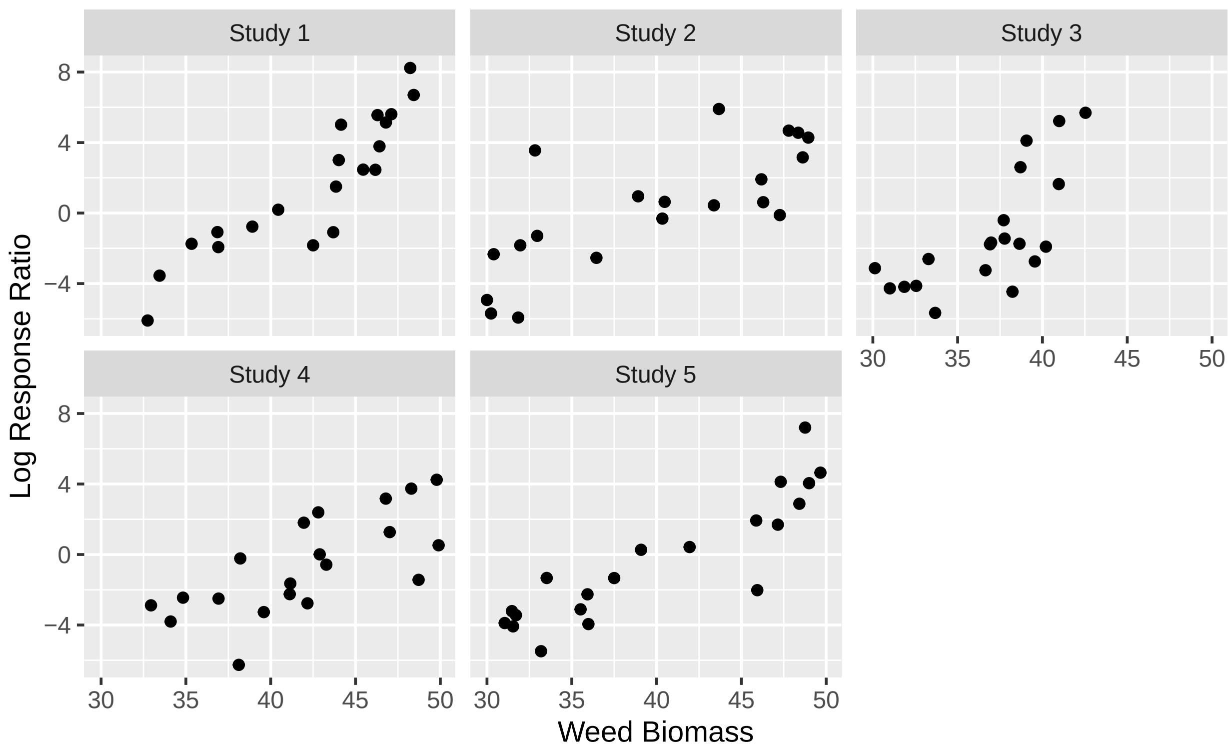
Standard Error for Inverse Prediction with Random Effect Models (no intercept model)

AES Statistical Consulting
Consultant: Katherine Goode
Client: Gina Nichols, graduate student in agronomy (vnichols@iastate.edu)
PI: Matt Liebman (mliebman@iastate.edu)
December 06, 2019

Data, Model, and Inverse Prediction

Gina is now fitting a model with no intercept. Again, I will work with the data I generated previously. The plot of the data is shown below.



The model that Gina is using has the form

$$\text{LRR}_i = \beta_1 \text{weed biomass}_i + \alpha_i + \epsilon_i$$

where $\alpha_i \sim N\left(0, \sigma^2_{study}\right)$ is a study random effect and $\epsilon_i \sim N\left(0, \sigma^2_{error}\right)$ is the error term for $i = 1, ..., n$. I fit this model to the example data. The code and summary of the model are included below.

```
m <- lmer(lrr ~ 0 + weed_biomass + (1|study), data = ex_data)
summary(m)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: lrr ~ 0 + weed_biomass + (1 | study)
##    Data: ex_data
##
```

```
## REML criterion at convergence: 474.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.05544 -0.56260 -0.06481  0.57581  3.02997
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  study    (Intercept) 334.430  18.29
##  Residual               4.492   2.12
## Number of obs: 100, groups:  study, 5
##
## Fixed effects:
##              Estimate Std. Error t value
## weed_biomass   0.4511     0.0371   12.16
```

Gina is interested in predicting the weed biomass given a LRR. In particular, she is interested in the case when LRR = -0.69, which corresponds to 50% weed control. Additionally, she would like the standard error associated with the predicted weed biomass. The estimate of the weed biomass for a given LRR can be computed using inverse prediction as

$$\hat{p}_{\text{weed control}} = \frac{\text{LRR}}{\hat{\beta}_1}.$$

In order to obtain the standard error for $\hat{p}_{\text{weed control}}$, we will use the delta method.

Notation and Derivatives for the Weed Biomass Problem

In this case, $\boldsymbol{\theta} = (\beta_1)$ and

$$g(\boldsymbol{\theta}) = \frac{\text{LRR}}{\beta_1}.$$

To obtain $\mathbf{d}$, we need to compute the derivative of $g(\boldsymbol{\theta})$ in terms of β_1. This can be computed as follows.

$$\begin{aligned}
\frac{dg(\boldsymbol{\theta})}{d\beta_1} &= \frac{d}{d\beta_1}\left(\frac{LRR}{\beta_1}\right) \\
&= \frac{d}{d\beta_1}(LRR)\left(\beta_1^{-1}\right) \\
&= -(LRR)\left(\beta_1^{-2}\right) \\
&= \frac{-LRR}{\beta_1^2}
\end{aligned}$$

Thus,

$$\mathbf{d} = \left[\begin{array}{c} \frac{-LRR}{\beta_1^2} \end{array}\right]$$

The standard error for $g(\hat{\boldsymbol{\theta}})$ will be

$$\sqrt{Var[g(\hat{\boldsymbol{\theta}})]} = \left(\mathbf{d}Cov\left[\hat{\boldsymbol{\theta}}\right]\mathbf{d}'\right)^{1/2}$$

where

$$Cov\left[\hat{\boldsymbol{\theta}}\right] = Var\left[\hat{\beta}_1\right].$$

Applying the Delta Method to the Inverse Prediction

R Function

I wrote a new function `compute_se_noint` to implement the delta method computations to compute the standard error for the weed biomass prediction for a given LRR when there is no intercept in the model. The function also returns the estimate of the LRR and a 95% confidence interval for the prediction. The inputs and outputs of the function are as follows.

Inputs:

- `lrr`: LRR for which to compute the weed biomass
- `betas`: estimated regression coefficients of β_1 (should be a vector of length 1)
- `vcov`: estimated variance of β_1 (should be a vector of length 1)

Outputs:

- data frame with the variables of
  - `lrr`: age that was specified for the computations
  - `pred_biomass`: estimated weed biomass for the specified `lrr`
  - `se`: standard error for the estimated weed biomass (computed using the delta method)
  - `ci_Lower`: lower bound of the 95% confidence interval for weed biomass
  - `ci_Upper`: upper bound of the 95% confidence interval for weed biomass

The code for the function `compute_se` is included below.

```
# Function for computing the delta-method standard error of weed biomass
compute_se_noint <- function(lrr, betas, vcov){

  # Shorten the name of betas
  b1 <- betas

  # Compute the inverse prediction of weed biomass
  pred_biomass <- lrr / b1

  # Compute d (partial derivatives of g(beta))
  d <- (-lrr) / (b1^2)

  # Compute the standard error (using the delta method)
  se <- sqrt(d %*% vcov %*% t(d))

  # Compute the lower and upper bounds of the 95% CI
  lower <- pred_biomass - (1.96 * se)
  upper <- pred_biomass + (1.96 * se)

  # Return the log response ratio, the predicted weed biomass given the
  # resopnse ration, the delta method standard error, and the lower
  # and upper bounds of the 95% CI for the predicted weed biomass
  return(data.frame(lrr = lrr,
                    pred_biomass = pred_biomass,
                    se = se,
                    ci_lower = lower,
```

```r
                    ci_upper = upper))

}
```

Standard Error Calculation for LRR = -0.69

I first extracted the estimated and variance of β_1.

```r
# Extract the estimate of beta1
betas <- as.vector(summary(m)$coefficients[,1])
betas
```

```
## [1] 0.4511447
```

```r
# Extract the variance for beta1
vcov <- as.vector(vcov(m))
vcov
```

```
## [1] 0.001376766
```

Then I applied the `compute_se_noint` function to compute the weed biomass estimate, standard error, and 95% confidence interval for a LRR of -0.69. The results are shown below.

```r
# Apply the compute_se function
res <- compute_se_noint(lrr = -0.69, betas = betas, vcov = vcov)
res
```

```
##     lrr pred_biomass        se  ci_lower  ci_upper
## 1 -0.69    -1.529443 0.1257904 -1.775992 -1.282894
```