Phylogenomic Dating with MCMCTree

Český Krumlov Phylogenomics Workshop 2019

Mario dos Reis

@mariodosreis

Queen Mary University of London

Phylogenomic Dating Tutorial

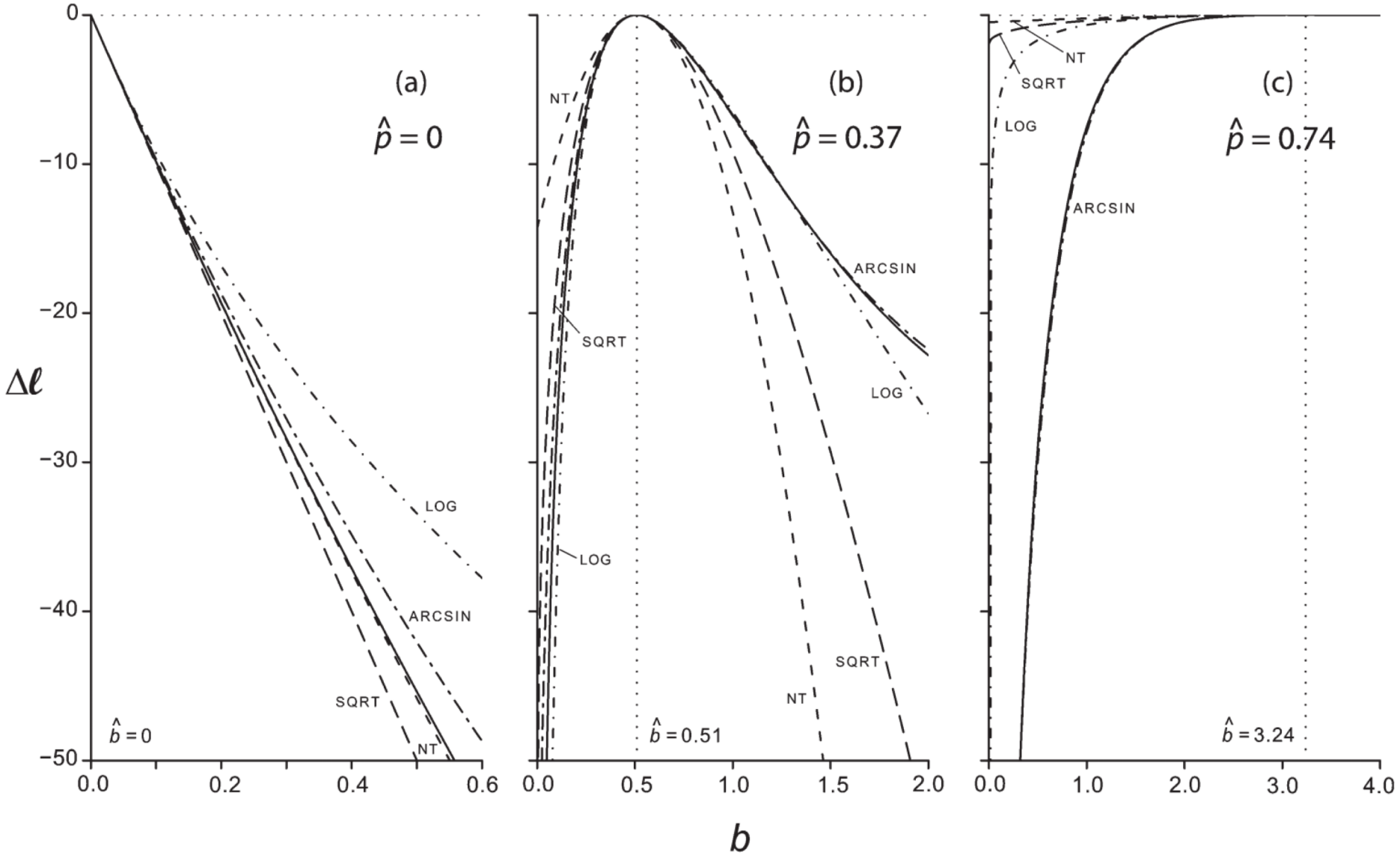
- Genome-scale alignments are normally too large for standard Bayesian MCMC sampling
- The likelihood of branch lengths on the tree can be approximated with a multivariate normal distribution
- This approximation, which speeds up computation substantially, is currently implemented in MCMCTree for molecular clock dating
- In this tutorial we will learn how to use this method

Phylogenomic Dating Tutorial

- This tutorial can be found in our forthcoming book chapter:
- dos Reis and Yang (2019) **Bayesian molecular clock dating using genome-scale datasets**
- In: Anisimova M (ed.) Evolutionary Genomics: Statistical and Computational Methods, Springer (in press)

- **http://bit.ly/phylodating**

Approximate Likelihood

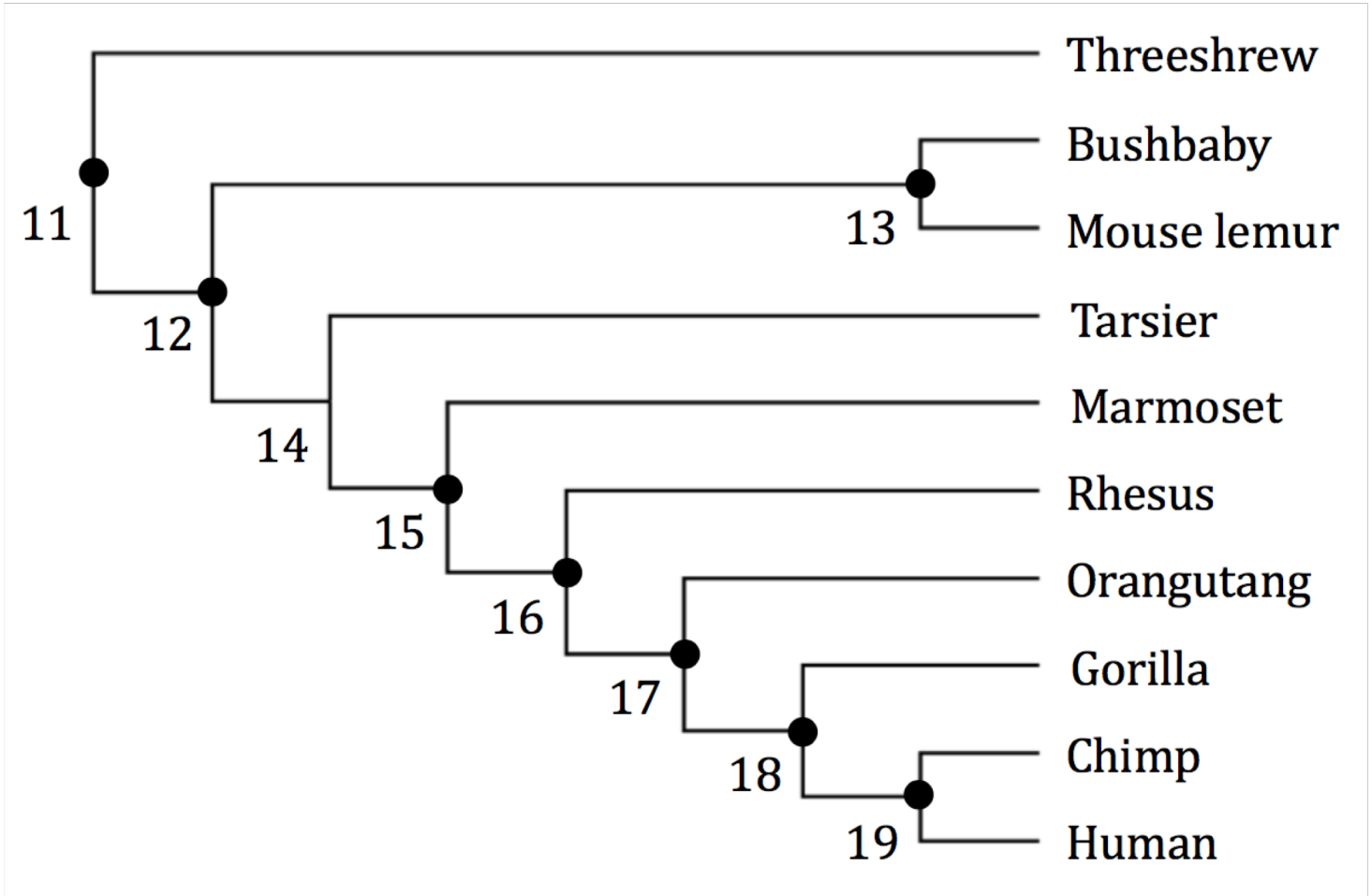


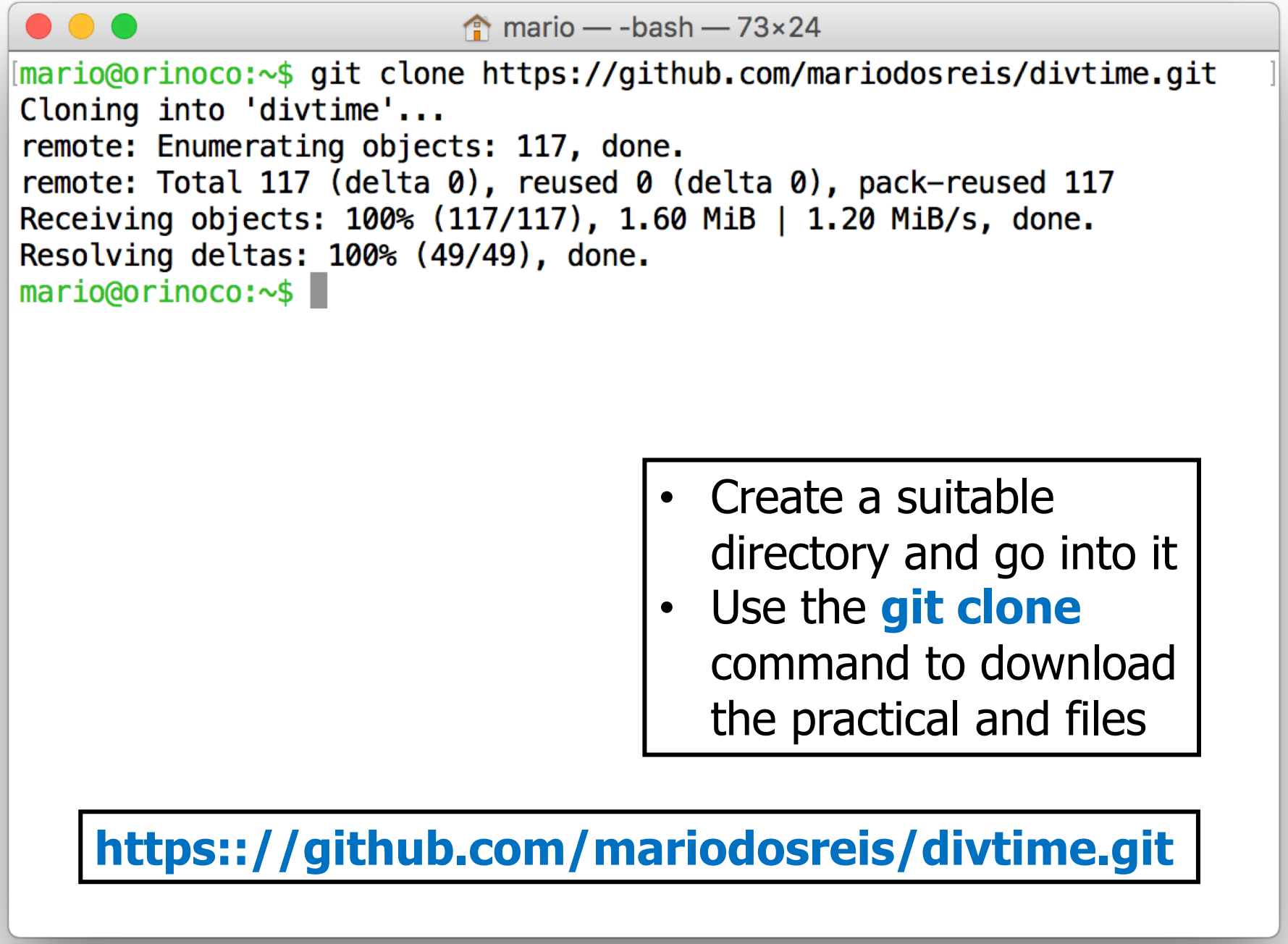
Approximate Likelihood

- The normal distribution is determined by two parameters:
  - The mean (the location of the mode)
  - The variance (the curvature of the mode)
- In the approximate likelihood:
  - The MLE of the branch lengths indicate the location of the likelihood mode = normal mean
  - The second derivative of the likelihood at the mode indicates the curvature → variances and covariances
- **Method:**
  - Use a fixed tree topology, then:
  - Estimate MLE vector branch lengths, $\hat{\mathbf{b}} = (\hat{b}_1, \ldots, \hat{b}_{2s-3})$
  - Calculate Hessian Matrix of 2nd derivatives, $-H^{-1} = \Sigma$
  - Σ is the covariance matrix
  - Run the MCMC using the MVN instead of the phylogenetic likelihood – much faster!

Tutorial

- We will estimate divergence times for 10 Primate genomes (about 3.36 million sites from > 5,000 genes) using MCMCTree





- Create a suitable directory and go into it
- Use the **git clone** command to download the practical and files

https://github.com/mariodosreis/divtime.git

3.1 Overview

We will use the approximate likelihood method to speed up the computation of the likelihood on the large genome alignment. The general strategy for the analysis is as follows:

1. *Approximate likelihood calcula*

   Hessian (**H**) matrix of the bran

   will need to use the MCMCTre

   the actual computation of **g** and **H**). The substitution model is chosen at this step.

2. *MCMC sampling from the posterior*: Once **g** and **H** have been calculated and we have decided on our priors, we can use MCMCTree to perform MCMC sampling from the posterior distribution of times and rates. We will then look at the

- We now follow from section 3.1 (page 8) of the tutorial PDF (with some modifications)

- To estimate MLEs of b, g, and H:
- Go into **gH/**
- Type:
- **../src/mcmctree mcmctree-outBV.ctl**

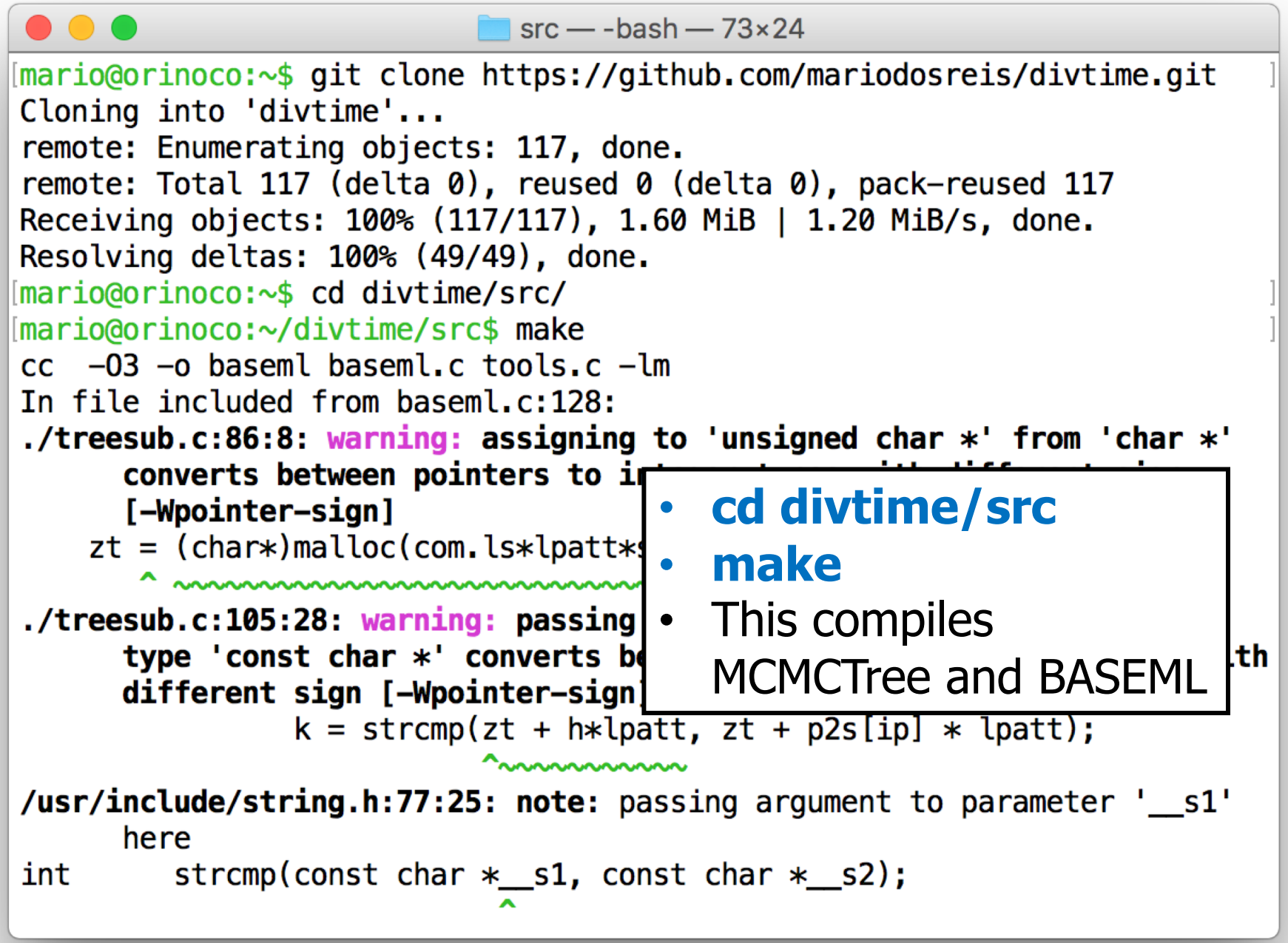
- Then type:
- **../src/baseml tmp0001.ctl**
- **cp rst2 out.BV1**

- And then:
- **../src/baseml tmp0002.ctl**
- **cp rst2 out.BV2**

- **cat out.BV1 out.BV2 >out.BV**

- out.BV has the MLEs, g and H

- Go into:
- **cd ../mcmc**
- **cp ../gH/out.BV in.BV**
- **../src/mcmctree**
- This now runs the MCMCTree analysis using the approximate likelihood method
- **cp mcmc.txt mcmc1.txt**
- **../src/mcmctree**
- **cp mcmc.txt mcmc2.txt**


src — -bash — 73×24
mario@orinoco:~$ git clone https://github.com/mariodosreis/divtime.git
Cloning into 'divtime'...
remote: Enumerating objects: 117, done.
remote: Total 117 (delta 0), reused 0 (delta 0), pack-reused 117
Receiving objects: 100% (117/117), 1.60 MiB | 1.20 MiB/s, done.
Resolving deltas: 100% (49/49), done.
mario@orinoco:~$ cd divtime/src/
mario@orinoco:~/divtime/src$ make
cc -O3 -o baseml baseml.c tools.c -lm
In file included from baseml.c:128:
./treesub.c:86:8: warning: assigning to 'unsigned char *' from 'char *'
converts between pointers to i
[-Wpointer-sign]
zt = (char*)malloc(com.ls*lpatt*
./treesub.c:105:28: warning: passing
type 'const char *' converts be
different sign [-Wpointer-sign]
k = strcmp(zt + h*lpatt, zt + p2s[ip] * lpatt);
/usr/include/string.h:77:25: note: passing argument to parameter '__s1'
here
int strcmp(const char *__s1, const char *__s2);
• cd divtime/src
• make
• This compiles MCMCTree and BASEML