CS 412: Fall'22
Introduction To Data Mining

Take-Home Final

(Due Saturday, December 10, 06:00 pm)

- The Take-Home Final is due at Saturday, December 10, 6 pm. We will be using Gradescope for the Finals. Please do NOT email a copy of your solution. Contact the TAs if you are having technical difficulties in submitting the assignment. We will NOT accept late submissions!
- Please use Slack first if you have questions about the midterm. You can also come to our (zoom) office hours and/or send us e-mails.
- You will have to answer the questions yourself, you cannot consult with other students in class. It is an open book exam, so you can use the textbook and the material shared in class, e.g., slides, lectures, etc.
- You are expected to typeset the solutions. If your solution has any handwritten components, e.g., equations, tables, figures, etc., please make sure they are legible—otherwise you may not get credit.
- Please attach your answers to the corresponding questions on Gradescope, otherwise there will be a penalty. Please see the canvas announcement on Sept 30 titled "Gradescope Submission Requirements" for details.
- For each question, you will NOT get full credit if you only give out 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**)

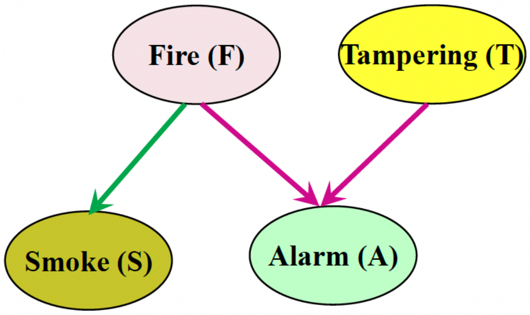

Figure 1: Bayesian network.

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	0.95
0	0.01

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. (12 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.

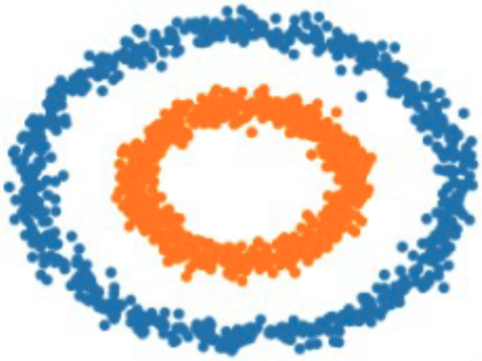

Figure 2: 2-class classification dataset.

(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.

3. (14 points) We consider comparing the performance of two classification algorithms A_1 and B_1 based on k-fold cross-validation. The comparison will be based on a t-test to assess statistical significance with significance level $\alpha = 5\%$. The null hypothesis is that the mean accuracy of the two algorithms A_1 and B_1 are exactly the same.

   (a) (4 points) We will assess the results for $k = 10$-fold cross-validation. What should be the degrees of freedom for the test? Briefly explain your answer.

   (b) (10 points) The accuracies for $k = 10$-fold cross-validation from two algorithms A_1 and B_1 are given in Table 2.

	1	2	3	4	5	6	7	8	9	10
A_1	0.908	0.962	0.878	0.956	0.939	0.955	0.944	0.933	0.881	0.949
B_1	0.449	0.585	0.381	0.433	0.475	0.430	0.520	0.590	0.565	0.443

Table 2: Accuracies on 10-folds for Algorithms A_1 and B_1.

Is the performance of one of the two algorithms significantly different than the other based a t-test at significance level $\alpha = 5\%$? Clearly explain your answer by showing details of (a) the computation of the t-statistic, and (b) the computation of the p-value. Given the t-statistic `t_stat` and degrees of freedom `df`, you should be able to compute the p-value using the following:[2]

```
from scipy.stats import t
p_val = (1-t.cdf(abs(t_stat), df)) * 2
```

[2] Alternatively, you can look a table for p-values for t-statistic, similar to how you had done it for the χ^2-statistic earlier in the semester.

4. (20 points) Let $\mathcal{Z} = \{(\mathbf{x}^1, y^1), \ldots, (\mathbf{x}^n, y^n)\}, \mathbf{x}^i \in \mathbb{R}^d, y^i \in \{0,1\}, i = 1, \ldots, n$ be a set of n training samples. The input $\mathbf{x}^i, i = 1, \ldots, n$ are d-dimensional features, and x_j^i denotes the j-th feature of the i-th data point $\mathbf{x}^i$. The output $y^i \in \{0,1\}, i = 1, \ldots, n$, are the class labels. We consider training a single layer perceptron where for any input $\mathbf{x}^i$, the output is given by

$$\hat{y}^i = g(a^i) = g(\mathbf{w}^T \mathbf{x}^i) = g\left(\sum_{j=1}^{d} w_j x_j^i\right) ,$$

where $g(a) = \max(a, 0)$, i.e., the ReLU activation function, and $a^i = \mathbf{w}^T \mathbf{x}^i$ is the input to the activation function. Note that the parameters $\mathbf{w} = [w_1 \cdots w_d]^T$ are the unknown parameters of the model. Consider a learning algorithm which focuses on minimizing squared loss between the true and predicted outputs:

$$L(\mathbf{w}|\mathcal{Z}) = \frac{1}{2}\sum_{i=1}^{n}(y^i - \hat{y}^i)^2 = \frac{1}{2}\sum_{i=1}^{n}(y^i - g(\mathbf{w}^T \mathbf{x}^i))^2 .$$

(a) (10 points) The stochastic gradient descent (SGD) algorithm updates the parameters based on a random chosen point $(\mathbf{x}^i, y^i)$ in each step. Show that the SGD update for parameter w_j with step size η is of the form

$$w_j^{\text{new}} = w_j + \eta g'(a^i)(y^i - \hat{y}^i)x_j^i , \tag{2}$$

where $a^i = \mathbf{w}^T \mathbf{x}^i$, and the gradient of the ReLU activation function is

$$g'(a^i) = \begin{cases} 1 , & \text{if } a^i \geq 0 , \\ 0 , & \text{otherwise} . \end{cases} \tag{3}$$

(b) (10 points) Instead of using the ReLU activation function $g(a^i) = \max(a^i, 0)$, we now consider the linear activation function $g(a^i) = a^i$. How will you modify (2) and/or (3) above to get the SGD algorithm corresponding to the linear activation function? Clearly explain your answer.

5. (26 points) This question considers clustering algorithms.

   (a) (6 points) What is the computational complexity of the Partitioning Around Medoids (PAM) k-medoids clustering algorithm? Briefly justify your answer.

   (b) (6 points) Is the k-medians algorithm more computationally demanding than k-means? Briefly explain your answer.

   (c) (14 points) Consider a dataset with 5 data points a, b, c, d, e with pairwise distances given by Table 3.

	a	b	c	d	e
a	0				
b	2.2	0			
c	4	6.1	0		
d	7.8	8.6	6.1	0	
e	7.3	7.2	7.3	3.2	0

Table 3: Pairwise distance matrices between data points. Since the distance is assumed to be symmetric, only the lower diagonal and diagonal entries are shown.

Consider running complete link agglomerative clustering algorithm on the dataset. For each step of the algorithm:

   i. (4 points) Show which two clusters will be merged at step based on pairwise distance, and

   ii. (10 points) Show the updated pairwise similarity matrix after merging.