

Methodological Details of the Bayesian GLM Model with AR(1) Structure

José Mauricio Gómez Julián

Table of Contents

1	Bayesian Framework for Time Series Modeling with Autoregressive Structure	2
1.1	Specification of the Bayesian Model with AR(1)	2
1.1.1	Complete Model Structure	2
1.1.2	Reference Baseline Model	3
1.1.3	Prior Specification	3
1.2	Temporal Cross-Validation with Leave-Future-Out (LFO).....	3
1.2.1	Rolling-Origin Implementation with Sliding Window.....	3
1.2.2	Predictive Evaluation Criteria.....	4
1.2.3	Victory Criterion per Fold	5
1.3	Bayesian Inference with Hamiltonian Monte Carlo.....	5
1.3.1	NUTS Sampler Configuration	5
1.3.2	Convergence Diagnostics	6
1.4	Determination of Temporal Stability and Support	6
1.4.1	Support Metric	6
1.4.2	Robustness Thresholds	6
1.4.3	Support Interpretation	6
1.5	Comparative Analysis with ECM-MARS Methodology	7
1.5.1	Fundamental Differences	7
1.5.2	Advantages of the Bayesian Approach	7
1.5.3	Relative Limitations	7
1.6	Technical Implementation and Optimizations.....	8
1.6.1	Memory Management and Parallelization	8
1.6.2	Handling Degenerate Cases.....	8

1.7	Complete Pipeline Pseudocode	8
1.8	Technical Notes and Special Considerations	9
1.8.1	Scaling and Numerical Stability	9
1.8.2	Interpretation of Negative Metrics	9
1.8.3	Potential Extensions	9
1.9	Methodological Conclusions	9

1 Bayesian Framework for Time Series Modeling with Autoregressive Structure

This document details the methodology implemented for analyzing causal relationships between production and circulation variables using a Bayesian Generalized Linear Model (BGLM) with first-order autoregressive structure AR(1). This approach differs substantially from the ECM-MARS methodology by adopting a complete Bayesian inferential paradigm, allowing comprehensive quantification of predictive uncertainty and model comparison through predictive information criteria.

1.1 Specification of the Bayesian Model with AR(1)

1.1.1 Complete Model Structure

Let Y_t be the dependent variable and X_t the independent variable at time t . The complete model is specified as:

$$Y_{t,s} = \alpha + \beta_0 \cdot t_s + \sum_{i=1}^L \beta_i X_{t-i,s} + \epsilon_t$$

where:

- $Y_{t,s}$ is the standardized dependent variable: $(Y_t - \mu_Y)/\sigma_Y$
- t_s is the standardized temporal index: $(t - \mu_t)/\sigma_t$
- $X_{t-i,s}$ are the standardized lags of the independent variable
- L is the maximum number of lags (default $L = 3$)
- ϵ_t follows an AR(1) process: $\epsilon_t = \phi\epsilon_{t-1} + \eta_t$, with $\eta_t \sim N(0, \sigma^2)$

Standardization is critical for the efficiency of the NUTS¹ (No-U-Turn Sampler) algorithm, ensuring that all parameters operate on comparable scales and avoiding convergence issues in the Hamiltonian sampler.

1.1.2 Reference Baseline Model

As a comparison benchmark, we estimate a baseline model that excludes information from the independent variable:

$$Y_{t,s} = \alpha + \beta_0 \cdot t_s + \epsilon_t$$

This model captures only the temporal trend and the intrinsic autoregressive structure of Y , serving as an effective null hypothesis against which to evaluate the predictive contribution of the lags of X .

1.1.3 Prior Specification

Priors are specified on the standardized scale to maintain consistency:

- **Regression coefficients** (β_i): $\beta_i \sim N(0,1)$
- **Intercept** (α): $\alpha \sim t_3(0,2.5)$ (Student-t with 3 degrees of freedom)
- **Residual standard deviation** (σ): $\sigma \sim \text{Exponential}(1)$
- **AR(1) coefficient** (ϕ): Implicit uniform prior on $(-1,1)$ for stationarity

The choice of weakly informative priors² reflects a balance between mild regularization and flexibility to learn from the data. The Student-t distribution for the intercept provides robustness against outliers, while the exponential prior for σ ensures positivity with a heavy tail.

1.2 Temporal Cross-Validation with Leave-Future-Out (LFO)

1.2.1 Rolling-Origin Implementation with Sliding Window

Unlike traditional cross-validation that violates the principle of temporal causality, we implement Leave-Future-Out (LFO) with a sliding window. This scheme strictly respects temporal ordering and simulates the real context of prospective forecasting.

The procedure is structured as follows:

¹ NUTS (No-U-Turn Sampler) is an adaptive extension of the Hamiltonian Monte Carlo algorithm that automatically adjusts trajectory lengths to maximize sampling efficiency without requiring manual tuning of the number of leapfrog steps.

² Weakly informative priors provide mild regularization without imposing strong a priori beliefs. In the limit of abundant data, their influence vanishes, recovering estimates similar to maximum likelihood.

1. **Initialization:** The initial training set comprises 70% of the data or a minimum of 90 observations, whichever is greater.
2. **Test horizon:** Each fold evaluates $h = 12$ months ahead, simulating annual forecasts.
3. **Step between folds:** The origin shifts 12 months between successive evaluations.
4. **Sliding window:** The training set size remains constant³, discarding old observations when incorporating new ones. This contrasts with the expanding window where the training set grows monotonically.

The advantage of the sliding window lies in its sensitivity to regime changes and its ability to adapt to broadly non-stationary dynamics. While an expanding window assumes constant parameters throughout the entire history, the sliding window recognizes that economic relationships may evolve, maintaining only the most relevant “recent memory” for forecasting.

1.2.2 Predictive Evaluation Criteria

For each fold k , we compare the predictive performance of the complete model against the baseline model using two complementary criteria:

1.2.2.1 Expected Log Predictive Density (ELPD)

The ELPD⁴ quantifies the expected log predictive density for new observations:

$$\text{ELPD}_k = \sum_{t \in \text{test}_k} \log \int p(y_t | \theta) p(\theta | \text{data}_{\text{train},k}) d\theta$$

In practice, we approximate this integral using the log-sum-exp estimator over posterior samples:

$$\widehat{\text{ELPD}}_k = \sum_{t \in \text{test}_k} \log \left(\frac{1}{S} \sum_{s=1}^S p(y_t | \theta^{(s)}) \right)$$

where $\theta^{(s)}$ are the posterior samples and S is the number of post-warmup iterations.

³ The constant size of the training set in sliding window ensures that all folds have comparable statistical power, avoiding bias toward late folds that occurs with expanding windows.

⁴ ELPD (Expected Log Predictive Density) is the fundamental predictive evaluation criterion in Bayesian statistics, generalized by WAIC and LOO-CV. It measures the model’s ability to assign high probability to new observations.

The difference $\Delta\text{ELPD}_k = \text{ELPD}_{k,\text{full}} - \text{ELPD}_{k,\text{base}}$ measures the gain in predictive capacity. A $\Delta\text{ELPD} > 0$ indicates that the complete model assigns higher probability to the observed out-of-sample data.

1.2.2.2 Root Mean Square Error (RMSE) and Classical Metrics

We complement the ELPD with traditional metrics on the original scale of the variables:

- **RMSE:** $\sqrt{\frac{1}{n_{\text{test}}} \sum_t (y_t - \hat{y}_t)^2}$
- **MAE:** $\frac{1}{n_{\text{test}}} \sum_t |y_t - \hat{y}_t|$
- **sMAPE:** $\frac{100}{n_{\text{test}}} \sum_t \frac{2|y_t - \hat{y}_t|}{|y_t| + |\hat{y}_t|}$
- **R²:** $1 - \frac{\sum_t (y_t - \hat{y}_t)^2}{\sum_t (y_t - \bar{y})^2}$ (protected against $SST \approx 0$)

where \hat{y}_t is the posterior mean of predictions, obtained via `posterior_epred()` and transformed back to the original scale.

1.2.3 Victory Criterion per Fold

A complete model “wins” in fold k if and only if:

$$\begin{cases} \Delta\text{ELPD}_k > 0 & \text{(better predictive density)} \\ \Delta\text{RMSE}_k < 0 & \text{(lower squared error)} \end{cases}$$

This dual criterion requires superiority in both probabilistic (ELPD) and deterministic (RMSE) terms, avoiding spurious victories from improvements in a single dimension.

1.3 Bayesian Inference with Hamiltonian Monte Carlo

1.3.1 NUTS Sampler Configuration

Inference is performed using the NUTS⁵ algorithm implemented in Stan through `brms` with `cmdstanr` backend. Sampling parameters are calibrated to ensure robust convergence:

- **Chains:** 4 independent chains
- **Total iterations:** 1500 per chain
- **Warmup:** 750 iterations (50% for adaptation)
- **Adapt delta:** 0.95 (reduces divergent steps)
- **Max treedepth:** 12 (avoids premature tree truncation)

⁵ The NUTS algorithm eliminates the need to manually tune the number of steps in HMC through a stopping criterion based on “U-turns” in parameter space, where the trajectory begins to double back toward its origin.

Parallelization is implemented at the chain level (`parallel_chains = 4`), not within each chain, to maintain sampler efficiency without CPU oversubscription.

1.3.2 Convergence Diagnostics

Although not explicitly reported in production code for efficiency, standard diagnostics include:

- **\hat{R} (Rhat):** Should be < 1.01 for all parameters
- **ESS (Effective Sample Size):** Bulk-ESS and Tail-ESS > 400 per chain
- **Divergences:** Ideally 0; few divergences ($< 1\%$) may be tolerable
- **BFMI (Bayesian Fraction of Missing Information):** > 0.2 indicates good behavior

1.4 Determination of Temporal Stability and Support

1.4.1 Support Metric

We define support as:

$$\text{support} = \frac{\text{folds_pass}}{\text{folds}}$$

where `folds_pass` counts the folds where the complete model satisfies the dual victory criterion. This metric quantifies the temporal consistency of predictive power.

1.4.2 Robustness Thresholds

We establish two confidence levels:

- **Strict threshold:** support ≥ 0.70 with minimum 5 valid folds
- **Moderate threshold:** support ≥ 0.60 with minimum 5 valid folds

The requirement of an absolute minimum of folds prevents false positives from small denominators (e.g., $1/1 = 100\%$ support does not constitute robust evidence).

1.4.3 Support Interpretation

High support indicates that the predictive relationship persists across different temporal regimes. Support values:

- **0.80-1.00:** Very stable relationship, robust to regime changes
- **0.60-0.79:** Moderately stable relationship, sensitive to some epochs
- **0.40-0.59:** Unstable relationship, dependent on temporal context
- **< 0.40 :** Spurious relationship or highly specific to particular periods

1.5 Comparative Analysis with ECM-MARS Methodology

1.5.1 Fundamental Differences

Aspect	ECM-MARS	Bayesian GLM AR(1)
Paradigm	Frequentist with sequential tests	Integral Bayesian
Cointegration	Explicit requirement (Engle-Granger/Johansen)	Implicit in AR(1) structure
Non-linearity	MARS with adaptive splines	Linear in parameters
Uncertainty	HAC standard errors	Complete posterior distribution
Comparison	Significance tests	ELPD + predictive metrics
Complexity	High (multiple pre-tests)	Moderate (direct model)

1.5.2 Advantages of the Bayesian Approach

1. **Comprehensive uncertainty quantification:** Posterior distributions capture all parametric uncertainty, naturally propagating it to predictions.
2. **Direct model comparison:** ELPD provides a unified metric for comparing non-nested models without multiplicity adjustments.
3. **Robustness to assumption violations:** Regularizing priors and AR(1) structure accommodate moderate deviations from classical assumptions.
4. **Interpretability:** Credible intervals have direct probabilistic interpretation, unlike frequentist confidence intervals.

1.5.3 Relative Limitations

1. **Computational cost:** MCMC sampling is orders of magnitude slower than OLS/MARS estimation.
2. **Prior sensitivity:** In small samples, prior choice can substantially influence results.
3. **Linearity:** The absence of non-linear terms (splines) may limit flexibility to capture complex relationships.

1.6 Technical Implementation and Optimizations

1.6.1 Memory Management and Parallelization

- **Pairwise parallelization:** Each pair ($X \rightarrow Y$) is processed sequentially, but MCMC chains within each model are parallelized.
- **Memory release:** Stan objects are implicitly released after each fold through R's garbage collector.
- **cmdstanr backend:** More efficient than rstan for large models, with better memory management and diagnostics.

1.6.2 Handling Degenerate Cases

The code includes safeguards for:

- **Near-zero variance:** If $\sigma_Y \approx 0$ in the training set, the fold is skipped.
- **Convergence errors:** Models that fail to converge return NULL and are excluded from analysis.
- **Undefined log-likelihoods:** Finite values are verified before calculating ELPD.

1.7 Complete Pipeline Pseudocode

FOR each pair (X, Y) in {production × circulation} × {2 directions}:

1. Build matrix with Y, X, and lags(X, 1:L)
2. Remove rows with missing values

FOR each fold in rolling_splits(sliding_window):

3. Split into train/test
4. Standardize variables using train statistics
5. Fit baseline_model: $Y_s \sim 1 + t_s + \text{AR}(1)$
6. Fit full_model: $Y_s \sim 1 + t_s + X_{\text{lags}} + \text{AR}(1)$
7. Calculate ELPD for both models on test
8. Generate point predictions (posterior_epred)
9. Transform predictions to original scale
10. Compute metrics (RMSE, MAE, sMAPE, R^2)
11. Determine victory: $(\Delta \text{ELPD} > 0) \wedge (\Delta \text{RMSE} < 0)$

12. Calculate support = wins / total_folds
13. Average metrics across folds

14. Ranking by (support DESC, ΔELPD DESC, ΔRMSE ASC)
15. Filter winners by support thresholds
16. Export results and visualizations

1.8 Technical Notes and Special Considerations

1.8.1 Scaling and Numerical Stability

Per-fold scaling (not global) is crucial because:

1. Normalization statistics must be calculated only with train data to avoid *data leakage*⁶
2. Variable scales may change substantially between temporal windows
3. NUTS requires all parameters to operate on $\mathcal{O}(1)$ scales for efficiency

1.8.2 Interpretation of Negative Metrics

It is possible to obtain $R^2 < 0$ in out-of-sample evaluation when the model predicts worse than the sample mean. This does not indicate calculation error but genuine poor predictive performance. Similarly, RMSE ratios > 1 indicate that the complete model worsens predictions relative to the baseline.

1.8.3 Potential Extensions

The current framework admits several natural extensions:

- **Heteroscedasticity:** Model σ_t as a function of time or covariates
- **Mixture of experts:** Multiple AR components with variable weights
- **Non-linearity:** Splines or basis functions via $s(\cdot)$ in brms
- **Random effects:** Hierarchical structure for multiple series
- **Multi-step forecasting:** Extend beyond one-step-ahead

1.9 Methodological Conclusions

The Bayesian GLM methodology with AR(1) offers a robust and principled framework for evaluating predictive relationships between economic variables. Unlike the ECM-MARS approach that requires sequential validation of multiple assumptions (stationarity, cointegration, adjustment speed), the Bayesian model integrates all uncertainty into a unified framework.

Results suggest that the most robust relationships (support > 0.70) typically correspond to pairs where economic theory predicts strong causality, while relationships with moderate support (0.60-0.70) may reflect indirect transmission channels or dependence on the prevailing economic regime.

⁶ Data leakage occurs when information from the test set contaminates the training process, artificially inflating performance metrics. Scaling with global statistics would constitute a subtle but pernicious form of leakage.

Dual evaluation via ELPD and RMSE ensures that selected models not only fit well in terms of squared error but also adequately capture the probabilistic structure of the data. This double requirement effectively filters spurious relationships that might appear promising under a single criterion.
