

MCS255 - Lab 3

Probabilistic Models, Simulated Data, and Tree Reconstruction

Getting the .m files and the data files:

You should map your home directory (otherwise your work will be on the local machine you are using at the time), make a subdirectory for this course and maybe this lab, and save any files you want to save there.

The files you will need are available at

<http://www.gustavus.edu/~mmcdermo/mcs255/j06/lab/>.

Right click on the files (markovJC,

seqgen.m, distJCarray.m, distJC.m, and mutateg.m) and save them either to the subdirectory of your home directory that you just created or to C:\MATLAB7\work.

You will also need nj.m, compseq.m, distH.m, hamming.m, and seqdata.mat if you don't have them from last time.

Changing the current directory. MATLAB needs to know where the mfiles are stored to access them. You should change the current directory to where the files are located either by typing `cd z:\mcs255\lab` or `C:\MATLAB7\work` in the command window or by editing the Current Directory in the MATLAB toolbar.

1. In MATLAB, type `load seqdata` to read in some simulated sequence data. The three pairs of sequences, `s0` and `s1`, `t0` and `t1`, and `u0` and `u1`, are simulated ancestor and descendant sequences produced according to three different models. Which one was made according to the Jukes-Cantor model? The Kimura 2-parameter model? A general Markov model? Explain how you can tell. To easily compare sequences by producing a frequency array, use a command like `compseq(s0,s1)`.
2. Although the Jukes-Cantor model assumes $p_0 = (.25, .25, .25, .25)$, a Jukes-Cantor matrix could describe mutations even with a different p_0 . In this problem, you will investigate the behavior of a model using a Jukes-Cantor matrix as you vary p_0 .

Create an m-file (File → New → M File) with the following MATLAB commands:

```
a=.03
M=markovJC(a)
p=[.2; .3; .4; .1]
P=p
for i=1:10
p=M*p
P=[P p]
end
plot(P')
```

Run your m-file.

- (a) With the value of M and p_0 suggested, do you see p_t approach its equilibrium value? Increase the number of steps by editing the line `for i=1:10` in your m-file. Approximately how many time steps are necessary for all the p_t to be within .05 of the equilibrium? Within .01?
- (b) Using $p_0 = (.25, .25, .25, .25)$, what do you observe? Why?
- (c) Using $p_0 = (0, 1, 0, 0)$, what do you observe? What is the biological meaning of this p_0 ?
3. In this problem, you will simulate ancestral and descendant sequence data according to a probabilistic model in order to get some familiarity with the difference between the theoretical model and finite-length sequence data.

Create an m-file (File → New → M File) with the following MATLAB commands:

```
a=.1                % Markov matrix parameter
rd=[.25, .25, .25, .25] % root distribution of bases
n=100              % length of sequence
M=markovJC(a)     % Markov matrix
s0=seqgen(rd,n)   % ancestral sequence
s1=mutatef(s0,M)  % descendent sequence
F=compseq(s0,s1)  % frequency of bases in simulated data
```

Run the m-file. If you were given the sequences `s0` and `s1` whose base composition is described by `F`, would your empirical estimates of the root base distribution and Markov matrix suggest a Jukes-Cantor model?

Now increase the sequence length `n` to 300, 600, and 900, repeating your work. How do your results differ as the sequence length is made longer?

Finally, for `n=300`, vary `a` and see its effect on the simulated data. In what range can `a` be chosen to be biologically meaningful? What happens when `a` is chosen at either extreme of this range?

4. In the lecture notes, the connection between the Jukes-Cantor distance and the Hamming distance was explored theoretically for 'perfect' data of infinite length sequences produced according to the Jukes-Cantor model. In this problem, you will explore the effects of finite sequence length, still assuming the data is produced according to the model.

At the end of you m-file from the last problem, add the following commands:

```
dtrue=-(3/4)*log(1-4*a/3) % true Jukes-Cantor distance
dsim=distJC(F)           % recovered Jukes-Cantor distance
error=dsim-dtrue         % error in recovered distance
error_ratio=error/dtrue  % scaled error
```

Run the m-file repeatedly to see how well we can typically recover a distance with the particular choice of `a=.1` and `n=100`.

Next, for fixed `a`, vary the sequence length `n` to see the effect on recovering the distance. Explain your observations.

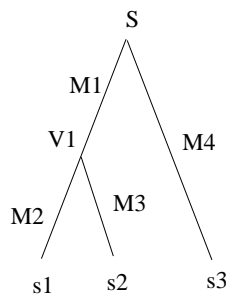
Finally, for fixed n , vary a to see the effect on recovering the distance. (Keep a in a range that is biologically meaningful.)

5. In this problem you will simulated sequence data along a phylogenetic tree, and then attempt to recover the tree from the data.

(a) Draw a rooted binary 5-taxon tree – choose your favorite among the $7!!$ possibilities. Then pick a root distribution vector and Markov matrices for each edge. You should begin simply, using a Jukes-Cantor model.

(b) Using the MATLAB programs `seqgen.m` and `mutatef.m` to produce an ancestral sequence of 300 bases and 'evolve' it according to your parameter choices. Create an `m`-file with all the necessary commands, so that it will be easy to generate other examples of simulated data.

Sample `m`-file for a 3-taxon tree with `s1` and `s2` as neighbors.



```

d=[.25,.25,.25,.25]
N=300
S=seqgen(d,N)
a1=.01
a2=.01
a3=.01
a4=.01
M1=markovJC(a1)
M2=markovJC(a2)
M3=markovJC(a3)
M4=markovJC(a4)
V1=mutatef(S,M1)
s1=mutatef(V1,M2)
s2=mutatef(V1,M3)
s3=mutatef(S,M4)
  
```

(c) Compute Hamming dissimilarities from the sequences with `hamming.m` and then use the program `nj.m` to apply the Neighbor Joining algorithm. Do you get your original tree back?

```

A=[s1;s2;s3;s4;s5]
H=hamming(A)
nj(H)
  
```

- (d) Repeat the last two steps several times, using the same parameter values. Are the results always the same?
- (e) If in your first few attempts you seemed to usually recover the correct tree, find parameter choices for which you often don't recover the correct one. If you usually didn't recover the correct one, find parameter choices for which you do.
- (f) Repeat the earlier steps using the Jukes-Cantor distance instead of the Hamming distance. Does this increase your recovered trees' similarity to the true tree.
A=[s1;s2;s3;s4;s5]
J=distJCarray(A)
nj(J)
- (g) Vary the sequence length in your simulated data. What affect does this have on tree recovery?
- (h) Change your parameters so that they are no longer in keeping with the Jukes-Cantor model. If you use the Jukes-Cantor distance and Neighbor Joining, do you still tend to do well a recovering the true tree? Investigate by doing repeated simulations and varying the parameters in interesting ways.