BE – Discussion #2, 2020-09-11

Nice to see you again!

- Devin Goodwin
  devinbg@bu.edu

- Leah Gaeta
  ltgaeta@bu.edu

*please unmute, show video, and fully participate

Announcements

- make sure you have all three exam and all quizzes listed in the syllabus on your calendar
  → first exam is 3 weeks from today!

- if you have questions, please check to see if your question has been answered on the discussion forum, and if it hasn't then post it! (Do not email Prof. Attaway)

- TA Open Hours:
  → Sunday 4-10 pm
  → Tuesday 6-10 pm
  → Thursday 4-10 pm {Devin's OH 6-8 pm Thursday}

- Practice quiz this afternoon ... please take like a real quiz!!!
  → it will release on Gradescope at 4:40 pm, due by 4:55 pm
  → quizzes will only be accepted via gradescope, not email
  → download quiz, fill it out, scan it, upload as a PDF.
    - .pdf must be correct number of pages
    - time yourself so that you're ready for week
  → will also be open for a couple days to get extra practice with uploading

- Classes next week:
  → in-person will be an option by invitation only
  → class will be conducted the same, so fully via zoom with breakout rooms like we have been doing
  → you can only come to the room if you received an invite

Review of Material

- vectors and matrices
- matrix operations and functions
- logical indexing vs. using find()
- anything else?

Vectors + Matrices

`1:0.5:3` → 1 1.5 2 2.5 3

`linspace(1,3,5)` → 1 1.5 2 2.5 3

↑ # of elements

spacing for linspace formula:
$(x_2 - x_1)/(n-1)$

`linspace(x1, x2, n)`

ex. `linspace(9,5,3)`

↳ $(5-9)/(3-1) \rightarrow (-4)/(2) \rightarrow -2$

⇒ 9 7 5

Matrix (Array) Multiplication

$$A = \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix} \quad 2 \times 2$$

$$B = \begin{bmatrix} 2 & 1 & 3 \\ 1 & 5 & 6 \\ 3 & 6 & 0 \end{bmatrix} \quad 3 \times 3$$

$$C = \begin{bmatrix} 3 & 2 & 5 \\ 4 & 1 & 2 \end{bmatrix} \quad 2 \times 3$$

ex. `3 * A`

3 12
9 6

ex. `A * B`

2×2 and 3×3

↑ dimensions don't match

INVALID

ex. `A * C`

$$\begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix} \quad \begin{bmatrix} 3 & 2 & 5 \\ 4 & 1 & 2 \end{bmatrix}$$

2×2 and 2×3

↑ dimensions match!

$$\begin{bmatrix} 1\cdot3 + 4\cdot4 & 1\cdot2 + 4\cdot1 & 1\cdot5 + 4\cdot2 \\ 3\cdot3 + 2\cdot4 & 3\cdot2 + 2\cdot1 & 3\cdot5 + 2\cdot2 \end{bmatrix}$$

$$\begin{bmatrix} 19 & 6 & 13 \\ 17 & 8 & 19 \end{bmatrix}$$ ← final answer, VALID

↑ 2 × 3 matrix (outer dimensions)

$$D = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix} \quad 2\times2 \qquad E = \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \quad 2\times2$$

$2\cdot0 + 4\cdot2 \qquad 2\cdot1 + 4\cdot3$

$6\cdot0 + 8\cdot2 \qquad 6\cdot1 + 8\cdot3$

D * E ... VALID, you'll get a 2x2 matrix $\Rightarrow \begin{bmatrix} 8 & 14 \\ 16 & 30 \end{bmatrix}$

D .* E ... ALSO VALID

$$\begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & 4 \\ 12 & 24 \end{bmatrix}$$

element-by-element operation

Changing Elements in an Array

```
>> v = 1:5
>> v =
     1   2   3   4   5
>> v(2:5) = 7:9
>> v =
     1   7   8   9   5
```

```
>> mat = [2:4, 5:7]
>> mat =
     2   3   4
     5   6   7
>> mat(1, :)
>> mat =
     1   1   1
     5   6   7
```

Rand() vs. Randi()

rand()*5 + 5 will generate random real numbers from 5 to 10

- rand() — generate # from 0 to 1
- * 5 — multiply that # by 5
- + 5 — will then add 5 to #*5

isequal vs. ==

```
>> vec1 = 1:3;        =>   1   2   3
>> vec2 = [1 0 3];    =>   1   0   3

>> isequal(vec1, vec2)
>> ans =
     0
```

← single logical value ... these aren't identical vectors

```
>> vec1 == vec2
>> ans =
     1   0   1
```

← element-by-element logical values

Logical Indexing vs. Using find()

```
>> vect = randi([-5, 10], 1, 6)
>> vect =
     -5  8  9  5  7  6
>> vect (vect > 0)
>> ans =
      8  9  5  7  6      ← gives the actual elements > 0
>> find(vect > 0)
>> ans =
      2  3  4  5  6      ← find() returns the indices
```

---

vecA =
2
4
6
8
(4 x 1)

vecb =
3 6 9 12
(1 x 4)

EXTRA STUFF

vecb * vecA

$$\begin{bmatrix} 3 & 6 & 9 & 12 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \\ 6 \\ 8 \end{bmatrix}$$

vecA .* vecb NOT VALID

vecA' .* vecb VALID

vecA .* 3 is valid
vecA * 3 is also valid
(MATLAB knows it's scalar mult.)