

Gibbs Sampling with Data Augmentation for Bayesian Analysis of Binary and Polychotomous Response Data

STAT 230 Final Report

Damon Bayer & Corey Katz

12/16/2019

Contents

1	Abstract	1
2	Introduction	1
3	Methodology	1
3.1	Probit Model	2
3.2	T-link to Approximate Logistic Regression	3
3.3	Multinomial	4
3.4	Implementation in Stan (NUTS)	4
4	Results	4
4.1	Small-Cell Carcinoma	5
4.2	Breast Cancer Data	8
4.3	Baseball	11
5	Discussion	14
6	References	15
7	Appendix	16

1 Abstract

Here's the abstract.

2 Introduction

We cite Albert and Chib (1993)

Here's the introduction.

3 Methodology

In this paper, we implement data-augmentated Gibbs sampling for probit regression on binary response data, multinomial probit regression on ordinal response data, and an extension of the probit regression model, the t-link regression model for binary response data (Albert and Chib 1993). These methods are used to generate samples from posterior distributions for Bayesian inference on regression coefficients. In this section, we detail each method, including pseudocode for its algorithm.

3.1 Probit Model

The first model that data augmentation can be used to find posterior distributions is the probit model. The following is a standard Bayesian Probit model with non informative priors on the regression coefficients:

$$y_i | \pi_i, \vec{\beta}, \vec{x}_i \sim \text{Bernoulli}(\pi_i) \quad \text{for } i = 0, \dots, n \quad (1)$$

$$\pi_i = \Phi(x_i^T \beta) \quad (2)$$

$$\beta_j \sim N(0, 100), \quad \text{for } j = 0, \dots, p \quad (3)$$

Note that Φ is the cumulative density function of the normal distribution.

The idea of data augmentation is to introduce a continuous latent variable derived from a binary response in order to make sampling from the posterior distributions of the probit model coefficients easier. By introducing independent latent variables Z_1, \dots, Z_n , we can now find the joint posterior of $\vec{\beta}$ and \vec{Z} given Y . We can then marginalize over the posterior distribution of $\vec{Z} | \vec{Y}$ and thus we have the conditional posterior of $\beta | \vec{Z}, \vec{Y}$. This method is further simplified if we assume (as we did above) non-informative priors on the regression coefficients, $\vec{\beta}$ (Albert and Chib 1993).

With this approach we are able to create a Gibbs sampler that only needs to sample from truncated normal distributions and multivariate normal distribution, which are extremely easy with today's computing power. Based on the fully conditional posterior of $\beta | \vec{Z}, \vec{Y}$, Albert and Chib equated this method of probit regression on binary Y to doing linear regression on the latent variable Z . This will be evident in the algorithm as we use the `lm` function to find the mean of the posterior distribution of $\beta | \vec{Z}, \vec{Y}$ (Albert and Chib 1993).

3.1.1 Algorithm:

	Input : \vec{Y}, \mathbf{X} Output : Posterior Samples of Regression Coefficients, $\vec{\beta}$
1	Set Number of Samples (Total (N_s) and Burn-in) Initialize $\vec{\beta}^{(0)}$
2	Set $\Sigma = (X^T X)^{-1}$ n = Number of Observations
3	for $k = 1$ to N_s do
4	for $i = 1$ to n do
5	if $y_i = 1$ then
6	Sample $z_i^{(k)}$ from $\text{trunc}\mathcal{N}(x_i^T \beta^{(k-1)}, 1, 0, \infty)$
7	else
8	Sample $z_i^{(k)}$ from $\text{trunc}\mathcal{N}(x_i^T \beta^{(k-1)}, 1, -\infty, 0)$
9	end
10	end
11	Regress \vec{Z} onto \mathbf{X} to find $\vec{\beta}_Z$
12	Sample $\beta^{(k)} \vec{Z}$ from $\mathcal{N}(\hat{\vec{\beta}}_Z^{(k)}, \Sigma)$
13	end

Algorithm 1: Probit Regression Using Gibbs Sampler with Data Augmentation

3.2 T-link to Approximate Logistic Regression

In this section we will discuss an extension of the Gibbs sampler discussed for the probit model. The purpose of this model is to use data augmentation with t-distributions to approximate the logistic regression model for a binary response. The simple Bayesian Logistic Regression model with non-informative priors is as follows:

$$y_i | \pi_i, \vec{\beta}, \vec{x}_i \sim \text{Bernoulli}(\pi_i) \quad \text{for } i = 0, \dots, n \quad (4)$$

$$\text{logit}(\pi_i) = x_i^T \beta \quad (5)$$

$$\beta_j \sim N(0, 100), \quad \text{for } j = 0, \dots, p \quad (6)$$

Where $\text{logit}(p) = \frac{p}{1-p}$

The T-link is an extension of the probit regression model because instead of using the normal cdf (as in the probit model) we use the t-distribution cdf as our link function. If we replace Φ with the cdf of the $t(\nu)$ in (2), we would have Bayesian t-link model. By generalizing the model, we can now choose a degrees of freedom, ν where $t(\nu)$ better fits our model and thus a link function that approximates other well-known link functions. For example, we are usually concerned with logistic regression because of the interpretability of the coefficients into odds ratios. The flexibility of the t-link function allows us to draw from a posterior distribution of $\vec{\beta}$ that is approximately the posterior distribution if we had chosen to use the logit link function. Note that if we set the degrees of freedom equal to infinity, we would revert back to the probit model (Albert and Chib 1993).

According to Albert and Chib, a t-distribution with a degrees of freedom of 8 is a fairly close approximation of the logistic regression model, once a correction factor of 0.634 is taken into account (Albert and Chib 1993).

To implement this link function, we must introduce a second set of latent variables, λ_i . This variable is introduced to simulate the heavier tails of the t-distribution compared to the normal distribution. Our implementation is slightly different from Albert and Chib's because of the ability to sample from a truncated t distribution and issues implementing their algorithm. We still need to simulate λ in order to sample from the conditional posterior of $\beta^{(k)} | \vec{Z}, \vec{Y}, \lambda, \nu$. We took the degrees of freedom equal to 8 to compare results to the logistic regression model. You could sample from the posterior distribution of ν and better understand how well your model fits the data, but we choose to focus on logistic regression.

3.2.1 Algorithm:

```

Input :  $\vec{Y}, \mathbf{X}$ 
Output: Posterior Samples of Regression Coefficients,  $\vec{\beta}$ 
1 Set Number of Samples (Total ( $N_s$ ) and Burn-in)
2  $n$  = Number of Observations
3 Initialize  $\vec{\beta}^{(0)}$  and set  $\lambda = \vec{1}_n$ 
4  $\nu = 8$ 
5 for  $k = 1$  to  $N_s$  do
6   for  $i = 1$  to  $n$  do
7     if  $y_i = 1$  then
8       | Sample  $z_i^{(k)}$  from  $\text{trunc}\mathcal{N}(x_i^T \beta^{(k-1)}, \lambda_i^{-1}, 0, \infty)$ 
9     else
10      | Sample  $z_i^{(k)}$  from  $\text{trunc}\mathcal{N}(x_i^T \beta^{(k-1)}, \lambda_i^{-1}, -\infty, 0)$ 
11    end
12  end
13   $\mathbf{W} = \text{diag}(\vec{\lambda})$ 
14  Set  $\Sigma = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1}$ 
15   $\vec{\beta}_Z = \Sigma \mathbf{X}^T \mathbf{W} \mathbf{Z}$ 
16  Sample  $\beta^{(k)} | \vec{Z}, \vec{\lambda}, \vec{Y} \nu$  from  $\mathcal{N}(\hat{\vec{\beta}}^{(k)}, \Sigma)$ 
17  Sample  $\lambda_i$  from  $\Gamma\left(\frac{\nu+1}{2}, \frac{\nu + (Z_i - x_i^T \beta)^2}{2}\right)$ 
18 end

```

Algorithm 2: Tobit Regression Using Gibbs Sampler with Data Augmentation ($\nu = 8$)

3.3 Multinomial

The previous two data augmentation Gibbs samplers tackled regression models where the response is binary. Data augmentation Gibbs samplers can also be applied to multinomial response variables. Although there are approaches for both ordered and unordered categories, we will focus on the case of ordered categories. Once again, using a probit model will simplify the conditional posterior distributions. Before discussing the algorithm in depth, we present the ordered multinomial probit model for which we will use Gibbs Sampling to fit. This model

3.3.1 Algorithm:

3.4 Implementation in Stan (NUTS)

4 Results

For three data sets, we apply the appropriate previously discussed model as well as standard maximum likelihood estimation and custom models written in Stan (Carpenter et al. 2017) and implemented in the RStan package (Stan Development Team 2019).

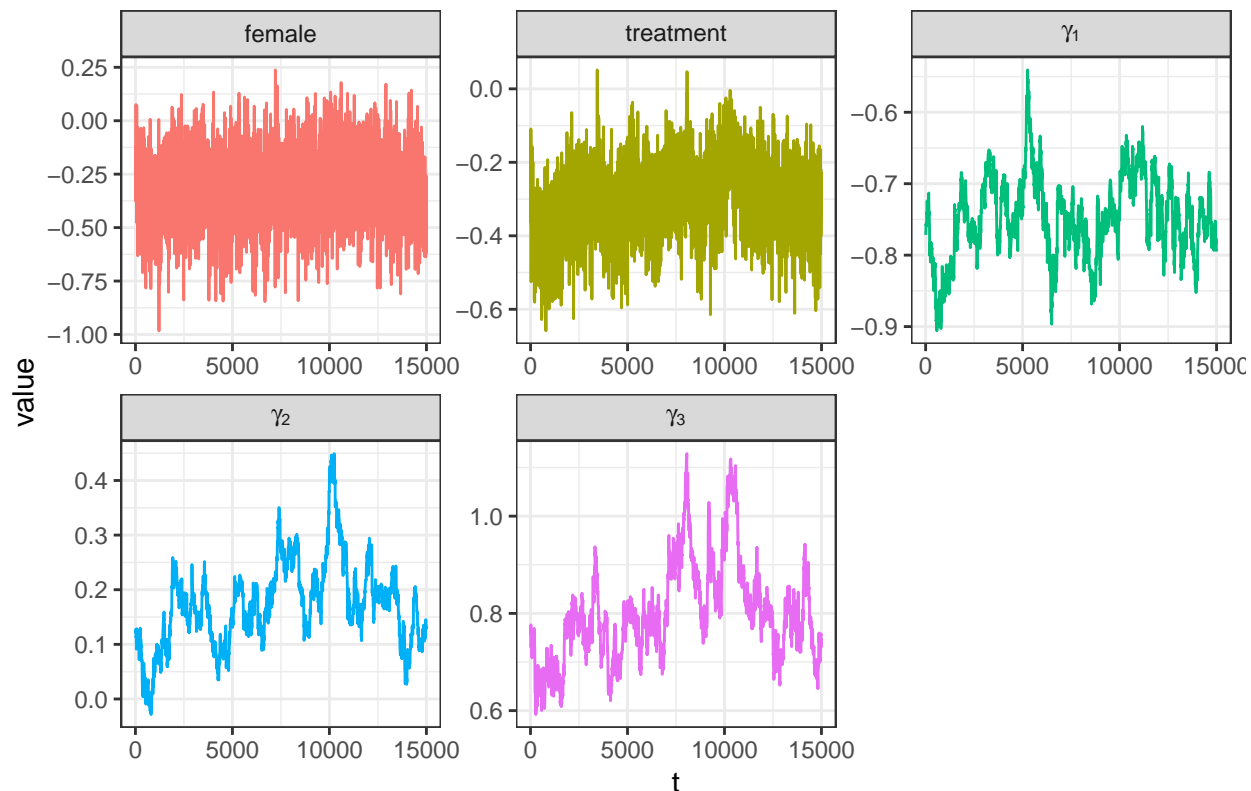
4.1 Small-Cell Carcinoma

The small-cell carcinoma data comes from our STAT 211 class. Small-cell carcinoma of the lung is an aggressive cancer that can be treated with chemotherapy. Patients with small-cell carcinoma were randomly assigned to one of two therapy options. The outcome of the therapy is recorded on an ordinal scale (1: Progressive, 2: No Change, 3: Partial Remission, 4: Complete Remission). We fit multinomial ordered probit regression models with with therapy option and sex as predictors of outcome. The model parameters are estimated with data augmented Gibbs sampling following the algorithm described in Albert and Chib (1993) and implemented in R, a custom model built in Stan, and traditional maximum likelihood estimation as implemented in the `polr` function in the MASS package (Venables and Ripley 2002). Both Bayesian methods are run to generate 4000 posterior samples, with the first 1000 discarded as burn-in samples, leaving 3000 samples for analysis. The Stan model took 1 minute and 13,9 seconds seconds to run, while the R model completed in 2.4 seconds.

Table 1: Posterior sample summaries for small-cell carcinoma data using R-Gibbs

Variable	mean	sd	2.5%	50%	97.5%
female	-0.3177006	0.1695306	-0.6483304	-0.3154434	0.0143534
treatment	-0.3081612	0.1039444	-0.5136893	-0.3092713	-0.1059862
gamma[1]	-0.7478856	0.0569057	-0.8576571	-0.7492453	-0.6420117
gamma[2]	0.1726966	0.0795173	0.0235177	0.1732758	0.3510630
gamma[3]	0.8011110	0.1027040	0.6390310	0.7890847	1.0615511

Posterior Trace (Small-Cell Carcinoma Using R-Gibbs)



Posterior Distribution (Small-Cell Carcinoma Using R-Gibbs)

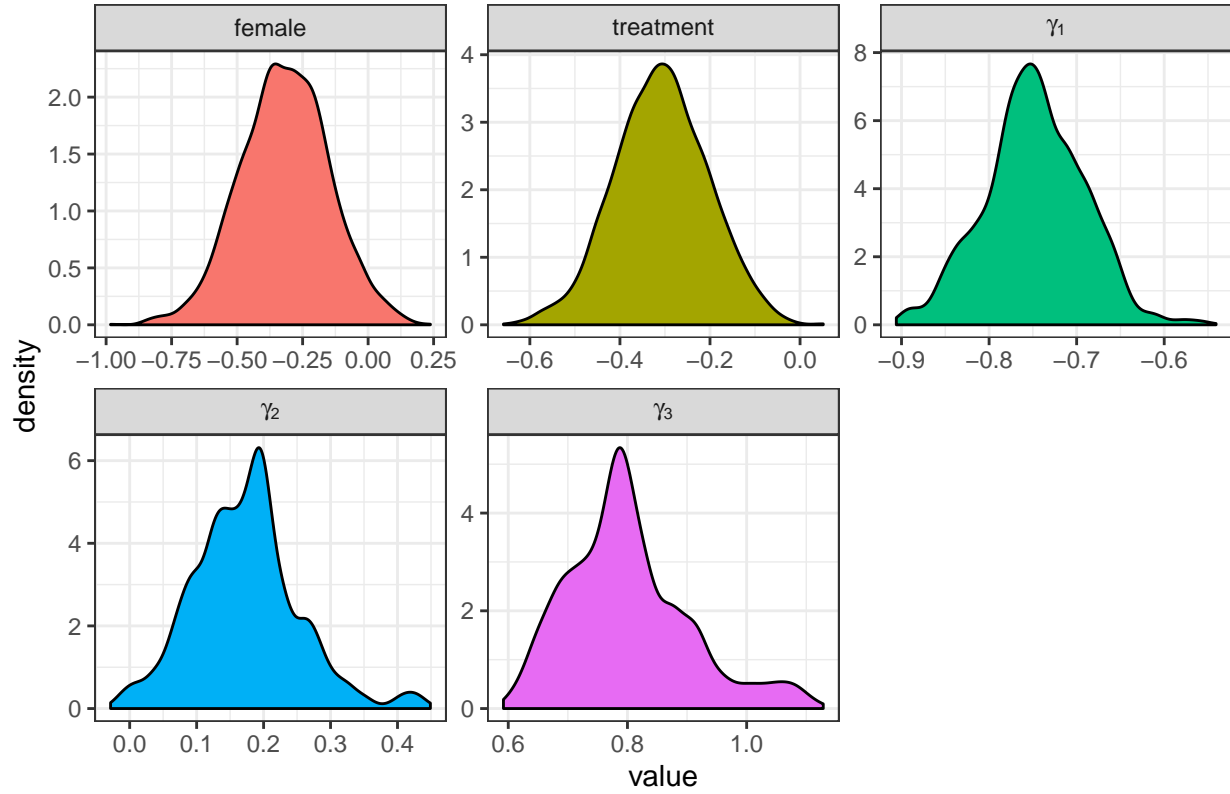
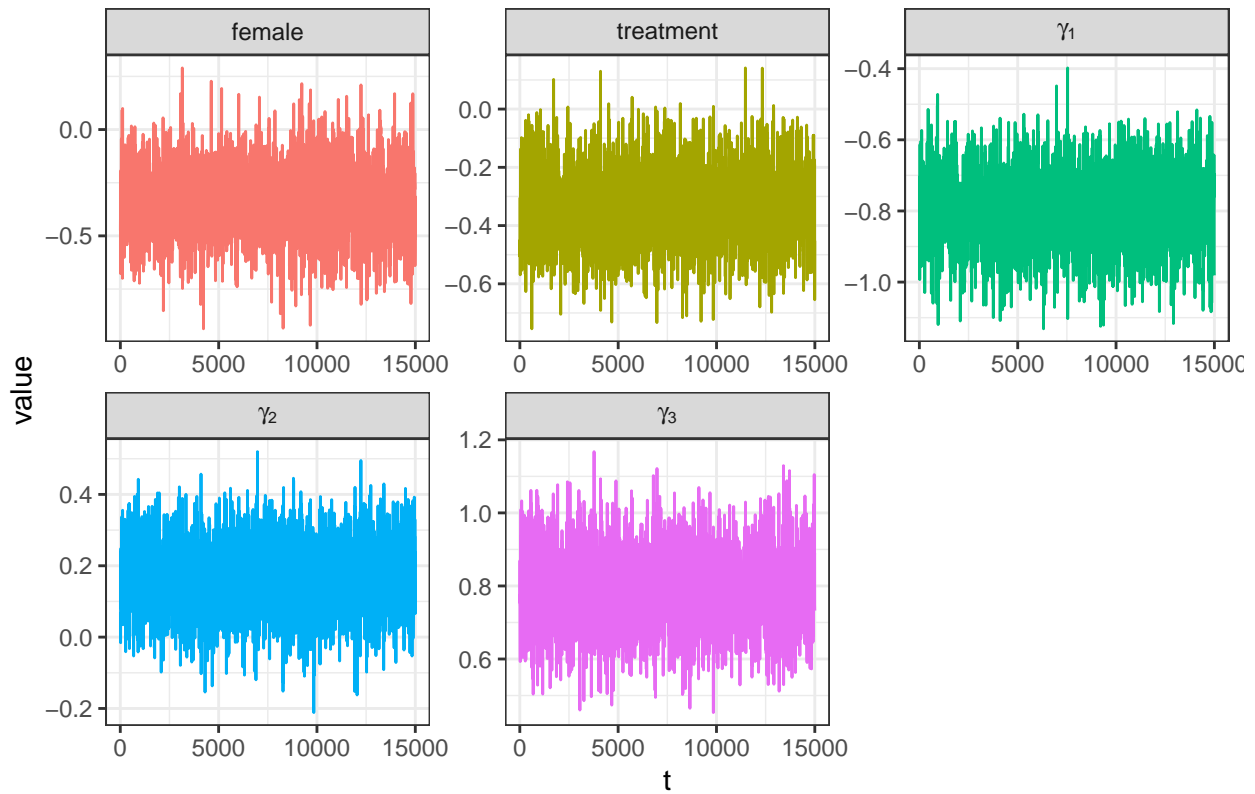


Table 2: Posterior sample summaries for small-cell carcinoma data using Stan

Variable	mean	sd	2.5%	50%	97.5%
female	-0.3383656	0.1702371	-0.6777031	-0.3369185	-0.0045383
treatment	-0.3311899	0.1282283	-0.5829356	-0.3309320	-0.0730097
gamma[1]	-0.7992210	0.1048007	-1.0079013	-0.7988986	-0.5917817
gamma[2]	0.1684474	0.1002620	-0.0302685	0.1692335	0.3625867
gamma[3]	0.7914654	0.1058342	0.5811806	0.7903738	1.0012122

Posterior Trace (Small-Cell Carcinoma Using Stan)



Posterior Distribution (Small-Cell Carcinoma Using Stan)

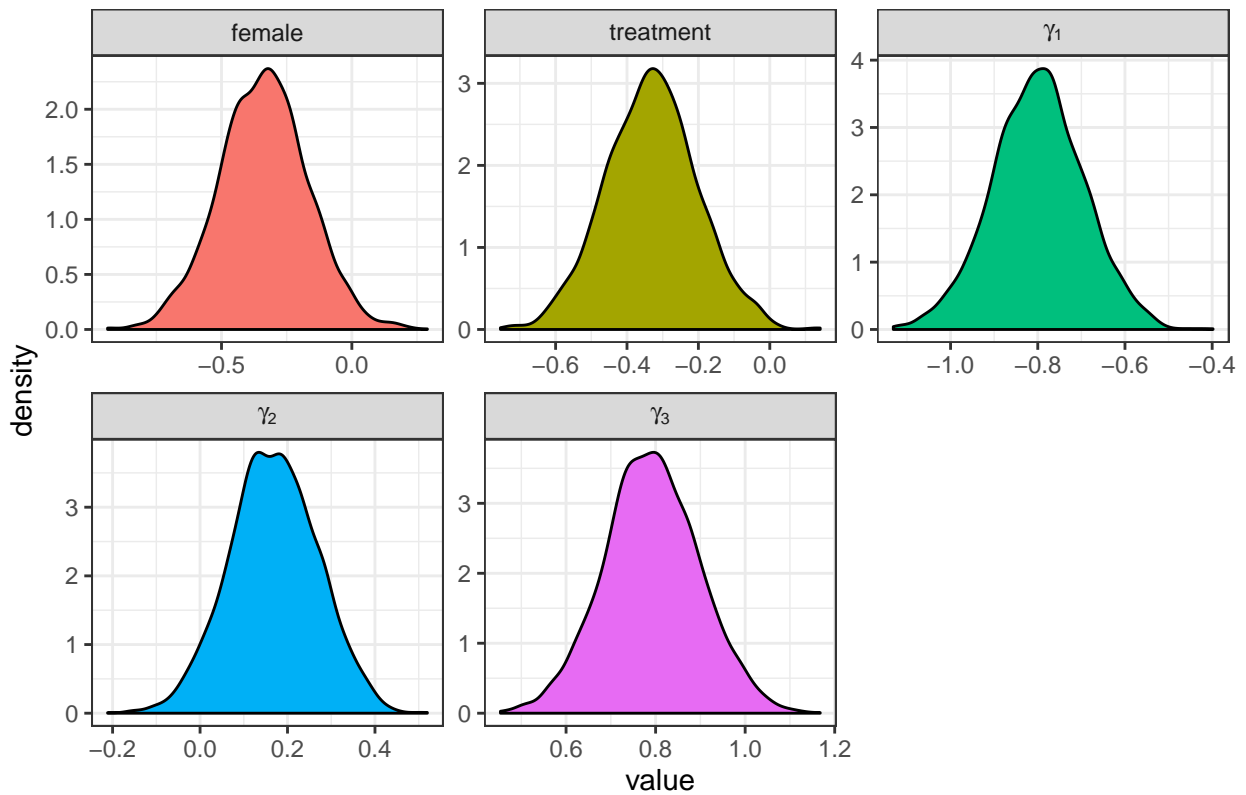


Table 3: MLE estimates for small-cell carcinoma data

Variable	Estimate	Std. Error
female	-0.3401606	0.1749021
treatment	-0.3344764	0.1254351
gamma[1]	-0.7994983	0.1053802
gamma[2]	0.1649279	0.0988553
gamma[3]	0.7818721	0.1068102

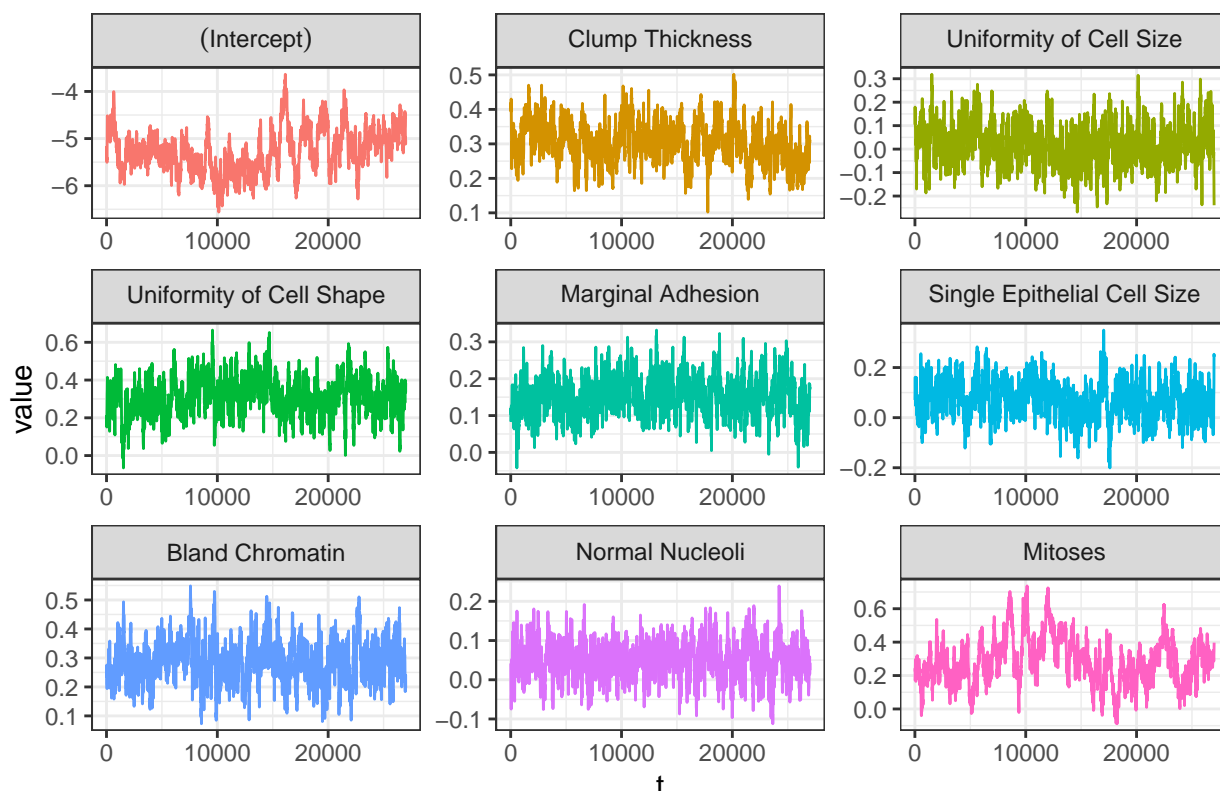
4.2 Breast Cancer Data

We evaluate the probit regression methods with the Wisconsin Breast Cancer dataset, obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg (Wolberg and Mangasarian 1990). The data reports 9 discrete measurements for 699 observations of clumps of breast cancer cells as well as the response variable, indicating whether or not the clump is malignant (1) or benign (0). The variables are Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, and Mitoses. Only the Bare Nuclei variable included any missing data, so it was not included in this analysis. The model parameters are estimated with data-augmented Gibbs sampling following the algorithm described in Albert and Chib (1993) and implemented in R, a custom model built in Stan, and traditional maximum likelihood estimation as implemented in the `glm` function in R (R Core Team 2019). Both Bayesian methods are run to generate 4000 posterior samples, with the first 1000 discarded as burn-in samples, leaving 3000 samples for analysis. The Stan model took 6 minutes and 10.5 seconds to run, while the R model completed in 7.8 seconds.

Table 4: Posterior sample summaries for breast cancer data using R-Gibbs

Variable	mean	sd	2.5%	50%	97.5%
(Intercept)	-5.2657995	0.4516886	-6.0869080	-5.2781142	-4.3524946
Clump Thickness	0.3030510	0.0570216	0.1891526	0.3017604	0.4131211
Uniformity of Cell Size	0.0229394	0.0891064	-0.1458150	0.0219254	0.2035911
Uniformity of Cell Shape	0.3118907	0.1020941	0.1119921	0.3143989	0.5132866
Marginal Adhesion	0.1532512	0.0538371	0.0546572	0.1515937	0.2611788
Single Epithelial Cell Size	0.0733985	0.0731480	-0.0708377	0.0732059	0.2141101
Bland Chromatin	0.2863431	0.0725573	0.1497924	0.2856310	0.4414197
Normal Nucleoli	0.0509178	0.0490190	-0.0488792	0.0528914	0.1421364
Mitoses	0.2845107	0.1376854	0.0269171	0.2745250	0.5957231

Posterior Trace (Breast Cancer Using R-Gibbs)



Posterior Distribution (Breast Cancer Using R-Gibbs)

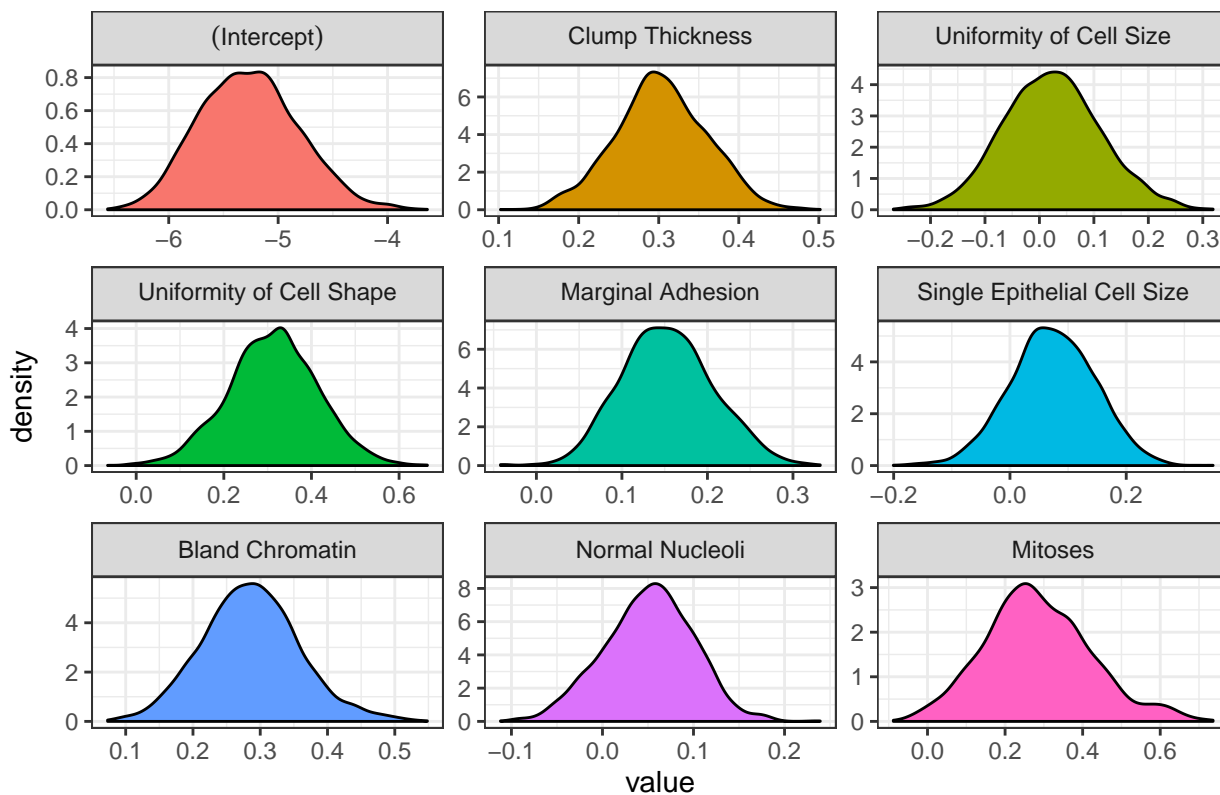
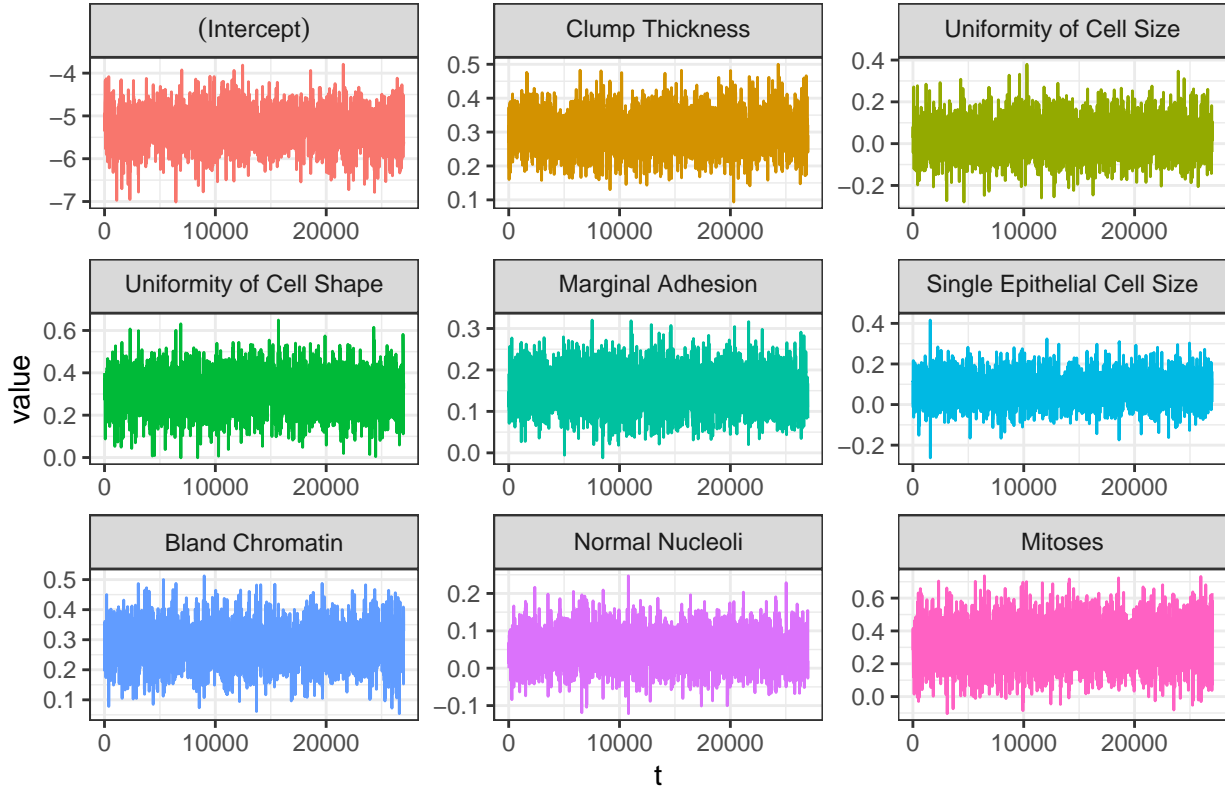


Table 5: Posterior sample summaries for breast cancer data using Stan

Variable	mean	sd	2.5%	50%	97.5%
(Intercept)	-5.2803740	0.4474952	-6.1788551	-5.2675580	-4.4570687
Clump Thickness	0.3020233	0.0581248	0.1891183	0.3010757	0.4166279
Uniformity of Cell Size	0.0283373	0.0899331	-0.1404464	0.0256264	0.2084892
Uniformity of Cell Shape	0.3066612	0.0950854	0.1178779	0.3088803	0.4907881
Marginal Adhesion	0.1525317	0.0502227	0.0576146	0.1520253	0.2512294
Single Epithelial Cell Size	0.0739874	0.0750008	-0.0656497	0.0713772	0.2254342
Bland Chromatin	0.2812440	0.0686278	0.1482095	0.2824382	0.4200620
Normal Nucleoli	0.0526038	0.0495117	-0.0427017	0.0537263	0.1511080
Mitoses	0.3191038	0.1369070	0.0561501	0.3168852	0.5875316

Posterior Trace (Breast Cancer Using Stan)



Posterior Distribution (Breast Cancer Using Stan)

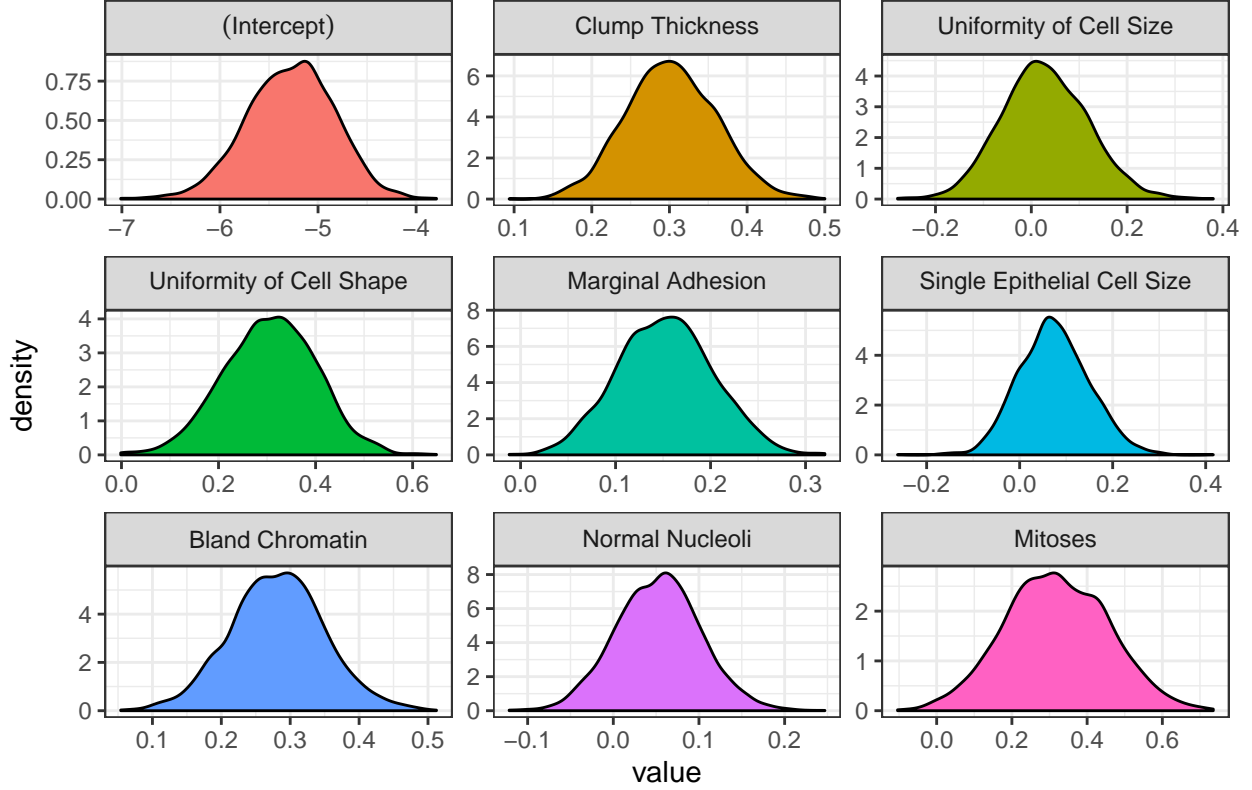


Table 6: MLE estimates for breast cancer data

Variable	Estimate	Std. Error
(Intercept)	-5.1320418	0.4516719
Clump Thickness	0.2897958	0.0572147
Uniformity of Cell Size	0.0156388	0.0871703
Uniformity of Cell Shape	0.3037402	0.0931808
Marginal Adhesion	0.1493079	0.0518700
Single Epithelial Cell Size	0.0715684	0.0743288
Bland Chromatin	0.2758152	0.0715357
Normal Nucleoli	0.0538840	0.0509148
Mitoses	0.3109816	0.1430757

4.3 Baseball

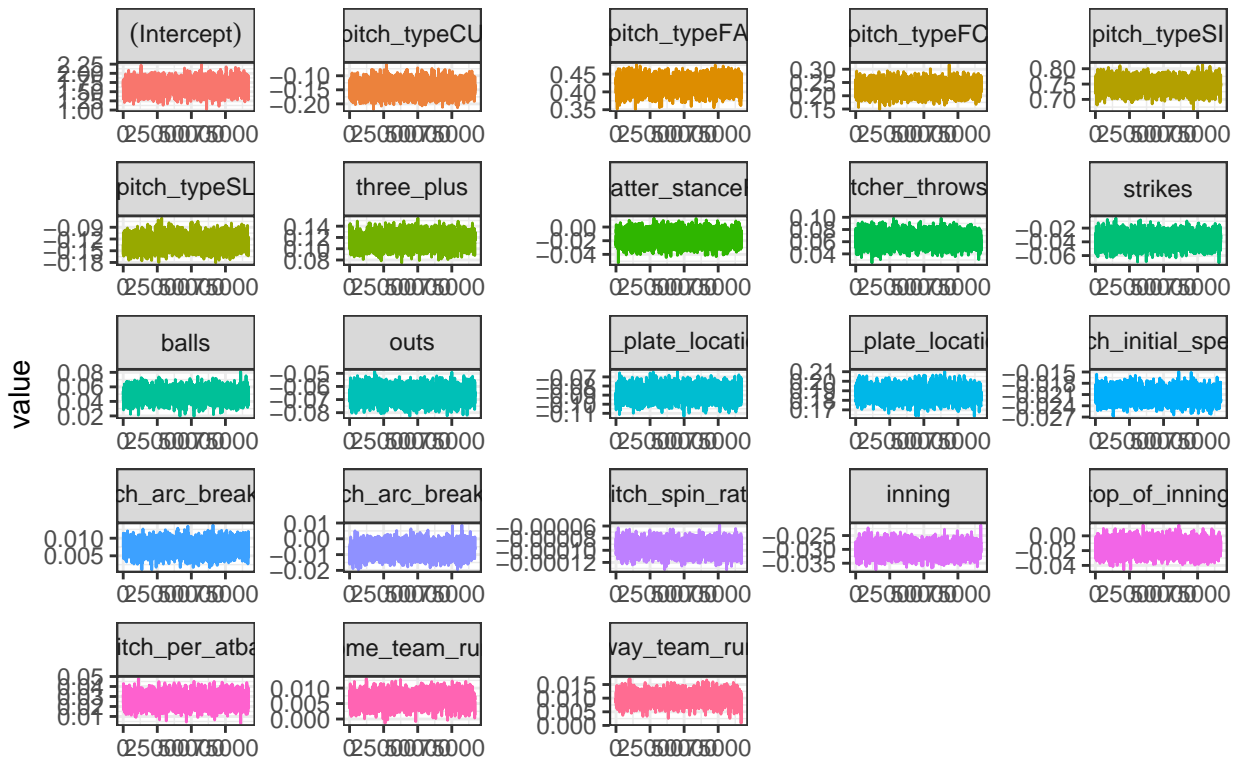
We evaluate the performance of data-augmented Gibbs sampling on a larger datasets (more predictors and observations than the breast cancer data), by fitting probit regression models on baseball data. The baseball data comes from Major League Baseball pitch tracking software and was prepared by the Los Angeles Dodgers for the purpose of predicting whether or not a pitch was put into play. There are 100,000 pitches (observations) thrown by Dodger's starting pitchers and 19 features of each pitch, including velocity, spin rate, and location. The binary response indicates whether or not a pitch was put into play (0: not put into play, 1: put into play). Missing data was removed, leaving 99,254 observations with 18 covariates for analysis. The model parameters are estimated with data augmented Gibbs sampling following the algorithm described in Albert and Chib (1993) and implemented in R and traditional maximum likelihood estimation as implemented in the `glm` function in R (R Core Team 2019). Both Bayesian methods are run to generate 5000 posterior samples,

with the first 1000 discarded as burn-in samples, leaving 4000 samples for analysis. We also attempted to fit a custom model in Stan but the samples were generated very slowly (fewer than 500 samples per hour). In contrast, the R model generated all 5000 samples in 15 minutes and 41 seconds.

Table 7: Posterior sample summaries for baseball data using R-Gibbs

Variable	mean	sd	2.5%	50%	97.5%
(Intercept)	1.6419615	0.1641598	1.3275671	1.6432777	1.9577907
pitch_typeCU	-0.1427578	0.0223181	-0.1875506	-0.1427039	-0.0990722
pitch_typeFA	0.4187926	0.0181987	0.3820077	0.4188624	0.4544004
pitch_typeFC	0.2290787	0.0213407	0.1871727	0.2289236	0.2717159
pitch_typeSI	0.7452006	0.0191260	0.7073330	0.7452225	0.7822887
pitch_typeSL	-0.1282329	0.0162129	-0.1598217	-0.1282685	-0.0963311
three_plus	0.1138644	0.0114908	0.0918353	0.1136950	0.1355962
batter_stanceR	-0.0157872	0.0088088	-0.0331719	-0.0157398	0.0011848
pitcher_throwsR	0.0640037	0.0100326	0.0445198	0.0640910	0.0836733
strikes	-0.0369098	0.0091613	-0.0552714	-0.0368880	-0.0193004
balls	0.0487882	0.0079058	0.0331798	0.0489391	0.0639353
outs	-0.0660833	0.0050520	-0.0760450	-0.0661052	-0.0563236
pitch_plate_location_x	-0.0880404	0.0067681	-0.1014258	-0.0879550	-0.0749833
pitch_plate_location_z	0.1874257	0.0063152	0.1750398	0.1874246	0.1995573
pitch_initial_speed	-0.0211755	0.0016717	-0.0244022	-0.0212001	-0.0179145
pitch_arc_break_x	0.0073551	0.0017686	0.0038227	0.0073578	0.0107850
pitch_arc_break_z	-0.0068673	0.0038864	-0.0145743	-0.0068384	0.0005285
pitch_spin_rate	-0.0000956	0.0000100	-0.0001149	-0.0000956	-0.0000758
inning	-0.0301068	0.0020301	-0.0340363	-0.0301332	-0.0260424
top_of_inning	-0.0157989	0.0084155	-0.0318350	-0.0159557	0.0010927
pitch_per_atbat	0.0263585	0.0060101	0.0148513	0.0263168	0.0383177
home_team_runs	0.0062439	0.0020007	0.0022450	0.0062631	0.0100559
away_team_runs	0.0099700	0.0019942	0.0061492	0.0099993	0.0137278

Posterior Trace (Bawball Using R-Gibbs)



Posterior Distribution (Baseball Using R-Gibbs)

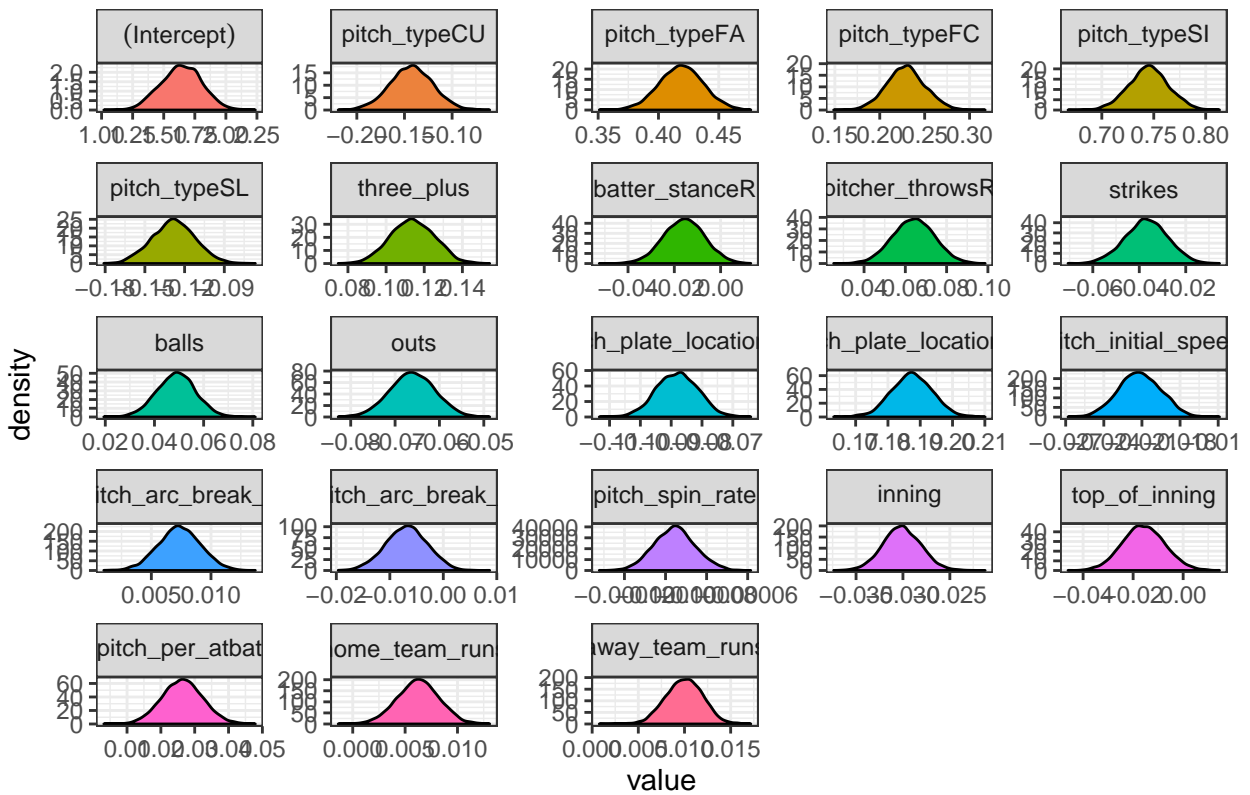


Table 8: MLE estimates for baseball data

Variable	Estimate	Std. Error
(Intercept)	1.6457301	0.1643384
pitch_typeCU	-0.1425477	0.0226587
pitch_typeFA	0.4184923	0.0185715
pitch_typeFC	0.2282183	0.0219511
pitch_typeSI	0.7453712	0.0188182
pitch_typeSL	-0.1279136	0.0167241
three_plus	0.1136658	0.0114023
batter_stanceR	-0.0156687	0.0088127
pitcher_throwsR	0.0637459	0.0099670
strikes	-0.0369765	0.0102701
balls	0.0489444	0.0088985
outs	-0.0661775	0.0050947
pitch_plate_location_x	-0.0881536	0.0068380
pitch_plate_location_z	0.1874354	0.0063953
pitch_initial_speed	-0.0212061	0.0016916
pitch_arc_break_x	0.0073317	0.0017433
pitch_arc_break_z	-0.0067832	0.0038488
pitch_spin_rate	-0.0000956	0.0000100
inning	-0.0300979	0.0020700
top_of_inning	-0.0160490	0.0083282
pitch_per_atbat	0.0262550	0.0070157
home_team_runs	0.0062091	0.0019981
away_team_runs	0.0100125	0.0020411
<!-- There were 23 model parameters -->		

5 Discussion

Here's the discussion.

Mention Corey's research?

Extensions of data augmentation for other models?

Mention poor mixing?

6 References

- Albert, James H., and Siddhartha Chib. 1993. “Bayesian Analysis of Binary and Polychotomous Response Data.” *Journal of the American Statistical Association* 88 (422). Taylor & Francis: 669–79. <https://doi.org/10.1080/01621459.1993.10476321>.
- Carpenter, Bob, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. 2017. “Stan: A Probabilistic Programming Language.” *Journal of Statistical Software, Articles* 76 (1): 1–32. <https://doi.org/10.18637/jss.v076.i01>.
- R Core Team. 2019. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Stan Development Team. 2019. “RStan: The R Interface to Stan.” <http://mc-stan.org/>.
- Venables, W. N., and B. D. Ripley. 2002. *Modern Applied Statistics with S*. Fourth. New York: Springer. <http://www.stats.ox.ac.uk/pub/MASS4>.
- Wolberg, William H, and Olvi L Mangasarian. 1990. “Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology.” *Proceedings of the National Academy of Sciences* 87 (23). National Academy of Sciences: 9193–6. <https://doi.org/10.1073/pnas.87.23.9193>.

7 Appendix