

# Arithmetic Coding:

## Lossless data compression Algorithm

### Alternative of Huffman

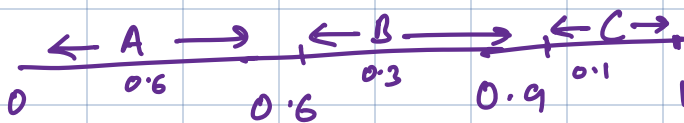
- Gives more optimum result than Huffman
- Execution time is  $\uparrow$
- Complex than Huffman
- Compression Ratio is more

→ Encodes entire message as a single number between 0 and 1

Example: string CAB

Char	A	B	C
Prob	0.6	0.3	0.1

① Divide range b/w 0-1



② First char to be encoded C A B

C range  $\Rightarrow$   $\boxed{0.9 - 1}$   
range = 0.1

code would start in this range

0.1 is divided among symbols.

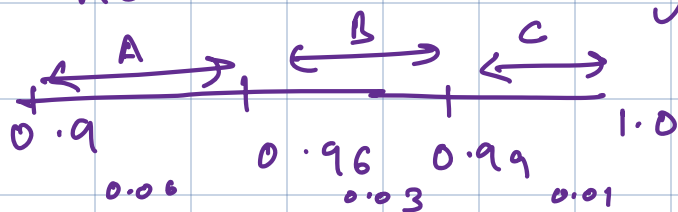
③ Cumulative Prob: acc. to prob. 0

$$A = 0.9 + 0.6 \times 0.1 = 0.96$$

$$B = 0.96 + 0.3 \times 0.1 = 0.99$$

$$C = 0.99 + 0.1 \times 0.1 = 1.0$$

④ Resultant new range is



⑤ encode A now

$$= 0.90 - 0.96 \quad \text{Range}$$

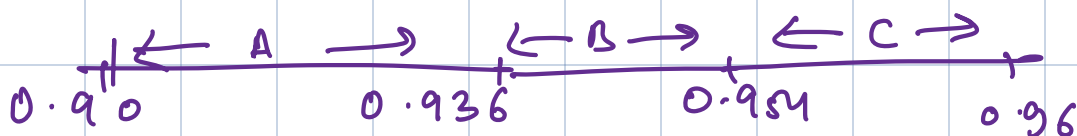
$$= 0.06 \quad \text{range}$$

com Prob

$$A = 0.90 + 0.6 \times 0.06 = 0.936$$

$$B = 0.936 + 0.3 \times 0.06 = 0.954$$

$$C = 0.954 + 0.1 \times 0.06 = 0.96$$



⑥ encode B now

$$0.936 - 0.954 = \text{Range}$$

String ends

Final code for string CAB  $\Rightarrow$  0.936 - 0.954 range.

$\therefore$  that's how it can be represented.

