UNIVERSITY COLLEGE DUBLIN

STAT30270: Statistical Machine Learning

PROJECT REPORT

on

CLASSIFYING LOWER BACK PAIN USING DIFFERENT CLASSIFICATION TECHNIQUES

Submitted by,

Justin Joseph

Student Number: 18201354

Submitted on: Apr 28, 2019

Contents

ABSTRACT

Lower back pain can be caused due to a variety of reasons and it is one of the most common medical condition in the modern world. Proper identification of the cause of the back pain can widely help in the treatment of lower back pain. Lower back pain can be caused due to various problems related with spine, nerves, ligaments, joints or tendons. Based on this, lower back pain can be categorized into two main categories, one caused by the damage to non-neural tissue with a functioning nervous system and the other caused by dysfunction in the nervous system. Usually clinical experts perform the task of identifying the cause of lower back pain from the results of various tests conducted on the patient, the symptoms recorded by the patient and their different attributes.

In this project we will use different machine learning classification techniques to classify the types of lower back pain based on the results of the tests, the symptoms recorded by the patient and their different attributes. Since this task is related to medical science and needs very accurate results, we will find the classification model that classifies the lower back pain with the best accuracy in a consistent manner. We will also identify the important variables/parameters that will aid in proper classification.

INTRODUCTION

The given dataset backpain.RData, contains information about patients suffering from back pain (includes the various test results, patient attributes and symptoms recorded by the patient). There are 380 observations in the dataset with **32** variables (**31** predictor variables and **1** response variable). Out of the **32** variables, **19** variables are categorical, **1** variable is ordinal and **12** variables are numerical. The response variable is divided into **2** categories, **Nociceptive** (pain arising from damage to non-neural tissue occurring with a functioning somatosensory nervous system) and **Neuropathic** (pain caused by a primary lesion or dysfunction in the nervous system). A brief description of the variables is given below.

- PainDiagnosis: "Expert pain diagnosis (as reference standard)." [response variable]
- Age: "Age"
- Gender: "Gender"
- DurationCurrent: "Duration of current episode"
- PainLocation: "Pain location"
- SurityRating: "Surity rating of expert clinical diagnosis."
- RMDQ: "Roland Morris Disability Questionnaire score."
- vNRS: "Verbal NRS for pain intensity."
- SF36PCS: "SF36 Physical Component Summary score."
- SF36MCS: "SF36 Mental Component Summary score."
- PF: "Physical Functioning"
- BP: "Bodily Pain"
- GH: "General Health"
- VT: "Vitality"
- MH: "Mental Health"
- HADSAnx: "HADS Anxiety score."
- HADSDep: "HADS Depression score."
- Criterion2: "Pain assoc'd trauma, pathology, movt."
- Criterion4: "Pain disproportionate to injury, pathology."
- Criterion6: "More constant, unremitting."
- Criterion7: "Burning, shooting, sharp, electric-shock like."
- Criterion8: "Localized to area of injury, dysfunction."
- Criterion9: "Referred in dermatomal, cutaneous distribution."
- Criterion10: "Widespread, non-anatomical distribution."
- Criterion13: "Disproportionate, non-mechanical pattern to aggs + eases."
- Criterion19: "Night pain, disturbed sleep."
- Criterion20: "Responsive to simple analgesia, NSAIDS."
- Criterion26: "Pain with high levels of functional disability."
- Criterion28: "Consistent, proportionate pain reproduction on mechanical testing."
- Criterion32: "Localized pain on palpation."
- Criterion33: "Diffuse, non-anatomic areas of pain on palpation."
- Criterion36: "Positive findings of hyperpathia.

Clinical experts have classified the lower back pain of different patients based on the tests above. We will use this data and try to classify the lower back pain using different machine learning

classification algorithms and to identify the classification method with highest accuracy and consistency. We will also identify the variables/parameters that are most important for proper classification.

METHODS

After loading the dataset in R, we analyze the data to check if any pre-processing is required. It can be seen that the dataset does not have any NA values or any potential outliers. All the categorical variables are defined as factors, except for the ordinal variable. The ordinal variable is converted as factor. Since the dataset is small and in order to make our model capable of handling a wide variety of data, we use bootstrap samples to create the train data and then use the rest of the data for validation data and test data.

The following machine learning classification techniques are used.

1. Logistic Regression

Logistic regression is a classification technique used to classify dichotomous (binary) response variables. Logistic regression basically works with probability and it models the probability of the first class. To ensure that the probability of first class lies between 0 and 1, the log odds of the first class is used. The general equation for logistic regression is,

$$\log\left(\frac{P(1st\ class)}{1-P(1st\ class)}\right) = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \cdots + \beta_n * X_n$$

where X_1 to X_n are the predictor variables. The probability of an observation to belong in 1st class will be closer to 1. Multinomial logistic regression, which is an extension of logistic regression can be used to classify response variables with more than 2 categories. In the given classification problem, even though the response variable is binary, we use multinomial logistic regression for classification as it produces better results than ordinary logistic regression. In R, multinom() function is used to perform multinomial regression.

2. Classification Trees

Classification tree is a simple classification technique having a flowchart like structure. Each node represents the predictor variables required to reach a particular class of the response variable. The variables which gives more information (or having lesser gini impurity) is chosen as the nodes of the tree. The size of the tree is always kept to a minimum so that it is not very sensitive to small changes in the data. Classification can be done in R using 2 functions, rpart() and ctree(). Both these classifications gives the same result and hence we are only using

rpart() for this classification problem.

3. Bagging

Bagging is an ensemble method that makes use of multiple classifiers to improve accuracy. Bagging is the short for bootstrap aggregating. Bagging takes bootstrap sample of the data to create diversity and run classification tree multiple time on the data. The classification tree which gives the highest accuracy is chosen as the best model and this will be the output of bagging method. Bagging is done in R using bagging() function. Since bagging takes a lot of time to complete, we reduce the default number of iterations from 100 to 25 using mfinal parameter.

4. Random Forest

Random forest is another ensemble method. It is a very powerful classification method that uses classification tree and bootstrapping extensively. It is more powerful than bagging. Random Forest is done in R using randomForest() function.

5. Boosting

Boosting is a powerful method for developing powerful classifiers. In boosting, the importance of the misclassified observations is increased and hence the performance can be improved. The importance is increased by introducing weights to the misclassified observations. Boosting is done in R using boosting() function. Since boosting takes time, we have reduced the number of iterations to 25 using mfinal parameter. Boosting is very sensitive and hence if any observations are labelled incorrectly, the classifier will never classify correctly. Since the dataset used here is decision made by clinicians, the chances of human error is high and if errors are present it can lead to incorrect classifications and hence this method is not very much preferred for this problem.

6. Support Vector Machine

Support Vector Machine is a popular method of building classifiers. The basic idea is to find the best plane that separate the different classes. SVM is a kernel based approach. SVM is implemented in R using ksvm() function. In this problem we are using the default radial basis function kernel since the results with other kernels are almost the same.

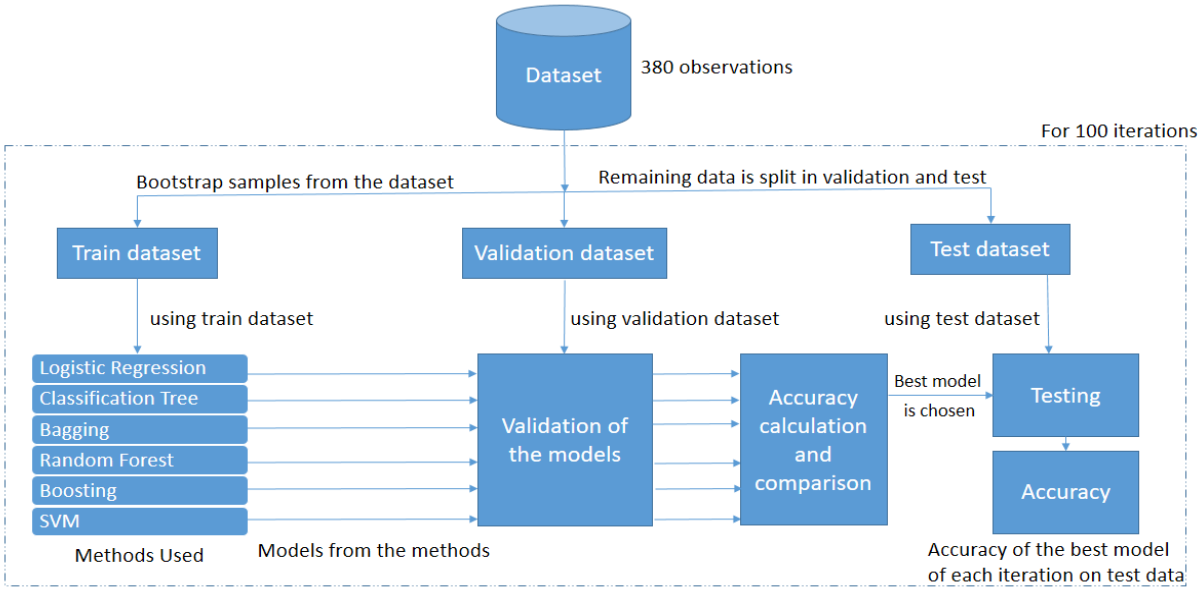

Figure 1: Algorithm to find the best model

In order to find the best model, we perform the below steps (Fig 1.):

1. Create a bootstrap sample for the train data from the dataset and create validation and test data using the remaining data.
2. Create models for each of the 6 classification algorithms discussed above using train data.
3. Predict the validation data using the above models and calculate the accuracy of each models. The model with the highest accuracy is selected as the best model of the round.
4. Repeat the process for 100 iterations. The classification technique that has more number of highest accuracy in 100 iterations is chosen as the best method.

RESULTS

Table 1: First 5 rows of the result from 100 iterations

Validation Logistic	Validation Classification	Validation Bagging	Validation Random Forest	Validation Boosting	Validation SVM	Best method	Test Accuracy using best method
0.8382	0.8529	0.9117	0.9705	0.9558	0.9411	Random F	0.9758
0.9275	0.8985	0.9275	0.9275	0.8985	0.9275	Logistic	0.8857
0.8000	0.9428	0.9428	0.9571	0.9285	0.9571	Random F	0.9571
0.8873	0.9014	0.9295	0.9718	0.9436	0.9436	Random F	0.9718
0.8888	0.8888	0.9305	0.9444	0.9305	0.9027	Random F	0.9444

Table 2: Range of Accuracy of all models on Validation Dataset (for 100 iterations)

	Logistic Regression	Classification Tree	Bagging	Random Forest	Boosting	Support Vector Machine
Minimum	0.7536	0.7353	0.8060	0.8438	0.8406	0.8154
Average	0.8750	0.8813	0.9059	0.9306	0.9228	0.9159
Maximum	0.9577	0.9589	0.9861	1.0000	1.0000	1.0000

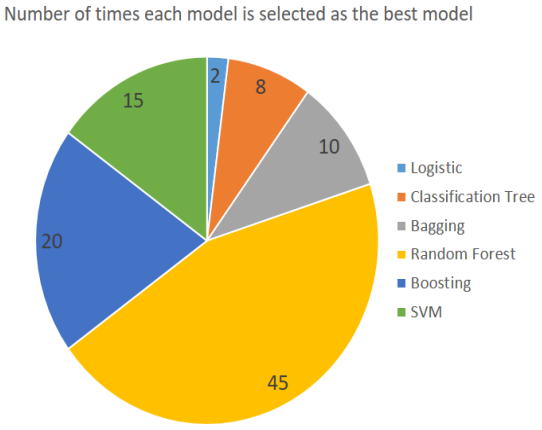

Figure 2: Number of times voted as best model

Table 3: Range of Accuracy of the best model on test data

Method	Minimum Accuracy	Mean Accuracy	Maximum Accuracy
Random Forest	0.8767	0.9451	1.0000

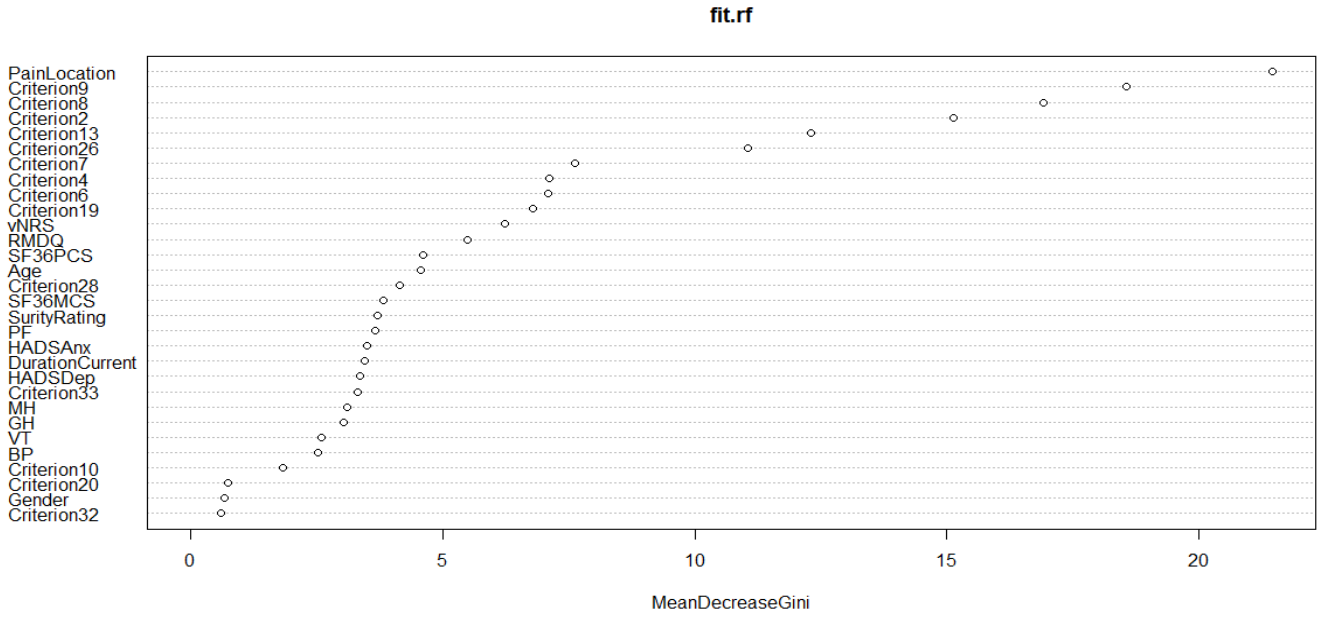

Figure 3: Variable Importance plot for Random Forest

Table 4: Top 6 Mean Decrease Gini

Variable	Mean Decrease Gini
PainLocation	21.4626733
Criterion9	18.5672317
Criterion8	16.9091886
Criterion2	15.1293905
Criterion13	12.3081033
Criterion26	11.0388332

Table 5: Lowest 6 Mean Decrease Gini

Variable	Mean Decrease Gini
Criterion36	0.3687897
Criterion32	0.5872591
Gender	0.6574696
Criterion20	0.7356528
Criterion10	1.8187474
BP	2.5191912

DISCUSSION

Table 1 shows the first 5 rows of the results from the 100 iterations. The accuracy of the model of different method for each iteration can be seen from this table.

Table 2 gives the range of accuracy of all the models on validation dataset for 100 iterations. It can be seen that the lowest value of accuracy is given by logistic regression and classification tree methods. The highest accuracy is given by Random Forest, Boosting and Support Vector Machine methods. Random Forest has the largest minimum value when compared to the other methods and spans over a smaller range. Random Forest has a good mean accuracy value of **93.06%**. Performance of Boosting method is also close to that of Random Forest.

Figure 2 shows the number of times each method was chosen as the best method in each iteration. It can be observed that Random Forest was chosen as the best model **45** times. Boosting was chosen **20** times and SVM was chosen **15** times.

Table 3 shows the range of accuracy of the best model on test data during the 100 iterations. It can be seen that the minimum accuracy of Random Forest when it was chosen as the best model is **87.67%** and the maximum accuracy is **100%**, with an average accuracy of **94.51%**.

Figure 3 is a plot between each of the variables and their Mean Decrease Gini, used for the Random Forest model. It is obtained using the VarImpPlot() function in R. The variables are arranged in the order of their importance. The variables at the top of the plot are important and the ones at the bottom are the least important.

Table 4 and **Table 5** shows the Mean Decrease Gini values of the 6 most important variables and 6 least important variables in the Random Forest model. The variables PainLocation, Criterion9 and Criterion8 are very important, whereas the variables Criterion36 and Criterion32 are the least important.

CONCLUSION

From the above discussions we can conclude that Random Forest is the better method to classify lower back pain in a better way. It gives the highest accuracy and also is more consistent than the other models that gives similar accuracy. The average accuracy of Random Forest method is **94.51%**.

The variable importance plots and values gives the variables that are the most important to classify the problem using Random Forest method. Hence the variables that gives the least information can be avoided for future classification. In other words, the Random Forest method will not need the entire set of variables and still can give better accuracy.

REFERENCES

1. Classification of Lower Back Pain Disorder Using Multiple Machine Learning Techniques and Identifying Degree of Importance of Each Parameter. <http://article.nadiapub.com/IJAST/vol105/2.pdf>.

2. Decision Tree in Machine Learning <https://towardsdatascience.com/decision-tree-in-machine-learning-e380942a4c96>

3. The Random Forest Agorithm <https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd>

4. Logistic Regression <https://www.statisticssolutions.com/what-is-logistic-regression/>

5. Machine Learning Classification Project: Finding Donors. <https://towardsdatascience.com/classification-project-finding-donors-853db66fbb8c>