EE241 Spring 2015: Tutorial #1

Friday, Jan. 16, 2015

Problem 1 (Rapidfire review): Fill in the right-hand column below given that A is a $m \times n$ matrix, B is a $k \times l$ matrix, and $\vec{b}$ is a r-vector.

You add 3 new constraints to the linear system described by A, how many constraints are there now?	$m + 3$
You add 3 new constraints to the linear system described by A, how many variables are there now?	n
Let $r = m$, $k = l$, and B be a scalar matrix. If $\vec{x}$ is the solution to $A\vec{x} = \vec{b}$, what is the solution to $(BA)\vec{x} = B\vec{b}$?	Still just $\vec{x}$

Mark true or false below

$A = A^T$ always	False
$AB = BA$ always	False
$A(BC) = (AB)C$ always	True
$B^n B = BB^n$ always	True
$BAB = AB^2$ always	False

Problem 2: Are the following two linear systems equivalent? Hint: use the allowed manipulations of equations to transform one system into the other

$$\begin{cases} (1.1) & 2x + 4y + 7z + 8w = 2 \\ (1.2) & 2y + 4w = 0 \end{cases} \qquad \begin{cases} (2.1) & x + 3.5z = 1 \\ (2.2) & 2y + 4w = 0 \end{cases}$$

Solution. Note the following

$$(1.1) = 2 \cdot (2.1) + 2 \cdot (2.2) \qquad \text{and} \qquad (1.2) = (2.2)$$

Thus, **yes** they are equivalent. □

Problem 3: Let A be a 1000×2 matrix and let B be a 1000×3 matrix. Given the dot products below, write down C when $C = A^T B$.

$$\begin{array}{lll} \text{col}_1(A) \cdot \text{col}_1(B) = 4 & \text{col}_1(A) \cdot \text{col}_2(B) = 7 & \text{col}_1(A) \cdot \text{col}_3(B) = -1 \\ \text{col}_2(A) \cdot \text{col}_1(B) = 4 & \text{col}_2(A) \cdot \text{col}_2(B) = 0 & \text{col}_2(A) \cdot \text{col}_3(B) = 2 \end{array}$$

Solution. First we note that since A is 1000×2, then A^T is 2×1000 and so $A^T B$ is well-defined. Furthermore, $C = A^T B$ is of size 2×3. □

Problem 4: Show that for an $n \times n$ scalar matrix A and any $n \times n$ square matrix B

$$AB = BA$$

Solution. Recall that in the resulting matrix multiplication, each element can be found via

$$(AB)_{ij} = \text{row}_i(A) \cdot \text{col}_j(B)$$

However, since A is scalar, $\text{row}_i(A)$ has a specific form, i.e.:

$$\text{row}_i(A) = \left[0\ ,\ 0\ ,\ \ldots\ ,\ \underset{i^{\text{th}}\text{ position}}{a}\ ,\ \ldots\ ,\ 0\right]$$

Thus, the dot product of interest becomes

$$\begin{aligned}(AB)_{ij} &= \sum_{k=1}^{n} (\text{row}_i(A))_k\, (\text{col}_j(B))_k \\ &= \sum_{k=1}^{n} a_{ik} b_{kj} \\ &= 0 \times b_{0j} + 0 \times b_{1j} + \cdots + a \times b_{ij} + \cdots + 0 \times b_{nj} \\ &= a \times b_{ij}\end{aligned}$$

Since we have that $(AB)_{ij} = a(B)_{ij}$, we know that $AB = aB$. We can perform this calculation again to find that $BA = aB$ and so $AB = BA$ for this special case. □

PROBLEM 5 (The matrix oracle): Consider the following two oracles for some unknown (but constant) $n \times n$ square matrix A

	Input	Output
Oracle 1	$\vec{x}$	$A\vec{x}$
Oracle 2	$\vec{x}, \vec{y}$	$\vec{y}^T A\vec{x}$

(a) What are the dimensions of $\vec{x}$ and $\vec{y}$?
(b) How can we use Oracle 1 to retrieve the 3rd row of A? How can we use Oracle 2 to retrieve the $(3, 4)$ entry of A?
(c) How can we use Oracle 2 to retrieve all of the diagonal elements of A?
(d) Assuming all elements of A are non-negative, how can we determine if A is diagonal using only n calls to Oracle 2?

Solution.

(a) Since $\vec{x}$ must match the n columns of A, it is an n-vector. Since $\vec{y}^T$ must match the n rows of A, $\vec{y}$ is also an n-vector. (Note that both are column vectors).
(b) Consider sending the vector

$$\vec{x} = [0\ ,\ 0\ ,\ 1\ ,\ 0\ ,\ \ldots\ ,\ 0]\,.$$

Recall also the formula for the product of a matrix and vector given by

$$A\vec{x} = \sum_{i=1}^{n} \text{col}_i(A)\, x_i.$$

In our case, since only the third element is non-zero, we get that $A\vec{x} = \text{col}_3(A)$ as required.
To find the $(3, 4)$ element, consider using

$$\vec{x} = [0\ ,\ 0\ ,\ 0\ ,\ 1\ ,\ 0\ ,\ \ldots\ ,\ 0]\,.$$

The result of Oracle 2 can now be written as

$$\vec{y}^T A\vec{x} = \vec{y}^T (A\vec{x}) = \vec{y}^T \text{col}_4(A)\,.$$

Note that the above expression is the product of a $1 \times n$ matrix and a $n \times 1$ matrix (or vector). This is precisely the definition of dot product and we can write

$$\vec{y}^T (A\vec{x}) = \sum_{i=1}^{n} y_i a_{i4}.$$

It remains only to use the the vector

$$\vec{y} = [0 \ , \ 0 \ , \ 1 \ , \ 0 \ , \ \dots \ , \ 0]$$

To find that $\vec{y}^T A\vec{x} = a_{34}$.

(c) Note that, in general, the formula for the output of Oracle 2 is

$$\vec{y}^T A\vec{x} = \sum_{ij} y_i a_{ij} x_j.$$

Thus, by choosing vectors $\vec{x}$ and $\vec{y}$ that contain all 0's and one 1 in a specific position, we can select out any element of the matrix A. In particular, we can use

$$\vec{x} = \vec{y} = \left[0 \ , \ \dots \ , \ \underset{i^{\text{th}} \text{ position}}{1} \ , \ \dots \ , \ 0\right]$$

to get

$$\vec{y}^T A\vec{x} = a_{ii}.$$

For any i.

(d) In order to check that A is diagonal, it suffices to check that all its *off-diagonal* elements are 0. Off-diagonal elements are those a_{ij} where $i \neq j$. First, let

$$\vec{x} \left[0 \ , \ \dots \ , \ \underset{i^{\text{th}} \text{ position}}{1} \ , \ \dots \ , \ 0\right].$$

Thus, $A\vec{x} = \text{col}_i (A)$. Now that we have the i^{th} column, we can use the following trick:

$$\boxed{\text{If } \ \sum_{j \neq i} a_{ij} = 0 \ \text{ and } \ a_{ij} \geq 0 \ \forall \ i, j \ \text{ then } \ a_{ij} = 0 \ \forall \ i \neq j}$$

So, how do we get the sum in the statement above using the vector $\vec{y}$? Simply choose

$$\vec{y} \left[1 \ , \ \dots \ , \ \underset{i^{\text{th}} \text{ position}}{0} \ , \ \dots \ , \ 1\right].$$

Then, $\vec{y}^T A\vec{x} = \sum_{i \neq j} a_{ij}$. If, for every value of i, this sum is 0, then we know that A is diagonal. In other words, we need only to call on Oracle 2 n times to fully check that A is diagonal.

$\square$