The Statistical Sleuth in R: Chapter 10

Linda Loi   Kate Aloisio   Ruobing Zhang   Nicholas J. Horton*

June 15, 2016

Contents

1 Introduction

This document is intended to help describe how to undertake analyses introduced as examples in the Third Edition of the *Statistical Sleuth* (2013) 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.math.smith.edu/~nhorton/sleuth3`.

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`).

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:

*Department of Mathematics and Statistics, Smith College, nhorton@smith.edu

```
> 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 `Sleuth3` package.

```
> install.packages('Sleuth3')                # note the quotation marks
```

```
> require(Sleuth3)
```

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

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

The specific goal of this document is to demonstrate how to calculate the quantities described in Chapter 10: Inferential Tools for Multiple Regression using R.

2 Galileo's data on the motion of falling bodies

Galileo investigated the relationship between height and horizontal distance. This is the question addressed in case study 10.1 in the *Sleuth*.

2.1 Data coding, summary statistics and graphical display

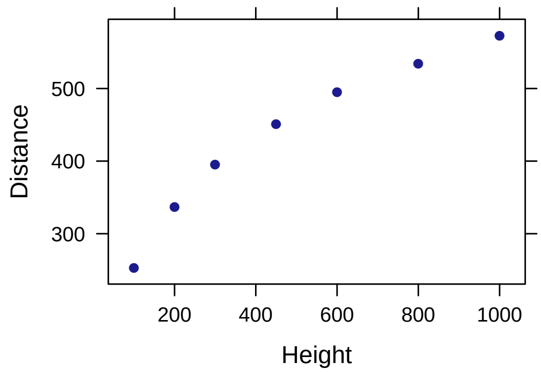
We begin by reading the data and summarizing the variables.

```
> summary(case1001)

   Distance       Height
 Min.   :253   Min.   : 100
 1st Qu.:366   1st Qu.: 250
 Median :451   Median : 450
 Mean   :434   Mean   : 493
 3rd Qu.:514   3rd Qu.: 700
 Max.   :573   Max.   :1000

> favstats(~ Distance, data=case1001)

 min  Q1 median  Q3 max mean  sd n missing
 253 366    451 514 573  434 113 7       0
```

There we a total of 7 trials of Galileo's experiment. For each trial, he recorded the initial height and then measured the horizontal distance as shown in Display 10.1 (page 272).

We can start to explore this relationship by creating a scatterplot of Galileo's horizontal distances versus initial heights. The following graph is akin to Display 10.2 (page 273).

```
> xyplot(Distance ~ Height, data=case1001)
```



2.2 Models

The first model that we created is a cubic model as interpreted on page 273 and summarized in Display 10.13 (page 291).

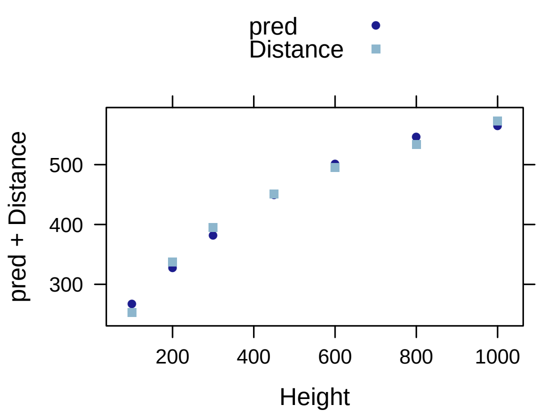
```
> lm1 = lm(Distance ~ Height+I(Height^2)+I(Height^3), data=case1001); summary(lm1)


Call:
lm(formula = Distance ~ Height + I(Height^2) + I(Height^3), data = case1001)

Residuals:
      1       2       3       4       5       6       7
-2.4036  3.5809  1.8917 -4.4688 -0.0804  2.3216 -0.8414

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.56e+02   8.33e+00   18.71  0.00033
Height       1.12e+00   6.57e-02   16.98  0.00044
I(Height^2) -1.24e-03   1.38e-04   -8.99  0.00290
I(Height^3)  5.48e-07   8.33e-08    6.58  0.00715

Residual standard error: 4.01 on 3 degrees of freedom
Multiple R-squared:  0.999,Adjusted R-squared:  0.999
F-statistic: 1.6e+03 on 3 and 3 DF,  p-value: 2.66e-05
```

We next decrease the polynomial for *Height* by one degree to obtain a quadratic model as interpreted on page 273 and summarized in Display 10.7 (page 281). This model is used for most of the following results.

```
> lm2 = lm(Distance ~ Height+I(Height^2), data=case1001); summary(lm2)


Call:
lm(formula = Distance ~ Height + I(Height^2), data = case1001)

Residuals:
     1      2      3      4      5      6      7
-14.31   9.17  13.52   1.94  -6.18 -12.61   8.46

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.00e+02   1.68e+01   11.93  0.00028
Height       7.08e-01   7.48e-02    9.47  0.00069
I(Height^2) -3.44e-04   6.68e-05   -5.15  0.00676

Residual standard error: 13.6 on 4 degrees of freedom
Multiple R-squared:  0.99,Adjusted R-squared:  0.986
F-statistic:  205 on 2 and 4 DF,  p-value: 9.33e-05
```

The following figure presents the predicted values from the quadratic model using the original data points akin to Display 10.2 (page 273).

```
> case1001$pred = predict(lm2)
> xyplot(pred+Distance ~ Height, auto.key=TRUE, data=case1001)
```



To obtain the expected values of $\hat{\mu}\,(\text{Distance}|\text{Height} = 0)$ and $\hat{\mu}\,(\text{Distance}|\text{Height} = 250)$, we used the `predict()` command with the quadratic model as shown in Display 10.7 (page 281).

```
> predict(lm2, interval="confidence", data.frame(Height=c(0, 250)))

  fit lwr upr
1 200 153 246
2 356 337 374
```

We can also verify the above confidence interval calculations with the following code:

```
> 355.1+c(-1, 1)*6.62*qt(.975, 4)

[1] 337 373
```

To verify numbers on page 284, an interval for the predicted values , we used the following code:

```
> predict(lm2, interval="predict", data.frame(Height=c(0, 250)))

  fit lwr upr
1 200 140 260
2 356 313 398
```

Lastly, we produced an ANOVA for the quadratic model interpreted on page 288 (Display 10.11).

```
> anova(lm2)

Analysis of Variance Table

Response: Distance
            Df Sum Sq Mean Sq F value Pr(>F)
Height       1  71351   71351   383.6  4e-05
I(Height^2)  1   4927    4927    26.5 0.0068
Residuals    4    744     186
```

3 Echolocation in bats

How do bats make their way about in the dark? Echolocation requires a lot of energy. Does it depend on mass and species? This is the question addressed in case study 10.2 in the *Sleuth*.

3.1 Data coding, summary statistics and graphical display

We begin by reading the data, performing transformations where necessary and summarizing the variables.

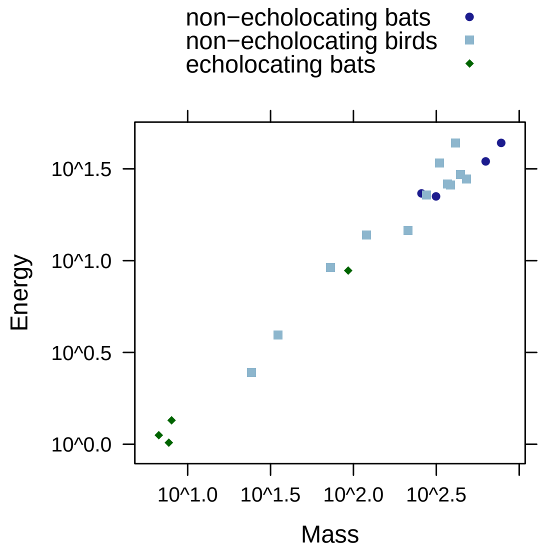
```
> case1002 = transform(case1002, Type = factor(Type, levels = c("non-echolocating bats","non-e
> case1002$logmass = log(case1002$Mass); case1002$logenergy = log(case1002$Energy)
> summary(case1002)

      Mass                            Type          Energy         logmass
 Min.   :  7   non-echolocating bats : 4   Min.   : 1.0   Min.   :1.90
 1st Qu.: 63   non-echolocating birds:12   1st Qu.: 7.6   1st Qu.:4.10
```

```
 Median :266   echolocating bats     : 4   Median :22.6   Median :5.58
 Mean   :263                               Mean   :19.5   Mean   :4.89
 3rd Qu.:391                               3rd Qu.:28.2   3rd Qu.:5.97
 Max.   :779                               Max.   :43.7   Max.   :6.66
   logenergy
 Min.   :0.02
 1st Qu.:1.98
 Median :3.12
 Mean   :2.48
 3rd Qu.:3.34
 Max.   :3.78

> favstats(Mass ~ Type, data=case1002)

                     Type   min     Q1 median    Q3 max  mean    sd  n
1  non-echolocating bats 258.0 300.75 471.50 665.8 779 495.0 249.6  4
2 non-echolocating birds  24.3 108.20 302.50 391.0 480 263.2 165.2 12
3      echolocating bats   6.7   7.45   7.85  29.2  93  28.9  42.8  4
  missing
1       0
2       0
3       0

> favstats(Energy ~ Type, data=case1002)

                     Type   min   Q1 median    Q3   max  mean    sd  n
1  non-echolocating bats 22.40 23.1  29.05 37.02 43.70 31.05 10.15  4
2 non-echolocating birds  2.46 12.6  24.35 28.23 43.70 21.15 12.52 12
3      echolocating bats  1.02  1.1   1.24  3.22  8.83  3.08  3.84  4
  missing
1       0
2       0
3       0
```

A total of 20 flying vertebrates were included in this study. There were 4 echolocating bats, 4 non-echolocating bats, and 12 non-echolocating birds. For each subject their *mass* and *flight energy expenditure* were recorded as shown in Display 10.3 (page 274).

We can next observe the pattern between log(energy expenditure) as a function of log(body mass) for each group with a scatterplot. The following figure is akin to Display 10.4 (page 275).

```
> xyplot(Energy ~ Mass, group=Type, scales=list(y=list(log=TRUE),
+     x=list(log=TRUE)), auto.key=TRUE, data=case1002)
```



3.2 Multiple regression

We first evaluate a multiple regression model for log(energy expenditure) given type of species and log(body mass) as defined on page 276 and shown in Display 10.6 (page 277).

```
> lm1 = lm(logenergy ~ logmass+Type, data=case1002); summary(lm1)


Call:
lm(formula = logenergy ~ logmass + Type, data = case1002)

Residuals:
    Min      1Q  Median      3Q     Max
-0.2322 -0.1220 -0.0364  0.1257  0.3446

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)                 -1.5764     0.2872   -5.49  5.0e-05
logmass                      0.8150     0.0445   18.30  3.8e-12
Typenon-echolocating birds   0.1023     0.1142    0.90     0.38
Typeecholocating bats        0.0787     0.2027    0.39     0.70

Residual standard error: 0.186 on 16 degrees of freedom
Multiple R-squared:  0.982,Adjusted R-squared:  0.978
F-statistic:  284 on 3 and 16 DF,  p-value: 4.46e-14
```

Next, we calculate confidence intervals for the coefficients which are interpreted on page 278.

```
> confint(lm1)

                             2.5 % 97.5 %
(Intercept)                 -2.185 -0.967
logmass                      0.721  0.909
Typenon-echolocating birds -0.140  0.344
Typeecholocating bats      -0.351  0.508

> exp(confint(lm1))

                           2.5 % 97.5 %
(Intercept)                0.112   0.38
logmass                    2.056   2.48
Typenon-echolocating birds 0.870   1.41
Typeecholocating bats      0.704   1.66
```

Since the significance of a model depends on which variables are included, the *Sleuth* proposes two other models, one only looking at the type of flying animal and the other allows the three groups to have different straight-line regressions with *mass*. These two models are displayed below and discussed on pages 278-279.

```
> summary(lm(logenergy ~ Type, data=case1002))


Call:
lm(formula = logenergy ~ Type, data = case1002)

Residuals:
    Min      1Q  Median      3Q     Max
-1.8872 -0.3994  0.0236  0.4932  1.5253

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)                   3.396      0.422    8.04  3.4e-07
Typenon-echolocating birds   -0.609      0.488   -1.25  0.22885
Typeecholocating bats        -2.743      0.597   -4.59  0.00026

Residual standard error: 0.845 on 17 degrees of freedom
Multiple R-squared:  0.595,Adjusted R-squared:  0.548
F-statistic: 12.5 on 2 and 17 DF,  p-value: 0.000458

> summary(lm(logenergy ~ Type * logmass, data=case1002))


Call:
lm(formula = logenergy ~ Type * logmass, data = case1002)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-0.2515 -0.1264 -0.0095  0.0812  0.3284

Coefficients:
                                   Estimate Std. Error t value Pr(>|t|)
(Intercept)                          -0.202      1.261   -0.16    0.875
Typenon-echolocating birds           -1.378      1.295   -1.06    0.305
Typeecholocating bats                -1.268      1.285   -0.99    0.341
logmass                               0.590      0.206    2.86    0.013
Typenon-echolocating birds:logmass    0.246      0.213    1.15    0.269
Typeecholocating bats:logmass         0.215      0.224    0.96    0.353

Residual standard error: 0.19 on 14 degrees of freedom
Multiple R-squared:  0.983,Adjusted R-squared:  0.977
F-statistic:  163 on 5 and 14 DF,  p-value: 6.7e-12
```

To construct the confidence bands discussed on page 282 and shown in Display 10.9 (page 283) we used the following code:

```
> pred = predict(lm1, se.fit=TRUE, newdata=data.frame(Type=c("non-echolocating birds", "non-ec
> pred.fit = pred$fit[1]; pred.fit

   1
2.28

> pred.se = pred$se.fit[1]; pred.se

     1
0.0604

> multiplier = sqrt(4*qf(.95, 4, 16)); multiplier

[1] 3.47

> lower = exp(pred.fit-pred.se*multiplier); lower

   1
7.92

> upper = exp(pred.fit+pred.se*multiplier); upper

 1
12
```

```
> # for the other reference points
> pred2 = predict(lm1, se.fit=TRUE, newdata=data.frame(Type=c("non-echolocating bats", "non-ec
> pred3 = predict(lm1, se.fit=TRUE, newdata=data.frame(Type=c("echolocating bats", "echolocati
>
> table10.9 = rbind(c("Intercept estimate", "Standard error"), round(cbind(pred2$fit, pred2$se

  [,1]                 [,2]
  "Intercept estimate" "Standard error"
1 "2.1767"             "0.1144"
2 "3.3064"             "0.0931"
1 "2.2553"             "0.1277"
2 "3.3851"             "0.1759"
```

Next we can assess the model by evaluating the extra sums of squares F-test for testing the equality of intercepts in the parallel regression lines model as shown in Display 10.10 (page 287).

```
> lm2 = lm(logenergy ~ logmass, data=case1002)
> anova(lm2, lm1)

Analysis of Variance Table

Model 1: logenergy ~ logmass
Model 2: logenergy ~ logmass + Type
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     18 0.583
2     16 0.553  2    0.0296 0.43   0.66
```

We can also compare the full model with interaction terms and the reduced model (without interaction terms) with the extra sum of squares F-test as described in Display 10.12 (page 290).

```
> lm3 = lm(logenergy ~ logmass*Type, data=case1002)
> anova(lm3, lm1)

Analysis of Variance Table

Model 1: logenergy ~ logmass * Type
Model 2: logenergy ~ logmass + Type
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     14 0.505
2     16 0.553 -2   -0.0484 0.67   0.53
```

Another way to test the equality of the groups is by using linear combinations which we can attain using the `estimable()` command as follows. These results can be found on page 276 and 289.

```
> require(gmodels)
> estimable(lm1, c(0, 0, -1, 1))

           Estimate Std. Error t value DF Pr(>|t|)
(0 0 -1 1)  -0.0236      0.158   -0.15 16    0.883
```