# PERFORMANCE ANALYSIS OF META LEARNING ALGORITHMS FOR INCREASING DATASET SIZE

[1]Manuja Sharma, [2]Dr. K.L. Bansal
[1]M.Tech Student, [2]Professor
Department of Computer Science,
Himachal Pradesh University, Shimla, India.

***Abstract:*** With the emergence of technology, the availability of raw data over the network has increased unprecedentedly and this data is continuously exploding day by day. This data needs to be processed and analyzed in some efficient ways, and the information thus extracted can help serving some or the other purpose. Data mining is one of the emerging research fields that offer various techniques and methods that are again used for predicting the patterns and future trends in the available data. There are numerous techniques which are used for predicting the class in which the particular data falls in. Classification of data becomes difficult with the unbounded size and imbalance nature of data. The problems of over-fitting and bias are somewhat solved with the help of Meta learning methods (also known as Ensemble methods). This research focuses on analyzing the behavior of Meta Learning techniques, such as Bagging, Boosting and Stacking, with the help of three base classifiers, i.e. Naïve Bayes, Decision Table and J48, over the increasing size of data. WEKA tool has been used for the experimental purpose and the parameters that helped in evaluating the performance are accuracy, root mean squared error, kappa statistics, precision and recall.

***Keywords*- Data Mining, Classification, Naïve Bayes, Decision Table, J48, Ensemble Methods, Bagging, Boosting, Stacking, WEKA Tool.**

## I. INTRODUCTION

The availability of raw data over the network has increased unprecedentedly and this data is continuously exploding with each passing day. This raw data is by itself of no relevance and serves no purpose if it remains unprocessed. For improving the utilization and usage of data, various significant patterns and trends are extracted from this data with the help of data mining techniques. Data mining is known as the process of extracting useful information from the whopping amount of data. It can also be defined as the process of discovering unseen trends and patterns from the existing data and then making use of these trends and patterns for the prediction of some future trends [1]. Numerous tools for data mining are available that help in analyzing the data like WEKA, Knime, Rapid Miner etc. These tools provide us with a vast collection of methods and techniques that are used to analyze the data in suitable ways. WEKA tool has been used in this research work to analyze the performance of various Meta methods with the help of some base classifiers over the data sets of varying sizes.

### 1.1 Classification

Classification is primarily a data analysis task where the model is constructed in order to help predicting the class of data objects whose class labels are yet unspecified. It is also known as the process that helps analyzing a dataset by generating some rules for grouping which are further used for classifying the future unseen data. This is basically a two phase process:

1. Learning step: In this phase, a model is built by describing some of the predetermined set of data classes. This is the training phase where the algorithm tends to build a model by analyzing or learning from a training set made up of database tuples and the associated class labels [2].
2. Classification step: In this phase, the model which is previously extracted from the learning phase is tested with the whole new test data set in order to measure and analyze the performance of previously trained model. If the performance measures are acceptable, then the rule or the model is ready to be applied to new data tuples and if it is not so then the first step is again performed [2].

Classification techniques that have been focused in this paper:

#### 1.1.1 Naïve Bayes

Naïve Bayes classifier is a supervised learning technique which is purely based on the Bayes theorem. The simplest Bayesian classifier is Naive Bayes Classifier. The assumption made in this case is class conditional independence which means that all the variables contribute toward classification and are mutually correlated. However it's an unrealistic assumption for most datasets. Naïve Bayes requires very less computational time for training and it can be applied to very large datasets [3].

### 1.1.2 Decision Table

Rule based classification methods are those where the data is classified using a set of IF-THEN rules, and these rules are generated either from the decision trees or from the training data. For extracting the rules from decision tree:

- For each and every path from the root node to leaf node, one rule is created.
- Each splitting criterion is logically ANDed to form the antecedent.
- Leaf nodes hold the class prediction which forms the consequent.

Decision Table is a method which is used to numerically predict the data from decision tree. Decision table is a rule based classifier which is an ordered set of IF-THEN rules that are much more compact and are much easier to comprehend than that of decision trees [4].

### 1.1.3 J48

Decision Tree methods are used to create a model that predicts the value of target variables on the basis of multiple input variables. Decision trees help in classifying the instances by sorting them on the basis of feature values. A decision tree is a flowchart like structure which includes a root node, branches and leaf nodes. The feature which best divides the data would be the root node of the tree. Each internal node in this tree will denote a test on an attribute and each branch will denote the outcome of that test, while each leaf node will hold a class label. J48 is an extension of ID3 algorithm and an open source implementation of C4.5 algorithm. This algorithm tends to generate rules for predicting the class of target variable. Some of the additional features of J48 are accounting for missing values, pruning decision trees, rule derivation, continuous attribute value ranges, etc. [5].

## 1.2 Meta Learning Methods

Multiple learning algorithms are used by Meta methods together for the same task to have better prediction rate than that of the individual learning model. Meta learning is also known as committee based learning or Ensemble learning system. These methods try to construct a set of learners and combine them or we can also say that Ensemble methods train multiple learners to solve the same problem. An ensemble contains a no. of learners called as base learners. Base learners are generated by base learning algorithms such as decision tree, neural network or any other kind of learning algorithms from training dataset. The main advantages of these methods are: Reduced Variance, Reduced Bias and Improved Prediction [1].

Ensemble Learning Techniques that have been focused in this paper:

### 1.2.1 Bagging

Bagging is also called Bootstrap Aggregation. Bagging is an ensemble meta-algorithm which is designed in such a way that it improves the stability and accuracy of learning algorithms used in statistical classification and regression. This algorithm also helps in reducing variance and helps to avoid the issue of overfitting [1].

### 1.2.2 Boosting

Boosting is a robust ensemble algorithm which is capable of reducing both bias & variance, and it also helps in converting weak learners (i.e., classifiers with weak correlations to the true classification) into strong learners (i.e., well-correlated classifiers). Boosting tends to create strong classification trees because it forces new classifiers to focus on the error produced by previous ones. Boosting works aggressively by decreasing the no. training errors; when this is done, all classifiers are combined by a weighted majority vote. This process places a higher weight to incorrectly classified records while decreasing the weight of correct classifications i.e. this effectively forces resultant models to put a larger stress on misclassified records. The algorithm then computes the weighted sum of votes for each class and assigns the best classification to the record. Boosting frequently gives better models than bagging, but is not capable of parallelization; consequently, if the dataset is very large (i.e., significant number of weak learners), then boosting may not serve as the most appropriate ensemble method [1].

### 1.2.3 Stacking

Stacking is also known as stacked generalization. Stacking is similar to boosting. It is an ensemble method where the models are combined using another machine learning algorithms. The basic idea here is to train machine learning algorithms with the training dataset and then a new dataset is generated with these models. Then this new dataset is then used as an input to the combined machine learning algorithm. The difference with respect to boosting lies here in the fact that we don't have just an empirical formula for the weight function, rather we introduce a meta-level and use another model/approach to estimate the input together with the outputs of every other model to estimate the weights, or, in other words to determine which models performed well and which performed badly over this input data [1].

## II. RELATED WORK

Some of the important works where the performance of Meta Learning methods has been analyzed are:

**Eibe Frank et al. (2004)** conducted a study on the WEKA machine learning workbench which provides an environment for classification, regression, clustering, feature selection etc. The study stated that this workbench contains a collection of numerous amounts of algorithms and data pre-processing methods that can be easily used and implemented over the graphical interface. The tool helps in assisting the user by helping to extract useful information from the data and also enables the user to identify the suitable algorithms for generating the accurate predictive models [6].

**Yanmin Sun et al. (2006)** studied cost sensitive boosting for classification of the imbalanced data. The study focused on the exploration of multiple learning techniques with the aim of advancing the classification of imbalanced data. For Boosting, AdaBoost has been applied over a base classifier to reduce time and improve accuracy of classification. While accuracy improvement is a bit trivial context in terms of imbalanced data, accuracy rates doesn't matter much. This work discussed the experimental results of classification of some real world imbalanced data **[**7].

**Arvind Sharma, P.C. Gupta (2012)** experimented to predict the number of blood donors with the help of classification techniques over WEKA tool. In this study, an attempt has been made to classify and predict the number of donors as per their age and blood groups and to build a model for extracting knowledge to aid clinical decisions in blood bank sector. J48 algorithm has been used in this work and implemented over the real world data. Training and evaluations showed that the generated classification rules performed with the accuracy rate of 89.9% [8].

**Aayushi Verma, Shikha Mehta (2017)** proposed a novel ensemble approach called the "BBS method", which stands for Bagging, Boosting and Stacking, with the help of some base classifiers to classify the data of five datasets taken from the bioinformatics field. WEKA tool and Java Eclipse have been used in this study. It was observed that the new approach gave better accuracy along with lower root mean square error rate. Experimental results showed that individually Bagging performed better than that of Boosting and Stacking in terms of accuracy while it was the other way round in case of root mean square error rates; the proposed method showed better results than the individual algorithms over both parameters. As a result it was suggested that the proposed BBS method is more suitable in handling the classification problem in bioinformatics domain [9].

**Shiva Kazempour Dehkordi, Hedieh Sajedi (2017)** proposed a prescription based automatic medical diagnosis system using a stacking method. This work intended to use data mining techniques to predict what kind of physician each patient has referred to and what kind of diseases are they suffering from. For labeling the instances, a group of pharmacy students and professors has determined each patient's disease. Experiments were performed to compare the performance of different techniques and the results illustrated that the proposed Stacking Model has higher accuracy as compared to other techniques such as k-Nearest Neighbor (kNN), Naïve Bayes, and Decision Tree etc [10].

**Kuldeep Randhawa et al. (2018)** analyzed AdaBoost and Majority Voting techniques for credit card fraud detection. In this work, firstly a publicly available data is analyzed over some base classifiers and then the hybrid techniques i.e. AdaBoost and Majority Voting techniques are applied. Then, these techniques are implemented over the real world data and the results are accordingly analyzed. It was indicated that the majority voting technique achieves good accuracy rate in detecting the frauds [11].

**Pelin YILDIRIM et al. (2018)** comparatively analyzed the ensemble methods for signal classification. Four ensemble methods: Bagging, Boosting, Stacking and Voting, were analyzed with the help of five basic classification algorithms: Neural Networks, Support Vector Machines, k-Nearest Neighbor, Naïve Bayes and C4.5. In this study, the ensemble methods were applied on 14 different signal datasets and their performance was measured in terms of classification accuracy. The results showed that a Bagging based method outperformed all the other algorithms with the highest classification accuracy rate [12].

**Saba Bashir et al. (2019)** presented a study to improve the heart disease prediction. This work focused on the feature selection techniques and algorithms. Multiple datasets were used in order to improve the accuracy rates. Decision Tree, Logistic Regression, Logistic Regression (SVM), Naïve Bayes and Random Forest algorithms are used for the feature selection purpose. The results showed that higher accuracy rate was achieved in SVM and Naïve Bayes and it was suggested that Logistic regression is the best feature selection method for predicting heart diseases [13].

## III. WEKA TOOL

WEKA stands for Waikato Environment for Knowledge Analysis. WEKA tool is an influential tool in data mining. WEKA provides us a vast collection of machine learning algorithms, developed by The University of Waikato, Hamilton, New Zealand. These algorithms are directly applicable to the data or data called from the Java code. WEKA is a free software licensed under the GNU General Public License. By default, WEKA uses ARFF (Attribute Relation File Format) file for the analysis of data. Other file formats like CSV (Comma Separated Values), C4.5 data files etc. are also supported and databases using ODBC, from where data can be imported. WEKA is a vast collection of algorithms for: Classification, Regression, Clustering, Association, Data pre-processing and Visualization. WEKA provides five interfaces: Explorer, Experimenter, Knowledge Flow, Simple CLI and Workbench [14].

## IV. RESULTS AND DISCUSSION

WEKA tool has been used for analyzing the behavior of different types of data in the increasing order of their size. Different classification algorithms along with the meta learning algorithms have been analyzed using the 10-fold cross validation method.

### 4.1 Datasets

The datasets have been downloaded from the UCI repository [15] and Kaggle [16] websites. Four distinct datasets having different sizes have been chosen:

Table 4.1: Description of Datasets

| Dataset Name | Abbreviation Used | Data Types | Default Task | Instances | Attributes | Size |
|---|---|---|---|---|---|---|
| Student's Academic Performance [17] | A | Multivariate | Classification | 480 | 16 | 38KB |
| Chronic Kidney Disease [18] | B | Multivariate | Classification | 2342 | 25 | 116KB |
| Spambase [19] | C | Multivariate | Classification | 4601 | 57 | 686KB |
| Adult [20] | D | Multivariate | Classification | 48842 | 14 | 3.91 MB |

- In dataset A, students are to be classified into three numerical intervals based on their total marks/grades.
- In dataset B, prediction of the early onset of chronic kidney disease is done.
- In dataset C, mails are classified as spam or non-spam by constructing a personalized spam filter.
- In dataset D, prediction task is to determine whether a person's income exceeds 50K a year or not.

Datasets in the increasing order of their size are as follows: **A<B<C<D**

### 4.2 Parameters for Evaluation

Parameters which have been used in this work for the evaluation of various algorithms are:

#### 4.2.1 Accuracy

Accuracy is defined as percentage of number of correctly classified instances.

i.e. $Accuracy = \frac{TP+TN}{TP+TN+FN+FP}$

where, TP is True Positives
TN is True Negatives
FP is False Positives
FN is False Negatives
TP+TN is correctly classified instances
TP+TN+FP+FN is Total number of instances

#### 4.2.2 Root Mean Squared Error

Mean Absolute Error (MAE) computes the average magnitude of errors. MAE is basically the average of absolute values of difference between the predicted and absolute observations. Root Mean Squared Error (RMSE) also computes the average magnitude of errors, but the difference lies in the way that the difference between the predicted and absolute observation is squared and is then averaged over the set of observations. The square root of the computed average is referred as RMSE. The value of these types of error ranges from zero to infinity. Since the errors are squared before they are averaged, RMSE gives a relatively high weight to large errors.

#### 4.2.3 Kappa Statistics

Kappa refers to a chance-corrected measure which is calculated between classification and true classes. Such a measure is computed by taking the expected attribute from the observed values of attributes. The value is then divided by the maximum value of the attribute. Value greater than zero indicates a better performance as compared to chance.

#### 4.2.4 Precision

Precision lists the proportion of those instances which are true to a particular class divided by overall instances classified with respect to that class. It is also called the positive predictive value.
It can be represented as:

$$Precision = \frac{TP}{TP + FP}$$

***True Positives (TP):*** Number of instances predicted positive that are actually positive.

***False Positive (FP):*** Number of instances predicted positive that are actually negative.

**4.2.5 Recall**

Recall defines the proportion of those instances that have been classified by a class divided by the total instances present in the class. Recall is the true positive rate (also referred to sensitivity). This parameter specifies the relative number of correctly positive classified instances among all the positive instances.
It can be represented as:

$$Recall = \frac{TP}{TP + FN}$$

***True Positive (TP):*** Number of instances predicted positive that are actually positive.

***False Negative (FN):*** Number of instances predicted negative that are actually positive.

**4.3 Comparing the Performance of Meta Algorithms on the Basis of the Size of Datasets**

- **Accuracy**

  Table 4.2 shows the accuracy rates of various algorithms. The results show that:
  - In case of Naïve Bayes, with the increase in the size of the data, accuracy of Naïve Bayes algorithm increases steadily and the accuracy rate of this algorithm is enhanced with the help of Bagging and Boosting algorithms;
  - In case of Decision Table, that with the increase in the size of the data, accuracy of Decision Table algorithm first increases and when the data becomes very large, it then decreases a bit; and the accuracy rate of this algorithm is enhanced with the help of Bagging and Boosting algorithms;
  - In case of J48, with the increase in the size of the data, accuracy of J48 algorithm increases and when the data becomes very large then it tends to decrease a bit; and the accuracy rate of this algorithm is enhanced with the help of Bagging and Boosting algorithms; but it can be seen that bagging resulted in decreasing the accuracy rate of this algorithm in case of dataset A and boosting resulted in lowering the accuracy rate of this algorithm in case of dataset D;
  - In case of Stacking, with the increase in the size of data, accuracy of this algorithm increases steadily.

  Figure 4.1 shows the graphical representation of these results.

Table 4.2: Accuracy using Base Classifiers and Meta Algorithms

| **Algorithms** | **Dataset A** | **Dataset B** | **Dataset C** | **Dataset D** |
|---|---|---|---|---|
| Naïve Bayes | 67.7083% | 78.1436% | 79.2871% | 83.4096% |
| Boosting (Naïve Bayes) | 72.2917% | 79.234% | 79.2871% | 83.4096% |
| Bagging (Naïve Bayes) | 67.7083% | 79.4326% | 79.7218% | 83.2929% |
| Decision Table | 72.7083% | 86.472% | 88.22% | 85.627% |
| Boosting (Decision Table) | 76.25% | 89.563% | 93.3927% | 85.7867% |
| Bagging (Decision Table) | 76.25% | 89.847% | 93.1319% | 85.756% |
| J48 | 75.8333% | 88.632% | 92.9798% | 86.1706% |
| Boosting (J48) | 77.9167% | 89.468% | 95.1532% | 83.5754% |

| | | | | |
|---|---|---|---|---|
| Bagging (J48) | 74.375% | 89.468% | 94.1752% | 86.1153% |
| Stacking | 58.5417% | 60.1447% | 60.5955% | 75.919% |

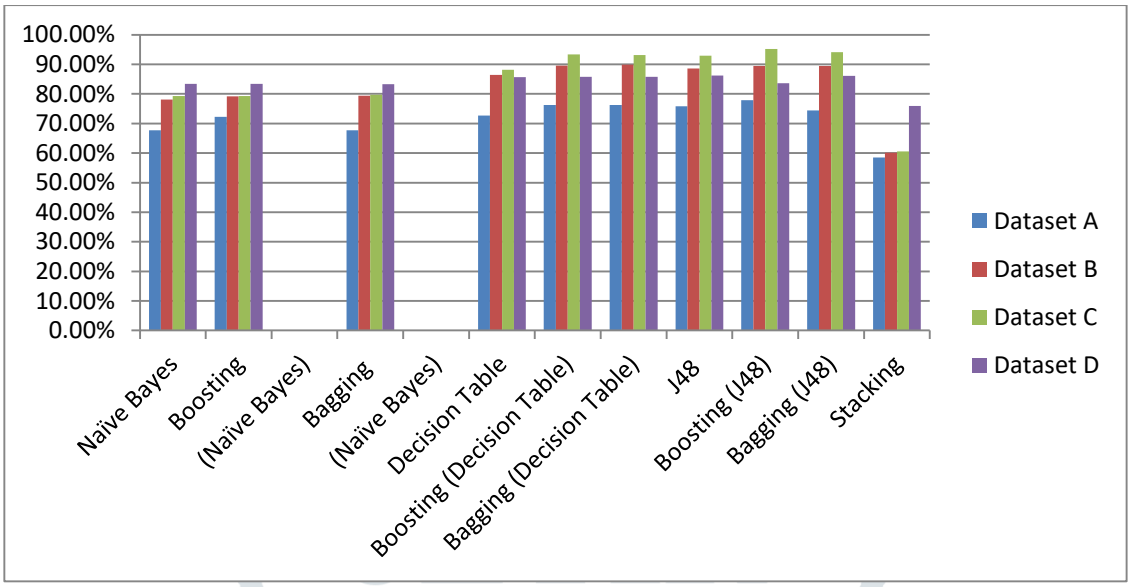


Fig. 4.1: Graphical Representation of Accuracy

- **Root Mean Squared Error**

Table 4.3 shows the root mean squared error of various algorithms. The results show that:

- In case of Naïve Bayes, with the increase in the size of the data from A to B, root mean square error of all algorithms decreases and same goes for the case when the size increases form C to D. Also, it is observed that Bagging and Boosting helps in lowering the error rate of this base classifier.
- In case of Decision Table, Bagging and Boosting helped in lowering the error rate of this base classifier; it is only in the case of Bagging on dataset A that the error rate has increased negligibly.
- In case of J48, with the increase in the size of the data, root mean square error of all algorithms decreases steadily but when the size increases to a large extent then it increases a bit. Also, it is observed that Bagging and Boosting helps in lowering the error rate of this base classifier; it is only in the case of dataset D that boosting has resulted in increasing the error rate.
- In case of Stacking, there is no specifically fixed pattern observed in the error rates of this algorithm with the increasing size of data.

Figure 4.2 shows the graphical representation of these results.

Table 4.3: Root Mean Squared Error using Base Classifiers and Meta Algorithms

| **Algorithms** | **Dataset A** | **Dataset B** | **Dataset C** | **Dataset D** |
|---|---|---|---|---|
| Naïve Bayes | 0.397 | 0.2945 | 0.4527 | 0.3723 |
| Boosting (Naïve Bayes) | 0.383 | 0.2848 | 0.4053 | 0.3502 |
| Bagging (Naïve Bayes) | 0.3935 | 0.283 | 0.4348 | 0.3728 |
| Decision Table | 0.3718 | 0.2571 | 0.2922 | 0.3183 |
| Boosting (Decision Table) | 0.3545 | 0.2483 | 0.2332 | 0.3159 |
| Bagging (Decision Table) | 0.3758 | 0.2164 | 0.2789 | 0.3172 |
| J48 | 0.3632 | 0.2772 | 0.2562 | 0.3201 |

| | | | | |
|---|---|---|---|---|
| Boosting (J48) | 0.3647 | 0.2665 | 0.2121 | 0.3824 |
| Bagging (J48) | 0.3366 | 0.224 | 0.2143 | 0.3144 |
| Stacking | 0.4325 | 0.3649 | 0.4886 | 0.4276 |

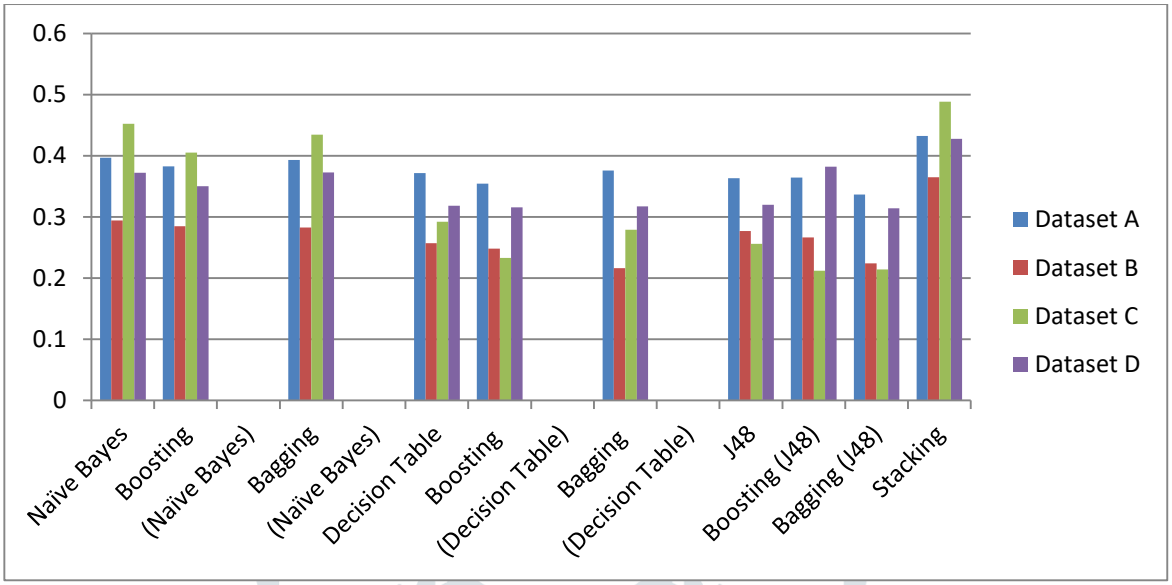


Fig. 4.2: Graphical Representation of Root Mean Squared Error

- **Kappa Statistics**

Table 4.4 shows the Kappa Statistics measure of various algorithms. The results show that:

- In case of Naïve Bayes, Bagging and Boosting resulted in lowering the kappa measure of this base classifier, while in ideal cases it should have been increased. It's only in the case of dataset D where Bagging resulted in increasing the measure by a very small amount.
- In case of Decision Table, Bagging and Boosting resulted in lowering the kappa measure of this base classifier also, while in ideal cases it should have been increased. It's only in the case of dataset A where Bagging resulted in increasing the measure by a very small amount.
- In case of J48, Bagging and Boosting resulted in lowering the kappa measure of this base classifier also, while in ideal cases it should have been increased. It's in the case of dataset A and D where Boosting resulted in increasing the measure.
- In case of Stacking, there is no specifically fixed pattern observed in the error rates of this algorithm with the increasing size of data.

Figure 4.3 shows the graphical representation of these results.

Table 4.4: Kappa Statistics measure using Base Classifiers and Meta Algorithms

| **Algorithms** | **Dataset A** | **Dataset B** | **Dataset C** | **Dataset D** |
|---|---|---|---|---|
| Naïve Bayes | 0.397 | 0.2945 | 0.4527 | 0.3723 |
| Boosting (Naïve Bayes) | 0.383 | 0.2848 | 0.4053 | 0.3502 |
| Bagging (Naïve Bayes) | 0.3935 | 0.283 | 0.4348 | 0.3728 |
| Decision Table | 0.3718 | 0.2571 | 0.2922 | 0.3183 |
| Boosting (Decision Table) | 0.3545 | 0.2483 | 0.2332 | 0.3159 |
| Bagging (Decision Table) | 0.3758 | 0.2164 | 0.2789 | 0.3172 |

| | | | | |
|---|---|---|---|---|
| J48 | 0.3632 | 0.2772 | 0.2562 | 0.3201 |
| Boosting (J48) | 0.3647 | 0.2665 | 0.2121 | 0.3824 |
| Bagging (J48) | 0.3366 | 0.224 | 0.2143 | 0.3144 |
| Stacking | 0.4325 | 0.3649 | 0.4886 | 0.4276 |

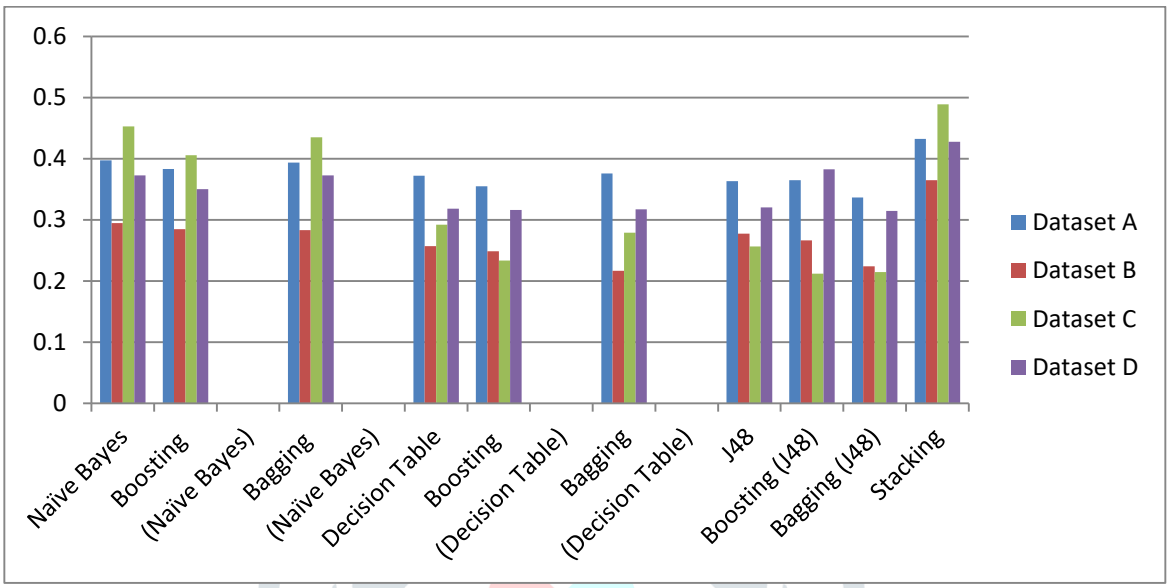


Fig. 4.3: Graphical Representation of Kappa Statistics measure

- **Precision**

Table 4.5 shows the precision values of various algorithms. The results show that:

- In case of Naïve Bayes, Bagging and Boosting resulted in improving the precision of this base classifier. It is also observed that precision increases when the size of data set increases steadily, and it decreases a bit when data becomes very large.
- In case of Decision Table, Bagging and Boosting resulted in improving the precision of this base classifier. It is also observed that precision increases when the size of data set increases steadily, and it decreases a bit when data becomes very large.
- In case of J48, Bagging and Boosting resulted in improving the precision of this base classifier. It is only in the case of dataset A that Bagging lowered the precision rate. It is also observed that precision increases when the size of data set increases steadily, and it decreases a bit when data becomes very large.
- In case of Stacking, precision value of this algorithm increases with the increasing size of data.

Figure 4.4 shows the graphical representation of these results.

Table 4.5: Precision using Base Classifiers and Meta Algorithms

| **Algorithms** | **Dataset A** | **Dataset B** | **Dataset C** | **Dataset D** |
|---|---|---|---|---|
| Naïve Bayes | 0.675 | 0.721 | 0.842 | 0.824 |
| Boosting (Naïve Bayes) | 0.724 | 0.724 | 0.842 | 0.824 |
| Bagging (Naïve Bayes) | 0.676 | 0.731 | 0.844 | 0.823 |
| Decision Table | 0.728 | 0.763 | 0.882 | 0.850 |
| Boosting (Decision Table) | 0.772 | 0.772 | 0.934 | 0.853 |
| Bagging (Decision Table) | 0.764 | 0.764 | 0.932 | 0.852 |

| | | | | |
|---|---|---|---|---|
| J48 | 0.760 | 0.775 | 0.930 | 0.856 |
| Boosting (J48) | 0.779 | 0.779 | 0.951 | 0.832 |
| Bagging (J48) | 0.743 | 0.781 | 0.942 | 0.856 |
| Stacking | 0.590 | 0.596 | 0.606 | 0.759 |

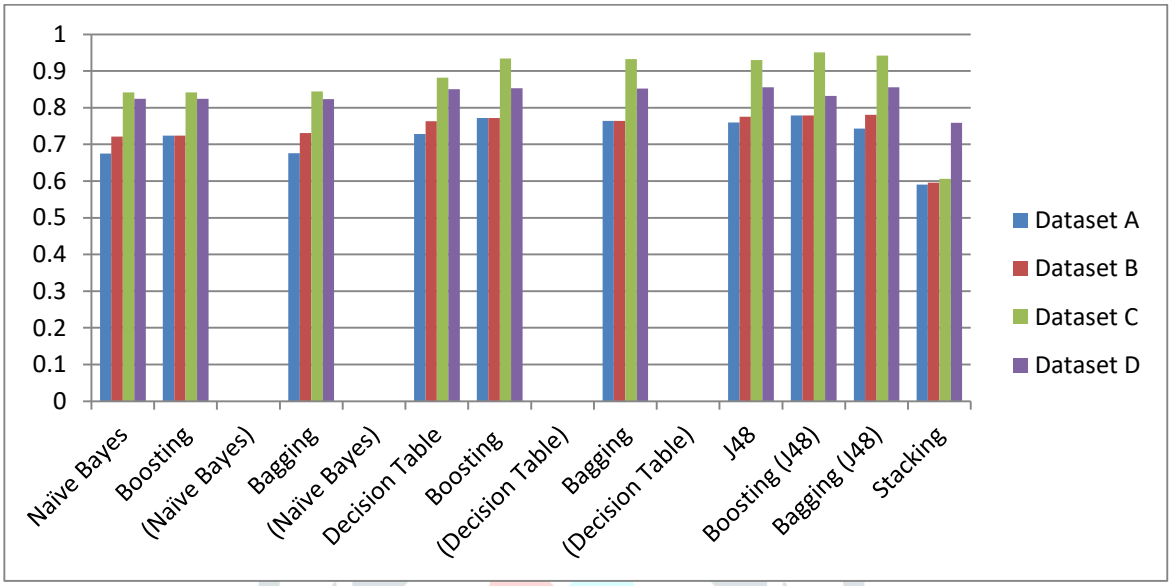


Fig. 4.4: Graphical Representation of Precision

- **Recall**

Table 4.6 shows the recall value of various algorithms. The results show that:

- In case of Naïve Bayes, Bagging and Boosting resulted in improving the recall of this base classifier. It is also observed that recall increases with the increasing size of data.
- In case of Decision Table, Bagging and Boosting resulted in improving the recall of this base classifier. It is also observed that recall increases when the size of data set increases steadily, and it decreases a bit when data becomes very large.
- In case of J48, Bagging and Boosting resulted in improving the precision of this base classifier. It is only in the case of dataset A and D that Bagging has lowered the recall rate. It is also observed that precision increases when the size of data set increases steadily, and it decreases a bit when data becomes very large.
- In case of Stacking, the recall value of this algorithm increases with the increasing size of data.

Figure 4.5 shows the graphical representation of these results.

Table 4.6: Recall using Base Classifiers and Meta Algorithms

| **Algorithms** | **Dataset A** | **Dataset B** | **Dataset C** | **Dataset D** |
|---|---|---|---|---|
| Naïve Bayes | 0.677 | 0.723 | 0.793 | 0.834 |
| Boosting (Naïve Bayes) | 0.723 | 0.725 | 0.793 | 0.834 |
| Bagging (Naïve Bayes) | 0.677 | 0.732 | 0.797 | 0.833 |
| Decision Table | 0.727 | 0.762 | 0.882 | 0.856 |
| Boosting (Decision Table) | 0.763 | 0.773 | 0.934 | 0.858 |
| Bagging (Decision Table) | 0.763 | 0.763 | 0.931 | 0.858 |

| | | | | |
|---|---|---|---|---|
| J48 | 0.758 | 0.773 | 0.930 | 0.862 |
| Boosting (J48) | 0.779 | 0.779 | 0.952 | 0.836 |
| Bagging (J48) | 0.744 | 0.782 | 0.942 | 0.831 |
| Stacking | 0.585 | 0.591 | 0.606 | 0.759 |

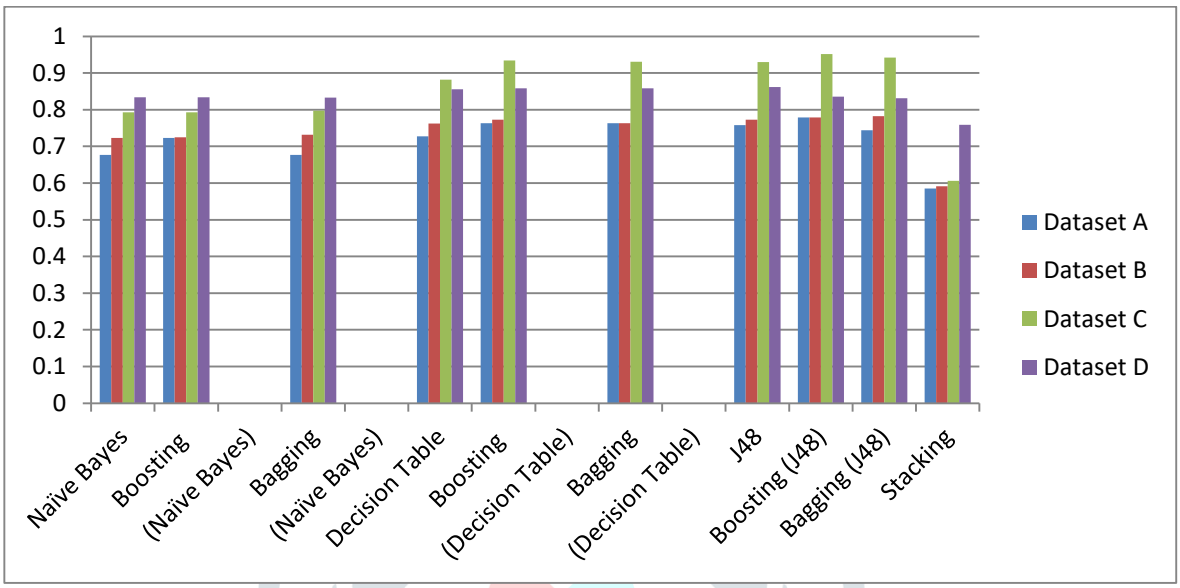


Fig. 4.5: Graphical Representation of Recall

## V. CONCLUSION AND FUTURE SCOPE

In this work, the performance of three base classifiers i.e. Naïve Bayes, Decision Table and J48, along with three Meta Learning techniques i.e. Bagging, Boosting and Stacking, have been analyzed with respect to the four datasets of varying sizes. All these algorithms have been compared on the basis of parameters like accuracy, root mean squared error, kappa statistics, precision and recall. Meta Learning techniques (also known as Ensemble Methods) use multiple learning algorithms for the same task in order to have better prediction than that of the individual base learning model. Reduced variance, reduced bias and improved prediction contributes to the advantages of Meta Learning methods. It has been observed form the experiments that J48 algorithm gave the highest accuracy rate among the base classifiers and the Meta methods helped improving the performance of J48 algorithm. From the results this has been observed that in majority of the cases Meta Learning algorithms (Ensemble methods) have the highest accuracy rates, least RMSE values and high rates of Kappa Statistics, Precision and Recall when compared to various classification algorithms. In some cases, Bagging and Boosting had the same values as that of the base algorithms but they never decreased the performance of base algorithms by a significant amount, though there were some negligible decreases at times. With respect to the size of the data, it has been observed that the accuracy rate of the algorithms increases with the steady increase in the data size and decreases a bit when the data grows at a larger scale; same behavior is experienced when the precision and recall rates are taken into consideration; in case of errors, error rates decreases with the increasing size of data; last but not the least, kappa statistics measure is expected to increase with the increasing size of data, but it did not behave so in case of the datasets taken here. However, few exceptional behaviors were also noticed. It can be concluded that, other than Kappa Statistics measure, all the other parameters behaved in the expected manner with respect to the increasing size of data.

For the future scope, same algorithms can be implemented on different application domain or tool instead of WEKA and their performance can be analyzed and improved with some other multiple learning techniques with different tools and algorithms can also be improvised with respect to some parameters. The impact of change in the number of folds of cross-validation can also be observed.

**REFERENCES**


[1] Jiawei Han, Micheline Kamber and Jian Pei, "Data Mining: Concepts and Techniques, Morgan Kaufmann", Third Edition, 2011.

[2] Sagar S. Nikam, "A Comparative Study of Classification Techniques in Data Mining Algorithms", Oriental Journal of Computer Science and Technology, Vol 8(1), pp 13-19,2015.

[3] G. Kesavaraj and S. Sukumaran, "A Study On Classification Techniques in Data Mining", ICCCNT Fourth International Conference on IEEE, pp 1-7,2013.

[4] G. Kesavaraj and S. Sukumaran, "A Study On Classification Techniques in Data Mining", ICCCNT Fourth International Conference on IEEE, pp 1-7,2013.

[5] V. Krishnaiah, Dr. G. Narsimha and Dr. N. Subhash Chandra, "Survey of Classification Techniques in Data Mining", International Journal of Computer Sciences and Engineering, Vol 2(9), pp 65-74, 2014.

[6] Data mining in bioinformatics using Weka", Eibe Frank Mark Hall Len Trigg Geoffrey Holmes Ian H. Witten,Bioinformatics, Volume 20, Issue 15, 12 October 2004, Pages 2479–2481

[7] Yanmin Sun et al., "Cost-Sensitive Boosting for the Classification of Multi-Class Imbalanced Data", IEEE, 2006

[8] Arvind Sharma, P.C.Gupta, " Predicting the Number of Blood Donors Through their Age and Blood Group using Data Mining Tool", International Journal of Communication and Computer Technologies (IJCCTS), Vol 01(6), 2012.

[9] Aayushi Verma, Shikha Mehta, "A Comparative Study of Ensemble Learning Methods for Classification in Bioinformatics", IEEE 2017

[10] Shiva Kazempour Dehkordi, Hedieh Sajedi, "A Prescription-based Automatic Medical Diagnosis System using a Stacking Method", IEEE 15th International Symposium on Intelligent Systems and Informatics, September 14-16, 2017

[11] Kuldeep Randhawa et al., "Credit Card Fraud Detection using AdaBoost and Majority Voting", IEEE, Vol 6, 2018

[12] Pelin YILDIRIM,Kökten Ulas BIRANT, Vladimir RADEVSKI, Alp KUT ve Derya BIRANT, "Comparative Analysis of Ensemble Learning Methods for Signal Classification", IEEE 2018

[13] Improving Heart Disease Prediction Using Feature Selection Approaches, Saba Bashir, Zain Sikander Khan, Farhan Hassan Khan, Aitzaz Anjum, Khurram Bashir, Proceedings of 2019 16th International Bhurban Conference on Applied Sciences & Technology (IBCAST), IEEE 2019

[14] WEKA, the University of Waikato, Available at: http://www.cs.waikato.ac.nz/ml/weka/

[15] UCI Repository, available at: https://archive.ics.uci.edu

[16] Kaggle Datasets, available at: https://www.kaggle.com

[17] Dataset A, available at: https://www.kaggle.com/aljarah/xAPI-Edu-Data/downloads/students-academicperformance-dataset.zip/6

[18] Dataset B, available at: https://www.kaggle.com/mansoordaku/ckdisease

[19] Dataset C, available at: https://archive.ics.uci.edu/ml/datasets/Spambase

[20] Dataset D, available at: https://archive.ics.uci.edu/ml/datasets/adult