Chapter 39
Enhancing the Classification Performance of Lower Back Pain Symptoms Using Genetic Algorithm-Based Feature Selection

Abdullah Al Imran, Md. Rifatul Islam Rifat and Rafeed Mohammad

1 Introduction

Studies over the past two decades have identified Lower Back Pain (LBP) as one of the predominant reasons for human physical disability that is preventing most of the people from engaging in work, as well as other daily activities at an early age. Medical experts claim that most of the people in the world experience LBP at least once at some point in their lifespan. LBP that comes on suddenly and cures within 6 weeks can be caused by a fall or heavy lifting and referred to as acute LBP. It is a common case that cures with home treatment and self-care. In contrast, if it lasts longer than 3 months, then it can be referred to as chronic LBP which is less common than acute LBP. It can occur from a wide variety of conditions including poor posture, spinal curvature, muscle strain, kidney infection, or a herniated disc and is characterized by a dull ache, sharp pain, spasm, and in some cases, localized inflammation [1]. Detection of the symptoms of chronic LBP is very sensitive and complex as it may conflict with the symptoms of kidney diseases. Hence, it is very crucial to classify chronic LBP with a better and promising classification performance.

Since the 1990s, the developed countries have evolved evidence-based treatment guidelines for LBP cases to efficiently differentiate severe diseases, to establish effective and highly safe treatment strategy, and to reduce the risk of transiting into chronic LBP. The clinical approach for the treatment of the LBP can be divided

A. Al Imran (✉) · R. Mohammad
American International University-Bangladesh, Kuratoli, Dhaka 1229, Bangladesh
e-mail: abdalimran@gmail.com

R. Mohammad
e-mail: rafeedmohammad1107@gmail.com

Md. Rifatul Islam Rifat
Rajshahi University of Engineering & Technology, Kazla, Rajshahi 6204, Bangladesh
e-mail: irifat.ruet@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Proceedings of International Joint Conference on Computational Intelligence*, Algorithms for Intelligent Systems, https://doi.org/10.1007/978-981-13-7564-4_39

into three chapters: Initial Assessment Methods, Clinical Care Methods, and Special Studies with Diagnostic Considerations [2].

If the LBP symptoms can be predicted at the primary period, it indisputably helps the doctors to prevent the LBP transiting into a chronic state. Many prominent researchers already tried to predict LBP by applying several machine learning techniques. Since LBP is a pain disorder, the dataset related to LBP usually composed of many spinal parameters and hence, it is indispensable to identify the most impactful parameters. One of the most popular machine learning approaches for identifying the most impactful features is feature selection. Additionally, a feature selection process can improve the performance of a model, which makes it time efficient and less complex. Also, a large number of alternative approaches have been developed over the past decades that can be used to improve the classification performance.

The primary aim of this paper is to figure out the best subset of features that significantly influences the classification performance of LBP. To serve this purpose, we have proposed a Genetic Algorithm (GA)-based feature selection technique to figure out the best feature subset from a larger feature space. To evaluate the impact of our proposed methodology, we have applied it to predict LBP symptoms by using a bunch of supervised machine learning techniques. We have conducted two experiments in this study: one is without applying any feature selection approach and another is after applying our proposed feature selection approach. To evaluate the classification performance of the models, we have used five evaluation metrics: accuracy, precision, recall, f1 score, and Area Under the ROC Curve (AUC). To evaluate the significance of our study, we have compared our experimental results with state-of-the-art performance.

2 Related Works

Over the past few decades, numerous studies have been conducted in the discipline of LBP. However, there exist relatively few studies in the area that are concerned with the computer-aided systems and machine learning techniques. Among these studies, some authors applied several types of machine learning techniques to diagnose and assist the medical personnel by predicting the LBP symptoms, selecting the optimized features for diagnosis, and building support system for the patients. This section briefly discusses some of the latest and relevant studies.

The authors in [3] identified the most significant physical parameters that contribute to spinal abnormalities by using unsupervised machine learning approach named Principal Component Analysis (PCA). After identifying the parameters, they predicted the spinal abnormalities by applying different supervised machine learning techniques, namely k-Nearest Neighbors (KNN) and Random Forest (RF)-based on the identical dataset that has also been used in this paper. In their experiment, they used two different train–test ratios for data partitioning: one is 80:20 and another one is 70:30. For the performance measurement, they used four evaluation metrics such as accuracy, sensitivity, specificity, and precision. From their experiment, they

found higher accuracy for the RF model with 30% test data (=79.57%) than the 20% test data (=79.03%). Also, they implemented three types of KNN algorithm (KNN, weighted triangular, and weighted rectangular) on the dataset for ten times and averaged the results of these experiments. Unlike RF, KNN performed better accuracy with 20% test data (=85.32%) rather than with 30% test data. Since KNN outperformed the RF model, they used it for data validation by using a different dataset from UCI machine learning repository and obtained a validation accuracy of 86.13%.

Detecting and supporting the patient with LBP is one of the challenging tasks in medical science as it becomes hardly detectable for some patients because of some overlapping symptoms with other diseases. The authors in [4–6] applied machine learning techniques to resolve this complex diagnosis issue. Nijeweme-d'Hollosy and Velsen [4] performed a study to assess the possibility of using machine learning in the design of a Clinical Decision Support System (CDSS) to support patients with LBP. For this purpose, they evaluated three classification models, namely Decision Tree, Random Forest, and Boosted Tree. In their study, the training dataset consisted of 1288 fictive cases of LBP and the testing dataset was constructed by a set of real-life cases. To compare the models, they considered 5 the performance metrics such as accuracy, kappa score, sensitivity, specificity, and precision. In their study, they reported the training accuracy with 70%, 69%, and 72% for the models Decision Tree, Random Forest, and Boosted Tree, respectively. And also, it reported the testing accuracy with 71%, 53%, and 71% for the three models, respectively. They found the Boosted Tree model as the best performing model with the highest accuracy score. However, in their study, no feature selection process was performed to improve the performance of the model. A similar kind of research was performed where the authors developed a generic text mining and decision support framework to detect chronic LBP from clinical narratives [5]. To prepare the dataset, they analyzed the encounter notes, which contained 7 years of unstructured narrative text data recorded by the primary care provider. The dataset they used was composed of 34 instances, which included the lab results, medical procedures, medications, and diagnoses including free-text data for social history, medical history, and clinical encounters. They applied four machine learning algorithms such as BernoulliNB, MultinomialNB, LinearSVC, and Perceptron to classify the cases of LBP for each patient. The cases were referred to as disc pain, compressed nerve pain, symptomatic spinal stenosis, facet joint pain, or without LBP. They found that the LinearSVC model outperformed the other models with 100% sensitivity and specificity, whereas the perceptron model produced the lowest result. A similar kind of study was performed in [6] where the authors tried to classify the chronic LBP by using the Structural MRI Data. They extracted brain Gray Matter (GM) density from MRI scans of 47 patients with chronic LBP and 47 healthy controls. Their study suggested that the pathology of cLBP involves changes in GM, which appeared throughout the distributed area within the brain. Because of the limited data sample, they applied Leave-Pair-Out Cross-Validation (LPOCV) technique for 100 times and the cLBP was classified with an accuracy of 76% by the analysis of SVM.

Nafiu [7] evaluated the feasibility and applicability of the different kernels of Support Vector Machine (Linear, Quadratic, Polynomial, and RBF) to classify LBP patients who were engaged with the functional restoration rehabilitation program. In their study, they applied a feature selection algorithm named Sequential Floating Forward Selection (SFFS) and found that the quadratic kernel outperformed the other kernels with an accuracy of 96.67%. To evaluate the models, they used accuracy, specificity, sensitivity, and Area Under the Curve (AUC) as performance metrics.

In addition, there exists a small number of empirical studies that used regression technique in the field of LBP. Reimer [8] carried out a study for prediction of treatment response to tapentadol for the patients with chronic low back pain. In this study, the authors attempted to identify predictors to predict the response to tapentadol treatment based on clinical pretreatment characteristics and used 46 baseline co-variables for the purpose of prediction analysis. Using multivariable regression (linear or logistic regression), they identified a set of potential predictors. They applied three selection processes (forward, backward and lasso) on resampled datasets via bootstrapping and to characterize the variables, they applied the F-change-test for linear models, as well as the likelihood test for the logistic models and this significantly improved their prediction. The analysis found that two alternative parameter quality of life and functionality were more relevant for response prediction.

From the above discussion, we can observe that few researchers applied machine learning techniques in LBP research. Different types of datasets had been used in their research studies such as MRI data and medical text narrative. However, there exists only one study [3] that used the dataset, composed of spinal measurements, that is identical to the dataset of this study. So, it is clear that there still exists a lot of research gaps in this domain. Especially, we can see that no significant work has been performed on feature selection, whereas selecting the best features from a dataset is very crucial in any applied machine learning study since it significantly influences the classification performance of the models. This study aims to cover this gap by proposing a GA-based feature selection approach to enhance the classification performance of LBP and also reduce the computational power and time compared to other studies.

3 Data Description and Preprocessing

3.1 Data Collection

In this study, we have worked with a medical dataset on LBP which has been collected from Kaggle repository [9]. The dataset is composed of 310 observations with 12 features such as pelvic incidence, pelvic tilt, lumbar lordosis angle, sacral slope, pelvic radius, degree spondylolisthesis, pelvic slope, direct tilt, thoracic slope, cervical tilt, sacrum angle, scoliosis slope, and a class attribute. The class attribute

column contains a binary class that indicates either the symptoms of LBP is "*normal*" or "*abnormal*". Among the 310 observations, 33.3% (=100) of them belongs to "normal" class and 67.7% (=210) of them belongs to "abnormal" class.

3.2 Outlier Detection and Handling

An outlier can be defined as a point that lies very far from most of the data points in a distribution. The distance can be measured with respect to a threshold, usually a number of times the standard deviation. Outliers in the training data significantly affect the learning and classification performance of the model. Hence, it is very crucial to handle the outliers carefully. We have performed a comprehensive outlier analysis on this dataset. The analysis is shown in the following Fig. 1.

From Fig. 1, it can be easily identified that six features: "*pelvic_incidence*", "*pelvic_tilt*", "*lumbar_lordosis_angle*", "*sacral_slope*", "*pelvic_radius*", "*degree_spondylolisthesis*" contain outliers. It is found that a total number of 8 outliers exists in these features.

For outlier detection, we have used the Interquartile Range (IQR) formula. If Q_1 and Q_3 be the first and third quartile, respectively, then the outliers can be detected by applying the following rules:

$$IQR = Q_3 - Q_1 \tag{1}$$

$$Lower\,fence = Q_1 - 1.5 * (IQR) \tag{2}$$

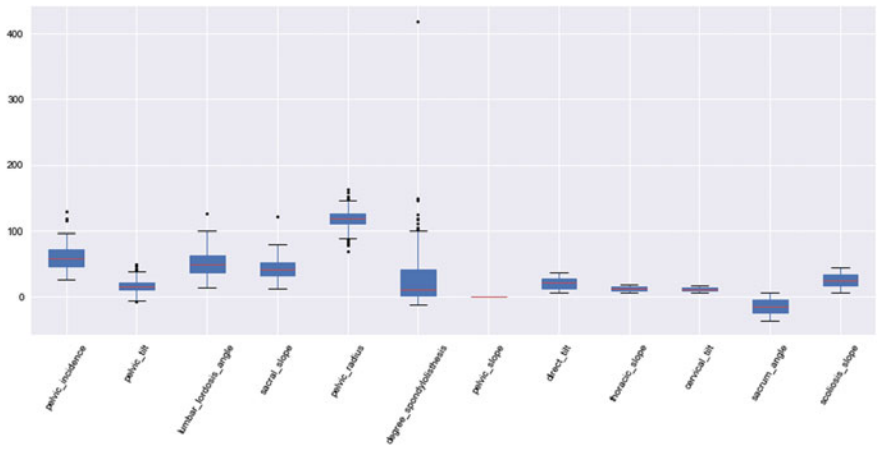

Fig. 1 Visualization of the outliers of different features

$$Upper\ fence = Q_3 + 1.5 * (IQR) \tag{3}$$

Here, the data point that falls outside the lower and upper fences is referred to as an outlier. Figure 1 shows the visualization of outlier analysis of different features in this dataset.

Since the dataset we have used contains a small number of observations, it is not an ideal choice to remove the outliers. Considering the fact, we have replaced the outliers with the median value of the corresponding features.

4 Methodology

In this study, we have maintained a proper workflow for our experiment. The workflow starts with data partitioning followed by feature selection with our proposed GA-based approach, classification with seven different classifiers from four different genres such as generalized linear models (Logistic Regression, Ridge Regression, and Naive Bayes), tree-based models (decision tree, random forest), distance-based (KNN), and kernel-based (SVM) models, and finally ends with the evaluation of the models. All the implementations in this study have been performed using the Python programing language [10] and machine learning library scikit-learn [11].

4.1 Data Partitioning

In a machine learning experiment, data partitioning is an exigent task as the ratio of partitioning can affect the experimental results. In our study, we have applied the stratified tenfold cross-validation technique as our dataset is formed with a small number of instances and also is an imbalanced dataset. In stratified tenfold cross-validation, the dataset is partitioned into 10 equal folds where it uses the proportional representation for each type of class labels in different folds. Of the 10 folds, a single is retained as validation data and the remaining 9 are used as the training data. This validation procedure is repeated 10 times where each of the 10 segments are used exactly once as validation data and finally, the 10 results are averaged to produce a single estimation.

4.2 Feature Selection

Feature selection is the process of identifying and removing unnecessary, irrelevant, and redundant features that do not contribute to or decrease the performance of the predictive model. The primary purpose of performing feature selection is to select the best subset of features that are most relevant for a predictive model. There

are numerous algorithms and techniques available for performing feature selection, however, different feature selection techniques perform differently depending on the dataset. In this study, we have proposed and implemented a Genetic Algorithm-based feature selection approach to find the best feature subset for the LBP dataset.

Genetic Algorithm-Based Feature Selection. Genetic Algorithm (GA) is an evolutionary algorithm [12] developed by John Holland inspired by Darwin's principle of survival for the fittest theory. It is a heuristic search method that is widely used in the field of artificial intelligence for optimization problems. The genetic algorithm produces new generations by repeatedly modifying a population of individual solutions. It uses three main types of rules at each step to create the next generation from the current population which is selection, crossover, and mutation. At each step, the genetic algorithm selects individuals according to their level of fitness from the current population to be parents and uses crossover and mutation rules to produce the children for the next generation. The population evolves toward an optimal solution after a number of successive generations. In this study, we have taken a standard genetic algorithm [12] with the rank-based selection strategy. For the implementation, we have used the Distributed Evolutionary Algorithms in Python (DEAP) [13] framework. The use of DEAP framework enabled our implementation to work with parallelization mechanisms.

The pseudocode of our implementation of the genetic algorithm for feature selection is given as follows:

```
Input: x, y, n_population, n_generation, cxpb, mutpb, cv
Output: best_feature_subset

population ← InitializePopulation(n_population)
EvaluatePopulation(n_population)
best_feature_subset ← GetBestIndividual(population)
while (repeat until n_generation)
      parents ← SelectParents()
      children ← Ø
      for (parent1, parent2 parents)
          child1, child2 ← Crossover(parent1,
                                     parent2, cxpb)
          children ← Mutate(child1, mutpb)
          children ← Mutate(child2, mutpb)
      end for
   EvaluatePopulation(children)
   best_feature_subset ← GetBestIndividual(children)
   population ← Replace(population, children)
end while
return (best_feature_subset)
```

The input parameters are discussed as follows:

- *x*: input data that includes all the training features
- *y*: the target variable for classification
- *n_population*: the population size
- *n_generation*: the number of generations we want
- *cxpb*: the probability of mating two individuals
- *mutpb*: the probability of mutating an individual
- *cv*: the value of k for k-fold cross-validation in the fitness function.

In our proposed approach, each of the individuals in the population space represents a candidate solution to the best feature subset. The feature subset is represented in a binary vector of dimension n (where n is the total number of features). In the binary vector, the bit 1 indicates that the corresponding feature is selected and the bit 0 indicates that the corresponding feature is not selected. The fitness of an individual is determined by applying an AdaBoost classifier over the selected subset of features with k-fold cross-validation. The reported results for each generation are based on k-fold cross-validation for each classification task in the fitness function. In our experiment we have used the values 20, 20, 0.5, 0.2, and 10 for the parameters *n_population, n_generation, cxpb, mutpb,* and *cv*, respectively.

4.3 Classification Algorithms

Logistic Regression. Logistic regression [14] is one of the most popular supervised machine learning technique for solving classification problems and used for the purpose of prediction when the target variable is a binary value, multi-category nominal, or ordinal. Several link functions have been used for modeling binary and ordinal dependent variable and most commonly used one is logit function, and is defined as

$$g(\pi) = \left(\frac{\pi}{1-\pi}\right) \tag{4}$$

In an attempt to estimate the probability of the events of the dependent variable, logistic regression used the maximum likelihood method in order to solve the model parameters.

Ridge Regression. Ridge regression [15] is basically a regularized linear regression technique that is mostly used to solve the multi-collinearity problem in OLS models through the incorporation of the shrinkage parameter, λ. This machine learning algorithm minimizes the impact of the irrelevant features on the trained model. This technique prevents the overfitting and under-fitting by using a user-defined matrix, Tikhonov matrix, that allows the algorithm to prefer a certain solution over others. However, the estimate of the ridge regression can be proceeded by adding a small value λ, that is a positive value less than 1 (usually less than 0.3), to the diagonal elements of the correlation matrix.

$$\beta_{ridge} = (X_T X + \lambda I)^{-1} X_T Y \tag{5}$$

We have used ridge regression as a classifier by coding the response labels as 0 and 1 and fitted the regression model as normal.

Gaussian NB. Gaussian NB [16] is a probabilistic classifier based on Bayes' theorem and also a supervised learning algorithm that uses the method of maximum likelihood for parameter estimation. Gaussian Naive Bayes (GNB) implements the classification by the conditional probability,

$$P(x_i|y_k) = \frac{1}{\sqrt{2\pi\sigma_{\mathrm{k}}^2}} e^{-\frac{(x_i-\mu_k)^2}{2\sigma_k^2}} \tag{6}$$

where μ_k and σ_{k}^2 be the mean and variance, respectively, of feature values x_i associated with class y_k and estimated by maximum likelihood. One of the most important advantages of this model is it can work with the missing value in the dataset.

Decision Tree. Decision Tree [17] is one of the most popular machine learning algorithms, which belongs to the family of supervised learning algorithms. The decision tree is a structure that includes a root node, branches, and leaf nodes. For split calculation, this algorithm provides two quality measures; the Gini index and the gain ratio. In our study, we have used the entropy as the split criteria. If training dataset S is split into k partitions and S_i be the subset where j be the possible values of attribute A then,

$$SplitEntropy(S, A) = \sum_{j=1}^{k} \frac{|S_i|}{|S|} \log \frac{|S_i|}{|S|} \tag{7}$$

Random Forest. Random Forest [18] is an ensemble learning method that is used for both classification and regression problems. It randomly finds the root node and split the feature nodes and use randomly created decision trees to predict the outcome. The splitting operation is performed by using Gain Ratio, Information Gain, or Gini Index. This algorithm can be used for identifying the most important features from the training dataset.

K-Nearest Neighbors (KNN). KNN [19] is one of the most popular learning approaches in data mining and is a nonparametric method used in classification, statistical estimation, and pattern recognition. It uses a distance function to compute the distances between the entities to classify an entity according to its k number of closest entities. The most prominent distance functions that are used in KNN are: Euclidean distance, Manhattan distance, and Minkowski distance. However, in this study, the Minkowski distance has been used as a metric that can be defined by

$$D(x, y) = \left[\sum_{i=1}^{k} (|x_i - y_i|)^p \right]^2 \tag{8}$$

where x is the data point from the dataset and y is the new data point that needs to predict.

Support Vector Machine (SVM). SVM [20] is a supervised learning method used to analyze the data and recognize the patterns and it is developed to solve the binary classification problems. During a training phase, SVM transforms the original training data into a higher dimensional feature space by using a nonlinear mapping. Then, it searches for an optimal separating hyperplane to separate the patterns belonging to different classes in high dimensional feature space. In the test phase, unknown samples are classified based on the position with respect to the hyperplane. A separating hyperplane can be written as

$$\boldsymbol{W} \cdot \boldsymbol{X} + b = 0 \tag{9}$$

where $\boldsymbol{W} = \{w_1, w_2, \ldots\ldots w_n\}$ is a weight vector and b is a scalar. In this study, we have used the linear kernel function which is given by

$$K(\boldsymbol{X}_i.\boldsymbol{X}_j) = \boldsymbol{X}_i\boldsymbol{X}_j \tag{10}$$

4.4 Evaluation

To evaluate the performance of the classifiers that are applied to the dataset, we used five different evaluation metrics, as follows:

1. $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$
2. $Precision = \frac{TP}{TP+FP}$
3. $Recall = \frac{TP}{TP+FN}$
4. $F1score = 2 \times \frac{(recall.precision)}{(recall+precision)}$
5. Area Under the Curve (AUC): AUC is the plot of Sensitivity versus Specificity at different points in the range [0, 1] and used for binary classification problem.

Where, True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are obtained from the confusion matrix.

Since our dataset is an imbalanced dataset, we have mainly focused on the f1 score and AUC in order to evaluate the performance of the models [21].

5 Results and Analysis

In this study, seven classification algorithms such as Logistic Regression, Ridge Classifier, Gaussian Naive Bayes, Random Forest, Decision Tree, k-Nearest Neighbors (KNN), and Support Vector Machine (SVM) have been tested on the dataset for the purpose of the prediction of chronic LBP. In this study, we have performed two

Table 1 Performance of the different models before feature selection

	Accuracy	Precision	Recall	F1 score	AUC
Logistic regression	0.826	0.880	0.862	0.870	0.897
Ridge classifier	0.813	0.858	0.871	0.862	0.897
Gaussian Naive Bayes	0.768	0.909	0.733	0.810	0.873
Random forest	0.794	0.858	0.838	0.846	0.862
Decision tree	0.755	0.828	0.814	0.818	0.722
K-nearest neighbors	0.813	0.888	0.833	0.858	0.884
Support vector machine	0.826	0.884	0.857	0.869	0.916

Table 2 Performance of the different models after GA-based feature selection approach

	Accuracy	Precision	Recall	F1 score	AUC
Logistic regression	0.826	0.880	0.862	0.869	0.922⇑
Ridge classifier	0.835⇑	0.873⇑	0.890⇑	0.880⇑	0.903⇑
Gaussian Naive Bayes	0.790⇑	0.884	0.800⇑	0.837⇑	0.869
Random forest	0.829⇑	0.873⇑	0.876⇑	0.874⇑	0.907⇑
Decision tree	0.832⇑	0.847⇑	0.924⇑	0.882⇑	0.897⇑
K-nearest neighbors	0.852⇑	0.899⇑	0.881⇑	0.889⇑	0.899⇑
Support vector machine	0.848⇑	0.894⇑	0.881⇑	0.887⇑	0.922⇑

Here, ⇑ indicates the improvement in performance of the models after feature selection

experiments: one is without feature selection and another one is with our proposed GA-based feature selection approach.

Tables 1 and 2 shows the classification performance of the models with different evaluation metrics.

From Tables 1 and 2, it is apparent that there is a significant improvement in the performance of the models after the application of the GA-based feature selection. Interestingly, the values of each evaluation metric have been improved for all the models except Logistic Regression and Gaussian Naive Bayes. In the case of Logistic Regression, only the AUC score has increased by 0.025 after applying the feature selection. However, the other remaining values have not changed significantly. The application of GA-based feature selection approach causes a significant average increment of accuracy, precision, recall, f1 score and AUC for all of the classifiers by 3.1%, 0.64%, 4.37%, 2.64%, and 3.83% respectively.

Figure 2 shows the comparison of the performance of the different models before and after the feature selection.

From Fig. 2, it can be observed that the k-Nearest Neighbors outperforms the other models with the highest accuracy (=85.2%), precision (=89.9%), and f1 score (=88.9%). Moreover, in terms of recall, Decision tree and in terms of AUC, Logistic Regression and SVM yields the highest score.

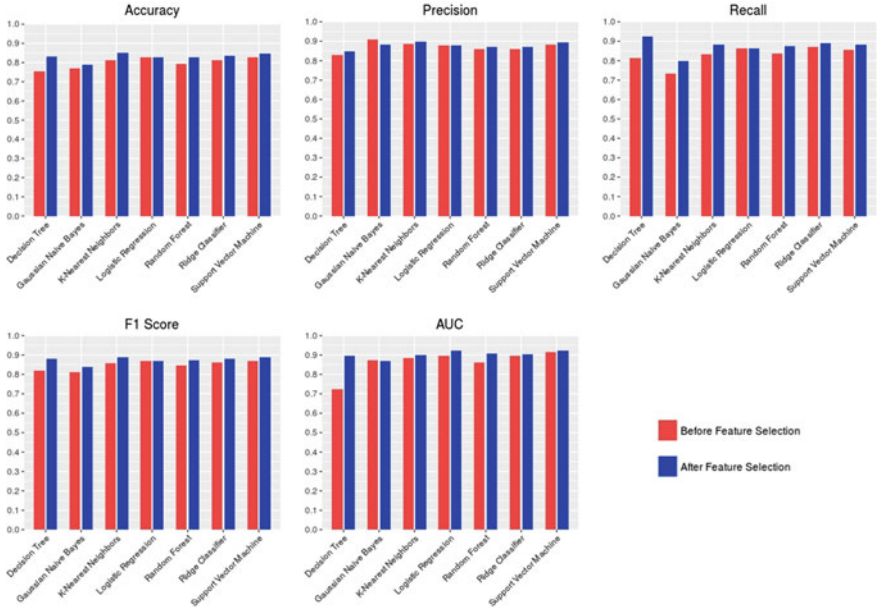

Fig. 2 Performance plot of different classifiers for different evaluation metrics

The GA fitness scores up to 20th generation starting from 0 are presented in Table 3.

We have already mentioned in Sect. 2 that the authors in [3] used the identical dataset in their experiment for the purpose of prediction of LBP. By considering the paper as state of the art, Table 4 shows the comparison between the classification performance of their model and the classification performance that has been obtained in our study in terms of KNN and Random Forest models.

In their study, they used the PCA approach to identify the significant physical parameters, whereas, we have used the GA-based feature selection approach in our study to identify the most significant features to improve the classification performance of the models. From Table 4, it can be observed that in terms of Random Forest, our model outperforms the above mentioned state-of-the-art model for all the three metrics and also outperforms their KNN model only for recall. Since they used the PCA approach, a limitation exists in their model that the model requires more computational power when it will be applied to a large data set, whereas our feature selection approach selects the features, and hence saves computational power.

Figure 3 shows the permutation importance of the selected features in case of LBP symptoms prediction from the Random Forest model. Random Forest permutation importance computes the importance of a variable by recording a baseline accuracy through passing a validation set to the classifier. The importance of a variable is the difference between the baseline and the drop in overall accuracy caused by permuting

Table 3 GA fitness scores

Gen	Nevals	Avg	Std	Min	Max
0	20	0.71565	0.06201	0.59032	0.80968
1	9	0.76774	0.03579	0.66452	0.80968
2	7	0.79613	0.01631	0.76129	0.82581
3	10	0.80661	0.01192	0.78387	0.82581
4	15	0.80726	0.02403	0.71613	0.82581
5	14	0.81548	0.01170	0.78387	0.82581
6	7	0.82307	0.00275	0.81613	0.82581
7	9	0.82468	0.00185	0.82258	0.82903
8	11	0.82210	0.01479	0.76452	0.82903
9	13	0.82613	0.00576	0.80323	0.82903
10	7	0.82548	0.01131	0.78710	0.83226
11	9	0.82419	0.01699	0.76452	0.83226
12	15	0.82048	0.02732	0.71613	0.83226
13	10	0.82952	0.01052	0.78387	0.83226
14	14	0.83226	0.00000	0.83226	0.83226
15	15	0.83210	0.00070	0.82903	0.83226
16	14	0.83048	0.00703	0.80000	0.83226
17	14	0.82339	0.01298	0.79032	0.83226
18	13	0.82984	0.00706	0.80323	0.83226
19	14	0.82790	0.01128	0.78710	0.83226
20	10	0.83145	0.00352	0.81613	0.83226

Table 4 Comparison between the performance of this study and state of the art

	State-of-the-art performance		Best performance in this study	
	KNN	Random forest	KNN	Random forest
Accuracy	0.85	0.79	0.85	0.82⇑
Precision	0.93	0.82	0.90	0.87⇑
Recall	0.84	0.87	0.88⇑	0.88⇑

that column. The results of permutation importance are more reliable and appreciated in the academia and industry.

From Fig. 3, the most important finding is, "*degree_spondylolisthesis*" is the feature from which the model learned the most approximately more than twice than the others. So, it can be concluded that "*degree_spondylolisthesis*" is one of the most impactful features among all of the selected features.

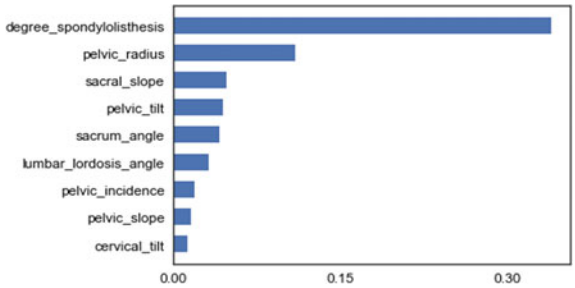

Fig. 3 The relative importance of the selected features

6 Conclusion

The most obvious finding to emerge from this study is that the application of the genetic algorithm-based feature selection approach can improve the classification performance for LBP. For the purpose of prediction, seven classification algorithms: Logistic Regression, Ridge Classifier, Gaussian Naive Bayes, Random Forest, Decision Tree, k-Nearest Neighbors, and Support Vector Machine (SVM) were tested on a LBP dataset. The k-Nearest Neighbors outperforms the other models with the highest accuracy (=85.2%), precision (=89.9%) and f1 score (=88.9%). The classification algorithms were applied on the dataset for two times: before the feature selection and after the feature selection. The application of GA-based feature selection approach causes a significant average increment of accuracy, precision, recall, f1 score, and AUC for all of the classifiers by 3.1%, 0.64%, 4.37%, 2.64%, and 3.83% respectively. We have also performed an empirical comparative analysis of our obtained results with the performance of the state of the art. Among all the selected features, the "*degree_spondylolisthesis*" was found as the most impactful feature by the Random Forest permutation importance.

References

1. Rayburn D (2007) Let's get natural with herbs. Ozark Mountain Publishing, Incorporated, Huntsville
2. Bigos S (1994) Acute lower back problems in adults; Rockville MD: agency for health care policy and research, clinical practice guideline no. 14, 95-0642
3. Abdullah A, Yaakob A (2018) Prediction of spinal abnormalities using machine learning techniques. In: 2018 international conference on computational approach in smart systems design and applications (ICASSDA). IEEE, pp 1–6
4. Nijeweme-d'Hollosy W, Velsen L (2018) Evaluation of three machine learning models for self-referral decision support on low back pain in primary care. Int J Med Inform 31–41
5. Judd M, Zulkernine F (2018) Detecting low back pain from clinical narratives using machine learning approaches. In: 2018 Springer international conference on database and expert systems applications, pp 126–137
6. Ung H, Brown J (2012) Multivariate classification of structural MRI data detects chronic low back pain. Cereb Cortex 1037–1044

7. Nafiu J (2017) A machine learning-based surface electromyography topography evaluation for prognostic prediction of functional restoration rehabilitation in chronic low back pain. Spine 42(21):1635–1642
8. Reimer M (2017) Prediction of response to tapentadol in chronic low back pain. Eur J Pain 322–333
9. Kaggle. https://www.kaggle.com/. Accessed 11 Oct 2018
10. Python Core Team (2015). Python: a dynamic, open source programming language. Python software foundation. https://www.python.org/. Accessed 12 Oct 2018
11. Pedregosa F, Varoquaux G (2011) Scikit-learn: machine learning in python. J Mach Learn Res 2825–2830
12. Forrest S (1993) Genetic algorithms: principles of natural selection applied to computation. Science 261(5123): 872–878
13. Fortin F, Rainville F (2012) DEAP: evolutionary algorithms made easy. J Mach Learn Res 2171–2175
14. Cramer JS (2002) The origins of logistic regression
15. Hoerl A, Kennard R (1970) Ridge regression: biased estimation for nonorthogonal problems. J Technometrics 55–67
16. Rish I (2001) An empirical study of the naive Bayes classifier. In: 2001 IJCAI workshop on empirical methods in artificial intelligence, vol 3, no 22, pp 41–46
17. Quinlan JR (1986) Induction of decision trees. J Mach Learn 81–106. Springer
18. Breiman L (2001) Random forests. J Mach Learn 5–32. Springer
19. Cover T, Peter H (1967) Nearest neighbor pattern classification. J IEEE Trans Inf Theory 21–27. IEEE
20. Hearst M, Dumais S (1998) Support vector machines. J IEEE Intell Syst Appl 18–28. IEEE
21. He H, Garcia E (2008) Learning from imbalanced data. J IEEE Trans Knowl Data Eng 1263–1284. IEEE