NC STATE UNIVERSITY

Unsupervised Learning: Clustering

Justin Post

Unsupervised Learning

Unsupervised learning - no *response* variable

Aside from PCA, *clustering* is widely used

- Clustering - Find subgroups in the data
    - Groups should have members that are “similar” to one another
    - # of subgroups subjective most of the time
- Two major methods
    - K-Means clustering
    - Hierarchical clustering

Visual of K Means Clustering (Clusters not Always the Same!)

Iris k-means clustering

X Variable

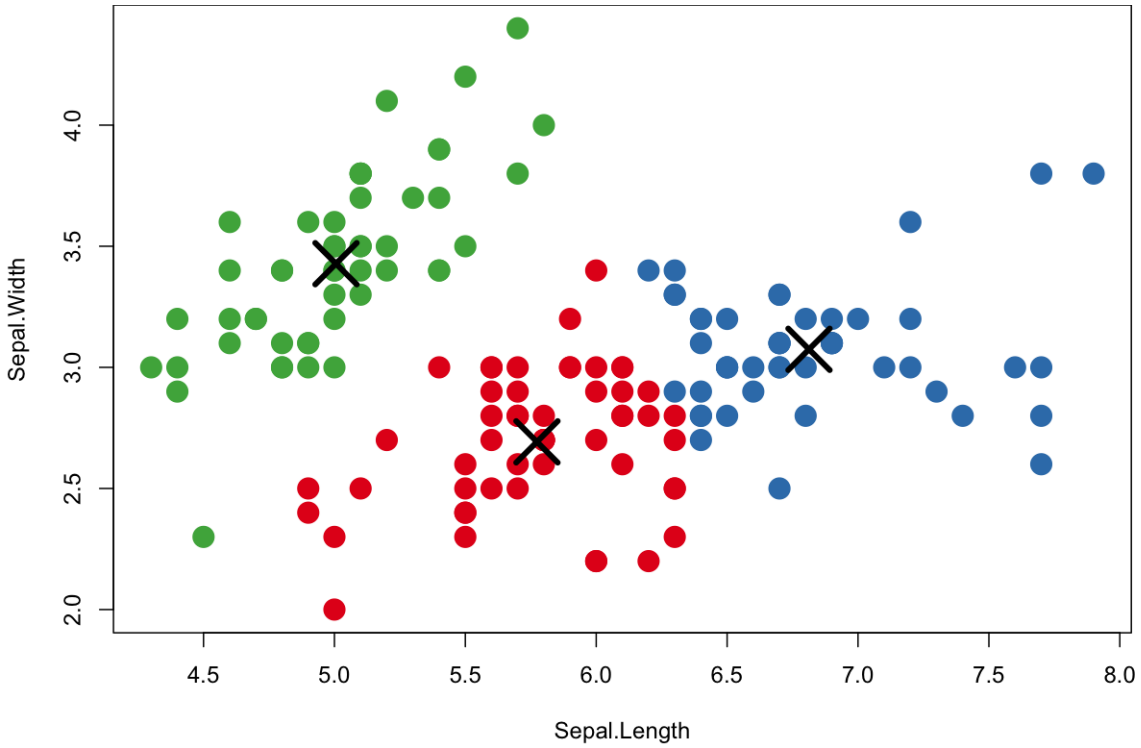
Sepal.Length

Y Variable

Sepal.Width

Cluster count

3



Details of K Means Clustering

Look at *within-cluster variation*

- For kth cluster, sum all pairwise squared Euclidean distances between the observations in the kth cluster. Divide by # of observations.

$$\frac{1}{\#\text{ of obs in cluster}} \sum_{\substack{\text{all pairs of obs} \\ \text{in cluster},i_1,i_2}} \sum_{\substack{\text{all variables,} \\ j=1}}^{p} (x_{i_1,j} - x_{i_2,j})^2$$

- Essentially, average of distances between all pairs of points!
- Find for each cluster, sum all and minimize

Details of K Means Clustering

- Difficult problem to find optimal clusters
- Common algorithm used finds *local min* for the function

Algorithm:

1. Randomly assign a number, from 1 to K, to each of the observations. (These serve as initial cluster assignments for the observations.)
2. Iterate until the cluster assignments stop changing:
    - For each of the K clusters, compute the mean for each variable across the values in that cluster (called a centroid).
    - Assign each observation to the cluster whose centroid is closest (where closest is defined using Euclidean distance).

Details of K Means Clustering

Since (probably) local min found each time

- leads to different clusters when run repeatedly
- Run algorithm many times and take the one with overall smallest objective function (sum of average cluster distances)

Visual of K Means Clustering (Initial choice done *intelligently*)

Iris k-means clustering

X Variable

Sepal.Length

Y Variable

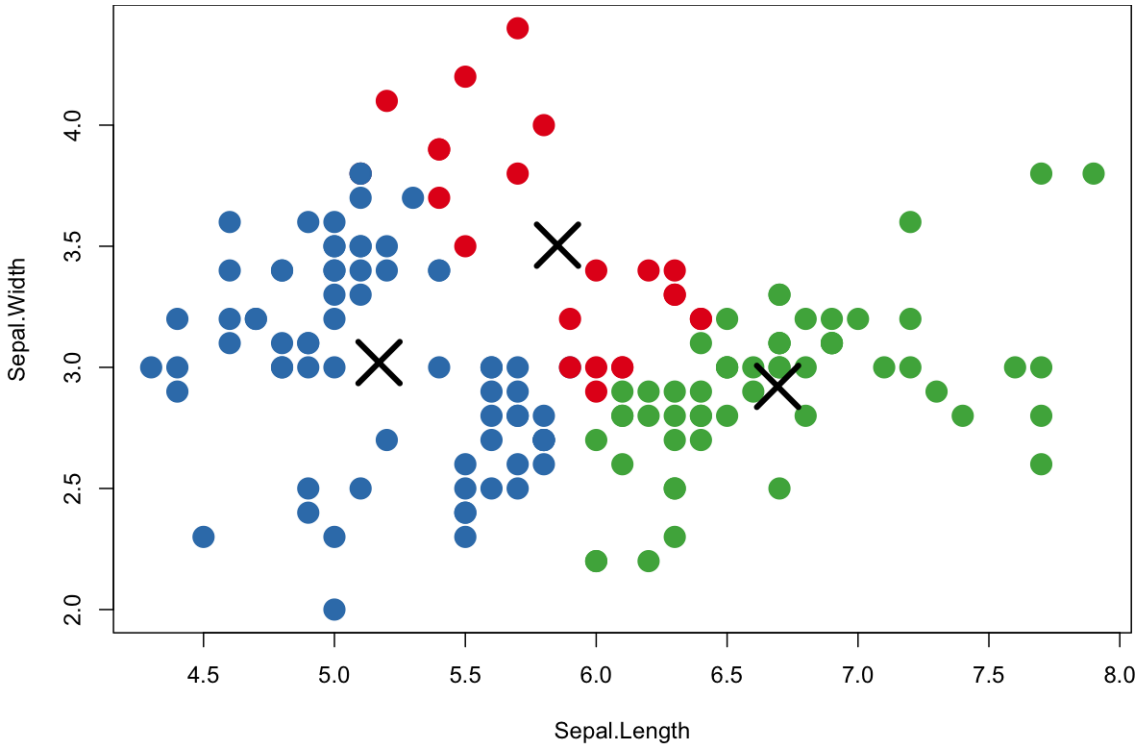
Sepal.Width

Cluster count

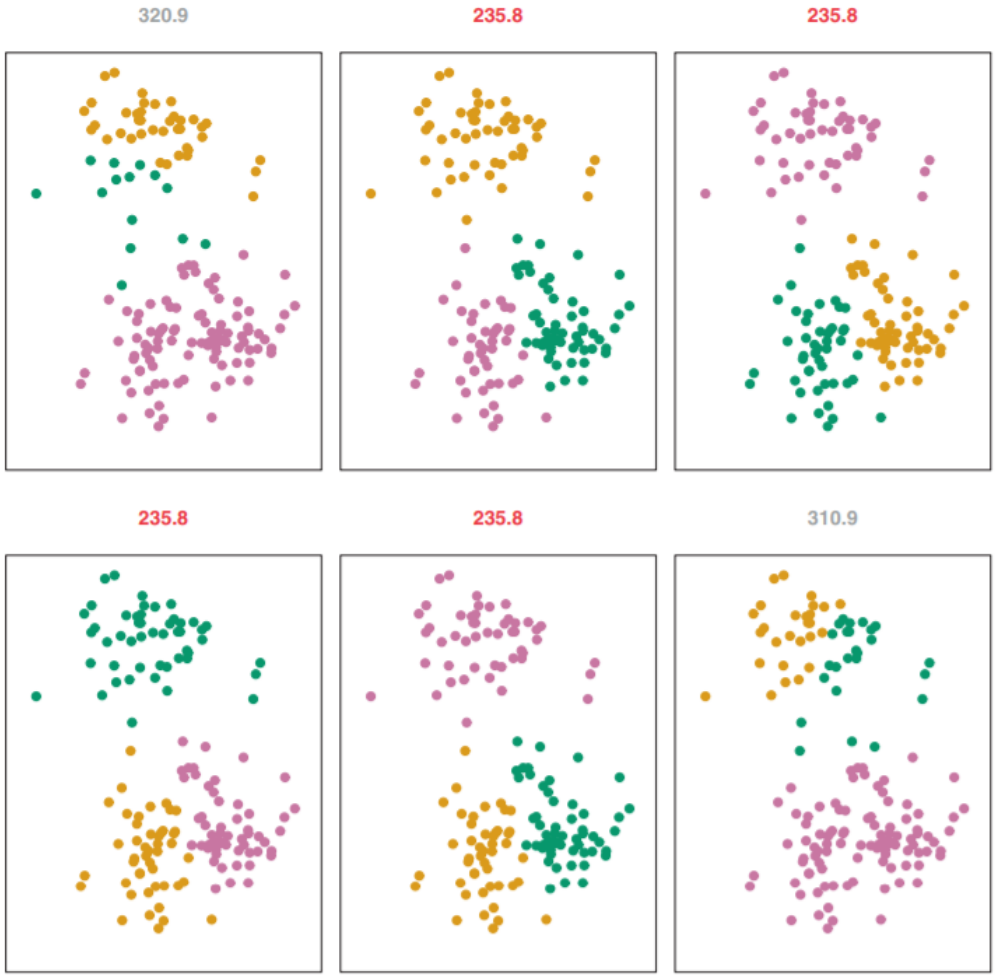
3

# of Iterations of Algorithm

1



K Means Clustering, Multiple Starting Points

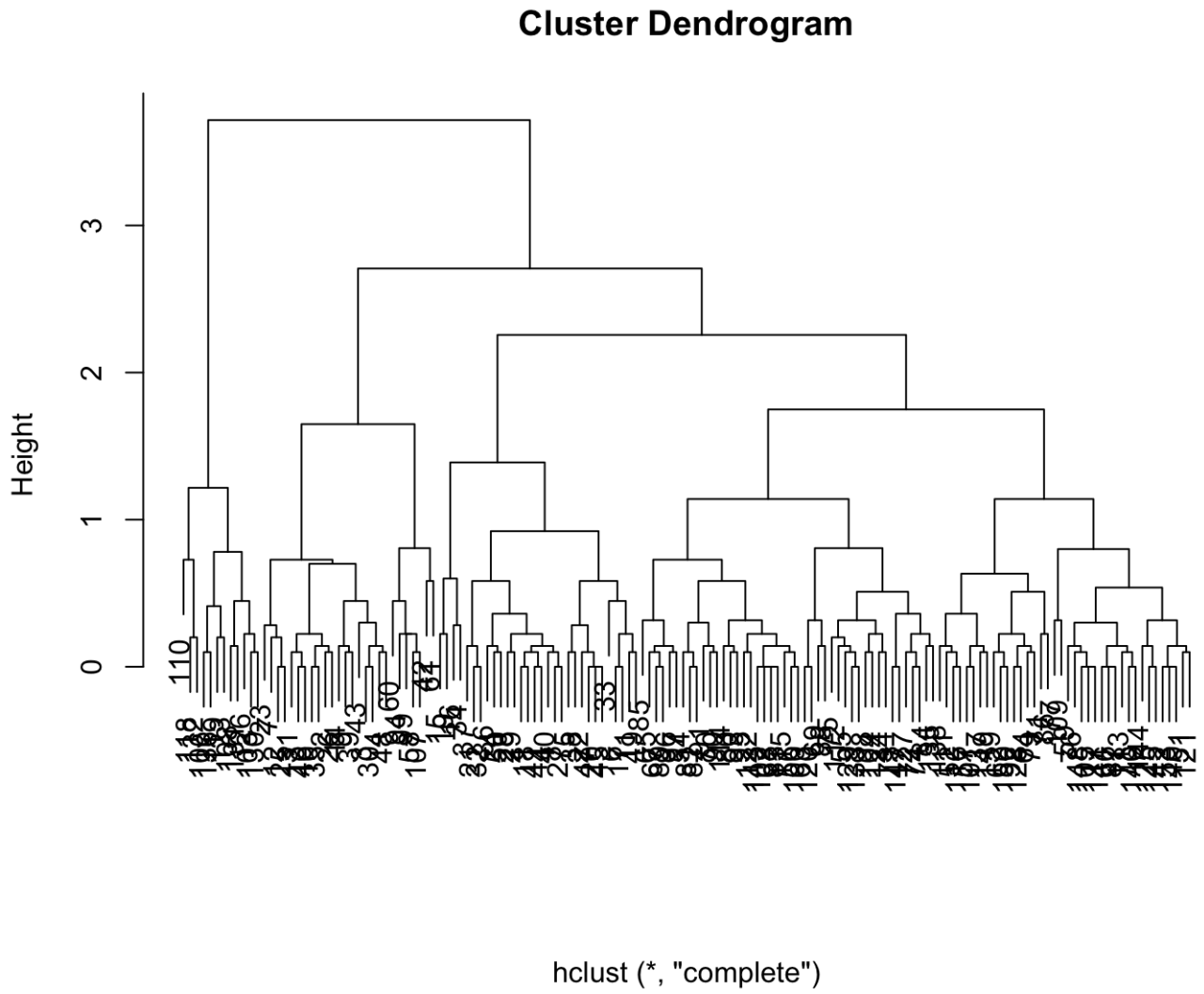


Hierarchical Clustering

Bottom up clustering can be used.

- No need to specify # of clusters
- Start with all observations in own cluster
- Join ‘closest’ observations (lessing clusters each time), until 1 cluster.
- Can be visualized with a dendogram

```
hierClust <- hclust(dist(data.frame(iris$Sepal.Length, iris$Sepal.Width)))
plot(hierClust, xlab = "")
```



Hierarchical Clustering

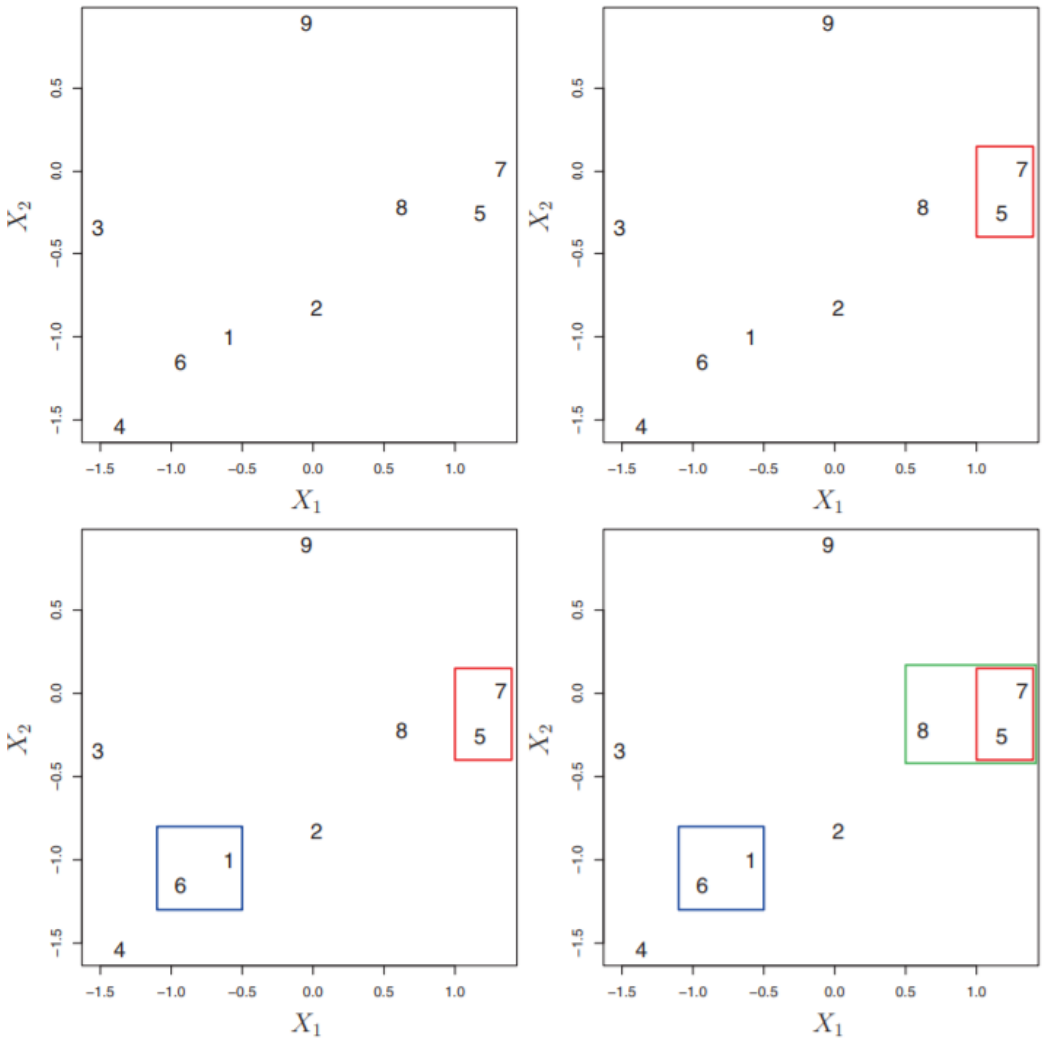
Clusters are nested in some sense (hence hierarchical)

- Different methods can be used to *join* observations

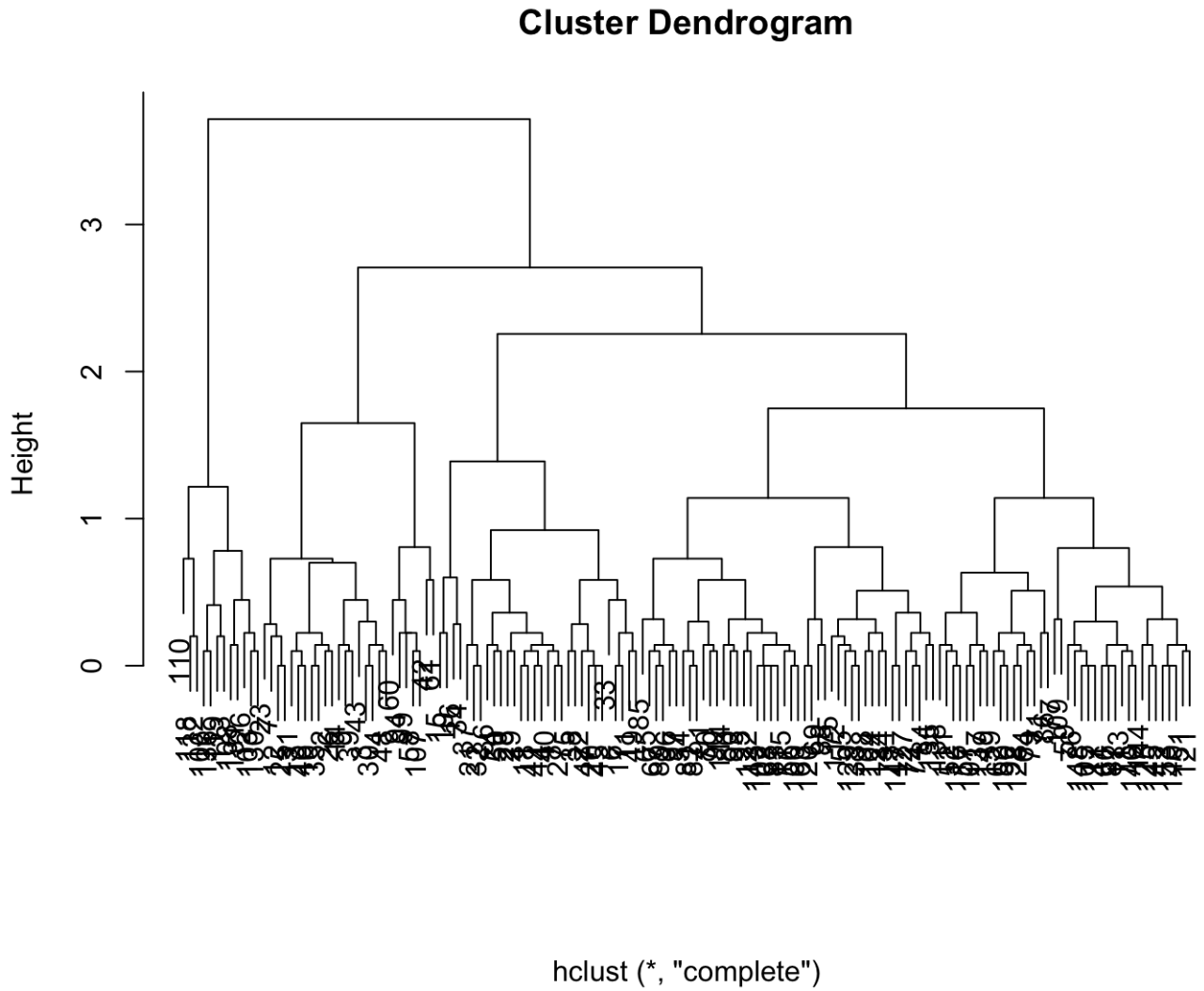
Linkage	*Description*
Complete	Maximal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities. Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.
Average	Mean intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

TABLE 10.2. *A summary of the four most commonly-used types of linkage in hierarchical clustering.*

Complete Linkage Example



Determine Cluster Membership Using ‘Horizontal Line’



Clustering Recap

KMeans

- `knn()` function in R
- Must specify # of clusters
- Usually run multiple starting points (`nstart` option in `knn`)

Hierarchical

- `hclust()` function in R
- Specify distance matrix as main argument
- Specify *dissimilarity* measure (many options: `ward.D`, `ward.D2`, `single`, `complete`, `average`, `mcquitty`, `median` or `centroid`
- Dendrogram for visualization