(PRINT) Name ____Student No__________

Signature ____Total Mark_____/100

University of Waterloo

Computer Science 489/689 – Machine Learning

Midterm Test
2018 February 14
Time: 8:35 am – 9:50 am

Time: 75 minutes
Total marks: 100

Answer all questions on this paper. This is an open book exam in which notes, lecture slides, textbooks and non-programmable calculators are permitted. However, computers, tablets, smart phones and smart watches are not permitted.

This examination has 8 pages. Check that you have a complete paper.

1	**/ 24**
2	**/ 20**
3	**/ 24**
4	**/ 16**
5	**/ 16**
Total	**/ 100**

Question 1 [24 pts] A grocery store sells papayas from 3 different countries (country A, country B and country C). All the papayas from country A are sweet, while 75% of the papayas from country B are sweet and 50% of the papayas from country C are sweet. Unfortunately, the sign in the grocery store that would normally indicate where the papayas are from is missing. Nevertheless, you buy a bag of papayas. Before tasting any papaya, you believe that they are from country A with probability 0.5, country B with probability 0.25 and country C with probability 0.25. After tasting 5 papayas, you noticed that 4 of them are sweet and 1 of them are not sweet.

a) i) **[4 pts]** Bayesian learning: what is your posterior belief (after tasting the 5 papayas mentioned above) that the papayas are from each country?

Prior: $\Pr(A) = 0.5$ $\quad\quad$ $\Pr(B) = 0.25$ $\quad\quad$ $\Pr(C) = 0.25$

Likelihood: $\Pr(sweet|A) = 1$

$\Pr(sweet|B) = 0.75$

$\Pr(sweet|C) = 0.5$

Unnormalized posterior:

$\Pr(A|4\ sweets, 1{\sim}sweet) = 0$ (1 pt)

$\Pr(B|4\ sweets, 1{\sim}sweet) \propto 0.25\ (0.75)^4\ (0.25)^1 = 0.01978$ (1 pt)

$\Pr(C|4\ sweets, 1{\sim}sweet) \propto 0.25\ (0.5)^5\ = 0.00781$ (1 pt)

Normalized posterior: (1 pt)

$\Pr(A|3\ sweets, 2{\sim}sweets) = 0$

$\Pr(B|3\ sweets, 2{\sim}sweets) = 0.7169$

$\Pr(C|3\ sweets, 2{\sim}sweets) = 0.2831$

ii) **[4 pts]** What is the probability that the next papaya that you will taste is sweet according to the Bayesian learning prediction?

$$\begin{aligned}\Pr(sweet|\ 4\ sweets, 1{\sim}sweet) &= \Pr(sweet|A)\Pr\ (A|4\ sweets, 1{\sim}sweet)\\ &+ \Pr(sweet|B)\Pr(B|4\ sweets, 1{\sim}sweet)\\ &+ \Pr(sweet|C)\Pr(C|4\ sweets, 1{\sim}sweet)\ \text{(formula 2 pts)}\\ &= (1)(0) + (0.75)(0.7169) + (0.5)(0.2831) = 0.6792\ \text{(2 pts)}\end{aligned}$$

b) i) **[4 pts]** Maximum a posteriori learning: what is the maximum a posteriori hypothesis (after tasting the 5 papayas mentioned above).

$h_{MAP} = argmax_{h_i} \Pr(h_i|e) = h_B$
thus the answer is hypothesis B

ii) **[4 pts]** What is the probability that the next papaya that you will taste is sweet according to the maximum a posteriori hypothesis?

$\Pr(sweet|B) = 0.75$

c) i) **[4 pts]** Maximum likelihood learning: what is the maximum likelihood hypothesis (for the taste of the 5 papayas mentioned above)?

$\Pr(4\ sweets, 1 \sim sweet)\ |A) = 0$
$\Pr(4\ sweets, 1 \sim sweet)\ |B) = (0.75)^4(0.25)^1 = 0.0791$
$\Pr(4\ sweets, 1 \sim sweet)\ |C) = (0.5)^5 = 0.03125$
Thus the maximum likelihood hypothesis is B

ii) **[4 pts]** What is the probability that the next papaya that you will taste is sweet according to the maximum likelihood hypothesis?

$\Pr(sweet|B) = 0.75$

Question 2 [20 pts] Consider the following dataset. The input space has one dimension and there are two classes (+ and -):

(0,+), (1,+), (0.9,+), (0.5,+), (1.5, -), (0.7, -), (1.2, -), (0.95,-)

a) **[8 pts]** Suppose that we are training a mixture of Gaussians model by maximum likelihood. What are the prior probabilities of each class Pr(+) and Pr(-)? What is the mean of the conditional distribution of each class and what is the variance (assuming that both class conditional distributions have the same variance)?

$\Pr(+) = \frac{4}{8} = 0.5$ (1 pt)
$\Pr(-) = \frac{4}{8} = 0.5$ (1 pt)

$\mu_+ = \frac{0+1+0.9+0.5}{4} = 0.6$ (1 pt)
$\mu_- = \frac{1.5+0.7+1.2+0.95}{4} = 1.0875$ (1 pt)

$S_+ = 0.25((0-0.6)^2 + (1-0.6)^2 + (0.9-0.6)^2 + (0.5-0.6)^2)$
$= 0.155$ (1pt)
$S_- = 0.25(0.4125^2 + 0.3875^2 + 0.1125^2 + 0.1375^2) = 0.08797$ (1 pt)

$\Sigma = \frac{4}{8}(0.155) + \frac{4}{8}(0.08797) = 0.1215$ (2 pts)

b) **[4 pts]** What is the probability that 0.92 belongs to class +?

$\Pr(x = 0.92|+) \propto \exp\left(-0.5(0.92-0.6)(0.1215)^{-1}(0.92-0.6)\right)$
$= 0.6561\ (1\ pt)$
$\Pr(x = 0.92|-) \propto \exp\left(-0.5(0.92-1.0875)(0.1215)^{-1}(0.92-1.0875)\right)$
$= 0.8910\ (1\ pt)$

$\Pr(+|x = 0.92) = \frac{\Pr(+)\Pr(x = 0.92|+)}{\Pr(+)\Pr(x = 0.92|+) + \Pr(-)\Pr(x=0.92|-)} = 0.424$
(1 pt for the formula, 1 pt for the solution)

c) **[4 pts]** Where is the decision boundary between the + and - classes?

$$boundary = \frac{N_+}{N}\mu_+ + \frac{N_-}{N}\mu_- = \frac{0.6+1.0875}{2} = 0.84375$$

Since there is only one dimension, we can take the weighted average of the means to find the boundary.

d) **[4 pts]** Suppose that the examples are not distributed according to a mixture of Gaussians, but according to a mixture of exponentials instead. What algorithm could you use to obtain better results for this kind of data?

Choices:	Logistic Regression
	Mixture of Exponentials
	Nonlinear equations (KNN)

Question 3 [24 pts] Indicate whether each statement is true or false. No justification required.

a) **[4 pts]** The back-propagation algorithm is guaranteed to find the best parameters for a given neural network architecture.

False, could get stuck in local optima

b) **[4 pts]** In linear regression, minimizing the squared loss and maximizing the likelihood of the data under the assumption of Gaussian noise gives the same result.

True

c) **[4 pts]** A two-layer neural network with a single output neuron and sigmoid activation functions at each neuron is the same as logistic regression with adaptable sigmoid basis functions.

True

d) **[4 pts]** In *k*-nearest neighbours, a small *k* may lead to underfitting while a large *k* may lead to overfitting.

False, it is the reverse

e) **[4 pts]** Linear models with suitable basis functions can be used for non-linear classification and regression.

True

f) **[4 pts]** Logistic regression is a classification technique, not a regression technique.

True

4) [16 pts] A local startup is using machine learning to classifier customers into several categories. The startup trained a classifier on a set of 1,000 labeled customers. Since this classifier achieved 100% accuracy with the customers in this training set, the startup deployed the classifier into a product. To its surprise, the classifier achieved an accuracy of only 80% after the deployment. Since you are a machine learning expert, the startup hires you as a consultant to help resolve this situation.

a) **[8 pts]** What mistake did the startup make? Explain how this mistake could explain why the accuracy of the classifier went from 100% before the deployment to 80% after the deployment.

Mistake: did no testing or used the training set as the test set

Problem: the classifier was overfitting the training set, thus the accuracy went down after deployment when trying to classify new examples.

b) **[8 pts]** Describe an approach to evaluate the classifier that should ensure that the accuracy observed after the deployment is close to the accuracy measured before the deployment.

Methods: cross-validation or regularization

5) [16 pts] Consider the problem of spam filtering. A classifier can be trained to classify email messages as legitimate or spam by minimizing the misclassification rate. However, in spam filtering there are different costs for different types of misclassification. Let c_1 be the cost of misclassifying a legitimate email as spam and let c_2 be the cost of misclassifying a spam message as legitimate. Note that c_1 is much higher than c_2 since missing a legitimate email because it ended up in a junk folder may have severe consequences while allowing a spam message to appear in the inbox only costs a bit of additional time to manually delete it.

a) **[8 pts]** Suppose that $c_1 = 10 \times c_2$. How would you modify the training set so that any training algorithm that minimizes the misclassification rate will automatically find a classifier that minimizes the expected cost of misclassification when run on the modified training set?

Create 10 copies of each legitimate email in the training set. This way, a legitimate email that is misclassified as spam will incur 10 times the cost.

b) **[8 pts]** Consider logistic regression, where the weights are selected to maximize the likelihood of the correct class (which is equivalent to minimizing the probability of misclassification).

$$\boldsymbol{w}^* = argmax_{\boldsymbol{w}} \prod_n \sigma(\boldsymbol{w}^T\overline{\boldsymbol{x}}_n)^{y_n}\left(1-\sigma(\boldsymbol{w}^T\overline{\boldsymbol{x}}_n)\right)^{1-y_n}$$

where $(\boldsymbol{x}_n, y_n)$ is the n^{th} data point in the training set

How would you modify this objective to minimize the average cost of misclassification by taking into account the costs c_1 and c_2 of the different types of misclassification in spam filtering?

The new objective function:

$$w^* = argmin_w \sum_n \left[c_1\left(1-\sigma(w^T\bar{x}_n)\right)\right]^{y_n}[c_2\sigma(w^T\bar{x}_n)]^{1-y_n}$$

Key points:
1) Unlike probabilities that are typically multiplied, we need to sum up the costs incurred.
2) We need to weigh each misclassification probability by its cost. This is done by multiplying each misclassification probability by the corresponding cost.
3) Since we are interested in the misclassification probabilities (instead of the correct classification probabilities), we swap σ and $1-\sigma$.

NB: 6 points are earned for the following objective which arises when we duplicate each data point a number of times that is proportional to its misclassification cost. This is a reasonable objective, but it does not minimize expected cost, which is why 6 points are earned instead of 8.

$$w^* = argmax_w \prod_{\mathrm{n}} \left(\sigma(w^T \bar{x}_n)\right)^{c_1 y_n} \left(1 - \sigma(w^T \bar{x}_n)\right)^{c_2(1-y_n)}$$