Class Result Prediction using Machine Learning

Pushpa S K
Associate Professor,
Dept. of ISE
BMSIT&M
Bangalore, India
pushpask@bmsit.in

Manjunath T N
Professor and Head
Dept. of ISE
BMSIT&M
Bangalore, India
hod_ise@bmsit.in

Mrunal T V
Dept. of ISE
BMSIT&M
Bangalore, India
mrunaltv@yahoo.co.in

Amartya Singh
Dept. of ISE
BMSIT&M
Bangalore, India
amartya.singh@gmail.com

C Suhas
Dept. of ISE
BMSIT&M
Bangalore, India
csuhas22@gmail.com

Abstract— **More than 2.5 quintillion bytes of data is being generated across the globe. In fact, this data is as much as 90% of the data in the world today, and has been created in the last two years alone. Big data describes the large volume of data that inundates a business on a day to day basis. Huge amount of data is being generated by everything around us at all times and is produced by every digital process and social media exchange through systems, sensors, mobile devices, etc. Big data analytics examines large amounts of data to uncover hidden patterns, correlations and other insights. To extract meaningful value from big data, one needs optimal processing power, analytics capabilities and skills. Using the concept of machine learning, a number of algorithms are explored in order to predict the result of class students. Based on the performance of the students in previous semester, and the scores of internal examinations of the current semester, the final result, whether the student passes or fails the current semester is computed before the final examination actually takes place.**

Keywords— ***Machine Learning, Class Result Prediction, Predictive Accuracy, Internal Scores, External Scores, Supervised Learning.***

I. Introduction

Since data, information and statistics available through Internet are increasing exponentially at a very rapid pace it is no longer possible to manage so much of information in the traditional way. Also, correct handling of global data, which can be available at the touch of a button, can open avenues for business, research and education as never before. But the question is how to correctly utilize the data without getting drowned in sheer numbers. The answer lies in Analytical Sciences or Big data, which has become a buzzword. Machine learning has become an integral part of many commercial application and research projects, but this field is not exclusive to large companies with extensive research items [1].

Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs that can change when exposed to new data [4]. Some of the applications like Pedestrian Detection [12], Multiple object recognition with visual attention [14], Neural networks behind Google voice & Google web search using AI concepts [11], Deep neural networks have been successful in solving parallel processors and math expression compilers [16], and many more. Using the concept of machine learning, a number of algorithms are explored in order to predict the result of class students. In Machine Learning, this problem is of classification type. Hence, the various supervised learning algorithms such as Support Vector Machine, Naïve Bayes Classifier, Random Forest Classifier, and Gradient Boosting Algorithm are used. The accuracy obtained by each of the algorithms are then compared in order to identify the algorithm that is most suitable for the problem.

II. LITERATURE SURVEY

Machine learning is a new approach to learn and analyze a complex and huge data. It is based on algorithms that can learn from data without relying on the conventional programming, i.e., rules-based programming. It emerged individually as a scientific discipline in the late 1990s as steady advances in digitization and cheap computing power enabled data. Scientists have stopped building finished models and have plunged into a novel adventure in training computers, through which an unmanageable volume of data and complexity of the big data can be processed and analyzed using the potentiality of machine learning. Hence machine learning is the emerging trend in the era of information technology.

Mekinsey Quarterly, in one of the article writes that statistical inference forms an important foundation for the current implementation of Artificial Intelligence [9]. Machine Learning results in higher degree of accuracy, in scenarios where human analytics could not visualize data on their own

978-1-5386-0569-1$31.00 ©2017 IEEE

and make predictions. Prescription stage of machine learning, ushering in a new era of man machine collaboration, will require the biggest change in the way we work.

Tom M. Mitchell describes some of the applications that are routinely used in various areas of computer science like machine learning algorithms for speech recognition, computer vision, bio surveillance, robot control and variety of other tasks, and has been considered in discovering hidden regularities in the continuously growing volumes of online data [10].

In particular, learning and evaluating such models have a variety of challenges like machine learning skills in domain area, collection of data and algorithmic complexity, etc. In this paper we have made an attempt to predict results of a batch of students based on the previous performance.

III. Methodology

The application of machine learning has been portrayed here. The general pipeline used for essentially all machine learning problems consists of [2]:

1. Define the problem
2. Collect data.
3. Design features.
4. Train the model.
5. Test the model.

The problem considered here is to predict the fourth semester results of the students based on the results of the students in their third semester, and current internal examination scores.

The data collected from BMSIT & M for the 2014-18 batch students of Information Science and Engineering is used as the sample.

The data consists of three features for each subject, the internal score, the external score, and the total score. The final feature shows the status of the result, whether the student passed or failed the semester. The criterion for this is for the student to score a minimum of 35 in external examination and a minimum total of 50 in every subject. The model is built to predict the results based on this criterion.

The third semester scores and the results are used as the sample for training the model. The internal scores is provided as the data input for the training model, and the result is given as the output. Hence, the training model is built on the dependability of the final result on the internal scores of the students.

The trained model is tested on the fourth semester results. The internal scores of the student is given as the input for the machine, and the final result is the output that is predicted by the machine.

The predicted result by the trained model with each of the algorithms is then compared with the actual results to compute the accuracy of the models for the selected problem.

A. *Support Vector Machine*

In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier [3].

In this algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes.

When two classes of data are linearly separable, infinitely many hyper-planes could be drawn two separate the two classes. All these hyper-planes can classify the data into two classes, and the best among all the hyper-planes is selected by the SVM classifier for the prediction model. One reasonable standard for judging the quality of these hyper-planes is via their margin lengths [2].

Since the output of the problem considered is a factor of two, i.e. pass or fail, SVM algorithm can be used to predict the required results.

B. *Naïve Bayes Classifier*

In machine learning, Naïve Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. It is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. It is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable [5].

One of the main assumptions of the Naïve Bayes algorithm is that each feature is independent, which holds good for the problem considered, since the score of the student in each subject is independent, though it could be related with similar subjects. Due to this assumption, this classifier is very effective for this problem.

C. *Random Forest Classifier*

Random Forests grows many classification trees. To classify a new object from an input vector, put the input vector down each of the trees in the forest. Each tree gives a classification, and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest) [6].

Random forests are very fast, and they do not over fit. The user can also manually run as many trees as they want in their classifier. Since the computation is done with a number of classifiers and the best among them is chosen by the model, theoretically, Random Forest Classifiers are more accurate and efficient than individual Naïve Bayes classifiers.

The output obtained by modifying the number of trees is tabulated, and the optimal value for the number of trees is chosen. This value is considered as the accuracy for the Random Forest classifier.

D. Gradient Boosting

Gradient boosting was developed by Stanford professor Jerome Friedman. Gradient boosting develops an ensemble of tree-based models by training each of the trees in the ensemble on different labels and then combining the trees [7].

Boosting is an ensemble learning algorithm which combines the prediction of several base estimators in order to improve robustness over a single estimator. It combines multiple weak or average predictors to a build strong predictor [8].

IV. Results

The models created by each of the algorithms are used to create an output for each of the student. This output is then compared with the actual results of that semester, and the accuracy is determined.

First, the number of trees and the output accuracy is tabulated for the Random Forest classifier, and the optimal number is selected.

TABLE I. Random forest trees

Random Forest Classifier	*Number of trees*	*Accuracy (%)*
1.	1	84.375
2.	2	71.875
3.	5	87.5
4.	10	89.0625
5.	20	85.9375
6.	50	82.8125

a. Table representing the number of trees & the corresponding accuracy.

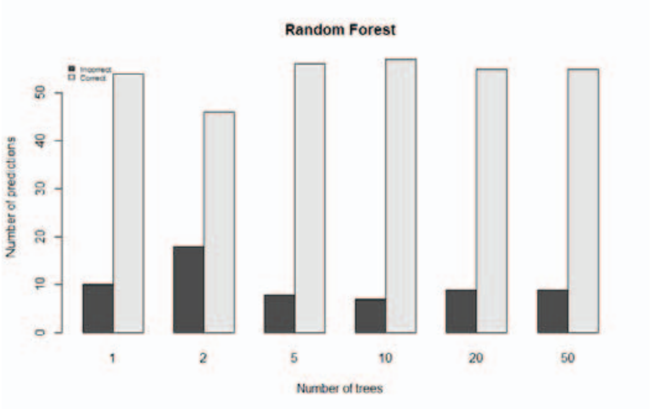

Fig. 1. Number of trees used in the RF Classifier, & corresponding accuracy.

From Fig. 1, the optimal value for the number of trees is 10. Hence, the corresponding accuracy is selected for the Random Forest classifier.

Now, barplot representing the actual result of each student is presented. The students are represented on the X-axis and the result is represented in the Y-axis. These plots show the clear distinction in the inaccuracies of each of the predicted outputs by the models.

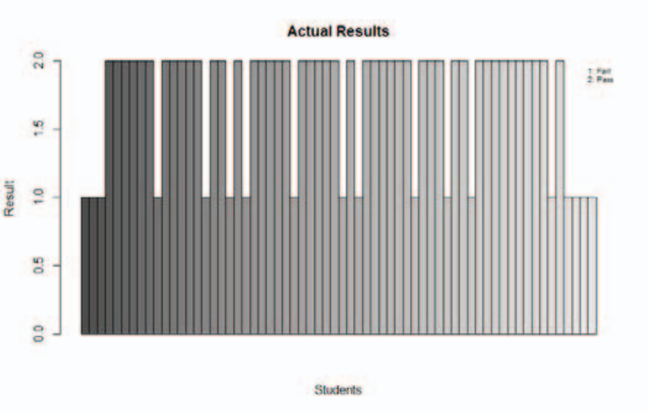

Fig. 2. Actual results of the class.

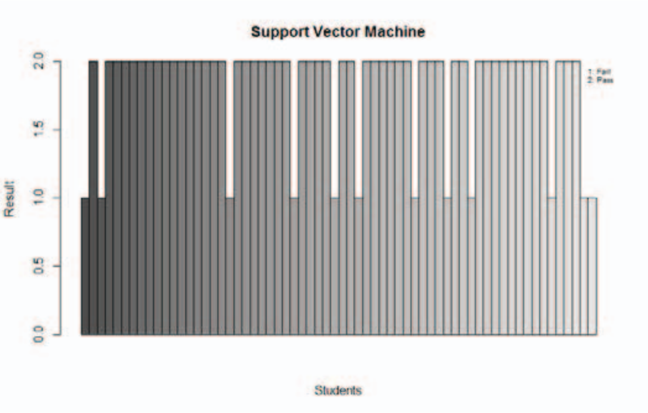

Fig. 3. Predicted result by the SVM algorithm.

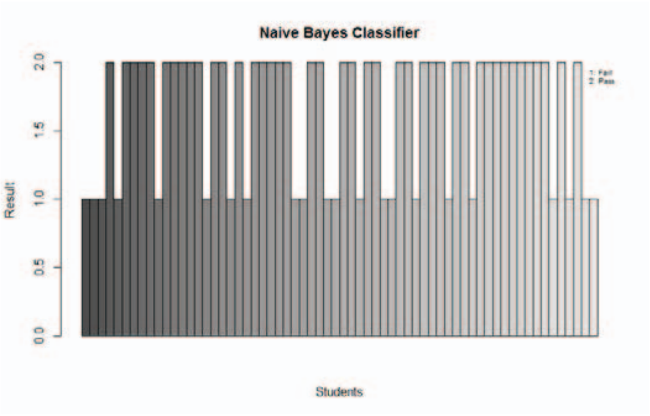

Fig. 4. Predicted result by the Naïve Bayes Classifier.

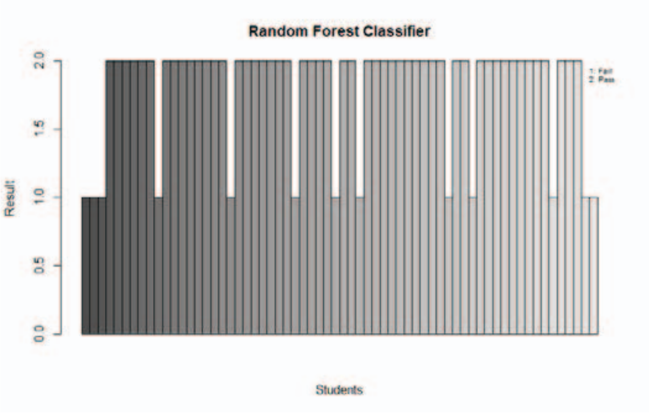

Fig. 5. Predicted result by the Random Forest Classifier.

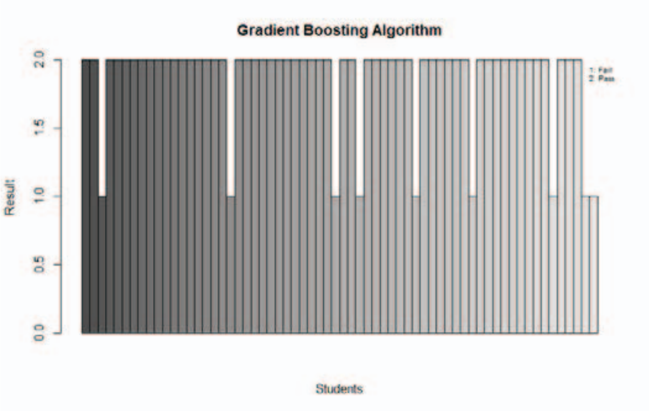

Fig. 6. Predicted result by the Gradient Boosting Algorithm.

The results predicted by each of these algorithms, when compared with the actual result of that semester, will give the accuracy of the predictions. The predictive accuracy of each of these algorithms is computed and is tabulated below.

TABLE II. PREDICTIVE ACCURACY OF THE ALGORITHMS

Serial No.	*Algorithm used*	*Accuracy (%)*
1.	Support Vector Machine	87.5
2.	Naïve Bayes Classifier	87.5
3.	Random Forest Classifier	89.0625
4.	Gradient Boosting	82.8125

b. Table representing the various algorithms used & the corresponding accuracy.

V. CONCLUSION

Machine learning has emerged as one of the most popular fields of Artificial Intelligence. The main cause for this is that development of machine learning leads to improved speed and accuracy of the functions performed by the system. In the future, machine learning could be applied to analyze the performance of the students, and could be used as a powerful tool for the analysis of academics.

The problem was created based on the assumption that a pattern exists in the results of every student. It suggests that the current result of the students is reliant on previous results. The results obtained here confirm that the pattern does exist. Although a lot of other factors influence the final result of a student, the high accuracy obtained by the machine learning models suggests that the internal scores secured by the students is one of the vital features in deciding the final result of the student.

Since the problem has a small data set, the obtained accuracy by each of the algorithms are analogous. From the results obtained, we can conclude that Random Forest Classifier gives the most accurate predictive model, by a small margin.

ACKNOWLEDGMENT

We thank the department of Information Science and Engineering, BMS Institute of Technology for providing us the data required.

REFERENCES

[1] Andreas C. Müller, Sarah Guido. Introduction to Machine Learning with Python. O'Reilly Media, September 2016.

[2] Aggelos Konstantinos Katsaggelos, Jeremy Watt, and Reza Borhani. Machine Learning Refined: Foundations, Algorithms, and Applications.

[3] Support vector machine, from Wikipedia, the free encyclopedia.
https://en.wikipedia.org/wiki/Support_vector_machine

[4] Definition of Machine Learning. http://whatis.techtarget.com/definition/machine-learning

[5] Naive Bayes classifier, from Wikipedia, the free encyclopedia.
https://en.wikipedia.org/wiki/Naive_Bayes_classifier

[6] Leo Breiman and Adele Cutler, Random Forests. https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm

[7] Micheal Bowles, Machine Learning in Python: Essential Techniques for Predictive Analysis. John Wiley & Sons, Inc. 2015

[8] Sunil Ray, Common Machine Learning Algorithms. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2015/08/common-machine-learning-algorithms/

[9] Mekinsey Quartely "An executive's guide to machine learning "

[10] Tom.M Mitchell, "The Discipline of Machine Learning", July 2006.

[11] Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Man´e, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Vi´egas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org

[12] Anelia Angelova, Alex Krizhevsky, and Vincent Vanhoucke. Pedestrian detection with a large-field-of-view deep network. In Robotics and Automation (ICRA), 2015 IEEE International Conference on, pages 704–711. IEEE, 2015. CalTech PDF.

[13] Arvind and Rishiyur S. Nikhil. Executing a program on the MIT tagged-token dataflow architecture. IEEE Trans. Comput., 39(3):300–318, 1990. dl.acm.org/citation.cfm?id=78583.

[14] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. arXiv preprint arXiv:1412.7755, 2014. arxiv.org/abs/1412.7755.

[15] Franc¸oise Beaufays. The neural networks behind Google Voice transcription, 2015. googleresearch.blogspot.com/2015/08/the-neuralnetworks-behind-google-voice.html.

[16] James Bergstra, Olivier Breuleux, Fr´ed´eric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: A CPU and GPU math expression compiler. In Proceedings of the Python for scientific computing conference (SciPy), volume 4, page 3. Austin, TX, 2010. UMontreal PDF.

[17] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cuDNN: Efficient primitives for deep learning. arXiv preprint arXiv:1410.0759, 2014. arxiv.org/abs/1410.0759.

[18] Trishul Chilimbi, Yutaka Suzue, Johnson Apacible, and Karthik Kalyanaraman. Project Adam: Building an efficient and scalable deep learning training system. In 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14), pages 571–582, 2014. www.usenix.org/system/files/conference/osdi14/osdi14-paper-chilimbi.pdf.

[19] Jack Clark. Google turning its lucrative web search over to AI machines, 2015. www.bloomberg.com/news/articles/2015-10-26/googleturning- its-lucrative-web-search-over-to-ai-machines.