Last topic on final exam

Data Structures and Algorithms

MinHash Sketch

CS 225
Brad Solomon

December 3, 2025

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN
Department of Computer Science

Announcements

Max 70 EC

This week's lab is OPTIONAL! (Worth 4 EC points)

MP_Mazes survey was only about 60% participation but I'll give EC points

This means you will get all the EC points so far (for forms)

Learning Objectives

Review the concept of cardinality and cardinality estimation

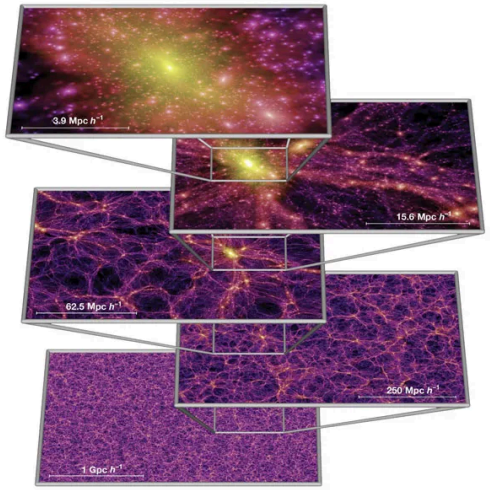
Improve our cardinality estimation approach

Demonstrate the relationship between cardinality and similarity

Introduce the MinHash Sketch for set similarity detection

Big Picture of Sketching

If you can't store or analyze a data collection using exact approaches…



Bloom Filter Sketch → "Find" (if item exists)

1) Hash every item one at a time
2) Store in a bloom filter

Cardinality Sketch → get estimate of # of unique items

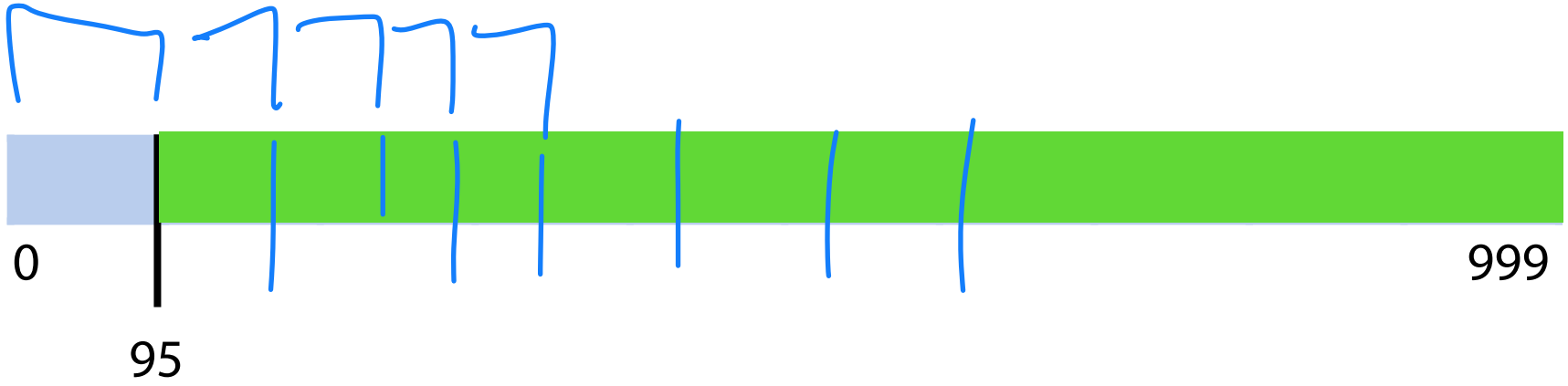
cardinality

1) Hash every item one at a time
2) Store the k-th minimum hash value

1 # (the Kth)

Cardinality Estimation

Let min = 95. Can we estimate N, the cardinality of the set?



Conceptually: If we scatter N points randomly across the interval, we end up with $N + 1$ partitions, each about $1000/(N + 1)$ long

Assuming our first 'partition' is about average:

$$95 \approx 1000/(N + 1)$$

$$N + 1 \approx 10.5$$

$$N \approx 9.5$$

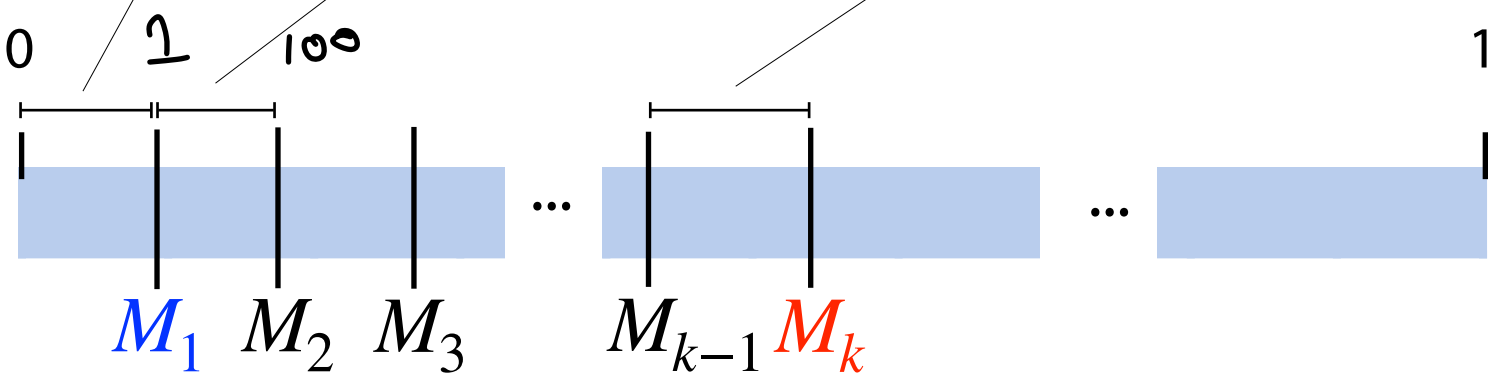
Cardinality Sketch

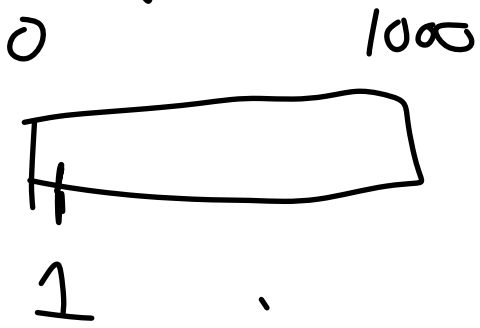
Average the Kth min value

$$\frac{1}{N+1} = \frac{\mathbf{E}[M_k]}{k}$$

$$= \Big[\underbrace{\mathbf{E}[M_1]} + \underbrace{(\mathbf{E}[M_2] - \mathbf{E}[M_1])} + \ldots + \underbrace{(\mathbf{E}[M_k] - \mathbf{E}[M_{k-1}])}\Big] \cdot \frac{1}{k}$$

This accounts for random variable





k^{th} minimum value (KMV)

Averages k estimates for $\frac{1}{N+1}$

Cardinality Sketch (Non-normalized)

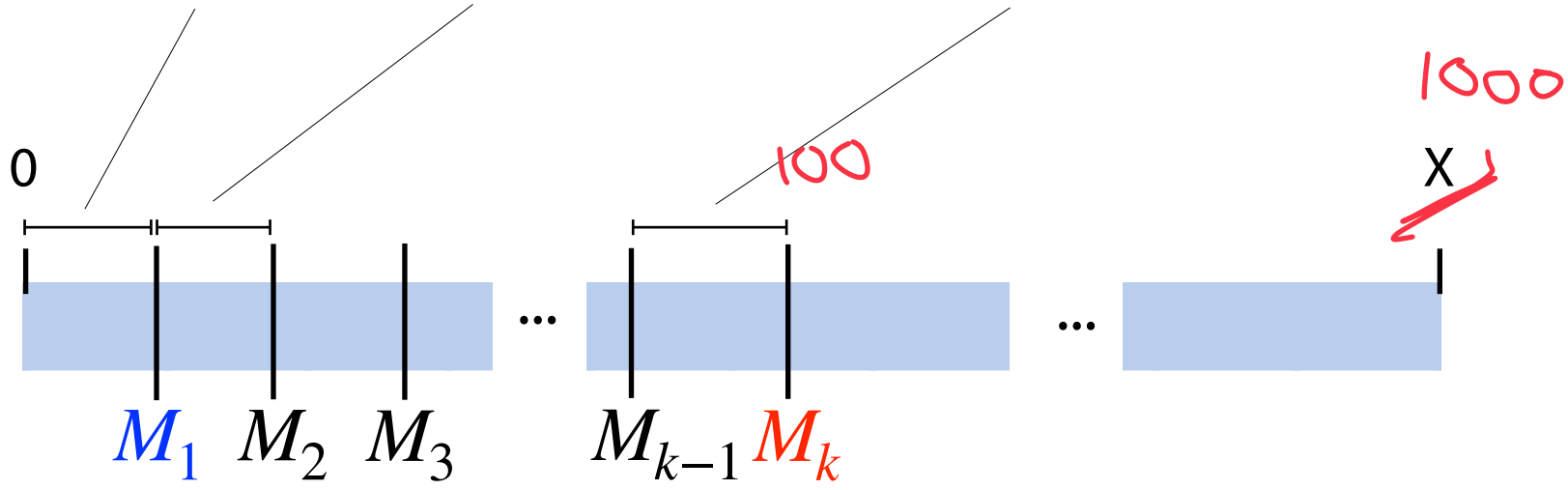
Ratio of universe size

$$\frac{1000\,X}{N+1} = \frac{\mathbf{E}[M_k]}{k}$$

100

$$\rightarrow \frac{1000}{100} \cdot k = N+1$$

$$= \left[\mathbf{E}[M_1] + (\mathbf{E}[M_2] - \mathbf{E}[M_1]) + \ldots + (\mathbf{E}[M_k] - \mathbf{E}[M_{k-1}])\right] \cdot \frac{1}{k}$$



k^{th} minimum value (KMV)

Averages k estimates for $\frac{1}{N+1}$

Cardinality Sketch

Given any dataset and a SUHA hash function, we can **estimate the number of unique items** by tracking the **k-th minimum hash value**.

Hash

0.253	0.839	0.327	0.655	0.491

Applied Cardinalities

Cardinalities

$$|A|$$

$$|B|$$

$$|A \cup B|$$

$$|A \cap B|$$

Set similarities

$$O = \frac{|A \cap B|}{min(|A|, |B|)}$$

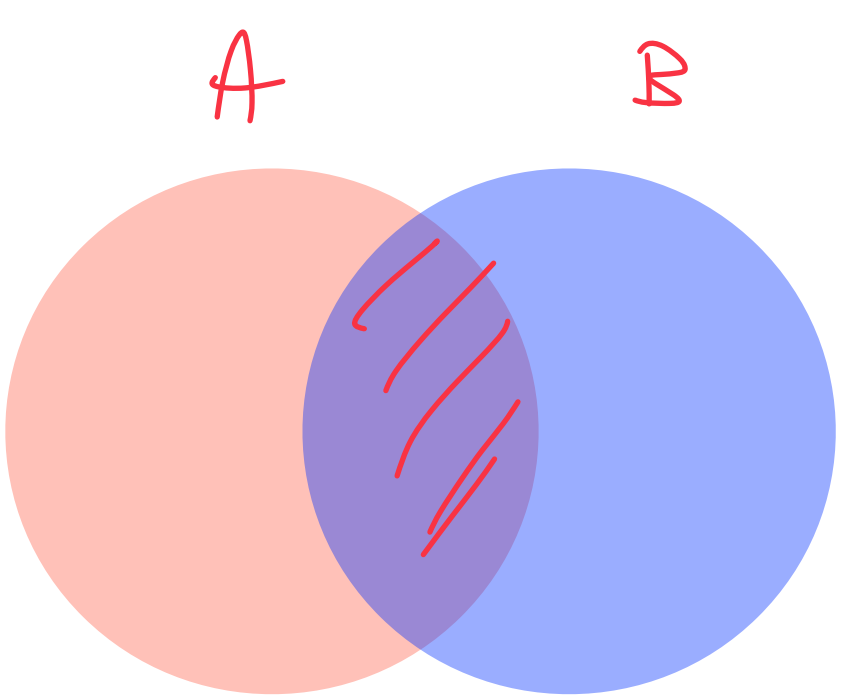
$$J = \frac{|A \cap B|}{|A \cup B|}$$

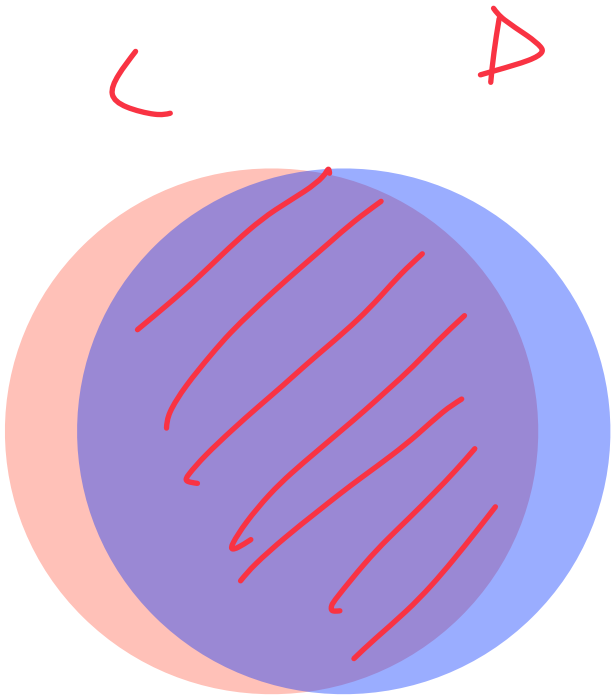
Real-world Meaning

```
AGGCCACAGTGTATTATGACTG
||||||||||||  |||||||||
AGGCCACAGTGAGTTATGACTG

AAAAAAAAAAAGATGT-AAGTA
|||||||||||||||||| |||||
AAAAAAAAAAAGATGTAAAGTA

GAGG--TCAGATTCACAGCCAC
||||  ||||||||||||||||||
GAGGGGTCAGATTCACAGCCAC
```

Set Similarity Review

How can we describe how ***similar*** two sets are?

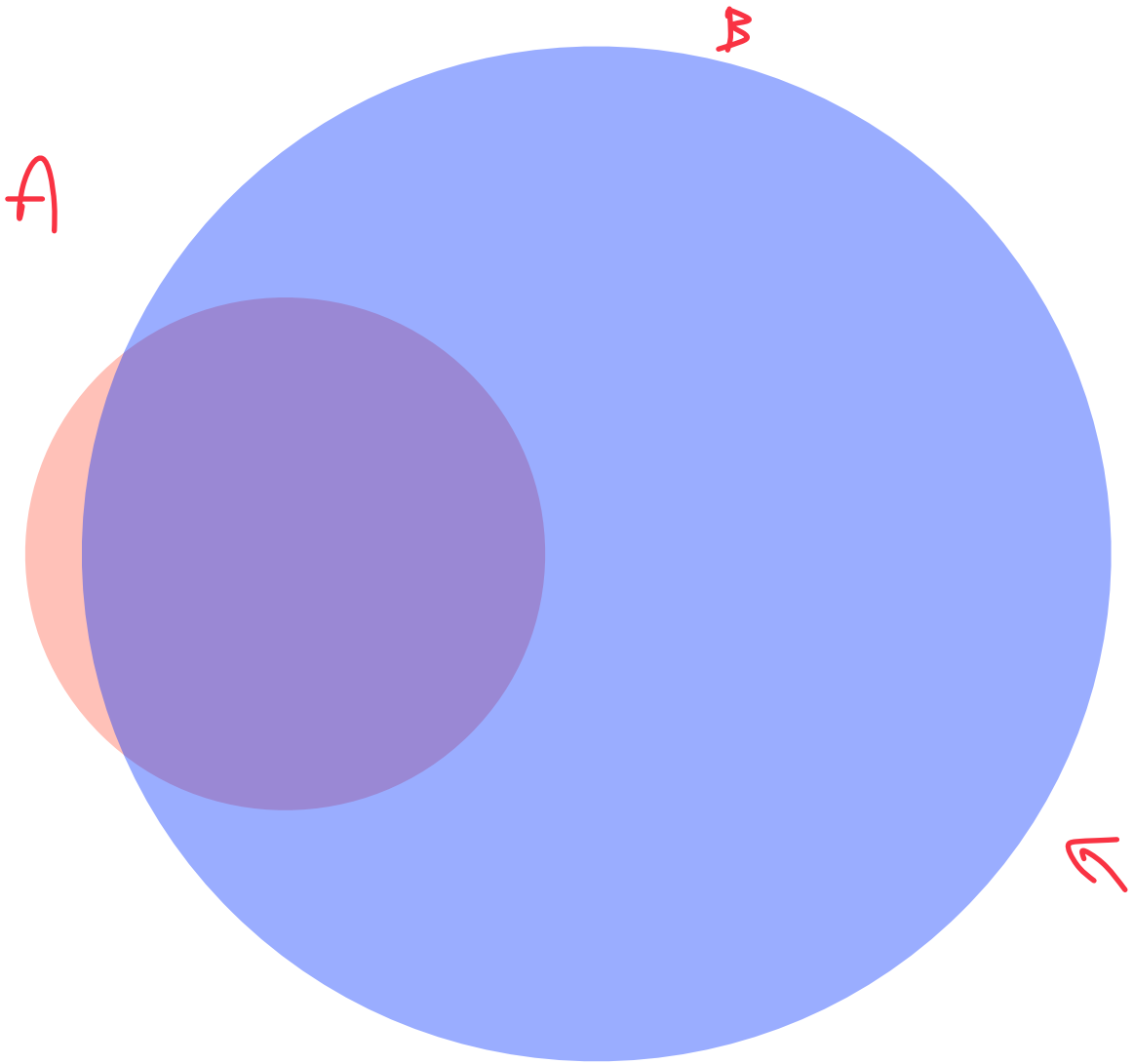


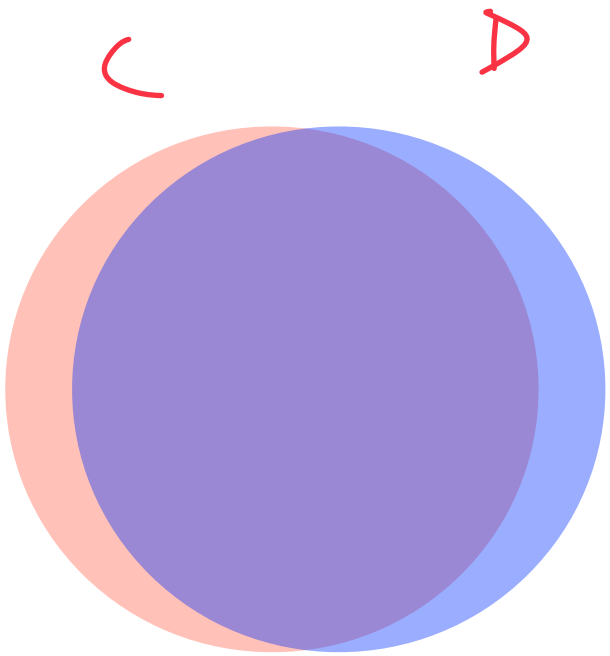


Intersection of C & D > A & B

Set Similarity Review

How can we describe how ***similar*** two sets are?

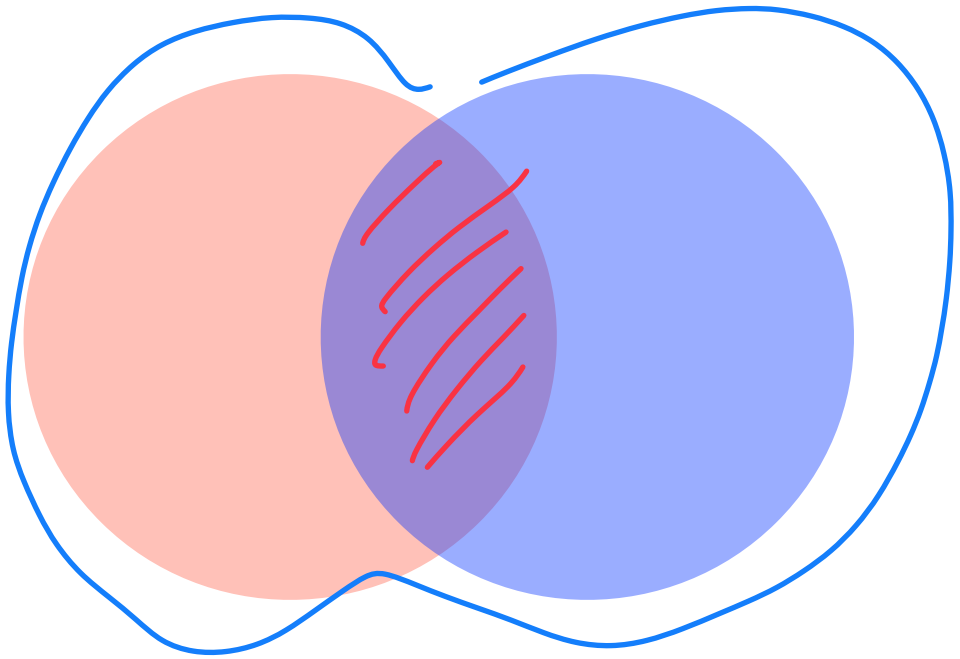




Intersection of A & B is larger than

Set Similarity Review

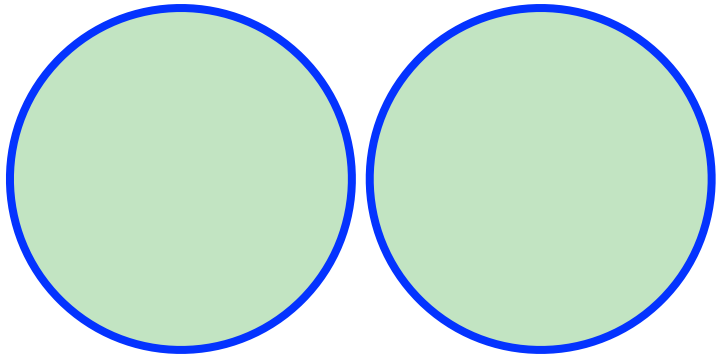
To measure **similarity** of A & B, we need both a measure of how similar the sets are but also the total size of both sets.



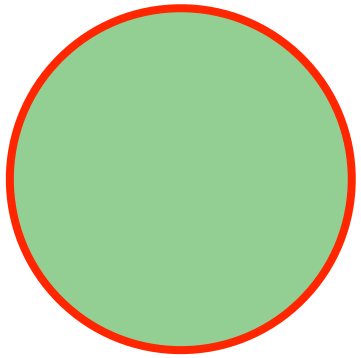
$$J = \frac{|A \cap B|}{|A \cup B|}$$

J is the ***Jaccard coefficient***

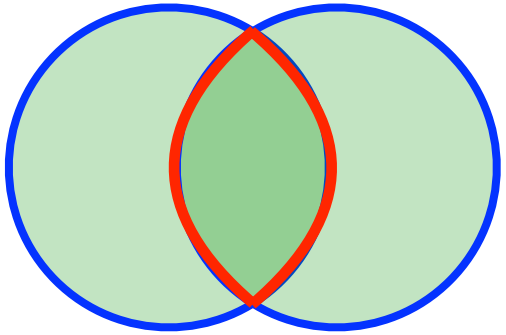
Set Similarity Review



$$\frac{|A \cap B|}{|A \cup B|} = 0$$



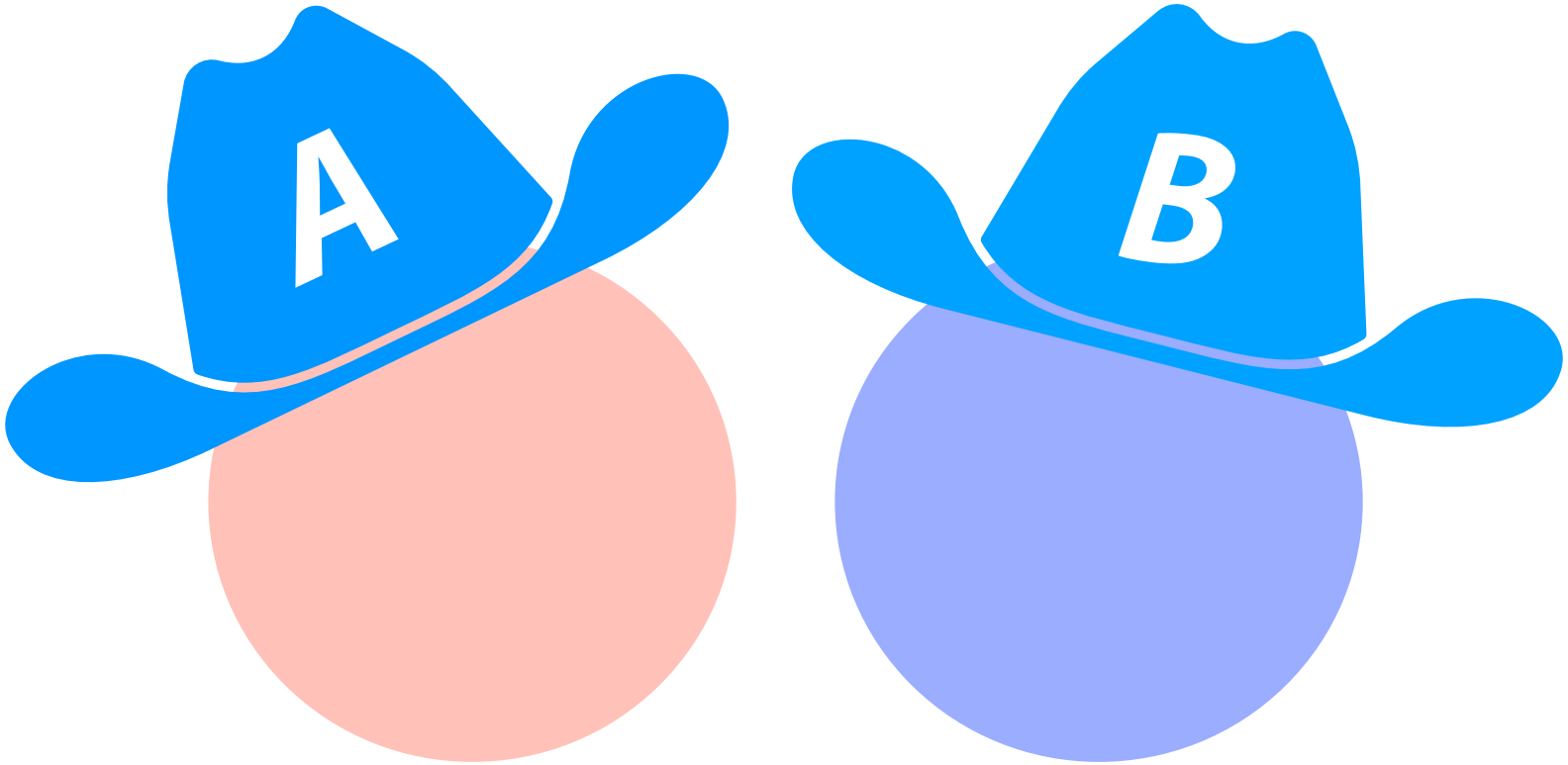
$$\frac{|A \cap B|}{|A \cup B|} = 1$$



$$0 < \frac{|A \cap B|}{|A \cup B|} < 1$$

Similarity Sketches

But what do we do when we only have a sketch?



Similarity Sketches

Imagine we have two datasets represented by their kth minimum values

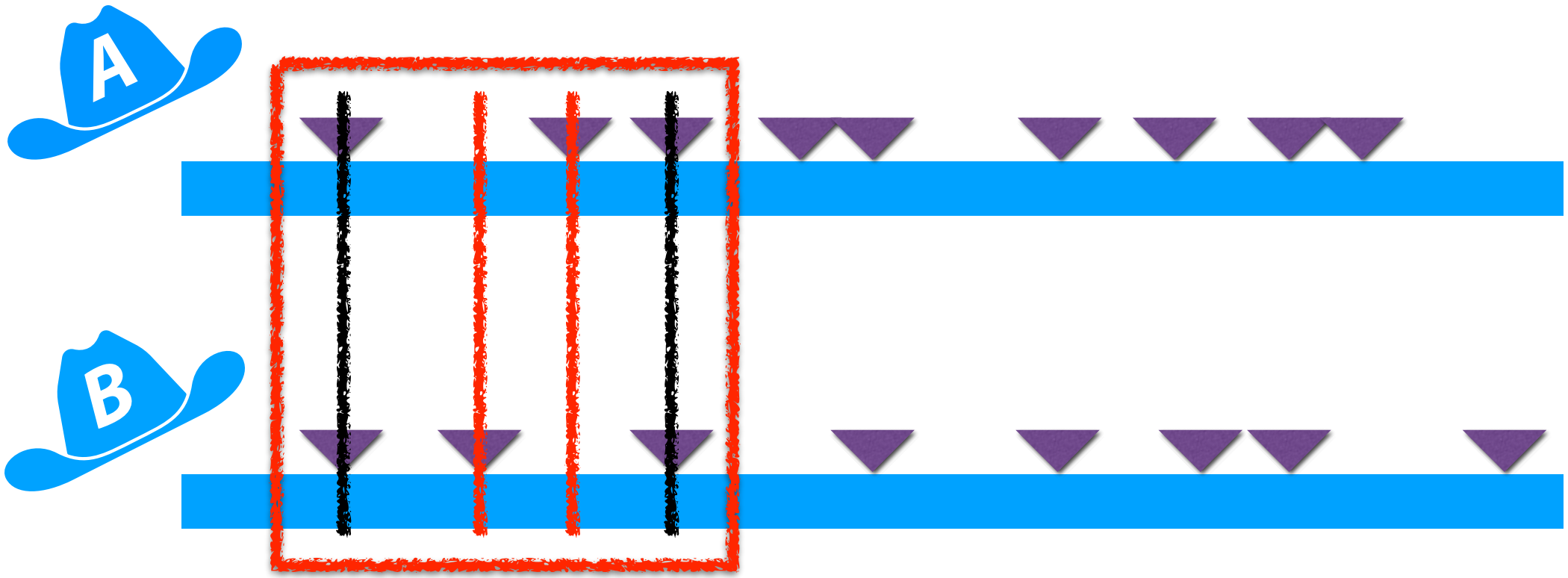


Image inspired by: Ondov B, Starrett G, Sappington A, Kostic A, Koren S, Buck CB, Phillippy AM. **Mash Screen: high-throughput sequence containment estimation for genome discovery**. *Genome Biol* 20, 232 (2019)

Similarity Sketches

Claim: Under SUHA, set similarity can be estimated by sketch similarity!

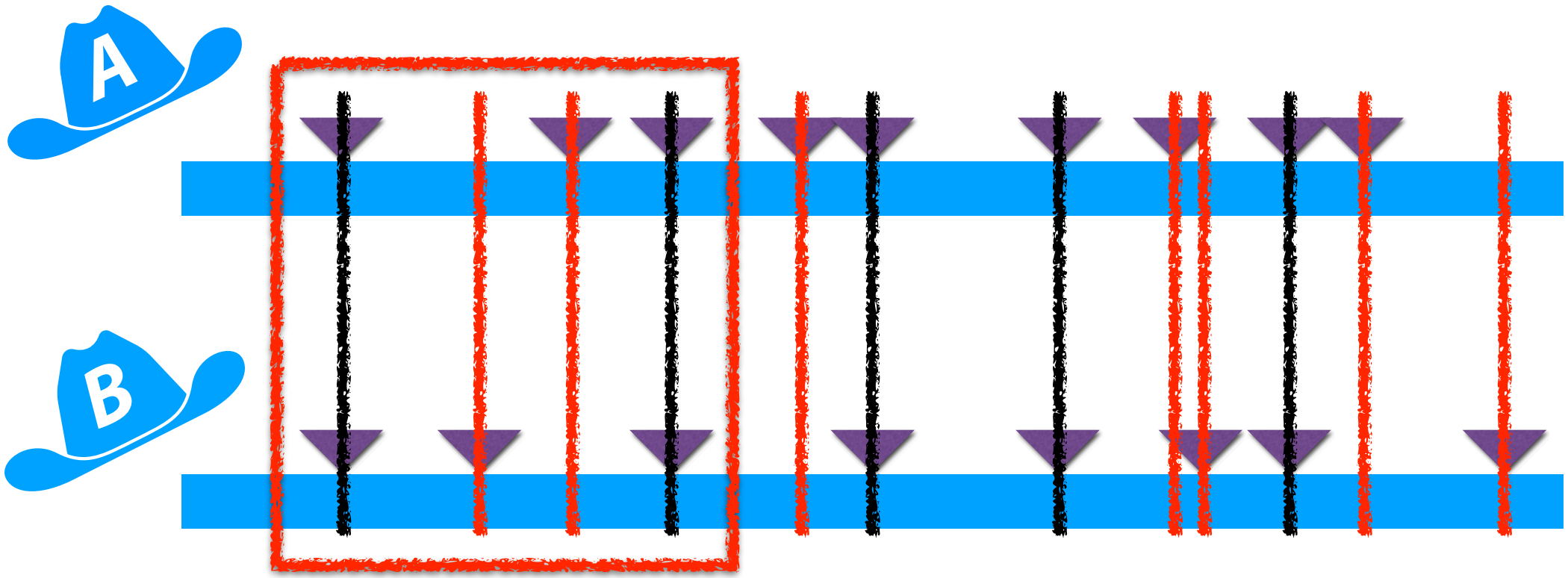


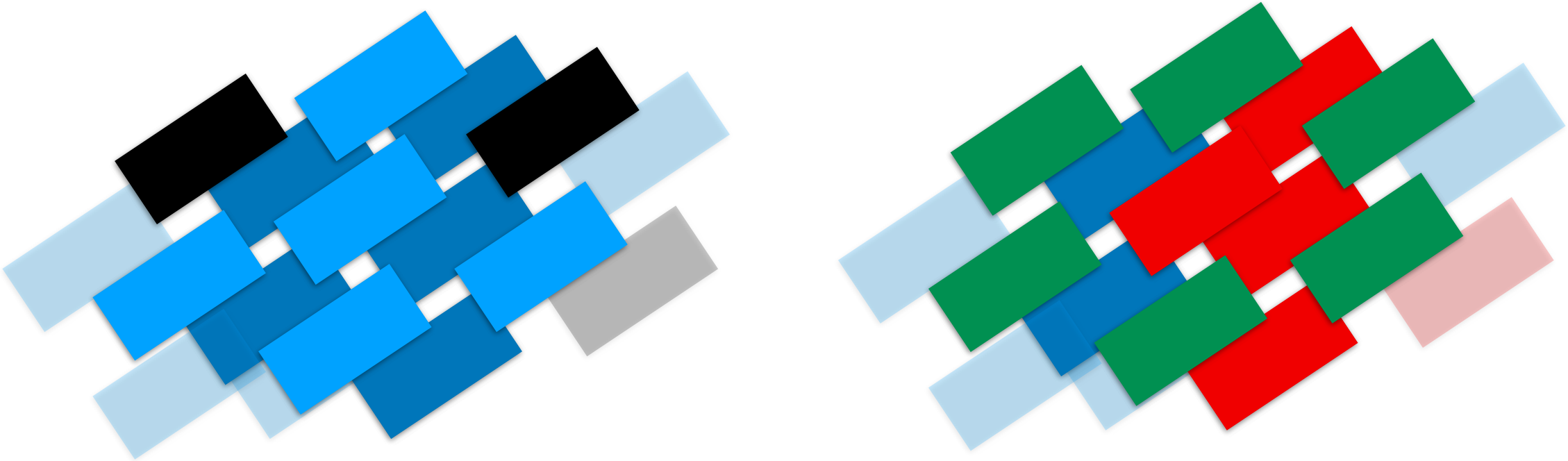
Image inspired by: Ondov B, Starrett G, Sappington A, Kostic A, Koren S, Buck CB, Phillippy AM. **Mash Screen: high-throughput sequence containment estimation for genome discovery**. *Genome Biol* 20, 232 (2019)

MinHash Sketch

The **k-th minimum value sketch** is built by tracking k minima but only uses one value (the k-th minima) to get **cardinality!**

We can extend this approach into a full **MinHash sketch** that can also estimate **set similarities.**

All minima up to k



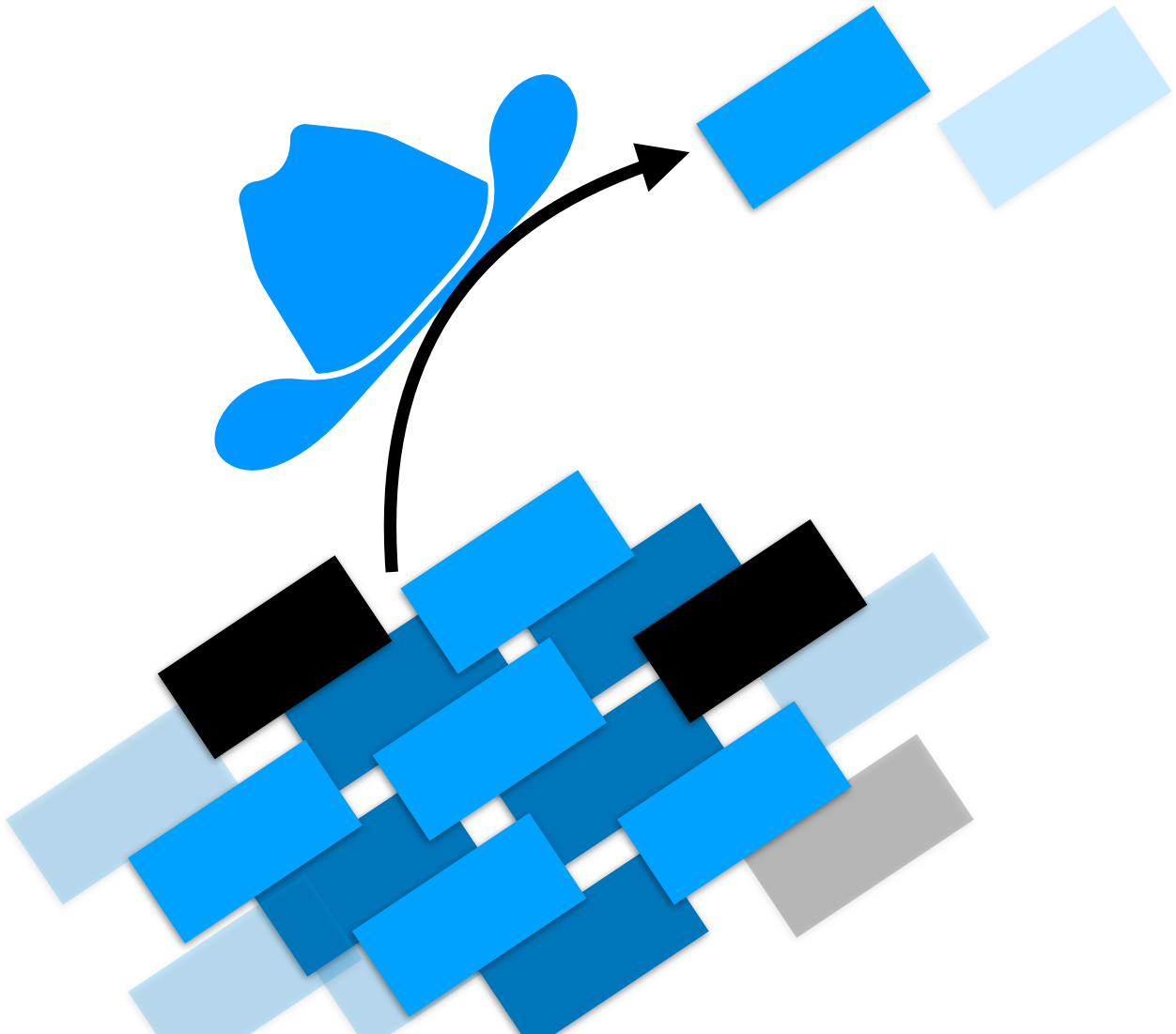
MinHash Construction

A MinHash sketch has three required inputs:

1. Dataset

2. Hash function

3. K (# of min hash values to store)



MinHash Construction

S = { 16, 8, 4, 13, 15}

h(x) = x % 7

k = 3

Algorithm is trivial:

1. Hash each item
2. Keep the k-minimum values in memory
   (Ignore collisions / duplicates)

we do not store duplicates

0	1
1	2
2	4

This stores min Hash values

MinHash Jaccard Estimation

Given sets A and B sampled uniformly from [0, 100], store the bottom-8 **MinHash:**

Goal: How similar are A & B?

Sketch A

3	15
7	17
8	22
11	23

Sketch B

2	9
3	11
6	17
7	23

MinHash Jaccard Estimation

We want to estimate the Jaccard Coefficient: $\frac{|A \cup B|}{|A \cap B|}$

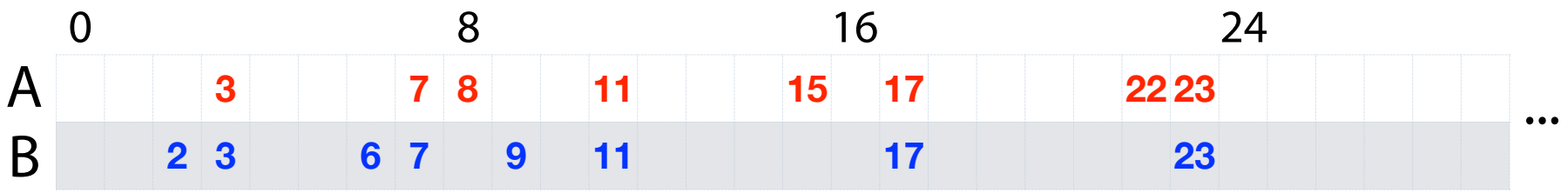
can we get union & intersection cardinalities

Sketch A

3	15
7	17
8	22
11	23

Sketch B

2	9
3	11
6	17
7	23



MinHash Jaccard Estimation

What do we know about $A \cup B$?

(Hash values of)

Sketch A

3	15
7	17
8	22
11	23

Sketch B

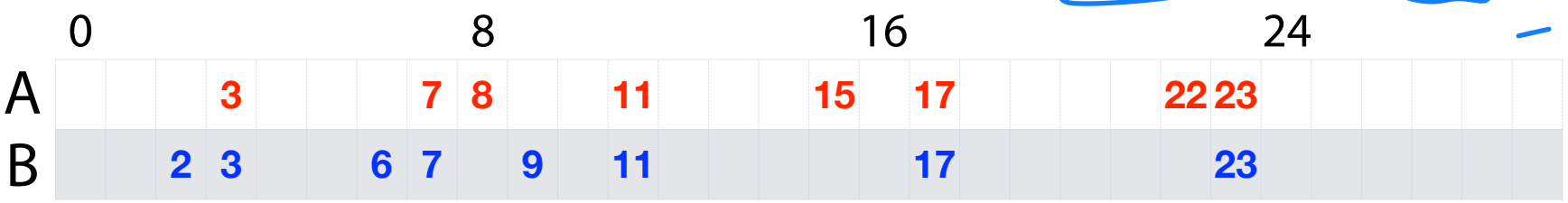
2	9
3	11
6	17
7	23

$A \cup B$

2	8	17
3	9	22
6	11	23
7	15	

...

could also have done 11 minhash



– Bottom 8 ... of $A \cup B$

MinHash Jaccard Estimation

8 smallest hash values in $A \cup B$ is in bottom 8 A or bottom 8 B

We dont *know* $A \cup B$, but we can make a sketch!

Sketch A

3	15
7	17
8	22
11	23

$\cup$

Sketch B

2	9
3	11
6	17
7	23

$=$

Sketch $A \cup B$

2	8
3	9
6	11
7	15

MinHash Jaccard Estimation

Estimate $|A \cup B|$ (the cardinality of the union) from sketch:

Sketch $A \cup B$

Our sets sampled from [0, 100].

2	8
3	9
6	11
7	15

Hash values

$$\frac{15}{100} = \frac{\text{kth min hash value}}{\text{universe max}} = \frac{k=8}{N+1}$$

$$\frac{15}{100} = \frac{8}{N+1}$$

$$N+1 = \frac{800}{15}$$

$$N = \frac{800}{15} - 1 = 52.3$$

MinHash Jaccard Estimation

Can we build a 8-Minhash of $A \cap B$?

Sketch A

3	15
7	17
8	22
11	23

$\cap$

Sketch B

2	9
3	11
6	17
7	23

$=$

Sketch $A \cap B$

3	23
7	??
11	??
17	??

	0							8							16							24							
A				3			7	8			11				15		17					22	23						
B			2	3		6	7		9		11						17						23						

...

MinHash Jaccard Estimation

Not guaranteed to be able to get a full sketch of the intersection!

8 min hash values which intersect

Sketch A

3	15
7	17
8	22
11	23

$\cap$

Sketch B

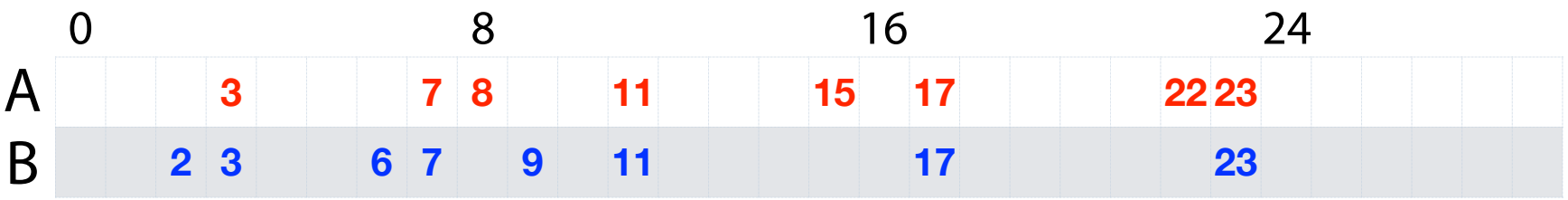
2	9
3	11
6	17
7	23

$=$

Sketch $A \cap B$

3	23
7	
11	
17	

5 th min hash cardinality estimate is possible



... 35 51 200

35 51 200

MinHash Jaccard Estimation

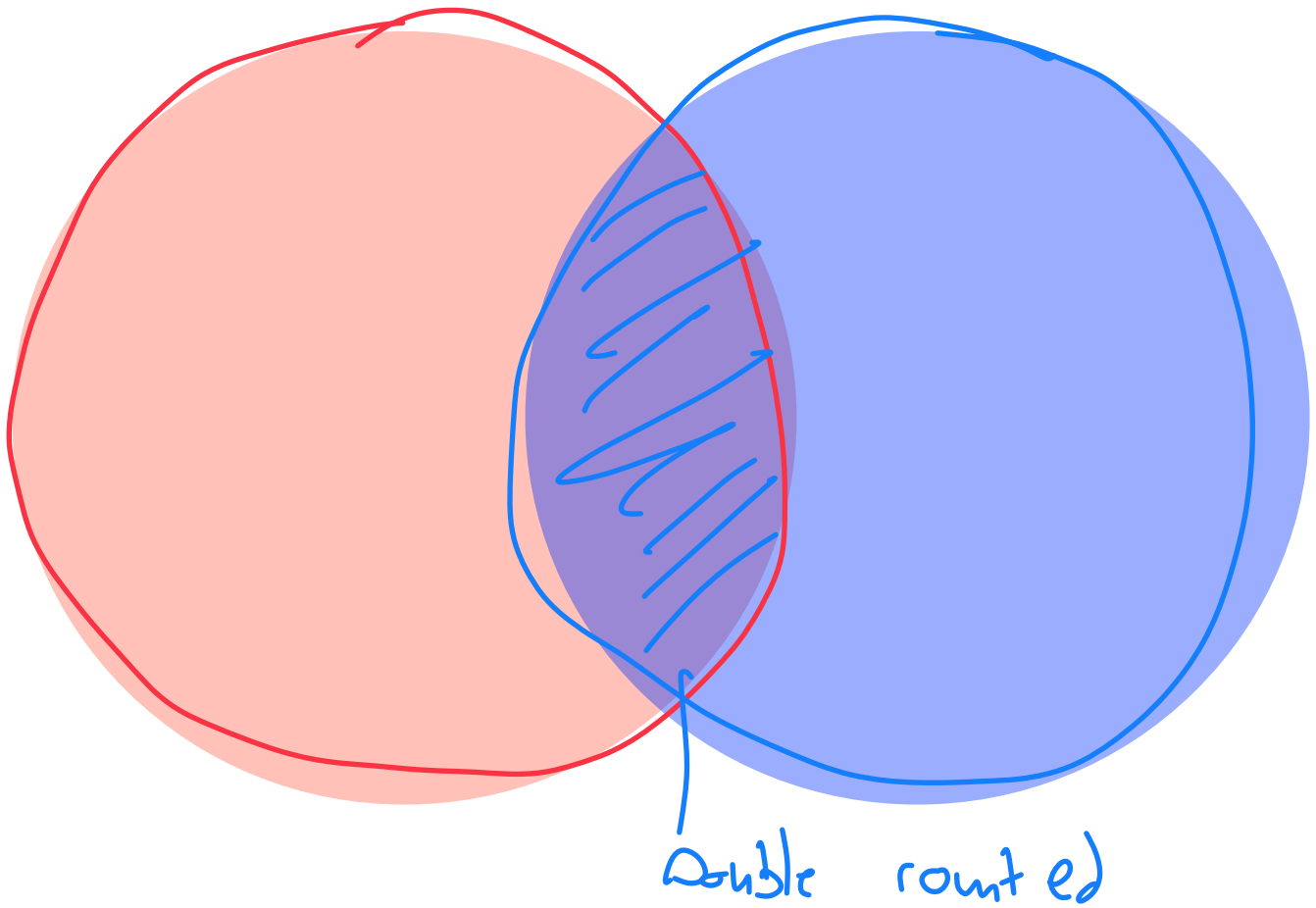
Using MinHash sketches, we can estimate $|A|$, $|B|$, and $|A \cup B|$

Is this enough to estimate the Jaccard?

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Inclusion-Exclusion Principle

$$|A \cap B| = |A| + |B| - |A \cup B|$$



MinHash Indirect Jaccard Estimation

$$\frac{|A| \cap |B|}{|A| \cup |B|} = \frac{|A| + |B| - |A \cup B|}{|A \cup B|}$$

$k = 8$ MinHash sketches

Our sets sampled from [0, 100]

Sketch A

3	15
7	17
8	22
11	23

Sketch B

2	9
3	11
6	17
7	23

Sketch of $|A \cup B|$

2	8
3	9
6	11
7	15

$$= \frac{\overset{|A|}{(800/23-1)} + \overset{|B|}{(800/23-1)} - (800/15-1)}{800/15-1}$$

$$= \frac{34.782 + 34.782 - 53.333 - 1}{53.333 - 1} \approx 0.29$$

(Jaccard) Set similarity

MinHash Direct Jaccard Estimate

We can also estimate cardinality directly using our sketches!

Sketch A

3	15
7	17
8	22
11	23

Sketch B

2	9
3	11
6	17
7	23

Intersection

3	23
7	
11	
17	

5

Union

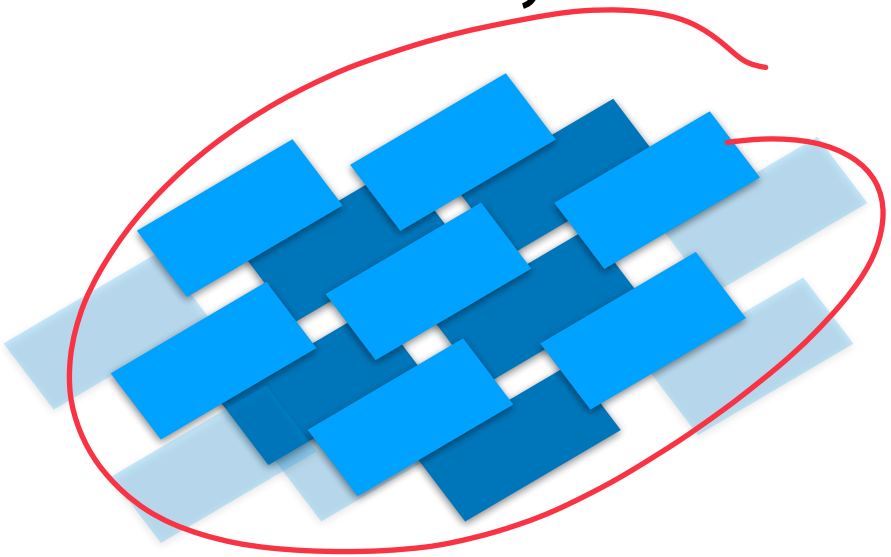
2	8	17
3	9	22
6	11	23
7	15	

11

5/11

MinHash Sketch

We can convert any hashable dataset into a **MinHash sketch**



{1 , 3 , 5}

We lose our original dataset, but we can still estimate two things:

1. Cardinality

2. Set similarity

Alternative MinHash Sketch Approaches

Rather than use one single hashes and take bottom-k, we can also use k hashes — **if you have access to that many independent hashes**!

1) Sequence decomposed into **kmers**

k-length subsequences

```
S1 : CATGGACCGACCAG        GCAGTACCGATCGT : S2
     CAT GAC GAC              GTA CGA CGT
      ATG ACC ACC            AGT CCG TCG
       TGG CCG CCA          CAG ACC ATC
        GGA CGA CAG        GCA TAC GAT
```

Assembling large genomes with single-molecule sequencing and locality-sensitive hashing
Berlin et al (2015) *Nature Biotechnology*

1) Sequence decomposed into **kmers**

2) Multiple hash functions (**Γ**) map kmers to values.

S_1 : CATGGACCGACCAG

CAT GAC GAC
ATG ACC ACC
TGG CCG CCA
GGA CGA CAG

Γ_1	Γ_2	Γ_3	Γ_4	
19	14	57	36	CAT
14	57	36	19	ATG
58	37	16	15	TGG
40	23	2	61	GGA
33	28	11	54	GAC
5	48	47	26	ACC
22	1	60	43	CCG
24	7	50	45	CGA
33	28	11	54	GAC
5	48	47	26	ACC
20	3	62	41	CCA
18	13	56	39	CAG

GCAGTACCGATCGT : S_2

GTA CGA CGT
AGT CCG TCG
CAG ACC ATC
GCA TAC GAT

	Γ_1	Γ_2	Γ_3	Γ_4
GCA	36	19	14	57
CAG	18	13	56	39
AGT	11	54	33	28
GTA	44	27	6	49
TAC	49	44	27	6
ACC	5	48	47	26
CCG	22	1	60	43
CGA	24	7	50	45
GAT	35	30	9	52
ATC	13	56	39	18
TCG	54	33	28	11
CGT	27	6	49	44

Assembling large genomes with single-molecule sequencing and locality-sensitive hashing
Berlin et al (2015) *Nature Biotechnology*

1) Sequence decomposed into **kmers**

2) Multiple hash functions (**Γ**) map kmers to values.

3) The smallest values for each hash function is chosen

S_1 : CATGGACCGACCAG

CAT GAC GAC
ATG ACC ACC
TGG CCG CCA
GGA CGA CAG

Γ_1	Γ_2	Γ_3	Γ_4	
19	14	57	36	CAT
14	57	36	19	ATG
58	37	16	15	TGG
40	23	2	61	GGA
33	28	11	54	GAC
5	48	47	26	ACC
22	1	60	43	CCG
24	7	50	45	CGA
33	28	11	54	GAC
5	48	47	26	ACC
20	3	62	41	CCA
18	13	56	39	CAG

[5, 1, 2, 15]
Sketch (S_1)

GCAGTACCGATCGT : S_2

GTA CGA CGT
AGT CCG TCG
CAG ACC ATC
GCA TAC GAT

	Γ_1	Γ_2	Γ_3	Γ_4
GCA	36	19	14	57
CAG	18	13	56	39
AGT	11	54	33	28
GTA	44	27	6	49
TAC	49	44	27	6
ACC	5	48	47	26
CCG	22	1	60	43
CGA	24	7	50	45
GAT	35	30	9	52
ATC	13	56	39	18
TCG	54	33	28	11
CGT	27	6	49	44

[5, 1, 6, 6]
Sketch (S_2)

Assembling large genomes with single-molecule sequencing and locality-sensitive hashing
Berlin et al (2015) *Nature Biotechnology*

1) Sequence decomposed into **kmers**

2) Multiple hash functions (**Γ**) map kmers to values.

3) The smallest values for each hash function is chosen

4) The Jaccard similarity can be estimated by the overlap in the **Min**imum **Hash**es (**MinHash)**

S_1 : CATGGACCGACCAG

CAT GAC GAC
ATG ACC ACC
TGG CCG CCA
GGA CGA CAG

GCAGTACCGATCGT : S_2

GTA CGA CGT
AGT CCG TCG
CAG ACC ATC
GCA TAC GAT

O(1)

Γ_1	Γ_2	Γ_3	Γ_4	
19	14	57	36	CAT
14	57	36	19	ATG
58	37	16	15	TGG
40	23	2	61	GGA
33	28	11	54	GAC
5	48	47	26	ACC
22	1	60	43	CCG
24	7	50	45	CGA
33	28	11	54	GAC
5	48	47	26	ACC
20	3	62	41	CCA
18	13	56	39	CAG

	Γ_1	Γ_2	Γ_3	Γ_4
GCA	36	19	14	57
CAG	18	13	56	39
AGT	11	54	33	28
GTA	44	27	6	49
TAC	49	44	27	6
ACC	5	48	47	26
CCG	22	1	60	43
CGA	24	7	50	45
GAT	35	30	9	52
ATC	13	56	39	18
TCG	54	33	28	11
CGT	27	6	49	44

O(n)

[5, 1, 2, 15]
Sketch (S_1)

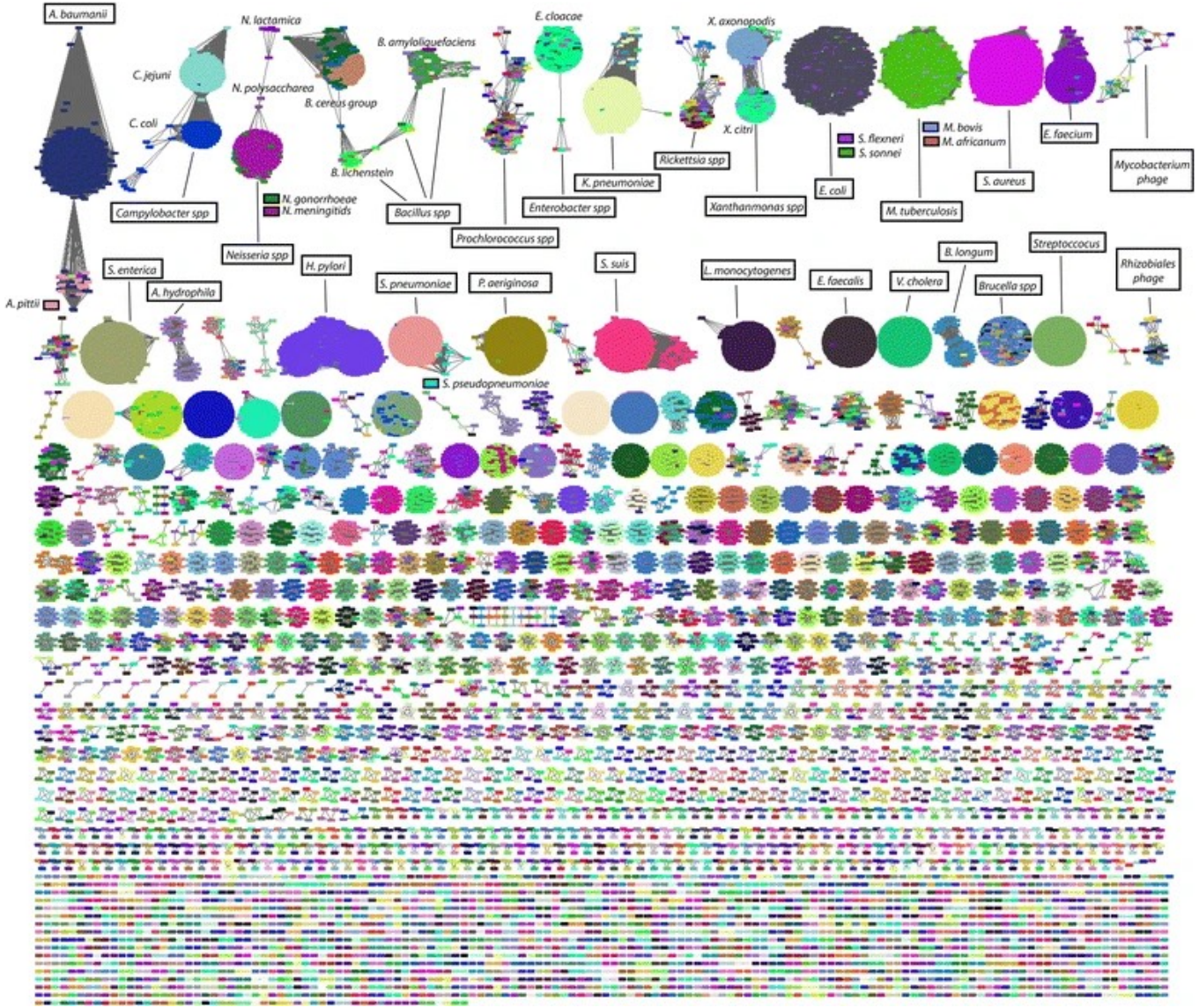
[5, 1, 6, 6]
Sketch (S_2)

$J(S_1, S_2) \approx 2/4 = 0.5$

S_1 : CATGGACCGACCAG
S_2 : GCAGTACCGATCGT

Assembling large genomes with single-molecule sequencing and locality-sensitive hashing
Berlin et al (2015) *Nature Biotechnology*

MinHash in practice



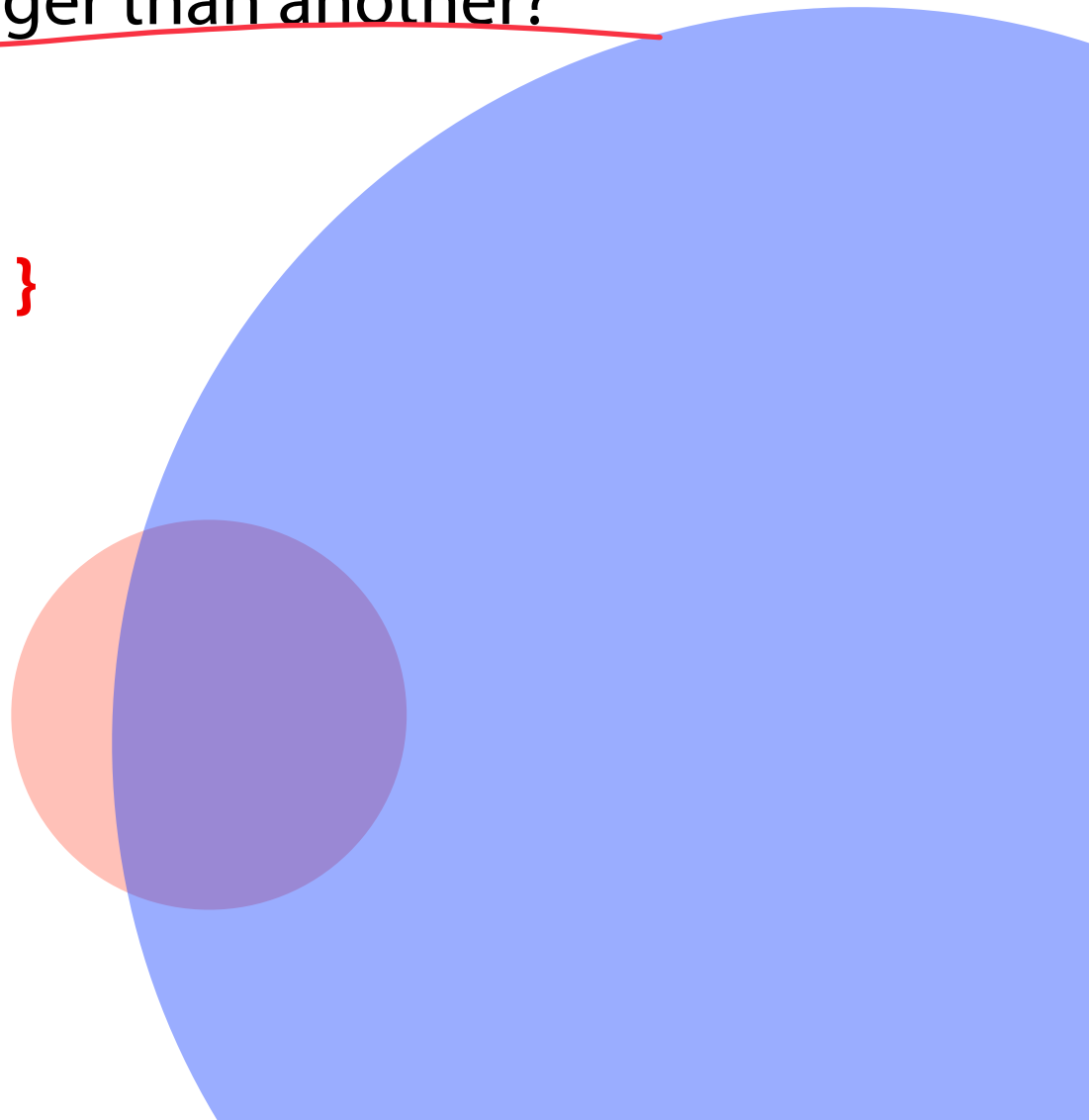
Mash: fast genome and metagenome distance estimation using MinHash
Ondov et al (2016) *Genome Biology*

Alternative MinHash Sketch Approaches

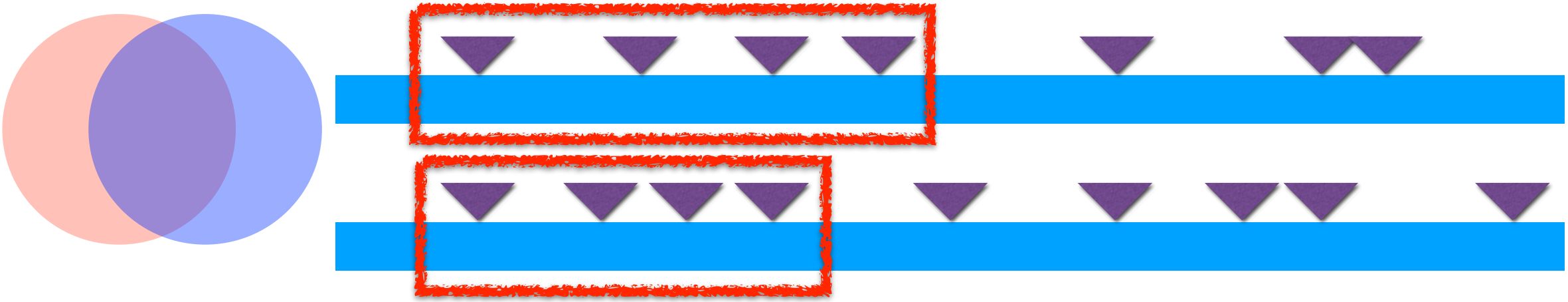
What if I have a dataset which is **much** larger than another?

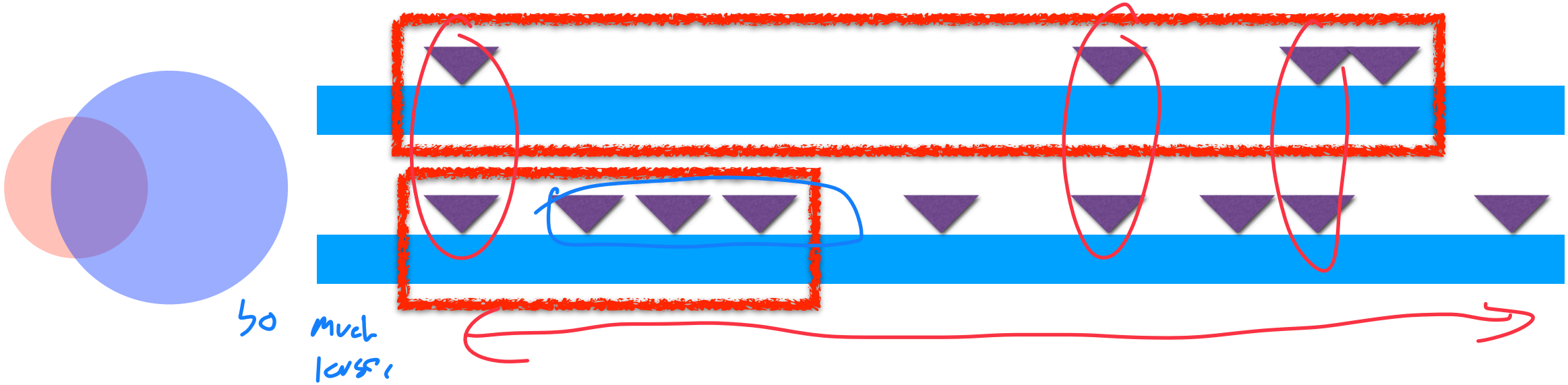
S_1 = { 1, 3, 40, 59, 82, 101 }

S_2 = { 1, 2, 3, 4, 5, 6, 7, … 59, 82, 101, … }



Bottom-K

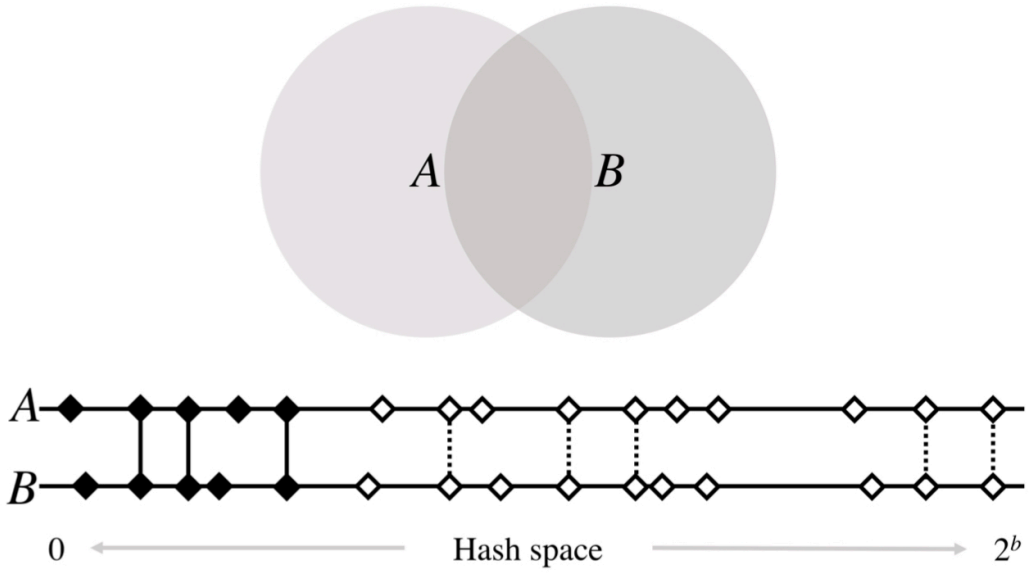


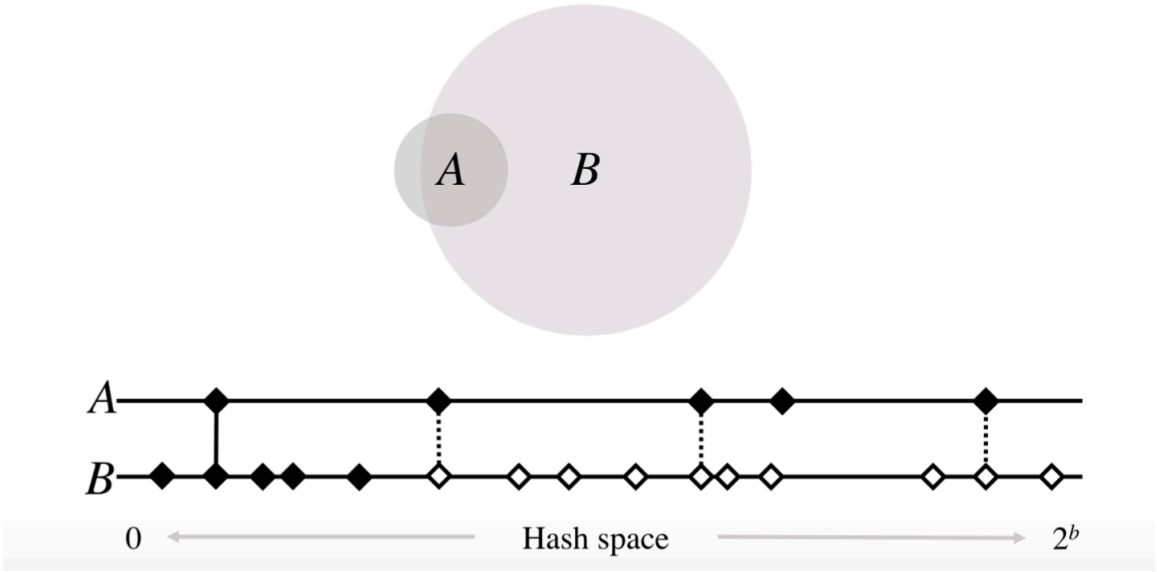


so much less.

Alternative MinHash sketches

Bottom-k minhash has low accuracy if the cardinality of sets are skewed

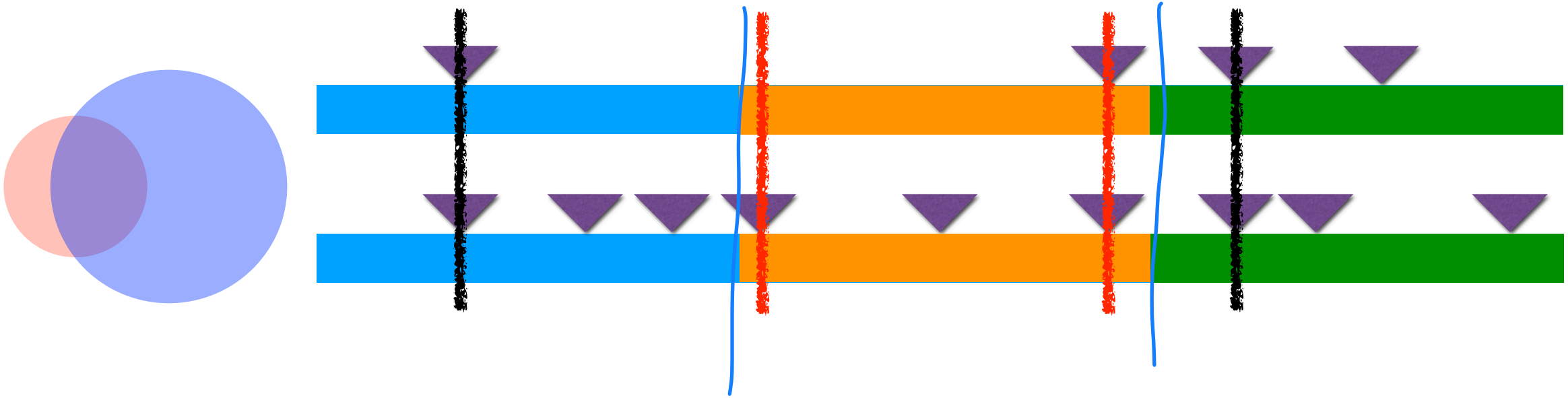




Ondov, Brian D., Gabriel J. Starrett, Anna Sappington, Aleksandra Kostic, Sergey Koren, Christopher B. Buck, and Adam M. Phillippy. **Mash Screen: High-throughput sequence containment estimation for genome discovery**. *Genome biology* 20.1 (2019): 1-13.

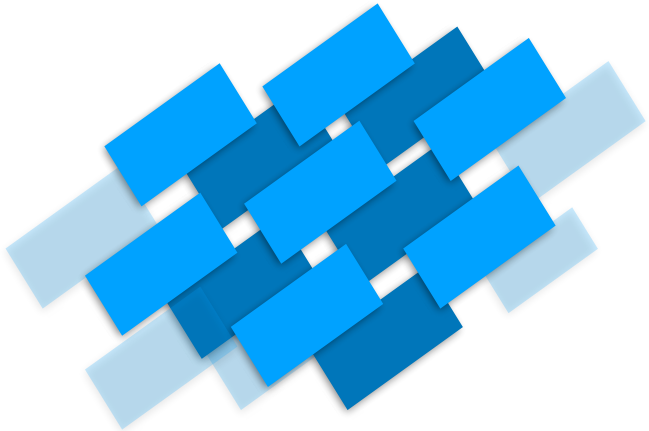
Alternative MinHash Sketch Approaches

If there is a large cardinality difference, **use k-partitions!**



Partition space, take min at each space

K-Partition Minhash



Hash

10 10110101
00 01111010
11 01101011
10 11010110
01 01100000
00 10001101

Partition

00
01111010
10001101

01
01100000

10
10110101
11010110

11
01101011

Probabilistic Data Structures

Probabilistic data structures trade accuracy for efficiency

Most can maintain surprisingly good accuracy

"Cheat" Big O limitations on conventional data analysis