3. LINEAR ALGEBRA REVIEW

-- OCTAVE IMPLEMENTATION

R – refers to set of **scalar** real numbers.

R^n – refers to set of **n-dimensional vectors** of real numbers.

$$\begin{bmatrix} w \\ x \\ y \\ z \end{bmatrix}$$

MATLAB/OCTAVE COMMANDS TO CREATE MATRICES AND VECTORS:

```
% The ; denotes we are going back to a new row.
A = [1, 2, 3; 4, 5, 6; 7, 8, 9; 10, 11, 12]

% Initialize a vector
v = [1;2;3]

% Get the dimension of the matrix A where m = rows and n = columns
[m,n] = size(A)

% You could also store it this way
dim_A = size(A)

% Get the dimension of the vector v
dim_v = size(v)

% Now let's index into the 2nd row 3rd column of matrix A
A_23 = A(2,3)

```

Operations on matrices:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} w & x \\ y & z \end{bmatrix} = \begin{bmatrix} a+w & b+x \\ c+y & d+z \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} - \begin{bmatrix} w & x \\ y & z \end{bmatrix} = \begin{bmatrix} a-w & b-x \\ c-y & d-z \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} * x = \begin{bmatrix} a * x & b * x \\ c * x & d * x \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} / x = \begin{bmatrix} a/x & b/x \\ c/x & d/x \end{bmatrix}$$

⇨ **MATLAB/OCTAVE COMMANDS**:

```
% Initialize matrix A and B
A = [1, 2, 4; 5, 3, 2]
B = [1, 3, 4; 1, 1, 1]

% Initialize constant s
s = 2

% See how element-wise addition works
add_AB = A + B

% See how element-wise subtraction works
sub_AB = A - B

% See how scalar multiplication works
mult_As = A * s

% Divide A by s
div_As = A / s

% What happens if we have a Matrix + scalar?
add_As = A + s
```

MULTIPLICATION OF MATRICES AND VECTORS:

House sizes:

- 2104
- 1416
- 1534
- 852

$$h_\theta(x) = -40 + 0.25x$$

$h_\theta(x)$

Matrix (4x2) × Vector (2x1) = 4x1 matrix

$$\begin{bmatrix}1 & 2104\\ 1 & 1416\\ 1 & 1534\\ 1 & 852\end{bmatrix} \times \begin{bmatrix}-40\\ 0.25\end{bmatrix} = \begin{bmatrix}-40\times 1 + 0.25\times 2104\\ -40\times 1 + 0.25\times 1416\\ \vdots \\ \vdots\end{bmatrix}$$

$h_\theta(2104)$, $h_\theta(1416)$

$\boxed{\text{prediction}}$ = DataMatrix * parameters.

4x1

for i = 1:1000,
prediction (i) = ...

➩ MATLAB CODE:

```
% Initialize matrix A
A = [1, 2, 3; 4, 5, 6;7, 8, 9]

% Initialize vector v
v = [1; 1; 1]

% Multiply A * v
Av = A * v
```

MATRIX TO MATRIX MULIPLICATION:

House sizes:

$$\begin{Bmatrix}2104\\ 1416\\ 1534\\ 852\end{Bmatrix}$$

Have 3 competing hypotheses:

1. $h_\theta(x) = -40 + 0.25x$
2. $h_\theta(x) = 200 + 0.1x$
3. $h_\theta(x) = -150 + 0.4x$

Matrix × Matrix

$$\begin{bmatrix}1 & 2104\\ 1 & 1416\\ 1 & 1534\\ 1 & 852\end{bmatrix} \times \begin{bmatrix}-40 & 200 & -150\\ 0.25 & 0.1 & 0.4\end{bmatrix} = \begin{bmatrix}486 & 410 & 692\\ 314 & 342 & 416\\ 344 & 353 & 464\\ 173 & 285 & 191\end{bmatrix}$$

Prediction of first h_θ

Predictions of 2nd h_θ

⇨ **MATLAB CODE**:

```
% Initialize a 3 by 2 matrix
A = [1, 2; 3, 4;5, 6]

% Initialize a 2 by 1 matrix
B = [1; 2]

% We expect a resulting matrix of (3 by 2)*(2 by 1) = (3 by 1)
mult_AB = A*B

% Make sure you understand why we got that result
```

Properties of matrix multiplication:

- Matrices are not commutative: $A * B \neq B * A$
- Matrices are associative: $(A * B) * C = A * (B * C)$

Identity Matrix

I is identity

Denoted $\underline{I}$ (or $\underline{I_{n\times n}}$).

Examples of identity matrices:

$$\underbrace{\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}}_{2 \times 2} \qquad \underbrace{\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}}_{3 \times 3} \qquad \underbrace{\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}}_{4 \times 4}$$

For any matrix A,

$$\underset{m\times n}{A}\cdot\underset{n\times n}{I}=\underset{m\times m}{I}\cdot\underset{m\times n}{A}=A$$

⇨ **MATLAB CODE**:

```
% Initialize random matrices A and B
A = [1,2;4,5]
B = [1,1;0,2]

% Initialize a 2 by 2 identity matrix
I = eye(2)

% The above notation is the same as I = [1,0;0,1]

% What happens when we multiply I*A ?
IA = I*A

% How about A*I ?
AI = A*I

% Compute A*B
AB = A*B

% Is it equal to B*A?
BA = B*A

% Note that IA = AI but AB != BA
```

INVERSE OF MATRIX

Matrix inverse:

square matrix (#rows = #columns) $\quad A^{-1} \quad A = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

If A is an m x m matrix, and if it has an inverse,

$$\rightarrow \quad A(A^{-1}) = A^{-1}A = I.$$

E.g.

$$\underbrace{\begin{bmatrix} 3 & 4 \\ 2 & 16 \end{bmatrix}}_{\substack{2\times 2 \\ A}} \underbrace{\begin{bmatrix} 0.4 & -0.1 \\ -0.05 & 0.075 \end{bmatrix}}_{A^{-1}} = \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{A^{-1}A} = I_{2\times 2}$$

Matrices that don't have an inverse are "singular" or "degenerate"

Andrew Ng

MATLAB CODE:

```
% Initialize matrix A
A = [1,2,0;0,5,6;7,0,9]

% Transpose A
A_trans = A'

% Take the inverse of A
A_inv = inv(A)

% What is A^(-1)*A?
A_invA = inv(A)*A
```