Pulsar Classification - Analysis

Anil Kumar Pallekonda
Student ID: 157959
ANLY 530-50 Late Spring 2017
Department of Analytics
apallekonda@my.harrisburgu.edu
Harrisburg University

Abstract:- Pulsars are a rare type of Neutron star that produces radio emission which detectable on Earth. Classifying a Pulsar or noise based on emission patterns is a bit difficult to identify through radio telescopes. They are of considerable scientific interest as probes of space-time, the inter-stellar medium, and states of matter.

Objective

The main objective of this analysis document is to compare & interpret the two model performances and choose the best model for the pulsar classification task. To develop the models, we have used *'Decision Tress'* and *'Naive Bayes'* classifiers to identify and classify the given data into pulsars or noise groups.

Model 1

As a part of model-1, we have used *'Decision Tress'* algorithm to classify the data into pulsars. To accomplish this task, we have used *rpart* object from *rpart* package. To check the accuracy of the model, few performance measures can be used like precision, recall, f-measure, area under curve, etc. When we use confusion matrix for identifying the performance of the model, we should focus on few measures like sensitivity, specificity, Kappa, accuracy and P-value, etc. Some part of confusion matrix result is shown under *'Model Performance'* section.

Model Performance

Confusion Matrix and Statistics

	Reference	
Prediction	0	1
0	1779	50
1	63	1689

Accuracy : 0.9684
95% CI: $(0.9622, 0.9739)$
No Information Rate : 0.5144
P-Value [Acc $>$ NIR] : $< 2e-16$
Kappa : 0.9369
Mcnemar's Test P-Value : 0.259
Sensitivity : 0.9658
Specificity : 0.9712
Pos Pred Value : 0.9727
Neg Pred Value : 0.9640
Prevalence : 0.5144
Detection Rate : 0.4968
Detection Prevalence : 0.5108
Balanced Accuracy : 0.9685

Interpretation

Using confusion matrix results, we can say that the accuracy of the *Decision Tree* algorithm is 96.8%. 95% confidence interval shows that with this model, accuracy would be between 96.2% and 97.4%.

The sensitivity value represents true positive rate which is also known as recall. In our scenario, this value is 0.966 that means 96.6% times we are classifying pulsars as pulsars. Specificity represents true negative rate. In our scenario, that would be 0.971 that means out of 100, 97.1 times we are predicting noise as noise.The positive predictive value for this model is 0.973. This value is also known as precision. The P-value confirms that our model is performing well to classify the pulsars from noise patterns. The Kappa value for this model is 0.937. This high Kappa value represents that there is a huge difference between accuracy and the null error rate of this model.

Model 2

As a part of model-2, we have used *'Naive Bayes'* algorithm to classify the data into pulsars. To accomplish this task, we have used *naiveBayes* object from *e1071* package. To check the accuracy of the model, few performance measures can be used like precision, recall, f-measure, area under curve, etc. When we use confusion matrix for identifying the performance of the model, we should focus on few measures like sensitivity, specificity, Kappa, accuracy and P-value, etc. Some part of confusion matrix result is shown under *'Model Performance'* section.

Model Performance

Confusion Matrix and Statistics

	Reference	
Prediction	0	1
0	1750	62
1	92	1677

Accuracy : 0.957

95% CI : $(0.9498, 0.9634)$
No Information Rate : 0.5144
P-Value [Acc $>$ NIR] : $< 2e-16$
Kappa : 0.914
Mcnemar's Test P-Value : 0.01945
Sensitivity : 0.9501
Specificity : 0.9643
Pos Pred Value : 0.9658
Neg Pred Value : 0.9480
Prevalence : 0.5144
Detection Rate : 0.4887
Detection Prevalence : 0.5060
Balanced Accuracy : 0.9572

Interpretation

Using confusion matrix results, we can say that the accuracy of the *Naive Bayes* algorithm is 95.7%. 95% confidence interval shows that with this model, accuracy would be between 94.9% and 96.3%.

The sensitivity value represents true positive rate which is also known as recall. In our scenario, this value is 0.95 that means 95% times we are classifying pulsars as pulsars. Specificity represents true negative rate. In our scenario, that would be 0.964 that means out of 100, 96.4 times we are predicting noise as noise.The positive predictive value for this model is 0.966. This value is also known as precision. The P-value confirms that our model is performing well to classify the pulsars from noise patterns. The Kappa value for this model is 0.914. This high Kappa value represents that there is a huge difference between accuracy and the null error rate of this model.

CONCLUSION

By comparing above mentioned models, model-1(*Decision Trees classifier*) performs better than model-2(*Naive Bayes classifier*). So, we will be using Decision Tree classifier in pulsar identification pipeline to classify the observed patterns as pulsars or noise groups.

REFERENCES

[1] K. J. Lee et al. Peace: pulsar evaluation algorithm for candidate extraction a software package for post-analysis processing of pulsar survey candidates, 2013.
[2] M. J. Keith et al. The high time resolution universe pulsar survey - i. system configuration and initial discoveries, 2010.
[3] R. J. Lyon et al. Fifty years of pulsar candidate selection: From simple filters to a new principled real-time classification approach.
[4] R. J. Lyon et al. Selection of radio pulsar candidates using artificial neural networks, 2010.
[5] S. D. Bates et al. The high time resolution universe pulsar survey vi. an artificial neural network and timing of 75 pulsars, 2012.
[6] V. Morello et al. Spinn: a straightforward machine learning solution to the pulsar candidate selection problem, 2014.
[7] R. J. Lyon; B. W. Stappers; S. Cooper; J. M. Brooke; J. D. Knowles. Fifty years of pulsar candidate selection: From simple filters to a new principled real-time classification approach.
[8] D. R. Lorimer and M. Kramer. Handbook of pulsar astronomy, 2005.
[9] R. J. Lyon. Why are pulsars hard to find?, 2016.
[10] R. J. Lyon. Htru2, 2017.
[11] D. Thornton. The high time resolution radio sky, 2013.

APPENDIX

'R' Source Code

```
library(ROSE)
library(rpart)
library(readr)
library(psych)
library(caret)
library(e1071)
library(rattle)
# library(naivebayes)
# library(Boruta)


HTRU_2 <- read_csv("D:/Academics/Semester
   -IV(Late Spring 2017)/Machine Learning
    - I/Project work/Data/HTRU2/HTRU_2.
   csv", col_types = cols(Class = col_
   factor(levels = c("0", "1"))), na = "
   NA")

# shuffling row qise

HTshuf <- HTRU_2[sample(nrow(HTRU_2)),]


# split into training and testing set
   being 80% data in training set

size <- round(nrow(HTshuf)*0.8)
trainset <- HTshuf[1:size,]
testset <- HTshuf[size: nrow(HTshuf),]

# feaSel <- Boruta(Class~., data =
   trainset, doTrace = 2)
#
# feaSel$ImpHistory

# Chekcing the proportion of training and
    testing set pulsar Vs. noise

# prop.table(table(trainset$Class))
# prop.table(table(testset$Class))

# Creting decision tree moel

tree <- rpart(Class~., data = trainset)

# tree$variable.importance

predtree <- predict(tree, newdata =
   testset)

# Checking for accuracy of the model.
```

```r
accuracy.meas(testset$Class, predtree
    [,2])
roc.curve(testset$Class, predtree[,2])

# Balancing imbalanced dataset with ROse
    Function

trainRose <- ROSE(Class~.,data = trainset
    , seed = 1)$data
testRose <- ROSE(Class~., data = testset,
     seed = 1)$data

# Checking the proportion of training and
     test set pulsar Vs. noise
# table(trainRose$Class)
# table(testRose$Class)

# Modeling with Decision Tree algorothm

treeRose <- rpart(Class~., data =
    trainRose)
# treeRose$variable.importance
PredtreeRose <- predict(treeRose, newdata
     = testRose)
# accuracy.meas(testRose$Class,
    PredtreeRose[,2])
# roc.curve(testRose$Class, PredtreeRose
    [,2])
# plotting tree
# fancyRpartPlot(treeRose, palettes=c("
    Greys", "Oranges"))


confusionMatrix(round(PredtreeRose[,2],
    digits = 0), testRose$Class)

# round(PredtreeRose[,2], digits = 0)
# plot(treeRose, uniform=TRUE, main="
    Classification Tree for Plusars")
# text(treeRose, use.n=TRUE, all=TRUE,
    cex=.7)
# labels(treeRose, digits = 4, minlength
    = 1L, pretty, collapse = FALSE)
#
# plotcp(treeRose)
# text(treeRose)


# treeRoseImp <- rpart(Class~SkeIGP+EKIGP
    +MeanIG+SDDMSNR+EKDMSNR+MeanDMSNR+
    SDIGP, data = trainRose)
# treeRoseImp$variable.importance
# PredtreeRoseImp <- predict(treeRoseImp,
     newdata = testRose)
# accuracy.meas(testRose$Class,
    PredtreeRose[,2])
# roc.curve(testRose$Class, PredtreeRose
    [,2])

# accuracy.meas(testRose$Class,
    PredtreeRose[,2])
# roc.curve(testRose$Class, PredtreeRose
    [,2])

 # confusionMatrix(PredtreeRose, testset$
     Class)
# naive bayes

# Modleing with Naive Bayes Algorithm



# model2 <- naive_bayes(Class~.,data =
    trainRose)
# PredModel2 <- predict(model2, newdata =
     testRose)
#
# plot(model2, which = NULL, ask = TRUE,
    legend = TRUE, main = "Naive Bayes
    Plot")

# Modeling with Naive Bayes Algorithn
model <- naiveBayes(Class~., data =
    trainRose)

# Stats of model
# class(model)
# summary(model)
# print(model)

# Predecting pulsar or noise using
    developed model
predmodel <- predict(model, newdata =
    testRose)

# Checking accuracy of the model
confusionMatrix(predmodel, testRose$Class
    )
# roc.curve(testRose$Class, predmodel )
```