CMSC 25300/35300, STAT 27700 (Spring 2023)
Homework 1: Vectors and Matrices

Submission Instructions:

- Please submit your homework in PDF to Gradescope (which can be accessed from the course's website on Canvas);
- Please paste your code in the submitted PDF. In other words, your submission should be a single PDF that contains both your writing solutions and your code.
- Note that you do not need to copy the problem statements in your solution, *as long as you clearly indicates the problem numbers* (e.g., 1.a, 2.c, etc).

1. Reading assignment. Read Boyd VMLS chapters 1 & 6.

2. Matrix multiplication. You run a small bakery that serves breakfast. You decide to look at products you need for making the most popular breakfast meals. You can use a matrix based on the following data: for omelette, you need two units of eggs, two units of milk, 0.5 units of butter; for pancakes, you need one unit of eggs, two units of milk, 0.5 units of butter, one unit of flour; for muffins, you need one unit of eggs, one unit of butter, three units of flour, and three units of berries. These ingredients are required for one portion of a meal.

a) (5 points) Write this information as a matrix. What do the rows represent? What do the columns represent?

b) (3 points) One unit of eggs is \$1, a unit of milk is \$0.2, a unit of butter is \$0.5, a unit of flour is \$0.1, a unit of berries is \$0.4. Write a matrix-vector multiplication that calculates expenses on every breakfast meal.

c) (3 points) How would you calculate total amount of ingredients needed for one portion of omelette and pancakes? You should use matrix multiplication again.

d) (3 points) Estimate your total expenses to make three portions of every meal (using, you guessed it, matrix multiplication).

e) (10 points) Get up and running with Python (or a similar language of your choice). Write a script that computes the matrix multiplications in the previous parts of this problem.

3. You are given a matrix

$$\mathbf{X} = \begin{bmatrix} 4 & 1 & 1 \\ -3 & 2 & 2 \\ 1 & -3 & 2 \\ 5 & 1 & 3 \end{bmatrix} \in \mathbb{R}^{4\times 3}.$$

a) (3 points) Assume $\mathbf{y} = \begin{bmatrix} 9 & 7 & 1 & 16 \end{bmatrix}^\top$. Does $\mathbf{w} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^\top$ satisfy the equation $\mathbf{X}\mathbf{w} = \mathbf{y}$?

b) (3 points) Given a number $i \in \{1, 2, 3, 4\}$, how would you construct a vector $\mathbf{w} \in \mathbb{R}^4$ so that $\mathbf{w}^\top \mathbf{X}$ is the i-th row of $\mathbf{X}$?

c) (3 points) How would you construct a vector $\mathbf{w} \in \mathbb{R}^4$ so that $\mathbf{w}^\top \mathbf{X}$ is a times the i-th row of $\mathbf{X}$ plus b times the j-th row of $\mathbf{X}$ for some $a, b \in \mathbb{R}$ and $j, k \in \{1, \ldots, 4\}$?

d) (3 points) A similar question but for columns: given a number $i \in \{1, 2, 3\}$, how would you construct a vector $w \in \mathbb{R}^3$ so that $\mathbf{X}\mathbf{w}$ is the i-th column of $\mathbf{X}$?

e) (3 points) Construct a vector $\mathbf{w} \in \mathbb{R}^3$ so that $\mathbf{X}\mathbf{w}$ is the 2-nd column of $\mathbf{X}$ multiplied by 10 minus 1-st column.

f) (3 points) Find $\mathbf{XB}$, where $\mathbf{B} = \begin{bmatrix} 0 & 1 & -1 & 1 \\ 2 & 3 & 1 & 1 \\ 0 & 2 & -2 & 2 \end{bmatrix}$.

g) (8 points) Write a Python script to verify your answers in (a)-(e).

4. Matrix rank. Solve the following problems:

a) (5 points) What's the rank of aa^T, where $a = [1\ 1\ 2]^\top$? (Hint: you don't need any tools beyond what was discussed in class.)

b) (5 points) This matrix has rank = 2:

$$\begin{bmatrix} 1 & 5 & -0.5 \\ 3 & 3 & -1.5 \\ 9 & 1 & -4.5 \\ 4 & 10 & -2 \end{bmatrix}.$$

Explain why. (Hint: you don't need any tools beyond what was discussed in class.)

5. (15 points) You learn a model that says:

$$\hat{y}_i = w_1 x_{i,1} + w_2 x_{i,2} + w_3$$

$$\text{predicted label } = \begin{cases} +1, & \hat{y}_i > 0 \\ -1, & \text{otherwise} \end{cases}.$$

Suppose that you know points $\mathbf{x} = (1, 10)$ and $(6, -5)$ are on the decision boundary, and the predicted label for $(5, 0)$ is $+1$, find $\mathbf{w} = \begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix}^\top$, draw a plot and indicate for which $\mathbf{x}$ your model would predict the label $+1$. Show two examples of $\mathbf{x}$ that are correctly classified by the model.

6. **Polynomials using linear models.** Suppose we observe pairs of scalar points (z_i, y_i), $i = 1, \ldots, n$. Imagine these points are measurements from a scientific experiment. The variables z_i are the experimental conditions with two dimensions, i.e. $\mathbf{z_i} = (z_{i,1}, z_{i,2})$ and the y_i correspond to the measured response in each condition. Suppose we wish to fit a degree 2 polynomial to these data. In other words, we want to find the coefficients of a degree 2 polynomial p so that $p(z_i) \approx y_i$ for $i = 1, 2, \ldots, n$. We want to use a linear model.

   a) (5 points) Suppose p is a degree 2 polynomial. Write the general expression for $p(z) = y$. (Hint: include the interaction terms $z_{i,1}^a z_{i,2}^b$, where integers $a, b >= 0$ and $a + b <= 2$.)

   b) (5 points) Express the $i = 1, \ldots, n$ equations as a system in matrix form $\boldsymbol{Xw} = \boldsymbol{y}$. Specifically, what is the form/structure of $\boldsymbol{X}$ in terms of the given z_i.

   c) (15 points) Write a Python script to generate a plot of a polynomial given $\mathbf{w}$ and points $z_1, \ldots, z_n$. Here is some starter code and you are asked to fill in the TODOs:

```
import numpy as np
import scipy . io as sio
import matplotlib . pyplot as plt
from mpl_toolkits import mplot3d

# n = number of points
# z = points where polynomial is evaluated
# p = array to store the values of the interpolated polynomials
n = 100
z_1 = np.linspace(-1, 1, n)
z_2 = np.linspace(-1, 1, n)

w_size = ##TODO
w = np.random.rand(w_size)
X = np.zeros((n,w_size))

# TODO: generate X-matrix

# TODO: evaluate polynomial at all points z,
# and store the result in p
# do NOT use a loop for this

# plot the datapoints and the best-fit polynomials
fig = plt.figure()
# syntax for 3-D projection
ax = plt.axes(projection ='3d')

ax.plot3D(z_1,z_2, p, 'green')
ax.set_xlabel("z_1")
ax.set_ylabel("z_2")
ax.set_zlabel("y")
```

```
ax.set_title('polynomial with coefficients w=%s'%w)
plt.show()
```