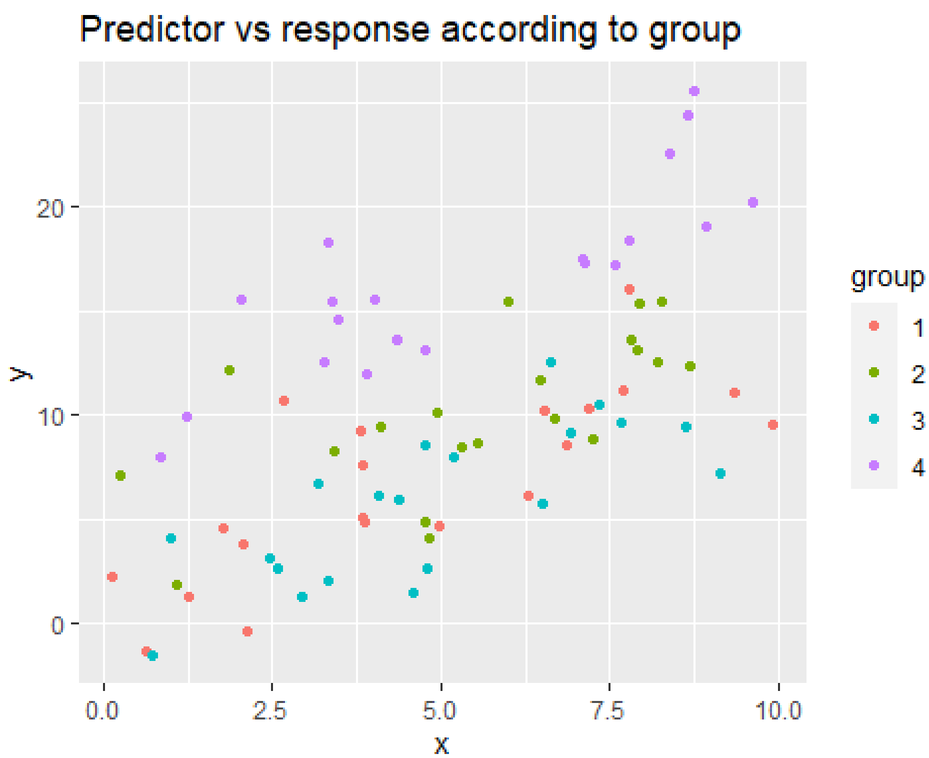
# HW4_AustinBrewer_4520

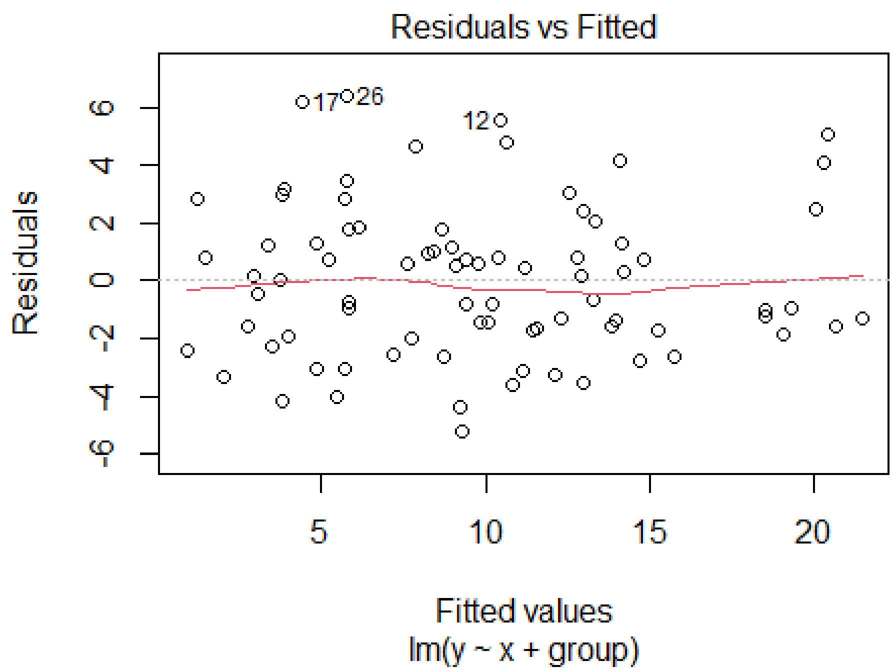
Austin Brewer

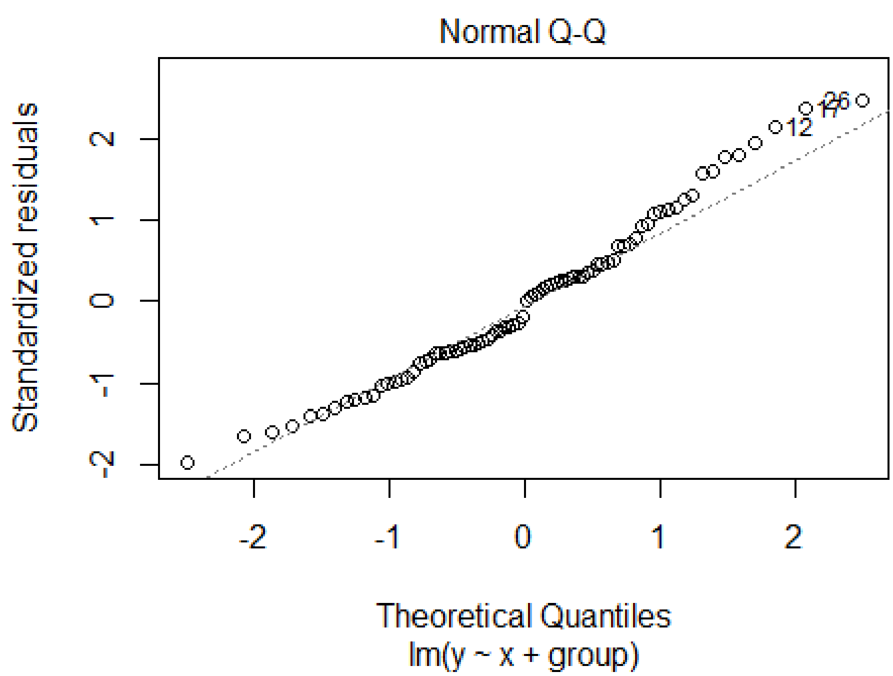
10/30/2023

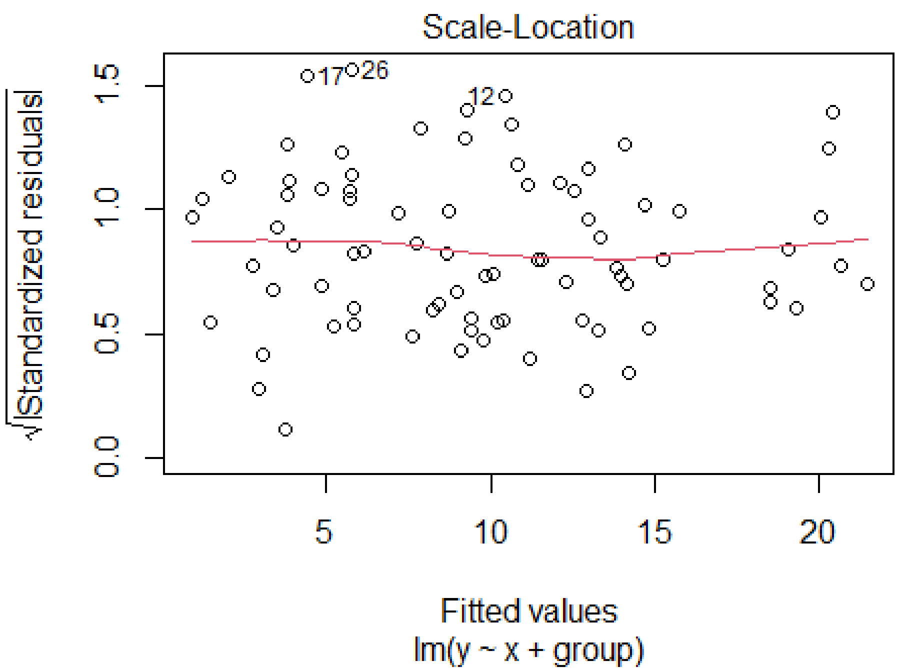
```r
#1a
ggplot(dat, aes(x=x, y=y, color=group))+
  geom_point()+
  labs(title="Predictor vs response according to group")
```

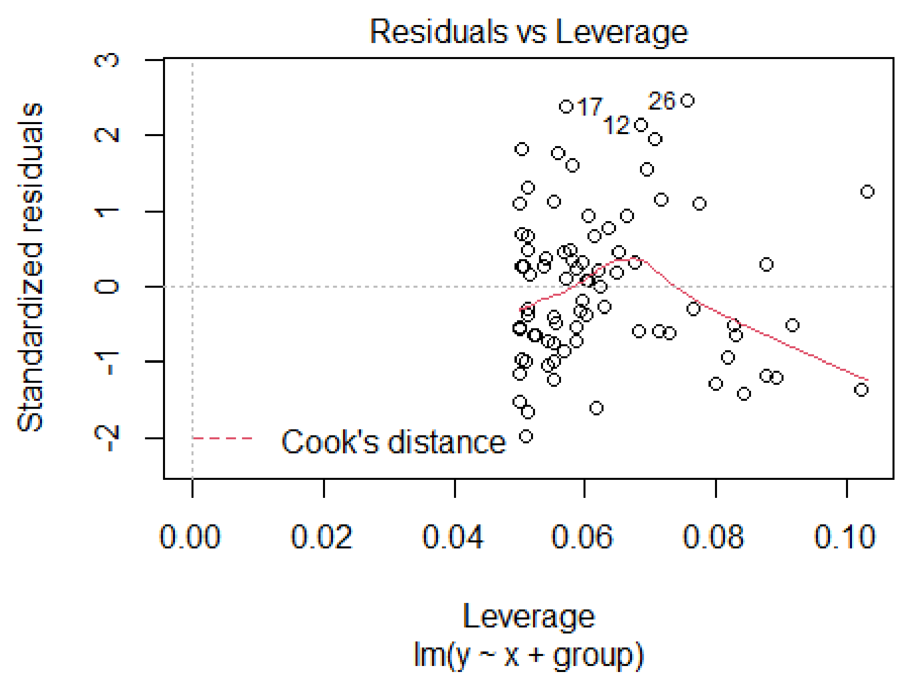



```r
#1b
op = options(contrasts = c("contr.sum","contr.poly"))
lmod = lm(y ~ x+group, data=dat)
#summary(lmod)
#Coefficients are as follows: intercept=3.7691, x=1.1804, group1=-2.4687,
group2=-0.1834, group3=-3.716. All are significant except for group 2.
plot(lmod)
```


Residuals vs Fitted
Residuals
Fitted values
lm(y ~ x + group)
17
26
12
6
4
2
0
-2
-4
-6
5
10
15
20


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(y ~ x + group)
26
17
12
2
1
0
-1
-2
-2
-1
0
1
2

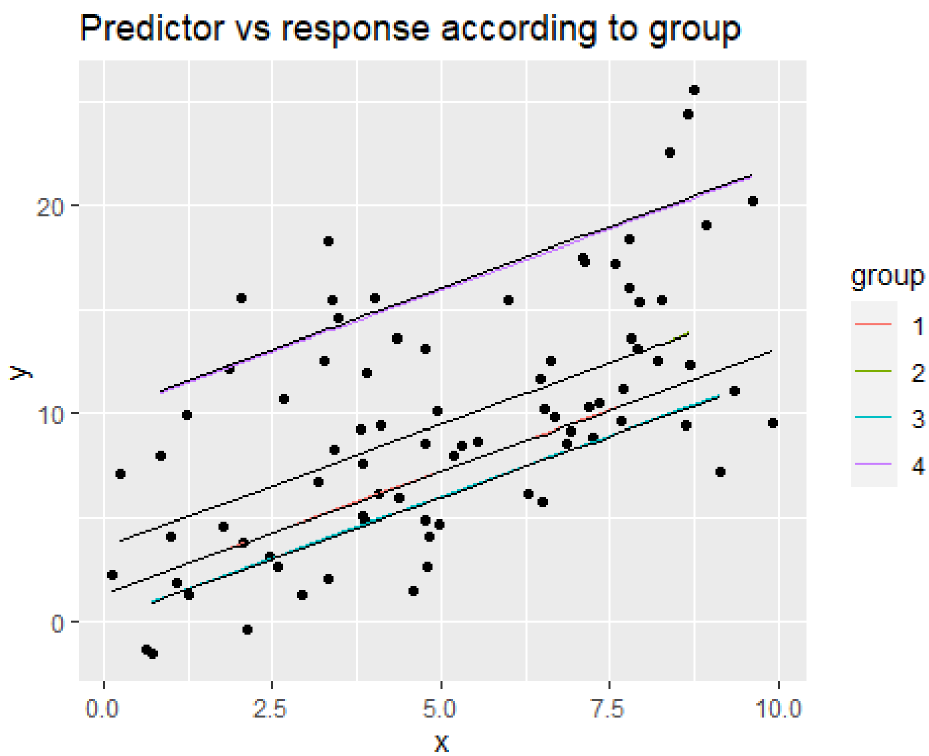



```
#There don't appear to be major issues with model specifications. The only
issues may be some high leverage points but they aren't extreme.
```

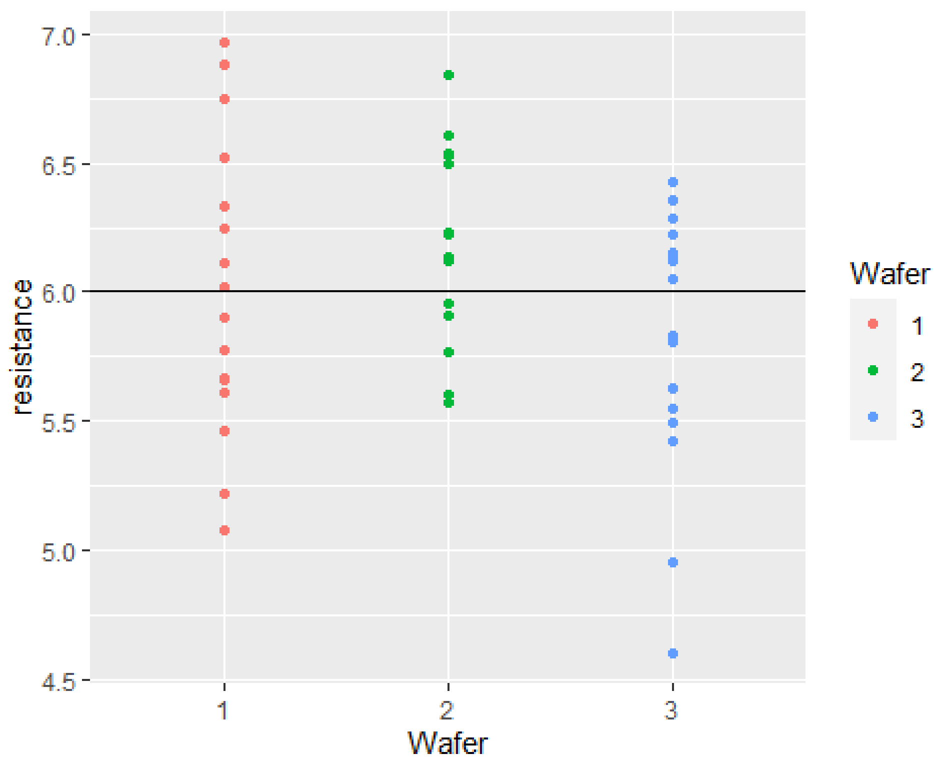
```r
#1c
mixmod = lmer(y ~ 1+x+(1|group), data=dat)
AIC(lmod)

## [1] 392.836

sumary(mixmod)

## Fixed Effects:
##             coef.est coef.se
## (Intercept) 3.75     2.32
## x           1.18     0.12
##
## Random Effects:
##  Groups   Name        Std.Dev.
##  group    (Intercept) 4.45
##  Residual             2.70
## ---
## number of obs: 80, groups: group, 4
## AIC = 407.1, DIC = 400.7
## deviance = 399.9

ranef(mixmod)$group

##   (Intercept)
## 1  -2.4226720
## 2  -0.1812754
## 3  -3.6480247
## 4   6.2519721

#Overall, the two models are very similar. In comparing the fits of the two
models, I looked at AIC for both. The AIC for the fixed model was slightly
lower at 392.8 than the mixed model which was at 407.1. When looking at the
predicted random effects, they are very similar to the predicted fixed
effects (all of them are slightly smaller than their counterpart). The
predicted fixed and random effects are also very close to the true effect
(alpha).

#1d
nullmod = lmer(y ~ 1+(1|group), data=dat)
KRmodcomp(mixmod,nullmod)

## large : y ~ 1 + x + (1 | group)
## small : y ~ 1 + (1 | group)
##          stat     ndf     ddf F.scaling   p.value
## Ftest 103.103   1.000  75.062         1 9.736e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#According to the test, the predictor x is a significant inclusion to the
model.

#1e
exactRLRT(mixmod)

##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 75.377, p-value < 2.2e-16

#According to the test, the variance of the random effect isn't zero and
therefore the random effect is significant.

#1f
dat2 = cbind(dat, mixmod=predict(mixmod), lmod=predict(lmod))
ggplot(dat2, aes(x=x, y=y))+
  geom_point()+
  geom_line(aes(x,mixmod, color=group))+
  geom_line(aes(x,lmod, group=group))+
  labs(title="Predictor vs response according to group")
```




```
#One major observation is that the two models perform almost identically. I
colored the mixed model groups and left the fixed model groups as black. The
lines follow identical slopes with very minor intercept differences. This
```

```r
leads into another major takeaway, which is that all groups have identical
slopes. This indicates the slope is measured by pooling the sample together,
while the intercepts are capturing all the variation between the groups.

#2a
ggplot(semicond, aes(x=Wafer, y=resistance, color=Wafer))+
  geom_point()+
  geom_hline(yintercept = (mean(semicond$resistance)))
```




```r
#It seems that the mean for wafer 2 is situated above the overall mean while
the wafer 3 mean is situated slightly below. Wafer 2 looks to have the lowest
variance, while wafer 1 and 3 appear to have a similar variance. With that
being said, the resistance for wafer 1 is evenly distributed, while wafer 3's
data is inconsistent and may contain a few high leverage points.

#2b
fixmod = lm(resistance ~ 1+ET+position+(ET*position), data=semicond)
#summary(fixmod)
#The only significant predictors in the model are the intercept, ET=1, and
position=3 The intercept is set at the mean of 6.0029.

#2c
mmod = lmer(resistance ~ 1+ET+position+(ET*position)+(1|Grp), data=semicond)
#sumary(mmod)
#The standard deviation of the random effect (Grp) in this model is 0.33.
```

```r
#2d
intmod = lmer(resistance ~ 1+position+ET+(1|Grp), data=semicond)
KRmodcomp(mmod,intmod)
```

```
## large : resistance ~ 1 + ET + position + (ET * position) + (1 | Grp)
## small : resistance ~ 1 + position + ET + (1 | Grp)
##          stat     ndf     ddf F.scaling p.value
## Ftest  0.8092  9.0000 24.0000         1  0.6125
```

```r
#P-val of 0.6, so the interaction can be removed.
ETmod = lmer(resistance ~ 1+position+(1|Grp), data=semicond)
KRmodcomp(intmod,ETmod)
```

```
## large : resistance ~ 1 + position + ET + (1 | Grp)
## small : resistance ~ 1 + position + (1 | Grp)
##         stat    ndf    ddf F.scaling p.value
## Ftest 1.9415 3.0000 8.0000         1  0.2015
```

```r
#P-val of 0.2015, so ET can be removed.
Posmod = lmer(resistance ~ 1+(1|Grp), data=semicond)
KRmodcomp(ETmod, Posmod)
```

```
## large : resistance ~ 1 + position + (1 | Grp)
## small : resistance ~ 1 + (1 | Grp)
##          stat     ndf     ddf F.scaling p.value
## Ftest  3.5714  3.0000 33.0000         1 0.02427 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#P-val of 0.0243, so position should be left in the model.

#2e
exactRLRT(ETmod)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 17.095, p-value < 2.2e-16
```

```r
#With a P-val of 0, the random effect should be left in the model. That means
the final model includes the intercept, position, and Grp (special
interaction between ET and Wafer).
```