

Course Name:Business Intelligence and Analytics
Professor Name:Prof. Saji.K.Mathew
Department Name:Department of Management Studies
Institute Name:Indian Institute of Technology Madras
Week:05
Lecture:20

OVERVIEW OF DATA MINING TECHNIQUES | Business Intelligence & Analytics

Now, we can now, we are now moving from the explanatory to predictive analytics, explanatory to predictive analytics. And we have already seen what is function discovery, what is model building and model parameter estimation. And whatever models build, when you actually use it for deployed, there will be some epsilon, some error. And because that is not captured in the model, so it is a model plus an error, which actually describes a phenomenon or an outcome called y . And statistical learning, as I already explained, is a part of machine learning, where statistical techniques are used for estimating the value of f . And we already understood residual sum of squares, residual sum of squares is the sum of squares of the error, when you fit a model to a set of data.

So, those points which are not falling in the model, they are residuals when you sum them up, and square and sum them up. Absolute sums may be misleading, so you square them. And they also have mathematical properties, when you actually do estimation, particularly through minimization. So, squared sum, squared sum of errors are used.

And that is about model estimation or model building. Now, in this slide, I am presenting a different scenario. And it may take a while for you to appreciate what these two different graphs represent. And let me make an effort to explain this to you. And these concepts are very important while making decisions about model selection.



OVERVIEW OF DATA MINING TECHNIQUES | BI&A | Prof. Saji Statistical learning

- ▶ More generally, suppose that we observe a quantitative response Y and p different predictors, X_1, X_2, \dots, X_p . We assume that there is some relationship between Y and $X = (X_1, X_2, \dots, X_p)$, which can be written in the very general form

$$Y = f(X) + \epsilon.$$

Statistical learning refers to a set of approaches for estimating f

Training: Residual Sum of Squares,

$$\text{RSS} = \sum_{i=1}^n (y_i - \hat{y}_i)^2.$$

This is, I would say, considerations in model selection. One aspect we already covered in model selection, that is, are you pursuing an explanatory project or are you pursuing a predictive project. In explanatory projects, the standardization of data is very important. So, that is one aspect we have seen in connection with model building. The other is related to selection of models.

So, let us closely look at the diagram on the left side. And you can see that again, this is simple linear regression, $y = f(x)$ and x on the y axis, x on the x axis, y on the y axis. And then, there is a scatter plot. So, you put all the points, x y points in the plane. And then you can see how they are, sort of distributed in the x y plane.

Now, as someone interested to do model building, you have different choices while you use, you select a model to fit this data or this pattern of data. You can see that this is not a straight line. But despite that, you can always put an approximate straight line. This is linear or this is your $y = \beta x + c$ kind of a model. This is the linear model.

But by looking at the linear model, you can easily see that there is error, there is

RSS. Many of these points are actually away from the line. And that is a concern for you. Or you can see that the R^2 , if you actually look at the coefficient of determination, the R^2 may be a 40 percent or a 50 percent. So, the model is not fully explaining the data.

Or you can say the total error, the total error or the RSS or it can be called bias, bias or the error or the RSS is very high for a linear model. The error is very high for a linear model. See other models which are attempted for the given data. There are non-linear models. You can see that the dark line which is, which is curvilinear or some non-linear mathematical function which is used to fit into this data.

When, if you actually calculate the RSS or the error or the bias of this second model, this is model 2, this is model 1 and model 2's error obviously is much less because it passes through most of the data points. So, therefore, the bias is much less. You also look at the green line. Somebody has attempted a much more sophisticated model which passes through most of the points, this green line. The three will have much lower bias in terms of the model building phase.

When we estimated a model, you have three models and the most non-linear model is the best in terms of lowest bias and the linear model is worst in terms of bias. It has very high bias or error, but that is just one aspect. But what happens? Suppose you change the data set. This is a sample collected from a phenomenon between, which contains y and x . Suppose you collect a different set of data and apply the same models, you can imagine that the distribution of the data in the xy plane would be different.

And therefore, the non-linear models obviously will have a different pattern or a different shape whereas the linear model may remain almost the same. The linear model will almost remain the same. The bias will continue to be the same. There is a high bias, but when you bring many other data sets from the same phenomenon and fit a linear model, the linear model may not have much variance, will not have much variance between models. The model will remain almost the same.

Whereas the non-linear models are going to change because it will try to pass through every data point that is there in the new sample. Even the outliers will influence the shape of the new model. So therefore, the variance among non-linear models would be much higher as compared to the variance among linear models. Whereas when it comes to bias, the bias of the linear model will be

very high whereas the non-linear models, the bias will be very low because every model will try to fit into every data point. So the training error or the error or the bias will be very low.

So this is one aspect of modeling where non-linear models have high variance and low bias and linear models have high bias and low variance. And I hope you understood my explanation. Now let us come to the next graph. The next graph explains flexibility as the x axis and mean square error as the y axis. And let me explain to you the axis.

Here flexibility actually means extent of non-linearity. A high flexible model like the curvilinear models here are very flexible. They are very flexible models. So the more the flexible, the more non-linear the model is. Here it tends towards linearity.

So one on the one side or one boundary you have the linear and on the other boundary you have the most non-linear models. And when you vary the non-linearity or when you vary the flexibility of the model, how does it influence the error? And error also, two types of error. The red line is the test MSE. This is the test MSE and the grey line is the training MSE. What does that mean? In order to explain that, I will go back to the previous. I will have a board and I will try to explain to you that.

Suppose your data set is this. This is the total size of the data is n . So you have $y_1, y_2, y_3, \dots, y_n$ and $x_1, x_2, x_3, \dots, x_n$. So in a pure explanatory project like what I explained in one of the previous slides, you will use the whole data or the whole sample for building the model because your purpose is to explain what is the relative importance of the different variables in a model. And it ends there. You may do significance test to see whether the model generalizes and so on and then you give your interpretation. Whereas if your interest is predictive modeling, then it is also important for you to test the model and see how well the model predicts for a data outside of the data that was used to build the model. For example, suppose in the simplistic example is, I partition the model into two.

Up till here, I set apart this data as training data. And the rest of the data I will call as test data. So this amount of data is what I am going to build the model. I build the model and my model is fit. I got my b and c estimated using this data set.

Now what do I do? And then when I do that, I can of course, calculate my residual sum of squares. I have my measures like the R^2 , adjusted R^2 etc. Also I

do significance test and all that to comment on mod's generalizability. So these are measures I have to comment on the goodness of the model or the goodness of fit.

I have done that. But in predictive models, RSS is not enough, because you also need to see how well the model predicts. In order to do that, what I do, I use the test data. So what I am doing here in testing is, I have a model already, which is $y = bx + c$, b and c are estimated, the values are known. And assume that that value of b is 3, value of c is 2.

So y equals $3x + 2$. Now what I do in the case of testing is, I bring the x data alone. I bring the x data alone. I input the test data, x test data. When I input each x value, the model computes the y value, because model knows the model parameters. So y , corresponding y is calculated from the value of x , value of b and value of c .

All the three are known. So they, therefore, you get a predicted output. t is a set of x data and y_t , y cap t is the predicted data from the input x data. Now, you have a , another error which you can compute here, because you have the actual y 's here. When model predicts, see for the point, so suppose this is x_{t_1} and this is y_{t_1} . For when I input y_{t_1} , model predicts y_{t_1} , that is predicted y_{t_1} .

NPTEL OVERVIEW OF DATA MINING TECHNIQUES | BI&A | Prof. Saji

The diagram illustrates the process of data partitioning and model testing. It features a table with two rows: the first row contains $y_1, y_2, y_3, \dots, y_n$ and the second row contains $x_1, x_2, x_3, \dots, x_n$. A vertical line separates the first three columns from the rest, with an arrow pointing left labeled "Training data" and an arrow pointing right labeled "Test data". To the left, a graph shows a line with a circled "RSS" label and handwritten notes $R^2, adj. R^2, K$. Below the graph, a box contains the equation $y = bx + c$.

So for any t_i which I inputs, there is a corresponding predicted y_{t_i} . And therefore, I am in a position now to calculate y_{t_i} - test data i , y_{t_i} cap. And I can submit for the entire range of this data. That is $i = 1$, say t , suppose the size of the data is t , for test data. Now, this particular error, sum of squares of the error is the prediction error or mean square error.

It is the mean square error or the prediction error. Prediction error is different from residual sum of squares. Residual sum of squares is the error of the fit, whereas prediction error is the error of prediction. And now MSE is can all this can also be called test error.

This RSS can also be called training error. Because model building is actually through training, you use training data, this data would be, this data is known as training data. And therefore, this is actually model building. So, the error during model building is the residual sum of squares or the training error and the error during prediction is the test error. Now, having got this fundamentals, now let us move on to understand the right side of the graph again. So, we discussed what is bias and what is variance and how this would differ between a linear model and a non-linear model.

Now, you see how the different types of models, this is about types of models based on their linearity, non-linearity and how the test error and training error. This is test error, this is training error. What you can observe here is that as you increase non-linearity or as you increase flexibility, as you make the models more non-linear, you can see that the error is going northeast or error is decreasing. The training error continuously decreases as you increase the model flexibility and that makes sense because you can see that as you make the model more non-linear, they pass through every data point and therefore, the error actually becomes much less. Whereas, if you look at a linear model, you know there is a lot of error because it does not, it is not flexible, it is not passing through different points.

So therefore, the training error is very high when, not very high, relatively high when the model is linear as in the case of the linear model. Here it is high, but when you increase its flexibility, the bias or the training error goes down. But you can actually see another phenomenon, that is when you increase its flexibility, what happens to the test error? When you build the model and test the model performance with respect to another set of data which was not used in model building, you can see the test error is going up. The test error is going up. When

you make the model more non-linear or make it pass through every point, it is good for an explanatory model.

But when you, if you have to use it for prediction, the prediction error may be higher because the variance, we have seen that already, the variance of the model for different data would be very high. You may have built the model using one data, we use another data to test it, you know the model perform very differently. Therefore, the model's prediction error actually goes up. That phenomenon, this is known as overfitting.

This is known as overfitting. The meaning of the term overfitting is that when you use highly non-linear models or highly flexible models, they overfit the training data. So that when you use another data set for prediction, the variance of the model is so high that it produces much higher test error. It is not able to predict for a different data set because it has, it has done overfitting into one data set. So this, you can of course find real life cases of overfitting where people get too much used to something and when they are, when their brain get trained to something, they cannot perform outside of that. So you know, I have several examples for that, but think about that overfitting is a problem.

And therefore, this particular graph explains why highly non-linear models are not desirable for predictive projects because of training error, high, potential high training error. Of course, you can test different models at different levels of flexibility and see which model performs best in terms of prediction error. But this is a general tendency, as model non-linearity increases, prediction error increases. Now, we are, we have moved from explanatory to prediction. So we are very familiar with \hat{y} is a function of $\hat{f}(x)$.



Prediction

$$\hat{Y} = \hat{f}(X)$$

- ▶ **Three sources of error in predicted Y:**
 - ▶ Reducible error due to inaccurate estimation of f
 - ▶ Irreducible error due to randomness
 - ▶ Test data variation


Reducible error can be reduced by better learning techniques

So \hat{f} is an estimated function and this will not be 100 percent, there will be modeling errors and therefore your predicted \hat{y} would not be exactly the actual y . And what are the sources of prediction errors? What are the sources of prediction errors? There are three sources of prediction error. We have now understood what is, how do you calculate prediction error of course, but how are prediction errors caused? There are three sources as you can see, one is called the reducible error, that is due to inaccurate estimation of the function f . What does that mean? We already discussed model selection, you can select a linear model versus non-linear model. Suppose you chose a very high non-linear model, instead of comparing different models and choosing the best model you did not do, then your model is flawed, you should actually go for a different model.

You selected a neural network model and you did not try any other model and probably you should have tried different models for classification, say for example, and compare their performance which you did not do. You did not ensure data quality, you did not do data preparation well, you did not do standardization of data or something. There is inconsistency in the data or whatever. All those are reducible errors. Whatever the researcher or the analyst can act on, that is known as reducible error, right from choice of models to quality of data.

But there is also irreducible error. Irreducible error is due to randomness, randomness in the data. As you know randomness has no pattern and therefore, you cannot model it and therefore, that will lead to an error which is a random error and that is not in the control of the analyst or the researcher or you, as the one who does the project. So therefore, this is not something you can control, this you can control. And there is a third source of error known as test data variation. What does that mean? The testing process itself can introduce error or getting the actual error calculated is a challenge and that is because of test data variation.

Let me give the simple example. I explained to you, you partition the data into training data and test data. Now, this is one way, how do I do the partitioning? I can simply say my 80 percent is training data, my 20 percent is test data. Somebody else would say my 70 percent is training data, my 30 percent is test data. Suppose you are just doing it without any order or principle, then every model will be different. Suppose instead of selecting from the old data to the new data for training, suppose you traverse in this direction, you take last to the 80 percent, your model will be different and therefore, the reported error will be different.



OVERVIEW OF DATA MINING TECHNIQUES | BI&A | Prof. Saj

Cross validation


- ▶ Training error vs. testing error (prediction error)
- ▶ Mean Squared Error (MSE), a measure of testing error:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}(x_i))^2$$

- ▶ Three kinds of cross validation:
 - ▶ Test set approach
 - ▶ Leave-one-out cross-validation (LOOCV)
 - ▶ K-fold cross validation

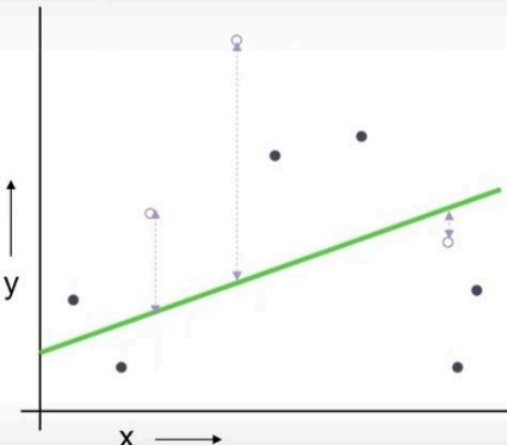
Suppose you randomly pick 80 percent of the data, you have another possibility, then again the model will be different and the test error will be different. So, test data itself is a source of error variation, reported error variation and therefore, how this problem can be addressed and that is known as cross validation. Cross validation is what we discussed. You build a model and then you test the model using the same data set or the data set that you collected. And for depending on the nature of model, so I am explaining it in my session, this session, the linear regression is the example because it is easy to understand and imaginable.

But soon we will discuss classification where the approaches that we suggested including MSE will not be applicable. So, there are different measures for model performance. We will see that separately. But as a general principle, what are the different kinds of cross validation to address test data variation? There is the first and the most rudimentary is the test data approach and then there is leave one out cross validation and then there is k fold cross validation.



OVERVIEW OF DATA MINING TECHNIQUES | BI&A | Prof. Saji

The test set method



(Linear regression example)
Mean Squared Error = 2.4


1. Randomly choose 30% of the data to be in a test set
2. The remainder is a training set
3. Perform your regression on the training set
4. Estimate your future performance with the test set

27:10 / 34:42

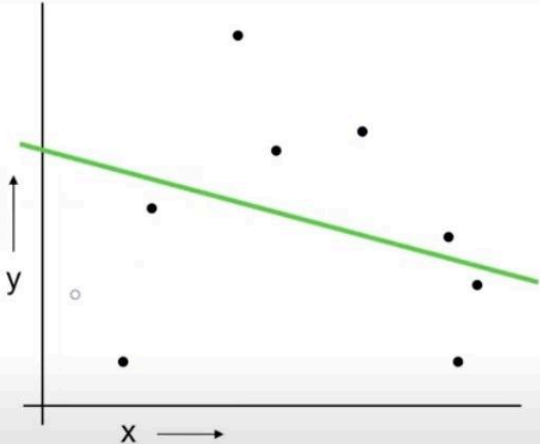
INTELLIGENCE & ANALYTICS

I will explain each of this in the next few slides. The test set method is explained here and I explained it already. You partition the data, you partition the data into test data and training and test data. And how do you do that? You can actually pick, say 80 percent of the data randomly as training data and remaining data as test data and then test the model and report the MSE or the performance measure. Now, another principle in the choice of the training data set is that if you have y and x , you try to run a plot like this. Now, we will discuss this when we will model a time series data using ANN.

So, one principle is that in the test data, most of the major variations in the data should be included. So, you have to ensure that this particular spike which is in the data should be included in the training data. So, the 30 percent for test data and 70 percent for training data is just a thumb rule. You also have to ensure that the major variations are covered. So, instead of 70 percent sometimes you may choose all the 80 percent for training because you do not want to miss a major variation. That has to be kept in mind.

 OVERVIEW OF DATA MINING TECHNIQUES | BI&A | Prof. Saji

LOOCV (Leave-one-out Cross validation)



For $k=1$ to R

1. Let (x_k, y_k) be the k^{th} record
2. Temporarily remove (x_k, y_k) from the dataset
3. Train on the remaining $R-1$ datapoints
4. Note your error (x_k, y_k)

When you've done all points, report the mean error.

28:50 / 34:42

BUSINESS INTELLIGENCE & ANALYTICS

Now, the second method is the leave one out cross validation, leave one out cross validation which is explained here. So, here what you do is you, this is your whole data x y data 1 to n . What you do here is, you do not partition the data into training set and testing set. You first choose one data point, maybe the first data point and this is the first point x_1 y_1 , leave this out. Use the rest of the data for training or build a model using $n-1$ data, the first point is left out and then you will get a MSE.

How do you get the MSE? You test the model using just one data point which is the first data point you left out, you get MSE 1. Now, what you do, put this first data back, put this back and you go and leave out the second data point and build the model using the first third to n . Again you have n minus 1 data points, build the model, test the model using this data, the left out data that is your MSE 2. So, in this way you will actually build n models and test it n times and you get n number of errors, n number of MSEs. You find the average MSE, very rigorous method of testing, building and testing models.

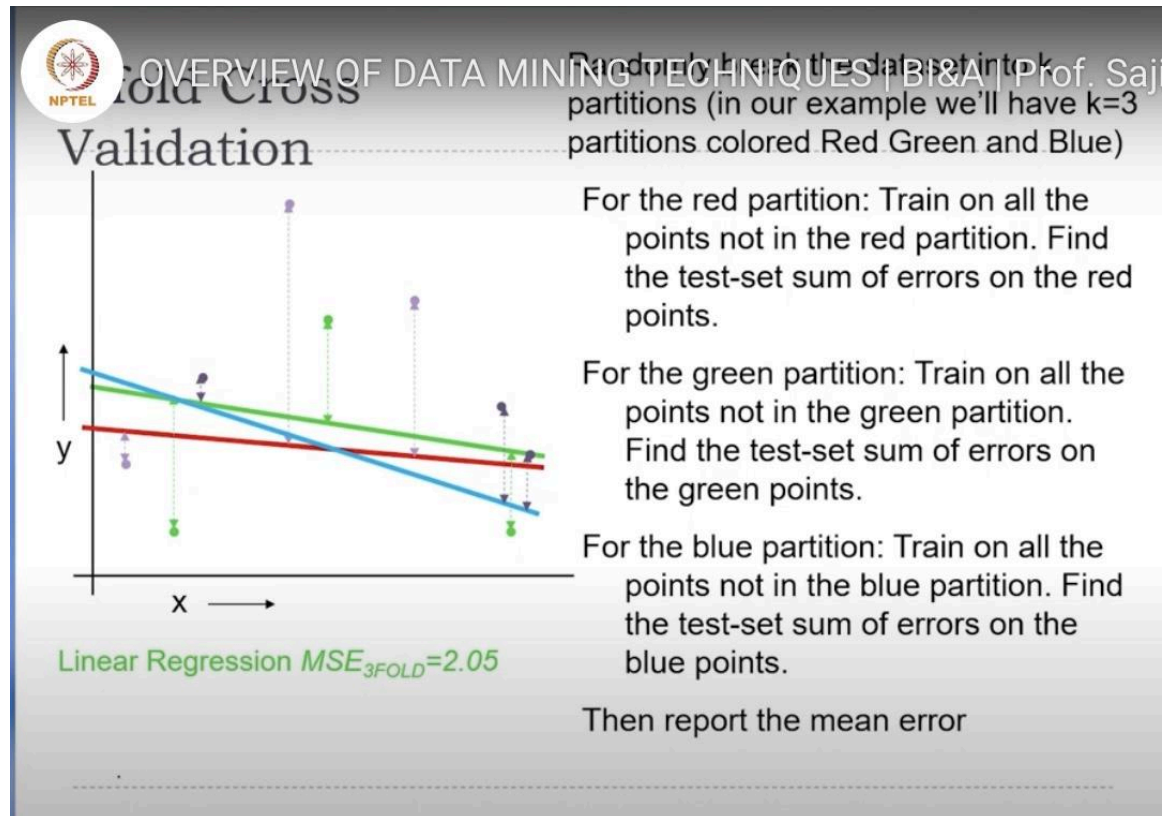
You have not left out any data point for building the model, you have not left out any data point for testing the model. It is very exhaustive and it looks like the ideal approach so that the final report of your performance of the model would be more accurate than, you know randomly selecting training data and testing data. But there is a trade-off here. The trade-off obviously is the computational cost, high cost of computation. You can see you need to build n models and suppose your data runs into millions.

So, the cost of computation is very high, but the prediction error reported is more reliable. And since there is a trade-off always you find, try to find a midpoint or a meeting point and that is actually known as K fold cross validation. K fold cross validation is easy to understand. Again, your number of data points is n . Now, instead of picking the first data point and leaving it out, leave one out cross validation, you do this fold wise, not singular data wise.

So, for example, in the graph that is shown, you divide the data into three folds. Here K equals 3. Now, what do you do? You build the model, you leave the first fold out, build the model using these two.

This is your training, this is your test. Then you get an MSE 1. Then you leave this out, build the model using this and this. You get MSE 2. Then you leave the last one out, build the model using these two, test it using the last fold that is MSE 3 divided by 3, K equals 3. This is known as K fold cross validation. Here

again, you are actually, you can obviously see that this is a trade-off between the two and you are trying to reach some optimum point where the computational cost is not very high.



At the same time, your test data variation is also not very high. So, K can be 3, K can be 4, K can be 10 and K can be n . So, when you have number of folds as much as the number of data points you have reached the loose CV, leave one out cross validation. And as the K increases, the computational cost increases, but the reported prediction accuracy is high. As K becomes lower, the computational cost is lower, but the prediction accuracy can be also lower.

So, that is the trade-off here. So, three major types of cross validation techniques, the partitioning approach or the test dataset approach. Number two, the leave one out cross validation and number three, the K fold cross validation. And that is all for today. Thank you for listening and see you in the next session. Thank you.