Performance Evaluation for Building a Predictive Model on Diabetic Health Record using PySpark

Rakesh S Raj1, Dr. Kusuma M2

1Assistant Professor, Department of Information Science & Engineering, Adichunchangiri Institute of Technology, Chickamagluru-577102, India

2Professor, Department of Information Science & Engineering, DSATM, Kanakpura Road, Bengaluru-560071, India

Abstract: The technique of extracting interesting patterns from massive data sets is known as data mining. Predictive analytics uses different statistical and machine learning algorithms on historic data to identify future outcomes. This work focuses on the classification algorithms such as Logistic Regression, Naïve Bayes, Gradient Boosting Technique, and Random Forest. A predictive model is built using these techniques on diabetic health records using spark machine learning libraries. We calculate various performance metrics using these algorithms and determine a better algorithm based on performance metrics to build a diabetic prediction model using Apache PySpark.

Index Terms – Data mining, Analytics, Machine Learning, Classification, PySpark

I. INTRODUCTION

In recent time, data mining and data analytics has become a part of daily life. They are incorporated in almost every aspect of a wide variety of applications like social networks, trade, e-commerce, sports, retail and entertainment, politics, and health care. The results derived from data analysis provide useful insights for future decision-making support systems.

Data analytics is classified into predictive, descriptive, diagnostic, perspective, and cognitive analytics. Predictive analytics makes use of historical data to anticipate what will happen in the future. Descriptive analytics is a type of analytics that describes or summarizes what has happened in the past. Diagnostic analytics looks back in the time to figure out why something has happened. Prescriptive analytics is a type of prediction which can be used to recommend a different course of action. Finally, cognitive analytics uses intelligent technologies like artificial intelligence; deep learning models to build models that think and behave like a human brain.

The health care industry produces a large volume of clinical data that are generated by hospitals, laboratories, pharmacies, and medical research institutes. But most of the large volume generated is either maintained poorly or unstructured. Also, different hospitals have different ways of storing data. Hence, the big challenge in health care is to maintain these health records digitally which can be very useful for data analytics.

This paper focuses on popular classification algorithms: logistic regression, Naïve Bayes, gradient boosting technique (GBT), and random forest. The following are the major goal of this work is

i. Using PySpark MLLib, create a predictive model for diabetic data.
ii. Compare the accuracy of the algorithms.
iii. Based on the comparison determine which technique is better for data analytics.

II. Background Study

2.1 Data Mining Techniques

There are many strategies used to solve different business problems and provide different insights. Based on the solution, one must be aware of the technique which derives better insights. Data mining techniques are broadly classified into Classification, Regression, Clustering, Association rules, Anomaly, or Outlier detection [1].

Classification is used to classify the object or the features based on class labels. Clustering is typically used to create groups that contain similar objects. Regression is the process of identifying the relationship between variables or objects [2]. Association is used to identify interesting relations among different objects in large data. Outlier detection is used to find anomalies or noise or less interesting patterns in the dataset.
2.2 Data Mining Process

Major steps in data mining include data collection, data pre-processing, and data analysis and evaluation. The process is shown in Figure 1.

Data collection is a process of collecting data from available resources. Data pre-processing involves cleaning the data, data integration, data reduction, and data transformation [3]. Data analytics uses intelligent methods, such as machine learning, deep learning, etc algorithms, which produce useful and interesting insights into the data.

![Figure 1. Data Mining Process](image)

Evaluation includes finding the accuracy and checking for the correctness of the obtained results.

2.3 Apache PySpark

Apache spark is an open-source tool used for processing a large amount of data [4]. PySpark provides an interface for apache spark in python. MLLib is built on top of the apache spark framework. Spark is a computational engine used for big data analytics which supports Python. It provides scalable machine learning libraries which can be used for machine learning.

Various methods for binary classification, multiclass classification, and regression analysis are supported by the spark.mllib package. Random Forest, Naive Bayes, Decision Tree, and other classification algorithms are some of the most popular.

2.4 Related Works

Woldemichael, Fikirte Girma, et al proposed a method to predict diabetes using backpropagation algorithm, J48, Naïve Baye's, and support vector machine using R language and R studio [9]. In this work, the backpropagation algorithm gave 83.11% accuracy, 86.11% sensitivity, and 76% specificity.

Sisodia, Deepti, and Dilip Singh, et al work focus on three machine learning algorithms namely decision tree, Naïve Bayes, and SVM [10]. This model shows Naïve Bayes outperforms other methods by achieving 76.30% accuracy.

For diabetes prediction, the authors used an Artificial Neural Network [11]. They gathered data on 250 diabetes patients aged 25 to 78. To train data, they used MATLAB. They used BFGS, Quasi-Newton, Bayesian Regulation, and Levenberg–Marquardt algorithms to perform regression analysis. They discovered that Bayesian Regulation had an accuracy rate of 88.8%.

M. A. Sarwar, N. Kamal, W. Hamid, and M. A. Shah developed a diabetes prediction model employing six machine learning algorithms in a Hadoop/MapReduce environment, including support vector machine, k-nearest neighbour, logistic regression, decision tree, and Naive Baye's. With an accuracy of 77%, they determined SVM and KNN to be the top approaches [12].

The author of [13] compiled 318 medical data containing nine nominal features, including the patient's gender, age, smoking status, history of hypertension, renal problem, cardiac problem, and eye problem, to construct a new model for the treatment of type 2 diabetic patients. The J48 method was utilised in the model, which produced an accuracy of 70.8 percent and a ROC rate of 0.624.

The authors were able to predict diabetes using supervised and unsupervised learning [14]. To find a superior machine learning prediction method, they used the software package WEKA. Finally, they came to the conclusion that the ANN or Decision tree is the best method for diabetes prediction.
III. Methodology

The overall workflow of the work is shown in Figure 2.

![Figure 2. Methodology](image)

3.1 Data Set

The data set used in this work is the Pima Indian diabetic dataset shown in Figure 3, which is available in an open-source data repository [15]. Description of attributes is given in Table 1. The class label is the “Outcome” attribute. “Outcome value =1” means the person has diabetes, while “Outcome value =0” means the person does not.

Table 1 Description of attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pregnancies</td>
<td>Number of pregnancies (Numeric)</td>
</tr>
<tr>
<td>Glucose</td>
<td>Glucose concentration (Numeric)</td>
</tr>
<tr>
<td>Blood Pressure</td>
<td>Diastolic blood pressure (mm Hg)</td>
</tr>
<tr>
<td>Skin Thickness</td>
<td>Triceps skinfold thickness (mm) (Numeric)</td>
</tr>
<tr>
<td>Insulin</td>
<td>Two-hour serum insulin (mu U/ml) (Numeric)</td>
</tr>
<tr>
<td>Bmi</td>
<td>Body mass index (Numeric)</td>
</tr>
<tr>
<td>Diab_pedi</td>
<td>Diabetes pedigree function (Numeric)</td>
</tr>
<tr>
<td>Age</td>
<td>Age of the person in years (Numeric)</td>
</tr>
<tr>
<td>Outcome</td>
<td>Class label, True if diabetic otherwise false</td>
</tr>
</tbody>
</table>
The dataset is divided into a training sample and a test sample. 70% of total data is considered for training data and 30% for testing. Training data set is used to train the model by fitting the parameters and test data is used to evaluate the model’s performance.

3.2 Data Pre-processing and feature selection

Data pre-processing is the crucial stage in data analytics that identifies missing values, outliers, and other anomalies. Feature selection is a part of pre-processing where essential attributes are selected for further analysis. This is done by finding the correlation among the attributes [8]. The correlation is determined by the following equation:

$$ r_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2(y_i - \bar{y})^2}} $$

Where,

- $r_{xy}$ - the correlation coefficient of the linear relationship between the variables x and y
- $x_i$ - the values of x variable in a sample
- $\bar{x}$ - the mean value of x variable in a sample
- $y_i$ - the values of y variable in a sample
- $\bar{y}$ - the mean value of y variable in a sample

The correlation among the attributes obtained using equation 2 is given in Figure 4.

![Correlation among the attributes](image)

The value towards 1 indicates that any two variables are highly correlated and value 0 indicated poorly correlated. Hence correlation among attributes is in the range of 0.0 to 1.0. Figure 4 shows that none of the attributes are highly correlated. Hence, we keep all attributes for building a predictive model.

3.3 Brief Description about Classification Techniques

3.3.1 Gradient Boost Technique

The gradient boosting technique (GBT) generates multiple decision trees. These trees are weak learners [5]. Each tree produces a prediction model. Then these models are combined to generate final predictions. Figure 5 shows trees built sequentially using GBT.
3.3.2 Logistic Regression

The statistical machine learning technique, logistic regression, is used to predict the likelihood of a target variable. It uses a logistic function called the sigmoid function [6]. The sigmoid function is an S-shaped curve that maps a real-valued number to a number between 0 and 1. Logistic regression can be divided into the following types: binary, multinomial, and ordinal. We use binary classification since the target class is either diabetic (1) or non-diabetic (0).

3.3.3 Naïve Bayes

Naïve Bayes algorithm is a classification approach based on the Bayes theorem. Bayes theorem states that all the predictors are independent of each other [6, 7]. The Bayes Theorem is given by the following equation

\[ P(a|b) = \frac{P(b|a)P(a)}{P(b)} \]

Where,
- \( a, b \) = events
- \( P(a|b) \) = probability of \( a \) given \( b \) is true
- \( P(b|a) \) = probability of \( b \) given \( a \) is true
- \( P(a), P(b) \) = independent probabilities of \( a \) and \( b \)

3.3.4 Random Forest Classifier

Random forest is a classification and regression supervised learning technique. However, it is mostly employed to tackle classification issues. It uses data to create numerous decision trees, finds predictions from each tree, and then uses the majority vote to choose the best answer. Figure 6 illustrates the working of the random forest [8].

This work is concentrated to build a predictive model using logistic regression, Naive Baye’s, gradient boosting technique, and random forest classifiers. These algorithms are implemented using PySpark which supports the machine learning library, MLlib. Table 2 describes the pseudo-code.
Table 2. Pseudocode

**Step 1 Create spark session and load data**
1. Import important libraries
2. Create Spark session
3. Load data set and assign to the data frame

**Step 2 Data Pre-processing**
4. Handling missing data
5. Dealing with imbalanced data

**Step 3 Build model**
6. Split data into training and test data
7. Create the model and validate
8. Generate confusion matrix
9. Calculate performance metrics

### 3.3.6 Performance Measurement

Performance measurement is a metric that describes how the algorithm is performing on the data set. It is also used to decide which algorithm is better. This decision can be made by a 2x2 confusion matrix, given in Table 3. The accuracy of the suggested model in terms of "True Positive Rate", "True Negative Rate", "False Positive Rate", "False Negative Rate" based on data set collected is given by confusion matrix. PySpark has a built-in method i.e "model.confusionMatrix ()", where the model is the model created by different machine learning techniques.

#### Table 3 Confusion Matrix

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
<th>TP</th>
<th>FN</th>
<th>FP</th>
<th>TN</th>
</tr>
</thead>
<tbody>
<tr>
<td>TP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

True positives (TP) are the number of positive samples that the model correctly classifies as positive. True Negatives (TN) are the number of negative samples that the model properly categorized as negative. False Positives (FP) are negative samples that are wrongly classified as positive by the model. False Negatives (FN) are the positive samples that have been labeled as negative wrongly. The confusion matrix generated by different algorithms is given in below Table 4.

#### Table 4. Confusion Matrix Generated

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>TP</th>
<th>FN</th>
<th>FP</th>
<th>TN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>358</td>
<td>92</td>
<td>42</td>
<td>102</td>
</tr>
<tr>
<td>Naive Baye’s</td>
<td>399</td>
<td>182</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>GBT</td>
<td>399</td>
<td>28</td>
<td>24</td>
<td>172</td>
</tr>
<tr>
<td>Random Forest</td>
<td>376</td>
<td>74</td>
<td>47</td>
<td>126</td>
</tr>
</tbody>
</table>

Based on the confusion matrix, as shown in Table 4, we can calculate performance metrics which are given in equations below:

\[
\text{Accuracy} = \frac{(TP + TN)}{TP + TN + FP + FN}
\]  \hspace{1cm} (3)

\[
\text{Recall} = \frac{TP}{TP + FN}
\]  \hspace{1cm} (4)

\[
\text{Precision} = \frac{TP}{TP + FP}
\]  \hspace{1cm} (5)

\[
\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]  \hspace{1cm} (6)
4. Results and Analysis

Performance evaluation of the algorithms is measured by the metrics like accuracy, recall, precision, F1 score which are given in Equations 3, 4, 5, 6. The number of correctly identified samples divided by the total number of samples is the accuracy. Positive predictive value is also called as precision which determines how accurate the model is out of predicted positive and how many of them are actual positives. The ratio of true positives to total actual positives is known as recall. F1 score is a metrics that used both precision and recall to measure a model’s accuracy. The highest F1 score achievable is 1, indicating perfect precision and recall. The lowest value is 0, indicating that either precision or recall is zero. Results obtained by substituting TP, TN, FP, FN in equation 3,4,6,7 is shown in Table 5.

Table 5. Comparison of performance measures

<table>
<thead>
<tr>
<th>Metrics In %</th>
<th>Logistic Regression</th>
<th>Naïve Bayes</th>
<th>Gradient Boost</th>
<th>Random Forest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>78.33</td>
<td>69.02</td>
<td>91.65</td>
<td>80.57</td>
</tr>
<tr>
<td>Recall</td>
<td>71.52</td>
<td>88.88</td>
<td>87.75</td>
<td>72.83</td>
</tr>
<tr>
<td>Precision</td>
<td>54</td>
<td>4</td>
<td>86</td>
<td>63</td>
</tr>
<tr>
<td>F1 score</td>
<td>61.53</td>
<td>7.6</td>
<td>86.86</td>
<td>67.56</td>
</tr>
</tbody>
</table>

The comparison of the algorithms is shown in Table 6. The results show that the accuracy, recall, precision, and F1 score of GBT is better compared to other methods. The random forest technique proves to be the second-best followed by logistic regression. Naïve Baye's being the least accurate. The graphical visualization of the results is shown in Figure 7.
Table 6. Comparison of our model with earlier models

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Authors</th>
<th>Methods Used</th>
<th>Tool Used</th>
<th>Accurate Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diabetes Prediction</td>
<td>Woldemichael, Fikirte Girma, et al</td>
<td>Backpropagation, J48, Naive Baye’s, SVM</td>
<td>R Studio, Backpropagation</td>
<td>(88.11)</td>
</tr>
<tr>
<td></td>
<td>Sisodia, Deepti, and Dilip Singh et al</td>
<td>Decision Tree, Naive Bayes, SVM</td>
<td>Weka, Naive Bayes</td>
<td>(76.30)</td>
</tr>
<tr>
<td></td>
<td>Sapon, M.A., Ismail, K., Zainudin, S.</td>
<td>BFGS, Quasi-Newton Bayesian Regulation, Levenberg–Marquardt</td>
<td>MATLAB, Bayesian Regulation</td>
<td>(Accuracy:88.8)</td>
</tr>
<tr>
<td>Diabetes Prediction</td>
<td>M. A. Sarwar, N. Kamal</td>
<td>KNN, Logistic Regression, Naive Baye’s</td>
<td>Hadoop/MapReduce, SVM &amp; KNN</td>
<td>(Accuracy:77%)</td>
</tr>
<tr>
<td>Diabetes Prediction</td>
<td>Rabina, Er, and Anshu Chopra</td>
<td>ANN, decision tree</td>
<td>Weka, Decision tree</td>
<td></td>
</tr>
<tr>
<td>Diabetic Prediction</td>
<td>Our Proposed Method</td>
<td>Logistic Regression, Naive Bayes, Gradient Boosting, Random Forest</td>
<td>PySpark</td>
<td>Gradient Boosting (Accuracy:91.65)</td>
</tr>
</tbody>
</table>
5. Conclusion
Accuracy plays an important role in data analytics. The result suggests that the accuracy of logistic regression is 78.33%, Naïve Bayes is 69.02%, gradient boosting technique is 91.65% and the random forest is 80.57%. Hence gradient boosting technique tends to prove better accuracy followed by random forest when compared with the rest of the methods used. Table 6 shows the comparison of our approach with the earlier works carried out so far. The previous approach never used gradient boosting technique for diabetes prediction. Also, various works are carried out using R, weka, SPSS modeller, Hadoop MapReduce. We use apache PySpark, which is a big data analytics tool. Hence for predicting diabetes, a novel approach using apache PySpark MLib and gradient boosting algorithm can be used to build a predictive model.

References