A short primer on (G)LMMs

Marco D. Visser Sean McMahon Lisa Huelsmann

June 29, 2019

Workshop schedule

1. Philosophy of multilevel modeling (*Sean*)
2. Computer lab: simple regression to multilevel models *(Marco)*
3. GLMM model diagnostics with DHARMa *(Lisa)*

Obligatory quote

"The computational ease with which an abundance of parameters can be estimated should not be allowed to obscure the probable unwisdom of such estimation from limited data"

- Arthur P. Dempster in "Covariance selection", Biometrics 28 (1), 157-175 (March 1972)

Workshop schedule

1. **Philosophy of multilevel modeling**(*Sean*)
2. Computer lab: simple regression to multilevel models *(Marco)*
3. GLMM model diagnostics with DHARMa *(Lisa)*

Workshop schedule

1. Philosophy of multilevel modeling(*Sean*)
2. **Computer lab: simple regression to multilevel models** *(Marco)*
3. GLMM model diagnostics with DHARMa *(Lisa)*

This workshop about

This workshop is not about:

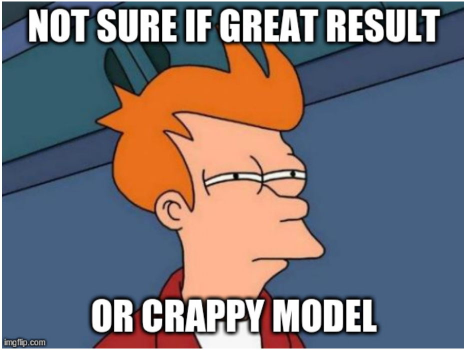
1. The math behinds mixed effect models
2. The technical details of model optimization

This workshop IS about how to fit mixed models in practice

Disclaimer: information given here is usually done in multiple semesters

This presentation, with all **knitr** code examples is available at github.com/MarcoDVisser/GLMMworkshop

General regression modeling steps

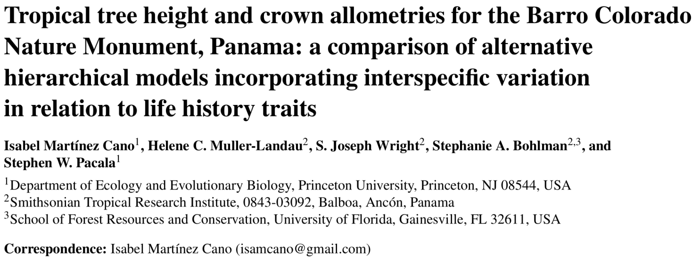


General regression modeling steps

1. Explore your data, make exploratory plots.
   - decide which model is appropriate to fit
   - distrust the model deeply
2. Evaluate the model fit
   - Check basic assumptions
   - Run model diagnostics
3. Is the model decent?
   - distrust the model slightly less
4. Does the model fail any test?
   - discard the model and start over

Tree Allometery

Exercise 1: find an unbiased function for predicting crown area from diameter



Tropical tree height and crown allometries for the Barro Colorado Nature Monument, Panama: a comparison of alternative hierarchical models incorporating interspecific variation in relation to life history traits

Isabel Martínez Cano[1], Helene C. Muller-Landau[2], S. Joseph Wright[2], Stephanie A. Bohlman[2,3], and Stephen W. Pacala[1]

[1]Department of Ecology and Evolutionary Biology, Princeton University, Princeton, NJ 08544, USA
[2]Smithsonian Tropical Research Institute, 0843-03092, Balboa, Ancón, Panama
[3]School of Forest Resources and Conservation, University of Florida, Gainesville, FL 32611, USA

Correspondence: Isabel Martínez Cano (isamcano@gmail.com)

Load the data

- Step 1: load the data

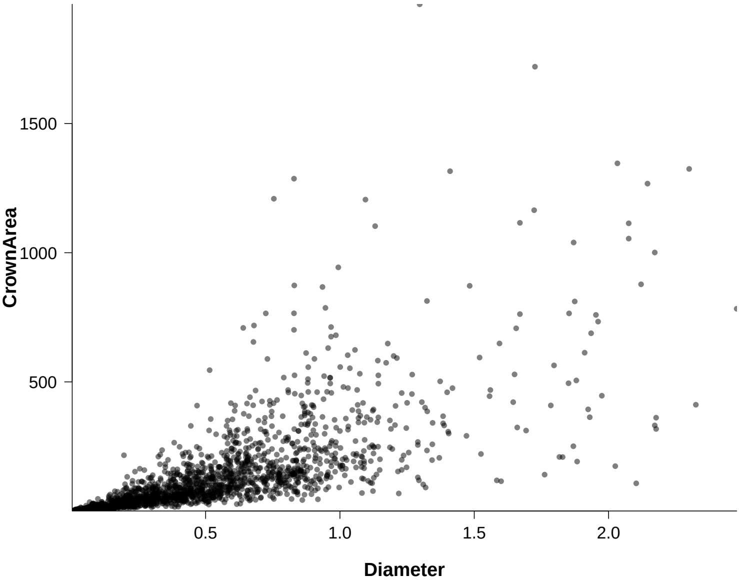
```
allo <-  read.csv("DiameterHeightCrownGLMMworkshop.csv")
traits <- read.csv("SpeciesTraits20190104.csv")
## look at the data
colnames(allo)

##  [1] "X.2"                  "X.1"                  "X"
##  [4] "SpeciesName"          "OriginalSource"       "Site"
##  [7] "Date"                 "Tag"                  "HeightOfMeasurement"
## [10] "Diameter"             "Height"               "CrownArea"
## [13] "sp.code"
```

- Step 2: fit a model
- Step 3: evaluate the model

- Plot the relationship

```r
## Explore the data
par(cex.main = 1.5, mar = c(4.4, 5, 2, 1) + 0.1,
    mgp = c(3.5, 1, 0), cex.lab = 1.5,
    font.lab = 2, cex.axis = 1.3,
    bty = "l", las = 1,
    mfrow=c(1,1),xaxs="i",yaxs="i")

## Look at relationship
plot(CrownArea~Diameter,allo,
     pch=16,col=rgb(0,0,0,alpha=0.5))
```

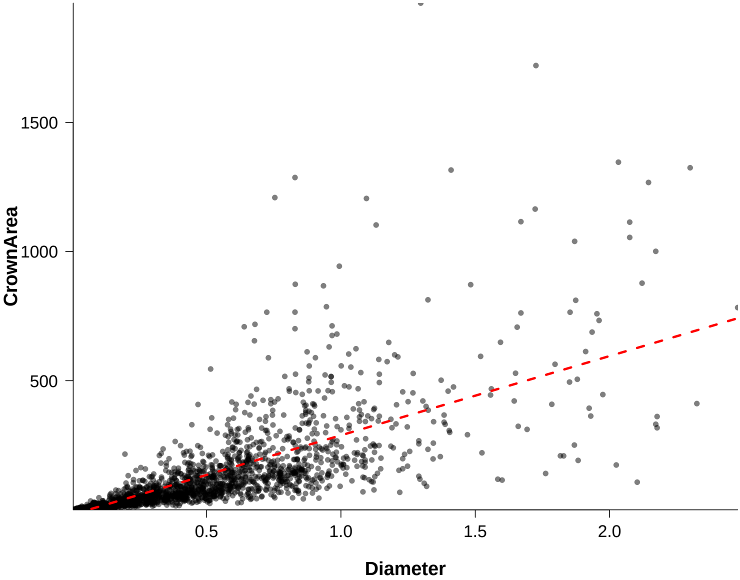
- Plot the relationship



- Fit a model

```r
## fit model
mod <- lm(CrownArea~Diameter,allo)

abline(mod,lty=2,col="red")

legend("bottomright",
       legend=bquote(R^2 == .(round(r.squaredLR(mod),2))),
      ,bty="n")
```

- Fit a model

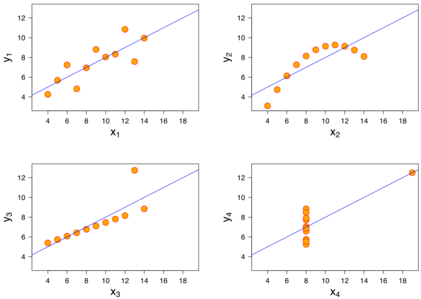


- Scrutinize the model

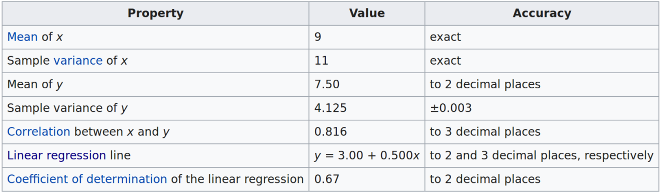
```
summary(mod)
```

```
##
## Call:
## lm(formula = CrownArea ~ Diameter, data = allo)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -519.93  -38.71    5.29   15.26 1582.22
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -18.187      3.071  -5.922 3.64e-09 ***
## Diameter     306.802      5.483  55.952  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 108.6 on 2423 degrees of freedom
## Multiple R-squared:  0.5637,Adjusted R-squared:  0.5635
## F-statistic:  3131 on 1 and 2423 DF,  p-value: < 2.2e-16
```

Anscombe's Quartet



Anscombe's Quartet



Property	Value	Accuracy
Mean of *x*	9	exact
Sample variance of *x*	11	exact
Mean of *y*	7.50	to 2 decimal places
Sample variance of *y*	4.125	±0.003
Correlation between *x* and *y*	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively
Coefficient of determination of the linear regression	0.67	to 2 decimal places

- What are regression assumptions?

- What are regression assumptions?

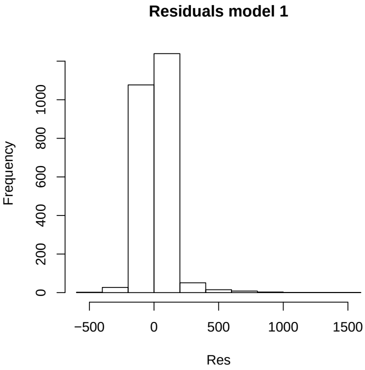
1. Linearity & unbiasedness (no correlation in ϵ)
2. Independence
3. Sample variation & little collinearity
4. Normality (of the residuals)
    - $Y \sim N(\mu = B_0 + B_1 X, \sigma = \epsilon)$
5. Homoskedasticity

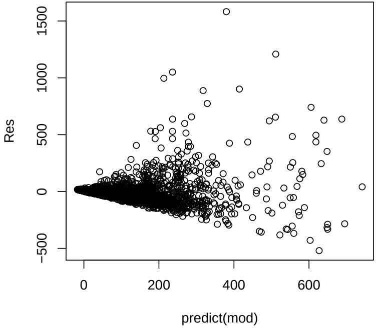
Basic model diagnostics

- Normality & lack of fit

```
## Test model (Diagnostics)
Res <- residuals(mod)
par(mfrow=c(1,2))
hist(Res,main="Residuals model 1")
plot(Res~predict(mod))
```

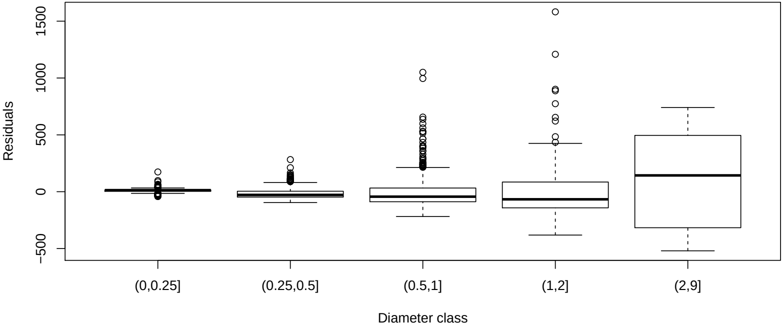
Basic model diagnostics





Basic model diagnostics

```
## Look at heterogen of variance
boxplot(Res~cut(allo$Diameter,c(0,0.25,0.50,1.00,2.00,9.00)),
        ylab="Residuals",xlab="Diameter class")
```



- $Y \sim N(\mu = B_0 + B_1 X, \sigma = B_2 \epsilon)$

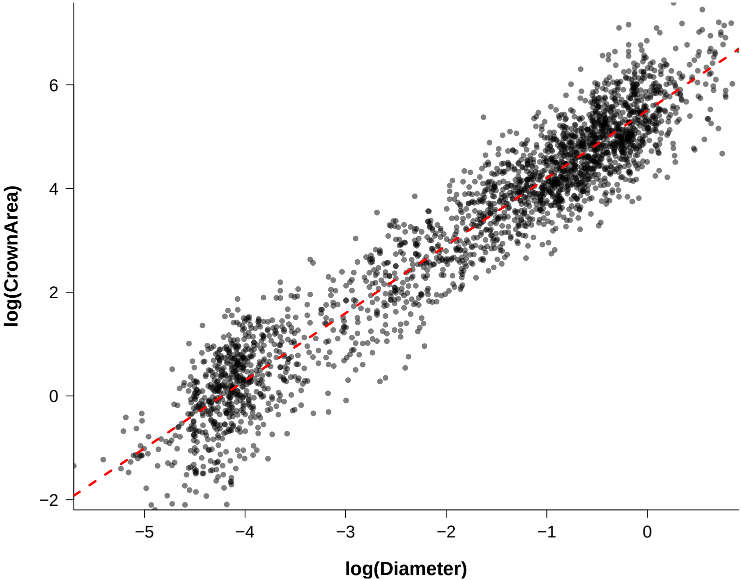
A better model

```r
par(mfrow=c(1,1))

## Look at relationship
plot(log(CrownArea)~log(Diameter),allo,
     pch=16,col=rgb(0,0,0,alpha=0.5))

## fit model
mod <- lm(log(CrownArea)~log(Diameter),allo)

abline(mod,lwd=3,lty=2,col="red")
```

Transformation are non linear models

- $log(Y) \sim N(log(B_0) + B_1 log(X), \sigma = log(B_2) + log(\epsilon))$
- $Y \sim N(B_0 X^{B_1}, \sigma = B_2 \epsilon)$

- Fit a slightly better model



A better model

```
summary(mod)
```

```
##
## Call:
## lm(formula = log(CrownArea) ~ log(Diameter), data = allo)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1498 -0.3962  0.0053  0.4332  1.9921
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.505493   0.018918   291.0   <2e-16 ***
## log(Diameter) 1.302883   0.007868   165.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6178 on 2423 degrees of freedom
## Multiple R-squared:  0.9188,Adjusted R-squared:  0.9188
## F-statistic: 2.742e+04 on 1 and 2423 DF,  p-value: < 2.2e-16
```

$$CA = \beta_0 DBH^{\beta_1} \tag{1}$$

Evaluate fit at the species level

```r
## Lets look at the fit per species
sp <- unique(allo$sp.code)

pdf("speciesLevelDiagnostics.pdf")
par(mfrow=c(2,2))

## get predictions and observations
Pred <- predict(mod)
Obser <- allo$CrownArea

for(i in sp){
    inc <- allo$sp.code%in%i
    plot(Pred[inc]~Obser[inc],
         ylab="Predicted",xlab="Observed",
         main=i)
    abline(0,1,lwd=2,lty=2)
}

dev.off()
```

Look at the PDF

Exercise 2: Improve species level fits

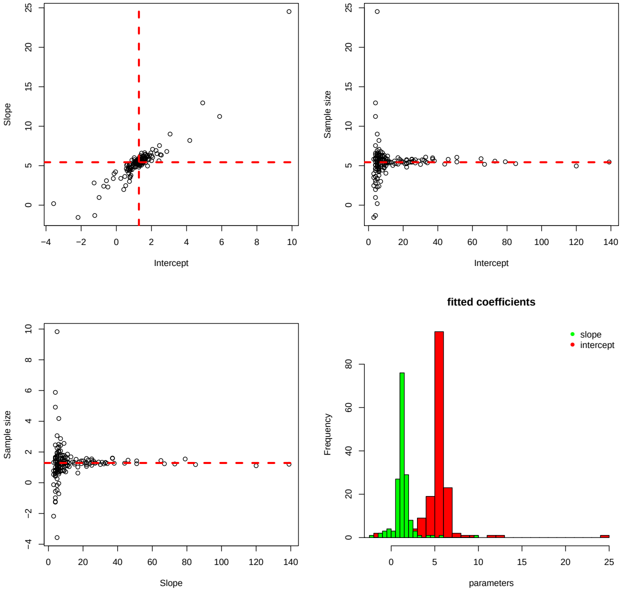
Start from the code below

```r
## Lets look at the fit per species
sp <- unique(allo$sp.code)

pdf("speciesLevelDiagnostics.pdf")
par(mfrow=c(2,2))

## get predictions and observations
Pred <- predict(mod)
Obser <- allo$CrownArea

for(i in sp){
    inc <- allo$sp.code%in%i
    plot(Pred[inc]~Obser[inc],
         ylab="Predicted",xlab="Observed",
         main=i)
    abline(0,1,lwd=2,lty=2)
}

dev.off()
```

```
## Fit for each species
sp <- unique(allo$sp.code)

pdf("speciesLevelFits.pdf")
par(mfrow=c(2,2))

## get predictions and observations
Dbh <- allo$Diameter
Obser <- allo$CrownArea
SpFits <- array(dim=c(length(sp),3))
N <- as.numeric(table(allo$sp.code))
colnames(SpFits) <- c("Intercept","Slope","N")

for(i in sp){
    inc <- allo$sp.code%in%i
    plot(log(Obser[inc])~log(Dbh[inc]),
         ylab="Predicted",xlab="Observed",
         main=i)
    mod <- lm(log(Obser[inc])~log(Dbh[inc]))
    abline(mod,lwd=2,lty=2)
    SpFits[which(sp==i),] <- c(coef(mod),N[which(sp==i)])
}

dev.off()


## pdf
##   2
```

Look at the PDF

```
## look at estimated coefficients compared to sample size
par(mfrow=c(2,2))

MeanStats <- colMeans(SpFits) # mean values

## slope and intercept
plot(SpFits[,1]~SpFits[,2],xlab="Intercept",ylab="Slope")
abline(h=MeanStats[1],v=MeanStats[2],col="red",lty=2,lwd=3)
## intercept and sample size
plot(SpFits[,1]~SpFits[,3],xlab="Intercept",ylab="Sample size")
abline(h=MeanStats[1],col="red",lty=2,lwd=3)

## intercept and sample size
plot(SpFits[,2]~SpFits[,3],xlab="Slope",ylab="Sample size")
abline(h=MeanStats[2],col="red",lty=2,lwd=3)

## distributions
hist(SpFits[,1],freq=TRUE,col="red",breaks=20,xlab="parameters",
     main="fitted coefficients")
hist(SpFits[,2],freq=TRUE,col="green",breaks=20,add=TRUE)
legend("topright",legend=c("slope","intercept"),
       bty="n",pch=16,col=c("green","red"))
```

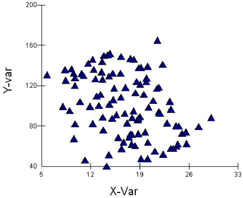

Slope
Intercept
Sample size
Intercept
Sample size
Slope
fitted coefficients
Frequency
parameters
● slope
● intercept
−4
−2
0
2
4
6
8
10
0
5
10
15
20
25
0
20
40
60
80
100
120
140

Multi-level modeling

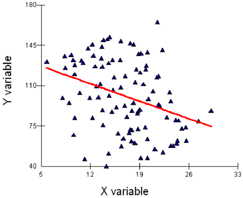
- $Y_{ij} \sim N(\mu = B_{0j} + B_1 X_{ij}, \sigma = \sigma_\epsilon)$
- $B_{0j} \sim N(\mu = \gamma, \sigma = \sigma_\gamma)$

In a hierarchical world some model can be misleading!

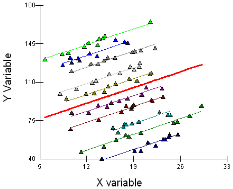
What is the relation between Y and X?



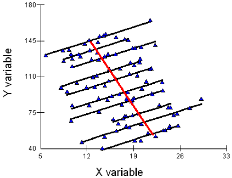
Negative: According to simple regression



Positive: according to a multilevel model



Positive and negative: within and between relations



```
panel2<-lm(Y~X)
panel3<-lmer(Y~X + (1| group))
panel4<-lmer(Y~X + (1+X|group))
```

How to find out which model to use?

- What are the regression assumptions of multi-level modeling?
  - $Y_{ij} \sim N(\mu = B_{0j} + B_1 X_{ij}, \sigma = \sigma_\epsilon)$
  - $B_{0j} \sim N(\mu = \gamma, \sigma = \sigma_\gamma)$

  1. Linearity & unbiasedness (no correlation in ϵ in i or j)
  2. Independence ("at the test level")
  3. Sample variation & little collinearity
  4. Normality (of the residuals)
     - $Y \sim N(\mu = B_0 + B_1 X, \sigma = \epsilon)$
  5. Homoskedasticity

Tree Allometery

Exercise 3: find the best multilevel model to predict crown area from diameter

Basic model diagnostics

```
## fit multi-level model
require(lme4)


## Loading required package:  lme4
## Loading required package:  Matrix

LMmod <- lm(log(CrownArea)~log(Diameter),allo)
LMMmod <- lmer(log(CrownArea)~log(Diameter)+(1+log(Diameter)|sp.code),allo)
```

```
## Summarize model

summary(LMMmod)


## Linear mixed model fit by REML ['lmerMod']
## Formula: log(CrownArea) ~ log(Diameter) + (1 + log(Diameter) | sp.code)
##    Data: allo
##
## REML criterion at convergence: 4062.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2354 -0.6382 -0.0163  0.6593  3.4589
##
## Random effects:
##  Groups   Name          Variance Std.Dev. Corr
##  sp.code  (Intercept)   0.06302  0.2510
##           log(Diameter) 0.01715  0.1310   0.36
##  Residual               0.27748  0.5268
## Number of obs: 2425, groups:  sp.code, 162
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    5.54102    0.03275  169.19
## log(Diameter)  1.31477    0.01847   71.17
##
## Correlation of Fixed Effects:
##             (Intr)
## log(Diamtr) 0.648
```

Multi-level modeling

- $Y_{ij} \sim N(\mu = B_{0j} + B_1 X_{ij}, \sigma = \sigma_\epsilon)$
- $B_{0j} \sim N(\mu = \gamma, \sigma = \sigma_\gamma)$
- $B_{1j} \sim N(\mu = \alpha, \sigma = \sigma_\alpha)$
- $B_j \sim N(\mu = [\gamma, \alpha], \sigma = \Sigma)$

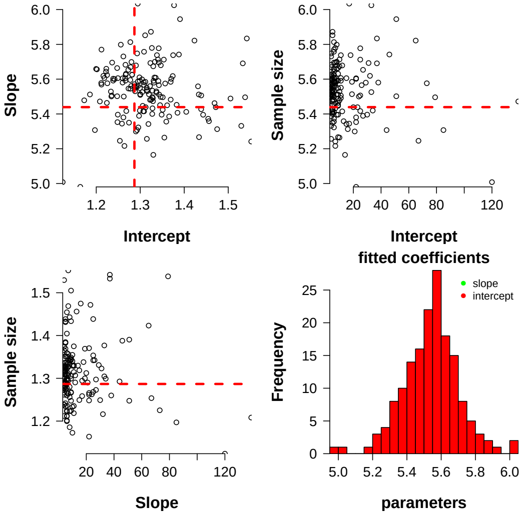
Explore the model fit

```r
## extract coefficients
LMMcoef <- as.data.frame(coef(LMMmod)$sp.code)
LMMcoef$n <- N
```

Explore the model fit

```
## Look at coefficients
par(mfrow=c(2,2))
MeanStats <- colMeans(LMMcoef)
print(MeanStats)
print(fixef(LMMmod))
plot(LMMcoef[,1]~LMMcoef[,2],xlab="Intercept",ylab="Slope")
abline(h=MeanStats[1],v=MeanStats[2],col="red",lty=2,lwd=3)
plot(LMMcoef[,1]~LMMcoef[,3],xlab="Intercept",ylab="Sample size")
abline(h=MeanStats[1],col="red",lty=2,lwd=3)
plot(LMMcoef[,2]~LMMcoef[,3],xlab="Slope",ylab="Sample size")
abline(h=MeanStats[2],col="red",lty=2,lwd=3)
hist(LMMcoef[,1],freq=TRUE,col="red",breaks=20,xlab="parameters",
     main="fitted coefficients")
hist(LMMcoef[,2],freq=TRUE,col="green",breaks=20,add=TRUE)
legend("topright",legend=c("slope","intercept"),
       bty="n",pch=16,col=c("green","red"))
```

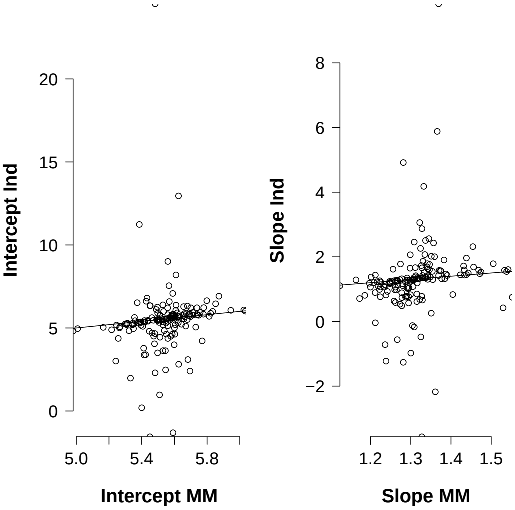
Explore the model fit



Explore the model fit

```
## Compared earlier model fits mixed model coefficients
par(mfrow=c(1,2))
plot(SpFits[,1]~LMMcoef[,1],xlab="Intercept MM",ylab="Intercept Ind")
abline(0,1)
plot(SpFits[,2]~LMMcoef[,2],xlab="Slope MM",ylab="Slope Ind")
abline(0,1)
```

Explore the model fit



Explore the model fit

```
colMeans(LMMcoef)
```

```
##   (Intercept) log(Diameter)             n
##      5.541024      1.314767     14.969136
```

```
colMeans(SpFits)
```

```
## Intercept     Slope         N
##  5.439121  1.286750 14.969136
```

Explore the model fit

```
(MMsd<-apply(LMMcoef,2,sd))
```

```
##   (Intercept) log(Diameter)             n
##    0.16311335    0.07675801   19.67765793
```

```
(LMsd<-apply(SpFits,2,sd))
```

```
## Intercept     Slope         N
##  2.153315  1.160481 19.677658
```

```
LMsd/MMsd
```

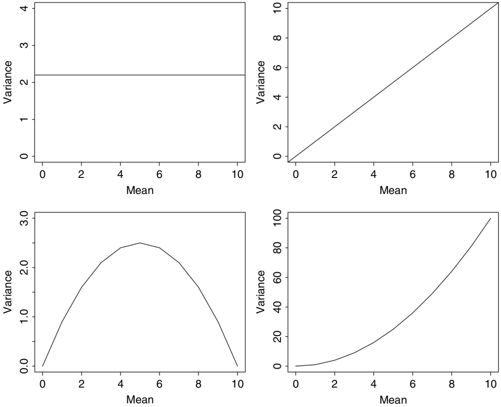
```
## Intercept     Slope         N
##  13.20134  15.11870   1.00000
```

LMM towards GLMM

- What are the regression assumptions of multi-level modeling?
  - $Y_{ij} \sim N(\mu = B_{0j} + B_1 X_{ij}, \sigma = \sigma_\epsilon)$
  - $B_{0j} \sim N(\mu = \gamma, \sigma = \sigma_\gamma)$

1. Linearity & unbiasedness (no correlation in ϵ in i or j)
2. Independence ("at the test level")
3. Sample variation & little collinearity
4. ~~Normality (of the residuals)~~
   - $Y \sim N(\mu = B_0 + B_1 X, \sigma = \epsilon)$
5. ~~Homoskedasticity~~

GLMs


Variance
0
1
2
3
4
0
2
4
6
8
10
Mean
Variance
0
2
4
6
8
10
0
2
4
6
8
10
Mean
Variance
0.0
1.0
2.0
3.0
0
2
4
6
8
10
Mean
Variance
0
20
40
60
80
100
0
2
4
6
8
10
Mean

Workshop schedule

1. Philosophy of multilevel modeling (*Sean*)
2. Computer lab: simple regression to multilevel models (*Marco*)
3. **GLMM model diagnostics with DHARMa** *(Lisa)*