

Algorithms for genomic data

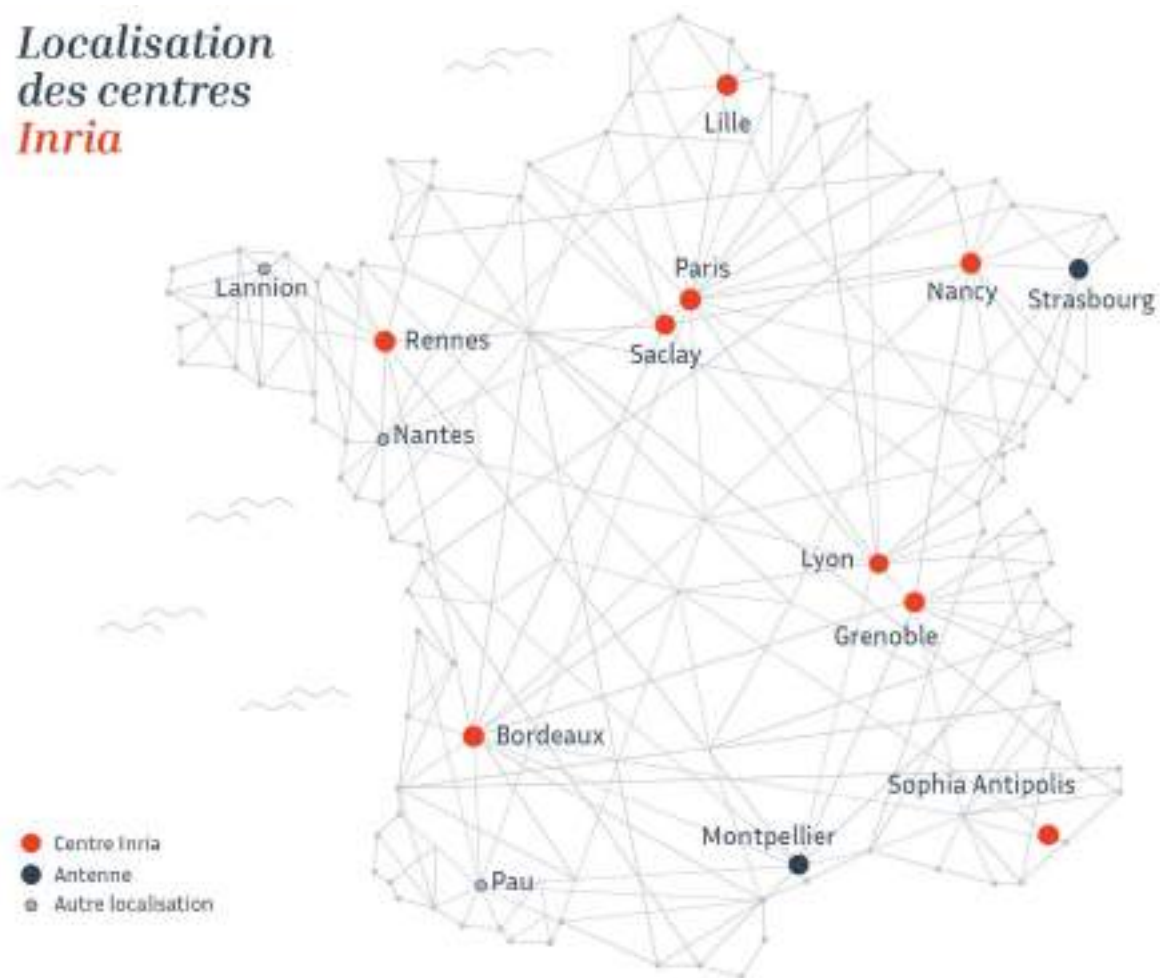
Pierre Peterlongo

DR Inria



Inria

Localisation
des centres
Inria

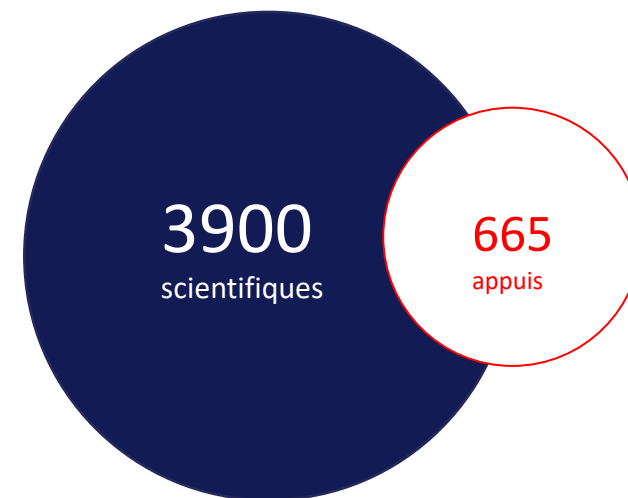


4565 collaborateurs



Inria 2670

Partenaires 1680

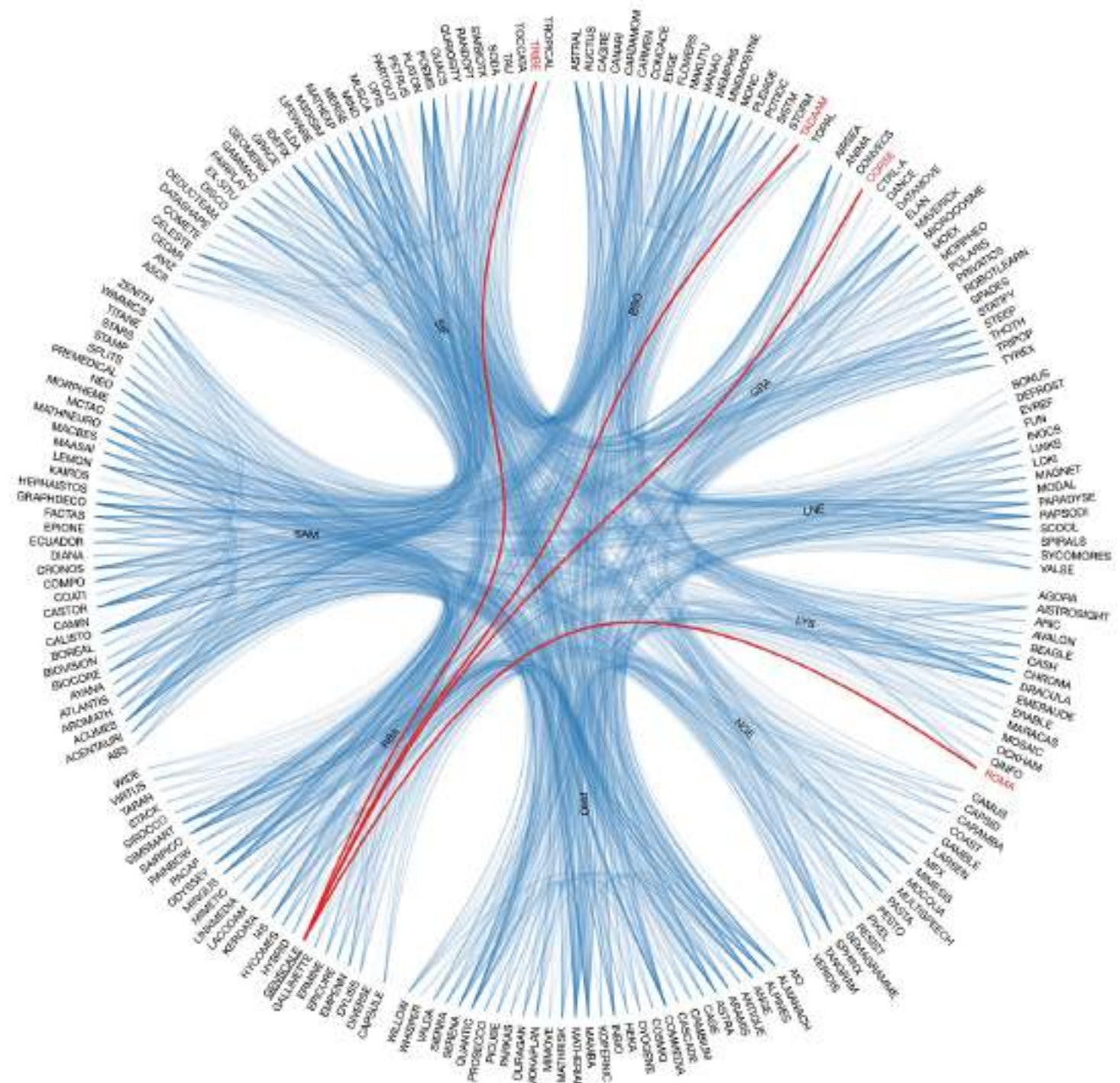


200 équipes

Inria

200 équipes

- Robotique
- IHM
- IA
- Drones
- Santé
- Visualisation
- Info théorique
- Simulations
- Compilation
- Embarqué
- ...
- **Bioinfo** (4 équipes, 2 à Rennes)



Bioinformatics at Inria Rennes



2 research teams



A platform



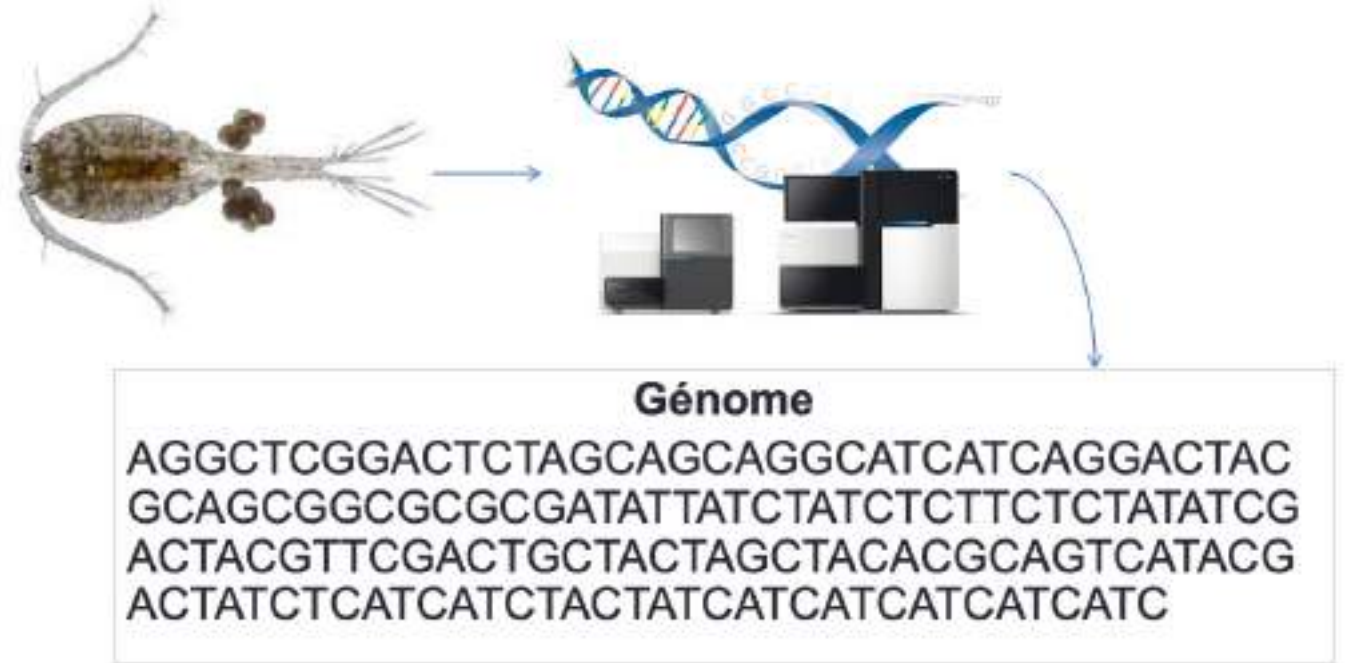
BIOINFO ?

BioInfo: Quand ?

Naissance : années 80 – 90

Après : analyse de

ADN, ARN, Acides Aminés...



Repère de taille

Génome humain (ACGGATTCTGGACTCAGCATCA...)

- A. Cent mille nucléotides ?
- B. Cinq cents mille ?
- C. 3 millions ?
- D. 3 milliards ?
- E. 30 milliards ?

Repère de taille

Génome humain (ACGGATTCTGGACTCAGCATCA...)

- A. Cent mille nucléotides ?
- B. Cinq cents mille ?
- C. 3 millions ?
- D. **3 milliards**
- E. 30 milliards ?

Repère de taille

Génome humain (ACGGATTCTGGACTCAGCATCA...)

- A. Cent mille nucléotides ?
- B. Cinq cents mille ?
- C. 3 millions ?
- D. **3 milliards**
- E. 30 milliards ?



Neoceratodus forsteri: **43 milliards**

Repères Historiques

| | |
|--------------|--|
| 1953 | Découverte de la structure en double hélice de l'ADN James Watson, Francis Crick, Rosalind Franklin |
| 1976 | Développement des 1ères techniques de séquençage d'ADN |
| 1977 | Premier génome séquencé : virus (5 386 nt) |
| 1984 | Génome du virus du sida (170 000 nt) |
| 1990 | Début du projet Génome humain |
| 1995 | Premier génome organisme vivant (bactérie H. influenza 1 800 000 nt) |
| 1996 | Premier génome eucaryote (levure 12 000 000 nt) |
| 2001 | Premier génome humain coût estimé à 3 milliards de dollars, une année |
| 2020 | Un génome humain <300 euros, une journée |
| Sept 2022 | 12 millions de génomes du covid séquencés |
| Juillet 2024 | 60 PetaOctets de données dans la banque SRA |

10 € /nt

1 € /nt

<0.0000001 € /nt

Repères Historiques

| | | |
|------|---|------------------|
| 1996 | Premier genome eucaryote (levure 12 000 000 nt) | 1 € /nt |
| 2001 | Premier génome humain coût estimé à 3 milliards de dollars, une année | |
| 2020 | Un génome humain <300 euros, une journée | <0.0000001 € /nt |

Si c'était un vol Paris -> New- York :

- 2001: 8h40 – 371 €
- 2020: ?? secondes - ?? €



Repères Historiques

| | | |
|------|---|------------------|
| 1996 | Premier genome eucaryote (levure 12 000 000 nt) | 1 € /nt |
| 2001 | Premier génome humain coût estimé à 3 milliards de dollars, une année | |
| 2020 | Un génome humain <300 euros, une journée | <0.0000001 € /nt |

Si c'était un vol Paris -> New- York :

- 2001: 8h40 – 371 €
- 2020: 42 secondes - 0.00004 €
(mille Aller-Retours pour 8 centimes)



Évolution

1997



2007



2015



- Chute des prix
- Augmentation de la capacité

Évolution – Aujourd'hui

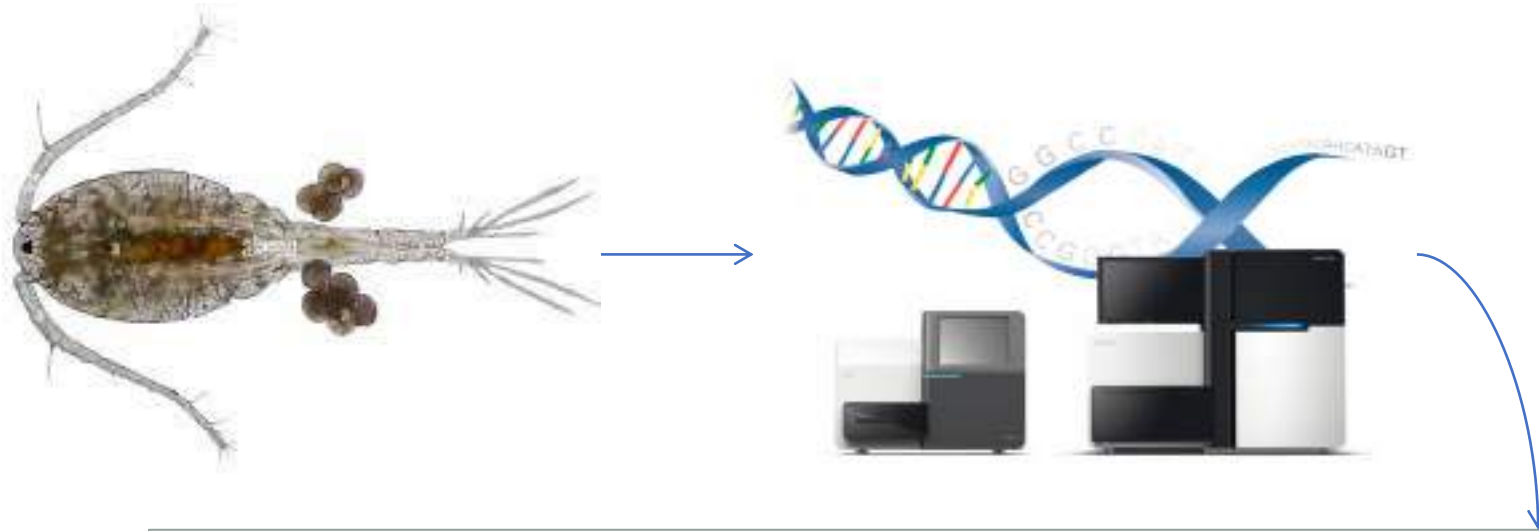


Les données brutes et l'assemblage



Les données brutes et l'assemblage

- Séquencer

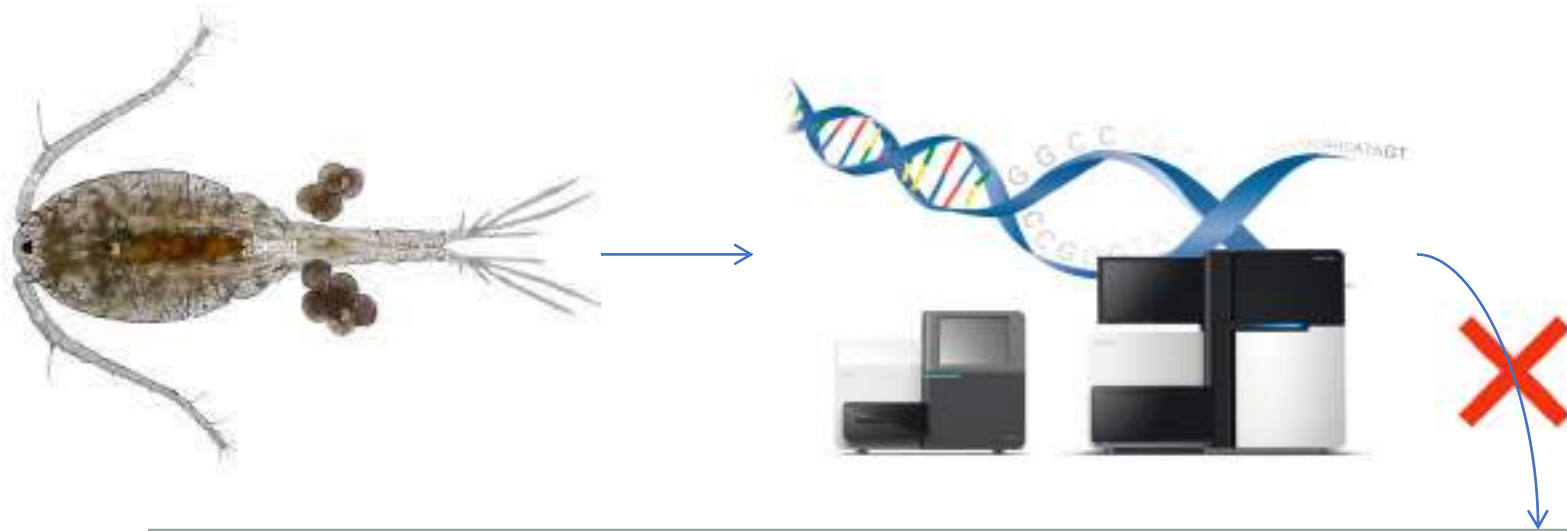


Génome

```
AGGCTCGGACTCTAGCAGCAGGCATCATCAGGACTAC
GCAGCGGCGCGCGATATTATCTATCTCTTCTCTATATCG
ACTACGTTTCGACTGCTACTAGCTACACGCAGTCATACG
ACTATCTCATCATCTACTATCATCATCATCATCATC
```

Les données brutes et l'assemblage

- Séquencer

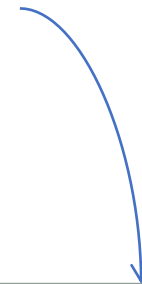
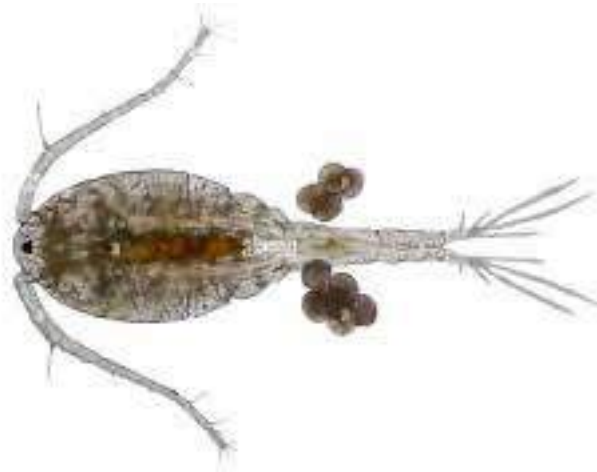


Génome

```
AGGCTCGGACTCTAGCAGCAGGCATCATCAGGACTAC
GCAGCGGCGCGCGATATTATCTATCTCTTCTCTATATCG
ACTACGTTTCGACTGCTACTAGCTACACGCAGTCATACG
ACTATCTCATCATCTACTATCATCATCATCATCATC
```

Les données brutes et l'assemblage

- Séquencer



0.1 à qq %
d'erreurs

« Reads » ou « lectures »

| | | | |
|---------|----------|-----------|------------|
| AGGCTCG | ATCGACT | ACGTCA | CAGTACTC |
| CAGCTTA | CAGTACTA | CAACTCA | ACGATC |
| ACGTACT | CACAGGT | ACGCAACT | TATCTCA |
| CATACT | CATTACG | CAGTCATAC | ACGACTAC |
| ACGACTC | ACGAGTC | ACGACTAT | ACGTAC ... |

L'assemblage

- Séquencer puis assembler

« Reads » ou « lectures »

AGGCTCG

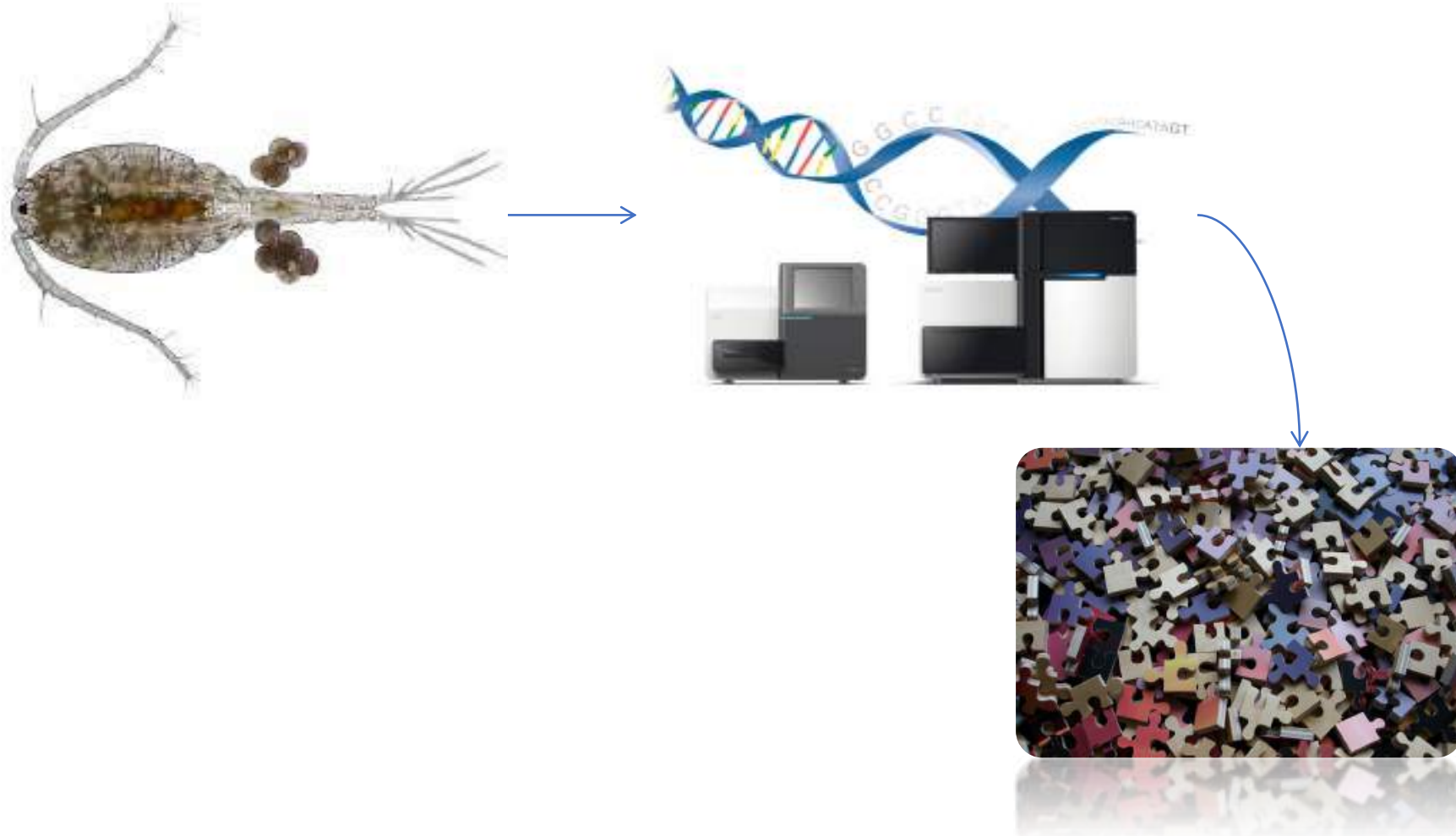
CTCGGATC

→

AGGCTCGGATC

L'assemblage

- Séquencer puis assembler



L'assemblage

| « Reads » ou « lectures » | | | |
|---------------------------|----------|-----------|------------|
| AGGCTCG | ATCGACT | ACGTCA | CAGTACTC |
| CAGCTTA | CAGTACTA | CAACTCA | ACGATC |
| ACGTACT | CACAGGT | ACGCAACT | TATCTCA |
| CATACT | CATTACG | CAGTCATAC | ACGACTAC |
| ACGACTC | ACGAGTC | ACGACTAT | ACGTAC ... |

Assemblage



Génome

...AGGCTCGGACTCTAGCAGCAGGCATCATCAGGACTA
CGCAGCGGCGCGCGATATTATCTATCTTTCTCTATATC
GACTACGTTGACTGCTACTAGCTACACGCAGTCATAC
GACTATCTCATCATCTACTATCATCATCATCATC...

L'assemblage: comment feriez vous ?

Première approche

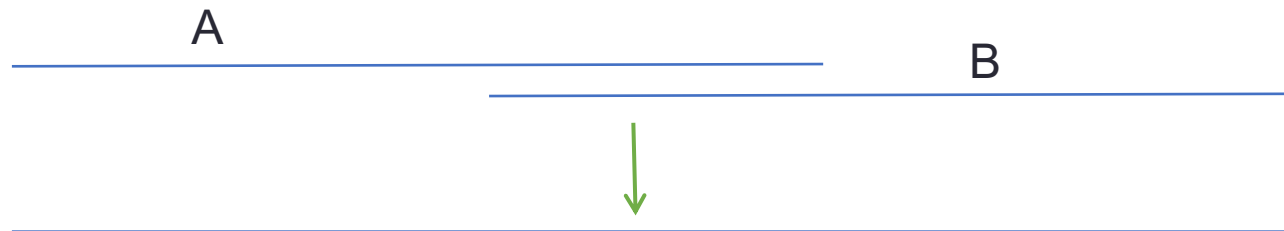
- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read A

Pour chaque (autre) read B

Si Suffixe de A == Préfixe de B :

Assembler A avec B



Première approche

- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read *A*

Pour chaque (autre) read *B*

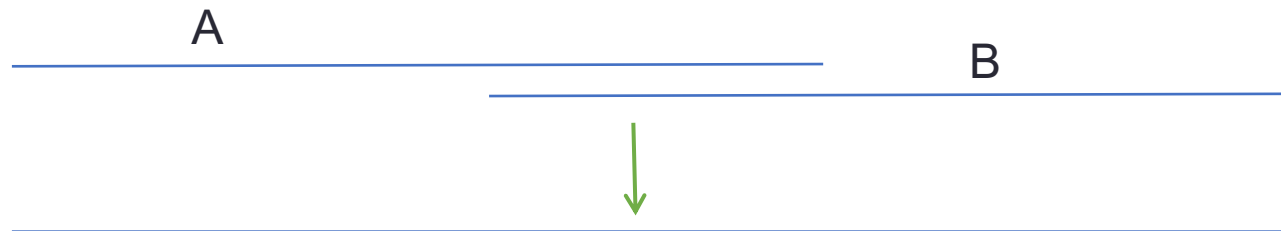
Si Suffixe de *A* == Prefixe de *B*:

Assembler *A* avec *B*

Que pensez vous de ces 2 là ?

A = ACGGCATGGCAGGCAGACTCAGT

B = AGGCAGACTCAGTACGTCATGCA



Première approche

- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read *A*

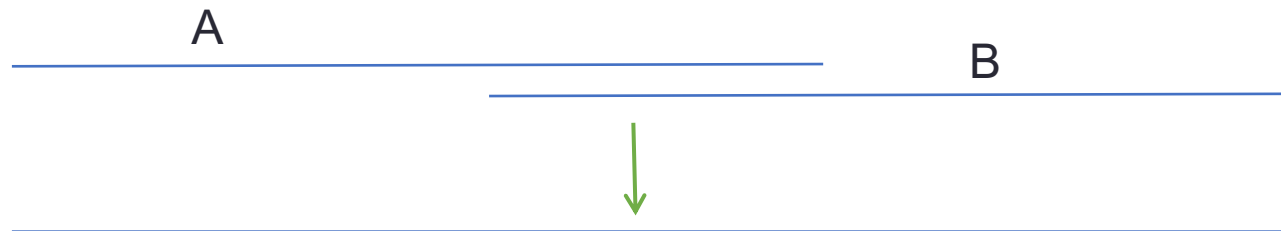
Pour chaque (autre) read *B*

Si Suffixe de *A* == Préfixe de *B*:

Assembler *A* avec *B*

Que pensez vous de ces 2 là ?

A = ACGGCATGGCAGGCAGACTCAGT
B = AGGCAGACTCAGTACGTCATGCA



Première approche

- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read *A*

Pour chaque (autre) read *B*

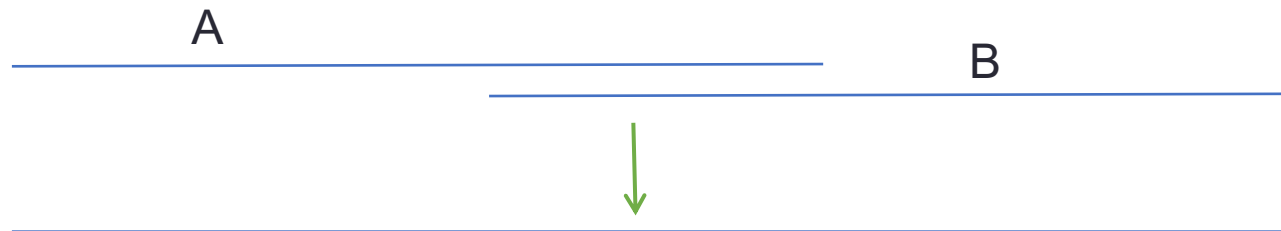
Si Suffixe de *A* == Préfixe de *B*:

Assembler *A* avec *B*

Que pensez vous de ces 2 là ?

A = ACGGCATGGCAGGCAGACTCAGT

B = AGGCAGACTCAGTACGTTCATGCA



Première approche

- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read *A*

Pour chaque (autre) read *B*

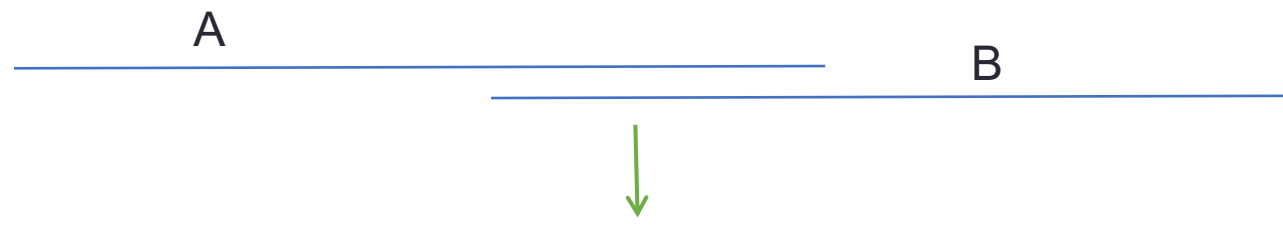
Si Suffixe de *A* == Prefixe de *B*:

Assembler *A* avec *B*

Que pensez vous de ces 2 là ?

A = ACGGCATGGCAGGCAGACTCAGT

B = AGGCAGACTCAGTACGTCATGCA



Première approche

- Comparer tous les reads entre eux.
- **Algo:**

Pour chaque read A

Pour chaque (autre) read B

Si Suffixe de A == Prefixe de B :

Assembler A avec B

```
ACGGCATGGCAGGCAGACTCAGT
                AGGCAGACTCAGTACGTCATGCA
                ↓
ACGGCATGGCAGGCAGACTCAGTACGTCATGCA
```

Première approche

- Comparer tous les reads entre eux.
- **Ok, mais combien de temps ça prend ?**

Avec 500 millions de reads (techno actuelle)

- 500 millions * 500 millions =
 - 250 millions de milliards de comparaisons.
- 0.000001 seconde par comparaison (très optimiste)

Première approche

- Comparer tous les reads entre eux.
- **Ok, mais combien de temps ça prend ?**

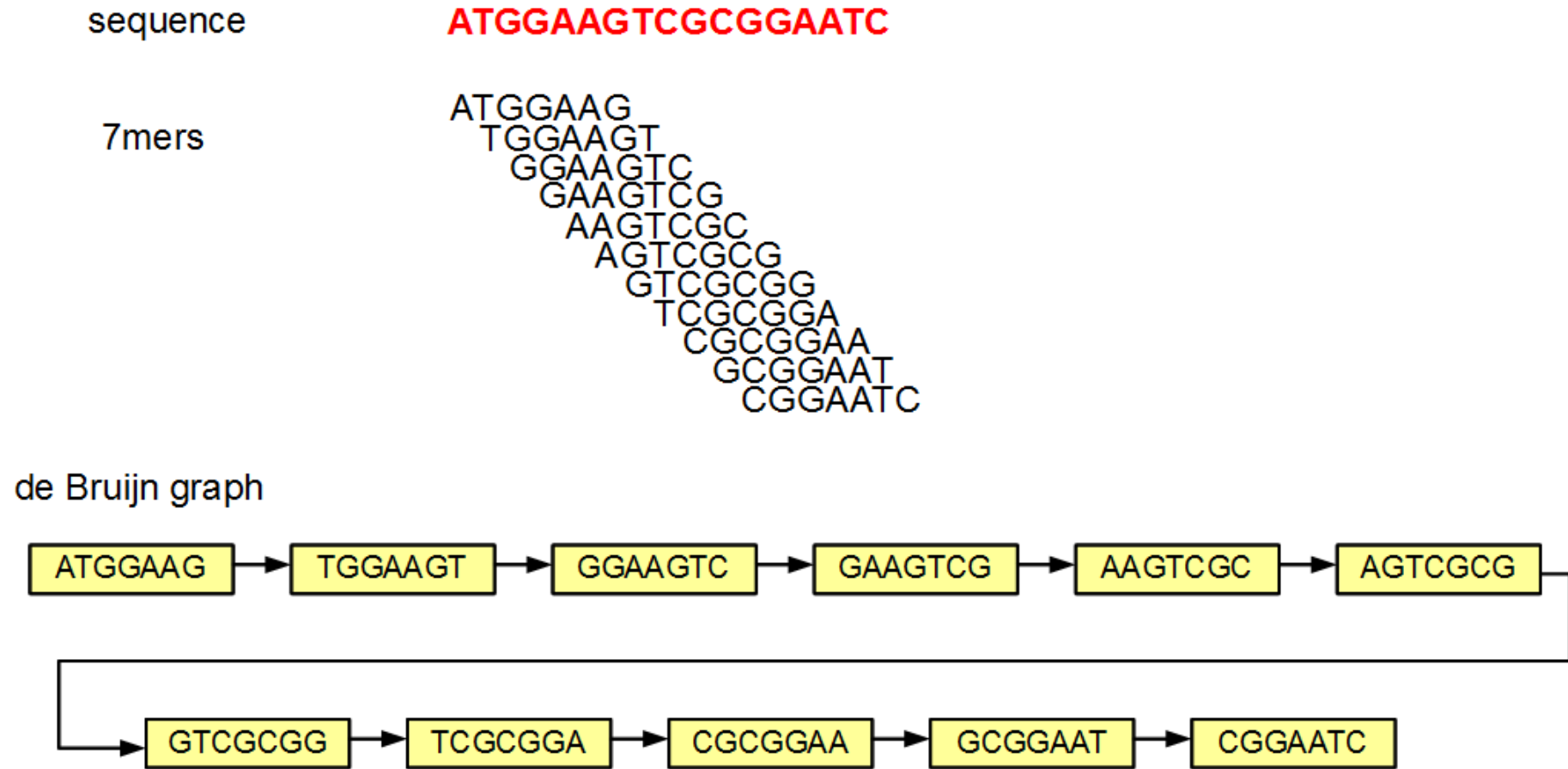
Avec 500 millions de reads (techno actuelle)

- 500 millions * 500 millions =
 - 250 millions de milliards de comparaisons.
- 0.000001 seconde par comparaison (très optimiste)
 - **~20300 siècles de calcul**



Deuxième approche kmers & graphe de « *de Bruijn* »

- De plusieurs années de calcul à quelques heures.



Comparer



- Trouver des **variants**

Comparer : pourquoi ?



Résistance
maladie ?

...GATT**ACGTCGTCATAC**GGCA...



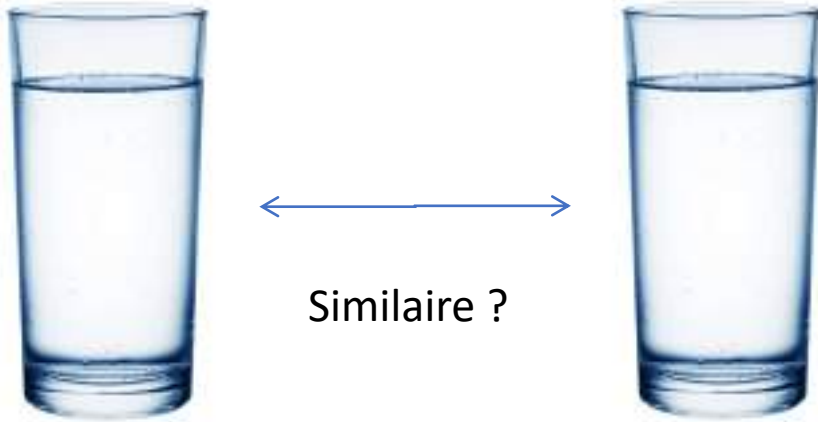
...GATT**GCATCTGGATTC**GGCA...

Mais aussi:

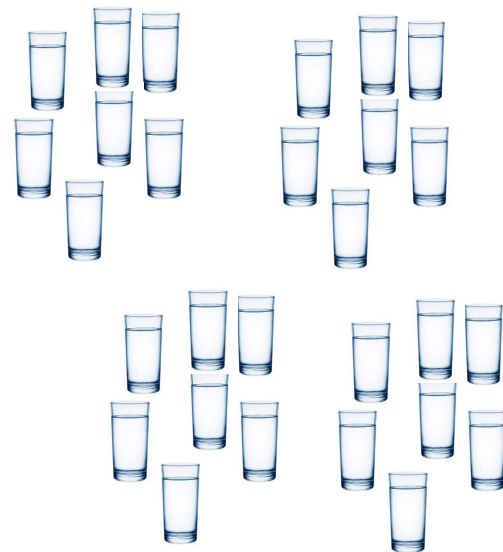
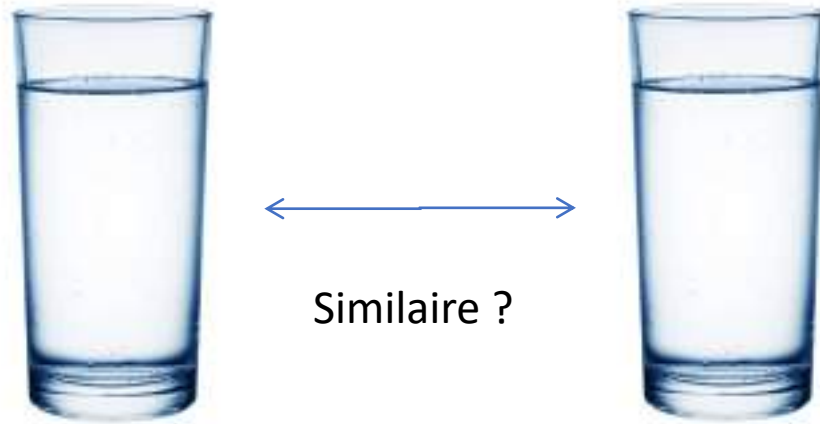
- Évolution / adaptation
- Traits phénotypiques
- ...

- Trouver des **variants**
- Estimer la **similarité**

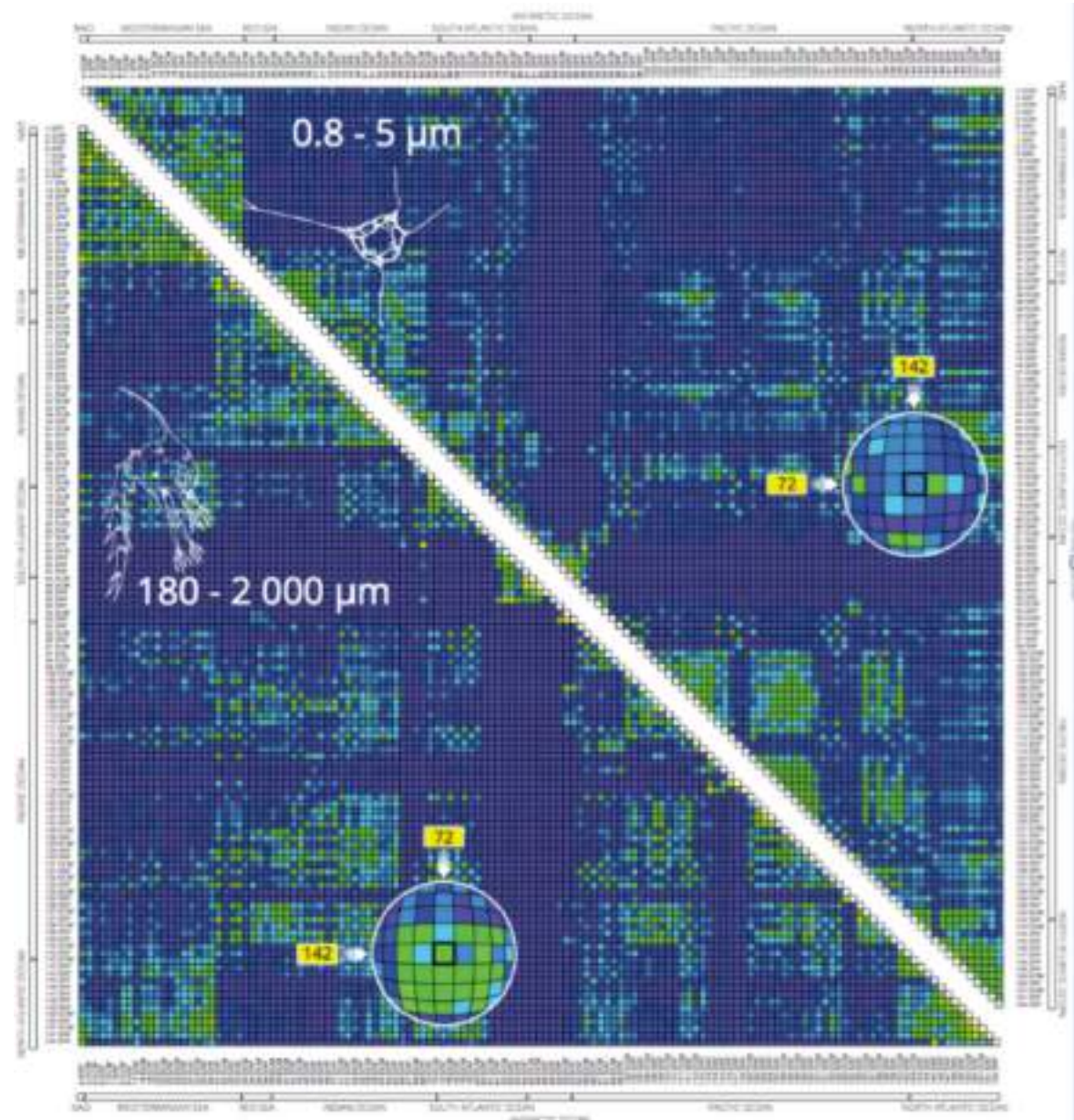
Comparer : pourquoi ?



Comparer : pourquoi ?



- Trouver des **variants**
- Estimer la **similarité**



Estimer la similarité : comment ?

GGGTGCAATACTCGCGATTCAATATCCTGTATGCGATTACAGTGAATAAAGTAGGCAGGAACAATGCATAGGATCGCACTATTTTACGGCAGAATCCTGCTCGTTCTCCCTAGCGGACGTGGTCCGTTTCCGAGCTATCTTACTGCTTAAAGCGGATCTGGAACACTCGAGCAATGATGCACAGGGATCCAACGTCCGTTCAACTCGTTTGTAGCCGTCTGATAATCCGGTCTCCTGCTTTTAAACAGTTCGCATGCAGACGCCGAAAAGAATGCAGAAGATAGTTTATGCAAATTTGGTTGGACGAGAATCGTTGGAGTATTAGATTGATACCTAGAACCCTTGCAAAGGAACAGCAGGCCGTATCATCCGTCGGAATACAATTTGATTGTCAATCGACCTCGCGCCTGGTGATAGGGTCCAATAGTACAGGCAATGAAGTGGTTGCTCGGTGCTAAATGTTATAGATTCACTACGTGCCATGCGTGCTGCCCCACTGTTTTGTTTTCCGAGCTAGGCTAAATGCTGTTTTGGGATCATGACGCCACTAGAACCTTTCCAATATAACCCAGTCTCTTTTAGCTCGGTCTAAACTTGGTTGTTTTGCACCTGTACAAAGGTGAGTTGGACGTTGCCACCTGTTTTAATCATGACTGGCTGCGTGACTGCATCACTTAATAAGACATGTAGAGCCTCAGATAGCGACATCCACCGTGGATGTTTTAAC TGCCTAGCGCACACAACGTAAGTTAATTAGCTCGTCCGCGTGAGTAGACGTGTGGCGCGACATACGCGACAATACTCACTGGTCCCTTACGATTATAGTAGTTGGACCGCAGCTCCTGGTAGTGACCAGGGGACGGTACATCTGACGCTATTCATAACCCATTCCGAGGACCAATGATAGGCTACTCGAGCGGGTGGCAACCGAACCATCAGCACCCAGCAGTATTTACATAAAACACAACCGGCAATAACGTGTATTGTCCAGCCGCTATCTCGATGGTTCTGTTCAAATAGACAGCACTCCTTAGTCCGAATGTAAGAACGCAGTTTGTCTTAATTTTCGGCTAGAAATGCAACAGACGCCCGACCGTCAGATTGCTTGTGTCCAAGATGCCTTGGCCATGCACATTTTCGTGAAACTAGGTAGGTGAGTCTCTGCGTTTCGCTTAGACAGCCCGGTAGAAAGTTCGTGCTTCATGTAGACCTCTAAGGTGAACTTTACGCACAGGGGAGGGAGTATGCACGACGTAGGATGCCGTTACGAAAACGCCCTGACCTGTGCGGCGCCTAAAGGACAGCGACTGATTTTGTCTTTGCCAATTAGTATGGAGTATTTTTCATAACCCCTGCCACCTAAGGCGCTACGCTACGGGGCCGCACCGGTGCTGACTGGTACAGCCGTAATGGAACCTTAGCAGGAGTAACCTGCGAACATTCGGGGTCATGGGTAAATCTTAAAAACAGTCAGACTGGGCCCTGTTTACGCCGATACTACTATGCTATGCGTAAGGGACGAGCAACGTTAGCTCAGGAAAGGCTCTAGGCGAGTGGTGTGAGAGTTTTTTCACCAATATCGGCTCCTCTGAAAGAGTGTATATCAGGCTACTTTTGGACGGCTAGGCGATCGATGGTTACTCACCGGTATGCTCTCAATCGGGCCGGTCAACACCATCGATAGATCTCTCCTCGAAAAGTCTTTGCCCTCATCGACAAAATAAAGGAATTACCACCAAACCTTAAAGTTACTCATGGGCTTGTGGGATCAGCCTCCGCCCTGTGAAAAGGAGGCTGTTACGAGAGTCATAGTTGAGGACAATAGATCGCTGGATTTTACTGGTCTGTATCGCTGAGCAGTATAACCTTCCCTACACCTTCTGACCCGCTGACGTTCCGACAGAATTTTGGCTACCCGAAAGCGGACACAGCAAAAAGATGCGAGTGTGAAGTGTGACTGCGCGGCTTTTCCGCTGATAAAGGA AAAATTTGAGGCTCGCTGCGCCGATCACTCTCCATCTACGGGGTCAATGTTAAAGTTCAAACCTCGTGGTCCA CTGTATACTAATCGAAAAGATCTCCGTGAGAAGGGGATAAGGGTCTTAAAGCATGGTTTTCCGGAGTTCTCGCCG TCGGGGAGTGGCGTGGTTCTACCCCGCAGCTGATTAGCCCGTTTACACCTTTGGTCTGCGGGCGGCCGATC CGTGGTCTCGGGCAGATGCTCTTTCCGGAGTAGTGAATTAGCGTTTCCCAATACCCATAGTAAATGGTTTTGCT AGCTACCTGACAACGAGGCGCTATGGTGTCTCATCAGAGGCGCGCTTGACCGGTTTTGCTAATTCGCGCTT CGACACCCCTATTTATATTTCTAACCCATATCCAGGCAAACATAACAACATGACCCCGGAACCTATATTTGGATA AGCCACAGCCCATGATGGCCATCTCGACACAACAGGGATTGACTGCCACCAACTACTCGTCAATGCTCTGGC CACGCATAACAGACTGTTTTGGCTGACAAGTGTTCACAACATAAAGGTTACAGACGTTTTTATCTAGGGTCCC TAACGATCTCATAACGGTTGACCTAAATCGCAACGGTTTTCTCACCAAAGCTGAAAGTACTACTTGTGACGGAAC CGGGGCCGATAGGACCATACGGACACGAGTTTCATAGCATACTCACGTTCCGGAGACTAGCACGGATGTAAT CTTCAATTGAGAGTATGTCCGCGGTAAGTCTTGGACTGCCGATTTTCCAACAGAAATCTCCGGAACACATCTCTG TAATAGGACGTGCAAAAGCAGATTTAGGTGACATCGTTGGGTGATTATGTGCGTGCCTTTGTTTACGGCATG CGAAGTCATCTATATACCCGAGACATACGGCACCTGGACAAAATTATGCTCCTCCGTTAGGAGTGGGAACCTTA TAATACTCAGAACCAGAGATTGGGTTCCGCTGTAGGGCCGAGAACAGTTAAGTTTCAGCTAGGGGGAATTTCC CAATGGCCCTGTGGATCGGAGGATCGGTGGAACAGTTTCGAGCCTCCCAATTTTGAATAGAAGTGGTTGCTT CCTGCTCTCTCTTCCAGCCGACTGCTAACAATTTTAAAGCCCTACAGCTCCGCTGCAACAGCTCCGCGCCGCA

AATACTCGCGATGCAATATCCTGTATGCGATTACAGTGAATAAAGTAGGCAGGAACAATGCATAGGATCGCAC TATTTTACGGCAGAATCCTGCTCGTTCTCCCTAGCGGACGTGGTCCGTTTCCGAGCTATCTTACTGCCTTAA GCGGATCTGGAACACTCGAGCAATGATGCACAGGGATCCAACGTCCGTTCAACTCGTTTGTAGCCGTCTGATA ATCCGGTCTCCTGCTTTTAAACAGTTCGCATGCAGACGCCGAAAAGAATGCAGAAGATAGTTTATGCAAATTTGG TTGGACGAGAATCGTTGGAGTATTAGATTGATACCTAGAACCCTTGCAAAGGAACAGCAGGCCGTATCATCCG TCTGGAATACAATTTGATTGTCAATCGACCTCGCGCCTGGTGATAGGGTCCAATAGTACAGGCAATGAAGTGG TTGCTCGGTGCTAAATGTTATAGATTCACTACGTGCCATGCGTGCTGCCCCACTGTTTTGTTTTCCGGAGCTA GGCTAAATGCTGTTTTGGGATCATGACGCCACTAGAACCTTTCCAATATAACCCAGTCTCTTTTAGCTCGGTGCT CTAAACTTTGGTTGTTTTGCACCTGTACAAAGGTGAGTTGGACGTTGCCACCTGTTTTAATCATGACTGGCTG CGTGACTGCATCACTTAATAAGACATATAGAGCCTCAGATAGCGACATCCACCGTGGATGTTTTAAGTGGCTA GCGCACACAACGTAAGTTAATTAGCTCGTCCGCGTGAGTAGACGTGTGGCGCGACATACGCGACAATACTC ACTGGTCCCTTACGATTATAGTAGTTGGACCGCAGCTCCTGGTAGTGACCAGGGGACGGTACATCTGACGCT ATTCATAACCCATTCCGAGGACCAATGATAGGCTACTCGAGCGGGTGGCAACCGAACCATCAGCACCCAGCAGT ATTTACATAAAACACAACCGGCAATAACGTGTATTGTCCAGCCGCTATCTCGATGGTTCTGTTCAAATAGACA GCACTCCTTAGTCCGAATGTAAGAACGCAGTTTGTCTTAATTTTCGGCTAGAAATGCAACAGACGCCCGACCG TCAGATTGCTTGTGTCCAAGATGCCTTGGCCATGCACATTTTCGTGAAACTAGGTAGGTGAGTCTCTGCGTTTT C GTCTAGACAGCCCGGTAGAAAGTTCGTGCTTCATGTAGACCTCTAAGGTGAACTTTACGCACAGGGGAGGGA GTATGCACGACGTAGGATGCCGTTACGAAAACGCCCTGACCTGTGCGGCGCCTAAAGGACAGCGACTGATTT TGCTTTGCCAATTAGTATGGAGTATTTTTCATAACCCCTGCCACCTAAGGCGCTACGCTACGGGGGCCGACC GGTGCTGACTGGTACAGCCGTAATGGAACCTTAGCAGGAGTAACCTGCGAACATTCGGGGTCATGGGTAAATCT TAAAAACAGTCAGACTGGGCCCTGTTTACGCCGATACTACTATGCTATGCGTAAGGGACGAGCAACGTTAGC TCAGGAAAGGCTCTAGGCGAGTGGTGTGAGAGTTTTTTCACCAATATCGGCTCCTCTGAAAGAGTGTATATCA GGCTACTTTTGGACGGCT**ACGGCAGCATCATGCTAGTCACTGTCAGTGTGACTCA**AGGCGATCGATGGTTACTCA CCGGTATGCTCTCAATCGGGCCGGTCACAACCATCGATAGATCTCTCCTCGAAAAGTCTTTGCCCTCATCGAC AAATAAAGGAATTACCACCAAACCTTAAAGTTACTCATGGGCTTGTGGGATCAGCCTCCGCCCTGTGAAAAGG GTTTACGCCCATTTATAGAAAATTTGAAGGAGGCTGTTACGAGAGTCATAGTTGAGGACAATAGATCGCTGGAT TTTGTTGGATGATCGCATTTTCGAGTATTAACCTGGTCTGTATCGCTGAGCAGTATACTTCCCTACACCTTCC TACCCGCTGACGTTCCGACAGAATTTGCGTACCCGAAAGGACACAGACGCAAAAAGATGCGAGTGTGAAGTG GTAGCGCCGCGCTTTTCCGCTGATAAAGGAAAAATTTGAGGCTCGCTGCGCCGATCACTCTCCATCTACGGGG TCATTTGTAAGGTTCAAACCTCGTGGTCCACTGTATACCTAATCGAAAAGATCTCCGTGAGAAGGGGATAAGGGTC TTAAGCATGGTTTTCCGGAGTTCTCGCCGTCGGGGAGTGGCGTGGTTCTACCCCGCAGCTGATTAGCCCGTT CACACCTTTGGTCTGCGGGCGGCCGATCCGTTGGTCTCGGGCAGATGCTCTTTCCGGAGTAGTGAATTAGCGTT CCTCAATACCCATAGTAAATGGTTTTGCTAGCTACCTGACAACGAGGCGCTATGGTGTCTCCTCATCAGAGGCG CGCTTGACCGGTTTTGCTAATTTCCGCTTCGACACCCCTATTTATATTTCTAACCCCTATCCAGGCAAACATA CAACTGAGCCCGGAACCTATATTTGGATAAGCCACAGCCCATGATGGCCATCTCGGACACAACAGGGATTGAC TGCCACCAACTACTCGTCAATGCTCTGGCCACGCATAACAGACTGTTTTGGCTGACAAGTGTTCACAACCTAT AAGGTTACAGACGTTTTTATCTAGGGTCCCTAACGATCTCATAACGGTTGACCTAAATCGCAACGGTTTTCTCAC CAAAGCTGAAAGTACTACTTGTGCA**TTT**AACCGGGCCGATAGGACCATACGGACACGAGTTTCATAGCATACTC ACGTTCCGGAGACTAGCACGGATGTAATCTTTCATTGAGAGTATGTCCGCGGTAAGTCTTGGACTGCCGTATT TCCAACAGAATCTCCGGAACACATCTCTGGTAATAGGACGTGCAAAAGCAGATTTAGGTGACATCGTTGGGTG ATTATGTGCGTGCCTTTGTTTACGGCATGCGAAGTCATCTATATACCCGAGACATACGGCACCTGGACAAAAT TATGCTCCTCCGTTAGGAGTGGGAACCTTATAATACTCAGAACCAGAGATTTGGGTTCCGCTGTAGGGCCGAC ACAGTTAAGTTTTTCAGCTAGGGGAATTTCCCAATGGCCCTGTTGGATCGAGGATCGGTGGGAAAACGTTGAC CACCTGCTTTCCTTCCAGCCGACTGCTAACAATTTTAAAGCCCTACAGCTCCGCTGCAACAGCTCCGCGCCGCA

Estimer la similarité : comment ?

GGGTGC AATACTCGCGAT
GGTGC AATACTCGCGATT
GTGC AATACTCGCGATTC
TGC AATACTCGCGATTCA
GC AATACTCGCGATTCAA
CAATACTCGCGATTCAAT
AATACTCGCGATTCAATA

kmerisation

GGGTGC AATACTCGCGAT
GGTGC AATACTCGCGATT
GTGC AATACTCGCGATTC
TGC AATACTCGCGATTCA
GC AATACTCGCGATTCAA
CAATACTCGCGATTCAAT
AATACTCGCGATTCAATA

ATACTCGCGATTCAATAT
TACTCGCGATTCAATATC
ACTCGCGATTCAATATCC
CTCGCGATTCAATATCCT
...

AATACTCGCGATTCAATA
AATACTCGCGATTCAATA
ATACTCGCGATTCAATAT
TACTCGCGATTCAATATC
ACTCGCGATTCAATATCC
CTCGCGATTCAATATCCT
...

kmerisation

AATACTCGCGATTCAATA
AATACTCGCGATTCAATA
ATACTCGCGATTCAATAT
TACTCGCGATTCAATATC
ACTCGCGATTCAATATCC
CTCGCGATTCAATATCCT
...



Estimer la similarité : comment ?

GGGTGC AATACTCGCGAT
 GGTGC AATACTCGCGATT
 GTGC AATACTCGCGATTC
 TGC AATACTCGCGATTCA
 GC AATACTCGCGATTCAA
 CAATACTCGCGATTCAAT
 AATACTCGCGATTCAATA

kmerisation

GGGTGC AATACTCGCGAT
 GGTGC AATACTCGCGATT
 GTGC AATACTCGCGATTC
 TGC AATACTCGCGATTCA
 GC AATACTCGCGATTCAA
 CAATACTCGCGATTCAAT
 AATACTCGCGATTCAATA

ATACTCGCGATTCAATAT
 TACTCGCGATTCAATATC
 ACTCGCGATTCAATATCC
 CTCGCGATTCAATATCCT
 ...

AATACTCGCGATTCAATA
 AATACTCGCGATTCAATA
 ATACTCGCGATTCAATAT
 TACTCGCGATTCAATATC
 ACTCGCGATTCAATATCC
 CTCGCGATTCAATATCCT
 ...

kmerisation

AATACTCGCGATTCAATA
 AATACTCGCGATTCAATA
 ATACTCGCGATTCAATAT
 TACTCGCGATTCAATATC
 ACTCGCGATTCAATATCC
 CTCGCGATTCAATATCCT
 ...

Distances basées sur le nombre de kmers spécifiques et le nombre de kmers partagés (eg. **Jaccard**)

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

Estimer la similarité : comment ?

```
GGGTGC AATACTCGCGAT
GGTGC  AATACTCGCGATT
GTGC   AATACTCGCGATTCC
TGC    AATACTCGCGATTCA
GC     AATACTCGCGATTCAA
C      AATACTCGCGATTCAAT
      AATACTCGCGATTCAATA
```

```
ATACTCGCGATTCAATAT
TACTCGCGATTCAATATC
ACTCGCGATTCAATATCC
CTCGCGATTCAATATCCT
...
```

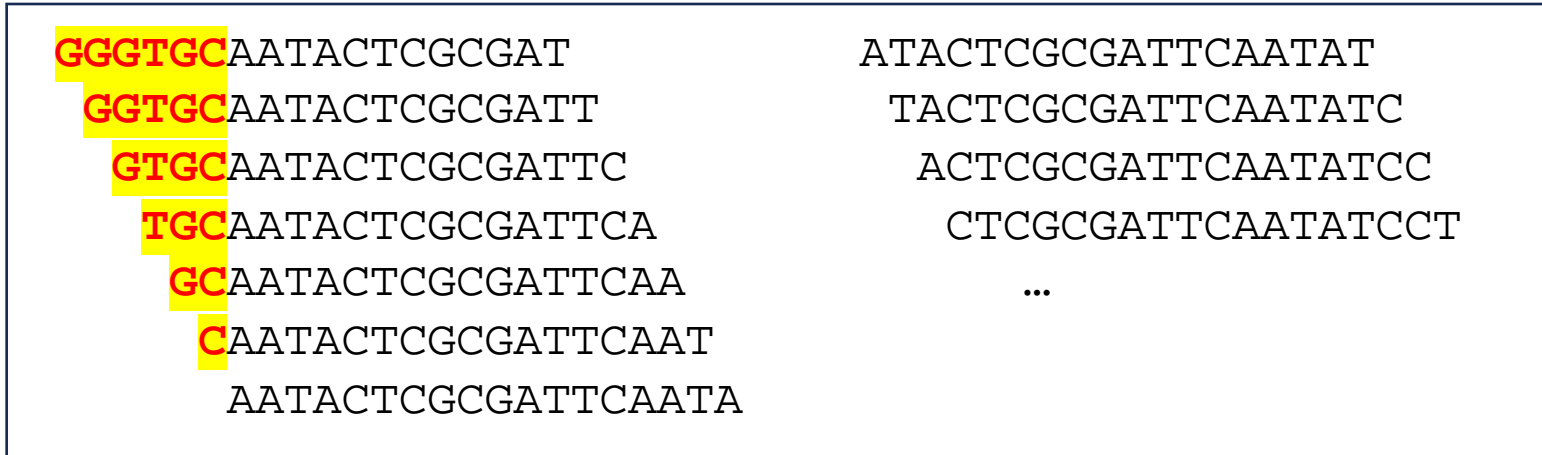
```
AATACTCGCGATTCAATA
AATACTCGCGATTCAATA
ATACTCGCGATTCAATAT
TACTCGCGATTCAATATC
ACTCGCGATTCAATATCC
CTCGCGATTCAATATCCT
...
```

Pas si facile:

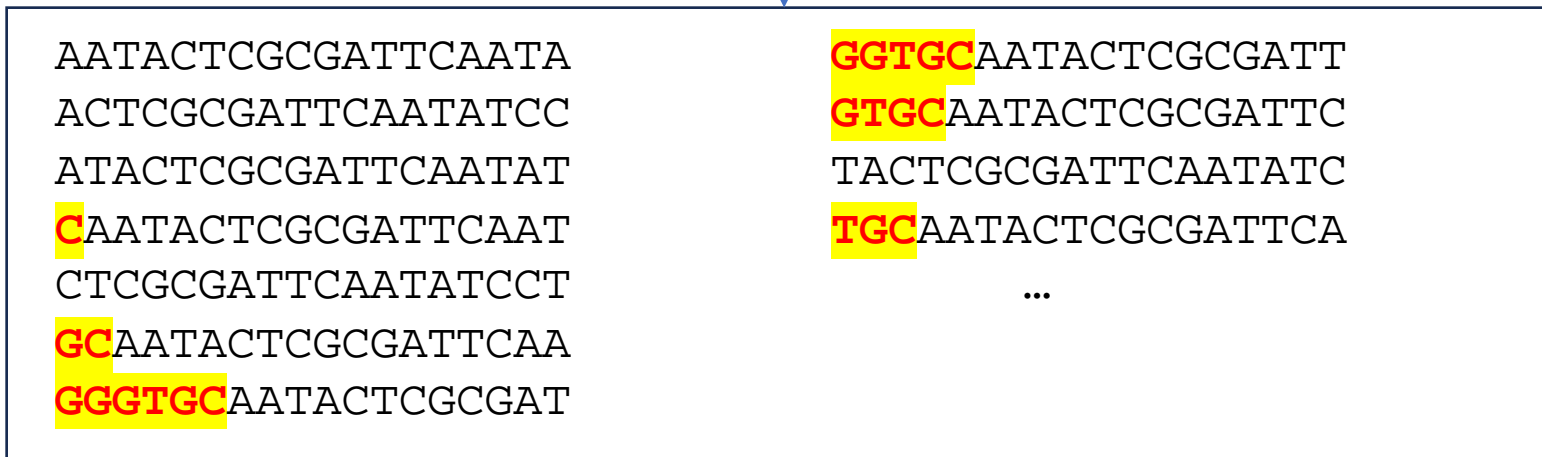
- 100aines de milliards de kmers
- non ordonnés
- (non orientés)
- (erronés)

Estimer la similarité : comment ?

Gros tas de kmers



Gros tas de kmers triés



Tri des kmers

Multithreading

Parallélisation par *minimizers*

Estimer la similarité : comment ?

```
AATACTCGCGATTCAATA      GGTGC AATACTCGCGATT
ACTCGCGATTCAATATCC      GTGC AATACTCGCGATT
ATACTCGCGATTCAATAT      TACTCGCGATTCAATATC
CAATACTCGCGATTCAAT      TGC AATACTCGCGATTCA
CTCGCGATTCAATATCCT
GCAATACTCGCGATTCAA
GGGTGC AATACTCGCGAT
```

```
AATACTCGCGATTCAATA
ACTCGCGATTCAATATCC
ATACTCGCGATTCAATAT
CTCGCGATTCAATATCCT
TACTCGCGATTCAATATC
...
```

Facile sur les kmers triés
- lecture de 2 fichiers simultanément

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

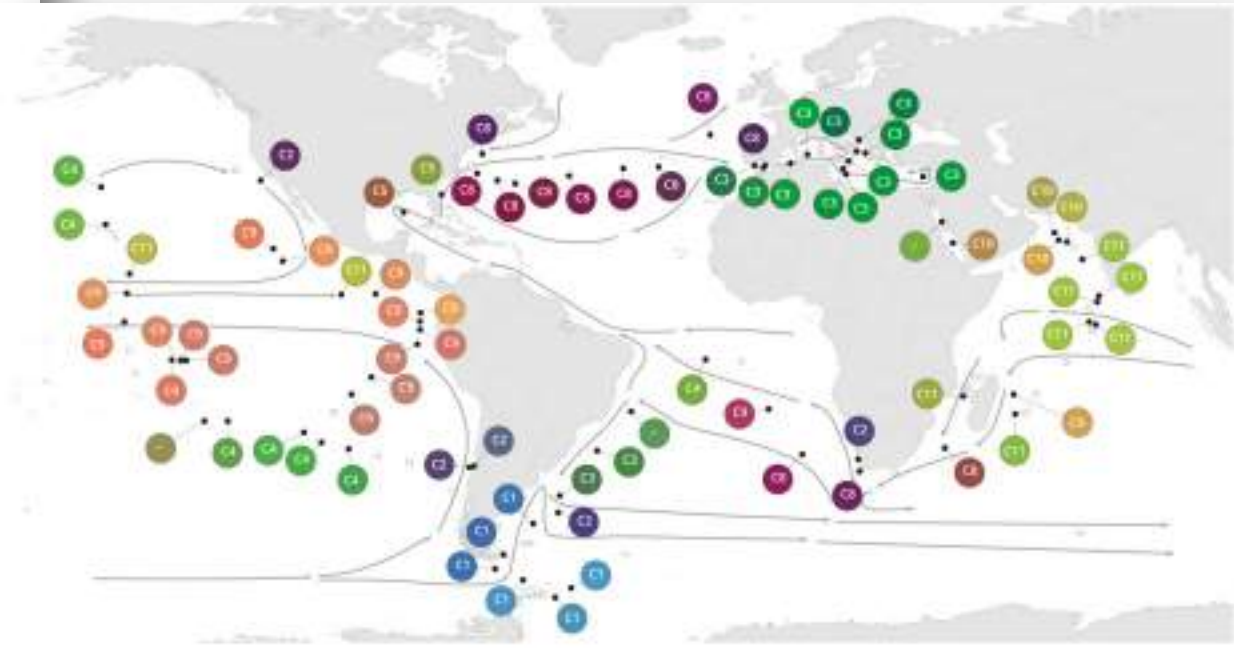
Estimer la similarité : application Tara

RESEARCH ARTICLE

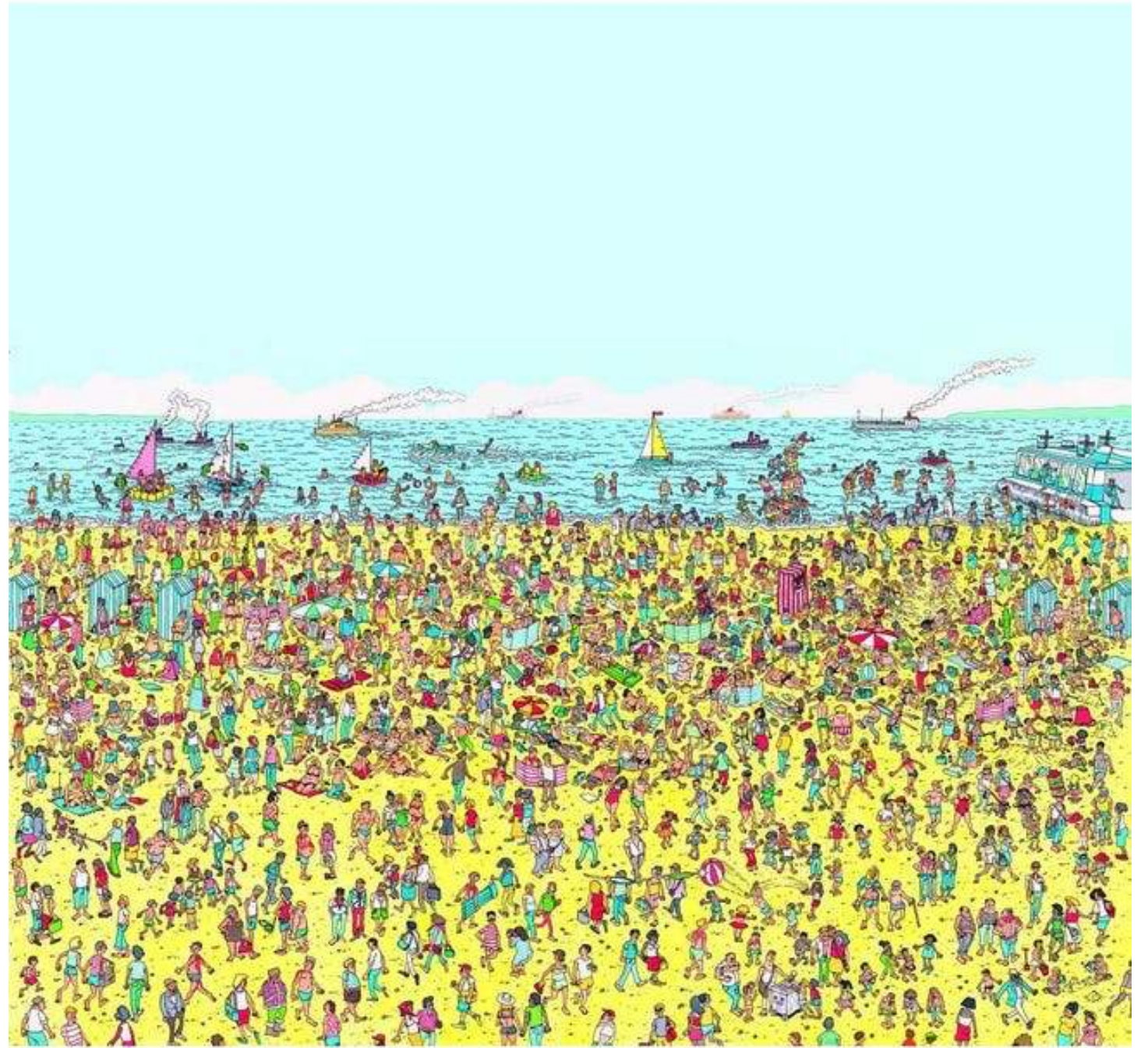


Genomic evidence for global ocean plankton biogeography shaped by large-scale current systems

Daniel J Richter^{1,2†}, Romain Watteaux^{3,4†}, Thomas Vannier^{5,6,7†}, Jade Leconte^{5,6}, Paul Frémont^{5,6}, Gabriel Reygondeau^{8,9}, Nicolas Maillet¹⁰, Nicolas Henry^{1,6}, Gaëtan Benoit¹¹, Ophélie Da Silva^{6,12}, Tom O Delmont^{5,6}, Antonio Fernández-Guerra^{13,14,15}, Samir Suweis¹⁶, Romain Narci¹⁷, Cédric Berney^{1,6}, Damien Eveillard^{6,18}, Frederick Gavory⁵, Lionel Guidi^{6,12}, Karine Labadie¹⁹, Eric Mahieu¹⁹, Julie Poulain^{5,6}, Sarah Romac^{1,6}, Simon Roux²⁰, Céline Dimier^{1,21}, Stefanie Kandels^{22,23}, Marc Picheral^{6,12}, Sarah Searson^{6,12}, Tara Oceans Coordinators, Stéphane Pesant^{24,25}, Jean-Marc Aury⁵, Jennifer R Brum^{20,26}, Claire Lemaitre¹¹, Eric Pelletier^{5,6}, Peer Bork^{22,27,28}, Shinichi Sunagawa^{22,29}, Fabien Lombard^{6,12,30}, Lee Karp-Boss³¹, Chris Bowler^{6,21,21}, Matthew B Sullivan^{20,32,33,34}, Eric Karsenti^{6,21,23}, Mahendra Mariadassou¹⁷, Ian Probert^{1,6}, Pierre Peterlongo¹¹, Patrick Wincker^{5,6}, Colomban de Vargas^{1,6*}, Maurizio Ribera d'Alcalà^{3*}, Daniele Iudicone^{3*}, Olivier Jaillon^{5,6*}



Rechercher



Fouiller les données



...GATT**ACGTCGTCATAC**GGCA...

...GATT**GCATCTGGATTC**GGCA...

Résistance
maladie ?

- Autres organismes ?
- Déjà étudié ?
- Quelles variations ?

GCATCTGGATTC



Google

AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCT



All

Images

Shopping

Videos

Maps

More

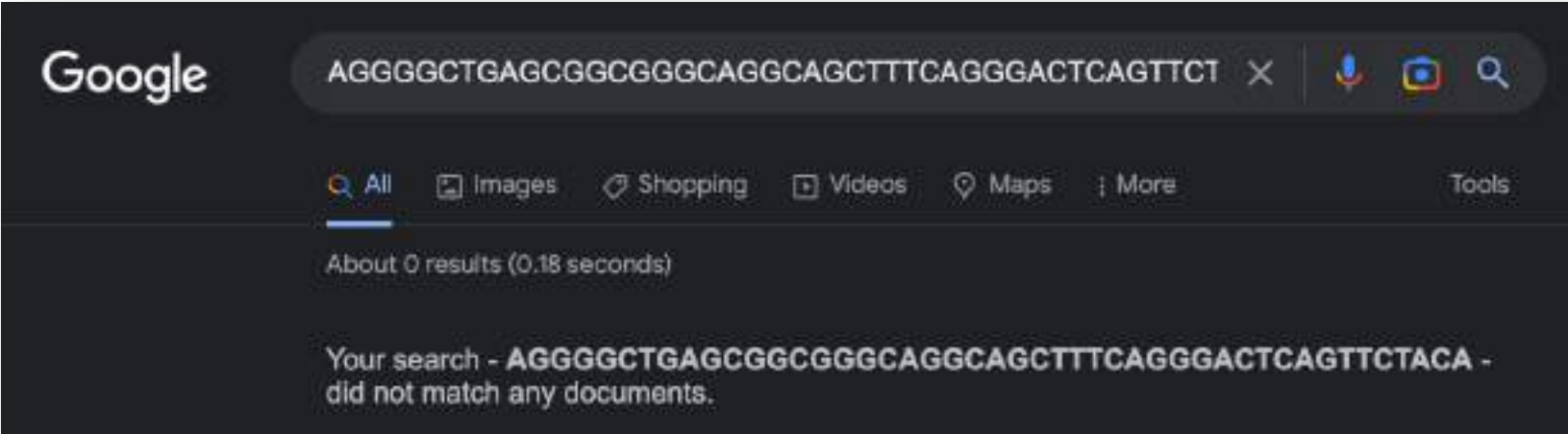
Tools

About 0 results (0.18 seconds)

Your search - **AGGGGCTGAGCGGCGGGCAGGCAGCTTTCAGGGACTCAGTTCTACA** - did not match any documents.



60 petabytes
24 millions samples
doubles: <2 years



Comment savoir si une requête existe dans un jeu de données ?

Query:

TTACCACCAAACCTTAAGTTACTCATGGGCTTGCT

Sample1

AGACCTCTAAGGTGAACTTTCACGCACAGGGGAGGGAGTA
TGCACGACGTAGGATGCCGTTACGAAAACGCCCTGACCT
GTGCGGCGCCTAAAGGACAGCGACTGATTTTGCTTTGCCA
ATTAGTATGGAGTATTTTCATAACCCCTGCCACCTAAGG
CGCTACGCTACGGGGGCCGCACCGGTCGTGACTGGTACAG
CCGGTAATGGAACCTTAGCAGGAGTAACTGCGAACATTCCG
GGTCATGGGTAATTCTTAAAAACAGTCAGACTGGGCCCT
GTTACGCCGATACTACTATGCTATGCGTAAGGGACGAGC
AACGTTAGCTCAGGAAAGGCTCTAGGCGAGTGGTGTGAG
AGTTTTTCACCAATATCGGCTCCTCTGAAAGAGTGTATAT
CAGGCTACTTTTGGACGGCTAGGCGATCGATGGTTACTCA
CCGCGTATGTCTCAATCGGGCCGGTCACAACCATCGATAG
ATCTCTCCTCGAAAAGTCTTTGCCCTCATCGACAAACTAA
AGGAATTACCACCAAACCTTAAGTTACTCATGGGCTTGCTG
GGATCAGCCTCCGCCCTGTGAAAAGGGTTTACGCCCATTA
TAGAAAATTTGAAGGAGGCTGTTACGAGAGTCATAGTTGA
GGACAATAGATCGCTGGATTTTCGTGGATGATCGCATTTTCG
CAGTATTAACCTGGTCTGTATCGCTGAGCAGCTATACCTTC
CTACACCTTCTGACCGCTGACGTTCCGACAGAATTTTGC
GTACCGGAAGCGGACCACAGACGCAAAGATGCGAGTGTG
A A C T G C C T A G C C C C C C C C C T T T T C C C C C C T C A T A A C C A A A A A A T T T

Comment savoir si une requête existe dans un jeu de données ?

Query:

TTACCACCAAACCTTAAGTTACTCATGGGCTTGCT

Sample1

AGACCTCTAAGGTGAACTTTCACGCACAGGGGAGGGAGTA
TGCACGACGTAGGATGCCGTTACGAAAACGCCCTGACCT
GTGCGGCGCCTAAAGGACAGCGACTGATTTTGGCTTTGCCA
ATTAGTATGGAGTATTTTTCATAACCCCTGCCACCTAAGG
CGCTACGCTACGGGGGCCGCACCGGTCGTGACTGGTACAG
CCGGTAATGGAACCTTAGCAGGAGTAACTGCGAACATTCCG
GGTCATGGGTAATTCTTAAAAACAGTCAGACTGGGCCCCT
GTTACGCCGATACTACTATGCTATGCGTAAGGGACGAGC
AACGTTAGCTCAGGAAAGGCTCTAGGCGAGTGGTGTGAG
AGTTTTTCACCAATATCGGCTCCTCTGAAAGAGTGTATAT
CAGGCTACTTTTGGACGGCTAGGCGATCGATGGTACTCA
CCGCGTATGTCTCAATCGGGCCGGTCACAACCATCGATAG
ATCTCTCCTCGAAAAGTCTTTGCCCTCATCGACAAACTAA
AGGAATTACCACCAAACCTTAAGTTACTCATGGGCTTGCTG
GGATCAGCCTCCGCCCTGTGAAAAGGGTTTACGCCCATTA
TAGAAAATTTGAAGGAGGCTGTTACGAGAGTCATAGTTGA
GGACAATAGATCGCTGGATTTTCGTGGATGATCGCATTTTCG
CAGTATTAACCTGGTCTGTATCGCTGAGCAGCTATACCTTC
CTACACCTTCTGACCGCTGACGTTCCGACAGAATTTTGC
GTACCGGAAGCGGACCACAGACGCAAAGATGCGAGTGTG
A A C T G C C T A G C C C C C C C C C T T T T C C C C C C T C A T A A C C A A A A A A T T T

Comment savoir si une requête existe dans un jeu de données ?

Query:

TTACCACCAAACCTTAAGTTACTCATGGGCTTGCT

kmers:

TTACCACCAAACCTTAAGTTACTCATGGGC
 TACCACCAAACCTTAAGTTACTCATGGGCT
 ACCACCAAACCTTAAGTTACTCATGGGCTT
 CCACCAAACCTTAAGTTACTCATGGGCTTG
 CACCAAACCTTAAGTTACTCATGGGCTTGC
 ACCAAACCTTAAGTTACTCATGGGCTTGCT

5 kmers sur 6 dans le Sample 1: ~83% de similarité

Sample1

AGACCTCTAAGGTGAACTTTCACGCACAGGGGAGGGAGTA
 TGCACGACGTAGGATGCCGTTACGAAAACGCCCTGACCT
 GTGCGGCGCCTAAAGGACAGCGACTGATTTTGCTTTGCCA
 ATTAGTATGGAGTATTTTTCATAACCCCTGCCACCTAAGG
 CGCTACGCTACGGGGGCCGCACCGGTCGTGACTGGTACAG
 CCGGTAATGGAACCTTAGCAGGAGTAACTGCGAACATTCCG
 GGTCATGGGTAATTCTTAAAAACAGTCAGACTGGGCCCT
 GTTCACGC **TTACCACCAAACCTTAAGTTACTCATGGGCTTG**
 TACTACTATGCTATGCGTAAGGGACGAGCAACGTTAGCTC
 AGGAAAGGCTCTAGGCGAGTGGTGTGAGAGTTTTTCACC
 AATATCGGCTCCTCTGAAAGAGTGTATATCAGGCTACTTT
 TGGACGGCTAGGCGATCGATGGTTACTCACCGCGTATGTC
 TCAAT **ACCAAACCTTAAGTTACTCATGGGCTTGCT**CGGGCC
 GGTCACAACCATCGATAGATCTCTCCTCGAAAAGTCTTTG
 CCCTCATCGACAAACTAAAGGAATTACCACCAAACCTTAAG
 TTACTCATGGGCTTGCTGGGATCAGCCTCCGCCCTGTGAA
 AAGGGTTTACGCCCATATAGAAAATTTGAAGGAGGCTGT
 TACGAGAGTCATAGTTGAGGACAATAGATCGCTGGATTTT
 GTGGATGATCGCATTTCGCAGTATTAACCTGGTCTGTATCG
 CTGAGCAGCTATACCTTCCTACACCTTCCTGACCGCTGAC
 CTTCCCAAGCAATTTTTCGCTACGCCCAAGCCCAAGCAAGCA

Comment savoir si une requête existe dans un jeu de données ?

Sample1

TTACCACCAAACCTTAAGTTACTCATGGGCTTGCT

kmers:

TTACCACCAAACCTTAAGTTACTCATGGGC
TACCACCAAACCTTAAGTTACTCATGGGCT
ACCACCAAACCTTAAGTTACTCATGGGCTT
CCACCAAACCTTAAGTTACTCATGGGCTTG
CACCAAACCTTAAGTTACTCATGGGCTTGC
ACCAAACCTTAAGTTACTCATGGGCTTGCT

5 kmers sur 6 dans le Sample 1: ~83% de similarité



Comment savoir si une requête existe dans un jeu de données ?

Sample1

Question fondamentale :

Est-ce que ce kmer

TTACCACCAA ACTTAAGTTACTCATGGGC

...est dans le sac ?



Comment savoir si une requête existe dans un jeu de données ?

Sample1

Index

TTACCACCAAAGTTAAGTTACTCATGGGC



- OUI
- NON

Comment savoir dans quel jeu de données une requête existe?

Sample1

Index

TTACCACCAAACCTTAAGTTACTCATGGGC



Set

- 12
- 28
- ...
- 18, 234, 234

Utilisation de structures de données

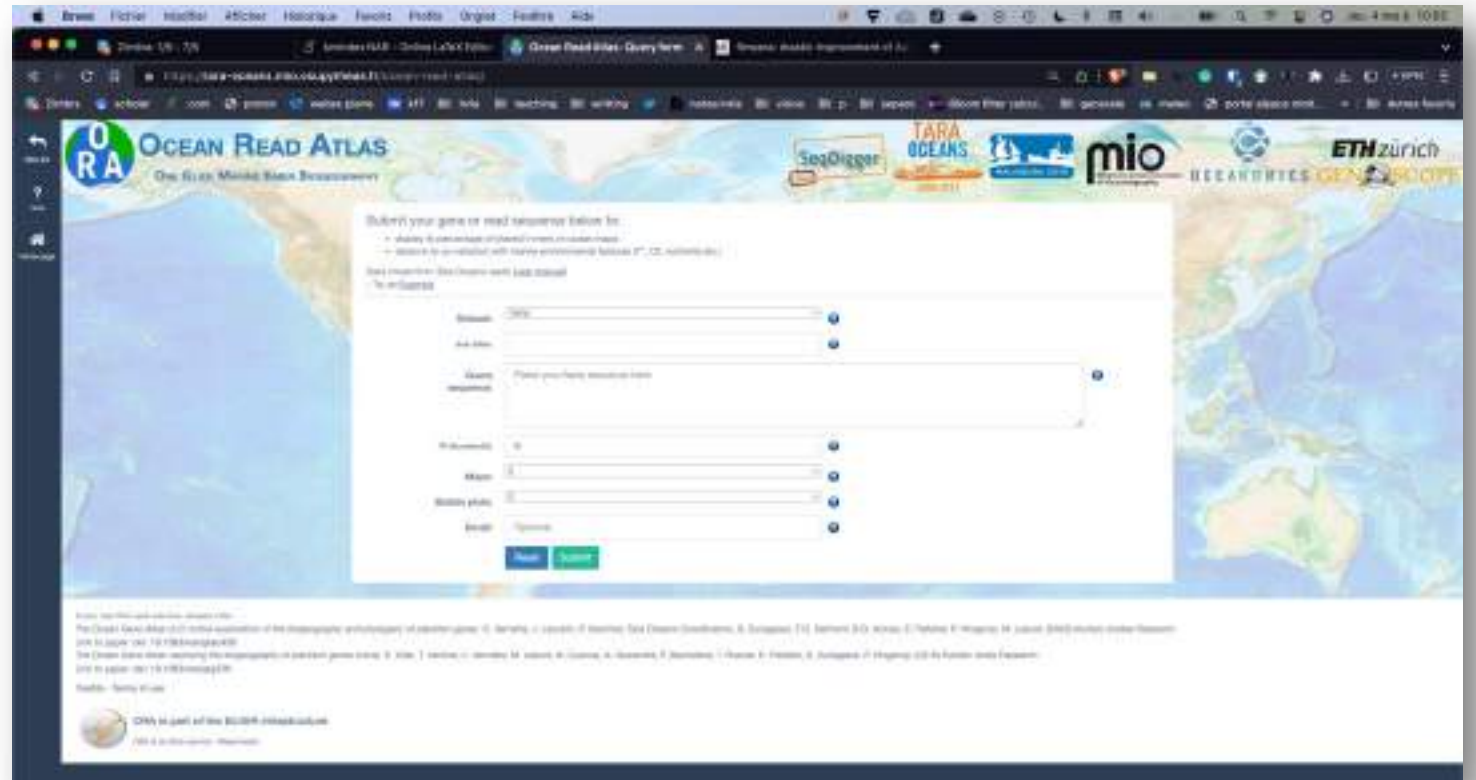
- 😊 Légères en mémoire
- 😊 Rapides à créer & requêter
- 😞 Inexactes: faux positifs

ORA Server

<https://ocean-read-atlas.mio.osupytheas.fr/>

Index: all Tara Ocean Metagenomic samples (no abundance yet)

- Input fastq.gz files
 - 6TB
 - 1,393 samples
- Final index size: 0.6TB



LOGAN & kminindex

48.2 petabases
of raw reads
(SRA)



tlemane / kminindex



Microsoft
Azure machines



SRA Indexed
Index size <1 petabyte
Tunable web server

LOGAN & kminindex

48.2 petabases
of raw reads
(SRA)



tlemane / kminindex



Microsoft
Azure machines



SRA Indexed
Index size <1 petabyte
Tunable web server


POC on
genomic primates

1.5 millions accessions
~430,000 billion kmers

Build: 4h30 (20k vCPUs)
Index: 101.5 terabytes
Query (100bp): 55s

LOGAN & kminindex

48.2 petabases
of raw reads
(SRA)

 tlemane / kminindex

Microsoft
Azure machines

SRA Indexed
Index size <1 petabyte
Tunable web server

POC on
genomic primates

1.5 millions accessions
~430,000 billion kmers

Build: 4h30 (20k vCPUs)
Index: 101.5 terabytes
Query (100bp): 55s

```
process katricks {
  container 'tlemane/katricks:zstd_azu'
  errorStrategy 'ignore'
  // memory = [ def mem = index.getSimpleName().tokenize("_")[1] as int; mem < 34 ? 128.GB : 256.GB ]

  machineType = "Standard_D4d_v5"
  cpus = [ def mem = index.getSimpleName().tokenize("_")[1] as int; mem < 34 ? 96 : 96 ]
  memory = [ def mem = index.getSimpleName().tokenize("_")[1] as int; mem < 34 ? 384.GB : 384.GB ]
  tag "$index"

  input:
  tuple path(index), path(runid_to_hash)

  output:
  path "${index.baseName}"

  script:
  ===
  export AWS_EC2_METADATA_DISABLED=true

  retry() {
    local n=1
    local max=5

    while true; do
      "$@" && return 0 || {
        if [[ $n -lt $max ]]; then
          ((n++))
          echo "Command failed. Attempt $n/$max:"
        else
          echo "The command has failed after $n attempts."
          return 1
        fi
      }
    }
  }
}
```

Merci <3

