

Data Structures and Algorithms

MinHash Sketch

CS 225

December 3, 2025

Brad Solomon



UNIVERSITY OF
ILLINOIS
URBANA - CHAMPAIGN

Department of Computer Science

Learning Objectives

Review the concept of cardinality and cardinality estimation

Improve our cardinality estimation approach

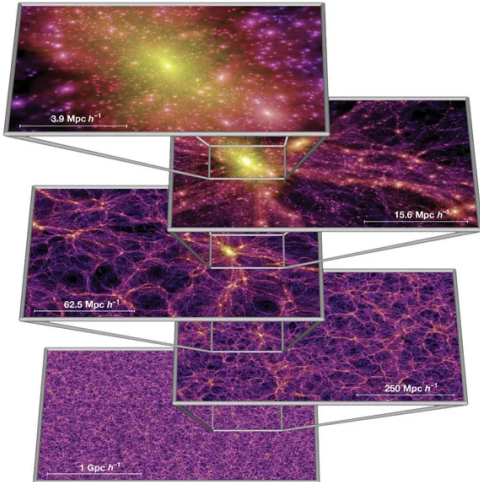
Demonstrate the relationship between cardinality and similarity

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

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

Cardinality Sketch

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

Cardinality Estimation

Let $\text{min} = 95$. 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: $95 \approx 1000/(N + 1)$

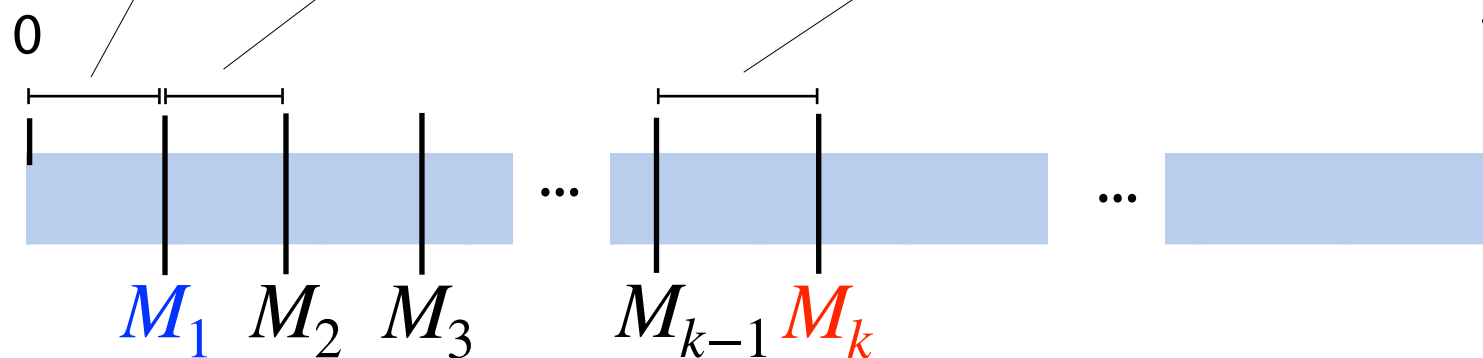
$$N + 1 \approx 10.5$$

$$N \approx 9.5$$

Cardinality Sketch

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

$$= \left[\underbrace{\mathbf{E}[M_1]}_{\text{blue}} + \underbrace{(\mathbf{E}[M_2] - \mathbf{E}[M_1])}_{\text{blue}} + \dots + \underbrace{(\mathbf{E}[M_k] - \mathbf{E}[M_{k-1}])}_{\text{red}} \right] \cdot \frac{1}{k}$$

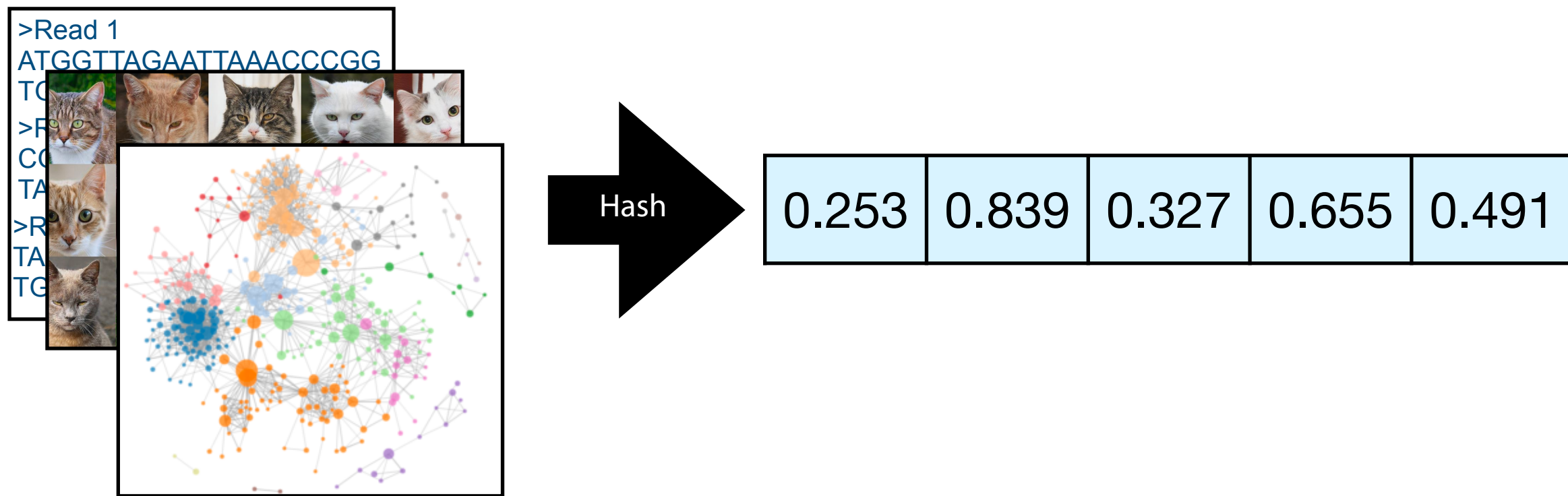


k^{th} minimum
value (KMV)

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**.



Applied Cardinalities

Cardinalities

 $|A|$ $|B|$ $|A \cup B|$ $|A \cap B|$

Set similarities

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

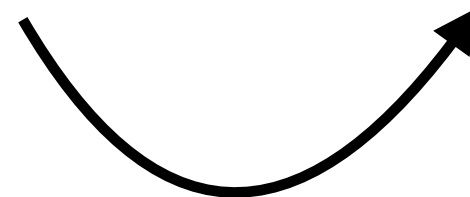
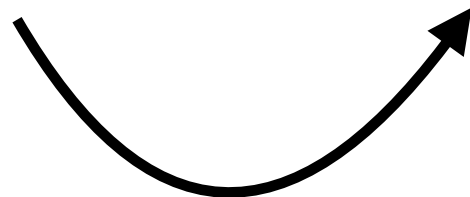
$$J = \frac{|A \cap B|}{|A \cup B|}$$

Real-world
Meaning

AGGCCACAGTGTATTATGACTG
||||||| |||||
AGGCCACAGTGAGTTATGACTG

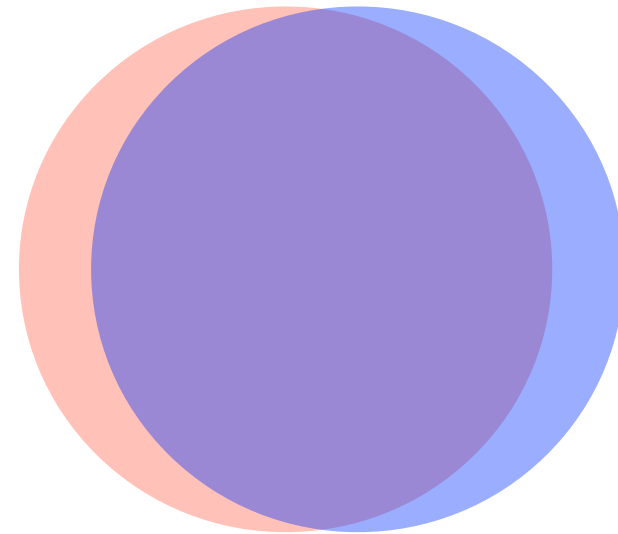
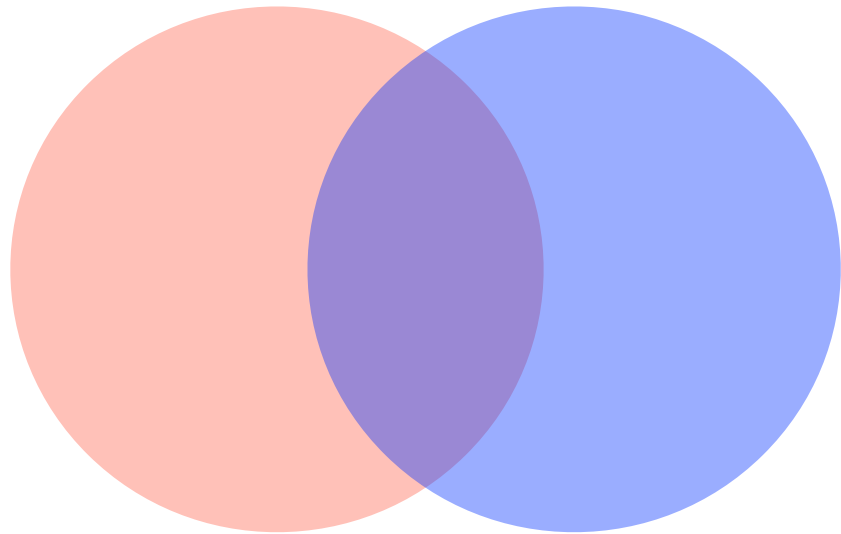
AAAAAAAAAAAGATGT-AAGTA
||||||| |||||
AAAAAAAAAAAGATGTAAAGTA

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



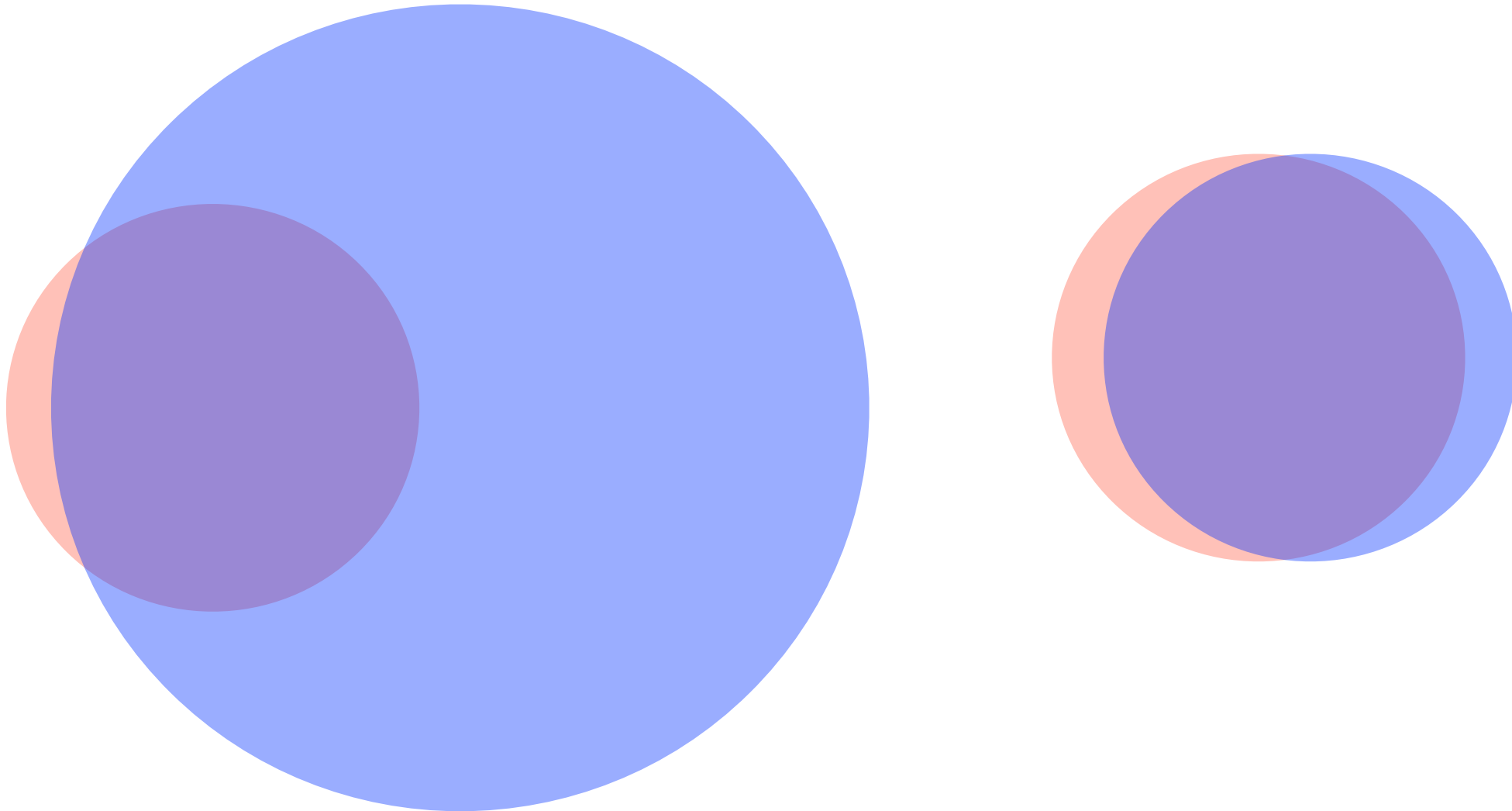
Set Similarity Review

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



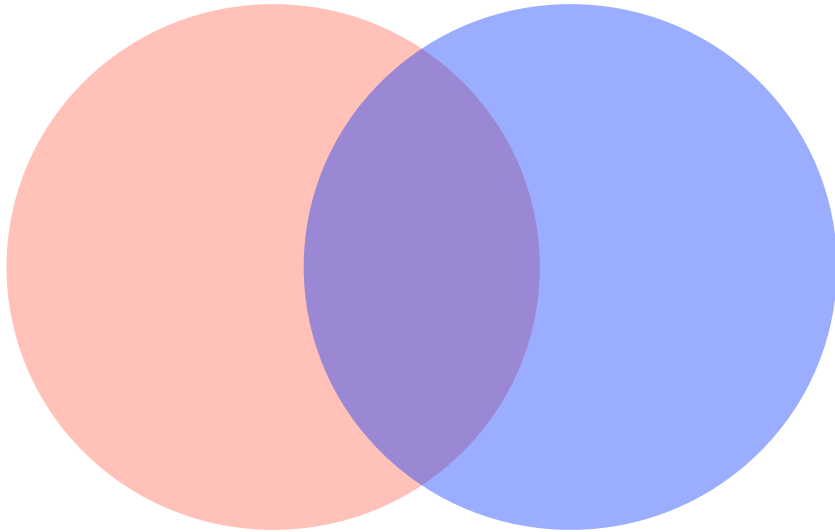
Set Similarity Review

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



Set Similarity Review

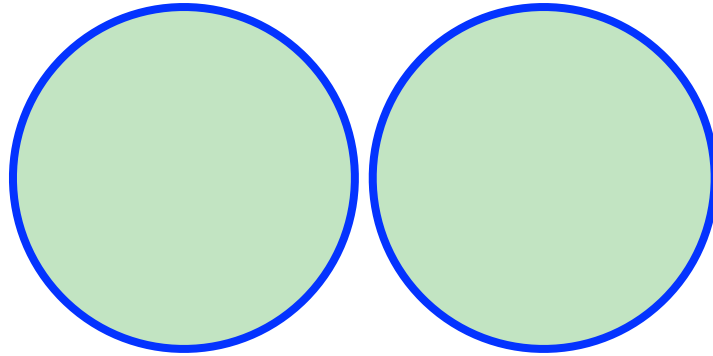
To measure **similarity** of A & B , we need both a measure of how similar the sets are but also the total size of both sets.



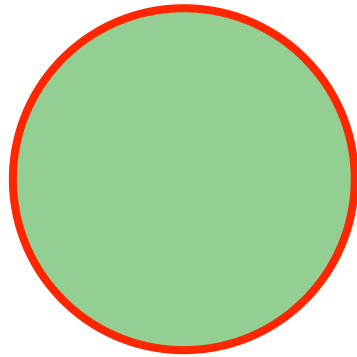
$$J = \frac{|A \cap B|}{|A \cup B|}$$

J is the ***Jaccard coefficient***

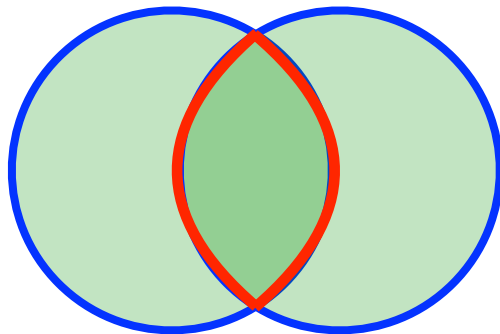
Set Similarity Review



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



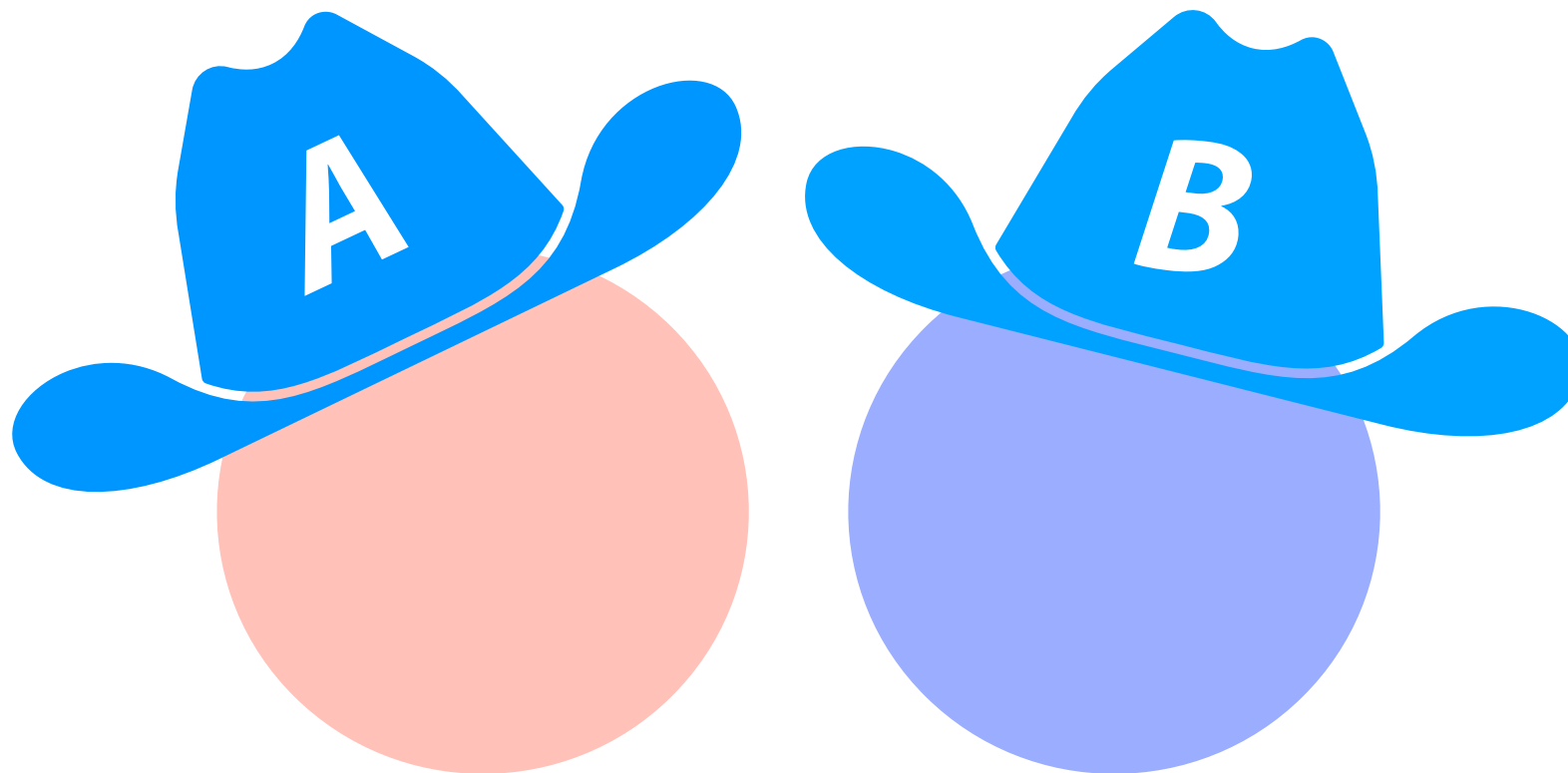
$$\frac{|A \cap B|}{|A \cup B|} = 1$$



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

Similarity Sketches

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



Similarity Sketches

Imagine we have two datasets represented by their k th minimum values

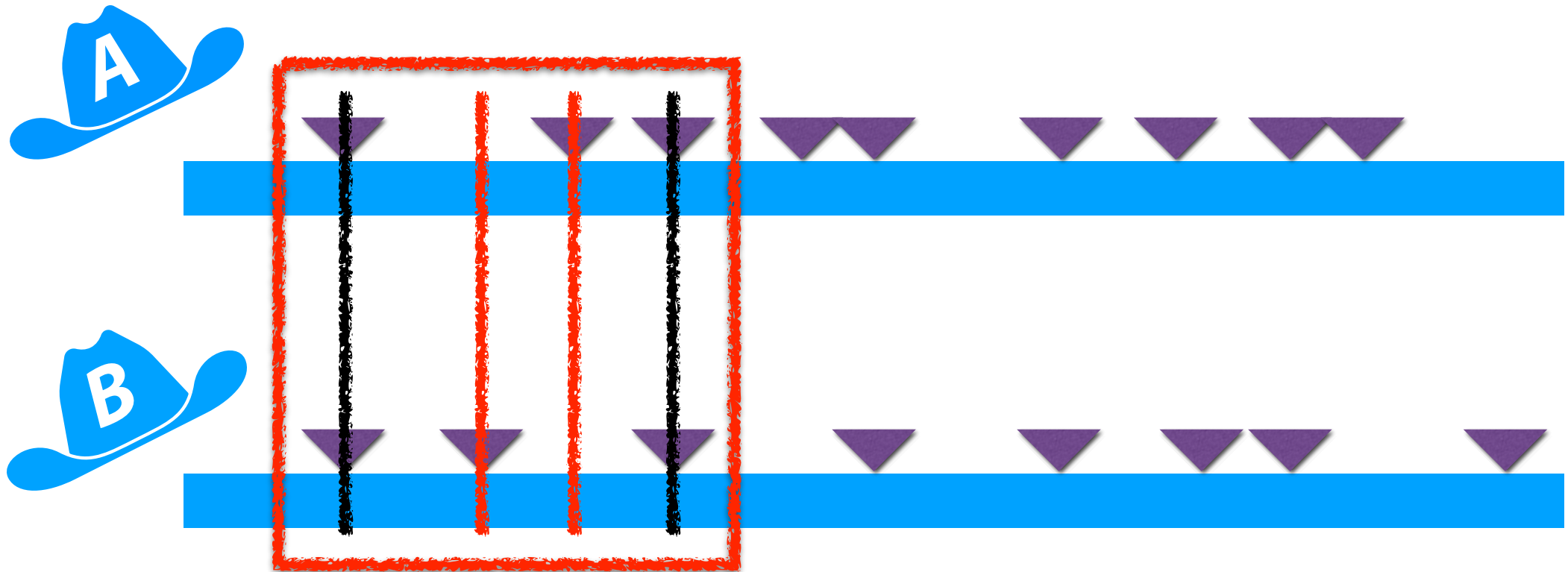


Image inspired by: Ondov B, Starrett G, Sappington A, Kostic A, Koren S, Buck CB, Phillippy AM. **Mash Screen: high-throughput sequence containment estimation for genome discovery.** *Genome Biol* 20, 232 (2019)

Similarity Sketches

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

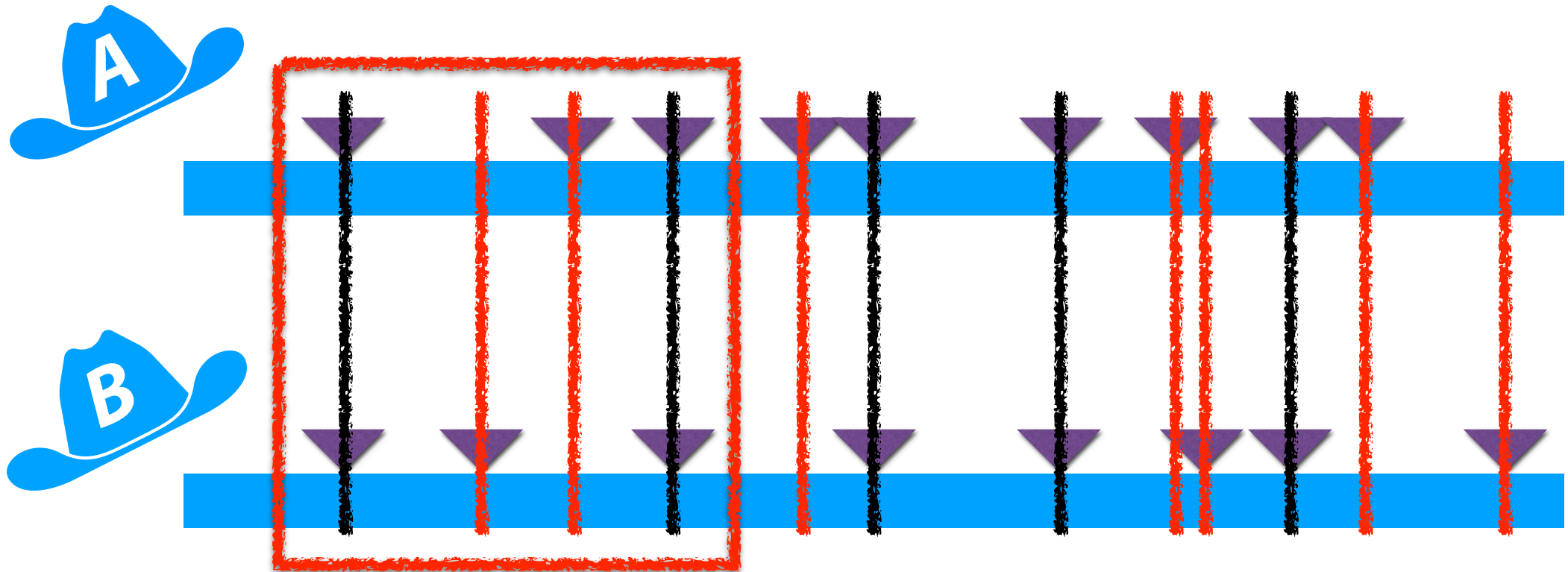


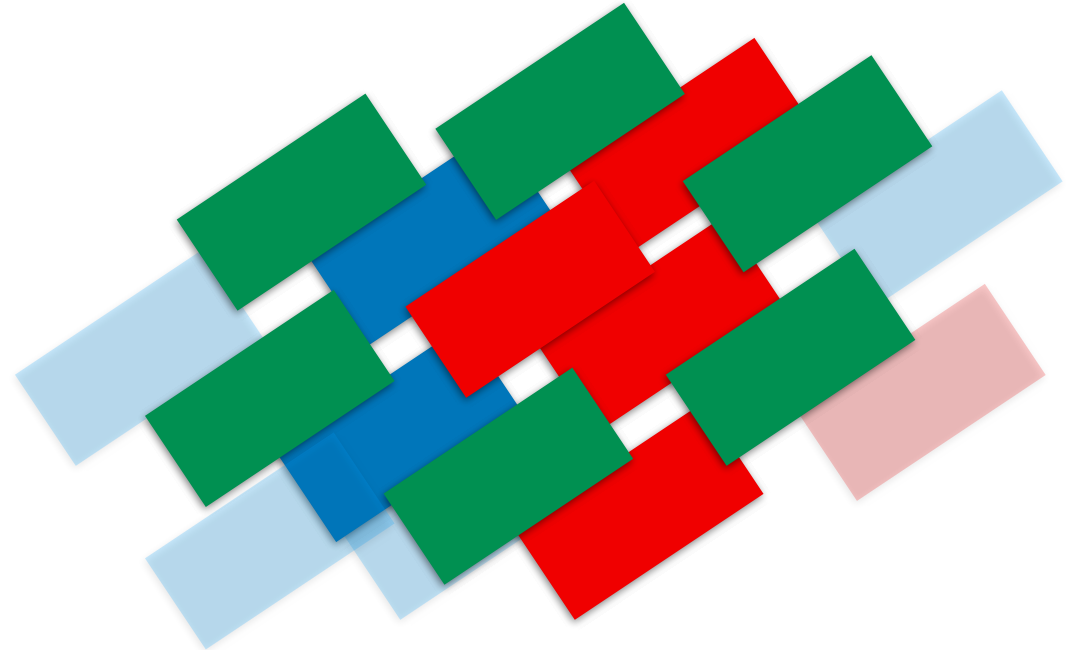
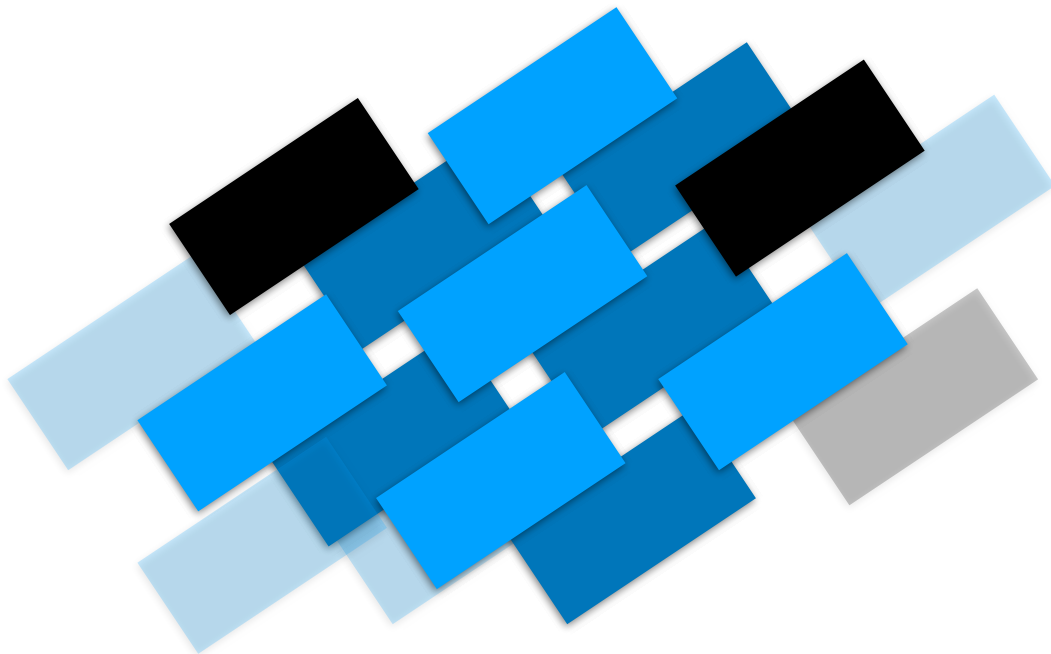
Image inspired by: Ondov B, Starrett G, Sappington A, Kostic A, Koren S, Buck CB, Phillippy AM. **Mash Screen: high-throughput sequence containment estimation for genome discovery.** *Genome Biol* 20, 232 (2019)

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**.



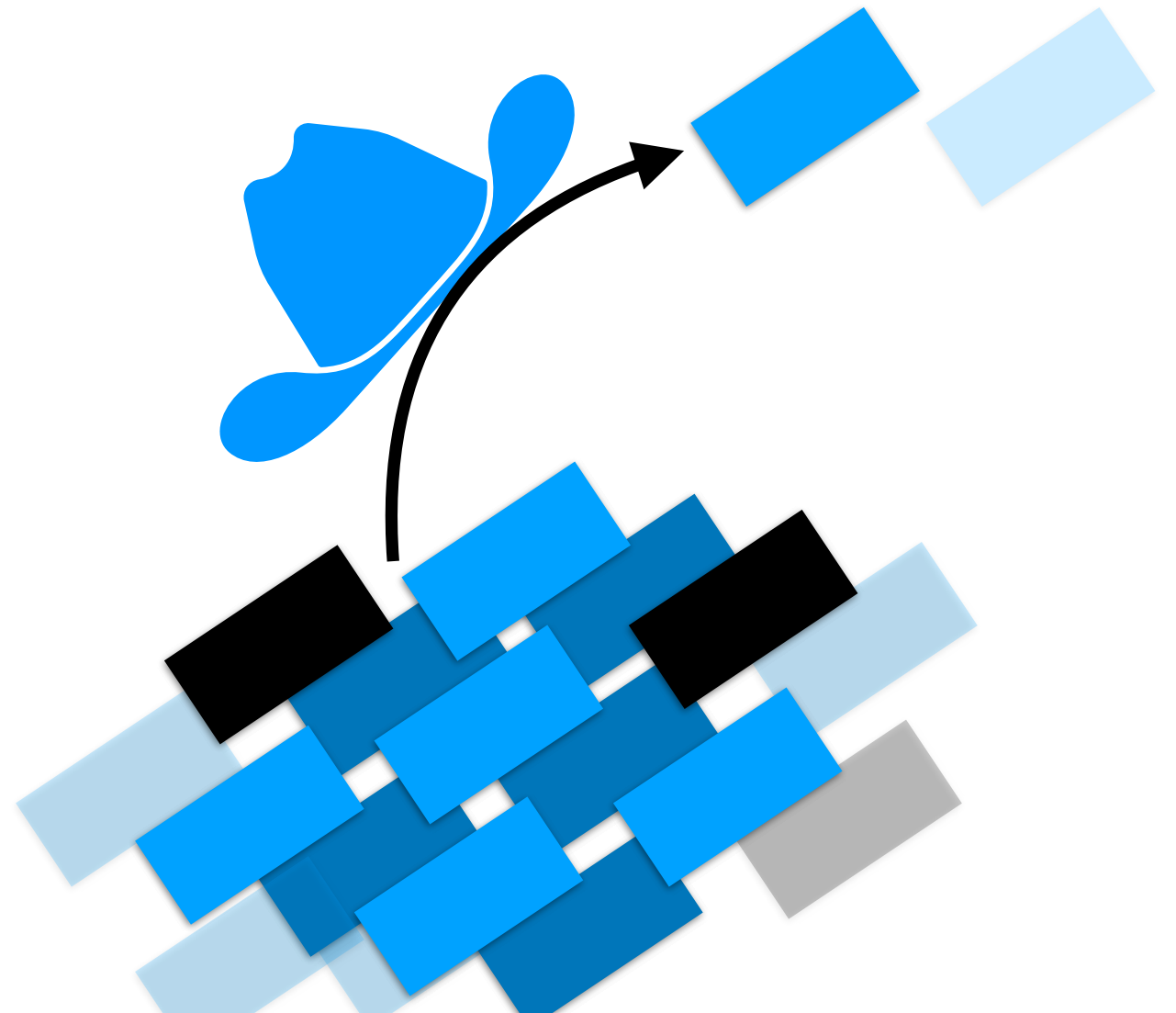
MinHash Construction

A MinHash sketch has three required inputs:

1.

2.

3.



MinHash Construction



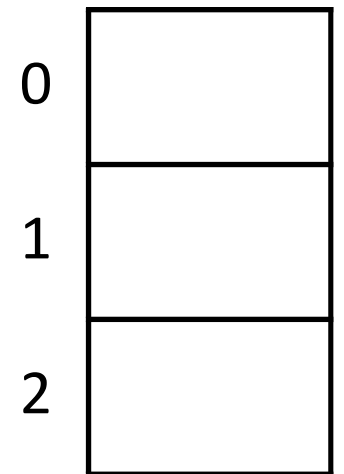
$S = \{ 16, 8, 4, 13, 15 \}$

$h(x) = x \% 7$

$k = 3$

Algorithm is trivial:

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



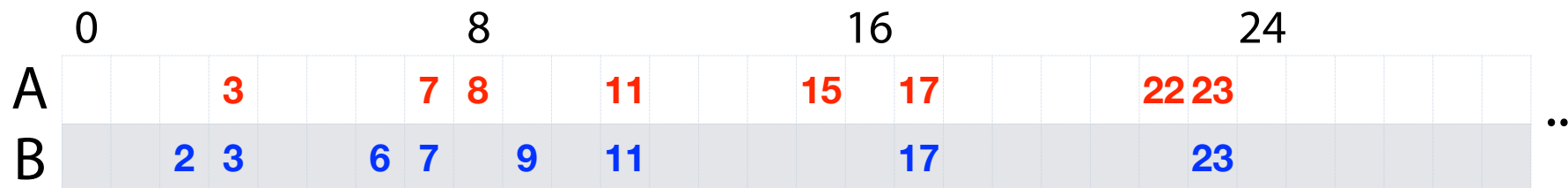
Given sets A and B sampled uniformly from $[0, 100]$, store the bottom-8 **MinHash**:

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

Sketch B

3	15
7	17
8	22
11	23

2	9
3	11
6	17
7	23



We want to estimate the Jaccard Coefficient:

Sketch A

Sketch B



B

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

Sketch A

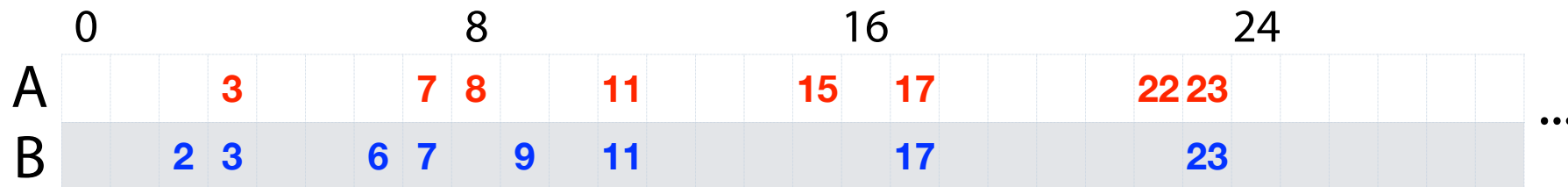
3	15
7	17
8	22
11	23

Sketch B

2	9
3	11
6	17
7	23

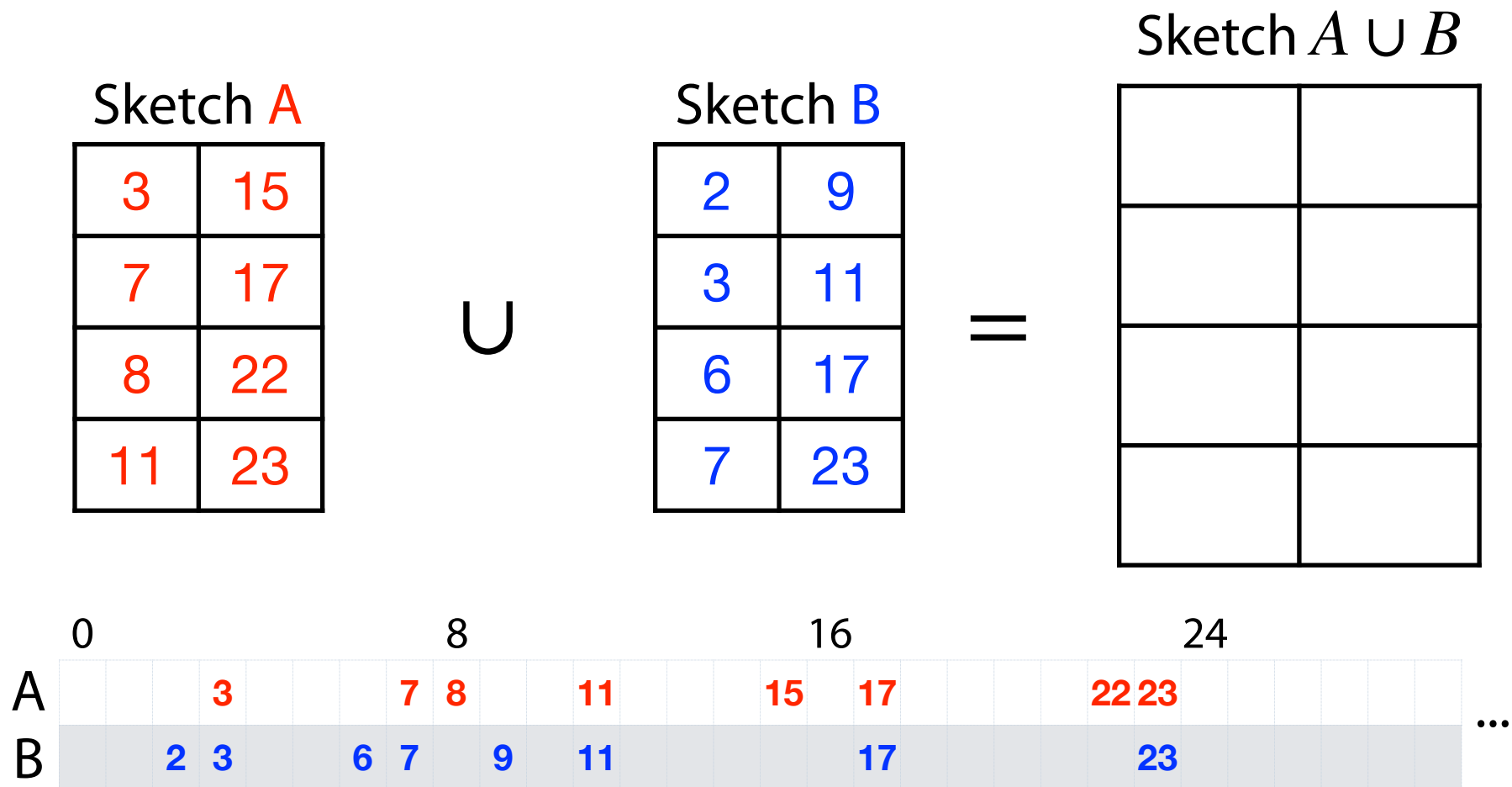
$$A \cup B$$

• •



MinHash Jaccard Estimation

We don't *know* $A \cup B$, but we can make a sketch!



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

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

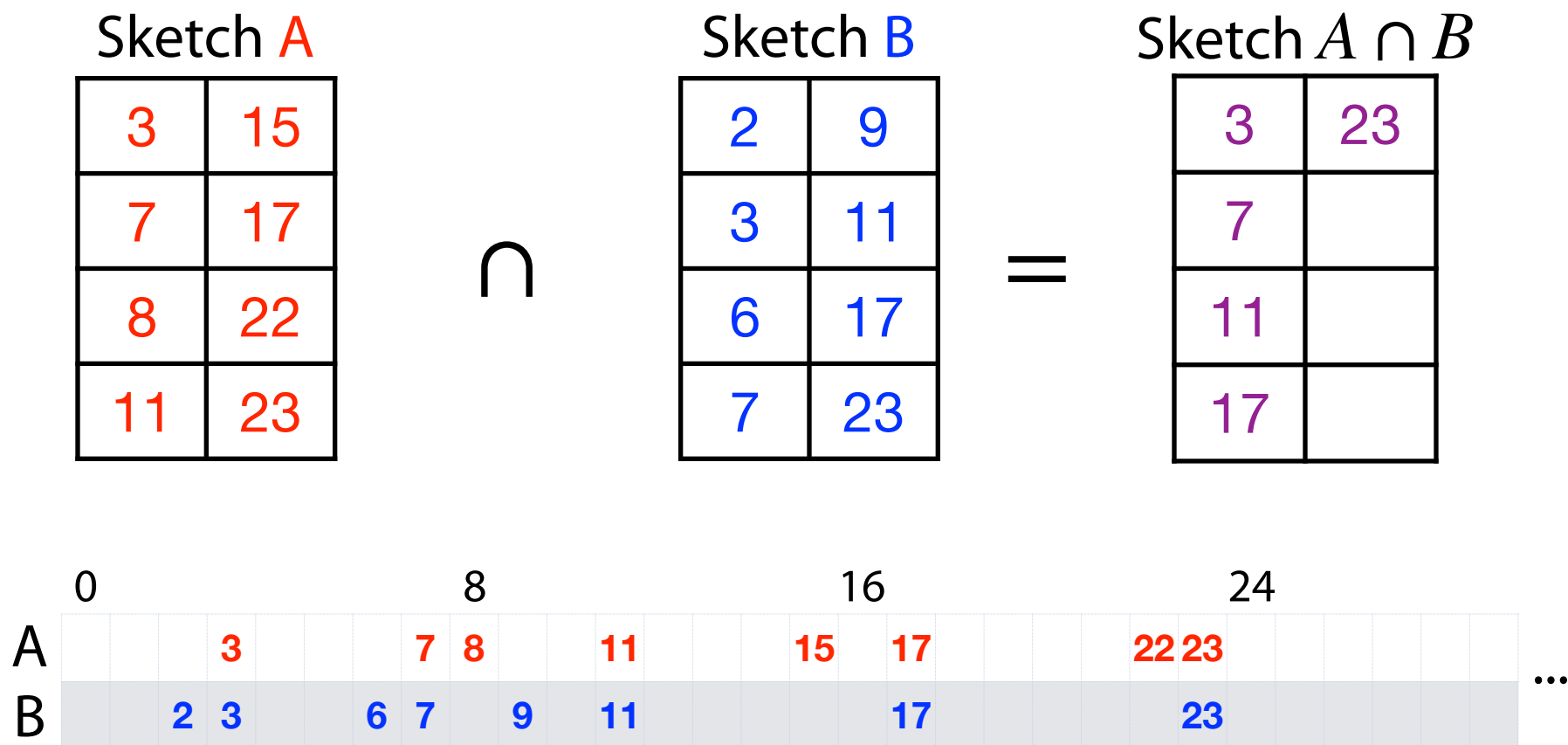
2	9
3	11
6	17
7	23

Sketch $A \cap B$

	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!



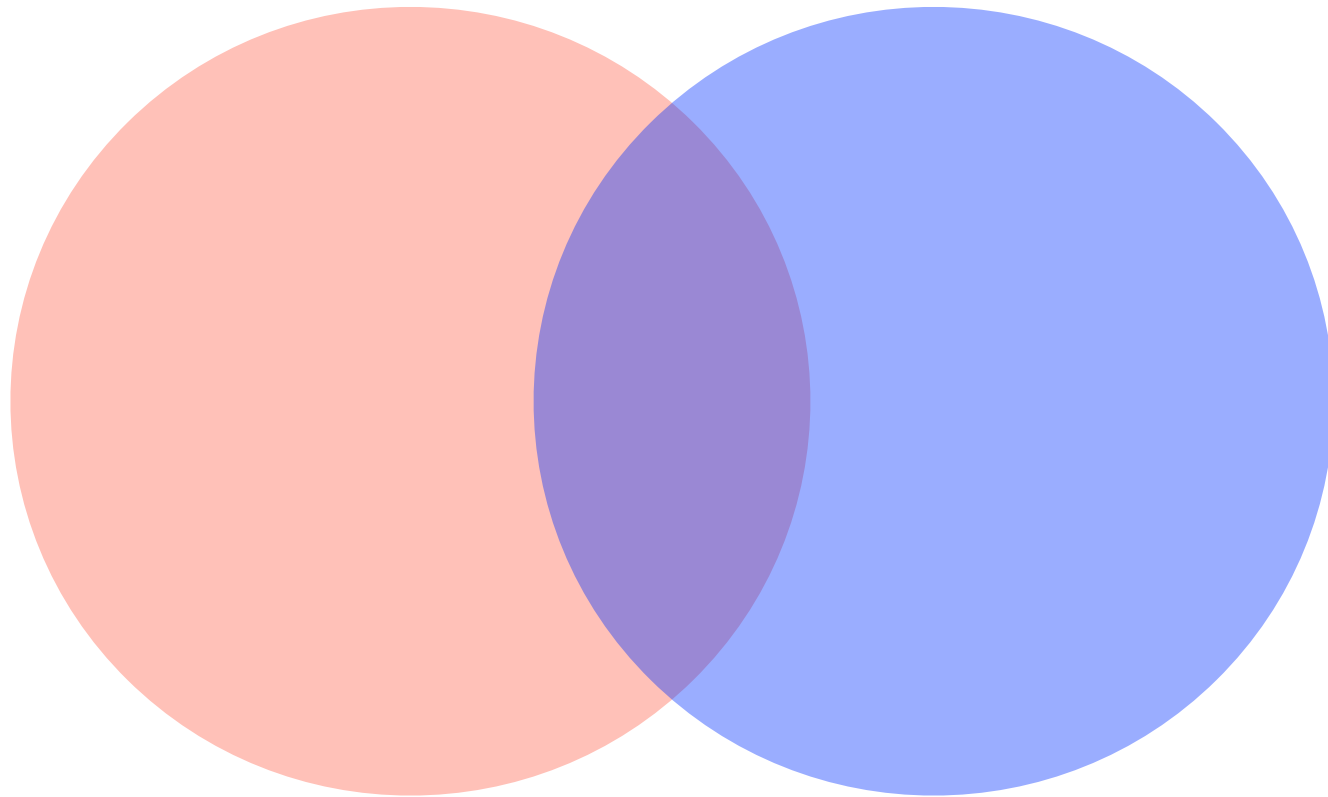
MinHash Jaccard Estimation

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

Is this enough to estimate the Jaccard?

Inclusion-Exclusion Principle

$$|A \cap B| =$$



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

$$= \frac{(800/23 - 1) + (800/23 - 1) - (800/15 - 1)}{800/15 - 1}$$

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

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

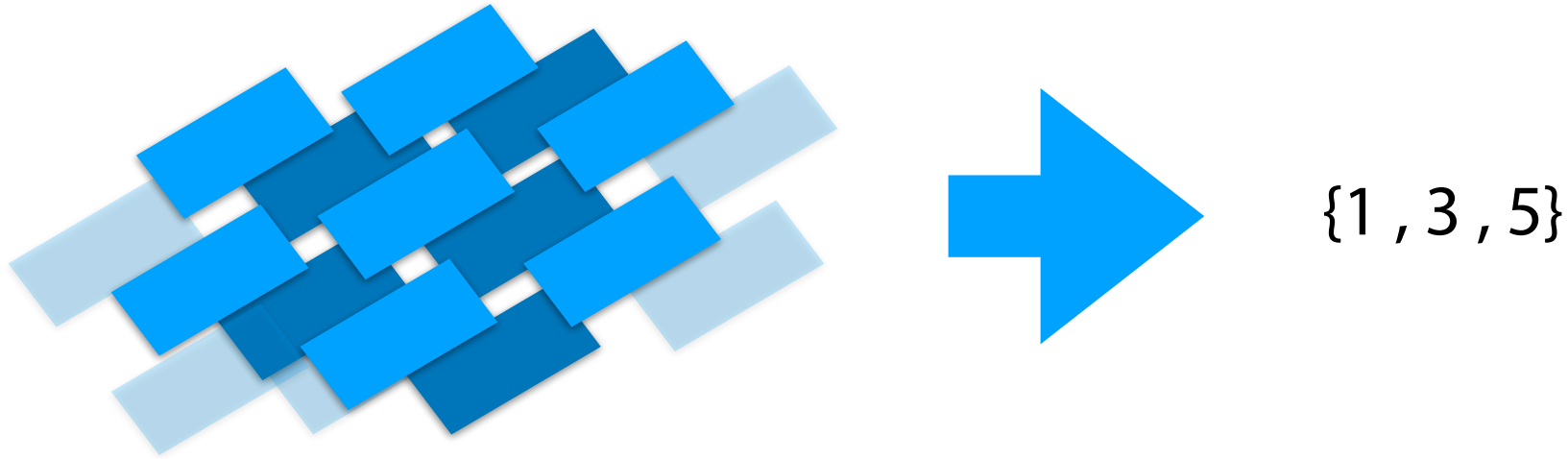
Intersection

Union

MinHash Sketch



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



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

1.

2.

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**

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

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

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)

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 **Minimum Hashes** (**MinHash**)

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

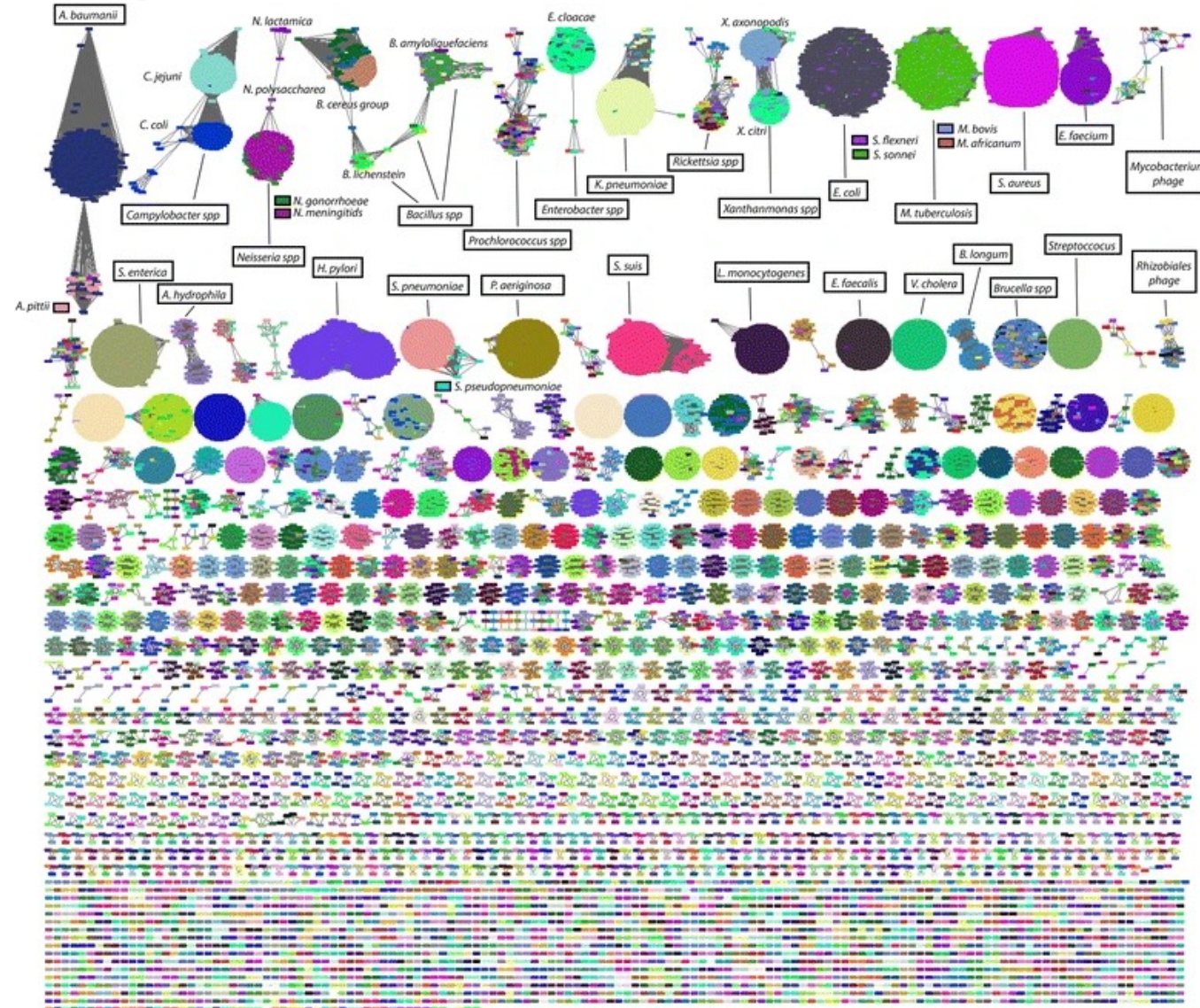
[5, 1, 2, 15]
 Sketch (S_1)

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

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

S_1 : CATTGGACCGACCAG
 | | | | |
 S_2 : GCAGTACCGATCGT

MinHash in practice



Mash: fast genome and metagenome distance estimation using MinHash

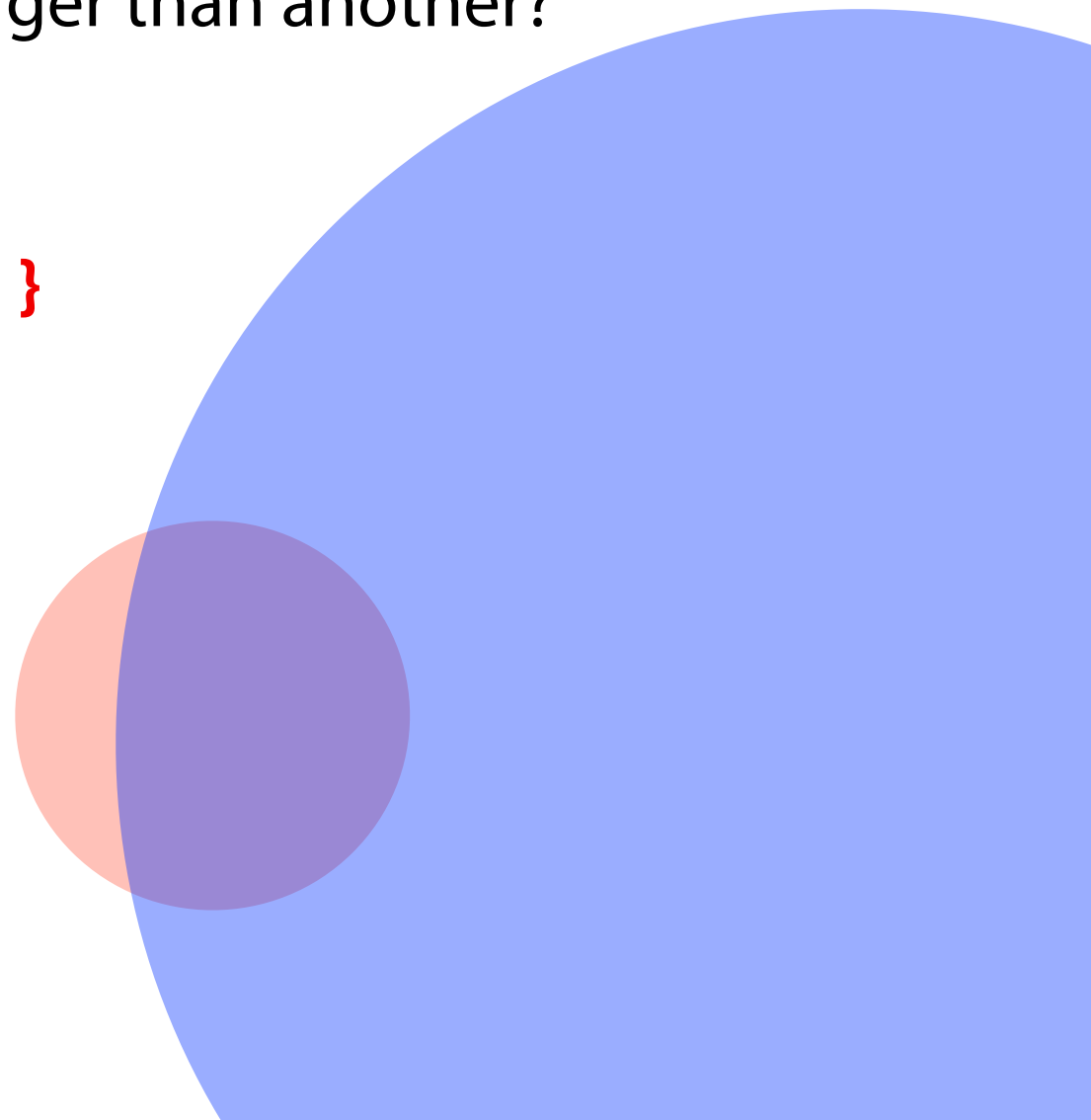
Ondov et al (2016) *Genome Biology*

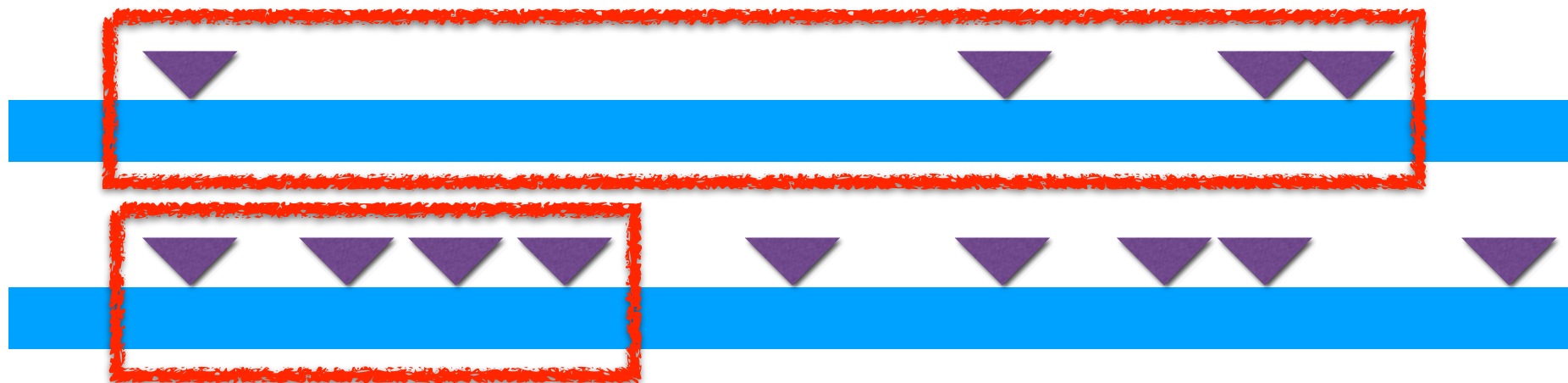
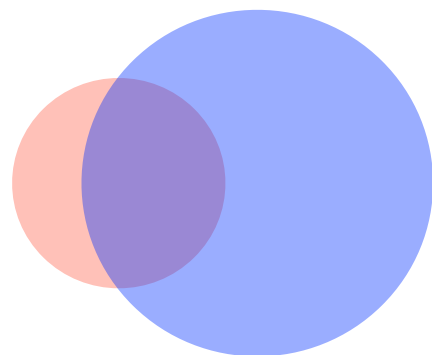
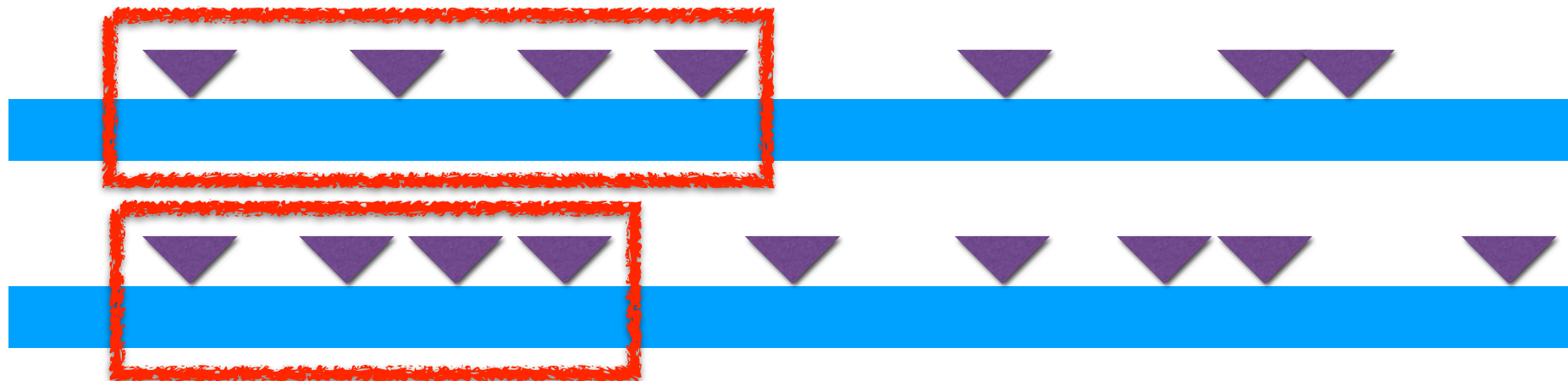
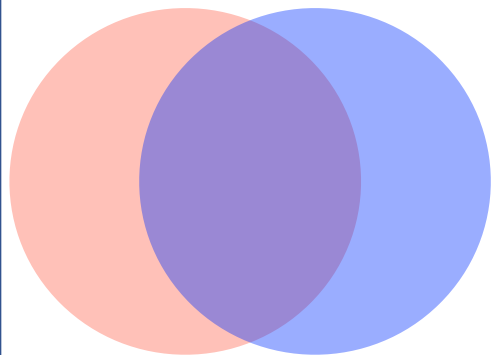
Alternative MinHash Sketch Approaches

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

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

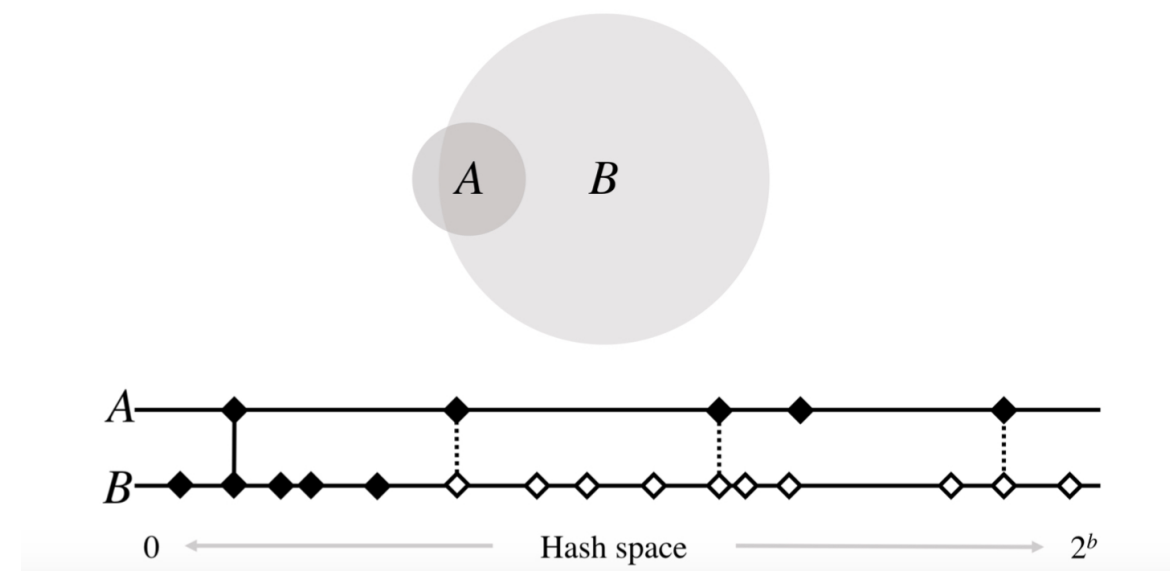
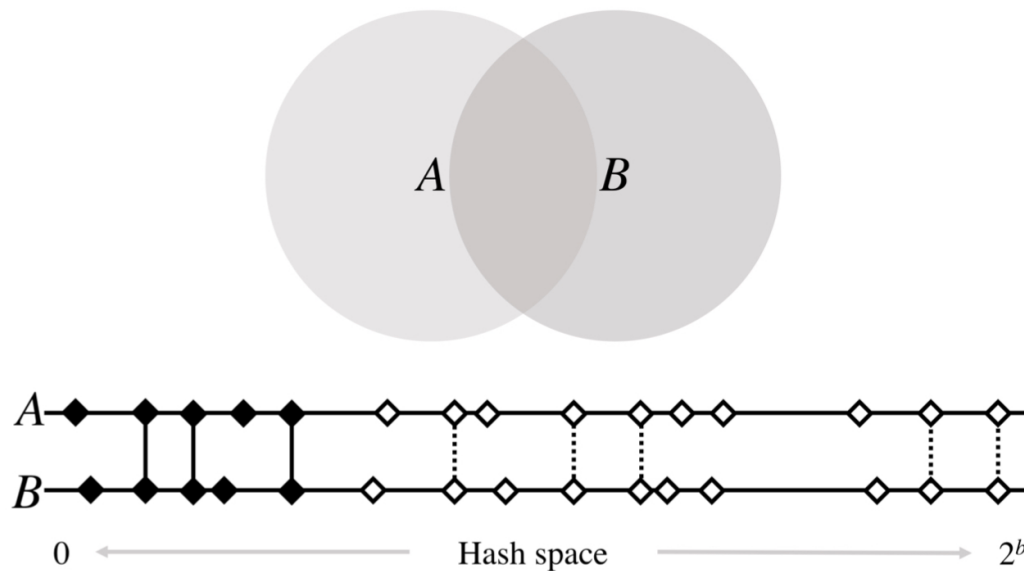
$$S_2 = \{ 1, 2, 3, 4, 5, 6, 7, \dots 59, 82, 101, \dots \}$$





Alternative MinHash sketches

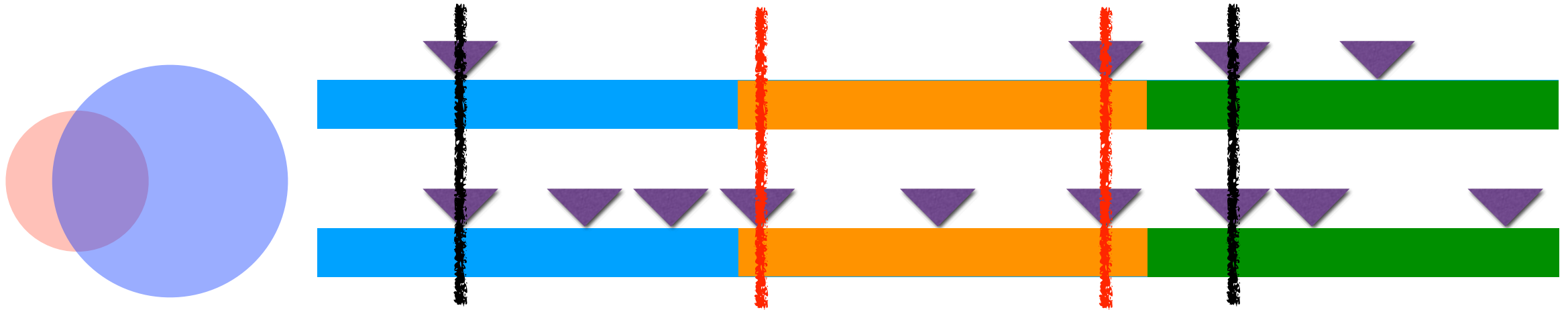
Bottom-k minhash has low accuracy if the cardinality of sets are skewed



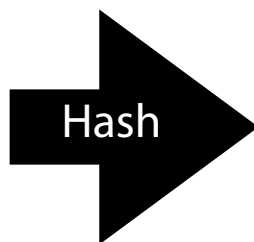
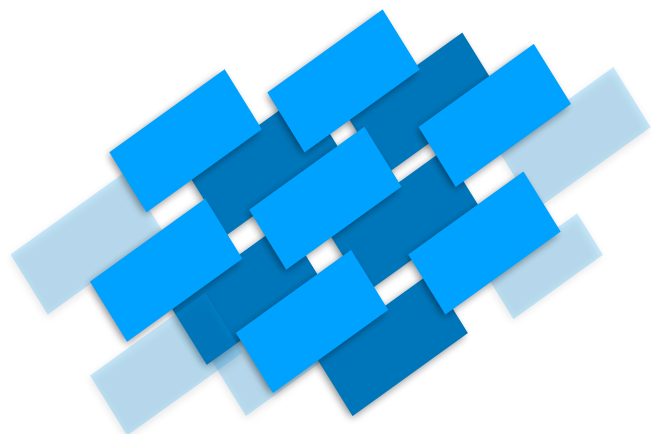
Ondov, Brian D., Gabriel J. Starrett, Anna Sappington, Aleksandra Kostic, Sergey Koren, Christopher B. Buck, and Adam M. Phillippy. **Mash Screen: High-throughput sequence containment estimation for genome discovery.** *Genome biology* 20.1 (2019): 1-13.

Alternative MinHash Sketch Approaches

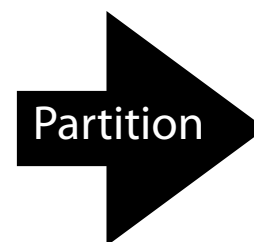
If there is a large cardinality difference, **use k-partitions!**



K-Partition Minhash



1010110101
0001111010
1101101011
1011010110
0101100000
0010001101



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