

# Pattern Matching versus Covid

Philippe Jacquet (IEEE Fellow)

Inria, France

[Philippe.jacquet@inria.fr](mailto:Philippe.jacquet@inria.fr)

# Definition

- Consider a finite alphabet  $A$ .
  - Eg  $A = \{a, b\}$
  - $A^*$  is the set of finite texts written with symbols in  $A$ .
    - $A^* = \{\epsilon, "a", "b", "aa", \dots\}$
- A text  $w$  is factor of a text  $X$  if  $X = uwv$  with  $u, v \in A^*$ 
  - $w = "aba", X = "abba\underline{abaa}"$ ,  $X = "\underline{aba}abb"$ ,  $X = "\underline{ababa}"$
- Pattern matching problem:
  - Given  $w$  and  $X$ , determine if  $w$  is factor of  $X$ .
- Application
  - Parsing, compilation, search, edition. compression, etc.

# Probabilistic model on sequence

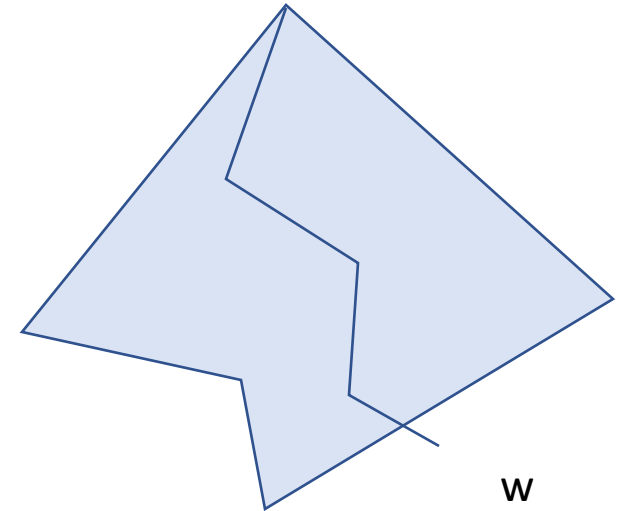
- The simplest: memoryless source,  $P(X)$  = product of symbol probabilities
  - $|X|$  length of sequence  $X$ .
  - Number of occurrences of  $w$  in  $X$ :  $|X|_w$
  - $E[|X|_w] = (|X| - |w|)P(w)$ .
  - When  $|w| = o(\sqrt{|X|})$  then  $E[|X|_w] \sim |X|P(w)$
- Straightforward extension for Markov sequence generation with finite memory

# Word sequence matching problem

- Let  $(w_1, w_2, \dots, w_k)$  be a sequence of different words and  $X$  be a text.
  - We want the  $w_i$  to be factors of  $X$  and appear in this order in  $X$ .
  - Let  $|X|_{(w_1, \dots, w_k)}$  be number of such occurrences in  $X$ 
    - $E[|X|_{(w_1, \dots, w_k)}] = \binom{|X| - |w_1| - \dots - |w_k|}{k} P(w_1) \dots P(w_k)$
    - When  $|w_1| + \dots + |w_k| = o(\sqrt{|X|})$ , then  $E[|X|_{(w_1, \dots, w_k)}] \sim \frac{|X|^k}{k!} P(w_1) \dots P(w_k)$
- If the order is indifferent
  - $E[|X|_{\{w_1, \dots, w_k\}}] \sim |X|^k P(w_1) \dots P(w_k)$

# The suffix/prefix trees

- useful data structures for linear pattern matching.
  - All suffixes (resp prefixes) are stored in a tree structure
  - Longest copy of  $w$  found by exploration of the tree of  $X$
- Computation of the suffix tree:
  - Naïve algorithm in  $|X|\log |X|$
  - Rissanen algorithm in  $|X|$
  - Search in tree is in  $\log |X|$

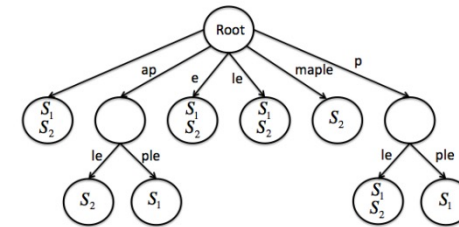


# Joint Complexity

- Definition: number of common factors between two texts
  - $\epsilon, a, e, i, n, s, t, \_, a\_, an, is, na, s\_, ti, \_a, \_e, \_i, ana, is\_, nan, \_is, anan, nana, \_is\_,anana$
  - $J(X1,X2)=25$
  - Fast to compute: linear via suffix trees superposition.
- Real time classification and trend propagation tracking
  - Agnostic of language
  - Resilient to mispealing

X1=The ape is eating a banana

X2=Ananas is an exotic fruit

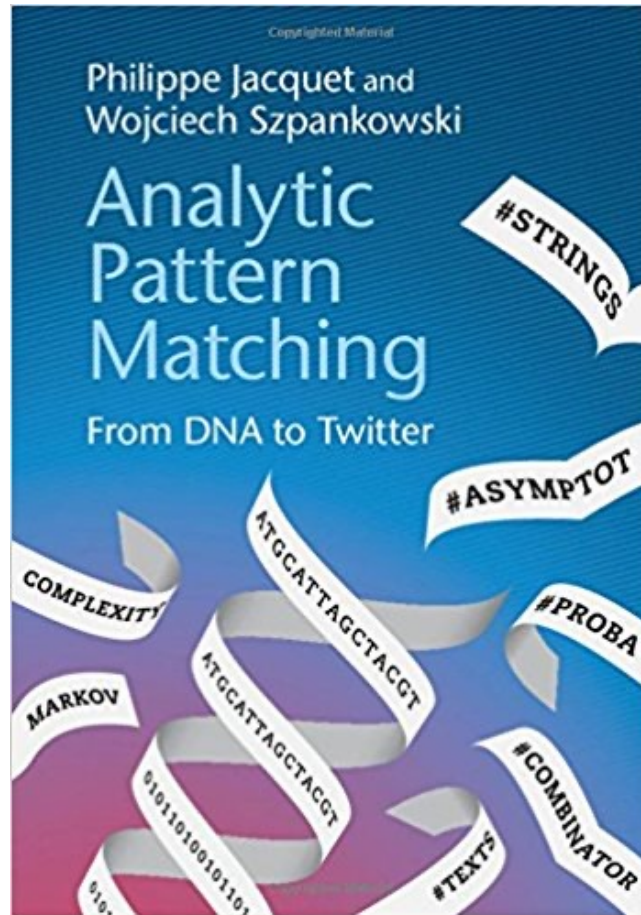


# Theoretical performance

- When texts  $X$  and  $Y$  are of same length but from sources with different parameters
  - $E[J(X, Y)] \sim \frac{|X|^\kappa}{\sqrt{a \log |X| + b}}$  with  $\kappa < 1$
- When texts are of different lengths but from sources with same parameters (eg Markov with same transition matrix)
  - $E[J(X, Y)] \sim \frac{(|X|+|Y|) \log(|X|+|Y|) - |X| \log |X| - |Y| \log |Y|}{h}$

**NEW RESULT!**

# Bibliography and theoretical corpus



Jacquet, P. (2007, June). Common words between two random strings. In *2007 IEEE International Symposium on Information Theory* (pp. 1481-1485).

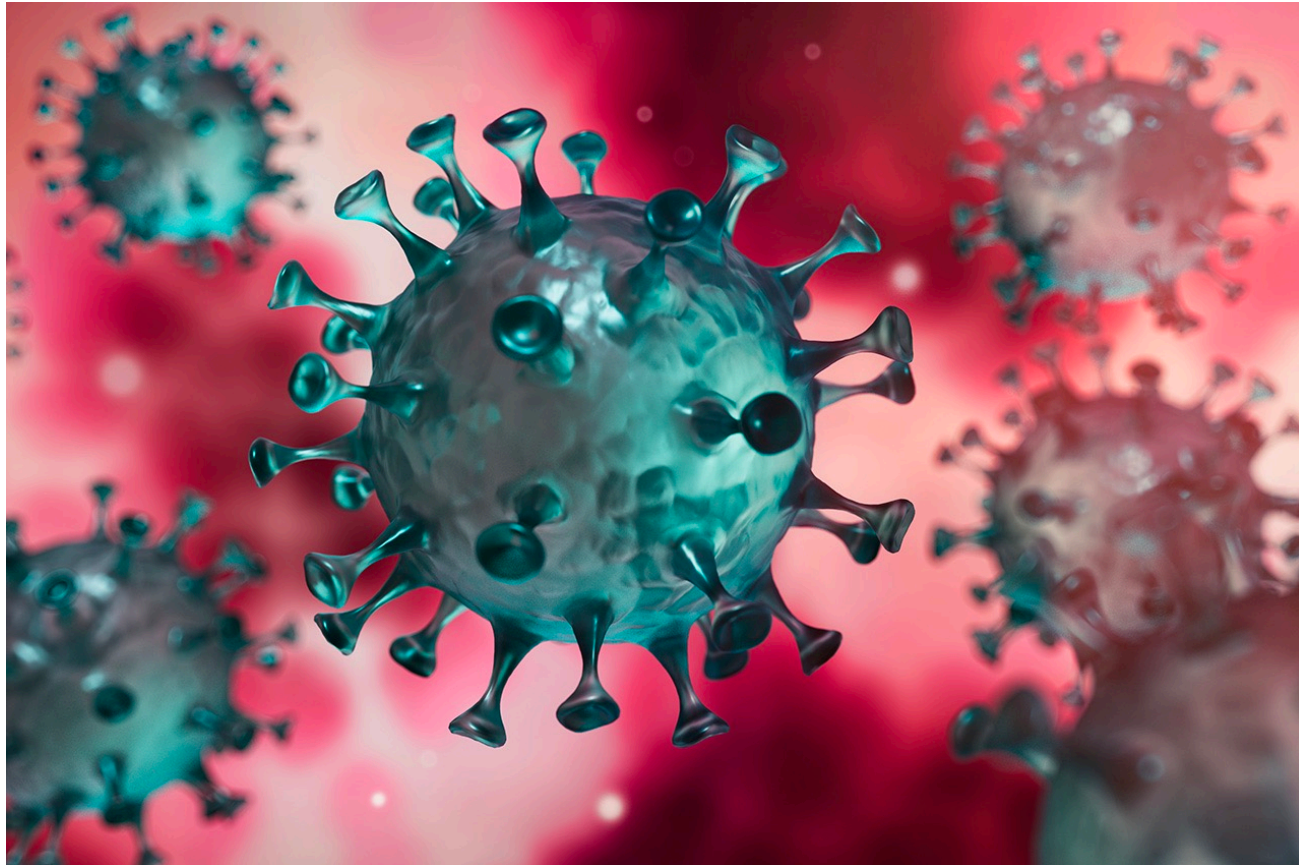
Jacquet, P., Mioris, D., & Szpankowski, W. (2013, July). Classification of Markov sources through joint string complexity: Theory and experiments. In *2013 IEEE International Symposium on Information Theory* (pp. 2289-293).

Mioris, D. (2018). Joint Sequence Complexity: Introduction and Theory. In *Topic Detection and Classification in Social Networks* (pp. 21-56). Springer.

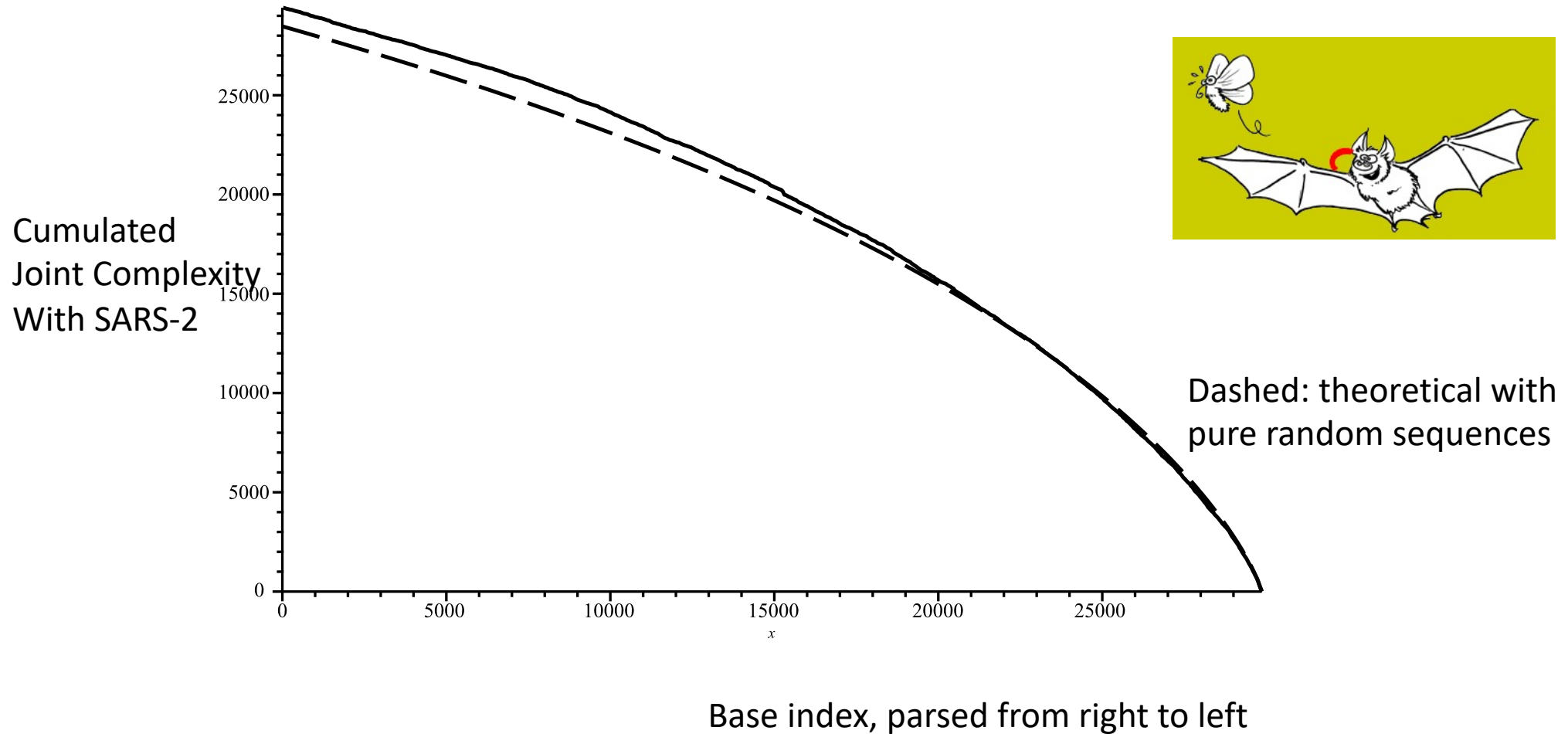




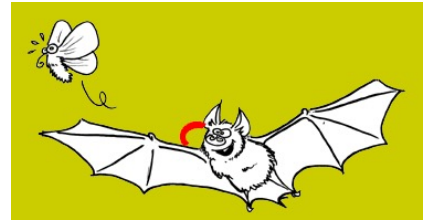
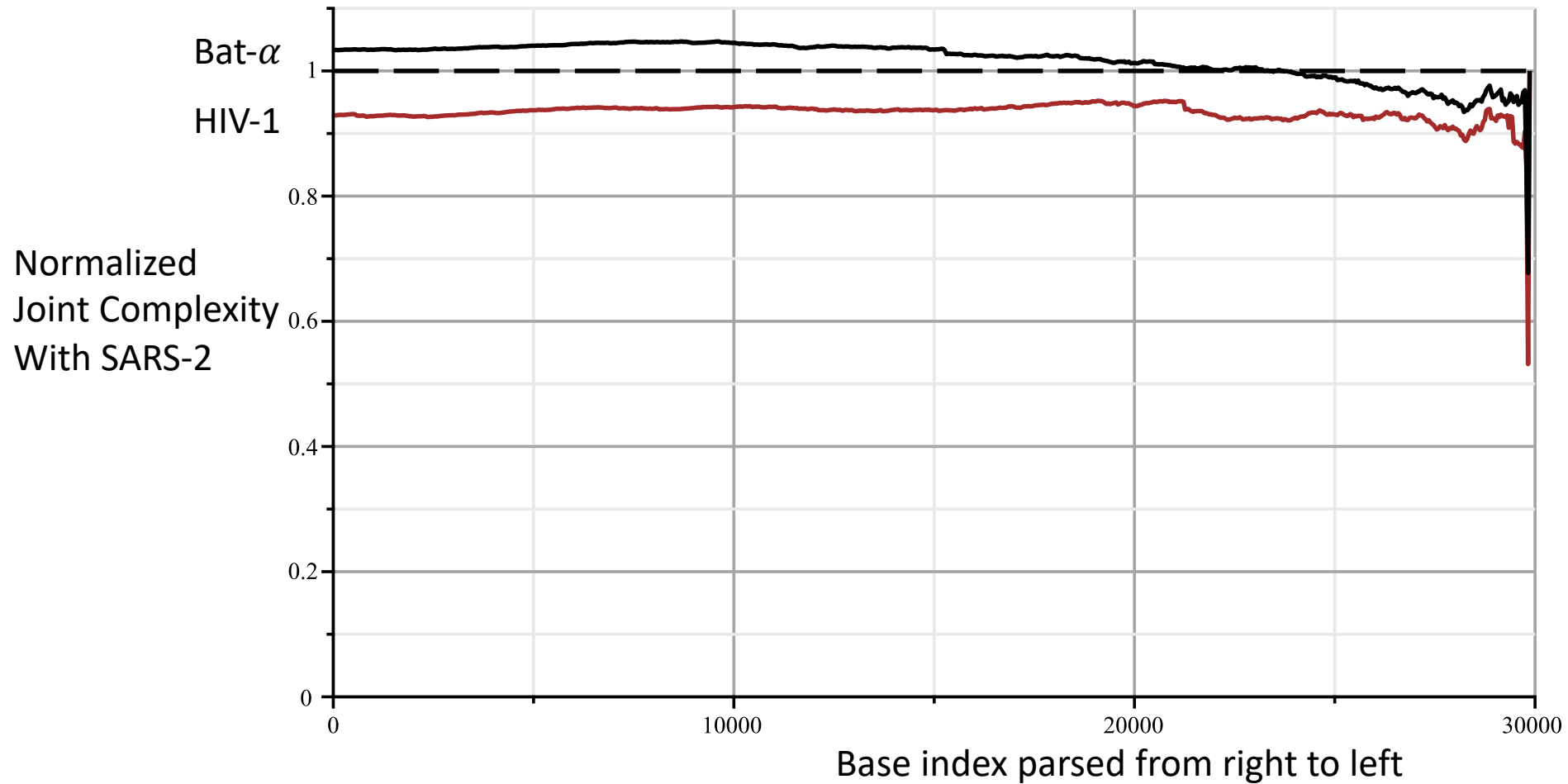
# Coronavirus SARS-2 sequence



# Joint Complexity of SARS-2 versus Bat- $\alpha$ Coronavirus (2007)

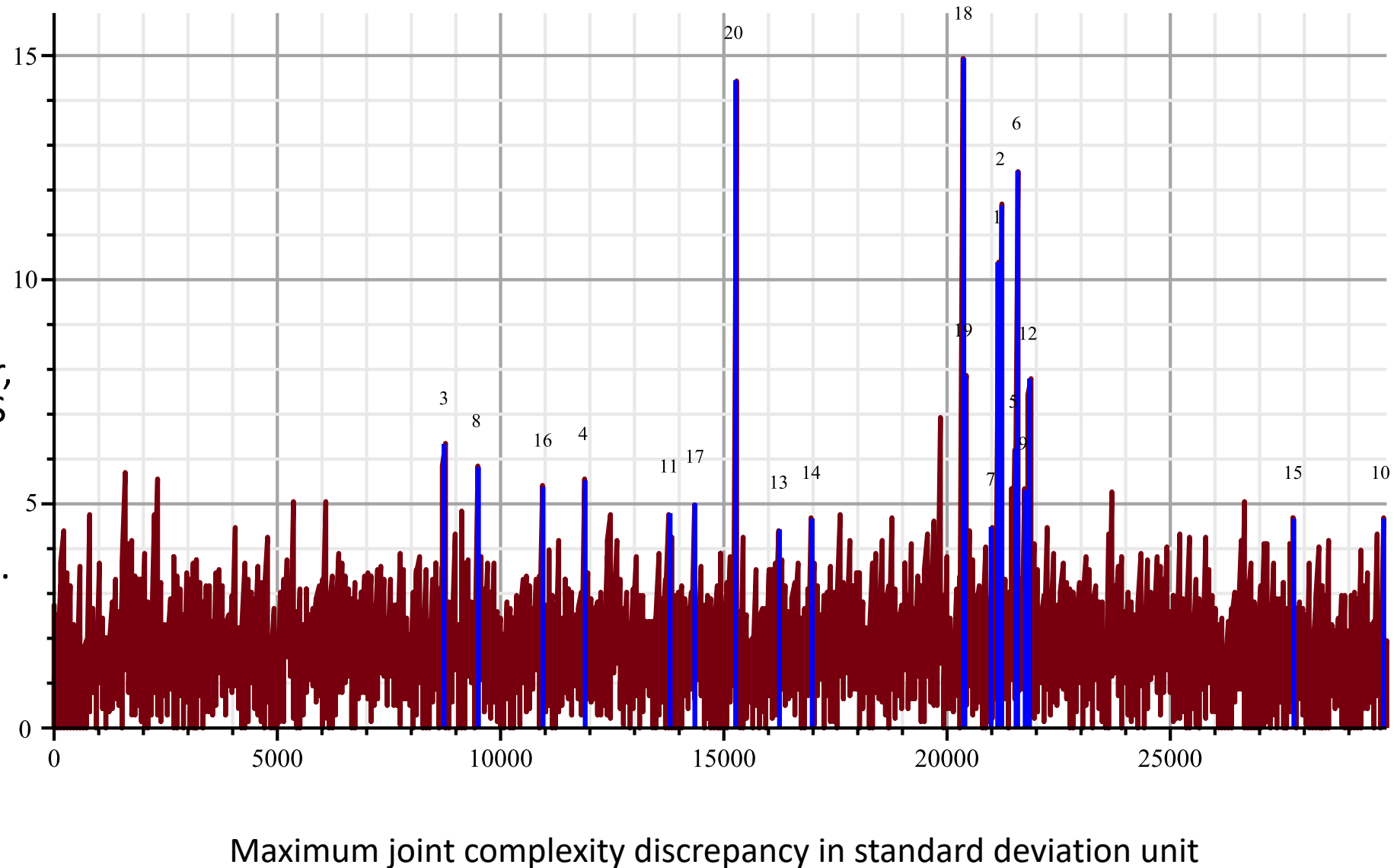


# SARS-2 versus Bat and HIV



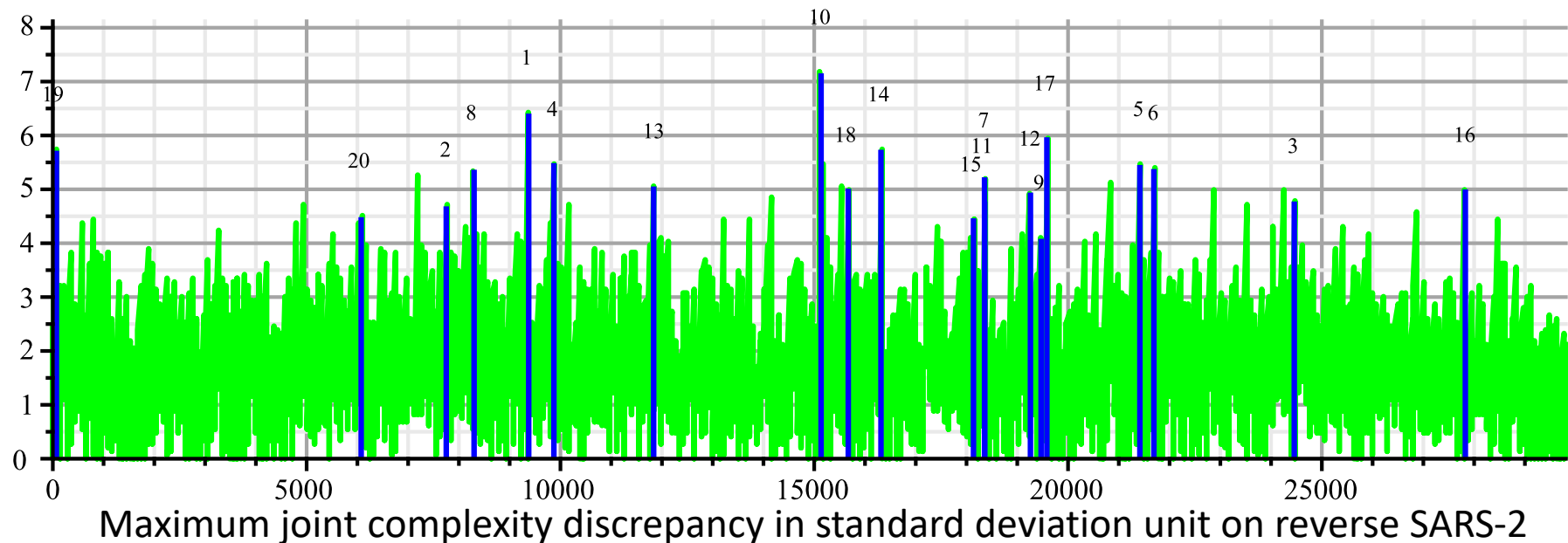
# “Accidental” Insertions of HIV in SARS-2 genome

- Despite not related, there are 19 short common factors of HIV sequence in SARS-2 genome
  - From 19 factors of  $20\overline{7}$  bases each
  - 20th is the bat- $\alpha$  virus
  - Perez, J. C., & Montagnier, L. (2020). COVID-19, SARS and Bats Coronaviruses Genomes Unexpected Exogenous RNA Sequences. *OSF Preprints*.



# Matching probabilities

- If the factors are fixed, then the probability of such accidental insertions will be smaller than  $30,000^{19} 4^{-19 \times 20} \approx 2.10^{-144}$ !
  - This is the word sequence problem
- Perez-Montagnier conclusion: the SARS-2 Cov virus has been “forged”.

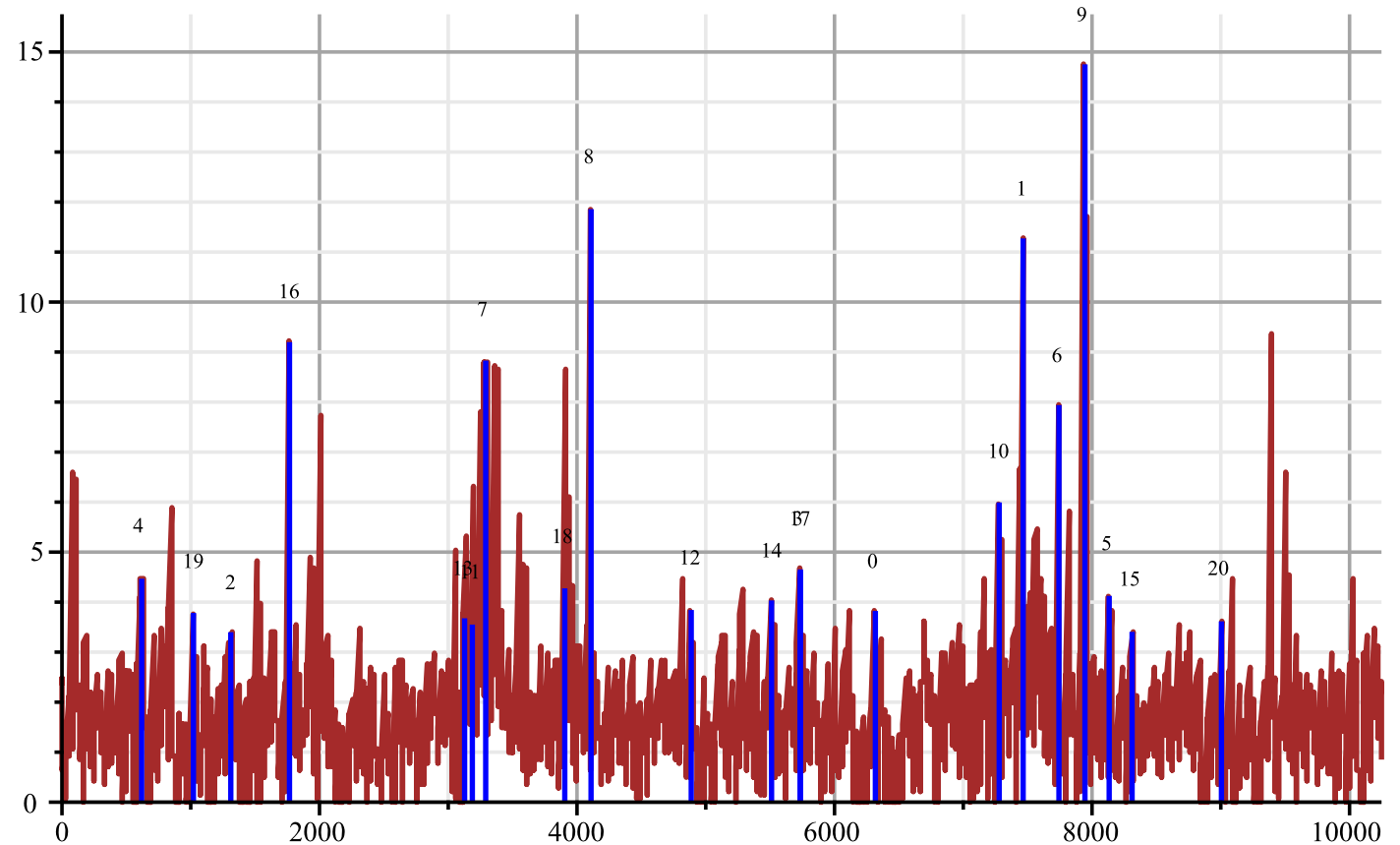


# Weak pattern matching

- Let unfix the matching factors
- Assume a corpus of  $M$  bases (probably around  $10^6$ ) about HIV sequences (simian, HIV-1, HIV-2 and reversed sequences)
  - Average number of matchings of the corpus over a genome of  $N$  bases is  $MN \times P(\text{matching})$ .
  - $P(k \text{ matchings}) < \frac{MN \times 4^{-20}}{k} \approx 1.4 \cdot 10^{-3}$  for  $k=19$  and  $M = 10^6$ ,  $N = 30,000$  and  $P(\text{matching}) = 4^{-20}$ .
    - Tolerance for max 3 errors multiplies by  $\binom{20}{3}$ :
  - $P(19 \text{ matchings}) < 1.6$

# Weak matching: The matching between the matchers

- The segment HIV-2-UCI1 (segment 11) versus the other segments
- Similar accidents
  - Is it “forged” (it is from 1993)?

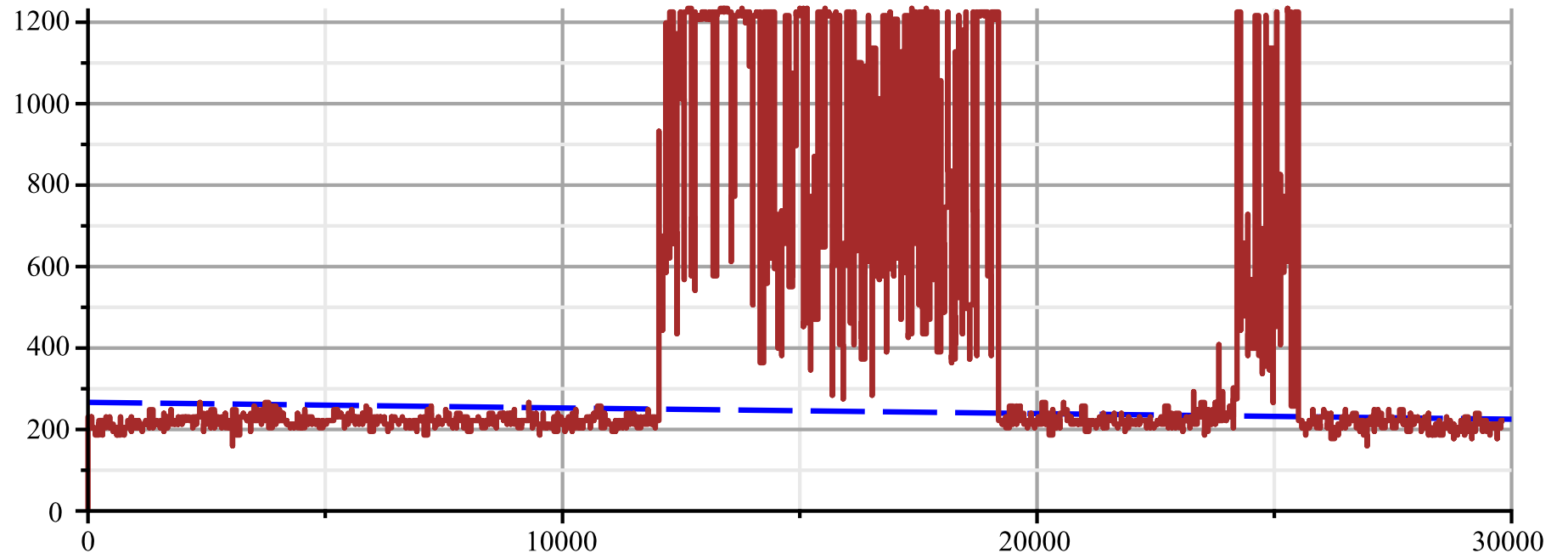




# Strong pattern matchings

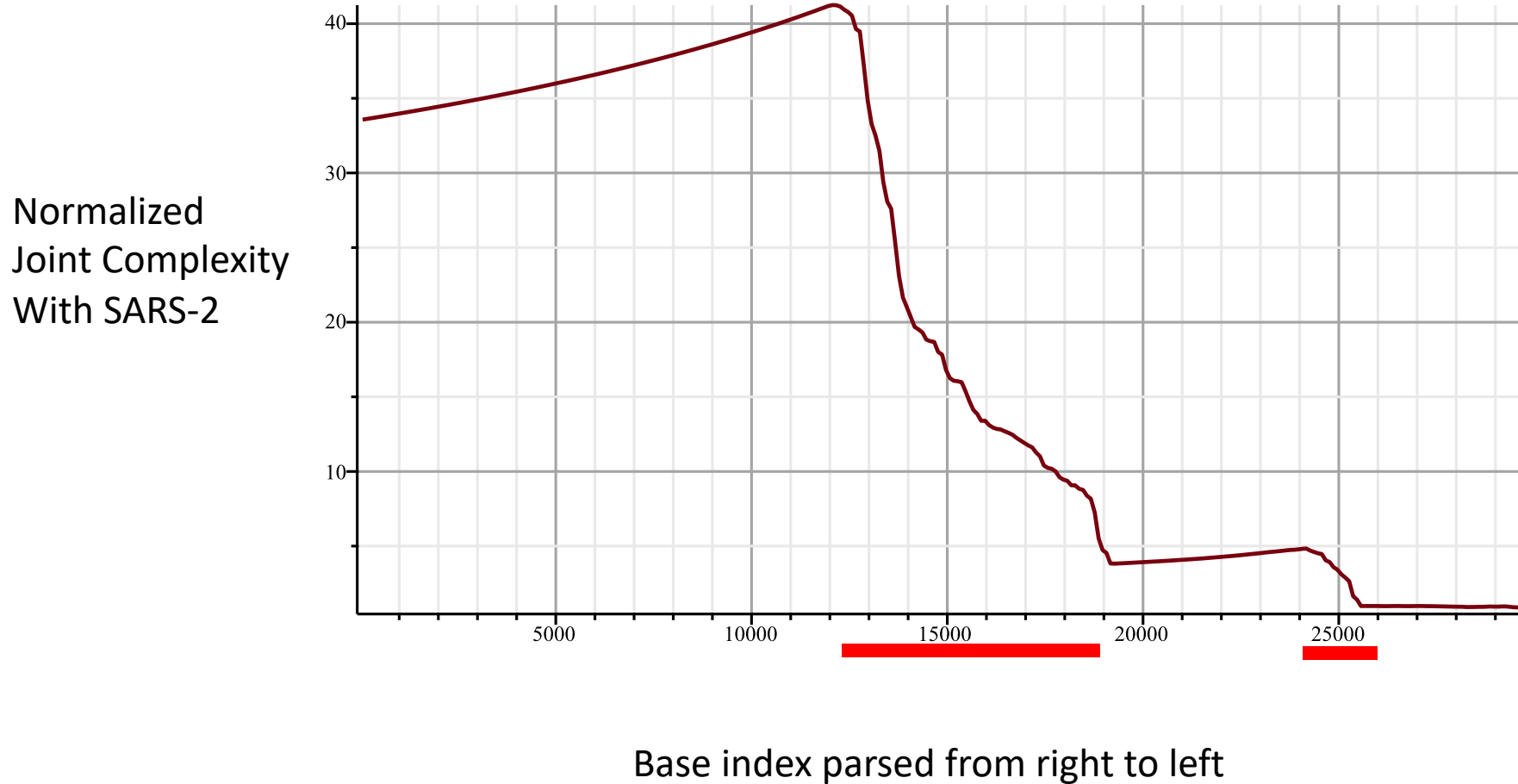
- RmYN02 “Closely Related to SARS-CoV-2” (Nov 2020)

Zhou H, Chen X, Hu T, Li J, Song H, Liu Y, Wang P, Liu D, Yang J, Holmes EC, Hughes AC, Bi Y and Shi W.: A Novel Bat Coronavirus Closely Related to SARS-CoV-2 Contains Natural Insertions at the S1/S2 Cleavage Site of the Spike Protein JOURNAL Curr Biol 30 (11), 2196-2203 (2020)



Sliced joint complexity discrepancy (50 bp slices) of RmYN02 over SARS-2

# Strong pattern matchings (continued)

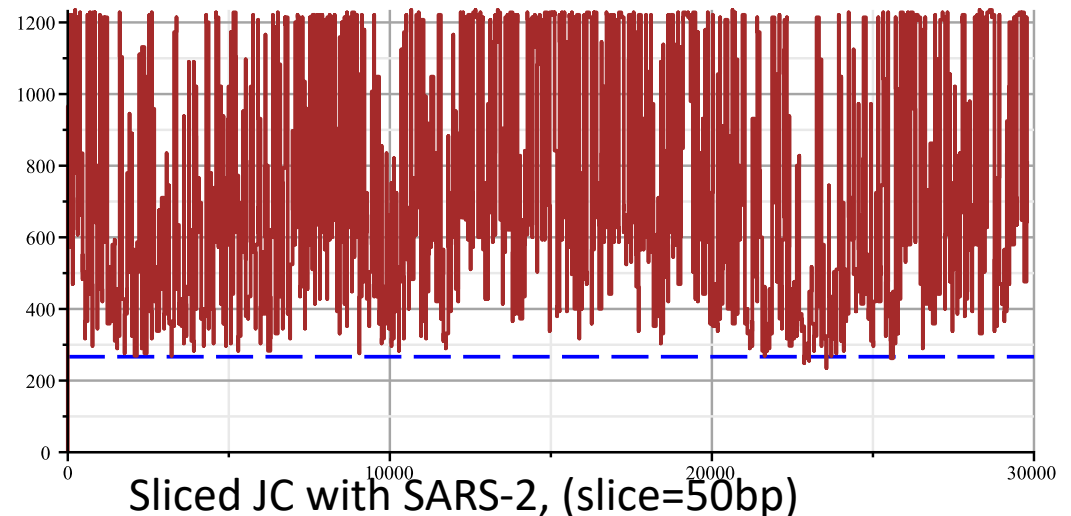
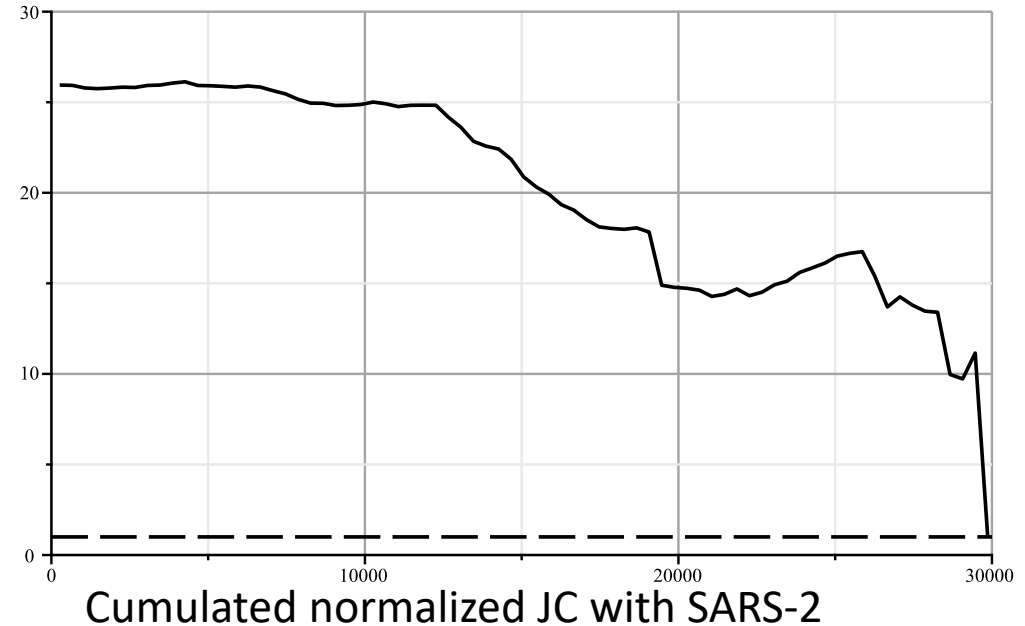


# SARS-2 Covid closest known ancestor

- RaTG13 Bat Coronavirus
  - Found in 2013 in a cave in Yunnan

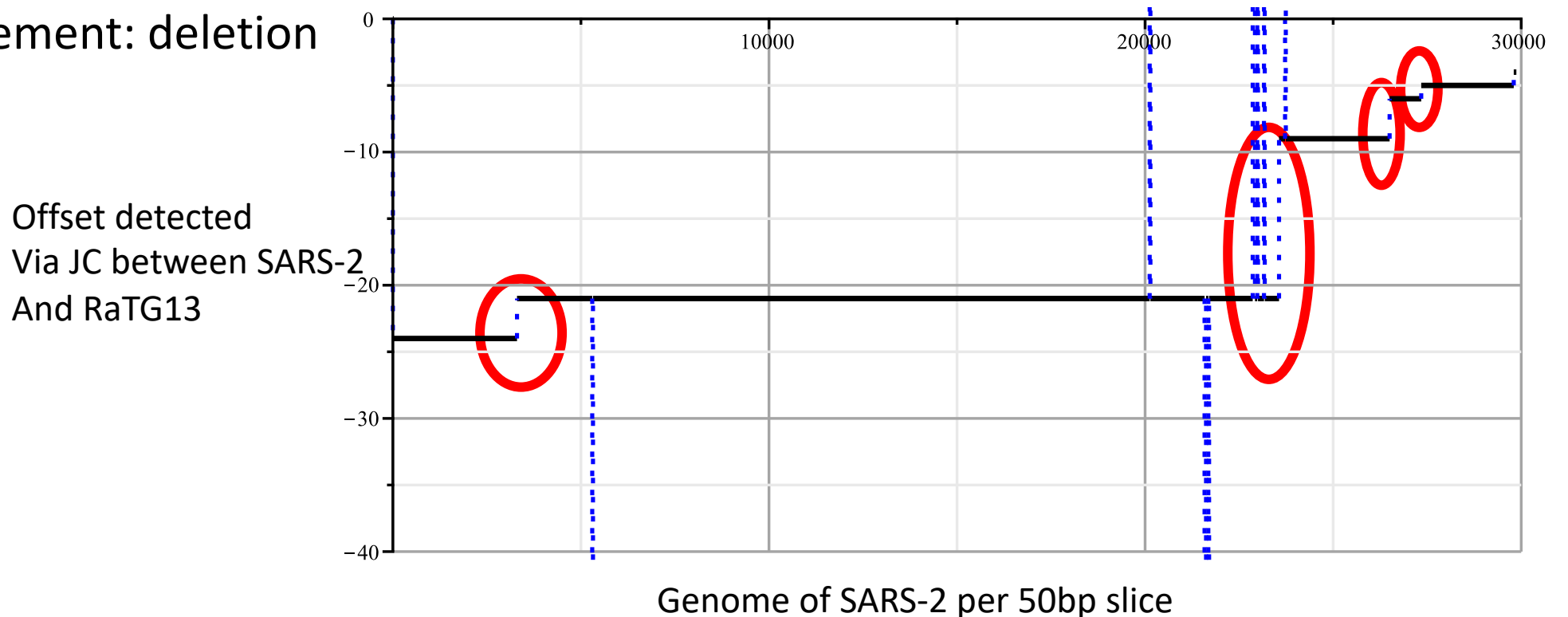


- Six years after Bat- $\alpha$ , low matching.
- Strong matching with SARS-2, six years later.



# insertion in SARS-2 from RaTG13

- with largest common factor, JC can detect the offsets between sequences
  - Offset increment: insertion
  - Decrement: deletion



Thank you

*Stay at home*  
and  
*Kill Coronavirus!*



*Safa*