Homework 4 – Deep Neural Networks (CS525 191N, Whitehill, Spring 2017)

You may complete this homework assignment either individually or in teams up to 2 people.

1. **Softmax regression (aka multinomial logistic regression)** [20 points]: In this problem you will train a 2-layer neural network to classify images of hand-written digits from the MNIST dataset. The input to the network will be a 28×28-pixel image (converted into a 784-dimensional vector); the output will be a vector of 10 probabilities (one for each digit). Specifically, the network you create should implement a function $f : \mathbb{R}^{784} \rightarrow \mathbb{R}^{10}$, where the kth component of $f(x)$ (i.e., the probability that input $\mathbf{x}$ belongs to class k) is given by

$$\frac{\exp(\mathbf{x}^\top \mathbf{w}_k)}{\sum_{k'=1}^{10} \exp(\mathbf{x}^\top \mathbf{w}_{k'})}$$

   The cross-entropy loss function should be

$$J(\mathbf{w}_1, \ldots, \mathbf{w}_{10}) = -\frac{1}{m} \sum_{j=1}^{m} \sum_{k=1}^{10} y_k^{(j)} \log \hat{y}_k^{(j)}$$

   where m is the number of examples. Note that each $\hat{y}_k$ implicitly depends on all the weights $\mathbf{w}_1, \ldots, \mathbf{w}_{10}$.

   To get started, first download the MNIST dataset (including both the training and testing subsets) from the following web links:

   - `https://s3.amazonaws.com/jrwprojects/mnist_train_images.npy`
   - `https://s3.amazonaws.com/jrwprojects/mnist_train_labels.npy`
   - `https://s3.amazonaws.com/jrwprojects/mnist_test_images.npy`
   - `https://s3.amazonaws.com/jrwprojects/mnist_test_labels.npy`

   These files can be loaded into `numpy` using `np.load`.

   Then implement gradient descent to minimize the cross-entropy loss function. Note that, since there are 784 inputs and 10 outputs, there will be 10 separate weight vectors, each with 784 components. Alternatively, you can conceptualize the weights as a 784×10 matrix.

   Finally, after optimizing the weights on the training set, compute both (1) the loss and (2) the accuracy (percent correctly classified images) on the **test** set. **Include both the cross-entropy loss values and the "percent-correct" accuracy in the screenshot that you submit.**

In addition to submitting your Python code in a file called `homework4_WPIUSERNAME1.py` (or `homework4_WPIUSERNAME1_WPIUSERNAME2.py` for teams), please submit a PDF file containing a screenshot of (1) the last 20 iterations of your gradient descent on the training data Name the file `homework4_WPIUSERNAME1.pdf` (or `homework4_WPIUSERNAME1_WPIUSERNAME2.pdf` for teams).

Update: *please submit both the PDF and Python files in a single Zip file.*