Pattern Recognition
LVQ

Useful matlab functions:

`repmat`, `randperm`

Guidelines for lab reports:

- Always give a (short) explanation of what you are doing.
- Do not forget to include your Matlab programs. Present and discuss the results of your programs, be it a number, a matrix or an image.
- Put large pieces of Matlab code in an appendix.
- One should be able to understand plots independently, be sure to label axes, add a legend for colors, etc.
- Refer to all plots, tables, code blocks, etc. in your report.
- If you print gray-scale make sure the colors used in the plots are distinguishable.

Assignment 1: *Supervised Learning: LVQ1.*

The files `data_lvq_A.mat` and `data_lvq_B.mat` each contain 100 two-dimensional feature vectors, belonging to class A and B, respectively. Your task is to build a LVQ1 classifier using this training data.

1. First investigate this data set by making a scatter plot of the two classes, where the classes are distinguishable. How many prototypes per class (at least) would be appropriate to use for this data set?

hint: Take a look at the guidelines concerning plots.

2. Implement the LVQ1 algorithm, see lecture slides. Use a constant learning rate $\eta = 0.01$. Determine the winner by minimal squared Euclidean distance.

   Think about how to choose the initial prototypes, discuss the method you used. After each epoch, determine the number of misclassified training examples. The training error E is defined as the number of misclassified training examples divided by the total number of data points. Stop training when E becomes approximately constant.

3. Consider the following cases:

a) 1 prototype for class A and 1 prototype for class B,

b) 1 prototype for class A and 2 prototypes for class B,

c) 2 prototypes for class A and 1 prototype for class B,

d) 2 prototypes for class A and 2 prototypes for class B.

For each of these cases

a) Plot E as a function of the number of epochs. Analyze each of these cases.

b) Create a 2D-plot which shows the LVQ1 prototypes and the train data.

hint: You will get benefits from your K nearest neighbor implementation from week 3.

4. (**bonus**) The learning rate η is the key factor in LVQ. You will get bonus points if you observe and report how the behavior depends on the learning rate.

Assignment 2: *Cross validation*

1. For case 3c in assignment 1, apply 10-fold cross validation for the estimation of the classification error.

   This means, divide the training set in 10 equal subsets. Use 9 subsets to train an LVQ1 classifier and then use the remaining 1 subset to test it and determine the classification error. Use each of the subsets as a test set once, with the other 9 subsets as training sets.

2. Compute the test error, i.e. the mean of the 10 values of the classification error computed in this way.

Assignment 3: *Relevance LVQ.*

1. For case 3c in assignment 1, implement the relevance LVQ algorithm with global relevances (i.e. relevances that apply to the whole feature space and not just to one prototype or a class). Use 0.5 as initial value of the global relevances λ_1 and λ_2. Stop the training when the training error becomes approximately constant.

hint: Read the lecture slides and use an appropriate value of the relevance learning rate η_λ.

2. Show the final relevances.

3. Create a 2D plot which shows the LVQ prototypes.

4. Plot the training error E and the relevances of the two features as a function of the number of epochs.

5. Use 10-fold cross validation for the estimation of the classification error.

6. Compute the test error as determined by 10-fold estimation.