PCA.R

#### ***Training***

#### ***Sat Mar 03 08:47:30 2018***

## Normalization  
# (x-xmax)/(xmax-xmin)  
### Create the PCA model  
iris.pc<- prcomp(iris[,c(-5,-6)])  
### Explore the names  
names(iris.pc)

## [1] "sdev" "rotation" "center" "scale" "x"

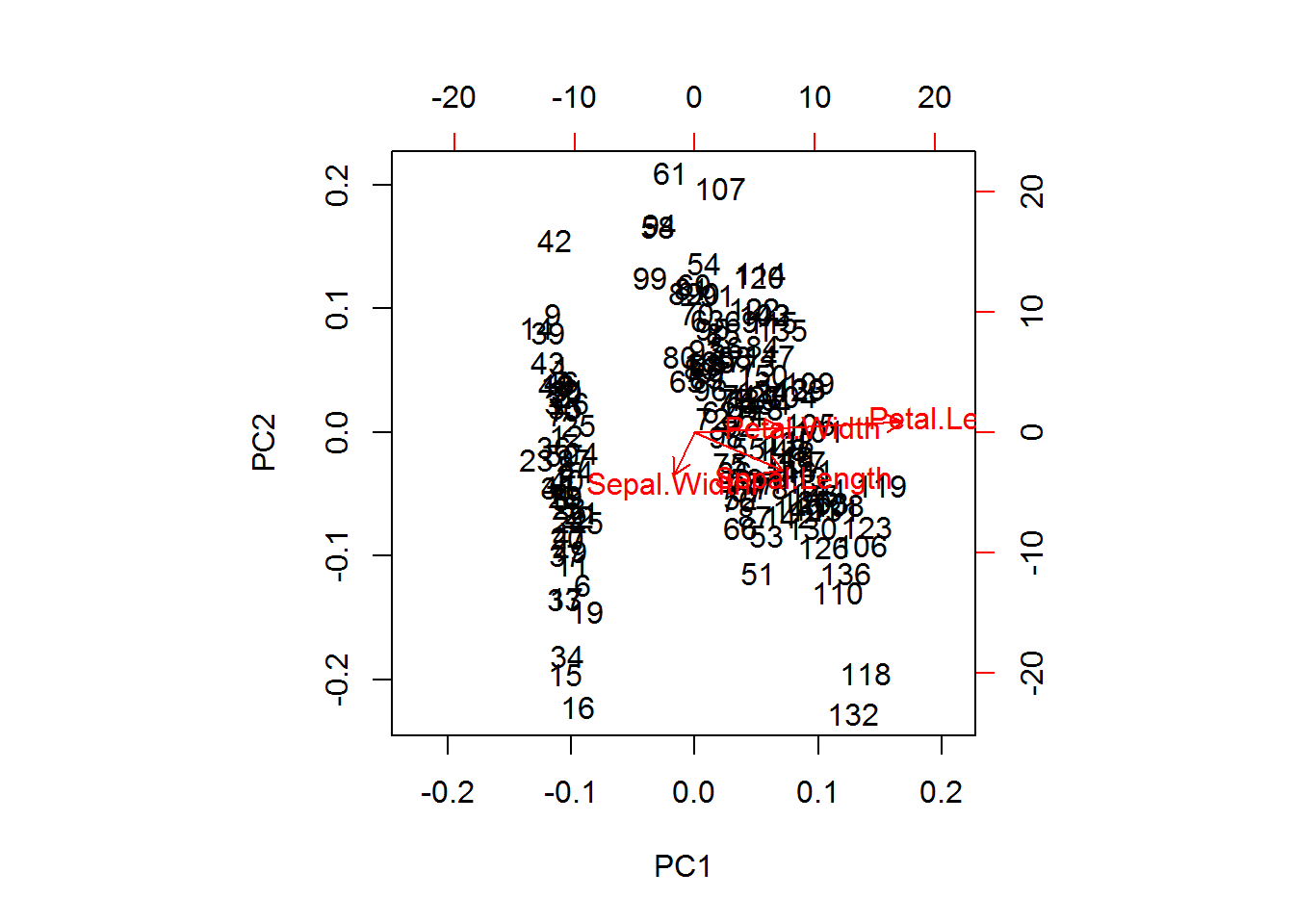
### Identify using the summary table to find the ideal number of  
### principal components needed to explain max cum variance  
summary(iris.pc)

## Importance of components:  
## PC1 PC2 PC3 PC4  
## Standard deviation 2.0563 0.49262 0.2797 0.15439  
## Proportion of Variance 0.9246 0.05307 0.0171 0.00521  
## Cumulative Proportion 0.9246 0.97769 0.9948 1.00000

### Use the rotation matrix to identify the loading of the components  
### onto the relevant PC's  
iris.pc$rotation

## PC1 PC2 PC3 PC4  
## 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

### Extract Std Dev from the matrix  
sdev <- iris.pc$sdev  
### Obtain variance  
pr\_var<- sdev^2  
### Extract the proportionate of variance to which is used for the  
### Scree plot to identify number of components needed  
prp\_var\_exp <- pr\_var/sum(pr\_var)  
### Use the biplot to identify components needed and also the need for  
### Scaling  
biplot(iris.pc,scale = T)



### Plot the Proportion of variance explained by the components  
### Plot the cumulative variance explained by the components  
par(mfrow=c(1,2))  
plot(prp\_var\_exp,xlab="Principal Components",ylab="Variance explained",type="b")  
plot(cumsum(prp\_var\_exp),xlab="Principal Components",ylab="Cumulative Variance explained",type="b")  
### Applying the PCA to model on data  
iris.t <- data.frame(Species = iris[,c(5)],iris.pc$x)  
iris.t <- iris.t[,1:3] ### Extract only the first 2 components  
### CART  
library(rpart)

## Warning: package 'rpart' was built under R version 3.4.3

iris.p.mod <- rpart(Species~PC1+PC2,data=iris.t,method = "class")  
### Convert the test data to the format of the PCA model  
iris.p <- predict(iris.pc,newdata = iris[,-c(5,6)])  
iris.p <- iris.p[,1:2]  
iris.p <- as.data.frame(iris.p)  
### Predict using the new model  
pr <- predict(iris.p.mod,newdata = iris.p,type = "class")  
head(pr)

## 1 2 3 4 5 6   
## setosa setosa setosa setosa setosa setosa   
## Levels: setosa versicolor virginica

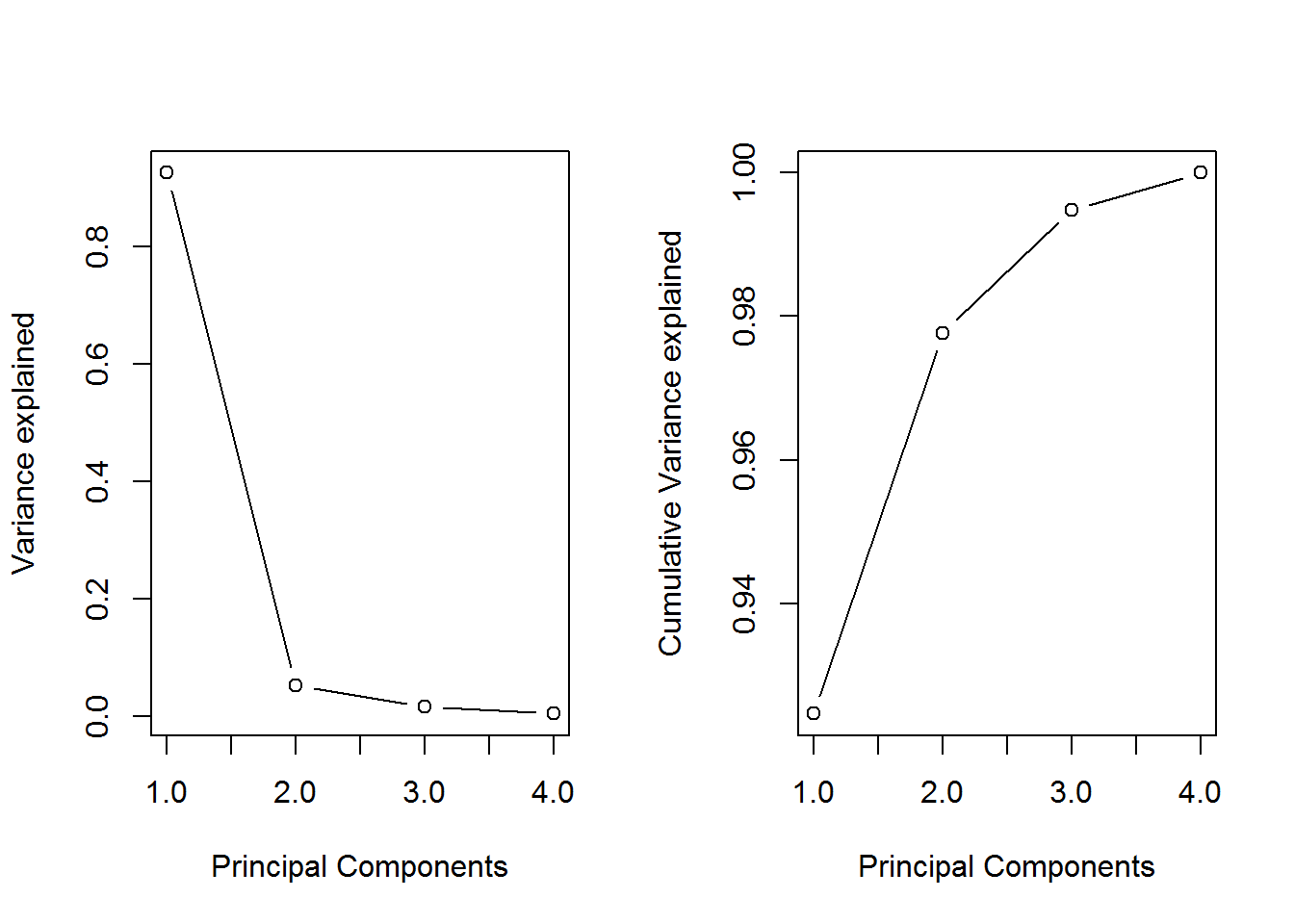
### Accuracy estimates  
library(caret)

## Warning: package 'caret' was built under R version 3.4.3

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.4.3



confusionMatrix(pr,iris.t$Species)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction setosa versicolor virginica  
## setosa 50 0 0  
## versicolor 0 43 1  
## virginica 0 7 49  
##   
## Overall Statistics  
##   
## Accuracy : 0.9467   
## 95% CI : (0.8976, 0.9767)  
## No Information Rate : 0.3333   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.92   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: setosa Class: versicolor Class: virginica  
## Sensitivity 1.0000 0.8600 0.9800  
## Specificity 1.0000 0.9900 0.9300  
## Pos Pred Value 1.0000 0.9773 0.8750  
## Neg Pred Value 1.0000 0.9340 0.9894  
## Prevalence 0.3333 0.3333 0.3333  
## Detection Rate 0.3333 0.2867 0.3267  
## Detection Prevalence 0.3333 0.2933 0.3733  
## Balanced Accuracy 1.0000 0.9250 0.9550