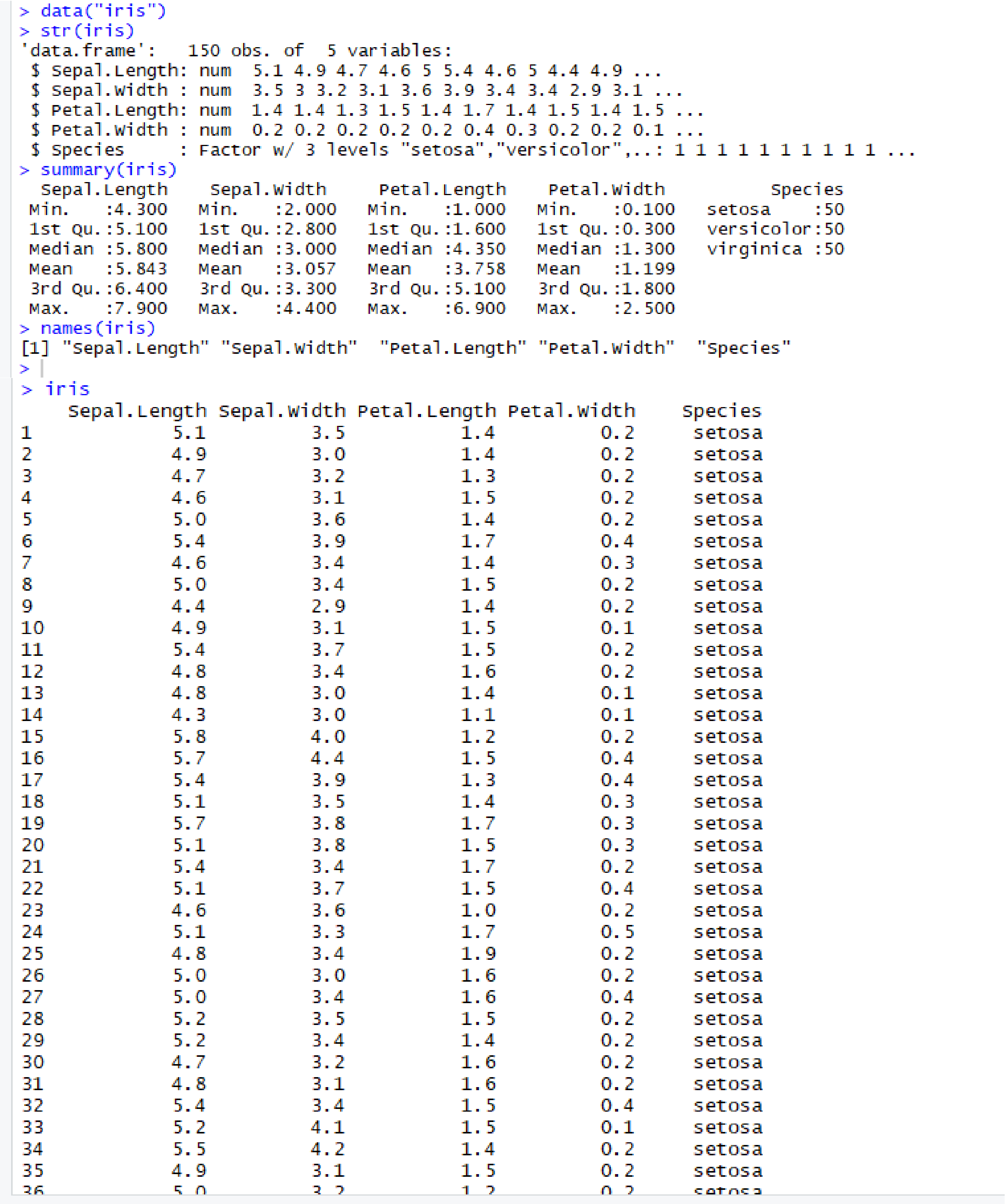
PRACTICAL 3: Practical of Principal Component Analysis

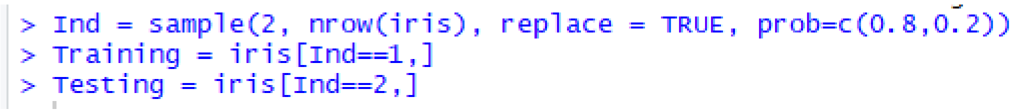
youtube video tutorial : https://www.youtube.com/watch?v=OowGKNgdowA

1) **Iris Data Set**



```
> data("iris")
> str(iris)
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species  
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50  
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50  
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                  
> names(iris)
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"     
> 
> 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
7            4.6         3.4          1.4         0.3     setosa
8            5.0         3.4          1.5         0.2     setosa
9            4.4         2.9          1.4         0.2     setosa
10           4.9         3.1          1.5         0.1     setosa
11           5.4         3.7          1.5         0.2     setosa
12           4.8         3.4          1.6         0.2     setosa
13           4.8         3.0          1.4         0.1     setosa
14           4.3         3.0          1.1         0.1     setosa
15           5.8         4.0          1.2         0.2     setosa
16           5.7         4.4          1.5         0.4     setosa
17           5.4         3.9          1.3         0.4     setosa
18           5.1         3.5          1.4         0.3     setosa
19           5.7         3.8          1.7         0.3     setosa
20           5.1         3.8          1.5         0.3     setosa
21           5.4         3.4          1.7         0.2     setosa
22           5.1         3.7          1.5         0.4     setosa
23           4.6         3.6          1.0         0.2     setosa
24           5.1         3.3          1.7         0.5     setosa
25           4.8         3.4          1.9         0.2     setosa
26           5.0         3.0          1.6         0.2     setosa
27           5.0         3.4          1.6         0.4     setosa
28           5.2         3.5          1.5         0.2     setosa
29           5.2         3.4          1.4         0.2     setosa
30           4.7         3.2          1.6         0.2     setosa
31           4.8         3.1          1.6         0.2     setosa
32           5.4         3.4          1.5         0.4     setosa
33           5.2         4.1          1.5         0.1     setosa
34           5.5         4.2          1.4         0.2     setosa
35           4.9         3.1          1.5         0.2     setosa
36           5.0         3.2          1.2         0.2     setosa
```

2) **partition Data**

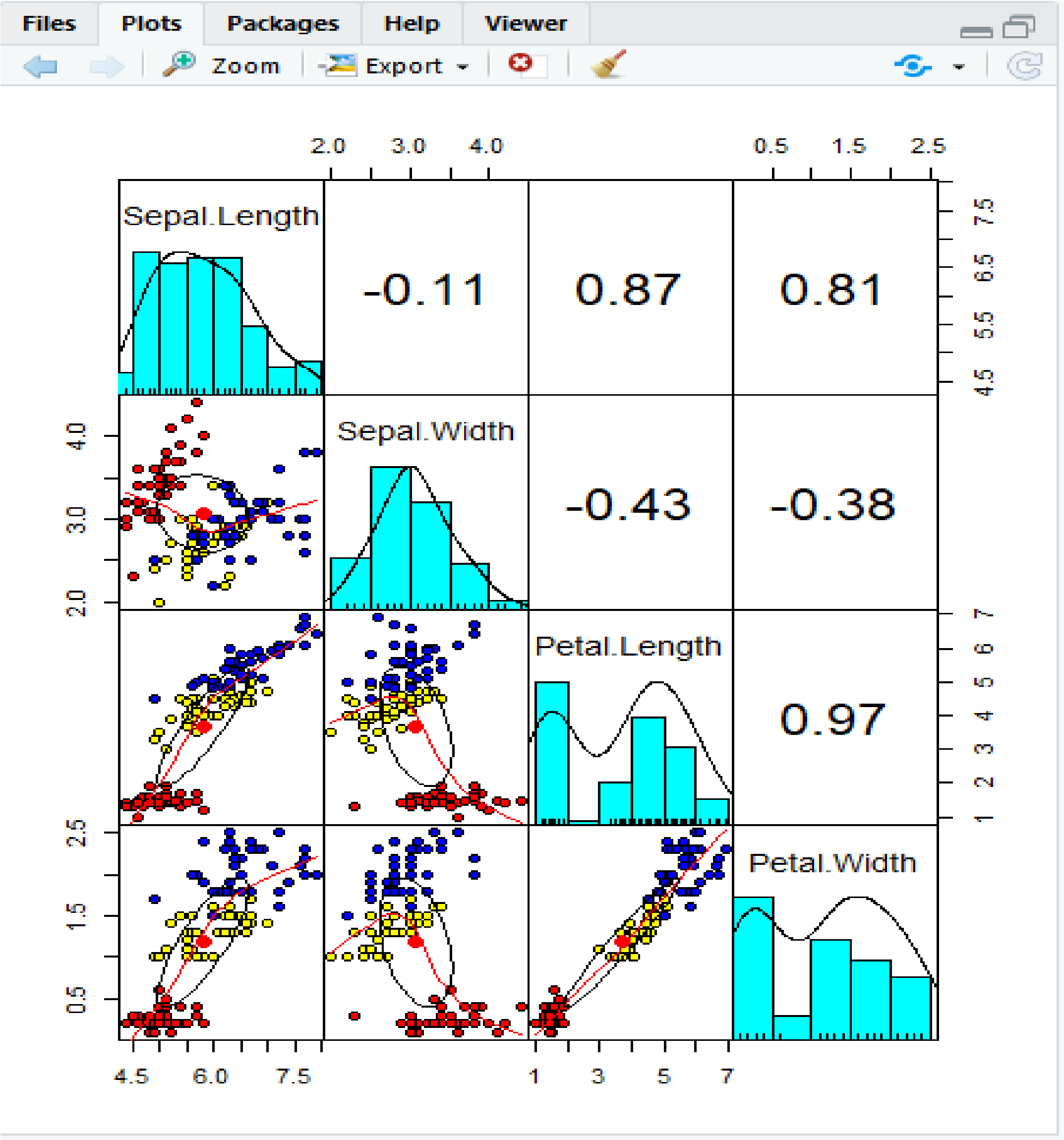


```
> Ind = sample(2, nrow(iris), replace = TRUE, prob=c(0.8,0.2))
> Training = iris[Ind==1,]
> Testing = iris[Ind==2,]
```

3) **plot the data (scatter plot and correlations)**

```
install.packages("psych")
library(psych)
pairs.panels(training[1:4],gap=0,bg=c("yellow","red","blue")[training$
Species],pch = 21)
```

Output:



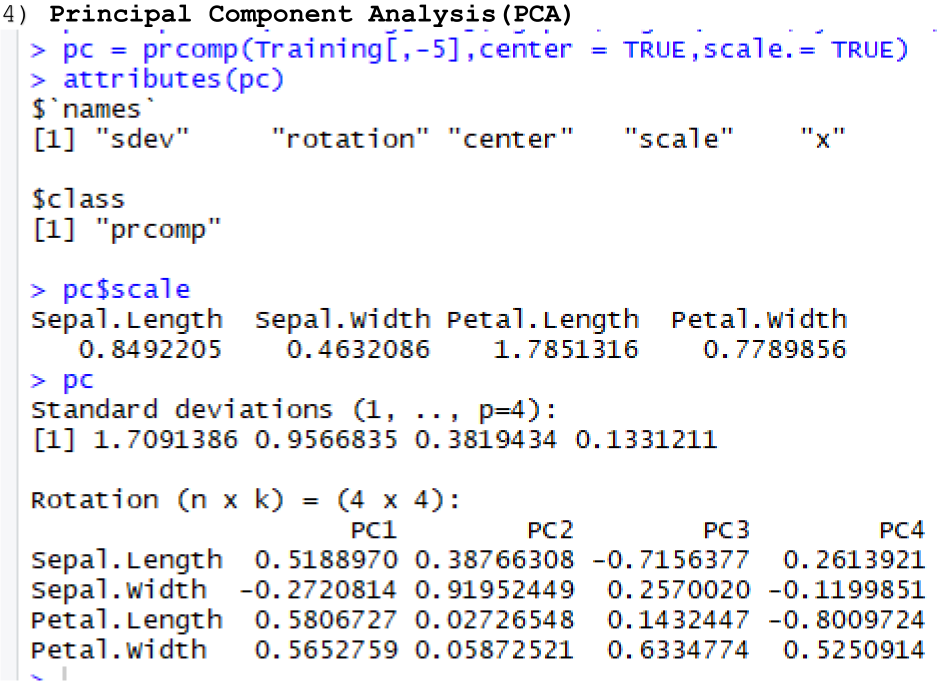
Analysis:

1. The lower triangle of the output gives a scatter plot whereas upper triangle gives correlation coefficient (used

to measure the strength of the relationship between two variables).

2. Correlation is highest between Petal.Length and Petal.Width i.e. 0.97. The two variables are positively correlated.
3. The lowest correlation is between Sepal.Length and Sepal.Width i.e -0.11
4. High correlations among independent variables gives rise to multicollinearity problems. Because of this, predictions are not very accurate. Hence, we use PCA (Principal Component Analysis).

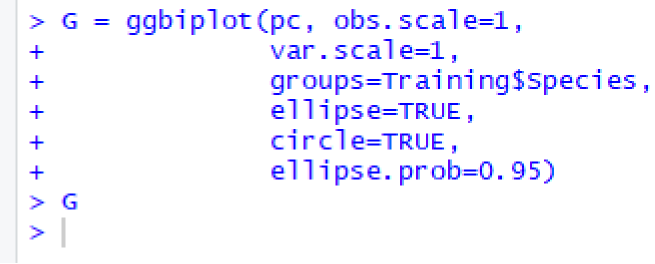
4) **Principal Component Analysis(PCA)**



```
> pc = prcomp(Training[,-5],center = TRUE,scale.= TRUE)
> attributes(pc)
$`names`
[1] "sdev"     "rotation" "center"   "scale"    "x"

$class
[1] "prcomp"

> pc$scale
Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
   0.8492205    0.4632086    1.7851316    0.7789856 
> pc
Standard deviations (1, .., p=4):
[1] 1.7091386 0.9566835 0.3819434 0.1331211

Rotation (n x k) = (4 x 4):
                    PC1        PC2        PC3        PC4
Sepal.Length  0.5188970 0.38766308 -0.7156377  0.2613921
Sepal.Width  -0.2720814 0.91952449  0.2570020 -0.1199851
Petal.Length  0.5806727 0.02726548  0.1432447 -0.8009724
Petal.Width   0.5652759 0.05872521  0.6334774  0.5250914
```

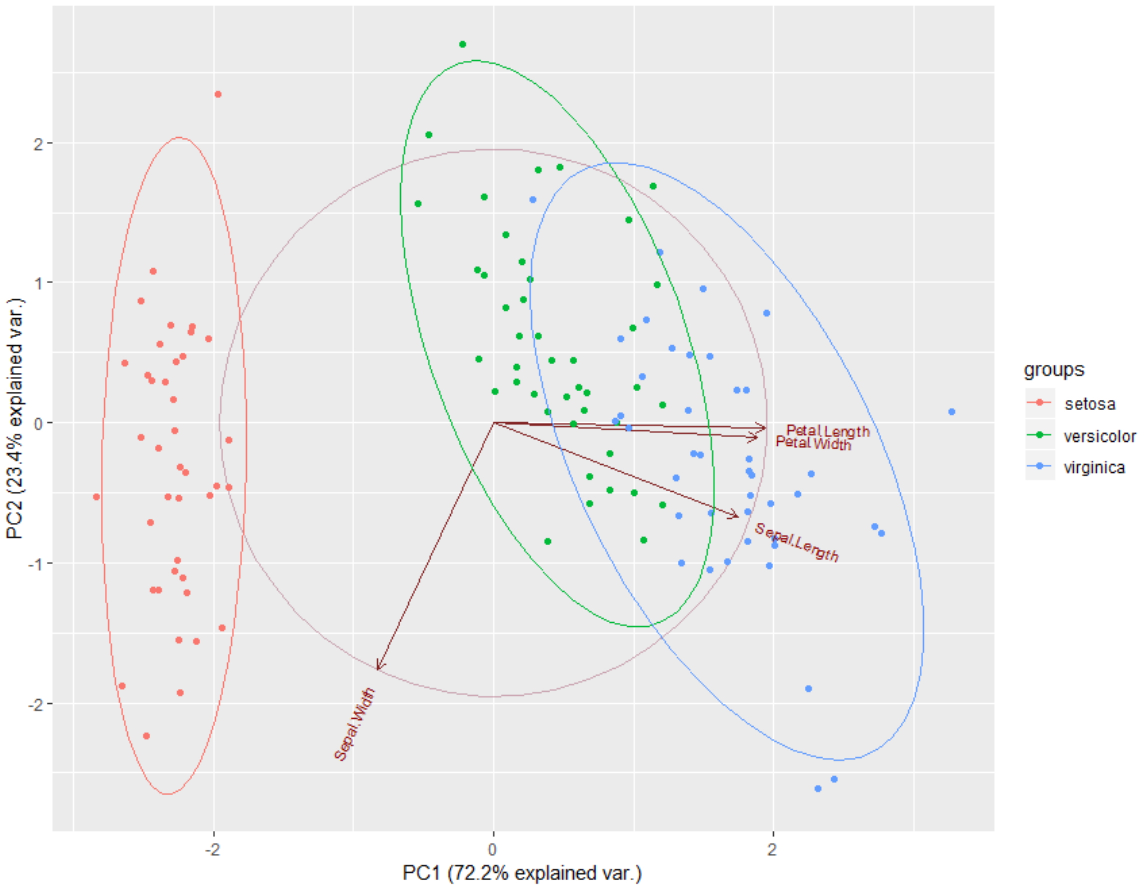
5) **ggbiplot**

```
install.packages("devtools")
library(devtools)
install_github("vqv/ggbiplot")
library(ggbiplot)
```



```
> G = ggbiplot(pc, obs.scale=1,
+              var.scale=1,
+              groups=Training$Species,
+              ellipse=TRUE,
+              circle=TRUE,
+              ellipse.prob=0.95)
> G
> 
```

Output:



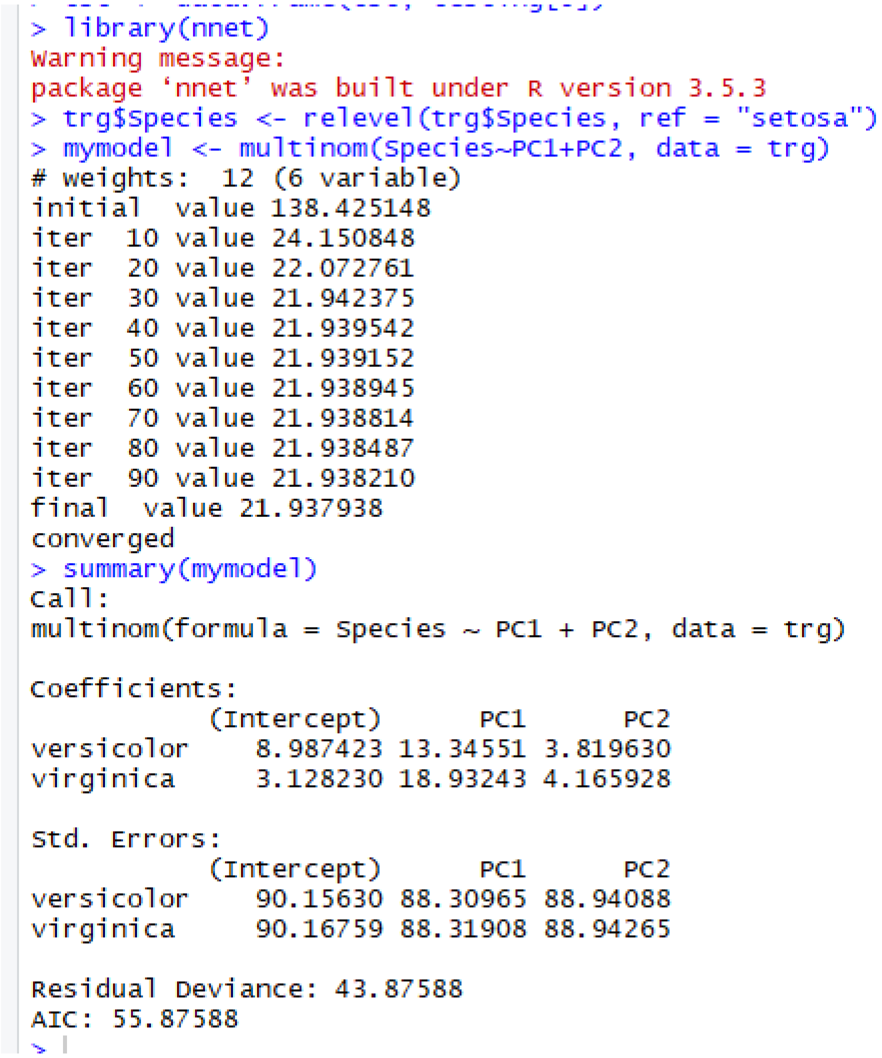
Analysis:

1. The first principal component PC1 explains 72.2% variability. Second principal component PC2 explains 23.4% variability.
2. Species are color-coded. Red-setosa, Green- versicolor, blue- virginica. Every colored ellipse covers 95% of the data points. This is defined by the ellipse.prob = 0.95
3. All the 4 variables are represented by 4 arrows. Petal.Length and Petal.Width are close to each other hence correlation coefficient between them is the highest.
4. Sepal.Length is also highly correlated with Petal.Length and Petal.Width.
5. Sepal.Width is very far away from other three variables hence its not highly correlated with any other variable.
6. Sepal.Width is on the negative side of PC1 hence correlation between PC1 and Sepal.Width is negative and correlation between other three variables and PC1 is positive.
7. The same analysis is for PC2.

6) **Prediction with Principal Components**

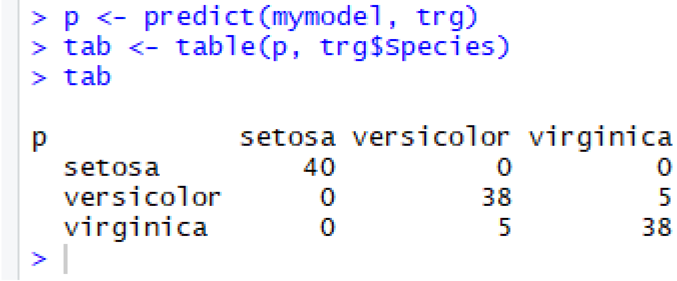
```
> trg <- predict(pc, training)
> trg <- data.frame(trg, training[5])
> tst <- predict(pc, testing)
> tst <- data.frame(tst, testing[5])
```

7) **Multinomial Logistic regression with first 2 PCs**



```
> library(nnet)
Warning message:
package ‘nnet’ was built under R version 3.5.3
> trg$Species <- relevel(trg$Species, ref = "setosa")
> mymodel <- multinom(Species~PC1+PC2, data = trg)
# weights:  12 (6 variable)
initial  value 138.425148
iter  10 value 24.150848
iter  20 value 22.072761
iter  30 value 21.942375
iter  40 value 21.939542
iter  50 value 21.939152
iter  60 value 21.938945
iter  70 value 21.938814
iter  80 value 21.938487
iter  90 value 21.938210
final  value 21.937938
converged
> summary(mymodel)
Call:
multinom(formula = Species ~ PC1 + PC2, data = trg)

Coefficients:
           (Intercept)      PC1      PC2
versicolor    8.987423 13.34551 3.819630
virginica     3.128230 18.93243 4.165928

Std. Errors:
           (Intercept)      PC1      PC2
versicolor    90.15630 88.30965 88.94088
virginica     90.16759 88.31908 88.94265

Residual Deviance: 43.87588
AIC: 55.87588
>
```

8) **Confusion Matrix and Misclassification Error – training**

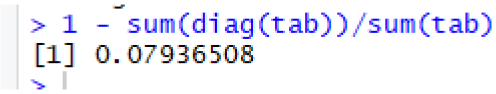


```
> p <- predict(mymodel, trg)
> tab <- table(p, trg$Species)
> tab

p            setosa versicolor virginica
  setosa         40          0         0
  versicolor      0         38         5
  virginica       0          5        38
>
```

Analysis:

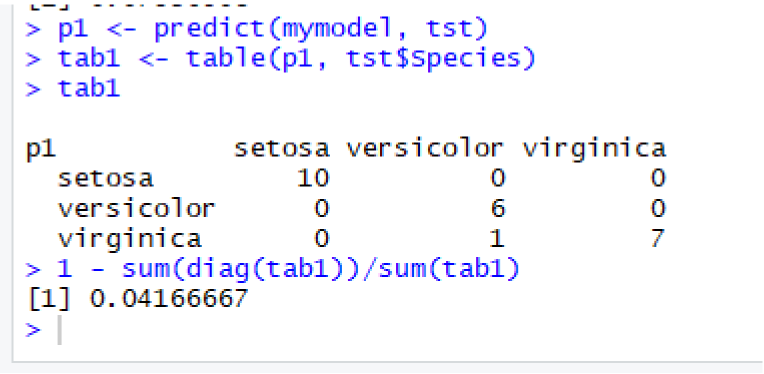
1. There are 40 correct classifications for 1st category – setosa.
2. There are 38 correct classifications for 2nd category and 5 misclassifications where actually they belong to versicolor but model predicts them to belong to virginica.
3. There are 38 correct classifications for 3rd category and 5 misclassifications where actually they belong to virginica but model predicts them to belong to versicolor.

To calculate misclassification error



```
> 1 - sum(diag(tab))/sum(tab)
[1] 0.07936508
>
```

9) **Confusion Matrix and Misclassification Error – testing data**



```
> p1 <- predict(mymodel, tst)
> tab1 <- table(p1, tst$Species)
> tab1

p1           setosa versicolor virginica
  setosa         10          0         0
  versicolor      0          6         0
  virginica       0          1         7
> 1 - sum(diag(tab1))/sum(tab1)
[1] 0.04166667
>
```

Analysis:

1. There are 10 correct classifications for 1st category – setosa in testing data.
2. There are 6 correct classifications for 2nd category and 1 misclassification where actually they belong to versicolor but model predicts them to belong to virginica. This shows that misclassification is reduced because in training data there were 5 misclassifications.
3. There are 7 correct classifications for 3rd category and 0 misclassifications.
4. The misclassification error is also reduced from 0.07936508 to 0.04166667.