Exposys Internship Program

Data Science Internship

By - Vaishnavi Pulate

Problem Statement

Task:

1. Prepare the data-set using several methods to train the model.

2. Build a model which can give high accuracy of predicting the disease.

Introduction

- Diabetes is one of the most acute diseases, and is present worldwide.
- A major reason of deaths in adults across the globe includes this chronic condition.
- In order to predict disease, the proposed model uses **Py-Torch framework.**
- Dataset used for building the model is download from Kaggle Datasets.
- Dataset consist of 768 rows and 9 columns.
- Prediction of the disease is done based on columns(Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, DiabetesPedigreeFunction**,** Age)
- The model is build using nn.Sequential from torch.nn.

Existing Method

- Artificial neural network (ANN), random forest (RF) and K-means clustering techniques were implemented for the prediction of diabetes.
- The ANN technique provided a accuracy of 75.7%, and may be useful to assist medical professionals.

1. **Artificial neural network:**The objective of ANN is to convert input into significant output. Input is the combination of a set of input values that are associated with the weight vector, where the weight can be negative or positive.
2. **Random Forest** : Random forest builds multiple decision trees and aggregates them to achieve more suitable and accurate results. This algorithm also solves the overfitting issue.
3. **K-means Clustering:** K-Means clustering works on numerical data, in which K is represented as centers of clusters. Taking the distance of each datapoint from the center it assigns each instance to a cluster, and moves cluster centers by taking the means of all the data points.

Proposed Method with Architecture

- The given problem statement is solved using Deep Learning model where Artificial Neural Network(ANN) is build.
- For building model **Py-Torch** framework is used.
- Model is build using nn.Sequential from torch.nn.
- In the Method there are 2 models build which has different number of layers (sigmoid and Linear).
- Model 1 : It Consist of 2 Linear Layer and 2 Sigmoid Layer
- Model 2: It Consists of 4 Linear and 4 Sigmoid layer

Proposed Method with Architecture

1. Model 1

```
[ ]  class FirstNetwork_1(nn.Module):

        def __init__(self):
          super().__init__()
          torch.manual_seed(0)
          self.net = nn.Sequential(

              nn.Linear(8, 4),
              nn.Sigmoid(),
              nn.Linear(4, 1),
              nn.Sigmoid()
              #nn.Softmax()
          )

        def forward(self, X):
          return self.net(X)
```

2. Model 2

```
[ ]  class FirstNetwork_2(nn.Module):

        def __init__(self):
          super().__init__()
          torch.manual_seed(0)
          self.net = nn.Sequential(

              nn.Linear(8, 6),
              nn.Sigmoid(),
              nn.Linear(6, 4),
              nn.Sigmoid(),
              nn.Linear(4, 2),
              nn.Sigmoid(),
              nn.Linear(2, 1),
              nn.Sigmoid()
              #nn.Softmax()
          )

        def forward(self, X):
          return self.net(X)
```

Methodology

1. Requirement analysis
2. Database collection - Downloading dataset from Kaggle
3. Data Preprocessing - Storing Outcome in other variable and Converting Pandas Dataframe to torch.tensor
4. Identification of suitable deep learning framework - Py-torch
5. Train: Test: Validation split
6. Building the model using nn.Sequential from torch.nn library

Implementation

- Specification:
    - Dataset - Pima Indians diabetes dataset.
    - Programming language: Python 3
    - Platform: Google Colab
    - Framework: PyTorch 1.9.0v

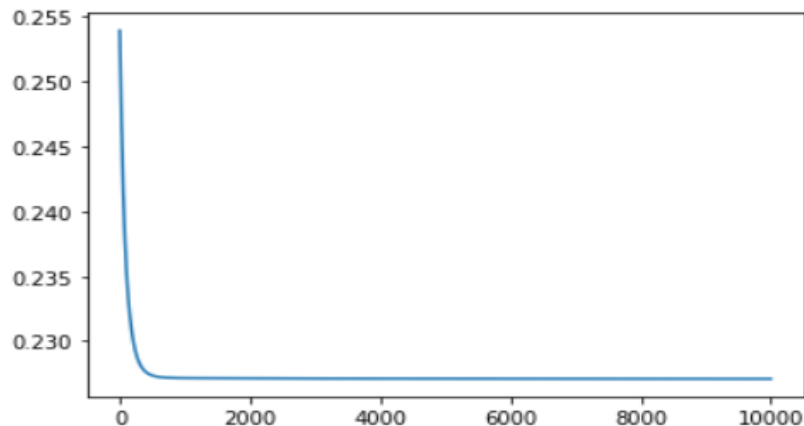For implementation Google Colab platform and Py-Torch framework is used .

Results

Traning Accuracy : 65.14657980456026

Test Accuracy: 66.88311688311688

Loss Plot:


```
Loss before training 0.2539098262786865
Loss after training 0.22707775235176086
Time taken:  15.93601942062378
```

Conclusion

- In the given problem statement we have used Deep Learning for building a suitable model in order to predict diabetes at early stage.
- Model was giving good accuracy in-order to predict the disease.
- Prediction was done based on certain characteristics of individual's health