CS 6780: Advanced Machine Learning
Homework 2

Due date: 12pm. **Oct 8, 2010**
(Printed submission.)

Note:
1. Write your name and net-id in BIG letters on the first page. Do not write anything else on the first-page of the homework.
2. For derivation questions, you need to show all the necessary steps. Just writing the answers won't do.
3. For data analysis / programming parts, you need to submit: (a) the specific values and plots requested, (b) explanation in the text if needed (don't give us the code printouts only!), and (c) also provide the code in the appendix.

1 Nearest Neighbor

(a) If the training data comes in pairs (x_i, y_i), $y_i \in \{0, 1\}$ then the distance weighted k nearest neighbor rule classifies a new point $x_* \neq x_i \forall i$ as:

$$\begin{aligned} S_* &= \{j | x_j \text{is one of the } k \text{ points with smallest } ||x_j - x_*||\} \\ \hat{y} &= \frac{\sum_{i \in S_*} w_i y_i}{\sum_{i \in S_*} w_i}, \text{ where } w_i = \frac{1}{||x_i - x_*||^2} \end{aligned}$$

If a learning algorithm can be expressed solely in terms of operations on inner products between the data, we can use the kernel trick, i.e. replacing all inner products with evaluations of a kernel function: $k : \mathcal{X} \times \mathcal{X} \rightarrow R$ which is implicitly computing the inner product of its two arguments as if they were first mapped in a potentially high dimensional (and "richer") space.

Reformulate the distance weighted k nearest neighbor classification rule so that it can use a kernel. (8 points)

(b) My bank's machines cannot read some checks because the software has trouble distinguishing between the digits 4 and 9. Your task is to use the data in the files (`4vs9-train.dat`,`4vs9-train.lab`, `4vs9-test.dat`,`4vs9-test.lab`) to come up with a good distance weighted classification rule. The data are digitized images of 4 and 9 written by different people. Each feature is a pixel and its value is the intensity of that pixel. In the .dat files, each row is a vector of features whose correct label is in the corresponding row of the .lab file, 0 for images of 4 and 1 for 9. Pick the nearest neighbors from the train files and use the test files to evaluate your classification rule. You have two degrees of freedom, specifying k, and specifying a kernel. Report the things your tried, the best combination of k and kernel you found, and the value of $||\hat{y} - y||$ that was attained for that combination. You can use `knn.m` as a starting point. Valid kernels include the linear: $k(x, x') = x \cdot x'$, the Gaussian $k(x, x') = \exp(-\gamma||x - x'||^2)$, the polynonial $k(x, x') = (x \cdot x' + c)^d$, and many others. (15 points[1])

2 Support Vector Machines and Kernels

2.1 Kernels

Prove that $K(x_i, x_j) = \exp(-||x_i - x_j||^2)$ is a valid kernel.

You can only use the identities given below:

If $k_1(x, x')$ and $k_2(x, x')$ are valid kernels, then $k(x, x')$ is also a valid kernel:

$$\begin{aligned} k(x, x') &= ck_1(x, x'), \quad \text{for } c > 0 \\ k(x, x') &= k_1(x, x') + k_2(x, x') \\ k(x, x') &= k_1(x, x')k_2(x, x') \\ k(x, x') &= \exp(k_1(x, x')) \\ k(x, x') &= f(x)k_1(x, x')f(x') \end{aligned}$$

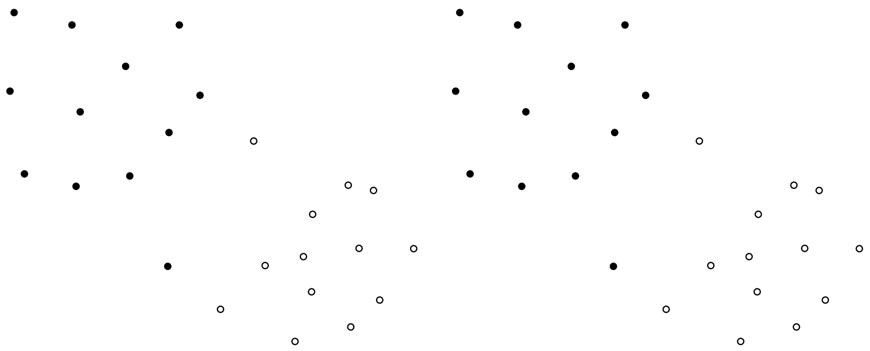
where $f(x)$ is any function of x.

(6 points)

[1] You would be evaluated based on how accurate you got it to work, the choices made (with explanation).

2.2 Non-separable case

(SVM). Draw the decision boundary (solid line) and the margins (dashed line) for a low value of C in figure below (left). Draw the same for a high value of C (right). (Here C is the parameter of the SVM in the non-separable case.) (4 points)



(a) low value of C (b) large value of C

2.3 Implementation

SVM^{perf} is a very fast SVM solver for linear SVMs that can optimize many performance measures apart from accuracy (hence the name "perf"). Download it from:
`http://www.cs.cornell.edu/People/tj/svm_light/svm_perf.html`
and download the astrophysics data from the course website. These represent abstracts of scientific papers from physics classified by whether they are about astrophysics. The data have 99757 features and they are split into 29882 training examples and 32487 test examples. Use 5-fold cross validation on the training set to pick a good value for the C parameter of the SVM and report the best test accuracy and the C for which it was attained.[2] (12 points)

[2] Hint: (a) Search for C on a log-scale. (b) It might be easier to run it on Linux.

3 Eigen-Grads

In 1990's, people came up with EigenFaces. In this part, we will try to apply it to faces of grad students.
(a) Download the dataset `face-data.zip`, and you can use the Matlab starter code `faces.m` in `face-code.zip` for this problem. Choose the *minimum* number of dimensions k needed so that you loose no more than 10 percent of the variance in the data. (Plot fraction error as a function of k.) (6 points)
(b) Show the k eigenfaces (the basis vectors) as an image. (4 points)

(c) Divide the dataset into two categories: *male* and *female* faces. Train a logistic classifier (you can use `glmfit` in Matlab) using the projected data into k dimensions (use k and the basis obtained from the previous part). Using Leave-one-out-cross-validation (LOOCV), report the following errors: (i) $E_1 = \frac{\text{total mistakes}}{\text{total}}$, and (ii) $E_2 = 0.5\frac{\text{mistakes class1}}{\text{class1 total}} + 0.5\frac{\text{mistakes class2}}{\text{class2 total}}$.
Now, vary k (and re-run the PCA on the 29 images) in order to improve the performance (as measured by 2nd definition of error.) With this optimal value of k, report the LOOCV errors (call them $\bar{E}_1$ and $\bar{E}_2$.

You have now build a "image-based gender classifer," and now imagine you were writing a research paper on this. Would you call the final LOOCV error $\bar{E}_2$ as "test error" or "training error" or something else in your research paper? Why? What would you call the error E_2? (20 points)

Bonus question: Is the accuracy good enough? If not, suggest two methods that you think would help fix the problem (write a phrase for each).

4 Mixture of Two Linear Regressions

Often a single generalized linear model does not fit the data well. In such cases, we can fit a set of k generalized linear models to our data. In this question, we will consider a mixture of two linear regressions. Suppose that we have i.i.d. training data in the form $(x^{(i)}, y^{(i)})$, $i = 1, \ldots, m$, $x^{(i)} \in \mathbb{R}^n$, $y^{(i)} \in \mathbb{R}$ and we assume that each $y^{(i)}$ is generated as a linear combination of $x^{(i)}$ with either w_0 or w_1, depending on the value of an unobserved Bernoulli variable $z^{(i)} \in \{0, 1\}$, plus noise. Here, we also model $z^{(i)}$ as a function of input data $x^{(i)}$. Formally,

$$\begin{aligned} P(z^{(i)} = 1 | x^{(i)}; \theta) &= \frac{1}{1 + \exp(-\theta^T x^{(i)})} \\ P(y^{(i)} | z^{(i)}, x^{(i)}; w_0, w_1) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y^{(i)} - w_{z^{(i)}}^T x^{(i)})^2}{2\sigma^2} \right) \end{aligned}$$

Explain why it is hard to estimate the parameters w_0, w_1, θ in a closed form. Derive an EM algorithm for this model to estimate the parameters w_0, w_1, θ.

Hints: (a) Our data are x and y, so in the E step you should compute the distribution of the unobserved variable z given *both*, not just x as in the case of the mixture of gaussians. (b) In the M step, you may discover that one of the updates still cannot be obtained in closed form. In such case, explain if/how we can overcome this. We look for the simplest correct explanation. (25 points)