Bayesian inference of power law distributions

Kristina Grigaityte[a,b] **and Gurinder Atwal**[b,1]

[a] Watson School of Biological Sciences
[b] Cold Spring Harbor Laboratory, Cold Spring Harbor, NY
[1] To whom correspondence should be addressed. E-mail: atwal@cshl.edu

Observed data from many research disciplines, ranging from cellular biology to economics, often follow a particular long-tailed distribution known as a power law. Despite the ubiquity of power laws, determining the exact form of the distribution from sampled data remains challenging. Further complications arise from the uncertainty that an unknown weighted mixture of power laws may provide a more complete description of the data. We present a principled and probabilistic solution to these issues by developing a Bayesian approach to accurately estimate power law exponents, the number of mixtures, and their weights, for both discrete and continuous data. We derive an uninformative prior distribution that is invariant to reparameterization of the power law exponents, and demonstrate its effectiveness to accurately infer exponents, even in the low sample limit. Finally, we provide a comprehensive and documented software package, written in Python, of our Bayesian inference methodology, freely available at https://github.com/AtwalLab/BayesPowerlaw.

Power Law | Bayesian Inference | Python Package | Jeffreys prior | Mixture Model

Power law distributions abound in empirical data. Some of the examples of where power laws appear include geography (city population sizes (1)), astronomy (moon crater sizes(2)), literature (word usage (3)), biology (T cell clone sizes (4, 5)), networks (connections per node) (6), statistical physics (order parameters in phase transitions) (7), computational neuroscience (power spectra of natural images) (8), and many more. The widespread and enigmatic appearance of these long-tailed distributions has generated much research and debate into their origins and their detection. A number of differing theories have been proposed to explain the generation of power law behavior and their study still provides an active rich area of interest. On the other hand, accurate detection of actual power law distributions in both simulated and real world data remains challenging, especially in the low sample limit, precisely due to the long-tailed nature of the distributions.

Formally, power law distributions over a variable of interest x depend on a single exponent (power) γ as shown below,

$$p(x|\gamma) = \frac{x^{-\gamma}}{\zeta(\gamma)}, \quad (x_{\min} \leq x \leq x_{\max}), \tag{1}$$

where, in full generality, the power law behavior is assumed to only exist within a certain range of x values, $(x_{\min}, x_{\max})$. The function $\zeta(\gamma)$ does not depend on x and ensures correct normalization of the probability distribution, taking on differing forms depending on whether x is a discrete or continuous variable,

$$\zeta(\gamma) = \sum_{n=x_{\min}}^{x_{\max}} \frac{1}{n^{\gamma}} \qquad \text{[discrete]}, \tag{2}$$

$$\zeta(\gamma) = \frac{x_{\max}^{-\gamma+1}}{1-\gamma} - \frac{x_{\min}^{-\gamma+1}}{1-\gamma} \qquad \text{[continuous]}. \tag{3}$$

Note that in the discrete case, when $x_{\min} = 1$ and $x_{\max} = \infty$, $\zeta(\gamma)$ becomes identical to the Riemann zeta function over real values of γ. A defining characteristic of power laws is their scale invariance, that is, rescaling the x variable by some constant factor k leaves the distribution unchanged, up to a normalization factor, $p(kx|\gamma) = k^{-\gamma}p(x|\gamma) \propto p(x|\gamma)$. This property underlies the appearance of power law distributions in critical physical systems, leading to the powerful concept of universality and renormalization group in statistical physics and field theory. The quickest way to suspect a power law distribution is plotting the data on a log-log scale and observing a straight line (Figure 1A,B) (3, 4). However, since the data is generally noisy or of limited sample size, the observed straight line often deviates at the tail (3). The tail is also interesting in that it could potentially be used to detect abnormalities in the data. For example, T cell clone size power law distribution exhibits a heavy tail, which is likely a representative of expanded T cell clones in addition to sampling noise (4). The deviation from the straight line can sometimes be observed with the smaller values as well. It may curve upwards (concave) or downwards (convex), which makes it questionable whether the distribution really follows a power law (Figure 1C,D). Observing a straight line on a log-log scale is not sufficient to conclude that the distribution indeed follows the power law. It is important to perform vigorous goodness-of-fit tests to determine how well the power law distribution model fits the data. One of the most common ways to perform such tests is using Kolmogorov-Smirnov statistic (9, 10).

Power law distributions are quantified by identifying the exponent γ. A simple, and still commonly used, method to fit power laws to data is to perform least-square linear regression on the logarithm of the histogramed data (3, 11). However, this method can result in significantly inaccurate estimates of the exponents due to the large and biased fluctuations in the long tail of the distributions, where data is sparse. Currently, the most accurate method for fitting power laws is using maximum likelihood algorithms (3, 9, 10, 12). While efficient and accurate this method, however, does not provide uncertainty of the inferred exponent value and, as we will show shortly, does not perform well with low sample sizes. In this article we introduce a

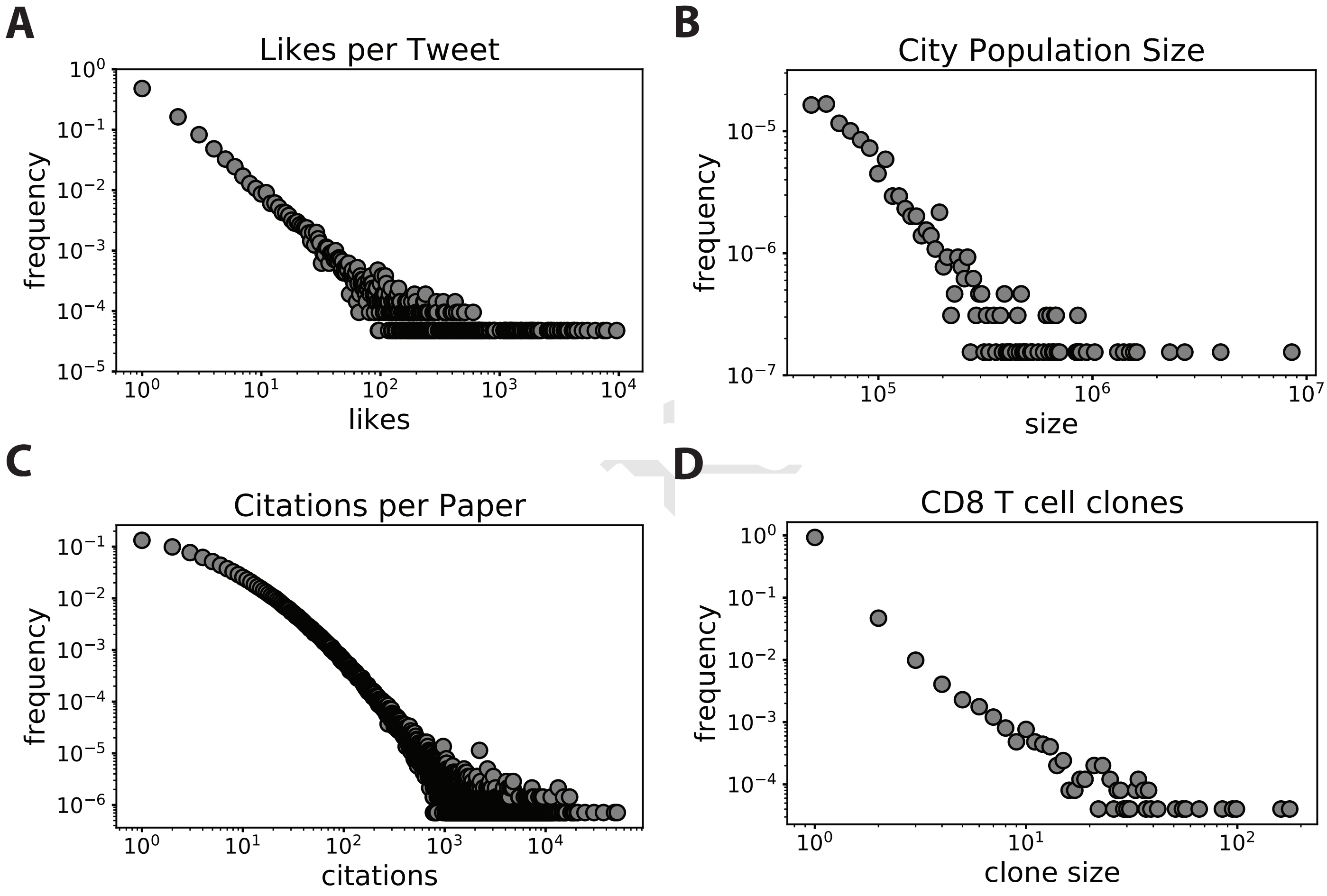

Fig. 1. Examples of power law distributions. A) A distribution of likes per tweet - discrete power law. B) A distributions of city population sizes - continuous power law. C) A distribution of citations per paper - an example of the concave power law. D) A distribution of CD8 T cell clone sizes - an example of the convex power law

novel power law fitting approach using Bayesian inference together with a developed Python software that overcomes limitations posed by maximum likelihood algorithms. In addition, all previous research assumed a single power law distribution and ignored the possibility of power law mixtures, which we also incorporated into our software.

Bayesian inference with Jeffreys prior. We developed a Python package for fitting power law distributions using Bayesian inference to identify the posterior distribution of exponents given the data,

$$p(\gamma|\text{data}) = \frac{p(\text{data}|\gamma)\text{p}(\gamma)}{p(\text{data})}. \qquad [4]$$

Specifically, we utilize Markov Chain Monte Carlo (MCMC), with a Metropolis-Hastings (MH) algorithm to sample the posterior. Briefly, MH algorithm samples from the posterior by performing a random walk in the parameter space. We draw potential exponent candidates from a Gaussian distribution and then either accept or reject them by comparing the likelihood of target exponent to the current one. First, we initialize the algorithm by picking an exponent value of 1.01 since generally power law exponents tend to be closer to 1. Next, we perform a burn in for at least 1000 iterations to quickly walk to a roughly correct value. Normally, burn in algorithm is identical to post burn in sampling with the difference that values sampled during burn in do not occur in the final posterior distribution (13, 14). However, we modified the burn in algorithm to make the random walk faster. Common practice is to reject or accept candidate values under some probability, which may lead to accepting a target exponent that is less likely to be correct than the current one (15). During burn in alone, we do not include this probability and only accept the target exponent if it is more likely to be correct. This allows for a much quicker burn in, saving time and getting to the right value range in fewer iterations. The burn in is then followed by the main MCMC in which we do introduce the acceptance probability and thus use the algorithm in the conventional way.

One of the challenges from using Bayesian inference is selecting a good prior. Here we derived Jeffreys prior for both continuous and discrete power law distributions and used it in our algorithm. Jeffreys prior is advantageous as it is non-informative for the parameter space (16). Moreover, Jeffreys prior has also been shown to maximize the mutual information between parameters and predictions, and thereby selects simple effective models in a principled way (17). For power law distributions, we find Jeffreys prior to be (see Supplementary for derivation)

$$p(\gamma) \propto \sqrt{\frac{(\zeta')^2}{\zeta^2} - \frac{\zeta''}{\zeta}}. \qquad [5]$$

where the derivatives of ζ are with respect to γ. This formula is correct for both discrete and continuous data, as long as an appropriate ζ function is used. In a case of continuous power law distribution where $x_{\min} = 1$ and $x_{\max} = \infty$ our Jeffreys prior simplifies into equation Eq. (6)

$$p(\gamma) \propto \frac{1}{\gamma - 1}, \quad (\gamma > 1) \qquad [6]$$

which, interestingly, is a truncated power law.

Power law mixtures. Historically, when a distribution follows a power law it is assumed that there is only one correct exponent (3, 10, 12). However, some data, particularly biological data, is often very complex and could potentially be a mix of multiple populations each of which exhibits a power law distribution with a distinct exponent. Fitting such mixture of power law distributions under assumption that only one power law is present will likely lead to inaccurate fit and will fail to recognize the mixture. Therefore, we modify the power law equation Eq. (1) to account for potential mixtures in the data

$$p(\text{data}|\{\gamma\}, \{\text{w}\}) = \sum_{m=1}^{M} w_m \frac{\text{data}^{-\gamma_\text{m}}}{\zeta(\gamma_m)}, \qquad [7]$$

where M is number of power laws in the mixture, and $\{\gamma\}$, $\{w\}$ are vectors of length M containing the exponents and their corresponding weights for each power law in the mixture respectively. This adjustment introduces extra parameters into our previously univariate Bayesian inference equation,

$$p(\{\gamma\}, \{w\}|\text{data}) = \frac{p(\text{data}|\{\gamma\}, \{\text{w}\})\text{p}(\{\gamma\}, \{\text{w}\})}{p(\text{data})}. \qquad [8]$$

Assuming conditional independence of the data, we obtain the following likelihood function,

$$\begin{aligned} l &= p(x_1|\{\gamma\}, \{w\})p(x_2|\{\gamma\}, \{w\})...p(x_n|\{\gamma\}, \{w\}) \\ &= \prod_{n=1}^{N} p(x_n|\{\gamma\}, \{w\}) = \prod_{n=1}^{N} \sum_{m=1}^{M} \omega_m \frac{x_n^{-\gamma_m}}{\zeta(\gamma_m)}, \end{aligned} \qquad [9]$$

where N and M are samples size and mixture size respectively. Subsequently, the log likelihood function becomes

$$L = \log l = \sum_{n=1}^{N} \log \sum_{m=1}^{M} \omega_m \frac{x_n^{-\gamma_m}}{\zeta(\gamma_m)}. \qquad [10]$$

Results

To demonstrate how our algorithm performs when fitting data sets that follow either a single or two power laws of distinct exponent, we performed simulations where we generated data from single and mixed power law probability distributions. We show an example of generated data sets of exponents 1.2 (Figure 2A), 2.7 (Figure 2C) and the mixture of the two with weights 0.25 and 0.75 respectively (Figure 2E). We applied our algorithm to identify the posterior distributions of exponents to see how well our estimated exponent compares to the exponent used for simulations. First, we ran the algorithm assuming that each simulated data set, including the mixed one consisted of only one power law. We found that the algorithm performs well for the data sets that did consist of a single power law (Figure 2B,D), but failed to identify any of the exponents from the mixture, instead resulting in the posterior somewhat in between the two correct values (Figure 2F). This implies that fitting a data set under the assumption of only one power law can lead to inaccurate and misleading quantification of the distribution if the distribution consists of more than one power law. Our algorithm, when specified, can perform multi-parameter fitting and identify the distinct exponents in the power law mixture as well as the weights of each power law. Running our algorithm on the mixed data set of simulated power laws when mix of two is specified resulted in the posterior distribution exhibiting two peaks corresponding to correct exponents (Figure 2G) as well as the weights of each power law distribution in the mixture (Figure 2H).

Comparing different fitting methods. To compare our Bayesian inference method to other methods widely used for fitting power law distributions - linear regression-based estimators (11, 18) and maximum likelihood (9, 12) - we performed simulations. Power law distributions of various sample sizes for a range of exponents from one to five were generated and fit using linear regression, maximum likelihood and Bayesian inference (either flat or Jeffreys prior) algorithms (Figure 3). We performed 20 simulations per each exponent and sample size, and calculated the mean and standard deviation across those 20 simulations. In cases of Bayesian inference algorithm, we also compare the mean of posterior distribution means and the mean of the maximum value of the posterior distribution. It is clear that Bayesian inference performs best in all instances with no difference in accuracy between mean and max. Although, mean is advantageous in that it comes with standard deviation, while max does not. Maximum likelihood performs somewhat accurately in most but not all cases, with linear regression performing very poorly. For all sample sizes - 1000, 100, and even as low as 30 - Bayesian inference performs extremely well. While maximum likelihood is almost as good as Bayesian inference when sample size is large (with an exception of exponents close to 1), it performs poorly when sample size is low such as 30. Linear regression here is done on only the initial part of the distribution ($x_{\min} = 1$, $x_{\max} = 10$) since otherwise it is completely incorrect due to a heavy and noisy tail. While it performs fairly well for large sample size of exponents in lower range values, it becomes completely inaccurate as sample sizes diminish.

Mixture fitting. To further validate our algorithm with regard to fitting mixtures of two power laws, we performed systematic mixture simulations of various exponent and sample size pairs. We show that the algorithm performs best if (i) the simulated distribution contains a power law with at least one exponent value close to 1 and (ii) if the power law distribution of higher exponent carries larger weight (Figure 4). The reason lower exponent results in better performance of our algorithm is that during the power law simulations there is a higher probability of sampling larger values thus creating a heavier tail. With high exponent, the probability of sampling high values gets lower and the resulting power law is made up of mostly low values. Thus, when combined, the power law with the lower exponent represents the full range of the resulting mixed distribution, while the power law with the higher exponent only represents a small part of the resulting distribution and is negligible in the tail. Thus, when fitting the mixed distribution, the power law with the lower exponent dominates the algorithm. However, this dominance of the lower exponent gets reduced when the higher exponent power law carries the larger weight. Nevertheless, even if the two conditions are not met, our algorithm can still identify the mixture of exponents in most cases, although less accurately and with the higher uncertainty. The algorithm performs similarly with regard to the weight values as well (Figure 4B).

Empirical Datasets. Finally, we used our algorithm to fit power law distributions found in empirical data, most of which have previously identified exponents via different methods (3, 10, 11). The difference in our case is that we attempt to fit these distributions assuming that the distribution consists of 1, 2 or 3 power laws. We determine which is the best fit by calculating the Bayesian Information Criteria (BIC). The obtained information on the mixture type, exponents and weights for each data set are shown in Table 1. It is important to note, that the exponents reported here were acquired from fitting the full distribution of each data set. This way we made sure that by eliminating some lower bound of data points we did not miss a potential mixture of power laws. We found that only T cell clone size distribution consisted of a mixture of 2 power laws suggesting two generating mechanisms of T cells. Interestingly, this distribution deviates from the straight line on a log-log scale in a convex manner (Figure 1C). This is consistent with the appearance of the simulated mixture distribution of two power laws with differing exponents (Figure 2E).

In addition, we compared the published data set fits acquired using our algorithm to the ones that are published in (3). For a fair comparison, we refit the data with matching $x_{\min}$ to those reported. The results comparing the exponents are shown in Table 2.

Table 1. Real dataset fits

Dataset	Type	Mix	Exponent	Weight	N
1. Alice in Wonderland	d	1	1.73±0.02	-	1465
2. Lithuanian novel	d	1	1.98±0.01	-	36896
3. Harry Potter	d	1	1.63±0.01	-	5690
4. Moby Dick	d	1	1.82±0.01	-	11975
5. Citations per paper	d	1	1.36±0.00	-	1401704
6. Species per genus	d	1	1.67±0.02	-	1250
7. Fatalities per terror incident	d	1	1.88±0.01	-	15531
8. Injuries per terror incident	d	1	1.51±0.01	-	12557
9. Last name usage	c	1	1.81±0.00	-	151670
10. Moon crater size	c	1	2.42±0.02	-	5184
11. Solar flares	c	1	1.03±0.00	-	12771
12. City population size	c	1	2.27±0.05	-	760
13. Likes per tweet	d	1	1.73±0.00	-	20996
14. All T cell clone size	d	2	2.10±0.01 and 4.26±0.01	0.04±0.01 and 0.96±0.01	173408
15. CD4 T cell clone size	d	2	1.94±0.01 and 4.39±0.01	0.01±0.01 and 0.99±0.01	66587
16. CD8 T cell clone size	d	2	2.14±0.01 and 4.82±0.01	0.1±0.01 and 0.9±0.01	24864

Table 2. Real dataset fits

Dataset	Xmin	Bayes Fit (mean)	Bayes Fit (max)	Maximum Likelihood (Clauset et al, 2009)
1. Moby Dick	7	1.96±0.02	1.96	1.95
2. Citations per paper	160	2.41±0.01	2.40	3.16
3. Species per genus	4	2.07±0.07	2.04	2.40
4. Fatalities per terror incident	12	2.39±0.05	2.37	2.40
5. Last name usage	111920	2.44±0.10	2.41	2.50
6. Solar flares	323	2.22±0.03	2.22	1.79
7. City population size	52460	2.34±0.05	2.33	2.37

Summary and conclusions

Here we introduced a new algorithm for estimating power law distribution exponents using Bayesian inference where the distribution may consist of more than one power law. We also derived a Jeffreys prior for power law distributions to be used as a prior in our software. We demonstrate the efficiency of our algorithm and compare it to previously used linear regression and maximum likelihood methods with simulated distributions, and showe that Bayesian inference method is superior to others. We also show the ability of our algorithm to fit mixtures of two power laws with multiple exponent pairs and varying weights of each power law in the mixture. Finally, we identified two power laws of varying exponents in T cell clone size distributions providing a real life example of a power law mixture.

ACKNOWLEDGMENTS. KG was funded by the Ferish-Gerry fellowship from the Watson School of Biological Sciences. GA was funded by the Simons Foundation, Stand Up To Cancer-Breast Cancer Research Foundation Convergence Team Translational Cancer Research Grant, Grant Number SU2C-BCRF 2015-001.

1. Gabaix X (1999) Zipf's law for cities: An explanation. *Q J Econ* 114(3):739–767.
2. Cross CA (1966) The size distribution of lunar craters. *Mon Not R Astron Soc* 134(3):245–252.
3. Newman M (2005) Power laws, pareto distributions and zipf's law. *Contemp Phys* 46(5):323–351.
4. Grigaityte K, et al. (2017) Single-cell sequencing reveals $\alpha\beta$ chain pairing shapes the t cell repertoire. *bioRxiv*.
5. Bolkhovskaya OV, Zorin DY, Ivanchenko MV (2014) Assessing t cell clonal size distribution: a non-parametric approach. *PLoS ONE* 9(9):e108658.
6. Barabasi A, Albert R (1999) Emergence of scaling in random networks. *Science* 286(5439):509–512.
7. Wilson K, Kogut J (1974) The renormalization group and the epsilon-expansion. *Physics Reports* 12:75–199.
8. Ruderman DL, Bialek W (1994) Statistics of natural images: Scaling in the woods. *Physical Review Letters* 73:814.
9. Goldstein ML, Morris SA, Yen GG (2004) Problems with fitting to the power-law distribution. *Eur. Phys. J. B* 41(2):255–258.
10. Clauset A, Shalizi CR, Newman MEJ (2009) Power-law distributions in empirical data. *SIAM Rev.* 51(4):661–703.
11. Albert R, Barabási AL (2002) Statistical mechanics of complex networks. *Rev. Mod. Phys.* 74(1):47–97.
12. Alstott J, Bullmore E, Plenz D (2014) Powerlaw: a python package for analysis of heavy-tailed distributions. *PLoS ONE* 9(1):e85777.
13. Hamra G, MacLehose R, Richardson D (2013) Markov chain monte carlo: an introduction for epidemiologists. *Int J Epidemiol* 42(2):627–634.
14. van Ravenzwaaij D, Cassey P, Brown SD (2018) A simple introduction to markov chain monte-carlo sampling. *Psychon Bull Rev* 25(1):143–154.
15. Gilks W (1995) *Markov chain monte carlo in practice.* (Chapman and Hall/CRC).
16. Gelman A (2009) Bayes, jeffreys, prior distributions and the philosophy of statistics. *Stat Sci* 24(2):176–178.
17. Mattingly HH, Transtrum MK, Abbott MC, Machta BB (2018) Maximizing the information learned from finite data selects a simple model. *PNAS* 115(8):1760–1765.
18. Xiao X, White EP, Hooten MB, Durham SL (2011) On the use of log-transformation vs. nonlinear regression for analyzing biological power laws. *Ecology* 92(10):1887–1894.

1. Supplementary section

A. Jeffrey's Prior. Here we derive a non-informative prior distribution, also known as Jeffreys prior, for the exponent. In one dimension of the parameter space the prior is proportional to the square root of the Fisher Information, I,

$$p(\gamma) \propto \sqrt{I(\gamma)}. \tag*{[11]}$$

The Fisher Information is defined as

$$I(\gamma) = -\mathbb{E}_{\gamma}\left[\frac{d^2 \ln p(x|\gamma)}{d\gamma^2}\right], \tag*{[12]}$$

where the expectation is taken with respect to the conditional distribution $p(x|\gamma)$. Taking the derivatives,

$$\begin{aligned}
\ln p(x|\gamma) &= -\gamma \ln x - \ln \zeta(\gamma) &\qquad [13]\\
\frac{d \ln p(x|\gamma)}{d\gamma} &= -\ln x - \frac{\zeta'}{\zeta} &\qquad [14]\\
\frac{d^2 \ln p(x|\gamma)}{d\gamma^2} &= -\frac{\zeta''}{\zeta} + \left(\frac{\zeta'}{\zeta}\right)^2. &\qquad [15]
\end{aligned}$$

Since every term on the right hand side Eq. (15) is a independent of x, the Fisher Information Eq. (12) is simply evaluated to be

$$I(\gamma) = \frac{\zeta''}{\zeta} - \left(\frac{\zeta'}{\zeta}\right)^2, \tag*{[16]}$$

from which the aforementioned Jeffreys prior result is derived Eq. (5).

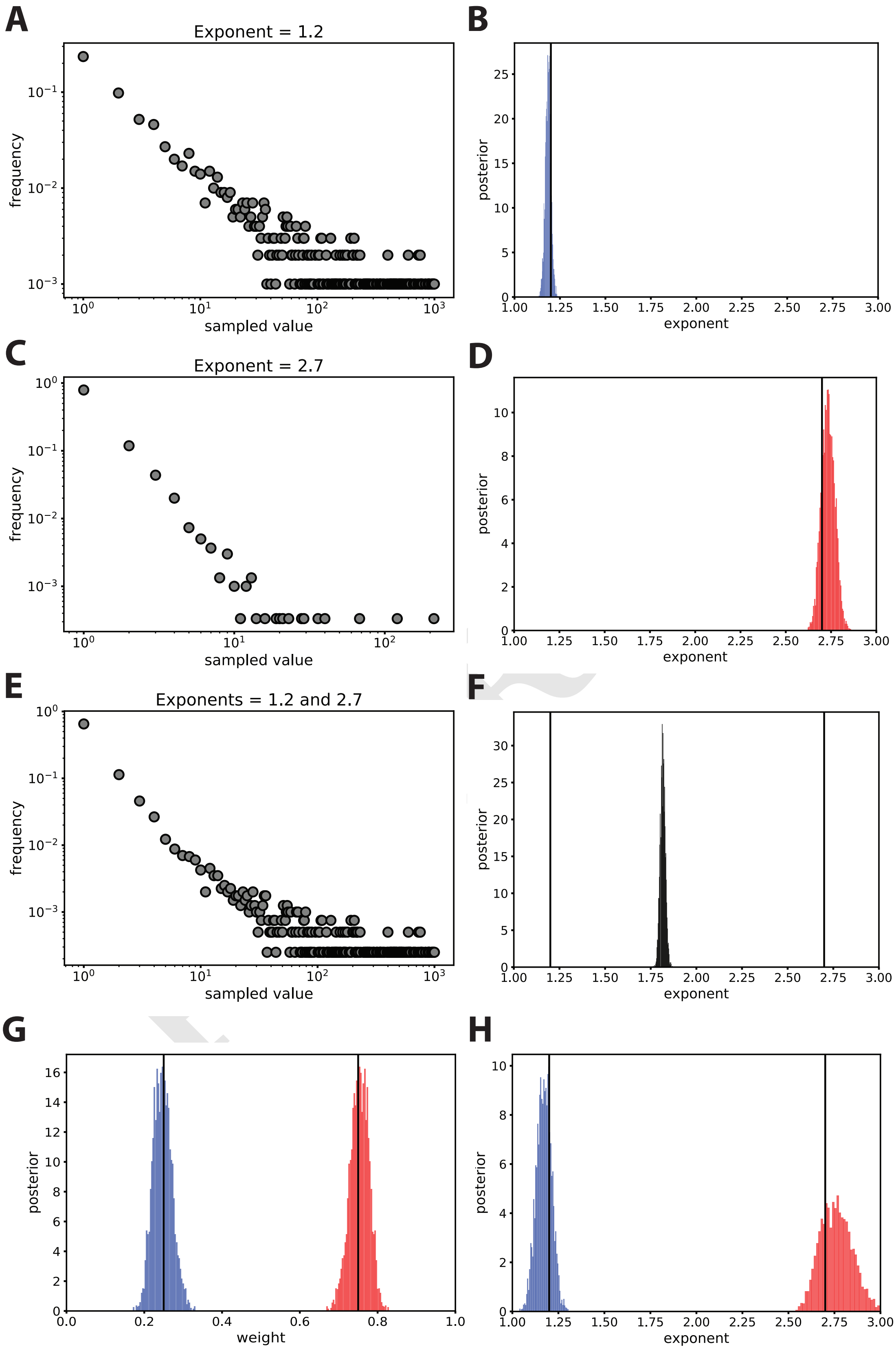

Fig. 2. Demonstration of single and mixed power law distribution fits. A) Simulated power law distribution of exponent 1.2 and sample size of 1000. B) Posterior distribution of power law exponent in (A). C) Simulated power law distribution of exponent 2.7 and sample size of 3000. D) Posterior distribution of power law exponent in (C). E) Mixed power law distribution. 1/4th of resulting power law comes from (A) and 3/4ths from power law (C). F) Posterior distribution of power law exponent in (E) if single power law is assumed. G) Posterior distribution of the two power law weights in the distribution (E). H) Posterior distribution of power law (E) exponents assuming the mix of two. Vertical black lines depict correct answers.

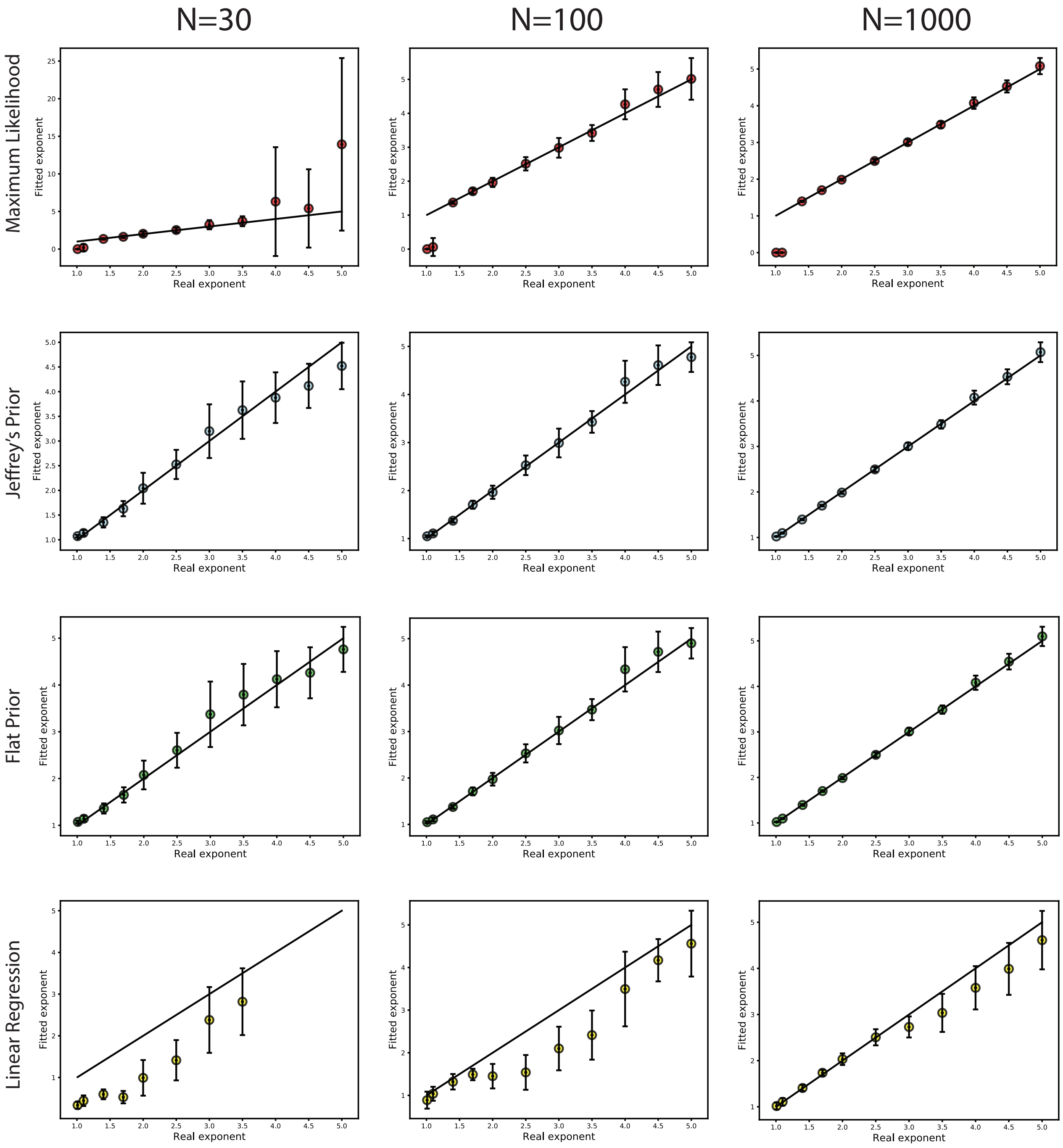

Fig. 3. Comparisons of different power law fitting algorithms. 20 power law distributions were simulated per each exponent of samples sizes 30,100 and 1000. Each power law was fitted using either maximum likelihood, Bayesian inference (Jeffreys and flat prior, mean vs max) and linear regression algorithms. Mean estimated exponent and standard deviation of the 20 power laws in each case is plotted against the correct exponent.

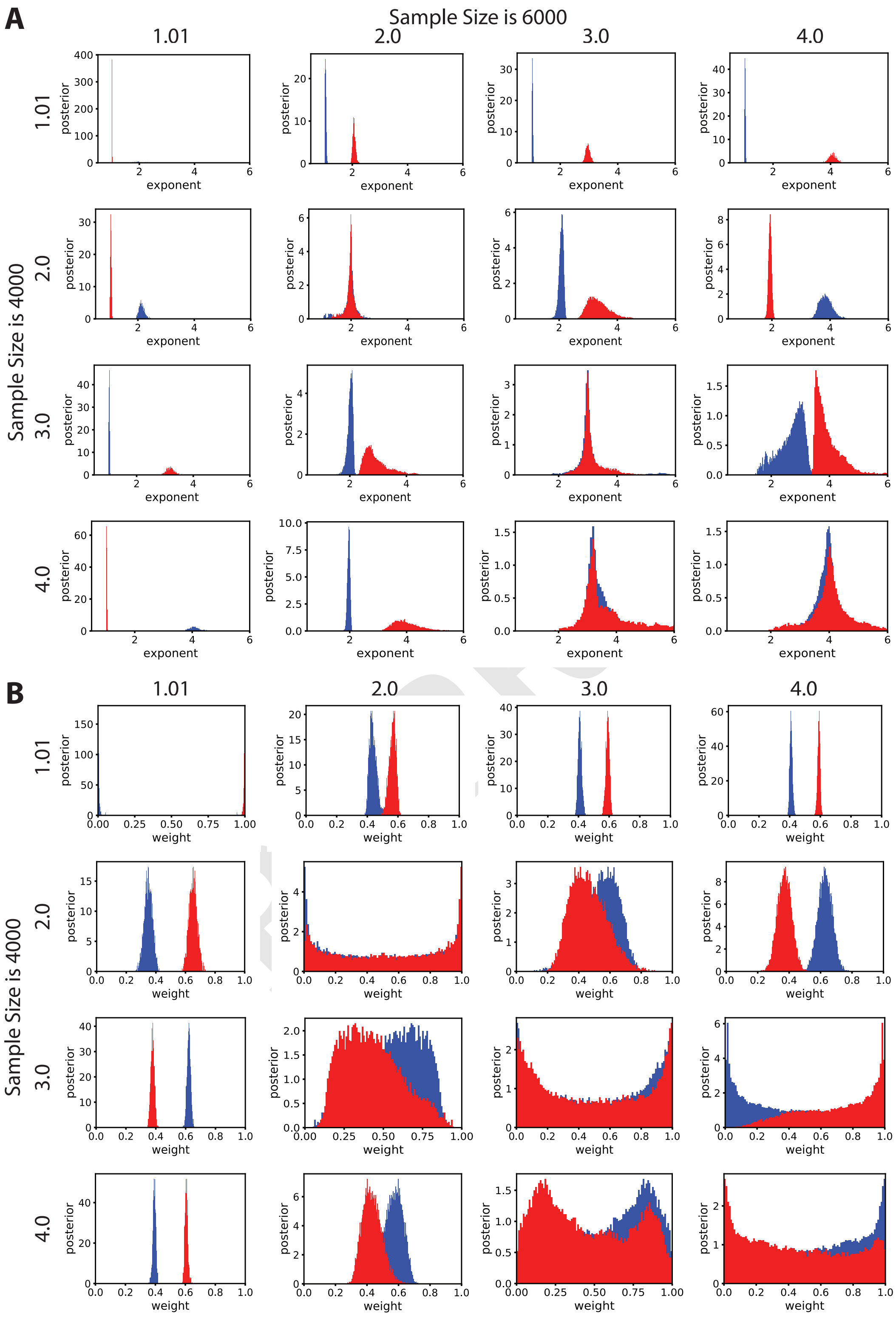

Fig. 4. Demonstration of fitting distributions consisting of two power laws. Mixtures of power law distributions were simulated for each exponent pair of samples sizes 4000 and 6000. The generated distributions were fit using Bayesian inference algorithm to estimate the exponent mixture. A) Posterior distributions of exponents for each power law mixture. B) Posterior distribution of weights for each power law mixture.