Comp3220
AI Assignment
Fall 2024

Due Date: November 15 2024

Description

You are given data on a number of loan applications. Your job is to use this data to create a model that can predict whether someone will be approved for a loan. Two csv files are provided: **train.csv** and **test.csv.** Your job is to create a jupyter notebook. In that jupyter notebook you must show how you extracted the data, preprocess the data and create a model then evaluate the model. You must also use this model to make predictions on a csv file that will be provided and submit the predictions to kaggle.

Data Description (train.csv)

- **person_age** - age of person
- **person_gender** - gender of person (male/female)
- **person_education** - education level of person eg. Bachelor, HighSchool etc
- **person_income** - income of the person
- **person_emp_exp** -
- **person_home_ownership** - whether the person wants rent or mortgage
- **loan_amnt** - how much the person wants to borrow
- **loan_intent** - purpose of the loan
- **loan_int_rate** - interest rate of the loan
- **loan_percent_income** -
- **cb_person_cred_hist_length**
- **credit_score -** credit score of the person
- **previous_loan_defaults_on_file**
- **loan_status** - whether the person got approved or not (1=approved, 0=declined)
- **loan_id** - id number for the loan

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

<u>Objectives</u>

1. Read from the file **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. An example is shown for you where the null values are removed from the age column. 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 . person_age, person_education, person_income, loan_amt etc It is your job to decide the best features to use.
4. Extract the loan_status 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 **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 COMP3220 Loan Prediction competition submission.
13. Try to use models other than logistic regression. Look into models such as decision trees and support vector

Create a document in which you should report the following:

- Accuracy rates of models.
- Discussion on the importance of normalization.
- Discussion on using a neural network with a lot of parameters
- Take a screenshot of your score on the leaderboard and include it in the report.

loan_id	loan_status
12332	1
34332	0
3899	1
1122	1

Submission

- Jupyter notebook
- Report file
  - Report on accuracy of models used in jupyter notebook
  - Comment on importance of normalization of data to training neural networks.
  - Discuss the phenomenon called overfitting with neural networks and discuss how the number of parameters can affect it.
- Submission File
  - A csv file with the predictions created from the fraud_test.csv file. The file should be formatted as shown in the screenshot below.
- Screenshot of Kaggle Place on Leaderboard.