Report on the Investigation: Design and Application of a Machine Learning System for a Practical Problem

Rahul Kithalamane Basavaraj - 2212064
Word Count - 1202

May 3, 2023

1 Introduction

In this report we have used four different classification algorithms such as decision tree classifier, random forest classifier, K-nearest neighbors classifier and SVC classifier and also we have used five different regression algorithms such Linear regressor, random forest regressor, Bayesian ridge regressor, gradient boosting regressor and SGD regressor for predicting the target variables from the given data sets. To perform analysis using these algorithms, first we do pre processing of the data sets for getting better results in terms of accuracies, precision, and recall. Among the classification algorithms random forest performed the best and among the regressors gradient boosting performed the best when compared with relative strengths and weaknesses of all the methods applied on to a data set..

2 Libraries

Here in this study, we use some of the libraries for performing machine learning algorithms.

1) Sklearn
2) Pandas
3) Matplotlib
4) Seaborn
5) Numpy
6) ProfileReport

7) Linear Regression
8) Random Forest Regressor
9) Gradient Boosting Regressor
10) Bayesian Ridge
11) SGDRegressor
12) KNeighbors Classifier

3 Project Design and Architecture

This machine learning problem was done in a structured manner. Firstly, the data is collected from the source file and is sent for pre-processing the data into standard scalable forms, to perform better when the feature is selected. We then scale the data into a standardized format so that the model performs better. We then build various models and compare them along various metrics. We then select the model which is performing the best along these metrics measured. We deal the missing values with various imputation strategies while building the model. The model is then trained and evaluated using performance metrics. Further on fine tuning the model we send it to the prediction phase.

4 Experimental Setup

The data set contains 1000 records with 21 input features and we split the data into two sets of which 70% is for training and 30% is for testing. In the other data set it contains 1400 records with 35 input features. We have an original data set and a test data set; these data sets contain 500 missing values in the first data set and categorical features in the second data set. To overcome this, we pre-process this data by filling the missing values through imputation techniques, feature scaling and one-hot encoding where we convert categorical features to numeric. Here we split the data into two sets of which 80% is for training and 30% is for testing. To implement various machine learning algorithms on these data sets in python we use Scikit-learn library.

Here for this data sets we generate descriptive statistics which summarize the central tendency, dispersion and shape of a data set's distribution, excluding NaN values.

Also to check for skewness we try to visualize how well the distribution of data set is in each columns by means of a histogram so that we

	count	mean	std	min	25%	50%	75%	max
F1	1000.0	1.397000	0.500201	0.890000	0.890000	1.89000	1.890000	1.890000
F2	1000.0	-4.695798	2.699756	-14.976000	-6.144750	-3.82440	-2.590725	-1.774140
F3	1000.0	-1.653968	0.765717	-7.209000	-1.843675	-1.34840	-1.178988	-1.130002
F4	1000.0	11.573618	2.708288	8.706600	9.559425	10.72110	12.762750	22.176000
F5	1000.0	6.108717	1.739357	4.264440	4.806650	5.59810	6.879000	13.290000
F6	1000.0	5515.453754	1534.747192	-4924.080000	5108.820000	5482.58550	5917.620000	17287.920000
F7	1000.0	11128.128917	1587.054314	1686.600000	10513.050000	10770.13500	11235.600000	27822.600000
F8	1000.0	-5.058608	0.901204	-8.675000	-5.427000	-4.77725	-4.381825	-4.122990
F9	1000.0	-2143.218490	702.890861	-5873.260000	-2424.337500	-2281.57000	-2049.885000	2671.740000
F10	1000.0	6594.407898	1494.941683	-3367.200000	6432.700000	6971.70000	7225.106500	14678.800000
F11	1000.0	-3.740958	0.902777	-7.238000	-4.153500	-3.46225	-3.047450	-2.752120
F12	1000.0	2.520013	0.853456	1.621290	1.889000	2.24875	2.885000	6.073000
F13	1000.0	0.508000	0.500186	0.000000	0.000000	1.00000	1.000000	1.000000
F14	1000.0	-120989.462060	5064.689413	-279151.140000	-120901.470000	-120843.48000	-120787.035000	-109690.140000
F15	1000.0	-2.930605	0.606601	-5.964000	-3.265750	-2.79235	-2.446800	-2.170056
F16	1000.0	0.516000	0.499994	0.000000	0.000000	1.00000	1.000000	1.000000
F17	1000.0	-2592.387215	493.077434	-5874.120000	-2766.202500	-2700.06000	-2565.445000	587.880000
F18	1000.0	-16777.828266	2074.355688	-32323.100000	-17004.850000	-16364.70000	-15977.540000	-7663.100000
F19	1000.0	90.724599	20.571395	70.424656	76.804000	84.58400	96.325000	233.980000
F20	500.0	31.280200	2.311671	24.560000	29.680000	31.24000	32.830000	38.040000

Figure 1: Descriptive statistics

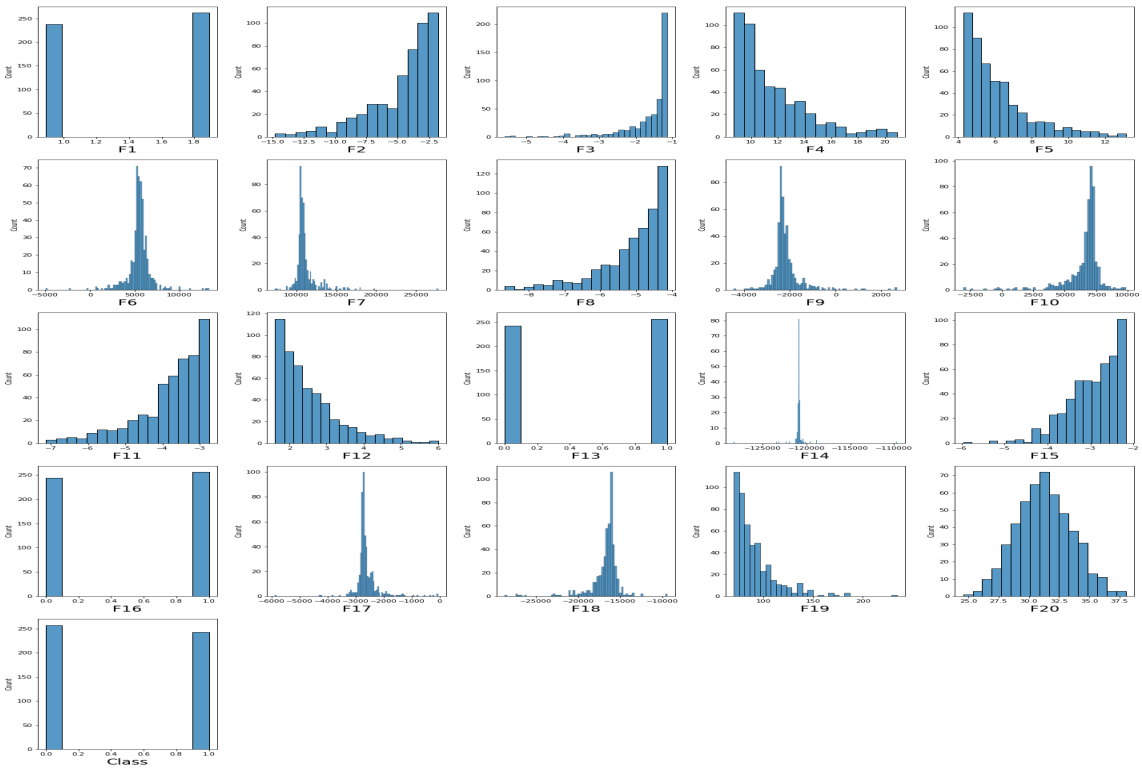


Figure 2: Distributions of data sets

can understand the frequency, density and how smooth the curve is for each column.

We then have correlation matrix which shows that there is strong relation between F20 and the target data set. Most of them seems to be having very less correlation.

Similarly, the we have visualized the descriptive statistics, distribution of data sets and correlation for the second experiment. When we take look at the descriptive statistical summary, we find that there are no missing values, hence we do not have to perform imputation. We checked for the skewness using histograms. The columns F1 and F21 contained categorical values, hence we encoded them using one-hot encoding and converted them to numeric values. The features have less correlation between them when we see the heat map.

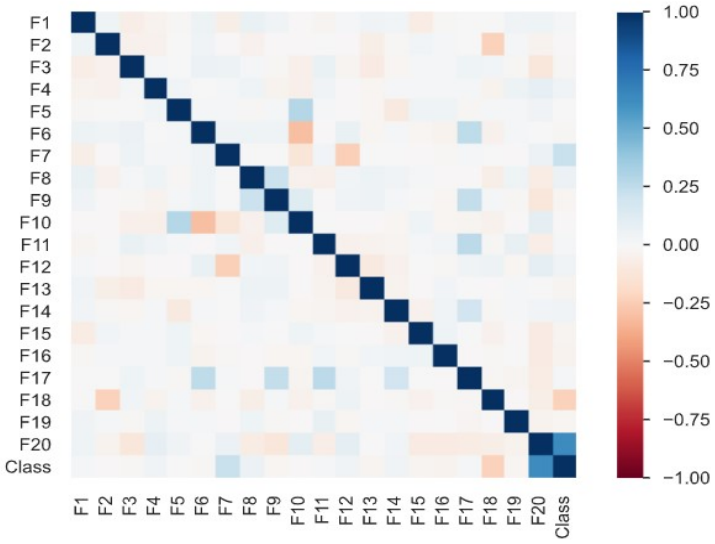


Figure 3: Correlation of columns

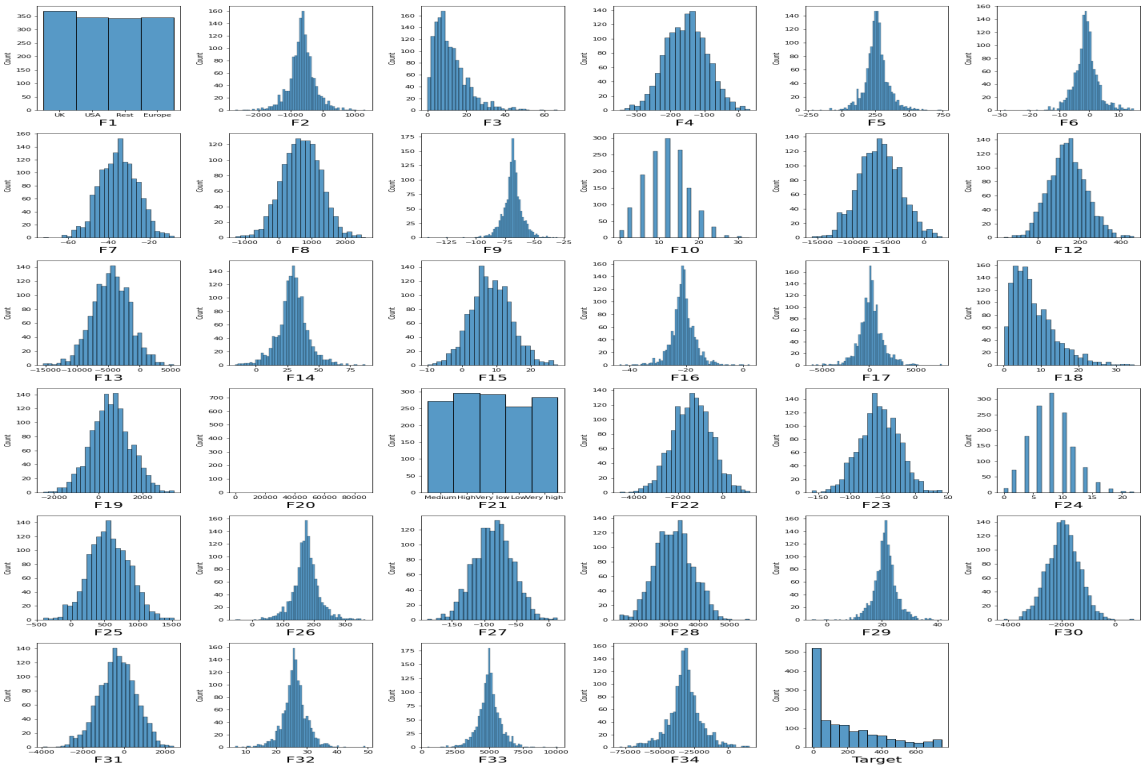


Figure 4: Distributions of data sets

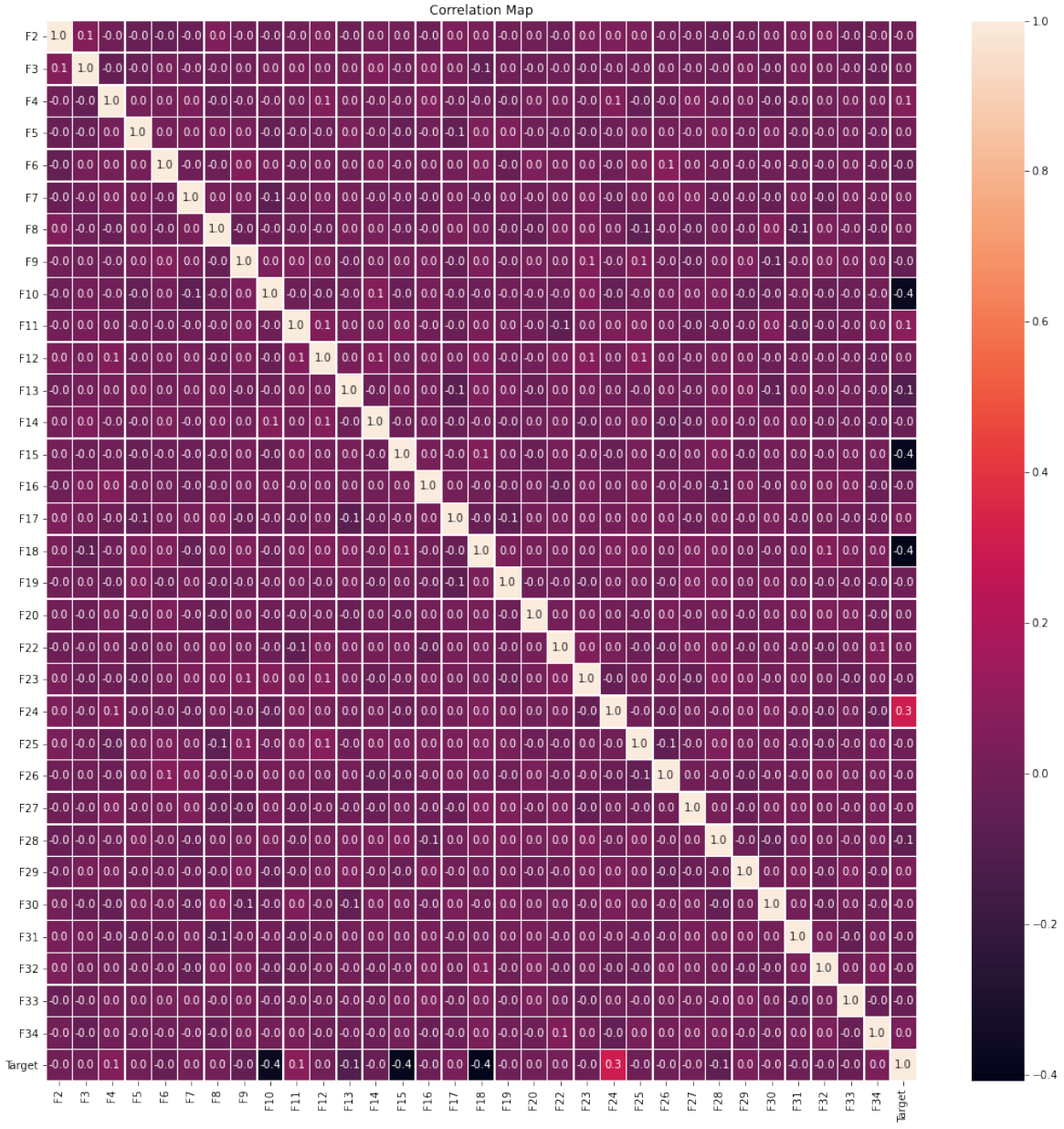
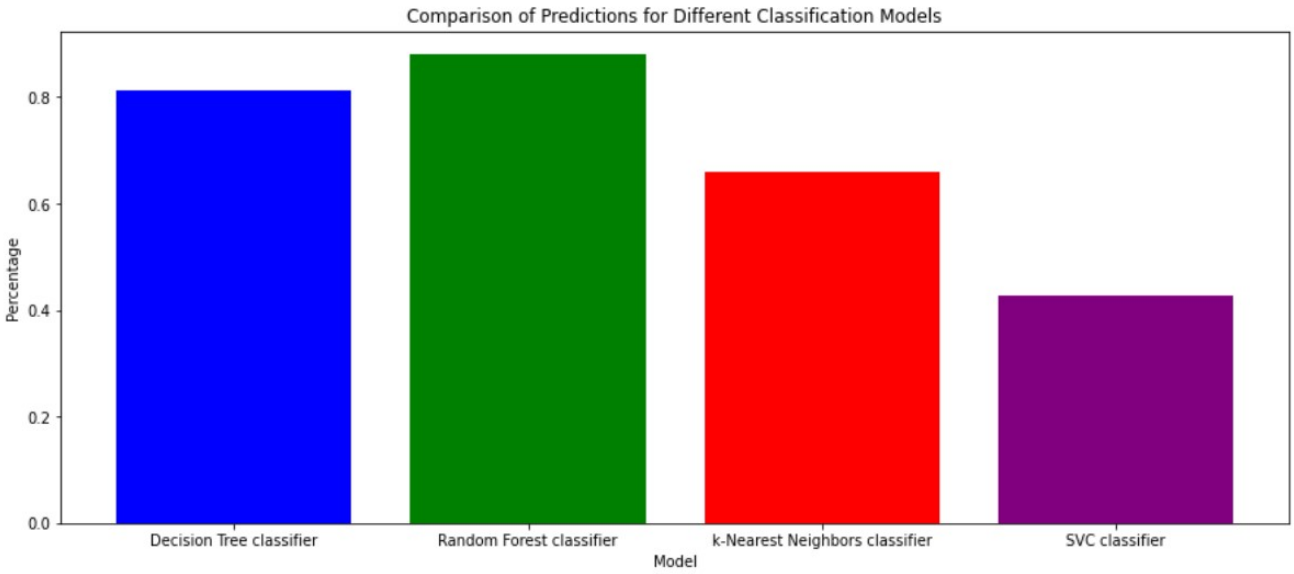


Figure 5: Correlation of columns



5 Part 2

5.1 Comparative study report

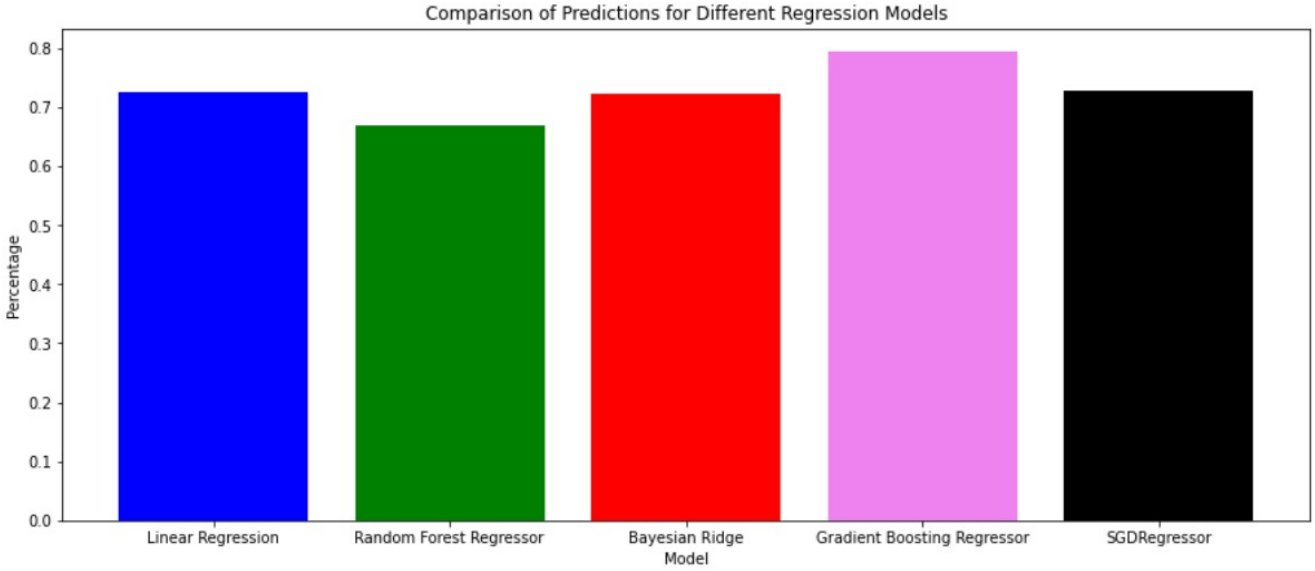
The classification algorithms such as decision tree classifier, random forest classifier, K-nearest neighbors classifier and SVC classifier were used for this study. We did not perform standardization because decision tree and random forest are not sensitive to input variables. We then analysed the performance of each model on the pre-processed data set using F1-score, which is the harmonic mean of precision and recall. The result shows that decision tree classification algorithm predicted 76% of the instances. Its precision, recall and F1 score are 0.7301, 0.7187, and 0.7244 respectively. Support vector classification has got the lowest accuracy compared to other algorithm however its recall of 1 indicates it identified all positive instances in the data set. The random forest classifier has got the highest accuracy of 0.84 which indicates it predicted 84% of the instances correctly in the data set. Its precision, recall and F1 scores are 0.8153, 0.8281, and 0.8217. The KNN classifier has got an accuracy of 0.66 which is lower than the accuracy of Random Forest but higher than the accuracy of decision tree algorithm. It precision, recall and F1 scores are 0.5970, 0.625 and 0.6106. Over all Random Forest algorithm appears to be the best performing model with respect to the evaluation metrics provided with accuracy, precision, recall and F1 score. Hence we used this algorithm for the prediction of target column in the data set.

	Accuracy	Precision	Recall	F1 Score
Decision Tree Classification	0.76	0.7301	0.7187	0.7244
Support vector Classification	0.42	0.4266	1	0.5981
Random Forest Classification	0.84	0.8153	0.8281	0.8217
KNN Classification	0.66	0.5970	0.625	0.6106

6 Part 3

6.1 Additional Comparative Study report

The regression algorithms such Linear regressor, random forest regressor, Bayesian ridge regressor, gradient boosting regressor and SGD regressor were used for this comparative study of data. For checking which model performs the best for the prediction we created a pipeline where all the data can be fed into them. We then analysed the performance of each model on pre processed data set. The results indicate the predictions made by each model and the average of the squared differences between predicted and actual values of the target variable. The linear regressor algorithm has an accuracy of 0.72453 and a Mean Square Error of 11623.99 which means that the prediction are kind of accurate. The random forest regressor has accuracy of 0.66988 which is lower than linear but with a high mean square error. It is not very accurate with predictions but there is a room for improvement. The Bayesian ridge regressor performs similar to linear regressor in terms of accuracy in prediction but there are some differences. The gradient boosting algorithm has the highest accuracy of 0.79259 which is more accurate compared to other algorithms. It has a mean square error of 8751.86 which is the lowest, which means it has lower errors in its predictions. The SGD regressor performs similar to linear regressor in terms of accuracy and prediction. Overall gradient boosting regressor appears to be the best performing model evaluating the metrics.



	Accuracy	Mean Square Error
Linear Regression	0.72453	11623.99
Random Forest Regression	0.66988	13930.09
Bayesian Ridge	0.72292	11692.00
Gradient Boosting Regressor	0.79259	8751.86
SGD Regressor	0.72625	11551.42

7 Conclusion

In this report we performed various predictions on various machine learning algorithms for both classification and regression. We conclude that with pre processing the accuracy in the data sets improves. Random forest classifier and gradient boosting regressor were the best performing models to identify whether the patient has eventually been diagnosed with diabetes or not and also to identify whether the person will develop diabetes, to how much the patient's average blood glucose level exceeds the diagnostic threshold. Therefore, these algorithms are recommended for further analysis and implementation by the health care providers for preventive measures.