Machine Learning - Assignment 3

Posted Thursday, October 15, 2009
Due Friday, October 23, 2009

1. [25 points] **Kernels** In this problem, we consider constructing new kernels by combining existing kernels. Recall that for some function $K(\mathbf{x}, \mathbf{z})$ to be a kernel, we need to be able to write it as a dot product of vectors from some high-dimensional feature space:

$$K(x, z) = \phi(\mathbf{x})^T \phi(\mathbf{z})$$

Mercer's theorem gives a necessary and sufficient condition for a function K to be a kernel: its corresponding kernel matrix has to be symmetric and positive semidefinite.

Suppose that $K_1(\mathbf{x}, \mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z})$ are kernels over $\Re^n \times \Re^n$. For each of the cases below, state whether K is also a kernel. If it is, prove it. If it is not, give a counterexample. You can use either Mercer's theorem, or the definition of a kernel as needed

   (a) $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z}) + bK_2(\mathbf{x}, \mathbf{z})$, where $a, b > 0$ are real numbers

   (b) $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z}) - bK_2(\mathbf{x}, \mathbf{z})$, where $a, b > 0$ are real numbers

   (c) $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z})K_2(\mathbf{x}, \mathbf{z})$

   (d) $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$ where $f : \Re^n \rightarrow \Re$ is a real-valued function

   (e) $K(\mathbf{x}, \mathbf{z}) = p(\mathbf{x})p(\mathbf{z})$ where p is a probability density function

2. [25 points] **SVM regression**

In class we discussed the use of support vector machines for classification. In this exercise, we study how they can be used for regression. In this case, we will be interested in minimizing the absolute error on the data points. More precisely, we will use the following loss function:

$$\begin{aligned} J_\epsilon &= \sum_{i=1}^{m} J_\epsilon(\mathbf{x_i}), \text{ where} \\ J_\epsilon(\mathbf{x_i}) &= \begin{cases} 0 & \text{if } |y_i - (\mathbf{w} \cdot \mathbf{x_i} + w_0)| \leq \epsilon \\ |y_i - (\mathbf{w} \cdot \mathbf{x_i} + w_0)| - \epsilon & \text{otherwise} \end{cases} \end{aligned}$$

This is called the ϵ-insensitive loss and allows for points to be mislabeled.

   (a) [5 points] Write the optimization problem that we will want to solve in this case (making sure you account for the fact that we want to have linear constraints)

   (b) [5 points] Give a computational reason why we are using the ϵ-insensitive loss rather than the squared error, as was more common before

   (c) [5 points] Write the Lagrange multiplier for this problem and state the dual problem

(d) [5 points] Show the KKT conditions for this problem and explain the form of the solution

(e) [5 points] Show the optimization problem and the solution obtained using kernels.

3. [20 points] **A midterm preparation question**

   For each of the learning problems outlined below, specify what is the best learning algorithm to use and why. Note that you should give *one* algorithm for each problem, even if there are several correct answers.

   (a) [5 points] You have about 1000 training examples in a 6-dimensional continuous feature space. You only expect to be asked to classify 100 test examples.

   (b) [5 points] You are going to develop a classifier to recommend which children should be assigned to special education classes in kindergaren. The classifier has to be justified to the board of education before it is implemented.

   (c) [5 points] You are working for a huge retailing company. You are trying to predict whether customer X will like a particular item, as a function of the input which is a vector of 1 million bits specifying whether each of the other customers liked the item. You will train a classifier on a very large data set of items, where the inputs are everyone elses preferences for that item, and the output is customer Xs preference for that item. The classifier will have to be updated frequently and efficiently as new data comes in.

   (d) [5 points] You are working for an oil company which is trying to decide where to drill. You have 40 attributes, both discrete and continuous, that describe a plot of land. Some of these attributes are noisy. For previous sites, you know whether they contained oil or not, but you only have data about 50 such sites.

4. [15 points] **Perceptrons**

   For each of the functions below, explain whether it can be represented using a *single* perceptron. If so, show the weights that would correctly represent the function. If no, justify why not.

   (a) $x_1 \vee x_2 \vee \ldots \vee x_n$

   (b) $\neg(x_1 \wedge x_2 \wedge \ldots \wedge x_n)$

   (c) the function which is true when exactly 2 of n Boolean variables $x_1, \ldots x_n$ are true.

5. [10 points] **Representational power and decision boundaries**

   Recall that a decision stump is a decision tree with just one node.

   (a) [5 points] Draw a small two-dimensional data set (no more than 10 points) which cannot be separated by a decision stump, but can be separated by a linear classifier

   (b) [5 points] Draw a small two-dimensional data set that cannot be separated by a linear classifier, but can be separated by a decision tree (regardless of the depth of the tree).

   (c) [5 points] Suppose you have data with 2 real-valued attributes. Is there any relationship between the decision boundary of a 1-nearest neighbor classier and that of a decision tree (assuming no pruning)? Justify your answer.