Section 1

The Basics:
Linear Algebra, Probability, Python, Recurrences

Linear Algebra

Basic Properties

Matrix rules				
scalar multiplication	$n\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}$		$=$	$\begin{bmatrix} na & nb & nc \\ nd & ne & nf \end{bmatrix}$
matrix addition	$\begin{bmatrix} a & b \\ c & d \\ e & f \end{bmatrix}$	$+ \begin{bmatrix} g & h \\ i & j \\ k & l \end{bmatrix}$	$=$	$\begin{bmatrix} a+g & b+h \\ c+i & d+j \\ e+k & f+l \end{bmatrix}$
matrix multiplication	$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}$	$\begin{bmatrix} g & h \\ i & j \\ k & l \end{bmatrix}$	$=$	$\begin{bmatrix} ag+bi+ck & ah+bj+cl \\ dg+ei+fk & dh+ej+fl \end{bmatrix}$

Basic Properties

$$\boldsymbol{v}^2 = ||\boldsymbol{v}||_2^2 = \boldsymbol{v}^T \boldsymbol{v}$$

$$(\boldsymbol{A} + \boldsymbol{B})^T = \boldsymbol{A}^T + \boldsymbol{B}^T$$

$$(\boldsymbol{A}\boldsymbol{B})^T = \boldsymbol{B}^T \boldsymbol{A}^T$$

Matrix Multiplication

Associative? : $(AB)C=A(BC)$

Distributive? : $A(B+C) = AB+AC$

Commutative? : $AB \neq BA$

Matrix Multiplication

Associative? : $(AB)C=A(BC)$

Distributive? : $A(B+C) = AB+AC$

~~**Commutative? : $AB \neq BA$**~~

Matrix Multiplication

$$C = AB = A \begin{bmatrix} | & | & & | \\ b_1 & b_2 & \cdots & b_p \\ | & | & & | \end{bmatrix} = \begin{bmatrix} | & | & & | \\ Ab_1 & Ab_2 & \cdots & Ab_p \\ | & | & & | \end{bmatrix}$$

$$C = AB = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} — & b_1^T & — \\ — & b_2^T & — \\ & \vdots & \\ — & b_n^T & — \end{bmatrix} = \sum_{i=1}^{n} a_i b_i^T \; .$$

Tip for HW

How can I do efficient dot-product multiplications?

$$a, b \in \mathbf{R}^d$$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} a_i * b_j$$

Matrix Calculus

$$f(\mathbf{w}) = (\mathbf{a} \cdot \mathbf{w} + 1)^2 + b\,\|\mathbf{w}\|_2^2 + \mathbf{w}^\top C \mathbf{w}$$

Compute $\nabla_{\mathbf{w}} f(\mathbf{w})$

$$\nabla_{\mathbf{w}} \mathbf{a} \cdot \mathbf{w} = \mathbf{a}$$

$$\nabla_{\mathbf{w}} \|\mathbf{w}\|_2^2 = \nabla_{\mathbf{w}} \mathbf{w} \cdot \mathbf{w} = 2\mathbf{w}$$

$$\nabla_{\mathbf{w}} \mathbf{w}^\top C \mathbf{w} = (C + C^\top)\mathbf{w}$$

Probability

Random Variables

- Discrete: $\mathbb{P}(X = a)$ OR $p_X(a)$
- Example: Rolling a dice. Outcomes {1, 2, 3, 4, 5, 6}

- Continuous: $\mathbb{P}(X \leq a) = \int_{-\infty}^{a} f_X(u)du$
- Example: Uniform random variable in [0, 1]

Conditional Probability

- What is the probability that event A occurs given that event B has occurred.
- Denoted $\mathbb{P}(A|B)$

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}$$

Example

	A = 0	**A = 1**	**A = 2**	**A = 3**
B = 0	0.1	0.25	0.1	0.05
B = 1	0.15	0	0.15	0.2

- What is $\mathbb{P}(A = 2)$
- What is $\mathbb{P}(A = 2 \mid B = 1)$

Independence

- A random variable $\boldsymbol{X}$ (event A) is independent of a random variable $\boldsymbol{Y}$ (event B) if the realization of $\boldsymbol{Y}$ (or B) does not affect the probability distribution of $\boldsymbol{X}$ (or A).
- Example: Suppose we toss a coin and roll a die. What is the probability that 5 appears on the die given that heads appeared on the coin?

Expectation

$$\mathbb{E}[A] = \sum_a a\,\mathbb{P}[A = a]$$

$$\mathbb{E}[A] = \int a f_A(a)\,da$$

Example

	$\mathbf{A = 0}$	$\mathbf{A = 1}$	$\mathbf{A = 2}$	$\mathbf{A = 3}$
$\mathbf{B = 0}$	0.1	0.25	0.1	0.05
$\mathbf{B = 1}$	0.15	0	0.15	0.2

- Are A and B independent?
- What are $\mathbb{E}[A]$, $\mathbb{E}[B]$, $\mathbb{E}[A+B]$

Example

	A = 0	**A = 1**	**A = 2**	**A = 3**
B = 0	0.1	0.25	0.1	0.05
B = 1	0.15	0	0.15	0.2

- Are A and B independent?
- What are $\mathbb{E}[A]$, $\mathbb{E}[B]$, $\mathbb{E}[A + B]$

Linearity of Expectation: $\mathbb{E}[A + B] = \mathbb{E}[A] + \mathbb{E}[B]$

True even when A and B are dependent!

Python Tips

Recurrences

Leveraging recursion

- Overlapping subproblems
- Optimal substructure
- Convert the given problem into a smaller (easier) one.

Example: Edit distance (In more detail)

- Question we are trying to answer is: What is the minimum number of edits do we need to make to transform word **a** into word **b?**
- (Also known as Levenshtein distance)