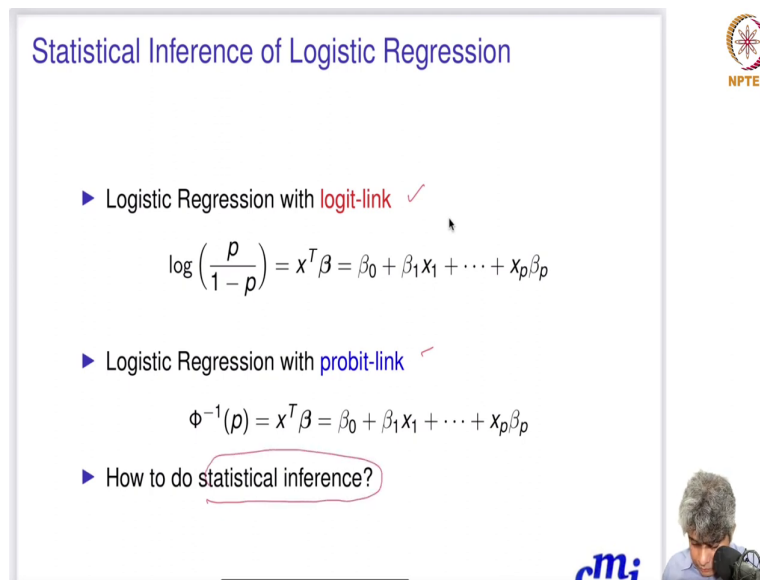


Predictive Analytics - Regression and Classification
Prof. Sourish Das
Department of Mathematics
Chennai Mathematical Institute

Lecture - 40
Statistical Inference of Logistic Regression


Welcome to the lecture 12 part A, on Predictive Analytics Regression and Classification course.

(Refer Slide Time: 00:25)



Statistical Inference of Logistic Regression

- ▶ Logistic Regression with **logit-link** ✓
$$\log\left(\frac{p}{1-p}\right) = x^T \beta = \beta_0 + \beta_1 x_1 + \dots + x_p \beta_p$$
- ▶ Logistic Regression with **probit-link** ✓
$$\Phi^{-1}(p) = x^T \beta = \beta_0 + \beta_1 x_1 + \dots + x_p \beta_p$$
- ▶ How to do **statistical inference?**




cm; 

We are going to discuss Statistical Inference of Logistic Regression. So, we in the previous lecture we have discussed the logistic regression with logit-link, and then logistic regression with probit-link, and then we also discussed how to do statistical inference?

(Refer Slide Time: 00:53)

Maximum Likelihood Estimates of β ✓

- ▶ The Maximum Likelihood Estimates (MLE) of β is
 - $\hat{\beta} = \underset{\beta}{\operatorname{argmax}} \mathcal{L}(\beta; \mathbf{y}, \mathbf{X})$ —
 - $= \underset{\beta}{\operatorname{argmax}} \ln \mathcal{L}(\beta; \mathbf{y}, \mathbf{X})$ —
 - $= \underset{\beta}{\operatorname{argmin}} [-\ln \mathcal{L}(\beta; \mathbf{y}, \mathbf{X})]$ —
- ▶ ✓ **Gradient Descent Algorithm** can be used to minimise negative log-likelihood function
- ▶ One can show
 - $\sqrt{n}(\hat{\beta} - \beta) \rightarrow \mathcal{N}(0, \Omega)$



Now, we then discussed maximum likelihood estimate of beta. How can you do compute that? We discussed about the likelihood function, maximizing likelihood function, maximizing log likelihood function and maximizing negative log likelihood function. All of them will give me the maximum likelihood estimates, one can use gradient descent algorithms to obtain the likelihood of functions.

(Refer Slide Time: 01:22)

β

- ▶ **Gradient Descent Algorithm** can be used to minimise negative log-likelihood function
- ▶ One can show $n \rightarrow \infty$
 $\sqrt{n}(\hat{\beta} - \beta) \rightarrow \mathcal{N}(0, \Omega)$

NPTEL

cmj

Asymptotic distribution of $\hat{\beta}$

- ▶ One can show



$\sqrt{n}(\hat{\beta} - \beta) \rightarrow \mathcal{N}(0, \Omega)$

One can also show that central limit theorem can nicely kick in the in case of maximum likelihood estimates that as n tends to infinity; that means, if you have large enough sample size, then beta hat with that sampling distribution of beta hat will behave like a normal distribution. So, this is a very important result.

(Refer Slide Time: 01:58)

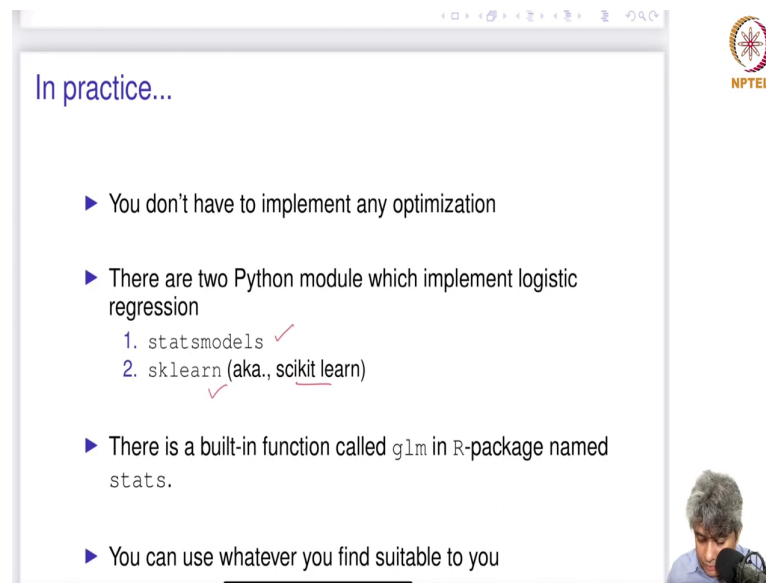
Asymptotic distribution of $\hat{\beta}$

- ▶ One can show
$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow \mathcal{N}(0, \Omega)$$
- ▶ As sample size n is large,
$$\hat{\beta} \sim \mathcal{N}(\beta, \Omega) \text{ approximately}$$
- ▶ We can run the statistical inference of logistic regression in the same way we did for simple linear regression.



Then, as here is a as sample size is n is large, approximately it means that sampling distribution will converge to behave like a normal distribution with centering around mean and some covariance matrix Ω , n comes from this you know scaling factor will take place.


(Refer Slide Time: 02:30)



In practice...

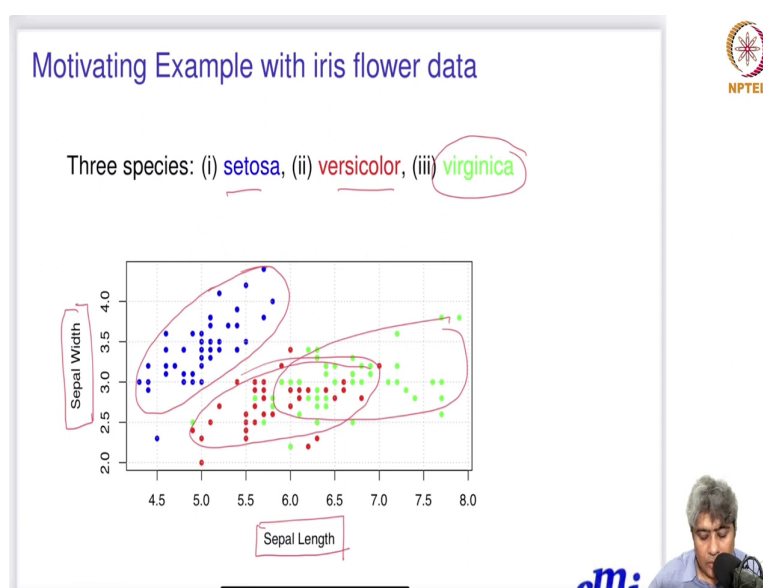
- ▶ You don't have to implement any optimization
- ▶ There are two Python module which implement logistic regression
 1. statsmodels ✓
 2. sklearn (aka., scikit learn) ✓
- ▶ There is a built-in function called `glm` in R-package named `stats`.
- ▶ You can use whatever you find suitable to you

NPTEL



Now, we can run statistical inference; that means, if we know that what is the sampling distribution of $\hat{\beta}$ then we can run the statistical inference of logistic regression in the same way we did for simple linear regressions. So, you do not have to implement any optimization as you saw last times that Python module with which implement logistic regression has two more package one is stat models, another is sklearn, also known as scikit learn package.

(Refer Slide Time: 03:10)






The built-in function called `glm` in R-package named `stats`, you can use whatever you find suitable to your; you know comfort zone. I am going to you know motivate you with one of the very very popular data set called iris flower data sets or fishers like RA fisher, who is the father of modern statistics. RA fisher he kind of collected this data in his agricultural lab.

So, this I am going to talk about this iris flower data sets, there are 3 species of iris; one is setosa, another is versicolor and this is virginica. Now, if you see the setosa and in on the X-axis we put sepal length, on the Y-axis we put sepal width you we see that you know the setosa are sort of you know completely separated versicolor is most point of versicolors is somewhere here and virginica is somewhere here. So, we see a reasonable overlap between virginica and versicolor.

(Refer Slide Time: 04:42)

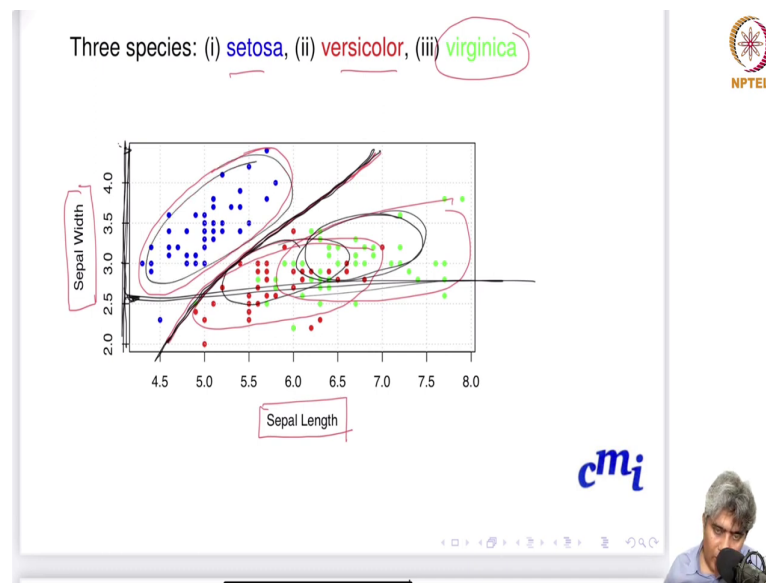
Identify *setosa*

- ▶ Given the sepal width of a flower we want to identify if the flower is *setosa*
- ▶ The model
$$y = \begin{cases} 1 & \text{if the flower is } \textit{setosa} \text{ with prob } p \\ 0 & \text{other species with prob } 1-p \end{cases}$$
- ▶ The logistic regression with logit link model will be
$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \text{sepal-width}$$



So, question is how we; question is identifying *setosa*, why? Because if we just have a; kind of a some kind of boundary, we can create a boundary here.

(Refer Slide Time: 04:59)



Essentially what logistic regression will do, logistic regression will create a boundary like this, if I have to identify, setosa then all you have to do just set up that boundary and. So, I am taking the easier problem in this case obviously, but you know and though I am taking identifying setosa will be ideal easy. But I think from understanding point of view, how it works will be easier if you start with the easy problem and then you move on to the difficult problem.

So, clearly identifying setosa will be easier than identifying versicolor or virginica. Now, given the sepal width of a flower, we want to identify if a flower is setosa ok. So, this is the idea. So, the model is like you know y equal to 1, if the flower is setosa with probability p and 0 if other species with probability 1 minus p ok. The logistic regression with logit-link model will be $\log \frac{p}{1-p}$ equal to β_0 plus β_1 sepal-width.

So, I am based on one predictor sepal-width can I identify setosa. So, that is a little difficult problem, because sepal-width is here and pretty much from 2 to 4, this 4.5 pretty much 2.5 to 4.5 everything is can be either say; so if I just make a cut off point like this a lot of points also fall in the virginica, versicolor also setosa.

(Refer Slide Time: 06:59)

Identify setosa

NPTEL


Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-15.7189 ✓	2.6217	-5.996	2.03e-09	**
Sepal.Width	4.7896 ✓	0.8246	5.809	6.29e-09	**

► The logistic regression with logit link model will be

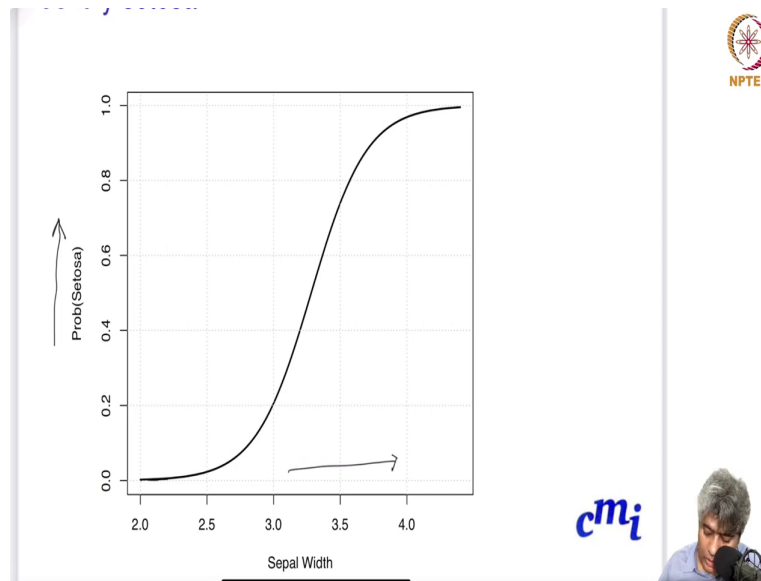
$$\log\left(\frac{p}{1-p}\right) = -15.72 + 4.79 \text{ sepal-width}$$

cmj



So, it will be slightly difficult problem, but let us see let us see. So, when we fit that model this model, I am talking about this model ok. When we fit this model in r, so my beta naught was negative 15.71 and beta 1 was 4.78, z value was very small or very large, p value is really really small. So, these coefficients are statistically significant

(Refer Slide Time: 07:39)




If we assume that normality will hold and you know like we have enough data to work on. If you believe that then what we are seeing is, that as the sepal-width increases the probability that possible flower belongs to the species setosa also increases.

(Refer Slide Time: 08:00)

Confusion Matrix

▶ The concept of **Confusion Matrix** is very popular in Machine Learning

	True y = 1	True y = 0
Predicted y = 1	correct	error 1
Predicted y = 0	error 2	correct



Now, next major thing is called confusion matrix, the concept of confusion matrix is very popular in machine learning. So, you do a out sample test, in the out of the sample test and you see true is y equal to 1, means it is setosa and y equal to 0 and predicted if it is y equal to 1 and y equal to 0. So, if it is y equal to 1, if it is indeed y equal to 1 and y equal to 1 then it is correct case or y equal to 0, y equal to 0 then it is also it is correct case.

Now, these two are correct case, but the problem is these two error 1 and error 2. So, there are two kinds of error. So, actually it is setosa, but model is saying it is not setosa that is error 2 and actually it is not setosa right and the model is saying it is indeed setose, so then it is a first type of error.


(Refer Slide Time: 09:10)


11 of 19

Confusion Matrix

► The concept of **Confusion Matrix** is very popular in Machine Learning

$$\begin{pmatrix} & \text{True } y = 1 & \text{True } y = 0 \\ \text{Predicted } y = 1 & \text{true positive} & \text{false positive} \\ \text{Predicted } y = 0 & \text{false negative} & \text{true negative} \end{pmatrix}$$

cmj 



So, this in sometimes in statistics language we call it true positive and true negative and false positive and false negative. So, false positive means the predict model is saying that yes, it is positive that it is setosa, but turns out that it is not setosa. So, the positive is false positive. Similarly, model is saying or your test is saying it is negative; that means, it is not setosa and; so it is negative result it is not and then you get a false negative. So, that is a false negative.

(Refer Slide Time: 09:55)

Confusion

Type I error
(false positive)

Type II error
(false negative)

You're pregnant

You're not pregnant

Source: <https://effectsizefaq.com>

NPTEL

cmj

So, question is whether it is indeed confusion, big confusion. So, I always use this cartoon of type 1 error, type 2 error and false positive and false negative. So, typically type 1 error is called false positive. Now, where what is it means type 1 error false positive. So, your test or your model is saying somebody that you did a test for pregnancy and test says you are pregnant and clearly this patient is not pregnant. I do not have to tell you, why he; he is not pregnant.

Now, another case possible case is type 2 error or second kind of error, which is typically called false negative. In this picture you can see that a test is being given and the doctor is saying you are not pregnant; so that means, and clearly you can see that she is pregnant. So, this is false negative.

So, so your test is saying it is negative result; that means you are not pregnant, but that is a wrong result, right. So, this is how I try to remember what is false positive or type 1 error and what is false negative, what is type 2 error.

(Refer Slide Time: 11:21)

precision, recall, F-measure

- ▶ precision is fraction of correct positive prediction

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$



- ▶ recall is the fraction of correct prediction among the all true positives

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

- ▶ F measure is the harmonic mean of precision and recall,

$$F = 2 \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

- ▶ Measures based on confusion matrix are specific to binary classification problem



There are some other matrix which are very popular in machine learning, called precision recall and F measure. Precision is a fraction of correct positive prediction. So, typically it is true positive divided by true positive plus false positive like, how many positive are being predicted and how many are true positive out of the how many are predicted. And recall is; so true and recall is true positive divided by true positive by false negative.

So, among the true cases, so how many of them are positive? So, recall is the fraction of correct prediction among all true positive and precision is fraction of correct positive prediction, how many correct positive prediction you made and among the actual true positive

how many correct prediction is made. And F measure is a harmonic mean of the precision and recall.

(Refer Slide Time: 12:34)

Model Selection with Akaike Information Criterion




- ▶ The logistic regression with logit link model will be

$$\log\left(\frac{p}{1-p}\right) = \mathbf{X}\beta$$

- ▶ Let k be the number of predictors whose corresponding coefficients needs to be estimated.
- ▶ Let $\mathcal{L}(\hat{\beta}, \mathbf{y}, \mathbf{X})$ be the maximum value of the likelihood function for the model. The AIC value of the model is:

$$AIC = 2k - 2 \ln(\mathcal{L}(\hat{\beta}, \mathbf{y}, \mathbf{X}))$$

- ▶ Given a set of candidate models for the data, the preferred model is the one with the minimum AIC value






So, these are the very popular measure of how good a model is from machine learning literature and from statistics literature, model selection with akaike information criteria is most popular logistic regression with logit-link will be like this and then let p be the number of the predictors, sorry I should have used something different maybe q or k , I should use k because p I have already used for probability.

So, k ; suppose k is the number of predictors whose corresponding coefficient needs to be estimated. Then you just calculate the $\mathcal{L}(\hat{\beta}, \mathbf{y}, \mathbf{X})$ with the maximum value of the likelihood function of the model and you can calculate the AIC.

(Refer Slide Time: 13:31)

Advantage of AIC

- ▶ One of the significant advantage of AIC is this measure is problem and model agnostic
- ▶ It means for any problem if you can precisely write down the likelihood function then AIC estimation is automatic for that model
- ▶ **The AIC is an estimator of out-of-sample prediction error. Therefore it represents the quality of statistical models for a given set of data.**



Once you calculate the AIC, given set of candidate models for the later preferred model with one minimum AIC is the one model that has the minimum AIC. So, what is the advantage of AIC? One of the significant advantage of AIC is this measure is problem and model agnostic ok. For any problem any model you can try AIC, it means for any problem if you can precisely write down the likelihood function then AIC is automatic for them for that model.

AIC is an estimator of out sample prediction error; therefore, it represents the quality of statistical models for given set of data. So, this is a very important thing ok.

(Refer Slide Time: 14:23)




As a single-layer perceptron

▶ The logit model has an equivalent formulation

$$p = \frac{1}{1 + \exp\{-(\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p)\}}$$

This functional form is commonly called a single-layer perceptron or single layer- artificial neural network.

deep neural






A single-layer perceptron the logit model has an equivalent formulation which is $p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p)}}$ this functional form is commonly called single layer perceptron or single layer artificial neural network. So, logistic regression is the brick which are being used to develop the neural network or deep neural network.

And going forward we will talk about it how you can use; you can think logistic regression models sort of a LEGO and you can like put one LEGO with another LEGO and create you know put lots of LEGO and create a beautiful artifact out of that. So, similarly, neural network each component of a neural network can be view as a perceptron, single layer perceptron and model and single layer perceptron model is nothing but a simple logistic regression.

(Refer Slide Time: 15:45)

Multicollinearity Issue in Logistic Regression

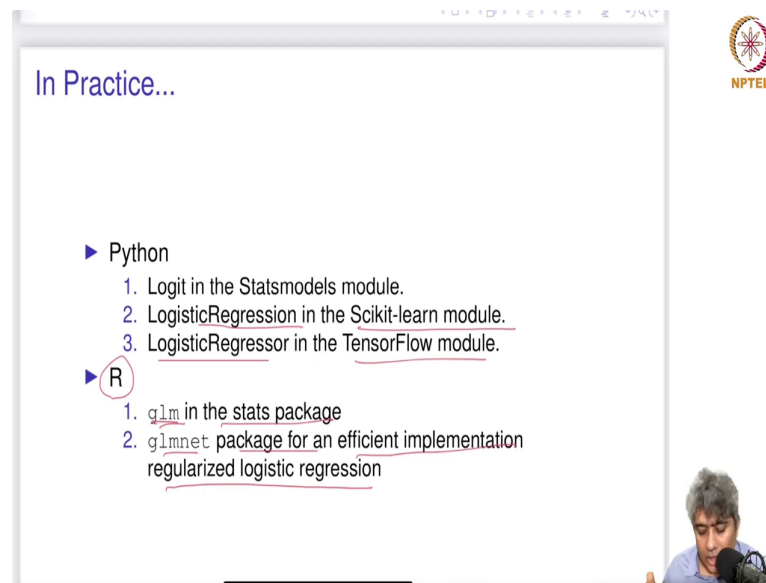
- ▶ Logistic regression can suffer from the multicollinearity issue
- ▶ In such cases, you have to apply ridge correction that we applied in case of simple linear regression.
- ▶ Unfortunately straight forward mathematically closed form solution for logistic regression is not available
- ▶ However, the Python and R packages are available, which implement the Ridge correction for logistic regression very efficiently.

Can logistic regression have the multicollinearity issue? Yes, logistic regression can suffer from logistic multicollinearity issue; you can; so, you have to be very careful you should check, if it does have the multicollinearity issue. In such case you have to apply ridge correction that we applied in case of simple linear regression.

Unfortunately, straightforward mathematically closed form solution for logistic regression is not available. However, Python, R packages are available which implement ridge correction for logistic regression very efficiently. So, you do not have to worry about those things.

(Refer Slide Time: 16:30)

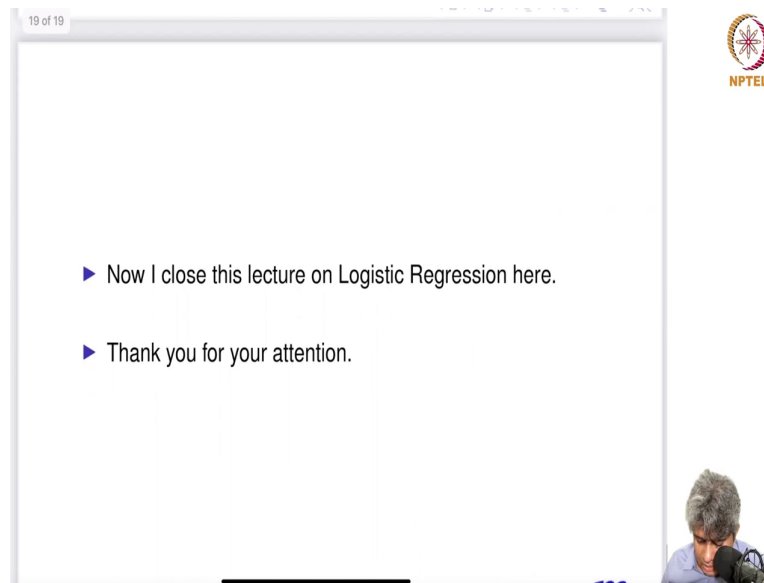


In Practice...

- ▶ Python
 1. Logit in the Statsmodels module.
 2. LogisticRegression in the Scikit-learn module.
 3. LogisticRegressor in the TensorFlow module.
- ▶ R
 1. glm in the stats package
 2. glmnet package for an efficient implementation of regularized logistic regression

So, implementation part you do not have to really worry. So, in practice logit in the stats model there is a logistic regression in Scikit-learn module, logistic regression with tensor flow module. So, yeah in glm in R, you have glm in stats package, glm net package an efficient implementation of regularized logistic regression for to handle multicollinearity is already available. So, you do not have to worry about that.

(Refer Slide Time: 17:03)



19 of 19

- ▶ Now I close this lecture on Logistic Regression here.
- ▶ Thank you for your attention.

NPTEL

A small inset image of a person speaking into a microphone is visible in the bottom right corner of the slide frame.

Now, I close the lecture on logistic regression here.

Thank you for your attention.