CPSC 340 Assignment 0 (due Friday September 15 ATE)

Rationale for Assignment 0: CPSC 340 is tough because it combines knowledge and skills across several disciplines. To succeed in the course, you will need:

- Math to the level of the course prerequisites: linear algebra, multivariable calculus, some probability.
- Basic Julia programming, and the ability to translate from math to programming and back.
- Statistics, algorithms, and data structures to the level of the course prerequisites.
- Some basic LaTeX skills so that you can typeset equations and submit your assignments.

The purpose of this assignment is to make sure you are prepared for this course. I anticipate that each of you will have different strengths and weaknesses, so don't be worried if you struggle with *some* aspects of the assignment. But if you find this assignment to be very difficult overall, that is an early warning sign that you may not be prepared to take CPSC 340 at this time. Future assignments will be more difficult than this one.

IMPORTANT!!!!! Before proceeding, please carefully read the homework instructions:
`www.cs.ubc.ca/~schmidtm/Courses/340-F17/assignments.pdf`

You may also want to read the answers to this Quora question as motivation:
`https://www.quora.com/Why-should-one-learn-machine-learning-from-scratch-rather-than-just-learning-to-use-the-available-libraries`

We use blue to highlight the deliverables that you must answer/do/submit with the assignment.

1 Linear Algebra Review

For these questions you may find it helpful to review these notes on linear algebra:
`http://www.cs.ubc.ca/~schmidtm/Documents/2009_Notes_LinearAlgebra.pdf`

1.1 Basic Operations

Use the definitions below,

$$\alpha = 2, \quad x = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}, \quad y = \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix}, \quad z = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}, \quad A = \begin{bmatrix} 3 & 2 & 2 \\ 1 & 3 & 1 \\ 1 & 1 & 3 \end{bmatrix},$$

and use x_i to denote element i of vector x. Evaluate the following expresions (you do not need to show your work).

1. $\sum_{i=1}^n x_i y_i$ (inner product).
2. $\sum_{i=1}^n x_i z_i$ (inner product between orthogonal vectors).
3. $\alpha(x + y)$ (vector addition and scalar multiplication).
4. $\|x\|$ (Euclidean norm of x).

5. x^T (vector tranpose).
6. A^T (matrix transpose).
7. Ax (matrix-vector multiplication).

1.2 Matrix Algebra Rules

Assume that $\{x, y, z\}$ are $n \times 1$ column vectors and $\{A, B, C\}$ are $n \times n$ real-valued matrices. State whether each of the below is true in general (you do not need to show your work).

1. $x^T y = \sum_{i=1}^n x_i y_i$.
2. $x^T x = \|x\|^2$.
3. $x^T(y + z) = z^T x + x^T y$.
4. $x^T(y^T z) = (x^T y)^T z$.
5. $AB = BA$.
6. $A(B + C) = AB + AC$.
7. $(AB)^T = A^T B^T$.
8. $x^T Ay = y^T A^T x$.
9. $\det A \neq 0 \iff A$ is invertible.

1.3 Special Matrices

In one sentence, write down the defining properties of the following special types of matrices:

1. Symmetric matrix.
2. Identity matrix.
3. Orthogonal matrix.

2 Probability Review

For these questions you may find it helpful to review these notes on probability:
`http://www.cs.ubc.ca/~schmidtm/Courses/340-F15/notes_probability.pdf`

2.1 Rules of probability

Answer the following questions. You do not need to show your work.

1. You flip 4 fair coins. What is the probability of observing exactly 3 heads?
2. You are offered the opportunity to play the following game: your opponent rolls 2 regular 6-sided dice. If the difference between the two rolls is at least 3, you win \$12. Otherwise, you get nothing. What is a fair price for a ticket to play this game once? In other words, what is the expected value of playing the game?

3. Consider two events A and B such that $\Pr(A, B) = 0$. If $\Pr(A) = 0.4$ and $\Pr(A \cup B) = 0.95$, what is $\Pr(B)$? Note: $p(A, B)$ means "probability of A and B" while $p(A \cup B)$ means "probability of A or B". It may be helpful to draw a Venn diagram.

2.2 Bayes Rule and Conditional Probability

Answer the following questions. You do not need to show your work.

Suppose a drug test produces a positive result with probability 0.95 for drug users, $P(T = 1|D = 1) = 0.95$. It also produces a negative result with probability 0.99 for non-drug users, $P(T = 0|D = 0) = 0.99$. The probability that a random person uses the drug is 0.0001, so $P(D = 1) = 0.0001$.

1. What is the probability that a random person would test positive, $P(T = 1)$?
2. In the above, do most of these positive tests come from true positives or from false positives?
3. What is the probability that a random person who tests positive is a user, $P(D = 1|T = 1)$?
4. Are your answers from part 2 and part 3 consistent with each other?
5. What is one factor you could change to make this a more useful test?

2.3 Bayes Rule and Independence

On a game show, a contestant is told the rules as follows:

> There are three doors, labelled 1, 2, 3. A single prize has been hidden behind one of them. You get to select one door. Initially your chosen door will *not* be opened. Instead, the gameshow host will open one of the other two doors, and *he will do so in such a way as not to reveal the prize.* For example, if you first choose door 1, he will then open one of doors 2 and 3, and it is guaranteed that he will choose which one to open so that the prize will not be revealed.
>
> At this point, you will be given a fresh choice of door: you can either stick with your first choice, or you can switch to the other closed door. All the doors will then be opened and you will receive whatever is behind your final choice of door.

Imagine that the contestant chooses door 1 first; then the gameshow host opens door 3, revealing nothing behind the door, as promised. Should the contestant (a) stick with door 1, or (b) switch to door 2, or (c) does it make no difference? Show your reasoning/calculations. Assume that initially the prize is equally likely to be behind any of the 3 doors, that the host always opens a door that doesn't contain a prize, and that if the prize is behind the selected door that the host is equally likely to choose door 2 or 3.

3 Calculus Review

For these questions you may find it helpful to review this list of useful Julia commands (which I'll update through the term):
`http://www.cs.ubc.ca/~schmidtm/Courses/340-F17/juliaCommands.txt`

3.1 One-variable derivatives

Answer the following questions. You do not need to show your work.

1. Find the minimum value of the function $f(x) = 3x^2 - 2x + 5$ for $x \in \mathbb{R}$.
2. Find the maximum value of the function $f(x) = x(1-x)$ for x in the interval $[0,1]$.
3. Find the minimum value of the function $f(x) = x(1-x)$ for x in the interval $[0,1]$.
4. Let $p(x) = \frac{1}{1+\exp(-x)}$ for $x \in \mathbb{R}$. Compute the derivative of the function $f(x) = -\log(p(x))$ and simplify it by using the function $p(x)$.

Remember that in this course we will $\log(x)$ to mean the "natural" logarithm of x, so that $\log(\exp(1)) = 1$. Also, obseve that $p(x) = 1 - p(-x)$ for the final part.

3.2 Multi-variable derivatives

Compute the gradient $\nabla f(x)$ of each of the following functions. You do not need to show your work.

1. $f(x) = x_1^2 + \exp(x_2)$ where $x \in \mathbb{R}^2$.
2. $f(x) = \exp(x_1 + x_2 x_3)$ where $x \in \mathbb{R}^3$.
3. $f(x) = a^T x$ where $x \in \mathbb{R}^2$ and $a \in \mathbb{R}^2$.
4. $f(x) = x^\top A x$ where $A = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}$ and $x \in \mathbb{R}^2$.
5. $f(x) = \frac{1}{2}\|x\|^2$ where $x \in \mathbb{R}^d$.

Hint: it is helpful to write out the linear algebra expressions in terms of summations.

3.3 Derivatives of code

The zip file `a0.zip` contains a Julia file named `grads.jl` which defines several functions. Complete the functions `grad1`, `grad2`, and `grad3` (which compute the gradients of `func1`, `func2`, and `func3`). Include the code in PDF file for this section, and also in your zip file.

Hint: for many people it's easiest to first understand on paper what the code is doing, then compute the gradient, and then translate this gradient back into code. We've given you `func0` and `grad0` as an example. Also, we've provided the function `numGrad` which approximates the gradient numerically. Below is an example of using these functions:

```
julia> cd("a0")

julia> include("grads.jl")
numGrad (generic function with 1 method)

julia> func0([10 2])
104

julia> grad0([10 2])
1x2 Array{Int32,2}:
 20  4

julia> numGrad(func0,[10 2])
2-element Array{Float64,1}:
 20.0
  4.0
```

Note: do not worry about the distinction between row vectors and column vectors here. For example, if the correct answer is a vector of length 5, we'll accept vectors of size 5×1 or 1×5. In future assignments we will be more careful to always use column vectors.

4 Algorithms and Data Structures Review

For these questions you may find it helpful to review these notes on big-O notation:
`http://www.cs.ubc.ca/~schmidtm/Courses/340-F15/notes_BigO.pdf`

4.1 Trees

Answer the following questions You do not need to show your work.

1. What is the maximum number of *leaves* you could have in a binary tree of depth l?
2. What is the maximum number of *internal nodes* (excluding leaves) you could have in a binary tree of depth l?

4.2 Common Runtimes

Answer the following questions using big-O notation You do not need to show your work.

1. What is the cost of running the mergesort algorithm to sort a list of n numbers?
2. What is the cost of finding the third-largest element of an unsorted list of n numbers?
3. What is the cost of finding the smallest element greater than 0 in a *sorted* list with n numbers.
4. What is the cost of computing the matrix-vector product Ax when A is $n \times d$ and x is $d \times 1$.
5. How does the answer to the previous question change if A has only z non-zeroes.

4.3 Running times of code

Included in `a0.zip` is file named `bigO.jl`, which defines several functions that take an integer argument n. For each function, state the running time as a function of n, using big-O notation.