

# Linear Mixed-Effects Models

Quantitative Analysis of Vertebrate Populations

# Advantages of Mixed Models

- ▶ Increased scope of inference
- ▶ More honest accounting of uncertainty
- ▶ Efficiency of estimation

# Random Effects

- ▶ Random intercept
- ▶ Random slope(s) *not generally realistic - don't use*
- ▶ Random intercept and slope(s) independent
- ▶ Random intercept and slope(s) correlated

# Random intercept model

$$y_i = \alpha_{j(i)} + \beta X_i + \epsilon_i$$

where

$$\alpha_j \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$$

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

- ▶ snake  $i$  at site  $j$
- ▶ Mean size of the snake varies normally among populations

## Random intercept and slope model

$$y_i = \alpha_{j(i)} + \beta X_{j(i)} + \epsilon_i$$

where

$$\alpha_j \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$$

$$\beta_j \sim \mathcal{N}(\mu_\beta, \sigma_\beta^2)$$

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

## Simulate Data

```
n.groups <- 56          # Number of populations
n.sample <- 10          # Number of vipers in each pop
n <- n.groups * n.sample # Total number of data points
pop <- gl(n = n.groups, k = n.sample) # Indicator for population
```

## Simulate Data

```
# Body length (cm)  
original.length <- runif(n, 45, 70)  
mn <- mean(original.length)  
sd <- sd(original.length)  
length <- (original.length - mn) / sd  
hist(length, col = "grey")
```



## Set up design matrix

```
Xmat <- model.matrix(~pop*length-1-length)
print(Xmat[1:21,], dig = 2)           # Print top 21 rows
```

[illegible]



## Choose parameter values

```
intercept.mean <- 230           # mu_alpha  
intercept.sd   <- 20            # sigma_alpha  
slope.mean    <- 60             # mu_beta  
slope.sd      <- 30             # sigma_beta
```

## Simulate effects

```
intercept.effects<-rnorm(n = n.groups,  
                        mean = intercept.mean,  
                        sd = intercept.sd)  
slope.effects <- rnorm(n = n.groups, mean = slope.mean, sd  
all.effects <- c(intercept.effects, slope.effects) # Put the
```

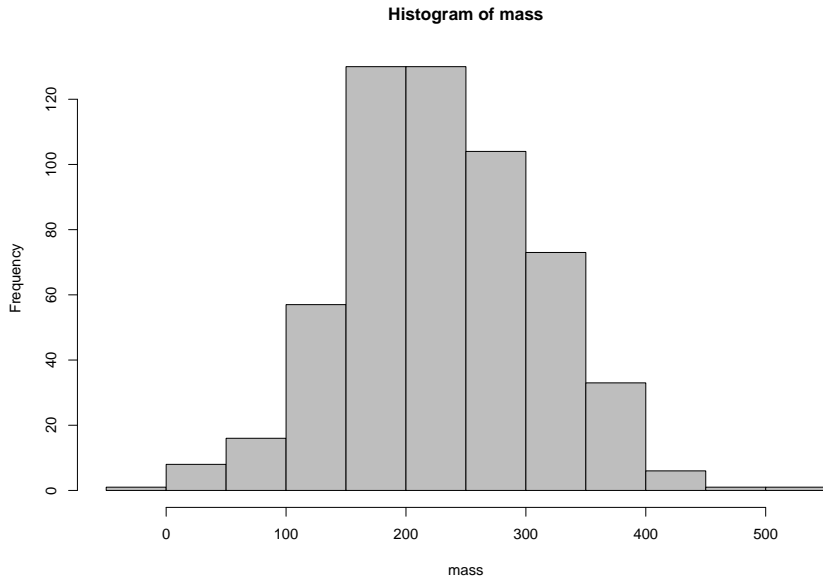
## Make linear model

```
lin.pred <- Xmat[,] %*% all.effects # Value of lin.predictions  
eps <- rnorm(n = n, mean = 0, sd = 30) # residuals  
mass <- lin.pred + eps # response = lin.pred + residuals
```

## Examine simulated data

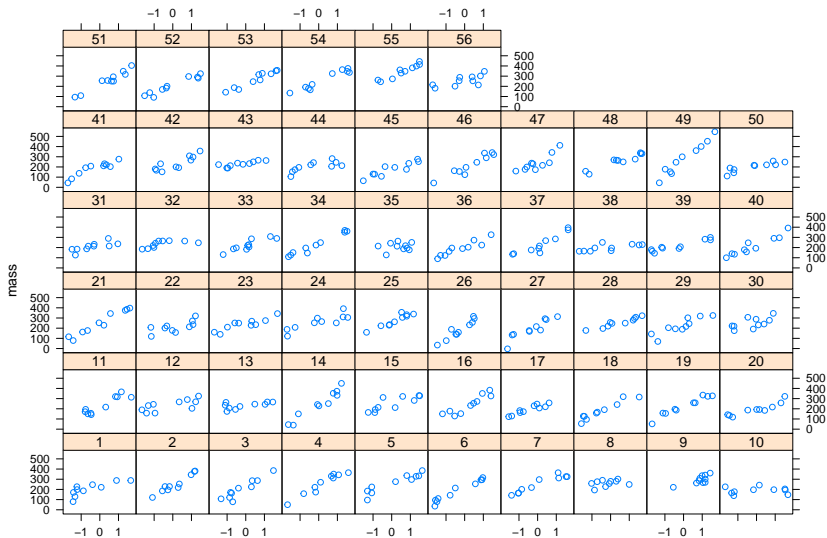
```
hist(mass, col = "grey")
```

*# Inspect what we've created*



# Examine simulated data

```
library(lattice)  
xyplot(mass ~ length | pop)
```



## Analysis under a random-intercepts model

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
lme.fit1 <- lmer(mass ~ length + (1 | pop), REML = TRUE)  
lme.fit1
```

```
## Linear mixed model fit by REML ['lmerMod']
```

```
## Formula: mass ~ length + (1 | pop)
```

```
## REML criterion at convergence: 5843.213
```

```
## Random effects:
```

```
##   Groups      Name          Std.Dev.
```

```
##   pop        (Intercept)  21.06
```

```
##   Residual                42.27
```

```
## Number of obs: 560, groups:  pop, 56
```

```
## Fixed Effects:
```

```
## (Intercept)          length
```

```
##      230.57           64.81
```

## Analysis under a random-coefficients model without correlation between intercept and slope

```
lme.fit2 <- lmer(mass ~ length + (1 | pop) + (0 + length |  
lme.fit2
```

```
## Linear mixed model fit by REML ['lmerMod']  
## Formula: mass ~ length + (1 | pop) + (0 + length | pop)  
## REML criterion at convergence: 5643.558  
## Random effects:  
##   Groups      Name          Std.Dev.  
##   pop        (Intercept)  20.15  
##   pop.1      length       29.19  
##   Residual                        31.16  
## Number of obs: 560, groups:  pop, 56  
## Fixed Effects:  
## (Intercept)          length  
##      228.96          66.02
```

## Random-coefficients model with correlation between intercept and slope

$$y_i = \alpha_{j(i)} + \beta X_{j(i)} + \epsilon_i$$

where

$$(\alpha_j, \beta_j) \sim MVN(\mu, \Sigma)$$

$$\mu = (\mu_\alpha, \mu_\beta)$$

$$\begin{pmatrix} \sigma_\alpha^2 & \sigma_{\alpha\beta} \\ \sigma_{\alpha\beta} & \sigma_\beta^2 \end{pmatrix}$$

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2)$$



## Data generation

```
n.groups <- 56  
n.sample <- 10  
n <- n.groups * n.sample  
pop <- gl(n = n.groups, k = n.sample)
```

## Data generation

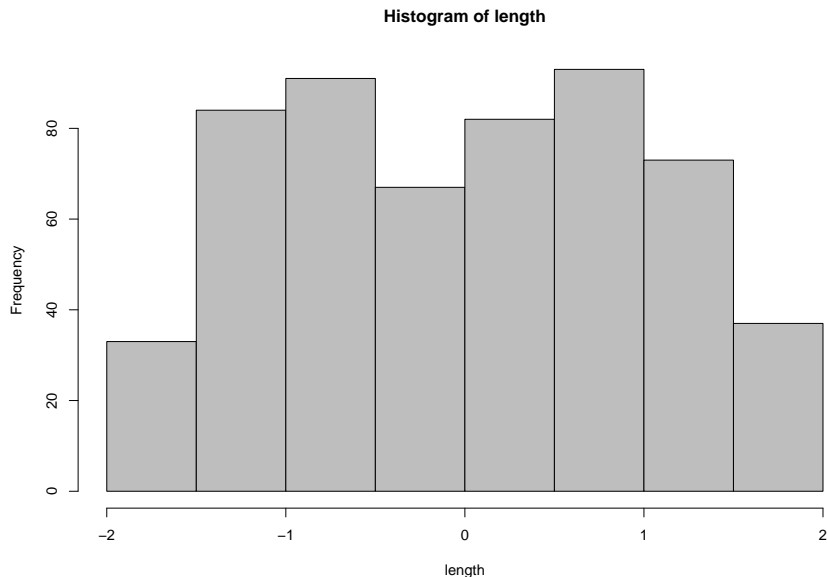
```
original.length <- runif(n, 45, 70)      # Body length (cm)
mn <- mean(original.length)
sd <- sd(original.length)
cat("Mean and sd used to normalise.original length:", mn, s
```

```
## Mean and sd used to normalise.original length: 57.26806
```

```
length <- (original.length - mn) / sd
```

## Data generation - View distribution

```
hist(length, col = "grey")
```



## Make model matrix

```
library(MASS)                # Load MASS
# ?mvnrm                      # Check syntax
Xmat <- model.matrix(~pop*length-1-length)
# print(Xmat[1:21,], dig = 2) # Print top 21 rows
```

## Choose values

```
intercept.mean <- 230           # Values for five hyperparameters  
intercept.sd <- 20  
slope.mean <- 60  
slope.sd <- 30  
intercept.slope.covariance <- 10
```

## Make correlated parameters

```
mu.vector <- c(intercept.mean, slope.mean)
var.cova.matrix <- matrix(c(intercept.sd^2, intercept.slope.
intercept.slope.covariance, slope.sd^2), 2, 2)

effects <- mvrnorm(n = n.groups, mu = mu.vector, Sigma = var.cova.matrix)
effects                                     # Look at what we've created
```

##		[,1]	[,2]
##	[1,]	238.3833	41.245677
##	[2,]	222.5691	29.248310
##	[3,]	254.8710	38.747259
##	[4,]	258.7976	90.089531
##	[5,]	202.9449	31.666917
##	[6,]	239.6325	23.526240
##	[7,]	235.8725	52.138177
##	[8,]	219.4745	104.352918
##	[9,]	241.9256	107.358104
##	[10,]	195.4122	25.648085

## Make correlated parameters

```
intercept.effects <- effects[,1]  
slope.effects <- effects[,2]  
all.effects <- c(intercept.effects, slope.effects) # Put the
```

## Make linear model

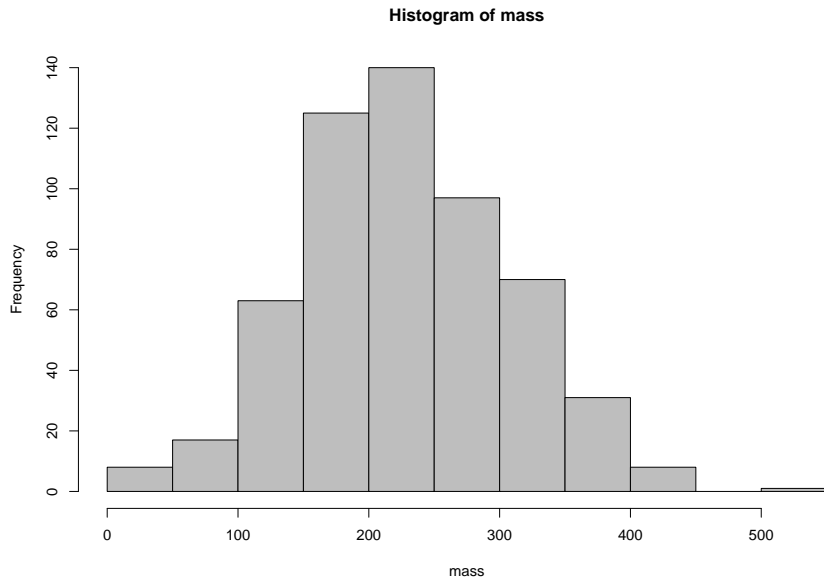
```
lin.pred <- Xmat[,] %*% all.effects # Value of lin.predictions  
eps <- rnorm(n = n, mean = 0, sd = 30) # residuals  
mass <- lin.pred + eps # response = lin.pred + residuals
```



## View histogram

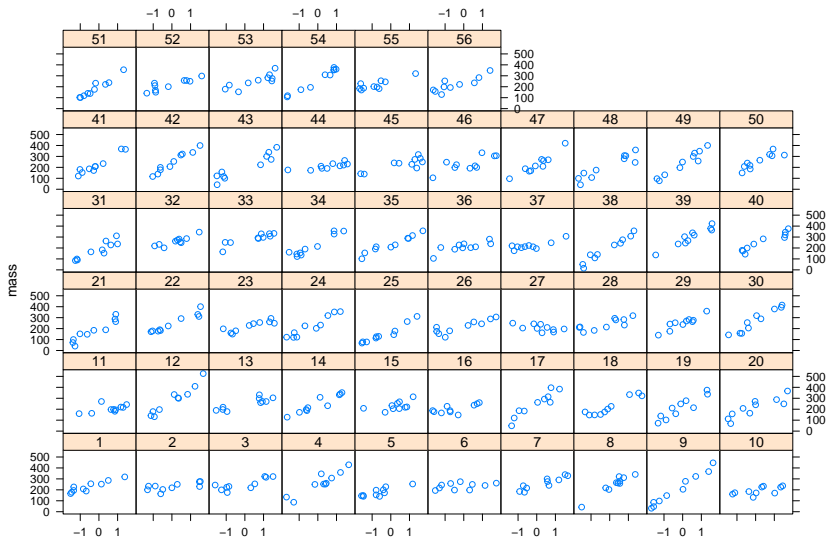
```
hist(mass, col = "grey")
```

*# Inspect what we've created*



# View lattice plot

```
library("lattice")  
xyplot(mass ~ length | pop)
```



## REML analysis using R

```
lme.fit3 <- lmer(mass ~ length + (length | pop))  
lme.fit3
```

```
## Linear mixed model fit by REML ['lmerMod']  
## Formula: mass ~ length + (length | pop)  
## REML criterion at convergence: 5604.301  
## Random effects:  
##   Groups      Name          Std.Dev.  Corr  
##   pop          (Intercept)  20.45  
##           length          30.56    0.29  
## Residual                29.77  
## Number of obs: 560, groups:  pop, 56  
## Fixed Effects:  
## (Intercept)          length  
##      229.11          64.74
```

# References

Kery, M. 2010. Introduction to WinBUGS for Ecologists. Academic Press.

Zuur et al. 2009. Mixed Effects Models and Extensions in Ecology with R. Springer.