Hierarchical Clustering

Generally agglomerative ("bottom-up"): each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy

- Put each data point in its own cluster
- Identify the closest (using linkage criteria) two clusters and combine them into one cluster
- Repeat the above step till all the data points are in a single cluster

Alternatively divisive ("top-down"): all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy

Linkage

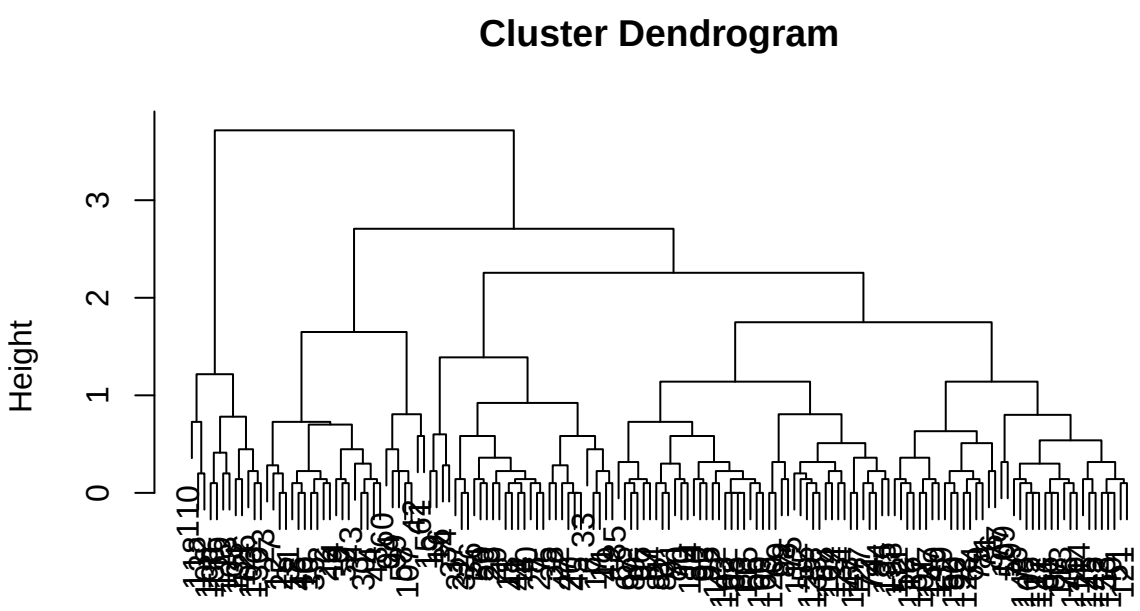
Determines the distance between sets of observations as a function of the pairwise distances between observations
Complete/single/average = $max/min/mean\{d(a,b) : a \in A, b \in B\}$

Iris Example

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

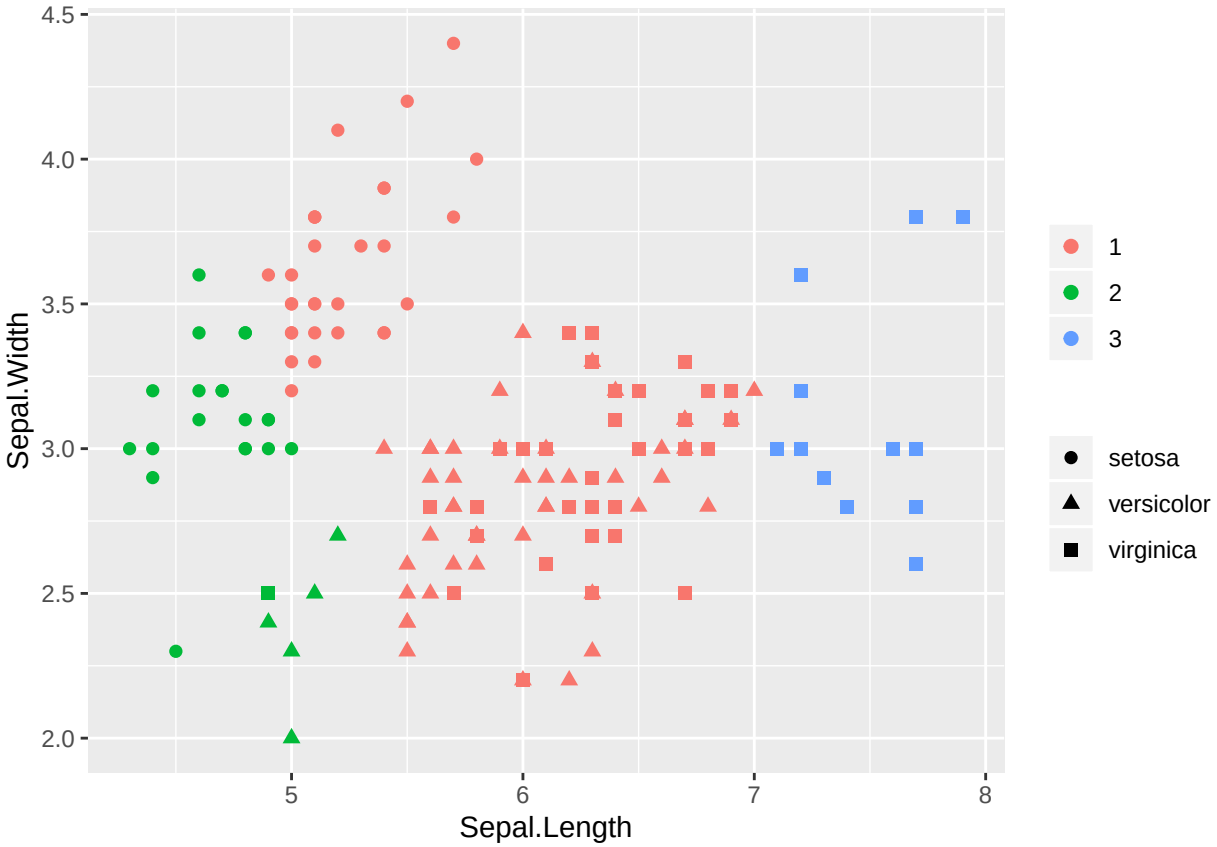
```
iris_dist = dist(iris[,1:2]) #defaults to euclidean distance
clusters = hclust(iris_dist) #defaults to complete linkage
plot(clusters)
```




```
iris$cluster = cutree(clusters,k=3)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.2
```

```
ggplot(iris, aes(Sepal.Length, Sepal.Width, color=as.factor(cluster), shape=Species)) + geom_point(size=
```



Here we knew there were 3 species of flower, we could alternatively cut at a height by specifying h in cutree. With complete linkage this tells us the maximum distance to all other members in a cluster is less than h