CIS 520, Machine Learning, Fall 2018: Assignment 8
Due: Monday, December 10th, 11:59pm
[70 points]

Instructions. Please write up your responses to the following problems clearly and concisely. We encourage 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 and SELECT PAGES when uploading to Gradescope. We will not accept paper copies of the homework.**

Note, there is no automatic grader for this homework and you do **not** need to submit your code.

Collaboration. You are allowed and encouraged to work together. You may discuss the homework to understand the problem and reach a solution in groups up to size **two students.** However, each student must write down the solution independently, and without referring to written notes from the joint session. **In addition, each student must write on the problem set the names of the people with whom you collaborated.** You must understand the solution well enough in order to reconstruct it by yourself. (This is for your own benefit: you have to take the exams alone.)

1 Active Learning [20 points]

Part 1. Consider a binary classification problem with labels $\{\pm 1\}$. You are given a small labeled training set, from which you learn a logistic regression model. You are also given three more unlabeled data points, $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$, and are allowed to query the label of one of these. Your logistic regression model predicts the probabilities of each of these instances having label $+1$ as follows:

$$\widehat{\eta}(\mathbf{x}_1) = 0.55\,; \quad \widehat{\eta}(\mathbf{x}_2) = 0.95\,; \quad \widehat{\eta}(\mathbf{x}_3) = 0.3\,.$$

1. (5 points) Suppose your final classifier will be evaluated in terms of 0-1 loss. Using an uncertainty sampling approach, which of the above three instances would you choose to query a label for? Briefly explain your answer.
2. (5 points) Now suppose your final classifier will be evaluated in terms of the following cost-sensitive loss:

$$\begin{array}{cc|cc} & & \multicolumn{2}{c}{\widehat{y}} \\ & & -1 & +1 \\ \hline y & -1 & 0 & 0.8 \\ & +1 & 0.2 & 0 \end{array}$$

   Using an uncertainty sampling approach, which of the above three instances would you choose to query a label for? Briefly explain your answer.

Part 2. Now consider a 3-class classification problem with labels $\{1, 2, 3\}$. You are given a small labeled training set, from which you learn a 3-class logistic regression model. You are also given three more unlabeled data points, $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$, and are allowed to query the label of one of these. Your logistic regression model predicts the probabilities of each of these instances belonging to the 3 classes as follows:

$$\begin{pmatrix} \widehat{\eta}_1(\mathbf{x}_1) \\ \widehat{\eta}_2(\mathbf{x}_1) \\ \widehat{\eta}_3(\mathbf{x}_1) \end{pmatrix} = \begin{pmatrix} 0.8 \\ 0.05 \\ 0.15 \end{pmatrix} ; \quad \begin{pmatrix} \widehat{\eta}_1(\mathbf{x}_2) \\ \widehat{\eta}_2(\mathbf{x}_2) \\ \widehat{\eta}_3(\mathbf{x}_2) \end{pmatrix} = \begin{pmatrix} 0.2 \\ 0.35 \\ 0.45 \end{pmatrix} ; \quad \begin{pmatrix} \widehat{\eta}_1(\mathbf{x}_3) \\ \widehat{\eta}_2(\mathbf{x}_3) \\ \widehat{\eta}_3(\mathbf{x}_3) \end{pmatrix} = \begin{pmatrix} 0.05 \\ 0.45 \\ 0.5 \end{pmatrix} .$$

1. (5 points) Using an uncertainty sampling approach based on *least confident prediction*, which of the above three instances would you choose to query a label for? Briefly explain your answer.

2. (5 points) Using an uncertainty sampling approach based on *margin* between the two highest-probability predictions, which of the above three instances would you choose to query a label for? Briefly explain your answer.

2 Reinforcement Learning [20 points]

Recall that in a reinforcement learning problem, an agent tries to learn an optimal behavior policy by interacting with an environment. Such a problem is usually formulated as a Markov decision process (MDP), given by a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, p, r, \gamma \rangle$; here $\mathcal{S}$ denotes the set of states in the environment, $\mathcal{A}$ denotes the set of actions available to the agent, p is the transition probability function, r is the reward function, and γ is the discount factor. When the MDP is not fully specified, i.e. when the transition probabilities p or the reward function r are unknown, one needs to use reinforcement learning techniques (such as Q-learning). However, if the MDP is fully specified, an optimal policy can be found using dynamic programming.

In this problem, you will consider the `Chain` environment shown in Figure 1. Specifically, in this environment, there are 5 states arranged in a chain, and 2 possible actions available to the agent in each state: `f` ("forward") and `b` ("backward"). Once an agent takes an action, there is some stochasticity in how the environment responds, and correspondingly, which state the agent ends up in/what reward it receives. Each action in a state is depicted by a small black circle; the arrows from actions to states depict possible transitions, labeled by the probability with which that transition occurs given the action and previous state, and the associated reward. As can be seen, when the agent takes the `f` action, it usually (with probability 0.9) moves one step forward, and receives zero reward (unless it is in state 4, in which case it usually stays in state 4 and receives a reward of 10), but it sometimes (with probability 0.1) falls back all the way to state 0 and receives a reward of 2. When the agent takes the `b` action, the reverse happens: it usually (with probability 0.9) falls back to state 0 and receives a reward of 2, but sometimes (with probability 0.1) moves a step forward/stays in state 4 and receives a different reward.

Assume a discount factor of 0.9. The MDP here is then fully specified, and you will use dynamic programming to find an optimal deterministic policy for the agent in this environment.

1. (5 points) Formulate an MDP for the above `Chain` environment by specifying each component of the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, p, r, \gamma \rangle$. For the components p and r, write down $p(s'|s, a)$ and $r(s, a, s')$ for each setting of s, a, s' for which the probability/reward is non-zero.

2. (10 points) Find the optimal state-value function V^*, i.e. find the optimal value $V^*(s)$ for each state s.

   (Hint: You will need to solve the Bellman optimality equation for V^*:

   $$V^*(s) = \max_a \sum_{s'} p(s'|s, a)\Big(r(s, a, s') + \gamma V^*(s')\Big) .$$

   You may write a small piece of code to solve this using value iteration. However, you do not need to turn in the code, and we will not grade your code. You can start with any initial value function V^0,

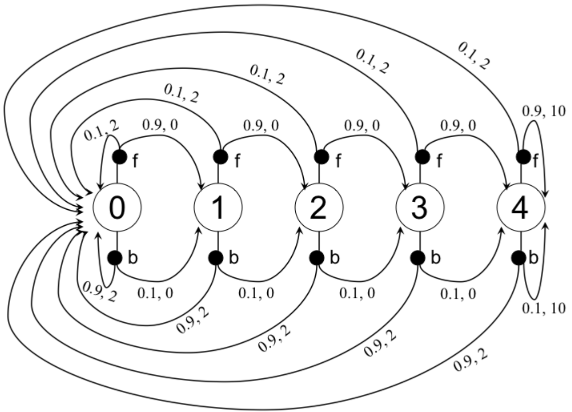

Figure 1: `Chain` environment

iteratively compute V^{t+1} by evaluating the RHS above on V^t, and repeat until convergence (until V^{t+1} becomes indistinguishable from V^t.)

3. (5 points) Using your solution to the second part above, find an optimal deterministic policy π^*, i.e. find an optimal action $\pi^*(s)$ for each state s.

   (Hint: Recall that an optimal policy is given by $\pi^*(s) \in \arg\max_a \underbrace{\sum_{s'} p(s'|s,a)\big(r(s,a,s') + \gamma V^*(s')\big)}_{Q^*(s,a)}$.

   You may write a small piece of code to find such an optimal policy. However, you do not need to turn in the code, and we will not grade your code.

3 Semi-Supervised Learning [30 points]

In this problem you will consider a semi-supervised extension of Naïve Bayes that incorporates unlabeled data via EM. In particular, you will simulate one step of the EM algorithm on a small data set.

For simplicity, consider a binary classification task with 2-dimensional Boolean instances $\mathbf{x} \in \{0,1\}^2$ and labels $y \in \{\pm 1\}$. Recall that the Naïve Bayes classifier assumes a generative probabilistic model of the form

$$p(\mathbf{x}, y) = p(y) \prod_{j=1}^{2} p(x_j \,|\, y)\,.$$

In our setting, there are 5 parameters, which we will collectively denote as $\boldsymbol{\theta}$:

$$\begin{aligned} \theta_{+1} &= \mathbf{P}(Y = +1)\,; \\ \theta_{j|k} &= \mathbf{P}(X_j = 1 \,|\, Y = k)\,, \quad \text{for each feature } j \in \{1,2\} \text{ and each label } k \in \{\pm 1\}\,. \end{aligned}$$

Thus the probability of a labeled example $\mathbf{x} = (1,0), y = -1$ under the above model would be

$$\mathbf{P}(Y = -1)\,\mathbf{P}(X_1 = 1 \,|\, Y = -1)\,\mathbf{P}(X_2 = 0 \,|\, Y = -1) = (1 - \theta_{+1})\theta_{1|-1}(1 - \theta_{2|-1})\,.$$

Standard (Supervised) Naïve Bayes. In the supervised setting, given labeled training data $S_L =$

$((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{m_L}, y_{m_L}))$, one computes maximum likelihood estimates as follows:

$$\begin{aligned}
\widehat{\theta}_{+1} &= \frac{1}{m_L} \sum_{i=1}^{m_L} \mathbf{1}(y_i = +1) \\
\widehat{\theta}_{j|k} &= \frac{\sum_{i=1}^{m_L} \mathbf{1}(y_i = k, x_{ij} = 1)}{\sum_{i=1}^{m_L} \mathbf{1}(y_i = k)}
\end{aligned}$$

Given a new instance $\mathbf{x}$, one then uses the estimated parameters together with Bayes rule to compute the probability $P(Y = +1 \,|\, \mathbf{x})$ under the learned model, and classifies the instance as $+1$ if this probability is greater than $\frac{1}{2}$ and as -1 otherwise.

Semi-Supervised Naïve Bayes. In the semi-supervised setting, given labeled training data S_L as above and additional unlabeled data $S_U = (\mathbf{x}_{m_L+1}, \ldots, \mathbf{x}_{m_L+m_U})$, one treats the missing labels in S_U as unobserved variables and uses EM to find maximum likelihood parameter estimates. In particular, one starts with initial parameter estimates $\widehat{\boldsymbol{\theta}}^0$ learned as above from the labeled data S_L alone, and then iteratively estimates posterior distributions on the missing labels of the unlabeled data points (E-step) and updates the parameter estimates via a weighted maximum likelihood estimation (M-step). Specifically, on each round t, in the E-step, for each unlabeled example $\mathbf{x}_i$ one computes the posterior distribution over labels, $q^t(k \,|\, \mathbf{x}_i) = P(Y_i = k \,|\, \mathbf{x}_i; \widehat{\boldsymbol{\theta}}^t)$, under the current parameter estimates $\widehat{\boldsymbol{\theta}}^t$; in the M-step, one then updates the parameter estimates as follows:

$$\begin{aligned}
\widehat{\theta}_{+1}^{t+1} &= \frac{1}{m_L + m_U} \left(\sum_{i=1}^{m_L} \mathbf{1}(y_i = +1) + \sum_{i=m_L+1}^{m_L+m_U} q^t(+1 \,|\, \mathbf{x}_i) \right) \\
\widehat{\theta}_{j|k}^{t+1} &= \frac{\sum_{i=1}^{m_L} \mathbf{1}(y_i = k, x_{ij} = 1) + \sum_{i=m_L+1}^{m_L+m_U} q^t(k \,|\, \mathbf{x}_i) \cdot \mathbf{1}(x_{ij} = 1)}{\sum_{i=1}^{m_L} \mathbf{1}(y_i = k) + \sum_{i=m_L+1}^{m_L+m_U} q^t(k \,|\, \mathbf{x}_i)}
\end{aligned}$$

On convergence, one uses the final parameter estimates to make predictions on new instances in the same manner as before.

Problem. Suppose you have a training sample of 8 labeled examples and 4 unlabeled examples, distributed as follows (note that since there are only 4 possible instances in our simple setup and 2 possible labels, some examples are repeated in the training sample below; this would be unlikely in real data, but the key ideas present in this example would carry over to real data as well):

Labeled data S_L	**Unlabeled data S_U**
$\mathbf{x} = (x_1, x_2), y$	$\mathbf{x} = (x_1, x_2)$
$(1,1), +1$	$(1,1)$
$(1,1), +1$	$(1,1)$
$(1,0), +1$	$(0,0)$
$(0,0), +1$	$(0,0)$
$(1,0), -1$	
$(0,1), -1$	
$(0,0), -1$	
$(0,1), -1$	
8	4

1. (5 points) Calculate the initial maximum likelihood parameter estimates based on the labeled data only: $\widehat{\boldsymbol{\theta}}^0 = (\widehat{\theta}_{+1}^0, \widehat{\theta}_{1|+1}^0, \widehat{\theta}_{2|+1}^0, \widehat{\theta}_{1|-1}^0, \widehat{\theta}_{2|-1}^0)$. Please use fractions instead of decimals as your results.

2. (8 points) For each instance $\mathbf{x} = (x_1, x_2)$ that appears in the unlabeled data, compute the posterior distribution over the label under the parameter estimates computed in the first part above. In particular,

compute each of the following:

$$\begin{aligned} q^0(+1 \mid \mathbf{x}=(1,1)) &= P(Y=+1 \mid \mathbf{x}=(1,1); \widehat{\boldsymbol{\theta}}^0) \\ q^0(+1 \mid \mathbf{x}=(0,0)) &= P(Y=+1 \mid \mathbf{x}=(0,0); \widehat{\boldsymbol{\theta}}^0) \end{aligned}$$

Show your calculation process. Please use fractions instead of decimals as your results.

(Hint: Use Bayes' rule.)

3. (10 points) Using the results of the first two parts above, find the updated parameter estimates after one step of EM: $\widehat{\boldsymbol{\theta}}^1 = (\widehat{\theta}^1_{+1}, \widehat{\theta}^1_{1|+1}, \widehat{\theta}^1_{2|+1}, \widehat{\theta}^1_{1|-1}, \widehat{\theta}^1_{2|-1})$. Show your calculation process. Please use fractions instead of decimals as your results.

4. (7 points) The log-likelihood of the labeled and unlabeled data $S = (S_L, S_U)$ under parameter estimates $\widehat{\boldsymbol{\theta}}^t$ is given by

$$\ln p(S; \widehat{\boldsymbol{\theta}}^t) \;=\; \sum_{i=1}^{m_L} \ln p(\mathbf{x}_i, y_i; \widehat{\boldsymbol{\theta}}^t) + \sum_{i=m_L+1}^{m_L+m_U} \underbrace{\ln \left(\sum_{y_i \in \{\pm 1\}} p(\mathbf{x}_i, y_i; \widehat{\boldsymbol{\theta}}^t) \right)}_{\ln p(\mathbf{x}_i; \widehat{\boldsymbol{\theta}}^t)} .$$

Write an expression for the log-likelihood of the given training data as a function of the 5 parameters $\widehat{\theta}^t_{+1}, \widehat{\theta}^t_{1|+1}, \widehat{\theta}^t_{2|+1}, \widehat{\theta}^t_{1|-1}, \widehat{\theta}^t_{2|-1}$ for any t.