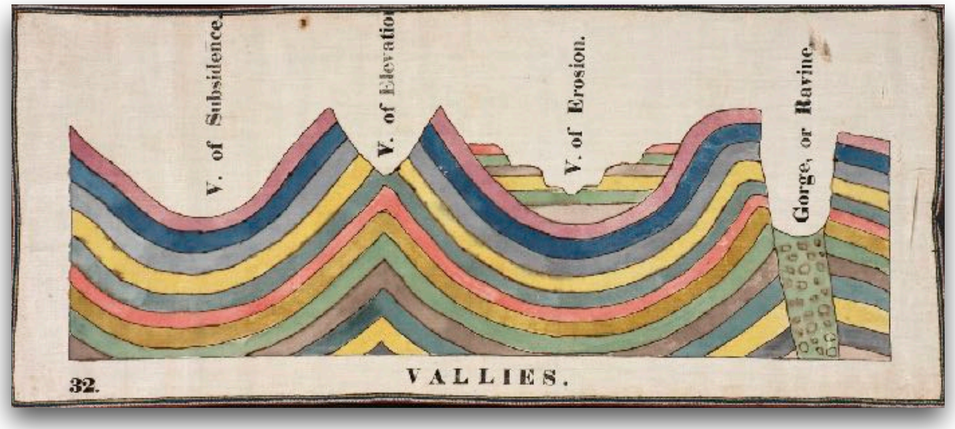
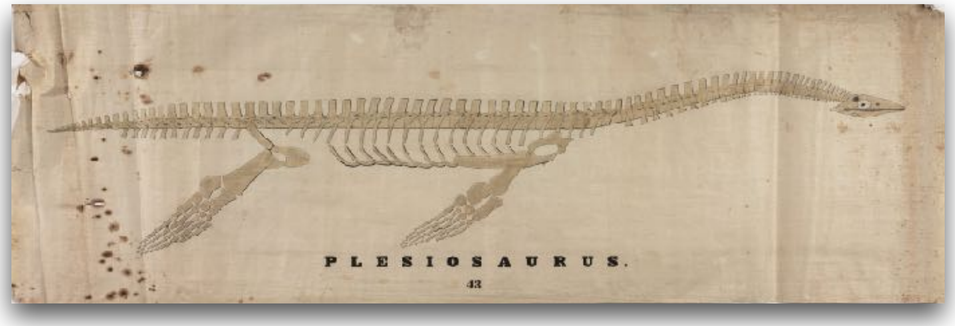
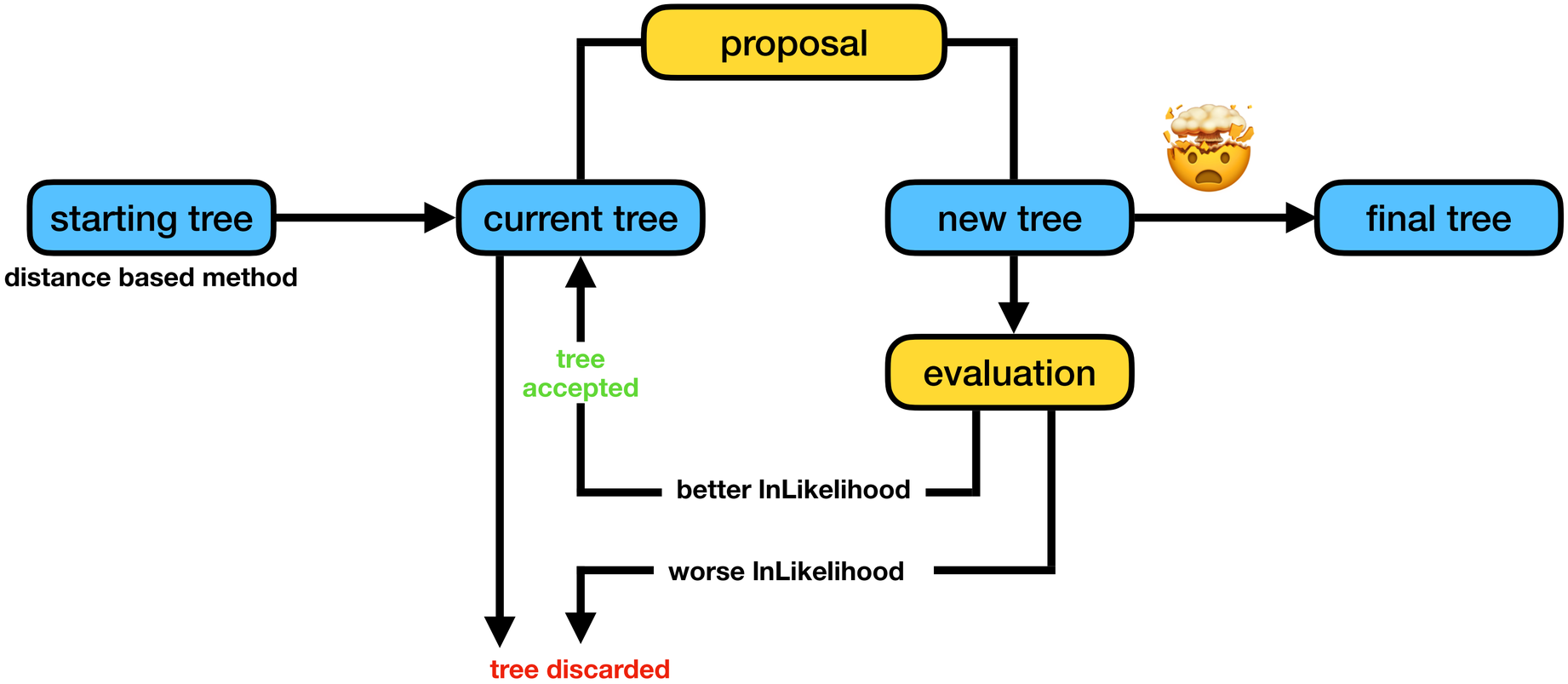




Bayesian Inference (BI)


proposal
starting tree
distance based method
current tree
new tree
final tree
tree
accepted
evaluation
better lnLikelihood
worse lnLikelihood
tree discarded

proposal

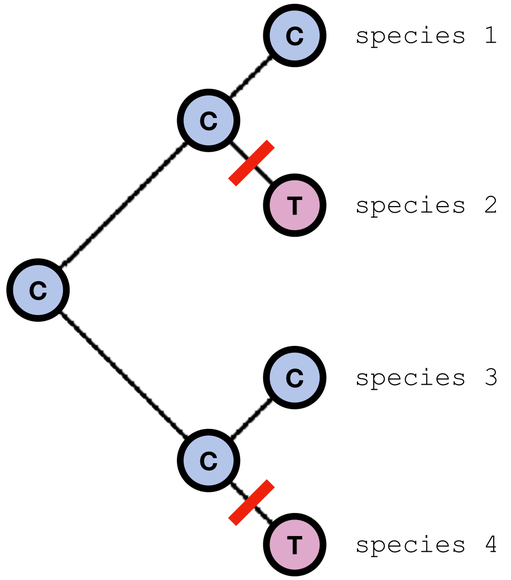
Nearest Neighbour Interchange (NNI) is a local search method that makes small, localized changes to the tree. It is the simplest and fastest tree rearrangement algorithm!

Subtree Pruning and Regrafting (SPR) is a more extensive tree rearrangement method than NNI. It moves entire subtrees rather than just swapping immediate neighbors.

Tree Bisection and Reconnection (TBR) is the most aggressive of these tree rearrangement methods and explores a much larger space of possible tree topologies. It has the lesser chance of local optima!

evaluation

```
          1 2 3 4 5 6 7 8
species 1 A G G A C C G A
species 2 A C T T T C G G
species 3 A G G A C C T T
species 4 A C G T T C T T
              *
```



$$Q = \begin{array}{c} \begin{array}{cccc} \mathbf{A} & \mathbf{G} & \mathbf{C} & \mathbf{T} \end{array} \\ \begin{pmatrix} -(\alpha\pi_G + \beta\pi_C + \gamma\pi_T) & \alpha\pi_G & \beta\pi_C & \gamma\pi_T \\ \alpha\pi_A & -(\alpha\pi_A + \delta\pi_C + \epsilon\pi_T) & \delta\pi_C & \epsilon\pi_T \\ \beta\pi_A & \delta\pi_G & -(\beta\pi_A + \delta\pi_G + \eta\pi_T) & \eta\pi_T \\ \gamma\pi_A & \epsilon\pi_G & \eta\pi_C & -(\gamma\pi_A + \epsilon\pi_G + \eta\pi_C) \end{pmatrix} \begin{array}{c} \mathbf{A} \\ \mathbf{G} \\ \mathbf{C} \\ \mathbf{T} \end{array} \end{array}$$

$P_{CT} = r_{CT}\pi_C$

$P_{site4} = P_{CT} \times P_{CT} \times P_C \times P_C \times P_C \times P_C$

Maximum Likelihood P(DIM)
probability of observed data given a model

Bayesian Inference P(MID)
probability the model is correct given the data

… but why Bayesian Inference ?!

We refer to Bayesian Inference due to the **Bayes Theorem**.

It describes how to **update our beliefs based on new evidence** …

… and it provides a mathematical framework for conditional probability!

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where:

- $P(A|B)$ is how often A happens *given that B happens*
- $P(B|A)$ is how often B happens *given that A happens*
- $P(A)$ is how likely is A to happen on its own
- $P(B)$ is how likely is B to happen on its own

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where:

- $P(A|B)$ ***- the posterior probability***
- $P(B|A)$ ***- the likelihood***
- $P(A)$ ***- the prior probability***
- $P(B)$ ***- the marginal likelihood***

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Example:

- **fires are rare: P (Fire) = P(A) = 0.01**
- **smoke is fairly common: P (Smoke) = P (B) = 0.1**
- **dangerous fires make smoke: P (Smoke | Fire) = P (B | A) = 0.9**

Probability of fire when I see smoke?! **P (Fire | Smoke) = P (A | B) = ?!**

… how are these concepts related to phylogeny?!

$$P(Tree|Data) = \frac{P(Data|Tree)P(Tree)}{P(Data)}$$

Where:

- $P(Tree|Data)$ probability of a given tree given the data **- *posterior probability***
- $P(Data|Tree)$ probability of the data given that the tree is correct **- *likelihood***
- $P(Tree)$ the prior belief about which trees are more likely **- *prior probability***
- $P(Data)$ ***the marginal likelihood …*** we are not getting deep into that 🥲

… but there is a statistical problem …

…. a direct computation of the marginal likelihood would involve integrating over all tree topologies and model of sequence evolution parameters, which is practically impossible due to the mindbending number of possibilities …

… and also a solution!

“From the earliest days of Bayesian phylogenetics, the numerical tool of choice for approximating the posterior distribution was Markov chain Monte Carlo (MCMC).

The popularity of MCMC was due, in no small part, to avoiding the calculation of the marginal likelihood of the model - the probability of the data under the model, averaged, with respect to the prior, over the whole parameter space.”

Oaks et al. 2019

MCMC is a class of algorithms used to approximate complex posterior distributions where direct sampling is challenging by generating a sequence of samples. Composed by:

- **Markov Chain:** a sequence of possible events where the probability of each event depends only on the state attained in the previous event.

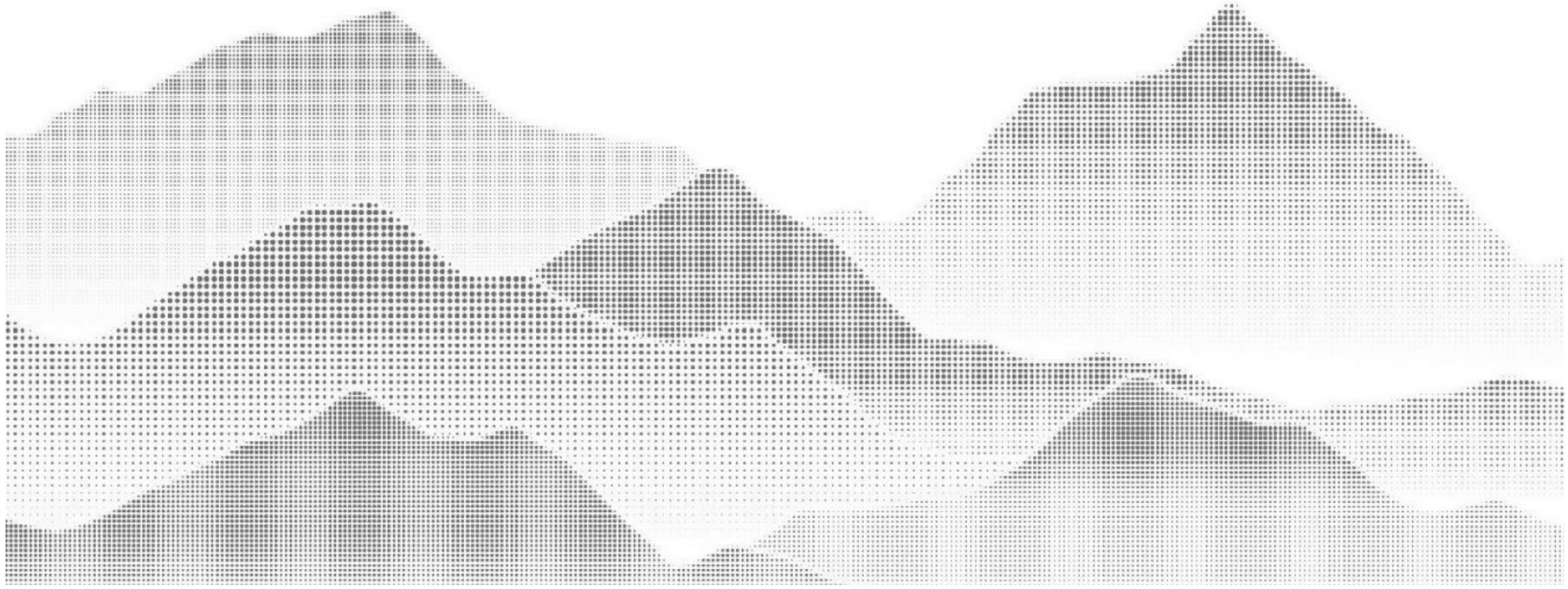
- **Monte Carlo:** random sampling to obtain numerical results, used to estimate functions and mimic the operations of complex systems.

How MCMC works:

- **Initialize:** start with an arbitrary point in the parameter space.
- **Propose:** generate a new point based on a proposal distribution.
- **Evaluate:** decide whether to accept or reject the new point
- **Iterate:** repeat proposal and acceptance to build a chain of samples

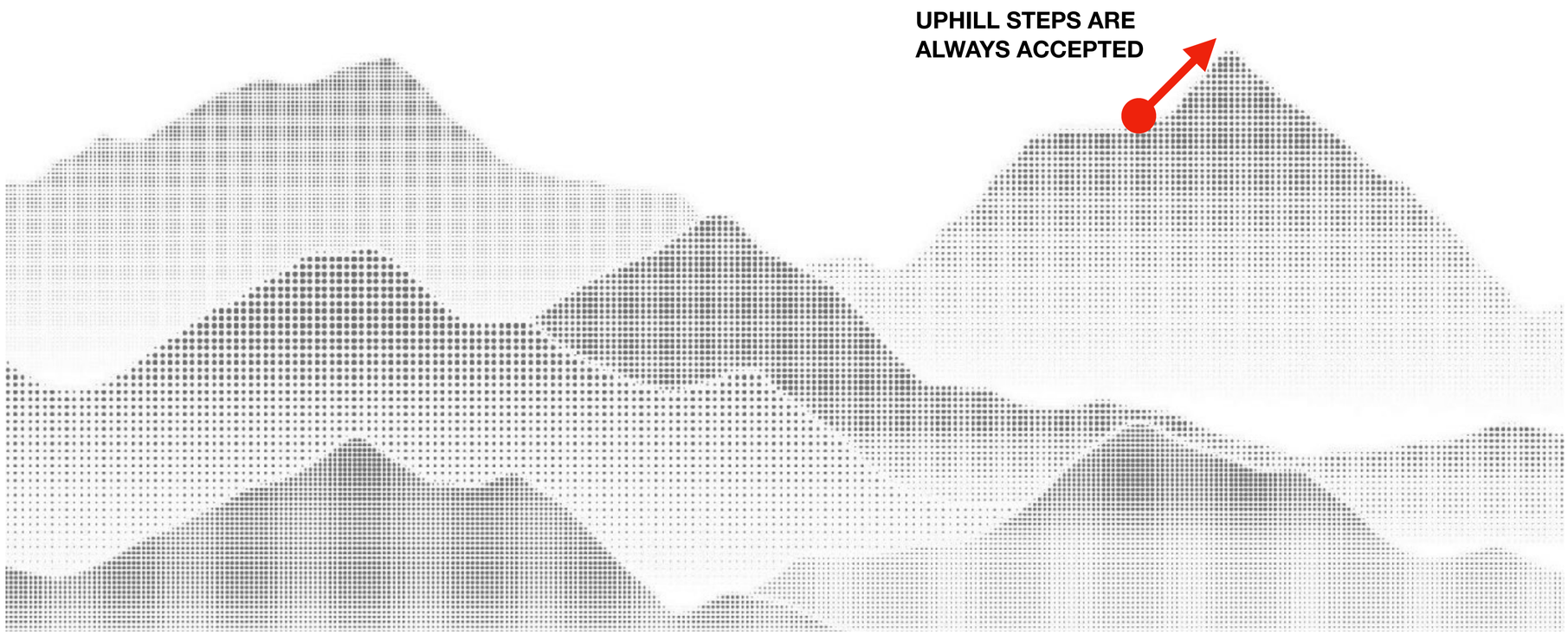
Metropolis-Hastings *criterion* takes a step if a random num. between 0 and 1 is less or equal to R.

R = proposed state / current state



Metropolis-Hastings *criterion* takes a step if a random num. between 0 and 1 is less or equal to R.

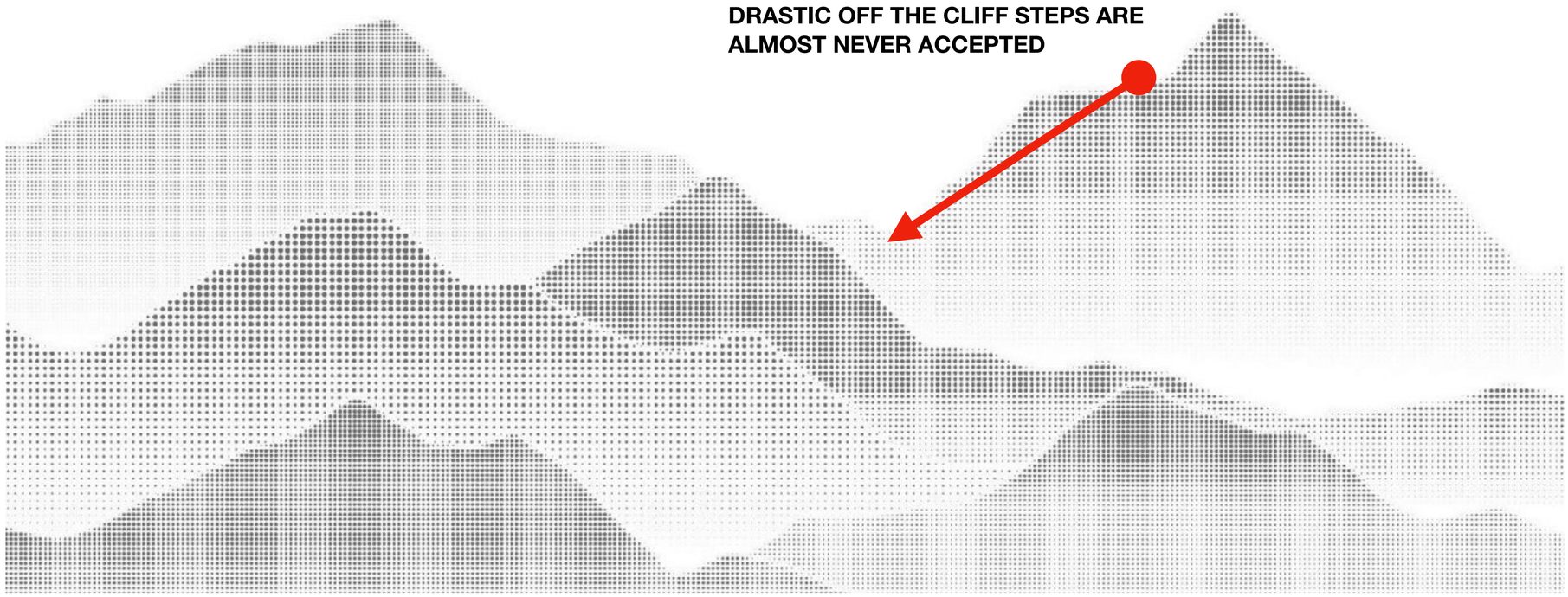
- **Moves to higher-probability states always accepted!**
  current state = 100; proposed state = 120; R = 120 / 100 = 1.20



Metropolis-Hastings *criterion* takes a step if a random num. between 0 and 1 is less or equal to R.

- **Moves to higher-probability states always accepted!**
  current state = 100; proposed state = 120; R = 120 / 100 = 1.20

- **Moves to very low probability states nearly always rejected!**
  current state = 115; proposed state = 20; R = 20 / 115 = 0.17

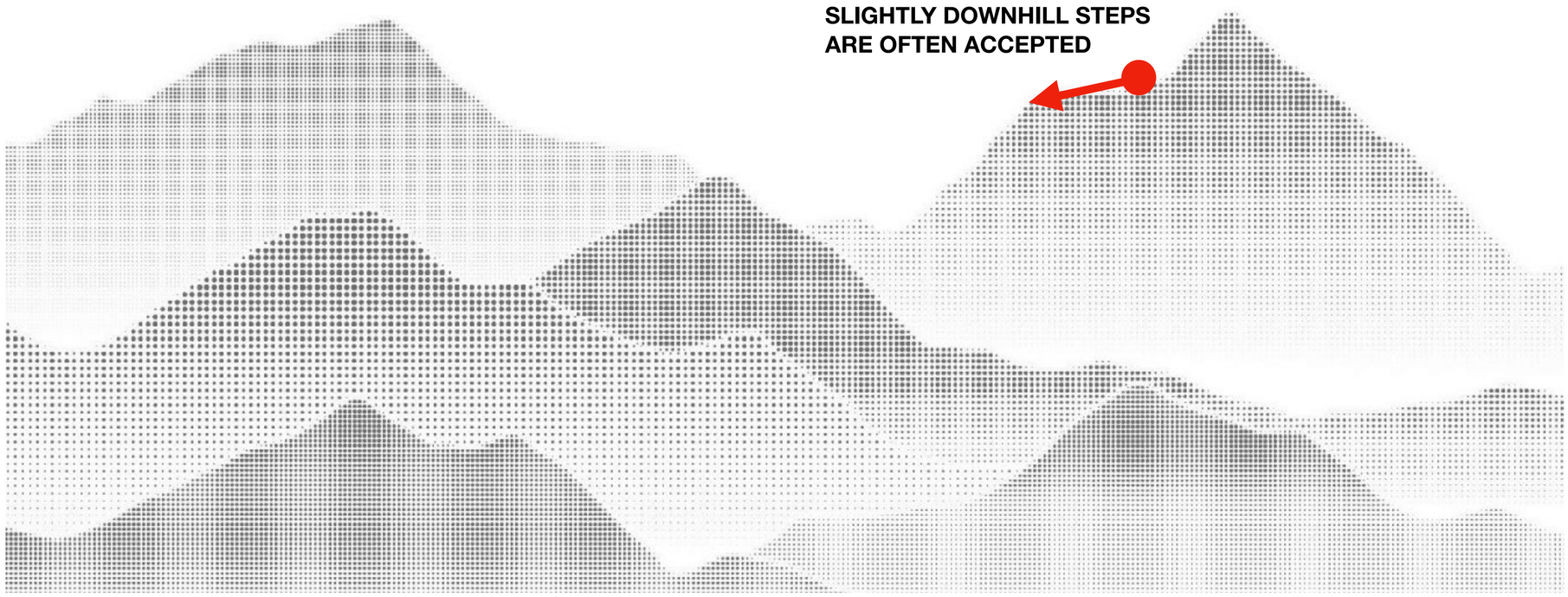


Metropolis-Hastings *criterion* takes a step if a random num. between 0 and 1 is less or equal to R.

- **Moves to higher-probability states always accepted!**
  current state = 100; proposed state = 120; R = 120 / 100 = 1.20

- **Moves to very low probability states nearly always rejected!**
  current state = 115; proposed state = 20; R = 20 / 115 = 0.17

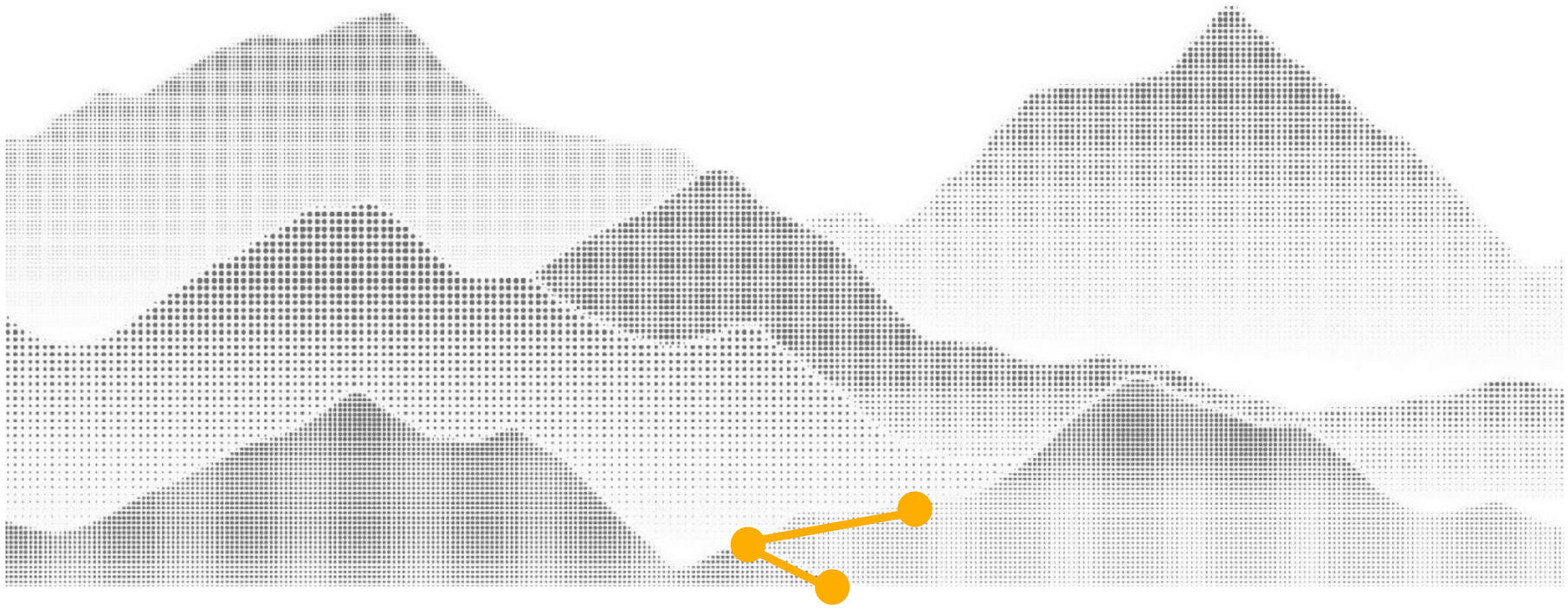
- **Moves to slightly low probability states nearly always accepted!**
  current state = 102; proposed state = 93; R = 93 / 102 = 0.92



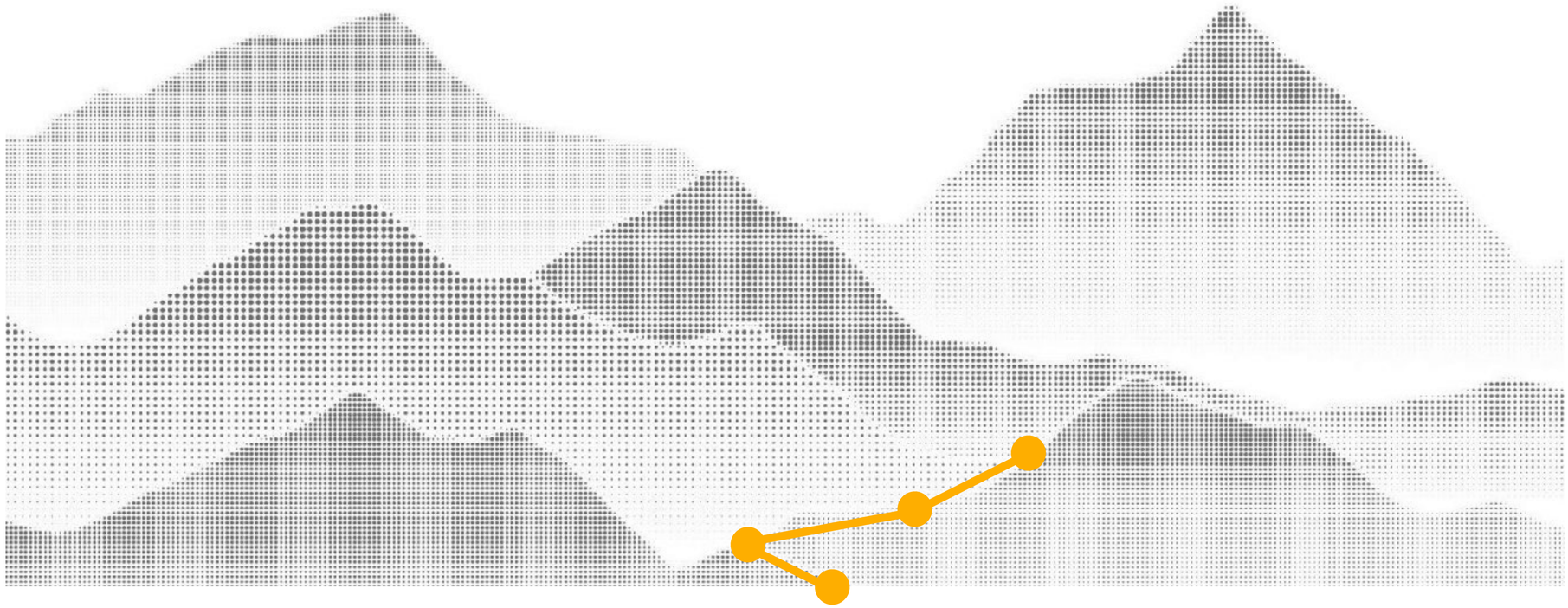
convergence and mixing

convergence and mixing

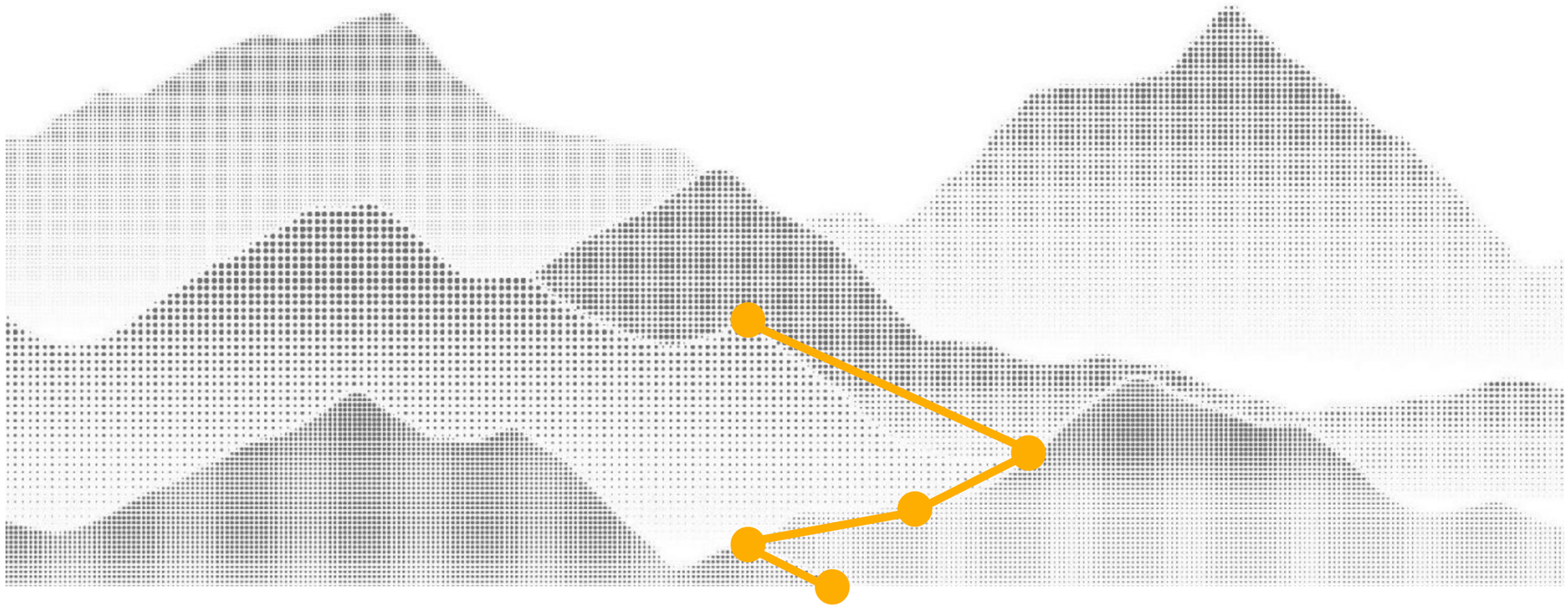
convergence and mixing



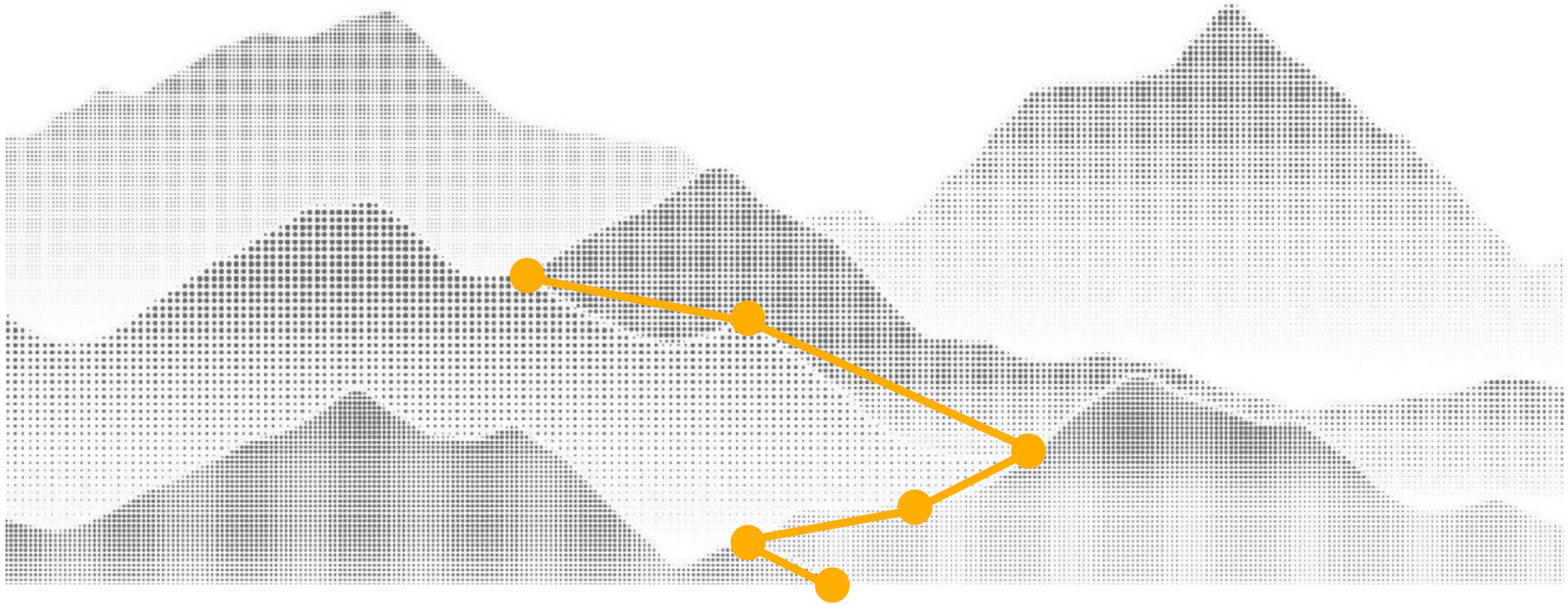
convergence and mixing



convergence and mixing



convergence and mixing



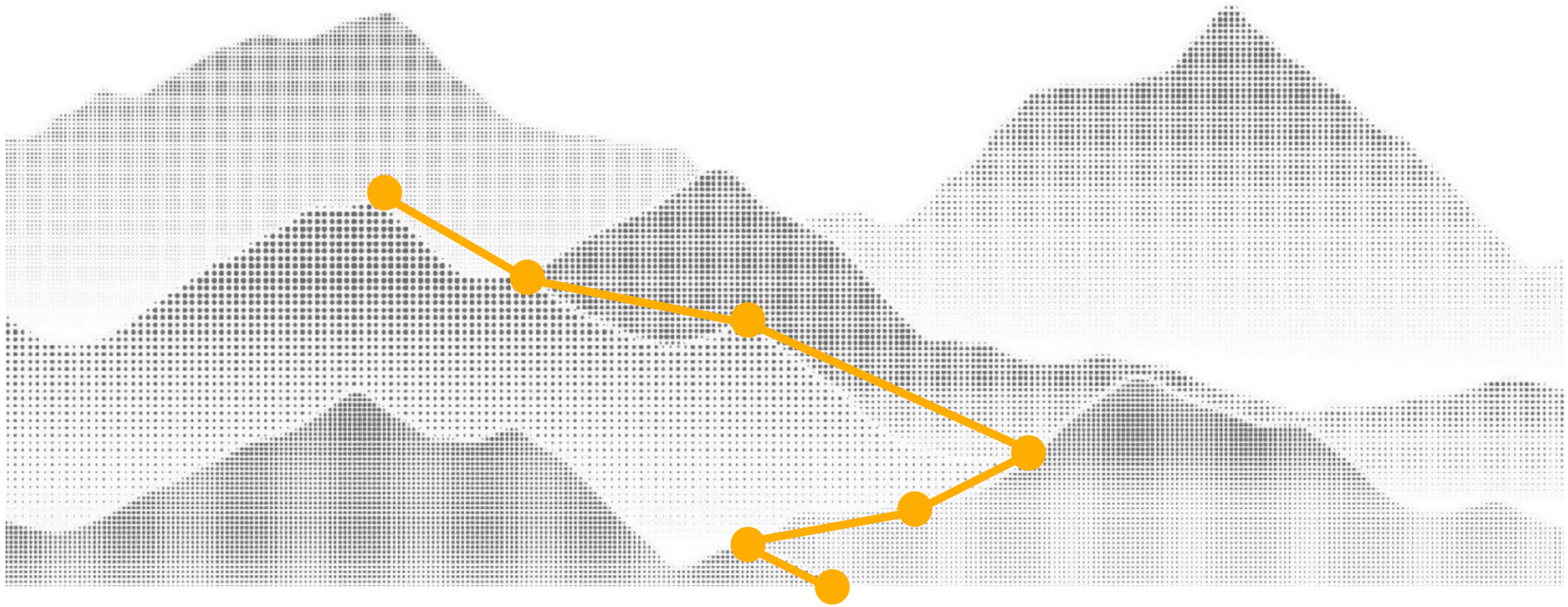
convergence and mixing

...

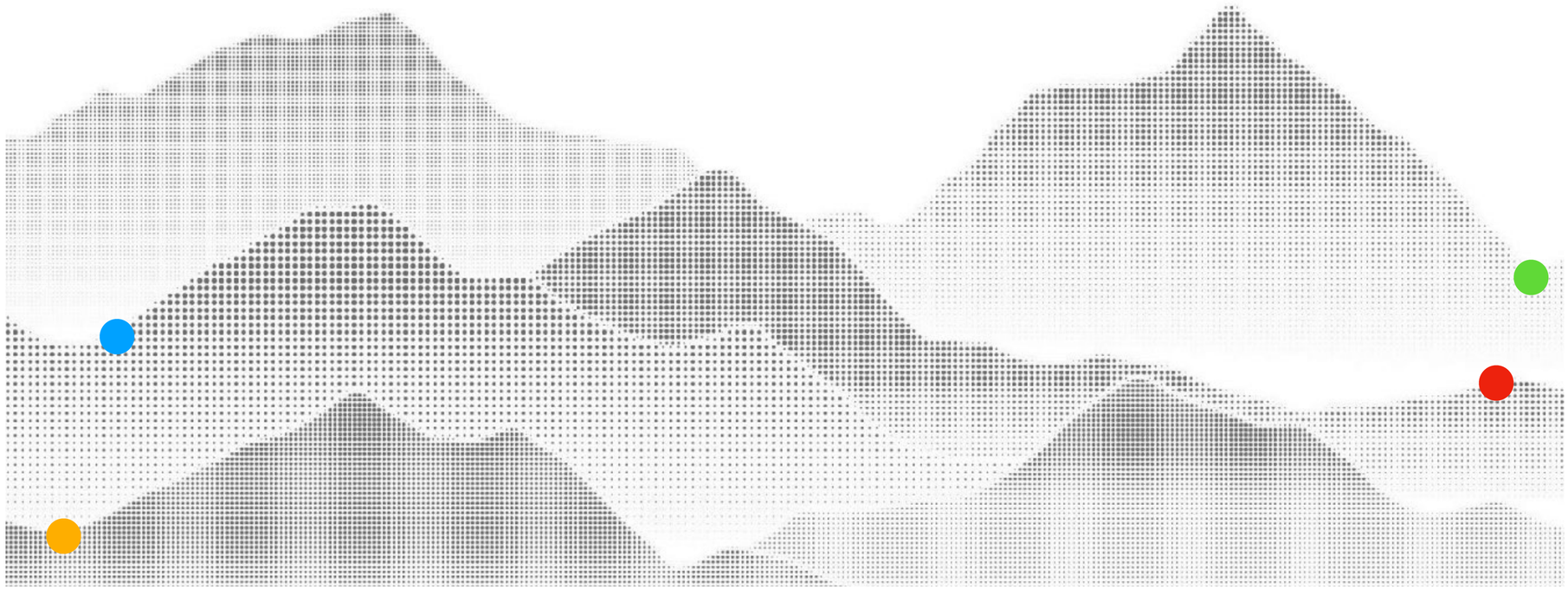
The problem is:

Are we on the highest peak?

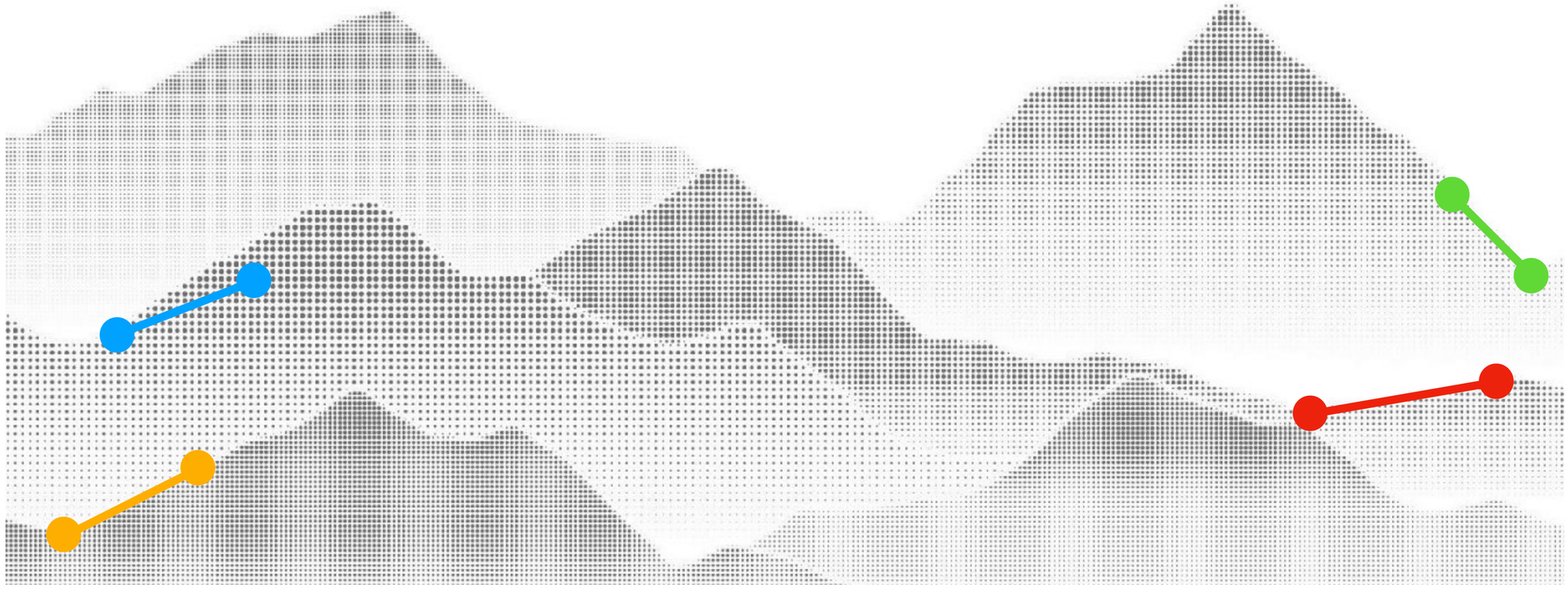
Are we still ascending?



convergence and mixing

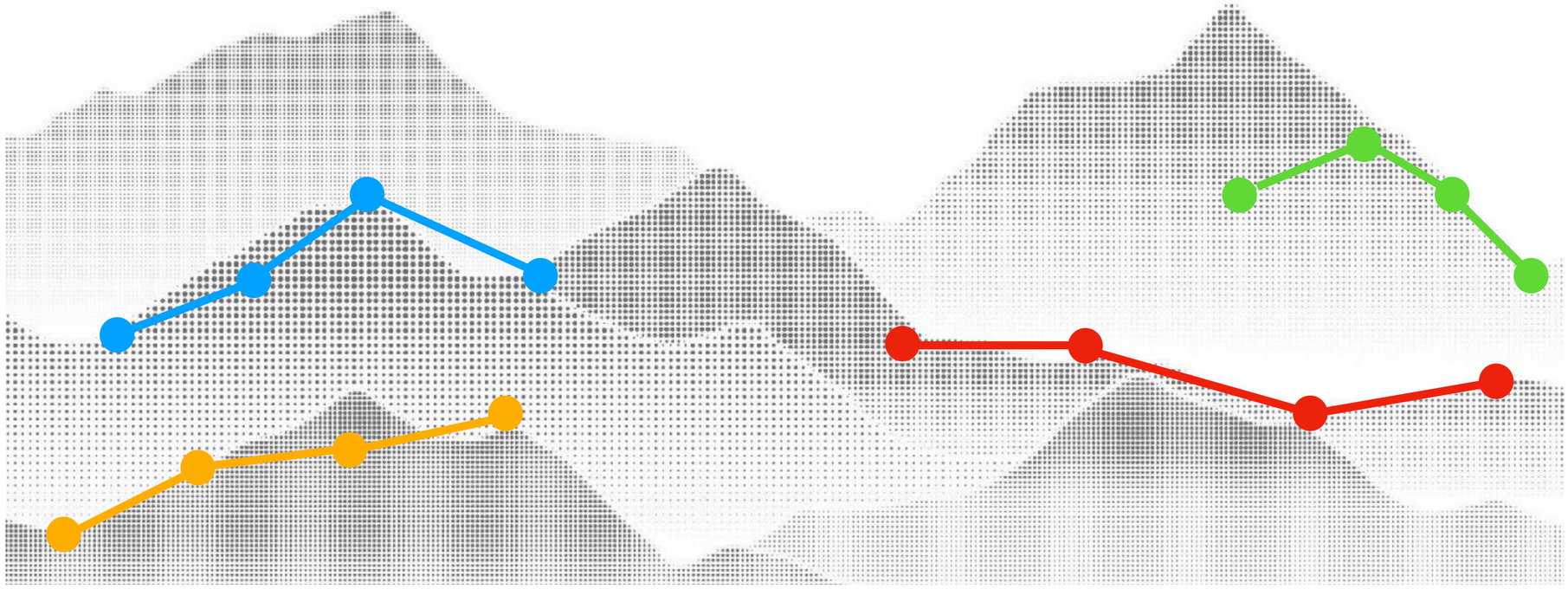


convergence and mixing

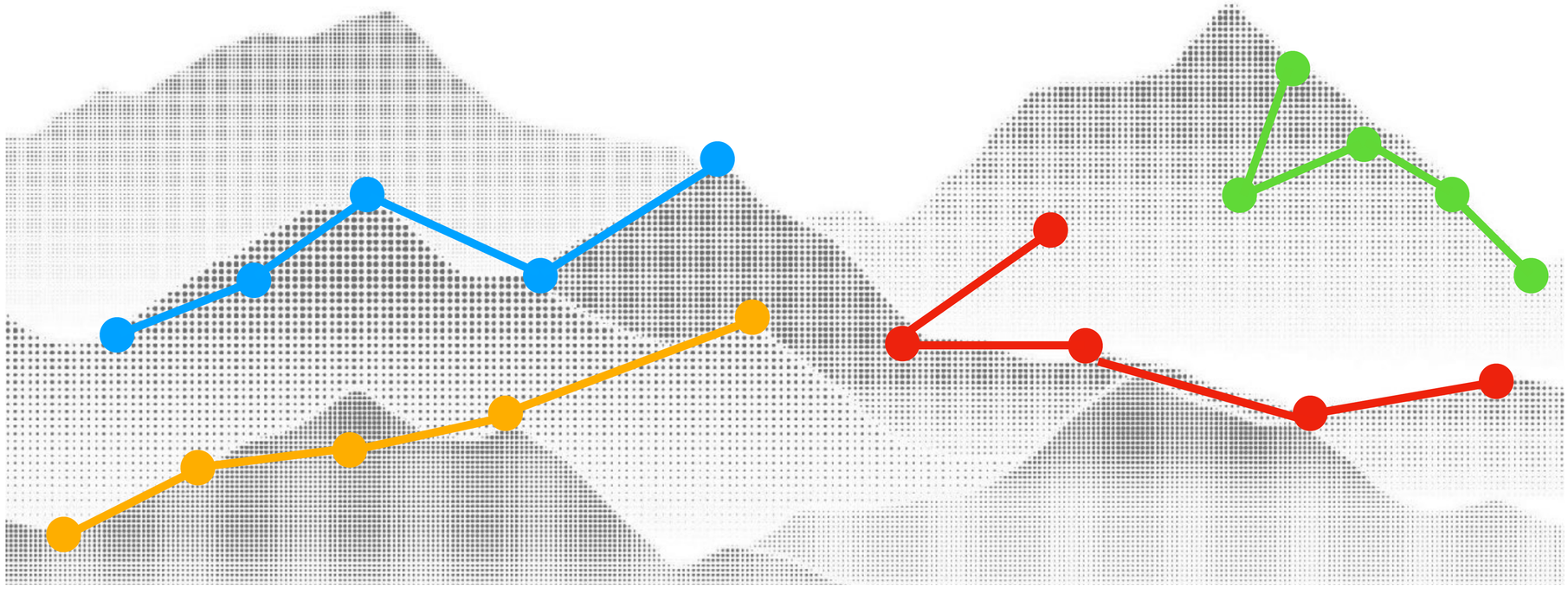


convergence and mixing

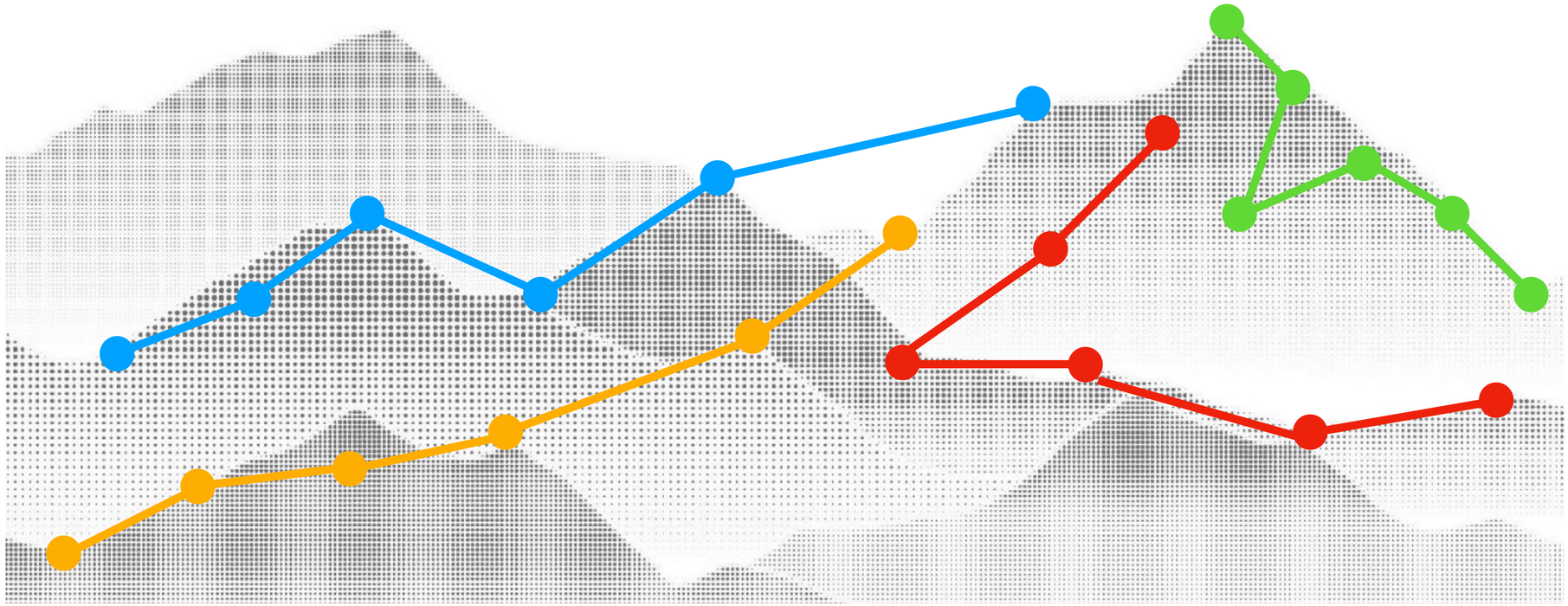
convergence and mixing



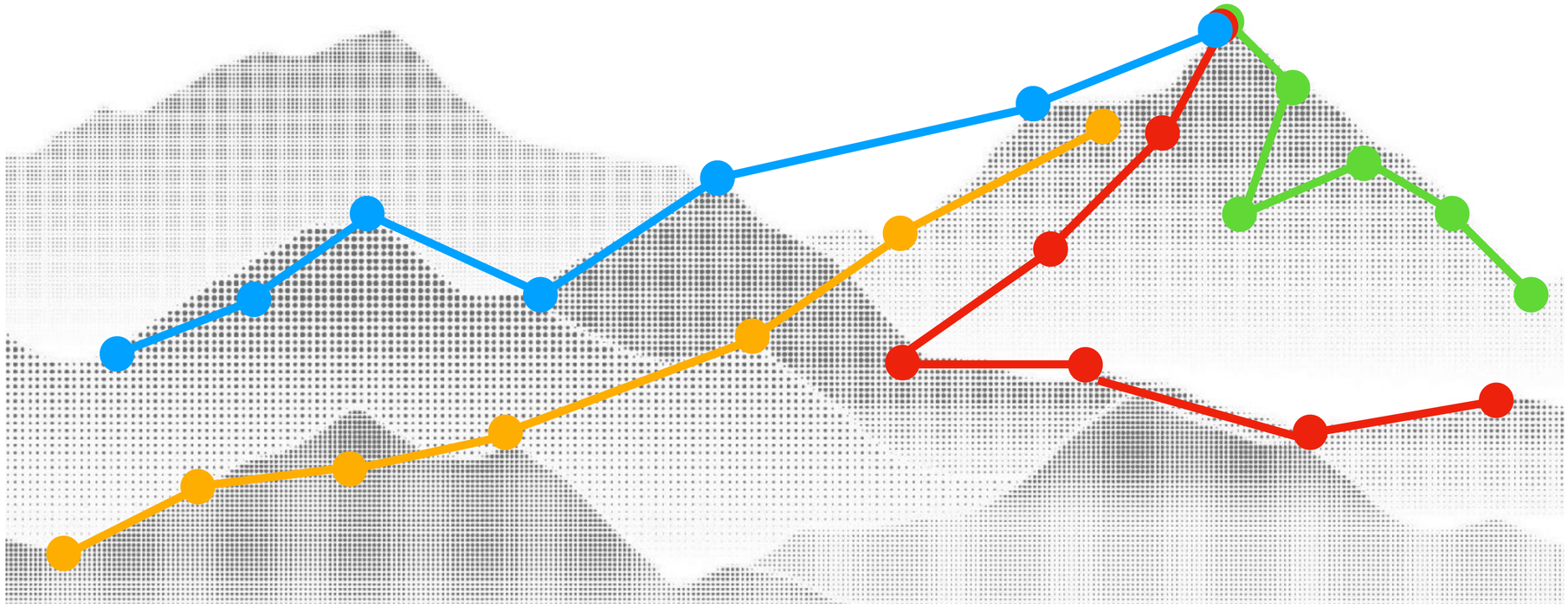
convergence and mixing



convergence and mixing



convergence and mixing

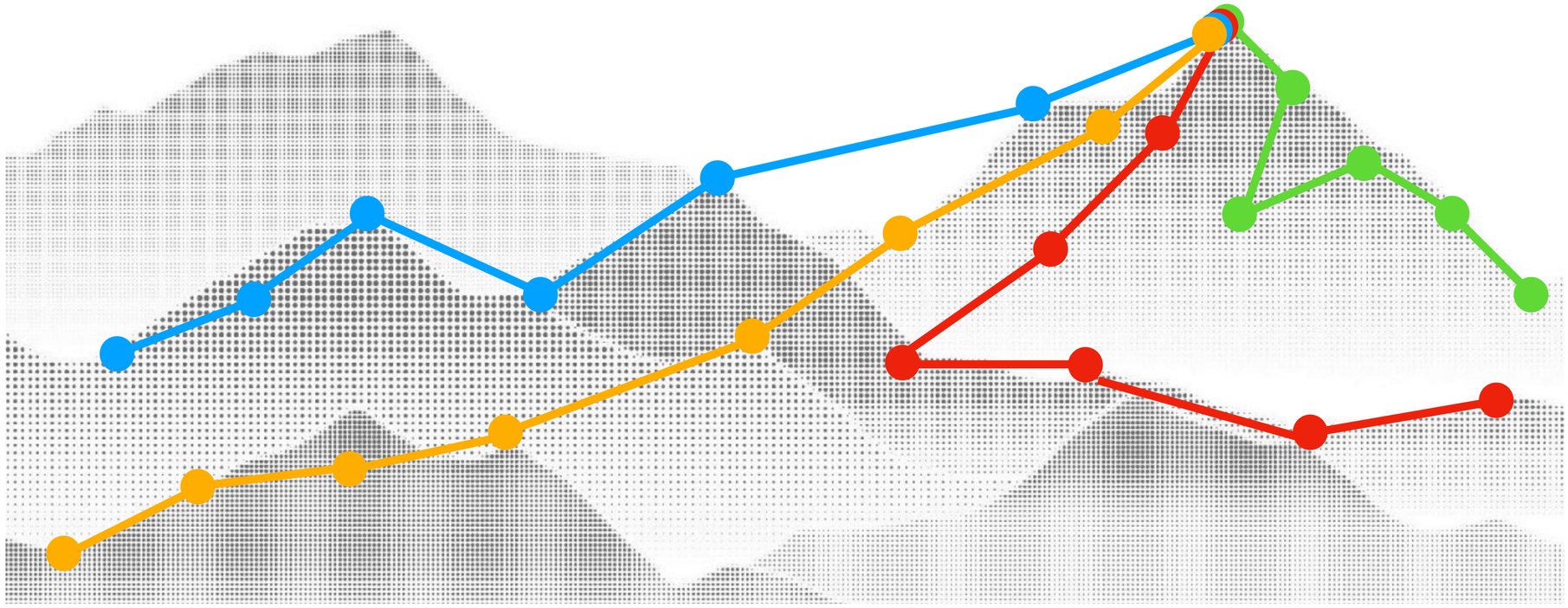


convergence and mixing

...

The solution is:

reproducibility of the results across independent runs
started from different initial conditions



What to expect during MCMC?

Log-Likelihood:

- Starts increasing rapidly as the chain explores better-fitting trees.
- Eventually stabilizes, indicating a *plateau*.

Tree Topology:

- Initially fluctuates as the chain explores different possibilities.
- Over time, a consistent topology emerges.

Model Parameters:

- Parameters fluctuate widely at first.
- As sampling continues, they settle within a narrower range.

What is at the peak?!

stationary / target distribution

The goal of MCMC is to construct a chain …
whose **stationary distribution** matches the desired **target distribution**!

By running the chain for a sufficient number of iterations,
the samples generated will approximate this target distribution,
allowing for **accurate estimation of statistical properties**.

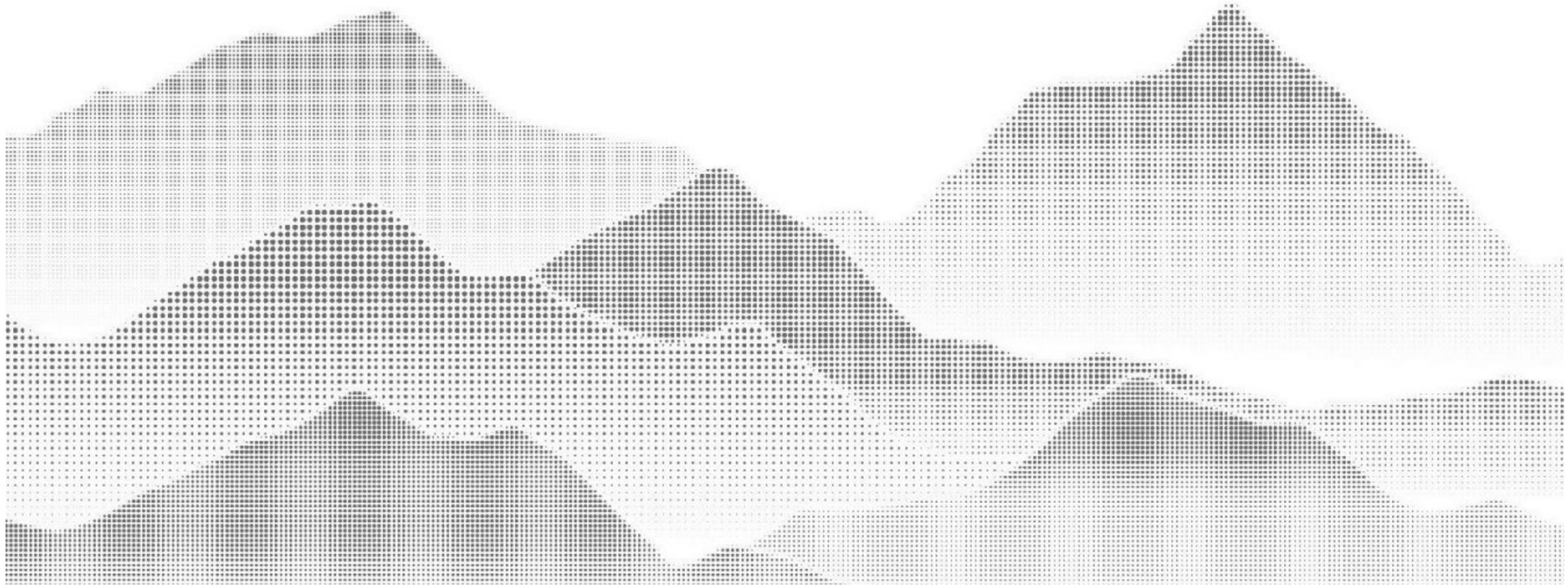
MCMC can get stuck in **local optima**, especially in complex phylogenetic landscapes. MCMC actually involves running **multiple chains** at the same time:

Cold Chain: represents the standard MCMC chain.

Heated Chains: explore a flattened posterior for broader exploration of the parameter space.

Hot escape local optima by exploring unlikely regions, **cold** refine high-probability trees.

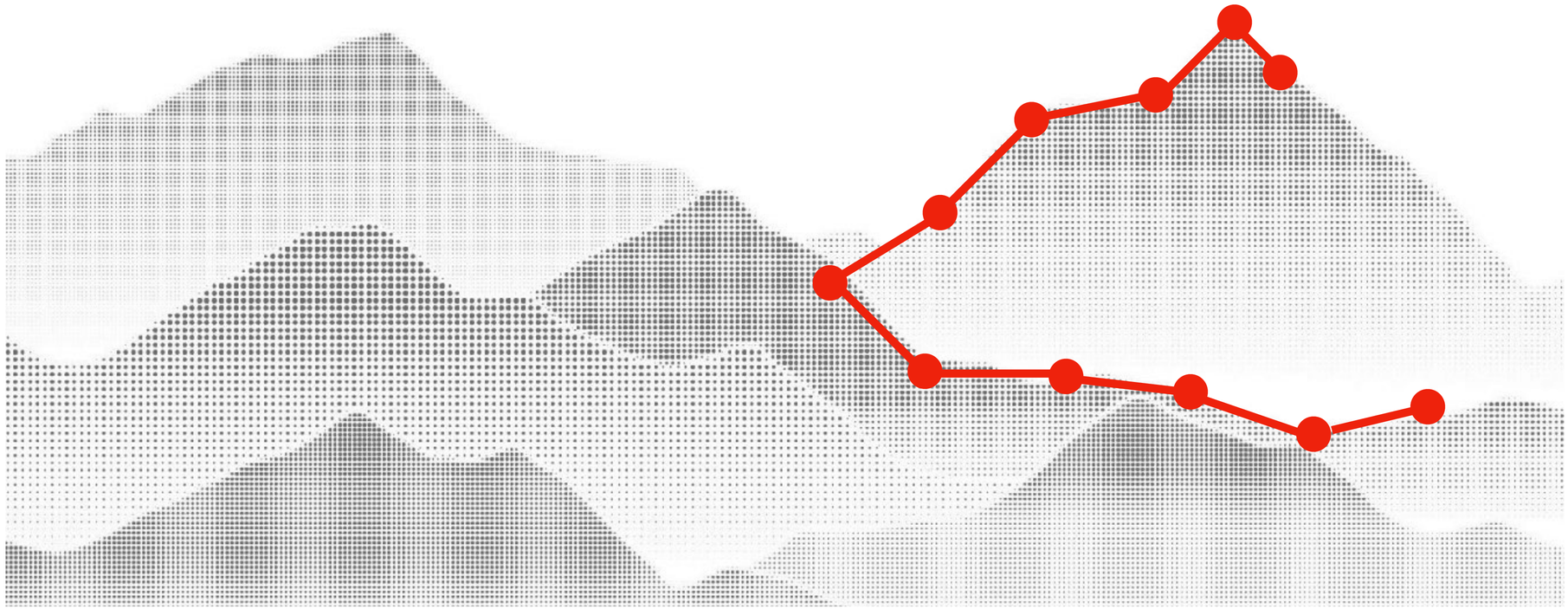
The likelihood function is raised to the power of 1/T, where T is the **temperature**. When T>1 the likelihood values become less extreme, allowing more movement between different trees.



Burn-in refers to the initial set of iterations where the chain may not have yet reached the target distribution.

These early samples do not represent the target / stationary distribution, which provide accurate estimates of the parameters of interest.

Typically, 10-25% of the MCMC samples are removed, but the exact percentage depends on convergence **diagnostics**.



Among the post-inference diagnostics **Effective Sample Size (ESS)** are the more important.

ESS estimates the **number of independent samples** in an MCMC chain.

Due to **autocorrelation**, MCMC samples are not fully independent.

Autocorrelation means that **consecutive samples are similar**.

Basically, we can think to it as a chain that is still going uphill.

If the ESS of a parameter is small, then the **estimate of its posterior distribution will be poor**.

What size ESS is adequate? Basically, the larger the better. ESSs < 100 are considered weak and having ESSs > 200 would be better. On the other hand, chasing ESSs > 10000 may simply be a waste of computational resources.

How can I make ESS better? Improved proposal strategies can enhance the exploration of the parameter space, reducing autocorrelation. Increased MCMC run length can help achieve a sufficient number of independent samples, mitigating the effects of autocorrelation.

Should all parameters have high ESS? Not necessarily. While low ESS values can indicate poor mixing, if parameters of interest have adequate ESS, occasional low ESS in other parameters might be acceptable. However, key metrics like likelihoods should maintain high ESS for robust analysis.

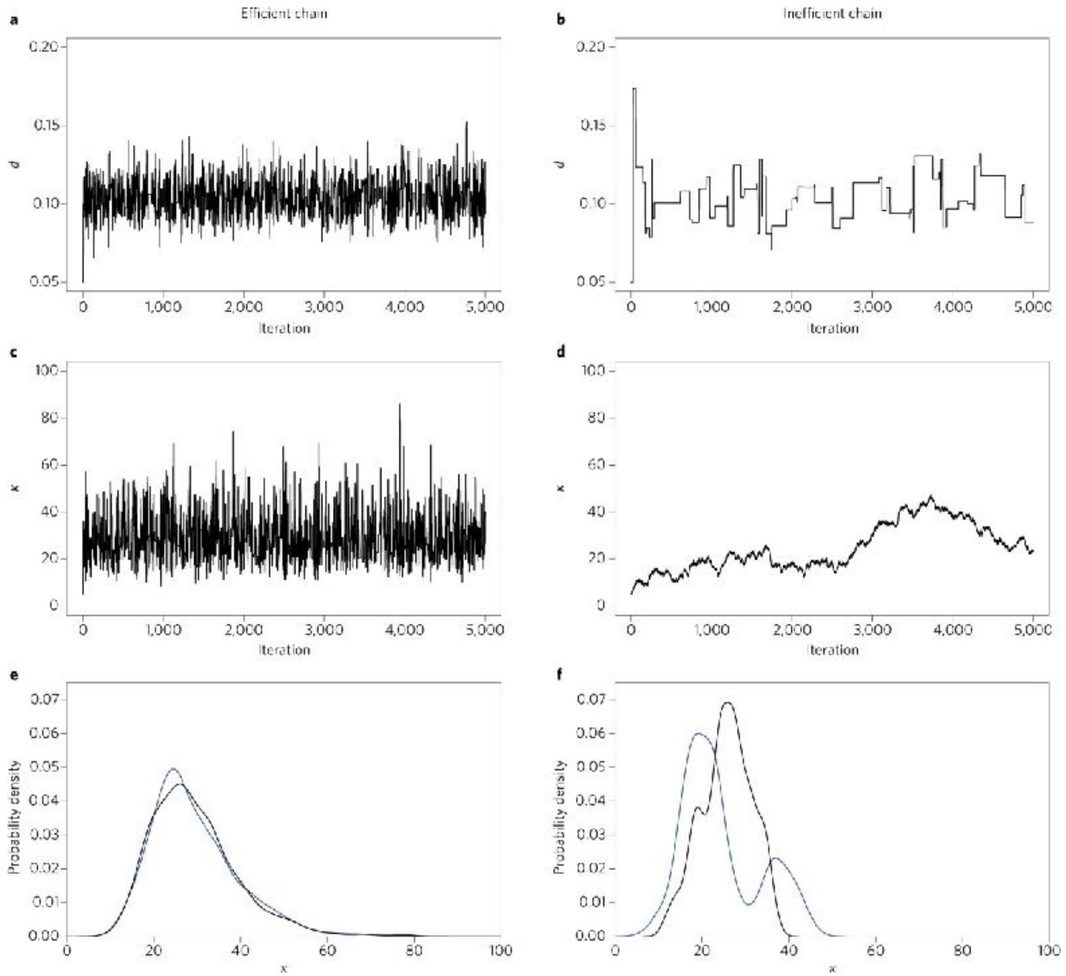


Fig. 2 | Trace plots and histograms for d and κ from sampling a posterior distribution using efficient and inefficient MCMC chains. The posterior distribution used is shown in Fig. 1c. **a**,**c**, Trace plots of d (**a**) and κ (**c**) for an efficient chain with good mixing. The window sizes are $w_d = 0.12$ and $w_\kappa = 180$, with acceptance proportions $P_{jump} = 30.4\%$ for d and 29.8% for κ, achieving Eff = 23% for d and 20% for κ. **b**,**d**, The corresponding trace plots for an inefficient chain with poor mixing, with $w_d = 5$ and $w_\kappa = 1$. In **b** the window for d is too wide, and most proposals are rejected ($P_{jump} = 1.5\%$), so the chain is often stuck at the same value for many iterations, leading to poor mixing with Eff = 1.79%. In **d** the window for κ is too small, so most of the proposals are accepted (with $P_{jump} = 98.6\%$), but the chain makes small baby steps and is very slow in traversing the posterior parameter space, with Eff = 1.28%. **e**,**f**, Histograms of κ for two runs (shown by the black and blue lines) of the efficient (**e**) and inefficient (**f**) chains (sample size $n = 10{,}000$). The posterior mean (and standard deviation) calculated using a very long run of the efficient chain is 0.104 (0.0114) for d, and 29.2 (10.0) for κ.

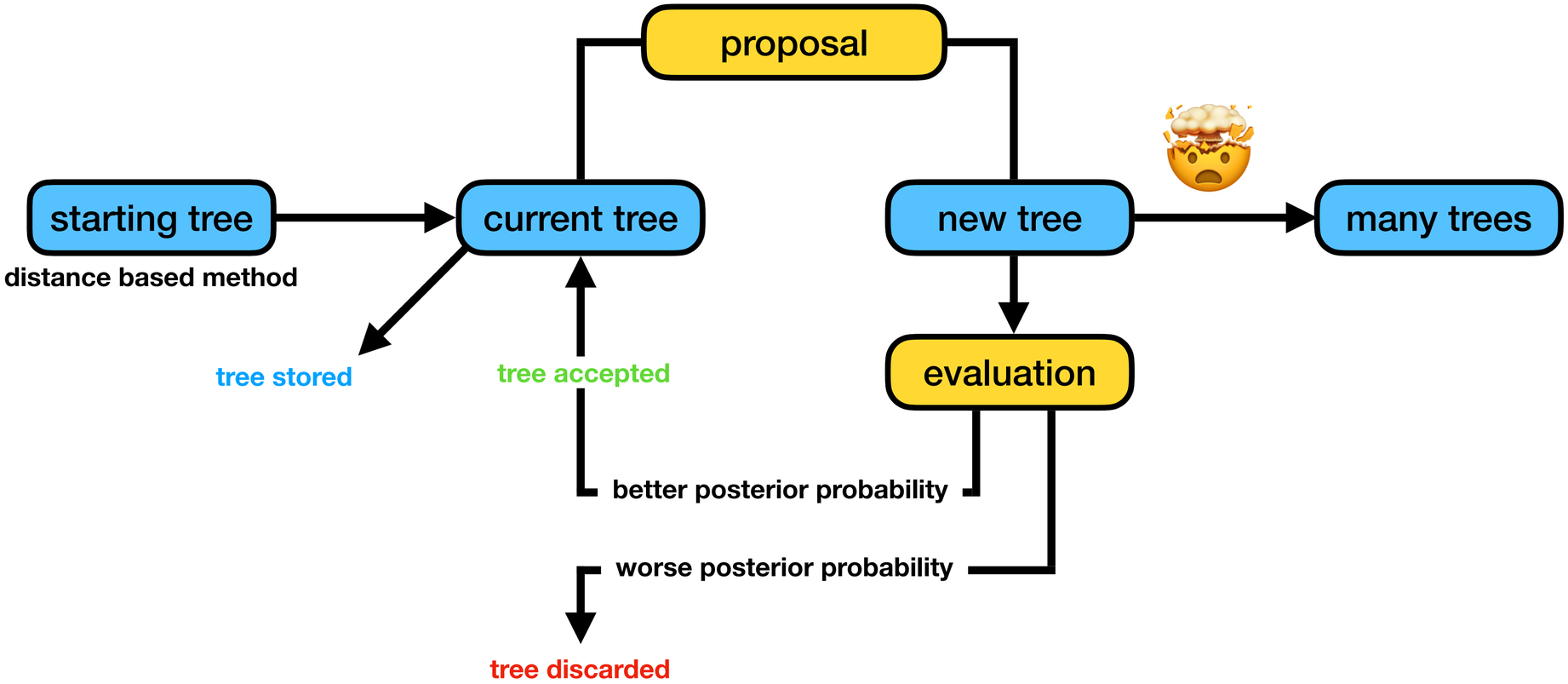
… we end up with a ton of trees! 🌳🌴🌲

… and then we need to summarize the posterior distribution of tree.

Majority-Rule Consensus Tree: this method identifies clades that appear in more than 50% of the posterior trees. It provides a summary of well-supported clades but may result in a tree topology that was not explicitly sampled during the analysis.

Maximum Clade Credibility (MCC) tree: the single tree from the posterior distribution that has the highest product of posterior probabilities across all its clades and where the combined support for all its clades is maximized. A poorly supported clade can significantly reduce the tree credibility.

Maximum Sum of Clade Credibilities (MSCC) tree: the tree with the highest sum of posterior probabilities for its clades. Less sensitive to individual low-probability clades than MCC. However, it favor trees with more clades, as each additional clade contributes positively to the total sum.


proposal
starting tree
distance based method
current tree
new tree
many trees
tree stored
tree accepted
evaluation
better posterior probability
worse posterior probability
tree discarded

FINISH