Reintroduction to Linear Regression

Brenton Kenkel — PSCI 8357
January 21, 2016

This week, we are going to set up the linear regression model in matrix notation and derive the ordinary least squares estimator. It is going to be the most abstract and theoretical week of the course. My goal is for you to understand linear regression in the broader statistical context we established last week.

- The linear model is a particular way to parameterize the relationship between an outcome of interest and other observed variables.
- Ordinary least squares is one estimator (among many) for the parameters of the linear model.
- We default to using OLS to estimate the linear model parameters because its sampling distribution has nice properties.

Regression, Generally Speaking

Let's begin with the basics. The *response variable* is the variable we are most interested in explaining or predicting. We have N observations of the response variable, each denoted Y_i, collected in the N-vector $Y = (Y_1, \ldots, Y_N)$. Throughout this class, and the course, I will treat all vectors as column vectors, so Y is $N \times 1$.

We are interested in the response variable as it relates to other observable variables, which we call the *covariates*. The covariates for each observation are collected in the p-vector $x_i = (x_{i1}, \ldots, x_{ip})$, where p is the number of covariates. I write x_i (lowercase) to refer to the p-vector of all covariates for the i'th observation, and X_j (uppercase) to refer to the N-vector of all observations of the j'th covariate. Finally, I write $\mathbf{X}$ to denote the $N \times p$ matrix of all observations of all covariates—the matrix whose rows are the x_is and whose columns are the X_js.

We will start with a fairly general model of the relationship between the covariates and the response. This is the additive error model, in which

$$Y_i = f(x_i) + \epsilon_i$$

for all $i = 1, \ldots, N$. Remember from last time that we are primarily interested in conditional expectations: given that the covariates are fixed at some value x, what do we expect the response to be? Let us impose a basic "white noise" condition on the error term: that its expected value be zero regardless of the value of the covariates, so $E[\epsilon_i \,|\, x_i] = 0$. We then have

$$\begin{aligned} E[Y_i \,|\, x_i = x] &= E[f(x_i) + \epsilon_i \,|\, x_i = x] \\ &= E[f(x_i) \,|\, x_i = x] + E[\epsilon_i \,|\, x_i = x] \\ &= f(x). \end{aligned}$$

Just like last time, we call $f(\cdot)$ the *regression function*. In real-world data analysis, we don't know the exact form of the regression function—it's what we're trying to learn about from our data. To make this task tractable, we need to impose some structure on $f(\cdot)$. Most often, we'll assume that there is a finite (and, typically, small) set of *parameters* that define the regression function for all possible values of the covariates. Formally, this entails assuming that there is a *known* function $g(\cdot)$ and a vector of *unknown* parameters θ such that

$$f(x) = g(x, \theta)$$

for all feasible values of x.

The Linear Model

To review, here's what we've done so far:

1. We've defined the problem of learning the relationship between covariates x_i and response Y_i as that of learning the *regression function*, $f(\cdot)$.
2. We've made it tractable to learn the regression function from data by assuming its shape is a function of a small set of *parameters*, θ.

The linear model, which you first encountered last semester, is one way to accomplish step 2.

The *linear model* assumes that the relationship between the covariates and the response takes the form

$$Y_i = \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \epsilon_i,$$

where the p-vector $\beta = (\beta_1, \dots, \beta_p)$ is the set of unknown parameters. We call usually call the parameters of the linear model *coefficients*. We can write the linear model more compactly as

$$Y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i$$

and even more compactly, using matrix notation,[1] as

$$Y_i = x_i^\top \beta + \epsilon_i.$$

In fact, we can write the model for every observation in matrix form as

$$Y = \sum_{j=1}^{p} \beta_j X_j + \epsilon = \mathbf{X}\beta + \epsilon,$$

where ϵ is the N-vector containing each ϵ_i. This will be convenient when we come to estimation of the linear model.

One apparent drawback of the linear model is that the conditional expectation of Y_i is zero if all the covariates equal zero. Sometimes this is sensible, like if the response is weight and the covariate is height. Other times it isn't, like if the response is turning out in the last election and the covariate is income from employment. To bypass this seeming problem, we usually define the first covariate as $X_1 = 1$, and we call the associated parameter, β_1, the *intercept*. In applied work, you should never estimate a regression model without an intercept.

Since β_1 is the intercept, that must mean the other β_js are slopes. True enough. If we were to speak in causal language—and I'd rather not for now, but alas—we would say that one implication of the linear model is that every covariate has a constant marginal effect on the response. The partial derivative of the regression function with respect to the j'th covariate is a constant,

$$\frac{\partial E[Y_i \mid x_i]}{\partial x_{ij}} = \beta_j.$$

As we'll talk about more in a few weeks, and as you saw last semester, you can relax this a bit by including higher-order terms of the covariates (quadratics, interactions, and the like) in the model.

[1] The symbol $^\top$ denotes the transpose. Since x_i is a $p \times 1$ column vector, its transpose $x_i^\top$ is a $1 \times p$ row vector.

The Ordinary Least Squares Estimator

Once we assume a linear model, our goal is to estimate β from our data. How we go about estimating β is up to us. As we saw last week, we can use any statistic—any function of the sample data—as an estimator. What we really want, though, is a good estimator.

To derive the ordinary least squares estimator of β, we're going to work backward from a metric for the quality of an estimate. Imagine we have some b, a $p \times 1$ vector we are using to estimate β. We can't directly compare b to β, because we don't know β. But we do observe the responses Y_i, and we know that the conditional expectation of each is $x_i^\top \beta$. So if b is close to the true value of β, we would expect the predictions $\hat{Y}_i = x_i^\top b$ to be relatively close to the observed values of Y_i. The *residuals* are the differences between the predicted and observed values,

$$e_i = Y_i - \hat{Y}_i,$$

collected in the vector $e = (e_1, \ldots, e_N)$.

We will look for the estimate b that minimizes the sum of squared residuals,

$$e^\top e = \sum_{i=1}^{N} e_i^2.$$

This means we will penalize being off by 2 more than twice as much as we penalize being off by 1. We'll take an estimate that's always a little bit off over one that's exactly right half the time and wildly wrong the other half of the time. This is a choice with consequences—for one thing, our estimates will be sensitive to outliers. But, as we will see, it turns out not to be a bad default choice.

Our task is to minimize the function

$$\begin{aligned} e^\top e &= (Y - \hat{Y})^\top (Y - \hat{Y}) \\ &= (Y - \mathbf{X}b)^\top (Y - \mathbf{X}b) \\ &= Y^\top Y - 2b^\top \mathbf{X}^\top Y + b^\top \mathbf{X}^\top \mathbf{X} b \end{aligned}$$

with respect to b. You'll remember from the math boot camp that if b minimizes $e^\top e$, then the partial derivative of the above expression with respect to each element of b must be zero. So we have

$$\frac{\partial e^\top e}{\partial b} = -2\mathbf{X}^\top Y + 2\mathbf{X}^\top \mathbf{X} b = \mathbf{0}.$$

Rearranging terms gives us

$$\mathbf{X}^\top \mathbf{X} b = \mathbf{X}^\top Y,$$

or,

$$b = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top Y.$$

So, the *ordinary least squares estimator* is the statistic defined by the function

$$\hat{\beta}_{\text{OLS}}(Y, \mathbf{X}) = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top Y.$$

For some data, the OLS estimate might not exist. The sticking point is that the cross-product matrix $\mathbf{X}^\top \mathbf{X}$ must be invertible, but sometimes it isn't. Two main reasons why this might be the case:

1. There are more covariates than observations, or $p > N$.
2. One of the covariates (possibly the intercept) is equal to a linear combination of some set of the other covariates.

The usual risk factor for this latter case is when you've included every category of a dummy variable. For example, imagine you run an experiment with 10 subjects in which every other subject is in the treatment group.

```r
treatment <- rep(c(1, 0), length.out = 10)
control <- 1 - treatment
X <- cbind(intercept = 1, treatment, control)
X
```

```
##       intercept treatment control
##  [1,]         1         1       0
##  [2,]         1         0       1
##  [3,]         1         1       0
##  [4,]         1         0       1
##  [5,]         1         1       0
##  [6,]         1         0       1
##  [7,]         1         1       0
##  [8,]         1         0       1
##  [9,]         1         1       0
## [10,]         1         0       1
```

```
XtX <- t(X) %*% X
XtX
```

```
##           intercept treatment control
## intercept        10         5       5
## treatment         5         5       0
## control           5         0       5
```

```
solve(XtX)  # "solve" does matrix inversions
```

```
## Error in solve.default(XtX): Lapack routine dgesv: system is exactly singular: U[3,3] =
```

```
qr(XtX)$rank
```

```
## [1] 2
```

Why OLS?

Let's once more take a step back and review where we've been.

1. We set up the general problem of learning the regression from our data.
2. We assumed a linear model to make the regression problem tractable.
3. We derived the OLS estimator for the parameters of the linear regression model.

But there are infinitely many estimators for β. Why do we like this one so much?

The first answer is that the OLS estimator is *unbiased*: its expected value is the population value of β. If we could take thousands of samples and run OLS on each of them, the average of the results would be very close to β. You can confirm this by running a simulation in R. But it's also easy to derive analytically. First, if we treat $\mathbf{X}$ as fixed, we have

$$\begin{aligned}
E[\hat{\beta}_{\text{OLS}}(Y, \mathbf{X}) \mid \mathbf{X}] &= E[(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top Y \mid \mathbf{X}] \\
&= (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top E[Y \mid \mathbf{X}] \\
&= (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{X} \beta \\
&= \beta
\end{aligned}$$

Then, averaging over the population distribution of $\mathbf{X}$ gives us the unconditional expectation

$$E[\hat{\beta}_{\text{OLS}}(Y,\mathbf{X})] = E\left[E[\hat{\beta}_{\text{OLS}}(Y,\mathbf{X}) \,|\, \mathbf{X}]\right] = E[\beta] = \beta.$$

Unbiasedness is a finite-sample property. No matter the sample size (as long as $N \geq p$), the expected value of the OLS estimator is the true parameter, β. We also care about asymptotic properties, which characterize the behavior of the estimator as the sample size grows without bound. The most important asymptotic property is *consistency*. An estimator is consistent if, as the sample size grows large ($N \to \infty$), the bias and variance of the estimator go to zero. With enough data, a consistent estimator almost always yields an estimate very close to the true value. We won't go through the math here—too much i-dotting and t-crossing—but suffice it to say, under a broad and plausible set of conditions, the OLS estimator is consistent. This is another reason we like OLS.

While we're in the land of asymptotics, it is also worth mentioning that OLS is *asymptotically normal*: as the sample size grows large, the sampling distribution of the OLS estimator is approximately normal. Like consistency, this depends on some plausible regularity conditions we will not discuss. Asymptotic normality is what lets us use Z scores for testing hypotheses about regression parameters from regression results.

The last property we typically care about is *efficiency*: is the standard error of the estimator lower than that of the alternatives? Remember that we like efficient estimators because they let us make inferences more precisely. Is OLS efficient? It depends—relative to what? According to the *Gauss-Markov Theorem*, the OLS estimator is efficient in the class of linear, unbiased estimators, assuming some additional regularity conditions hold.[2] So, like Tobias Funke, OLS is BLUE (Best Linear Unbiased Estimator). The restriction to *linear* estimators is a crucial point here. OLS is efficient relative to any unbiased estimator of the form

$$\hat{\beta}(Y,\mathbf{X}) = C_{\mathbf{X}} Y,$$

where $C_{\mathbf{X}}$ is a $p \times N$ matrix whose value may depend on the matrix of covariates (but not on Y). But there might be a biased estimator whose variance is so

[2] In particular, these conditions are that, across observations, the error terms be uncorrelated ($E[\epsilon_i \epsilon_j] = 0$) and have identical variance ($V[\epsilon_i] = V[\epsilon_j] = \sigma^2$). We will return to these conditions in a few weeks when we discuss estimators for non-constant variance.

much lower that it's worth using, or an unbiased nonlinear estimator that is harder to compute (as nonlinear estimators tend to be) but even more efficient.

Summing Up

- We want to model a response as a function of covariates.
- We use the linear model because it is simple and its parameters are easy to interpret.
- We estimate the linear model by OLS because the OLS estimator is unbiased, consistent, asymptotically normal, and efficient relative to other linear unbiased estimators.

Next week, we will talk about the disconnect between the statistical theory we just laid out and the estimates that get reported in scientific publications. This will also lead us into a discussion of how we make inferences from regression estimates, a topic we've left mostly untouched until now.