Comp3220 AI Assignment

Due Date: November 30 2025
Time: 11:59 pm
Marks - 40

Description

You are given medical data on several patients . Your job is to use this data to create a model that can predict whether a patient has heart disease. A jupyter notebook is provided with two csv files namely **heart_train.csv** and **heart_test.csv**. Your job is to edit this jupyter notebook and complete the objectives below. (This assignment is based on the Kaggle Heart Failure Prediction COMP3220 (UWI).

Data Description (heart_train.csv)

- Age: age of the patient [years]
- Sex: sex of the patient [M: Male, F: Female]
- ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
- RestingBP: resting blood pressure [mm Hg]
- Cholesterol: serum cholesterol [mm/dl]
- FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
- RestingECG: resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]
- MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]
- ExerciseAngina: exercise-induced angina [Y: Yes, N: No]
- Oldpeak: oldpeak = ST [Numeric value measured in depression]
- ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
- HeartDisease: output class [1: heart disease, 0: Normal]

Note: **heart_train.csv** should be used to build a model and test its accuracy. **heart_test.csv** does not have the attribute HeartDisease. Use this data file to make predictions using the model.

Objectives

1. Read from the **heart_train.csv** file into a pandas dataframe.
2. Pre-process and clean your data. This is the process of making sure your data is ready for training, for example removing null values from the table. Another example of preprocessing is normalizing your data. Normalization involves converting columns that have huge values to values more manageable by your model eg. (-1, 1).
3. Extract features to use from the table e.g RestingBP, Cholesterol, FastingBS etc. It is your job to decide the best features to use. **(Hint, try using the columns that have numeric values).** For categorical columns, try to convert those to numerical.
4. Extract the HeartDisease column to use as the labels for your model.
5. Split the dataset into train and test using the train_test_split function provided
6. Create a Logistic Regression model and fit it to your train data.
7. Test the results on your test data. Report on the percentage accuracy.
8. Create a simple neural network to fit your train data.
9. Test the results on your test data. Report on the percentage accuracy.
10. Comment on why normalization is important and how it affects neural networks.
11. Try adding more neurons to your neural net (About a 100) and comment on the test accuracy. If it went down, comment on the reason you think it did.
12. Lastly, create a kaggle account. Use your model to predict the data in **heart_test.csv.** A sample submission file has also been included for you.Create a new csv as shown below and submit your model to the UWI Heart Failure competition submission.

Create a document in which you should report the following:

- Accuracy rates of models.
- Discussion on the importance of normalization.
- Discussion on dangers of overfitting with large neural networks(Number 11 above)
- Take a screenshot of your score on the leaderboard and include it in the report.

PatientId	HeartDiseas
818	0
819	0
820	1
821	0
822	1
823	1
824	0
825	1
826	0
827	0
828	1
829	1
830	1

Submit three documents - your notebook (ipynb), report file (e.g. doc, pdf) and predictions on the heart_test.csv as a csv file. On VLE submit the files as a zipped file. Make sure the id numbers of both persons are in the report file.