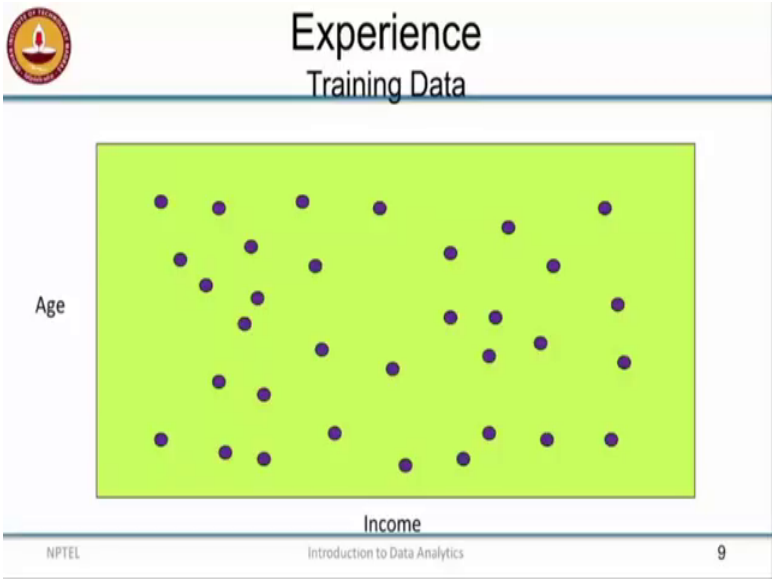
Introduction to Data Analytics
Prof. Nandan Sudarsanam and
Prof. B. Ravindran
Department of Management Studies and
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module – 04
Lecture – 17
Supervised Learning

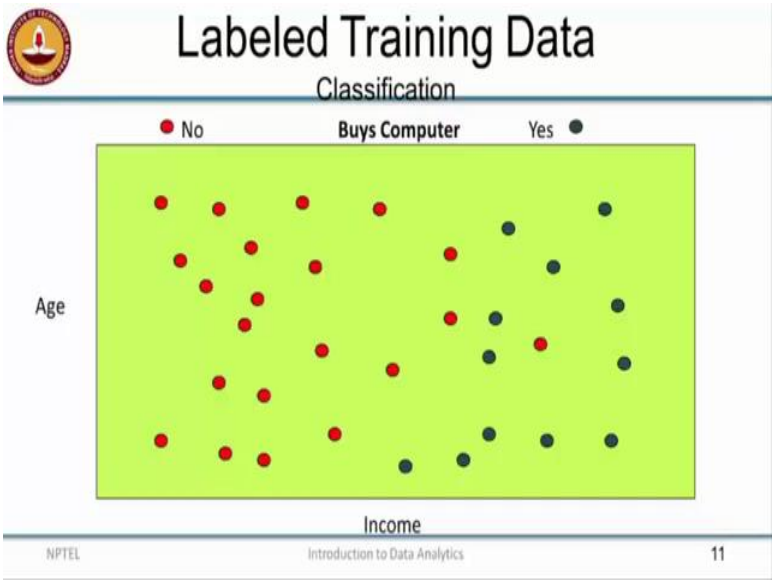
Hi and welcome to this module, where will be I am introducing you to various machine learning tasks. So, we already saw in the previous module and that, machine learning is essentially improving the performance of artificial agent with experience.

(Refer Slide Time: 00:34)



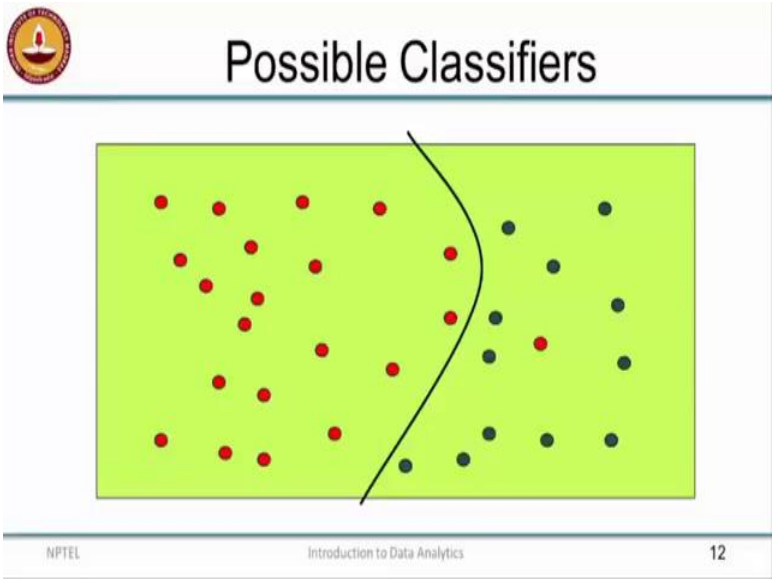
So, today we are going to look at the first supervised learning, where the experience we are going to call the experience as training data here. So, in this case I am showing you data points that this distributed in a 2D plane, so it could be let us say age and income that describe customers that come to a particular store. So, in the case of supervised learning, this training data is going to carry labels.

(Refer Slide Time: 00:57)



So, it could be that whether the customer is going to buy a computer or not going to buy a computer. So, the customers marked in green are going to buy a computer and the customers marked in red are not going to buy a computer. So, this is the kind of experience that is going to come to you and in the case of classification task, which is this, so we are going to call it label training data, where the labels are drawn from small discrete set, in this case it is just yes or no.

(Refer Slide Time: 01:36)



So, what is your goal here? So, your goal in this problem is to figure out, how are the yes's and no's distributed in this two dimensional play. So, the simplest model for this is going to be to draw a straight line. So, what does this straight line mean here? So, it's essentially saying that people, who have an income below a certain level are not going to buy the computer; people, who have income above a certain level are going to buy the computer.

So, if you think about it, we are really coordinate more or less correct, there are a few points like that and that, which are incorrectly classified and here is another one, who is actually going to buy a computer, but we are classifier is going to say, it is not going to buy a computer, because it is on the wrong side of the line. So, you can do better, so that is a slightly better classifier. So, the two, the red points that we had incorrectly classified previously that one and that one are now correctly classified, but we still are making a few errors.

So, what has happened from the previous classifier to this one is that, you have made it slightly more complex. So, if you think about it, the previously classifier was essentially say x equal to some constant, so it is just saying income is equal to some number, if it is less than that, it is essentially no, if it is greater than that it is yes. So, now, what has happened is we have added a slope to the line, it is no longer; just based on the income, but the age also has to play a role here.

So, there are more parameters that are needed for describing this classifier, earlier we just needed to have one parameter, now we need at least two parameter to describe this classifier. So, can we do better? Obviously, so we can do better we can draw a parabola like this, so in this case you need more parameters. So, you have $ax^2 + bx$, so you are now going to get additional parameters that you need to describe this classifier.

So, are you doing good here or should can we do better, it looks like we could do better because we still have an error. But, it starts looking a little weird, so we are essentially now, this is going to acquire a lot of parameters to describe what this classifier does, you know that little wiggle there that goes out and gets that additional red that you missed and say now, becomes increasing complicated.

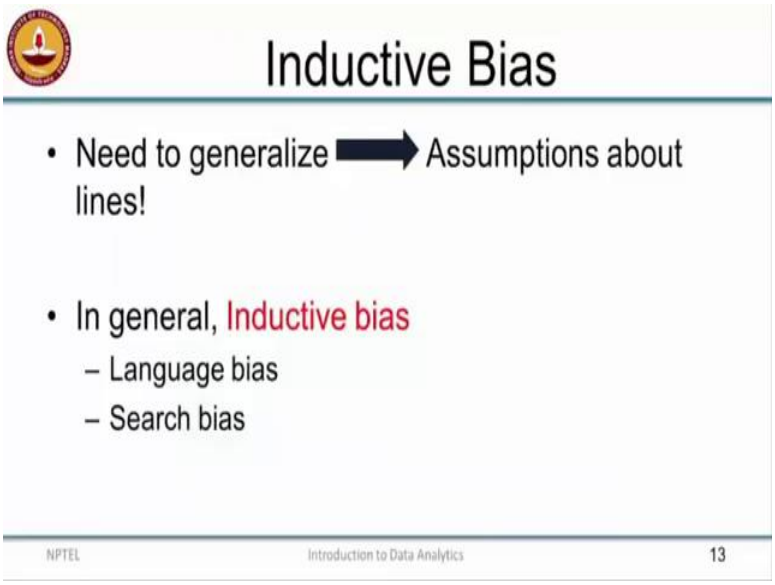
So, what really was happening is probably that data point is a noisy point, it could be noisy due to variety of factors, it could be that person came to the shop to buy a

computer, but for some reason he received a call and then, had to leave immediately without buying one or may be the data has been erroneously recorded you know person actually bought a computer, but it has not been recorded.

So, in there are several situations, where which says noise could enter into the system. And we have to be very careful that we do not end up modeling such noise in the data, which will lead us to make really wrong predictions in the feature data. So, probably the best solution to this problem is this parabolic curve, because it gives us a good balance between the complexity of the classifier verses the accuracy that we have on the training data.

So, remember that this is all evaluated on the training data and what you really are looking for it is, how well is this classifier going to work on the test data on the deployment. So, I am going to train this classifier and then, I am going to use this to predict whether the new customers who come to my store are going to by a computer or not. So, when I am actually operating it with this new data I want this classifier to do well and therefore, over fitting the classifier to the training data might be a bad idea.

(Refer Slide Time: 05:47)



So, one thing that you should remember here is that my goal is not just to do well on the training data if I am only interested in doing well on the training data, what is the best way to do it yes I just have to remember all the data points it was given to me. So, I can never make a mistake ever again and I can be perfectly accurate in making predictions on

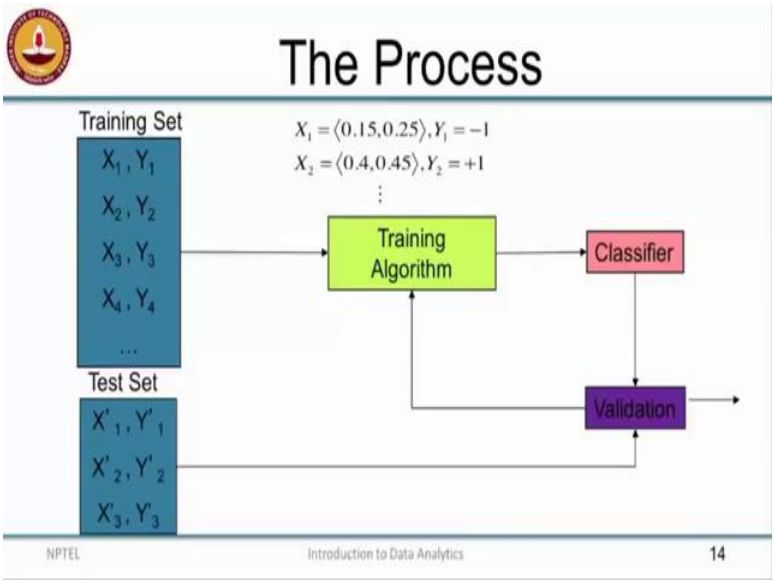
my training data, what I am interested in is to be able to perform well on data that I have not seen before and therefore, I need to be able to generalize to unseen data.

And the only way I can generalize to unseen data is by making assumptions about the model I have to make some kind of assumptions about, what kind of lines in this case, what kind of lines should be separating the buyers from the non-buyers. So, whenever I have this kind of need to generalize it translates to assumptions on these lines. So, such assumptions on the models that you are generating are called inductive bias.

So, the whole paradigm of learning that we are talking about is called inductive learning and the assumptions that allow us to get this kind of generalization are known as inductive bias and there are two forms of inductive bias generally. So, one is called the language bias, which essentially is the restriction that we had on the kind of lines if you say I am going to only consider straight lines that are parallel to the x or y axis that is one kind of a language bias or if you say that am going to consider only parabolas that is another kind of language bias and the other kind of bias is search bias.

So, given that you have picked a language in which, you are going to represent your classifier, how do you search among the possible classifiers in that language in order to find the right one. And that again influences what kind of classifier that you are going to find, because you typically will end up with the first classifier that you find that has an acceptable performance. And therefore, that will be influenced by your search bias and we will elaborate on this as we go along and this just to give you a feeling of what is involved in trying to do this kind of learning.

(Refer Slide Time: 08:11)



So, I like to elaborate a little bit more on the whole process of this supervised learning and this is kind of common to the other algorithms you look at as well, but am just going to talk about supervised learning little detail in this module. So, you start off with training data that is given to you. So, training data consists of vector x and an output y, so if you think about it. So, in this example that we just saw on the previous slide, so x would be looking like a the tuple of income and age.

So, the first data point could be that person has a 30000 rupees per month income and he is 25 years old and he did not buy a computer and the second person had a 80000 income he is 45 years old and he buys a computer. And, so he is essentially going to have a series of data points like these that are given to you and you have to use your now, your training algorithm in order to learn a classifier. But, if you stop and think about it, so for us to be able to implement this in a numeric fashion, so you are going to have to encode your data.

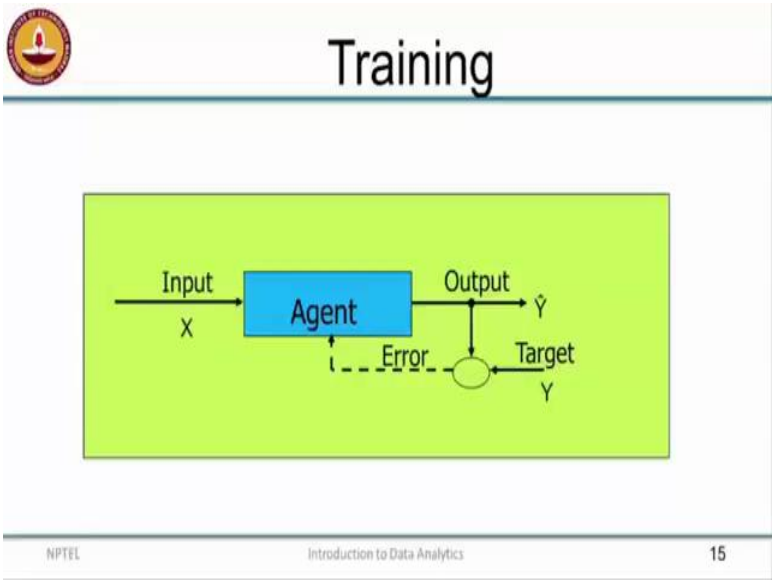
So, you probably take your income levels and then, do some kind of a normalization, so I have normalized it between 2 lakhs and 0 income. So, and that gives me like the first person has a income of 0.15 the second person has a income of 0.4 and so on, so forth. And likewise I have normalized the age between 0 and 100 and I get numeric value for the age. And the labels are encoded again, so not buying a computer is now represented as -1 and buying a computer is represented as +1.

So, whether you use a -1 or a +1 or whether you use 0 or 1 of whether you can use y and n depends on the kind of algorithm that you are working with and as we go long as we different machine learning algorithms you will find you will learn about the importance of the selection. So, once a have this encoded data, so the training algorithm is going to work on it and it is going to produce the classifier, but I need to know how good the classifier is and I need to know if I can stop.

So, if I go back on the training data and then, ask how good the classifier is typically we might end up over fitting the data like I was showing you in the couple of slides ago. So, what we do is set aside a testing set, so which is not something, which was used to build the classifier right and then, we validated the classifier on this testing set. And then, if the validated if the validation tells us the classifier is good enough, then we can go ahead and output it or if the validation process tell us known the classifier is not yet good not yet good enough.

And then, I can go back and modify the parameters of my training algorithm or iterate over the data again until I converge to something that is acceptable. There are many different ways from this validation can be done and again you look at some of this later.

(Refer Slide Time: 11:22)

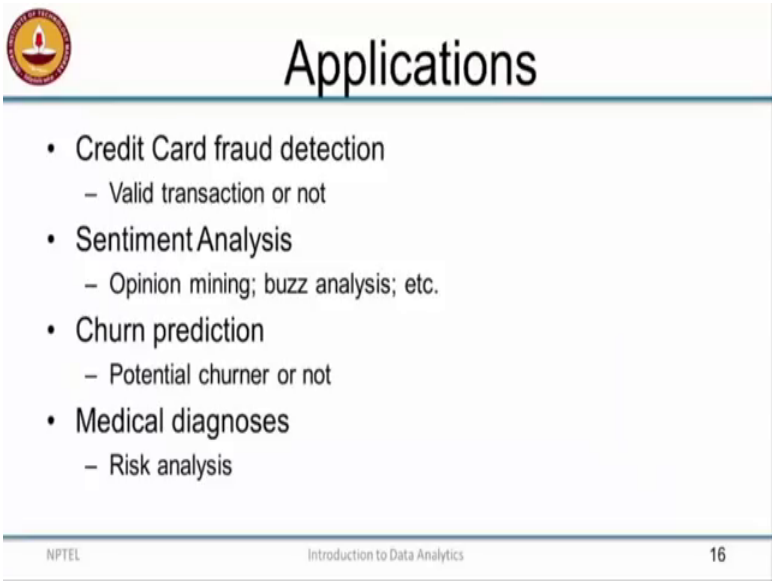


So, what exactly goes on in the training module, so the input, which is denote by x comes to the learning agent the learning agents uses the current setting of the parameters and it the outputs a value, which is ŷ, which is its guess of what the output should be for

this input x. And if you remember in the training data we already have a target y, which is the actual output that you expect from the agent.

So, I am going to compare the target y with the guess the agent output, which is ŷ that allows me to form an error, which is the error in the prediction and it gets fed back in to the agent and then, the agent uses the error in order to modify its parameters. So, this is something, which you have already seen to some extent in the regression setting, but we will look at a more of a learning approach regression in the next few classes, but you can also see that this is exactly, what the classifier needs to do.

(Refer Slide Time: 12:33)



So, there are many, many applications for classifications, so that a credit card fraud detection, which is a very I know that a few years back this was talked as a marquee application. So, when a person swipes a card you can say whether that is a valid transaction or not. And the other application which there a lot of buzzer around it nowadays a lot of startups and many companies are focusing on it is sentiment analysis or some time called opinion mining or buzz analysis etcetera this is looking at social media data whether the tweets or whether they are blog posts, so on, so forth.

And then, trying to figure out whether they are saying something positive about you or something negative about you, so by negative about by you I mean whatever is of interest to you it could be a movie, it could be a new album, it could be a product for those release. So, instead of marketers going door to door asking people what they feel

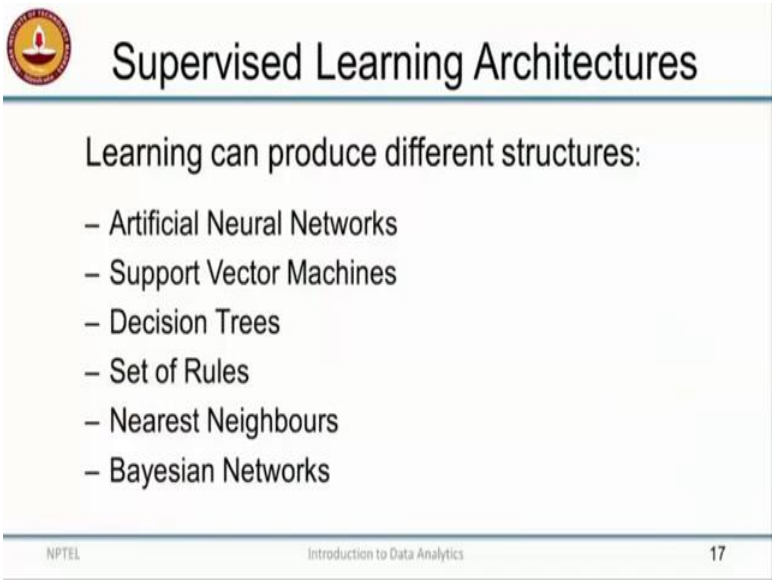
about the products we are now able to analyze the post that people put in the public forum and automatically figure out whether the opinion is positive or negative. Another application which has lot of commercial impact is one churn prediction.

So, a churner is somebody is your user from your service who is likely to leave it is like am going to switch from Vodafone to Airtel. So, all I am going to switch from a windows machine to a Mac. So, these are these people are churners you know they are habitual users of a of a particular service who are going to leave it and move to another service. So, many companies are very interested in identifying such churners and then, taking measures to retain them.

Because, attracting newer customers to your services is little harder than keeping the customers that you already have and people are willing to spend money to do that, so that is another interesting application. And of course, increasingly medical diagnosis is proving to be a very fertile ground for classification algorithms though it is the medical communities really not in a position to completely accept, what machine learning people tell them.

But, the machinery people any way keep working on lot of medical domains and there are many competitions that are run that ask people to build classifiers that can predict whether a patient is sick or whether a particular blotch that they see on a x-ray or a scan whether that is a cancerous thing or whether it is benign. So, all this kind of diagnosis questions are asked of machine learning algorithms. Then, we fair amount of success, but then still quite a bit of way to go and another place, where machine learning is used in medical diagnosis is in risk analysis.

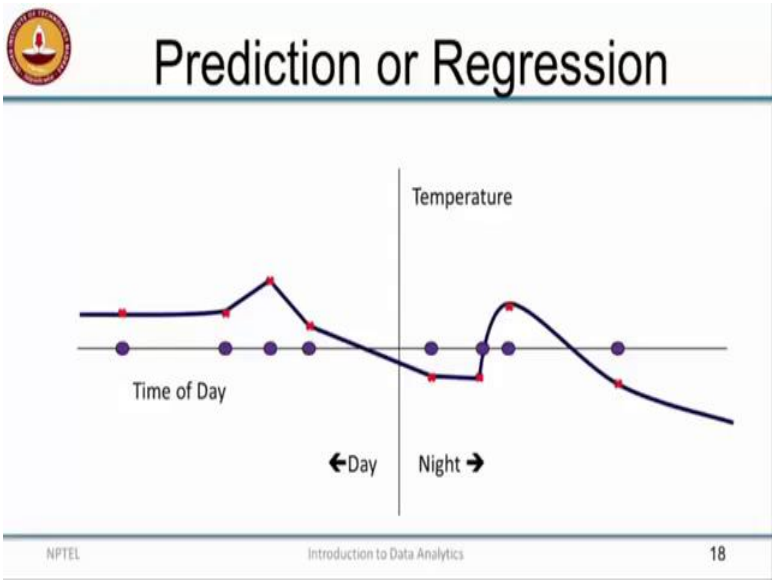
(Refer Slide Time: 15:31)



So, I will elaborate on a little bit more when I talk about regression and there are in supervised learning. In fact, the classification is one of the most widely studied machine learning paradigm. So, that many, many approaches for classification and some of them are very famous a few or few might recognize the names here they can produce learning supervised learning can produce different architectures.

So, it could be artificial neutral networks which we will talk about later support vector machines, decision trees or just sets of rules you know and other popular methods such a nearest neighbor methods, where you remember all your training data and then when a new data point comes in then, you try to make a prediction based on which, of the training data looks like the new data point right and also problestic methods are typically based on Bayesian approach to learning.

(Refer Slide Time: 16:34)

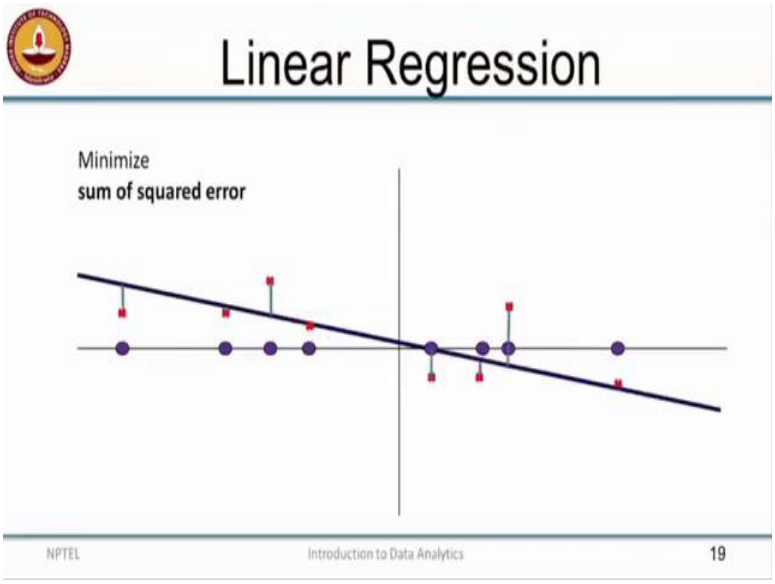


So, we will be covering at least the first three in this in this course in detail the other supervise learning problem that we are going to talk about is regression or prediction. So, here the data is going to have continuous outputs, so the input in this case this is simple example I have is temperature. So, the x axis is the time of day at which, the which temperature is measured and the y axis is the temperature, which is the output.

So, we could see that typically day time temperatures are higher and night time temperatures are lower and now I can try and fit a curve to this, so ideally what you would like to fit well all of us know about linear regression. So, what we are going to do first is try to fit a straight line, but then you can try to be little cleverer and you can try to fit a more complicated curve like that that sounds its looks a little reasonable or it could over fit the data like we talked about earlier.

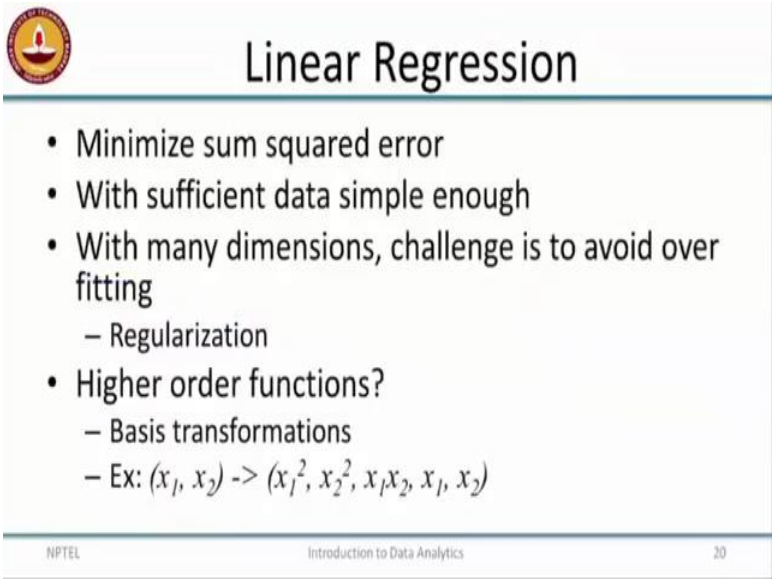
And then, try to make the curve really complex and then, try to follow all the micro variations in the temperature. So; obviously, in the night time you see a data points that is temperature is almost as high as the day time that is noise, but if you try to over fit it you are going to get incorrect results.

(Refer Slide Time: 17:48)



So, we already looked at the regression little bit of detail, so that we know that we are essentially trying to minimize the sum of square errors when you are trying to make a prediction. So, how do I fit this line I just figure out the difference between the line the prediction made by the line that I fit and the actual data point that was recorded take the square of the error and try to minimize that, so that is the typical approach that we use for fitting lines.

(Refer Slide Time: 18:17)



So, as we have already seen with a sufficient data doing linear regression is simple

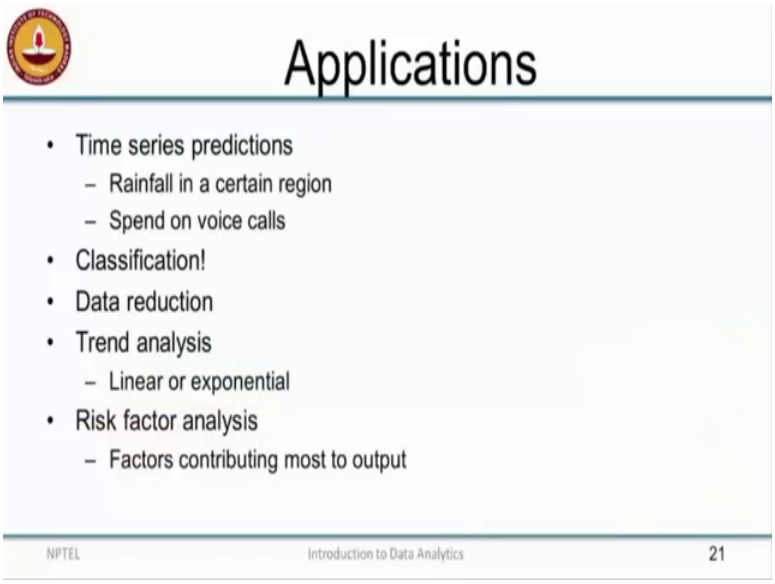
enough just a set of matrix operation. But, then if we have many, many dimensions in the last slide our data had only one dimension is essentially, what was the time of day that was my x vector. But, then if you have many, many, many dimensions and then, you really need a lot of data points in order to avoid over fitting, because if I have like a 1000 dimension data and I have only 100 data points it is very easy to over fit to those 100 data points, because I have a 1000 parameters.

So, how do I avoid that kind of over fitting is by adapting something called regularization. So, we will ensure that of all the models that can fit the data to a certain extent we will try to try fix find something that is simple enough, so that we do not over fit the data. So, here we are talking about linear regression, but suppose I want to do a higher order regression. So, if you remember in the previous slide, so we looked at an example where, so again it its looked like a quadratic curve was slightly better fit to the data than a linear fit.

So, how do we handle higher order functions one simple way of doing this is to look at, what are known as basis transformations. So, where you take the input, so input could be say x_1 and x_2 there are two variables at that describe the data and then, I translate that in to a much larger description by adding second order terms. So, I take x_1 x_2 I create x_1^2 x_2^2 the product x_1 x_2 and then, the original variables x_1 and x_2, now that gives me a five dimensional data, so I took the one two dimensional data and I converted it in to a five dimensional data.

And now, I can do linear regression on this five dimensional data and this is going to end up giving me quadratics. So, I can use the same technique of linear regression and I can get a more complex fits by doing this kinds of basis transformation. So, linear regression is not really that weak of a method because I can do more complex fix using linear regression.

(Refer Slide Time: 20:52)



There are many. many different applications of regression, so the one thing, which I already mentioned earlier was an time series predictions. So, you could try to build a model that predicts rain fall in a certain region or you can try to build a model that predicts, how much money a user is likely to spend on a particular service let us say calling right sometime you can even use linear regression to do classification.

So, instead of saying whether the person will buy the computer or will not buy the computer you can try to predict what is the probability the person will buy a computer and you can try to solve the cerebration problem and is roughly. So, I have more I have more no answers to that right and you can use regression as a data reduction tool. So, instead of giving you like a 100000 data points I can just fit a low dimensional line low dimensional curve to that it could be even a straight line.

And then, I can just tell you that these are the parameters of the line instead of giving you the hundred thousand data points and that allows me to have a very, very large data reduction and the other thing is to look at trend analysis. So, it is slightly different from the time series prediction problem that I was mentioning earlier, because I am not really interested in making predictions here. But, I am more interested in the data analysis part you know not necessarily in the predictive analytics part here.

So, I would like to know whether the growth rate is linear or exponential or somewhere in between. So, those kinds of questions can be answered by suitably solving a

regression problem and then, going back to what I mentioned earlier about risk analysis or in the classification case. So, you could think of risk analysis here, so if you remember in linear regression you learn coefficients for each of the input variables.

So, the input variables that has a larger coefficient is the one that is going to influence the output the most. So, that way you can look at risk analysis by looking at the factors that contribute most to the output. So, this is essentially on supervised learning method, so we looked at two of these regression and classification.