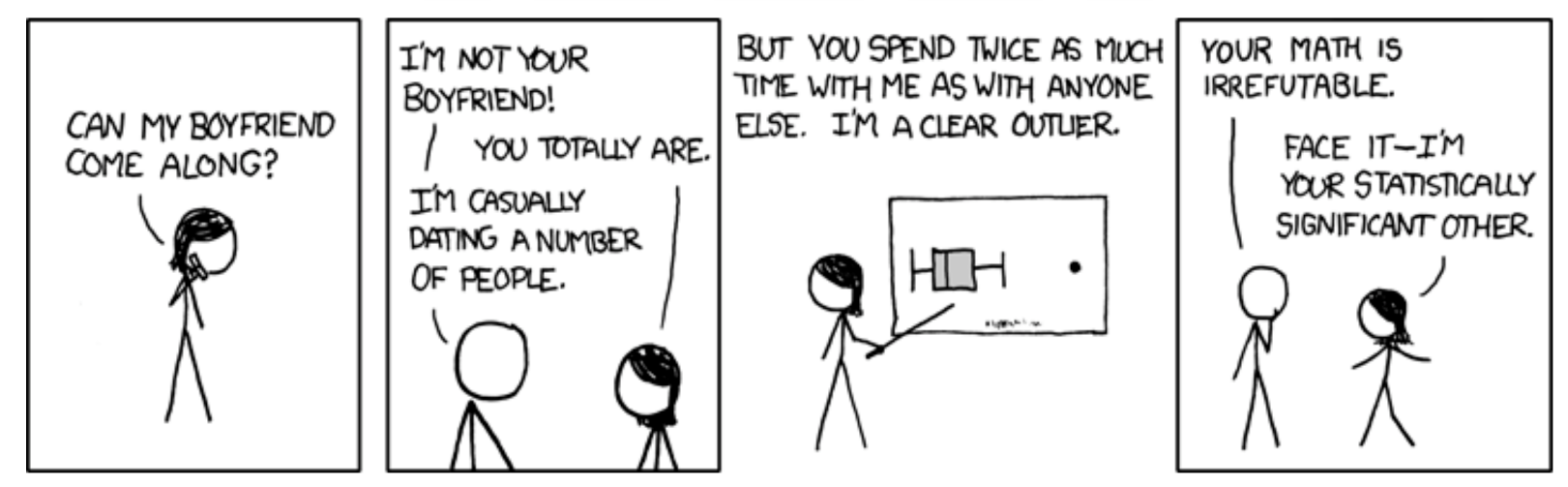
Binary Response Variables, Random vs Fixed Effects, and Outliers

Biology 683

Lecture GLM2

Heath Blackmon



Likelihood and AIC model selection

Think of likelihood of how probable the observed data under the null model. When we fit a glm we are getting the likelihood of the data assuming the best values for the parameters (beta coefficients). For any empirical dataset these will be very small numbers. It is easy for computers to store the log of very small numbers which means that log likelihoods will be negative numbers.

likelihood = .01	ln(L) = -4.6	better fit
likelihood = .0001	ln(L) = -9.2	worse fit

Adding additional predictor variables will always improve the fit of the data under the model.

R example

So we must penalize for extra parameters

R example

```
fit1 <- glm(y ~ a)
fit2 <- glm(y ~ a + b)
anova(fit1, fit2, test="LRT")
```

Likelihood and AIC model selection

$$AIC = 2k - 2\ln(L)$$

k number of parameters
ln(L) log likelihood of the data (a big negative number)

Models with smaller numbers are better

Differences in AIC can be used to evaluate evidence for one model vs. another.

A model with a ΔAIC value within 1-2 of the best model has substantial support, and should be considered along with the best model.

A ΔAIC value within only 4-7 units of the best model has considerably less support.

A ΔAIC value > 10 indicates that the worse model has virtually no support and should not be consideration.

Burnham and Anderson 1998

Likelihood and AIC model selection

LRT: Does adding this predictable variable reduce the deviation from predictions more than I would expect if the predictor had no impact on the response variable (alpha = .05)

AIC: Is about comparing models and finding the model(s) that seem to match with the data we are observing.

If you are going to use AIC to choose “best” model pay attention to close seconds and whether they “tell a different story”

Logistic Regression

Questions where the response variable is binary are common.

- Dispersal vs Non-dispersal
- Survival
- Presence/Absence of a trait
- Infected/uninfected

Logistic Regression

```r
# simulating data for amount of time practiced
coding.pract <- runif(n=100, min=10, max=70)

# creating a vector of probabilites that a student
# passes based on their amount of practice time
probs <- (coding.pract/45)
probs[probs>1] <- 1

# simulating student outcomes based on probabilites
pf <- rbinom(n=100, size=1, prob=probs)
```

Logistic Regression

```
# fitting the model to the data
fit <- glm(pf ~ coding.pract, family=binomial)

# inspecting the model fit
summary(fit)
```

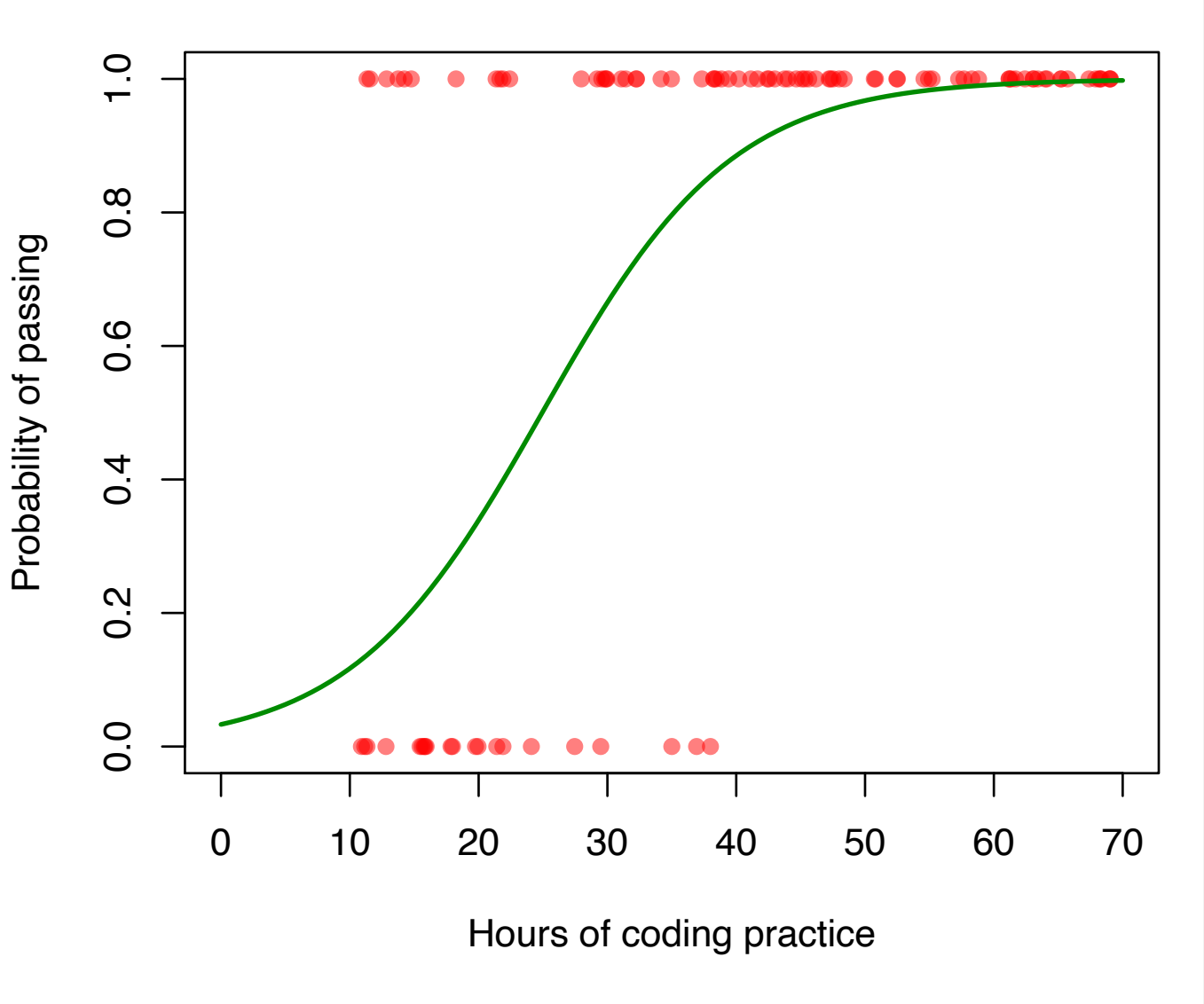
Logistic Regression

```
Call:
glm(formula = pf ~ coding.pract, family = binomial)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.14200   0.07636   0.19673   0.45411   1.63292

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.39414    0.75650  -3.165  0.00155 **
coding.pract  0.12059    0.02778   4.341 1.42e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 102.791  on 99  degrees of freedom
Residual deviance:  65.506  on 98  degrees of freedom
AIC: 69.506

Number of Fisher Scoring iterations: 6
```

Logistic Regression

```
# getting the result from fitting the model
newdat <- data.frame(coding.pract=seq(0, 70,len=100))
newdat$pf = predict(fit, newdata=newdat, type="response")

plot(pf ~ coding.pract, pch=16, col=rgb(1,0,0,.5),xlim=c(0,70),
     ylab="Probability of passing",
     xlab="Hours of coding practice")
lines(newdat$pf ~ newdat$coding.pract, col="green4", lwd=2)
```

Logistic Regression



Logistic Regression

```
# find out when a student has a 50% chance of passing
newdat$coding.pract[min(which(newdat$pf>.5))]
```

> 25.45

Logistic Regression

```
# find out when a student has a 50% chance of passing
newdat$coding.pract[min(which(newdat$pf>.5))]
```

```
> 25.45
```

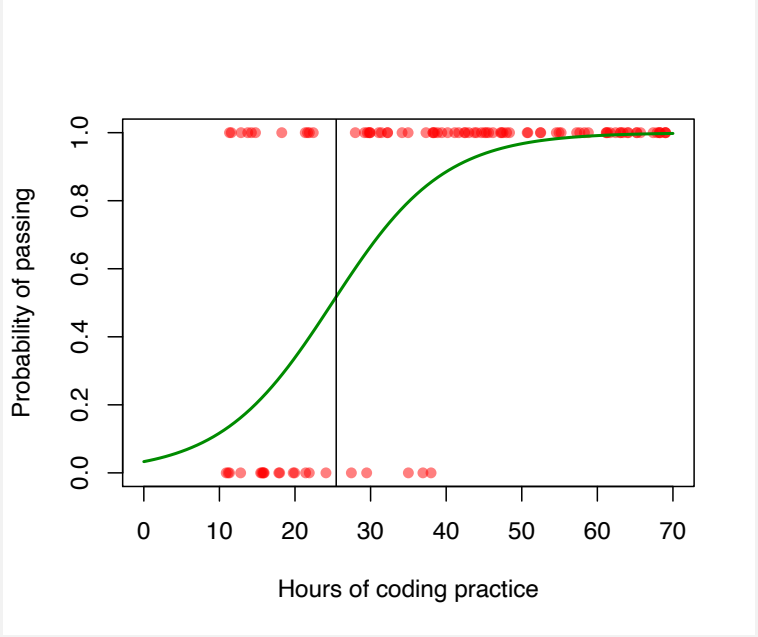
```
plot(pf ~ coding.pract, pch=16, col=rgb(1,0,0,.5),xlim=c(0,70),
     ylab="Probability of passing",
     xlab="Hours of coding practice")
lines(newdat$pf ~ newdat$coding.pract, col="green4", lwd=2)
abline(v=25.45)
```

Logistic Regression

```
# find out when a student has a 50% chance of passing
newdat$coding.pract[min(which(newdat$pf>.5))]
```

```
> 25.45
```

```
plot(pf ~ coding.pract, pch=16, col=rgb(1,0,0,.5),xlim=c(0,70),
     ylab="Probability of passing",
     xlab="Hours of coding practice")
lines(newdat$pf ~ newdat$coding.pract, col="green4", lwd=2)
abline(v=25.45)
```



Logistic Regression

```
# find out when a student has a 50% chance of passing
newdat$coding.pract[min(which(newdat$pf>.5))]
```

> 25.45

How does this compare to what we expected from the simulation?

```
# creating a vector of probabilites that a student
# passes based on their amount of practice time
probs <- (coding.pract/45)
probs[probs>1] <- 1
```

> 25/45
[1] 0.5555556

Mixed models

Mixed models are models that include fixed and and random effects.

Fixed effects can be repeated by other researchers. These are the variables that you are interested in studying.

Random effects are usually nuisance parameters. These are variables that other researchers cannot replicate and you are not interested in inferring anything about them.

Mixed models

Fixed effects are the variables whose impact we wish to determine

- Characteristics of the media or habitat
- Experimental treatments
- Age groups
- Time points
- Mutant genotypes

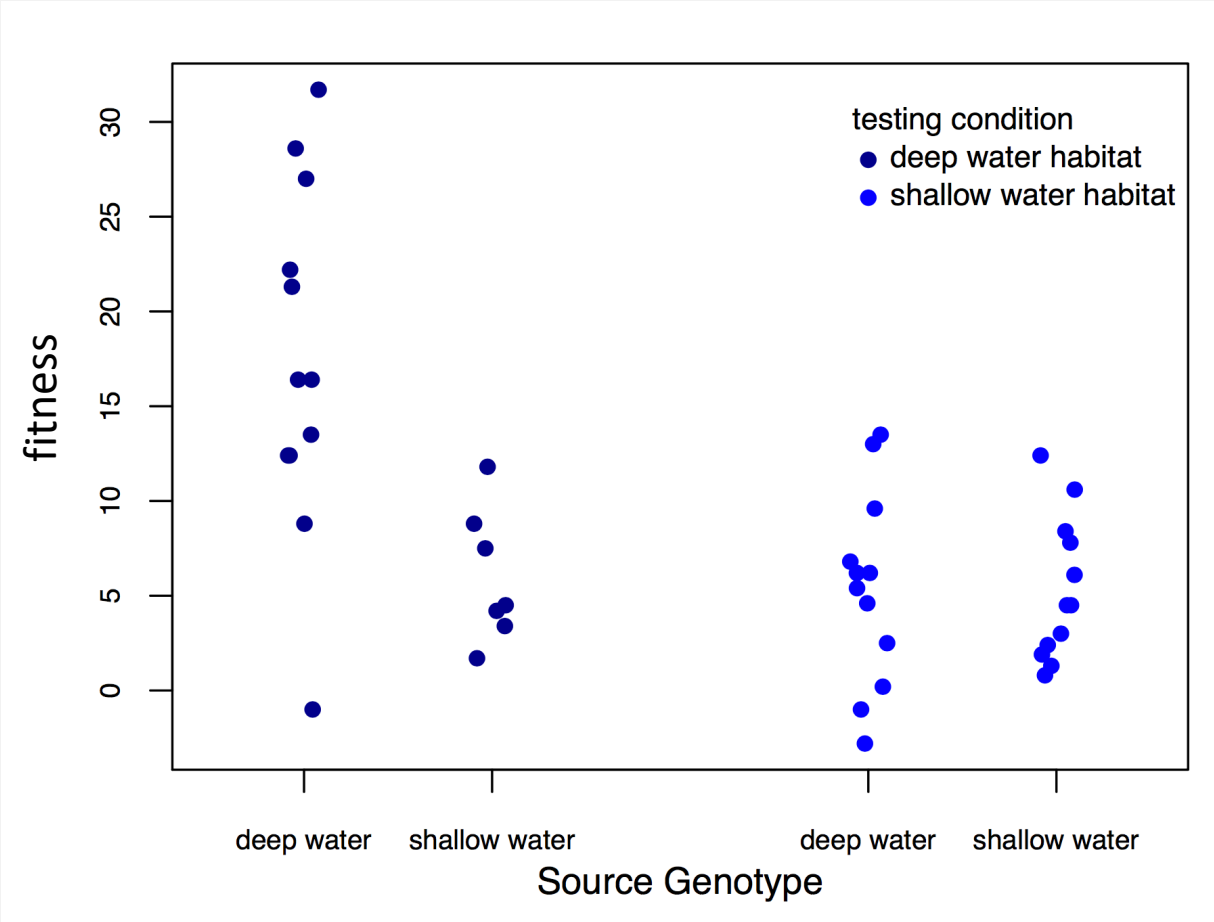
Conclusions that you reach are only applicable to the groups or treatments you include in the study

Example of fixed effects

Reciprocal relocation experiment to investigate how genotype and habitat interact to determine the fitness of stickleback fish (Rundle 2002).

		Source habitat	
		Shallow	Deep
Test habitat	Shallow	12 fish	11 fish
	Deep	7 fish	11 fish

Example of fixed effects (two factor ANOVA)

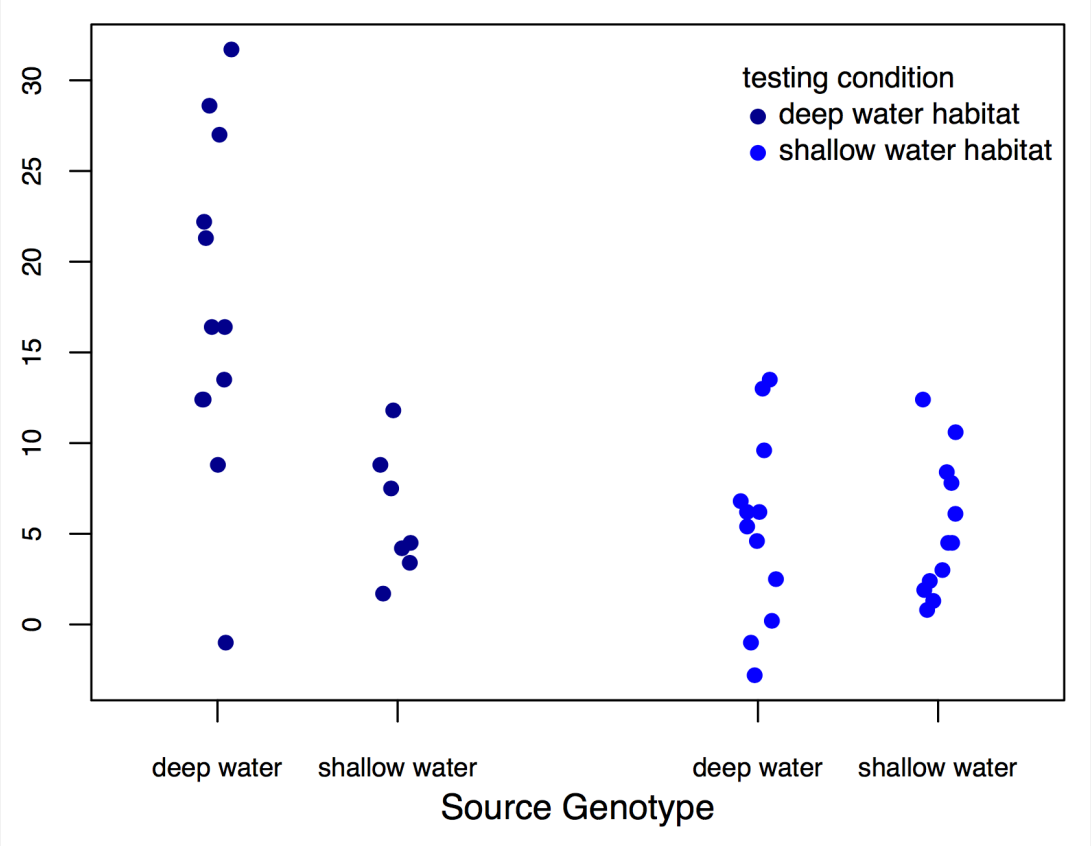



```
> anova(lm(fitness ~ genotype * test.habitat))
Analysis of Variance Table

Response: fitness
                      Df  Sum Sq Mean Sq F value    Pr(>F)
genotype               1  363.49  363.49  9.7045 0.0034403 **
test.habitat           1  550.55  550.55 14.6986 0.0004485 ***
genotype:test.habitat  1  333.58  333.58  8.9059 0.0048864 **
Residuals             39 1460.77   37.46
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> summary(lm(fitness ~ genotype * test.habitat))

Call:
lm(formula = fitness ~ genotype * test.habitat)

Residuals:
     Min       1Q   Median       3Q      Max
-18.4750  -3.6917  -0.8083   3.4583  14.2250

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                       17.475      1.767   9.891 3.49e-12 ***
genotypeshallow                  -11.489      2.911  -3.947 0.000321 ***
test.habitatshallow              -12.125      2.499  -4.853 1.99e-05 ***
genotypeshallow:test.habitatshallow  11.448      3.836   2.984 0.004886 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.12 on 39 degrees of freedom
Multiple R-squared:  0.4606,    Adjusted R-squared:  0.4192
F-statistic:  11.1 on 3 and 39 DF,  p-value: 2.093e-05
```

Interpreting Coefficients

```
> summary(lm(fitness ~ genotype * test.habitat))

Call:
lm(formula = fitness ~ genotype * test.habitat)

Residuals:
     Min       1Q   Median       3Q      Max
-18.4750  -3.6917  -0.8083   3.4583  14.2250

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                        17.475      1.767   9.891 3.49e-12 ***
genotypeshallow                   -11.489      2.911  -3.947 0.000321 ***
test.habitatshallow               -12.125      2.499  -4.853 1.99e-05 ***
genotypeshallow:test.habitatshallow  11.448      3.836   2.984 0.004886 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.12 on 39 degrees of freedom
Multiple R-squared:  0.4606,    Adjusted R-squared:  0.4192
F-statistic:  11.1 on 3 and 39 DF,  p-value: 2.093e-05
```



What is a random effect

These are randomly sampled categories of a variable that represent groups of individual measurements. Usually random effects are not repeatable.

- Study sites
- Environmental chambers
- Families made up of siblings
- Measurements within individuals

Conclusions that you reach are applicable only to the sample being studied.

What is a random effect

Sometimes random effects are a nuisance

- Field sites
- Environmental chambers
- Field plots
- Repeated measures

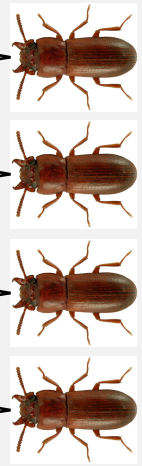
Occasionally random effects are of great interest

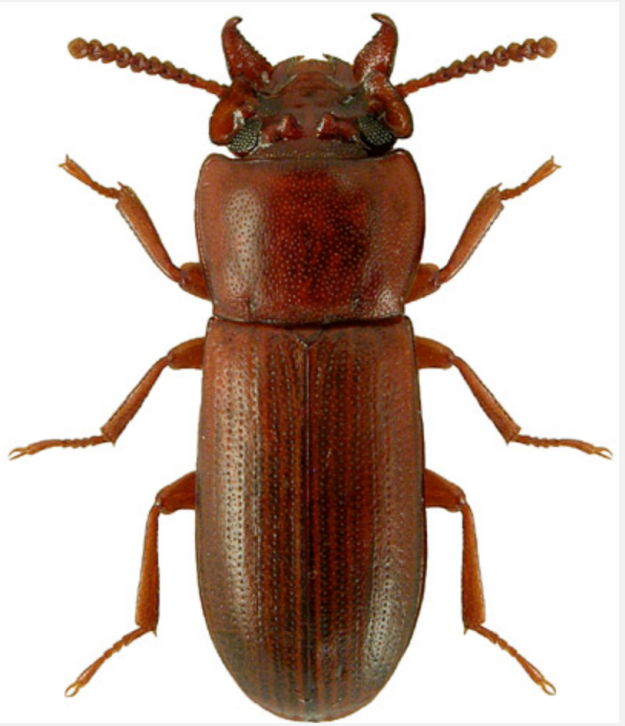
- Families - Heritability
- Individuals – Breeding value

Example of random effects

Impact of selective regime on horn size.
Measure both the left and right horn in 25 beetles from two different selective regimes.

Horn size	Beetle	Selective regime
256	1	High
276	1	High
321	2	High
321	2	High
423	3	Low
401	3	Low
381	4	Low
409	4	Low

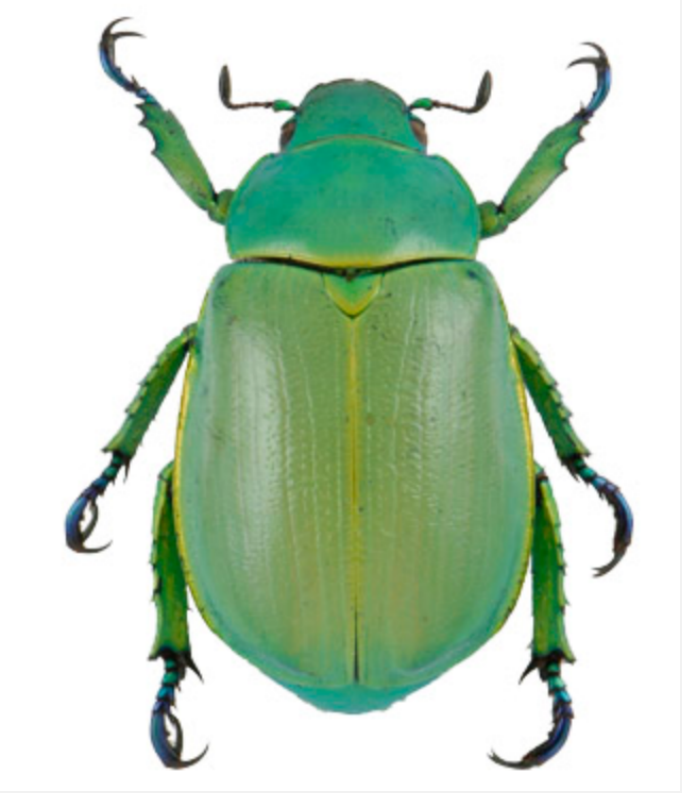




Example of random effects

Identifying the predictors for the presence or absence of Chrysina beetles.

Number collected	oak	juniper	site	date	trip
8	1	0	21	210	A
2	1	1	13	210	A
1	0	1	31	211	A
5	0	1	15	212	A
4	1	1	21	242	B
6	1	0	13	242	B
0	1	1	31	245	B
7	1	1	15	245	B



Implementing a mixed effects model

Mixed effect models can be fit using the LME function from the package nlme.

```
library(nlme)
fit <- lme(sqrt(beyeri) ~ oaks + jun + elev,
           random = list(~1|site,~1|trip),
           data=dat)
summary(fit)
```

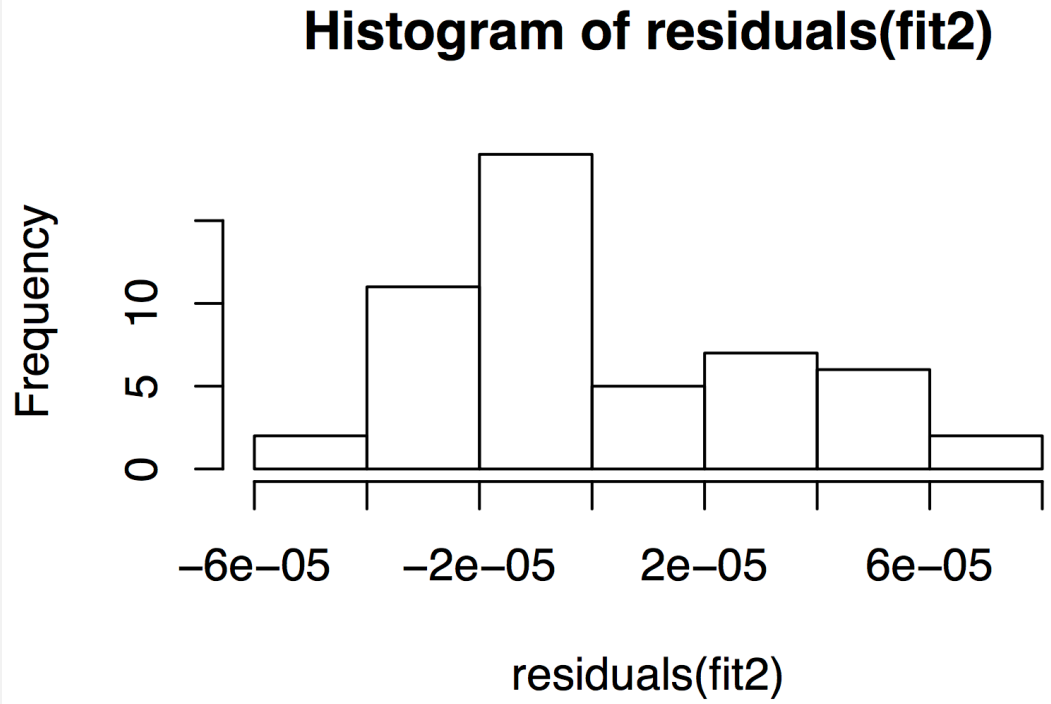
Fixed effects

Random effects

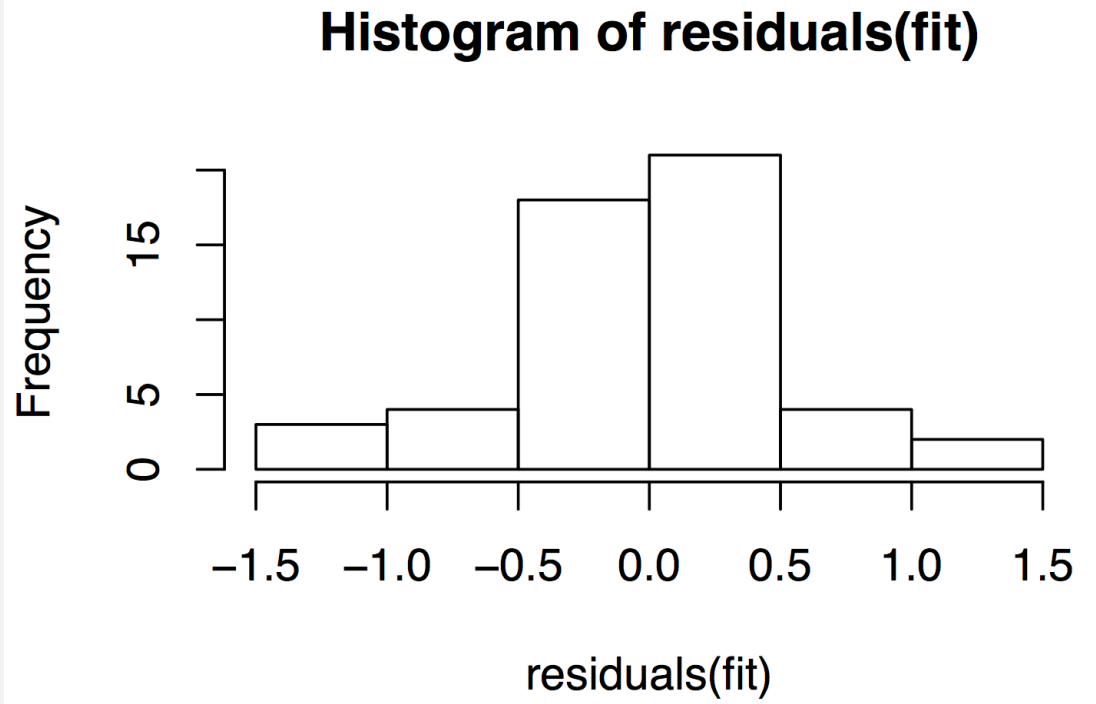
Repeated measures at sites can't be treated as independent

Implementing a mixed effects model

```
fit2 <- lme(beyeri ~ oaks + jun + elev,
            random = list(~1|site, ~1|trip),
            data=dat)
```




```
fit <- lme(sqrt(beyeri) ~ oaks + jun + elev,
           random = list(~1|site,~1|trip),
           data=dat)
```



Implementing a mixed effects model

Mixed effect models can be fit using the LME function from the package nlme.

```
fit <- lme(sqrt(beyeri) ~ oaks + jun + elev,
           random = list(~1|site,~1|trip),
           data=dat)
> summary(fit)
Linear mixed-effects model fit by REML
 Data: dat
       AIC      BIC    logLik
  153.6247 166.7231 -69.81233

Random effects:
 Formula: ~1 | site
        (Intercept)
StdDev: 2.272341e-05

 Formula: ~1 | trip %in% site
       (Intercept)   Residual
StdDev:   0.8169537 0.00264004

Fixed effects: sqrt(beyeri) ~ oaks + jun + elev
                Value Std.Error DF   t-value p-value
(Intercept) -1.6788177 1.5125428 26 -1.109931  0.2772
oaks         0.9707025 0.2943403 22  3.297892  0.0033
jun         -0.0860503 0.2460159 22 -0.349775  0.7298
elev         0.0012513 0.0008072 22  1.550167  0.1354
 Correlation:
     (Intr) oaks   jun
oaks  0.530
jun   0.197 -0.097
elev -0.993 -0.584 -0.250
```

```
fit <- lme(sqrt(beyeri) ~ oaks + elev,
           random = list(~1|site,~1|trip),
           data=dat)
> summary(fit)
Linear mixed-effects model fit by REML
 Data: dat
       AIC      BIC   logLik
  150.7722 162.1231 -69.3861

Random effects:
 Formula: ~1 | site
        (Intercept)
StdDev: 2.191491e-05

 Formula: ~1 | trip %in% site
       (Intercept)    Residual
StdDev:   0.8096043 0.002598713

Fixed effects: sqrt(beyeri) ~ oaks + elev
                Value Std.Error DF   t-value p-value
(Intercept) -1.5744919 1.4695032 26 -1.071445  0.2938
oaks         0.9607576 0.2903283 23  3.309211  0.0031
elev         0.0011807 0.0007746 23  1.524404  0.1410
 Correlation:
     (Intr) oaks
oaks  0.563
elev -0.994 -0.631
```

```
fit <- lme(sqrt(beyeri) ~ oaks,
           random = list(~1|site,~1|trip),
           data=dat)
> summary(fit)
Linear mixed-effects model fit by REML
 Data: dat
       AIC      BIC    logLik
  138.5902 148.1504 -64.29512

Random effects:
 Formula: ~1 | site
        (Intercept)
StdDev: 2.309943e-05

 Formula: ~1 | trip %in% site
       (Intercept)    Residual
StdDev:   0.8202517 0.002667513

Fixed effects: sqrt(beyeri) ~ oaks
                Value Std.Error DF  t-value p-value
(Intercept) 0.6514133 0.1674341 26 3.890566   6e-04
oaks        1.2400606 0.2281742 24 5.434710   0e+00
 Correlation:
     (Intr)
oaks -0.734
```

If you report your df with your F-statistic the reviewer will know if you did the right type of model

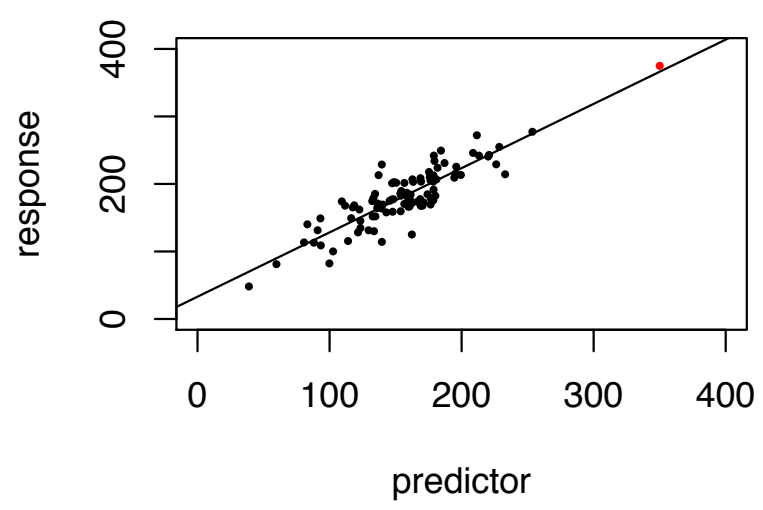
Considerations for models with random effects

- Most software will assume that all factors are fixed unless you specify them as mixed.
- Designating factors as random effects takes extra work.
- The lm function treats all predictors as fixed effects.
- Treating random effects as fixed effects is fundamentally wrong.

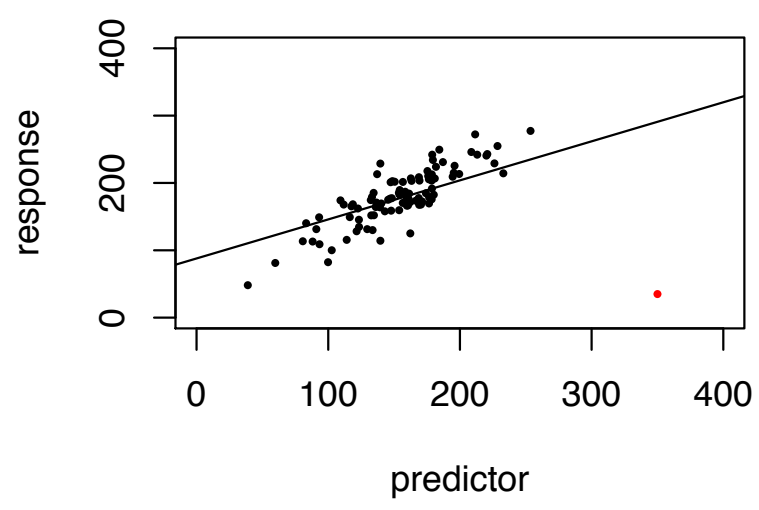
4 Types of outliers

1) Obviously erroneously recorded or measured data

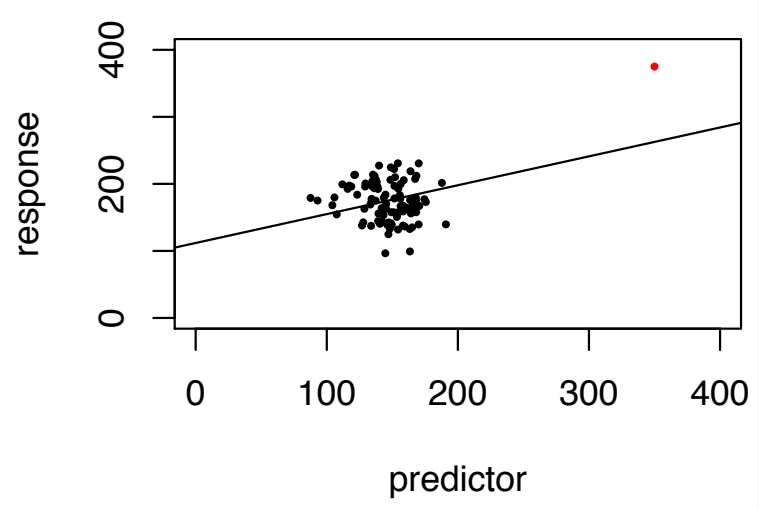
2) Extreme data point that doesn't impact statistic of interest but does impact p-value.



3) Extreme data point that impacts statistic of interest.



4) Extreme data point that creates significance.



Links

MCMCMglmm: Fit mixed models with phylogenetic or pedigree information in a Bayesian framework.

Outlier Package: Apply outlier tests to identify possible outlier datapoints - I don’t recommend this.