

# 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

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

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

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

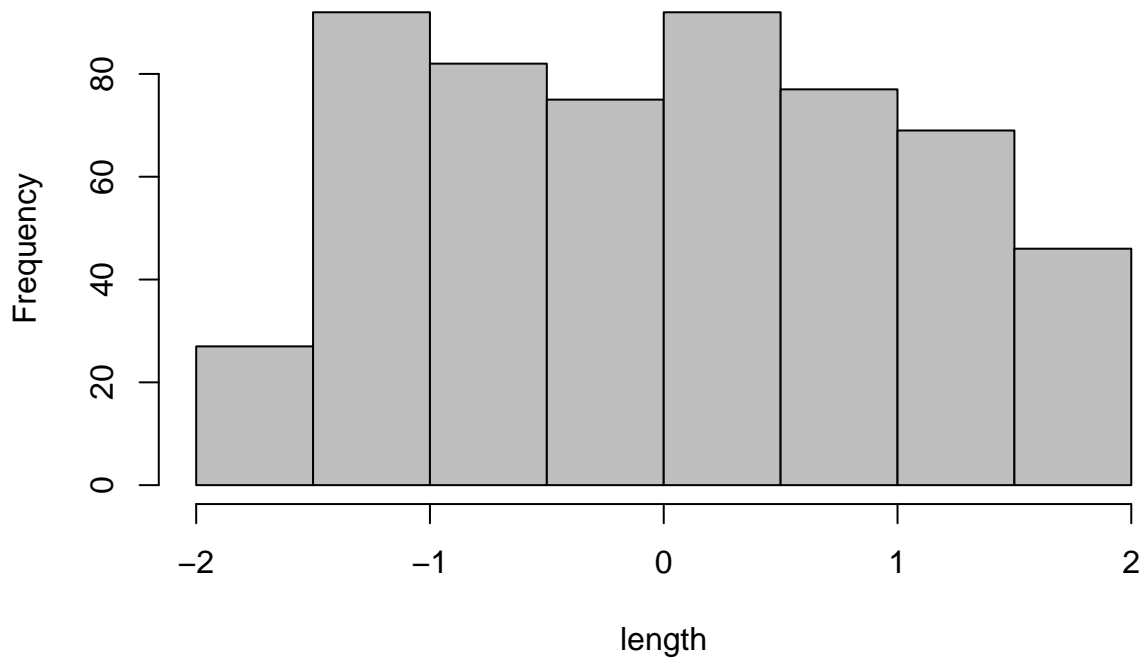
### Simulate Data

```
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")
```

Histogram of length



## Set up design matrix

```
Xmat <- model.matrix(~pop*length-1-length)
print(Xmat[1:5, 1:5], dig = 2) # Print 5 rows and columns
```

```
##   pop1 pop2 pop3 pop4 pop5
## 1    1    0    0    0    0
## 2    1    0    0    0    0
## 3    1    0    0    0    0
## 4    1    0    0    0    0
## 5    1    0    0    0    0
```

## 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

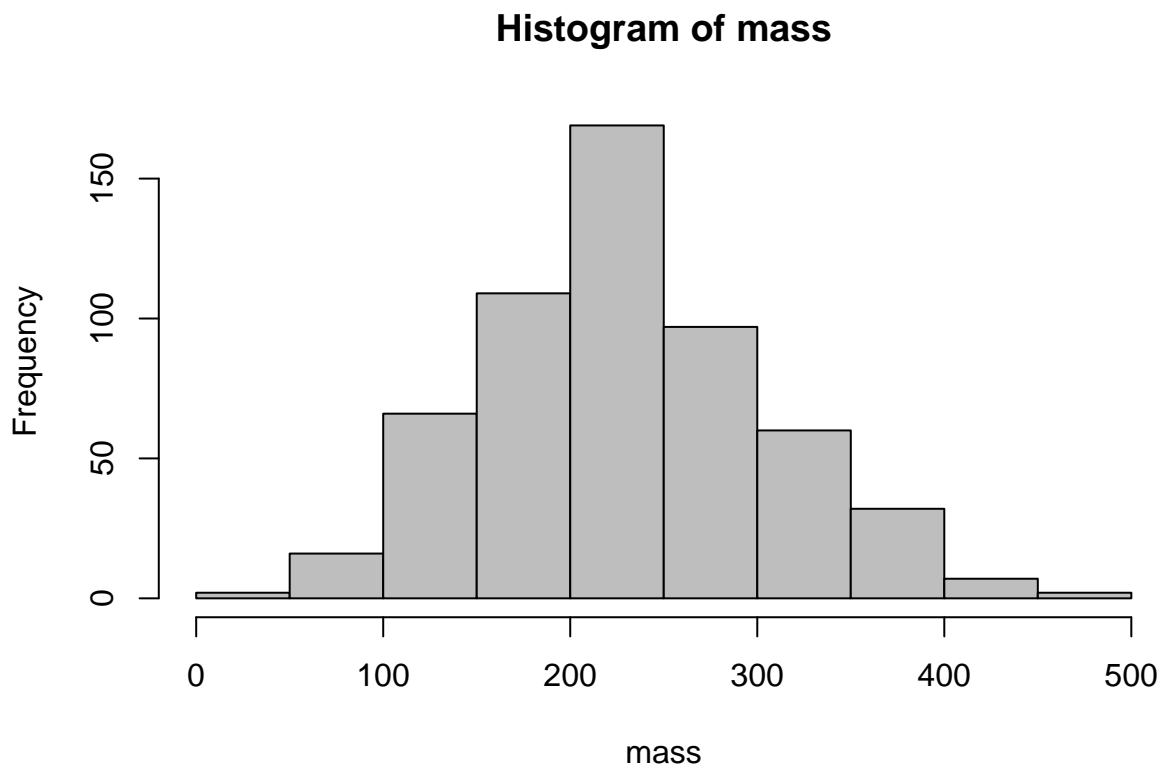
```
intercept.effects<-rnorm(n = n.groups,  
                        mean = intercept.mean,  
                        sd = intercept.sd)  
slope.effects <- rnorm(n = n.groups, mean = slope.mean, sd = slope.sd)  
all.effects <- c(intercept.effects, slope.effects) # Put them all together
```

## Make linear model

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

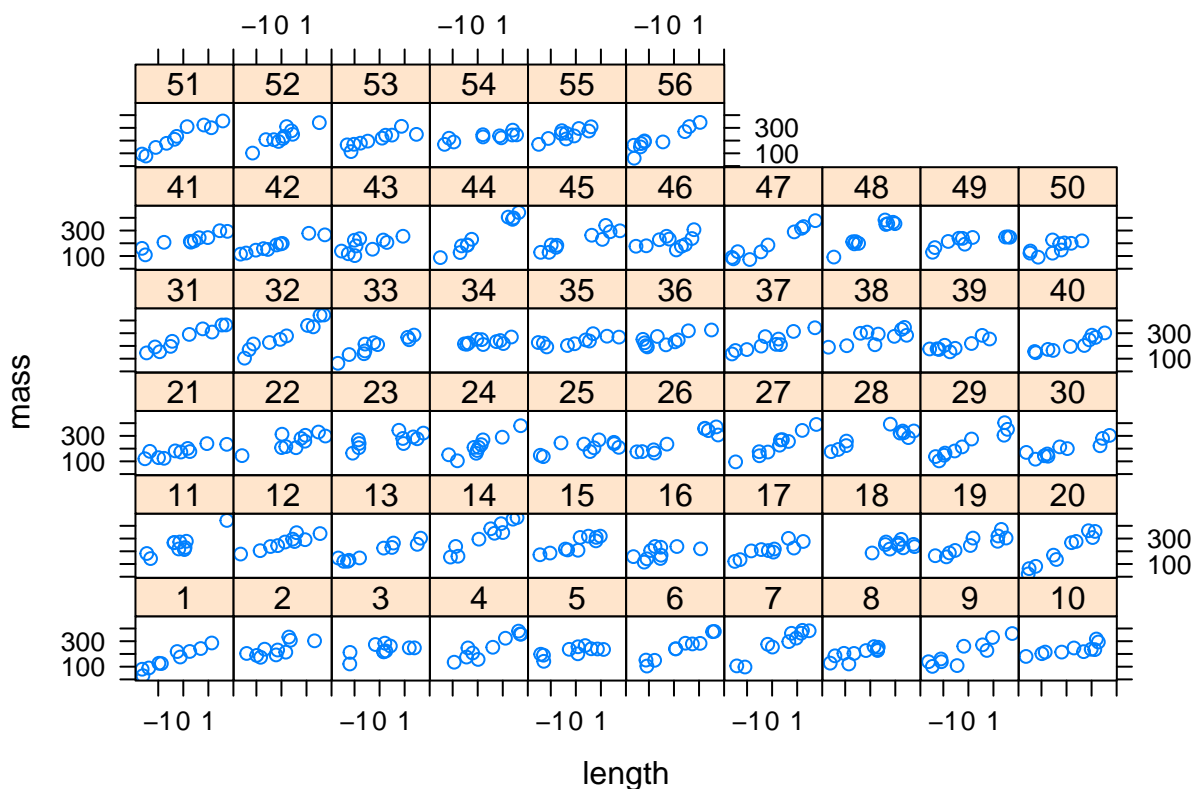
## Examine simulated data

```
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: 5763.146
## Random effects:
## Groups Name Std.Dev.
## pop (Intercept) 22.86
## Residual 38.87
## Number of obs: 560, groups: pop, 56
## Fixed Effects:
## (Intercept) length
## 230.04 60.97
```

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

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

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: mass ~ length + (1 | pop) + (0 + length | pop)
## REML criterion at convergence: 5610.291
## Random effects:
##   Groups      Name                Std.Dev.
##   pop         (Intercept) 21.94
##   pop.1       length      24.84
##   Residual                        30.33
## Number of obs: 560, groups: pop, 56
## Fixed Effects:
##   (Intercept)      length
##      230.29         60.19
```

## 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, sd, "\n\n")
```

```
## Mean and sd used to normalise.original length: 57.72975 7.034552
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.covariance,
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,] 247.5095 117.190353
## [2,] 213.3186  50.012729
## [3,] 216.7129  38.568336
## [4,] 224.8456  22.120889
## [5,] 217.7700 119.027027
## [6,] 234.9464  11.190415
## [7,] 213.7074  44.101412
## [8,] 260.8958  46.941491
## [9,] 237.7864  58.964155
## [10,] 222.3564  15.717432
## [11,] 219.5133  68.140491
## [12,] 209.6504  41.454354
## [13,] 243.4700  99.219669
## [14,] 197.1603  34.927048
## [15,] 252.8135  97.274954
## [16,] 251.4054  41.922869
## [17,] 258.8622 -6.525771
## [18,] 216.3109  61.058121
## [19,] 240.1424  47.564259
## [20,] 216.6882 -1.899624
## [21,] 240.7760  39.776863
## [22,] 236.9720  54.448297
## [23,] 196.3705  39.203522
## [24,] 229.5028  44.546125
## [25,] 251.7283  89.973130
## [26,] 220.5860  74.070061
## [27,] 237.5477  50.850614
## [28,] 220.4861 116.203733
## [29,] 244.9114 108.421188
## [30,] 238.4765  64.216505
## [31,] 236.2542  55.640207
## [32,] 219.6187  84.954393
## [33,] 245.6886 105.759053
## [34,] 212.1625  53.096532
## [35,] 240.1726   3.325184
## [36,] 247.2012  17.210284
## [37,] 216.3681  61.371122
## [38,] 240.2669  43.784690
## [39,] 201.7442  31.987222
## [40,] 221.4148  43.248842
## [41,] 253.3643  46.080339
## [42,] 216.1741  96.785531
## [43,] 242.7688  80.920167
## [44,] 204.9083  20.011030
```

```
## [45,] 210.5476 38.740584
## [46,] 230.2548 102.561121
## [47,] 251.8931 60.618026
## [48,] 173.8259 93.844125
## [49,] 250.2029 15.970737
## [50,] 225.1846 67.225732
## [51,] 208.2493 65.111852
## [52,] 249.6289 98.418515
## [53,] 227.2580 82.497934
## [54,] 217.5447 73.657079
## [55,] 223.4005 84.015270
## [56,] 239.4840 113.379228
```

```
apply(effects, 2, mean)
```

```
## [1] 228.90723 59.44456
```

```
var(effects)
```

```
##           [,1]      [,2]
## [1,] 327.24331 29.64843
## [2,] 29.64843 1050.52520
```

## Make correlated parameters

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

## Make linear model

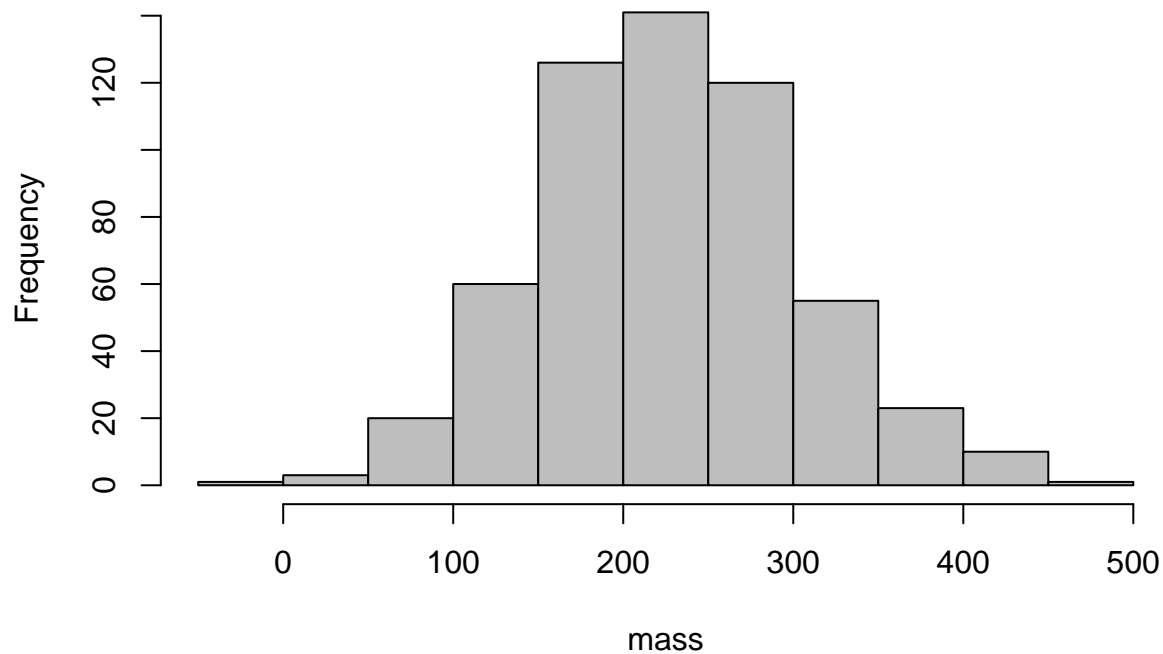
```
lin.pred <- Xmat[,] %*% all.effects # Value of lin.predictor
eps <- rnorm(n = n, mean = 0, sd = 30) # residuals
mass <- lin.pred + eps # response = lin.pred + residual
```

## View histogram

```
hist(mass, col = "grey") # Inspect what we've created
```

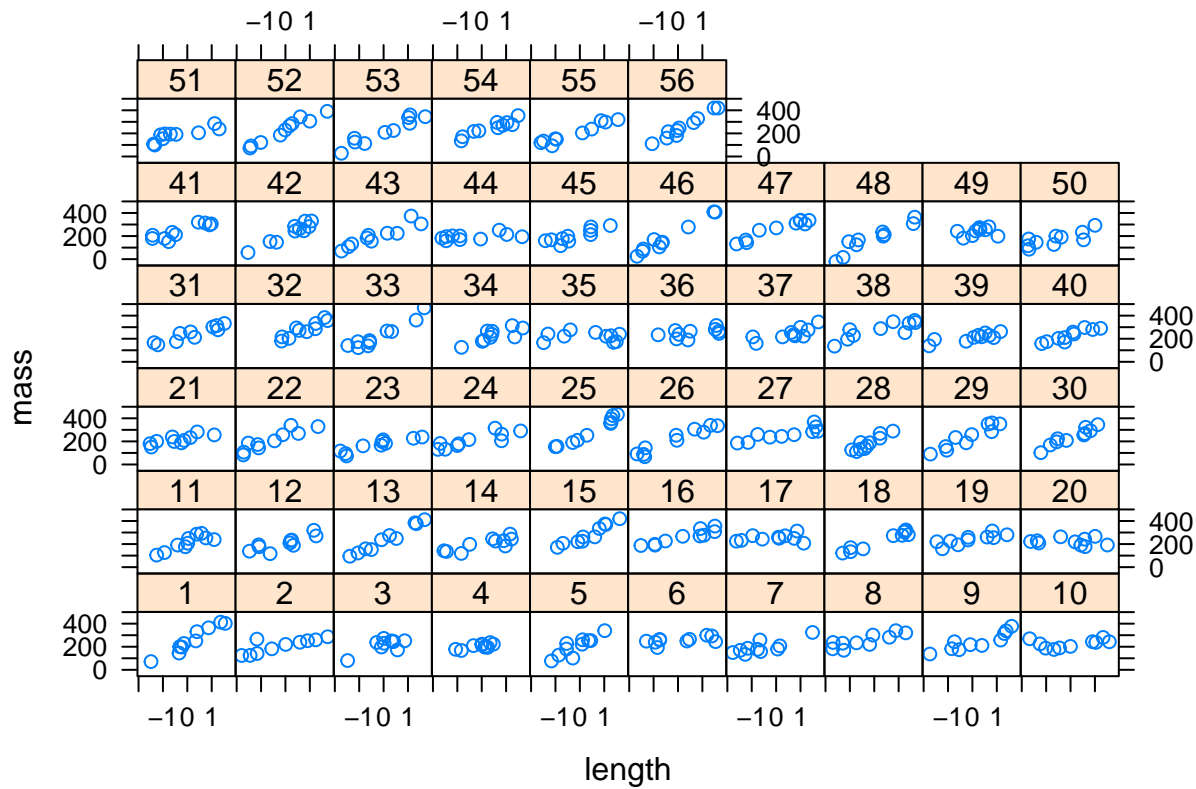


**Histogram of mass**



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: 5623.103
## Random effects:
## Groups   Name      Std.Dev. Corr
## pop      (Intercept) 16.83
##          length      30.93  0.02
## Residual                    30.83
## Number of obs: 560, groups:  pop, 56
## Fixed Effects:
## (Intercept)      length
##      226.4         61.9
```

## 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.