Assignment 4

Homework assignments will be done individually: each student must hand in their own answers. Use of partial or entire solutions obtained from others or online is strictly prohibited. Electronic submission on Canvas is mandatory.

1. **Nearest Neighbors** (10 points) Implement a basic k-NN model on the yeast dataset. The task is to predict the compartment in a cell that a yeast protein will localize to based on properties of its sequence. Apply cross-validation and report your best performance

   Solution : Kindly refer the attached file name **Assignment_#4_Q1_Akshay_Rane.ipynb** for the solution.

2. **Clustering** (5 points) Suppose we clustered a set of N data points using two different clustering algorithms: k-means and Gaussian mixtures. In both cases we obtained 5 clusters and in both cases the centers of the clusters are exactly the same. Can a few (say 3) points that are assigned to different clusters in the kmeans solution be assigned to the same cluster in the Gaussian mixture solution? If no, explain. If so, sketch an example or explain in 1-2 sentences.

   Solution:

   Gaussian Mixture Model(GMM) and K-Means algorithm differ in the fundamental evaluation of clusters. K-Means does hard clustering whereas GMM does soft clustering. K-Means assign every data point to a cluster that is formed by train data whereas GMM will evaluate the probability of data point to each cluster. This also depends on the parameters used for GMM. If the GMM model uses Gaussian's with constant variance then GMM produces clusters similar to K-Means. Therefore all the test data points that are assigned to different clusters in the K-means solution be assigned to the same cluster in the Gaussian mixture solution.

   This depends on how the clusters are formed in Hard Clustering in K-means and soft clustering in GMM and how the centers are calculated. If the assignments in GMM based clusters is done on the maximum probability then, all the test data points that are assigned to different clusters in the K-means solution be assigned to the same cluster in the Gaussian mixture solution.

   This is only possible if the data points are easily separable and GMM has constant variance. If the mean vectors obtained from GMM are used then all the test data points that are assigned to different clusters in the K-means solution would not be assigned to the same cluster in the Gaussian mixture solution.

(a)(b)

3. **Bayesian Networks** (10 points) Do the following statements hold in each of the above networks ? Please explain your reasoning

   - $A \perp C | B, D$
   - $B \perp D | A, C$

   Solution:

   Kindly refer **Assignment_#4_Q3_Akshay_Rane.pdf** for this solution.

Figure 1: Scatter plot of datasets and the initialized centers of 3 clusters

4. **K-means** (30 points) Given the matrix X whose rows represent different data points, you are asked to perform a k-means clustering on this dataset using the Euclidean distance as the distance function. Here k is chosen as 3. The Euclidean distance d between a vector x and a vector y both in $\mathcal{R}^d$ is defined as $d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$. All data in X were plotted in Figure 1. The centers of 3 clusters were initialized as $\mu_1 = (6.2, 3.2)$(red), $\mu_2 = (6.6, 3.7)$(green), $\mu_3 = (6.5, 3.0)$(blue).

   Points Classification and Labelling:

Points	x1	x2
P1	5.5	4.2
P2	5.1	3.8
P3	4.7	3.2
P4	4.9	3.1
P5	5.0	3.0
P6	4.6	2.9
P7	5.9	3.2
P8	6.0	3.0
P9	6.2	2.8
P10	6.7	3.1
μ_{red}	6.2	3.2
μ_{blue}	6.5	3.0
μ_{green}	6.6	3.7

   (a) Whats the center of the first cluster (red) after one iteration? (Answer in the format of $[x_1, x_2]$, round your results to three decimal places)

   Solution: The center for first cluster after one iteration is
   $|5.171, 3.171|$

   (b) Whats the center of the second cluster (green) after two iteration?

   Solution: The center for second cluster after two iteration is
   $|5.300, 4.000|$

   (c) Whats the center of the third cluster (blue) when the clustering converges?

   Solution:

   The center for third cluster when the clustering converges is
   $|6.200, 3.025|$

   (d) How many iterations are required for the clusters to converge?

   Solution:

   The clustering converges in 2 iterations. The 3rd iteration verifies the distribution.

Note : Kindly refer the attached file name **Assignment_#4_Q4_Akshay_Rane.xls** for the explanation.

5. **Expectation Maximization (EM)** (25 points) In this question you will implement the EM algorithm for Gaussian Mixture Models. A good read on gaussian mixture EM can be found at this link. A sample dataset for this problem can be downloaded in canvas files. For this problem:

- n is the number of training points
- f is the number of features
- k is the number of gaussians
- X is an $n \times f$ matrix of training data
- w is an $n \times k$ matrix of membership weights. $w(i, j)$ is the probability that x_i was generated by gaussian j
- π is a $k \times 1$ vector of mixture weights (gaussian prior probabilities). π_i is the prior probability that any point belongs to cluster i
- μ is a $k \times f$ matrix containing the means of each gaussian
- Σ is an $f \times f \times k$ tensor of covariance matrices. $\Sigma(:, :, i)$ is the covariance of gaussian i

(a) **Expectation**: Complete the function $[w] = \text{Expectation}(X, k, \pi, \mu, \Sigma)$. This function takes in a set of parameters of a gaussian mixture model, and outputs the membership weights of each data point

(b) **Maximization of Means**: Complete the function $[\mu] = \text{MaximizeMean}(X, k, w)$. This function takes in the training data along with the membership weights, and calculates the new maximum likelihood mean for each gaussian.

(c) **Maximization of Covariances**: Complete the function $[\Sigma] = \text{MaximizeCovariance}(X, k, w, \mu)$. This function takes in the training data along with membership weights and means for each gaussian, and calculates the new maximum likelihood covariance for each gaussian

(d) **Maximization of Mixture Weights** : Complete the function $[\pi] = \text{MaximizeMixtures}(k, w)$. This function takes in the membership weights, and calculates the new maximum likelihood mixture weight for each gaussian.

(e) **EM**: Put everything together and implement the function $[\pi, \mu, \Sigma] = \text{EM}(X, k, \pi_0, \mu_0, \Sigma_0, \text{nIter})$. This function runs the EM algorithm for nIter steps and returns the parameters of the underlying GMM. Note: Since this code will call your other functions, make sure that they are correct first. A good way to test your EM function offline is to check that the log likelihood, $\log P(X|\pi, \mu, \Sigma)$ is increasing for each iteration of EM.

Solution : Kindly refer the attached file name **Assignment_#4_Q5_Akshay_Rane.ipynb** for the solution.

6. **Convolutional Neural Networks** (20 points) Develop a Convolutional Neural Network (CNN) model to predict a handwritten digit images into 0 to 9 (You can use Keras or other packages). The pickled file represents a tuple of 3 lists: the training set, the validation set and the testing set. Each of the three lists is a pair formed from a list of images and a list of class labels for each of the images. An image is represented as numpy 1-dimensional array of 784 (28 x 28) float values between 0 and 1 (0 stands for black, 1 for white). The labels are numbers between 0 and 9 indicating which digit the image represents. The code block below shows how to load the dataset.

```
import cPickle, gzip, numpy

# Load the dataset
f = gzip.open('mnist.pkl.gz', 'rb')
train_set, valid_set, test_set = cPickle.load(f)
f.close()
```

- Choose the proper activation and loss function.
- Plot the train, validation, and test errors as a function of the epochs.
- Report the best accuracy on the validation and test data sets. Discuss the parameter choices such as the filter size, number of filters etc.
- Apply early stopping using the validation set to avoid overfitting.
- Give a brief description of your observations.
- Does pooling make the model more or less sensitive to small changes in the input images? Why? By small changes, we mean moving the input images to the left or right, rotating them slightly etc.

Solution : Kindly refer the attached file name **Assignment_#4_Q6_Akshay_Rane.ipynb** for the solution.

A convolution layer (not dense layer) connected to the input layer. Each convolution layer applies different filters. Typically hundreds or thousands filters used. The results of filters are concatenated. A pooling layer is used to sub-sample the result of convolution layer.

The kernel size used is 2. The output layer has 10 neurons as there are 10 classes of numbers.

Observation:

As per the plots the training accuracy increases from each epoch. The change in accuracy and loss for training is high in the first few epoch where as in the next epoch the change is very low. The same can be observed with validation and testing data-points.

Change in image such as rotation would not change the pooling process as max pooling fetches the maximum value from the frame and if the images are changed left or right the max value will always remain the same.