On the Resemblance and Containment of Genomes

Kim Lab Journal Lab

Mash Screen: high-throughput sequence containment estimation for genome discovery

Brian D. Ondov et al. (Nov. 2019)

On the resemblance and containment of documents

1998, IEEE

Andrei Z. Broder
DIGITAL Systems Research Center
130 Lytton Avenue, Palo Alto, CA 94301, USA
broder@pa.dec.com

Ondov *et al. Genome Biology* (2016) 17:132
DOI 10.1186/s13059-016-0997-x

Genome Biology

SOFTWARE Open Access

Mash: fast genome and metagenome distance estimation using MinHash

Brian D. Ondov[1], Todd J. Treangen[1], Páll Melsted[2], Adam B. Mallonee[1], Nicholas H. Bergman[1], Sergey Koren[3] and Adam M. Phillippy[3*]

Ondov *et al. Genome Biology* (2019) 20:232
https://doi.org/10.1186/s13059-019-1841-x

Genome Biology

METHOD Open Access

Mash Screen: high-throughput sequence containment estimation for genome discovery

Brian D. Ondov[1,2*], Gabriel J. Starrett[3], Anna Sappington[4], Aleksandra Kostic[5], Sergey Koren[1], Christopher B. Buck[3] and Adam M. Phillippy[1]

Outlines

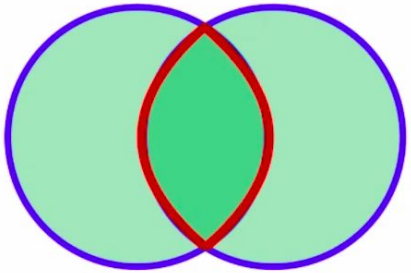
- Background: Jaccard Coefficient and Min Hash Algorithm
- Mash: Use Min Hash to estimate resemblance
- Mash Screen: Updated version of Mash to estimate containment

Double Counting Principle

Suppose A and B are two sets.

|A| is the **cardinality** of set A.

Then it is easy to write down the relationship between the union and intersection of A and B.



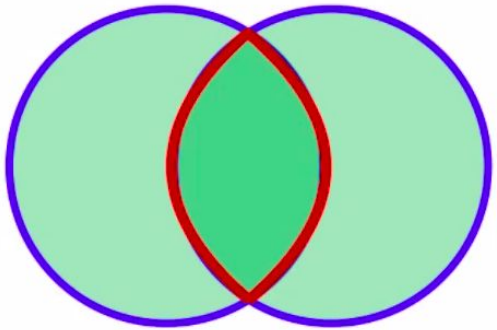
$$|A \cup B| = |A| + |B| - |A \cap B|$$

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

Resemblance can be reduced to intersection.

Suppose we have two documents (or genome), and we can divide each documents into a set of token (words, sentences, or k-mers).

Therefore the resemblance of two docs can be evaluated by **the relative size of set intersection**, in an fancy way, Jaccard coefficient.



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

J is the *Jaccard coefficient*

MinHash is used to estimate Jaccard coefficient

Finding intersection of A and B is a computational heavy task because of the large cardinalities of those two sets.

Sample from the sets and estimate the Jaccard coefficient.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \approx \frac{|S(A \cup B) \cap S(A) \cap S(B)|}{|S(A \cup B)|}$$

How to sample from a set of k-mer?

- Use Hash function to assign a random number to each k-mer
- Select |S| k-mers with the smallest hash value from set A as a **sketch**
- The Sketch is a random sample and this is MinHash Method!

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \approx \frac{|S(A \cup B) \cap S(A) \cap S(B)|}{|S(A \cup B)|}$$

Example

- The sequences of two datasets are decomposed into their constituent k-mers
- Each k-mer is passed through a hash function h to obtain a hash

J(A, B) = shared hash / union hash = 1 / 5

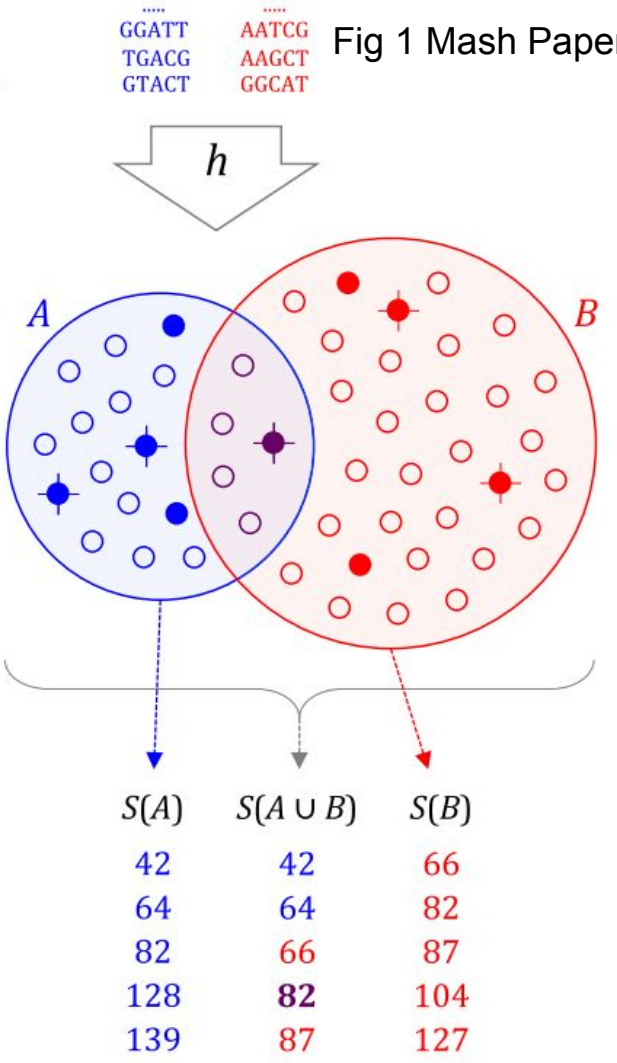


Fig 1 Mash Paper

Thus sketches composing just a few hundred values can be used to approximate the similarity of arbitrarily large dataset.

The error bound of J is $\varepsilon = O(\frac{1}{\sqrt{s}})$, which is independent to input A or B.

Mash p value

The probability of a given k-mer **K** appearing in a random genome **X** of size **n**

$$P(K \in X) = 1 - \left(1 - |\Sigma|^{-k}\right)^{n} \qquad \Sigma = \{A,C,G,T\}$$

The cardinality of |X| can be estimated by the max hash in Sketch:

$$n = \frac{s}{v/2^{b}}$$

The expected Jaccard index r between two genome X and Y is

$$r = \frac{P(K \in X)P(K \in Y)}{P(K \in X) + P(K \in Y) - P(K \in X)P(K \in Y)}$$

$$m = |X \cup Y| = |X| + |Y| - w$$

Mash p value

The probability **p** of observing **x** or more matches between sketches of two genomes can be computed by the **hypergeometric** distribution.

$$p(x; s; w; m) = 1 - \sum_{i=0}^{x-1} \frac{\binom{w}{i}\binom{m-w}{s-i}}{\binom{m}{s}}$$

The population size **m** is typically large.
The sketch size **s** << **m**.

Therefore the hypergeometric can be seen as a **binomial** distribution

$$p(x; s; r) = 1 - \sum_{i=0}^{x-1} \binom{s}{i} r^i (1-r)^{s-i}$$

where the expected value of r is Jaccard Index

$$r = \frac{w}{m}$$

Then we can do hypothesis testing to get p-val.

Problem of Mash

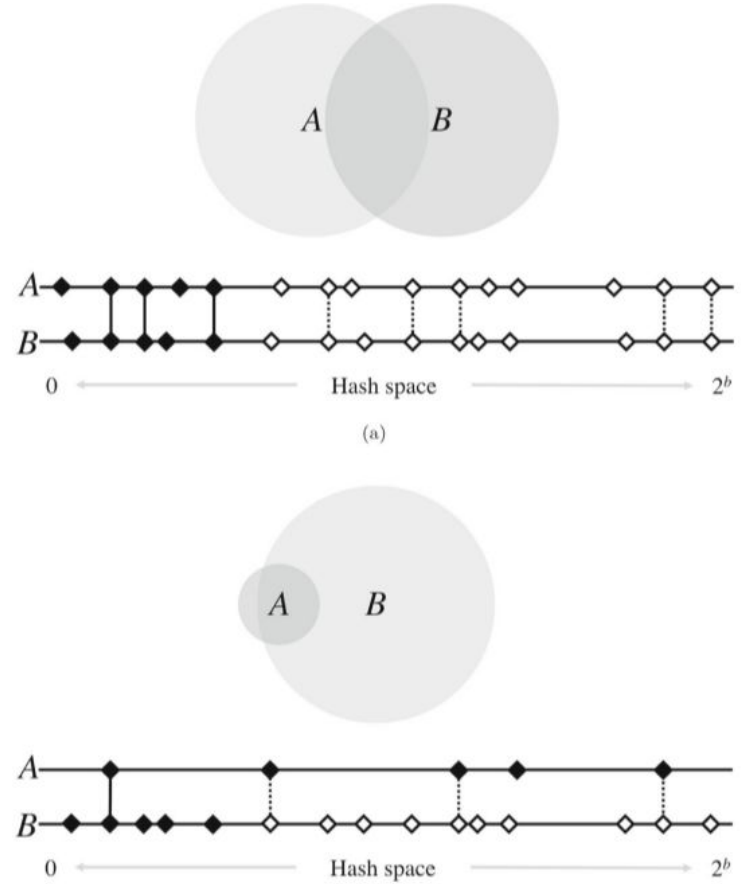
Genomes of similar sizes are well-suited for resemblance estimation.

We consider containment for sequences with difference size. Containment is asymmetric.

$$c(A,B) = \frac{|A| \cap |B|}{|B|} \quad c(B,A) = \frac{|A| \cap |B|}{|A|}$$

The sketch size is dependent with the size of genomes.

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



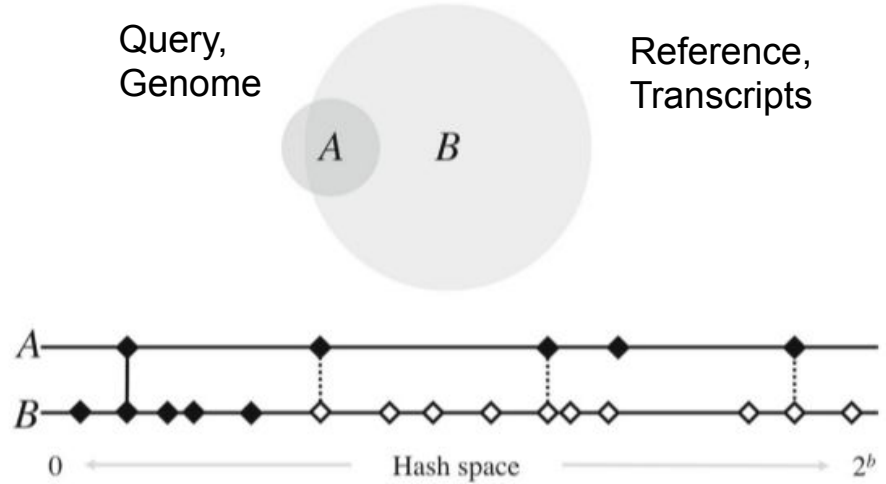
Problem of Mash

In bioinformatics, A is typically a known genome and B could be a metagenome or a set of sequencing reads.

For Mash (and similarly in BLAST), they try to identify all the items in the reference transcripts set which containing some query reference.

Therefore the output would like
“80% of the reads are matched to Mouse.“

$$\frac{|A| \cap |B|}{|B|}$$



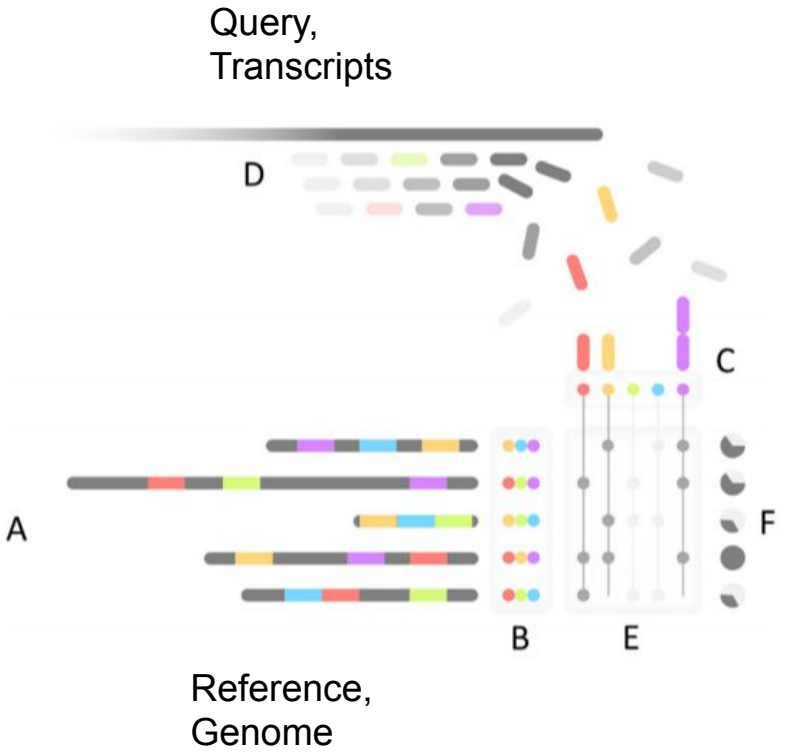
Mash Screen

Mash Screen address this problem by considering in converse:

Given a reference database of genomes, identify all those contained in some query metagenomes.

Therefore the output would like
“80% of mouse genome are in the reads.“

For each reference genome, Mash Screen computes a containment score that measures the similarity of the reference genome to a sequence within the metagenome.



Mash Screen Containment

To estimate the containment index ck of a set of reference k-mers A within some mixture B, we compare the sketch of A against all of B

$$c_k(a,b) \approx \frac{|S(A) \cap \pi(B)|}{|S(A)|}$$

We use Mash containment score (Mc(a, b))to estimate the identity of a genome and its assumed counterpart in a mixture.

$$M_c(a,b) = c_k(a,b)^{\frac{1}{k}}$$

Output from Mash

```
mash-Linux64-v2.2/mash screen -w -p 10 \
    ref-msh/RefSeq88n.msh \
    fastq/trim/unaligned_ctrl10.fq \
    > result/trim-88/unaligned_ctrl10.tab &
```

ctrl10				
identity	shared-hashes	median-multiplicity	p-value	query-ID
1	1000/1000	46	0	refseq-NZ-1095697-PRJNA184771-SAMN01055189-NZ_ANRC-.-Neisseria_meningitidis_97021.fna
0.999184	983/1000	970	0	refseq-NC-10092-PRJNA13767-.-.-.-Mus_musculus_domesticus.fna
0.993985	881/1000	22	0	refseq-NR-565-PRJNA33175-.-.-.-Escherichia_hermannii.fna
0.993931	880/1000	1377	0	refseq-NC-10116-PRJNA12455-.-.-.-Rattus_norvegicus.fna
0.985637	738/1000	25	0	refseq-NG-43215-PRJNA39195-.-.-.-Ramularia_endophylla.fna
0.977343	618/1000	18	0	refseq-NG-43215-PRJNA51803-.-.-.-Ramularia_endophylla.fna
0.972767	560/1000	3	0	refseq-NR-202950-PRJNA33175-.-.-.-Acinetobacter_baylyi.fna
0.971849	549/1000	15	0	refseq-NG-90371-PRJNA188943-.-.-.-Salmonella_enterica_subsp._enterica_serovar_Typhimurium.fna
0.965001	398/841	1	0	refseq-NG-562-.-.-.-pEPEC105-Escherichia_coli.fna
0.958528	242/589	3	0	refseq-NR-60171-PRJNA177353-.-.-.-Penicillium_verrucosum.fna

Resources

- MinHash Origin Paper [Link]
- YouTube Video about MinHash [Link]
- YouTube Video about Murmur Hash Function [Link]