VMLS - Study Guide Matrices Chapter 6 Name: Ben Chen

Always include this title page with your PDF. Include your name above.

- Submit your work in Gradescope as a PDF - you will identify where your “questions are.”
- Identify the question number as you submit. Since we grade "blind" if the questions are NOT identified, the work WILL NOT BE GRADED and a 0 will be recorded. Always leave enough time to identify the questions when submitting.
- One section per page (if a page or less) - We prefer to grade the main solution in a single page, extra work can be included on the following page.
- Long instructions may be removed to fit on a single page.
- **Do not start a new question in the middle of a page.**
- Solutions to book questions are provided for reference.
- You may NOT submit given solutions - this includes minor modifications - as your own.
- Solutions that do not show individual engagement with the solutions will be marked as no credit and can be considered a violation of honor code.
- If you use the given solutions you must reference or explain how you used them, in particular...

For full credit, EACH book exercise in the Study Guides must use one or more of the following methods and FOR EACH QUESTION. Identify the number the method by number to ensure full credit.

Method 1 - Provide original examples which demonstrate the ideas of the exercise in addition to your solution.

Method 2 - Include and discuss the specific topics needed from the chapter and how they relate to the question.

Method 3 - Include original Python code, of reasonable length (as screenshot r text) to show how the topic or concept was explored.

Method 4 - Expand the given solution in a significant way, with additional steps and comments. All steps are justified. This is a good method for a proof for which you are only given a basic outline.

Method 5 - Attempt the exercise without looking at the solution and then the solution is used to check work. Words are used to describe the results.

Method 6 - Provide an analysis of the strategies used to understand the exercise, describing in detail what was challenging, who helped you or what resources were used. The process of understanding is described.

"There is a minor typo in VLMS pg. 119: A1 is the sum of the rows of A, not the columns"

Method #5

1. Pick a section of Chapter 6 to annotate. (10 pts)

Matrices

In this chapter we introduce matrices and some basic operations on them. We give some applications in which they arise.

6.1 Matrices

A *matrix* is a rectangular array of numbers written between rectangular brackets, as in

$$\begin{bmatrix} 0 & 1 & -2.3 & 0.1 \\ 1.3 & 4 & -0.1 & 0 \\ 4.1 & -1 & 0 & 1.7 \end{bmatrix}.$$

Same but which better?

It is also common to use large parentheses instead of rectangular brackets, as in

$$\begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{pmatrix} 0 & 1 & -2.3 & 0.1 \\ 1.3 & 4 & -0.1 & 0 \\ 4.1 & -1 & 0 & 1.7 \end{pmatrix}.$$

An important attribute of a matrix is its *size* or *dimensions*, *i.e.*, the numbers of rows and columns. The matrix above has 3 rows and 4 columns, so its size is 3×4. A matrix of size $m \times n$ is called an $m \times n$ matrix.

The *elements* (or *entries* or *coefficients*) of a matrix are the values in the array. The i, j element is the value in the ith row and jth column, denoted by double subscripts: the i, j element of a matrix A is denoted A_{ij} (or $A_{i,j}$, when i or j is more than one digit or character). The positive integers i and j are called the (row and column) *indices*. If A is an $m \times n$ matrix, then the row index i runs from 1 to m and the column index j runs from 1 to n. Row indices go from top to bottom, so row 1 is the top row and row m is the bottom row. Column indices go from left to right, so column 1 is the left column and column n is the right column.

If the matrix above is B, then we have $B_{13} = -2.3$, $B_{32} = -1$. The row index of the bottom left element (which has value 4.1) is 3; its column index is 1.

Two matrices are equal if they have the same size, and the corresponding entries are all equal. As with vectors, we normally deal with matrices with entries that

$i \quad j$

B_{32}

not

B_{30}

+

B_2

Method #5

2. (10 pts) Solve and explain the solution to 6.3 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

6.3 *Block matrix.* Assuming the matrix

$$K = \begin{bmatrix} I & A^T \\ A & 0 \end{bmatrix}$$

makes sense, which of the following statements must be true? ('Must be true' means that it follows with no additional assumptions.)

(a) K is square.

(b) A is square or wide.

(c) K is symmetric, *i.e.*, $K^T = K$.

(d) The identity and zero submatrices in K have the same dimensions.

(e) The zero submatrix is square.

b = rows, c = columns

a) is true because the number of rows must equal the number of columns in order for K to be true. "I" is an identity of matrix, so it must be square (like b x b). A transpose must have b rows and some c columns. A must have c rows and b columns since A transpose has c columns and b rows. The zero submatrix msut then be c x c to match the size of A. K is (b + c) x (b + c) matrix

b) is not true because A can be tall, square, or wide without contradicting the structure of K. There is no requirement from the structure of K that b >= c.

c) is true because the transpose of K was computed using the block matrix transpose rule so all the parts of K satisfy symmetry conditions as in identity matrix I is symmetric, zero matrix 0 is symmetric, placement of A and A transpose natrually satisfies the symmetry. K transpose = K, so c is true.

d) is not necessarily true because identity matrix I is b x b, zero matrix is c x c, no requirement of K enforces b = c. Identity and zero matrix might be different sizes.

e) is true because the zero matrix is placed in the bottom right of K. For K to be square, this zero submatrix must be square (balancing structure of K) so if the zero submatrix was not square, K couldn't be a square shape.

Method #5

There is a minor typo in VMLS exercise 6.9: Should be n by m matrix not m by n.

3. (20 pts) Solve and Explain the solution to 6.9 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

6.9 *Multiple channel marketing campaign.* Potential customers are divided into m market segments, which are groups of customers with similar demographics, *e.g.*, college educated women aged 25–29. A company markets its products by purchasing advertising in a set of n channels, *i.e.*, specific TV or radio shows, magazines, web sites, blogs, direct mail, and so on. The ability of each channel to deliver impressions or views by potential customers is characterized by the *reach matrix*, the $m \times n$ matrix R, where R_{ij} is the number of views of customers in segment i for each dollar spent on channel j. (We assume that the total number of views in each market segment is the sum of the views from each channel, and that the views from each channel scale linearly with spending.) The n-vector c will denote the company's purchases of advertising, in dollars, in the n channels. The m-vector v gives the total number of impressions in the m market segments due to the advertising in all channels. Finally, we introduce the m-vector a, where a_i gives the profit in dollars per impression in market segment i. The entries of R, c, v, and a are all nonnegative.

(a) Express the total amount of money the company spends on advertising using vector/matrix notation.

(b) Express v using vector/matrix notation, in terms of the other vectors and matrices.

(c) Express the total profit from all market segments using vector/matrix notation.

(d) How would you find the single channel most effective at reaching market segment 3, in terms of impressions per dollar spent?

(e) What does it mean if R_{35} is very small (compared to other entries of R)?

a) R is n x m where n = # of channels, m = # of market segments, Rij = views of customers in segment j per dollar spent on channel i, and c is n-vector (spending per channel), v = m-vector (total impressions in each market segment), a = m-vector (profit per impresssion per segment)

total spending = 1 transpose c where 1 is a n-vector of all 1's, and 1 transpose c will compute the sum of all elements in c where ci represents the amount spent on channel i, and the total amount spent is the sum of all elements in c.

b) v = Rc because r is a n x m matrix of channels x segments where c is a m-vector representing budget/spending per channel. Rc gives the n-vector v representing the impressions per segment.

c) a transpose v = a transpose Rc because v = Rc gives the impressions per segment. a is an n-vector of profit per impression so a transpose v computes the total profit

d) argmaxj R3j is correct because R3j represents the impressions per dollar for channel j in segment 3. The best channel is the one with the maximum R3j so the argmax is there to calculate that.

e) R35 is very small compared to other entries of R would mean that the 5th channel makes a lot less impressions on the third market segment per dollar spent compared to the other channels, which is accurate because R35 represents the impressions per dollar for channel 5 in segment 3, a small R35 implies inefficiency in reaching segment 3 with channel 5

Method #5

4. (20pts) Solve and Explain the solution to 6.13 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

6.13 *Polynomial differentiation.* Suppose p is a polynomial of degree $n-1$ or less, given by $p(t) = c_1 + c_2 t + \cdots + c_n t^{n-1}$. Its derivative (with respect to t) $p'(t)$ is a polynomial of degree $n-2$ or less, given by $p'(t) = d_1 + d_2 t + \cdots + d_{n-1} t^{n-2}$. Find a matrix D for which $d = Dc$. (Give the entries of D, and be sure to specify its dimensions.)

How matrix D behaves relative to c

for $d_i = i \cdot c_{i+1}$, matrix D multiplies each c_i by i and shift down one index

$$D = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & (n-1) \end{bmatrix}$$

D is $(n-1) \times n$ dimensions

derivative $= c_2 + 2c_3 t + \cdots + c_n(n-1)t^{n-2}$

$0_{(n-1)\times 1}$ adds col. of zeros ~~on~~ the lef

$\Rightarrow c_1$ no effect on d

diag$(1,2,\ldots,n-1)$ is diag matrix with $1,2,\ldots\ n-1$

$\Rightarrow$ multiplies each c_{i+1} by i

leftmost col. is ~~the~~ $(n-1)\times 1$ zeros

right part is $(n-1)(n-1)$ diag matrix

scales coefficient by i

$d = Dc$ produces coefficients.

✓ $d_i = i \cdot c_{i+1}$

Method #5

5. (20pts) Solve and Explain the solution to 6.17 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

6.17 *Stacked matrix.* Let A be an $m \times n$ matrix, and consider the stacked matrix S defined by

$$S = \begin{bmatrix} A \\ I \end{bmatrix}.$$

When does S have linearly independent columns? When does S have linearly independent rows? Your answer can depend on m, n, or whether or not A has linearly independent columns or rows.

Let's break down Matrix S: - A is an m x n matrix, I is an n x n matrix, and the stacked S is going to have dimensions of (m+n) x n, where S has columns and m+n rows,

If we want to find when the columns of S are linearly independent, we have to recognize that this only happens when Sx=0 when x = 0. Since I is a part of S, any vector x in the null space of S must also have x = 0. A must have linearly independent columns and if A has dependent columns, S will also. If A has full column rank(A) =n, then S will have ind. columns. m >= n is not a requirement here as long as A has a full column rank, S has ind. columns regardless of m.

If we want to find when ros of S are linearly dependent, we have to recognize that rows of S are ind. if S has a full row rank(S) = m + n. Matrix I contributes n ind. rows. A must have linearly ind. rows, must have full row rank(A) = minimum of (m,n) and m <= n ensures that adding I does not create dependencies. Rows of S are linearly ind. if A has full row rank and m <= n.

Combine the two, and S has independent rows and columns.

Method #5

6. (20pts) Solve and Explain the solution to 6.22 here in your own words. (Since you are given a solution, you will be graded on your ability to explain).

6.22 *Distribute or not?* Suppose you need to compute $z = (A+B)(x+y)$, where A and B are $m \times n$ matrices and x and y are n-vectors.

(a) What is the approximate flop count if you evaluate z as expressed, *i.e.*, by adding A and B, adding x and y, and then carrying out the matrix-vector multiplication?

(b) What is the approximate flop count if you evaluate z as $z = Ax + Ay + Bx + By$, *i.e.*, with four matrix-vector multiplies and three vector additions?

(c) Which method requires fewer flops? Your answer can depend on m and n. *Remark.* When comparing two computation methods, we usually do not consider a factor of 2 or 3 in flop counts to be significant, but in this exercise you can.

A) flop count for z = (A+B)(x+y)

A and B are m by n matrices, adding these two matrices of this size requires m x b additions so the flops are m x n. Then x + y are n-vectors, adding these two vectors of size n requires n additions. Then (A+B)(x+y) is matrix-vector multiplication for m x n matrix and n-vector where it is 2mn flops to perform mn multiplication and mn additions. So the total is mn + n + 2mn = 3mn + n.

B) flop count for z = Ax + Ay + Bx + By

For Ax, Ay, Bx, By, each matrix-vector multiplication (m x n) x n-vector requires 2mn flops there are 4 multiplication operations so it will be 4 x 2mn = 8mn. There are 3 vector additions with m-size vectors where each of the additions require m-flops so it's 3m.

The total flop count would be 8mn + 3m.

C) Which method requires less flops?

The first method A) uses less flops for most values of m and n. For a large m and n, the dominant term would be mn. 3mn + n is a lot smaller than 8mn + 3m and so that means method a generally requires less flops and the difference is bigger as the values of m and n are bigger. 3mn + n > 8mn + 3m can also be true, but only if m is < 1/5 and n is very very large. But typically, m >= 1/5 and so we can say that method A) is more efficient.

My answer is exactly the same as the book because there should only be 1 correct answer.