Non-equilibrium evolution of volatility in origination and extinction explains fat-tailed fluctuations in Phanerozoic biodiversity

Andrew J. Rominger[1, *], Miguel A. Fuentes[1, 2, 3], and Pablo A. Marquet[1, 4, 5, 6, 7]

[1]Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, US
[2]Instituto de Investigaciones Filosóficas, SADAF, CONICET, Bulnes 642, 1428 Buenos Aires, Argentin
[3]Facultad de Ingeniería y Tecnología, Universidad San Sebastián, Lota 2465, Santiago 7510157, Chile
[4]Departamento de Ecología, Facultad de Ciencias Biológicas, Pontificia Universidad de Chile, Alameda 340, Santiago, Chile
[5]Instituto de Ecología y Biodiversidad (IEB), Casilla 653, Santiago, Chile
[6]Laboratorio Internacional de Cambio Global (LINCGlobal), and Centro de Cambio Global UC, Pontificia Universidad Catolica de Chile, Santiago, Chile.
[7]Centro Cambio Global UC, Av. Vicuña Mackenna 4860, Campus San Vicuña, Santiago, Chile
[8]Centro de Ciencias de la Complejidad (C3), Universidad Nacional Autónoma de México.
[*]To whom correspondence should be addressed, e-mail: rominger@santafe.edu

Fluctuations in biodiversity, both large and small, are pervasive through the fossil record, yet we do not understand the processes generating them. Here we extend theory from non-equilibrium statistical physics to describe the previously unaccounted for fat-tailed form of fluctuations in marine invertebrate richness through the Phanerozoic. Using this theory, known as superstatistics, we show that the simple fact of heterogeneous rates of origination and extinction between clades and conserved rates within clades is sufficient to account for this fat-tailed form. We identify orders and the families they subsume as the taxonomic level at which clades experience inter-clade heterogeneity and within clade homogeneity of rates. Following superstatistics we would thus posit that orders and families are subsystems in local statistical equilibrium while the entire system is not in equilibrium. The separation of timescales between background origination and extinction within clades compared to the origin of major ecological and evolutionary innovations leading to new orders and families allows within-clade dynamics to reach equilibrium, while between-clade diversification is non-equilibrial. This between clade non-equilibrium accounts for the fat-tailed nature of the system as a whole. The distribution of shifts in diversification dynamics across orders and families is consistent with niche conservatism and pulsed exploration of adaptive landscapes by higher taxa. Compared to other approaches that have used simple birth-death processes, simple equilibrial dynamics, or non-linear theories from complexity science, superstatistics is superior in its ability to account for both small and extreme fluctuations in the richness of fossil taxa. Its success opens up new research directions to better understand the evolutionary processes leading to the stasis of order- and family-level occupancy in an adaptive landscape interrupted by innovations that lead to novel forms.

1 Introduction

Biodiversity has not remained constant nor followed a simple trajectory through geologic time (*1–5*). Instead, it has been marked by fluctuations in the richness of taxa, both positive in the case of net origination, or negative in the case of net extinction. Major events, such as adaptive radiations and mass extinctions have received special attention (*6, 7*), but fluctuations of all sizes are ubiquitous (*2, 5*) and follow a fat-tailed distribution where large events are more probable compared to, e.g. a Gaussian distribution. Understanding the fat-tailed nature of these fluctuations continues to elude paleobiologists and biodiversity theoreticians.

The fat-tailed distribution of fluctuations in taxon richness inspired earlier researchers to invoke ideas from complex systems with similar distributions. Such ideas include the hypotheses that biological systems self-organize to the brink of critical phase-transitions (*8, 9*) and that environmental perturbations are highly non-linear (*10*). New data and analyses have not, however, supported these hypotheses at the scale of the entire Phanerozoic marine invertebrate fauna (*5, 11*). Other studies have modeled the mean trend in taxon richness as tracking a potentially evolving equilibrium (*2, 12, 13*) and yet ignore the role of stochasticity and non-equilibrium dynamics in producing observed patterns (*4, 14, 15*). Individual, population, and local ecosystem scale processes that could produce complex dynamics, such as escalatory co-evolutionary interactions (*16*), have not been documented to scale up to global patterns (*17*) and indeed should not be expected to do so (*18*). Thus, we still lack a theory to describe the striking fat-tailed nature of fluctuations throughout the Phanerozoic.

Despite the heterogeneity of explanations of Phanerozoic biodiversity, consensus has emerged on one property of macroevolution: clades experience different rates of morpho-

logical evolution, origination and extinction (*2, 3, 19, 20*). Here we show that the simple fact of conserved rates within clades and variable rates across clades is sufficient to describe pervasive, fat-tailed fluctuations in taxonomic richness throughout the marine Phanerozoic. This biological mechanism has a precise correspondence to the non-equilibrial theory from statistical physics known as "superstatistics" (*21*) which has been applied across the physical and social sciences (*22, 23*). We leverage this correspondence to explain the distribution of fluctuations in the standing richness of marine invertebrates preserved in the Phanerozoic fossil record. We further show that the specific mathematical form of this superstatistical distribution is consistent with niche conservatism (*24, 25*) and pulsed exploration on an adaptive landscape by higher taxa (*19, 25–27*). We operationally define "adaptive landscape" to mean a clade's set characteristics that influences its macroevolution. Those characteristics could be ecological (e.g. substrate preference (*25, 28, 29*)), morphological (e.g. body plan (*14*)), or macroecological (e.g. range size (*30, 31*)).

1.1 Superstatistics of fossil biodiversity

Superstatistics (*21*) proposes that non-equilibrial systems can be decomposed into many local sub-systems, each of which attains a unique dynamic equilibrium. The evolution of these dynamic equilibria across sub-systems occurs more slowly. This separation in time scales allows local systems to reach equilibrium while the system as a whole is not (*21*). In the context of macroevolution we propose that a clade with conserved macroevolutionary rates corresponds to a sub-system in dynamic equilibrial.

In statistical mechanics, local sub-systems can be defined by a simple statistical parameter β often corresponding to inverse temperature. In macroevolutionary "mechanics" we define the β_k of clade k as the inverse variance of fluctuations x_k in the number of genera within that clade, i.e. fluctuations in the genus richness. The β_k thus represent the in-

verse variances, what we term volatilities, of the stationary distribution of a homogeneous origination-extinction processes of genera. Fluctuations from this stationary process will be approximately Gaussian if the clades' diversification dynamics are independent and in local equilibrium (see Supplemental Section S1; (*32*)).

We make the hypothesis of dynamic equilibrium within a clade following MacArthur and Wilson (*33*) in recognition that while the identity and exact number of taxa will fluctuate stochastically from random origination and extinction (taking the place of local immigration and extinction in island biogeography), the overall process determining the number of taxa, and by extension, fluctuations in that number, is in equilibrium. Indeed, the different regions of adaptive space occupied by different clades can be conceptualized as islands with unique dynamic equilibria, albeit with macroevolutionary processes determining the "colonization" of adaptive peaks, as opposed to short timescale biogeographic processes.

The volatility of richness fluctuations will vary across these islands in adaptive space as an emergent trait of a clade. Ultimately, volatility emerges from the life histories, ecologies, and evolutionary histories that characterize each clade's occupancy in different regions of an adaptive landscape. We do not attempt to diagnose which characteristics of different regions account for volatility differences, but others have found rates of origination and extinction to depend on larval type (*34*), body plan (*14*), body size (*30*), range size (*30, 31*), and substrate preference (*25*). Not all of these traits would be considered dimensions of an ecological niche or characteristics of a guild (*28, 29, 35*), but they all point to different strategies that influence a clade's macroevolutionary success. These characteristics result from interactions between heritable traits and environments, which themselves may be semi-heritable (*36*). Thus different regions of adaptive space, and the clades occupying them, will experience different magnitudes of stochastic fluctuations in

taxonomic richness. As clades occasionally split to fill new regions of adaptive space their pulsed diversification determines the non-equilibrium nature of the entire biota.

1.2 Real paleontological data to test superstatistics

To uncover the superstatistical nature of the marine invertebrate Phanerozoic fauna we analyzed the distribution of fluctuations in genus richness (the lowest reliably recorded taxonomic resolution) using the Paleobiology Database (PBDB; `paleobiodb.org`). We corrected these raw data for incomplete sampling and bias using a new approach described in the methods section. Occurrences from the PBDB were matched to 49 standard time bins all of approximately 11MY duration following previous publications (*5, 12*). Fluctuations in genus richness were calculated as the simple difference between bias-corrected richnesses in adjacent time bins.

To focus attention on the variance of fluctuations we zero-centered each clade's fluctuation distribution. In this way we focus on fluctuations about any possible trend toward net diversification or extinction. Because "equilibrium" in the statistical mechanical sense means a system undergoes coherent, concerted responses to perturbation, the mean trend line (positive or negative) is of less interest than deviations from it. We also note that the distributions of fluctuations for most clades are already very close to a mean of 0 (mean at the family level: 0.038 ± 0.176 SD), and so centering has little influence on clade-specific fluctuation distributions, consistent with the observation that origination is often roughly equal to extinction (*37*).

We define potentially equilibrial sub-systems based on taxonomic hierarchies as a full phylogenetic hypothesis for all marine invertebrates is lacking. Taxa ideally represent groups of organisms that descend from a common ancestor and share similar ecologically and evolutionary relevant traits (*38, 39*). Thus our model assumes that at a given higher

taxonomic level, within-taxon fluctuations in richness are driven by equilibrial processes characterized by Gaussian distributions. We further assume that new higher taxa arise due to the emergence of sufficiently novel traits (be they ecological, morphological, life history, or macroecological) so that those new taxa occupy a new region of an adaptive landscape. We lastly assume that different regions of adaptive space are characterized by different volatilities in origination and extinction.

To evaluate the optimal taxonomic level for sub-system designation, we test our superstatistical theory using taxonomic levels from family to phylum. Additionally, we compare our results to randomized taxonomies and confirm that the observed fit of superstatistics is not an artifact of arbitrary classification but instead represents real, biologically relevant diversification processes within and between clades. We find that families and orders conform to the assumptions of our superstatistical model while classes and phyla do not.

2 Results

We first evaluate the local equilibria of clades from family level to phylum. We find that family level fluctuation distributions are well approximated by Gaussians (Figs. 1 and S3). Three exemplar family-level dynamics are highlighted in Figure 1 to illustrate how different volatility equilibria express themselves as actual richness timeseries. This Gaussian approximation also largely holds for orders, but classes and phyla increasingly show deviations from Gaussian with greater kurtosis corresponding to more frequent outliers at these taxonomic levels (Fig. S3).

To predict the superstatistical behavior of the entire marine invertebrate Phanerozoic fauna we must integrate over all possible local equilibria that each clade could experience. The stationary distribution of β_k values describes these possible equilibria, specifying the probability that a given clade, chosen at random, will occupy a region of adaptive space

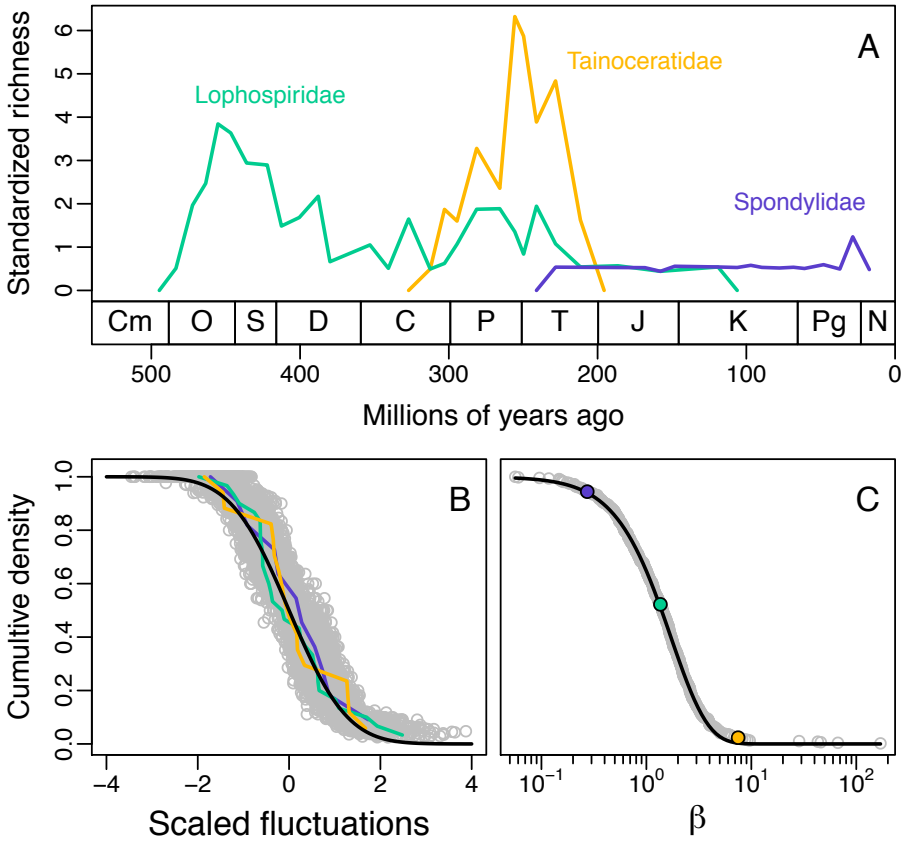

Figure 1: The distributions of within-family fluctuations in genus richness shown for the trajectories of three exemplar families (A) and shown as an empirical cumulative density functions aggregated across all families (B). To display all families simultaneously we simply collapse their fluctuation distributions by dividing by their standard deviations. If families conform to the Gaussian hypothesis their scaled fluctuations should fall along the cumulative density line of a normal N(0, 1) distribution, as shown in (B). We further confirm this normal distribution in the supplement (Fig. S3). In (C) the distribution of inverse variances β_k across all families matches very closely to a Gamma distribution (black line); exemplar families are again highlighted.

characterized by β_k.

We estimate the distribution of β_k's simply as the maximum likelihood distribution describing the set of volatilities for all families, orders, classes, or phyla. Phanerozoic marine invertebrate families clearly follow a Gamma distribution in their β_k values (Fig. 1). The Gamma distribution also holds for orders but shows increasing deviations again

for classes and especially phyla (Fig. S4).

Using the observation of within family statistical equilibrium and Gamma-distributed β_k parameters we can calculate, without further adjusting free parameters, the distributions of family-level fluctuations for the entire marine Phanerozoic, $P(x)$, as

$$P(x) = \int_0^\infty p_k(x \mid \beta) f(\beta) d\beta \tag{1}$$

where $p_k(x \mid \beta) = \sqrt{\frac{\beta}{2\pi}} e^{-\frac{\beta x^2}{2}}$ is the distribution of fluctuations within a family and $f(\beta) = \frac{1}{\Gamma(b_1/2)} \left(\frac{b_1}{2b_0}\right)^{b_1/2} \beta^{(b_1/2)-1} \exp\left(-\frac{b_1\beta}{2b_0}\right)$ is the stationary distribution of volatilities in richness fluctuations. The integral in (1) leads to

$$P(x) = \frac{\Gamma\left(\frac{b_1+1}{2}\right)}{\Gamma\left(\frac{b_1}{2}\right)} \sqrt{\frac{b_0}{\pi b_1}} \left(1 + \frac{b_0 x^2}{b_1}\right)^{-\frac{b_1+1}{2}} \tag{2}$$

This corresponds to a non-Gaussian, fat-tailed prediction for $P(x)$ which closely matches aggregated family level fluctuations in the bias-corrected PBDB (Fig. 2).

To quantitatively evaluate how well the superstatistical prediction matches the family-level data we constructed a 95% confidence envelope from bootstrapped maximum likelihood estimates of $P(x)$. Observed fluctuations fall within this 95% confidence envelope (Fig. 2), indicating that the data do not reject the superstatistical prediction. For further comparison, we fit a Gaussian distribution to the observed fluctuations, which corresponds to the equilibrium hypothesis that all families conform to the same dynamic. Using Akaike Information Criterion (AIC) we find that observed fluctuations are considerably better explained by the superstatistical prediction than by the Gaussian hypothesis (ΔAIC = 1895.622). Thus, as expected under the superstatistical hypothesis, the fat-tailed distribution of fluctuations arise from the superposition of independent Gaussian statistics of fluctuations within families. Computing the distribution of aggregated fluctuations using orders also closely matches the observed data (Fig. 2) but as we further coarsen the

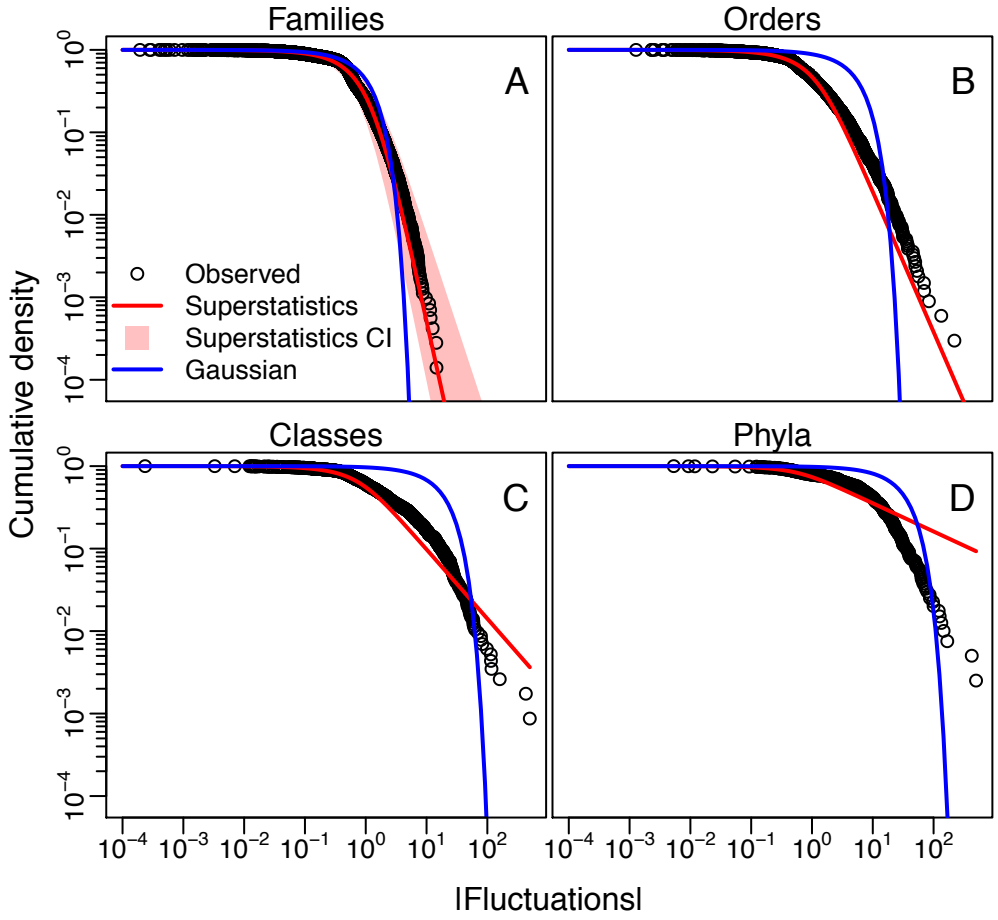

Figure 2: Distribution of fluctuations in genus richness within different taxonomic groupings of marine invertebrates in the Paleobiology Database (*5*) after sampling correction. The distribution is fat-tailed as compared to the maximum likelihood estimate of the normal distribution (blue line). At the family and order level the empirical distribution of fluctuations are well described by our superstatistical approach, both when computed from integrating over the distribution of observed variances (red line) and when fit via maximum likelihood (95% confidence interval; red shading in (A)).

taxonomy to classes and phyla we see increasingly poorer correspondence between data and theory (Fig. 2).

We quantify this change in the goodness of fit with the Kolmogorov-Smirnov statistic (Fig. 3). We can see that both families and orders have low Kolmogorov-Smirnov

statistics, and in fact order level designation of equilibrial subsystems performs slightly better than the family level. Classes are substantially worse and phyla worse yet with the Kolmogorov-Smirnov statistic of phyla being no different than the null randomized taxonomies described below.

However, if superstatistical theory explains the data, this worsening fit with increasing taxonomic scale is expected as the different classes and phyla should not represent dynamically equilibrial sub-systems in their fluctuation dynamics. Instead, classes and phyla aggregate increasingly disparate groups of organisms, and thus effectively mix their associated Gaussian fluctuations, meaning that one statistic should no longer be sufficient to describe class- and phylum-level dynamics. We see this confirmed by the increasing frequency of outlier fluctuations in within class and phylum level fluctuation distributions (Fig. S3). We can also see that families and orders represent, on average, 1 to 2 ecospace hypercubes (defined by taxon environment, motility, life habit, vision, diet, reproduction, and ontogeny (*28, 29, 35*)), respectively. In contrast, classes and phyla represent, on average, 8 to 30 hypercubes, respectively (Fig. S5).

Our analysis indicates that orders and families are evolutionarily coherent units with all subsumed taxa sharing key ecological and evolutionary attributes allowing them to reach steady state diversification independently from other clades at global scale. The fact that both orders and families conform to theoretical predictions is consistent with superstatistics. If superstatistics operates at the order level, then the families subsumed by these orders should represent random realizations of their order's stationary $\beta_k^{(order)}$ volatility. The sum of Gamma random variables is still Gamma, but with new parameters, thus the family level distribution of $\beta_k^{(family)}$ is still Gamma.

To further test the evolutionary coherence of families we conducted a permutation experiment in which genera were randomly reassigned to families while maintaining the

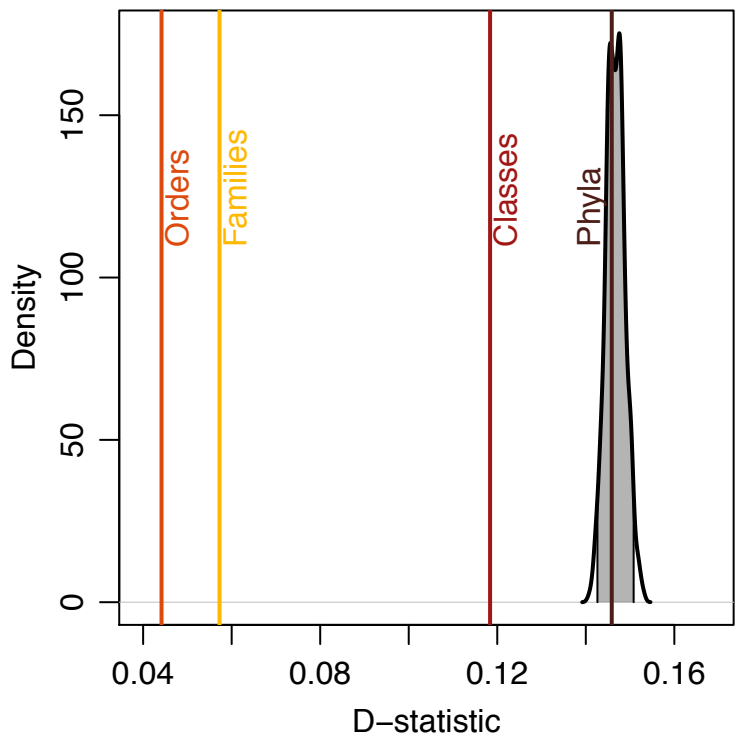

Figure 3: Distribution of Kolmogorov-Smirnov (KS) statistics from randomly permuting genera within familes (gray shading represents 95% confidence interval). Solid colored lines are observed KS statistics at different taxonomic levels as indicated.

number of genera in each family. For each permutation, we calculated the superstatistical prediction and its Kolmogorov-Smirnov statistic. The permutation simulates a null model in which common evolutionary history is stripped away (genera are placed in random families) but the total number of observed genera per family is held constant. Controlling for the total number of genera per family is key because this could be purely an artifact of an arbitrary taxonomic process (*40, 41*) and genus richness alone could be solely responsible for differences in the β_k across clades. Indeed, the number of genera in a family and that family's β_k value are correlated (Fig. S6). Thus we want to know if this correlation alone accounts for all downstream superstatistical results.

Repeating the null permutation of genera in families 500 times yields a null distribution

of Kolmogorov-Smirnov statistics that is far separated from the observed values at the family and order levels (Fig. 3) suggesting that the good fit at these levels is not merely a statistical artifact of classification or the richness of clades, but carries important biological information. Classes approach the null and phyla are no different. It should also be noted that the width of 95% confidence interval of this null distribution is not far from the distance between the Kolmogorov-Smirnov statistics of orders versus families, suggesting that differences of fit between these taxonomic levels is at least partially accounted for by the randomness of the sampling distribution of Kolmogorov-Smirnov statistics.

3 Discussion

Our analysis makes no assumption that orders and families should correspond to superstatistical subsystems, but identifies them as the appropriate level for marine invertebrates. Our study is the first to demonstrate that complex patterns in the fluctuation of taxon richness in the fossil record are the result of a simple underlying process analogous to the statistical mechanisms by which complexity emerges in large, non-equilibrium physical (*22*) and social systems (*23*). We do so by identifying the biological scale at which clades conform to locally independent dynamic equilibria in fluctuations. Equilibrium could result from many processes, including neutrality (*33, 42*), diversity-dependence (*43,44*) and processes that dampen—rather than exacerbate—fluctuations in complex ecological networks (*45*). These candidate processes are directly opposed to the presumption of instability underlying the self-organized criticality hypothesis for paleo biodiversity (*8, 9*).

We show that the distribution describing the evolution to different equilibria between orders and families is Gamma (Fig. 1). A Gamma distribution, while consistent with multiple processes, could result from evolution of diversification rates across an adaptive

landscape that promotes niche conservatism and pulsed exploration of niche space (*46*). Specifically, if β_k values are associated with a clade's macroevolutionarily-relevant traits, and those traits evolve via Ornstein-Uhlenbeck-like exploration of an adaptive landscape, the resulting stationary distribution of β_k will be Gamma (*46*). For macroevolutionary rates to vary across an adaptive landscape, this landscape cannot be flat, and thus niche conservatism around local optima in adaptive space interrupted by adaptive exploration is likely (*27, 47*). The specifics of how this adaptive landscape is shaped and is traversed by evolving clades determine the exact form of the distribution of β_k volatilities, in the case of the marine Phanerozoic resulting in a Gamma distribution. Our work thus motivates further study of the trait spaces and evolutionary shifts consistent with Gamma-distributed equilibria in richness fluctuation volatilities.

We show that the pulsed shift to different equilibria between orders and the families they subsume is sufficient to explain the characteristically fat-tailed distribution of richness fluctuations when the marine Phanerozoic invertebrate fauna is viewed as a whole macrosystem. Armed with an understanding of the statistical origin of this diversification pattern we can explore which models of niche conservatism and pulsed adaptive radiation are consistent with the statistical behavior of the Phanerozoic. Our statistical theory provides new motivation for identifying the eco-evolutionary causes of innovations between lineages and how those innovations are eventually conserved within lineages. Using the superstatistical prediction as a theoretical baseline, we can also go on to identify and robustly examine the mechanisms underlying deviations from statistical theory. For example, some clades wax and wane systematically, and possibly non-symmetrically, through time (*4, 13, 31*), a pattern that we cannot explain with superstatistics alone.

Superstatistics could also be applied to other areas of evolution and macroecology. For example new phylogenetic models already consider heterogeneous rates of diversifica-

tion (e.g., (*20*)) as expected between different subsystems. The superstatistics of clades in adaptive landscapes could motivate models that jointly predict changes in traits and diversification, a research area currently struggling with model inadequacy (*48*). This framework could also provide a new paradigm in modeling the distributions of richness, abundance, and resource use in non-neutral communities which can be viewed as emerging from the combination of locally equilibrium subsystems. Non-neutral models in ecology are criticized for their over-parameterization (*49*), yet a persistent counter argument to neutral theory (*42*) is the unrealistic assumption of ecological equivalency and poor prediction of real dynamics (*49*). If ecosystems are viewed as the superposition of many individualistically evolving clades, each exploiting the environment differently and thus obeying a different set of statistics, then diversity dynamics could be parsimoniously predicted with superstatistics while incorporating real biological information on ecological differences between taxa.

Superstatistics is a powerful tool to derive macro-scale predictions from locally fluctuating sub-systems whose evolution is driven by interesting, but complex and difficult to model, biological mechanisms. As such, applications of superstatistics to a wide variety of patterns in ecological and evolutionary systems are ripe for exploration.

4 Methods and Materials

All data processing and analyses were preformed in R (*50*) and all code needed to reproduce our study are provided, with added explanation, in supplemental Appendix A.

4.1 Paleobiology Database data download and filtering

Data on individual fossil occurrences and the ecospace characteristics of Phanerozoic marine invertebrates were downloaded from the Paleobiology Database (PBDB;

https://paleobiodb.org) on 16 November 2018 via the database's API (data retrieval and processing script available in the supplement). Collections were filtered using the same approach as Alroy (*5*) to insure that only well preserved marine invertebrate occurrences were used in subsequent analyses. This filtering resulted in 815,222 unique genus-level occurrences. These were further filtered to exclude those occurrences without family-level taxonomy and those collections with age estimate resolutions outside the 11MY time bins proposed by Alroy (*5*) resulting in 454,033 occurrences. Time bins were compiled from http://fossilworks.org with a custom script reproduced in the supplement. The first and last of these time bins, corresponding to the earliest Cambrian and the latest Cenozoic, were excluded from analysis because their sampling completeness (see below) could not be assessed.

4.2 Correcting for imperfect and potentially biased sampling

We use a new and flexible method to correct for known sampling incompleteness and biases in publication-based specimen databases (*5, 12*). Incompleteness is inherent in all biodiversity samples, the fossil record being no exception (*51–54*). In addition to incompleteness, bias may result from preferential publication of novel taxa (*12*) which exacerbates the difference between poorly-sampled and well-sampled time periods. We therefore develop a simple two-step method: we first correct genus richness for incomplete sampling using the "three-timer" correction (*5*) and then further correct this three-timer richness estimate by accounting for any correlation between the number of genera and the number of publications in a time period.

The three-timer correction estimates the probability of failure to observe a genus in a given time period p_t as the number of times any genus is recorded before and after that period but not during, divided by the number of genera whose occurrence histories span

the period t. To calculate the sampling-corrected richness $\hat{D}_{kt}$ of a clade k in the time period in question, the observed genera within that clade and time period are divided by $1 - p_t$ and their occurrences summed:

$$\hat{D}_{kt} = \sum_{j \in k} \frac{I_{jt}}{1 - p_t} \tag{3}$$

where $j \in k$ designates genera in clade k and I_{jt} is an indicator equal to 1 if genus j occurs in time period t.

$\hat{D}_{kt}$ is the maximum likelihood estimator of richness in a simple occupancy through time type model assuming binomial sampling (*55*), and in that way mimics other proposed methods for the fossil record (*52, 53*). We avoid parametrically modeling the sampling process through time by instead taking a sliding window of time bins from the Cambrian to the Cenozoic. It should be noted that the three-timer correction compares favorably to other similar methods to account for imperfect detection (*56*)

To eliminate further bias due to preferential publication of novel taxa (*12*) we divide the three-timer-corrected number of genera per family per time period by the expected number of genera given publications in that time period. The expected number is calculated by regressing the log-transformed three-timer-corrected number of genera on log-transformed number of publications. There is only a weak trend toward higher richness with more publications (Fig. S1) meaning that the most important correction comes from the three timer correction.

Our new method re-scales each genus occurrence from 0 or 1 (absent or present) to a weighted number continuously ranging between 0 and 1. Because these weighted numbers represent sampling and bias-corrected *occurrences* we can add them arbitrarily, corresponding to the membership of any given genus in any given higher taxonomic group. We must, however, choose a taxonomic level at which to evaluate the relationship between

richness and publications; we choose the level of family because this is the most finely resolved option.

We opt not to use subsampling methods (*12, 51, 57*) because these approaches would not be advisable for clades with few genera. However, our new method achieves similar results to subsampling procedures at the global scale across all clades. We directly compare our predicted time series of global fluctuations in genus richness with results derived from rarefaction and shareholder quorum subsampling (SQS) in Figure S2. Our method shows very minor differences with these subsampling-based predictions and any discrepancies do not impact the statistical distribution of fluctuations (Fig. S2).

4.3 Superstatistical methods

We first derive the superstatistical distribution $P(x)$ by fitting Gaussian distributions to clade-level distributions of fluctuations $p_k(x)$, extracting the inverse variances β_k of those $p_k(x)$, testing the best function to describe the distribution of β_k, and then integrating $P(x) = \int_\beta p_k(x|\beta)f(\beta)$. This process allows no free parameters to hone the fit of $P(x)$ to the data. However, each inverse variance must of course be estimated for each clade, making its good fit to data all the more surprising. To do so we use least squares instead of maximum likelihood because the asymmetric fluctuation distributions of small clades were more reliably fit with curve fitting than with maximum likelihood.

We also estimated $P(x)$ directly from the family-level data using maximum likelihood to compare the fit of our superstatistical prediction and that of a simple Gaussian distribution using AIC. To calculate a likelihood-based confidence interval on our prediction we bootstrapped the data, subsampling fluctuations with replacement from all families and fit superstatistics using maximum likelihood to the aggregated fluctuation distribution of each bootstrap replicate.

References

1. D. M. Raup, J. J. Sepkoski Jr, *et al.*, *Science* **215**, 1501 (1982).

2. J. J. Sepkoski, *Paleobiology* **10**, 246 (1984).

3. N. L. Gilinsky, *Paleobiology* pp. 445–458 (1994).

4. L. H. Liow, N. C. Stenseth, *Proceedings of the Royal Society B: Biological Sciences* **274**, 2745 (2007).

5. J. Alroy, *et al.*, *Science* **321**, 97 (2008).

6. M. Benton, *Science* **268**, 52 (1995).

7. D. H. Erwin, *Trends in Ecology and Evolution* **13**, 344 (1998).

8. P. Bak, K. Sneppen, *Phys. Rev. Lett.* **71**, 4083 (1993).

9. R. V. Solé, S. C. Manrubia, M. Benton, P. Bak, *Nature* **388**, 764 (1997).

10. M. E. J. Newman, B. W. Roberts, *Proceedings of the Royal Society of London B* **260**, 31 (1995).

11. J. W. Kirchner, A. Weil, *Nature* **395**, 337 (1998).

12. J. Alroy, *Science* **329**, 1191 (2010).

13. T. B. Quental, C. R. Marshall, *Science* (2013).

14. D. H. Erwin, *Journal of Experimental Zoology Part B: Molecular and Developmental Evolution* **318**, 460 (2012).

15. S. M. Jordan, T. G. Barraclough, J. Rosindell, *Phil. Trans. R. Soc. B* **371**, 20150221 (2016).

16. G. J. Vermeij, *Evolution and Escalation* (Princeton University Press, Princeton, N.J., 1987).

17. J. S. Madin, *et al.*, *Science* **312**, 897 (2006).

18. G. J. Vermeij, *Palaeogeography, Palaeoclimatology, Palaeoecology* **263**, 3 (2008).

19. G. Simpson, *The Major Features of Evolution* pp. 313–337 (1953).

20. D. L. Rabosky, *PloS one* **9**, e89543 (2014).

21. C. Beck, E. Cohen, *Physica A: Statistical Mechanics and its Applications* **322**, 267 (2003).

22. C. Beck, *Physica D: Nonlinear Phenomena* **193**, 195 (2004).

23. M. A. Fuentes, A. Gerig, J. Vicente, *PLoS ONE* **4**, e8243 (2009).

24. K. Roy, G. Hunt, D. Jablonski, A. Z. Krug, J. W. Valentine, *Proceedings of the Royal Society B: Biological Sciences* **276**, 1485 (2009).

25. M. J. Hopkins, C. Simpson, W. Kiessling, *Ecology letters* **17**, 314 (2014).

26. N. Eldredge, S. J. Gould, *Models in paleobiology* **82**, 115 (1972).

27. C. Newman, J. Cohen, C. Kipnis, *Nature* **315**, 400 (1985).

28. R. K. Bambach, *Biotic interactions in recent and fossil benthic communities* (Springer, 1983), pp. 719–746.

29. A. M. Bush, R. K. Bambach, G. M. Daley, *Paleobiology* **33**, 76 (2007).

30. P. G. Harnik, *Proceedings of the National Academy of Sciences* **108**, 13594 (2011).

31. M. Foote, J. S. Crampton, A. G. Beu, R. A. Cooper, *Paleobiology* **34**, 421 (2008).

32. W. K. Grassmann, *Annals of Operations Research* **8**, 165 (1987).

33. R. H. MacArthur, E. O. Wilson, *The theory of island biogeography* (Princeton University Press, 1967).

34. D. Jablonski, *Annual Review of Ecology, Evolution, and Systematics* **39**, 501 (2008).

35. R. K. Bambach, A. M. Bush, D. H. Erwin, *Palaeontology* **50**, 1 (2007).

36. F. J. Odling-Smee, K. N. Laland, M. W. Feldman, *Niche construction: the neglected process in evolution* (Princeton university press, 2003).

37. M. Foote, *Evolution since Darwin: the first 150 years* pp. 479–510 (2010).

38. E. Mayr, *Systematic Zoology* **14**, 73 (1965).

39. D. H. Erwin, *Palaeontology* **50**, 57 (2007).

40. G. U. Yule, *Philosophical Transactions of the Royal Society of London Series B* **213**, 21 (1925).

41. A. Capocci, G. Caldarelli, *Journal of Physics A: Mathematical and Theoretical* **41**, 224016 (2008).

42. S. P. Hubbell, *The unified neutral theory of biodiversity and biogeography (MPB-32)*, vol. 32 (Princeton University Press, 2001).

43. D. Moen, H. Morlon, *Trends in Ecology & Evolution* **29**, 190 (2014).

401 44. M. Foote, R. A. Cooper, J. S. Crampton, P. M. Sadler, *Proc. R. Soc. B* **285**, 20180122
402 (2018).

403 45. E. L. Berlow, *et al.*, *Proceedings of the National Academy of Sciences* **106**, 187 (2009).

404 46. J. C. Cox, J. E. Ingersoll Jr, S. A. Ross, *Econometrica: Journal of the Econometric*
405 *Society* pp. 385–407 (1985).

406 47. S. Gavrilets, *Fitness landscapes and the origin of species*, vol. 41 (Princeton University
407 Press, 2004).

408 48. D. L. Rabosky, E. E. Goldberg, *Evolution* **71**, 1432 (2017).

409 49. J. Rosindell, S. P. Hubbell, R. S. Etienne, *Trends in ecology & evolution* **26**, 340
410 (2011).

411 50. R Core Team, *R: A Language and Environment for Statistical Computing*, R Foun-
412 dation for Statistical Computing, Vienna, Austria (2018).

413 51. A. I. Miller, M. Foote, *Paleobiology* **22**, 304 (1996).

414 52. M. Foote, *Paleobiology* **42**, 707 (2016).

415 53. J. Starrfelt, L. H. Liow, *Phil. Trans. R. Soc. B* **371**, 20150219 (2016).

416 54. R. A. Close, S. W. Evers, J. Alroy, R. J. Butler, *Methods in Ecology and Evolution*
417 **9**, 1386 (2018).

418 55. J. A. Royle, R. M. Dorazio, *Hierarchical modeling and inference in ecology: the anal-*
419 *ysis of data from populations, metapopulations and communities* (Academic Press,
420 2008).

56. J. Alroy, *Paleobiology* **40**, 374 (2014).

57. A. T. Kocsis, C. J. Reddin, J. Alroy, W. Kiessling, *bioRxiv* p. 423780 (2018).

Acknowledgments

General: We thank John Harte, Rosemary Gillespie, Linden Schneider, Jun Ying Lim, and David Jablonski for helpful discussion. We thank Aaron Clauset and four anonymous reviewers for greatly improving the quality of this manuscript. We thank the many contributors to the Paleobiology Database for making data available.

Funding: AJR thanks funding from Fulbright Chile, the National Science Foundation Graduate Research Fellowship Program and the Omidyar Program at the Santa Fe Institute; MAF thanks FONDECYT 1140278; PM thanks CONICYT PFB-023, ICM-P05-002 and FONDECYT 1161023.

Author contributions: AJR, MAF and PAM designed the study; AJR and MAF preformed the analyses; AJR, MAF and PAM interpreted the results and wrote the manuscript.

Competing interests: none.

Data and materials availability: Data are available through the Paleobiology Database (`paleobiodb.org`) and all code needed to interface with the `paleobiodb.org` API, process, clean, and ultimately analyze the data are available online at `github.com/ajrominger/pa` This github repository also hosts the exact download from `paleobiodb.org` used in this analysis. All required scripts are also available and explained in supplemental Appendix A.

Supplementary materials

S1 Limit distribution of a time-averaged homogeneous origination-extinction process

Fossil taxa gain and lose genera according to an origination-extinction process. We assume that most fossil occurrences of a taxon come from the period of its history when it is dominant and in steady state. In a time slice of duration τ during such a period of steady state the latent per capita rates of origination and extinction would be equal (i.e. $\lambda = \mu \equiv \rho$) and the number of origination or extinctions events (call such events Y) each follow an inhomogeneous Poisson process with rate ρN_t where N_t is the number of genera in the taxon of interest at time t. Allowing N_t to vary smoothly with time, and recognizing that the sum of Poisson random variables remains Poisson, we arrive at the number Y of extinction *or* origination events in τ being distributed

$$Y \sim \text{Pois}\left(\rho \int_{t=0}^{\tau} N(t)dt\right). \tag{4}$$

Under the steady state assumption we can approximate $N(t)$ by $\bar{N}$, the steady state richness, leading to

$$Y \sim \text{Pois}(\rho \bar{N} \tau). \tag{5}$$

This Poisson distribution is asymptotically Gaussian, which is a more appropriate distribution for our sampling and bias-corrected richness estimates because these estimates are not integer-valued but rather continuous random variables. Furthermore, because we use standard time periods of average duration $\tau = 11$MY the the distribution of fluctuations within taxa will be independent of the specific time periods considered. The

Gaussian asymptotics of time-averaged birth-death processes have been proven and explored elsewhere as well (*32*).

S2 Evaluation of sampling bias correction methods

Our sampling and bias-correction method first accounts for imperfect detection within a binomial sampling framework as described in the main text, and then further corrects for potential publication bias using simple log-log regression. We reproduce that regression of log-richness versus log-number of publications here (Fig. S1).

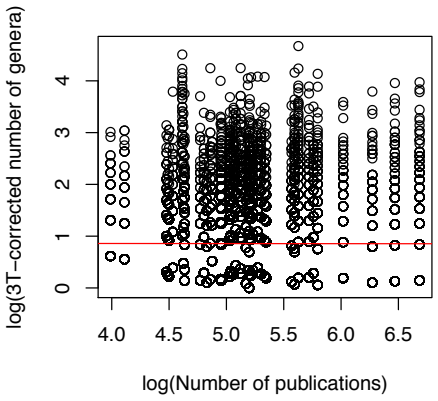

Figure S1: Relationship between number of publications and genus richness at the family level as recorded by the PBDB.

We compare our sampling and bias-correction method to other more established approaches. Specifically we use the newly available R package *divDyn* (*57*) to produce subsampling-based richness estimates for the Phanerozoic timeseries of marine invertebrates. In Figure S2 we compare classical rarefaction and shareholder quorum subsampling (SQS) with our method. All samples were rarified to 120 occurrences, which is approximately the maximum possible rarefied sample size across all time bins, and the

SQS quorum was set to 0.75 to similarly approximate this common sampling denominator across time bins. For both rarefaction and SQS we averaged 50 subsampled replicates.

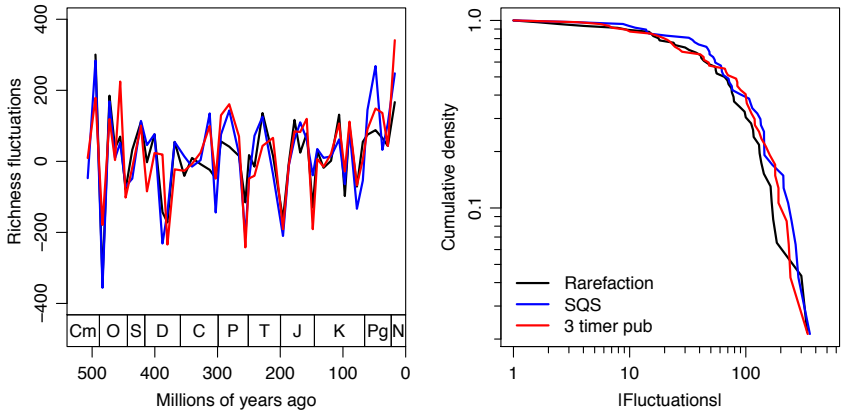

Figure S2: Comparison of rarefaction (black line) and SQS (blue) with our three-timer and publication bias correction method (red). The time-series of all marine invertebrate genera shows general agreement with the only major deviations toward the modern (A). Despite these differences the distribution of fluctuations in genus richness across all marine invertebrates show good agreement (B).

S3 Understanding deviations from superstatistics at higher taxonomic levels

To explore why deviations from super statistics increase with increasing taxonomic level we explore how the distributions of richness fluctuations $p_k(x|\beta_k)$ and fluctuation volatilities $f(\beta_k)$ change with changing taxonomic level. We find that richness fluctuation distributions experience increasing frequencies of outliers (increasing kurtosis) with higher taxonomic level (Fig. S3). We also find that observed fluctuation volatility distributions increasingly depart from a Gamma distribution at the levels of classes and phyla (Fig. S4).

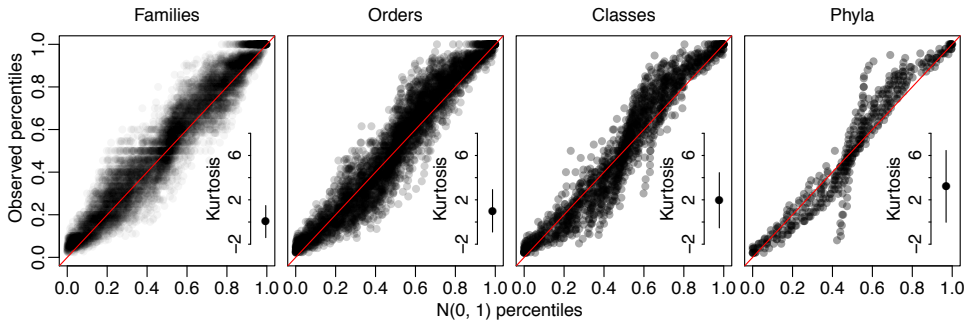

Figure S3: Change in within clade richness fluctuation distributions with increasing taxonomic level. The percentile-percentile plots show how the percentiles of observed re-scaled fluctuation distributions compare to expected percentiles from a Gaussian distribution with mean 0 and variance 1. We can see that families conform to a linear relationship while higher taxa, even at the order level, begin to show s-shaped relationships. Inset plots show how kurtosis increases from 0 (the value for a Gaussian distribution) at the family level to increasingly larger values at higher taxonomic levels.

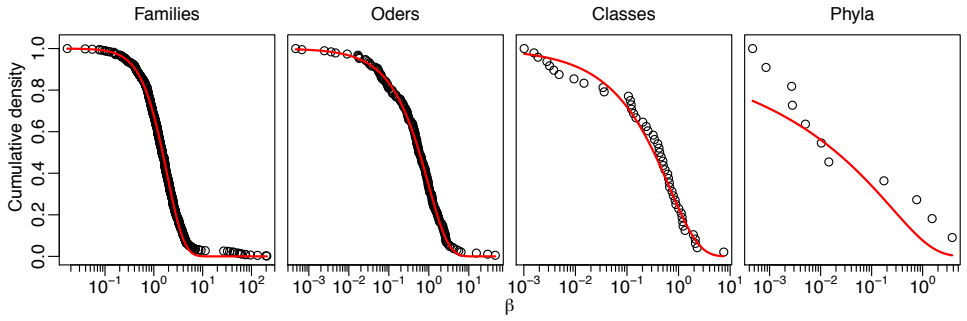

Figure S4: Change in the distributions of β_k across clades of increasing taxonomic level. Points are observed β_k values and red lines are the best-fit Gamma distributions. Deviations increase particularly at the class and phylum levels.

S4 Ecospace occupation of higher taxa

We posit that part of the increasing divergence between superstatistics and observed fluctuations and the increase in fluctuation outliers at higher taxonomic levels is that these higher taxa increasingly aggregate disparate types of organisms. One way to evaluate this idea is to count the ecospace hypercubes (*28, 29, 35*) occupied by taxa at different levels. We use the ecological characteristics reported by the PBDB: taxon environment, motility, life habit, vision, diet, reproduction, and ontogeny. In Figure S5 we find that families comprise, on average, 1 hypercube, families comprise 2 hypercubes on average, and classes and phyla comprise many more.

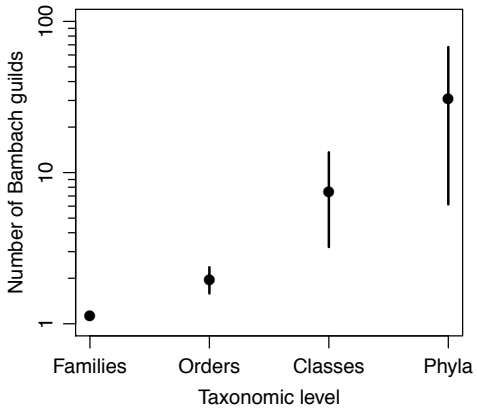

Figure S5: Relationship between number of ecospace hypercubes occupied and taxonomic level.

S5 Relationship between β_k and clade richness

There is likely to be a relationship between richness of clade k and its fluctuation volatility β_k because both extinction and origination (i.e. the formation of new genera) contribute to volatility. Thus we expect that higher variance in richness fluctuations (i.e. smaller $\beta_k = 1/\text{variance}$) will be correlated with higher richness. Indeed, Figure S6 shows this

to be true. In the main text we use permutation to evaluate whether this correlation is responsible for the observed good fit of superstatistics, and find that this correlation alone is not sufficient.

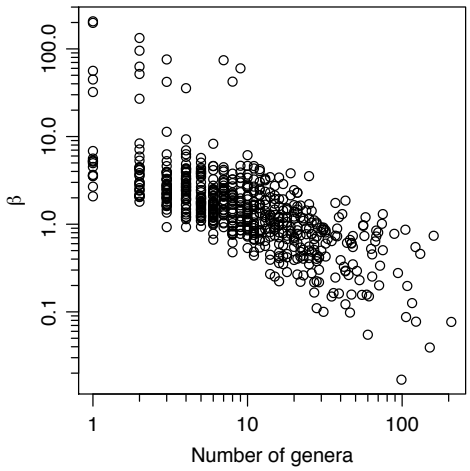

Figure S6: Relationship between fluctuation volatility β_k and genus richness at the family level.

Appendix A: R code to reproduce the study

The easiest way to reproduce this study is to download, clone, or fork the GitHub repository at `github.com/ajrominger/paleo_supStat`. All scripts can then be run on new downloads of the PBDB and modifications to the analyses can be made. The repository is organized into directories `data` containing data and data-cleaning scripts; `R` containing R functions for general use; and `analysis` containing analysis scripts that use the data and R functions. The GitHub repository also contains a manuscript directory (`ms`), this document (`sstat_notebook.Rmd`), and an R script (`sstat_make.R`) that calls each data cleaning and analysis script in sequence to automatically reproduce the entire study. The `data`, `R`, and `analysis` directories can also be recreated from the scripts reproduced below.

The accompanying explanations below (organized by the flow of data acquisition, cleaning, and then analysis) will help the user understand the purpose of each script/function such that they can reproduce the results, or modify the routine.

Lastly, this study depends on the contributed packages *divDyn*, *parallel*, and *R.utils* which should be installed from CRAN, and on a custom package *socorro* which must be installed from GitHub (using the *devtools* package):

```
devtools::install_github('ajrominger/socorro')
```

A1 Getting the data

A1.1 PBDB API

To obtain the PBDB data we make use of the API in script `data/pbdb_data_get.R`, which accesses the API and cleans the data by:

- removing poorly lithified specimens
- removing collections at the basin scale
- including only fine-scale stratigraphy (below the "group" level)
- resolving taxonomy to the genus or subgenus level where available (storing genus or subgenus as `otu`)
- combining multiple records of the same OTU per collection
- importing standardized time bins from fossilworks.org (time bins are scraped with script `data/fossilworks_tbins_intervals.R`)

The data gathering script `data/pbdb_data_get.R` is shown below:

```
# **script to interface with PBDB API and clean resulting data**

# call to the API
show <- paste0(c('ident', 'phylo', 'lith', 'loc', 'time', 'geo', 'stratext',
                 'ecospace'),
               collapse = ',')
version <- '1.2'
base_name <- 'Animalia^Craniata'
min_ma <- 0
max_ma <- 560
timerule <- 'contain'
envtype <- 'marine'

# break-up backbone URI just so it can be nicely displayed
bbURI <- paste0('https://paleobiodb.org/data%s/occs/list.csv?',
```

```
                'base_name=%s&show=%s&limit=all&min_ma=%s&max_ma=%s&',
                'timerule=%s&envtype=%s')

# the actual call to the URI
uri <- sprintf(bbURI,
               version,
               base_name,
               show,
               min_ma,
               max_ma,
               timerule,
               envtype)

# get pbdb occurences
x <- read.csv(uri, as.is = TRUE)

# write out raw data
write.csv(x, 'data/pbdb_data_raw.csv', row.names = FALSE)


# clean up

# remove unnecceary columns
c2rm <- c('record_type', 'reid_no', 'flags', 'identified_name',
          'identified_rank', 'identified_no', 'difference', 'species_name',
          'species_reso', 'lithdescript', 'lithology1', 'minor_lithology1',
          'lithology2', 'lithification2', 'minor_lithology2', 'cc', 'state',
          'county', 'latlng_basis', 'geogcomments', 'geology_comments',
          'zone', 'localsection', 'localbed', 'localorder',
          'regionalsection', 'regionalbed', 'regionalorder',
          'stratcomments')
x <- x[, !(names(x) %in% c2rm)]

# only well lithified specimens
x <- x[x$lithification1 %in% c('', 'lithified'), ]


# no basin-scale collections
x <- x[x$geogscale != 'basin', ]

# fine scale stratigraphy only
x <- x[!(x$stratscale %in% c('group', 'supergroup')), ]

# resolve taxonomy to genus or subgenus where availible
otu <- x$genus
otu[x$subgenus_name != ''] <- ifelse(x$subgenus_reso[x$subgenus_name != ''] == '',
                                     x$subgenus_name[x$subgenus_name != ''],
                                     otu[x$subgenus_name != ''])
otu[x$primary_reso != ''] <- ''
x$otu <- otu
x <- x[x$otu != '', ]
```

```r
# combine multiple records of same otu per collection
x <- x[!duplicated(x[, c('collection_no', 'otu')]), ]


# standard time bins
stages <- read.csv('data/tbins_stages.csv', as.is = TRUE)
earlyTbin <- stages$tbin[match(x$early_interval, stages$name)]
lateTbin <- stages$tbin[match(x$late_interval, stages$name)]
lateTbin[is.na(lateTbin)] <- earlyTbin[is.na(lateTbin)]
earlyTbin[earlyTbin != lateTbin] <- NA

x$tbin <- earlyTbin
x <- x[!is.na(x$tbin), ]


# write out fully processed data
write.csv(x, 'data/pbdb_data.csv', row.names = FALSE)
```

A1.2 Scraping fossilworks

The script to pull Alory's time bins (`data/fossilworks_tbins_intervals.R`) is below:

```r
# **script to scrape time bins from fossilworks.org**

options(stringsAsFactors = FALSE)

# loop through interval info on fossilworks.org
coreURI <- 'http://fossilworks.org/bridge.pl?a=displayInterval&interval_no='

tbinInfo <- lapply(1:1108, function(i) {
    print(i)
    linfo <- try(readLines(paste0(coreURI, i), n = 150))

    if('try-error' %in% class(linfo))
        linfo <- try(readLines(paste0(coreURI, i), n = 150))

    if('try-error' %in% class(linfo)) {
        thisTbin <- thisMax <- thisMin <- thisName <- NA
    } else {
        thisTbin <- gsub('^.*10 million year bin: |<br>.*$', '',
                         linfo[grep('10 million year bin', linfo)])
        thisMax <- as.numeric(gsub('^.*Lower boundary: equal to | Ma.*$|[^0-9\\.]', '',
                                   linfo[grep('Lower boundary: equal to', linfo)]))
        thisMin <- as.numeric(gsub('^.*Upper boundary: equal to | Ma.*$|[^0-9\\.]', '',
                                   linfo[grep('Upper boundary: equal to', linfo)]))
        thisName <- gsub('^.*<p class="pageTitle">|</p>.*$', '',
                         linfo[grep('class="pageTitle"', linfo)])
    }

    return(data.frame(name = ifelse(length(thisName) == 0, NA, thisName),
                      tbin = ifelse(length(thisTbin) == 0, NA, thisTbin),
                      ma_min = ifelse(length(thisMin) == 0, NA, thisMin),
                      ma_max = ifelse(length(thisMax) == 0, NA, thisMax)))
```

```
})

tbinInfo <- do.call(rbind, tbinInfo)


# clean up
# --------

tbinInfo <- tbinInfo[!is.na(tbinInfo$name), ]

# remove 'stage' and equivilant from name
tbinInfo$name <- gsub(' [[:lower:]].*', '', tbinInfo$name)

# split up stages with a '/' into both names
temp <- tbinInfo[grep('/', tbinInfo$name), ]
tbinInfo$name <- gsub('.*/', '', tbinInfo$name)
temp$name <- gsub('/.* ', ' ', temp$name)
tbinInfo <- rbind(tbinInfo, temp)

# fix random typo
tbinInfo$name[tbinInfo$name == 'Cazenovia'] <- 'Cazenovian'


# write out
write.csv(tbinInfo, 'data/tbins_stages.csv', row.names = FALSE)


# also write out summary of each time bin, most importantly (for plottin)
# its midpoint

tbinmid <- sapply(unique(tbinInfo$tbin[!is.na(tbinInfo$tbin)]), function(tbin) {
    tt <- unlist(tbinInfo[tbinInfo$tbin == tbin, c('ma_min', 'ma_max')])
    return(mean(range(tt, na.rm = TRUE)))
})

tbinmid <- sort(tbinmid, decreasing = TRUE)

write.csv(data.frame(tbin = names(tbinmid), ma_mid = as.numeric(tbinmid)),
          'data/tbinsMid.csv', row.names = FALSE)


# lastly confirm the durations of tbins
length(tbinmid)

tbinrange <- sapply(unique(tbinInfo$tbin[!is.na(tbinInfo$tbin)]), function(tbin) {
    tt <- unlist(tbinInfo[tbinInfo$tbin == tbin, c('ma_min', 'ma_max')])
    return(diff(range(tt, na.rm = TRUE)))
})

mean(tbinrange)
```

A2 Three timer and publication bias correction

Once the data have been downloaded and cleaned, we correct for incomplete and biased sampling with the script `data/pbdb_3TPub_make.R` which sources the function `R/make3TPub.R` to generate the main output: a matrix with time bins as rows, taxa (families in this case) as columns and bias-corrected richness as cells.

```
# **this script produces diversity estimates per family per time bin from PBDB data
# corrected by the '3 timers method' and for possible publication bias**

# source function to produce a matrix of time by taxon with cells
# of corrected diversity
source('R/make3TPub.R')

# load other needed funcitons

# source('code/sstat_comp.R')
# source('code/sstat_methods.R')
# source('code/Px_gam.R')

# load and prepare data
# ---------------------

pbdbDat <- read.csv('data/pbdb_data.csv', as.is = TRUE)

# make column for midpoint ma
pbdbDat$ma_mid <- (pbdbDat$max_ma + pbdbDat$min_ma) / 2

# get rid of poor temporal resolution
pbdbDat <- pbdbDat[pbdbDat$tbin != '', ]

# get rid of bad taxonomy
pbdbDat <- pbdbDat[pbdbDat$family != '', ]
pbdbDat <- pbdbDat[pbdbDat$otu != '', ]

# get bin times
pbdbDat$mid_ma <- (pbdbDat$min_ma + pbdbDat$max_ma) / 2
pbdbTime <- sort(tapply(pbdbDat$mid_ma, pbdbDat$tbin, mean))
pbdbDat$tbin <- factor(pbdbDat$tbin, levels = names(pbdbTime))


# data.frame to hold publication, diversity and 3T stat
famTbinBias <- aggregate(list(div = pbdbDat$otu), list(fam = pbdbDat$family,
                                                       tbin = pbdbDat$tbin),
                         function(x) length(unique(x)))


# three timer stat and publication bias
# -------------------------------------

# matrix to determine three timers and part timers (sensu alroy 2008)
mt <- matrix(0, nrow = nlevels(pbdbDat$tbin),
             ncol = nlevels(pbdbDat$tbin))
diag(mt) <- -10
mt[abs(row(mt) - col(mt)) == 1] <- 1
```

```
# loop through and compute three timers and part timers
timers <- lapply(split(pbdbDat$tbin, pbdbDat$otu),
                 function(x) {
                     # browser()
                     tbins <- integer(nlevels(x))
                     tbins[as.integer(unique(x))] <- 1
                     t3 <- as.integer(mt %*% tbins == 2)
                     tp <- as.integer(mt %*% tbins == -8)

                     return(cbind(t3, tp))
                 })

# compute 3 timer stat from 3 timers and part timers
timers <- array(unlist(timers), dim = c(nrow(timers[[1]]), 2, length(timers)))
t3stat <- 1 - rowSums(timers[, 1, ]) / (rowSums(timers[, 1, ]) + rowSums(timers[, 2, ]))

# add to data.frame holding all info to be saved
famTbinBias$T3Stat <- t3stat[match(famTbinBias$tbin,
                                   levels(pbdbDat$tbin))]
famTbinBias$T3Div <- famTbinBias$div / famTbinBias$T3Stat

# record pubs per tbin
tbinPub <- tapply(pbdbDat$reference_no, pbdbDat$tbin,
                  function(x) length(unique(x)))
famTbinBias$tbinPub <- tbinPub[famTbinBias$tbin]

# calculate corrected diversity
pdf('ms/figSupp_divByPub_foo.pdf', width = 4, height = 4)
pbdbFamDiv <- with(famTbinBias,
                  make3TPub(div, T3Stat, tbinPub, fam, tbin, pbdbTime,
                            minPub = 10, plotit = TRUE))
dev.off()

# write out corrected diversity
write.csv(pbdbFamDiv, 'data/pbdb_3TPub-corrected.csv')


# for permutational d-stat tests we need diversity at the genus level;
# make that here

# a data.frame holding only one record per genus per family per time bin
pbdbOcc <- pbdbDat[!duplicated(pbdbDat[, c('tbin', 'family', 'otu')]), ]

genTbinBias <- parallel::mclapply(which(!is.nan(famTbinBias$T3Stat)), mc.cores = 3,
                                  FUN = function(i) {
                                      dat <- pbdbOcc[pbdbOcc$family == famTbinBias$fam[i] &
                                                         pbdbOcc$tbin == famTbinBias$tbin[i],
                                                     c('tbin', 'family', 'otu')]
                                      dat$T3Occ <- 1 / famTbinBias$T3Stat[i]
                                      dat$tbinPub <- famTbinBias$tbinPub[i]

                                      return(dat)
                                  }
```

```
)

genTbinBias <- do.call(rbind, genTbinBias)
pbdbGenDiv <- data.frame(genTbinBias[, c('tbin', 'family', 'otu')],
                         T3PubDiv = genTbinBias$T3Occ /
                             exp(predict(pbdbPubLM,
                                         newdata = data.frame(
                                             logPub = log(genTbinBias$tbinPub)))))

# write it out as a tidy data frame (not turned into a matrix) this will be easier
# for permuting
write.csv(pbdbGenDiv, file = 'data/pbdb_3TPub_genera.csv', row.names = FALSE)
```

Here is the guts of the `make3TPub` function

```
#' @description function to produce a matrix of time by taxa with cells of corrected diversity
#' @param rawDiv the raw diversity of each taxon in each time interval
#' @param t3stat the 3 timer stat for each diversity record
#' @param pub the number of publications associated with each diversity record
#' @param taxa the taxon names for each diversity record
#' @param tbin the time interval of each diversity record
#' @param tbinTime times associated with each `tbin`
#' @param minPub minimum number of publications for inclusion in regression analysis
#' @param plotit logical, should plot of taxon richness versus number of publications be made
#' @return a matrix with rows corresponding to time intervals and columns to the given taxa
#' each cell in the matrix represents corrected taxon richness


make3TPub <- function(rawDiv,  t3stat,  pub,  taxa,  tbin,  tbinTime,
                      minPub = 10,  plotit = FALSE) {
    # put data together so can be universally manipulated
    x <- data.frame(rawDiv = rawDiv, t3stat = t3stat, pub = pub, taxa = taxa, tbin = tbin)
    x$tbin <- as.character(x$tbin)
    x$taxa <- as.character(x$taxa)

    x <- x[!is.na(t3stat) & pub >=  minPub, ]

    tbinTime <- tbinTime[names(tbinTime) %in% x$tbin]

    # 3-timer correction
    t3cor <- x$rawDiv/x$t3stat

    # publication correction
    logPub <- log(x$pub)
    pubLM <- lm(log(t3cor)~logPub)
    pbdbPubLM <<- pubLM # save regression to global env

    pubResid <- exp(pubLM$residuals)

    # plot so you can verify cuttoff etc.
    if(plotit) {
        plot(log(x$pub), log(t3cor),
             xlab = 'log(Number of publications)',
             ylab = 'log(3T-corrected number of genera)')
```

```
        abline(pubLM, col = 'red')
    }

    tbinTaxa <- socorro::tidy2mat(x$tbin, x$taxa, pubResid)

    return(tbinTaxa[names(sort(tbinTime, decreasing = TRUE)), ])
}
```

Our publication correction is simple: we take the exponential of the residual of this relationship:

$$\log(\text{3T-corrected richness}) = \beta_0 + \beta_1 \log(\text{number of publications}) + \epsilon$$

The exponentiated residual amounts to dividing the three-timer corrected richness by a publication correction factor: (3T-corrected richness)$/e^{\hat{y}}$ where $\hat{y}$ is the predicted trend line from the above relationship.

So we can use this multiplicative publication correction factor in addition to the similarly multiplicative three-timer correction to bias-correct individual genus-level occurrences. This is important when we permute these bias-corrected genera across families to create a null set of d-statistics for our superstatistical fit.

We can check our correction against other popular methods. In the script `analysis/pbdb_divCurve.R` we specifically compare simple rarefaction, with the SQS method, with our three-time and publication bias correction methods. All these various methods have close agreement. The script `analysis/pbdb_divCurve.R` is shown below:

```
# **a script to compare our 3TPub curve to other estimates of richness through
# the Phanerozoic**

# package with diversity dynamics subsampling functions
library(divDyn)

# package for plotting
library(socorro)

# load and prep data
pbdbFamDiv <- read.csv('data/pbdb_3TPub-corrected.csv', row.names = 1)
pbdbDat <- read.csv('data/pbdb_data.csv', as.is = TRUE)
tbin <- read.csv('data/tbinsMid.csv', as.is = TRUE)
tbin$tbin <- factor(tbin$tbin, levels = tbin$tbin)
pbdbDat$tbin <- factor(pbdbDat$tbin, levels = levels(tbin$tbin))
pbdbDat$tbinNum <- as.integer(pbdbDat$tbin)

pbdbDatUnique <- pbdbDat[!duplicated(paste0(pbdbDat$collection_no, pbdbDat$otu)), ]

# subsampled richness
pbdbCR <- subsample(pbdbDatUnique, bin = 'tbinNum', tax = 'otu', iter = 50, q = 120,
                    type = 'cr', unit = 'reference_no')
pbdbSQS <- subsample(pbdbDatUnique, bin = 'tbinNum', tax = 'otu', iter = 50, q = 0.75,
                     ref = 'reference_no', type = 'sqs', singleton = 'ref')

# our new richness estimate
pbdbT3Pub <- rowSums(pbdbFamDiv)


# plot fluctuations to see that they're comprable
pdf('ms/figSupp_divEstComp.pdf', width = 8, height = 4)
layout(matrix(1:2, nrow = 1))
```

```
par(mar = c(4.5, 2.5, 0, 0.5) + 0.5, mgp = c(2, 0.75, 0))
plot(1, xlim = c(540, 0), ylim = c(-400, 400), type = 'n', xaxt = 'n',
     xlab = '', ylab = 'Richness fluctuations', xaxs = 'i')
paleoAxis(1)
mtext('Millions of years ago', side = 1, line = 3.5)

lines(tbin$ma_mid[-1], diff(pbdbCR$divCSIB), col = 'black', lwd = 2)
lines(tbin$ma_mid[-1], diff(pbdbSQS$divCSIB), col = 'blue', lwd = 2)
lines(tbin$ma_mid[-c(1:2, nrow(tbin))], diff(pbdbT3Pub), col = 'red', lwd = 2)


par(mar = c(3, 3, 0, 0) + 0.5, mgp = c(2, 0.75, 0))
plot(simpECDF(c(1, abs(diff(pbdbT3Pub))), complement = TRUE), col = 'red', log = 'xy',
     type = 'l', lwd = 2, xlim = c(1, 500),
     panel.first = {
         lines(simpECDF(c(1, abs(diff(pbdbCR$divCSIB))), complement = TRUE),
               col = 'black', lwd = 2)
         lines(simpECDF(c(1, abs(diff(pbdbSQS$divCSIB))), complement = TRUE),
               col = 'blue', lwd = 2)
     },
     axes = FALSE, frame.plot = TRUE,
     xlab = '|Fluctuations|', ylab = 'Cumulative density')
logAxis(1:2)


legend('bottomleft', legend = c('Rarefaction', 'SQS', '3 timer pub'),
       lty = 1, lwd = 2, col = c('black', 'blue', 'red'), bty = 'n')

dev.off()
```

A3 Super statistical analysis of 3TPub-corrected PBDB data

Once data have been bias-corrected we can complete their super-statistical analysis. We do that in the script `analysis/pbdb_sstat.R` shown here:

```
# **script to run super stat analysis on PBDB data and make plots**

# source needed functions
R.utils::sourceDirectory('R', modifiedOnly = FALSE)
library(socorro) # for plotting

# load and prepare data
# ---------------------

pbdbFamDiv <- read.csv('data/pbdb_3TPub-corrected.csv', row.names = 1)


# coarsen to higher taxonomic groupings

pbdbTax <- read.csv('data/pbdb_taxa.csv', as.is = TRUE)

#' helper function to coarsen taxonomic resolution of `pbdbFamDiv` object
```

```r
#' @param level a character string specifying the taxonomic level (from order through phylum)

coarsenTaxa <- function(level) {
    m <- tidy2mat(pbdbTax$family[match(colnames(pbdbFamDiv), pbdbTax$family)],
                  pbdbTax[match(colnames(pbdbFamDiv), pbdbTax$family), level],
                  rep(1, ncol(pbdbFamDiv)))
    m <- m[colnames(pbdbFamDiv), ]

    out <- as.matrix(pbdbFamDiv) %*% m
    out <- out[, colnames(out) != '']

    return(out)
}

pbdbOrdDiv <- coarsenTaxa('order')
pbdbClsDiv <- coarsenTaxa('class')
pbdbPhyDiv <- coarsenTaxa('phylum')


# tbin midpoints
tbinMid <- read.csv('data/tbinsMid.csv', as.is = TRUE)
tbinNames <- tbinMid$tbin
tbinMid <- as.numeric(tbinMid[, 2])
names(tbinMid) <- tbinNames

tbinMid <- tbinMid[rownames(pbdbFamDiv)]


# super stat analysis
# -------------------


# calculate flux for families
pbdbFamFlux <- calcFlux(pbdbFamDiv)

# calculate the mean flux
famMeans <- sapply(pbdbFamFlux, mean)
mean(famMeans)
sd(famMeans)


# make sstat object for families
sstatPBDBfam3TP <- sstatComp(pbdbFamFlux, minN = 10, plotit = FALSE)

# deltaAIC
logLik(sstatPBDBfam3TP) - sum(dnorm(unlist(sstatPBDBfam3TP$Px.sub), log = TRUE))

# likelihood CI for family-level sstat analysis
sstatPBDBfam3TPCI <- bootMLE.sstat(sstatPBDBfam3TP, B = 1000, useAll = FALSE)


# do the same for higher taxo levels
pbdbOrdFlux <- calcFlux(pbdbOrdDiv)
```

```r
sstatPBDBOrd <- sstatComp(pbdbOrdFlux, minN = 10, plotit = FALSE)

pbdbClsFlux <- calcFlux(pbdbClsDiv)
sstatPBDBCls <- sstatComp(pbdbClsFlux, minN = 10, plotit = FALSE)

pbdbPhyFlux <- calcFlux(pbdbPhyDiv)
sstatPBDBPhy <- sstatComp(pbdbPhyFlux, minN = 10, plotit = FALSE)

# save the sstat analyses for future use
save(sstatPBDBfam3TP, sstatPBDBOrd, sstatPBDBCls, sstatPBDBPhy,
     file = 'data/pbdb_sstat_objects.RData')


# plot all sstat analyses
pdf('ms/fig_Px.pdf', width = 4.25 * 1.25, height = 4 * 1.25)

layout(matrix(1:4, nrow = 2, byrow = TRUE))
par(oma = c(3, 3, 0, 0) + 0.5, mar = c(0.1, 0.1, 1.51, 0.1),
    mgp = c(2, 0.5, 0), cex.lab = 1.4)

plot(sstatPBDBfam3TP, xlim = c(1e-04, 5e+02), ylim = c(8e-05, 1),
     xaxt = 'n', yaxt = 'n',
     panel.first = quote(mlePoly(sstatPBDBfam3TPCI$sstat, PPx.gam,
                                 col = hsv(alpha = 0.25), border = NA)))
mtext('Families', side = 3, line = 0)
logAxis(2, expLab = TRUE)
legend('topright', legend = 'A', bty = 'n', cex = 1.4)

plot(sstatPBDBOrd, xlim = c(1e-04, 5e+02), ylim = c(8e-05, 1), xaxt = 'n', yaxt = 'n',
     addLegend = FALSE)
mtext('Orders', side = 3, line = 0)
legend('topright', legend = 'B', bty = 'n', cex = 1.4)

plot(sstatPBDBCls, xlim = c(1e-04, 5e+02), ylim = c(8e-05, 1), xaxt = 'n', yaxt = 'n',
     addLegend = FALSE)
mtext('Classes', side = 3, line = 0)
logAxis(1:2, expLab = TRUE)
legend('topright', legend = 'C', bty = 'n', cex = 1.4)

plot(sstatPBDBPhy, xlim = c(1e-04, 5e+02), ylim = c(8e-05, 1), xaxt = 'n', yaxt = 'n',
     addLegend = FALSE)
mtext('Phyla', side = 3, line = 0)
logAxis(1, expLab = TRUE)
legend('topright', legend = 'D', bty = 'n', cex = 1.4)

mtext('|Fluctuations|', side = 1, outer = TRUE, line = 2)
mtext('Cumulative density', side = 2, outer = TRUE, line = 2)
dev.off()




# plot p_k(x|b) and f(beta) for families
```

```r
# ------------------------------------

# idea for normality test: sample 1 from each order and do ks test on that subsampled set

# highlight individual trajectories
loFam <- 'Tainoceratidae' # nautiloid
miFam <- 'Lophospiridae' # sea snails
hiFam <- 'Spondylidae' # bivalve
lo <- pbdbFamDiv[, loFam]
mi <- pbdbFamDiv[, miFam]
hi <- pbdbFamDiv[, hiFam]
cols <- hsv(h = c(0.7, 0.45, 0.12), s = c(0.7, 1, 1), v = c(0.8, 0.8, 1))
names(cols) <- c('hi', 'mi', 'lo')

# make CDF for all scale family-level fluctuations
pAll <- lapply(sstatPBDBfam3TP$raw.pk,
               function(x) simpECDF(scale(x)[, 1], complement = TRUE))

pHighlight <- pAll[c(loFam, miFam, hiFam)]

pAll <- do.call(rbind, pAll)

# function to help with individual trajectory plotting
trajLines <- function(t, x, ...) {
    x[x == 0] <- NA
    alive <- range(which(!is.na(x)))

    x[min(alive) - 1] <- 0
    x[max(alive) + 1] <- 0

    t <- t[!is.na(x)]
    x <- x[!is.na(x)]

    lines(t, x, ...)
}

# the actual plotting

pdf('ms/fig_pkx-fbeta.pdf', width = 4.25 * 1.25, height = 4 * 1.25)

layout(matrix(c(1, 2, 1, 3), nrow = 2))

par(oma = c(0, 3, 0, 0) + 0.25, mar = c(4, 0, 0, 0) + 0.25,
    mgp = c(2, 0.5, 0), cex.lab = 1.4)

plot(1, xlim = c(540, 0), xaxt = 'n', xaxs = 'i', xlab = '',
     ylim = c(0, max(lo, mi, hi, na.rm = TRUE)), type = 'n')

trajLines(tbinMid, lo, col = cols['lo'], lwd = 2)
trajLines(tbinMid, mi, col = cols['mi'], lwd = 2)
trajLines(tbinMid, hi, col = cols['hi'], lwd = 2)

text(c(450, 230, 10), c(4, 5.25, 2), labels = c(miFam, loFam, hiFam),
```

```
     col = cols[c('mi', 'lo', 'hi')], pos = c(3, 4, 2))

paleoAxis(1)
mtext('Millions of years ago', side = 1, line = 3.5)
mtext('Standardized richness', side = 2, line = 2)

legend('topright', legend = 'A', pch = NA, bty = 'n', cex = 1.4)

# scale fluctuations
par(mar = c(3, 0, 1, 0) + 0.25)
plot(pAll, xlim = c(-4, 4), col = 'gray', ylim = c(0, 1.025),
     xlab = 'Scaled fluctuations')
mtext('Cumultive density', side = 2, line = 2)

for(i in 1:length(pHighlight)) lines(pHighlight[[i]], col = cols[i], lwd = 2)

curve(pnorm(x, lower.tail = FALSE), lwd = 2, add = TRUE)

legend('topright', legend = 'B', pch = NA, bty = 'n', cex = 1.4)

# CDF of beta
betaCDF <- simpECDF(sstatPBDBfam3TP$beta, complement = TRUE)
plot(betaCDF, ylim = c(0, 1.025),
     log = 'x', xaxt = 'n', yaxt = 'n',
     xlab = expression(beta), col = 'gray')

theseBeta <- sstatPBDBfam3TP$beta[c(loFam, miFam, hiFam)]
points(theseBeta, approxfun(betaCDF)(theseBeta), bg = cols, pch = 21, cex = 1.2)

logAxis(1, expLab = TRUE)

curve(pgamma(x, sstatPBDBfam3TP$gam.par[1], sstatPBDBfam3TP$gam.par[2],
             lower.tail = FALSE),
      col = 'black', lwd = 2, add = TRUE)

legend('topright', legend = 'C', pch = NA, bty = 'n', cex = 1.4)

dev.off()
```

Now we can calculate a measure of goodness of fit with the Kolmogorov-Smirnov test statistics D. That is done in the script `analysis/pbdb_dperm.R`. This script uses a permutation approach to create a null distribution of test statistics. The goal is to see if the good fit of super-statistics at the family and order levels is purely from the number of different groupings at those levels, regardless of the biology that might be going on to make those levels actually mechanistically meaningful. So to achieve such a null, we permute orders across families, calculate the D-statistics of those permuted groupings, and compare to the real D-statistics from the actual biological groupings. The script is shown below:

```
# **script to caculate d stats on sstat objects and null permutations**

library(socorro)
R.utils::sourceDirectory('R', modifiedOnly = FALSE)

# read in data
pbdbGenDiv <- read.csv('data/pbdb_3TPub_genera.csv', as.is = TRUE)
```

```r
pbdbTax <- read.csv('data/pbdb_taxa.csv', as.is = TRUE)
# pbdbFamDiv <- read.csv('data/pbdb_3TPub-corrected.csv', row.names = 1)
load('data/pbdb_sstat_objects.RData')

# the indeces of otu names in `pbdbTax` ordered by their occurence in `pbdbGenDiv`
# needed to match permuted families to genera
genHash <- match(pbdbGenDiv$otu, pbdbTax$otu)


#' function to calculate KS test d-stat on sstat objects
#' @param x an sstat object

ks.sstat <- function(x) {
    dat <- unlist(x$Px.sub)
    dat <- abs(dat)

    # cumulative density function
    pfun <- function(X) x$PPx(X, comp = TRUE)

    # cumulative prob observed and from theory
    n <- length(dat)
    pobs <- (n:1) / n
    pthr <- pfun(sort(dat))

    # the statisic is the difference between obs and thr
    out <- pthr - pobs

    return(max(out, 1 / n - out, na.rm = TRUE))
}


# sstat on real (non-permuted) data
dObsFam <- ks.sstat(sstatPBDBfam3TP)
dObsOrd <- ks.sstat(sstatPBDBOrd)
dObsCls <- ks.sstat(sstatPBDBCls)
dObsPhy <- ks.sstat(sstatPBDBPhy)


# repeatedly permute data and calculate null ks statistics
B <- 500
dNull <- parallel::mclapply(1:B, mc.cores = 8, FUN = function(i) {
    newFam <- sample(pbdbTax$family)
    newDiv <- tidy2mat(pbdbGenDiv$tbin, newFam[genHash], pbdbGenDiv$T3PubDiv)
    newFlux <- calcFlux(newDiv)
    newSstat <- sstatComp(newFlux, minN = 10, plotit = FALSE)

    ks.sstat(newSstat)
    # return(ks.sstat(newSstat))
})

dNull <- unlist(dNull)
```

```
# save output in case it's ever needed
save(dNull, file = 'data/dnull.RData')


# plotting
pdf('ms/fig_dStat.pdf', width = 4, height = 4)
# colors for plotting taxa
tcols <- colorRampPalette(hsv(c(0.12, 0, 0.02), c(1, 0.9, 0.7), c(1, 0.8, 0.3)))(4)

par(mar = c(3, 3, 0, 0) + 0.5, mgp = c(2, 0.75, 0))
denFill(dNull, xlim = range(dNull, dObsFam, dObsOrd, dObsCls, dObsPhy) * c(0.9, 1.1),
        xlab = 'D-statistic', main = '')

abline(v = dObsFam, lwd = 2, col = tcols[1])
text(dObsFam, 1.25 * mean(par('usr')[3:4]), labels = 'Families', col = tcols[1],
     srt = 90, pos = 4)

abline(v = dObsOrd, lwd = 2, col = tcols[2])
text(dObsOrd, 1.25 * mean(par('usr')[3:4]), labels = 'Orders', col = tcols[2],
     srt = 90, pos = 4)

abline(v = dObsCls, lwd = 2, col = tcols[3])
text(dObsCls, 1.25 * mean(par('usr')[3:4]), labels = 'Classes', col = tcols[3],
     srt = 90, pos = 4)

abline(v = dObsPhy, lwd = 2, col = tcols[4])
text(dObsPhy, 1.25 * mean(par('usr')[3:4]), labels = 'Phyla', col = tcols[4],
     srt = 90, adj = c(0, -0.5))

dev.off()
```

This permutation null test is in part motivated by the correlation between the genus richness of a family and its β_k value. We demonstrate this correlation in the script `analysis/pbdb_betaRichness.R` which is reproduced below:

```
library(socorro)

load('data/pbdb_sstat_objects.RData')
pbdbDat <- read.csv('data/pbdb_data.csv', as.is = TRUE)

divFamRaw <- tapply(pbdbDat$otu, pbdbDat$family, function(x) length(unique(x)))

pdf('ms/figSupp_betaByRich.pdf', width = 4, height = 4)
par(mar = c(3, 3, 0, 0) + 0.5, mgp = c(2, 0.75, 0))
plot(divFamRaw[names(sstatPBDBfam3TP$beta)], sstatPBDBfam3TP$beta, log = 'xy',
     xlab = 'Number of genera', ylab = expression(beta),
     axes = FALSE, frame.plot = TRUE)
logAxis(1:2)
dev.off()
```

Now that we are reasonably convinced that these superstatistical findings are not just an artifact of taxonomy or clade size, we can explore why we see deviations from super statistics with increasing taxonomic level. We first explore how well the Gaussian $p_k(x|\beta_k)$ fit at each taxonomic level in the script `analysis/pkx_diffK.R` shown here:

```r
# **script to compare within clade fluctuation distributions at different taxonomic levels**

library(socorro)

# source needed functions
R.utils::sourceDirectory('R', modifiedOnly = FALSE)

load('data/pbdb_sstat_objects.RData')


# for each family, calculate aggregated eCDF and distribution of kurtosis values
famECDF <- lapply(sstatPBDBfam3TP$Px.sub, function(x) {
    simpECDF(scale(x)[, 1], complement = TRUE)
})
famECDF <- do.call(rbind, famECDF)
famKurt <- sapply(sstatPBDBfam3TP$Px.sub, kurt)

ordECDF <- lapply(sstatPBDBOrd$Px.sub, function(x) {
    simpECDF(scale(x)[, 1], complement = TRUE)
})
ordECDF <- do.call(rbind, ordECDF)
ordKurt <- sapply(sstatPBDBOrd$Px.sub, kurt)

clsECDF <- lapply(sstatPBDBCls$Px.sub, function(x) {
    simpECDF(scale(x)[, 1], complement = TRUE)
})
clsECDF <- do.call(rbind, clsECDF)
clsKurt <- sapply(sstatPBDBCls$Px.sub, kurt)

phyECDF <- lapply(sstatPBDBPhy$Px.sub, function(x) {
    simpECDF(scale(x)[, 1], complement = TRUE)
})
phyECDF <- do.call(rbind, phyECDF)
phyKurt <- sapply(sstatPBDBPhy$Px.sub, kurt)


#' @description function to plot theoretical and observed percentiles
#' @param x aggregated eCDF
#' @param ... additional plotting parameters

ppECDF <- function(x, ...) {
    alpha <- 0.75 / (1 + exp(0.0003 * (nrow(x) - 300))) # nicely scale transparency
    plot(pnorm(x[, 1], lower.tail = FALSE), x[, 2], pch = 16,
         col = gray(0, alpha = alpha), xlim = 0:1, ylim = 0:1, ...)

    abline(0, 1, col = 'red')
}


#' @description function to plot summary of kurtosis values distribution
#' @param x kurtosis values
#' @param ... additional plotting parameters
```

```r
kurtInset <- function(x, ...) {
    allMean <- c(mean(famKurt), mean(ordKurt), mean(clsKurt), mean(phyKurt))
    allSD <- c(sd(famKurt), sd(ordKurt), sd(clsKurt), sd(phyKurt))

    plot(1, mean(x), pch = 16,
         ylim = range(allMean - allSD, allMean + allSD) * c(1.5, 1.25),
         xaxt = 'n', frame.plot = FALSE, yaxs = 'i',
         ...)
    segments(x0 = 1, y0 = mean(x) - sd(x), y1 = mean(x) + sd(x))
}




# plot it
pdf('ms/figSupp_pkx_allTaxa.pdf', width = 9, height = 3)

split.screen(c(1, 4))

# first plots of the ECDF's
screen(1, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
ppECDF(famECDF)
mtext('Families', side = 3, line = 0.5)

screen(2, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
ppECDF(ordECDF, yaxt = 'n')
mtext('Orders', side = 3, line = 0.5)

screen(3, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
ppECDF(clsECDF, yaxt = 'n')
mtext('Classes', side = 3, line = 0.5)

screen(4, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
ppECDF(phyECDF, yaxt = 'n')
mtext('Phyla', side = 3, line = 0.5)


mtext('N(0, 1) percentiles', side = 1, outer = TRUE, line = 1.25)
mtext('Observed percentiles', side = 2, outer = TRUE, line = 1.25)

close.screen(all.screens = TRUE)


# now inset plots of kurtosis
```

```
start <- 1/4 + 0.01
swidth <- 1/32
increment <- 1/4 - 1/64
s <- split.screen(matrix(c(start + 0 * increment, start + 0 * increment + swidth, 0.25, 0.6,
                           start + 1 * increment, start + 1 * increment + swidth, 0.25, 0.6,
                           start + 2 * increment, start + 2 * increment + swidth, 0.25, 0.6,
                           start + 3 * increment, start + 3 * increment + swidth, 0.25, 0.6),
                         ncol = 4, byrow = TRUE), erase = FALSE)

for(i in 1:4) {
    screen(s[i], new = FALSE)
    par(mar = rep(0, 4), mgp = c(1, 0.25, 0))
    kurtInset(switch(i,
                     `1` = famKurt,
                     `2` = ordKurt,
                     `3` = clsKurt,
                     `4` = phyKurt),
              tcl = -0.1)
    mtext('Kurtosis', side = 2, line = 1.25)
}


close.screen(all.screens = TRUE)

dev.off()
```

We can also explore how well the $f(\beta_k)$ fit at different taxonomic levels in the script `analysis/pbeta_diffK.R` reproduced below:

```
# **script to compare within clade volatility distributions at different taxonomic levels**

library(socorro)

# source needed functions
R.utils::sourceDirectory('R', modifiedOnly = FALSE)

load('data/pbdb_sstat_objects.RData')


#' @description function to plot f(beta) distribution
#' @param obj the sstat object
#' @param thrCol color for plotting of theoretical curve

fbetaPlot <- function(obj, thrCol = 'red', ...) {
    betaCDF <- simpECDF(obj$beta, complement = TRUE)
    plot(betaCDF, ylim = c(0, 1.025),
         log = 'x', xaxt = 'n', yaxt = 'n',
         xlab = expression(beta), ...)

    logAxis(1, expLab = TRUE)

    curve(pgamma(x, obj$gam.par[1], obj$gam.par[2],
                 lower.tail = FALSE),
          col = thrCol, lwd = 2, add = TRUE)
```

```
}


# the plotting
pdf('ms/figSupp_fbeta_allTaxa.pdf', width = 9, height = 3)

split.screen(c(1, 4))

screen(1, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
fbetaPlot(sstatPBDBfam3TP)
axis(2)
mtext('Families', side = 3, line = 0.5)

screen(2, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
fbetaPlot(sstatPBDBOrd)
mtext('Oders', side = 3, line = 0.5)

screen(3, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
fbetaPlot(sstatPBDBCls)
mtext('Classes', side = 3, line = 0.5)

screen(4, new = FALSE)
par(mar = c(0.3, 0.3, 1.5, 0.3), oma = c(2.5, 2.5, 0, 0),
    mgp = c(2, 0.5, 0))
fbetaPlot(sstatPBDBPhy)
mtext('Phyla', side = 3, line = 0.5)

mtext(expression(beta), side = 1, outer = TRUE, line = 1.25)
mtext('Cumulative density', side = 2, outer = TRUE, line = 1.25)

close.screen(all.screens = TRUE)
dev.off()
```

Part of our argument about the failure of superstatistics at higher taxonomic levels is that these higher taxa aggregate increasingly disparate subtaxa. To investigate this idea we look at the number of ecospace hypercubes represented by the average taxon at each taxonomic level in the script `analysis/pbdb_ecoEvoSpace.R` shown here:

```
# **a script to evaluate hour occupancy of eco-evolutionary space changes
# across taxonomy**

library(socorro)

pbdbDat <- read.csv('data/pbdb_data.csv', as.is = TRUE)

# extract only the eco/evo/life history data and remove duplicates
# `taxon_environment`, `reproduction`, `ontogeny`
eeSpaceDat <- pbdbDat[, c('phylum', 'class', 'order', 'family', 'otu',
```

```
                          'taxon_environment', 'motility', 'life_habit',
                          'vision', 'diet', 'reproduction', 'ontogeny')]
eeSpaceDat <- eeSpaceDat[!duplicated(eeSpaceDat), ]

# remove entries that are all missing
eeSpaceDat <- eeSpaceDat[rowSums(eeSpaceDat[, -(1:5)] != '') != 0, ]


#' function to determine how many eco-evo hypercubes are occupied by each taxonomic level
#' @param tax the taxonomic unit to consider
#' @param eeDat a data.frame containing eco-evo data
eeOcc <- function(tax, eeDat) {
    sapply(split(eeDat[tax != '', ], tax[tax != '']),
           function(x) sum(!duplicated(x)))
}

#' bootstraps ee space occupancy
#' @param x the vector of niche occupancies
#' @param B number of bootrap replicates
#' @param fun the function to apply to each replicate
eeOccBoot <- function(x, B, fun) {
    replicate(B, fun(sample(x, length(x), replace = TRUE)))
}


# calculate eco-evolutionary space occupancy of each taxonomic level
famEE <- eeOccBoot(eeOcc(eeSpaceDat$family, eeSpaceDat[, -(1:5)]), 500, mean)
ordEE <- eeOccBoot(eeOcc(eeSpaceDat$order, eeSpaceDat[, -(1:5)]), 500, mean)
clsEE <- eeOccBoot(eeOcc(eeSpaceDat$class, eeSpaceDat[, -(1:5)]), 500, mean)
phyEE <- eeOccBoot(eeOcc(eeSpaceDat$phylum, eeSpaceDat[, -(1:5)]), 500, mean)


# plotting
pdf('ms/figSupp_eeSpaceOcc.pdf', width = 4, height = 3.5)

par(mar = c(3, 3, 0, 0) + 0.5, mgp = c(2, 0.75, 0))
plot(1:4, ylim = c(1, 100), type = 'n', log = 'y', yaxt = 'n', xaxt = 'n',
     xlab = 'Taxonomic level', ylab = 'Number of Bambach guilds')
axis(1, at = 1:4, labels = c('Families', 'Orders', 'Classes', 'Phyla'))
logAxis(2)
segments(x0 = 1:4, y0 = c(min(famEE), min(ordEE), min(clsEE), min(phyEE)),
         y1 = c(max(famEE), max(ordEE), max(clsEE), max(phyEE)), lwd = 2)
points(1:4, c(mean(famEE), mean(ordEE), mean(clsEE), mean(phyEE)), pch = 16, cex = 1.2)

dev.off()
```

A4 Helper functions

All the above analyses make use of helpful functions in the `R` directory. We reproduce those functions below:

```
#' helper function to calculate corrected flux
#' @param x the matrix of corrected diversities over which to calculate fluxes
```

```
calcFlux <- function(x) {
    apply(x, 2, function(X) {
        flux <- diff(c(0, X))
        return(flux[flux != 0])
    })
}
```

```
gammaLS <- function(data,comp=FALSE) {
    par.init <- c(mean(data)^2,mean(data))/var(data)
    optim(par.init,gamma.ss,data=data,comp=comp)
}

gamma.ss <- function(pars,data,comp) {
    shape <- pars[1]
    rate <- pars[2]
    tabz <- table(data)
    xval <- as.numeric(names(tabz))
    yval <- cumsum(as.numeric(tabz))/sum(tabz)
    if(comp) {
        yval <- 1 - yval
        yval <- c(1,yval[-length(yval)])
        lower <- FALSE
    } else {
        lower <- TRUE
    }
    difz <- pgamma(xval,shape=shape,rate=rate,lower.tail=lower) - yval
    sum(difz^2)
}
```

```
#' @description function to produce a matrix of time by taxa with cells of corrected diversity
#' @param rawDiv the raw diversity of each taxon in each time interval
#' @param t3stat the 3 timer stat for each diversity record
#' @param pub the number of publications associated with each diversity record
#' @param taxa the taxon names for each diversity record
#' @param tbin the time interval of each diversity record
#' @param tbinTime times associated with each `tbin`
#' @param minPub minimum number of publications for inclusion in regression analysis
#' @param plotit logical, should plot of taxon richness versus number of publications be made
#' @return a matrix with rows corresponding to time intervals and columns to the given taxa
#' each cell in the matrix represents corrected taxon richness


make3TPub <- function(rawDiv,  t3stat,  pub,  taxa,  tbin,  tbinTime,
                      minPub = 10,  plotit = FALSE) {
    # put data together so can be universally manipulated
    x <- data.frame(rawDiv = rawDiv, t3stat = t3stat, pub = pub, taxa = taxa, tbin = tbin)
    x$tbin <- as.character(x$tbin)
    x$taxa <- as.character(x$taxa)

    x <- x[!is.na(t3stat) & pub >=  minPub, ]

    tbinTime <- tbinTime[names(tbinTime) %in% x$tbin]

    # 3-timer correction
```

```
    t3cor <- x$rawDiv/x$t3stat

    # publication correction
    logPub <- log(x$pub)
    pubLM <- lm(log(t3cor)~logPub)
    pbdbPubLM <<- pubLM # save regression to global env

    pubResid <- exp(pubLM$residuals)

    # plot so you can verify cuttoff etc.
    if(plotit) {
        plot(log(x$pub), log(t3cor),
             xlab = 'log(Number of publications)',
             ylab = 'log(3T-corrected number of genera)')
        abline(pubLM, col = 'red')
    }

    tbinTaxa <- socorro::tidy2mat(x$tbin, x$taxa, pubResid)

    return(tbinTaxa[names(sort(tbinTime, decreasing = TRUE)), ])
}
```

```
normLS <- function(data,comp=FALSE) {
    par.init <- c(mean(data),sd(data))
    optim(par.init,norm.ss,data=data,comp=comp)
}

norm.ss <- function(pars,data,comp) {
    mean <- pars[1]
    sd <- pars[2]
    tabz <- table(data)
    xval <- as.numeric(names(tabz))
    yval <- cumsum(as.numeric(tabz))/sum(tabz)
    if(comp) {
        yval <- 1 - yval
        yval <- c(1,yval[-length(yval)])
        lower <- FALSE
    } else {
        lower <- TRUE
    }
    difz <- pnorm(xval,mean=mean,sd=sd,lower.tail=lower) - yval
    sum(difz^2)
}
```

```
# pdf for P(x) with f(beat) ~ Gamma
#' @param x diversity fluctuation value
#' @param shape the shape parameter of the gamma distribution
#' @param rate the rate parameter  of the gamma distribution

Px.gam <- PxGam <- function(x, shape, rate) {
    scale <- 1 / rate
    n <- 2 * shape
    b0 <- scale * shape
```

```
    t1 <- gamma((n+1) / 2) / gamma(n / 2)
    t2 <- sqrt(b0 / (pi * n))
    t3 <- (1 + (b0 * x^2) / n)^-((n + 1) / 2)

    t1 * t2 * t3
}

# cdf for P(x) with f(beat) ~ Gamma
#' @param x diversity fluctuation value
#' @param shape the shape parameter of the gamma distribution
#' @param rate the rate parameter  of the gamma distribution
#' @param comp logical, whether to compute the complement or not (`comp = TRUE` is
#' equivilant to `lower.tail = FALSE` for typical `p` functions [e.g. `pnorm`])

PPx.gam <- PPxGam <- function(x, shape, rate, comp=TRUE) {
    if(length(x) == 1) {
        intgral <- integrate(PxGam, 0, x, shape = shape, rate = rate)
        if(intgral$message != 'OK') print(intrgral$message)
        val <- intgral$value

        if(comp) {
            return(1 - 2 * val)
        } else {
            return(2 * val)
        }

    } else {
        # recursive handeling for multiple `x` values
        return(sapply(x, function(X) PPxGam(X, shape, rate, comp)))
    }
}
```

```
#' @description gives the log likelihood function under sstat model
#' @param par the parameter values
#' @param dat the data

sstatLL <- function(par, dat) {
    -sum(log(Px.gam(dat,par[1], par[2])))
}


#' @description finds the maximum likelihood estimate of the superstats model
#' @param dat the data to fit

sstatMLE <- function(dat) {
    optim(c(0.55, 0.17), sstatLL, method = 'BFGS', hessian = TRUE,
          dat = dat)
}


#' @description bootstrap likelihood for super stats model
#' @param x the `sstat` object
#' @param B the number of boostrap replicates
#' @param useAll logical, whether all orders, or only those with the minimum number of
```

```
#' occurences as specified
#' in `make3TPub` argument `minPub` should be used

bootMLE.sstat <- function(x, B = 1000, useAll = FALSE) {
    if(useAll) {
        theseDat <- x$Px.raw
    } else {
        theseDat <- x$Px.sub
    }

    boots <- replicate(B, {
        subDat <- sapply(theseDat, sample, size = 1)

        thisMLE <- try(sstatMLE(subDat), silent = TRUE)

        if(class(thisMLE) !=  'try-error') {
            if(thisMLE$convergence !=  0) {
                out <- rep(NA, 2)
            } else {
                out <- thisMLE$par
            }
        } else {
            out <- rep(NA, 2)
        }
        out
    })

    sstatOut <- rbind(quantile(boots[1, ], c(0.025, 0.975), na.rm = TRUE),
                      quantile(boots[2, ], c(0.025, 0.975), na.rm = TRUE))
    rownames(sstatOut) <- c('shape', 'rate')

    return(list(sstat = sstatOut))
}


#' @description logLik for sstat class
#' @param x the `sstat` object
#' @param fitted logical, was the model fitted by max likelihood or computed from first
#' principles
#' @param useAll logical, should all data be used, or only those taxa that have greater
#' than `minN` occurrences
#' as specified in `sstatComp`

logLik.sstat <- function(x, fitted = TRUE, useAll = FALSE) {
    if(useAll) {
        theseDat <- unlist(x$Px.raw)
    } else {
        theseDat <- unlist(x$Px.sub)
    }

    lik <- sum(log(x$Px(theseDat)))

    if(fitted) {
```

```
        attr(lik, 'df') <- 2
    } else {
        attr(lik, 'df') <- 0
    }

    class(lik) <- 'logLik'

    return(lik)
}
```

```
#' @description plot method for sstat class
#' @param x the `sstat` object
#' @param sstatCol color for super stats fit
#' @param normCol color for Gaussian fit
#' @param showNorm logical, should Gaussian fit be shown
#' @param addLegend logical, should legend be added
#' @param ... other parameters passed to `plot.default`

plot.sstat <- function(x, sstatCol = 'red', normCol = 'blue',
                       showNorm = TRUE, addLegend = TRUE, ...) {
    thisECDF <- socorro::simpECDF(abs(unlist(x$Px.sub)), complement = TRUE)

    # helper function to deal with optional axis arguments
    .axissetup <- function(side) {
        if(sprintf('%saxt', side) %in% names(pargs)) {
            if(pargs[[sprintf('%saxt', side)]] == 'n') {
                assign(sprintf('%saxfun', side), function(...) {}, pos = 1)
            } else {
                if(side %in% pargs$log) {
                    assign(sprintf('%saxfun', side), socorro::logAxis, pos = 1)
                } else {
                    assign(sprintf('%saxfun', side), axis, pos = 1)
                }
            }
        } else {
            if(side %in% pargs$log) {
                assign(sprintf('%saxfun', side), socorro::logAxis, pos = 1)
            } else {
                assign(sprintf('%saxfun', side), axis, pos = 1)
            }
        }
    }

    pargs <- list(...)
    if(!('log' %in% names(pargs))) pargs$log <- 'xy'
    if(!('xlab' %in% names(pargs))) pargs$xlab <- '|Fluctuations|'
    if(!('ylab' %in% names(pargs))) pargs$ylab <- 'Cumulative density'
    .axissetup('x')
    .axissetup('y')

    pargs$xaxt <- 'n'
    pargs$yaxt <- 'n'

    do.call(plot, c(list(x = thisECDF), pargs))
```

```
    xaxfun(1)
    yaxfun(2)

    PPx <- x$PPx
    curve(PPx(x, comp = TRUE), col = sstatCol, lwd = 2, add = TRUE)

    if(showNorm) {
        thisSD <- sd(unlist(x$Px.sub))
        curve(2*pnorm(x, 0, thisSD, lower.tail = FALSE), col = normCol, lwd = 2, add = TRUE)
    }

    if(addLegend) {
        leg <- c('Observed', 'Superstatistics')
        col <- c(par('fg'), sstatCol)
        pch <- c(ifelse('pch' %in% names(list(...)), list(...)$pch, 1), NA)
        pt.lwd <- c(1, NA)
        pt.cex <- c(1, NA)
        lwd <- c(NA, 2)

        if('panel.first' %in% names(list(...))) {
            leg <- c(leg, 'Superstatistics CI')
            col <- c(col, socorro::colAlpha(sstatCol, 0.25))
            pt.lwd <- c(pt.lwd, 1)
            pt.cex <- c(pt.cex, 2)
            lwd <- c(lwd, NA)
            pch <- c(pch, 15)
        }

        if(showNorm) {
            leg <- c(leg, 'Gaussian')
            col <- c(col, normCol)
            pt.lwd <- c(pt.lwd, NA)
            pt.cex <- c(pt.cex, NA)
            lwd <- c(lwd, 2)
            pch <- c(pch, NA)
        }

        extracex <- ifelse('cex' %in% names(list(...)), list(...)$cex, 1)
        legend('bottomleft', legend = leg, col = col, pch = pch, pt.lwd = pt.lwd,
               pt.cex = pt.cex*extracex, lwd = lwd, bty = 'n')
    }
}


#' @description function to add confidence interval polygon from ML analysis
#' @param ci the matrix of CI intervals for the parameter values returned by `bootMLE.sstat`
#' @param fun the CDF function to plug the parameter values into
#' @param ... further arguments passed to `polygon` (e.g. `col`, `boarder`, etc.)

mlePoly <- function(ci, fun, ...) {
    n <- 50
    x <- seq(par('usr')[1], par('usr')[2], length = n)
    x <- c(x, rev(x))
```

```
    if(par('xlog')) x <- 10^x

    y <- c(fun(x[1:n], ci[1, 1], ci[2, 2]), fun(x[(1:n) + n], ci[1, 2], ci[2, 1]))

    polygon(x = x, y = y, ...)
}
```

```
sstatComp <- function(grp.data,minN=15,xlab="Absolute Fluctuation",
                      ylab="Cumulative Density",leg=TRUE,plotit=TRUE) {
    these2use <- sapply(grp.data,length) >= minN
    p2use <- grp.data[these2use]

    cat("computing Gaussian fit for p_k(x|sigma) \n")
    pk.par <- sapply(p2use,function(x) unlist(normLS(x)[c("par","value")]))
    pk.par <- t(pk.par)
    colnames(pk.par) <- c("mu","sig","ss")

    cat("re-centering \n")
    for(i in 1:length(p2use)) {
        p2use[[i]] <- p2use[[i]] - pk.par[i,"mu"]
    }

    cat("computing f(beta) \n")
    f.beta.par <- gammaLS(1/(pk.par[,"sig"])^2)$par
    fuent.par <- c(n=2*f.beta.par[1],b0=f.beta.par[1]*f.beta.par[2])

    cat("computing P(x) \n")
    this.Px <- function(x) Px.gam(x,f.beta.par[1],f.beta.par[2])
    this.PPx <- function(x,comp=TRUE) PPxGam(x,f.beta.par[1],f.beta.par[2],comp)

    out <- list(gam.par=f.beta.par,sspar=fuent.par,beta=1/(pk.par[,"sig"])^2,
                sumSq=pk.par[,"ss"],minN=minN,raw.pk=p2use,
                Px.raw=grp.data,Px.sub=p2use,incld=these2use,Px=this.Px,PPx=this.PPx)

    class(out) <- "sstat"

    if(plotit) plot(out)

    return(out)
}
```