CS 412: Fall'23
Introduction To Data Mining

Assignment 4

(Due Wednesday, November 29, 11:59 pm)

- The homework is due on October 13 at 11:59 p.m. We will be using Gradescope for homework assignments. If you are still having issues with joining Gradescope, you may use the code shared on Canvas Announcements dated September 5. Please do NOT email a copy of your solution. Contact the TAs if you are having technical difficulties in submitting the assignment. Unfortunately, We will NOT accept late submissions without a reasonable justification.
- Please use Slack or Canvas first if you have questions about the homework. You can also join office hours and/or send us emails. If you are sending us emails with questions on the problems, please start the subject with "CS 412 Fall'23: " and send the email to *all of us* (Arindam, Ruby, Hyunsik, Rohan, Kowshika, Sayar) for faster response.
- Please write down your solutions entirely by yourself and make sure the answers are clear. The homework should be submitted in PDF format; there is no need to submit source code about your computing.
- We will be enforcing the following two aspects:
  - You are expected to typeset the solutions, we encourage you to use Latex. **Handwritten solutions will not be graded and you will not get any credit for handwritten solutions**.
  - When submitting Assignment 4, Gradescope will ask you to select pages for each problem; please do this precisely. **Any pages which are not correctly selected for a problem will not be graded and you will not get any credit for those pages.**
- Please make sure to mark each question clearly. You may use as many pages as needed. Please do not change the order of the questions and answers.
- For each question, you will NOT get full credit if you only give a final result. Please show the necessary steps and details, including any formulae you use, what the variables mean in the formulae, any derivation or calculation steps, and explanations as appropriate. Consider the following two examples:

Example 1 **Q:** Given a dataset $\mathcal{X} = \{3.1, 4.2, -1\}$, compute the mean.
A: For any set of n numbers $\mathcal{X} = \{x_1, \ldots, x_n\}$, the mean can be computed as $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$. For the given dataset $\mathcal{X}$, the mean is $\mu = \frac{3.1+4.2-1}{3} = 2.1$

Example 2 **Q:** A coin claimed to be unbiased has been tossed 100 times, with 54 heads and 46 tails. What is the χ^2 statistic?
A: For a categorical variable taking k possible values, if the expected values are $e_i, i = 1, \ldots, k$ and the observed values are $o_i, i = 1, \ldots, k$, then the χ^2 statistic can be computed

as: $S = \sum_{i=1}^{k} \frac{(o_i - e_i)^2}{o_i}$. For the problem, since the coin is claimed to be unbiased, the expected values are $50, 50$. Further, the observed values are $54, 46$. Then, the chi-squared statistic is given by $S = \frac{(54-50)^2}{50} + \frac{(46-50)^2}{50} = 0.64$.

1. (28 points) Consider the Bayesian network in Figure 1. We denote the random variables Fire as F, Tampering as T, Smoke as S, and Alarm as A. Each of the four variables can take two values: 1 or 0. [1](Please round to **5 decimals** or use **fractions**)

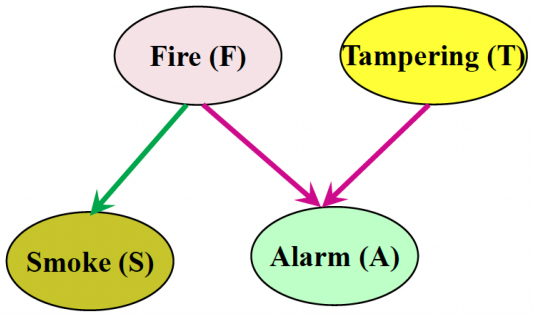

Figure 1: Bayesian network.

The prior probability of Fire and Tampering are: $P(\text{Fire}= 1) = 0.1, P(\text{Tampering}= 1) = 0.01$. The completely specified conditional probability tables (CPTs) for Smoke and Alarm are in Table 1.

Fire	Smoke = 1
1	0.9
0	0.1

Fire	Tampering	Alarm = 1
1	1	0.8
1	0	0.95
0	1	0.8
0	0	0.0001

Table 1: Conditional Probability Tables for Bayesian Network in Figure 1.

(a) (10 points) Using the Bayesian network and the CPTs, compute the joint probability of the following two events:

i. (5 points) $P(F = 1, T = 0, S = 1, A = 1)$.

ii. (5 points) $P(F = 1, T = 1, S = 0, A = 1)$.

(b) (12 points) Recall that by marginalization, the probability of any event can be computed by summing the joint distribution over all possible values of the other variables, e.g.,

$$P(F = 1, A = 1) = \sum_{T\in\{0,1\}} \sum_{S\in\{0,1\}} P(F = 1, A = 1, T, S) \ . \tag{1}$$

Using such marginalization, compute the probabilities of the following events:

i. (6 points) $P(F = 1, A = 1)$.

ii. (6 points) $P(A = 1)$.

(c) (6 points) Using the previous calculations and Bayes rule, compute the probability of the event: $P(F = 1|A = 1)$, i.e., probability of fire given the alarm is ringing.

[1]We use 1, 0 instead of True, False to avoid confusion with T (Tampering) and F (Fire).

2. (20 points) Consider the following dataset for 2-class classification (Figure 2), where the blue points belong to one class and the orange points belong to another class. Each data point has two features $\mathbf{x} = (x_1, x_2)$. We will consider learning support vector machine (SVM) classifiers on the dataset.

Figure 2: 2-class classification dataset.

(a) (6 points) Recall that the soft margin linear SVM learns a linear predictor $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}+b$ using slack variables $\xi_i, i = 1, \ldots, n$, by solving the following optimization:

$$\min_{\mathbf{w},b,\{\xi_i\}} \quad \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i \ ,$$
$$\text{such that} \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i \ , \ \xi_i \geq 0 \ , \ i = 1, \ldots, n.$$

Can we train such a soft margin linear SVM, i.e., with slack variables ξ_i, on the given dataset? If we can train such a classifier, is it possible that all slack variables will be zero, i.e., $\xi_i = 0, i = 1, \ldots, n$ for the dataset? Briefly justify your answers.

(b) (6 points) Professor Poly Kernel claims that mapping each feature vector $\mathbf{x}^i = (x_1^i, x_2^i)$ to a 8-dimensional space given by

$$\phi(\mathbf{x}^i) = [1 \quad x_1^i \quad x_2^i \quad x_1^i x_2^i \quad (x_1^i)^2 \quad (x_2^i)^2 \quad (x_1^i)^4 \quad (x_2^i)^4]^T$$

and training a linear hard-margin SVM in that mapped space would give a highly accurate predictor. Do you agree with Professor Kernel's claim? Clearly explain your answer.

(c) (8 points) Consider running a partitional clustering algorithm on the dataset in Figure 2 ignoring the class labels.

i. (4 points) If we run the kmeans clustering algorithm, will the algorithm be able to identify the inner circle and outer cicrle as different clusters? Please clearly explain your answer.

ii. (4 points) If we run the kernel kmeans clustering algorithm using the RBF (radial basis function) kernel, will the algroithm be able to identify the inner circle and outer cicrle as different clusters? Please clearly explain your answer.

3. (23 points) This question considers Random Forests (RFs).

   (a) (6 points) Briefly describe the three key parameters in RFs Forest-RI: d, the tree depth; m, the number of attributes randomly selected as candidates for splits; and T, the total number of trees.

   (b) (6 points) In the context of classification, clearly describe how RFs Forest-RI (random input selection) are trained and how prediction is done on a test point. Your answer can assume the use of the CART methodology without describing the methodology.

   (c) (6 points) RFs are built by bootstrap sampling, i.e., given an original set of samples of size n, the bootstrapped sample is obtained by sampling with replacement n times. Assuming n is large, what is the expected number of unique samples from the original set of n samples in the bootstrapped sample?

   (d) (5 points) Professor Very Random Forest claims to have a brilliant idea to make RFs Forest-RI more powerful: since RFs prefers trees which are diverse, i.e., not strongly correlated, Professor Forest proposes setting $m = 1$ for Forest-RI, where m is the number of random features used in each node of each decision tree. Professor Forest claims that this will improve accuracy while reducing variance. Do you agree with Professor Forest's claims? Clearly explain your answer.

4. (16 points) We consider optimization especially in the context of deep learning in this question.

   (a) (6 points) Using suitable update equations, clearly differentiate between basic gradient descent and adaptive gradient descent (Adagrad) updates on a loss function $L(\theta)$ where the parameters $\theta \in \mathbb{R}^p$ are high-dimensional, i.e., p is large. Emphasize what aspects of the updates are similar and what aspects are different.

   (b) (4 points) In Adagrad, different parameters get different step sizes. Clearly discuss which parameters get large vs small step sizes. In particular, if the loss $L(\theta)$ does not depend on $\theta_i \in \mathbb{R}$, the i-th component of $\theta \in \mathbb{R}^p$, discuss whether the step sizes for updating θ_i be small or large.

   (c) (6 points) What are the main limitations of Adagrad for deep learning, and how does Adam attempt to fix it. Please describe using the update equations for Adam.

5. (13 points) We consider weighted regression and the attention mechanism. In particular, given a dataset $(\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathbb{R}^d, \mathbf{y}_i \in \mathbb{R}^k, i \in [n]$, and a test point $\mathbf{x} \in \mathbb{R}^d$, consider a regression model of the form

$$\hat{\mathbf{y}}(\mathbf{x}) = \sum_{i=1}^{n} w(\mathbf{x}, \mathbf{x}_i)\mathbf{y}_i \ , \tag{2}$$

where $w(\mathbf{x}, \mathbf{x}_i)$ is a given "similarity" measure which serves as the weight on $\mathbf{y}_i$ in such weighted regression.

(a) (3 points) Show that nearest neighbor regression is a regression model of the form (2).

(b) (2 points) What is the similarity measure $w(\mathbf{x}, \mathbf{x}_i)$ for the k-nearest neighbor regression model?

(c) (3 points) Given a dataset $(\mathbf{x}_i, \mathbf{y}_i), i \in [n]$, can the k-nearest neighbor prediction $\hat{\mathbf{y}}(\mathbf{x})$ be computed in $O(k)$ time? Clearly justify your answer.[2]

(d) (5 points) Show that attention is a regression model of the form (2) with a parameterized similarity measure.

[2]We are not allowed to use additional datastructures or approximate computations.