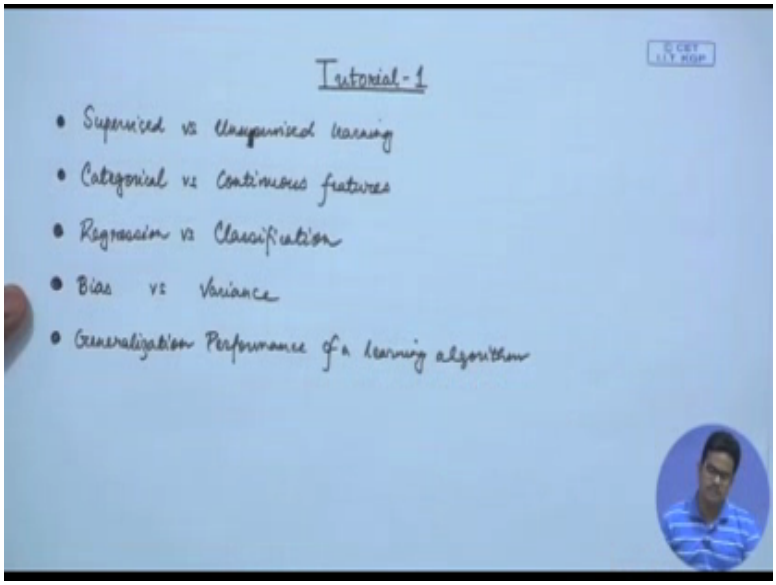
Introduction to Machine Learning
Prof. Mr. Anirban Santara
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Tutorial I

Hello friends, welcome to the first tutorial session of this course. I am Anirban Santara, I am doing my PhD in Machine Learning, and I am a teaching assistant of this course. And in these tutorial sessions, which we will hold one every week, we will discuss the topics that have been covered in the particular week, and do a quick summary of all the topics that have been covered in this week.

And then, we will learn how to solve questions and these questions will be the ones that you would expect in the assignments and in the final exams, all right. So, in this first tutorial class, we will be taking the following topics.

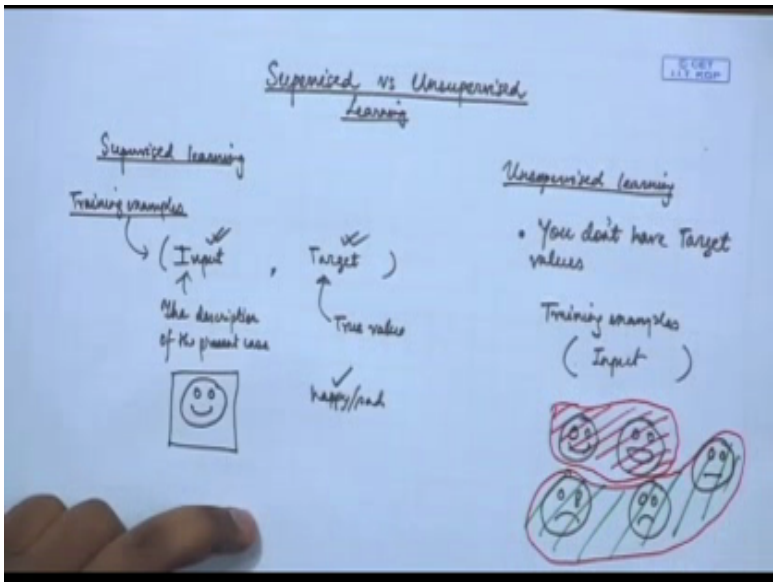
(Refer Slide Time: 00:59)



So, let us go ahead. So, the topics that we will be discussing today will are supervised and versus unsupervised learning. Then different kinds of features - categorical versus continuous features; different kinds of supervise learning problems - regression versus classification. Then we will go ahead to bias variance trade off -bias versus variance and we will study how bias and variance they vary with different parameters of the learning algorithm.

And then we will study how to control, and how to you know detect and analyze generalization performance - generalization performance of a learning algorithm of a learning algorithm. So, let us first take up the first topic of today's tutorial class that is supervised versus unsupervised learning.

(Refer Slide Time: 02:39)



Supervised versus Unsupervised learning; so this supervised and unsupervised learning are two major broad categories of the machine learning algorithms, the only difference, so whenever you have been given a question, and asked whether a particular learning algorithm has been described. So, you have been so there is a big paragraph describing a particular scenario, and you have to identify what kind of learning algorithm is going on in this particular scenario all right.

So, the first thing, whether it is supervised or unsupervised, so the first thing that you are going to look for is the kind of training examples that have been presented to you. So, supervised learning always comes when wait, so let me first, supervised learning. So, in supervised learning, the training examples will always come as pairs. So, the pair so these pairs will be the first term will always be the input, and then you will have the target value.

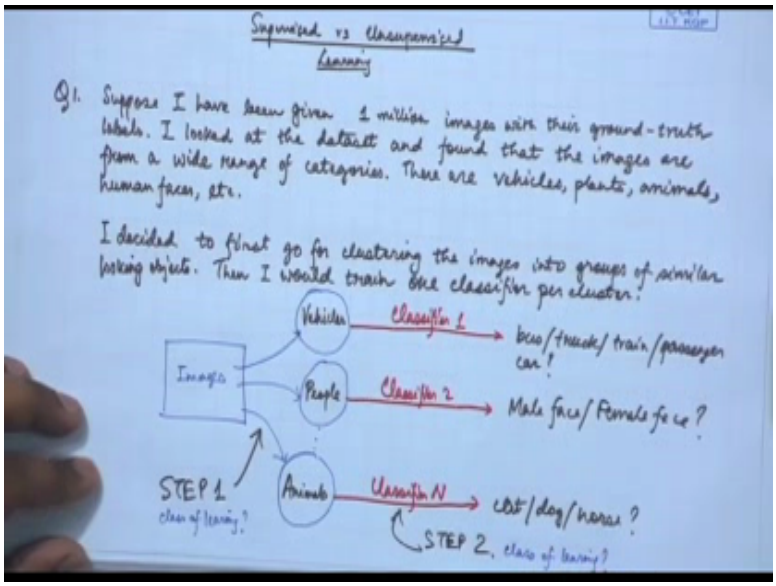
So, this is the true value and this is the description of the present case. So, it could be like an image and the true value may be whether it is happy or sad; so it is already given that this particular image describes a happy face. And so the input examples will always

come as pairs of input and target values. So, if the target values are given, and if have been asked to predict the target values after looking at the input then it is a supervised learning problem. Whereas in unsupervised learning you do not have targets.

In this case, what happens is you are just presented with inputs. So, the examples - the training examples are just input value all right. And you will be asked to find some regular patterns or some kind of information you have to predict just by looking at the inputs. So, maybe you have a set of smileys like this one or this one or maybe you have this, you have this one, or you have this one so just these faces are given. And you do not know, so the information that has been given to you for learning it does not include the information about the kind of gesture that is being expressed through these faces.

So, you do not know a priory which is a happy face and which is not. So, you have been just asked to learn some regularity within this data or find some clusters within this data, so what you will be your unsupervised learning algorithm, if it is a clustering algorithm what it will do is it will try to identify groups based on just the appearance of the example. So, it will group these two faces together maybe, it will group these two faces together, and these three faces in another class, and these three faces together. So, this is just based on the visual appearance of the images, so it is just based on the input data not any kind of target data. So, we have been given this kind of a scenario. So, this kind of a scenario is described.

(Refer Slide Time: 07:19)



So, this is again supervised, unsupervised. Supervised versus unsupervised learning, and let us have this kind of a question. So, suppose I have been given one million images with their ground truth labels. So, I have a huge collection of images, and I know from the data from the data that is given to us. So, the data consist of these image and its corresponding label pairs. So, say there is an image of a truck, and it has been given along with that image that yeah this is the image of a truck, then again may be we have the image of a human face, and it has been labeled that yeah this is the image of a male human face for example.

So, the input that has been presented to you is appearing as the data that has been presented to you is appearing as input label pairs. So, typical supervise learning scenario, but the problem can be made really tricky so let us see what comes next. So, what I do is and this is the general practice in machine learning. So, whenever you are doing this kind of practical or machine learning tasks, so this is what you have to do. So, see what we do next I looked at the dataset and found that the images are from a wide range of categories. And there are vehicles, plants, animals, human faces etcetera. So, what I did is I first did a quick look at had a quick look at the data that has been given to us.

And I found that oh my god, the images are coming from a huge variety of classes, so they are there are all kinds of images. There are images of vehicles, there are images of animals, birds, may be human beings and like everything, so it is very difficult to really train one single classifier that would be specialized that would be able to identify these objects coming from this kind of a wide range of categories this is a very hard learning problem.

So, I just decided to do a trick. Just go a bit different way different from a bit more make the problem a bit more interesting. So, what I decided is I decided to first go for clustering the images into groups of similar looking objects. So, what I first decided is to identify clusters within the image data that has been provided to me. So, maybe if I try to I if I ask my algorithm to identify clusters within the data, data that has given to me within the images just the image part.

So, maybe it will be clustering all the vehicles together, so all trucks, cars, buses, trains, airplanes they all come together under one cluster; may be all human faces come together in a cluster, may be all plants go together in a cluster, may be all animals again

move to another cluster. And then once this clustering has been done, then I decide to train one classifier per cluster, so that particular classifier is going to be specialized for that particular cluster. So, then I would train one classifier per cluster.

So, I can describe the problem graphically this way. So, I had this huge dataset of images, so these are all images. And first I try to identify clusters within this data, maybe I have the example just showing three of them, may be these clusters are like vehicles here and people and animals. So, these kind of clusters first; and then we would make one classifier per cluster say N. We have n clusters and we have N classifiers.

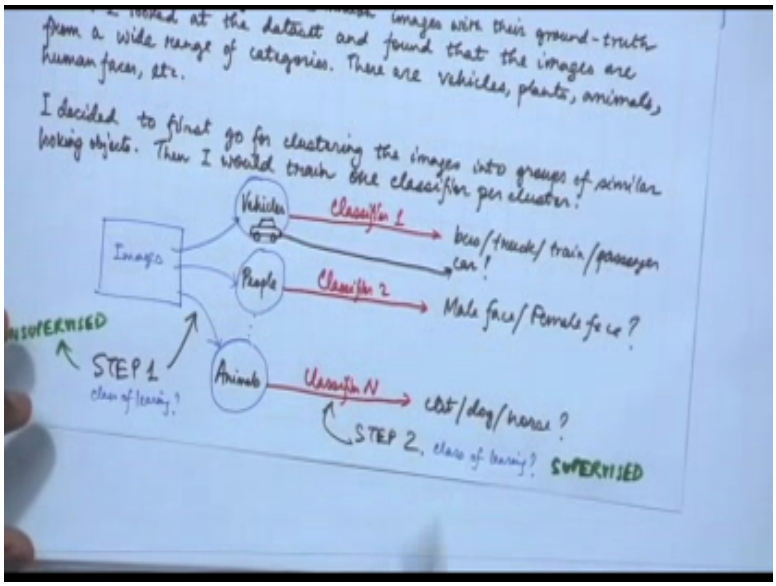
So, each of these classifiers will then try to predict, whether which particular class that particular image is in, for example, this one classifier would be able to tell me whether the output whether the presented image is a bus, or a truck, or a train or may be a passenger car or so this way. And second would be like male face, female face and maybe this is cat, dog, maybe horse.

So, now, tell me which part so there are like this learning algorithm has two steps. So, this is my step one, where I am just looking at the images, and I am trying to group them into a similar looking classes, similar looking I am sorry similar looking groups or clusters. And in the second step, I am trying to I am bring in the class labels that had been had been presented to me.

So, for each particular cluster, I am going to make one classifier, and train it to match the image to the label that has been given to me. So, now, think for a minute, and you pause the video here, and try to identify that what other two different classes of learning that class of learning, so which class of learning is going here, and class of learning, which class of learning is going here, so which part is supervised which part is unsupervised.

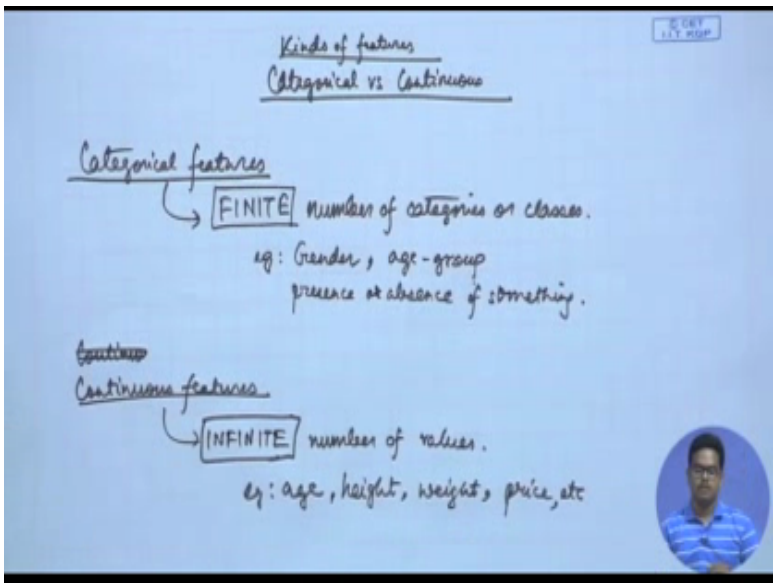
So, just pause the video and for a second and try to identify. So, I hope that you are done and the answer is the first step is an unsupervised learning problem. So, let me write in green, so this is unsupervised, because we do not use the class labels here right, we are just looking at the images so and deciding. So, over here as we had decided and as we had discussed before that in unsupervised learning, we do not have target values; so we are not using the class labels over here. We are just looking at the images and trying to find groups or clusters of images which look similar to each other.

(Refer Slide Time: 16:22)



The second step is supervised hope you can see it, because each classifier will take the images of its own cluster and map them to the target class value of that particular cluster of that particular image within that cluster. So, say an image of a car comes over here, so it is going to image of a car and that gets mapped to a passenger car so this classifier learns this mapping. So, this is the supervised learning problem I hope this makes the things clear, so you are going to face this kind of problems in the exam and the in the assignment.

(Refer Slide Time: 17:18)



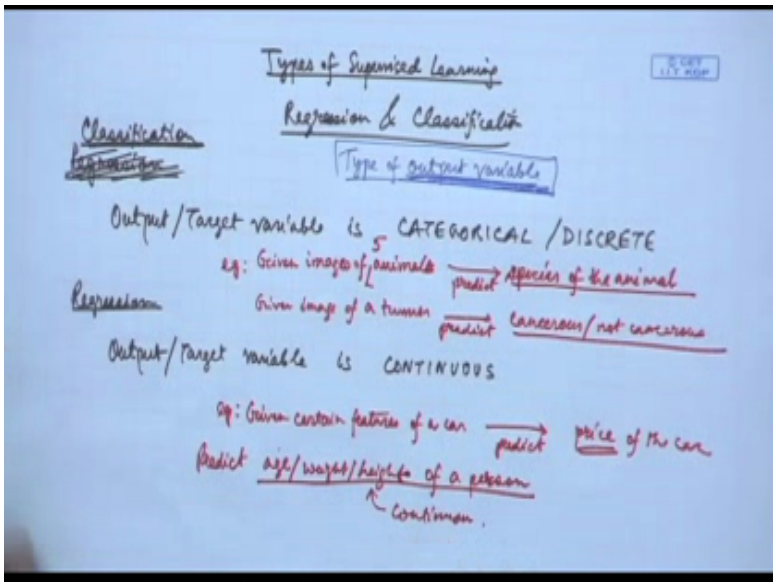
So, let us go and move on to the next topic, which is kind of features. So, we have categorical versus continuous features. So, let us discuss what these two different kinds of features are. So, categorical features are features that have so these features they appear in a finite number of categories or classes. So, for example, gender of a person, then age group of a person like teenagers or children that is those who are below say 13, they are children; 13 to 19 - we are teenagers; and then ahead then we have like young people till 35; and then another age group may be 35 to for 60, so we are like working adults; and then above 60, we are senior citizens and all.

So, these different age groups so this kind of features they are all gender age group. So, these are categorical features alright so these appear in a finite number of categories so this is important. So, you can you can like have a finite or fixed n different values which so that the feature can take. Whereas so even presence or absence of something, for example, maybe you are a doctor, and you are looking at x-ray scans, and trying to predict whether there is a fracture or not.

So, maybe there is some kind of feature like you know sudden change of intensity, so in the middle of a bone, so you suddenly see that at a particular place there is a black crack noticeable in the white region of the bone in an x-ray. And that thing so the presence of that would indicate that there is a fracture, so the bone is broken from main between in the in the middle. Whereas, the absence of that would mean that yeah would not like directly indicate that the bone has a fracture, maybe there is some other features that would further like you can carry on the diagnosis using them. So, this kind of feature like presence or absence of something, this is also a categorical feature because it can take just two values right presence or absence yes or no so these are all categorical feature.

Whereas, continuous features are those, which can take an infinite number of values theoretically infinite number of values. For example, age of a person or something all right height; theoretically, they can take an infinite number of values, weight of a person, then price of something, so these are continuous features. So, it is very easy to understand that which one is a categorical feature, it has to take some an infinite number of values and what a continuous features which can take infinite number of values.

(Refer Slide Time: 21:27)



So, now let us look into types of learning algorithms so that is continued (Refer Time: 21:30). So, types of supervised learning algorithm. So, in this you can face this kind of questions in the exam that these kind of like few of these examples will be given like gender may be or age group, presence or absence of something, or height, age, weight. So, these things will be given to you, and you will be asked to identify whether it is a categorical feature or a continuous feature.

So, you should be able to first like what you question yourself, you just question yourself whether this particular attribute it comes in a finite number of values or finite number of categories or is it a continuous value random variable. So, whether it is a continuous value random variable or it is a categorical value it has a fixed number of values so that so you go ahead and like check the correct answer whatever you think is correct. So, if it is a finite value variable then it is a categorical feature and if it is it can take infinite value theoretically then it is a continuous feature.

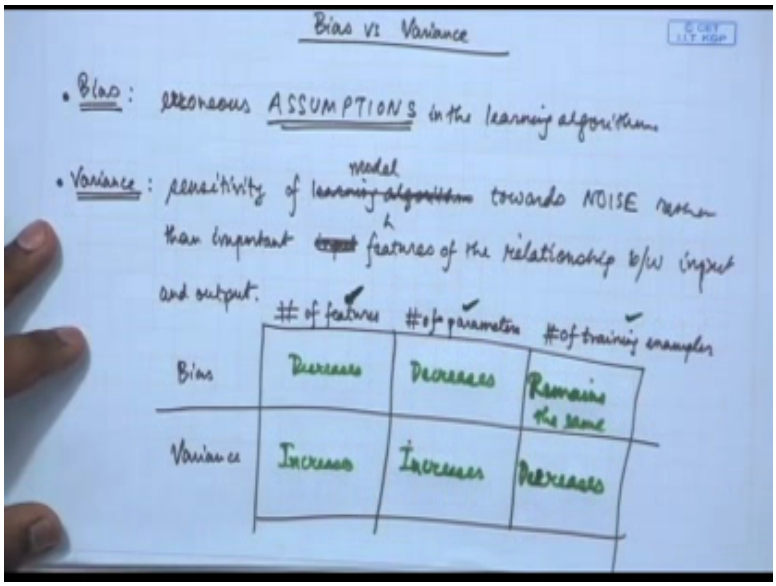
So, next let us move to the types of supervised learning, and yes, supervised learning mind it, regression and classification. So, what is regression? In regression, so it is dependent upon the definitions they are dependent upon the type of the output variable. So, these are dependent upon the type of output variable. And this output variable is what matters.

So, whenever you are given a problem that whether certain learning problem is certain kind of supervised learning problem is supervise regression or classification then you have to just check what is the nature of the output variable whether it is discrete or continuous. So, in regression problem, the output variable or target variable may be is categorical or discrete, whereas oops, sorry in classification. I am sorry; so in classification, the output or target value variable is categorical or discrete. Whereas, in regression, the output or target variable is continuous.

So, let us take some examples. So, classification problems can be like, say you have been given images of animals, and like and they could be either of like images of like I would say 5 animals. And you have to predict like species of the animal or the kind of animal, you can assume. So, what kind of animal it is whether it is a dog, or a cat, or a horse. So, this is a classification problem, because the output variable you know that it is discrete it is like five different values only. And may be something given image of a Tumor, predict cancerous or not cancerous is malignant or (Refer Time: 25:54). So, you can see that the output variable is appearing in two different like two distinct classes. So, it is a classification problem.

Whereas the examples of regression problem could be predicted, they given certain features of a car, predict the price of the car. Now the price of the car could be any continuous number, so this is a continuous valued variable that is you are going to predict and it is a regression problem. Or similarly like the age or weight height person of a person it is predict. So, these all are continuous variables and prediction of this continuous variable from certain attributes of the person is a regression problem.

(Refer Slide Time: 27:13)



So, let us move on to the next topic, which is bias variance versus variance. Well, what is the bias of a learning algorithm? The bias of a learning algorithm is a set of erroneous assumptions in the learning algorithm; and so this particular thing bias, it is due to the learning algorithm. So, it is not due to the training examples that have been given to you, so there are certain assumptions within the learning algorithm which keeps your model limited and its capacity to learn, so that is what is bias.

And due to bias, you can miss out important relations between inputs and outputs. So, you would scan c, you can observe that yeah there is a valid very valid input-output relationship, but your learning algorithm cannot capture it just because it is not (Refer Time: 28:18) enough of it is not capable enough to learn that particular thing that particular relationship between your input and output, so that is a bias of your learning algorithm.

Whereas of variance of a learning algorithm, it is due to sensitivity of your algorithm towards noise rather learn I would say model. So, the model you are learning towards noise rather than important output or important features of the relationship between input and output. So, variance happens when you are learning algorithm, it tries to concentrate on unwanted like trivial or noisy variations in the data rather than important aspects of the data that needs to be captured.

And it happens when you have too many features or you have too many you know too many parameters in your learning algorithm in your model and or you do not have much training data. So it does not understand, does not cannot like really realize what kind of variations are predominant in the data between important or what kind of relationships between inputs and outputs are actually you know important, and what actually matters. So, it does not have enough examples and cannot realize.

So, let us try to predict how bias and variance varies with if you very different aspects of our learning algorithm. And we will study the variation of bias and variance with three things. So, number of features, number of parameters of your model and number of training examples. So, when you increase the number of features of your learning algorithm, then the bias decreases, because your learning algorithm it looks at more and more features of the data, and becomes more and more sophisticated, so the bias decreases, whereas the variance increases. Because the more the number of input features it looks at the more you know is the amount of noise that it is gets exposed to alright so more it becomes acceptable to or yeah more it becomes acceptable to modeling the noise or other important features.

So, with a number of parameters of the learning algorithm, so again bias decreases; why, because it becomes more and more capable, so the number of parameters as you increase the number of parameters of your learning algorithm, it gets the model becomes more and more sophisticated. And you know it can it is capacity learning capacity increases.

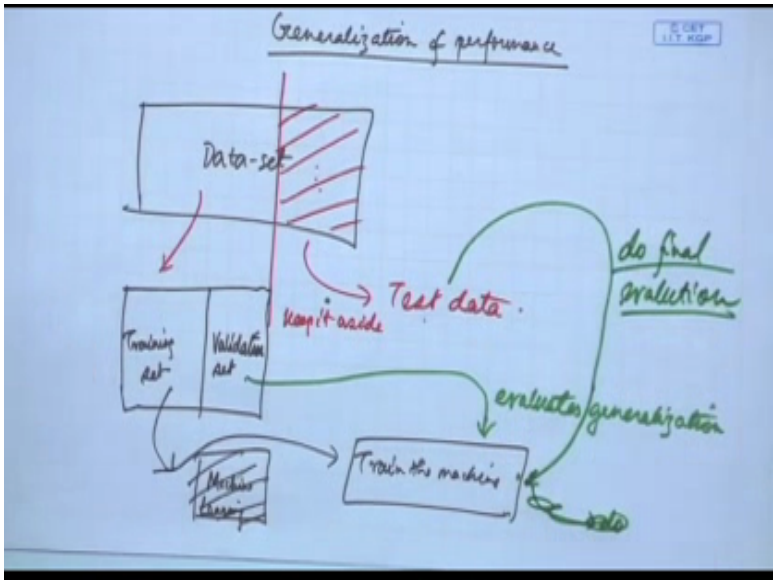
Whereas the variance again increases, because it can use the parameters the more parameters it got to model noise more efficiently. Whereas with the number of training examples, bias remains constant remains the same, because bias is not a property that is directly related to the number of training examples that has been presented to the learning algorithm rather it is a limitation of the model of the learning algorithm. So your learning algorithm, your model is not capable enough to learn important features in the data and that is why bias comes in and it is not a problem which comes directly from the training data.

So, if we increase the number of training examples, and your model remains equally incapable as before, so the bias process and you cannot do anything; whereas the variance goes down, it decreases. This is because you have variance is coming because

like your model is sophisticated, it has the capacity to learn, so which was like grabbing on the noise and trying to model the noise.

But whereas you show it more and more data valid data points then it learn to really understand what kind of properties occur in general within the data. So, what kind of important which particular relationships between input and output, they are predominant within the examples, and hence are important to model. So, you will be asked how bias and variance of a particular learning algorithm will vary as these different kinds of parameters of a learning algorithm are changed, you know modify it.

(Refer Slide Time: 33:48)



And the last thing that we would going to we are going to discuss today is generalization of performance. So, what do you mean by generalization of the performance of the learning algorithm. The generalization means that how good the learning algorithm would perform when it is presented examples, which it has not seen before. So, you have a machine learning algorithm and you have got a say a million training examples, and you trained it on the million training examples; however, you have no idea of what kind of training examples what kind of examples it is going to see when it is deployed in the real world.

So, at any time say, you have a machine learning algorithm which is looking at a scans of x-rays and trying to or may be let us make it more interesting, may be scans of you know may be microscopy images of cells, and trying to predict whether there is a cancer or not,

and biopsy results and trying to predict whether the result is cancerous or not. So you have trained the machine learning algorithm with a huge number of training examples, but still you do not know in a clinical setting rare type of cancer or rare kind of like set of symptoms may appear within you know image that has been presented to it to the learning algorithm to the machine that you trained.

And it may mesh things are because this particular set of symptoms which is really rare, did not appear in the training example set, so that is why it will fail in that particular scenario. So, what is going to happen in the real world, when the system is really deployed is what really matters and that is why that is what we mean by generalization performance.

So, you have a certain training error, but and that is say we say the let us assume that that training error is really small and so you are confident that yeah your system is performing really great on your training set. But to get an idea about how good it would perform in the real world when it is really deployed, you have to go and check the generalization performance of the learning algorithm. So, the generalization performance of the learning algorithm can be controlled by controlling the bias and variance of the algorithm and that is what you will be taught in the rest of the course, as we go along. We study one model after another, and we will check, we will study how to control the generalization performance by controlling the amount of bias or variance.

And, but how to check the generalization performance, so that is why whenever we have been given a big data set, we make you know what do we do we just randomly select a particular section of the dataset and we keep it aside. So, keep it aside as test data. So, we buy we buy no means are going to show the algorithm, this particular part of the dataset that has been given to us.

So, this particular set of examples will be unseen to the learning algorithm to our machine while it is getting trained, and the remaining dataset what we have, so we are going to split it again into a validation set and the training set. So, the training set is what is used for machine learning for training the machine rather I would say that this goes to train the machine. So, say after a certain period of training, so say you are doing some like may be 30 rounds or 50 rounds of training, you are showing the training examples is getting out giving outputs. So, you are checking the amount of error that has been

incurred in that particular episode and then you are trying to reduce the error by gradient descent or some other algorithm.

So, after training the machine for sometimes you want to go ahead and check that how good the machine is generalizing. So, this validation set was held out, so it was not being shown to the machine while being trained so you will be just going to train it on the training set. And then this validation set comes in and evaluates generalization. So, you check how good how the machine is performing on the validation set.

So, if you see that the validation error is quite high, while the training error was low you can say that yeah the machine is not generalizing, my machine is not generalizing, so I need to check the learning algorithm. I must you know train it all over again, because when you are showing some data that it has not seen in the training examples, then its messing things up. So, we go like in the entire process of training at regular intervals of time, we stop the training of the machine and then checks his performance on the validation set. And this gives us a measure of how well the system is generalizing.

Now this way we train the machine. So, we started from a randomly initialized machine, we trained it on the training set, and periodically you know evaluated on the validation set and finally, we have a machine which has low training error as well as low validation error. So, I have a little bit of confidence that yeah the machine is has learned the problem very well and so it is having a small amount of training error, also it is generalizing quite well because it has a small amount of validation error as well. Now, as of first now you must be careful to notice that this validation data validation set was not use for training the system, but it was used to evaluate the system while it was getting trained.

So on the basis of the validation set results; we were going and tuning the learning algorithm, just to make sure that the next that next time the validation error is small. So, in some way this, the machine has seen the validation data, but it must be evaluated to get a taste of the true error that the machine would incur when it is deployed in the real world, we must show it some data, we must evaluate it on this show on some data which it has never seen before and that is where this test data comes in.

So, this comes in and do so I will write here do final evaluation, so after the system is completely trained and it is ready to be deployed. We bring in the test data and we check

the performance of the machine on the test data we check the test error and we report it that, yes, this is an estimate of the true error that the system is going to incur when it is deployed.

So the kind of questions that you are going to face from this section is like you would be asked to like say whether the training set is a measure of the true error or the training error is a measure of the true error, or whether the test error or the validation error are measures of the true error.

So, the training error is not at all the measure of the true error because it is the error that has been you knows incurred by the system on a set of training examples, which it has already seen during its training, and it was improving on that particular set. Whereas the test data is completely unseen, so it is the measure of the true error when it will be deployed in the real world. And the validation set it gives an estimate of the generalization performance while the system is (Refer Time: 41:46) be trained.

So, best of luck, all right, go ahead and solve this week's assignment, and the deadline is 27th of July, 11 P.M - Indian Standard Time all right. So just go ahead and you will find familiar questions in this assignment, and solving this assignment will give you a good understanding of the topics that have been covered in this course. So, best of luck, bye, bye, see you next time, peace.