# Background

In the linear regression context, it is common to use the ***F-test*** to test whether a proposed regression model fits the data well. Say we have predictors, and we are comparing the model fit for

1. Linear regression where  are allowed to vary freely but  are fixed at zero, vs.
2. Linear regression where  are allowed to vary freely.

( is some fixed parameter.) We call the first model the ***“restricted model”***, and the secondthe ***“full model”***. We say that these models are ***nested*** since the second model is a superset of the first. In the hypothesis testing framework, comparing the model fits would be testing



If we let  and  denote the residual sum of squares under the restricted and full models respectively, and and  denote the degrees of freedom under the restricted and full models respectively, then under the null hypothesis, the ***F-statistic***



has the  distribution. If is large, the null hypothesis is rejected and we conclude that the full model fits the data better than the restricted model. (See Reference 1 for more details.)

# The problem

In R, we can use the anova() function to do these comparisons. In the following code, we compare the fits of mpg ~ wt (full model) vs. mpg ~ 1 (restricted model, intercept only):

data(mtcars)

mod1 <- lm(mpg ~ 1, data = mtcars mod2 <- lm(mpg ~ wt, data = mtcars

mod3 <- lm(mpg ~ wt + hp, data = mtcars

anova(mod1, mod2)

# Analysis of Variance Table #

# Model 1: mpg ~ 1 # Model 2: mpg ~ wt

# Res.Df RSS Df Sum of Sq F Pr(>F)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | 1 | 31 | 1126.05 |  | | |
| # | 2 | 30 | 278.32 | 1 | 847.73 | 91.375 1.294e-10 \*\*\* |
| # | --- |  |  |  |  |  |

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

From the table, we see that the -statistic is equal to 91.375.

The anova() function is pretty powerful: if we have a series of nested models, we can test them all at once with one function call. For example, the code below computes the -statistics for mod2 vs. mod1 and mod3 vs. mod2:

anova(mod1, mod2, mod3)

# Analysis of Variance Table #

# Model 1: mpg ~ 1 # Model 2: mpg ~ wt

# Model 3: mpg ~ wt + hp

# Res.Df RSS Df Sum of Sq F Pr(>F) # 1 31 1126.05

# 2 30 278.32 1 847.73 126.041 4.488e-12 \*\*\*

# 3 29 195.05 1 83.27 12.381 0.001451 \*\* # ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

***But wait: the -statistic for* mod2 vs. mod1 has changed!** It was previously 91.375, and now it is 126.041. ***What happened?***

# Resolution (Part 1)

The answer lies in a paragraph within the help file for **anova.lm()** (emphasis mine):

Optionally the table can include test statistics. Normally the F statistic is most appropriate, **which compares the mean square for a row to the residual sum of squares for the largest model considered**. If scale is specified chi-squared tests can be used. Mallows’ *Cp* statistic is the residual sum of squares plus twice the estimate of *sigma^2* times the residual degrees of freedom.

In other words, the denominator of the F-statistic is based on the largest model in the anova() call. We can verify this with the computations below. In anova(mod1, mod2)<, the denominator depends on the RSS and Res.Df values for model 2; in anova(mod1, mod2, mod3), in depends on the RSS and Res.Df values for model 3.

((1126.05 - 278.32) / (31 - 30)) / (278.32 / 30)

# [1] 91.37647

((1126.05 - 278.32) / (31 - 30)) / (195.05 / 29)

# [1] 126.0403

# Resolution (Part 2)

***Why would anova() determine the denominator in this way?*** I think the reason lies in what the F-statistic is trying to compare (see Reference 2 for details). The F-statistic is comparing two different estimates of the variance, and the estimate in the denominator is akin to the typical variance estimate we get from the residuals of a regression model. In our example above, one

F-statistic used the residuals from mod2, while the other used the residuals from mod3.

***Which F-statistic should you use in practice?*** I think this might depend on your data analysis pipeline, but my gut says that the F-statistic from the ***anova()*** call with just 2 models is probably the one you want to use. It’s a lot easier to interpret and understand.

I haven’t seen any discussion on this in my internet searches, so I would love to hear views on what one should do in practice!