

# Bayesian Thinking: Fundamentals, Regression and Multilevel Modeling

Jim Albert and Jingchen (Monika) Hu

November 2020

# Many Regressions

- Imagine that you have many regressions, divided by a grouping variable.
- Can pool the data over the grouping variable.
- Or can run separate regressions.
- Both approaches are unsatisfactory.
- A Bayesian multilevel model is a compromise solution.

# Situational Effects in Sports

- In baseball, one is generally interested in learning about player abilities and making predictions of future performance.
- Some types of performance are relatively obscure, specifically the performance of players in different situations.
- We use multilevel models to explore one important situational effect – a player's tendency to hit better in balls in play when he has an advantage in the count.

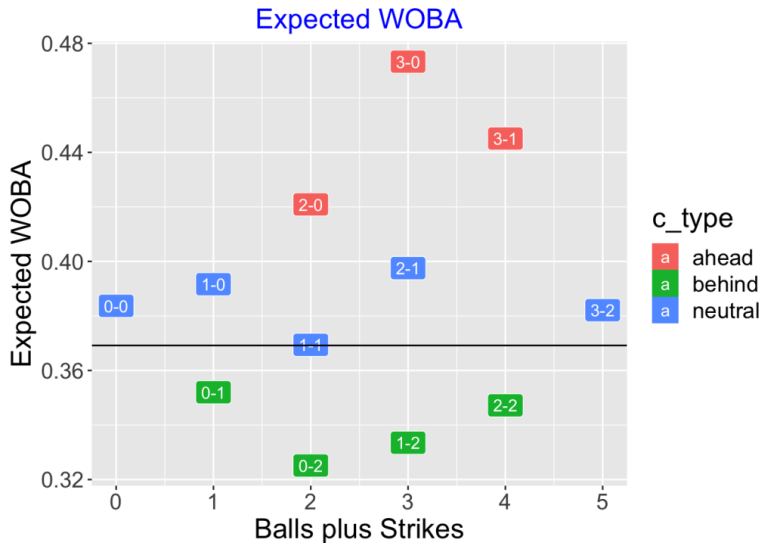
## Plate Appearance, Count, and a Ball in Play

- In baseball, in a “plate appearance” (PA), a pitcher throws the batter a series of pitches.
- Each pitch is recorded as a “ball” or a “strike”.
- The “count” is the current number of balls and strikes in a PA – for example “2-1” means the count is currently two balls and one strike.
- The batter wants to put the ball “in play” (BIP) – measure of the quality of the BIP is called wOBA.

## Quality of the BIP Depends on the Count

- This wOBA on balls in play varies by the count as shown in a graph on the next slide.
- The values cluster naturally into three groups.
- “**AHEAD**” counts (2-0, 3-0, 3-1) have wOBA values between 0.42 and 0.48
- “**BEHIND**” counts (0-1, 0-2, 1-2, 2-2) have wOBA values between 0.32 and 0.36
- “**NEUTRAL**” counts (0-0, 1-0, 1-1, 2-1, 3-2) have wOBA values between 0.36 and 0.40.

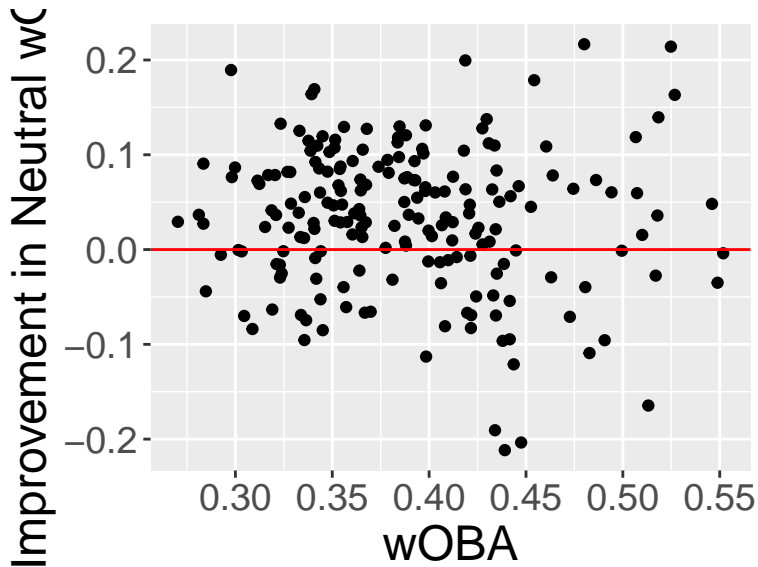
# Graph of wOBA on Balls in Play by Count



## Individual Count Effects?

- Hitters' performance on balls in play depends on the count.
- Does this count advantage depends on the hitter?
- For each player, I computed the mean wOBA in each of the behind and neutral count situations on balls in play.
- Plot the wOBA improvement (NEUTRAL minus BEHIND) against the average wOBA for these players.

## Comparing Neutral and Behind Counts





## Comments

- In each case, most of the points fall above the red line indicating that players tend to do better in the more advantageous count situation.
- But many points fall below the red line in each plot – many players actually hit poorer in the more advantageous situation.
- Does one believe that some hitters are not able to take advantage of the more favorable count?

# What to Do? (To Pool or Not to Pool?)

- If we pool all of the hitting data together, there is a clear count effect. Collectively, batters hit better on balls in play when they are in favorable count situations.
- We computed individual-level advantages in count for all players. Although we see some general effects, some hitters actually hit worse in more advantageous count situations.
- Can we really believe these individual-level results? They may not correspond to real effects that reflect baseball talents.

## Preview: Multilevel Modeling

- Good illustration of multilevel modeling.
- Interested in both a player's wOBA talent and also in his ability to take advantage of a particular count situation.
- Put this problem into a regression context and so the issue will be how to combine a collection of individual-level regression estimates.

## Compare Three Approaches

- **Pool the Data.** We can assume that all batters are the same with respect to their BABIP ability and their ability to take advantage of the favorable count situation. We fit a single regression model.
- **Individual Estimates.** On the other extreme, we can assume that batters differ with respect to their hitting ability and also their ability to take advantage of the count situation. Then we would fit separate regression models for the hitters.
- **Multilevel Model Estimates.** Last, we'll consider a so-called “varying intercepts, varying slopes” multilevel model which states that the individual regression parameters come from a common prior distribution with unknown parameters.

# A Simple Regression Model

- Let  $y$  denote the expected wOBA.
- Define the effect to be 0.5 if the batter is in a neutral count and -0.5 if the batter is behind in the count.
- Assume that  $y$  is normal with mean  $\mu$  and standard deviation  $\sigma$  where the mean response is written as

$$\mu = \beta_0 + \beta_1 \text{EFFECT}$$

- In this regression model  $\beta_0 = (\mu_1 + \mu_2)/2$  represents the overall expected wOBA and  $\beta_1 = \mu_1 - \mu_2$  represents the improvement (from BEHIND-in-the-count to NEUTRAL-count) in expected wOBA.

## A Pooled Data Estimate

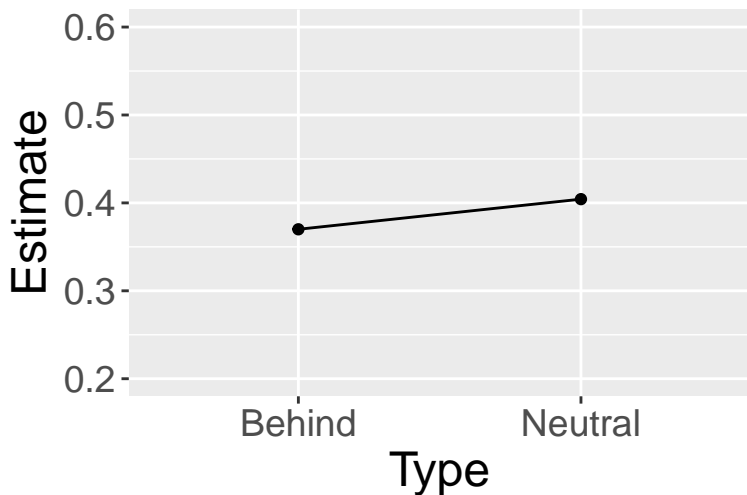
- Fit a regression model to all of the balls in play for all players.
- Note that 0.387 is an average expected wOBA and hitters tend to hit 0.034 better when they are in a neutral count as opposed to behind in the count.

```
fit <- lm(estimated_woba_using_speedangle ~ effect,  
          data = sc_regular_2)  
fit$coef
```

```
## (Intercept)      effect  
## 0.38711110 0.03428241
```

## Graph pf Pooled Data Estimate

### Pooled Regression Curve



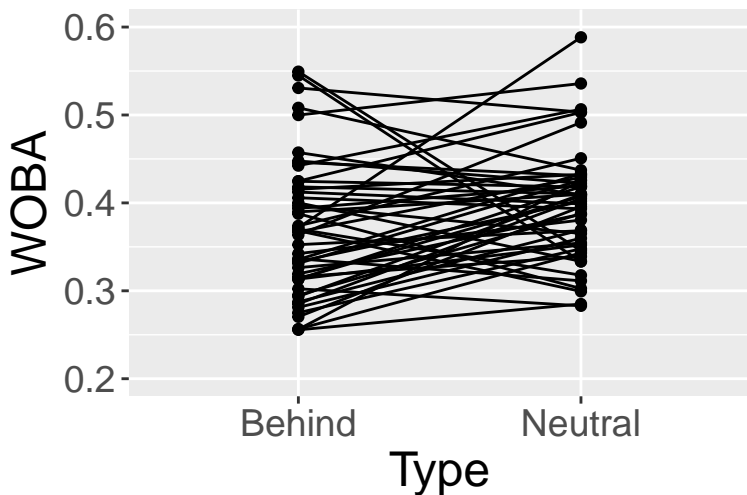
# Individual Estimates

- We implement separate regression fits for all players, obtaining many sets of regression coefficients.



## Graph of 50 Individual Fits

### 50 Individual Regression Curves



# Comments on Individual Fits

- Most regression curves are increasing from Behind to Neutral.
- But some players display decreasing curves.
- How to interpret?

# A Multilevel Model - Sampling

- $y_{ij}$  denotes the  $j$ th measurement of performance for the  $i$ th player
- Assume

$$y_{ij} \sim N(\beta_{0j} + \beta_{1j}EFFECT_{ij}, \sigma)$$

- Let  $\beta_i = (\beta_{0j}, \beta_{1j})$  denote vector of regression coefficients for the  $i$ th player
- Called a “varying intercepts, varying slopes” model

## A Multilevel Model - Prior

- Let  $\beta_1, \dots, \beta_k$  represent the coefficient vectors for the  $k$  regular players.
- Assume that these (random) parameter vectors come from a common bivariate normal distribution with mean  $\beta_0$  and variance-covariance matrix  $\Sigma$ :

$$\Sigma = \begin{bmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{bmatrix}$$

- $\sigma_0$  represents the variation between the player intercepts,  $\sigma_1$  represents the variation between the player effects, and  $\rho$  is the correlation between the player abilities and their situational abilities.

## Two Ways to Fit this Model

- In a full Bayesian model, we can place a weakly informative prior on all these unknown parameters to complete the multilevel model and fit using Stan.
- Or one can do a quick fit of this “random effects” model by using the `lmer()` function in the `lme4` package.
- This quick fit finds point estimates of the unknown parameters at the 2nd stage of the multilevel prior.

## Quick lmer() Fit

```
newfit <- lmer(estimated_woba_using_speedangle ~  
              effect + (1 + effect | player_name),  
              data = sc_regular_2)
```

## Output of lmer() Fit

```
newfit
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: estimated_woba_using_speedangle ~ effect + (1 + ef
## Data: sc_regular_2
## REML criterion at convergence: 25943.96
## Random effects:
## Groups          Name          Std.Dev. Corr
## player_name (Intercept) 0.04842
##              effect      0.02138  0.14
## Residual              0.40123
## Number of obs: 25415, groups:  player_name, 194
## Fixed Effects:
## (Intercept)          effect
##    0.38805         0.03275
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimize
```

# Interpret

- Overall, the mean wOBA value is 0.388 and the advantage of being in the neutral count is 0.033.
- Obtain estimates at the components of the variance-covariance matrix  $\Sigma$ .
- Estimate at the variation of intercepts is  $\sigma_0 = 0.0484$  – this indicates that players do differ in overall BABIP talent and the spread of these talents is given by 0.0484.
- Estimate of variation of slopes is given by  $\sigma_1 = 0.021$ . This is a relatively small value which means that players have little variation in their ability to take advantage of the count situation.



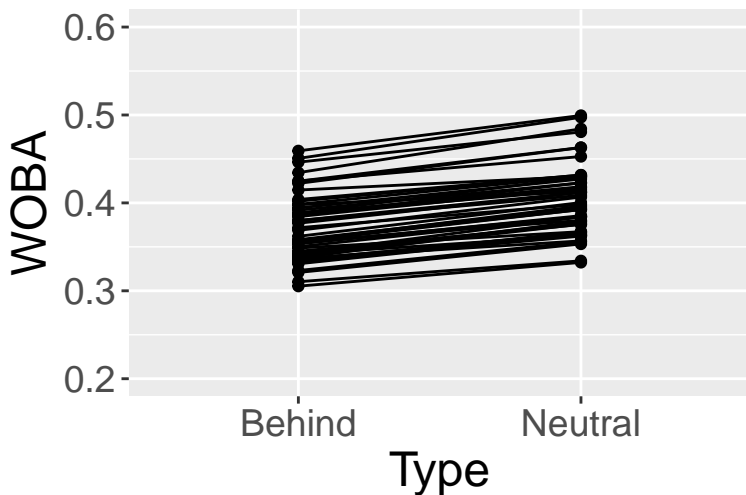
# Performance

- The `lmer()` fitting function provides estimates at the “random effects”, that is, the regression intercepts and slopes for the individual players.

##		beta0	beta1	player_name
## 1	0.3864822	0.03088216	Aaron Hicks	
## 2	0.3368094	0.02122074	Adalberto Mondesi	
## 3	0.4102266	0.03543511	Adam Duvall	
## 4	0.3518056	0.03743973	Adam Eaton	
## 5	0.3367651	0.02852329	Adam Frazier	
## 6	0.4030884	0.03625320	AJ Pollock	

## Graph of These Estimates

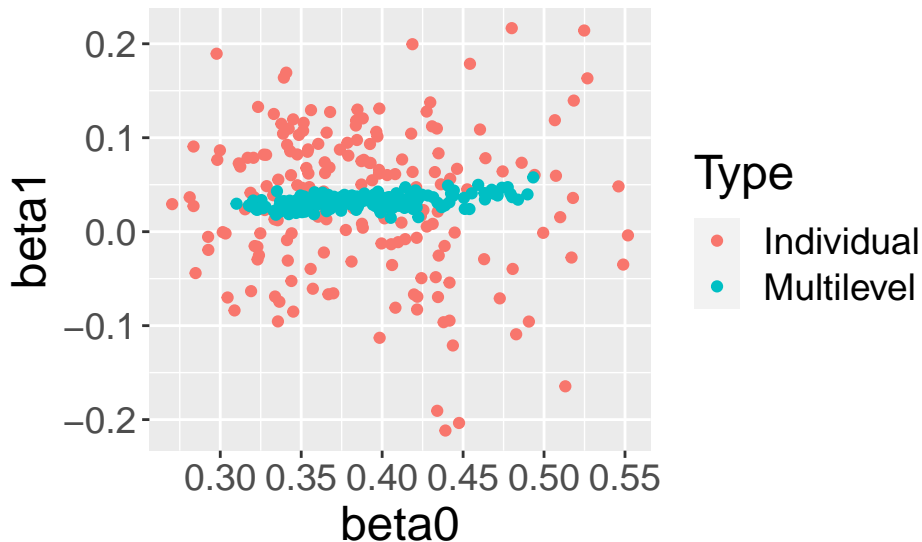
### 50 Multilevel Regression Curves



# Interpretation

- Note that all of these curves are increasing.
- Although some players had decreasing individual regression curves, this fitted model corrects this and tells us that actually everyone has increasing curve talents.
- But the size of the increase can differ between players.

## Graph Comparing Individual and Multilevel Estimates



# Interpretation

- Note that many of the individual slope estimates (the  $\beta_1$ ) are negative but none of the multilevel slope estimates are negative.
- The individual slope estimates are shrunk or adjusted strongly towards an average slope value.
- We see shrinkage both in the slopes and the intercepts, but the degree of shrinkage is smaller for the intercepts (the player wOBA abilities).
- Another interesting thing to see is that there is a positive correlation between the multilevel intercepts and slopes – this means that better hitters tend to have larger count effects.

# Fit the Full Bayesian Model

- Prior? (Show the default ones that are used by the brms package.)
- Focus on inference on second-stage standard deviations.
- Short discussion of MCMC diagnostics.

# Bayesian Fit

```
library(brms)
bfit <- brm(estimated_woba_using_speedangle ~
  effect + (1 + effect | player_name),
  data = sc_regular_2)
```

- Relatively slow – 17 minutes for this example
- About 200 players (400+ parameters)

# Prior

```
> prior_summary(bfit)
```

	prior	class	coef	group	resp
	(flat)	b			
	(flat)	b	effect		
student_t(3, 0.3, 2.5)		Intercept			
lkj_corr_cholesky(1)		L			
lkj_corr_cholesky(1)		L		player_name	
student_t(3, 0, 2.5)		sd			
student_t(3, 0, 2.5)		sd		player_name	
student_t(3, 0, 2.5)		sd	effect	player_name	
student_t(3, 0, 2.5)		sd	Intercept	player_name	
student_t(3, 0, 2.5)		sigma			



# Some Basic MCMC Diagnostic Graphs

- Trace plots of each parameter of interest (simulated value against iteration number)
- Autocorrelation plots (autocorrelations against lag number)
- Density estimates

# Summary MCMC Diagnostics

- $\hat{R}$  (potential scale reduction statistic): compares behavior of one chain with other randomly initialized chains – if chains are at equilibrium,  $\hat{R}$  will be close to 1
- ESS (effective sample size): estimate of the number of independent draws from the posterior distribution of the parameter of interest
- Typically ESS is smaller than simulation sample size

# Posterior Summaries

Group-Level Effects:

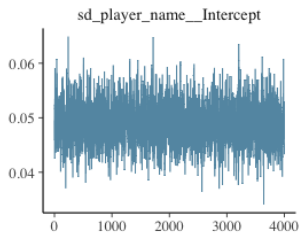
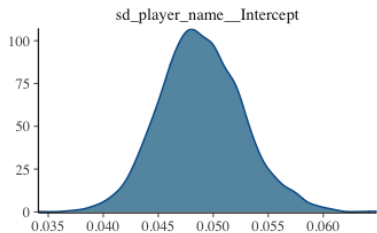
~player\_name (Number of levels: 194)

	Estimate	Est.Error	l-95% CI	
sd(Intercept)	0.05	0.00	0.04	
sd(effect)	0.02	0.01	0.00	
cor(Intercept,effect)	0.13	0.43	-0.79	
	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sd(Intercept)	0.06	1.00	1766	2623
sd(effect)	0.04	1.00	885	1968
cor(Intercept,effect)	0.92	1.00	4162	1739

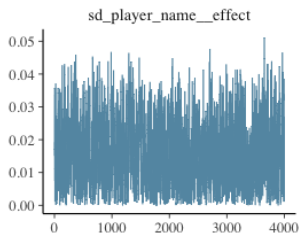
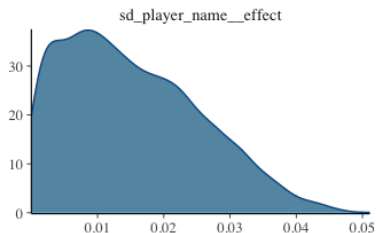
# Trace and Density Estimates

```
post <- posterior_samples(bfit)
library(bayesplot)
mcmc_combo(post, pars = c("sd_player_name__Intercept",
                          "sd_player_name__effect"))
```

# Trace and Density Estimates

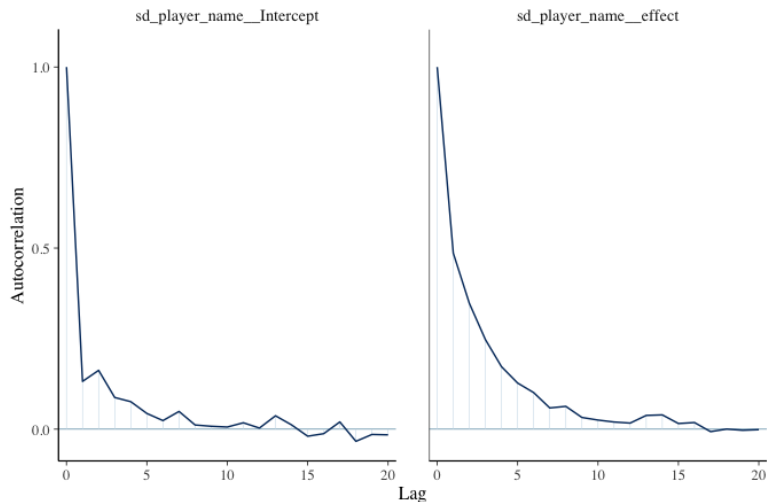


Chain  
— 1



# Autocorrelation Graphs

```
mcmc_acf(post, pars = c("sd_player_name__Intercept",  
                        "sd_player_name__effect"))
```



# Takeaways

- See better mixing of  $\sigma_0$  than for  $\sigma_1$
- This demonstrated through the acf and trace plots, and the values of the ESS
- Would suggest rerunning fit with a large simulation sample size

# Wrapup

- Baseball folks are always fascinated with situational effects.
- Also situational data is hard to understand since there is so much variability in these observed situational effects due to the small sample sizes.
- Multilevel model estimates adjust the individual estimates towards a common value and the degree of adjustment depends on the particular situation.
- Here the variation in individual count effects appears to be mainly luck-driven and the multilevel estimates shrink them heavily towards a constant value.