```
#Hold Out
library(caret)
library(klaR)
data(iris)
trainIndex=createDataPartition(iris$Species, p=0.8, list=FALSE)
dataTrain=iris[trainIndex,]
dataTest=iris[-trainIndex,]
fit=NaiveBayes(Species~., data=dataTrain)
predictions=predict(fit, dataTest[,1:4])
confusionMatrix(predictions$class, dataTest$Species)

#Bootstrap
library(caret)
data(iris)
trainControl=trainControl(method="boot", number=100)
fit=train(Species~., data=iris, trControl=trainControl, method="nb")
print(fit)

#K-Fold Cross-validation
library(caret)
data(iris)
trainControl=trainControl(method="cv",number=10)
fit=train(Species~., data=iris, trControl=trainControl, method="nb")
print(fit)

#Repeated k-fold Cross-validation
library(caret)
data(iris)
trainControl=trainControl(method="repeatedcv",number=10, repeats=3)
fit=train(Species~., data=iris, trControl=trainControl, method="nb")
print(fit)

#Leave-one-out Cross-validation
library(caret)
data(iris)
trainControl=trainControl(method="LOOCV")
fit=train(Species~., data=iris, trControl=trainControl, method="nb")
print(fit)
```