CS432
Term project
Dataset project report
Abdullah Alrebdi - 382122696

One of the greatest powerness of AI and machine learning is that you don't need to write a program explicitly to do a difficult task. Instead, you make the program learn and embed the previous experience into itself.

Introduction

Every year, 12 million deaths that the World Health Organization has estimated that caused by heart diseases. In this project report I will be doing a prediction classifier (logistic regression) with an interesting dataset that has been created by doctors which are the artists of making features in that field and it'll predict "does a given patient will have a risk in ten years of heart disease" or not.

Dataset

In this project I will be using an interesting dataset about heart disease with lots of features. It contains **15 features**/attributes and almost **1300 instances**/row/entity after downsizing (details later). The features are:

gender	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp

continue…

diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD

Features details as follows:

- **Current Smoker:** Does the patient smoke.
- **CigsPerDay:** Average number of cigarettes a patient smokes a day.
- **BPMeds:** have a patient take blood pressure medications.
- **PrevalentStroke:** Have the patient ever had a stroke.
- **PrevalentHyp:** is the patient ever was hypertensive.
- **Diabetes:** did the patient have diabetes.
- **TotChol:** cholesterol level.
- **SysBP:** systolic blod pressure.
- **DiaBP:** diastolic blood pressure.
- **BMI:** Body mass.
- **Heart Rate:** heart rate.
- **Glucose:** glucose level.
- **tenYearRisk:** of heart disease CHD.

Preprocessing:

Using a variety of tools you can make datasets cleaner, balanced and easier to either make a model predict properly or make it more readable. **Preprocessing have gone into problems as follows:**

- We have empty fields(a feature/instance value)
- unbalanced data that makes tons of problems
  - -overfitting
  - -false positive outcome

Overall , having LOTS of instances often drives a model to these problems. However, with some undersampling from 4000+ to 1300 rows solved it.

So what did I do?

- eliminate any missing values since we have 4000+ rows
- downsizing the data set to 1400
- made some changes in features values to make them even better
  - -gender has gone from (male , female) to (1 , 0) and that applies to

any feature like gender.
-any INT value or string has been changed to float (algorithm favorite)
-split data into X_train y_train X_test y_test

Last preprocessing which is the most important one to get correct predictions that is **normalization**:

- All the values in the dataset are in the range of -infinity to +infinity thus, we want to make them in the range of [0,1] to easily predict by getting min & max of every feature and apply [0,1] scaling.

Prediction

After a brief introduction to the dataset and preprocessing now the data is loaded to the model and ready for training which is its job to train the model for many steps and measure the loss function which is how far are we from the real class and then optimized by gradient descent until the result convergence.

Component:

- sigmoid function returns y.
- Train function (gradient descent in its core).
- predict function.

Final result

Here are ten samples:

```
0        : 0
0        : 0
0        : 0
0        : 0
1        : 1
0        : 0
1        : 1
0        : 0
0        : 0
0        : 0
```

```
Accuracy score is: 75.53%
precision score is: 78.26%
```

Although we got perfect predictions here in above samples but nothing is perfect this is why we need measures like accuracy.