CMPE 258, Spring 2018

Midterm Exam #1

Due 11:59pm on Saturday, March 10th, 2018

Notes

This Midterm exam should be submitted in Canvas as a format of ipython (Jupyter) notebook.
The filename should be 'exam_1_yourFirstName_LastnName.ipynb'.
The submitted ipynb should be executable without any extra work and supposed be finished within 10 minutes in my computer (windows 7, 4 core, 16GB memory).
Python 2.7 environment is preferred but python 3.X may be acceptable. However, you should clearly state that if your code cannot be executable in python 2.7.

You should not discuss how to solve the problem with other students and the work should be your own. If any portion of the code is similar to others, it will be treated as cheating.

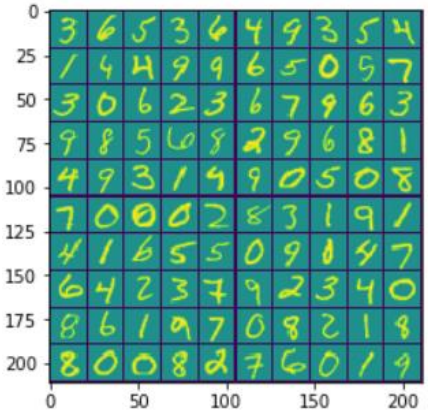
Please do not use any library except pandas, numpy, and matplotlib.pyplot

Grading policy

If the code can be executable without any extra effort and get reasonable result, I gave point based on the accuracy of the testing data.
If extra effort is needed to get reasonable result (whatever it is), I will take out 5 to 10 points from the accuracy of the testing data.
If the code cannot be executable, I take out 20 to 100 points depending on the error even though it looks reasonable.

Deep Neural Network with many (more than two) hidden layers

Download data files (exam1_train.csv and exam1_test.csv) from canvas/files/exam_1.
Using Jupyter notebook, load the data.
This data is a subset of the MNIST handwritten digit dataset (http://yann.lecun.com/exdb/minst/).
Each row is a 20 pixel by 20 pixel grayscale image of the digit. Each pixel is represented by a floating point number indicating the grayscale intensity at that location.



< MNIST handwritten digit>

The 20 by 20 pixels is unrolled into a 400-dimensional vector.
The last column ‘y’ is the label for the row.

	0	1	2	3	4	5	6	7	8	9	...	391	392	393	394	395	396	397	398	399	y
0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	5
1	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	9
2	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	7
3	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	6
4	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	5

1. **(30pts) Define functions**

Please define the following functions.

- One-hot encoding
- Activation function(s)
- Forward propagation with regularization
- Backward propagation with regularization
- Gradient descent with regularization
- Softmax

2. Split data

Please split each data (Train & Test) set as input (x) and output (y) set.
Input set is the columns starting 0 to 399.
Output set is the column of ‘y’.

3. (10pts) Initialize parameters

Please initialize weight coefficients and bias terms for each layer properly. You can use any initializing methods as you wish.

4. (30pts) Deep Neural Network model with more than 2 hidden layers

Please build neural network model using input layer (400 neurons), many (at least two) hidden layers, and output layer (10 neurons) using training data set. For each hidden layer, you can assign many neurons as you wish. For activation functions and regularization methods, you can use any method as you wish.

5. (10pts) Predictions

Please predict digit using softmax function.
Please calculate accuracy for the prediction using training data set and testing data set.

6. (20pts) Optimization

Please optimize your model using various hyper parameters such as number of hidden layer, number of neurons in hidden layer, activation function, regularization method, initialization method, learning rate, and number of iteration.
Please print out the optimized accuracy for training and testing data set.