**Diabetes prediction using Adaboost Random Forest Algorithm and logistic regression Analysis**

Stuti Rastogi\*, Simmi Deol, Anju Gera, Bhoopendra Dwivedi

GL Bajaj institute of technology and management, Greater Noida, Uttar Pradesh, India,

\*Author to whom correspondence should be addressed:

Email: [stutirastogi3101@gmail.com](mailto:stutirastogi3101@gmail.com)

(Received month date, year; Revised month date, year; accepted month date, year).

**Abstract:** Diabetes is sometimes called the silent killer as it takes long for the symptoms to appear. In this paper we applied technology to predict diabetes at an earlier stage based on various features. For this, we used and applied three machine learning algorithms on the dataset obtained from UCI repository having 17 attributes and 520 records of different patients. Algorithms including Random forest, Adaboost and Logistic regression were applied using the train-test split method. The accuracies obtained were further compared and best results were found using Random forest giving training accuracy of 100% and testing accuracy of 99.39%.

**Keywords:** Random forest, Adaboost, Logistic Regression, Machine learning algorithm, Train-test split, comparative study, accuracy

1. **Introduction:**

Diabetes is not a disease, it is a widespread disorder. Nowadays it is also found in teenagers which was earlier observed or found in adults. various aspects such as family health history, obesity, high bp, and quality of food are responsible for developing diabetes.

Two types of diabetes are observed in people. Type 1: In this insulin-producing cell in the pancreas are affected by the immune system thus production of insulin is less this causes diabetes and mainly occur due to family history. Type 2: This diabetes occurs due to family history, obesity, and an inactive lifestyle if diabetes is not detected in the early stage then multiple risk factors are involved. Diabetes can result in multiple organ failures like kidneys, liver, heart, nerves and so on. Diabetes patients require daily attention and any negligence in the treatment can lead to the patient's life at risk.

Machine learning is a field of study which if used properly can help us predict diabetes in the early stage using various factors. Detecting diabetes in the early stages will help the diabetic patient to take preventive measures and improve their health algorithms such as random forest, naive Bayes, AdaBoost, logistic regression, etc. can help us to recognize patterns in the data of patients and reach a conclusion. Many researchers have done many experiments to identify diabetes. For this, a dataset of patients is required a machine learning algorithm to work on them.

Studies have demonstrated the effectiveness of machine learning algorithms for the diagnosis of diseases. Machine learning algorithms can combine data from many sources and incorporate the metadata from the research since they can manage a big volume of patient data.

1. **Literature Review:**

Naveen Kishore et al. [1] “used Pima Indian Diabetes Dataset for their research. They used some of the Machine Learning algorithms which are SVM, Random Forest, Decision Tree, KNN, and logistic regression. The proposed model obtained the highest accuracy by random forest (74.4%) followed by SVM (73.43%)”[1].

Jobeda Jamal Khanam et al. [2] developed a neural model consisting of two hidden layers, and various epochs and observed the accuracy as 88.6%. Accuracy was varied by changing the epochs and hidden layers. 88.6% was observed using 2 hidden layers and 400 epochs, the highest among the SVM, logistic regression, naive bayes, and random forest. He also found an accuracy of 76.825 for SVM and 79.9% for KNN.

Hakim El Massari et al. [3] discussed diabetes prediction using two different methods. A comparative study was done between the popular machine learning algorithms and ontology-based Machine Learning classification. It was done on the PIMA dataset. The ontology classifier provided the highest accuracy of 77.5%, followed by SVM.

Deepti Sisodia et al. [4] presented Diabetes Prediction using Weka Tool. “Some machine learning algorithms such as Naïve Bayes, SVM, and Decision Tree were run on the Tool. The model achieved the highest accuracy of 76.30 % using the Naive Bayes ML Algorithm”[4].

Shafi et al. [5] used three machine learning algorithms and implemented them on PIMA India Diabetes dataset. A graph visualizing the accuracy of the algorithms. Naive Bayes provided better results than SVM and Decision Tree.

Leila Ismail et al. [6] worked on the analysis of Mellitus type 2 diabetes and used the UCI diabetes dataset has 12 features. Models were evaluated in terms of accuracy, F-measure, and execution time. Algorithms like naïve Bayes, KNN, etc were used. The highest accuracy of 77.5% was achieved by Naive Bayes. It was obtained without feature selection and execution time with or without feature selection was 0.016 minutes.

Olta Llahaa and Amarildo Ristab et al. [7] implemented some machine-learning algorithms and some data-mining techniques on the dataset collected from a Public Health Institute. The work was done using the WEKA tool. The decision Tree provided the highest accuracy of 79% in the prediction.

Refat et al. [8] used a dataset with 17 attributes including class and applied various algorithms such as XGBoost, Decision tree, random forest, KNN, CNN e, etc, and found the highest testing accuracy of 99% with XBoost. XBoost results were superior as compared to other machine learning deep learning approaches.

 Ali A. Abaker et al. [9] predicted using a dataset collected from Alsukari hospital with 644 records and 29 features. Feature selection model was used to reduce the attributes of the dataset to 6 which made the model simple and understandable. Algorithms like the random forest, KNN, and logistic regression were used which resulted in the highest accuracy of 81% using logistic regression.

Mitushi Soni et al. [10] proposed “an approach using machine learning classification algorithms and an ensemble learning method. A classifier model was built for the machine learning algorithms: SVM, Random forest, KNN, and Decision Tree. An accuracy of 77% was obtained using Random Forest” [10].

 M.R. Murthy et al. [11] presented a system for diabetes prediction using three Machine learning algorithms (Random Forest (RF), Naive Bayes, and Decision Tree). The Dataset he worked on, was collected from Kaggle consisting of 253679 entries of data and 22 attributes. Random Forest provided the highest accuracy.

Dr. L. Arockiam et al. [12] used big data analytics techniques for early-stage diabetes prediction. They aimed at building a data warehouse, used big data analytics to generate patterns, and predicted diabetes using the generated patterns.

Mujumdar et al. [13], for getting better accuracy in the proposed model the authors took some external factors and some internal factors like BMI, age, insulin, etc. In the study working on different algorithms, they got 96% accuracy using the Logistic Regression algorithm and by application of pipeline, AdaBoost classifier as the est model which resulted in the highest accuracy of 98.8%.

Quan Zou et al. [14] used a dataset containing 14 attributes. They found that the accuracy of the model was not good using PCA, so he used every feature of the dataset and usedmRMR to have better results. According to them, fasting glucose gave better results, especially on the Luzhou dataset. The best result obtained on the Luzhou dataset was 0.8084. The best result for Pima Indians was 0.7721. The result they got on prediction with the Random Forest was 0.8084 when all the attributes were used.

Jingyu Xue et al. [15] used the “Support vector machine (SVM), Naïve Bayes Classifier, and LightGBM Machine learning algorithms”[15] to train the dataset. They took a minimum of 17 attributes to train the dataset. In the research, the ratio between the training set and the prediction set was 80:20. Naïve Bayes Classifier Algorithm gave 93 .27% accuracy whereas the Support Vector Machine gave 96.54%. They found that SVM gave the highest accuracy result.

Umair Muneer et al. [16] Butt took PIMA India Diabetes dataset. A total of 128 data points was used to test and validate the model. In his study, he used mostly the algorithms to train the dataset: Logistic Regression, Random Forest, and Proposed Fine-Tuned MLP. The accuracy of the algorithms was Logistic Regression gave 73.05%, Random Forest provided 77.4%, and fine-tuned MLP – with 86.083% accuracy.

1. **Methodology:**
   1. **Libraries Used:**

We have used four “python libraries: NumPy, Pandas, matplotlib, and Seaborn”[18]. The NumPy library is used for working with multidimensional arrays. It also consists of a set of operations performed on the arrays. Pandas is the most used library which is used for data manipulation along with its analysis. Matplotlib library helps in the visualization of data whereas Seaborn makes the visualized data more interactive.

To import the libraries in our Jupyter notebook, add a new python file to it. Use the following “commands:

import numpy as np import pandas as pd import matplotlib as plt import seaborn as sns” [18].

np, pd, plt, sns are the prefixes used for the libraries.

Run the cell. Now, all the libraries are imported into our python file.

* 1. **Importing datasets**

After importing the libraries, using pandas create a data frame to read the CSV dataset file. The following syntax is used to import datasets.

Variable\_name = pd. read\_csv (‘file path’)

The variable name is the name given to a data frame. Read\_csv is a method in pandas that is used to create a dataframe using the CSV dataset. The file path is the location of the dataset where it is stored. Now, run the cell and perform the required operations on the dataset.

* 1. **Dataset Description:**

We take this data set from Kaggle uploaded by Ishan Dutta. “This has been collected using direct sampling from the patients of Sylhet Diabetes Hospital in Sylhet, Bangladesh and approved by a doctor. The dataset consists of 521 rows and 17 columns with different attributes like Age, Gender, Polyuria, Polydipsia, sudden weight loss, weakness, Polyphagia, Genital thrush, visual blurring, Itching, Irritability, delayed healing, partial paresis, muscle stiffness, Alopecia Obesity and class” [17].

* 1. **Data Pre-processing:**

As the data is collected from several sources and the dataset is made using that data, there is a possibility of the dataset being inconsistent, containing null values, duplicate or missing values. So, it becomes necessary to pre-process the data so that it can easily be used for further processing.

1. Data cleaning is the first most step. Firstly, missing values are found. If there exist some values missing, then they are filled with some relevant values. In our dataset, there is no missing data.
2. The second step is Data Integration. In this, data collected from various different sources are merged together to form a single dataset. As we have collected data from a single source, there is no need for data integration.
3. Data Transformation. In the dataset, there is a need to transform the data type from yes/no to 1/0. So, we use the map function in the panda’s library to the required format.
   1. **Algorithms used:**
   2. **Random forest**

A supervised machine learning approach called the Random Forest Algorithm is employed to solve classification and regression issues. The forest contains a variety of trees, and it will be stronger if there are more of them. In a similar vein, the algorithm's accuracy and capacity for solving problems will rise as the number of trees increases. To improve the prediction accuracy of the provided dataset, Random Forest removes the average from many decision trees on different subsets of the given datasets. It is based on the concept of \*ensemble learning.

* + 1. **Applying decision trees in random forest**

The main distinction between the Decision Tree and Random forest is that the first one randomly place root node and separates nodes. The bagging method is used by the Random forest to produce the necessary prediction. Bagging involves using different training data instead of just one sample.

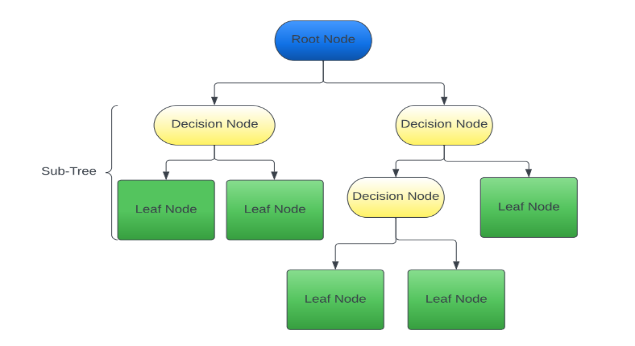


Fig 1: Random Forest

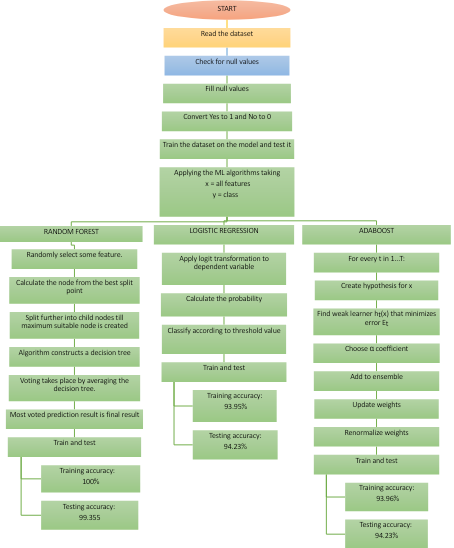
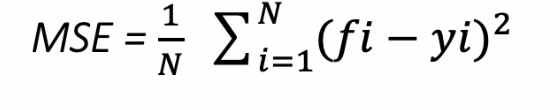


Fig 2: Proposed Workflow

NDT is the process of looking for discontinuities or differences in characteristics in materials, components, or assemblies while maintaining the structural integrity. The property of the concrete to be observed, such as strength, corrosion, crack monitoring, etc., determines which NDT method should be used.

* + 1. **The Mathematics Behind Random Forest Regression Problems**

In Random Forest Algorithm to solve regression problems, we are use the **MSE (Mean Squared Error)** methodto know how the data Branches from each node.



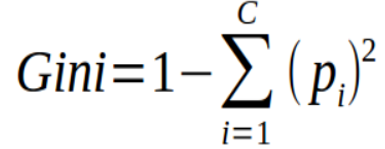
* Where n is the number of dataPoints
* Fi is the value returned by the model
* Yi is the actual value for data point i.

This algorithm calculates the distance between each node and the expected actual value, which helps us to find which branch is best for the given forest.

In this case, Yi stands for the value of the data point for testing on a certain node, and fi is the result given by the decision tree.

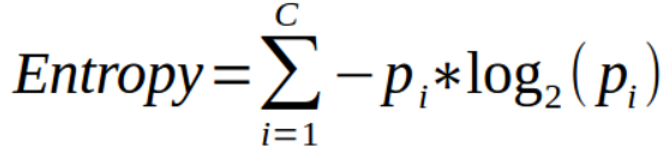
* + 1. **Classification problems**

When we work on random forests, we know they are based on classification of data. To find how nodes are placed on each branch of decision tree, we use the formula of Gini index, which is given by-



* + 1. **We can also use entropy to determine how the node branch in a decision tree.**

Entropy makes use of the probability of the result to decide how the node should be branched, unlike the Gini Index.



* 1. **Logistic regression**

The supervised machine learning algorithm called Logistic regression is an upgrade version of linear regression is used and explained here. It is used for solving classification problems. It helps in calculating the possibility of an occurrence of an event. It models the data by the use of sigmoid function which is a logistic function. The sigmoid function is S-shaped and it returns a value between 0 and 1 for a given number.

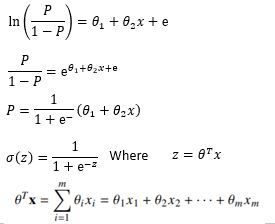
The Logistic function is:

where

e -> base of natural logarithms

x -> any real number

Logistic Regression uses logarithms of odd or logit function for calculation of probability. We can find odds by finding the ratio between probability of an event to occur to not occurring probability. Probability is calculated as follows:



where P: probability of an event

* + 1. **Working:**

1. Apply logit transformation to the dependent variable
2. Estimate the probability using the Logistic regression function.
3. Classify result according to the function (for binary)

y = {0, p<0.5; 1, p>=0.5

* 1. **Adaboost**

Adaboost, also called adaptive boosting, is a boosting algorithm which works by using an ensemble of weak learners and then combining them through the use of weighted sum.

Adaboost uses the previously created weak learners to adapt in order to adjust misclassified instances for the next created weak learner.

* A weak model is a model that is too simple to perform well on its own.

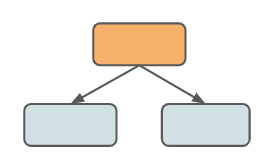


Fig 3: Simplest weak learner

* Weakest decision tree or a stump with a node and two leaves

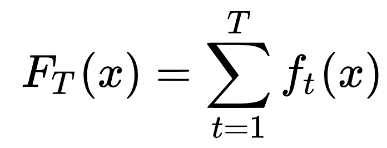
Adaboost works by aggregating multiple weak learners not like single decision tree which fits data all at once, allowing the ensemble model to learn the features slowly.

* + 1. **Main formulas of Adaboost:**

Boosted model = Ft(x)

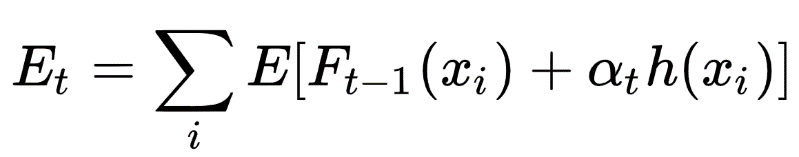
T = number of weak learners

ft(x) = single weak learner



  ft(x) = αt h(x)

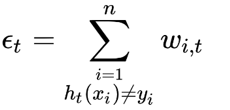
α = critical component of particular weak learner



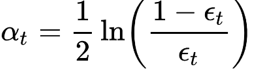
Where Et = sum of training error

* + 1. **Working of Adaboost:**

1. For t in 1...T:
2. Choose hypothesis ht(x)
   1. We find weak learner ht(x) such that it minimizes the error term, then weighted sum error for misclassified points is given by,



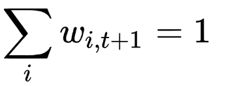
* 1. Choose critical parameter



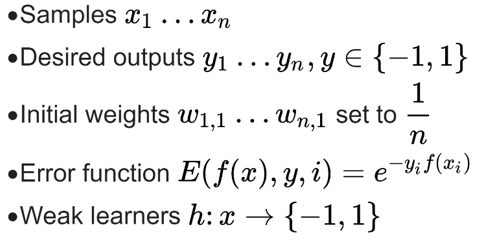
1. Add to ensemble



1. Update weights
2. Renormalize



Where:



1. **Implementation:**

We apply the above machine learning algorithms on the processed dataset and the method we used here to check accuracy is Train-Test split and further the results are generated.

**6.2. Train-Test split method:**

In this we divide the dataset into two parts: namely for training and testing. Larger percentage of dataset is used for training and the smaller one is used for testing the accuracy of the applied algorithm. The training dataset is provided to the model so that it effectively maps to the output. The testing dataset is then used to check the output given by the model. If close to or same as actual output, then the model is assumed to be efficient.

1. **Results:**

After applying the train-test split method on the above algorithms, different training and testing accuracies were found.

**7.1. Results after random forest algorithm**

Accuracies of 100% and 99.36% were obtained as training and testing accuracies. Figure 4 shows graph for classification report for training and figure 5 shows graph for classification report for testing.

Table 1: classification report for training random forest algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **precision** | 1 | 1 | 1 | 1 | 1 |
| **recall** | 1 | 1 | 1 | 1 | 1 |
| **f1-score** | 1 | 1 | 1 | 1 | 1 |

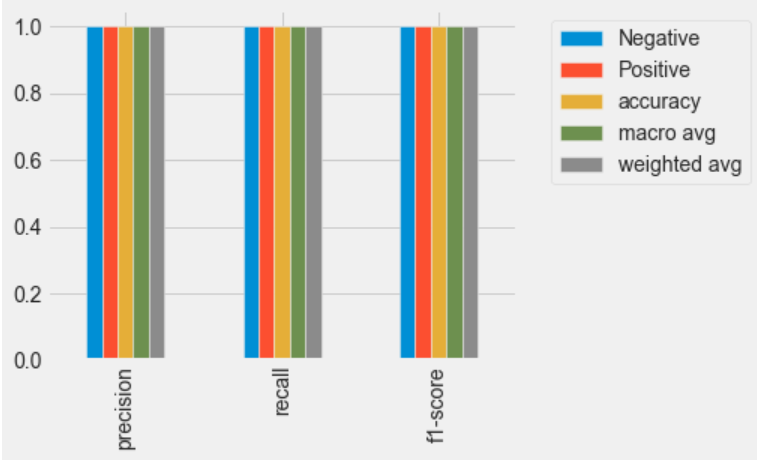


Fig 4: classification report for training random forest algorithm of the dataset.

Table 2: classification report for testing random forest algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **precision** | 0.981818 | 1 | 0.99359 | 0.990909 | 0.993706 |
| **recall** | 1 | 0.990196 | 0.99359 | 0.995098 | 0.99359 |
| **f1-score** | 0.990826 | 0.995074 | 0.99359 | 0.99295 | 0.993603 |

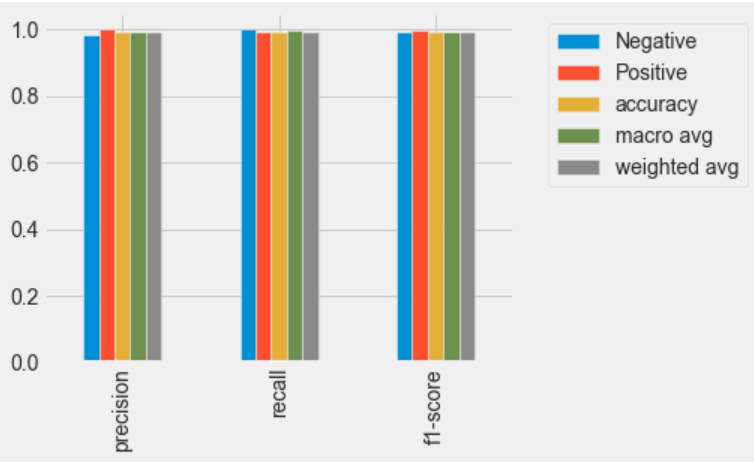


Fig 5: classification report for testing random forest algorithm on the dataset.

**7.2. Results after logistic regression algorithm**

Accuracies of 93.13% and 93.59% were obtained as training and testing accuracies. Figure 6 shows a graph for classification report for training and figure 7 shows graph for classification report for testing.

Table 3: classification report for training logistic regression algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **precision** | 0.900662 | 0.953052 | 0.931319 | 0.926857 | 0.932038 |
| **recall** | 0.931507 | 0.931193 | 0.931319 | 0.93135 | 0.931319 |
| **f1-score** | 0.915825 | 0.941995 | 0.931319 | 0.92891 | 0.931498 |

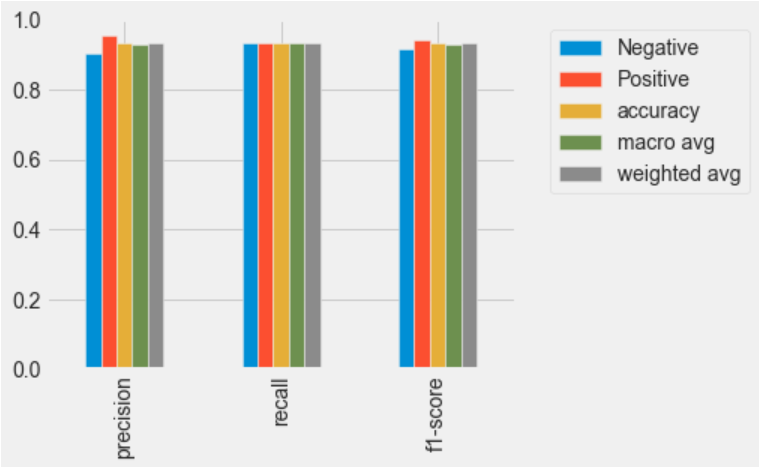


Fig 6: classification report for training logistic regression algorithm on the dataset.

Table 4: classification report for testing logistic regression algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **Precision** | 0.907407 | 0.95098 | 0.935897 | 0.929194 | 0.935897 |
| **Recall** | 0.907407 | 0.95098 | 0.935897 | 0.929194 | 0.935897 |
| **f1-score** | 0.907407 | 0.95098 | 0.935897 | 0.929194 | 0.935897 |

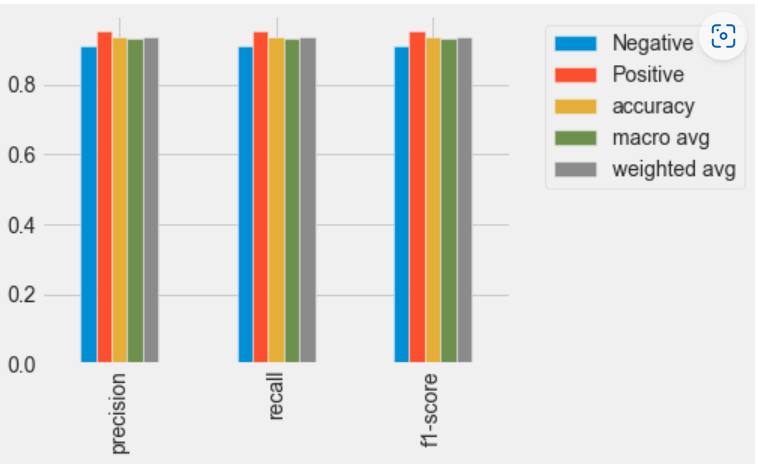


Fig 7: classification report for testing logistic regression algorithm on the dataset

**7.3. Results after Adaboost algorithm:**

Accuracies of 93.96% and 94.23% were obtained as training and testing accuracies. Figure 8 shows a graph for classification report for training and figure 9 shows graph for classification report for testing.

Table 5: classification report for training adaboost algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **precision** | 0.907895 | 0.962264 | 0.93956 | 0.935079 | 0.940457 |
| **recall** | 0.945205 | 0.93578 | 0.93956 | 0.940493 | 0.93956 |
| **f1-score** | 0.926174 | 0.948837 | 0.93956 | 0.937506 | 0.939747 |

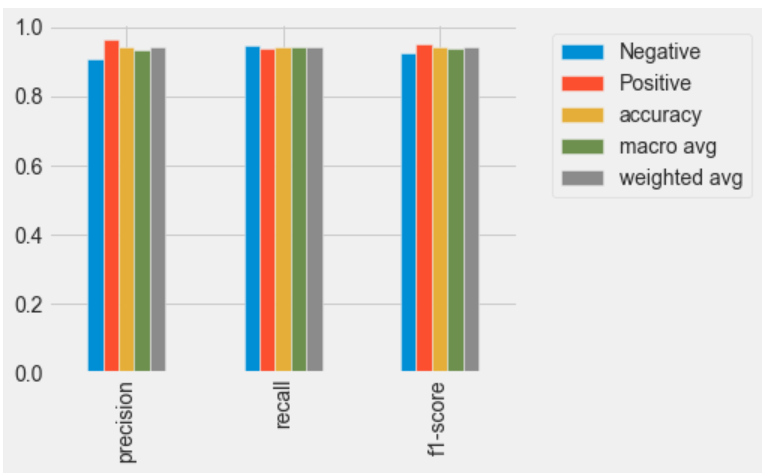


Fig 8: classification report for training adaboost algorithm on the dataset.

Table 6: classification report for testing adaboost algorithm

|  | **Negative** | **Positive** | **accuracy** | **macro avg** | **weighted avg** |
| --- | --- | --- | --- | --- | --- |
| **precision** | 0.909091 | 0.960396 | 0.942308 | 0.934743 | 0.942637 |
| **recall** | 0.925926 | 0.95098 | 0.942308 | 0.938453 | 0.942308 |
| **f1-score** | 0.917431 | 0.955665 | 0.942308 | 0.936548 | 0.94243 |

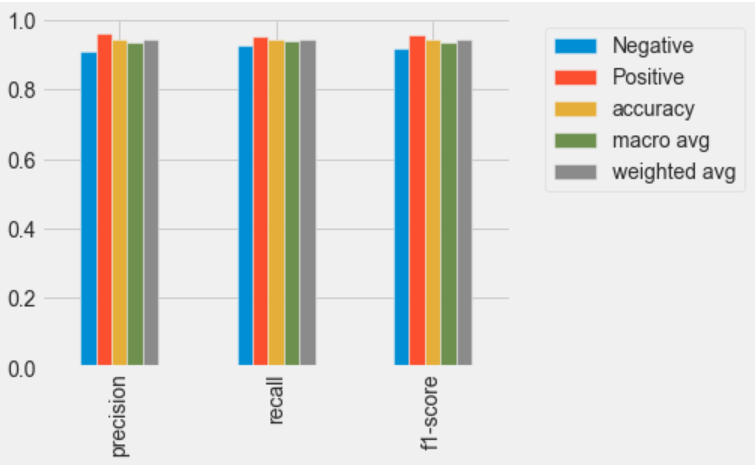


Fig 9: classification report for testing adaboost algorithm on the dataset

1. **COMPARATIVE STUDY:**

After comparing all the above used algorithms, we obtained highest accuracy by Random forest algorithm, which gave training accuracy of 100% followed by testing accuracy of 99.36%.

Table 7: different training and testing accuracies of the algorithms used.

| Algorithm | Training accuracy | Testing accuracy |
| --- | --- | --- |
| Adaboost | 93.95% | 94.23% |
| Random forest | 100% | 99.35% |
| Logistic Regression | 93.95% | 94.23% |

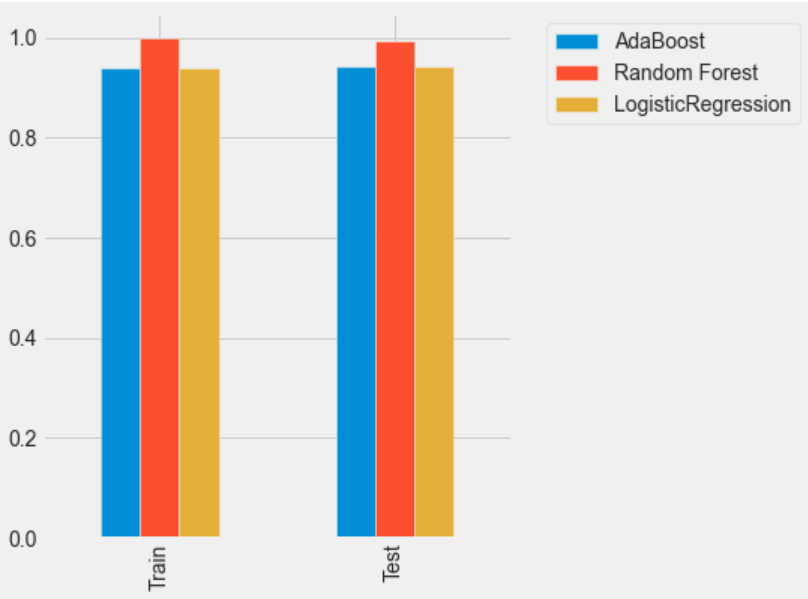


Fig 10: comparative graph of accuracies of the machine learning algorithms

1. **CONCLUSION:**

In this paper, we used a diabetes dataset obtained from UCI machine learning repository to apply and compare machine learning algorithms like Random forest, Adaboost and Logistic regression. We used train-test split method for applying algorithms and the best performance was given by Random Forest algorithm among all the algorithms giving training accuracy of 100% and testing accuracy of 99.35%.

1. **LIMITATIONS:**

The size of the dataset used is very small thus the accuracies obtained may show deviation when applied on larger datasets as machine learning algorithms require very large datasets to give accurate results which are highly required by the medical industry.

1. **FUTURE SCOPE:**

This study can be extended by using a larger dataset and using more attributes in the dataset. Further we can use this model on other chronic diseases which if detected or predicted earlier at an earlier stage can save lives of various people.

**REFERENCES:**

1. Naveen Kishore, G., Rajesh, V., Vamsi Akki Reddy, A., Sumedh, K., & Rajesh Sai Reddy, T. Prediction of diabetes using machine learning classification algorithms. *Int J Sci Technol Res*, *9*(01), 1805-1808 (2020).
2. Khanam, J. J., & Foo, S. Y. A comparison of machine learning algorithms for diabetes prediction. *ICT Express*, *7*(4), 432-439. (2021).
3. El Massari, H., Sabouri, Z., Mhammedi, S., & Gherabi, N. Diabetes Prediction Using Machine Learning Algorithms and Ontology. *Journal of ICT Standardization*, 319-338 (2022). Doi: 10.13052/jicts2245-800X.10212
4. Sisodia, D., & Sisodia, D. S. Prediction of diabetes using classification algorithms. *Procedia computer science*, *132*, 1578-1585. (2018). Doi: 10.1016/j.procs.2018.05.122.
5. Shafi, S., & Ansari, G. A. Early prediction of diabetes disease & classification of algorithms using machine learning approach. In *Proceedings of the International Conference on Smart Data Intelligence (ICSMDI 2021)*. (2021, May).
6. L. Ismail and H. Materwala, "Comparative Analysis of Machine Learning Models for Diabetes Mellitus Type 2 Prediction," 2020 International Conference on Computational Science and Computational Intelligence (CSCI), 2020, pp. 527-533, doi: 10.1109/CSCI51800.2020.00095.
7. Llaha, O., & Rista, A.. Prediction and Detection of Diabetes using Machine Learning. In *RTA-CSIT* (pp. 94-102) (2021, May).
8. M. A. R. Refat, M. A. Amin, C. Kaushal, M. N. Yeasmin and M. K. Islam, "A Comparative Analysis of Early Stage Diabetes Prediction using Machine Learning and Deep Learning Approach," *2021 6th International Conference on Signal Processing, Computing and Control (ISPCC)*, 2021, pp. 654-659, doi: 10.1109/ISPCC53510.2021.9609364.
9. Abaker, A. A., & Saeed, F. A. A comparative analysis of machine learning algorithms to build a predictive model for detecting diabetes complications. *Informatica*, *45*(1) (2021). doi:10.31449/inf.v45i1.3111
10. Mitushi, S., & Sunita, V.Diabetes prediction using machine learning techniques. *International Journal of Engineering Research & Technology (IJERT)*. (2020).
11. Murthy, M. Ramakrishna, P. Ruchita, Ch Bharat Teja, M. Manoj Kumar, and T. V. S. Lingeswararao. "DIABETES PREDICTION USING MACHINE LEARNING ALGORITHMS."
12. Joshitta, R. S. M., & Arockiam, L. A predictive model to forecast and pre-treat diabetes mellitus using clinical big data in cloud. *International Journal of Applied Engineering Research*, *10*(82), 2015.
13. Mujumdar, A., & Vaidehi, V. Diabetes prediction using machine learning algorithms. *Procedia Computer Science*, *165*, 292-299 (2019).
14. Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., & Tang, H. (2018). Predicting diabetes mellitus with machine learning techniques. *Frontiers in genetics*, *9*, 515. (2018). DOI=10.3389/fgene.2018.00515
15. Xue, J., Min, F., & Ma, F. (2020, November). Research on diabetes prediction method based on machine learning. In *Journal of Physics: Conference Series* (Vol. 1684, No. 1, p. 012062). IOP Publishing. (2020, November).
16. Butt, U. M., Letchmunan, S., Ali, M., Hassan, F. H., Baqir, A., & Sherazi, H. H. R. Machine learning based diabetes classification and prediction for healthcare applications. *Journal of healthcare engineering*, *2021*.
17. <https://www.kaggle.com/datasets/ishandutta/early-stage-diabetes-risk-prediction-dataset>
18. <https://towardsdatascience.com/how-to-use-seaborn-for-data-visualization-4c61fc488ec1>
19. Nugraha, G. D., Sudiarto, B., & Ramli, KMachine learning-based energy management system for prosumer. . (2020).
20. M.K. Barai, and B.B. Saha, “Energy security and sustainability in japan,” Evergreen, 2 (1) 49–56 (2015). doi:10.5109/1500427
21. T. Sato, “How is a sustainable society established? a case study of cities in japan and germany,” Evergreen, 3 (2) 25–35 (2016). doi:10.5109/1800869
22. T.Fujisaki, "Evaluation of Green Paradox:Case Study of Japan,"Evergreen 5(4) 26-31(2018).
23. MD. Ahsan Habib, K. M. Ariful Kabir, Jun Tanimoto,“Do humans play according to the game theory when facing the social dilemma situation?” A survey study,”Evergreen 7(1) 7-14(2020). Doi: 10.5109/2740936
24. Yadav, V. K., Yadav, V. K., & Yadav, J. P. Cognizance on pandemic corona virus infectious disease (covid-19) by using statistical technique: a study and analysis. (2020). doi.org/10.5109/4068611
25. Choubey, D. K., Paul, S., Kumar, S., & Kumar, S. Classification of Pima indian diabetes dataset using naive bayes with genetic algorithm as an attribute selection. In *Communication and computing systems: proceedings of the international conference on communication and computing system (ICCCS 2016)* (pp. 451-455) (2017, February).