Neural Networks and Deep Learning
26:711:685:01 Spring 2020
Homework 1

Instructor: Farid Alizadeh
Due Date: Wednesday February 12, 2020, at 11:50PM

last updated on February 11, 2020

Please answer the following questions in **electronic** form and upload and submit your files to Sakai Assignment site, before the due date. Make sure to click on the **submit** button. The file format should be **pdf**. You may either typeset or hand write your answers. In case of hand-write transform your work into a **pdf** file using apps such as **CamScanner**. You should have only a single pdf file.
For this homework, you also have Python scripts, include the text of the script with your pdf submission (no separate file is needed).

1. **Keras/Python Project:** Keras comes with another set of data very similar to the NIST data for hand written digits. The data set is called `Fashion-MNIST`. Similarly to NIST data, there are ten classes, showing pictures of clothing items. These items are stored in low-resolution 28×28 images made up of integers between zero and 255, indicating the grey scale of the pixel they are representing. This data is stored in the Keras package in `keras.datasets.fashion_mnist`.

   For programming questions supply ample documentation, explaining what you are doing. I expect to be able to run your code on my laptop Running Python 3.7.

   1a) Write a script implementing a one–hidden-layer neural network to learn various clothing items. Use the Keras **Sequential** approach to building your neural net. Use the *ReLU* activation function for the layer. Use 512 units for the hidden layer. Using five epochs fit the model to the data and print the results on test data as follows:

   Import from `sklearn.metrics` the function `confusion_matrix` (and study its documentation.) Print the confusion matrix of the errors on the test data. When printing the confusion matrix, write the actual names of classes (e.g. trousers, sandals, etc.,) not the numerical codes, for both actual and predicted values. These names are in order:

   ```
   names=["T-shirt/top", "Trouser", "Pullover", "Dress", "Coat",
   "Sandal", "Shirt", "Sneaker", "Bag", "Ankle boot"]
   ```

   So item '0' is `T-shirt/Top`, item '1' is `Trouser`, etc. Write the script in file called `Q1a.py`.

   1b) Repeat part 1a) but this time use the **functional** form to build the same model. Experiment with the number of epochs set at 5, and set at 20. Make sure to print the confusion table for bith epoch set at 5, and set at 20. Write the script in a file called `Q1b.py`.

   1c) Write a Python script that runs through the test data. For each one if your model in parts a) or b) predicts incorrectly, it should draw the pattern, and also show the value of the predicted softmax function for this incorrectly classified pattern. Examine and print the percentage of times among incorrect predictions where the second choice is the correct one. Again, write the actual names of classes not their numerical codes for both actual and predicted values, as well as the class probabilities. Write the script in a file called `Q1c.py`.

   1d) Repeat part b) but this time use the logistic (sigmoid) function as your activation function. Compare the results. Write the script in a file called `Q1d.py`.

2. **Theory/conceptual questions:**

   2a) A set of random numbers $X_1, X_2, \ldots, X_m$ are drawn *uniformly* from the interval $[0, a]$. We observe these numbers but we don't know a.

Formulate and solve the maximum likelihood estimate of a from this data. (**Tricky!**)

2b) Repeat part a) but suppose the data are derived uniformly from the interval $[a, b]$. Having observed, $X_1, X_2, \ldots, X_m$ find the maximum likelihood estimate of a, b.

2c) **Bonus question:** A set of m points $x_1, x_2, \ldots, x_m$ all in d-dimensional Euclidean space are drawn uniformly from a ball centered at point $\mathbf{c}$ and with radius r. We don't know r and $\mathbf{c}$. Formulate the optimization problem that finds the maximum likelihood estimates of r and $\mathbf{c}$. (**Note:** I do not expect you to solve the optimization problem, only to formulate it.)