Stat 5309 Midterm Project

Tom Wilson

Mar 21, 2019

1

Three brands of batteries are under study. It is suspected that the lives (in weeks) of the three brands are different. Five batteries of each brand are tested with the following results.

```
brands <- c("b1","b2","b3")
battery_data <- data.frame(brand=rep(brands,5),
                   weeks_of_life=c(100,76,108,
                                   96,80,100,
                                   92,75,96,
                                   96,84,98,
                                   92,82,100))
battery_data %>% kable()
```

brand	weeks_of_life
b1	100
b2	76
b3	108
b1	96
b2	80
b3	100
b1	92
b2	75
b3	96
b1	96
b2	84
b3	98
b1	92
b2	82
b3	100

a

build a linear model, using aov. Are the lives of these brands of batteries different? which brand gives the longest life?

```
battery_model <- aov(formula = weeks_of_life ~ brand,data=battery_data)
summary(battery_model)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## brand        2 1196.1   598.1   38.34 6.14e-06 ***
## Residuals   12  187.2    15.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

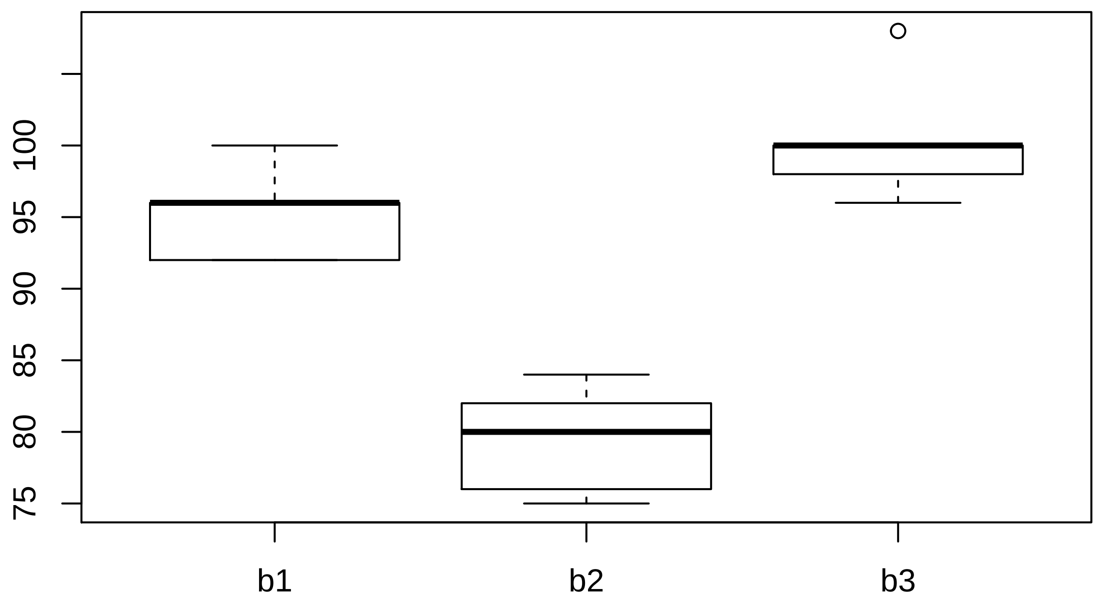
```
brand_means <- tapply(battery_data$weeks_of_life,battery_data$brand,mean)
brand_means
```

```
##    b1    b2    b3
##  95.2  79.4 100.4
```

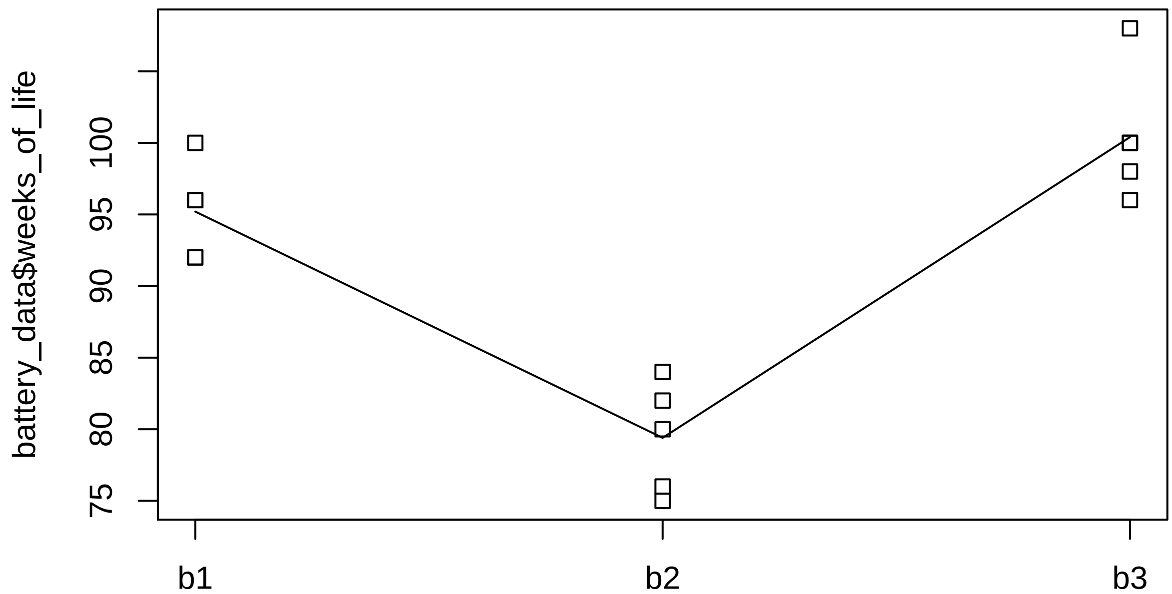
b

Perform a boxplot and a stripchart.

```
boxplot(battery_data$weeks_of_life~battery_data$brand)
```




```
stripchart(battery_data$weeks_of_life~battery_data$brand,vertical = TRUE)
lines(brand_means)
```



c

construct a 95% CI for the mean life of brand 2. Using Fisher LSD. Construct a 99% CI for the mean difference between the lives of brand 2 and 3, using Tukey HSD.

```
battery_model_anova <- anova(battery_model)
MSError=battery_model_anova$`Mean Sq`[2]
LSD.test(battery_model, "brand", MSerror = MSError,console = TRUE)
```

```
##
## Study: battery_model ~ "brand"
##
## LSD t Test for weeks_of_life
##
## Mean Square Error:  15.6
##
## brand,  means and individual ( 95 %) CI
##
##    weeks_of_life      std r      LCL       UCL Min Max
## b1          95.2 3.346640 5 91.35145  99.04855  92 100
## b2          79.4 3.847077 5 75.55145  83.24855  75  84
## b3         100.4 4.560702 5 96.55145 104.24855  96 108
##
## Alpha: 0.05 ; DF Error: 12
## Critical Value of t: 2.178813
##
```

```
## least Significant Difference: 5.442673
##
## Treatments with the same letter are not significantly different.
##
##    weeks_of_life groups
## b3         100.4      a
## b1          95.2      a
## b2          79.4      b
```

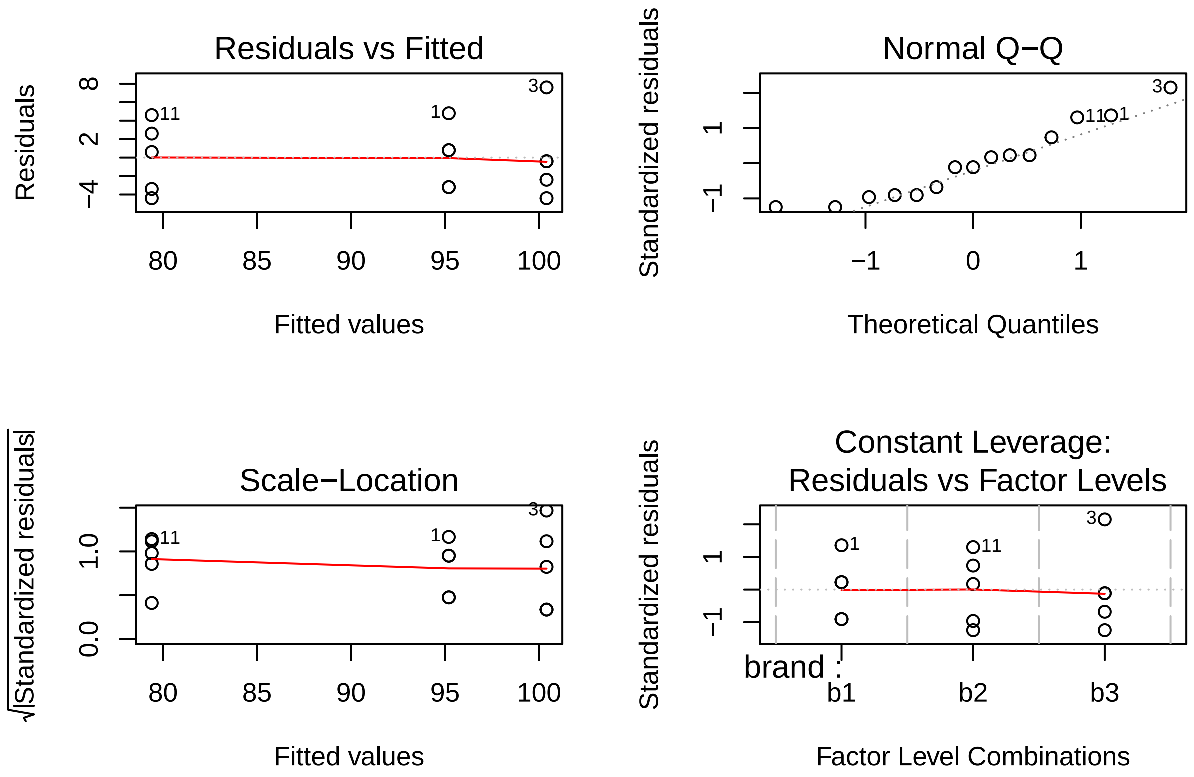
```r
TukeyHSD(battery_model, conf.level=0.99)
```

```
##   Tukey multiple comparisons of means
##     99% family-wise confidence level
##
## Fit: aov(formula = weeks_of_life ~ brand, data = battery_data)
##
## $brand
##        diff        lwr       upr     p adj
## b2-b1 -15.8 -24.712897 -6.887103 0.0001044
## b3-b1   5.2  -3.712897 14.112897 0.1355226
## b3-b2  21.0  12.087103 29.912897 0.0000063
```

d

perform the 3part residual check.

```r
par( mfrow = c(2,2) )
plot(battery_model)
```



e

calculate the number of replicates for a power of 0.9

```
brand_means <- battery_data %>%
               group_by(brand) %>%
               summarise(trt_mean = mean(weeks_of_life))

power.anova.test(groups = nrow(brand_means),
                 between.var = var(brand_means$trt_mean),
                 within.var = battery_model_anova$`Mean Sq`[2],
                 power=0.9
                 )
```

```
##
##      Balanced one-way analysis of variance power calculation
##
##          groups = 3
##               n = 2.206129
##     between.var = 119.6133
##      within.var = 15.6
##       sig.level = 0.05
##           power = 0.9
##
## NOTE: n is number in each group
```

3 replicates are needed for a power of 0.9

2

A manufacturer of television sets is interested in the effect of tube conductivity of four different types of coating for color picture tubes. The following conductivity data are obtained.

```
coatings <- c("c1","c2","c3","c4")
conductivity_data<- data.frame(coating = rep(coatings,each=4),
                   conductivity = c(143,141,150,146,
                                    152,149,137,143,
                                    134,136,132,127,
                                    129,127,132,129))
conductivity_data%>% kable()
```

coating	conductivity
c1	143
c1	141
c1	150
c1	146
c2	152
c2	149
c2	137
c2	143
c3	134
c3	136
c3	132
c3	127
c4	129
c4	127
c4	132
c4	129

a

Is there a difference in conductivity due to coating type?

```
conductivity_model <- aov(formula = conductivity~coating,data=conductivity_data)
summary(conductivity_model)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## coating      3  844.7  281.56    14.3 0.000288 ***
## Residuals   12  236.3   19.69
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

b

Estimate the overall mean and the treatment effects.

```r
grand_mean <- mean(conductivity_data$conductivity)
grand_mean
```

```
## [1] 137.9375
```

```r
trt_means <- tapply(conductivity_data$conductivity,conductivity_data$coating,mean)
trt_means
```

```
##     c1     c2     c3     c4
## 145.00 145.25 132.25 129.25
```

c

Compute a 95% CI of the mean of coating type 4.

```r
conductivity_model_anova <- anova(conductivity_model)
MSError=conductivity_model_anova$`Mean Sq`[2]
LSD.test(conductivity_model, "coating", MSerror = MSError,console = TRUE)
```

```
##
## Study: conductivity_model ~ "coating"
##
## LSD t Test for conductivity
##
## Mean Square Error:  19.6875
##
## coating,  means and individual ( 95 %) CI
##
##    conductivity      std r      LCL      UCL Min Max
## c1       145.00 3.915780 4 140.1662 149.8338 141 150
## c2       145.25 6.652067 4 140.4162 150.0838 137 152
## c3       132.25 3.862210 4 127.4162 137.0838 127 136
## c4       129.25 2.061553 4 124.4162 134.0838 127 132
##
## Alpha: 0.05 ; DF Error: 12
## Critical Value of t: 2.178813
##
## least Significant Difference: 6.835971
##
## Treatments with the same letter are not significantly different.
##
##    conductivity groups
## c2       145.25      a
## c1       145.00      a
## c3       132.25      b
## c4       129.25      b
```

124.4162 to 134.0838 is a 95% CI for coating 4.

Compute a 99% CI for the difference between types 1 and 4.

```r
TukeyHSD(conductivity_model, conf.level=0.99)
```

```
##   Tukey multiple comparisons of means
##     99% family-wise confidence level
##
```

```
## Fit: aov(formula = conductivity ~ coating, data = conductivity_data)
##
## $coating
##          diff       lwr        upr     p adj
## c2-c1   0.25 -11.95552 12.4555225 0.9998078
## c3-c1 -12.75 -24.95552 -0.5444775 0.0073964
## c4-c1 -15.75 -27.95552 -3.5444775 0.0014707
## c3-c2 -13.00 -25.20552 -0.7944775 0.0064441
## c4-c2 -16.00 -28.20552 -3.7944775 0.0012913
## c4-c3  -3.00 -15.20552  9.2055225 0.7759360
```

-27.9555 to -3.5445 is a 99% CI for the difference between coating 1 and 4.

d

Test all pairs of means using fisher lsd.

```r
pairwise.t.test(conductivity_data$conductivity,conductivity_data$coating, p.adj = "bonf")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  conductivity_data$conductivity and conductivity_data$coating
##
##    c1     c2     c3
## c2 1.0000 -      -
## c3 0.0094 0.0082 -
## c4 0.0018 0.0016 1.0000
##
## P value adjustment method: bonferroni
```

1 and 2 are different from 3 and 4

3

An article in the fire safety journal describes an experiment in which a shape factor was determined for several different nozzle designs at six levels of jet efflux velocity. Interest focused on potential differences between nozzle designs, with velocity considered as a nuisance variable. The data are shown below.

```r
nozzles <- c("n1","n2","n3","n4","n5")
jet_efflux_velocities <- c(11.73,14.37,16.59,20.43,23.46,28.74)
shape_data <- expand.grid(jet_efflux_velocity = jet_efflux_velocities,nozzle=nozzles)
shape_data <- cbind(shape_data, shape_factor=c(0.78,0.80,0.81,0.75,0.77,0.78,
                                               0.85,0.85,0.92,0.86,0.81,0.83,
                                               0.93,0.92,0.95,0.89,0.89,0.83,
                                               1.14,0.97,0.98,0.88,0.86,0.83,
                                               0.97,0.86,0.78,0.76,0.76,0.75
                                               )
                    )
shape_data %>% kable()
```

jet_efflux_velocity	nozzle	shape_factor
11.73	n1	0.78

jet_efflux_velocity	nozzle	shape_factor
14.37	n1	0.80
16.59	n1	0.81
20.43	n1	0.75
23.46	n1	0.77
28.74	n1	0.78
11.73	n2	0.85
14.37	n2	0.85
16.59	n2	0.92
20.43	n2	0.86
23.46	n2	0.81
28.74	n2	0.83
11.73	n3	0.93
14.37	n3	0.92
16.59	n3	0.95
20.43	n3	0.89
23.46	n3	0.89
28.74	n3	0.83
11.73	n4	1.14
14.37	n4	0.97
16.59	n4	0.98
20.43	n4	0.88
23.46	n4	0.86
28.74	n4	0.83
11.73	n5	0.97
14.37	n5	0.86
16.59	n5	0.78
20.43	n5	0.76
23.46	n5	0.76
28.74	n5	0.75

a

build a linear model nozzle as a blocking factor. Does the nozzle design affect the shape factor?

```r
shape_model <- aov(formula = shape_factor~nozzle+jet_efflux_velocity,data=shape_data)
summary(shape_model)
```

```
##                     Df  Sum Sq Mean Sq F value   Pr(>F)    
## nozzle               4 0.10218 0.02555   9.453 9.79e-05 ***
## jet_efflux_velocity  1 0.05531 0.05531  20.470 0.000139 ***
## Residuals           24 0.06485 0.00270                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nozzel design has a significant effect on shape factor.

b

which nozzle design are different with respect to the shape factor?

```r
TukeyHSD(shape_model, conf.level=0.95)
```

```
## Warning in replications(paste("~", xx), data = mf): non-factors ignored:
## jet_efflux_velocity

## Warning in TukeyHSD.aov(shape_model, conf.level = 0.95): 'which' specified
## some non-factors which will be dropped

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = shape_factor ~ nozzle + jet_efflux_velocity, data = shape_data)
##
## $nozzle
##              diff          lwr           upr     p adj
## n2-n1  0.07166667 -0.016749772  1.600831e-01 0.1530502
## n3-n1  0.12000000  0.031583562  2.084164e-01 0.0043762
## n4-n1  0.16166667  0.073250228  2.500831e-01 0.0001409
## n5-n1  0.03166667 -0.056749772  1.200831e-01 0.8270774
## n3-n2  0.04833333 -0.040083105  1.367498e-01 0.5053306
## n4-n2  0.09000000  0.001583562  1.784164e-01 0.0446210
## n5-n2 -0.04000000 -0.128416438  4.841644e-02 0.6741865
## n4-n3  0.04166667 -0.046749772  1.300831e-01 0.6406338
## n5-n3 -0.08833333 -0.176749772  8.310495e-05 0.0502979
## n5-n4 -0.13000000 -0.218416438 -4.158356e-02 0.0019336
```

In order of significance, these pairs are different:

n4-n1 = 0.16166667 pvalue = 0.0001409 n5-n4 = -0.13000000 pvalue = 0.0019336 n3-n1 = 0.12000000 pvalue = 0.0043762 n4-n2 = 0.09000000 pvalue = 0.0446210

c

Is the velocity effect significant? At a pvalue of 0.0001 velocity has a significant effect on shape factor.

d

Check if nozzle is an effective blocking factor. At a pvalue of 9.79e-05 nozzle has a significant effect on shape factor, therefore the blocked design is effective.

4

Johnson and Leone describe an experiment to investigate warping of copper plates. The two factors studied were the temperature and the copper content of the plates. The response variable was a measure of the amount of warping. The data were as follows.

```r
temperatures <- c("50","75","100","125")
cu_contents <- c("40","60","80","100")
warp_data <- expand.grid(cu_content=rep(cu_contents,2),temperature=temperatures)
warp_data <- cbind(warp_data,warping=c(17,16,24,28,
                                       20,21,22,27,
                                       12,18,17,27,
                                       09,13,12,31,
                                       16,18,25,30,
                                       12,21,23,23,
```

```
                                                    21,23,23,29,
                                                    17,21,22,31
                                                    )
                    )
warp_data %>% kable()
```

cu_content	temperature	warping
40	50	17
60	50	16
80	50	24
100	50	28
40	50	20
60	50	21
80	50	22
100	50	27
40	75	12
60	75	18
80	75	17
100	75	27
40	75	9
60	75	13
80	75	12
100	75	31
40	100	16
60	100	18
80	100	25
100	100	30
40	100	12
60	100	21
80	100	23
100	100	23
40	125	21
60	125	23
80	125	23
100	125	29
40	125	17
60	125	21
80	125	22
100	125	31
## a		
is there any	indication tha	t either factor affects the amount of warping?

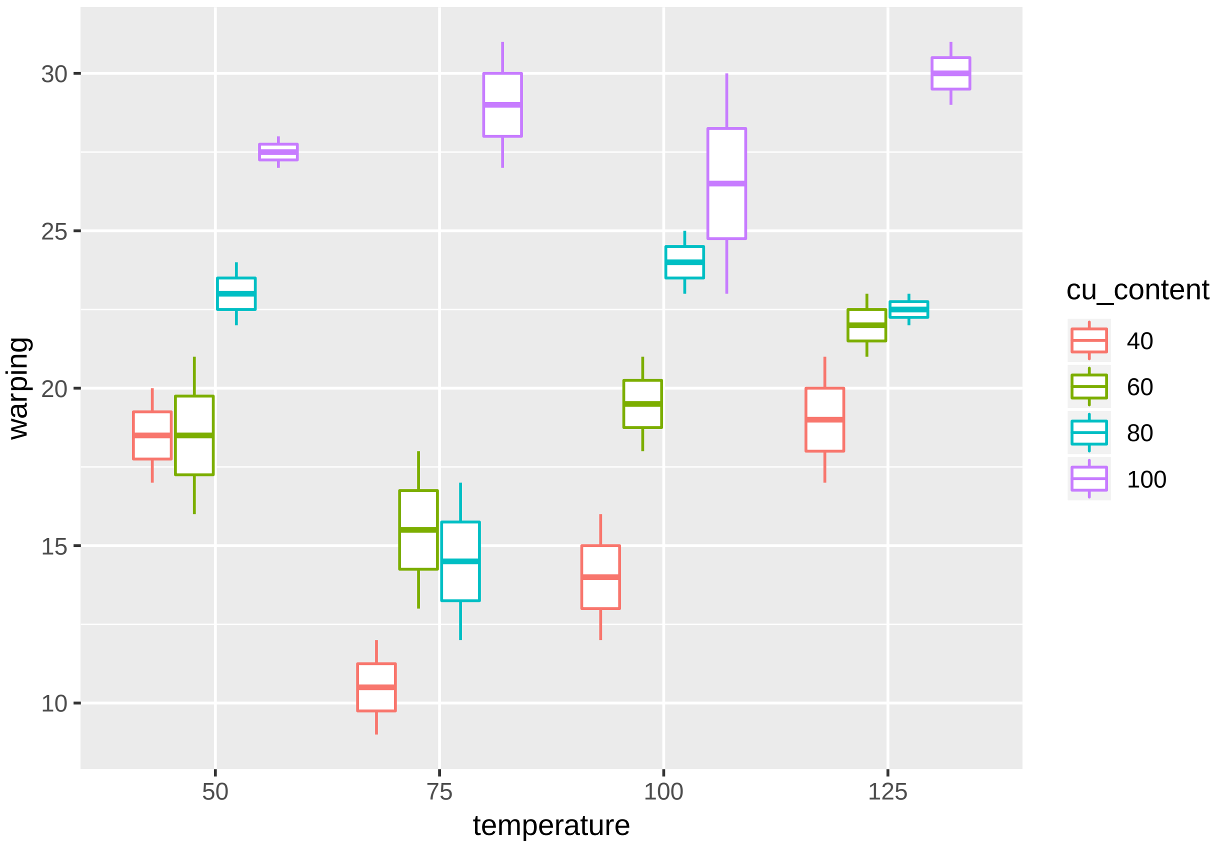
```
warp_model <- aov(formula = warping~cu_content+temperature,data=warp_data)
summary(warp_model)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)    
## cu_content   3  698.3  232.78  26.181 6.98e-08 ***
## temperature  3  156.1   52.03   5.852  0.00359 ** 
## Residuals   25  222.3    8.89                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

b

Do a box plot of warp vs temp and content together. which level combination gives the lowest warpage?

```
ggplot(data = warp_data,
       aes(x=temperature,
           y=warping,
           color=cu_content
           )
       ) + geom_boxplot()
```

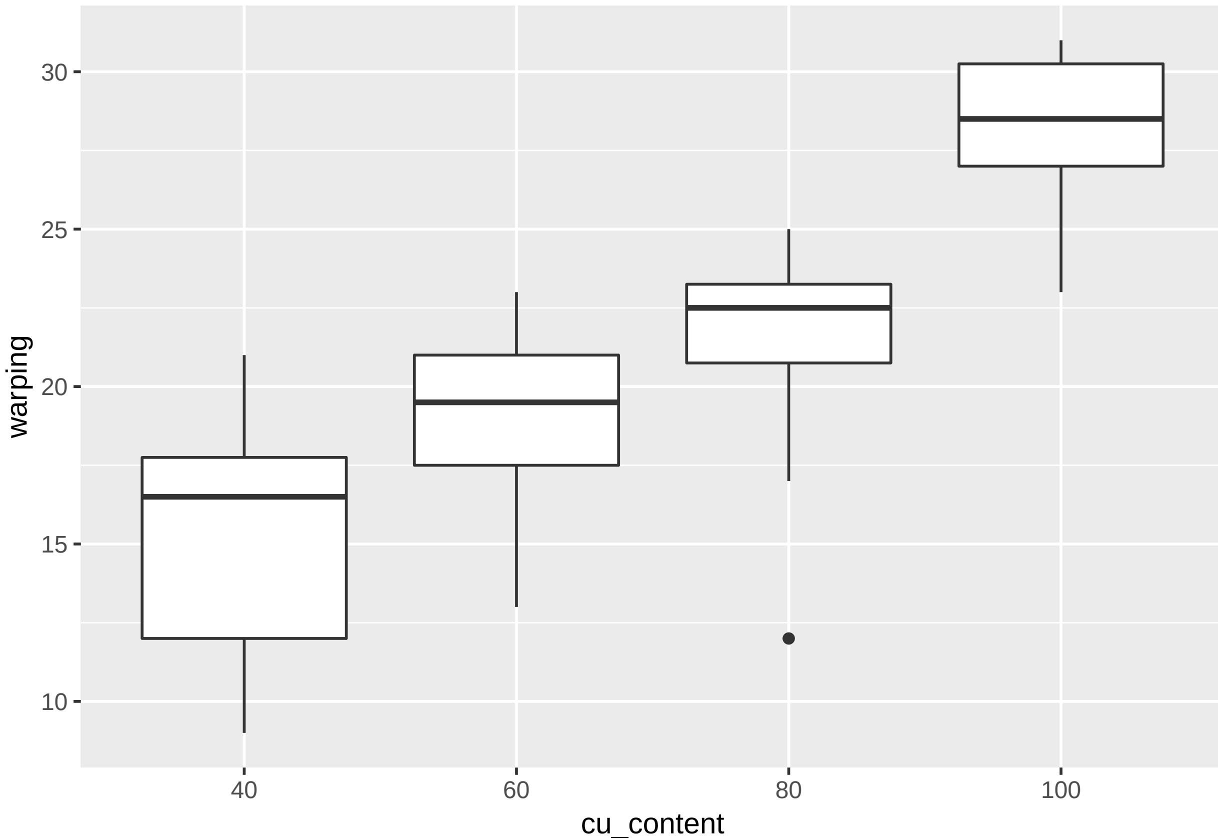


temperature = 75 and Cu content = 40 is associated with the lowest warpage.

c

Suppose that temperature cannot be controlled in where the copper plate are used. Any conclusion about the content? (i.e. any difference among means?)

```
ggplot(data = warp_data,
       aes(x=cu_content,
           y=warping,
           )
       ) + geom_boxplot()
```

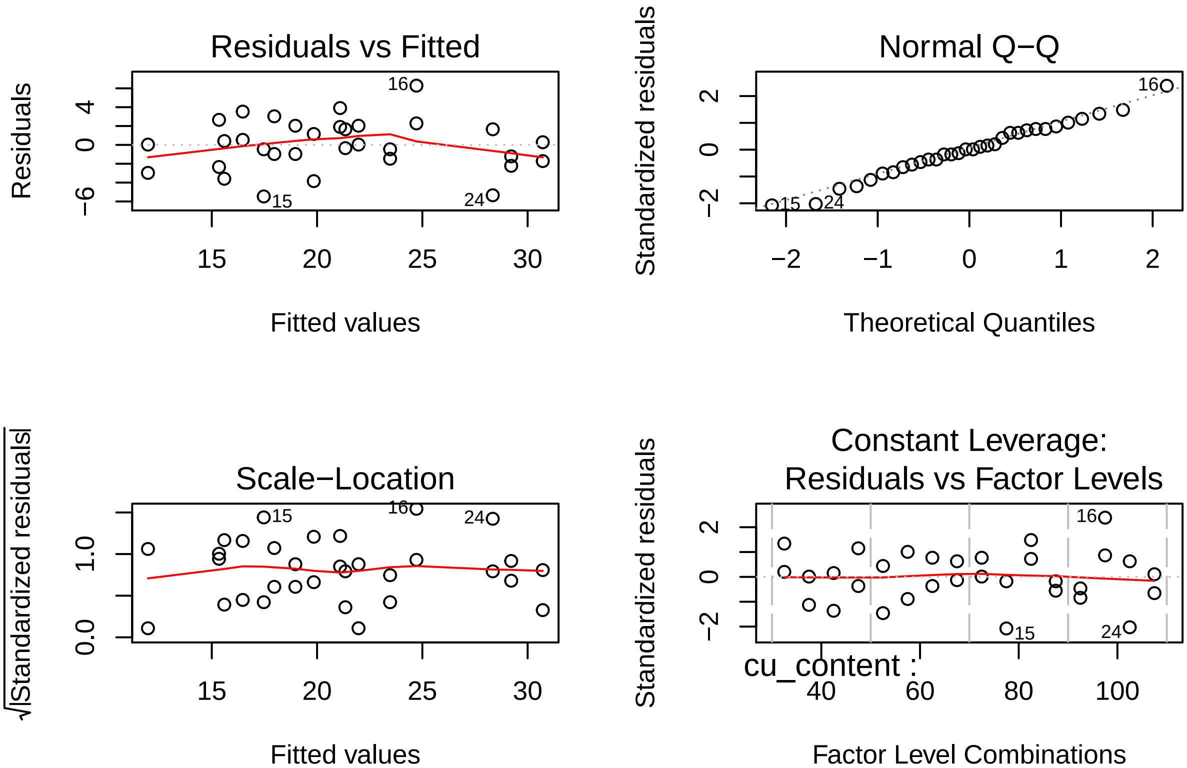


If temperature cannot be controlled, Cu content of 60 has less variation in warpage than 40 or 80 while maintaining significantly less warpage than 100.

d

perform a 3-part residual check.

```
par( mfrow = c(2,2) )
plot(warp_model)
```



Independence

Based on a plot of standardized residual vs factor level (in this case, temperature and cu content) the residuals are nearly independent of cu content or temperature.

Normality

Based on a quantile-quantile plot te residuals are very close to normally distrbuted with a mean of zero.

Homoscedasticity

based on a plot of residual vs fitted values, variation in residual nearly constant.

We can be confident in making statistical inferences about the coefficients of this model.