Exploration librairie swirl : Overfitting and Underfitting

> swirl()

Welcome to swirl! Please sign in. If you've been here before, use the same name as you did then. If you are new, call yourself something unique.

What shall I call you? ***jlbellier***

Would you like to continue with one of these lessons?

1: Regression Models Least Squares Estimation
2: No. Let me start something new.

Selection: ***2***

Please choose a course, or type 0 to exit swirl.

1: Regression Models
2: Statistical Inference
3: Take me to the swirl course repository!

Selection: ***1***

Please choose a lesson, or type 0 to return to course menu.

1: Introduction
2: Residuals
3: Least Squares Estimation
4: Residual Variation
5: Introduction to Multivariable Regression
6: MultiVar Examples
7: MultiVar Examples2
8: MultiVar Examples3
9: Residuals Diagnostics and Variation
10: Variance Inflation Factors
11: Overfitting and Underfitting
12: Binary Outcomes
13: Count Outcomes

Selection: ***11***

0%

Overfitting and Underfitting. (Slides for this and other Data Science courses may be found at github https://github.com/DataScienceSpecialization/courses. If you care to use them, they must be downloaded as a zip file and viewed locally. This lesson corresponds to Regression_Models/02_04_residuals_variation_diagnostics.)

=== ***4%***

The Variance Inflation Factors lesson demonstrated that including new variables will increase standard errors of coefficient estimates of other, correlated regressors. Hence, we don't want to idly throw variables into the model. On the other hand, omitting variables results in bias in coefficients of regressors which are correlated with the omitted ones. In this lesson we demonstrate the effect of omitted variables and discuss the use of ANOVA to construct parsimonious, interpretable representations of the data.

======= ***7%***

First, I would like to illustrate how omitting a correlated regressor can bias estimates of a coefficient. The relevant source code is in a file named fitting.R which I have copied into your working directory and tried to display in your source code editor. If I've failed to display it, you should open it manually.

========== ***11%***

Find the function simbias() at the top of fitting.R. Below the comment labeled Point A three regressors, x1, x2, and x3, are defined. Which of these two are correlated?

1: x2 and x3
2: x1 and x2
3: x1 and x3

Selection: ***2***

You are really on a roll!

========== ***15%***

Within simbias() another function, f(n), is defined. It forms a dependent variable, y, and at Point C returns the coefficient of x1 as estimated by two models, y ~ x1 + x2, and y ~ x1 + x3. One regressor is missing in each model. In the expression for y (Point B,) what is the actual coefficient of x1?

1: 1
2: 1/sqrt(2)
3: 0.3

Selection: ***1***

All that practice is paying off!

========== ***19%***

At Point D in simbias() the internal function, f(), is applied 150 times and the results returned as a 2x150 matrix. The first row of this matrix contains independent estimates of x1's coefficient in the case that x3, the regressor uncorrelated with x1, is omitted. The second row contains estimates of x1's coefficient when the correlated regressor, x2, is omitted. Use simbias(), accepting the default argument, to form these estimates and store the result in a variable called x1c. (The default argument just guarantees a nice histogram, in a figure to follow.)

> ***x1c <- simbias()***

You got it right!

========== **22%**

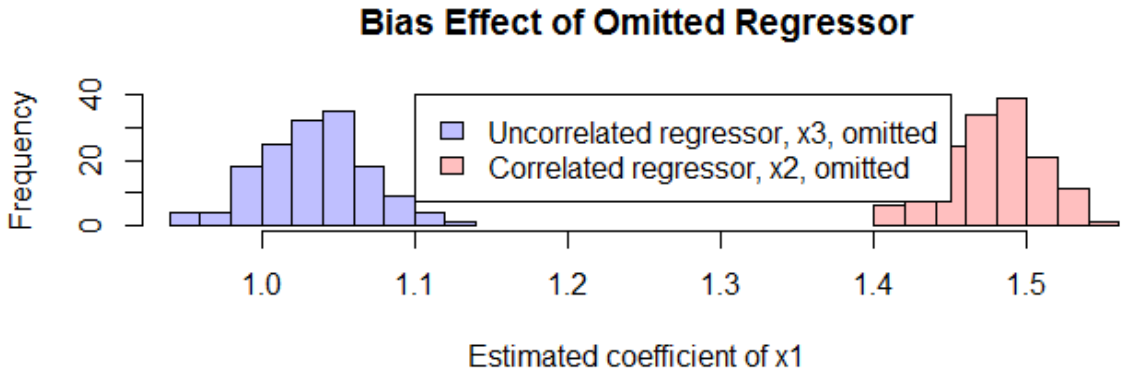
The actual coefficient of x1 is 1. Having been warned that omitting a correlated regressor would bias estimates of x1's coefficient, we would expect the mean estimate of x1c's second row to be farther from 1 than the mean of x1c's first row. Using apply(x1c, 1, mean), find the means of each row.

```
> apply(x1c,1,mean)
    x1      x1
1.034403 1.476944
```

You are quite good my friend!

========== **26%**

Histograms of estimates from x1c's first row (blue) and second row (red) are shown. Estimates from the second row are clearly more than two standard deviations from the correct value of 1, and the bias due to omitting the correlated regressor is evident. (The code which produced this figure is incidental to the lesson, but is available as the function x1hist(), at the bottom of fitting.R.)

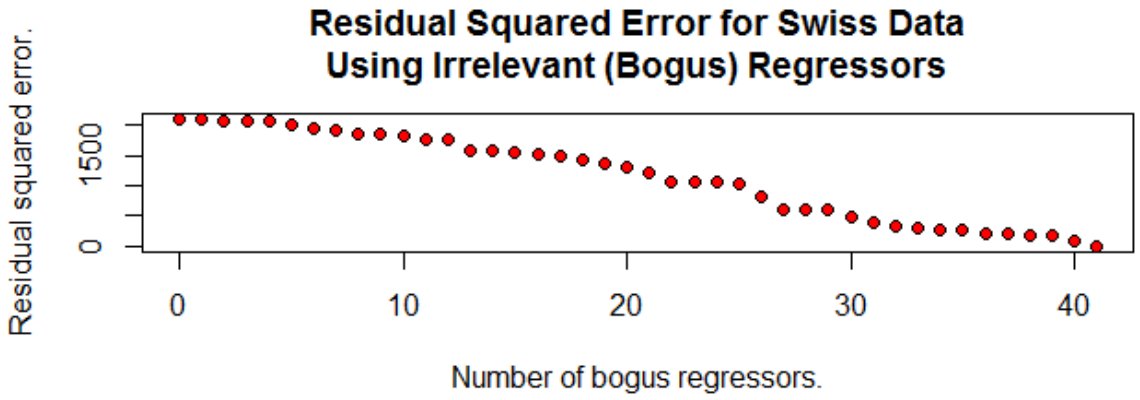


========== **30%**

Adding even irrelevant regressors can cause a model to tend toward a perfect fit. We illustrate this by adding random regressors to the swiss data and regressing on progressively more of them. As the number of regressors approaches the number of data points (47), the residual sum of squares, also known as the deviance, approaches 0. (The source code for this figure can be found as function bogus() in fitting.R.

========== **33%**

In the figure, adding random regressors decreased deviance, but we would be mistaken to believe that such decreases are significant. To assess significance, we should take into account that adding regressors reduces residual degrees of freedom. Analysis of variance (ANOVA) is a useful way to quantify the significance of additional regressors. To exemplify its use, we will use the swiss data.



========== **37%**

Recall that the Swiss data set consists of a standardized fertility measure and socioeconomic indicators for each of 47 French-speaking provinces of Switzerland in 1888. Fertility was thought to depend on an intercept and five factors denoted as Agriculture, Examination, Education, Catholic, and Infant Mortality. To begin our ANOVA example, regress Fertility on Agriculture and store the result in a variable named fit1.

> ***fit1 <- lm(Fertility~Agriculture,swiss)***

You are doing so well!

========== **41%**

Create another model, named fit3, by regressing Fertility on Agriculture and two additonal regressors, Examination and Education.

> ***fit3 <- lm(Fertility~Agriculture+Examination+Education,swiss)***

You are doing so well!

========== **44%**

We'll now use anova to assess the significance of the two added regressors. The null hypothesis is that the added regressors are not significant. We'll explain in detail shortly, but right now just apply the significance test by entering anova(fit1, fit3).

> ***anova(fit1,fit3)***

Analysis of Variance Table

Model 1: Fertility ~ Agriculture
Model 2: Fertility ~ Agriculture + Examination + Education

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	45	6283.1			
2	43	3180.9	2	3102.2	20.968 4.407e-07 ***

---
*Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1*

You are amazing!

========== 48%

*The three asterisks, ***, at the lower right of the printed table indicate that the null hypothesis is rejected at the 0.001 level, so at least one of the two additional regressors is significant. Rejection is based on a right-tailed F test, Pr(>F), applied to an F value. According to the table, what is that F value?*

1: 3102.2
2: 20.968
3: 45

*Selection: **2***

You nailed it! Good job!

========== 52%

An F statistic is a ratio of two sums of squares divided by their respective degrees of freedom. If the two scaled sums are independent and centrally chi-squared distributed with the same variance, the statistic will have an F distribution with parameters given by the two degrees of freedom. In our case, the two sums are residual sums of squares which, as we know, have mean zero hence are centrally chi-squared provided the residuals themselves are normally distributed. The two relevant sums are given in the RSS (Residual Sum of Squares) column of the table. What are they?

1: 45 and 43
2: 2 and 3102.2
3: 6283.1 and 3180.9

*Selection: **3***

Keep up the great work!

========== 56%

R's function, deviance(model), calculates the residual sum of squares, also known as the deviance, of the linear model given as its argument. Using deviance(fit3), verify that 3180.9 is fit3's residual sum of squares. (Of course, fit3 is called Model 2 in the table.)

*> **deviance(fit3)***
[1] 3180.925

You are doing so well!

========== 59%

In the next several steps, we will show how to calculate the F value, 20.968, which appears in the table printed by anova(). We'll begin with the denominator, which is fit3's residual sum of squares divided by its degrees of freedom. Fit3 has 43 residual degrees of freedom. This figure is obtained by subtracting 4, the the number of fit3's predictors (the 3 named and the intercept,) from 47, the number of samples in swiss. Store the value of deviance(fit3)/43 in a variable named d.

*> **d <- deviance(fit3)/43***

Great job!

========== 63%

The numerator is the difference, deviance(fit1)-deviance(fit3), divided by the difference in the residual degrees of freedom of fit1 and fit3, namely 2. This calculation requires some theoretical justification which we omit, but the essential idea is that fit3 has 2 predictors in addition to those of fit1. Calculate the numerator and store it in a variable named n.

```
> n <- (deviance(fit1)-deviance(fit3))/2
```

That's correct!

========== 67%

Calculate the ratio, n/d, to show it is essentially equal to the F value, 20.968, given by anova().

```
> n/d
[1] 20.96783
```

Keep up the great work!

========== 70%

We'll now calculate the p-value, which is the probability that a value of n/d or larger would be drawn from an F distribution which has parameters 2 and 43. This value was given as 4.407e-07 in the column labeled Pr(>F) in the table printed by anova(), a very unlikely value if the null hypothesis were true. Calculate this p-value using pf(n/d, 2, 43, lower.tail=FALSE).

```
> pf(n/d,2,43,lower.tail = FALSE)
[1] 4.406913e-07
```

You are quite good my friend!

========== 74%

Based on the calculated p-value, a false rejection of the null hypothesis is extremely unlikely. We are confident that fit3 is significantly better than fit1, with one caveat: analysis of variance is sensitive to its assumption that model residuals are approximately normal. If they are not, we could get a small p-value for that reason. It is thus worth testing residuals for normality. The Shapiro-Wilk test is quick and easy in R. Normality is its null hypothesis. Use shapiro.test(fit3$residuals) to test the residual of fit3.

```
> shapiro.test(fit3$residuals)

        Shapiro-Wilk normality test

data:  fit3$residuals
W = 0.97276, p-value = 0.336
```

Excellent job!

========== 78%

The Shapiro-Wilk p-value of 0.336 fails to reject normality, supporting confidence in our analysis of variance. In order to illustrate the use of anova() with more than two models, I have constructed fit5 and fit6 using the first 5 and all 6 regressors (including the intercept) respectively. Thus fit1, fit3, fit5, and fit6 form a nested sequence of models; the regressors of one are included in those of the next. Enter anova(fit1, fit3, fit5, fit6) at the R prompt now to get the flavor.

*> **anova(fit1, fit3, fit5, fit6)***
Analysis of Variance Table

Model 1: Fertility ~ Agriculture
Model 2: Fertility ~ Agriculture + Examination + Education
Model 3: Fertility ~ Agriculture + Examination + Education + Catholic
Model 4: Fertility ~ Agriculture + Examination + Education + Catholic +
Infant.Mortality

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	45	6283.1				
2	43	3180.9	2	3102.19	30.2107	8.638e-09 ***
3	42	2513.8	1	667.13	12.9937	0.0008387 ***
4	41	2105.0	1	408.75	7.9612	0.0073357 **

---
*Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1*

That's correct!

========== 81%

It appears that each model is a significant improvement on its predecessor. Before ending the lesson, let's review a few salient points.

========== 85%

Omitting a regressor can bias estimation of the coefficient of certain other regressors. Which ones?

1: Correlated regressors
2: Uncorrelated regressors

*Selection: **1***

You nailed it! Good job!

========== 89%

Including more regressors will reduce a model's residual sum of squares, even if the new regressors are irrelevant. True or False?

1: True
2: False
3: It depends on circumstances.

*Selection: **3***

Not exactly. Give it another go.
It doesn't depend on circumstances.

1: It depends on circumstances.
2: True
3: False

Selection: ***2***

You got it right!
========== 93%
When adding regressors, the reduction in residual sums of squares should be tested for significance above and beyond that of reducing residual degrees of freedom. R's anova() function uses an F-test for this purpose. What else should be done to insure that anova() applies?

1: Model residuals should be tested for normality.
2: The residuals should be tested for having zero means.
3: Regressors should be tested for normality.

Selection: ***2***

Keep trying!
F-tests are sensitive to the assumption of normality.

1: Model residuals should be tested for normality.
2: The residuals should be tested for having zero means.
3: Regressors should be tested for normality.

Selection: ***1***

Excellent job!
========== 96%
That completes the lesson on underfitting and overfitting.
========== 100%