Using the Effective Sample Size (ESS) to Monitor Convergence and Quality of Results with Bayesian Estimation

Tihomir Asparouhov & Bengt Muthén

June 10, 2025

In this note we discuss the computation of the effective sample size (ESS), also sometimes referred to as the estimated sample size, in Bayesian estimation. We describe its recommended use for monitoring convergence and assessing the quality of the results as well as its interpretation.

1 PSR Convergence Criteria and Its Limitations

The Potential Scale Reduction (PSR) convergence criterion is computed for each individual parameter, and the maximum value across all parameters is used for direct convergence determination. The maximum PSR is reported in the console output during iterations and can also be obtained in the Mplus output using the OUTPUT:TECH8 option. The parameter number with the highest PSR is also reported.

However, PSR can occasionally become misleadingly low without achieving complete convergence. Since we generate random sequences, the PSR based on these sequences can become small even when true convergence has not been reached. This problem typically manifests in the following way: the maximum PSR value drops below the required 1.1 cutoff threshold at a certain iteration (note that 1.1 is an approximation—the actual threshold depends on the number of parameters). However, if we run a longer sequence using the FBITER or BITER options, the PSR value does not remain below the threshold but increases again. Sometimes many additional iterations are needed before the PSR settles permanently below the threshold.

The BITER option specifies minimum and maximum MCMC iterations using the syntax BITER=max(min). For example, BITER=(10000) means that 10,000 iterations will be performed, after which the sequence continues until the PSR convergence criterion is satisfied. This convergence issue is more likely to occur with fewer MCMC chains (such as the Mplus default of 2 chains). The problem can also be detected through careful inspection of traceplots for all parameters—if convergence has been claimed prematurely, at least one traceplot will not resemble the desired "caterpillar" pattern.

2 Understanding the Effective Sample Size (ESS)

ESS provides an additional tool for convergence evaluation. It is computed for each individual parameter and indicates how accurate the estimated posterior distribution is. More precisely, ESS estimates the number of independent observations drawn from the posterior distribution.

When Bayesian estimation uses I iterations with C MCMC chains, Mplus uses draws from the second half of each chain. This means the posterior distribution used for point estimates and confidence intervals contains exactly $C \cdot I/2$ values. However, these values are not independent—within each chain, the draws are autocorrelated.

The ESS estimate represents the adjusted number of draws that accounts for this autocorrelation. More accurately, the estimated posterior distribution is as reliable as if we had drawn ESS truly independent samples.

ESS quantifies the reliability of point estimates, standard errors, and confidence intervals. Higher ESS values indicate greater stability—if we change the random sequence (using the BSEED option), the new estimates are more likely to be similar to the original ones.

For a point estimate based on the posterior mean, with estimated mean m and standard deviation σ, we can be 95% confident that changing BSEED would yield a point estimate within the interval $(m - 1.96\sigma/\sqrt{ESS}, m + 1.96\sigma/\sqrt{ESS})$. As we increase the number of iterations I (using FBITER or BITER), ESS increases as well, allowing us to make this interval as narrow as practically meaningful.

3 Practical Guidelines for Using ESS

Practical guidance for ESS thresholds varies somewhat in the literature. Popular recommendations suggest running MCMC chains until all parameters achieve ESS values of at least 200 or 400. Mplus generates the following warning when ESS falls below 100:

```
THE EFFECTIVE SAMPLE SIZE FOR THE POSTERIOR DISTRIBUTION OF PARAMETER 7 IS ONLY 24.
ESS ¿ 200-400 IS RECOMMENDED. USE BITER=(1600) OR SIMPLIFY THE MODEL.
```

The parameters with the worst (lowest) ESS value is identified in the above error message. Also, an estimate is generated for how many iterations are needed to raise ESS above 200. In the above example, it is estimated that a minimum of 1600 iterations are needed to obtain acceptable levels of ESS. The primary method for improving ESS is to increase the number of iterations used in the estimation with the BITER or FBITER options. If the required number of iterations is too high, the THIN option can be used just as well to reduce the number of stored iterations. In the above example BITER=(800); THIN=2; will be equally likely to improve ESS.

The parameters with the worst (lowest) ESS values can be found in OUTPUT:TECH8; the 10 lowest values are reported there. When ESS estimates fall below 20, they become difficult to calculate accurately, and Mplus simply reports them as "less than 20"—indicating that more iterations are needed. ESS values for all parameters can also be found in autocorrelation plots, available through PLOT:TYPE=PLOT3.

ESS is typically very low when autocorrelations in the estimates are very high, which occurs when traceplots don't resemble the desired caterpillar pattern. Therefore, monitoring ESS across all parameters can serve as an alternative to monitoring traceplots. Monitoring ESS is also faster and more objective than examining traceplots.

4 Using ESS for Significance Testing

ESS proves particularly useful when a parameter of importance shows borderline significance—that is, when the confidence (credibility) interval boundary lies very close to zero. We can roughly estimate how much the interval might change with a different random sequence or longer chain using $1.96\sigma/\sqrt{ESS}$.

If we've run the sequence long enough so that the distance between zero and the confidence interval endpoint exceeds $1.96\sigma/\sqrt{ESS}$, we can assume that significance won't change with longer MCMC sequences. If this distance is smaller, we need longer estimation to achieve higher ESS values and a smaller reliability error $1.96\sigma/\sqrt{ESS}$, which would then be less than the distance between zero and the end of the confidence interval.

5 Optimizing Computational Efficiency

ESS can also help reduce the number of iterations needed for a particular estimation. It's not uncommon to use half a million MCMC iterations simply to ensure that Bayesian estimates are sufficiently reliable. ESS allows us to estimate the precision of Bayesian estimates directly, helping us avoid excessive and unnecessarily large numbers of iterations while maintaining adequate reliability.

6 When ESS Remains Low Despite Many Iterations

Sometimes, the Effective Sample Size (ESS) for certain parameters stays low even after running many iterations. This happens for two main reasons:

1. **Poor model identification** – The model structure makes it difficult to estimate parameters accurately
2. **Poor algorithm mixing** – The estimation method struggles to explore the parameter space effectively

7 Why Complex Models Create Problems

Mplus offers many estimation options, and small model changes can dramatically alter which algorithm is used by default. More complex models are more likely to experience poor mixing because:

- **Seemingly minor model additions** may force the software to use more complex, less efficient algorithms

- **The Metropolis-Hastings (MH) algorithm** is often the culprit behind slow mixing. The algorithm is used for estimating parameters without explicit conditional distributions

- **Specific examples** include:
  - Thresholds for ordered categorical outcomes (always estimated with MH)
  - Mixture models with correlations between class indicators (switches to less efficient latent class generation)

8 Solutions: Simplify Your Model

When low ESS persists despite many iterations, try simplifying your model. This approach offers several benefits:

- **Parsimony:** Models with fewer parameters have more power to detect significance

- **Easier identification:** Simpler models are generally easier for the software to identify and estimate

- **More efficient algorithms:** Simplification may allow Mplus to switch to faster, more reliable estimation methods

- **Better mixing:** This leads to higher ESS values and more trustworthy results

9 How to Simplify Strategically

Use your low ESS results as a guide:

- **Remove non-significant parameters** from the model

- **Focus on parameters with low ESS** – these often point to problematic areas that need simplification

- **Make targeted modifications** based on which parts of the model are struggling

10 Technical Computation of ESS

ESS is computed for each individual parameter as follows. Suppose the parameter estimates for parameter p in iteration i for chain c is p_{ic}. Using only the second half of the generated values, we compute the j-th sample autocorrelation in each chain: $\rho_{jc} = \mathrm{Corr}(p_{i,c}, p_{i-j,c})$. The average of these autocorrelations across chains is denoted by ρ_j. Then ESS is computed as:

$$ESS = \frac{CI/2}{1 + 2\sum_{j=1}^{\infty} \rho_j}$$

The denominator serves as a correction factor. When there are no autocorrelations, the posterior distribution elements can be assumed independent, and ESS equals essentially the exact number of draws used to construct the posterior distribution—meaning no reduction in sampling distribution size.

Since the denominator formula extends to infinity, Mplus truncates the sum at the first autocorrelation smaller than 0.01. The numerator is divided by 2 because we use only the second half of the MCMC sequence for this evaluation of the posterior distribution.

References

[1] Vehtari, A., Gelman, A., Simpson, D., Carpenter, B. & Bürkner, P.C. (2021) Rank-normalization, folding, and localization: an improved R for assessing convergence of MCMC. Bayesian Anal. 16, 667-718.

[2] Gelman, A., and K. Shirley (2011) Inference from simulations and monitoring convergence. Pages 163–174 in S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, editors. Handbook of Markov chain Monte Carlo. Chapman and Hall/CRC, New York, New York, USA