Mutual Information-based feature selection of phylo-k-mers

Nikolai Romashchenko, Benjamin Linard, Fabio Pardi, Eric Rivals

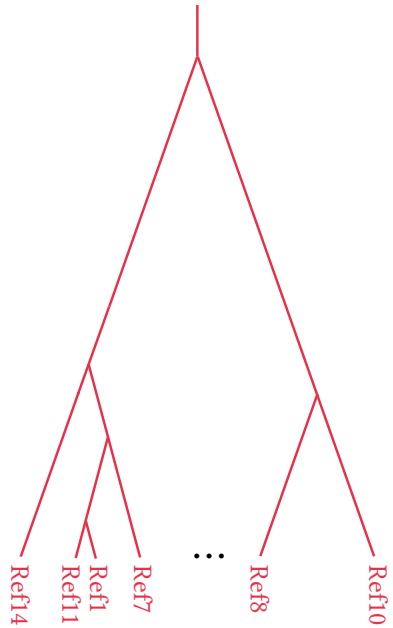
CNRS, University of Montpellier

17 November, 2022

Phylogenetic placement

```
Ref1:  C T G T C C T A T G A G A A G A T . . . .
Ref2:  C T T T T A T G T G G G G A T A A A G T G
Ref3:  C T G T T A T A C G G G A C G A A C G G C
Ref4:  C T G T T G T T A G G G A A G A A A A A A
Ref5:  C T T T T A T A T G G G A A G A T . . . .
Ref6:  C T A T C A G A A G G G A A G A A . . . .
Ref7:  C T T T G G A C A G G G A C G A A G A . .
Ref8:  C T T T T A G T A G G G A A G A A A G . .
Ref9:  C T T T T A T A T G G G A A G A T . . . .
Ref10: C T T T T T C T C G T G A A A A A G G C A
Ref11: . . . . . .
Ref12: C T T T T G T T T G G G A A T A A T C G A
Ref13: C T G T G A T T C G G G A C G A A A G G C
Ref14: C T T T T A T A T G G G A A G A T . . . .
Ref15: C T G T G A T T C G G G A C G A A A G A T
Ref16: C T T T A G G C G G G G A G T A A T G T G
Ref17: C T T T T A T A T G G G G A T A A A G T G
Ref18: C T T T A G G C G G G G A G T A A T G T G
Ref19: C T T T T G T A T G G G A T T A A A G T C
Ref20: C T T T T A T G C G G G G A T A A A G G A
Ref21: C T G T T G T T T G G T G A A G A A G G A C
Ref22: C T T T T G T A T G G G A A G A A . . . .
Ref23: C T T T T A T G C G G G G A T A A A G G A
```

```
Q1: T G T T T G G G A A T A A T C G A
Q2: C T T T T G T A T G A A G A A
Q3: C T T T T A T G C G G G G
```



The topology of the reference tree is given and remains fixed.

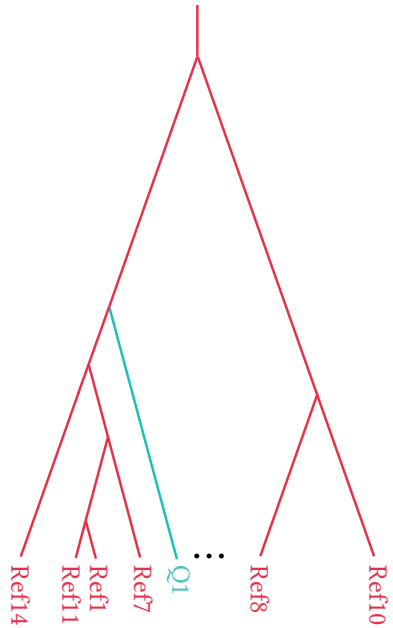
Queries are inserted one-by-one independently.

Possible with millions of queries.

Phylogenetic placement

```
Ref1:  C T G T C C T A T G A G A A G A T . . . .
Ref2:  C T T T T A T G T G G G G A T A A A G T G
Ref3:  C T G T T A T A C G G G A C G A A C G G C
Ref4:  C T G T T G T T A G G G A A G A A A A A A
Ref5:  C T T T T A T A T G G G A A G A T . . . .
Ref6:  C T A T C A G A A G G G A A G A A . . . .
Ref7:  C T T T G G A C A G G G A C G A A G A . .
Ref8:  C T T T T A G T A G G G A A G A A A G . .
Ref9:  C T T T T A T A T G G G A A G A T . . . .
Ref10: C T T T T T C T C G T G A A A A A G G C A
Ref11: . . . . . .
Ref12: C T T T T G T T T G G G A A T A A T C G A
Ref13: C T G T G A T T C G G G A C G A A A G G C
Ref14: C T T T T A T A T G G G A A G A T . . . .
Ref15: C T G T G A T T C G G G A C G A A A G A T
Ref16: C T T T A G G C G G G G A G T A A T G T G
Ref17: C T T T T A T A T G G G G A T A A A G T G
Ref18: C T T T A G G C G G G G A G T A A T G T G
Ref19: C T T T T G T A T G G G A T T A A A G T C
Ref20: C T T T T A T G C G G G G A T A A A G G A
Ref21: C T G T T G T T T G G T G A A G A A G G A C
Ref22: C T T T T G T A T G G G A A G A A . . . .
Ref23: C T T T T A T G C G G G G A T A A A G G A
```

```
Q1: T G T T T G G G A A T A A T C G A
Q2: C T T T T G T A T G A A G A A
Q3: C T T T T A T G C G G G G
```



The topology of the reference tree is given and remains fixed.

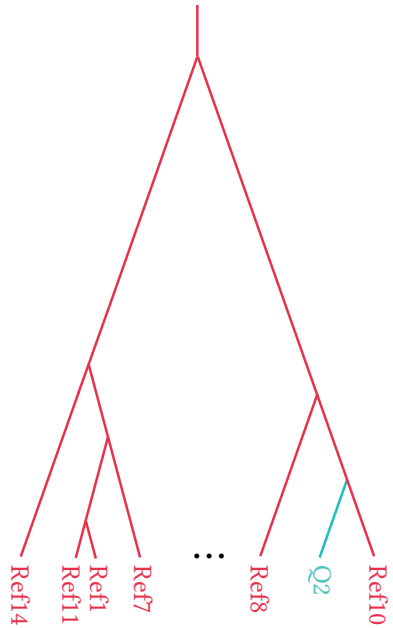
Queries are inserted one-by-one independently.

Possible with millions of queries.

Phylogenetic placement

```
Ref1:  C T G T C C T A T G A G A A G A T . . . .
Ref2:  C T T T T A T G T G G G G A T A A A G T G
Ref3:  C T G T T A T A C G G G A C G A A C G G C
Ref4:  C T G T T G T T A G G G A A G A A A A A A
Ref5:  C T T T T A T A T G G G A A G A T . . . .
Ref6:  C T A T C A G A A G G G A A G A A . . . .
Ref7:  C T T T G G A C A G G G A C G A A G A . .
Ref8:  C T T T T A G T A G G G A A G A A A G . .
Ref9:  C T T T T A T A T G G G A A G A T . . . .
Ref10: C T T T T T C T C G T G A A A A A G G C A
Ref11: . . . . . .
Ref12: C T T T T G T T T G G G A A T A A T C G A
Ref13: C T G T G A T T C G G G A C G A A A G G C
Ref14: C T T T T A T A T G G G A A G A T . . . .
Ref15: C T G T G A T T C G G G A C G A A A G A T
Ref16: C T T T A G G C G G G G A G T A A T G T G
Ref17: C T T T T A T A T G G G G A T A A A G T G
Ref18: C T T T A G G C G G G G A G T A A T G T G
Ref19: C T T T T G T A T G G G A T T A A A G T C
Ref20: C T T T T A T G C G G G G A T A A A G G A
Ref21: C T G T T G T T T G G T G A A G A A G G A C
Ref22: C T T T T G T A T G G G A A G A A . . . .
Ref23: C T T T T A T G C G G G G A T A A A G G A
```

```
Q1: T G T T T G G G A A T A A T C G A
Q2: C T T T T G T A T G A A G A A
Q3: C T T T T A T G C G G G G
```

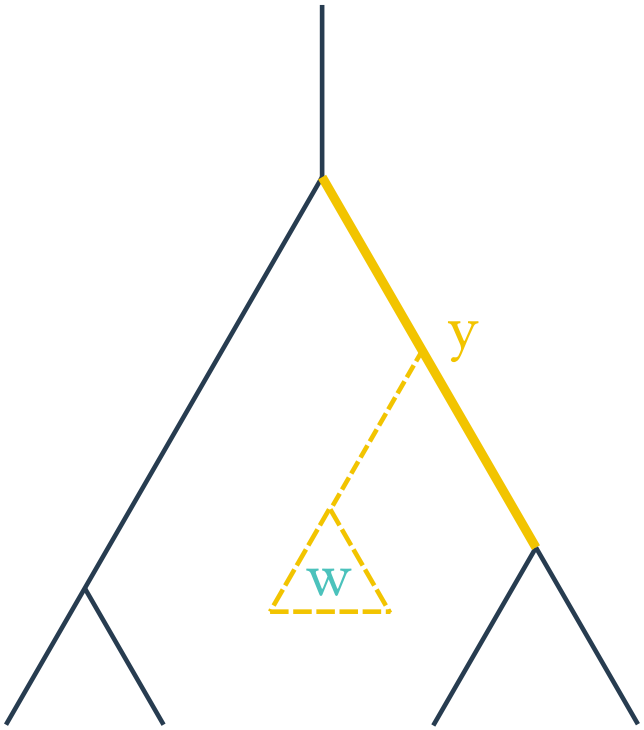


The topology of the reference tree is given and remains fixed.

Queries are inserted one-by-one independently.

Possible with millions of queries.

Phylo-k-mers

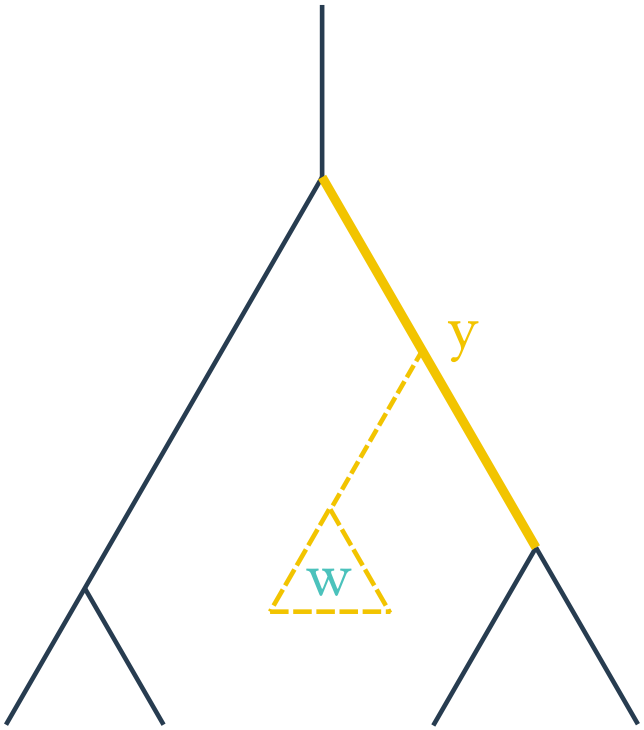


$S_y(w)$ estimates the probability of observing w in sequences diverged from y.

- w is k-mer (string of size k)
- y, a branch of the reference tree.

Phylo-k-mer is a triple
$(w, y, S_y(w))$

Phylo-k-mers



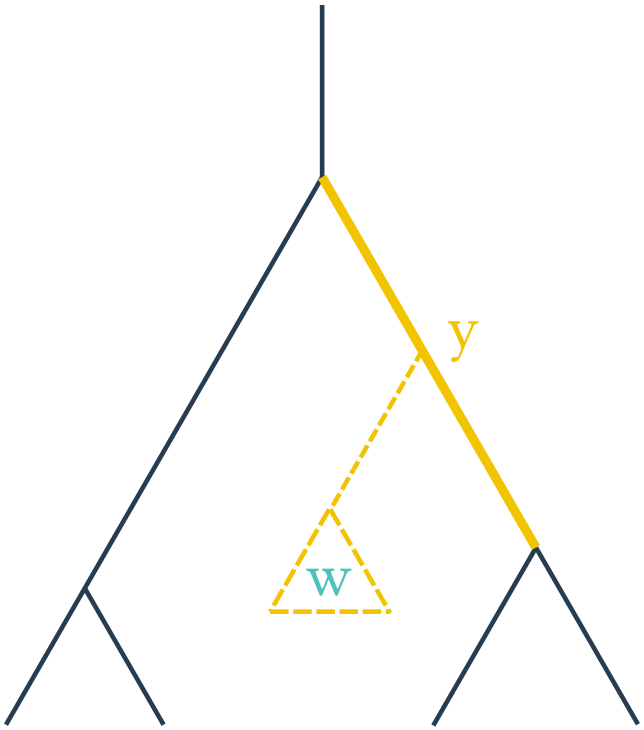
$S_y(w)$ estimates the probability of observing w in sequences diverged from y.

- w is k-mer (string of size k)
- y, a branch of the reference tree.

Phylo-k-mer is a triple $(w, y, S_y(w))$

k-mer	branch	score
AAA	y_1	0.9

Phylo-k-mers



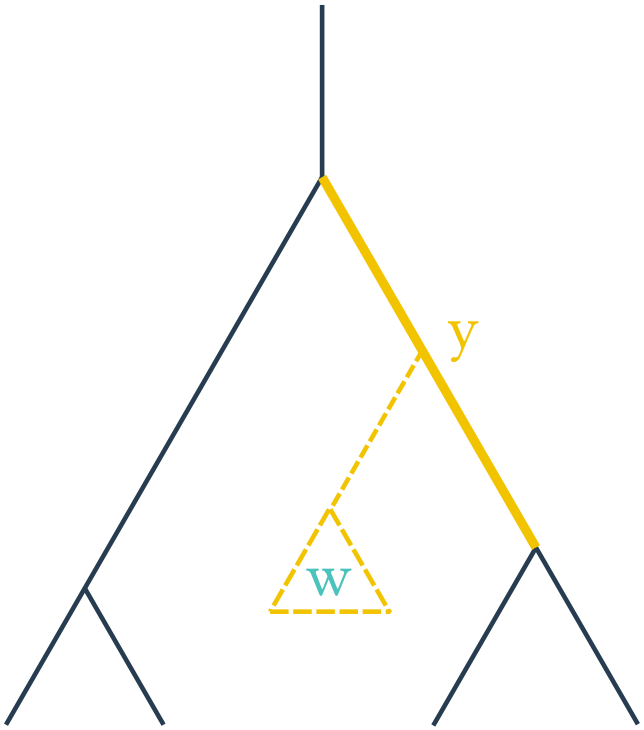
$S_y(w)$ estimates the probability of observing w in sequences diverged from y.

- w is k-mer (string of size k)
- y, a branch of the reference tree.

Phylo-k-mer is a triple $(w, y, S_y(w))$

k-mer	branch	score
AAA	y_1	0.9
	y_2	0.4

Phylo-k-mers



$S_y(w)$ estimates the probability of observing w in sequences diverged from y.

- w is k-mer (string of size k)
- y, a branch of the reference tree.

Phylo-k-mer is a triple $(w, y, S_y(w))$

k-mer	branch	score
AAA	y_1	0.9
	y_2	0.4
ATC	y_2	0.3
	...	

RAPPAS2: PLACEMENT ALGORITHM

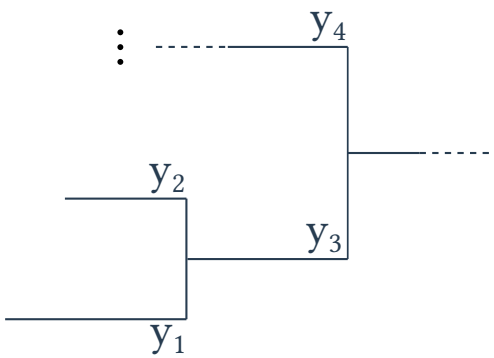
database

AAA	y_1	0.9
	...	
ATG	y_2	0.4
	y_3	0.2
	...	
TAT	y_1	0.7
	y_3	0.43

query

TATGAG..

Every k-mer of the query is searched in the database.



RAPPAS2: PLACEMENT ALGORITHM

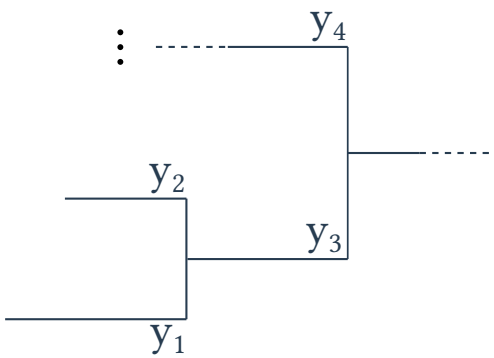
database		
AAA	y_1	0.9
	...	
ATG	y_2	0.4
	y_3	0.2
	...	
TAT	y_1	0.7
	y_3	0.43

query

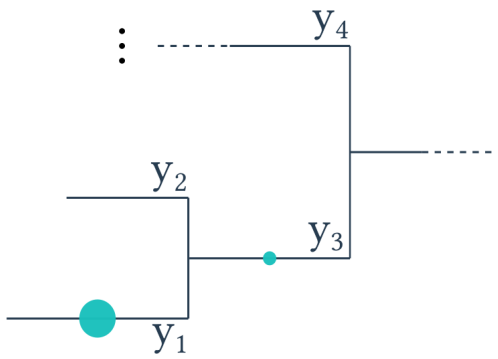
TATGAG..

TAT
ATG
TGA

Every k-mer of the query is searched in the database.

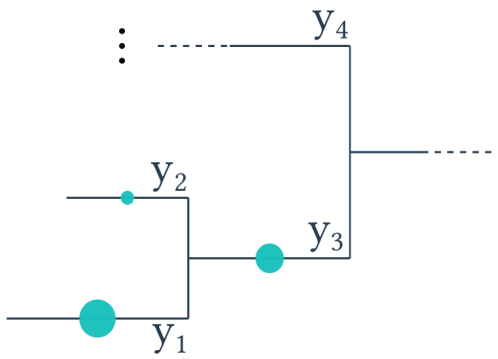


RAPPAS2: PLACEMENT ALGORITHM



Every k-mer of the query is searched in the database.

RAPPAS2: PLACEMENT ALGORITHM



Every k-mer of the query is searched in the database.

Branch y is scored with

$$\prod_w S_y(w)$$

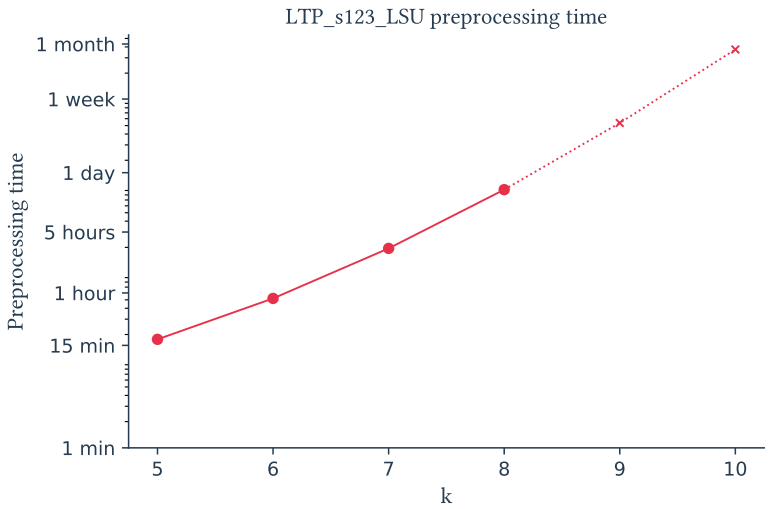
Equivalently:

$$\sum_w \log S_y(w)$$

Challenges of the method

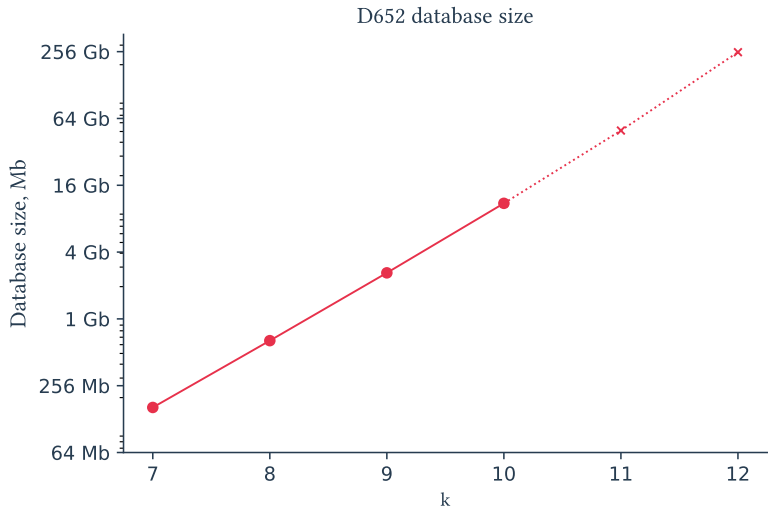
Archaea / Bacteria 23S rRNA

$1.7Kbp, \approx 14K$ taxa



Bacteria 16S rRNA (bv)

$1.7Kbp$, 652 taxa



Preprocessing is long

Databases are large

$\Longrightarrow$ can not be applied to long alignments and large trees.

Informative k-mers: intuition

ACAT

branch	score
y_1	ε
y_2	ε
y_3	0.87
y_4	0.44
y_5	ε

informative

Some *k*-mers are more *informative* for placement than others.

(better at distinguishing between branches)

Informative k-mers: intuition

ACAT

branch	score
y_1	ε
y_2	ε
y_3	0.87
y_4	0.44
y_5	ε

informative

GATA

branch	score
y_1	0.99
y_2	0.99
y_3	0.99
y_4	0.99
y_5	0.99

not informative

Some *k*-mers are more *informative* for placement than others.

(better at distinguishing between branches)

Finding informative ones can be done with *feature selection*.

Placement as classification

Placement classifier:

$$f : X \rightarrow Y$$

X is a set of possible query sequences
Y are branches of the reference tree

Note that RAPPAS is a *classifier*, but **NOT** a *machine learning* algorithm

Text classification (example)

Input: sequences of words

Output: a label $y \in Y = \{spam, ham\}$

A classic example: spam detection.

Text classification (example)

Input: sequences of words

A classic example: spam detection.

Output: a label $y \in Y = \{spam, ham\}$

x	$\hat{y}(x)$
SALE -146%! Claim your prize...	spam
Hi guys! The meeting on reducing the number of meetings is...	ham

Bernoulli Naïve Bayes text classification

query	$B_w = \{0, 1, 0 \ldots, 0\}$	presence of words
parameters	$P_y(w)$	$\mathbb{P}(\textit{sale} \mid \textit{spam}) = 0.7$
classifier	$\prod_{w: b_w = 1} P_y(w) \prod_{w: b_w = 0} \big(1 - P_y(w)\big)$	

Bernoulli NB and RAPPAS

	Bernoulli NB	RAPPAS
goal	text classification	placement
query	$B_w = \{0, 1, 0 \ldots, 0\}$ presence of words	$N_w = \{1, 0, 2, \ldots, 0\}$ counts of k-mers
parameters	$P_y(w)$ $\mathbb{P}(sale \mid spam) = 0.7$	$S_y(w)$ $\mathbb{P}(ACT \mid y) = 0.42$
classifier	$\prod_{w:b_w=1} P_y(w) \prod_{w:b_w=0} \big(1 - P_y(w)\big)$	$\prod_{w:n_w>0} S_y(w)^{n_w}$

Bernoulli placement

	Bernoulli placement	RAPPAS
goal	placement	placement
query	$B_w = \{0, 1, 0 \ldots, 0\}$ presence of k-mers	$N_w = \{1, 0, 2, \ldots, 0\}$ counts of k-mers
parameters	$P_y(w) := f \cdot S_y(w)$ corrected $S_y(w)$	$S_y(w)$ $\mathbb{P}(ACT \mid y) = 0.42$
classifier	$\prod_{w:b_w=1} P_y(w) \prod_{w:b_w=0} \left(1 - P_y(w)\right)$	$\prod_{w:n_w>0} S_y(w)^{n_w}$

f is the relative query length

The connection

$$f \to 0 \implies$$

Bernoulli placement = RAPPAS.

$f = \frac{|q|-k+1}{m-k+1}$, relative query length

$|q|$ is the query length

m is the alignment length

Interpretation: when placing short queries with long reference alignments.

Maximal Mutual Information filter

Mutual Information (MI): how informative one random variable is about another one.

MI filter for RAPPAS maximizes

$$\mathbb{P}(B_w) \cdot \Big(H(Y) - H(Y \mid B_w = 1) \Big)$$

class variable Y, presence of k-mers B_w

Maximal Mutual Information filter

Mutual Information (MI): how informative one random variable is about another one.

MI filter for RAPPAS maximizes

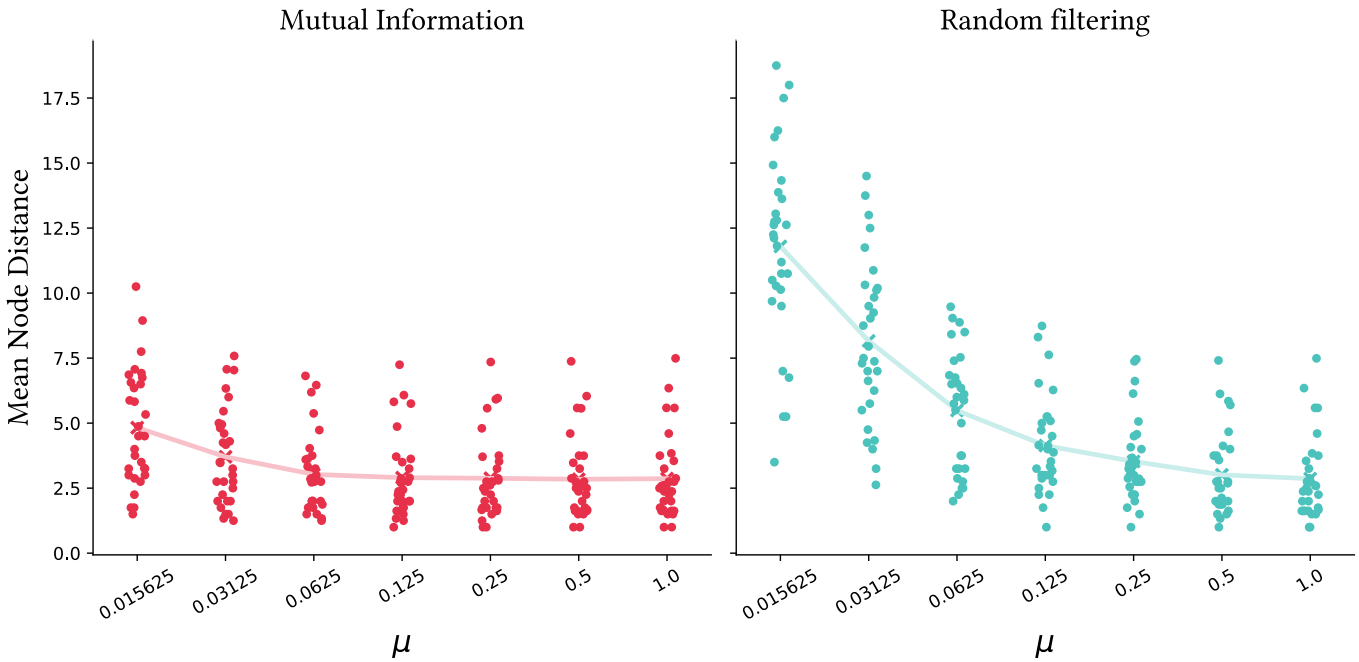
$$\mathbb{P}(B_w) \cdot \Big(H(Y) - H(Y \mid B_w = 1) \Big)$$

class variable Y, presence of k-mers B_w

Interpetation:

- how probable is the k-mer w
- how informative it is when observed

Experimental results

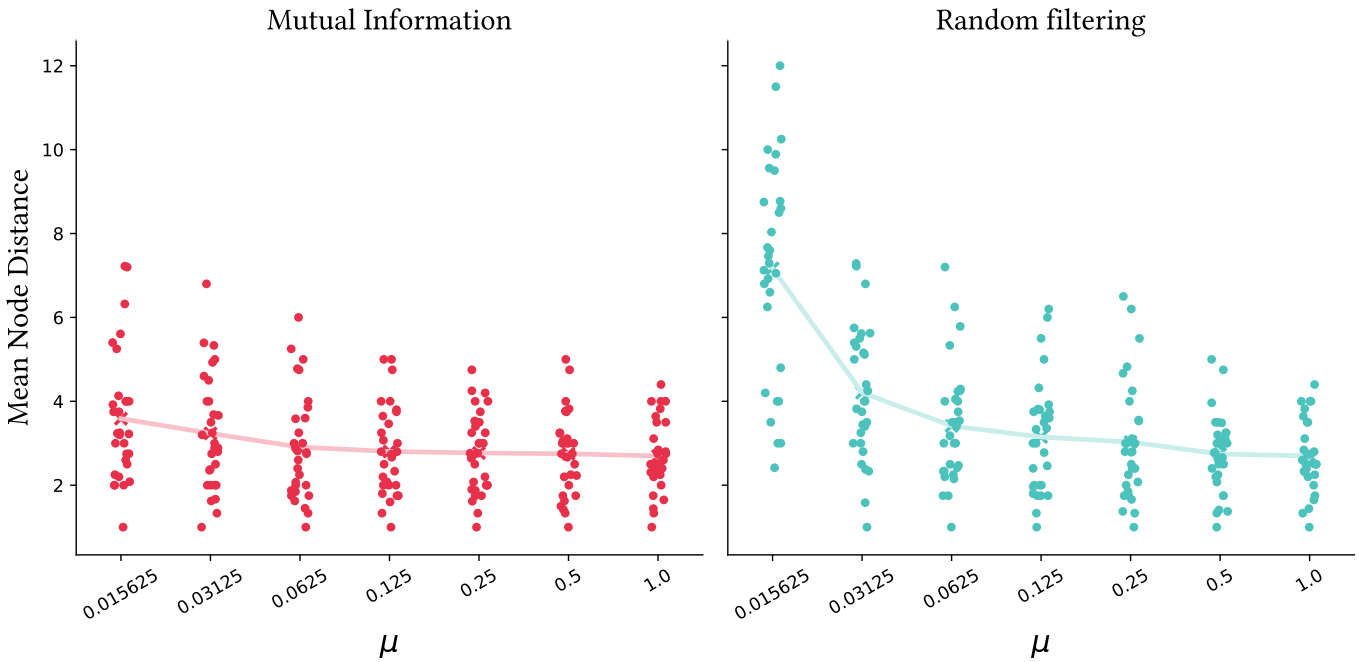


Data: D500 | Chloroplast rbcL | 1.4 Kbp | query length 300 bp | $k = 10$

μ is the fraction of phylo-k-mers kept

Interpretation: can keep only 6% of information with a negligible loss in accuracy.

Experimental results (contd.)



Data: D652 (bv) | Bacterial 16S rRNA | 1.7 Kbp | query length 300 bp | $k = 10$

μ is the fraction of phylo-k-mers kept.

Interpretation: can keep 6% of information a negligible loss in accuracy.

Conclusion

1. Mutual Information-based feature selection works (on plants, bacteria, viruses)
2. Phylo-k-mers work.
3. Filtering will help dealing with placing onto large phylogenies.

Thanks for your attention!