Diabetes Diagnostic Prediction using Scikit-learn, NumPy and Pandas

by **Tamojit Ghosh**

*(Department of Computer Science & Engineering, 2nd year)*

# Abstract

[Diabetes, also known as diabetes mellitus, is a group of metabolic disorders characterized by a high blood sugar level (hyperglycemia) over a prolonged period of time. Symptoms often include](https://en.wikipedia.org/wiki/Hyperglycemia) [frequent urination](https://en.wikipedia.org/wiki/Frequent_urination), [increased thirst](https://en.wikipedia.org/wiki/Polydipsia) and [increased appetite](https://en.wikipedia.org/wiki/Polyphagia). If left untreated, diabetes can cause [many health complications. Acute complications can include diabetic ketoacidosis, hyperosmolar hyperglycemic state, or](https://en.wikipedia.org/wiki/Hyperosmolar_hyperglycemic_state) [death](https://en.wikipedia.org/wiki/Death)[. Serious long-term complications include](https://en.wikipedia.org/wiki/Hyperosmolar_hyperglycemic_state) [cardiovascular disease](https://en.wikipedia.org/wiki/Cardiovascular_disease)[,](https://en.wikipedia.org/wiki/Hyperosmolar_hyperglycemic_state) [stroke, chronic kidney disease, foot ulcers, damage to the nerves, damage to the eyes, and cognitive impairment. Diagnosis of diabetes by a doctor is complicated, because several factors are involved](https://en.wikipedia.org/wiki/Cognitive_impairment) in the disease, and the diagnosis is subject to human error. A blood test does not provide enough information to make a correct diagnosis of the disease. So, we are using **support vector machine(svm)** to predict the diagnosis of diabetes based on several factors, viz. **Blood Glucose level, Blood Pressure level, Insulin, BMI**, etc., of patients from different ethnic background. The model here is trained on a small amount of data so we get to achieve an accuracy of **78%(approx.)**, a greater accuracy can be achieved if the experiment can be done on a large dataset.

# Methodoly

**NumPy** is used for for faster processing of the data and it's reshaping. **Pandas** is used to create DataFrame. **Scikit-learn** is used for standardization of the data, for splitting the data into training set and test set, for using svm and to check the accuracy score.

Firstly, we have to separate the dataset and labels, so that we can predict the outcome from data during testing :

The level of Glucose, BP, BMI, etc. vary excessively from person to person. This varying data can create confusion for our machine learning training model ,so we need to standardize the data :



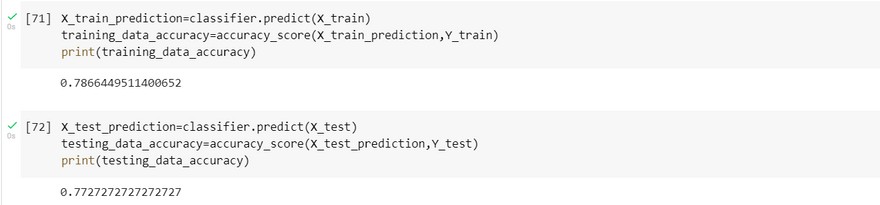
Our final model is going to train on the standardized dataset, so we need change the variable containing varying data into standardized data :

Developing a successful ml model involves two different types of data: **a training data** and **a testing data**:



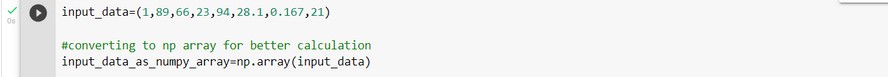
For training we are using **support-vector machine**, as we training the model with a small dataset so we are using **SVC(support-vector classifier)**, otherwise we have to use **LinearSVC**. We are using a linear model so we are specifying kernel as 'Linear'. Finally, we are training our svm classifier with training data(X\_train) and training label(Y\_train) and our trained model is getting stored in the classifier :

Our model is trained on 768 datasets so the accuracy achieved is a bit low, but the accuracy can be increased by using a much larger dataset :



**Creating the predictive system** :

For prediction we are taking any random data as input that is in form of csv(comma-separated value) ,then we are transforming the list into a **NumPy** array for faster calculations and for performing several operations on the input data:



We also need to reshape our input data as we are predicting only for one instance. Our model is trained on 768 datasets, so if we do not reshape our model it would expect 768 datasets instead of one during prediction and that might lead to an error:



Our model is trained on a standardized dataset, so if the input data during prediction is not standardized our model would fail to make any successful prediction. Thus, we need to standardize our input:



Finally, we can make a prediction out of the standardized dataset:

