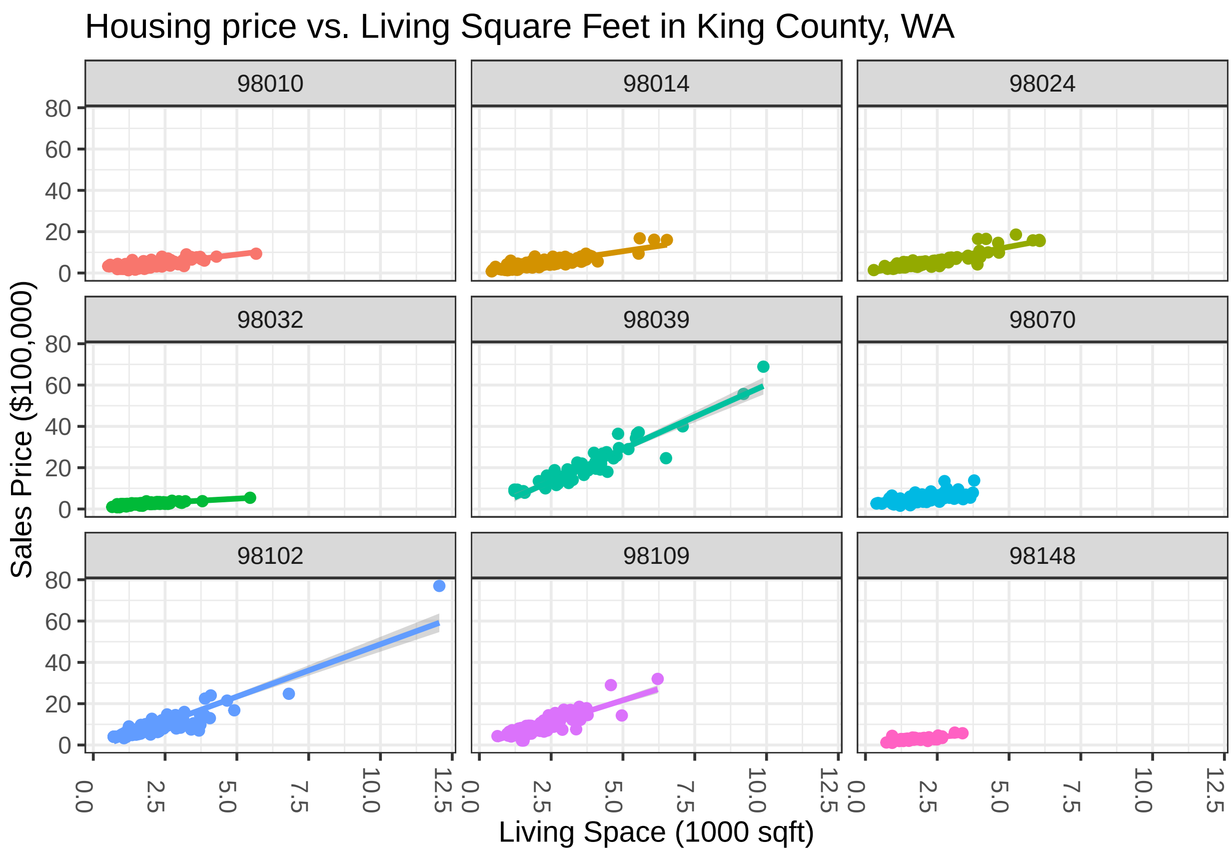
Hierarchical Models

Motivating Dataset

Recall the housing dataset from King County, WA that contains sales prices of homes across the Seattle area.



Multilevel models

While we will initially just look at a model with the varying intercepts,

There are several different, but equivalent specifications in GH 12.5, but here is one way to look at the model.

lmer

One common approach for hierarchical models is to use the `lmer` function in the `lme4` package. Note that the hierarchical structure we have detailed can also be applied to GLMs using `glmer`. Note that most of this code (and the textbook) is "pre-rstanarm", so it might be more intuitive to use `stan_glmer`, which we will also look at a Bayesian version in a little bit using `stan_glmer`.

```
lmer1 <- lmer(price ~ (1 | zipcode) , data = seattle)
display(lmer1)
```

```
## lmer(formula = price ~ (1 | zipcode), data = seattle)
##  coef.est   coef.se
## 713204.24 195580.94
##
## Error terms:
##  Groups   Name        Std.Dev.
##  zipcode  (Intercept) 584666.88
##  Residual             460890.53
## ---
## number of obs: 869, groups: zipcode, 9
## AIC = 25155.1, DIC = 25201.4
## deviance = 25175.2
```

```
coef(lmer1)
```

```
## $zipcode
##       (Intercept)
## 98010    425454.1
## 98014    456901.5
## 98024    581647.2
## 98032    253581.2
## 98039   2143523.7
## 98070    488663.0
## 98102    900408.3
## 98109    879131.8
## 98148    289527.5
##
## attr(,"class")
## [1] "coef.mer"
```

Note the coefficients for a specific group are defined as the fixed effect + the random effect.

```
fixef(lmer1)
```

```
## (Intercept)
##    713204.2
```

The fixed effect here corresponds to μ_α. The standard component associated with the random effect can also be extracted.

```
sigma.hat(lmer1)$sigma$zipcode
```

```
## (Intercept)
##    584666.9
```

```r
ranef(lmer1)
```

```
## $zipcode
##       (Intercept)
## 98010   -287750.1
## 98014   -256302.7
## 98024   -131557.1
## 98032   -459623.1
## 98039   1430319.4
## 98070   -224541.3
## 98102    187204.0
## 98109    165927.6
## 98148   -423676.7
##
## with conditional variances for "zipcode"
```

```r
fixed_ci <- round(fixef(lmer1)['(Intercept)'] + c(-2,2) * se.fixef(lmer1)['(Intercept)'])
```

Summarizing the model The 95% interval for the fixed effects intercept is (322,042, 1,104,366). This can be interpreted as the overall mean price of a house. Formally, this is more the mean of the group means.

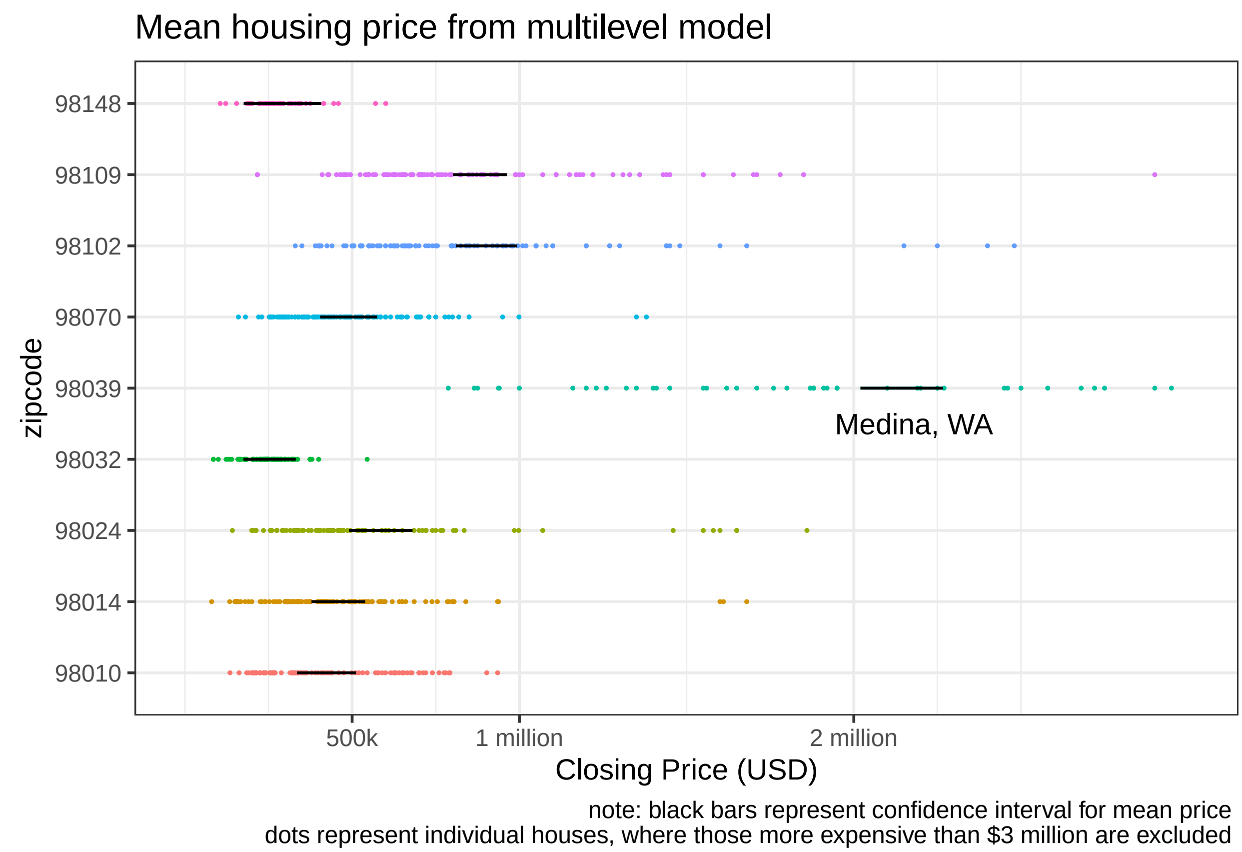
The 95% intervals for the group effects (or deviations from the mean price) are:

zipcode	lower	upper
98010	-379643	-195857
98014	-338874	-173731
98024	-233587	-29528
98032	-541866	-377381
98039	1300763	1559876
98070	-309176	-139907
98102	97512	276896
98109	77888	253968
98148	-545109	-302244

A more useful way to summarize the data would be to create 95% intervals for the overall intercept

```r
samples <- arm::sim(lmer1, n.sims = 1000)
overall <- fixef(samples)
group <- matrix(ranef(samples)$zipcode[,,1], nrow = 1000, ncol = ngrps(lmer1), byrow = F)
group_totals <- group + matrix(overall, nrow = 1000, ncol = ngrps(lmer1))
```

```
## Warning: Removed 9 rows containing missing values (geom_point).
```



Prediction

Note the previous figure contains uncertainty for the mean price within a particular zipcode. Similar to before you could also make predictions for a new home in an existing dataset.

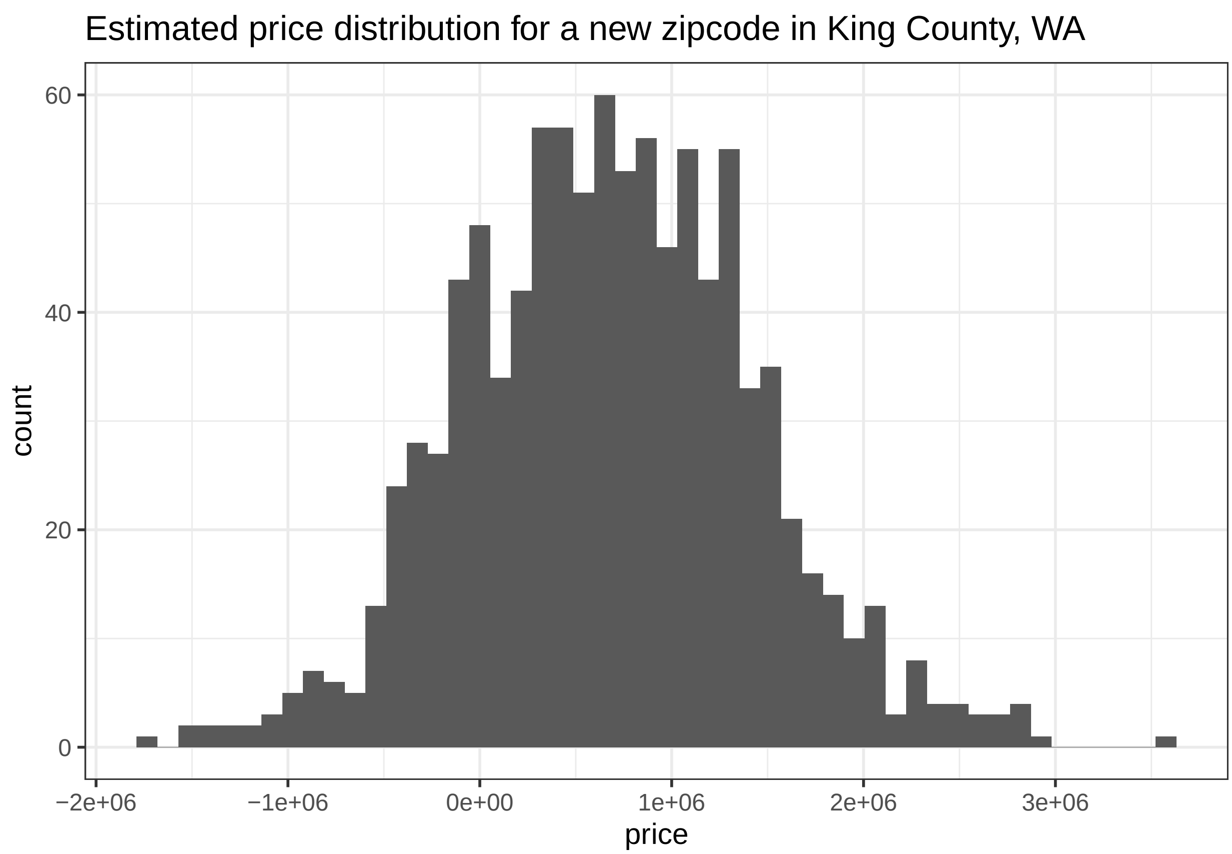
```
sigma_alpha <- sigma.hat(lmer1)$sigma$zipcode
mu_alpha <- fixef(lmer1)["(Intercept)"]
rnorm(10, mu_alpha, sigma_alpha)
```

```
##  [1]  597392.9 1331880.0  582058.6  520589.4 1131877.4  765969.3  511249.9
##  [8] 1042227.0  171613.6  863484.0
```

```
alpha_samples <- rnorm(1000, mu_alpha, sigma_alpha)
```

```
sigma_y <- sigma.hat(lmer1)$sigma$data

new_zip <- rnorm(1000, mean = alpha_samples, sd = sigma_y)
```



Adding Coefficients The model we have just outlined does not include any additional covariates.

```
lmer2 <- lmer(price ~ scale_sqft + (1 |zipcode), data = seattle)

display(lmer2)
```

```
## lmer(formula = price ~ scale_sqft + (1 | zipcode), data = seattle)
##             coef.est  coef.se
## (Intercept) 682210.16 127976.83
## scale_sqft  403385.07  10167.55
##
## Error terms:
##  Groups   Name        Std.Dev.
##  zipcode  (Intercept) 382797.06
##  Residual             274619.10
## ---
## number of obs: 869, groups: zipcode, 9
## AIC = 24238.4, DIC = 24321.6
## deviance = 24276.0
```

Note: you may have to adjust the REML and optimizer options to achieve convergence

```
lmer_nonconverge <- lmer(price ~ scale_sqft + (1 + scale_sqft|zipcode), data = seattle)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00246273 (tol = 0.002, component 1)
```

```
lmer3 <- lmer(price ~ scale_sqft + (1 + scale_sqft|zipcode), data = seattle,
      REML = FALSE)
display(lmer3)
```

```
## lmer(formula = price ~ scale_sqft + (1 + scale_sqft | zipcode),
##     data = seattle, REML = FALSE)
##             coef.est  coef.se
## (Intercept) 606247.84  90723.36
## scale_sqft  330602.83  69904.63
##
## Error terms:
##  Groups   Name        Std.Dev.  Corr
##  zipcode  (Intercept) 271254.68
##           scale_sqft  208120.29 0.99
##  Residual             196377.48
## ---
## number of obs: 869, groups: zipcode, 9
## AIC = 23716.8, DIC = 23704.8
## deviance = 23704.8
```

The fixed-effects or means of the group-level effects can be extracted.

```
fixef(lmer3)
```

```
## (Intercept)  scale_sqft
##    606247.8    330602.8
```

Similarly, the variance of those group-level effects can also be obtained from the model.

```
sigma.hat(lmer3)$sigma
```

```
## $data
## [1] 196377.5
##
## $zipcode
## (Intercept)  scale_sqft
##    271254.7    208120.3
```

stan_glmer

Similar to how we have used `stan_glm()`, we can also use `stan_glmer()` to fit these models.

```
stan_lmer1 <- stan_glmer(price ~ (1 | zipcode) , data = seattle)
```

```
print(stan_lmer1)
```

```
## stan_glmer
##  family:       gaussian [identity]
##  formula:      price ~ (1 | zipcode)
##  observations: 869
## ------
##             Median   MAD_SD
## (Intercept) 710809.8 198010.9
##
## Auxiliary parameter(s):
##       Median   MAD_SD
## sigma 461293.0  10675.3
##
## Error terms:
##  Groups   Name        Std.Dev.
##  zipcode  (Intercept) 642984
##  Residual             461531
## Num. levels: zipcode 9
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
display(lmer1)
```

```
## lmer(formula = price ~ (1 | zipcode), data = seattle)
##  coef.est   coef.se
## 713204.24 195580.94
##
## Error terms:
##  Groups   Name        Std.Dev.
##  zipcode  (Intercept) 584666.88
##  Residual             460890.53
## ---
## number of obs: 869, groups: zipcode, 9
## AIC = 25155.1, DIC = 25201.4
## deviance = 25175.2
```

```r
coef(stan_lmer1)
```

```
## $zipcode
##       (Intercept)
## 98010    426999.7
## 98014    457520.6
## 98024    577670.1
## 98032    251697.5
## 98039   2135335.9
## 98070    486970.1
## 98102    898307.5
## 98109    878468.9
## 98148    288285.6
##
## attr(,"class")
## [1] "coef.mer"
```

```r
coef(lmer1)
```

```
## $zipcode
##       (Intercept)
## 98010    425454.1
## 98014    456901.5
## 98024    581647.2
## 98032    253581.2
## 98039   2143523.7
## 98070    488663.0
## 98102    900408.3
## 98109    879131.8
## 98148    289527.5
##
## attr(,"class")
## [1] "coef.mer"
```

```
##
## Model Info:
##  function:     stan_glmer
##  family:       gaussian [identity]
##  formula:      price ~ (1 | zipcode)
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 869
##  groups:       zipcode (9)
##
## Estimates:
##                                          mean          sd
## (Intercept)                             7.226994e+05  2.162493e+05
## b[(Intercept) zipcode:98010]           -2.970435e+05  2.202122e+05
## b[(Intercept) zipcode:98014]           -2.663803e+05  2.198860e+05
## b[(Intercept) zipcode:98024]           -1.412039e+05  2.225942e+05
## b[(Intercept) zipcode:98032]           -4.690052e+05  2.191224e+05
## b[(Intercept) zipcode:98039]            1.418502e+06  2.250469e+05
## b[(Intercept) zipcode:98070]           -2.336724e+05  2.204370e+05
## b[(Intercept) zipcode:98102]            1.774991e+05  2.200245e+05
## b[(Intercept) zipcode:98109]            1.563999e+05  2.201449e+05
## b[(Intercept) zipcode:98148]           -4.334978e+05  2.225132e+05
## sigma                                   4.615311e+05  1.085570e+04
## Sigma[zipcode:(Intercept),(Intercept)]  4.134289e+11  2.368805e+11
##                                          2.5%          97.5%
## (Intercept)                             3.136974e+05  1.170196e+06
## b[(Intercept) zipcode:98010]           -7.539289e+05  1.220869e+05
## b[(Intercept) zipcode:98014]           -7.254979e+05  1.505290e+05
## b[(Intercept) zipcode:98024]           -5.976696e+05  2.715432e+05
## b[(Intercept) zipcode:98032]           -9.351709e+05 -5.510100e+04
## b[(Intercept) zipcode:98039]            9.548104e+05  1.848364e+06
## b[(Intercept) zipcode:98070]           -6.949283e+05  1.850929e+05
## b[(Intercept) zipcode:98102]           -2.671425e+05  6.000109e+05
## b[(Intercept) zipcode:98109]           -3.001614e+05  5.733417e+05
## b[(Intercept) zipcode:98148]           -9.006296e+05 -1.586970e+04
## sigma                                   4.409736e+05  4.835914e+05
## Sigma[zipcode:(Intercept),(Intercept)]  1.507795e+11  1.030117e+12
##
## Fit Diagnostics:
##           mean     sd       2.5%     97.5%
## mean_PPD 632378.9  22459.8 588142.8 675822.8
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (for de
##
## MCMC diagnostics
##                                        mcse         Rhat         n_eff
## (Intercept)                                  8138.0          1.0  706
## b[(Intercept) zipcode:98010]                 8220.8          1.0  718
## b[(Intercept) zipcode:98014]                 8153.5          1.0  727
## b[(Intercept) zipcode:98024]                 8281.1          1.0  723
## b[(Intercept) zipcode:98032]                 8144.1          1.0  724
## b[(Intercept) zipcode:98039]                 8219.8          1.0  750
## b[(Intercept) zipcode:98070]                 8226.5          1.0  718
```

```
## b[(Intercept) zipcode:98102]                8165.1          1.0  726
## b[(Intercept) zipcode:98109]                8100.4          1.0  739
## b[(Intercept) zipcode:98148]                8156.7          1.0  744
## sigma                                        219.2          1.0 2453
## Sigma[zipcode:(Intercept),(Intercept)] 8939298248.4          1.0  702
## mean_PPD                                     350.1          1.0 4116
## log-posterior                                  0.1          1.0  793
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective sample
```

We can also directly extract the simulations from the stan object.

```
##           parameters
## iterations (Intercept) b[(Intercept) zipcode:98010]
##       [1,]    686160.8                      -249142.9
##       [2,]    771784.1                      -369880.3
##       [3,]    800358.2                      -364324.7
##       [4,]    796693.1                      -350150.9
##       [5,]    745118.0                      -314297.6
##       [6,]    861878.1                      -424325.7
##           parameters
## iterations b[(Intercept) zipcode:98014] b[(Intercept) zipcode:98024]
##       [1,]                    -197022.7                    -128271.3
##       [2,]                    -359239.1                    -160552.2
##       [3,]                    -305496.3                    -297369.7
##       [4,]                    -289467.3                    -223583.0
##       [5,]                    -299151.1                    -157636.2
##       [6,]                    -406094.9                    -287223.3
##           parameters
## iterations b[(Intercept) zipcode:98032] b[(Intercept) zipcode:98039]
##       [1,]                    -411654.6                      1448309
##       [2,]                    -520055.0                      1363944
##       [3,]                    -575881.9                      1305553
##       [4,]                    -585284.2                      1344038
##       [5,]                    -476055.7                      1314614
##       [6,]                    -557884.6                      1219907
##           parameters
## iterations b[(Intercept) zipcode:98070] b[(Intercept) zipcode:98102]
##       [1,]                    -179055.7                    242842.53
##       [2,]                    -311546.7                     82564.86
##       [3,]                    -332253.2                    161304.87
##       [4,]                    -345078.1                    116599.81
##       [5,]                    -241227.9                    116904.20
##       [6,]                    -353190.2                    -36029.14
##           parameters
## iterations b[(Intercept) zipcode:98109] b[(Intercept) zipcode:98148]    sigma
##       [1,]                     98391.34                    -426525.7 450870.6
##       [2,]                    192382.06                    -411371.2 475565.6
##       [3,]                     58137.32                    -498705.6 466419.0
##       [4,]                     52251.08                    -452755.2 466798.0
##       [5,]                     96251.70                    -503230.2 459271.1
##       [6,]                    -44350.73                    -696508.7 458542.9
##           parameters
## iterations Sigma[zipcode:(Intercept),(Intercept)]
##       [1,]                           248000673221
##       [2,]                           269792218544
##       [3,]                           299394111338
##       [4,]                           283297434571
##       [5,]                           293207544285
##       [6,]                           515156734718
```

This can be used for generating predictions and credible intervals.

Final Connections **Group-level covariates**:

Interactions:

Shrinkage:

$$\hat{\alpha}_j \approx \frac{\frac{n_j}{\sigma_y^2}\bar{y}_j + \frac{1}{\sigma_\alpha^2}\bar{y}_{all}}{\frac{n_j}{\sigma_y^2} + \frac{1}{\sigma_\alpha^2}}$$

where σ_y^2 is the variance of the data and σ_α^2 is the variance of the group-level averages.

Selection of Random Effects: these varying effect models necessarily impose additional complexity on our modeling framework; however, GH suggest embracing the complexity (as it often helps directly answer research questions), moreover, they don't recommend using evidence statements to select specific random effects.