

```

> library(rpart)
> library(rpart.plot)
> library(ggplot2)
> library(klaR)
> library("caret")

> #read the data set
> data <- read.csv("heart.csv")
> set.seed(1259)
> #split up the data set into two different sets
> indexSet <- sample(2,nrow(data),replace = T, prob = c(0.7,0.3))
> train <- data[indexSet==1,]
> test <- data[indexSet==2,]
> #start of the rpart part
> tree <- rpart(
+   target~.,
+   data=train,
+   method = "class",
+   control = rpart.control(cp=0.01)
+ )
> #plotting tree
> rpart.plot(tree)
> predictions <- predict(tree, newdata = test, type = "class")
> rpartTable <- table(values = test$target, predValues = predictions)
> caret::confusionMatrix(rpartTable)
Confusion Matrix and Statistics

```

	predValues	
values	0	1
0	136	28
1	22	158

```

                Accuracy : 0.8547
                95% CI : (0.8129, 0.8902)
    No Information Rate : 0.5407
    P-Value [Acc > NIR] : <2e-16

```

```

                Kappa : 0.7082

```

```

McNemar's Test P-Value : 0.4795

```

```

    Sensitivity : 0.8608
    Specificity : 0.8495
    Pos Pred Value : 0.8293
    Neg Pred Value : 0.8778
    Prevalence : 0.4593
    Detection Rate : 0.3953
    Detection Prevalence : 0.4767
    Balanced Accuracy : 0.8551

```

'Positive' Class : 0

```
> #start of naive bayes part
> train$target = factor(train$target)
> nb <- NaiveBayes(target ~ age+sex+chest.pain.type+resting.bp.s+cholesterol+
+
+ fasting.blood.sugar+resting.ecg+max.heart.rate+exercise.angina
+
+ oldpeak,
+
+ data = train)
> # Make predictions on the test dataset
> pred <- predict(nb,test)
> naiveTable = table(pred$class, test$target)
> caret::confusionMatrix(naiveTable)
Confusion Matrix and Statistics
```

	0	1
0	142	28
1	22	152

```

                Accuracy : 0.8547
                95% CI : (0.8129, 0.8902)
   No Information Rate : 0.5233
   P-Value [Acc > NIR] : <2e-16

                Kappa : 0.7091

Mcnemar's Test P-Value : 0.4795

   Sensitivity : 0.8659
   Specificity : 0.8444
   Pos Pred Value : 0.8353
   Neg Pred Value : 0.8736
   Prevalence : 0.4767
   Detection Rate : 0.4128
   Detection Prevalence : 0.4942
   Balanced Accuracy : 0.8551

'Positive' Class : 0
```