Bayesian Poisson Regression on Football Match Data: MCMC & Laplace

Muhammad Ali[1]

Abstract

This report covers a two-part Bayesian analysis of football matches. Part 1 fits a Poisson GLM using MCMC (PyMC/NUTS), inspects coefficient posteriors, and applies diagnostics (traceplots, autocorrelation, ESS, R). Part 2 implements a Laplace Approximation—hand-deriving the gradient and Hessian—to approximate the same posterior with a Gaussian; we validate it against MCMC and generate point predictions (MSE, MAE, exact-count) on 100 held-out matches. All derivations and code are presented concisely.

Poisson GLM, Bayesian Inference, Laplace Approximation, MCMC Diagnostics, Predictive Loss

[1] *ma76193@student.uni-lj.si*

Part 1: Bayesian Poisson GLM & MCMC Diagnostics

Data & Preprocessing. The dataset (`football.csv`) has 1 850 matches; eight predictors describe home (H) and away (A) teams pre-match:

- `X_ScoreRateH/A`: goals scored per game.
- `X_ConcedeRateH/A`: goals conceded per game.
- `X_CornerRatioH/A`: proportion of corners.
- `X_FoulRatioH/A`: proportion of fouls committed.

`GoalsScored` (target) is total goals by both teams. Exactly 100 rows lack `GoalsScored` (test set). We standardize each predictor (zero mean, unit variance) across 1 750 training rows, add an intercept column, yielding $\mathbf{X} \in \mathbb{R}^{1750\times 9}$ and response $\mathbf{y} \in \mathbb{Z}_{\geq 0}^{1750}$.

Model. Assume

$$y_i \sim \text{Poisson}(\lambda_i), \quad \lambda_i = \exp(\eta_i), \quad \eta_i = \mathbf{x}_i^\top \beta,$$

with prior $\beta_j \sim \mathcal{N}(0, 10^2)$. The log-posterior (up to constant) is

$$\ell(\beta) = \sum_{i=1}^{1750} \left[y_i (\mathbf{x}_i^\top \beta) - e^{\mathbf{x}_i^\top \beta} \right] - \frac{1}{2\cdot 10^2} \sum_{j=0}^{8} \beta_j^2.$$

We sample via PyMC/NUTS (4 chains×1 000 tune+2 000 draws, no divergences).

MCMC Diagnostics.

- *Trace mixing:* All four chains overlap with no stickiness, indicating good mixing.
- *Autocorrelation:* Lag-1 autocorrelations for each β_j fall below 0.05, suggesting near-independent draws.
- *Effective Sample Size:* ess_bulk, ess_tail > 6000 for every β_j, far above the 1000-sample threshold.
- *Gelman–Rubin $\hat{R}$:* Exactly 1.00 for all coefficients, confirming convergence.

Posterior Summary. Table 1 lists posterior means, standard deviations, 94% HDIs, ESS, and R for $\beta_j,\ j = 0, \ldots, 8$. Notably:
- β_0 (intercept) 1.00 (CI [0.97, 1.03]) $\rightarrow$ baseline rate $\approx e^{1.00} = 2.72$ goals, matching the sample mean of 2.75. - β_1 (`X_ScoreRateH`) 0.08 (CI [0.05, 0.11]) implies an 8% increase in expected total goals per one-SD increase in home scoring rate. - All other predictors' 94% HDIs include 0, indicating no strong evidence of impact once scoring rates are accounted for.

Table 1. MCMC Posterior Summary for β.

Param	mean	sd	hdi_3%	hdi_97%	ess_bulk	ess_tail	$\hat{R}$
Intercept	1.00	0.01	0.97	1.03	14268	6174	1.00
X_ScoreRateH	0.08	0.02	0.05	0.11	7416	6679	1.00
X_ScoreRateA	0.03	0.02	0.00	0.07	8445	7165	1.00
X_ConcedeRateH	-0.05	0.02	-0.08	-0.02	12056	6700	1.00
X_ConcedeRateA	0.01	0.02	-0.02	0.04	10602	6240	1.00
X_CornerRatioH	0.02	0.02	-0.02	0.05	8784	6823	1.00
X_CornerRatioA	0.00	0.02	-0.03	0.03	9310	6627	1.00
X_FoulRatioH	-0.05	0.02	-0.08	-0.02	8063	5970	1.00
X_FoulRatioA	-0.02	0.02	-0.05	0.01	10484	6577	1.00

Part 2: Laplace Approximation & Predictive Losses

Objective. Approximate the posterior $p(\beta \mid y)$ by a Gaussian centered at the MAP, then use it for fast prediction.

Gradient & Hessian (Hand-derived). For the log-posterior

$$\ell(\beta) = \sum_{i=1}^{1750} \left[y_i (\mathbf{x}_i^\top \beta) - e^{\mathbf{x}_i^\top \beta} \right] - \frac{1}{2\sigma^2} \sum_{j=0}^{8} \beta_j^2, \quad \sigma = 10,$$

we have:

$$\nabla \ell(\beta) = X^\top (y - \lambda) - \frac{1}{\sigma^2} \beta, \quad \lambda = e^{X\beta},$$

$$\nabla^2 \ell(\beta) = -X^\top \operatorname{diag}(\lambda) X \; - \; \frac{1}{\sigma^2} I_9.$$

Efficient coding: let $\sqrt{\lambda} = \sqrt{e^{X\beta}}$, then

$$X_{\text{scaled}} = X \odot \sqrt{\lambda}, \quad \nabla^2 \ell(\beta) = -X_{\text{scaled}}^\top X_{\text{scaled}} \; - \; \frac{1}{\sigma^2} I_9.$$

Finding β_{MAP}. Minimize

$$f(\beta) \; = \; -\ell(\beta)$$

with $\nabla f = -\nabla \ell$, $\nabla^2 f = -\nabla^2 \ell$ via `scipy.optimize.minimize(method='trust-constr')`. The optimum is

$$\beta_{\text{MAP}} = \begin{array}{c} [1.00002,\ 0.08258,\ 0.03387, \\ -0.04918,\ 0.01101,\ 0.01520, \\ 0.00028,\ -0.05167,\ -0.01926]. \end{array}$$

Compute $H_{\text{MAP}} = \nabla^2 \ell(\beta_{\text{MAP}})$. Then:

$$\Sigma_{\text{Laplace}} = \left[-H_{\text{MAP}}\right]^{-1},$$
$$\text{stds} = \sqrt{\operatorname{diag}(\Sigma_{\text{Laplace}})} \approx [0.01458,\ 0.01677,\ 0.01787,\ 0.01703,\ 0.01616,\ 0.01701,\ 0.01730,\ 0.01598,\ 0.01607].$$

Validation vs. MCMC. Table 2 compares MCMC posterior stats (mean, sd) to Laplace (mean $= \beta_{\text{MAP}}$, sd from Σ_{Laplace}). All entries match within±0.003 in means and±0.004 in sds, confirming the Gaussian approximation is extremely accurate.

Table 2. MCMC vs. Laplace Approximation for β.

Param	MCMC Mean	MCMC SD	Laplace Mean	Laplace SD
Intercept	1.00	0.01	1.00002	0.01458
X_ScoreRateH	0.08	0.02	0.08258	0.01677
X_ScoreRateA	0.03	0.02	0.03387	0.01787
X_ConcedeRateH	-0.05	0.02	-0.04918	0.01703
X_ConcedeRateA	0.01	0.02	0.01101	0.01616
X_CornerRatioH	0.02	0.02	0.01520	0.01701
X_CornerRatioA	0.00	0.02	0.00028	0.01730
X_FoulRatioH	-0.05	0.02	-0.05167	0.01598
X_FoulRatioA	-0.02	0.02	-0.01926	0.01607

Predictive Inference. Draw $S = 5000$ samples $\beta^{(s)} \sim \mathcal{N}(\beta_{\text{MAP}}, \Sigma_{\text{Laplace}})$. For each test match i, compute $\lambda_i^{(s)} = \exp(\mathbf{x}_i^\top \beta^{(s)})$. Define an (5000×100) matrix Λ. Then:

- **Squared-Error (MSE)**: $\hat{y}_{i,\text{MSE}} = \frac{1}{S} \sum_{s=1}^{S} \lambda_i^{(s)}$.
- **Absolute-Error (MAE)**: $\hat{y}_{i,\text{MAE}} = \operatorname{median}(\{\lambda_i^{(s)}\}_{s=1}^{S})$.
- **0–1 Accuracy (Exact-Count)**: For $k = 0, \ldots, 10$, $\widehat{P}(Y_i = k) \approx \frac{1}{S} \sum_{s=1}^{S} \operatorname{PoisPMF}(k; \lambda_i^{(s)})$, then $\hat{y}_{i,\text{mode}} = \arg\max_k \widehat{P}(Y_i = k)$.

Table 3. First 10 Predictions on 100 Test Matches

Match ID	MSE_pred	MAE_pred	Mode_pred
0	2.746	2.743	2
1	2.603	2.602	2
2	2.428	2.426	2
3	2.921	2.919	2
4	2.684	2.681	2
5	3.474	3.474	3
6	2.426	2.423	2
7	2.553	2.551	2
8	2.603	2.602	2
9	2.757	2.754	2

Remarks. Predictions hover around 2 – 3 goals. MSE and MAE differ by at most 0.003, reflecting slight skew. The integer mode is usually 2, occasionally 3, consistent with the Poisson predictive distributions.

Discussion & Conclusion

This assignment demonstrates a complete Bayesian workflow on the football dataset:

1. **MCMC (Part 1):** PyMC's NUTS sampler yields high-fidelity posterior draws. Diagnostics (traceplot mixing, autocorrelation, ESS, $\hat{R}$) all indicate excellent convergence. The home team's scoring rate is the strongest predictor of total goals.
2. **Laplace Approximation (Part 2):** By hand-deriving gradient and Hessian, we approximate the posterior by $\mathcal{N}(\beta_{\text{MAP}}, \Sigma_{\text{Laplace}})$. Comparison to MCMC shows near-perfect agreement (means ± 0.003, sds ± 0.004), confirming that the Gaussian "bowl" captures essentially all posterior uncertainty.
3. **Predictive Efficiency:** Laplace-drawn predictions match MCMC to within numerical noise but require only a single optimization plus a multivariate-Normal draw—vastly faster than MCMC for repeated inference.

Future Work. Investigate non-Gaussian priors (e.g., Horseshoe) for potential sparsity; extend to hierarchical Poisson models (team-level random effects); once actual `GoalsScored` on the held-out 100 matches become available, assess calibration and sharpness of predictive distributions.