```
logLik.bamlss <- function(object, ..., optimizer = FALSE, samples = FALSE)
{
  Call <- match.call()
  Call <- Call[!(names(Call) %in% c("optimizer", "samples"))]
  mn <- as.character(Call)[-1L]
  object <- list(object, ...)
  mstop <- object$mstop
  if(any(names(object) != "")) {
    i <- names(object) == ""
    object <- object[i]
    mn <- mn[i]
  }
  object <- object[mn != "mstop"]
```

Advanced Bayesian Methods: Theory and Applications in R

Monitoring Mixing and Convergence

Nikolaus Umlauf
https://nikum.org/abm.html

Remarks

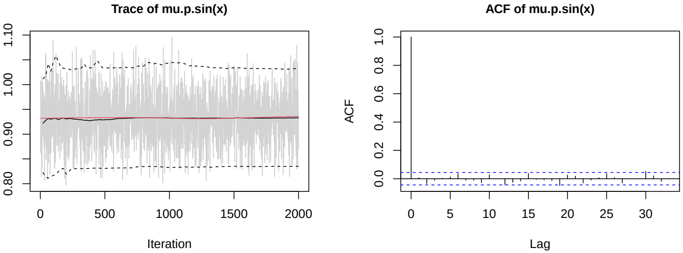
- Monitoring the mixing and convergence of Markov Chain Monte Carlo (MCMC) samples is crucial for reliable inference.
- Poor mixing or lack of convergence may result in biased parameter estimates or incorrect conclusions.
- Several diagnostic techniques and visual tools are available to assess mixing and convergence.

Key Concepts in MCMC Diagnostics

- **Mixing**: How well the Markov chain explores the target distribution.
- **Convergence**: Whether the chain has reached the stationary distribution, i.e., the target posterior distribution.
- **Burn-in**: Initial samples may not represent the stationary distribution and are usually discarded.
- **Effective Sample Size (ESS)**: Indicates how independent the samples are.

Trace Plots

- A trace plot shows the values of the parameters over iterations.
- Well-mixing chains should exhibit random fluctuations around a stable mean.
- Poorly mixing chains may show strong autocorrelation or trend.



Autocorrelation Function (ACF)

- The ACF of a parameter indicates how correlated the samples are at different lags.
- If the ACF decays quickly, the chain is well-mixed.
- Slow decay indicates strong autocorrelation, which reduces the effective sample size.

Effective Sample Size (ESS)

- Effective sample size is a measure of how many independent samples your chain has.
- High autocorrelation leads to lower ESS, meaning fewer independent samples.
- ESS can be calculated in R using the *coda* package.

R Example: Effective Sample Size

```
R> ## Calculate ESS using the coda package.
R> library("coda")
R> samps <- samples(b)
R> print(dim(samps))
[1] 2000   10
R> ess_value <- effectiveSize(samps[, 1:2])
R> print(ess_value)
mu.p.(Intercept)      mu.p.sin(x)
            2000             2000
```

Gelman-Rubin Diagnostic (R-hat)

- The Gelman-Rubin diagnostic compares the variance within chains to the variance between chains.
- $\hat{R}$ close to 1 indicates convergence.
- $\hat{R}$ values significantly greater than 1 suggest lack of convergence.

R Example: Gelman-Rubin Diagnostic

Calculate $\hat{R}$ using the *coda* package.

```
R> library("coda")
R> gelman_diag <- gelman.diag(samples(b, combine = FALSE))
```

Conclusion

- Monitoring the mixing and convergence of MCMC samples is crucial for obtaining reliable estimates.
- Key diagnostics include trace plots, autocorrelation, effective sample size, and the Gelman-Rubin diagnostic.
- R provides useful tools for performing these diagnostics.