Name: _____

CS57300 Midterm: Spring 2019

This is a closed-book, closed-notes exam. Non-programmable calculators are allowed for probability calculations.

There are 8 pages including the cover page. The total number of points for the exam is 60 and you have 75 minutes to complete the exam. Note the point value of each question and allocate your time accordingly. Read each question carefully and show your work.

Question	Score
1	
2	
3	
4	
5	
6	
7	
8	
Total	

1 Model Components (9 pts)

List the correct components for the methods on the right, using the available choices on the left. Note: you may use a letter more than once.
Grading: +1 pt for correct answers, -0.5pt for incorrect answers, 0pt for blank.

A.	Gradient descent	___N___ Naive Bayes Classifier:	*model space*
B.	Cross validation	___E___ Naive Bayes Classifier:	*score function*
C.	Implicit search	___J___ Naive Bayes Classifier:	*search procedure*
D.	Set of hyperplane boundaries	___K___ Decision Tree:	*model space*
E.	Likelihood of data	___M___ Decision Tree:	*score function*
F.	Greedy search	___F___ Decision Tree:	*search procedure*
G.	Size of margin	___D___ Support Vector Machine:	*model space*
H.	Squared loss	___G___ Support Vector Machine:	*score function*
I.	Information gain	___A___ Support Vector Machine:	*search procedure*
J.	Maximum likelihood estimation		
K.	Set of trees		
L.	Regularization		
M.	Misclassification rate		
N.	Set of probability distributions		
O.	Set of tessellations		

2 True or False (10 pts, 2 pts each)

You only need to indicate True/False for the following questions. You do **not** need to provide any justifications.

(a) If random variables X and Y are conditionally independent given Z, then X and Y are independent with each other.

False. Suppose X is a person's gender, Y is whether a person has cancer, and Z is whether a person smokes. X and Y are conditionally independent given Z, but X and Y are not independent with each other.

(b) If the data violates the Naive Bayes assumption (i.e., the attributes are conditionally independent given the class), applying Naive Bayes classifier on this data will lead to biased posterior probability estimates and thus significantly degrade classification accuracy.

False. Violations of the independence assumption generally bias the probability estimates but it does not degrade classification accuracy much.

(c) Gradient descent can be used to find the minimum value for both convex functions and non-convex functions, as long as the function is differentiable.

True.

(d) The set of logistic regression parameters (i.e., weights) that maximizes likelihood function value on the training dataset always leads to the highest accuracy on the testing dataset.

False. Parameters that give the highest likelihood function value on the training dataset may actually be an overfit of the training dataset, and thus they do not necessarily lead to the highest accuracy on the testing dataset.

(e) When chi-square score is used to select attributes for a decision tree, one would choose the attribute with the highest chi-square score in each iteration.

True.

3 Short Questions (10 pts, 2 pts each)

The following short questions should be answered with **at most** two sentences, and/or a picture.

(a) Name two different types of classification output. For each, list a model that provides that type of output.

Class label: Decision tree, SVM, KNN, Perceptron, etc.
Posterior probability: NBC, Logistic regression, etc.

(b) Describe one advantage of Mahalanobis distance over Euclidean distance and one drawback of Mahalanobis distance compared to Euclidean distance.

Advantage: Mahalanobis adjusts for covariance among features.
Disadvantage: Mahalanobis requires estimation of the covariance matrix (i.e., additional p^2 parameters).

(c) What are the two types of pruning methods for decision trees? For each type of pruning method, give an example.

Post-pruning: reduce error pruning
Pre-pruning: Use cross-validation / statistical test to decide the threshold of selection criterion.

(d) In a dataset with p binary attributes and a binary class label, how many independent parameters need to be learned for a Naive Bayes Classifier, and how many parameters need to be learned for a linear SVM? How do these two numbers change if for each data point, we add all pairwise attribute multiplications—$F_{ij} = X_i X_j$—as additional attributes?

NBC: $2p + 1$, SVM: $p + 1$.

After adding additional attributes: NBC: $2p + 1 + 2\binom{p}{2}$, SVM: $p + 1 + \binom{p}{2}$.

(e) Describe how does the change of K in a K Nearest Neighbor (KNN) classifier influence its performance on training and testing dataset.

K is too small: Overfitting; good training dataset performance, bad testing dataset performance
As K increases, performance on training dataset decreases, performance on testing dataset first increase and then decrease
When K is too large: Underfitting: performance on both training and testing dataset is not very good

4 Principle Component Analysis (6 pts)

Consider the following set of data D:

X_1	X_2
1	1
4	1
1	4

(a) **(4 pts)** Compute the principal component vectors of the above data. (Please have the first dimension of the principal component vectors to be nonnegative.)

First, transform the data matrix such that each dimension has a mean of 0.

$$X = \begin{bmatrix} -1 & -1 \\ 2 & -1 \\ -1 & 2 \end{bmatrix}$$

Compute the covariance matrix:

$$X^T X = \begin{bmatrix} -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} -1 & -1 \\ 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 6 & -3 \\ -3 & 6 \end{bmatrix}$$

Next, we conduct eigendecomposition with respect to the covariance matrix:

$$X^T X - \lambda I = \begin{bmatrix} 6-\lambda & -3 \\ -3 & 6-\lambda \end{bmatrix}$$

Set the determinant of $X^T X - \lambda I$ to be 0: $(6-\lambda)^2 - (-3)^2 = \lambda^2 - 12\lambda + 27 = (\lambda - 9)(\lambda - 3) = 0$.
The two eigenvalues are thus: $\lambda_1 = 9, \lambda_2 = 3$.
Correspondingly, the eigenvectors associated with the first eigenvalue $\lambda_1 = 9$ is:

$$(X^T X - \lambda_1 I)x = \begin{bmatrix} -3 & -3 \\ -3 & -3 \end{bmatrix} x = 0$$

$$x_1 = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix}$$

The eigenvectors associated with the second eigenvalue $\lambda_2 = 3$ is:

$$(X^T X - \lambda_2 I)x = \begin{bmatrix} 3 & -3 \\ -3 & 3 \end{bmatrix} x = 0$$

$$x_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$$

x_1 and x_2 are thus the principal component vectors.

(b) **(2 pts)** Suppose we reduce the dimensionality of the data to 1. What's the transformation of the data point (4, 1) after principal component analysis is applied?

As we will reduce the dimensionality of the data to 1, we will project all data points to the principal component vector that corresponds to the largest eigenvalue. In our case, it's x_1.
Thus, after applying principal component analysis, the data point $d = (4, 1)$ will be transformed as:

$$d^T x = \begin{bmatrix} 4 & 1 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix} = \frac{3\sqrt{2}}{2}$$

5 Naive Bayes (8 pts)

Consider the following set of training data D, in which each data point has two attributes X_1 and X_2; X_1 is a binary attribute and X_2 has three possible values $\{0, 1, 2\}$:

Y	X_1	X_2
+	1	0
+	1	2
+	1	0
+	0	0
-	1	2
-	0	1
-	0	2
-	1	0
-	0	0
-	0	1

(a) **(1 pts)** Give the precise Naive Bayes equation to predict the posterior probability of a data point having a positive label given its attributes.

$P(Y = +|X_1, X_2) = \frac{P(Y=+)P(X_1|Y=+)P(X_2|Y=+)}{P(Y=+)P(X_1|Y=+)P(X_2|Y=+)+P(Y=-)P(X_1|Y=-)P(X_2|Y=-)}$.

(b) **(3 pts)** Specify the maximum likelihood estimates for the class prior and the conditional distribution for the attributes. You do not need to consider Laplace smoothing here.

Class priors: $P(Y = +) = \frac{4}{10} = \frac{2}{5}, P(Y = -) = \frac{6}{10} = \frac{3}{5}$

Conditional distributions for X_1:

$P(X_1 = 0|Y = +) = \frac{1}{4}, P(X_1 = 1|Y = +) = \frac{3}{4}$

$P(X_1 = 0|Y = -) = \frac{4}{6} = \frac{2}{3}, P(X_1 = 1|Y = -) = \frac{2}{6} = \frac{1}{3}$

Conditional distributions for X_2:

$P(X_2 = 0|Y = +) = \frac{3}{4}, P(X_2 = 1|Y = +) = 0, P(X_2 = 2|Y = +) = \frac{1}{4}$

$P(X_2 = 0|Y = -) = \frac{2}{6} = \frac{1}{3}, P(X_2 = 1|Y = -) = \frac{2}{6} = \frac{1}{3}, P(X_2 = 2|Y = -) = \frac{2}{6} = \frac{1}{3}$

(c) **(2 pts)** Given a new test example $\{X_1 = 1, X_2 = 2\}$, according to your computed probabilities in (b), what's your class label prediction to this test example (using a posterior probability threshold of 0.5)?

Posterior probability of a positive label is:
$P(Y = +|X_1 = 1, X_2 = 2) = \frac{P(Y=+)P(X_1=1|Y=+)P(X_2=2|Y=+)}{P(Y=+)P(X_1=1|Y=+)P(X_2=2|Y=+)+P(Y=-)P(X_1=1|Y=-)P(X_2=2|Y=-)}$
$= \frac{\frac{2}{5}\times\frac{3}{4}\times\frac{1}{4}}{\frac{2}{5}\times\frac{3}{4}\times\frac{1}{4}+\frac{3}{5}\times\frac{1}{3}\times\frac{1}{3}} = \frac{9}{17} > 0.5$

Hence, the predicted class label will be +.

(d) **(2 pts)** What are the conditional distributions for X_2 if Laplacian smoothing is used? Why Laplacian smoothing is useful?

If Laplacian smoothing is used, the conditional distributions for X_2 would be:

$P(X_2 = 0|Y = +) = \frac{3+1}{4+3} = \frac{4}{7}, P(X_2 = 1|Y = +) = \frac{0+1}{4+3} = \frac{1}{7}, P(X_2 = 2|Y = +) = \frac{1+1}{4+3} = \frac{2}{7}$

$P(X_2 = 0|Y = -) = \frac{2+1}{6+3} = \frac{1}{3}, P(X_2 = 1|Y = +) = \frac{2+1}{6+3} = \frac{1}{3}, P(X_2 = 2|Y = +) = \frac{2+1}{6+3} = \frac{1}{3}$

Laplacian smoothing is useful when in the training dataset, there are "zero-count" problems, that is, some combinations of class label and attribute values never show in the training dataset. If Laplacian smoothing is not used, the maximum likelihood estimation will estimate the corresponding conditional probability to be 0, and thus any testing data point having that attribute values will get a posterior probability of 0 belong to the corresponding class. This is likely an overfitting of the training dataset. Laplacian smoothing is thus a method that can alleviate overfitting.

6 Logistic Regression (8 pts)

Consider the following set of training data D for predicting whether the rating of a restaurant is above 4 (on a 5 star scale):

Above 4?	Food Type	Price	No. of Reviews
Yes	American	$	54
No	Asian	$$	25
No	American	$$$	193
Yes	French	$$$	127

(a) **(2 pts)** How would you deal with the categorical attributes Food Type and Price? Please provide the new table of training data after you process the categorical attributes.

For nominal variable "Food Type," we use one-hot encoding to transform it: American: [0, 0], Asian: [1, 0], French: [0, 1]; for ordinal variable "Price," we map it to an increasing sequence of numbers: $: 1, $$: 2, $$$: 3.

The new data table is thus:

Above 4?	Asian Food	French Food	Price	No. of Reviews
Yes	0	0	1	54
No	1	0	2	25
No	0	0	3	193
Yes	0	1	3	127

(b) **(2 pts)** Suppose there are N examples in your training dataset. What's the scoring function for logistic regression? Please provide the formula of logistic regression's scoring function.

In logistic regression, $P(y=1|\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w^T x}}}$

The scoring function of logistic regression is log likelihood function, that is:

$log(D|\mathbf{w}) = \sum_{i=1}^{N} \log p(y_i|\mathbf{w}, \mathbf{x_i}) = \sum_{i=1}^{N} \log[(\frac{1}{1+e^{-\mathbf{w^T x_i}}})^{y_i}(\frac{e^{-\mathbf{w^T x_i}}}{1+e^{-\mathbf{w^T x_i}}})^{1-y_i}]$

$= \sum_{i=1}^{N}[-y_i \log(1+e^{-\mathbf{w^T x_i}}) + (1-y_i)(-\mathbf{w^T x_i} - \log(1+e^{-\mathbf{w^T x_i}}))]$

$= \sum_{i=1}^{N}[(y_i-1)\mathbf{w^T x_i} - \log(1+e^{-\mathbf{w^T x_i}})]$

$= \sum_{i=1}^{N}[y_i \mathbf{w^T x_i} - \log(1+e^{\mathbf{w^T x_i}})]$

(c) **(3 pts)** Use D as the training dataset. Conduct one step of gradient descent/ascent to show how you would update the value for the weight parameter *that is associated with the attribute "No. of Reviews."* Consider the initial parameter with zeros on all attributes, and step size $\eta = 0.01$. Briefly describe when you will stop your gradient descent/ascent algorithm. (Please map label "Yes" to 1.)

Take a derivative of the scoring function with respect to w_j:

$\frac{dlog(D|\mathbf{w})}{dw_j} = \sum_{i=1}^{N}[y_i x_{ij} - \frac{x_{ij} e^{\mathbf{w^T x_i}}}{1+e^{\mathbf{w^T x_i}}}]$

At $\mathbf{w_0} = (0,0,0,0,0)$, each dimension of the gradient ∇w becomes: $\sum_{i=1}^{4}[y_i x_{ij} - \frac{x_{ij}}{2}]$.

For the weight associated with "No. of Reviews": $(1-\frac{1}{2}) \times 54 + (0-\frac{1}{2}) \times 25 + (0-\frac{1}{2}) \times 193 + (1-\frac{1}{2}) \times 127 = -18.5$

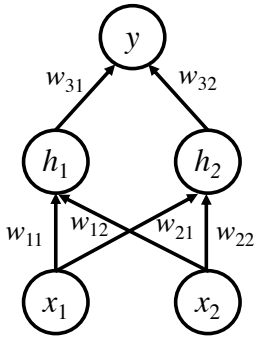
Given a step size of $\eta = 0.01$, we apply one step of gradient ascent, and the new weight for "No. of Reviews" will be -0.185.
The gradient ascent algorithm will stop when the length of gradient is sufficiently small, the change of scoring function value is sufficiently small, or the subsequent two parameters are sufficiently close.

(d) **(1 pt)** How to avoid overfitting for logistic regression?

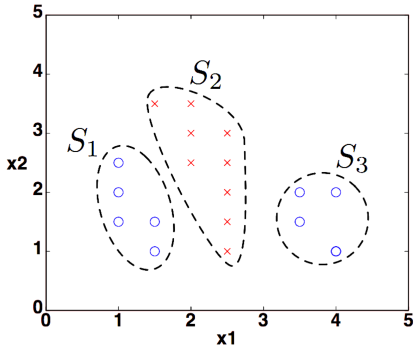
Add regularization term in the scoring function such as $\|\mathbf{w}\|^2$.

7 Neural Networks (4 pts)

Consider the following neural network structure:



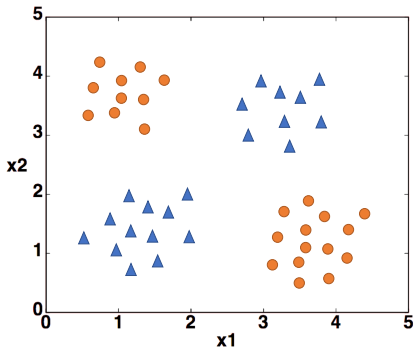
(a) **(2 pts)** Does there exist a set of parameter values for the above neural network that can correctly classify the dataset given in the following figure?



Yes. Set w_{11}, w_{12} in a way such that $h_1(\mathbf{x}) = -1$ when $\mathbf{x} \in \{S_1, S_2\}$ and $h_1(\mathbf{x}) = +1$ when $\mathbf{x} \in \{S_3\}$; and set w_{21}, w_{22} in a way such that $h_2(\mathbf{x}) = -1$ when $\mathbf{x} \in \{S_2, S_3\}$ and $h_2(\mathbf{x}) = +1$ when $\mathbf{x} \in \{S_1\}$. Then $y = sign(h_1 + h_2 + 0.5)$, i.e., $w_{31} = w_{32} = 1$.

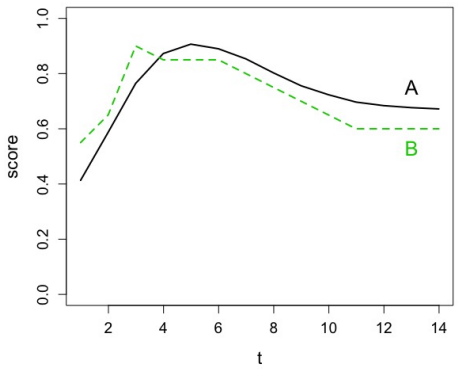
(b) **(2 pts)** Does there exist a set of parameter values for the above neural network that can correctly classify the dataset given in the following figure?

No. We need two layers of hidden nodes to capture XOR functions. (However, if you answered yes, we also give you full points—given the current graphical representation of the data points, it is possible to find two roughly 45 degree lines to separate the data...)



8 Scoring functions (5 pts)

Consider the plot below, which shows the scoring function over a model space for two datasets (A and B). The model is a simple threshold rule (`If` $x > t$ `then` $y = +$ `else` -).



(a) **(1 pt)** What could be on the Y axis represent? Justify your answer.

Accuracy, AUC, or other reasonable function that is maximized with scores from 0 to 1.

(b) **(2 pt)** What model will be returned if the learning algorithm uses dataset A?
What will be returned in the algorithm uses dataset B?

t≈5 will be returned if A is used; t≈3 will be returned if B is used.

(c) **(2 pts)** Suppose A is the result on the full population of data, and B is the result on a sample of the data. Do you observe any problems in terms of model learning in the above figure? If yes, please describe an approach that could be used to avoid this problem for the given type of model.

When dataset B is used for estimating model parameters, the model returned is an overfit of the training dataset. We can use cross-validation to address this overfit problem.