

# More than one treatment: factorials

Tim Hanson

Department of Statistics  
University of South Carolina

February, 2017

*Modified from originals by Gary W. Oehlert*

# One factor vs. two or more...

So far we have seen a few one factor data sets:

- Temperature for the lifetime of resin bond data.
- $g = 14$  different treatments to control weeds among soybeans; outcome was percent weeds.
- Seeding vs. not seeding clouds and measuring rainfall.

But we also saw some data that effectively had two treatments:

- For the fruit fly lifetimes  $A =$  number of partners (1 or 8), and  $B =$  unreceptive (pregnant) or receptive (virgin).
- For the cheese inoculant data, there were  $A =$  first strain, and  $B =$  second strain.

Experiments with multiple factors can be analyzed using oneway ANOVA as we've done up until now, or they can be analyzed using multiway ANOVA.

Factorial treatment structure is simply the case where treatments are created by combining factors.

These could be Nisin and Vitamin E factors in potential antimicrobials; water/rice ratio and cooking time in steamed rice sensory properties; or intake temperature, intake pressure, injection pressure, and injection timing in fuel efficiency of diesel engines.

In each case, the treatments are the combinations of factor levels.

We will often refer to the factors generically as A, B, C, and so on. Factor A has  $a$  levels; factor B has  $b$  levels; and so on.

With two factors, there are  $g = ab$  treatments; with four factors, it is  $g = abcd$  treatments.

We are still using a completely randomized design with  $N$  units applied to  $g$  treatments, and if you want to, you can ignore the factorial nature when you analyze.

Consider a two-factor design. We have data

$$y_{ijk} = \mu_{ij} + \epsilon_{ijk}$$

where  $i = 1, \dots, a$ ;  $j = 1, \dots, b$ ; and  $k = 1, \dots, n$ .

Here  $i, j$  together index the treatment by factor levels, and  $k$  indexes the replication within each treatment.

Note we have  $n$  and not  $n_{ij}$ . We begin with the case of balanced designs where every treatment has the same number of replications.

## Example where A has $a = 4$ levels and B has $b = 3$ levels

For concreteness, consider a two-factor design with  $a = 4$ ,  $b = 3$ , and  $g = 12$ . You can visualize this as a table of means:

Factor A	Factor B		
	1	2	3
1	$\mu_{11}$	$\mu_{12}$	$\mu_{13}$
2	$\mu_{21}$	$\mu_{22}$	$\mu_{23}$
3	$\mu_{31}$	$\mu_{32}$	$\mu_{33}$
4	$\mu_{41}$	$\mu_{42}$	$\mu_{43}$

We can do oneway ANOVA on these 12 treatment groups, we can do pairwise comparisons, contrasts (with coefficients  $w_{ij}$  summing to 0), etc. Everything we've done to date just carries through.

# Factorial treatment structure

On previous slide define the mean for  $A = i$  and  $B = j$  as

$$\mu_{ij} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij},$$

giving the model

$$y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + \epsilon_{ijk}.$$

- $\alpha_1, \dots, \alpha_a$  are the main effects of A.
- $\beta_1, \dots, \beta_b$  are the main effects of B.
- $\alpha\beta_{ij}$  are the interaction effects between A and B.

A model that includes only main effects but not interaction is called additive.

# Benefits of factorial design

- If an interaction is present, a factorial will allow you to study, estimate, and test it.
- When the interaction is absent, a factorial is more efficient than two designs that study A and B separately. (In the factorial, each data point tells you about A and about B.)



Our definition of interaction is based on statistical modeling, not on scientific principles or considerations. In particular,

The need (or not) for interaction in modeling a data set depends on the scale of the response.

This means that a data set could look interactive on the natural scale, but look additive after transformation, or vice versa.

Interaction is always in the context of scale. Interaction as we use it is a modeling concept, not a scientific concept.

We begin with a two-factor model and then generalize.

When we just thought about  $g$  treatments and used an overall mean plus treatment effects, we had an extra parameter. We solved that via a constraint (e.g.,  $\alpha_1 = 0$  or  $\sum_{i=1}^a \alpha_i = 0$ ).

The standard factorial model has an overall mean, main effects for each factor and interaction effects. We will have a lot of extra parameters and need quite a few constraints to settle things down.

# A slide that will drive you crazy

Overall mean, main effects, interaction effects

$$y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + \epsilon_{ijk}$$

$$i = 1, 2, \dots, a; \quad j = 1, 2, \dots, b; \quad k = 1, 2, \dots, n$$

$$\sum_{i=1}^a \alpha_i = \sum_{j=1}^b \beta_j = \sum_{i=1}^a \alpha\beta_{ij} = \sum_{j=1}^b \alpha\beta_{ij} = 0$$

$$g = ab; \quad N = nab = Ng$$

$$df_A = a - 1; \quad df_B = b - 1; \quad df_{AB} = (a - 1)(b - 1)$$

$$df_E = ab(n - 1) = N - ab = N - g$$

Note that  $(a - 1) + (b - 1) + (a - 1)(b - 1) = ab - 1 = g - 1$ .

# Decomposing the data

$$\begin{aligned} y_{ijk} &= \bar{y}_{\dots} + && \hat{\mu} \\ &(\bar{y}_{i\bullet\bullet} - \bar{y}_{\dots}) + && \hat{\alpha}_i \\ &(\bar{y}_{\bullet j\bullet} - \bar{y}_{\dots}) + && \hat{\beta}_j \\ &(\bar{y}_{ij\bullet} - \bar{y}_{i\bullet\bullet} - \bar{y}_{\bullet j\bullet} + \bar{y}_{\dots}) + && \hat{\alpha}\hat{\beta}_{ij} \\ &(y_{ijk} - \bar{y}_{ij\bullet}) && r_{ijk} \end{aligned}$$

$$\begin{aligned} y_{ijk} &= \bar{y}_{\dots} + && \hat{\mu} \\ &(\bar{y}_{i\bullet\bullet} - \hat{\mu}) + && \hat{\alpha}_i \\ &(\bar{y}_{\bullet j\bullet} - \hat{\mu}) + && \hat{\beta}_j \\ &(\bar{y}_{ij\bullet} - [\hat{\mu} + \hat{\alpha}_i + \hat{\beta}_j]) + && \hat{\alpha}\hat{\beta}_{ij} \\ &(y_{ijk} - [\hat{\mu} + \hat{\alpha}_i + \hat{\beta}_j + \hat{\alpha}\hat{\beta}_{ij}]) && r_{ijk} \end{aligned}$$

For balanced data, the SS decomposition is also easy.

$$\begin{aligned}\sum_{ijk} y_{ijk}^2 &= \sum_{ijk} (\hat{\mu} + \hat{\alpha}_i + \hat{\beta}_j + \hat{\alpha}\hat{\beta}_{ij} + r_{ijk})^2 \\ &= \sum_{ijk} \hat{\mu}^2 + \sum_{ijk} \hat{\alpha}_i^2 + \sum_{ijk} \hat{\beta}_j^2 + \sum_{ijk} \hat{\alpha}\hat{\beta}_{ij}^2 + \sum_{ijk} r_{ijk}^2 \\ &= N\hat{\mu}^2 + \sum_i nb \hat{\alpha}_i^2 + \sum_j na \hat{\beta}_j^2 + \sum_{ij} n \hat{\alpha}\hat{\beta}_{ij}^2 + \sum_{ijk} r_{ijk}^2 \\ &= SS_{Const} + SS_A + SS_B + SS_{AB} + SS_E\end{aligned}$$

Balance lets us go from the first line of the decomposition to the second, because all the cross products add to 0 in the balanced case. Without balance, life is much harder.

$SS_{Const}$  is usually ignored.

# ANOVA table

Source	df	SS	MS	F
A	$a - 1$	$\sum_i nb \widehat{\alpha}_i^2$	$SS_A / (a - 1)$	$MS_A / MS_E$
B	$b - 1$	$\sum_j na \widehat{\beta}_j^2$	$SS_B / (b - 1)$	$MS_B / MS_E$
AB	$(a - 1)(b - 1)$	$\sum_{ij} n \widehat{\alpha\beta}_{ij}^2$	$SS_{AB} / [(a - 1)(b - 1)]$	$MS_{AB} / MS_E$
Error	$N - ab$	$\sum_{ijk} r_{ijk}^2$	$SS_E / (N - ab)$	

If  $H_0 : \alpha_i \equiv 0$  is true,  $MS_A / MS_E$  is F with  $(a - 1)$  and  $N - ab$  df.

If  $H_0 : \beta_j \equiv 0$  is true,  $MS_B / MS_E$  is F with  $(b - 1)$  and  $N - ab$  df.

If  $H_0 : \alpha\beta_{ij} \equiv 0$  is true,  $MS_{AB} / MS_E$  is F with  $(a - 1)(b - 1)$  and  $N - ab$  df.

Reject for big F (and small p-value), but still need to check assumptions.

## What about $n = 1$ , one replication for each $(i, j)$ ?

When  $n = 1$  there the degrees of freedom for error is  $df_E = N - ab = ab - ab = 0$ ; we cannot estimate  $\sigma^2$ . We can also drop the  $k$  subscript as  $y_{ij}$  is the only observation observed for the  $(i, j)$  treatment combination.

In this case we can only fit the additive model

$$y_{ij} = \mu + \alpha_i + \beta_j + \epsilon_{ij}.$$

Interaction plots (coming up) and Tukey's test for additivity are two methods we can use in this situation to gauge whether the additive model is okay.

# Tukey's 1 df test for additivity

Model is

$$y_{ij} = \mu + \alpha_i + \beta_j + D\alpha_i\beta_j + \epsilon_{ij}.$$

Test  $H_0 : \gamma = 0$ , giving additivity, vs. a multiplicative alternative. The alternative is not the usual interaction with  $(a - 1)(b - 1)$  additional parameters, but a simplified version with only one, the  $D$ .

Only need to use Tukey's 1df test when there is no replication, e.g.  $n = 1$ .



## Iron in rat livers

Lynch and Strain (1990) experiment: six treatments studying how milk-based diets and copper supplements affect trace element levels in rat livers. Six treatments were the combinations of three milk-based diets (skim milk protein, whey, or casein) and two copper supplements (low and high levels).

```
irondata=data.frame(iron.in.liver=c(.70,.93,2.11,1.28,1.87,2.53),
diet=factor(rep(c("skim","whey","casein"),2)),
cu=factor(rep(c("control","deficient"),rep(3,2))))
f=lm(iron.in.liver~diet*cu,data=irondata)
anova(f) # n=1; R doesn't let you fit the interaction model
f=lm(iron.in.liver~diet+cu,data=irondata)
anova(f) # assuming additive model okay, diet significant & copper almost
summary(f)
model.effects(f,"diet")
model.effects(f,"cu")
source("http://people.stat.sc.edu/hansont/stat506/tukey.R")
tukeys.add.test(irondata$iron.in.liver,irondata$diet,irondata$cu)
lines(pairwise(f,diet)) # Tukey HSD
```

Accept no (Tukey) interaction at 5% level. Diagnostics...

Data from Moore and McCabe (1999) problem 13.11: modulus of elasticity of wood chips made from three different species of wood (aspen, birch, and maple) cut to two different sizes (.015 inch thick or .025 inch thick). Have replication with  $n = 3$  for each of the 6 treatment combinations.

```
library(cfcdae)
moedata=data.frame(moe=c(308,278,428,398,426,331,
214,534,433,512,231,320,272,158,376,503,322,220),
size=factor(rep(c(.015,.025),9)),
species=factor(rep(c("aspen","birch","maple"),each=6)))
f=lm(moe~species*size,data=moedata)
anova(f)
```

Nothing is significant! Let's look at some diagnostics...

The SS for various terms can also be considered as “improvement” SS for model fit.  
With the model  $A + B + AB$ :

- The SS for A is the improvement in adding A main effects to a constant mean model.
- The SS for B is the improvement in going from a model with just A to an additive model with A and B.
- The SS for AB is the improvement in going from an additive model to the full model with  $g = ab$  treatment means.

# Interaction plots

ANOVA will allow us to determine whether interaction terms in our model are statistically significant, but it won't help us understand the interaction.

A tool for visualizing interaction is the interaction plot.

This is a plot where we put points at  $(i, \bar{y}_{ij\bullet})$  for all  $i, j$  combinations. Then we “connect-the-dots” between adjacent  $(i, \bar{y}_{ij\bullet})$  that share the same  $j$ .

Alternatively, you can reverse the roles of  $i$  and  $j$  and plot the pairs  $(j, \bar{y}_{ij\bullet})$  and then connect the dots between adjacent points that share the same  $i$ .

Interaction plots are approximately parallel (up to error in the sample means) for additive models.

The two directions of plotting often give very different impressions.

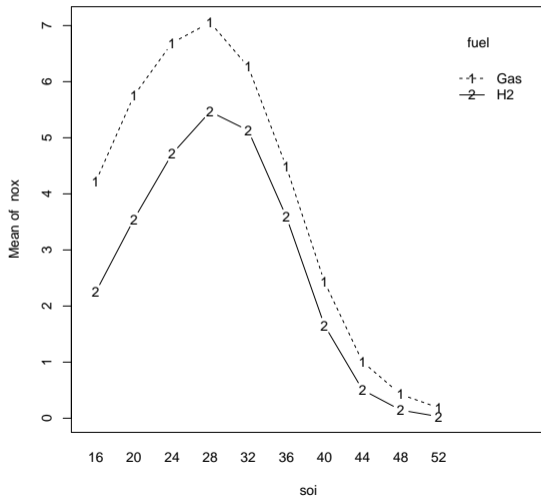
NOX emissions from dual fuel engine (diesel and something, either gasoline or hydrogen) with the start of injection (SOI) varied between 16 and 52 degrees before top of dead center.

Both plots show us a much greater effect of fuel at low SOI, although I like the first one better.

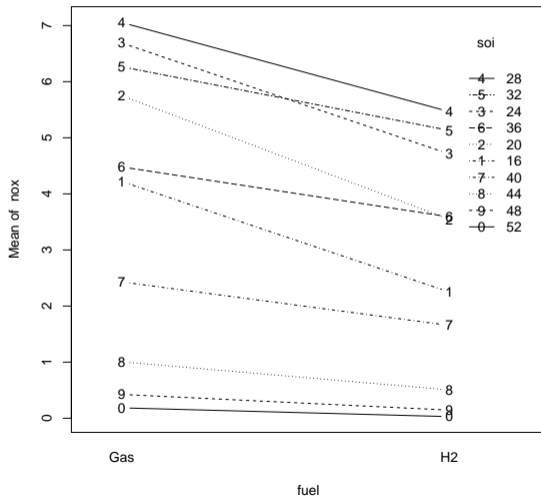
The third plot uses the square root of NOX, which is what is needed to stabilize the residual variance.

Variability in these data is tiny relative to mean differences; everything is highly significant.

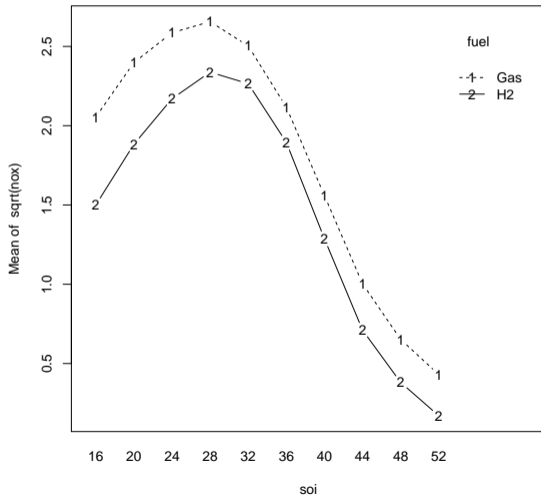
# NOX



# NOX



# $\sqrt{\text{NOX}}$ makes interaction “disappear”





Freezing time of small samples of ice cream mix using different kinds of milk and different amounts of salt in the ice slurry.

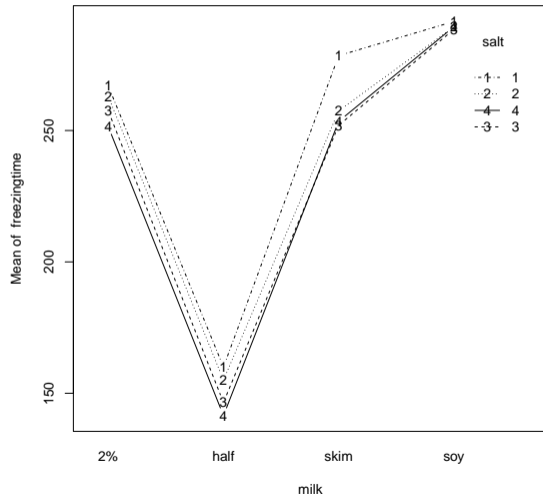
Both main effects are significant, but interaction is not.

Response: freezingtime

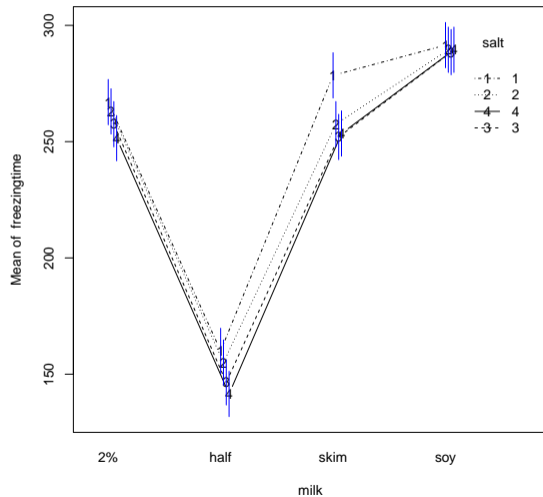
	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
milk	3	89966	29988.6	710.3151	< 2.2e-16	***
salt	3	1104	368.1	8.7192	0.001169	**
milk:salt	9	501	55.6	1.3173	0.301901	
Residuals	16	675	42.2			

Let's look at interaction plots...

# Ice cream: additive?



# Ice cream: error bars help



# Interaction plots in R

```
with(moedata,interactplot(species,size,moe))
with(moedata,interactplot(size,species,moe))
with(moedata, interactplot(size,species,moe,confidence=.95))
# last one shows variability of sample means too large to
# assess whether the "interaction" is real!

with(irondata,interactplot(diet,cu,iron.in.liver))
with(irondata, interactplot(cu,diet,iron.in.liver))
# additivity seems like a good bet visually
```

## Looking at main effects vs. interacting effects

If a factor does not interact without factors you can look at contrasts in the that factor's main effects, e.g. pairwise differences via pairwise.

If A does not interact with B, then you can examine, e.g.  $c = \sum_{i=1}^a w_i \alpha_i$ , or pairwise differences  $\alpha_i - \alpha_j$ , etc. and interpret them as usual. These get at how the mean response changes with levels of A, for a fixed value of B, i.e. conditional on B.

If A and B interact, i.e. the p-value for  $H_0 : \alpha\beta_{ij} = 0$  is small, then you have to look at differences in A at each level of B; these differences will change with B. For example, look at the first interaction plot for the NOX data. Let's see another example...

# Overpredicted programmer-days

$N = 24$  programmers asked to predict how long a big project would take in programmer-days;  $n = 4$  replicates per treatment pair. After project over  $y_{ijk}$  is actual minus predicted programmer-days (prediction errors). Programmers classified by type of experience (A=1 small systems, A=2 small & large); and experience (B=1 is  $< 5$  years, B=2 is 5 – 10 years, B=3 is  $\geq 10$  years).

```
library(cfcdae)
library(lsmmeans) # useful for "slicing" variables that interact
library(lattice)
errors=read.table("http://people.stat.sc.edu/hansont/stat506/errors.txt",header=T)
with(errors,interactplot(exper,years,days))
with(errors,interactplot(years,exper,days))
errors$exper=factor(errors$exper)
errors$years=factor(errors$years)
f1=lm(days~years*exper,data=errors)
anova(f1) # years & exper interact; keep interaction model
```

# Overpredicted programmer-days

Since the two factors significantly interact we cannot fit the additive model and use `pairwise` to examine main effects for experience and main effects for years. Instead we need to examine pairwise differences of one factor separately for each level of the other factor. This will help us estimate the differences we actually see in the interaction plots. The `lsmeans` package allows us to do this easily.

```
# error differences in experience stratified by years
with(errors, interactplot(years, exper, days))
pairs(lsmeans(f1, "exper", by="years"))
confint(pairs(lsmeans(f1, "exper", by="years")))
```

```
# error differences in years stratified by experience
with(errors, interactplot(exper, years, days))
pairs(lsmeans(f1, "years", by="exper"))
confint(pairs(lsmeans(f1, "years", by="exper")))
```

Factorials with more than two factors are just like factorials with two factors, only more so:

- More factors, so more subscripts.
- More factors usually means more data.
- More terms; each additional factor doubles the number of terms in the model.
- More sum-to-zero (or other) constraints on coefficients.
- More confusion with higher order interactions.

But the ideas are just like for two-factor factorials.



## Example: four factor model

$$\begin{aligned} y_{ijklm} = & \mu + \\ & \alpha_i + \beta_j + \gamma_k + \delta_l + \\ & \alpha\beta_{ij} + \alpha\gamma_{ik} + \alpha\delta_{il} + \beta\gamma_{jk} + \beta\delta_{jl} + \gamma\delta_{kl} + \\ & \alpha\beta\gamma_{ijk} + \alpha\beta\delta_{ijl} + \alpha\gamma\delta_{ikl} + \beta\gamma\delta_{jkl} + \\ & \alpha\beta\gamma\delta_{ijkl} + \\ & \epsilon_{ijklm} \end{aligned}$$

$\beta\gamma_{jk}$ : how B effects change across levels of C (or vice versa).

$\beta\gamma\delta_{jkl}$ : how the BC interaction changes across levels of D (or some other version of two in and one out).

$\alpha\beta\gamma\delta_{ijkl}$ : how the BCD interaction changes across levels of A (or some other version of three in and one out).

## Estimation and SS for balanced data

These terms add to zero across any subscript (32 total zero sum constraints).

For balanced data these terms are estimated by taking the mean in the data for the corresponding subscripts and subtracting out estimates of any “lower order” terms. So for  $\widehat{\alpha\beta\gamma}_{ijk}$  you would use

$$\widehat{\alpha\beta\gamma}_{ijk} = \bar{y}_{ijk\bullet\bullet} - [\widehat{\mu} + \widehat{\alpha}_i + \widehat{\beta}_j + \widehat{\gamma}_k + \widehat{\alpha\beta}_{ij} + \widehat{\alpha\gamma}_{ik} + \widehat{\beta\gamma}_{jk}]$$

SS are estimated effect squared, times number of units receiving the effect, added over levels.  $SS_{BCD} = \sum_{jkl} na \widehat{\beta\gamma\delta}_{jkl}^2$ .

DF are the product of the levels of factors appearing in the term, each reduced by 1. BCD has  $(b - 1)(c - 1)(d - 1)$  df.

ANOVA has usual columns with MS as SS over DF and F tests for each term as the MS for the term over MS for error. (And you need to be careful in your naming once you get to five factors!)

To test the null hypothesis that all parameters of a given term are zero, compute the p-value for the F statistic from the F distribution with corresponding df.

Check assumptions as usual. Remember that interaction depends on scale.

## Exercise tolerance

Effects of gender (A, 1=male vs. 2=female), body fat % (B, 1=low fat vs. 2=high), and smoking history (C, 1=light smoking vs. 2=heavy) of subjects on exercise tolerance;  $y_{ijkl}$  is minutes of bicycling until fatigue, measured in small study of  $N = 24$  subjects 25–35 years old. There are  $g = 2 \times 2 \times 2 = 8$  treatment groups, so  $n = 3$  replications per gender/fat/smoking combination. Let's fit a full hierarchical threeway interaction model and see if we can carve out unnecessary interactions...

```
library(cfcdae)
library(lsmmeans) # useful for "slicing" variables that interact
tol=read.table("http://people.stat.sc.edu/hansont/stat506/tol.txt",header=T)
tol$gender=factor(tol$gender)
tol$smoking=factor(tol$smoking)
tol$fat=factor(tol$fat)
levels(tol$gender)=c("male","female")
levels(tol$smoking)=c("light","heavy")
levels(tol$fat)=c("low","high")
f1=lm(tol~gender*fat*smoking,data=tol) # hier. model w/ all interactions
anova(f1) # drop gender:fat, gender:smoking, and gender:fat:smoking?
f2=lm(tol~fat*smoking+gender,data=tol)
anova(f2)
```

# Exercise tolerance

```
anova(f2,f1) # reduced model fits okay w/ p=0.44
par(mfrow=c(2,2))

pairwise(f2,gender) # gender difference the same across fat & smoking

par(mfrow=c(1,1))
with(tol,interactplot(fat,smoking,tol))
pairs(lsmmeans(f2,"smoking",by="fat"))
confint(pairs(lsmmeans(f2,"smoking",by="fat"))))

with(tol,interactplot(smoking,fat,tol))
pairs(lsmmeans(f2,"fat",by="smoking"))
confint(pairs(lsmmeans(f2,"fat",by="smoking"))))
```

Only need a two-way interaction between smoking and body fat. This means we can consider the difference between levels of gender (fixing smoking and body fat level), and look at slices of differences in smoking by body fat (fixing gender); or else differences in body fat by smoking (fixing gender).

## Model choice with unbalanced data

Say the sample sizes are not the same across groups; for twoway ANOVA let  $n_{ij}$  be the sample size for the treatment  $A = i$  and  $B = j$ . The easiest approach is to remove non-significant higher order interactions via Type III tests until what you have left is significant; at all stages make sure you have a hierarchical model. This is called “backwards elimination” of variables, a common approach described in many textbooks on regression.

A type III test is simply testing a larger model with one extra variable vs. a smaller model without that variable; exactly like using `anova(f1, f2)` as we did before. The `Anova` function (note the capital ‘A’) in the `car` package performs Type III tests on each variable in the model.

Your textbook and some other textbooks warn against this, because it is a form of “data snooping”. Other textbooks seem fine with it. I like it because it simplifies the model and the interpretation.

A hierarchical model (in our sense) is one where the presence of a term implies the presence of all included terms. For example, presence of ABC in the model would imply the presence of the overall mean, A, B, C, AB, AC, and BC.

It is good statistical practice to use hierarchical models; only very rarely will a non-hierarchical model make sense.

# Unbalanced free plasma leucine data

Problem 10.8. Animal nutrition experiment was conducted to study the effects of protein in the diet on the level of leucine in the plasma of pigs. Pigs randomly assigned to one of twelve treatments: combinations of protein source (fish meal, soybean meal, and dried skim milk) and protein concentration in the diet (9, 12, 15, or 18 percent). The response is the free plasma leucine level in mcg/ml (Windels, 1964).

```
library(cfcdae); library(MASS); library(lsmmeans); library(car)
d=read.table("http://users.stat.umn.edu/~gary/book/fcdae.data/pr10.8",header=T)
colnames(d)=c("source","percent","conc")
d$source=factor(d$source); d$percent=factor(d$percent)
levels(d$source)=c("fish meal","soybean meal","dried skim milk")
levels(d$percent)=c("9%","12%","15%","18%")
attach(d)
f=lm(conc~source*percent)
Anova(f,type=3) # gives different results than from anova(f)
par(mfrow=c(2,2)); plot(f) # yikes!
par(mfrow=c(1,1))
boxcox(conc~source*percent) # log-transformation
lconc=log(conc) # could also use 1/sqrt(conc)
f=lm(lconc~source*percent)
Anova(f,type=3)
```



Interaction needed so we concentrate on slices.

```
par(mfrow=c(2,2))  
plot(f) # better but interaction still needed
```

```
par(mfrow=c(1,1))  
with(d,interactplot(source,percent,lconc))  
pairs(lsmmeans(f,"percent",by="source"))  
confint(pairs(lsmmeans(f,"percent",by="source")))
```

```
with(d,interactplot(percent,source,lconc))  
pairs(lsmmeans(f,"source",by="percent"))  
confint(pairs(lsmmeans(f,"source",by="percent")))
```

# Case hardening

Experiment on case hardening of lightweight shafts machined from alloy bars run to study effects of amount of chemical agent added to molten alloy (A=low or high), temperature of the hardening process (B=low or high), and time duration of the hardening process (C=low or high). Hardness measured in Brinell units. It is of interest to determine the effect of the three factors on hardness.

```
library(car); library(cfcdae); library(lsmmeans); library(MASS)
d=read.table("http://people.stat.sc.edu/hansont/stat506/case.txt",header=F)
colnames(d)=c("hardness","agent","temp","time","rep")
d$agent=factor(d$agent)
d$temp=factor(d$temp)
d$time=factor(d$time)
f1=lm(hardness~agent*temp*time,data=d) # hier. model w/ all interactions
Anova(f1,type=3) # drop all pairwise and threeway interactions?
f2=lm(hardness~agent+temp+time,data=d)
anova(f2)
anova(f2,f1) # reduced model fits okay w/ p=0.88
```

No interactions! Can concentrate on differences in hardness across each factor holding the other two constant.

```
pairwise(f2,agent)
pairwise(f2,temp)
pairwise(f2,time)
```

## Last example: growth hormone treatment

Synthetic growth hormone administered at a clinical research center to growth hormone deficient, short, prepubescent children. Two factors: gender (male & female) and bone development (severely depressed, moderately depressed, mildly depressed). Response is difference in growth rate during growth hormone treatment vs. previous normal growth rate, in cm/month. Four children dropped out early, creating an imbalanced design. Note that both factors are observational; there are no treatments here. Want to find out how growth hormone affects these children, comparing bone development and gender.

```
diff=c(1.4,2.4,2.2,2.1,1.7,0.7,1.1,2.4,2.5,1.8,2.0,0.5,0.9,1.3)
depress=factor(c(1,1,1,2,2,3,3,1,2,2,2,3,3,3))
gender=factor(c(1,1,1,1,1,1,1,2,2,2,2,2,2,2))
d=data.frame(diff,depress,gender)
levels(d$depress)=c("severe","moderate","mild")
levels(d$gender)=c("male","female")

par(mfrow=c(1,1))
with(d,interactplot(depress,gender,diff))
```

Let's keep going...