Bayesian Logistic Regression
with Polya-Gamma latent variables

Kaspar Märtens
Sherman Ip

23rd October 2015

Classical logistic regression

- Modelling association between binary outcome y_i and feature vector x_i

$$y_i \mid (x_i, \beta) \sim \text{Bernoulli}(p_i)$$
$$p_i = \frac{1}{1 + e^{-x_i^T \beta}}$$

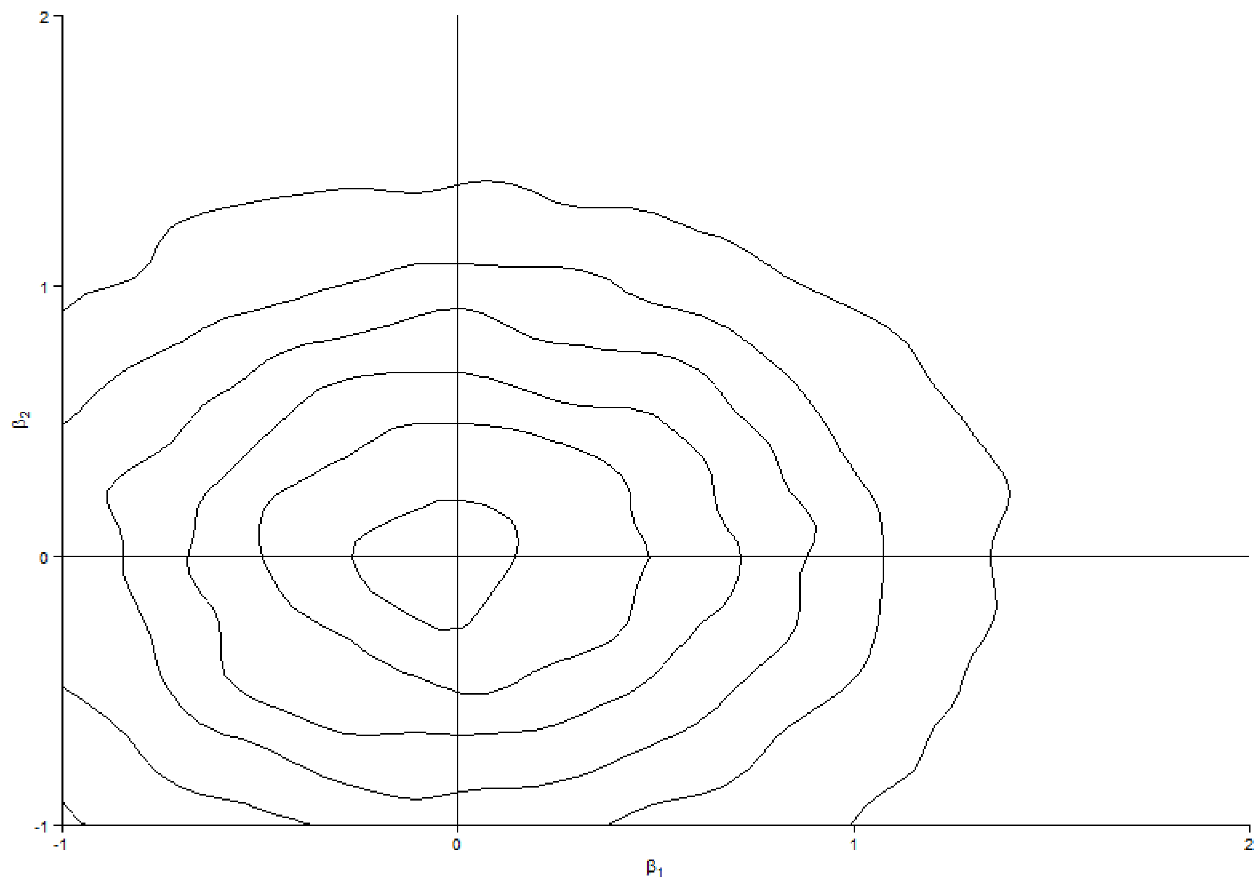
Bayesian logistic regression

- Parameter vector β can be treated as a random variable
  - Prior $\beta \sim \mathrm{N}(b, B)$

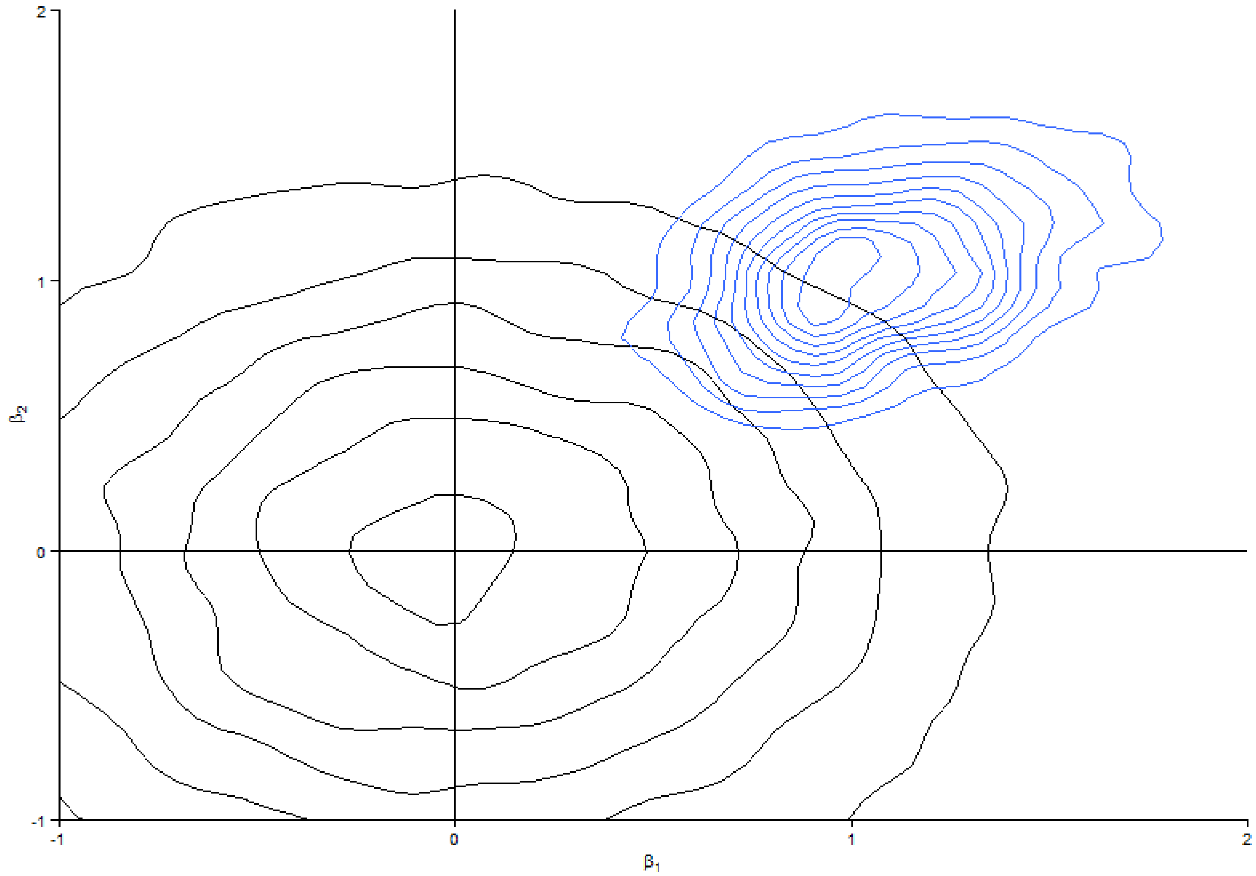
Bayesian logistic regression

- Parameter vector β can be treated as a random variable
    - Prior $\beta \sim \mathrm{N}(b, B)$
- However, no closed form for the posterior $\mathrm{p}(\beta|y)$

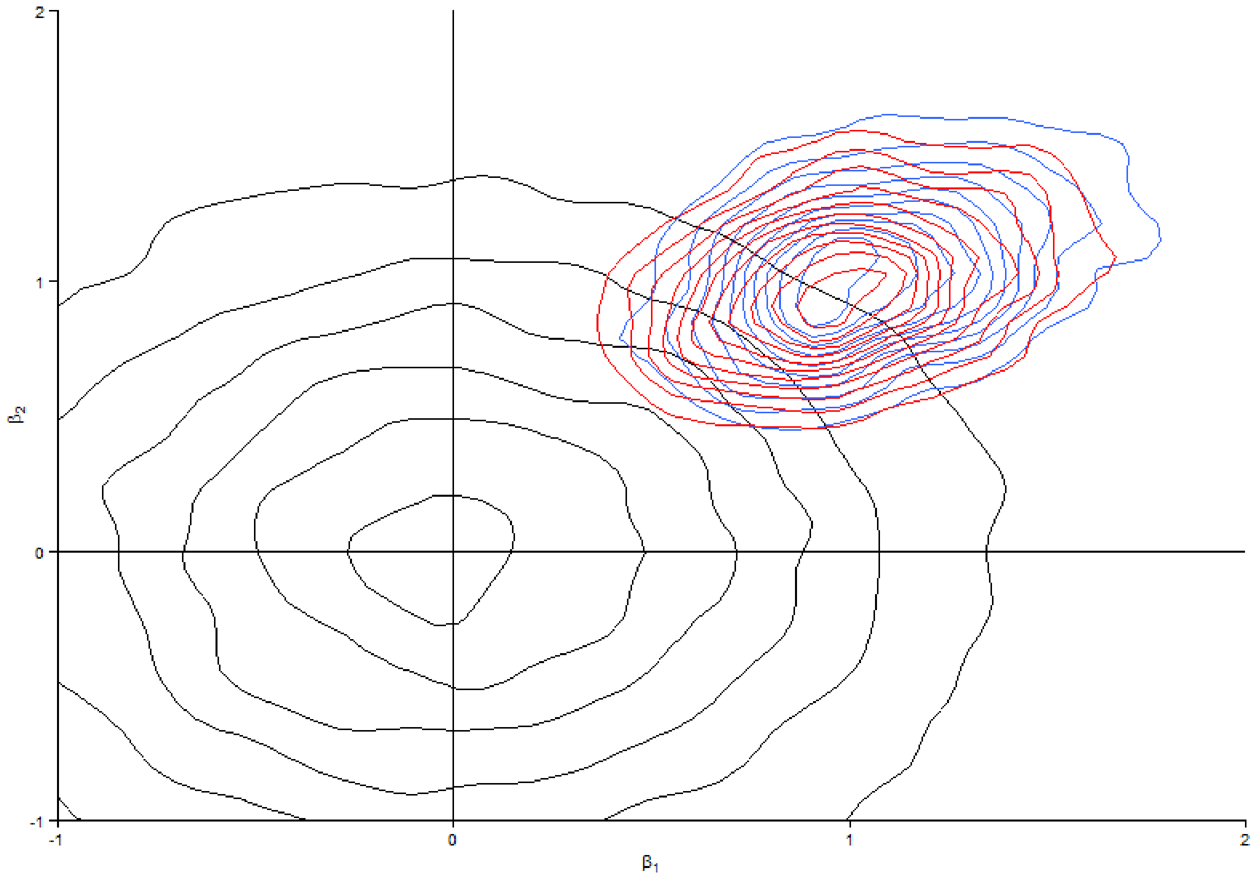
Prior



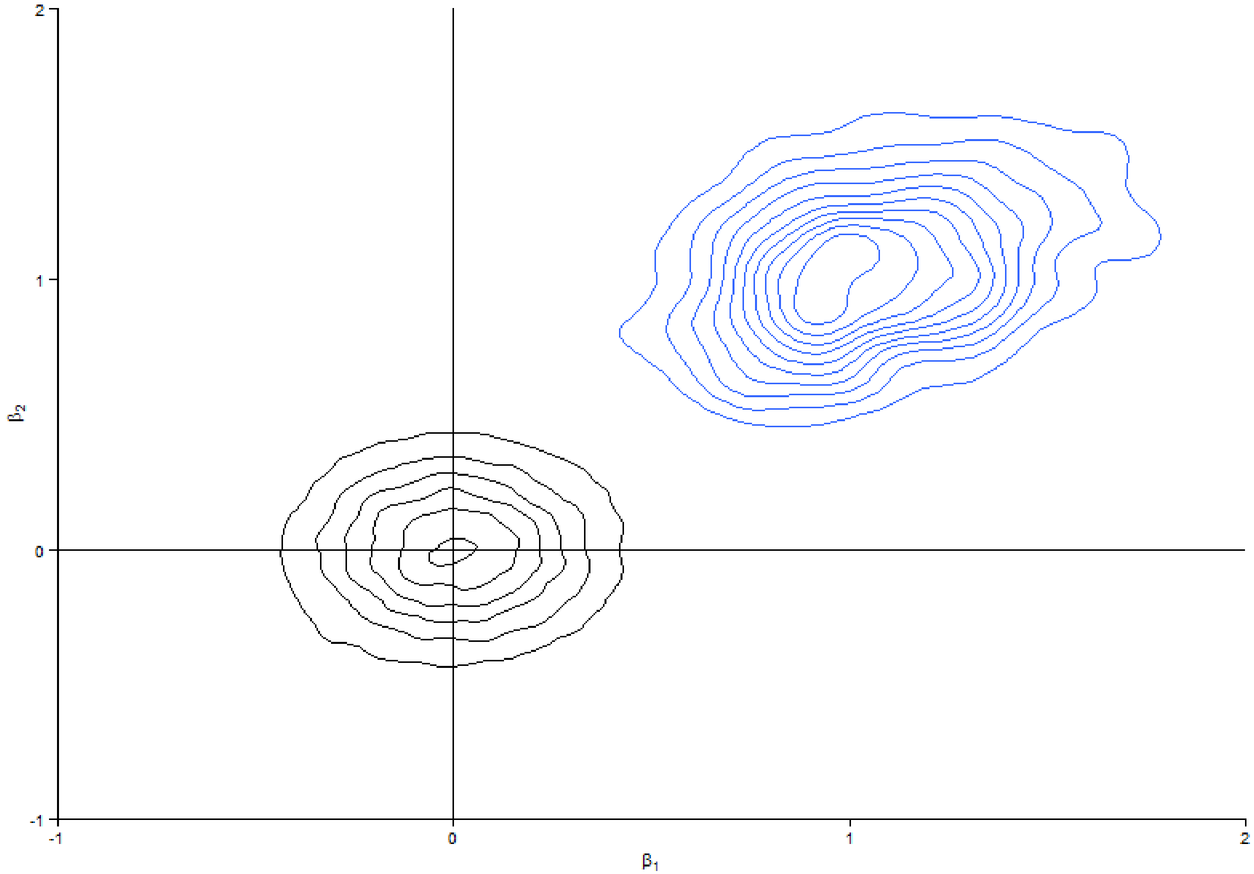
Prior, likelihood



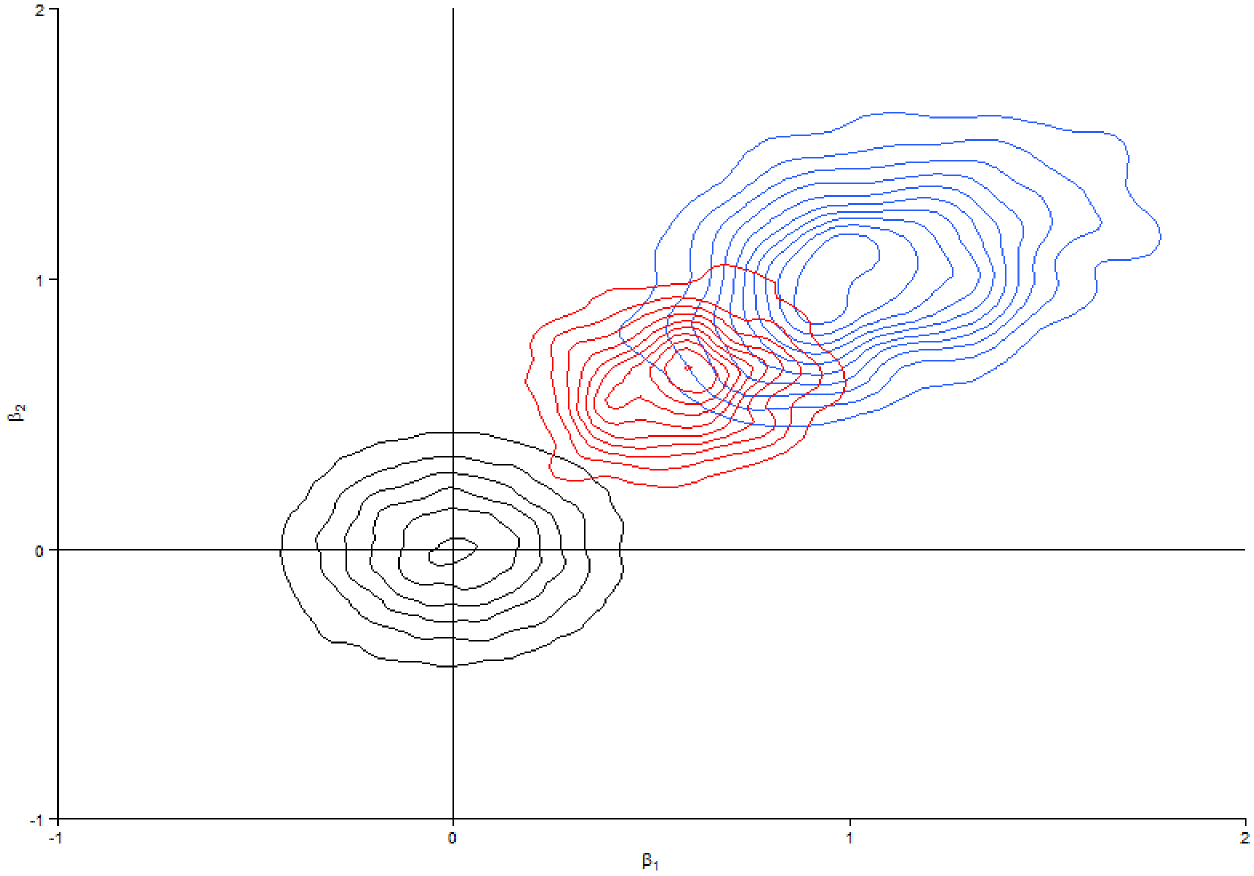
Prior, likelihood, posterior



Informative prior, likelihood



Informatave prior, likelihood, posterior



Data augmentation scheme

- Introduce latent variable ω such that
  - prior $p(\beta)$
  - posterior $p(\beta|y, \omega)$

would be conjugate distributions

Data augmentation scheme

- Introduce latent variable ω such that
  - prior $p(\beta)$
  - posterior $p(\beta|y, \omega)$

would be conjugate distributions

- It turns out that Polya-Gamma is a suitable distribution

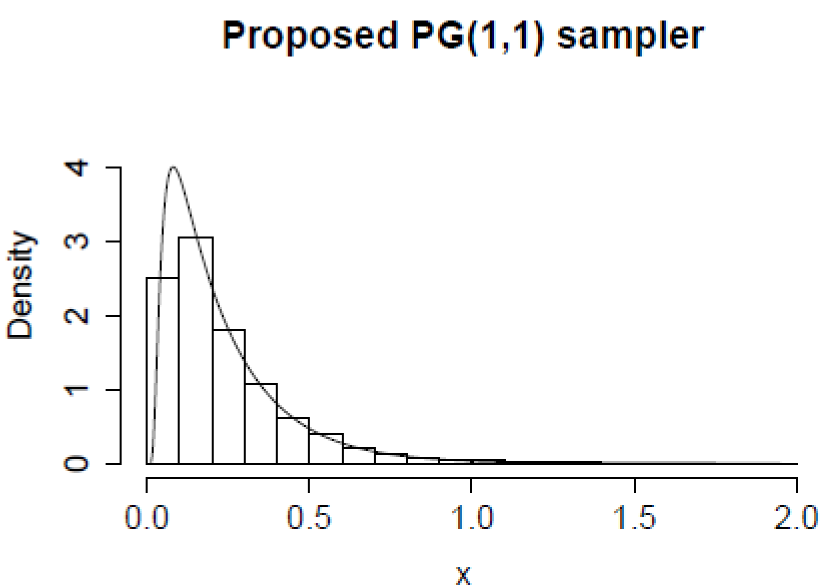
$$\omega_i|\beta \sim PG\left(1, x_i^T \beta\right) \qquad i = 1, \ldots, N$$

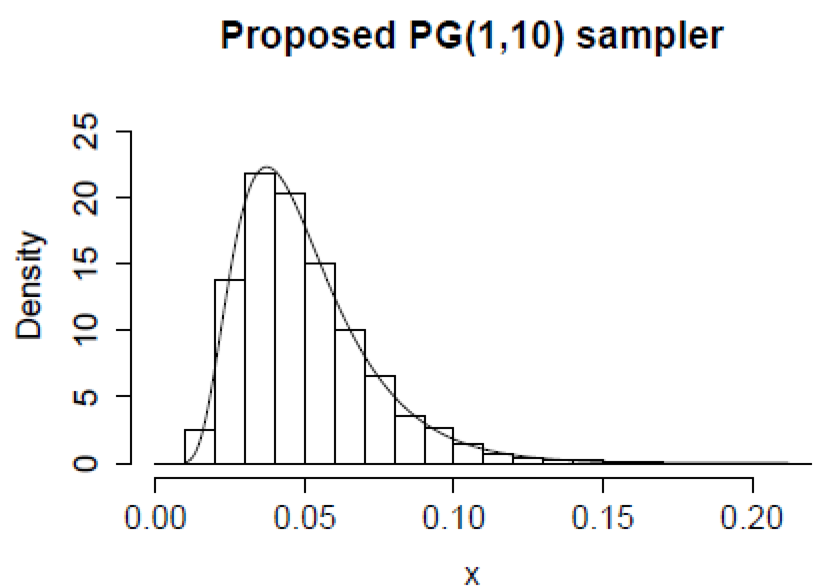
Polya-Gamma distribution PG(1, z)

$$\sum_{k=1}^{\infty} \frac{g_k}{(k-0.5)^2 + \frac{z^2}{4\pi^2}} \quad \text{where } \mathrm{g_k} \sim \Gamma(1,1)$$

Polya-Gamma distribution PG(1, z)

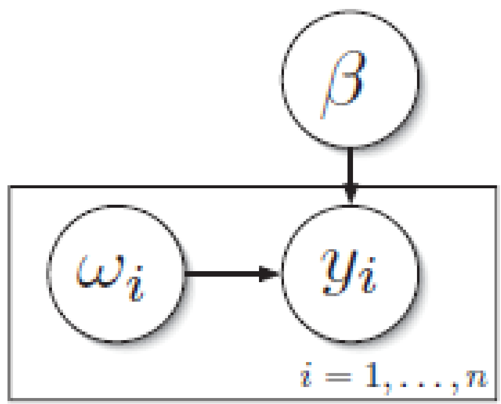
$$\sum_{k=1}^{\infty} \frac{g_k}{(k-0.5)^2 + \frac{z^2}{4\pi^2}} \quad \text{where } \mathrm{g_k} \sim \Gamma(1,1)$$





Gibbs sampling scheme

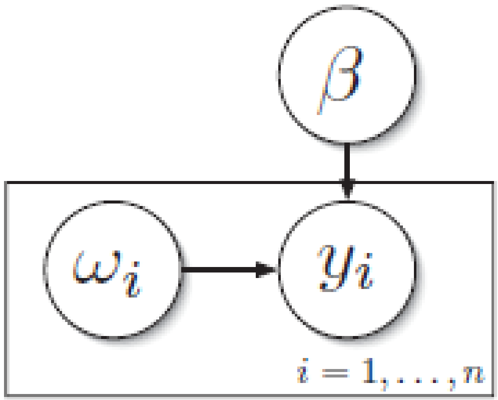
1. $\omega_i | \beta \sim \mathrm{PG}(1, x_i^T \beta)$
2. $\beta | y, \omega \sim \mathrm{N}(m_\omega, V_\omega)$



Gibbs sampling scheme

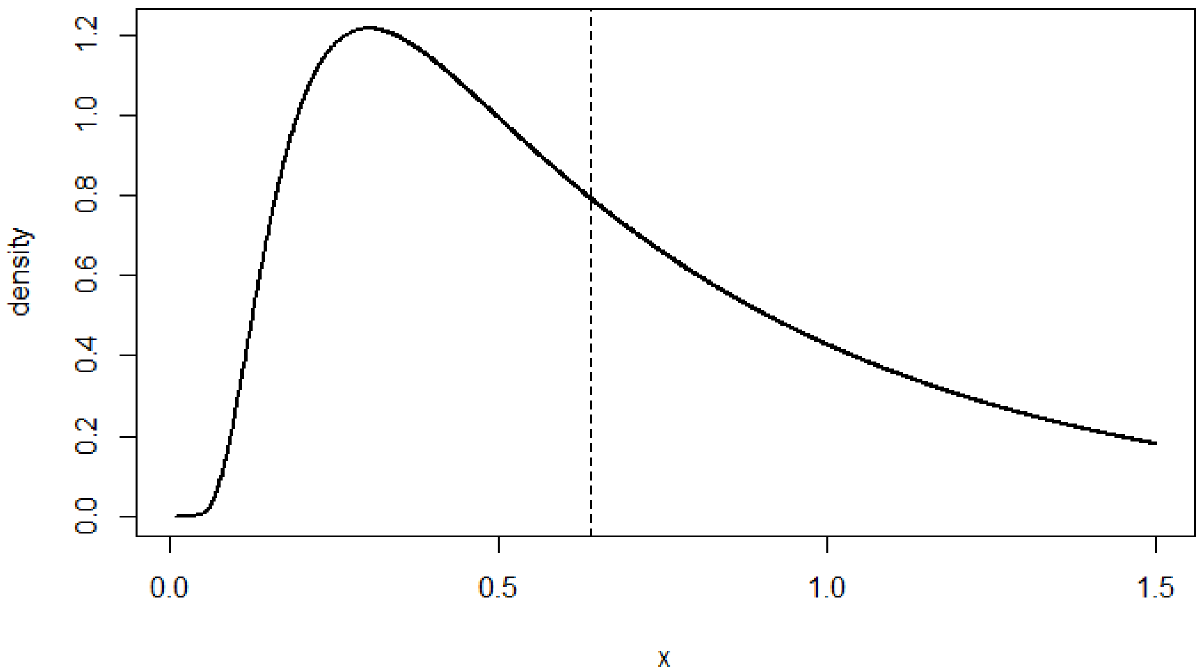
1. $\omega_i | \beta \sim \boxed{\mathrm{PG}\left(1, x_i^T \beta\right)}$
2. $\beta | y, \omega \sim \mathrm{N}(m_\omega, V_\omega)$

How to sample from this distribution?



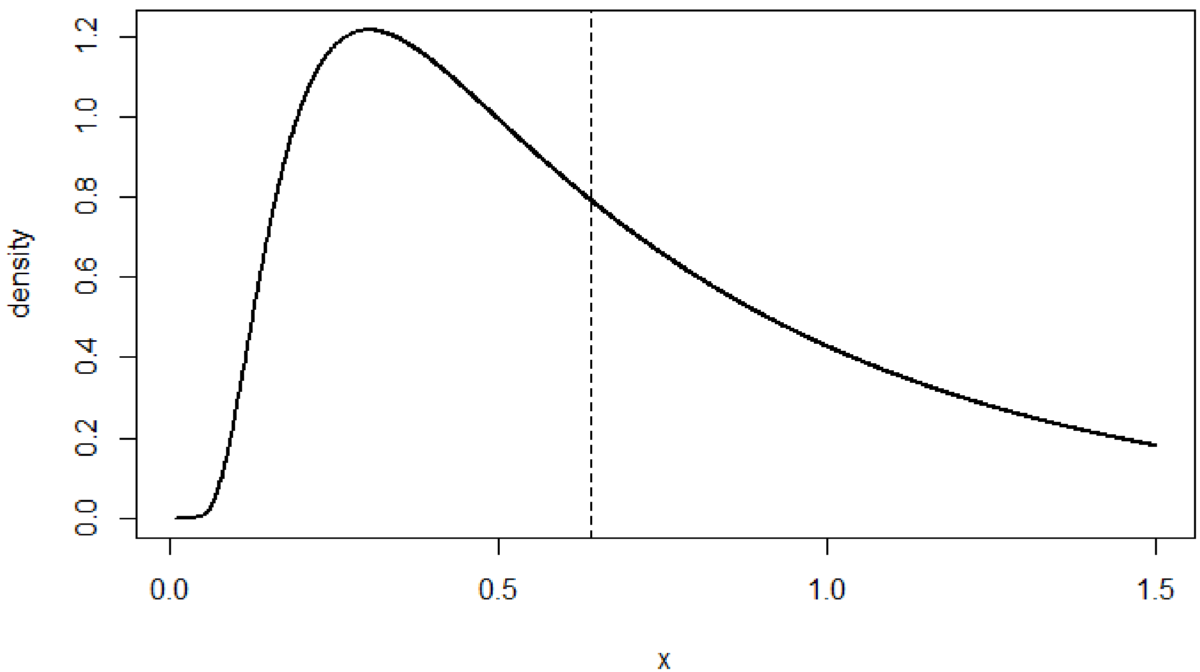
Polya-Gamma sampler

$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x)$$



Polya-Gamma sampler

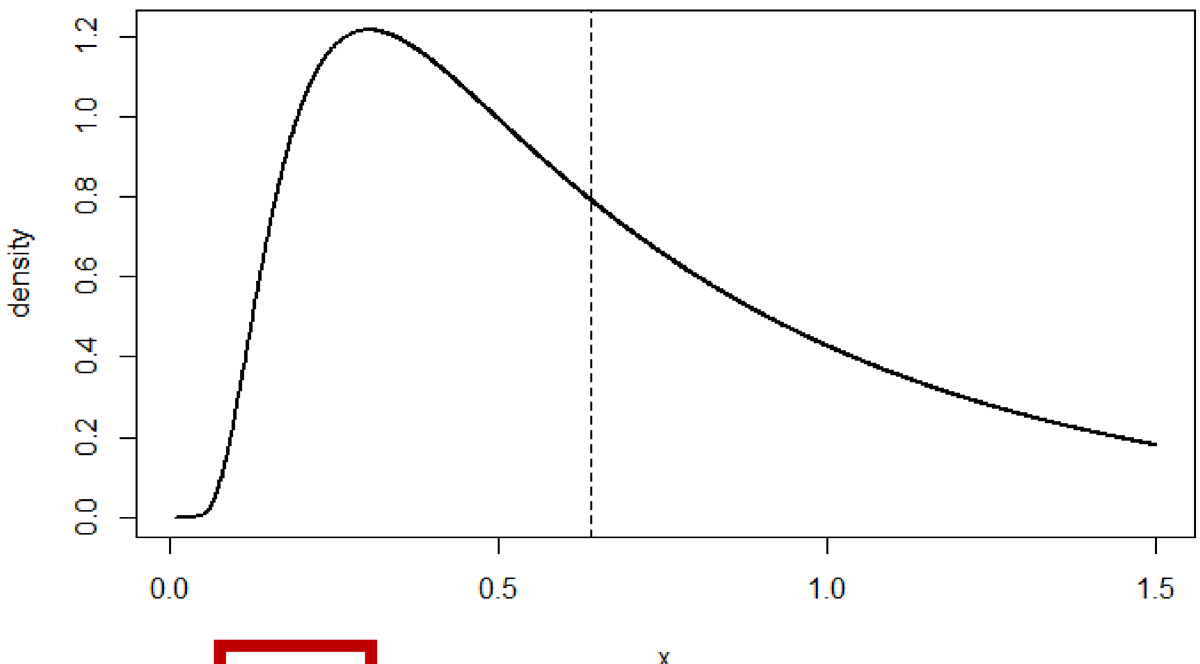
$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x) \qquad S_k(x) = \sum_{n=0}^{k} (-1)^n a_n(x)$$



$$S_0(x) > S_2(x) > \cdots > f(x) > \cdots > S_3(x) > S_1(x)$$

Polya-Gamma sampler

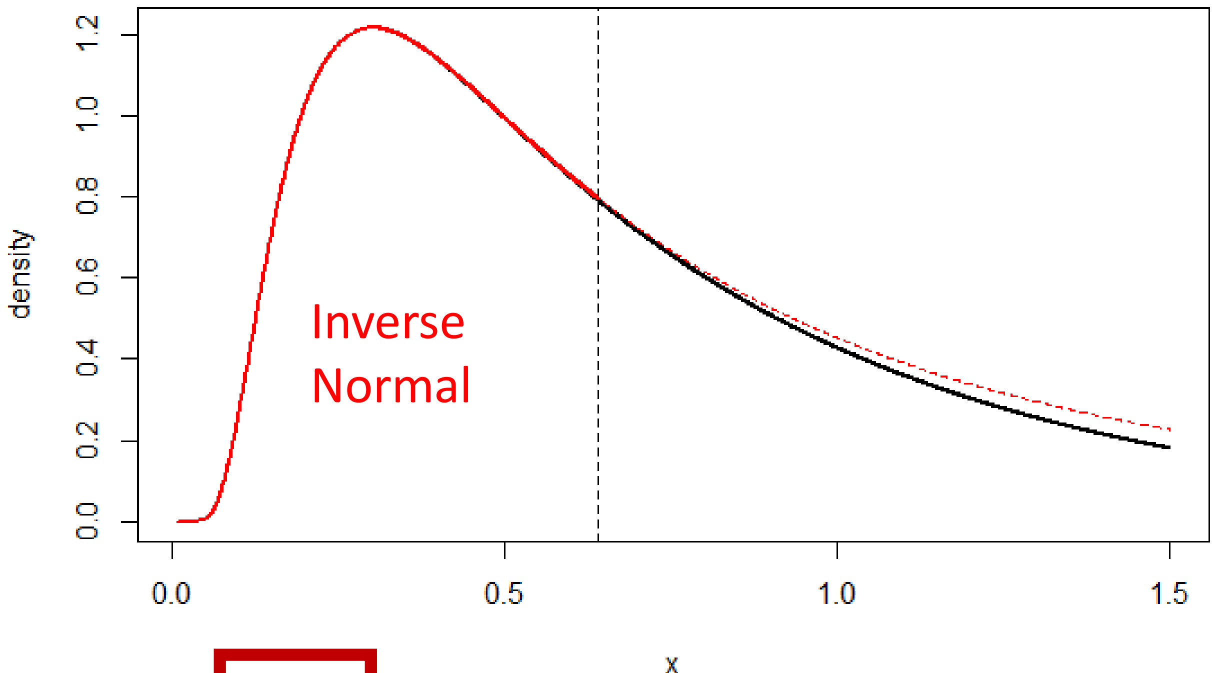
$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x) \qquad S_k(x) = \sum_{n=0}^{k} (-1)^n a_n(x)$$



proposal $\boxed{S_0(x)} > S_2(x) > \cdots > f(x) > \cdots > S_3(x) > S_1(x)$

Polya-Gamma sampler

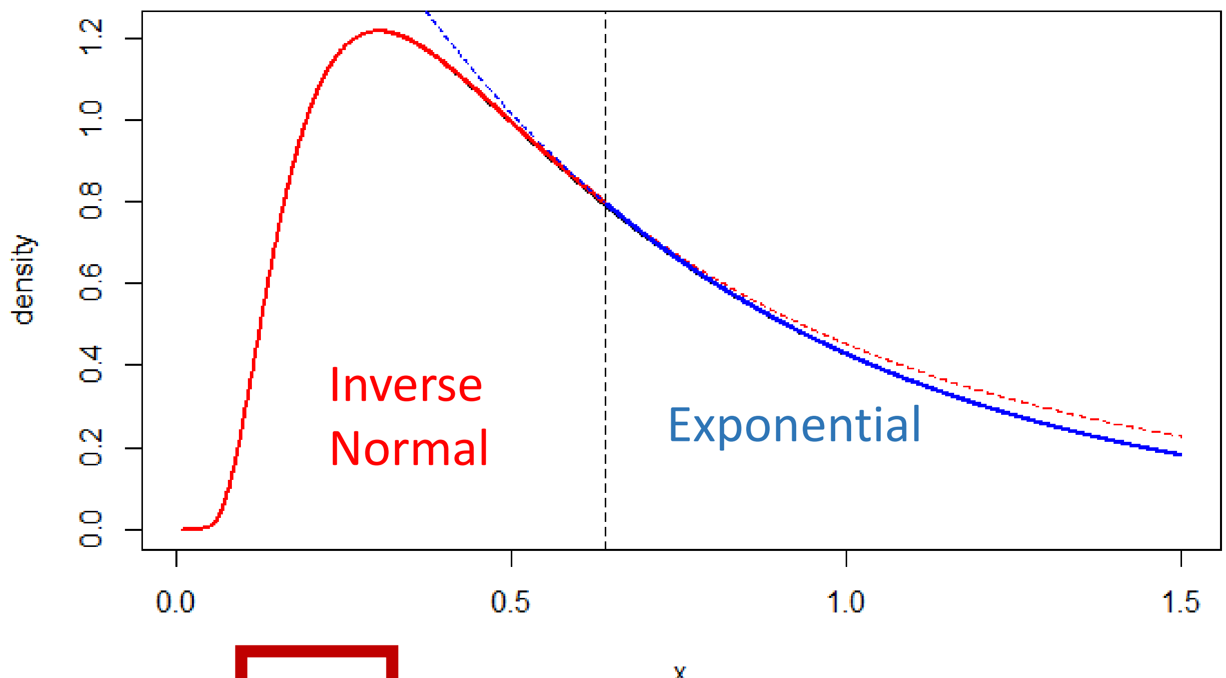
$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x) \qquad S_k(x) = \sum_{n=0}^{k} (-1)^n a_n(x)$$



proposal $\boxed{S_0(x)} > S_2(x) > \cdots > f(x) > \cdots > S_3(x) > S_1(x)$

Polya-Gamma sampler

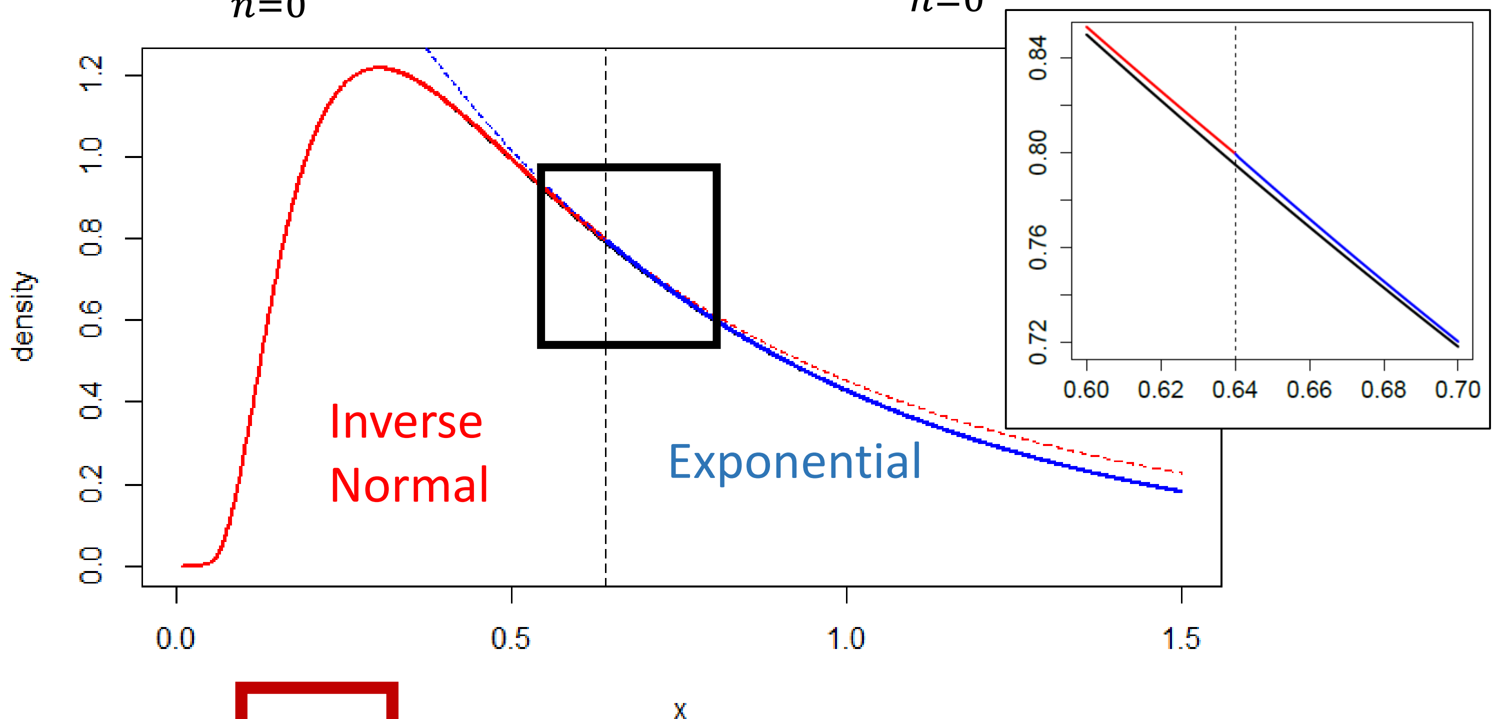
$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x) \qquad S_k(x) = \sum_{n=0}^{k} (-1)^n a_n(x)$$



proposal $\boxed{S_0(x)} > S_2(x) > \cdots > f(x) > \cdots > S_3(x) > S_1(x)$

Polya-Gamma sampler

$$f(x) = \sum_{n=0}^{\infty} (-1)^n a_n(x) \qquad S_k(x) = \sum_{n=0}^{k} (-1)^n a_n(x)$$



proposal $\boxed{S_0(x)} > S_2(x) > \cdots > f(x) > \cdots > S_3(x) > S_1(x)$

Our implementation

GitHub, Inc. [US] https://github.com/kasparmartens/PolyaGamma

PolyaGamma

An R package for Bayesian logistic regression. The posterior distribution of the parameters is obtained via Gibbs sampling using Polya-Gamma latent variables (see paper "Bayesian Inference for Logistic Models Using Pólya–Gamma Latent Variables" for details).

This package can be installed as follows:

```
devtools::install_github("kasparmartens/PolyaGamma")
```

Our implementation

GitHub, Inc. [US] https://github.com/kasparmartens/PolyaGamma

PolyaGamma

An R package for Bayesian logistic regression. The posterior distribution of the parameters is obtained via Gibbs sampling using Polya-Gamma latent variables (see paper "Bayesian Inference for Logistic Models Using Pólya–Gamma Latent Variables" for details).

This package can be installed as follows:

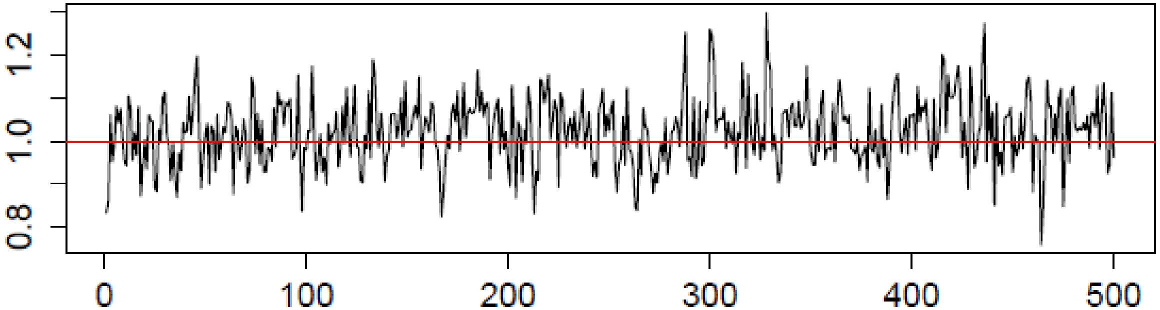
```
devtools::install_github("kasparmartens/PolyaGamma")
```

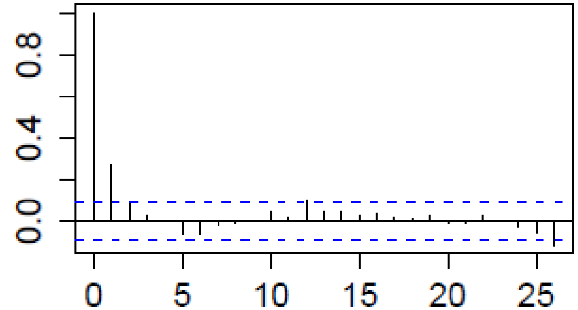
Gibbs sampling

1. $\omega_i | \beta \sim \mathrm{PG}(1, x_i^T \beta)$
2. $\beta | y, \omega \sim \mathrm{N}(m_\omega, V_\omega)$

Tests on simulated data

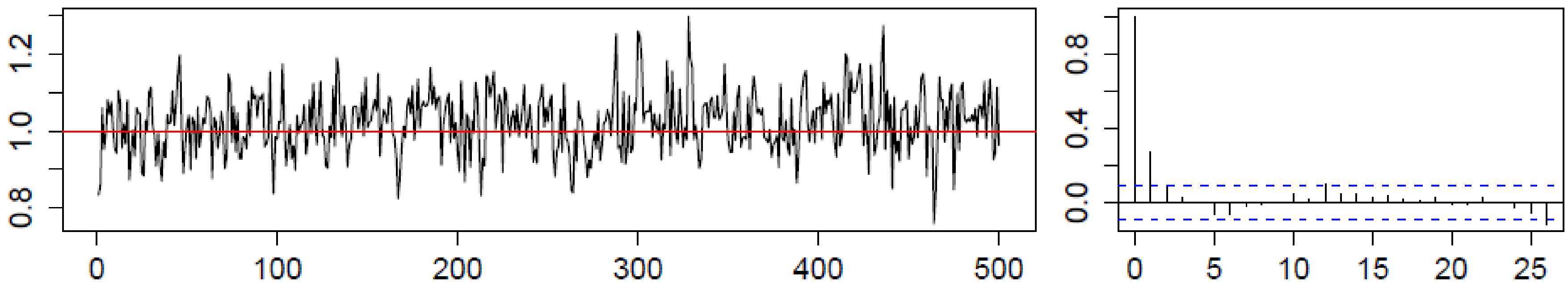
- Generated data with $\beta = (1, 1)^T$

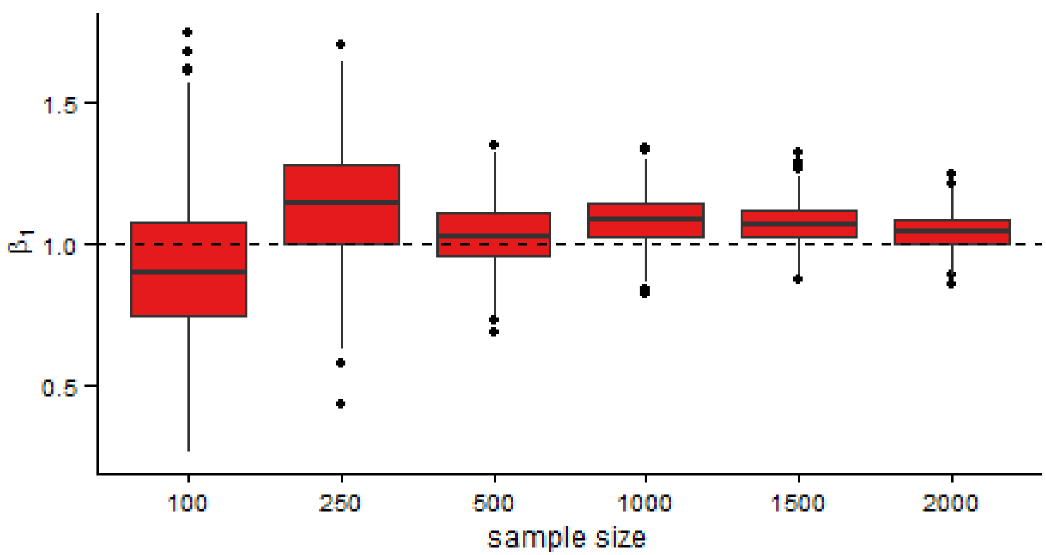




Tests on simulated data

- Generated data with $\beta = (1, 1)^T$





Tests on real data

- We set prior $\beta \sim N\left(0, \frac{1}{\lambda} I\right)$ and varied prior precision λ

- Test error is comparable to classical logistic regression