JOURNAL OF
EMERGING INVESTIGATORS

Article

Prediction of diabetes using supervised classification

Andre Sun[1,2], Faneng Sun[1]
[1]American High School, Fremont, California
[2]Horizon Academic Research Program, Roeland Park, Kansas

SUMMARY

Diabetes is one of the common chronic diseases that impacts 28.7 million people in the US as of 2019, accounting for 8.7% of the total population. Early identification of diabetes is very important in disease control and management. A number of prior studies provided compelling evidence that machine learning can help identify diabetes early allowing for timely treatment. It remains a challenge to appropriately assess, optimize and refine the classification models based on specific dataset for diabetes prediction with high accuracy. In this study, we aimed to develop a model with improved accuracy for diabetes prediction. We employed six learning algorithms, logistic regression, k-nearest neighbors (k-NN), support vector machine (SVM), decision tree, random forest, and gradient boosting on the Pima Indians Diabetes Dataset. The performance of each model was evaluated for the prediction of diabetes in validation datasets using accuracy, precision, recall, and F1-score. Gradient boosting provided an accuracy of 81.8%, outperforming all other classification models in most of the performance measures. Collectively, the gradient boosting model appeared to provide an appropriate algorithm for diabetes prediction with high accuracy based on the diagnostic measurements gathered in this specific dataset. Of note, the insights yielded from this exploratory study may only be applicable to this subpopulation of diabetes patients. It remains to be further validated with datasets derived from more diverse diabetes populations before the findings can be generalized to a wider diabetes patient population.

INTRODUCTION

Diabetes is a common chronic disease, affecting 8.7% of the population in the United States (1). Diabetes occurs mainly because of insufficient insulin content in the blood, resulting in dysregulation of blood glucose metabolism (2). Typical symptoms of diabetes include frequent urination, thirst, and hunger (3). If the disease is left untreated at the initial stage, significant complications such as stroke, lung illness, vision impairment, renal failure, and mortality may occur (4). The practical challenge for the early detection of diabetes includes the slow progression of disease, which often goes undetected. Screening to predict the people at high risk of diabetes, early identification, is very important in controlling and managing the illness.

Various prominent research studies have focused on diabetes prediction. A number of classification algorithms have been shown to work well for prediction of diabetes in patients. Sisodia and Sisodia conducted research focused on pregnant women suffering from diabetes and evaluated three predictive models based on Naive Bayes, support vector machine (SVM), and decision tree classification algorithms on the Pima Indian Diabetes Dataset (PIDD) (5). The highest prediction accuracy of 76.30% was achieved by Naïve Bayes (5). Alam *et al.* compared the performance of artificial neural network, random forest, and K-means cluttering for prediction of diabetes using PIDD and reported the artificial neural network provided a best accuracy of 75.7% (8). Several studies also evaluated several predictive models based on artificial neural network, k-Nearest Neighbor (k-NN) and others for diabetes prediction (7-8).

In this study, we extensively evaluated six machine learning algorithms including logistic regression, k-nearest neighbor (k-NN), SVM, decision tree, random forest and gradient boosting for prediction of diabetes based on the PIDD. PIDD consists of seven medical predictor variables including the number of pregnancies the patient has had, glucose, insulin, diastolic blood pressure, tricep skin fold thickness, body mass index, diabetes pedigree function, and age of 768 female patients as well as their diabetic status. Here is a brief description of each algorithm used. Logistic regression is used to model the probability of a certain class or event and predict a dependent categorical target variable (9). k-NN uses proximal k neighbors to make classifications or predictions about the grouping of an individual data point (10). SVM creates the best decision boundary that can segregate n-dimensional space into classes (11). Decision tree is a tree-structured-based model which describes the classification process based on input features (12, 13). Random forest is an ensemble learning method that constructs multiple decision trees (called estimators), with each tree producing their own predictions and the predictions of the estimators are combined to produce a more accurate prediction (13, 14). Gradient boosting is a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. It fits a new predictor to the residual error made by the previous predictor (13, 15, 16). The prediction performance of all six algorithms are compared on various measures. Our hypothesis is that effective prediction of diabetes with at least 80% accuracy can be achieved with appropriate algorithms together with various features related to diabetes onset. In this work, we evaluated the performance of six different algorithms, among which gradient boosting provided an appropriate algorithm for diabetes prediction with the highest prediction accuracy (81.8%) for the particular population evaluated in this study.

DOI: https://doi.org/10.59720/23-062

RESULTS

Model training and prediction of new instances consisted of three main steps: data preprocessing, classification model development, and prediction of a validation dataset (**Figure 1**). First, we conducted data preprocessing to handle any missing information, mainly to handle missing data and to normalize the values of each feature. Next, we explored various machine learning algorithms to develop classification models. Finally, we used the prediction of unseen data to evaluate each trained model.

We conducted analysis and prediction of diabetes on the PIDD (17). In this dataset, there were 768 female patients with eight features, which included the number of times the patient has been pregnant, plasma glucose concentration at two hours in an oral glucose tolerance test, diastolic blood pressure, tricep skin fold thickness, two-hour serum insulin, body mass index (BMI), diabetes pedigree function, and age. Five hundred records were non-diabetic while 268 were diabetic patients.

The age of patients ranged from 21 to 81 and the average number of pregnancies was 3.8. Distribution of some attributes among diabetic and non-diabetic patients is different. The mean BMI for the people who were non-diabetic was 30.9 while the average was 35.4 for those with diabetes. The mean plasma glucose concentration was 142.3 for the people with diabetes and 110.6 for those without diabetes (**Table 1**). Features including glucose, blood pressure, skin thickness, insulin, and BMI had various number of missing values.

The distribution of age, number of pregnancies, insulin, diabetes pedigree function, and skin thickness were right-skewed (**Figure 2A**). The skew of the skin thickness and insulin data was largely due to a large number of missing values, which appeared as 0 values in the dataset. After the imputation of missing values using their means, the distribution of skin thickness and insulin was approximately normally distributed (**Figure 2B**). Glucose levels, BMI, age, and number of pregnancies have relatively high correlations with diabetic status when evaluated using Spearman's rank correlation (**Figure 3**). In addition, correlations between pairs of features were also observed, like age and pregnancies, or insulin and skin thickness, with correlation coefficients of 0.54 and 0.44, respectively.

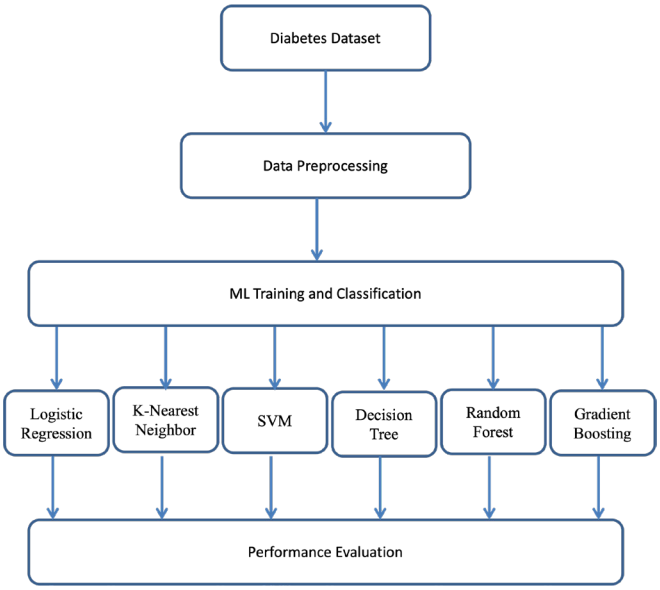

Figure 1: Proposed work procedure. Framework for evaluating predictive models. Data preprocessing involved imputation of missing data and standardization of values for each feature. Model training and classification included development of prediction models using various classification algorithms based on the training data. Prediction and performance evaluation includes the prediction of the diabetic/non-diabetic status of unseen data in the testing dataset and evaluation of the performance of the classification algorithms.

		All	Diabetic	Non-Diabetic
Number of Subjects		768	268	500
Pregnancy (times)	mean (SD)	3.8 (3.4)	4.9 (3.7)	3.3 (3.0)
	(min, max)	(0, 17)	(0, 17)	(0, 13)
Glucose (mg/dL)	mean (SD)	121.7 (30.5)	142.3 (29.6)	110.6 (24.8)
	(min, max)	(44, 199)	(78, 199)	(44, 197)
	missing	5	2	3
Blood Pressure (mmHg)	mean (SD)	72.4 (12.4)	75.3 (12.3)	70.9 (12.2)
	(min, max)	(24, 122)	(30, 114)	(24, 122)
	missing	35	16	19
Skin Thickness (mm)	mean (SD)	29.2 (10.5)	33.0 (10.3)	27.2 (10.0)
	(min, max)	(7, 99)	(7, 99)	(7, 60)
	missing	227	88	139
Insulin (μU/ml)	mean (SD)	155.5 (118.8)	206.8 (132.7)	130.3 (102.5)
	(min, max)	(14, 846)	(14, 846)	(15, 744)
	missing	374	138	236
BMI (Kg/m^2)	mean (SD)	32.5 (6.9)	35.4 (6.6)	30. 9 (6.6)
	(min, max)	(18.2, 67.1)	(22.9, 67.1)	(18.2, 57.3)
	missing	11	2	9
Diabetes Pedigree Function	mean (SD)	0.47 (0.33)	0.55 (0.37)	0.43 (0.30)
	(min, max)	(0.08, 2.42)	(0.09, 2.42)	(0.08, 2.32)
Age (Years)	mean (SD)	33.2 (11.8)	37.1 (11.0)	31.2 (11.7)
	(min, max)	(21, 81)	(21, 70)	(21, 81)

Table 1: Dataset characteristics in Diabetic and non-Diabetic patients. All: all subjects in the dataset, Diabetic: patients with diabetes, Non-Diabetic: non-diabetic patients, mean: average values of non-missing data, SD: standard deviation, missing: number of missing values in each group.

Six supervised machine learning models were developed using the PIDD. Predictive models were trained with 80% of the dataset and tested with the remaining 20% of the data. The distribution of each feature in the training and testing dataset appears to be comparable (**Figure 4**). Five-fold cross-validation was employed to optimize the model parameters. The optimal parameters such as regulation strength, regulation technique for logistic regression, number of neighbors for k-NN, kernel for SVM, tree-related parameters for decision tree, random forest and gradient boosting used to train each classification model are summarized here (**Table 2**).

The confusion matrix shows the classified instances in the training and testing datasets based on different classification models (**Table 3**). In the training dataset, the determined accuracy for the most accurate model gradient boosting was 81.6 and 75.9% for the least accurate model logistic regression. In the testing dataset, the highest predicted accuracy was observed with gradient boosting (81.8%), followed by random forest (80.5%), while the least accuracy (76.6%) was observed with k-NN and decision tree.

In addition to the accuracy, which determines how well diabetic status determined by algorithms agree with patients' true diabetic status, we also compared the performance of each model for prediction of diabetes in the training and testing dataset using precision, recall, F1-core, and area under the curve of the receiver operating characteristic (AUC-ROC). Precision determines the classifier's ability to provide correct positive predictions of diabetes. Recall is the proportion of

DOI: https://doi.org/10.59720/23-062

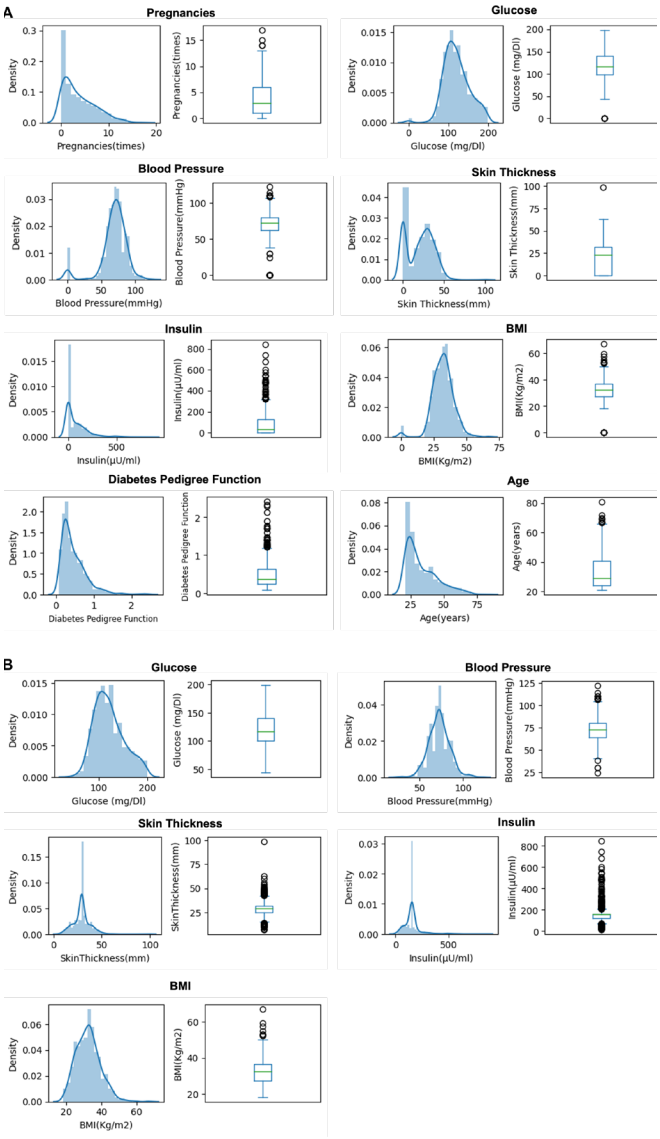

Figure 2: Distribution of each feature in the original data and after imputation. A) Density plots and box plots display the distribution of numeric values for the number of pregnancies (Pregnancies), plasma glucose concentration (Glucose), blood pressure (BloodPressure), skin thickness (SkinThickness), Insulin, BMI, diabetes degree function (DiabetesPedigreeFunction), and Age of the patients in the PIDD. **B)** Density plots and box plots display the distribution of numeric values after imputation of missing values with their means for plasma glucose concentration (Glucose), blood pressure (BloodPressure), skin thickness (SkinThickness), Insulin, BMI.

actual positive cases of diabetes correctly identified by the classifier used. F1-score is the weighted average of precision and recall. AUC-ROC is the measure of the ability of classifier to distinguish between classes. We observed that the gradient boosting model outperformed other models with regard to F1-score and precision, and better than most other models in recall and AUC-ROC (**Table 4**).

DISCUSSION

Detection of diabetes at an early stage is very important for timely treatment. In this study, we conducted diabetes prediction based on six different machine learning algorithms using PIDD, with 80% of the data used for training and 20% for testing. Classification algorithms that we focused on included logistic regression, k-NN, SVM, decision tree, random forest, and gradient boosting. They were first trained with all possible features from the dataset and respective diabetic/non-diabetic status in the training dataset; then the same algorithms were used for the prediction of the diabetic/non-diabetic status of unseen data in the testing dataset.

In this study, following the pre-processing of the datasets, we tuned the hyperparameters and identified the optimal parameters based on five-fold cross validation for each of the six machine learning algorithms to achieve the best performance. The predictive performance of each model was evaluated on the unseen testing dataset using various performance metrics. The gradient boosting classification model outperformed all other classification models evaluated in this study in terms of accuracy of diabetic prediction for this particular patient population. Using the same dataset, Birjais *et al.* conducted a study on detecting diabetes using gradient boosting, logistic regression, and Naive Bayes machine learning algorithms and also found that gradient boosting had the highest accuracy (18). In their study, the highest accuracy in the testing dataset was obtained with gradient boosting at 86%, followed by logistic regression at 79% and Naive Bayes at 77% (18). Raja *et al.* compared a gradient boosted classifier with random forest and neural networks employed on the PIDD (7). It was found that gradient boosting also outperformed others (7). Using a different dataset, Seto *et al.* reported that gradient boosting provides a more reliable model than logistic regression in predicting diabetes probability (19). Those prior studies laid a solid foundation for future investigation, but they involved limited options of the machine learning classification or missed reporting information about the specific classification parameters. Our study intended to explore a thorough and

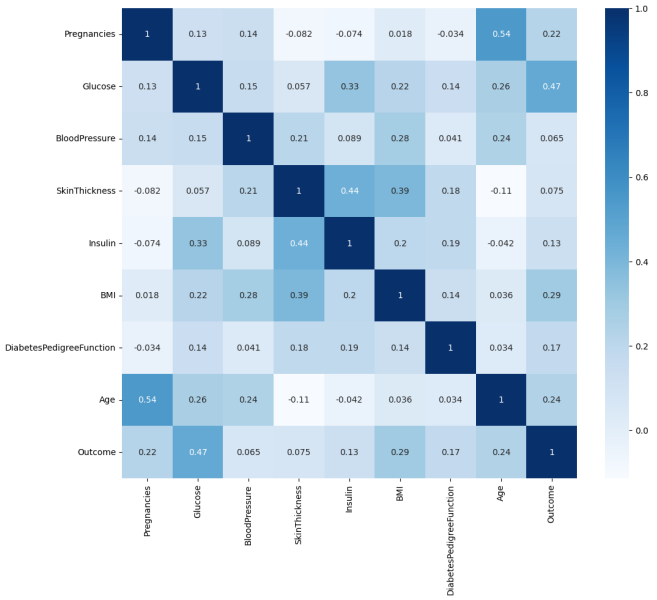

Figure 3: Correlation between features and diabetic status (outcome). Heatmap showing the correlation between the variables on each axis. Outcome is diabetic/non-diabetic status. Correlation between variables was evaluated using the Spearman method.

DOI: https://doi.org/10.59720/23-062

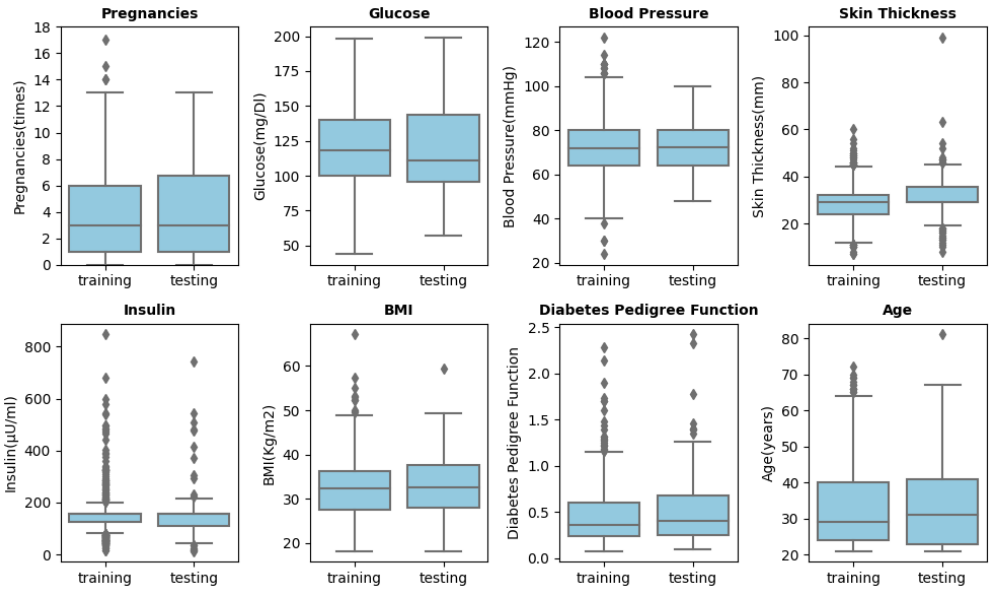

Figure 4. Distribution of each feature in the training and testing datasets. Distribution of numeric values in the training and testing datasets for the number of pregnancies (Pregnancies), plasma glucose concentration (Glucose), blood pressure (Blood Pressure), skin thickness (Skin Thickness), Insulin, BMI, diabetes degree function (DiabetesPedigreeFunction), and Age.

detailed comparison of broader and commonly used machine learning algorithms for the PIDD. Here we report our findings that gradient boosting provided better prediction accuracy, aligning with the observations from prior studies.

To obtain higher accuracy of diabetes prediction, the application of advanced classification algorithms could be further assessed in the future. For example, feature selection could be applied to remove irrelevant feature(s). Imputation of missing values could be conducted by using different methods such as interpolation and expectation maximization (20 - 21). Different k-fold, for example, 10-fold cross-validation could be explored to optimize models. Z-scores or inter quantile range could provide different ways to detect outliers; subsequently, the quantile-based flooring and capping, or mean/median imputation could also be further assessed to improve the model performance.

This exploratory study provided promising results in the prediction of diabetes, with gradient boosting classification showing the highest accuracy. Future studies with larger datasets and more diverse patient populations will provide necessary insights whether gradient boosting could achieve a similar and satisfactory prediction accuracy for a general population with a more diverse background.

MATERIALS AND METHODS

Data Set

The Pima Indians Diabetes Dataset was used for this study. This dataset consists of 768 female patients with eight features. Features included the number of times the patient had been pregnant, plasma glucose concentration at two hours in an oral glucose tolerance test, diastolic blood pressure, tricep skin fold thickness, two-hour serum insulin, body mass index, diabetes pedigree function, and age.

Data Visualization

The distribution of each feature was visualized using a density plot and a box plot. The distribution of glucose, blood pressure, skin thickness, insulin and BMI were re-plotted in the same way after the missing values were imputed. These features contained zero values, which were treated as missing values. Zero values of these features were imputed with their mean values calculated based on non-missing data. The correlation between features and diabetic status was evaluated using the Spearman method using the following

Classification Algorithms	Parameters	Definition of Parameters
logistic regression	C=1.623776739188721	inverse of regularization strength
	max_iter = 5	maximum number of iterations taken for the solvers to converge
	penalty='l2'	regulation technique to prevent overfitting
	solver= 'sag'	algorithm used to optimize
k-NN	n_neighbors=31	number of neighbors
SVM	C = 0.015	inverse of regularization strength
	kernel = linear	a method used to take data as input and transform it into the required form of processing data
decision tree	criterion=entropy	function to measure the quality of a split
	max_depth=4	maximum depth of the tree
	max_features=8	number of features to consider when looking for the best split
random forest	bootstrap = True	whether bootstrap samples are used when building trees
	max_depth = 4	maximum depth of the tree
	max_features =sqrt of number of features	number of features to consider when looking for the best split
	min_samples_leaf = 6	minimum number of samples required to be at a leaf node
	min_samples_split = 25	minimum number of samples required to split an internal node
	n_estimators = 30	number of trees in the forest
gradient boosting	learning_rate = 0.01	learning rate shrinks the contribution of each tree by learning rate
	n_estimators = 200	number of trees
	max_depth=3	maximum depth of the tree
	max_features = sqrt of number of features	number of features to consider when looking for the best split
	min_samples_leaf= 5	minimum number of samples required to be at a leaf node
	min_samples_split = 5	minimum number of samples required to split an internal node
	subsample=0.8	fraction of samples used for fitting the individual base learners

Table 2: Specific parameters used to train each classification model.

DOI: https://doi.org/10.59720/23-062

Classification Algorithms	Diabetes Status	Predicted (Training)		Training Accuracy	Predicted (Testing)		Testing Accuracy
		non-diabetic	diabetic		non-diabetic	diabetic	
logistic regression	non-diabetic	TN=345	FP=55	75.9%	TN=85	FP=15	79.2%
	diabetic	FN=93	TP=121		FN=17	TP=37	
k-NN	non-diabetic	TN=355	FP=45	77.9%	TN=86	FP=14	76.6%
	diabetic	FN=91	TP=123		FN=22	TP=32	
SVM	non-diabetic	TN=355	FP=45	76.0%	TN=90	FP=10	80.5%
	diabetic	FN=102	TP=112		FN=20	TP=34	
decision tree	non-diabetic	TN=366	FP=34	77.7%	TN=87	FP=13	76.6%
	diabetic	FN=103	TP=111		FN=23	TP=31	
random forest	non-diabetic	TN=366	FP=34	80.9%	TN=89	FP=11	80.5%
	diabetic	FN=84	TP=130		FN=19	TP=35	
gradient boosting	non-diabetic	TN=372	FP=28	81.6%	TN=91	FP=9	81.8%
	diabetic	FN=85	TP=129		FN=19	TP=35	

Table 3: Confusion matrix and prediction accuracy of various classification algorithms for training and testing datasets. TP: True Positive, FP: False Positive, FN: False Negative, TN: True Negative.

formula:

$$r_s = 1 - \frac{6\sum_1^n \sum d_i^2}{n(n^2-1)}$$

where r_s denotes spearman correlation coefficient, and d_i is the difference between two ranks of each observation and n is the number of observations. Correlation values were displayed in heatmaps generated using the Python seaborn package.

Data Preprocessing for Missing Values

In this dataset, we did not observe any missing values. However, there were five features (Glucose, Blood Pressure, Skin Thickness, Insulin, and BMI) that contained zero values, which are not biologically possible. These values were treated as missing values and imputed with their mean values.

Data Splitting

The dataset was randomly divided with a ratio of 80/20 into the training dataset and testing dataset, respectively. The training dataset, consisting of 614 samples, was used for developing classification models. The testing dataset, consisting of 154 samples, was used to evaluate the performance of classification models.

Data Normalization

Each feature in the training dataset was normalized to have a mean of 0 and a standard deviation of 1. Specifically, the mean was subtracted from each feature and the resulting values were divided by the feature's standard deviation. To avoid data leakage, each feature in the testing dataset was also normalized using the mean and standard deviation of the corresponding feature in the training dataset. The normalization was performed using the following algorithm:

$$x' = \frac{x - \text{mean}(x_{\text{train}})}{\text{sd}(x_{\text{train}})}$$

Classification Models

Six supervised algorithms were employed to develop predictive models for diabetes detection: logistic regression, k-NN, SVM, decision tree, random forest, and gradient boosting.

All classification models were fit, evaluated and optimized using python's scikit-learn (sklearn) library. For each model, hyperparameters were tuned to optimize the model architecture. Grid search or random search was used to search hyperparameter space for optimum values. Models were built for each possible combination of all the hyperparameter values or random sampling of the distribution for each hyperparameter from which values were provided. Model parameters were learned during training by optimizing a loss function. Five-fold cross validation was used to evaluate each model. That is, training data was randomly split into five folds, then the model was trained on four of the five folds, while one fold was left to test a model. The classification error was computed in the held-out fold. This procedure was repeated five times for each model. A different group of observations was treated as a validation set each time.

Logistic Regression

The logistic regression model was fit, evaluated and optimized using LogisticRegression class from Python's sklearn library. The hyperparameters for logistic regression were tuned including the basis of the algorithm (solver), regularization penalty, and regularization strength. Tuning of regularization penalty and penalty strength was intended to avoid the risk of overfitting due to learning a complex model. The evaluated solvers included lbfgs, newton-cg, liblinear, sag and saga. Regularization penalty included l1, l2 and elasticnet. Twenty different regulation strengths were assessed. Inverses of these regulation strengths are evenly spaced values between 0.0001 and 10000 on logspace.

k-Nearest Neighbors (k-NN)

k-NN algorithm was developed using sklearn's KneighborsClassifier class. K nearest neighbors were selected based on distance, which was calculated using the Minkowski method. The class label is assigned on the basis of a majority agreement. The number of neighbors was tuned for k-NN model.

Support Vector Machine (SVM)

Linear and radial base function (RBF) SVM classifiers were developed and optimized using svm.SVC class sklearn library. SVM was tuned for the kernel (linear and RBF) that controls the manner in which the input variables are projected and the penalty that affects the shape of the resulting regions for each class.

Datasets	Performance Measure	logistic regression	k-NN	SVM	decision tree	random forest	gradient boosting
Training	Accuracy (%)	75.9	77.9	76.1	77.7	80.8	81.6
	Precision	0.69	0.73	0.71	0.77	0.79	0.82
	Recall	0.57	0.57	0.52	0.52	0.61	0.60
	F1-score	0.62	0.64	0.60	0.62	0.69	0.70
	AUC-ROC	0.84	0.85	0.84	0.86	0.89	0.91
Testing	Accuracy (%)	79.2	76.6	80.5	76.6	80.5	81.8
	Precision	0.71	0.70	0.77	0.70	0.76	0.80
	Recall	0.69	0.59	0.63	0.57	0.65	0.65
	F1-score	0.70	0.64	0.69	0.63	0.70	0.71
	AUC-ROC	0.85	0.82	0.85	0.82	0.84	0.85

Table 4: Performance measures of the classification algorithms for prediction of training and testing datasets. k-NN: K-nearest neighbors, Accuracy: proportion of correctly predicted diabetic and non-diabetic instances, Precision: proportion of patients have diabetes among all the instances predicted to be diabetic. Recall: proportion of actual positive cases of diabetes correctly identified by the classifier used. F1 score is the weighted average of precision and recall. AUC-ROC: area under the curve of the receiver operating characteristics.

DOI: https://doi.org/10.59720/23-062

Decision Tree

The decision tree classifier was constructed using sklearn's DecisionTreeClassifier class. The classification algorithm was tuned with respect to the maximum depth of a tree, number of features to consider when looking for the best split, the minimum number of samples required to split at an internal node, and the impurity of a split.

Random Forest

Random forest algorithm was created and optimized using RandomForestClassifier class from sklearn library. The random forest was tuned with respect to the number of decision tress, the maximum depth, number of features to consider when looking for the best split, minimum number of samples required to split an internal node, and minimum samples (or observations) required in a terminal node or leaf.

Gradient Boosting

The gradient boosting classifier was built and optimized using sklearn's GradietBoostingClassifier class. The overall parameters of this model include tree-specific parameters and boosting parameters. The main hyperparameters that were tuned are maximum depth of a tree, number of features to consider when looking for the best split, minimum number of samples required to split an internal node, minimum samples (or observations) required in a terminal node or leaf and learning rate.

Performance Evaluations

Performance of all six models for predicting diabetes was evaluated using metrics including accuracy, precision, recall, f1-score and area under AUC-ROC. The formulas used to calculate these performance metrics are shown below:

		Predicted	
		Negative	Positive
Actual	Negative	A	b
	Positive	C	d

$$\text{Accuracy} = \frac{(a + d)}{(a + b + c + d)} * 100\%$$

$$\text{Precision} = \frac{d}{(d + b)}$$

$$\text{Recall} = \frac{d}{(d + c)}$$

$$f1 = 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Software and packages

Data preprocessing, plotting, model fitting, model selection, and model evaluation were performed using python (version 3.10.11) in collaboration with the following versions of packages: numpy (version 1.22.4), panda (version 1.5.3), matplotlib (version 3.7.1), seaborn (version 0.12.2) and scikit-learn (version 1.2.2) libraries.

Received: March 29, 2023
Accepted: July 2, 2023
Published: March 17, 2024

REFERENCES

1. "Prevalence of Diagnosed Diabetes." *Centers for Disease Control and Prevention*, 30 Sept. 2022, www.cdc.gov/diabetes/data/statistics-report/diagnosed-diabetes.html.
2. American Diabetes Association. "Diagnosis and Classification of Diabetes Mellitus." *Diabetes Care*, vol. 33, no. Supplement_1, 2010, https://doi.org/10.2337/dc10-s062.
3. Clark, Nathaniel G., et al. "Symptoms of Diabetes and Their Association with the Risk and Presence of Diabetes." *Diabetes Care*, vol. 30, no. 11, 2007, pp. 2868–2873, https://doi.org/10.2337/dc07-0816.
4. Saedi, Elham, et al. "Diabetes Mellitus and Cognitive Impairments." *World Journal of Diabetes*, vol. 7, no. 17, 2016, p. 412, https://doi.org/10.4239/wjd.v7.i17.412.
5. Sisodia, Deepti, and Dilip Singh Sisodia. "Prediction of Diabetes Using Classification Algorithms." *Procedia Computer Science*, vol. 132, 2018, pp. 1578–1585, https://doi.org/10.1016/j.procs.2018.05.122.
6. Alam, Talha Mahboob, et al. "A Model for Early Prediction of Diabetes." *Informatics in Medicine Unlocked*, vol. 16, 2019, p. 100204, https://doi.org/10.1016/j.imu.2019.100204.
7. Raja, J.Beschi, et al. "Diabetics Prediction Using Gradient Boosted Classifier." *International Journal of Engineering and Advanced Technology*, vol. 9, no. 1, 2019, pp. 3181–3183, https://doi.org/10.35940/ijeat.a9898.109119.
8. Kaur, Harleen, and Vinita Kumari. "Predictive Modelling and Analytics for Diabetes Using a Machine Learning Approach." *Applied Computing and Informatics*, vol. 18, no. 1/2, Jul. 2020, pp. 90–100, https://doi.org/10.1016/j.aci.2018.12.004.
9. Cramer, J. S. *The Origins of Logistic Regression*. Tinbergen Institute, 2002.
10. Zhang, Zhongheng. "Introduction to Machine Learning: K-Nearest Neighbors." *Annals of Translational Medicine*, vol. 4, no. 11, 2016, pp. 218–218, https://doi.org/10.21037/atm.2016.03.37.
11. Cortes, Corinna, and Vladimir Vapnik. "Support-Vector Networks." *Machine Learning*, vol. 20, no. 3, 1995, pp. 273–297, https://doi.org/10.1007/bf00994018.
12. Maimon, Oded, and Lior Rokach. *Data Mining and Knowledge Discovery Handbook*. Springer US, 2005.
13. Hastie, Trevor, et al. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2017.
14. Breiman, Leo. "Random Forests." *Machine Learning*, vol. 45, no. 1, 2001, pp. 5–32, https://doi.org/10.1023/a:1010933404324.
15. Schapire, Robert E. "The Strength of Weak Learnability." *Machine Learning*, vol. 5, no. 2, 1990, pp. 197–227, https://doi.org/10.1007/bf00116037.
16. Zhou, Zhi-Hua. *Ensemble Methods: Foundationa and Algorithms*. Taylor & Francis, 2012.
17. Akturk, Mehmet. "Diabetes Dataset." *Kaggle*, 5 Aug. 2020, www.kaggle.com/datasets/mathchi/diabetes-data-set.
18. Birjais, Roshan, et al. "Prediction and Diagnosis of Future Diabetes Risk: A Machine Learning Approach." *SN Applied Sciences*, vol. 1, no. 9, 2019, https://doi.org/10.1007/s42452-019-1117-9.
19. Seto, Hiroe, et al. "Gradient Boosting Decision Tree Becomes More Reliable than Logistic Regression in Predicting Probability for Diabetes with Big Data." *Scientific Reports*, vol. 12, no. 1, 2022, https://doi.org/10.1038/

DOI: https://doi.org/10.59720/23-062

s41598-022-20149-z.
20. Robinson, Andrew, and Jeff D. Hamann. *Forest Analytics with r: An Introduction*. Springer, 2011.
21. Molenberghs, Geert, and Geert Verbeke. *Models for Discrete Longitudinal Data: With 61 Figures*. Springer, 2005.