LONGITUDINAL ANALYSIS IN R

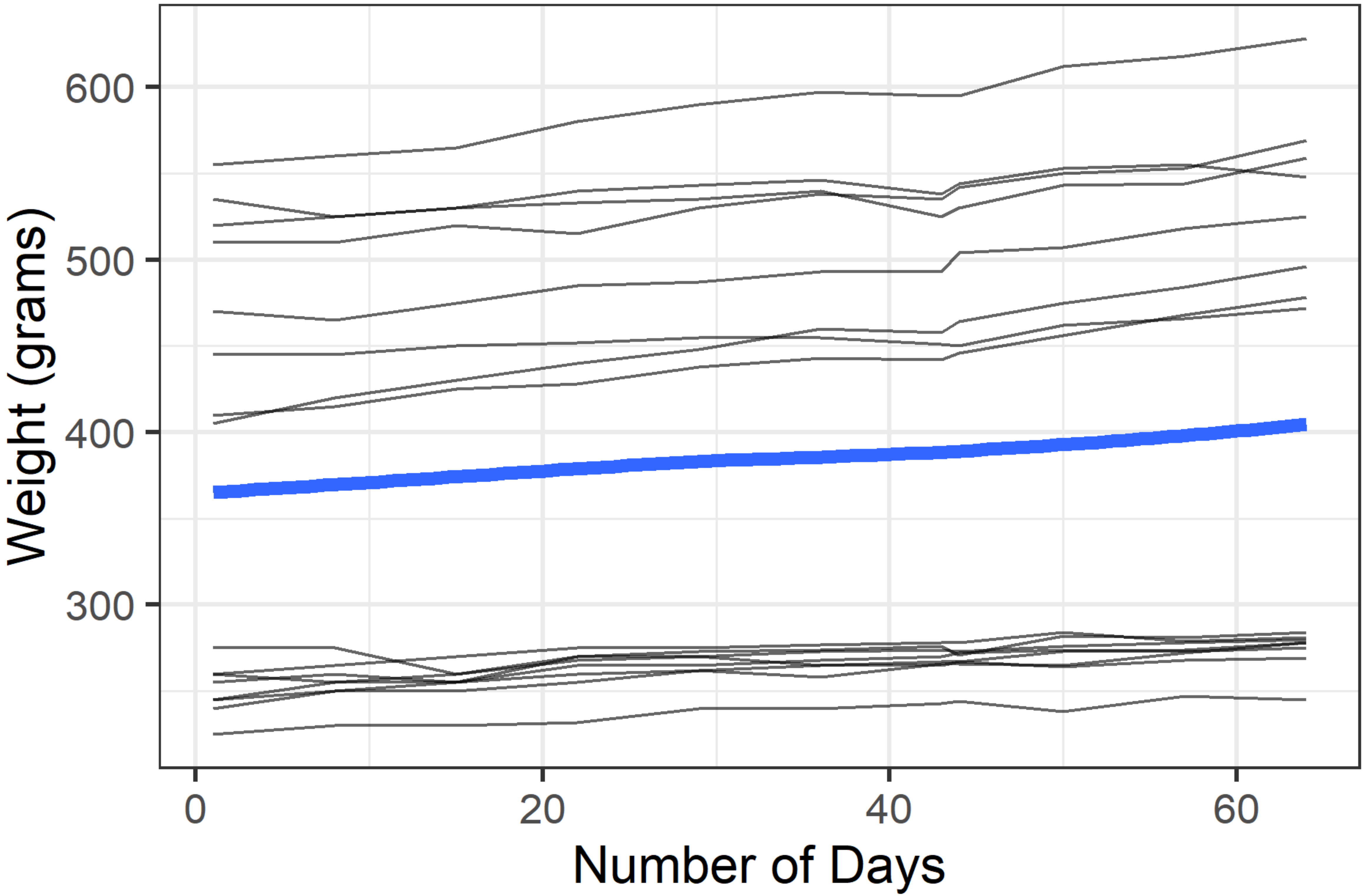
Longitudinal Analysis for Continuous Outcomes

Brandon LeBeau
Assistant Professor

Visualizing longitudinal data code

- Let's explore the data!

```
library(nlme)

ggplot(BodyWeight, aes(x = Time, y = weight)) +
     geom_line(aes(group = Rat), alpha = 0.6) +
     geom_smooth(se = FALSE, size = 2) +
     theme_bw(base_size = 16) +
     xlab("Number of Days") +
     ylab("Weight (grams)")
```


Weight (grams)
600
500
400
300
0
20
40
60
Number of Days

Introducing the lmer function

- `lmer` stands for **L**inear **M**ixed **E**ffects **R**egression
- Used for continuous outcomes
- Other names:
  - Hierarchical linear models
  - Linear mixed models
  - Multi-level models
  - Growth models
- `lmer` arguments :

```
lmer(outcome ~ fixed_effects + (random_effects | individual), data = data)
```

lmer formula

- Previous formula:

```
outcome ~ fixed_effects + (random_effects | individual)
```

- `outcome` = the variable we wish to explain or predict
- `fixed_effects` = terms representing the average trajectory
- `random_effects` = reflect deviations from the average trajectory for each individual
- `individual` = ID for individuals that were measured repeatedly

lmer random intercept example

```
library(nlme)
library(dplyr)
library(lme4)

BodyWeight <- mutate(BodyWeight, Time = Time - 1)

body_ri <- lmer(weight ~ 1 + Time + (1 | Rat), data = BodyWeight)

summary(body_ri)
```

lmer summary output

```
Linear mixed model fit by REML ['lmerMod']
Formula: weight ~ 1 + Time + (1 | Rat)
   Data: BodyWeight
REML criterion at convergence: 1360.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.5029 -0.5458 -0.0394  0.5608  3.1139

Random effects:
 Groups   Name        Variance Std.Dev.
 Rat      (Intercept) 16940.81 130.157
 Residual                66.85   8.176
Number of obs: 176, groups:  Rat, 16

Fixed effects:
             Estimate Std. Error t value
(Intercept) 365.42163   32.56138   11.22
Time          0.58568    0.03168   18.49
Correlation of Fixed Effects:
     (Intr)
Time -0.032
```

Exploring output - random effects

- Variability estimates for random intercept and the random within rat error (Residual)

```
Random effects:
 Groups   Name        Variance Std.Dev.
 Rat      (Intercept) 16940.81 130.157
 Residual                66.85   8.176
Number of obs: 176, groups:  Rat, 16
```

Exploring output - fixed effects

- Average starting place (Intercept) and average change in weight for each day

```
Fixed effects:
             Estimate Std. Error t value
(Intercept) 365.42163   32.56138   11.22
Time          0.58568    0.03168   18.49

Correlation of Fixed Effects:
     (Intr)
Time -0.032
```

LONGITUDINAL ANALYSIS IN R

Let's practice!

LONGITUDINAL ANALYSIS IN R

Addition of Random Slope Terms

Brandon LeBeau
Assistant Professor

What are random slopes?

- Allow each individual to have their own trajectory
- Random intercept formula:

```
weight ~ 1 + Time + (1 | Rat)
```

- To add a random slope, add another term to the formula in parentheses:

```
weight ~ 1 + Time + (1  + Time | Rat)
```

Random slope with lmer

```
library(nlme)
library(dplyr)
library(lme4)

BodyWeight <- mutate(BodyWeight, Time = Time - 1)

body_rs <- lmer(weight ~ 1 + Time + (1 + Time | Rat), data = BodyWeight)

summary(body_rs)
```

Random slope summary output

```
Linear mixed model fit by REML ['lmerMod']
Formula: weight ~ 1 + Time + (1 + Time | Rat)
   Data: BodyWeight

REML criterion at convergence: 1208.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2658 -0.4256  0.0711  0.5871  2.7485

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Rat      (Intercept) 15246.80 123.4779
          Time            0.12   0.3463 0.56
 Residual                19.75   4.4436
Number of obs: 176, groups:  Rat, 16

Fixed effects:
             Estimate Std. Error t value
(Intercept) 365.42163   30.87638  11.835
Time          0.58568    0.08828   6.634
```

Random effects output

```
Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Rat      (Intercept) 15246.80 123.4779
          Time            0.12   0.3463 0.56
 Residual                19.75   4.4436
Number of obs: 176, groups:  Rat, 16
```

- Time term indicates small variation in individual rat trajectories
- Variation in random slopes is much smaller compared to the intercepts
    - Common in longitudinal data

Which model fits the data better?

- `anova()` function compares models
- Fit indices of the `anova()` function:
  - Akaike Information Criterion (AIC): smaller is better
  - Bayesian Information Criterion (BIC): smaller is better
- Log Likelihood: value minimized during estimation
- AIC recommended when true model is not included in the comparison
- `anova()` function also performs nested model comparisons

Comparing models

```
body_ri <- lmer(weight ~ 1 + Time + (1 | Rat), data = BodyWeight)
body_rs <- lmer(weight ~ 1 + Time + (1 + Time | Rat), data = BodyWeight)

anova(body_rs, body_ri)
```

```
refitting model(s) with ML (instead of REML)
Data: BodyWeight
Models:
body_ri: weight ~ 1 + Time + (1 | Rat)
body_rs: weight ~ 1 + Time + (1 + Time | Rat)
        Df    AIC    BIC  logLik deviance Chisq Chi Df Pr(>Chisq)
body_ri  4 1372.0 1384.7 -682.00   1364.0
body_rs  6 1225.7 1244.7 -606.85   1213.7 150.3      2  < 2.2e-16 ***
```

LONGITUDINAL ANALYSIS IN R

Let's practice!

LONGITUDINAL ANALYSIS IN R

Visualize and Interpret Output

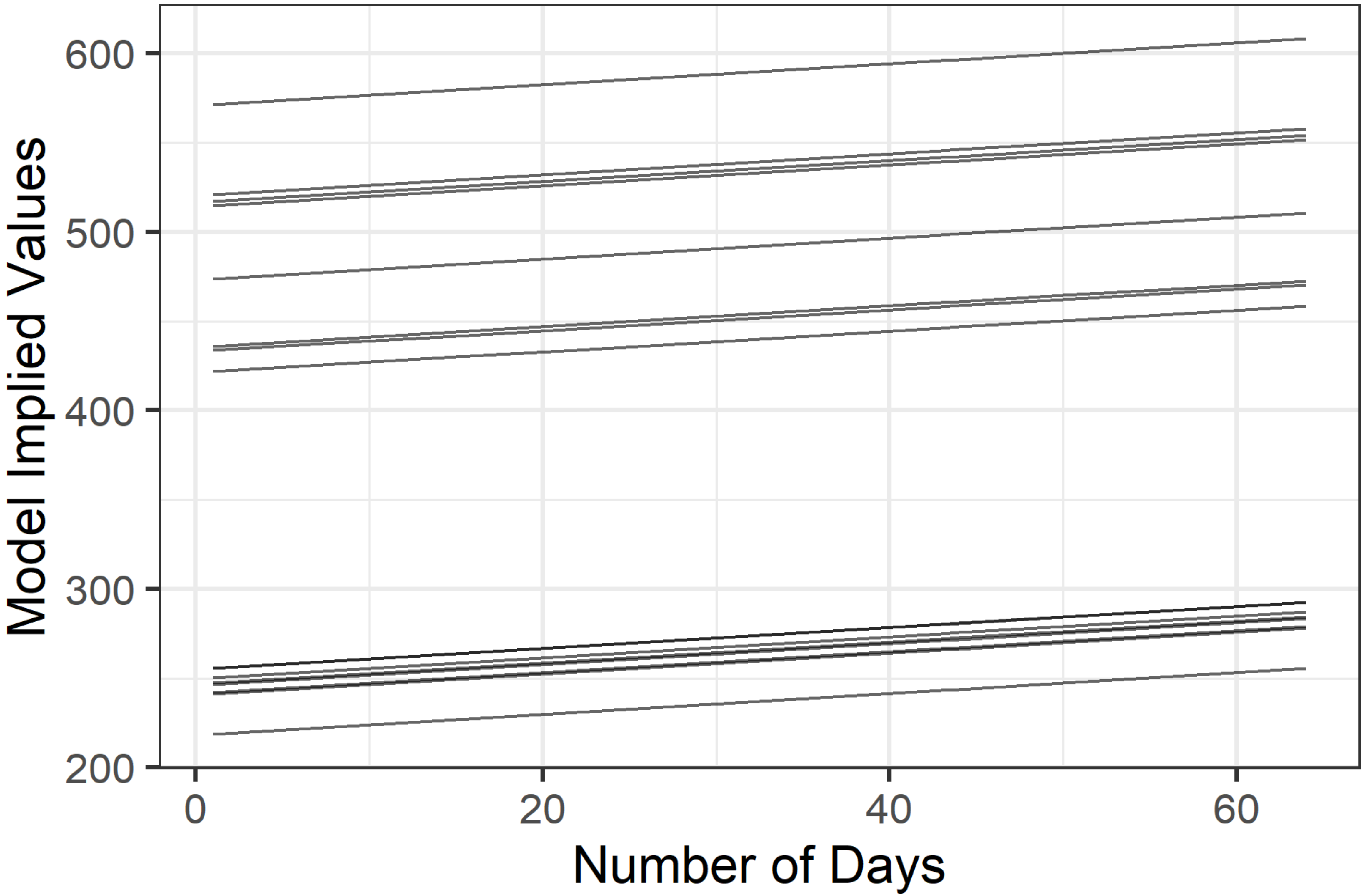
Brandon LeBeau
Assistant Professor

Generate predicted values

- `predict()` function can generate predicted values based on the model
- `predict()` and `mutate()` from dplyr will add a new column

```
library(nlme)

body_agg <- BodyWeight %>%
   mutate(pred_values = predict(body_ri))
```

```
# Visualize predicted values
ggplot(body_agg, aes(x = Time, y = pred_values)) +
   geom_line(aes(group = Rats), alpha = 0.6) +
   theme_bw(base_size = 16) +  # changes default theme
   xlab("Number of Days") +  # changes x-axis label
   ylab("Model Implied Values")   # changes y-axis label
```


Model Implied Values
600
500
400
300
200
0
20
40
60
Number of Days

How does lmer adjust for data dependency?

- Random effects help control dependency
- Custom function (next slide) will help explore model implied correlations

Correlation structure function

```
corr_structure <- function(object, num_timepoints, intercept_only = TRUE) {

  variance <- VarCorr(object)

  if(intercept_only) {
    random_matrix <- as.matrix(object@pp$X[1:num_timepoints, 1])

   var_cor <- random_matrix %*% variance[[1]][1] %*% t(random_matrix) +
      diag(attr(variance, "sc")^2, nrow = num_timepoints,
          ncol = num_timepoints)
  } else {
    random_matrix <- as.matrix(object@pp$X[1:num_timepoints, ])

    var_cor <- random_matrix %*% variance[[1]][1:2, 1:2] %*%
          t(random_matrix) + diag(attr(variance, "sc")^2,
             nrow = num_timepoints, ncol = num_timepoints)
  }

  Matrix::cov2cor(var_cor)

}
```

Correlation structure for a single rat

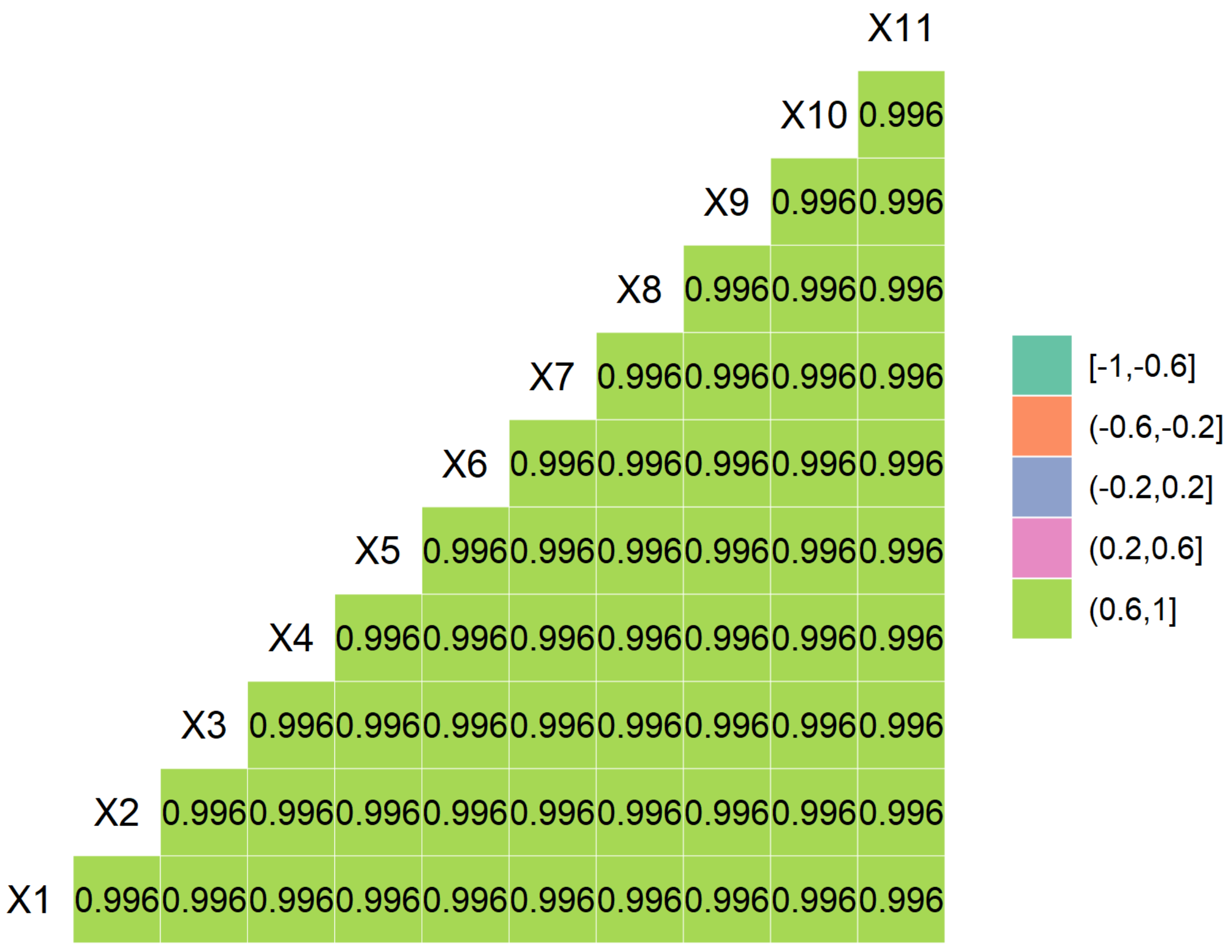
```
body_ri <- lmer(weight ~ 1 + Time + (1 | Rat), data = BodyWeight)

corr_structure(body_ri, 11) %>%
   round(3)
```

- Compound Symmetry

```
       1     2     3     4     5     6     7     8     9    10    11
1  1.000 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996
2  0.996 1.000 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996
3  0.996 0.996 1.000 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996
4  0.996 0.996 0.996 1.000 0.996 0.996 0.996 0.996 0.996 0.996 0.996
5  0.996 0.996 0.996 0.996 1.000 0.996 0.996 0.996 0.996 0.996 0.996
6  0.996 0.996 0.996 0.996 0.996 1.000 0.996 0.996 0.996 0.996 0.996
7  0.996 0.996 0.996 0.996 0.996 0.996 1.000 0.996 0.996 0.996 0.996
8  0.996 0.996 0.996 0.996 0.996 0.996 0.996 1.000 0.996 0.996 0.996
9  0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 1.000 0.996 0.996
10 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 1.000 0.996
11 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 0.996 1.000
```

Visually show dependency

```
library(GGally)
ggcorr(data = NULL, cor_matrix = corr_structure(body_ri, 11),
  label = TRUE, label_round = 3, label_size = 3.5, palette = 'Set2',
  nbreaks = 5)
```



LONGITUDINAL ANALYSIS IN R

Let's practice!