A Comprehensive Performance Evaluation of Parameterized Machine Learning Models for Diabetes Risk Prediction

Arighna Deb
School of Electronics Engineering,
KIIT University, Bhubaneswar, India
airghna.debfet@kiit.ac.in

January 21, 2022

1 Experimental Evaluation

The overall objective of the work is to evaluate the current potential of machine learning models for predicting the risk of having diabetes. The conducted experiments are twofold. In the first part, we compare the performance of a wide-variety of machine learning models for diabetes risk prediction and determine the best model to provide maximal accuracy. As we modify the parameters of the machine learning models, the models are further tested and compared with the existing works on diabetes risk prediction in the second part. In this section, we present the experimental set up, the obtained results and the discussion of those results.

1.1 Set up

We have considered six machine learning models that include logistic regression (LR), K-nearest neighbors (KNN), support vector machine (SVM), decision tree (DT), naive Bayes (NB) and random forest (RF) to predict the risk of diabetes at the early stage. We have changed the parameters of KNN, SVM, NB and RF models and evaluated their performance. More precisely, we have used two different values, 5 and 10 as the number of neighbors in the KNN model and three different values, 10, 20 and 100 as the number of trees in the RF model. We have taken four different types of kernel: *linear*, *radial basis function (RBF)*, *sigmoid*, *polynomial of degree 4* in the SVM model and four different types of naive Bayes classifier: *Gaussian*, *Bernoulli*, *Categorical*, *multinomial* in the NB machine learning technique. The names of all the machine learning models and their abbreviations are listed in Table 1 for better readability.

Table 1: Machine learning models, parameters and abbreviated names

ML model	Parameter name and value	Abbreviations
Logistic regression	-	LR
K-nearest neighbors	Number of neighbors = 5	KNN_5
K-nearest neighbors	Number of neighbors = 10	KNN_10
Random forest	Number of trees = 10	RF_10
Random forest	Number of trees = 20	RF_20
Random forest	Number of trees = 100	RF_100
Support vector machine	Kernel type = RBF	SVM
Support vector machine	Kernel type = Linear	LSVM
Support vector machine	Kernel type = Sigmoid	sigSVM
Support vector machine	Kernel type = Polynomial	polySVM
Naive Bayes	Classifier = Gaussian	GNB
Naive Bayes	Classifier = Bernoulli	BNB
Naive Bayes	Classifier = Categorical	CNB
Naive Bayes	Classifier = Multinomial	MNB

The diabetes risk prediction dataset is randomly split into the training and test samples with four different split ratio.The considered split ratios are 80:20 (80% training and 20% test samples), 75:25 (75% training and 25% test samples), 70:30 (70% training and 30% test samples) and 65:35 (65% training and 35% test samples). Further, we have applied the *k-fold cross-validation* on training samples, where we have used $k = 5$ and $k = 10$.

We have implemented all the machine learning models, splitting of dataset into train-test samples and k-fold cross-validation in Python 3.6 with the use of Scikit-learn package. All the simulations are conducted in the Google Colaboratory running in a computer having Intel(R) Core(TM) i5-6200U CPU 2.40 GHz and 8 GB RAM.

1.2 Comparisons among different machine learning models

The prediction accuracy obtained using all the machine learning models on test samples of diabetes risk prediction dataset has been reported in Table 2. We have listed the prediction accuracy for each machine learning model on the test set where the test set are 20%, 25%, 30% and 35% of the complete dataset. From Table 2, it can be observed that the highest prediction accuracy of 98.07% can be obtained using random forest with number of trees is set to either 20 or 100, when the 20% test samples are randomly considered. The random forest models, RF_10 and RF_100 allow us to achieve the highest test accuracy of 97.69% and 97.25% in cases of 25% and 35% test samples, respectively. When 30% test samples are randomly considered, the maximum test accuracy of 96.79% is obtained using SVM model. A negligible decrease in test accuracy is observed if the RF_20 and RF_100 models are applied on 30% test samples. Considering the accuracy over different test set size, we can conclude that the RF_100, i.e. the random forest model with number of trees being 100 is the most suitable machine learning model for diabetes risk prediction.

Table 2: Test accuracy of different ML models over different test set size

	Train-test split ratio			
ML model	80:20	75:25	70:30	65:25
RF_100	98.07%	96.15%	96.15%	97.25%
RF_20	98.07%	95.38%	96.15%	95.60%
RF_10	96.15%	97.69%	95.51%	94.50%
DT	96.15%	94.62%	96.15%	95.60%
GNB	88.46%	90.77%	89.74%	90.10%
CNB	81.73%	84.61%	86.53%	86.81%
BNB	81.73%	84.62%	86.53%	86.81%
MNB	92.31%	91.53%	91.67%	91.21%
SVM	94.23%	93.84%	92.94%	93.40%
LSVM	97.11%	95.38%	96.79%	94.50%
sigSVM	75.96%	79.23%	76.28%	75.82%
polySVM	91.34%	92.31%	92.95%	91.76%
KNN_5	91.34%	93.07%	91.67%	92.30%
KNN_10	90.38%	87.69%	91.02%	89.56%
LR	91.34%	92.30%	92.30%	91.75%

Moreover, the confusion matrix generated after applying each machine learning model on different test set size is shown in Table 3. It is important to note that the test result showing *false negative* suggests that a person has no chance to have diabetes in future when the reality is exactly opposite. The impact of false negative is more harmful than that of false positive since a false negative result puts a person at a higher risk of getting diabetes because the corresponding person may avoid any restriction which keeps him or her away from diabetes. It is a well-established fact that higher the number of false negative results, greater will be the possibility of having diabetes in future and lower will be the reliability of the model. Ideally, the number of false negative result should be zero. From Table 3, we can observe that the number of false negative is minimum in case of The random forest models, RF_10, RF_20, RF_100 over different test set size.

Depending on the confusion matrix, we estimate the *precision*, *recall* and *F1-score* of each machine learning model. Table 4 summarizes the detailed results. A machine learning model is expected to have precision, recall and F1-score as high as possible. The random forest models, RF_20 and RF_100 result in the maximum precision, recall and F1-score. In fact, using RF_20 and RF_100, we can achieve a perfect recall value of 1.0 alongside highest

Table 3: Confusion matrix of all the machine learning models for different test set size

	[TN,FP,FN,TP]			
ML model	20% test samples	25% test samples	30% test samples	35% test samples
RF_10	[36,2,2,64]	[43,2,1,84]	[55,3,4,94]	[67,2,8,105]
RF_20	[36,2,0,66]	[43,2,4,81]	[55,3,3,95]	[67,2,6,107]
RF_100	[36,2,0,66]	[43,2,3,82]	[55,3,3,95]	[67,2,6,107]
DT	[36,2,2,64]	[43,2,5,80]	[53,5,1,97]	[67,2,6,107]
polySVM	[37,1,8,58]	[44,1,9,76]	[57,1,10,88]	[69,0,15,98]
SVM	[35,3,3,63]	[41,4,4,81]	[52,6,5,93]	[63,6,6,107]
LSVM	[37,1,2,64]	[44,1,5,80]	[55,3,2,96]	[67,2,8,105]
sigSVM	[24,14,11,55]	[29,16,11,74]	[36,22,15,83]	[45,24,20,93]
GNB	[32,6,6,60]	[39,6,6,79]	[51,7,9,89]	[60,9,9,104]
CNB	[32,6,13,53]	[39,6,14,71]	[52,6,15,83]	[61,8,16,97]
BNB	[32,6,13,53]	[39,6,14,71]	[52,6,15,83]	[61,8,16,97]
MNB	[34,4,4,62]	[38,7,4,81]	[50,8,5,93]	[59,10,6,107]
KNN_5	[36,2,7,59]	[43,2,7,78]	[54,4,9,89]	[65,4,10,103]
KNN_10	[36,2,8,58]	[43,2,14,71]	[55,3,11,87]	[66,3,16,97]
LR	[35,3,6,60]	[42,3,7,78]	[54,4,8,90]	[64,5,10,103]

TN: True Negative, FP: False Positive, FN: False Negative, TP: True Positive

precision and F1-score.

Table 4: Precision, recall and F1-score of various ML models for different test set size

	20% test samples			25% test samples			30% test samples			35% test samples		
ML model	precision	recall	F1-score	precision	recall	F1-score	precision	recall	F1-score	precision	recall	F1-score
RF_10	0.97	0.97	0.97	0.98	0.99	0.98	0.97	0.96	0.96	0.98	0.93	0.95
RF_20	0.97	1.0	0.99	0.98	0.95	0.96	0.97	0.97	0.97	0.98	0.95	0.96
RF_100	0.97	1.0	0.99	0.98	0.96	0.97	0.96	1.0	0.98	0.97	0.99	0.98
DT	0.97	0.97	0.97	0.98	0.94	0.96	0.95	0.99	0.97	0.98	0.95	0.96
KNN_5	0.97	0.89	0.93	0.97	0.92	0.95	0.96	0.91	0.93	0.96	0.91	0.94
KNN_10	0.97	0.88	0.92	0.97	0.84	0.9	0.97	0.89	0.93	0.97	0.86	0.91
SVM	0.95	0.95	0.95	0.95	0.95	0.95	0.94	0.95	0.94	0.95	0.95	0.95
LSVM	0.98	0.97	0.98	0.99	0.94	0.96	0.97	0.98	0.97	0.98	0.93	0.95
sigSVM	0.8	0.83	0.81	0.82	0.87	0.85	0.79	0.85	0.82	0.79	0.82	0.81
polySVM	0.98	0.88	0.93	0.99	0.89	0.94	0.99	0.9	0.94	1.0	0.87	0.93
GNB	0.91	0.91	0.91	0.93	0.93	0.93	0.93	0.91	0.92	0.92	0.92	0.92
CNB	0.9	0.8	0.85	0.92	0.84	0.88	0.93	0.85	0.89	0.92	0.86	0.89
BNB	0.9	0.8	0.85	0.92	0.84	0.88	0.93	0.85	0.89	0.92	0.86	0.89
MNB	0.94	0.94	0.94	0.92	0.95	0.94	0.92	0.95	0.93	0.91	0.95	0.93
LR	0.95	0.91	0.93	0.96	0.92	0.94	0.96	0.92	0.94	0.95	0.91	0.93

The area under the receiver operating characteristics or AUROC is an important evaluation metrics for estimating the machine learning models' performance. The AUROC value lies in between 0 and 1. Higher the AUROC value, the better is the machine learning model at distinguishing between person having a risk of diabetes and no risk of diabetes. In terms of AUROC performance metric, the random forest models RF_20 and RF_100 clearly outperform all the remaining considered models over the different test set size as can be seen from Table 5.

Table 5: Area under the receiver operating characteristics (AUROC) of various ML models for different test set size

ML model	20% test samples	25% test samples	30% test samples	35% test samples
RF_10	0.96	0.97	0.95	0.95
RF_20	0.97	0.95	0.96	0.96
RF_100	0.97	0.96	0.97	0.97
DT	0.96	0.95	0.95	0.96
KNN_5	0.92	0.94	0.92	0.93
KNN_10	0.91	0.9	0.92	0.91
SVM	0.94	0.93	0.92	0.93
LSVM	0.97	0.96	0.96	0.95
sigSVM	0.73	0.76	0.73	0.74
polySVM	0.93	0.94	0.94	0.93
GNB	0.88	0.9	0.89	0.89
CNB	0.82	0.85	0.87	0.87
BNB	0.82	0.85	0.87	0.87
MNB	0.92	0.9	0.91	0.9
LR	0.92	0.93	0.92	0.92

Table 6 and Table 7 list the mean accuracy and standard deviation of all the machine learning models when 5-fold and 10-fold cross-validations, respectively, are applied on the training samples. From Table 6, we can observe that the random forest model RF_100 allows us to achieve the highest mean accuracy with a small percentage of standard deviation in case of 5-fold cross-validation over different training set size. However, when 10-fold cross-validation is applied on training samples, the random forest models, RF_10, RF_20 and RF_100 result in the desired mean accuracy with small percentage of standard deviation as shown in Table 7.

Table 6: Results of 5-fold cross-validation

	80% training samples		75% training samples		70% training samples		65% training samples	
ML model	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)
Rf_10	95.91	3.62	94.10	2.99	95.59	2.06	95.25	3.72
RF_20	95.67	4.08	94.62	3.84	94.21	4.17	94.94	4.50
RF_100	96.15	3.18	95.38	3.10	95.31	4.00	95.25	3.83
DT	94.22	2.80	93.85	3.18	93.12	2.94	92.30	3.46
KNN_5	92.30	2.25	91.54	2.24	92.30	2.25	91.40	3.85
KNN_10	90.62	2.80	89.49	3.28	89.83	2.96	90.23	3.58
SVM	93.27	4.07	91.28	3.18	92.01	3.70	90.23	3.62
LSVM	94.95	2.57	94.10	2.88	95.04	3.14	95.25	3.06
polySVM	92.78	2.76	93.59	2.69	93.40	2.69	92.88	2.59
sigSVM	77.89	1.94	77.44	1.31	78.57	2.28	78.69	4.56
GNB	88.45	4.16	87.18	5.50	88.45	3.70	88.45	4.16
CNB	88.45	3.31	88.46	4.44	86.81	2.72	86.38	4.54
BNB	88.45	3.31	88.46	4.44	86.81	2.72	86.38	4.54
MNB	89.18	2.97	88.72	3.18	90.10	3.76	88.75	3.58
LR	91.34	3.92	90.51	3.94	92.29	3.60	89.64	3.66

Considering all the performance metrics, the random forest model with the number of trees set to 100 i.e. RF_100 applied on the diabetes risk

Table 7: Results of 10-fold cross-validation

	80% training samples		75% training samples		70% training samples		65% training samples	
ML model	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)	mean acc. (%)	std. dev. (%)
Rf_10	96.86	2.65	95.38	2.51	96.42	1.77	95.85	3.31
RF_20	96.36	3.97	94.87	2.81	96.69	2.71	96.14	3.00
RF_100	96.37	2.93	95.64	2.58	96.69	2.71	96.74	2.46
DT	94.70	2.61	95.90	2.86	94.77	2.61	94.36	2.84
KNN_5	93	3.84	91.79	3.40	92.58	3.31	93.18	4.60
KNN_10	91.81	3.91	91.03	4.17	90.38	3.78	90.83	5.49
SVM	92.28	4.49	91.79	4.56	92.56	4.67	92.55	6.09
LSVM	94.70	2.37	95.13	3.13	95.88	2.81	95.24	3.34
polySVM	94.47	2.85	94.62	3.13	94.50	2.49	93.48	4.15
sigSVM	78.13	7.57	77.95	5.15	78.33	5.81	76.91	6.75
GNB	89.39	5.89	88.21	6.09	88.70	6.01	88.12	6.76
CNB	88.68	6.04	88.21	5.40	86.23	3.81	87.84	5.75
BNB	88.68	6.04	88.21	5.40	86.23	3.81	87.84	5.75
MNB	90.36	4.64	89.23	4.10	89.83	3.47	90.51	3.99
LR	92.53	5.29	91.79	5.71	91.18	5.25	91.67	6.32

prediction dataset is the most suitable machine learning model to classify and predict the risk of having diabetes at an early stage.

1.3 Comparisons with existing works

We have compared our results with the existing works [1–6] and reported the comparative results in Table 8. In [1], authors have reported the test accuracy of 73.82% using naive Bayes machine learning model. In [2], authors have used the random forest model and have achieved the test accuracy of only 76.3%. Authors in [3] have reported the test accuracy of 98% using logistic regression, while in [4], the test accuracy of 93% using the decision tree model is reported. In [5], the random forest model is used which results in the test accuracy of 94.02%, while the test accuracy of 97.40% using the random forest model is reported in [6].

Table 8: Comparisons of test accuracy with the existing works

ML model	Test accuracy
NB [1]	73.82%
RF [2]	76.3%
LR [3]	98%
DT [4]	93%
RF [5]	94.02%
RF [6]	97.40%
RF_100 [Proposed work]	98.07%

From the above discussion, we can infer that the random forest model is the most suitable model for predicting the risk of having diabetes at an early stage as evident from the recent works [5, 6] and the proposed work. The previous works have only reported the test accuracy and the machine learning model with default parameters over a single training-test split ratio. In contrast, in this work, we have provided a comprehensive evaluation of all the machine learning models by considering different training-test split ratio, different parameters of machine learning models, various performance metrics and cross-validation. Finally, we conclude that the random forest model with the number of trees set to 100 performs best on the diabetes risk prediction dataset.

References

[1] Selvakumar, S., Kannan, K.S., GothaiNachiyar, S.: Prediction of diabetes diagnosis using classification based data mining techniques. International Journal of Statistics and Systems **12**(2), 183–188 (2017)

[2] Sisodia, D., Sisodia, D.S.: Prediction of diabetes using classification algorithms. Procedia computer science **132**, 1578–1585 (2018)

[3] Dwivedi, A.K.: Analysis of computational intelligence techniques for diabetes mellitus prediction. Neural Computing and Applications **30**(12), 3837–3845 (2018)

[4] Sowjanya, K., Singhal, A., Choudhary, C.: Mobdbtest: A machine learning based system for predicting diabetes risk using mobile devices. In: 2015 IEEE International Advance Computing Conference (IACC), pp. 397–402 (2015). IEEE

[5] Ferdousi, R., Hossain, M.A., El Saddik, A.: Early-stage risk prediction of non-communicable disease using machine learning in health cps. IEEE Access **9**, 96823–96837 (2021)

[6] Hossain, M.A., Ferdousi, R., Alhamid, M.F.: Knowledge-driven machine learning based framework for early-stage disease risk prediction in edge environment. Journal of Parallel and Distributed Computing **146**, 25–34 (2020)