36-617: Applied Linear Models

Two Vignettes:
Sensitivity, and Correlated Eta’s

Brian Junker
132E Baker Hall
brian@stat.cmu.edu

Announcements

- **HW10** Due Weds 1159
- **Quiz08** Survey for credit: Feedback to me on the course (Due Tues 7pm as usual)
- **Last two classes**
  - Today: Two vignettes: Sensitivity, and Correlated Etas
  - Weds: Two vignettes: Meta Analysis and Extending Multilevel glm's
- **Final Papers** Due Friday 1159pm

Outline

- Vignette 1: Sensitivity Analysis
  - Example 1: Sensitivity analysis for prior on β_0
  - Example 2: Sensitivity analysis for distribution of η's
  - Example 3: Estimating the shape of distribution of η's
- Vignette 2: Correlated η's: lmer vs stan
  - Example 4: Naturally uncorrelated η's in stan
  - Example 5: Implementing correlated η's in stan
  - Building Covariance Matrices…

Sensitivity Analysis

- Examining how much estimates, inferences, and/or predictions change when we change modeling assumptions
  - George Box: *"All models are wrong; some are useful"*
  - Therefore, we should always check to see if we get the same answers from a different (wrong) model that is close to the (wrong) model we chose!
  - This is a good habit for any statistician or data scientist to acquire *(whether doing Bayes or anything else!)*
- You do not have to check everything
  - Check the things that will matter for your inferences

Example 1: Random intercept cd4…

- In the model

$$
\begin{aligned}
sqrt(CD4PCT)_i &= \alpha_{0j[i]} + \beta_1 VISIT_i + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} N(0, \sigma^2) \\
\alpha_{0j} &= \beta_0 + \eta_{0j}, \quad \eta_{0j} \overset{iid}{\sim} N(0, \tau_0^2) \\
\tau_0 &\sim Unif(0, 10) \\
\sigma &\sim Unif(0, 10) \\
\beta_0 &\sim N(\mu_{\beta_0}, \sigma_{\beta_0}) \\
\beta_1 &\sim N(0, 1000000)
\end{aligned}
$$

  we chose $\mu_{\beta_0} = 0$ and $\sigma_{\beta_0} = 1{,}000{,}000$.
- How much does changing these values affect parameter estimates in the model?

Example 1: “blmer_1.stan”

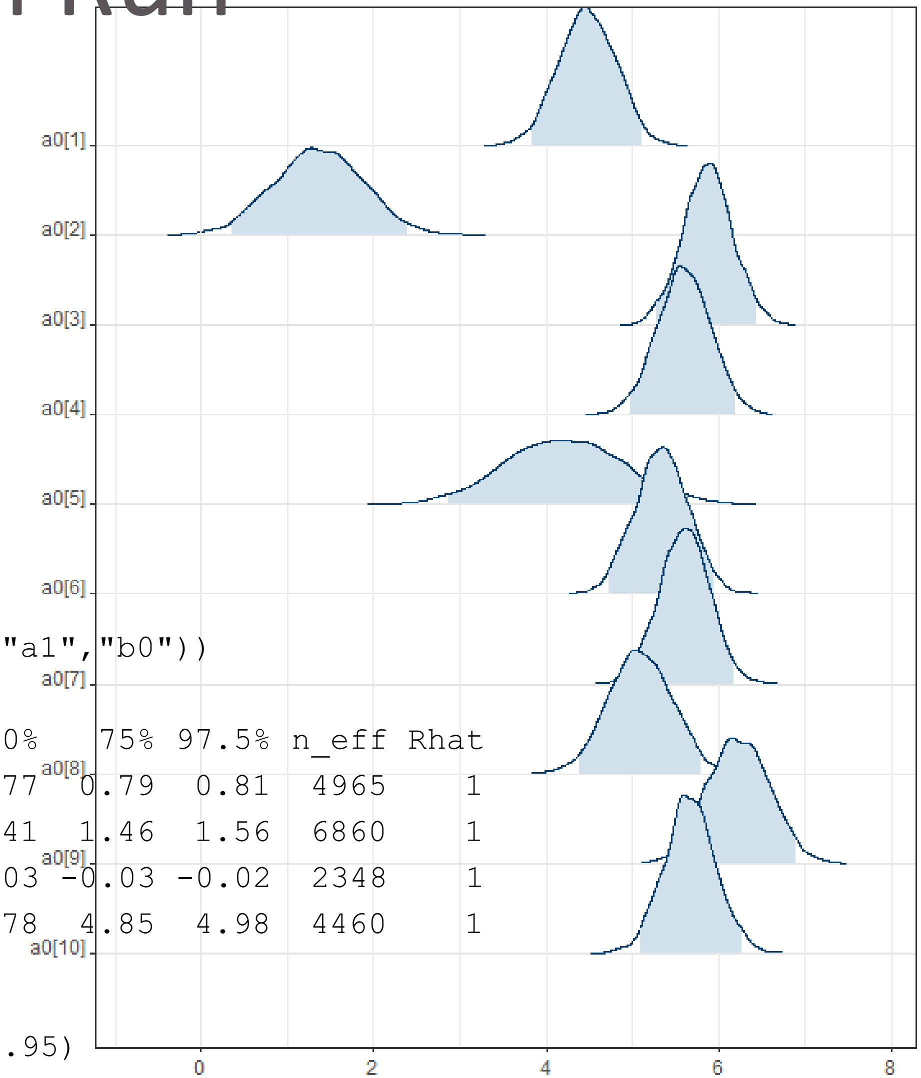
```
data {
    // data  needed for the model
    int N;
    real y[N];
    int newpid[N];
    real VISIT[N];
    int J;
    // mean and SD for beta0
    real mu_b0;
    real sig_b0;
}
parameters {
    real<lower=0, upper=10> sig;
    real a0[J];
    real a1;
    real b0;
    real<lower=0, upper=10> tau0;
}
```

```
model {
  real mu[N];

  // level 1: likelihood
  for (i in 1:N) {
    mu[i] <- a0[newpid[i]] + a1*VISIT[i];
    y[i] ~ normal(mu[i],sig);
  }

  // level 2: prior
  for (j in 1:J) {
    a0[j] ~ normal(b0,tau0);
  }

  // add'l priors
  b0 ~ normal(mu_b0,sig_b0);
  a1 ~ normal(0,1e+6);
  sig ~ uniform(0,10);
  tau0 ~ uniform(0,10);
}
```

We can try different mu_b0 and sig_b0 without recompiling the stan program

Example 1: Typical Run

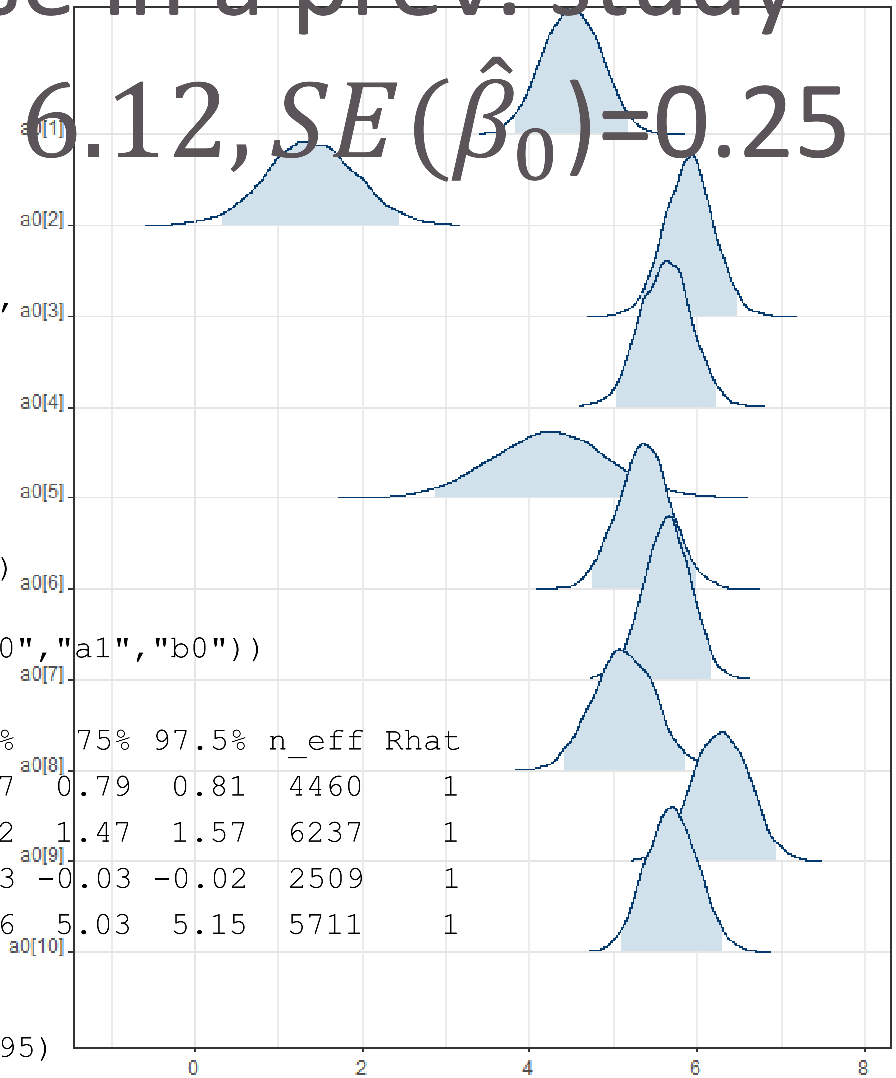
```
blmer_1.data <- c(as.list(cd4),
                 list(y=sqrt(cd4$CD4PCT),
                 N=length(cd4$CD4PCT),
                 J=max(cd4$newpid),
                 mu_b0=0, sig_b0=1e+6))
blmer_1.model <-
  stan(file="blmer_1.stan",
  data=blmer_1.data, chains=0)

blmer_1.results <-
  stan(fit=blmer_1.model,
  data=blmer_1.data)

print(blmer_1.results,pars=c("sig","tau0","a1","b0"))

##        mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## sig    0.77       0 0.02  0.74  0.76  0.77  0.79  0.81  4965    1
## tau0   1.41       0 0.07  1.28  1.36  1.41  1.46  1.56  6860    1
## a1    -0.03       0 0.00 -0.04 -0.03 -0.03 -0.03 -0.02  2348    1
## b0     4.78       0 0.10  4.58  4.71  4.78  4.85  4.98  4460    1

mcmc_areas_ridges(blmer_1.results,
  pars=paste0("a0[",1:10,"]"),prob_outer=0.95)
```



R code for running all the examples is in the file “sensitivity-and-correlated-etas.r”

Example 1: Suppose in a prev. study we obtained $\hat{\beta}_0 = 6.12, SE(\hat{\beta}_0)$=0.25

```
blmer_1.data <- c(as.list(cd4),
                  list(y=sqrt(cd4$CD4PCT),
                       N=length(cd4$CD4PCT),
                       J=max(cd4$newpid),
                       mu_b0=6.12,
                       sig_b0=0.25))


blmer_1.results.new <-
  stan(fit=blmer_1.model, data=blmer_1.data)

print(blmer_1.results.new,pars=c("sig","tau0","a1","b0"))

##         mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## sig     0.77       0 0.02  0.74  0.76  0.77  0.79  0.81  4460    1
## tau0    1.42       0 0.07  1.29  1.37  1.42  1.47  1.57  6237    1
## a1     -0.03       0 0.00 -0.04 -0.03 -0.03 -0.03 -0.02  2509    1
## b0      4.96       0 0.09  4.78  4.90  4.96  5.03  5.15  5711    1

mcmc_areas_ridges(blmer_1.results.new,
  pars=paste0("a0[",1:10,"]"),prob_outer=0.95)
```



Example 1: Sensitivity to $\mu_{\beta_0}, \sigma_{\beta_0}$

```
mb0 <- c(4,8,12)
sb0 <- c(0.1,0.2,0.4)
parlist <- c("sig","tau0","a1","b0")
res <- NULL
for (m in mb0) {
  for (s in sb0) {
    blmer_1.data <- c(as.list(cd4),
     list(y=sqrt(cd4$CD4PCT),
     N=length(cd4$CD4PCT),
     J=max(cd4$newpid),
     mu_b0=m,sig_b0=s))
    tmp.fit <-
     stan(fit=blmer_1.model,
      data=blmer_1.data)
     tmp.res <- summary(tmp.fit)$
      summary[parlist,c(1,3)]
     row.names(tmp.res) <-
      paste("m =",m,"s =",s,":",
         row.names(tmp.res))
     res <- rbind(res,tmp.res)
  }}
round(res,4)
```

	mean	sd
m = 4 s = 0.1 : sig	0.7742	0.0190
m = 4 s = 0.1 : tau0	1.4546	0.0734
m = 4 s = 0.1 : a1	-0.0242	0.0044
m = 4 s = 0.1 : b0	4.3869	0.0745
m = 4 s = 0.2 : sig	0.7743	0.0195
m = 4 s = 0.2 : tau0	1.4178	0.0702
m = 4 s = 0.2 : a1	-0.0273	0.0044
m = 4 s = 0.2 : b0	4.6281	0.0908
m = 4 s = 0.4 : sig	0.7741	0.0194
m = 4 s = 0.4 : tau0	1.4125	0.0693
m = 4 s = 0.4 : a1	-0.0289	0.0044
m = 4 s = 0.4 : b0	4.7361	0.0930
m = 8 s = 0.1 : sig	0.7763	0.0193
m = 8 s = 0.1 : tau0	2.7867	0.1637
m = 8 s = 0.1 : a1	-0.0413	0.0045
m = 8 s = 0.1 : b0	7.2429	0.1087
m = 8 s = 0.2 : sig	0.7755	0.0194
m = 8 s = 0.2 : tau0	1.5571	0.0880
m = 8 s = 0.2 : a1	-0.0384	0.0044
m = 8 s = 0.2 : b0	5.4943	0.1087

	mean	sd
m = 8 s = 0.4 : sig	0.7739	0.0189
m = 8 s = 0.4 : tau0	1.4234	0.0690
m = 8 s = 0.4 : a1	-0.0322	0.0045
m = 8 s = 0.4 : b0	4.9681	0.0962
m = 12 s = 0.1 : sig	0.7735	0.0193
m = 12 s = 0.1 : tau0	7.0191	0.3339
m = 12 s = 0.1 : a1	-0.0356	0.0044
m = 12 s = 0.1 : b0	11.6489	0.1042
m = 12 s = 0.2 : sig	0.7738	0.0192
m = 12 s = 0.2 : tau0	5.6746	0.3335
m = 12 s = 0.2 : a1	-0.0368	0.0045
m = 12 s = 0.2 : b0	10.2884	0.2302
m = 12 s = 0.4 : sig	0.7744	0.0194
m = 12 s = 0.4 : tau0	1.4688	0.0770
m = 12 s = 0.4 : a1	-0.0354	0.0046
m = 12 s = 0.4 : b0	5.2186	0.1062

- As μ_{β_0} gets further from 4.78, $\hat{\beta}_0$ gets dragged too
- The effect gets weaker as σ_{β_0} grows
- τ_0 also affected; other parameters not so much…

Example 2: Sensitivity to shape of η distribution

- Suppose there are outliers that we can't get rid of, but they affect the values of α_{0j} and β_0
- We could replace $\eta_{0j} \sim N(0, \tau_0^2)$ with $\eta_{0j} \sim t_{df}(0, \tau_0^2)$, a scaled t-distribution
  - The t_{df} distribution has fatter tails than the normal
    - Outliers will affect means less
    - Since the α_{0j} will vary less, τ_0 should be smaller
    - Maybe σ^2 will be smaller too?
  - Try this for several different *df* values…

lmer() can't do this!)

Example 2: blmer_1_t_prior.stan

```
model {
    real mu[N];
  // level 1: likelihood
  for (i in 1:N) {
    mu[i] <- a0[newpid[i]] + a1*VISIT[i];
    y[i] ~ normal(mu[i],sig);
  }

  // level 2: prior
  for (j in 1:J) {
    a0[j] ~ student_t(df,b0,tau0);
  }

  // add'l priors
  b0 ~ normal(0,1e+6);
  a1 ~ normal(0,1e+6);
  sig ~ uniform(0,10);
  tau0 ~ uniform(0,10);
}
```

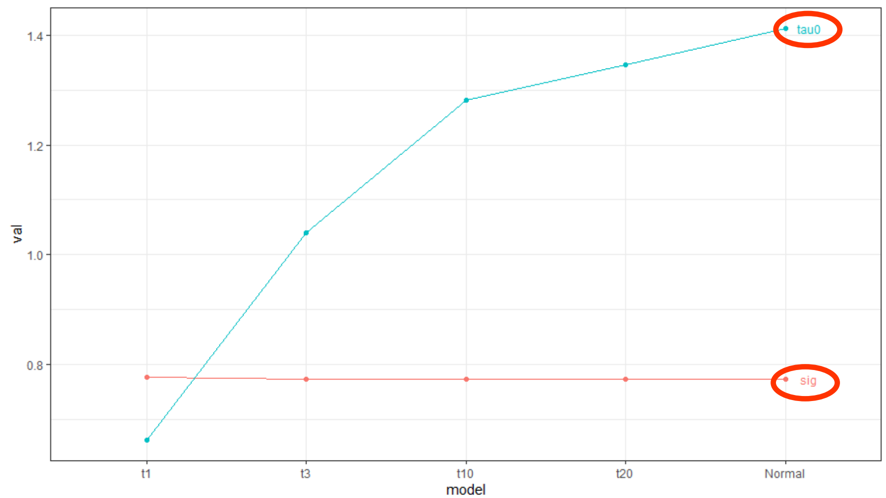
```
blmer_t_prior.data <- c(as.list(cd4),
  list(y=sqrt(cd4$CD4PCT),
  N=length(cd4$CD4PCT),
  J=max(cd4$newpid),
  df=1))
blmer_t_1_prior <-
  stan(fit=blmer_t_prior.model,
  data=blmer_t_prior.data)

blmer_t_prior.data <- . . . df=3 . . .
blmer_t_3_prior <-
  stan(fit=blmer_t_prior.model,
  data=blmer_t_prior.data)

blmer_t_prior.data <- . . . df=10  . .
blmer_t_10_prior <-
stan(fit=blmer_t_prior.model,
  data=blmer_t_prior.data)

blmer_t_prior.data <- . . . df=20 . . .
blmer_t_20_prior <-
  stan(fit=blmer_t_prior.model,
  data=blmer_t_prior.data)
```

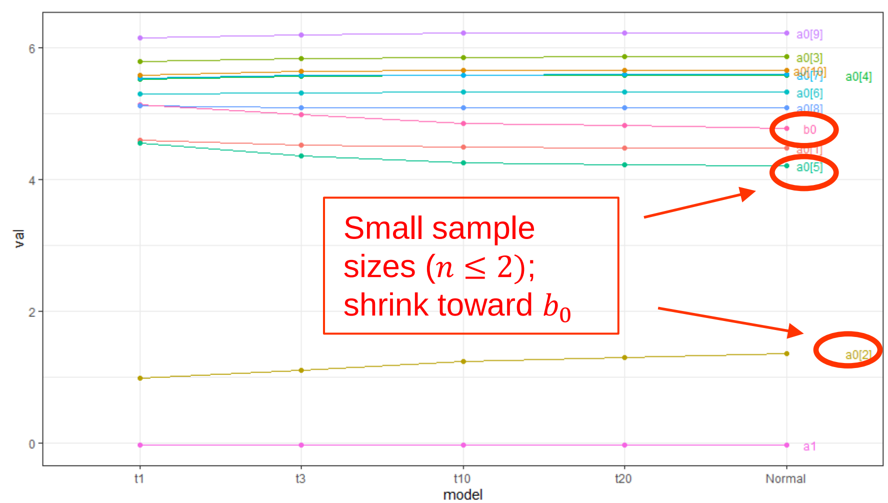
Example 2: Sensitivity to shape of η distribution

```
## R code to set up the table...
```

	t1	t3	t10	t20	Normal
sig	0.78	0.77	0.77	0.77	0.77
tau0	0.66	1.04	1.28	1.35	1.41
a1	-0.03	-0.03	-0.03	-0.03	-0.03
b0	5.14	4.99	4.86	4.82	4.78
a0[1]	4.60	4.52	4.49	4.49	4.48
a0[2]	0.99	1.11	1.25	1.29	1.36
a0[3]	5.80	5.84	5.86	5.87	5.87
a0[4]	5.53	5.57	5.58	5.59	5.60
a0[5]	4.55	4.37	4.26	4.22	4.21
a0[6]	5.30	5.33	5.34	5.34	5.34
a0[7]	5.54	5.58	5.59	5.60	5.60
a0[8]	5.12	5.10	5.10	5.10	5.09
a0[9]	6.16	6.20	6.22	6.23	6.23
a0[10]	5.59	5.65	5.66	5.66	5.67

```
## ggplot stuff...
```





Example 3: Estimating shape of η distribution

- Because of the flexibility of Bayesian model-building, we can also consider

$$
\begin{aligned}
sqrt(CD4PCT)_i &= \alpha_{0j[i]} + \beta_1 VISIT_i + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} N(0, \sigma^2) \\
\alpha_{0j} &= \beta_0 + \eta_{0j}, \quad \boldsymbol{\eta_{0j} \overset{iid}{\sim} t_{df}(0, \tau_0^2)} \\
\tau_0 &\sim Unif(0, 10) \\
\sigma &\sim Unif(0, 10) \\
\beta_0 &\sim N(0, 1000000) \\
\beta_1 &\sim N(0, 1000000) \\
\mathbf{df} &\sim \mathbf{Expon(1)}
\end{aligned}
$$

and estimate *df* along with all of the other parameters in the model

- *Let the data choose the η distribution!*

lmer() can't do this!)

Example 3: blmer_1_df_est.stan

```
model {
  real mu[N];

  // level 1: likelihood
  for (i in 1:N) {
    mu[i] <- a0[newpid[i]] + a1*VISIT[i];
    y[i] ~ normal(mu[i],sig);
  }

  // level 2: prior
  for (j in 1:J) {
    a0[j] ~ student_t(df,b0,tau0);
  }

  // add'l priors
  b0 ~ normal(0,1e+6);
  a1 ~ normal(0,1e+6);
  sig ~ uniform(0,10);
  tau0 ~ uniform(0,10);
  df ~ gamma(1,1);  // exponential dist.
}
```

Looks like df≈5 is about right…

```
blmer_df_est.data <- c(as.list(cd4),
  list(y=sqrt(cd4$CD4PCT),
  N=length(cd4$CD4PCT),
  J=max(cd4$newpid))

blmer_df_est.model <-
  stan(file="blmer_1_df_est.stan",
  data=blmer_df_est.data,chains=0)

blmer_df_est<-
  stan(fit=blmer_df_est.model,
  data=blmer_df_est.data)

print(blmer_df_est,par=c("df",parlist))
         mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
df       4.79    0.03 1.46  2.62  3.77  4.56  5.56  8.32  2915    1
sig      0.77    0.00 0.02  0.74  0.76  0.77  0.79  0.81  4520    1
tau0     1.14    0.00 0.10  0.95  1.08  1.14  1.21  1.33  3740    1
a1      -0.03    0.00 0.00 -0.04 -0.03 -0.03 -0.03 -0.02  1961    1
b0       4.94    0.00 0.10  4.74  4.87  4.94  5.01  5.14  3715    1
a0[1]    4.51    0.00 0.32  3.87  4.29  4.51  4.73  5.14  9152    1
a0[2]    1.16    0.01 0.56  0.07  0.78  1.15  1.54  2.25  8510    1
a0[3]    5.85    0.00 0.29  5.30  5.65  5.85  6.06  6.43  8229    1
a0[4]    5.58    0.00 0.30  4.99  5.38  5.58  5.79  6.19  7024    1
a0[5]    4.32    0.01 0.64  3.00  3.92  4.32  4.74  5.54  8011    1
a0[6]    5.33    0.00 0.31  4.73  5.12  5.33  5.53  5.96  7266    1
a0[7]    5.59    0.00 0.29  5.01  5.40  5.58  5.78  6.15  8496    1
a0[8]    5.09    0.00 0.36  4.37  4.86  5.10  5.33  5.81  7823    1
a0[9]    6.21    0.00 0.35  5.55  5.97  6.21  6.45  6.89  6648    1
a0[10]   5.66    0.00 0.30  5.07  5.45  5.65  5.86  6.26  8797    1
```

You would probably want to try some different priors on df to see how much the estimated df is affected (sensitivity analysis again!).

Correlated η's: lmer vs stan

- For the random-intercepts/random-slopes model

$$
\begin{aligned}
sqrt(CD4PCT)_i &= \alpha_{0j[i]} + \alpha_{1j[i]} VISIT_i + \epsilon_i, \quad \epsilon_i \stackrel{iid}{\sim} N(0, \sigma^2) \\
\alpha_{0j} &= \beta_0 + \eta_{0j} \\
\alpha_{1j} &= \beta_1 + \eta_{1j} \\
\begin{pmatrix} \eta_{0j} \\ \eta_{1j} \end{pmatrix} &\sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_0^2 & \tau_0\tau_1\rho_{01} \\ \tau_1\tau_0\rho_{01} & \tau_1^2 \end{pmatrix} \right)
\end{aligned}
$$

  it's generally better to estimate ρ_{01} rather than forcing $\rho_{01} \equiv 0$.
  - This happens "automatically" with

    ```
    lmer(y ~ 1 + VISIT + (1 + VISIT | newpid))
    ```

  - OTOH Stan assumes independence unless you build ρ_{01} into the model!

Example 4: Uncorrelated η's in stan

```
model {
  real mu[N];

  // level 1: likelihood
  for (i in 1:N) {
    mu[i] <- a0[newpid[i]] +
             a1[newpid[i]]*VISIT[i];
    y[i] ~ normal(mu[i],sig);
  }

  // level 2: prior
  for (j in 1:J) {
    a0[j] ~ normal(b0,tau0);
    a1[j] ~ normal(b1,tau1);
  }

  // add'l priors
  b0 ~ normal(0,1e+6);
  b1 ~ normal(0,1e+6);
  sig ~ uniform(0,10);
  tau0 ~ uniform(0,10);
  tau1 ~ uniform(0,10);
}
```

- This model corresponds to the lmer model

```
lmer(y ~ 1 + VISIT + (1|newpid) +
                 (0+VISIT|newpid))
```

- Since we didn't specify a correlation in the stan model, stan assumes the correlation is zero.
- We'll put ρ in the model, and build up the covariance matrix from τ_0^2, τ_1^2, and $\tau_{01} = \tau_0 \tau_1 \rho$

This stan model is in the file "blmer_2_indep.stan"

R code for running this stan model and comparing it with corresponding lmer() models is in the file "sensitivity-and-correlated-etas.r"

Example 5: Correlated η's in stan

```
data {
    int N;
    real y[N];
    int newpid[N];
    real VISIT[N];
    int J;
}
parameters {
    real<lower=0, upper=10> sig;
    matrix[J,2] A;
    real b0;
    real b1;
    real<lower=0, upper=10> tau0;
    real<lower=0, upper=10> tau1;
    real<lower=-1,upper=+1> rho;
}
```

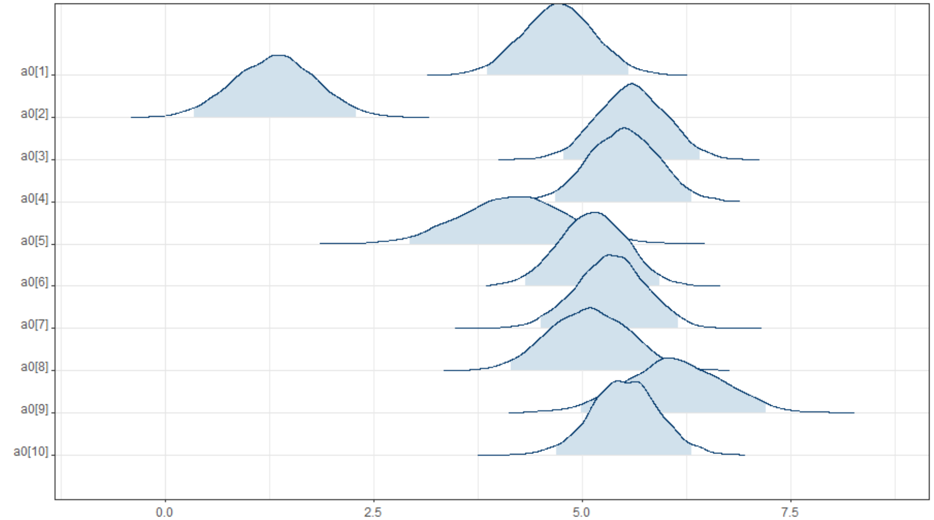
```
transformed parameters {
    real a0[J];
    real a1[J];
    vector[2] B;
    matrix[2,2] Tau;

    for (j in 1:J) {
      a0[j] <- A[j,1];
      a1[j] <- A[j,2];
    }

    B[1] <- b0;
    B[2] <- b1;
    Tau[1,1] <- tau0^2;
    Tau[2,2] <- tau1^2;
    Tau[1,2] <- rho*tau0*tau1;
    Tau[2,1] <- Tau[1,2];
}
```

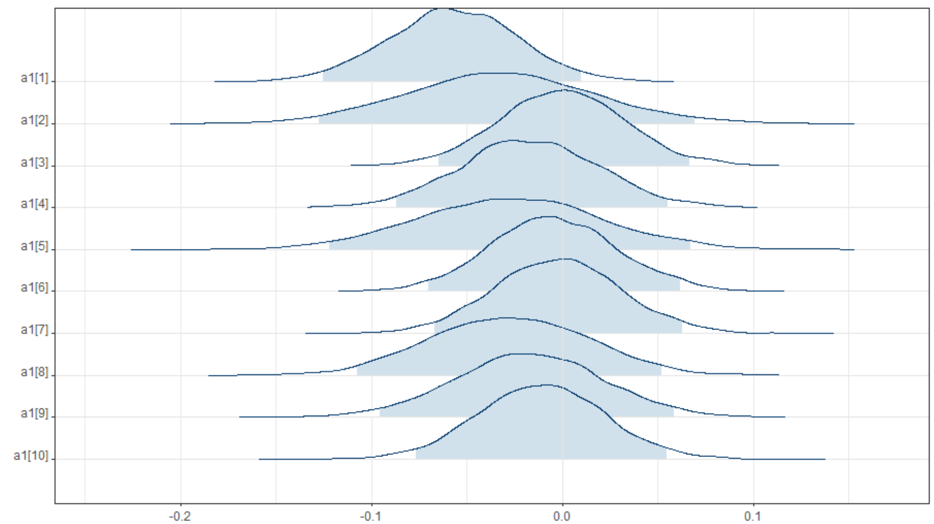
```
model {
  real mu[N];

  // level 1: likelihood
  for (i in 1:N) {
    mu[i] <- a0[newpid[i]] +
        a1[newpid[i]]*VISIT[i];
    y[i] ~ normal(mu[i],sig);
  }

  // level 2: prior
  for (j in 1:J) {
    A[j,1:2] ~
      multi_normal(B,Tau);
  }

  // add'l priors
  b0 ~ normal(0,1e+6);
  b1 ~ normal(0,1e+6);
  sig ~ uniform(0,10);
  tau0 ~ uniform(0,10);
  tau1 ~ uniform(0,10);
  rho ~ uniform(-1,1);
}
```


This stan model is in the file “blmer_2.stan”

Example 5: Correlated η's in stan

```
blmer_2.data <- c(as.list(cd4),
  list(y=sqrt(cd4$CD4PCT),
  N=length(cd4$CD4PCT),
  J=max(cd4$newpid)))


blmer_2.model <-stan(file="blmer_2.stan",
  data=blmer_2.data, chains=0)


blmer_2.results <-stan(fit=blmer_2.model,
  data=blmer_2.data)


print(blmer_2.results,pars=parlist)
 mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
sig     0.72    0.00 0.02  0.68  0.70  0.72  0.73  0.76  2136 1.00
tau0    1.40    0.00 0.07  1.26  1.35  1.40  1.45  1.56  2580 1.00
tau1    0.05    0.00 0.01  0.03  0.04  0.05  0.05  0.06   452 1.01
rho    -0.07    0.00 0.13 -0.31 -0.17 -0.07  0.02  0.19   987 1.01
b0      4.77    0.00 0.10  4.58  4.71  4.77  4.84  4.97  5321 1.00
b1     -0.03    0.00 0.01 -0.04 -0.03 -0.03 -0.02 -0.02  2489 1.00
display(lmer(y~VISIT + (VISIT|newpid))
           coef.est  coef.se
(Intercept)  4.77      0.10
VISIT       -0.03      0.01

Groups   Name        Std.Dev. Corr
 newpid   (Intercept) 1.40
          VISIT       0.05     -0.09
 Residual             0.72
## bayesplot stuff…
```





Building Covariance Matrices "by Hand"…

- The basic idea was to recognize that if we have $K + 1 \ \ \eta' s$, then

$$\begin{pmatrix} \tau_0^2 & \tau_{01} & \cdots & \tau_{0K} \\ \tau_{10} & \tau_1^2 & \cdots & \tau_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ \tau_{K0} & \tau_{K1} & \cdots & \tau_K^2 \end{pmatrix} = \begin{pmatrix} \tau_0^2 & \tau_0\tau_1\rho_{01} & \cdots & \tau_0\tau_K\rho_{0K} \\ \tau_1\tau_0\rho_{01} & \tau_1^2 & \cdots & \tau_1\tau_K\rho_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ \tau_K\tau_0\rho_{0K} & \tau_K\tau_1\rho_{1K} & \cdots & \tau_K^2 \end{pmatrix}$$

  where $\tau_j^2 = Var(\eta_j)$, $\rho_{jk} = Corr(\eta_j, \eta_k)$, and then put uniform priors on all τ_j's and ρ_{jk}'s.
- This "should work" but can sometimes produce unworkable sets of correlations (see next slide)…

Building Covariance Matrices…

- The "by-hand" approach above can have "bad initialization" in stan…
  - By default, stan initializes parameters essentially from prior
  - Some sets of ρ's initialized this way are impossible[1] as correlations.
    - E.g. can't have $\rho_{12} = -1, \;\; \rho_{23} = -1 \;\; and \;\; \rho_{13} = -1 \ldots$
  - Stan offers a more principled approach that won't fail; see
    - https://stla.github.io/stlapblog/posts/StanLKJprior.html
    - https://arxiv.org/abs/1506.06201v1
    - https://mc-stan.org/docs/2_25/reference-manual/correlation-matrix-transform-section.html
  - Gelman & Hill discuss a different approach, using the "Wishart distribution" (like a matrix version of the chi-squared or gamma distribution)

[1]See for example https://stats.stackexchange.com/questions/72790/bound-for-the-correlation-of-three-random-variables or https://www.jstor.org/stable/23020502

Summary

- Vignette 1: Sensitivity Analysis
  - Example 1: Sensitivity analysis for prior on β_0
  - Example 2: Sensitivity analysis for distribution of η's
  - Example 3: Estimating the shape of distribution of η's
- Vignette 2: Correlated η's: lmer vs stan
  - Example 4: Naturally uncorrelated η's in stan
  - Example 5: Implementing correlated η's in stan
  - Building Covariance Matrices…