

2023-August-Mathematics of Network Algorithms

Quiz 3

-
- Deadline: 7th Nov, 1:10 pm. Please submit your assignment in the specified format [here](#).
 - You **must** submit python file named as: *enrollment-nr-assignment-nr-question-nr-student-name.py*
For example, for the student XYZ with enrolment number 20251010, a solution for the first question should be in the file 20251010-03-01-XYZ.py.
 - Your code will be evaluated with the command `$ python3 20251010-03-01-XYZ.py`.
 - Any deviation from these instructions related to submission will adversely affect the number of test cases your algorithm can solve.
 - The points for each question will be determined by the quality of the output.
-

1. (15 pts) [Fashion MNIST] Write a python program that creates a neural network to train `fashion_mnist` dataset from `keras` that has one hidden layer.

Your output should be a single line containing your roll number, the number of perceptrons in the hidden layer, and the accuracy rate on test data.

For example, if your neural network uses 64 neurons in the hidden layer, and its test data accuracy is 78.56%, then the output should **only** be

```
20251010 64 0.7856
```

and no other information.

2. (15 pts) [Email Spam Detector] We have two csv files named *quiz-3-Q2-train.csv* and *quiz-3-Q2-test.csv*, containing information of 1000 and 300 randomly picked email files and their respective labels for spam or not-spam classification, respectively.

The csv files respectively contains 1000 and 300 rows, each row for an email. There are 3002 columns. The first column indicates Email name. The name has been set with numbers and not recipients' name to protect privacy. The last column has the labels for prediction : 1 for spam, 0 for not spam. The remaining 3000 columns are the 3000 most common words in all the emails, after excluding the non-alphabetical characters/words. For each row, the count of each word(column) in that email(row) is stored in the respective cells. Thus, information regarding all emails are stored in a compact dataframe rather than as separate text files.

Write a python code that creates a neural network with one hidden layer that fits the above training data and evaluates test data.

Your output should be a single line containing your roll number, the number of perceptrons in the hidden layer, and the accuracy rate on test data.

For example, if your neural network uses 32 neurons in the hidden layer, and its test data accuracy is 85.96%, then the output should **only** be

```
20251010 32 0.8596
```

and no other information.

Hints: `reshape`, `astype`, `verbose`, `to_categorical` from `keras.utils`