Git-Hub

Web-Link

PRML

Major Project

Early Diabetes Classification

Introduction

Diabetes is one of the fastest growing chronic life threatening diseases that have already affected **422 million people worldwide** according to the report of World Health Organization (WHO), in 2018. Due to the presence of a relatively long asymptomatic

phase, early detection of diabetes is always desired for a clinically meaningful outcome. Around **50% of all people suffering from diabetes are undiagnosed because of its long-term asymptomatic phase.**

About this dataset.

This dataset contains 520 observations with 17 characteristics, collected using direct questionnaires and diagnosis results from the patients in the Sylhet Diabetes Hospital in Sylhet, Bangladesh.

Data Features

This Dataset consists of 520 observations with 17 characteristics which are:-

- **Age -** Given dataset is between 16 to 80.
- **Gender -** Male and Female only.
- **Polyuria** - Whether the patient experienced excessive urination or not.
- **Polydipsia** - Whether the patient experienced excessive thirst/excess drinking or not.
- **Sudden weight loss** - Whether the patient had an episode of sudden weight loss or not.
- **Weakness** - Whether the patient had an episode of feeling weak.
- **Polyphagia** - Whether the patient had an episode of excessive/extreme hunger or not.
- **Genital thrush** - Whether the patient had a yeast infection or not.
- **Visual blurring** - Whether the patient had an episode of blurred vision.
- **Itching** - Whether a patient had an episode of itch.
- **Irritability** - Whether a patient had an episode of irritability.

- **Delayed_healing** - Whether a patient had any noticed delayed healing when wounded.
- **Partial_paresis** - Whether a patient had an episode of weakening of a muscle/group of muscles or not.
- **Muscle_stiffness** - Whether the patient had an episode of muscle stiffness.
- **Alopecia** - Whether the patient experienced hair loss or not.
- **Obesity** - Whether a patient can be considered obese or not using his body mass index.
- **Class** - Presence of Diabetes.

Data PreProcessing

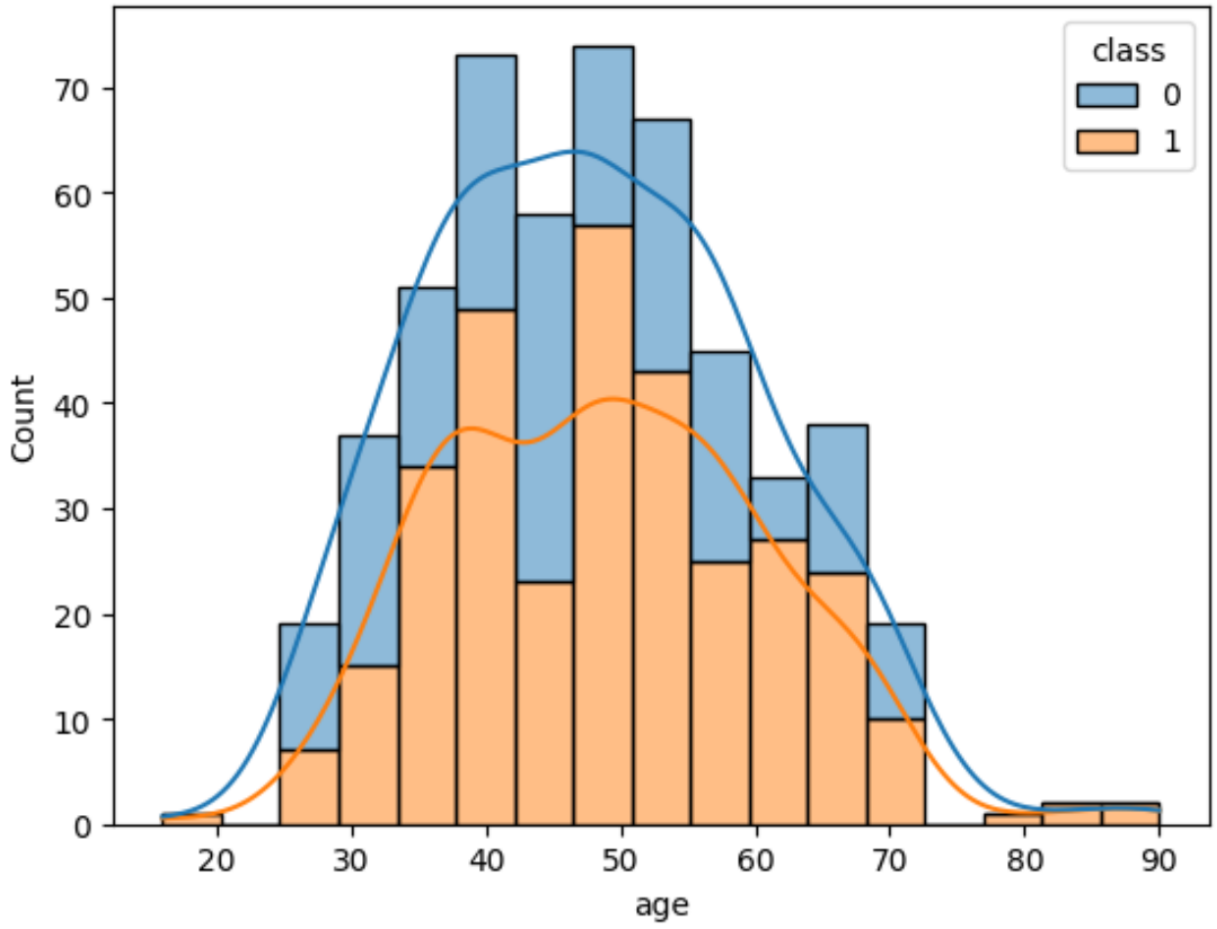
Data Info:- After encoding column 'Gender', we get our Data info as

```
RangeIndex: 520 entries, 0 to 519
Data columns (total 17 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   age                 520 non-null    int64
 1   gender              520 non-null    int64
 2   polyuria            520 non-null    int64
 3   polydipsia          520 non-null    int64
 4   sudden_weight_loss  520 non-null    int64
 5   weakness            520 non-null    int64
 6   polyphagia          520 non-null    int64
 7   genital_thrush      520 non-null    int64
 8   visual_blurring     520 non-null    int64
 9   itching             520 non-null    int64
 10  irritability        520 non-null    int64
 11  delayed_healing     520 non-null    int64
 12  partial_paresis     520 non-null    int64
 13  muscle_stiffness    520 non-null    int64
 14  alopecia            520 non-null    int64
 15  obesity             520 non-null    int64
 16  class               520 non-null    int64
dtypes: int64(17)
```

Now, we split the data into train and test with test_size = 0.2

For making the application more easy, we apply MinMaxScaler to X_train and X_test set which transforms features by scaling each feature in a given range(Here, it is 0 to 1).

The plot between count and age is



Data Prediction Models

Logistic Regression

We apply Logistic Regression to our dataset and find the test score = 95.19%.

Five-Fold Cross Validation

Taking n_splits = 5, we apply the Five-Fold Cross Validation and find that score = 91.11%.

Test Accuracy

The Test Accuracy is equal to 0.95.

F1-Score

The F1-Score between y_test and ypred_logistic(as described in the code) is 0.9612.

Recall

The Recall is 0.96.

Neural Network

Preparing data for Neural Network:-

Now, we define a PyTorch ‘Dataset’ class called ‘ddata’ that reads in a CSV file and converts it into tensors to be used for training a machine learning model.

The constructor (__init__) takes a file_name argument, reads in the CSV file using pandas read_csv function, and then converts the gender column from string labels (i.e., "Male" and "Female") to numerical labels (i.e., 0 and 1) using the replace method. It then selects the first 16 columns as input features and the last column as the target variable, and stores them as torch.tensor objects.

The __len__ method returns the number of samples in the dataset, which is the length of the target variable y_train.

The __getitem__ method takes an index idx and returns a tuple containing the input features and target variable for the corresponding sample at that index. Specifically, it returns the idx-th row of x_train and y_train tensors as (self.x_train[idx], self.y_train[idx]).

Then we load the diabetes data using ‘ddata’. Now, we load the data with batch size for train_loader = 30 and the same for test_loader = 10.

Neural Network Making:-

Here, we form a ‘diabetes_classifier’ named Neural Network with ‘**sigmoid’** as the activation function and some inputs:-

The neural network has 4 layers with 3 hidden layers.

No of neuron in each layer:-

1st layer = 32 neurons.

2nd layer = 16 neurons.

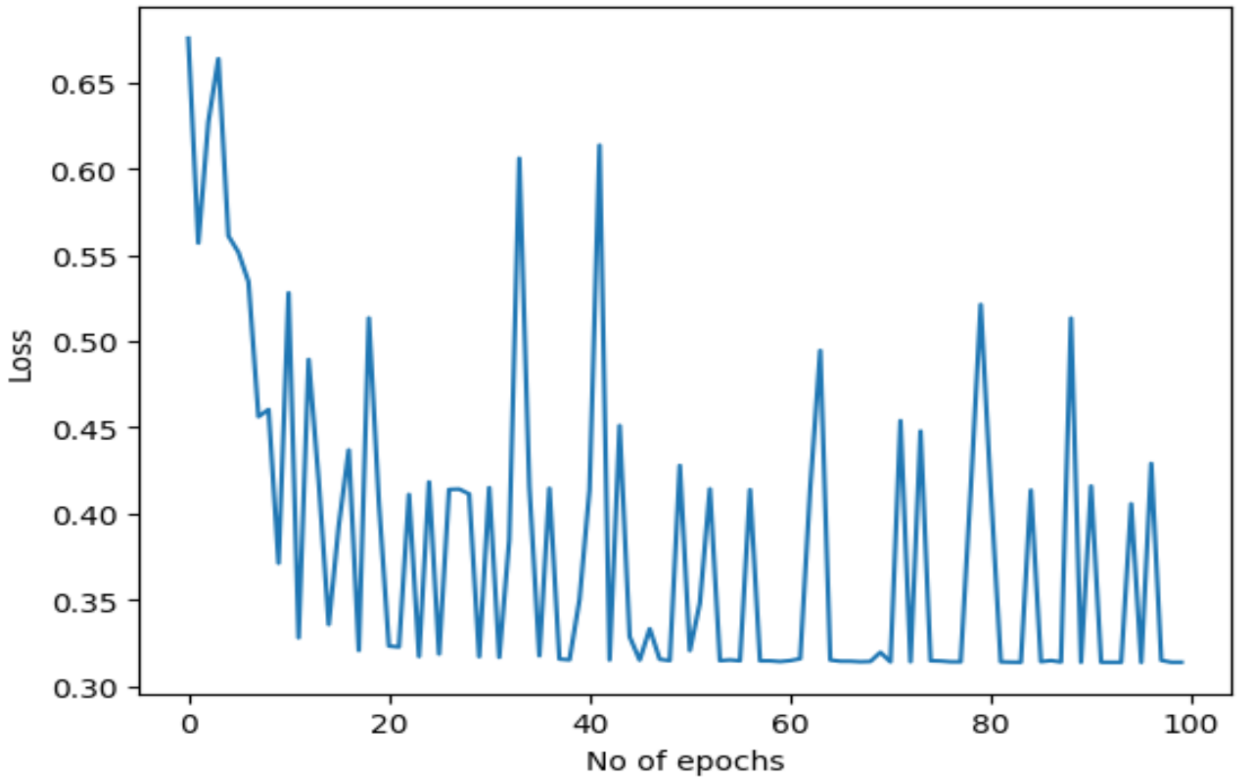
3rd layer = 8 neurons.

We use ‘Cross Entropy Loss’ as the criterion and ‘Adam Optimiser’ as the optimiser.

Train Model:-

Here, we train the model and also calculate the loss with each epoch.

The graph below shows the plot for the same.



Testing the Neural Network:-

We here first create a function ‘check_accuracy’ to check the accuracy of test_loader and train_loader.

The accuracy of test_loader is 97.5%.

The accuracy of the train_loader is 95.75%.

Test Accuracy

This is equal to 0.97.

F1-Score

This is equal to 0.97.

Recall

This is equal to 0.98.

SVM

Here, we train the svm_classifier model and then find various measures for it.

Five-Fold Cross Validation

With n_splits = 5,the cross_val_score for svm_classifier is 0.935.

Test Accuracy

This is equal to 0.99 for the svm_classifier.

F1-Score

This is equal to 0.99 for the svm_classifier.

Recall

The Recall is 1.0.

Decision Tree Classifier

Here, we train the dtc(decision_tree_classifier) model for the dataset and then find various measures for it.

Five-Fold Cross Validation

With n_splits = 5,the cross_val_score for dtc_classifier is 0.961.

Test Accuracy

The test accuracy for the dtc() model is 97.11%.

F1-Score

The F1-Score for the dtc() model is 0.976.

Recall

The Recall is 0.96.

Random Forest Classifier

We here train the Random Classifier model.

Five-Fold Cross Validation

With n_splits = 5,the cross_val_score for RandomForestClassifier is 0.968.

Test Accuracy

This is equal to 99.03% for the RandomForestClassifier model.

F1-Score

This is equal to 0.992 for the RandomForestClassifier model.

Recall

The Recall is 0.98.

Bernoulli Naive Bayes

We here train the Bernoulli Naive Bayes model.

Five-Fold Cross Validation

With n_splits = 5,the cross_val_score for Bernoulli Naive Bayes is 0.865.

Test Accuracy

This is equal to 90.38% for the Bernoulli Naive Bayes model.

F1-Score

This is equal to 0.921 for the Bernoulli Naive Bayes model.

Recall

The Recall is 0.92.

Scores Table

	Logistic Regression	Neural Network	SVM	DTC	Random Forest	Bernoulli Naive Bayes
Cross Validation Score	0.91	-	0.93	0.96	0.96	0.86
Test Accuracy	0.95	0.97	0.99	0.97	0.99	0.90
F1-Score	0.96	0.97	0.99	0.97	0.99	0.92
Recall	0.96	0.98	1.0	0.96	0.98	0.92

Conclusion

This was a Diabetes Classification problem and we know that **Recall** is always the most important measure score in any disease classification problem since we want to keep a **minimum number of False Negatives**. Observing the table above, we can see that recall for SVM classifiers is equal to **1.0** and also the other parameters of SVM classification perform better than any of the other classification approaches. Thus, if we prioritize, we keep SVM above the other methods.

Presented By:-

- Lokesh Tanwar (B21EE035)
- Mohit Sharma (B21EE037)
- Mridul Gupta (B21EE038)