How R represents sums of squares in its ANOVA output

The question

What are the sums of squares in R's ANOVA table?

Here's a sample ANOVA table:

```
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq  F value    Pr(>F)    
x1         1   512.5   512.5   7.6762   0.03240 *  
x2         1 15669.5 15669.5 234.7021 4.886e-06 ***
x3         1   343.4   343.4   5.1440   0.06384 .  
Residuals  6   400.6    66.8                       
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What are all those "Sum Sq" values?

Test data

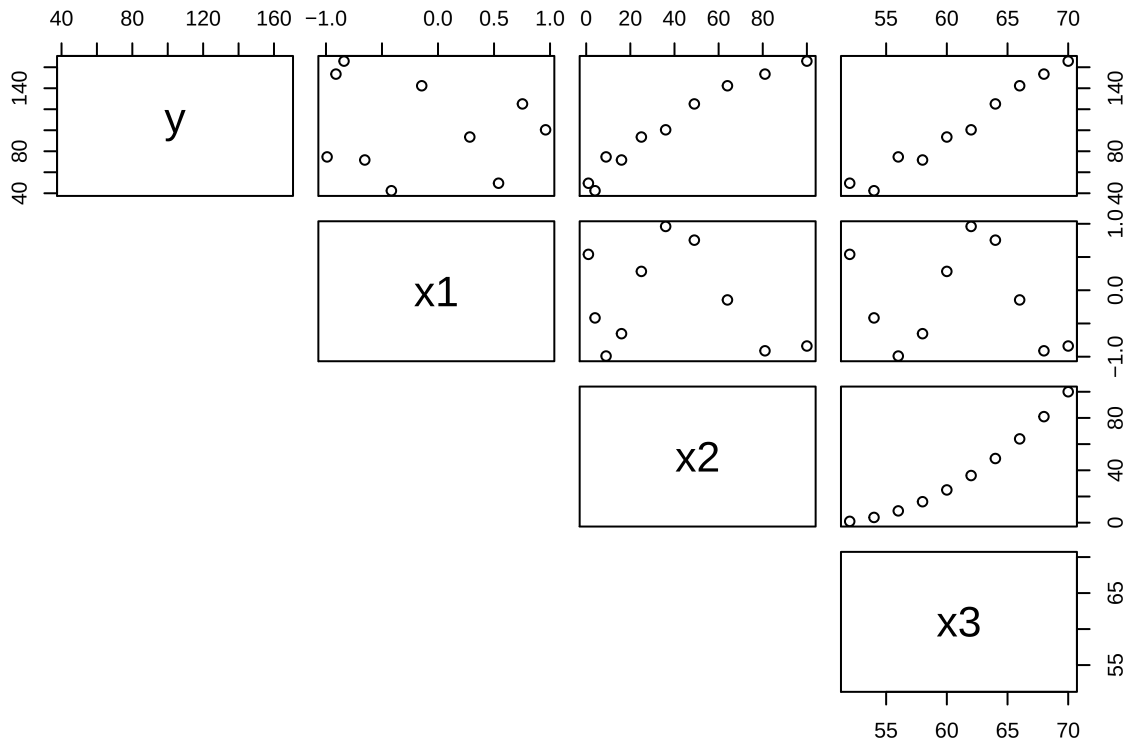
We start by making fake data:

```r
# You can use this to make your own:
#  t = 1:10
#  x1 = cos(t)
#  x2 = t^2
#  x3 = 2 * t + 50
#  e = rnorm(length(t), sd = 10)  # some random variation
#  y = x1 + x2 + x3 +  e
#  values = data.frame(y, x1, x2, x3)

# pre-built list
values = structure(list(
    y = c(49.5584284590433, 42.4098374952543, 74.6177379465143, 71.6750998780002,
          93.5607052659616, 100.411067048661, 125.072059320225, 142.363597693467,
          153.460788703202, 165.812936223002),
    x1 = c(0.54030230586814, -0.416146836547142, -0.989992496600445, -0.653643620863612,
           0.283662185463226, 0.960170286650366, 0.753902254343305, -0.145500033808614,
           -0.911130261884677, -0.839071529076452),
    x2 = c(1, 4, 9, 16, 25, 36, 49, 64, 81, 100),
    x3 = c(52, 54, 56, 58, 60, 62, 64, 66, 68, 70)),
    class = "data.frame",
    row.names = c(NA, -10L)
)
```

We look at our model:

```r
pairs(values, lower.panel = NULL)
```




```r
model = lm(y ~ x1 + x2 + x3, data = values)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3, data = values)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.273  -5.419   1.404   5.053   9.940
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -205.4710   114.4745  -1.795   0.1228
## x1             1.1407     4.0350   0.283   0.7869
## x2             0.4191     0.3790   1.106   0.3112
## x3             4.7769     2.1062   2.268   0.0638 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.171 on 6 degrees of freedom
## Multiple R-squared:  0.9763, Adjusted R-squared:  0.9645
## F-statistic: 82.51 on 3 and 6 DF,  p-value: 2.874e-05
```

On to the ANOVA table, but first, let's agree on some notation.

Notation

We shall use $SS(y \sim x_1 + ... + x_k)$ to mean the SS_R calculated when fitting $y \sim x_1 + \cdots + x_k$ using multiple linear regression. See p. 86 of the textbook for additional details for how to calculate this value.

The individual rows

Let's break the table down line by line:

```
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq  F value    Pr(>F)

                .----- This one is (SS_R for y ~ x1)
               .                     ^-- 512.5
x1         1   512.5   512.5   7.6762   0.03240 *

               .---------- This one is (SS_R for y ~ x1 + x2) - (SS_R for y ~ x1).
               .                        ^-- 16181.964           ^-- 512.5
x2         1 15669.5 15669.5 234.7021 4.886e-06 ***

                .----------   (SS_R for (y ~ x1 + x2 + x3)) - (SS_R for y ~ x1 + x2))
                .                        ^-- 16525.392          ^-- 16181.964
x3         1   343.4   343.4   5.1440   0.06384 .
Residuals  6   400.6    66.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Hence, you can calculate the "Sum Sq" values for individual models by adding your way up:

```
SS_R (y ~ x1 + x2 + x3)
  = (Sum Sq value for x3) + (Sum Sq values for preceding regressors)
```

Hence,

$$SS_R(y \sim x_1, ..., x_k) = \sum_{i=1}^{k} [\text{ Sum Sq in R output for } x_i].$$

Alternate interpretation

Note also that the "Sum Sq" rows have another interpretation:

$$\text{Sum Sq : } x_i = SS_R(\beta_k|\beta_0, \beta_1, \ldots, \beta_{k-1}) = SS(y \sim x_1 + \cdots + x_k) - SS(y \sim x_1 + \ldots x_{k-1}).$$

Order matters

Note that the order of rows matter because adding up rows gives you the SS for the set of regressors whose rows you included in the sum. Thus, if you fit `lm(y ~ x1 + x2 + x3)`, as you add rows, you get in turn:

$$
\begin{aligned}
\text{row}_1 &: SS(y \sim x_1) \\
\text{row}_1 + \text{row}_2 &: SS(y \sim x_1 + x_2) \\
\text{row}_1 + \text{row}_2 + \text{row}_3 &: SS(y \sim x_1 + x_2 + x_3) \\
\text{row}_1 + \text{row}_2 + \text{row}_3 + \text{row}_{Residuals} &: SS_T
\end{aligned}
$$

If you want $SS(y \sim x_2)$ or $SS(y \sim x_1 + x_3)$, you will not be able to simply add the rows in order.

SS for the F statistic

Suppose we want to consider removing one or more predictors from the full model:

$$y \sim \frac{\text{reduced}}{x_1 + x_2} + \frac{\text{can we remove?}}{x_3}$$

Then we want to calculate our F test. We will need $SS(\beta_2|\beta_1) = SS(\beta) - SS(\beta_1)$. Using our notational convention, we write this as follows:

$$
\begin{aligned}
SS(\beta) &= SS(y \sim x_1 + x_2 + x_3) \\
SS(\beta_1) &= SS(y \sim x_1 + x_2)
\end{aligned}
$$

Seen in this light, you can tell that we can simply pick the "Sum Sq" value for x_3 as it already has the difference between these two. However, if we had more predictors that we're thinking of removing, we'd have to compute our sum like this:

$$
\begin{aligned}
&SS(\beta \text{ for } y \sim x_1 + \cdots + x_k) - SS(\beta \text{ for } y \sim x_1 + \cdots + x_{k-r}) \\
&= \textstyle\sum_{i=1}^{k} [\text{ Sum Sq in R output for } x_i] - \sum_{i=1}^{k-r} [\text{ Sum Sq in R output for } x_i]
\end{aligned}
$$

This only works if you have ordered the predictors you want to remove to the end of the fitted model in R.

What about β_0?

Note: This section is speculative. Do you agree or disagree?

Suppose you want to fit the response to a line without a regressor, say $y \sim 1$. Then there's a single coefficient β_0 and $\hat{y}$ is a constant value. In fact, with only β_0, we're fitting $\hat{y} = \bar{y}$ as our model!

In this case, without regressors, $SS_R(\beta_0) = SS(y \sim 1) = 0$ (using our notation). Also, $SS_{Res} = SS_T$. Note that SS_T is the same with or without regressors because it only depends on the y_i levels.

We can observe how R handles this situation:

```
tibble::tibble(
  "Using R" = anova(lm(values$y ~ 1))$"Sum Sq",
  "Using our formula" = sum((values$y - mean(values$y))^2)
)
```

```
## # A tibble: 1 x 2
##   `Using R` `Using our formula`
##       <dbl>               <dbl>
## 1    16926.              16926.
```

```
anova(lm(values$y ~ 1))
```

```
## Analysis of Variance Table
##
## Response: values$y
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals  9  16926  1880.7
```

What about SS_T?

We may calculate SS_T from R's ANOVA output:

```
model_anova = anova(model)
sum(model_anova$`Sum Sq`) # includes SS_{Res}
```

```
## [1] 16925.97
```

In other words, add all the predictor "Sum Sq" and then add the Residual Sum Sq to boot, and you get SS_T.