December 2, 2012
Draft Version

R Companion to Montgomery's *Design and Analysis of Experiments* (2005)

Décembre 2006

December 2, 2012
Draft Version

Introduction

This document has been conceived as a supplemental reference material to accompany the excellent book of Douglas C. Montgomery, *Design and Analysis of Experiments* (hereafter abbreviated as DAE). Now published in its 6th edition, this book covers numerous techniques used in the design and analysis of experiments. This includes: Classical comparative experiments (two groups of observations, independant or not), the natural extension to the case of k means to be compared (one-way ANOVA), various ways of blocking (randomized blocks, latin squares and derived), the factorial– in particular the 2^k ones –and fractional designs, the fitting of regression models and response surface methodology, a review of techniques for robust parameter estimation, and the various derivation of standard design with fixed effects (random factor, nested and split-plot designs).

Motivation for writting such a computer oriented document was initially started when I was reading the document elaborated by Laura Thompson to accompany Agresti's famous book, *Categorical Data Analysis*[1]. Indeed, I found that this really was a great idea as it brings to the other side of the statistian's activity, that of computing. This document is somewhat different of `splusdiscrete` since I don't pretend to be as exhaustive as she is in her own manuscript.

While this textbook contains the same material as the original book written by Montgomery, it is obviously not meant to be a partial electronic copy, nor to be a complete replacement of the original book. Rather, I put some emphasis on modern computer methods used to analyse data. Briefly, each chapter of this textbook is organized as follow: first, I give a short summary of the main concepts presented by Montgomery; then I try to illustrate some of the most important (to my opinion!) ones with R. Exemples used by Montgomery are completely re-analysed using R. However, I do not answer to the proposed exercices that can be found at the end of each chapter of DAE. I left them to the interested reader, giving occasionnally some advice on "R way" to do the intended analysis.

About R

Montgomery mainly uses non-free software to analyse the dataset presented in each chapter. Though these dedicated softwares have proved to be very good packages for statistical analysis, their cost restrict their use to people working in laboratory where specific credits are devoted to such investment. Now, it seems that the avalailability of open-source software, like R, offers an elegant alternative to such solutions (often inaccessible to students).

R has been developed based on the S programming language and S-PLUS software, although it is not a free completely rewritten clone of S-PLUS. In fact, there are several

[1] A revised version of her textbook can be found here: https://home.comcast.net/ lthompson221/Splusdiscrete2.pdf.

December 2, 2012
Draft Version

differences between the two, and the interested reader can refer to the following adress for a deeper understanding of the way R has been built: www.

R can be freely downloaded on CRAN website (www.cran.r-project.org), and many documentation and tutorials can be found at the same address. What makes R a better choice to closed software like the ones Montgomery uses in his book is that the source code of all the statistical built-in routines is available and can be studied separately. In addition, users can add their own function to suit their needs for a particular analysis, or for batching several analysis process at once.

Exemple Scripts

All the analyses done by Montgomery in his book are replicated here using R, version 2.7, on Mac OS X though they were initiated with R 2.4 running on a Linux plateform. The source code for all the exemples is available at the following address:

www.aliquote.org/articles/stat/dae/

Datasets used in this textbook can also be found on this webpage. R scripts should run without any problem with any version of R ≥ 2.0. However, in case you encounter any problem, please send me an email ( ) with some detailed information on the bug found. I don't use `Sweave` to process this document, because at the time of the first writing of this textbook I felt more comfortable without it; further, as there aren't any simulated data, nor too strong packages dependency, a simple verbatim environment should be sufficient for most of what I need. So all the included code is static, except for some pieces of code in Chapter 2, and compilation relies on `dvips` + `ps2pdf` only. Furthermore, I haven't splitted the `tex` source into distinct chapters, so there is a "huge" source file that can be downloaded from there if anyone is interested in getting the main `tex` file : www.aliquote.org/articles/stat/dae/dae.tex.

December 2, 2012
Draft Version

Contents

December 2, 2012
Draft Version

December 2, 2012
Draft Version

Chapter 1

Introduction

The 6th edition of Montgomery's book, *Design and Analysis of Experiments*, has many more to do with the various kind of experimental setups commonly used in biomedical research or industrial engineering, and how to reach significant conclusions from the observed results. This is an art and it is called the Design of Experiment (DOE). The approach taken along the textbook differs from most of the related books in that it provides both a deep understanding of the underlying statistical theory and covers a broad range of experimental setups, e.g. balanced incomplete block design, split-plot design, or response surface. As all these DOE are rarely presented altogether in an unified statistical framework, this textbook provides valuable information about their common anchoring in the basic ANOVA Model.

Quoting Wiley's website comments,

> Douglas Montgomery arms readers with the most effective approach for learning how to design, conduct, and analyze experiments that optimize performance in products and processes. He shows how to use statistically designed experiments to obtain information for characterization and optimization of systems, improve manufacturing processes, and design and develop new processes and products. You will also learn how to evaluate material alternatives in product design, improve the field performance, reliability, and manufacturing aspects of products, and conduct experiments effectively and efficiently.

Modern computer statistical software now offer an increasingly "power" and allow to run computationally intensive procedures (bootstrap, jacknife, permuation tests,...) without leaving the computer desktop for one night or more. Furthermore, multivariate exploratory statistics have brought novel and exciting graphical displays to highlight the relations between several variables at once. As they are part of results reporting, they complement very kindly the statistical models tested against the observed data.

We propose to analyze some the data provided in this textbook with the open-source R statistical software. The official website, www.r-project.org, contains additional information and several handbook wrote by international contributors. To my opinion, R has benefited from the earlier development of the S language as a statistical program-

December 2, 2012
Draft Version

ming language, and as such offers a very flexible way to handle every kind of dataset. Its grahical capabilities, as well as its inferential engine, design it as the more flexible statistical framework at this time.

The R packages used throughout these chapters are listed below, in alphabetical order. A brief description is provided, but refer to the on-line help (`help(package="xx")`) for further indications on how to use certain functions.

Package listing. Since 2007, some packages are now organized in what are called *Task Views* on CRAN website. Good news: There is a Task View called `ExperimentalDesign`. By the time I started to write this textbook, there were really few available ressources to create complex designs like fractional factorial or latin hypercube designs, nor was there any in-depth coverage of DOE analysis with R, except [?] who dedicated some attention to blocking and factorial designs, J. Faraway's handbook, *Practical Regression and Anova using R* [?] (but see CRAN contributed documentation[1]), and G. Vikneswaran who wrote An R companion to "Experimental Design" which accompagnies Berger and Maurer's book [?].

car provides a set of useful functions for ANOVA designs and Regression Models;

lattice provides some graphical enhancements compared to traditional R graphics, as well as multivariate displays capabilities;
For *Trellis Displays*, see `http://stat.bell-labs.com/project/trellis/`

lme4 the newer and enhanced version of the `nlme` package, for which additional data structure are available (nested or hierarchical model,...);

nlme for handling mixed-effects models, developped by Pinheiro & Bates [?];

npmc implements two procedures for non-parametric multiple comparisons procedures;

Further Readings. Additional references are given in each chapter, when necessary. However, there are plenty of other general textbooks on DOE, e.g. [?, ?, ?] (English) and [?, ?, ?, ?] (French), among the most recent ones.

[1] Faraway has now published two books on the analysis of (Non-)Linear Models, GLM, and Mixed-effects Models, see [?, ?].

December 2, 2012
Draft Version

Chapter 2

Simple Comparative Experiments

2.1 Summary of Chapter 2

After having defined the way simple comparative experiments are planned (treatments or conditions, levels of a factor), Montgomery briefly explains basic statistical concepts related to the analysis of such designs. This includes the ideas of sampling distributions, or hypothesis formulation. Two samples related problems are covered, both under specific distributional assumption or in an alternative non-parametric way. The t-test is probably the core concept that one has to understand before starting with more complex models. Indeed, the construction of a test statistic, the distribution assumption of this statistic under the null hypothesis (always stated as an absence of difference between treatment means), and the way one can conclude from the results are of primary importance. This chapter must be read by every scientist looking at a first primer in statistical inference.

2.2 Sampling distributions

Several probabillity distributions are avalaible in R. They are all prefixed with one of the following letters: d, p, q, and r, which respectively refers to: density function, probability value, quantile value and random number generated from the distribution. For example, a sample of ten normal deviates, randomly choosen from the standard normal distribution (also refered to as $\mathcal{N}(0;1)$, or Z distribution), can be obtained using

```
x <- rnorm(10)
```

Since each call to the random number generator (RNG) of R involves a different random seed[1], it could be convenient to fix its value such that the same results can be obtained later. To do so, use something like:

[1] Random number generators were originally based on a congruential recurrence relation, e.g. $x_{k+1} = a_0 + b \cdot x_k \pmod{c}$, where a_0 is the initial (fixed) seed for a given sequence. Now, several sophisticated algorithms are available; refer to ?RNGkind.

December 2, 2012
Draft Version

```
set.seed(891)
```

The function `set.seed` is used to set the RNG to a specified state, and it takes any integer between 1 and 1023. Random values generation is part of statistical theory and these techniques are widely used in simulation sstudies. Moreover, random numbers are the core of several computational intensive algorithms, like the Bootstrap estimation procedure or Monte Carlo simulation design. A very elegant introduction to this topic is provided in [?] (see Chapter 8 for some hints on the use of R RNG).

R can be used to produce different kind of graphical representation. It's probably the most challenging statistical tool for that particular option. Among them, dot diagram and histogram are useful tools to visualize continuous variable. Figure 2.1 has been created with the following commands:

```
# Tension Bond Strength data (Tab. 2-1, p. 24)
y1 <- c(16.85,16.40,17.21,16.35,16.52,17.04,16.96,17.15,16.59,16.57)
y2 <- c(16.62,16.75,17.37,17.12,16.98,16.87,17.34,17.02,17.08,17.27)
y <- data.frame(Modified=y1,Unmodified=y2)
y.means <- as.numeric(apply(y,2,mean))
opar <- par(mfrow=c(2,1),mar=c(5,7,4,2),las=1)
stripchart(y,xlab=expression("Strength (kgf/cm^2)"),pch=19)
arrows(y.means,rep(1.5,2),y.means,c(1.1,1.9),length=.1)
text(y.means,c(1.2,1.8),round(y.means,2),pos=4,cex=.8)
# Random deviates (instead of data from metal recovery used in the book)
rd <- rnorm(200,mean=70,sd=5)
hist(rd,xlab="quantile",ylab="Relative frequency",
     main="Random Normal Deviates\n N(70,5)")
par(opar)
```

As can be seen with this snippet of code, relatively few commands allow to produce powerful graphics... Indeed, several books or website are dedicated to exploratory multivariate graphics. Among others, the interested reader may have a look at:

- S. Deepayan (2008). *Lattice. Multivariate Data Visualization with R*[2]. Springer. http://www.springer.com/statistics/computational/book/978-0-387-75968-5
- R Graph Gallery, http://addictedtor.free.fr/graphiques/
- Trellis Display, http://stat.bell-labs.com/project/trellis/
- P. Murrel (2005). *R Graphics*[3]. Chapman & Hall/CRC.

and, of course, the must-have-one book that Venables & Ripley wrote on the S language, now in its fourth edition, [?].

Histograms are naturally more appropriate when there are numerous observations (e.g. $n > 20$). It is not uncommon to express the data as a density plotted against the observed value, and to superimpose a normal density function with the corresponding

[2] R code and Figures can be found on http://dsarkar.fhcrc.org/lattice/book/figures.html.

[3] R code and Figures can be found on http://www.stat.auckland.ac.nz/~paul/RGraphics/rgraphics.html.

December 2, 2012
Draft Version

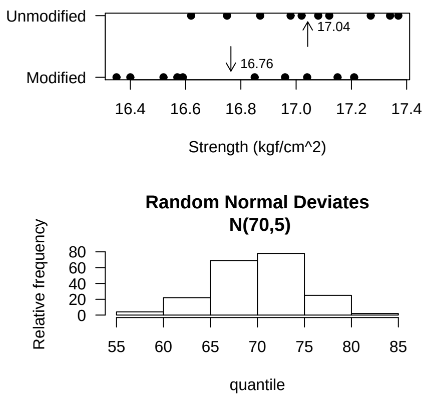


Figure 2.1: Dot diagram for the tension bond strength data (upper panel) and Histogram for 200 normal random deviates (lower panel).

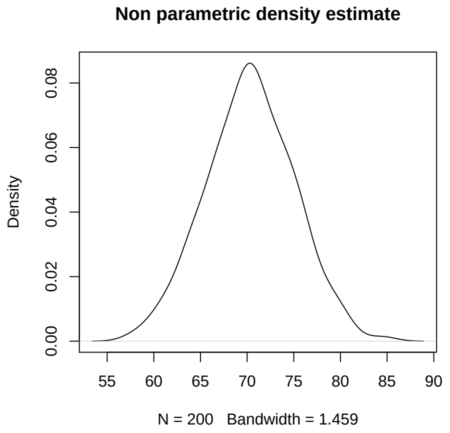


Figure 2.2: Density estimate for the same 200 normal random deviates.

mean and SD. However, a better way to highlight the distribution of the variable under study, especially in its continuous aspect, is to draw a non-parametric density curve, as shown in Figure 2.2. We often get a clearer picture of the underlying distribution, while the appropriate the number of bins used to display the histogram is not always an easy choice. But see [**?**] (pp. 126–130) for additional discussion on this topic.

An other solution is to use a box-and-whisker plot, also called a **boxplot**. As illustrated in Figure 2.3, a lot of information can be found in a boxplot. First, the rectangle box displays half of the total observations, the median being shown inside as an horizontal segment. The upper side of the box is thus the third quartile, while the first quartile is located at the lower side. The extreme tickmarks correspond to the min and max values. However, when an observation exceeds ±1.5 times the inter-quartile range from the median, it is explicitely drawn on the plot, and the extreme tickmarks then correspond to these reference values. This way of handling what could be considered as “extreme values” in R is known as the Tukey’s method. To get such a grahics, one use `boxplot()` function which accept either formula or variable + factor inputs. Figure 2.3 is thus simply produced using

John Tukey (1915–2000) introduced modern techniques for the estimation of spectra of time series, notably the Fast Fourier Transform.

```
boxplot(y,ylab="Strength (kgf/cm^2)",las=1)
```

An example of a Laplace-Gauss—or the “normal”, for short—distribution, with mean 0 and SD 1, is shown in Figure 2.4. As it is a density function, its area equals 1 and any are comprised between two x-values can be calculated very easily using modern computer software. For instance, the shaded gray area, which is the probability $P(1.2 \leq y < 2.4$, is estimated to be 0.107. With R, it is obtained using `pnorm(2.4)-pnorm(1.2)`.

December 2, 2012
Draft Version

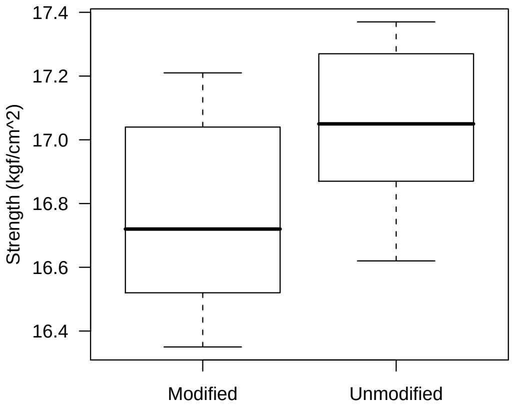


Figure 2.3: Boxplot for the portland cement tension bond strength experiment.

```
x <- seq(-3.5,3.5,by=0.01)
y <- dnorm(x)
plot(x,y,xlab="",ylab="",type="l",axes=F,lwd=2)
axis(side=1,cex.axis=.8); axis(2,pos=0,las=1,cex.axis=.8)
mtext(expression(mu),side=1,line=2,at=0)
mtext(expression(paste(frac(1, sigma*sqrt(2*pi)), " ",
                       plain(e)^{frac(-(x-mu)^2,
                       2*sigma^2)})),side=3,line=0)
# highlight a specific area (drawing is from left to right,
# then from right to left)
polygon(c(x[471:591],rev(x[471:591])),c(rep(0,121),rev(y[471:591])),
        col="lightgray",border=NA)
```

2.3 Testing hypotheses

Statistical hypothesis are generally formulated, based on a given model, as a set of two opposite assertions, the *null hypothesis* being that the statistics reflecting some knowledge about the treatment effect are not different one from the other. Consider a possible analytical model that describes two-sample related outcomes:

$$y_{ij} = \mu_i + \varepsilon_{ij} \qquad i = 1,2;\; j = 1,2,\ldots,n_i, \tag{2.1}$$

where y_{ij} are the observations gathered from (statistical) unit j in group i, and μ_i is the group mean. Then, the corresponding hypothesis that can be formulated is

$$\begin{array}{ll} H_0: & \mu_1 = \mu_2 \\ H_1: & \mu_1 \neq \mu_2. \end{array} \tag{2.2}$$

December 2, 2012
Draft Version

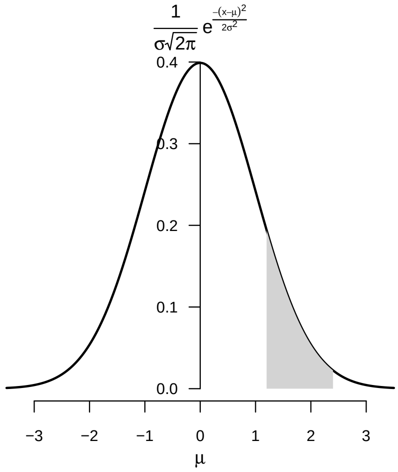


Figure 2.4: The "normal" density function.

Here H_0 denotes the null hypothesis of the absence of effect while H_1 (also denoted H_A by some authors) is the logical negation of H_0.

This testing framework lead to consider two kind of potential errors: Type I error (α) when we reject the null while it is true in the real world, Type II error (β) when the null is not rejected while it should have been. Formally, this is equivalent to

$$\begin{aligned} \alpha &= \Pr(\text{Type I error}) &&= \Pr(\text{reject } H_0 \mid H_0 \text{ is true}) \\ \beta &= \Pr(\text{Type II error}) &&= \Pr(\text{fail to reject } H_0 \mid H_0 \text{ is false}) \end{aligned} \tag{2.3}$$

Using this notation, α is generally refered to as the significance level, and it is what is reported by statistical software when running a given test. Both kind of error are equally important, although Type II error tends to be neglected in many studies. Figure 2.5 highlights the relation between these two quantities, based on two hypothetical distributions. The script is taken from CRAN website (but it is not very difficult to reproduce with a few commands).

2.4 The two-sample t-test

Comparing two set of observations on a response variable involves three steps: (1) constructing a test statistics, (2) defining its sampling distribution, and (3) computing the associated p-value. As already said, the p-value represents the probability of observing a value at least as extremal as that observed using the present data. This is obviously a purely frequentist approach, but it proves to be sufficient in most cases.

December 2, 2012
Draft Version

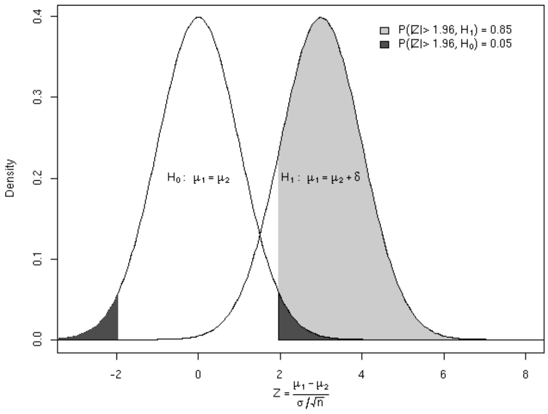


Figure 2.5: Type I and II errors.

The test statistic is given by

$$t_0 = \frac{\bar{y}_1 - \bar{y}_2}{S_p\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}, \tag{2.4}$$

where $\bar{y}_{1,2}$ are the group means, $n_{1,2}$ the sample sizes and S_p an estimate of what is called the pooled variance. When $n_1 = n_2$, the design is said to be *balanced.* The pooled variance is simply the average of the within-group variance, and is computed, in the general case, as

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2}. \tag{2.5}$$

The quantity $n_1 + n_2 - 2$ is called the *degrees of freedom* of the test statistics, that is the number of observations free to vary independently.

There, we must distinguish two approaches in the inferential paradigm and the interpretation of the p-value. According to the Neyman & Pearson's view, the statistical test provides an answer to a purely binary decision (accept or reject the null hypothesis) and the value of the p is not to be interpreted further than its position with respect to a criterion value, say 5%, defined before the start of the experiment[4]. On the contrary, **Fisher** [?] has defended the idea that the value of p itself provides an indication of the strength of the result against the null hypothesis.

Sir Ronald Aylmer Fisher (1890–1962) significantly contributed to the development of methods and sampling distributions suitable for small samples, and he's considered the father of analysis of variance.

There are very long-standing debates on these two approaches and on the way statistical results can be interpreted. We will use most of the time the former approach (binary decision rule) but also provide the value of the resulting p, though it is generally computed based on asymptotic theoretical results.

[4] The Neyman-Pearson criterion says that we should construct our decision rule to have maximum probability of detection while not allowing the probability of false alarm to exceed a certain value α. It can be shown that a likelihood ratio test that reject H_0 in favor of the alternative hypothesis is the most powerful test of size α, though in most case, this test is not used.

December 2, 2012
Draft Version

Confidence interval (CI) can be computed easily based on the sampling distribution of the test statistic, which is the well-known Student $\mathcal{T}(\nu)$ distribution whose quantiles are available in R (see `?qt`). The general formulation of a $100(1-\alpha)\%$ confidence interval for a difference of two means, say $\bar{y}_1 - \bar{y}_2$, is easily obtained as

$$(\bar{y}_1 - \bar{y}_2) \pm t_{\alpha/2,n_1+n_2-2} S_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \tag{2.6}$$

where $\alpha = 0.05$ means a 95% CI. Interesting discussions on the use and interpretation of a confidence interval can be found in articles wrote by Lecoutre and coworkers, e.g. [?, ?].

The function `t.test()` can be applied to the Tension Bond Strength data.

```
t.test(y1,y2,var.equal=TRUE)
```

The output is shown below:

```
Two Sample t-test

data:  y1 and y2
t = -2.1869, df = 18, p-value = 0.0422
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.54507339 -0.01092661
sample estimates:
mean of x mean of y
   16.764    17.042
```

R gives both the t_0, degrees of freedom and p-value, as well as the 95% confidence interval computed using Formula 2.6. The test is significant at the commonly admitted 5% level, or, alternatively, the p-value provides strengthening evidence against the null. We reach a similar conclusion when interpreting the 95% CI as it does not cover 0. Overall, there is a 0.278 kgf/cm^2 difference between the two treatments.

```
as.numeric(diff(apply(y,2,mean)))
```

If we omit the `var.equal=TRUE` option, R computes the Welch modified t-test. In this case, instead of using a pooled variance estimate, degrees of freedom are approximate to get a less liberal p-value; this is also refered to as *Satterthwaite approximate p-value* [?, ?]. The formula for computing degree of freedom is then

$$\nu = \frac{2(w_1 + w_2)}{w_{12}/(n_1 - 1) + w_{22}/(n_2 - 1)} \tag{2.7}$$

Applied to the preceding example, this gives a t-value of -2.187, with 17.025 df, and a p-value of 0.043.

```
t.test(y1,y2)
```

As reporting a non-integer degree of freedom may be confusing, it is often neglected. Here, as variance are not too different between the two groups, we get quite comparable p-value because it isn't necessary to adjust very strongly the degrees of freedom of the test statistic.

December 2, 2012
Draft Version

2.5 Comparing a single mean to a criterion value

2.6 Application to paired samples

Another situation arises when the two samples are related in some way. For example, we can imagine an experiment where a number of specimens are tested by both tip 1 and tip 2. Data are in `hardness.txt`.

```
tmp <- scan("hardness.txt",sep=",")
hardness <- data.frame(y=tmp,tip=gl(2,10))
t.test(y~tip,data=hardness,paired=TRUE)
```

Here, we cannot conclude to a significant difference between the two tips ($t(9) = -0.26$, $p = 0.798$). If we look at a plot of the two evaluations (Fig. 2.6, left), we can see that both are distributed along a line with slope 1 suggesting a close agreement between Tip 1 and Tip 2. In this particular context, a more useful way of checking agreement is to plot the difference between Tip 1 and Tip 2 as a function of the sum of the two evaluations (Fig. 2.6, right). This was initially proposed for assessing biomedical agreement by [?].

```
with(hardness, plot(y[tip==1],y[tip==2],xlab="Tip 1",ylab="Tip 2"))
abline(0,1)
with(hardness, plot(y[tip==1]+y[tip==2],y[tip==1]-y[tip==2],
                    xlab="Tip 1 + Tip 2",ylab="Tip 1 - Tip 2",ylim=c(-3,3)))
abline(h=0)
```

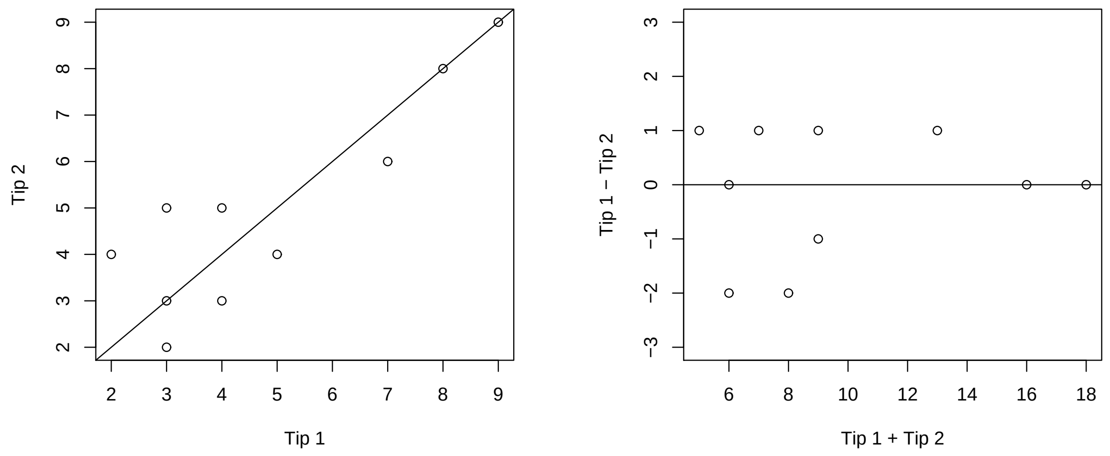


Figure 2.6: The Hardness testing experiment.

Let's look at we would get if we ignore the pairing:

December 2, 2012
Draft Version

```
t.test(y~tip,data=hardness,var.equal=TRUE)
```

As expected, the degree of freedoms are twice the previous ones ($n_1 + n_2 - 2 = 2(n-1)$ when $n_1 = n_2 = n$) and the t-value is larger reflecting the extra variance not accounted for.

2.7 Non-parametric alternative

For two-sample comparisons, two non-parametric tests can be used, depending on the way data are collected. If both sample are independent, we use Mann-Whitney-Wilcoxon rank sum test, while for paired sample the corresponding test is called Wilcoxon signed rank test.

Both are called using R function `wilcox.test` and the option `paired={TRUE|FALSE}`. For the previous examples, we get

```
wilcox.test(y1,y2)
wilcox.test(y~tip,data=hardness,paired=TRUE)
```

December 2, 2012
Draft Version

Chapter 3

Experiments with a Single Factor: The Analysis of Variance

3.1 Summary of Chapter 3

Montgomery reviews the basic principles underlying the one-way ANOVA Model under both the “classical” approach (based on sum of squares colated in the so-called ANOVA table) and the regression approach (based on the estimation of model parameters and solving normal equations). Once the full model has been evaluated, it is often necessary to determine which of the treatment means really differ one from the other. Thus, it calls for multiple comparison procedures which take care of the Type I error inflation caused by the multiplicity of hypothesis tests. Another approach includes the design of orthogonal contrasts which do not inflate the experiment-wise error rate. Finally, a non parametric alternative, the Kruskal-Wallis ANOVA, is presented, as well as its multiple comparisons counterpart.

3.2 Analysis of the fixed effects model

The Etch Rate data ara available in the file `etchrate.txt`. Before starting the analysis, we may want to view graphically the evolution of the observed response (Fig. 3.1).

```
etch.rate <- read.table("etchrate.txt",header=T)
grp.means <- with(etch.rate, tapply(rate,RF,mean))
with(etch.rate, stripchart(rate~RF,vert=T,method="overplot",pch=1))
stripchart(as.numeric(grp.means)~as.numeric(names(grp.means)),pch="x",
           cex=1.5,vert=T,add=T)
title(main="Etch Rate data",ylab=expression(paste("Observed Etch Rate (",
      ring(A),"/min)")),xlab="RF Power (W)")
legend("bottomright","Group Means",pch="x",bty="n")
```

As can be seen from this scatterplot, there is a clear increase in observed etch rate as RF Power also increases. Indeed, mean etch rate evolves from 551.2 Å/min at 160 W to

December 2, 2012
Draft Version

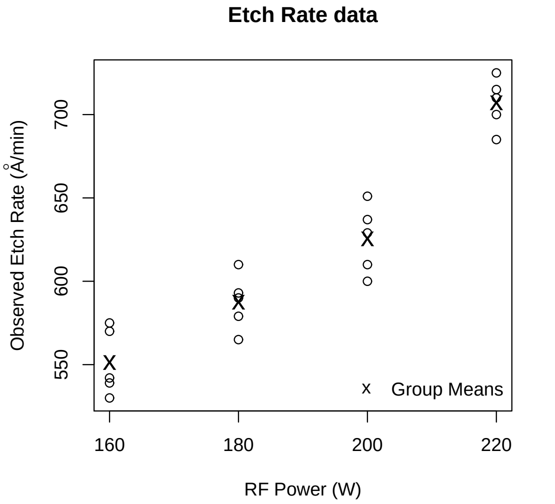


Figure 3.1: The Etch Rate data.

707.0 Å/min at 220 W. Moreover, it seems that this increase occurs in a linear fashion, but we will return to this point later on.

In its most basic formulation, the one-way model of ANOVA is expressed as

$$y_{ij} = \mu_i + \varepsilon_{ij} \qquad i = 1, \ldots, a;\ j = 1, \ldots, n, \tag{3.1}$$

where y_{ij} is the jth observation associated to treatment (or group) i, μ_i is the treatment mean, and ε_{ij} is the so-called residual value assumed to be NIID. Equation 3.1 is called the *means model*. If we consider the μ_i with respect to the overall mean, denoted as μ with $\mu_i = \mu + \tau_i$, then we can rewrite Equation 3.1 as

$$y_{ij} = \mu + \tau_i + \varepsilon_{ij} \qquad i = 1, \ldots, a;\ j = 1, \ldots, n. \tag{3.2}$$

Now, it can be seen that the τ_i represent the difference between treatment means and the overall mean, and they are called the effects, thus we talked about an *effect model*.

3.3 Estimating Model parameters

The ANOVA table (Tab. 3.1) is produced using the next commands. The `aov()` and `lm()` functions are of particular significance when running any ANOVA Model, but it is important to emphasize that the coding of variable is very important, especially when using the `lm()` command. In that particular case, *categorical variables should be* `factor` *in the R terminology*, otherwise a linear regression will be performed!

```
# first, we convert each variable to factor
etch.rate$RF <- as.factor(etch.rate$RF)
```

December 2, 2012
Draft Version

```
etch.rate$run <- as.factor(etch.rate$run)
# next, we run the model
etch.rate.aov <- aov(rate~RF,etch.rate)
summary(etch.rate.aov)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
RF	3	66870.55	22290.18	66.80	0.0000
Residuals	16	5339.20	333.70		

Table 3.1: Results of the ANOVA model for the Etch Rate Experiment.

RF Mean Square largely overcomes residuals (also called *error*) MS which is computed simply as the mean of the within-treatment variances. In other words, there is much more between-treatment variability than within-treatment ones, and such a result is very unlikely to occur by chance alone. Note that R doens't produce a separate row for Total SS as it doesn't provide any additional information most of the time. Here, Total SS simply equals $66870.55 + 5339.20 = 72209.75$ (with $3 + 16 = 19$ df).

Turning back to the ANOVA result, we can estimate the overall mean and the treatment effects $\hat{\tau}_i = \bar{y}_{1\cdot} - \bar{y}_{\cdot\cdot}$ as follows:

```
# overall mean
(erate.mean <- mean(etch.rate$rate))
# treatment effects
with(etch.rate, tapply(rate,RF,function(x) mean(x)-erate.mean))
```

This gives

RF	160	180	200	220
$\hat{\tau}_i$	-66.55	-30.35	7.65	89.25

Another way to get the same result is to use the `model.tables()` function which provides valuable effect size information with complex design.

```
model.tables(etch.rate.aov)
```

Finally, to get $100(1-\alpha)\%$ confidence intervals for treatment effects, we may compute them by hand using the following formula

$$\bar{y}_{i\cdot} \pm t_{\alpha/2,N-a}\sqrt{\frac{\mathrm{MS}_e}{n}} \tag{3.3}$$

whereas for any two treatments comparison the above formula becomes

$$(\bar{y}_{i\cdot} - \bar{y}_{j\cdot}) \pm t_{\alpha/2,N-a}\sqrt{\frac{2\mathrm{MS}_e}{n}} \tag{3.4}$$

Using R directly, we only have to compute the pooled SD and get the value of the corresponding t quantile.

December 2, 2012
Draft Version

```
MSe <- summary(etch.rate.aov)[[1]][2,3]
SD.pool <- sqrt(MSe/5)
t.crit <- c(-1,1)*qt(.975,16)
```

Thus, we can compute any 95% CI for a single mean as $\bar{y}_{i\cdot} \pm 17.3$, and for any pair of means as $(\bar{y}_{i\cdot} - \bar{y}_{j\cdot}) \pm 24.5$. For the latter, we can compare what we would get with a t-test for independent samples, eg. for $\bar{y}_{1\cdot} - \bar{y}_{2\cdot}$:

```
with(etch.rate, t.test(rate[RF==160],rate[RF==180],var.equal=TRUE))
```

The corresponing 95% IC for the difference of means is: [-63.11;-9.29]. This happens to be slighlty different from the above calculations which lead to [-60.7;-11.7]. In any two cases, the IC doesn't include 0, suggesting a significant difference between the two means. However, in the ANOVA framework, we are using the pooled SD calculated on all subsamples (SD=8.17)

```
mean(tapply(etch.rate$rate,etch.rate$RF,var))/5
```

while for the t-test only the variances related to the two samples are taken into account. In this case, the pooled SD is larger and equals 13.05 which explain why the previous CI differ in their respective amplitude.

```
mean(c(var(etch.rate$rate[etch.rate$RF==160]),
       var(etch.rate$rate[etch.rate$RF==180])))
```

Generally speaking, when the F-test from an ANOVA table is significant, it means that at least one pair of means is significantly different (or alternatively, $\exists\, i,j\ (i \neq j)\,|\,\mu_{i\cdot} - \mu_{j\cdot} \neq 0$). We should observe a convergent result using a t-test on the corresponding groups. But it may not be advised to simultaneously test all pairs of means using simple t-test as the Type I error would increase with the number of comparisons made. This way, the probability of detecting a significant difference would be largely greater than $\alpha = 0.05$. One way to confine the experimentwise error rate to a given α is to replace $\alpha/2$ in the previous expression with $\alpha/2r$, where r is the number of intervals computed at the same time. This results from **Bonferroni**'s Theorem.

Carlo Bonferroni (1892–1960) contributes to probability theory more than to simultaneous statistical inference. Bonferroni's adjustement relies rather on *Boole's inequality* although he wrote two articles about this subject.

As R is an Object-Oriented language—and this is even more apparent with the recent development of S4 classes—we may be tempted to apply the `confint()` function on the `aov` object directly. However, if we observe the resulting R output:

```
> confint(etch.rate.aov)
                2.5 %    97.5 %
(Intercept) 533.88153 568.51847
RF180        11.70798  60.69202
RF200        49.70798  98.69202
RF220       131.30798 180.29202
```

we see that it is far from the expected 95% CI associated to treatment effect if they are expressed as difference between group means and overall etch rate mean (τ_i). What is computed instead is the estimated 95% CI for treatment effect substracted to a baseline

December 2, 2012
Draft Version

or reference level, here the first level of RF (ie. 160 W). So, the difference $\bar{y}_{4\cdot} - \bar{y}_{1\cdot}$ is estimated to lie between 131.3 and 180.3 95% of the time[1]. We can check the correctness of the result using Equation 3.4, eg. for the last row labeled `RF220`:

```
as.numeric(grp.means[4]-grp.means[1])+c(-1,1)*qt(.975,16)*sqrt(2*MSe/5)
```

3.4 Model checking

Model checking includes the verification of the following assumptions (in decreasing order of importance):

1. independence,
2. homoscedasticity (homogeneity of the within-group variances),
3. normality of the residuals.

In short, residuals values, defined as $e_{ij} = y_{ij} - \hat{y}_{ij}$, should be structureless and well balanced between treatments.

Model checking can be done graphically and this often is the recommended way, although there exists a formal test for each of the above hypotheses. Several diagnostic plots are proposed in Figure 3.2.

```
opar <- par(mfrow=c(2,2),cex=.8)
plot(etch.rate.aov)
par(opar)
```

When applied on the result of a call to `aov()`, the `plot` method produces different graphics which can be controlled with the `which=` option. R defaults to produce (see `?plot.lm`): a plot of residuals against fitted values, a Scale-Location plot of $\sqrt{e_{ij}}$ against fitted values, a Normal Q-Q plot, and a plot of residuals against leverages. The first two subplots are very useful to check for any departure from the homoscedasticity and normality assumptions. The last plot (residuals vs. leverages) provides insight in possible influencing observations. This is a convenient wrapper function for diagnostic plots that have to be plotted separetely otherwise. For instance, `plot(fitted(etch.rate.aov),residuals(etch.rate.ac` produces the first plot (residuals against fitted values), while `qqnorm(residuals(etch.rate.aov)); qqline(re` corresponds to a Normal Q-Q plot. It could be useful to use these commands when we're interested only in derived plan or subset of predictors.

Standardized residuals may be used as a rough check for outliers. They are defined as

$$d_{ij} = \frac{e_{ij}}{\sqrt{MSe}} \tag{3.5}$$

and it can be shown that they are distributed as $\mathcal{N}(0,1)$ provided $\varepsilon_{ij} \sim \mathcal{N}(0,\sigma^2)$. About 95% of the standardized residuals should fall within ±2.

[1] Recall that the probability is attached to the confidence interval which is random, not to the true (population) parameter.

December 2, 2012
Draft Version

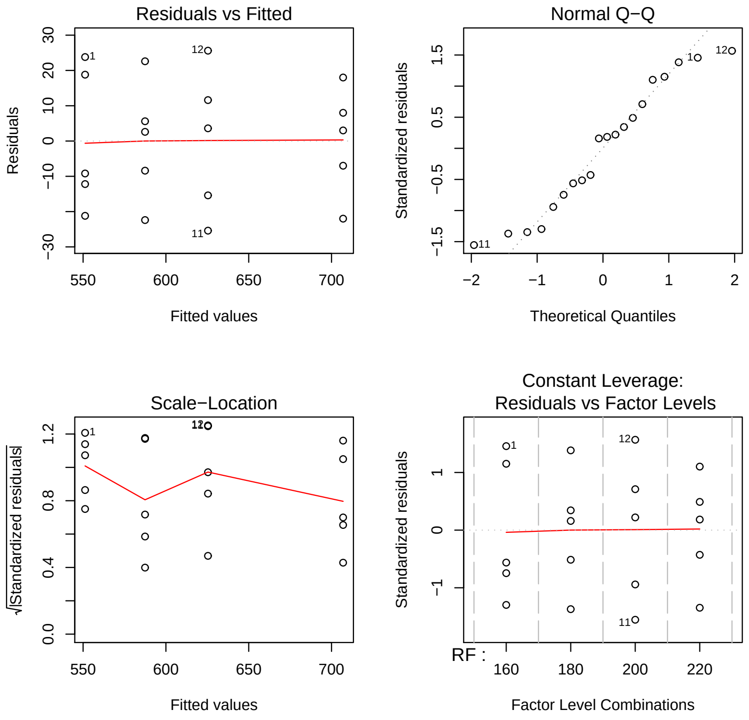


Figure 3.2: Model checking for the ANOVA model.

Independence of observations is largely a matter of the experimental design and the way data are collected. Perhaps the simplest graphical way to check for independence is to plot the residuals against run order or any time index (Fig. 3.3). This also allows to check for the homoscedasticity assumption since any departure from constant variance would be reflected in localized subsets of observations differing in their mean response, or any systematic pattern of outlyiers.

Looking at the plot in Figure 3.3, no such pattern are visible thus we have no reason to reject the independence hypothesis. A more formal test, and an historical ones, is called the Durbin-Watson. This procedures aims at testing the serial autocorrelation of errors and by default makes use of constant lag of 1. It is readily available in the `car` and `lmtest` packages.

```
require(car)
durbin.watson(etch.rate.aov)
```

The assumption of constant variance, or homoscedasticity, is probably the most important in practice since we compute a pooled variance estimate by averaging the within-treatment variance. Any departure from this hypothesis means that some of the groups have larger or smaller variance than other, and this causes our estimate to be somewhat inaccurate. The question of what should be considered as significantly

December 2, 2012
Draft Version

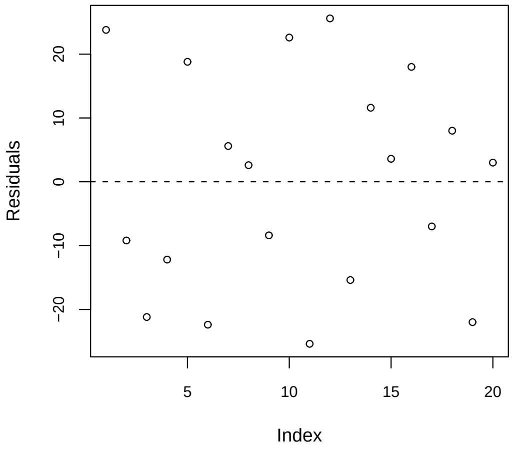


Figure 3.3: Model checking for the ANOVA model (2).

"larger" or "smaller" depends on what is being measured, but it is worth noting that any formal test leading to the rejection of the hypothesis of constant variance cannot help to answer this question. Indeed, if we reject $H_0 : \sigma_1^2 = \sigma_2^2 = \cdots = \sigma_a^2$, what can we say then? Nevertheless, the most widely recommended test of homoscedasticity is **Bartlett**'s test.

Maurice Stevenson Bartlett (1910–2002) worked on the analysis of data with spatial and temporal patterns. He is also known for his contribution in the theory of statistical inference and multivariate analysis.

```
bartlett.test(rate~RF,data=etch.rate)
```

In case one suspect strong departures from normality, we may use Levene's testa s an laternative test for homogeneity of variances. This test is available in the `car` package.

```
levene.test(etch.rate.aov)
```

Finally, the normality of the residuals can be assessed directly using a Q-Q plot as in Figure 3.2 (the so-called *droite de Henry*, in French) where we expect the values to lie approximately on the first bisecting line, or using the Shapiro-Wilk's test. Note that in this latter case, the test should be carried out on each subsample separately, which might be problematic with few replications per subgroup.

```
shapiro.test(etch.rate$rate[etch.rate$RF==160])
```

3.5 Comparison among treatment means

Given our $a = 4$ treatments, we have a set of $4(4-1)/2$ comparisons, the null hypothesis being $H_0 : \mu_i = \mu_j$ for a given (i, j) pair of treatment means. There are several ways to carry out parametric multiple comparisons within R. Perhaps the most common and easy to understand is the systematic pairwise comparison between every treatment

December 2, 2012
Draft Version

means. To prevent from inflating Type I error, again, several methods have been proposed. Among them, the most conservative is the Bonferroni correction which adjust the nominal α value by the number of comparisons (we already discussed this kind of procedure page 15).

First, a pairwise t-test with either Bonferroni or Hochberg correction lead to the rejection of all null hypotheses regarding equality of treatment means (Tab. 3.2 and 3.3). There are some differences in the p-values computed in each case because of the adaptive way of handling the correction factor in the Hochberg case.

```
pairwise.t.test(etch.rate$rate,etch.rate$RF,p.adjust.method="bonferroni")
pairwise.t.test(etch.rate$rate,etch.rate$RF,p.adjust.method="hochberg")
```

	160	180	200
180	0.038	–	–
200	5.1e-05	0.028	–
220	2.2e-09	1.0e-07	1.6e-05

Table 3.2: Bonferroni method.

	160	180	200
180	0.0064	–	–
200	2.5e-05	0.0064	–
220	2.2e-09	8.5e-08	1.1e-05

Table 3.3: Hochberg method.

Another alternative is to use a modified test statistic, to take into account the Type I error inflated by multiple test. This is the approach taken by the Tukey HSD[2] test [?]. R function `TukeyHSD()` gives both adjusted p-value and 95% CI. Furthermore, there is a `plot` method that provides a nice graphical summary (Fig. 3.4). Applying the Tukey HSD test, we raise to the same conclusions as with the "protected" t-tests. Results are given in Table 3.4 and Figure 3.4 where it can be seen that none of the 95% CI includes 0.

```
TukeyHSD(etch.rate.aov)
plot(TukeyHSD(etch.rate.aov),las=1)
```

$i-j$	δ	LB-CI	UP-CI	adj. p
180-160	36.2	3.145624	69.25438	0.0294279
200-160	74.2	41.145624	107.25438	0.0000455
220-160	155.8	122.745624	188.85438	0.0000000
200-180	38.0	4.945624	71.05438	0.0215995
220-180	119.6	86.545624	152.65438	0.0000001
220-200	81.6	48.545624	114.65438	0.0000146

Table 3.4: Tukey HSD method.

The 160–180 and 200–180 pairs of treatment means lead as before to p-values comprised between 0.05 and 0.01, well above the other p-values. This also apparent from the lower bound of the 95% CI shown in Figure 3.4.

[2] HSD stands for *Honest Statistical Difference*.

December 2, 2012
Draft Version

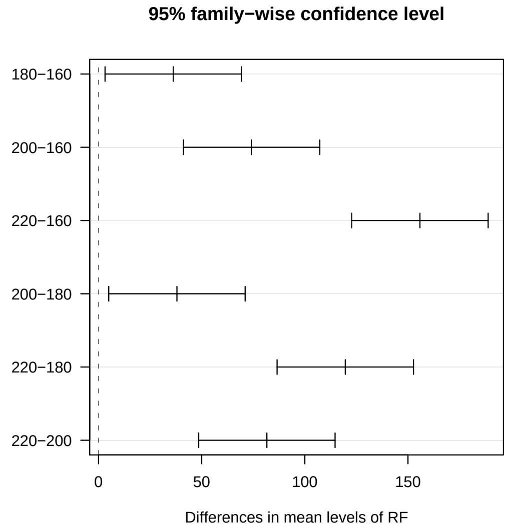


Figure 3.4: Example of an Operating Characteristic curve for the one-way ANOVA (Etch rate example).

Other methods will not be discussed here, but the interested reader is referred to [?] (Chapter 5) or [?] for further descriptions of the pro and cons of the different procedures. R offers a dedicated package called `multcomp` (see also the `Design` package) which can handle multiple comparisons for Linear Models. Another useful reference is [?] with the accompanying package `multtest`[3].

As an alternative to the previous techniques, one can construct specific contrasts for testing only some of treatment means one to the other. If these contrasts, or difference of means, are designed such that they are orthogonal altogether, then tests can be done at a nominal 0.05 level without inflating the overall error rate.

There are various ways to design such contrasts in R. We here review only two of them, with the hope that the reader will not be confused by some of the matrix algebra involved in the former method.

3.6 Power and Sample size

Power and sample size determination are two related concepts. In R, the function `power.t.test()` allows for the necessary computations for the one and two-sample t-test. In the case of the one-way ANOVA (with fixed effects), there is a function called `power.anova.test()` which do that job, as well as `powerF()` in the `QuantPsyc` package. This last function relies on the idea that the F distribution can be manipulated such that arranging its degrees of freedom (especially that in the denominator for sample

[3] For the moment, I only tried some of its functionnalities, and I wrote a very brief note entitled *Multiple comparisons and p-value adjustment* which can be consulted from here: www.aliquote.org/memos/

December 2, 2012
Draft Version

size calculation) or the effect size reflected in the value of any F test computed from an ANOVA or Regression analysis allows the user to get an estimate of either sample size or power for a given design [?]. Generally, power calculation relies on Operating Charactristic curves, where the probability of a Type II error is plotted against a parameter Φ (see Eq. 3.6). An example of such an OC curve, applied to the etch rate experiment, is given is Figure 3.5.

There are basically two very common situations: one in which the experimenter specifies the expected treatment means under the alternative, and the other where the experimenter specifies the minimum difference between any two pair of treatment means.

For the first case, we consider an application using the plasma etching experiment. Suppose that the experimenter expects to reject the null with a power of 0.90 if (and only if) the four treatment means are

$$\mu_1 = 575 \qquad \mu_2 = 600 \qquad \mu_3 = 650 \quad \text{and} \quad \mu_4 = 675,$$

considering $\alpha = 0.01$ and $\sigma = 25$ Å/min. This way, we have

$$\Phi^2 = \frac{n\sum_{i=1}^{4}\tau_i^2}{a\sigma^2} = \frac{n(6250)}{4(25)^2} = 2.5n \tag{3.6}$$

Using R, the following code computes the required sample size to get a power of 0.90 (ie. $\beta \leq 0.01$).

```
grp.means <- c(575,600,650,675)
power.anova.test(groups=4,between.var=var(grp.means),within.var=25^2,
                 sig.level=.01,power=.90)
```

This gives the following results:

```
     Balanced one-way analysis of variance power calculation

         groups = 4
              n = 3.520243
    between.var = 2083.333
     within.var = 625
      sig.level = 0.01
          power = 0.9

 NOTE: n is number in each group
```

We conclude that $n = 4$ for each treatment group would allow us to detect the above effect sizes wth a power of 0.90. If we're interested in the way the SD (specified a priori) may influence the resulting power, for a fixed sample size, we can run the following script:

```
sd <- seq(20,80,by=2)
nn <- seq(4,20,by=2)
beta <- matrix(NA,nr=length(sd),nc=length(nn))
```

December 2, 2012
Draft Version

```
for (i in 1:length(sd))
  beta[i,] <- power.anova.test(groups=4,n=nn,between.var=var(grp.means),
                               within.var=sd[i]^2,sig.level=.01)$power
colnames(beta) <- nn; rownames(beta) <- sd
opar <- par(las=1,cex=.8)
matplot(sd,beta,type="l",xlab=expression(sigma),ylab=expression(1-beta),col=1,
        lty=1)
grid()
text(rep(80,10),beta[length(sd),],as.character(nn),pos=3)
title("Operating Characteristic Curve\n for a=4 treatment means")
par(opar)
```

As can be seen from Figure 3.5, increasing SD or decreasing sample size (shown on the right of the plot) results in a loss of power. For instance, with $\sigma = 50$ (twice the value postulated in the previous calculation) and $n = 6$, we get only a power of 0.60 (for a given $\alpha = 0.01$). We would have to increase the sample size up to $n = 10$ to get at least a power > 0.80.

As an illustration of the second power calculation objective, consider the folowing situation. The experimenter now whishes to reject the null with probability ≥ 0.90 if any two treatment means differed by as much as 75 Å/min, with $\alpha = 0.01$. Considering an SD of $\sigma = 25$, the minimum value of Φ^2 is estimated to be

$$\Phi^2 = \frac{nD^2}{2a\sigma^2} = \frac{n(75)^2}{2(4)(25^2)} = 1.125n \tag{3.7}$$

Estimation of n can be done using OC curve; however, I haven't found any R function that do that computation automatically.

3.7 Non-parametric methods in ANOVA

A non-parametric alternative to the one-way ANOVA is the Kruskal-Wallis ANOVA, which is quite similar except it is based on the ranks of the observations rather than their nominal values. The Kruskal-Wallis test is used to test the null hypothesis that the a treatments are identical with respect to their central position (here, the median). It should be kept in mind that non-parametric tests assume that within-groups dispersion is also homogeneous as in the parametric approach. Indeed, if this was not the case, we could hardly segregate any individual from one group from an other one as their distance might be accounted for by the difference in variance of the subpopulations.

Carrying out the K-W test,

```
kruskal.test(rate~RF,data=etch.rate)
```

gives a test statistic of 16.91 with 3 df and a p-value < 0.001. This result doesn't refute our preliminary conclusion using the classical ANOVA Model.

In the non-parametric approach, multiple comparisons techniques challenge the usual "easiness" of parametric ones. Indeed, this raises the question of how to test

December 2, 2012
Draft Version

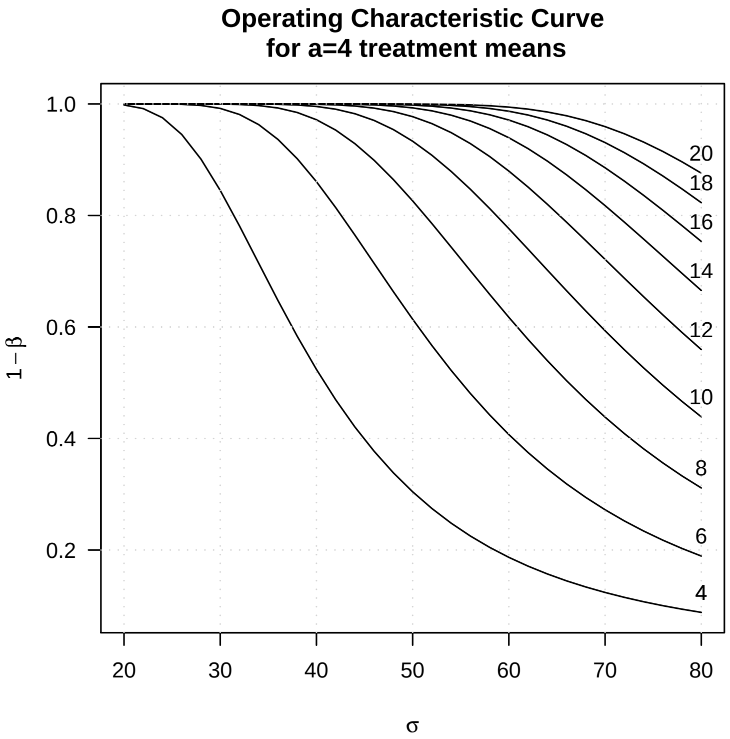


Figure 3.5: Example of an Operating Characteristic curve for the one-way ANOVA (Etch rate example).

The `npmc` packages offers NP multiple hypothesis testing for the unbalanced one-way layout, based on Behrens-Fisher and Steel procedures. These procedures come from [?].

```
library(npmc)
# we need to reformat the data.frame with var/class names
etch.rate2 <- etch.rate
names(etch.rate2) <- c("class","run","var")
summary(npmc(etch.rate2),type="BF")
```

The results are summarized in Table 3.5.

Compared to parametric multiple comparisons (§ 3.5), we reach similar conclusions: every pair of treatment means can be considered as significantly different one from the other.

Further Notes on distribution assumptions. It is worth to remind the reader that non-parametric tests share common assumptions with parametric counterparts, in particular the hypothesis of comparable dispersion. It also applies to permutation tests. If we were to compare two distributions for which variance (or whatever dispersion measure is used) strongly differ one from the other, it would not make sense to use

December 2, 2012
Draft Version

i–j	effect	LB-CI	UP-CI	p-value 1s	p-value 2s
			Behrens-Fisher		
1-2	0.92	0.5764163	1.263584	0.011450580	0.020539156
1-3	1.00	0.9998842	1.000116	0.000000000	0.000000000
1-4	1.00	0.9998842	1.000116	0.000000000	0.000000000
2-3	0.94	0.6758851	1.204115	0.002301579	0.004440345
2-4	1.00	0.9998842	1.000116	0.000000000	0.000000000
3-4	1.00	0.9998842	1.000116	0.000000000	0.000000000
			Steel		
1-2	0.92	0.4254941	1.414506	0.07123615	0.13078270
1-3	1.00	0.5054941	1.494506	0.02446374	0.04602880
1-4	1.00	0.5054941	1.494506	0.02417453	0.04631413
2-3	0.94	0.4469949	1.433005	0.05465670	0.10154286
2-4	1.00	0.5054941	1.494506	0.02412958	0.04654181
3-4	1.00	0.5054941	1.494506	0.02414774	0.04635531

Table 3.5: Results from the NP multiple comparisons procedures applied to the etch rate data.

LB/UP-CI: lower and upper-bound of 95% CI; p-values 1s/2s: one-sided and two-sided p-value.

either a parametric or a non-parametric hypothesis test. Indeed, a given observation might be drawn from one or the other distribution, but due to overlapping of the two distributions with differing variance, it wouldn't be possible to associate the individual observation with any of them. In other word, we loose the *exchangeable* hypothesis.

However, a minor modification of the test statistics, as proposed by Welch [**?**], may be used for the case of non-constant variance. Applying the following principle to the etch rate data,

```
oneway.test(rate~RF,etch.rate)
```

gives a F value of 68.72 and a p-value largely $< .001$. As was said for the Welch modified t-test (p. 9), degrees of freedom for the denominator (the residual) are adjusted, they are less commonly reported.

December 2, 2012
Draft Version

Chapter 4

Randomized Blocks, Latin Squares, and Related Designs

4.1 Summary of Chapter 4

4.2 Randomized Complete Block Design

Randomized Complete Block Design (RCBD) is a widely used tools to study some effect of interest while controlling for potential nuisance factor(s). It should not be confounded with covariance analysis whereby response are adjusted a posteriori to take into account nuisance factors.

The so-called Effects model can be expressed as

$$y_{ij} = \mu + \tau_i + \beta_j + \varepsilon_{ij} \qquad (i = 1, 2, \dots, a; j = 1, 2, \dots, b) \tag{4.1}$$

subject to

$$\sum_{i=1}^{a} \tau_i = 0 \quad \text{and} \quad \sum_{j=1}^{b} \beta_j = 0 \tag{4.2}$$

The fundamental ANOVA equation for the RCBD resumes to

$$SS_T = SS_{\text{treat}} + SS_{\text{block}} + SS_E \tag{4.3}$$

where *treat* denotes the treatment factor and *block* the blocking variable. Residual SS, with $(a-1)(b-1)$ degrees of freedom, captures the variance unexplained by the two other factors. The layout of this design is quite comparable to that of a two-way ANOVA with one observation per cell: no interaction term is estimable and the design is orthogonal, so terms can be entered in any order in the model. Note that such an additive formulation of the response variations is not always possible, especially if some interaction between blocks and the factor of interest is to be expected, or is discovered when inspecting residuals vs. fitted values. In this case, a factorial design (Chap. 5 and 6) should be more appropriate to uncover the interaction effect.

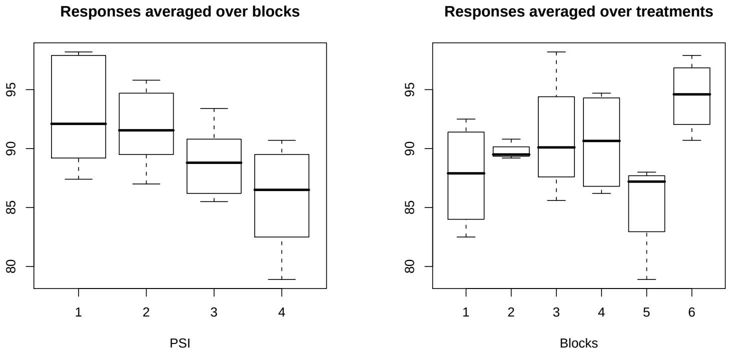
December 2, 2012
Draft Version

Let's consider the following example (Tab. 4-3). A product developer decides to investigate the effect of four different levels of extrusion pressure on flicks using a RCBD considering batches of resin as blocks. The data are contained in the file `vascgraft.txt` and are shown in the following Table.

	Batch of Resin (Block)						
PSI	1	2	3	4	5	6	Total
1	90.30	89.20	98.20	93.90	87.40	97.90	556.9
2	92.50	89.50	90.60	94.70	87.00	95.80	550.1
3	85.50	90.80	89.60	86.20	88.00	93.40	533.5
4	82.50	89.50	85.60	87.40	78.90	90.70	514.6
Total	350.8	359.0	364.0	362.2	341.3	377.8	$y_{..} = 2155.1$

```
x <- scan("vascgraft.txt")
PSI.labels <- c(8500,8700,8900,9100)
vasc.graft <- data.frame(PSI=gl(4,6,24),block=gl(6,1,24),x)
vasc.graft.aov <- aov(x~block+PSI,vasc.graft)
```

Figure 4.1 gives two different pictures of the data. On the left panel, responses have been averaged over blocks, while on the right, data have been averaged on the treatment.



(a) (b)

Figure 4.1: Results of the Vascular Graft Experiment.

If we want to plot both information, we can use a so-called *interaction plot*, even if we are not considering interaction between the two factors. Figure 4.2 plots response as a function of treatment (x-axis) for each block labelled with different color and line type.

Classical ANOVA results, obtained by issuing `summary(vasc.graft.aov)`) in the R shell, are reported in Table 4.1. As can be seen, the effect of treatment (PSI) is highly significant, and the results clearly argue against the null.

December 2, 2012
Draft Version

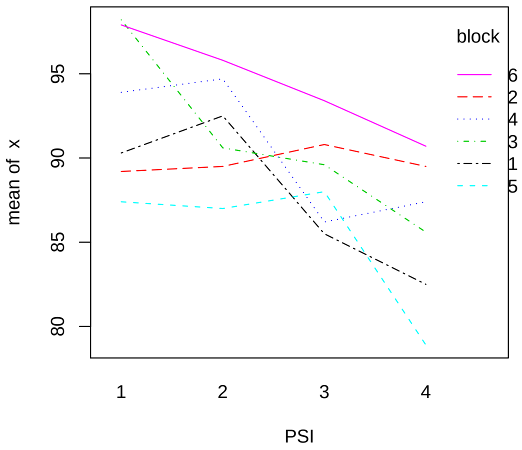


Figure 4.2: Results of the Vascular Graft Experiment (*cont.*).

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
block	5	192.25	38.45	5.25	0.0055
PSI	3	178.17	59.39	8.11	0.0019
Residuals	15	109.89	7.33		

Table 4.1: Results for the Model $y = \mu + \text{PSI}_i + \text{block}_j$.

Ignoring the blocking structure would yield incorrect result, though still significant.

It is always a good practice to check model adequacy after running the ANOVA model. To do so, we have to check the relation between fitted values and residuals (*homoscedasticity*), as well as the normality (of the residuals) hypothesis. Various plots are reproduced in Figure 4.3, including (standardized and raw) residuals vs. fitted values, QQ-plot and leverage effect.

```
opar <- par(mfrow=c(2,2),cex=.8)
plot(vasc.graft.aov)
par(opar)
```

Handling Missing Values. If some missing value occurs in an RCBD, treatments are no longer orthogonal to blocks. Two general solutions are available:

- *approximate analysis*: the missing observation is estimated, and the analysis proceeds as usual, substracting one degree of freedom to the residual SS;
- *exact analysis*: it relies on the general regression significance test, that will not be covered here (please refer to pp. 133–136 of Montgomery's handbook).

December 2, 2012
Draft Version

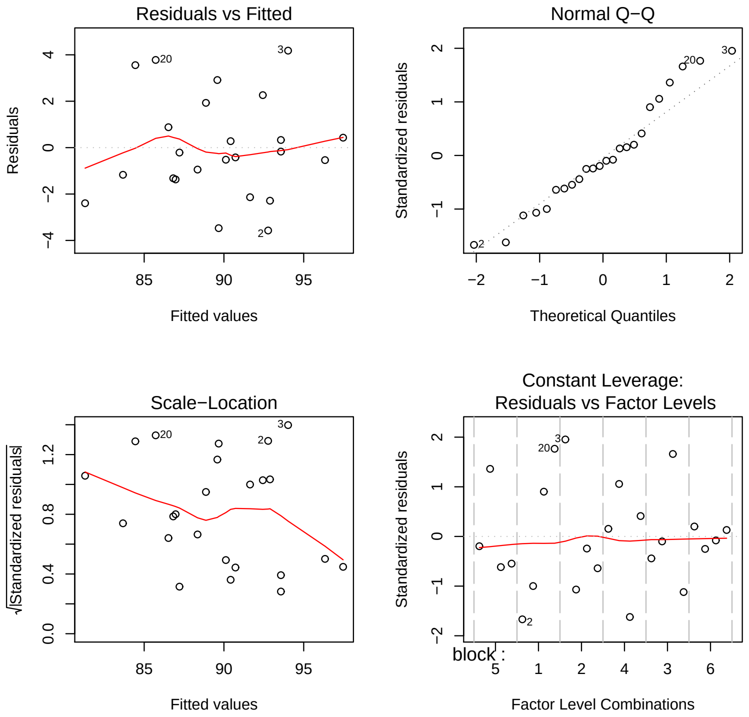


Figure 4.3: Model checking for the Vascular Graft data.

```
# we delete the 10th observation
x2 <- x
x2[10] <- NA
vasc.graft2 <- data.frame(PSI=gl(4,6,24),block=gl(6,1,24),x2)
```

We wish to estimate the missing value, say y_{10}, so that its contribution to error sum of squares is minimal. This is equivalent to finding $\tilde{y_{10}}$ satisfying

$$\min_{y} \sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij}^2 - \frac{1}{b}\sum_{i=1}^{a}\left(\sum_{j=1}^{b}\right)^2 - \frac{1}{a}\sum_{j=1}^{b}\left(\sum_{i=1}^{a}\right)^2 + \frac{1}{ab}\left(\sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij}\right)^2$$

Considering that $\frac{dSS_E}{dy} = 0$, we get

$$\tilde{y_{10}} = \frac{ay'_{i.} + by'_{.j} - y'_{..}}{(a-1)(b-1)}$$

and y_{10} will be imputed a value of 91.08.

December 2, 2012
Draft Version

4.3 Latin Square Design

The Latin Square design is another way to include blocking factors in a given design. This way, we can account for 2 nuisance factors.

Latin squares are arranged by combining two circular permutations of a sequence of treatment (e.g. $\{A, B, C, D, E\}$) on the rows and columns.

The example given by Montgomery on the Rocket Propellant Problem is available in the file `rocket.txt`, which can be imported using

```
rocket <- read.table("rocket.txt",header=T)
```

Treatment allocation is illustrated in the following table.

	Operators				
Batch	1	2	3	4	5
1	A	B	C	D	E
2	B	C	D	E	A
3	C	D	E	A	B
4	D	E	A	B	C
5	E	A	B	C	D

The following command

```
plot(y~op+batch+treat,rocket)
```

allows to sequantially inspect different boxplots of `y` as a function of one of the three factors.

```
rocket.lm <- lm(y~factor(op)+factor(batch)+treat,rocket)
anova(rocket.lm)
```

Because the design is balanced, the order for entering term does not matter. Here are the results of the ANOVA (Tab. 4.2). We should only interpret the F-value associated with the treatment effect (`treat`).

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
factor(op)	4	150.00	37.50	3.52	0.0404
factor(batch)	4	68.00	17.00	1.59	0.2391
treat	4	330.00	82.50	7.73	0.0025
Residuals	12	128.00	10.67		

Table 4.2: Analysis of the Rocket data.

Figure 4.4 is an attempt to resume the structure of the factors effects on the measured response. As can be seen, a large proportion of overall variability in responses is captured by the treatment condition, and operator appear to convey much variance than the blocking factor itself.

December 2, 2012
Draft Version

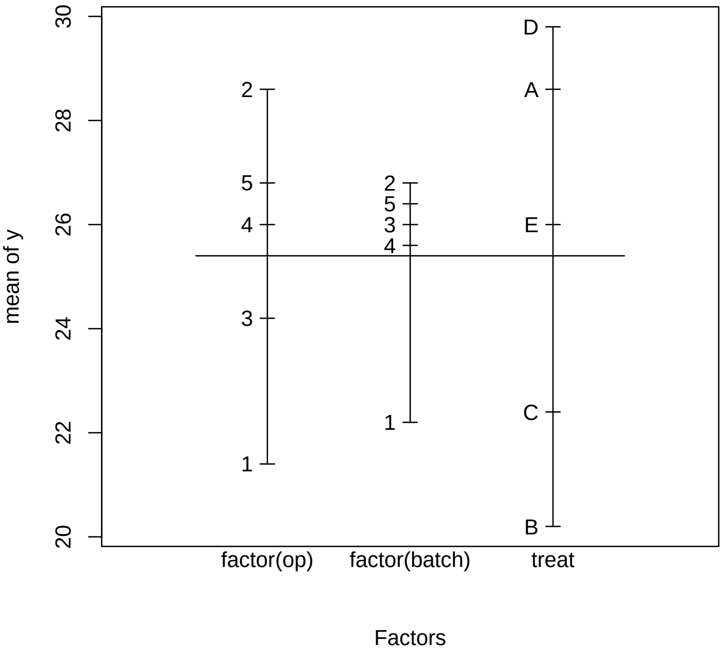


Figure 4.4: Factors effects plot.

4.4 Graeco-Latin Square Design

4.5 Balanced Incomplete Block Designs

Balanced Incomplete Block Designs (BIBD) are a class of randomized block designs whereby every treatment is not observed for every block present in the experiment. If we denote by a the number of treatments, and k the maximum number of treatments for each block ($k < a$), then a BIBD consists in different arrangement of the $\binom{a}{k}$ combinations.

Douglas Montgomery gives a pretty introduction to this class of design, widely used in educational assessment or clinical trials. For additional development on this topic, please refer to [**?**, **?**]. Note, however, that in an educational perspective, what is classicaly refered to a BIBD is not really a BIBD in a formal sense. Indeed, blocks are treated as factor and factor as blocks (e.g. [**?**]).

Consider the following example (Tab. 4-21) of a catalyst experiment, in which the time of reaction for a chemical process is studied as a function of catalyst type administered to four different batch of raw material. These batch are considered as the blocking elements.

Let a be the number of treatments, and b the number of blocks. We consider that each block contains k treatments, with an overall replication of r times in the design. We

December 2, 2012
Draft Version

	Block (Batch of Raw Material)				
Treatment	1	2	3	4	$y_{i\cdot}$
1	73	74	–	71	218
2	–	75	67	72	214
3	73	75	68	–	216
4	75	–	72	75	222
$y_{\cdot j}$	221	224	207	218	$870 = y_{\cdot\cdot}$

thus have $N = ar = bk$ observations, and the number of times each pair of treatments apperas in the same block is:

$$\lambda = \frac{r(k-1)}{a-1}, \qquad \lambda \in \{0, 1, 2, \dots\}$$

When $a = b$, we have a *symmetric* design. As λ has to be an integer, the space of admissible solutions can be considerably reduced for some design. For example, the following constraints: $r = 4$, $t = 4$, $b = 8$, and $k = 2$, are not possible for a BIB.[1]

```
tab.4.21 <- matrix(c(73,NA,73,75,74,75,75,NA,NA,67,68,72,71,72,NA,75),nc=4)
tab.4.21.df <- data.frame(rep=as.vector(tab.4.21),
                          treat=factor(rep(1:4,4)),
                          block=factor(rep(1:4,each=4)))
summary(aov(rep˜treat+block+Error(block),tab.4.21.df))
```

Specifying `block` is required to get a correct estimation of residuals and an adjusted estimate of `treatment` effect. Here, the type of catalyst induce a significant main effect. However, one can also “force” R to use the appropriate SS by using `lm()` with ordered effect terms, e.g.

```
anova(lm(rep˜block+treat,tab.4.21.df))
```

gives the correct treatment effect, adjusted for the blocking factor.

The detailed `aov` output is reproduced below and, as can be seen, R correctly separates the total variability into a block stratum and residual random variations.

```
Error: block
      Df Sum Sq Mean Sq
treat  3 55.000  18.333

Error: Within
          Df  Sum Sq Mean Sq F value  Pr(>F)
treat      3 22.7500  7.5833  11.667 0.01074 *
Residuals  5  3.2500  0.6500
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

[1] But, a partially balanced incomplete block design can be used instead.

December 2, 2012
Draft Version

This way, we have computed adjusted MS for the catalyst effect. We might be interested in the adjusted MS for the block effect. This can easily be found using the appropriate error term, `Error(treat)`, and we get

```
Error: treat
      Df  Sum Sq Mean Sq
treat  3 11.6667  3.8889

Error: Within
          Df Sum Sq Mean Sq F value    Pr(>F)
block      3 66.083  22.028  33.889 0.0009528 ***
Residuals  5  3.250   0.650
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

If we want to get both estimates in a single pass, like **Minitab**, we can wrap the two calls to the `aov()` function in a single function with little effort. Table 4.3 summarizes both estimates (unadjusted and adjusted) and associated p-values.

Effect	df	MS	F	p-value
`treat`	3	3.889		
`treat` (Adj.)	3	7.583	11.667	0.01074
`block`	3	18.333		
`block` (Adj.)	3	22.028	33.889	0.00095

Table 4.3: Summary of BIB analysis.

Another solution is to use the `BIB.test()` function located in the `agricolae` package. Actually, there is no formula interface in function call, so we have to pass separetly the blocking factor, th fixed treatment and the response variable.

```
require(agricolae)
BIB.test(tab.4.21.df$treat,tab.4.21.df$treat,tab.4.21.df$rep,
         method="tukey",group=FALSE)
```

Note. Actually, I did not explore all the functionnalities of this function and its behavior (e.g. parameter `group=`). Further, I cannot get correct result with the above code!

Tukey pairwise differences (`treat` factor) can be computed as follow:

```
tab.4.21.lm <- lm(rep~block+treat,tab.4.21.df)
treat.coef <- tab.4.21.lm$coef[5:7]
# effect for catalyst 4 (baseline) is missing, so we add it
treat.coef <- c(0,treat.coef)
pairwise.diff <- outer(treat.coef,treat.coef,"-")
```

December 2, 2012
Draft Version

Inspecting the output of `summary(tab.4.21.lm)`, we see that the standard error is estimated to be 0.6982. More generally, SE can be obtained as $\sqrt{\frac{2k}{\lambda t}}\hat{\sigma}$. The corresponding Tukey critical value ($1-\alpha = 0.95$) is given by

```
crit.val <- qtukey(0.95,4,5)
```

However, the `BIB.test()` function directly gives LSD, Tukey or Waller-Duncan tests for comparing the treatment levels.

Instead of a tabular summary, we may plot the confidence intervals as a function of each paired comparison (Fig. 4.6, right).

```
ic.width <- crit.val*0.6982/sqrt(2)
xx <- pairwise.diff[lower.tri(pairwise.diff)]
plot(xx,1:6,xlab="Pairwise Difference
    (95% CI)",ylab="",xlim=c(-5,5),pch=19,cex=1.2,axes=F)
axis(1,seq(-5,5))
mtext(c("4-1","4-2","4-3","1-2","1-3","2-3"),side=2,at=1:6,line=2,las=2)
segments(xx-ic.width,1:6,xx+ic.width,1:6,lwd=2)
abline(v=0,lty=2,col="lightgray")
```

Next, we may be interested in assessing whether this BIB performs better than a complete randomized design (without blocking). Following [?], we can compute the relative efficiency as

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

In this case, it happens to be computed as

```
tab.4.21.lm.crd <- lm(rep˜treat,tab.4.21.df)
(summary(tab.4.21.lm.crd)$sig/summary(tab.4.21.lm)$sig)ˆ2
```

and we get a RE of 13%. Thus, a CRD would require 13% more observations to obtain the same level of precision as our BIB.

Recovering Interblock Information calls for some tricky manipulation within R. The best combined estimator can be expressed as $\tau_i^* = \alpha_1\hat{\tau}_i + \alpha_2\tilde{\tau}_i$: it is linear combination of the intrablock and interblock estimators, with weights inversely proportional to the variances of $\hat{\tau}_i$ and $\tilde{\tau}_i$. Following Equation 4-45, it can be estimated as

$$\tau_i^* = \frac{kQ_i(\sigma^2 + k\sigma^2_\beta) + \left(\sum_{j=1}^b n_{ij}y_{\cdot j} - kr\bar{y}_{\cdot\cdot}\right)\sigma^2}{(r-\lambda)\sigma^2 + \lambda a(\sigma^2 + k\sigma^2_\beta)} \qquad i = 1,2,\ldots,a \tag{4.4}$$

As σ^2 and σ^2_β are unknown parameters, they are replaced by their estimates. We will use the error MS from the intrablock analysis, for σ^2.

Figure 4.5 (left) shows the observed response for this 4×4 design. OLS fit have been superimposed for each "random" block. This treillis graphics [?] has been produced using the following commands:

December 2, 2012
Draft Version

```
require(lattice)
xyplot(rep˜treat|block,tab.4.21.df,
       aspect="xy",xlab="Catalyst",ylab="Response",
       panel=function(x,y) {
         panel.xyplot(x,y)
         panel.lmline(x,y)
       })
```

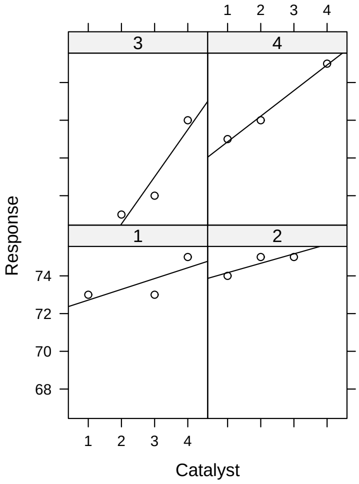


Figure 4.5: The catalyst experiment. Response measured in each block as a function of the type of catalyst $(1, 2, 3, 4)$ used.

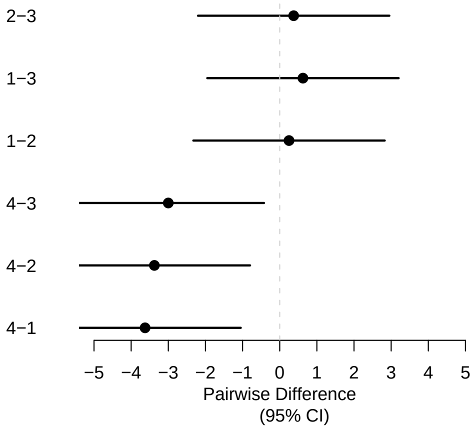


Figure 4.6: Tukey 95% simultaneaous confidence intervals.

We would obtain the same results if we were to use the `lme4` package, which rests in this case on REML estimation.

```
require(lme4)
print(tab.4.21.lm <- lmer(rep˜treat+(1|block),tab.4.21.df),corr=F)
```

Now, we get a rather more informative output including variance components estimates together with their standard errors.

```
Linear mixed-effects model fit by REML
Formula: rep ˜ treat + (1 | block)
   Data: tab.4.21.df
   AIC   BIC logLik MLdeviance REMLdeviance
 44.22 46.64 -17.11      38.57        34.22
Random effects:
 Groups   Name        Variance Std.Dev.
 block    (Intercept) 8.00907  2.83003
```

December 2, 2012
Draft Version

```
 Residual             0.65035  0.80644
number of obs: 12, groups: block, 4

Fixed effects:
            Estimate Std. Error t value
(Intercept)  74.9704     1.4963   50.11
treat1       -3.5573     0.6973   -5.10
treat2       -3.3541     0.6973   -4.81
treat3       -2.9704     0.6973   -4.26
```

Should it be of interest to use other linear contrasts for `treat`, we shall simply remove the intercept from the previous model.

```
print(tab.4.21.lm0 <- lmer(rep~-1+treat+(1|block),tab.4.21.df))
```

However, we can notice in the above output that the intercept term equals 74.97, which doesn't correspond to the observed mean when `treat=4`. In fact, this is the mean of the predicted means (for the 4th catalyst) across the four blocks. This follows from the fact that we introduce `block` as a random effect for which separate intercept have to be computed (`(1|block)`).

```
coef(tab.4.21.lm)[[1]]$`(Intercept)`
mean(coef(tab.4.21.lm)[[1]][,1])
```

Remark. Constructing a BIB is not an easy task. One way to build such a design is to start with a multifactorial design (most often used as a *screening* device when we have multiple factor of interest but want to include only the relevant ones in the mail trial). Quoting [?], we can build a $L_{12}2^{11}$ design by selecting the first column (1st factor), then applying a circular permutation to its elements. The following code is a quick implementation of such a construction but is not optimized anyway.[2]

```
col <- c(1,1,0,1,1,1,0,0,0,1,0)
perm <- function(x) {
  s <- length(x)
  m <- matrix(nc=s,nr=s)
  y <- rep(x,2)
  m[,1] <- x
  for (i in 2:s) { m[,i] <- y[i:(s+i-1)] }
  m
}
col.perm <- perm(col)
bib11 <- rbind(rep(0,11),col.perm)
# check that the design is well balanced
apply(bib11[-1,],1,sum)
apply(bib11,2,sum)
```

[2] More efficient algorithms for permutations are available in various R packages (e.g. `vegan`, `sna`, `dprep`, `e1071`, `gdata`, `magic`).

December 2, 2012
Draft Version

With this particular design, we can check that there are exactly 6 factors per block, and, reciprocally, only 6 blocks are associated with each factor. For reading easiness (at least from my point of view), we can plot the design matrix rather than displaying it in a tabular format (Fig. 4.7). This way, it looks like a confusion matrix.

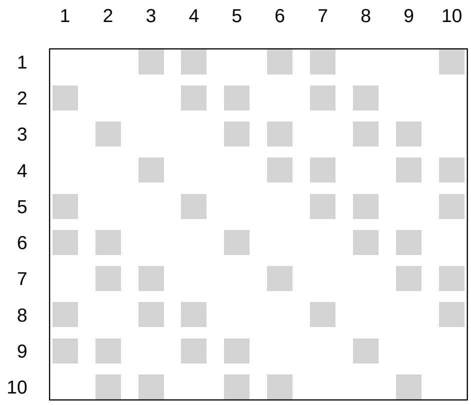


Figure 4.7: A BIBD with 10 blocks × 10 factors.

Other examples of block designs analyzed with R are covered in [?] (Chapter 16).

December 2, 2012
Draft Version

Chapter 5

Introduction to Factorial Design

5.1 Summary of Chapter 5

Chapter 5 deals with the analysis f balanced two-factors design. When appropriately used, factorial designs increase design efficiency, and it can be shown that the same accuracy can be obtained with a minimum of essays compared to separate one-way experiment. The fundamental ANOVA equation is extended to account for the variability explained by a second factor and a possible interaction between the two factors. The concept of interaction is often of primary interest and need to be well understood, both from a scientific and a statistical point of view.

5.2 The two-factor factorial design

In the general case, the effects model ressembles

$$y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \varepsilon_{ijk} \tag{5.1}$$

where i, j ($i = 1 \dots a$, $j = 1 \dots b$) span the levels of factor A and B, while k stands for the observation number ($k = 1 \dots n$). The order in which the abn observations are taken is selected at random, so this design is said to be a *completely randomized design*.

In case one or more factor are quantitative, a regression model is even easily formalized. Note that if we write down the normal equations related to the above model, it can be shown that there are $a + b + 1$ linear dependencies in the system of equations. As a consequence, the parameters are not uniquely determined and we say that the model is not directly *estimable* without imposing some constraints. This happens to be: $\sum_{i=1}^{a} \hat{\tau}_i = 0$, $\sum_{j=1}^{b} \hat{\beta}_j = 0$, $\sum_{i=1}^{a} \widehat{\tau\beta}_{ij} = 0$ ($j = 1, 2, \dots, b$) and $\sum_{j=1}^{b} \widehat{\tau\beta}_{ij} = 0$ ($i = 1, 2, \dots, a$).

December 2, 2012
Draft Version

With some algebra, 5.1 can be expressed as a (corrected) total sum of sum of squares:

$$\begin{aligned}\sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{n}(y_{ijk}-\bar{y}_{\cdot\cdot\cdot})^2 &= \sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{n}[(\bar{y}_{i\cdot\cdot}-\bar{y}_{\cdot\cdot\cdot})+(\bar{y}_{\cdot j\cdot}-\bar{y}_{\cdot\cdot\cdot})\\ &\quad+(\bar{y}_{ij\cdot}-\bar{y}_{i\cdot\cdot}-\bar{y}_{\cdot j\cdot}+\bar{y}_{\cdot\cdot\cdot})+(y_{ijk}-\bar{y}_{ij\cdot})]^2\\ &= bn\sum_{i=1}^{a}(\bar{y}_{i\cdot\cdot}-\bar{y}_{\cdot\cdot\cdot})^2+an\sum_{j=1}^{b}(\bar{y}_{\cdot j\cdot}-\bar{y}_{\cdot\cdot\cdot})^2\\ &\quad+n\sum_{i=1}^{a}\sum_{j=1}^{b}(\bar{y}_{ij\cdot}-\bar{y}_{i\cdot\cdot}-\bar{y}_{\cdot j\cdot}+\bar{y}_{\cdot\cdot\cdot})^2\\ &\quad+\sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{n}(y_{ijk}-\bar{y}_{ij\cdot})^2\end{aligned}\tag{5.2}$$

Symbolically, this decomposition can also be expressed as:

$$SS_T = SS_A + SS_B + SS_{AB} + SS_E \tag{5.3}$$

and as can be seen from the last component of the right-hand side of Equation 5.2, there must be at least two replicates ($n \geq 2$) to obtain an error sum of squares. As for the one-way layout, this component will be called the residual or the error term.

Hypotheses testing proceeds in three steps:

- equality of row treatment effects
  $H_0: \ \tau_1 = \tau_2 = \cdots = \tau_a = 0$
- equality of column treatment effects
  $H_0: \ \beta_1 = \beta_2 = \cdots = \beta_b = 0$
- no interaction between row and column treatment
  $H_0: \ (\tau\beta)_{ij} = 0 \quad \text{for all } i, j$

Applied to the data found in `battery.txt`, we can set up a 3^2 factorial design (two factors at three levels) very easily. The data consists in a study of the effect of temperature (°F) and a design parameter with three possible choices. The aim is to design a battery for use in a device subjected to extreme variations of temperature.

```
battery <- read.table("battery.txt",header=TRUE)
battery$Material <- as.factor(battery$Material)
battery$Temperature <- as.factor(battery$Temperature)
summary(battery)
```

Now, the two-way ANOVA model, including an interaction effect, is computed as follows:

```
battery.aov <- aov(Life~Material*Temperature,data=battery)
```

December 2, 2012
Draft Version

Note that `Life~Material*Temperature` is equivalent to `Life~Material+Temperature+Material*Temperature`, where each effect is given explicitely, or `Life~.+.^2`, where all factor included in the data frame are included, together with the second-order interaction(s).

Results are obtained using `summary(battery.aov)`, and are printed in Table 5.1. All three effects are significant, especially the Temperature effect which account for about 50% of the total variability in battery life.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Material	2	10683.72	5341.86	7.91	0.0020
Temperature	2	39118.72	19559.36	28.97	0.0000
Material:Temperature	4	9613.78	2403.44	3.56	0.0186
Residuals	27	18230.75	675.21		

Table 5.1: ANOVA table for the 3^2 battery experiment.

Most of the time, a plot of the averaged response variable will be very useful to gain insight into the effects displayed in the ANOVA table. In Figure 5.1, we have plotted the average Life $\bar{y}_{ij\cdot}$ as a function of Temperature, for each Material type. Each point in the graph is thus the mean of 4 observations. We call this an interaction plot.

```
with(battery, interaction.plot(Temperature,Material,Life,type="b",pch=19,
                    fixed=T,xlab="Temperature (°F)",ylab="Average life"))
```

It can be seen that average life decreases as temperature increases, with Material type 3 leading to extended battery life compared to the other, especially at higher temperature, hence the interaction effect.

Another useful plot is the effects plot, which can be obtained with `plot.design()` which takes as an argument the same formula as that passed to the `aov()` function. Thus,

```
plot.design(Life~Material*Temperature,data=battery)
```

gives the picture given in Figure 5.2a. The large Temperature effect is reflected in the range of battery life variation induced by its manipulation.

Now, we have to follow the same routes as in Chapter 3 and run multiple comparisons as well as check model adequacy. These are basically the same principles that what we described pp. 16 and 18, so we don't go further into details for this chapter. Note, however, that model checking should be done on each "treatment" (i.e. crossing each factor level together).

With such a design, Tukey's HSD are widely appreciated from researchers. Applying `TukeyHSD(battery.aov,which="Material")` gives the following results:

```
Tukey multiple comparisons of means
  95% family-wise confidence level
```

December 2, 2012
Draft Version

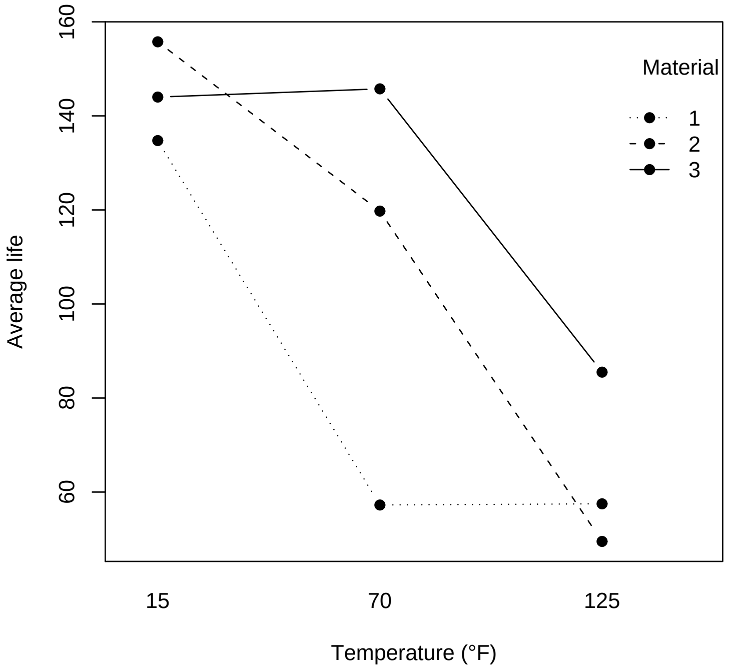


Figure 5.1: Material type–temperature plot for the battery life experiment.

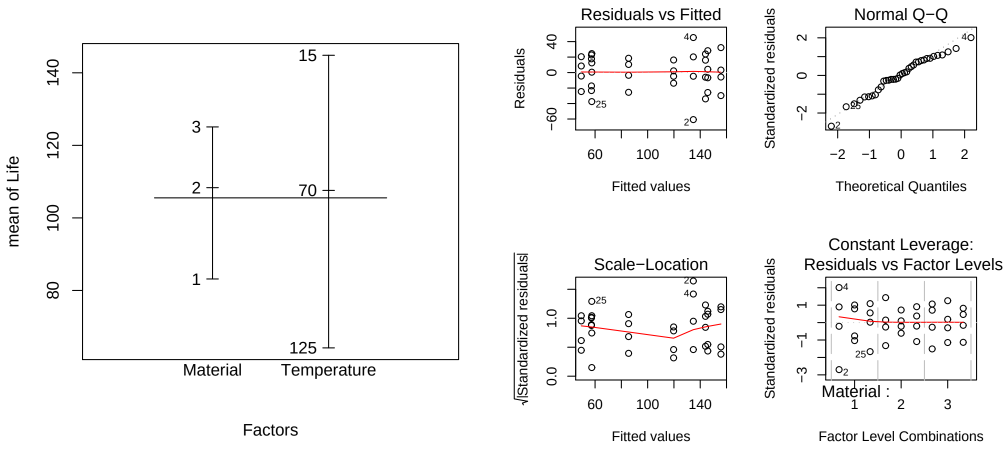
```
Fit: aov(formula = Life ~ . + .^2, data = battery)

$Material
        diff       lwr      upr     p adj
2-1 25.16667 -1.135677 51.46901 0.0627571
3-1 41.91667 15.614323 68.21901 0.0014162
3-2 16.75000 -9.552344 43.05234 0.2717815
```

But this not actually what we should compute because the interaction is significant. Thus the effect of Material depends on which level of Temperature is considered. If we decide to study the material effect at 70°F, we get a slightly comparable picture (I do it by hand as I cannot find a proper R way), but it the right way to compute means contrast in presence of a significant interaction.

```
  # we compute the three means at Temperature=70°F
  mm <- with(subset(battery, Temperature==70),
             aggregate(Life,list(M=Material),mean))
  # next the studentized t quantile times the error type (based on pooled SD
  # from ANOVA)
  val.crit <- qtukey(.95,3,27)*sqrt(unlist(summary(battery.aov))[["Mean Sq4"]]/4)
  # finally compare the observed difference of means with the critical value
```

December 2, 2012
Draft Version



(a) (b)

Figure 5.2: (a) Effect display. (b) Diagnostic plots.

```
diff.mm <- c(d.3.1=mm$x[3]-mm$x[1],d.3.2=mm$x[3]-mm$x[2],d.2.1=mm$x[2]-mm$x[1])
names(which(diff.mm > val.crit))
```

In conclusion, only Material type 3 vs. type 1 and Material type 2 vs. type 1 appear to be significantly different when Temperature is fixed at 70°F.

Model adequacy, or *residual analysis*, is shown in Figure 5.2b: This includes a plot of residuals or standardized residuals against fitted values, a Q-Q plot, and a plt of leverage and Cook's distance. For the two-factor factorial model, residuals are defined as $e_{ijk} = y_{ijk} - \hat{y}_{ijk}$. Since $\hat{y}_{ijk} = \bar{y}_{ij\cdot}$ (we average over observations in the ijth cell), the above equation is equivalent to

$$e_{ijk} = y_{ijk} - \bar{y}_{ij\cdot} \tag{5.4}$$

Examining the plot of residuals vs. fitted values, we can see that a larger variance is associated to larger fitted value, and two observations (2 and 4) are highlighted in Figure 5.2b (top left panel); in other words, the 15°F-material type 1 cell contains extreme residuals that account for the inequality of variance. This is easily seen using a command like `with(battery, tapply(Life,list(Material,Temperature),var))`, which gives

```
         15       70      125
1 2056.9167 556.9167 721.0000
2  656.2500 160.2500 371.0000
3  674.6667 508.2500 371.6667
```

We could also imagine using a Model without interaction, where appropriate. This resumes to removing the $(\tau\beta)_{ij}$ term in Model 5.1. Applied to the battery life data, `summary(battery.aov2 <- aov(Life~Material+Temperature,data=battery))` leads to the following results:

December 2, 2012
Draft Version

```
            Df Sum Sq Mean Sq F value    Pr(>F)
Material     2  10684    5342  5.9472  0.006515 **
Temperature  2  39119   19559 21.7759 1.239e-06 ***
Residuals   31  27845     898
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obviously, the two main effets are still highly significant. However, residual analysis of this “reduced” model (Fig. 5.3) shows that a no interaction model is not appropriate. In this figure, we plot the $\bar{y}_{ij\cdot}$ against fitted values for the no interaction model, $\hat{y}_{ijk} = \bar{y}_{i\cdot\cdot} + \bar{y}_{\cdot j\cdot} - \bar{y}_{\cdot\cdot\cdot}$. This can be viewed as the difference between the observed cell means and the estimated cell means assuming no interaction; any pattern in this plot is thus suggestive of the presence of an interaction.

```
mm2 <- with(battery, tapply(Life,list(Material,Temperature),mean))
mm2 <- as.vector(mm2)
plot(fitted(battery.aov2)[seq(1,36,by=4)],mm2-fitted(battery.aov2)[seq(1,36,by=4)],
     xlab="",ylab=expression(bar(y)[ij.]-hat(y)[ijk]),
     pch=19,axes=FALSE,ylim=c(-30,30))
axis(1,at=seq(0,200,by=50),pos=0)
text(155,4,expression(hat(y)[ijk]),pos=4)
axis(2,at=seq(-30,30,by=10),las=1)
```

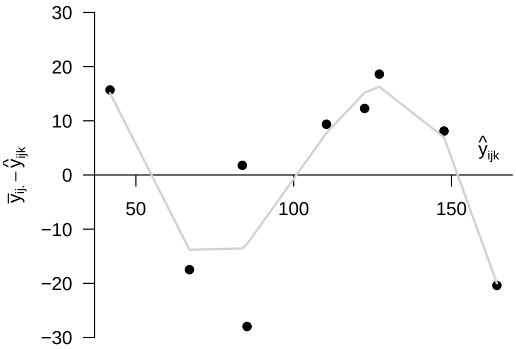


Figure 5.3: Plot of $\bar{y}_{ij\cdot}$ vs. $\hat{y}_{ijk}$.

There is a clear tendency toward alternated fitted values differences with increasing predicted values, $\bar{y}_{ij\cdot}$. To highlight that pattern, we can superimpose a loess line on the above plot.

```
yy <- order(fitted(battery.aov2)[seq(1,36,by=4)])
```

December 2, 2012
Draft Version

```
  xx.fit <- fitted(battery.aov2)[seq(1,36,by=4)]
  yy.fit <- mm2-fitted(battery.aov2)[seq(1,36,by=4)]
  lines(xx.fit[yy],predict(loess(yy.fit[yy]~xx.fit[yy])),col="lightgray",lwd=2)
```

Now, suppose that we have only a single replicate per combination of factor. If we turn back to Equation 5.2, we see that the error term is not estimable because it is now confounded with the interaction term. The corresponding expected mean square proved to be

$$\sigma^2 + \frac{\sum\sum(\tau\beta)^2_{ij}}{(a-1)(b-1)}.$$

As a consequence, no tests on main effects can be carried out unless the interaction effect is zero, i.e. $(\tau\beta)_{ij} = 0$. If, after removing this term, the effects model is correct, then the residual MS above is an unbiased estimator of σ^2.

There is a test for determining whether interaction is present or not. The Tukey's non-additivity, or curvature, test [?] consists in separating the residual SS into a single degree of freedom component due to non-additivity, or interaction, and a component for error with $(a-1)(b-1)-1$ degrees of freedom. An F test of $\frac{SS_N}{SS_E/[(a-1)(b-1)-1]}$, with 1 and $(a-1)(b-1)-1$ degrees of freedom, at the α level, allow to reject the null hypothesis of no interaction.

An example of this procedure is shown using `impurity.txt` which contains data from a study of a chemical product and its impurities resulting from two factors—pressure and temperature.

```
  impurity <- read.table("impurity.txt",header=TRUE)
  impurity$Temperature <- as.factor(impurity$Temperature)
  impurity$Pressure <- as.factor(impurity$Pressure)
```

Applying a full factorial model with interaction gives no tests for main effects as expected. The command used to display is rather simple:

```
summary(aov(N~Temperature*Pressure,impurity))
```

```
                     Df Sum Sq Mean Sq
Temperature           2 23.333  11.667
Pressure              4 11.600   2.900
Temperature:Pressure  8  2.000   0.250
```

The idea is thus to decompose the SS of the Temperature×Pressure term into two additional terms. Although we can do it ourselves, there is an interesting function in the `alr3` package which provides both a graphical assessment of the presence of a potential interaction as well as the formal Tukey's test. In fact, it amounts to be a test on a quadratic term added to the model $y \sim A + B$.

```
  library(alr3)
  residual.plots(lm(N~Temperature+Pressure,impurity))
```

As a result, we get a test statistic of 0.602 with a p-value of 0.547. Montgomery provides an F test, as described in the procedure above, which equals 0.36. This is exactly 0.602^2

December 2, 2012
Draft Version

because `residual.plots()` returns a t-test. As a side-effect, this function also a plot studentized residuals against fitted values, with the fitted quadratic term (dotted line), as shown in Figure 5.4 (lower left panel).

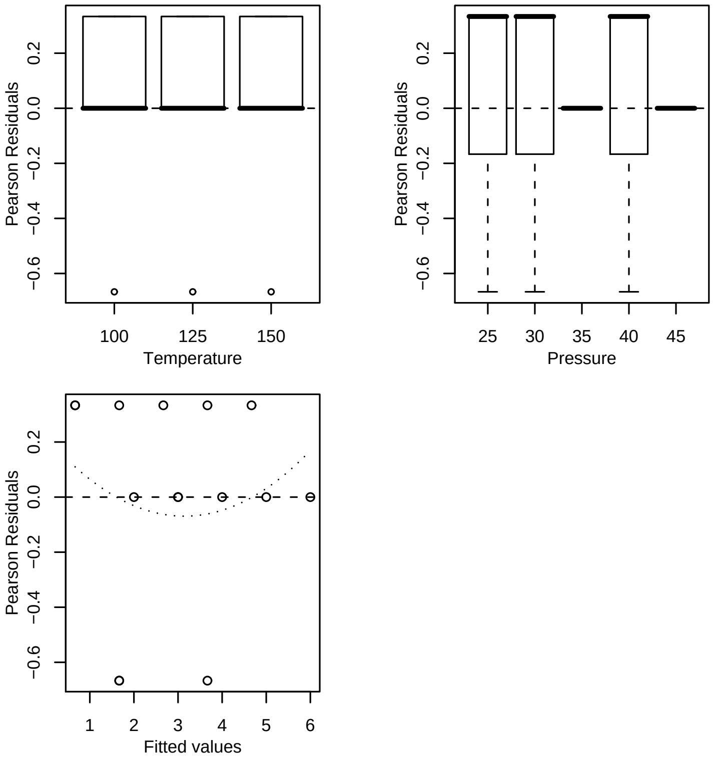


Figure 5.4: Curvature test for the impurity data.

5.3 General factorial design, response curves and surfaces

The model described in the preceding section can be generalized to any number of fixed effects, and there will be as much second order interaction terms as there are factors, plus third order interaction term(s).

As an example of a three-factors design, we can consider the data in the `bottling.txt` file. In this study, a soft drink bottler is interested in obtaining more uniform fill heights in the bottles produced by his manufacturing process. The process engineer can control three variables during the filling process: the percent carbonation (A), the oprating pressure in the filler (B), and the bottles produced per minute on the line speed (C). The factorial model can be written as

$$y \sim A + B + C + AB + AC + BC + ABC$$

where y is the response variable, i.e. the fill height deviation.

December 2, 2012
Draft Version

We happen to set up the data as follows:

```
bottling <- read.table("bottling.txt",header=TRUE,
                       colClasses=c("numeric",rep("factor",3)))
summary(bottling)
```

and brief summary plot is shown in Figure 5.5. Commands used for this display are rather simple, and we could probably do better.

```
opar <- par(mfrow=c(2,2),cex=.8)
boxplot(Deviation~.,data=bottling,las=2,cex.axis=.8,ylab="Deviation")
abline(h=0,lty=2)
par(las=1)
mm <- with(bottling, tapply(Deviation,Carbonation,mean))
ss <- with(bottling, tapply(Deviation,Carbonation,sd))
bp <- barplot(mm,xlab="Carbonation",ylab="Deviation",ylim=c(-2,9))
arrows(bp,mm-ss/sqrt(4),bp,mm+ss/sqrt(4),code=3,angle=90,length=.1)
with(bottling, interaction.plot(Carbonation,Pressure,Deviation,type="b"))
with(bottling, interaction.plot(Carbonation,Speed,Deviation,type="b"))
par(opar)
```

It is used here to highlight the different possible viewings of this highly structured dataset. Top left panel is a plot of each "cell", i.e. each combination of the three factors, with $n = 4$ observations per cell. This is rather uninformative as we are mainly interested in the effect of the factors themselves, as well as their potential interaction of the variation of the response variable. Top right panel presents aggregated data on B and C: The means observed, together with their standard errors, are plotted against the three level of Carbonation. Finally, the two lower panels are interaction plot, but only display two factors at the same time, thus we are still lacking a complete overview of the main and interaction effects.

The three-way ANOVA model is summarized in Table 5.2.

```
summary(bottling.aov <- aov(Deviation~.^3,bottling))
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Carbonation	2	252.75	126.37	178.41	0.0000
Pressure	1	45.37	45.37	64.06	0.0000
Speed	1	22.04	22.04	31.12	0.0001
Carbonation:Pressure	2	5.25	2.63	3.71	0.0558
Carbonation:Speed	2	0.58	0.29	0.41	0.6715
Pressure:Speed	1	1.04	1.04	1.47	0.2486
Carbonation:Pressure:Speed	2	1.08	0.54	0.76	0.4869
Residuals	12	8.50	0.71		

Table 5.2: Results of the saturated model for the bottling data.

December 2, 2012
Draft Version

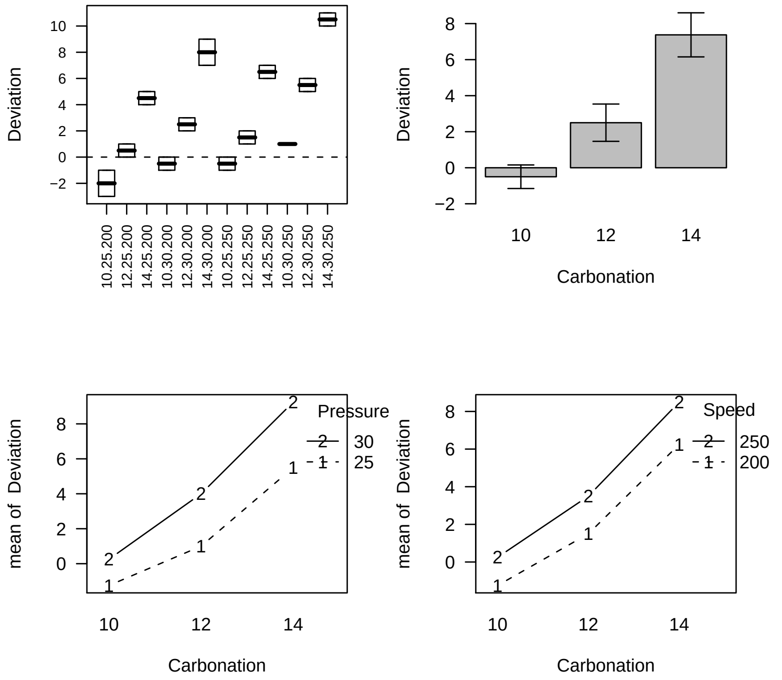


Figure 5.5: The bottling dataset.

As can be seen from the ANOVA table, main effects are all significant, while none of the four interaction effects are. Note, however, that the Carbonation×Pressure interaction is marginally significant but exceeds the conventional 5% significance level. Such results suggest that we may remove the interaction terms, which is done in the next step (Note that we could have used the `update()` command which allows to quickly update a given model, but in this case it is rather borrying to remove all interaction effects).

```
bottling.aov2 <- aov(Deviation~.,bottling)
anova(bottling.aov2,bottling.aov)
```

Table 5.3 show the new p-values associated with the F tests on the main effects only. The last R command in the above code allows to formally test for the benefit of including the interaction terms in the model. As the F-test is non-significant ($p = 0.225$), we could undoubetly remove these terms. We should have been able to suppose that no interaction are present by simply looking at the interaction plots shown in Figure 5.5. In each case, both line are about to be strictly paralell, which is implied by the absence of interaction in the factorial model.

Now, if we turn back to the battery life experiment, what if we include a quadratic term for the quantitative variable, Temperature?

```
battery$Temperature.num <- as.numeric(as.character(battery$Temperature))
```

December 2, 2012
Draft Version

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Carbonation	2	252.75	126.37	145.89	0.0000
Pressure	1	45.37	45.37	52.38	0.0000
Speed	1	22.04	22.04	25.45	0.0001
Residuals	19	16.46	0.87		

Table 5.3: Results of the reduced model for the bottling data, showing only significant main effects.

```
battery.aov3 <- aov(Life~Material+Temperature.num+I(Temperature.num^2)
                    +Material:Temperature.num+Material:I(Temperature.num^2),
                    data=battery)
summary(battery.aov3)
```

Note the use of `as.character` conversion to get a proper interpretation of numerical values of the Temperature factor. If instead we use directly `as.numeric(battery$Temperature)`, we would get a $1, 2, 3$ coding which is not what we really want, since we are planning to predict actual values of y (see below).

This time, the model can be written symbolically as

$$y \sim A + B + B^2 + AB + AB^2$$

with B denoting the Temperature factor. Summary of the results is shown in Table 5.4.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Material	2	10683.72	5341.86	7.91	0.0020
Temperature.num	1	39042.67	39042.67	57.82	0.0000
I(Temperature.num^2)	1	76.06	76.06	0.11	0.7398
Material:Temperature.num	2	2315.08	1157.54	1.71	0.1991
Material:I(Temperature.num^2)	2	7298.69	3649.35	5.40	0.0106
Residuals	27	18230.75	675.21		

Table 5.4: Fitting the battery life data with an additional quadratic effect of Temperature.

If we look at the predicted values for this model, the results shown in Figure 5.6 are more in agreement with the intuitive idea that there is an optimal Temperature that depends of Material type (cf. the significant AB^2 interaction effect in Table 5.4), and for which battery life reaches its maximum.

```
new <- data.frame(Temperature.num=rep(seq(15,125,by=5),3),Material=gl(3,23))
new$fit <- predict(battery.aov3,new)
opar <- par(las=1)
# we first plot the fitted values
```

December 2, 2012
Draft Version

```
with(new, interaction.plot(Temperature.num,Material,fit,legend=FALSE,
                           xlab="Temperature",ylab="Life",ylim=c(20,190)))
txt.leg <- paste("Material type",1:3)
text(5:7,new$fit[new$Temperature.num==c(45,55,65)]-c(3,3,-20),txt.leg,pos=1)
# next the observed values
points(rep(c(1,15,23),each=12),battery$Life,pch=19)
par(opar)
```

Note that to get a smooth curve, we have to interpolate the predictions over all the domain of temperature, when it is considered as a numeric factor. This is done using the statement

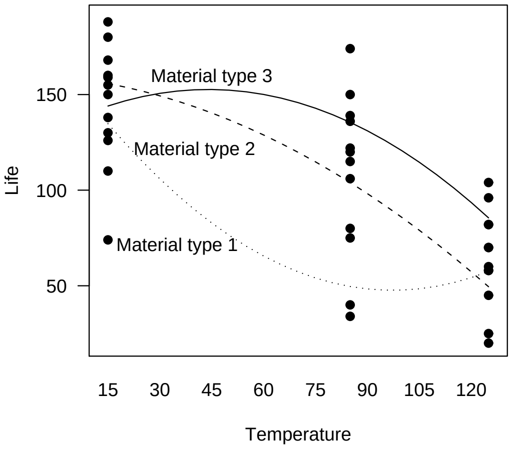


Figure 5.6: Model fit for the battery life experiment, including non-linear effect of temperature.

When more than one factor are quantitative, we may use a response surface to model the relationship between the response variable and the design factor. This is illustrated using the data contained in `toollife.txt`, which describes results gathered through a study of the effect of cutting speed (A) and tool angle (B) on the effective life of a cutting tool. Model fitted to the data is of the following form

$$y \sim A + B + A^2 + B^2 + AB^2 + A^2B + AB$$

We are mainly interested in some of the quadratic terms, both quadratic and linear effects are crossed in the interaction terms, to preserve the *hierarchical principle* (inclusion of high-order term should be followed by inclusion of the lower-order terms that compose it), as in the preceding example.

```
tool <- read.table("toollife.txt",header=TRUE)
tool.lm <- lm(Life~Angle*Speed+I(Angle^2)*I(Speed^2)+Angle:I(Speed^2)
```

December 2, 2012
Draft Version

```
              +I(Angle^2):Speed,tool)
tmp.angle <- seq(15,25,by=.1)
tmp.speed <- seq(125,175,by=.5)
tmp <- list(Angle=tmp.angle,Speed=tmp.speed)
new <- expand.grid(tmp)
new$fit <- c(predict(tool.lm,new))
```

Partial output is reproduced below. Note that these are sequential sum of squares so that they are orthogonal (because of the structure of the 3^2 design itself).

```
Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)           -1.068e+03  7.022e+02  -1.521   0.1626
Angle                  1.363e+02  7.261e+01   1.877   0.0932 .
Speed                  1.448e+01  9.503e+00   1.524   0.1619
I(Angle^2)            -4.080e+00  1.810e+00  -2.254   0.0507 .
I(Speed^2)            -4.960e-02  3.164e-02  -1.568   0.1514
Angle:Speed           -1.864e+00  9.827e-01  -1.897   0.0903 .
I(Angle^2):I(Speed^2) -1.920e-04  8.158e-05  -2.353   0.0431 *
Angle:I(Speed^2)       6.400e-03  3.272e-03   1.956   0.0822 .
Speed:I(Angle^2)       5.600e-02  2.450e-02   2.285   0.0481 *
```

Rather than interpreting these results, we are mainly interested in the way we can convey this information in an informative graphical display. To this end, we may plot the fitted values against both continuous predictor values, after interpolation. This is called a contour plot. Figure 5.7 has been produced using the function `contourplot()` of the `lattice` package.

```
require(lattice)
contourplot(fit~Angle*Speed,data=new,cuts=8,region=T,col.regions=gray(7:16/16))
```

5.4 Blocking in a factorial design

Until now, we have only discussed factorial designs for completely randomized experiment. As already discussed in Chapter 4, such approach have to be avoided when possible nuisance factors that can't be controlled are present. The idea is now to run a single replicate of a complete factorial experiment within each block. This leads to the model

$$y_{ijk} = \mu + \tau_i + \beta_{ij} + (\tau\beta)_{ij} + \delta_k + \varepsilon_{ijk} \tag{5.5}$$

with the same notation as those of Model 5.1. Now, we have included an effect for the kth block, but we assume that interaction between blocks and treatment is negligible (as in the RCBD). Interaction terms, $(\tau\delta)_{ik}$, $(\beta\delta)_{jk}$, and $(\tau\beta\delta)_{ijk}$, are thus confounded within the error term.

The `intensity.txt` file contains data on an experiment aiming at improvinf the ability to detect targets on a radar scope. Twho factors were considered, namely the amount of background noise ("ground clutter") on the scope and the type of filter placed

December 2, 2012
Draft Version

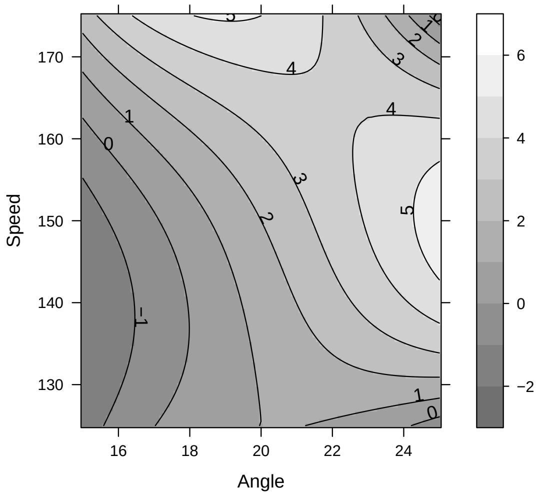


Figure 5.7: Contour plot for the cutting tool study.

over the screen. They are considered as fixed effects. Because of operator availability and varying degree of knowledge, it is convenient to select an operator and keep him at the scope until all the necessary runs have been made. They will be considered as blocks. We have thus 3 × 2 treatment combinations arranged in a randomized complete block. Data are summarized in Figure 5.8.

```
intensity <- read.table("intensity.txt",header=TRUE,
                        colClasses=c("numeric",rep("factor",3)))
require(lattice)
xyplot(Intensity~Ground|Operator,data=intensity,groups=Filter,
       panel=function(x,y,...){
         subs <- list(...)$subscripts
         panel.xyplot(x,y,pch=c(1,19),...)
         panel.superpose(x,y,panel.groups="panel.lmline",lty=c(1,2),...)
       },key=list(text=list(lab=as.character(1:2)),lines=list(lty=1:2,col=1),
       corner=c(1,.95),title="Filter Type",cex.title=.8),col=1)
```

We also fit an OLS line separetely for each Filter type and each operator, to better appreciate visually the pattern of response. As can be seen, there is a notable variability between individual slopes and intercept. Blocking is thus a good way to uncover operator's intrinsic effect.

To apply the ANOVA model, we basically follow the steps explained in Chapter 4 and specify blocks as a separate strata using the `Error()` option.

```
intensity.aov <- aov(Intensity~Ground*Filter+Error(Operator),intensity)
summary(intensity.aov)
```

In Table 5.5, we see that R does not display the Blocks effects when we use the `summary()`

December 2, 2012
Draft Version

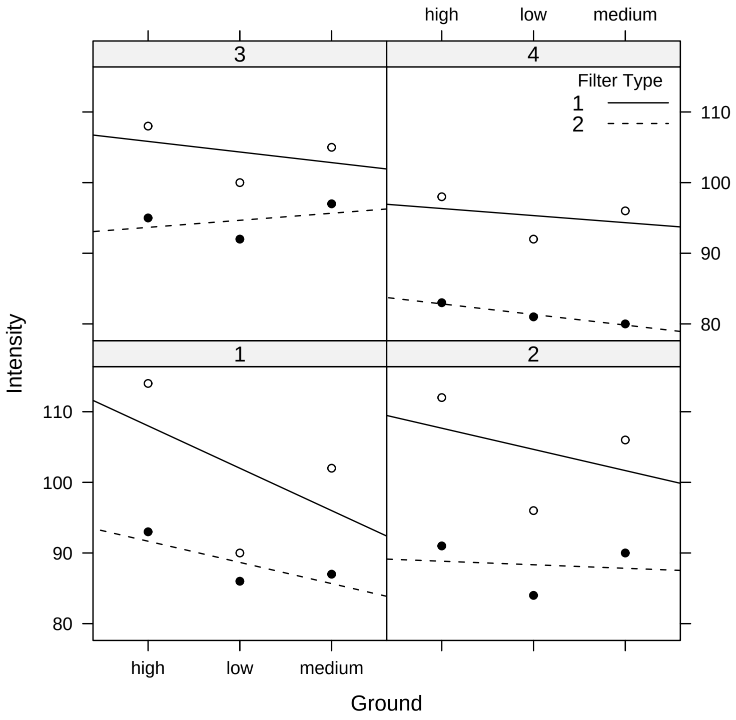


Figure 5.8: The intensity data.

command. To obtain SS specific to Blocks, we rather call directly the `aov` object, like `intensity.aov` at the R command prompt.

```
Call:
aov(formula = Intensity ~ Ground * Filter + Error(Operator),
    data = intensity)

Grand Mean: 94.91667

Stratum 1: Operator

Terms:
                Residuals
Sum of Squares   402.1667
Deg. of Freedom         3

Residual standard error: 11.57824

(...)
```

We discarded the rest of the output which contains stratum 2 SS already included in Table 5.5. What should be noted is that the blocking factor SS is rather large compared

December 2, 2012
Draft Version

to other main effects SS (e.g., Ground, 335.58) or error term (166.33). This confirms our intuitive idea based on the inspection of Figure 5.8 that there are large inter-individual variation with respect to the response variable. Computing SS for the blocking factor follows from the above formulation (Equation 5.5) and it can be shown that

$$SS_{\text{blocks}} = \frac{1}{ab}\sum_{k=1}^{n} y_{\cdot\cdot k}^2 - \frac{y_{\cdots}^2}{abn} \tag{5.6}$$

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Residuals	3	402.17	134.06		
Ground	2	335.58	167.79	15.13	0.0003
Filter	1	1066.67	1066.67	96.19	0.0000
Ground:Filter	2	77.08	38.54	3.48	0.0575
Residuals	15	166.33	11.09		

Table 5.5: Results of the ANOVA model applied to the intensity data.

December 2, 2012
Draft Version

Chapter 6

The 2^k Factorial Design

6.1 Summary of Chapter 6

6.2 The 2^2 design

The 2^2 design is the simpler design that belong to the general family of the 2^k design. We consider two factors, A and B, with two levels each which can be tought of as "low" and "high" levels. The experiment may be replicated a number of times, say k, and in this case we have $2 \times 2 \times k$ trials or runs, yielding a *completely randomized experiment*.

Suppose that we are interesting in investigating the effect of the concentration of the reactant and the amount of the catalyst on the conversion (yield) in a chemical process, with three replicates. The objective is to study how reactant concentration (15 or 25%) and the catalyst (1 or 2 pounds) impact yield (`yield.txt`). Results for the different treatment combinations of the above experiment are summarized in Figure 6.2, where a "+" sign means the high level and a "−" sign means the corresponding low level.

The average effect of a factor is defined as the change in response produced by a change in the level of that factor averaged over the levels of the other factors. In the preceding figure, the symbols (1), a, b, and ab represent the total of all n replicates taken at the treatment combination. The effect of A at the low level of B is then defined as $[a-(1)]/n$, and the effect of A at the high level of B as $[ab-b]/n$. The average of these two quantities yields the *main effect* of A:

$$\begin{aligned} A &= \frac{1}{2n}\big\{[ab-b]+[a-(1)]\big\} \\ &= \frac{1}{2n}[ab+a-b-(1)]. \end{aligned} \tag{6.1}$$

Likewise, for B, we have:

$$\begin{aligned} B &= \frac{1}{2n}\big\{[ab-a]+[b-(1)]\big\} \\ &= \frac{1}{2n}[ab+b-a-(1)]; \end{aligned} \tag{6.2}$$

December 2, 2012
Draft Version

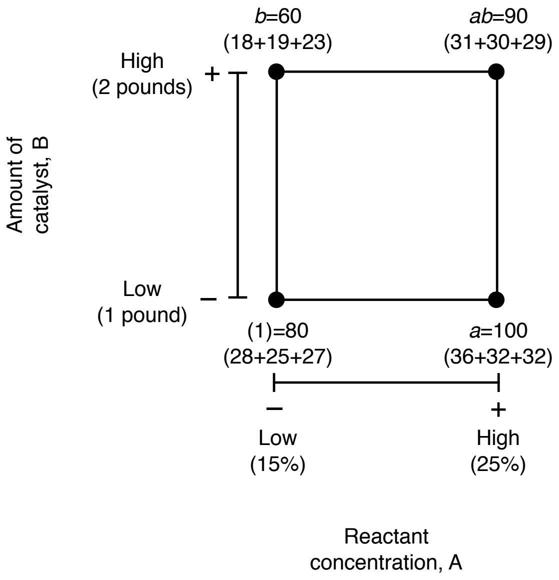


Figure 6.1: Treatment combinations in the 2^2 design.

and we define the *interaction effect* AB as the average difference between the effect of A at the high level of B and the effect of A at the low level of B:

$$\begin{aligned} AB &= \frac{1}{2n}\Big\{[ab-b]-[a-(1)]\Big\} \\ &= \frac{1}{2n}[ab+(1)-a-b]. \end{aligned} \tag{6.3}$$

As an alternative, one may consider that the effect of A can be computed as

$$\begin{aligned} A &= \bar{y}_{A^+}-\bar{y}_{A^-} \\ &= \frac{ab+a}{2n}-\frac{b+(1)}{2n} \\ &= \frac{1}{2n}[ab+a-b-(1)], \end{aligned} \tag{6.4}$$

which is exactly the results of 6.1. The same applies for the computation of the effect of B and AB. Numerically, we have:

$$A=\frac{1}{2\times 3}(90+100-60-80)=8.33,$$

or using R:

```
yield <- read.table("yield.txt",header=T)
attach(yield)
rm(yield)
yield.sums <- aggregate(yield,list(reactant=reactant,catalyst=catalyst),sum)
```

December 2, 2012
Draft Version

which gives all necessary information:

```
  reactant catalyst   x
1     high     high  90
2      low     high  60
3     high      low 100
4      low      low  80
```

It should be noted that the effect of A is positive which suggests that increasing A from the low level (15%) to the high level (25%) will increase the yield. The reverse would apply for B, whereas the interaction effect appears rather limited.

The ANOVA Model and contrasts formulation. An analysis of variance will help to estimate the direction and magnitude of the factor effects. We already showed that a contrast is used when estimating A, namely $ab + a - b - (1)$ (Eq. 6.1). This contrast is called the *total effect* of A. All three contrasts derived above are *orthogonal*. As the sum of squares for any contrast is equal to the contrast squared and divided by the number of observations in each total in the contrast times the SS of the contrast coefficients (Chapter 3), we have:

$$\begin{cases} SS_A & = \frac{[ab+a-b-(1)]^2}{4n} \\ SS_B & = \frac{[ab+b-a-(1)]^2}{4n} \\ SS_{AB} & = \frac{[ab+(1)-a-b]^2}{4n} \end{cases} \tag{6.5}$$

The total SS has $4n-1$ degrees of freedom and it is found in the usual way, that is

$$SS_T \sum_{i=1}^{2} \sum_{j=1}^{2} \sum_{k=1}^{2} y_{ijk}^2 - \frac{y_{\dots}^2}{4n}, \tag{6.6}$$

whereas the error sum of squares, with $4(n-1)$ degrees of freedom, is $SS_E = SS_T - SS_A - SS_B - SS_{AB}$.

The treatment combinations may be written as

Effects	(1)	a	b	ab
A	-1	$+1$	-1	$+1$
B	-1	-1	$+1$	$+1$
AB	$+1$	-1	-1	$+1$

and this order is refered to as Yates's order . Since all contrasts are orthogonal, the 2^2 (and all 2^k designs) is an *orthogonal design.*

```
summary(aov(yield~reactant*catalyst))
```

From Table 6.1, we can verify that both main effects are significant but the interaction term, AB, is not.

Frank Yates (1902–1994) worked on sample survey design and analysis. He is also the author of a book on the design and analysis of factorial experiments.

December 2, 2012
Draft Version

Table 6.1: Analysis of variance table for the "yield" experiment.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
reactant	1	208.33	208.33	53.19	0.0001
catalyst	1	75.00	75.00	19.15	0.0024
reactant:catalyst	1	8.33	8.33	2.13	0.1828
Residuals	8	31.33	3.92		

The Regression Model. The coefficients estimated from a regression model (see below) yield the following prediction equation:

$$\hat{y} = 18.333 + 0.833 x_{\text{reactant}} - 5.000 x_{\text{catalyst}},$$

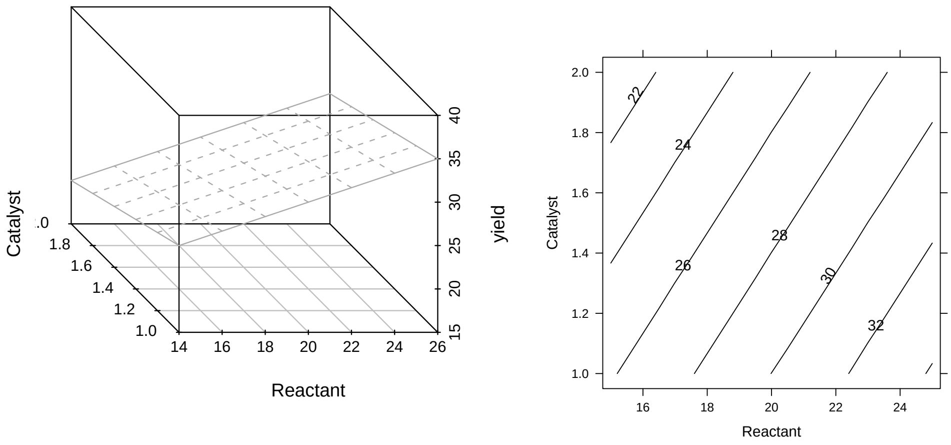
where x_{reactant} and x_{catalyst} refer to the values taken by the two factors. Here, factors levels are treated with their corresponding numerical values (1/2 for catalyst, 15/25 for reactant), but the ANOVA table would remain the same whatever the values we assign to their levels. However, the model parameters depend on the unit of measurement. In the next R script we convert the binary variables to ordinal variables, with adequate values. Note that the somewhat tricky manipulation ensures that the level are correctly mapped to their numeric value.

```
reactant.num <- reactant
levels(reactant.num) <- c(25,15)
reactant.num <- as.numeric(as.character(reactant.num))
catalyst.num <- catalyst
levels(catalyst.num) <- c(2,1)
catalyst.num <- as.numeric(as.character(catalyst.num))
yield.lm <- lm(yield~reactant.num+catalyst.num)
yield.lm  ## gives the coefficients of the LM
```

Figure 6.2 shows the response surface and a contour plot of the fitted response as a function of reactant and catalyst values. Because we deal with a *first-order model*, which includes only the main effects, the fitted response surface is a plane. Such a response surface is used to find a direction of potential improvement for a process, although as will be seen in Chapter 11, the method of *steepest ascent* constitutes a more formal way to do so.

```
s3d <- scatterplot3d(reactant.num,catalyst.num,yield,type="n",
                     angle=135,scale.y=1,xlab="Reactant",ylab="Catalyst")
s3d$plane3d(yield.lm,lty.box="solid",col="darkgray")
tmp <- list(reactant.num=seq(15,25,by=.5),catalyst.num=seq(1,2,by=.1))
new.data <- expand.grid(tmp)
new.data$fit <- predict(yield.lm,new.data)
contourplot(fit~reactant.num+catalyst.num,new.data,xlab="Reactant",
            ylab="Catalyst")
```

December 2, 2012
Draft Version



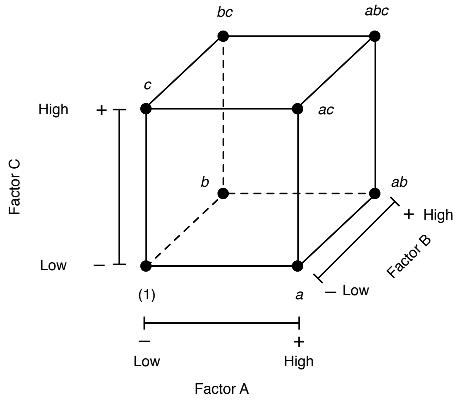
(a) Response surface.

(b) Contour plot.

Figure 6.2: Response surface plot for the yield data.

6.3 The 2^3 design

If we now consider three factors, A, B, and C, the design is called a 2^3 factorial design. There are then eight treatment combinations that can be displayed as a cube (Figure 6.3) and are refered to as the *design matrix*. There are seven degrees of freedom between the eight treatment combinations in the 2^3 design: Three DF are associated with each main effect, four DF are associated with interaction terms (three second-order interactions and one third-order).



(a) Geometric view

	Factor		
Run	A	B	C
1	−	−	−
2	+	−	−
3	−	+	−
4	+	+	−
5	−	−	+
6	+	−	+
7	−	+	+
8	+	+	+

(b) Design matrix

Figure 6.3: Treatment combinations in the 2^3 design.

December 2, 2012
Draft Version

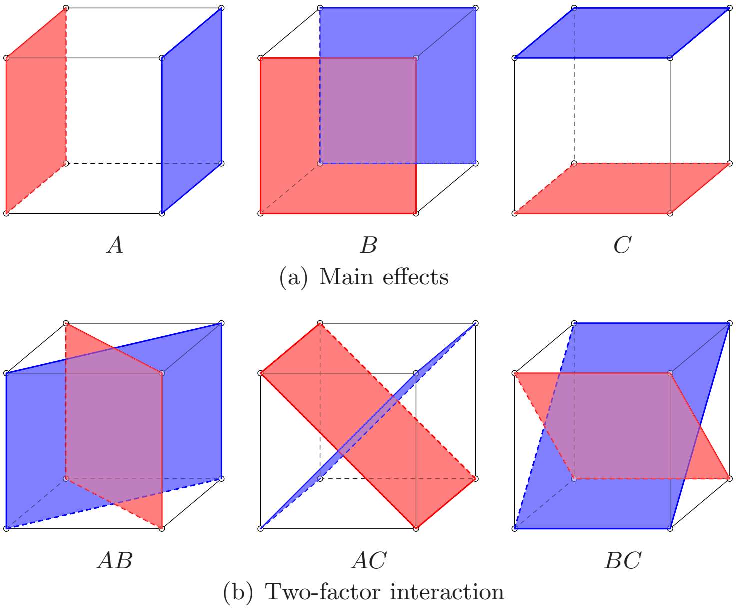
The effect of A when B and C are at the low level is $[a-(1)]/n$. When B is at the high level and C at its low level, it is $[ab-b]/n$. The effect of A when C is at the high level and B at the low level is $[ac-c]/n$. Finally, when both B and C are at the high level, the effect of A is $[abc-bc]/n$. Thus, the average effect of A is:

$$A = \frac{1}{4n}\big[a-(1)+ab-b+ac-c+abc-bc\big]. \tag{6.7}$$

This can found as a contrast between the four treatment combinations in the right face of the cube in Figure 6.4: The effect of A is simply the average of the four runs where A is at the high level ($\bar{y}_{A^+}$) minus the average of the four runs where A is at the low level ($\bar{y}_{A^-}$), or

$$\begin{aligned} A &= \bar{y}_{A^+} - \bar{y}_{A^-} \\ &= \frac{a+ab+ac+abc}{4n} - \frac{(1)+b+c+bc}{4n}. \end{aligned} \tag{6.8}$$

The effect for B and C are computed in a similar manner.



(a) Main effects

(b) Two-factor interaction

Figure 6.4: Geometric presentation of contrasts. In each case, high levels are highlighted in blue, low levels in red.

The two-factor interaction effects are also computed easily since they reflect the difference between the average of the effects of one factor at the two levels of the other factor. By convention, one-half of this difference is called e.g. the AB interaction:

$$AB = \frac{\big[abc-bc+ab-b-ac+c-a+(1)\big]}{4n} \tag{6.9}$$

December 2, 2012
Draft Version

or, equivalently,

$$AB = \frac{\big[abc + ab + c + (1)\big]}{4n} - \frac{\big[bc + b + ac + a\big]}{4n}. \tag{6.10}$$

Finally, the ABC interaction is defined as the average difference between the AB interactions for the two different levels of C, that is

$$\begin{aligned} ABC &= \frac{1}{4n}\Big\{[abc - bc] - [ac - c] - [ab - b] + [a - (1)]\Big\} \\ &= \frac{1}{4n}\big[abc - bc - ac + c - ab + b + a - (1)\big]. \end{aligned} \tag{6.11}$$

Again, all preceding equations reflect the contrast associated with the estimation of each effect. From these contrasts, we can expand a table of "+" (high level) and "−" (low level) as shown in Table 6.2. Once the signs for the main effects have been established, the remaining effects may be obtained by multiplying the appropriate columns, row by row. Table 6.2 has a number of interesting properties: (1) Except for column I, every column has an equal number of plus and minus signs; (2) The sum of the products of the signs in any two columns is zero (due to orthogonality); (3) Column I multiplied times any column leaves that column unchanged because I is an *identity element*; (4) The product of any two columns yields a column in the table. For example, $A \times B = AB$ and $AB \times B = AB^2 = A$.

Finally, sum of squares for the effects are now simply defined as

$$SS = \frac{(\text{Contrast})^2}{8n}.$$

Table 6.2: Algebric signs for calculating effects in the 2^3 design.

Treatment	**Factorial Effect**							
combination	I	A	B	AB	C	AC	BC	ABC
(1)	+	−	−	+	−	+	+	−
a	+	−	−	+	−	+	+	−
b	+	−	−	+	−	+	+	−
ab	+	−	−	+	−	+	+	−
c	+	−	−	+	−	+	+	−
ac	+	−	−	+	−	+	+	−
bc	+	−	−	+	−	+	+	−
abc	+	−	−	+	−	+	+	−

The "plasma etch" experiment (`plasma.txt`) is a 2^3 design used to develop a nitride etch process on a single-wafer plasma etching tool. The design factor are the gap between the electrodes, the gas flow (C_2F_6 is used a the reactant gas), and the RF power applied to the cathode. Each factor is run at two levels, and the design is replicated twice.

The data file may be loaded in the usual way:

December 2, 2012
Draft Version

```
plasma <- read.table("plasma.txt",header=TRUE)
plasma
```

Data are shown below, before transforming them in an appropriate R `data.frame`.

```
  Run  A  B  C   R1   R2
1   1 -1 -1 -1  550  604
2   2  1 -1 -1  669  650
3   3 -1  1 -1  633  601
4   4  1  1 -1  642  635
5   5 -1 -1  1 1037 1052
6   6  1 -1  1  749  868
7   7 -1  1  1 1075 1063
8   8  1  1  1  729  860
```

Before applying the ANOVA model, we need to switch to R "long" format, as proposed below:

```
plasma.df <- data.frame(etch=c(plasma$R1,plasma$R2),
                        rbind(plasma[,2:4],plasma[,2:4]))
plasma.df[,2:4] <- lapply(plasma.df[,2:4],factor)
```

Next, we simply use the `aov()` function to estimate all main effects and all interactions (Table 6.3):

```
plasma.df.aov <- aov(etch~A*B*C, data=plasma.df)
summary(plasma.df.aov)
```

Table 6.3 shows that the effect of A, C and their two-way interaction are significant. None of the other effects appears to be significant at the usual 5% level. The contribution of each effect may easily computed from their respective sum of squares.

Table 6.3: ANOVA table for the "plasma etch" experiment.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
A	1	41310.56	41310.56	18.34	0.0027
B	1	217.56	217.56	0.10	0.7639
C	1	374850.06	374850.06	166.41	0.0000
A:B	1	2475.06	2475.06	1.10	0.3252
A:C	1	94402.56	94402.56	41.91	0.0002
B:C	1	18.06	18.06	0.01	0.9308
A:B:C	1	126.56	126.56	0.06	0.8186
Residuals	8	18020.50	2252.56		

To compute the response surface, we need to recode the factor with their corresponding numerical values, which are given below:

December 2, 2012
Draft Version

	Low (-1)	High $(+1)$
A (Gap, cm)	0.80	1.20
B (C_2F_6 flow, SCCM)	125	200
C (Power, W)	275	325

This is easily done using, e.g.:

```
plasma.num.df <- plasma.df
levels(plasma.num.df$A) <- c(0.8,1.2)
levels(plasma.num.df$C) <- c(275,325)
plasma.num.df$A <- as.numeric(as.character(plasma.num.df$A))
plasma.num.df$C <- as.numeric(as.character(plasma.num.df$C))
plasma.num.df.lm <- lm(etch~A*C, plasma.num.df)
```

Next, we proceeded as before and store in a new `data.frame` values for which we want to compute predictions from the above regression model.

```
tmp <- list(C=seq(275,325,by=1),A=seq(0.8,1.2,by=.1))
new.data <- expand.grid(tmp)
new.data$fit <- predict(plasma.num.df.lm, new.data)
require(scatterplot3d)
s3d <- scatterplot3d(plasma.num.df$A,plasma.num.df$C,
                     plasma.num.df$etch,type="n",
                     angle=135,scale.y=1,xlab="Gap",ylab="Power")
s3d$plane3d(plasma.num.df.lm,lty.box="solid",col="darkgray")
```

December 2, 2012
Draft Version

Chapter 7

Blocking and Confounding in the 2^k Factorial Design

December 2, 2012
Draft Version

Chapter 8

Two-Level Fractional Factorial Designs

8.1 Summary of Chapter 8

Chapter 8 is another extension of the 2^k factorial design where ones aims at limiting the number of runs needed to study a large number of two-level factors. For example, a complete replicate of a 2^6 design requires 64 runs, where only 6 out of the 63 degrees of freedom correspond to main effects. If we are willing to assume that certain high-order interaction terms are negligible, we can run only a fraction of a complete factorial design. This kind of design is maily used in *screening experiments* where lot of factors are of interest and we want to determine which ones to include in a future study because of their larger effects.

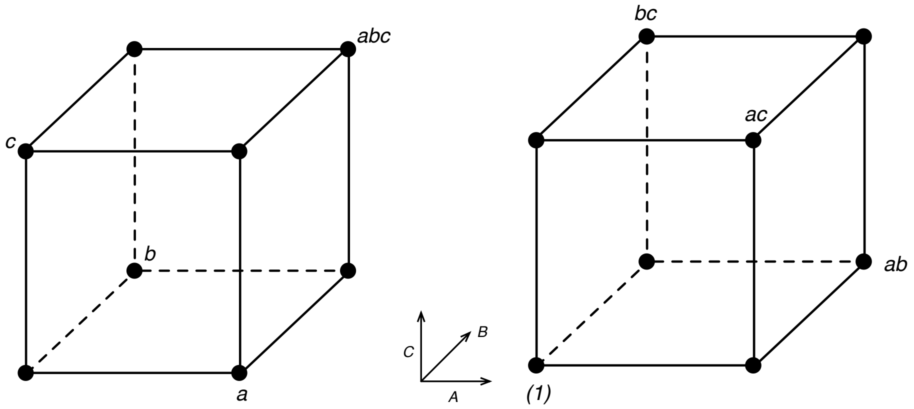
8.2 The one-half fraction of the 2^k design

Let's say we want to study 3 factors but we cannot afford to runs all 8 treatment combinations. Using one-half fraction, we now have $2^{3-1} = 4$ treatment combinations, e.g. a, b, c, and abc. Practically, in Table 8.1 we select only treatment combinations for which the generator ABC has a plus sign.

December 2, 2012
Draft Version

Table 8.1: The 2^3 factorial design. The 2^{3-1} is formed by considering only the upper part of this design.

Treatment	**Factorial Effect**							
combination	I	A	B	C	AB	AC	BC	ABC
a	+	+	−	−	−	−	+	+
b	+	−	+	−	−	+	−	+
c	+	−	−	+	+	−	−	+
abc	+	+	+	+	+	+	+	+
ab	+	+	+	−	+	−	−	−
ac	+	+	−	+	−	+	−	−
bc	+	−	+	+	−	−	+	−
(1)	+	−	−	−	+	+	+	−



(a) Principal fraction, $I = +ABC$.

(b) Alternate fraction, $I = -ABC$.

Figure 8.1: The two one-half fractions of the 2^3 design.

December 2, 2012
Draft Version

Chapter 9

Three-Level and Mixed-Level Factorial and Fractional Factorial Designs

December 2, 2012
Draft Version

Chapter 10

Fitting Regression Models

10.1 Summary of Chapter 10

Chapter 10 offers a gentle introduction to regression analysis in simple designs. The multiple linear regression model is detailed in its scalar and matrix form, and it is shown how unbiased estimator for the regression coefficients and the residual variance might be obtained when solving the least squares normal equations. Some illustrations are provided to make the connection with the analysis of designed experiments. Finally, model diagnostics and the assessment of goodness of fit are presented.

10.2 Linear Regression Models

When we are facing a situation with one *dependent variable* or *response* y that depends on k *independent* or *regression variables*, then we can postulate a given relationship relating these variables, though in most cases the true functional relationship between y and $x_1, x_2, \ldots, x_k$ is unknown. The idea is to fit an empirical model, that is a linear regression model in this particular settingwhich might take the form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon \tag{10.1}$$

if we consider two regressors, x_1 and x_2. The β's are called regression coefficients, and the model describes an hyperplane in a k dimensional space (here, $k = 2$). We could also consider adding an interaction term between x_1 and x_2 so that the previous model becomes

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2 + \varepsilon.$$

A second-order response surface might be represented as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \varepsilon;$$

this model is still linear in its parameters (after appropriate substitution).

December 2, 2012
Draft Version

10.2.1 Parameters estimation

The method of least squares (OLS) is generally used to estimate the regression coefficients. Let assume that the errors are gaussian i.i.d. such $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}(\varepsilon) = \sigma^2$. The previous model equation 10.1 can be rewritten as:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \varepsilon_i \\ &= \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} + \varepsilon_i, \quad i = 1, 2, \ldots, n \end{aligned} \tag{10.2}$$

The OLS solution minimizes the sum of the squared errors, that is

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j x_{ij} \right)^2 . \tag{10.3}$$

L has to be minimized with respect to $\beta_0, \beta_1, \ldots, \beta_k$, which yields the so-called least squares *normal equations*. Using matrix notation, Equation 10.2 may be written as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

where

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \text{ and } \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

We want to find the vector $\hat{\boldsymbol{\beta}}$ that minimizes

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \boldsymbol{\varepsilon}'\boldsymbol{\varepsilon} = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}),$$

which can also be expressed as

$$\begin{aligned} L &= \boldsymbol{y}'\boldsymbol{y} - \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y} - \boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta} \\ &= \boldsymbol{y}'\boldsymbol{y} - 2\boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}. \end{aligned} \tag{10.4}$$

The OLS estimators must satisfy

$$\left. \frac{\partial L}{\partial \boldsymbol{\beta}} \right|_{\hat{\beta}} = -2\boldsymbol{X}'\boldsymbol{y} + 2\boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = 0$$

which simplifies to $\boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}'\boldsymbol{y}$. This is the matrix form of the aforementioned normal equations, and the least squares estimator of $\boldsymbol{\beta}$ is

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y} \tag{10.5}$$

December 2, 2012
Draft Version

and the normal equations read:

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_{i1} & \sum_{i=1}^{n} x_{i2} & \cdots & \sum_{i=1}^{n} x_{ik} \\ \sum_{i=1}^{n} x_{i1} & \sum_{i=1}^{n} x_{i1}^2 & \sum_{i=1}^{n} x_{i1}x_{i2} & \cdots & \sum_{i=1}^{n} x_{i1}x_{ik} \\ \vdots & \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n} x_{ik} & \sum_{i=1}^{n} x_{ik}x_{i1} & \sum_{i=1}^{n} x_{ik}x_{i2} & \cdots & \sum_{i=1}^{n} x_{ik}^2 \end{bmatrix} \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_{i1}y_i \\ \vdots \\ \sum_{i=1}^{n} x_{ik}y_i \end{bmatrix}.$$

As can be seen, the diagonal elements of $\boldsymbol{X'X}$ are the sum of squares of the elements in $\boldsymbol{X}$.

The fitted regression model is

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}}, \tag{10.6}$$

and the $(n \times 1)$ vector of residuals, which reflects the difference between the actual observation y_i and the fitted value $\hat{y}_i$ is denoted by

$$\boldsymbol{e} = \boldsymbol{y} - \hat{\boldsymbol{y}}. \tag{10.7}$$

To estimate σ^2, we use the SS of the residuals, $\mathrm{SS}_E = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} e_i^2 = \boldsymbol{e'e}$. After some re-expressions, we obtain the following *error* or *residual sum of squares* with $n - p$ degrees of freedom

$$\mathrm{SS}_E = \boldsymbol{y'y} - \hat{\boldsymbol{\beta}}'\boldsymbol{X'y}. \tag{10.8}$$

It can be shown that $\mathbb{E}(\mathrm{SS}_E) = \sigma^2(n-p)$, so that an unbiased estimator of σ^2 is

$$\hat{\sigma}^2 = \frac{\mathrm{SS}_E}{n-p}. \tag{10.9}$$

10.2.2 A simple multiple regression example

Let's consider the measurement of the viscosity of a polymer and two process variables– Temperature (°C) and Catalyst feed rate (lb/h), that will be modeled as $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon$.

```
visc <- read.table("viscosity.txt", header=TRUE)
lm.fit <- lm(Viscosity ~ Temperature+Catalyst, data=visc)
summary(lm.fit)
```

From the commands above, we see that the OLS fit is $\hat{y} = 1566.08 + 7.62x_1 + 8.58x_2$, where x_1 stands for the Temperature and x_2 is the Catalyst feed rate.

Diagnostic or residual plots are useful to check that the residuals do not depend on the fitted values $\hat{y}$. In R,

December 2, 2012
Draft Version

```
op <- par(mfrow=c(2,2), las=1)
plot(lm.fit)
par(op)
```

would plot up to six diagnostic graphics; the four default ones are: (a) residuals against fitted values, (b) Scale-Location, i.e. $\sqrt{|\tilde{e}_i|}$ against fitted values, (c) normal Q-Q plot, and (d) residuals against leverage. Plots b–c use standardized residuals defined as $e_i/(\hat{\sigma}\sqrt{1-h_{ii}})$, where h_{ii} are the diagonal elements of the "hat matrix" $\boldsymbol{X}(\boldsymbol{X'X})^{-1}\boldsymbol{X'}$; they are assumed to have constance variance. In figure 10.1, we plot two of them to show how the variance of the observed viscosity tends to increase with increasing viscosity magnitude.

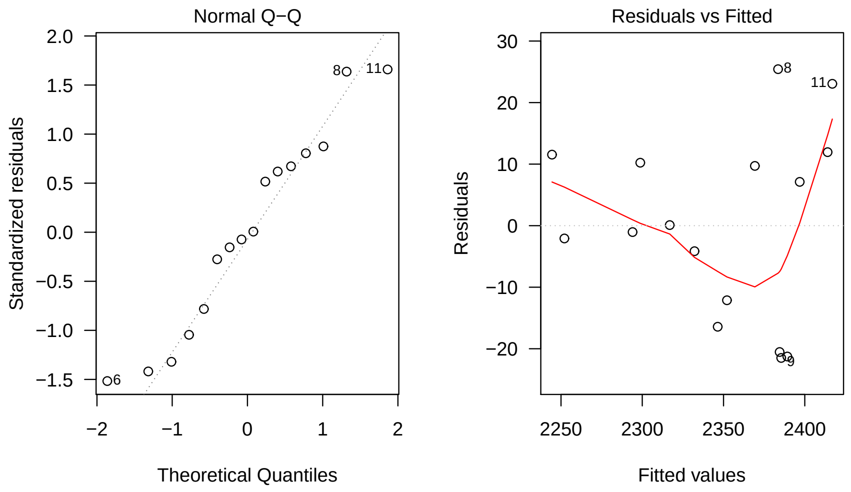


Figure 10.1: Normality probability plot of residuals (left) and plot of residuals against predicted viscosity (right).

Looking at partial residual plots (`plot(resid(lm.fit) ~ visc$Temperature)`) further suggests that viscosity variance increases with temperature (Fig. 10.2). Of note, more elaborated partial residuals plots are available in the `car` package, e.g.

```
library(car)
crPlots(lm.fit)
```

10.2.3 Regression analysis of a 2^3 factorial design

In this designed experiment, we are interested in investigating the yield of a process. The variables of interest are: temperature (x_1), pressure (x_2), and catalyst concentration (x_3). Each variable can be run at a low and a high level, yielding a 2^3 design with four center points (Fig. 10.3); factor levels are coded as ±1.

December 2, 2012
Draft Version

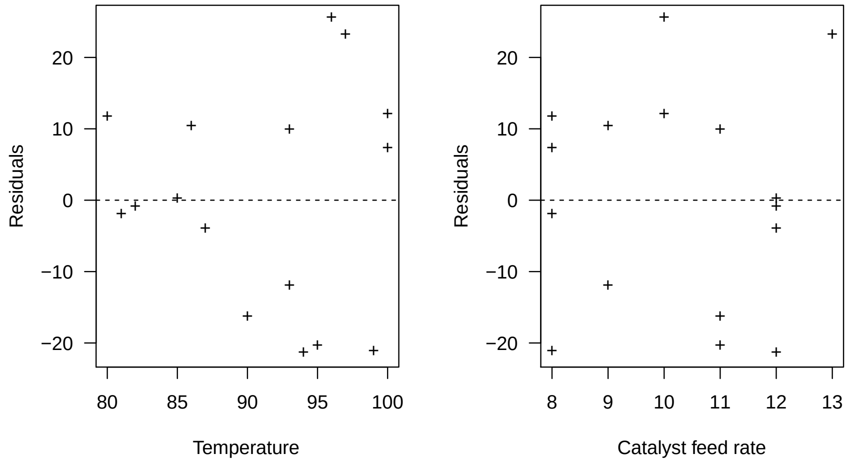


Figure 10.2: Plot of residuals *vs.* each predictor.

Let's consider the following model, with main effects only:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon.$$

It is easy to check that

$$\boldsymbol{X'X} = \begin{bmatrix} 12 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 8 \end{bmatrix} \qquad \boldsymbol{X'y} = \begin{bmatrix} 612 \\ 45 \\ 85 \\ 9 \end{bmatrix}$$

Now, using the `lm()` function,

```
yield2 <- read.table("yield2.txt", header=TRUE,
                     colClasses=c(rep("factor",3),"numeric"))
levels(yield2$Temp) <- c(-1,0,1)
levels(yield2$Pressure) <- c(-1,0,1)
levels(yield2$Conc) <- c(-1,0,1)
lm.fit <- lm(Yield ~ as.numeric(as.character(Temp))+
             as.numeric(as.character(Pressure))+
             as.numeric(as.character(Conc)), data=yield2)
summary(lm.fit)
```

the fitted regression model is

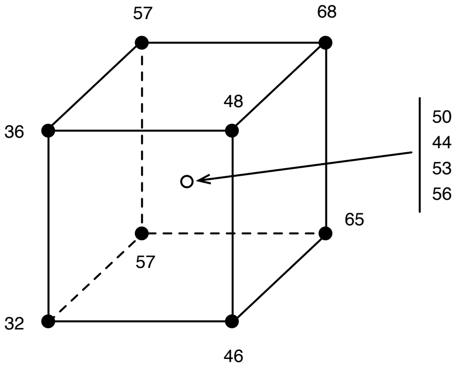
$$\hat{y} = 51.000 + 5.625x_1 + 10.625x_2 + 1.125x3.$$

Note that we need to convert the original scales into the recoded factor levels, and using simply `as.numeric()` would not be sufficient as it would automatically convert factor values into $\{1, 2, 3\}$ (this won't change anything except for the intercept term).

December 2, 2012
Draft Version

	Factor		
Run	x_1	x_2	x_3
1	-1	-1	-1
2	$+1$	-1	-1
3	-1	$+1$	-1
4	$+1$	$+1$	-1
5	-1	-1	$+1$
6	$+1$	-1	$+1$
7	-1	$+1$	$+1$
8	$+1$	$+1$	$+1$
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0

(a) Design matrix



(b) Geometric view

Figure 10.3: Experiment design for the chemical experiment.

To see the connection with the analysis of 2^3 design discussed in Chapter ??, let compute the effect of temperature as the difference of means between high and low levels, that is $T = \bar{y}_{T+} - \bar{y}_{T-}$:

```
diff(with(subset(yield2, Temp != 0),
          tapply(Yield, droplevels(Temp), mean)))
```

This is twice the regression coefficient for temperature. For every 2^k design, the regression coefficient is always one-half the usual effect estimate. Moreover, this shows that effect estimates from a 2^k design are least squares estimates.

In the previous example, $\boldsymbol{X'X}$ was diagonal, so that its inverse was also diagonal. This means that the estimators of all regression coefficients are uncorrelated, or $\text{Cov}(\hat{\beta}_i, \hat{\beta}_j) = 0$, $\forall i \neq j$. To deliberately achieve such a result, one must choose a design matrix where columns are orthogonal (i.e., linearly independent): This is called an *orthogonal design*, as is the case for 2^k factorial designs. Montgomery gives two examples of how regression analysis might be useful in the case of non-orthogonal designs, either after removing one run from the preceding experiment or after introducing small discrepancies in the precision of the factor levels.

10.2.4 Aliasing and fractional design

Another application of regression analysis in the spirit of the analysis of designed experiments is the de-aliasing of interactions in a fractional factorial plan.

December 2, 2012
Draft Version

10.3 Regression analysis: Hypothesis Testing and Confidence intervals

10.4 Prediction of new observations

10.5 Regression Model diagnostics

December 2, 2012
Draft Version

Chapter 11

Response Surface Methods and Designs

December 2, 2012
Draft Version

Chapter 12

Robust Parameter Design and Process Robustness Studies

December 2, 2012
Draft Version

Chapter 13

Experiments with Random Factors

```
strength <- read.table("strength.txt",header=TRUE)
strength$Looms <- factor(strength$Looms)
strength$Obs <- factor(strength$Obs)
library(nlme)
strength.lmm <- lme(y~as.factor(Looms),data=strength,random=~1)
```

December 2, 2012
Draft Version

Chapter 14

Nested and Split-Plot Designs

December 2, 2012
Draft Version

Chapter 15

Other Design and Analysis Topics

December 2, 2012
Draft Version

Bibliography

1. J-M Azaïs and J-M Bardet. *Le modèle linéaire par l'exemple.* Dunod, 2005.

2. R A Becker, W S Cleveland, M J Shyu, and S Kaluzny. A tour of trellis display. Technical report, Bell Labs, 1995. Also available as http://cm.bell-labs.com/cm/ms/departments/sia/wsc/webpapers.html.

3. D Benoist. *Plans d'Expériences : Construction et Analyse.* Lavoisier, 1994.

4. P D Berger and R E Maurer. *Experimental Design with Applications in Management, Engineering and the Sciences.* Duxbury Press, 2002.

5. J M Bland and D G Altman. Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet*, pages 307–310, 1986.

6. R Christensen. *Plane Answers to Complex Questions. The Theory of Linear Models.* New York: Springer, 3rd edition, 2002.

7. P Dagnelie. *Principes d'expérimentation. Planification des expériences et analyses de leurs résultats.* Gembloux, Presses agronomiques, 2003.

8. S Dudoit and M J van der Laan. *Multiple Testing Procedures with Applications to Genomics.* New York: Springer, 2008.

9. J J Faraway. *Linear Models with R.* Chapman & Hall/CRC, 2005.

10. J J Faraway. *Extending the inear Model with R. Generalized Linear, Mixed Effects and Nonparamaetric Regression Models.* Chapman & Hall/CRC, 2006.

11. R A Fisher. *Statistical Methods, Experimental Design, and Scientific Infe- rences.* Oxford: Oxford University Press, 1990.

12. J E Gentle. *Random Number Generation and Monte Carlo Methods.* New York: Springer, 2003.

13. J Goupy. *Pratiquer les plans d'expérience.* Genève, Ellipses, 2004.

14. K Hinkelmann and O Kempthorne. *Design and Analysis of Experiments. Volume 2: Advanced Experimental Design.* John Wiley & Sons, Inc., 2005.

December 2, 2012
Draft Version

15. Julian Faraway July. Practical regression and anova using r, 2002.

16. B Lecoutre, M-P Lecoutre, and J Poitevineau. Uses, abuses and misuses of significance tests in the scientific community: won't the bayesian choice be unavoidable? *International Statistical Review*, 69:399–418, 2001.

17. M-P Lecoutre, J Poitevineau, and B Lecoutre. Even statisticians are not immune to misinterpretations of null hypothesis significance tests. *International Journal of Psychology*, 38:37–45, 2003.

18. U Munzel and L A Hothorn. A unified approach to simultaneous rank test procedures in the unbalanced one-way layout. *Biometrical Journal*, 43(5):553–569, 2001.

19. K R Murphy and B Myors. *Statistical Power Analysis: A Simple and General Model for Traditional and Modern Hypothesis Tests.* Mahwah, NJ: Lawrence Erlbaum Associates, 2nd edition, 2004.

20. J C Pinheiro and D M Bates. *Mixed-Effects Models in S and S-PLUS.* New York: Springer, 2000.

21. F W Satterthwaite. An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2:110–114, 1946.

22. L E Toothaker. *Multiple Comparison Procedures.* SAGE Publications, 1993.

23. J W Tukey. Comparing individual means in the analysis of variance. *Biometrics*, 5:99–114, 1949.

24. J W Tukey. One degree of freedom for non-additivity. *Biometrics*, 5:232–242, 1949.

25. W J van der Linden, B P Veldkamp, and J E Carlson. Optimizing balanced incomplete block designs for educational assessments. *Applied Psychological Measurement*, 28:317–331, 2004.

26. W N Venables and B D Ripley. *Modern Applied Statistics with S.* New York: Springer, 4th edition, 2002.

27. B L Welch. The generalization of "student's" problem when several different population variances are involved. *Biometrika*, 34:28–35, 1947.

28. B L Welch. On the comparison of several mean values: an alternative approach. *Biometrika*, 38:330–336, 1951.

December 2, 2012
Draft Version

Standard Normal and Student t distributions

Table 1: Standard Normal Distribution. Cumulative distribution function of the $\mathcal{N}(0,1)$.

x	**0.00**	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.06**	**0.07**	**0.08**	**0.09**
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990
3.1	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998
3.5	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998
3.6	0.9998	0.9998	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.7	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.8	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.9	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

December 2, 2012
Draft Version

Table 2: Standard Normal Distribution. Quantiles $u(\beta)$ of the order β from $\mathcal{N}(0,1)$.

β	**0.000**	**0.001**	**0.002**	**0.003**	**0.004**	**0.005**	**0.006**	**0.007**	**0.008**	**0.009**
0.50	0.0000	0.0025	0.0050	0.0075	0.0100	0.0125	0.0150	0.0175	0.0201	0.0226
0.51	0.0251	0.0276	0.0301	0.0326	0.0351	0.0376	0.0401	0.0426	0.0451	0.0476
0.52	0.0502	0.0527	0.0552	0.0577	0.0602	0.0627	0.0652	0.0677	0.0702	0.0728
0.53	0.0753	0.0778	0.0803	0.0828	0.0853	0.0878	0.0904	0.0929	0.0954	0.0979
0.54	0.1004	0.1030	0.1055	0.1080	0.1105	0.1130	0.1156	0.1181	0.1206	0.1231
0.55	0.1257	0.1282	0.1307	0.1332	0.1358	0.1383	0.1408	0.1434	0.1459	0.1484
0.56	0.1510	0.1535	0.1560	0.1586	0.1611	0.1637	0.1662	0.1687	0.1713	0.1738
0.57	0.1764	0.1789	0.1815	0.1840	0.1866	0.1891	0.1917	0.1942	0.1968	0.1993
0.58	0.2019	0.2045	0.2070	0.2096	0.2121	0.2147	0.2173	0.2198	0.2224	0.2250
0.59	0.2275	0.2301	0.2327	0.2353	0.2378	0.2404	0.2430	0.2456	0.2482	0.2508
0.60	0.2533	0.2559	0.2585	0.2611	0.2637	0.2663	0.2689	0.2715	0.2741	0.2767
0.61	0.2793	0.2819	0.2845	0.2871	0.2898	0.2924	0.2950	0.2976	0.3002	0.3029
0.62	0.3055	0.3081	0.3107	0.3134	0.3160	0.3186	0.3213	0.3239	0.3266	0.3292
0.63	0.3319	0.3345	0.3372	0.3398	0.3425	0.3451	0.3478	0.3505	0.3531	0.3558
0.64	0.3585	0.3611	0.3638	0.3665	0.3692	0.3719	0.3745	0.3772	0.3799	0.3826
0.65	0.3853	0.3880	0.3907	0.3934	0.3961	0.3989	0.4016	0.4043	0.4070	0.4097
0.66	0.4125	0.4152	0.4179	0.4207	0.4234	0.4261	0.4289	0.4316	0.4344	0.4372
0.67	0.4399	0.4427	0.4454	0.4482	0.4510	0.4538	0.4565	0.4593	0.4621	0.4649
0.68	0.4677	0.4705	0.4733	0.4761	0.4789	0.4817	0.4845	0.4874	0.4902	0.4930
0.69	0.4959	0.4987	0.5015	0.5044	0.5072	0.5101	0.5129	0.5158	0.5187	0.5215
0.70	0.5244	0.5273	0.5302	0.5330	0.5359	0.5388	0.5417	0.5446	0.5476	0.5505
0.71	0.5534	0.5563	0.5592	0.5622	0.5651	0.5681	0.5710	0.5740	0.5769	0.5799
0.72	0.5828	0.5858	0.5888	0.5918	0.5948	0.5978	0.6008	0.6038	0.6068	0.6098
0.73	0.6128	0.6158	0.6189	0.6219	0.6250	0.6280	0.6311	0.6341	0.6372	0.6403
0.74	0.6433	0.6464	0.6495	0.6526	0.6557	0.6588	0.6620	0.6651	0.6682	0.6713
0.75	0.6745	0.6776	0.6808	0.6840	0.6871	0.6903	0.6935	0.6967	0.6999	0.7031
0.76	0.7063	0.7095	0.7128	0.7160	0.7192	0.7225	0.7257	0.7290	0.7323	0.7356
0.77	0.7388	0.7421	0.7454	0.7488	0.7521	0.7554	0.7588	0.7621	0.7655	0.7688
0.78	0.7722	0.7756	0.7790	0.7824	0.7858	0.7892	0.7926	0.7961	0.7995	0.8030
0.79	0.8064	0.8099	0.8134	0.8169	0.8204	0.8239	0.8274	0.8310	0.8345	0.8381
0.80	0.8416	0.8452	0.8488	0.8524	0.8560	0.8596	0.8633	0.8669	0.8705	0.8742
0.81	0.8779	0.8816	0.8853	0.8890	0.8927	0.8965	0.9002	0.9040	0.9078	0.9116
0.82	0.9154	0.9192	0.9230	0.9269	0.9307	0.9346	0.9385	0.9424	0.9463	0.9502
0.83	0.9542	0.9581	0.9621	0.9661	0.9701	0.9741	0.9782	0.9822	0.9863	0.9904
0.84	0.9945	0.9986	1.0027	1.0069	1.0110	1.0152	1.0194	1.0237	1.0279	1.0322
0.85	1.0364	1.0407	1.0450	1.0494	1.0537	1.0581	1.0625	1.0669	1.0714	1.0758
0.86	1.0803	1.0848	1.0893	1.0939	1.0985	1.1031	1.1077	1.1123	1.1170	1.1217
0.87	1.1264	1.1311	1.1359	1.1407	1.1455	1.1503	1.1552	1.1601	1.1650	1.1700
0.88	1.1750	1.1800	1.1850	1.1901	1.1952	1.2004	1.2055	1.2107	1.2160	1.2212
0.89	1.2265	1.2319	1.2372	1.2426	1.2481	1.2536	1.2591	1.2646	1.2702	1.2759
0.90	1.2816	1.2873	1.2930	1.2988	1.3047	1.3106	1.3165	1.3225	1.3285	1.3346
0.91	1.3408	1.3469	1.3532	1.3595	1.3658	1.3722	1.3787	1.3852	1.3917	1.3984
0.92	1.4051	1.4118	1.4187	1.4255	1.4325	1.4395	1.4466	1.4538	1.4611	1.4684
0.93	1.4758	1.4833	1.4909	1.4985	1.5063	1.5141	1.5220	1.5301	1.5382	1.5464
0.94	1.5548	1.5632	1.5718	1.5805	1.5893	1.5982	1.6072	1.6164	1.6258	1.6352
0.95	1.6449	1.6546	1.6646	1.6747	1.6849	1.6954	1.7060	1.7169	1.7279	1.7392
0.96	1.7507	1.7624	1.7744	1.7866	1.7991	1.8119	1.8250	1.8384	1.8522	1.8663
0.97	1.8808	1.8957	1.9110	1.9268	1.9431	1.9600	1.9774	1.9954	2.0141	2.0335
0.98	2.0537	2.0749	2.0969	2.1201	2.1444	2.1701	2.1973	2.2262	2.2571	2.2904
0.99	2.3263	2.3656	2.4089	2.4573	2.5121	2.5758	2.6521	2.7478	2.8782	3.0902

December 2, 2012
Draft Version

Table 3: Student distribution. Quantiles $u(\beta)$ of the order β from $\mathcal{T}(\nu)$.

mβ	**0.60**	**0.70**	**0.80**	**0.90**	**0.95**	**0.9750**	**0.9900**	**0.9950**	**0.9990**	**0.9995**
1	0.325	0.727	1.376	3.078	6.314	12.706	31.821	63.657	318.309	636.619
2	0.289	0.617	1.061	1.886	2.920	4.303	6.965	9.925	22.327	31.599
3	0.277	0.584	0.978	1.638	2.353	3.182	4.541	5.841	10.215	12.924
4	0.271	0.569	0.941	1.533	2.132	2.776	3.747	4.604	7.173	8.610
5	0.267	0.559	0.920	1.476	2.015	2.571	3.365	4.032	5.893	6.869
6	0.265	0.553	0.906	1.440	1.943	2.447	3.143	3.707	5.208	5.959
7	0.263	0.549	0.896	1.415	1.895	2.365	2.998	3.499	4.785	5.408
8	0.262	0.546	0.889	1.397	1.860	2.306	2.896	3.355	4.501	5.041
9	0.261	0.543	0.883	1.383	1.833	2.262	2.821	3.250	4.297	4.781
10	0.260	0.542	0.879	1.372	1.812	2.228	2.764	3.169	4.144	4.587
11	0.260	0.540	0.876	1.363	1.796	2.201	2.718	3.106	4.025	4.437
12	0.259	0.539	0.873	1.356	1.782	2.179	2.681	3.055	3.930	4.318
13	0.259	0.538	0.870	1.350	1.771	2.160	2.650	3.012	3.852	4.221
14	0.258	0.537	0.868	1.345	1.761	2.145	2.624	2.977	3.787	4.140
15	0.258	0.536	0.866	1.341	1.753	2.131	2.602	2.947	3.733	4.073
16	0.258	0.535	0.865	1.337	1.746	2.120	2.583	2.921	3.686	4.015
17	0.257	0.534	0.863	1.333	1.740	2.110	2.567	2.898	3.646	3.965
18	0.257	0.534	0.862	1.330	1.734	2.101	2.552	2.878	3.610	3.922
19	0.257	0.533	0.861	1.328	1.729	2.093	2.539	2.861	3.579	3.883
20	0.257	0.533	0.860	1.325	1.725	2.086	2.528	2.845	3.552	3.850
21	0.257	0.532	0.859	1.323	1.721	2.080	2.518	2.831	3.527	3.819
22	0.256	0.532	0.858	1.321	1.717	2.074	2.508	2.819	3.505	3.792
23	0.256	0.532	0.858	1.319	1.714	2.069	2.500	2.807	3.485	3.768
24	0.256	0.531	0.857	1.318	1.711	2.064	2.492	2.797	3.467	3.745
25	0.256	0.531	0.856	1.316	1.708	2.060	2.485	2.787	3.450	3.725
26	0.256	0.531	0.856	1.315	1.706	2.056	2.479	2.779	3.435	3.707
27	0.256	0.531	0.855	1.314	1.703	2.052	2.473	2.771	3.421	3.690
28	0.256	0.530	0.855	1.313	1.701	2.048	2.467	2.763	3.408	3.674
29	0.256	0.530	0.854	1.311	1.699	2.045	2.462	2.756	3.396	3.659
30	0.256	0.530	0.854	1.310	1.697	2.042	2.457	2.750	3.385	3.646
31	0.256	0.530	0.853	1.309	1.696	2.040	2.453	2.744	3.375	3.633
32	0.255	0.530	0.853	1.309	1.694	2.037	2.449	2.738	3.365	3.622
33	0.255	0.530	0.853	1.308	1.692	2.035	2.445	2.733	3.356	3.611
34	0.255	0.529	0.852	1.307	1.691	2.032	2.441	2.728	3.348	3.601
35	0.255	0.529	0.852	1.306	1.690	2.030	2.438	2.724	3.340	3.591
36	0.255	0.529	0.852	1.306	1.688	2.028	2.434	2.719	3.333	3.582
37	0.255	0.529	0.851	1.305	1.687	2.026	2.431	2.715	3.326	3.574
38	0.255	0.529	0.851	1.304	1.686	2.024	2.429	2.712	3.319	3.566
39	0.255	0.529	0.851	1.304	1.685	2.023	2.426	2.708	3.313	3.558
40	0.255	0.529	0.851	1.303	1.684	2.021	2.423	2.704	3.307	3.551
41	0.255	0.529	0.850	1.303	1.683	2.020	2.421	2.701	3.301	3.544
42	0.255	0.528	0.850	1.302	1.682	2.018	2.418	2.698	3.296	3.538
43	0.255	0.528	0.850	1.302	1.681	2.017	2.416	2.695	3.291	3.532
44	0.255	0.528	0.850	1.301	1.680	2.015	2.414	2.692	3.286	3.526
45	0.255	0.528	0.850	1.301	1.679	2.014	2.412	2.690	3.281	3.520
46	0.255	0.528	0.850	1.300	1.679	2.013	2.410	2.687	3.277	3.515
47	0.255	0.528	0.849	1.300	1.678	2.012	2.408	2.685	3.273	3.510
48	0.255	0.528	0.849	1.299	1.677	2.011	2.407	2.682	3.269	3.505
49	0.255	0.528	0.849	1.299	1.677	2.010	2.405	2.680	3.265	3.500
50	0.255	0.528	0.849	1.299	1.676	2.009	2.403	2.678	3.261	3.496

December 2, 2012
Draft Version

List of Figures

December 2, 2012
Draft Version

December 2, 2012
Draft Version

List of Tables