**Practical no 3**

**Aim:Practical of Principal Component Analysis(PCA**)

**#Taking the numeric part of IRIS data**> data\_iris<-iris[1:4]

**#calculating the covariance matrix**

> Cov\_data<-cov(data\_iris)

**#find out the eigen vectors and eigen values using the covariance matrix**

> Eigen\_data<-eigen(Cov\_data)

**#Using the inbuilt function PCA calculation**

> PCA\_data<-princomp(data\_iris,cor = "False")

**#Lets now compare the output variable**

> Eigen\_data$values

[1] 4.22824171 0.24267075 0.07820950 0.02383509

> PCA\_data$sdev^2

Comp.1 Comp.2 Comp.3 Comp.4

4.20005343 0.24105294 0.07768810 0.02367619

**#there is a slight difference due to squaring in PCA\_data but the outputs are more or less similar.we can also compare the eigen vectors of both models.**

> PCA\_data$loadings[,1:4]

Comp.1 Comp.2 Comp.3 Comp.4

Sepal.Length 0.36138659 0.65658877 0.58202985 0.3154872

Sepal.Width -0.08452251 0.73016143 -0.59791083 -0.3197231

Petal.Length 0.85667061 -0.17337266 -0.07623608 -0.4798390

Petal.Width 0.35828920 -0.07548102 -0.54583143 0.7536574

|  |
| --- |
| > Eigen\_data$vectors  [,1] [,2] [,3] [,4]  [1,] 0.36138659 -0.65658877 -0.58202985 0.3154872  [2,] -0.08452251 -0.73016143 0.59791083 -0.3197231  [3,] 0.85667061 0.17337266 0.07623608 -0.4798390  [4,] 0.35828920 0.07548102 0.54583143 0.7536574  **#To know the importance of the first component, we can view the summary of the model**  > summary(PCA\_data)  Importance of components:  Comp.1 Comp.2 Comp.3 Comp.4  Standard deviation 2.0494032 0.49097143 0.27872586 0.153870700  Proportion of Variance 0.9246187 0.05306648 0.01710261 0.005212184  Cumulative Proportion 0.9246187 0.97768521 0.99478782 1.000000000  > biplot(PCA\_data)    > screeplot(PCA\_data,type = "lines")    **#we will calculate the difference in accuracy between these two models**  **#select the first principal component for the second model**  > model2=PCA\_data$loadings[,1]  **#for the second model,we need to calculate scores by multiplying our loadings with the data**  > model2\_scores<-as.matrix(data\_iris)%\*%model2  **#loading libraries for naiveBayes model**  > library(class)  > install.packages("e1071")  > library(e1071)  **#fitting the first model over the entire data**  > mod1<-naiveBayes(iris[,1:4],iris[,5])  **#fitting the second model using the first principal component**  > mod2<-naiveBayes(model2\_scores,iris[,5])  **#Accuracy for the first model**  > table(predict(mod1,iris[,1:4]),iris[,5])    setosa versicolor virginica  setosa 50 0 0  versicolor 0 47 3  virginica 0 3 47  **#Accuracy of the second model**  > table(predict(mod2,model2\_scores),iris[,5])    setosa versicolor virginica  setosa 50 0 0  versicolor 0 46 5  virginica 0 4 45 |
|  |
| |  | | --- | |  | |