Module 5: Linear Regression: Part One

Video Transcript

Video 5.1: Module Introduction: Building and Interpreting Managerial Predictive Models (6:22)

Welcome everyone to this module on linear regression. In earlier modules in this class, you started playing with data. You've visualized the data, you've learned a little bit about clustering, we've even been introduced to notions of probability and statistics so you can you can start formalizing your thinking about data. What we're going to be doing starting today is building models with data. So, to begin with, what is a model? Now, the way I think about a model, I think about a model in a relatively abstract fashion which I find useful. So, this is how I think about a model. I simply think about a model as a box that takes in inputs, and it gives out outputs. I mean, how much more general does it get?

So, let's try and make, let's try and refine it a tiny bit more. So, maybe here's a slight refinement from a box that takes in inputs and spits out outputs. It's a box that takes in independent variables. So, on the left over here, you're seeing independent variables X_1, X_2, through X_k. We're calling those our independent variables; you'll see why in a second. But our model is this box that takes in these independent variables and outcomes a dependent variable Y. This feels a little bit abstract. And to appreciate the abstraction over here, I think it's useful to make it a little bit more concrete and then come back to the abstraction. So, let's try and make this model a little bit more concrete. Here's a concrete example. Let's say you're building a fast-food restaurant. Stands to reason that one of the things you might want to understand is, what are your sales going to look like at this fast-food restaurant? And so, what we're going to do is we're going to say, "Hey, listen. There's a whole bunch of things we control. We control pricing, we control the menu, we control advertising, there might be the demo, the demographic of the customers that might come to this restaurant based on where it is, the competitors in the area, and a whole bunch of these things." All of these together form the independent variables for our model up here. What does this predictive model do? What is it predicting? It's trying to predict sales.

So, what comes out of this model is presumably sales. You're looking at this and one of the places your head's probably going is you're naturally thinking of these independent variables as numerical quantities like pricing, advertising. These lend themselves very naturally to numerical descriptions. But a model doesn't have to take in this very structured view of what the independent variable is. Here's a different sort of model. On the left, my independent variable is a picture. So, the picture might be a picture of a dog or a cat and the job of my model is to tell me whether the picture we're looking at is a picture of a dog or a cat. You're probably looking at this and you're saying to yourself, "Wait a second. The earlier one, I can see how that's useful, predicting sales. I can recognize a dog and a cat. Why do I need that?" But if you think about this a little bit more, you might say, "Well, wait a second. What if this picture were going into the brain of a self-driving car?" And as opposed to recognizing a dog or a cat, we wanted to recognize a stop sign in what the car is actually seeing or a person walking into the frame or something like that. You can see how something like this might be useful. But again, like what we're looking at

the, it's still a model. We have an independent variable or variables, in this case, the independent variable was a picture. The output was what sort of picture are we actually looking at.

So, coming back to that abstraction we were talking about, which we can hopefully appreciate a little bit better now. The model takes in a whole bunch of these independent variables, spits out a dependent variable. And the big question is, well, what's the model? What goes into the box in the middle that sucks in the independent variable, spits out the dependent variable? And there's a whole bunch of things that could actually go in there. For instance, for the example with the dogs and the cats or equivalently detecting a stop sign in an image, we'll need to actually worry about models that look like deep neural nets. What we're going to do today is, we're going to start with the basics. We're going to start with linear regression, and we'll talk about logistic regression. I like to think about these models as Swiss army knives. These are the first models you want to actually start with. And I could go on forever talking about, why I'm calling this a Swiss army knife. But I think my sentiment over here is best captured by this quote.

Now, I don't know who actually said this so I don't know who to attribute this to but here's the quote and let me just read it to you. "When you raise venture capital, talk about AI. When you hire your team, talk about machine learning. But when you actually build a predictive model, start with linear aggression." Now this is aimed at potentially, a technologist starting a technology company or something like that. But I think the sentiment here captures the sentiment that I have thinking about why linear regression and logistic regression are positioned where they are. These are truly the first things you want to start with. They're robust predictive models that we can get going with relatively minimal effort as you'll see. And they'll already get us to a really useful kind of 80, 20 points in thinking about predictive modeling. They're also interpretable models. So, you'll try, and you'll get to understand what's actually going on in your data. But enough with this. What I want to do now is get going with an actual modeling task.

Video 5.2: Motivating Application: Predicting Bluebikes' Rentals Demand (7:29)

All right, so let's actually get going with a real predictive modeling exercise. Enough of this abstraction. And so, here's the actual modeling exercise I want to talk about, and it pertains to Bluebikes. So, Bluebikes is actually a bike share type program, and it's here in Boston, where I am. It's a program that's owned by the municipalities of Boston and neighboring suburbs, Brooklyn, Cambridge, and so forth. And what the program is, basically, you can walk up to a hub, so to speak, where there's a whole bunch of bikes available, rent a bike, ride it to where you want to go, and then park it at a different hub. And the idea, of course, is to get people riding bikes, which is better than driving your car around, presumably. What you see is a picture over here of the Boston area. I'm in Cambridge right now. What you see in the picture, though, the little dots are locations of these sort of bike hubs. Now, what I promised was an actual predictive modeling task, and here's the predictive modeling task. So, let's say that our goal is to understand or to predict the number of total Bluebike rentals tomorrow. Let's make up a date; let's say tomorrow is Thursday, September the 30th, just for specificity.

And we care about the number of Bluebike rentals tomorrow, Thursday, September the 30th, between 5:00 and 6:00 p.m. Now, the first thing I think about when I'm faced with questions like this is the following: Why would you care about this? Often, what I find, and you'll see this as a recurring theme, is understanding this question of "Why do you care?" actually guides us in a useful way in terms of how we might want to build a model, how we might want to actually understand whether the model is doing its job, and so forth. But in this particular case, the answer is a relatively easy one. If we expect things to be particularly busy, we might actually be able to change staffing policies, for instance. In fact, this is not all that different from the concrete example I gave you earlier about trying to predict fast food sales. Over here, I'm trying to predict blue bike rentals at a specific time with perhaps the sort of ostensible goal of staffing my call center, customer service center, or something like that differently if I expect demand to be exceptionally high. Now, you're probably thinking to yourself, "Well, what you talked about earlier was predictive modeling that had independent variables and dependent variables, and so forth.

The dependent variable sort of makes sense. What's the dependent variable over here?" If you guessed while the dependent variable is rentals tomorrow, you're exactly right. That's what we want to predict. So, of course, that's our dependent variable. But what's the independent variable? As a modeler, that's not always clear, actually. As a modeler, sometimes you have to actually step back and ask yourself, "What is the independent variable of your?" Put in a really simple, colloquial way, "What's going to impact rentals tomorrow?" Now, if you live here in the Boston area or more generally in the Northeast, you're probably saying to yourself, "Well, I know one thing that's going to actually make a difference, it's the weather." In fact, Boston, the Northeast in general, we have a tremendous dynamic range in the outside temperature. You might see a picture like the one over here where we look at Boston daily temperatures. You can see they go from 0 to 100. They run that entire dynamic range, and as you might predict, people don't like riding a bike when it's 10 F outdoors. That's what we're seeing in the picture. And so, indeed, you might say to yourself, "Okay, great. Actually, I can refine the question. I can refine that predictive modeling question we were asked earlier.

And as opposed to just simply saying, 'Hey, what are Bluebike rentals going to be tomorrow September 30th between 5:00 and 6:00?' I'm going to ask that question armed with, let's say, a prediction of the weather." And so, let's say you had a prediction of the weather, and the weather tomorrow at between 5 and 6 was supposed to be 70 F, 40% humidity, 15mph wind speed. By the way, that's a beautiful day. That gives you some information. And so when we get into this modeling task, which we will in a second, what we're thinking to ourselves here is, you know what, the dependent variable, that's clear, that's Bluebike rentals over a specific hour. So obviously, one independent variable then becomes the day, Thursday, the time of the day, between 5 and 6, that was par for the course with the original question, but in addition, we're arming ourselves with independent variables that pertain to the weather. Again, I should say, if we were actually building one of these Bluebike rental models, we might want to go even further. We might want to spend way more time thinking about what the right independent variables ought to be. To get along with things, I'm going to just stop over here, but we'll see many, many more examples in the course of this course, where we look at a whole bunch of different modeling tasks and dive into what sort of independent variables ought to be.

So, that's sort of the question we're going to be going after. Now, because I'm greedy, I might actually want us to do even more. And this goes all the way back to what I was saying earlier when I said, "Well, what is the model going to be used for?" If, at the end of the day, let's say the model is going to be used to make some sort of managerial decision of how many people do I have should I staff at my call center or something like that, you

might care about how confident you are in your prediction. If, in particular, the prediction says that, you know what, tomorrow, between 5 and 6, you're going to have five times as many people renting bikes than you normally expect or than you were expecting, your reaction as a manager might be great, that's sort of a point estimate. But what's the chance that that thing is actually going to happen? Because my decisions might actually depend on the likelihood of something like that happening. And so, later on, after we get our arms around predictive modeling and linear regression for this sort of task, I want to go a little bit further and say can we answer questions like how likely is it going to be that we'll exceed, let's say, 500 rentals tomorrow? And obviously, the answer to that question is a probability, a likelihood. And so we'll see how to get to that as well.

Video 5.3: Predicting Bluebikes Rental Demand: The Data (6:59)

So, we've been talking about this Bluebike problem. Hopefully, you're excited to start, getting your hands dirty with the problem. Now, what you're going to have access to is a Colab where you can actually play along with the data. So, as I talk through this, you might want to actually pull up that Colab and do the same things I'm doing. If you don't want to, that's fine too. But really, as a data scientist, as a modeler, our job is about working with data, so we'd better get going with that as quickly as possible. Now the exciting thing about the dataset that we have over here is that it's real Bluebike data. So, really, we have about 9000 odd records. And essentially what these records are, they're basically hour by hour records of Bluebike rentals in the Boston area. So, if you look at the top of the dataset, the first six rows of the data, that's the df.head, DF being the frame, the data frame we're looking at. And you'll see over here that, in addition to the rentals, we have information on the time. So, the month of the year, the day number, the hour number, the day of the week that we're actually looking at, whether or not it's a weekend and then all of that sort of weather related, independent variable stuff that we were talking about. Two different ways of measuring temperature: temp and temp wet-bulb. Humidity, wind speed, and then precipitation. So, we have all of this information. And that's great. Like you look at the bottom of the dataset, this is the last few records. You see sort of the same sort of story. Again, rentals, what month of the year, and obviously, where it stands to reason the top of the dataset, it's the first month January. The bottom of the dataset is December, so month 12 and the last day of the year.

So, this is sort of New Year's Eve. The hour, the day of the week, and so forth. Now every time I'm given a dataset, the first thing I like to do and the first thing, to be honest, you ought to do is, play around with the data. Right now, you already did that in the first few modules when you did visualization and clustering and that sort of stuff. Honestly, I feel, and I'm getting a little philosophical here. When you play around with the data, that's really what gives birth to these 'aha' moments, where you say 'aha'. I suspect that this thing matters, and this other thing doesn't matter or something like that. And so, one of the things I like to do is, I like skimming a dataset. And what do I mean by skimming the dataset literally running this command called skim. It's a command that data scientists that have used 'R' which is a different language used pretty often, pretty frequently.

And we're working with Python but it's available here too. But what's skimming a dataset? In this particular case, 'skim the dataset', we get the bird's eye view of what's going on in the data. So, as I look at the frame over here, when we skim this dataset immediately as you look at the top, you see that, "Hey, this is a dataset with as I promised earlier close to 9000 odd entries, 8,603 to be precise." And there's two sorts of variables over here. There's sort of numerical variables that go under the name 'Number'. So, you see that box that has number at the

top of it and then you have 'categorical variables' that go under the name category. We'll say more about these numerical variables and categorical variables later. But for now, suffice to say that numerical variables, they are numbers, they're naturally numerate. Whereas categorical variables, they're not quite numerate, they're naturally category.

So, a month, for instance. A month could be January, it could be July and that's 'categoric'. It could be the weekend, or it could not be the weekend. That's categoric. On the other end rentals, that's a number, temperature, that's a number. So, we have these two sorts of variables. When we look at this, you immediately see a whole bunch of things. My eye goes immediately to rentals over here. That's what I'm being asked to predict. The most striking thing to me when I look at rentals is the dynamic range. There are some hours in our dataset. Some days, some hours in those days where we, like in all of Boston we rented one bike. On the other hand, there are some hours if you look at P-100, that's kind of the max where we rented close to 1300 bikes. So, the dynamic range is pretty large here. There are some hours where we rent out one bike, some hours where we rent out 1300 bikes. And that, you should be saying to yourself, well, wait a second, with given that dynamic range, if I build a predictive model, that's potentially valuable.

The things I do if I were renting 20 bikes an hour, a very different from the things I would do if I were renting out, I don't know, a 1000 bikes an hour. At the same time, if you look at temperature, as promised, look at that dynamic range. There are hours where the recorded temperature is -2 F, that's pretty cold. And on the other hand, there are hours where the temperature is 97 F. Pretty darn hot. Same story with humidity. There are hours where it's 100% humidity. That doesn't feel great by the way. And on the other hand, there are hours where the humidity is like 16%.You might as well have like a dehumidifier or something like that going. So, there's tremendous dynamic range in this data and as such, as a data scientist, I'd be pretty excited about understanding whether all this independent variable information I have. What are the independent variables again? Everything other than rentals. The temperature, the relative humidity, the wind speed, the precipitation, the time of the year, the hour of the day. All of these things are potentially valuable independent variables. And our goal next is simply to say, "Hey, can we build a model that relates these independent variables to the dependent variable we care about, which is rentals?"

Video 5.4: A Baseline Model (4:01)

So, we've looked at all of this wonderful Bluebike data, and we're sort of really getting ready to go over here. We really want to build our predictive model that predicts rentals, but I want to slow you down a bit. It's very tempting to just get going because we're so excited to do this stuff. What I like to do before doing that is asking myself, "Well, what's my baseline model?" What is a baseline model? Well, as the name suggests, it's a very simple model that we'd like to compare ourselves to to understand whether what we're doing has value to it. So, that's hopefully not very controversial. But it's striking that, as data scientists, people often forget to build a baseline model. Because you're so excited to get going. What's the baseline model we can do over here? Now, coming back to our data, here's the data again. I'm looking at a few more rows of the data. Really, what we care about predicting is rentals. And what I'm asking right now is, how might you think of building a baseline model over here? I think this is a good place to just pause your video. And just think to yourself: what's the simplest model you can think of building? The simplest baseline model I can build is a prediction of rentals absent knowing any information.

That is, what can I say about rentals if I had no independent variables whatsoever? So, in particular, if I looked at this a little bit more graphically, as it were, you've obviously done a whole bunch of visualizations so far as you've played with data. I might like to want to do a histogram of the number of rentals. And so as I look at the data set and I histogram out across hours, what's the number of rentals in any given hour, as we sort of expected, the dynamic range is large. It goes from one rental to some days that have 1300 rentals, and the histogram you're seeing here is just a depiction of that. Well, you don't have to have the histogram, but this is how I was thinking about it. I thought let me just look at the data and see if looking at the data gave me a sense of how I might approach this task of building a baseline model. And what I thought was, after looking at this was, "Why don't I just predict the mean?" What's the mean number of rentals in any given hour? And in this particular case, in this particular dataset, it's 210. So, that's a baseline prediction. Okay so, as we sort of think about this baseline prediction, a natural next question to ask is: How good is that baseline prediction?

Another good place to, I would say, pause your video. Pause the video and think to yourself: How would I measure whether this baseline prediction of 210 was any good? Now, if you're thinking to yourself, given everything you learned about probability and statistics is why don't I simply look at the standard deviation of the number of rentals? You're dead right. That's what I would do. In particular, if I look at the standard deviation of the number of rentals, which in this case is 230 bikes, you could think about that standard deviation as being an average error. Okay so, just summarizing, as I think about my baseline model, the average number of rentals, my baseline prediction is 210 bikes an hour. And the average error over there, which I'm measuring as a standard deviation, is 230 bucks.

Video 5.5: Gathering Relevant Variables (8:49)

The first question you should be asking yourself at this point is that's great, but we've been talking all along about independent variables and using these independent variables. Shouldn't that actually help us build a much better model? Now, in general, it's not clear, because you might actually have independent variables that are not relevant to the problem at hand. And so before going crazy and building a model, I want to stay in this realm for a bit where we're still playing with data to get a sense of this question of, if I had information on an independent variable, is that information useful to me or not? And so, in particular, what I actually want to be thinking about over here, just as an exemplary question is, early on, we hypothesized that the weather made a big difference to bike rentals, let's see if it does. So, in particular, if you told me that the temperature was going to be 25° Fahrenheit, what would our prediction be then? Hopefully, what you're saying to yourself is why don't I do exactly what I did with all of the rentals, except this time what I'll do is look, I'll filter down that set of rentals and only look at those rentals when the temperature was around 25° Fahrenheit.

So, that's exactly what we've done over here. We've looked at only the rentals when the temperature is 25° Fahrenheit and what we get is a very different looking histogram. What do you notice about this histogram? Hopefully, one of the things you notice is unlike the earlier histogram where we saw a very big dynamic range in rentals from no rentals to essentially 1300 rentals, over here the rentals are clumped to the left. Does it make sense to you that the mean number of rentals over here is 73? Well, hopefully what you're saying to yourself is the following. Given information that it's an exceptionally cold day or a cold hour, it's 25° Fahrenheit, presumably that means I'm actually seeing fewer rentals and indeed what we see is that's exactly what happens in the data. When

we look at all hours of the day, when we look at a histogram across the entire dataset, we see that the average number of bike rentals is 210 bike rentals an hour.

On the other hand, if I told you that it was 25°, that average drops to 73, and hopefully that makes intuitive sense. This is the gut checks that we want to do before we do any sort of modeling. We're actually seeing the hypotheses we had kind of play out in the data. Now, digging one level deeper. There's something else you should notice over here. Look at the standard deviation which was our quote unquote measure of the quality of a prediction. On the left-hand side, absent any information on independent variables, the standard deviation was 230 bikes. But when I told you that the temperature was 25°, what happened to that measure of quality? It became 91. What are you seeing over there? What you're seeing over there is something we hope will become generic. When you give me more information that's pertinent to my prediction task, the uncertainty that I have in my prediction, the quality of my prediction effectively goes up. So, when you tell me that it's going to be 25 degrees, that's actually relevant. It actually gives me information that lets me tell you whether this is going to be hour where you're going to have a lot of rentals or hour where you're not going to have very much in terms of rentals. Why stop here? We don't have to stop here. Here's a different take. What if, instead of telling you the temperature, I told you the time. I told you that it was 5 in the morning. Now, hopefully you're saying to yourself again, this is like the equivalent of that 25° hour. It's really early in the morning who's out there riding a Bluebike. We can look at what the histograms tell us. And so, in particular, what you're seeing over here on the left is the same old histogram we've shown earlier. And that's the histogram of bike rentals over the entire dataset and I've just flipped it on its side. The reason I flipped it on its side is now the quote unquote Y axis over here is number of rentals. And so what you see with respect to number of rentals. You're seeing that big dynamic range from to 1300, all right. On the other hand, let's look at what that dynamic range becomes when you tell me that it's going to be 5 in the morning. We go from that massive dynamic range 0 to 1300 rentals to I don't know 100 rentals. That tiny little bar over there.

So, again, we're seeing that same dynamic. The more information you give me, the lower the variability, the lower the uncertainty in the prediction I'm going to give you. Can we look at this dataset in like a more holistic fashion? So, here's my attempt at looking at the dataset in a more holistic fashion. And again, we're not building models yet. We're still kind of playing with the data to get a feel for the data, to get a feel for what matters and what does not matter. And so, what I'm showing over here as opposed to histogram is a scatter plot. And it's a scatter plot where the X coordinate of a point is the temperature at that recorded hour, and the Y coordinate is the number of rentals during that hour. So, I create a scatter plot of all of these points. And what you have, what you get is the picture on your screen. Now, I want you to pause for a second, stare at this picture, and try and come up with a description for this picture. So, hopefully you've looked at the picture. What you're coming away with is you know what there's a bit of a trend here. The trend is that as it gets warmer, people rent more bikes and perhaps that actually makes sense, that as it gets warmer people rent more bikes. Can we make that trend even more visual, even stronger?

And so, here's what I'm going to do. I'm going look at every one of these recorded temperatures, 0, 20, 10 whatever it is. And for every one of those temperatures, I'm going to go figure out what the average number of rentals at that temperature is. And I'm going to plot that. I'm going to overlay that that average on this picture over here. So, that's actually exactly what I've done with the orange dots. The orange dots are the following for any different temperature, let's call it 60 Fahrenheit. I'm going to look at all the records I have where the temperature was 60 Fahrenheit, and I'm going to look at the average number of Bluebike rentals at that temperature. And that would give me the orange dot for 60 Fahrenheit. Look at this. This curve of averages tells us a beautiful story. It's telling us that, "Hey, look,

as it gets warmer, there's this definitive trend in the number of rentals that you see." This to some extent is the starting point for linear regression. You look at a picture like that. You ought to be saying to yourself, "Well, can I actually build a model given this picture?" And one of the models I could build is, well, why don't I just pick, fit a line to all of those orange dots? At the simplest of levels, and it's very, very simplest, linear regression is about fitting that line. It's much more than this. As you'll see, as we kind of peel the onion. But at its basic, at its simplest level, linear regression is about fitting the red line that you see in the picture here to all of the orange dots. We want to essentially extract a linear trend as it were in this particular case that relates temperature to the number of rentals. Now, that's great. That's very visual. And I think we're actually in a fantastic position to actually really exploit this dataset and go further.

Video 5.6: Simple Linear Regression: Part One (12:44)

So, we produced that beautiful scatter plot where we looked at rentals versus temperature. We saw this really beautiful trend emerge of rentals versus temperature. We're going to begin with simple linear regression, which is regression where you have one independent variable. That's where we're going to begin. What's it trying to do? What it's trying to do is it's trying to find a straight line that best fits the average values, the thick orange dots. That's really what it's trying to do. Now, what might it do? It might say, okay, let's try the blue dash line, and the blue dash line may not be such a great fit. And they say, well, you know what, let's go try the green dotted line. Not such a great fit. Let's go look at the red solid line. Just right. That's my mental model of what linear regression, simple linear regression is trying to do. Now, of course, it's not exactly doing that. What is it actually doing? Let's get into it. Here's one way, one formalism of simple linear regression. And we have all the tools to understand this formalism. So, y over here, our dependent variable, that's the number of rentals and that's a random quantity.

It's a random variable actually. y is effectively our dependent variable. It's a random variable. And you might imagine that this random variable was generated as follows. It was generated as the sum of three terms. The first term was an intercept, beta_0, the Greek letter Beta. So, the intercept is the first term. That's just a constant. That, nothing changes over there. The second term is a slope, beta_1 times our independent variable. Our independent variable in this case is temperature. So, beta_1 times x is the second term. And of course, what we're seeing over there is that as our independent variable x changes, it gets multiplied by beta_1, and as long as beta_1 is not zero, y is going to change. And then finally, because we don't expect y to be determined exactly by just that constant and temperature, there's going to be a bunch of unaccounted stuff. Stuff that we're not really able to explain our model, we're going to think of that as noise. And we're going to denote that noise by epsilon. So, that's the model we have behind the scenes, and our goal is the following. Our goal is to quote unquote estimate what that model is.

So, we don't really know what beta_0 is, but we're going to try and learn what beta_0 is. And we're going to call our estimate of beta_0, b_0. In the same way, we don't know what beta_1 is. And so, we're going to try and learn beta_1. We're going to call our estimate of that b_1. And by its very nature, noise is not something we're trying to chase. Noise is something that remains unmodified. So, that's really the game over here, to go from data to really estimating these parameters, b_0 and b_1. Now, at the start, we said, what was going on was we were trying to find that best fitting line. Here's a slightly more formal way of thinking about how that process of finding the best

fitting line might actually go. And so, imagine, if you will, that your data consists of the red dots. Each red dot corresponds to a temperature, and you observed rentals at that temperature. That's what we have as data. What are we trying to do with this picture? We're trying to build a line. What is a line over here? It has an intercept, the intercept is b_0, and it has a slope, that slope is b_1. So really, the line is fully determined if I tell you what b_0 and b_1 is. And so, my job is to give you a line that is equivalently come up with a b_0 and a b_1 that fit this data as best as possible. Now, what does fitting the data well actually mean? Well, here's one very natural way of thinking about what fitting this data actually means. Pick one of the points in the dataset. Let's say we pick the third point from the left, and we ask ourselves under a proposed model, under this blue line, how good is the blue line at predicting what happened at that data point? Well, what happened at that data point is what I'm calling the actual y_i, the red solid dot that says Actual. And what we're predicting is the green box. We're calling that y hat i. The hat's used over here to say that's what we're estimating. Well, it's not a perfect estimate. We don't expect to come up with perfect estimates. There's an error. What's the error? The error is literally the vertical distance between y_i and y hat i. We're going to call that the error. We're also sometimes going to call it the residual. There's a lot of names for this. So, if you hear me talk about residual in one breath and error in the other breath, I'm saying the same thing. These words are used interchangeably. So, that's the error. Given that that's the error, what might we do? Given that that's the error, what we might want to do is find the blue line that minimizes, in a sense, the total error right across our data, across all of our points, not just that point. But we got to be careful about this because the error on some points might be positive and the error on some points might be negative. And what if all the positive errors canceled out with the negative errors? We might fool ourselves into saying we have no error. Well, that's not the case. Because error is error. And so, what we do is, as opposed to just saying error, we actually look at the square of the error so that irrespective of whether it's positive or negative, we don't like error. It's going to cost us. And that's it. We're going to find a line.

We're going to find equivalently an intercept b_0 and a slope b_1 that minimizes the sum of the squared errors across all of our data. Now, we're never actually going to have to do this manually. I'm actually forcing us to go through this so that we develop a sound understanding of what's actually going on over here. In particular, when I talked earlier about simple linear regression as finding the line that's the best fit to the data, of course, simple linear regression was not trying every possible line. It wasn't actually looking at the dash line and the dotted line and the solid line and whatever and trying to find the best fit. No, it was simply trying to find a b_0 and a b_1 that minimized the sum of the squared errors. And hopefully, out of that popped the solid red line. That's really what's happening behind the scenes. Now, we have computers, we have code, we have big data. In fact, we're going to see the power of that as we go along. Of course, we're not going to be sitting there manually minimizing the sum of the squared errors. In fact, in Python, doing this is really simple.

If you go back to your Collab, if you're following along in your Collab, and you want to produce that solid red line, so to speak, what does it take in code? What it takes in code is literally one line. And I want to actually go through that line of code because at some level, it just shows you how transparent doing this is. If we think about the model we're building, the model we're building is we're trying to explain rentals, that's our dependent variable, with this independent variable of temperature. And so, all we do is we say, "Hey, listen. We're going to use ordinary least squares". That's really what this module is called in Python, and in general, by the way. O for ordinary least squares. And the formula we're going to try and fit to our data is explaining rentals with temperature. That's it. And that's pretty darn readable. It's one line of code and out of it comes all of this gobbledygook that you actually see

on the screen. There's a whole bunch of stuff going on over here, and I'll be the first one to say, you look at this for the first time, maybe it's a little overwhelming. Don't worry about it. We're going to actually master all of this stuff.

At least we're going to master what matters. And I want to start really simple. Earlier, we were saying we're fitting a line. What's a line? A line when you have one independent variable is an intercept on a slope - b_0 and b_1. And so, your first question should be, what is b_0 and what is b_1? So, b_0 over here is simply the intercept, -115. And the temperature, the temperature slope, the b_1 is 6.0176. There's a whole bunch of other stuff going on. Let's forget about it for now. Let's just focus on the main thing happening over here, the model that we learned and those coefficients. b_0, that's -115, b_1, that's 6.0176. What actually going on over here. What's actually happened is, essentially the code has minimized the sum of the squared errors, the sum of the squared residuals, and it said that that solid red line has a slope of 6.01 and an intercept of -115. We didn't have to manually try a whole bunch of different lines. It said that this solid red line was the best fitting line. Let's try and interpret this. Is this intercept of -115 plus about six times whatever the temperature is in Fahrenheit.

So given this, just start thinking, let's say, we wanted to go with this. The boss needs a model right now. Let's say we go look up what tomorrow's temperature is going to be, and we say, "Well, tomorrow, it turns out, it's going to be 75. Tomorrow evening, it's going to be 75 degrees F." If you knew that, if you want to look this up on your favorite weather prediction app, and it said the temperature tomorrow between 5:00 and 6:00 was going to be 75 degrees F, what's your prediction going to be? Let's just plug in 75 degrees F into the model. And so, plugging it in, what do we get? We get -115 + 6 * 75, that's 336 rentals. So, that's great. What we've done so far is we've said, "Look, we expect the temperature to actually impact us". In order to make this real, we went and looked at our data and we fit a simple line, simple linear regression that tried to predict rentals versus temperature, and out of this came this model that said, "You know what? The line that fits this data best has an intercept of -115 and a slope of six." And we use that simple line to predict what rentals were going to be if the temperature tomorrow was 75 F, and we got 336 rental.

Here's what I want to do next. We don't just have temperature in the data. We have a whole bunch of others. We have humidity, we have windspeed, we have whatever. We also know things like the time of the day, the day of the week, and all of this sort of stuff. So, what I'd like to do just to turn the crank over here so that we get familiar with this is very quickly run through this same simple linear regression modeling exercise for a whole bunch of other choices of the independent variable.

Video 5.7: Simple Linear Regression: Part Two (10:58)

So, we did simple linear regression where we looked at rentals as a function of temperature. What about relative humidity? Maybe humidity matters. And just like I did bike rentals as a function of temperature, I produced a different scatter plot. So, this is a scatter plot of rentals versus relative humidity. Okay. Same sort of story like we can try and fit a red line to it. If you eyeball the data, maybe you're sort of seeing there's a solid red line and the slope is a little bit opposite, right? And that seems to make sense. As it gets more humid, we don't expect to rent as many bikes. We don't have to guess at this stuff anymore. We could go back to our trusty Python code, and this time all we got to do is change like one little thing. As opposed to the formula being rentals versus temperature, the formula now is going to be rentals versus relative humidity. That's it from the model changes. We get a different straight line, we

get a different intercept, we get a different slope. The intercept is now 340. The slope is -2.0146. Let's pause here for a second. Does this model make sense to you?

So, if you've given this some thought, you're looking at it, you say, "Well, the coefficient in front of relative humidity is negative." So, what does that say? Relative humidity decreases, rentals are going to go up. That's really what the model is predicting. Hopefully, that drives without thinking that like, hey, not a super humid day. You going to be more willing to ride your bike. Or, conversely, it is a super humid day, we all want to ride our bike and get all sweaty. And so maybe that makes sense. But why stop here? Okay. Let's keep going. We did rentals versus humidity. Let's, now look at rentals versus whether or not it's the weekend. Okay. Now this one is a little bit different because if you think about weekend, it's not quite a numerical variable anymore. It's one of those categorical variables that we talked about earlier. So, in this particular case, if it's the weekend, we're calling it a one. If it's not the weekend, we're calling it a zero. And if we scatter plot it, you see what you see over here. There's a wider range of rentals when it's not the weekend. Relatively narrow range of rentals when it is, and it feels like, if you were to force me to draw a red line, I'd imagine that there'd be a slope that goes slightly negative, but I don't have to guess at this. I can simply build a model, and all that's required over here is plugging in a new formula as opposed to rentals versus temperature. All I have to say is rentals versus weekend. And what comes out of this is a straight line fit where the coefficient is 222 and the slope in front of the indicator of whether or not it's the weekend is -61. What is that telling me? That's telling me that on a weekend, we expect all else being the same. We expect to rent 61 fewer bikes than if it were not the weekend. Let's keep going, right? Why stop at the weekend? We also new day of week.

Here's a scatter plot of rentals versus the day of the week. You see the picture over here. How do we actually learn something like this? This one is actually a little bit tricky. This is technically not simple in your aggression. And why is it not simple in your aggression? Well, we think about how we're going to translate this to our model. Really the way we're going to think about this as we think about the day of the week. These aren't like sort of numerous things. It's a Sunday, and Monday, and Tuesday; I mean, there's sure they come one after the other, but each of them could have their own sort of properties. For whatever reason here in the Boston area, and so in particular as you look at these distinct days of the week, the way we're going to encode these is we're going to encode them as categorical variable. Okay. So, we're going to have a categorical variable potentially for Sunday, and a categorical variable for Monday, and a categorical variable for Tuesday, and so forth.

So, in other words, the number of dependent variables is not one anymore. There's sort of six independent variables. And you're wondering why not a seventh? Well, if all six are zero, the seventh is sort of automatically one. So, we don't need the seventh. And so, we have six of these dependent variables over here that seek to indicate to us which day of the week it is. And so, you could do exactly that. And by the way, you don't need to tell Python to go do this stuff for you in terms of a formula. The formula is still super simple. Okay, look at the formula. The formula we put in just simply says build me a model that predicts rentals as a function of the day of the week. We switched out temperature per day of the week. But now as we expected when we look at the model that we fit, we don't have two numbers anymore. We have an intercept as we did before, and then we have six potential slopes. Really six number, six offsets, depending on which day of the week it actually is. So, in particular, if it's a Monday, the offset is -22.221. If it's a Saturday, the offset is -57.2245, and so forth. And so, in other words, really out of this, we have a whole bunch of predictive models that have actually emerged. Seven. What are those seven models? The seven models basically tell us on a Sunday, what we're expecting is the intercept 225.24 minus the slope for Sunday, which if you look on the left was 71.0166 times whether or not it was a Sunday. So, if it's a Sunday, the 71.0166

gets multiplied by one and that's our prediction. Similarly, on a Monday same story, the intercept is 225.24. And if indeed it is a Monday, the slope in front of Monday is -22.22. And so our prediction of rentals on a Monday is simply 225.24 - 22.22. Now, just as a way to understand this, what I'd like for you to do is look at this for a second and tell me what's the busiest day of the week? What's our prediction of rentals on that busiest day of the week? So, take a second, go look at the output over here. If you're following along with the Collab, you can play around with the Collab as well. Go figure out what that busiest day of the week is, depend based on the output of this particular model. Write and calculate what we expect average rentals to be. So hopefully, what you did is you came back with Wednesday. You said, "Hey, when I look at this, Wednesday is clearly the busiest day of the week because on a Wednesday..."

And of course, I've deliberately not listed it, I am deliberately not showing you the answer; on a Wednesday, the intercept would be the same thing, 225.24. But if I look at the slopes, the slope would be the largest, it would be 10.21. And so in other words, on Wednesday we expect the rent 225.24 plus 10.2, roughly about 200 and, let's call it, 36; 235, 236 average rentals per hour on a Wednesday. Let's sort of take stock of where we are. Okay, we've blazed through this dataset. We set up a way of thinking where we were doing these sort of scatter plots, but we've kind of in a subtle way moved from having to plot everything and visualize this right to instead thinking about things in terms of this linear model. As opposed to simply scatter plotting things, we're now thinking about things in terms of a formula. That formula being as simple as rentals versus day of the week, or rentals versus temperature or something like that. We don't have to scatter plot all of this anymore. And we produced a whole bunch of different models in doing this. We produce rentals as a function of temperature. We produce rentals as a function of relative humidity, as a function of whether or not it was the weekend, the day of the week, and so forth. Now, here's why I want to actually pause for a second and reflect on what we've done. Each of these models gave us something interesting.

The temperature model, for instance, told us that as temperature goes up, we expect to rent more. The humidity model told us that as humidity goes up, we expect to rent less. The weekend model told us that on a weekend we expect to rent less. The day of the week model told us that depending on the day of the week, you have a certain day of week effect that actually moves the average number of rentals either up or down depending on the day of the week we're actually looking at. And one way of thinking about which of these is the best model, is simply looking at, you guessed it, the sum of the squared errors, the sum of the squared residuals for that specific model. Now, if you need to go rewind the lecture, get to the point where we talked about setting up linear regression, we produced this little plot that had just four data points on it. And we asked ourselves, as we rejigger that straight line fit, what are we trying to do? We're trying to minimize the sum of the squared residuals. Well, that sum of the squared residuals is our measure of how good our model is. And so, if that number small, good model; if the number is big, well not so great. And as we look at this I want you to digest what the baseline model brought to the table. The baseline model was simply not looking at independent variables at all. Just saying, "Hey, this is kind of the average I predict, 210 or something like that." And relative to that average model where the sum of the squared residuals was 443 million or whatever, something like that, some big number. What did each of these guys actually do? And from there, what I want you to ask yourself is which of these brings the most to the table; which of these brings the least to the table.

Video 5.8: Multiple Linear Regression (12:15)

So, we ran a bunch of these simple linear regression models, where we're looking at rentals versus one of these things' temperature, humidity, whatever. And I'd ask you to think about as we look at the sum of squared errors across these models, which one was bringing most of the table and you look at this and our gut was right. Like temperature seems to bring the most to the table. It seems to explain the most. We go from a sum of squared residuals are 443 million to a sum of squared residuals of 339 million. It's like a reduction of, let's call it 30%, in the uncertainty that we had to begin with. But we're greedy, we want the best model. So, if I look at something like this, I need to be asking myself each of these brings something to the table. Why can't I just look at all of them together? And that would be the right instinct. And what do we do after simple linear regression? Not complex, but the word for it is multiple linear regression.

The idea of being that now as opposed to looking at a single independent variable, we're going to look at multiple independent variables. So, just like before, I want to start with what the presumed underlying model over here actually is. Now, if you remember, and you can rewind the lecture to go look at this, if you like, for the simple linear regression model, we said the world was described by this dependent variable y, that looked like some intercept, some constant term beta_0 + beta_1 times independent variable and noise, was unexplained, the epsilon. All we've got to do in going from simple linear regression to multiple linear regression is add each of those new independent variables. So, if we have k independent variables, our underlying model is beta_0, the intercept, plus beta_1 * x_1, beta_2 * x_2 and so forth upto beta_k * x_k. And like before, we don't know what beta_0 is, we don't know what beta_1 is, and we don't know what beta_2, beta_3 all the way up to beta_k are either.

And our goal will be simply to look at the data and guess from the data an estimate for beta_0, an estimate for beta_1, an estimate for beta_2, which over here I'm calling b_0, b_1, b_2 and so forth. Now, here's the thing. With one of these independent variables, it was possible to scatter plot this and guess how best line fit and so on and so forth. What happens with 100? That's where the power of this formalism comes in. In other words, to some extent, finding a line through in the page in the plane is no different from finding a best linear fit to k of these things. All that we need to do is, you guessed it, minimize the sum of the squared residuals. That being the error between what this now model takes in k things predicts and the actual. That's it. That's going to be our game. Now, as we get into this, the natural question we're asking ourselves is the following. If you looked at the earlier slide, you were saying to yourself, the best model was rentals versus temperature.

Where should I go next? What is our multiple linear regression model? What are my independent variables going to actually be over here? And I have a choice. My choice could be any of temp, temp_wb, rel_humidity, windspeed, precipitation, and so forth. It could be any of these things. Now, let's say we've picked temperature, and one reason we might pick temperature is the following. What I'm showing you over here is a cross tab that gives me correlations between any one of these things. So, I want you to just focus on the first row. What does this tell you? First off, you see that under rentals, the number is 1.00. Why? Because any number is perfectly correlated with itself. So, rentals is perfectly correlated with rentals. No shock. But look at the rest of the guys though. We see a variety of different numbers. A 0.00, by the way, or a number close to 0.00, is like saying these things are basically not correlated with each other.

A number very close to 1.00 is saying, "Hey, these two things are really correlated with each other, and they're positively correlated." So, if one thing moves upward, the other thing tends to move upward. If you see a correlation number that's very close to -1, again, these are two things that are very correlated with each other but negatively correlated. So, that if one thing goes up, the other thing tends to go down. And looking at this, it makes perfect sense actually. That the model we picked was rentals versus temp because temperature is very correlated with rentals. In fact, it's the highest of all of these, 0.48. So, let's say that you actually have picked temperature. What would you want to pick next?

This is a natural question to ask in terms of what we might want to actually pick next. I want you to pause over here and think about this. The tool that we have in front of us, again is like a very simple tool. It's correlations. A correlation matrix in this particular case. And I believe that this correlation matrix is actually very useful as we do what we're doing over here, which is actually model selection. Now, hopefully you gave this some thought. And one temptation is to look at this and say, "Why don't I throw in temp_wb, wet bulb temperature, because that has a pretty high correlation. It's the next highest. It's 0.43. Once you did that though, hopefully, you paused over there for a second. You thought to yourself, wait a second. Sure, wet bulb temperature is heavily correlated with rentals, but is it giving me new information? Is it bringing something to the table that temperature isn't quite already bringing? And that should have given you some pause. That at least would give me pause. In particular, if I look at the correlation between temp and temp_wb, which you can actually look at if you look at the second row of this chart over here, the correlation between temp and temp_wb. You see that that correlation is 0.98. So, temp and temp_wb super correlated with each other. So, if I put in temp, do I really need to put in wet bulb temp?

Probably not. Because really what I needed to capture is already captured. It's already over there. Now we'll talk about this much more carefully later, when we talk about this concept of multi-collinearity. Let's forget about that for now. Just heuristically, intuitively say, I've put in temp. Temp_wb is very correlated with temp. So, it's probably not bringing a whole bunch to the table. What should I look at next? We'll we have rel_humidity, we have windspeed, we have precipitation and of those three, relative humidity at -0.17 has the biggest correlation in some sense. Let's bring in relative humidity. And so, that's what I'm going to do. And from a code perspective, to switch from simple linear regression to multiple linear regression is like typing in one more word. All we do is go back to that trusty OLS formula. Remember, OLS stands for ordinary least squares. What's actually being done to find that kind of best fit. And we change the formula.

We're changing the formula saying, "Hey, now we want a model that predicts rentals as a function of not just temperature but temperature and relative humidity as well." So, we do that, and we look at the model that comes out, and as we expect, there's actually three numbers over here. There's an intercept. The intercept is 52.95. There's the slope in front of temperature, which is 6.38, and there's a slope in front of rel_humidity, which is -2.79. And so, great place for us to pause, and now we've got a better model than before. Before, if we were asked to actually predict what rentals were going to be like tomorrow, we would simply ask ourselves, "Hey, what do we expect the temperature to be?" And if you told me, it was 75 F, this was an exercise we did earlier in this lecture, you'd go produce a forecast of at 75 F, this is what I'm actually expecting to rent. Now, we have a new model. And this new model has the power to account for multiple independent variables at the same time as opposed to accounting for just temperature.

I can collectively or jointly account for both temperature and humidity, and my model will give me a prediction that accounts for both. In this particular case, let's say, the temperature is 75, but the humidity is 80, what does your model now predict? Hopefully, you plug this in and what you got was rentals of 308 bikes an hour. The fact that it was 75 degrees helped us, warmer days, higher rentals. The fact that it was 80% humidity, that did not so much help us. Very humid days, we don't quite like that. So, let's take a step back. Let's take a step back and ask ourselves, where we are. So, when we started out, we said to ourselves, what are the various models we could build? model was a baseline model. We quickly moved away from that baseline model, and we built simple linear regressions. We could have built a simple linear regression just accounting for temperature and our sum of squared residuals would have been 339 million. If we did just rentals versus humidity, we know that that matters.

Our sum of squared residuals would have been 431 million. When we put the two things together, when we put together temperature and humidity, look at what happens. The sum of the squared residuals is now 316 million. In some ways, the sum quote unquote is better than any of the individual parts. Accounting for both temperature and humidity does a better job than accounting for just one of those two. And so, we go from a model that either give us 339 million or 431 million to a model that gives us a sum of squared residuals of 316 million. Now, I don't know about you, but there's something that bothers me about this way of thinking about the model. It's fine, by the way, it's all good. But it's weird to talk about 339 million and 500 million and 40 million and whatever because these numbers feel so specific to the blue bike problem, because they are, by the way, they are very specific to the blue bike problem. They're not portable numbers. So, if you go talk to someone and you say, "Hey, I got the sum of squared residuals of 316 million and I don't know what that means," sounds a little crazy. And so, this feels a little bit too contextual. We want in some ways, a relative way of talking about this. And so, that's what I want to develop next.

Video 5.9: Fit Quality: The Coefficient of Determination (R^2) (11:38)

So, we've actually built a bunch of models at this point. We built simple linear regression models. We built multiple linear regression models. And we were at this juncture, where I asked you to think about the quality of these models. We do believe that building a multiple linear regression model gives us a better fit to the data. We saw this when we looked at the sum of the squared residuals. But where that left us was in this weird spot where this measure of the sum of the squared residuals felt like an absolute measure, very specific to the problem at hand. And the challenge was to go to a measure which was portable, where it means the same thing across problems. And so, that's exactly what I want to introduce next. In particular, I'm actually going to introduce a quantity called the coefficient of determination or the R^2. You'll often hear people talk about the R^2 of a model. We're going to get into exactly what it is. So, here's really what the R^2 actually is. The R^2 needs to be, by design, some sort of relative measure. Well, relative to what is the natural question. And so, really, the basic idea is this. If I threw in my baseline model, our baseline model was simply predicting the average. That's our baseline. And our baseline has a certain quality to it. It has a sum of squared residuals associated with it. A natural question to ask is, "Look, whatever model you come up with, what is the sum of squared residuals for that model relative to the baseline? So, how much reduction are you actually going to bring about to the original uncertainty that the baseline actually had?

And so, really, making this a little bit more precise. So, the first thing is the error of the fit of our model, which at this point we're really familiar with, the sum of the squared residuals. What is the error of our model? What is the (y_i - y hat i)^2 across all of our data points? So, if we had n data points, in particular, we'd sum up the squared error across these data points for our model, would simply call that the sum of the squared residuals. The problem was we needed to normalize that. What are we normalizing it against? What we're simply normalizing this against is the same sum of squared errors. except against our baseline model. What does that baseline model do? Our baseline model, for every data point, i from 1, 2, 3 all the way up to n predicts the exact same number. That number is the mean, which I'm calling y-bar over here. So, if I think about our baseline sum of squared residuals, we're going to call that SST, for essentially the total variation of the model. That's r denominated. That's what we're going to normalize against. So, we got two numbers, the SSR, which is the sum of the squared residuals, and the SST, which is the total variation of the model, the sum of the squared deviations from the mean. And if you're thinking, "Hey, isn't that SST just the standard deviation?" You're super close. It's the square of the standard deviation. It's basically the variance. So, those are the two numbers we're actually going to look at. Given these two numbers, the R^2, the coefficient of determination, simply says how much of that original variability of that the SST model that the baseline model actually had, do you actually eliminate. How much of the original baseline uncertainty are you actually getting rid of? And so, it's a formula. Let's just part of the formula over here. R^2, the denominator is SST. That's the original variability. If I just always predicted the mean, my baseline is simply that. The sum of the squared errors, if I always predict the mean. What do I have in the numerator? What I have in the numerator is how much of that am I actually taking out.

So, I start with SST. My error is SSR. Imagine for a quick second that SSR was zero. That is to say we went from the original variability to zero variability. Well, you've removed 100% of the variability and so you're R^2 is one, you're perfectly determined. That's as good as it gets. On the other hand, what if your SSR, the SSR that you got with your model, was the same as your baseline? It was the same as the baseline. You're doing no better than the baseline. So, SST - SSR is zero. Your coefficient of determination is zero. You've eliminated 0% of the excess variability leftover relative to that baseline model, your model is no good. And so, as we think about that R^2, again, as we look at that formula at one, we've eliminated 100% of our variability. The worst we can get is we're no better than that baseline model, SSR is the same as SST and R^2 is just simply zero. So, given this tool of R^2, as opposed to talking about the quality of fit being 443 million or 339 million or one of these numbers. We can be much more numeric about it. In a way where we can talk about our model in general terms. And so, just sticking with the very first model we built today, the very first model we built in this session was rentals versus temperature. The sum of the squared residuals over there was 339 million. The sum of the squared residuals when we just predicted the mean, that's our baseline, that was 443 million. The SSR of the mean is the same as what we've been calling SST. That was 443 million. And so, we plugged these into our formula for R^2, the R^2, if the model is just looking at temperature (443-339) / 443, 23.5. But 0.235. What is this saying in words?

What this is saying in words is that we've eliminated 23% of the uncertainty that we actually had when we had just simply the baseline model. And by the way, you don't ever have to calculate this. If you come back to all that gobbledygook that was produced on your screen, every time you ran OLS, we were just looking at coefficients. I want you to now draw your attention to the thing that's at the top on the right. When you look at R, you got R^2 over there, very prominently. You don't ever have to go calculate this. You'll see that when we ran rentals versus temperature, it told us that the R^2 of this model was 23.5%. And now, you know exactly what that means. Now,

just to test our thinking over here, I want us to pause and just think about what we've done. Now, I have two questions for you. That will just serve as a test of are we getting this. So, the first question. If you're looking at R^2, what can the R^2 be? Is it between -1 and 1? Is it between 0 and 1? None of the above. And my second question for you, if you've nailed that first question is if you added a new independent variable to your regression model, what does that actually do? Can that only increase R^2? Can that only decrease R^2? Or can it both increase or decrease R^2? Before moving on, I want you to chew on these. I want you to really think about them because this is a good test of whether we're understanding this concept, whether we're rocking this concept or not. So, hopefully, you've had a chance to think about this and starting with that first question. What's the R^2 of a model? Well, let's just think about this.

One of the things we could always do in building one of these linear regression models is we could simply ignore all of the coefficients, all of these independent variables, force those coefficients to be zero. And just have the intercept. Well, if we're fitting just an intercept to the data, that's nothing but our original model. Our original baseline models. We'd simply put in the mean value over there. And that would give us an R^2 of zero. Why? Because if I actually think about the SSR for that model, it should be the same as the SSR for the baseline. The SSR for the baseline, minus the SSR for the baseline, that's just zero. That would give us an R^2 of zero. On the other hand, maybe, in putting in all these new independent variables, we build a perfect prediction of the world. Now, by the way later on, we'll see that's something to be like fishy. That's something scary. We don't like that. But in principle, that could happen. And if we've eliminated all of the uncertainty, the SSR is precisely zero. And so, the SST, the original variation minus zero, divided by the SST, that's one. And so, the R^2 is a number between zero and one. But as we think about this, this gives us this natural segue into the next question, which is, what's the value in a sense of adding a new independent variable to our model? As I just said, you add a new independent variable, you always have the option of simply ignoring it. You could put the coefficient in front of that, just set that coefficient to be zero. And if we set that coefficient to be zero, but we're back at whatever original model we actually had, we're no worse than that. And so, as a result, we can only increase R^2.

So, coming back to that question from earlier, if the R^2 could only improve, why don't I just end over here? And just simply say, "Just keep throwing in new variables. You're going to have the perfect model. Let's just declare victory at that point." Well, sadly, it's not quite as simple as that. It turns out that doing this thing where you keep adding variables and seeing that R^2 number go up, that has a few pitfalls to it. And as a good data scientist, we have to understand how to avoid those pitfalls. This isn't just a theoretical thing. This really goes towards building the best possible model we can build for use in the wild. And so, when we come back with our next session, we're going to pick up again on linear regression, but go into the subtleties that we need to go into that will equip us to understand what we need to do to build the best possible model rather, that we can build.