MITOCW | MIT15_071S17_Session_2.2.03_300k

Let's discuss the method Ashenfelter used to build his model, linear regression.

We'll start with one-variable linear regression, which just uses one independent variable to predict the dependent variable.

This figure shows a plot of one of the independent variables, average growing season temperature, and the dependent variable, wine price.

The goal of linear regression is to create a predictive line through the data.

There are many different lines that could be drawn to predict wine price using average growing season temperature.

A simple option would be a flat line at the average price, in this case 7.07.

The equation for this line is y equals 7.07.

This linear regression model would predict 7.07 regardless of the temperature.

But it looks like a better line would have a positive slope, such as this line in blue.

The equation for this line is y equals 0.5*(AGST) -1.25.

This linear regression model would predict a higher price when the temperature is higher.

Let's make this idea a little more formal.

In general form a one-variable linear regression model is a linear equation to predict the dependent variable, y, using the independent variable, x.

Beta 0 is the intercept term or intercept coefficient, and Beta 1 is the slope of the line or coefficient for the independent variable, x.

For each observation, i, we have data for the dependent variable Yi and data for the independent variable, Xi.

Using this equation we make a prediction beta 0 plus Beta 1 times Xi for each data point, i.

This prediction is hopefully close to the true outcome, Yi.

But since the coefficients have to be the same for all data points, i, we often make a small error, which we'll call epsilon i.

This error term is also often called a residual.

Our errors will only all be 0 if all our points lie perfectly on the same line.

This rarely happens, so we know that our model will probably make some errors.

The best model or best choice of coefficients Beta 0 and Beta 1 has the smallest error terms or smallest residuals.

This figure shows the blue line that we drew in the beginning.

We can compute the residuals or errors of this line for each data point.

For example, for this point the actual value is about 6.2.

Using our regression model we predict about 6.5.

So the error for this data point is negative 0.3, which is the actual value minus our prediction.

As another example for this point, the actual value is about 8.

Using our regression model we predict about 7.5.

So the error for this data point is about 0.5.

Again the actual value minus our prediction.

One measure of the quality of a regression line is the sum of squared errors, or SSE.

This is the sum of the squared residuals or error terms.

Let n equal the number of data points that we have in our data set.

Then the sum of squared errors is equal to the error we make on the first data point squared plus the error we make on the second data point squared plus the errors that you make on all data points up to the n-th data point squared.

We can compute the sum of squared errors for both the red line and the blue line.

As expected the blue line is a better fit than the red line since it has a smaller sum of squared errors.

The line that gives the minimum sum of squared errors is shown in green.

This is the line that our regression model will find.

Although sum of squared errors allows us to compare lines on the same data set, it's hard to interpret for two reasons.

The first is that it scales with n, the number of data points.

If we built the same model with twice as much data, the sum of squared errors might be twice as big.

But this doesn't mean it's a worse model.

The second is that the units are hard to understand.

Some of squared errors is in squared units of the dependent variable.

Because of these problems, Root Means Squared Error, or RMSE, is often used.

This divides sum of squared errors by n and then takes a square root.

So it's normalized by n and is in the same units as the dependent variable.

Another common error measure for linear regression is R squared.

This error measure is nice because it compares the best model to a baseline model, the model that does not use any variables, or the red line from before.

The baseline model predicts the average value of the dependent variable regardless of the value of the independent variable.

We can compute that the sum of squared errors for the best fit line or the green line is 5.73.

And the sum of squared errors for the baseline or the red line is 10.15.

The sum of squared errors for the baseline model is also known as the total sum of squares, commonly referred to as SST.

Then the formula for R squared is R squared equals 1 minus sum of squared errors divided by total sum of squares.

In this case it equals 1 minus 5.73 divided by 10.15 which equals 0.44.

R squared is nice because it captures the value added from using a linear regression model over just predicting the average outcome for every data point.

So what values do we expect to see for R squared?

Well both the sum of squared errors and the total sum of squares have to be greater than or equal to zero because they're the sum of squared terms so they can't be negative.

Additionally the sum of squared errors has to be less than or equal to the total sum of squares.

This is because our linear regression model could just set the coefficient for the independent variable to 0 and then we would have the baseline model.

So our linear regression model will never be worse than the baseline model.

So in the worst case the sum of squares errors equals the total sum of squares, and our R squared is equal to 0.

So this means no improvement over the baseline.

In the best case our linear regression model makes no errors, and the sum of squared errors is equal to 0.

And then our R squared is equal to 1.

So an R squared equal to 1 or close to 1 means a perfect or almost perfect predictive model.

R squared is nice because it's unitless and therefore universally interpretable between problems.

However, it can still be hard to compare between problems.

Good models for easy problems will have an R squared close to 1.

But good models for hard problems can still have an R squared close to zero.

Throughout this course we will see examples of both types of problems.