Journal of Advances and Scholarly Researches in Allied Education
Vol. 21, Issue No. 7, October-2024, ISSN 2230-7540

https://doi.org/10.29070/9b2tdk74

Analyze the Factors that Contribute to Accurate performance Predictions, Including Data Preprocessing, Feature Selection, and Model Hyperparameters

Vinod K C[1]*, Dr. Suhas Rajaram Mache[2]

[1] Research Scholar, University of Technology, Jaipur, Rajasthan, India

Email: kckcvkc@gmail.com

[2] Professor, Department of Computer Science, University of Technology, Jaipur, Rajasthan, India

Abstract - A solid academic record boosts a university's standing and promotes student career chances, hence predicting academic performance has attracted attention in education. Using clusters obtained by Davies' Bouldin approach, a clustering data mining technique known as K-means is used in this study to identify critical characteristics impacting students' performance. Machine learning techniques find use in many fields, including medical diagnostics, image processing, cluster analysis, pattern identification, and natural language processing. Among the algorithms tested, SVM produced the most accurate predictions (96% accuracy rate) after parameter tweaking. The researchers in this study have looked at how the SVM, Decision Tree, naive Bayes, and KNN classifiers work. The results of adjusting the parameters significantly improved the four prediction models' accuracy. Feature selection algorithms and hyperparameter optimisation, two critical components for enhancing model performance, are also addressed. The findings highlight the need of carefully evaluating models, with Random Forest emerging as a dependable choice for accurate diabetes prediction.

Keywords: Prediction, Parameter tuning, Feature Selection, model hyperparameters

----------X----------

INTRODUCTION

The art of "machine learning," or ML, is "the study of how computers learn to do tasks that do not require explicit programming." Data access criteria include things like data usage agreements, a complete protocol that needs to be prepared and approved, a data request form that needs to be finalised, an ethical assessment that needs to be accepted, and the cost of obtaining datasets that aren't in the public domain. The promise of synthetic data to provide access to real-time healthcare data for study and technological development has piqued the interest of many in the field of artificially generated data (or simply data). Healthcare providers have long been experts in making data-driven prognostic projections and risk factor assessments, thus prediction is nothing new in their field. When it comes to making accurate predictions, machine learning methods can beat the traditional regression models. A model that may be used to estimate the likelihood of a certain illness outcome for a given patient is called a prediction model. As more and more prediction models become available, the issue of when, what, and how to employ them naturally emerges.

The values of hyperparameters, which are parameters, are determined before training begins. Improving machine learning models' efficiency is as simple as tweaking these hyperparameters. To fine-tune hyperparameters and enhance model generalizability, one often uses Bayesian optimisation, grid search, and random search. Intelligent manufacturing cannot function without the sensors. The process begins with collecting sensor signals, continues with a number of data processing steps to extract the relevant information, and culminates with feeding the collected data into an artificial intelligence model for subject classification or clustering.

LITERATURE REVIEW

Frank Hutter et.al (2014) Unexpectedly, one might anticipate the time required to execute an algorithm on an unknown input by constructing a model of the method's runtime based on problem-specific instance information using machine learning techniques. The automated construction of parameterized algorithms, portfolio-based algorithm selection, and algorithm analysis are three key areas

that greatly benefit from these models. Many different methods for creating these models have been investigated within the last ten years. In this paper, we detail new model families, enhanced and expanded versions of current ones, and, most crucially, a far more comprehensive approach to using algorithm parameters as inputs to models. Additionally, we detail both new and old features for forecasting the runtime of algorithms for propositional satisfiability (SAT), travelling salesperson (TSP), and mixed integer programming (MIP) issues. In order to assess these advancements, we compared them to several runtime modelling methodologies found in the literature and conducted the biggest empirical study of its kind. Our experimental results include a large spectrum of SAT, MIP, and TSP examples, with the least structured instances having been created uniformly at random and the most structured cases having developed from genuine industrial applications. Eleven algorithms and thirty-five instance distributions are considered. Our novel models outperform earlier methods in terms of runtime forecasts and generalizability to new problem cases, parameterized space algorithms, and both at the same time.

Ashir Javeed et.al (2020) Researchers have developed a number of clever diagnostic technologies to aid with the challenging task of heart disease diagnosis. Unfortunately, these technologies still have an issue with poor accuracy when it comes to predicting cardiac disease. Our proposed feature selection approach, FWAFE, employs a floating window with adjustable size to improve the accuracy of heart risk prediction. Following the feature extraction, two classification frameworks—deep neural networks (DNN) and artificial neural networks (ANN)—are used. Hence, FWAFE-ANN and FWAFE-DNN are the two hybrid diagnostic systems suggested in this work. Using data obtained from the Cleveland online cardiac disease database, experiments are conducted to evaluate the efficacy of the suggested strategies. It is the receiver operating characteristics (ROC) curve, accuracy, sensitivity, and specificity that are used to evaluate the suggested procedures. The results of the experiments show that the suggested models are more effective than the eighteen previous techniques that were presented, with accuracies ranging from 50.00 to 91.83%. When compared to other cutting-edge machine learning methods for diagnosing cardiac disease, the suggested models also perform very well. In addition, the suggested approaches may aid doctors in making precise diagnoses of cardiac disease.

Isabella M. Tromba (2018) Knowledge workers without coding skills can now rapidly and simply build machine learning models using Make ML, and these models can compete with those of skilled data scientists who build them by hand. For feature selection and target column prediction, Make ML provides a web-based tool resembling a spreadsheet. Then, Make ML takes care of robotically designing features, selecting models, training, and optimising hyperparameters. An evaluation of the model's performance and the ability to generate predictions on fresh data may be done using the web interface when training is complete. We demonstrate that ninety percent of the Titanic issue submissions on the public data science platform Kaggle achieve accuracy better than a machine produced model using Make ML.

Subhash Chandra Gupta et.al (2021) It is crucial for a diabetes prediction model to have a classifier with good prediction abilities. Performance of the model is directly proportional to the accuracy of the classifier's predictions. Plenty of studies have looked at this, but there's always room to make the model better. In this experimental study, we try to do it by using three methods: finding the right preprocessing action, oversampling to create a balanced class dataset, and tuning the hyperparameter of classifiers to make them operate better. Procedures: Using the PIMA diabetes dataset and a variety of preprocessing procedures, four separate prediction models were constructed using oversampled balanced class datasets. In order to distinguish between diabetes and non-diabetic data, each model employs hypertuned classifiers such as KNN, SVM, DT, and random forest. After logging and analysing the data, the optimal model is chosen by taking into account the F1 score of the classifiers in all prediction models. The findings are as follows: for datasets D1, D2, D3, and D4, the greatest F1score of classifiers for each model is 88.52%, 88.59%, 93.33%, and 95.23%, respectively. This is accomplished for every model using the random forest classifier. Conclusion: After analysing the findings from various models, it was determined that dataset D4, which was constructed by removing outliers and rows with missing values during preprocessing, had the best prediction model.

METHODOLOGY

First, there is preprocessing; second, there is model selection; third, there is model training; fourth, there is parameter tuning; and last, there is prediction. Our planned work is shown in Figure 1 as a block diagram.

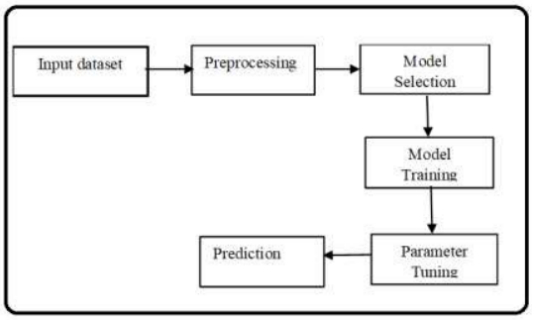

Figure 1: Proposed System

To begin, we need to get information from Wollo University's A+ learning management system. We then proceeded to use three steps for preparing the data. Data cleaning, classification, and reduction are all parts of data preprocessing, which prepares the dataset for data mining algorithm training. Feature

extraction was the tenth step in our process for extracting useful features. The next step in improving the algorithm was to employ hyperparameter tweaking. Automatically improving a model's hyperparameters is possible with hyperparameter tweaking. To configure the method to decrease the cost function of the learning rate for the gradient descent algorithm, one must set hyperparameters, which are all the model parameters that are not changed throughout the learning process.

Dataset. Academic institutions Wollo University and Kombucha Institute of Technology contributed to the dataset. The student information portal system exported the data of the students from the academic years 2017–2022. In its final form, the dataset had eight columns. Each student's ID, gender, area, entrance result, number of previous tries, studied credits, handicap, and final result are all included in their own column in the dataset.

Data Preprocessing. For the models to function to their full potential, the dataset was preprocessed before they were fitted. Due to the nonnumerical nature of our data, extensive preprocessing was necessary. There were three preprocessing steps employed in this research. Data cleaning was the first step in ensuring the dataset was free of errors caused by missing values and noise. After that, we dealt with numerical quantities by using data classification. Data standardisation was achieved via the use of label coding. The label encoder was designed to transform text values like "pass," "withdrawn," "pass," and "fail" into numerical values. Machine learning algorithms work better with numerical values rather than category ones. There are only two potential categories for categorical data, which often takes the form of strings or categories. The number of possible values is limited.

Feature Selection. Due to the presence of both numerical and categorical characteristics in our dataset, careful curation was required. To account for the differences between numerical and categorical data, we used separate approaches to each. To help with the analysis that followed, a short explanation was provided for each characteristic. We mainly used the random forest approach to find the most relevant features in this varied dataset.

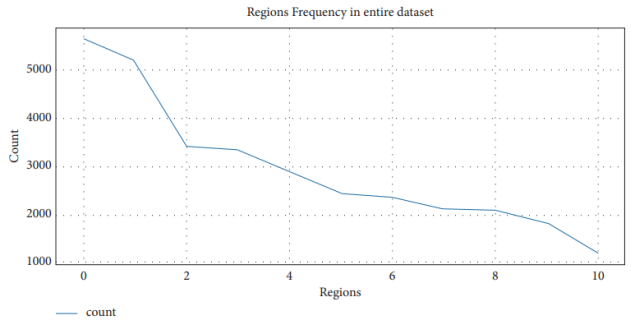

Figure 2: Region frequency in the dataset

In addition, using a small number of folds has its benefits as each fold may capture a significant portion of the data. The technique guarantees that every iteration captures a varied and representative sample, which adds to the overall dependability of the model's performance assessment, given the size of the dataset.

Hyperparameter Optimization. In order to improve the algorithm's performance, it is essential to tune its parameters before displaying the findings or preparing the system for production. It is also known as optimisation of hyperparameters. The goal of machine learning is to develop an automated computer system capable of building models from data without the need for tedious and time-consuming human intervention. One of the issues is setting the settings for the learning algorithms before employing the models.

Discovering the optimal hyperparameter values in machine learning is like to trying to locate a needle in a haystack. As part of our study, we use grid search to traverse this intricate search area. To achieve the best possible accuracy, we compare predicted values to actual values and adjust the model parameters accordingly. Hyperparameter tuning, on the other hand, has its own set of problems that may be solved via methods like dataset trimming.

Prediction Methods. This research compares and contrasts four different prediction/classification algorithms. These include support vector machines, decision trees, KNN, and Naive Bayes. The superior modelling capabilities of these algorithms make them ideal for use in classification-type prediction problems.

The output of a machine learning algorithm that has been trained on a training dataset and then applied to test data in order to forecast the value of a given result is called a prediction. Equation 1 determines the prediction's accuracy.

Accuracy = (Number of Correct Predictions) / (Total Number of Predictions)

Performance Measures. A confusion matrix was used to summarise the efficacy of a classification technique in this study. Caution should be used when interpreting classification accuracy in datasets with more than two classes or with unequal numbers of observations in each class. The strengths and weaknesses of the categorization model may be better understood by the computation of the confusion matrix. The three main metrics used to evaluate performance are recall, accuracy, and precision. Accuracy, measured as the percentage of expected positive observations that really materialise, is known as precision.

RESULTS AND DISCUSSION

Python on a Windows environment is used to prepare the test bed. For this investigation, we utilised the diabetic data set and documented the outcomes as follows. Data split ratios affect the GNB classifier's accuracy; table 1 shows the effects of several ratios. Here we look at three different ratios:

60:40, 70:30, and 80:20. The most effective setup, with the best accuracy for the GNB classifier at 79.22%, is the 80-20 data split ratio.

Table 1 Identifying Optimal Training and Testing Ratio in GNB Classifier

	Training/Test Ratio	Accuracy
GNB	60/40	75.0000
	70/30	76.1904
	80/20	79.2207

To demonstrate how data split ratios affect model accuracy, this context uses the GNB classifier as an example model. With a bigger training dataset made available to it by the 80/20 split ratio, the GNB classifier is able to provide better predictions. Choosing the right data split ratio is crucial for training machine learning models to make correct predictions, as this observation shows.

Data Preprocessing. Datasets including student information from 2017–2022, collected by Wollo University, Kombucha Institute of Technology, were used in this research. The data underwent pre-processing using Python software after missing data was removed. In order to guarantee that the predictive model was fed high-quality data, data preparation was necessary since the original dataset had many duplicate entries and missing values. As a result, problems like duplicate entries and missing values were eliminated by thorough preparation of the information. The data was preprocessed using Python tools to guarantee high-quality input for the predictive modelling activities that followed.

Algorithms Comparison. This is a comparative study of several machine learning methods, such as decision trees, support vector machine (SVM), Naïve Bayes, and K.

Table 2: Entrance qualification analysis

Feature entrance result	C0 (grade A) (%)	C1 (grade B) (%)	C2 (grade C) (%)
No formal quals	0.62	1.71	0
Lower than a level	19.75	37.36	58.36
A level or equivalent	57.72	51.28	32.22

to see how well they predicted student results, a study using closest neighbours (KNN) was carried out. Precision, recall, accuracy, and kappa statistics were some of the criteria used to evaluate each algorithm's performance.

A few examples of the many categorization and prediction tasks performed by decision trees (DT) include the forecasting of student performance, attrition rates, and final grade point averages. The simplicity, computational economy, and high accuracy of Naive Bayesian make it a popular classification method. The most accurate prediction of graduate students' GPAs is made using Naïve Bayes in educational contexts, which is based on student performance predictions from the previous semester. Another reliable method for predicting how well a pupil will do in school is Support Vector Machine. Researchers Ramesh et al. looked at how well Naïve Bayes performed. For the purpose of forecasting students' performance, we compared SMO with simple, multilayer perceptron, J48, and REP tree approaches; we found that multilayer perceptron was the best suitable algorithm, while SMO was competitive. We compare KNN, SVM, decision trees, and naïve Bayes classifiers in this work.

The results of the prediction models used in this study are shown in Table 3. After modifying the settings, Table 2 shows the results of the prediction algorithms. The results demonstrate that, prior to parameter change, SVM Linear produced the most accurate predictions (95.4%), decision tree came in second (90.9%), and Naive Bayes came in third (77.3%). The results of adjusting the parameters significantly improved the three prediction systems' accuracy. A rise from 95.4% to 96.0% in SVM Linear's prediction accuracy is seen. The accuracy of the decision tree was increased from 90.9% to 93.4%. The most notable improvement was shown in the Naive Bayes model, whose prediction accuracy was up from 77.3% to 83.3%.

This study's results shed light on the topic of student performance prediction in higher education. The research ensures the reliability of future studies by applying rigorous data preparation and feature selection procedures to construct a strong prediction model.

Machine learning algorithms, such as SVM, decision trees, Naïve Bayes, and K-nearest neighbours (KNN), were compared and found to be

Table 3: Prediction results of all methods before parameter tuning

Prediction method	Precision	Recall	Accuracy	Kappa statistic
SVM linear	0.9402	0.9789	0.9541	0.9305368
Naïve bayes	0.8186	0.8943	0.7738	0.6545808
Decision tree	0.8890	0.8784	0.9099	0.8632813
KNN	0.8441	0.8584	0.8538	0.8232813

how well they foretell the results for students. These results are in line with previous research, which shows that these classifiers are effective and flexible in educational contexts.

For this research, we utilised grid search, a technique for training and evaluating k-nearest neighbours (kNN) models with different values of k. We found that 8 was the greatest value of k after using 10-fold cross-validation, as it produced the best performance metrics.

CONCLUSION

Results from data mining may be influenced by the approach used. Our study made advantage of

repeated k-fold cross-validation to reduce sample-related bias. This is a contributing factor to the reliable results of predictions. After that, hyperparameter optimisation or parameter tweaking was used to further improve the prediction models' accuracy. Pre- and post-parameter change findings demonstrated improved accuracy. SVM's strength lies in its ability to maximise margins, handle high-dimensional data well, use efficient kernel functions with fewer hyperparameters, and be resilient to overfitting. However, out of all the classifiers tested, decision trees had the second-highest accuracy at 93.4%. Decision trees simplify the decision-making process by providing a clear and comprehensible model. The results of adjusting the parameters significantly improved the three prediction systems' accuracy. A rise from 95.4% to 96.0% in SVM Linear's prediction accuracy is seen. The accuracy of the decision tree was increased from 90.9% to 93.4%. The most notable improvement was shown in the Naive Bayes model, whose prediction accuracy was up from 77.3% to 83.3%. The 80-20 data split proved to be the most successful in improving predictive performance among the three examined ratios (60-40, 70-30, and 80-20), as it consistently produced the best accuracy across several models.

REFRENCE

1. Frank Hutter et.al "Algorithm runtime prediction: Methods & evaluation" Artificial Intelligence Volume 206, January 2014, Pages 79-111

2. Ashir Javeed, Sanam Shahla Rizvi, Shijie Zhou, Rabia Riaz, Shafqat Ullah Khan, Se Jin Kwon, "Heart Risk Failure Prediction Using a Novel Feature Selection Method for Feature Refinement and Neural Network for Classification", *Mobile Information Systems*, vol. 2020, Article ID 8843115, 11 pages, 2020. https://doi.org/10.1155/2020/8843115

3. Isabella M. Tromba "MakeML: Automated Machine Learning from Data to Predictions"2018

4. Subhash Chandra Gupta et.al "Enhancing The Performance Of Diabetes Prediction Using Tuning Of Hyperparameters Of Classifiers On Imbalanced Dataset" DOI : 10.21817/indjcse/2021/v12i6/211206049 Vol. 12 No. 6 Nov-Dec 2021

5. Jia_Wu et.al "Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization" Journal of Electronic Science and Technology Volume 17, Issue 1, March 2019, Pages 26-40

6. Ouyang, B., Song, Y., Li, Y., Sant, G. and Bauchy, M. (2021) Ebod: An Ensemble-Based Outlier Detection Algorithm for Noisy Datasets. Knowledge-Based Systems, 231, Article ID: 107400. https://doi.org/10.1016/j.knosys.2021.107400

7. Li, L. and Talwalkar, A. (2020) Random Search and Reproducibility for Neural Architecture Search. Proceedings of the 35th Uncertainty in Artificial Intelligence Conference, Vol. 115, 367-377.

8. Jian, S.-W., Cheng, H.-Y., Huang, X.-T. and Liu, D.-P. (2020) Contact Tracing with Digital Assistance in Taiwan's Covid-19 Outbreak Response. International Journal of Infectious Diseases, 101, 348-352. https://doi.org/10.1016/j.ijid.2020.09.1483

9. Mahesh, B. (2020) Machine Learning Algorithms—A Review. International Journal of Science and Research, 9, 381-386.

10. Jain, G., Mittal, D., Thakur, D. and Mittal, M.K. (2020) A Deep Learning Approach to Detect Covid-19 Coronavirus with X-Ray Images. Biocybernetics and Biomedical Engineering, 40, 1391-1405. https://doi.org/10.1016/j.bbe.2020.08.008

11. Aggarwal, C. C. (2014). *Data classification: Algorithms and applications*. CRC Press.

12. Aljawarneh, S. A. (2020). Reviewing and exploring innovative ubiquitous learning tools in higher education. *Journal of Computing in Higher Education, 32*(1), 57–73.

13. Baker, R. S. (2014). Educational data mining: An advance for intelligent systems in education. *IEEE Intelligent Systems, 29*(3), 78–82.

14. Belanche, L.A, & González, F.F. (2011). Review and evaluation of feature selection algorithms in synthetic problems. *arXiv preprint arXiv: 1101.2320*.

15. Bollier, D., & Firestone, C. M. (2010). *The promise and peril of big data* (pp. 1–66). Aspen Institute, Communications and Society Program.

Corresponding Author

Vinod K C*

Research Scholar, University of Technology, Jaipur, Rajasthan, India

Email: kckcvkc@gmail.com