**Practical No.4**

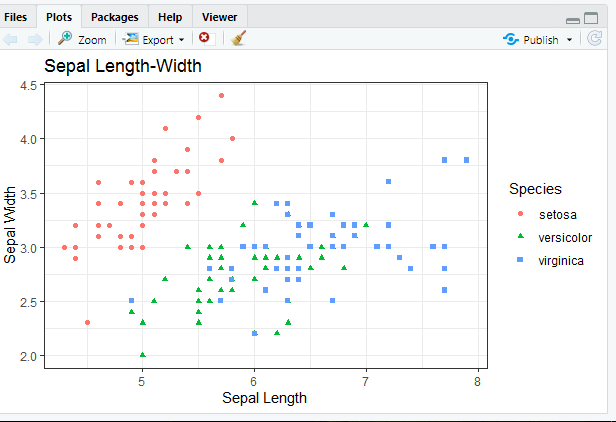
**Aim: Practical of Clustering**

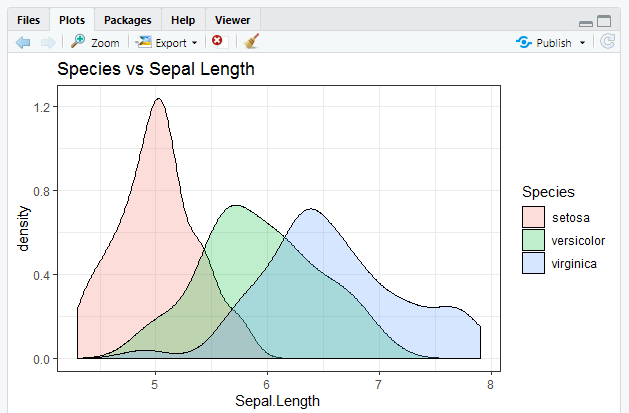
> library(ggplot2)

> install.packages("ggplot2")

> scatter<-ggplot(data=iris,aes(x=Sepal.Length,y=Sepal.Width))

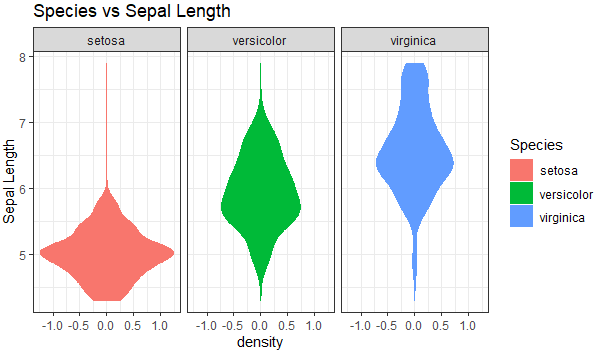
> scatter+geom\_point(aes(color=Species,shape=Species))+theme\_bw()+ xlab("Sepal Length")+ ylab("Sepal Width")+ggtitle("Sepal Length-Width")

  
> ggplot(data=iris,aes(Sepal.Length,fill=Species))+ theme\_bw()+geom\_density(alpha=0.25)+labs(x="Sepal.Length",title="Species vs Sepal Length")



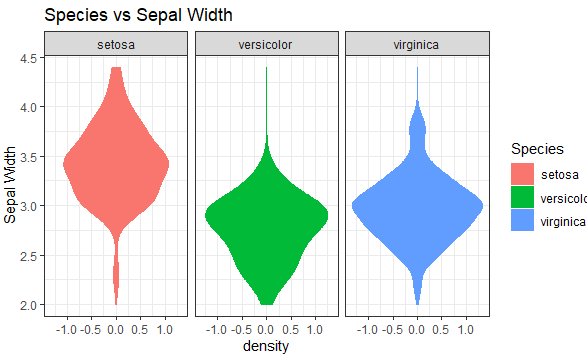
> vol<-ggplot(data = iris,aes(x=Sepal.Length))

> vol+stat\_density(aes(ymax=..density..,ymin=-..density..,fill=Species,color=Species),geom = "ribbon",poosition="identity")+facet\_grid(.~Species)+coord\_flip()+theme\_bw()+labs(x="Sepal Length",title = "Species vs Sepal Length")



> vol<-ggplot(data = iris,aes(x=Sepal.Width))

> vol+stat\_density(aes(ymax=..density..,ymin=-..density..,fill=Species,color=Species),geom = "ribbon",poosition="identity")+facet\_grid(.~Species)+coord\_flip()+theme\_bw()+labs(x="Sepal Width",title = "Species vs Sepal Width")



**#clustering method1**

> irisData<-iris[,1:4]

> totalwSS<-c()

**#Method1 kmeans clustering for 15 times in a loop**

> for(i in 1:15)

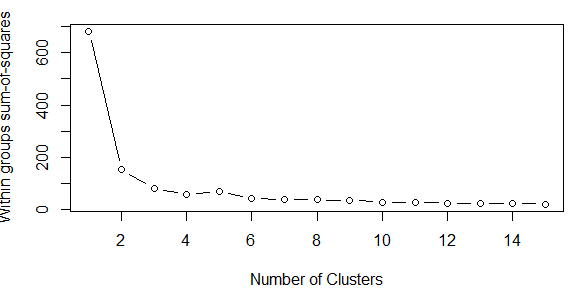
+ {clusterIRIS<-kmeans(irisData,centers=i)

+ totalwSS[i]<-clusterIRIS$tot.withinss

+ }

**# Use plot function to plot values of tot\_wss against no-of-clusters**

> plot(x=1:15,y=totalwSS,type = "b",xlab="Number of Clusters",ylab = "Within groups sum-of-squares")



**#Method2 Using NbClust-Uses huge no of cluster suitability measuring critera**

> install.packages("NbClust")

> library(NbClust)

**#Set margins as:c(bottom,left,top,right)**

par(mar=c(2,2,2,2))

> nb<-NbClust(irisData,method = "kmeans")

\*\*\* : The Hubert index is a graphical method of determining the number of clusters.

In the plot of Hubert index, we seek a significant knee that corresponds to a

significant increase of the value of the measure i.e the significant peak in Hubert

index second differences plot.

\*\*\* : The D index is a graphical method of determining the number of clusters.

In the plot of D index, we seek a significant knee (the significant peak in Dindex

second differences plot) that corresponds to a significant increase of the value of

the measure.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Among all indices:

\* 11 proposed 2 as the best number of clusters

\* 11 proposed 3 as the best number of clusters

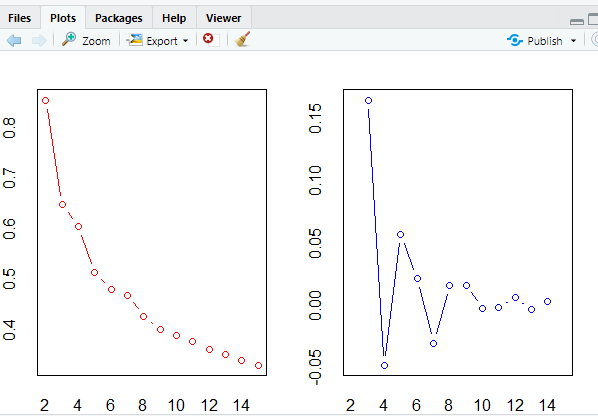
\* 1 proposed 8 as the best number of clusters

\* 1 proposed 12 as the best number of clusters

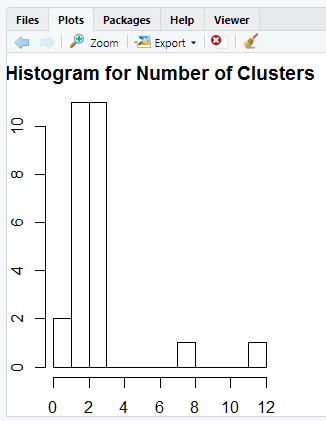
\*\*\*\*\* Conclusion \*\*\*\*\*

\* According to the majority rule, the best number of clusters is 2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



> hist(nb$Best.nc[1,],breaks=15,main="Histogram for Number of Clusters")



**#Method3 Clustering Data with Silhouette plot**

> library(cluster)

> cl<-kmeans(iris[,-5],2)

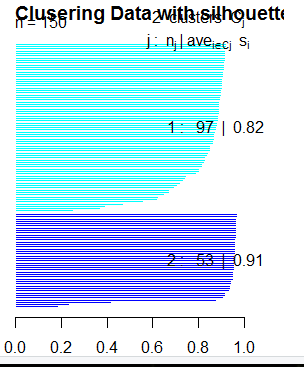
**#Compute and returns the distance matrix computed by using euclidean distance measure to # the distance between rows of a data matrix**

> dis<-dist(iris[,-5])^2

**# Get silhouette coefficient**

> sil=silhouette(cl$cluster,dis)

> plot(sil,main = "Clusering Data with silhouette plot using 2 Clusters",col=c("cyan","blue"))



**#For silhouette()**

> library(cluster)

> cl<-kmeans(iris[,-5],8)

**# the distance between the rows of a data matrix**

> dis<-dist(iris[,-5])^2

**#Get silhouette coefficient**

> sil=silhouette(cl$cluster,dis)

> plot(sil,main = "Clustering Data with Silhouette plot using 8 Clusters",col = c("cyan","blue","orange","yellow","red","gray","green","maroon"))

