```
#extracting samples having class setosa and training classifier on first half of data.
setos<-iris[iris[,5]=="setosa",]
mean_setosa=as.matrix(colMeans(setos[1:25,1:4]))
cov_setosa=as.matrix(cov(setos[1:25,1:4]))

#extracting samples having class versicolor and training classifier on first half of data.
versicol<-iris[iris[,5]=="versicolor",]
mean_versicol=as.matrix(colMeans(versicol[1:25,1:4]))
cov_versicol=as.matrix(cov(versicol[1:25,1:4]))

#extracting samples having class virginica and training classifier on first half of data.
virgin=iris[iris[,5]=="virginica",]
mean_virginica=as.matrix(colMeans(virgin[1:25,1:4]))
cov_virginica=as.matrix(cov(virgin[1:25,1:4]))



i1<-data.frame()
#extracting the test set form second half of the data.
i1<-rbind(setos[26:50,],versicol[26:50,],virgin[26:50,])
i1$Species<-factor(i1$Species)

#converting mean to a vector form.
mean_setosa<-as.vector(mean_setosa)
mean_versicol<-as.vector(mean_versicol)
mean_virginica<-as.vector(mean_virginica)
k=1
v<-NULL
count=0
v2<-NULL

for(i in 1:(length(rownames(i1))))
{

  d<-i1[i,1:4]
  d<-as.vector(unlist(d))
  #calcualting multivariate density for setosa
  a=dmvnorm(d,mu=mean_setosa,Sigma = cov_setosa)
  #calculating multivariate density for versicolor
  b=dmvnorm(d,mu=mean_versicol,Sigma = cov_versicol)
  #calculating multivariate
  c=dmvnorm(d,mu=mean_virginica,Sigma = cov_virginica)

  vec<-c(a,b,c)
  #finding the maximum of the calculated densities
  v[k]<-which.max(vec)
  if(v[k]==1)
    v[k]=toString("setosa")
  if(v[k]==2)
    v[k]=toString("versicolor")
  if(v[k]==3)
    v[k]=toString("virginica")

  if(v[k]==i1[i,5])
    {
    v[k]="green"
    #calculating the ones that are correctly classified and maintaining a count.
    count=count+1
  }
  else
    v[k]="red"


  k=k+1
}
print(count/75)
```