

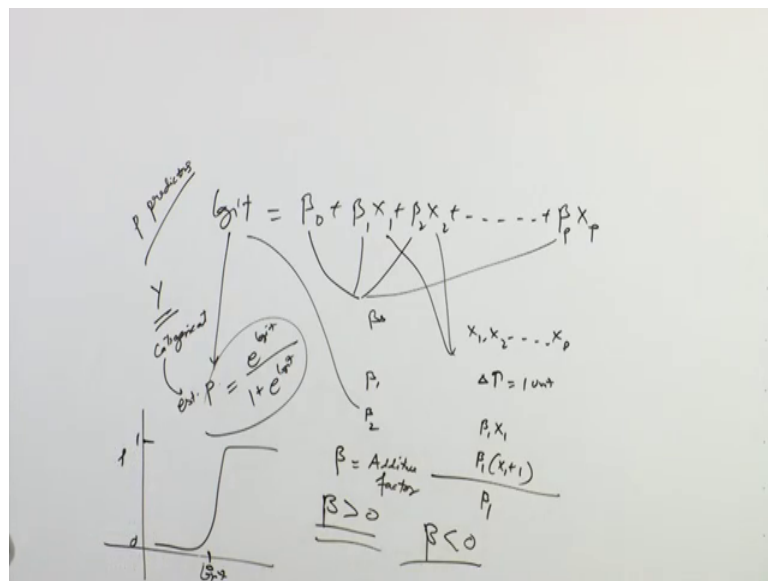
Business Analytics & Data Mining Modeling Using R
Dr. Gaurav Dixit
Department of Management Studies
Indian Institute of Technology, Roorkee

Lecture - 48
Logistic Regression-Part III

Welcome to the course Business Analytics and Data Mining Modeling Using R. So, in previous lectures we have been discussing logistic regression and in previous lecture we talked about different aspects of interpretation model interpretation in logistic regression.

So, we will discuss further on the same interpretation of results and to will further discuss the issues that could be there, because of the non-linear; a non-linear function and three different models that we typically use the logit model the odds model and the probability model. So, let us start.

(Refer Slide Time: 01:09)



So, in the logit model that we typically use let's write a particular logit model. So, if this is the model that we have let us say we have a P predictors. So, if we have P predictors and this is our model; so, how do we interpret results?

So, once we have built the model will have estimate of all these values. So, all the betas right all the betas. So, these estimates will have. So, these estimates we will have. So, how do we interpret the results, because as you know that the logistic regression model

is typically used for the classification tasks and we have already understood that the dependent variable outcome variable is categorical and so, we have already understood all these issues all right.

So, we look to estimate probabilities values right. So, we look to estimate probabilities values and for that; we need this particular formulation logit. So, once the model the estimates have been computed using these particular estimate we can use the direct values to compute the corresponding to compute the corresponding probability value as we saw in the a previous lecture as well right.

So, in the previous lecture as well this particular thing we saw there that the corresponding using the right value. We can compute the corresponding a probabilities value, now the interpretation remains slightly tricky. For example, here in this case if there is a unit increases in this X_1 , X_2 these predictors. So, we have you know you know unit increase a 1 unit 1 unit change in these values.

So, the corresponding change in logit values value would be based on these estimates. So, β_1 , β_2 right because simply you can see that β_1 , X_2 ; and if this was the a you know previous value and if there is a one unit change in the value of X_1 . So, from this you would see that the corresponding change in logit value is a going to be just β_1 .

So, this is the; so betas they are the additive factor the additive factor that actually changes. So, irrespective of you know the values of X the actual specific values of X the change in terms of change, if we look to interpret this in terms of change all depends on the beta values. So, that is the same thing is mentioned here. So, beta plays the role of an additive factor. So, if there is if there is a unit change in any of the predictors values the respect to beta change would be seen in the a logit values right.

So, increase in X would lead to corresponding increase in logit values; if beta is positive if beta is a negative. So, the direction would change and therefore, any increase in X that is predictor's values would lead to a corresponding decrease in logit values and in previous lecture itself; we have understood the relationship between logit and probabilities values right.

So, we had created this plot wherein we saw the relationship between P and logit values right. So, from this also we can understand that, if there is some change you know if there is one unit change in predictors values and beta is positive the logit values will; so increased. So, therefore, the probability values will logic value will increase. So, therefore, the probability value will also increase and that could lead to increase in acceptance of for example, the promotional offer example that we have been using.

So, did this increase in probabilities value will actually increase in the acceptance level of promotional offer right; however, as you can see from this particular curve you know, if the corresponding increases you know in this particular zone a you know zone, then probably the acceptance it might not reach to the acceptance you know a label it might not be classified as class one right.

This is the zone where the probabilities value significantly starts to change. So, this value is around 0 as we saw in previous lecture. So, you would see that the values logic value is close to 0, we see a sudden re a start to see sudden spikes in probabilities values and then finally, it becomes one right.

So, the interpretation; however, for the interpretation purpose if we are using logit model right. Logic values to interpret the results if beta is positive the one unit change in X_1 , X_2 the predictors value will have the corresponding change in the logit values and therefore, higher probabilities value and therefore, higher level of acceptance rate for promotional offer in this for example, that we have been using so, for any value of x .

So, for any value of X the interpretive statements of results they are going to remain same. So, increases either by this additive factor you know beta 1, beta 2 depending on the predictor. Now, let us move to the a odds model. So, in odds model the interpretation would change.

(Refer Slide Time: 07:53)

The image shows handwritten mathematical derivations for logistic regression. At the top, the probability P is given as $P = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p)}}$. Below this, the logit is defined as $\text{logit} = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$. The odds are then shown as $\text{odds} = e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}$. A note indicates that $\beta_1 > 0$ implies $\text{odds} > 1$ and $\beta_1 < 0$ implies $\text{odds} < 1$. The probability is also expressed as $\text{prob} = \frac{e^{\text{logit}}}{1 + e^{\text{logit}}}$. A 'Specific odds' is circled. The multiplicative factor is shown as $\text{Multiplicative factor} = e^{\beta_0} \times (e^{\beta_1 x_1}) \times (e^{\beta_2 x_2}) \times \dots \times (e^{\beta_p x_p})$. A note states that the multiplicative factor is in the range $(0, \infty)$. The final formulation is $\text{Specific odds} = e^{\beta_0} \times e^{\beta_1 x_1} \times e^{\beta_2 x_2} \times \dots \times e^{\beta_p x_p}$. A note at the bottom indicates that $\beta_1 > 0$ and $\beta_1 < 0$ are both possible.

So, let us look at the odds formulation now; so, if we go back to our odds formulation right. So, odds formulation if we go back it was e , and then minus minus of as we can see in this e and then this is the formulation e to the power β_0 , β_1 , X_1 , β_2 , X_2 up to β_p , X_p . So, if there are P predictors. So, if there are P predictors this is going to be the formulation odds model formulation right. And since we have already estimated these β_0 s; so these β_0 s have been already estimated right.

So, they can be directly so values of these you know coefficients the estimates can be directly plugged in into this model as well and we will have the odds model right. So, we can rearrange this we will have e^{β_0} , this is going to be like our you know constant a you know multiplicative factor and then well have e^{β_1} right. And we can write in this fashion X_1 and then we will have e^{β_2} , then we can write in this fashion X_2 and then we will have $e^{\beta_3} X_3$.

And in this fashion; and finally, we will have e^{β_p} for the P th predictor and X_p . So, you would see that this particular a model right. You would see for this particular model for a for a unit change, if we do a unit change in X_1 . So, if X_1 becomes; so I know if we do a unit change X_1 plus 1. So, you would see e^{β_1} and a X_1 plus 1. So, effectively what we get is e^{β_1} . So, that we get a multiplicative. So, all these are multiplications right.

So, this particular model is also the multiplicative models. So, what we actually get is multiplicative factor; e^{β} . So, for all the predictors X_1, X_2 to X_P you know, if there is one unit change in those predictors values the corresponding change in the odds of accepting the offer is going to be e to the power β and this is going to be the multiplicative factor.

So, that this particular value tends to the odds value will increase by a factor of this. So, earlier if the odd value was let us say 1.2. So, now, it will have a increase in e^{β} times 1.2, if all other variables and everything else is kept constant right; say if we look at the logit model that we discussed right. So, there it was the additive factors right.

. So, in the logit model we had the additive factor and you know if one unit change would increase the value the corresponding increase in logit values would be by β values now it would be. So, this would be plus you know that particular β now here it would be you know e^{β} this would be a multiplication. So, if there is an; if there is a unit increase in predictor's value.

So, the a corresponding increase in odds values would be by a multiplicative factor of e to the power β . So, increase in X_1 ; so let us go back to our slide. So, you can see here that if β is greater than 0. So, β is greater than 0. So, increase in X_1 would lead to increase in odds value right; and if β is less a than 0 then increase in X_1 will lead to decrease in odds.

However; as you see that this is an you know exponential formulation. So, you would see the values will range 0 to infinite. So, the values you know. So, therefore, even though the a β values even though even though the β is less than 0, the values will still remain greater than 0 for odds right; and a you would also see that a in the logit formulation that we had a logic formulation was like this in the logit formulation the values which are which are going to be negative.

So, the β which are going to be a negative; so they will become a value less than 1 between 0 and 1 and the β s which are going to be positive they will become values greater than 1. So, for example, if β_1 was positive and β_2 is was you know negative then this one would be this particular value would be greater than 1 and this particular value β_2 , 1 negative this would be less than 1; how less than 1; however, the values you know even this value as well would be greater than 0.

So, the negative β values there in logit model transform into smaller values in this odds model and positive values in logit model transform into a value greater than 1. In this particular model and the interpretive statements; so they can be still made with respect to the predictor value. So, irrespective of the for any value of X unit or one unit change in that particular value of X will lead to the corresponding increase of a multi by in multiplicative factor of e^{β} in the odds value. So, for both logit models and odds models, this is how we can go about the making interpreting the results.

However, if we look at the probability model right; however, we look at the probability models. So, the probability model is going to be like this. So, this was the; a probability model that we can use. So, here in the in this particular expression as you can see this particular equation this is this is also nothing, but logit right this is logit. So, we can the same thing as we have seen; before we can write it in this first a particular form and therefore, it will become logit divided by 1 plus e^{logit} .

So, in this form also we can like; so this was the model. So, from here you can see that if there is a unit change in X 1; right X 1 it is X 1 plus 1, then the corresponding a change in the probability value of P is not going to be constant, because the way the expression is right. So, the expression the β (Refer Time: 16:10) change in X 1 the corresponding change till P is going to depend on the actual value of X 1 right.

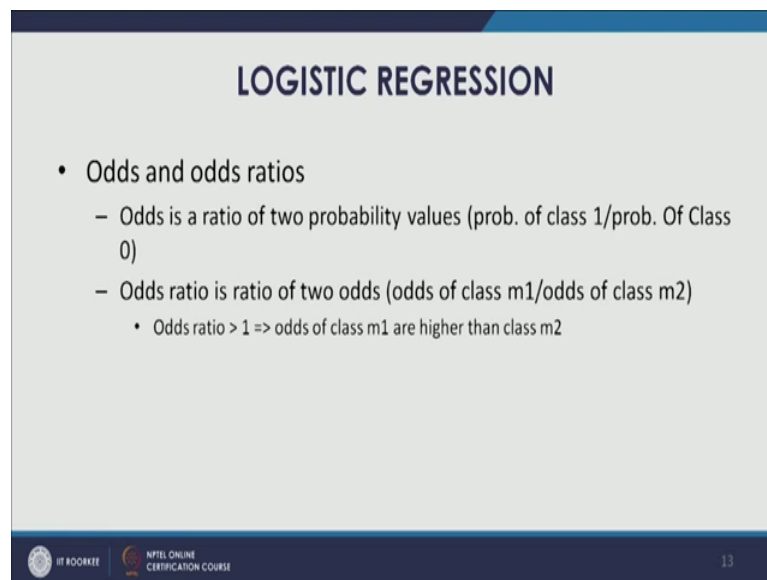
So, X 1 from this you know change in X 1 and the corresponding change in probability value. Now will depend on the actual value of X 1 as well; however, in other two formulation we can see a change in a one unit change in X 1 the corresponding change in logit was β 1, one unit change in X 1 here the corresponding change in odds values was e^{β} to the power β 1; however, in this particular case from this particular expression we can derive that a we can detect that the one unit change in X 1 and the corresponding change in probability value will depend on the actual value of X 1. So, we cannot eliminate X 1.

So, therefore, when we talk about the probability model the interpretation of the results would be for specific observations right. If the interpretation depends on the actual value of X 1 the change depends on the actual value of action. So, therefore, probabilities values β can be interpreted should be interpreted for specific observations, because one unit change in X 1 and the corresponding change in probability value will also

depend on the actual value of X as well. So, when we talk about the probability model we will discuss it in terms of a for with respect to the specific observations and general interpretation of predictors and their importance can be done either through logit model or odds model.

So, a there is another important aspect that; we would like to discuss here this is between odds and odds ratio.

(Refer Slide Time: 18:22)



LOGISTIC REGRESSION

- Odds and odds ratios
 - Odds is a ratio of two probability values (prob. of class 1/prob. Of Class 0)
 - Odds ratio is ratio of two odds (odds of class m1/odds of class m2)
 - Odds ratio $> 1 \Rightarrow$ odds of class m1 are higher than class m2

IT ROOKIE | NPTEL ONLINE CERTIFICATION COURSE | 13

So, in many domains these two terms are quite frequently used odds and odds ratio; however, they do not; however, these two terms are not same they are different. So, the particular term odds model, in this particular case logistic regression model that term that we have been using.

So, we had also given a what we mean by odd odds, in our case in logistic regression model it was the ratio of two probabilities value; that is a probability of a object a if probability of an a belonging to class 1 and probability of belonging to class 0. So, odds for our logistic regression model, it was probability of belonging to class 1 divided by probability of a class probability of longing to class 0.

However, another term odds ratio which is also popular in many domains; so which is slightly different. So, odds ratio is actually ratio of two odds, when we just say odds is it is the ratio of two probability values and when we say odds ratio it is actually the ratio of

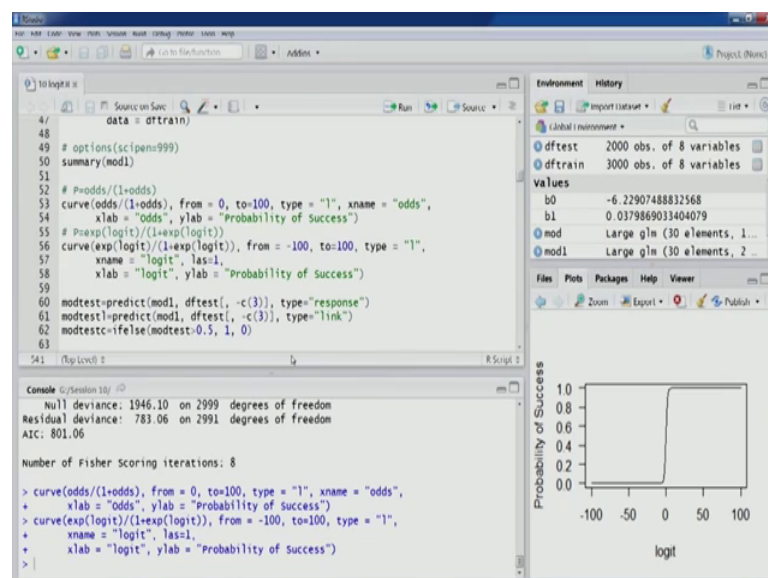
two odds. So, odds of for example, if we if we have a categorical outcome variable or categorical variable having m classes and there is m 1 class and there is m 2 class.

So, we can compare we can you know we can compute odds ratio a value for these two classes. So, it can be computed using ratio of odds of class one to odds of class m 2. So, odds of class m 1 divided by odds of class m 2. So, odds ratio is the ratio of two odds values and when we use just the term odds. So, it is ratio of two probabilities values.

In terms of interpretation odds ratio, when we say odds ratio greater than 1. So, for example, in this particular case odds of class m 1 are high we can say that odds of class m 1 or higher than odds of class m 2; however, in the in the in the case of just you know if we are saying odds and odds greater than 1, then we can say a that the probability of belonging to class 1; or the particular class is greater than the probability of belonging to class 0 right.

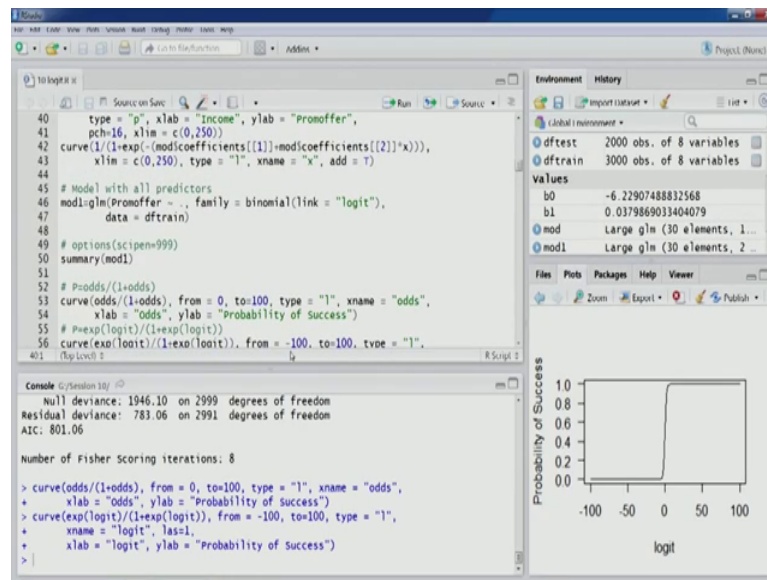
So, the interpretation of the definition of odds and odds ratios this difference we should be clear about. So, that there is no confusion when it comes to logistic regression model. So, what we will do?

(Refer Slide Time: 21:10)



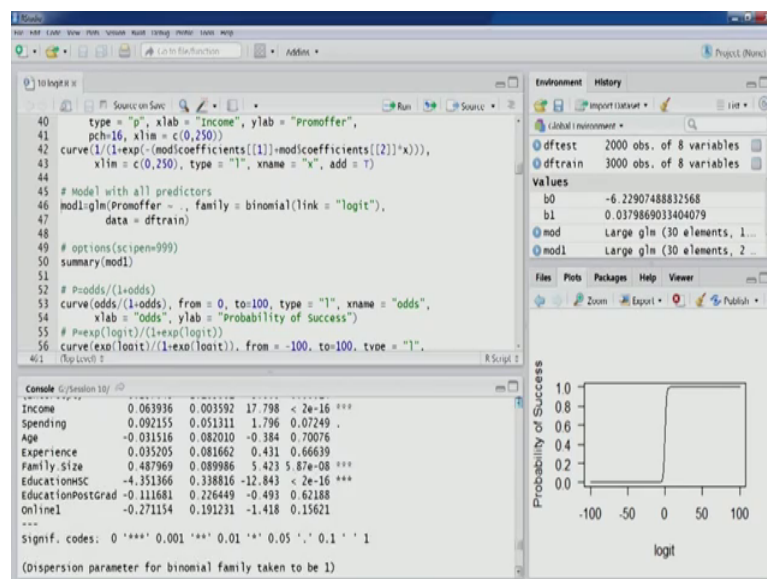
We will come back to R studio and the model that we have developed built using all the predictors.

(Refer Slide Time: 22:16)



So, this was the model mod 1. So, we had used the promotional offers data set and the promotional offer versus all the predictors. So, this model we had built and summary results as you can see in the output here.

(Refer Slide Time: 21:32)



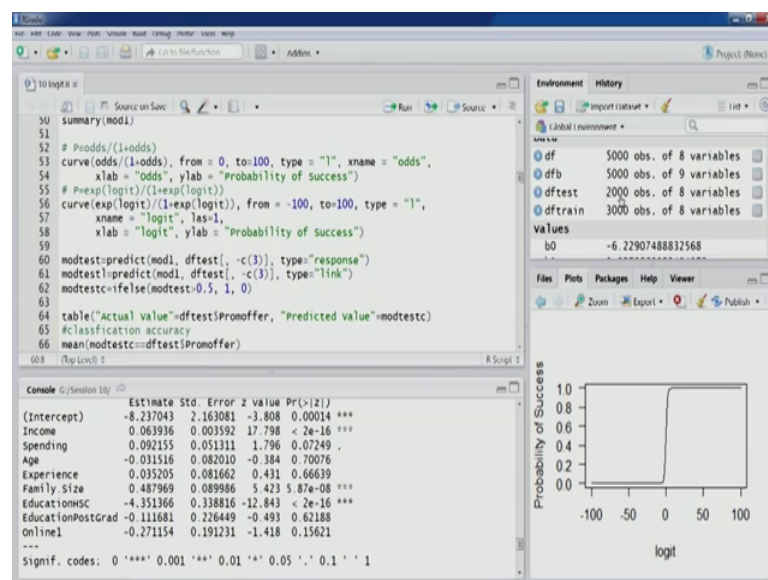
So, this is the model that we had built in the previous lecture; so we had already talked about these a predictors income and spending, and then family size and education HSC. So these being the significant predictors; and if we look at just the 99.9 percent

confidence interval the family size income and education HSC so, these three are the significant predictors.

So, we also in the previous lecture we also created these two plots probability versus odds and probability versus logit and we saw how from the odds model and logit model. So, using you know a by analyzing these plots we can understand further, how the odds model and logic a logit model can actually be used to make interpretation about probabilities values right.

So, a what we will a do now we will a score the training partition the test partition training and test partition using the model that we have just built. So, if we look at the test partition. So, in this as we can see the test df test that we had already created.

(Refer Slide Time: 22:59)



We have 2000 observations here. So, the model mod 1 will use the; so we are going to use the predict function to score of this particular partition. So, a model is mod 1 our logistic regression model with all the variables and, then this is the partition we are for the clarity we are not including the outcome variable for scoring the model, then we have this another argument type.

So, a this particular variable this particular argument indicates a gives us the values the estimated the logit values.

(Refer Slide Time: 23:42)

```

50 summary(mod1)
51
52 # P(odds/(1+odds))
53 curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
54       xlab = "Odds", ylab = "Probability of Success")
55 # P(exp(logit)/(1+exp(logit)))
56 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
57       xname = "logit", las=1,
58       xlab = "logit", ylab = "Probability of Success")
59
60 mod1$st.predict(mod1, dftest[, -c(3)], type="response")
61 mod1$st.predict(mod1, dftest[, -c(3)], type="link")
62 mod1$st.ifelse(mod1$st, 1, 0)
63
64 table("Actual value"=dftest$Promoffer, "Predicted value"=mod1$st)
65 #classification accuracy
66 mean(mod1$st==dftest$Promoffer)
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

	Estimate	Std. Error	Z value	Pr(> Z)
(Intercept)	-8.237043	2.163081	-3.808	0.00014 ***
Income	0.063936	0.003592	17.798	< 2e-16 ***
Spending	0.092155	0.051311	1.796	0.07249 .
Age	-0.031516	0.082010	-0.384	0.70076
Experience	0.035205	0.081662	0.431	0.66639
Family size	0.487969	0.089986	5.423	5.87e-08 ***
EducationMSc	-4.351366	0.338816	-12.843	< 2e-16 ***
EducationPostGrad	-0.111681	0.226449	-0.493	0.62188
Online1	-0.271154	0.191231	-1.418	0.15621

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

So, let us look at the help function, because we need to model gives us the logit values, then we need to yes compute the probabilities values from those logit values.

So, please look at the we look at the; this particular help section also for predict dot glm.

(Refer Slide Time: 24:01)

```

50 summary(mod1)
51
52 # P(odds/(1+odds))
53 curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
54       xlab = "Odds", ylab = "Probability of Success")
55 # P(exp(logit)/(1+exp(logit)))
56 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
57       xname = "logit", las=1,
58       xlab = "logit", ylab = "Probability of Success")
59
60 mod1$st.predict(mod1, dftest[, -c(3)], type="response")
61 mod1$st.predict(mod1, dftest[, -c(3)], type="link")
62 mod1$st.ifelse(mod1$st, 1, 0)
63
64 table("Actual value"=dftest$Promoffer, "Predicted value"=mod1$st)
65 #classification accuracy
66 mean(mod1$st==dftest$Promoffer)
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

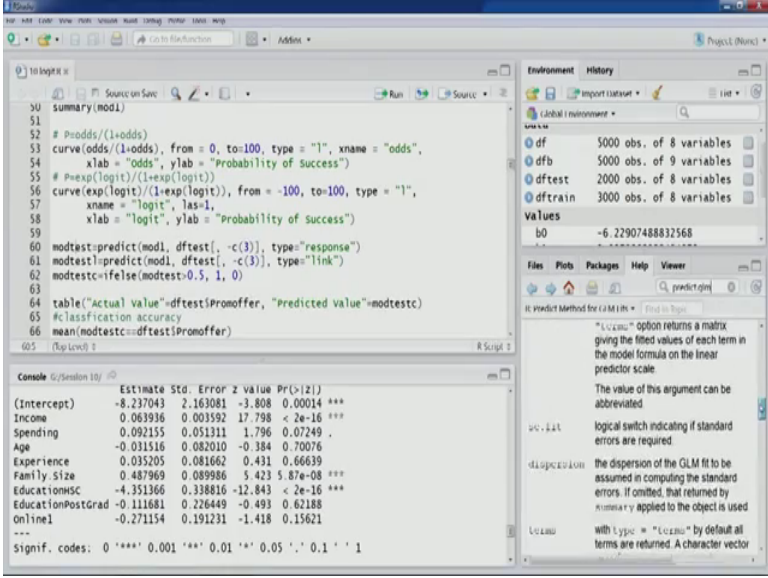
```

S3 method for class 'glm'
predict(object, newdata, type = c("link", "response",
na.action = na.pass, ...)

Arguments
object a fitted object of class inheriting from
"glm".
newdata optionally, a data frame in which to look
for variables with which to predict. If
omitted, the fitted linear predictors are

So, you can see that in predict we have a type and within type we have these; we can have these three options link response and terms right. So, as you can see first I have used response. So, let us see what these terms are about. So, we then type here; you can see the type of prediction required.

(Refer Slide Time: 24:19)



```
summary(mod1)
# P-odds/(1+odds)
curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
      xlab = "Odds", ylab = "Probability of Success")
# P=exp(logit)/(1+exp(logit))
curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
      xname = "logit", las=1,
      xlab = "logit", ylab = "Probability of Success")
modtest1=predict(mod1, dftest[, -c(3)], type="response")
modtest1=predict(mod1, dftest[, -c(3)], type="link")
modtestc=ifelse(modtest>0.5, 1, 0)
table("Actual Value"=dftest$Promoffer, "Predicted Value"=modtestc)
#classification accuracy
mean(modtestc==dftest$Promoffer)
```

	Estimate	Std. Error	Z value	Pr(> z)
(Intercept)	-8.237043	2.163081	-3.808	0.00014 ***
Income	0.063936	0.003592	17.798	< 2e-16 ***
Spending	0.092155	0.051311	1.796	0.07249 .
Age	-0.031516	0.082010	-0.384	0.70076
Experience	0.035205	0.081662	0.431	0.66639
Family Size	0.487969	0.089986	5.423	5.87e-08 ***
EducationMC	-4.351366	0.338816	-12.843	< 2e-16 ***
EducationPostGrad	-0.111681	0.226449	-0.493	0.62188
Online1	-0.271154	0.191231	-1.418	0.15621

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

So, the default is on this scale of linear predictors the alternative responses on the scale of the response variable right. So, the response variable in our case is logit right, because we have built logistic regression model. So, probably if we want to get the logit values will have to give this as an argument.

So, this is on a scale of response wherever thus and then for a default binomial model default binomial models the default prediction all of log odds and the probabilities on logit scale. So, you get the probability. So, actually from response we actually get the probabilities values right. So, if we want to get the probabilities values directly then we will have to use this response a type right.

And then we have terms option. So, this will return a matrix right this is will return a matrix is giving the fitted values of each term in the model formula right. So, these are three options that we have right, and then we have linked as well.

(Refer Slide Time: 25:48)

The screenshot shows the RStudio environment. The script editor contains the following code:

```

50 summary(mod1)
51
52 # P(odds/(1+odds))
53 curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
54       xlab = "Odds", ylab = "Probability of Success")
55 # P(exp(logit)/(1+exp(logit)))
56 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
57       xname = "logit", las=1,
58       xlab = "logit", ylab = "Probability of Success")
59
60 modtest=predict(mod1, dftest[, -c(3)], type="response")
61 modtest1=predict(mod1, dftest[, -c(3)], type="link")
62 modtestc=ifelse(modtest>0.5, 1, 0)
63
64 table("Actual value"=dftest$Promoffer, "Predicted value"=modtestc)
65 #classification accuracy
66 mean(modtestc==dftest$Promoffer)

```

The console shows the following output:

```

Residual deviance: 783.06 on 2991 degrees of freedom
AIC: 801.06

Number of Fisher Scoring iterations: 8

> curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
+       xlab = "Odds", ylab = "Probability of Success")
> curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
+       xname = "logit", las=1,
+       xlab = "logit", ylab = "Probability of Success")
> modtest=predict(mod1, dftest[, -c(3)], type="response")
>

```

The Environment pane shows the following objects:

- `dftest`: 3000 obs. of 8 variables
- `b0`: -6.22907488832568
- `b1`: 0.0379869033404079
- `mod`: Large glm (30 elements, 1...)
- `mod1`: Large glm (30 elements, 2...)
- `modtest`: Named num [1:2000] 3.01e-0...

So, let us use these options and see what the values are there so mod test. So, we look at the; first 6 values of this particular the these purpose scores. So, you can see for the training partition observations.

(Refer Slide Time: 26:08)

The screenshot shows the RStudio environment with the same script as before. The console output now includes the command `head(modtest)` and its result:

```

> head(modtest)
1      3      7     10     13     16
1.007609e-04 5.468725e-06 2.615117e-02 9.807546e-01 3.924793e-01 6.692638e-04

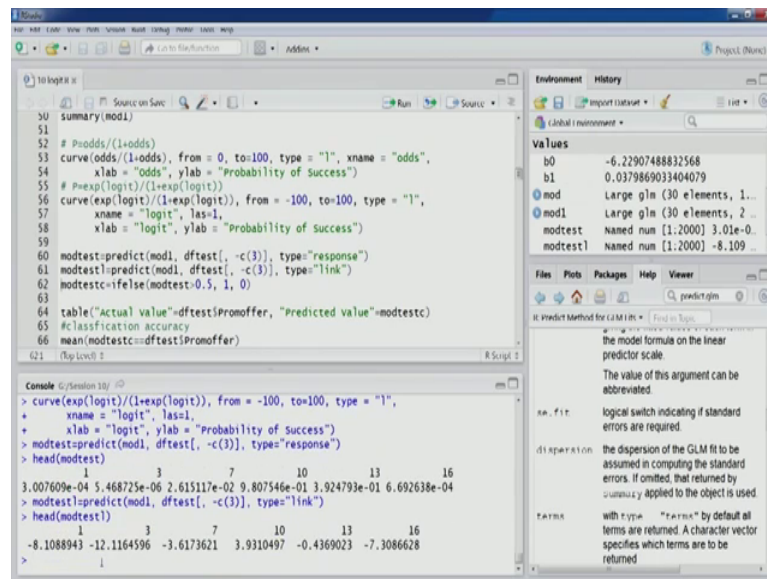
```

The Environment pane remains the same as in the previous screenshot.

We can see here the values. So, these are these are the probabilities values right. So, from these probabilities values these estimated probabilities values, we can then compute the we can then compute the determine the classification we can do our classification. So, the next one is mod test 1. So, here we are using the type as link.

So, let us compute this one as well. So, we look at the first 6 values of this one.

(Refer Slide Time: 26:43)



The screenshot shows an R Studio interface with the following components:

- Source Editor:** Contains R code for fitting a logistic regression model and generating predictions. The code includes comments and function calls like `curve()`, `glm()`, `predict()`, and `table()`.
- Environment/History:** Shows the objects created in the environment, including `b0`, `b1`, `mod`, `mod1`, `modtest`, and `modtest1`.
- Console:** Displays the output of the R commands, showing the coefficients of the logistic regression model and the predicted values for the test data.
- Viewer:** Shows the help documentation for the `predict` method for GLM models.

```
30 summary(mod1)
31
32 # Psodds/(1+odds)
33 curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
34       xlab = "odds", ylab = "Probability of Success")
35 # P=exp(logit)/(1+exp(logit))
36 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
37       xname = "logit", las=1,
38       xlab = "logit", ylab = "Probability of Success")
39
40 modtest1=predict(mod1, dfest[, -c(3)], type="response")
41 modtest1=predict(mod1, dfest[, -c(3)], type="link")
42 modtestc=ifelse(modtest>0.5, 1, 0)
43
44 table("Actual value"=dfest$Promoffer, "Predicted value"=modtestc)
45 #classification accuracy
46 mean(modtestc==dfest$Promoffer)
47
48 > curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
+       xname = "logit", las=1,
+       xlab = "logit", ylab = "Probability of Success")
+ modtest1=predict(mod1, dfest[, -c(3)], type="response")
+ head(modtest)
+ modtest1=predict(mod1, dfest[, -c(3)], type="link")
+ head(modtest1)
+
1 3 7 10 13 16
3.007609e-04 5.468725e-06 2.615117e-02 9.807546e-01 3.924793e-01 6.692638e-04
-8.1088943 -12.1164596 -3.6173621 3.9310497 -0.4369023 -7.3086628
```

So, from here you would actually see that a this link type actually gives us the logistic values here the logit values right. So, you can see the values already they are in the negative side and the earlier output that; we saw a these were probabilities values and the probabilities values are quite close to 0, and you can also see the corresponding logit values are negative right. And a as we move forward right, as we move forward you can see this particular observation the value start you know the; this particular observation this value is the logit value is positive and a here the probabilities value probability value is also close to 1.

So, in this fashion we can see the plot a that we had saw earlier. So, that is the same kind of results we are able to see here, now we just need probabilities values a the mod test values that; we have just computed to assign the observations to classify observations into different categories; So, if because this is a two class case.

So, if the default value the default cutoff value could be 0.5 to use the most probability class method right. So, we can use this for code. So, if else is the function that we can use an; and if the mod test value is greater than 0.5, then we can a assign that observation into class 1 and otherwise class 0.

So, this in this fashion will have the classification. So, let us compute this let us look at the first 6 values of this particular result.

(Refer Slide Time: 28:35)

The screenshot shows the R Studio environment. The script editor contains the following code:

```

51 # P=odds/(1+odds)
52 curve(odds/(1+odds), from = 0, to=100, type = "l", xname = "odds",
53       xlab = "Odds", ylab = "Probability of Success")
54 # P=exp(logit)/(1+exp(logit))
55 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
56       xname = "logit", las=1,
57       xlab = "logit", ylab = "Probability of Success")
58
59 modtest=predict(mod1, dftest[, -c(3)], type="response")
60 modtestl=predict(mod1, dftest[, -c(3)], type="link")
61 modtestc=false(modtest-0.5, 1, 0)
62
63 table("Actual Value"=dftest$Promoffer, "Predicted value"=modtestc)
64 #classification accuracy
65 mean(modtestc==dfest$Promoffer)
66 #misclassification error

```

The console shows the following output:

```

> head(modtest)
      1      3      7     10     13     16
3.007609e-04 5.468725e-06 2.615117e-02 9.807546e-01 3.924793e-01 6.692638e-04
> modtestl=predict(mod1, dfest[, -c(3)], type="link")
> head(modtestl)
      1      3      7     10     13     16
-8.1088943 -12.1164596 -3.6173621 3.9310497 -0.4369023 -7.3086628
> modtestc=false(modtest-0.5, 1, 0)
> head(modtestc)
      1      3      7     10     13     16
0 0 0 1 0 0

```

The Environment pane on the right shows the following objects:

- b0: -6.22907488832568
- b1: 0.0379869033404079
- mod: Large glm (30 elements, 1...)
- mod1: Large glm (30 elements, 2...)
- modtest: Named num [1:2000] 3.01e-0...
- modtestc: Named num [1:2000] 0 0 0 1...

So, you would see for a different these different observations as indicated in the indices, in the test partition the classification has been appropriately done you can see a first three values the probabilities values were also quite close to 0. So, all have been classified as 0 the a this for observation number 10. So, the probability value was close to a one. In this particular case as indicated in the logit values as well and this particular value has been this particular observation division has been classified as one class 1 and the others, because of the smaller probabilities values and negative logit values the classification has been done to class 0.

So, once the classification is done we can create our classification matrix. So, df test and promotional offers. So, this will have our actual values and the predicted values as we have computer just now mod test c.

(Refer Slide Time: 29:33)

```

55 # P = exp(logit)/(1+exp(logit))
56 curve(exp(logit)/(1+exp(logit)), from = -100, to=100, type = "l",
57       xname = "logit", las=1,
58       xlab = "logit", ylab = "Probability of Success")
59
60 modtest=predict(mod1, dftest[, -c(3)], type="response")
61 modtestl=predict(mod1, dftest[, -c(3)], type="link")
62 modtestc=ifelse(modtest>0.5, 1, 0)
63
64 table("Actual value"=dftest$Promoffer, "Predicted value"=modtestc)
65 #classification accuracy
66 mean(modtestc==dftest$Promoffer)
67 #misclassification error
68 mean(modtestc!=dftest$Promoffer)
69
70 head(data.frame("Predicted class"=modtestc,
71                 "Actual class"=dftest$Promoffer,

```

Console Output:

```

1 3 7 10 13 16
-8.1088943 -12.1164596 -3.6173621 3.9310497 -0.4369023 -7.3086628
> modtestc=ifelse(modtest>0.5, 1, 0)
> head(modtestc)
1 3 7 10 13 16
0 0 0 1 0 0
> table("Actual value"=dftest$Promoffer, "Predicted value"=modtestc)
      Predicted value
Actual value 0 1
0      1800 19
1       68 113

```

Environment:

```

b0      -6.22907488832568
b1      0.0379869033404079
mod      Large glm (30 elements, 1...
mod1     Large glm (30 elements, 2...
modtest  named num [1:2000] 3.01e-0...
modtestc named num [1:2000] 0 0 0 1...

```

So, we will have our classification matrix. So, as you can see in the matrix 1800 observations have been correctly classified as class 0, and then 113 observation have been correctly classified as class 1, and then we have these 68 and 19 which have been in a in a in correctly classified. So, we can compute the classification accuracy and error numbers for the same you can see.

(Refer Slide Time: 29:58)

```

70 head(data.frame("Predicted class"=modtestc,
71                 "Actual class"=dftest$Promoffer,
72                 "Prob for 1(success)"=modtest,
73                 "Log odds"=modtestl,
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

Console Output:

```

1 3 7 10 13 16
0 0 0 1 0 0
> table("Actual value"=dftest$Promoffer, "Predicted value"=modtestc)
      Predicted value
Actual value 0 1
0      1800 19
1       68 113
> mean(modtestc==dftest$Promoffer)
[1] 0.9565
> mean(modtestc!=dftest$Promoffer)
[1] 0.0435

```

Environment:

```

b0      -6.22907488832568
b1      0.0379869033404079
mod      Large glm (30 elements, 1...
mod1     Large glm (30 elements, 2...
modtest  named num [1:2000] 3.01e-0...
modtestc named num [1:2000] 0 0 0 1...

```

So, for the logistic model using the particular data set that we have built we get the 95.65 percent classification accuracy right; and the error is point error is 4.35 percent. Now, I

would like to a you know if you are able to recall that; previously we had used the same data set using classification and regression trees and the performance, that we saw there was you know for especially for training partitions the kind of performance that we saw there was around 98 percent.

So, the classification in regression three as we talked about in those lectures that are typically over fits the data. So, you can see here the performance is 95 and there it was 98 and when we had ruined the classification tree the performance had dropped down to 97, 96 a percentage.

However, if we look at them; So, if we look at this structure model like logistic regression the performance was performance close to 95.6 and we look at the data driven model like classification and regression tree the power the performance was you know something like 97. So, there is 2 percent increase, that we can clearly see in a data driven models. So, with this we will stop here and we will continue our discussion on logistic regression in coming lectures.

Thank you.