Classification of Diabetes and Meal Recommendation

Prof. Vijay Jumb*, Chinmay Patil[1], Manish Yadav[2], Rutuja Tarale[3]

**Assistant Professor, Department of Computer Engineering, Xavier Institute of Engineering, Mumbai, Maharashtra, India*
[1,2,3]*B.E student, Computer Engineering, Xavier Institute of Engineering, Mumbai, Maharashtra, India*

---***---

Abstract - Diabetes is the most common endocrine disorder. Statistics show that in every family there is someone who has this type of problem. The number of people suffering from this disease is increasing and there needs to be a system that tells the diet of the patients which will help them to make in control their sugar level in the body. This system is about the algorithm which takes various input from the user and gives them the desired output. This application has a various algorithm which will help to save time and also prove more efficient rather than directly going to the dietician. The users can get the diet according to their respective age group, height, weight, and type of diabetes they have. Thus, a Diet Recommendation system for diabetic patients can be used to give the patients their diet at any moment of time and thus saving their time. There are two parts in the project first part will be classification and detection and the second part will be meals recommendation. For classification, we are using various algorithms like a decision tree.

***KeyWords*: Fats, Sodium, Carbohydrates, Protein, Glycemic index.**

1. INTRODUCTION

Many people in real life suffer from diabetes in the early stage of life and in their busy schedule they don't have enough time and money to go to the doctor or any specialized dietician. We are building a system that will give them a specialized diet at its figure tips. Various machine learning algorithms are used for the same. For classification, we are using various algorithms like decision tree, KNN, logistic regression, support vector machine, gradient boost, random forest. On the basis of this best models are taken and the level of diabetes is displayed. For the second part, a different dataset is taken the one that we made where considering calories, fats, proteins, sodium, carbohydrates, and glycemic index prediction is done whether that a particular food item having so and so those six things are healthy or harmful for a diabetic patient.

1.1 Flow of the Project

1) Input and Output. 2) Analysis of Parameters.3) Recommendation through different Algorithms. 4) Selecting the Best Algorithm and Recommendation its output.

Information of Output is for the clarification of each factor which is introduced in the datasets. Investigation of Parameters is for comprehension and clarification of leading examination process overall dataset's sections. Choosing the factors which really show connection, covariance or reliance over the objective factors. At that point, we could process through different calculations and have nearly studied them. Thus, the algorithm which gives a better exact outcome is chosen as the yield. gives a better exact outcome is chosen as the yield.

2. Input and Output

Information such as the age, glucose, blood pressure, sick, blood sugar is collected and the diet according to that is shown such as the required amount of protein, carbohydrates, fats, GI is given as the output. So, there are two parts in the project first part will be classification and detection and the second part will be meals recommendation. For classification, we are using various algorithms like decision trees.

1) fats – an important parameter for maintaining fats in the body.

2) GI – calculate the GI number

3) Sodium – an important factor for blood pressure.

4) Carbs – it is important to factor for regulations control of carbs.

diabetes.csv - This file contains various parameters such as pregnancies, glucose, blood pressure, insulin, BMI, age, outcome.

dia_class_db.csv - This file contains additional data related to the calories, protein, fat, sodium, GI, carbohydrates

3. Analysis of Parameters

For the analysis of various parameters. Need for Understanding of various attributes present in every table. Also need to the analysis of the dependency of those parameters to the target variable i.e., Weekly show of blood sugar level.

Now Coming toward the variables checking. There are various things calculated in the first database such as glucose, insulin pregnancies,outcome.

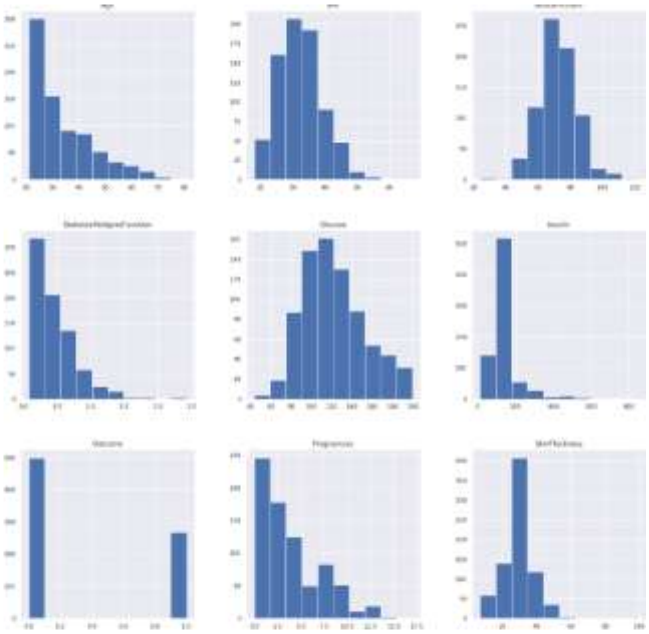


Fig -1: The clean data obtained.

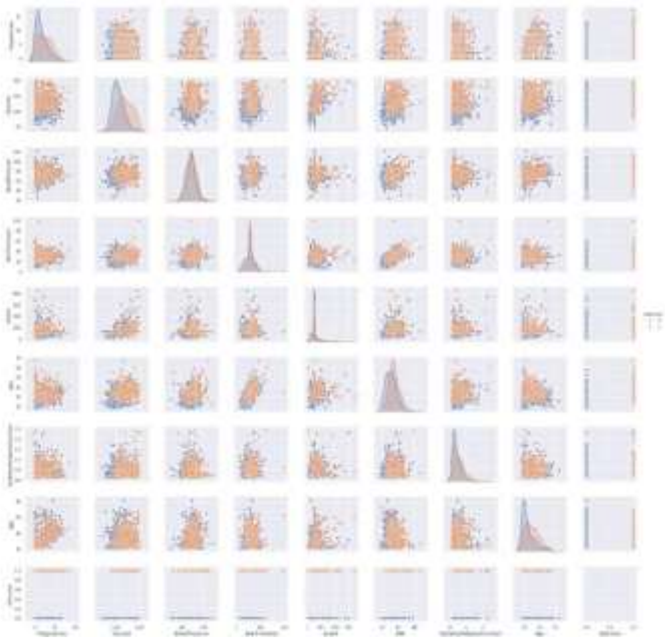
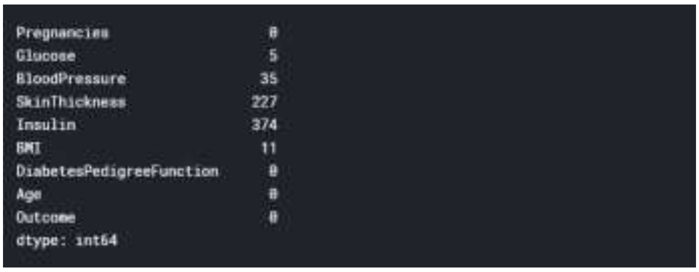


Fig -3: result after compiling



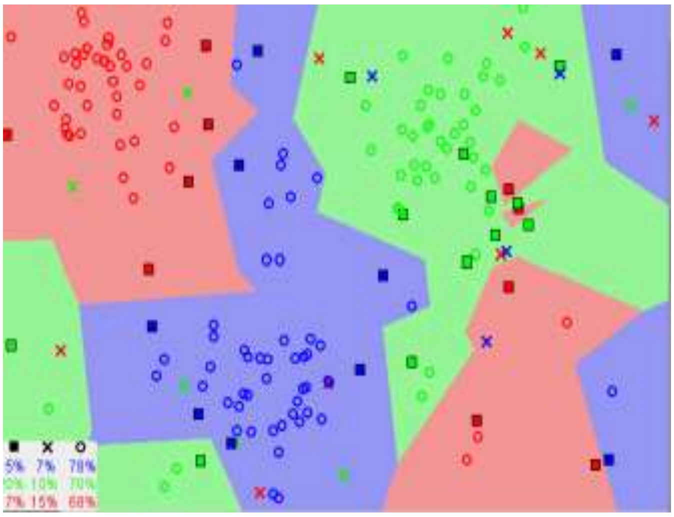
```
Pregnancies                 0
Glucose                     5
BloodPressure              35
SkinThickness             227
Insulin                   374
BMI                        11
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

Fig -4: VIF values of Variables

Pearson's Correlation Coefficient: helps you to trace the relationship between two quantities. It gives you the measure of the strength of the association between two variables. The value of Pearson's Correlation Coefficient can range from -1 to +1. 1 which means that they are highly correlated and 0 means no correlation can happen between them.

4.1 k-Nearest Neighbors:

The k-NN algorithm is the simplest machine learning algorithm. Creating the model consists of storing the training dataset and to make a prediction for new data, The algorithm finds the closest data points in its "nearest neighbors."



4.2 Support Vector Machine

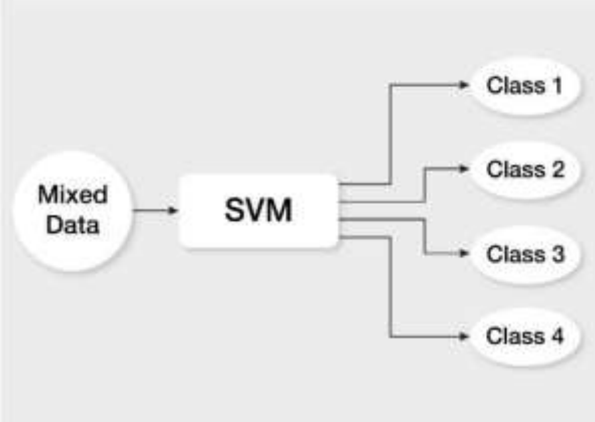
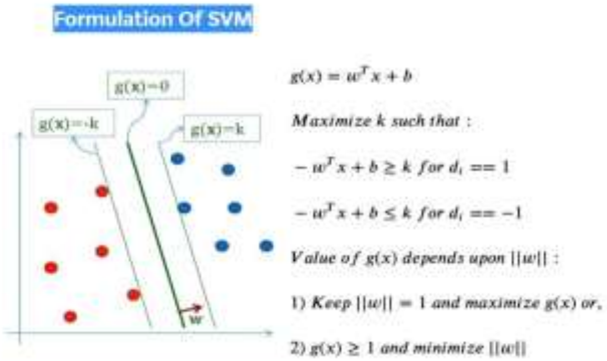


Fig -7: SVM Diagram



$g(x) = w^T x + b$

Maximize k such that :

$-w^T x + b \geq k \text{ for } d_i == 1$

$-w^T x + b \leq k \text{ for } d_i == -1$

Value of $g(x)$ *depends upon* $||w||$:

1) *Keep* $||w|| = 1$ *and maximize* $g(x)$ *or,*

2) $g(x) \geq 1$ *and minimize* $||w||$

Fig -8: Formulation in SVM

4.3 Infinite Dimensions

Theoretically data set would be linearly separable if mapped to infinite dimension hyperplane. Hence, if we can find a kernel that would give a product of infinite hyperplane mapping our job is done.

Here comes Mercer's theorem, it states that iff K(X, Y) is symmetric, continuous and positive semi-definite(Mercer's condition then), it can be represented as

$$K(X,Y) = \sum_{i=1}^{\infty} \lambda_i \, \phi_i(x) \cdot \phi_i(y) \; \forall \, \lambda_i > 0$$

Fig -9: Formula

4.4 Logistic Regression:

Logistic Regression is the most widely used Machine Learning algorithm for binary classification. It is the simplest Algorithm that you can use as a performance baseline, it is easy to implement. The building block concepts of Logistic Regression is also used in deep learning.

Fig -11: Formula

$$g(E(y)) = \alpha + \beta x1 + \gamma x2$$

Here, g() is link function, E(y) is the expectation of target variable and α + βx1 + γx2 is the linear predictor (α,β,γ to be predicted).
Important Points

1. GLM does not assume a linear relationship between dependent and independent variables. However, it assumes a linear relationship between link function and independent variables in the logit model.
2. The dependent variable need not to be normally distributed.
3. It does not use Ordinary Least Square for parameter estimation. Instead, it uses maximum likelihood estimation (MLE).
4. Errors need to be independent but not normally distributed.

Let's understand it further using an example:
We are provided a sample of 1000 customers. We need to predict the probability of whether a customer will buy (y) a particular magazine or not. As you can see, we have a categorical outcome variable, we'll use logistic regression. We will first write the simple linear regression equation with dependent variable enclosed in a link function:

g(y) = βo + β(Ag) ------------- (a)

Note: For ease of understanding, I've considered 'Ag' as an independent variable.

In logistic regression, the probability of outcome dependent variable (success or failure). As described above, g() is the link function. This function is obtained using two things: Probability of Success(p) and Probability of Failure(1-p). p should meet the following criteria:

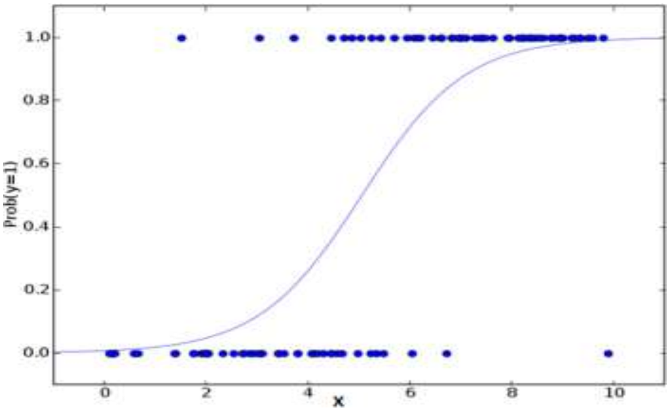
1. It should be positive (since p >= 0)
2. It should be less than equals to 1 (since p <= 1) Thus we satisfy these two conditions and get to the core of logistic regression. Let us denote g() with 'p' initially and eventually end up deriving this function.

Since probability must always be positive, we'll put the linear equation is in exponential form. For any value of slope and dependent variable, exponent of this equation will be positive.
p = exp(βo + β(Ag)) = e^(βo + β(Ag)) ----------------- (b)
To create the probability less than 1, let us divide p by a number greater than p.
p = exp(βo + β(Ag)) / exp(βo + β(Ag)) + 1 = e^(βo + β(Ag)) / e^(βo + β(Ag)) + 1 (c)
Using (a), (b) and (c), we can redefine the probability as: p = e^y/ 1 + e^y ------------------ (d)
where p is the probability of success. This (d) is the Logit Function
q = 1 - p = 1 - (e^y/ 1 + e^y) ---------- (e)
where q is the probability of failure On dividing, (d) / (e), we get,

l After taking log on both sides, we get,

log(p/1-p) is the link function.
After substituting the value of y, we'll get:

This is the equation used in Logistic Regression. Here (p/1-p) is the odds ratio. Whenever the log of odds ratio is found to be positive, the probability of success is always more than 50%. A typical logistic model plot is shown below. You can notice probability never goes below 0 to above 1.



5. Selecting the Best Algorithm

K-Nearest Neighbor for classification works as the best model as it gives the highest accuracy while testing which is visible from the above comparisons. It gives an accuracy of around 79% for training and 78% for testing set respectively.

Logistic regression works properly over the data. Much better than other models, but it couldn't get extract all the information present in data which is done by KNN. This could be seen in the plots. It also takes less accuracy overtraining and testingdatasets.

Decision Tree actually takes a lot of time as it runs which selects all variables automatically according to the data. This leads to a lot of computations and it takes approx.

74% accuracy for testing the data.
Random Forest could be better processed. But mostly works for long term processing. As we need to process for a short term period there would be loss of accuracy while testing. Hence it has higher accuracy than any other model for training which is 80%.

Gradient Boosting has various features and needed the data to be more precise. The plotting shows different features used for classification more accurately. Thus giving the highest accuracy value for training while loses its accuracy for testing. But for meal recommendation, it gives the highest accuracy as compared to SVM.Support Vector Machine works best for any numerical valued dataset. But here while processing such data its accuracy is not much use. Thus providing only 76% for testing, this is higher than the Decision Tree and Random Forest. But for meal recommendation it gives lower accuracy and precision as compared to Gradient Boosting which lowers the accurate recommendation.

The accuracy of the MLP is not as good as the other models at all, this is likely due to scaling of the data. Neural networks also expect all input features to vary in a similar way, and ideally to have a mean of 0, and a variance of 1. We must rescale our data so that it fulfills these requirements which takes a lot of time for processing and thus reduces its accuracy.

Hence, it would be better to select K-Nearest Neighboring model as the best model for classification of diabetes with training accuracy of 79% and testing accuracy of78% to process whole dataset and Gradient Boosting with accuracy approx. 52% for meal recommendation dataset.

6. CONCLUSION

The conclusion of our project is to identify the best food recommended for the diabetic patient using a decision tree, logistic regression, support vector machine, gradient boost, random forest. This plan will recommend the food- items based on not only Indian standards but also various dishes and recipes around the globe. The first part consists of various parameters such as pregnancies, glucose, blood pressure, sick, insulin, BMI, age, outcome. For the second part, a different dataset is taken the one that we made where considering calories, fats, proteins, sodium, carbohydrates, and glycaemic index prediction is done whether that a particular food item having so and so those six things are healthy or harmful for a diabetic patient. This will thus, will be very beneficial for the diabetic patient to follow up on different kinds of diets and foodstuff and hence save a lot of time.

REFERENCES

[1] Chang-Shing Lee,. Intelligent Ontological Agent for Diabetic Food Recommendation (2016) System Structure II

[2] Food Recommendation System Using Clustering Analysis for diabetic patients III Data Preparation, Diabetics diet.

[3] Kaggle Sites for food facts https://www.kaggle.com/openfoodfacts/world- food-facts

[4] Kaggle Site for Nutrition https://www.kaggle.com/tags/nutrition

[5] Kaggle Site for Recipe https://www.kaggle.com/hugodarwood/epirecipe