

COL341: Assignment 2

Neural Network

Name

neural - Run the executable program for linear regression

Synopsis

```
./neural <part> <tr> <ts> <out> <other_options>
```

Description

This program will train neural network model using given code on train data, make predictions on test data and write final predictions in given output file.

Note: You should be able to find the input_size and num_output_classes from training data.

Options

- **part:** Part as per question i.e. a/b/c.
- **tr:** File containing training data in csv format where 1st entry is the target
- **ts:** File containing test data in csv format where 1st entry is the target
- **out:** Output file for predictions. One value in each line.
- **other_options:** Only for part a
 - batch_size
 - η_0 (Initial Learning rate)
 - activation_function: relu, tanh, sigmoid
 - space seperated list of hidden layer sizes

Example

1. Suppose
 - batch_size: 100
 - activation_function: relu
 - three hidden layers containing 50, 10 and 5 perceptrons each

```
./neural a train.csv test.csv output 100 relu 50 10 5
```
2.

```
./neural b train.csv test.csv output
```
3.

```
./neural c train.csv test.csv output
```

Data

- devnagri_train.csv: Train data
- devnagri_test_public.csv: Public Test data

Note: In the Public test data, actual class labels are replaced with -1

Marking scheme

Marks will be given based on following categories:

- For code, you can get 0 (error), half (code runs fine but predictions are incorrect within some predefined threshold) and full (works as expected).
- For part-b and part-c, marks will be given based on training time and accuracy on test data-set. There will be relative marking for this part.
- For part-b and part-c marking will be done in two parts: code (75%) and report(25%).

Submission

1. Your submission should be "ENTRY_NO.zip".
2. Make sure you clean up extra files/directories such as "__MACOSX"
3. Command "unzip ENTRY_NO.zip", should result in a single directory "ENTRY_NO".

Naive Bayes

Name

naive - Run the executable program for Naive Bayes

Synopsis

```
./naive <part> <tr> <ts> <output>
```

Description

This program will train naive bayes model using given code on train data, make predictions on test data and write final predictions in given output file.

Options

- part
Part as per question i.e. a,b or c.
- tr
File containing training data in csv format where 1st entry is the target

- ts
File containing test data in csv format where 1st entry is the target
- out
Output file (write your predictions in this file)

Example

```
./naive a train.csv test.csv output
```

Data

- amazon_train.csv: Train data
- amazon_test_public.csv: Public Test data

Note: In the Public test data, actual class labels are replaced with -1

Marking scheme

Marks will be given based on following categories:

- For code: you can get 0 (error), half (code runs fine but predictions are incorrect within some predefined threshold) and full (works as expected).
- For part-c, marks will be given based on training time and macro-Fscore on test data-set. There will be relative marking for this part.
- For part-c marking will be done in two parts: code (10) and report(5).

Submission

1. Your submission should be "ENTRY_NO.zip".
2. Make sure you clean up extra files/directories such as "__MACOSX"
3. Command "unzip ENTRY_NO.zip", should result in a single directory "ENTRY_NO".