Machine Learning with Networks *ECEN765.600, Fall 2016*

Programming Assignment #3: SVM & GMM

Due date: Bonus Assignments (December 13th, 2017)

"On my honor, as an Aggie, I have neither given nor received unauthorized aid on this academic work."

Signature:

Name:

0. Reading Assignments: The relevant content in BRML Chapters 11, 17, 20.

1. Support Vector Machines (50pts):

For this assignment, I suggest you use one of the SVM implementations available at `http://www.support-vector-machines.org/SVM_soft.html`. There are also MATLAB SVM implementations that you can use, including the toolboxes in BRMLToolBox (`http://web4.cs.ucl.ac.uk/staff/d.barber/pmwiki/pmwiki.php?n=Brml.Software`) and PMTK3 (`https://github.com/probml/pmtk3`) with discussions (`https://code.google.com/p/pmtk3/`). You might need to transform the data format.

Using the same binary data set in the last programming assignment, train the following SVMs (using just the training data): a linear SVM, and an RBF SVM (for (**extra credit** 10pts). For the linear SVM, try different values of C ranging in $0.25, 0.5, 1, 2, 4$. For the RBF SVM, try τ values (bandwidth) of $0.25, 0.5, 1, 2, 4$. Plot error rates on both the development data and test data for the different values of C. How many support vectors are used for each model? Should this increase or decrease with C (why?)?

2. Gaussian Mixture Models (GMM) (50pts):

Take the previous data but without using the label information. Implement the EM (Expectation-Maximization) for Gaussian mixture models. You need to implement: (1) initialization; (2) the E-step; (3) the M-step. You should use the algorithm with full covariance matrices.

In the process of building the Gaussian mixture models, you will plot by iteration the data log-likelihood. A good way to debug your code, which is especially difficult for unsupervised learning algorithms, is to make sure that the incomplete data log-likelihood (lower bound function) monotonically increases.

Once you have the GMM algorithm running, take different values of $k \in \{2; 3; 4; 5; 6; 7; 8; 9; 10\}$. For each of these, you should run the GMM with 10 different initializations and choose as your final clustering the one among these 10 with the highest data log-likelihood. Plot it as a function of k. Which value of k would you choose based on these plots? Repeat this for another data set in gmm.zip.

Note: A Matlab pseudo code is provided if you would like to develop the EM algorithm by yourself.