Supplemental Material

Load data

```
load("~/HLM/Sample_data.Rda") #note, you will need to insert the path to the data file here
```

Load packages

```
library(tidyr)
library(lme4)
library(multcomp)
library(Matrix)
library(ggplot2)
library(lattice)
library(stringr)
library(dplyr)
```

Creating new variables

```
  class_means <- SampleData %>% group_by(crse_id) %>% summarise(pre_mean_class = mean(pre_scor)) # Crea
  class_means$class_pre_cent <- class_means$pre_mean_class - mean(class_means$pre_mean_class) # Grand c
  SampleData <- left_join(SampleData,class_means, by="crse_id") # adds the course means back into the s
  SampleData$stud_pre_cent <- SampleData$pre_scor - SampleData$pre_mean_class # Group centers student p
  SampleData$stud_pre_grand <- SampleData$pre_scor-mean(SampleData$pre_scor)  # Grand centers student p
  SampleData$gain <- SampleData$pst_scor - SampleData$pre_scor # calculates the gain
  SampleData$collabnla <- ifelse(SampleData$colablrn==1,ifelse(SampleData$used_las==0,1,0),0) # Creates
```

Calculating the descriptive statistics

```
#Make one categorical variable with all three types of instruction
  SampleData$instruction <- ifelse(SampleData$used_las==1,"Used_LAs",ifelse(SampleData$collabnla==1,"Co

# Make data frame of student means by instruction type (disaggrgation)
  student_means <- SampleData %>% group_by(instruction) %>% summarise(mean_gain = mean(gain))
  return <- student_means

#Make a data frame of course means by instruction type (aggregation)
  class_means <- SampleData %>% group_by(crse_id) %>% summarise(gain = mean(gain))
  class_means <- left_join(class_means,unique(SampleData[c(3,13)]), by = "crse_id") #need to replace th
  class_means <- class_means %>% group_by(instruction) %>% summarise(gain = mean(gain))
  return <- class_means
```

Define models (We ultimately used Model 3 as our simplest model that explained the most variance)

```
#HLM models
hlm_mod1 <- (gain ~ 1 + (1|crse_id))
hlm_mod2 <- (gain ~ 1 + used_las + collabnla + (1|crse_id))
hlm_mod3 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + (1|crse_id))
hlm_mod4 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + (1+ stud_pre_cent|crse_id))
hlm_mod5 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + class_pre_cent + (1|crse_id))
hlm_mod6 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + FMCE + (1|crse_id))

#MLR models
mlr_mod1 <- (gain ~ 1)
mlr_mod2 <- (gain ~ 1 + used_las + collabnla)
mlr_mod3 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla)
```

```r
mlr_mod4 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla)
mlr_mod5 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + class_pre_cent)
mlr_mod6 <- (gain ~ 1 + stud_pre_cent +  used_las + collabnla + FMCE)
```

Run models

```r
#HLM models
HLM1 <- lmer(hlm_mod1, data=SampleData)
HLM2 <- lmer(hlm_mod2, data=SampleData)
HLM3 <- lmer(hlm_mod3, data=SampleData)
HLM4 <- lmer(hlm_mod4, data=SampleData)
HLM5 <- lmer(hlm_mod5, data=SampleData)
HLM6 <- lmer(hlm_mod6, data=SampleData)

#MLR models
MLR1 <- lm(mlr_mod1, data=SampleData)
MLR2 <- lm(mlr_mod2, data=SampleData)
MLR3 <- lm(mlr_mod3, data=SampleData)
MLR4 <- lm(mlr_mod4, data=SampleData)
MLR5 <- lm(mlr_mod5, data=SampleData)
MLR6 <- lm(mlr_mod6, data=SampleData)
```

Model outputs

```r
#HLM models
summary(HLM1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: gain ~ 1 + (1 | crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 52992.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8389 -0.6228 -0.0285  0.6071  3.8036
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  crse_id  (Intercept)  63.01    7.938
##  Residual             411.31   20.281
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  18.4317     0.8332   22.12
```

```r
summary(HLM2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: gain ~ 1 + used_las + collabnla + (1 | crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 52976.9
##
## Scaled residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -4.8423 -0.6224 -0.0272  0.6050  3.8016
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  crse_id  (Intercept)  58.36    7.64
##  Residual             411.31   20.28
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   14.090      2.048   6.880
## used_las       6.080      2.279   2.668
## collabnla      1.997      2.755   0.725
##
## Correlation of Fixed Effects:
##           (Intr) usd_ls
## used_las  -0.899
## collabnla -0.743  0.668
```

```r
summary(HLM3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: gain ~ 1 + stud_pre_cent + used_las + collabnla + (1 | crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 51894.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4168 -0.6587 -0.0103  0.6496  3.7686
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  crse_id  (Intercept)  60.87    7.802
##  Residual             341.47   18.479
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   14.19542    2.05732   6.900
## stud_pre_cent -0.45137    0.01305 -34.595
## used_las       5.97094    2.28891   2.609
## collabnla      1.69127    2.75784   0.613
##
## Correlation of Fixed Effects:
##             (Intr) std_p_ usd_ls
## stud_pr_cnt  0.000
## used_las    -0.899  0.000
## collabnla   -0.746  0.000  0.671
```

```r
summary(HLM4)
```

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

```
## gain ~ 1 + stud_pre_cent + used_las + collabnla + (1 + stud_pre_cent |
##     crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 51861.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3551 -0.6516 -0.0026  0.6422  3.8390
##
## Random effects:
##  Groups   Name          Variance  Std.Dev. Corr
##  crse_id  (Intercept)    61.25363  7.8265
##           stud_pre_cent   0.01174  0.1083  -0.61
##  Residual               337.58933 18.3736
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   13.54778    1.98378   6.829
## stud_pre_cent -0.43741    0.01826 -23.950
## used_las       7.17291    2.18797   3.278
## collabnla      1.02653    2.63541   0.390
##
## Correlation of Fixed Effects:
##             (Intr) std_p_ usd_ls
## stud_pr_cnt -0.102
## used_las    -0.893 -0.045
## collabnla   -0.743 -0.014  0.676
## convergence code: 0
## Model is nearly unidentifiable: very large eigenvalue
##  - Rescale variables?
```

```r
summary(HLM5)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## gain ~ 1 + stud_pre_cent + used_las + collabnla + class_pre_cent +
##     (1 | crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 51897.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4171 -0.6588 -0.0109  0.6487  3.7768
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  crse_id  (Intercept)  61.32    7.831
##  Residual             341.49   18.479
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##                Estimate Std. Error t value
```

```
## (Intercept)    14.37811    2.16418   6.644
## stud_pre_cent  -0.45137    0.01305 -34.594
## used_las        5.78287    2.39202   2.418
## collabnla       1.42664    2.91740   0.489
## class_pre_cent  0.02489    0.09034   0.276
##
## Correlation of Fixed Effects:
##             (Intr) std_p_ usd_ls cllbnl
## stud_pr_cnt  0.000
## used_las    -0.907  0.000
## collabnla   -0.770  0.000  0.699
## clss_pr_cnt  0.301  0.000 -0.280 -0.318
```

```r
summary(HLM6)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: gain ~ 1 + stud_pre_cent + used_las + collabnla + FMCE + (1 |
##     crse_id)
##    Data: SampleData
##
## REML criterion at convergence: 51891.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4172 -0.6589 -0.0097  0.6495  3.7714
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  crse_id  (Intercept)  61.59    7.848
##  Residual             341.47   18.479
## Number of obs: 5959, groups:  crse_id, 112
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)   14.18570    2.07169   6.847
## stud_pre_cent -0.45137    0.01305 -34.595
## used_las       5.92076    2.33039   2.541
## collabnla      1.66011    2.77476   0.598
## FMCE           0.25675    2.13447   0.120
##
## Correlation of Fixed Effects:
##             (Intr) std_p_ usd_ls cllbnl
## stud_pr_cnt  0.000
## used_las    -0.876  0.000
## collabnla   -0.741  0.000  0.669
## FMCE        -0.060  0.000 -0.159 -0.048
```

```r
#MLR models
summary(MLR1)
```

```
##
## Call:
## lm(formula = mlr_mod1, data = SampleData)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -96.028 -14.892  -1.801  13.901  76.291
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  19.4323     0.2806   69.24   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.66 on 5958 degrees of freedom
```

```r
summary(MLR2)
```

```
##
## Call:
## lm(formula = mlr_mod2, data = SampleData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -96.143 -14.243  -1.057  13.786  78.964
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  12.1419     0.7609   15.96   <2e-16 ***
## used_las      7.4055     0.8311    8.91   <2e-16 ***
## collabnla    12.2481     1.0039   12.20   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.4 on 5956 degrees of freedom
## Multiple R-squared:  0.02444,    Adjusted R-squared:  0.02411
## F-statistic: 74.61 on 2 and 5956 DF,  p-value: < 2.2e-16
```

```r
summary(MLR3)
```

```
##
## Call:
## lm(formula = mlr_mod3, data = SampleData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -79.541 -13.766  -0.706  13.506  65.319
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   12.14188    0.70168  17.304   <2e-16 ***
## stud_pre_cent -0.45137    0.01393 -32.394   <2e-16 ***
## used_las       7.40550    0.76639   9.663   <2e-16 ***
## collabnla     12.24810    0.92576  13.230   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.73 on 5955 degrees of freedom
## Multiple R-squared:  0.1706, Adjusted R-squared:  0.1702
## F-statistic: 408.3 on 3 and 5955 DF,  p-value: < 2.2e-16
```

```r
summary(MLR4)
```

```
##
## Call:
## lm(formula = mlr_mod4, data = SampleData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -79.541 -13.766  -0.706  13.506  65.319
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  12.14188    0.70168  17.304   <2e-16 ***
## stud_pre_cent -0.45137    0.01393 -32.394   <2e-16 ***
## used_las      7.40550    0.76639   9.663   <2e-16 ***
## collabnla    12.24810    0.92576  13.230   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.73 on 5955 degrees of freedom
## Multiple R-squared:  0.1706, Adjusted R-squared:  0.1702
## F-statistic: 408.3 on 3 and 5955 DF,  p-value: < 2.2e-16
```

```r
summary(MLR5)
```

```
##
## Call:
## lm(formula = mlr_mod5, data = SampleData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -80.173 -13.674  -0.841  13.577  66.268
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    12.62950    0.73975  17.073   <2e-16 ***
## stud_pre_cent  -0.45137    0.01393 -32.403   <2e-16 ***
## used_las        6.95050    0.79689   8.722   <2e-16 ***
## collabnla      11.51532    0.99049  11.626   <2e-16 ***
## class_pre_cent  0.06258    0.03014   2.077   0.0379 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.73 on 5954 degrees of freedom
## Multiple R-squared:  0.1712, Adjusted R-squared:  0.1706
## F-statistic: 307.5 on 4 and 5954 DF,  p-value: < 2.2e-16
```

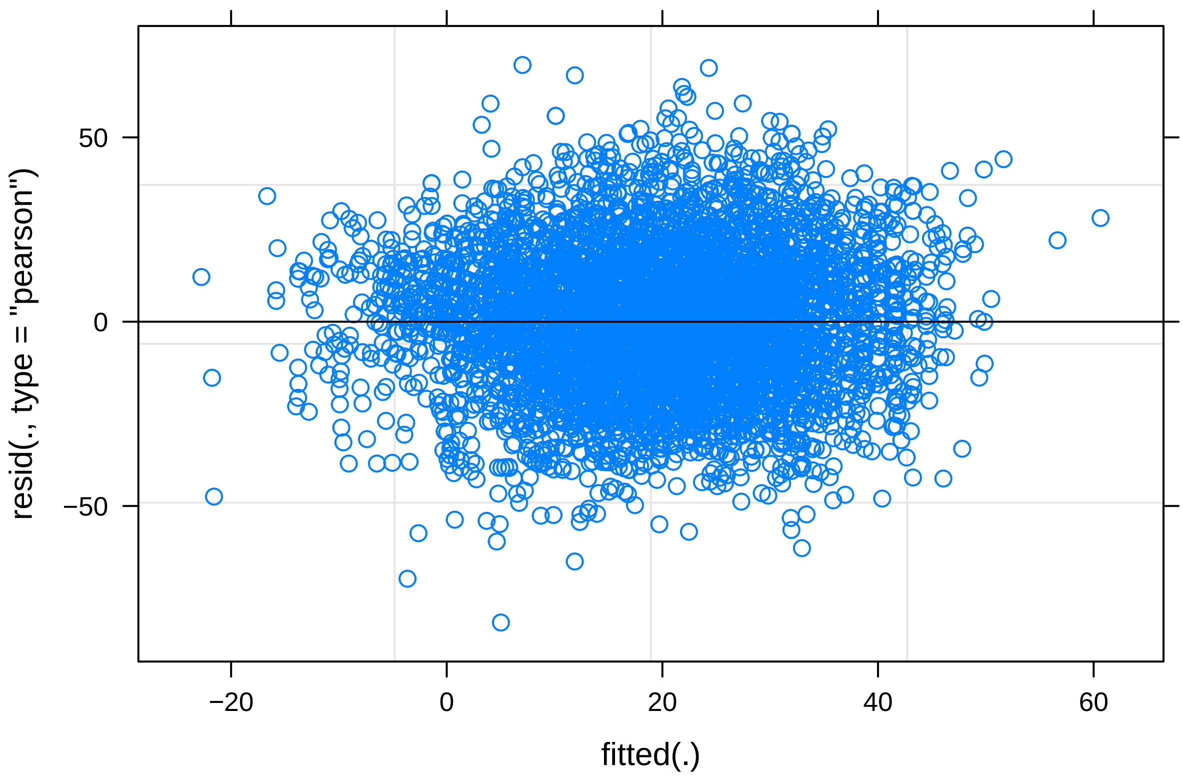
```r
summary(MLR6)
```

```
##
## Call:
## lm(formula = mlr_mod6, data = SampleData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -79.087 -13.721  -0.768  13.573  65.773
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   12.17253    0.70200  17.340   <2e-16 ***
## stud_pre_cent -0.45137    0.01393 -32.396   <2e-16 ***
## used_las       7.72885    0.80275   9.628   <2e-16 ***
## collabnla     12.25982    0.92573  13.243   <2e-16 ***
## FMCE          -0.80815    0.59738  -1.353    0.176
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.73 on 5954 degrees of freedom
## Multiple R-squared:  0.1709, Adjusted R-squared:  0.1703
## F-statistic: 306.7 on 4 and 5954 DF,  p-value: < 2.2e-16
```

Assumption checking

```r
#linearity: Shouldn't see a pattern
plot(HLM3)
```



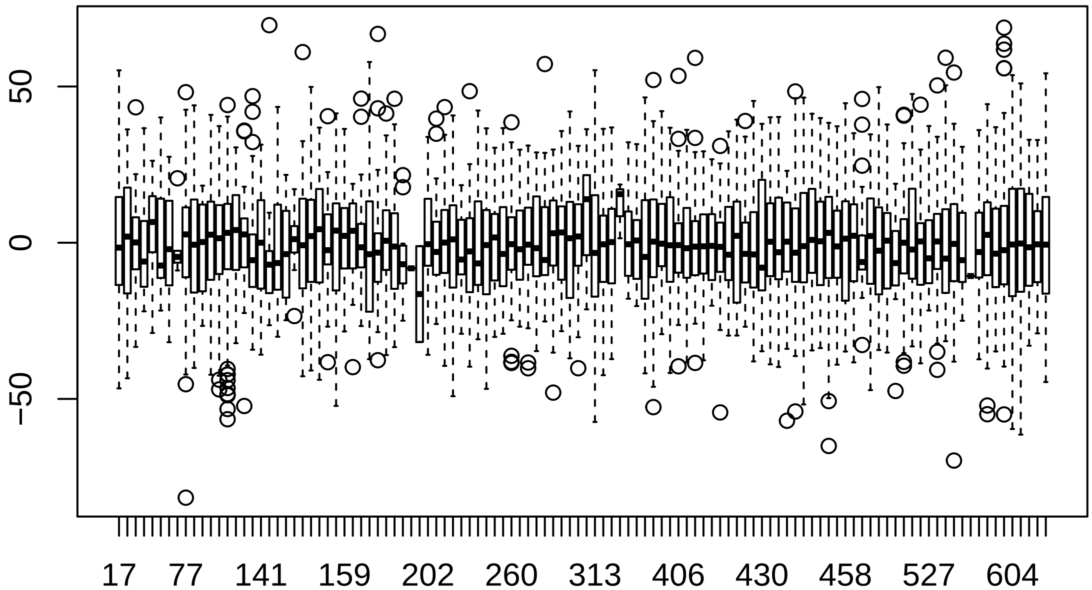

```r
#quantitative homogeneity of variance
SampleData$Model.F.Res<- residuals(HLM3) #extracts the residuals and places them in a new column in our
SampleData$Abs.Model.F.Res <-abs(SampleData$Model.F.Res) #creates a new column with the absolute value

Levene.Model.F <- lm(Model.F.Res ~ crse_id, data=SampleData) #ANOVA of the residuals
anova(Levene.Model.F) #displays the results: want a p>0.05
```

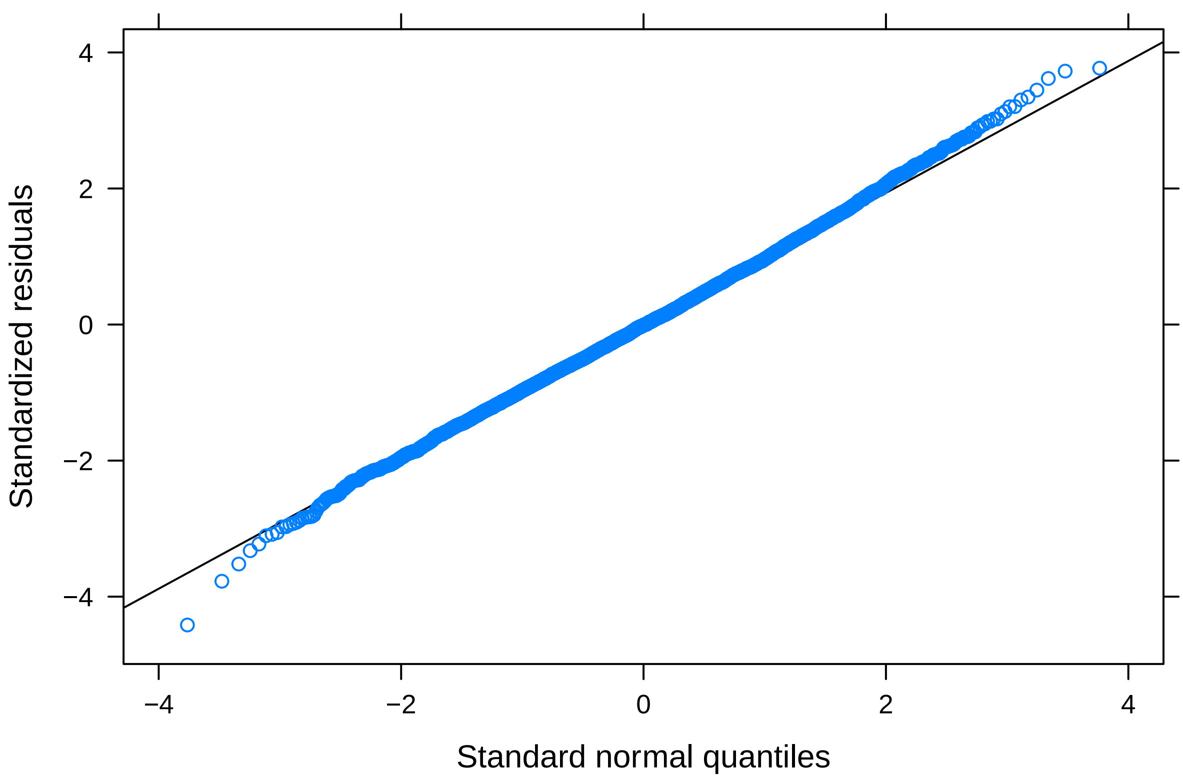
```
## Analysis of Variance Table
##
## Response: Model.F.Res
```

```
##                Df  Sum Sq Mean Sq F value Pr(>F)
## crse_id        1      47   46.98  0.1397 0.7085
## Residuals 5957 2002574  336.17
```

```
#visual homogeneity of variance
boxplot(SampleData$Model.F.Res ~ SampleData$crse_id)
```




```
#Assumption of Normality or residuals: want points to be near the line
qqmath(HLM3)
```



Creating groups for final model

```r
Trad = c(1,0,0,0)
Collab = c(1,0,0,1)
LA = c(1,0,1,0)


HLM_preds <- rbind( 'Lecture'=Trad, 'Collaborative'=Collab,
                          'LAs'=LA)
```

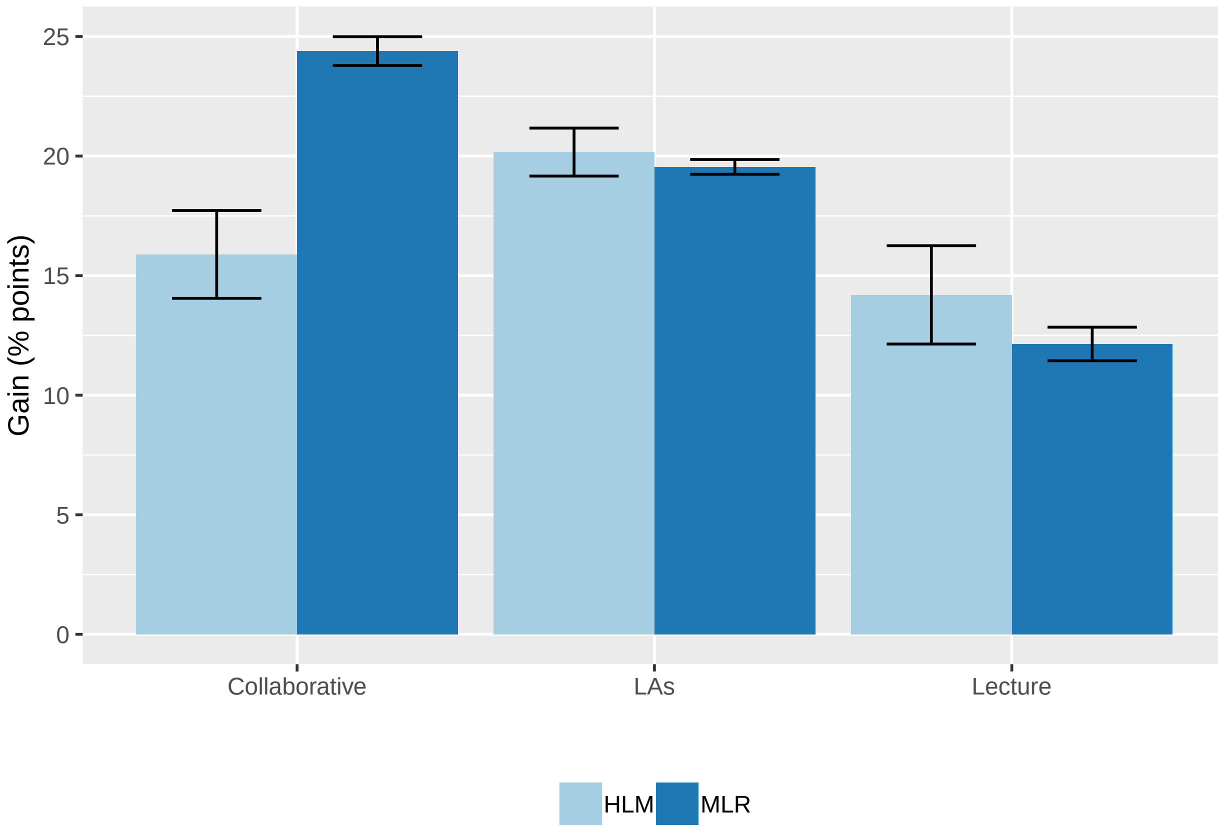
```r
# getting summary statistics from HLM model for plot
  sxp3 <- summary(glht(HLM3, linfct=HLM_preds)) #getting the summary from the HLM models
  get.est<- data.frame(analysis = c("HLM","HLM","HLM"), #simplifying that summary for the plots
              group=rownames(sxp3$linfct),
             coeff = sxp3$test$coefficients,
             se = sxp3$test$sigma)


# getting summary statistics from MLR model for plot
  sxp3 <- summary(glht(MLR3, linfct=HLM_preds))
  temp<- data.frame(analysis = c("MLR","MLR","MLR"),
                   group=rownames(sxp3$linfct),
             coeff = sxp3$test$coefficients,
             se = sxp3$test$sigma)

#combine MLR and HLM summaries for plot
  get.est <- bind_rows(get.est,temp)
```

Graph of Model 3 predicted values with error bars representing 1 standard error

```r
ggplot(get.est, aes(y=coeff, fill=analysis, x=group )) +
  geom_bar(stat="identity", position = position_dodge(width=0.9)) +
  geom_errorbar(aes(ymax=coeff+se, ymin=coeff-se), position=position_dodge(0.9), width=0.5)  +
  scale_fill_brewer(palette="Paired")+
  ylab("Gain (% points)") +
  xlab("") +
  theme(legend.position = "bottom", legend.direction = "horizontal", legend.title = element_blank())
```


Gain (% points)
25
20
15
10
5
0
Collaborative
LAs
Lecture
HLM
MLR