Volume Title
*ASP Conference Series, Vol. **Volume Number***
Author

pyblocxs: Bayesian Low-Counts X-ray Spectral Analysis in Sherpa

Aneta Siemiginowska[1], Vinay Kashyap[1], Brian Refsdal[1], David Van Dyk[2], Alanna Connors[3], Taeyoung Park[4]

[1] *Smithsonian Astrophysical Observatory, Cambridge, MA 02138*

[2]*University of California, Irvine, CA 92697*

[3]*Eureka Scientific, Oakland, CA 94602*

[4]*Yonsei University, Seul, South Korea*

Abstract. Typical X-ray spectra have low counts and should be modeled using the Poisson distribution. However, χ^2 statistic is often applied as an alternative and the data are assumed to follow the Gaussian distribution. A variety of weights to the statistic or a binning of the data is performed to overcome the low counts issues. However, such modifications introduce biases or/and a loss of information. Standard modeling packages such as XSPEC and *Sherpa* provide the Poisson likelihood and allow computation of rudimentary MCMC chains, but so far do not allow for setting a full Bayesian model. We have implemented a sophisticated Bayesian MCMC-based algorithm to carry out spectral fitting of low counts sources in the *Sherpa* environment. The code is a Python extension to *Sherpa* and allows to fit a predefined *Sherpa* model to high-energy X-ray spectral data and other generic data. We present the algorithm and discuss several issues related to the implementation, including flexible definition of priors and allowing for variations in the calibration information.

1. Introduction

Standard spectral modeling packages provide a library of physical models, sets of statistics and optimization methods to fit spectral data (e.g. *Sherpa*, XSPEC or ISIS). Two classes of statistics are available: (1) Many flavors of χ^2 statistics with different weights to allow for fitting low counts X-ray spectra. However, even these statistics can lead to biased results when applied to the non-Gaussian X-ray data (see Arnaud et al. 2011); (2) Poisson based likelihood statistics provide unbiased results, e.g. *cash* (derived by Cash (1979)) or *C*, i.e. a slightly modified form of *cash*. In this case the background and source data have to be modeled simultaneously which is not trivial and there is no simple goodness-of-fit test so often the various modifications of χ^2 have been used.

Poisson likelihood methods appropriate for low counts data require techniques for checking model selections and assessing "goodness-of-fit" that involve sampling from the posterior probability distribution. Available software packages contain the Poisson likelihood and standard optimization methods. However, there is no generally available software to probe the posterior probability and check the applied models using the Bayesian methods which include prior. Markov Chain Monte Carlo (MCMC) methods explore the posterior probability in Bayesian analysis. They are more reliable than the

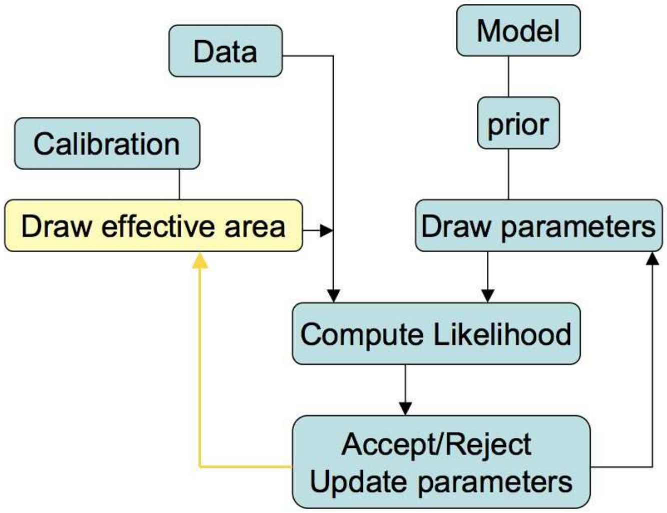

Figure 1. Data flow diagram for the PyBLoCXS algorithm: Draw parameters from a ”proposal distribution”, calculate likelihood and posterior probability of the ”proposed” parameter value given the observed data, use a Metropolis-Hastings criterion to accept or reject the ”proposed” values. The step “draw effective area” to account for calibration uncertainties in the simulations is marked in yellow.

standard downhill optimization algorithms which can get stuck in local minima and are highly sensitive to stopping rules, especially for complex likelihood surfaces. The MCMC provides the full view of the posterior, and gives a direct way to calculate parameter uncertainties and p-values (and ppp-values).

We have developed a Bayesian model for exploring the posterior probability (van Dyk et al. 2001). The method has been implemented in a Python based package `pyblocxs` which can be used in *Sherpa* modeling and fitting application.

2. PyBLoCXS

PyBLoCXS is a sophisticated MCMC based algorithm designed to carry out Bayesian Low-Count X-ray Spectral (BLoCXS) analysis in the *Sherpa* environment. The code is a Python extension to *Sherpa* that explores parameter space at a suspected minimum using a predefined *Sherpa* model. It includes a flexible definition of priors and allows for variations in the calibration information. It can be used to compute posterior predictive p-values for the likelihood ratio test (see Protassov et al. 2002).

`pyblocxs` is based on the methods described in van Dyk et al. (2001) but employs a different MCMC sampler than the one presented in that article. In particular, `pyblocxs` has two sampling algorithms. The first uses a Metropolis-Hastings jumping rule that is a multivariate t-distribution with user specified degrees of freedom centered on the best spectral fit and with multivariate scale determined by the *Sherpa* function, `covar()`, applied to the best fit. The second algorithm mixes this Metropolis-Hastings jumping rule with a Metropolis jumping rule centered at the current draw, also sampling according to a t-distribution with user specified degrees of freedom and a multivariate scale determined by a specified scalar multiple of `covar()` applied to the best fit.

A general description of the MCMC techniques we employ along with their convergence diagnostics can be found in Appendices A.2 - A 4 of van Dyk et al. (2001) and in more detail in Chapter 11 of Gelman et al. (2004)

3. Applications

`pyblocxs` is a generally available code. It can be used to perform several important statistical tasks:

- Explores parameter space and summarizes the full posterior or profile posterior distributions.
- Computes parameter uncertainties that can include calibration errors.
- Simulates data from the posterior predictive distributions.
- Tests for added spectral components by computing the Likelihood Ratio Statistic on replicate data and the ppp-value (posterior-predictive-p-values).

3.1. Calibration Uncertainties

Instrument calibration measurements such as an effective area of a telescope have known uncertainties. These uncertainties are often non-linear and cannot be simply added to the statistical uncertainties. A standard approach is to just ignore these uncertainties, mainly because there have been no methods to account for them in the analysis software. However, these uncertainties are important as they limit the final parameters constraints given by the observations. Also their impact is more significant in the high signal to noise spectra (see Drake et al. 2006; Kashyap et al. 2008; Lee et al. 2011).

PyBLoCXS MCMC methods can take into account calibration uncertainty, by including an additional “update calibration” step in the MCMC loop (e.g. ’draw effective area’ step in Fig. 1). The new calibration data (e.g.effective area) is drawn just before each computation of the likelihood and is used in the model evaluation and the final acceptance of the parameters in the loop. Lee et al. (2011) discuss the model that includes the calibration uncertainties and was applied to Chandra spectra. They also compare several methods to account for these uncertainties and discuss some implications on the overall data analysis. Figure 2 shows an impact of the calibration errors on the uncertainties. The departure between the statistical and total errors is larger for the data with the highest signal to noise indicating the limit in the constraints that can be put on model parameters, e.g. we cannot improve our knowledge about these sources with a larger number of counts.

4. Summary

`pyblocxs` is used to analyze astronomical counts data. It provides the MCMC simulations to explore parameter space of models applied to Poisson data. It requires *Sherpa* and was only tested on applications to simple one component models, while a parameter space can be complex for composite models. It is available as a *Sherpa* Python extension at `http://hea-www.harvard.edu/AstroStat/pyBLoCXS/index.html`

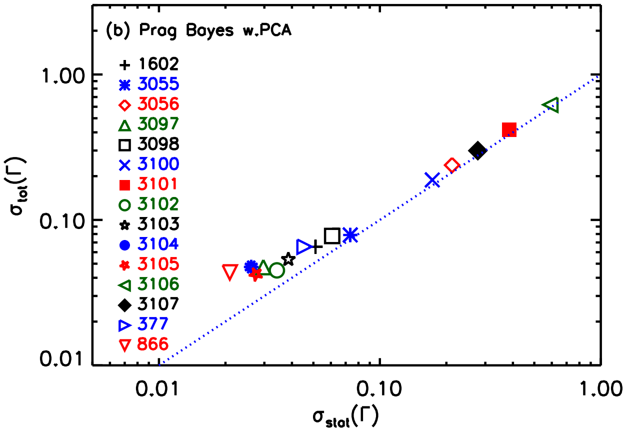

Figure 2. Comparison of the statistical and total error which accounts for calibration uncertainties. Points are results of a `pyblocxs` fit to the data including the calibration step in Fig.1 The dotted line represents equality between the statistical(x-axis) and total (y-axis) errors (Lee et al. 2011). Note that for high counts sources where the effect of calibration uncertainty is most prominent, the overall error reaches a minimum even as the statistical error continues to decrease with increasing data quality.

Acknowledgments. This project was supported by CXC NASA contract NAS8-39073 and NASA AISRP grant NNG06GF17G.

References

Arnaud, K., Smith, R., & Siemiginowska, A. 2011, Handbook of X-ray Astronomy (Cambridge: Cambridge University Press), 1st ed.

Cash, W. 1979, ApJ, 228, 939

Drake, J. J., Ratzlaff, P., Kashyap, V., Edgar, R., Izem, R., Jerius, D., Siemiginowska, A., & Vikhlinin, A. 2006, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 6270 of Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. 2004, Bayesian Data Analysis (London: Chapman & Hall), 2nd ed.

Kashyap, V. L., Lee, H., Siemiginowska, A., McDowell, J., Rots, A., Drake, J., Ratzlaff, P., Zezas, A., Izem, R., Connors, A., van Dyk, D., & Park, T. 2008, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 7016 of Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference

Lee, H., Kashyap, V., van Dyk, D., Connors, A., Drake, J., Izem, R., Meng, X.-L., Min, S., Park, T., Ratzlaff, P., Siemiginowska, A., & Zezas, A. 2011, ApJ, submitted

Protassov, R., van Dyk, D. A., Connors, A., Kashyap, V. L., & Siemiginowska, A. 2002, ApJ, 571, 545

van Dyk, D. A., Connors, A., Kashyap, V. L., & Siemiginowska, A. 2001, ApJ, 548, 224