Lecture #4: Matrix Multiplication & Inverses

Today we will talk about how to multiply matrices and what it means.

There are a few ways to think about it.

**View #1: As a formula**

\[
C_{ik} = \sum_{j=1}^{n} A_{ij} B_{jk}
\]

It's hard to see why this definition is natural, but an example helps.

**Application: Counting Walks**

we'll be working with graphs - has vertices and edges, e.g.
This is a natural abstraction for doing things like describing who is friends with who on face book (vertizes = people, edges = friendships)

**def:** A walk is a sequence of vertizes connected by edges, where repetitions are o.k.

e.g. a – b – a – b – d is a walk of length 4

**Goal:** Count the number of walks of a given length

It turns out you can do it through matrix multiplication

Q: But how do we represent a graph as a matrix?

**def:** An adjacency matrix is a matrix where each row/column represents a vertex, and there is a 1 in row i, column j if there is an edge between the corresponding vertices
e.g. \[
\begin{bmatrix}
a & 0 & 1 & 0 & 0 \\
b & 1 & 0 & 1 & 1 \\
c & 0 & 1 & 0 & 1 \\
d & 0 & 1 & 1 & 0 \\
a & b & c & d
\end{bmatrix}
\]

what happens if we multiply \( A \) by itself?

\[(a \begin{bmatrix} a & b \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} c)
\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} = a \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 3 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{bmatrix} \]

\[0 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 1 = 1\]

\[
a \times a - c \quad a - b - c \quad a - c \times \quad a \times a - c
\]

Each term is a walk of length two.

And the adjacency matrix tells us if edges are there.

Q2: What does \((A^2)_{a,b} = 0\) tell us?

There are no walks of length two that start at \(a\) and end at \(b\).

More generally \(A \times A \times \ldots \times A\), \(l\) times, counts length \(l\) walks. Row = starting vertex, column = ending vertex.
Another way to think about matrix multiplication:

**def:** The *inner-product* of two vectors $x$ and $y$ with the same dimension $n$

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i$$

**View #2:** In terms of inner products

$$A \in \mathbb{R}^{m \times n}, \quad B \in \mathbb{R}^{n \times p} \quad \Rightarrow \quad C = \mathbf{AB} \in \mathbb{R}^{m \times p}$$

$C_{ik} =$ the inner product b/t row $i$th row in $A$ and column $k$th column in $B$

**Let's do another application**

**Application: Linear Dynamical Systems**

In science and engineering we often want to understand how a (linear) system evolves

$$\mathbf{X}_{t+1} = \mathbf{A} \mathbf{X}_t$$

- $\mathbf{A}$ is a matrix representing how the system updates
- $\mathbf{X}_t$ is a length $n$ vector representing the state of the system at time $t$
e.g. the Tacoma bridge example from earlier

Today: the **predator-prey model**

\[
\begin{align*}
g(t) &= \# \text{ frogs at time } t \\
y(t) &= \# \text{ flies at time } t \\
\end{align*}
\]

\[
\begin{align*}
g(t+1) &= 0.4 \, g(t) + 0.2 \, y(t) \\
y(t+1) &= -0.6 \, g(t) + 1.8 \, y(t) \\
\end{align*}
\]

Q3: Can we model this using matrices?

\[
\begin{bmatrix}
g(t+1) \\
y(t+1)
\end{bmatrix} =
\begin{bmatrix}
0.4 & 0.2 \\
-0.6 & 1.8
\end{bmatrix}
\begin{bmatrix}
g(t) \\
y(t)
\end{bmatrix}
\]

Q4: If we start at \( g(0), y(0) \) what does the population look like at time \( n \)?

**Fact:** matrix multiplication is **associative**

\[
A \, (BC) = (AB)C
\]

**Note:** dimensions need to match, but it's the same constraint on both sides.

What this means for us is we can rewrite

\[
A \times (A \times (A \times \ldots \times (A \times \begin{bmatrix} g(0) \\ y(0) \end{bmatrix}) \ldots ))
\]

\( n \) times
Instead as \(( ((A) \times A) \times A) \ldots \times A \) \(n\) times

We'll call this \(A^n\)

**Takeaway:** So if we want to simulate the model for many different initial conditions, we can just compute \(A^n\) once.

**Some other key facts**

**Fact:** Matrix multiplication is **distributive**

\[ A(B+C) = AB + AC \]
\[ (A+B)C = AC + BC \]

Q5: Why did I write this rule out two different ways?

**Anti-Fact:** Matrix multiplication is generally not **commutative** \(AB \neq BA\)

In fact, the dimensions do not necessarily make sense

\([m \times n] \times [n \times p] \) vs. \([n \times p] \times [m \times n] \)
Also, there is a "unit" element
acts like 1

**Definition:** The identity matrix is
\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1
\end{bmatrix} = I_n
\]

**Fact:** \( I_A = A \) \(_{n \times n}^{n \times n} \); \( A I = A \) \(_{m \times m}^{m \times m} \)

Now we are ready for a GREAT idea:

When we want to solve the linear system \( A x = b \), wouldn't it be easier if there was a matrix \( A^{-1} \) so that \( A^{-1} A = I \)?

First, why would this help?

\[
A x = b \implies A^{-1} (A x) = A^{-1} b \\
\implies (A^{-1} A) x = A^{-1} b \implies x = A^{-1} b
\]

\[\text{Q6: what rule did I use here?}\]

Even better: I claim we already know how to find \( A^{-1} \)
Recall in Gauss-Jordan elimination the first step when solving

\[
\begin{align*}
\begin{bmatrix}
1 & -1 & 2 \\
-2 & 2 & -3 \\
-3 & -1 & 2
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z
\end{bmatrix}
&=
\begin{bmatrix}
1 \\
-1 \\
-3
\end{bmatrix}
\end{align*}
\]

was to add 2\(r_1\) to \(r_2\).

Q7: How can I describe this operation using matrix multiplication? 1×3 vector

\[
\begin{bmatrix}
1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3
\end{bmatrix}
\]

\[
\begin{bmatrix}
2 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3
\end{bmatrix}
\]

In Gauss-Jordan elimination, sometimes we need to swap rows, which can be done by

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 0 & 1 \\
0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
r_1 \\
r_2 \\
r_3
\end{bmatrix}
\]

Each step in Gauss-Jordan elimination can be implemented as a matrix mult:

\[
A \rightarrow B_1A \rightarrow B_2B_1A \rightarrow \cdots \rightarrow (B_p \cdots B_1)A = I
\]

\[
A^{-1}
\]

But we don't always get so lucky.
Sometimes we get a row of zeros, e.g.

\[
A = \begin{bmatrix}
2 & -3 & 0 \\
1 & 1 & 2 \\
3 & -2 & 2
\end{bmatrix} \quad \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} = \begin{bmatrix}
1 \\
1 \\
3
\end{bmatrix}
\]

\[\sim\]

\[
\begin{bmatrix}
2 & -3 & 0 \\
0 & \frac{5}{2} & 2 \\
0 & 0 & 0
\end{bmatrix} \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} = \begin{bmatrix}
1 \\
\frac{1}{2} \\
1
\end{bmatrix}
\]

**Q8:** What does this tell us about \( A^{-1} \)?

(a) It doesn't exist

(b) For this choice of \( b \) it doesn't exist, but for others it might

(c) You can't trust matrices

Note: We introduced \( A^{-1} \) as the matrix s.t.

\[A^{-1}A = I \quad \text{(left inverse)}\]

If \( A \) is square, there could also be a matrix \( B \) s.t.

\[AB = I \quad \text{(right inverse)}\]

**Q9:** Could these be different?
Fact: If A is square then its right and left inverse are the same.

Q10: Can you come up with a non-square A that has a left but no right inverse?

One last view: As matrix vector products can also think about

\[
\begin{bmatrix}
A \\
\end{bmatrix}^n \begin{bmatrix}
B \\
B_p \\
\end{bmatrix}
\]

as \( p \) matrix vector products

Let \( B_1, B_2, \ldots, B_p \) be the columns of \( B \)

Then \( AB = [AB_1, AB_2, \ldots, AB_p] \)