Assignment 4

Machine Learning, Summer term 2013, Ulrike von Luxburg

Solutions due by May 6

Read `prepare04.pdf` for an example of linear discriminant analysis(LDA) in MATLAB.

Exercise 1 (Digit classification with LDA, 4 points) We want to use LDA for written digit classification on the USPS dataset from Assignment 1.

1. Use `ClassificationDiscriminant.fit` to train a LDA classifier for digit 4 versus 9. Use `imagesc` (see Assignment 1) to plot the learned weights in a 16x16 image. Here you do not need to change the colormap.
2. Use your learned weights(`.Linear`) and the constant weight(`.Const`) to predict labels of the test data for digits 4 and 9. Report the 0-1 loss of your prediction.

Multiclass classification: So far, all classification algorithms you learned were designed to deal with two class (binary) classification problems. In many real world applications(e.g. digit recognition), our data has several classes. A common approach to solve such problems is to build a multiclass classifier from several binary classifiers. Assume we have k classes $C_1, ..., C_k$. Usually we follow one of these strategies:

- One vs. All: A single classifier is trained per class to distinguish that class from all other classes. Prediction is then performed by predicting using each binary classifier, and choosing the prediction with the highest confidence score (in LDA, $w_i x + b_i$ is the score for class i).
- One vs. One: For each pair of classes we construct a binary classifier ($c(c-1)/2$ classifiers in total). Usually, classification of an unknown pattern is done according to the maximum voting , where each of $c(c-1)/2$ classifiers votes for one class.

Exercise 2 (Multiclass classification with LDA, 5 points) Implement both strategies (One vs. All, One vs. One) for LDA classifier to classify digits 1, 2, 3 and 4 from USPS dataset. Report the test error for each strategy.

Exercise 3 (Complexity of multiclass classification, 2 points) You have a multiclass classification problem with n training points and k classes. Each class has the same number of training points. A binary classifier `myClassifier` is given which requires $c(m_1^2 + m_2^2)$ operations to learn the classifier. Here m_i is the number of training points in class i and c is a constant.

- Compare the training time of multiclass classification with *One vs. All* and *One vs. One* strategies using `myClassifier`. Which one is faster?
- Assume our binary classifier `myClassifier` requires $c(m_1 + m_2)$ operations for training. Which of two methods for multiclass classification would run faster?

Cross validation (m-fold): In many machine learning algorithms, we need to choose parameters of our learning algorithm. The regularization parameter λ in ridge regression, or the number of neighbors k in kNN classification are examples of such parameters. In *m-fold cross-validation* the original sample is randomly partitioned into m equal sized subsamples. Of the m subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $m-1$ subsamples are used as training data. The cross-validation process is then repeated m times (the folds), with each of the m subsamples used exactly once as the validation data. The m results from the folds then can be combined to produce a single estimation.

1. Split the data into n equal-sized groups.
2. For i=1 to m,
   (a) Select group i to be the validation set and all other (m-1) groups to be the training set.
   (b) Train the model on the training set and evaluate on the validation set.
3. Select the parameter which performed better in these m folds.

Exercise 4 (Parameter selection by the training error, 1 point) The naive approach for choosing the learning parameter is to select the one which minimizes the training error. Describe why this is not a good idea in ridge regression and also in kNN classification.

Exercise 5 (Cross validation in ridge regression, 3 points) Use your ridge regression code (or the one available in the course webpage) and the training data in `dataRidge.mat` from Assignment 3. Choose the regularization parameter from $\lambda \in \{2^{-i} | i = -15, -14, ..., 7, 8\}$. Use 10-fold cross validation to select the regularization parameter with best performance (test error with L2 loss). To partition the data in MATLAB, first call `c = cvpartition(n,'kfold',10);`. Then use `c.training(i)` and `c.test(i)` to access the training and validation data in i'th fold.

Exercise 6 (Your own exam questions, 6 points)
In this exercise, everybody is supposed to come up with suggestions for exam questions. This is a good way to recap the material and to study for the exam itself.
Put yourself in our place! We don't want to ask stupid questions but "nice questions". In general, written exams contain three types of questions:

- Questions which are just about **reproducing** knowledge.
- Questions for testing whether the person **understands** the concepts and can apply them to simple situations.
- Questions that requires to **transfer** knowledge to new situations.

Your task is now to design at least three exam questions, one of each type. The more questions you come up with, the better. Please enter your questions to the following webpage:

`http://www2.informatik.uni-hamburg.de/ML/ML-AL-2013/questions.php`

At the end of the course, these questions can help everybody to prepare for the exam. So take your time to invent good questions!