

Prediction Using SVM and DT



Dr T K Ganga¹, Dr S Selvakani², Mrs K Vasumathi³

¹Associate Professor Muthurangam Government Arts College (A), Vellore

^{1,2}Assistant Professor, Department of Computer Science, Government Arts and Science College, Arakkonam.

ABSTRACT: Diabetes is recognized as a severe and persistent illness that leads to an elevation in blood sugar levels. Untreated and undetected diabetes can give rise to numerous complications. The laborious process of identification typically involves a patient visiting a diagnostic centre and consulting with a doctor. However, the advancement of machine learning approaches has addressed this critical issue. The objective of this study is to develop a model capable of predicting the likelihood of diabetes in patients with the utmost accuracy.

To achieve this goal, three machine learning classification algorithms—Decision Tree, SVM, and Naive Bayes—are employed in this experiment to identify diabetes at an early stage. The experiments are conducted on the Pima Indians Diabetes Database (PIDD), sourced from the UCI machine learning repository. The performance of each algorithm is assessed using various metrics such as Precision, Accuracy, F-Measure, and Recall. Accuracy is gauged based on both correctly and incorrectly classified instances. The results indicate that Naive Bayes outperforms the other algorithms, achieving the highest accuracy of 76.30%. These findings are corroborated through a meticulous examination of Receiver Operating Characteristic (ROC) curves.

KEYWORDS: Decision Tree Algorithm, Machine Learning, Prediction, Diabetes, Pima Indians Diabetes Database

I. INTRODUCTION

Classification methodologies are widely employed within the medical domain to categorize data into distinct classes based on predefined criteria, particularly within the context of an individualized classifier. Diabetes, a malady characterized by the compromised ability of the body to produce the hormone insulin, results in an aberration of carbohydrate metabolism and an elevation in blood glucose levels. Individuals afflicted with diabetes typically experience heightened blood sugar levels, manifesting as increased thirst, augmented hunger, and frequent urination. Failure to address diabetes can lead to a multitude of complications, among which are severe conditions such as diabetic ketoacidosis and nonketotic hyperosmolar coma [1].

Diabetes is regarded as a significant and critical health concern characterized by an inability to regulate sugar levels. This condition is influenced by various factors, including height, weight, genetic predisposition, and insulin; however, sugar concentration is deemed the primary contributing factor. Early detection stands as the sole recourse to avert ensuing complications [2].

Numerous researchers are engaged in empirical investigations aimed at disease diagnosis, employing diverse classification algorithms rooted in machine learning methodologies such as J48, SVM, Naive Bayes, Decision Tree, and Decision Table. Empirical evidence substantiates the superior efficacy of machine-learning algorithms in disease diagnosis [1], [2], [6]. The amalgamation of Data Mining [5], [2], and Machine learning algorithms derives strength from their adeptness in handling extensive datasets, amalgamating information from diverse sources, and assimilating background knowledge into the study [8].

This research specifically delves into the realm of gestational diabetes among pregnant women. Within this investigation, the Naive Bayes, SVM, and Decision Tree machine learning classification algorithms are employed and scrutinized using the PIDD dataset to ascertain the predictive capability for diabetes in patients. The experimental performance of these three algorithms is meticulously compared across various metrics, resulting in commendable accuracy [7].

II. RELATED WORK

Sajida et al. expound upon the efficacy of Adaboost and Bagging ensemble machine learning methodologies [18], utilizing the J48 decision tree as the foundational framework for classifying Diabetes Mellitus and distinguishing patients as either diabetic or non-

Prediction Using SVM and DT

diabetic based on diabetes risk factors [20]. The experimental outcomes substantiate that the Adaboost machine learning ensemble technique outperforms both bagging and the J48 decision tree.

In a parallel vein, Orabi et al. [16] devised a system dedicated to diabetes prediction, with a primary focus on forecasting diabetes occurrences in individuals at specific ages. This innovative system, grounded in machine learning principles and driven by decision tree algorithms, demonstrates satisfactory results. The efficacy of the designed system in predicting diabetes incidents at precise ages is notably enhanced, achieving heightened accuracy through the utilization of Decision tree[12], [7].

Pradhan et al. [17], in their research endeavor, harnessed Genetic Programming (GP) for the training and testing of a database aimed at predicting diabetes. Utilizing a Diabetes dataset sourced from the UCI repository, the results obtained through Genetic Programming [5], [12] showcased optimal accuracy in comparison to other implemented techniques. The expeditious generation of classifiers also contributes to a notable improvement in accuracy, rendering this approach particularly useful for diabetes prediction at a reduced cost.

Rashid et al. [19] conceived a predictive model with dual sub-modules to forecast diabetes as a chronic disease. The initial module employs Artificial Neural Network (ANN), while the second module incorporates Fasting Blood Sugar (FBS). The discernment of diabetes symptoms in patient health is facilitated through the use of Decision Tree (DT)[10].

Nongyao et al. [12] executed an algorithm to assess the risk of diabetes mellitus, employing four distinguished machine learning classification methods: Decision Tree, Artificial Neural Networks, Logistic Regression, and Naive Bayes. To fortify the model's robustness, Bagging and Boosting techniques are implemented. Experimental results underscore that the Random Forest algorithm yields optimal outcomes among all the algorithms applied.

III. DATA SET

The Pima Indian Diabetes Database stands as a renowned dataset widely employed in the intersection of machine learning and healthcare research. It serves as a valuable repository of information for the exploration and prediction of diabetes mellitus within the Pima Indian population, an indigenous community situated in Arizona, United States. This dataset has played a pivotal role in the development and assessment of various predictive models and algorithms geared towards identifying individuals at susceptibility to diabetes.

Originally curated by the National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK) during the 1980s, the Pima Indian Diabetes Database encompasses several variables. These include clinical, demographic, and diagnostic measurements obtained from 768 female Pima Indian individuals aged 21 and above. The variables encapsulate diverse parameters such as the number of pregnancies, glucose concentration, blood pressure, skinfold thickness, insulin levels, body mass index (BMI), and the diabetes pedigree function, among others. The dataset also incorporates a binary target variable signifying the presence or absence of diabetes within a five-year timeframe subsequent to the recorded measurements.

A salient advantage of the Pima Indian Diabetes Database lies in its real-world relevance. The dataset authentically mirrors the attributes of a specific population, offering nuanced insights into the prevalence and risk factors associated with diabetes within the Pima Indian community. This real-world applicability renders it an invaluable resource for the formulation and evaluation of predictive models in the realm of diabetes diagnosis and prevention.

Number of Instances: 768

Number of Attributes: 8 plus class

For Each Attribute: (all numeric-valued)

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)²)
7. Diabetes pedigree function
8. Age (years)
9. Class variable (0 or 1)

Prediction Using SVM and DT

Table I. Diabetes Prediction

Preg	Gluc o	BP	ST	Insu	BMI	Diabet	Ped Func	Age	Outcom e
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Data Pre-processing - Among the crucial phases of data analysis, data preprocessing assumes paramount importance. Particularly, healthcare-related datasets often harbour missing values and other impurities that can compromise data efficacy. To enhance the quality and effectiveness derived from the mining process, meticulous data preprocessing is imperative. For the effective application of Machine Learning Techniques to the dataset, this procedural step becomes indispensable, ensuring accurate results and the success of predictive endeavors. In the context of the Pima Indian diabetes dataset, a twofold preprocessing approach is deemed necessary.

Missing Values Removal - The elimination of instances featuring zero (0) as their attribute value constitutes the initial step. The presence of zero as an attribute value is deemed implausible and, therefore, warrants elimination. This process extends beyond mere elimination, transforming the dataset into a more relevant and condensed feature subset through feature subset selection. Such an undertaking serves to reduce the dimensionality of the data, thereby facilitating expedited computational processes.

Splitting of Data - Following the cleansing of the data, it undergoes normalization to facilitate the training and testing of the model. The data is segregated into training and testing sets, allowing for the algorithm to be trained on the former while keeping the latter set aside for subsequent testing. The training process engenders the creation of a model based on logical algorithms and the values of features within the training dataset. The overarching objective of normalization is to standardize all attributes to a uniform scale.

Apply Machine Learning - With the prepared dataset, the application of Machine Learning Techniques ensues. Various classification and ensemble techniques are employed to predict diabetes within the Pima Indian dataset. The primary aim is to scrutinize the performance of these techniques, determine their accuracy, and discern the salient features pivotal in the prediction process.

IV. METHODOLOGY USED

A. Support Vector Machine

The Support Vector Machine (SVM) stands as an established paradigm within the realm of supervised machine learning models, specifically utilized for classification purposes. When presented with a two-class training sample, the primary objective of a support vector machine is to ascertain the optimal highest-margin separating hyperplane between the two classes [16]. To enhance generalization, it is imperative that the hyperplane is not in close proximity to data points belonging to the alternative class. The selection of the hyperplane necessitates its strategic placement at a considerable distance from data points in both categories. Those data points residing in close proximity to the margin of the classifier are identified as support vectors [12].

Recognized as a preeminent supervised machine learning algorithm, SVM is a favoured classification technique. Operating by creating a hyperplane to segregate two classes, SVM extends its utility to high-dimensional spaces by establishing a set of hyperplanes. This hyperplane can be applied for both classification and regression purposes, effectively discerning instances within specific classes and even classifying entities lacking direct support in the data. The separation is meticulously executed through a hyperplane, ensuring the demarcation to the closest training point of any class.

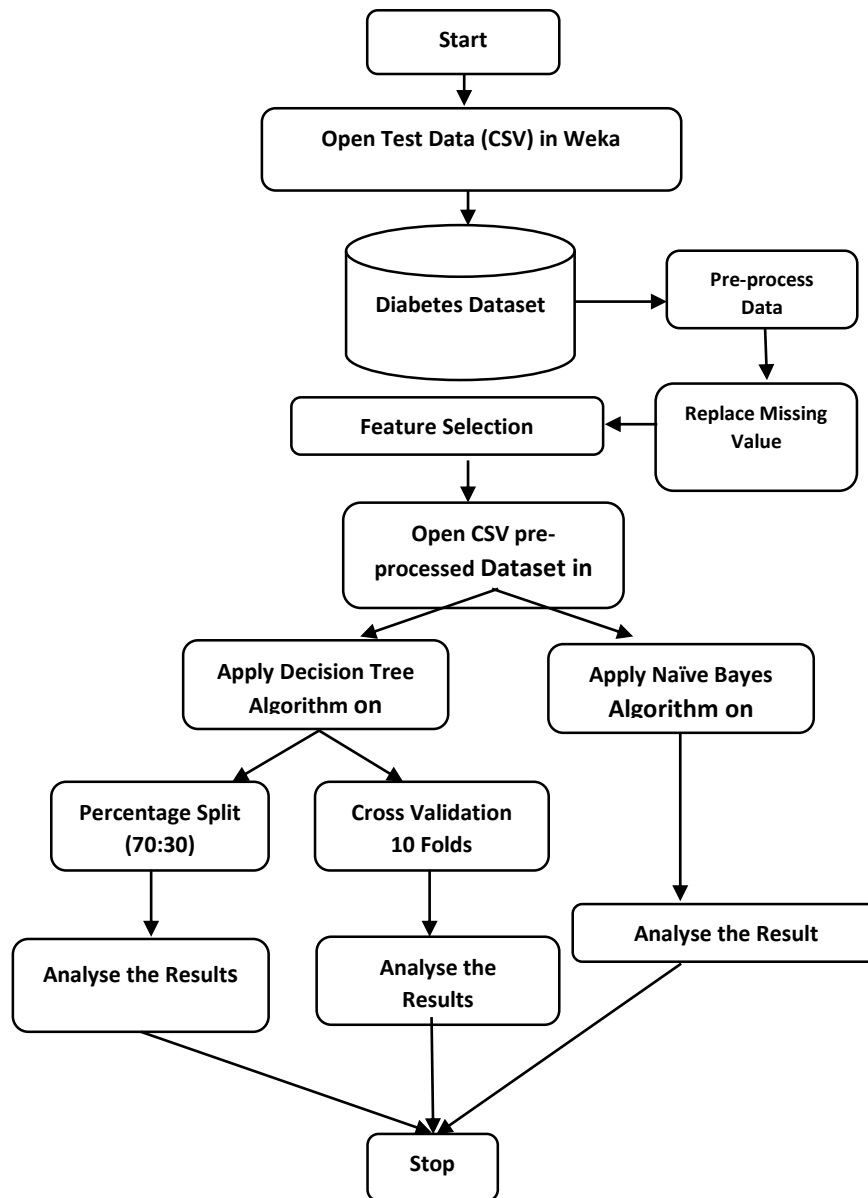


Figure 1. Prediction Flow Model

B. Naïve Bayes Classifier

Naive Bayes represents a classification methodology predicated on the premise that all features are mutually independent and unrelated. This paradigm asserts that the state or condition of a specific feature within a class holds no influence over the state of another feature. Leveraging conditional probability as its foundational principle, Naive Bayes is esteemed as a potent algorithm employed for classification endeavours. Notably, its efficacy is pronounced when dealing with datasets fraught with imbalanced distributions and instances of missing values.

C. Decision Tree Classifier

The Decision Tree stands as a supervised machine learning algorithm strategically applied to address classification challenges. Within the ambit of this research, the primary aim is to predict the target class by formulating decision rules derived from antecedent data. Decision Trees employ nodes and internodes to facilitate prediction and classification. The root nodes, equipped to handle instances with diverse features, can bifurcate into two or more branches, with the ultimate classification manifested in the leaf nodes. At each stage of decision-making, the Decision Tree selects nodes by evaluating the highest information gain among all the attributes [11].

D. Model Evaluation Error Metrics

The selection of an appropriate evaluation matrix for classification models is crucial in the construction of a robust model. In this study, the primary criterion for gauging the performance of each model is the accuracy of test predictions. However, it is imperative to note that other metrics, including the confusion matrix and Kappa, will also be taken into account. Table 3: shows the Confusion matrix

Prediction Using SVM and DT

Table II. The Confusion Matrix.

	Ref:0	Ref:1
Pred: 0	True Negative	False Negative
Pred:1	False Positive	True Positive

The confusion matrix is delineated into four distinct sectors, each encapsulating distinct classification outcomes. True positive denotes the count of accurately predicted diabetes cases as diabetes. Conversely, false positive represents the tally of healthy cases inaccurately identified as diabetes. True negative signifies the instances where healthy cases are correctly identified as such. Finally, false negative characterizes the count of diabetes cases erroneously predicted as healthy.

- Accuracy: This is used to determine the amount of a particular class that is correctly predicted over the total number of sample. It can be calculated as $\frac{TP+TN}{TP+TN+FP+FN}$
- Sensitivity: The ratio of the number of correctly predicted diabetes cases over the total number of the diabetes cases. It can be calculated as $\frac{TP}{TP+FN}$
- Specificity: The ratio of the number of correctly predicted healthy cases over the total number of healthy cases. It can be calculated as $\frac{TN}{FP+TN}$
- Positive predictive value: The ratio of the number of correctly predicted diabetes cases over the total number of cases predicted as diabetes. It can be calculated as $\frac{TP}{TP+FP}$
- Negative predictive value: The ratio of the number of correctly predicted as healthy cases over the total number of the cases predicted as healthy. It can be calculated as $\frac{TN}{TN+FN}$
- Prevalence: The total ratio of the number of diabetes cases. $\frac{FN+TP}{TP+TN+FP+FN}$
- Detection Rate: The ratio of the number of correctly predicted as diabetes over the total number of cases. $\frac{TP}{TP+TN+FP+FN}$
- Detection Prevalence: The ratio of the number of predicted as diabetes over the total number of cases. $\frac{FP+TP}{TP+TN+FP+FN}$
- Balanced Accuracy: The average of the proportion corrects of each class individually. $\left(\frac{TP}{TP+FN} + \frac{TN}{FP+FN}\right)/2$.

V. ALGORITHM

1. Employ recursive binary splitting to cultivate an expansive tree on the training dataset. Elect one of the inputs, denoted as X_j (where $j \in 1, \dots, p$), along with a designated cut-point s . This selection partitions the input space into two distinct half-spaces, namely $\{X : X_j < s\}$ and $\{X : X_j > s\}$. Iteratively execute the splitting process for each delineated region until a predetermined stopping criterion is met.
2. Implement cost complexity pruning on the enlarged tree to derive a series of optimal subtrees, contingent upon the parameter α .
3. Utilize K-fold cross-validation to ascertain the optimal value for α . This involves segmenting the training observations into K folds. For each $k = 1, \dots, K$:
 - (a) Replicate Steps 1 and 2 on all folds except the kth fold of the training data.
 - (b) Assess the error on the data within the omitted kth fold, while considering α as a variable. The results are averaged for each value of α , and the selection of α is made to minimize the average error.
4. Conclude the process by returning the subtree derived from Step 2 that aligns with the chosen value of α .

VI. EXPERIMENTS AND RESULTS

A. Checking Correlation

Upon the meticulous completion of data cleansing and pre-processing procedures, the dataset attains a state of preparedness suitable for both training and testing phases. Employing a dual strategy, we conducted K-fold cross-validation and employed an 85% train/test splitting methodology to assess the efficacy of various machine learning models.

Prediction Using SVM and DT

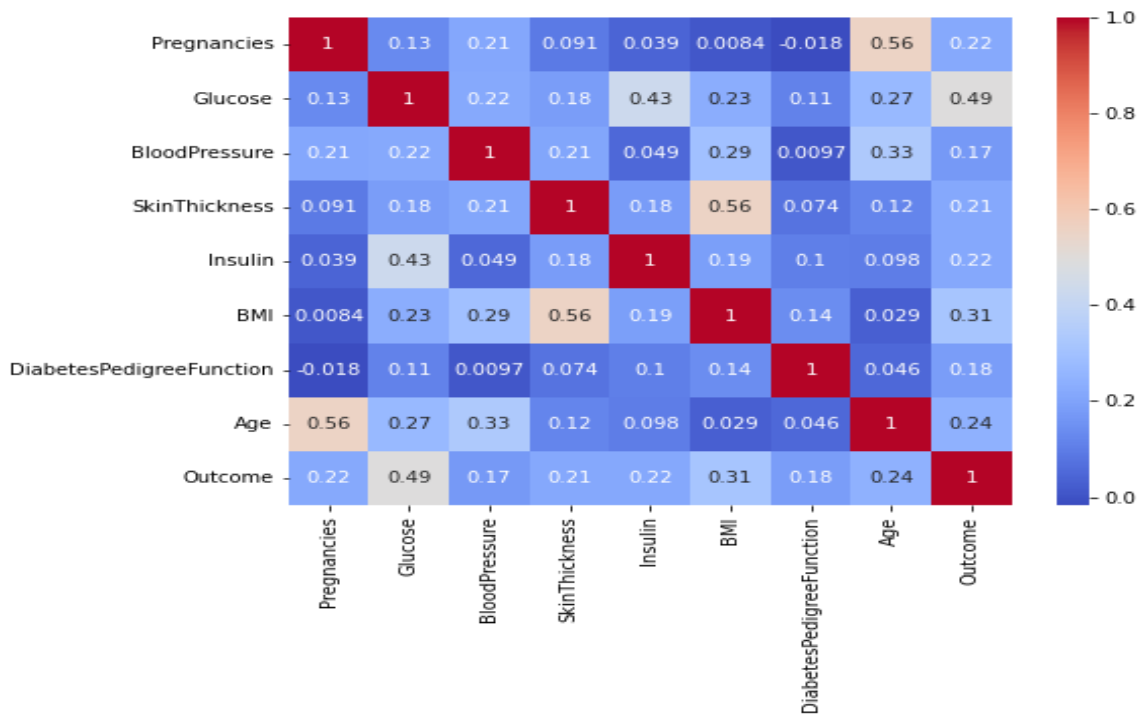


Figure 2. Correlation between Features

In the train/split paradigm, the dataset is bifurcated in a randomized fashion into training and testing subsets. Contrarily, in the K-fold cross-validation paradigm, the data is partitioned into K folds, with one fold exclusively allocated for validation/testing purposes, while the remaining K-1 folds are dedicated to training. This iterative process persists until each of the K folds has served as a distinct test set. The evaluation of performance is orchestrated through the derivation of the average from the compilation of all recorded scores pertaining to the Kth test.

B. Sample Code

```

from sklearn.tree import DecisionTreeClassifier # Import Decision Tree Classifier
from sklearn.model_selection import train_test_split # Import train_test_split function
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=1)
# Create Decision Tree classifier object
model = DecisionTreeClassifier()
# Train Decision Tree Classifier
model = model.fit(x_train,y_train)
#Predict the response for test dataset
y_pred = model.predict(x_test)
#Evaluation using Accuracy score
from sklearn import metrics #Import scikit-learn metrics module for accuracy calculation
print("Accuracy:",metrics.accuracy_score(y_test, y_pred)*100)
Accuracy: 67.53246753246754
#Evaluation using Confusion matrix
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test,y_pred)
array([[76, 23],
       [27, 28]])
print("Accuracy:",((82+27)/154))
Accuracy: 0.7077922077922078
#Evaluation using Classification report
from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))

```

Prediction Using SVM and DT

Table III. Classification Report

	Precision	Recall	f1-score	Support
0	0.74	0.77	0.75	99
1	0.55	0.51	0.53	55
Accuracy			0.68	154
macro avg	0.64	0.64	0.64	154
weighted avg	0.67	0.68	0.67	154
	Pred 0	Pred 1		
Act 0	332	39		
Act 1	84	121		

```

#Import modules for Visualizing Decision trees
from sklearn.tree import export_graphviz
from sklearn.externals.six import StringIO
from IPython.display import Image
import pydotplus
features=x.columns
features
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
      'BMI', 'DiabetesPedigreeFunction', 'Age'],
      dtype='object')
dot_data = StringIO()
export_graphviz(model,out_file=dot_data,filled=True,rounded=True,special_characters=True,feature_names
features,class_names=['0','1'])
graph = pydotplus.graph_from_dot_data(dot_data.getvalue())
graph.write_png('diabetes_set.png')
Image(graph.create_png())
    
```

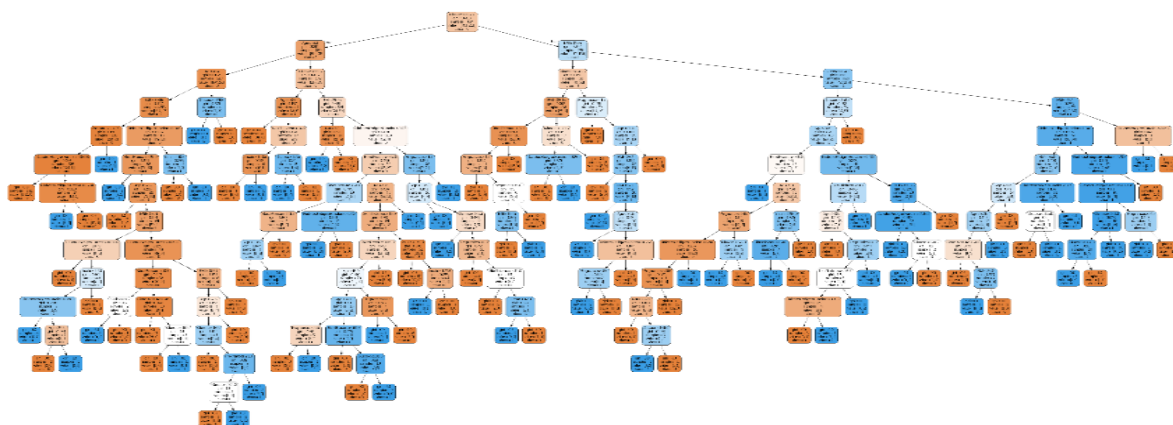


Figure 3. Decision Tree Classification for Max Depth

```

# Create Decision Tree classifier object
model = DecisionTreeClassifier(criterion="entropy", max_depth=3)

# Train Decision Tree Classifier
model = model.fit(x_train,y_train)

#Predict the response for test dataset
y_pred = model.predict(x_test)

# Model Accuracy
print("Accuracy:",metrics.accuracy_score(y_test, y_pred)*100)
    
```

Prediction Using SVM and DT

Accuracy: 79.87012987012987

The classification rate increased to 79.87%, which is better accuracy than the previous model.

#Better Decision Tree Visualisation

from sklearn.externals.six import StringIO

from IPython.display import Image

from sklearn.tree import export_graphviz

import pydotplus

dot_data = StringIO()

export_graphviz(model,out_file=dot_data,filled=True, rounded=True,special_characters=True,

feature_names = features,class_names=['0','1'])

graph = pydotplus.graph_from_dot_data(dot_data.getvalue())

graph.write_png('diabetes_set.png')

Image(graph.create_png())

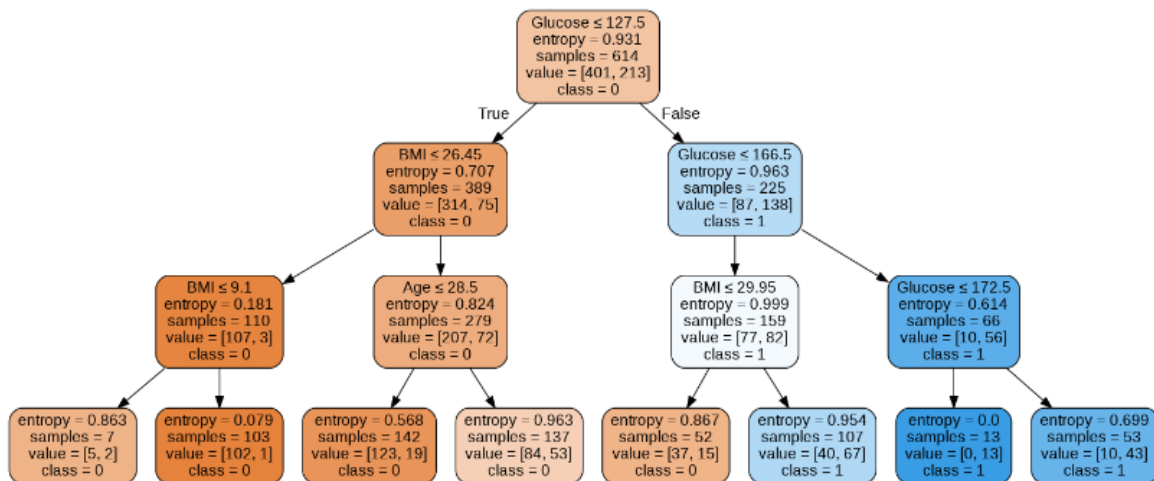


Figure 4. Decision Tree Classification for Max Depth 3

C. Receiver Operating Characteristics

Receiver Operating Characteristics

```
probs=log.predict_proba(x_train)[:,1]
```

```
fpr, tpr, threshold=metrics.roc_curve(y_train,probs )
```

```
plt.plot([0,1],[0,1],'k--')
```

```
plt.plot(fpr,tpr, label='logistic')
```

```
plt.xlabel('fpr')
```

```
plt.ylabel('tpr')
```

```
plt.show()
```

The Receiver Operating Characteristics (ROC) serves as an illustrative graphical representation, wherein sensitivity is juxtaposed against '1-specificity.' The Area Under the Curve (AUC), derived from the ROC curve, functions as a pivotal metric for assessing the efficacy of classifiers [45]. The AUC, a scalar ranging between '0' and '1,' encapsulates the discriminative power of the classifier as shown in the figure 5.

Prediction Using SVM and DT

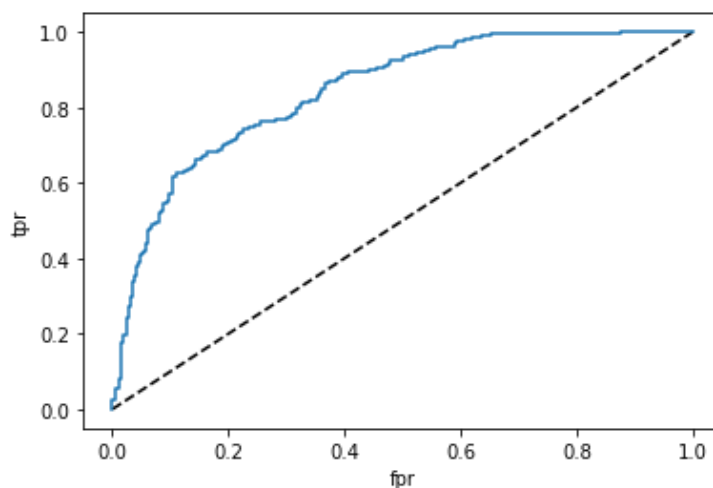


Figure 5. Roc Curve

Derived from the acquired results, we have demonstrated not only the heightened accuracy attainable through the adept management of imbalanced datasets but also the superior precision achievable via judicious feature selection. Noteworthy in our findings is the discerning efficacy of the decision tree classifier algorithm, surpassing its counterparts in performance. This discernment holds particular promise for the healthcare industry, given the discerning attributes of the model. Its cost-effectiveness and temporal efficiency make it an appealing prospect, promising not only enhanced accuracy but also practical feasibility in real-world healthcare applications.

The efficacy of the proposed models, enriched with interaction terms, surpasses that of their counterparts devoid of such interactions. This heightened efficiency is attributed to the deliberate inclusion of interactions with pivotal risk factors influencing diabetes, namely, body mass index, and a familial predisposition to diabetes. The astute integration of these interaction terms elevates the models, rendering them more discerning in their predictive capacity.

The outcomes of this research proffer a promising avenue for the development of a program tailored to the judicious screening of prospective diabetes patients in the times ahead. Nonetheless, it is imperative to acknowledge that additional facets pertaining to physical activity, lifestyle considerations (such as the waist-to-height ratio), and meticulous dietary management (embracing control over protein, fat, and sugar intake) have emerged as salient risk factors for diabetes [19].

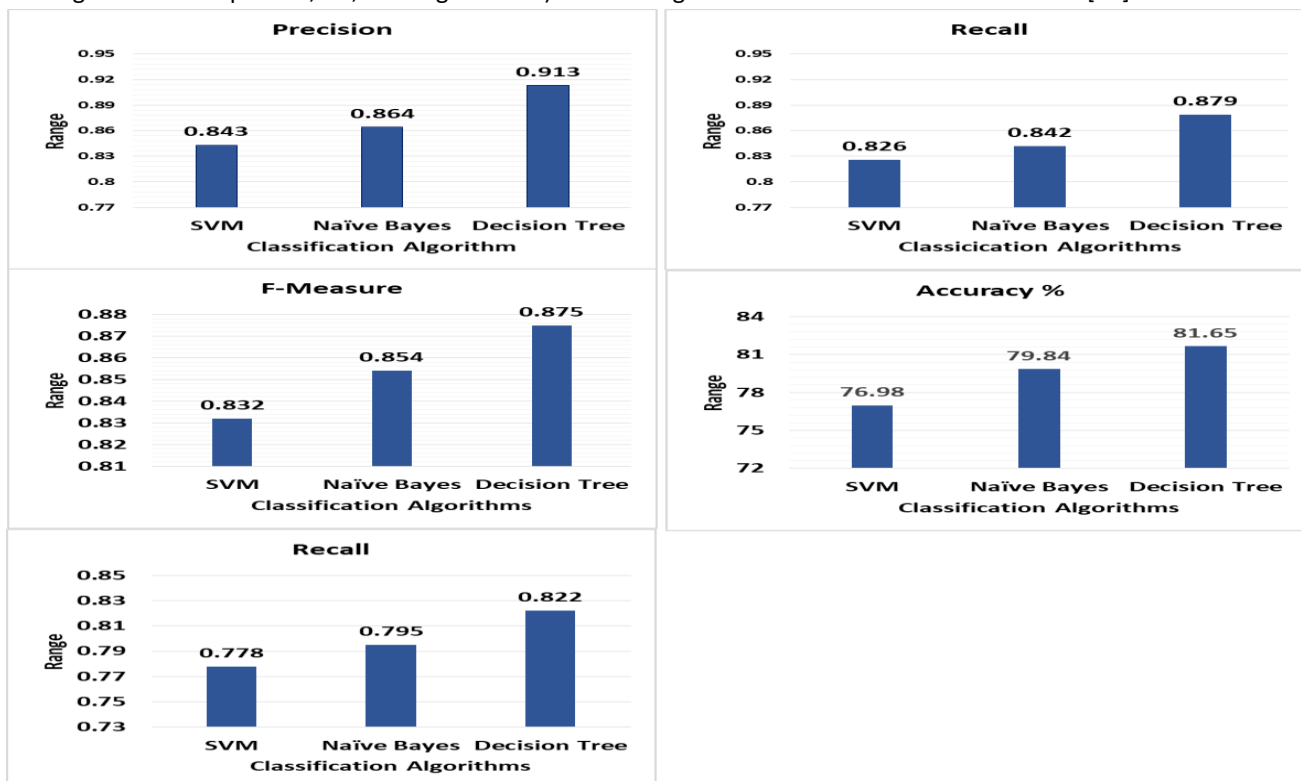


Figure 6. Error Metrics Interpretation

Prediction Using SVM and DT

Furthermore, the nexus between certain metabolites and prediabetes/diabetes has been elucidated in extant literature [20]. Consequently, future investigations may find merit in incorporating these multifarious risk factors into the purview of consideration when crafting classification models for diabetes.

VII. CONCLUSION

In a broader context, it is asserted that decision tree analysis stands as a predictive modelling tool with versatile applications across various domains. Employing an algorithmic approach, decision trees can be crafted to partition datasets in diverse ways contingent upon specific conditions.

Upon completion of the undertaken work, a definitive observation emerges: the augmentation of data allocated for training the model directly correlates with an improved accuracy estimate. In our specific case, optimal results are attained by bifurcating the data evenly, dedicating 50% for training the model and an equivalent 50% for testing, yielding an accuracy of 0.71.

Upon scrutinizing the developed diagnostic model, several advantages come to the fore: a rapid learning process, the formulation of rules in domains challenging for an expert to formalize, an intuitive classification model, elevated prediction accuracy comparable to alternative data analysis methods such as statistics and neural networks, and the construction of nonparametric models.

REFERENCES

- 1) Aishwarya, R., Gayathri, P., Jaisankar, N., 2013. A Method for Classification Using Machine Learning Technique for Diabetes. *International Journal of Engineering and Technology (IJET)* 5, 2903–2908.
- 2) Aljumah, A.A., Ahamad, M.G., Siddiqui, M.K., 2013. Application of data mining: Diabetes health care in young and old patients. *Journal of King Saud University - Computer and Information Sciences* 25, 127–136.
- 3) Arora, R., Suman, 2012. Comparative Analysis of Classification Algorithms on Different Datasets using WEKA. *International Journal of Computer Applications* 54, 21–25.
- 4) Bamnote, M.P., G.R., 2014. Design of Classifier for Detection of Diabetes Mellitus Using Genetic Programming. *Advances in Intelligent Systems and Computing* 1, 763–770.
- 5) Choubey, D.K., Paul, S., Kumar, S., Kumar, S., 2017. 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.
- 6) Dhomse Kanchan B., M.K.M., 2016. Study of Machine Learning Algorithms for Special Disease Prediction using Principal of Component Analysis, in: *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication*, IEEE. pp. 5–10.
- 7) Esposito, F., Malerba, D., Semeraro, G., Kay, J., 1997. A comparative analysis of methods for pruning decision trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19, 476–491.
- 8) Fatima, M., Pasha, M., 2017. Survey of Machine Learning Algorithms for Disease Diagnostic. *Journal of Intelligent Learning Systems and Applications* 09, 1–16.
- 9) Garner, S.R., 1995. Weka: The Waikato Environment for Knowledge Analysis, in: *Proceedings of the New Zealand computer science research students conference*, Citeseer. pp. 57–64.
- 10) Han, J., Rodriguez, J.C., Beheshti, M., 2008. Discovering decision tree based diabetes prediction model, in: *International Conference on Advanced Software Engineering and Its Applications*, Springer. pp. 99–109.
- 11) Kumar, D.A., Govindasamy, R., 2015. Performance and Evaluation of Classification Data Mining Techniques in Diabetes. *International Journal of Computer Science and Information Technologies*, 6, 1312–1319.
- 12) Kumar, P.S., Umatejaswi, V., 2017. Diagnosing Diabetes using Data Mining Techniques. *International Journal of Scientific and Research Publications* 7, 705–709.
- 13) Kumari, V.A., Chitra, R., 2013. Classification Of Diabetes Disease Using Support Vector Machine. *International Journal of Engineering Research and Applications (IJERA)* 1797–1801.
- 14) Nai-Arun, N., Mounghmai, R., 2015. Comparison of Classifiers for the Risk of Diabetes Prediction. *Procedia Computer Science* 69, 132–142.
- 15) Nai-Arun, N., Sittidech, P., 2014. Ensemble Learning Model for Diabetes Classification. *Advanced Materials Research* 931-932, 1427–1431.
- 16) Orabi, K.M., Kamal, Y.M., Rabah, T.M., 2016. Early Predictive System for Diabetes Mellitus Disease, in: *Industrial Conference on Data Mining*, Springer. pp. 420–427

Prediction Using SVM and DT

- 17) Pradhan, P.M.A., Bamnote, G.R., Tribhuvan, V., Jadhav, K., Chabukswar, V., Dhobale, V., 2012. A Genetic Programming Approach for Detection of Diabetes. International Journal Of Computational Engineering Research 2, 91–94.
- 18) Sharief, A.A., Sheta, A., 2014. Developing a Mathematical Model to Detect Diabetes Using Multigene Genetic Programming. International Journal of Advanced Research in Artificial Intelligence (IJARAI) 3, 54–59.
- 19) Tarik A. Rashid, S. M.A., Abdullah, R.M., Abstract, 2016. An Intelligent Approach for Diabetes Classification, Prediction and Description. Advances in Intelligent Systems and Computing 424, 323–335.



There is an Open Access article, distributed under the term of the Creative Commons Attribution – Non Commercial 4.0 International (CC BY-NC 4.0) (<https://creativecommons.org/licenses/by-nc/4.0/>), which permits remixing, adapting and building upon the work for non-commercial use, provided the original work is properly cited.