Publications of the Astronomical Society of the Pacific, 131:108005 (8pp), 2019 October https://doi.org/10.1088/1538-3873/aae7fc

Collaborative Nested Sampling: Big Data versus Complex Physical Models

Johannes Buchner[1,2,3]
[1] Millenium Institute of Astrophysics, Vicuña. MacKenna 4860, 7820436 Macul, Santiago, Chile; johannes.buchner.acad@gmx.com
[2] Pontificia Universidad Católica de Chile, Instituto de Astrofsica, Casilla 306, Santiago 22, Chile
[3] Excellence Cluster Universe, Boltzmannstr. 2, D-85748, Garching, Germany
Received 2018 May 9; accepted 2018 October 12; published 2019 August 30

Abstract

The data torrent unleashed by current and upcoming astronomical surveys demands scalable analysis methods. Many machine learning approaches scale well, but separating the instrument measurement from the physical effects of interest, dealing with variable errors, and deriving parameter uncertainties is often an afterthought. Classic forward-folding analyses with Markov chain Monte Carlo or nested sampling enable parameter estimation and model comparison, even for complex and slow-to-evaluate physical models. However, these approaches require independent runs for each data set, implying an unfeasible number of model evaluations in the Big Data regime. Here I present a new algorithm, collaborative nested sampling, for deriving parameter probability distributions for each observation. Importantly, the number of physical model evaluations scales sub-linearly with the number of data sets, and no assumptions about homogeneous errors, Gaussianity, the form of the model, or heterogeneity/completeness of the observations need to be made. Collaborative nested sampling has immediate applications in speeding up analyses of large surveys, integral-field-unit observations, and Monte Carlo simulations.

Key words: methods: data analysis – methods: statistical – surveys

Online material: color figures

1. Introduction

Big Data has arrived in astronomy (Feigelson & Babu 2012; Zhang & Zhao 2015; Mickaelian 2016; Kremer et al. 2017). In the previous century it was common to analyse a few dozen objects in detail. For instance, one would use Markov chain Monte Carlo to forward-fold a physical model and constrain its parameters. This would be repeated for each member of the sample. However, current and upcoming instruments provide a wealth of data (some millions of independent sources) where it becomes computationally difficult to follow the same approach, even though it is embarrassingly parallel. Currently, much effort is put into studying and applying machine learning algorithms such as (deep learning) neural networks or random forests for the analysis of massive data sets. This can work well if the measurement errors are homogeneous, but typically these methods make it difficult to insert existing physical knowledge into the analysis, to deal with variable errors and missing data points, and generally to separate the instrument measurement process from the physical effects of interest. Furthermore, we would like to derive probability density distributions of physical parameters for each object, and do model comparison between physical effects/source classes.

In this work I show how nested sampling can be used to analyse N data sets simultaneously (Section 2). The key insight is that nested sampling allows effective sharing of evaluation points across data sets, requiring much fewer model evaluations than if the N data sets were analysed individually. I assume only that the model can be split into two components: a slow-to-evaluate physical model which performs a prediction into observable space, and a fast-to-compute comparison to the individual data sets (e.g., the likelihood of a probability distribution). Otherwise, the user is free to choose arbitrary physical models and likelihoods. Section 3 presents a line fitting of a hypothetical many-object spectroscopic survey as a toy example; Section 4 constrains the properties of stellar populations in a real imaging-spectroscopy observation.

2. Methodology

2.1. Introduction to Classic Nested Sampling

Nested sampling (Skilling 2004) is a global parameter space exploration algorithm, which zooms in from the entire volume towards the best-fit models by steadily increasing the likelihood threshold. In the process it produces parameter posterior probability distributions and computes the integral over the parameter space. Assume that the parameter space is a k-dimensional cube. A number of live points $N_{\rm live}$ are randomly[4]

[4] In general, following the prior. For most problems one can uniformly sample with appropriate stretching of the parameter space under the inverse cumulative of the prior distributions (see Section 5.1 in Feroz et al. 2009).

placed in the parameter space. Their likelihood is evaluated. Each point represents $1/N_{\rm live}$ of the entire volume. The live point with the lowest likelihood $L_{\rm min}$ is then removed, implying the removal of space with likelihood below $L_{\rm min}$ and shrinkage of the volume to $1 - \exp(-1/N_{\rm live})$, on average. A new random live point is drawn, with the requirement that its likelihood must be above $L_{\rm min}$. This replacement procedure is iterated, shrinking the volume exponentially. Each removed ("dead") point and its likelihood L_i is stored. The integral over the parameter space can then be approximated by $Z = \sum_i L_i \times w_i$, where w_i is the removed volume at the iteration. At a late stage in the algorithm the volume probed is tiny and the likelihood L_i increase is negligible, so that the weights $L_i \times w_i$ of the remaining live points become small. Then the iterative procedure can be stopped (the algorithm has converged). The posterior probability distribution of the parameters is approximated as importance samples of weight $L_i \times w_i$ at the dead point locations, and can be resampled into a set of points with equal weights, for posterior analyses similar to those with Markov chains. More details on the convergence and error estimates can be found in Skilling (2009).

Efficient general solutions exist for drawing a new point above a likelihood threshold in low dimensions ($n_{\rm dim} < 20$). The idea is to draw only in the neighbourhood of the current live points, which already fulfil the likelihood threshold. The best-known algorithm in astrophysics and cosmology is MULTINEST (Shaw et al. 2007; Feroz et al. 2009). There, the contours traced out by the points are clustered into ellipses, and new points are drawn from the ellipses. To avoid accidentally cutting away too much of the parameter space, the tightest-fitting ellipses are enlarged by an empirical (problem-specific) factor. Another algorithm is RADFRIENDS (Buchner 2014), which defines the neighbourhood as all points within a radius r of an existing live point. By leaving out randomly a portion of the live points, and determining their distance to the remaining live points, the largest nearest-neighbour radius r is determined. The worst-case analysis through bootstrapping cross-validation over multiple rounds makes RADFRIENDS robust, independent of contour shapes, and free of tuning parameters. Figure 1 illustrates the generated regions. RADFRIENDS is efficient if one chooses a standardized Euclidean metric (i.e., normalize by the standard deviation of the live points along each axis). The extension to nested sampling proposed in this paper works with any constrained drawing method.

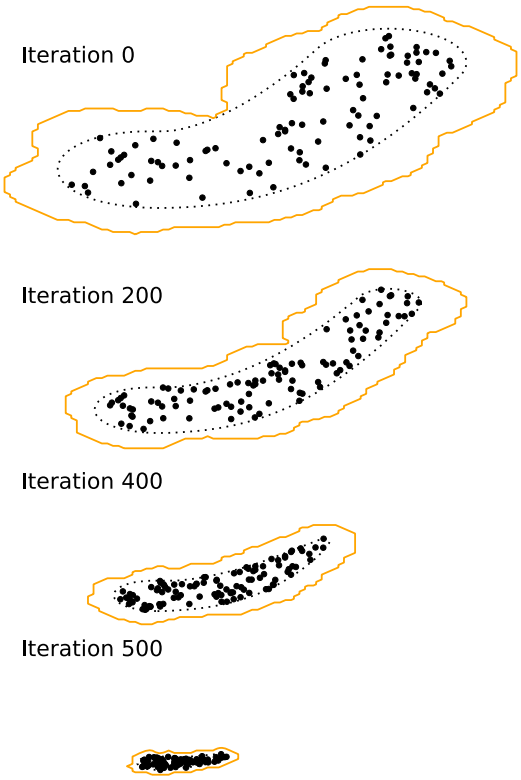

Figure 1. Illustration of nested sampling. At a given iteration of the nested sampling algorithm, the live points (black) trace out the current likelihood constraint, a region (dashed) which is unknown. The RADFRIENDS algorithm conservatively reconstructs the region (orange) by including everything within a certain, adaptively chosen radius of the current live points. Between iterations, the likelihood contour is elevated, making the sampled volume smaller and smaller. MULTINEST works similarly, but clusters point into ellipsoids.
(A color version of this figure is available in the online journal.)

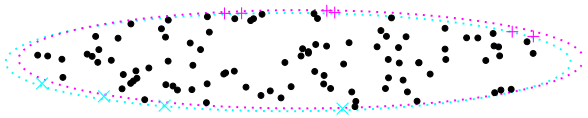

Figure 2. Analysis of two similar data sets yielding at the same iteration similar likelihood contours (the two dotted ellipses). In the presented algorithm a large fraction of live points are shared across data sets (black points), which reduces the number of model evaluations. The differences (cyan crosses and magenta pluses) require additional draws.
(A color version of this figure is available in the online journal.)

2.2. Simplified Description of the Idea

Consider two independent nested sampling runs on different data sets, but initialized to the same random number generator state. Initially points are generated from across the entire parameter space, typically giving bad fits. If the data sets are somewhat similar, the phase of zooming to the relevant parameter space will be the same for the two runs. Importantly, while the exact likelihood value will be different for the same point, the ordering of the points will be similar. In other words, for both, the worst-fitting point to be removed is likely the same. The next key insight is that new points can be drawn efficiently from a contour which is the union of the likelihood contours from both runs. Ideally, the point can be accepted by both runs, keeping the runs similar (black points in Figure 2).

When a point is shared, the (slow) predicting model has to be evaluated only once, speeding up the run. The model prediction is then compared against the data to produce a likelihood for each data set, an operation which I presume to be fast, e.g., when computing

$$\mathcal{L}_j = -\sum_i (x_{ij} - m_i)^2 / (2\sigma_{ij}^2), \quad (1)$$

where m_i, x_{ij} and σ_{ij} are the predictions, measurements and errors in data space respectively for data set j.

What if the point can be accepted by only one run? It cannot simply be rejected or accepted in both, otherwise the uniform sampling property of nested sampling is broken. Instead, accepted points are stored in queues, one for each run/data set. Once both runs have a non-empty queue, the first accepted point is removed from each queue and replaces the dead point of each data set. Joint sampling also helps even if a point is not useful right away. If a point was only accepted by run A, but the following point is accepted by both runs, the second point becomes a live point immediately for run B, but can later also become a live point for run A (if it satisfies the likelihood threshold at that later iteration). This technique allows sustained sharing of points, decreasing the number of unique live points and increasing the speed-up.

At a later point in the algorithm, the contours may significantly diverge and not share any live points. This is because the best-fit parameters of data sets will differ. Then, nested sampling runs can continue as in the classic case, without speed-up, falling back to a linear scaling. The more different the data sets are, the earlier this happens. The run is longer for data sets with high signal-to-noise, making the algorithm most efficient when most observations are near the detection limit. This is typically the case in surveys as a consequence of power-law distributions.

2.3. Collaborative Nested Sampling

I now describe the collaborative nested sampling algorithm. A proof-of-concept reference implementation is available at https://github.com/JohannesBuchner/massivedatans/. The algorithm components are the nested sampling integrator, the constrained sampler, and the likelihood function, as in classic nested sampling, except that works on N data sets simultaneously, with N a large number. The constrained sampler behaves substantially differently in this algorithm.

2.3.1. Likelihood Function

The likelihood function receives a parameter vector, and information on which data sets to consider. It evaluates the physical model with the parameter vector to produce a prediction into data space. The physical model may perform complex and slow numerical computations/simulations at this point. Finally the prediction is compared with the individual data sets to produce a likelihood for each considered data set. The likelihood at this point can be Gaussian (Equation (1)), Poissonian, a red noise process, or any other probability distribution appropriate for the instrument. In any case, this computation must be fast compared with producing the model predictions to receive any performance gains.

2.3.2. Nested Sampling Integrator

The integrator deals with each run individually just as in standard nested sampling. It keeps track of the remaining volume at the current iteration, and stores the live points and their weights for each data set individually. It calls the constrained sampler (see below), which holds the live points, to receive the next dead point (for all data sets simultaneously). The integrator must also test for convergence, and advance further only those runs that have not yet converged. Here I use the standard criterion that the nested sampling error is $\delta Z < 0.5$ (from the final equation in Skilling 2009). Once all runs have terminated, corresponding to each data set the integral estimates Z and posterior samples are returned, giving the user the same output as, e.g., a MULTINEST analysis.

2.3.3. Constrained Sampler

The sampler initially draws $N_{\rm live}$ live points and stores their likelihoods in an array of size $N \times N_{\rm live}$. Sequential IDs are assigned to live points and the mapping between live point IDs and data sets ($N \times N_{\rm live}$ indices) is stored. The integrator informs the sampler when it should remove the lowest likelihood point and replace it. The integrator also informs the sampler when some data sets have finished and can be discarded from further consideration, in which case the sampler works as if they had never participated.

The main task of the constrained sampler is to do joint draws under the likelihood constraint $L > L_{\rm min}$ to replace the lowest likelihood point in each of the d data sets. For this, d initially empty queues are introduced (see Figure 3). First, it attempts to draw from the joint contour over all data sets (*superset draw*), i.e., letting RADFRIENDS define a region based on the all unique live points. From this region a point is drawn which has $L > L_{\rm min}$ for at least one data set. The point will be accepted for some data sets, and the corresponding queues are filled. If this fails to fill all queues after several (e.g., 10) attempts, a *focused draw* is done. In that case, only the data sets with empty queues are considered, the region is constructed from their live points, and the likelihood only evaluated for these data sets. For example, in Figure 3, only data set 3 would be considered. Once all queues have at least one entry, nested sampling can proceed: for each data set, the first queue entry is removed and replaces the dead live point. In Figure 3 this is illustrated by the queues pushing out the lowest live points. These dead points are returned to the integrator.

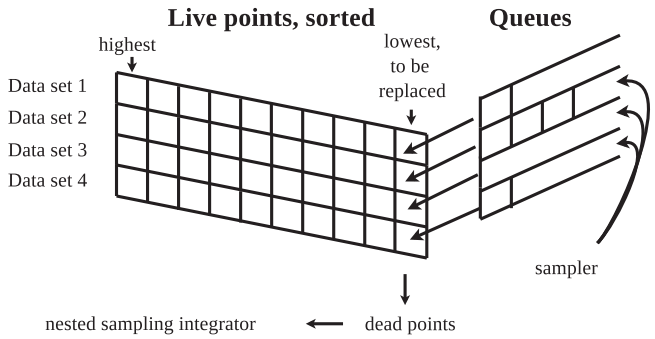

Figure 3. To replace the least likely live point, new points are sampled and placed in queues if they have a high enough likelihood. Once every data set has a non-empty queue, the lowest points are pushed out and stored as dead points by the integrator. In this illustration, $N = 4$ data sets are sampled with $N_{\rm live} = 10$ live points.

Storing queue entries is only useful if they can replace live points in future nested sampling iterations. To be accepted into the queue at position j, the live points' likelihood must be higher than that of j points from the run's live points and existing entries of the queue. In other words, the first entry must merely beat a single existing live point, while the second entry must beat both a live point and either another live point or the first queue entry (which will become a live point in the next iteration).

2.3.4. Data Set Clustering

It can occur that between two groups of data sets the live points are no longer shared, i.e., the live point sets are disjoint (see Figure 4). For example, one may have a dichotomy between broad- and narrow-line objects, and the contours identify some of the data sets in the former class, some in the latter. Distinct groups caused by diverging likelihood contours are an interesting aspect of the exploration: they define a data set similarity through the constraints in parameter space, based on the likelihood ordering unique to nested sampling. This is different to clustering data sets in data space, which can be non-trivial for varying errors and completeness, and clustering in parameter space could scale poorly with model dimensionality. In practice, diverging live points groups can be identified by finding connected subsets in a graph. As illustrated in Figure 4, the necessary graph can be constructed with nodes corresponding to the data sets, nodes corresponding to the live points, and connecting the graph according to the current live point statuses. Algorithms for identifying connected subsets of graphs are well-known. These dataset groups can be processed independently, avoiding multi-modal contours. In the numerical examples shown in this work, this however does not yield substantial speed-ups.

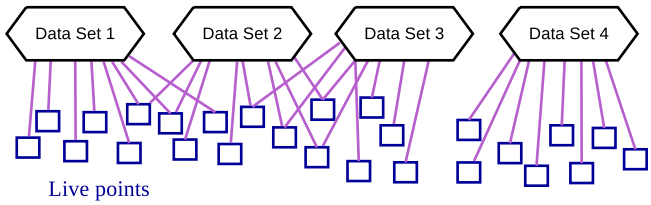

Figure 4. Association of live point objects with data sets. In this illustration, some live points are shared between the group of data sets 1, 2, and 3; these form a connected subgraph. Data set 4 has separate live points and can be treated independently. In this illustration, $N_{\rm live} = 8$ and $N = 4$, but there are only 26 unique live points.
(A color version of this figure is available in the online journal.)

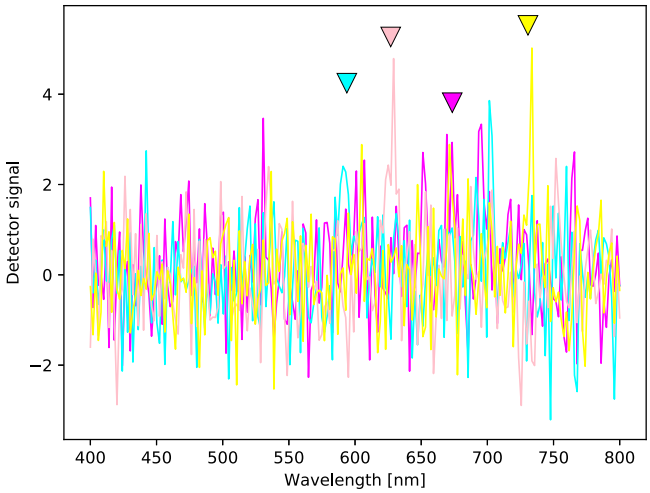

Figure 5. Simulated noisy data. The location, width, and amplitude of a single line are sought in Gaussian noise for the illustrative problem. The true line locations of the four spectra are indicated by triangles. The cyan data set shows a random fluctuation at 700 nm.
(A color version of this figure is available in the online journal.)

3. Toy Application: Single-line Fitting

A simple toy example problem illustrates the use and scaling of the algorithm. Let us consider a spectroscopic survey which collected N spectra in the 400–800 nm wavelength range. We look for a Gaussian line at 654 nm (in rest frame, randomly redshifted) with standard deviation of 0.5 nm. The amplitudes vary with a power-law distribution with index 3, with a signal-to-noise ratio of at least two. I generate a large random data set and analyse the first N data sets simultaneously to understand the scaling of the algorithm, with $N = 1$ to $N = 10^4$. Figure 5 presents some high and low signal-to-noise examples of the simulated data set.

The parameter space of the analysis has three dimensions: the amplitude, width, and location of a single Gaussian line, with log-uniform/log-uniform/uniform priors from 10^{0-2}, 0.15–15 nm, and 600–1000 nm respectively. The Gaussian line

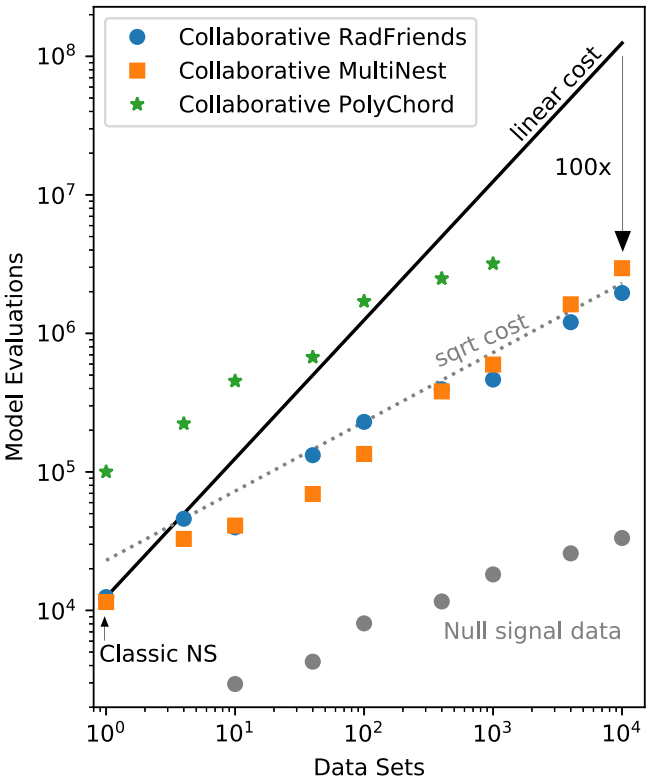

Figure 6. Number of model evaluations of collaborative nested sampling applied to RADFRIENDS, multi-ellipsoidal sampling (MULTINEST) and whitened slice sampling (POLYCHORD). A naive approach of independent nested sampling analyses would have a linear scaling (black line). The algorithm scales substantially better, similar to $O(\sqrt{N})$ in the considered problem, giving a 100× speed-up when analysing 10,000 data sets. Analysing Monte Carlo simulated data without signal (gray points) is also faster.
(A color version of this figure is available in the online journal.)

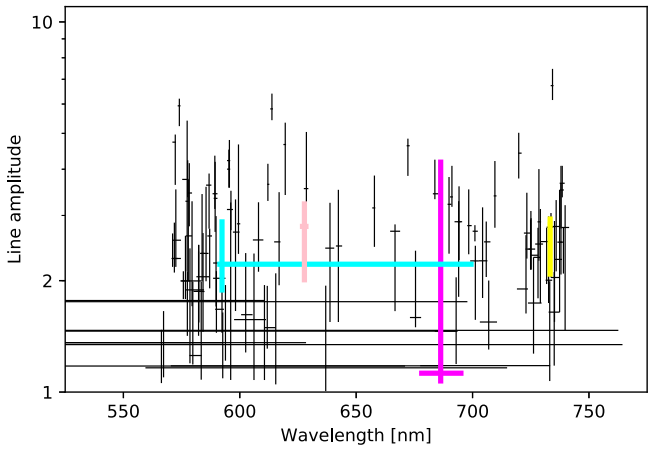

Figure 7. Parameter posterior constraints. Each error bar shows a simulated data set; the four examples from Figure 5 are shown in the same colors. The pink and yellow data sets have been well-detected and characterized, while the magenta line has larger uncertainties. The cyan constraints cover two solutions (see Figure 5). Error bars are centred at the median of the posteriors with the line lengths reflecting the 1-sigma equivalent quantiles.
(A color version of this figure is available in the online journal.)

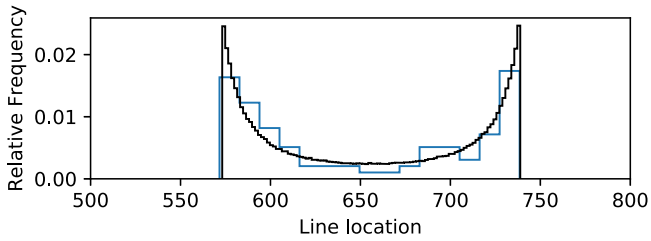

Figure 8. Line location distribution for objects where the line was well-constrained (blue) compared to the input distribution (black).
(A color version of this figure is available in the online journal.)

is our "slow-to-compute" physical model. The likelihood function is as in Equation (1). A more elaborate example would include physical modelling of an ionized outflow emitting multiple lines with Doppler broadening and red detector noise, without necessitating any modification of the presented algorithm.

Figure 6 shows the number of model evaluations necessary for analysing N data sets. We implemented our nested sampling variant on top of three constrained drawing methods, RADFRIENDS (Buchner 2014), multi-ellipsoidal sampling (MULTINEST; Shaw et al. 2007; Feroz et al. 2009) and eigenvector slice sampling (POLYCHORD; Handley et al. 2015, here for simplicity implemented without clustering). The black line shows the baseline linear scaling $O(N)$, i.e., analysing the data sets one by one. The algorithm scales much better, close to $O(\sqrt{N})$. For instance, it takes only 100 times more model evaluations to analyse 10,000 observations than a single observations, a 100× speedup.

We can now plot the posterior distributions of the found line locations. Figure 7 demonstrates the wide variety of uncertainties. The spectra of Figure 5 are shown in the same colors. For many, the line could be identified and characterized with small uncertainties (yellow, pink, black); for others, the method remains uncertain (cyan, magenta). Figure 8 shows that the input redshift distribution is correctly recovered.

After parameter estimation we can consider model comparison: is the line significantly detected? For this, let us consider the Bayes factor, $B = Z_1/Z_0$, where Z_1 is the integral computed by nested sampling under the single-line model, and Z_0 is the same for the null hypothesis (no line). The latter can be analytically computed as $\ln Z_0 = -\frac{1}{2}[\sum(x_i/\sigma_i)^2 + \ln 2\pi\sigma_i^2]$. Figure 9 shows in black the derived Bayes factors. To define a lower threshold for significant detections, I Monte Carlo simulate a data set with $N = 10^4$ spectra without signal, and derive Z_1 values. This can be done rapidly with the presented

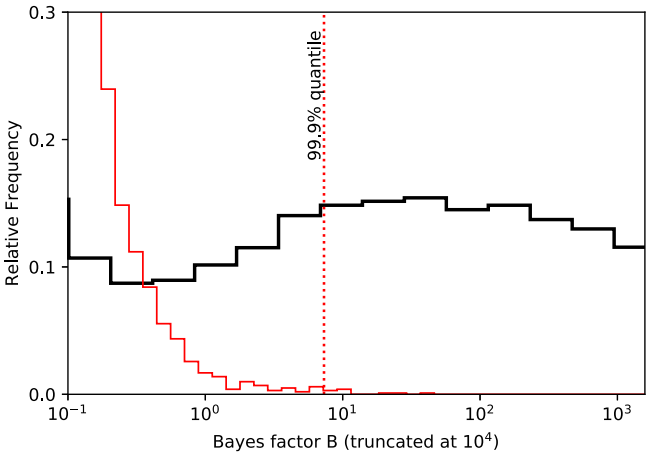

Figure 9. Bayes factors between the single-line model and a no-line model. The black histogram shows Bayes factors from analysing the test data set. The red histogram shows Bayes factors from noise-only data. Because the latter has very few values above $B \gtrapprox 10$, a line can be claimed detected beyond that threshold with a low false positive fraction.
(A color version of this figure is available in the online journal.)

algorithm. The red histogram in Figure 9 shows the resulting Bayes factors. The 99.9% quantile of B-values in this signal-free data set is $B \approx 10$. Therefore, in the "real" data, those with a Bayes factor $B > 10$ can be securely claimed to have a line, with a small fraction of false detections ($p < 0.001$).

4. Application to Imaging Spectroscopy

Finally, I apply collaborative nested sampling to a real-world data set. Integral-field unit observations, where many spectra are taken in proximity on the sky, are ideal for applying the algorithm. The MUSE spectrograph (1 arcmin^2 field of view, wavelength range 480–930 nm; Bacon et al. 2010) observed the Abell 370 galaxy cluster in November 2014 (PI: Richard) for one hour.[5] Following, e.g., Lagattuta et al. (2017) and Patrício et al. (2018), standard data reduction procedures and sky line subtraction (Soto et al. 2016) were used, and the errors increased by 20% of the data value to account for model inaccuracies. The chosen region in the sky (A370-sys1 in Patrício et al. 2018) includes several galaxies, some of which are heavily distorted by strong lensing. Its white image (sum across the spectrum) is shown in the top left panel of Figure 10. The 169 arcsec^2 region is covered by 4223 fibers, each of which provides a spectrum with measured intensity and error.

A simple stellar population is used to model the spectra. The classic Bruzual & Charlot (2003) model stellar spectra are weighted by an exponential star formation (decay timescale τ) at a time t Gyr in the past. Additionally, dust extinction is allowed through a Calzetti law (Calzetti et al. 2000) and the model is redshifted. To avoid a degeneracy with star formation age, I assume solar metallicity. This model thus has four parameters: redshift z (0–1), star formation age t (0–13 Gyr), star formation decay time τ (10^{6-9} Gyr), and extinction $E(B-V)$ (0–1). Uniform priors are applied on z, t, $\log\tau$, and $E(B-V)$.

A Gaussian likelihood compares the model spectrum M_i against the measurements μ_i and errors σ_i. To avoid having the model normalization s as a fitting parameter, it is marginalized over, by setting

$$s = \sum_i (\mu_i M_i \sigma_i^{-2}) / \sum_i (M_i^2 \sigma_i^{-2}) \quad (2)$$

and neglecting constants in the likelihood (see Arnouts et al. 1999 for more details):

$$\log L = -\frac{1}{2} \sum [(\mu_i - s \cdot M_i)/\sigma_i]^2 . \quad (3)$$

With the model, likelihood, and data defined, I apply collaborative nested sampling with multi-ellipsoidal sampling and derive the posterior parameter distribution at each spaxel. Evidence values are also obtained. Similar to the previous section, Bayes factors are computed and shown in the top right panel in Figure 10 at each spaxel. The red areas indicate where the data prefer no input signal over the stellar model.

The second row of Figure 10 shows the derived redshift. Uncertainties (right panel) are extremely small (typically 0.001) over most of the image. Two solutions are visible in the left panel: the arc (green) is at distinctly higher redshifts than the blobs (yellow). Extended emission at the same redshift as the blobs is detected.

The third and fourth rows present the star formation properties in each pixel. The arc shows evidence for recent star formation. While uncertainties on the decay parameter are not small, generally the arc has a longer (10^8 Gyr) star formation episode than the blobs (10^7 Gyr). The extinction is constant over image and shows small values (<0.1; not shown). The model used here is overly simple and I do not interpret the physical meaning in great detail (see Patrício et al. 2018 instead). In particular, the relation between metallicity and age should be explored further. Patrício et al. (2018), however, derived similar values, e.g., for the star formation age and timescale when considering the stacked spectrum across the entire arc.

This application demonstrates the usefulness of collaborative nested sampling in a realistic data set. Physical parameters were extracted while exploiting that spatial neighbours have similar physical properties. However, no assumption about smoothness or neighbourhood was made, i.e., the constraints at each pixel are independent. Also note that, in every pixel, a posterior distribution over the parameters is derived. Here, the collaborative nested sampling analysis of 4223 fibers required 14.4 million likelihood evaluations (140 hr). This corresponds to a quadrupling of the efficiency compared to analysing only 100 fibers, which required 2.8 million likelihood evaluations (14.9 hr).

[5] Additional observations have been made since then; however, here the demonstration is intended to show information extraction in the low-signal regime.

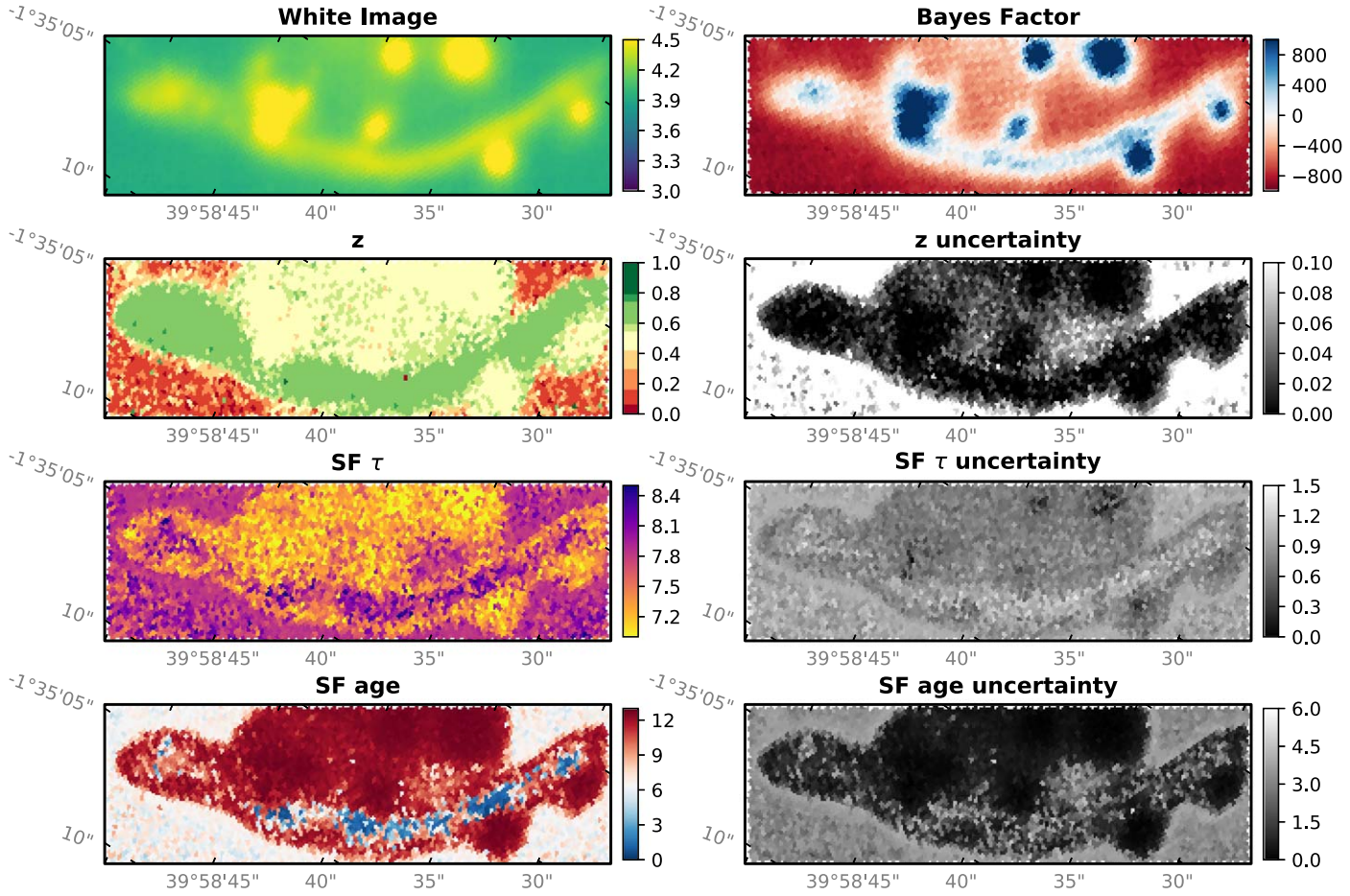

Figure 10. MUSE integral-field unit observation data analysed using collaborative nested sampling. Top left panel: white image of the input data. Several blobs and an extended arc is visible. Top right panel: Bayes factor B comparing the single-stellar population model to a no-signal model. In the arc and blobs, $\log B > 0$. The remaining panels present posterior parameters (left) and uncertainties (right). Redshift z is well-constrained across the arc ($z \approx 0.6$) and the closer blobs ($z \approx 0.4$). While the blobs had a brief star formation episode ($\log \tau/\mathrm{yr} \approx 7$) long ago (age $> 10^{10}$ yr), the arc shows evidence for recent star formation (younger age, slower decay τ).

(A color version of this figure is available in the online journal.)

5. Discussion

Collaborative nested sampling is a scalable algorithm suitable for analysing massive data sets with arbitrarily complex physical models and complex, inhomogeneous noise properties. The algorithm brings to the Big Data regime parameter estimation with uncertainties, classification of objects, and distinction between physical processes.

The key insight in this work is to take advantage of a property specific to nested sampling: The sampling regions can look similar across similar data sets, and rejection sampling from the union of contours is permitted.[6] Collaborative nested sampling reduces the number of unique model evaluations, in particular at the beginning of the nested sampling run. The same approach cannot be followed with Markov chain proposals; there, the proposal depends on the current point, and deviating acceptances prohibit a later joint proposal.

Compared to embarrassingly parallel analyses, collaborative nested sampling excels in specific types of problems, which have many uncertain data sets of similar structure into which a slow physical model is predicting. The algorithm has some overhead related to the management of live points, particularly in determining the unique set of live points across a dynamically selected subgroup of data sets. The memory usage also grows if large data sets have to be held in the same machine. If only chunks of N are managable, the analyses can be split into such sizes and analysed in parallel across multiple machines. In that case, one can take advantage of the scaling of the algorithm until N.

[6] As in classic nested sampling, the volume shrinkage estimates are valid on average. Multiple runs can test whether this leads to additional scatter in the integral estimate. In practice, single runs already give correct uncertainties for many problems.

The algorithm can be applied immediately to any existing large data sets. Compared to other Big Data analysis approaches, nested sampling supports model comparison and yields full probability distributions on arbitrary models, allowing the exploration of degenerate solutions. Furthermore, the instrument response can be modelled out and separated from the process of interest. To give one application example, *eROSITA* (Predehl et al. 2014) requires the source classification and characterization of three million point sources in its all-sky X-ray survey (Kolodzig et al. 2013). The position-dependent detector response and non-Gaussianity of count data make standard machine learning approaches difficult to apply.

Even in the analysis of single objects the presented algorithm can help. One might test the correctness of selecting a more complex model, e.g., based on Bayes factors, as in the toy example presented. Large Monte Carlo simulations of a null hypothesis model can be quickly analyzed with the presented method, with a model evaluation cost that is essentially independent of the number of generated data sets. Going further, approaches to validate models and Bayesian inference (e.g., Talts et al. 2018) over the entire parameter space can be speed up.

I thank Surangkhana Rukdee and Frederik Beaujean for reading the manuscript, and Franz E. Bauer for help with MUSE data. This work made use of the NESTLE free software implementation[7] of the MULTINEST algorithm and the matplotlib plotting library (Hunter 2007). I thank the two anonymous referees for helpful comments and suggestions.

I acknowledge support from the CONICYT-Chile grants Basal-CATA PFB-06/2007, FONDECYT Postdoctorados 3160439 and the Ministry of Economy, Development, and Tourism's Millennium Science Initiative through grant IC120009, awarded to The Millennium Institute of Astrophysics, MAS. This research was supported by the DFG cluster of excellence "Origin and Structure of the Universe".

ORCID iDs

Johannes Buchner https://orcid.org/0000-0003-0426-6634

References

Arnouts, S., Cristiani, S., Moscardini, L., et al. 1999, MNRAS, 310, 540
Bacon, R., Accardo, M., Adjali, L., et al. 2010, Proc. SPIE, 7735, 773508
Bruzual, G., & Charlot, S. 2003, MNRAS, 344, 1000
Buchner, J. 2014, Stat. Comput., 26, 383
Calzetti, D., Armus, L., Bohlin, R. C., et al. 2000, ApJL, 533, 682
Feigelson, E. D., & Babu, G. J. 2012, Modern Statistical Methods for Astronomy (Cambridge: Cambridge Univ. Press)
Feroz, F., Hobson, M. P., & Bridges, M. 2009, MNRAS, 398, 1601
Handley, W. J., Hobson, M. P., & Lasenby, A. N. 2015, MNRAS, 453, 4384
Hunter, J. D. 2007, CSE, 9, 90
Kolodzig, A., Gilfanov, M., Sunyaev, R., Sazonov, S., & Brusa, M. 2013, A&A, 558, A89
Kremer, J., Stensbo-Smidt, K., Gieseke, F., Steenstrup Pedersen, K., & Igel, C. 2017, IISys, 32, 16
Lagattuta, D. J., Richard, J., Clément, B., et al. 2017, MNRAS, 469, 3946
Mickaelian, A. M. 2016, BaltA, 25, 75
Patrício, V., Richard, J., Carton, D., et al. 2018, MNRAS, 477, 18
Predehl, P., Andritschke, R., Becker, W., et al. 2014, Proc. SPIE, 9144, 91441T
Shaw, J. R., Bridges, M., & Hobson, M. P. 2007, MNRAS, 378, 1365
Skilling, J. 2004, in AIP Conf. Proc. 735, Bayesian Inference and Maximum Entropy Methods in Science and Engineering: 24th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, ed. R. Fischer et al. (New York: AIP), 395
Skilling, J. 2009, J. Nested sampling's convergence. In: BAYESIAN INFERENCE AND MAXIMUM ENTROPY METHODS IN SCIENCE AND ENGINEERING: The 29th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, vol 1193 (AIP Publishing), 277, http://scitation.aip.org/content/aip/proceeding/aipcp/10.1063/1.3275625
Soto, K. T., Lilly, S. J., Bacon, R., Richard, J., & Conseil, S. 2016, MNRAS, 458, 3210
Talts, S., Betancourt, M., Simpson, D., Vehtari, A., & Gelman, A. 2018, arXiv:1804.06788
Zhang, Y., & Zhao, Y. 2015, DatSJ, 14, 11

[7] https://github.com/kbarbary/nestle/