CSCI-567 Fall 2019 Midterm Exam 2

Problem	1	2	3	4	5	Total
Points	30	10	15	25	20	100

Please read the following instructions carefully:

- The exam has a total of **11 pages** (including this cover and two blank pages in the end). Each problem have several questions. Once you are permitted to open your exam (and not before), you should check and make sure that you are not missing any pages.
- Duration of the exam is **2 hours**. Questions are not ordered by their difficulty. Budget your time on each question carefully.
- Select **one and only one answer** for all multiple choice questions.
- Answers should be **concise** and written down **legibly**. All questions can be done within 5-12 lines.
- You must answer each question on the page provided. You can use the last two blank pages as scratch paper. Raise your hand to ask a proctor for more if needed.
- This is a **closed-book/notes** exam. Consulting any resources is NOT permitted.
- Any kind of cheating will lead to **score 0** for the entire exam and be reported to SJACS.
- You **may not** leave your seat **for any reason** unless you submit your exam at that point.

1 Multiple Choice, True or False

1. The Adaboost algorithm will eventually give zero training error regardless of the type of weak classifier it uses, provided enough iterations are performed.

   (a) True

   (b) False

2. Boosting algorithm will not select the same weak classifier more than once.

   (a) True

   (b) False

3. In the Adaboost algorithm, weights of the misclassified examples goes up.

   (a) True

   (b) False

4. Which of the following is not a true statement about Lagrangian duality?

   (a) The lagrangian dual function is convex.

   (b) They can be solved with convex optimization

   (c) Duality lets us formulate optimality conditions for constrained optimization problems.

   (d) They can be optimized in the dual space.

5. In a soft margin SVM, what is the behavior of the width of the margin $\left(\frac{1}{\|w\|}\right)$ as $C \rightarrow \infty$

   (a) Behaves like hard margin.

   (b) Goes to zero

   (c) Goes to infinity.

   (d) None of the above.

6. Which of the following statements is not true about SVM?

   (a) For two dimensional data points, the separating hyperplane learned by a linear SVM will be a straight line.

   (b) In theory, a Gaussian kernel SVM can model any complex separating hyperplane.

   (c) The support vectors are expected to remain the same between linear kernels and higher?order polynomial kernels.

   (d) Overfitting in an SVM is a function of number of support vectors.

7. Which of the following statements of hidden Markov model (HMM) is true?

   (a) We can infer the backward message at time t from the backward message at time t + 1 using the backward algorithm.

(b) Given a sequence of observations and a learned HMM, we can infer the real corresponding path of hidden states.

(c) We can learn a HMM using the forward algorithm.

(d) None of the above.

8. Both GMM and HMM can be learned by applying the EM algorithm.

    (a) True

    (b) False

9. Which is not true about the Baum-Welsh algorithm?

    (a) It is used to find unknown parameters of a hidden markov model.

    (b) It uses a forward-backward algorithm to maximize the probability of an observation.

    (c) It computes the most likely sequence of hidden states given an observation sequence.

    (d) It is a special case of the EM algorithm.

2 Mixture Models (20 points)

The general Expectation-Maximization (EM) algorithm is summarized as follow:

```
Algorithm 1: General EM algorithm
    Step 0
      Initialize $\theta^{(1)}$, $t = 1$
    Step 1 (E-Step)
      1-1 Update the posterior of latent variables
          $q_n^{(t)}(\cdot) = p\left(\cdot|\mathbf{x}_n; \boldsymbol{\theta}^{(t)}\right)$
      1-2 Obtain Expectation of complete likelihood
          $Q\left(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}\right) = \sum_{n=1}^{N} \mathbb{E}_{z_n \sim q_n^{(t)}} \left[\ln p\left(\mathbf{x}_n, z_n; \boldsymbol{\theta}\right)\right]$
    Step 2 (M-Step)
      Update the model parameter via Maximization
          $\boldsymbol{\theta}^{(t+1)} \leftarrow \arg\max_{\boldsymbol{\theta}} Q\left(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}\right)$
    Step 3
      $t \leftarrow t + 1$ and return to Step 1 if not converged
```

Consider a GMM with $K \times L$ components. One latent variable, $z_1 \in \{1, 2, ..., K\}$, governs the mean; the other, $z_2 \in \{1, 2, ..., L\}$ governs the covariance. The two latent variables are independent and gives the following PDF

$$p(\mathbf{x}; \boldsymbol{\theta}) = \sum_{k=1}^{K} \sum_{l=1}^{L} \omega_k \nu_l \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_k, \Sigma_l\right). \tag{1}$$

As usual, instead of finding the MLE of the model parameters $\boldsymbol{\theta} = \{\omega_k, \nu_l, \boldsymbol{\mu}_k, \Sigma_l\}_{k=1}^{K}{}_{l=1}^{L}$, we apply EM algorithm to solve it iteratively.

10. Express $p(\mathbf{z}_n = [k,\ l]^T |\ \mathbf{x}_n; \boldsymbol{\theta}^{(t)})$, which is the posterior of latent variables for the E-Step, with model parameters. You may omit $\boldsymbol{\theta}$, the superscript t, and the subscript n for simplicity during derivation; there is no need to expand $\mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_k, \Sigma_l\right)$.

11. Recall that the solution of the simplex optimization problem

$$\begin{aligned} \arg\max_{\mathbf{q}} & \sum_{k=1}^{K} a_k \ln q_k \\ \text{s.t. } & q_k \geq 0 \\ & \sum_{k=1}^{K} q_k = 1 \end{aligned}$$

with $a_1, \cdots a_k \geq 0$ is $q_k^* = \frac{a_k}{\sum_{k'=1}^{K} a_{k'}}$. Let $a_{nkl} = p(\mathbf{z}_n = [k,\ l]^T |\ \mathbf{x}_n; \boldsymbol{\theta}^{(t)})$. Derive the update of parameters ω_k, ν_l for the M-Step.

3 Support Vector Machine (20 points)

Consider a max-margin linear SVM that solves the following optimization problem.

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n \end{aligned}$$

where the optimization parameters are $\mathbf{w}, b$, and the training set consists of points $\{(\mathbf{x}, y)\}_{i=1}^n$. $\mathbf{x}$ is a vector of real values, and $y \in \{-1, 1\}$ is the class label. In this section, you will derive the dual optimization problems.

12. Write down the Lagrangian $\mathcal{L}$ of the above SVM optimization problem. Express your answer as the Lagrangian $\mathcal{L}$ in terms of $\mathbf{w}, b, \alpha_i$, where α_i is the Lagrange multiplier for the inequality constraint at each data instance i. (5 points)

13. Support vector machines learn a decision boundary leading to the largest margin from both classes. You are training SVM on a tiny dataset with 4 points shown in the below Figure. This dataset consists of two examples with class label -1 (denoted with plus), and two examples with class label +1 (denoted with triangles). Find the weight vector $\mathbf{w}$ and bias b. What's the equation corresponding to the decision boundary?

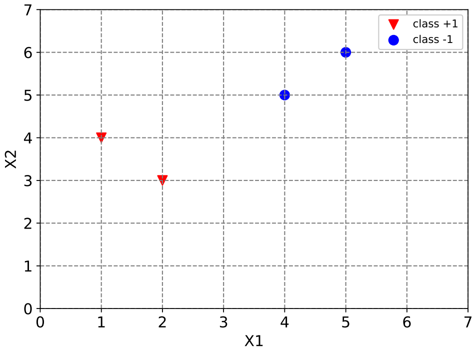

Figure 1

14. Circle the support vectors and draw the decision boundary.

4 Boosting (?? points)

In this question we will look into the AdaBoost algorithm (shown in Alg. 2), where the base algorithm is simply searching for a classifier with the smallest weighted error from a fixed classifier set $\mathcal{H}$.

Algorithm 2: Adaboost

```
Given: A training set $\{(\boldsymbol{x}_n, y_n \in \{+1,-1\})\}_{n=1}^N$, and a set of classifier $\mathcal{H}$, where each $h \in \mathcal{H}$ takes a
  feature vector as input and outputs $+1$ or $-1$.
Goal: Learn $H(\boldsymbol{x}) = \text{sign}\left(\sum_{t=1}^T \beta_t h_t(\boldsymbol{x})\right)$, where $h_t \in \mathcal{H}$, $\beta_t \in \mathbb{R}$, and $\text{sign}(a) = \begin{cases} +1, & \text{if } a \geq 0, \\ -1, & \text{otherwise.} \end{cases}$
Initialization: $D_1(n) = \frac{1}{N}$, $\forall n \in [N]$.
for $t = 1, 2, \cdots, T$ do
    Find $h_t = \arg\min_{h \in \mathcal{H}} \sum_{n: y_n \neq h(\boldsymbol{x}_n)} D_t(n)$.
    Compute
        $$\epsilon_t = \sum_{n: y_n \neq h_t(\boldsymbol{x}_n)} D_t(n) \qquad \text{and} \qquad \beta_t = \frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}.$$
    Compute
        $$D_{t+1}(n) = \frac{D_t(n) e^{-\beta_t y_n h_t(\boldsymbol{x}_n)}}{\sum_{n'=1}^N D_t(n') e^{-\beta_t y_{n'} h_t(\boldsymbol{x}_{n'})}}$$
     for each $n \in [N]$
```

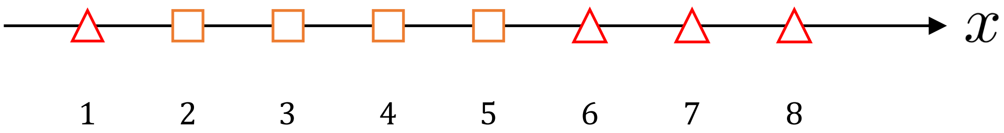

Figure 2: The 1-dimensional training set with 8 data. The square means the class of the data is $+1$, *i.e.* $y = +1$ and the triangle means $y = -1$. The number under each data is its x coordinate.

Now we are given a training set of 8 data as shown in Fig. 2. Each training data is 1-dimension and denoted as a square or a triangle in the figure, where the square means the class of the data is $+1$, *i.e.* $y = +1$ and the triangle means $y = -1$. You are going to experiment on the given training set with the learning process of the AdaBoost algorithm as shown in Alg. 2 for $T = 2$. The base classifier set $\mathcal{H}$ consists of all decision stumps, where each of them is parameterized by a pair $(s, b) \in \{+1, -1\} \times \mathbb{R}$ such that

$$h_{(s,b)}(x) = \begin{cases} s, & \text{if } x > b, \\ -s, & \text{otherwise.} \end{cases}$$

15. Please write down the pair (s, b) of the best decision stump h_1, ϵ_1 and the mis-classified data at $t = 1$. If there are multiple equally optimal stump functions, just randomly pick **ONE** of them to be h_1. **(6 points)**

16. Please write down the pair (s, b) of the best decision stump h_2, ϵ_2 and the mis-classified data at $t = 2$. If there are multiple equally best stump functions, just randomly pick **ONE** of them to be h_2. **(6 points)**

17. Suppose we run AdaBoost for two rounds and observe that β_1 and β_2 are both positive but not equal. Will the training accuracy of the final classifier H after these two rounds (see Line 2) be 1? Explain why or why not. **(3 points)**

5 HMM (15 points)

Recall a hidden Markov model is parameterized by

- initial state distribution $P\left(X_1 = s\right) = \pi_s$
- transition distribution $P\left(X_{t+1} = s' | X_t = s\right) = a_{s,s'}$
- emission distribution $P\left(O_t = o | X_t = s\right) = b_{s,o}$

1. Given a sequence of observations $O_{1:t}$, compute the previous Viterbi path probability from the current time step, which is $\max_{x_{1:t-1}} P\left(X_t = s, X_{1:t-1}, O_{1:t}\right)$ in terms of forward message, backward message, transition probabilities, emission probabilities as needed. **(4 points)**

$$\alpha_s(t) = P\left(X_t = s, O_{1:t} = o_{1:t}\right)$$

$$\beta_s(t) = P\left(O_{t+1:T} = o_{t+1:T} | X_t = s\right)$$

2. Suppose we have a HMM model, each O_t takes a value in $\{A, C, G, T\}$ and each X_t takes one of the two possible states $\{s_1, s_2\}$. This HMM has the following parameters $\boldsymbol{\Theta} = \{\pi_i, a_{ij}, b_{ik}\}$ for $i \in \{1, 2\}$, $j \in \{1, 2\}$, and $k \in \{A, C, G, T\}$:

$$\pi_1 = P(X_1 = s_1) = 0.7; \quad \pi_2 = P(X_1 = s_2) = 0.3$$

$$a_{11} = 0.8, \quad a_{12} = 0.2, \quad a_{21} = 0.4, \quad a_{22} = 0.6$$

$$b_{1A} = 0.4, \quad b_{1C} = 0.1, \quad b_{1G} = 0.4, \quad b_{1T} = 0.1$$

$$b_{2A} = 0.2, \quad b_{2C} = 0.3, \quad b_{2G} = 0.2, \quad b_{2T} = 0.3$$

$$\delta_s(t) = \max_{x_{1:t-1}} P(X_t = s, X_{1:t-1}, O_{1:t}) = b_{s,o_t} \max_{s'} a_{s',s} \delta_{s'}(t-1)$$

$$\Delta_s(t) = \operatorname*{argmax}_{x_{t-1}} \max_{x_{1:t-2}} P(X_t = s, X_{1:t-1} | O_{1:t}) = \operatorname*{argmax}_{\boldsymbol{s'}} a_{s',s} \delta_{s'}(t-1)$$

Now we have observed the output sequence [A G], what is the most likely sequence of states that produce this? **(7 points)**

3. For an arbitrary $T_0 < T$, it is possible to compute the most likely hidden state path $X_1^*, \ldots, X_{T_0}^*$ given the entire observations $O_1, \ldots, O_T$. Your task is to compute ONLY the last state $X_{T_0}^*$. Feel free to use δ_s, forward and backward messages as needed. **(4 points)**

You may use this page as scratch paper, but nothing written on it will be graded.