80-629 – Machine Learning for Large Scale Data Analysis and Decision Making
Exam. December 2018.

Instructor's name: Laurent Charlin

This exam has 5 questions, for a total of 85 points.

Name: ____

Student ID: ____

Question:	1	2	3	4	5	Total
Points:	17	15	22	8	23	85
Bonus Points:	0	0	0	7	0	7
Score:						

Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.

1. [17 points]
   High-level questions.

   (a) (3 points) What is the goal of ML?

   (b) (5 points) What is the difference between supervised, unsupervised, and reinforcement learning? Please be as specific as possible.

(c) (4 points) Describe what is over fitting. You are encouraged to answer using a simple sketch (a figure).

(d) (5 points) Describe the bias/variance tradeoff.

2. [15 points]
Probabilistic modelling.

(a) (5 points) What are the advantages of probabilistic modelling over non-probabilistic modelling? List at least two advantages.

(b) (3 points) Imagine that you have defined the distributions in a probabilistic model with parametric distributions (e.g., a Bernoulli distribution for 0-1 data). How would you then learn (estimate) the parameters of that probabilistic model given data?

(c) (5 points) In class we discussed a class of probabilistic models called Naive Bayes models. We applied a Naive Bayes model to collection of documents. Describe the specificities of the Naive Bayes model (you can either describe it in general or using a specific example).

(d) (2 points) Considering the previous question, in one of the Naive Bayes models we described in class we encoded each document using a bag-of-words approach. How would you adapt this Naive Bayes model if you were trying to model continuous data (e.g., if instead of word counts you wanted to model people heights)?

3. [22 points]

Neural networks and deep learning.

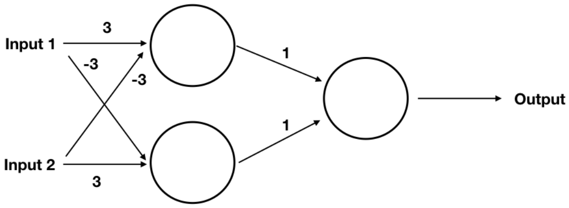
(a) (3 points) Describe in a few sentences (2 or 3) what is a neural network. In addition, list two advantages of neural networks compared to other machine learning models.

(b) (5 points) How are parameters of a neural network typically learned (estimated)?

(c) (7 points) Would it be possible to optimize the parameters of a neural network using one of the distributed computing framework we discussed in class (i.e., MapReduce or Spark)? Please properly justify your answer by proposing a way to do so.

(d) (3 points) How do you pick the appropriate activation function of the output unit of a neural network?

(e) (4 points) Consider the neural network below. We have estimated its parameters (shown next to their corresponding arrows).



The activation function of each unit in the network is a simple thresholding function:

$$\text{threshold}(x) = \begin{cases} 0 & \text{if } x \leq 0, \\ 1 & \text{if } x > 0. \end{cases} \tag{1}$$

For each of these four sets of inputs write down the network's output (i.e., its prediction) in the "Output" column of the table below.

Input 1	Input 2	Output
0	0	
1	1	
0	1	
1	0	

4. [8 points]
Distributed Computing.

(a) (4 points) How can distributed computing be useful in machine learning?

(b) (4 points) Imagine that you are running your distributed computations on a cluster of 10 computers. 9 of these nodes happen to be located in Canada while the 10th is located in Australia. Imagine that communicating with the Australian computer (i.e., transmitting information to it from the rest of the cluster) takes ten times as long as communication within two of the Canadian computers. What would be the impact of this Australian computer on computation time (e.g., on the length per iteration of your learning procedure)? It can be useful to use a specific problem here, for example doing ALS for a matrix factorization model.

(c) (7 points (bonus)) Suggest a modification to your distributed computations that would (at least) partially alleviate this problem while still using all 10 computers.

5. [23 points]
Sequential Decision Making.

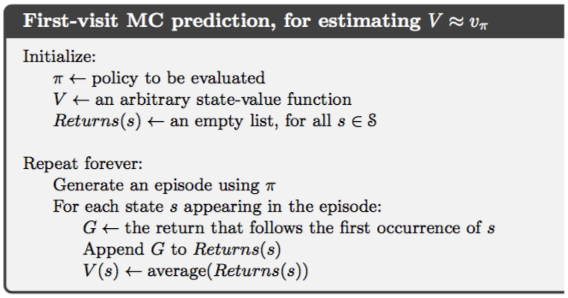
(a) (5 points) Describe what it means to solve an Markov Decision Process (MDP)? How does it differ from solving a reinforcement learning problem?

(b) (10 points) Imagine that you have observed two episodes of an agent acting in an unknown environment.

Trajectory 1: $(A, 1, 5) \rightarrow (A, 1, 5) \rightarrow (B, 2, -2) \rightarrow (A, 2, 5) \rightarrow (C, 1, -10)$;
Trajectory 2: $(B, 1, -2) \rightarrow (A, 2, 5) \rightarrow (C, 2, -10)$;

Each tuple represents a state, action, reward triplet. E.g., (A,1,5) means that the agent started in state A, executed action 1 and received a reward of 5.

Use first-visit Monte-Carlo prediction to estimate the state-value functions of states A and B. The first-visit Monte-Carlo algorithm is provided below.



First-visit MC prediction, for estimating $V \approx v_\pi$

```
Initialize:
    $\pi \leftarrow$ policy to be evaluated
    $V \leftarrow$ an arbitrary state-value function
    $Returns(s) \leftarrow$ an empty list, for all $s \in \mathcal{S}$

Repeat forever:
    Generate an episode using $\pi$
    For each state $s$ appearing in the episode:
        $G \leftarrow$ the return that follows the first occurrence of $s$
        Append $G$ to $Returns(s)$
        $V(s) \leftarrow$ average($Returns(s)$)
```

(c) You are tasked with creating a reinforcement learning agent for the game of Tic-tac-toe (see below for a game description). Your agent must control only one of the (two) players. Describe exactly how you would setup the reinforcement learning problem in terms of actions, states, and rewards.

i. (2 points) *A*ctions:

ii. (4 points) *S*tates:

iii. (2 points) *R*ewards:

Game Description from wikipedia: Tic-tac-toe (also known as noughts and crosses or Xs and Os or "morpion" or "oxo" is a paper-and-pencil game for two players, X and O, who take turns marking the spaces in a 3x3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.

The following example game is won by the first player, X (there are a total of 7 turns/images, starting from the leftmost image):