# MCMC Samples

MLE applies a Bayesian algorithm to sample the posterior probability of the 2-D source model parameters. The posterior samples (draws) are used to calculate error ellipses for the source positions.

MLE applies Sherpa implementation of pyBLoCXS (van Dyk et al. 2001), Markov chain Monte Carlo (MCMC) based algorithm, which samples the posterior probability distribution of the parameters starting from the maximum likelihood step in the MLE optimization. The Sherpa function *get_draws()* is used to run MCMC chains using the 2-D model information for a specific dataset, the selected sampler, the defined priors, and the specified number of iterations. This function returns an array of statistic values, an array of acceptance Booleans, and an array of sampled parameter values, i.e. *draws*.

## Sampler Selection

The multivariate t-distribution is the sampling distribution for the MCMC. This distribution is defined by the multivariate normal (for the model parameter values and the covariance matrix), and chi2 distribution for a given degrees of freedom. The algorithm uses a mixture jumping rule of Metropolis (symmetric) and Metropolis-Hastings (asymmetric) for selection of the proposed parameters. MLE assumes 0.5 probability for the jumps from the best-fit (Metropolis rule) or from the current sampled (Metropolis-Hastings) parameter values (see Gelman et al. 2013).

The algorithm uses multivariate normal distribution which requires the parameter values and the corresponding covariance matrix. MLE assumes the diagonal covariance matrix based on Sherpa *int_unc()* function.

Additional scale parameter allows to adjust the scale size of the multivariate normal in the definition of the t-distribution. This is to improve the efficiency of the sampler and obtain an acceptance of about 35%, based on tests the scale = 10 was assumed.

## Model Parameters and Priors:

Sherpa 2-D Gaussian model **sigmagauss2d** is selected as a 2-D source model with the following parameters:

- *x,y* - center of the gaussian
- *sigma_a, sigma_b* - the sigma of the gaussian along the major and minor axis
- *theta* - the angle of the major axis, in radians, measured counterclockwise from the x axis (i.e. the line y=0)
- *Ampl* - the maximum peak value of the gaussian model.

When fitting with a point source model, only the center of the gaussian and the amplitude of the model are fit, while the other parameters are kept frozen at default values (*sigma_a, sigma_b* are set to the size of 1 image pixel; *theta* is set to 0), and full set of parameters is fit for extended sources.

Bounded uniform flat priors for the center and sigma parameters (*x,y, sigma_a, sigma_b*) are used. This is accomplished with **box2d** Sherpa model with the limits based on the minimum and maximum allowed parameter values, i.e. the limits of the source image. The Sherpa default flat priors are used for *theta* and *ampl* parameters within the default parameter limits.. Note that MLE sets the *ampl* limits for the maximum likelihood fit using a specific *guess_amplitude()* function which depends on a source count rate, so the flat prior for the *ampl* has the soft minimum and maximum limit based on the specific value for each source.

## Number of iterations

*get_draws()* samples the posterior probability density starting from the model parameters at the maximum likelihood given by the MLE optimization. Simulations were used to set the number of iterations which was based on the quality of the draws and the runtimes. The number of iterations in the pipeline  is set to 5000 for a point source (3 parameters) and 15000 for an extended source (6 parameters).

The quality of the draws in the pipeline can be assessed with the *rhat* parameter (Gelman and Rubin, 1992). To calculate *rhat* values, the chain for each parameter was divided into two parts, the second half of the chains (2500 iterations) was then divided into 10 sections and the within section and between sections variances were used in the calculation of the *rhat.*

## Output files

The fits file table contains the output of *get_draws()*, with each row specifying a draw and columns containing statistics (*cash* in Sherpa), acceptance flag and parameter values. The first row in the file lists the initial parameter values (best-fit from MLE optimization).  Two separate blocks are used to store the draws for the point source model (SRCDRAWS) and the extended source model (EXTDRAWS), if the extended source model was also fit in the MLE.

Acceptance fraction, and *rhat* values for each parameter are given in the header of the fits file.

Examples of the draws files for 3C273 observations in the CSC2:

**Point source:**

$ dmlist acisf01711_001N021_r0052b_draws3.fits.gz blocks

```
--------------------------------------------------------------------------------
Dataset: acisf01711_001N021_r0052b_draws3.fits.gz
--------------------------------------------------------------------------------

    Block Name                      Type        Dimensions
--------------------------------------------------------------------------------
Block    1: PRIMARY                    Null
Block    2: SRCDRAWS                     Table        6 cols x 5001     rows
```

$ dmlist acisf01711_001N021_r0052b_draws3.fits.gz header |more

```
--------------------------------------------------------------------------------
Header keys for block SRCDRAWS
--------------------------------------------------------------------------------

0001 STAT_TYP           CASH                    String       Fitting statistic
0002 ACC_FRAC              0.33080               Real8       MCMC draws acceptance fraction
0003 RHATAMPL                1.0104938294         Real8       Amplitude draws convergence criterion
0004 RHAT_X                 1.0277521659         Real8       X position draws convergence criterion
0005 RHAT_Y                 1.0123222439         Real8       Y position draws convergence criterion
```

**Extended source:**

$ dmlist acisf01711_001N021_r0096b_draws3.fits.gz blocks

```
--------------------------------------------------------------------------------
Dataset: acisf01711_001N021_r0096b_draws3.fits.gz
--------------------------------------------------------------------------------

     Block Name                        Type         Dimensions
--------------------------------------------------------------------------------
Block    1: PRIMARY                      Null
Block    2: SRCDRAWS                     Table        6 cols x 5001     rows
Block    3: EXTDRAWS                     Table       12 cols x 15001    rows
```

$ dmlist acisf01711_001N021_r0096b_draws3.fits.gz[EXTDRAWS] header |more

```
--------------------------------------------------------------------------------
Header keys for block EXTDRAWS
--------------------------------------------------------------------------------

0001 STAT_TYP           CASH                    String      Fitting statistic
0002 ACC_FRAC             0.36880                Real8       MCMC draws acceptance fraction
0003 RHATSIGA              1.4093168273          Real8       Sigma_a draws convergence criterion
0004 RHATSIGB              1.1257415111          Real8       Sigma_b draws convergence criterion
0005 RHATAMPL              1.1456318595          Real8       Amplitude draws convergence criterion
0006 RHATROTA              1.0124760525          Real8       Rotang_a draws convergence criterion
0007 RHAT_X                1.0174783236          Real8       X position draws convergence criterion
0008 RHAT_Y                1.0149884518          Real8       Y position draws convergence criterion
```

**References:**


Andrew Gelman and Donald B Rubin. Inference from iterative simulation using multiple sequences (with discussion). Statistical Science, 7(4):457–511, 1992.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. Bayesian Data Analysis, third edition. CRC Press, 2013.

van Dyk, D.A., Connors, A., Kashyap, V.L., & Siemiginowska, A. 2001, Ap.J., 548, 224, Analysis of Energy Spectra with Low Photon Counts via Bayesian Posterior Simulation.