IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES
ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 12, Iss 03, 2023

PREDICTION OF ILLNESS IN AGED PEOPLE: ADABOOST ALGORITHM

Dr. S. Jesssica Saritha
Assistant Professor, Dept of CSE, JNTUA College of Engineering Pulivendula.
sjsaritha.cse@jntua.ac.in

Abstract: Machine Learning is widely used to program and develop computer systems that are able to perform without explicit instructions, by using existing algorithms to analyse and find patterns in data. Machine learning uses existing algorithms and makes a prediction. If the predictions made are ambiguous and not so clear we use meta learning for the predictions. Meta-learning learns and makes a prediction of predictions made by machine learning algorithms. Simply, Meta-Learning makes the best use of predictions made by machine learning algorithms to make an accurate prediction. This research makes use of algorithm like adaboost to boost the performance of existing machine learning algorithms. Simply put it is used to achieve higher accuracy rate for the existing machine learning algorithms or models. Adaboost algorithm mostly uses a one split decision trees for ensembling. Abstract Boosting is an approach to machine learning based on the thought of making a profoundly precise forecast run the show by combining numerous generally powerless and wrong rules. In this model we predict the illness of five individuals of middle aged using adaboost model.

Keywords: Meta-Learning, Adaboost, Weights, Machine Learning, Cumulative weights, Ensembling.

I. INTRODUCTION

Meta-Learning can be used to predict and define outcomes of repetitive process with variable inputs in each step. The steps in each process is same whereas the performance in each step may vary. To predict the best among people, teams or workers, We use meta-learning algorithms like Adaboost adjusts adaptively the errors of the weak hypotheses by weak learner. „ Not at all like the customary boosting calculation, the earlier blunder require not be known ahead of time. „ The upgrade run the show diminishes the likelihood doled out to those illustrations on which the speculation makes a great forecasts and increments the likelihood of the cases on which the expectation is destitute.The previous discussion is restricted to binary classification problems. The traing data could have any number of labels, which is a multi-class problems. „ The multi-class case (AdaBoost.M1) requires the accuracy of the weak hypothesis greater than ½. This condition in the multi-class is stronger than that in the binary classification cases.

To get it adaboost the exceptionally common and including vc hypothesis is the first sensible starting point all examinations of learning techniques depend in some way on assumptions since something else learning is exceptionally incomprehensible from the especially beginning much of the work looking at boosting has been based on the doubt that each of the feeble hypotheses has exactness reasonable a little bit predominant than self-assertive hypothesizing for two-class issues. Such an analysis predicts the kind of behaviour appears the mistake both preparing and test of the ultimate theory as a work of the number of rounds of boosting as

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 12, Iss 03, 2023

famous we anticipate preparing mistake to drop exceptionally rapidly but at the same time the vc-dimension of the ultimate theory is expanding generally straightly with the number of rounds t in this way with progressed fit to the preparing set the test blunder drops at to begin with but then rises once more as a result of the ultimate speculation getting to be excessively complex this can be classic overfitting behavior in fact overfitting can happen with adaboost as seen on the proper side of the figure which appears preparing and test mistake on an genuine benchmark dataset be that as it may as we are going see in no time adaboost regularly does not overfit clearly in coordinate inconsistency of what is anticipated by vc theory summarizing this to begin with approach to understanding adaboost a coordinate application of vc hypothesis shows up that adaboost can work within the occasion that given with adequate data and essential slight classifiers which fulfill the slight learning condition and in case run for en sufficient but not as well numerous rounds. The hypothesis captures the cases in which AdaBoost does overfit, but too predicts (inaccurately) that AdaBoost will continuously overfit. Like all of the approaches to be examined in this chapter, the numerical bounds on generalization blunder that can be gotten utilizing this method are unpleasantly free.

II. LITERATURE SURVEY

Adaboost algorithm:

The AdaBoost algorithm of Freund and Schapire [10] was the primary down to earth boosting calculation, and remains one of the foremost broadly utilized and examined, with applications in various areas. Over the a long time, a extraordinary assortment of endeavors have been made to "explain" AdaBoost as a learning calculation, that's , to get it why it works, how it works, and when it works (or falls flat). It is by understanding the nature of learning at its establishment — both for the most part and with respect to specific calculations and wonders — that the field is able to move forward. In fact, this has been the lesson of Vapnik's life work. AdaBoost can be analyzed hypothetically along precisely these lines. It is conceivable to demonstrate to begin with a bound on the generalization mistake of AdaBoost — or any other voting strategy — that depends as it were on the edges of the preparing cases, and not on the number of rounds of boosting. Such a bound predicts that AdaBoost will not overfit notwithstanding of how long it is run, given that expansive edges can be accomplished (and given, of course, that the powerless classifiers are not as well complex relative to the measure of the preparing set).

III. ARCHITECTURE

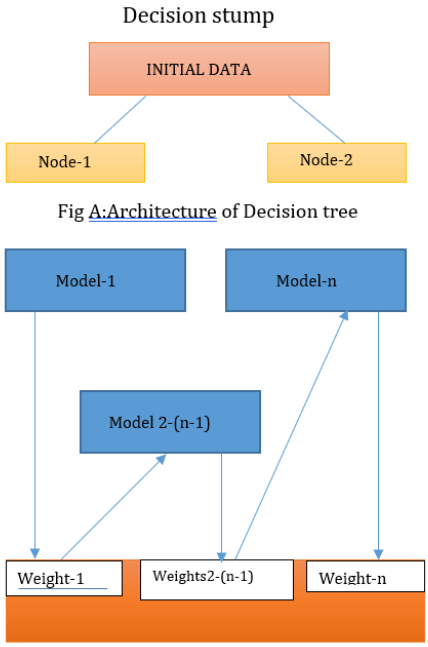


Fig A: Architecture of Decision tree

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 12, Iss 03, 2023

The output data by existing machine learning algorithms is analysed and the respective weights are added to each participant and corresponding metrices of particular algorithm are calculated. The above steps repeat for n number of times until we get the desired output. We update weights for every respective rounds based on participant performance and find out the one with least error. In this model we get the ambiguous output for some rounds. We repeat the same steps until we get the right output with least error in prediction.

IV. ANALYSIS

The objective of analysis step is to increase the understanding of the problem from the data. To get the most accurate prediction from the output we calculate and analyse the weights of each participant.

Dataset information

The attributes in the dataset include:

Weights – Initially, all the weights will be equal.

The formula to calculate the sample weights is:

$$w(x_i, y_i) = \frac{1}{N}, \; i = 1, 2, \ldots . n$$

Adaboost algorithm :

We'll create a decision stump for each of the features and then calculate the **Gini Index** of each tree. The tree with the lowest Gini Index will be our first stump.

The full mistake is nothing, but the summation of all the test weights of misclassified information points.

The off-base predictions will be given more weight though the proper expectations weights will be diminished. Presently when we construct our another show after overhauling the weights, more inclination will be given to the focuses with higher weights.

V. METRICS

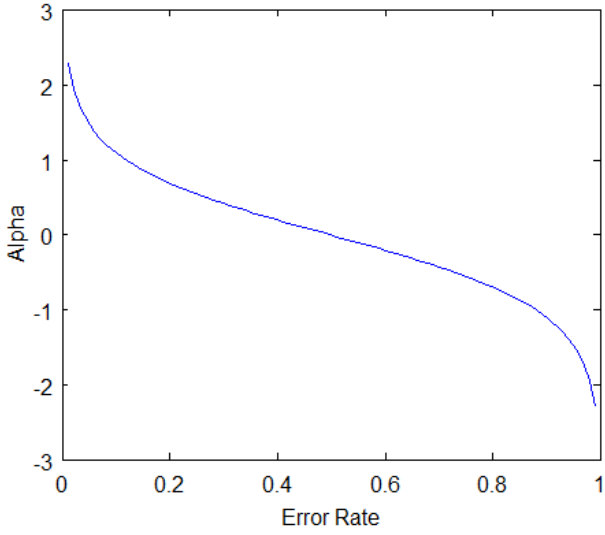
Here we use ,

Gini Index: The Decision tree with least Gini Index will be our first stump.

Amount of say:

$$\frac{1}{2}\log\frac{1 - Total\ Error}{Total\ Error}$$

0 Indicates perfect stump and 1 indicates horrible stump.



VI .PREDICTION

The implementation process is same at every step.

The example is shown below.

Row No.	Gender	Age	Income	Illness	Sample Weights
1	Male	41	40000	Yes	1/5
2	Male	54	30000	No	1/5
3	Female	42	25000	No	1/5
4	Female	40	60000	Yes	1/5
5	Male	46	50000	Yes	1/5

ISSN PRINT 2319 1775 Online 2320 7876
 UGC CARE Listed (Group -I) Journal Volume 12, Iss 03, 2023

Row No.	Gender	Age	Income	Illness	Sample Weights	New Sample Weights
1	Male	41	40000	Yes	1/5	0.1004
2	Male	54	30000	No	1/5	0.1004
3	Female	42	25000	No	1/5	0.1004
4	Female	40	60000	Yes	1/5	0.3988
5	Male	46	50000	Yes	1/5	0.1004

The calculated metrices of the above data:

Row No.	Gender	Age	Income	Illness	New Sample Weights	Buckets
1	Male	41	40000	Yes	0.1004/0.8004= 0.1254	0 to 0.1254
2	Male	54	30000	No	0.1004/0.8004= 0.1254	0.1254 to 0.2508
3	Female	42	25000	No	0.1004/0.8004= 0.1254	0.2508 to 0.3762
4	Female	40	60000	Yes	0.3988/0.8004= 0.4982	0.3762 to 0.8744
5	Male	46	50000	Yes	0.1004/0.8004= 0.1254	0.8744 to 0.9998

Row No.	Gender	Age	Income	Illness
1	Female	40	60000	Yes
2	Male	54	30000	No
3	Female	42	25000	No
4	Female	40	60000	Yes
5	Female	40	60000	Yes

VII. CONCLUSION

After repetition for required number of rounds, the female of aged 40 is more prone of illness by using adaboost algorithm.

VIII. REFERENCES

1. Bartlett, P.L., Traskin, M.: AdaBoost is consistent. Journal of Machine Learning Research 8, 2347–2368 (2007)
2. Bickel, P.J., Ritov, Y., Zakai, A.: Some theory for generalized boosting algorithms. Journal of Machine Learning Research 7, 705–732 (2006)
3. Breiman, L.: Population theory for boosting ensembles. Annals of Statistics 32(1), 1–11 (2004)
4. Dietterich, T.G.: An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine Learning 40(2), 139–158 (2000)
5. Freund, Y.: An adaptive version of the boost by majority algorithm. Machine Learning 43(3), 293–318 (2001)
6. Jiang, W.: Process consistency for AdaBoost. Annals of Statistics 32(1), 13–29 (2004)
7. Onoda, T., Ratsch, G., M ¨ uller, K.R.: An asymptotic analysis of AdaBoost in the binary classifi- ¨ cation case. In: Proceedings of the 8th International Conference on Artificial Neural Networks, pp. 195–200 (1998)
8. . Ratsch, G., Onoda, T., M ¨ uller, K.R.: Soft margins for AdaBoost. Machine Learning ¨ 42(3), 287–320 (2001)
9. . Schapire, R.E., Singer, Y.: Improved boosting algorithms using confidence-rated predictions. Machine Learning 37(3), 297–336 (1999)
10. Mease, D., Wyner, A.: Evidence contrary to the statistical view of boosting. Journal of Machine Learning Research 9, 131–156 (2008)
11. . Schapire, R.E., Singer, Y.: Improved boosting algorithms using confidence-rated predictions. Machine Learning 37(3), 297–336 (1999)
12. Zhang, T., Yu, B.: Boosting with early stopping: Convergence and consistency. Annals of Statistics 33(4), 1538–1579 (2005)
13. Explaining AdaBoost Robert E. Schapire