Equivalence of the Reduced versus Full Model F test and the F test of C beta = d

	Storage Temperature	
Storage Time	20^oC	30^oC
3 months	2 5	9 12 15
6 months	6 6 7 7	16

```
time=factor(rep(c(3,6),each=5))
temp=factor(rep(c(20,30,20,30),c(2,3,4,1)))
y=c(2,5,9,12,15,6,6,7,7,16)
d=data.frame(time,temp,y)
```

```
d
   time temp  y
1     3   20  2
2     3   20  5
3     3   30  9
4     3   30 12
5     3   30 15
6     6   20  6
7     6   20  6
8     6   20  7
9     6   20  7
10    6   30 16

full=lm(y~time+temp+time:temp,data=d)
```

```
model.matrix(full)
```

	(Intercept)	time6	temp30	time6:temp30
1	1	0	0	0
2	1	0	0	0
3	1	0	1	0
4	1	0	1	0
5	1	0	1	0
6	1	1	0	0
7	1	1	0	0
8	1	1	0	0
9	1	1	0	0
10	1	1	1	1

```
coef(full)
 (Intercept)         time6         temp30 time6:temp30
         3.5           3.0            8.5           1.0

#########################################################
#            temp 20   temp 30
#            ------------------------
# time 3    mu         mu+temp30
#
# time 6    mu+time6   mu+time6+temp30+time6:temp30
#
#########################################################
```

```
test=function(lmout,C,d=0){
  b=coef(lmout)
  V=vcov(lmout)
  dfn=nrow(C)
  dfd=lmout$df
  Cb.d=C%*%b-d
  Fstat=drop(t(Cb.d)%*%solve(C%*%V%*%t(C))%*%Cb.d/dfn)
  pvalue=1-pf(Fstat,dfn,dfd)
  list(Fstat=Fstat,pvalue=pvalue)
}

Coverall=matrix(c(
0,1,0,0,
0,0,1,0,
0,0,0,1
),nrow=3,byrow=T)
```

```
test(full,Coverall)
$Fstat
[1] 13.53191

$pvalue
[1] 0.004438826

reduced=lm(y~1,data=d)

model.matrix(reduced)
   (Intercept)
1            1
2            1
3            1
4            1
5            1
6            1
7            1
8            1
9            1
10           1
```

```
rvsf=function(reduced,full)
{
  sser=deviance(reduced)
  ssef=deviance(full)
  dfer=reduced$df
  dfef=full$df
  dfn=dfer-dfef
  Fstat=(sser-ssef)/dfn/
        (ssef/dfef)
  pvalue=1-pf(Fstat,dfn,dfef)
  list(Fstat=Fstat,dfn=dfn,dfd=dfef,pvalue=pvalue)
}

rvsf(reduced,full)
$Fstat
[1] 13.53191

$dfn
[1] 3
```

```
$dfd
[1] 6

$pvalue
[1] 0.004438826

anova(reduced,full)
Analysis of Variance Table

Model 1: y ~ 1
Model 2: y ~ time temp time:temp
  Res.Df   RSS Df Sum of Sq      F   Pr(>F)
1      9 182.5
2      6  23.5  3       159 13.532 0.004439 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Cinteraction=matrix(c(
0,0,0,1
),nrow=1,byrow=T)

test(full,Cinteraction)
$Fstat
[1] 0.1225532

$pvalue
[1] 0.7382431

anova(full)
Analysis of Variance Table
Response: y
           Df Sum Sq Mean Sq F value   Pr(>F)
time        1   0.10   0.100  0.0255 0.878292
temp        1 158.42 158.420 40.4477 0.000709 ***
time:temp   1   0.48   0.480  0.1226 0.738243
Residuals   6  23.50   3.917
```

```
summary(full)

Call:
lm(formula = y ~ time temp time:temp, data = d)

Residuals:
       Min          1Q      Median          3Q         Max
-3.000e+00 -5.000e-01 -7.606e-17  5.000e-01  3.000e+00

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     3.500      1.399   2.501  0.04646 *
time6           3.000      1.714   1.750  0.13062
temp30          8.500      1.807   4.705  0.00331 **
time6:temp30    1.000      2.857   0.350  0.73824
---

Residual standard error: 1.979 on 6 degrees of freedom
Multiple R-squared: 0.8712,     Adjusted R-squared:
0.8068
F-statistic: 13.53 on 3 and 6 DF,  p-value: 0.004439
```

```
additive=lm(y~time+temp,data=d)
model.matrix(additive)
   (Intercept) time6 temp30
1            1     0      0
2            1     0      0
3            1     0      1
4            1     0      1
5            1     0      1
6            1     1      0
7            1     1      0
8            1     1      0
9            1     1      0
10           1     1      1
```

```
coef(additive)
(Intercept)       time6      temp30
       3.26        3.36        8.90

#####################################
#           temp 20   temp 30
#           ------------------------
# time 3    mu        mu+temp30
#
# time 6    mu+time6  mu+time6+temp30
#
#####################################

rvsf(additive,full)
$Fstat
[1] 0.1225532

$dfn
[1] 1
```

```
$dfd
[1] 6

$pvalue
[1] 0.7382431

anova(additive,full)
Analysis of Variance Table

Model 1: y ~ time temp
Model 2: y ~ time temp time:temp
  Res.Df   RSS Df Sum of Sq      F Pr(>F)
1      7 23.98
2      6 23.50  1      0.48 0.1226 0.7382
```

```
drop1(full,test="F")
Single term deletions

Model:
y ~ time + temp + time:temp
          Df Sum of Sq   RSS    AIC F value  Pr(F)
<none>                  23.50 16.544
time:temp  1      0.48 23.98 14.746  0.1226 0.7382

anova(additive)
Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq F value    Pr(>F)
time       1   0.10   0.100  0.0292 0.8691725
temp       1 158.42 158.420 46.2444 0.0002531 ***
Residuals  7  23.98   3.426
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
drop1(additive,test="F")
Single term deletions

Model:
y ~ time + temp
       Df Sum of Sq    RSS    AIC F value     Pr(F)
<none>                23.98 14.746
time    1     23.52  47.50 19.581  6.8657 0.0343966 *
temp    1    158.42 182.40 33.036 46.2444 0.0002531
***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

anova(lm(y~temp+time,data=d))
Analysis of Variance Table
Response: y
          Df Sum Sq Mean Sq F value   Pr(>F)
temp       1 135.00 135.000 39.4078 0.000413 ***
time       1  23.52  23.520  6.8657 0.034397 *
Residuals  7  23.98   3.426
```

```
#Previously we saw how to test for time main effects
#and temp main effects in the full model by testing
#H0: C beta = d.
#
#It is possible but not as easy to test for these main
#effects using the reduced versus full model approach.
#
#We will use the test for time main effects as an
#example.
#
#We need to find a matrix whose column space allows
#for temp main effects and time-by-temp interaction
#but no time main effects.
#
#It is natural to try the following model
#specification.
```

```
o=lm(y~temp+time:temp,data=d)
model.matrix(o)
   (Intercept) temp30 temp20:time6 temp30:time6
1            1      0             0             0
2            1      0             0             0
3            1      1             0             0
4            1      1             0             0
5            1      1             0             0
6            1      0             1             0
7            1      0             1             0
8            1      0             1             0
9            1      0             1             0
10           1      1             0             1
#Examination of this design matrix shows that the
#cell means are modeled as
#
#           temp 20           temp 30
#           ------------------------------
# time 3    mu                mu+temp30
#
# time 6    mu+temp20:time6   mu+temp30+temp30:time6
```

```
#It is easy to see that this is just the full model
#in which each treatment group is allowed to have
#its own mean.  Thus, we can't use this code to
#obtain our reduced model fit.

#One way to obtain the test for time main effects
#by comparing a reduced and full model is as follows.

full=lm(y~time+temp+time:temp,data=d)

C=matrix(c(
+ 0,1,0,.5
+ ),nrow=1,byrow=T)

B=matrix(c(
+ 1,0,0,0,
+ 0,0,1,0,
+ 0,0,0,1,
+ 0,1,0,.5
+ ),nrow=4,byrow=T)
```

```
#Note that X beta = X B^{-1} B beta.
#
#Let W = X B^{-1} and alpha = B beta.
#
#Then C beta = 0 is equivalent to alpha_4=0.
```

```
W=model.matrix(full)%*%solve(B)

W0=W[,1:3]

newfull=lm(y~W-1)

newreduced=lm(y~W0-1)

anova(newreduced,newfull)
Analysis of Variance Table

Model 1: y ~ W0 - 1
Model 2: y ~ W - 1
  Res.Df   RSS Df Sum of Sq      F  Pr(>F)
1      7 47.02
2      6 23.50  1     23.52 6.0051 0.04975 *
```

```
rvsf(newreduced,newfull)
$Fstat
[1] 6.005106

$dfn
[1] 1

$dfd
[1] 6

$pvalue
[1] 0.04975481

test(full,C)
$Fstat
[1] 6.005106

$pvalue
[1] 0.04975481
```