HW 8

Ted Henson

4/3/2020

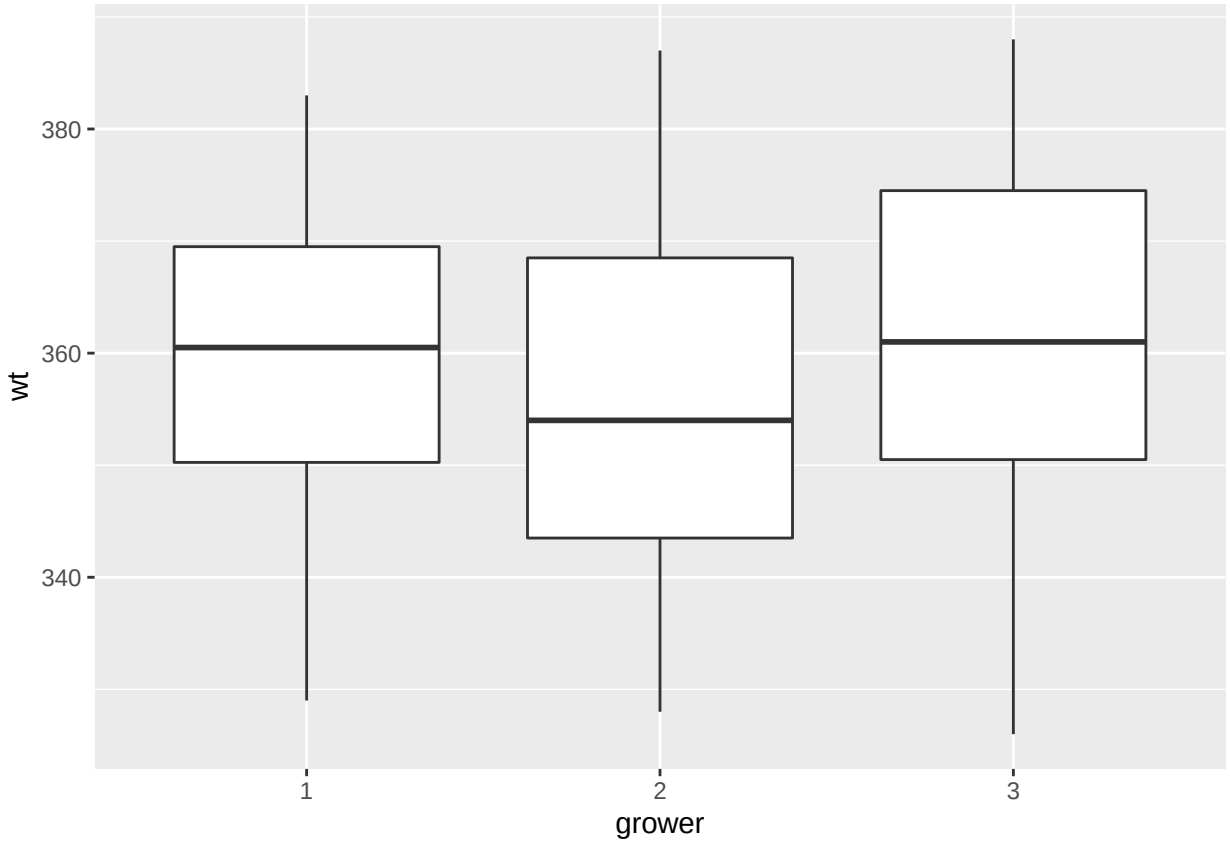
Question 5

a)

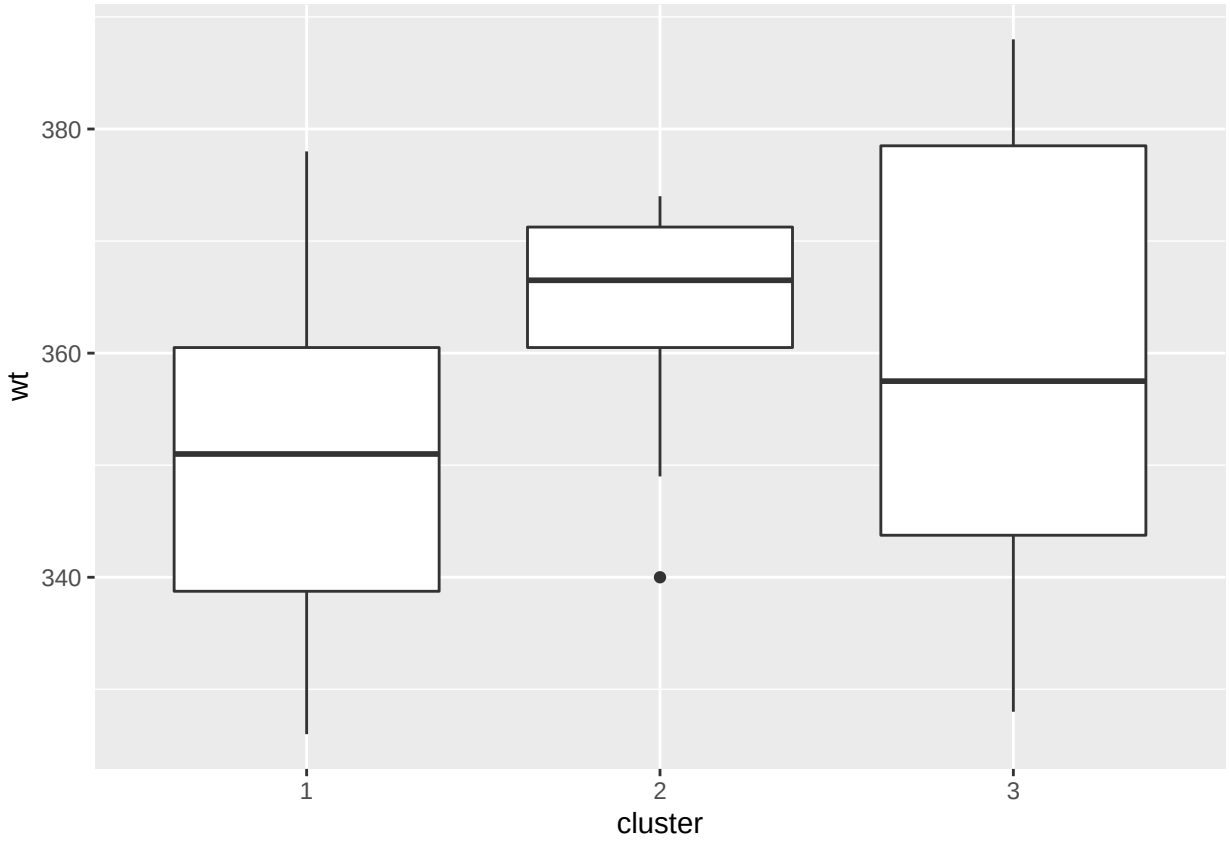
```
library(faraway)
data("broccoli")
head(broccoli)
```

```
##    wt grower box cluster
## 1 352      1   1       1
## 2 369      1   1       2
## 3 383      1   1       3
## 4 339      2   1       1
## 5 367      2   1       2
## 6 328      2   1       3
```

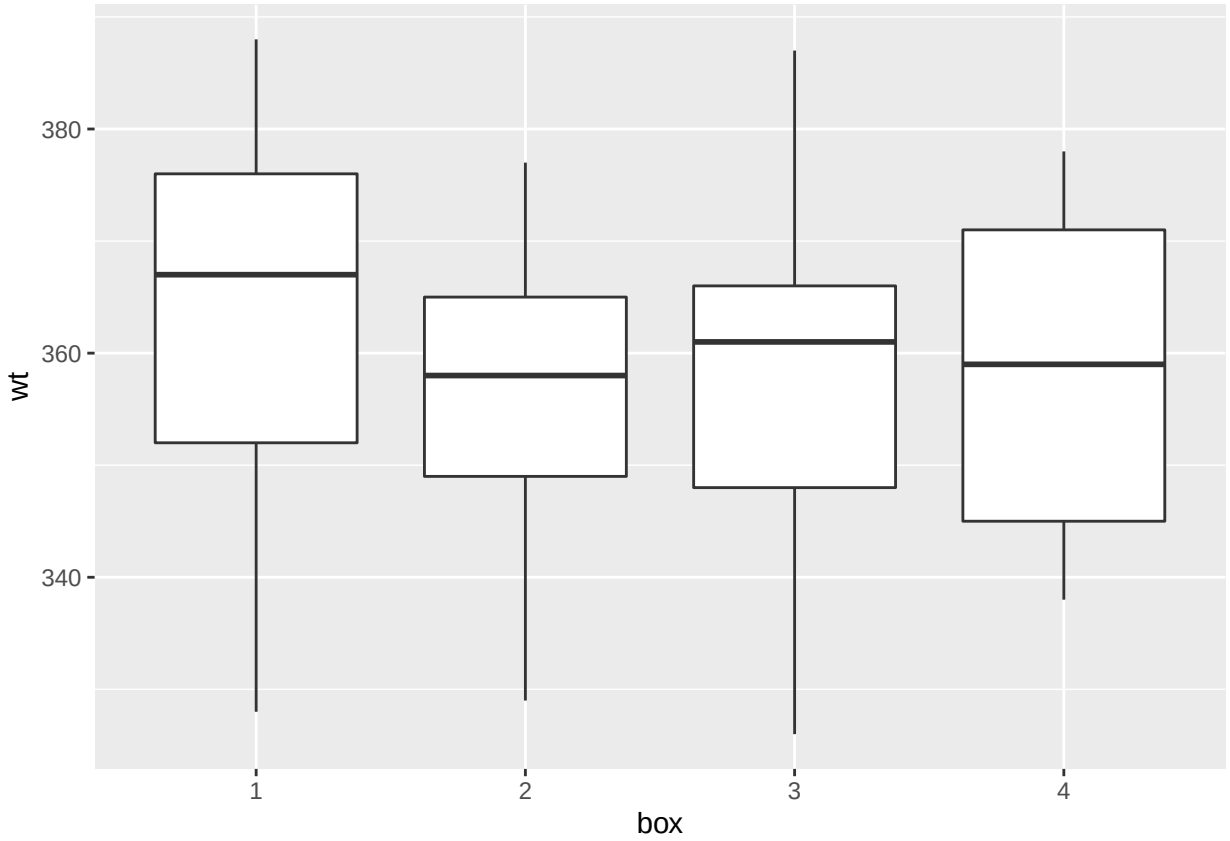
```
library(pbkrtest)
library(ggplot2)
library(lme4)

ggplot(broccoli, aes(x=grower, y = wt)) + geom_boxplot()
```



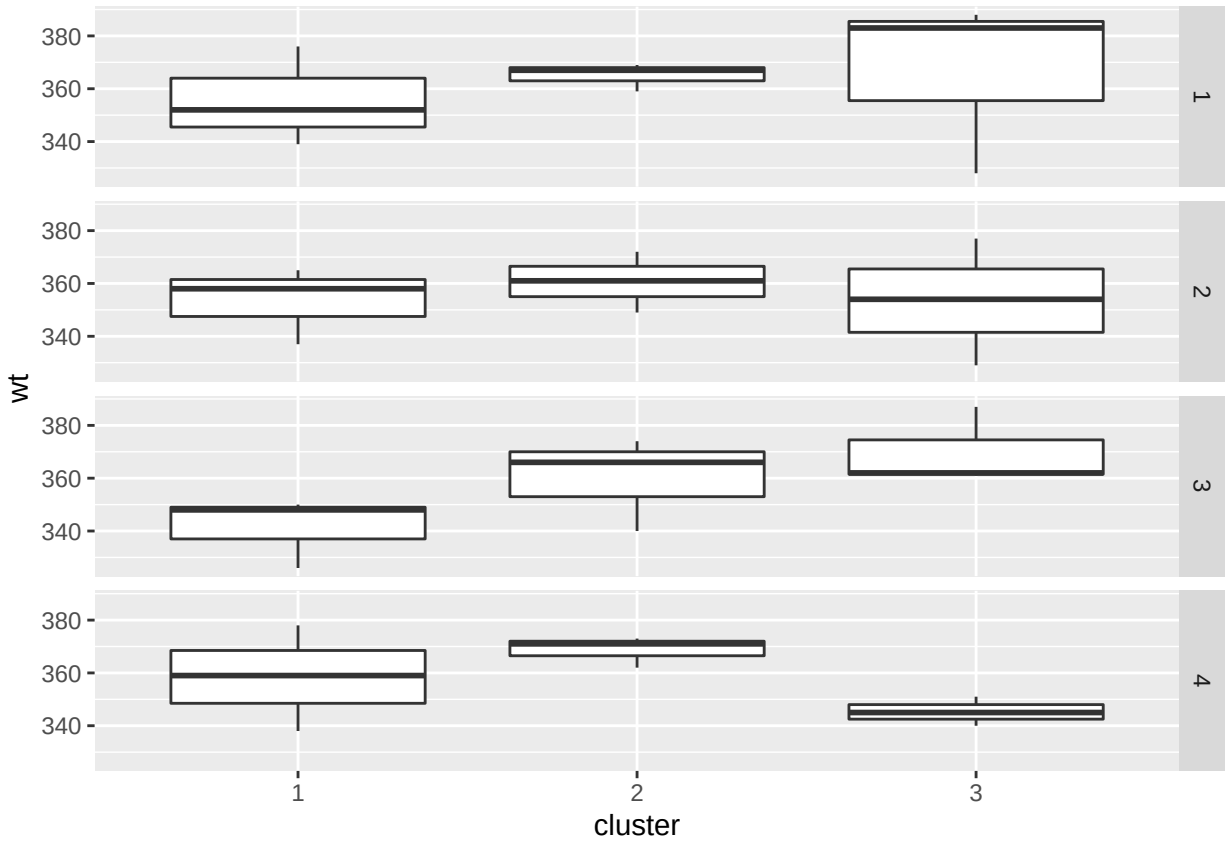

```
ggplot(broccoli, aes(x=cluster, y = wt)) + geom_boxplot()
```



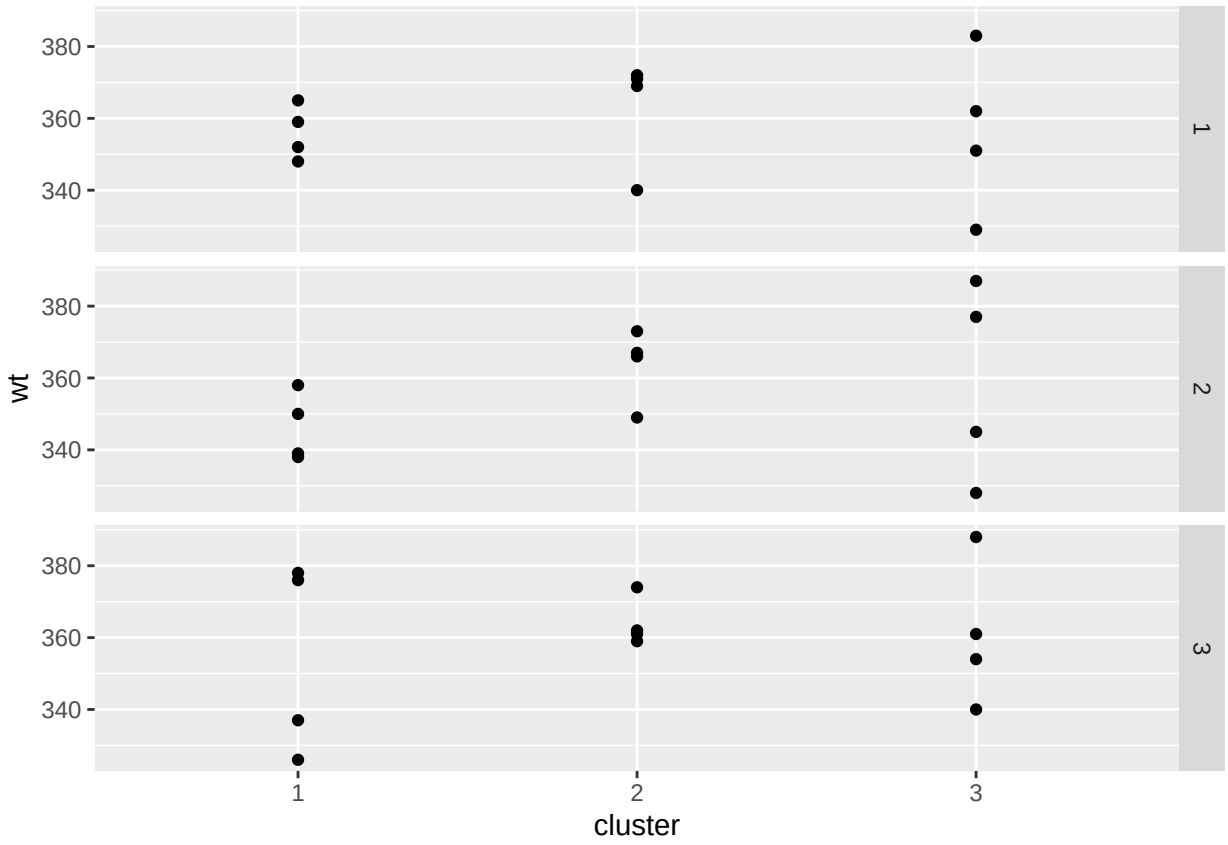

```
ggplot(broccoli, aes(x=box, y = wt)) + geom_boxplot()
```



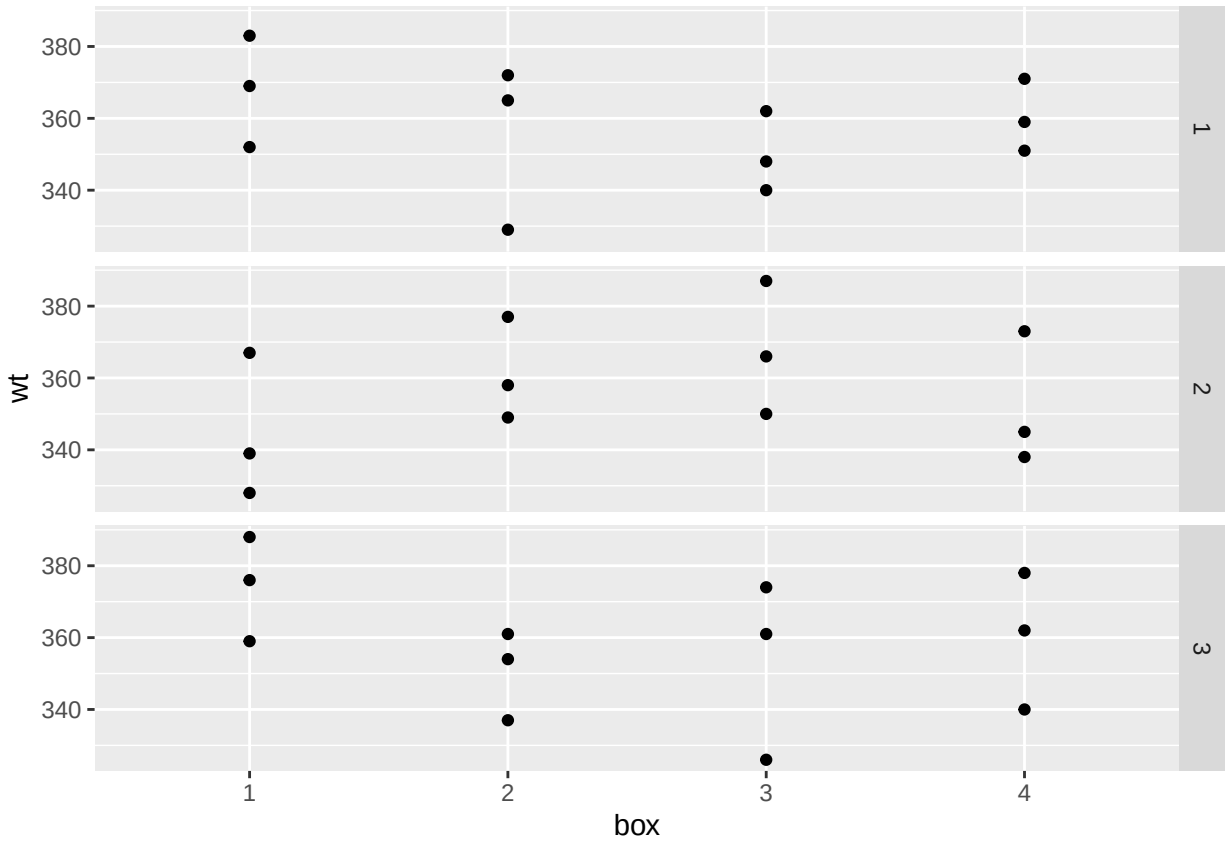

```
ggplot(broccoli, aes(x=cluster, y = wt)) + geom_boxplot() + facet_grid(box ~ .)
```



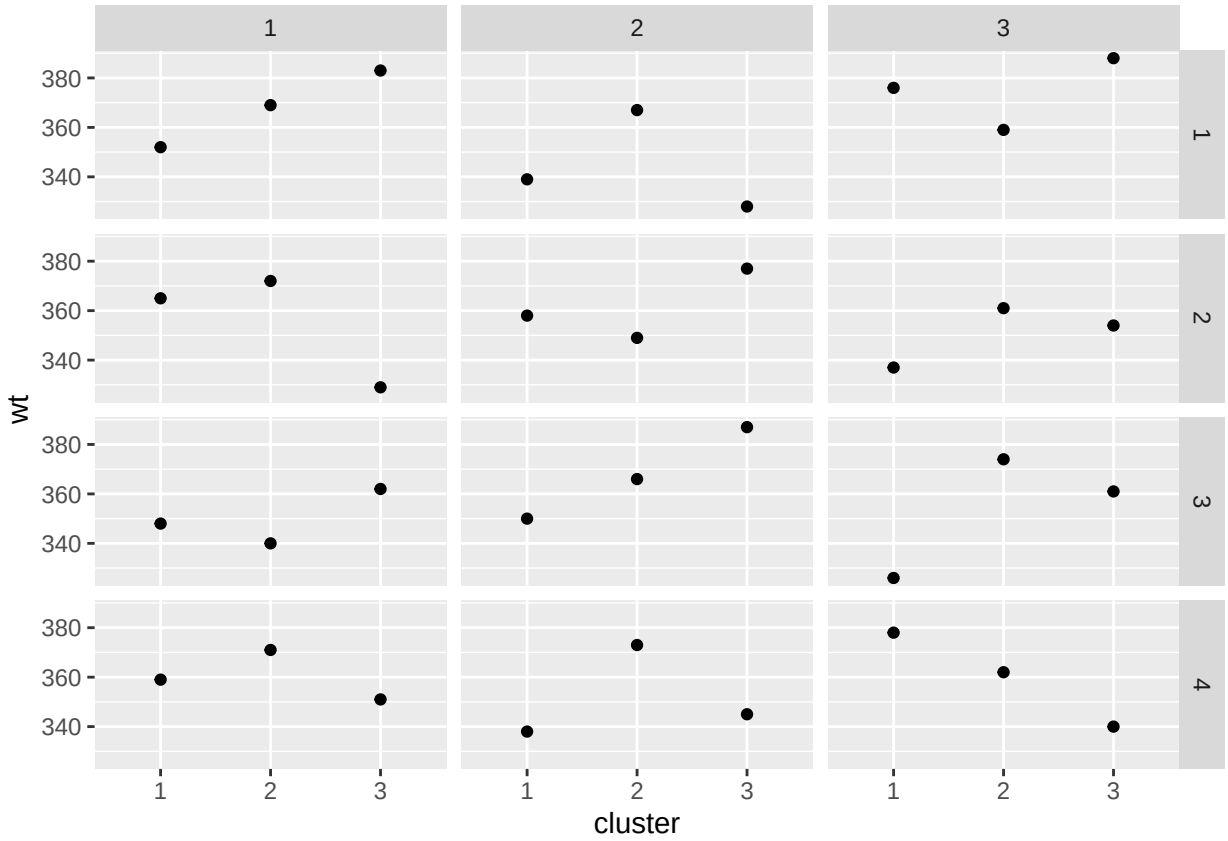

```
ggplot(broccoli, aes(x=cluster, y = wt)) + geom_point() + facet_grid(grower ~ .)
```




```
ggplot(broccoli, aes(x=box, y = wt)) + geom_point() + facet_grid(grower ~ .)
```




```
ggplot(broccoli, aes(x=cluster, y = wt)) + geom_point() + facet_grid( box ~ grower)
```



Plots of the data do not show any obivious trends regarding the mean weight whether from viewing each variable separately, all together, or in pairs, but there may be some random variation within groups, specifically within clusters.

b)

```
library(tidyverse)

box.means = broccoli %>% group_by(box) %>% summarise(weight = mean(wt))
box.means
```

```
## # A tibble: 4 x 2
##   box   weight
##   <fct>  <dbl>
## 1 1       362.
## 2 2       356.
## 3 3       357.
## 4 4       357.
```

```
grower.means = broccoli %>% group_by(grower) %>% summarise(weight = mean(wt))
grower.means
```

```
## # A tibble: 3 x 2
##   grower weight
##   <fct>   <dbl>
## 1 1        358.
## 2 2        356.
```

```
## 3 3        360.
```

```
cluster.means = broccoli %>% group_by(cluster) %>% summarise(weight = mean(wt))
cluster.means
```

```
## # A tibble: 3 x 2
##   cluster weight
##   <fct>    <dbl>
## 1 1         352.
## 2 2         364.
## 3 3         359.
```

c)

```
fmod=lmer(wt~ box*grower + (1|cluster),broccoli)
summary(fmod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: wt ~ box * grower + (1 | cluster)
##    Data: broccoli
##
## REML criterion at convergence: 218.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.54158 -0.71271 -0.00544  0.71433  1.22094
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  cluster  (Intercept)   8.253   2.873
##  Residual             295.053  17.177
## Number of obs: 36, groups:  cluster, 3
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   368.000     10.055  36.599
## box2          -12.667     14.025  -0.903
## box3          -18.000     14.025  -1.283
## box4           -7.667     14.025  -0.547
## grower2       -23.333     14.025  -1.664
## grower3         6.333     14.025   0.452
## box2:grower2   29.333     19.834   1.479
## box3:grower2   41.000     19.834   2.067
## box4:grower2   15.000     19.834   0.756
## box2:grower3  -11.000     19.834  -0.555
## box3:grower3   -2.667     19.834  -0.134
## box4:grower3   -6.667     19.834  -0.336
##
## Correlation of Fixed Effects:
##             (Intr) box2   box3   box4   growr2 growr3 bx2:g2 bx3:g2 bx4:g2
## box2        -0.697
## box3        -0.697  0.500
## box4        -0.697  0.500  0.500
## grower2     -0.697  0.500  0.500  0.500
## grower3     -0.697  0.500  0.500  0.500  0.500
```

```
## box2:growr2  0.493 -0.707 -0.354 -0.354 -0.707 -0.354
## box3:growr2  0.493 -0.354 -0.707 -0.354 -0.707 -0.354  0.500
## box4:growr2  0.493 -0.354 -0.354 -0.707 -0.707 -0.354  0.500  0.500
## box2:growr3  0.493 -0.707 -0.354 -0.354 -0.354 -0.707  0.500  0.250  0.250
## box3:growr3  0.493 -0.354 -0.707 -0.354 -0.354 -0.707  0.250  0.500  0.250
## box4:growr3  0.493 -0.354 -0.354 -0.707 -0.354 -0.707  0.250  0.250  0.500
##             bx2:g3 bx3:g3
## box2
## box3
## box4
## grower2
## grower3
## box2:growr2
## box3:growr2
## box4:growr2
## box2:growr3
## box3:growr3  0.500
## box4:growr3  0.500  0.500
```

```
fixef(fmod)
```

```
##  (Intercept)         box2         box3         box4      grower2      grower3
##   368.000000   -12.666667   -18.000000    -7.666667   -23.333333     6.333333
## box2:grower2 box3:grower2 box4:grower2 box2:grower3 box3:grower3 box4:grower3
##    29.333333    41.000000    15.000000   -11.000000    -2.666667    -6.666667
```

The model assigned no random effects to the clusters with a low variance and standard deviation between clusters, but a high variance and standard deviation within clusters. It assigned fixed effects to the boxes and growers, but they had fairly high standard errors.

d)

```
mmod=lmer(wt~ grower + (1|cluster),broccoli)
nmod=lmer(wt~ 1 + (1|cluster),broccoli)
KRmodcomp(mmod,nmod)
```

```
## F-test with Kenward-Roger approximation; computing time: 0.11 sec.
## large : wt ~ grower + (1 | cluster)
## small : wt ~ 1 + (1 | cluster)
##          stat     ndf     ddf F.scaling p.value
## Ftest  0.1118  2.0000 31.0000         1  0.8946
```

The Kenward Rodgers test shows that no variation can be attributed to the growers.

e)

```
mmod=lmer(wt~ box + (1|cluster),broccoli)
nmod=lmer(wt~ 1 + (1|cluster),broccoli)
KRmodcomp(mmod,nmod)
```

```
## F-test with Kenward-Roger approximation; computing time: 0.04 sec.
## large : wt ~ box + (1 | cluster)
## small : wt ~ 1 + (1 | cluster)
##          stat     ndf     ddf F.scaling p.value
## Ftest  0.2532  3.0000 30.0000         1  0.8585
```

The test also confirms no variation can be attributed to the boxes.

f)

```
confint(fmod)
```

```
##                    2.5 %      97.5 %
## .sig01          0.000000  13.2110782
## .sigma         11.218064  18.1818049
## (Intercept)  351.470037 384.5302824
## box2         -35.780333  10.4469974
## box3         -41.113666   5.1136640
## box4         -30.780333  15.4469974
## grower2      -46.447000  -0.2196693
## grower3      -16.780333  29.4469974
## box2:grower2  -3.354327  62.0209905
## box3:grower2   8.312340  73.6876572
## box4:grower2 -17.687660  47.6876572
## box2:grower3 -43.687660  21.6876572
## box3:grower3 -35.354327  30.0209905
## box4:grower3 -39.354327  26.0209905
```

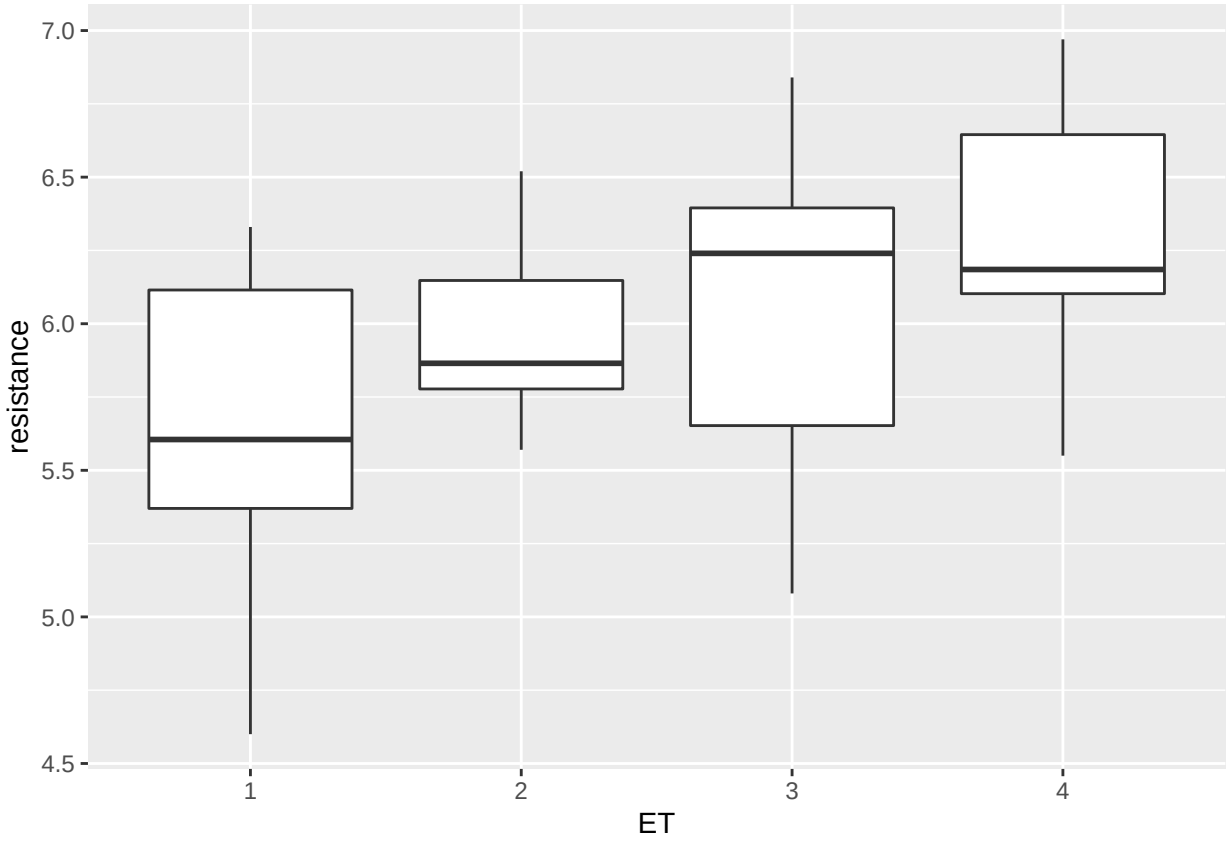
Question 7

a)

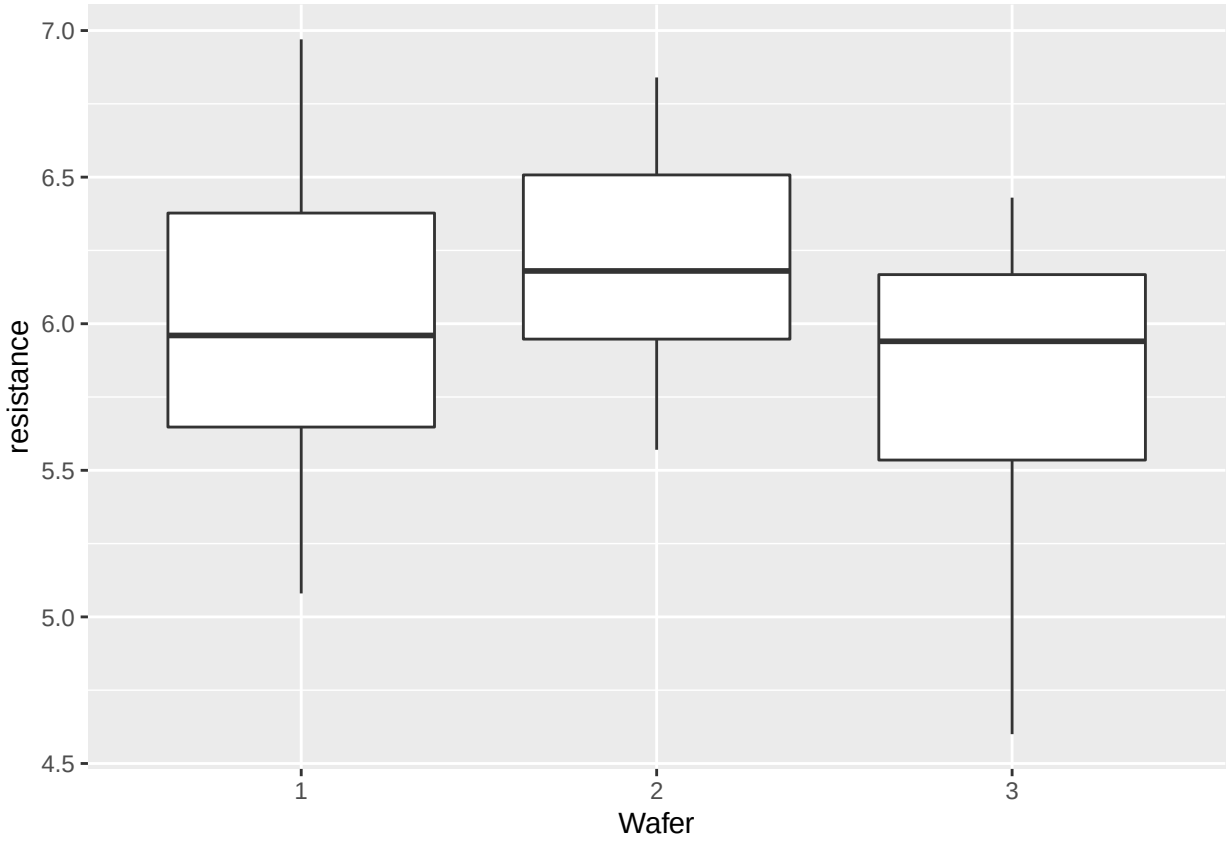
```
data(semicond)
head(semicond)
```

```
##   resistance ET Wafer position Grp
## 1       5.22  1     1        1 1/1
## 2       5.61  1     1        2 1/1
## 3       6.11  1     1        3 1/1
## 4       6.33  1     1        4 1/1
## 5       6.13  1     2        1 1/2
## 6       6.14  1     2        2 1/2
```

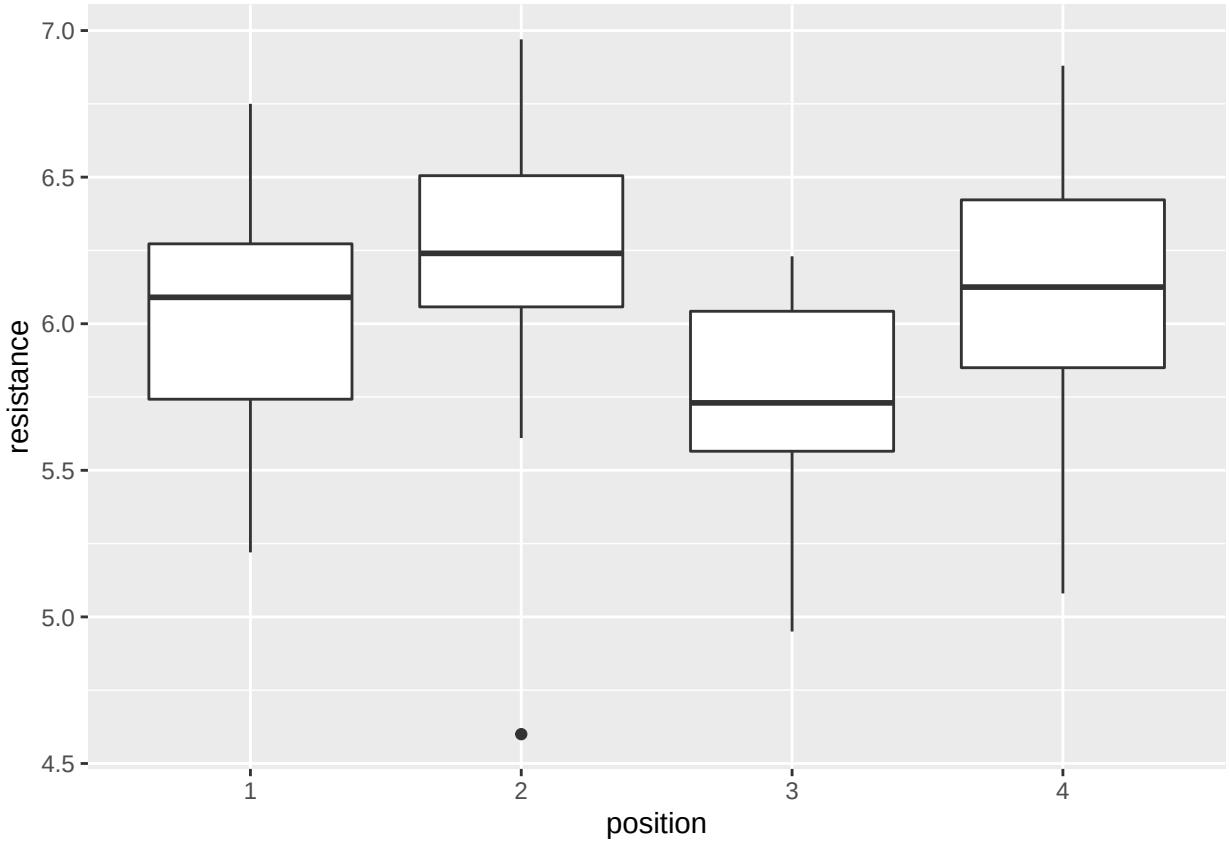
```
ggplot(semicond, aes(x=ET, y = resistance)) + geom_boxplot()
```



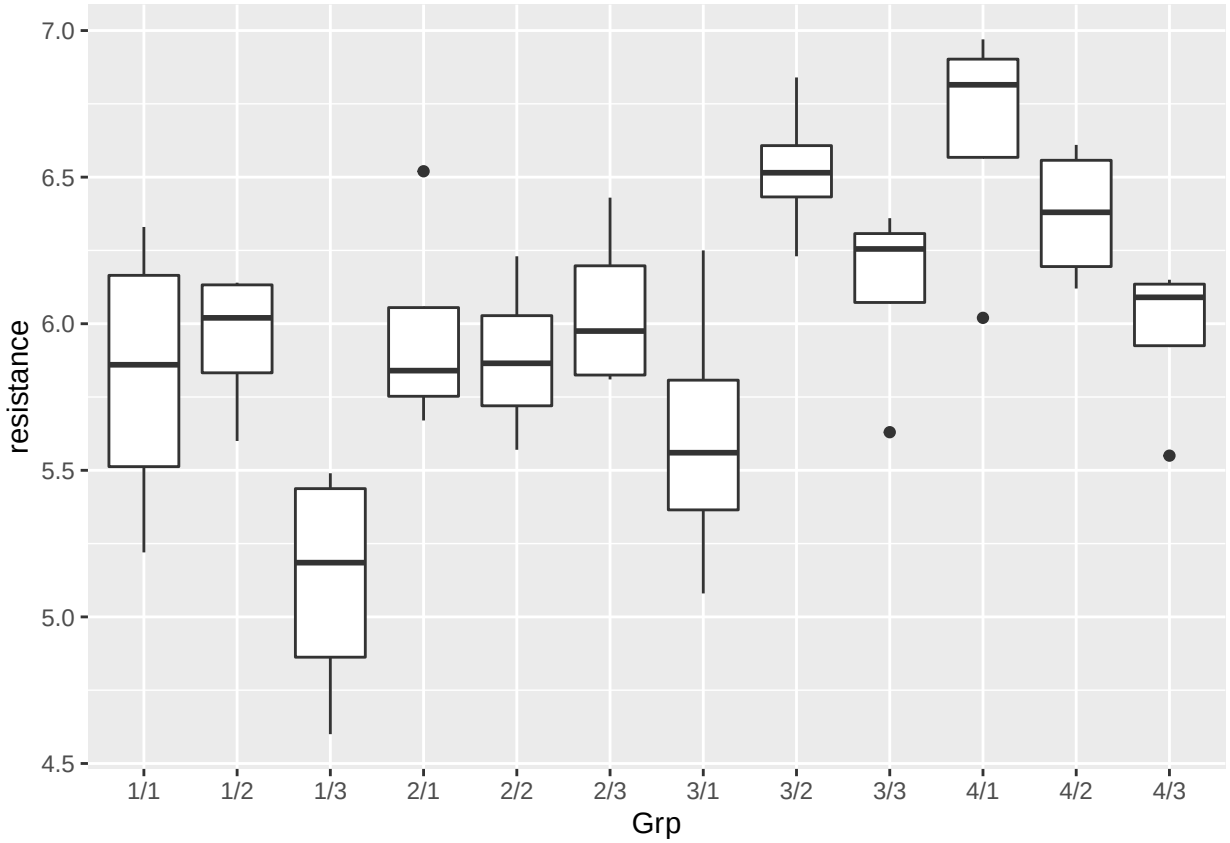

```
ggplot(semicond, aes(x=Wafer, y = resistance)) + geom_boxplot()
```




```
ggplot(semicond, aes(x=position, y = resistance)) + geom_boxplot()
```




```
ggplot(semicond, aes(x=Grp, y = resistance)) + geom_boxplot()
```



The Grp variable appears to be the most significant in terms of the expected resistance. There may be differences with variation within the ET and other variables.

b)

```
mod = lm(resistance ~ position*ET, data = semicond)
summary(mod)
```

```
##
## Call:
## lm(formula = resistance ~ position * ET, data = semicond)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.01333 -0.25750  0.04333  0.28333  0.74667
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.61333    0.26891  20.874   <2e-16 ***
## position2     -0.16333    0.38030  -0.429    0.670
## position3     -0.06000    0.38030  -0.158    0.876
## position4      0.27333    0.38030   0.719    0.478
## ET2            0.38000    0.38030   0.999    0.325
## ET3            0.52333    0.38030   1.376    0.178
## ET4            0.72667    0.38030   1.911    0.065 .
## position2:ET2  0.35667    0.53782   0.663    0.512
```

```
## position3:ET2 -0.16667    0.53782  -0.310    0.759
## position4:ET2 -0.35000    0.53782  -0.651    0.520
## position2:ET3  0.37333    0.53782   0.694    0.493
## position3:ET3 -0.30333    0.53782  -0.564    0.577
## position4:ET3 -0.31667    0.53782  -0.589    0.560
## position2:ET4  0.37667    0.53782   0.700    0.489
## position3:ET4 -0.38333    0.53782  -0.713    0.481
## position4:ET4 -0.07333    0.53782  -0.136    0.892
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4658 on 32 degrees of freedom
## Multiple R-squared:  0.4211, Adjusted R-squared:  0.1498
## F-statistic: 1.552 on 15 and 32 DF,  p-value: 0.1449
```

According to this model no terms are significant aside from the intercept, although ET4 was close. The problem with this model is it does not include a term accounting for the random variability within the GRP blocks.

c)

```
mmod=lmer(resistance~ ET*position+ (1|Grp),semicond)
summary(mmod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: resistance ~ ET * position + (1 | Grp)
##    Data: semicond
##
## REML criterion at convergence: 50.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.91111 -0.45920  0.01029  0.46868  1.31146
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Grp      (Intercept) 0.1058   0.3253
##  Residual             0.1111   0.3334
## Number of obs: 48, groups:  Grp, 12
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    5.61333    0.26891  20.874
## ET2            0.38000    0.38030   0.999
## ET3            0.52333    0.38030   1.376
## ET4            0.72667    0.38030   1.911
## position2     -0.16333    0.27221  -0.600
## position3     -0.06000    0.27221  -0.220
## position4      0.27333    0.27221   1.004
## ET2:position2  0.35667    0.38497   0.926
## ET3:position2  0.37333    0.38497   0.970
## ET4:position2  0.37667    0.38497   0.978
## ET2:position3 -0.16667    0.38497  -0.433
## ET3:position3 -0.30333    0.38497  -0.788
```

```
## ET4:position3 -0.38333    0.38497  -0.996
## ET2:position4 -0.35000    0.38497  -0.909
## ET3:position4 -0.31667    0.38497  -0.823
## ET4:position4 -0.07333    0.38497  -0.190
```

The variation between Grp groups is .1 with an SD of .32 and the variation within a group is .1 with a variation of .33.

d)

```
nmod=lmer(resistance~ ET+ (1|Grp),semicond)
KRmodcomp(mmod,nmod)
```

```
## F-test with Kenward-Roger approximation; computing time: 0.05 sec.
## large : resistance ~ ET * position + (1 | Grp)
## small : resistance ~ ET + (1 | Grp)
##          stat     ndf     ddf F.scaling p.value
## Ftest  1.4533 12.0000 24.0000         1    0.21
```

The Kenward Rodgers test shows that there is not statistical evidence for the effect of the position variable.

e)

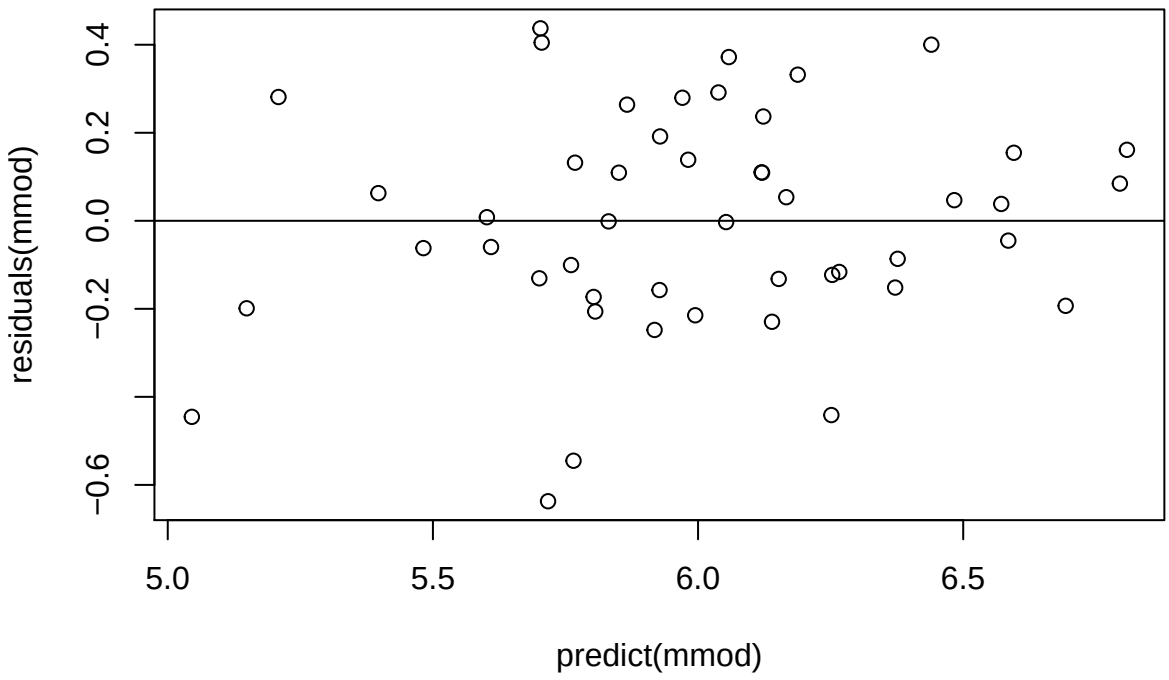
```
fixef(mmod)
```

```
##   (Intercept)           ET2           ET3           ET4     position2
##    5.61333333    0.38000000    0.52333333    0.72666667   -0.16333333
##     position3     position4 ET2:position2 ET3:position2 ET4:position2
##   -0.06000000    0.27333333    0.35666667    0.37333333    0.37666667
## ET2:position3 ET3:position3 ET4:position3 ET2:position4 ET3:position4
##   -0.16666667   -0.30333333   -0.38333333   -0.35000000   -0.31666667
## ET4:position4
##   -0.07333333
```

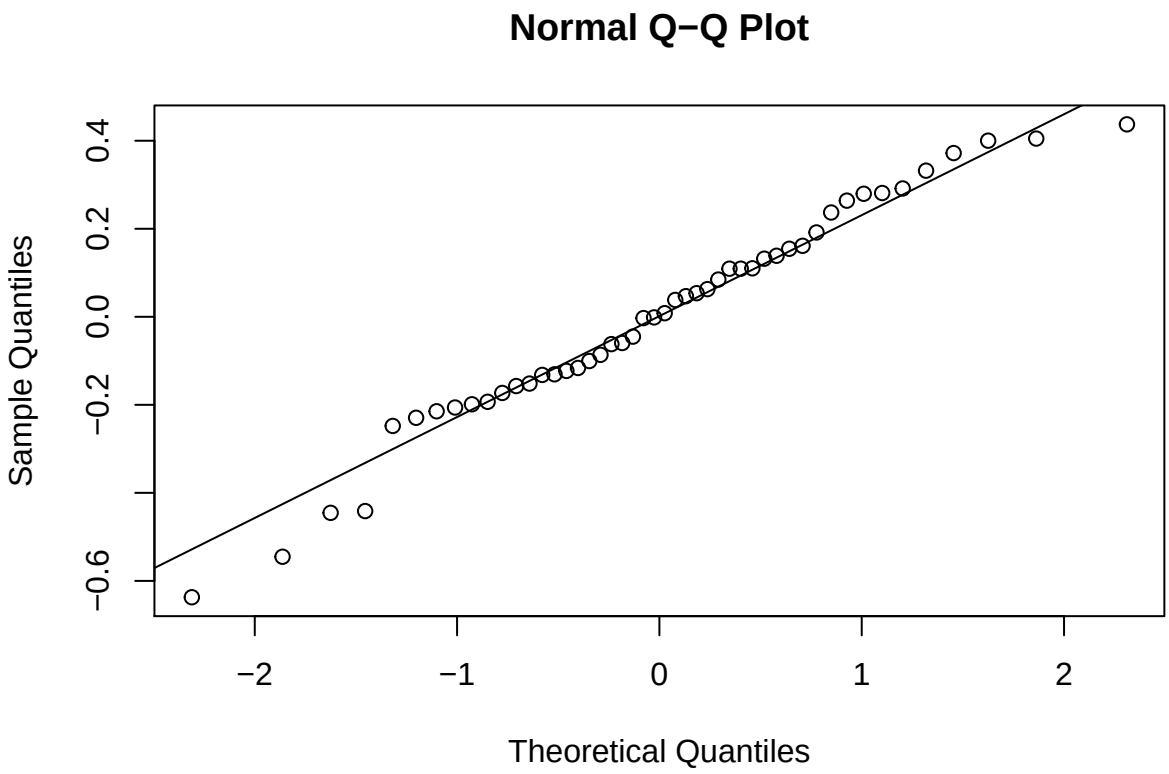
ET4 Results in the higest resistence, but we cannot be sure due to the interaction terms, but mainly because of the random effects by the Grp variable. That random variability must be considered when deciding the ET yielding highest resistence.

f)

```
plot(predict(mmod), residuals(mmod))
abline(a = 0, b=0)
```




```
qqnorm(residuals(mmod))
qqline(residuals(mmod))
```



The residuals appear to be fairly normal and centered around zero.