

Pattern Recognition and Similarity Search

Homework 4

Due 11/15/2004

1. Consider the problem of separating N data points into positive and negative examples using a linear separator. Clearly, this can always be done for $N = 2$ points on a line of dimension $d = 1$, regardless of how the points are labelled or where they are located (unless the points are in the same place).
 - (a) Show that it can always be done for $N = 3$ points on a plane of dimension $d = 2$, unless they are collinear.
 - (b) Show that it cannot always be done for $N = 4$ points on a plane of dimension $d = 2$.
 - (c) Show that it can always be done for $N = 4$ points in a space of dimension $d = 3$, unless they are coplanar.
 - (d) Show that it cannot always be done for $N = 5$ points in a space of dimension $d = 3$.
 - (e) **[BONUS]** The ambitious student may wish to prove that N points in general position (but not $N + 1$) are linearly separable in space of dimension $N - 1$. From this it follows that the **VC dimension** of linear half spaces in dimension $N - 1$ is N .
2. In this problem I want you to learn how to use some of the standard software and data that are available on the Web. You will try out support vector machines on some benchmark datasets. On the website www.kernel-machines.org, you will find public domain software for the SVM. Download one or more of these programs and compile (if necessary) on your machine. Read the documentation and learn how to use the various options that the software provides.
 - (a) Run the SVM on the data ellipse.dat on the course webpage. (You may have to change the data to fit the format required by your program.) Try four different kernels: a linear kernel $k(x, y) = x^T y$, a polynomial kernel $k(x, y) = (1 + x^T y)^2$, and a Gaussian kernel $k(x, y) = \exp(-0.5(x - y)^2 / \sigma)$, where σ is set to 1.0 and to 3.0. How many support vectors do you obtain in each case? What is your error rate on the training set in each case?
 - (b) Download two data sets for binary classification from the web. There are several places where you can find public domain data:
 - (Kernel Machines): <http://www.kernel-machines.org>
 - (Statlib): <http://lib.stat.cmu.edu/~datasets>
 - (Delve): <http://www.cs.toronto.edu/~delve>
 - (UC Irvine): <http://www.ics.uci.edu/~mlearn/MLRepository.html>Try the SVM on your datasets, evaluating the performance on a held-out test set.
3. In the last question, you are asked to compare the nearest-neighbor search performances between sequential search and a fast indexing method of your choice on two datasets. There are many implementations of indexing methods that are publicly available. Some of them are listed below:
 - Simple KD-tree from the Auton project (<http://www.autonlab.org/autonweb/showSoftware/169>)
 - KD-tree in matlab (<http://www.technion.ac.il/~gshec/software.html>)

- Various indexing methods in the R-tree portal (<http://www.rtreeportal.org/>)
- M-tree (<http://www-db.deis.unibo.it/Mtree/>)

Choose your favorite software and measure the average search time on two datasets `highD1.dat` and `highD2.dat` for the query dataset `query.dat`, all provided in the course website. Use either l_1 or l_2 distance. Also implement a sequential search method and measure the average search time as well. In order to minimize the I/O effect, you may assume that the dataset is entirely inside the main memory. Comment on your results.