Solution to Exercise 1 in Chapter 2

BY MAREK RYCHLIK

March 1, 2009

1 Formulation of the problem

Problem 1. A traffic engineering study on traffic delay was conducted at intersetions with signals on urban streets. Three types of traffic signals were utilized in the study:

1. pretimed;
2. semi-actuated;
3. fully-actuated.

Five intersections were used for each type of signal. The measure of traffic delay used in the study was the average stopped time per vehicle at each of the intersections (seconds/vehicle). The data follow:

Pretimed	Semi-actuated	Fully activated
36.6	17.5	15.0
39.2	30.6	10.4
30.4	18.7	18.9
37.1	25.7	10.5
34.1	22.0	15.2

Table 1. Source: W. Reilly, C. Gardner, and J. Kell (1976). A technique for measurement of delay at intersections. *Technical Report FHWA-RD-76-135*, Federal Highway Administration, Office of R & D, Washington, DC.

a) Write the linear statistical model for this study, and explain the model components.

b) State the assumptions necessary for an analysis of variance of the data.

c) Compute the analysis of variance for the data.

d) Compute the least squares mean of the traffic delay and their standard error for each signal type.

e) Compute the 95% confidence interval estimates of the signal type means.

2 The statistical model

Since the emphasis of our solution is the R implementation, we define the model in R in several session fragments, interspersed with minimal comments.

There is one factor. the type of the traffic signal. The factor has three levels. Let us define this factor in R:

```
> TrafficLightType <- as.factor(c("Pretimed", "Semi-actuated", "Fully
  activated"))
> TrafficLightType
  [1] Pretimed        Semi-actuated   Fully activated
  Levels: Fully activated Pretimed Semi-actuated

>
```

The numerical response is the average stopping time. We define tye response directly in R. There are multiple ways to achieve this step, and we chose one using just the most primitive operations: `c`, `rep` and `data.frame`.

```
> Pretimed <- c(36.6, 39.2, 30.4, 37.1, 34.1)
> SemiActivated <- c(17.5, 30.6, 18.7, 25.7, 22.0)
> FullyActivated <- c(15.0, 10.4, 18.9, 10.5, 15.2)
> AverageStoppedTime <- c(Pretimed, SemiActivated, FullyActivated)
> N <- length(Pretimed)
> TrafficLightType <- c(rep("Pretimed", N), rep("Semi-activated",N), rep("Fully
  activated", N))
> StudyData <- data.frame(TrafficLightType, AverageStoppedTime)
> StudyData

     TrafficLightType AverageStoppedTime
  1          Pretimed               36.6
  2          Pretimed               39.2
  3          Pretimed               30.4
  4          Pretimed               37.1
  5          Pretimed               34.1
  6    Semi-activated               17.5
  7    Semi-activated               30.6
  8    Semi-activated               18.7
  9    Semi-activated               25.7
  10   Semi-activated               22.0
  11  Fully activated               15.0
  12  Fully activated               10.4
  13  Fully activated               18.9
  14  Fully activated               10.5
  15  Fully activated               15.2

>
```

Thus, the statistical model for this study is:

$$\text{AverageStoppedTime}_{ij} = \overline{\text{TrafficLightType}_i} + \text{Error}_{ij}$$

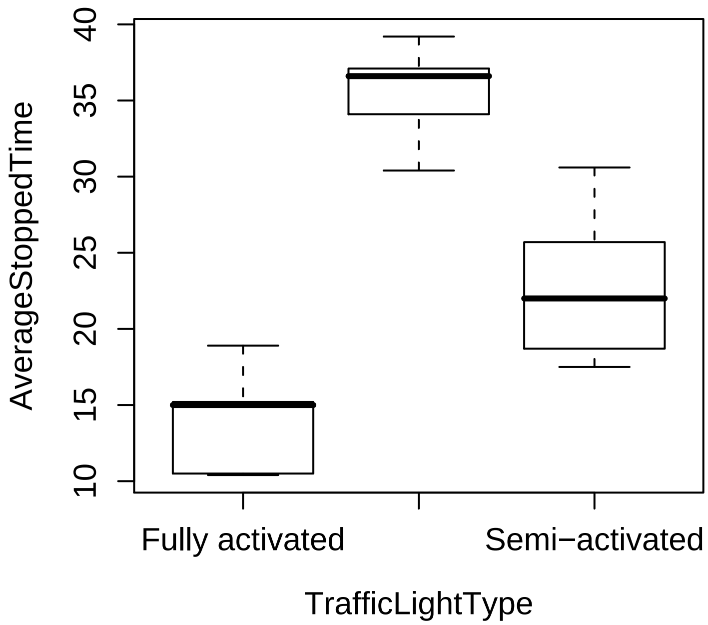
In R the model is expressed as a formula:

```
> fmla <- AverageStoppedTime ~ TrafficLightType
> fmla

  AverageStoppedTime ~ TrafficLightType

>
```

3 Plot of the data

TEXMACS allows for easy integration of graphics with the document. This is how it is done:

```
> X11(pointsize=6);plot(StudyData);v()
```



>

We note that the basic mechanics of incorporating R graphics in TeXmacs documents is to do the plotting as usual, except at the end we need to call function `v()`. This function is not part of standard R distribution, but added by TeXmacs when it launches an R session. The call to `X11(pointsize=6)` is a minor adjustment which reduces the font size used in the plot. Without it, the graphics appear truncated.

4 Assumptions

The usual assumptions of analysis of variance apply. Hence,

i. We assume that the experimental design is a completely randomized design.

ii. We assume that the average stopping time is normally distributed. This, of course, can only be approximately true due to the fact that time is always positive.

iii. We note that the stopping time for an individual vehicle arriving at an intersection at random time would be modeled a uniform distribution perhaps with some Gaussian noise added. Hence, the sample of cards used to compute each individual mean in the table should be large enough to ensure that the Central Limit Theorem is applicable. We note that the average of 6 normal distributions is approximately normal for most practical applications. As we do not know the sample size, we can only hope that the researchers used a reasonable sample size.

iv. We assume that each measurement is a mean of a larger number of cars passing through an intersection and a pre-determined number of cars were in each experimental unit to ensure parity of information.

v. We assume that the variance of the average stopping time does not depend on the intersection or traffic light type.

5 Computation of analysis of variance (ANOVA)

We use the R function `aov` to compute the analysis of variance table for this example, using the defaults for all arguments except the first two (formula and data).

```
> aov <- aov(fmla, StudyData)
> summary(aov)

                 Df  Sum Sq Mean Sq F value    Pr(>F)
TrafficLightType  2 1164.76  582.38  32.992 1.328e-05 ***
Residuals        12  211.83   17.65
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.1 âĂŸ
âĂŹ 1

>
```

We see that the we should reject H_0 and accept the alternative hypothesis that the means a different, with significance level of approximately 99.5%.

6 Computation of the least square means and standard errors

6.1 The least squares means

The table returned by `aov` contains both answers.

```
> aov$coefficients

                 (Intercept)       TrafficLightTypePretimed
                       14.00                          21.48
TrafficLightTypeSemi-activated
                        8.90

> levels(StudyData$TrafficLightType)

[1] "Fully activated" "Pretimed"        "Semi-activated"

> coefficients(aov)

                 (Intercept)       TrafficLightTypePretimed
                       14.00                          21.48
TrafficLightTypeSemi-activated
                        8.90

>
```

We note that these are group means of the treatment groups. However, we should look carefully at the order of the levels of the factor `TrafficLightType`. R auto-ordered them starting with `Fully activated`. The named value `(Intercept)` is the mean of the group corresponding to the first treatment level. The named value `TrafficLightTypePretimed` plus the intercept is the group mean of the second treatment group, `Pretimed`, and finally, the third value corresponds to the level `Semi-activated`.

We note that the generic function `coefficients` can be used to extract the coefficients from the structure returned by `aov`.

We note that the generic function `model.tables` also can be used to compute the treatment means. The *effects* are simply differences between the group means and the overall mean.

```
> tbl <- model.tables(aov, type="means", se=T)
> tbl
```

```
Tables of means
Grand mean

24.12667

 TrafficLightType
TrafficLightType
Fully activated        Pretimed  Semi-activated
          14.00           35.48           22.90

Standard errors for differences of means
        TrafficLightType
                   2.657
replic.                5
```

```
>
```

We note that the value of `aov$coefficients` depends on the choice of contrasts. This solution assumes that the default setting for the contrasts is used. The default setting is a `contr.default`. Since contrasts appear in Chapter 3, we discuss the subject of contrasts no further.

All essential components returned by `aov` (and some of the less essential ones) are returned by a call to `attributes`:

```
> attributes(aov)
```

```
$names
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "contrasts"     "xlevels"       "call"          "terms"
[13] "model"

$class
[1] "aov" "lm"
```

```
>
```

For many of these components, there exists a generic function to extract or analyze them. The the name of the generic either coincides with or contains the name of the component, and thus it can be located with standard help facilities, such as `help` and `apropos`.

7 Confidence intervals of the means

Confidendce intervals can be computed using the generic function `confint`. Let us review the methods of this generic:

```
> methods(confint)
```

```
[1] confint.default confint.glm*    confint.lm*     confint.nls*

   Non-visible functions are asterisked
```

```
>
```

As we can see, any model fitted with `glm`, `lm` and `nls` can be passed as an argument to `confint`. In particular, the model originated by `aov` can be used as an argument.

We conclude the solution of our excercise by making a call to `confint`. The problem requests confidence intervals for the means, and thus we pass the extra argument `type`. The argument `level` has a default value of 95% and is not required in our situation.

```
> ivals <- confint(aov, level=.95)
> ivals
                                  2.5 %   97.5 %
(Intercept)                    9.906112 18.09389
TrafficLightTypePretimed      15.690368 27.26963
TrafficLightTypeSemi-activated  3.110368 14.68963
> attributes(ivals)
$dim
[1] 3 2

$dimnames
$dimnames[[1]]
[1] "(Intercept)"                    "TrafficLightTypePretimed"
[3] "TrafficLightTypeSemi-activated"

$dimnames[[2]]
[1] "2.5 %"  "97.5 %"
> class(ivals)
[1] "matrix"
>
```

Again, the row labeled `(Intercept)` refers to the first level of the factor, i.e. `Fully activated`. We note that the class of the answer is `matrix` and the rows of the matrix are named after the levels of the applicable factor: `TrafficLightType`.