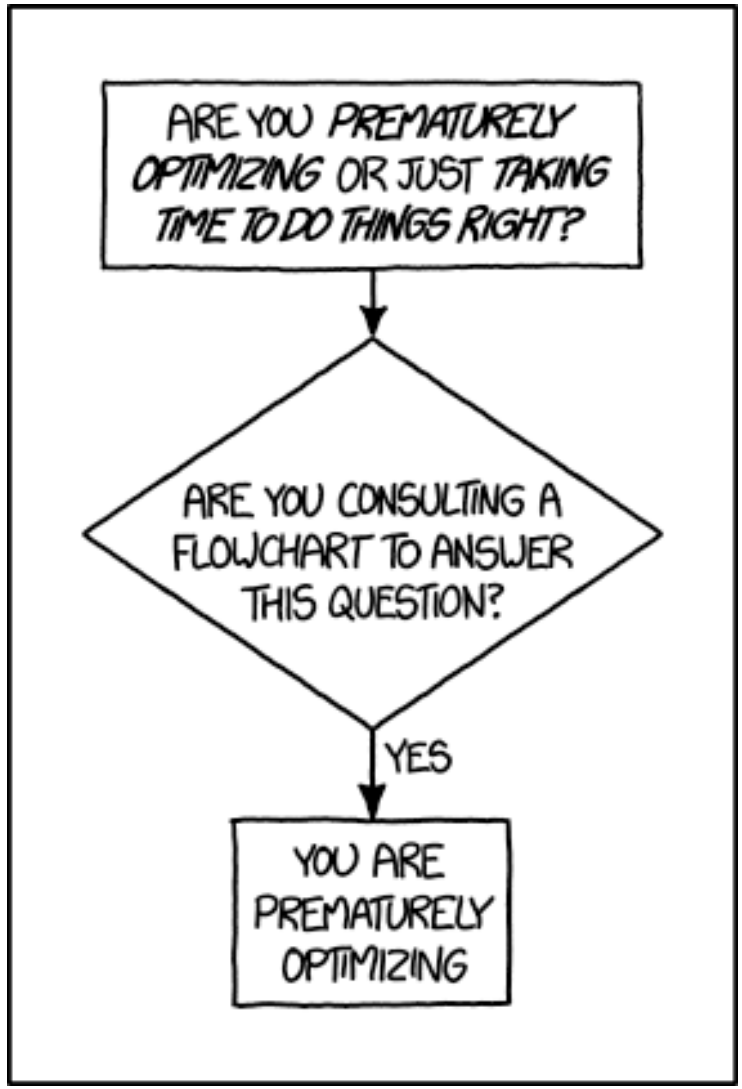


https://xkcd.com/1691/

Mash: fast genome and metagenome distance estimation

Don L. Armstrong

Institute for Genomic Biology, Computing Genomes for Reproductive Health, University of Illinois, Urbana-Champaign

June 23, 2016

Code and slides are here:

http://dla2.us/p/mashminhash2016

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*]

Abstract

Mash extends the MinHash dimensionality-reduction technique to include a pairwise mutation distance and *P* value significance test, enabling the efficient clustering and search of massive sequence collections. Mash reduces large sequences and sequence sets to small, representative sketches, from which global mutation distances can be rapidly estimated. We demonstrate several use cases, including the clustering of all 54,118 NCBI RefSeq genomes in 33 CPU h; real-time database search using assembled or unassembled Illumina, Pacific Biosciences, and Oxford Nanopore data; and the scalable clustering of hundreds of metagenomic samples by composition. Mash is freely released under a BSD license (https://github.com/marbl/mash).

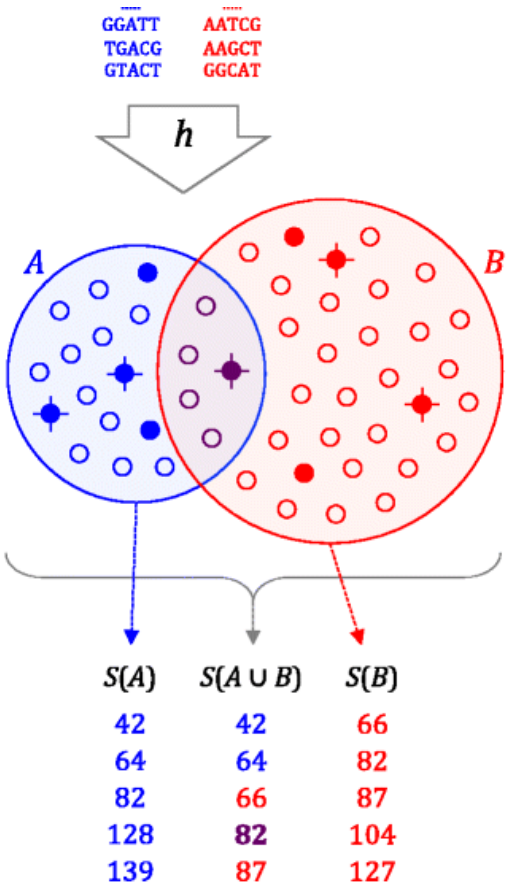
Keywords: Comparative genomics, Genomic distance, Alignment, Sequencing, Nanopore, Metagenomics

Background

When BLAST was first published in 1990 [1], there were less than 50 million bases of nucleotide sequence in the any problem where an approximate, global distance is acceptable, e.g. to triage and cluster sequence data, assign species labels, build large guide trees, identify

The Problem

MinHash Algorithm

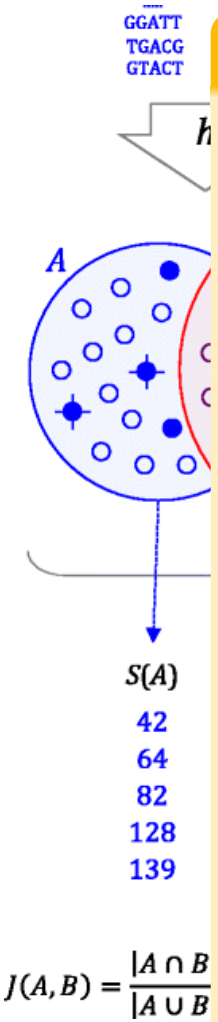


$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \approx \frac{|S(A \cup B) \cap S(A) \cap S(B)|}{|S(A \cup B)|}$$

- Decompose dataset into k-mers
- Hash k-mers (32/64bit)
- Estimate Jaccard Index $J(A, B)$

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

MinHash Algorithm

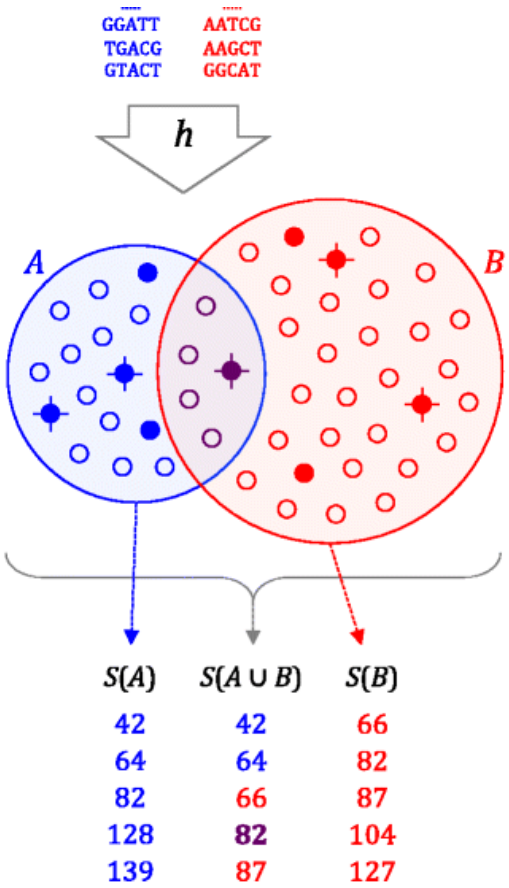


What about strandedness

- Take lexically lowest sequence
  - Given 7-mers 5'-ACTGCAC-3' and its reverse complement, 5'-GTGCAGT-3'
  - A G
  - Use ACTGCAC
- Because $S(A \cup B)$ is a random sample of $A \cup B$ the fraction of elements in $S(A \cup B)$ which are shared by $S(A)$ and $S(B)$ is an unbiased estimate of $J(A, B)$
- $J(A, B) \approx \frac{|S(A \cup B) \cap S(A) \cap (B)}{|S(A \cup B)|}$

nto k-mers
it)
ex $J(A, B)$

MinHash Algorithm

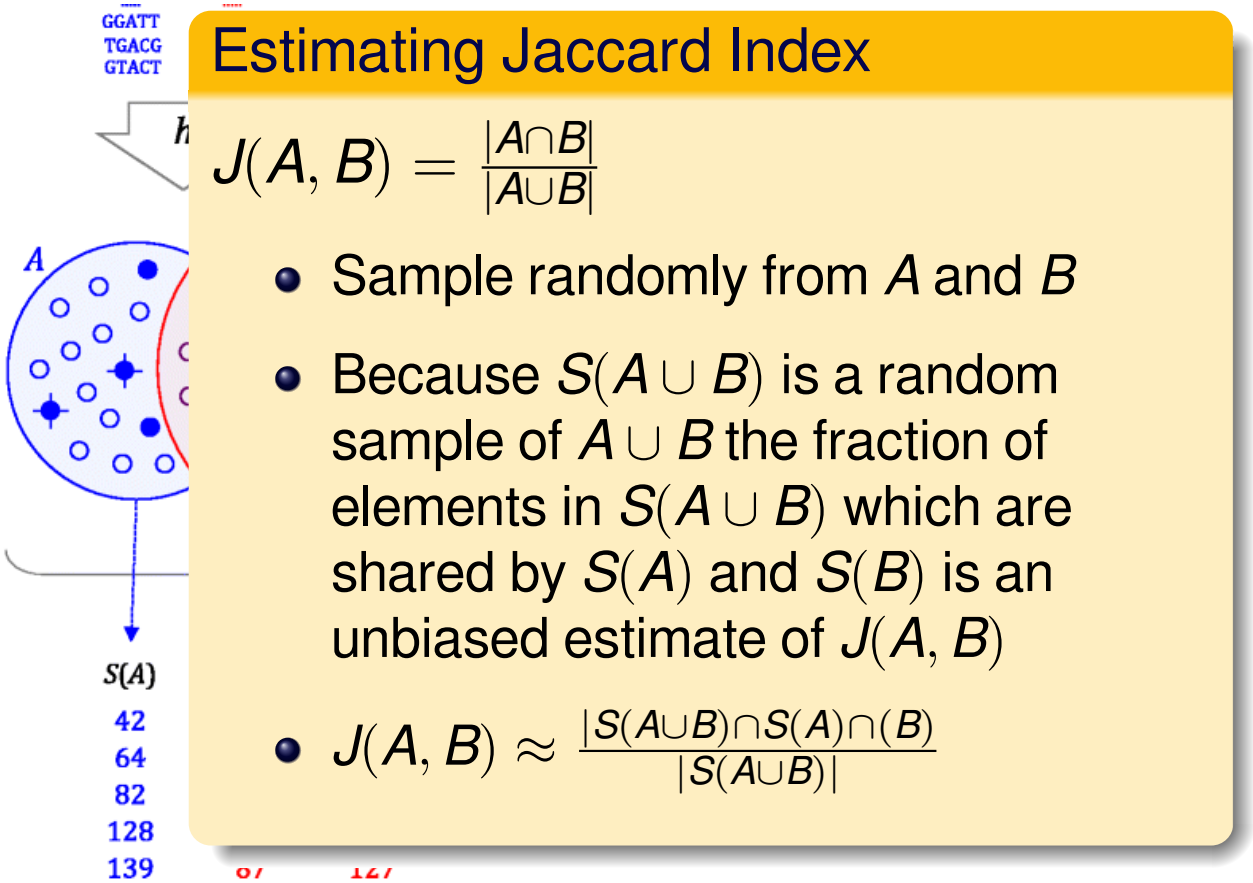


$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \approx \frac{|S(A \cup B) \cap S(A) \cap S(B)|}{|S(A \cup B)|}$$

- Decompose dataset into k-mers
- Hash k-mers (32/64bit)
- Estimate Jaccard Index $J(A, B)$

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

MinHash Algorithm



Estimating Jaccard Index

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

- Sample randomly from A and B
- Because $S(A \cup B)$ is a random sample of $A \cup B$ the fraction of elements in $S(A \cup B)$ which are shared by $S(A)$ and $S(B)$ is an unbiased estimate of $J(A,B)$
- $J(A,B) \approx \frac{|S(A \cup B) \cap S(A) \cap (B)}{|S(A \cup B)|}$

nto k-mers

it)

ex $J(A,B)$

y sample?

n!

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

Hash functions and their properties

- A function which can map arbitrary sized data to output of fixed size
- They're used everywhere: caches, duplicate filtering, cryptography, finding similar substrings, finding similar records, etc.
- Good hash functions are deterministic, uniform, and rarely produce collisions.
- Cryptographic hash functions are non-invertible (SHA-512, etc.)
- Other hash functions may have continuity (IE, AAAB and AAAC will hash to nearby values)
- for example, the MD5 of "DON" is 78ed35b3eb1f21c6179d824610d6d5c9 while "DO" is 194199909f0994cd6244c4442642f3ff

MurmurHash3 – properties

- Simple
- Good distribution
- Good collision resistance
- Good avalanche properties (single bit in the input changes the output significantly).

MurmurHash3 – properties

- Simple
- Good distribution
- Good collision resistance
- Good avalanche properties (single bit in the input changes the output significantly).
  - 0 -> 25dd8f33676144d5e42427eff1c8546724d63d15
  - 1 -> 6c30934a0ea2c0473d37b6d8bb5b955b435a8bc1
  - 2 -> 315a5aa84aa6cfa4f3fb4b652a596770be0365e8

MurmurHash3 – properties

- Simple
- Good distribution
- Good collision resistance
- Good avalanche properties (single bit in the input changes the output significantly).
- Fast
- Key questions:
  1. Does it map DNA and protein sequences uniformly into output?
  2. Does the non-random input distribution of DNA/protein sequences expose pathologic behavior of the hash function?
- Partially answered by the smhasher test suite which exposed issues with MurmurHash2 (which is ideally the DieHarder of hash testing.)

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Why is this quick

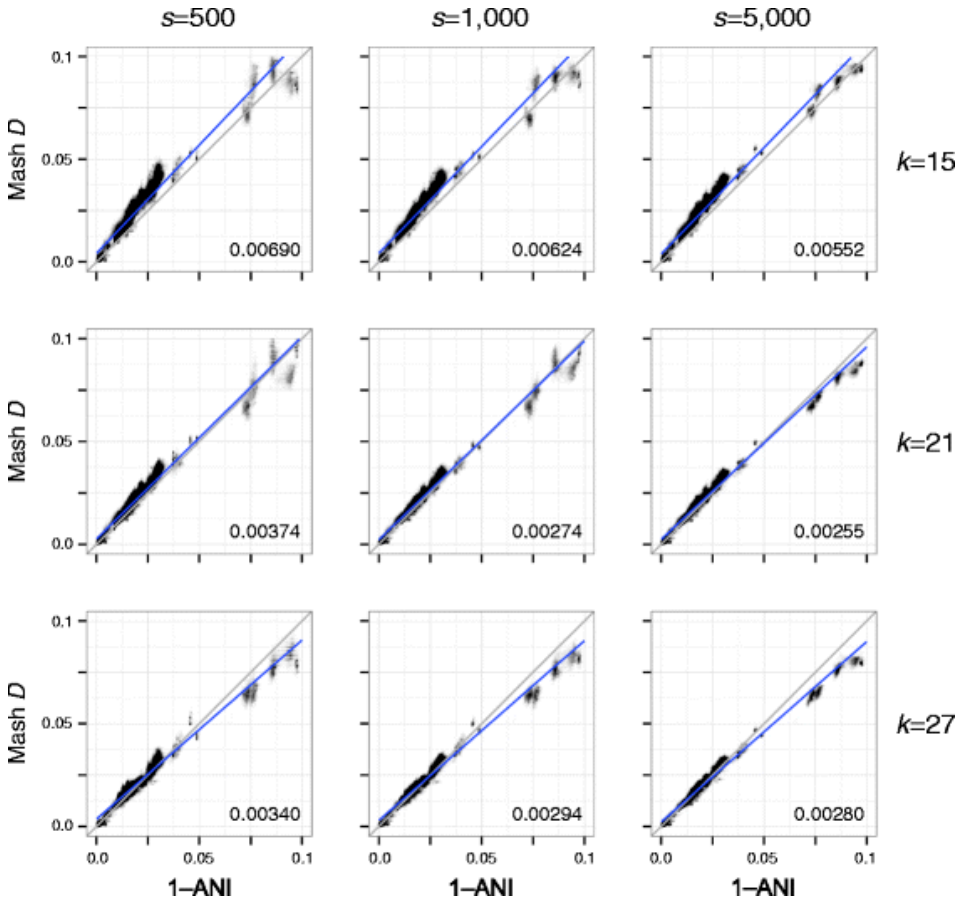
- Hashing is relatively fast
- Only read the data once
- Can be trivially parallized
- Only need to keep the *k* lowest hash values for each dataset
- Much faster than string alignment

Table 2 Mash runtime and output size for all-pairs RefSeq computation using various sketch and k-mer sizes

	k = 16				k = 21			
Sketch size	Sketch (CPU h)	Dist (CPU h)	Size (Mb)	gzip (Mb)	Sketch (CPU h)	Dist (CPU h)	Size (Mb)	gzip (Mb)
500	26.4	8.4	120.1	89.7	31.3	9.0	229.8	201.8
1000	27.7	15.9	224.9	179.7	31.3	17.4	439.2	399.6
5000	26.4	74.5	1022.5	873.8	31.6	83.6	2034.5	1924.6
10,000	26.8	146.9	1961.8	1691.1	31.7	164.0	3913.0	3696.2

Sketch: CPU h required for the Mash *sketch* operation for all 54,118 RefSeq genomes. *Dist:* CPU h required for the Mash *dist* table operation for all pairs of sketches. *Size:* combined size of the resulting sketches in megabytes. *gzip:* combined size of the resulting sketches after gzip compression

How does it stack up against ANI?



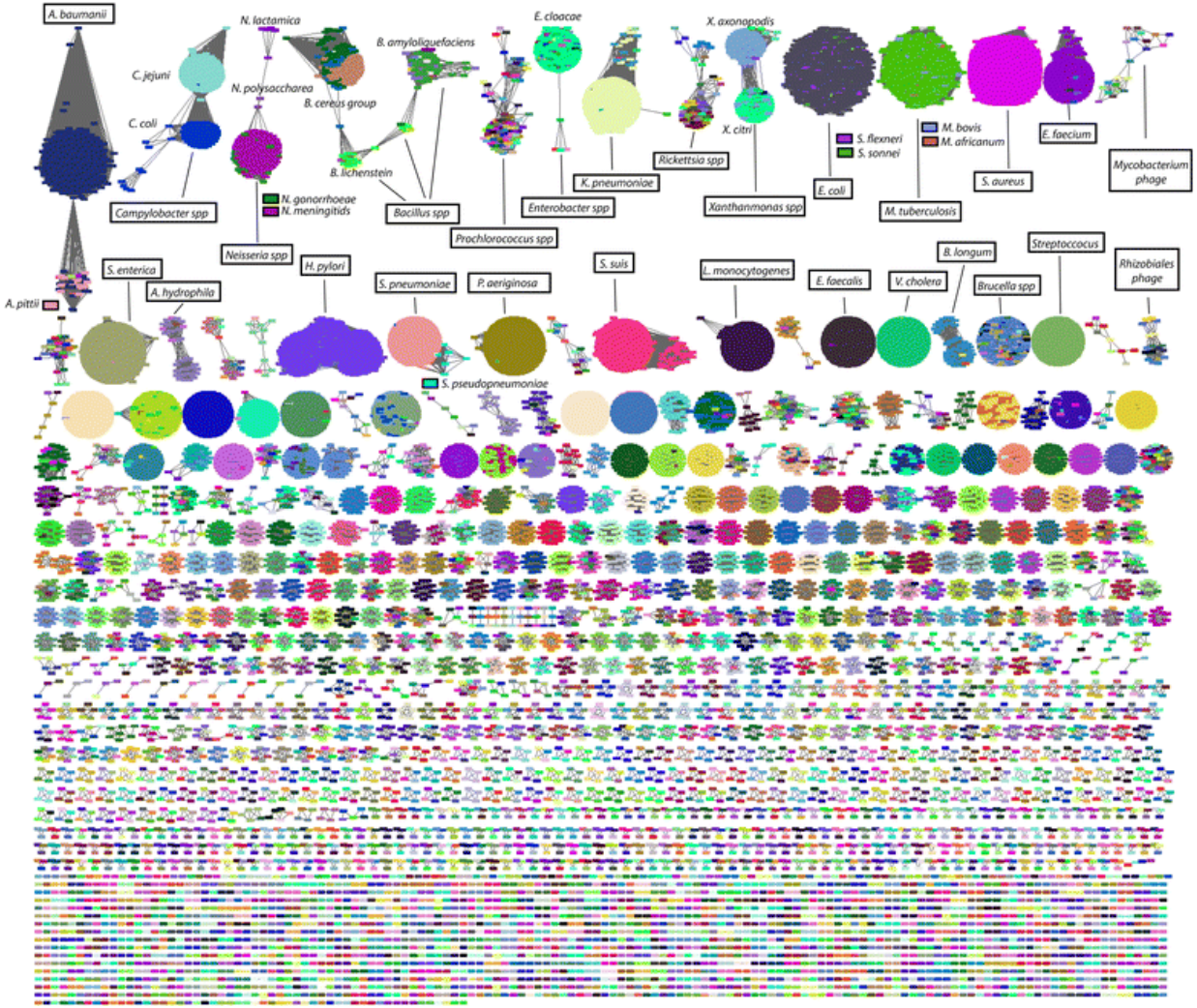
- Sketch size *s*
- k-mer size *k*
- Mash D is the mash distance
- ANI is the average nucleotide identity between genomes
- Increasing *s* improves match; *k* seems to max near 27.
- ANI only considers core genome, so at some point ANI and Mash will diverge

Mash error bounds with $k = 21$

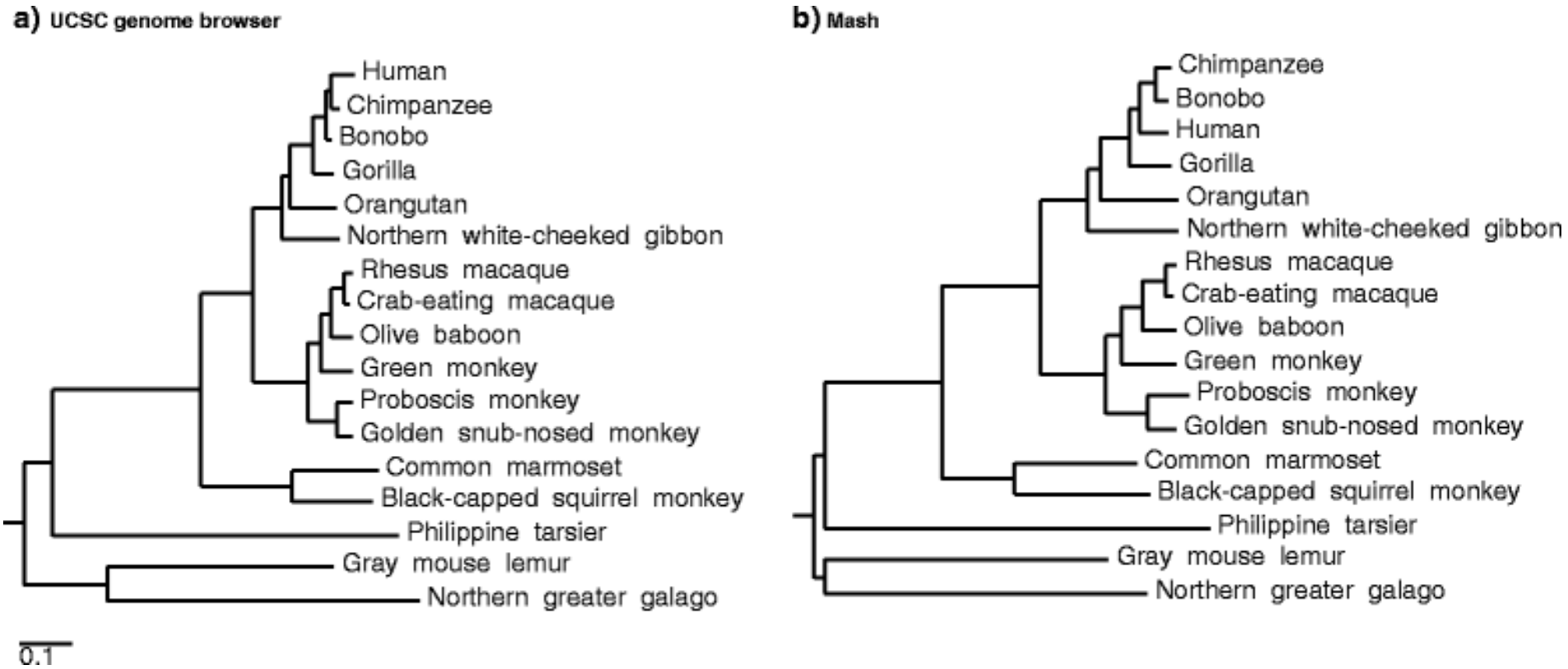
Table 1 Example Mash error bounds for a k-mer size of 21 and increasing sketch sizes

	Mash distance							
Sketch size	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
100	0.0271	0.0868	–	–	–	–	–	–
500	0.0098	0.0245	0.0473	–	–	–	–	–
1000	0.0068	0.0158	0.0323	0.0630	–	–	–	–
5000	0.0029	0.0065	0.0124	0.0235	0.0460	–	–	–
10,000	0.0020	0.0046	0.0086	0.0159	0.0300	0.0726	–	–
50,000	0.0009	0.0020	0.0037	0.0065	0.0116	0.0219	0.0396	0.0822
100,000	0.0006	0.0014	0.0026	0.0046	0.0081	0.0143	0.0250	0.0492
500,000	0.0003	0.0006	0.0011	0.0020	0.0035	0.0060	0.0105	0.0187
1,000,000	0.0002	0.0004	0.0008	0.0014	0.0024	0.0042	0.0072	0.0128

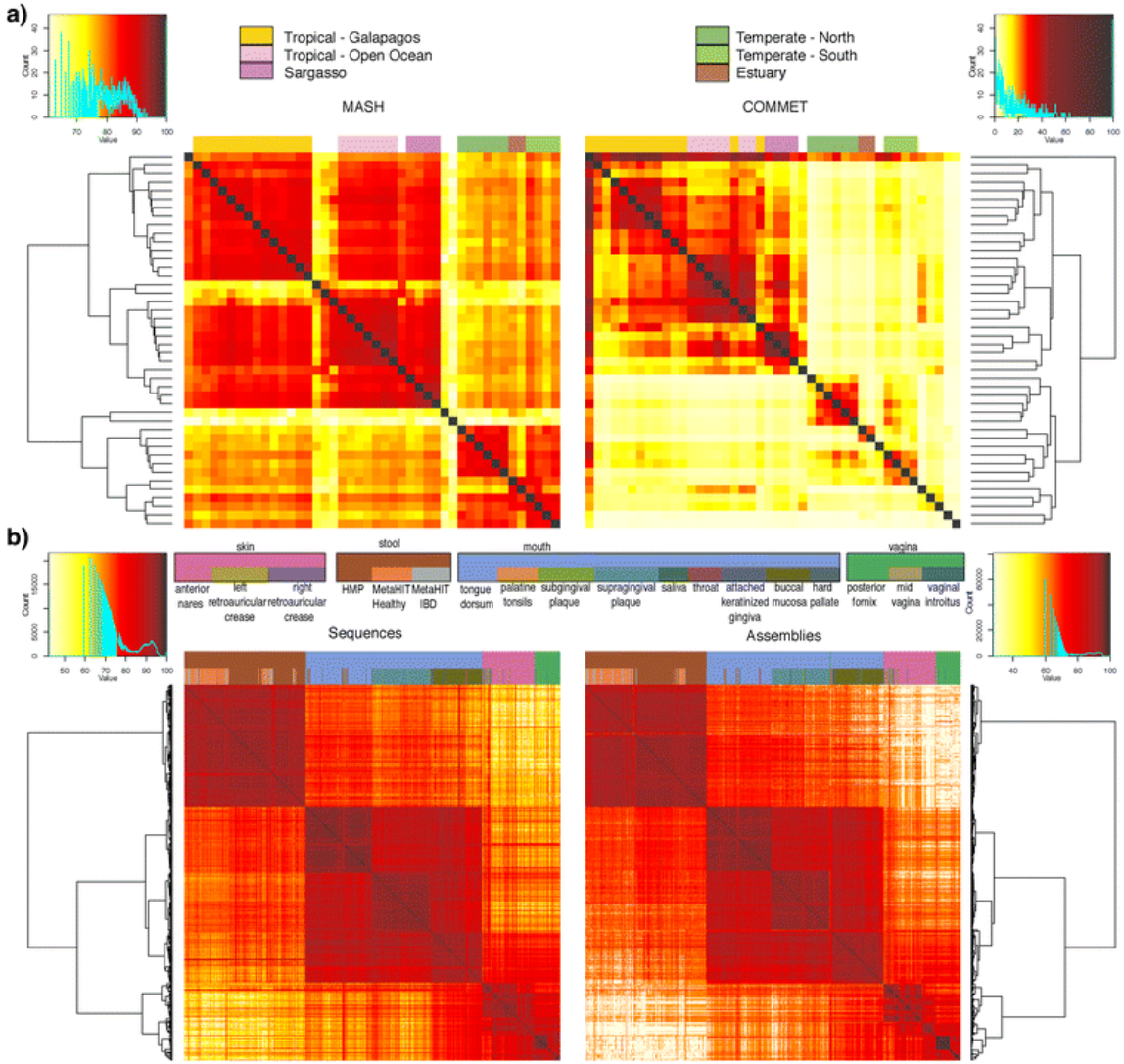
For a given sketch size and Mash distance, the Mash estimation error will be less than the given value with 0.99 probability, as calculated by the binomial inverse cumulative distribution function. Missing values indicate that the estimate is unbounded, i.e. there is a chance that no matching k-mers will be found and the Mash distance will be undefined. Plots of the upper and lower error bounds for $k = 16$ and $k = 21$ are given in Additional file 1: Figure S2



- *De novo* clustering of all RefSeq genomes
- Each node is a genome
- Nodes are connected if Mash distance ≤ 0.05 and $p \leq 10^{-10}$

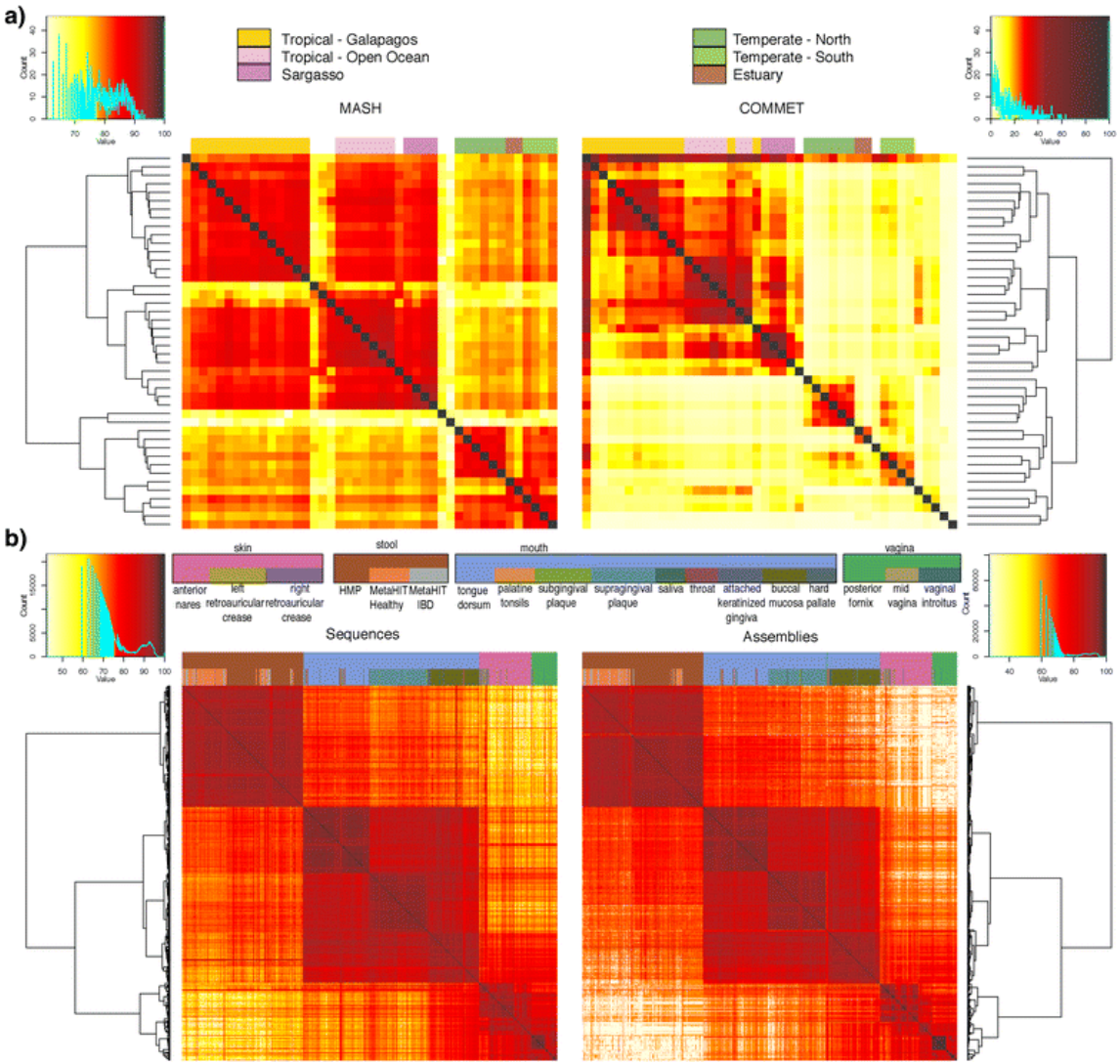


Primate species trees calculated by mash (right) in comparison to UCSC genome browser trees (left)



- Comparison between mash and COMMET using Global ocean survey clustering
- Mostly reproduces the clusters seen in the original study

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN



- Comparison of sequences and assemblies from 888 sequencing runs and 879 assemblies from MHP and MetaHit projects
- Mostly reproduces the clusters seen in the original study
- COMMET would have required 140,000 hours to run this analysis; mash did it in 25 hours or so.

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Conclusions

- Compare genomes on a massive scale which would be impossible for typical techniques
- Good for rapid triaging of sequencing data to identify outliers
- Metagenomics and identification of samples
- Caveats
  - Based on k-mer sketch; possibility of batch effects
  - Phylogeny reconstructions are approximate, and do not track mutation rates, etc.
  - What about deconvolution of mixed samples? Probably need different techniques.

References