1. (Linear classifiers, 20 points) Suppose we have the following training instances

\{(x_i, y_i) : i = 1, \ldots, 6\} = \{(-10, -1), (0, 1), (1, -1), (2, -1), (4, 1), (5, 1)\}.

Now we are going to training three linear classifiers \((w, w_0)\) with three different loss functions. The first one is the SVM classifier; the second one is a linear classifier with exponential loss; and the third one is the linear regression model with squared loss. The three loss functions can are specified in Eq. (7.4), (7.6), and (7.7) of the CIML book. The optimization objectives of the three classifiers are listed as follows.

SVM: \[
\min_{(w, w_0)} \sum_{i=1}^{6} \max(1 - y_i(wx_i + w_0), 0) + \frac{\lambda}{2} w^2
\] (1)

exponential-loss classifier: \[
\min_{(w, w_0)} \sum_{i=1}^{6} \exp(-y_i(wx_i + w_0)) + \frac{\lambda}{2} w^2
\] (2)

linear regression: \[
\min_{(w, w_0)} \sum_{i=1}^{6} (wx_i + w_0 - y_i)^2 + \frac{\lambda}{2} w^2
\] (3)

i (6 points) Set \(\lambda = 0.5\), and calculate the (sub)gradient of the three objectives at \((w = -0.5, w_0 = 1)\).

ii (10 points) Set \(\lambda = 0.5\), and optimize the three linear classifiers listed above. You can either use a package to do the calculation or write a gradient descent program yourself. To guarantee the correctness of your solution, you may want to calculate (sub)gradients at your solutions and check whether they are zero. Write out classifiers \((w, w_0)\) you have trained. Mark the decision boundaries of your classifiers at a figure similar to the one above (you can draw a figure yourself).

iii (4 points) Find a classifier \((w^*, w_0^*)\) that is optimal for the binary loss. Mark its decision boundary in the same figure.

2. (Tree classifiers, 10 points) Suppose the training set has \(N\) instances and \(d\) features. We plan to train a tree for our classification task.

i (3 points) What is the computational complexity of training a tree classifier?

ii (3 points) What is the computational complexity of testing the tree with one test instance?

iii (4 points) Suppose we train a tree classifier and a KNN classifier on the same training set. Can you discuss the difference between decision boundaries of the two classifiers? Can we use the tree to speed up KNN prediction?

3. (Confidence Interval, 10 points) Suppose you are given a classifier for a particular task, you test it on a test set of 1000 independent instances, and you observe an accuracy of 0.89.

i (7 points) Calculate a 0.95 confidence interval for the accuracy of the classifier.
ii (3 points) After you have done the calculation, you are told that the classifier has been trained on a training set corrupted by label noises. Is the confidence interval you just calculate still valid? Explain your answer.