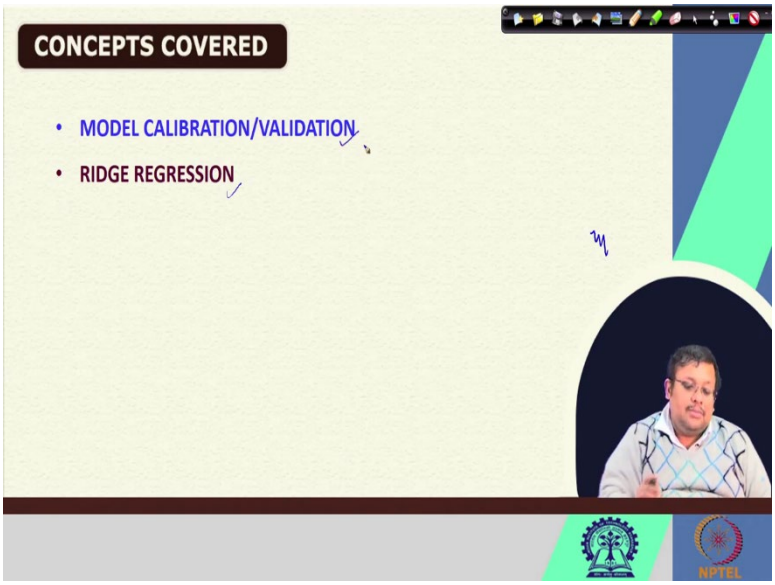
Machine Learning for Soil and Crop Management
Professor Somsubhra Chakraborty
Agricultural and Food Engineering Department
Indian Institute of Technology Kharagpur
Lecture 31: ML and DL for Soil and Crop Image Processing

Welcome friends to this NPTEL online certification course of Machine Learning for Soil and Crop Management, and today we are going to start week 7. And this week 7, the first, the topic of this week 7 is machine learning and deep learning for soil and crop image processing. So, in this week, and in our next week, we will be mostly focusing on the application of different machine learning, and deep learning methods for soil and image crop processing, crop image processing, and how these image processing has become an advance tool for prediction and characterization of multiple soil and crop properties we are going to learn.

(Refer Slide Time: 01:38)



KEYWORDS

- LOOCV
- RIDGE REGRESSION
- Holdout
- K-fold validation
- L2 regularization

MODEL CALIBRATION/ VALIDATION

- Two types
  - Additional independent dataset (unbiased)
    - Simple random sampling
    - Stratified random sampling
  - Data sub-setting
    - Random hold-back/holdout
      - 70% calibration
      - 30% validation
    - K-fold validation
    - LOOCV

MODEL CALIBRATION/ VALIDATION

- Two types
  - Additional independent dataset (unbiased)
    - Simple random sampling
    - Stratified random sampling
  - Data sub-setting
    - Random hold-back/holdout
      - 70% calibration
      - 30% validation
    - K-fold validation
    - LOOCV

So, in this first lecture, lecture number 31, we are going to talk about the two important concept. One is model calibration and validation, as you can see in this slide, model calibration and validation and secondly, we are going to discuss another important regularization method to deal with the overfitting and multicollinearity that is Ridge Regulation. So, first, we are going to discuss the about model calibration and validation, and secondly, we are going to talk about this ridge regulation. So, let us start.

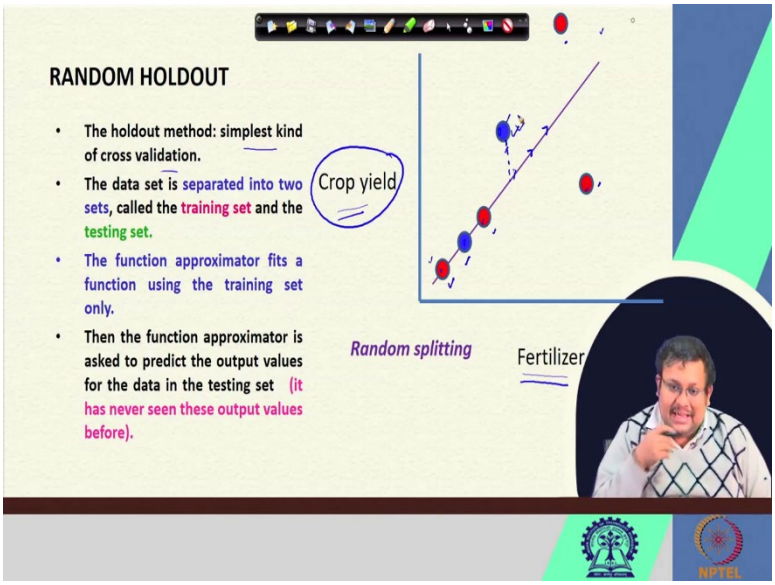
And also these are the keywords, which we are going to discuss in today's lecture. LOOCV, which is a short form of leave one out cross validation, then ridge regulation, holdout, K-fold validation and L2 regularization. So, these are some of the keywords which we are going to discuss in this lecture number 31.

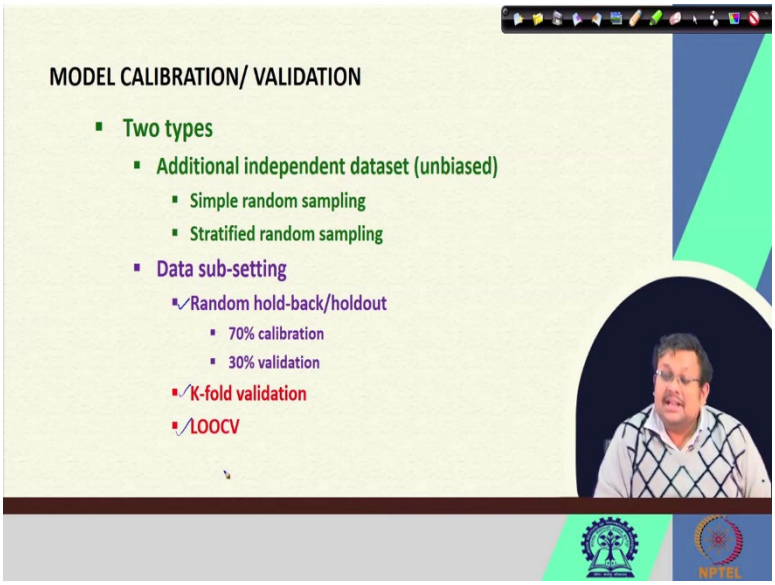
So, you all know that we talk about these calibration model and validation statistics previously in our previous lectures. So, it is important to understand what do you mean by calibration and validation? And secondly, it is important to understand what are the general methods for calibration and validation? So, it is always desirable that when you have the data and you are building a model you are developing, you are learning a relationship from the data, you are developing a model this model should be tested for predicting the independent or unknown samples.

So, the built model is known as calibration model, and validation is basically testing the efficacy of that model. So, the calibration and validation are that complementary and indispensable parts of the modeling exercise. Without validation you cannot make a conclusion about the importance or about the utility of the calibration model. So, the question comes to our mind, so what are the two you know different types of calibration validation strategies when you deal with data set.

So, there are generally two types of validation, we generally talk about. One is, independent additional independent data sets. So, we develop a calibration model and then we validate or test that model using an independent data set. And this method is ideal method which is an unbiased method. And these additional sample selection can be done by either simple random sampling or stratified random sampling. But the problem is sometimes our resources are limited and it is not possible for us for getting an additional independent data set.

(Refer Slide Time: 05:21)





So, most of the time, we are bound to develop our validation data set from the whole dataset, which we already have. So, then we have to take the help of data sub-setting we have to take the help of data sub-setting let me go back to the previous slide. So, these data sub-setting is another important part of model validation, there are three types of data sub-setting one is a random holdout or hold-back and the second one is k-fold validation and the third one is leave one out cross validation or LOOCV. So, today we are going to discuss about these all these three types of data sub-setting methods and their advantages.

So, let us consider that we have a total of this these 6 we have these data set of these 1,2,3,4,5,6 different points, and so, we plot. Here the, our independent variable is fertilizer and our dependent variable is crop yield. So, we want to predict the crop yield based on the

fertilizers. So, in the holdout method, what we do. Let me just move it here for better understanding.

So, you can clearly see that the whole data set which is consist of 6 points, you know 1,2,3,4,5,6 one-third of the data has been selected as the validation set and two-third is utilized as the calibration data set. So, here these rate points at the calibration data set and these one-third that is these blue points are validation data set. Remember, the sub-setting is being done in a random way, but at the same time, we should see that the range of this model should be bounded by the calibration by this calibration data set otherwise, what will happen our validation data will go beyond this calibration range and that would create extrapolation.

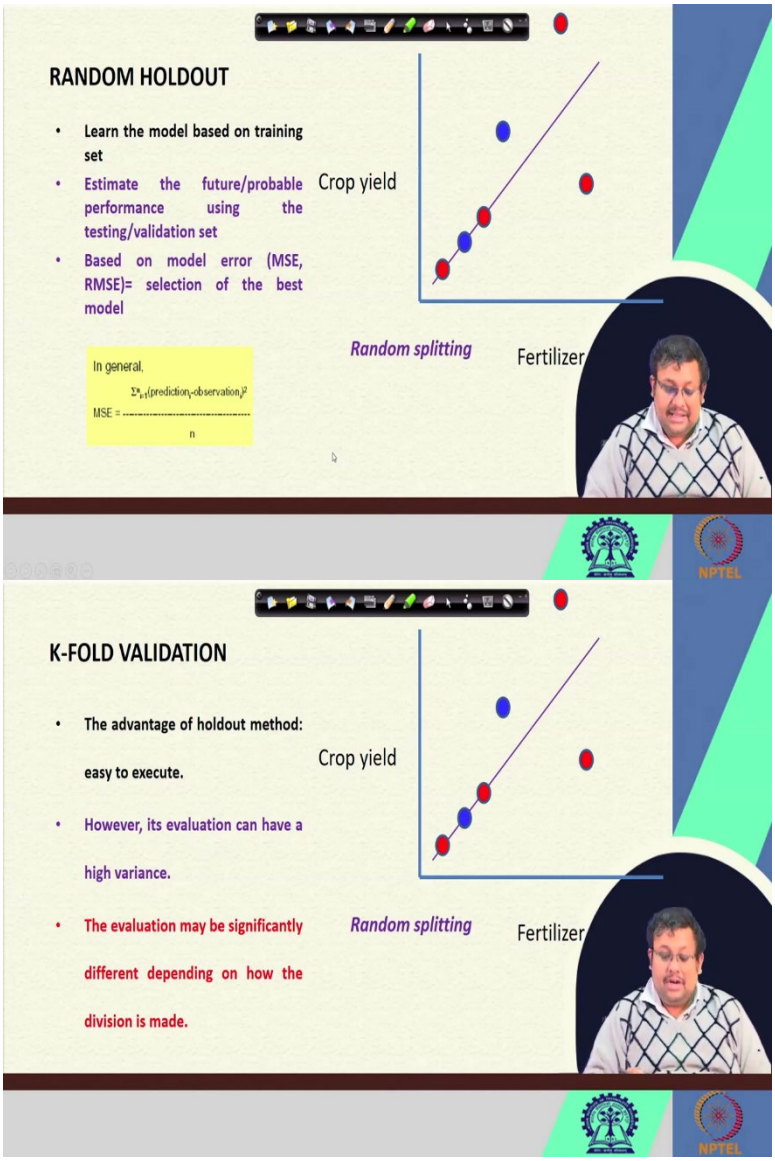
So, this simple, this holdout method, this random holdout because we are randomly selecting the validation set. So, this holdout method is simplest kind of cross validation. So, what happens, we randomly distribute the data into these calibration data and validation data, and then, we fit the model so this line is fitted based on our calibration data and then we measure the variability of our validation data from this model. So, this is how this model is validated.

In other words, first, we develop this calibration model based on this calibration data set and then we test the value the goodness of fit of this model based on the validation data set. And remember, one thing that it does not matter whether your calibration model is giving a very good r square value or not, if your validation is not producing good results, your calibration model is not useful or in other words that may show that may show the overfitting, which we are going to also discuss in details.

So, the data set is separated into two sets, calling training data set and testing set as we can see, the function approximator fits the function, this is the function using the training set, then the function approximator is asked to predict the output values for the data in the testing set or validation set, so another name of testing is a validation set.

And remember that because these validation set has never seen these output values before that gives the proper evaluation of the utility of the model. So, once we have the calibration model, then we get the predicted values for these validation samples, and then we compare the predicted values with this original actual values. So, if they are very close then we can say that this calibration model is producing good generalization capacity.

(Refer Slide Time: 10:07)



So, this is one method random holdout, and then learn the model based on training sets, so we know that the model has been learned due to by the training set and then you estimate the future of probable performance using the testing or validation set. And this how good these validation statistics are that can be determined by several statistics, mostly based on the error. Either MSE or RMSE sometimes r square RPD, RPIQ and so on so forth.

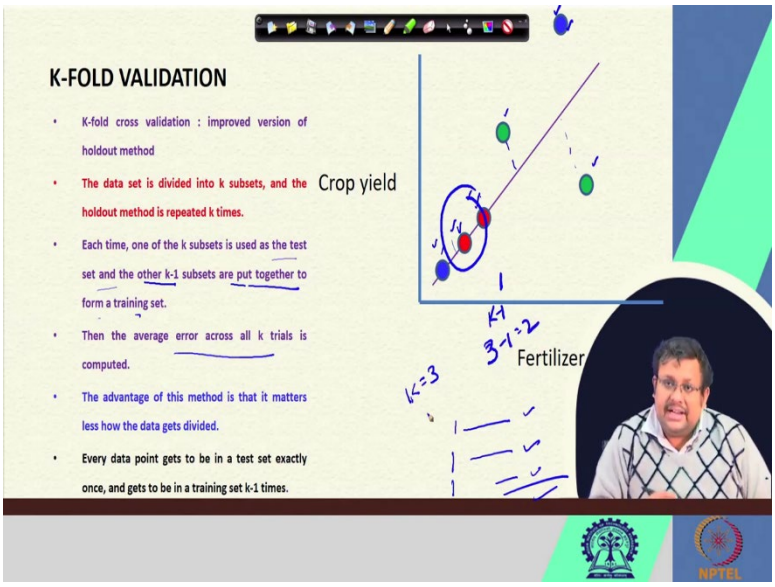
So, based on this is the formula of mean squared error. So, mean square is basically the summation of all the points, but their predicted values minus observed values then taking the squared divided by n, so this is the mean squared error. And when you take the root of this MSE, that gives us the root mean squared error. So, this is called the random holdout method.

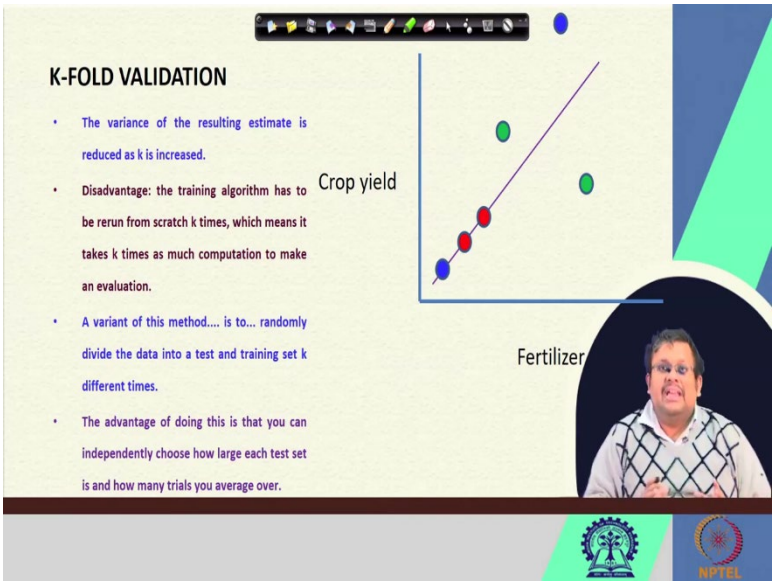
Now, the another method is known as k-fold validation. K-fold validation is basically the, but before we go to the k-fold validation, let us also discuss another thing that what is the advantage of this holdout method. The advantage of this holdout method is it is easy to execute. However, it is evaluation can have a high variance because you do not have any control over these random sub-setting and you do not know which sample will be there in the validation set.

So, these evaluation can have high variance. Because, if your validation samples are quite deviating from these random from this linear model, then you can have high variance. High variance means the validation samples are actually differing from this calibration model. So, the evaluation may be significantly different depending on how the divisions are made.

So, these random sub-setting nature also can influence the production of the validation statistics. You may get different types of values or different types of results based on your random splitting, sometimes you may get good results, sometimes you may not get good results.

(Refer Slide Time: 12:36)





So, another way of doing this measure validation is called k-fold validation. Now, k-fold validation or k-fold cross validation, it is an improved version of holdout method. So, what we do here, here, we basically divide the data into k-folds, k could be any number which is greater than 1. So, here you can see, we have divided the data where k is equal to 3. So, in that three subsets we have divided the data. So, this is one subset this is second subset, this is three subsets, say third subset. So, it is an improved version of holdout method and that data set is divided into k subsets and the holdout method is repeated k times it is very simple.

So, in the first go, what we will do, we will suppose, keep this blue subset and this red subset, and we will consider these four samples, these 1, 2, 3, 4 samples add as calibration model, and then we will validate the calibration model based on these two green samples. So, based on one subset, and we will repeat this thing for all these subsets. So, in the second go, we will consider these two red sample as validation samples, and we will build the model using the rest four calibration samples.

So, in this case, their model will be repeated three times, and from there, we will see, we will get an average of the model performance. So, each time one of the k-subset is used as the test set and the other k minus 1 subsets are put together to form a training set. And then the average error across all k trials will be computed.

So, in the first go will get an error, in the second go we will get an error, in the third go, we will get an error, and then we will average it. So, the advantage of this method is that it matters less how the data gets divided, because here we are sub-setting the data and then we are taking the average values, so here these variance problem in case of random holdout method, maybe somewhat addressed.

So, every data point because every data point gets to be in the test set exactly once and get to be in a training set k minus one-time. Of course, you can see here, here these suppose these red sample. So, these red sample can be in the training set can be test set exactly once, where we will consider this as a validation set and they will be in the training cases by k minus 1.

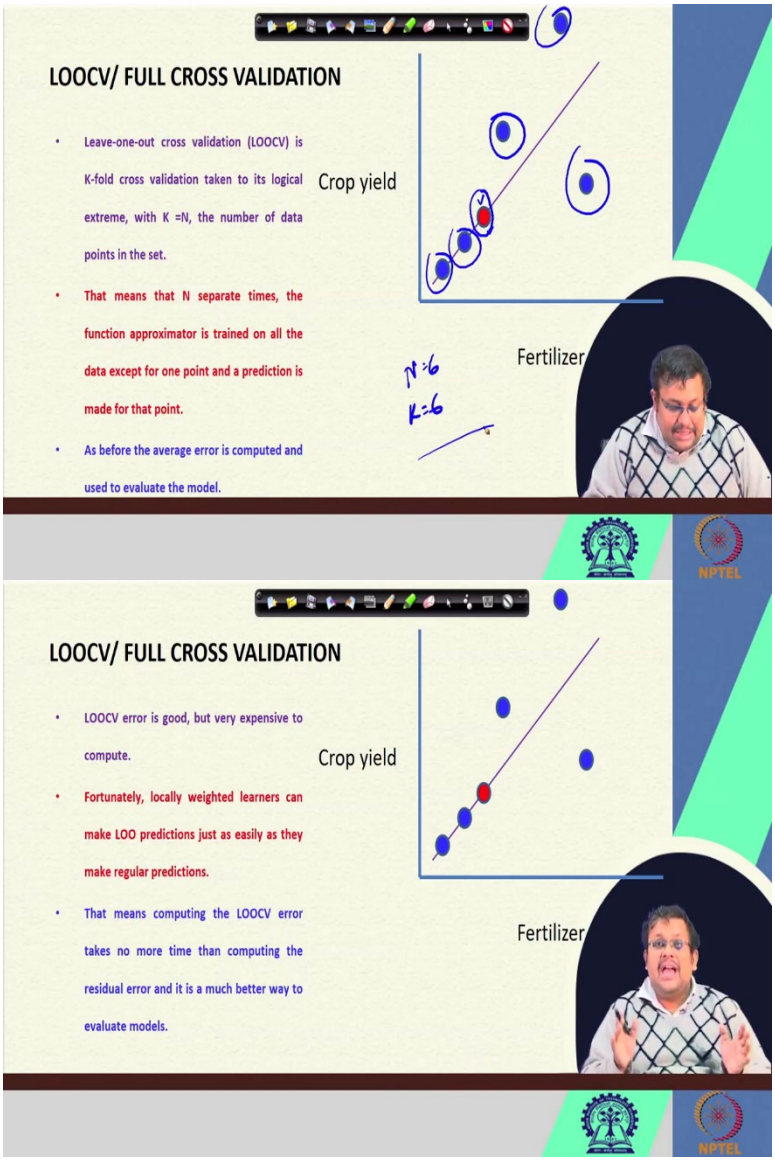
So, here the total k was 3 minus 1, 2. So, of course, two in the two iteration these two samples will be in the calibration set, and only one iteration it will be in the validation set. So, I think it is clear to you what is this k-fold cross validation.

Now, the K-fold cross validation, the variance of the resulting estimate is reduced as k is increased. So, of course, when we are increasing the k, the subsets or the value of k that means, we are getting more number of subsets when we are getting more number of subsets and we are averaging the results that means, the variance of the resulting estimate is getting down.

What is the disadvantage? Disadvantage is the training algorithm has to rerun from scratch to k times. So, if consider let us assume that k is 100. So, that means, this model has to run k times 100 times this model has to run, which means it takes k times as much as computing to make an evaluation. So, of course, a variant of this method is to randomly divide the data into a test and training set k different times.

So, instead of this we can randomly divide the data just like in case of random holdout method, we have randomly divided the data into calibration set and validation set and we can repeat this process k times. So, this is an variant of this k-fold validation. The advantage of these is that you can independently choose how large each test set is and how many trials you average over. So, you have that control. So, sometimes this variant of k-fold validation is also very useful.

(Refer Slide Time: 17:20)



So, the third method is leave-one-out cross validation or full cross validation also known as full cross validation. So, leave-one-out cross validation is k-fold cross validation taken it is logical extreme. So, remember when in case of k-fold cross validation the logical extreme will be k equal to n. So, if there are 100 of data points, if our value of k is 100 that means, in each of these subset there will be exactly one sample. So, there will be 100 subset and each subset there will be exactly one sample.

So, that means, say n separate time that is 100 times the function approximator is trained on all the data except for one point and a prediction is made at that point. Suppose, let us consider here in this case, you can see that, this is the total data set consists of 6 points and then here we are just selecting from the first go suppose, this is a validation samples. So, here n equal to 6 and here k equal to also 6.

So, what will happen, the model will run 6 times and in each of these time one sample will be kept outside and rest five will be used for calibration model and that calibration model will be validated by this set aside validation samples, and it will be repeated for each of these observation. So, in each go each of these observation will be set aside as the validation samples and the calibration equation will be built based on the rest five samples, and it will be repeated 6 times and then we will take the average.

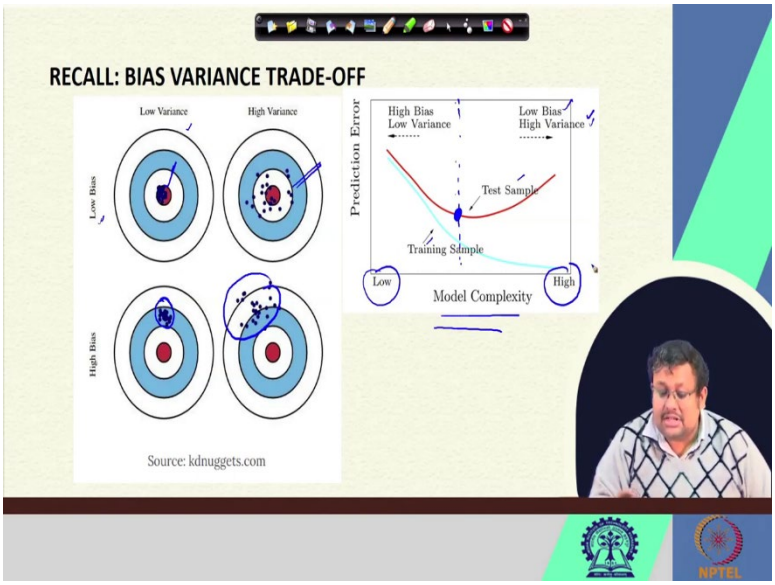
So, you can see it is a logical extreme of k-fold cross validation where k equal to n. So, that so as before the average rate is computed and used to evaluate the model. So, now I hope it is clear to you what is the difference between random holdout and then k-fold cross k-fold validation and leave-one-out cross validation.

Now, remember one important point that leave-one-out cross validation error is good, but very expensive to compute. Fortunately, locally weighted learners can make these leave-one-out predictions just as easily as they make regular prediction. So, that means the computation of the leave-one-out cross validation error takes no more time than computing the residual error and it is much better way to evaluate the model.

So, of course, due to the model advancements and software advancement, it is now possible to run these leave-one-out cross validation easily, so leave-one-out cross validation is the preferred method, when you have limited amount of data. Because suppose, if you have a limited amount of data, you divide it randomly into calibration set and validation set, then the number of samples in the calibration set and the validation set or in other words, the number of samples in the calibration set may not be adequate to produce a robust model performance.

So, for that, when the data is limited, we generally prefer to go with the leave-one-out cross validation, and when we have enough amount of data then we generally go for the random holdout method.

(Refer Slide Time: 21:09)



Now, please recall these before we go to the bias of the ridge regression. It is important to recall these bias variance trade-off. I just found these very good photographs based on these bulls eye and dart. And you can see that these gives you the clear picture of what do you mean by low variance low bias, high variance low bias, low variance high bias and high variance high bias.

So, in this case you can see all the aims are at the point. So, here you can see low variance as well as low bias. Similarly, in this case, here we are getting high variance that means, there is difference among the among these points, but they are having low bias, they are not deviating too much from the center of this bullseye. And then high bias and high bias low variance.

So, here you can see the model is biased, but they have very low variance because the data set data samples are not very scattered. And this is another condition where both high variance and high bias they are having they are showing much deviation from the center of this bullseye at the same time they are very much scattered also. So, this is the condition of high bias and high variance.

So, if you recall the bias variance trade-off of a model and now, you understand what is the difference between a training sample and test sample, when we try to fit the model and model becomes too complex, that means, when the model becomes too complex, that means, it shows low bias but high variance. So, in this case, this will be the overfitting problem. Low bias, but high variance. So, in this case, there will be low bias and high variance.

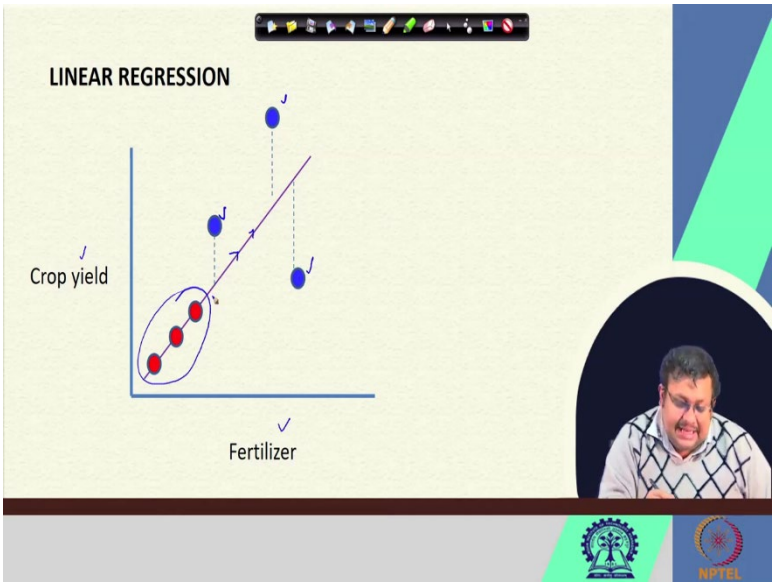
What will happen? The training model will perform very good, you can see that, but test model will show huge error because you can see the variance is large. So, the training testing model or validation set will produce higher variance. So, that shows that your training model is not accurate, because this will be the ideal situation right, this will be the ideal situation. But in this case, when the training model is too complex or learned too much based on this calibration data set, then they show low bias and high variance.

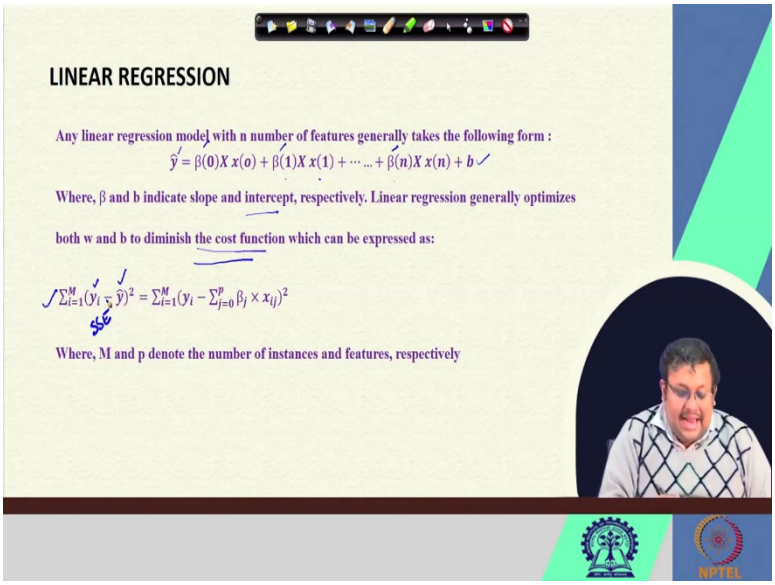
But, when in the other in hand on the other hand, you can see that. So, basically what happens in this case, the training set will show very less error, but test set will give you higher error after a certain point. So, this is called the a high variance, low bias condition or high model complexity condition.

Just opposite in the low model complexity condition there will be high bias, but there will be low variance. So, you can see here there will be high bias but low variance. So, in this case also we will get this kind of situation, but we have to understand that we have to find the sweet spot where we can say that the sweet spot will be somewhere here where these test set will produce the minimum error. So, we have to always find the sweet spot, so that we can train the model in a better way.

So, we should not go for too low complexity or too much high complexity also, we have to so this is called the trade-off. So, this trade-off you want to keep in mind while go for the linear regression.

(Refer Slide Time: 25:21)





Now, linear regression you know, let us consider one situation where there are 6 samples and this first three samples these red samples are considered as the linear regression or linear, sorry, calibration data set at this calibration. Based on this calibration data set, we have developed this calibration model linear regression. Suppose, here the target is crop yield and our independent variable is fertilizers, so we can see here we can develop this calibration model, but you can see that there is much variance from this calibration model these testing samples are validation samples.

So, that means, our model is overfitting. Our model is too trained or too adapted to this calibration data set that it does not generalize our validation data set. So, that means our data set is high, it shows the overfitting problem.

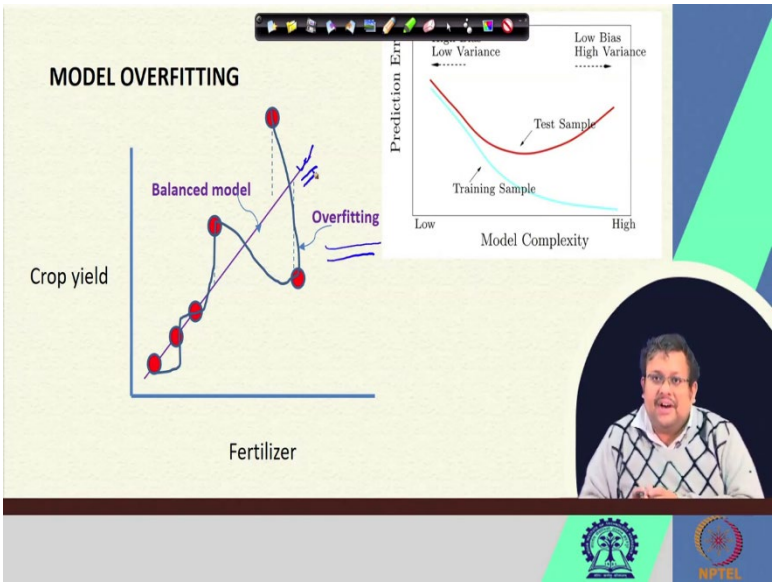
So, any linear regression model with n number of features generally takes the following form which we already know where our target is basically B0, X0, B1, X1, sorry, beta 0 x0, then beta 1 x1 beta n xn plus b where these beta 0, beta 1, beta n all these are basically the slope coefficient and this is the intercept, we know that, it is a multiple linear regression.

Now, where beta and b indicate the slope and intercept and linear regression. Why we call it ordinary least squares regression because in the linear regression the cost function which can be expressed by these, that means, this is that actual value and this is the predicted value. So, of course, the linear regression will always try to reduce this is some, this is basically error term actual minus predicted and then square means sum squared error.

We can tell it is a sum squared error. So, the cost function of any linear regression is to reduce this sum square error, and the sum square error can be generalized in this way. So,

here M and p denotes the number of instances and the features we know that. Now, so this is we already know in the linear regression.

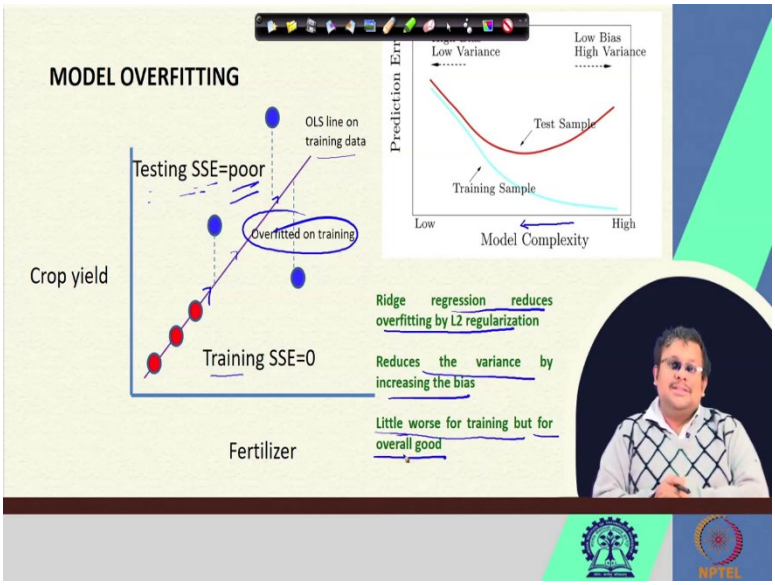
(Refer Slide Time: 28:00)



So, but what happens in case of, in case of these. Suppose, this is a condition where we have 1, 2, 3, 4, 5, 6 total samples and in the balanced model, we will get this type of straight line, but if our model is overfitted, then we will see that our model will look like this. That means, it is too much flexible or too heavily learned based on this calibration service.

So, we can say this is an overfitting, and this is a balanced model. But we have to find a, but in real life situation a balanced model may not be optimum for testing data set. So, we have to regularize that balance linear model and ridge regression is a form, it is a form of linear regression, we know which can regularize this cost function by adding some penalty.

(Refer Slide Time: 29:18)



So, if you consider this, so suppose these three are the training samples and based on these three training samples or calibration samples, we have developed this model, which is linear. This is the ordinary least squares line on training data. And you can see this line perfectly goes through these three points. So, the training sum squared error is 0. Of course, they are lying, the predicted values are, the actual values are lying over the predicted values we know that.

But if you see, these are the three testing samples or validation samples. The variation from these testing samples to this model is quite high. So, this testing sum square error is quite poor. So, that means, this model although it is linear, it is overfitted model, so it is overfitted on the training data set.

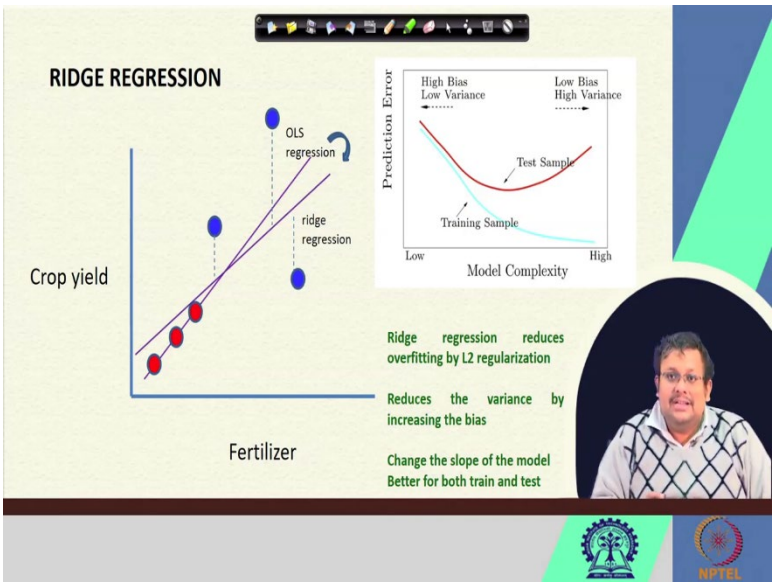
So, that means, in future if we want to predict any sample based on this model that will miserably fail. So, this is called overfitting. Now, what is the solution, the solution is ridge regression because this ridge regression reduces the overfitting by regularization, we call it L2 regularization, because it reduces the variance by increasing the bias. Remember, here the model when it is too overfitted that means, it has low bias high variance, but in the ridge regression what we try, we try to add some bias so that the model can be less complex, but at the same time we can reduce the variance.

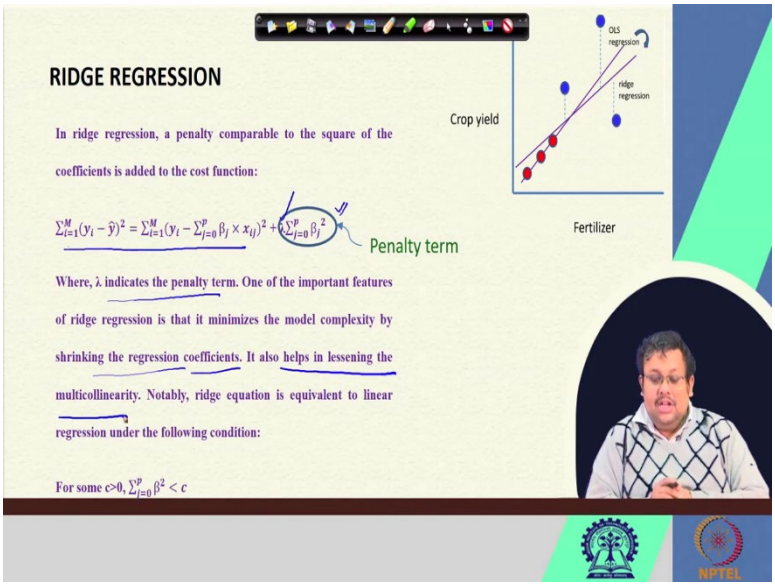
That means, reduce the variance means, we are ensuring that the model will perform better for the testing or validation samples. So, in this for this, we go for the ridge regression. So, we ensure that the model is little words for training, but for overall good. What is overall good? Overall good means, we may not get 90 percent accuracy in the calibration model we

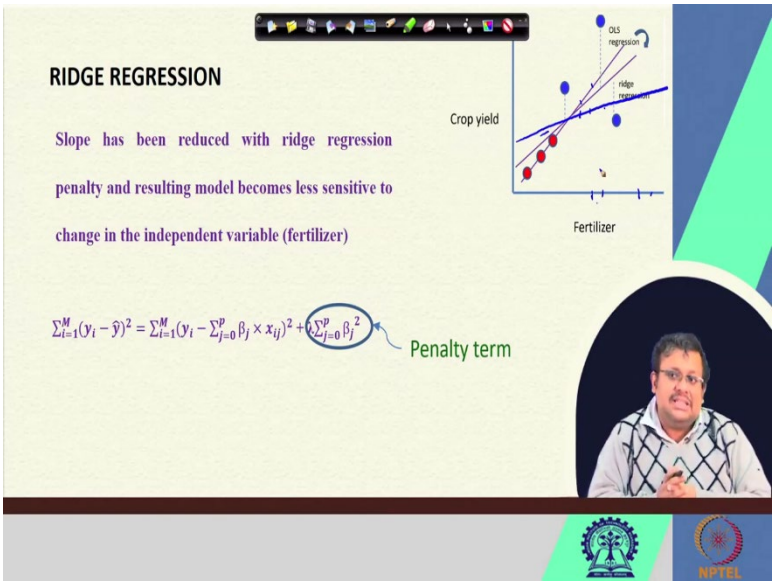
may get we may get now, up to 80 percent, but earlier we are getting r square values for validation samples suppose, 0.5 but we have now increased the r squared values for the validation sample suppose 7.0.

So, although we are getting little worse training results, but at the same time, we are getting overall good performance for as denoted by improvement of this testing data set. So, this is called the model overfitting.

(Refer Slide Time: 32:11)







So, how does it occur? So, here you can see basically in the ridge regression you know, here basically what happens, we can draw another line. So, it is the original OLS regression in the ridge integration they just rotate the line. They just rotate the line or change the slope. They change the slope of the model better for both train and test it.

So, you can see that when we change the slope of the model and we rotate this line, then the validation samples are showing less variance. Earlier it was showing higher variance, but if you consider these two lines, these two validation samples, this two validation samples, now by rotating this line, the variance from the validation samples are getting less.

So, this is what rigid regression does. It basically reduces the variance by putting an extra bias. Because when we are rotating these lines we are changing the optimum line to by putting, by giving some extra, we are rotating by some angle, so we are biasing the model, but at the same time we are reducing the variance. So, this is what is called the ridge regression.

Now, in ridge regression what we do, a penalty comparable to the square of the coefficient is added to the cost function. Now, we know for linear regression the cost function is up to this, but in case of ridge regression, we are adding a penalty term. So, this penalty term is specifically this lambda is the penalty term, so basically, we are adding this whole penalty term where this lambda indicates this penalty term.

One of the important feature of this ridge regression is that it minimizes the model complexity by shrinking the regression coefficient. It also helps in lessening the

multicollinearity. So, it by reducing the overfitting we can also address the multicollinearity issue which we have already discussed.

So, here you can see, the slope has been reduced with ridge regression. Now, the ordinary regression, ordinary least square regression where it is having higher slope, but now, while we rotated this line, we are having reduced slope with ridge regression penalty.

So, when we are adding more penalty we are getting now less slope and resulting model become less sensitive to change in the independent variable, that means, when we are getting more flattened, when we are adding these penalty term and progressively we are getting more flat curve you can see, for per unit increase in fertilizer we are getting far less increase in crop yield that we have got in case of ordinary least square regression.

In case of ordinary least squared regression for unit increase of fertilizer dose we have got higher increase in crop yield. But as we are doing, dealing with the ridge regression, for per unit increase of fertilizer, now we are getting less increase. So, that means, our resulting model become less sensitive to change in the independent variable of fertilizer. So, this is how we go for these ridge regression. Let us wrap-up our lecture here, and in the next lecture we will start from here, we will discuss more about this ridge regression, and then, we will go to LASSO regression and artificial neural network. Thank you.