

Exercise 1

$A = \{1, 5, 8, 9\}$      $B = \{2, 3, 8\}$      $C = \{2, 5, 8\}$

$P_1 = 4x + 2 \pmod{11}$

$A_1 = \{6, \underline{0}, 1, 5\}$      $B_1 = \{10, 3, \underline{1}\}$      $C_1 = \{10, \underline{0}, 1\}$

$P_2 = 2x \pmod{11}$

$A_2 = \{\underline{2}, 10, 5, 7\}$      $B_2 = \{\underline{4}, 6, 5\}$      $C_2 = \{\underline{4}, 10, 5\}$

$P_3 = 5x + 1 \pmod{11}$

$A_3 = \{6, 4, 8, \underline{2}\}$      $B_3 = \{\underline{0}, 5, 8\}$      $C_3 = \{\underline{0}, 4, 8\}$

thus the sketches are:

$A = \langle 0, 2, 2 \rangle$      $B = \langle 1, 4, 0 \rangle$      $C = \langle 0, 4, 0 \rangle$

Given the sketches of size 3 the estimates for the Jaccard

distance are:

$J(A, B) = \frac{\# \text{ matching components}}{3} = 0$

$J(A, C) = \frac{1}{3}$

$J(B, C) = \frac{2}{3}$

## Exercise 2

4 → 7, 8, 13, 14, 18, 19

5 → 7, 9, 11, 13, 14, 18

6 → 7, 8, 11, 13, 14, 19

We notice that 5's list can be compressed only looking at previous 4's list, which is the only previous one. Whereas for the 6's list we could use 4's or 5's list to make the compression, we choose the one that shares most items. This is 4's list.

4 → 7, 8, 13, 14, 18, 19

5 → 10, 111, 0, [9, 11]

↓  
1002

we can drop the last 0 by storing 6 = list length

6 → ~~10~~ 1111, 0, 1, [11]

↓  
1:30

we can drop the last 1 by storing 6 = list length

## Exercise 3

f<sub>old</sub> = AAABB

f<sub>new</sub> = AABAAAC

rsync

AAA, BB ↓  
client

→  
h<sub>1</sub>, h<sub>2</sub>

server  
AABAAAC  
h<sub>1</sub>

↓  
AAB h<sub>1</sub> C

zsync

server  
AABAAAC \$\$  
 $h_1$     $h_2$     $h_3$

$h_1, h_2, h_3$  →

client  
AAABB \$\$  
 $h_2$   
 $h_1$

overlapping is admitted  
 by the way zsync works  
 because blocks are not  
 overlapping in flow!!

← 1, 1, 0

AABAAA | C  $\xrightarrow{\text{strip}}$   $\langle 0, 0, c \rangle$

Exercise 4

$k=2, e=1$

- \$a → 1, 2, 3
- aa → 1
- ab → 1, 2
- bb → 1
- ba → 2
- ac → 3
- ea → 3

$D = \{ aabb, aba, acac \}$

Given the query string  $Q = aabc \rightarrow$

- \$a → 1, 2, 3
- aa → 1
- ab → 1, 2
- bc → 1

- String 1 = aabb has 3 matching bigrams
- String 2 = aba has 2 matching " "
- " 3 = acac " 1 " "

its bigrams  
 $\$a, aa, ab, bc$

The formula is  $L - e \cdot k = 4 - 2 = 2$  hence the candidates are "aabb" and "aba".

	$\phi$	A	A	B	B
$\phi$	0	1	2	3	4
A	1	0	1	2	3
A	2	1	0	1	2
B	3	2	1	0	1
C	4	3	2	1	<b>1</b>

	$\phi$	A	B	A
$\phi$	0	1	2	3
A	1	0	1	2
A	2	1	1	1
B	3	2	1	2
C	4	3	2	<b>3</b>

The string  $s_1 = AABBB$  is returned