STAT 425 Course Notes

Factorial Structure

Spring 2016

Two-Way Factorials

The data set we will examine consists of the survival times of animals randomly allocated to three poisons (I, II, III) and four "treatments" (A, B, C, D).

The arrangement used is called a 3×4 **factorial treatment structure**: each of the 3 poisons appears together with each of the 4 "treatments".

Poisons and "treatments" are the two **factors** in the treatment structure, each having a certain number of **levels**: 3 for poisons, 4 for "treatments".

Each *combination* of a poison and a "treatment" constitutes one *treatment* in the experimental design. For example, one treatment would be: administer Poison III and "Treatment" B.

The factors are (fully) **crossed**, because each combination of their levels is represented, for a total of $3 \times 4 = 12$ treatments.

The design is *replicated* 4 times: There are 4 animals in each of the 12 treatment groups.

Since treatment groups have an equal number of animals, the design is said to be *balanced*.

Survival times (unit: 10 hours) in 3×4 factorial structure

	"Treatment"			
Poison	A	B	C	D
I	.31	.82	.43	.45
	.45	1.10	.45	.71
	.46	.88	.63	.66
	.43	.72	.76	.62
II	.36	.92	.44	.56
	.29	.61	.35	1.02
	.40	.49	.31	.71
	.23	1.24	.40	.38
III	.22	.30	.23	.30
	.21	.37	.25	.36
	.18	.38	.24	.31
	.23	.29	.22	.33

The design is *completely randomized*: The $12 \times 4 = 48$ animals are assigned completely at random to the 12 treatments, subject to having exactly 4 per group.

(Though they define the 12 treatment groups, the factors have *no* influence on how the randomization is performed.)

Suppose both factors (poisons and "treatments") are of equal interest, and it is possible that they *interact* (more later).

Let

Y_{ijk} = response (survival time) for kth animal that receives the ith poison and the jth "treatment"

Then the **(cell) means model** is

$$Y_{ijk} = \mu_{ij} + e_{ijk}$$

where the μ_{ij}'s are the "population" means of the treatment groups, and the errors e_{ijk} are independent and distributed as $N(0, \sigma^2)$ for some unknown σ^2.

More commonly, the means μ_{ij} are decomposed into pieces corresponding to the individual factors and their interaction:

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

where μ is an overall mean, α_i and β_j represent the **main effects** of the factors (poisons and "treatments"), and $\alpha\beta_{ij}$ represents the **interaction effect**.

(Technically, "α_i" represents a *term* in the expansion, and the individual values $\alpha_1, \alpha_2, \alpha_3$ are the *effects*. Similarly for β and $\alpha\beta$.)

(Note that "$\alpha\beta$" should be interpreted as a single symbol, *not* as a product.)

The **factor effects model** is

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + e_{ijk}$$

where the errors e_{ijk} satisfy the same conditions as before.

It is equivalent to the means model, but more useful, since its terms correspond to formula terms commonly used in software like R.

(The effects are not uniquely defined unless some restrictions are imposed on them. In software, such as R, those restrictions are implemented in the *coding* of the factors.)

Let's take a look at the data ...

```
> library(faraway)  # contains rats data set

> summary(rats)
      time         poison   treat
 Min.   :0.1800   I  :16   A:12
 1st Qu.:0.3000   II :16   B:12
 Median :0.4000   III:16   C:12
 Mean   :0.4794            D:12
 3rd Qu.:0.6225
 Max.   :1.2400

> class(rats$poison)
[1] "factor"

> class(rats$treat)
[1] "factor"
```

Note that `poison` and `treat` are *already* factor variables, so it will *not* be necessary to use `factor(poison)` or `factor(treat)` in formula expressions.

Now let's fit the full factor effects model (with interaction) ...

```
> fullmod <- lm(time ~ poison * treat, data=rats)
```

The "*" is a *formula expansion operator* that creates the main effects *and* their interaction(s). In this case,

```
poison * treat  =  poison + treat + poison:treat
```

(Note: `poison:treat` represents a "pure" interaction)

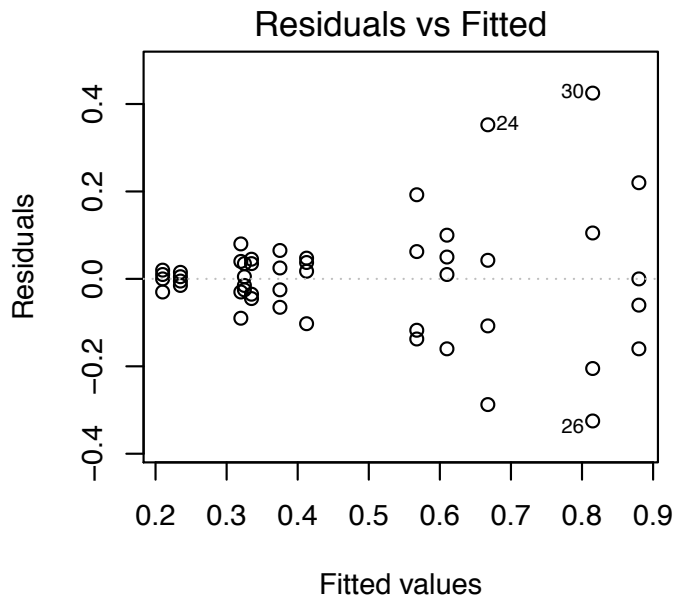
```
> summary(fullmod)
...

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       0.41250    0.07457   5.532 2.94e-06 ***
poisonII         -0.09250    0.10546  -0.877   0.3862
poisonIII        -0.20250    0.10546  -1.920   0.0628 .
treatB            0.46750    0.10546   4.433 8.37e-05 ***
treatC            0.15500    0.10546   1.470   0.1503
treatD            0.19750    0.10546   1.873   0.0692 .
poisonII:treatB   0.02750    0.14914   0.184   0.8547
poisonIII:treatB -0.34250    0.14914  -2.297   0.0276 *
poisonII:treatC  -0.10000    0.14914  -0.671   0.5068
poisonIII:treatC -0.13000    0.14914  -0.872   0.3892
poisonII:treatD   0.15000    0.14914   1.006   0.3212
poisonIII:treatD -0.08250    0.14914  -0.553   0.5836
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.1491 on 36 degrees of freedom
Multiple R-squared:  0.7335, Adjusted R-squared:  0.6521
F-statistic:  9.01 on 11 and 36 DF,  p-value: 1.986e-07
```

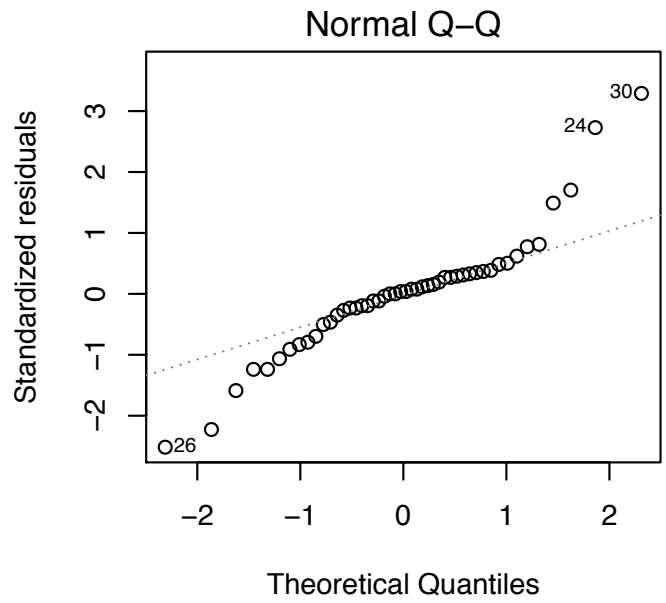
As in one-way ANOVA, the interpretation of these coefficients depends on the *coding* of the factors (see textbook).

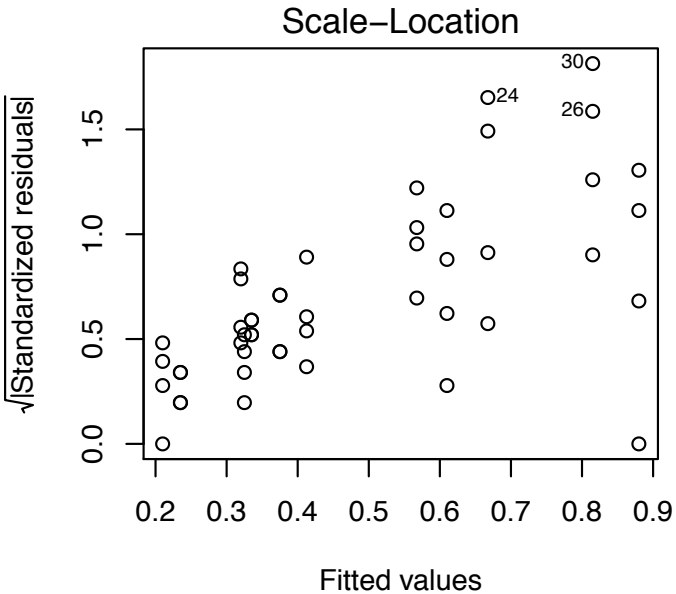
Model assumptions are assessed in the same way as for one-way ANOVA.

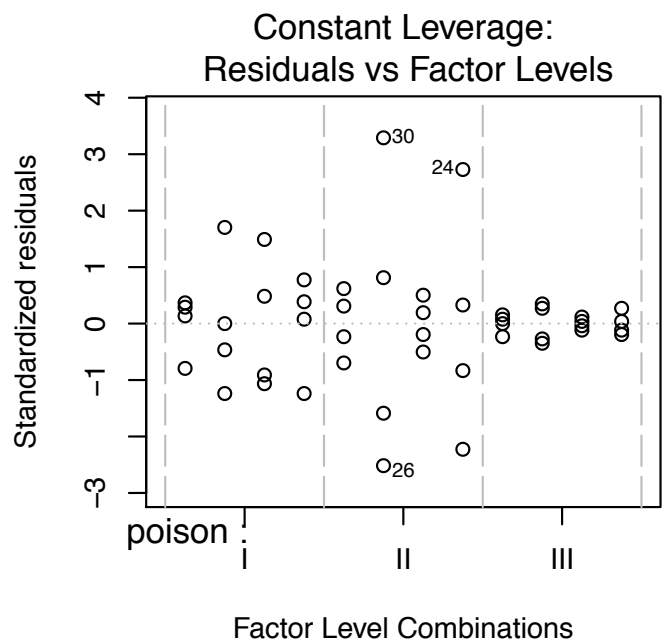
Let's just try using the usual (default) diagnostic plots ...

```
> par(mfrow=c(2,2))
> plot(fullmod, add.smooth=FALSE)
```


Residuals vs Fitted
Residuals
Fitted values
30
24
26


Normal Q−Q
Standardized residuals
Theoretical Quantiles
30
24
26

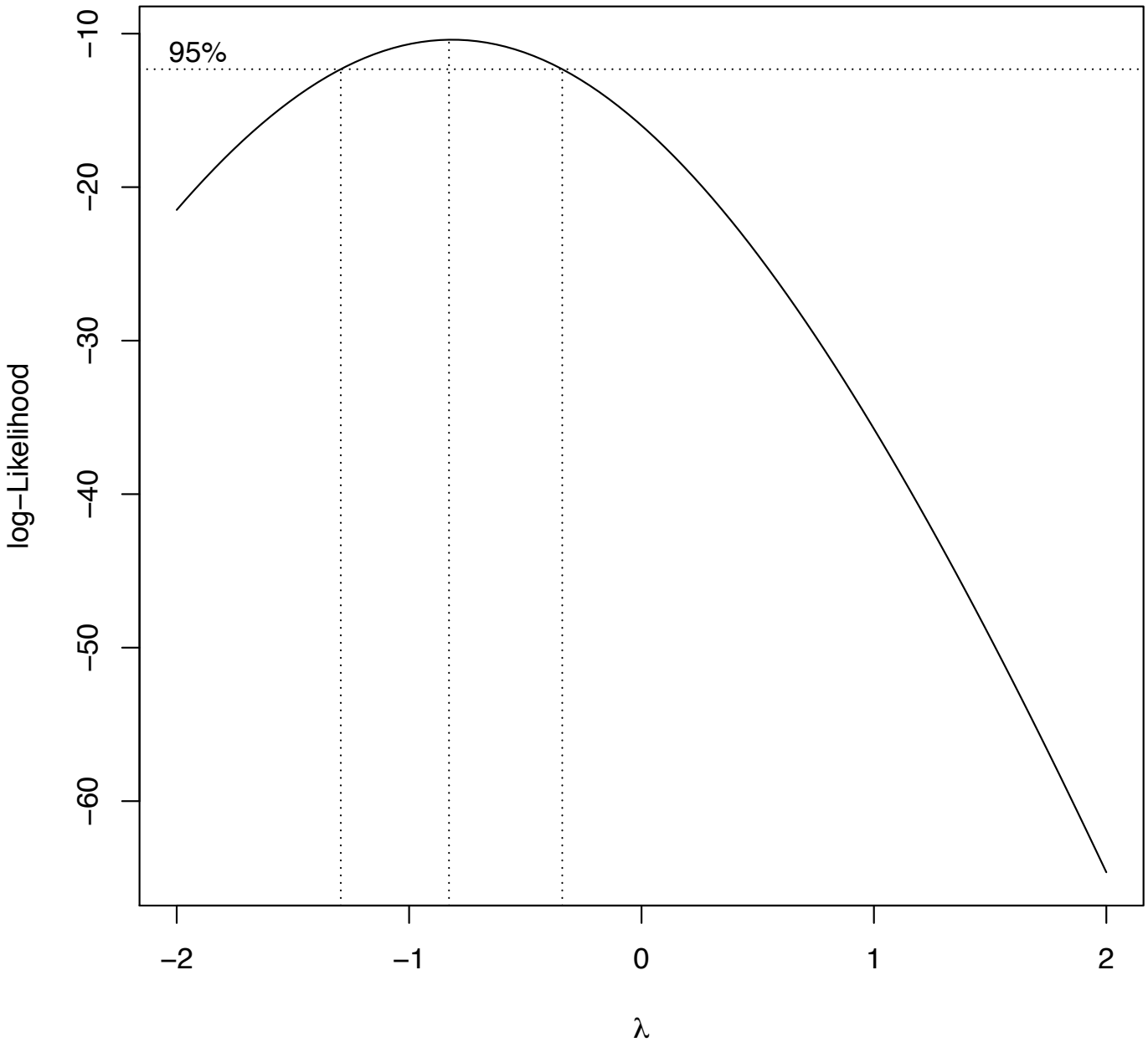

Scale−Location
√|Standardized residuals|
Fitted values
30
24
26


Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
Factor Level Combinations
poison
I
II
III
30
24
26

The plots clearly show non-constant variance — the variance increases with the fitted value.

Let's see if a Box-Cox (power) transformation on survival times can fix things ...

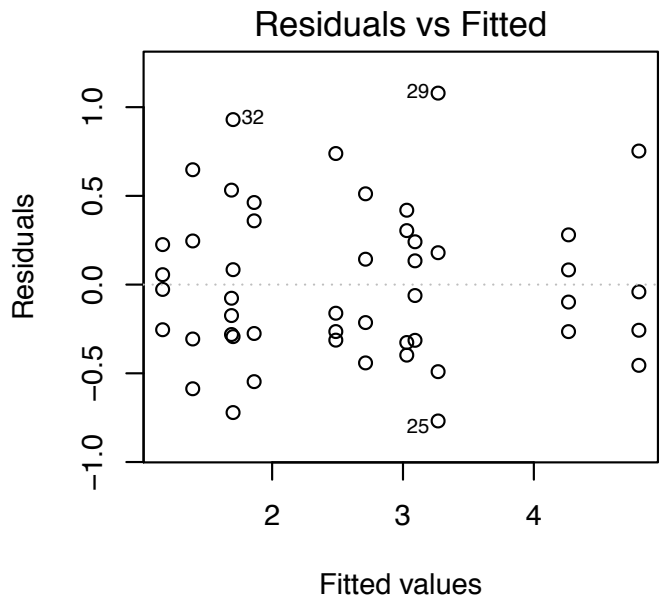
```
> library(MASS)

> par(mfrow=c(1,1))
> boxcox(fullmod)
```

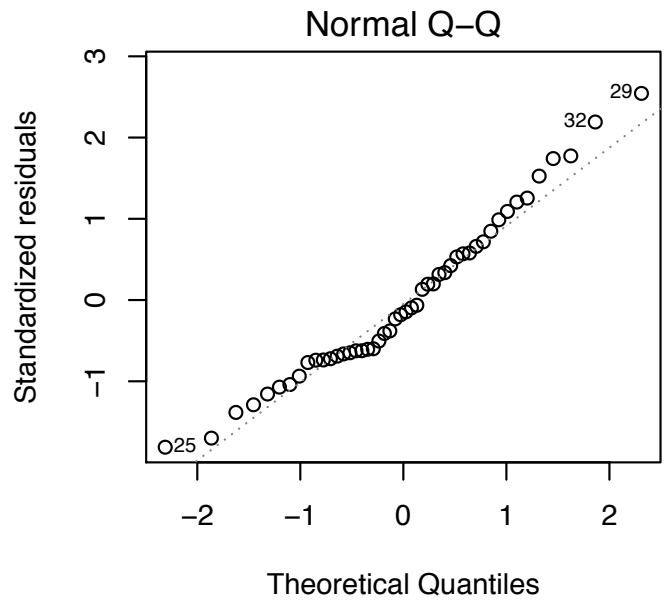

95%
log−Likelihood
−10
−20
−30
−40
−50
−60
−2
−1
0
1
2
λ

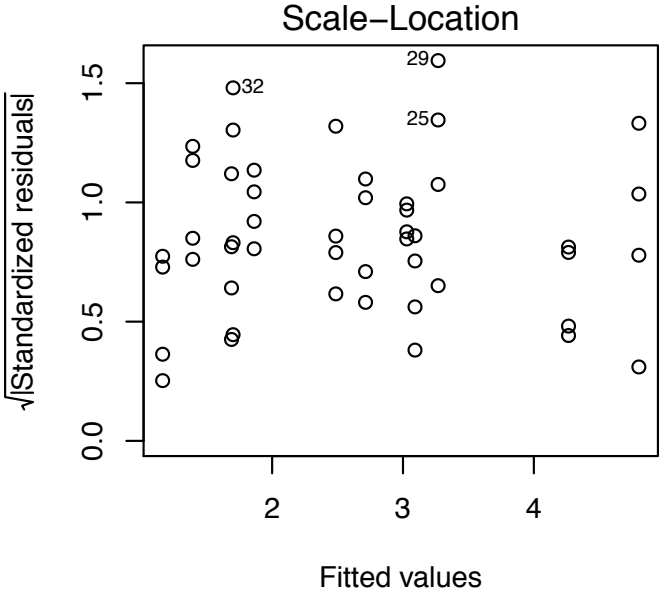
It looks like taking "inverse" (reciprocal) survival, that is, death rate, would be a good choice.

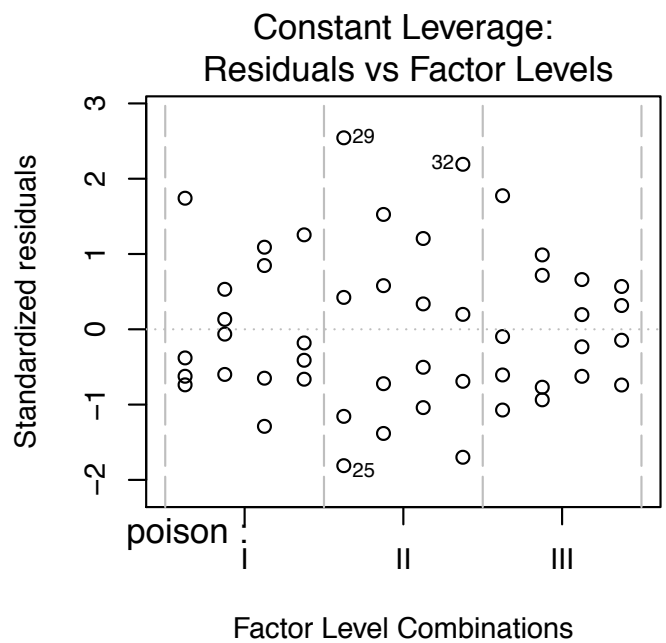
Let's fit the full model, this time with reciprocal survival times, and see if the variance issue has been corrected ...

```
> fullmodinv <- lm(time^-1 ~ poison * treat, data=rats)

> summary(fullmodinv)

...

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       2.48688    0.24499  10.151 4.16e-12 ***
poisonII          0.78159    0.34647   2.256 0.030252 *
poisonIII         2.31580    0.34647   6.684 8.56e-08 ***
treatB           -1.32342    0.34647  -3.820 0.000508 ***
treatC           -0.62416    0.34647  -1.801 0.080010 .
treatD           -0.79720    0.34647  -2.301 0.027297 *
poisonII:treatB  -0.55166    0.48999  -1.126 0.267669
poisonIII:treatB -0.45030    0.48999  -0.919 0.364213
poisonII:treatC   0.06961    0.48999   0.142 0.887826
poisonIII:treatC  0.08646    0.48999   0.176 0.860928
poisonII:treatD  -0.76974    0.48999  -1.571 0.124946
poisonIII:treatD -0.91368    0.48999  -1.865 0.070391 .

...

> par(mfrow=c(2,2))
> plot(fullmodinv, add.smooth=FALSE)
```


Residuals vs Fitted
Residuals
Fitted values
29
32
25


Normal Q−Q
Standardized residuals
Theoretical Quantiles
29
32
25


Scale−Location
√|Standardized residuals|
Fitted values
29
32
25

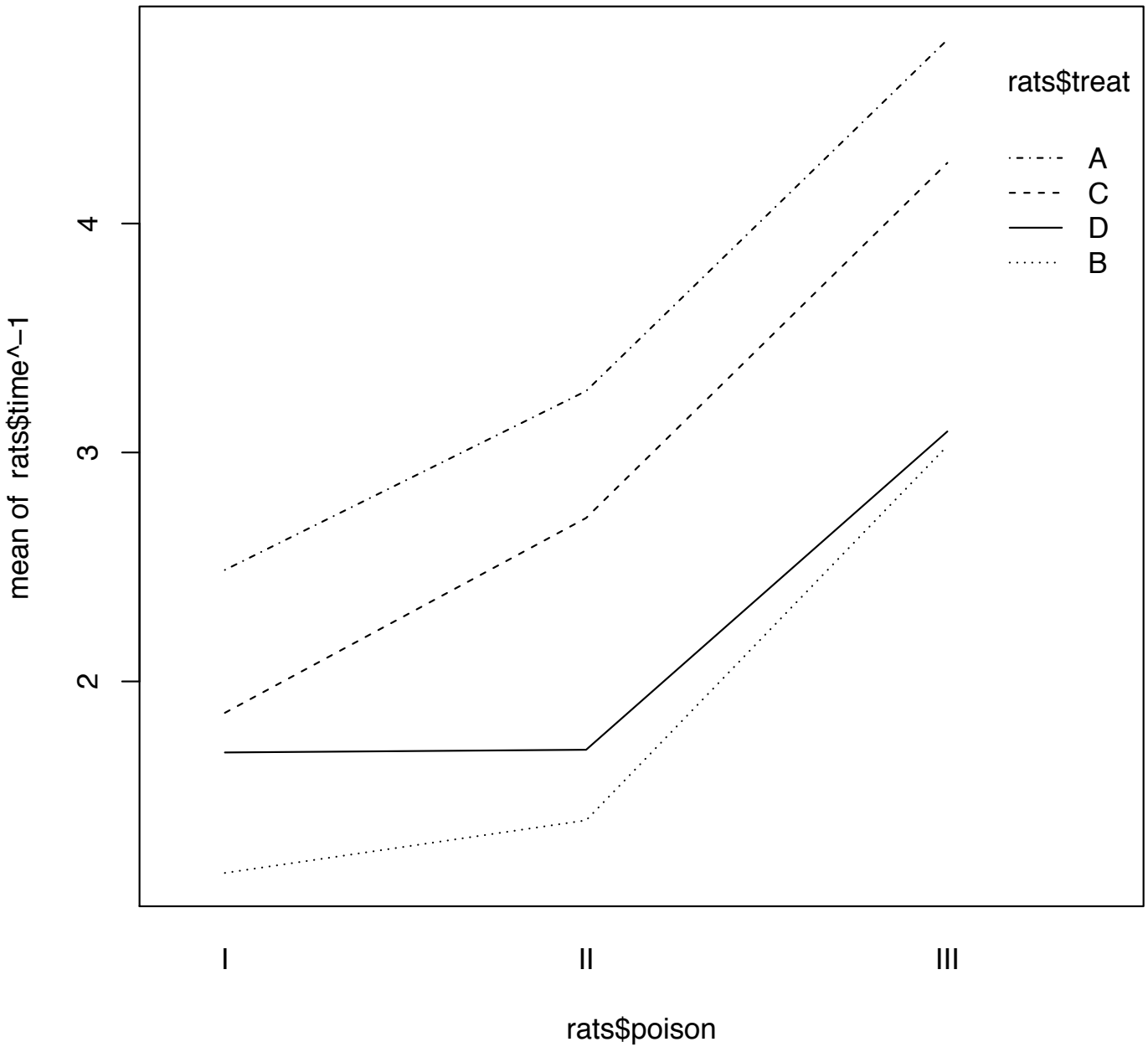

Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
Factor Level Combinations
poison
I
II
III
29
32
25

Diagnostics now look good.

But we still need something better than the raw coefficient estimates to help understand the effects ...

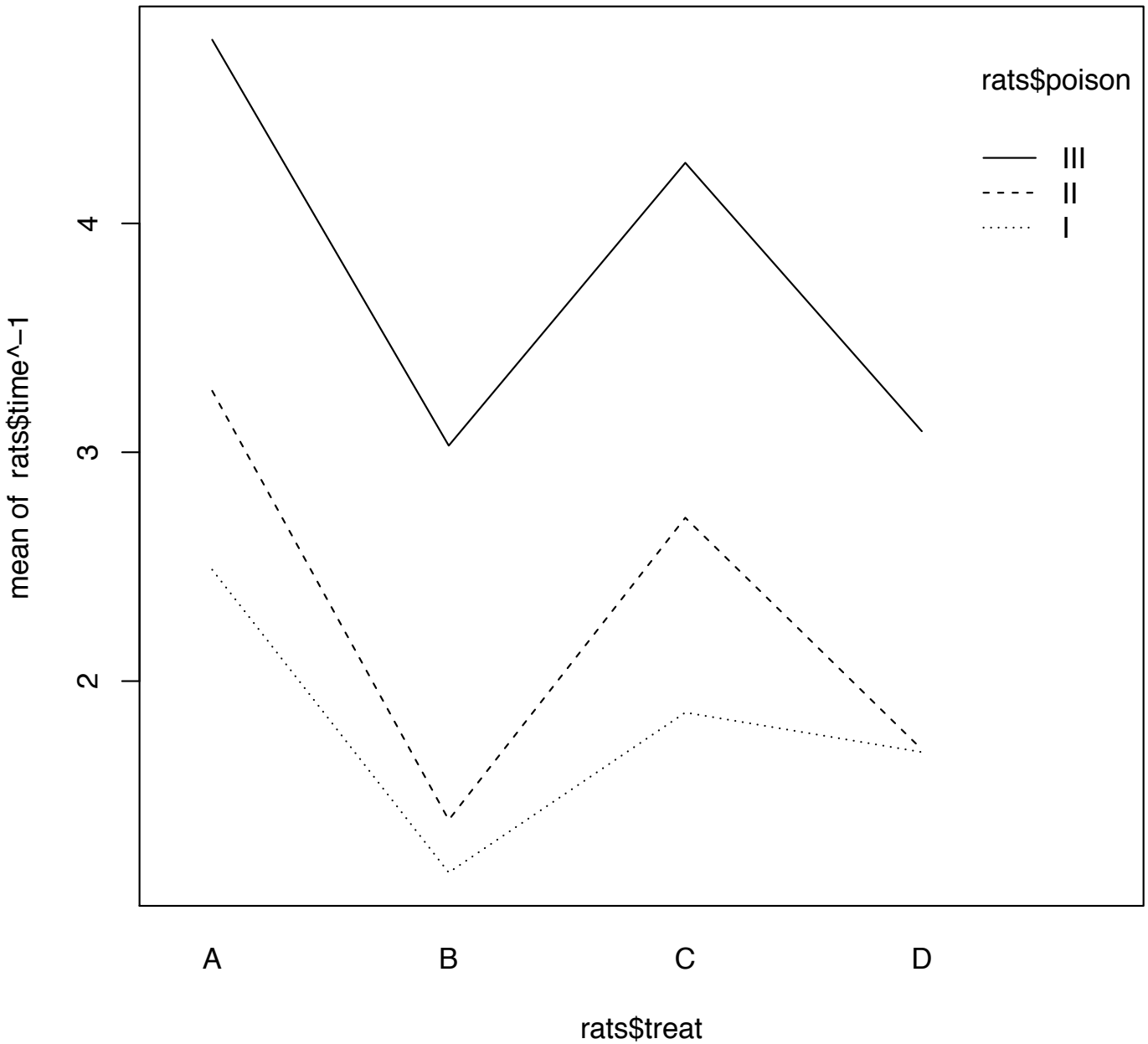
An **interaction plot** displays estimated response means versus factor level combinations. It can be used to visually assess how the factors interact (if at all), and the nature of the main effects ...

```
> par(mfrow=c(1,1))
> interaction.plot(rats$poison, rats$treat, rats$time^-1)
```


rats$treat
A
C
D
B
mean of rats$time^−1
2
3
4
I
II
III
rats$poison

Could alternatively produce an interaction plot with "treatment" on the horizontal axis:

```
> interaction.plot(rats$treat, rats$poison, rats$time^-1)
```


rats$poison
III
II
I
mean of rats$time^−1
4
3
2
A
B
C
D
rats$treat

Let's fit a *main effects model*: an effects model that does not have the interaction term:

$$Y_{ijk}^{-1} = \mu + \alpha_i + \beta_j + e_{ijk}$$

This is clearly a reduced model (relative to the full factor effects model with response Y_{ijk}^{-1}), so it will help us test whether we need interaction terms ...

```
> maineffectsinv <- lm(time^-1 ~ poison + treat, data=rats)

> summary(maineffectsinv)

...

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.6977     0.1744  15.473  < 2e-16 ***
poisonII      0.4686     0.1744   2.688  0.01026 *
poisonIII     1.9964     0.1744  11.451 1.69e-14 ***
treatB       -1.6574     0.2013  -8.233 2.66e-10 ***
treatC       -0.5721     0.2013  -2.842  0.00689 **
treatD       -1.3583     0.2013  -6.747 3.35e-08 ***

...

> anova(maineffectsinv, fullmodinv)
Analysis of Variance Table

Model 1: time^-1 ~ poison + treat
Model 2: time^-1 ~ poison * treat
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     42 10.2139
2     36  8.6431  6    1.5708 1.0904 0.3867
```

We can perform the same test using an **ANOVA table**:

```
> anova(fullmodinv)
Analysis of Variance Table

Response: time^-1
             Df Sum Sq Mean Sq F value    Pr(>F)
poison        2 34.877 17.4386 72.6347 2.310e-13 ***
treat         3 20.414  6.8048 28.3431 1.376e-09 ***
poison:treat  6  1.571  0.2618  1.0904    0.3867
Residuals    36  8.643  0.2401
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Important: Because of hierarchy, we test for the main effects *only* if the interaction is *not* significant, as in this case.

Because this data has a *balanced* design, both F-tests for main effects in the table are also valid. (Not necessarily true for unbalanced data.)

To further understand the main effects, let's use Tukey's method for looking at differences between *factor* means ...

```
> TukeyHSD(aov(time^-1 ~ poison + treat, data=rats))
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = time^-1 ~ poison + treat, data = rats)

$poison
             diff        lwr       upr     p adj
II-I   0.4686413 0.04505584 0.8922267 0.0271587
III-I  1.9964249 1.57283950 2.4200103 0.0000000
III-II 1.5277837 1.10419824 1.9513691 0.0000000

$treat
          diff        lwr         upr     p adj
B-A -1.6574024 -2.1959343 -1.11887050 0.0000000
C-A -0.5721354 -1.1106673 -0.03360355 0.0335202
D-A -1.3583383 -1.8968702 -0.81980640 0.0000002
C-B  1.0852669  0.5467351  1.62379883 0.0000172
D-B  0.2990641 -0.2394678  0.83759598 0.4550931
D-C -0.7862029 -1.3247347 -0.24767096 0.0018399
```

Remark:

This analysis approach will work only if there is more than one replication (for at least some of the treatment groups).

If the treatment groups all have only one replication (one observation per group), the full (interacting) model is *saturated* — no way to estimate error, so the usual interaction test isn't available.

(A main effects model could still be used, but only under the *assumption* of no interaction.)

In this case, there is a limited interaction test available: *Tukey's nonadditivity test*. See Weisberg Section 9.2 or LMR2 Section 16.1.

Remark:

In some cases, the number of experimental units available is less than the number of treatment groups — so not even a single full replication is possible!

In this case, it may be useful to try a *fractional factorial*.

See LMR2 Section 16.4.

Remark:

The principle of hierarchy in a factorial ANOVA model is also called the *marginality principle* (see Weisberg Section 6.2).