Chapter 16

ANOVA

Predictors are now all categorical/ qualitative. The name ANOVA stands for Analysis of Variance is used because the original thinking was to try to partition the overall variance in the response to that due to each of the factors and the error. Predictors are now typically called factors which have some number of levels. The parameters are now often called *effects*. We shall first consider only models where the parameters are considered fixed but unknown — called *fixed-effects* models but *random-effects* models are also used where parameters are taken to be random variables.

16.1 One-Way Anova

16.1.1 The model

Given a factor α occurring at $i = 1, \ldots, I$ levels, with $j = 1, \ldots, J_i$ observations per level. We use the model

$$y_{ij} = \mu + \alpha_i + \varepsilon_{ij} \quad i = 1, \ldots, I \quad j = 1, \ldots, J_i$$

As it stands not all the parameters are identifiable and some restriction is necessary:

1. Set $\mu = 0$ and use I different dummy variables.
2. Set $\alpha_1 = 0$ — this corresponds to treatment contrasts
3. Set $\sum_i J_i \alpha_i = 0$ which leads to the least squares estimates

$$\hat{\mu} = \bar{y}_{\cdot\cdot} \qquad \hat{\alpha}_i = \bar{y}_{i\cdot} - \bar{y}_{\cdot\cdot}$$

where $\cdot$ indicates which index or indices the mean is taken over.

This last method is the most commonly recommended for manual calculation in older textbooks although it is harder to represent within in the $y = X\beta + \varepsilon$ framework. The first two are easier to implement for computations. As usual, some preliminary graphical analysis is appropriate before fitting. A side-by-side boxplot is often the most useful plot. Look for equality of variance, transformations, outliers (influence is not relevant here since leverages won't differ unless the design is very unbalanced).

16.1.2 Estimation and testing

The effects can be estimated using direct formulae as above or by using the least squares approach (the outcome is the same). The first test of interest is whether there is a difference in the levels of the factor. We compare

- $H_0 : \alpha_i = 0 \qquad \forall i$
- H_a : at least one α_i is non zero.

We use the same F-test as we have used for regression. The outcome of this test will be the same no matter what coding/restriction we use. If the null is accepted then we are done (subject to an investigation of transformation and outliers). If we reject the null, we must investigate which levels differ.

16.1.3 An example

The example dataset we will use is a set of 24 blood coagulation times. 24 animals were randomly assigned to four different diets and the samples were taken in a random order. This data comes from Box, Hunter, and Hunter (1978).

```
> data(coagulation)
> coagulation
   coag diet
1    62    A
2    60    A
...etc...
23   63    D
24   59    D
```

The first step is to plot the data - boxplots are useful:

```
> plot(coag ~ diet, data=coagulation)
```

See the first panel of Figure 16.1.3. We are hoping *not* to see

1. Outliers — these will be apparent as separated points on the boxplots. The default is to extend the *whiskers* of the boxplot no more than one and half times the interquartiles range from the quartiles. Any points further away than this are plotted separately.
2. Skewness — this will be apparent from an asymmetrical form for the boxes.
3. Unequal variance — this will be apparent from clearly unequal box sizes. Some care is required because often there is very little data be used in the construction of the boxplots and so even when the variances truly are equal in the groups, we can expect a great deal of variability

In this case, there are no obvious problems. For group C, there are only 4 distinct observations and one is somewhat separated which accounts for the slightly odd looking plot.

Now let's fit the model.

```
> g <- lm(coag ~ diet, coagulation)
> summary(g)
Coefficients:
             Estimate Std. Error  t value Pr(>|t|)
(Intercept)  6.10e+01   1.18e+00    51.55  < 2e-16
dietB        5.00e+00   1.53e+00     3.27  0.00380
dietC        7.00e+00   1.53e+00     4.58  0.00018
dietD       -1.00e-14   1.45e+00 -7.4e-15  1.00000
```

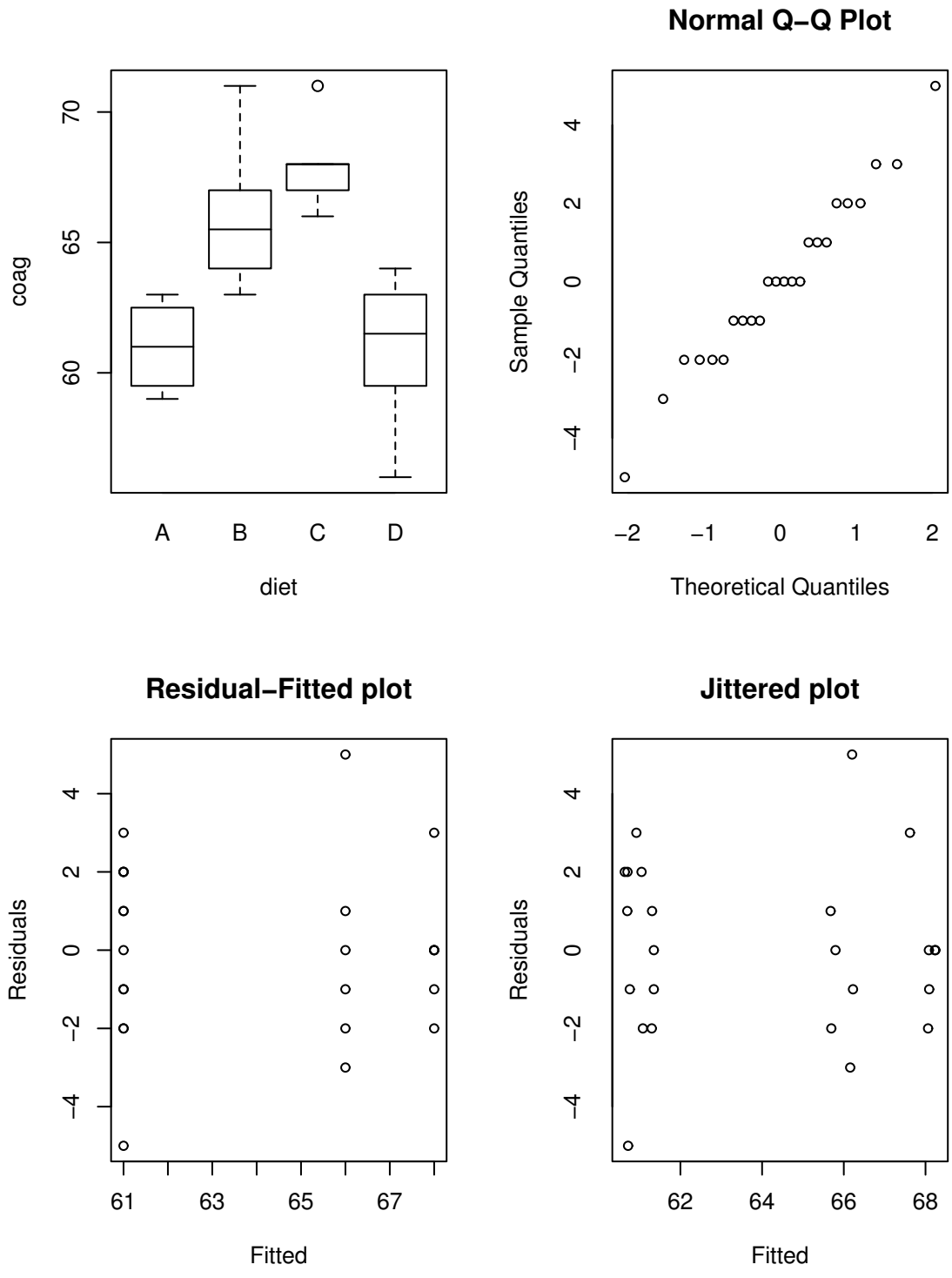

Figure 16.1: One way anova plots

```
Residual standard error: 2.37 on 20 degrees of freedom
Multiple R-Squared: 0.671,      Adjusted R-squared: 0.621
F-statistic: 13.6 on 3 and 20 degrees of freedom,       p-value: 4.66e-05
```

We conclude from the small p-value for the F-statistic that there is some difference between the groups? Group A is the reference level and has a mean of 61, groups B, C and D are 5, 7 and 0 seconds larger on average. Examine the design matrix to understand the coding:

```
> model.matrix(g)
```

We can fit the model without an intercept term as in

```
> gi <- lm(coag ~ diet -1, coagulation)
> summary(gi)
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
dietA   61.000      1.183    51.5   <2e-16
dietB   66.000      0.966    68.3   <2e-16
dietC   68.000      0.966    70.4   <2e-16
dietD   61.000      0.837    72.9   <2e-16

Residual standard error: 2.37 on 20 degrees of freedom
Multiple R-Squared: 0.999,      Adjusted R-squared: 0.999
F-statistic: 4.4e+03 on 4 and 20 degrees of freedom,    p-value:    0
```

We can directly read the level means but the tests are not useful since they involve comparisons with zero. Note the miscalculation of R^2.

16.1.4 Diagnostics

Remember to plot the residuals/fitted values and do the QQ plot. Influential points and transforming the predictors are not an issue although it is reasonable to consider transforming the response if the situation demands it.

See the last three panels of Figure 16.1.3.

```
> qqnorm(g$res)
> plot(g$fit,g$res,xlab="Fitted",ylab="Residuals",
  main="Residual-Fitted plot")
> plot(jitter(g$fit),g$res,xlab="Fitted",ylab="Residuals",
  main="Jittered plot")
```

Because the data are integers and the fitted values turn out to integers also, some discreteness is obvious in the Q-Q plot. Of course, discrete data can't be normally distributed. However, here it is approximately normal and so we can go ahead with the inference without any qualms. The discreteness in the residuals and fitted values shows up in the residual-fitted plot because we can see fewer points than the sample size. This is because of overplotting of the point symbols. There are several ways round this problem. One simple solution is to add a small amount of noise to the data. This is called *jittering*. Sometimes you have to tune the amount of noise but the default setting is adequate here.

16.1.5 Multiple Comparisons

After detecting some difference in the levels of the factor, interest centers on which levels or combinations of levels are different. Note that it does not make sense to ask whether a particular level is significant since this begs the question, "significantly different from what". Any meaningful test must involve a comparison of some kind.

It is important to ascertain whether the comparison made were decided on before or after examining the data. After fitting a model, one might decide to test only those differences that look large. To make such a decision, you also have to examine the small differences. Even if you do not actually test these small differences, it does have an effect on the inference.

If the comparisons were decided on prior to examining the data, there are three cases:

1. Just one comparison — use the standard t-based confidence intervals that we have used before.
2. Few comparisons — use the Bonferroni adjustment for the t. If there are m comparisons, use α/m for the critical value.
3. Many comparisons — Bonferroni becomes increasingly conservative as m increases. At some point it is better to use the Tukey or Scheffé or related methods described below.

It is difficult to be honest and be seen to be honest when using pre-data comparisons. Will people really believe that you only planned to make certain comparisons? Although some might make a distinction between pre and post-data comparisons, I think it is best to consider all comparisons as post-data.

If the comparisons were decided on after examining the data, you must adjust the CI to allow for the possibility of all comparisons of the type to be made.

There are two important cases:

1. Pairwise comparisons only. Use the Tukey method.
2. All contrasts i.e. linear combinations. Use the Scheffé method.

We consider pairwise comparisons first. A simple C.I. for $\alpha_i - \alpha_j$ is

$$\hat{\alpha}_i - \hat{\alpha}_j \pm t_{n-I}^{\alpha/2} \hat{\sigma} \sqrt{\frac{1}{J_i} + \frac{1}{J_j}}$$

A test for $\alpha_i = \alpha_j$ amounts to seeing whether zero lies in this interval or not. This is fine for just one test but suppose we do all possible pairwise tests when $\alpha = 5\%$ and the null hypothesis is in fact true. In Table 16.1, we see effects of multiple comparisons on the true error rates.

I	2	3	4	5	6
Nominal Type I error	5%	5%	5%	5%	5%
Actual overall Type I error	5%	12.2%	20.3%	28.6%	36.6%

Table 16.1: True error rates for multiple comparisons

We see that the true type I error can get quite high. Using the t-based CI for multiple comparisons is called least significant differences or LSD but this one is a bad trip. When comparisons are only made after the overall F-test shows a difference, it's called Fisher's LSD — this one isn't quite so bad but the type I error will still be too big.

We can use a simulation to illustrate the issues involved in multiple comparisons. Because random numbers are random, your results may differ but the message will be the same.

Suppose we have a factor of 6 levels with 4 observations per level:

```
> x <- factor(rep(LETTERS[1:6],rep(4,6)))
> x
 [1] A A A A B B B B C C C C D D D D E E E E F F F F
Levels:  A B C D E F
```

and suppose the response has no relationship to the factor (i.e. the null hypothesis holds):

```
> g <- lm(rnorm(24) ~ x)
> gs <- summary(g)
```

Here are the coefficients:

```
> g$coef
(Intercept)          xB          xC          xD          xE          xF
   0.221638    0.331200    0.058631   -0.536102    0.295339    0.067889
```

The t-statistic for testing whether level A = level B is $(\hat{\alpha}_A - \hat{\alpha}_B)/se(\hat{\alpha}_A - \hat{\alpha}_B)$ where $se((\hat{\alpha}_A - \hat{\alpha}_B)) = \hat{\sigma}\sqrt{1/4+1/4} = \hat{\sigma}/\sqrt{(2)}$

```
> g$coef[2]*sqrt(2)/gs$sig
     xB
0.41881
```

This would (in absolute value) need exceed this t-critical value for significance at the 5% level:

```
> qt(0.975,24-6)
[1] 2.1009
```

Out of all the possible pairwise comparisons, we may compute the maximum t-statistic as

```
> range(c(0,g$coef[-1]))
[1] -0.5361  0.3312
> rg <- range(c(0,g$coef[-1]))
> (rg[2]-rg[1])*sqrt(2)/gs$sig
[1] 1.0967
```

which just fails to meet significance. Now let's repeat the experiment 1000 times.

```
> res <- matrix(0,1000,2)
> for(i in 1:1000){
g <- lm(rnorm(24) ~ x)
gs <- summary(g)
res[i,1] <- abs(g$coef[2]*sqrt(2)/gs$sig)
rg <- range(c(0,g$coef[-1]))
res[i,2] <- (rg[2]-rg[1])*sqrt(2)/gs$sig
}
```

Now see how many of the test statistics for comparing level A and level B were significant at the 5% level:

```
> sum(res[,1] > 2.1)/1000
[1] 0.045
```

Just a shade under the 5% it should be. Now see how many times the maximum difference is significant at the 5% level.

```
> sum(res[,2] > 2.1)/1000
[1] 0.306
```

About 1/3 of the time. So in cases where there is no difference between levels of the factor, about 1/3 of the time, an investigator will find a statistically significant difference between some levels of the factor. Clearly there is a big danger that one might conclude there is a difference when none truly exists.

We need to make the critical value larger so that the null is rejected only 5% of the time. Using the simulation results we estimate this value to be:

```
> quantile(res[,2],0.95)
   95%
3.1627
```

It turns out that this value may be calculated using the "Studentized Range distribution":

```
> qtukey(0.95,6,18)/sqrt(2)
[1] 3.1780
```

which is close to the simulated value.

Now let's take a look at the densities of our simulated t-statistics:

```
> dmax <- density(res[,2],from=0,to=5)
> d2 <- density(res[,1],from=0,to=5)
> matplot(d2$x,cbind(dmax$y,d2$y),type="l",xlab="Test statistic",
  ylab="Density")
> abline(h=0)
> abline(v=2.1,lty=2)
> abline(v=3.178)
```

We see the result in Figure 16.2. We see that the distribution of the maximum t-statistic has a much heavier tail than the distribution for a prespecified difference. The true density for the prespecified difference is a the upper half of a t-distribution — the maximum in the estimated distribution does not occur at zero because boundary error effects in the density estimator.

Now we return to our real data. We've found that there is a significant difference among the diets but which diets can be said to be different and which diets are not distinguishable. Let's do the calculations for the difference between diet B and diet C which is 2. First we do the LSD calculation:

```
> qt(1-.05/2,20)*2.366*sqrt(1/6+1/6)
[1] 2.8494
> c(2-2.85,2+2.85)
[1] -0.85  4.85
```

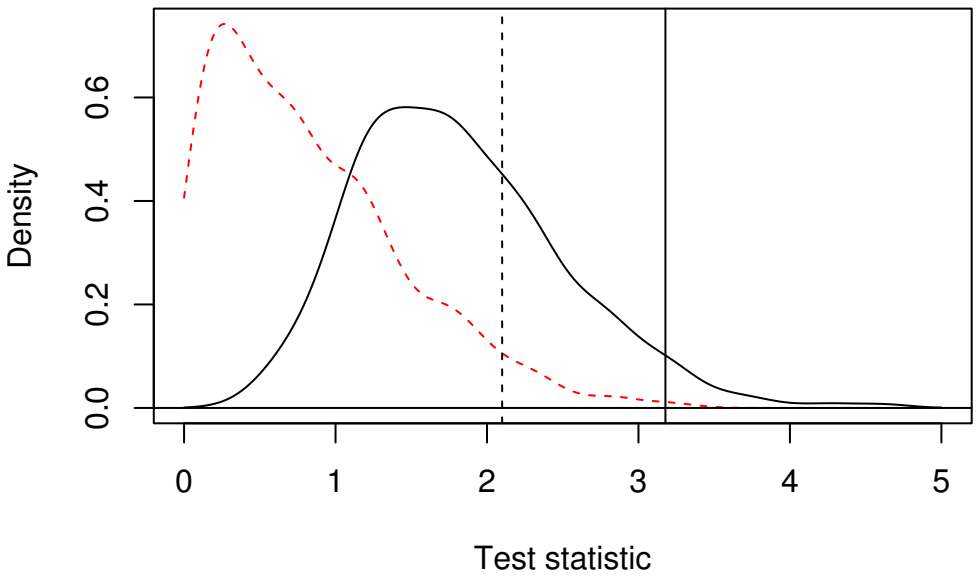

Figure 16.2: Estimated densities of the maximum t-statistic (solid line) and t-statistic for a prespecified difference (dashed line). The corresponding theoretical 95% quantiles are marked with vertical lines

An alternative way we can make the same calculation is to recode the factor with B as the reference level and refit the model:

```
> coagulation$diet <- relevel(coagulation$diet,ref="B")
> g <- lm(coag ~ diet, coagulation)
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   66.000      0.966   68.32  < 2e-16
dietA         -5.000      1.528   -3.27  0.00380
dietC          2.000      1.366    1.46  0.15878
dietD         -5.000      1.278   -3.91  0.00086

Residual standard error: 2.37 on 20 degrees of freedom
Multiple R-Squared: 0.671,      Adjusted R-squared: 0.621
F-statistic: 13.6 on 3 and 20 degrees of freedom,       p-value: 4.66e-05
```

We can read the B vs. C difference directly from the output now as 2 and compute the width of the confidence band using the corresponding standard error:

```
> qt(0.975,20)*1.366
[1] 2.8494
```

We can verify that the standard error of 1.366 can also be obtained directly as

```
> 2.366*sqrt(1/6+1/6)
[1] 1.366
```

As can be seen, the result is the same as before.

Suppose two comparisons were pre-planned, then critical value is now this, using the Bonferroni correction.

```
> qt(1-.05/4,20)*2.37*sqrt(1/6+1/6)
[1] 3.3156
> c(2-3.32,2+3.32)
[1] -1.32  5.32
```

Tukey's Honest Significant Difference (HSD) is designed for all pairwise comparisons and depends on the studentized range distribution. Let $X_1, \ldots, X_n$ be i.i.d. $N(\mu, \sigma^2)$ and let $R = \max_i X_i - \min_i X_i$ be the range. Then $R/\hat{\sigma}$ has the studentized range distribution $q_{n,\nu}$ where ν is the number of degrees of freedom used in estimating σ.

The Tukey C.I.'s are

$$\hat{\alpha}_i - \hat{\alpha}_j \pm q_{I,n-I} \frac{\hat{\sigma}}{\sqrt{2}} \sqrt{\frac{1}{J_i} + \frac{1}{J_j}}$$

When the sample sizes J_i are very unequal, Tukey's HSD may be too conservative but in general they are narrower than those produced by Scheffé's theorem. There are several other methods for multiple comparisons — the Tukey method tends to be more conservative than most because it takes the rather pessimistic approach based on the maximum difference. Not all the differences will be as large as the maximum and so some competing methods take advantage of this to get tighter intervals.

For future reference, a more general form for the Tukey intervals is

$$(\text{difference}) \pm (q_{l,df}/\sqrt{2}) \times (\text{se of difference})$$

where l is the number of levels of the factor on which we are making multiple comparisons and df is the degrees of freedom for the error.

We compute the Tukey HSD bands for the diet data. First we need the critical value from the studentized range distribution.

```
> qtukey(0.95,4,20)
[1] 3.9583
```

and the interval is:

```
> (3.96/sqrt(2))*2.37*sqrt(1/6+1/6)
[1] 3.8315
> c(2-3.83,2+3.83)
[1] -1.83  5.83
```

A convenient way to obtain all the intervals is

```
> TukeyHSD(aov(coag ~ diet, coagulation))
  Tukey multiple comparisons of means
    95% family-wise confidence level

$diet
           diff      lwr      upr
```

```
A-B -5.0000e+00  -9.2754 -0.72455
C-B  2.0000e+00  -1.8241  5.82407
D-B -5.0000e+00  -8.5771 -1.42291
C-A  7.0000e+00   2.7246 11.27545
D-A -1.4211e-14  -4.0560  4.05604
D-C -7.0000e+00 -10.5771 -3.42291
```

The Bonferroni based bands would have been just slightly wider:

```
> qt(1-.05/12,20)*2.37*sqrt(1/6+1/6)
[1] 4.0052
```

We divide by 12 here because there are 6 possible pairwise differences and we want a two-sided confidence interval: $6 \times 2 = 12$. With a bit of work we find that only the A-D and B-C differences are not significant.

The Tukey method assumes the worst by focusing on the largest difference. There are other competitors like the Newman-Keuls, Duncan's multiple range and the Waller-Duncan procedure. For a detailed description of the many availabe alternatives see Hsu (1996). Some other pairwise comparison tests may be found in the R package `ctest`.

16.1.6 Contrasts

A contrast among the effects $\alpha_1, \ldots, \alpha_I$ is a linear combination $\sum_i c_i \alpha_i$ where the c_i are known and $\sum_i c_i = 0$. For example

1. $\alpha_1 - \alpha_2$ is a contrast with $c_1 = 1, c_2 = -1$ and the other $c_i = 0$. All pairwise differences are contrasts.
2. $(\alpha_1 + \alpha_2)/2 - (\alpha_3 + \alpha_4)/2$ with $c_1 = c_2 = 1/2$ and $c_3 = c_4 = -1/2$ and the other $c_i = 0$. This contrast is directly interpretable.

16.1.7 Scheffé's theorem for multiple comparisons

An estimable function of the parameters is one that can be estimated given the data we have. More precisely, a linear combination $\psi = c^T\beta$ is estimable if there exists an $a^T y$ such that $Ea^T y = c^T\beta$. Contrasts are estimable but something like α_i is not because it will depend on the coding used. Now $\hat{\psi} = a^T y$ and var $\hat{\psi} = \sigma^2 a^T a$ which can be estimated by $\hat{\sigma}^2 a^T a$. Suppose we let the dimension of the space of possible c be q and the rank$(X) = r$. ($r = p$ if we have complete identifiability.)

Scheffé's theorem

A $100(1-\alpha)\%$ simultaneous confidence interval for all estimable ψ is

$$\hat{\psi} \pm \sqrt{qF^{\alpha}_{q,n-r}}\sqrt{\hat{\text{var}}\,\hat{\psi}}$$

Example: Simultaneous confidence interval for the regression surface:

$$x^T\hat{\beta} \pm \sqrt{pF^{\alpha}_{p,n-p}}\hat{\sigma}\sqrt{x^T(X^TX)^{-1}x}$$

We can illustrate this with corrosion data used in the lack of fit chapter. We compute the usual t-based pointwise bands along with the simultaneous Scheffé bands:

```
> data(corrosion)
> gf <- lm(loss ~ Fe, corrosion)
> grid <- seq(0,3,by=0.1)
> p <- predict(gf,data.frame(Fe=grid),se=T)
> fmult <- sqrt(2*qf(0.95,2,11))
> tmult <- qt(0.975,11)
> matplot(grid,cbind(p$fit,p$fit-fmult*p$se,p$fit+fmult*p$se,
  p$fit-tmult*p$se,p$fit+tmult*p$se),type="l",lty=c(1,2,2,5,5),
  ylab="loss",xlab="Iron Content")
> points(corrosion$Fe,corrosion$loss)
```

The plot is shown in Figure 16.3. The bands form a 95% simultaneous confidence region for the true regression line. These bands are slightly wider than the t-based pointwise confidence bands described in Chapter 3. This is because they hold over the whole real line and not just a single point.

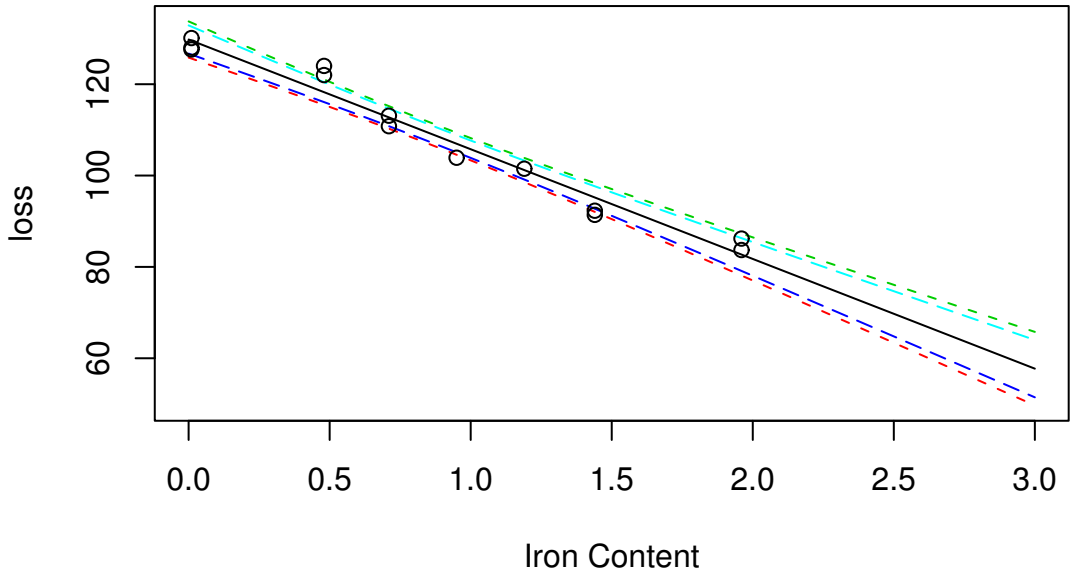

Figure 16.3: Scheffé 95% simultaneous confidence bands are shown as dotted lines surrounding the least squares fit. The interior (dashed) lines represent the pointwise confidence intervals.

Example: One-way anova: Consider $\psi = \sum_i c_i \alpha_i$, a contrast which is therefore estimable. We compute

$$\hat{\text{var}}\ \hat{\psi} = \hat{\sigma}^2 \sum_{i=1}^{I} \frac{c_i^2}{J_i}$$

and the SCI for ψ is then

$$\sum_i c_i \hat{\alpha}_i \pm \sqrt{(I-1)F^{\alpha}_{I-1,n-I}}\hat{\sigma}\sqrt{\sum_{i=1}^{I}\frac{c_i^2}{J_i}}$$

Here we apply the Scheffé method for (B+C)/2 -(A+D)/2 so that $c_2 = c_3 = 1/2$ and $c_1 = c_4 = -1/2$

```
> sqrt(3*qf(0.95,3,20))*2.37*sqrt(1/4+1/6+1/6+1/8)/2
[1] 3.0406
```

```
> (5+7)/2-(0+0)/2
[1] 6
> c(6-3.04,6+3.04)
[1] 2.96 9.04
```

We see that this difference is significantly different from 0 and so we may conclude that there is significant difference between the average of B and C and the average of A and D despite the fact that we may have chosen to test this difference after seeing the data.

16.1.8 Testing for homogeneity of variance

This can be done using Levene's test. Simply compute the absolute values of the residuals and use these as the response in a new one-way anova. A significant difference would indicate non constant variance.

There are other tests but this one is quite insensitive to non-normality and is simple to execute. Most tests and CI's are relatively insensitive to non-constant variance so there is no need to take action unless the Levene test is significant at the 1% level.

Applying this to the diet data, we find:

```
> summary(lm( abs(g$res) ~ coagulation$diet))
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)           1.500      0.716    2.10    0.049
coagulation$dietB     0.500      0.924    0.54    0.594
coagulation$dietC    -0.500      0.924   -0.54    0.594
coagulation$dietD     0.500      0.877    0.57    0.575

Residual standard error: 1.43 on 20 degrees of freedom
Multiple R-Squared: 0.0956,     Adjusted R-squared: -0.0401
F-statistic: 0.705 on 3 and 20 degrees of freedom,      p-value: 0.56
```

Since the p-value is large, we conclude that there is no evidence of a non-constant variance.

16.2 Two-Way Anova

Suppose we have two factors, α at I levels and β at J levels. Let n_{ij} be the number of observations at level i of α and level j of β and let those observations be $y_{ij1}, y_{ij2}, \ldots$. A complete layout has $n_{ij} \geq 1$ for all i, j. The most general model that may be considered is

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}$$

The interaction effect $(\alpha\beta)_{ij}$ is interpreted as that part of the mean response not attributable to the additive effect of α_i and β_j. For example, you may enjoy strawberries and cream individually, but the combination is superior. In contrast, you may like fish and ice cream but not together.

A balanced layout requires that $n_{ij} = n$. Not all the parameters are identifiable but if the main effects α and β are coded appropriately and the interaction effects coding is then derived from the product of these codings, then every contrast of parameters can be estimated.

16.2.1 One observation per cell

When $n_{ij} = 1$ we would have as many observation as parameters if we tried to fit the full model as above. The parameters could be estimated but no further inference would be possible.

We can assume $(\alpha\beta)_{ij} = 0$ to free up degrees of freedom to make some tests and CI's. This assumption can be checked graphically using an interaction plot - plot the cell means on the vertical axis and the factor α on the horizontal. Join points with same level of β. The role of α and β can be reversed. Parallel lines on the plot are a sign of a lack of interaction. Tukey's non-additivity test provides another way of investigating an interaction - the model

$$y_{ij} = \mu + \alpha_i + \beta_j + \phi\alpha_i\beta_j + \varepsilon_{ijk}$$

is fit to the data and then we test if $\phi = 0$. This is a nonlinear model and that it makes the assumption that the interaction effect is multiplicative in a form which seems somewhat tenuous.

Barring any trouble with interaction, because of the balanced design, the factors are orthogonal and their significance can be tested in the usual way.

16.2.2 More than one observation per cell

When $n_{ij} = n$ i.e. the same number of observations per cell, we have orthogonality. Orthogonality can also occur if the row/column cell numbers are proportional. Orthogonality is desirable and experiments are usually designed to ensure it.

With more than one observation per cell we are now free to fit and test the model:

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}$$

The interaction effect may be tested by comparison to the model

$$y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}$$

and computing the usual F-test. If the interaction effect is found to be significant, do not test the main effects even if they appear not to be significant. The estimation of the main effects and their significance is coding dependent when interactions are included in the model.

If the interaction effect is found to be insignificant, then test the main effects but use RSS/df from the full model in the denominator of the F-tests — this has been shown to maintain the type I error better. So the F-statistic used is

$$\frac{F = RSS_{small} - RSS_{large}/(df_{small} - df_{large})}{\hat{\sigma}^2_{full}}$$

16.2.3 Interpreting the interaction effect

No interactions You can do pairwise comparisons on α without regard to β and vice versa.

Interaction present A comparison of the levels of α will depend on the level of β. Interpretation is not simple. Consider the following two layouts of $\hat{\mu}_{ij}$ in a 2x2 case:

	Male	Female	Male	Female
drug 1	3	5	2	1
drug 2	1	2	1	2

The response is a measure of performance. In the case on the left, we can say that drug 1 is better than drug 2 although the interaction means that its superiority over drug 2 depends on the gender. In the case on the right, which drug is best depends on the gender. We can also plot this as in Figure 16.4. We see that neither case are the lines parallel indicating interaction but the superiority of drug 1 is clear in the first plot and the ambiguous conclusion is clear in the second. I recommend making plots like this when you want to understand an interaction effect.

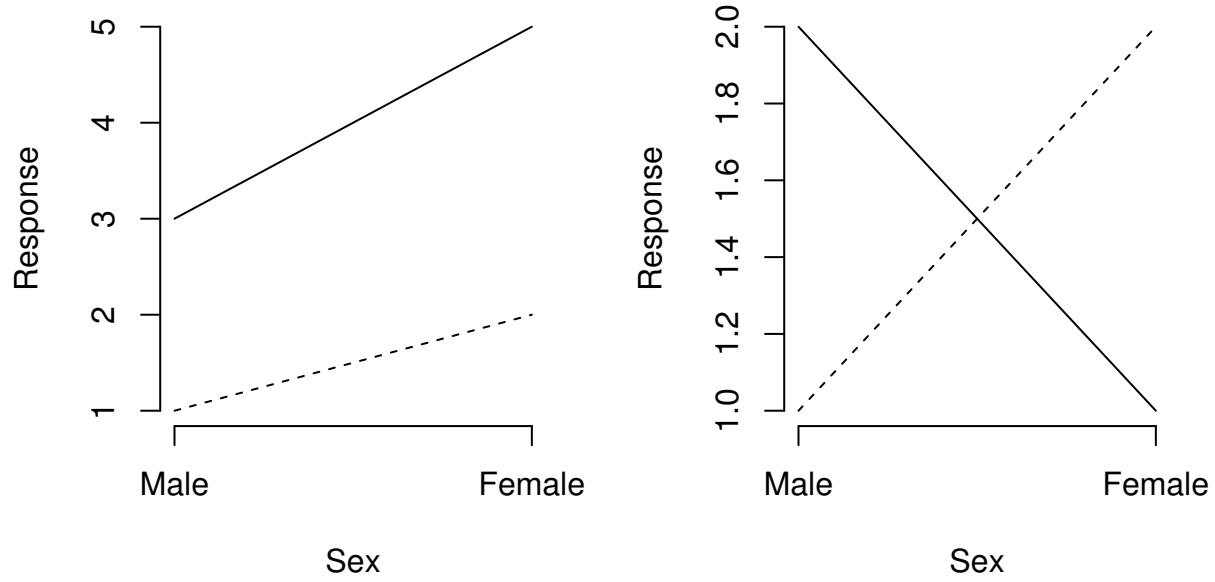

Figure 16.4: Two 2x2 tables with the response plotted by the factors, sex on the horizontal axis and drug 1 as the solid line and drug 2 as the dotted line.

When the interaction is significant, the main effects cannot be defined in an obvious and universal way. For example, we could the gender effect as the effect for females, the effect for males, the effect for the average males and females or something else. If there was no interaction effect, the gender effect could be defined unambiguously.

When you have a significant inteaction, you can fit a model

$$y_{ijk} = \mu_{ijk} + \varepsilon_{ijk}$$

and then treat the data as a one-way anova with IJ levels. Obviously this makes for more complex comparisons but this is unavoidable when interactions exist.

Here is a two-way anova design where there are 4 replicates. As part of an investigation of toxic agents, 48 rats were allocated to 3 poisons (I,II,III) and 4 treatments (A,B,C,D). The response was survival time in tens of hours. The Data:

```
          A       B       C       D
I         0.31    0.82    0.43    0.45
          0.45    1.10    0.45    0.71
          0.46    0.88    0.63    0.66
          0.43    0.72    0.76    0.62

II        0.36    0.92    0.44    0.56
```

```
          0.29    0.61    0.35    1.02
          0.40    0.49    0.31    0.71
          0.23    1.24    0.40    0.38

III       0.22    0.30    0.23    0.30
          0.21    0.37    0.25    0.36
          0.18    0.38    0.24    0.31
          0.23    0.29    0.22    0.33
```

We make some plots:

```
> data(rats)
> plot(time ~ treat + poison, data=rats)
```

Some evidence of skewness can be seen, especially since it appears that variance is in some way related to the mean response. We now check for an interaction using graphical methods:

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

Do these look parallel? The trouble with interaction plots is that we expect there to be some random variation regardless so it is difficult to distinguish true interaction from just noise. Fortunately, in this case, we have replication so we can directly test for an interaction effect.

Now fit the full model and see the significance of the factors:

```
> g <- lm(time ~ poison*treat, rats)
> anova(g)
Analysis of Variance Table

Response: time
             Df Sum Sq Mean Sq F value  Pr(>F)
poison        2  1.033   0.517   23.22 3.3e-07
treat         3  0.921   0.307   13.81 3.8e-06
poison:treat  6  0.250   0.042    1.87    0.11
Residuals    36  0.801   0.022
```

We see that the interaction effect is not significant but the main effects are. We check the diagnostics:

```
> qqnorm(g$res)
> plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals")
```

Clearly there's a problem - perhaps transforming the data will help. Try logs first:

```
> g <- lm(log(time) ~ poison*treat,rats)
> plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals",main="Log response")
```

Not enough so try the reciprocal:

```
> g <- lm(1/time ~ poison*treat,rats)
> plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals",
  main="Reciprocal response")
```

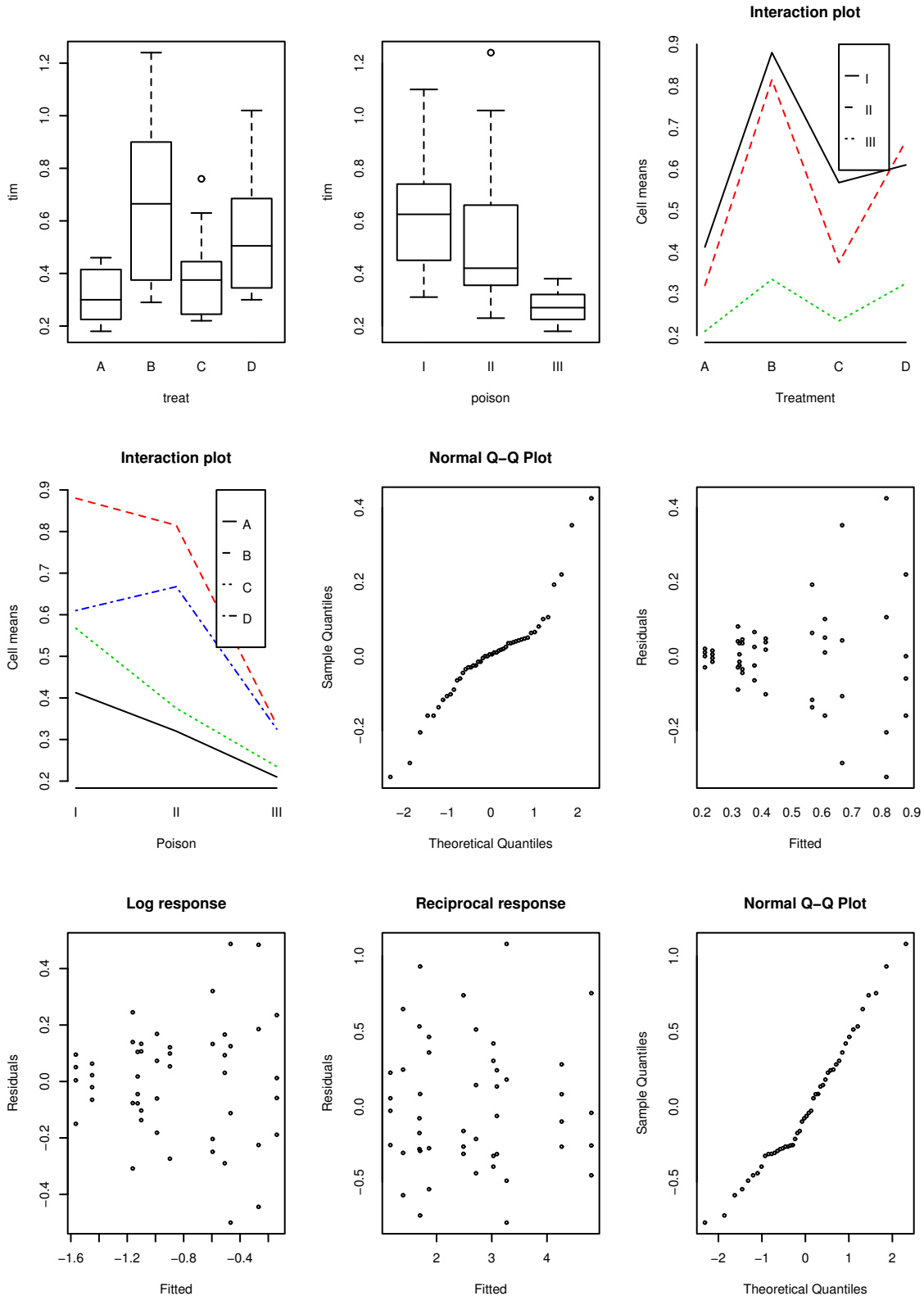

Figure 16.5: Two way anova plots

Looks good - the reciprocal can be interpreted as the rate of dying. Better check the Q-Q plot again:

```
> qqnorm(g$res)
```

This looks better than the first Q-Q plot. We now check the ANOVA table again, find the interaction is not significant, simplify the model and examine the fit:

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

Response: 1/time
             Df Sum Sq Mean Sq F value  Pr(>F)
poison        2   34.9    17.4   72.63 2.3e-13
treat         3   20.4     6.8   28.34 1.4e-09
poison:treat  6    1.6     0.3    1.09    0.39
Residuals    36    8.6     0.2

> g <- lm(1/time ~ poison+treat, rats)
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.698      0.174   15.47  < 2e-16
poisonII       0.469      0.174    2.69   0.0103
poisonIII      1.996      0.174   11.45  1.7e-14
treatB        -1.657      0.201   -8.23  2.7e-10
treatC        -0.572      0.201   -2.84   0.0069
treatD        -1.358      0.201   -6.75  3.3e-08

Residual standard error: 0.493 on 42 degrees of freedom
Multiple R-Squared: 0.844,      Adjusted R-squared: 0.826
F-statistic: 45.5 on 5 and 42 degrees of freedom,       p-value: 6.66e-16
```

Let's construct pairwise confidence intervals for the treatment factor using the Tukey method. Because of the balance, the CI's will all be the same width. First the standard error for a pairwise difference may be obtained from the output as 0.201. We then compute the width of the interval as

```
> qtukey(0.95,4,42)*0.201/sqrt(2)
[1] 0.53767
```

So the bands will be difference plus or minus 0.54. All the bands except the B-D do not include 0 so we can conclude that all these other pairs of treatments are significantly different. The treatment reduces the rats survival time the most is A since it is the reference level and all other treatments reduce the response (rate of dying). Can you distinguish between the poisons?

16.2.4 Replication

It's important that the observations observed in each cell are genuine replications. If this is not true, then the observations will be correlated and the analysis will need to be adjusted. It is a common scientific practice

to repeat measurements and take the average to reduce measurement errors. These repeat measurements are not independent observations. Data where the replicates are correlated can be handled with repeated measures models.

For example, imagine that the experiment above involved the reaction times of human subjects under two factors. We need to distinguish between an experiment that uses 48 subjects and one that uses 12 subjects where each subject repeats their assigned factor combination 4 times. In the latter case, the responses will not be independent and a repeated measures style of analysis will be necessary.

16.3 Blocking designs

In completely randomized designs (CRD) like the one and two-way anova, the treatments are assigned to the experimental units at random. This is appropriate when the units are homogenous. Sometimes, we may suspect that the units are heterogenous, but we can't describe the form it takes - for example, we may know a group of patients are not identical but we may have no further information about them. In this case, it is still appropriate to use a CRD. Of course, the randomization will tend to spread the heterogeneity around to reduce bias, but the real justification lies in the randomization test discussed earlier for regression. The usual testing argument may be applied. Under the null hypothesis, there is no link between a factor and the response. In other words, the responses have been assigned to the units in a way that is unlinked to the factor. This corresponds to the randomization used in assigning the levels of the factor to the units. This is why the randomization is crucial because it allows us to make this argument. Now if the difference in the response between levels of the factor seems too unlikely to have occurred by chance, we can reject the null hypothesis. The normal-based inference is approximately equivalent to the permutation-based test. The normal-based inference is much quicker so we might prefer to use that.

When the experimental units are heterogenous in a known way and can be arranged into *blocks* where the intrablock variation is ideally small but the interblock variation is large, a *block design* can be more efficient than a CRD.

The contrast between the two designs is shown in Figure 16.6.

Examples:

Suppose we want to compare 4 treatments and have 20 patients available. We might be able divide the patients in 5 blocks of 4 patients each where the patients in each block have some relevant similarity. We would then randomly assign the treatments within each block.

Suppose we want to test 3 crop varieties on 5 fields. Divide each field into 3 strips and randomly assign the crop variety.

Note: We prefer to have block size equal to the number of treatments. If this is not done or possible, an *incomplete* block design must be used.

Notice that under the randomized block design the randomization used in assigning the treatments to the units is restricted relative to the full randomization used in the CRD. This has consequences for the inference.

16.3.1 Randomized Block design

We have one factor (or treatment) at t levels and one blocking variable at r levels. The model is

$$y_{ij} = \mu + \tau_i + \rho_j + \varepsilon_{ij}$$

The analysis is then very similar to the two-way anova with one observation per cell. We can check for interaction and check for a treatment effect. We can also check the block effect but this is only useful for future reference. Blocking is a feature of the experimental units and restricts the randomized assignment of

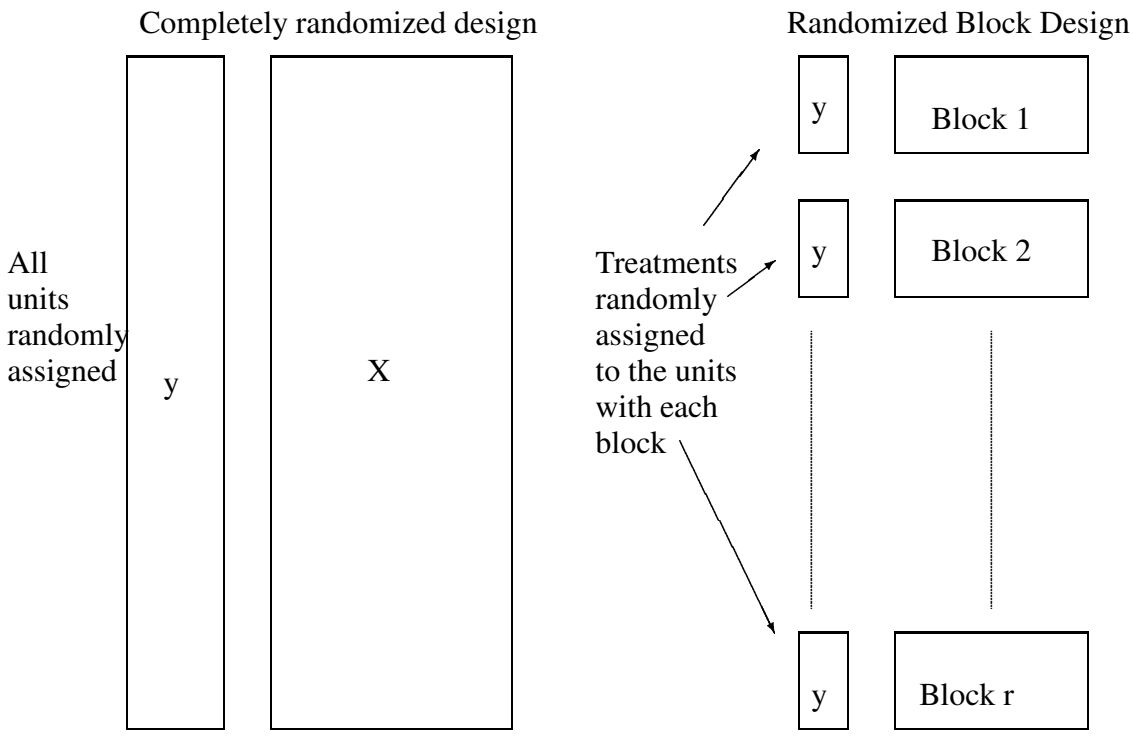

Figure 16.6: Completely randomized design vs. Randomized Block design

the treatments. This means that we cannot regain the degrees of freedom devoted to blocking even if the blocking effect turns out not to be significant. The randomization test-based argument means that we must judge the magnitude of the treatment effect within the context of the restricted randomization that has been used.

We illustrate with an experiment to compare 4 processes, A,B,C,D for the production of penicillin. These are the treatments. The raw material, corn steep liquor, is quite variable and can only be made in blends sufficient for 4 runs. Thus a randomized complete block design is definitely suggested by the nature of the experimental units. The data is:

	A	B	C	D
Blend 1	89	88	97	94
Blend 2	84	77	92	79
Blend 3	81	87	87	85
Blend 4	87	92	89	84
Blend 5	79	81	80	88

We start with some graphical checks:

```
> data(penicillin)
> plot(yield ~ blend+treat,data=penicillin)
```

See the first two panels of Figure 16.3.1

Did you see any problems? Now check for interactions:

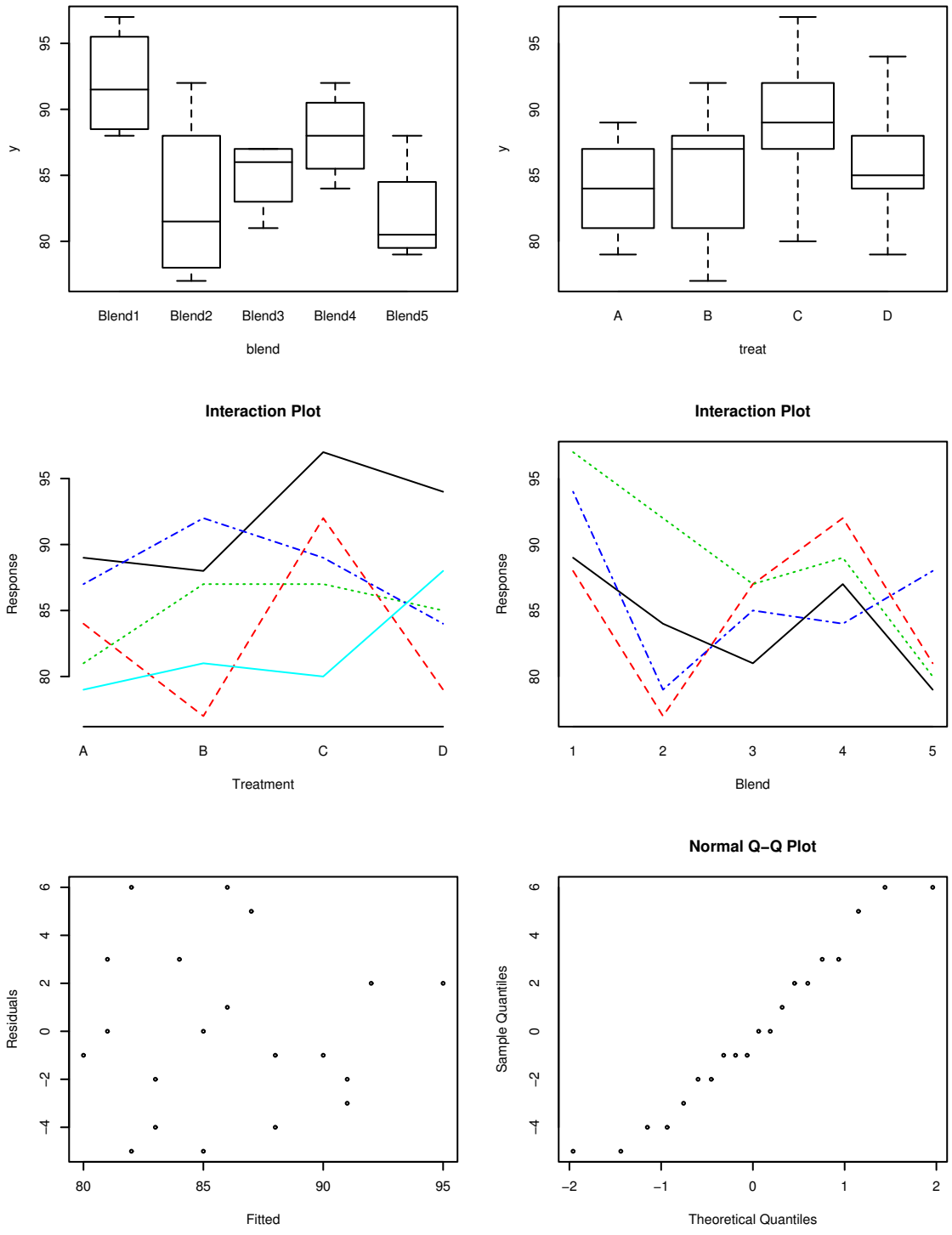

Figure 16.7: RCBD plots for the penicillin data

```
> interaction.plot(penicillin$treat,penicillin$blend,penicillin$yield)
> interaction.plot(penicillin$blend,penicillin$treat,penicillin$yield)
```

What do you think? It is hard to tell — interaction plots are only suggestive, not definitive.

Regardless, we now fit the model:

```
> g <- lm(yield ~ treat+blend,penicillin)
> anova(g)
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq F value Pr(>F)
treat      3   70.0    23.3    1.24  0.339
blend      4  264.0    66.0    3.50  0.041
Residuals 12  226.0    18.8
```

We see no significant treatment effect but the block effect is, as suspected, significant. The analysis of variance table corresponds to a sequential testing of models, here corresponding to the sequence

```
y ~ 1
y ~ treat
y ~ treat + blend
```

So here the p-value 0.339 corresponds to a comparison of the first two models in this list, while the p-value of 0.041 corresponds to the test comparing the second two. One small point to note is that the denominator in both F-test is the mean square from the full model, here 18.8

Notice that if we change the order of the terms in the ANOVA, it makes no difference because of the orthogonal design:

```
> anova(lm(yield ~ blend+treat,penicillin))
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq F value Pr(>F)
blend      4  264.0    66.0    3.50  0.041
treat      3   70.0    23.3    1.24  0.339
Residuals 12  226.0    18.8
```

By way of comparison, see what happens if we omit the first observation in the dataset — this might happen in practice if this run was lost:

```
> anova(lm(yield ~ blend+treat,penicillin[-1,]))
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq F value Pr(>F)
blend      4  266.5    66.6    3.27  0.054
treat      3   59.7    19.9    0.98  0.439
Residuals 11  224.3    20.4
```

```
> anova(lm(yield ~ treat+blend,penicillin[-1,]))
Analysis of Variance Table

Response: yield
          Df Sum Sq Mean Sq F value Pr(>F)
treat      3   91.8    30.6    1.50  0.269
blend      4  234.4    58.6    2.87  0.075
Residuals 11  224.3    20.4
```

Notice that now the order does matter. If we want to test for a treatment effect, we would prefer the first table since in that version the blocking factor `blend` is already included when we test the treatment factor. Since the blocking factor is an unalterable feature of the chosen design, this is as it should be.

Check the diagnostics:

```
> plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals")
> qqnorm(g$res)
```

And that might be the end of the story except for that worrying interaction effect possibility. We execute the Tukey non-additivity test:

```
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    90.00       2.74   32.79  4.1e-13
treatB          1.00       2.74    0.36   0.7219
treatC          5.00       2.74    1.82   0.0935
treatD          2.00       2.74    0.73   0.4802
blendBlend2    -9.00       3.07   -2.93   0.0125
blendBlend3    -7.00       3.07   -2.28   0.0416
blendBlend4    -4.00       3.07   -1.30   0.2169
blendBlend5   -10.00       3.07   -3.26   0.0068

Residual standard error: 4.34 on 12 degrees of freedom
Multiple R-Squared: 0.596,      Adjusted R-squared: 0.361
F-statistic: 2.53 on 7 and 12 degrees of freedom,       p-value: 0.0754

> alpha <- c(0,g$coef[2:4])
> alpha
       treatB treatC treatD
     0      1      5      2
> beta <- c(0,g$coef[5:8])
> beta
           blendBlend2 blendBlend3 blendBlend4 blendBlend5
          0          -9          -7          -4         -10
> ab <- rep(alpha,5)*rep(beta,rep(4,5))
> h <- update(g,.~.+ab)
> anova(h)
Analysis of Variance Table
```

```
Response: yield
          Df Sum Sq Mean Sq F value Pr(>F)
treat      3   70.0    23.3    1.15  0.374
blend      4  264.0    66.0    3.24  0.055
ab         1    2.0     2.0    0.10  0.760
Residuals 11  224.0    20.4
```

Because the p-value of the treat times block effect is .76 we accept the null hypothesis of no interaction. Of course, the interaction may be of a non-multiplicative form, but there is little we can do about that.

We can do multiple comparisons for the treatment effects, using the Tukey or the Scheffé method as appropriate:

$$\hat{\tau}_i - \hat{\tau}_j \pm q_{t,(t-1)(r-1)}\hat{\sigma}/\sqrt{r}$$

or

$$\sum_i c_i\hat{\tau}_i \pm \sqrt{(t-1)F_{t-1,(t-1)(r-1)}}\hat{\sigma}\sqrt{\sum_i c_i^2/r}$$

Now, just for the sake of the example, we compute the Tukey pairwise confidence intervals for the treatment effects:

```
> qtukey(0.95,4,12)
[1] 4.1987
```

The standard errors for the differences are

```
> 4.34*sqrt(1/5+1/5)
[1] 2.7449
```

Can you find this in the output above? The bands are difference plus or minus this:

```
> 4.2*2.745/sqrt(2)
[1] 8.1522
```

How does this compare with the observed difference in the treatment effects?

16.3.2 Relative advantage of RCBD over CRD

We can measure precision by considering var $\hat{\tau}$ (or equivalently $\hat{\sigma}^2$). Compare the $\hat{\sigma}^2$ for designs with the same sample size. We define *relative efficiency* as

$$\frac{\hat{\sigma}^2_{CRD}}{\hat{\sigma}^2_{RCBD}}$$

where the quantities can be computed by fitting models with and without the blocking effect. For example, suppose $\hat{\sigma}^2_{CRD} = (226+264)/(12+4) = 30.6$ and $\hat{\sigma}^2_{RCBD} = 18.8$, as it is in the example above, then the relative efficiency is 1.62. The $\hat{\sigma}^2_{CRD}$ numbers come from combining the sums of squares and degrees of freedom for the residuals and the blend in the first anova table we made for this data. An alternative method would be to simply fit the model `yield ~ treat` and read off $\hat{\sigma}_{CRD}$ from the output.

The interpretation is that a CRD would require 62% more observations to obtain the same level of precision as a RCBD.

The efficiency is not guaranteed to be greater than one. Only use blocking where there is some heterogeneity in the experimental units. The decision to block is a matter of judgment prior to the experiment. There is no guarantee that it will increase precision.

16.4 Latin Squares

These are useful when there are two blocking variables. For example, in a field used for agricultural experiments, the level of moisture may vary across the field in one direction and the fertility in another. In an industrial experiment, suppose we wish to compare 4 production methods (the treatment) — A, B, C, and D. We have available 4 machines 1, 2, 3, and 4, and 4 operators, I, II, III, IV. A Latin square design is

	1	2	3	4
I	A	B	C	D
II	B	D	A	C
III	C	A	D	B
IV	D	C	B	A

Table 16.2: Latin Square

- Each treatment is assigned to each block once and only once.
- The design and assignment of treatments and blocks should be random.

We use the model

$$y_{ijk} = \mu + \tau_i + \beta_j + \gamma_k + \varepsilon_{ijk} \qquad i, j, k = 1, \ldots, t$$

To test for a treatment effect simply fit a model without the treatment effect and compare using the F-test. The Tukey pairwise CI's are

$$\hat{\tau}_l - \hat{\tau}_m \pm q_{t,(t-1)(t-2)} \hat{\sigma} \sqrt{1/t}$$

- The Latin square can be even more efficient than the RCBD provided that the blocking effects are sizable.
- We need to have both block sizes to be equal to the number of treatments. This may be difficult to achieve. Latin rectangle designs are possible by adjoining latin squares.
- The Latin square can be used for comparing 3 treatment factors. Only t^2 runs are required compared to the t^3 required if all combinations were run. (The downside is that you can't estimate the interactions if they exist). This is an example of a *fractional factorial*.
- The Latin square can be replicated if more runs are available.
- When there are 3 blocking variables, a Graeco-Latin square may be used but these rarely arise in practice.

An engineer wants to compare the qualities of raw materials from four suppliers, A, B, C, D. The raw material is used to produce a component whose breaking strength is measured. It takes an operator a whole day to make one component and there are 4 operators and 4 days on which the experiment will take place. A Latin square design is appropriate here where the operator and the day are the blocking effects.

```
> data(breaking)
> breaking
      y operator  day supplier
```

```
 1  810      op1 day1        B
 2 1080      op1 day2        C
...etc...
15 1025      op4 day3        D
16  900      op4 day4        C
```

We can check the Latin square structure:

```
> matrix(breaking$supplier,4,4)
     [,1] [,2] [,3] [,4]
[1,] "B"  "C"  "D"  "A"
[2,] "C"  "D"  "A"  "B"
[3,] "A"  "B"  "C"  "D"
[4,] "D"  "A"  "B"  "C"
```

Plot the data:

```
> plot(y ~ operator + day + supplier, breaking)
```

Examine the boxplots in Figure 16.4. There appear to be differences in suppliers but not in the two blocking variables. No outlier, skewness or unequal variance is apparent.

Now fit the Latin squares model:

```
> g <- lm(y ~ operator + day + supplier, breaking)
> anova(g)
Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq F value Pr(>F)
operator   3   7662    2554    0.41 0.7510
day        3  17600    5867    0.94 0.4759
supplier   3 371138  123712   19.93 0.0016
Residuals  6  37250    6208
```

Does it make a difference if we change the order of fitting? Let's see:

```
> anova(lm(y ~ day + supplier + operator, breaking))
Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq F value Pr(>F)
day        3  17600    5867    0.94 0.4759
supplier   3 371137  123712   19.93 0.0016
operator   3   7662    2554    0.41 0.7510
Residuals  6  37250    6208
```

They are the same because of the balanced design. We see that there is clear supplier effect but no evidence of an effect due to day or operator.

Now check the diagnostics

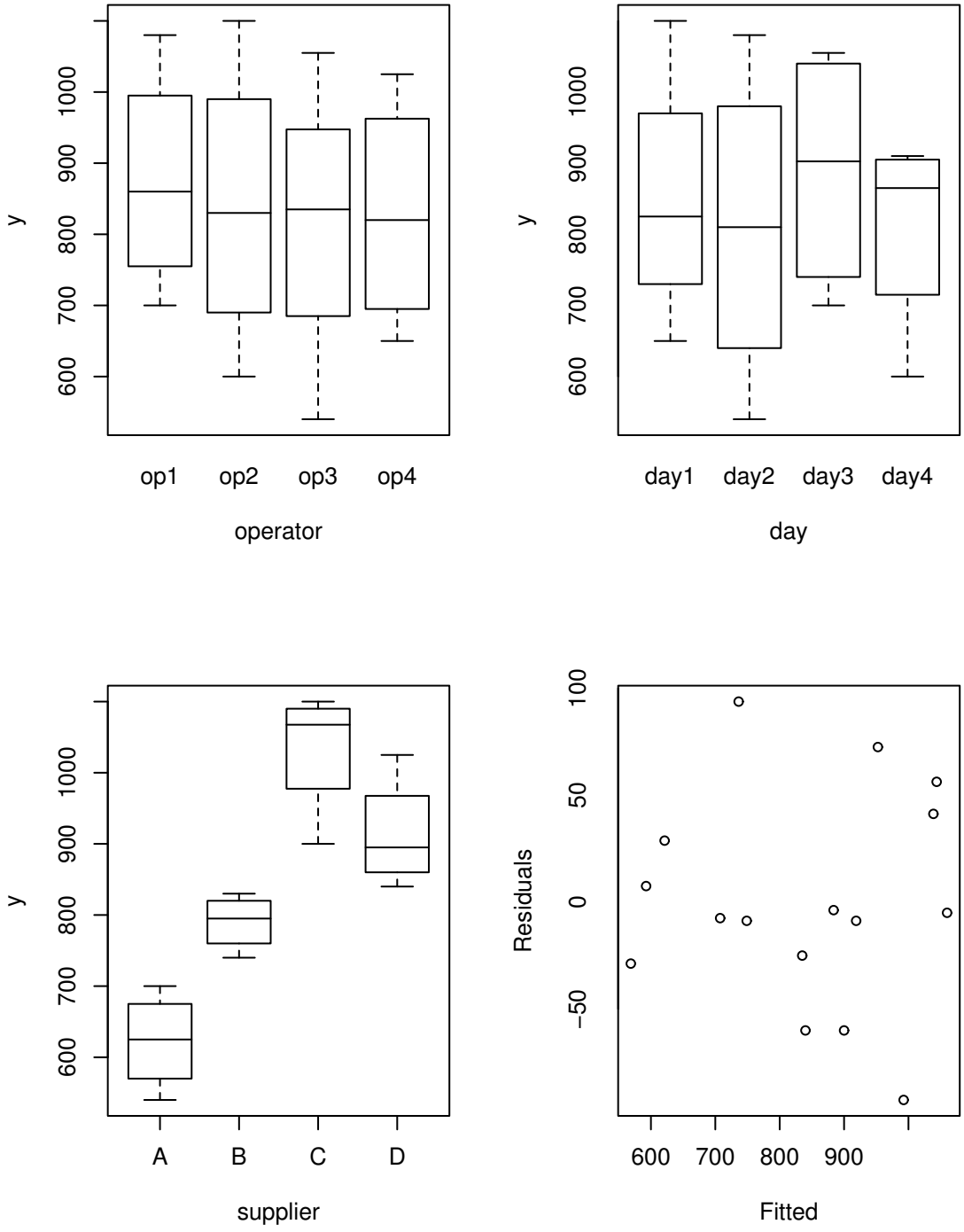

Figure 16.8: Latin square analysis

```
> plot(g$fit,g$res,xlab="Fitted",ylab="Residuals")
> qqnorm(g$res,ylab="Residuals")
```

I show only the residual-fitted plot which is fine as was the Q-Q plot. Now look at the estimates of the effects:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    667.5       62.3   10.72  3.9e-05
operatorop2    -35.0       55.7   -0.63  0.55302
operatorop3    -58.7       55.7   -1.05  0.33227
operatorop4    -46.2       55.7   -0.83  0.43825
dayday2        -40.0       55.7   -0.72  0.49978
dayday3         40.0       55.7    0.72  0.49978
dayday4        -40.0       55.7   -0.72  0.49978
supplierB      167.5       55.7    3.01  0.02381
supplierC      411.2       55.7    7.38  0.00032
supplierD      291.2       55.7    5.23  0.00196

Residual standard error: 78.8 on 6 degrees of freedom
Multiple R-Squared: 0.914,      Adjusted R-squared: 0.785
F-statistic: 7.09 on 9 and 6 degrees of freedom,        p-value: 0.0135
```

We see that Supplier C looks best followed by D. Is the difference significant though? Which suppliers in general are significantly better than others? We need the Tukey pairwise intervals to help determine this. The width of the bands calculated in the usual manner:

```
> qtukey(0.95,4,6)*55.7/sqrt(2)
[1] 193
```

The width of the interval is 193 - what can we say about differences between suppliers? We can make a handy table of the supplier differences:

```
> scoefs <- c(0,g$coef[8:10])
> outer(scoefs,scoefs,"-")
                supplierB supplierC supplierD
           0.00   -167.50   -411.25   -291.25
supplierB 167.50      0.00   -243.75   -123.75
supplierC 411.25    243.75      0.00    120.00
supplierD 291.25    123.75   -120.00      0.00
```

We see that the (A,B), (B,D) and (D,C) differences are not significant at the 5% level. Notice that it would not be reasonable to include that A is no different from C by chaining these comparisons together because each comparison is made using a statistical test where doubt exists about the conclusion and not a logical and definite assertion of equality.

If maximizing breaking strength is our aim, we would pick supplier C but if supplier D offered a better price we might have some cause to consider switching to D. The decision would need to be made with cost-quality trade-offs in mind.

How much more (or less) efficient is the Latin square compared to other designs? First compare to the completely randomized design:

```
> gr <- lm(y ~ supplier,breaking)
> (summary(gr)$sig/summary(g)$sig)^2
[1] 0.8391
```

We see that the LS is 16% less efficient than the CRD. Now compare to the blocked designs:

```
> gr <- lm(y ~ supplier+operator,breaking)
> (summary(gr)$sig/summary(g)$sig)^2
[1] 0.98166
> gr <- lm(y ~ supplier+day,breaking)
> (summary(gr)$sig/summary(g)$sig)^2
[1] 0.8038
```

We see that the Latin square turned out to be a bad choice of design because there was very little if any difference between the operators and days but we did not know that until after the experiment! Next time we will know better.

16.5 Balanced Incomplete Block design

For a complete block design, the block size is equal to the number of treatments. When the block size is less than the number of treatments, an incomplete block design must be used. For example, in the penicillin example, suppose 6 production processes were to be compared but each batch of material was only sufficient for four runs.

In an incomplete block design, the treatments and blocks are *not* orthogonal. Some treatment contrasts will not be identifiable from certain block contrasts - this is an example of *confounding*. This means that those treatment contrasts effectively cannot be examined. In a *balanced incomplete block design*, all the pairwise differences are identifiable and have the same standard error. Pairwise differences are more likely to be interesting than other contrasts. Here is an example:

We have 4 treatments (t=4) A,B,C,D and the block size, k = 3 and there are $b = 4$ blocks. Therefore, each treatment appears $r = 3$ times in the design. One possible BIB design is

1	A	B	C
2	A	B	D
3	A	C	D
4	B	C	D

Table 16.3: BIB design

Each pair of treatments appears in the same block $\lambda = 2$ times — this feature enables simple pairwise comparison. For a BIB design, we require

$$\begin{aligned} b &\geq t > k \\ rt &= bk = n \\ \lambda(t-1) &= r(k-1) \end{aligned}$$

This last relation holds because the number of pairs in a block is $k(k-1)/2$ so the total number of pairs must be $bk(k-1)/2$. On the other hand the number of treatment pairs is $t(t-1)/2$. The ratio of these two quantities must be λ.

Since λ has to be integer, a BIB design is not always possible even when the first two conditions are satisfied. For example, consider $r = 4, t = 3, b = 6, k = 2$ then $\lambda = 2$ which is OK but if $r = 4, t = 4, b = 8, k = 2$ then $\lambda = 4/3$ so no BIB is possible. (Something called a partially balanced incomplete block design can then be used). BIB's are also useful for competitions where not all contestants can fit in the same race.

The model we fit is the same as for the RCBD:

$$y_{ij} = \mu + \tau_i + \rho_j + \varepsilon_{ij}$$

A nutrition specialist studied the effects of six diets, a, b, c, d, e, and f on weight gain of domestic rabbits. When rabbits reached 4 weeks of age they were assigned to a diet. It was decided to block on litters of rabbits but from past experience about sizes of litters, it was felt that only 3 uniform rabbits could be selected from each available litter. There were ten litters available forming blocks of size three. Each pair of diets appear in the same block twice. Examine the data.

```
> data(rabbit)
> rabbit
   block treat gain
 1    b1     f 42.2
 2    b1     b 32.6
etc.
30   b10     a 37.3
```

We can see the BIB structure:

```
> matrix(rabbit$treat,nrow=3)
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,] "f"  "c"  "c"  "a"  "e"  "b"  "d"  "a"  "d"  "f"
[2,] "b"  "a"  "f"  "e"  "c"  "f"  "a"  "e"  "b"  "d"
[3,] "c"  "b"  "d"  "c"  "d"  "e"  "b"  "f"  "e"  "a"
```

Now plot the data:

```
> plot(gain ~ block + treat, rabbit)
```

See Figure 16.5. What do you conclude? Now fit the model:

```
> g <- lm(gain ~ block+treat,data=rabbit)
> anova(g)
Analysis of Variance Table

Response: gain
          Df Sum Sq Mean Sq F value  Pr(>F)
block      9    730      81    8.07 0.00025
treat      5    159      32    3.16 0.03817
Residuals 15    151      10
```

Changing the order of treatment and block:

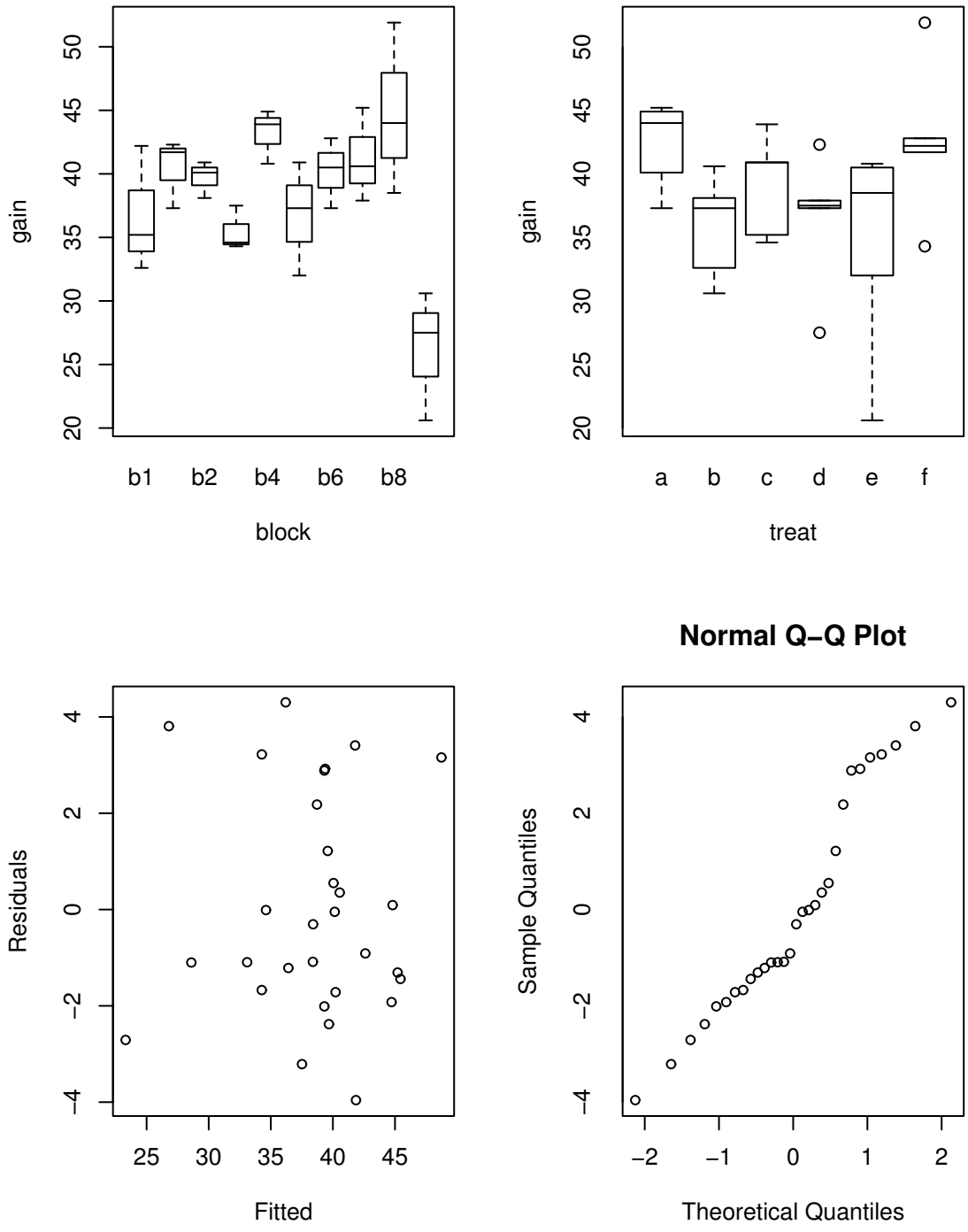

Figure 16.9: Balanced incomplete block analysis

```
> anova(lm(gain ~ treat+block,data=rabbit))
Analysis of Variance Table

Response: gain
          Df Sum Sq Mean Sq F value  Pr(>F)
treat      5    293      59    5.84 0.00345
block      9    596      66    6.59 0.00076
Residuals 15    151      10
```

Does make a difference because the design is not orthogonal because of the incompleteness. Which table is appropriate for testing the treatment effect or block effect? The first one, because we want to test for a treatment effect after the blocking effect has been allowed for.

Now check the diagnostics

```
> plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals")
> qqnorm(g$res)
```

Which treatments differ? We need to do pairwise comparisons. Tukey pairwise confidence intervals are easily constructed:

$$\hat{\tau}_l - \hat{\tau}_m \pm \frac{q_{t,n-b-t+1}}{\sqrt{2}}\sqrt{\frac{2k}{\lambda t}}\hat{\sigma}$$

First we figure out the difference between the treatment effects:

```
> tcoefs <- c(0,g$coef[11:15])
> outer(tcoefs,tcoefs,"-")
                 treatb   treatc    treatd treate  treatf
       0.000000  1.7417 -0.40000 -0.066667 5.2250 -3.3000
treatb -1.741667  0.0000 -2.14167 -1.808333 3.4833 -5.0417
treatc  0.400000  2.1417  0.00000  0.333333 5.6250 -2.9000
treatd  0.066667  1.8083 -0.33333  0.000000 5.2917 -3.2333
treate -5.225000 -3.4833 -5.62500 -5.291667 0.0000 -8.5250
treatf  3.300000  5.0417  2.90000  3.233333 8.5250  0.0000
```

Now we want the standard error for the pairwise comparisons:

```
> summary(g)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  36.0139     2.5886   13.91  5.6e-10
blockb10      3.2972     2.7960    1.18   0.2567
blockb2       4.1333     2.6943    1.53   0.1458
blockb3      -1.8028     2.6943   -0.67   0.5136
blockb4       8.7944     2.7960    3.15   0.0067
blockb5       2.3056     2.7960    0.82   0.4225
blockb6       5.4083     2.6943    2.01   0.0631
blockb7       5.7778     2.7960    2.07   0.0565
blockb8       9.4278     2.7960    3.37   0.0042
blockb9      -7.4806     2.7960   -2.68   0.0173
```

```
treatb       -1.7417     2.2418   -0.78   0.4493
treatc        0.4000     2.2418    0.18   0.8608
treatd        0.0667     2.2418    0.03   0.9767
treate       -5.2250     2.2418   -2.33   0.0341
treatf        3.3000     2.2418    1.47   0.1617

Residual standard error: 3.17 on 15 degrees of freedom
Multiple R-Squared: 0.855,      Adjusted R-squared: 0.72
F-statistic: 6.32 on 14 and 15 degrees of freedom,      p-value: 0.000518
```

We see that the standard error for the pairwise comparison is 2.24. This can also be obtained as $\sqrt{\frac{2k}{\lambda t}}\hat{\sigma}$:

```
> sqrt((2*3)/(2*6))*3.17
[1] 2.2415
```

Notice that all the treatment standard errors are equal because of the BIB. If the roles of blocks and treatments were reversed, we see that the design would not be balanced and hence the unequal standard errors for the blocks.

Now compute the Tukey critical value:

```
> qtukey(0.95,6,15)
[1] 4.5947
```

So the intervals have width

```
> 4.59*2.24/sqrt(2)
[1] 7.2702
```

We check which pairs are significantly different:

```
> abs(outer(tcoefs,tcoefs,"-")) > 7.27
            treatb treatc treatd treate treatf
      FALSE  FALSE  FALSE  FALSE  FALSE  FALSE
treatb FALSE  FALSE  FALSE  FALSE  FALSE  FALSE
treatc FALSE  FALSE  FALSE  FALSE  FALSE  FALSE
treatd FALSE  FALSE  FALSE  FALSE  FALSE  FALSE
treate FALSE  FALSE  FALSE  FALSE  FALSE   TRUE
treatf FALSE  FALSE  FALSE  FALSE   TRUE  FALSE
```

Only the e-f difference is significant.

How much better is this blocked design than the CRD? We compute the relative efficiency:

```
> gr <- lm(gain ~ treat,rabbit)
> (summary(gr)$sig/summary(g)$sig)^2
[1] 3.0945
```

Blocking was well worthwhile here.

16.6 Factorial experiments

Suppose we have

- Factors $\alpha, \beta, \gamma \ldots$
- with levels $l_\alpha, l_\beta, l_\gamma \ldots$

A *full* factorial experiment has at least one run for each combination of the levels. The number of combinations is $l_\alpha l_\beta l_\gamma \ldots$ which could easily be very large. The biggest model for a full factorial contains all possible interaction terms which may be of quite high order.

Advantages of factorial designs

1. If no interactions are significant, we get several one-way experiments for the price of one. Compare this with doing a sequence of one-way experiments.
2. Factorial experiments are efficient — it is often better to use replication for investigating another factor instead. For example, instead of doing a 2 factor experiment with replication, it is often better to use that replication to investigate another factor.

Disadvantage of factorial designs

Experiment may be too large and so cost too much time or money.

Analysis

The analysis of full factorial experiments is an extension of that used for the two-way anova. Typically, there is no replication due to cost concerns so it is necessary to assume that some higher order interactions are zero in order to free up degrees of freedom for testing the lower order effects. Not many phenomena require a precise combination of several factors so this is not unreasonable.

Fractional Factorials

Fractional factorials use only a fraction of the number of runs in a full factorial experiment. This is done to save the cost of the full experiment or because the experimental material is limited and only a few runs can be made. It is often possible to estimate the lower order effects with just a fraction. Consider an experiment with 7 factors, each at 2 levels

	mean	main	2-way	3-way	4	5	6	7
no. of pars	1	7	21	35	35	21	7	1

Table 16.4: No. of parameters

If we are going to assume that higher order interactions are negligible then we don't really need $2^7 = 128$ runs to estimate the remaining parameters. We could run only a quarter of that, 32, and still be able to estimate main and 2-way effects. (Although, in this particular example, it is not possible to estimate all the two-way interactions uniquely. This is because, in the language of experimental design, there is no available resolution V design, only a resolution IV design is possible.)

A Latin square where all predictors are considered as factors is another example of a fractional factorial.

In fractional factorial experiments, we try to estimate many parameters with as little data as possible. This means there is often not many degrees of freedom left over. We require that σ^2 be small, otherwise there will be little chance of distinguishing significant effects. Fractional factorials are popular in engineering applications where the experiment and materials can be tightly controlled. In the social sciences and medicine,

the experimental materials, often human or animal, are much less homgenous and less controllable so σ^2 tends to be larger. In such cases, fractional factorials are of no value.

Fractional factorials are popular in product design because they allow for the screening of a large number of factors. Factors identified in a screening experiment can then be more closely investigated.

Speedometer cables can be noisy because of shrinkage in the plastic casing material, so an experiment was conducted to find out what caused shrinkage. The engineers started with 15 different factors: liner O.D., liner die, liner material, liner line speed, wire braid type, braiding tension, wire diameter, liner tension, liner temperature, coating material, coating die type, melt temperature, screen pack, cooling method and line speed, labelled a through o. Response is percentage shrinkage per specimen. There were two levels of each factor. A full factorial would take 2^{15} runs, which is highly impractical so a design with only 16 runs was used where the particular runs have been chosen specially so as to estimate the the mean and the 15 main effects. We assume that there is no interaction effect of any kind. The purpose of such an experiment is to screen a large number of factors to identify which are important. Examine the data. The + indicates the high level of a factor, the - the low level. The data comes from Box, Bisgaard, and Fung (1988)

Read in and check the data.

```
> data(speedo)
> speedo
   h d l b j f n a i e m c k g o      y
 1 - - + - + + - - + + - + - - + 0.4850
 2 + - - - - + + - - + + + + - - 0.5750
 3 - + - - + - + - + - + + - + - 0.0875
 4 + + + - - - - - - - - + + + + 0.1750
 5 - - + + - - + - + + - - + + - 0.1950
 6 + - - + + - - - - + + - - + + 0.1450
 7 - + - + - + - - + - + - + - + 0.2250
 8 + + + + + + + - - - - - - - - 0.1750
 9 - - + - + + - + - - + - + + - 0.1250
10 + - - - - + + + + - - - - + + 0.1200
11 - + - - + - + + - + - - + - + 0.4550
12 + + + - - - - + + + + - - - - 0.5350
13 - - + + - - + + - - + + - - + 0.1700
14 + - - + + - - + + - - + + - - 0.2750
15 - + - + - + - + - + - + - + - 0.3425
16 + + + + + + + + + + + + + + + 0.5825
```

Fit and examine a main effects only model:

```
> g <- lm(y ~ .,speedo)
> summary(g)
Residuals:
ALL 16 residuals are 0: no residual degrees of freedom!

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.582500
h           -0.062188
d           -0.060938
```

```
l           -0.027188
b            0.055937
j            0.000938
f           -0.074062
n           -0.006562
a           -0.067813
i           -0.042813
e           -0.245312
m           -0.027813
c           -0.089687
k           -0.068438
g            0.140312
o           -0.005937

Residual standard error: NaN on 0 degrees of freedom
Multiple R-Squared:    1,       Adjusted R-squared:  NaN
F-statistic:  NaN on 15 and 0 degrees of freedom,       p-value:  NaN
```

Why are there no degrees of freedom? Why do we have so many "NA"'s in the display? Because there are as many parameters as cases.

It's important to understand the coding here, so look at the X-matrix.

```
> model.matrix(g)
   (Intercept) h d l b j f n a i e m c k g o
 1           1 1 1 0 1 0 0 1 1 0 0 1 0 1 1 0
...etc...
```

We see that "+" is coded as 0 and "-" is coded as 1. This unnatural ordering is because of their order in the ASCII alphabet.

We don't have any degrees of freedom so we can't make the usual F-tests. We need a different method. Suppose there were no significant effects and the errors are normally distributed. The estimated effects would then just be linear combinations of the errors and hence normal. We now make a normal quantile plot of the main effects with the idea that outliers represent significant effects. The `qqnorm()` function is not suitable because we want to label the points.

```
> coef <- g$coef[-1]
> i <- order(coef)
> plot(qnorm(1:15/16),coef[i],type="n",xlab="Normal Quantiles",
   ylab="Effects")
> text(qnorm(1:15/16),coef[i],names(coef)[i])
```

See Figure 16.6. Notice that "e" and possibly "g" are extreme. Since the "e" effect is negative, the + level of "e" increases the response. Since shrinkage is a bad thing, increasing the response is not good so we'd prefer what ever "wire braid" type corresponds to the - level of e. The same reasoning for g leads us to expect that a larger (assuming that is +) would decrease shrinkage.

A half-normal plot is better for detecting extreme points. This plots the sorted absolute values against $\Phi^{-1}((n+i)/(2n+1))$. Thus it compares the absolute values of the data against the upper half of a normal distribution. We don't particularly care if the coefficients are not normally distributed, it's just the extreme

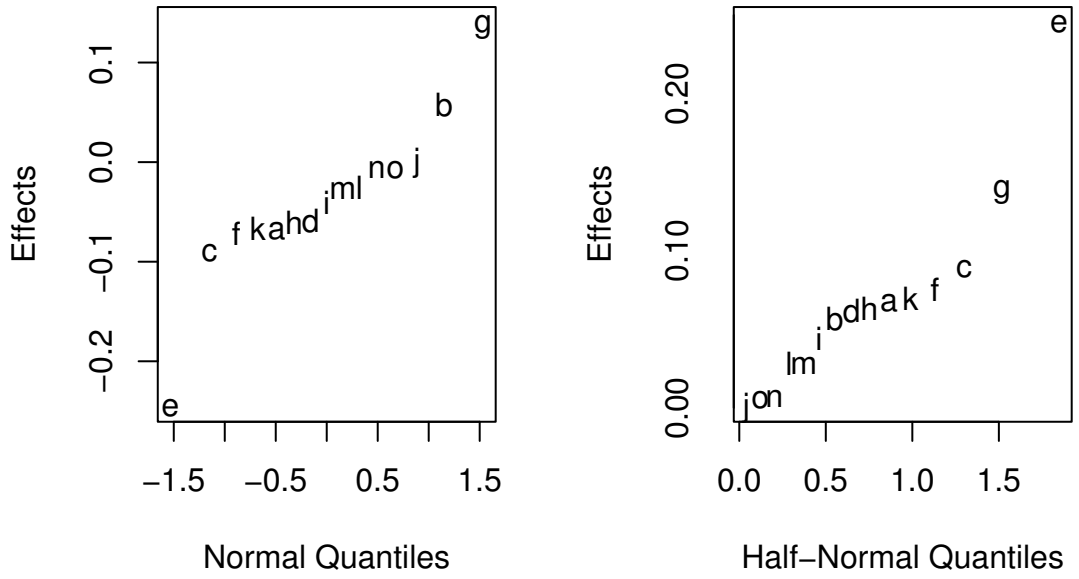

Figure 16.10: Fractional Factorial analysis

cases we want to detect. Because the half-normal folds over the ends of a QQ plot it "doubles" our resolution for the detection of outliers.

```
> coef <- abs(coef)
> i <- order(coef)
> plot(qnorm(16:30/31),coef[i],type="n",xlab="Half-Normal Quantiles",
 ylab="Effects")
> text(qnorm(16:30/31),coef[i],names(coef)[i])
```

We might now conduct another experiment focusing on the effect of "e" and "g".