The Statistical Sleuth in R: Chapter 5

Kate Aloisio Ruobing Zhang Nicholas J. Horton*

September 30, 2013

Contents

1 Introduction

This document is intended to help describe how to undertake analyses introduced as examples in the Second Edition of the *Statistical Sleuth* (2002) by Fred Ramsey and Dan Schafer. More information about the book can be found at `http://www.proaxis.com/~panorama/home.htm`. This file as well as the associated `knitr` reproducible analysis source file can be found at `http://www.amherst.edu/~nhorton/sleuth`.

This work leverages initiatives undertaken by Project MOSAIC (`http://www.mosaic-web.org`), an NSF-funded effort to improve the teaching of statistics, calculus, science and computing in the undergraduate curriculum. In particular, we utilize the `mosaic` package, which was written to simplify the use of R for introductory statistics courses. A short summary of the R needed to teach introductory statistics can be found in the mosaic package vignette (`http://cran.r-project.org/web/packages/mosaic/vignettes/MinimalR.pdf`).

*Department of Mathematics, Amherst College, nhorton@amherst.edu

To use a package within R, it must be installed (one time), and loaded (each session). The package can be installed using the following command:

```
> install.packages("mosaic")  # note the quotation marks
```

Once this is installed, it can be loaded by running the command:

```
> require(mosaic)
```

This needs to be done once per session.

In addition the data files for the *Sleuth* case studies can be accessed by installing the `Sleuth2` package.

```
> install.packages("Sleuth2")  # note the quotation marks
```

```
> require(Sleuth2)
```

We also set some options to improve legibility of graphs and output.

```
> trellis.par.set(theme = col.mosaic())  # get a better color scheme
> options(digits = 3)
>
```

The specific goal of this document is to demonstrate how to calculate the quantities described in Chapter 5: Comparisons Among Several Samples using R.

2 Diet and lifespan

Does restricting the diet of female mice lead to increased lifespan? This is the question addressed in case study 5.1 in the *Sleuth*.

2.1 Summary statistics and graphical display

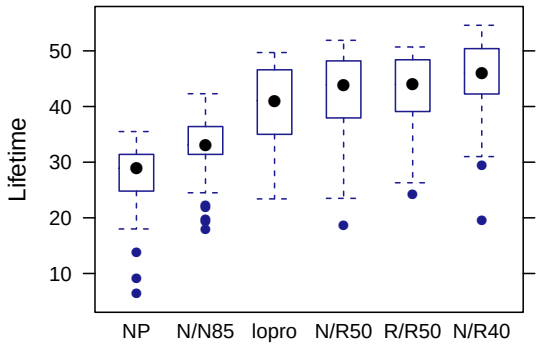
We begin by reading the data and summarizing the variables.

```
> summary(case0501)

    Lifetime       Diet
 Min.   : 6.4   NP   :49
 1st Qu.:31.8   N/N85:57
 Median :39.5   lopro:56
 Mean   :38.8   N/R50:71
 3rd Qu.:46.9   R/R50:56
 Max.   :54.6   N/R40:60
```

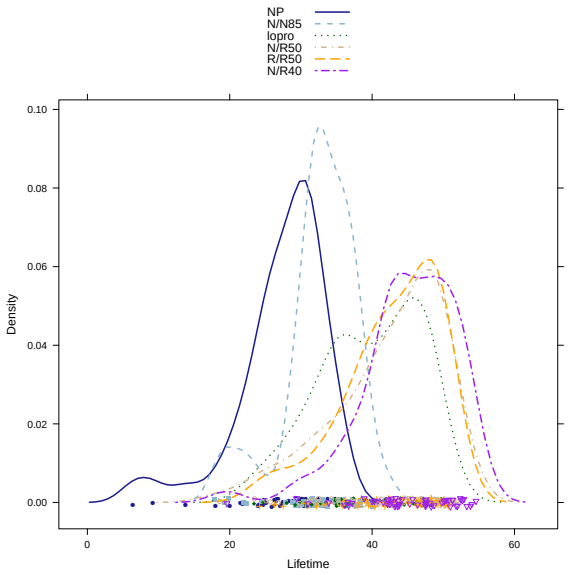
```
> favstats(Lifetime ~ Diet, data = case0501)

       min    Q1 median    Q3  max  mean    sd  n missing
NP     6.4 24.80  28.90 31.40 35.5 27.40 6.134 49       0
N/N85 17.9 31.40  33.10 36.40 42.3 32.69 5.125 57       0
lopro 23.4 35.00  41.05 46.45 49.7 39.69 6.992 56       0
N/R50 18.6 37.95  43.90 48.20 51.9 42.30 7.768 71       0
R/R50 24.2 39.15  43.95 48.35 50.7 42.89 6.683 56       0
N/R40 19.6 42.27  46.05 50.35 54.6 45.12 6.703 60       0
```

There were a total of 349 female mice. These mice were randomly assigned to one of 6 diets. Their lifetimes were then recorded, as shown in Display 5.2 (page 115 of the *Sleuth*).

```
> bwplot(Lifetime ~ Diet, data = case0501)  # Display 5.1
```




```
> densityplot(~Lifetime, groups = Diet, auto.key = TRUE, data = case0501)
```



2.2 One-way ANOVA

First we fit the one way analysis of variance (ANOVA) model, using all of the groups.

```
> anova(lm(Lifetime ~ Diet, data = case0501))

Analysis of Variance Table

Response: Lifetime
           Df Sum Sq Mean Sq F value Pr(>F)
Diet        5  12734    2547    57.1 <2e-16 ***
Residuals 343  15297      45
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is evidence of a highly statistically significant difference between the diets.

By default, the use of the linear model (regression) function displays the pairwise differences between the first group and each of the other groups. Note that the overall test of the model is the same.

```
> summary(lm(Lifetime ~ Diet, data = case0501))


Call:
lm(formula = Lifetime ~ Diet, data = case0501)

Residuals:
    Min      1Q  Median      3Q     Max
-25.517  -3.386   0.814   5.183  10.014

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   27.402      0.954   28.72  < 2e-16 ***
DietN/N85      5.289      1.301    4.07  5.9e-05 ***
Dietlopro     12.284      1.306    9.40  < 2e-16 ***
DietN/R50     14.895      1.240   12.01  < 2e-16 ***
DietR/R50     15.484      1.306   11.85  < 2e-16 ***
DietN/R40     17.715      1.286   13.78  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.68 on 343 degrees of freedom
Multiple R-squared:  0.454,Adjusted R-squared:  0.446
F-statistic: 57.1 on 5 and 343 DF,  p-value: <2e-16
```

The reference group is *NP*, followed by *N/N85, lopro, N/R50, R/R50, N/R40.*

2.3 Pairwise comparisons

Next we used contrasts for the results on page 121, Display 5.7, and part **(a)** on page 115:

```
> require(gmodels)

Loading required package: gmodels

> # N/N85 vs N/R50
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(0, -1, 0, 1, 0,
+     0), conf.int = 0.95)

                       Estimate Std. Error t value  Pr(>|t|) lower CI
Diet c=( 0 -1 0 1 0 0 )    9.606      1.188   8.088 1.057e-14     7.27
                       upper CI
Diet c=( 0 -1 0 1 0 0 )    11.94
```

The results for **(b)** on page 115-116:

```
> # N/R50 vs R/R50 (b)
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(0, 0, 0, -1, 1,
+     0), conf.int = 0.95)

                       Estimate Std. Error t value Pr(>|t|) lower CI
Diet c=( 0 0 0 -1 1 0 )   0.5885      1.194  0.4931   0.6223   -1.759
                       upper CI
Diet c=( 0 0 0 -1 1 0 )    2.936
```

The results for **(c)** on page 116:

```
> # N/R40 vs N/R50 (c)
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(0, 0, 0, -1, 0,
+     1), conf.int = 0.95)

                       Estimate Std. Error t value Pr(>|t|) lower CI
Diet c=( 0 0 0 -1 0 1 )    2.819      1.171   2.408  0.01659    0.516
                       upper CI
Diet c=( 0 0 0 -1 0 1 )    5.123

> # N/N85 vs N/R40
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(0, -1, 0, 0, 0,
+     1), conf.int = 0.95)

                       Estimate Std. Error t value  Pr(>|t|) lower CI
Diet c=( 0 -1 0 0 0 1 )    12.43      1.235   10.06 4.964e-21    9.996
                       upper CI
Diet c=( 0 -1 0 0 0 1 )    14.85
```

The results for **(d)** on page 116:

```
> # N/R50 vs N/R50 lopro (d)
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(0, 0, 1, -1, 0,
+     0), conf.int = 0.95)

                        Estimate Std. Error t value Pr(>|t|) lower CI
Diet c=( 0 0 1 -1 0 0 )   -2.611      1.194  -2.188  0.02935   -4.959
                        upper CI
Diet c=( 0 0 1 -1 0 0 )  -0.2639
```

The results for **(e)** on page 116:

```
> # N/N85 vs NP (e)
> fit.contrast(lm(Lifetime ~ Diet, data = case0501), "Diet", c(-1, 1, 0, 0, 0,
+     0), conf.int = 0.95)

                        Estimate Std. Error t value  Pr(>|t|) lower CI
Diet c=( -1 1 0 0 0 0 )    5.289      1.301   4.065 5.949e-05     2.73
                        upper CI
Diet c=( -1 1 0 0 0 0 )    7.848
```

Another way of viewing these results is through a model table, which displays the differences between the grand mean and the group means.

```
> model.tables(aov(lm(Lifetime ~ Diet, data = case0501)))

Tables of effects

 Diet
       NP  N/N85   lopro N/R50  R/R50 N/R40
    -11.4 -6.106  0.8886   3.5  4.089  6.32
rep  49.0 57.000 56.0000  71.0 56.000 60.00
```

Another way of calculating the above results is done with the following code:

```
> mean(Lifetime ~ Diet, data = case0501) - mean(~Lifetime, data = case0501)

      NP    N/N85    lopro    N/R50    R/R50    N/R40
-11.3951  -6.1059   0.8886   3.5000   4.0886   6.3195
```

2.4 Other analyses

We will next demonstrate how to calculate the quantities on page 120 (Display 5.6).

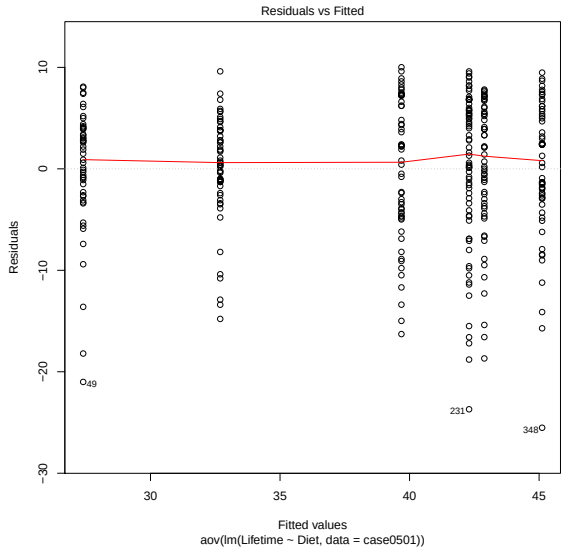
```
> df = length(case0501$Diet) - length(unique(case0501$Diet))
> df

[1] 343

> sdvals = with(case0501, tapply(Lifetime, Diet, sd))
> sdvals

   NP N/N85 lopro N/R50 R/R50 N/R40
6.134 5.125 6.992 7.768 6.683 6.703

> nvals = with(case0501, tapply(Lifetime, Diet, length))
> nvals

   NP N/N85 lopro N/R50 R/R50 N/R40
   49    57    56    71    56    60

> pooledsd = sum(sdvals * nvals)/sum(nvals)
> pooledsd

[1] 6.625
```

Note that the pooled standard deviation reported in chapter 5 is not the same as the root MSE from the ANOVA. For the rest of this document we will use the ANOVA estimate of the root mean squared error.

2.5 Residual analysis and diagnostics

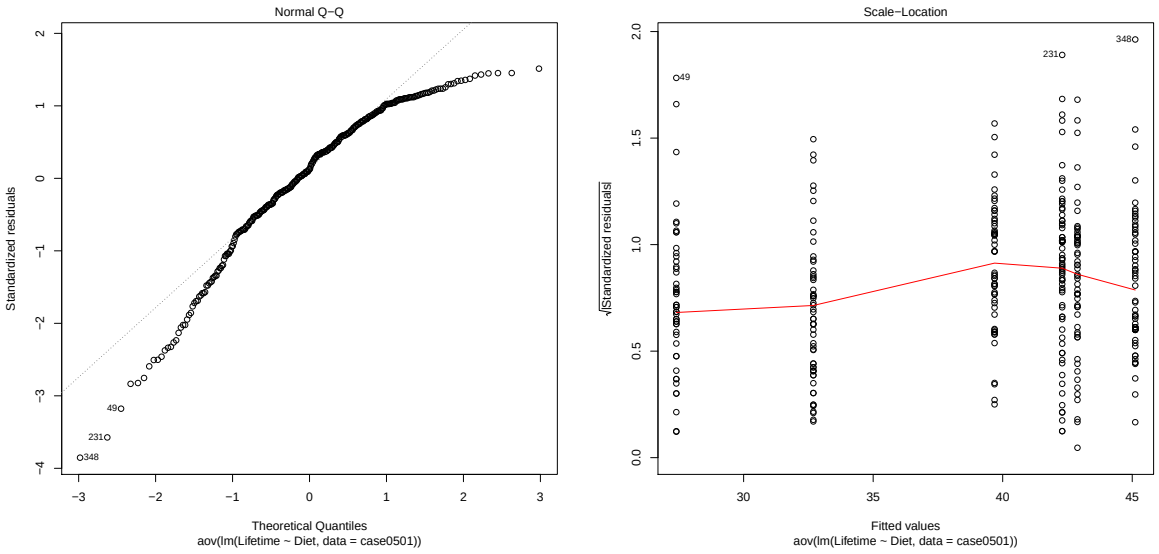
The residuals versus fitted graph does not demonstrate dramatic lack of fit (though some of the mice had very small residuals). The following figure is akin to Display 5.14 (page 132).

```
> aov1 = aov(lm(Lifetime ~ Diet, data = case0501))
> plot(aov1, which = 1)
```



The quantile plot of the residuals indicates that the normality assumption may be violated.

```
> plot(aov1, which = 2)
> plot(aov1, which = 3)
```



3 Spock Conspiracy Trial

Did Dr. Benjamin Spock have a fair trial? More specifically, were women underrepresented on his jury pool? This is the question considered in case study 5.2 in the *Sleuth*.

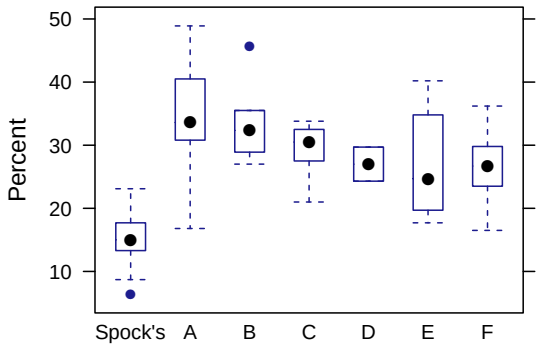
3.1 Summary statistics and graphical display

We begin by reading the data and summarizing the variables.

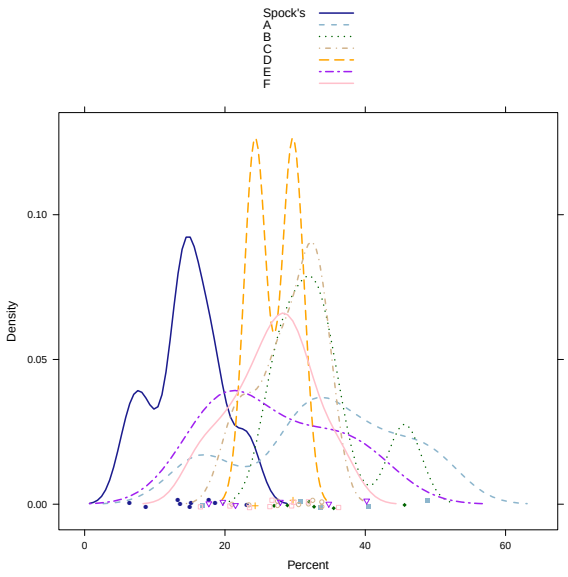
```
> summary(case0502)

    Percent         Judge
 Min.   : 6.4   Spock's:9
 1st Qu.:19.9   A      :5
 Median :27.5   B      :6
 Mean   :26.6   C      :9
 3rd Qu.:32.4   D      :2
 Max.   :48.9   E      :6
                F      :9

> case0502$Judge = with(case0502, as.factor(Judge))
> favstats(Percent ~ Judge, data = case0502)

         min    Q1 median    Q3  max  mean     sd n missing
Spock's  6.4 13.30  15.00 17.70 23.1 14.62  5.039 9       0
A       16.8 30.80  33.60 40.50 48.9 34.12 11.942 5       0
B       27.0 29.67  32.35 34.80 45.6 33.62  6.582 6       0
C       21.0 27.50  30.50 32.50 33.8 29.10  4.593 9       0
D       24.3 25.65  27.00 28.35 29.7 27.00  3.818 2       0
E       17.7 20.15  24.70 33.07 40.2 26.97  9.010 6       0
F       16.5 23.50  26.70 29.80 36.2 26.80  5.969 9       0
```

There were a total of 46 venires. They compared Spock's judge with 6 other judges. The precent of women within each venire was recorded as shown in Display 5.4 (page 117 of the *Sleuth*).

```
> bwplot(Percent ~ Judge, data = case0502)  # Display 5.5 (page 118)
```



```
> densityplot(~Percent, groups = Judge, auto.key = TRUE, data = case0502)
```



3.2 One-way ANOVA

First we fit the one way analysis of variance (ANOVA) model, with all of the groups. These results are summarized on page 118 and shown in Display 5.10 (page 127).

```
> aov1 = anova(lm(Percent ~ Judge, data = case0502))
> aov1

Analysis of Variance Table

Response: Percent
          Df Sum Sq Mean Sq F value  Pr(>F)
Judge      6   1927     321    6.72 6.1e-05 ***
Residuals 39   1864      48
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By default, the use of the linear model (regression) function displays the pairwise differences between the first group and each of the other groups. Note that the overall test of the model is the same.

```
> summary(lm(Percent ~ Judge, data = case0502))


Call:
lm(formula = Percent ~ Judge, data = case0502)
```

```
Residuals:
   Min     1Q Median     3Q    Max
-17.32  -4.37  -0.25   3.32  14.78

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    14.62       2.30    6.34  1.7e-07 ***
JudgeA         19.50       3.86    5.06  1.1e-05 ***
JudgeB         18.99       3.64    5.21  6.4e-06 ***
JudgeC         14.48       3.26    4.44  7.2e-05 ***
JudgeD         12.38       5.41    2.29   0.0275 *
JudgeE         12.34       3.64    3.39   0.0016 **
JudgeF         12.18       3.26    3.74   0.0006 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.91 on 39 degrees of freedom
Multiple R-squared:  0.508,Adjusted R-squared:  0.433
F-statistic: 6.72 on 6 and 39 DF,  p-value: 6.1e-05
```

```
> model.tables(aov(lm(Percent ~ Judge, data = case0502)))

Tables of effects

 Judge
   Spock's     A     B     C      D      E      F
    -11.96 7.537 7.034 2.517 0.4174 0.3841 0.2174
rep   9.00 5.000 6.000 9.000 2.0000 6.0000 9.0000
```

Then we can fit the one way analysis of variance F-test of whether the mean percentage is the same for judges A-F (page 118).

```
> with(subset(case0502, Judge != "Spock's"), anova(lm(Percent ~ Judge)))

Analysis of Variance Table

Response: Percent
          Df Sum Sq Mean Sq F value Pr(>F)
Judge      5    326    65.3    1.22   0.32
Residuals 31   1661    53.6
```

3.3 Additional analyses

Now we will demonstrate how to fit the reduced model comparing Spock's judge to a combination of the other judges. First we create a 2 level version of the grouping variable.

```
> case0502$twoJudge = as.character(case0502$Judge)
> case0502$twoJudge[case0502$Judge != "Spock's"] = "notspock"
> tally(twoJudge ~ Judge, format = "count", data = case0502)

          Judge
twoJudge   Spock's A B C D E F
  notspock       0 5 6 9 2 6 9
  Spock's        9 0 0 0 0 0 0
  Total          9 5 6 9 2 6 9
```

Recall that the book calculates the extra sum of squares as (2,190.90 - 1864.45)/(44-39)) / (1864.45 / 39) = 1.37, with df 5 and 39. P(F > 1.366) = 0.26 (page 130). Below are the calculations for the results found on page 128.

```
> numdf1 = aov1["Residuals", "Df"]
> numdf1  # Within

[1] 39

> ss1 = aov1["Residuals", "Sum Sq"]
> ss1  # Within

[1] 1864

> aov2 = anova(lm(Percent ~ as.factor(twoJudge), data = case0502))
> aov2

Analysis of Variance Table

Response: Percent
                   Df Sum Sq Mean Sq F value Pr(>F)
as.factor(twoJudge)  1   1601    1601    32.1  1e-06 ***
Residuals           44   2191      50
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> df2 = aov2["Residuals", "Df"]
> df2  # Spock and others

[1] 44

> ss2 = aov2["Residuals", "Sum Sq"]
> ss2  # Spock and others

[1] 2191

> Fstat = ((ss2 - ss1)/(df2 - numdf1))/(ss1/numdf1)
> Fstat
```

```
[1] 1.366

> 1 - pf(Fstat, length(levels(case0502$Judge)) - 2, numdf1)

[1] 0.2582
```

We can also compare the two models using ANOVA (Display 5.12, page 130).

```
> anova(lm(Percent ~ as.factor(twoJudge), data = case0502), lm(Percent ~ as.factor(Judge),
+     data = case0502))

Analysis of Variance Table

Model 1: Percent ~ as.factor(twoJudge)
Model 2: Percent ~ as.factor(Judge)
  Res.Df  RSS Df Sum of Sq    F Pr(>F)
1     44 2191
2     39 1864  5       326 1.37   0.26
```

There are some other ways to compare whether the other judges differ from Dr. Spock's judge in their female composition using contrasts.

```
> # test all of the other judges vs. Spock's judge using a contrast page 118
> fit.contrast(lm(Percent ~ Judge, data = case0502), "Judge", c(-6, 1, 1, 1, 1,
+     1, 1), conf.int = 0.95)

                          Estimate Std. Error t value  Pr(>|t|) lower CI
Judge c=( -6 1 1 1 1 1 1 )    89.87      15.85    5.67 1.489e-06    57.81
                          upper CI
Judge c=( -6 1 1 1 1 1 1 )    121.9

>
> # calculate the 95% confidence interval for Dr. Spock's jury female
> # composition page 118
> estimable(lm(Percent ~ Judge, data = case0502), c(1, 0, 0, 0, 0, 0, 0), conf.int = 0.95)

               Estimate Std. Error t value DF  Pr(>|t|) Lower.CI Upper.CI
(1 0 0 0 0 0 0)    14.62      2.305   6.344 39 1.722e-07     9.96    19.28
```

3.3.1 Kruskal-Wallis Nonparametric Analysis of Variance

For the results of the Kruskal-Wallis test on page 136 we can use the following code:

```
> kruskal.test(Percent ~ Judge, data = case0502)
```

```
Kruskal-Wallis rank sum test

data:  Percent by Judge
Kruskal-Wallis chi-squared = 21.96, df = 6, p-value = 0.001229
```