Assignment 2

Machine Learning, Summer term 2013, Ulrike von Luxburg

Solutions due by April 22

You can download Matlab functions `mixGaussian1d, mixGaussian2d, multimodal1d` and the dataset `20Newsgroup.mat` from the course web page.

Exercise 1 (Text classification, 3 points) In this exercise, we apply the kNN classifier for automatically classifying user posts in a newsgroup. The 20 Newsgroups dataset is a collection of approximately 19,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. The data is organized into 20 different newsgroups (see `20NgClasses.txt`).
To simplify the work, we use a bag of words representation of documents. In this representation, the order of words are ignored and we only count the appearance of a word in a document. Your training data is a 11269 x 53975 matrix, where each row corresponds to a document and element (i, j) shows how often the word j occurs in the document i. You do not need to know the corresponding word for each feature, but they are available in `vocabulary.txt`.
We want to classify documents in class 6 (`comp.windows.x`) and 8 (`rec.autos`).

1. Load the dataset and prepare the training set for classes 6 and 8:

   ```
   load('20Newsgroup.mat');
   trList = find(y_train==6 | y_train==8);
   x_train_6_8 = x_train(trList,:);
   y_train_6_8 = y_train(trList);
   ```

2. Do the same for the test data.

3. Note that the train and the test data have different feature sizes. The reason is the existence of words in the test set which never appeared in the training set. You can ignore those features by simply cropping them: `x_test_cropped = x_test(:,1:size(x_train,2));`.

4. Plot the training and the test error for `k=1,3,5,7,10,15,20`.

Exercise 2 (Bayes classifier, 8 points) In this exercise you will do the maximum likelihood and the Bayes classification by hand. First generate a simple synthetic dataset

```
[X,Y] = mixGaussian1d(1000,0.5,0.5,0,6,1,2);
```

1. Read the Matlab code in `mixGaussian1d.m` and describe its functionality in words.

2. Plot the points `[X,Y]` in 2d (this reflects the joint distribution $P(X, Y)$).

3. As your data is 1-dimensional, it is hard to interpret this plot. A better way to display the data is to plot the marginal $P(X)$. Estimate the density by its histogram:

   ```
   figure (1); clf; hold all;
   [countC,binsX] = hist(X,30);
   PX = countC/size(X,1);
   plot(binsX,PX,'.-');
   ```

4. Plot the class conditional distributions $P(X|Y = 1)$ and $P(X|Y = 2)$ with different colors in the same figure. To make histogram bins compatible with each other, pass `binsX` as an argument to `hist`.

Web page: `http://www.informatik.uni-hamburg.de/ML/contents/people/luxburg/teaching/2013-ss-vorlesung-ml/`
Login: "machine", Password: "learning"

5. Estimate priors $P(Y = 1)$ and $P(Y = 2)$.

6. Based only on looking your plots, predict the labels for $X' = \{0, 1, 2, 2.5, 3, 4, 5\}$ using maximum likelihood and Bayes methods. For the Bayes method, you need to draw a new plot that makes the prediction possible.

7. Repeat the exercise with `[X,Y] = mixGaussian1d(1000,0.1,0.6,0,6,1,2);`.

8. Repeat the exercise with `[X,Y] = multimodal1d(3000);`. Predict the labels for

$$X' = \{0, 1, 2, 3, 4, 5, 6, 7\}$$

from your plots using maximum likelihood and Bayes methods. Increase the number of bins for histograms if it is necessary.

Exercise 3 (Letter classification, 2 points) Assume you are going to write a two-class classifier which distinguishes the letter j from the letter k. The frequency of the letter j in a typical English text is 0.015 and for the letter k is 0.045.

(a) Assume your input data are only from letters j and k. Will you buy a classifier with probability of error 0.3? What would be the lowest probability of error if you do the classification without even looking at your input data?

(b) Assume your input data are all English letters and you want to separate the letter j from others. Will you buy a classifier with probability of error 0.02?

Exercise 4 (Decision boundary, 4 points) Generate a dataset by calling

`[X Y] = mixGaussian2d(100,0.4,0.6);`

X is a set of 2d points belonging to two different classes with labels in Y. As a prior knowledge, you know that the density of each class is a Gaussian.

(a) Compute the sample mean and sample variance for each class.

(b) Using the estimated mean and variance from part (a), find the analytic decision boundary for the Bayes classifier (see Section 2.6.3 from the pattern recognition book by Duda et al. available in the course web page).

Exercise 5 (Types of errors, 6 point) You are asked to write a spam filtering algorithm. Two types of error can occur during your classification: A false positive occurs when spam filtering wrongly classify a legitimate email message as spam. While most anti-spam tactics can block or filter a high percentage of unwanted emails, doing so without creating significant false-positive results is a much more demanding task.

A false negative occurs when a spam email is not detected as spam, but is classified as non-spam. A low number of false negatives is an indicator of the efficiency of spam filtering.

- Your algorithm finds 85% of spams, and miss-classify in 5% of legitimate emails. Additionally you know that about 60% of your incoming emails are spam. What are false positive, false negative and average error of you classifier?
- Find a classification algorithm with false positive rate 0. Find a classification algorithm with false negative rate 0.
- In Exercise 2, assume Class 1 represents non-spams and Class 2 represents spams. In Part 6, the labels are given as $Y' = \{1, 1, 2, 1, 2, 2, 2\}$. Which entries of X' lead to a false positive error and which ones to a false negative error in the Bayes classification?
- In Part 4 of Exercise 2, sketch the approximate Bayesian decision boundary by hand with respect to the following loss functions
  - 0-1 loss
  - Unsymmetric lost: $l(spam, non - spam) = 1, l(non - spam, spam) = 100$.