

Extended methods

Extra methodological details that were not relevant to report in the main text are listed here.

Code

Diet and Teat number

In the code there is a small section dedicated to re-coding diet and teat number to make them categorical. This was done to make the `fitdiscrete()` function happy without changing our definitions and re-entering all the data for a particular trait.

For example, the trait “diet” had several categories in the listed definition and dataset. The definition is: What kind of food the animal eats. Species were defined as (1) frugivores, (2) granivores, (3) nectivores, (4) folivores, (5) lignivores or (6) multiple kinds.

For some reason, probably due to a lack of data, we only had data for ungulates that fell into categories 1,4, and 6. The `fitdiscrete()` function was not happy with this and would not accept a gap between 1 and 4 and 4 and 6. The solution in this case was to make categories for each value so that there was an ascending numerical list with no gaps.

To put it simply, ungulates in category 1 were recoded with the value 1

Ungulates in category 4 were re-coded with the value 2

Ungulates in category 6 were re-coded with the value 3

If more data is to be added in the future, either a better line of code must be written that will automatically update given new data, or the categories must be re-coded.

The same logic applies for teat number. Our dataset included ungulates with 2,3,4,6,8,11, and 12 teats. Since the gaps were not accepted by `fitdiscrete()`,

2 was re-coded to 1

3 was re-coded to 2

4 was re-coded to 3

6 was re-coded to 4

8 was re-coded to 5

11 was re-coded to 6

12 was re-coded to 7

Head Ornaments

There is a line of R code which changes all 0 values into 2. This change was in essence a reversion from a previous iteration of code for another project. The definition is as follows: Whether there is a sexual dimorphism in head ornaments. The species were defined as (1) there is sexual dimorphism in head ornaments and (2) there is no sexual dimorphism in head ornaments.

Essentially, a previous iteration of code had re-coded the 2 values into 0 values, to make the data for this trait binary (0 or 1) because the code could read it better. However, `fitdiscrete()` did not like 0 and 1 values, so we reverted it to 1 and 2.

Out of order code

The code written to acquire lambda values for the traits is out of order compared to what is shown in the figure and dataset. This was simply because we did not know which traits would show high or low phylogenetic signal and were simply run in order of appearance in the dataset. After all the lambda values were collected and recorded in excel, they were manually put in order from high to low to make the figure aesthetically pleasing.