7

Graphical Insight and Data Analysis for the 2,2,2 Crossover Design

Bill Pikounis
Merck Research Laboratories, Rahway, NJ, USA

Thomas E. Bradstreet
Merck Research Laboratories, Blue Bell, PA, USA

Steven P. Millard
Probability, Statistics & Information, Seattle, WA, USA

7.1 Introduction

S-PLUS code is presented for the graphical insight into, and the statistical analysis of, a two-treatment, two-period, two-treatment-sequence, or 2,2,2 crossover design. In this introductory section, we describe the 2,2,2 crossover design and its uses in the pharmaceutical industry with emphasis on food interaction studies. We also introduce a specific example and a dataset which will be used pedagogically throughout the chapter. In Section 7.2, we provide a brief introduction to data management in S-PLUS demonstrating just enough manipulations to facilitate the graphical methods and data analyses which follow. Section 7.3 presents a series of graphs for the initial exploration and discovery stage of the analysis of the 2,2,2 crossover design. In Section 7.4, we perform the usual normal theory ANOVA and provide a clear and decision-oriented summary and inference plot. Section 7.5 presents several graphical tools for the "visualization of the ANOVA" and a subsequent model fit assessment, and we end with a summary in Section 7.6.

7.1.1 The 2,2,2 Crossover Design

The 2,2,2 crossover design has been a standard tool of medical researchers for decades. Although it has been used frequently in the pharmaceutical industry for initial studies investigating the safety and efficacy of new drugs, this design is probably best known for its use in evaluating the pharmacokinetics of a drug; most notably average bioequivalence, relative bioavailability, drug interaction, alcohol interaction, and food interaction.

In the simplest form of the 2,2,2 crossover design, half of the subjects are randomized to receive one of two treatments in the first treatment period, and the second of two treatments in the second treatment period. The other half of the subjects receive the two treatments in the reverse sequence. Between the two treatments in each treatment sequence, there is a "washout" period of adequate length to, in concept, prevent the effect of whichever treatment is given in the first treatment period from carrying over to affect the treatment given in the second treatment period.

The 2,2,2 crossover design allows for the evaluation of three effects: treatments, periods, and carryover, although carryover is confounded with both sequence (sometimes referred to as subject group) effects and treatment-by-period interaction. Whether or not to, and how to, evaluate carryover effects, and how to proceed if these are suspected, has been a controversial issue. For further details, see Jones and Kenward (1989) and Senn (1993).

7.1.2 Food Interaction Studies

In some cases, taking a drug with food increases the amount of the drug in the blood stream. This increase may be considered medically safe and provide additional therapeutic benefits to the patient. Or, the increase may be viewed as a potential toxicity problem, regardless of the potential benefit. In other cases, taking the drug with food decreases the amount of drug in the blood stream which may decrease the drug's effectiveness. In still other cases, taking the drug with food may not substantially affect the amount of drug which is delivered to the blood stream, and for all medically meaningful purposes, the effect of the drug is therapeutically equivalent when it is taken either with or without food. Therefore, it is important to evaluate the degree, if any, that a drug interacts with food.

The primary medical objective of a food interaction study is to "prove" the absence of a clinically meaningful food interaction. The clinical investigators may assume initially that there exists some degree of interaction, but it is their intention to show that its magnitude is not of clinical importance, possibly being nonexistent. To accomplish this, food interaction studies investigate the pharmacokinetic properties (ADME: absorption, distribution, metabolism, excretion) of a drug. These properties help to clarify when, where, and how the body processes and uses the drug. By measuring these properties, the bioavailability of the drug both in the presence and the absence of food is compared.

The bioavailability of a drug is often characterized by summarizing its plasma concentration versus time course in the blood with three measurements:

1. Area under the plasma concentration versus time curve (AUC), a measure of total absorption.
2. Maximum plasma concentration (C_{max}), a measure of the extent of absorption.
3. Time to maximum plasma concentration (T_{max}), a measure of the rate of absorption.

If the bioavailability of the drug is clinically similar when taken either with (+) or without (–) food as evaluated through all or a subset of AUC, C_{max}, and T_{max} as is appropriate for a particular drug, then by clinical extrapolation from the pharmacokinetic measurements to expected clinical responses, it should not matter therapeutically whether or not the drug is administered with food. Thus, in a food interaction study, it is important to estimate the relative (Fed versus Fasted) bioavailability of a drug administered both with and without food.

Relative bioavailability can be estimated following a paradigm similar to that for evaluating average bioequivalence. Confidence intervals for the true proportional differences in the mean Fed (+) and mean Fasted (–) values of targeted pharmacokinetic variables (e.g., AUC and C_{max}) can be calculated. Further details on average bioequivalence can be found in Chow and Liu (2000).

7.1.3 Our Example

In our example, eight healthy male subjects participated in a food interaction study to evaluate the magnitude of the food interaction of a new hypertensive therapy, Drug P, and its metabolite, Drug M. Four subjects (numbers 1, 2, 5, 7) were randomized to take Drug P with food in the first treatment period but without food in the second treatment period. The other four subjects (numbers 3, 4, 6, 8) took Drug P without food in the first treatment period but with food in the second treatment period. Four plasma concentration versus time curves were constructed for each of the eight subjects; one for Drug P and one for Drug M in each of the two treatment periods. Drug concentration values were assayed from plasma samples taken at 0, 10, 20, 30, 40, and 50 min, and 1, 1.25, 1.5, 1.75, 2, 3, 4, 5, 6, 8, 10, 12, 18, 24, 30, and 36 h after dosing. AUC was estimated from zero hours to 36 h using the trapezoidal rule, and then using extrapolation from 36 h to infinity incorporating the elimination rate constant. C_{max} and T_{max} were simply observed from the plasma concentration versus time curve. Four AUC, C_{max}, and T_{max} values were obtained for each of the eight subjects in the study, corresponding to the Fed and Fasted states for each of the parent drug and the metabolite. Due to space considerations, in Section 7.2 we present only a subset of these data to analyze, namely the AUC (ng × h/mL) data for the parent drug. The full data set can be retrieved from Part 2, Table 5, at the web site http://www.villanova.edu/~tshort/Bradstreet/ and it is printed in Bradstreet (1992).

7.2 Data Management

In this section, we provide a brief introduction to data management in S-PLUS for the graphical methods and data analysis which follow in Sections 7.3, 7.4, and 7.5. Specifically, we read the data into S-PLUS using a comma-separated text file, and create a data frame named `food.df`. This is a convenient way to begin the data management process for this type of data as a tabular form with

cases as rows and variables as columns. The first row of the `food.csv` file printed out below will be a header of character strings that S-PLUS will use to label the variables in the resulting data frame:

```
subj,seq,AUC.Fed,AUC.Fasted
1,+/-,809.44,967.82
2,+/-,428,746.45
3,-/+,757.71,901.11
4,-/+,906.83,1146.96
5,+/-,712.24,678.16
6,-/+,561.77,745.51
7,+/-,511.84,568.98
8,-/+,756.6,852.86
```

The command

```
> food.df.orig <- read.table("food.csv", header = T,
    sep = ",", as.is = T)
```

produces an S-PLUS data frame named `food.df.orig`. The `as.is=T` argument ensures that character data variables do not get converted to factors here. We'll do this later. To help us later in the chapter with producing graphs, formal statistical analyses, and subsequent model checking, some further manipulations are performed:

```
> food.df <- data.frame(
    subj = rep(food.df.orig$subj, each = 2),
    seq = rep(food.df.orig$seq, each = 2),
    trt = factor(rep(c("Fed", "Fasted"), times = 8),
      levels = c("Fed", "Fasted")),
    per=c(1, 2, 1, 2, 2, 1, 2, 1, 1, 2, 2, 1, 1, 2, 2, 1),
    AUC = as.vector(
      t(food.df.orig[ , c("AUC.Fed", "AUC.Fasted")])))
```

The new data frame `food.df` is created by binding columns of data vectors with the help of functions to replicate values (`rep`), and to transpose (`t`) and unwind (`as.vector`) a rectangular block of data in column order.

We also create some factor and numeric variables to impose a desired comparative order when plotting and tabulating the data. The first three statements below create nominal factors so that S-PLUS analysis functions do not interpret values like the subject numbers 1, 2, …, 8 as having any ordered meaning. The last two statements create numeric codings for sequence (1 = "Fed/Fasted", 2 = "Fasted/Fed") and treatment (1 = "Fed", 2 = "Fasted").

```
> food.df$subjf <- factor(food.df$subj,
    levels = as.character(1:8))
> food.df$perf <- factor(food.df$per,
    levels = as.character(1:2))
```

```
> food.df$seqf <- factor(food.df$seq,
    levels=c("+/-", "-/+"))
> food.df$seqn <- as.numeric(food.df$seqf)
> food.df$trtn <- as.numeric(food.df$trt)
```

Lastly we sort the data frame by sequence number and get a listing by typing its name

```
> food.df <- food.df[order(food.df$seqn), ]
>food.df
   subj seq    trt per     AUC subjf perf seqf seqn trtn
 1    1 +/-    Fed   1  809.44     1    1  +/-    1    1
 2    1 +/- Fasted   2  967.82     1    2  +/-    1    2
 3    2 +/-    Fed   1  428.00     2    1  +/-    1    1
 4    2 +/- Fasted   2  746.45     2    2  +/-    1    2
 9    5 +/-    Fed   1  712.24     5    1  +/-    1    1
10    5 +/- Fasted   2  678.16     5    2  +/-    1    2
13    7 +/-    Fed   1  511.84     7    1  +/-    1    1
14    7 +/- Fasted   2  568.98     7    2  +/-    1    2
 5    3 -/+    Fed   2  757.71     3    2  -/+    2    1
 6    3 -/+ Fasted   1  901.11     3    1  -/+    2    2
 7    4 -/+    Fed   2  906.83     4    2  -/+    2    1
 8    4 -/+ Fasted   1 1146.96     4    1  -/+    2    2
11    6 -/+    Fed   2  561.77     6    2  -/+    2    1
12    6 -/+ Fasted   1  745.51     6    1  -/+    2    2
15    8 -/+    Fed   2  756.60     8    2  -/+    2    1
16    8 -/+ Fasted   1  852.86     8    1  -/+    2    2
```

7.3 Initial Exploration and Discovery

Section 7.3 presents graphics for the initial exploration and discovery stage of the data analysis of the 2,2,2 crossover design. Section 7.3.1 presents individual subject (sometimes called “spaghetti”) plots ordered both by treatment (Fed, Fasted) within treatment sequence, and also by study period within treatment sequence. Section 7.3.2 presents a graphic useful for evaluating marginal treatment and variance effects. Section 7.3.3 presents a series of three graphs which provide an initial look at the sample bivariate relationships. Section 7.3.4 presents a graph which provides a preliminary look at the 2,2,2 crossover ANOVA.

7.3.1 Individual Subject Plots

Since there are only eight subjects, plotting individual subject profiles over the two study periods should be informative. When there are many subjects, the individual subject profiles may be less informative.

We first set up some parameters for the graphical displays. As opposed to the default style of enclosed box and tick-labels, we request that axis labels be more extreme than any data values for both the *x*-axis and *y*-axis, that the box layout in the plot area is open L-shaped, and that the plotting region is square:

```
> par(xaxs = "e", yaxs = "e", bty = "l", pty = "s")
```

The endpoint AUC (ng × h/mL) is analyzed in the log scale, so let us add the log-transformed endpoint to the data frame.

```
> food.df$logAUC <- log(food.df$AUC)
```

To call variables in the `food.df` data frame more easily, we attach `food.df` to the S-PLUS search path:

```
> attach(food.df)
```

We want axes in the log scale, but the numeric labels to be in the original scale. The `axislog` function we create (see Appendix 7.A.1) provides a full range of tick-marks that the usual `log="y" or "x"` option in the `plot` function typically misses.

The series of calls starts with setting up the graph without plotting points or axes labels, so that we may later add customized symbols and text. The result is Figure 7.1, which displays the subjects' results ordered by treatment (Fed, Fasted) within each treatment sequence:

```
> plot(trtn, logAUC, xlim = c(0, 5), type = "n",
    axes = F, xlab = "", ylab = "AUC (ng x hr/ml)")
> axislog(AUC, line = 1, srt = 90, cex = 0.9)
> axis(1, at = c(1:2, 4:5), rep(levels(trt), 2), ticks = F)
> subjseq1 <- levels(subjf[seqn == 1, drop = T])
> for(i in seq(along = subjseq1)) {
    points(trtn[subj == subjseq1[i]],
      logAUC[subj == subjseq1[i]],
      pch = subjseq1[i], type = "b")
  }
> subjseq2 <- levels(subjf[seqn == 2, drop = T])
> for(i in seq(along = subjseq2)) {
    points(3 + trtn[subj == subjseq2[i]],
      logAUC[subj == subjseq2[i]],
      pch = subjseq2[i], type = "b")
  }
> mtext(side = 1, at = c(1.5, 4.5),
    text = c("Sequence 1", "Sequence 2"), line = 2)
```

In Figure 7.1, each line represents an individual subject. We see that AUC was less when the subjects received the Fed regimen, most notably for Subject number 2. The only exception was Subject number 5. With such a small sample size, we suspect that the results of Subjects number 2 and 5 may be quite

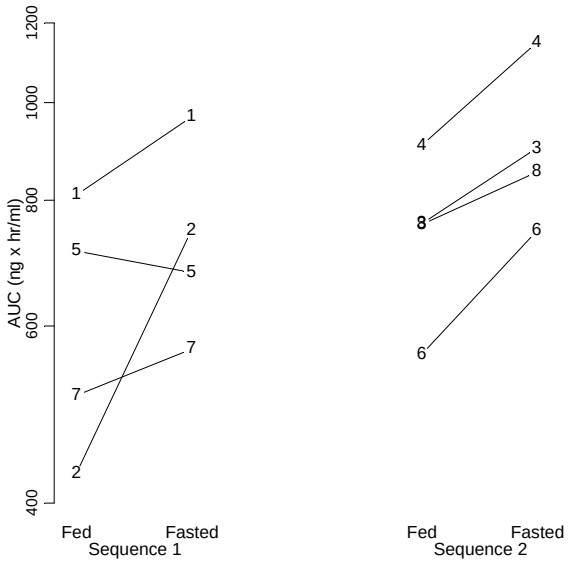


Figure 7.1. Subject plots ordered by treatment sequence.

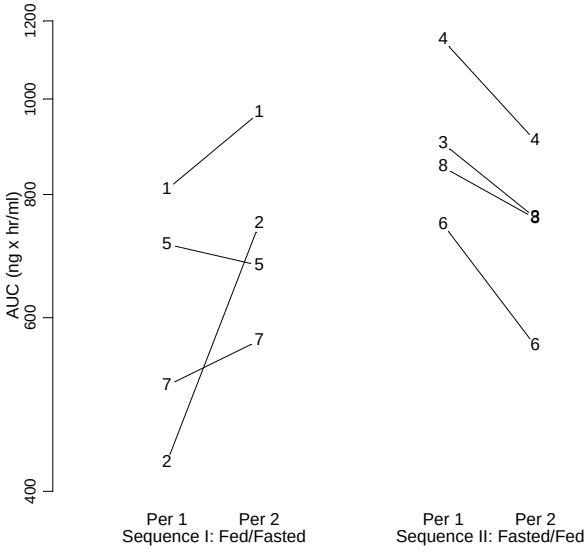


Figure 7.2. Subject plots ordered by period within treatment sequence.

influential on the normal-theory analysis which is presented later in Section 7.4. Also note that the AUC values for Subjects number 1, 2, 5, and 7 in Sequence 1 were somewhat less on average as compared to those for Subjects number 3, 4, 6, and 8 in Sequence 2, suggesting a modest sequence effect.

Figure 7.2 displays the subject profiles ordered by period within treatment sequence. This reflects the order in which the data were collected in the study, and can be insightful for evaluating period effects; and sometimes for discriminating between carryover effects, sequence effects, and treatment-by-period interaction. As in Figure 7.1, a sequence effect is suggested in that the log AUC values for the subjects in Sequence 2 are generally greater than those for the subjects in Sequence 1, regardless of the Fed or Fasted states. However, Figure 7.2 also suggests a treatment-by-period interaction as the log AUC values in the Fed and Fasted state are somewhat similar in Period 2, but the Fed values are much less than the Fasted values in Period 1. The code to accomplish this is very similar to that for Figure 7.1 and is left as an exercise for the reader.

7.3.2 Marginal Treatment and Variance Effects

We next present a point graph of the data sorted by treatment, which enables us to get a first glimpse of marginal location and variance.

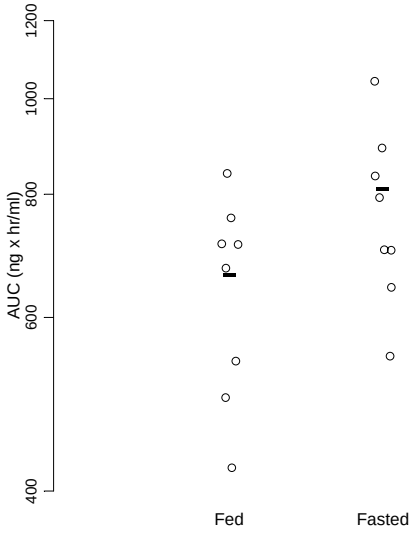


Figure 7.3. Marginal spread and location plot.

The code to produce this plot is given by:

```
> plot(jitter(trtn, factor = 3), logAUC, xlim = c(0, 3),
    axes = F, xlab = "", ylab = "AUC (ng x hr/ml)”,
    pch = 1)
> axislog(AUC, line = 1, srt = 90, cex = 0.9)
> axis(1, at = 1:2, labels = levels(trt),ticks = F)
> points(x=1:2, y = c(mean(logAUC[trt == "Fed"]),
    mean(logAUC[trt == "Fasted"])), pch = "-", cex = 3)
```

We applied the `jitter` function (Chambers et al., 1983) to the *x*-axis values to alleviate overlap of points by adding enough random noise without distorting the structure of the data. The horizontal bars in Figure 7.3 display the geometric means of the treatment groups. Based on a marginal evaluation, the sample variability appears similar between the two treatments, and the sample location differs as we conjectured from examining Figures 7.1 and 7.2. So we have no data-driven reason to suspect unequal variances in the bivariate treatment population.

7.3.3 Bivariate Treatment and Other Relationships

Figure 7.4 is a scatter plot of the data pairs from the eight subjects. The symbols indicate which treatment sequence each subject received. In addition to assessing the location of the bivariate point cloud versus the diagonal line, it is important to identify both concordant and discordant outliers. In the food interaction framework, concordant outliers are those bivariate points that are distant from the center of the point cloud but vertically close to the diagonal line (and thus situated near either end of the diagonal line). These points represent those subjects whose paired responses (Fed versus Fasted AUC) are somewhat similar to each other for that subject, but are notably different (smaller–lower end of the diagonal line; larger–upper end of the diagonal line) than the magnitudes of the responses of the other subjects. Discordant outliers are those bivariate points which stray from the point cloud and stray from the diagonal line. These represent subjects whose paired responses are not similar to each other for that subject, and are also notably different from the responses of the other subjects. The last statement call is defined in Appendix 7.A.1, and is a wrapper that we will use repeatedly to place a key in the upper left or other corner of a graph:

```
> plot(logAUC[trt == "Fasted"], logAUC[trt == "Fed"],
    type = "n", axes = F,
    xlab = "Fasted AUC (ng x hr/ml)",
    ylab = "Fed AUC (ng x hr/ml)")
> axislog(AUC, line = 1, srt = 90, cex = 0.9)
> axislog(AUC, side = 1, line = 1)
> points(logAUC[trt == "Fasted" & seqn == 1],
    logAUC[trt == "Fed" & seqn == 1], pch = 1, cex = 1.2)
> points(logAUC[trt == "Fasted" & seqn == 2],
    logAUC[trt == "Fed" & seqn == 2], pch = 2, cex =1.2)
```

```
> # make symbols 20% larger than default
> abline(0, 1) # adds the bivariate identity line
> place.keyseq(0, 1)
```

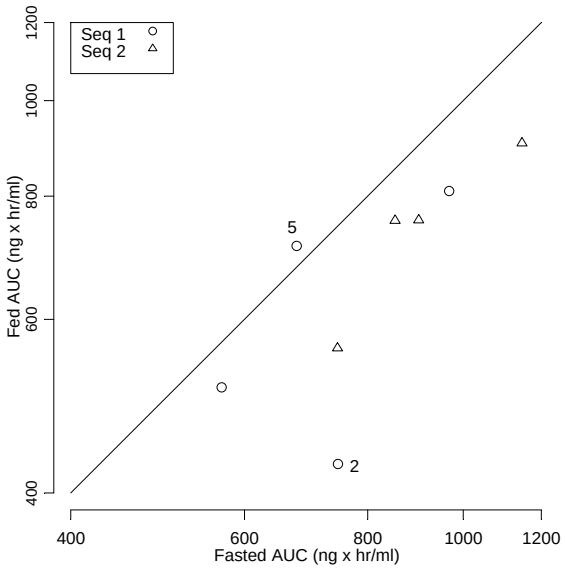


Figure 7.4. Scatter plot.

Again, we see that the Fed AUCs tend to be less than the Fasted AUCs, and the AUCs for subjects in Sequence 2 tend to be larger than those for subjects in Sequence 1, regardless of the Fed or Fasted state. For Subject number 2, who is a discordant outlier with Fed AUC = 428.00 and Fasted AUC = 746.45 ng × h/mL (see Section 7.2 data listing), the difference is relatively large. We identify the bivariate response for Subject number 2 on the plot with

```
> text(x = logAUC[subj == 2 & trt == "Fasted"] + .05,
    y = logAUC[subj == 2 & trt == "Fed"], "2", adj = 1)
```

and with analogous code we mark Subject number 5 as the lone point which is located above the bivariate identity line.

Another informative look at these data is a Tukey sum–difference plot (Cleveland, 1993; Tukey, 1977), which allows us to study the Fed versus Fasted relationship through differences (log Fed – log Fasted) and sums (log Fed + log Fasted) of the bivariate data points. In general, this plot clarifies magnitudes of treatment differences (ratios in our case) and permits discovery of trends across the observed ranges of the data. (Note how it is easier to judge differences from the flat zero line in Figure 7.5 as compared to the bivariate identity line in Figure 7.4.) But we will see in Section 7.5 that the Tukey sum–difference plot has

added value for the initial exploration of the data collected from a 2,2,2 crossover trial as it simultaneously displays information on the carryover and period effects, as well as suggests treatment effects. The code to generate Figure 7.5 is in Appendix 7.A.2.

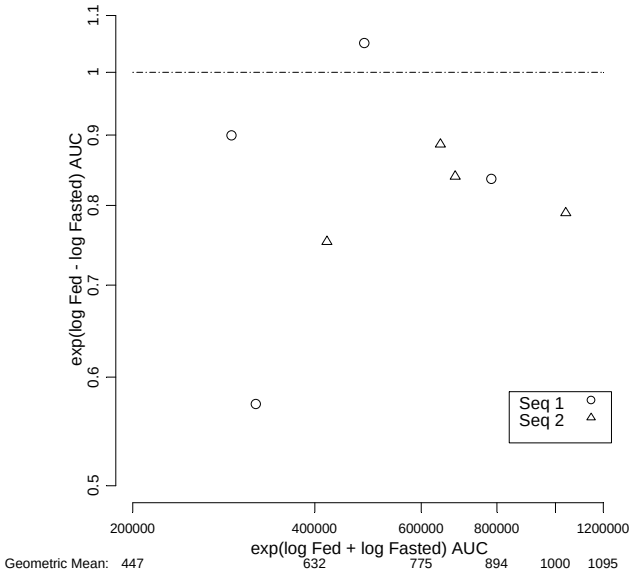


Figure 7.5. Tukey sum–difference plot.

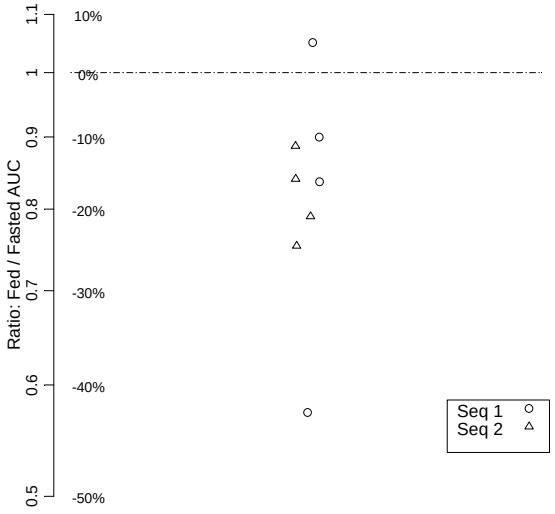


Figure 7.6. Plot of AUC ratios by sequence.

Figure 7.6 is a graph of the Fed/Fasted AUC ratios where the ratio points are labeled by sequence. The code is in Appendix 7.A.2.

It is of interest to observe what shape the sample distribution of ratios has, and how the sample distribution relates to a reference ratio of one. To aid study of the graph, labels in the percentage change scale were placed next to the y-axis ratio labels. We see that six of the eight subjects exhibited roughly a 10–25% decrease in Fed AUC relative to Fasted AUC, and one subject (number 2) exhibited a little more than a 40% decrease. In one subject (number 5), the Fed response increased relative to the Fasted response by about 5%.

7.3.4 A Preliminary Look at the 2,2,2 Crossover ANOVA

Figure 7.7 plots the individual Fed and Fasted AUC values by period along with the corresponding period geometric means. The lines connect the geometric means across periods by sequence. The code is in Appendix 7.A.2. Figure 7.7 indicates an inconsistency from Period 1 to Period 2 in the Fed and Fasted AUCs. In Period 1, the individual Fed AUCs are notably less than the Fasted AUCs; this is reflected in the corresponding geometric means. However, in Period 2, the two groups of individual Fed and Fasted AUC values are quite similar, and this is reflected by the similar geometric mean values which we also note are located centrally between the two geometric mean values from Period 1. Thus, Figure 7.7 alone suggests the presence of a treatment-by-period interaction.

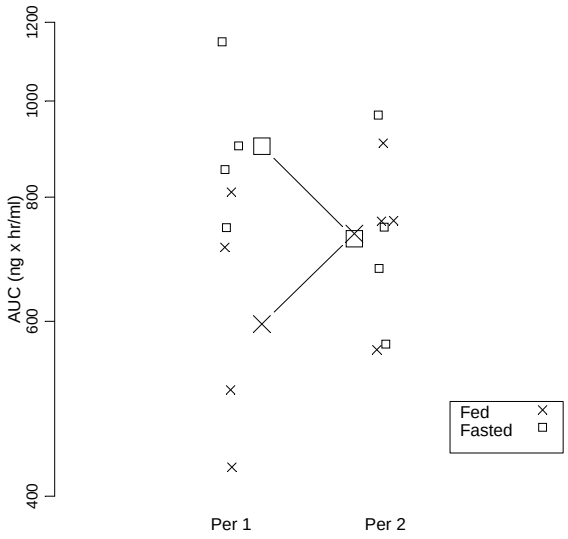


Figure 7.7. Geometric mean and individual responses by period.

7.4 Formal Data Analysis

In this section, we perform the usual normal-theory ANOVA for the 2,2,2 crossover design. For the full modeling and algebraic details, see Jones and Kenward (1989; pp. 9–10, 22–28, 30–34). The model representation is

$$Y_{ijk} = \mu + s_{ik} + \pi_j + \tau_{d[i,j]} + \lambda_{d[i,\,j-1]} + e_{ijk,} \tag{7.1}$$

where

- μ = the overall mean effect.
- s_{ik} = the effect of subject k in group i; $i = 1, 2$.
- π_j = the effect of period j; $j = 1, 2$.
- $\tau_{d[i,j]}$ = the effect of treatment administered in period j to group i.
- $\lambda_{d[i,\,j-1]}$ = the carryover effect of the treatment administered in period j to group i, where $\tau_{d[i,0]} = 0$ (no carryover in the first period).
- e_{ijk} = the random error term for subject k in period j in group i, and is assumed to be normally distributed with mean 0 and variance σ^2.

Given our exploratory graphical work in Section 7.3, we can anticipate what the normal-theory ANOVA may confirm. Specifically, Figures 7.1 to 7.7 indicate a treatment effect in that, on average, Fed AUCs appear to be less than Fasted AUCs. Figures 7.1, 7.2, and 7.4 suggest a modest sequence effect in that on average, AUCs are greater in the subjects in Sequence 2 as compared to Sequence 1. Figures 7.2 and 7.7 suggest a treatment-by-period interaction where in Period 1, on average and individually, Fed AUCs are less than Fasted AUCs but in Period 2 they are quite similar. Figures 7.2 and 7.7 also suggest that there will be no period effect in that, on average, AUCs are similar from Period 1 to Period 2 when disregarding treatment. So, in a relative sense, we might anticipate a smaller *p*-value for the test of treatment (Fed versus Fasted) effects, a smaller *p*-value for the test of carryover effects which is confounded with both sequence effects and treatment-by-period interaction, and a larger *p*-value for period effects.

The initial model fit for log AUC specifies that subjects within sequence constitute a partition term in the error structure:

```
> fit.food <- aov(logAUC ~ seqf + Error(subjf) + trt +
    perf, data=food.df)
```

Application of the `summary` command demonstrates the two-strata error structure (between and within subjects) of the 2,2,2 crossover design:

```
> summary(fit.food)
Error: subjf
          Df Sum of Sq   Mean Sq  F Value     Pr(F)
     seqf  1 0.1797439 0.1797439 2.096941 0.1977559
Residuals  6 0.5143031 0.0857172
```

```
Error: Within
          Df Sum of Sq   Mean Sq  F Value     Pr(F)
      trt  1 0.1605383 0.1605383 9.026330 0.0238727
     perf  1 0.0000232 0.0000232 0.001305 0.9723585
Residuals  6 0.1067133 0.0177856
```

According to the ANOVA results, the sequence term, `seqf`, is not statistically significant ($p = 0.198$). Based upon the p-value alone and the intrinsic aliasing of sequence, carryover, and treatment-by-period interaction in a single degree-of-freedom contrast, it might be concluded that none of these effects were an issue with these data. However, Figures 7.1 to 7.7 in Section 7.3 suggest the presence of a modest sequence effect or a treatment-by-period interaction. To choose between these two requires an examination of response-related covariates (e.g., age, gender, weight) as they are distributed among the subjects in the two treatment sequences. Biological carryover is unlikely as the washout period was planned to be of sufficient length, and plasma levels in all eight subjects at the beginning of Period 2 were observed to be zero.

Despite suspicion about a potentially meaningful sequence effect or a treatment-by-period interaction, we continue with the usual analyses of period and treatment effects to continue illustrating S-PLUS. Indeed, the ANOVA results support our initial graphical investigation: there is a statistically significant ($p = 0.024$) treatment effect which we know from our graphical investigations is due to the lower Fed than Fasted AUCs. And, there is no statistically significant ($p = 0.972$) period effect as was seen graphically, as on average, AUCs are not strikingly different from Period 1 to Period 2 regardless of treatment.

In Section 7.5 we present some 2,2,2 crossover ANOVA model checking and diagnostics, but for now let us presume that the underlying assumptions of normality and equal variances are tenable and proceed with formal estimation inference. Our parameter of interest is the proportional difference in AUC for the Fed regimen relative to the Fasted regimen. We need to construct a sample point estimate and a 90% confidence interval for the true value.

Although there are several approaches in S-PLUS to compute the needed quantities, we will use `model.tables` to compute the means and standard error of the difference in the means on the log scale, and we also use the fit above to calculate the critical value of the t-distribution:

```
> tblmeans <- model.tables(fit.food, type = "means", se=T)
> meandiff.se <- tblmeans$se$trt
> t.crit <- qt(0.95, df = fit.food$Within$df.residual)
```

This code is sufficient to compute the sample point estimate of the difference in log AUC and the 90% confidence interval for the true difference, Fed–Fasted, in the log scale. We exponentiate the log scale results to arrive at the corresponding results for the ratio, Fed/Fasted , in the original scale:

```
> meandiff <- diff(rev(tblmeans$tables$trt))
> orig.ratio <- exp(meandiff)
> # Use Delta method to estimate SE of orig.ratio
```

```
> orig.ratio.se <- orig.ratio * meandiff.se
> lcl.ratio <- exp(meandiff - t.crit * meandiff.se)
> ucl.ratio <- exp(meandiff + t.crit * meandiff.se)
```

See Appendix 7.A.3 for a few more statements that build on these to alternatively express the results as percent change in AUCs. We summarize these results and display them in a table:

```
> tbldiff <- matrix(c(orig.ratio, lcl.ratio, ucl.ratio,
    orig.pctchg, lcl.pctchg, ucl.pctchg), byrow = T,
    nrow = 2)
> # pctchg refers to percent change analogues to ratio.
> dimnames(tbldiff) <- list(c("Ratio", "Pct Chg"),
    c("Point Est", "Lower", "Upper"))

> # Needed to display title with table
> disply <- function()
  {
    cat("Differences with 95% Confidence Bounds", "\n")
    round(tbldiff,2)
  }

> disply()
Differences with 95% Confidence Bounds
        Point Est  Lower Upper
  Ratio      0.82   0.72  0.93
Pct Chg    -18.15 -28.10 -6.83
```

We use similar code to obtain the individual treatment geometric means and approximated standard errors based on the "Delta" method (Agresti, 1990, Ch. 12).

```
> indv.sem <- meandiff.se / sqrt(2)
> orig.means <- exp(tblmeans$tables$trt)
> orig.sems <-  orig.means * indv.sem
> tblindv <- matrix(c(orig.means, orig.sems), ncol = 2)
> dimnames(tblindv) <- list(c("Fed", "Fasted"),
    c("Geo Mean", "SEM"))

> round(tblindv, 2)
       Geo Mean   SEM
   Fed   662.00 31.21
Fasted   808.84 38.14
```

To summarize the results of our food interaction study, we construct Figure 7.8 which presents the results averaging across periods but also for each period separately, due to the suspected treatment-by-period interaction. The code for all this is in Appendix 7.A.3.

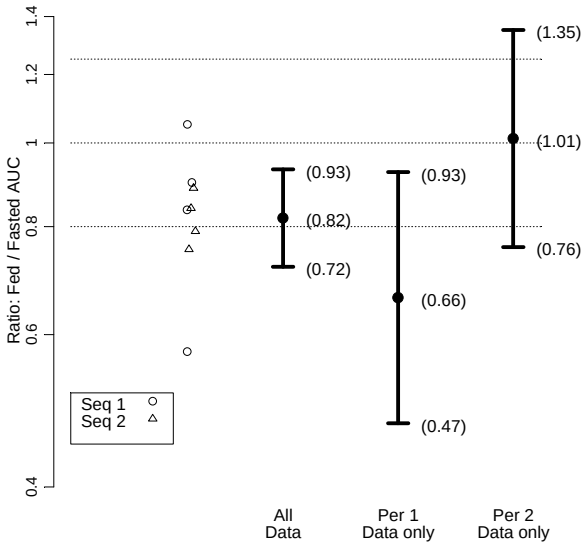


Figure 7.8. Treatment effect, Fed versus Fasted.

The usual average bioequivalence criteria based on ratios of data are shown by the dashed lines located at the *y*-axis ratio values of 0.80 and 1.25. The Fed/Fasted AUC ratios for the individual subjects are represented by the open circles (Sequence 1) and the open triangles (Sequence 2). The geometric mean ratio estimated from all of the data is represented by the solid dot with value 0.82, and the corresponding 90% confidence interval (0.72, 0.93) for the true proportioned difference in mean Fed AUC and mean Fasted AUC is represented by the bold error bars. These results indicate that, on average, the Fed and Fasted states are not bioequivalent; Fed AUC is, on average, less than Fasted AUC.

Also shown are the ratios (Fed/Fasted) of geometric means (solid dots) and corresponding 90% confidence intervals (bold error bars) which were computed separately using the between-subject variability from each study period. This additional part of the display is of interest when there is reasonable evidence of unequal carryover effects (usually only the first period is plotted), or in the case of a suspected treatment-by-period interaction. The results are not surprising given our previous graphical and statistical analyses. The Period 1 data suggest inequivalence of the Fed and Fasted states; the Period 2 data are consistent with equivalence. Thus we have summarized the major individual, mean, and inferential results in a single graphic display which is easily understood by the clinical, pharmacokinetic, and statistical professionals involved in bioequivalence evaluations.

7.5 Visualizing the 2,2,2 Crossover ANOVA, Model Checking, and Diagnostics

In this section we present several valuable graphical tools for visualizing the data which are used in the three single degree-of-freedom ANOVA contrasts testing for carryover effects (confounded with sequence effects and treatment-by-period interaction), period effects, and treatment effects. We also present a series of graphics for assessing model fit. Some of the graphical techniques are reasonably common ones with special adaptations to the 2,2,2 crossover design; others are specific to the 2,2,2 crossover design.

7.5.1 Visualizing the ANOVA

We illustrate two sets of graphical presentations that visualize the data used in the three single degree-of-freedom ANOVA contrasts. The first is based on displaying treatment means by period; the second displays various linear combinations of the two data points collected for each subject.

Plots of Treatment Means by Period

To evaluate period effects, treatment effects, and to help diagnose carryover effects versus sequence effects versus treatment-by-period interaction, we plot the four treatment means, two for Fed and two for Fasted, by study period connecting the treatment means either by treatment sequence (Figure 7.9) or by treatment group (Figure 7.10). The pattern of the connected means illustrates three characteristics:

1. The relative ordering of the two treatments within each study period.
2. The magnitude of the difference between the two treatments within each period.
3. The magnitude of any difference between periods in each treatment group.

The range of the *y*-axis is determined by the range of the individual data points excluding any outliers which might distort the message in the graph.

We create the needed means with

```
> attach(food.df) # not needed if already attached
> tbpmeans.auc <- aggregate(logAUC,
    list(Treatment = trt, Period = per), mean)
> names(tbpmeans.auc)[3] <- "meanlogAUC"
> mean.fed.per1 <- tbpmeans.auc[1, "meanlogAUC"]
> mean.fasted.per1 <- tbpmeans.auc[2, "meanlogAUC"]
> mean.fed.per2 <- tbpmeans.auc[3, "meanlogAUC"]
> mean.fasted.per2 <- tbpmeans.auc[4, "meanlogAUC"]
```

We construct Figure 7.9 connecting the means by treatment sequence:

```
> plot(rep(1:2, 2), tbpmeans.auc$meanlogAUC,
    type = "n", axes = F, xlab="",
    ylab = "Geometric Mean AUC (ng x hr/ml)",
    xlim = c(0, 3))
> axislog(exp(tbpmeans.auc$meanlogAUC), line = 1, srt = 90,
    cex = 0.9)
> points(c(1, 2), c(mean.fed.per1, mean.fed.per2), pch = 4,
    cex = 1.2)
> points(c(1, 2), c(mean.fasted.per1, mean.fasted.per2),
    pch = 0, cex = 1.2)
> lines(c(1, 2),c(mean.fed.per1, mean.fasted.per2),
    pch=" ", type = "b", cex = 2)
> lines(c(1, 2), c(mean.fasted.per1, mean.fed.per2),
    pch = " ", type = "b", cex = 2, lty = 3)
> axis(1, at = 1:2, c("Per 1", "Per 2"), ticks = F)
> place.keytrt(0,1)
```

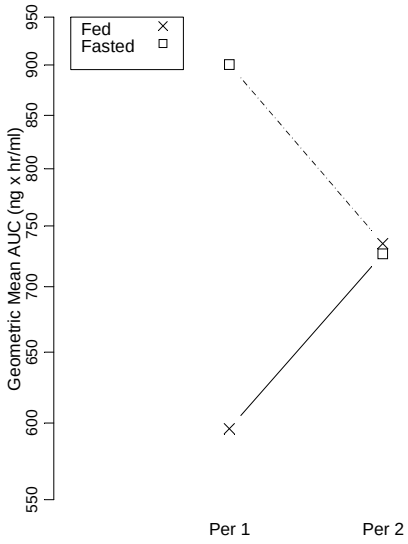


Figure 7.9. Treatment geometric means connected by sequence.

Figure 7.10, which connects the means by treatment, is created by the same sequence of steps except the two `lines` calls are slightly modified to read:

```
> lines(c(1, 2), c(mean.fed.per1, mean.fed.per2),
    pch = " ", type = "b", cex = 2)
> lines(c(1, 2), c(mean.fasted.per1, mean.fasted.per2),
    pch = " ", type = "b", cex = 2, lty = 3)
```

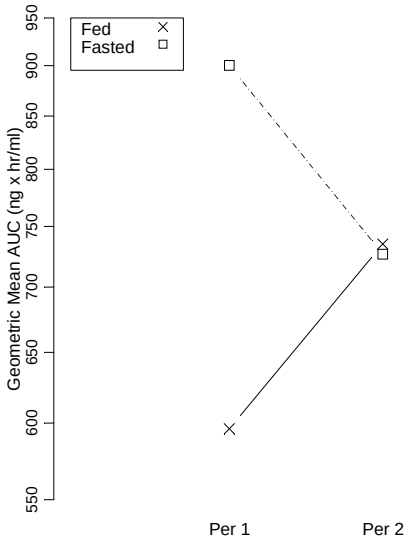


Figure 7.10. Treatment geometric means connected by treatment.

For our example, it is difficult to see much difference in the two display options since the Period 2 means are so close together. We see that in Period 1, the geometric mean AUC for the Fed state is substantially less than that for the Fasted state while in Period 2 they are quite similar. And, the geometric mean for the Fed state is less in Period 1 than in Period 2, and the reverse holds for the Fasted state.

Plotting Linear Combinations of the Data

Recall from Section 7.3 that each of the three single degree-of-freedom ANOVA contrasts can be evaluated using either *t*-tests or *F*-tests. More specifically, the contrasts are constructed by comparing between the two treatment sequences various linear combinations of the two observations for each subject. A series of three plots displays the data which are evaluated by each of the three single degree-of-freedom contrasts which test for carryover effects, period effects, or treatment effects. For example, to visualize the data in the contrast testing carryover effects (confounded with sequence effects and treatment-by-period interaction), plot the individual within subject sums (Period 1 log AUC + Period 2 log AUC) for each sequence (Figure 7.11). From the two sample distributions of data and the respective mean sums, the corresponding *t*-test can be evaluated visually. A difference in location between the two distributions suggests a carryover effect (or a sequence effect or a treatment-by-period interaction). Similarly, plot the individual and mean within-subject treatment differences (log Fed AUC–log Fasted AUC) by treatment sequence to visualize the data in the contrast for period effects (Figure 7.12). And, plot the individual and mean

within-subject period differences (Period 2 log AUC–Period 1 log AUC) by treatment sequence to visualize the data in the contrast for treatment effects (Figure 7.13). These plots also allow for a visual assessment of nonnormality, heteroscedasticity, and outliers, for each of the three two-sample *t*-tests.

To construct these plots, we create a new data frame.

```
> food.comb <- aggregate(logAUC,
    list(Subject = subj, Sequence = seqn), sum)
> names(food.comb)[3] <- "sumPer"
> food.comb <- cbind(food.comb, diffTrt =
    aggregate(log(AUC), list(Subject = subj,
      Sequence = seqn), function(x) {-diff(x)})[,3])
> food.comb$diffPer <- c(
    -food.comb$diffTrt[food.comb$Sequence == 1],
    food.comb$diffTrt[food.comb$Sequence==2])
```

We attach our new data frame `food.comb` and create Figures 7.11 through 7.13 with its variables. The code for these plots is in Appendix 7.A.4.

The shift in location in Figure 7.11 suggests either a carryover effect or a sequence effect or a treatment-by-period interaction. Earlier in the chapter, Figures 7.2, 7.7, 7.9, and 7.10 suggested a treatment-by-period interaction; Figures 7.1, 7.2, and 7.4 suggested a modest sequence effect.

The lack of a shift in location in Figure 7.12 indicates no evidence of a period effect, although the single lower point (Subject number 2) in the Fed then Fasted sequence influences the analysis quite a bit.

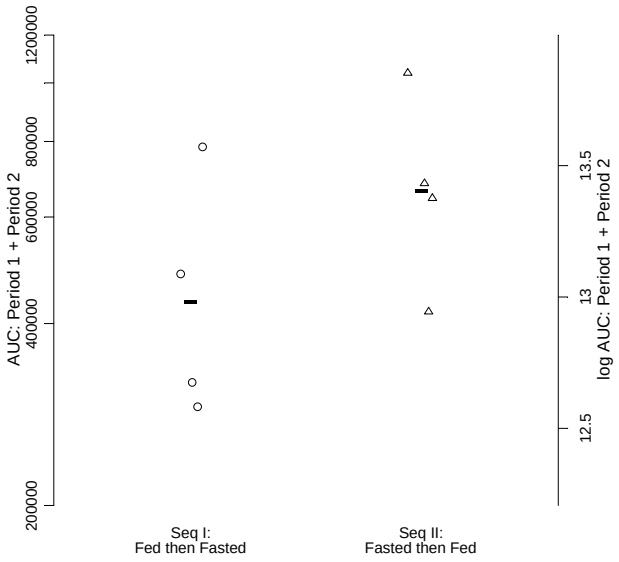


Figure 7.11. Evaluation of carryover effect.

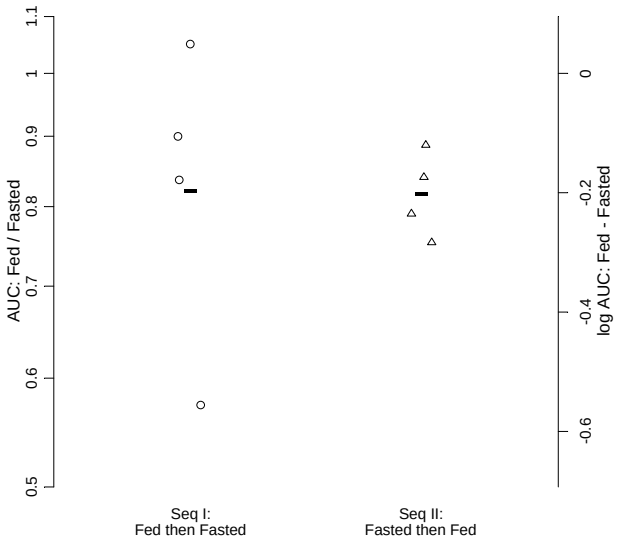


Figure 7.12. Evaluation of period effect.

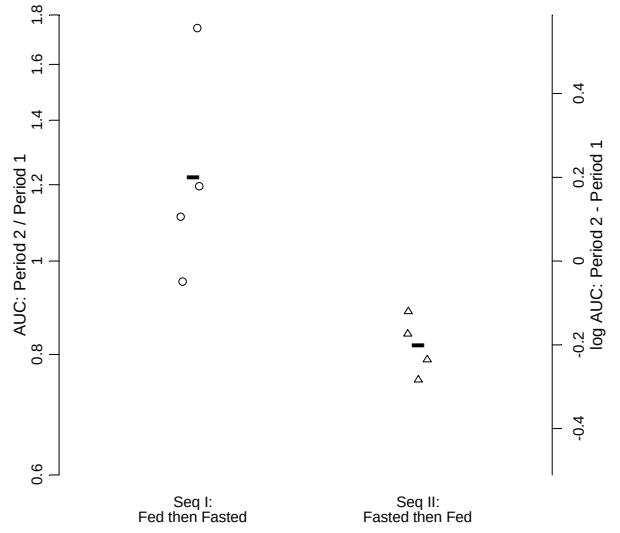


Figure 7.13. Evaluation of treatment effect.

The notable shift in location in Figure 7.13 indicates a treatment effect and again, the single upper point (Subject number 2) in the Fed then Fasted sequence makes this more pronounced. We end the session and clean up with the command

```
> detach("food.comb")
```

7.5.2 Model Checking and Diagnostics

Another series of plots assesses the appropriateness of the normal theory model for the 2,2,2 crossover design. These diagnostic plots are somewhat routine in concept, but there is an interesting twist in their construction for the 2,2,2 crossover design. The residual values from the normal theory analysis sum to zero within a subject. Therefore, the residuals from the two treatment periods for a given subject will have the same magnitudes and opposite signs. No additional information is gained from the second set of residuals. (This idea extends to *t* treatment, *p* period, *s* sequence crossover trials where $t = p$, $p > 2$. All relevant information is contained in the residuals from only $p-1$ periods.)

We demonstrate five plots. The first is a normal probability plot (Figure 7.14); the second is a plot of fitted versus observed values (Figure 7.15); the third is a plot of raw residuals versus fitted values (Figure 7.16); the fourth motivated by John W. Tukey plots the square root of the absolute value of the residuals against fitted values (Figure 7.17); and the fifth illustrates Cook's distance (Figure 7.18).

We begin by refitting the model in a slightly different way:

```
> fit2.food <- lm(logAUC ~ subjf + trt + perf,
    data = food.df)
```

We are interested in within-subject residuals, so calls to the fitted object will provide these and also corresponding identification and diagnostic quantities. Projections (see the `proj` function) could alternatively be used to obtain the residuals from the multistratum `fit.food` object in Section 7.4, but manual work would be required to construct the identification and diagnostic quantities. As mentioned above, the two residuals sum to zero, so we plot only the Period 1 residuals.

We need to make sure the MASS library is attached (Venables and Ripley, 1999) in order to use their standardized and Studentized residual functions `stdres` and `studres`. Next we create a data frame for our upcoming analysis that contains only observations from the first period:

```
> half.index <- food.df$per == 1
> half.resid <- residuals(fit2.food)[half.index]
> library(MASS) # Venables and Ripley library
> half.studres <- studres(fit2.food)[half.index]
> half.stdres <- stdres(fit2.food)[half.index]
> half.pred <- predict(fit2.food)[half.index]
```

```
> half.cooks <- Cooks.lm(fit2.food)[half.index]
> food.half <- data.frame(food.df[half.index, ],
    resid = half.resid, studres = half.studres,
    stdres = half.stdres, pred = half.pred,
    cooks = half.cooks)
> # synchronize row names and subject ID
> row.names(food.half) <- food.half$subj
```

A listing of this data frame is given in Appendix 7.A.4. The `Cooks.lm` function is taken from the `plot.lm` function definition in S-PLUS and is also listed in Appendix 7.A.4. With some modification, we also make use of another function defined in `plot.lm` called `id.n` to help us identify values of large magnitude. See Appendix 7.A.1 for its definition.

We do not need to be concerned with transforming back axes from the log scale to the original scale, so our call sequences are simplified. We first look at a normal probability plot:

```
> attach(food.half)
> qqxy <- qqnorm(studres, plot = F)
> qqxy.x <- qqxy$x; qqxy.y <- qqxy$y
> names(qqxy.x) <- names(qqxy.y) <- row.names(food.half)
> plot(qqxy.x, qqxy.y, type = "n",
    xlab = "Standard Normal Quantiles",
    ylab = "Studentized Residuals, Period 1")
> points(qqxy.x[seqn == 1], qqxy.y[seqn == 1], pch = 1)
> points(qqxy.x[seqn == 2], qqxy.y[seqn == 2], pch = 2)
> qqline(studres)
> id.n(qqxy.x, qqxy.y, how.many = 2, offset = 0.2)
> place.keyseq(0, 1)
```

Figure 7.14 plots the eight Studentized residuals from the first period against expected normal order statistics. We observe two notable deviations from linearity: Subjects number 2 and 5, both of whom were randomized to the first treatment sequence, Fed then Fasted. The fact that these two residuals deviate from the other six is not surprising given our initial graphical exploration of this dataset.

Figures 7.15 through 7.17 are generated by similar codes and the full displays of the commands are in Appendix 7.A.4. Figure 7.15 plots the fitted log AUC values from Period 1 against the observed log AUC values for Period 1. The fit is not unreasonable and only one point (Subject number 2) deviates notably from the 45° identity line. Note the location of the open triangles (upper right) for Sequence 2 versus the open circles (lower left) for Sequence 1.

Figure 7.16 plots the raw residuals from Period 1 against the fitted values from Period 1. Subjects number 2 and 5 demonstrate residuals which are the most different from the others.

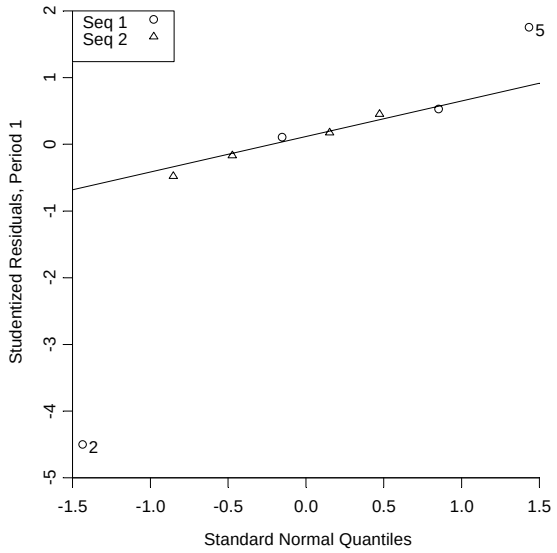


Figure 7.14. Normal probability plot.

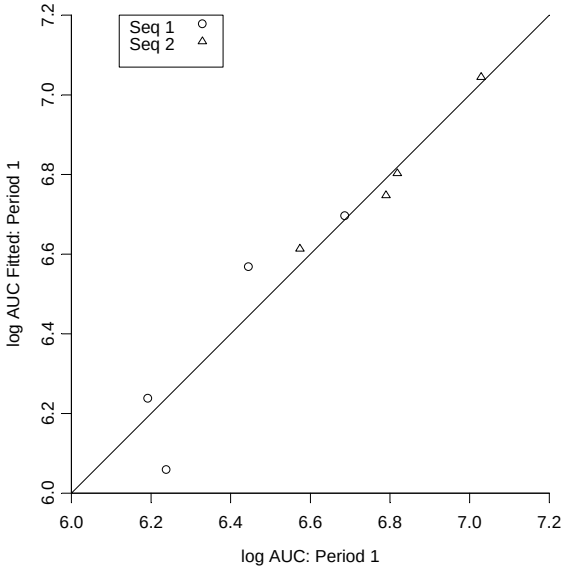


Figure 7.15. Evaluation of model fit.

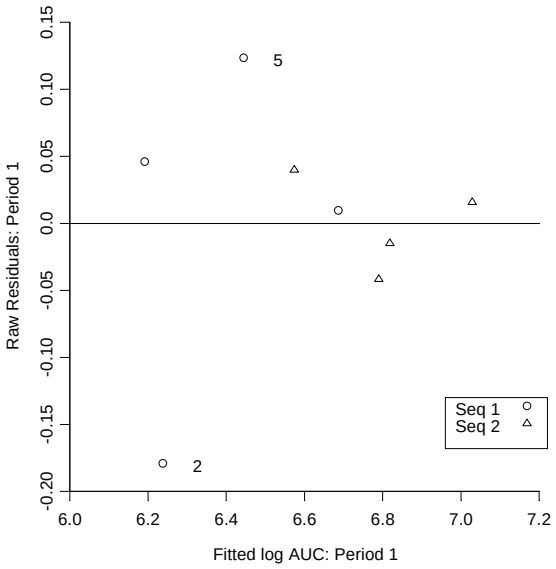


Figure 7.16. Raw residuals versus fitted values.

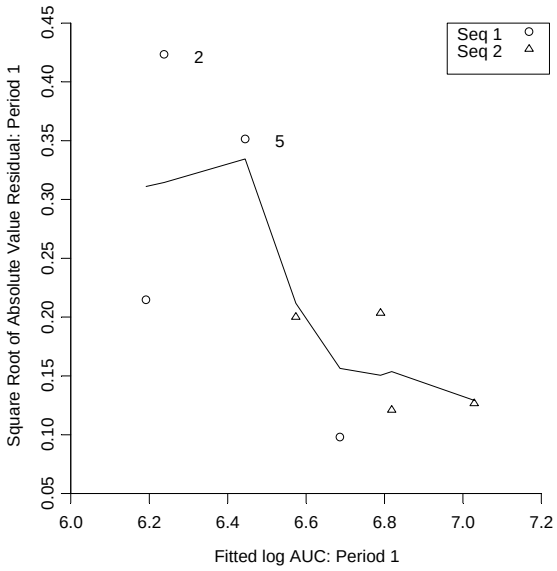


Figure 7.17. Equal variance plot.

Figure 7.17 evaluates the equal variance assumption. The plot method was first suggested by John Tukey; a reference example is shown in Cleveland (1993). A lowess (Cleveland, 1979) robust smoothing curve is superposed as a guide to help discover any trends. Ideally we would like to see an approximately flat line. The apparent volatility of these data is not unexpected given the small number of data points.

And finally, Cook's distance as plotted in Figure 7.18 identifies the data from Subjects number 2 and 5 as the most influential:

```
> plot(1:8, cooks, type = "h", axes = F,
    ylab = "Cook\'s Distance: Period 1",
    xlab="Subject",xlim=c(0,9))
> axis(2)
> mtext(side = 1, at = 1:8, paste(subj))
> detach("food.half") # cleanup
```

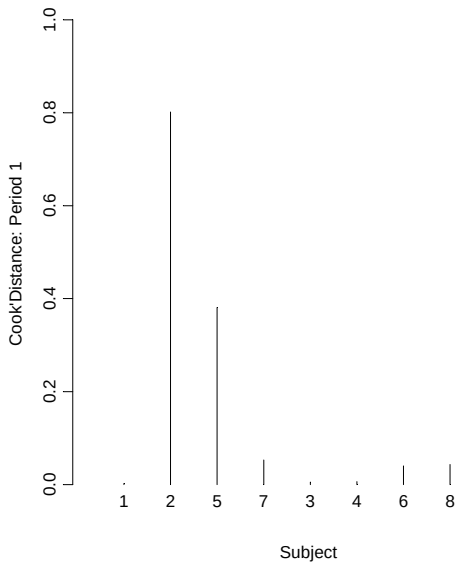


Figure 7.18. Cook's distance plot.

7.6 Concluding Remarks

In this chapter, we provided S-PLUS tools for graphical insight and the normal theory statistical analysis of data from a 2,2,2 crossover design. We demonstrated the value of graphical analyses from initial data exploration and discovery, into understanding the ANOVA with particular emphasis on the single degree of freedom contrast which confounds carryover effects with sequence ef-

fects and treatment-by-period interaction, through to model checking and diagnostic work following the ANOVA. We illustrated the use of S-PLUS on a dataset from a food interaction study where there was some evidence of a food effect in the presence of either a modest sequence effect or a treatment-by-period interaction, and where two subjects had notably more influence on the results of the trial than the other six enrolled. And, we provided a useful summary graph for presenting the individual, average, and inferential results from a 2,2,2 crossover average bioequivalence or interaction trial.

7.7 References

Agresti (1990). *Analysis of Categorical Data*. Wiley, New York.

Bradstreet, T.E. (1992). Favorite data sets from early phases of drug research–Part 2. *Proceedings of the Section on Statistical Education.* American Statistical Association, pp. 219–223.

Chambers, J.M., Cleveland, W.S., Kleiner, B., and Tukey, P.A. (1983). *Graphical Methods for Data Analysis*. Wadsworth, Pacific Grove, CA.

Chow, S.–C. and Liu, J.P. (2000). *Design and Analysis of Bioavailability Studies.* Second Edition. Marcel Dekker, New York.

Cleveland, W.S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, **74**, 829–836.

Cleveland, W.S. (1993). *Visualizing Data*. Hobart Press, Summit, NJ.

Jones, B. and Kenward, M.G. (1989). *Design and Analysis of Crossover Trials.* Chapman & Hall, London.

Senn, S. (1993). *Cross–over Trials in Clinical Research.* Wiley, New York.

Tukey, J.W. (1977). *Exploratory Data Analysis*. Addison–Wesley, Reading, MA.

Venables, W.N. and Ripley, B.D. (1999). *Modern Data Analysis*. Third Edition. Springer–Verlag, New York.

7.A. Appendix

7.A.1 Functions for Graphical Annotations

Utility Function for Generating Log Axes Ticks

```
axislog <- function(data, nint = 5, side = 2, opp.side = F,
  ...) {
  orig <- pretty(data, nint)
  orig.px <- orig[orig > 0]
  logrange <- log(orig.px)
  usrnow <- par("usr")  # graphical layout settings
  if(side == 2) {
    par(usr = c(usrnow[1:2], min(logrange), max(logrange)))
    axis(2, at = logrange, labels = paste(orig.px), ...)
  }
  else if(side == 1) {
    par(usr = c(min(logrange), max(logrange), usrnow[3:4]))
    axis(1, at = logrange, labels = paste(orig.px), ...)
  }
  if(opp.side == T) {
    axis(side + 2, at = pretty(logrange),
      labels = paste(pretty(logrange)), ...)
  }
  invisible()
}
```

Utility Function for Placing Sequence Identifier Key on Graphs

```
# x = 0 and y = 1 refers to upper left corner,
# x = 1 and y = 0 refers to lower right corner

place.keyseq <- function(x = 0, y = 1, ...)
{
  oldpar <- par(gr.state)
  par(usr=c(0, 1, 0, 1))
  on.exit(par(oldpar))
  on.exit(par(new = F), add = T)
  key(x, y, text = list(c("Seq 1", "Seq 2", " ")),
    points = list(pch = c(1, 2, 32)), border = T)
# Adds space between text and bottom border
  invisible()
}
```

Similar Function for Treatment Identifier Key

```
place.keytrt <- function(x = 0, y = 1, ...) {
  oldpar <- par(gr.state)
  par(usr = c(0, 1, 0, 1))
  on.exit(par(oldpar))
  on.exit(par(new = F), add = T)
  key(x, y, text = list(c("Fed", "Fasted", " ")),
    points = list(pch = c(4, 0, 32)), border = T)
  invisible()
}
```

Utility Function to Help Identify Values of Large Magnitude on Figures 7.14 – 7.16, Section 7.5.2

```
id.n <- function(x, y, how.many = F, offset = 0)
  if(how.many) {
# Identify "how.many" greatest y-values (in absolute value)
# based on sub-function definition in
# plot.lm function method
    n <- length(y)
    oy <- order(abs(y))
    names(oy) <- names(abs(y))[oy]
    which <- oy[(n - how.many + 1):n]
    text(x[which] + offset, y[which], names(which),
      adj = 1)
  }
```

7.A.2 Additional Code for Section 7.3

Prior to running the code in this section, the data frame `food.df` should be attached to the search list and graphics parameters should be changed from their default values using the following commands:

```
attach(food.df)
par(xaxs = "e", yaxs = "e", bty = "l", pty = "s")
```

Figure 7.5

```
plot(x = logAUC[trt == "Fed"] + logAUC[trt == "Fasted"],
  y = logAUC[trt == "Fed"] - logAUC[trt == "Fasted"],
  type = "n", axes = F,
  xlab = "exp(log Fed + log Fasted) AUC",
  ylab = "exp(log Fed - log Fasted) AUC")
```

```
axislog(exp(logAUC[trt == "Fed"] –
    logAUC[trt == "Fasted"]), line = 1, srt = 90,
  cex = 0.9)
axislog(exp(logAUC[trt == "Fed"] +
    logAUC[trt == "Fasted"]), side = 1, line = 1,
  cex = 0.8)
points(x = logAUC[trt == "Fed" & seqn == 1] +
    logAUC[trt == "Fasted" & seqn == 1],
  y = logAUC[trt == "Fed" & seqn == 1] –
    logAUC[trt == "Fasted" & seqn == 1], pch = 1,
  cex = 1.2)
points(x = logAUC[trt == "Fed" & seqn == 2] +
    logAUC[trt == "Fasted" & seqn == 2],
  y = logAUC[trt == "Fed" & seqn == 2] –
    logAUC[trt == "Fasted" & seqn == 2], pch = 2,
  cex = 1.2)
abline(0, 0, lty = 3)
mtext(side = 1, at = log(c(150000, 200000, 400000, 600000,
  800000, 1000000, 1200000)),
  text = c("Geometric Mean:",
    paste(round(sqrt(c(200000, 400000, 600000, 800000,
      1000000, 1200000)), 0))), line = 4, cex = 0.8)
place.keyseq(0.8, 0.2)
# a little off lower right-hand corner due to
# x-axis placement
```

Figure 7.6

```
plot(x = rep(1, 8),
  y = log(AUC[trt == "Fed"]/AUC[trt == "Fasted"]),
  type = "n", axes = F, xlab = "",
  ylab = "Ratio: Fed / Fasted AUC")
axislog(AUC[trt == "Fed"]/AUC[trt == "Fasted"], line = 1,
  srt = 90, cex = 0.9)
points(x = jitter(rep(1, 4)),
  y = log(AUC[trt == "Fed" & seqn == 1]/
    AUC[trt == "Fasted" & seqn == 1]), pch = 1)
points(x = jitter(rep(1, 4)),
  y = log(AUC[trt == "Fed" & seqn == 2]/
    AUC[trt == "Fasted" & seqn == 2]), pch = 2)
abline(h = log(1), lty = 3)
axis(2, at = log((seq(0.5, 1.1, by = 0.1))),
  labels = paste(seq(-50, 10, by = 10), "%", sep = ""),
  line = -2, srt = 0, cex = 0.8, ticks = F)
place.keyseq(0.8, 0.2)
```

Figure 7.7

```
plot(per, logAUC, type = "n", axes = F, xlab = "",
  ylab = "AUC (ng x hr/ml)", xlim = c(0, 3))
axislog(AUC, line = 1, srt = 90, cex = 0.9)
points(x = jitter(per[trt == "Fasted"], factor = 3),
  y = logAUC[trt == "Fasted"], pch = 0)
points(x = jitter(per[trt == "Fed"], factor = 3),
  y = logAUC[trt == "Fed"], pch = 4)
lines(c(1.2, 1.8),
  c(mean(logAUC[trt == "Fasted" & per == 1]),
    mean(logAUC[trt == "Fasted" & per == 2])),
  type = "b", pch = 0, cex = 2)
lines(c(1.2, 1.8), c(mean(logAUC[trt == "Fed" & per == 1]),
    mean(logAUC[trt == "Fed" & per == 2])), type = "b",
  pch = 4, cex = 2)
axis(1, at = 1:2, c("Per 1", "Per 2"), ticks = F)
place.keytrt(0.8, 0.2)
```

7.A.3 Additional Code for Section 7.4

Percent Change Expressions

Percent change expressions of Fed relative to Fasted; these build on the ratios computed.

```
orig.pctchg <- (orig.ratio - 1)*100
orig.pctchg.se <- 100*orig.ratio.se
lcl.pctchg <- (lcl.ratio - 1) * 100
ucl.pctchg <- (ucl.ratio - 1) * 100
```

Figure 7.8

```
attach(food.df) # not needed if already attached
plot(x = rep(1, 8),
  y = log(AUC[trt == "Fed"]/AUC[trt == "Fasted"]),
  type = "n", axes = F, xlab = "", xlim = c(0.5, 2.5),
  ylim = log(c(0.5, 1.25)),
  ylab = "Ratio: Fed / Fasted AUC")
axislog(c(0.5, 1.25), line = 1, srt = 90, cex = 0.9)
abline(h = log(c(0.8, 1, 1.25)), lty = 7)
points(x = jitter(rep(1, 4)),
  y = log(AUC[trt == "Fed" & seqn == 1]/
    AUC[trt == "Fasted" & seqn == 1]), pch = 1)
```

```
points(x = jitter(rep(1, 4)),
  y = log(AUC[trt == "Fed" & seqn == 2]/
    AUC[trt == "Fasted" & seqn == 2]), pch = 2)
points(1.4, meandiff, cex = 1.25, pch = 16)
par(lwd = 5)
error.bar(1.4, meandiff, lower = t.crit * meandiff.se,
  add = T, gap = F)
par(lwd = 1)
text(1.5, meandiff - t.crit * meandiff.se,
  paste("(", round(lcl.ratio, 2), ")", sep = ""), adj = 0)
text(1.5, meandiff, paste("(", round(orig.ratio, 2), ")",
    sep = ""), adj = 0)
text(1.5, meandiff + t.crit * meandiff.se,
  paste("(", round(ucl.ratio, 2), ")", sep = ""), adj = 0)

# For Period 1 only
per1data <- t.test(log(AUC[trt == "Fed" & per == 1]),
  log(AUC[trt == "Fasted" & per == 1]), conf.level = 0.9)
meandiff.per1 <-  - diff(per1data$estimate)
lcl.per1 <- per1data$conf.int[1]
ucl.per1 <- per1data$conf.int[2]
points(1.9, meandiff.per1, cex = 1.25, pch = 16)
par(lwd = 5)
error.bar(1.9, meandiff.per1, lower = lcl.per1,
  upper = ucl.per1, incr = F, add = T, gap = F)
par(lwd = 1)
text(2, lcl.per1, paste("(", round(exp(lcl.per1), 2), ")",
    sep = ""), adj = 0)
text(2, meandiff.per1, paste("(", round(exp(meandiff.per1),
    2), ")", sep = ""), adj = 0)
text(2, ucl.per1, paste("(", round(exp(ucl.per1), 2), ")",
    sep = ""), adj = 0)

# For Period 2 only
per2data <- t.test(log(AUC[trt == "Fed" & per == 2]),
  log(AUC[trt == "Fasted" & per == 2]), conf.level = 0.9)
meandiff.per2 <-  - diff(per2data$estimate)
lcl.per2 <- per2data$conf.int[1]
ucl.per2 <- per2data$conf.int[2]
points(2.4, meandiff.per2, cex = 1.25, pch = 16)
par(lwd = 5)
error.bar(2.4, meandiff.per2, lower = lcl.per2,
  upper = ucl.per2, incr = F, add = T, gap = F)
par(lwd = 1)
```

```
text(2.5, lcl.per2, paste("(",
  round(exp(lcl.per2), 2), ")", sep = ""), adj = 0)
text(2.5, meandiff.per2, paste("(",
  round(exp(meandiff.per2), 2), ")", sep = ""), adj = 0)
text(2.5, ucl.per2, paste("(",
  round(exp(ucl.per2), 2), ")", sep = ""), adj = 0)
par(lwd = 1)
axis(1, at = c(1.4, 1.9, 2.4), c("All\nData",
    "Per 1\nData only", "Per 2\nData only"), ticks = F)
place.keyseq(0, 0.2)
detach("food.df") # cleanup
```

7.A.4 Additional Code for Section 7.5

Prior to running the code to create Figures 7.11–7.13, the data frame `food.comb` should be attached to the search list. Prior to running the code to create Figures 7.14–7.18, the data frame `food.half` should be attached to the search list.

```
attach(food.comb)
```

Figure 7.11: Carryover Effect

```
plot(as.numeric(Sequence), sumPer,
  type = "n", axes = F, xlab = "",
  ylab = "AUC: Period 1 + Period 2", xlim = c(0.5, 2.5))
axislog(exp(sumPer),opp.side = T, line = 1, srt = 90,
  cex = 0.9)
points(jitter(rep(1, 4), factor = 3),
  sumPer[Sequence == 1], pch = 1)
points(jitter(rep(2, 4), factor = 3),
  sumPer[Sequence == 2], pch = 2)
points(x = c(1, 2), c(mean(sumPer[Sequence == 1]),
    mean(sumPer[Sequence == 2])), pch = "-", cex = 3)
axis(1, at = c(1, 2), labels = c("Seq I:\nFed then Fasted",
    "Seq II:\nFasted then Fed"), ticks = F, cex = 0.9)
mtext("log AUC: Period 1 + Period 2", side = 4, line = 3)
```

Figure 7.12: Period Effect

```
plot(as.numeric(Sequence), diffTrt, type = "n", axes = F,
  xlab = "", ylab = "AUC: Fed / Fasted",
  xlim = c(0.5, 2.5))
axislog(exp(diffTrt), opp.side = T, line = 1, srt = 90,
  cex = 0.9)
points(jitter(rep(1, 4), factor = 3),
  diffTrt[Sequence == 1], pch = 1)
points(jitter(rep(2, 4), factor = 3),
  diffTrt[Sequence == 2], pch = 2)
points(x = c(1, 2), c(mean(diffTrt[Sequence == 1]),
    mean(diffTrt[Sequence == 2])), pch = "-", cex = 3)
axis(1, at = c(1, 2), labels = c("Seq I:\nFed then Fasted",
    "Seq II:\nFasted then Fed"), ticks = F, cex = 0.9)
mtext("log AUC: Fed - Fasted", side = 4, line = 3)
```

Figure 7.13: Treatment Effect

```
plot(as.numeric(Sequence), diffPer, type = "n", axes = F,
  xlab = "", ylab = "AUC: Period 2 / Period 1",
  xlim = c(0.5, 2.5))
axislog(exp(diffPer), opp.side = T, line = 1, srt = 90,
  cex = 0.9)
points(jitter(rep(1, 4), factor = 3),
  diffPer[Sequence == 1], pch = 1)
points(jitter(rep(2, 4), factor = 3),
  diffPer[Sequence == 2], pch = 2)
points(x = c(1, 2), c(mean(diffPer[Sequence == 1]),
    mean(diffPer[Sequence == 2])), pch = "-", cex = 3)
axis(1, at = c(1, 2), labels = c("Seq I:\nFed then Fasted",
    "Seq II:\nFasted then Fed"), ticks = F, cex = 0.9)
mtext("log AUC: Period 2 - Period 1", side = 4, line = 3)
```

Listing of Food Half Data Used to Study Period 1 Residuals in Figures 7.14–7.18

```
> food.half
  subj seq    trt per     AUC subjf perf seqf seqn trtn   logAUC
1    1 +/-    Fed   1  809.44     1    1  +/-    1    1 6.696343
2    2 +/-    Fed   1  428.00     2    1  +/-    1    1 6.059123
5    5 +/-    Fed   1  712.24     5    1  +/-    1    1 6.568415
7    7 +/-    Fed   1  511.84     7    1  +/-    1    1 6.238012
3    3 -/+ Fasted   1  901.11     3    1  -/+    2    2 6.803627
4    4 -/+ Fasted   1 1146.96     4    1  -/+    2    2 7.044870
6    6 -/+ Fasted   1  745.51     6    1  -/+    2    2 6.614069
8    8 -/+ Fasted   1  852.86     8    1  -/+    2    2 6.748595

          resid    studres     stdres     pred       cooks
1  0.009612069  0.1075670  0.1176976 6.686731 0.002308787
2 -0.179138916 -4.4991205 -2.1935151 6.238262 0.801918105
5  0.123479646  1.7543451  1.5119801 6.444935 0.381013961
7  0.046047200  0.5289139  0.5638375 6.191965 0.052985445
3 -0.014709018 -0.1648621 -0.1801086 6.818336 0.005406516
4  0.016085298  0.1803839  0.1969608 7.028785 0.006465593
6  0.040115695  0.4577066  0.4912075 6.573953 0.040214138
8 -0.041491975 -0.4741032 -0.5080598 6.790087 0.043020787
```

S-PLUS Function to Compute Cook's Distance

This function is defined within the S-PLUS function `plot.lm`.

```
Cooks.lm <- function(fit)
{
  lmi <- lm.influence(fit)
  fit.s <- summary.lm(fit)
  s <- fit.s$sigma
  h <- lmi$hat
  p <- fit$rank
  stdres <- fit$residuals/(s * (1 - h)^0.5)  #standardized
  cooks <- (1/p * stdres^2 * h)/(1 - h)
  if(!is.null(fit$na.action))
    cooks <- nafitted(fit$na.action, cooks)
  return(cooks)
}
```

Figure 7.15: Response versus Fitted

```
plot(pred, logAUC, type = "n", xlab = "log AUC: Period 1",
   ylab = "log AUC Fitted: Period 1")
abline(0, 1)
points(pred[seqn == 1], logAUC[seqn == 1], pch = 1)
points(pred[seqn == 2], logAUC[seqn == 2], pch = 2)
place.keyseq(0.1, 1)
```

Figure 7.16: Raw Residuals versus Fitted

```
plot(pred, resid, type = "n",
  xlab = "Fitted log AUC: Period 1",
  ylab = "Raw Residuals: Period 1")
points(pred[seqn == 1], resid[seqn == 1], pch = 1)
points(pred[seqn == 2], resid[seqn == 2], pch = 2)
abline(0, 0)
id.n(pred, resid, how.many = 2,offset = 0.1)
place.keyseq(0.8, 0.2)
```

Figure 7.17: Equal Variance

```
plot(pred, sqrt(abs(resid)), type = "n",
  xlab = "Fitted log AUC: Period 1",
 ylab = "Square Root of Absolute Value Residual: Period 1")
points(pred[seqn == 1], sqrt(abs(resid))[seqn == 1],
  pch = 1)
points(pred[seqn == 2], sqrt(abs(resid))[seqn == 2],
  pch = 2)
lines(lowess(pred,sqrt(abs(resid))))
id.n(pred, sqrt(abs(resid)), how.many = 2, offset =0.1)
place.keyseq(0.8, 1.0)
```



http://www.springer.com/978-0-387-98814-6

Applied Statistics in the Pharmaceutical Industry
With Case Studies Using S-Plus
Millard, S.P.; Krause, A. (Eds.)
2001, XVIII, 514 p., Hardcover
ISBN: 978-0-387-98814-6