

# Hurdle Modeling in R Using Bayesian Inference

Taylor Trippe, Dr. Earvin Balderama

Department of Mathematics & Statistics, Loyola University Chicago, Chicago, IL, USA



## Motivation

- **Need:** Effective modeling methods for **zero-inflated** and/or **over-dispersed** count data.
- **Goal:** Develop a package of user-friendly functions, utilizing **MCMC sampling**, that will best model problematic count data that cannot be fit to any typical distribution.

## Discription

- **hurdle(...):** Used to fit single or double-hurdle regression models to count data via **Bayesian inference**.
- **hurdle\_control(...):** Various parameters for fitting control of **hurdle model** regression.

## Usage

- `hurdle(y, x = NULL, hurdle = Inf, dist = c("poisson", "nb", "gpd"), dist.2 = c("none", "gpd", "poisson", "nb"), control = hurdle_control(...), iters = 1000, burn = 500, nthin = 1, plots = T, progress.bar = T)`
- `hurdle_control(a = 1, b = 1, size = 1, beta.prior.mean = 0, beta.prior.sd = 1000, beta.tune = 1, pars.tune = 0.2, lam.start = 1, mu.start = 1, sigma.start = 1, xi.start = 1)`

## Arguments

- **hurdle(...)**
  - ▷ **y:** numeric response vector.
  - ▷ **x:** optional numeric predictor matrix.
  - ▷ **hurdle:** numeric threshold ( $\psi$ ) for 'extreme' observations of two-hurdle models. **NULL** for one-hurdle models.
  - ▷ **dist:** character specification of response distribution.
  - ▷ **dist.2:** character specification of response distribution for 'extreme' observations of two-hurdle models.
  - ▷ **control:** list of parameters for controlling the fitting process, specified by `hurdle_control()`.
  - ▷ **iters:** number of iterations for the Markov chain to run.
  - ▷ **burn:** numeric burn-in length.
  - ▷ **nthin:** numeric thinning rate.
  - ▷ **plots:** logical operator. **TRUE** to print plots.
  - ▷ **progress.bar:** logical operator. **TRUE** to print progress bar.
- **hurdle\_control(...)**
  - ▷ **a:** shape parameter for Gamma( $a, b$ ) prior distributions.
  - ▷ **b:** rate parameter for Gamma( $a, b$ ) prior distributions.
  - ▷ **size:** size ( $r$ ) parameter for NB( $r, \mu$ ) likelihood distributions.
  - ▷ **beta.prior.mean:** mean ( $\mu$ ) for Normal( $\mu, \sigma^2$ ) prior distributions.
  - ▷ **beta.prior.sd:** st. deviation ( $\sigma$ ) for Normal( $\mu, \sigma^2$ ) prior distributions.
  - ▷ **beta.tune:** MCMC tuning for regression coefficient estimation.
  - ▷ **pars.tune:** MCMC tuning for parameter estimation.
  - ▷ **lam.start, mu.start, sigma.start, xi.start:** initial value(s) for parameter(s) of 'extreme' observations distribution.

## Functionality & Applications

Response data:

Surveys → Boat/aerial continuous-time strip transects.

Environmental covariates:

$\mathbf{x}_1$  = Sea surface temperature.  
 $\mathbf{x}_2$  = Ocean depth.  
 $\mathbf{x}_3$  = Chlorophyll-a level.  
 $\mathbf{x}_4$  = Distance-to-shore.

Temporal effects (Fourier basis):

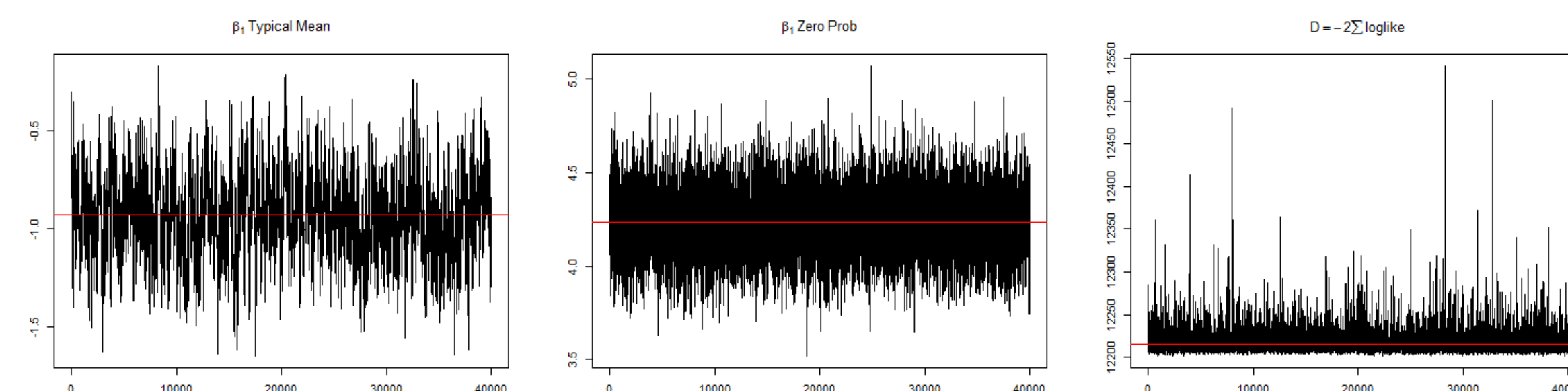
$\mathbf{x}_5 = \sin(\frac{\pi}{6} \cdot \text{Month})$ .  
 $\mathbf{x}_6 = \cos(\frac{\pi}{6} \cdot \text{Month})$ .

Avian Counts: Sooty Shearwater

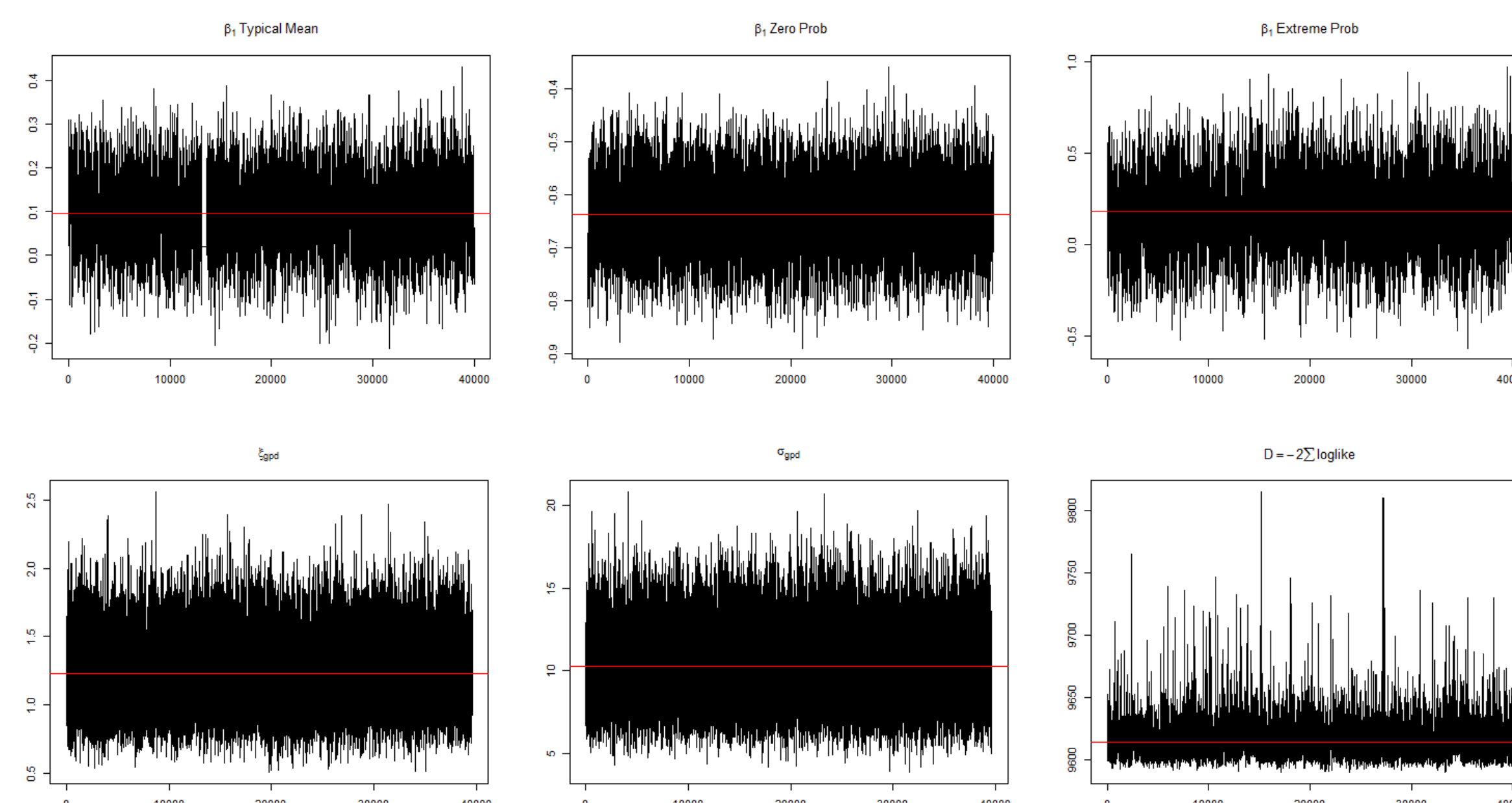
Count	Frequency
0	33503
1 – 10	412
11 – 100	88
101 – 500	16
501 +	3

Fit a model to the data in using **hurdlr** package functions:

```
f <- hurdle(y, x = x, hurdle = Inf,
  dist = "nb",
  dist.2 = "none",
  control = control,
  iters = 45000, burn = 5000, nthin = 10,
  plots = T, progress.bar = T)
```



```
f <- hurdle(y, x = x, hurdle = hurd,
  dist = "nb",
  dist.2 = "gpd",
  control = control,
  iters = 45000, burn = 5000, nthin = 10,
  plots = T, progress.bar = T)
```



Compare Single vs Double Hurdle model:

- Improved convergence of model parameters.
- Decrease in deviance (supported by DIC and pD).
- Increase in predictive power (based on predictive ordinates PPO and CPO).

## Model

- **Single-Hurdle modeling** is used to fit **zero-inflated** data.

**Likelihood of observing count  $y_i$ :**

$$f(y_i|\theta) = \begin{cases} p_i, & y_i = 0, \\ [1 - p_i] \cdot \text{NB}(\mu_i, r), & 1 \leq y_{ij} < \psi, \end{cases}$$

- **Double-Hurdle modeling** may account for both excessive **zero-inflation** and extreme **over-dispersion**.

**Likelihood of observing count  $y_i$ :**

$$f(y_i|\theta) = \begin{cases} p_i, & y_i = 0, \\ [1 - p_i] \cdot [1 - q_i] \cdot \text{NB}(\mu_i, r), & 1 \leq y_{ij} < \psi, \\ [1 - p_i] \cdot q_i \cdot \text{GPD}(\psi, \sigma, \xi), & y_{ij} \geq \psi. \end{cases}$$

- **Negative binomial (NB)** for small, "typical" counts.
  - ▷ Left-truncated at 0 and right-truncated at threshold  $\psi$ .
    - ▷ Single-hurdle models are truncated only at 0.
    - ▷ ZIP, ZINB, Poisson-hurdle, NB-hurdle distributions are common.
- **Generalized Pareto (GPD)** for large, right-tail counts.
  - ▷ GPD density is  $> 0$  at threshold  $\psi$  or above.

## Bayesian Regression

- A series of **linear regressions** are run to estimate:
  - $\mathbf{p} = P(\text{zero-count})$   
 $\text{logit}(\mathbf{p}) = \mathbf{X}\gamma$
  - $\mu = \text{mean of typical-count distribution.}$   
 $\text{log}(\mu) = \mathbf{X}\beta$
  - $\mathbf{q} = P(\text{large-count} \mid \text{nonzero-count})$   
 $\text{logit}(\mathbf{q}) = \mathbf{X}\delta$
- A Bayesian approach to linear regression allows for the user to **characterize the uncertainty** in the **response vector  $\mathbf{y}$**  through a probability distribution  $f(\mathbf{y}|\theta)$ .
- Parameters are updated using a home-grown **Markov chain Monte Carlo** algorithm utilizing **Metropolis** sampling.

## Current Work & Future Considerations

- Incorporate other distributions; i.e., **log-normal** models.
- Treat threshold parameter  $\psi$  as unknown.
- Create similar functions for applying models to zero-inflated (ZIP, ZINP) count distributions.
- Increase functionality to allow for **hierarchical regression** of **nested data**.
- Expand on function output to include clean and variable plots, convergence and coverage diagnostics, predictive power, etc.
- Release **hurdlr** package to CRAN for public use.

## Acknowledgements

- Software used: ([www.r-project.org](http://www.r-project.org))
- Data acquired from: **Avian Compendium** (NOAA)
- Special thanks to Timothy O'Brien and the Loyola University Chicago Department of Mathematics and Statistics.