Bayesian logistic regression, regularization and model comparison for predicting "barrels" in Major League Baseball

Thomas Martins

July 3, 2023

Abstract

Starting in 2015, Major League Baseball has implemented the Statcast system, which represents a significant progress in baseball data analysis, producing pitch-level data with many covariates. Among the novel Statcast data, the "barrel" hit classification has been gaining popularity among analysts and fans. This study employs Bayesian binomial outcome GLMs (logistic regression) in order to evaluate the predictive ability of models with different sets of covariates for estimating the probability of a barrel conditional on a batted ball event from a random sample of the 2020 MLB season Statcast data. A regularization framework is used, with Laplace priors (Bayesian Lasso equivalent) for the coefficients of the linear model and the fitted models are compared through WAIC. We found a model with strike zone subdivision dummies and strike zone length has shown the best predictive power among all models fitted with our sample.

1 Introduction

Baseball has a long tradition of statistical analysis, just like sports in general also do, but perhaps baseball is the single sport where statistical analysis has been the most successful. Large sample sizes combined with easily measurable state variables, such as base runners and outs, are likely contributing factors to this. Stories about the use of statistics in baseball have gained notoriety in mainstream media, with [1] being one of the best known examples. Traditionally, baseball has always been a "statistics-heavy" sport, where figures such as batting averages (BA) and earned run averages (ERA) shape fan discussions about the

skill of different players and are a key part of decision-making by teams' front offices [2].

Predicting hitting ability throughout history has been usually done with the batting average (BA) and slugging percentage[1] (SLG) statistics, whilst earned run average (ERA) is the go-to number for comparing pitchers. These three statistics are all calculated from single formulas:

$$\text{BA} = \frac{\text{H}}{\text{AB}} \qquad \text{SLG} = \frac{1\text{B} + 2 * 2\text{B} + 3 * 3\text{B} + 4 * \text{HR}}{\text{AB}} \qquad \text{ERA} = 9 * \frac{\text{ER}}{\text{IP}}$$

where H stands for base hits, AB stands for at bats, 1B, 2B, 3B and HR are base hits broken down by number of bases, respectively singles, doubles, triples and home runs. ER are earned runs, i.e. runs credited to the offense of the hitting team and not errors by the defense, and IP stands for innings pitched. Every time a player comes to bat he is credited a plate appearance (PA). Ignoring some rarer in-game events[2] plate appearances that do not result in the hitter reaching base on balls (BB), also known as a walk, he is credited an at bat (AB). At bats can either result in the hitter getting striked out (SO) or putting the ball in play. When the ball is in play, the hitter can be out on a play by the defense, or reach base, what is known as a (base) hit. Hits are broken down on the amount of bases the hitter is able to run. [3] presents a breakdown on the outcomes of plate appearances:

A number of issues with the batting average statistic have been noted. Besides confounding the abilities of not getting striked out and of turning a batted ball into a base hit, research has found batting averages to be quite subject to random variation [3]. The factor known as batting average on balls in play[3] (BABIP) is very much subject to both defensive skill and luck, therefore an imperfect measure of offensive ability [4]. BABIP is known to usually regress back to the level of .300 (30%) with large enough sample size, although good hitters can consistently keep their BABIP a little above the benchmark of .300. BABIP also affects pitcher ERA, and, because the pitcher has very little control over what happens to the ball once it gets struck by the bat. BABIP represents mostly noise when using ERA to measure pitcher skill [5].

Starting in the 2015 season, Major League Baseball (MLB) has implemented a tracking system known as Statcast in all MLB stadiums. Statcast is able

[1] Although the slugging percentage is known by that name, it is not an actual percentage as its value ranges from 0 to 4

[2] Examples of this include sacrifice hits or getting hit by a pitch

[3] Here the definition of "in play" excludes home runs

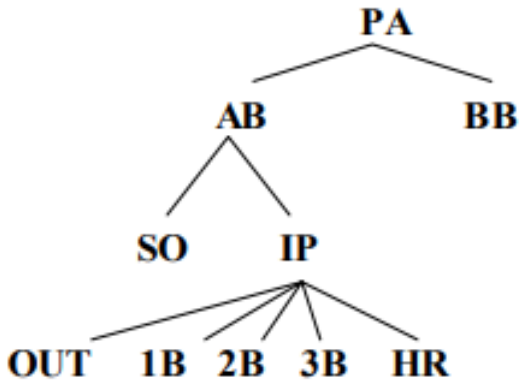

Figure 1: Breakdown of plate appearances in baseball

to track both pitches and batted balls, producing pitch-level data with many covariates that can be useful in baseball statistical analysis [13]. For batted balls, it is able to track launch angle (LA) and exit velocity (EV), and a regressed version of the batting average statistic (xBA) can be calculated in an attempt to measure whether the outcome of the batted ball was far from what would be expected having in mind past events with similar characteristics, and a similar statistic can be calculated for slugging percentage (xSLG). A new statistic called "barrel" has been gaining more and more fame in baseball lately. A barrel is a classification for a batted ball event in which the combination of launch angle and exit velocity results in an expected batting average (xBA) of .500 and an expected slugging percentage (xSLG) of 1.500. A hitter that can consistently hit barrels will have a good offensive performance, while avoiding giving up barrels would be a valuable skill for pitchers, if that were possible. [6] is an attempt to build a new statistic based on barrels for pitcher evaluation, and measuring its predictive power with regards to ERA. [7] combines Statcast data with the proprietary PECOTA algorithm from Baseball Prospectus in order to obtain improved forecast of batting averages.

This present article is an attempt on discovering which pitch covariates available at the Statcast system combined together will result in the best predictiveness for barrels. As barrels are a binary outcome, a generalized linear model (GLM) with a binomially distributed outcome variable is a good candidate for a model. Bayesian estimation shall be employed for model fitting, and Bayesian model selection techniques such as WAIC will be useful for comparing different

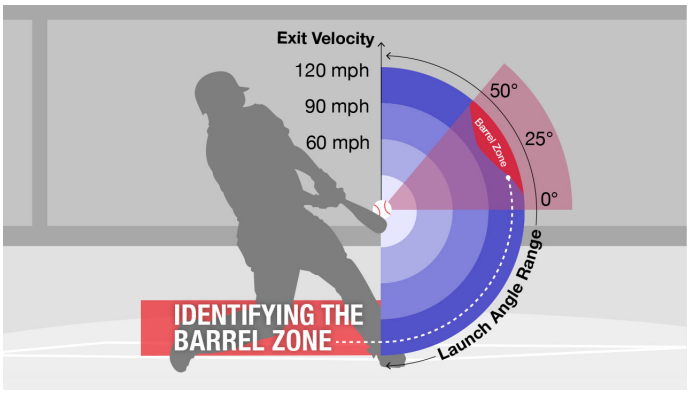

Figure 2: Illustration of the barrel classification. Source: MLB.com

models. [8] compares different hitting metrics through hierarchical Bayesian models, and [10] studies outcomes of plays with a novel multinomial regression method. Neither of these studies use Statcast data.

2 Methodology

2.1 Data

As noted in the previous section, our data comes from the Statcast system by Major League Baseball (MLB). The Statcast API is maintained by MLB Advanced Media (MLBAM) and is freely accessible. Packages such as `baseballr` and `pybaseball` make Statcast data easily available for the most common Statistics and Data Science programming languages. In this study we shall use the `pybaseball` Python library to scrape pitch-level data with covariates such as horizontal and vertical movement, break, velocity, acceleration, spin rate, strike zone size and batter stance. Dummy variables for pitch types, counts and strike zone breakdowns shall also be constructed[4]. Some covariates shall also be scaled in order to estimate our models. The data used for estimation consists of a random sample of 5000 pitch-level observations from games played in the 2020 Major League Baseball regular season consisting only of pitches where the ball was batted and produced a result i.e. batted ball events (BBE) in Statcast nomenclature.

[4] A full dictionary with covariate names and interpretations can be found on `https://baseballsavant.mlb.com/csv-docs`

2.2 Model specification, estimation and variable choice

We shall start with a "basic" model containing the variables [14] has found by employing a gradient boosting method to be the most relevant features in classifying pitch types. These are vertical and horizontal break[5], the initial speed of the pitch and the spin rate. The Statcast names for these four variables are, respectively, `breakz`, `breakx`, `vy0` and `release_spin_rate`, with the last two being scaled. Our dependent variable is a dummy variable which indicates whether that batted-ball received the "barrel" classification by the Statcast system or not constructed from the `launch_speed_angle` column in the dataset[6].

Because the dataset is large and our goal involves dimensionality reduction, we shall employ Laplace priors for regression coefficients, as doing so is equivalent to imposing a Lasso penalty in non-Bayesian settings [11] [12], while the intercept has a Normal prior. We also use partial pooling for the Laplace scale hyperparameter b, which shall have a Half-Cauchy distribution ([16] recommends the Half-Cauchy prior for global scale parameters). Our models can be specified in general form for n observations and k covariates as

$$
\begin{aligned}
y_i &\sim Bernoulli(\theta_i) \\
logit(\theta_i) &= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ji} \\
\beta_0 &\sim N(\mu = 0, \sigma = 10) \\
\beta_j &\sim Laplace(\mu = 0, b) \\
b &\sim HalfCauchy(\beta = 2)
\end{aligned}
$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, k$.

The models in this study will be coded and fitted through the `PyMC3` [9] package for Python. It allows for straightforward implementation of Bayesian GLMs with very few lines of code. The `ArviZ` package is able to compute information criteria, such as WAIC and LOO-CV, for model comparison and perform a variety of graphics plotting and other diagnostics. `PyMC3` uses by default the No U-Turn Sampler (NUTS) [15], which is a subvariant of the Hamiltonian Monte Carlo (HMC). NUTS eliminates the need to manually specify the number

[5] The break of a pitch is the distance in inches between where a pitch crossed the plate and where a hypothetical spinless pitch would have

[6] Numbers from 1 to 6 are assigned to each Statcast classification. The "barrel" classification corresponds to number 6

of steps for the HMC. `PyMC3` offers a variety of methods for initializing NUTS, and in our models we employ the `adapt_diag` method, which starts with an identity mass matrix and then adapts a diagonal based on the variance of the tuning samples, usually setting the prior mean as the initial value for the chains. For each model we sample 2 chains with 2000 iterations plus 1000 burn-in each.

2.2.1 More specific pitch variables

It is possible to add even more variables pertaining to pitch characteristics, such as initial velocity and acceleration in all three dimensions, as only the initial velocity in the y-dimension is present in the base model. Statcast also produces categorical classification for each pitch, and using dummy variables for each pitch type is an alternative. In this part we construct one model with all continuous variables as covariates, and another with the base model continuous variables plus a dummy variable for each of the most common pitch types: four-seamer fastball, changeup, two-seamer fastball, slider, cutter, curveball and sinker. We then compare the two models with the base model through WAIC. The continuous variables, aside those already in the base model, are horizontal and vertical initial velocity (`vx0` and `vz0`), acceleration in all three dimensions (`ax`, `ay` and `az`) with the last two being scaled and horizontal and vertical release position (`release_pos_x` and `release_pos_z`).

2.2.2 Strike zone information

More information about the strike zone for each batted ball event can be added. As the top and bottom of the strike zone are officially defined to be, respectively, the midpoint of the batter's torso and his knees, the size of the strike zone will vary from batter to batter, with taller players having larger strike zones. Batting stance is also a known factor in changing strike zone length. Also, Statcast breaks down the strike zone and its surrounding in smaller parts, allowing us to know in which subzone the ball was batted. We fit three models, one with the top and bottom of the strike zone as covariates, another with subzone dummies and a final one with both and compare the models with the base model through WAIC. The dummies we use correspond to subzones from 1 to 9 and the strike zone length variables `sz_top` and `sz_bot` are automatically set from video by the Statcast system.

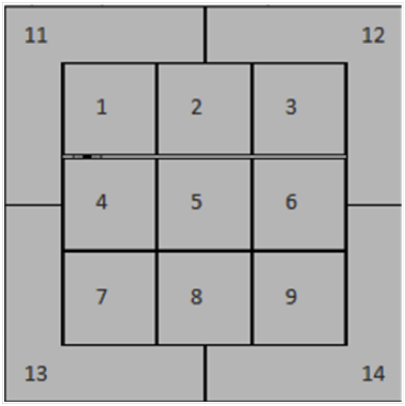

Figure 3: Subzone classification in Statcast. Source:`baseballsavant.com`

2.2.3 Pitch count and batter stance

Variables such as the pitch count (balls-strikes) and the stance of the batter i.e. left or right-handed at a particular batted ball event can also be included as covariates in a model. As the batter is perceived to have an advantage over an opposite-stance pitcher e.g. a left-handed batter presumably has an advantage over a right-handed pitcher, we create a dummy variable for when the batter and pitcher have opposite stances and fit a model with that variable included. We also create dummies for each ball-strike count (with the 0-0 count being omitted) and fit another model. Then, another model is fitted with both stance and count information and the three models are compared with the base model through WAIC.

3 Results

The first comparison involves the models described in section 2.2.1, and, among those, both the model with more continuous variables and the pitch type dummies model seem to perform better than the base model, but rather close to each other, with the slight edge to the continuous variables model. The respective computed model weights are 56% and 41% inside the first comparison.

For the second comparison we use the models from section 2.2.2. Here the model with both zone dummies and strike zone length outperforms the rest, although the model with zone dummies only is not far behind. These two models have computed weights of 65% and 35%, respectively.

The third comparison employs the models with game state variables as described in section 2.2.3. The model with both count and opposite stance dummies has the lowest WAIC, with a computed weight of 61%.

The fourth and final WAIC comparison involves each of the models with lowest WAIC in the previous three comparisons, the base model, a model with selected variables from the first three models (pitch continuous variables, zone dummies and strike zone length, count and opposite stance dummies) and a model with all studied variables at once. The model with only zone dummies and strike zone length is the best performing one, with lowest WAIC and computed weight of 78%. The log-scale WAIC plot can be seen in Figure 5 and both kernel density estimates and trace plots for each parameter of the lowest WAIC model in Figure 4. Table 5 presents a summary of the model.

Full code for this study can be found on `https://github.com/thomasmartins/statcast_bayes_glm/blob/main/statcast_bayes_glm.ipynb`

4 Discussion

In the present article the model with best predictiveness (lowest WAIC) of "barrel" hits conditioned on batted ball events, is the model with the "base" variables plus strike zone subdivisions dummies and strike zone length. Particularly, batted ball events on subzones 2, 3, 5 and 6 were estimated with a higher probability of resulting in a barrel. This suggests barrel hits happen more often on the top and middle of the strike zone, and closer to the left side of the home plate, which is the side where right-handed hitters usually hit from. Future research could try including interaction terms between subzone dummies and batter stance to control for this possible confounding. Still, given the prevalence of right-handed hitters, the estimates could indicate that barrel hits happen more often on batted balls closer to the batter, which could also be more common among so-called "pull hitters" i.e. hitters who usually bat the ball to the same side of the field from which he bats at the home plate. Also worthy of note is that in our analysis the estimated probability of a barrel is conditioned on a batted ball event. Further research could focus on the probability of barrels among all pitches.

This study only analyses the prediction of barrel hits at the entire Major League Baseball level. An interesting development for further work might be analysing predictiveness of covariates at a player level, both for pitchers (which pitchers give up the least amount of barrels and how that skill relates to pitch

characteristics) and batters (certain players might try to hit pitches with certain characteristics, also relating to plate discipline). Partial pooling might also be useful in this. Our analysis is also limited because of the rather small 5000-sample data. Increased computational power could use more observations from the 2020 and previous seasons.

References

[1] Lewis, Michael. (2003) "Moneyball: The Art of Winning an Unfair Game." W.W. Norton.

[2] Albert, Jim. and Bennett, Jay. (2001) "Curve Ball: Baseball, Statistics, and the Role of Chance in the Game." Springer Science & Business Media.

[3] Albert, Jim. (2004) "A Batting Average: Does it Represent Ability or Luck?" `https://bayesball.github.io/papers/paper_bavg.pdf`. Accessed on November 18, 2020.

[4] Slowinski, Steve. (2010) "BABIP." In *Fangraphs Library.* `https://library.fangraphs.com/pitching/babip/`. Accessed on November 18, 2020.

[5] McCracken, Voros. (2001) "Pitching and Defense: How Much Control Do Hurlers Have?" Baseball Prospectus. `https://www.baseballprospectus.com/news/article/878/pitching-and-defense-how-much-control-do-hurlers-have/`. Accessed on November 18, 2020.

[6] Ben-Porat, Eli. (2018) "SANTA: A Binary Approach to Pitcher Evaluation" In *The 2018 Hardball Times Annual.* `https://tht.fangraphs.com/tht-annual-2018/santa-a-binary-approach-to-pitcher-evaluation/`. Accessed on November 18, 2020.

[7] Bailey, Sarah R., Jason Loeppky, and Tim B. Swartz. (2020) "The prediction of batting averages in major league baseball." In *Stats 3.2: 84-93.*

[8] McShane, Blakeley B., Alexander Braunstein, James Piette and Shane T. Jensen. (2011) "A Hierarchical Bayesian Variable Selection Approach to Major League Baseball Hitting Metrics" In *Journal of Quantitative Analysis in Sports 7.4: Article 2.*

[9] Salvatier J., Wiecki T.V., Fonnesbeck C. (2016) "Probabilistic programming in Python using PyMC3." In *PeerJ Computer Science 2:e55 DOI: 10.7717/peerj-cs.55.*

[10] Powers, Scott, Trevor Hastie, and Robert Tibshirani. (2018) "Nuclear penalized multinomial regression with an application to predicting at bat outcomes in baseball." In *Statistical modelling 18.5-6: 388-410.*

[11] Tibshirani, Robert. (1996) "Regression shrinkage and selection via the lasso." In *Journal of the Royal Statistical Society: Series B (Methodological) 58.1: 267-288.*

[12] Park, Trevor, and George Casella. (2008) "The bayesian lasso." In *Journal of the American Statistical Association 103.482: 681-686.*

[13] Healey, Glenn. (2017) "The new Moneyball: How ballpark sensors are changing baseball." In *Proceedings of the IEEE 105.11: 1999-2002.*

[14] Fonnesbeck, Chris. (2019) "Pitch Classification" In *Baseball data analysis in Python.* GitHub repository. `https://github.com/fonnesbeck/baseball/blob/master/notebooks/Pitch%20Classification.ipynb`. Accessed on December 2, 2020.

[15] Hoffman, Matthew D., and Andrew Gelman. (2014) "The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo." In *J. Mach. Learn. Res. 15.1: 1593-1623.*

[16] Polson, Nicholas G., and James G. Scott. (2012) "On the half-Cauchy prior for a global scale parameter." In *Bayesian Analysis 7.4: 887-902.*

A Legend for tables

The columns for tables 1 to 4 stand for the computed WAIC in log scale (waic), the estimated effective number of parameters (p_waic), relative difference in WAIC between the model and the lowest WAIC model (d_waic, always 0 for lowest WAIC model), model weight (weight, loosely interpreted as the probability of that model being the correct one), standard error of the WAIC estimate (se) and standard error of the relative difference between that model and the lowest WAIC model (dse, always 0 for the lowest WAIC model)

Table 5 columns represent, respectively, mean and standard deviation of the posterior estimate for that parameter, 3% and 97% bounds of the 94% highest density interval (hdi), mean and standard deviation of the Monte Carlo standard error (mcse), mean, standard deviation, bulk and tail estimates of effective sample size (ess) and the Gelman-Rubin statistic (r_hat)

B Tables and figures

	waic	p_waic	d_waic	weight	se	dse
contvars	-1332.24	7.77	0	0.56	47.75	0
pitchdummies	-1333.27	7.86	1.03	0.41	47.88	3.85
base	-1337.27	4.18	5.03	0.03	47.64	3.01

Table 1: WAIC comparison for models with base variables only, continuous pitch variables and pitch type dummies

	waic	p_waic	d_waic	weight	se	dse
zone_topbot	-1299.81	14.96	0	0.65	45.62	0
zonedummies	-1300.73	13.63	0.92	0.35	45.62	1.74
sz_topbot	-1336.63	5.03	36.82	0	44.44	7.69
base	-1337.27	4.18	37.47	0	44.47	7.81

Table 2: WAIC comparison for models with base variables only, strike zone dummies, strike zone vertical length and both strike zone dummies and vertical length

	waic	p_waic	d_waic	weight	se	dse
count_stance	-1331.95	11.47	0	0.61	47.14	0
countdummies	-1333.14	10.78	1.19	0.26	46.88	1.66
opp_stance	-1336.42	5.31	4.47	0.08	47.21	3.44
base	-1337.27	4.18	5.32	0.05	46.89	3.83

Table 3: WAIC comparison for models with base variables only, balls-strikes count dummies, opposite stance dummy and both count and stance dummies

	waic	p_waic	d_waic	weight	se	dse
zone_topbot	-1299.81	14.96	0	0.78	44.04	0
selection	-1304.52	24.84	4.72	0.11	44.76	4.84
all_variables	-1305.28	26.81	5.47	0.11	44.19	5.48
count_stance	-1331.95	11.47	32.14	0	44.89	8.49
contvars	-1332.24	7.77	32.43	0	44.52	7.61
base	-1337.27	4.18	37.47	0	43.71	7.81

Table 4: WAIC comparison for models with base variables only, continuous pitch variables, zone dummies and vertical length, count and stance dummies, selected variables (continuous pitch variables + zone dummies and vertical length + count and stance dummies) and all variables

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_mean	ess_sd	ess_bulk	ess_tail	r_hat
Intercept	-6.08	1.38	-8.69	-3.57	0.04	0.02	1479	1479	1478	2151	1.0
pfx_x	0.06	0.06	-0.07	0.17	0.00	0.00	2426	2247	2430	2428	1.0
pfx_z	0.22	0.10	0.04	0.42	0.00	0.00	1502	1502	1504	2111	1.0
vy0_sc	0.30	0.63	-0.80	1.63	0.02	0.01	1201	1201	1260	1637	1.0
release_spin_rate_sc	0.35	0.17	0.03	0.67	0.00	0.00	2313	2223	2326	2593	1.0
sz_top	0.39	0.43	-0.33	1.24	0.01	0.01	1542	1345	1547	1891	1.0
sz_bot	0.51	0.73	-0.83	1.94	0.02	0.01	1608	1417	1677	1673	1.0
zone_1	0.79	0.33	0.19	1.40	0.01	0.01	1252	1252	1253	2317	1.0
zone_2	1.35	0.27	0.87	1.87	0.01	0.01	991	991	992	1848	1.0
zone_3	1.05	0.34	0.40	1.68	0.01	0.01	1231	1225	1227	2010	1.0
zone_4	0.93	0.26	0.44	1.41	0.01	0.01	978	978	977	1728	1.0
zone_5	1.43	0.23	1.01	1.87	0.01	0.01	843	843	842	1620	1.0
zone_6	1.23	0.26	0.77	1.71	0.01	0.01	901	901	902	1704	1.0
zone_7	0.53	0.31	-0.06	1.11	0.01	0.01	1296	1296	1302	2158	1.0
zone_8	0.92	0.25	0.46	1.40	0.01	0.01	921	921	921	1650	1.0
zone_9	0.47	0.30	-0.08	1.03	0.01	0.01	1240	1240	1253	2125	1.0
scale	0.82	0.28	0.34	1.29	0.01	0.00	1373	1373	1252	2072	1.0

Table 5: Parameter estimation summary for the model with lowest WAIC

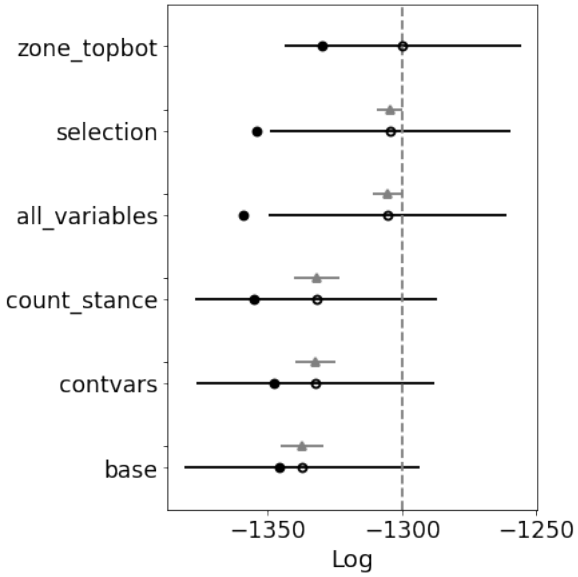

Figure 4: WAIC comparison plot for table 4

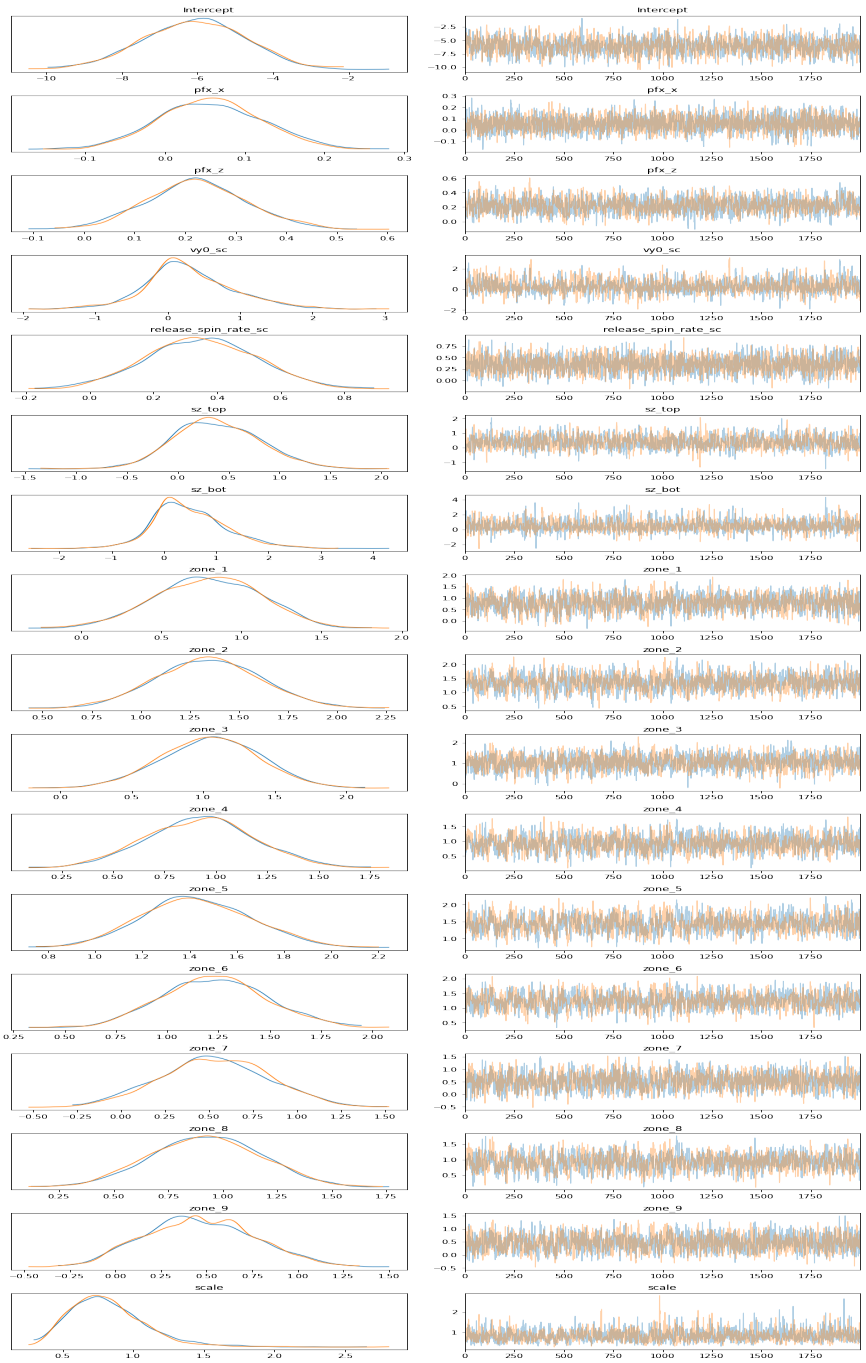

Figure 5: Kernel density estimates and trace plots for the parameters of the model with lowest WAIC