Homework 0
Due: Thursday, January 31, 2019 at 12:00pm (Noon)

Introduction: Solidifying Background
The purpose of this portion is to fortify your background in probability and statistics, linear algebra, and algorithmic analysis. The topics explored here, in addition to the practical skills developed in the programming section, will be used many times throughout this course.

- Note 1: This problem set is not meant to take more than 3 hours. If you are not confident with any of the sections, or find yourself stuck, please make use of tutorials, Piazza, and TA hours.
- Note 2: You may be able to find answers to these problems by searching the problem text. Please search instead for the concepts being applied; the goal is not to solve these specific problems, but to be comfortable with the principles that will be applied later in the course.

Problem 1: Bayes’ Rule
Bayes’ Rule, or Bayes’ Theorem is an oft-used identity coming from probability theory. If we have two events of interest, $A$ and $B$, we might want to ask what the probability of $B$ is, given that we know $A$ happened.

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Note that this is the same as

$$P(B|A) = \frac{P(A \cap B)}{P(A)}.$$

Later in this course, the parts of this formula may be relabeled:

$$Posterior = \frac{Likelihood \ast Prior}{Evidence}$$

This rule will be explicitly used in Bayesian algorithms, but it is also a principle that will implicitly underlie almost all of our machine learning algorithms. This problem consists of four parts, each worth 3 points (1 point if the answer is correct and an additional 2 points for showing correct work). As a hint, none of the four parts have the same answer. For the purposes of this question, assume that whales have equal probability of calving a male or female, and uniform probability of being born on any day of the week. Fun whale fact: No whale has ever been observed giving birth to twins.

Part 1
(3 points)
Suppose a whale has two offspring. What is the probability that both offspring are male?

Part 2
(3 points)
Suppose a whale has two offspring and the eldest is male. What is the probability that both offspring are male?
Part 3

(3 points)
Suppose a whale has two offspring and at least one is male. What is the probability that both offspring are male?

Part 4

(Extra Credit: 3 points)
Suppose a whale has two offspring and at least one is a male whale born on a Wednesday. What is the probability that both offspring are male?

Problem 2: Linear Algebra - Singular Value Decomposition

One goal of a singular value decomposition is to represent a large matrix as a product of smaller ones. See Figure 1.

Figure 1: Singular Value Decomposition (SVD) represents a large matrix as a product of smaller ones. Note that the columns of $U$ and $V$ are orthonormal, and $D$ is a real-valued diagonal matrix.

If we have an $n \times d$ matrix of $n$ datapoints each in $d$ dimensions, SVD finds a $d \times r$ matrix $V$ that allows us to project the data to a smaller number of dimensions. This is the basis (pun intended) of Principal Component Analysis, as well as several other algorithms we will cover this semester. Note that $V$ and $U$ are orthonormal. An orthonormal matrix has the property that its columns are mutually orthogonal (the dot product of any pair of distinct columns is 0) and normalized (the dot product of any column with itself is 1). The combination of these properties means that the product of an orthonormal matrix’s transpose with that matrix is an identity matrix.

$$V^T \times V = I$$
$$U^T \times U = I$$

This problem has 3 parts, each worth 3 points. Points will be awarded for clean derivations/proofs and for the compactness of the result (that is, the result should be expressed in simplest terms). Assume that expressions may freely include the transpose of any matrix, the identity matrix, and the zero matrix, in addition to any given terms.

Part 1

(3 points)
Let $\begin{bmatrix} A \\ A \end{bmatrix}$ be a $2n \times d$ matrix made by extending $A$ with itself. Given a SVD of matrix $A$ into $U \times D \times V^T$, 

$$\begin{bmatrix} A \\ A \end{bmatrix} = U \times D \times V^T$$

Figure 1: Singular Value Decomposition (SVD) represents a large matrix as a product of smaller ones. Note that the columns of $U$ and $V$ are orthonormal, and $D$ is a real-valued diagonal matrix.
where \( D \) is a diagonal matrix with dimension \( r \), express \( \text{SVD}(\begin{bmatrix}A \\ A \end{bmatrix}) \) in terms of \( U, D, \) and \( V \). Provide an explanation for your answer.

**Part 2**

(3 points)

Given \( A = U_A \times D_A \times V_A^T \), use \( V_A \) to reduce the dimensionality of \( A \). That is, let \( B = A \times V_A \). Express the \( \text{SVD}(B) \) in terms of \( U_A, D_A, \) and \( V_A \). Provide a justification for your answer.

**Part 3**

(3 points)

Suppose we have \( A = U_A \times D_A \times V_A^T \), and we let \( B = A \times V_A \). In the same way, let \( C = B \times V_B \). Write a SVD of \( C \) in terms of \( U_A, D_A, \) and \( V_A \). Is there a maximum to the number of times the dimensionality of a dataset can be reduced using SVD?

**Problem 3: Runtime Complexity**

(12 points)

The Fibonacci sequence is defined as \( F(n) = F(n-1) + F(n-2) \) for \( n \geq 2 \) where \( F(0) = 0 \) and \( F(1) = 1 \). We can make use of the fact (likely first noted by Edsger Dijkstra) that

\[
\begin{pmatrix}
1 & 1 \\
1 & 0
\end{pmatrix}^n = \begin{pmatrix}
F(n+1) & F(n) \\
F(n) & F(n-1)
\end{pmatrix}
\]

for any positive integer \( n \). So to compute \( F(n) \) we can compute \( \left( \begin{pmatrix}1 & 1 \\ 1 & 0 \end{pmatrix} \right)^{n-1} \) and return the upper left number. Describe an algorithm to compute \( F(n) \) for arbitrary \( n \) that uses fewer than \( O(n) \) additions and multiplications (don’t worry about the size of the integers), and prove that it satisfies this complexity bound. That is, prove a sublinear complexity bound in terms of \( n \) on the number of additions and multiplications this algorithm performs. Feel free to search for such an algorithm, but please generate the proof of the complexity bound yourself.

**Problem 4: Numpy, Scipy, and Matplotlib**

**Introduction**

The purpose of this section is to introduce you to some tools that you will find useful and/or necessary in order to complete future homeworks. By the end of this assignment, you will have used numpy to perform efficient computations, loaded standard datasets using SciPy, and used matplotlib to visualize several performance metrics you will be using this semester. This homework also serves as an environment/install test and will get you familiar with the hand-in process for physical documents.

**Installing Anaconda**

Python 3.7, numpy, scipy, and matplotlib are considered necessary in order to do the homeworks in the remainder of this course. Fortunately, all of these can be installed on your own machine without admin privileges in a single package, called Anaconda. You can follow the directions at www.anaconda.com/download. Be sure to use Python version 3.7. For your convenience, we also have a course-wide virtual environment set up on the department machines at /course/cs1420/cs142_env. It can be activated from your own folder by running: source /course/cs1420/cs142_env/bin/activate. After this, you can run your program with the necessary environment/packages.
Part 1: Matplotlib

(8 points)
The code for this homework assignment contains the file hw0.py. This contains two functions you need to fill out, graph_iris_data and graph_series_data. In order to fill out graph_iris_data. Note that the plt.show() call should be the last call, so add all of your graph customization below the TODO, but above the show() call!

1. call plt.scatter(x, y, c=None), giving it the following arguments:
   (a) the x argument will be xs
   (b) the y argument will be ys
   (c) in order to give the plotted points color, we will specify the optional c argument. Thus, we will pass a third argument, c=iris.target

2. Look into the matplotlib.pyplot documentation and learn how to add titles and axis labels to plots. Add a title to the plot of the form "Made by: [Course ID]" where [Course ID] is replaced with your unique identifier for this class.

3. Examine the column names of the iris data to find appropriate x and y labels for the plot, and use pyplot commands to label the axes of the two iris plots.

We will use a similar process for graph_series_data.

1. call plt.plot(x, y, format), giving it the following arguments:
   (a) the x argument will be xs
   (b) the y argument will be y1s
   (c) the format argument will be ’.r’

2. call plt.plot(x, y, format) again, giving it the following arguments:
   (a) the x argument will be xs
   (b) the y argument will be y2s
   (c) the format argument will be ’-b’

3. Add a title to the plot of the form "Made by: [Course ID]" where [Course ID] is replaced with your unique identifier for this class.

4. Use pyplot commands to add a legend to the series data plot, where each series is labeled with its function. Look here to start!

Afterwards, print the two graphs and turn these in with the rest of the assignment. Each graph is worth 4 points.

Numpy

Answer the following questions using numpy functions. Note that when importing numpy, it is often abbreviated to np (i.e., import numpy as np), so when calling numpy functions, you can use np.[function]. Further, note that you may NOT use the np array constructor to solve these questions (i.e., np.array(...)). Some functions you may want to consider to approach this problem are np.arange, np.zeros, np.ones, np.eye, np.sum, np.hstack, np.vstack, np.transpose, np.matmul, np.inner, np.where and np.dot

1. (1 point) Using numpy, how would you create a 1D array containing the values 2 through 6?
2. (1 point) Using numpy, how would you create a 4x4 matrix where all the values are 1?

3. (1 point) Using numpy, how would you create a 6x6 identity matrix?

4. (1 point) Using numpy, how would you sum the values of each column of matrix A?

5. (1 point) Using numpy, given the matrices A and B, how would you find the matrix C, where C = A^T B?

6. (2 points) Using numpy, using either np.vstack or np.hstack, how would you create a 4x2 matrix where all the values in the first column are 0's and all the values in the second column are 1's?

7. (2 points) Given a matrix of floats A, using numpy, how would you return an array of the same shape, where all values > 3.0 are set to 1, and the rest to 0?

Note that with anaconda set up, you have numpy downloaded, so you can check your answers by running python from the command line!

**Handing In**

**Important:** You will need to fill out the Collaboration Policy Form available on the course website in order to hand in this assignment.

You will turn in your final handin via Gradescope, as detailed in the email sent to the course. If you have questions on how to set up or use Gradescope, ask on Piazza! For this assignment, you should have written answers for Questions 1, 2, 3, and the Numpy part of question 4. For the Matplotlib portion of question 4, you should turn in your two graphs.

**Obligatory Note on Academic Integrity**

Plagiarism — don’t do it.

As outlined in the [Brown Academic Code](https://www.brown.edu/about/administration/policies/academic-code), attempting to pass off another’s work as your own can result in failing the assignment, failing this course, or even dismissal or expulsion from Brown. More than that, you will be missing out on the goal of your education, which is the cultivation of your own mind, thoughts, and abilities. Please review this course’s collaboration policy and if you have any questions, please contact a member of the course staff.