

DateLife Workflows

Luna L. Sanchez Reyes

2019-04-23

Taxon Cetacea

I. Query source data

There are 198 species in the Open Tree of Life Taxonomy for the taxon Cetacea. Information on time of divergence is available for 89 of these species across 6 published and peer-reviewed chronograms. Original study citations as well as proportion of Cetacea species found across those source chronograms is shown in Table 1.

All source chronograms are fully ultrametric.

Table 1: Cetacea source chronogram studies information.

	<i>Citation</i>	<i>Source N</i>	<i>Taxon N</i>
1.	Bininda-Emonds, Olaf R. P., Marcel Cardillo, Kate E. Jones, Ross D. E. MacPhee, Robin M. D. Beck, Richard Grenyer, Samantha A. Price, Rutger A. Vos, John L. Gittleman, Andy Purvis. 2007. The delayed rise of present-day mammals. <i>Nature</i> 446 (7135): 507-512	3	78/198
2.	Hedges, S. Blair, Julie Marin, Michael Suleski, Madeline Paymer, Sudhir Kumar. 2015. Tree of life reveals clock-like speciation and diversification. <i>Molecular Biology and Evolution</i> 32 (4): 835-845	1	79/198
3.	Steeman, M., Hebsgaard M., Fordyce R., Ho S., Rabosky D., Nielsen R., Rahbek C., Glenner H., Sørensen M., & Willerslev E. 2009. Radiation of Extant Cetaceans Driven by Restructuring of the Oceans. <i>Systematic Biology</i> 58 (6): 573-585.	1	86/198
4.	Toljagić O., Voje K.L., Matschiner M., Liow L., & Hansen T.F. 2017. Millions of Years Behind: Slow Adaptation of Ruminants to Grasslands. <i>Systematic Biology</i> , .	1	32/198

Source N: Number of source chronograms reported in study.

Taxon N: Number of queried taxa found in source chronograms.

Source chronograms maximum age range from 33.5 to 55.5 million years ago (MYA). As a means for comparison, lineage through time plots of all source chronograms available in data base are shown in Fig. 1

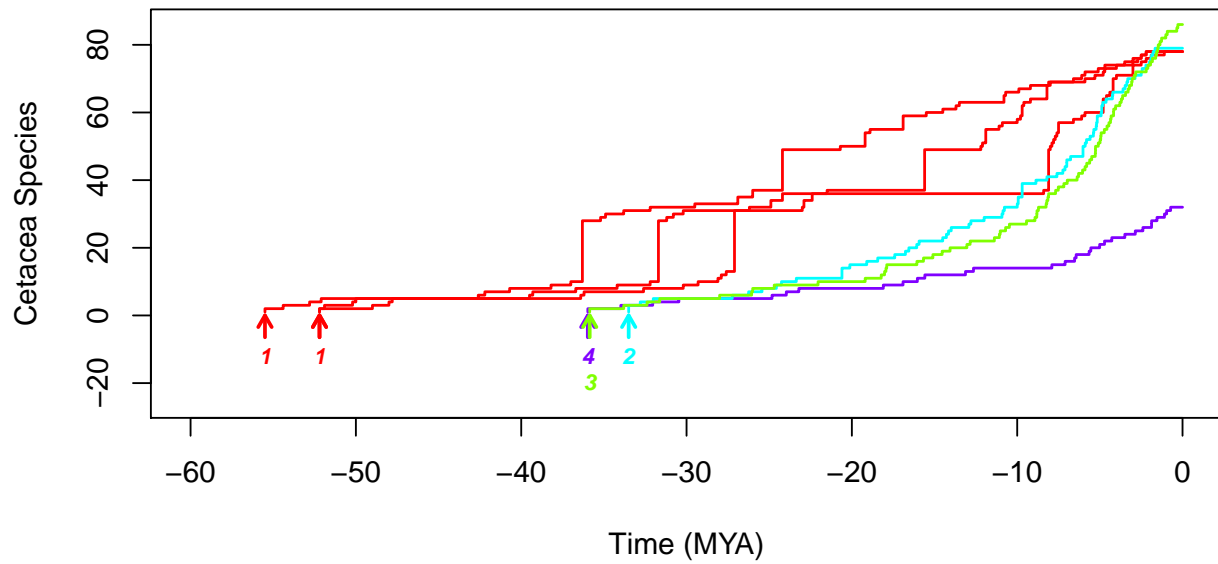


Figure 1: Lineage through time (LTT) plots of source chronograms available in data base for species in the Cetacea. Numbers correspond to original studies in Table 1. Arrows indicate maximum age of chronograms.

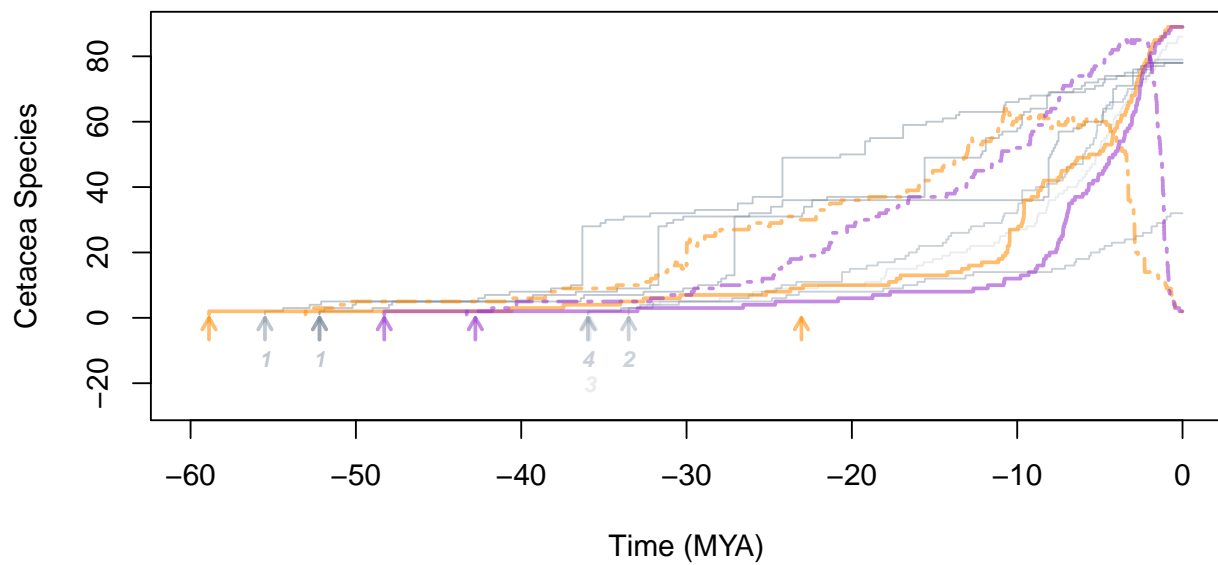


Figure 2: Test of make_lttplot_summ2 function

II. Summarize results.

LTT plots are a nice way to visually compare several trees. But what if you want to summarize all that information into a single chronogram?

The first step is to identify the degree of species overlap among your source chronograms: if each source chronogram has a unique sample of species, it will not be possible to combine them into a single summary chronogram. To identify the set of trees or *grove* with the most source chronograms that have at least two overlapping taxa, we followed Ané et al. 2016. In this case, not all source chronograms found for the Cetacea have at least two overlapping species. The largest grove has 2 chronograms (out of 6 total source chronograms). Now that we have identified a suitable grove we can go on to summarize it by translating the source chronograms into patristic distance matrices and then averaging them into a single summary matrix; yes, this first step is *that* straightforward. We can average the source matrices by simply using the mean or median distances, or we can use more complicated approaches that involve transforming the original distance matrices—such as the super distance matrix (SDM) approach of Criscuolo et al. 2006—by minimizing the distances across source matrices.

Once with a summary matrix, a distance-based clustering algorithm can be used to reconstruct the tree. Algorithms such as neighbour joining (NJ) and unweighted pair group method with arithmetic mean (UPGMA) are fast and work well when there are no missing values in the matrices. However, summary matrices coming from source chronograms usually have several NAs and missing rows. When this happens, even available variants of NJ and UPGMA algorithms that are designed to deal with missing data do not work well, as shown in the next section. Other methods designed to deal with missing data are BIONJ*, MVR*, and the triangle method, but we have not tried them yet.

II.A. Diagnosing clustering issues.

Clustering algorithms used to go from a summary distance matrix to a tree return trees that are too old (generally with UPGMA algorithms) or non-ultrametric (generally with NJ algorithms). In most studied cases, UPGMA returns fully ultrametric trees but with very old ages (we had to multiply the matrix by 0.25 to get ages approximate to source chronograms ages, however this number is not justified, it is just the number that approximates ages to source maximum ages the most). NJ returned reasonable ages, but trees are way non ultrametric, as you can see in Fig. S1 and Fig. 2.

This taxon’s SDM matrix has some negative values in the following taxa: *Eubalaena japonica*, *Eubalaena glacialis*. This taxon’s Median matrix has NO negative values.

II.B. Age distributions from Median and SDM summary trees.

Comparison of summary chronograms reconstructed with min and max ages.

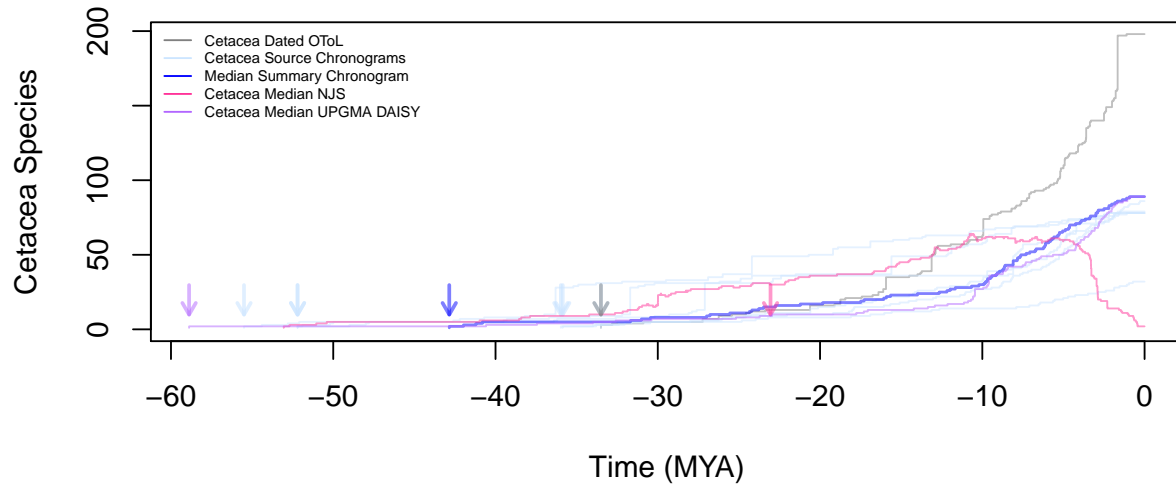


Figure 3: Cetacea lineage through time (LTT) plots from source chronograms and Median summary matrix converted to phylo with different methods (NJ and UPGMA). Clustering algorithms used often are returning non-ultrametric trees or with maximum ages that are just off (too old or too young). So we developed an alternative algorithm in **datelife** to go from a summary matrix to a fully ultrametric tree.

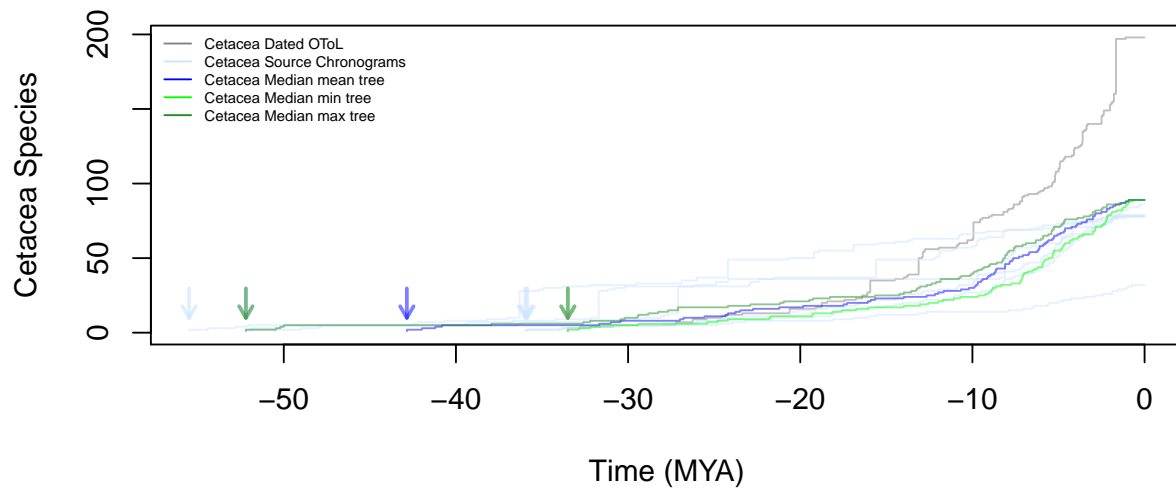


Figure 4: Cetacea lineage through time (LTT) plots from source chronograms and Median summary matrix converted to phylo with **datelife** algorithm.

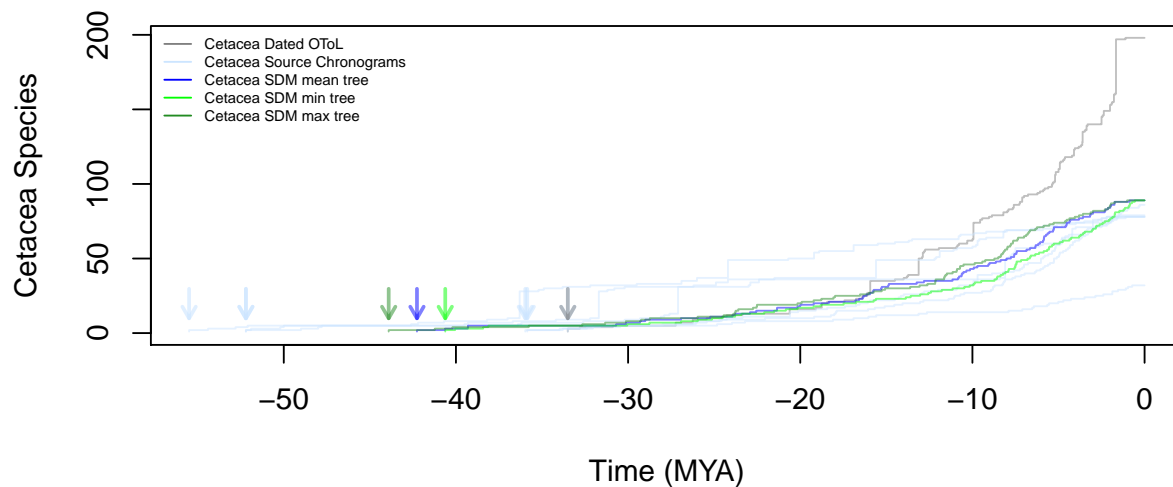


Figure 5: Cetacea lineage through time (LTT) plots from source chronograms and SDM summary matrix converted to phylo with `datelife` algorithm.

III. Create new data

As an example, we're gonna date the Open Tree Synthetic tree (mainly because the taxonomic tree is usually less well resolved.)

Now, let's say you like the Open Tree of Life Taxonomy and you want to stick to that tree. Dates from available studies were tested over the Open Tree of Life Synthetic tree of Cetacea and a tree was constructed, but all branch lengths are NA. We also tried each source chronogram independently, with the Dated OTOL and with each other, as a form of cross validation in Table 2. This is not working perfectly yet, but we are developping new ways to use all calibrations efficiently.

Table 2: Was it successful to use each source chronogram independently as calibration (CalibN) against the Dated Open Tree of Life (dOToL) and each other (ChronoN)?

	dOToL	Chrono1	Chrono2	Chrono3	Chrono4	Chrono5	Chrono6
Calibrations1	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
Calibrations2	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
Calibrations3	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
Calibrations4	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
Calibrations5	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
Calibrations6	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE

III. Simulate data

An alternative to generate a dated tree from a set of taxa is to take the available information and simulate into it the missing data. We will take the median and sdm summary chronograms to date the Synthetic tree of Life:

```
#> Error in paste0("\n![" , figcap_lttplot_sdm, "](plots/", taxon, "_LTTplot_sdm.pdf)\n"): object 'figcap'
#> Error in cat(lttplot): object 'lttplot' not found
```

Appendix

The following species were completely absent from the chronogram data base: *Amphiptera pacifica*, *Balaena agamachschik*, *Balaena mangidach*, *Balaenoptera andrejewi*, *Balaenoptera caerulea*, *Balaenoptera emarginata*, *Balaenoptera grimmii*, *Balaenoptera maculata*, *Balaenoptera nigra*, *Balaenoptera punctulata*, *Catodon polycyphus*, *Catodon polyscyphus*, *Catodon vineval*, *Cephalorhynchus commersonii*, *Cephalorhynchus heavisidii*, *Clymenia gadamu*, *Delphinapterus senedetta*, *Delphinorhynchus maculatus*, *Delphinorhynchus pernettyi*, *Delphinorhynchus santonicus*, *Delphinus abusalam*, *Delphinus anarnacus*, *Delphinus attenuatus*, *Delphinus bertini*, *Delphinus bonnaterrei*, *Delphinus boryi*, *Delphinus caeruleus*, *Delphinus carbonarius*, *Delphinus coronatus*, *Delphinus cymodice*, *Delphinus cymodoce*, *Delphinus epidon*, *Delphinus eurynome*, *Delphinus fabricii*, *Delphinus feres*, *Delphinus gadamu*, *Delphinus hamatus*, *Delphinus harlani*, *Delphinus leucocephalus*, *Delphinus livittatus*, *Delphinus maculatus*, *Delphinus maculiventer*, *Delphinus minimus*, *Delphinus nesarnac*, *Delphinus niger*, *Delphinus pernettyensis*, *Delphinus pernettyi*, *Delphinus perniger*, *Delphinus rappii*, *Delphinus rhinoceros*, *Delphinus salam*, *Delphinus siculus*, *Delphinus symodice*, *Delphinus walkeri*, *Epidon rafinesque*, *Epidon urganantus*, *Eudelphinus tasmaniensis*, *Globicephala macrorhynchus*, *Globicephalus fuscus*, *Globicephalus unedens*, *Globiocephalus chinensis*, *Inia araguaiaensis*, *Inia boliviensis*, *Lagenodelphis australis*, *Lagenodelphis obliquidens*, *Lagenodelphis hosei*, *Lagenorhynchus bombifrons*, *Lagenorhynchus nilssonii*, *Lagenorhynchus posidonia*, *Lagenorhynchus superciliosus*, *Lagenorhynchus acutus*, *Lagenorhynchus albirostris*, *Lagenorhynchus australis*, *Lagenorhynchus cruciger*, *Lagenorhynchus obliquidens*, *Lagenorhynchus obscurus*, *Mesoplodon hotaula*, *Mesoplodon lazaardii*, *Monodon spurius*, *Neophocaena asiaorientalis*, *Phocaena posidonia*, *Physeter gibbosus*, *Physeter katadon*, *Physeter krefftii*, *Physeter polycephalus*, *Physeter polycystus*, *Physeter pterodon*, *Platanista indi*, *Prodelphinus malayanus*, *Sotalia gadamu*, *Sotalia maculiventer*, *Sotalia perniger*, *Sotalia santonicus*, *Sousa gadamu*, *Sousa plumbea*, *Sousa sahalensis*, *Steno fuscus*, *Steno gadamu*, *Steno malayanus*, *Steno perniger*, *Tursio catalania*, *Tursio cymodoce*, *Tursio eurynome*, *Tursiops australis*, *Tursiops catalania*, *Tursiops cymodice*, *Tursiops dawsoni*, *Tursiops fergusonii*, *Tursiops nesarnack*

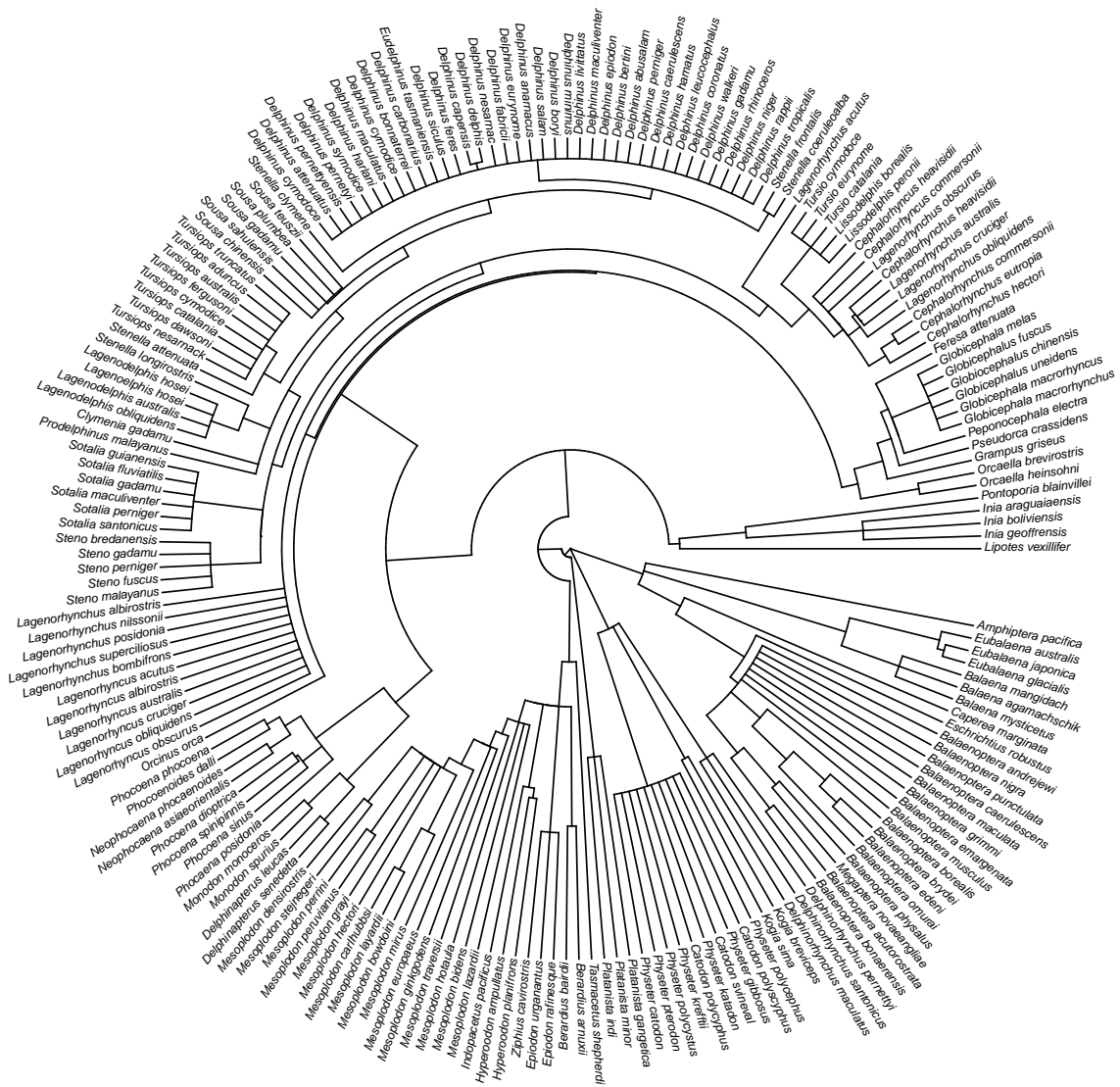


Figure 6: Cetacea Species Dated Open Tree of Life Induced Subtree. This chronogram was obtained with `get_dated_otol_induced_subtree()` function.

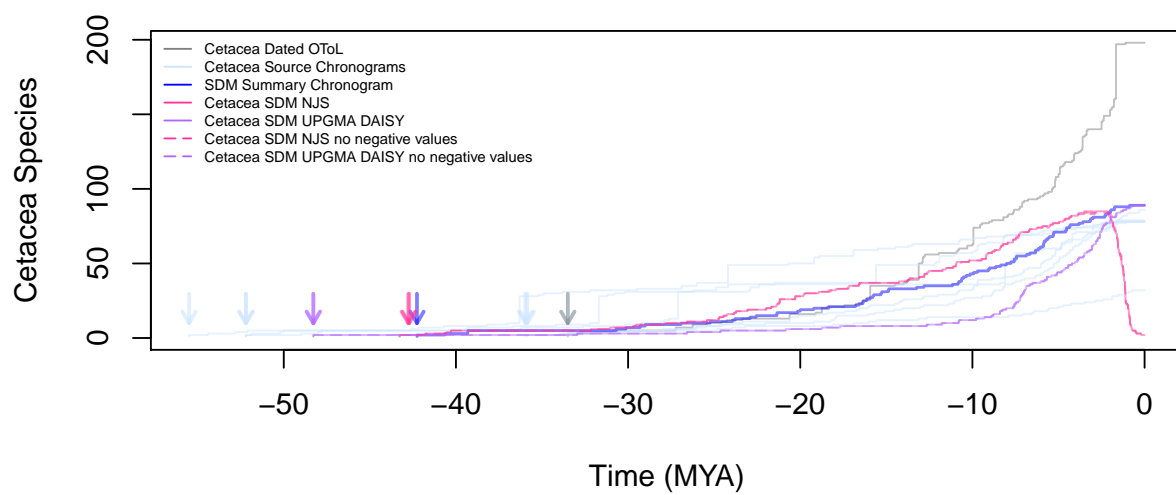


Figure 7: Cetacea lineage through time (LTT) plots from source chronograms and SDM summary matrix converted to phylo with **datelife** algorithm.