ANALYSIS OF BACK PAIN USING MACHINE LEARNING CLASSIFICATION ALGORITHMS

Individual Project

Author:

SONNET PADAYATTY REJU (18200682)

Professor:

KEEFE MURPHY

STAT30270 – STATISTICAL MACHINE LEARNING

DEPARTMENT OF MATHEMATICS

UNIVERSITY COLLEGE DUBLIN

Belfield, Dublin 4, Ireland

April 2019

ABSTRACT

The project analyzes the information about subjects suffering from lower back pain. Our aim is to classify subjects according to the type of lower back pain they experience, using a collection of numerical and categorical clinical indicators. The response categories provide a clinically meaningful binary classification of lower back pain, named as “Nociceptive” & “Neuropathic”. This project is targeted to create an efficient machine learning technique which identifies the pain diagnosis required for each patient. As the target variable in this particular dataset is not continuous, we can apply variety of classification models that are good at prediction. Due to the availability of various classifiers, it is our duty to find the best one from the collection and fit the optimal model for this dataset. Once the best model is found, the prediction for future observations can be done easily. The best model is fitted to a test data and the accuracy of the model is estimated. If the performance of the model is acceptable in test data, then the interpretation of rules used by the best model to predict the response variable is analyzed. This may help us to find out hidden relationships between the predictor and response variables.

TABLE OF CONTENTS

INTRODUCTION

The dataset contains information about subjects suffering from lower back pain. Our interest is to classify subjects according to the type of lower back pain they experience, using a collection of numerical and categorical clinical indicators. The response categories provide a clinically meaningful binary classification of lower back pain, and are defined as follows

Nociceptive

Pain arising from actual or threatened damage to non-neural tissue occurring with a normally functioning somatosensory nervous system.

Neuropathic

Pain initiated or caused by a primary lesion or dysfunction in the nervous system.

It is thought that classifying patients lower back pain based on a clinical judgment regarding the likely dominant category of neurophysiological mechanisms responsible for its generation and/or persistence may usefully inform treatment by inviting clinicians to select treatments either known or hypothesized to target those mechanisms in an attempt to optimize clinical outcomes

METHODS

Pre-processing

Once the given dataset is loaded, we are supposed to do exploratory analysis on it. First of all, we can check whether the involved variables are in proper format or not. Using the “str” command in R, it is found that there are 19 categorical variables and 13 continuous variables, and all of them are in proper format by default. In addition to that, we got a dataset where all the observations are complete. Hence we can rule out missing value analysis as well. If we check the data distribution of each continuous variable involved, it is found that there are no outliers which can single handedly make influence on the response variable. By considering all this, it is not at all necessary to make any extensive pre-processing steps. The libraries for all the machine learning techniques are loaded initially. Before doing any further steps, we can set the seed to a constant value- 1000, which helps to reproduce the results.

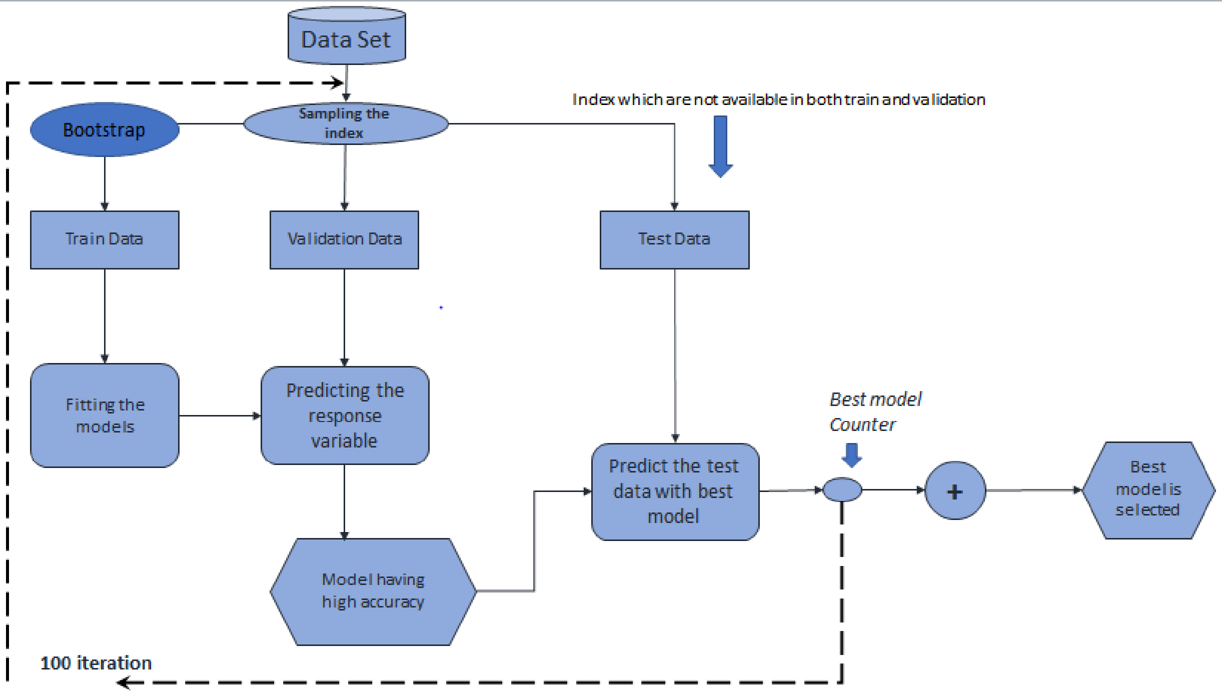
Modeling

Now we are splitting the dataset into 3 subsets – train, validation & test. The splitting can be achieved by sampling the index of the raw dataset. We can achieve a train dataset having 380 observations by applying bootstrapping technique. Once we obtained the train dataset, we can sample the index for validation and test dataset by confirming that the observations of each dataset are not available in others. As our aim is to classify the response variable -PainDiagnosis, the following machine learning techniques / Classifiers are used.

1. Classification trees (R part)
2. Logistic regression (multinom)
3. Bagging (Ensemble method)
4. Random Forest (Ensemble method)
5. SVM (Kernel method)
6. Boosting

Process

The train dataset is fitted with the above mentioned machine learning techniques and the resulted models are compared by predicting the response variable – PainDiagnosis of validation dataset. When we pass the dataset to the trained models, we can see that the best model is "Boosting" compared to rest of the models. The summary of results of all models over a single iteration is shown in Table 1. From this we can find the best model based on the accuracy of each in validation dataset. Apart from accuracy of the model, we need to assess the sensitivity, specificity and ROC curves in each case. ROC curve of each model is shown in Fig 1. Once we find the best model, we can predict the response variable – PainDiagnosis of test dataset, in order to find the performance of the final model. It is not possible to confirm that the best model is "Random forest" by fitting the datasets obtained in a single splitting process/bootstrapping. We might get a better result if we iterate the whole process again. So it is recommended to store the results of each iteration in a matrix, and compare the models each other. For this, all the modeling steps are performed in a "for" loop and iterate it for 100 times. The most chosen models with maximum mean accuracy is considered as the best model. The flow chart of the whole process is shown below.


Data Set
Index which are not available in both train and validation
Bootstrap
Sampling the index
Train Data
Validation Data
Test Data
Fitting the models
Predicting the response variable
Best model Counter
Predict the test data with best model
+
Best model is selected
Model having high accuracy
100 iteration

RESULTS

Table 1: Performance analyses of all models comparing with Validation dataset

Method	**Accuracy**
Classification tree	0.828947
Logistic regression	0.881578
Ensemble (Bagging)	0.868421
Ensemble(RandomForest)	0.986842
SVM	0.960526
Boosting	0.973684

Fig 1: ROC curves of all models (validation dataset)

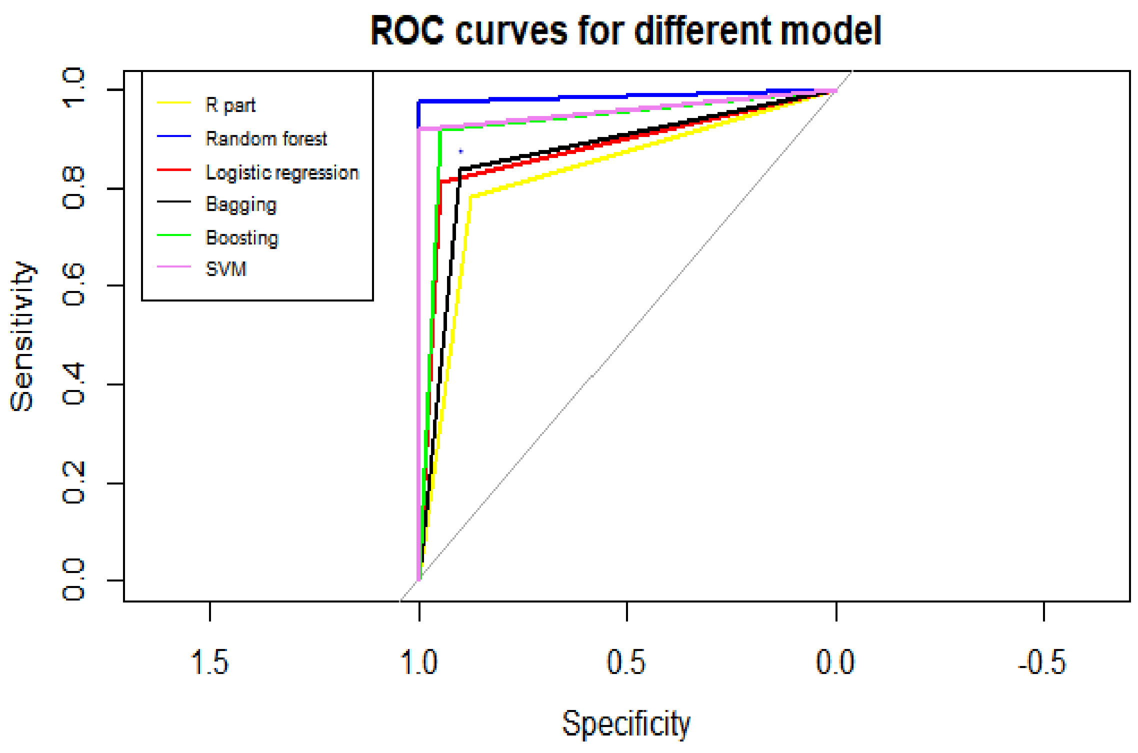


Table 2: Performance analysis of best model in test dataset

Method	Accuracy
Random Forest	0.906666

Table 3: Performance analysis of all models based on following factors

Methods	Specificity	Sensitivity	False Negative Rate
Classification tree	0.8529412	0.8095238	19%
Logistic regression	0.9375	0.8409091	16%
Random forest	1	0.975	2.50%
SVM	1	0.9285714	7.20%

Table 4: performance analysis of all models in validation dataset and best model in test dataset for 100 iterations(snapshot first first 5 iterations)

	valid.r	valid.l	valid.ran	valid.svm	valid.boost	chosen	test
[1,]	0.9157895	0.9368421	0.9368421	0.9263158	0.8947368	2	0.9047619
[2,]	0.8315789	0.8736842	0.9368421	0.9578947	0.9157895	3	0.8333333
[3,]	0.8736842	0.8736842	0.8947368	0.8631579	0.9052632	5	0.9052632
[4,]	0.8315789	0.8631579	0.8736842	0.9052632	0.8947368	3	0.9555556
[5,]	0.8210526	0.8842105	0.9368421	0.9473684	0.9473684	3	0.9285714

Table 5: Summary statistics of the above table

	valid.r	valid.l	valid.ran	valid.svm	valid.boost	test
Min.	0.7157895	0.7578947	0.8210526	0.8210526	0.8105263	0.7575758
1st Qu.	0.8500000	0.8526316	0.9052632	0.8947368	0.8947368	0.8934271
Median	0.8842105	0.8736842	0.9315789	0.9263158	0.9263158	0.9261209
Mean	0.8754737	0.8744211	0.9266316	0.9176842	0.9191579	0.9202070
3rd Qu.	0.9052632	0.8973684	0.9473684	0.9368421	0.9368421	0.9555556
Max.	0.9473684	0.9578947	0.9894737	0.9789474	0.9789474	1.0000000

Table 6: Number of times each model selected as best model

Method	Frequency
Classification(Rpart)	5
Logistic regression	10
SVM	37
Random Forest	34
Boosting	14

Fig 2: Variable importance plot

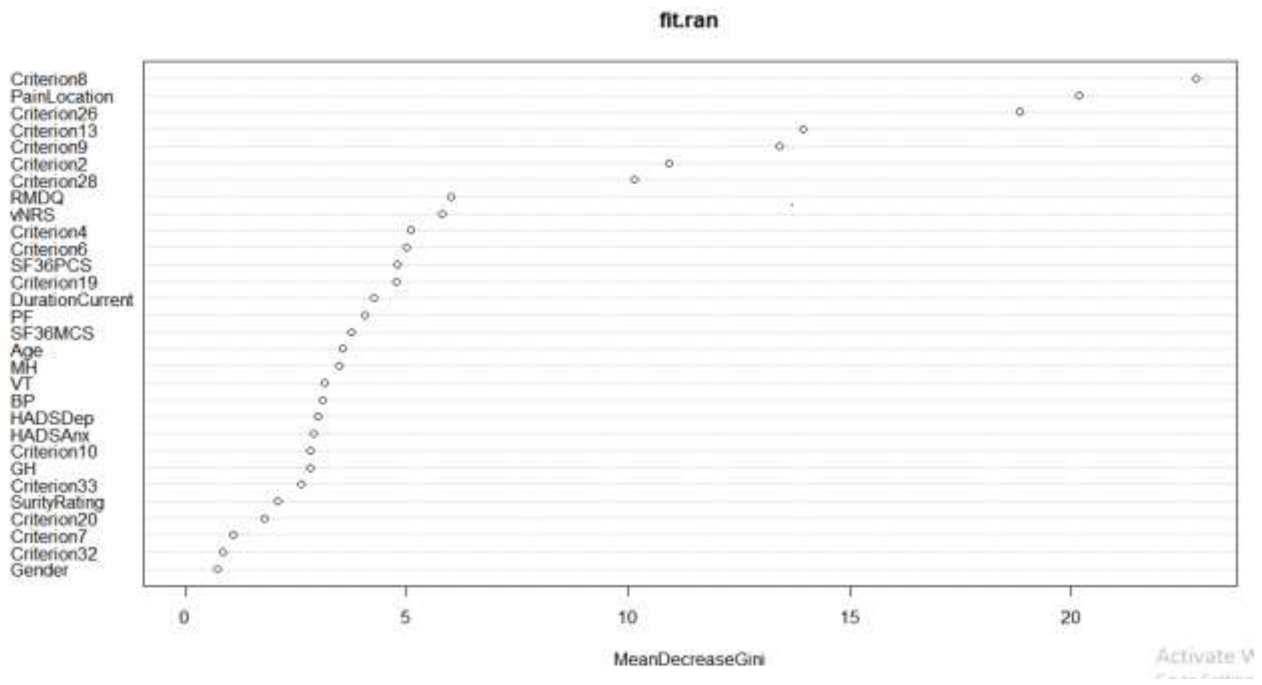


Table 7: Important variables of random forest model

Variables	Overall	Variables	Overall	Variables	Overall
Criterion8	22.812606	Criterion4	5.106051015	BP	3.099870368
PainLocation	20.17772036	Criterion6	5.005355376	HADSDep	3.016065895
Criterion26	18.82872183	SF36PCS	4.793163894	HADSAnx	2.89498296
Criterion13	13.95185371	Criterion19	4.765972141	Criterion10	2.828866296
Criterion9	13.40844798	DurationCurrent	4.269038738	GH	2.827613449
Criterion2	10.914966	PF	4.076748462	Criterion33	2.623302881

Output 1: Important rules that extracted from Random forest

```
len freq    err
[1,] "3" "0.542" "0.0679999999999999"
[2,] "3" "0.455" "0.012"
[3,] "2" "0.468" "0.022"
[4,] "3" "0.058" "0.091"
[5,] "3" "0.161" "0.033"
     condition
[1,] "X[,4] %in% c('Back','Back+Thigh') & X[,21] %in% c('Present') & X[,24] %
in% c('Absent')"
[2,] "X[,17] %in% c('Present') & X[,22] %in% c('Absent') & X[,30] %in% c('Abs
ent')"
[3,] "X[,17] %in% c('Present') & X[,22] %in% c('Absent')"
[4,] "X[,5]<=7.5 & X[,14]>45.8 & X[,27] %in% c('Absent')"
[5,] "X[,13]<=39.6 & X[,20] %in% c('Present') & X[,25] %in% c('Present')"
     pred          impRRF
[1,] "Nociceptive" "1"
[2,] "Nociceptive" "0.183713357227392"
[3,] "Nociceptive" "0.146850998852421"
[4,] "Nociceptive" "0.0363072291938474"
[5,] "Neuropathic" "0.0346989167488646"
```

DISCUSSIONS

From Table 1, it is evident that modeling with Random forest is efficient than Bagging. So when we do performance analysis of all models in 100 iterations (later), Random forest is chosen as the best ensemble method. The dataset used here is validation and random forest came out as the best model among all 5 models. The best model cannot be determined by comparing the accuracies alone. There are other factors that need to be considered. For example, sensitivity, specificity, false negative rate, area under ROC curve etc. ROC curve is drawn by plotting sensitivity against specificity, and from the Fig 1 it is obvious that the area under the curve of Random forest model is larger compared to rest of the models. From table 3, it is found that the false negative rate is minimum for Random forest compared to rest of the models. For some practical reasons, tests which have sensitivity and specificity greater than 90% have higher credibility. Hence Random forest can be considered as the best model, even if the SVM model has better accuracy in some of cases which we will discuss later.

False negative rate = 1- sensitivity & False positive rate = 1- specificity

The best model is applied on test dataset to predict the response variable – PainDiagnosis and it is found that the accuracy of the best model is 0.9066667 % as shown in Table 2.

Performance analysis of all models in validation dataset, and its best model in test dataset, in all iterations, is shown in Table 4 & 5. As explained before, we are supposed to check the performance of all models over different datasets obtained as a result of sampling process. Table 4 shows the results of first 5 iterations. Table 6 shows the frequency of each model that is selected as the best model (out of 100 iterations). From Table 5, it is found that the random Forest is still the best model with a frequency of 34 and mean accuracy 92%. The second best model can be considered as the model with SVM machine learning technique with mean accuracy slightly less than random forest. Even though SVM got selected as best model for 37 times, we are considering Random forest as the best model due to its higher mean accuracy and sensitivity. Classification tree and logistic regression cannot be considered as good models as they come out as best models in only 5 and 10 cases. The highest accuracy obtained by a model in test dataset is 100%.

Once we fit the best model with the test dataset, we can obtain the important variables from varI mpPlot command available for random forest models. The Fig 2 shows the graph between Varia bles and Mean Decrease Gini. As the value of x increases, the importance of variable also increas es. So the variables which are mentioned on the top of the y axis can be considered as important variables. Table 5 shows the variables involved with its corresponding mean decrease gini value. From both plot and table, it is obvious that the most important variables are Criterion8 (Localised to area of injury, dysfunction) and Pain location.

From the rules extraction results as shown in Output 1, we can say that if the following conditions are satisfied:-

1. Pain location is at back or back+thigh.
2. Criterion 8 (Localised to area of injury, dysfunction) is present
3. Criterion13 (Disproportionate, non-mechanical pattern to aggs + eases) is absent.

There is a high chance that the Pain diagnosis required is “Nociceptive”. Similarly we can interpret rest of the rules.

CONCLUSION

From overall analysis, we found out that Random Forest model & SVM model are the efficient classifiers as they were selected as the best models in numerous cases. Due to the higher ROC curve area & higher mean accuracy in 100 iterations, the model- Random Forest is little bit more efficient than other machine learning techniques. According to the important variables found in random forest, we can say that the Pain diagnosis is highly dependent on Pain location and the criterion – Pain localized to the area of injury/ dysfunction”.

There are many cases which decides the pain diagnosis, but the most significant rules are followings:-

If we found that the pain location is at back or back+thigh & the pain localized to the area of injury/dysfunction is present & Disproportionate/ non-mechanical pattern to aggs + eases is absent, then there is a high probability that the diagnosis required is “Nociceptive”. If vitality is less than 39.6 & patient suffers from night pain or disturbed sleep & presence of burning / sharp / electric shock like pain is found, the Pain diagnosis type can be “Neuropathic”.

REFERENCE

1. **REVOLUTIONS**

ROC Curves in Two Lines of R Code

by Bob Horton, Microsoft Data Scientist

link: https://blog.revolutionanalytics.com/2016/08/roc-curves-in-two-lines-of-code.html

2. **Rdrr.io**

Link: https://rdrr.io/cran/inTrees/src/R/extractRules.R