Massachusetts Institute of Technology
6.877 Computational Evolutionary Biology, Fall, 2005
Laboratory 3: Detecting selection

Handed out: November 28 **Due: December 14**

Part 2. Detecting selection – likelihood methods – PAML (Phylogenetic Analysis by Maximum Likelihood)

If one wants to be even more sophisticated about our selection testing, then as we've seen, the most advanced methods use something like maximum likelihood. We can use this approach to test for specific sites under selection, or even specific lineages. The 'leader of the pack' here is PAML.

Part 2.1: warm-up with PAML – pairwise estimation of dN/dS and sensitivity of this ratio to assumptions

In this part of the lab, we shall get acquainted with the use likelihood methods to estimate dN/dS, aka 'omega,' ω, using the PAML program. We shall first see how the likelihood method works to estimate ω by having you figure out what the 'most likely' ω value is 'by hand,' and then have the PAML program do it for you. In addition, we'll be able to use PAML to test basic evolutionary assumptions about transition/transversion ratios and codon evolutionary changes.

What does PAML do? It's quite a sophisticated package, and includes modules to do all the following things and more:

- estimating synonymous and nonsynonymous rates
- testing hypotheses concerning dN/dS rate ratios
- various amino acid-based likelihood analysis
- ancestral sequence reconstruction (DNA, codon, or AAs)
- various molecular clock models
- simulating nucleotide, codon, or AA sequence data sets

We will only look at the first three components here.

PAML and likelihood methods.

Recall first that the likelihood method works by setting up some 'null hypothesis' and then calculating its likelihood, given some (sequence) data, and, possibly, a phylogenetic tree. We then specify some alternative hypothesis to test vs. this null case. PAML calls this scenario a model. The PAML model that we will first use for a warm-up is the PAML default model, dubbed Model 0: it assumes that the null hypothesis is no selection and it tests against this null model the possibility that there is selection (either positive or negative). (We shall see in section 2.3 below that by specifying other Models, one can emulate the scenarios of 'slightly neutral evolution' vs. selection; site-specific selection; and lineage specific selection.)

As described in class, we test the null hypothesis likelihood against an alternative hypothesis that is *nested*, or more specific than, the null hypothesis. Once again, in our case, the null hypothesis is that of no selection (omega equal to 1), while the alternative is that there is selection (omega greater than 1 or less than 1, greater than 0). Twice the log likelihood difference between these hypotheses is approximately distributed as a chi-square, so we can accept/reject the null hypothesis at a particular statistical significance level. (You can read some of the reference papers that discuss whether this chi-square assumption is valid or not.)

PAML incorporates a general codon Markov model that includes the possibility of transition/transversion ratios, probability of codon-codon transitions, and a possible selective force. In this section, we want to examine the sensitivity of no selection/selection to various assumptions about transition/transversion ratios and codon transition probabilities. As a preliminary, and to get you familiar with running one component of PAML, we'll first see how to run the basic program and see 'by hand' how it works.

Part 2.1.1 Pairwise estimation of a dN/dS ratio

Objective 1: Learn how to run PAML and in particular its `codeml` module.

Objective 2: 'Simulate' PAML's method to estimate dN/dS by evaluating its likelihood function for a variety of fixed values for the parameter ω, first, "by hand;" then estimate dN/dS via the PAML module `codeml`, which uses a hill-climbing algorithm to maximize likelihood.

Dataset: *GstD1* genes of *Drosophila melanogaster* and *D. simulans* (600 codons). This data file is located in the labs section. Please download this to your local computer for input to the program.

Procedure:
Below we describe in a bit of detail how to obtain and run PAML. It's a sophisticated program. As a summary 'cheat sheet,' after you read through the 2-page summary given in the labs section.

Running PAML
PAML itself is extremely user-unfriendly – much like statistical programs of days gone by. For PAML you edit the control file (codonml.ctl) to change all parameters. The program then reads this control file and writes the output files – one file is a name you specify the other is called "rst." In more detail, to use PAML, one provides two or three files: (1) a control file which specifies the particular model and estimation method to use, as well as the location/type of input and output files; (2) a data sequence file (with the name of the sequence file actually specified in the control sequence file); and (3) an (optional) phylogenetic tree file, also specified in the control file (since in some cases PAML estimates its values based on a (given) phylogenetic tree). For our exercises, we will be using the `codeml` or "codon maximum likelihood" module in PAML. So the actual way you run the program is as follows, from a command line (either Linux, PC, or Mac terminal); this will spit back its results to the terminal as well as to an output file that you specify in the control file:

```
>codeml <name of control file>
```

Obtaining PAML
PAML will run on Windows, MacOSX, and Linux. There are three ways you can get binaries to run.

1. The basic PAML site is here: http://abacus.gene.ucl.ac.uk/software/paml.html
   and the download section is here: http://abacus.gene.ucl.ac.uk/software/paml.html#download

   The download file includes pre-compiled Windows and G5 Macintosh binaries – see the Readme file in the download. For other folks, the download page gives instructions for how to download and compile the source code for other Macintosh OS X machines and Linux; be sure to use gcc if you've got it, and optimize the code according to the README file.

2. I have also compiled binaries for Mac laptops, so please ask me for these if you want to run this on your Mac; if you want to compile it yourself, you'll need to have the Apple developer package installed. For linux, compilation is also fairly straightforward, as per the instructions on the download page.
3. If you want to use linux, an alternative is that I can burn some .iso disks based on Knoppix, that will boot on any x86 machine. Please email me if you want to go this route.

Consult the documentation file at http://abacus.gene.ucl.ac.uk/software/pamlDOC.pdf – beware, it's long, you should read just the first few pages plus page 21 on, re the `codeml` package.

Another useful source is the PAML FAQs pdf, available at:
http://abacus.gene.ucl.ac.uk/software/pamlFAQs.pdf

Running PAML

If PAML is in the PATH of your environment variables (eg, the Windows PATH variable; or Linux/MacOS X PATH), then you can just invoke `codeml` as follows, where the 'control file' is the name we have given to the specifications to be given to PAML for this exercise (see just below). You'll have to modify this file a tiny bit in this exercise:

```
>codeml paml-exercise1.ctl
```

The input file formats

As mentioned, PAML requires two (and optionally three) input files: a control file (discussed below); a (DNA, codon, or amino acid) aligned sequence file; and an (optional) phylogenetic tree file (for those hypotheses that require testing for different evolutionary rates along different lineages).

A sample (and simple) codon/DNA sequence file looks like the following (see page 10 and 11 of the PAML pdf documentation file). The number of sequences is followed by one or more spaces, then the number of nucleotides (in the case of a codon analysis done by `codem`), then a <cr>. This is followed by the sequences themselves. (See pages 10-14 of the PAML pdf documentation file for additional details.)

```
4 20
sequence_1  TCATTCTATCTATCGTGATG
sequence_2  TCATTCTATCTATCGTGATG
sequence_3  TCATTCTATCTATCGTGATG
sequence_4  TCATTCTATCTATCGTGATG
```

A pre-specified phylogenetic tree file, which one uses to test the possibility of different evolutionary rates, is provided in a parenthesized format. You won't be using a tree file for the first two PAML exercises. (See pages 13-15 of the PAML documentation for additional details.)

```
4 5 // 4 species, 5 trees
(1,2,3,4); // the star tree
((1,2),3,4); // species 1 and 2 are clustered together
((1,2),3,4); // Commas are needed with more than 9 species
((human,chimpanzee),gorilla,orangutan);
((human:.1,chimpanzee:.2):.05,gorilla:.3,orangutan:.5);
```

Understanding the control file

The PAML control file specifies the input and output file locations as well as the type of data and model tests to run. (In the documentation this file is usually given with a `.ctl` suffix, but it can be any text file.) Typically, not all control options are used; in this example, I have stripped out

most of them. Let's take a look at the control file that you will use for this first exercise, and see how it is put together. Your job: modify the last section of this control file for this particular exercise, by commenting out and uncommenting single lines to 'run through' a variety of possible choices for dN/dS ratios. The file itself can be downloaded from the labs section.

```
  seqfile = paml_gstd1_seqfile.txt  * sequence data filename (replace)
  outfile = your_results_file.txt   * main result file name to write to

    noisy = 9      * 0,1,2,3,9: how much rubbish on the screen; 0:concise
  verbose = 1      * 1: detailed output; 0: concise output
  runmode = -2     * -2:pairwise estimation of dN/dS (if omega is estimated)

  seqtype = 1      * 1: use codons (ie, assume nucleotide triplets)
CodonFreq = 3      * 0: equal codon freq, 1:F1X4, 2:F3X4, 3:F61
    model = 0      * 0: one omega ratio for all branches
  NSsites = 0      * dN/dS ratio among sites. 0:no variation, 1:neutral, 2:positive
    icode = 0      * 0:universal genetic code

fix_kappa = 0      * 1:kappa (transversion rate) fixed, 0:kappa estimated
    kappa = 2      * initial value for kappa estimate or the fixed kappa value

fix_omega = 1      * 1:omega fixed, 0:omega to be estimated
    omega = 0.001  * 1st fixed omega value, to be commented out on subsequent runs

   *alternate fixed omega values -  you uncomment these for each run as described
   *omega = 0.005  * 2nd fixed value
   *omega = 0.01   * 3rd fixed value
   *omega = 0.05   * 4th fixed value
   *omega = 0.10   * 5th fixed value
   *omega = 0.20   * 6th fixed value
   *omega = 0.40   * 7th fixed value
   *omega = 0.80   * 8th fixed value
   *omega = 1.60   * 9th fixed value
   *omega = 2.00   * 10th fixed value
```

Let's run through the four main sections of this control file.

(1) The first section contains the specs for the input/output files. The `seqfile` and `outfile` specs say where the system will read its sequence data from, and where it will write its results. (These should be files in your own directory – please copy the sequence data for Drosophila from the lab directory to your own dir.) Note the spaces around the '=' sign, and the use of * as a 'begin comment' character for a command line. In addition, this beginning section can also include a `treefile` specification, where the program can find the file that gives the tree topology to assume for the analysis.

```
  seqfile = paml_gstd1_seqfile.txt  * sequence data filename (replace)
  outfile = your_results_file.txt   * main result file name to write to
```

(2) The next section of 3 lines controls the level of verbiage to terminal output, the results file, and what 'mode' the program will use. 0 gives the lowest level of output; the values given in the file above are fine for our purposes. The 'run mode' for our exercise should be -2, as this is what we want to (eventually) estimate dN/dS by maximum likelihood.

```
    noisy = 9      * 0,1,2,3,9: how much rubbish on the screen; 0:concise
  verbose = 1      * 1: detailed output; 0: concise output
  runmode = -2     * -2:pairwise estimation of dN/dS (if omega is estimated)
```

(3) Next, we specify properties for the sequence data, and then the two key parameters, the `model` and `NSites` values. The seqtype, CodonFreq, and icode values should be readily understandable. (We will assume equal codon frequencies, i.e., 1/61; if there is systematic bias that you know about, you can put it in, or assume unequal base frequencies – see the documentation.) What

about the remaining two parameters? The value for `model` tells the system what to set up as the basic null and contrasting assumptions about the dN/dS ratios among branches in the (possibly null) lineages. `Model = 0` means that the system will assume that there is one ratio for all lineages (branches). Thus, a value of 0 is the basic one to use if we are simply testing all sites, rather than phylogenetic variation. This establishes (part of) hypothesis whose likelihood will be computed. The `NSites` value sets up the remainder of the hypothesis to test. If `NSites` is set to 0, this is the hypothesis that there is no variation among any of the sites in the sequences. Thus, a combination of model and NSites values at 0 establishes a null hypothesis that there is one dN/dS ratio and there is no variation at any of the sites. (Please note that this is a bit distinct from saying that there is neutral variation, i.e., the hypothesis that dN/dS is 1!) This 0-0 combination is referred to as "M0" in the PAML author Yang's papers.

```
  seqtype = 1      * 1: use codons (ie, assume nucleotide triplets)
CodonFreq = 3      * 0: equal codon freq, 1:F1X4, 2:F3X4, 3:F61
    model = 0      * 0: one omega ratio for all branches
  NSsites = 0      * dN/dS among sites. 0:no variation, 1:neutral, 2:positive
    icode = 0      * 0:universal genetic code
```

(4) The control file ends with the specification of the transition/transversion ratio, kappa, and the dN/dS estimation:

```
fix_kappa = 0      * 1:kappa (transversion rate) fixed, 0:kappa estimated
    kappa = 2      * initial value for kappa estimate or the fixed kappa value

fix_omega = 1      * 1:omega fixed, 0:omega to be estimated
    omega = 0.001  * 1st fixed omega value
```

For this exercise, we will want to assume a kappa value estimated from the data, and start off the estimation at 2, so we set the parameter `fix_kappa` to 0 and the `kappa` value to 2.
Finally – and this is the key point of this exercise – since we want you to 'hand simulate' the discovery of the maximum likelihood value for dN/dS, we want to use a fixed dN/dS value and have the PAML program calculate what its likelihood is (given the other assumptions and the data). So, we will set `fix_omega` to be 1 and `omega` initially to be 0.001. Using this control file, the system will go ahead and output the log-likelihood that dN/dS is, in fact, 0.001.
You should now try running the `codeml` program and check to see that it works for you. For comparison, here's the output that I get, edited a bit. Note that PAML will compute the 'ordinary' counting dN/dS method due to Nei and Gojobori as well. The key result to note, of course, is the log likelihood value of -786.353736.

```
CODONML (in paml 3.14b, May 2005)
paml_gstd1_seqfile.txt   Model: One dN/dS ratio  omega = 0.001 fixed
Nei & Gojobori 1986. dN/dS (dN, dS)
(Note: This matrix is not used in later m.l. analysis. Use runmode = -2 for ML pairwise
comparison.)

[…omitted output about codons, etc.]

Mel
Sim                  0.2883 (0.0218 0.0755)

pairwise comparison, codon frequencies: Fcodon.

2 (Sim) ... 1 (Mel)
lnL = -786.353736
```

We hope that you get the same value as this for the log-likelihood – if not, please ask and we'll figure out what has happened…

Questions.
Now, to start the exercise in earnest, we want you to plot this value of the log likelihood against dN/dS, i.e., ω, with the likelihood on the y-axis and ω on the x-axis. In this case, you've got your first (x,y) point, with ω=0.001 and log-likelihood= -786. Now to complete the exercise, what you are to do is to comment out the line in the control file where ω was fixed at 0.001 by adding a * to the beginning of that line, and uncomment out the line below where omega is set to 0.005. Now re-run the program, record the log-likelihood, and repeat this procedure for each different value of ω as shown in the control file, commenting and uncommenting lines as appropriate. (Ah yes – you could write a script to do this also…) There will be 10 (x,y) values in all – plot them, and you should get an inverted U-shaped curve. Estimate where the maximum of this curve is. This is the maximum likelihood estimate of ω. Please report your findings in a table, the graph, and your estimate. Now, go back and see what to change to have the program estimate the likelihood value, to check your results – report what the program does in this case.

Part 2.2 Using PAML to compare hypotheses

Part 2.2.1 Basic use on MHC data: which model is better?

Objective:
Using PAML to test hypotheses.

Data:
MHC sequence data.

Procedure:
Now that you know how to use the program, let's do some real work. The major histocompatibility locus (MHC) is a well-known 'hot spot' for evolutionary change. Perform a pairwise comparison of sequences in the file `mhc.phy` to get maximum likelihood estimates of dN and dS, for each pair of sequences. Do this by making sure the associated `codonmlpair.ctl` specifies a pairwise comparison `"runmode = -2"`. You can download the file
from the labs section.

You can download the sequence file you need, in the proper format, from the labs section.

To assess the significance of your results, calculate the maximum likelihood from the same datafile, but now altering the control file, fixing the ω ratio = 1. Note that you'll have to do the remaining statistical work yourself: subtract the two log-likelihood results, multiply by –2, and look up the statistical significance in a chi-square table. (What is the number of degrees of freedom to use when looking in the chi-square table? It's the number of free parameters different between the null hypothesis and the more specific, 'nested' hypothesis, in this case, the difference between the 'fixed value' that ω=1 and letting ω vary, i.e., there is one free parameter. See page 2 of the 'crib sheet' or the PAML manual for additional information.)

Questions

1. What are the estimates of dN, dS and dN/dS that you obtain?
2. How many degrees of freedom are there between the estimated and fixed omega models?
3. Which model fits the data better, and why?
4. From this analysis would you conclude the genes have been subjected to positive selection? Why or why not?

Part 2.2.2 Using PAML to find out whether amino acid sites vary in selective pressure

Objective:
Using PAML to discover whether ω varies over sites or not (the null hypothesis is no variation over sites, which we want to reject at some significance level).

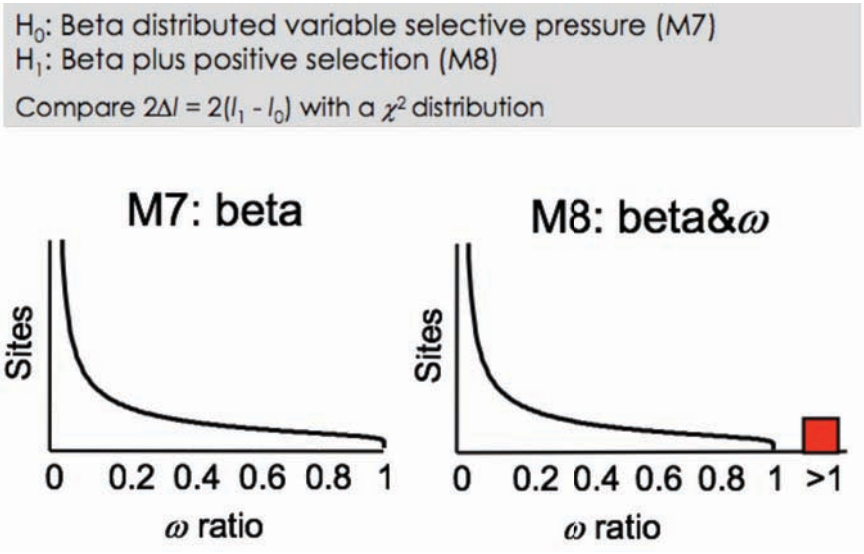
Data:
As in the previous section.

Procedure:
Perform an analysis for variation in the dN/dS between sites using the same `mhc.phy` sequence file but now using in addition `the mhc.tre` treefile. Change `runmode` = 0, `NSsites` = 0 1 2 7 8. This will run the models 0, 1, 2, 7, and 8 all at once. The parameter `ncatG` should be set automatically. See file `codonsites.ctl`. The models have the following interpretation:

1. M0 hypothesizes uniform selective pressure among sites
2. M1 hypothesizes variable selective pressure but no positive selection
3. M2 hypothesizes variable selective pressure **and with** positive selection
4. M7 hypothesizes selective pressure is variable **and** beta-distributed over sites (see picture)
5. M8 hypothesizes that selective pressure is variable, beta-distributed **and with** positive selection

You can download the relevant control file from the labs section.

You will note how M1 is 'nested within' (more restricted than) M0; M2 nested within M1; and M2 nested within M8. Here is a picture of the M7 vs. M8 'beta' distribution models, with the 'red box' indicating that there is some positive selection in some appreciable number of sites (this is to give you an idea of what the beta distribution is assuming about selective pressure and sites):



Since the maximum likelihood search method is not guaranteed to find a maximum, be sure to check convergence by performing the analysis with different starting ω values, one with ω much smaller than 1, say 0.1, and one with ω larger than 1 (say, 3 or 4). **Warning:** the maximum likelihood method for these models takes a bit longer than the previous excercises. Be patient.

Then, using your results, compare the likelihood of the different models (M1 vs. M2; M7 vs. M8), as in the previous exercise so as to answer the following questions.

Questions

1. How many degrees of freedom should you use for the comparison of M1 vs M2 and M7 vs M8? (Think about how many free parameters there are in the 'difference' between the two models).
2. Which model fits the data better, M1 or M2? M7 or M8?
3. What are the parameter estimates for the models with the highest likelihood?
4. From this analysis would you conclude the genes have been subjected to selection?
5. Which sites, if any, have been subjected to positive selection in the comparison M1 vs M2? (Note: this is found by looking at the posterior probabilities of the site classes in the two models. PAML identifies these and flags those statistically significant.)
6. Which sites, if any, have been subjected to positive selection in the comparison M7 vs M8?

Part 2.2.3 Using PAML to find out whether different lineages differ in selective pressure

Objective:
In the very first problem in this lab, we wanted to see whether the rate of evolution differed among different lineages – perhaps due to an adaptive burst. PAML can do this in a much more statistically rigorous way. Let's see how.

Data:
The same `mhc.phy` sequence file and the `mhc.tre` treefile as in the previous problem.

Procedure:
Set `NSsites` = 0. Run PAML with `model` = 1. See the file `codonmllineage.ctl`. This will estimate a different dN/dS ratio for each lineage. Compare the resulting likelihood to the model 0 likelihood above (one dN/dS ratio for each lineage, no variation between sites).
The control file is available in the labs section.

Questions

1. How many degrees of freedom between these models (the one you just ran and model 0)?
2. Which model fits the data better?
3. From this analysis, would you conclude the genes involved have been subjected to positive selection? Which lineages seem to be distinct? (If you want to probe more deeply into this, feel free to do so – you might also want to see how sensitive your results here and in the previous section are to variation in the codon estimation methods, and kappa.)

[This is the conclusion of the laboratory.]