Last (Family) Name:

First (Given) Name:

Student Number:

Section *(circle one)*: L0101 = Mon, L0201 = Wed, L0301 = Th

PRACTICE FINAL EXAM

CSC311 Fall 2019
Introduction to Machine Learning

University of Toronto
Faculty of Arts & Science

Duration - 3 hours
Aids allowed: Two double-sided handwritten $8.5'' \times 11''$ or A4 aid sheets.

Exam reminders:

- Fill out your name and student number on the top of this page.
- Do not begin writing the actual exam until the announcements have ended and the Exam Facilitator has started the exam.
- Write all answers in the provided answer booklets.
- Blank scrap paper is provided at the back of the exam.
- If you possess an unauthorized aid during an exam, you may be charged with an academic offence.
- Turn off and place all cell phones, smart watches, electronic devices, and unauthorized study materials in your bag under your desk. If it is left in your pocket, it may be an academic offence.
- When you are done your exam, raise your hand for someone to come and collect your exam. Do not collect your bag and jacket before your exam is handed in.
- If you are feeling ill and unable to finish your exam, please bring it to the attention of an Exam Facilitator so it can be recorded before leaving the exam hall.
- In the event of a fire alarm, do not check your cell phone when escorted outside.

Hand in all examination materials at the end
DO NOT WRITE ANY ANSWERS ON THIS PAPER

1. True/False. For each statement below, say whether it is true or false, and give a **one or two sentence** justification of your answer.

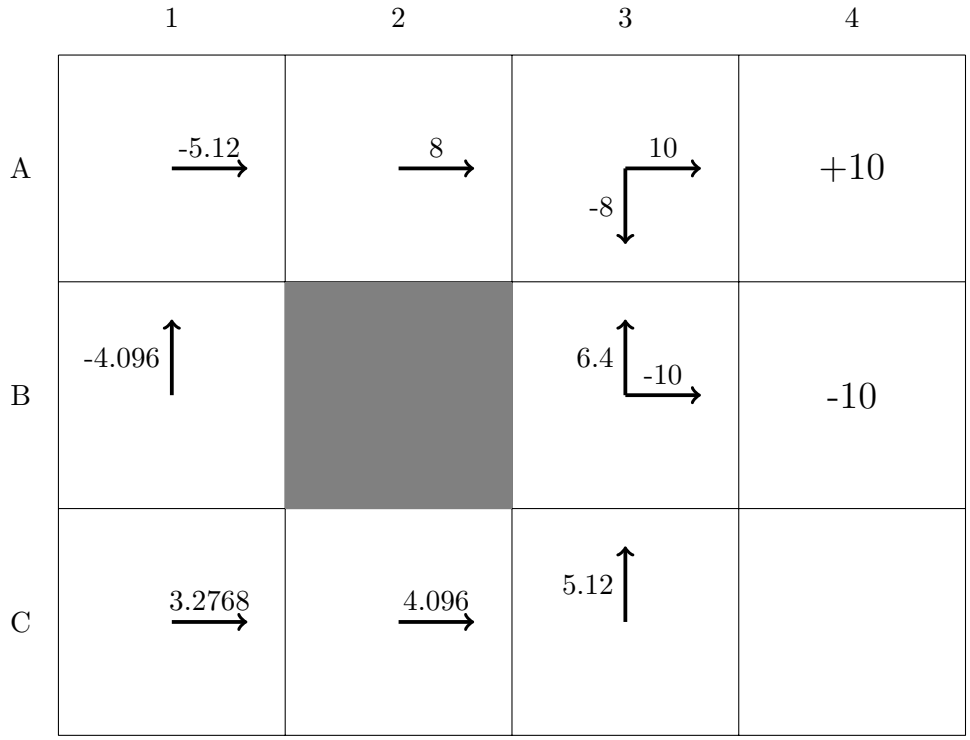
a) Adding more training data always reduces overfitting.
False. Although in general adding more training data should reduce overfitting, in some cases it may not help. For example, if our model already has high bias, or if the new data we add is very similar to our existing data.

b) For small k, the k-means algorithm is equivalent to the k-nearest neighbors algorithm.
False. k-means is an algorithm used for clustering, while k-nearest neighbors is an algorithm used for classification.

c) An ensemble of models always has more capacity than a single model.
False. An ensemble of a few very weak learners can have less capacity than a much larger single model.

d) A linear SVM will find the same decision boundary as logistic regression.
False. A linear SVM uses hinge loss while logistic regression uses logistic loss. Since logistic regression still penalizes and tries to minimize loss on correctly classified examples, it will learn a different decision boundary.

2. Reinforcement Learning.



Consider the familiar robot navigation task within the gridworld shown above. You can move in any of the four directions (left/right/up/down) unless blocked by one of the gray obstacles at B2 and B3. The rewards are +10 for state C4, and -10 for state B4. A4 and B4 are both absorbing states. The reward for every other state is 0.

a) Assume that the state transitions are deterministic. Recall that under the simple Q-learning algorithm, the estimate Q values are update using the following rule:

$$\hat{Q}(s,a) = r(s') + \gamma \max_{a'} \hat{Q}(s',a')$$

Consider applying this algorithm when all the $\hat{Q}$ values are initialized to zero and $\gamma = 0.8$. Write the Q estimates on the figure as labeled arrows after the robot has executed the following state sequences:

- B1 → A1 → A2 → A3 → B3 → B4
- A2 → A3 → A4
- C1 → C2 → C3 → B3 → A3 → A4

b) Assume the robot will now use the policy of always performing the action having the greatest Q value. Is this the optimal policy? Why or why not?

No, although the policy is optimal for the states that the agent has explored. It is able to reach the highest reward absorbing state A4 in the least steps possible from any start state. However, since it never explored state C4, it has no explicit policy for this state and may take a suboptimal action upwards to state B4.

c) Suppose state A3 also has a reward of -10, and states A3 and B3 are no longer absorbing. How can we ensure that our agent is still able to find the optimal policy in this new environment?

To ensure we are still able to find the optimal policy, we need to make sure we balance exploitation and exploration. If we do not allow the agent to explore enough, it will not be able to overcome the group of negative reward states surrounding the positive reward state in order to collect the optimal reward. We can do this by adopting an ϵ-greedy approach.

3. Backpropagation. Consider a L_2 regularized single layer neural network model that predicts continuous 1d targets $y = \sigma(z) \in \mathbb{R}$ where $z = wh + b$ and $h = \sigma(v)$ where $v = w'x + b'$, and σ is an activation function. To train, we use mean squared error from the targets $t \in \mathbb{R}$ with L_2 penalty on w, b' given by $\mathcal{L} = (y - t)^2/2 + w^2 + b'^2$

a) Write the loss as a function of the parameters w, b and compute directly $\frac{\partial \mathcal{L}}{\partial w}$, $\frac{\partial \mathcal{L}}{\partial b'}$.

$$\begin{aligned}
\mathcal{L} &= (\sigma(w\sigma(w'x + b') + b) - t)^2/2 + w^2 + b'^2 \\
\frac{\partial \mathcal{L}}{\partial w} &= (\sigma(w\sigma(w'x + b') + b) - t) \cdot \sigma'(w\sigma(w'x + b') + b) \cdot \sigma(w'x + b') + 2w \\
\frac{\partial \mathcal{L}}{\partial b'} &= (\sigma(w\sigma(w'x + b') + b) - t) \cdot \sigma'(w\sigma(w'x + b') + b) \cdot w\sigma'(w'x + b') \cdot 1 + 2b'
\end{aligned}$$

b) Now compute $\frac{\partial \mathcal{L}}{\partial w}$, $\frac{\partial \mathcal{L}}{\partial b'}$ using the backprogation algorithm.

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w} &= \frac{\partial \mathcal{L}}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial \mathcal{L}}{\partial w} = \frac{\partial \mathcal{L}}{\partial y}\frac{\partial y}{\partial z}\frac{\partial z}{\partial w} + \frac{\partial \mathcal{L}}{\partial w} = (y - t) \cdot \sigma'(z) \cdot h + 2w \\
\frac{\partial \mathcal{L}}{\partial b'} &= \frac{\partial \mathcal{L}}{\partial y}\frac{\partial y}{\partial z}\frac{\partial z}{\partial h}\frac{\partial h}{\partial v}\frac{\partial v}{\partial b'} + \frac{\partial \mathcal{L}}{\partial b'} = (y - t) \cdot \sigma'(z) \cdot w \cdot \sigma'(v) \cdot 1 + 2b'
\end{aligned}$$

c) What are the disadvantages of doing a) versus backpropagation? Why do we use backpropagation in machine learning as opposed to direct differentiation?
SOL : as the computational graph of the network increases it gets much more difficult to write down the full expression in a) In ML we use back-prop as we want to write down a program that efficiently computes the derivatives so that we can do gradient descent, we don't care about closed form expressions.

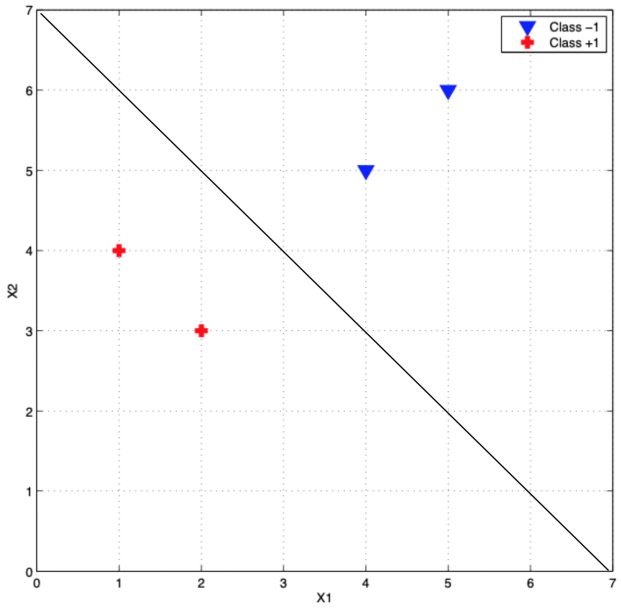
4. Principal Component Analysis. Recall that the optimal PCA subspace can be determined from the eigendecomposition of the empirical covariance matrix $\hat{\mathbf{\Sigma}}$. Also recall that the eigendecomposition can be expressed in terms of the following spectral decomposition of $\hat{\mathbf{\Sigma}}$:

$$\hat{\mathbf{\Sigma}} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top,$$

where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{\Lambda}$ is a diagonal matrix. Assume the eigenvalues are sorted from largest to smallest. You may assume all of the eigenvalues are distinct.

1. If you've already computed the eigendecomposition (i.e. $\mathbf{Q}$ and $\mathbf{\Lambda}$), how do you obtain the orthogonal basis $\mathbf{U}$ for the optimal PCA subspace? (You do not need to justify your answer.)
   SOL: pick the first k columns of Q.
2. The PCA code vector for a data point $\mathbf{x}$ is given by $\mathbf{z} = \mathbf{U}^\top(\mathbf{x} - \hat{\boldsymbol{\mu}})$ where $\hat{\boldsymbol{\mu}}$ is the data mean. Show that the entries of $\mathbf{z}$ are uncorrelated.
   SOL $\text{Cov}(\mathbf{z}) = \mathbf{U}^T\text{Cov}(\mathbf{x})\mathbf{U} = \mathbf{U}^\top\hat{\mathbf{\Sigma}}\mathbf{U} = \mathbf{U}^\top\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top\mathbf{U} = \begin{pmatrix} I & 0 \end{pmatrix}\mathbf{\Lambda}\begin{pmatrix} I \\ 0 \end{pmatrix} =$ top left $K \times K$ block of $\mathbf{\Lambda}$ this covariance matrix is diagonal, this means the entries are uncorrelated

5. Support Vector Machines.
Support vector machines learn a decision boundary leading to the largest margin from both classes. You are training SVM on a tiny dataset with 4 points shown below. This dataset consists of two examples with class label -1 (denoted with plus), and two examples with class label +1 (denoted with triangles).



a) Write down the SVM loss function for this data and state how to find the weight vector $\mathbf{w}$ and bias b.

Our loss function is the hinge loss shown below:

$$\min_{\mathbf{w},b} \sum_{i=1}^{N} \max\{0, 1 - t^{(i)} z^{(i)}(\mathbf{w}, b)\}$$

Since we correctly classify all 4 of our points, our loss is 0. Our weight vector $\mathbf{w} = [-1/2, -1/2]$ and bias $b = 7/2$. Notice that we choose our weight vector such that the values of $\mathbf{wx} + b = 1$ for our positive class support vectors at $[1, 4], [2, 3]$, and $\mathbf{wx} + b = -1$ for our negative class support vector at $[4, 5]$.

b) Draw the (approximate) decision boundary.

6. Probabilistic Models .

7. Probabilistic Models .
The Laplace distribution, parameterized by μ and β, is defined as follows:

$$\text{Laplace}(w; \mu, \beta) = \frac{1}{2\beta} \exp\left(-\frac{|w - \mu|}{\beta}\right).$$

Consider a variant of the homework2 question where we assume that the prior over the weights $\mathbf{w}$ consists of an independent zero-centered Laplace distribution for each dimension, with shared parameter β:

$$w_j \sim \text{Laplace}(0, \beta)$$
$$t \,|\, \mathbf{w} \sim \mathcal{N}(t; \mathbf{w}^\top \mathbf{x}, \sigma^2)$$

For reference, the Gaussian PDF is:

$$\mathcal{N}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right).$$

1. Suppose you have a labeled training set $\{(\mathbf{x}^{(i)}, t^{(i)})\}_{i=1}^N$. Give the cost function you would minimize to find the MAP estimate of $\mathbf{w}$.

   SOL : $\mathcal{L}(\sigma, \mathbf{w}, \beta) = \frac{1}{\beta} \sum_j |w_j| + \frac{1}{2\sigma^2} \sum_i (t^{(i)} - \mathbf{w}^\top \mathbf{x}^{(i)})^2$

2. Based on your answer to part (a), how might the MAP solution for a Laplace prior differ from the MAP solution if you use a Gaussian prior (which is exactly homework2)?

   SOL : The Laplace prior will encourage sparsity in the weights, since it corresponds to the L_1 norm.

8. EM Algorithm.

1. Is EM algorithm a supervised or an unsupervised learning method? Explain your answer.
   SOL: The EM algorithm is designed for unsupervised learning specifically latent variable models with a marginal likelihood that is difficult to optimize. We used the EM algorithm in order to do clustering : grouping data points into clusters, with no observed labels – ie unsupervised learning.

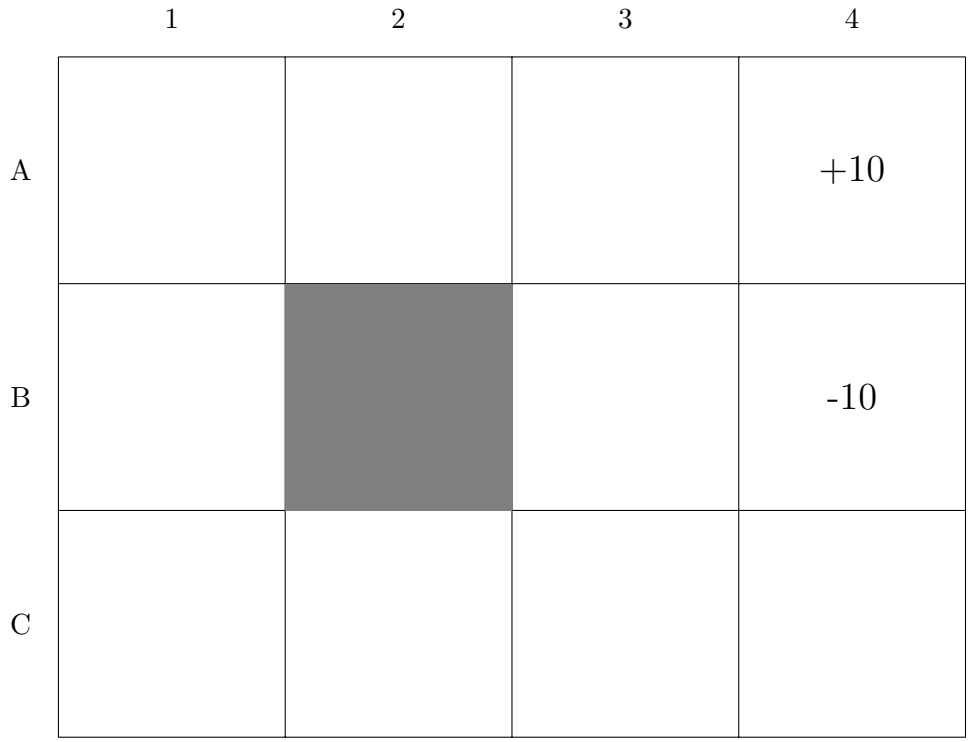
2. How does EM algorithm and k-means compare? Write 3 similarities and 3 differences.

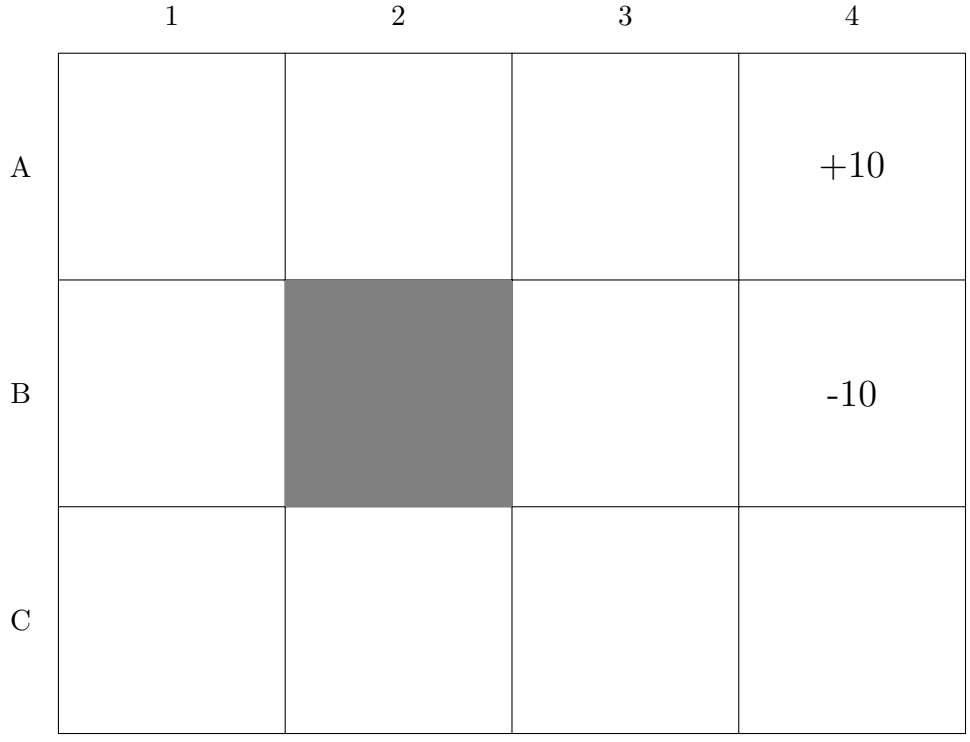
   SOL
   similarities :
   1) Both are clustering algorithms that rely on 2) alternating optimization methods and 3) can suffer from bad local optima.

   differences :
   1) in k-means the objective is the sum of squared distances of data points to their assigned cluster centers while in EM algorithm its the expected complete data likelihood
   2) in the k-means in the M step we move each cluster center to the average of the data assigned to it while in the EM algorithm we maximize the probability that it would generate the data it is currently responsible for
   3) for k-means in the E step we assign each data point to the closest cluster while for the EM algorithm we compute the posterior probability over z given our current model

3. Explain why we call these steps expectation and maximization steps. What is it that we take expectation of and what is it that we maximize?
   SOL : E-step we compute the **expectation** : $\mathbb{E}[\mathbb{I}[z = k]]$ under $p(z|\mathbf{x})$ in order to evaluate the responsibilities while in the M-step we maximize the expected complete data log-likelihood to update the model parameters

SCRATCH WORK ONLY: THIS PAGE WILL NOT BE GRADED





SCRATCH WORK ONLY: THIS PAGE WILL NOT BE GRADED

SCRATCH WORK ONLY: THIS PAGE WILL NOT BE GRADED

SCRATCH WORK ONLY: THIS PAGE WILL NOT BE GRADED