

# Assignment 4

Michelle Evans

October 30, 2017

```
overdisp_fun <- function(model) {  
  ## number of variance parameters in  
  ##   an n-by-n variance-covariance matrix  
  vpars <- function(m) {  
    nrow(m)*(nrow(m)+1)/2  
  }  
  model.df <- sum(apply(VarCorr(model), vpars))+length(fixef(model))  
  rdf <- nrow(model.frame(model))-model.df  
  rp <- residuals(model, type="pearson")  
  Pearson.chisq <- sum(rp^2)  
  prat <- Pearson.chisq/rdf  
  pval <- pchisq(Pearson.chisq, df=rdf, lower.tail=FALSE)  
  c(chisq=Pearson.chisq, ratio=prat, rdf=rdf, p=pval)  
}
```

## 1

Create function from script. Note that Plotfact is now included in the model under column PlotID:

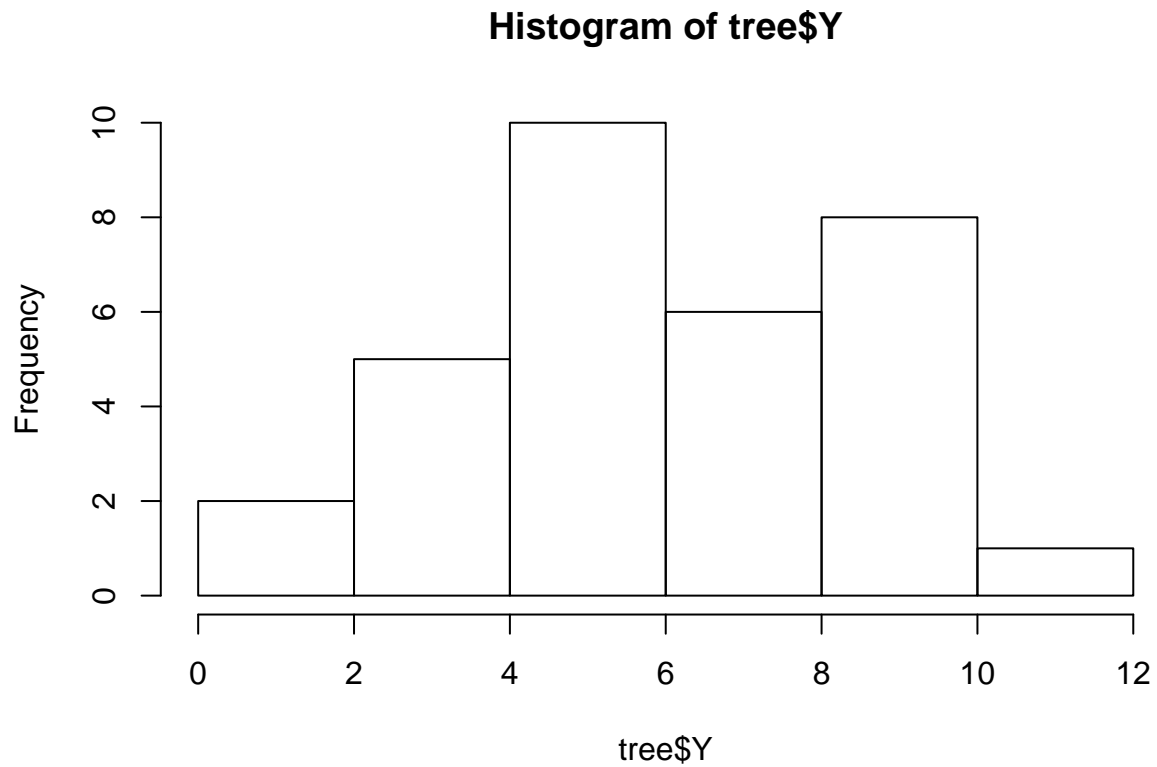
```
createDF <- function(N=32){  
  splist <- c('A', 'B', 'C')  
  Plot <- sample(1:5, N, replace=TRUE)  
  tree <- data.frame(Plot)  
  tree$Species <- factor(sample(splist, N, replace = TRUE))  
  tree$Y <- 0 # Placeholder for Y (growth rate)  
  alpha.A <- 8 # species A effect  
  alpha.B <- 6 # species B effect  
  alpha.C <- 4.5 # species C effect  
  sigma.sq <- 2 # standard deviation of epsilon error term  
  sigma.sq.b <- 0.5  
  b <- rnorm(5, mean=0, sd=sigma.sq.b) #plot level effects  
  epsilon <- rnorm(N, mean=0, sd=sigma.sq) # generate error terms  
  for (k in 1:N){  
    if (tree$Species[k]=="A") alpha <- alpha.A  
    else if (tree$Species[k]=="B") alpha <- alpha.B  
    else alpha <- alpha.C  
    tree$Y[k] <- alpha + b[tree$Plot[k]] + epsilon[k] # Growth model  
  }  
  tree$PlotID <- factor(tree$Plot) # Tell R that Plot is a categorical variable  
  return(tree)  
}
```

Create simulated data (N=32)

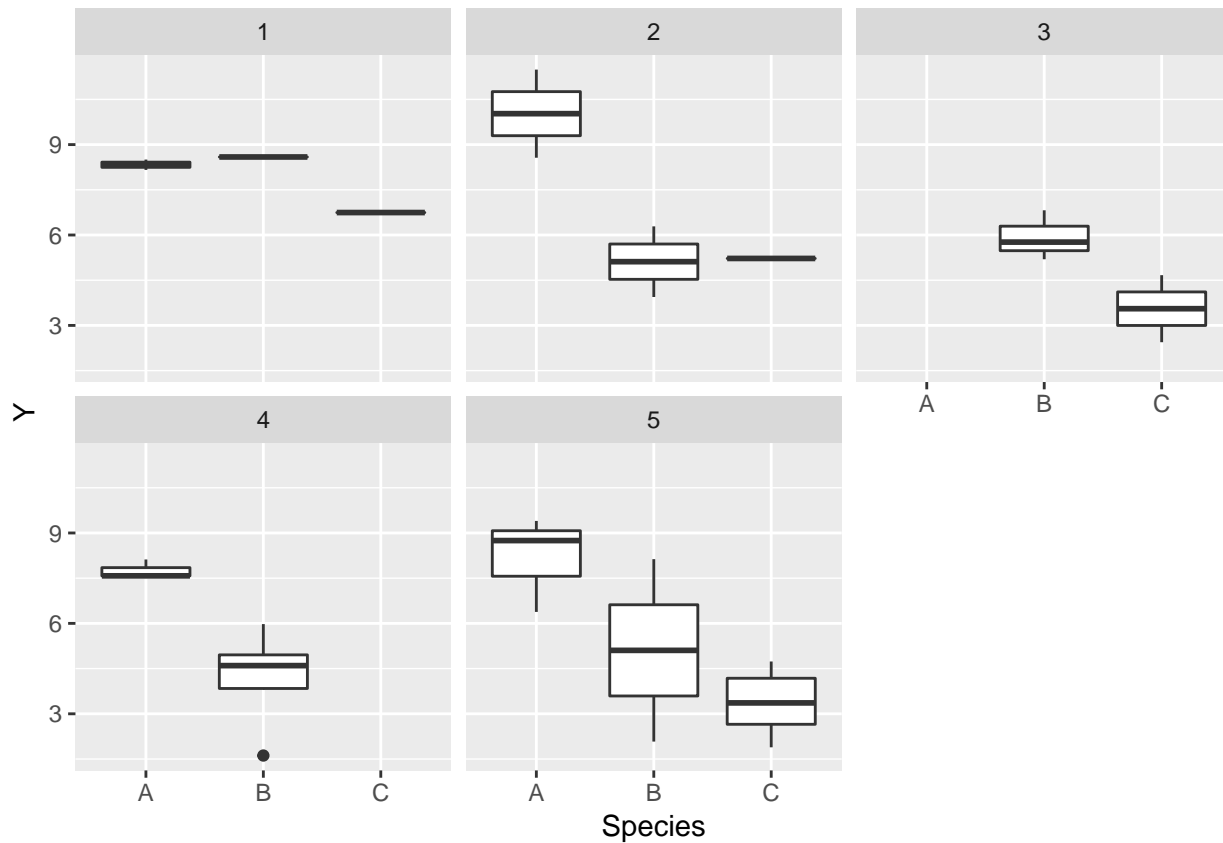
```
set.seed(8675309)  
tree <- createDF(N=32)
```

Visualize data:

```
hist(tree$Y)
```



```
ggplot(data=tree, aes(x=Species, y=Y))+  
  geom_boxplot()+  
  facet_wrap(~PlotID)
```



Analyze using lmer model:

```
mod0 <- lmer(Y~1 + (1|PlotID),
              data=tree)

mod1 <- lmer(Y~Species + (1|PlotID),
              data=tree)
summary(mod1)
```

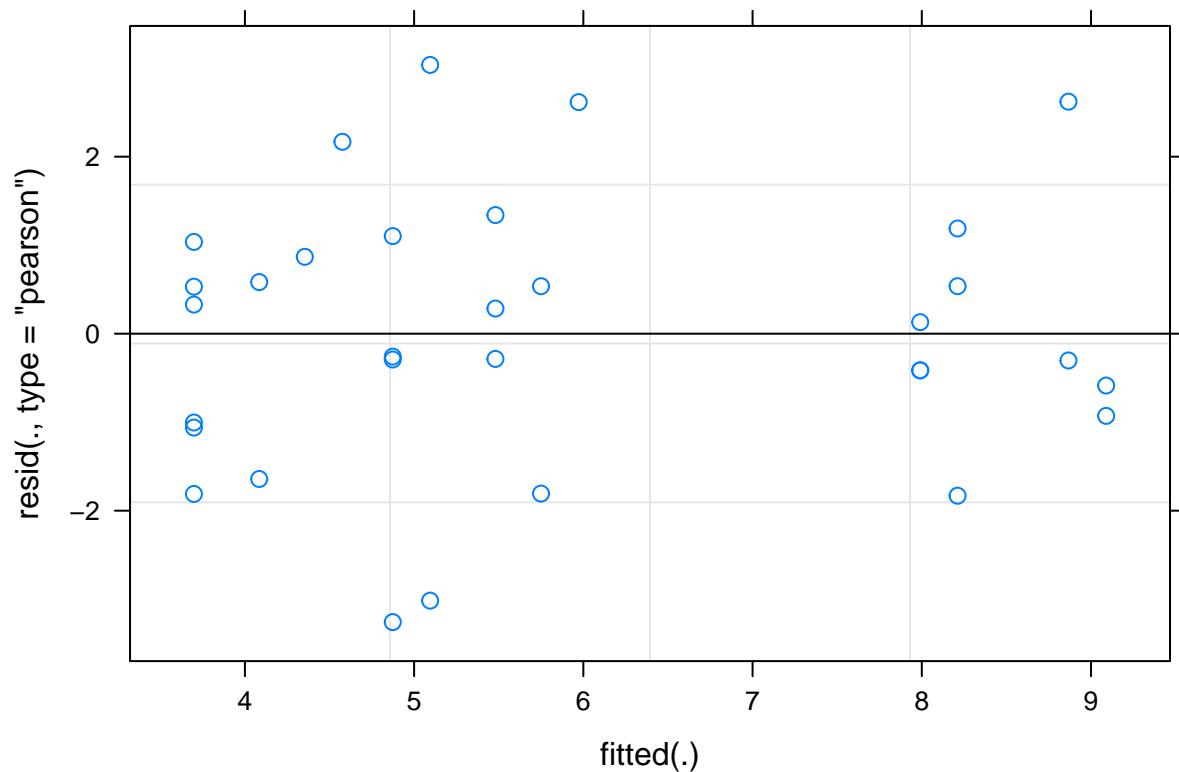
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Y ~ Species + (1 | PlotID)
## Data: tree
##
## REML criterion at convergence: 120.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.00504 -0.58301 -0.03963  0.56038  1.86847
##
## Random effects:
## Groups Name Variance Std.Dev.
## PlotID (Intercept) 0.4348  0.6594
## Residual 2.6416  1.6253
## Number of obs: 32, groups: PlotID, 5
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 8.5511 0.6040 14.158
```

```
## SpeciesB      -3.1168      0.7121    -4.377
## SpeciesC      -4.5127      0.7621    -5.922
##
## Correlation of Fixed Effects:
##      (Intr) SpecsB
## SpeciesB -0.647
## SpeciesC -0.599  0.500
anova(mod0, mod1)

## refitting model(s) with ML (instead of REML)

## Data: tree
## Models:
## mod0: Y ~ 1 + (1 | PlotID)
## mod1: Y ~ Species + (1 | PlotID)
##      Df      AIC      BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## mod0  3 154.50 158.90 -74.250   148.50
## mod1  5 132.16 139.49 -61.079   122.16 26.341    2 1.906e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(mod1)
```



```
overdisp_fun(mod1)
```

```
##      chisq      ratio      rdf      p
## 7.161976e+01 2.557848e+00 2.800000e+01 1.101710e-05
```

```
confint(mod1)
```

```
## Computing profile confidence intervals ...
```

```
## Warning in nextpar(mat, cc, i, delta, lowcut, upcut): unexpected decrease
## in profile: using minstep

## Warning in profile.merMod(object, which = parm, signames = oldNames, ...):
## non-monotonic profile for SpeciesC

## Warning in confint.thpr(pp, level = level, zeta = zeta): bad spline fit for
## SpeciesC: falling back to linear interpolation

##           2.5 %    97.5 %
## .sig01      0.000000  1.692351
## .sigma      1.228547  2.111413
## (Intercept)  7.404737  9.730304
## SpeciesB    -4.526124 -1.747099
## SpeciesC    -5.990128 -3.040982
```

The model is overdispersed, however the effect of Species is significant, as measured via a comparison to the null model with an anova and 95% confidence intervals.

## 2

Aggregate data

```
treePlot <- tree %>%
  group_by(PlotID, Species) %>%
  summarise(growth = mean(Y))
```

Re-analyze:

```
modNew0 <- lmer(growth ~ 1 + (1|PlotID),
  data = treePlot)
modNew1 <- lmer(growth ~ Species + (1|PlotID),
  data = treePlot)
summary(modNew1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: growth ~ Species + (1 | PlotID)
## Data: treePlot
##
## REML criterion at convergence: 39.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -0.9691 -0.6662 -0.1357  0.3048  1.5347
##
## Random effects:
## Groups Name Variance Std.Dev.
## PlotID (Intercept) 0.7374  0.8587
## Residual 1.4240  1.1933
## Number of obs: 13, groups: PlotID, 5
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 8.5172 0.7225 11.789
## SpeciesB -2.7306 0.8120 -3.363
## SpeciesC -3.9436 0.8720 -4.523
```

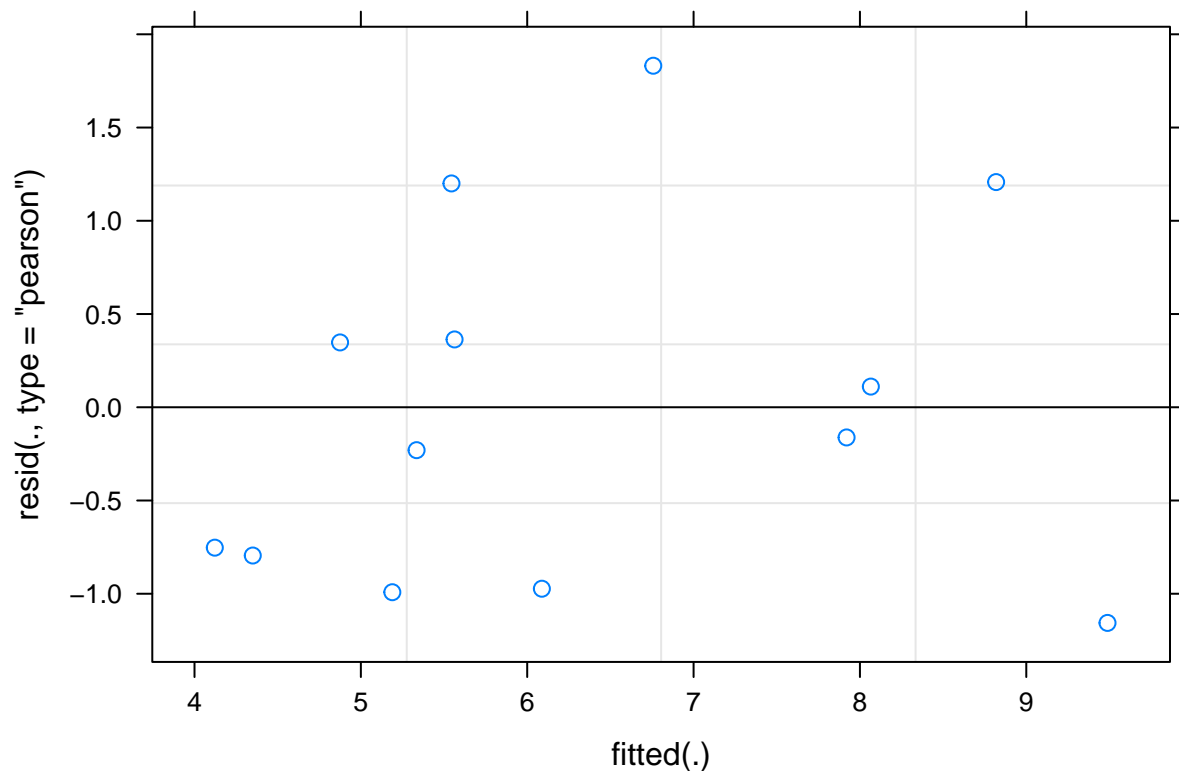
```
##
## Correlation of Fixed Effects:
##      (Intr) SpecsB
## SpeciesB -0.638
## SpeciesC -0.603  0.537
anova(modNew0, modNew1)

## refitting model(s) with ML (instead of REML)

## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.derivs
## = TRUE, : convergence code 3 from bobyqa: bobyqa -- a trust region step
## failed to reduce q

## Data: treePlot
## Models:
## modNew0: growth ~ 1 + (1 | PlotID)
## modNew1: growth ~ Species + (1 | PlotID)
##      Df      AIC      BIC logLik deviance Chisq Chi Df Pr(>Chisq)
## modNew0  3 61.350 63.044 -27.675  55.350
## modNew1  5 52.309 55.134 -21.155  42.309 13.04      2  0.001473 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

plot(modNew1)
```



```
overdisp_fun(modNew1)

##      chisq      ratio      rdf      p
## 11.0638389  1.2293154  9.0000000  0.2713585

confint(modNew1)
```

```
## Computing profile confidence intervals ...
```

```
##           2.5 %    97.5 %
## .sig01      0.0000000  2.090877
## .sigma      0.6840788  1.832964
## (Intercept)  7.1478525  9.864834
## SpeciesB    -4.3112225 -1.189418
## SpeciesC    -5.3322352 -2.270201
```

The results are qualitatively similar to those above. The effect of species is significant in both cases, with Species A > Species B > Species C in growth rates. However, the model at the plot level is not overdispersed as the one at the level of the individual tree is.

### 3

```
set.seed(8675309)
Nvec <- c(32, 64, 100, 150, 200)
results <- data.frame(N = Nvec,
                      pvalueInd = NA,
                      pvaluePlot = NA)
for (i in 1:length(Nvec)){
  treeDF <- createDF(N = Nvec[i])

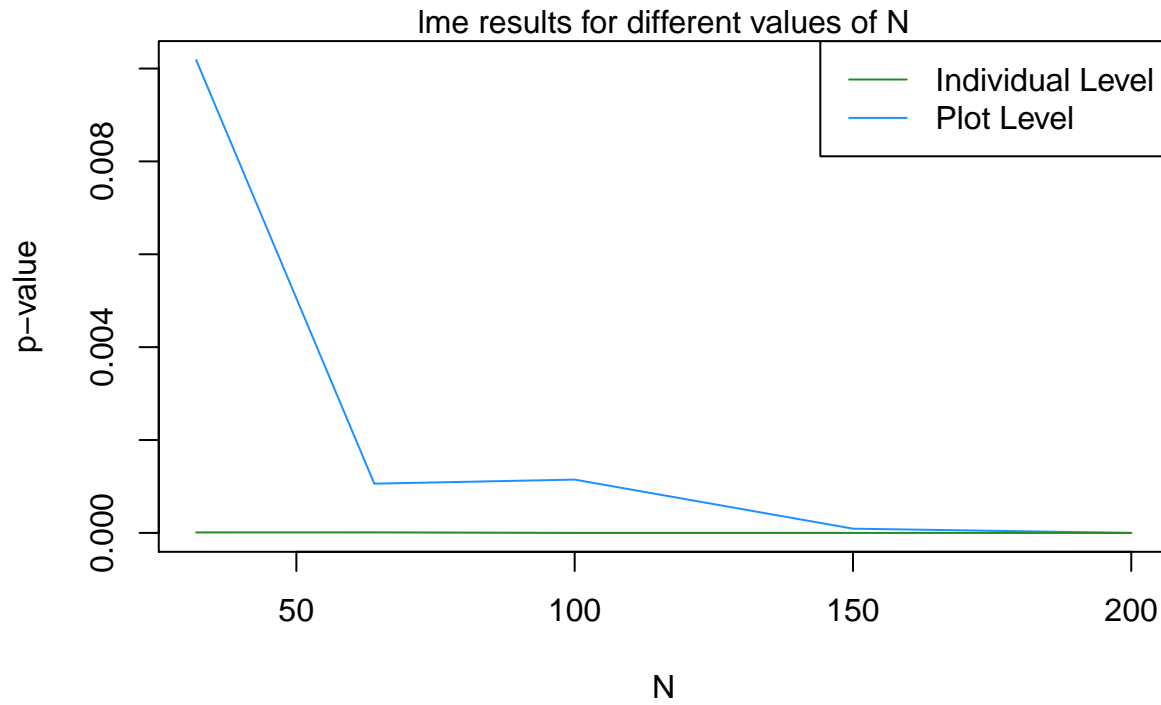
  modInd <- lme(Y ~ Species,
               random = ~1|PlotID,
               data = treeDF)

  treePlot <- treeDF %>%
    group_by(Species, PlotID)%>%
    summarise(Y = mean(Y))

  modPlot <- lme(Y ~ Species,
                random = ~1|PlotID,
                data = treePlot)

  results$pvalueInd[i] <- anova(modInd)["Species", "p-value"]
  results$pvaluePlot[i] <- anova(modPlot)["Species", "p-value"]
}

plot(results$N, results$pvaluePlot, col = NA, xlab = "N", ylab = "p-value")
lines(results$N, results$pvaluePlot, col = "dodgerblue")
lines(results$N, results$pvalueInd, col = "forestgreen")
legend(col = c("forestgreen", "dodgerblue"),
       lty = c(1,1),
       legend = c("Individual Level", "Plot Level"),
       "topright")
mtext("lme results for different values of N")
```



Analyzing data at the plot level reduces the power of the analysis because information is lost by aggregating to the mean, however as N increases, this becomes less of an issue because the variance within a species becomes smaller (even though information is being lost, the mean is more precise). It is important to note, however, that this is the result of one simulation, and rerunning it with different seeds results in some qualitatively different results.