Homework 2 Suglia

Elena Suglia

11/5/2019

Question 1

Using the Char Height data set from class (char_with_fake.csv), construct a model with random effects for these data, using CharHt as the response variable. Assume that Transect is the only relevant grouping factor, and that Steepness (of the topography) and Diameter (of the trees) are the only available predictors. In reporting about the model please include:

- A brief explanation of how you chose variables, and which (if any) you decided to allow to vary by group (Transect).

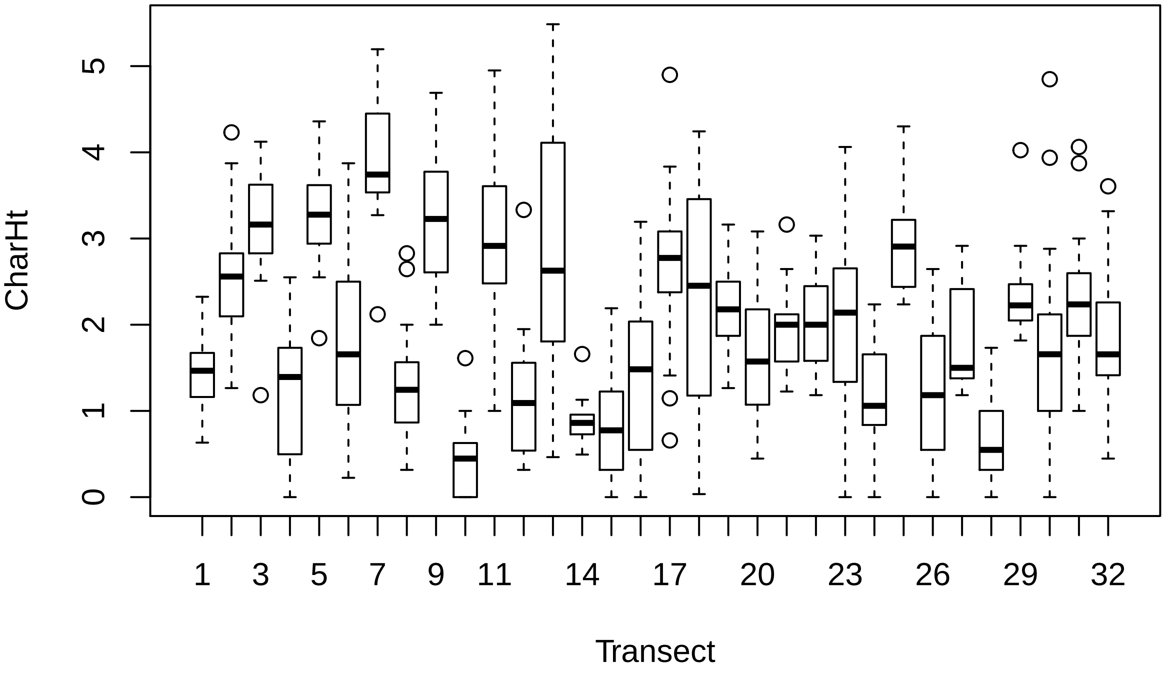
- An assessment of how much variation there is in the group-level random effects.

- A brief assessment of how well your selected model fits the data.

Notes about this data set: *This data set adds a fake, randomly generated predictor that is just noise (rnorm). Why are we doing this? Recall that the penalty term in AIC is designed to offset the improvement in fit you would typically get if you added a randomly generated explanatory variable to your model – one that has no "true" relationship to the response variable.*

Look at variation in data across groups

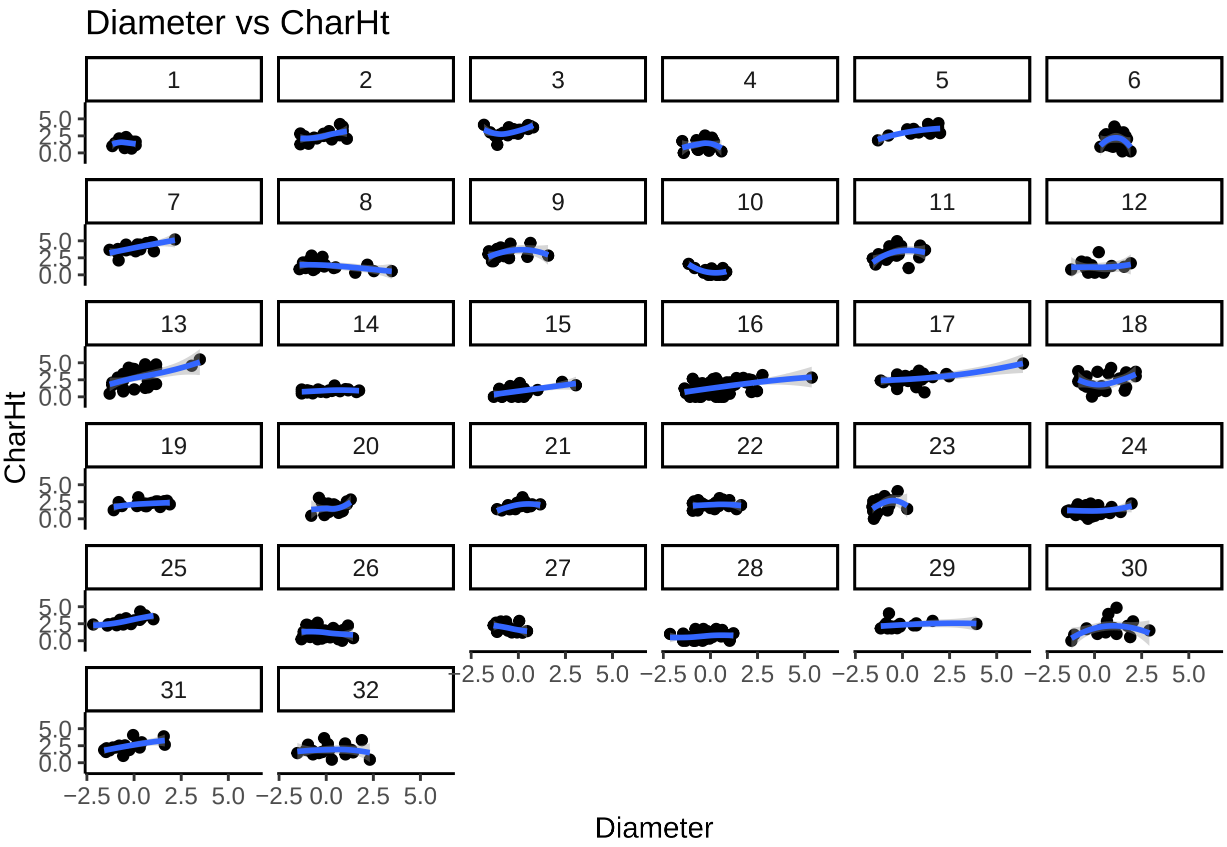
```r
boxplot(CharHt~Transect, d, xlab = "Transect")
```



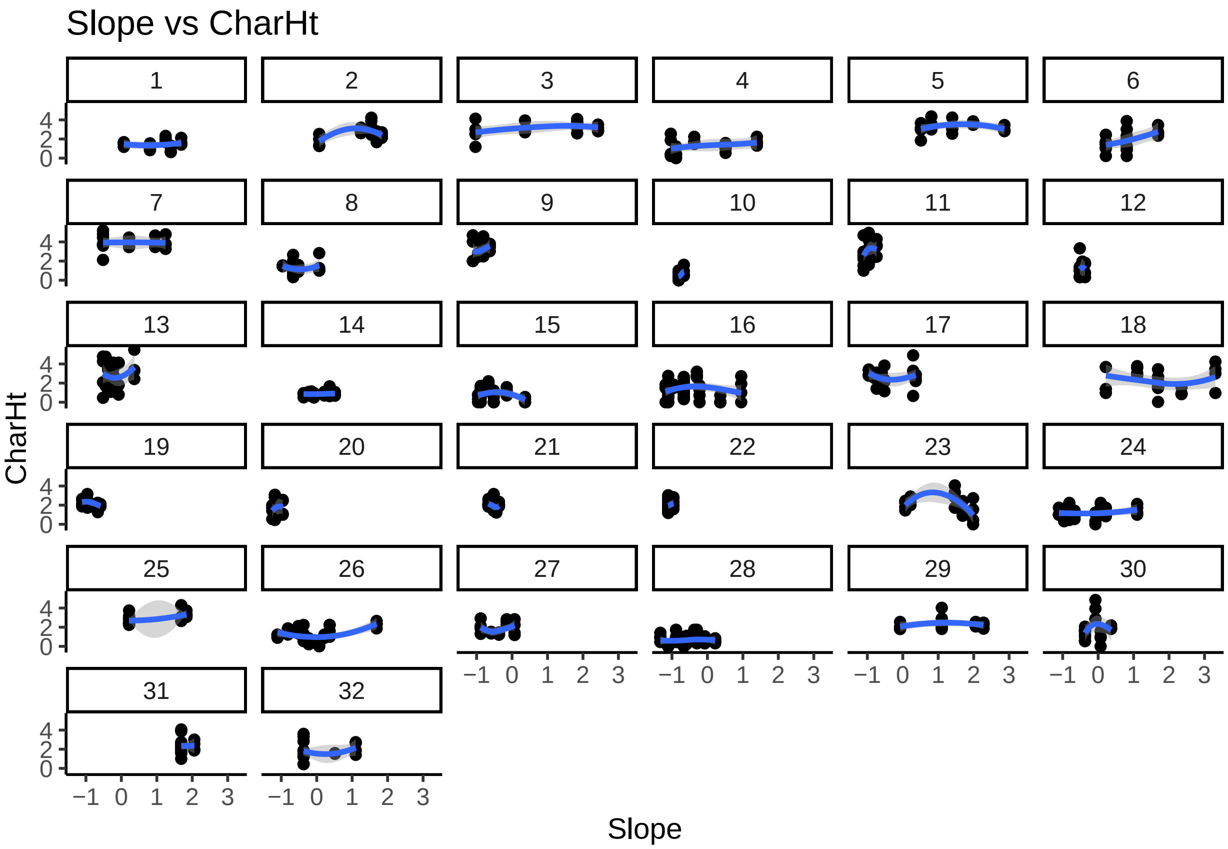
Lots of variation among groups.

What about relationships between predictor and response variables?

```
ggplot(d, aes(x=Diameter, y=CharHt)) +
  geom_point() +
  geom_smooth(span=2) +
  theme_classic() +
  ggtitle("Diameter vs CharHt") +
  xlab("Diameter") +
  ylab("CharHt") +
  facet_wrap(~Transect)
```



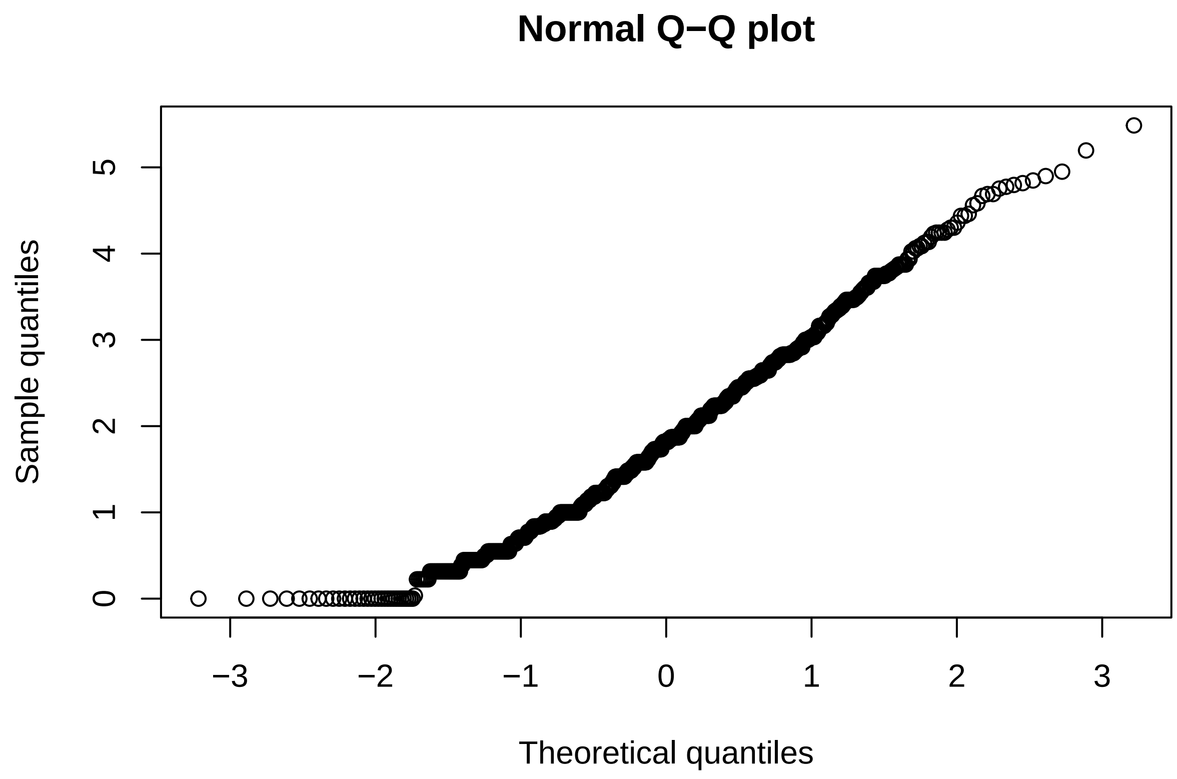
```
ggplot(d, aes(x=Slope, y=CharHt)) +
  geom_point() +
  geom_smooth(span=2) +
  theme_classic() +
  ggtitle("Slope vs CharHt") +
  xlab("Slope") +
  ylab("CharHt") +
  facet_wrap(~Transect)
```



There appears to be some variation in response to both explanatory variables.

Check normality using a Q-Q plot

```
qqnorm(d$CharHt, main = "Normal Q-Q plot", xlab = "Theoretical quantiles", ylab = "Sample quantiles")
```



- Not perfectly normal but not too bad either

Check for collinearity (correlation between explanatory variables)

```
x = select(d, Slope, Diameter)
round(cor(x), 2)
```

```
##          Slope Diameter
## Slope     1.00    -0.02
## Diameter -0.02     1.00
```

They're only correlated a little (0.2); I think this is acceptable

Fit model

First, decide which predictor(s) to include; slope and/or diameter (fixed effects models):

```
# Let's fit some models and compare them

# Fixed effects only
m1 = lm(CharHt~Slope, data =d)
m2 = lm(CharHt~Diameter, data = d)
m3 = lm(CharHt~Slope+Diameter, data = d)
m4 = lm(CharHt~Slope*Diameter, data = d)

# Mixed models including both fixed and random effects
m5 <- lmer(CharHt~Slope+(1|Transect), data=d, REML = FALSE)
```

```
m6 <- lmer(CharHt~Diameter+(1|Transect), data=d, REML = FALSE)
m7 <- lmer(CharHt~Diameter+(1+Slope|Transect), data=d, REML = FALSE) # failed to converge
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.00277968
## (tol = 0.002, component 1)
```

```
m8 <- lmer(CharHt~Slope+(1+Diameter|Transect), data=d, REML = FALSE) # best fit
m9 <- lmer(CharHt~Diameter+Slope+(1|Transect), data=d, REML=FALSE) # close second best fit
m10 <- lmer(CharHt~Diameter*Slope+(1|Transect), data=d, REML = FALSE)

AIC(m1, m2, m3, m4, m5, m6, m7, m8, m9, m10)
```

```
##     df      AIC
## m1   3 2351.505
## m2   3 2371.017
## m3   4 2319.466
## m4   5 2317.610
## m5   4 1900.769
## m6   4 1846.443
## m7   6 1847.964
## m8   6 1838.063
## m9   5 1838.334
## m10  6 1839.534
```

```
BIC(m1, m2, m3, m4, m5, m6, m7, m8, m9, m10)
```

```
##     df      BIC
## m1   3 2365.468
## m2   3 2384.980
## m3   4 2338.082
## m4   5 2340.881
## m5   4 1919.385
## m6   4 1865.060
## m7   6 1875.888
## m8   6 1865.988
## m9   5 1861.605
## m10  6 1867.459
```

Including the random effects improves model fit. The model with varying slopes + intercepts for the explanatory variable Diameter fits the best.

How much variation is there in the group-level random effects?

```
display(m8)
```

```
## lmer(formula = CharHt ~ Slope + (1 + Diameter | Transect), data = d,
##     REML = FALSE)
##             coef.est coef.se
## (Intercept) 1.65     0.13
## Slope       0.11     0.05
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  Transect (Intercept) 0.88
##           Diameter    0.29     0.69
##  Residual             0.72
## ---
## number of obs: 776, groups: Transect, 32
## AIC = 1838.1, DIC = 1826.1
## deviance = 1826.1
```

- Standard deviation in the varying Intercept for transect is 0.88
- S.d of varying slope for Diameter is 0.29
- The residual error is 0.72

Do we include the random effects or not?

Use MuMIn package to compare modified R2 values for models with fixed effects only vs models with both fixed and random effects.

```
# Useful notes from the help page for the package:
# The marginal R2 value represents the variance explained by the fixed effects
# The conditional R2 value is interpreted as the variance explained by
# the entire model, including both fixed and random effects

library(MuMIn)
r.squaredGLMM(m5)
```

```
##             R2m       R2c
## [1,] 0.01541361 0.5249388
```

```
r.squaredGLMM(m6)
```

```
##             R2m       R2c
## [1,] 0.04552809 0.5801066
```

```
r.squaredGLMM(m7)
```

```
##             R2m       R2c
## [1,] 0.04708417 0.5876593
```

```
r.squaredGLMM(m8)
```

```
##              R2m       R2c
## [1,] 0.009462121 0.6280514
```

```
r.squaredGLMM(m9)
```

```
##             R2m       R2c
## [1,] 0.06270771 0.5706351
```

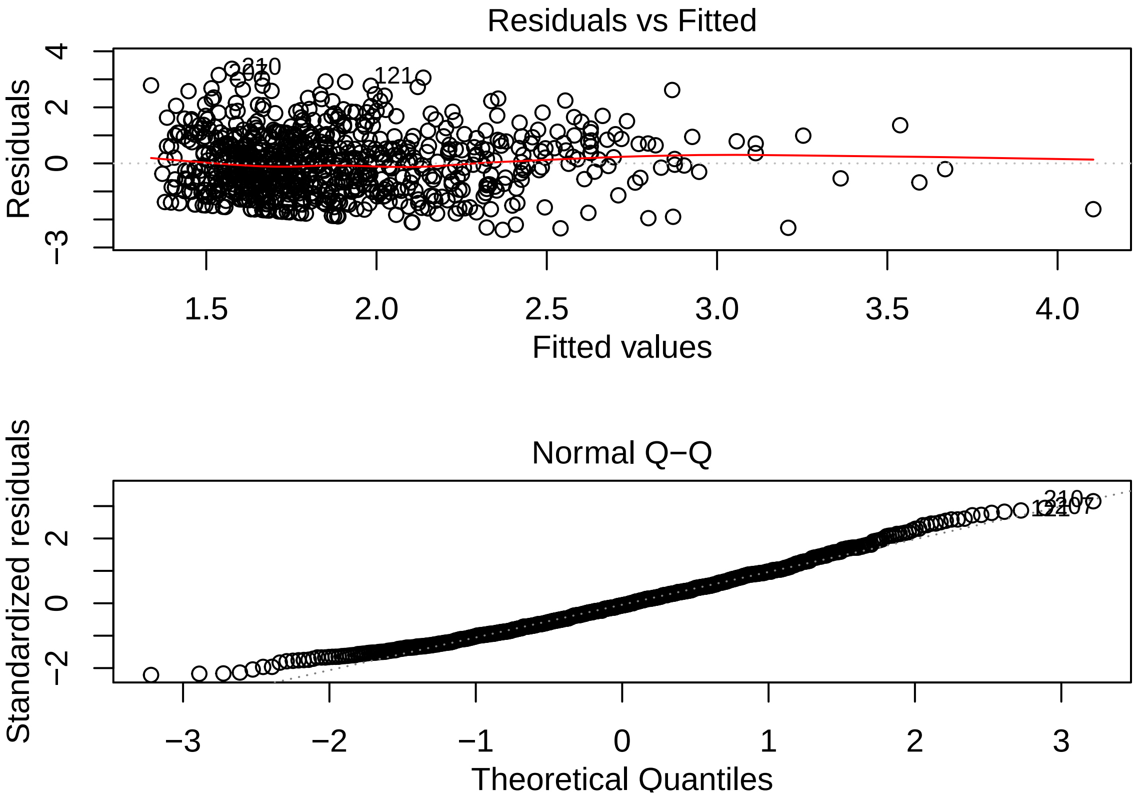
```
r.squaredGLMM(m10)
```

```
##             R2m       R2c
## [1,] 0.06475953 0.5703532
```

Including the random effects does allow the model to explain more of the variation in the response variable. The R2 values are also pretty good, explaining 50-60% of the variation.

Testing model assumptions

```r
par(mfrow=c(2, 1), mar=rep(3,4), mgp=c(2,1,0))
plot(m4, which=1:2)
```



These assumptions appear to be adequately met

Cross-validation

Another way to assess model performance is with cross-validation.

There are 6 fires in total: let's withhold 3 and allow the model to predict the last 3:

```r
d.fit <- filter(d, Fire %in% (1:3)) # training data
d.holdout <- filter(d, Fire %in% (4:6)) # prediction data
```

Fit the models

```r
char.m1 <- lmer(CharHt~Slope+(1+Diameter|Transect), data=d.fit, REML = FALSE)
char.m2 <- lmer(CharHt~Slope*fake+(1+Diameter|Transect), data=d.fit, REML = FALSE)
```

Compare raw sum of squared error and penalized fit terms

```r
# mean squared error of the model fit
mean(resid(char.m1)^2)
```

```
## [1] 0.5397458
```

```
mean(resid(char.m2)^2)
```

```
## [1] 0.5396532
```

```
mean((predict(char.m1, newdata=d.holdout, allow.new.levels = TRUE) - d.holdout$CharHt)^2)
```

```
## [1] 0.7684981
```

```
mean((predict(char.m2, newdata=d.holdout, allow.new.levels = TRUE) - d.holdout$CharHt)^2)
```

```
## [1] 0.7697621
```

```
# Which model does better in this cross-validation test?
```

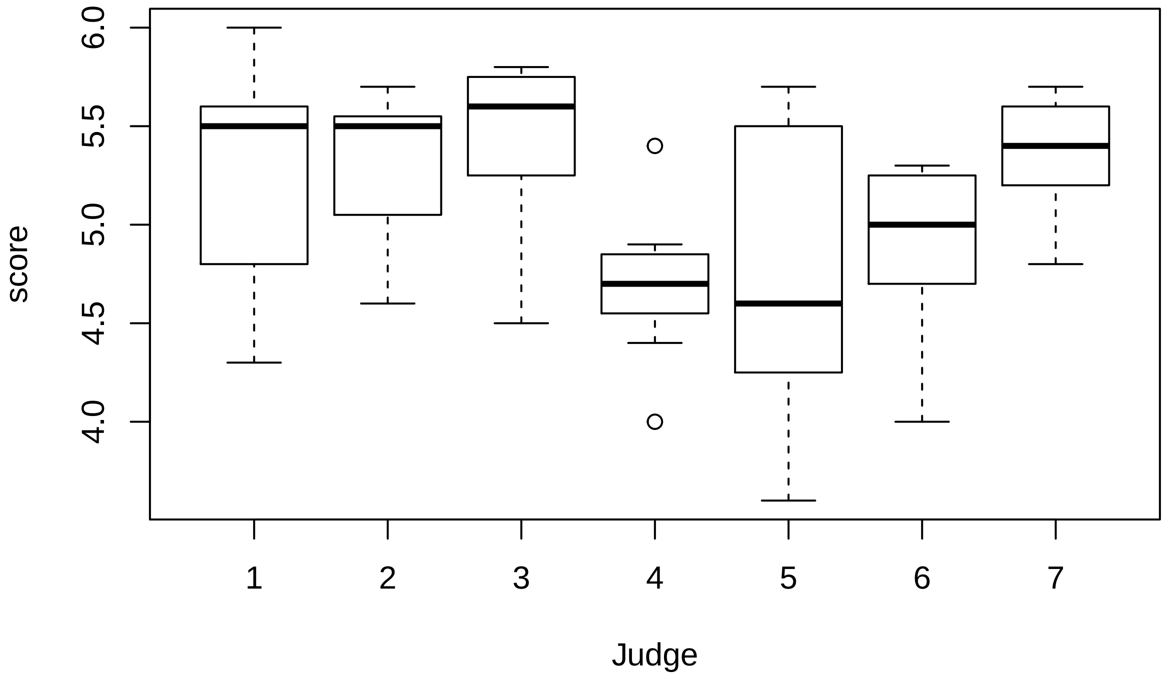
They both appear to be performing equally well; maybe this is because the fake randomly varying data is swamped by the rest of the data? Not sure how to interpret this particular result, but I think the model is performing adequately well.

Question 2

Get data on performance scores for pairs figure skating in the 1932 olympics (from http://www.stat.columbia.edu/~gelman/arm/examples/olympics/olympics1932.txt). This is formatted for R as "olympics.csv" on Smartsite in the homework folder.

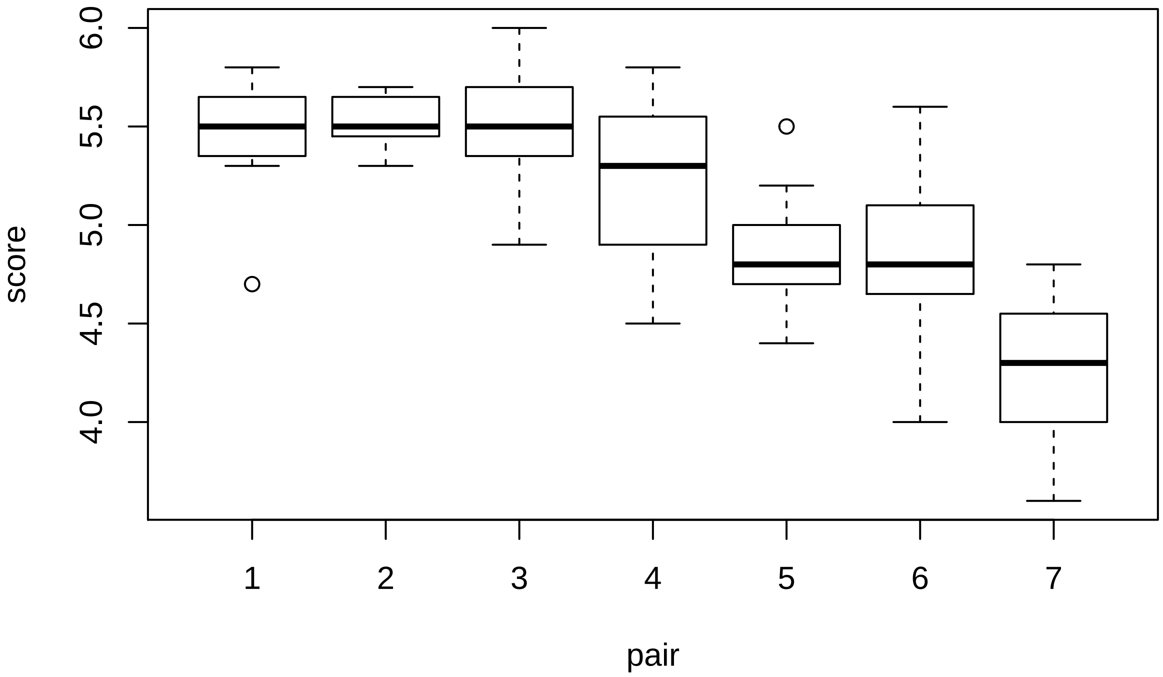
Let's assume the question is: which is the bigger source of variation in the scores for skating programs, the judges or the skating pair? Fit a mixed model for these data with "score" as the response variable, and random effects for judge and skating pair ("judge" and "pair"). Interpret the results (coefficients and their standard errors, standard errors of the random effects). Is there a judge that tends to give consistently higher scores?

```
d2 = read.csv("olympics.csv", header = TRUE)
boxplot(score~judge, d2, xlab = "Judge")
```



Just by looking at the graphs, judge 7 appears to give consistently higher scores than the others, and perhaps judge 2 as well

```
boxplot(score~pair, d2, xlab = "pair")
```



Out of curiosity, looked at variation among pairs; not much variation of scores within pairs but plenty of variation among pairs

Fit model

```
mskate <- lmer(score~ (1|judge) + (1|pair), data = d2)
```

Interpet results

```
display(mskate)
```

```
## lmer(formula = score ~ (1 | judge) + (1 | pair), data = d2)
## coef.est  coef.se
##     5.09     0.20
##
## Error terms:
##  Groups   Name        Std.Dev.
##  judge    (Intercept) 0.28
##  pair     (Intercept) 0.45
##  Residual             0.27
## ---
## number of obs: 49, groups: judge, 7; pair, 7
## AIC = 54.2, DIC = 43.4
## deviance = 44.8
```

```
summary(mskate)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: score ~ (1 | judge) + (1 | pair)
##    Data: d2
##
## REML criterion at convergence: 46.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.10128 -0.50470 -0.09885  0.40875  2.10488
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  judge    (Intercept) 0.07758  0.2785
##  pair     (Intercept) 0.20486  0.4526
##  Residual             0.07446  0.2729
## Number of obs: 49, groups:  judge, 7; pair, 7
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   5.0918     0.2046   24.89
```

- Standard deviation of random effect for pair is higher than that for judge
- I would conclude that pair explains more of the variation in scores than judge does