Modeling Variance with Multilevel/Mixed Models

Moving into Modeling Variance

So far we have be fitting

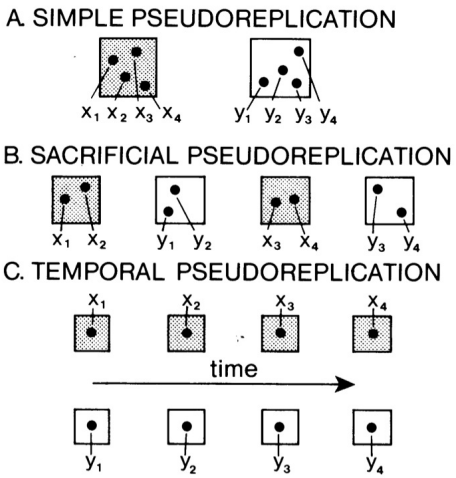
$$y_i = \beta_i X + \epsilon_i$$

where X is a number of predictors and $epsilon$ is random variation due to other processes. We assume data points are independent. But what if they're not? What if clusters of data points vary due to some random variation unique to just those points. We need a new model. One where

$$y_i = \alpha_{j[i]} + \beta_i X + \epsilon_{ij}$$

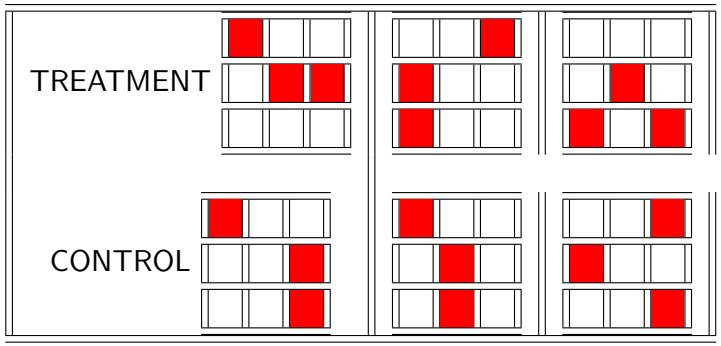
where i = individual data points, j = cluster, or group

This Framework Addresses Psuedoreplication Naturally



(Hurlbert 1984)

For Example, the Nested Design



Examples of Nesting

- Plots in a field with 1 treatment each
- Sampling a subject over time (where time doesn't influence the response)
- Gender of individuals (individual nested in gender)
- Experimental units manipulated by the same machine

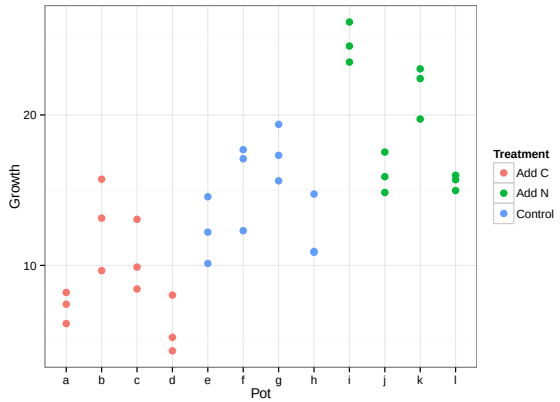
A Greenhouse Experiment testing C:N Ratios

Sam was testing how changing the C:N Ratio of soil affected plant leaf growth. He had 3 treatments. A control, a C addition, and a N addition. To ensure that any one measurement of one leaf wasn't a fluke, Sam measured 3 leaves per plant. The design is as follows:

3 Treatments (Control, C, N)
4 Pots of Plants per Treatment
3 Leaves Measured Per Pot

1) How many replicates are there per treatment?
2) Are measurements independent?
3) What do we use for our denominator Mean Square for F Test?
4) What is the denominator degrees of freedom?

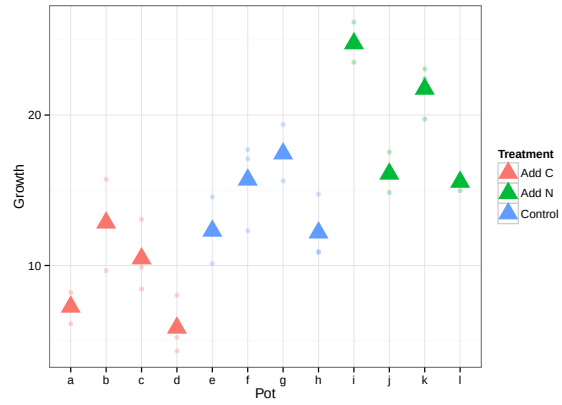
A Greenhouse Experiment testing C:N Ratios



Data Points are Not Independent!

Option 1: Averaging

If your design is balanced, and you don't care about the within pot variance, just average within each pot.



Option 1: Averaging

```
# Anova Table (Type II tests)
#
# Response: Growth
#           Sum Sq Df F value Pr(>F)
# Treatment    218  2    8.92 0.0073
# Residuals    110  9
```

You can use residuals to evaluate within plot variation.

Option 2: Classical ANOVA Error Decomposition with Expected Mean Squares

$$SS_{Total} = SS_{Treatment} + SS_{PotError} + SS_{WithinPotError}$$

```
plantAOV <- aov(Growth ~ Treatment + Error(Pot), data=plants)
```

```
summary(plantAOV)

#
# Error: Pot
#           Df Sum Sq Mean Sq F value Pr(>F)
# Treatment  2    653     327    8.92 0.0073
# Residuals  9    330      37
#
# Error: Within
#           Df Sum Sq Mean Sq F value Pr(>F)
# Residuals 24   97.7    4.07
```

Option 3: Multilevel/Clustered/Hierarchical/Mixed Model

$$y_i = \alpha_{j[i]} + \beta_i X + \epsilon_{ij}$$

$$\alpha_{j[i]} \sim N(\mu_\alpha, \sigma^2_\alpha)$$

$$\epsilon_{ij} \sim N(0, \sigma^2)$$

where i = individual sample, j = group

Types of Multilevel Models

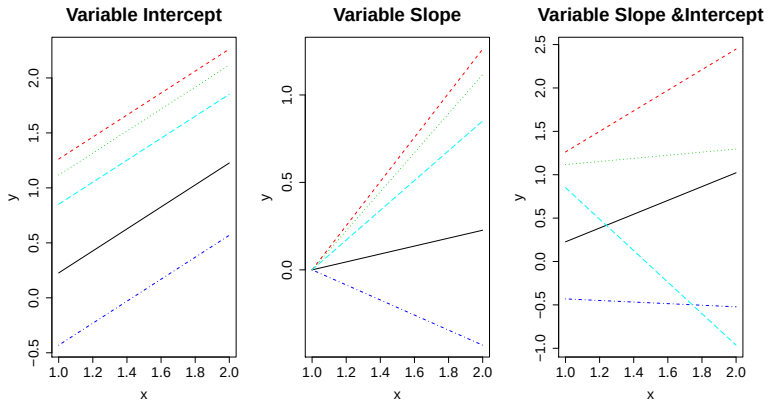
Varying Intercept: $y_i = \alpha_{j[i]} + \beta_i X + \epsilon_{ij}$
$\alpha_{j[i]} \sim N(\mu_\alpha, \sigma^2_\alpha)$

Varying Slope: $y_i = \alpha + \beta_{[j]i} X + \epsilon_{ij}$
$\beta_{j[i]} \sim N(\mu_\beta, \sigma^2_\beta)$

Varying Slope & Intercept: $y_i = \alpha_{j[i]} + \beta_{[j]i} X + \epsilon_{ij}$

$$\begin{pmatrix} \alpha_{[i]j} \\ \beta_{[i]j} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \begin{pmatrix} \sigma^2_\alpha & \rho\sigma_\alpha\sigma_\beta \\ \rho\sigma_\alpha\sigma_\beta & \sigma^2_\beta \end{pmatrix} \right)$$

Types of Multilevel Models



Unlike the General Linear Model, slopes and Intercepts are Constrained

Fixed versus Random Effects

Fixed Effect: Effects that are constant across populations.
Random Effect: Effects that vary are random outcomes of underlying processes.

Gelman and Hill (2007) see the distinction as artificial. Fixed effects are special cases of random effects where the variance is infinite. The model is what you should focus on.

You will also hear that 'random effects' are effects with many levels, but that you have not sampled all of them, wheras a for fixed effects, you have sampled across the entire range of variation. This is subtly different, and artificial.

Some Points about Multilevel Models

- Flexible. Can accomodate varying slope, intercept, intercept-slope models
- Solved using Restricted Maximum Likelihood (REML). ML estimation produces downward biased estimates of random effect variances.
- As group level effects are drawn from the same distribution, Best Linear Unbiased Predictors (BLUPs) are shrunk towards grand mean - basically, we use information from all groups to inform within group means - useful for unbalanced designs.
- We will use one formulation to evaluate DF for p values, etc., but this is an ongoing research topic.

This is a BIG Topic, We are Diving Skin Deep

Useful Texts (which may show up in the future)

- Gelman A, Hill J, (2006) Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press
- Zuur AF, Ieno EN, Walker NJ, Saveliev AA, Smith G (2009) Mixed Effects Models and Extensions in Ecology with R. Springer, New York.
- Pinheiro J, Bates D (2000) Mixed Effects Models in S and S-Plus. Springer-Verlag, New York, USA.

This is a BIG Topic, We are Diving Skin Deep

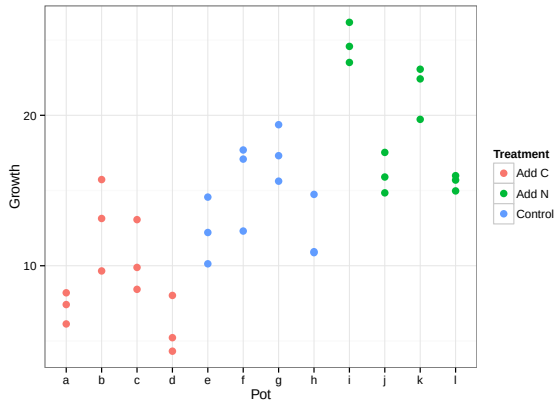
Blogs that Discuss Mixed Models Regularly

- `http://www.quantumforest.com/`
- `http://andrewgelman.com`
- `https://stat.ethz.ch/mailman/listinfo/r-sig-mixed-models`

Many R Packages for Multilevel Models

- nlme - from Pinhero and Bates 2009
- lmer - bleeding edge by Doug Bates
- MCMCglmm - uses Bayesian techniques & MCMC (next week!)
- glmmADMB - interface for AD Model Builder

Back to A Greenhouse Experiment testing C:N Ratios



Leaf Growth = Treatment Effect + Pot Variation + Error

Fitting a Varying Intercept Model for the Greenhouse Experiment

```
library(nlme)
plantLME <- lme(Growth ~ Treatment, random = ~ 1|Pot, data=plants)
```

Diagnostics:

1. Is there a relationship between fitted and residual values?
2. Are the residuals normally distributed?
3. Is there a relationship between fitted and residual values *at* the group level?
4. Are the random effects normally distributed?

Fitted Values at the Group or Individual Level

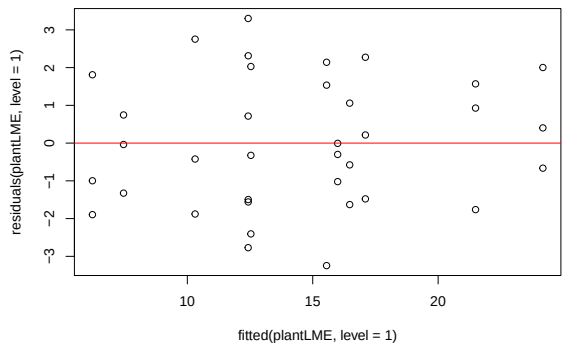
```
fitted(plantLME, level=0)


#      a      a      a      b      b      b      c      c
#  9.103  9.103  9.103  9.103  9.103  9.103  9.103  9.103
#      c      d      d      d      e      e      e      f
#  9.103  9.103  9.103  9.103 14.404 14.404 14.404 14.404
#      f      f      g      g      g      h      h      h
# 14.404 14.404 14.404 14.404 14.404 14.404 14.404 14.404
#      i      i      i      j      j      j      k      k
# 19.535 19.535 19.535 19.535 19.535 19.535 19.535 19.535
#      k      l      l      l
# 19.535 19.535 19.535 19.535
# attr(,"label")
# [1] "Fitted values"


fitted(plantLME, level=1)


#      a      a      a      b      b      b      c      c
#  7.457  7.457  7.457 12.426 12.426 12.426 10.311 10.311
#      c      d      d      d      e      e      e      f
# 10.311  6.216  6.216  6.216 12.534 12.534 12.534 15.552
#      f      f      g      g      g      h      h      h
# 15.552 15.552 17.100 17.100 17.100 12.429 12.429 12.429
#      i      i      i      j      j      j      k      k
# 24.177 24.177 24.177 16.475 16.475 16.475 21.493 21.493
#      k      l      l      l
# 21.493 15.995 15.995 15.995
```

Residual Values at the Group or Individual Level

```
residuals(plantLME, level=0)


#       a       a       a       b       b       b       c
# -2.9707 -1.6822 -0.8998  6.6254  0.5521  4.0388  0.7860
#       c       c       d       d       d       e       e
#  3.9623 -0.6705 -1.0759 -4.7816 -3.8838 -2.1944  0.1592
#       e       f       f       f       g       g       g
# -4.2751  2.6841  3.2901 -2.0975  1.2180  4.9706  2.9112
#       h       h       h       i       i       i       j
# -3.5338 -3.4706  0.3382  5.0448  3.9781  6.6445 -4.6892
#       j       j       k       k       k       l       l
# -3.6370 -2.0015  3.5274  0.1952  2.8855 -4.5608 -3.5465
#       l
# -3.8405
# attr(,"label")
# [1] "Residuals"


residuals(plantLME, level=1)


#         a         a         a         b         b         b
# -1.325521 -0.037016  0.745372  3.302139 -2.771118  0.715627
#         c         c         c         d         d         d
# -0.422165  2.754101 -1.878708  1.810270 -1.895354 -0.997627
#         e         e         e         f         f         f
# -0.324754  2.028845 -2.405449  1.535488  2.141472 -3.246080
#         g         g         g         h         h         h
# -1.478108  2.274437  0.215073 -1.558699 -1.495517  2.313292
```

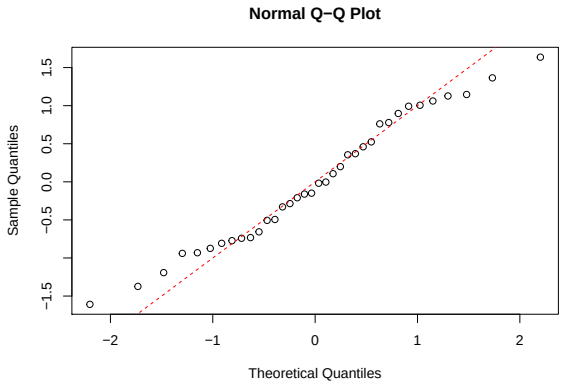
Residuals v. Fitted at Individual Level

```
plot(fitted(plantLME, level=1), residuals(plantLME, level=1))
abline(a=0, b=0,col="red")
```



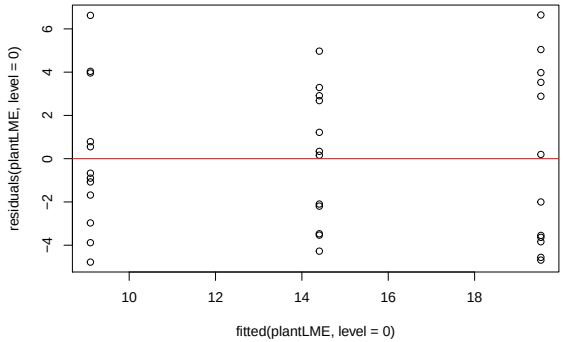
Normality of Residuals

```
qqnorm(residuals(plantLME,  type="pearson"))
abline(a=0,b=1, col="red", lty=2)
```



Residuals v. Fitted at Group Level

```
plot(fitted(plantLME, level=0), residuals(plantLME, level=0))
abline(a=0, b=0,col="red")
```

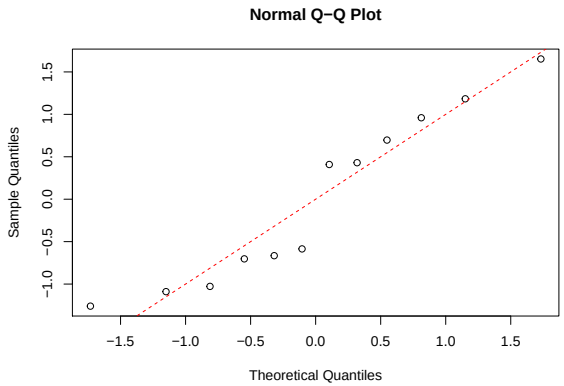


BLUPs of Random Effects

```
ranef(plantLME)

#   (Intercept)
# a      -1.645
# b       3.323
# c       1.208
# d      -2.886
# e      -1.870
# f       1.149
# g       2.696
# h      -1.975
# i       4.642
# j      -3.060
# k       1.958
# l      -3.540
```

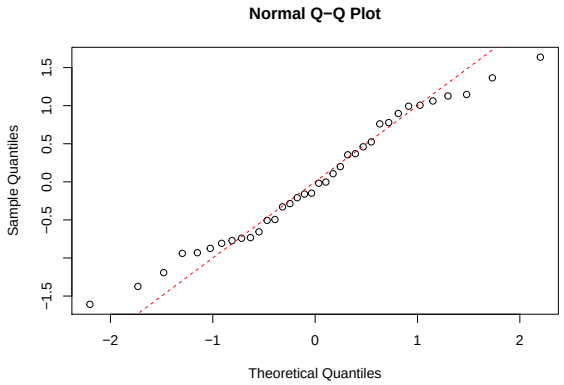
Normality of Random Effects

```
r<-ranef(plantLME)[,1]
qqnorm(r/sd(r))
abline(a=0,b=1, col="red", lty=2)
```



Normality of Residuals

```
qqnorm(residuals(plantLME,  type="pearson"))
abline(a=0,b=1, col="red", lty=2)
```



Evaluating the Greenhouse Experiment

```
anova(plantLME, type="marginal")

#             numDF denDF F-value p-value
# (Intercept)     1    24  27.150  <.0001
# Treatment       2     9   8.916  0.0073
```

DF Denominator = # Groups - DF Treatment - 1
Note type="marginal" - type II

Why F-Tests for Fixed Effects?

- F values calculated using differences in Residual Sums of Squares
- F tests with DF = # Groups - DF Treatment - 1 are conservative
- But, χ^2 tests for fixed effects are anti-conservative (type I prone)
- Use χ^2 tests for random effects - for a REML fit without any random effects, use `gls`

Random Effects

```
summary(plantLME)

# Linear mixed-effects model fit by REML
#  Data: plants
#      AIC   BIC logLik
#   177.2 184.7  -83.6
#
# Random effects:
#  Formula: ~1 | Pot
#         (Intercept) Residual
# StdDev:       3.294    2.018
#
# Fixed effects: Growth ~ Treatment
....
```

T-Tests for Fixed Effects

```
....

#                    Value Std.Error DF t-value p-value
# (Intercept)        9.103     1.747 24   5.211  0.0000
# TreatmentAdd N    10.432     2.471  9   4.223  0.0022
# TreatmentControl   5.301     2.471  9   2.146  0.0604
....
```

Correlation Between Fixed Effects

```
....

#  Correlation:
#                    (Intr) TrtmAN
# TreatmentAdd N     -0.707
# TreatmentControl -0.707  0.500
#
# Standardized Within-Group Residuals:
....
```

The Rest...

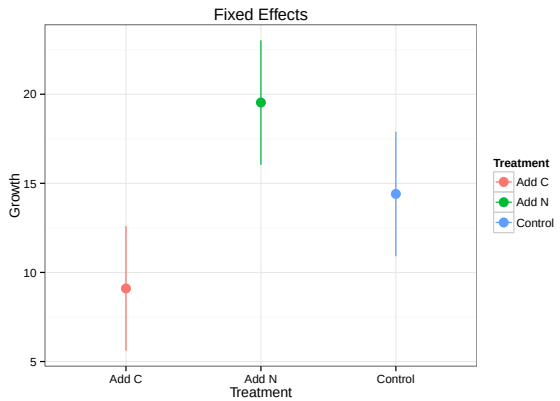
```
....

# Standardized Within-Group Residuals:
#      Min       Q1      Med       Q3      Max
# -1.60894 -0.73479 -0.08366  0.76529  1.63673
#
# Number of Observations: 36
# Number of Groups: 12
```

Fixed Effects v. Net Coefficients

```
fixef(plantLME)

#      (Intercept)   TreatmentAdd N TreatmentControl
#            9.103           10.432             5.301

coef(plantLME)

#   (Intercept) TreatmentAdd N TreatmentControl
# a       7.457          10.43             5.301
# b      12.426          10.43             5.301
# c      10.311          10.43             5.301
# d       6.216          10.43             5.301
# e       7.233          10.43             5.301
# f      10.251          10.43             5.301
# g      11.799          10.43             5.301
# h       7.127          10.43             5.301
# i      13.745          10.43             5.301
# j       6.043          10.43             5.301
```

Visualizing Fixed Effects

```
plantLME2 <- lme(Growth ~ Treatment-1, random = ~ 1|Pot, data=plants)
```

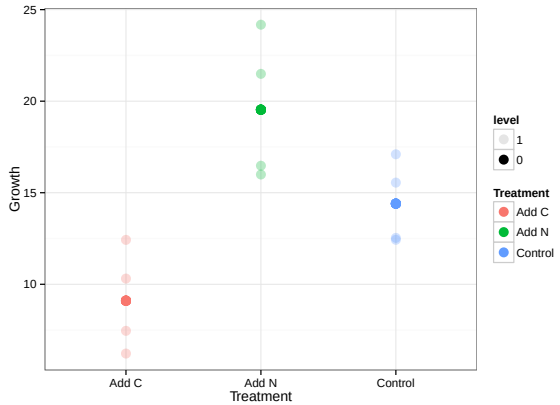


Note use of altered model for ease of plotting

Contrasts with Fixed Effects

```
library(contrast)
contrast(plantLME,
         list(Treatment = c("Add C", "Add N")),
         list(Treatment = "Control"))

# lme model parameter contrast
#
#  Contrast  S.E.    Lower   Upper     t df Pr(>|t|)
#    -5.301 2.471 -10.1435 -0.4591 -2.15 31   0.0398
#     5.131 2.471   0.2886  9.9730  2.08 31   0.0462
```

Visualizing Fixed and Random Effects



For more on confidence intervals, see
`http://glmm.wikidot.com/faq`

Exercise: Random Effects on Richness

- Fit data from RIKZ survey
- Random Effect of Beach ONLY
- Compare to No Beach Effect Model (gls)
- Visualize Random Effects

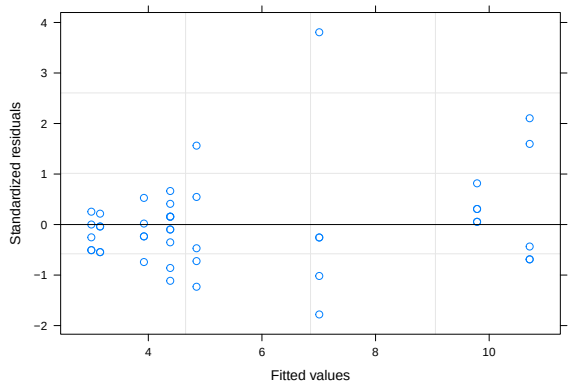
```
se.ranef <- function(obj)
  ranef(obj, standardized=T)/sapply(ranef(obj), sd)
```

Fitting Comparison Models

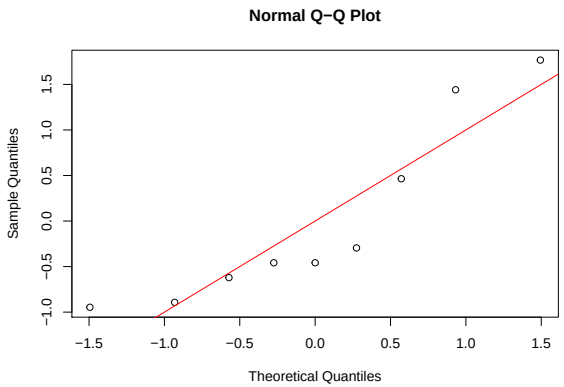
```
rikzInt <- lme(Richness ~ 1, random = ~1|Beach, data=rikz)
#
rikzNoBeach <- gls(Richness ~ 1,  data=rikz)
```

```
anova(rikzInt, rikzNoBeach)

#             Model df   AIC   BIC logLik   Test L.Ratio
# rikzInt         1  3 267.1 272.5 -130.6
# rikzNoBeach     2  2 274.4 277.9 -135.2 1 vs 2   9.255
#             p-value
# rikzInt
# rikzNoBeach  0.0023
```

Fit Is Ok...



Fit Is Ok...



Fit Is Ok...

```
se.ranef <- function(obj)
  ranef(obj, standardized=T)/sapply(ranef(obj), sd)
```