Neural networks for likelihood-free inference in evolutionary genomics

Laurent Jacob

Phylogeny and Cophylogeny: Tree for a Tango, November 5th 2024

Acknowledgement

Samuel Alizon

Luc Blassel

Bastien Boussau

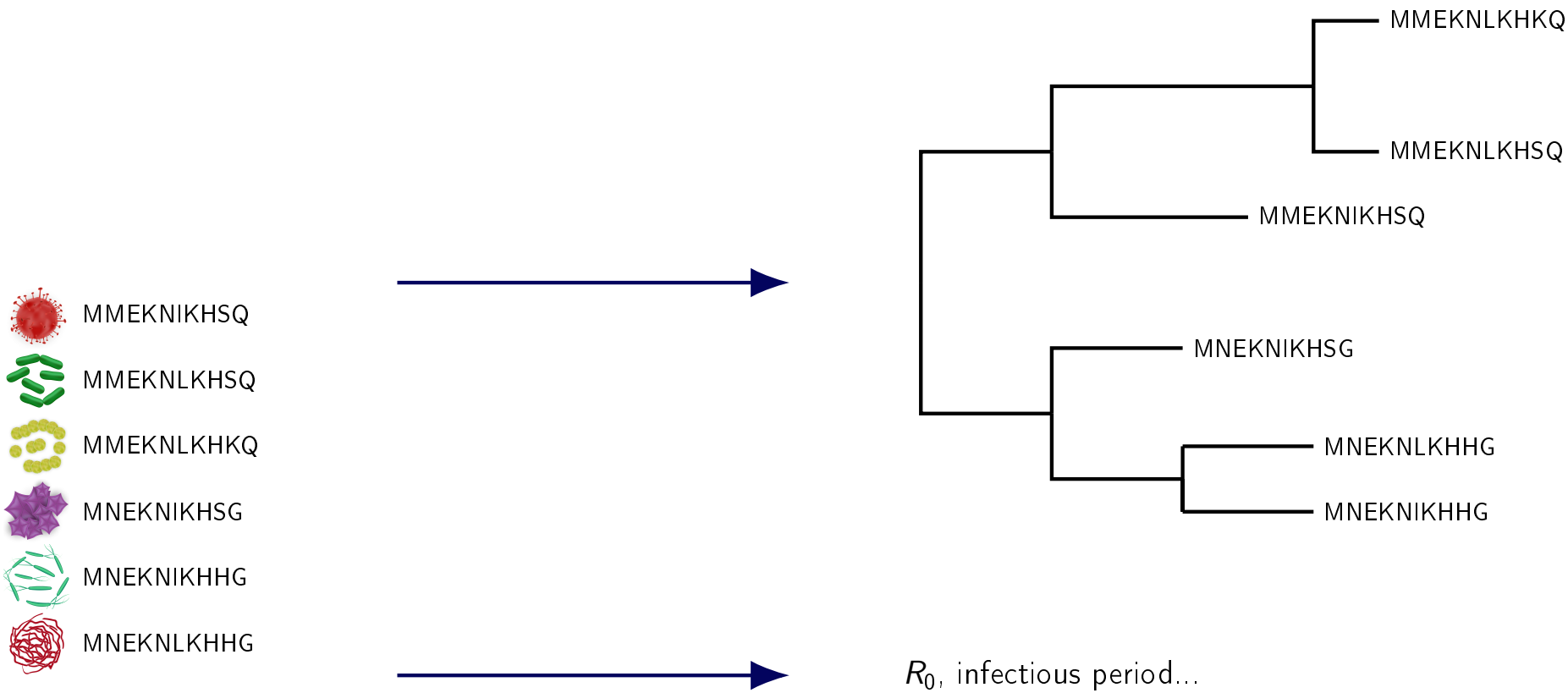
Vincent Garot

Luca Nesterenko

Johanna Trost

Anna Zhukova

Inference in evolutionary genomics



- **Observe** homologous sequences.
- **Infer** their evolutionary history: phylogeny, reproduction number...

Relies on probabilistic models that relate data to parameters.

Statistical inference

Model $p(\text{sequences}|\text{tree})$
Observed sequences
prior $p(\text{tree})$ (optional)

$\rightarrow$

Point estimate $\widehat{\text{tree}}$
or
posterior $p(\text{tree}|\text{sequences})$

Likelihood-based inference

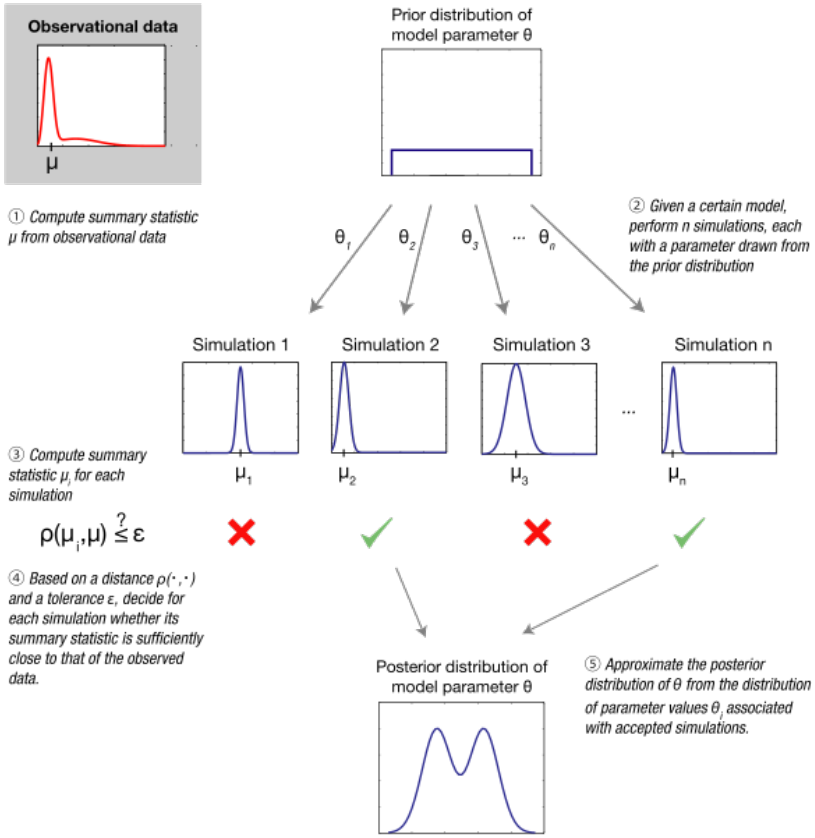
- Maximum likelihood: $\widehat{\text{tree}} = \arg\max_{\text{tree}} p(\text{sequences}|\text{tree})$.
- Estimate or sample from the posterior $p(\text{tree}|\text{sequences})$ (typically also involves computing $p(\text{sequences}|\text{tree})$).

Likelihood-**free** inference

- Realistic models:
  computing $p(\text{sequences}|\text{tree})$ is expensive.
- But *sampling* from it can be cheap.

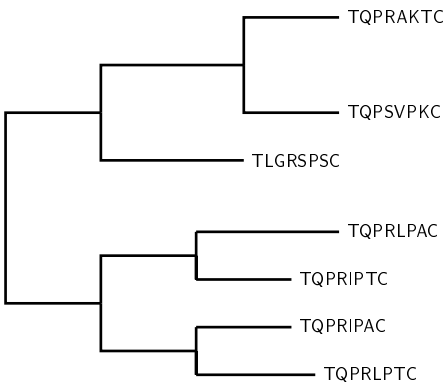
Likelihood-**free** inference

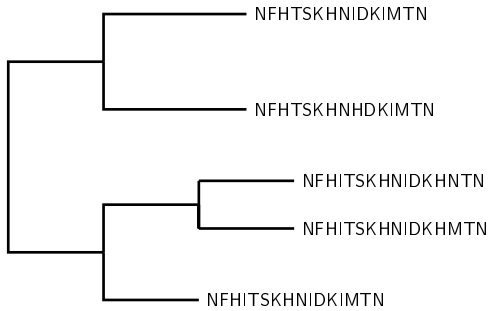
- Idea: perform inference by **sampling**, and not **evaluating** $p(\text{sequences}|\text{tree})$.
- Example: Approximate Bayesian Computation (ABC)



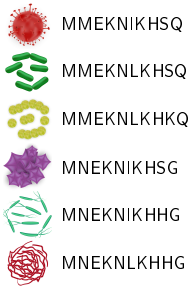
From Sunnåker *et al.* 2013

Amortized, likelihood-free neural inference

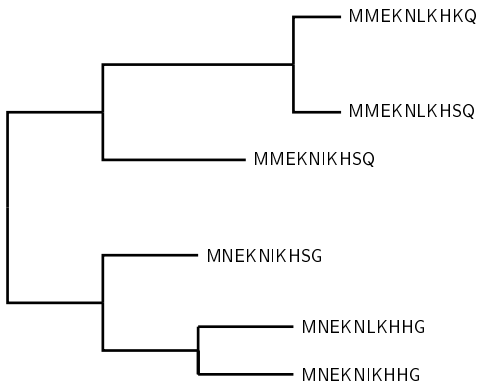




Simulate examples of:
Trees
Evolved sequences



Learn

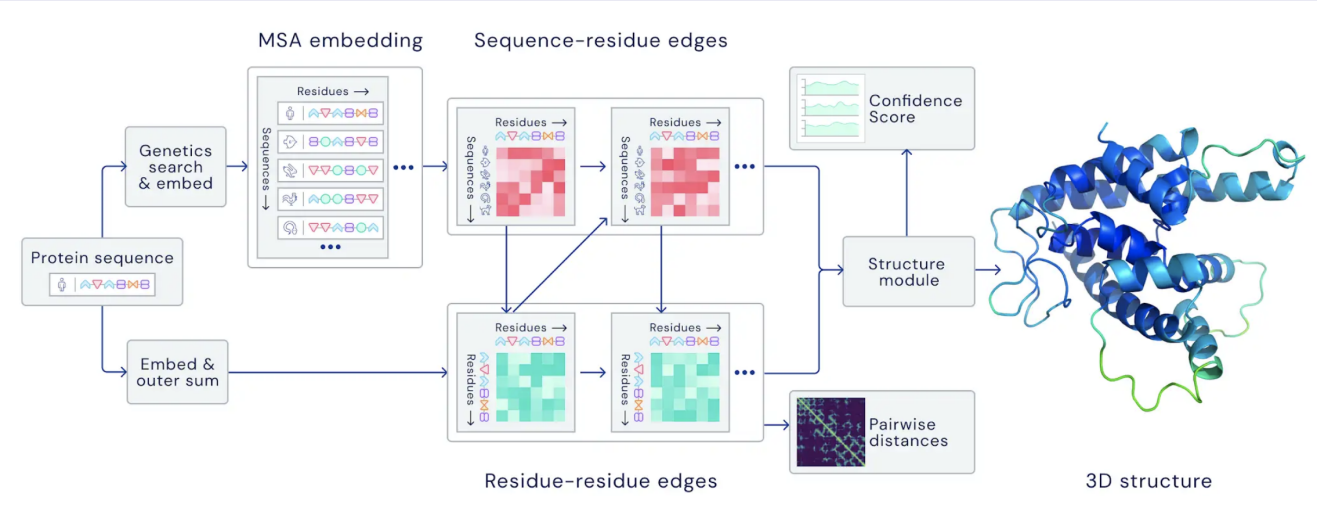


Compared to ABC:

- No rejection.
- No summary statistics.

Unusual setting for supervised learning

Ordinarily used for induction on real-world data



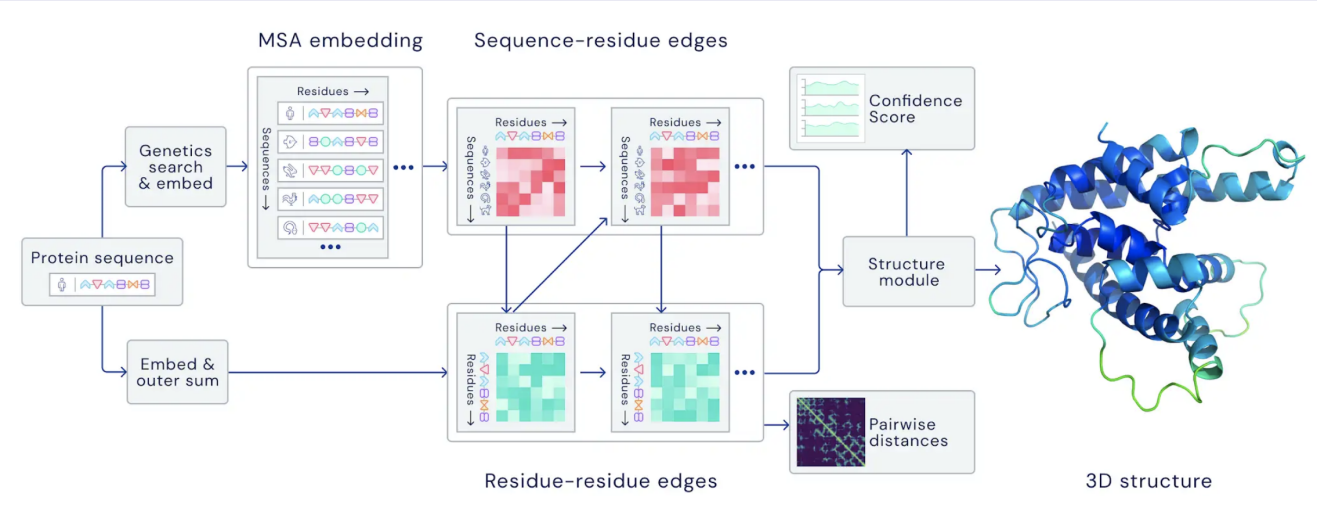
(adapted from Jumper *et al.*, 2021)

Common misconceptions

- Proxy "before we get real data"?
- "What if your model is off"?

Unusual setting for supervised learning

Ordinarily used for induction on real-world data

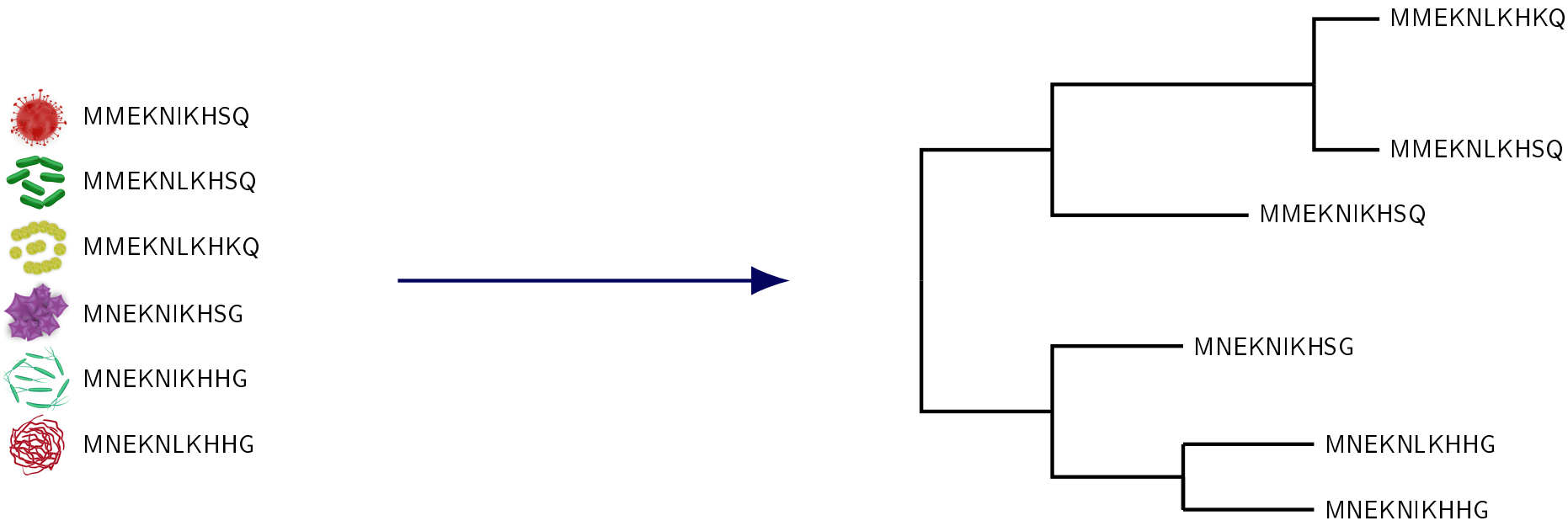


(adapted from Jumper *et al.*, 2021)

Common misconceptions

- Proxy "before we get real data"?
  → simulated data is just our way to access the model.
- "What if your model is off"?
  → Valid concern, but not specific to neural estimation.

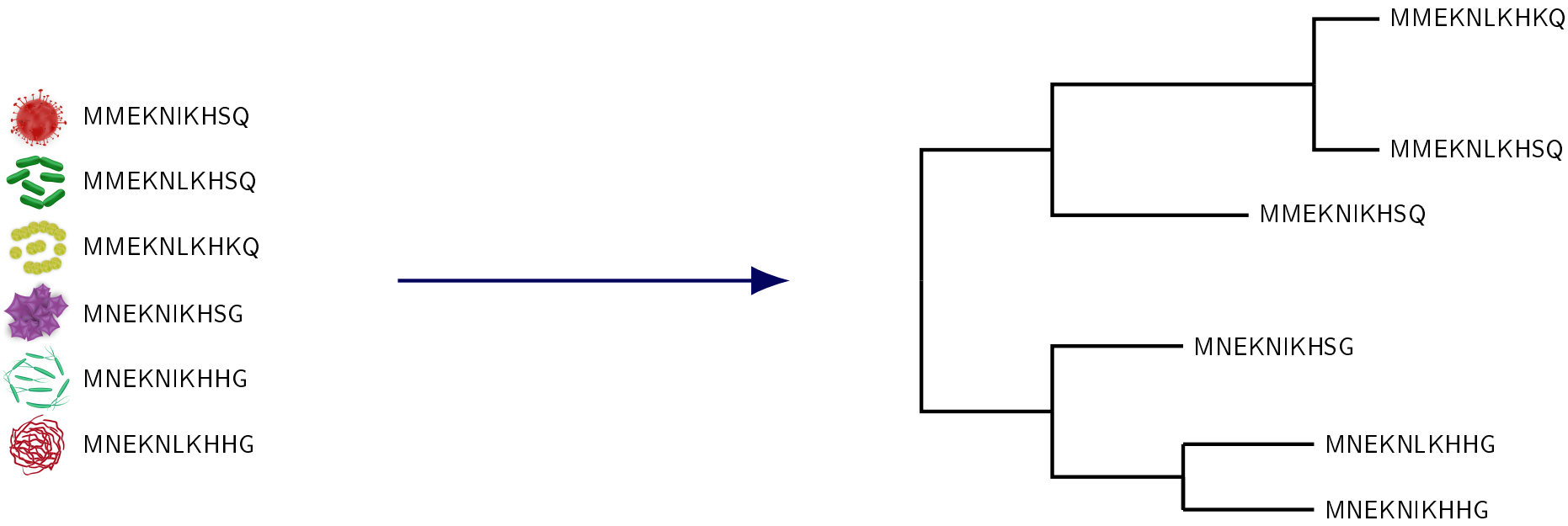
Neural inference for phylogenetics with Phyloformer



We need a learnable function that:

- outputs a phylogenetic tree,
- takes as input a set of homologous sequences (MSA)

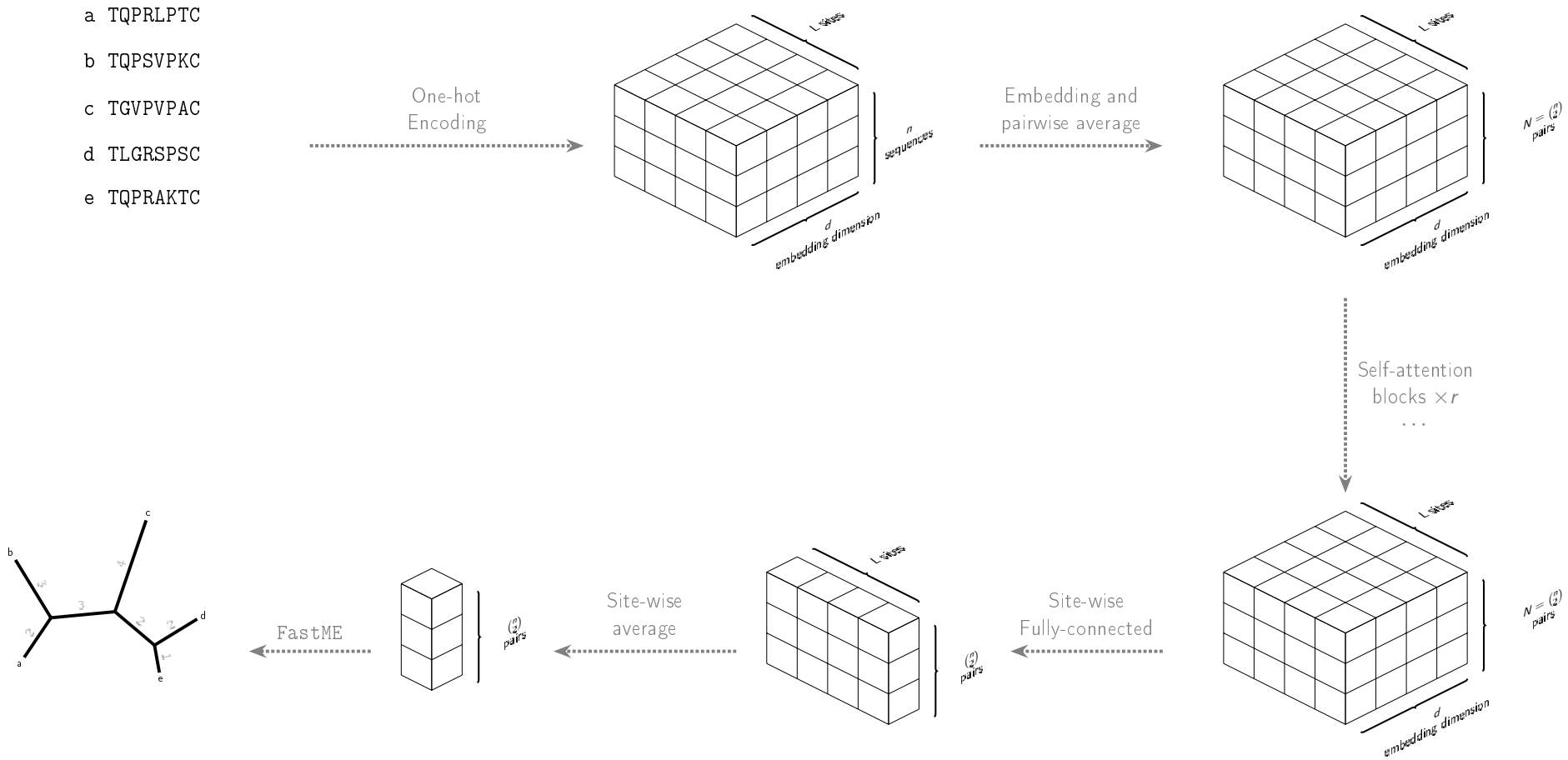
Neural inference for phylogenetics with Phyloformer



We need a learnable function that:

- outputs a phylogenetic tree,
  $\rightarrow$ **use evolutionary distances as a proxy.**
- takes as input a set of homologous sequences (MSA)
  $\rightarrow$ **use self-attention**.

Phyloformer overview

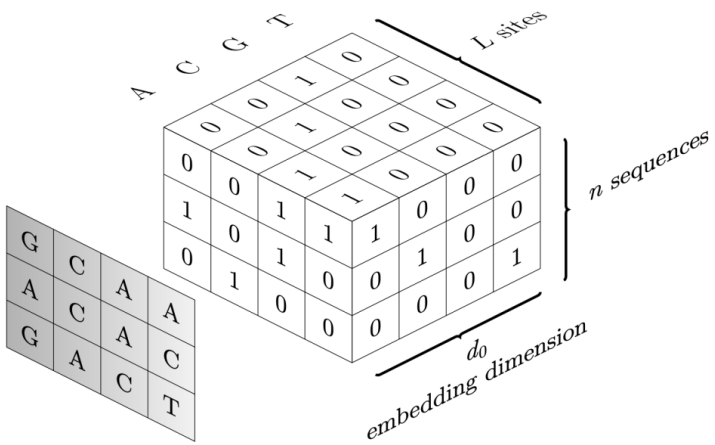


One-hot encoding for aligned sequences

A single sequence:

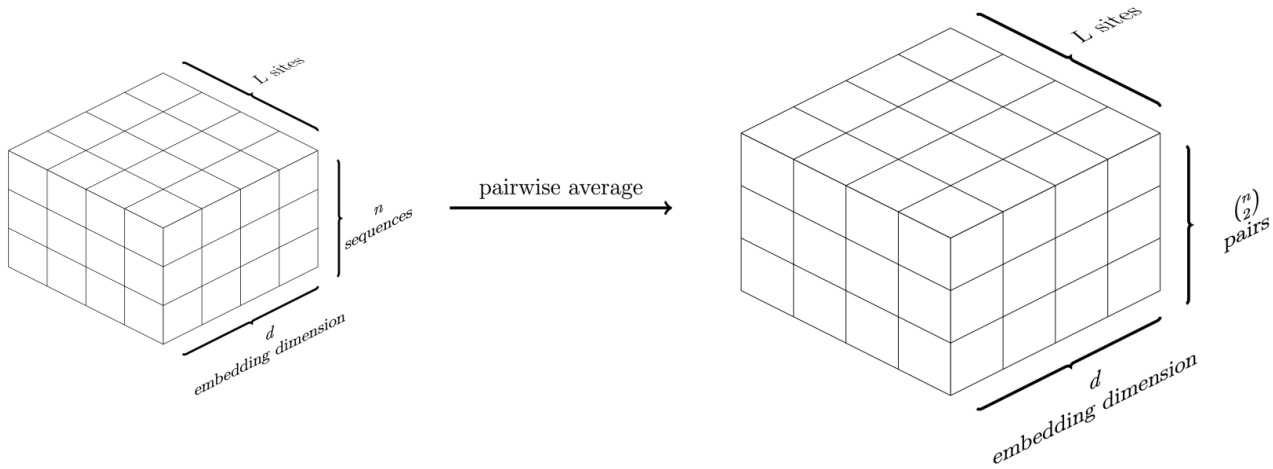
	A	A	C	G	T	$\ldots$
A	1	1	0	0	0	$\ldots$
C	0	0	1	0	0	$\ldots$
T	0	0	0	0	1	$\ldots$
G	0	0	0	1	0	$\ldots$

A set of aligned sequences:



Our alphabet is actually {A, R, N, D, ..., Y, V, X, -} so $d_0 = 22$.

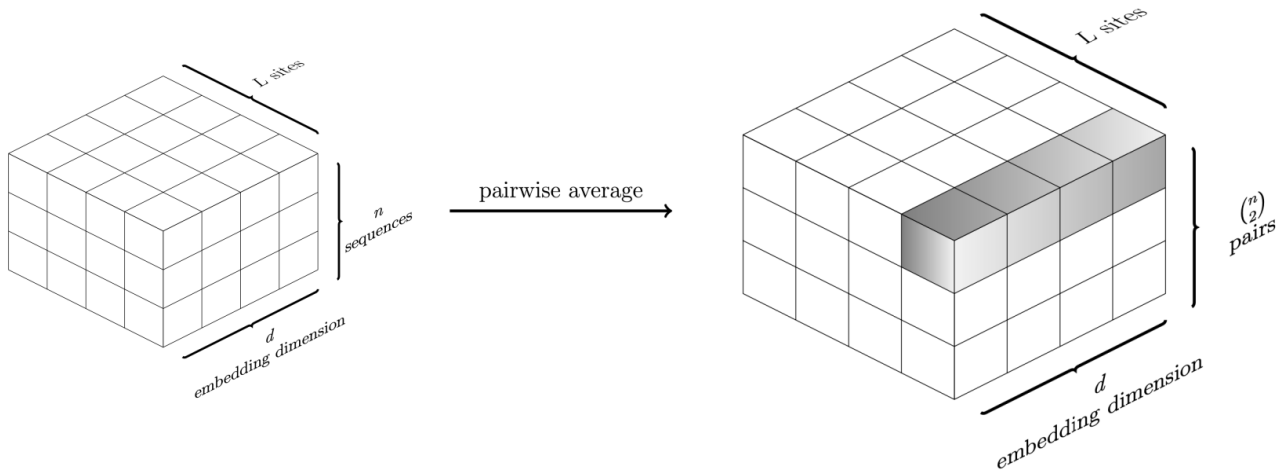
Encoding **pairs** of aligned sequences



- We choose to work on pairs of sequences (predict distance for each).
- We represent each pair by simply averaging over sequences.

	A	A	C	G	T	...
	A	T	C	C	T	...
A	1	0.5	0	0	0	...
C	0	0	1	0.5	0	...
T	0	0.5	0	0	1	...
G	0	0	0	0.5	0	...

Encoding **pairs** of aligned sequences

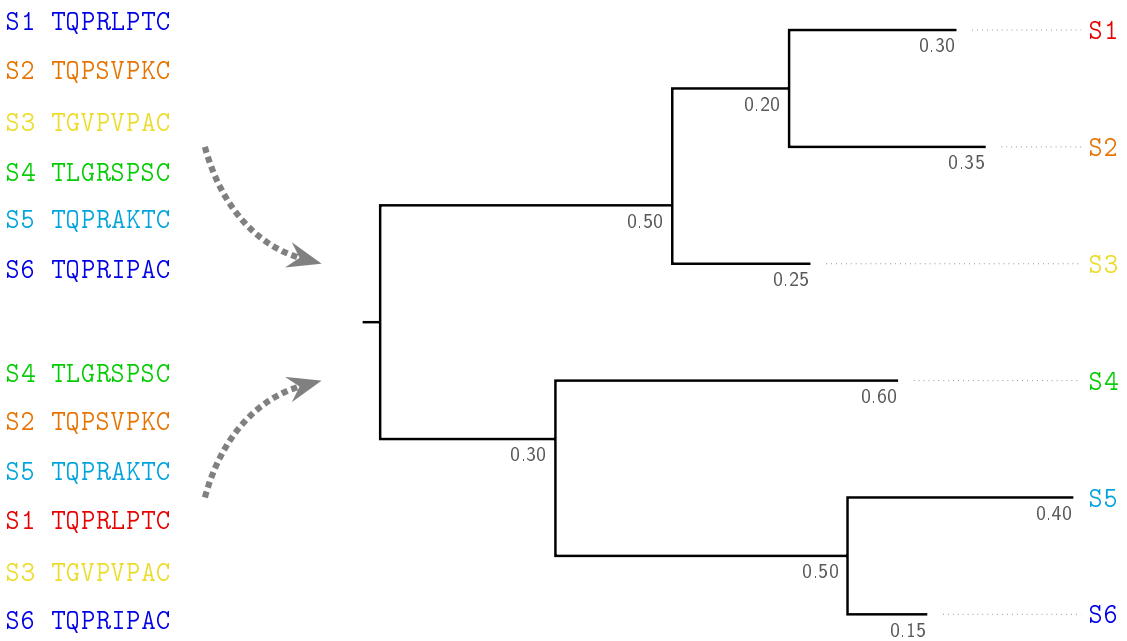


- We choose to work on pairs of sequences (predict distance for each).
- We represent each pair by simply averaging over sequences.

	A	A	C	G	T	...
	A	T	C	C	T	...
A	1	0.5	0	0	0	...
C	0	0	1	0.5	0	...
T	0	0.5	0	0	1	...
G	0	0	0	0.5	0	...

- We now have a set of $\binom{n}{2} \times L$ amino acids encoded as $\mathbb{R}^{d=22}$ vectors.

Accounting for permutation invariance with self-attention



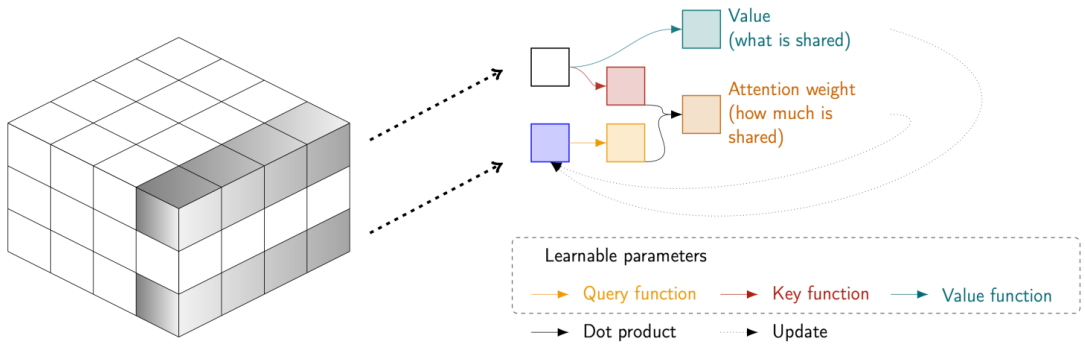
This has **no reason to be true in general** (*e.g.* linear function)!

Need to retain some expressivity.
E.g. average provides invariance but discards a lot of information.

Self-attention in a nutshell

Functions acting on unordered sets

- Updates each element as a linear combination of all of them.
- Output is a new representation of the same set. Iterate.

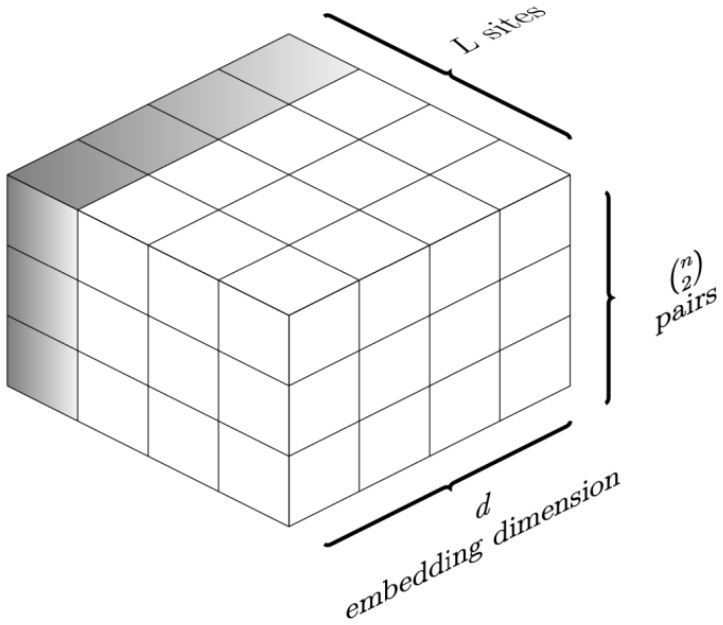


Updates

- Learnable part: function of two elements, giving weight of one in the update of the other.
- Provides equivariance, modularity to any cardinal.
- Iteratively builds a set-aware representation for each pair.

Axial attention

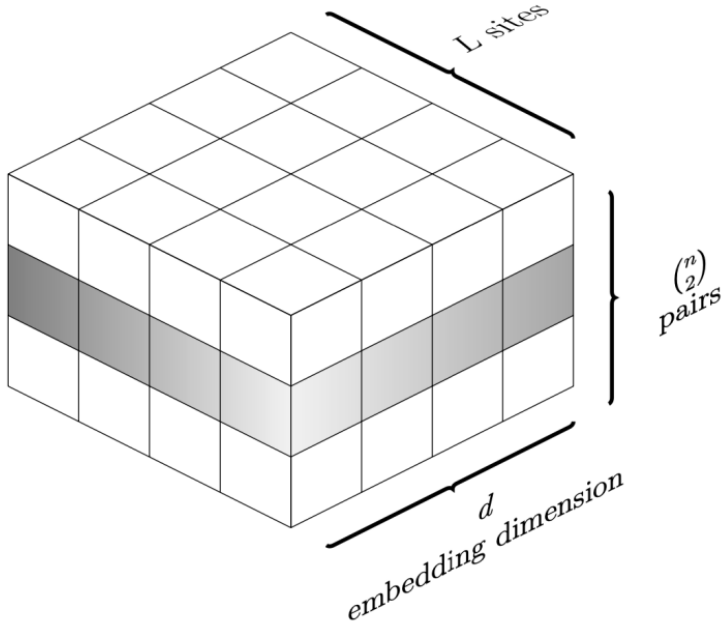
- We need equivariance both across pairs and sites.
- Alternate between column- and row-wise attention.



For each site, update each pair using all others.

Axial attention

- We need equivariance both across pairs and sites.
- Alternate between column- and row-wise attention.



For each pair, update each site using all others.

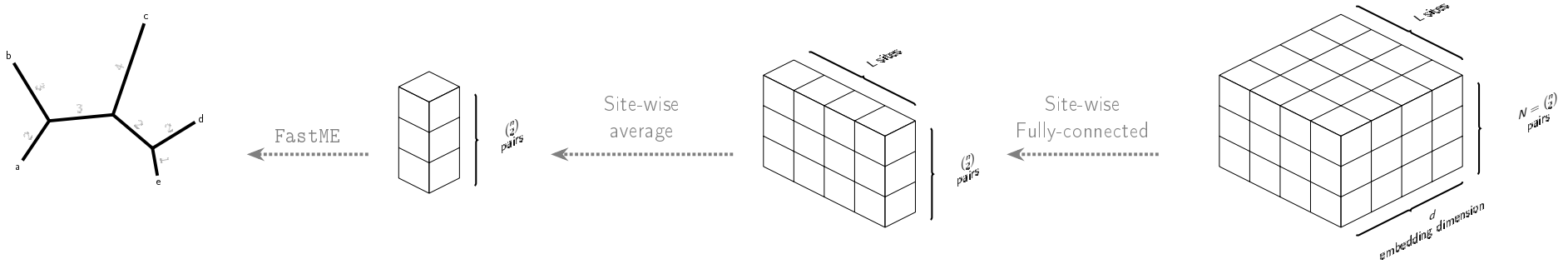
Builds a **set-aware representation** of residuals

"I'm an Alanine" $\rightarrow$

- "I'm an Alanine,
- some homologous sequences have Serines,
- many residues in the sequence are hydrophobic,
- this site is conserved,
- ..."

This representation is optimized with respect to the prediction objective.

Phyloformer overview

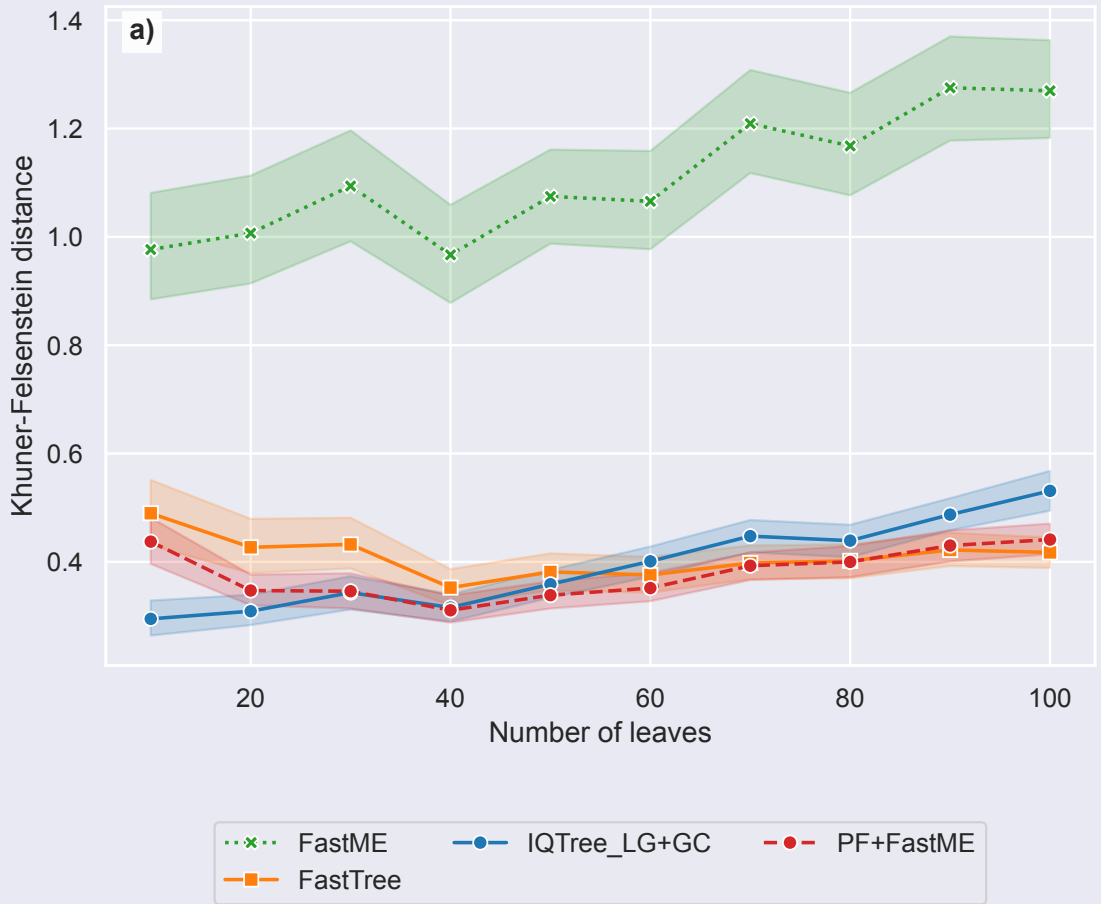


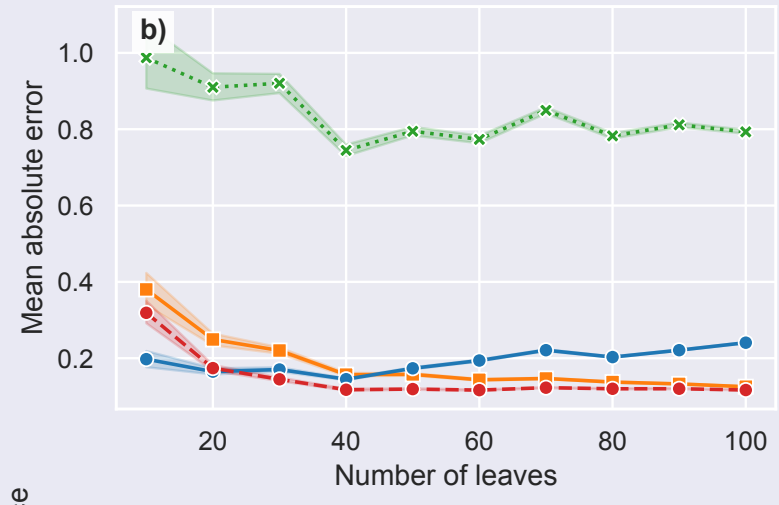
Final step: predict pairwise distances

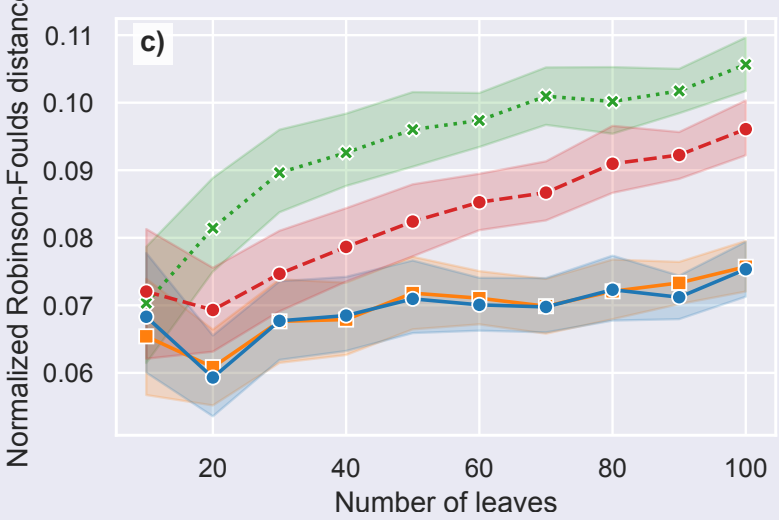
- Predict one number for each residual.
- Pool across sites to obtain a single value per pair.
- Loss function happens at this level:
  compare to true distance on simulated data.

We then use a distance method to build the tree (not end-to-end).

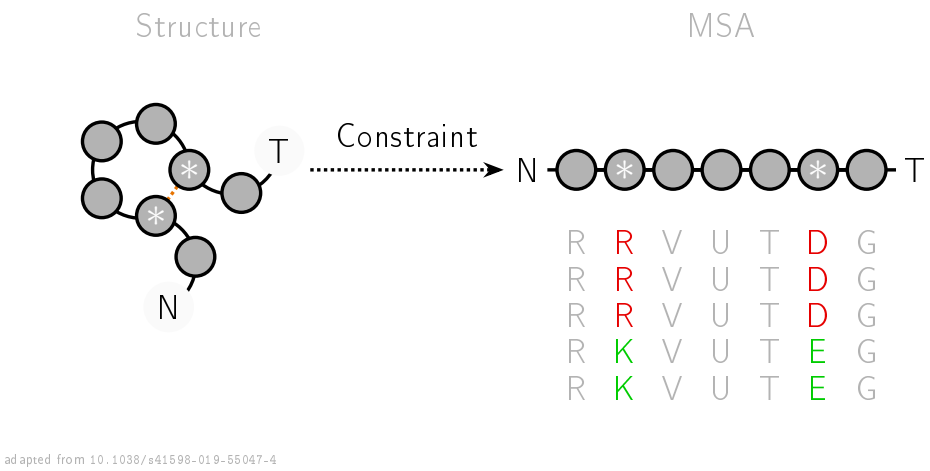
Results - Under LG+GC model, PF performs on par with ML







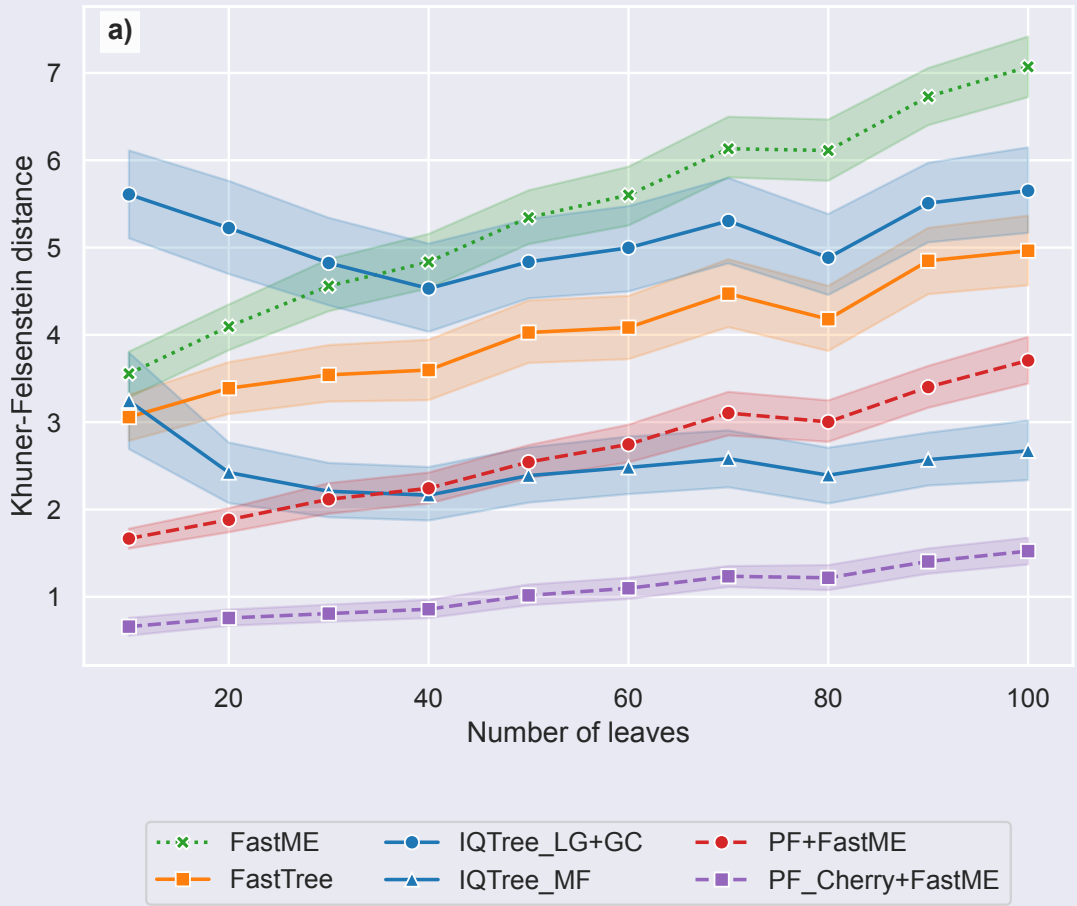
Results - What about a more complex model ?

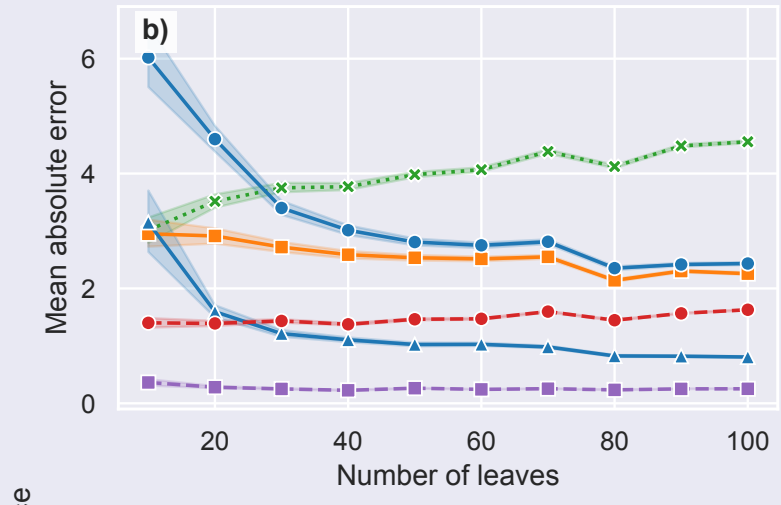


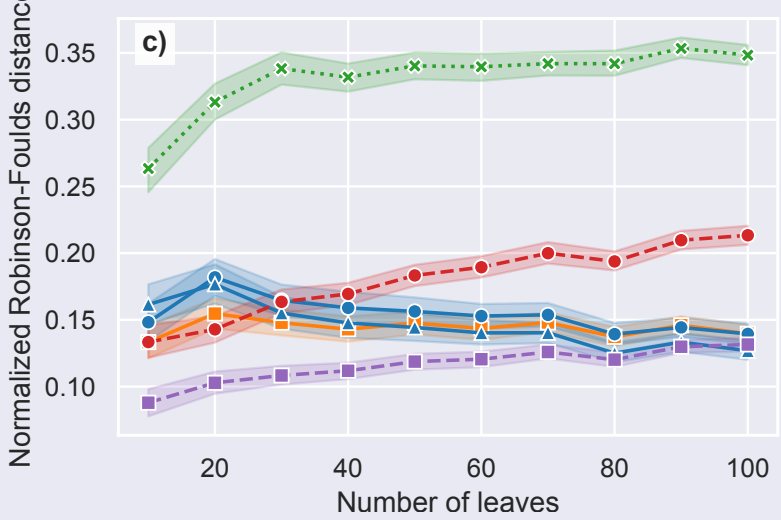
adapted from 10.1038/s41598-019-55047-4

- We **simulate** 250 pairs of **adjacent co-evolving sites**
- We use a 400×400 substitution **matrix** to describe residue **co-evolution**, from **CherryML**
- Most **ML** methods would consider **sites independent**

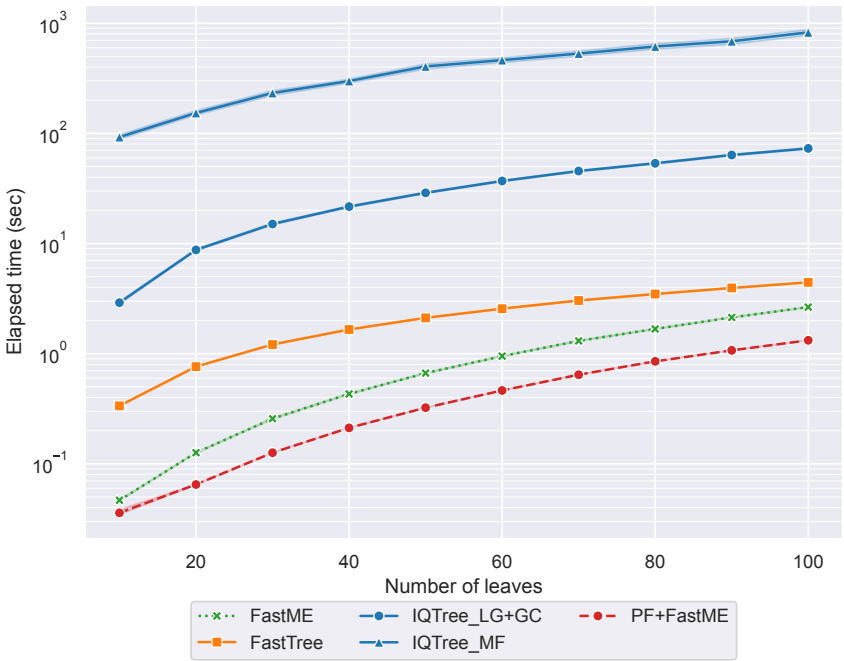
Results - Under a co-evolution model, PF performs the best







Results - Inference speed

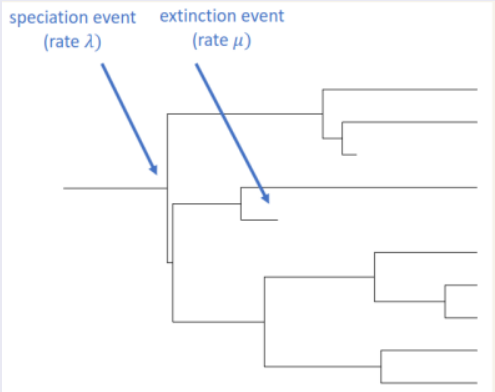


- **Phyloformer** is the **fastest** method
- Phyloformer is even **faster than FastME** on its own
- Inference **speed** is **independent** from model **complexity**

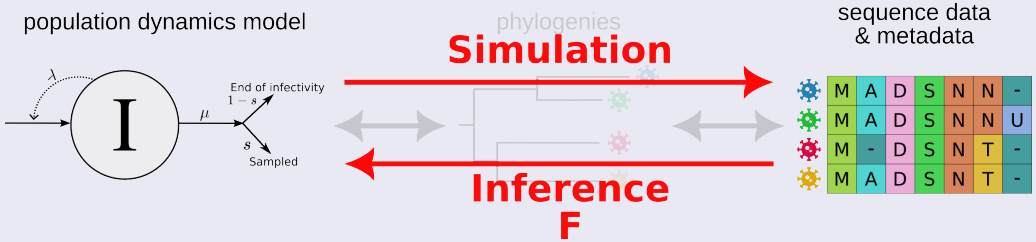
Phylodynamics: evolutionary parameter inference

Phylodynamics vs Phylogenetics

- So far we have sampled trees from a **parameterized distribution**.
- These parameters themselves have a meaning in
  - epidemiology (R_0, duration),
  - ecology (biodiversification).



Phylodynamics from sequences (skip the tree)



- Existing likelihood-free phylodynamics methods start from phylogenies.
- Skipping the tree: faster, handles phylogenetic uncertainty and cases where there is no tree (e.g. recombination).

Differences with Phyloformer

Posterior inference on $(R_0, \text{duration})$ with quantile regression

- Reminder: $\arg\min_m \sum_i |m - R_0^i|$ estimates the median of $p(R_0)$.
- We are interested in the *conditional* median of $p(R_0|\text{sequence})$.
- Our network m_θ minimizes $\arg\min_\theta \sum_i |m_\theta(\text{sequence}_i) - R_0^i|$.
- Generalizes to other quantiles with the pinball loss (asymetric).

Accounting for dates

- In epidemiology, we have (and need) dated sequences.
- We incorporate this information through positional encodings.

Permutation invariance vs equivariance

- We want a single prediction per MSA, not per pair.
- We don't form pairs (better scaling).
- We use special CLS tokens for global pooling.

Transformers for EpiDemiological DYnamics (TEDDY)

Setting

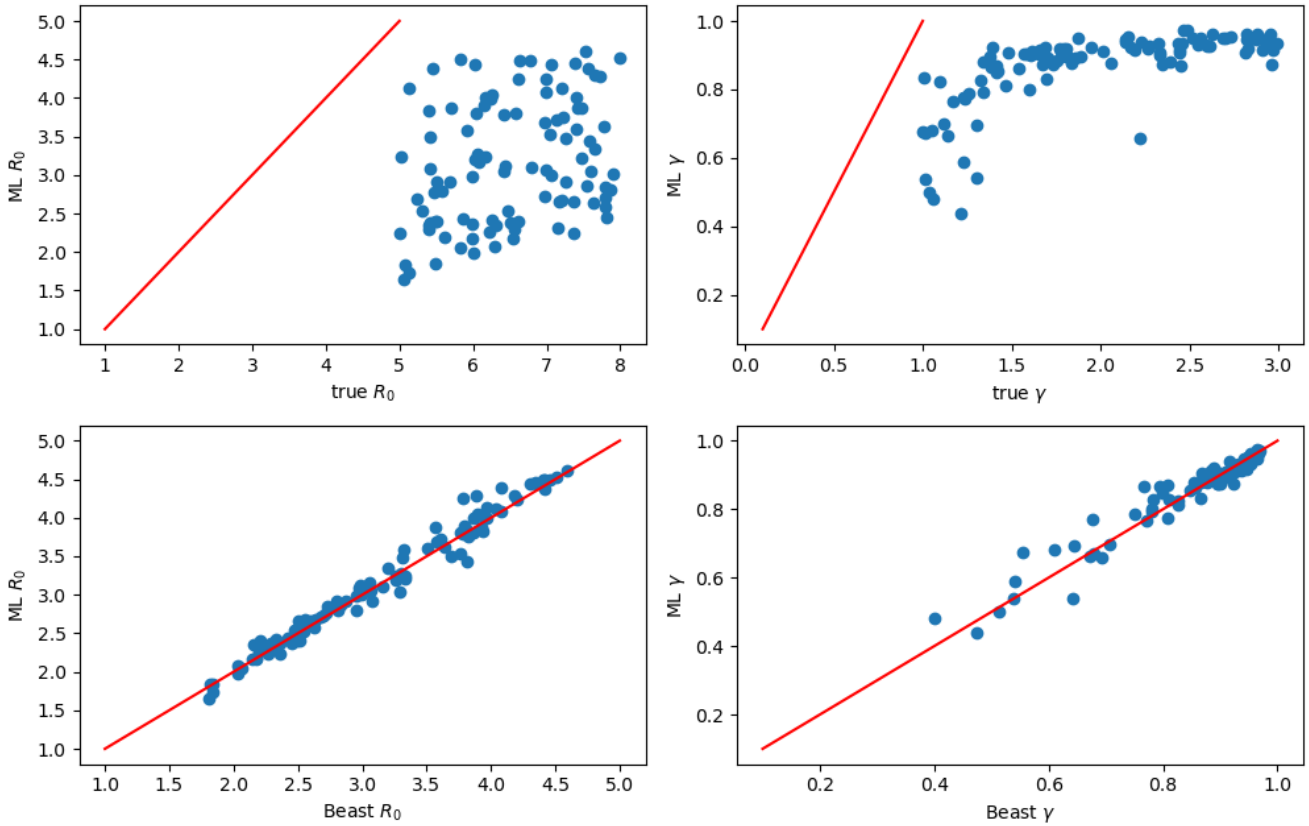
- Sample $R_0 \sim \mathcal{U}(1,5)$ and duration $\sim \mathcal{U}(0.1,1)$.
- Then 50-leave trees from birth-death(R_0, duration)
- Then 1000-long sequences from these trees.

Parameter	BEAST2	Teddy (ours)
R_0	0.18	0.18
duration	0.25	0.26
Time for 1000 runs	17 days	50s

- Same relative errors as BEAST2 (SOTA), 1e5 x faster.
- 95% credible intervals correctly estimated in both cases.

(Non-)robustness to strong prior misspecification

- Network trained on $R_0 \in [1, 5] \times \gamma \in [0.1, 1]$.



- Performs poorly on data where $R_0 \in [5, 8] \times \gamma \in [1, 3]$.
- But behaves exactly like BEAST2.

Wrapping up

Summary

- Neural inference of evolutionary parameters.
- Sequences to tree (Phyloformer), or to upstream parameters (Teddy).
- Much faster than likelihood-based alternatives under simple models.
- Additionally, more accurate under complex models.

Perspectives

- Calibration assessment, full posteriors.
- Train and assess networks under more complex models.
- End-to-end from sequences to the tree.

Thank you.