A tree- and model-based composition fit test

Peter G. Foster *

January 30, 2004

A model fit test ≠ homogeneity test

The tree- and model-based compostion fit test described here asks whether the composition of the data fits the composition of the model on the tree. It is related to, but different from, composition homogeneity tests that ask whether the data are homogeneous in composition. One difference is that the tree- and model-based composition fit test can be applied to heterogeneous data with a heterogeneous model.

A model fit test such such as that described here could be a compositional homogeneity test if the model is homogeneous. For example, if the model is homogeneous and the composition of the model is the observed composition of the data, then the model fit test described here *is* a compositional homogeneity test. However, the model fit test is more general. Consider DNA data that truly is homogeneous in composition, for example exactly 25% each of the four nucleotides in all the sequences. If we ask whether the data are homogeneous, the answer will be yes. If we ask whether the data fits a Jukes-Cantor model composition, a model fit test, the answer will also be yes, and that also says that the data are homogeneous in composition. However, we could ask whether the data fit a homogeneous F81 model with a wildly different composition, and the answer would be no (which incidentally says nothing about whether the data are homogeneous in composition). Both the data and the model are homogeneous, but the composition of the model does not fit the composition of the data, and so the data fails the model fit test.

Consider a DNA alignment that is heterogeneous, perhaps where one or more of the sequences has a very different composition to the rest of the data. If we test for compositional homogeneity the data fail the test, showing that they are heterogeneous. We might then use the tree- and model-based composition fit test to ask whether the data fits the composition of a particular heterogeneous model. Of course a composition homogeneity test would be unable to answer that question, and so we need the model fit test.

The X_m^2 statistic

In the χ^2 test for compositional homogeneity, we use the statistic X^2 (in the sense used by Sokal and Rohlf, Biometry, *cf* χ^2, which is a distribution), where $X^2 = \sum[(obs - exp)^2/exp]$, where the expected values are from the mean composition of the data. This is the statistic that PAUP gives you when you do the `basefreq` command. The composition fit test described here uses a similar statistic X_m^2, where the m subscript indicates that the expected values come from the model, not from the data. X_m^2 is calculated in the same way as X^2, except for the origin of the *exp* values. In a model that is heterogeneous over the tree, expected values can differ in different taxa.

*Department of Zoology, The Natural History Museum. email: `p.foster@nhm.ac.uk`

Calculation of the expected compostion

In a homogeneous model, the expected composition is simply the composition of the model. Here I describe how to calculate the expected composition of sequences on a tree under a heterogeneous composition. For a start, let's consider a DNA sequence composed of all A's that evolves under a Jukes-Cantor model on a branch length of 0.1 mutation/site. The probability of the A staying as an A is 0.906380 and the probability of the A changing to something else is 0.031207 for each of the other nucleotides. So the expected compostion of the resulting sequence is $\boldsymbol{\pi}_{exp} = \begin{bmatrix}0.906380 & 0.031207 & 0.031207 & 0.031207\end{bmatrix}$ (here and elsewhere the order of the bases is A, C, G, and T). If the starting sequence was 40% A and 60% C, the expected composition of the resulting sequence would be

$$\begin{aligned}\boldsymbol{\pi}_{exp} &= 0.4 \times \begin{bmatrix}0.906380 & 0.031207 & 0.031207 & 0.031207\end{bmatrix} \\ &\quad + 0.6 \times \begin{bmatrix}0.031207 & 0.906380 & 0.031207 & 0.031207\end{bmatrix} \\ &= \begin{bmatrix}0.3812762 & 0.5563108 & 0.031207 & 0.031207\end{bmatrix}\end{aligned}$$

We can generalize to the GTR matrix. For a starting sequence with composition $\boldsymbol{\pi}$ over a branch with probability matrix $\boldsymbol{P}$, the composition of the resulting sequence is the sum of the columns of $\boldsymbol{\pi P}$. For example, using the notation as given in Swofford's chapter in Hillis96 (SOWH96), if

$$\boldsymbol{R} = \begin{bmatrix} - & 2 & 3 & 4 \\ 2 & - & 5 & 6 \\ 3 & 5 & - & 1 \\ 4 & 6 & 1 & - \end{bmatrix}$$

and the composition of the model is

$$\boldsymbol{\pi}_m = \begin{bmatrix}0.4 & 0.3 & 0.2 & 0.1\end{bmatrix}$$

and the composition of the ancestor sequence is

$$\boldsymbol{\pi}_a = \begin{bmatrix}0.3 & 0.2 & 0.1 & 0.4\end{bmatrix}$$

then the $\boldsymbol{Q}$ matrix, after normalizing so that branch lengths will be 1 mutation per site, will be

$$\boldsymbol{Q} = \begin{bmatrix} -0.877863 & 0.152672 & 0.114504 & 0.610687 \\ 0.229008 & -1.335878 & 0.190840 & 0.916031 \\ 0.343511 & 0.381679 & -0.877863 & 0.152672 \\ 0.458015 & 0.458015 & 0.038168 & -0.954198 \end{bmatrix}$$

and the $\boldsymbol{P}$ matrix for a branch length of 0.1 will be

$$\boldsymbol{P} = \begin{bmatrix} 0.933884 & 0.025548 & 0.024525 & 0.016042 \\ 0.034065 & 0.902757 & 0.039709 & 0.023469 \\ 0.049050 & 0.059564 & 0.886385 & 0.005001 \\ 0.064168 & 0.070407 & 0.010002 & 0.855423 \end{bmatrix}$$

and $\boldsymbol{\pi}_a \boldsymbol{P}$ will be

$$\boldsymbol{\pi}_a \boldsymbol{P} = \begin{bmatrix} 0.280165 & 0.007665 & 0.007358 & 0.004813 \\ 0.006813 & 0.180551 & 0.007942 & 0.004694 \\ 0.004905 & 0.005956 & 0.088638 & 0.000500 \\ 0.025667 & 0.028163 & 0.004001 & 0.342169 \end{bmatrix}$$

and the expected $\boldsymbol{\pi}$ will be the sum of the columns

$$\boldsymbol{\pi}_{exp} = \begin{bmatrix} 0.317551 & 0.222335 & 0.107939 & 0.352176 \end{bmatrix}$$

It is not a stationary process, so the composition for that node is somewhere in between $\boldsymbol{\pi}_a$ and $\boldsymbol{\pi}_m$. Its on a short branch, so the expected composition is closer to $\boldsymbol{\pi}_a$ than it is to $\boldsymbol{\pi}_m$. The longer the branch, the more the expected composition will approach $\boldsymbol{\pi}_m$.

The expected composition of child nodes can be calculated in the same way, where $\boldsymbol{\pi}_a$ will be the expected composition that we just calculated. In this way the compositions of all the nodes can be calculated, starting with the root and working out to the tips, in pre-order.

If there is among-site rate variation, then that needs to be taken into account. If there is a proportion of invariant sites, then that proportion of sites remains at a constant composition in all the calculations above. If there is discrete gamma-distributed among site rate variation of the variable sites, then it is assumed that each site in the data stays in the same rate category in each node.

The test

Now that we have the expected compositions in all the nodes, we can use it in calculating the X_m^2 statistic for the tip nodes. A null distribution of X_m^2 values is made with simulations, to which the realized X_m^2 value from the original data can be compared. The realized value is considered significant if it is larger than 95% of the null distribution.

If we are dealing with a model that has a heterogeneous composition, since it is not stationary, it is best to estimate the model compositions by ML. Since the composition values bear on the value of X_m^2, it is not sufficient to use emprical compositions. Branch lengths and other model parameters also bear on this value, and so everything needs to be optimized. This is true for the simulations used for the null distribution as well, which is unfortunate because it makes the test slow.