Streamlining the calculation of likelihood under PIP

Recall that the classical formula to compute the likelihood of an alignment-tree pair (MSA, τ) within the PIP model is

$$\mathcal{L}(\text{MSA} \mid \tau) = \varphi(\mathbb{P}(c_\emptyset \mid \tau), |\text{MSA}|) \prod_{c \in \text{MSA}} \mathbb{P}(c \mid \tau),$$

where

$$\varphi(p, m) = \frac{1}{m!} \|\nu\|^m e^{\|\nu\|(p-1)}.$$

Within this document we assume the model parameters (i.e. insertion rate λ, deletion rate μ and substitution matrix Q) and the weighted tree τ to be fixed, and we will use notation $p(X)$ for conditional probability of event X under those parameter values. We begin with writing $\varphi(\mathbb{P}(c_\emptyset \mid \tau)$ explicitly and regrouping the terms.

$$\mathcal{L}(\text{MSA} \mid \tau) = \frac{1}{|MSA|!} \|\nu\|^m e^{\|\nu\|(p(c_\emptyset)-1)} \prod_{c \in \text{MSA}} p(c) = \frac{1}{|MSA|!} e^{(\|\nu\| p(c_\emptyset)) - \|\nu\|)} \prod_{c \in \text{MSA}} (\|\nu\| p(c)).$$

Taking the logarithm results in a formula for the log-likelihood

$$l(\text{MSA}) = \log \mathcal{L}(\text{MSA} \mid \tau) = -\log(|\text{MSA}|!) + (\|\nu\| p(c_\emptyset) - \|\nu\|) + \sum_{c \in \text{MSA}} \log(\|\nu\| p(c)).$$

Note that $\log(|\text{MSA}|!)$ can be calculated using Stirling's approximation

$$\log(m!) \approx m \log m - m + \frac{1}{2} \log m + \log \sqrt{2\pi} + \frac{1}{12m}.$$

In practical cases, the error introduced by this approximation is negligible, so it suitable for our needs. Since changing all log-likelihoods by a constant value $\log \sqrt{2\pi}$ has no effect on the maximal likelihood method, we choose to remove it, but this change is subject to discussion.

We now turn our attention to computing $p(c)$ and $p(c_\emptyset)$, starting with $p(c)$. We recite the formulae of the norm of the intensity measure and the insertion and survival probailities.

$$\|\nu\| = \lambda(\frac{1}{\mu} + \|\tau\|),$$

$$\iota_v = \begin{cases} \frac{b(v)}{1/\mu + \|\tau\|} & \text{if } v \text{ is not the root,} \\ \frac{1/\mu}{1/\mu + \|\tau\|} & \text{if } v \text{ is the root.} \end{cases}$$

$$\beta_v = \begin{cases} \frac{1 - e^{-b(v)\mu}}{b(v)\mu} & \text{if } v \text{ is not the root,} \\ 1 & \text{if } v \text{ is the root.} \end{cases}$$

Here $b(v)$ denotes the length of the branch connecting the node v to its parent. Recall that for non-empty columns c *(do we handle MSAs with completely empty columns properly?)* we have

$$p(c) = \sum_{v \in A(c)} \iota_v f_v = \sum_{v \in A(c)} \iota_v \beta_v \langle \tilde{f}_v, \sigma \rangle$$

(we use σ instead of π to denote stationary frequencies to avoid confusion with the mathematical constant),

$$\|\nu\|p(c) = \sum_{v\in A(c)} \|\nu\|\iota_v\beta_v\langle\tilde{f}_v,\sigma\rangle.$$

We introduce a new array

$$w_v = \|\nu\|\iota_v\beta_v = \begin{cases} \lambda\frac{1-e^{-b(v)\mu}}{\mu} & \text{if } v \text{ is not the root,} \\ \frac{\lambda}{\mu} & \text{if } v \text{ is the root} \end{cases}$$

which will replace three arrays ι_v, β_v, f_v and allows us to compute the likelihood of a non-empty column with fewer multiplication-divison operations. Note that the total intensity (or, equivalently, total tree length) no longer needs to be maintained to do these calculations.

The same optimisation is applicable for the empty column. For $c_\emptyset$ we have

$$p(c_\emptyset) = \sum_{v\in V(\tau)} \iota_v f_v = \sum_{v\in V(\tau)} \iota_v(1-\beta_v(1-\langle\tilde{f}_v,\sigma\rangle)).$$

Note that we abuse the notation and redefine $\tilde{f}$ in this formula, but values of ι_v and β_v depend only on tree structure and not the alignment.

$$\|\nu\|p(c_\emptyset) = \sum_{v\in V(\tau)} \big(\|\nu\|\iota_v + \|\nu\|\iota_v\beta_v(\tilde{f}_v - 1)\big) = \|\nu\| + \sum_{v\in V(\tau)} w_v(\langle\tilde{f}_v,\sigma\rangle - 1)\big).$$

In the last transition we use the fact that insertion probabilities ι_v add up to 1. The norm $\|\nu\|$ in the formula now cancels with the exterior $-\|\nu\|$. Moreover, since deletion rate is constant, the value of Felsentein's dynamic $\tilde{f}_v$ does not depend on the initial parent symbol. To avoid matrix multiplication, we compute the value of $\tilde{f} - 1$ for a "unary alphabet" (that is, we only track whether a character is extinct or not) using the same dynamic programming algorithm and denote the result by $\tilde{f}_{1v}(c_\emptyset)$. All our optimisations combine into

$$l(\mathrm{MSA}) + \log\sqrt{2\pi} \approx \sum_{v\in V(\tau)} w_v\tilde{f}_{1v}(c_\emptyset) + \sum_{c\in\mathrm{MSA}} \log\Big(\sum_{v\in A(c)} \big(w_v\langle\tilde{f}_v(c),\sigma\rangle\big)\Big) -$$

$$- \big(|\mathrm{MSA}|\log|\mathrm{MSA}| - |\mathrm{MSA}| + \frac{1}{2}\log|\mathrm{MSA}| + \frac{1}{12|\mathrm{MSA}|}\big).$$