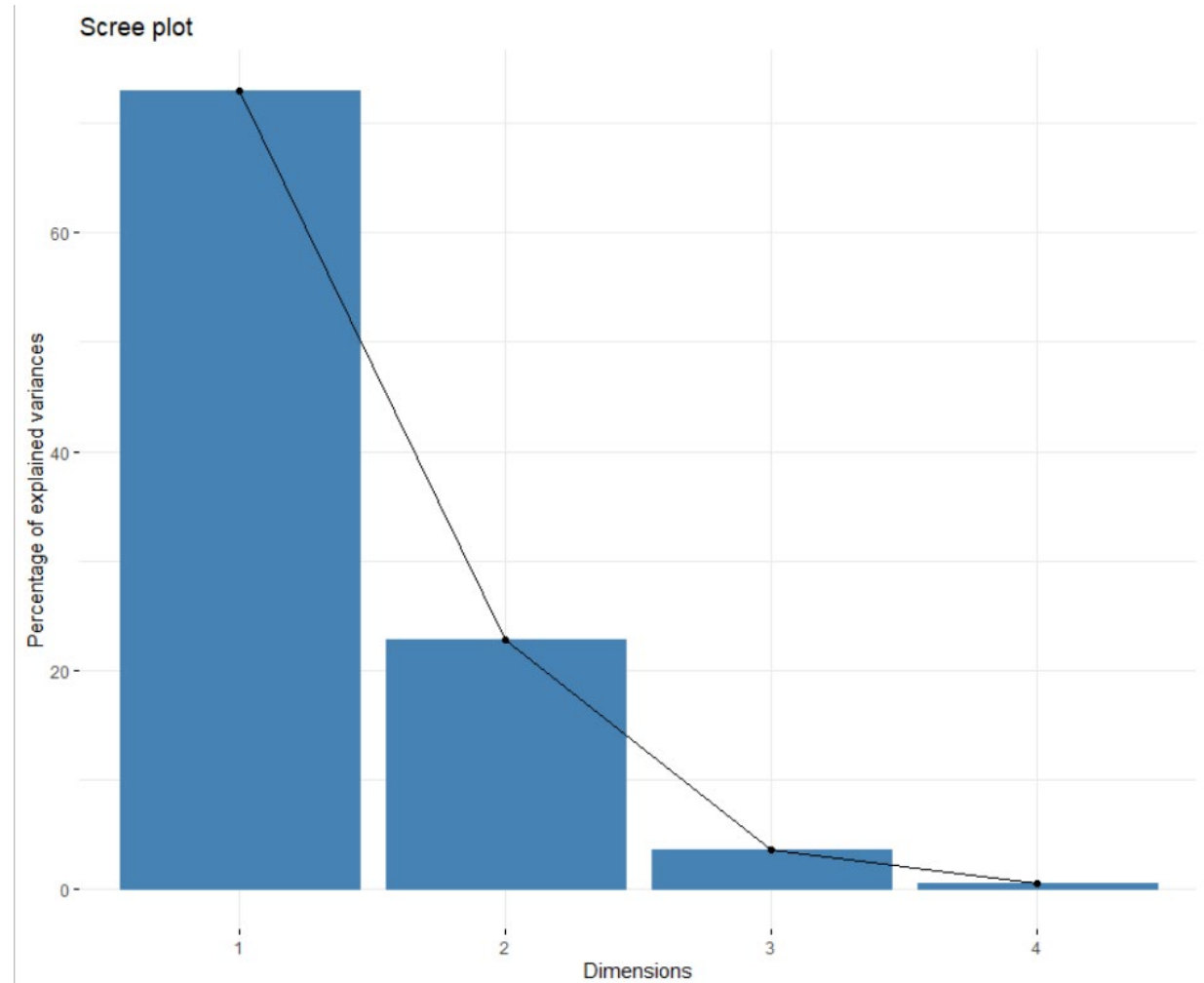




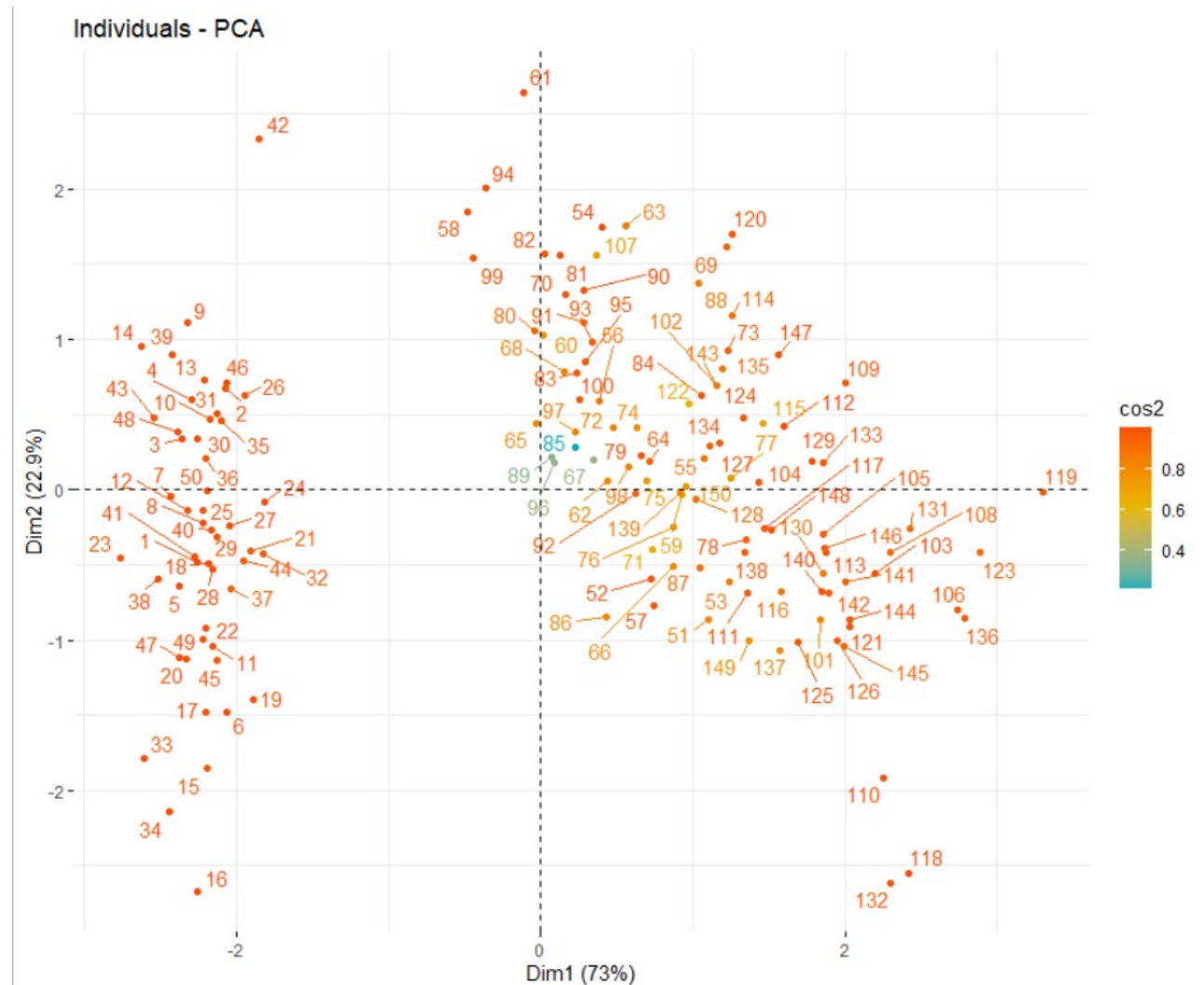
Principal Components Analysis

```
res.pca <- prcomp(irisb[,-5], scale = TRUE)  
fviz_eig(res.pca)
```

- First two dimensions capture most of variability

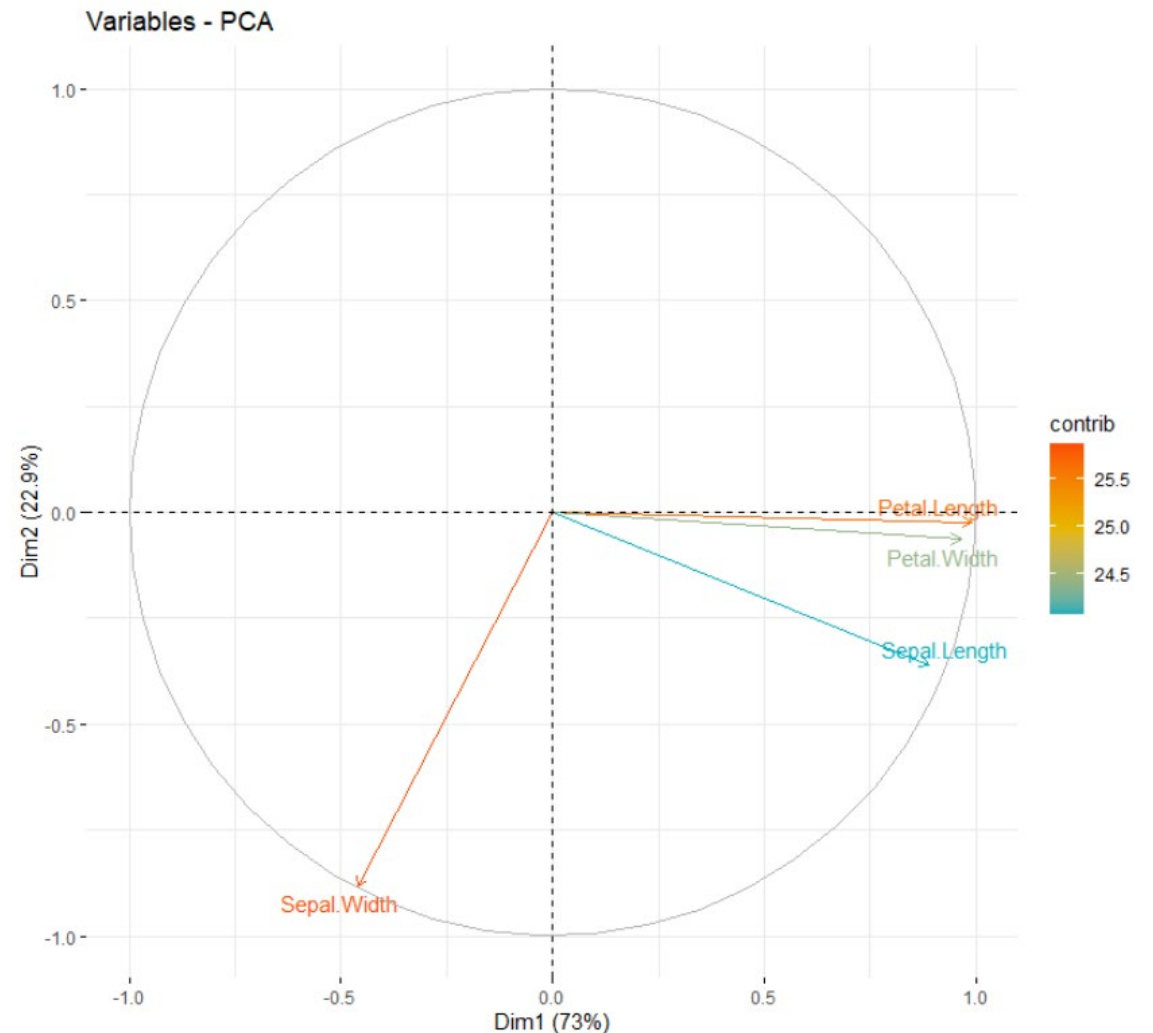


Cases on first two PCs



Graph of variables in PCA space

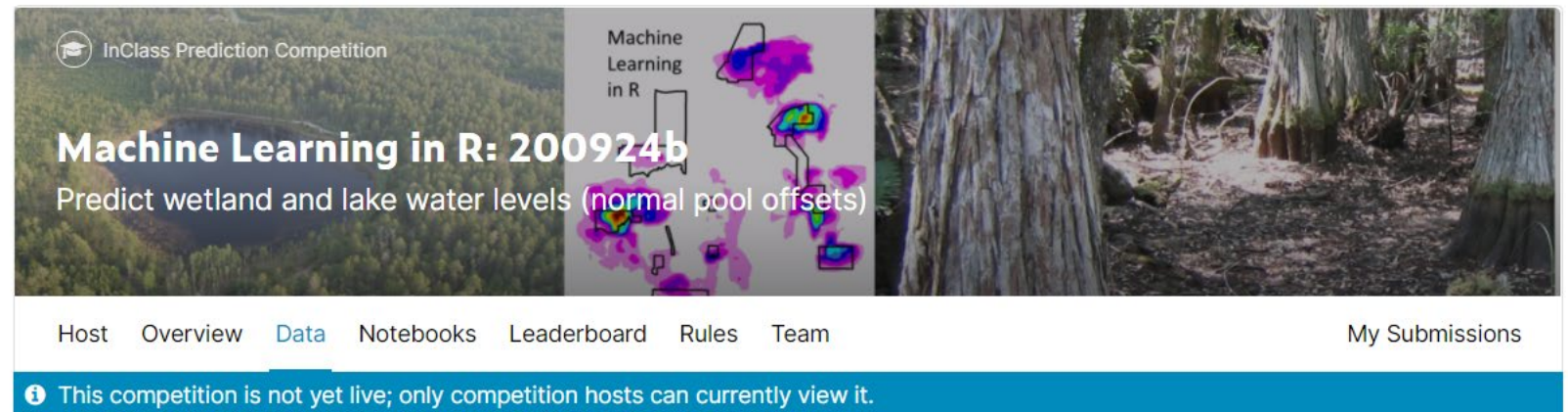
- Positive correlated variables point to the same side of the plot. Negative correlated variables point to opposite sides of the graph.
- Sepal Width different than all the other variables



Learning Competition

- Let's check out Kaggle together
- I'm working on a contest
- In the meantime, there are many opportunities to learn more

Search



The screenshot shows a Kaggle competition page. At the top, there's a search bar. Below it, the competition title 'Machine Learning in R: 200924b' is displayed, along with the subtitle 'Predict wetland and lake water levels (normal pool offsets)'. The page features a navigation bar with links: Host, Overview, Data, Notebooks, Leaderboard, Rules, Team, and My Submissions. A blue banner at the bottom states: 'This competition is not yet live; only competition hosts can currently view it.'

InClass Prediction Competition

Machine Learning in R: 200924b
Predict wetland and lake water levels (normal pool offsets)

Machine Learning in R

Host Overview **Data** Notebooks Leaderboard Rules Team My Submissions

i This competition is not yet live; only competition hosts can currently view it.