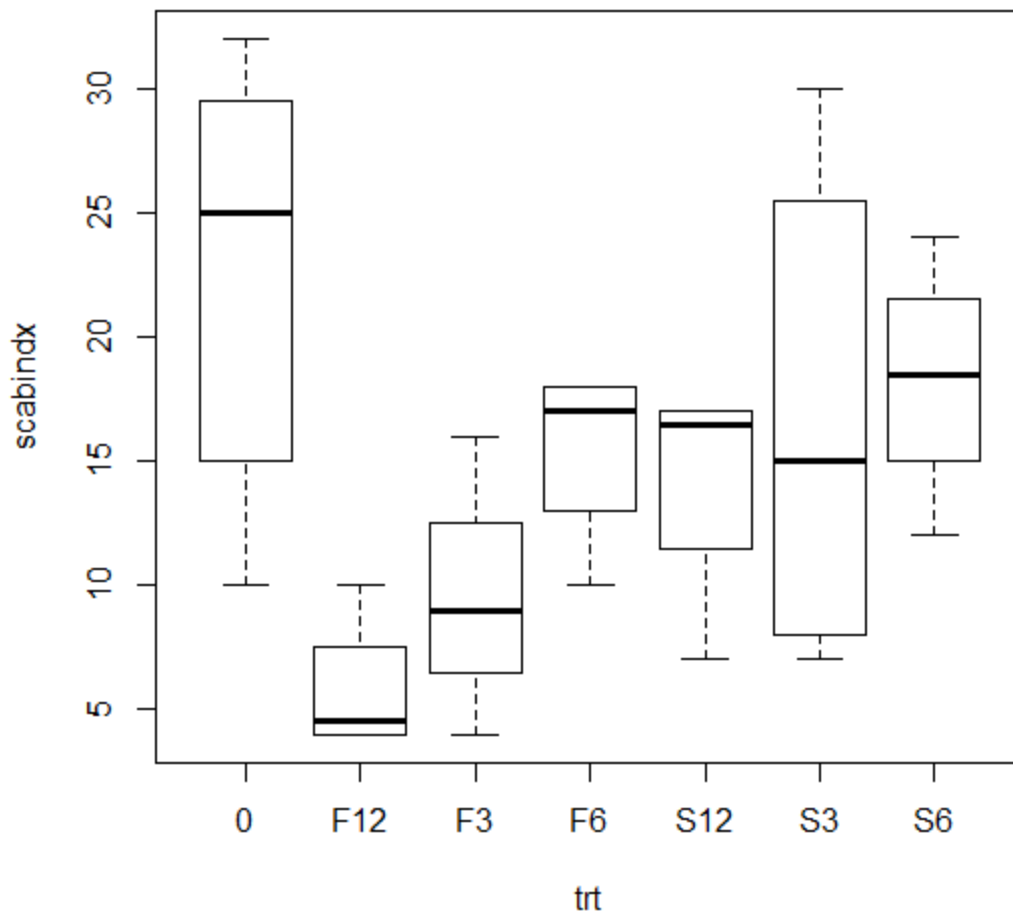


Output from scab1.R

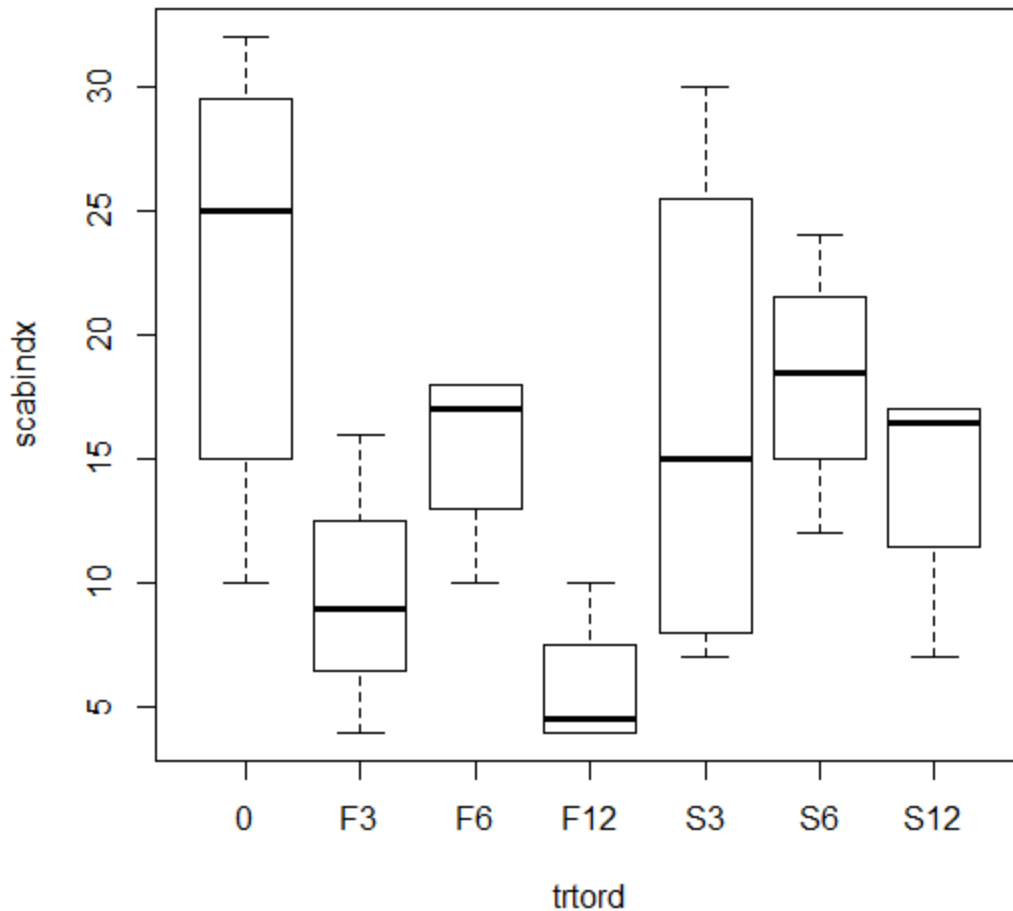
```
> # scab1.R
>
>
> scabdata <- read.table(file="scab.dat",header=T) # read the data from an
>                                                    # external file and grab the
>                                                    # variable names from the
>                                                    # 1st line of that file.
>                                                    # The data are placed into
>                                                    # a "data frame", which I've
>                                                    # called scabdata
> head(scabdata) #look at the first few rows of scabdata
  trt scabindx
1   0         10
2   0         12
3   0         18
4   0         30
5   0         32
6   0         24
>
> #side-by-side boxplots. Notice the ordering of the treatments
> plot(scabindx~trt,data=scabdata)
> title(main="Boxplots of scab index for each treatment")
```

Boxplots of scab index for each treatment



```
>
> is.factor(scabdata$trt)
[1] TRUE
> is.ordered(scabdata$trt)
[1] FALSE
> scabdata$trtord <- ordered(scabdata$trt,
+ levels=c("0", "F3", "F6", "F12", "S3", "S6", "S12"))
>
> #side-by-side boxplots (reordered treatments)
> plot(scabindx~trtord,data=scabdata)
> title(main="Boxplots of scab index for each treatment")
```

Boxplots of scab index for each treatment



```
>
>
> # Set "treatment" contrasts for ordered factors (for this application)
> # instead of the default, which is orthogonal polynomial contrasts for
> # ordered factors (we'll talk about what this means later)
> op <- options(contrasts=c("contr.treatment", "contr.treatment"))
> options()$contrasts # now you can see that they've been changed
[1] "contr.treatment" "contr.treatment"
> op # and op holds the original value of contrasts so that we can reset back to
$contrasts
[1] "contr.treatment" "contr.treatment"

> # the original more easily later.
>
> m1 <- aov(scabindx~trtord,scabdata)
> m1 # this "prints" the fitted model
Call:
aov(formula = scabindx ~ trtord, data = scabdata)
```

Terms:

	trtord	Residuals
Sum of Squares	972.3438	1122.8750
Deg. of Freedom	6	25

```

Residual standard error: 6.701865
Estimated effects may be unbalanced
> summary(m1) # this summarizes the fitted model.
      Df Sum Sq Mean Sq F value Pr(>F)
trtord  6  972.3  162.06   3.608 0.0103 *
Residuals 25 1122.9   44.92
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> coefficients(m1) # this gives the actual fitted model parameters
(Intercept)      trtordF3      trtordF6      trtordF12      trtordS3      trtordS6
      22.625      -13.125      -7.125      -16.875      -5.875      -4.375
      trtordS12
      -8.375
>
> summary.lm(m1) # this summarizes the fitted model treating it as generic lm

```

```

Call:
aov(formula = scabindx ~ trtord, data = scabdata)

```

```

Residuals:
      Min       1Q   Median       3Q      Max
-12.625  -4.844   0.625   3.594  13.250

```

```

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
(Intercept)   22.625     2.369   9.549 8.08e-10 ***
trtordF3      -13.125     4.104  -3.198 0.003734 **
trtordF6       -7.125     4.104  -1.736 0.094858 .
trtordF12     -16.875     4.104  -4.112 0.000372 ***
trtordS3       -5.875     4.104  -1.432 0.164666
trtordS6       -4.375     4.104  -1.066 0.296601
trtordS12     -8.375     4.104  -2.041 0.051977 .

```

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 6.702 on 25 degrees of freedom
Multiple R-squared:  0.4641,    Adjusted R-squared:  0.3355
F-statistic: 3.608 on 6 and 25 DF,  p-value: 0.01026

```

```

> # not necessarily an aov model. This is an easy way to get some
> # additional useful output.
>
>
>
> # now estimate the mean for each treatment:
> lsmeans(m1, specs = ~ trtord)
$`trtord lsmeans`
      trtord lsmean      SE df  lower.CL upper.CL
      0 22.625 2.369467 25 17.744991 27.50501
      F3  9.500 3.350933 25  2.598625 16.40138
      F6 15.500 3.350933 25  8.598625 22.40138
      F12  5.750 3.350933 25 -1.151375 12.65138
      S3 16.750 3.350933 25  9.848625 23.65138
      S6 18.250 3.350933 25 11.348625 25.15138
      S12 14.250 3.350933 25  7.348625 21.15138

```

```

>

```

```

>
> # Contrasts in the model paramters can be tested using the linearHypothesis
> # function from the car package. This is useful, but sometimes a more
> # convenient alternative is to define the corresponding contrast in the
> # lsmeans and use the lsmeans() function to test the contrast.
>
> ( c1 <- c(6,-1,-1,-1,-1,-1,-1) )
[1] 6 -1 -1 -1 -1 -1 -1
> ( c2 <- c(0,1,1,1,-1,-1,-1) )
[1] 0 1 1 1 -1 -1 -1
> c3 <- c(0,2,-1,-1,2,-1,-1)
> c4 <- c(0,0,-1,1,0,-1,1)
> c5 <- c2*c3
> c6 <- c2*c4
>
> lsmeans(m1, specs = lsm ~ trtord,
+   contr=list(lsm=list( ctrl.vs.trted=c1/6, fall.vs.sp=c2/3 ) ) )
$`trtord lsmeans`
  trtord lsmean      SE df  lower.CL upper.CL
    0 22.625 2.369467 25 17.744991 27.50501
   F3  9.500 3.350933 25  2.598625 16.40138
   F6 15.500 3.350933 25  8.598625 22.40138
  F12  5.750 3.350933 25 -1.151375 12.65138
   S3 16.750 3.350933 25  9.848625 23.65138
   S6 18.250 3.350933 25 11.348625 25.15138
  S12 14.250 3.350933 25  7.348625 21.15138

$`trtord lsm`
      estimate      SE df  t.ratio p.value
ctrl.vs.trted  9.291667 2.736025 25  3.39605 0.00229
fall.vs.sp    -6.166667 2.736025 25 -2.25388 0.03322
  p values are not adjusted

>
> # The above contrasts can be tested with the linearHypothesis function.
> # It is convenient to switch to a cell-means model first, though, to
> # make the specification of the contrasts easier
>
> m1a <- lm(scabindx~trtord-1,scabdata) # the "-1" removes the intercept
> # now print out the coefficients from this fitted model, which are the
> # estimated treatment means
> m1a

Call:
lm(formula = scabindx ~ trtord - 1, data = scabdata)

Coefficients:
  trtord0  trtordF3  trtordF6  trtordF12  trtordS3  trtordS6  trtordS12
    22.62     9.50    15.50     5.75    16.75    18.25    14.25

>
> linearHypothesis(m1a,hypothesis.matrix=c1) # ctrl vs treated
Linear hypothesis test

Hypothesis:
6 trtord0 - trtordF3 - trtordF6 - trtordF12 - trtordS3 - trtordS6 - trtordS12 = 0

```

```
Model 1: restricted model
Model 2: scabindx ~ trtord - 1
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	26	1640.9				
2	25	1122.9	1	518.01	11.533	0.002289 **

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> linearHypothesis(m1a,hypothesis.matrix=c2) # fall vs spring
Linear hypothesis test
```

Hypothesis:

$$\text{trtordF3} + \text{trtordF6} + \text{trtordF12} - \text{trtordS3} - \text{trtordS6} - \text{trtordS12} = 0$$

```
Model 1: restricted model
Model 2: scabindx ~ trtord - 1
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	26	1351.0				
2	25	1122.9	1	228.17	5.08	0.03322 *

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> linearHypothesis(m1a,hypothesis.matrix=rbind(c3,c4)) # dose
Linear hypothesis test
```

Hypothesis:

$$2 \text{ trtordF3} - \text{trtordF6} - \text{trtordF12} + 2 \text{ trtordS3} - \text{trtordS6} - \text{trtordS12} = 0$$
$$- \text{trtordF6} + \text{trtordF12} - \text{trtordS6} + \text{trtordS12} = 0$$

```
Model 1: restricted model
Model 2: scabindx ~ trtord - 1
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	27	1312.5				
2	25	1122.9	2	189.58	2.1105	0.1423

```
> linearHypothesis(m1a,hypothesis.matrix=rbind(c5,c6)) # dose*season
Linear hypothesis test
```

Hypothesis:

$$2 \text{ trtordF3} - \text{trtordF6} - \text{trtordF12} - 2 \text{ trtordS3} + \text{trtordS6} + \text{trtordS12} = 0$$
$$- \text{trtordF6} + \text{trtordF12} + \text{trtordS6} - \text{trtordS12} = 0$$

```
Model 1: restricted model
Model 2: scabindx ~ trtord - 1
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	27	1159.5				
2	25	1122.9	2	36.583	0.4073	0.6698

>

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

>