Anjukutty Joseph

R00171244

Diabetics prediction model using machine learning

# **Abstract**

Diabetics a matter of serious concern at present since millions of people across different countries are having this disease. People with diabetics are unable to properly use and store glucose produced from food. Glucose backs up in the bloodstream - causing one’s blood glucose sugar to rise too high.[1]

This paper tries to create a machine learning model to predict whether or not a patient has diabetics or not based on several medical predictor variables. This research has a high-level impact in real world since lack  [physical activity](https://en.wikipedia.org/wiki/Physical_activity), poor [diet](https://en.wikipedia.org/wiki/Diet_(nutrition)), [stress](https://en.wikipedia.org/wiki/Stress_(biology)), and [urbanization](https://en.wikipedia.org/wiki/Urbanization) are considered to be the cause of diabetics in humans.

I analyzed real samples data from patients with Pima Indian heritage whose excess body fat is known to be associated with 100% of Pima Indians having diabetics. This being known, my objective is to create a machine learning model to accurately classify whether a given patient is having the disease or not. Five different classification models are proposed which are based on classification algorithms such as *Logistic regression, K-Nearest Neighbour, Naïve-Bayes, SVC, LSVC, Random Forest Classifier.* Among these, the best model is proved to be Logistic Regression with *F1 Score: 0.74*. Along with best model derivation, an attempt has been made to research on the effect of re-sampling techniques on accuracy of the model. The investigation on this area revealed that resampling techniques can improve the overall performance of the model.

1. **Introduction**

Early and timely detection of diabetics can create a big difference in handling this disease. Hence, if a machine learning model could help in diagnose the problem, it will be a big support for humanity.

Figure 1[2] illustrates an overall picture of the abundance of this disease across the world. This chart is created using data up to 2013 and numbers would have definitely gone up in 2019.

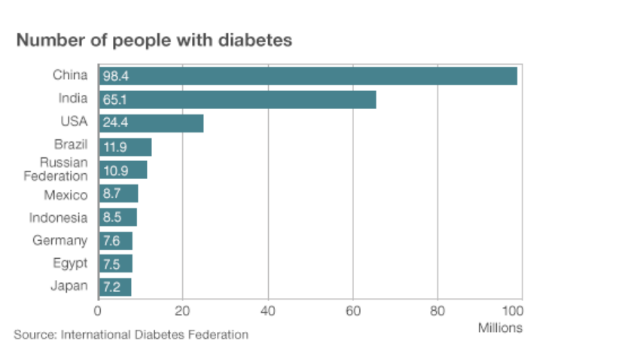


Figure :Number of diabetic patients across the world

The dataset chosen for analysis is Pima [3] Indian Diabetics dataset. This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage. The datasets consist of 8 medical predictor variables and one target variable- *Outcome.*[4]

Predictor variables includes the number of pregnancies the patient has had, their BMI, insulin level, age, blood pressure, skin thickness, Diabetes Pedigree Function. ­ This function provides some information on diabetes mellitus history in relatives and the genetic relationship of those relatives to the patient. There are 769 samples. *Outcome* variable 0 corresponds to no diabetics and 1 corresponds to diabetic patient. All other variables are continuous numerical values.

1. **Research**

In this study, I focus particularly on the effect of resampling techniques on best model’s prediction accuracy.

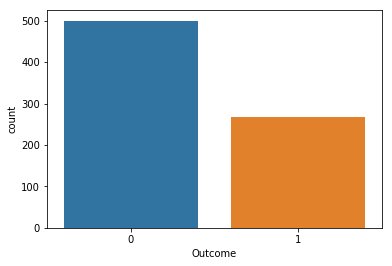


Figure :Distribution of target variable. 0-Non-Diabetic, 1- Diabetic

As shown in Figure 2, the number of samples for diabetic patients is only half of that of non-diabetic patients. To be precise, there are 501 non- diabetic patient records whereas only 269 diabetic patient records. This is a big imbalance in the dataset. A model trained on the given dataset cannot accurately predict diabetics in patients since it has seen more normal cases than abnormal predictor measurements and such model will predict people as non-diabetic and it can be mis leading. Another important thing to notice is that, the cost of false negatives is very high in health domain since we cannot start proper medication on time and it can eventually cause even the death of patient. So, it is very crucial that the machine learning model should be trained equally well on both positive and negative samples.

In order to address this issue, two main over sampling techniques are employed. Since the dataset consisted only 769 samples, trying to balance the dataset using under sampling techniques where we remove samples from majority class can cause loss of information. Therefore, over sampling methods are tried with the help of python imbalanced learn module.

In over sampling methodology, samples are added to minority class until both classes are balanced. Two popular over sampling techniques are compared in this study which are SMOTE (Synthetic Minority Oversampling Technique) and ADASYN (Adaptive Synthetic) along with Random Over Sampling. Random over sampling strategy generate new samples by randomly sampling with replacement of the current available samples. But, SMOTE and ADASYN techniques generate new samples by interpolation and they use same algorithm to achieve this.

Considering a sample x_i, a new sample x_{new} will be generated considering its k nearest-neighbours. Then, another nearest-neighbours x_{zi} is selected and a sample is generated as follows[5]:

x_{new} = x_i + \lambda \times (x_{zi} - x_i)

where \lambda is a random number in the range [0, 1]. This interpolation will create a sample on the line between x_{i} and x_{zi} as illustrated in the image below:

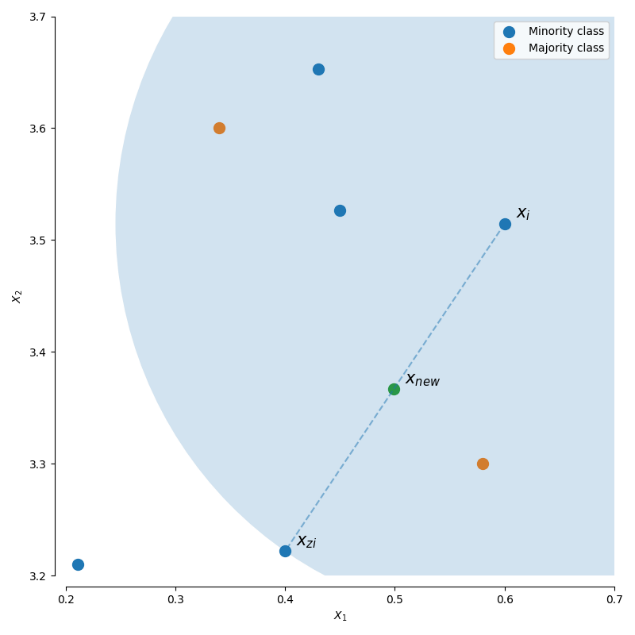


Figure : New sample generation using SMOTE and ADASYS

The chosen Pima diabetics dataset has features which are numerical in nature. Therefore, predicting best performing system was impossible unless applied on full data set. The selected 3 over sampling methods are applied on given dataset and the new data is used to train the best model with tuned parameters. The evaluation results are displayed in the following table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Technique** | **F1Score** | **Precision** | **Recall** |
| Random | 0.7432 | 0.7971 | 0.6962 |
| SMOTE | 0.7457 | 0.8029 | 0.6962 |
| ADASYN | 0.7132 | 0.7822 | 0.6554 |
| No resampling | 0.5882 | 0.6896 | 0.5128 |

Figure : Comparison of model accuracy using different over sampling technique

As shown in table, overall accuracy metrics are improved significantly when resampling is applied on dataset. The performance of the 3 balancing techniques were almost comparable. However, Random over sampling and SMOTE processes have slightly greater F1Score, precision and recall compared to ADASYN. Another noticeable point is the progress in Recall metric which is best comparison metric when there is a high cost associated with False Negative which in this context is applicable. This improvement in Recall tells that, the model is able to classify more actual diabetic patients (minority class) as diabetic when compared with baseline model which is trained on imbalanced data set. So, using resampling, the accuracy of classification for minority class is improved.

1. **Methodology**

The dataset is analysed to identify missing values, outliers and feature scaling.

**3.1 Missing Value**

All the features in data had 0 values. For some of the features, this measurement make sense for instance, number of pregnancies, diabetics pedigree function. Hence, the remaining features were filtered and analysed for the extent of missing values. Figure 5 depicts the extent of missingness in each feature.

Features- skin thickness and insulin level of patient is missing for a marginal number of samples. Other features have low level of missing data. As the dataset size is small, removal of missing feature samples could cause large volume of data loss. Considering this fact, imputation techniques are applied to replace measurements with 0 values with the mean of observations in particular feature.

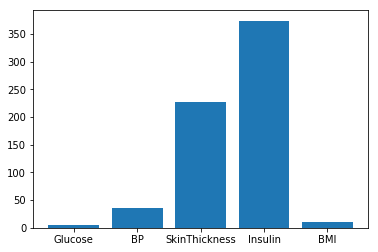
****

Figure : Extent of missing value in features

A detailed look at the data made it clear that, the average BP, insulin, glucose etc. varies for diabetic and non-diabetic patients.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Target | Glucose | BP | Skin | Insulin | BMI |
| Non-Diabetic | 109.9 | 68.2 | 19.6 | 68.8 | 30.3 |
| Diabetic | 141.3 | 70.8 | 22.2 | 100.3 | 35.1 |

For this reason, missing values in these 5 features are replaced with their average for diabetic and non-diabetic patients separately.

**3.2 Outlier Detection**

Figure 6 represents box plot showing the range of values in feature variables in dataset. It can be seen that, there are some outliers present in each feature but many of these are justifiable. Thus, no attempt is made in this study to remove outliers.

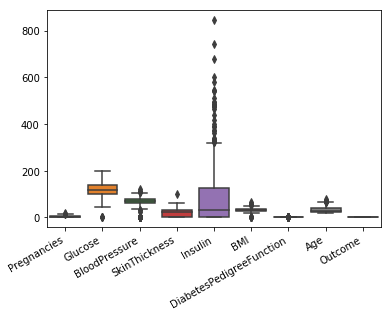
****

Figure :Box plot of predictor values

**3.3 Feature Scaling**

Machine learning algorithms behave better if features are on same scale. In chosen dataset, the feature values range differs a lot which is clear in Figure 6 box plot as well as in Figure 7 below. So, scaling data is essential to derive a highly accurate model.

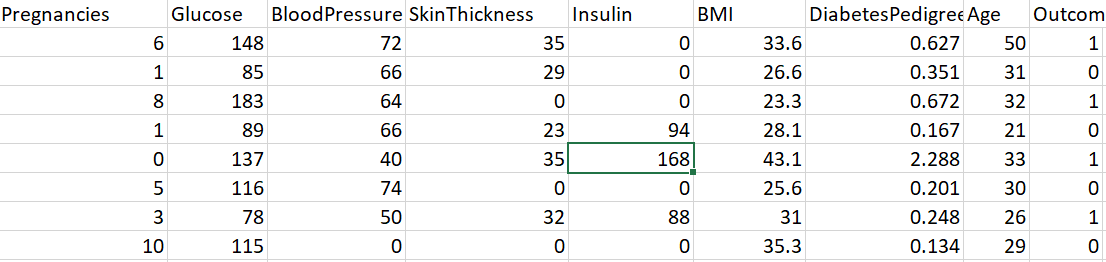


Figure : Dataset overview

Two common data transformation techniques- Normalisation and Standardisation are tried and compared in this study with the help of python- pre-processing- MinMaxScaler functions. Even though the two techniques gave comparable results on final model results, normalisation appear to have a slight edge over standardisation. So, data is transformed by applying normalisation as part of pre-processing.

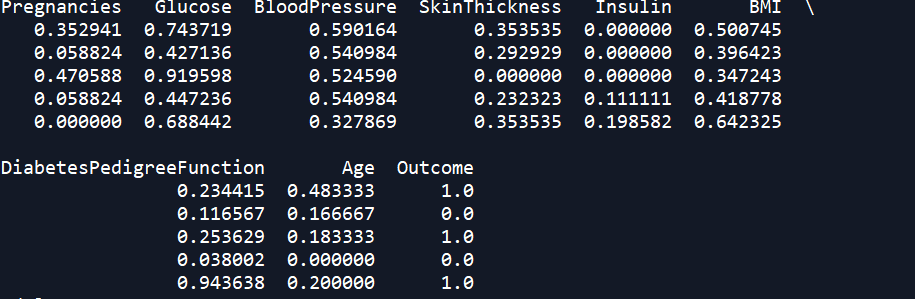


Figure : Data after normalisation

As shown in Figure 8, all features are re scaled into the range between 0 and 1.

1. **Experiments**

In this paper, 5 different models are proposed using 5 different algorithms. The dataset is divided into training, validation and test sets. 10-fold cross validation is performed to train and validate the model by adopting 70:30 ratio for training and test split of data.

**Logistic Regression**

This methodology to create a predictive model is well known when the problem is binary classification which in this case holds true. It is simple and powerful algorithm.

**k-nearest neighbours**

This algorithm is a non-parametric method used for classification. In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive [integer](https://en.wikipedia.org/wiki/Integer), typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbour[6]

**Naïve Bayes**

A Naive Bayes classifier is a probabilistic machine learning model that’s used for classification task[7]. The underlying principle of this algorithm is Bayes theorem using which we can find the probability of a patient being diabetic given that, another patient with same characteristic has got this disease or not.

P(A|B) = P(B|A) \* P(A)

P(B)

**Support Vector Classifier**

Support vector machine method is yet another effective classification algorithm when the classification needs to be done in high dimensional feature space. In the selected data set, there are 8 predictor variables and hence this classifier is expected to perform well in this problem.

**Random Forest Classifier**

This is an ensemble learning method which operates by constructing multiple decision tree at training time and aggregates the decision of all tress to decide the class of given input sample.

The accuracy score of 5 proposed models are compared in below Table

|  |  |  |
| --- | --- | --- |
| **Model Name** | **Accuracy** | **Standard Deviation** |
| LR | 0.7654 | 0.065 |
| KNN | 0.7319 | 0.054 |
| NB | 0.7485 | 0.042 |
| SVC | 0.7580 | 0.066 |
| RFC | 0.7577 | 0.047 |

Figure : Accuracy comparison of different models

When comparing the accuracy of all 5 proposed models, Linear regression and SVC gives the best prediction accuracy on validation dataset. Both are able to classify nearly 76% percent of patients as diabetic or not correctly during validation. As a result, I chose these two models for further analysis and tuning.

**Hyper Parameter Optimisation**

An algorithm does not give its best performance with default parameters. So fine tuning model parameters is essential. Parameter optimisation is attempted for SVC model as well as LR model. However, when the optimised SVC model is tested on re sampled testing data, the performance of logistic regression model was still a foot ahead than SVC model. So, the next section explains hyper parameter estimation of only logistic regression.

In order to perform tuning, python- grid search is used by providing parameter grid as input. Logistic regression accepts a range of input parameters like penalization norm, tolerance for stopping criteria, inverse regularization strength, class weight etc. [8]. Among these, penalization norm (penalty) and inverse regularization strength(C) are chosen for optimisation in this study.

Figure 10 below shows the attempted values for these two parameters.

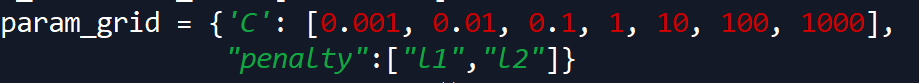


Figure :Parameter tuning

L1, L2 are regularization technique used to address over fitting and feature selection. L2 adds “*squared magnitude*” of coefficient as penalty term to the loss function whereas L1 adds “*absolute value of magnitude*” of coefficient as penalty term to the loss function. C (1/λ) is control variable for regularising the complexity of the model. If λ is small the complexity of model with be high. On the other hand, if λ is large, the model will be too simple and chance of underfitting will be high [9]

From the various values tried for these parameters, the best combination is proved to be C-1 and penalty- L1 in the best estimator.

1. **Evaluation**

From the study and research on resampling techniques, the best model after parameter tuning is found to be linear regression on validation set. Later, the model is tested on test set. Below table shows the accuracy metrics of model on test data which again contains resampled data.

|  |  |  |  |
| --- | --- | --- | --- |
| **Technique** | **F1Score** | **Precision** | **Recall** |
| Random | 0.7432 | 0.7971 | 0.6962 |
| SMOTE | 0.7457 | 0.8029 | 0.6962 |
| ADASYN | 0.7132 | 0.7822 | 0.6554 |
| No resampling | 0.5882 | 0.6896 | 0.5128 |

The precision of the best model is 0.8 which means the model could accurately classify 80% of new patients having diabetics in test data correctly. F1- Score of 0.75 is good so the model has less false positives and false negatives. As explained in section 2, improvement in Recall is a great achievement as the model false negative rate is reduced.

Confusion matrix of model when tested on test set before and after resampling is given below.

**Before**  **After**

[[135 18] [[115 27]

[ 38 40]] [ 48 110]]

It is evident that, the diabetic and non-diabetic samples well balanced in test set after resampling and prediction accuracy of minority class improved from 51% to 69%.

**Conclusion and Future Work**

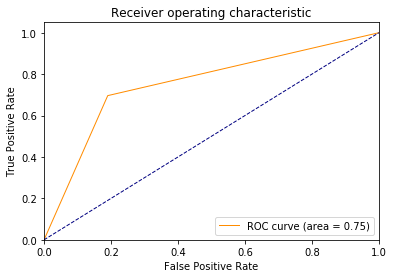


Figure :ROC Curve of best model[10]

Even though the final model is having an AUC of 0.75 as shown in Figure 11, the decision of the model needs to be more accurate because we are deciding whether a patient has disease or not. Incorrect classification of patient as diabetic or not diabetic has serious consequences. In future, I plan to train the model on bigger and diverse dataset to achieve better accuracy. Also, not all possible pre-processing techniques are used to prepare the data and I plan to study more in this regard.

**References**

[1] “Diabetes mellitus,” *Wikipedia*. 16-Apr-2019.

[2] “Alarm across Asia as diabetes figures leap,” *ucanews.com*. [Online]. Available: https://www.ucanews.com/news/alarm-across-asia-as-diabetes-figures-leap/69704. [Accessed: 17-Apr-2019].

[3] “Pima people,” *Wikipedia*. 25-Mar-2019.

[4] “Pima Indians Diabetes Database.” [Online]. Available: https://kaggle.com/uciml/pima-indians-diabetes-database. [Accessed: 17-Apr-2019].

[5] “2. Over-sampling — imbalanced-learn 0.4.3 documentation.” [Online]. Available: https://imbalanced-learn.readthedocs.io/en/stable/over\_sampling.html#smote-adasyn. [Accessed: 17-Apr-2019].

[6] “*k*-nearest neighbors algorithm,” *Wikipedia*. 09-Apr-2019.

[7] R. Gandhi, “Naive Bayes Classifier,” *Towards Data Science*, 05-May-2018. [Online]. Available: https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c. [Accessed: 18-Apr-2019].

[8] “2. Tuning parameters for logistic regression.” [Online]. Available: https://kaggle.com/joparga3/2-tuning-parameters-for-logistic-regression. [Accessed: 18-Apr-2019].

[9] A. Nagpal, “L1 and L2 Regularization Methods,” *Towards Data Science*, 13-Oct-2017. [Online]. Available: https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c. [Accessed: 18-Apr-2019].

[10] “Plot ROC Curve for Binary Classification with Matplotlib,” *Qiita*. [Online]. Available: https://qiita.com/bmj0114/items/460424c110a8ce22d945. [Accessed: 18-Apr-2019].

[11] www. [scikit-learn.org](https://scikit-learn.org/)