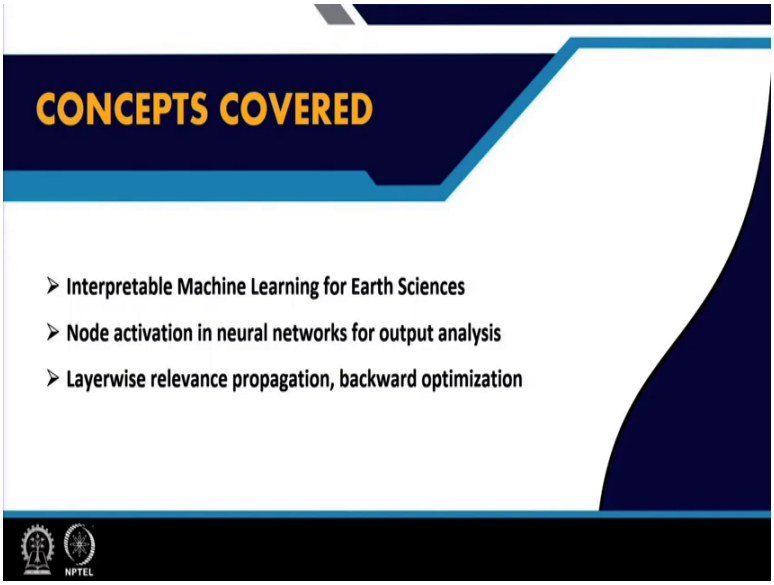
Machine Learning for Earth System Sciences
Prof. Adway Mitra
Department of Computer Science and Engineering
Centre of Excellence in Artificial Intelligence
Indian Institute of Technology, Kharagpur

Module - 03
Machine Learning for Discovering New Insights
Lecture - 26
Interpretable Machine Learning for Earth System Science

Hello everyone, welcome to lecture 26 of this course on Machine Learning for Earth System Science. This is the last lecture of module 3, where we are considering machine learning for discovering new insights in the geosciences. Today's topic is Interpretable Machine Learning for Earth System Science.

(Refer Slide Time: 00:45)

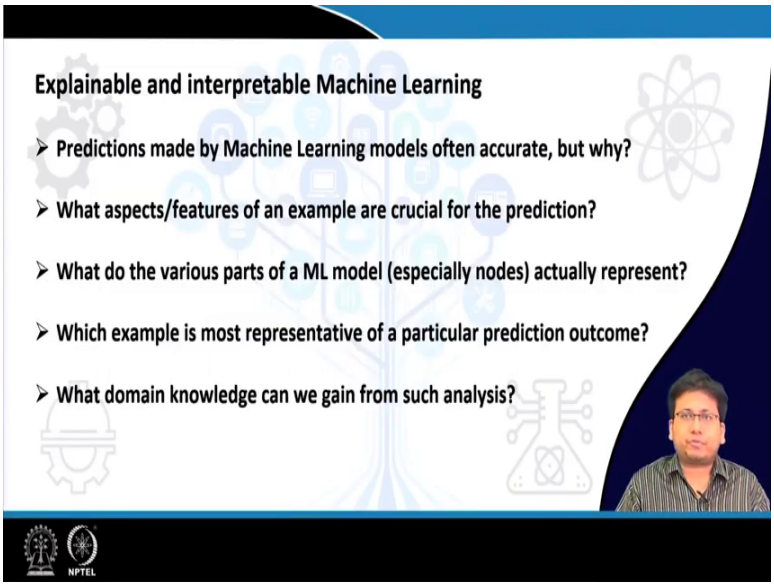


So, the concepts which we are going to cover today are how machine interpretable machine learning can be used in this domain. Secondly what is like in case of neural networks, how like how node activation can be used for output analysis for the analysis of input and output.

And based on that we will see like see other concepts like layerwise relevance propagation backward optimization etcetera. These are all concepts related to neural networks which

basically or rather which are related to like under making sense of how a neural network makes its predictions.

(Refer Slide Time: 01:21)



So, first of all like what is explainable or interpretable machine learning. Now these two terms explainable and interpretable, they are which are often used interchangeably though they should not be, but often they are. So, basically the idea is that.

So, we know that machine learning models often make very accurate predictions. We have seen that in the in several of the past lectures in this course itself, where we have seen where we have shown how machine learning models are capable of making very good predictions much better than what the classical methods would do.

But the question might arise that, why is it so? And like which aspects or which features of a particular example are like does the machine learning model focus on to make the prediction. Because the deep neural networks that are being constructed they are like mostly black boxes.

We do not know what calculations exactly happens in the nodes and the edges etcetera, we see the like the output that we see often looks more like magic and the climate scientists may like or the geoscientist, who are receiving these predictions; they may be like curious that why at all did the model predict in the way they did.

Because it is not only about making predictions; we also want to know the science behind it. We want to discover new insights about the process. In fact, that is what this module is all about. So, just making great predictions is not enough we have to understand why those predictions were made.

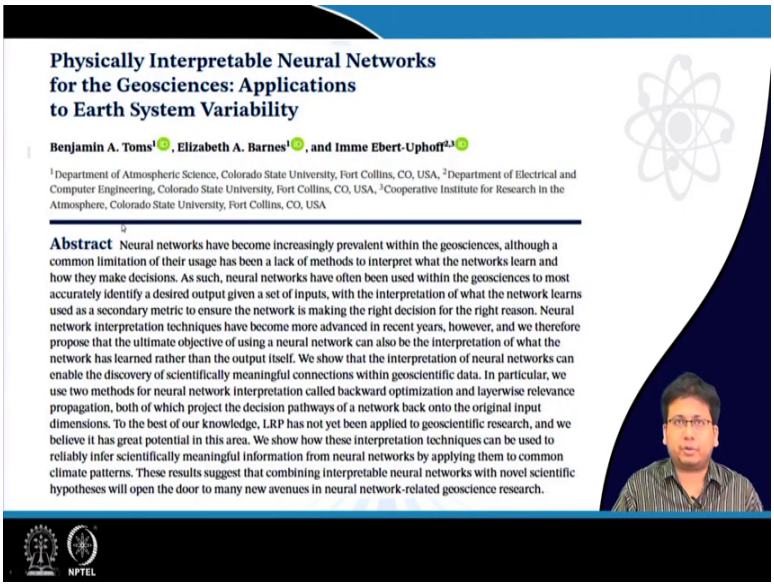
So, and one part of that is asking like, for any given example which parts of the example or which features of the example were most relevant in making that prediction. And secondly, what are the exact roles of the different parts of the model that we developed.

So, if it is a neural network model. So, all so many nodes etcetera there. So, many layers are there what information is actually calculated by them. So, now the we may also be so like, but can those mathematical operations be like represented in a more comprehensive way. That is or that is by a domain scientists can they make any sense of those computations.

Also some questions might be asked that, so many examples are there. So, many training examples are there each of which are like each of which are mapped to some kind of predictions by the by whatever model we are using. So, for now for a particular outcome for a particular prediction outcome if you are doing some kind of a classification analysis.

Now is there something like the optimal or the prototypical input which causes which causes that prediction outcome. And finally, what is the domain knowledge of about the process that we can get from this kind of analysis.

(Refer Slide Time: 04:52)



So, all these are questions related to explainable and interpretable machine learning which we will study here through, three research papers that have appeared mostly in the last 2 or 3 years. So, the here is the first paper physically interpretable neural networks for the geosciences applicability for earth system variability. Neural networks have become increasingly prevalent within the geosciences, although a common limitation of their usage has been a lack of methods to interpret what the networks learn and how they make decisions.

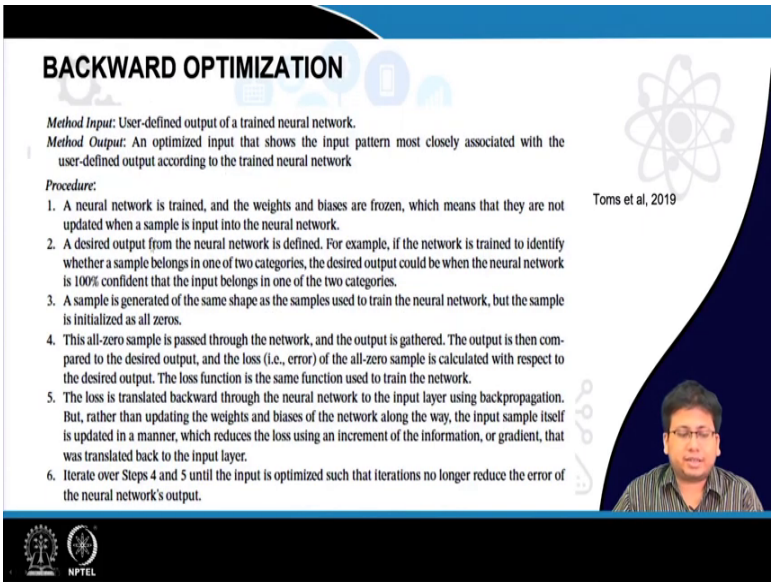
As such neural networks have often been used within the geosciences to most accurately identify, a desired output given a set of inputs with the interpretation of what the network learns used as a secondary metric to ensure the network is making the right decision for the right reason; this is important.

Because that is it is not enough to make an accurate prediction, but rather the prediction should be done for like for on the basis of some features or something which are physically meaningful. Otherwise like it becomes a case of say spurious correlation or something like that, that is not what scientists want we like.

Neural network interpretation techniques have become more advanced in recent years. However, and we therefore, propose that the ultimate objective of using a neural network can also be the interpretation of what the network has learnt rather than the output itself.

We show that the interpretation; we show that the interpretation of the neural networks can enable the discovery of scientifically meaningful connections within the geoscientific data. In particular, we use two methods for neural network interpretation called backward optimization and layerwise relevance propagation both of which project the decision pathways of a network back on to the original input dimensions.

(Refer Slide Time: 07:16)



So, like let us first understand these two methods backward optimization and layerwise relevance propagation. So, backward optimization, the method input is a user defined output of a trained neural network. So, we have a neural network and let us say let us begin with an output. So, let us say the neural network predicts the class labels and for a particular example let us say the possible. Let us say for a particular task the possible outputs are class A, class B and class C.

So, let us say for example, focus on class A. Now the question which I am asking is what kind of input will result in a prediction of class A? Or what kind of a input will result in a prediction of class B? What is a typical input for class B? So, the method output is an optimized input that

shows the input pattern most closely associated with the user defined output according to the trained neural network.

It will become a bit more clear when we come to the specific examples. So, the procedure is as follows: first the neural network is trained and the weights and the biases all the weights and biases are etcetera are frozen which means that they will no longer be updated.

So, like the training of the neural network etcetera is done all its parameters are like fixed and frozen. Now a desired output from the neural network is defined as obtained by the as specified by the user say class A or whatever. For example, if the network is trained to identify whether a sample belongs in one of two categories, the desired output could be when the neural network is hundred percent confident that the input belongs to those categories right.

So, as I said again class A, class B and class C these are the three possibilities. Let us say for class A if there any input for which the model will be 100 percent certain. That yes, it is indeed of class A. So, that is a typical input like, we can say at that typical prototypical input for class A as far as the model is concerned.

Now, what we do is a sample is generated of the same shape as the samples used to train the neural network, but it is the sample is all initialized with as zeros. So, basically its like if the it takes the zero if it takes $m \times n$ matrices as input the neural network. Then we start off with a like, so like basically we are trying to estimate the values of x. So, let us start with some x which is like all zeros initially.

This all zero sample is passed through the network and the output is generated. The output is then compared to the desired output and the loss of the all zero sample is calculated with respect to the desired output. So, for the all zero sample some like the network will make some kind of prediction that prediction is compared to, the particular output we have in mind which for which we are searching for the optimal input.

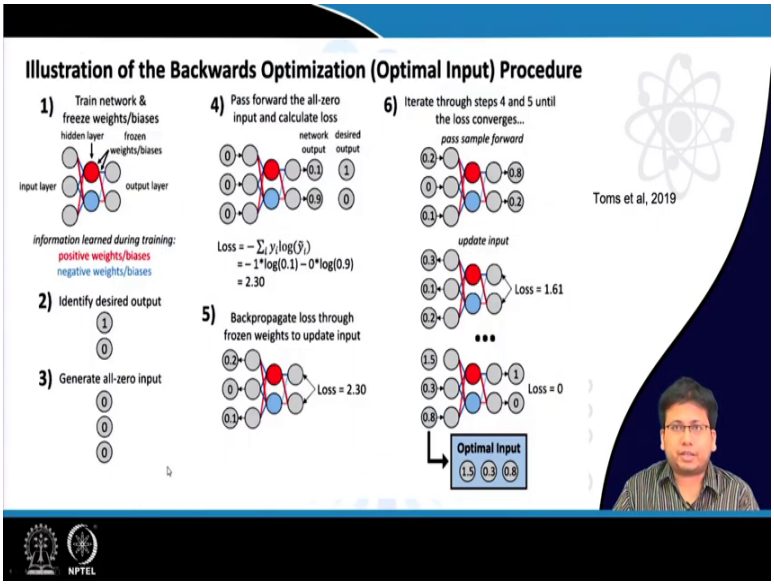
And so they are compared and some loss some value of the loss is calculated. Now, the loss function is the like same function as is used to train the network. The loss is translated backward through the neural network to the input layer using backpropagation. But rather than updating the weights and biases of the network along the way the input sample itself is updated in a manner,

which reduces the loss using an increment of the information or gradient that was translated back to the input layer.

So, like in normal backpropagation the input remains constant, but its the really the weights the $w's$ that are updated through gradient descent. In this case is the reverse in this case we will assume that all the $w's$ the weights they remain unchanged, they are fixed at the values that was obtained by the process of training. But rather it is the input values x that are going to be updated when we do the backpropagation.

So, the variable is no longer w, but instead x itself is now the variable and they are going to be. So, the input value to every node is going to be updated like as the error is back propagated and that is why the this updation of the input values will take place from the output to the input. So, the these steps are iterated, the steps 4 and 5 are iterated until the input is optimized such that the iterations, no longer reduce the error that is basically it converges.

(Refer Slide Time: 12:20)



So, just like in the in normal backpropagation that like we say it has converged when the w values no longer change in this case, we will say convergence has happened if x no longer change.

So, here is an like cartoon which shows the illustration. So, let us say that this is the neural network. So, this is like let us for simplicity let us assume that this there is only one hidden layer with these two variables and so like these are the different edges whose weights have already been estimated and this is the output layer.

So, like there are like. So, so like let us assume say that these indicate the probabilities of the two classes that, so let us say that its a binary classification problem. So, there are two classes and these node these two output nodes they indicate the relative probabilities of the two classes. So, here is an in let us say that the desired output is one zero; that is class A is I have the basically, I want to see the prototypical input for class A.

So, that means, the situation where the first node which represents class A that has full probability one the second node which represents class B that has probability 0. So, I am looking for the input which causes 1 and 0 at these output variables. So, as discussed here, we start with the all 0 samples.

So, this is the all 0 sample it is provided like it is provided as the input, so it like this the all the calculations take place and let us say we find the output values come to 0.1 and 0.9 which is of course, very different from 1 to 0. So, we calculate the loss function.

Now since this is a classification problem the loss function is probably the cross entropy loss function. So, this is the probability distribution over the classes and this is the desired probability distribution. So, two probability distributions can be compared using the this thing like using the cross entropy.

So, we get the loss value of 2.3. So now, this loss is now going to be backpropagated through all these through all the nodes like this from the output towards the input. So, like as I said as the backpropagation happens, it is the input values which get updated not the weights.

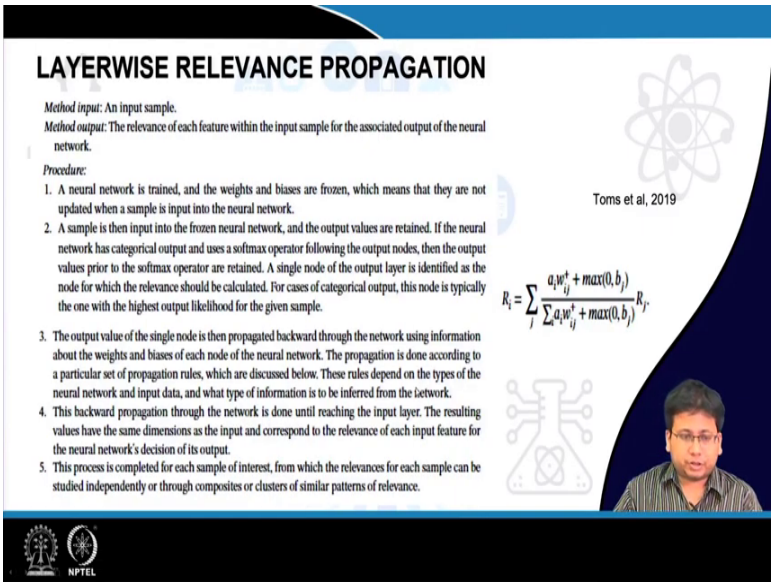
So, like so based on these 0.1 and 0.9 first these two nodes that their values get updated then these two then based on their values the values of these two nodes get updated and finally, the input values also get updated. So, let us say that this becomes the updated input values 0.2, 0, 0.1. Now we just keep on iterating like this over and over.

So, let us say in this case when we use this as an input 0.2, 0, 0.1 again by using this as the input. Let us say the output changes to 0.8, 0.2, but that is still different from the desired 1 and 0 some residual is still there. So, again that error we calculate and again we backpropagate it like this again updating all the input values, but keeping all the weights constant.

So, let us say after that we get this as the updated input 0.3, 1, 0.2. Again this see and in this case the loss is 1.61. So, this is better than the loss we had obtained earlier which was 2.3. So, we just keep on doing this doing this until the loss is very nearly equal to 0 and then we find that like whatever the out the input we get like that that is the desired input.

That is we see that for one the input of in this case 1.5, 0.3, 0.8; the output prediction is class A probability is one class B probability is 0 and this is exactly what we desired. So, this is then prototypical feature vector which will be like confidently classified as class A. Similarly for class B where this one has 0 value and this one has 1 value we can calculate the prototypical example ok.

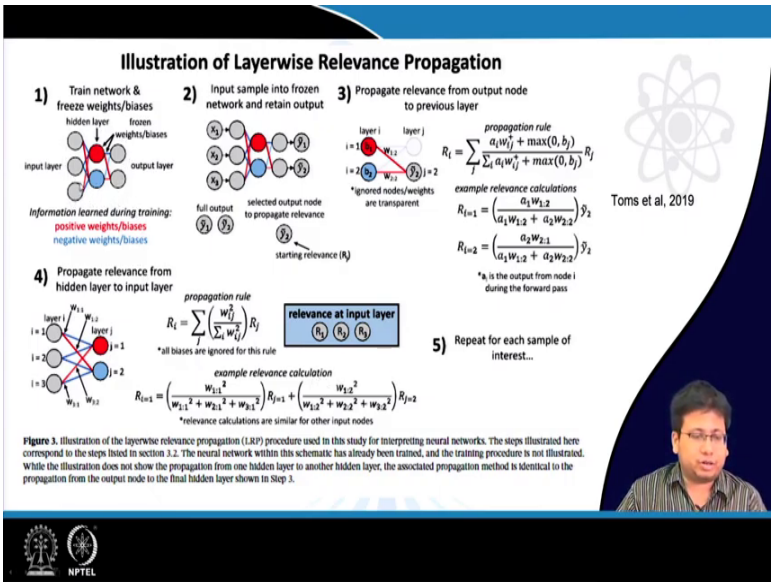
(Refer Slide Time: 16:50)



So, this is the task of backward optimization, where for a particular output we try to find it is its input. The next task is layerwise relevance propagation which is like in this case the input is again is an input sample. And the methods output is the relevance of each feature within the

input sample for the associated output that is the. So, when the input is presented the neural network which you have created that will definitely give some output, but which of the input features was actually most crucial or most important in determining that particular output.

(Refer Slide Time: 17:35)



So, if you look at this neural network. So, like here let us say that these are this is my input x_1, x_2, x_3. And once again let the there is no optimization of the weights to be done that is the neural network has already been trained all the weights etcetera are frozen. So, now let us say that when we provide x_1, x_2, x_3 as the input the output is found out to be y_1 and y_2.

So, now, I will be so there are two output nodes. So, now what I am interested is that what is the relative importance of these features in choosing the or in like or in creating this value of y_1. That is which of these features determined that this particular output node is going to have the value of y_1, instead of some other value or in for this node which the features which ensure that is value was y_2 right.

So, for that what we do is layerwise relevance propagation that is we calculate some a quantity known as relevance and then we propagate it backwards like that. So, like when like that. So, the

initial the let us say we are focusing on this particular node and we are trying this out particular output nodes.

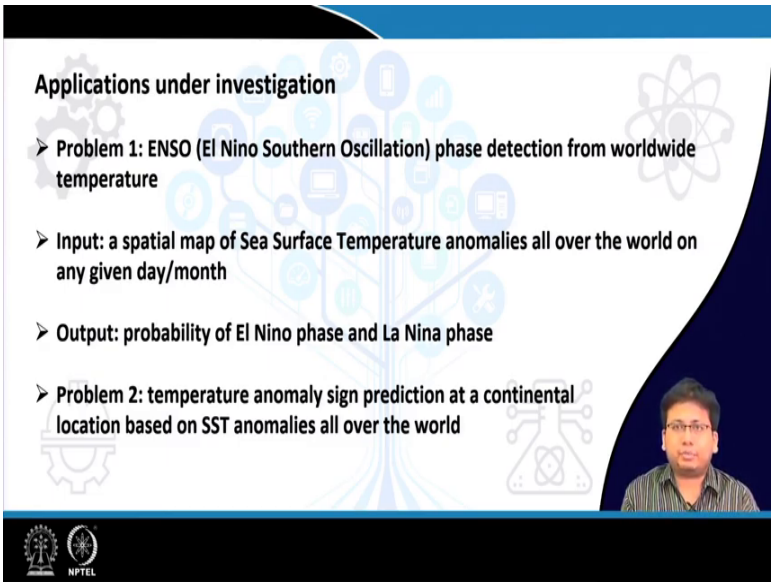
So, let us say the relevance value is initially set to this y_2 value itself. Now according to that this value is connected to these input nodes then this input node is connected to these two hidden nodes etcetera so for.

So, whenever we see this kind of a situation that is two successive layers i and j in the j^{th} layer I have a like a value whose relevance. I mean there is a node whose relevance I know or I have calculated. Now that relevance I have to propagate back to the connecting connected nodes in the previous layer and for that we have a rule for propagating these relevance values so like. So, initially this one will have its relevance value which is its the output value itself.

Based on that in this node we will calculate the relevance value according to the formula. So, see R_i for any node i which is in let us say in a in any layer, I am calculating that relevance value R_i in terms of R_j which is the next layer towards the output. So, from the output again the relevance values are flowing backwards towards the input according to these formula. So, that way for starting from this output node we calculate the relevance values for all the nodes that lie on its path back to the input.

So, the so that way like for the input variables also or till the input variables these relevance value is backpropagated. And so these relevance values what they give us at once they reach the input variable the input nodes what they give us is the like something like the relevance of the different inputs.

(Refer Slide Time: 20:56)



So, so these are all conceptual ideas of interpretable or explainable machine learning. Now, they are application in the domain of earth system science. So, we consider two problems you can say that they are something like toy problems, I mean which are not really the biggest questions of earth sciences, but like mostly they are for demonstration purposes. So, the first problem is the El Nino southern oscillation is a phenomena in the central pacific ocean where a particular part of the central pacific ocean known as the Nino 3.4 region that gets very hot in some years and very cool in some other years.

So, those two so it is like there is a oscillation in a phase; there is a positive phase when it is very hot that that phase is known as El Nino and then there is a negative phase called La Nina when that region becomes cold. And now this is like a very fundamental thing concept in the very fundamental phenomena in, climate sciences in the sense that although it happens in central pacific ocean, but its repercussions are felt all over the world, in different parts of the world.

So, the input is a spatial map of the sea surface temperature anomalies all over the world on any given day or any month ok. And the output is the probability of the like El Nino or the La Nina phase or on that particular day or month. That is we know that this like what happens in the central pacific it determines or like which influences the sea surface temperature all over the world.

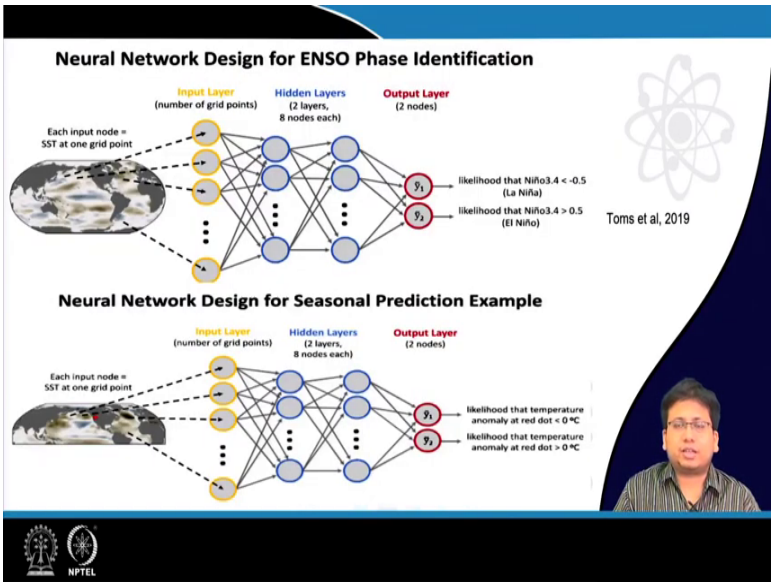
So, the question is that is being asked is suppose in a given month let us say January 2022 this is what the sea surface temperature all over the world looks like. Now based on that tell me are we currently having an El Nino or a are we having a La Nina ok. So, the that is the output probabilities this is the first problem we will consider.

The second toy problem we will consider is like the temperature like let us consider the temperature anomaly or specifically the sign of the temperature anomaly positive or negative, on a particular continental region that is some location on the land let us say Kharagpur.

So, let us say today is or ok let us say you have the sea surface temperature anomalies from all over the world, based on that you have to predict whether today in Kharagpur is going to be hotter than usual or colder than usual. That is are we going to have a like positive temperature anomaly or negative temperature anomaly this like this can also be cast as a seasonal prediction problem instead of a daily anomaly we can consider as monthly anomaly or something like that.

So, is it a hot time in Kharagpur or is it a cold time in Kharagpur at in that is what you have to understand based on the sea surface temperature anomalies on all over the world.

(Refer Slide Time: 24:14)



So, these are the two problems for like both are like basically something like binary classification problems. So, we train a neural network for that the in both cases the input is the world map of

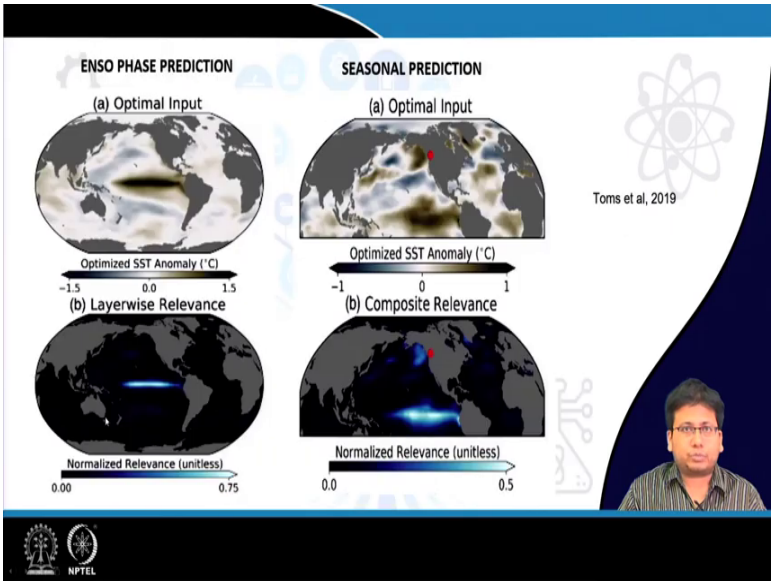
the sea surface temperature and the output is like one of two values or rather we can say the probabilities of the two classes.

So, we have already seen how like the spatial map of a particular variable in this case sea surface temperature can be provided as the input. So, there will be convolutional layers and all kinds of things. So, let us say that instead of going for a let us say we have built some kind of simple neural network, which is capable of doing this prediction reasonably well.

Now the question I am asking is, suppose based on the world map of sea surface temperature anomalies it is actually able to predict like is it El Nino or is it La Nina. Then the question is exactly which like that is I may be interested in asking for what kind of pattern of the these temperature anomaly is over the world will I be certain that it is indeed El Nino or it is indeed La Nina.

Similarly, in the second case also suppose like it is a hot day in or a hot season in the location that is like that that is being investigated. So, then like on like what kind of a spatial map of these sea surface temperature anomalies, will make us most confident that it is the it should indeed be a hot day or it should indeed be a cold day at the location that we are interested in. So, like in these maps they have indicated the location by a red dot.

(Refer Slide Time: 26:14)



So, this is what the answers to the questions is. So, if you in the first problem of ENSO phase prediction. So, this is what the optimal output looks like. So, optimal output so basically it took. So, it shows that to like it might be taking the whole worlds sea surface temperature anomaly as inputs, but while predicting whether there is ENSO or whether there is El Nino or La Nina is really this part which like which is which it is focusing on.

Similarly, if you look at the layerwise do the layerwise relevance propagation where like in different parts of or different input features you are checking its relevance. So, you can see that. So, this black value means low relevance and this white value means high relevance.

So, you see that it is the observations in this central pacific region these are the region what have the high relevance in predicting whether it is an El Nino or La Nina while the others other feature they have no relevance little or no relevance.

So, in a sense it is like telling this is like this should the this answer should be obvious, because we know that El Nino happen or La Nina happens due to a hot phase of in this particular region only in the central pacific region which is known as Nino 3.4. So, if we have to understand from sea surface temperature anomalies if we have to understand whether it is a El Nino or La Nina then we should be focusing on this region only not on any other region, that is common sensical.

But that common sense had not been had not been communicated to the model in any way it was just given maps of the sea surface temperature all over the world and we had been informed that this is a an El Nino phase or this is a La Nina phase the model is itself able to understand that, it is because of this region that you are calling it either the El Nino or the La Nina.

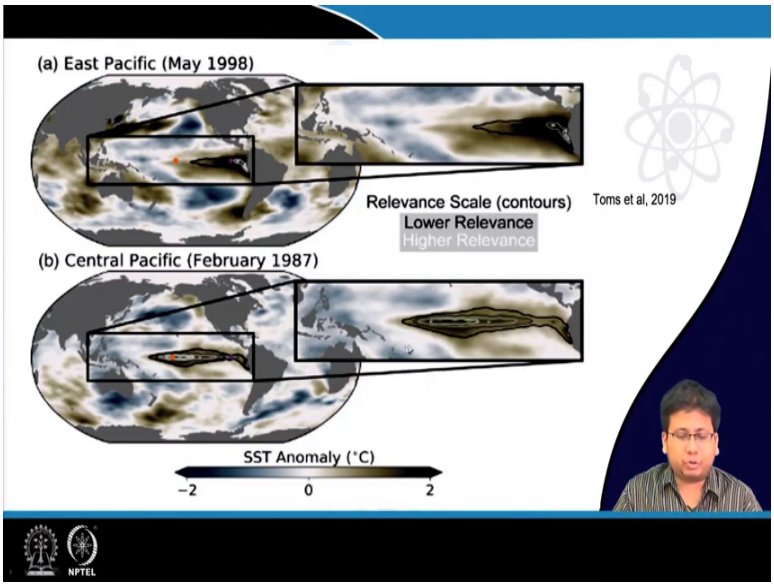
Similarly in case of layerwise relevance propagation also it is really it is precisely these locations which have the relevance not in other not any other location. This is something which is like which the model had did not know, but it is able to figure it out from the data.

Similarly in case of the seasonal prediction problem like we find that the optimal like if we are focusing on this particular region in along the western coast of Canada like let us say it is a hot day in the western coast of Canada then what does the spatial map of sea surface temperature look like and it is this is what is shown.

And similarly when we are calculating the composite the relevance we see that the its really these regions the southern pacific or the we can say the south central pacific ocean, as well as some regions of the pacific ocean here which have the most relevance in like determining whether this location of interest is going to have or is having like is having high temperature or not.

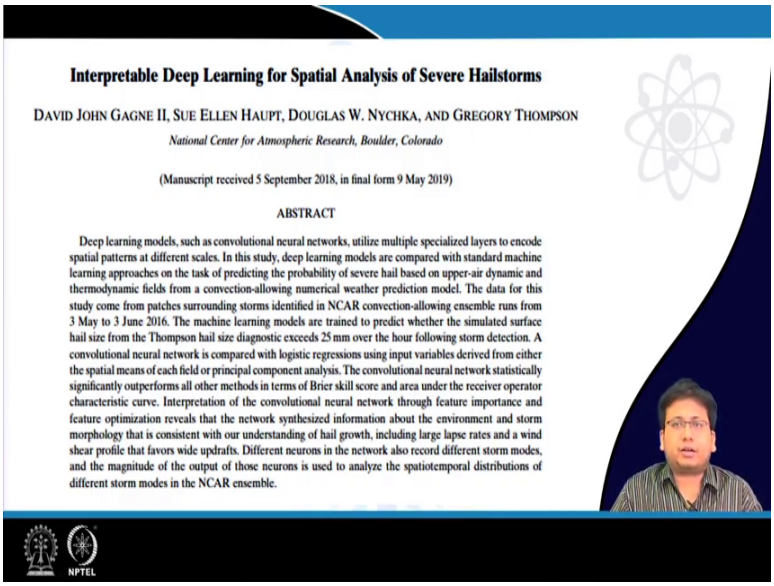
So, this is the result we get for this location if we consider some other location like Kharagpur as I mentioned then we may get some other result.

(Refer Slide Time: 29:54)



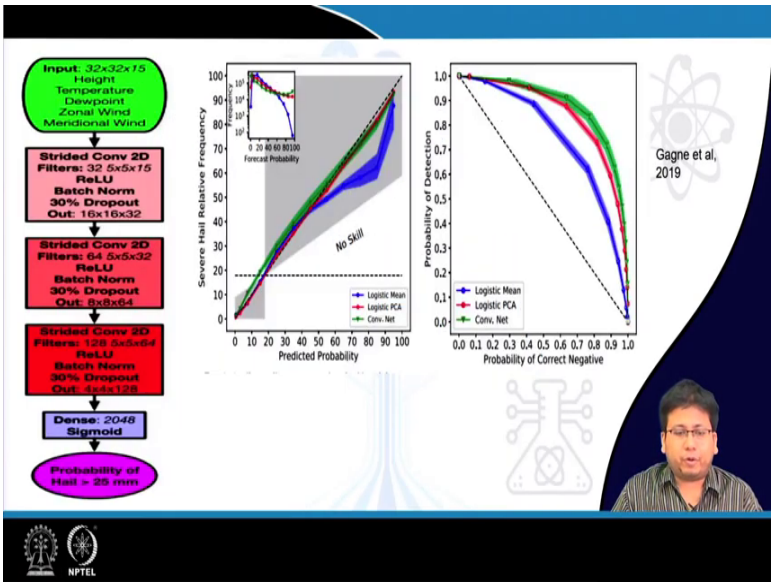
So, like these are the more detailed maps of the of those relevance.

(Refer Slide Time: 30:00)



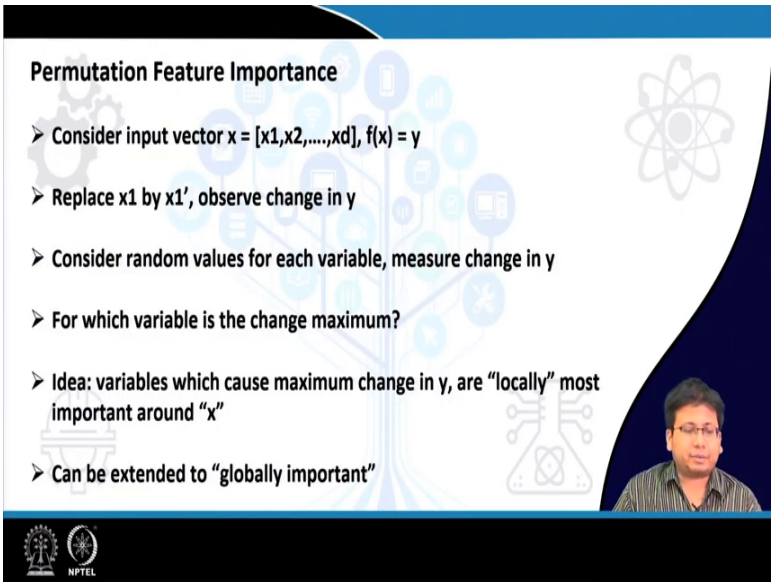
Now this is the what we discussed here is one kind of.

(Refer Slide Time: 30:07)



One method of calculating the or the importance of features and so on.

(Refer Slide Time: 30:12)



Another method of importance of the features is the permutation feature importance. So, let us say we have a feature vector x with these particular like it is a d-dimensional feature vector with these values which produces the output of y. Now suppose I replace one of these values let us say the first feature its value is its current value is x_1. Let us say I change it to some other value I make it $x_1^{'}$. Then the y should change, but by how much I measured that change.

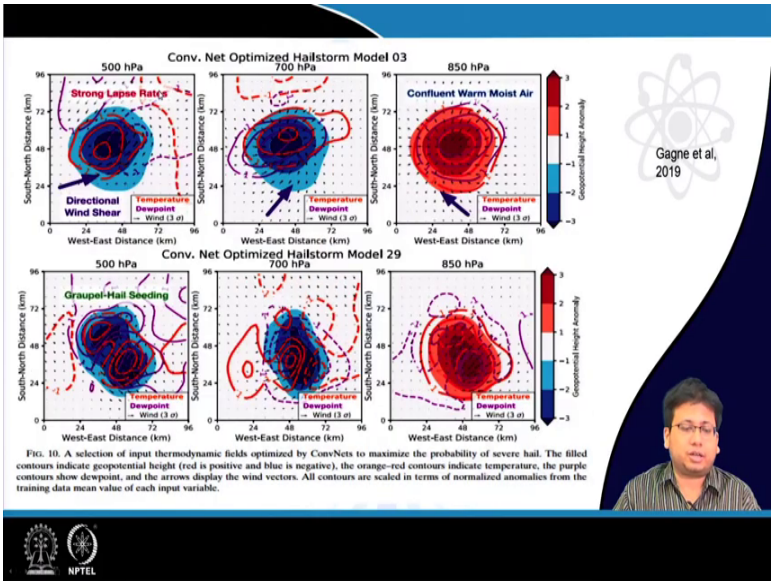
And then I do this kind of thing for all variables and we and I see that changing which variable has the maximum impact on the output variable y. And in fact, like changing by how much has how much effect on the output variable y. So, the idea here is that the variables which cause maximum change in y, they are the most important variables at least locally.

That means, for these values of the other variables these feature is most important for some other values of the other variables some other feature might be most important. So, so like its we are basically looking for the locally most important features in this situation.

So, that is the permutation feature importance. So, like in this paper what they have tried to do is the probability of severe hailstorms that is like suppose a thunderstorm is going to take place. So, they are calculating what is the probability that hailstorms will occur.

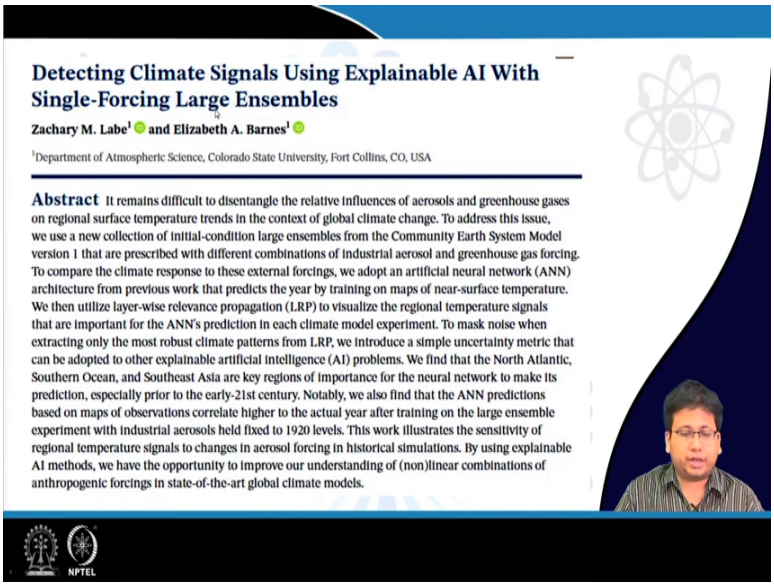
So, these are the inputs like the geopotential height, temperature, dewpoint, zonal, wind meridional, wind etcetera all these are inputs to the neural network based on that the neural network like estimates the probability that there is going to be a hailstorm. So, so fine the neural network is trained and it let us say it performs quite well and as has been found out by this study. It performs better than the other methods like logistic regression etcetera, but then the question is why did it predict so?

(Refer Slide Time: 32:26)



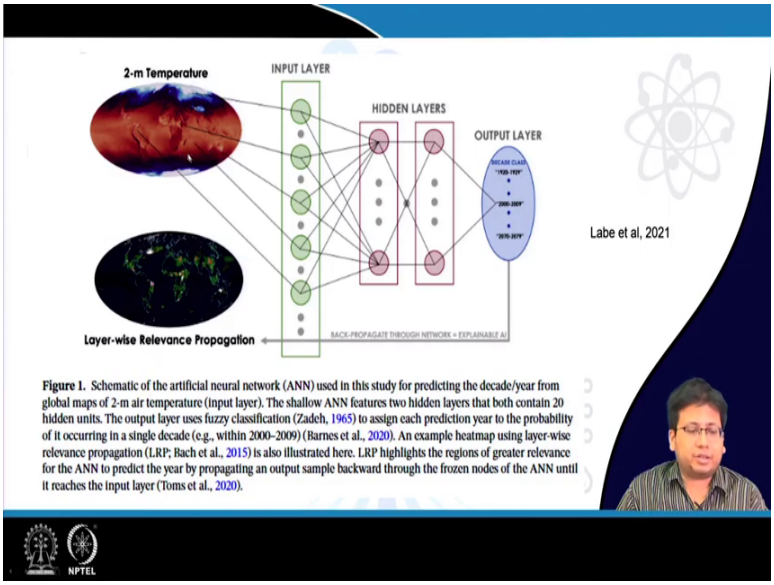
So, the authors of this paper they have used permutation feature importance to find out like really which are the like which are the features that resulted in the prediction of these things. So, like for each of the variables like which of the for each of the input variables that were considered, they are like the their relevance values or rather their this permutation feature importance values are calculated and.

(Refer Slide Time: 32:54)



And similarly like we have some other interesting applications of these things. So, like detecting climate change signals using explainable AI with single forcing large ensembles.

(Refer Slide Time: 33:07)



So, here the basically the idea is that you, so we all know that the like the temperature across the world is changing due to global warming and climate change. Now if you are provided the and this like and we know that the earth has been gradually warming up since the 1900s and so on.

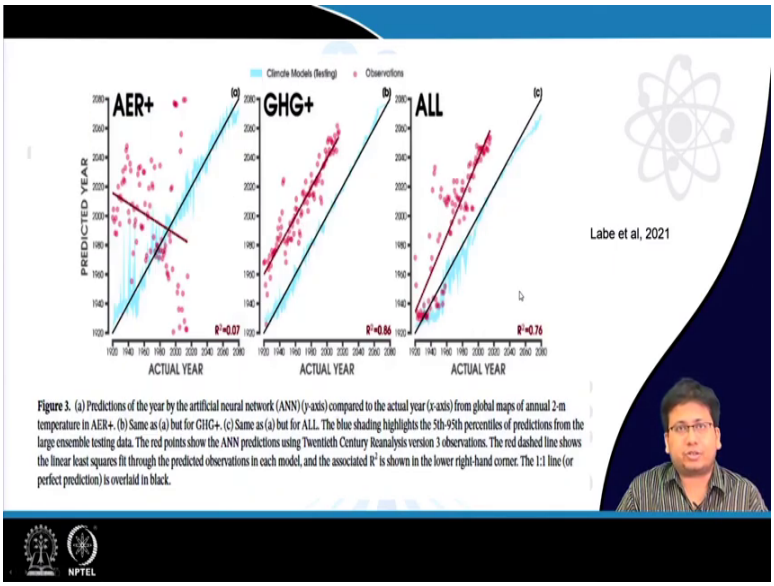
So now if you are given the temperature map of the world then from that map can you can a neural network predict which year that map belongs to. And if so if it can indeed predict then which are the most important regions to which will be looking at. That is in a sense it is like asking which are the regions which have contributed maximum to the global warming from 1900s to say the current times and so on.

So, that is what is achieved by this kind of task. So, like it is found that the they have trained a neural network in which the instead of focusing or predicting the exact year they predict some the neural network can predict some year range like which decade, let us say the 1920s decade or 1930s decade and so on. So, it turns out that this neural network is that is once it is it receives spatial map of the temperature over the world.

It is able to predict the decade from which this map was taken it is able to predict that quite accurately, but then we do the layerwise relevance propagation etcetera to find out which are the locations having maximum relevance. So, it turns out that the like as you can see that the regions which are marked in white they are the maximum like they have the maximum relevance.

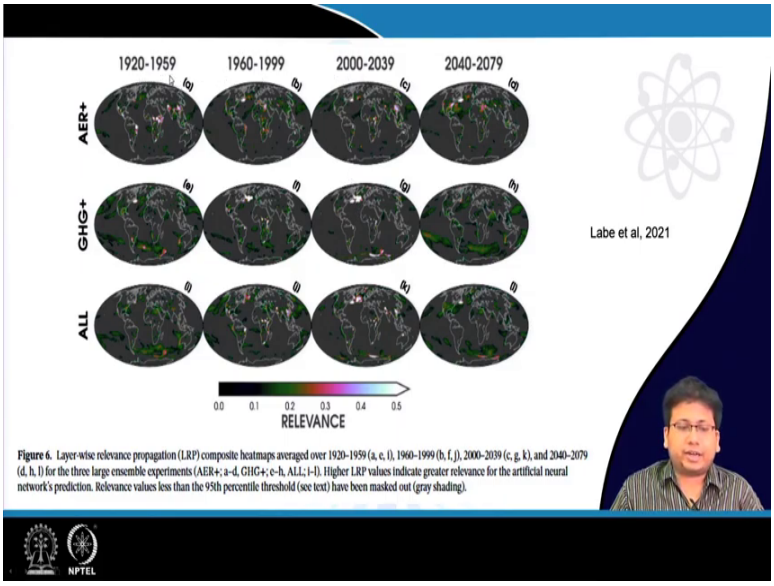
That is to say from let us say from 1900s to the current times it is these regions which probably have had the most significant changes and hence by looking at its by looking at them that we can understand most clearly which year it is or which range of year it is and.

(Refer Slide Time: 35:11)



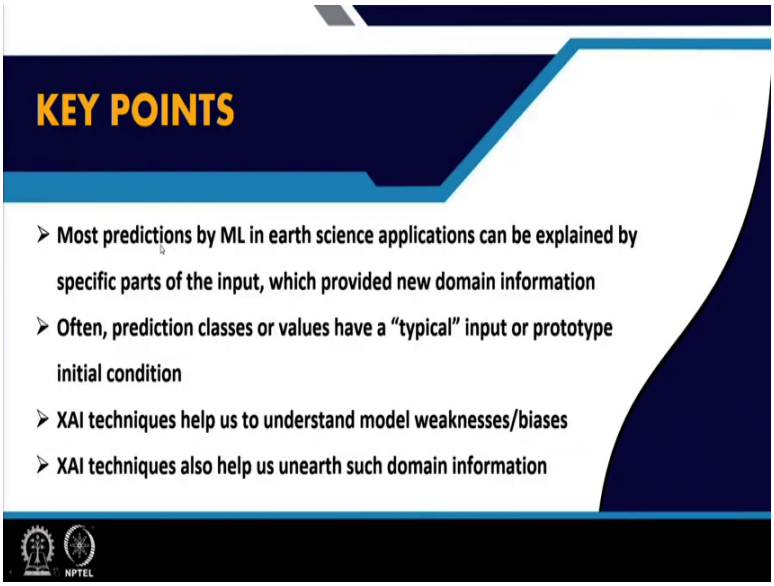
So, like this kind of study can be done for different variables I mean not only for temperature, but let us say also from the aerosol distribution all over the world or the concentration of different greenhouse gases from all over the world etcetera.

(Refer Slide Time: 35:31)



And of course, for different different variables we see different amounts.

(Refer Slide Time: 35:34)



So, that brings us to the end of this module. So, the key, so these are the different papers that we discussed today in today's lecture. The key points to be taken from this from this lecture are first of all most predictions by machine learning in earth science applications; they can be explained by only specific parts of the input and this which provides new domain information.

Like in this case only certain it is only certain locations which carry most of the information behind the prediction like actually like for the different year groups in this case in the last problem that we mentioned.

So, these are the for each set of year say for the 1920 to 1959 region decade its all its really this is what the optimal maps of or the or the optimized inputs for the aerosol map for the greenhouse gas maps etcetera look like. Similarly, for the 2000 to 2039 year this is what the optimized maps look like.

(Refer Slide Time: 36:48)

So, these actually give us the domain knowledge in the sense like how these maps have changed over time and like which are the like which of these maps is the most is the distinctive characteristic of this particular decade.

So, often the prediction classes or values they have a typical or typical input which is what we are calculating by the process of backward optimization. So, we also these explainable AI techniques help us to understand the weaknesses and the biases of the difference models and also help us to unearth such domain information.

So, with that we come to the end of module 3. From the next lecture onwards we will move to module 4 where we will see how machine learning can help in the earth observation systems. So, till then bye.