Hello I'm Prof.

Sridhar Narasimhan of the Scheller College of Business at Georgia Tech.

I'l be teaching this module on logistic regression,

in this course on data analytics and business.

There are six lessons in this module.

The first one introduces odds and how it is defined.

The second lesson talks about situations BDV,

that is the dependent variable that takes on zero or one values.

The third lesson shows how the logistic regression model is defined.

In the next lesson we use the default data set, and

then introduce four models for logistic regression.

Then, in lesson E, we talk about predictions and the confusion matrix.

And in lesson F, we talk about some metrics, sensitivity,

specificity, the ROC Curve, and area of the code.

So, on to odds.

Odds are one way to express the likelihood that an event will take place,

example, a horse winning a race.

Odds are written as X to Y,

or X:Y, or X/Y.

There's also the notion of gambling odds, so what are they?

They're called odds against.

So this means the probability that the event will not happen is

greater than that it will happen.

So in gambling, 10 to 1 odds means that if you bet $1 and

you win, you get paid 10 times $1, that is $10.

If you had bet $100 you'd win $1,000,

and you'd also get back your original bet.

However, in this lesson we'll deal with odds for, or odds on,

the probability that the event is more likely to happen than not.

So, 2 to 1 odds means that the event

is twice as likely to happen as not.

The gambler who bets at 2 to 1 odds and

wins will get their $1 back, and also his or her stake of $2.

We then look for odds for, and how it's handled in statistics.

In statistics, odds are viewed as a ratio of probabilities.

Odds is generally used as odds in favor of an event happening.

So odds for is a ratio, defined as a probability that the event will happen,

over the probability that the event will not happen.

So, if p is a probability that an event will happen,

then Odds (for) = p over (1-p).

Using the previous example, if Odds for

is 2:1, then Odds (for) = 2 over 1,

which is = p over (1-p).

Thus, following the steps here,

you get a value of p equal 2 thirds, or

0.6667, or 66.67%.

If you know Odds (for9, you can then compute the value

of p by using the equation showing here.

And here are some examples of odds, and

their respective p values shown in this table.

So if odds falls 2:1, then the p value is 0.67.

If it's 10:1, then the p value is 0.91.

If it's 1:4, then the p value is 0.2, and so on.

Here's an interesting question.

A betting site shows that the odds of the New England Patriots

winning the next Super Bowl is 5 to 1, Or 5/1.

Note that this site lists odds against.

So what's the probability of the New England Patriots

winning the next Super Bowl?

Is it 1%, 5%, 16.67%, or 25%?

So what's the correct answer?

So, in this case, the correct answer is C.

So, Odds for = 1/5 = 0.2, so

we know p = Odds (for) over 1 + Odds (for),

and you do the computation.

There's one more question.

Team Germany has a 12.5% probability of winning the next Soccer World Cup.

What's the odds for team Germany winning the next world cup?

Is it A, B, C or D?

What the correct answer?

So, the correct answer is B, p = 12.5%, or

0.125, Odds (for) = p over (1- p),

do the computations and you get 1 over 7.

In this lession we have learned odds and how they're expressed, and

how to convert them to probabilities for an event occurring.

[MUSIC]

In this lesson, we'll look at regression models where the y, or the outcome or

dependent variable, is binary.

That is, y has 0 or 1 as the only possible values.

We'll see what happens when you fit a linear regression model in one such case.

So relationship between Odds for and p.

You seen that there's a relationship between Odds for

and p, knowing p, we can get Odds for by using the equation.

And similarly, knowing Odds for, we can get the value for p using this equation.

There's some examples of situations where you have binary dependent variables.

For example in course whether a student will get an A in that course,

whether a firm will go bankrupt in the next year.

Whether a customer will make a purchase.

Whether a customer will default on his or her mortgage.

Whether a request for a loan.

Will it be approved or not, and you may have seen lots of other examples.

So I've created a synthetic data set called GradesR.csv.

You can download it and use it in R for this example.

So the way it works is that a grade of 1 means the student got an A on the exam and

a grade of 0 means that the student did not get an A.

Hours is another variable that refers to the amount of time that

a students spend studying for the exam.

So the task is to do a boxplot of Hours vs Grade,

run a regression of Grades on Hours, and

then do a scatterplot with Hours on the X-axis and Grades on the Y-axis.

So this is the boxplot that I get from the data set that I used.

Grades could be either 0 or 1 that is, And

then hours is on the y axis, the amount of time.

Students who got that specific grade, how much did they study.

I've used a linear regression model with grade as

the y variable and hours of study as the x variable.

And there are some questions that you get when you fit a linear regression model,

right?

Are all the predicted values either 0 or 1?

It doesn't seem to be the case.

Are some of the predicted values below zero, yeah?

Are some of the predicted values above 1, yeah?

All the predicted values lie on the regression line.

So the question is how does one predict a grade of 1 or

0 using these predicted values on the regression line?

So what should we do?

[SOUND]

So we go to the next lesson, where we define the logistic function and

develop the logistic regression model.

So logistic regression is similar to linear regression,

but with two main differences.

The Y variable is categorical, Yes/No, Approve/Reject,

Responded/Did Not Respond, Pass/Fail, etc.

And the second, the result is expressed as a probability of being in a group.

This implies that the predicted value is always between 0 and 1.

So I have fitted a logistic regression model to the grades example that I

introduced earlier.

Notice the shape of the logistic curve,

it is nonlinear, and its output lies between 0 and

1 for any value of hours, which is the x variable.

So we use the logistic function in logistic regression,

which gives us the probability of being in a group.

So we define p(x) is the probability that y = 1, given a value of x.

So we define the logistic function as shown here.

It's e to the b0 + b1x over 1 + the same term.

So p(x) has the property that it is always

between 0 and 1 for all values of x.

So I'm gonna use p instead of p(x) just to simplify the notation,

so p is given by that expression And

1- p is given by the expression there.

Therefore, p over (1- p) = exponent(b0 + b1x).

If you take the natural logs on both sides,

we get log(p/(1- p)) = b0 + b1x.

We have previously defined Odds(for) as p/(1- p).

Therefore, log of p/(1- p) is the log of odds for, or logit.

And the logit model is logit(p)

= log(p over (1- p)) = b0 + b1x.

All of the components of the regression model are the same.

So why transform from probability to log odds?

So why do we the transformation from probability to log odds?

Because it's usually difficult to model a variable which has restricted range,

such as probability.

The transformation is an attempt to get around this restricted range problem.

Because it maps probabilities ranging from 0 and 1 to log odds,

ranging from negative infinity to positive infinity.

Another reason is that, among all of the infinite choices of transformation,

the log of odds is one of the easiest to understand and interpret.

As we've seen earlier,

this transformation is called logit, and given by the expression here.

So interpreting the logistic regression model, we've seen this expression for

logit, and this means that as x increases by 1 unit,

the natural law of the odds increases by b1.

This is the same as the odds increasing by a factor of e to the b1,

which is roughly 100 times b1 percent.

Note that the exact odds changes (e to the b1- 1) times 100%.

This is a table with values for p with corresponding value for

odds and the natural log of odds.

As p ranges from near 0 to near 1,

odds keeps increasing from a very small value to very large value.

And log of odds keeps increasing from very small, and

negative values to large and positive values.

We could have also plotted those values here.

You see p on the x axis going from 0 to 1.

However, log of odds can go from negative to positive values.

If you plot log(odds) vs odds, this is what you see.

This is a typical log function.

So here's a quiz, so the logistic function p returns values between what?

A,B, C, or D, which one of these options is correct?

So the correct answer is D, the return value is between 0 and 1.

Here's another quiz, log(p over (1- p)) = b0 + b1x

means that as x increases by one unit, which one of these options is correct?

What's the correct answer?

So the correct answer is D.

A, B, and C are all correct.

So in lesson,

we defined the logistic function which gives us values between 0 and 1.

And then we needed that to develop the logistic regression model.

[MUSIC]

In this lesson, we use the default dataset, and

run four different logic models, and interpret their results.

So the default dataset is in the ISLR library.

It has 10,000 observations with 4 variables.

Default student balance and income,

I'm going to create a new data frame, which is a copy of default,

and then add two other variables or indicator variables.

One is dft and one is student stdt.

So dft is 1, if default equals yes, stdt equals 1

if the student variable is equal to yes, 0 otherwise.

I'm plotting income versus balance, so the defaulters are shown in blue.

This scatter plot shows generally that defaulters typically have higher balances.

I can also do a BoxPlot showing the distribution of credit balance for

non- defaulters who are in red and defaulters who are in blue.

So I'm going to look at four different

logistic regression models.

The first one is a model that has no predictor variables.

The second one has a single predictor variable,

it is a binary predictor variable stdt.

The third one has a single continuous predictor variable

where logit of p is regressed on balance.

And the fourth one has multiple predictors as shown you.

Again, p is defined as odds (for) over 1 plus odds (for),

a nd the probability that default equals yes,

and logit of p is log of p of 1 minus p.

So on to the first model, we have no predictive variables.

So this is the model, logit of p regressed

just on a constant and we get this table.

So the intercept from this model with no predictor variable is the estimated

log odds of being in default for the whole population of interest.

We can also transform the log of odds back to a probability.

So our model is this, and from this model,

we get b0 = -3.368.

Therefore, odds equals p over (1-p)

= exponent of that value which is 0.03447.

We know that p is this expression, therefore p =

0.03447 divided 1 + 0.03447 and

you get a value of point 0.0333.

So, 0.0333 is the probability that an individual

is in default, that's the probability of y = 1.

If you do a count of default = yes in the data frame,

you get 333 out of 10,000 records.

That is p = 0.0333, which matches the p that we are calculated above.

In the second model, we gonna have a single 0, 1 predictor variable,

Which is stdt, so this is the model as shown there.

Logit of p = b0 + b1 times the 0, 1 predictor variable, stdt.

And this is the result that we get, so

the intercept b0 = -3.50 is the log odds for

non-students since they are the reference group or

the base case of stdt = 0.

So odds for non-students = the exponent of that value, 0.03.

So for non-students, the probability of default =

Yes = odds over (1 + odds) = 0.0292.

Now, we want to calculate the probability of default for students,

so this is the results again.

The coefficient for students, b1 is that 0.40489,

is the amount that we have to add to b0 to get the log odds for students.

You get a value -3.499, so odds for

students is given by that expression there, 2.045.

So odds for students, the p value turns out to be 0.0431.

Therefore students have higher default probability than non-students.

On to the third model, so in this model, we have a single continuous predictor,

which is balance, and we get these result by fitting it in r.

Note that b1 = 0.0055,

an increase in the balance is associated

with increasing the log odds of default.

Hence the odds and hence the probability of default.

Adding one unit, that is $1 to the balance

increases the log odds of default 0.0055.

So making predictions using model 3, so

p(x) is given by that expression there, and you get the values of b0 and b1.

So if x = $1,000 then p(F) is 0.005,

which is less than 1%.

If x is $1,500, then p = 8.3%,

and if x = $2,000, then p(x) is 58.7%.

So for the logic model,

increasing x by 500 units has a nonlinear

effect on of probability of x as shown here.

We're on to the fourth model,

where we have multiple predictors in this logistic model.

So we have balance, income, and stdt.

So this is the model, and we have the values for b0, b1, and b2, and b3.

An increase in the balance is associated with increasing the log odds of default,

hence the odds, and hence the probability of default.

Adding one unit to the balance increases the log odds by about 0.0057.

The coefficient of student = -0.065 implies that a fixed value of balance and

income, students are less likely to default than non-students.

This is different from the result you saw in model 2.

So looking at the default rate for students and non-students,

at same credit card balance, an individual student will have a lower probability or

default than a non-student, looking at the two logistics curves.

On the whole, students carry more credit card balances, thus,

overall, student default is at a higher rate than non-student,

that's shown by the two horizontal lines.

This phenomena is called confounding.

And students on average have a bigger balance than non-students.

So there's some correlation between students and balance.

Take a moment to answer this quiz.

For the logistic model, logit p = b0,

where y = 1 is the default case,

what is the interpretation of b0?

Look, we don't have any predictor variables, so

what's the interpretation of b?

So the correct answer is B, so b0 is the log odds of being

in default for the whole population of interest.

In this lesson, we've seen four logistic regression models that used

the default data set, and I've learned how to interpret their output.

[MUSIC]

In this lesson, we'll see how created values of y are used to construct matrix

such as true positive, false positive, true negative, false negative.

And we're all to see how these matrix are capture in a confusion matrix.

This matrix will be use later on to gauge the efficacy of the logic models.

So we're interested in making predictions on the ferret data.

So after we fit a logistic regression, how do we do that?

We know that the predictive value of p will be between 0 and 1.

And we can use the predict function now to get the value of p.

So they can make predictions on the fitted data as shown here.

In this our code, and now we've created two new variables.

Predict the probability of model 4 and predict outcome of model 4.

And predict outcome is a binary variable 1 or

0 if the predict probability for

an observation is greater than .5.

So we're going to define true negative and false positives here.

So these are done by comparing the predicted Y variables

against the actual Y equals 0 observations.

So for each of these observations where Y equals 0,

we're using the logic model to make a prediction using the X values.

If the predicted value is 0, we get a true negative.

If the predicted value is 1, we get a false positive.

If the increase the cutoff value of p, the true negatives

will increase, and the false positives will decrease.

Seeing this graphically, so we have p on the x-axis,

and we're looking at the cases where Y=0.

And let's assume we have this distribution of these values.

So two negatives are the case where the predicted

model says you, I get a value of 0.

And for all the Y = 0 observations, when the predicted moral predict

0 we have two negatives, otherwise we have false positives.

If I increase the cut off value,

I increase the number of true negatives and

decrease the number of false positives.

Now I'm going to look at the Y=1 observations.

There will have to look at how true positives and

false negatives are defined.

So consider all the Y=1 data observations.

For each observation, you use a logic model to make a prediction.

If the predicted value equals 1, you get a true positive.

If the predicted value is = to 0, you get a false negative.

If we increase the cutoff, the false negatives have increased,

and the true positives will decrease.

You're looking at it graphically.

You're looking at all the Y=1 observations.

Assuming that's a distribution.

If the model critics in observation.

To have a value of p greater than the cutoff value you have true positive.

If not, it's a false negative.

And if they increase the cutoff, you see that false

negatives increase and true positives decrease.

So we're going to use these four matrix in the confusion matrix.

You may have seen this in some of your other courses,

it shows the intersection of actual and predicted values from a model.

So in our case, we have the actual values for default for 0 and 1.

You'll see there are 333 observations where

someone actually defaulted out of the 10,000 observations.

So a confusion matrix is where you record the performance of a classifier.

In this lesson we want to gauge how our logit models perform.

So in the default dataset we had 333 cases with default = "Yes" (or Y=1) and

the remaining with default = "No" (or Y=0).

We now have the row totals for this confusion matrix as shown above.

We need to record the predicted value or

y hat after we fit our logic model on this dataset.

So this is what I've shown you, the predicted values after running the model.

And you can do that using the xtabs function.

We have fitted model 4 and used the estimated

predicted p = .5 as a cutoff for setting Y hat to 1.

Otherwise, we set Y hat to 0.

So what is the count of True Negatives?

It's 9627.

What's the count of True Positives?

It's 105.

What;s the count of False Positives?

40.

False negatives, its 228.

So, in this lesson we've seen how predicted

values can be used these for matrix true positive,

false positive, true negative, and false negative.

And then how they can be used to

construct the confusion matrix.

[SOUND]

In this lesson, we see how the entries from the confusion metrics

are used to gauge the performance of the model.

An example, we'll assess the performance of the fitted training data.

So we have sensitivity, specificity, false positive rates.

I'm gonna define these metrics.

So sensitivity is true positive rate, is defined by that expression,

true positive over true positive plus false negative.

Specificity is the true negative rate,

true negative over true negative plus false positive.

False positive rate is 1 minus specificity.

So it's false positive over true negative plus false positive.

There are two more terms, precision is given by what's the probability

the actual observation is equal to 1 if the predicted value is equal to 1.

It's true positive over a true positive plus false positive.

And accuracy is straightforward,

it's true positive plus true negative over the total number of observations.

So relate to all this are type I and type II errors and

they are linked with the confusion matrix.

So false positive, a type I error,

is that you falsely reject the true null hypothesis.

A false positive occurs when the true value of y is equal to 0,

but the predicted value is 1.

You therefore increase the cutoff, decreases type I error.

False negative or a type II error occurs when

you incorrectly retain a false null hypothesis.

Remember that a false negative occurs when the true value of y is equal to 1,

but the predicate value is 0, increasing the cutoff increases type II error.

So there are type I and

type II errors which you encounter in business applications.

And the cost of these errors depends on the application for

which you're making predictions.

So assume you have marketing application, Y =1 means customers who will purchase.

So if you classify a non-purchaser,

there is Y=0 as a purchaser, Y hat =1 there is a false positive.

You typically have moderate loss for false positives or

type I errors in this marketing application.

Essentially, you're incurring some marketing

cost to folks who will not buy your product, for example.

Another example, let's say you have banking application and

Y = 1 means customers who will default, okay?

So suppose you have a customer who will default that is, Y=1 and

you classify them as a non defaulter, Y hat =0, it's a false negative.

This false negative, or type II error could be costly since

the customer's loan may have to be written off.

We had defined sensitivity and specificity, and we can compute it for

this confusion matrix which we saw earlier, where the cutoff value was p=0.5.

So we've fitted model 4 and used p = 0.05 as a cut off for

setting Y hat to 1, otherwise we set Y hat equals 0.

So sensitivity by definition is true positive over true

positive plus false negative.

So that's looks at the confusion matrix,

look rule 0 for Y=1, so true positive is 105 and

false negative is 228 so you get 105 over 333 is 0.32.

Specificity is you look at the row for Y=0 and

you look at true negative, true negative is 9,627 or

the total for that row and you get 0.996

You can use the same model, and use a different cutoff,

I'm using p =0.9, and you get a slightly different

confusion metrics and re-use the higher value of p.

So sensitivity, so in this case,

true positive goes to 10 over 333 and

specificity true negative rate is increased to 0.9998.

So increasing the cutoff value of p,

true negatives will increase and false positives will decrease.

Again, false negatives will increase and true positives will decrease.

So sensitivity is given by this, specificity is that, and

then we saw sensitivity decrease from 0.32 to 0.03,

and specificity increased from a 0.996 to 0.9998.

You can use the logit prediction in drawing the ROC curve or

Receiver Operating Characteristic curve in R, here's the code.

And you can also find what's called the Area Under the Curve.

So you can use the data from the confusion matrix

to draw the ROC curve, which is shown here.

The ROC curve is shown here for model four, this is used to show the diagnostic

ability of a binary classifier, as a cutoff value's increased.

We want the area under the curve and see what its value is,

and want it to be greater than 0.5.

If you notice in the ROC curve the x axis is the false

positive rate and the y axis is the true positive rate.

Here's a quiz, take a moment to answer these two quizzes.

First one, True or

false, sensitivity is 105 over 333.

Answer is true, it's true positive over true positive plus false negative.

Specificity is 40 over 9667, is it true or false?

Answer is false, because specificity is true

negative over true negative plus false positive.

In this module, we had several lessons, we looked at odds,

we looked at binary dependent variables situations.

We looked at the logistic function and logistic regression, and how it is setup.

We looked at using the default data set and fitted four different logic models.

We looked at some matrix, and then we looked at sensitivity,

specificity and the ROC Curve.

Thank you.

[MUSIC]