INTERNATIONAL JOURNAL OF ACADEMIC RESEARCH IN BUSINESS & SOCIAL SCIENCES

Vol 11, Issue 12, (2021) E-ISSN: 2222-6990

Machine Learning Model Development for Screening Potential Entrepreneurs in the B40 (Bottom 40%) for Targeting Assistance

Sagaran Gopal[1], Sulochana Nair[2]

[1]Associate Professor and Program Leader, Binary University of Management and Entrepreneurship, Malaysia, [2]Vice Chancellor and Professor, Binary University of Management and Entrepreneurship, Malaysia

To Link this Article: http://dx.doi.org/10.6007/IJARBSS/v11-i12/11896 DOI:10.6007/IJARBSS/v11-i12/11896

Published Date: 10 December 2021

Abstract

The research aims to identify a suitable machine learning model between various machine learning (ML) systems analysed. The input data are factors identified through principal component analysis (PCA) of potential entrepreneurs in the B40 category by analysing 1000 responses from this group through a survey instrument. The following machine learning systems are tested; Random Forest, Extra Trees, K-Neighbots, SVC, Ridge Classifier, Logistic Regression and Decision Tree to select and identify a case-based reasoning artificial intelligence (AI) system best suited in this scenario. Given the data set size, results based on accuracy indicate the best algorithm is Logistic Regression.

Keywords: Machine Learning, Model Development, Potential Entrepreneurs, B40, Targeting Assistance

Introduction

Governments and private organizations are implementing artificial intelligence (AI) projects in a wide range of applications including autonomous systems to predictive analytics. These projects have a common denomination centred on a problem and with an understanding that data and machine learning algorithms can be applied to solve the problem, through machine learning model building.

The Malaysian government had also undertaken many projects, albeit, non-machine learning in the belief that entrepreneurs are able to develop the nation's agenda for a developed nation status by 2030. These projects involve both direct and the participation of various organisations. The Budget 2019 also gives great emphasis to this endeavour. However, many of these initiatives did not research the intended audience, namely the individuals in B40 who have the penchant, passion and right traits for entrepreneurship.

This study, therefore, will analyse the data collected from a survey of 1000 B40 (Bottom 40%) respondents as a starting point to identify important factors on entrepreneurial intentions of this community and use those data to analyse various ML models. Finally, it aims to identify the most appropriate predictive model that can be utilised to target assistance to members of the B40 community who have the potential of becoming future entrepreneurs.

Problem Statement

The Malaysian government through its agencies and public participation had undertaken numerous projects to develop the entrepreneurship skills among the B40. However, many of these projects were not successful resulting in failures. There is a likelihood that such projects are not robust enough to identify likely factors and tested thoroughly prior to implementation. One way out of such predicament is to analyse the data using AI. Machine learning models can mitigate lapses in the traditional approach. It can be used to overcome such deficiency through rigorous analyses of data to find the most appropriate model.
Hence, this research intends to build a model that can be used to screen potential entrepreneurs for effective targeted assistance.

Objectives of the Research

To analyse various ML predictive models and identify the most appropriate model for targeting assistance.

Literature Review

Machine Learning (ML) is a branch of Artificial Intelligence (AI) used to solve real life problems. Machine Learning allows the systems to make decisions independently without any external assistance. These decisions are made when the machine is able to learn from the data and understand the underlying patterns that are contained within it. Then, through pattern matching and further analysis, they return the outcome which can be a classification or a prediction. ML uses one or several combine mathematical models to learn pattern in data and hence make predictions (Sarker, 2021; Alajlan, 2012).

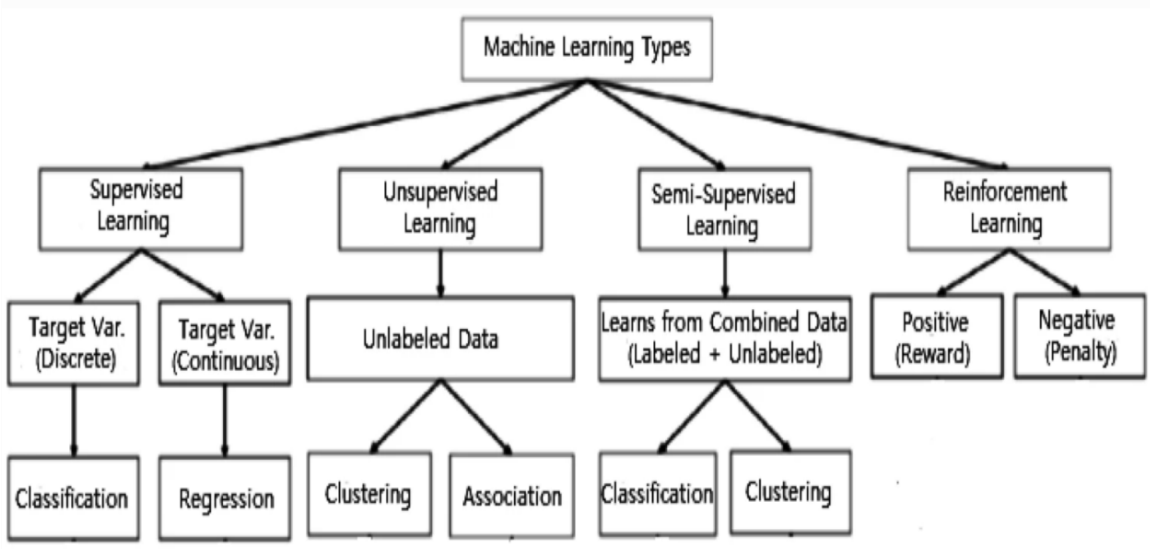


Fig 2.1 The general classifications of ML (Sarker, 2021).

Machine Learning (ML) problems fall into three categories, namely Supervised, Unsupervised problems and Reinforcement learning. Supervised learning can be broken down to Classification where samples (predictors) belong to two or more classes (target) that is to be predicted, or Regression where the training data corresponds to one or more continuous variable. On the other hand, Unsupervised learning is when the training set consists of input vectors (predictors) without any corresponding target label. The algorithm tries to derive knowledge from a general input without the help of a set of pre-classified examples that are used to build descriptive models. The reinforcement learning algorithm is able to learn depending on the changes that occur in the environment in which it is performed. In fact, since every action has some effect on the environment concerned, the algorithm is driven by the same feedback environment (Fumo, 2017).

Supervised learning is the most common scheme found in solving a wide range of learning problems (Opitz & Maclin, 1999). Given an unforeseen input instance, supervised learning model should predict the class of the instance based on what it has learned from the training set. However, the prediction accuracy of the supervised learning model depends highly on several factors such as algorithm deployed (Bowles, 2015), nature of input data by which the model is trained and the method of pre-processing, and algorithm parameter running.

All algorithms used in supervised learning perform the same task of capturing the pattern from the training set, then applies them to a test set to measure some type of accuracy metric for either classification or prediction (Kotsiantis, 2007). Classification here refers to a discrete number of classes for several data points and prediction is referring to a method of regression. Given that there are several mathematical algorithms by which a predictive model can be trained, a method of taking advantage of the collective accuracies of each predictive model used to solve a learning problem can be employed. This method has gained attention since the 90's and currently it is more often than not most production machine learning models are nothing but a stacked collection of individual prediction model. This method is referred to as ensemble learner (Xu & Yang, 2015). Some of the algorithms that come under supervised learning are discussed next.

Linear Regression is a supervised machine learning algorithm that is used whenever there is a need to predict a variable based on another variable. The relationship between two or more variables is then used to perform predictions that follow a linear pattern. Random Forests are an ensemble learning method that is for performing classification, regression as well as other tasks through the construction of decision trees and providing the output as a class which is the mode or mean of the underlying individual trees. Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model within the sort of an ensemble of weak prediction models, typically decision trees. It is an ensemble learning method that is a collection of several weak decision trees which results in a powerful classifier. Support Vector Machine (SVM) are powerful classifiers that are used for classifying the binary dataset into two classes with the help of hyperplanes. Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary variable. Artificial Neural Networks are modelled after the human brain and they learn from the data over time. They form a much larger portion of machine learning called Deep Learning (Fumo, 2017).

From a practical perspective, the success or acceptance of any machine learner outcome cannot be measured by its accuracy on test sets and validation sets alone. From experience based on building 10's of learners for different problems ranging from supervised to reinforcement problems, it is found that there are several factors contributing to the acceptance of the final learner - some of these factors had been researched and published such as data heterogeneity, multicollinearity, and data redundancy. The first and most important step in building a good learner is to profile the data set at hand and understand the specific data point types in each of the predictors. For an example, Linear Regression, and Support Vector Machine (SVM) are among the very popular algorithms used in the ML - these algorithms require the data of the predictors to be numerical in nature and scaled to the appropriate range. Although it is difficult to assess first hand, which algorithm will perform best during the data pre-processing step, it is recommended to try different pre-processing methods before feeding the data set to any ML learner.

Another factor that contributes to the success and acceptance of a learner is, its parameters and the final values of these parameters. Every learning has certain parameters required to perform its learning process which are referred to as hyper-parameters. The problem of hyper-parameters is that they are configuration variables external to the model in which its values cannot be estimated from the data set. An example is coefficients in a linear regression or logistic regression.

The solution to hyper-parameter estimation is usually referred to as Grid Search among the ML community. Grid Search involves an approach to hyper-parameter tuning by building and evaluating a model for each combination of the learner's parameters specific in a grid. The good news is the availability of well-developed libraries that can perform this task with great ease. The down side however is such process is time consuming and based on the problem in hand can be computationally very expensive.

Methodology

The current data set, and after performing a Factor Analysis and Principal Component Analysis (PCA), a class of unsupervised learning paradigm, for dimensionality reduction, falls into Supervised Learning category where dimensions are reduced to 12 with one more factor dropped from the total dimensions. Target labels are divided into four classes after appropriate clustering was performed:

1. poor,
2. below average,
3. good,
4. excellent.

The final 11 predictors used to train the ML model as below:

1. Communication ability
2. Training and Development
3. Customs and Habits
4. Self-Initiative
5. Coaching and Guidance
6. View on Enterprise
7. Past Experience
8. Relevant Organization Support

9. Attitude and Outlook
10. Ability, Skills and Knowledge
11. Inclination and perspective

ML Model Development and Components

Figure 3.1 is a flowchart outlining the development process of the model. Many of the individual steps are internally following software engineering iterative process where each step is being revisited for evaluation till satisfactory accuracy of the model is reached.

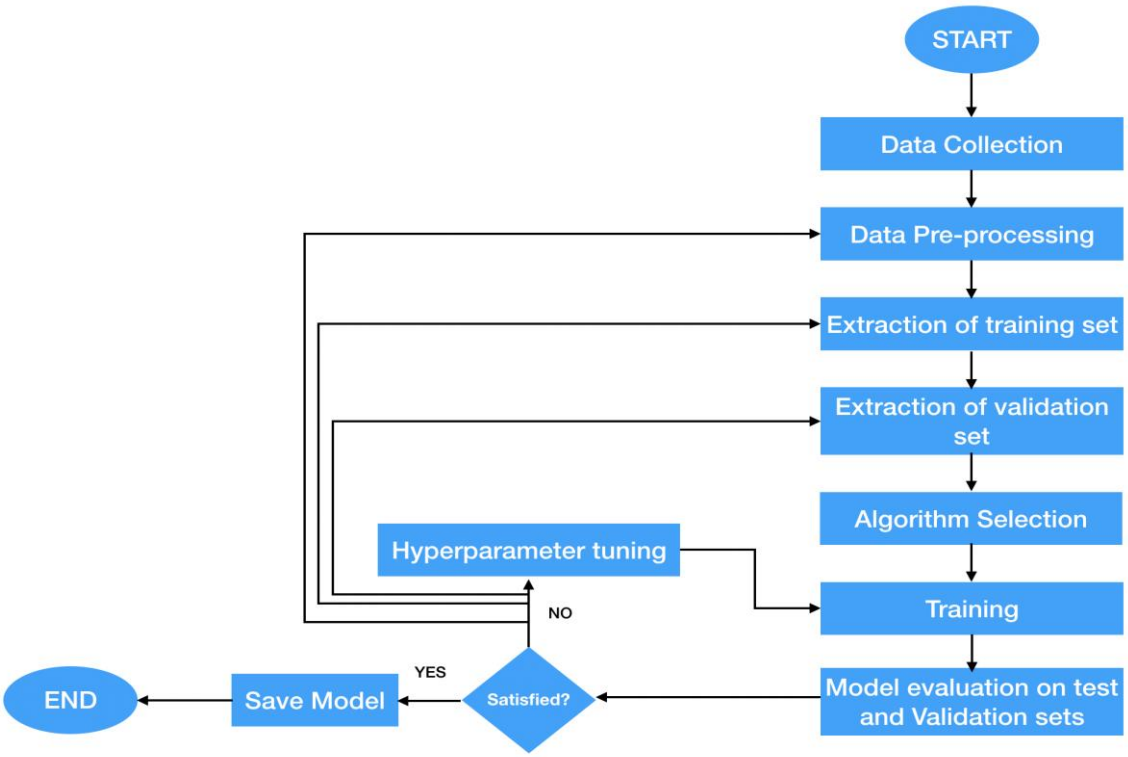


Fig 3.1 Model Development Flowchart

As any ML model training and evaluations, several components with different functionalities must be employed together with each component playing a vital role in the production of the final model. Table 3.1 is a non-exhaustive list of the different components used during the course of the development.

Table 3.1:
Components used to build ML Model

Component	Function
Jupiter Notebook	Semi I integrated Development Environment for Python and Scala
Python 3	Programming language used as a host for other ML libraries
scikit ML Package	A well-developed ML library widely used in research and development projects as well as production
Numpy	A package for scientific computing in Python
Seaborn	Statistical visualisation library for Python
Matplotlib	Python plotting libra such as histogram, scatter charts … etc
Yellowbrick	A library for classification visualisation among other functions

Several machine learning algorithms are evaluated during the course of the development. Finally, a combined model prediction is combined into an ensemble for a boot in accuracy. This step is very important and used mostly in the production ML model as well. Different models will perform differently based on the data at hand. Hence, combining these models will ensure a wider spectrum in which the model can operate with high and consistent accuracy possible.

The most popular methods used for combining predictions from different models are:
1. **Bagging**: Multiple model predictions typical of the same family but using different sub-sample of the training data set.
2. **Boosting**: Multiple model predictions of the same family of algorithms with each algorithm attempting to fix prediction error from prior model in the chain of models
3. **Voting**: the simplest form of ensemble. Its advantage is that it adapts well to models from the different algorithm's family. It calculates the final predicted by computing the mean of all predictions of the models in the chain.

The chain of algorithms used as the base of the research are listed in table 3.2.

Table 3.2:
Individual Algorithms with Final Hyper-Parameter Values

Algorithm	Final Hyperparameters
RandomForestClassifier	n_estimators=10, random_state=0
ExtraTreesClassifier	n_estimators=5, random_state=0
KNeighborsClassifier	n_neighbors=4
SVC	C=10000.0, kernel='rbf', random_state=0

RidgeClassifier	alpha=0.1, random_state=0
LogisticRegression	C=20000, penalty='l2', random_state=0
DecisionTreeClassifier	criterion='gini', random_state=0
AdaBoostClassifier	n_estimators=5,learning_rate=0.001

The training and test sets are usually split by 80% for training and 20% for testing. However, several training runs showed a training set of 66% and test set of 33% is the best fit. Cross validation was also employed with cross fold by 10 being the best.

Classification Report

Classification reports are indicative measure of the quality of learner from a classification algorithm. It is based on the number True and False prediction or more specifically True Positive, False Positive, True Negatives and False Negatives. Table 4.1 lists the different metrics found in a typical classification report.

Table 4.1:
Classification Report Metrics

Classification Report Metric	**Definition**	**Equation**
Precision	Accuracy of positive predictions	* TP/(TP + FP)
Recall	Fraction of positives that were correctly identified.	* TP/(TP+FN)
F1 score	percent of positive predictions	2*(Recall * Precision) / (Recall + Precision)

* TP: True Positive; FP: False Positive; FN: False Negative.

Classification reports are vital in understanding the performance of individual algorithms in classifying data points in a given test or validation sets. It provides a hint of the error made by the learner and more importantly the types of errors being made. Following is the classification report for each classifier.

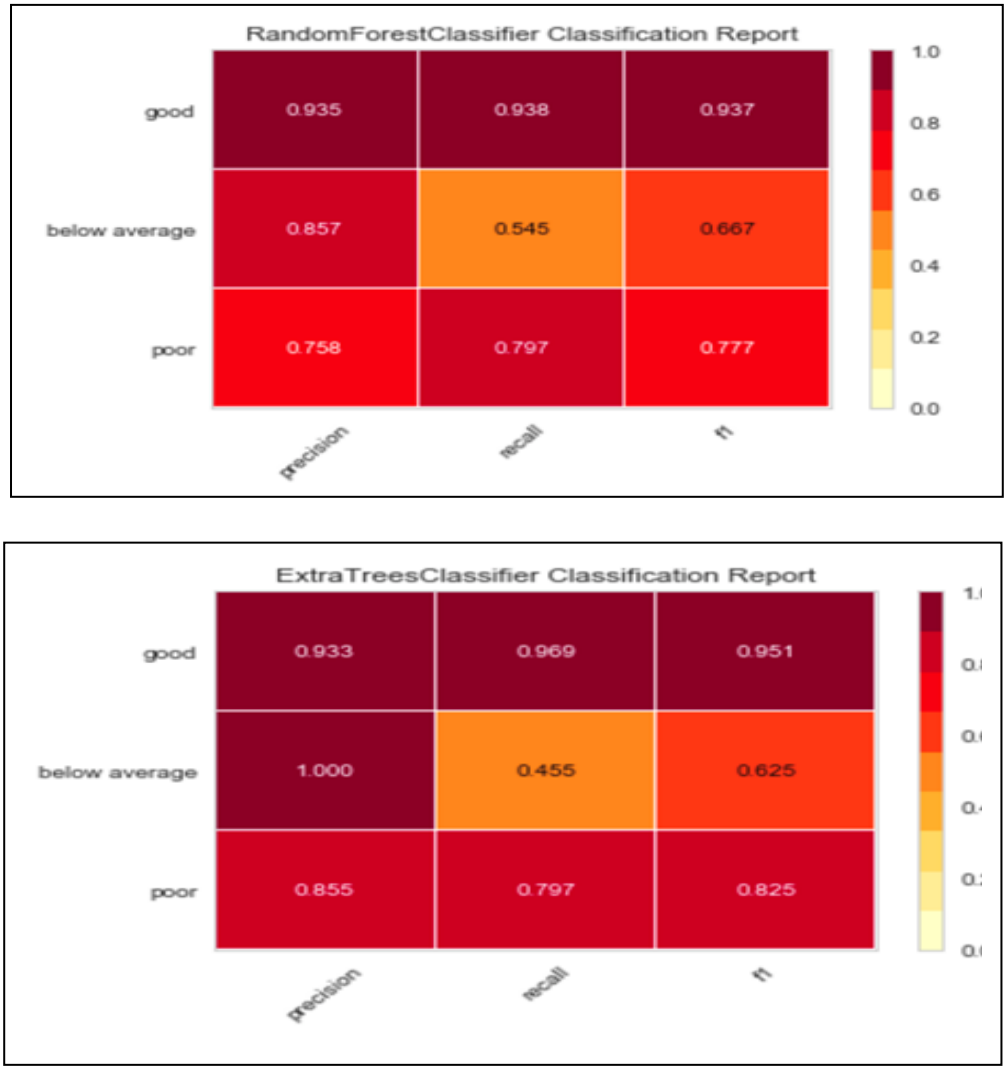


Figure 4.1: Random Forest

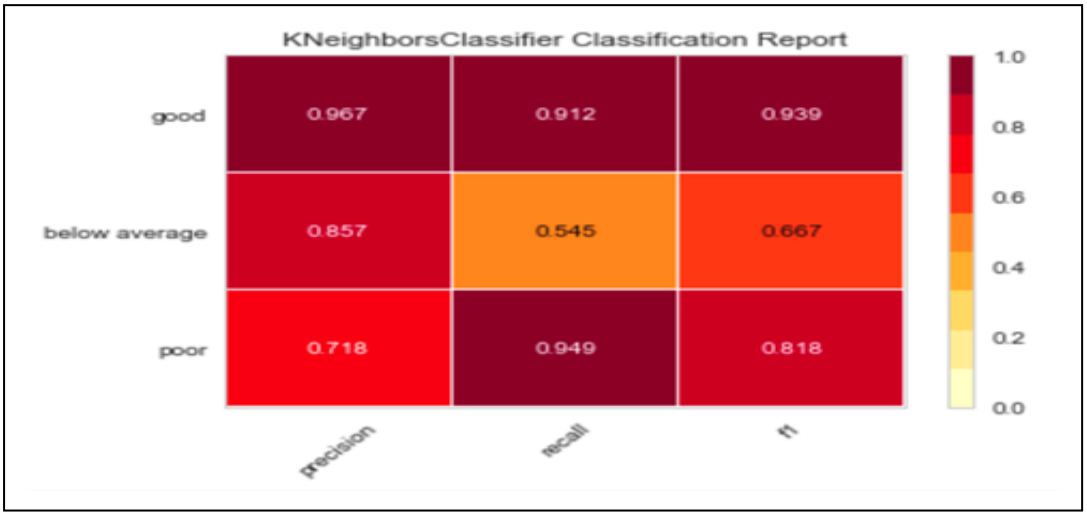


Figure 4.3: K-Neighbots

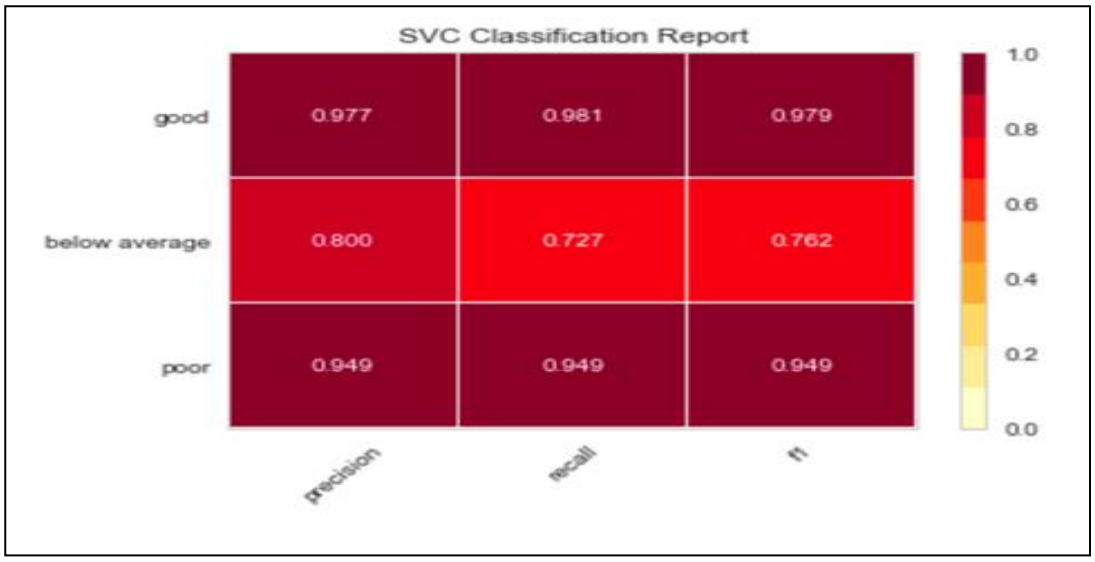
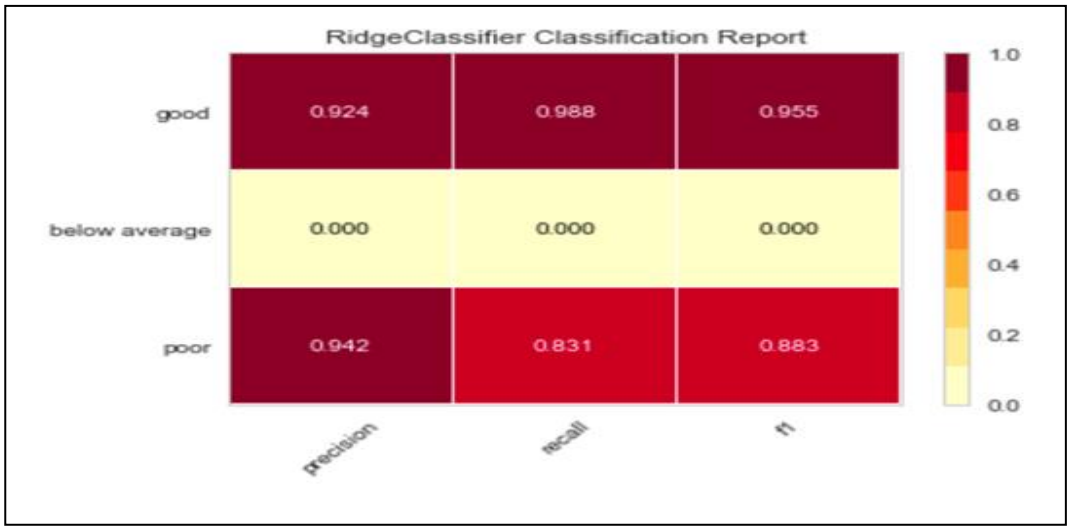


Figure 4.4: SVC



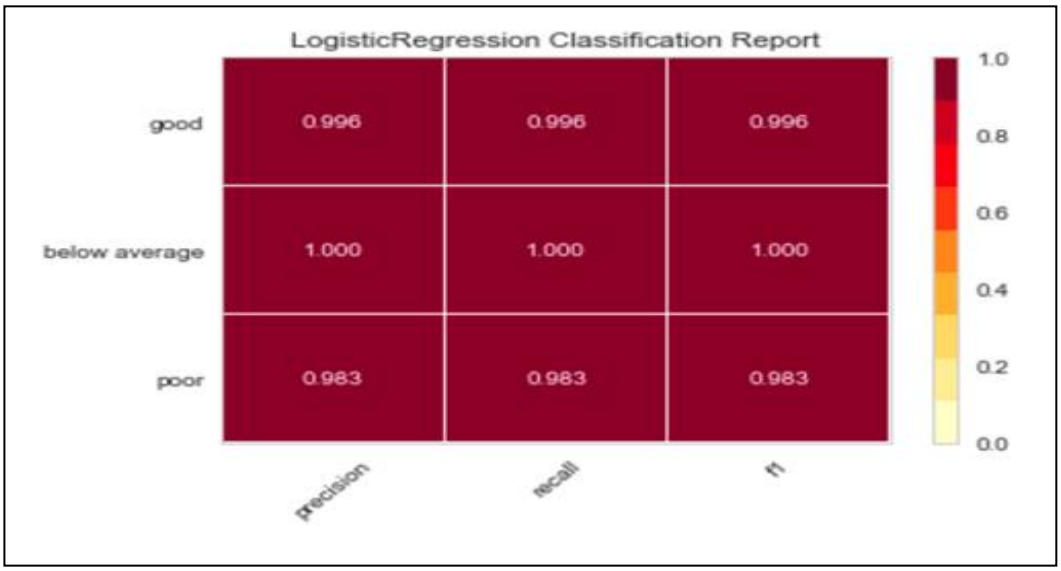


Figure 4.6: Logistic Regression

Figure 4.7: Decision Tree

Conclusion

Accuracy scores shown in Table 5.1 with Cross Validation of 10 being the best value among other values tested from 3 to 10.

Table 5.1:
Individual Algorithms Accuracy

Accuracy	Algorithm
0.9070 (+/- 0.0387)	RandomForestClassifier
0.8930 (+/- 0.0580)	ExtraTreesClassifier
0.9180 (+/- 0.0421)	KNeighborsClassifier
0.9640 (+/- 0.0174)	SVC
0.9150 (+/- 0.0383)	RidgeClassifier
0.9870 (+/- 0.0142)	LogisticRegression
0.8290 (+/- 0.0550)	DecisionTreeClassifier
0.8310 (+/- 0.0558)	AdaBoostClassifier

Given the dataset size, results based on accuracy indicate the best algorithm is Logistic Regression. Logistic Regression in this instance is producing the best accuracy after data pre-processing and Principal Component Analysis for dimensionality reduction. The matter to note here is that Logistic Regression is commonly the algorithm of choice for binary classification where the target label is either 0 or 1. However, the dataset used for training and validation is a multi-class problem and not a regression problem where Logistic Regression being a probabilistic classification model exhibited a very excellent fit.

Further data collection which increases the total data points may require additional work and more hyper-parameter tuning. Hence, it is recommended to gather more data points in future work and experiment with employing Neural Networks for a more versatile model.

References

Alajlan, N., Bazi, Y., Melgani, F., Yager, R. R. (2012). Fusion of supervised and unsupervised learning for improved classification of hyperspectral images (2012) *Information Sciences*, 217, 39-55.

Bowles, M. (2015). *Machine Learning in Python: Essential Techniques for Predictive Analytics*, John Wiley & Sons Inc.

Fumo, D. (2017). Types of Machine Learning Algorithms You Should Know. Retrieved from: https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861. Assessed on 11th Nov 2020

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques, *Informatica* 31, 249-268

Opitz, D. R., & Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study, *Journal of Artificial Intelligence Research,* 11, 169-198.

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. SN COMPUT. SCI. 2, 160. Retrieved from: https://doi.org/10.1007/s42979-021-00592-x.

Xu, X., & Yang, G. (2013). Robust manifold classification based on semi supervised learning. *International Journal of Advancements in Computing Technology*, 5 (8), 174-183.