Ensemble Theory – Boosting / Bagging

Sohail Dua
High Integrity Systems
Frankfurt University of Applied Sciences
Frankfurt am Main, Germany
sohail.dua@stud.fra-uas.de
Matriculation No:1322512

***Abstract*—Ensemble techniques is a powerful machine learning algorithm that is getting popular for research area in recent years especially in classification.It consists of several trained classifier such as decision trees or neural networks those are needed and combined to classify new data.It uses multiple classifiers to build a classification model which improves accuracy considerably. "Diversity" is one of the elements required for accurate prediction when using an ensemble.[1].It is used in many areas such as pattern recognition, Natural Language Processing and many more. In this paper we are gonna explain about the ensemble techniques that are used and how it improves ensemble learning.**

***Index Terms*—Machine learning, Ensemble theory, Bagging, Boosting, Prediction**

I. Introduction

Nowadays in the field of prediction the ensemble methods are becoming more interesting and useful.It is a very useful technique to combine multiple learning algorithms to improve accuracy of the prediction.Today we see many uses of ensemble methods in finance, health, bioinformatics etc.Ensemble methods such as bagging, boosting and stacking have been a significant advantage in offline learning i.e you have a static dataset settings.Ensemble methods and techniques have been successfully used to overcome machine learning issues, such as feature selection, confidence estimation, missing feature, incremental learning, error correction, learning concept drift from non stationary distributions, among others.[2]
The original goal for using ensemble systems is quite similar to how we use in our daily lives.Consider the fact that we make a right decision based on facts, opinions and combining with our intellect we come to a final proper decision.Same with the machine learning applications of ensemble systems .We have to select features, impute missing data, combining of different data types and addressing various other problems. In this paper we are gonna discuss about Ensemble methods or techniques in section 2 ,applications in section 3 and in depth analysis with code in section 4.

II. Ensemble Methods

Ensemble learning is a machine learning paradigm where multiple models (often called "weak learners") are trained to solve the same problem and combined to get better results. The main hypothesis is that when weak models are correctly combined we can obtain more accurate and/or robust models.[3]

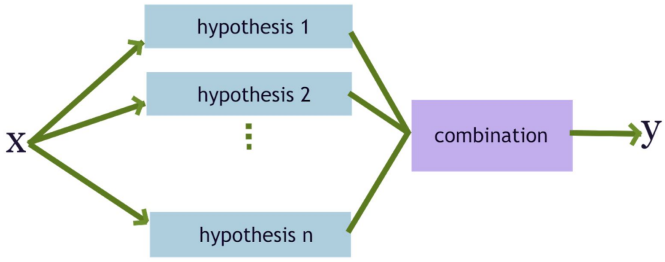

Fig. 1. A general Ensemble Architecture[4]

Ensemble methods can be divided in two groups:

- Sequential ensemble methods:- In these methods the base learners are generated in a sequential way for example in AdaBoost.The motivation of this type of model is to explore the dependency between the learner used.This will help in increasing performance by weighting those are previously mislabelled.

- Parallel ensemble methods :-In these methods the base learners are generated in parallel like in case of Random forest .The motivation of this type of algorithm is explore the dependency between the learner used to reduce the error by averaging

A. Simple Ensemble Techniques

This sections defines about simple ensemble methods:

1) Max Voting:-The max-voting method is generally used for general classification problems.It is one of the simplest ways of combining prediction from the many multiple machine learning algorithms.The prediction from every algorithm is considered as a vote.The prediction which occur the most time is considered as final prediction.For example when we rate a product on product site like amazon and we can rate it between one to five.Suppose it was rate three by most number of persons so the rating of product will be three.

2) Averaging:-It is very similar to max voting technique, multiple prediction are taken for each data point and then we average them. In this case, we take the average

of all the predictions we get from the models and then we give the final prediction. Averaging is a good tool for making predictions in regression and always we can calculate probability in classification problems.

3) Weighted Average:-A weighted average technique is an extension to the average technique as we have discussed.In this technique each machine learning algorithm contributes an equal amount to the final prediction.The final prediction of each algorithm used is weighted by performance of the model.For instance, if two of your colleagues are critics, while others have no prior experience in this field, then the answers by these two friends are given more importance as compared to the other people.[5]

B. Advanced Ensemble techniques

In the last part we discussed about the simple techniques and in this section we are going to discuss about the more advanced and common techniques used-

1) Bagging:-Bagging takes several weak models, and we take all the aggregate of the predictions and select the best prediction.The weak models specialize in distinct sections of the feature space, which enables bagging leverage predictions to come from every model to reach the utmost purpose.[6] The bagging technique is useful for both classification as well as regression problems.It is used with decision trees which increase its accuracy, lessen the variance and also reduces the chance of overfitting.It can be shown in Fig.2.

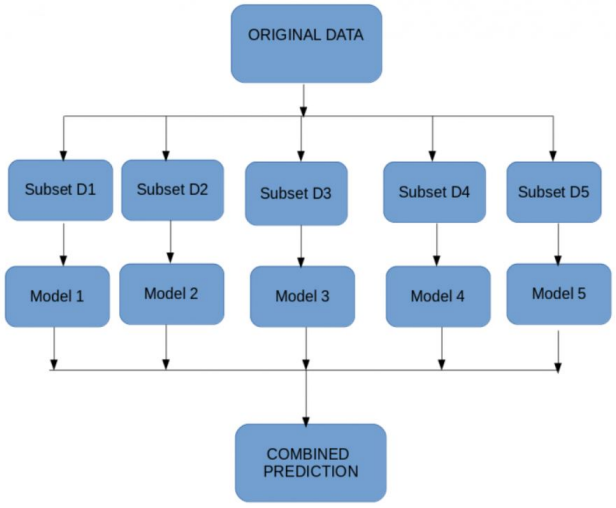

Fig. 2. Bagging technique[4]

The following step are followed in bagging technique :

- Multiple subsets are creating from original dataset, selecting observations with replacement
- A weak model is created on these subsets.
- All models run are independent and parallel to each other.
- The final predictions will be determined by combining all the predictions from all different models
  There are 2 types of bagging techniques:
  a) Bagging meta-estimator:-This algorithm is a ensembling algorithm that can be used for both regression and classification problems.Predictions are made using the normal bagging technique.These are step are for the bagging meta-estimator algorithm.
  - Creating random sets.
  - Subsets should include all the features as compared to the original dataset.
  - Then we use a user base estimator on all the subset need to be fitted.
  - Combining the prediction of every model to get the final result

  b) Random forest:-Random forest is another famous ensemble machine learning technique that is used now a days .It is a extension of bagging estimator algorithm.The only difference we use the decision trees as base estimators.Also differ from the bagging meta estimator as random forest does not select all features but select the features randomly and the split of each node is decided in decision tree.The following steps the algorithm uses
  - Original data are divided into random subsets.
  - We take only a random set of features at each node to decide the best split .
  - Each of the subsets is evaluated by the decision tree.
  - Then the final prediction is calculated by averaging all the predictions from all the decision trees

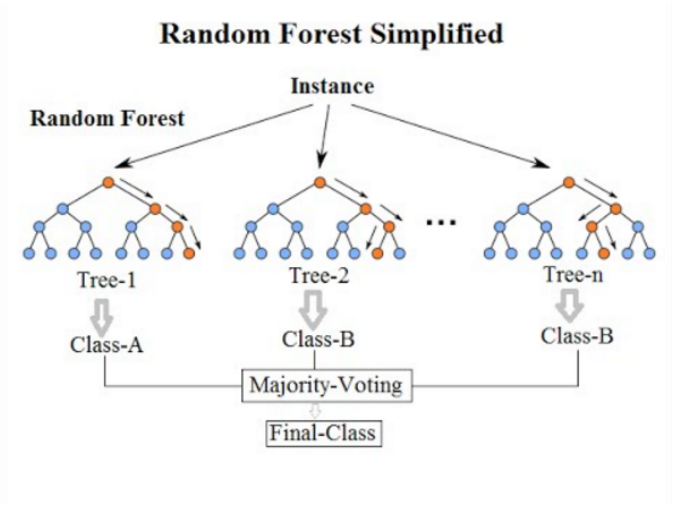

Fig. 3. Decision Tree[4]

Advantages of Bagging:

- Bagging has an advantage when we face overfitting and variance issue as it provides a base to deal with the variance and by using N learners of same size on same algorithm.
- Bagging using a sampling method that is called Bootstrap
- When we sample train data, overlaps has been a issue.So we have combinations of learners to overcome the issue

Disadvantages of Bagging:

- The problem with bagging is not effective in case of bias or underfitting of data.
- It also ignores the high value and lowest value which have a huge difference and it provides a mid range result

2) Boosting:- Boosting is an averaging algorithm that uses a set of machine learning models that need increased accuracy by converting weak learners to strong learners. It is one of the most powerful learning algorithms used.This method was actually made for classification but this gets extended for regression also.In the original boosting algorithm we use there weak learners to generate a strong learner.

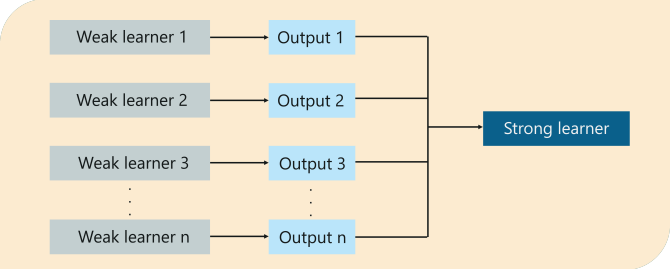

Fig. 4. Boosting Technique[8]

The following steps are required for boosting algorithm:-

- The base algorithm reads the data and assigns equal weight to each sample observation.[8]
- The predictions that are actually false are identified by the base learner.So in the next iteration,these false predictions are then assigned to the next base learner with a higher weightage on these incorrect predictions
- In the end keep repeating step 2 until this algorithm can correctly classify the output

The types of boosting algorithms are:-

a) AdaBoost:-AdaBoost is considered one of the most simple boosting algorithms. In this algorithm decision tree is used for modelling. It involves one level decision tree as weak learners that are sequentially added to the ensemble. Each model present try to make prediction of the previous model correct before it in the sequence. This is achieved by

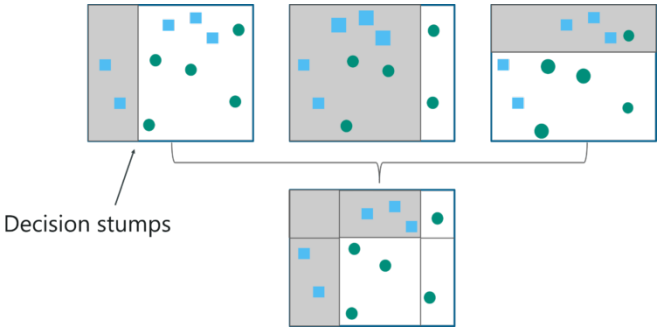

Fig. 5. How Boosting Works[8]

weighing the training dataset to put more focus on training examples on which prior models made prediction errors.[9] The following steps are persent int the algorithm

- All observations that are present in the original data set will be given equal weights A subset of data built up a model
- We make predictions on the whole data with this model we made in step 2.
- When we compare the actual values and the predicted values to get errors.
- So when we create a next model, we change the weights with higher weights to the data points where the prediction was incorrect
- We can determined the weights using the error value.Higher the error more more weight is assigned to the observation
- So repeat this process until the error function does not change or the maximum numbers of estimators is reached

b) Gradient Boosting (GBM):- It is one of the other ensemble machine learning algorithm or technique that is use for both regression as well classification.GBM uses the same technique formation of strong learner by combining weak learner.Regression trees are used as base learner and all the other trees are built on the errors calculated by the previous tree.

- Calculate the mean of the target variable.
- Calculate the residuals by mean value subtracting the actual value.
- We have to make a decision tree based on a goal on predicting the residuals.
- We should predict the target label using all the trees with the ensemble technique
- Again calculate the residuals and we consider it as new residuals
- Now repeat the steps 3 to 5 till the number of the iterations are fulfilled that are present in hyperparameter
- Once trained, use all of the trees in the ensemble to make a final prediction as to value of the

target variable. The final prediction will be equal to the mean we computed in Step 1 plus all the residuals predicted by the trees that make up the forest multiplied by the learning rate.[10]

c) XGBoost (extreme Gradient Boosting):-It is an extension of gradient boosting algorithm.XGBoost This algorithm is fast so mainly used in hackathons and competitions for quick result.It is 10 times faster than other gradient boosting algorithms.This technique is also called regularized boosting technique because as it reduces overfitting and increases overall performance.

d) Light GBM:-Light GBM grows tree vertically while other algorithm grows trees horizontally meaning that Light GBM grows tree leaf-wise while other algorithm grows level-wise. It will choose the leaf with max delta loss to grow. When growing the same leaf, Leaf-wise algorithm can reduce more loss than a level-wise algorithm.[5]

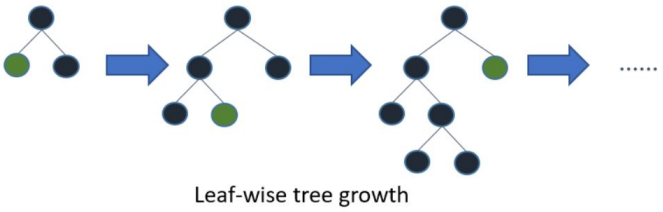

Fig. 6. Leaf Wise growth[5]

e) CatBoost:-This algorithm is used when handling a large number of such variables.As handling categorical variables is a very tough option.When your categorical variables have too many labels, and when performing one-hot-coding on them exponentially increases the dimensionality and sometimes it become difficult to work with dataset.It does not need data processing and CatBoost can deal with categorical variables.

Advantages of Boosting:

- This algorithm is best for the both weightage for the higher accuracy sample as wells as lower accuracy sample and then it combines the results.
- Net error is always calculated with learning steps.
- It takes away the issue of bagging as it works with dealing the bias or underfitting in the data.
- There are so many techniques available as discussed above.

Disadvantages of Boosting:

- The main issue of boosting is that it often ignores the overfitting or variance that is present in the dataset.
- Computations present can be expensive.

3) Stacking:-Stacking is a technique which makes predictions to make a new model from multiple models for example decision trees, knn.It is a way to ensemble classifications or regression model. It is also called stacked generalization.The importance in stacking is to explore a space of different models for the same problem.It means we can solve a problem with a different type of model which can only learn a part of the problem but not the whole problem.

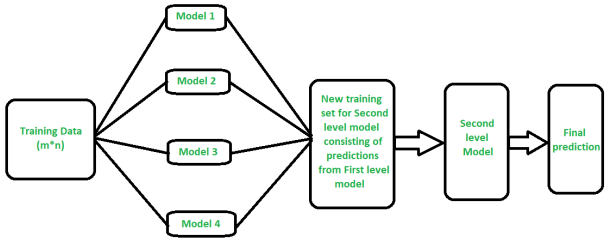

Fig. 7. Stacking technique[11]

- We use the splitting technique using training data and test data by K-fold cross validation.
- We will fit the base model for K-1 parts and predictions are made for Kth part.
- So we have to do each part for every training data.
- We use based model is then fitted on the whole train data set to calculate the performance of the dataset.
- Use the last 3 steps again for other base models
- Predictions from the training set are used as feature for second level in a model
- Then we second level model to make prediction on the test data

Advantages of Stacking:

- Stacking can increase model performance
- It also help in reducing variance and it helps in creating a more robust model by combining the prediction of multiple models.

Disadvantages of Stacking:

- It requires more significantly more time to train a simple model and also requires more memory.
- It is very expensive.
- Predictions made using this algorithm are slow.

4) Blending:-It is same as stacking but only difference is we use holdout method in which set from train set is used to make predictions.In other words the predictions are made only on the holdout set.The holdout set and predictions are used are combined to make a model which is going to run on the test set. The following steps are present in the algorithm:-

- The train set is divided into a training set and validation set.
- All the models are fitted on a training set.
- The predictions are made on test sets and validations set

- The validation set and the predictions we get are used to build a new build as features.
- The model is used make the final predictions and meta-features

Advantages of Blending:

- In blending the stacking ratio is very simple as we dont use any k-th stacker feature,
- An information disclosure is used to avoid the issues:stacker and generalizers use different data set

Disadvantages of Blending:

- Very little data Blender may be over-fitting Stacking use of multiple CV would be more robust

5) Naive bayes classifier:- Naïve Bayes classifier ensemble is a predictive model that we want to construct or discover from the dataset.[12] Then how we generate models from data is called learning or training.It can be accomplished by a learning algorithm.In supervise learning we predict the value of a target feature without knowing what is there and the learned model is called predictor.. For example, if we want to predict the color of the synthetic data points, we call "yellow" and "red" labels, and the predictor should be able to predict the label of an instance for which the label information is unknown, for example, (0.7, 0.7). If the label is categorical, such as color, the task is called classification and the learner is also called classifier.[12]
Advantages of Naive bayes classifier:

- This algorithm is very fast and saves a lot of time
- It is highly useful for solving multiple class prediction problems
- If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.[13]
- It is best use for categorical data rather than numerical data

Disadvantages of Naive bayes classifier:

- It assumes that all the features present in database are independent.This limits its application and usage in real world scenario.
- Estimations can be wrong so it can lead to wrong outputs.
- This algorithm faces the 'zero-frequency problem' where it assigns zero probability to a categorical variable whose category in the test data set wasn't available in the training dataset. It would be best if you used a smoothing technique to overcome this issue.[13]

III. Applications of Ensemble Methods

There are many applications that are related to ensemble method. In this section we are going to discuss some the applications:

1) Remote sensing:-Remote sensing is the process of detecting and monitoring the physical characteristics of an area by measuring its reflected and emitted radiation at a distance (typically from satellite or aircraft). Special cameras collect remotely sensed images, which help researchers "sense" things about the Earth.[14]

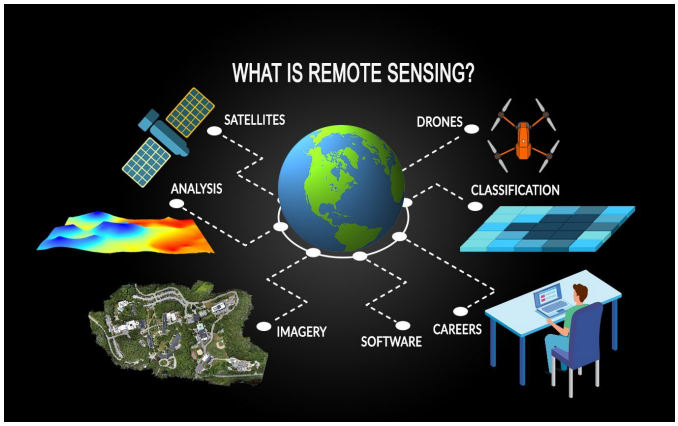

Fig. 8. Remote Sensing[15]

To do remote sensing we need :

- A large number of inputs as patterns are collected and refined so it can be used for larger spaces.
- A large number of features are needed as data is collected many hundred of bands large number of outputs.
- A large number of outputs are required like we have to distinguish between forest , water and other house,streets which are man made.
- Data can be missing or corrupted so we have to impute it as satellite can fail to collect data .
- Most of the data is unlabeled in this world so it needs to be processed and assigned to classes

Some applications are:Random forests and mountains, Majority voting for agricultural land, classification of wetlands and Information fusion for Urban areas

2) Person recognition: It is basically a problem to verify people identity by different characteristics of a person.It is used for security purpose mostly: We have various types of recognition as follows:

- Iris recognition:-It is an automated method of biometric identification that uses mathematical pattern-recognition techniques on video images of one or both of the irises of an individual's eyes, whose complex patterns are unique, stable, and can be seen from some distance.[16]
- Fingerprint recognition:-It is most common now a days we use finger as an biometric identifier of people as fingerprint is unique for everyone
- Face recognition:-It is used basically now a days in our phones it makes certain points on our face and mark it and scale it to form a image.

- Behavior recognition:It can be used for both speech as well as handwriting.It can also be used if we have specific knowledge of the person.

There are several issues with the Person recognition using the ensemble techniques:

- As there are multiple number of features present
- It is really difficult to correct the proper amount of data
- If suppose misclassfication happens then it can be a security issue

The applications are easy person identification, Face recognition, and User-specific speech recognition

3) Medicine:-In medicine today we have so many things to do we have analyze X-ray images ,human genes analysis and as well as we have to look the data of medical test for any anomalies.We also need data to analyze the health of a human being. The characteristics of medicine data:
   - Today we have limited training and test example which lead to nature of issues and some privacy concern.
   - we also have a imbalanced data set that causes issues can have few anomalies.
   - sometime we can to many attributes then there are more number of training and test data
   - we can also have a issue of misclassification

   The applications are related to MRI and ECG classification.

4) One vs. all recognition:-All provides a way to leverage binary classification. Given a classification problem with N possible solutions, a one-vs.-all solution consists of N separate binary classifiers—one binary classifier for each possible outcome.[17]
   The different types are:
   - Anomaly detection:This is a way of detecting unusual patterns. It also about the data that does not fit into the set of patterns that are identified
   - Target recognition:We have to also find what fits into a identified pattern
   - Intrusion detection:It can be solved by two methods that are discussed above.

   The applications are Modular intrusion detection, Hierarchical intrusion detection and Intrusion detection in mobile ad-hoc networks.

IV. In-Depth:Ensemble Algorithms using python

In this section we are going to discuss about the dataset we are using for performing our ensemble methods, then will use python cor ensemble methods and then will discuss the outputs.

A. Dataset

The dataset used for bagging and boosting is the titanic dataset. The titanic data frames describe the survival status of individual passengers on the Titanic. The titanic data frame does not contain information from the crew, but it does contain actual ages of half of the passengers.In this dataset we have variables that are as follows:

- survival : Defines the survival of people
- pclass : The person ticket class
- sex : male or female
- Age : Age of a person
- sibsp : of siblings / spouses aboard the Titanic
- parch : of parents / children aboard the Titanic
- ticket : Ticket Number
- fare : Passenger fare
- cabin : Cabin number
- embarked : Port of Embarkation

The dataset that used for blending is the breast cancer.It has 33 columns.It is the inbuilt database in sklearn library in python.

B. Ensemble Codes using Python

To perform ensemble techniques we need three input variables that are 'sex', 'age' and 'pclass'.Using these input variables we will try to predict the output that is the survival.The chances of survival of the person.

1) Bagging in Depth: in this part we are going to perform the bagging techniques and will discuss the outputs.We are going to discuss the random forest and Bagging meta estimator.The following code is taken from [18]

EnsembleBagging.py > y

```
1
2   import numpy as np
3   import seaborn as sns
4   import matplotlib.pyplot as plt
5   import pandas as pd
6
7   from sklearn import preprocessing
8   from sklearn.tree import DecisionTreeClassifier
9   from sklearn.ensemble import BaggingClassifier
10
11  from sklearn.model_selection import train_test_split
12  from sklearn.model_selection import cross_val_score, cross_val_predict
13  from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
14
15  lb = preprocessing.LabelBinarizer()
16  df = sns.load_dataset('titanic')
17  df.dropna(inplace=True)
18  df['pclass'].unique()
19  df['pclass'].value_counts()
20  df['sex'].unique()
21  df['sex'].value_counts()
22  df['age'].hist(bins=50);
23
24  #Data preprocessing
25
26  X = df[['pclass', 'sex', 'age']]
27  X['sex'] = lb.fit_transform(X['sex'])
28  y = df['survived']
29  y.value_counts()
30
```

Fig. 9. Check data content and preprocessing

As seen in figure 9 we have use the scikit-learn library to perform ensemble techniques as line 8 and line 9 tells us about the BaggingClassifier and DecisionTreeClassifier.Then we have perform Data preprocessing that involves transforming raw data into an understandable format as start from line 24.We need to impute and clean data as there are missing values in the function as seaborn library is used for that.The

other libraries that are used: numpy, matplotlib, and pandas

```
EnsembleBagging.py > ...
30
31  #fit model
32  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
33
34  def print_score(clf, X_train, y_train, X_test, y_test, train=True):
35      '''
36      print the accuracy score, classification report and confusion matrix of classifier
37      '''
38      if train:
39          '''
40          training performance
41          '''
42          print("Train Result:\n")
43          print("accuracy score: {0:.4f}\n".format(accuracy_score(y_train, clf.predict(X_train))))
44          print("Classification Report: \n {}\n".format(classification_report(y_train, clf.predict(X_train))))
45          print("Confusion Matrix: \n {}\n".format(confusion_matrix(y_train, clf.predict(X_train))))
46
47          res = cross_val_score(clf, X_train, y_train, cv=10, scoring='accuracy')
48          print("Average Accuracy: \t {0:.4f}".format(np.mean(res)))
49          print("Accuracy SD: \t\t {0:.4f}".format(np.std(res)))
50
51      elif train==False:
52          '''
53          test performance
54          '''
55          print("Test Result:\n")
56          print("accuracy score: {0:.4f}\n".format(accuracy_score(y_test, clf.predict(X_test))))
57          print("Classification Report: \n {}\n".format(classification_report(y_test, clf.predict(X_test))))
58          print("Confusion Matrix: \n {}\n".format(confusion_matrix(y_test, clf.predict(X_test))))
59
```

Fig. 10. Fit data and get performance parameters

Now we divide the data into train data and test data with test size is 30 percent of original data.There is a print_score function for all the performance parameters of training and also for test as shown in figure 10

```
EnsembleBagging.py > ...
59
60  print("------------------ Desicion Tree----------------------------")
61
62  clf = DecisionTreeClassifier(random_state=42)
63
64  clf.fit(X_train, y_train)
65
66  print_score(clf, X_train, y_train, X_test, y_test, train=True)
67
68  # Bagging tree estimator
69  print("------------------ bagging estimator----------------------------")
70  bgt =  BaggingClassifier(random_state = 42)
71
72  bgt.fit(X_train, y_train)
73  print_score(bgt, X_train, y_train, X_test, y_test, train=True)
74
```

Fig. 11. Performing Bagging methods

So this is where we perform our ensemble techniques as shown in figure 11. We can also put train=false

a) Output: This part we are gonna define the output corresponding to bagging techniques:

- Random Forest performance:-The figure 12 shows all the results as we can see it has of 94 percent accuracy which is quite good.
- Bagging meta estimator performance:- The figure 13 shows all the results as we can see it has of 93 percent which is quite good but less than random forest.

2) Boosting in Depth: :-in this part we are going to perform the bagging techniques and will discuss the outputs.We are going to discuss the Adaboost and Gradient Boosting.The following code is taken from [18]

As seen in figure 9 we have use the scikit-learn library to perform ensemble techniques as line 7 ,line 8 and line 9 tells

```
OUTPUT   TERMINAL   PROBLEMS   DEBUG CONSOLE

------------------ Desicion Tree----------------------------
Train Result:

accuracy score: 0.9449

Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.95      0.92        43
           1       0.98      0.94      0.96        84

    accuracy                           0.94       127
   macro avg       0.93      0.95      0.94       127
weighted avg       0.95      0.94      0.95       127


Confusion Matrix:
 [[41  2]
 [ 5 79]]

Average Accuracy:        0.7628
Accuracy SD:             0.0980
```

Fig. 12. Random Forest performance

```
OUTPUT   TERMINAL   PROBLEMS   DEBUG CONSOLE

------------------ bagging estimator----------------------------
Train Result:

accuracy score: 0.9370

Classification Report:
              precision    recall  f1-score   support

           0       0.93      0.88      0.90        43
           1       0.94      0.96      0.95        84

    accuracy                           0.94       127
   macro avg       0.93      0.92      0.93       127
weighted avg       0.94      0.94      0.94       127


Confusion Matrix:
 [[38  5]
 [ 3 81]]

Average Accuracy:        0.7635
Accuracy SD:             0.0619
```

Fig. 13. Bagging meta estimator performance

us about the AdaBoostClassifier, RandomForestClassifier and DecisionTreeClassifier as shown in figure 14.Then we have perform Data preprocessing that involves transforming raw data into an understandable format as start from line 23.We need to impute and clean data as there are missing values in the function as seaborn library is used for that.The other libraries that are used: numpy, matplotlib, and pandas. Now we divide the data into train data and test data with test size is 30 percent of original data.There is a print_score function for all the performance parameters of training and also for test as shown in figure 10 as same for the bagging.

These are boosting methods as shown in figure 15.

a) Output: This part we are gonna define the output corresponding to bagging techniques:

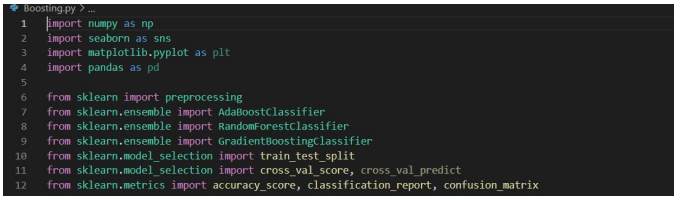

Fig. 14. Imports from sklearn library

Fig. 15. Performing boosting methods

- Adaboost performance:-The figure 16 shows all the results as we can see it has of 85 percent accuracy which is quite good.

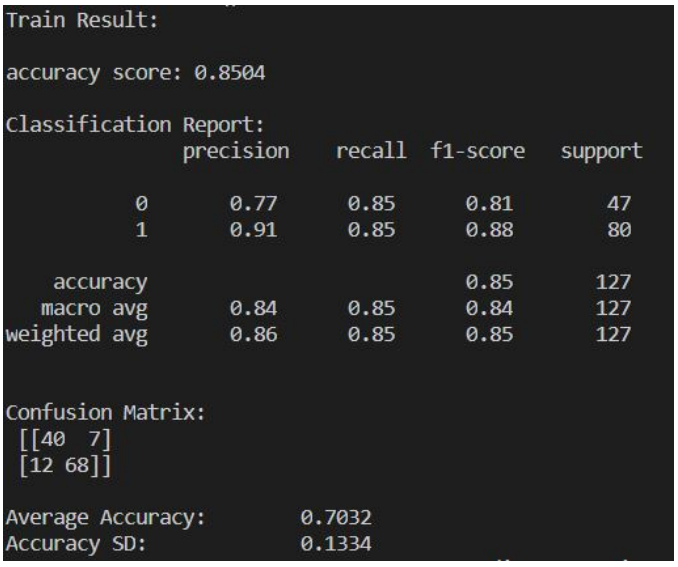

Fig. 16. Adaboost performance

- Gradient Boosting performance:- The figure 17 shows all the results as we can see it has of 92 percent which is quite good but less than random forest.

C. Blending in depth

The following code is taken from the repo[19]

- Part one of code of Blending in figure 18
- Part two of code of Blending in figure 19
- Part three of code of Blending in figure 20

a) Output: : The output is shown in figure 21.The train accuracy is 95 percent and test accuracy is 93 percent.

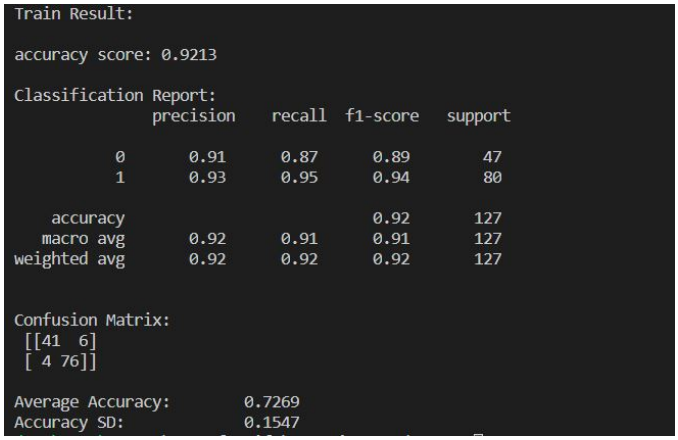

Fig. 17. Gradient Boosting performance

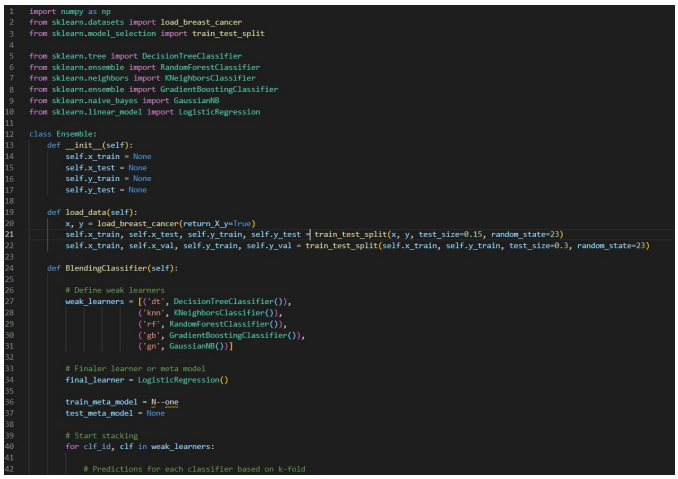

Fig. 18. Part one of code of Blending

CONCLUSION

As we have discussed about ensemble theory, it is a method that trains numerous learning machines and combines them to obtain a main model which is more accurate and take better decisions. Today in this world the ensemble methods are used everywhere.

As we have explained in previous sections, these ensemble techniques are quite popular and can be used among different domains? The only question left is which algorithm of all is better to use?With the above discussion it can be concluded only one algorithm is not better to achieve better accuracy as any algorithm can make poor choice anytime.In my opinion Bagging is considered to have most accuracy because of its easy implementation as well as its functionality on limited data size.All techniques can have their own pros and cons.It certainly depends on data and also the classifier used in ensemble techniques.

ACKNOWLEDGEMENT

The following references are use are given below:

```
    # Start stacking
    for clf_id, clf in weak_learners:

        # Predictions for each classifier based on k-fold
        val_predictions, test_predictions = self.train_level_0(clf)

        # Stack predictions which will form
        # the inputa data for the data model
        if isinstance(train_meta_model, np.ndarray):
            train_meta_model = np.vstack((train_meta_model, val_predictions))
        else:
            train_meta_model = val_predictions

        # Stack predictions from test set
        # which will form test data for meta model
        if isinstance(test_meta_model, np.ndarray):
            test_meta_model = np.vstack((test_meta_model, test_predictions))
        else:
            test_meta_model = test_predictions

    # Transpose train_meta_model
    train_meta_model = train_meta_model.T

    # Transpose test_meta_model
    test_meta_model = test_meta_model.T

    # Training level 1
    self.train_level_1(final_learner, train_meta_model, test_meta_model)
```

Fig. 19. Part two of code of Blending

```
    def train_level_0(self, clf):
        # Train with base x_train
        clf.fit(self.x_train, self.y_train)

        # Generate predictions for the holdout set (validation)
        # These predictions will build the input for the meta model
        val_predictions = clf.predict(self.x_val)

        # Generate predictions for original test set
        # These predictions will be used to test the meta model
        test_predictions = clf.predict(self.x_test)

        return val_predictions, test_predictions

    def train_level_1(self, final_learner, train_meta_model, test_meta_model):
        # Train is carried out with final learner or meta model
        final_learner.fit(train_meta_model, self.y_val)

        # Getting train and test accuracies from meta_model
        print(f"Train accuracy: {final_learner.score(train_meta_model,  self.y_val)}")
        print(f"Test accuracy: {final_learner.score(test_meta_model, self.y_test)}")


if __name__ == "__main__":
    ensemble = Ensemble()
    ensemble.load_data()
    ensemble.BlendingClassifier()
```

Fig. 20. Part three of code of Blending

```
Train accuracy: 0.9517241379310345
Test accuracy: 0.9302325581395349
```

Fig. 21. Output for blending

REFERENCES

[1] D.Gopika, and B.Azhagusundari, "An Analysis on Ensemble Methods In Classification Tasks," International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 7, July 2014

[2] Cha Zhang ,and Yunqian Ma, Ensemble Machine Learning, 3rd ed., vol. 2. Springer, 2012, pp.1–2.

[3] Joseph Rocca,Ensemble methods: bagging, boosting and stacking, 2019, URL:-https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205, (visited on 03/04/2021)

[4] Basics Of Ensemble Learning In Classification Techniques Explained, 2019, URL:-https://analyticsindiamag.com/basics-of-ensemble-learning-in-classification-techniques-explained/,(visited on 03/05/2021)

[5] A Comprehensive Guide to Ensemble Learning, 2016, URL:-https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-for-ensemble-models/, (visited on 03/06/2021)

[6] Bagging (Bootstrap Aggregation), URL:-https://corporatefinanceinstitute.com/resources/knowledge/other/bagging-bootstrap-aggregation/, (visited on 03/06/2021)

[7] Ensemble Learning Explained! Part 2, 2018, URL:-https://vigneshmadanan.medium.com/ensemble-learning-explained-part-2-498ab87788b5, (visited on 03/07/2021)

[8] A Comprehensive Guide To Boosting Machine Learning Algorithms, 2019, URL:-https://www.edureka.co/blog/boosting-machine-learning, (visited on 03/07/2021)

[9] How to Develop an AdaBoost Ensemble in Python, 2020, URL:-https://machinelearningmastery.com/adaboost-ensemble-in-python/, (visited on 03/10/2021)

[10] Introduction to the Gradient Boosting Algorithm, 2020, URL:-https://medium.com/analytics-vidhya/introduction-to-the-gradient-boosting-algorithm-c25c653f826b, (visited on 03/11/2021)

[11] Stacking in Machine Learning, 2019, URL:-https://www.geeksforgeeks.org/stacking-in-machine-learning/, (visited on 03/13/2021)

[12] Stacking in Machine Learning, 2019, URL:-https://www.geeksforgeeks.org/stacking-in-machine-learning/, (visited on 03/13/2021)

[13] Qingchao Liu, Jian Lu, Shuyan Chen,and Kangjia Zhao,Multiple Naïve Bayes Classifiers Ensemble for Traffic Incident Detection, Vol. 2014

[14] What is remote sensing and what is it used for? , URL:-https://www.usgs.gov/faqs/what-remote-sensing-and-what-it-used?qt-news$_science_products$, ($visitedon 03/15/2021$)

[15] Remote Sensing, Mapping, etc, URL:- https://frg.berkeley.edu/remote-sensing-mapping-etc/, (visited on 03/17/2021)

[16] Iris recognition, URL:-https://en.wikipedia.org/wiki/Iris$_recognition$/, ($visited\ on\ 03/21/2021$)

[17] Multi-Class Neural Networks: One vs. All , URL:-https://developers.google.com/machine-learning/crash-course/multi-class-neural-networks/one-vs-all, (visited on 03/22/2021)

[18] ML-Ensembles-Methods,2018 , URL:-https://github.com/Davisy/ML-Ensembles-Methods, (visited on 03/22/2021)

[19] Stacking-Blending-Voting-Ensembles, 2020 , URL:-https://github.com/FernandoLpz/Stacking-Blending-Voting-Ensembles, (visited on 03/23/2021)