Standard Errors in Multivariate Regression

Tom Glinnan

February 2022

Connecting to MT Material on Regression Anatomy

To make things simple, we're gong to focus on the homoscedastic case: where $Var(\varepsilon) = \sigma^2 I$. Have a look at the other GitHub document (on Robust SEs, from the MT material) for an explanation of other ways to think about errors. We're also going to suppose all of the X are fixed, though of course you could look at the case when X is stochastic by the usual procedure (fix X, then use the LIE)

To link to the MT material, we have to explain the different notation we use. If the true model is:

$$y = X_1\beta_1 + X_2\beta_2 + \varepsilon$$

Then in MT we referred to the residuals from regressing y or X_1 on X_2 as $\tilde{y}$ and $\tilde{X}_1$, and gave the following formulas (in lectures 11 and 12):

$$\hat{\beta}_1 = \frac{\hat{Cov}(\tilde{y}, \tilde{X}_1)}{\hat{Var}(\tilde{X}_1)} \qquad \text{and} \qquad Var(\hat{\beta}_1) = \frac{Var(\varepsilon_i)}{n\hat{Var}(\tilde{X}_1)}$$

Now for the LT. We know that we can write the residuals in matrix language as $M_2 y$ and $M_2 X_1$, and we showed that:

$$\hat{\beta}_1 = (X_1' M_2 X_1)^{-1} X_1' M_2 y \qquad \text{and} \qquad Var(\hat{\beta}_1) = \sigma^2 (X_1' M_2 X_1)^{-1}$$

If you look at this enough, you can see that we have used matrices to generalise the regression anatomy formula from MT so we can have more than two regressors. To clarify the similarities:

- The FWL theorem says that we can get $\hat{\beta}_1$ by first multiplying by M_2:

$$M_2 y = M_2 X_1 \beta_1 + \underbrace{M_2 X_2}_{0} \beta_2 + M_2 \varepsilon$$

  then doing OLS on the remaining part: $M_2 y = M_2 X_1 \beta_1 + M_2 \varepsilon$, or in MT notation $\tilde{y} = \tilde{X}_1 \beta_1 + u$. When β_1 is a scalar the OLS estimator here is exactly $\hat{Cov}(\tilde{y}, \tilde{X}_1)/\hat{Var}(\tilde{X}_1)$, which is the MT formula above. However, the LT formula is more general as it allows for β_1 to be a vector (and so it can deal with multivariate regressions). Writing out OLS explicitly in this case

$$\hat{\beta}_1 = ((M_2 X_1)' M_2 X_1)^{-1} (M_2 X_1)' M_2 y$$

and simplifying using $(AB)' = B'A'$ and the symmetric-ness and idempotency of M_2 gets to the LT formula

- By definition, $\sigma^2 = Var(\varepsilon_i)$. As $M_2X_1 = \tilde{X}_1$, $X_1'M_2X_1 = X_1'M_2'M_2X_1 = \tilde{X}_1'\tilde{X}_1$. To connect to the MT material, suppose that $\tilde{X}_1$ is a vector rather than a matrix (which means that β_1 is a scalar rather than a vector). Then $\tilde{X}_1'\tilde{X}_1$ is actually the sum of the squared components of $\tilde{X}_1$; if we call $\hat{u} = \tilde{X}_1$, then $\tilde{X}_1'\tilde{X}_1 = \sum \hat{u}_i^2$. As the u_i are the residuals from a regression they must have 0 mean, and so $1/n \sum u_i = 0$. Therefore the sample variance $\hat{Var}(u_i) := \frac{1}{n}\sum u_i^2 - (\frac{1}{n}\sum u_i)^2 = \frac{1}{n}\sum u_i^2 = \frac{1}{n}u'u = \frac{1}{n}\tilde{X}_1'\tilde{X}_1$. Because of this, we can write $X_1'M_2X_1 = n\hat{Var}(\tilde{X}_1)$ in the special case where $\tilde{X}_1$ is a vector rather than a matrix, so the LT formula generalises the MT one

Near Multicollinearity: PS3 Q3c

This is a very similar exercise to what we just did above, which gives another interpretation of the variance of $\hat{\beta}_1$. Starting with $X_1'M_2X_1$ we do the same as before: $X_1'M_2X_1 = X_1'M_2'M_2X_1 = \tilde{X}_1'\tilde{X}_1$ (called $\hat{v}$ in the solutions). As we know, $\tilde{X}_1 = M_2X_1$ is the residual in the regression

$$X_1 = X_2\gamma + v$$

By definition, the R^2 of this regression is given by $1 - SSR/SST$, where $SSR = \hat{v}'\hat{v}$ is the sum of the squared residuals, and SST is the total sum of squares[1] $\sum(x_{i1} - \bar{x}_1)^2$. By re-arranging, $SSR = \hat{v}'\hat{v} = SST(1 - R^2)$, and so:

$$Var(\hat{\beta}_1) = \sigma^2[SST(1 - R^2)]^{-1} = \frac{\sigma^2}{SST(1 - R^2)}$$

which is what we wanted to show

Interpretation

We haven't done any magic here: essentially all we've done is re-written the $\hat{Var}(\tilde{X}_1)$ from MT as $SST(1 - R^2)$. However this second expression gives us a view about multicollinearity and how it affects the variance of $\hat{\beta}_1$. Remember that $0 \leq R^2 \leq 1$. Let's look at some cases:

- $R^2 = 0$ means that X_2 has no explanatory power to (linearly) predict X_1. This is what it means for X_1 and X_2 to be *orthogonal* (or to put it in MT language, it means that the coefficient in the auxiliary regression is 0, so there is no OVB). As X_2 is in a sense 'irrelevant' to X_1, intuitively the variance of β_1 in the short and long regressions should be the same (as it doesn't matter whether you include X_2 or not). This turns out to be the case: the variance of $\hat{\beta}_1$ is $\sigma^2/SST = \sigma^2/\sum(x_{i1} - \bar{x}_1)^2$. This last term is exactly $n\hat{Var}(X_1)$, which is what we would find if we ran a short regression just of y on X_1
- $R^2 = 1$ means that X_1 can be perfectly linearly predicted from X_2. This makes $Var(\hat{\beta}_1) = \infty$. This captures the idea that if we have $R^2 = 1$ then we have multicollinearity, so we can't actually solve the OLS problem

[1] As x_1 is on the left hand side of this regression

- The interesting part is when we have $R^2 \sim 1$, known as *near multicollinearity*. In other words, the X_2 can predict X_1 very well, but not quite perfectly. We can see here that this makes $Var(\hat{\beta}_1)$ very large, blowing up our standard errors. Intuitively: if X_1 and X_2 capture 'the same kind of' thing (such as reading and math scores) then it's very difficult for OLS to assign values to $\hat{\beta}_1$ and $\hat{\beta}_2$. Formally, if we had got slightly different data then we would potentially assign very different values to $\hat{\beta}_1$ and $\hat{\beta}_2$, as this choice is very sensitive to the data we have. This is exactly what it means for the variance of $\hat{\beta}_1$ to be large

Near Multicollinearity another way

This is **completely off-syllabus**, but I mentioned to a couple of you in office hours last week a different way to look at multicollinearity that's much more linear algebra-heavy which you might be interested in. For intuition, I would stick with the stuff done above rather than thinking of this

Instead of looking at the partitioned regression formula, another way is to look at the whole regression in one go. Under homoscedasticity, we know if $y = X'\beta + \varepsilon$ then $\hat{\beta} = (X'X)^{-1}X'y$, with $Var(\hat{\beta}) = \sigma^2(X'X)^{-1}$. One thing we can do is diagonalise $X'X$ (as it's symmetric and psd) to give PDP'. We know that a condition for $X'X$ to be invertible is that $det(X'X) \neq 0$. As the determinant of a matrix is the product of its eigenvalues, and all eigenvalues of $X'X$ are ≥ 0 as it is psd, saying that $det(X'X) = 0$ is the same as saying that the product of its eigenvalues is 0, and so its smallest eigenvalue is 0. Therefore another way to talk about near multicollinearity (instead of the partitioned regression way above) is a situation where the smallest eigenvalue of $X'X$ is close to 0, and so the biggest eigenvalue of $(X'X)^{-1}$ is large[2]

Why does this lead to $Var(\hat{\beta})$ being large? Intuitively, near multicollinearity is a situation where $(X'X)^{-1}$ has big eigenvalues, so for any z (thinking of the matrix as a linear transformation) $(X'X)^{-1}z$ must be stretching z a lot (at least in the directions where its eigenvectors have big eigenvalues). Thinking now of $(X'X)^{-1}$ as a table, it must contain big numbers if it's going to stretch a lot, and so $Var(\hat{\beta})$ is big

This approach gives a different way to view near multicollinearity: it causes big standard errors only in those directions where $X'X$ is 'not stretching much', which turns out to be the places where the R^2 of X_2 on X_1 is very big. I would stick with the previous view of multicollinearity if I were you, but in case you're interested this eigenvalue-like approach is the way we think about it when we go about inventing estimators that work well even in situations where we have near multicollinearity (such as the Lasso, which is very famous in Machine Learning)

[2]Since the eigenvalue of A and 1 over the eigenvalues of A^{-1}. This follows as if $A = PDP^{-1}$ then note that $PDP^{-1}PD^{-1}P = I$ and so $PD^{-1}P^{-1}$ must be A^{-1}. Since D is diagonal, D^{-1} is the matrix containing 1 over the eigenvalues (which make up D)