Hypothesis Testing

Tom Glinnan

March 2022

Testing: a reminder of the big idea

All of the testing we've done in this course has the same underlying philosophy. There are a couple of different ways to think about it, but one way that can help with the intuition is to draw analogies with a court case. In theory, court cases are run on the principle of *innocent until proven guilty*: we start by presuming that the defendant is innocent, and ask ourselves how likely we would be to see the evidence we see under this assumption. One subtly different thing that we're *not* doing is assessing whether or not they are more likely to be innocent or guilty[1]. Instead, if we see some fingerprints on a murder weapon, we reason that we would be very unlikely to see this if someone were innocent, and so they're probably guilty. What we're doing here is a generalisation of a proof by contradiction. In a proof by contradiction, we assume something, and if that implies something false then our original assumption wasn't true. In a court case, if we see some evidence that's unlikely to be seen given the assumption of innocence, then we conclude that the defendant probably isn't innocent

Hypothesis testing is an exact analogy with the court case. We make some assumption (H_0) and say that under that assumption a test statistic T has a certain distribution. We can then calculate the value of the statistic we see in our data, and if it's 'too high' or 'too low', as measured by the distribution it's meant to have under H_0, then we conclude H_0 was probably not true; in other words, reject H_0. p-values, critical values and confidence intervals are different ways to formalise what counts as 'too high' or 'too low', which is known in law as proving something *beyond reasonable doubt*

Type I and II errors are about questions of how many innocent people we send to jail, and how many guilty people go free. We can immediately see that by raising the threshold for the evidence needed to convict[2] will send fewer people to jail, which has a good part (fewer innocents in jail, so a lower Type I error) and a bad part (fewer guilty people in jail, so a larger Type II error). So there is a trade-off here[3], and it's down to you which level of significance you test at. We can also see that

[1] The analogy to this in statistics is so-called Bayesian Hypothesis testing

[2] Raising the critical value, testing at a higher level of significance...

[3] You can view this like an indifference curve. More data or more efficient test statistics pull the curve inwards towards the origin

the way to improve this trade off is to get more evidence - in other words more data - or to use the evidence in a 'better way' - choosing a more efficient test statistic[4]

Wald, Lagrange Multiplier, Likelihood Ratio

The next question is how we actually make test statistics. Intuitively, we want them to have low Type II error - we want them to say you're innocent if you're innocent, and guilty if you're guilty. Suppose we're testing some hypothesis $H_0 : \beta = \beta_0$. In order to get a low Type II error, we want our test statistic to reflect some kind of distance between $\hat{\beta}$ (our best guess for β) and our hypothesis β_0. This makes sense: if β and β_0 are far away then we would expect $\hat{\beta}$ and β_0 to also be far away; and if β and β_0 are close we would expect $\hat{\beta}$ and β_0 to be close. Making a test statistic that has something to do with this distance is therefore a good way to get powerful statistics (low Type II error)[5]

Wald, Lagrange Multiplier and Lagrange Ratio just refer to some different ways we can talk about distance. We can apply any of these principles to any estimator $\hat{\beta}$ which can be thought of as the minimiser of some function $Q(\cdot)$. For example:

$$\hat{\beta}_{ols} = \arg\min_{\beta} Q_{ols}(\beta) := \sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \sum_{i=1}^{n} (y_i - x_i'\beta)^2$$
$$\hat{\beta}_{mle} = \arg\min_{\beta} Q_{mle}(\beta) := -\log p(y|x, \beta)$$

As maximising a function is the same as minimising the negative of it. In fact, every estimator we've covered in this course can be motivated as the minimiser of some function Q - there are very few estimators in all of Econometrics that can't be written in this way[6]. We can view the three methods on a diagram:

- **Wald Tests** are ones in which we consider distance to be the distance in the β space, and so involve $\hat{\beta} - \beta_0$. An example of this is the T-statistic for testing the regression coefficient $H_0 : \beta = \beta_0$:

  $$T = \frac{\hat{\beta} - \beta_0}{se(\hat{\beta})} \sim T_{n-k}$$

  under H_0 and the assumption of $A1 - A4$, plus normality of the errors. Without normality of the errors, we lose the ability to talk about the exact distribution but we still have that T is *asymptotically* distributed as a $N(0, 1)$, so we do testing based on $N(0, 1)$. Clearly, the $\hat{\beta} - \beta_0$ part is what makes this a Wald statistic, and the rest is a normalisation so that we have a 'nice' distribution of T under H_0

[4] We don't talk about the efficiency of test statistics in EC221, but it's a very similar thing to the efficiency of estimators. Here it means that (out of a certain class of test statistics) it has the lowest Type II error for any given Type I

[5] Notice that in principle you could use anything at all as a test statistic and the logic of Hypothesis Testing would still work. Using a Test Statistic which reflect a distance is just about getting low Type II error

[6] The name for estimators which can be written this way are *extremum* estimators, in case you're interested

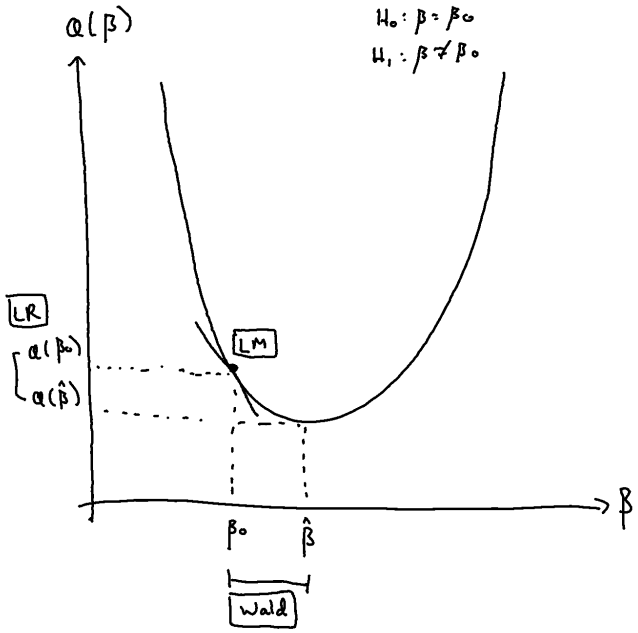

Figure 1: W, LM, LR on a Diagram

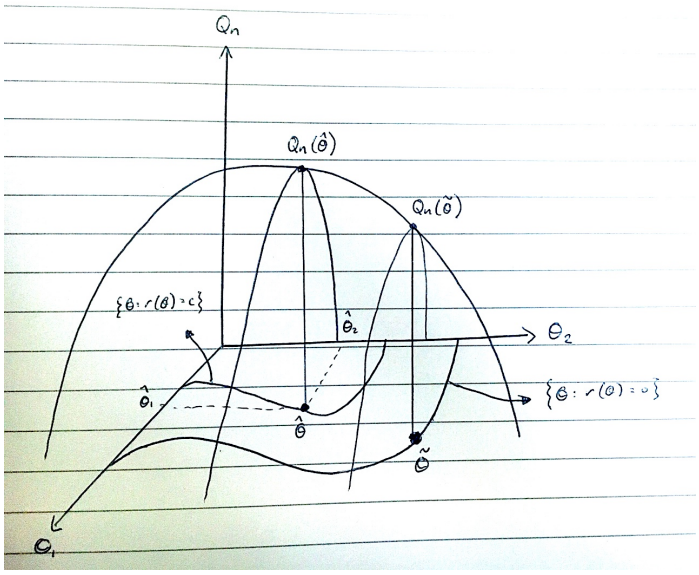

Figure 2: A more complicated diagram. $\hat{\theta}$ is the unrestricted and $\tilde{\theta}$ is the restricted estimator

- **LR tests** are about looking at distance in $Q(\cdot)$-space. In other words, our test statistic involves $Q(\beta_0) - Q(\hat{\beta})$, which is always positive as $\hat{\beta}$ minimises $Q(\cdot)$ by definition. A simple example of this is the F-test statistic for assessing if all of the $\beta_i = 0$ in OLS:

$$F = \frac{(RRSS - URSS)/J}{URSS/(n-k)} \sim F_{J,n-k}$$

under H_0 and the assumption of $A1 - A4$, plus normality of the errors. As OLS is about minimising the sum of squared residuals, $Q(\hat{\beta})$ is exactly the 'unrestricted sum of squares' URSS. Similarly, $Q(\beta_0)$ is the sum of squares when enforcing the hypothesis, so it is the RRSS. Therefore the $RRSS - URSS$ part of the test statistic is what makes this an LR test. Just like the above case, everything else is just a normalisation to get F a 'nice' distribution under H_0

- **LM tests**, also known as **Score Tests**, are the more difficult ones to explain. As we know that $\nabla Q(\hat{\beta}) = 0$ (by definition of a minimum), we can calculate $\nabla Q(\beta_0)$ and see how close it is to 0. Again, we can find a simple example by looking at OLS. The FOC for OLS are its moment conditions:

$$\nabla Q_{ols}(\beta) := \sum_{i=1}^{n} x_i \hat{\varepsilon}_i = \sum_{i=1}^{n} x_i (y_i - x_i'\beta)$$

So the test for whether or not $\nabla Q_{ols}(\beta_0) \approx 0$ is effectively testing if the moment conditions for OLS approximately hold at β_0. For all GMM estimators[7], LM tests can be interpreted as seeing how well the moment conditions hold. What we've given above is the *Score* motivation for the test - the score is the derivative of Q. It turns out that by some simple re-arranging you can also interpret the statistic as doing the constrained optimisation problem:

$$\min_{\beta} Q(\beta) + \lambda(\beta_0 - \beta)$$

and testing if the Lagrange multiplier $\lambda \approx 0$. The intuition behind this is that when you do a constrained optimisation problem, the Lagrange multiplier represents 'how binding' the constraint is (full proof here)[8]. Some people use the phrase *shadow prices* to describe this - use whatever intuition works for you.

[7] Which again, is pretty much all of them

[8] Note: **not** on the syllabus. If we define Q^* the *minimised* value of Q, then a result called the envelope theorem means we have $\lambda = \partial Q^*/\partial \beta_0$. In other words, if we were to make β_0 slightly larger, how much would that push up Q^*. A good intuition for this comes from a micro theory example which in EC202. In a consumer problem, we can define the *indirect utility* $V(p, m) = \max_x u(x) - \lambda(\sum p_i x_i - m)$ as the utility that the consumer actually gets from making their choice. Note that this is the max, not the argmax (which would be Marshallian, aka Uncompensated, demand!). Then we have that $\lambda = \partial V/\partial m$. If there was some choice that the consumer would love just outside of the budget constraint, then λ will be very large, as slightly expanding the income will give much more utility. This result is also why $\lambda = 0$ means the same as saying that the constraint doesn't bind: if we increased income m then that wouldn't change the utility. The takeaway is that large λ means that the constraint that $\beta = \beta_0$ is very binding, and so is the same as saying that the score ∇Q is not close to 0. Please feel free to talk with me more about this if you'd like

An example: building a Wald Test

At first, you might look at Test statistics and just see a mess of symbols. The key to learning test statistics is to try to understand what they're doing by building up to them in steps. As a running example, we'll talk about the χ^2 statistic for testing $H_0 : R\beta = c$ and $H_1 : R\beta \neq c$, where we estimate a vector β by OLS. To start, if we define $d = R\beta - c$ then of course it's equivalent to test $H_0 : d = 0$ vs $H_1 : d \neq 0$. Under our GM assumptions (plus normality of the errors), we get that $\hat{\beta} \sim N(\beta, U)$ for some matrix U, so $\hat{d} := R\hat{\beta} - c \sim N(R\beta - c, RUR')$. Therefore $\hat{d} \sim N(0, V)$ under H_0 if we define $V = RUR'$

When designing test statistics, we really need three things. Firstly, the distribution under H_0 needs to be some *standard* distribution, so that we can look up critical values in tables. Here, we know that $N(0, V)$ isn't standard, but $N(0, I)$ is, so it makes sense to normalise using $V^{-1/2}$:

$$V^{-1/2}\hat{d} \sim N(0, I)$$

Secondly, our test statistic must be something we can calculate. We don't know V because it contains unknown constants, but we can replace it by an estimator $\hat{V}$. This means we no longer know the distribution exactly, only approximately[9], so now we have:

$$z := \hat{V}^{-1/2}\hat{d} \ \sim^a \ N(0, I)$$

Thirdly, remember that in order to do testing we calculate the value of the test statistic we see in our data, then see if that's greater than some critical value. However $\hat{V}^{-1/2}\hat{d}$ is a vector, and we can't easily compare if one vector is 'greater than' another. Therefore, we usually require that our test statistics are *scalars*. The way to do this is quite simple: instead of z, we consider $z'z = \sum z_i^2$ because this gives us a nice distribution. Remember that a χ^2 is just the sum of many (independent) squared-normals. Since each component z_i is normal, and all of them are independent[10], $\sum_{i=1}^{k} z_i^2$ is distributed as χ^2_k, where k is the dimension of z. Therefore we can finally say that:

$$z'z = \hat{d}'\hat{V}^{-1}\hat{d} = (R\hat{\beta} - c)'\hat{V}^{-1}(R\hat{\beta} - c) \ \sim^a \ \chi^2_k$$

which is the Wald test statistic for testing $H_0 : R\hat{\beta} - c$ vs $H_1 : R\hat{\beta} \neq c$

How to Learn Test Statistics

What I hoped to show you with the above example is how we build test statistics, so you can see where they come from. To recap: the main steps are to start with some estimator whose distribution we know (or know approximately), then we usually normalise by dividing by something (to get a 'nice' distribution) and sometimes have to make adjustments to make the test statistic a scalar rather than a vector. If you see a test statistic with a quadratic form $x'Ax$, then it's a dead giveaway of the last step

[9] i.e. this is what the distribution converges to as $n \to \infty$

[10] As they are uncorrelated. Remember this is a special property for normal distributions

My advice for learning test statistics is to stare at them and try to understand how we came up with them. It's easier than it might sound. Usually it's either a W, LM or LR type of distance that's been suitably normalised. Try to do this earlier in revision - only consider blindly memorising test statistics if it's right before the exam and they haven't sunk in yet

Some Implementations

In reality, we can implement all of the EC221 test statistics in Stata. For computational reasons, Stata sometimes re-arranges test statistics to be functions of things it has already calculated. The easiest example is almost trivial: to do the test that $H_0 : \beta_1 = 0$ after regression, we calculate the test statistic as $\hat{\beta}_1 / se(\hat{\beta}_1)$. As Stata has already calculated $\hat{\beta}_1$ and $se(\hat{\beta}_1)$ to display elsewhere on a regression table, it's no extra computational work to calculate this. Many of the procedures in Lecture 8 and 9 are just the same idea, such as the LM test - we'll talk about it here, and it'd also shown in the visualiser notes for lecture 9 (VN9.10)

If $Q(\beta)$ is the function[11] which is minimised by $\hat{\beta}$, then remember that the *score* is the gradient $\nabla Q(\beta)$ - the vector of partial derivatives. To change the notation so that we're consistent with the notes, let's talk about θ and $\hat{\theta}$ instead of β and $\hat{\beta}$, and define $s(\theta)$ as the score vector $\nabla Q(\beta)$. Remember that the LM / Score test tests whether or not the score $s(\tilde{\theta}) \approx 0$, where $\tilde{\theta}$ is the minimiser of Q but imposing the hypothesis. If you remember, that gives the test statistic:

$$s(\tilde{\theta})' \Big[\hat{Var}(s(\tilde{\theta})) \Big]^{-1} s(\tilde{\theta}) = \sum_{i=1}^{n} s_i(\tilde{\theta})' \Big[\sum_{i=1}^{n} s_i(\tilde{\theta}) s_i(\tilde{\theta})' \Big]^{-1} \sum_{i=1}^{n} s_i(\tilde{\theta}) \sim \chi^2_r$$

where the first part is in matrix notation and the second is in vector notation, using the fact that $\hat{Var}(s(\tilde{\theta})) = \sum_{i=1}^{n} s_i(\tilde{\theta}) s_i(\tilde{\theta})'$. This isn't something we emphasise much in the course, but it turns out to be true, so it's a way to simplify things

It might help to use an example. Suppose that we're looking at OLS. As explained earlier, since we get OLS by minimising the sum of squared residuals, the score vector will be $s(\theta) = \nabla \sum_i (y_i - x_i'\theta)^2 = \sum_i x_i (y_i - x_i'\theta)$. $s(\theta)$ is a vector - notice that x_i is a vector and $(y_i - x_i'\theta)$ is a scalar, so $s(\theta)$ is a vector made up by summing a load of vectors that look like $x_i(y_i - x_i'\theta)$. If we call each of these s_i, then we can write $s(\theta) = \sum_i s_i(\theta)$, where each of the s_i are vectors

With all this machinery, we can now move onto looking at nR^2. The algebra can be a bit involved, but it's worth seeing once.

nR^2 is just an algebra trick

Like we were saying above, sometimes Stata uses tricks to calculate test statistics that's easier for it than simply calculating the test statistic above (which of course, it could do directly). One thing that Stata finds computationally very easy to do is OLS. As such, let's look at the regression:

$$1_i = s_i(\tilde{\theta})' \gamma + \varepsilon_i$$

[11] For example, the sum of squared residuals if $\hat{\beta}$ is OLS; $-p(\text{data}|\beta)$ if $\hat{\beta}$ is MLE...

where 1_i is the number 1 no matter what i is, and s_i is as above. We can write the regression in matrix notation too[12]. As $s_i(\tilde{\theta})$ is a vector, we can stack them into a matrix S and write:

$$\iota = S\gamma + \varepsilon$$

where ι is a vector with a 1 in every component. One thing that's calculated as a bi-product when we estimate this regression is known as the *uncentered*[13] R^2, which is in general defined as:

$$R_u^2 := \frac{\sum_i \hat{y}_i^2}{\sum_i y_i^2} = \frac{\hat{y}'\hat{y}}{y'y}$$

Remember that $\hat{y}$ is the predicted value of y[14], so in our notation this becomes $R_u^2 = (S\hat{\gamma})'(S\hat{\gamma})/(\iota'\iota)$. Now we can simplify. $\iota'\iota = n$ (as we're taking the inner product of two vectors which have 1 in all components), and we can re-arrange the top using our usual rules of matrix transposes. This gives us that:

$$R_u^2 = \frac{1}{n}\hat{\gamma}'S'S\hat{\gamma}$$

Now we replace[15] $\hat{\gamma} = (S'S)^{-1}S'\iota$ to get

$$\begin{aligned} R_u^2 &= \frac{1}{n}[(S'S)^{-1}S'\iota]'(S'S)[(S'S)^{-1}S'\iota] \\ &= \frac{1}{n}[\iota'S(S'S)^{-1}](S'S)[(S'S)^{-1}S'\iota] \\ &= \frac{1}{n}\iota'S(S'S)^{-1}S'\iota \end{aligned}$$

Now we do one final thing: go back to vector notation. As ι we just a vector of ones, $S'\iota = \sum_i s_i(\tilde{\theta})$. It's also not too difficult to show that $S'S = \sum_i s_i(\tilde{\theta})s_i(\tilde{\theta})'$ if you're careful with the dimensions. Therefore we can write that

$$nR_u^2 = \sum_i s(\tilde{\theta})'[\sum_i s(\tilde{\theta})s(\tilde{\theta})']^{-1}\sum_i s(\tilde{\theta})$$

which is the LM test statistic. We have to use uncentered R^2 instead of the usual (centred) one

Some Intuition

What we've seen above is that the procedure suggested in the lecture notes is an 'easier' way to calculate the LM test statistic (as we easily get R_u^2 when we run the above regression) - it's nothing to do with getting better estimation or anything like that. I would normally leave it here: many times in Econometrics we can find these faster ways to calculate something and we don't have to worry about intuition. However we can interpret it if we really want to

We want to test if $s(\tilde{\theta}) \approx 0$. Remember that $s(\tilde{\theta}) = \sum_i s_i(\tilde{\theta}) = S'\iota$ in matrix notation. R_u^2 is a measure of 'how well'[16] we can use $\hat{\iota} = P_S\iota$ to predict ι, where P_S is the predictor matrix

[12] Compare to how we write $y_i = x_i'\beta + \varepsilon_i$ and $y = X\beta + \varepsilon$

[13] The name comes because in the usual R^2, we take away the mean of $\hat{y}$ and of y inside the sums. This is sometimes called 'centering' them

[14] In our usual notation of $y = X\beta + \varepsilon$, $\hat{y}$ is defined as $X\hat{\beta}$

[15] Compare to $\hat{\beta} = (X'X)^{-1}X'y$

[16] In our usual notation: how well we can predict y using $\hat{y} = X\hat{\beta} = P_X y$

$S(S'S)^{-1}S'$. If $\tilde{\theta} = \hat{\theta}$ - meaning that our hypothesis is completely correct - by the definition of $\hat{\theta}$ minimising Q we get that $s(\hat{\theta}) = S'\iota = 0$. Therefore $\hat{\iota} = S(S'S)^{-1}S'\iota = 0$ as $S'\iota = 0$. So if our hypothesis is exactly true then $\hat{\iota}$ will be a very poor predictor of ι, and so R_u^2 will be 0. The further we get from H_0 the better $\hat{\iota}$ is, so R_u^2 is higher. Because of this it sort of makes sense why R_u^2 might be part of a good test statistic. It also makes sense that if n is larger then we should be able to more easily reject H_0 if H_0 isn't true, so nR_u^2 isn't a crazy test statistic

We regress on a 1 on the left hand side just to normalise things: if we regressed on another constant c that would also be fine, but then we'd have to use something like $nc^2R_u^2$ as the test statistic: this follows because in this case $R_u^2 = \sum \hat{y}_i^2 / \sum y_i^2 = \sum \hat{y}_i^2 / nc^2$, so $c^2R_u^2$ in this case would be what we called R_u^2 before. Keeping $c = 1$ keeps it simple!