# Diabetes Classification

# Table of Contents

## Introduction

In recent decades, both the number of cases and the prevalence of diabetes have been steadily increasing in most developed and developing countries[1]. Diabetes is one of the largest global public health concerns which leads to a heavy global burden on public health as well as socio-economic development.

In this project, the main objective is to determine which features are the best indicators for predicting whether a patient has diabetes and to generate insights into each of these risk factors in our model based on certain diagnostic measurements included in the dataset. This study demonstrated that multivariate statistical techniques are effective for diabetes classification.

## Data and Data Pre-processing

In this study, the dataset was chosen from Kaggle, and it was originally from the National Institute of Diabetes and Digestive and Kidney Diseases. Before building the models, our group performed exploratory data analysis for the dataset. In this case, all patients are females at least 21 years old of Pima Indians heritage. The dataset consists of 8 medical predictor variables and 1 target variable[2]. Predictor variables include The Number of Pregnancies, BMI, Insulin Level, Age, Diabetes Pedigree Function, Skin Thickness, Blood Pressure, and Plasma Glucose Concentration.

Although there were no null values in this dataset, some of the features of the data had unreasonable zero values, such as glucose, blood pressure, and skin thickness. Those zeros in those features were treated as missing values. Each missing value was imputed by the median of each class. If the sample with the missing value belongs to the diabetes class, the missing value was imputed by the median of the diabetes class. The log transformation was adopted for the features: Insulin, Diabetes Pedigree Function, and Age because those characteristics have a skewness distribution.

This data is an imbalance dataset[3]and it includes 500 non-diabetic patients and 268 non-diabetic patients. An imbalanced dataset may cause an accuracy paradox, which means the model has high accuracy since the model has strong classification power for the majority class. However, the model cannot classify the minority class correctly. Therefore, the Synthetic Minority Oversampling Technique (SMOTE) (Yam, Fan, 2022) was adopted in this project. By the linear interpolation of two minority samples, SMOTE can generate a new minority observation (a synthetic sample).

[1] There were approximately 450 million people diagnosed with diabetes that resulted in around 1.37 million deaths globally in 2017. (Cho, Shaw, Karuranga, Huang, da Rocha Fernandes, Ohlrogge, Malanda, 2018)

[2] Binary Variable - Outcome: 0 = Non-diabetic patient; 1= Diabetic patient

[3] The minority class has significantly less than the majority class

The SMOTE runs as follows: Randomly pick a sample of all minority samples; denote as y, then perform K-nearest neighbor on y; denote as K-nearest neighbor as x(1), x(2), ... x(k), select one neighbor randomly; denote as x(j).

$$\text{Synthetic sample} = \mathrm{y}(1-\mathrm{p}) + \mathrm{p} * \mathrm{x}(\mathrm{j}) \qquad \mathrm{p} \sim \mathrm{U}[0,1]$$

Repeat the above process until sum of synthetic sample and minority sample equals to major sample.

Unity-based Normalization is one of the feature scaling methods in machine learning, min-max feature scaling can convert the actual range to data value to standard range of value i.e. [0,1]. Unity-based Normalization can avoid the number overflow (Yam, Fan, 2022) issue by ensuring the input data value in a standard range. The computer does not need to calculate very small and very large value.

## Methodology

### ● Logistic Regression

Logistic regression is used to define the relationship between such independent variables as $X_1, \ldots, X_{n,}$ and Y binary dependent variables which is coded as 0 or 1 for two possible categories. It is also used to predict the probability of a target binary variable. Our group would like to use it to predict whether the patient has diabetes. The logistic function is defined as follows: $logit\ (\pi_i) = \frac{\pi_i}{1-\pi_i}$ where $\pi_i$ is the probability of success. (Meyers, 2002)

### ● Decision Tree

Decision tree is a classical analysis method based on probabilities of various situations. The main usage of this method is to assess the risk of some dataset and judge its feasibility. In a decision tree, the algorithm starts from the root node, which represents the whole dataset of the tree for predicting the class of the dataset. The algorithm would compare the record attribute with the values of the root attribute. Then based on the comparison, follow decision trees flow and go to the next node. For the next node, the algorithm would compare the attribute value with the other sub-nodes and move further. The process will be continuous until the datum reaches the leaf node of the tree.

### ● Random Forest

Random forest classifier, which is a tree-based method proposed by Breiman, is a combination of tree classifiers that use the random selection of features at each node to split the samples. The classification procedure started by drawing the in-bag datasets with the bootstrap sampling method and building plenty of trees for each in-bag dataset. Features selection and splitting procedures were repeated recursively until the node reached the minimum sample size. The out-of-bag dataset is used to test for each tree, average the misclassification error, and finally select the majority vote.

Implementing the appropriate randomness could increase classification accuracy, release the sensitivity to noise, and minimize the correlation among features (Breiman, 2001). Therefore, it is important to tune some hyperparameters. Due to the Law of Large Numbers, the random forest classifier does not overfit, hence there are 6 hyperparameters tuned in a wide range. Three significant hyperparameters are the number of trees, the maximum depth of tree, and the minimum sample required to split node which are tuned to be 450 trees, 100 layers, and 2 samples respectively.

Random forest could also give an internal estimate of permutation features' importance which is critical to investigate the interaction of features. Regarding the randomness of measurement, we implemented the random forest in 10 repetitions using different combinations of features in out-of-bag samples. A significant fall in the accuracy score under the permutation of the values of the feature indicated that this feature carried significant predictive information.

## • Support Vector Machine (SVM)

Support Vector Machine (SVM) is a classification algorithm. The objective of the Support Vector Machine is to create an N-dimensional hyperplane (N is the number of features) to separate the two classes of samples. If N = 2, the hyperplane will be a line. (Gandhi,2018) If N = 3, the hyperplane will be a two-dimensional plane. To achieve the best performance on classification, the hyperplane is adjusted by support vectors in the Support Vector Machine Algorithm. Support vectors are data points whose hyperplane is close to the hyperplane. The distance between support vectors and the hyperplane is called the margin. The Support Vector Machine Algorithm will maximize the margin so that it gets the best classification performance.

There are 3 main hyperparameters of SVM (Liu,2020): kernels, C, and Gamma. Transform the input data into the required form for handling the data by different mathematics functions. C: The penalty parameter tells the algorithm how much error is acceptable. Lower misclassification data points for larger C, but more complex decision boundary. Smaller C results in more misclassification of data points but a simpler decision boundary. Gamma: how many support vectors are considered when adjusting the hyperplane?

## • Extreme Gradient Boosting (XGB)

Boosting is an ensemble algorithm, that combines a set of weak learners and integrates them to form a strong learner. The procedure started by building the weak model from the training data with equal weight. After fitting the data, the model is used to identify the misclassified data and increase their weight. Therefore, these data points will be more likely to be corrected in the next fitting of the model. By recursively constructing models, the optimal model is obtained until the training data is correctly predicted, or the maximum number of models is reached. Extreme gradient boosting is based on the gradient boosting algorithm. By comparing with other boosting models, gradient boosting models assign different weights to the incorrect data points, which are based on minimizing the residual errors calculated by a loss function. Especially, XGB is applied an objective function, which is the combination of the loss function and regularization parameter. It can push the limit of computation resources since the regularization parameter controls the complexity of the model to avoid overfitting. (Chen, Guestrin, 2016)

In this project, there were several major hyperparameters. The maximum depth of the base tree model is important, which controls the model complexity and problems of overfitting. In boosting models, it is rarely set higher than 10, because the motivation of ensemble models is to combine weak models to form a strong model, instead of building several strong models directly. Another main hyperparameter is the learning rate of the regularization parameter, which controls the feature weights.

- ### K-Nearest Neighbors (KNN) Algorithm

KNN is a commonly used clustering method. Under the assumption, similar inputs lead to similar outputs, it aims to classify data into groups by similarity. (Ryan, 2022) By determining the distance method, the algorithm was performed under different values of k, the optimal k we found was 1 for this dataset. (Shruti, 2020) The test score is gradually decreasing from 0.855 (when k=1, too extreme that may be overfitted) to 0.835 (k=3) and then 0.805 (k=5). The scoring fluctuated around 0.80 afterward. It reflects the accuracy of correct detection at around 80% approximately.

- ### Hyperparameter Tuning

In order to optimize the performance of the above model, Random Search was adopted for finding the best combination of hyperparameters. In random search, hyperparameters are provided in a feasible range and different combinations of hyperparameters are randomly sampled to get a set of hyperparameters that has the best performance. F1 score was adopted to evaluate the performance of the 10-cross-validated model since disease classifier should not miss any potential positive observations.

## Results & Analysis

To analyze the classification algorithm, several criteria were used to evaluate the result:

| | SVM | XGB | RF | DT | LR | KNN |
|---|---|---|---|---|---|---|
| Recall | 0.825 | 0.866 | 0.907 | 0.784 | 0.773 | 0.897 |
| Precision | 0.792 | 0.857 | 0.88 | 0.817 | 0.773 | 0.77 |
| Accuracy | 0.81 | 0.865 | 0.895 | 0.81 | 0.78 | 0.82 |

| | | | | | | |
|---|---|---|---|---|---|---|
| F1 score | 0.808 | 0.862 | 0.893 | 0.8 | 0.773 | 0.829 |
| Auc | 0.81 | 0.865 | 0.895 | 0.809 | 0.78 | 0.822 |

*Table 1: Evaluation on Different Models*

As for the error analysis from the test data, it was observed that random forest was the best model that had the highest score in all the criteria with a large leading, especially with 89.5% accuracy and 89.3% F1 score. The second-best model was XGB, apart from recall, it obtained the second-highest score in all other aspects, with 86.5% accuracy and 86.2% F1 score.

## Feature importance

The feature importance describes the underlying impact of variables on diabetes prediction. Four of our models, which are LR, RF, DT, and XGB, were used to find out the feature importance. All these models suggested that "insulin level" carried the most significant impact with a large lead. As for the model of random forest, since the order of random selection of variables may affect the results of feature importance, permutation is applied to ensure our results (Breiman, 2001). According to Appendix 3.1, the 10 repeats of permutation gave us similar results that "insulin level" had the determinant effect on prediction.

## Conclusion and Further Work

In this study, our group has made reasonable predictions on whether a female patient has diabetes. Our best two classification models are random forest and XGB, which are both ensemble models. Besides, all models, that able to find out the feature importance, suggested that "insulin level" has the determinant influence on prediction.

Since imbalanced data exist popularly in reality, the way to conduct resampling (Jason, 2022) matters much in the result. First, the resampling method could be applied after data splitting to avoid the shape being affected. Besides, different resampling methods can be applied to test which makes a better result, such as under-sampling or combined resampling. Apart from resampling methods, we can try different methods to perform better data cleaning. For those non-reasonable values, we can try to apply KNN imputation or mean imputation to investigate whether they could enhance the accuracy of the model instead of implementing median imputation for further improvement.

Lastly, an ensemble machine learning (ML) algorithm by stacking two or more ML algorithms could be a new technique to do improvement. To conclude, we hope that our models can be applied to make a reasonable prediction of diabetes so that it could be used to curb the incidence of diabetes and reduce associated costs.

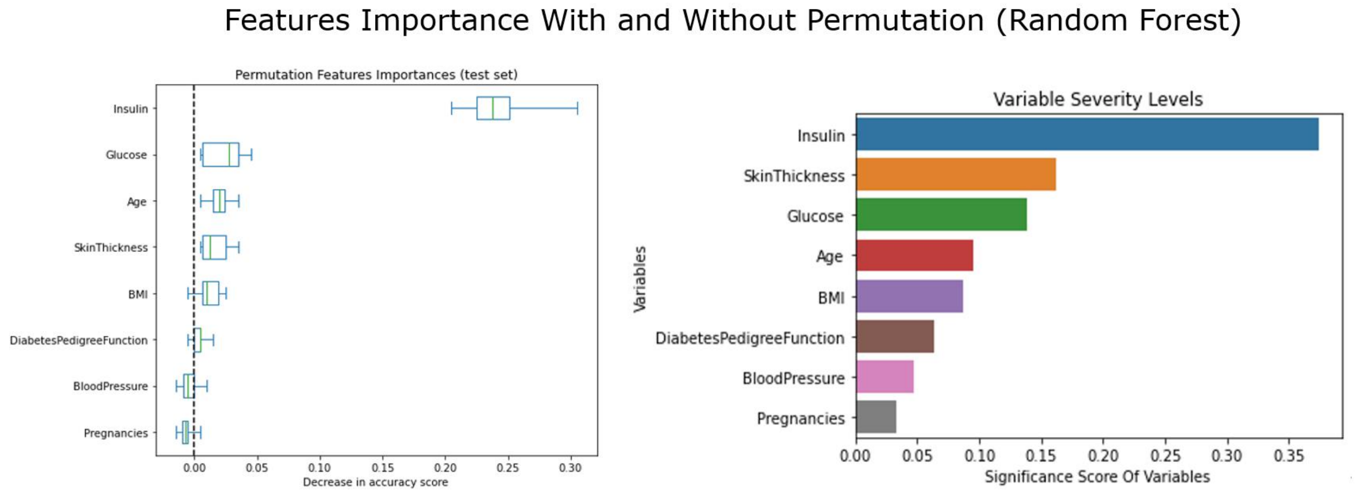
## Appendix

| | *Before Data Pre-processing* | | *After Data Pre-processing* | |
|---|---|---|---|---|
| ***(i)Dataset*** | **Imbalance Dataset:**<br>1: 268 diabetic patients<br>0: 500 non-diabetic patients<br>Name: Outcome | | **Balance Dataset:**<br>1: 500 diabetic patients<br>0: 500 non-diabetic patients<br>Name: Outcome | |
| ***(ii)Outliers*** | **Input Variables** | **No. of Outliers** | **Input Variables** | **No. of Outliers** |
| | • Pregnancies | 4 | • Pregnancies | 4 |
| | • Glucose | 5 | • Glucose | 0 |
| | • Blood Pressure | 35 | • Blood Pressure | 8 |
| | • Skin Thickness | 1 | • Skin Thickness | 4 |
| | • Insulin | 18 | • Insulin | 13 |
| | • BMI | 14 | • BMI | 5 |
| | • Diabetes Pedigree Function | 11 | • Diabetes Pedigree Function | 0 |
| | • Age | 5 | • Age | 0 |
| ***(iii) Correlation of Input variables and Output Variable*** | **Input Variables ($x_i$)** | **Correlation** | **Input Variables** | **Correlation** |
| | • Pregnancies | 0.22 | • Pregnancies | 0.22 |
| | • Glucose | 0.47 | • Glucose | 0.50 |
| | • Blood Pressure | 0.07 | • Blood Pressure | 0.17 |
| | • Skin Thickness | 0.07 | • Skin Thickness | 0.30 |
| | • Insulin | 0.13 | • Insulin | 0.46 |
| | • BMI | 0.29 | • BMI | 0.32 |
| | • Diabetes Pedigree Function | 0.17 | • Diabetes Pedigree Function | 0.18 |
| | • Age | 0.24 | • Age | 0.28 |

*Appendix 1: Exploratory Data Analysis Before and After Data Pre-processing*

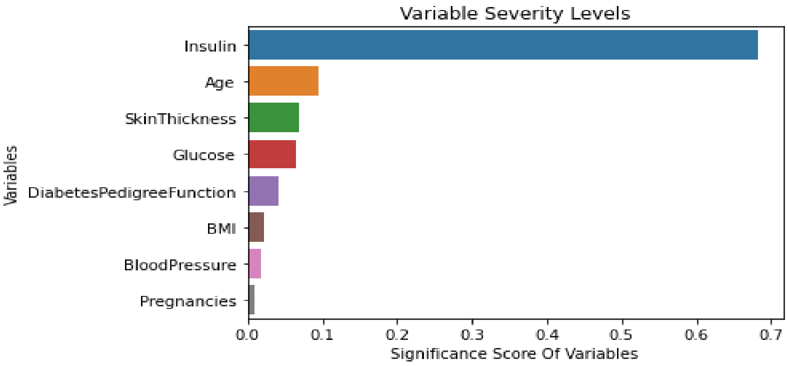
| ***Evaluation Methods*** | ***Definition*** |
|---|---|
| Recall | The ratio of correct positive to total number of positive samples |
| Precision | The ratio of correct positive to total number of all positive prediction |
| Accuracy | The ratio of correct classification to total number of samples |
| F1 score | The Harmonic mean of Recall and Precision |
| Auc | Area Under Receiver Operating Characteristic<br>(Curves draw by true positive rate and false positive rate) |

*Appendix 2: Definition of Different Evaluation Methods*



*Appendix 3.1:*

*Features Importance plots with and without Permutation (random forest)*



*Appendix 3.2:*

*Features Importance (XGB)*

*Appendix 3: Plots of Features Importance of Different Machine Learning Algorithm*

# Reference

1. Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). Retrieved from https://doi.org/10.1023/A:1010933404324

2. Clare Liu (2020). SVM Hyperparameter Tuning using GridSearchCV. Retrieved from https://www.vebuso.com/2020/03/svm-hyperparameter-tuning-using-gridsearchcv/

3. Chen, T. Q. and Guestrin, C. (2016) XGBoost: A Scalable Tree Boosting System. arXiv:1603.02754v3.

4. Cho N., Shaw J., Karuranga S., Huang Y., da Rocha Fernandes J., Ohlrogge A., Malanda B. IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045. Diabetes Res. Clin. Pract. 2018;138:271–281. doi: 10.1016/j.diabres.2018.02.023.

5. Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction. 2nd ed. New York, Springer.

6. Jason Brownlee (2021). *Smote for imbalanced classification with python*. Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

7. Kaggle. (2016). Pima Indians Diabetes Database. India: National Institute of Diabetes; Digestive; Kidney Diseases. https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

8. Luay Fraiwan, Khaldon Lweesy, Natheer Khasawneh, Heinrich Wenz, Hartmut Dickhaus, Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier, Computer Methods and Programs in Biomedicine, Volume 108, Issue 1,2012, Pages 10-19, ISSN 0169-2607, https://doi.org/10.1016/j.cmpb.2011.11.005

9. Luo (2022). a complete guide to K-nearest neighbours. Kaggle. Retrieved from https://www.kaggle.com/code/ryanluoli2/a-complete-guide-to-k-nearest-neighbours/notebook

10. Meyers, R. A. (2002). Encyclopedia of Physical Science and Technology. Academic Press.

11. Philip. Yam & Kaiser. Fan (2022) lecture note of STAT 4012 Statistical Principles of Deep Learning with Business Applications

12. Rohith Gandhi. (2018). Support Vector Machine — Introduction to Machine Learning Algorithms. Retrieved from https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca4

13. Shrutimechlearn. (2020). *Step by step diabetes classification-KNN-detailed*. Kaggle. Retrieved from https://www.kaggle.com/code/shrutimechlearn/step-by-step-diabetes-classification-knn-detailed