CIS 520, Machine Learning, Fall 2021
Homework 5
Due: Monday, October 11, 11:59pm
Submit to Gradescope

Instructions. Please write up your responses to the following problems clearly and concisely. We require you to write up your responses using LaTeX; we have provided a LaTeX template, available on Canvas, to make this easier. **Submit your answers in PDF form to Gradescope. We will not accept paper copies of the homework.**

Collaboration. You are allowed and encouraged to work together. You may discuss the **written homework** to understand the problem and reach a solution in groups. However, **it is recommended that each student also write down the solution independently and without referring to written notes from the joint session.** You must understand the solution well enough to reconstruct it by yourself. (This is for your own benefit: you have to take the exams alone.)

1 Multiclass Adaboost and Support Vector Machines [30 points]

1.1 Adaboost: Theory

In this problem you will analyze the AdaBoost.M1 algorithm, a multiclass extension of AdaBoost. Given a training sample $S = ((x_1, y_1), \dots, (x_m, y_m))$, where x_i are instances in some instance space $\mathcal{X}$ and y_i are multiclass labels that take values in $\{1, \dots, K\}$, the algorithm maintains weights $D_t(i)$ over the examples (x_i, y_i) as in AdaBoost, and on round t, gives the weighted sample (S, D_t) to the weak learner. The weak learner returns a multiclass classifier $h_t : \mathcal{X} \to \{1, \dots, K\}$ with weighted error less than $\frac{1}{2}$; here the weighted error of h_t is measured as

$$\mathrm{er}_t = \sum_{i=1}^{m} D_t(i) \cdot \mathbf{1}(h_t(x_i) \neq y_i)\,.$$

Note that the assumption on the weak classifiers is stronger here than in the binary case, since we require the weak classifiers to do more than simply improve upon random guessing (there are other multiclass boosting algorithms that allow for weaker classifiers; you will analyze the simplest case here). For convenience, we will encode the weak classifier h_t as $\widetilde{h}_t : \mathcal{X} \to \{\pm 1\}^K$, where

$$\widetilde{h}_{t,k}(x) = \begin{cases} +1 & \text{if } h_t(x) = k \\ -1 & \text{otherwise.} \end{cases}$$

In other words, $\widetilde{h}_t(x)$ is a K-dimensional vector that contains $+1$ in the position of the predicted class for x and -1 in all other $(K-1)$ positions. On each round, AdaBoost.M1 re-weights examples such that examples misclassified by the current weak classifier receive higher weight in the next round. At the end, the algorithm combines the weak classifiers h_t via a weighted majority vote to produce a final multiclass classifier H:

```
Algorithm **AdaBoost.M1**
Inputs: Training sample $S = ((x_1, y_1), \ldots, (x_m, y_m)) \in (\mathcal{X} \times \{1, \ldots, K\})^m$
        Number of iterations $T$
Initialize: $D_1(i) = \frac{1}{m} \quad \forall i \in [m]$
For $t = 1, \ldots, T$:
  – Train weak learner on weighted sample $(S, D_t)$; get weak classifier $h_t : \mathcal{X} \to \{1, \ldots, K\}$
  – Set $\alpha_t \leftarrow \frac{1}{2} \ln\left(\frac{1 - \mathrm{er}_t}{\mathrm{er}_t}\right)$
  – Update:
        $D_{t+1}(i) \leftarrow \frac{D_t(i) \exp(-\alpha_t \, \widetilde{h}_{t,y_i}(x_i))}{Z_t}$
    where $Z_t = \sum_{j=1}^m D_t(j) \exp(-\alpha_t \, \widetilde{h}_{t,y_j}(x_j))$
Output final hypothesis:
    $H(x) \in \arg\max_{k \in \{1,\ldots,K\}} \underbrace{\sum_{t=1}^T \alpha_t \widetilde{h}_{t,k}(x)}_{F_{T,k}(x)}$
```

You will show, in five parts below, that if all the weak classifiers have error er_t at most $\frac{1}{2} - \gamma$, then after T rounds, the training error of the final classifier H, given by

$$\mathrm{er}_S[H] = \frac{1}{m} \sum_{i=1}^m \mathbf{1}(H(x_i) \neq y_i)\,,$$

is at most $e^{-2T\gamma^2}$ (which means that for large enough T, the final error $\mathrm{er}_S[H]$ can be made as small as desired).

(a) **[5 points]** Show that

$$D_{T+1}(i) \;=\; \frac{\frac{1}{m} \, e^{-F_{T,y_i}(x_i)}}{\prod_{t=1}^T Z_t}\,.$$

(b) **[5 points]** Show that

$$\mathbf{1}(H(x_i) \neq y_i) \;\leq\; \mathbf{1}\big(F_{T,y_i}(x_i) < 0\big)\,.$$

Hint: Consider separately the two cases $H(x_i) \neq y_i$ and $H(x_i) = y_i$. For the $H(x_i) \neq y_i$ case, you need to show $F_{T,y_i}(x_i) \leq 0$. Note that $\sum_{k=1}^K F_{T,k}(x_i) = -(K-2)\sum_{t=1}^T \alpha_t$, which will be a useful equality.

(c) **[5 points]** Show that

$$\mathrm{er}_S[H] \;\leq\; \frac{1}{m} \sum_{i=1}^m e^{-F_{T,y_i}(x_i)} \;=\; \prod_{t=1}^T Z_t\,.$$

Hint: For the inequality, use the result of part (b) above, and the fact that $\mathbf{1}(u < 0) \leq e^{-u}$; for the equality, use the result of part (a) above.

(d) **[5 points]** Show that for the given choice of α_t, we have

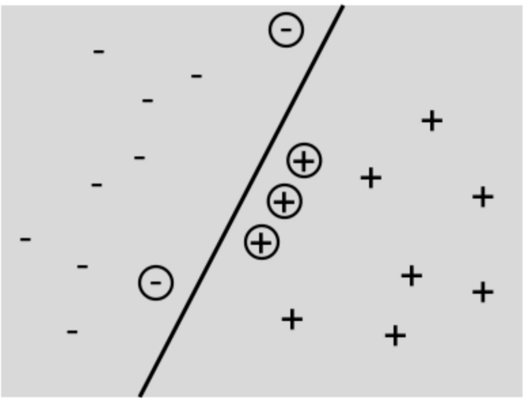
$$Z_t \;=\; 2\sqrt{\mathrm{er}_t(1-\mathrm{er}_t)}\,.$$

(e) **[5 points]** Suppose $\mathrm{er}_t \le \frac{1}{2} - \gamma$ for all t (where $0 < \gamma \le \frac{1}{2}$). Then show that

$$\mathrm{er}_S[H] \;\le\; e^{-2T\gamma^2}\,.$$

1.2 Support Vector Machine

[5 points] Consider a binary classification problem in a 2-dimensional instance space $\mathcal{X} = \mathbb{R}^2$. You are given a linearly separable training set containing 10 positive and 10 negative training examples. You run the hard-margin SVM algorithm and obtain the separating hyperplane below (support vectors are circled):



What is the largest number of data points that can be removed from the training set without changing the hard margin SVM solution? Explain your solution.

2 Programming [20 points]

For this question, refer the Jupyter Notebook. You will be using functions from sklearn and call them in the notebook. We will be using the MNIST digits dataset for a classification task. Familiarize yourself with the dataset on the sklearn website, here.

2.1 Random Forests

[3 points]

Tabulate the prediction results on the test set in Table 1.

n_estimators	**Accuracy (%)**
1	—
5	—
10	—
50	—
100	—
500	—

Table 1: Accuracy for the Random Forests classification problem on the test set

2.2 Kernel SVM

[3 points]

Tabulate the prediction results on the test set in Table 2.

kernel	**Accuracy (%)**
Linear	—
Poly	—
RBF	—

Table 2: Accuracy for the kernel SVM classification problem on the test set

2.3 Multi Layer Perceptron

[3 points]

Tabulate the prediction results on the test set in Table 3.

Network Architecture	**Accuracy (%)**
(3)	—
(10)	—
(10,10,10)	—
(20,50,20)	—

Table 3: Accuracy for the MLP classification problem on the test set

2.4 AdaBoost

[3 points]

Tabulate the prediction results on the test set in Table 4.

n_estimators	**Accuracy (%)**
1	—
5	—
10	—
50	—
100	—
150	—

Table 4: Accuracy for the AdaBoost classification problem on the test set

2.5 Short Answer

[**8 points**] Write a short paragraph describing your results for each of the four models with regards to performance trends and explain the influence of hyperparameters on these results.