Stony Brook University
CSE512 – Machine Learning – Fall 18
Homework 7, Due: Thur Dec 6, 2018, 11:59PM

This homework contains 3 questions. The last two questions require programming. The maximum number of points is 100.

1 Manual calculation of one round of EM for a GMM [20 points]

(Extended version of: Murphy Exercise 11.7) In this question, we consider clustering 1D data with a mixture of 2 Gaussians using the EM algorithm. You are given the 1-D data points $x = [1 \;\; 10 \;\; 20]$.

M step

Suppose the output of the E step is the following matrix:

$$R = \begin{pmatrix} 1 & 0 \\ 0.3 & 0.7 \\ 0 & 1 \end{pmatrix}$$

where entry $R_{i,c}$ is the probability of observation x_i belonging to cluster c (the responsibility of cluster c for data point i). You just have to compute the M step. You may state the equations for maximum likelihood estimates of these quantities (which you should know) without proof; you just have to apply the equations to this data set. You may leave your answer in fractional form. Show your work.

1. *[2 points]* Write down the likelihood function you are trying to optimize.
2. *[4 points]* After performing the M step for the mixing weights π_1, π_2, what are the new values?
3. *[4 points]* After performing the M step for the means μ_1 and μ_2, what are the new values?
4. *[4 points]* After performing the M step for the standard deviations σ_1 and σ_2, what are the new values?

E step

Now suppose the output of the M step is the answer to the previous section. You will compute the subsequent E step.

1. *[2 points]* Write down the formula for the probability of observation x_i belonging to cluster c.
2. *[4 points]* After performing the E step, what is the new value of R?

2 Generative Adversarial Networks (Programming) [40 points]

In this section, you will train generative adversarial networks (GAN) to generate images using PyTorch. We use the MNIST data which is 60,000 training and 10,000 test images. Refer to the *jupyter notebook* for details.

You will first train a GAN for generating new images. Then try to improve the network architecture and attach your results with the jupyter notebook. Also add the hyper-parameters explored.

The detailed instructions and questions are in the jupyter notebook $GAN.ipynb$. In this file, there are 7 "To-Do" locations for you to fill. The score of each To-Do is specified at the spot.

We recommend using virtual environment for the project. If you choose not to use a virtual environment, it is up to you to make sure that all dependencies for the code are installed globally on your machine. To set up a virtual environment, run the following in the command-line interface:

```
cd your_hw7_folder
sudo pip install virtualenv      # This may already be installed
virtualenv .env                  # Create a virtual environment
source .env/bin/activate         # Activate the virtual environment
pip install -r requirements.txt  # Install dependencies
# Note that this does NOT install TensorFlow or PyTorch,
# which you need to do yourself.
#
# Work on the assignment for a while ...
# ... and when you're done:
deactivate                       # Exit the virtual environment
```

Note that every time you want to work on the assignment, you should run 'source .env/bin/activate' (from within your hw7 folder) to re-activate the virtual environment, and deactivate again whenever you are done.

3 Action Classification Using RNN (40 points)

In this section, you will train recurrent neural networks (RNNs) to classify human actions. RNNs are designed handle sequential data.

For human action recognition, you will be using skeleton data that encodes the 3D locations of 25 body joints. The data is collected by Kinect v2. There are 10 different action classes. There are 4000 training sequences, 800 validation sequences, and 1000 test sequences. Each sequence has 15 frames, each frame is a 75-dimension vector (the xyz positions of 25 joints).

You will first train a LSTM for action classification. Then try to improve the network architecture and attach your results with the jupyter notebook. Also add the hyper-parameters explored.

The detailed instructions and questions are in the jupyter notebook *RNN_ActionClassify.ipynb*. In this file, there are 4 ToDo locations for you to fill. The score of each ToDo is specified at the spot.

You will need to install the following extra packages:

```
pip install h5py
pip install git+https://github.com/pytorch/tnt.git@master
```

4 What to submit?

4.1 Blackboard submission

You will need to submit both your code and your answers to questions on Blackboard. Put the answer file and your code in a folder named: SBUID_FirstName_LastName (e.g., *10947XXXX_lionel_messi*). Zip this folder and submit the zip file on Blackboard. Your submission must be a zip file, i.e, SBUID_FirstName_LastName.zip. The answer file should be named: answers.pdf. The first page of the answers.pdf should be the filled cover page at the end of this homework. The remaining of the answer file should contain:

1. Answers to Question 1.
2. Kaggle rank and test accuracy of the best model.

You can use Latex if you wish, but it is not compulsory.

4.2 Kaggle submission

Experiment with architectures, hyperparameters, loss functions, and optimizers to train a model that achieves better accuracy on the action recognition validation set for RNN code.

Use the following link to access the Kaggle competition page for this question: `https://www.kaggle.com/t/934b80879bd741e6ac1967195604d4d9`. Use your @stonybrook.edu ID to login and make sure your username (as displayed on the leaderboard) is your SBU ID.

5 Cheating warnings

Don't cheat. You must do the homework yourself, otherwise you won't learn. You cannot ask and discuss with students from previous years. You cannot look up the solution online.

Cover page for answers.pdf
CSE512 Fall 2018 - Machine Learning - Homework 7

Your Name: Sriram Reddy Kalluri

Solar ID: 111878857

NetID email address: skalluri@cs.stonybrook.edu and sriram.kalluri@stonybrook.edu

Names of people whom you discussed the homework with: