CS 5350/6350: Machine Learining Fall 2016

Homework 5

Handed out: Nov 2, 2016
Due date: Nov 15, 2016

General Instructions

- You are welcome to talk to other members of the class about the homework. I am more concerned that you understand the underlying concepts. However, you should write down your own solution. Please keep the class collaboration policy in mind.
- Feel free discuss the homework with the instructor or the TAs.
- Your written solutions should be brief and clear. You need to show your work, not just the final answer, but you do *not* need to write it in gory detail. Your assignment should be **no more than 10 pages**. Every extra page will cost a point.
- Handwritten solutions will not be accepted.
- The homework is due by midnight of the due date. Please submit the homework on Canvas.
- Some questions are marked **For 6350 students**. Students who are registered for CS 6350 should do these questions. Of course, if you are registered for CS 5350, you are welcome to do the question too, but you will not get any credit for it.

Note

Do not just put down an answer. We want an explanation. No points will be given for just a statement of the results of a proof. You will be graded on your reasoning, not just on your final result.

Please follow good proof technique; what this means is if you make assumptions, state them. If what you do between one step and the next is not trivial or obvious, then state how and why you are doing what you are doing. A good rule of thumb is if you have to ask yourself whether what you're doing is obvious, then it's probably not obvious. Try to make the proof clean and easy to follow.

1 Warm up: Margins

1. [5 points] Suppose we want to use an SVM to learn the XOR function in two dimensions. We know that XOR is not linearly separable, so we apply a feature transformation. In order to do so, we map the input $[x_1, x_2]$ into a space consisting of two features:

x_1 and x_1x_2. Variables are boolean which takes $\{-1, 1\}$. What is the maximal margin? Draw the separating line back in original Euclidean input space.

2. [10 points] Consider the following collection of points:

Point	coordinate	label	Point	coordinate	label
x_1	$(0, 0)$	$+$	x_5	$(1, 0)$	$-$
x_2	$(0, 1)$	$+$	x_6	$(\frac{1}{2}, \frac{\sqrt{3}}{2})$	$-$
x_3	$(1, 1)$	$+$	x_7	$(\frac{3}{2}, 0)$	$-$
x_4	$(\frac{1}{2}, 0)$	$+$	x_8	$(1, \frac{1}{2})$	$-$

Table 1: A collection of points

Suppose we have three training sets comprising of subsets of these points. We have

$$D_1 = \{x_1, x_2, x_3, x_5, x_7\}$$

$$D_2 = \{x_1, x_5, x_6, x_8\}$$

$$D_3 = \{x_3, x_4, x_5, x_7\}$$

(a) [6 points] Give the maximum possible margin for D_1, D_2 and D_3.

(b) [2 points] What is the Perceptron mistake bound for these dataset. Which has the greatest Perceptron mistake bound.

(c) [2 points] Rank the datasets in terms of "ease of learning". Justify your answer.

2 Kernels

1. [15 points] If $K_1(\mathbf{x}, \mathbf{z})$ and $K_2(\mathbf{x}, \mathbf{z})$ are both valid kernel functions. In this question, you will prove that certain functions of kernels are valid kernels.

   [Hint: For both the proofs below, use the the definition of a kernel as a dot product in a high dimensional space.]

   (a) [5 points] Show that the product of two kernels is a kernel. That is, show that K in the expression below is a kernel:

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

   (b) [10 points] Show that a polynomial over a kernel that is constructed using positive coefficients is a kernel. That is, if P is any polynomial with positive coefficients, show that K below is a kernel:

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

   Hint: You may need show $K(\mathbf{x}, \mathbf{z}) = \alpha K_1(\mathbf{x}, \mathbf{z}) + \beta K_2(\mathbf{x}, \mathbf{z})$ is a valid kernel and use the conclusion in the previous question.

2. [10 points] Given two examples $\mathbf{x} \in \Re^2$ and $\mathbf{z} \in \Re^2$, let

$$K(\mathbf{x}, \mathbf{z}) = 15 \left(\mathbf{x}^T \mathbf{z}\right)^2 \exp\left(-||\mathbf{x} - \mathbf{z}||^2\right) \tag{1}$$

Prove that this is a valid kernel function.

3. (**For 6350 students**)[10 points]An valid kernel can always be expressed as inner product. Prove that the Gaussian kernel

$$K(\mathbf{x}, \mathbf{z}) = \exp(\frac{-\|\mathbf{x} - \mathbf{z}\|^2}{2\sigma^2})$$

can be written down as the inner product of an feature space with infinite dimension. Hint: You may do some expansion and then show the middle factor can be expanded as a power series.

3 Experiments

In this question, you will implement the support vector machine (SVM) and a variant of random forest which combine SVMs and decision trees.

We will use two datasets for this question:

1. `semelion handwritten digits data`: This dataset contains 1593 handwritten digits from around 80 persons were scanned, stretched in a rectangular box 16x16 in a gray scale of 256 values. Our goal is implement svm in this data set to determine whether is a number 6.

2. `madelon`: This is one of five datasets used in the NIPS 2003 feature selection challenge. There are 2000 examples in training set and 600 examples in test set.

You may reuse your code in decision tree. If **you have problems in decision tree**, please contact with TA to get help. You may use Java, Python, Matlab, C/C++ for this assignment. If you want to use a different language, you must contact the instructor first. Any other language you may want to use **MUST** run on the CADE machines.

3.1 Support Vector Machines

1. [6 points] Implement SVM in handwriting dataset with hyperparameter $C = 1$ and $\gamma_0 = 0.01$. Report the accuracy in test set and training set.

   Note:

   (a) update learning rate according to:

$$\gamma_t = \frac{\gamma_0}{1 + \gamma_0 * t/c}$$

   in this question, as well as the following questions related to SVM.

(b) Don't forget to add bias item as the first dimension of $\mathbf{x}$.

2. [8 points] Run your SVM code on the `madelon` dataset and use 5-fold cross-validation to choose suitable parameters. At least attempt 6 different values for C and 3 different values for γ_0. Report the average accuracy for each group of parameters. Report the accuracy in your test set as well as training set.

   Hint: You should try out C in exponential steps, for example, $2^1, 2^{-1}, 2^{-2}, \cdots$.

3. [6 points] Precision , recall and F_1 score are another metrics besides accuracy, which are useful if the dataset is unbalanced with respect to the positive and negative examples. To compute these quantities, you should count the number of true positives (that is, examples that your classifier predicts as positive and are truly positive), the false positives (i.e, examples that your classifier predicts as positive, but are actually labeled negative) and the false negatives (i.e., examples that are predicted as negative by your classifier, but are actually positive).

   Denote true positives, false positive and false negative as TP, FP and FN respectively. The precision (p), recall (r) and f-value F_1 are defined as:

$$\begin{aligned} p &= \frac{TP}{TP+FP} \\ r &= \frac{TP}{TP+FN} \\ F_1 &= 2\frac{p \cdot r}{p+r} \end{aligned}$$

   Give precision, recall and F_1 score for your classifiers constructed in the previous two questions.

3.2 Ensemble of decision trees

Recall that a random forest is an ensemble based on bagging and decision tree. For bagging, we draw m samples *with replacement* from the training set. According to

$$\lim_{m\to\infty}(1-\frac{1}{m})^m \to \frac{1}{e} \simeq 0.368$$

there are about 63.2 percent items may not appear that set. In random forest, we use this sampling method N times, to construct N training sets and grow N decision trees. Note that we build unpruned decision trees.

In each node, instead of using the best feature by ID3 , we choose k features randomly and than use the ID3 heuristic to find the best feature to split on. Generally, $k = log_2 d$ is a good choice where d is the number of features for our data.

Since there are N trees, there will be N predictions for each example. Generally, the final prediction is voted on by these trees. However, we would like to use SVM to combine these predictions for this question. Specifically, after growing the N decision trees, you should construct a new D consisting of transformed features. The feature transformation $\phi(x)$ is defined using the N trees as follows:

$$\phi(x) = [tree_1(x), tree_2(x), \cdots, tree_N(x)]$$

In other words, you will build an N dimensional vector consisting of the prediction (1 or -1) of each tree that you created. Thus, you have a *learned* feature transformation.

You will finally train an SVM on this new dataset D.

1. [15 points] Using the method mentioned above, construct $N = 5$ decision trees for the `handwriting` dataset. For each node, select $k = log_2 d = 8$ features randomly and then use the ID3 heuristic to find the best feature for splitting.

   Train the SVM meta-classifier and report the accuracy for both training set and test set. (No cross-validation is required but please choose good parameters for SVM, we will take out points for very low accuracy.)

2. [25 points] Implement same method on the `madelon` dataset. ($k = log_2 d = 11$)

   (a) [20 points] Try $N = 10, 30, 100$. For each N, report accuracy on training set and test set. (cross-validation is not required, but please choose good parameters for the SVM.)

   (b) [5 points] Choose the best N among those you have tried (you may try some new numbers). Report the accuracy, precision, recall and f_1 score on test set.