Explanation of Methods for Lineage Fitting

Thomas House

1 Definition of relative growth advantage

Suppose we have two species with exponential growth rates r_1 and r_2 and therefore expected counts

$$\mu_1(t) = \mu_1(0)\mathrm{e}^{r_1 t} \ , \qquad \mu_2(t) = \mu_2(0)\mathrm{e}^{r_2 t} \ , \tag{1}$$

respectively. The ratio of these species is

$$\frac{\mu_1(t)}{\mu_2(t)} = \frac{\mu_1(0)}{\mu_2(0)}\mathrm{e}^{(r_1-r_2)t} \ . \tag{2}$$

Therefore, this ratio grows exponentially with a rate we call the *relative growth advantage* of species 1 against 2, $r_1 - r_2$. In particular, the ratio will double after a time period

$$\tau_D = \frac{\log(2)}{r_1 - r_2} \ .$$

We are interested in determining an instantaneous relative growth advantage estimate for the case where there is not exponential growth as in (1) over a sustained period of time, but which will reduce to the constant-rate result as a special case. If we let $\mu_1(t)$ and $\mu_2(t)$ be more general functions of time, then defining

$$\delta r = \frac{\mathrm{d}}{\mathrm{d}t}\log\left(\frac{\mu_1(t)}{\mu_2(t)}\right) \tag{3}$$

achieves this since from (2) we have

$$\delta r = r_1 - r_2$$

in the constant-rate case.

2 Data

Suppose we have a length-n vector $\mathbf{X}$ whose i-th element, X_i is the time at which a sample is taken, and a length-n vector $\mathbf{y}$ whose i-th element, y_i indicates whether the sample is species 1 or 2, i.e.

$$y_i = \begin{cases} 1 & \text{if sample is species 1,} \\ 0 & \text{if sample is species 2.} \end{cases}$$

We will also define the counts of times in terms of

$$t_{\min} = \min(\mathbf{X}) \ , \qquad t_{\max} = \max(\mathbf{X}) \ ,$$

as

$$Z_1(t) := \sum_{i=1}^{n} y_i \mathbf{1}_{\{X_i = t\}} \ , \quad Z_2(t) := \sum_{i=1}^{n} (1-y_i)\mathbf{1}_{\{X_i = t\}} \ , \quad t \in \{t_{\min}, \ldots, t_{\max}\} \ , \tag{4}$$

where $\mathbf{1}$ is the indicator function. So $Z_a(t)$ is the observed count of species a at time t.

3 Estimate from proportions

Suppose we knew the probability of a uniform random sample from the population being of species 1,

$$\pi(t) := \frac{\mu_1(t)}{\mu_1(t) + \mu_2(t)} \ . \tag{5}$$

This can be related to the instantaneous relative growth advantage using the **log odds**, $f(t)$ of being species 1 over time; using the standard definition of these then substituting (1) into (5) we obtain

$$f(t) := \log\left(\frac{\pi(t)}{1-\pi(t)}\right) = \log\left(\frac{\mu_1(t)}{\mu_2(t)}\right) \ . \tag{6}$$

So from (3), we have

$$\delta r = \frac{\mathrm{d}}{\mathrm{d}t}\log\left(\frac{\pi(t)}{1-\pi(t)}\right) \ . \tag{7}$$

In practice, we do not measure $\pi(t)$ directly, but will instead typically estimate it using a non-parametric method. Because we wish to differentiate a complex function of the time trend as in (6), traditional splines that have penalised or zero derivatives may not be appropriate, and so we place a Gaussian process prior on f with the Radial Basis Function (RBF) kernel, which has C_∞ samples (i.e. all derivatives exist).

To implement this we can use the approach from Chapters 3 and 5 of Rasmussen and Williams (2005), as implemented in Scikit-learn's *GaussianProcessClassifier* class (Pedregosa et al., 2011). This returns an estimate for $\pi(t)$ as well as optimised hyperparameters for the RBF kernel. To assess uncertaintly, we are most interested in the role of finite data size and the distribution over possible trajectories of the relative growth advantage, and so bootstrap the data vectors $\mathbf{X}$ and $\mathbf{y}$, then use the kernel hyperparameters optimised on real data to produce an ensemble of bootstrapped curves for $\pi(t)$. For the original data and bootstrapped curves, we can then also produce estimates of growth advantage from Equation (7).

4 Estimate from counts

Gaussian process classification is very memory intensive as implemented, and so we find that for $n > 10^4$ it is preferable to work with the counts directly. Suppose we have smoothers $s_1(t)$ and $s_2(t)$ estimating $\log(\mu_1(t))$ and $\log(\mu_2(t))$ respectively. Expanding (3) and substituting in gives

$$\delta r = \frac{\mathrm{d}}{\mathrm{d}t}(\log(\mu_1(t))) - \frac{\mathrm{d}}{\mathrm{d}t}(\log(\mu_2(t))) = \frac{\mathrm{d}s_1}{\mathrm{d}t} - \frac{\mathrm{d}s_2}{\mathrm{d}t} \ . \tag{8}$$

We then need to estimate the smoothers. If we assume that the count observed over a day, z, is generated by a Poisson process of rate μ, then it is a standard result that the posterior distribution is

$$p(\mu) = \mathrm{Gamma}(\mu|\alpha = z+1; \beta = 1) \ . \tag{9}$$

The mean of this distribution is $\alpha/\beta = z+1$. We therefore construct vectors $\mathbf{t} = (t)$, $\mathbf{y}_1 = (\log(Z_1(t)+1))$ and $\mathbf{y}_2 = (\log(Z_2(t)+1))$ where $t \in \{t_{\min}, \ldots, t_{\max}\}$ for the independent variable, and dependent variables for species 1 and 2 respectively. We then obtain the smoothers required by (8) through Gaussian process regression as implemented in Scikit-learn's *GaussianProcessRegressor* class (Pedregosa et al., 2011). As for the classification method, we assess uncertainty through bootstrapping the data.

References

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, Cambridge, Massachusetts, 2005.