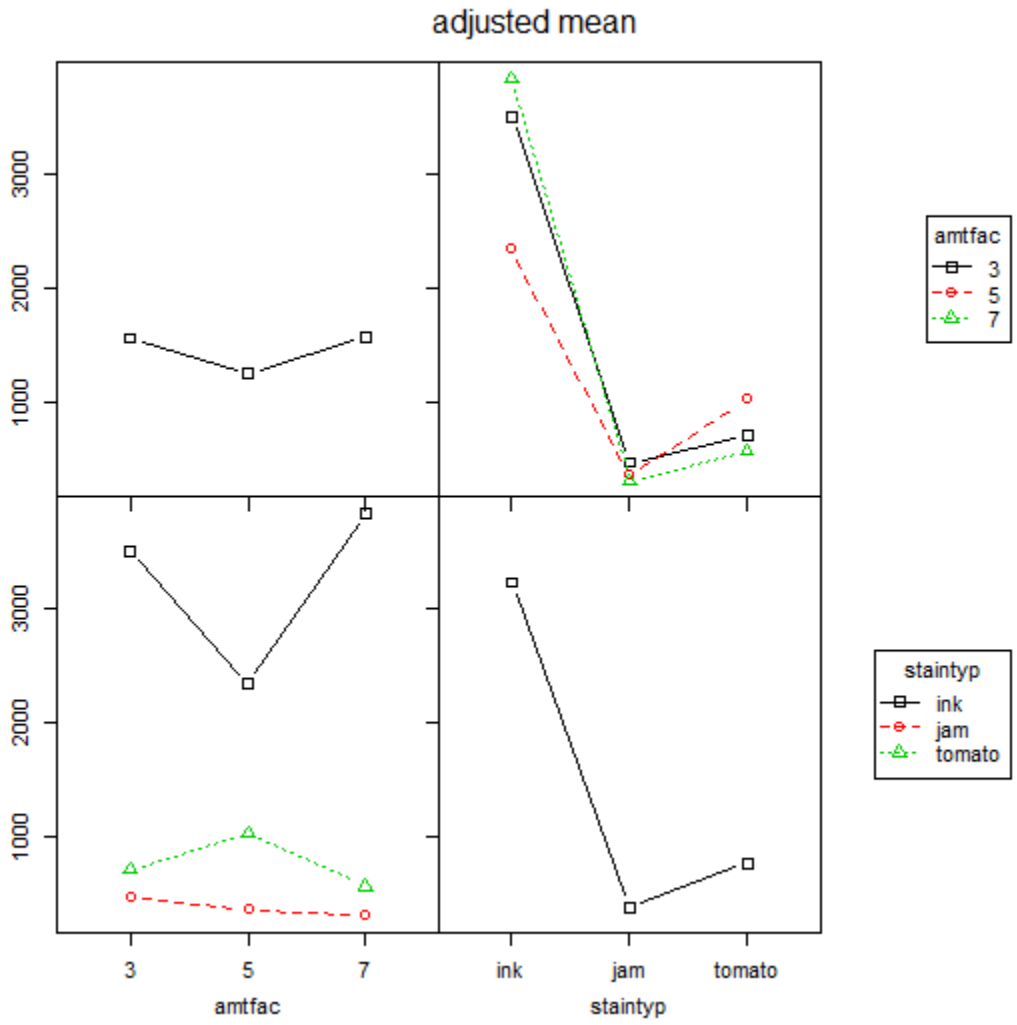
Output from bleach.R

```
> # bleach.R
>
> library(lsmeans)
> library(car)
> library(phia)
> library(doBy)
>
> # get the data
> bleach<-read.table(file="bleach.dat",header=T)
> head(bleach)
  amtblch staintyp time
1       3      ink 3600
2       3      ink 3920
3       3      ink 3340
4       3      ink 3173
5       3      jam  495
6       3      jam  236
> bleach$amtfac <- factor(bleach$amtblch)
> is.ordered(bleach$amtfac)
[1] FALSE
> is.ordered(bleach$staintyp)
[1] FALSE
>
> ######PART 1
>
> # fit the two-way anova model
> m1 <- aov(time~amtfac*staintyp,data=bleach)
> summary(m1)
               Df   Sum Sq  Mean Sq F value   Pr(>F)
amtfac          2  1229568   614784   7.662  0.00191 **
staintyp        2 64714567 32357284 403.245  < 2e-16 ***
amtfac:staintyp  4  5411147  1352787  16.859 1.57e-07 ***
Residuals      32  2567752    80242
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # note that the summary function gives F tests based on type I
> # or sequential sums of squares. To get F tests based on the type II and III
> # SS, we need the Anova() function from the car package.
>
> Anova(m1,type=2)
Anova Table (Type II tests)

Response: time
                 Sum Sq Df  F value    Pr(>F)
amtfac           1184748  2   7.3823  0.002311 **
staintyp        64714567  2 403.2449 < 2.2e-16 ***
amtfac:staintyp  5411147  4  16.8588 1.569e-07 ***
Residuals        2567752 32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # the Anova() function from the car package claims to produce Type III
> # SSs and F tests, but if you run the line below you'll see that you get
> # results quite different from those provided by SAS.
> Anova(m1,type=3)
```

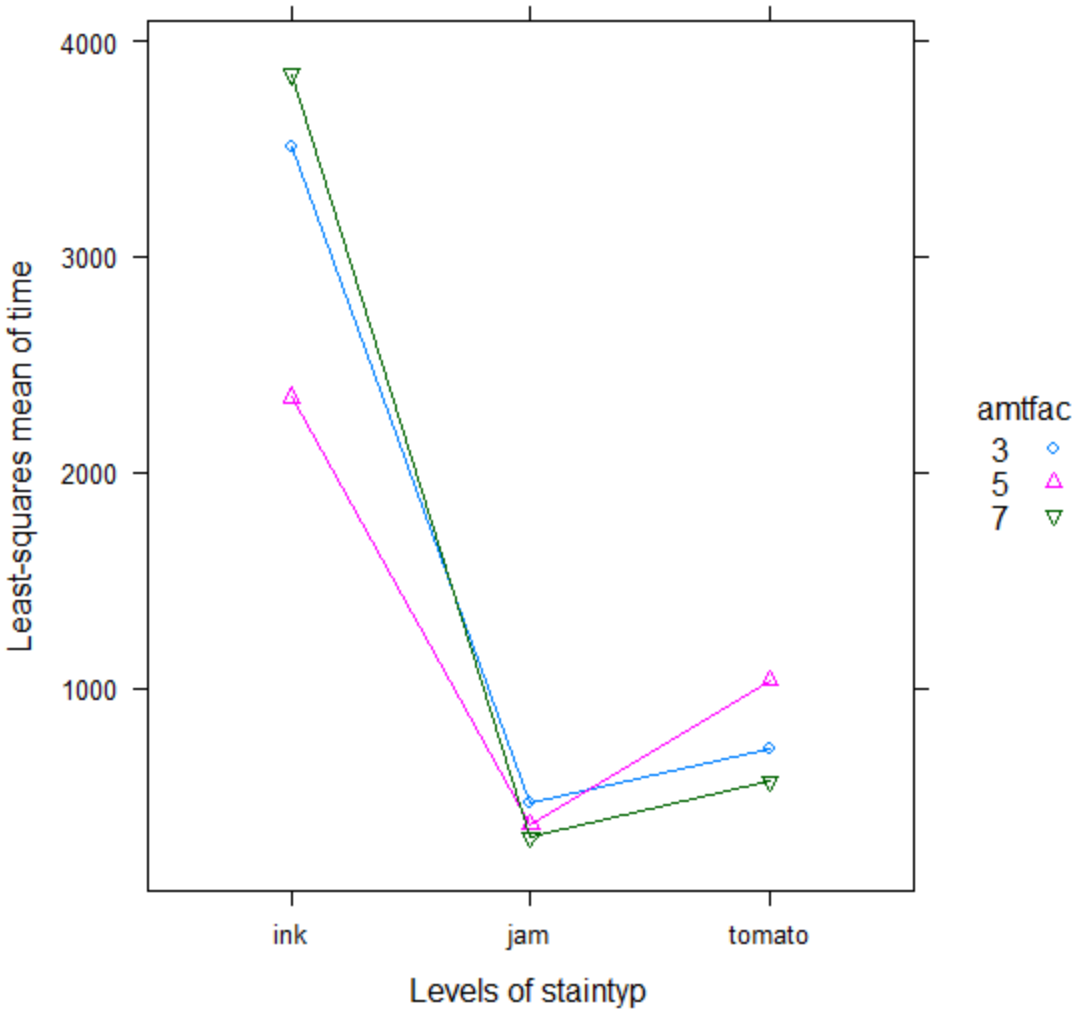
```
Anova Table (Type III tests)

Response: time
                  Sum Sq Df   F value    Pr(>F)
(Intercept)     85505806  1 1065.5957 < 2.2e-16 ***
amtfac            917710  2    5.7184  0.007528 **
staintyp        65287167  2  406.8129 < 2.2e-16 ***
amtfac:staintyp  5411147  4   16.8588 1.569e-07 ***
Residuals        2567752 32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # The reason for this discrepancy is realted to the method of computing
> # type III SSs in the Anova() function. That method depends upon the
> # parameterization of the model. R and SAS parameterize factors differently.
> # SAS uses a "non-full rank" parameterization (aka an overparameterization),
> # whereas R removes the overparameterization. The user has some control over
> # how that overparameterization is removed. That is, the user has some control
> # over the parameterization that R uses. For the Anova(,type=3) function to
> # produce the "right" type III SSs (the ones that agree with SAS and test
> # hypotheses on marginal means as described in our lecture notes), one
> # has to choose a sum-to-zero parameterization of factor effects. This is
> # not the default in R. Therefore it has to be changed and the model refit.
>
> # Change the handling of unordered factors (like amtfac and staintyp) to
> # use the sum-to-zero constraints:
> op <- options(contrasts=c("contr.sum", "contr.poly"))
> options()$contrasts # now you can see that they've been changed
[1] "contr.sum"  "contr.poly"
>
> # Now refit the model with the new parameterization:
> m1a <- aov(time~amtfac*staintyp,data=bleach)
> # now check the type 1 SSs (they haven't changed):
> summary(m1a)
               Df   Sum Sq  Mean Sq F value   Pr(>F)
amtfac          2  1229568   614784   7.662  0.00191 **
staintyp        2 64714567 32357284 403.245  < 2e-16 ***
amtfac:staintyp  4  5411147  1352787  16.859 1.57e-07 ***
Residuals       32  2567752    80242
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # and check the type II SSs (they haven't changed):
> Anova(m1a,type=2)
Anova Table (Type II tests)

Response: time
                  Sum Sq Df  F value    Pr(>F)
amtfac           1184748  2   7.3823  0.002311 **
staintyp        64714567  2 403.2449 < 2.2e-16 ***
amtfac:staintyp  5411147  4  16.8588 1.569e-07 ***
Residuals        2567752 32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # and check the type III SSs (they have changed and now agree with SAS):
> Anova(m1a,type=3)
```

```
Anova Table (Type III tests)

Response: time
                  Sum Sq Df   F value    Pr(>F)
(Intercept)     85505806  1 1065.5957 < 2.2e-16 ***
amtfac            917710  2    5.7184  0.007528 **
staintyp        65287167  2  406.8129 < 2.2e-16 ***
amtfac:staintyp  5411147  4   16.8588 1.569e-07 ***
Residuals        2567752 32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> ######PART 2
>
> # Now get the marginal and joint lsmeans:
> lsmeans(m1, specs= list(~staintyp,~amtfac,~amtfac:staintyp))
$`staintyp lsmeans`
 staintyp    lsmean       SE df  lower.CL  upper.CL
      ink 3234.4167 76.12670 32 3079.3517 3389.4817
      jam  385.9833 76.12670 32  230.9183  541.0483
   tomato  774.8222 80.85949 32  610.1168  939.5276

$`amtfac lsmeans`
 amtfac   lsmean       SE df lower.CL upper.CL
      3 1566.817 76.12670 32 1411.752 1721.882
      5 1252.650 76.12670 32 1097.585 1407.715
      7 1575.756 80.85949 32 1411.050 1740.461

$`amtfac:staintyp lsmeans`
 amtfac staintyp    lsmean       SE df   lower.CL  upper.CL
      3      ink 3508.2500 141.6353 32 3219.74829 3796.7517
      5      ink 2350.8000 126.6825 32 2092.75623 2608.8438
      7      ink 3844.2000 126.6825 32 3586.15623 4102.2438
      3      jam  474.8000 126.6825 32  216.75623  732.8438
      5      jam  370.7500 141.6353 32   82.24829  659.2517
      7      jam  312.4000 126.6825 32   54.35623  570.4438
      3   tomato  717.4000 126.6825 32  459.35623  975.4438
      5   tomato 1036.4000 126.6825 32  778.35623 1294.4438
      7   tomato  570.6667 163.5464 32  237.53359  903.7997

Warning messages:
1: In lsmeans(m1, specs = list(~staintyp, ~amtfac, ~amtfac:staintyp)) :
  lsmeans of staintyp may be misleading due to interaction with other
predictor(s)
2: In lsmeans(m1, specs = list(~staintyp, ~amtfac, ~amtfac:staintyp)) :
  lsmeans of amtfac may be misleading due to interaction with other predictor(s)
>
> # get profile plots (uses package phia)
> ( trtmns <- interactionMeans(m1) )
  amtfac staintyp adjusted mean
1      3      ink     3508.2500
2      5      ink     2350.8000
3      7      ink     3844.2000
4      3      jam      474.8000
5      5      jam      370.7500
6      7      jam      312.4000
7      3   tomato      717.4000
```

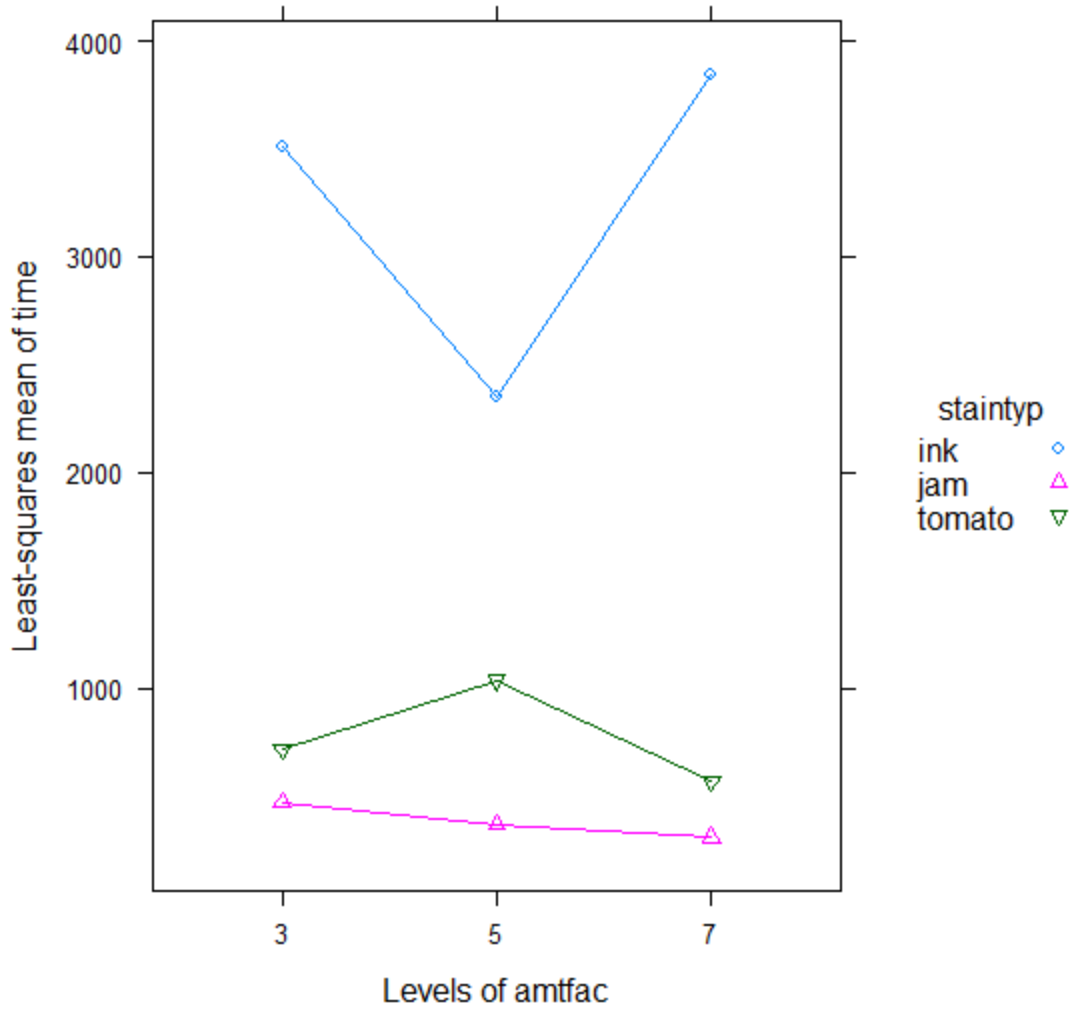
```
8       5   tomato     1036.4000
9       7   tomato      570.6667
> plot(trtmns)
```



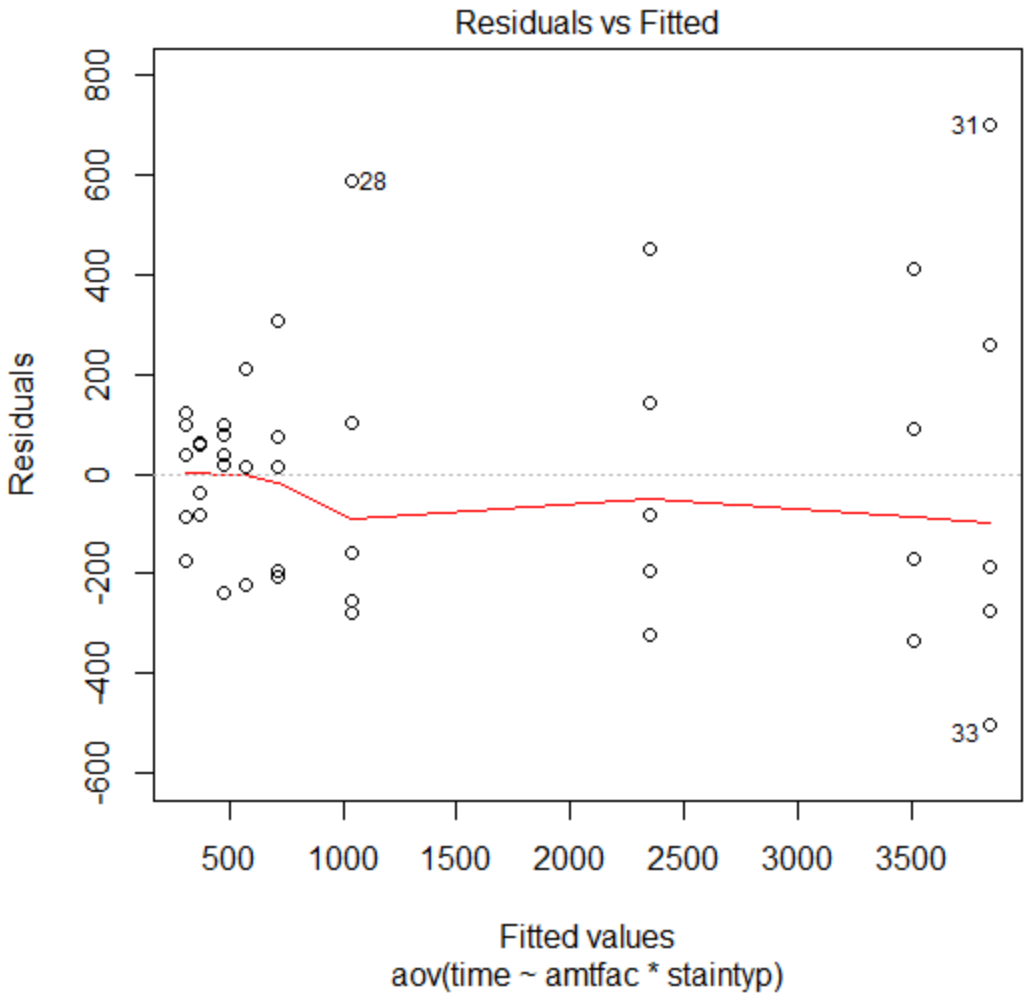

```
>
> # alternatively can use function lsmip from lsmeans package
>
> lsmip(m1,amtfac~staintyp)
```



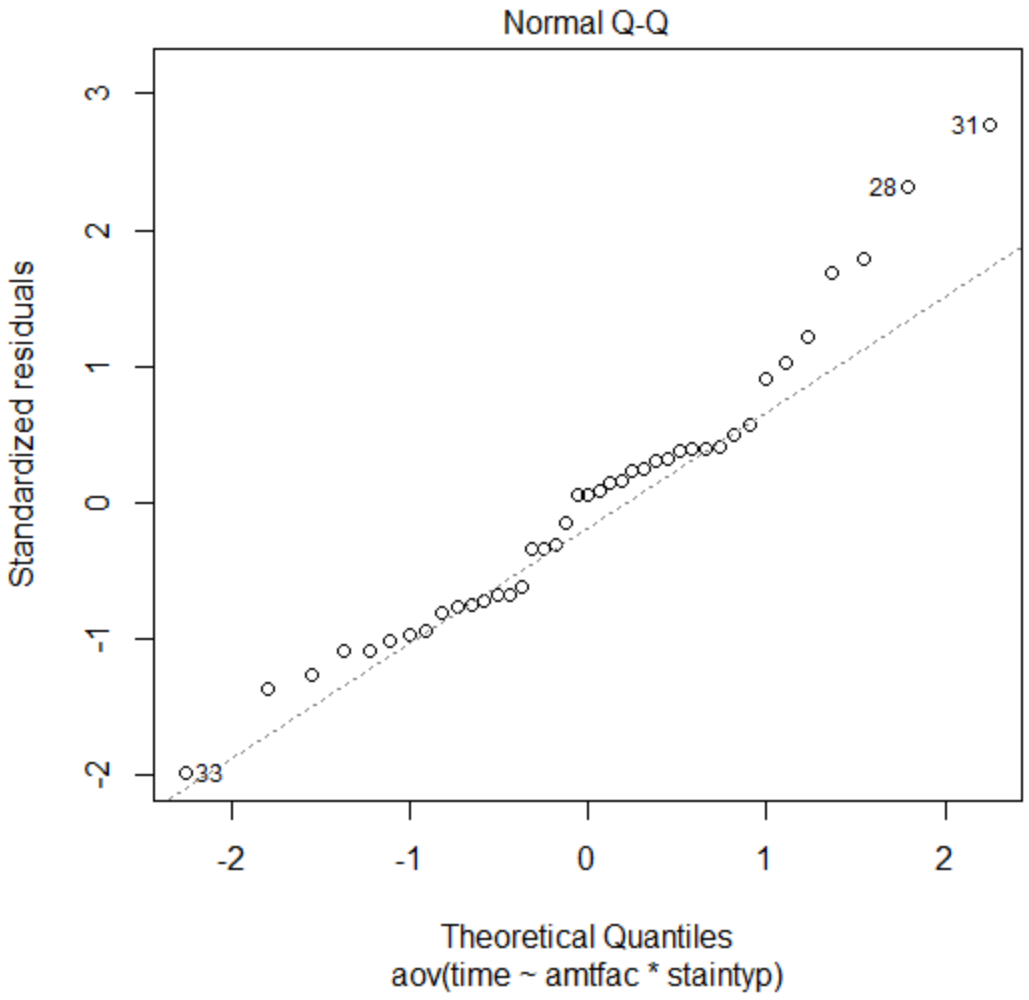

```
> lsmip(m1,staintyp~amtfac)
```




```
>
> ######PARTS 3 & 4
>
> # check resids versus fitted plot and normal QQ plot of resids
> dev.off(which=dev.cur())
null device
          1
> plot(m1,which=1)
```



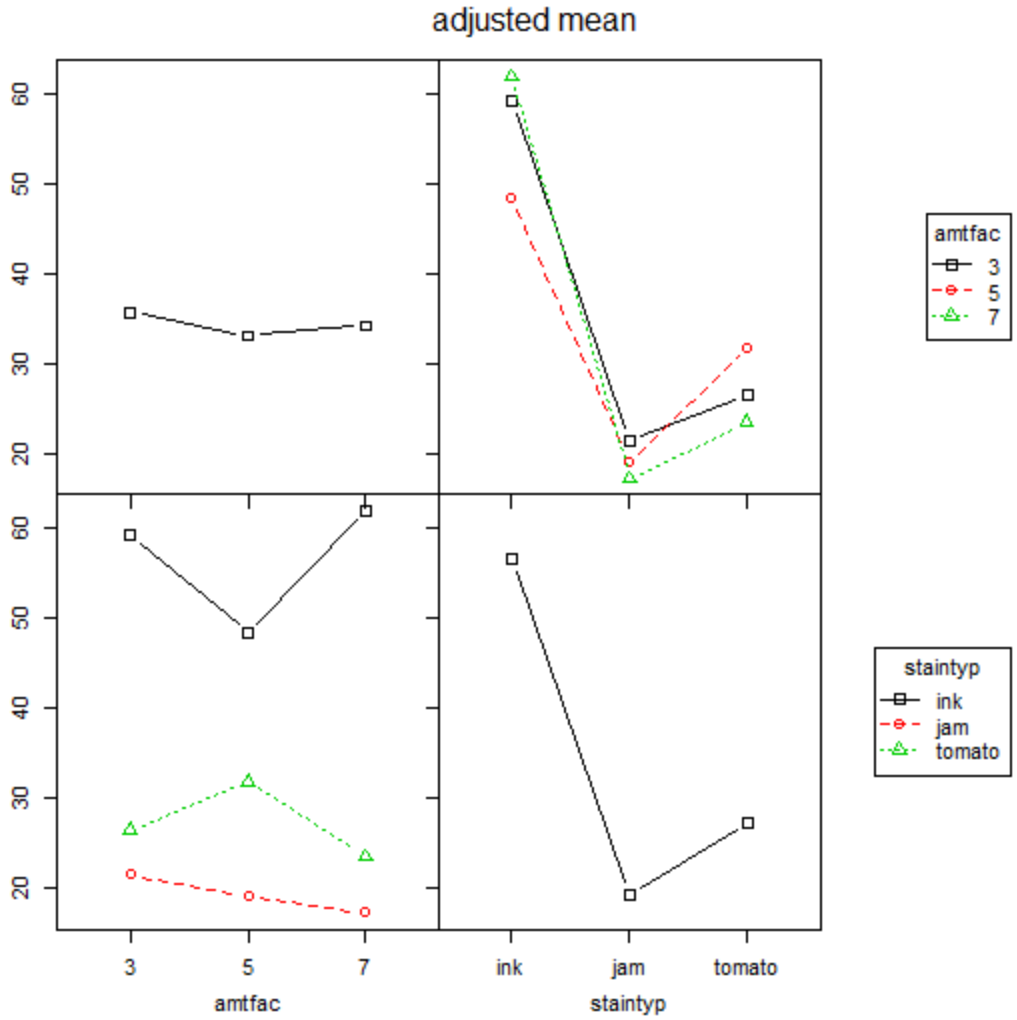

```
> plot(m1,which=2)
```



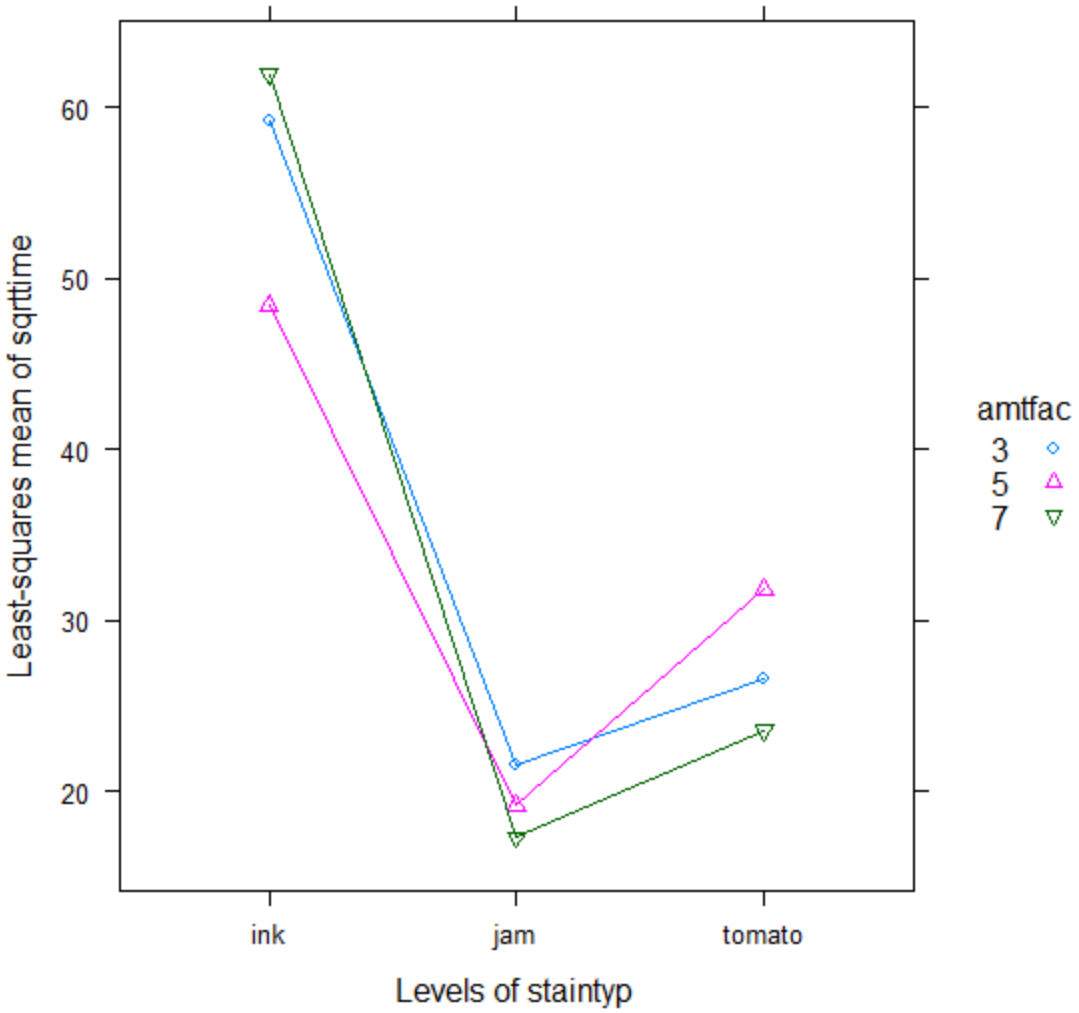

```
>
>
> # looks like non-constant variance. Use regression method to estimate
> # transformation:
>
> # to get summary statistics (mean, sd) by treatment we can use the
> # summaryBy function in the doBy package
>
> (sumStats <- summaryBy(time ~ amtfac + staintyp, data = bleach,
+   FUN = function(x) { c(m = mean(x), s = sd(x)) } ) )
  amtfac staintyp    time.m    time.s
1      3      ink 3508.2500 325.91244
2      3      jam  474.8000 137.04452
3      3   tomato  717.4000 212.85277
4      5      ink 2350.8000 305.93659
5      5      jam  370.7500  71.07449
6      5   tomato 1036.4000 361.91339
7      7      ink 3844.2000 479.84550
8      7      jam  312.4000 126.31825
9      7   tomato  570.6667 217.30700
> # produces time.m and time.s for each
> # combination of the levels of amtfac and staintyp
>
> #regress log(SD) on log(mean) to estimate power transformation
```

```
> lm(I(log(time.s)) ~ I(log(time.m)),data=sumStats)

Call:
lm(formula = I(log(time.s)) ~ I(log(time.m)), data = sumStats)

Coefficients:
   (Intercept)  I(log(time.m))
        1.6941          0.5349

>
> # slope is .53 which is approximately 0.5, so this suggests
> # taking y to the 1-.5=0.5 power (sqrt(y))
>
> # creat sqrttime=sqrt(time) in data bleach
> bleach$sqrttime<-sqrt(bleach$tim)
>
> #  fit Model 2: 2-way ANOVA model after taking sqrt transformation for time
> m2 <- aov(sqrttime~amtfac*staintyp,data=bleach)
> summary(m2)
                Df Sum Sq Mean Sq F value   Pr(>F)
amtfac           2     29      14   1.009    0.376
staintyp         2  10446    5223 364.473  < 2e-16 ***
amtfac:staintyp  4    640     160  11.157 8.83e-06 ***
Residuals       32    459      14
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # Type III anova table:
> Anova(m2,type=3)
Anova Table (Type III tests)

Response: sqrttime
                Sum Sq Df   F value    Pr(>F)
(Intercept)      47125  1 3288.4074 < 2.2e-16 ***
amtfac              48  2    1.6702    0.2042
staintyp         10504  2  366.5018 < 2.2e-16 ***
amtfac:staintyp    640  4   11.1566  8.83e-06 ***
Residuals          459 32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> # Now get the marginal and joint lsmeans:
> lsmeans(m2, specs= list(~staintyp,~amtfac,~amtfac:staintyp))
$`staintyp lsmeans`
 staintyp   lsmean       SE df lower.CL upper.CL
      ink 56.49836 1.017349 32 54.42609 58.57063
      jam 19.36178 1.017349 32 17.28951 21.43405
   tomato 27.32340 1.080598 32 25.12229 29.52450

$`amtfac lsmeans`
 amtfac   lsmean       SE df lower.CL upper.CL
      3 35.76513 1.017349 32 33.69285 37.83740
      5 33.14307 1.017349 32 31.07080 35.21534
      7 34.27534 1.080598 32 32.07423 36.47644

$`amtfac:staintyp lsmeans`
 amtfac staintyp   lsmean       SE df lower.CL upper.CL
      3      ink 59.18301 1.892799 32 55.32750 63.03851
```

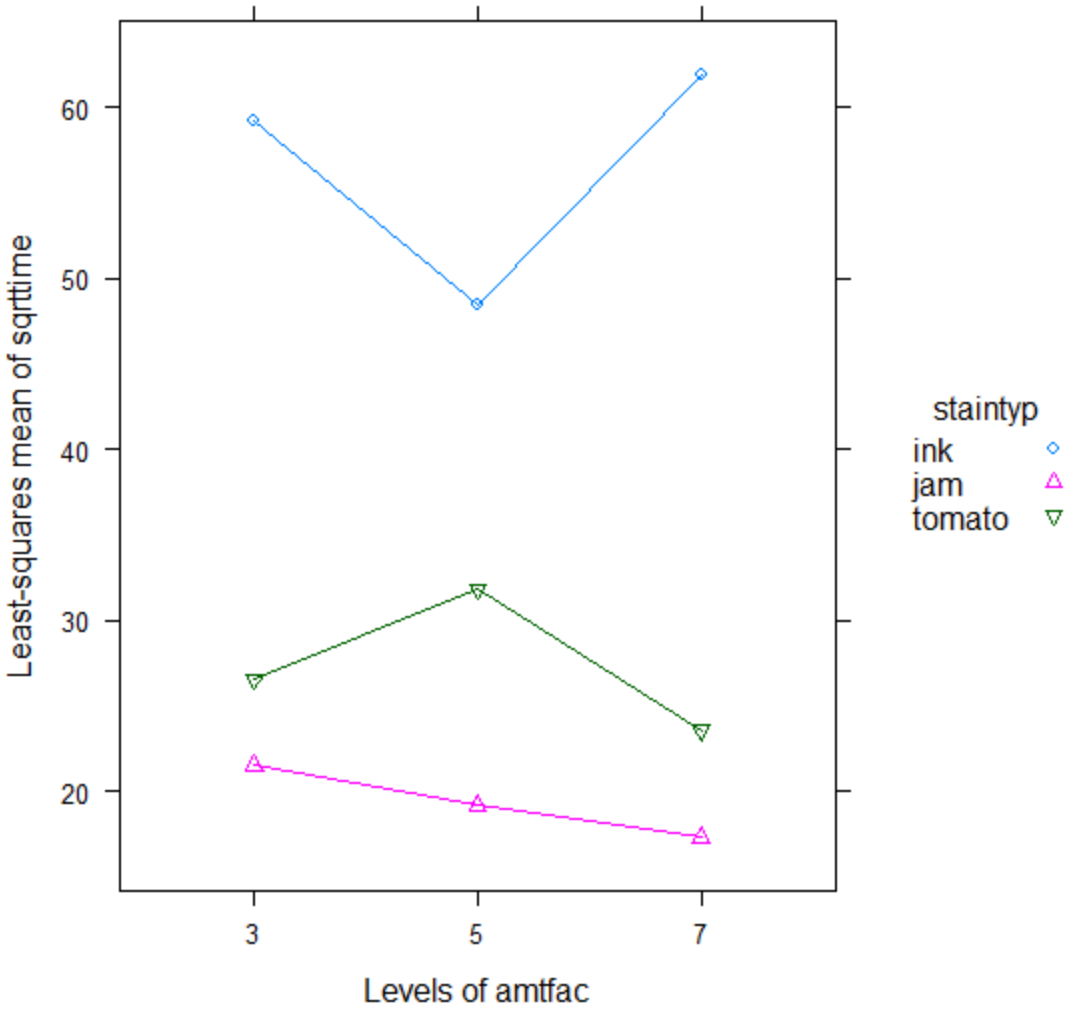
```
      5      ink 48.40488 1.692971 32 44.95641 51.85335
      7      ink 61.90719 1.692971 32 58.45872 65.35566
      3      jam 21.56007 1.692971 32 18.11160 25.00854
      5      jam 19.18658 1.892799 32 15.33108 23.04209
      7      jam 17.33869 1.692971 32 13.89022 20.78716
      3   tomato 26.55230 1.692971 32 23.10383 30.00077
      5   tomato 31.83775 1.692971 32 28.38929 35.28622
      7   tomato 23.58014 2.185616 32 19.12818 28.03209

Warning messages:
1: In lsmeans(m2, specs = list(~staintyp, ~amtfac, ~amtfac:staintyp)) :
  lsmeans of staintyp may be misleading due to interaction with other
predictor(s)
2: In lsmeans(m2, specs = list(~staintyp, ~amtfac, ~amtfac:staintyp)) :
  lsmeans of amtfac may be misleading due to interaction with other predictor(s)
>
> # get profile plots (uses package phia)
> ( trtmns2 <- interactionMeans(m2) )
  amtfac staintyp adjusted mean
1      3      ink      59.18301
2      5      ink      48.40488
3      7      ink      61.90719
4      3      jam      21.56007
5      5      jam      19.18658
6      7      jam      17.33869
7      3   tomato      26.55230
8      5   tomato      31.83775
9      7   tomato      23.58014
> plot(trtmns2)
```



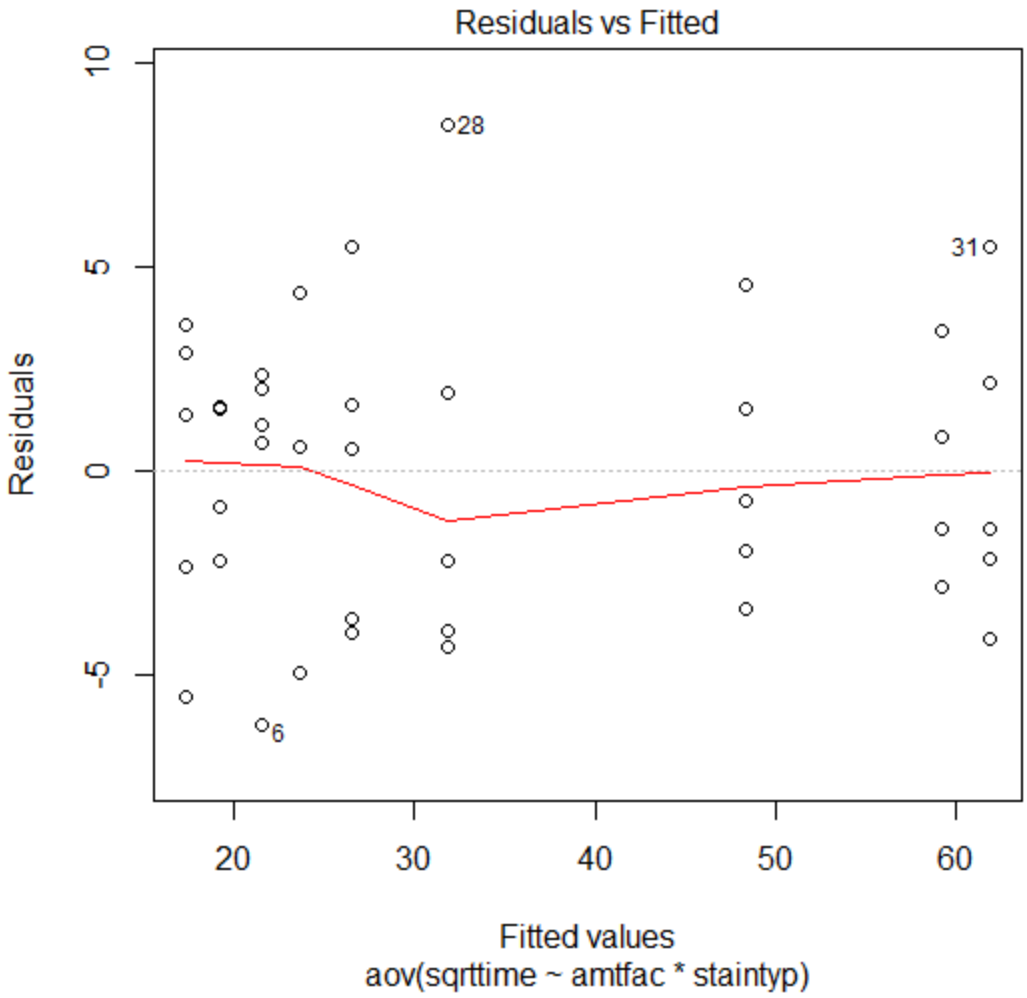

```
>
> # alternatively can use function lsmip from lsmeans package
>
> lsmip(m2,amtfac~staintyp)
```



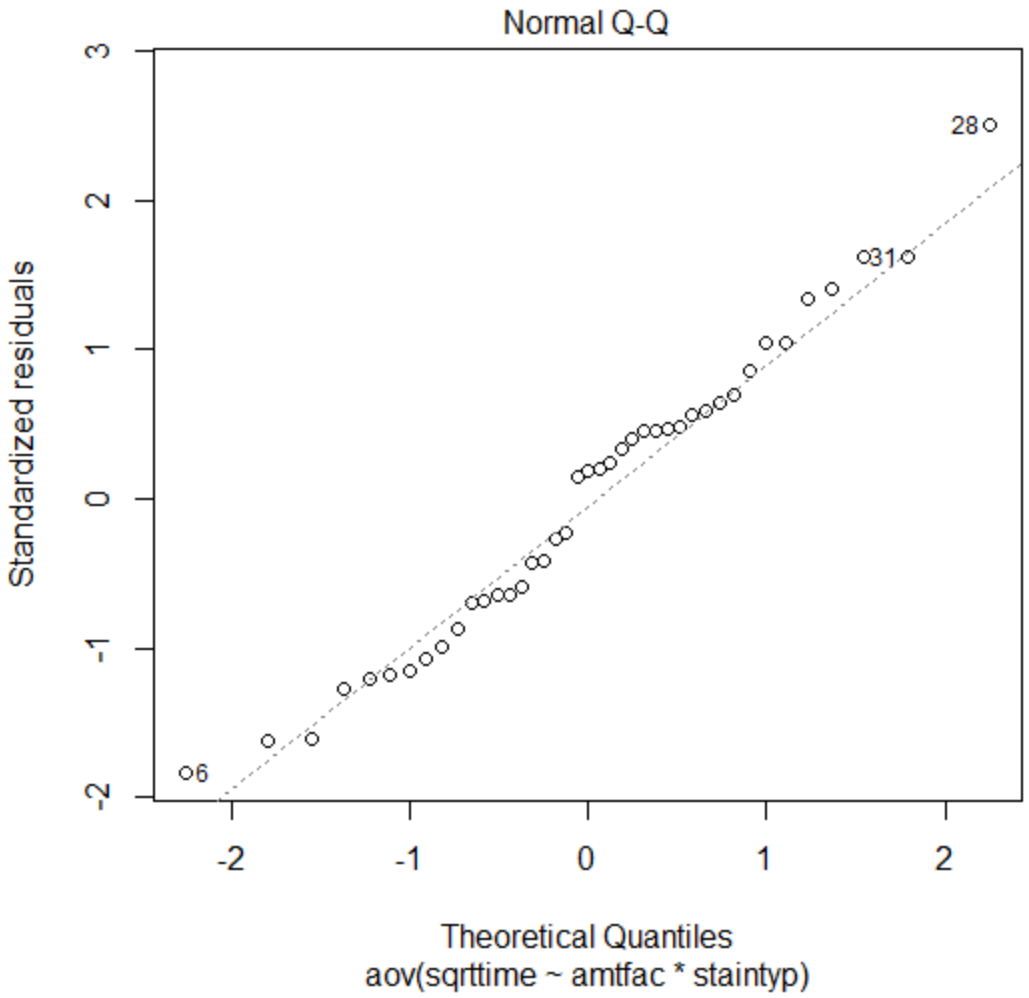

```
> lsmip(m2,staintyp~amtfac)
```




```
>
> # check resids versus fitted plot and normal QQ plot of resids
> dev.off(which=dev.cur())
null device
          1
> plot(m2,which=1)
```




```
> plot(m2,which=2)
```




```
>
> # These look better.
>
> ######PART 5
>
> # below are the linear and quadratic contrasts for amt of bleach within each
> # stain type:
> c1<-c(-1,0,1,0,0,0,0,0,0)
> c2<-c(1,-2,1,0,0,0,0,0,0)
> c3<-c(0,0,0,-1,0,1,0,0,0)
> c4<-c(0,0,0,1,-2,1,0,0,0)
> c5<-c(0,0,0,0,0,0,-1,0,1)
> c6<-c(0,0,0,0,0,0,-1,2,1)
>
> lsmeans(m2,specs=lsm~amtfac:staintyp,
+   contr=list(lsm=list(linear.bleach.stain1=c1,
+                       nonlinear.bleach.stain1=c2,
+                       linear.bleach.stain2=c3,
+                       nonlinear.bleach.stain2=c4,
+                       linear.bleach.stain3=c5,
+                       nonlinear.bleach.stain3=c6)))
$`amtfac:staintyp lsmeans`
 amtfac staintyp   lsmean       SE df lower.CL upper.CL
      3      ink 59.18301 1.892799 32 55.32750 63.03851
```

```
      5      ink 48.40488 1.692971 32 44.95641 51.85335
      7      ink 61.90719 1.692971 32 58.45872 65.35566
      3      jam 21.56007 1.692971 32 18.11160 25.00854
      5      jam 19.18658 1.892799 32 15.33108 23.04209
      7      jam 17.33869 1.692971 32 13.89022 20.78716
      3   tomato 26.55230 1.692971 32 23.10383 30.00077
      5   tomato 31.83775 1.692971 32 28.38929 35.28622
      7   tomato 23.58014 2.185616 32 19.12818 28.03209

$`amtfac:staintyp lsm`
                          estimate       SE df  t.ratio p.value
linear.bleach.stain1     2.7241867 2.539457 32  1.07274 0.29141
nonlinear.bleach.stain1 24.2804466 4.232428 32  5.73677 0.00000
linear.bleach.stain2    -4.2213813 2.394223 32 -1.76315 0.08742
nonlinear.bleach.stain2  0.5255913 4.479181 32  0.11734 0.90732
linear.bleach.stain3    -2.9721692 2.764610 32 -1.07508 0.29038
nonlinear.bleach.stain3 60.7033400 4.371233 32 13.88701 0.00000
    p values are not adjusted

>
> options(op) # reset contrasts to their original values
>
>
```