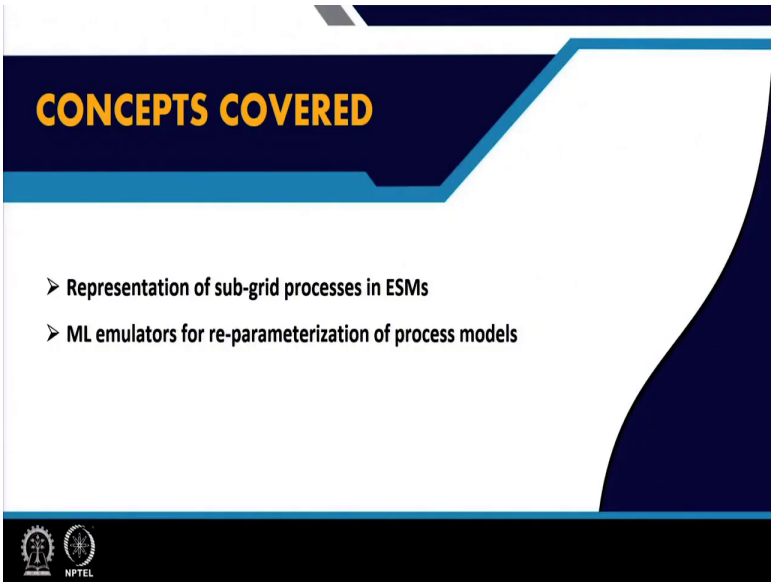
Machine Learning for Earth System Sciences
Prof. Adway Mitra
Department of Computer Science and Engineering
Centre of Excellence in Artificial Intelligence
Indian Institute of Technology, Kharagpur

Module - 05
Machine Learning for Earth System Modelling
Lecture - 38
Parameterizations for Sub-Grid Processes Using ML

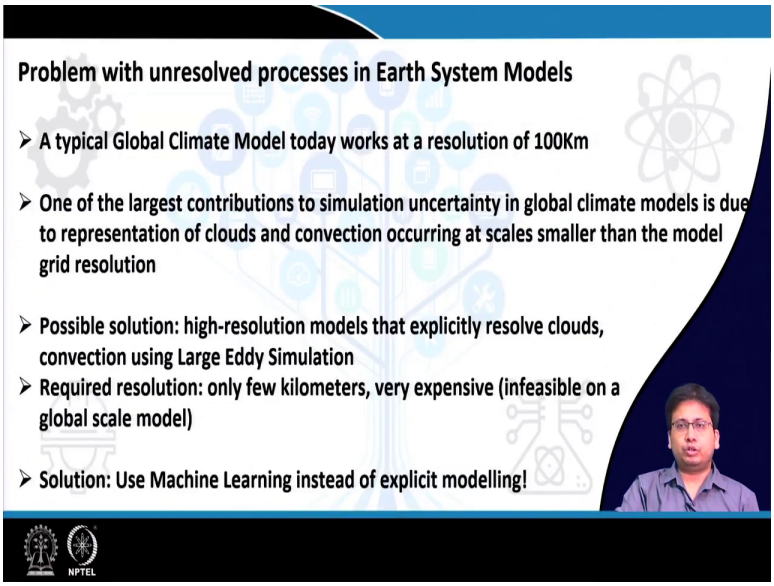
Hello everyone, welcome to lecture 38 of this course on Machine Learning for Earth System Science. We are in the last module called module 5, where we are dealing with how machine learning can help in earth system modeling. The topic of this lecture is Parameterizations for Sub-Grid Processes Using Machine Learning.

(Refer Slide Time: 00:43)



So, we will basically discuss what are these sub-grid processes, how they are like modeled in earth system models and how machine Learning based emulators for can be helped in like re-parameterization of these sub-grid processes in the different process models. So, machine learning emulations we have been dealing with for the past 2 lectures and so we will see a concrete application of why this emulation can be useful. In the like in Earth System Modeling.

(Refer Slide Time: 01:12)



Now, when we are developing earth system models there are like the we have a major problem with what is known as unresolved processes. Now, a typical global climate model works at a resolution of 100 kilometers which can be considered as reasonably coarse resolution. But where it is often found that if we like these kinds of global models are developed by different research groups.

Now if we run these models the models from the different research groups or even if we run the model from the same group under slightly different initial conditions, we see like very different results coming out of it. I mean certain broad trends might be preserved, but as far as details are concerned we see lots of uncertainties. Now why is this happening? That is because like these models they work at a certain resolution as already mentioned.

But there are certain like all of all earth the earth system also involves various localized processes which occur at a lower or at a lower scale than the resolution of these models. So, a like these most of these are related to clouds and convections, because these are things which these are processes that occur typically in a few kilometers. So, model which works at the resolution of 100 kilometer is not able to handle these specifically. So, these are called as the unresolved processes.

So, the problem is these kinds of unresolved processes are like they are handled in the different models in different ways usually as some kind of parameters which take these into account. But the parameters are of course different in the in different models or and so on. Besides the like if we can just these processes they have their own complexities they are time-varying and so on.

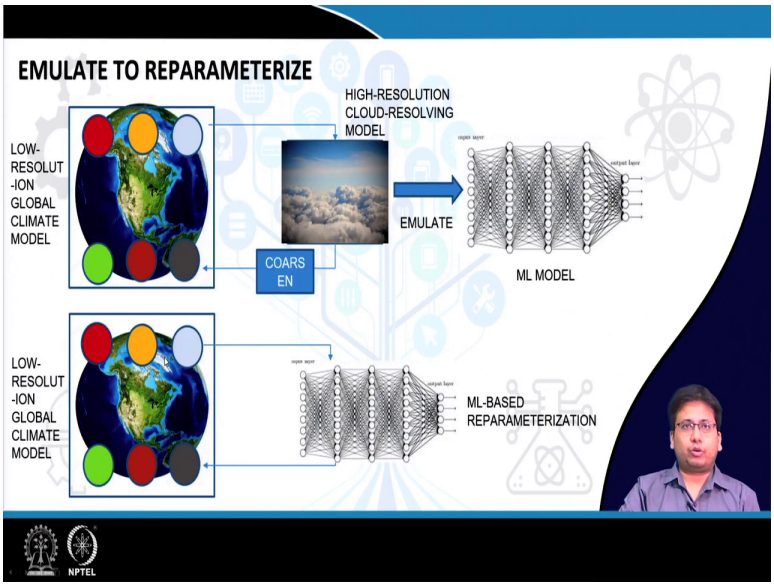
So, just replacing them by a single parameter is unlikely to be very effective, that is why these kind of like we see some various biases in the simulations, that is they the bias usually means some kind of losses of the spatio temporal variability. Like it like the even if the mean value of a particular variable is produced well by these models, they are unable most often they are unable to capture the variance; that is because of the presence of such because they do not take into account the these unresolved processes.

So, how to solve the problem? The solution the obvious solution is of course, to build high resolution models that explicitly resolve these processes like the clouds. So, there are cloud simulation models which are like which the scientist of which of fluid dynamics they deal with. So, there is something known as large eddy simulation numerical models, where the like formation of clouds and how they the clouds resulting precipitation all these things are explicitly simulated.

But these are like very expensive and like because they require like because their resolution is only a few kilometers. So, like for a 100 kilometer by 100 kilometer region we have to develop lots of these kinds of simulations each of which is very expensive. So, like if we have to do the do the same process at a global scale that is for every say 5 to 10 kilometer or 5 kilometer by 5 kilometer region on the earth surface we need to run a large eddy simulation parallelly that is going to take like so much computational resources that no one can afford that.

So, the solution is to somehow use machine learning instead of explicitly modeling these kinds of cloud operations.

(Refer Slide Time: 05:08)



So, the broad idea is like this. So, suppose we have a low resolution global climate model. So, these are some of the different variables. So, let us say that these are some variables which influence the clouds and these are some other variables which are impacted by the clouds.

Say for example, the temperature over the sea is result in convection and which in turn helps in the formation of clouds and let us say these variables the amount of precipitation or the ground water etcetera these are the results of clouds and or the outgoing long wave radiation. So, all these variables so the input variables and which help in the or which influence formation of clouds and the output variables which are influenced by the clouds. So, these are all part of the low resolution model.

So, these what happen like what can be done is these input variables from the we as simulated by the low resolution earth system models can be fed as inputs to a high resolution cloud resolving model. So, it will like simulate the cloud processes at very high resolution. Now that will be coarsened and the outputs of that will be I mean the cloud the cloud simulations are very high resolution that will be coarsened and presented to the earth system models.

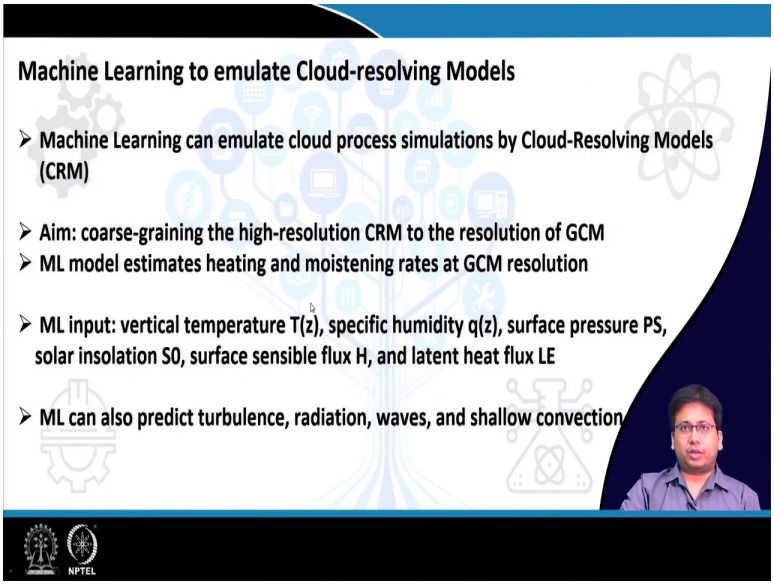
So, that the variables which are the effects they can be updated accordingly. Now the problem here is the is this part the large the cloud simulation which is done through some kind of large

eddy simulation and the further the coarsen. So, these are the things which are causing the computational bottleneck.

So, the approach is going to be use machine learning to emulate this cloud the cloud simulation models. So, we have seen in the previous lecture that machine learnings models can emulate various kinds of physics based processes process models, in this case also we will produce the like we will use it. So, that the whole thing becomes like this the input variables are or the cloud causing variables will be produced provided as inputs to the machine learning model.

The machine learning model will directly simulate the or the predict the values of the output variables and those predictions will be fed back to the low resolution global simulation model. So, as to do away with the need of explicit simulation and coarsening of the of the cloud process.

(Refer Slide Time: 07:43)



So, the machine learning it has been seen in research can emulate the cloud process simulations which are obtained by these CRM or the cloud resolving models. So, the aim is the coarse graining of high resolution CRM to the resolution of the GCM the low resolution 100 kilometer by 100 kilometer at which the typical GCM works.

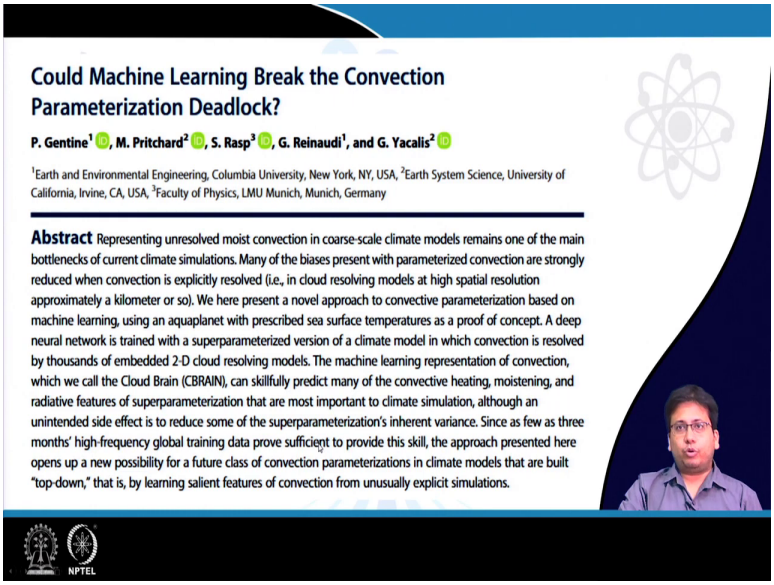
And the so the like there are certain research papers which where ML models are used for estimating the different effects of the clouds, which include heating and moistening rates. And

these heating and moistening rates as emulated by the ML model they are fed back to the low resolution GCM for further processing.

So, the inputs to these models is the like either the cloud resolving model or the emulating machine learning model these are the input variables are the vertical temperature, the specific humidity, surface pressure, solar insulation, surface sensible flux, latent heat etcetera. All of these help in determining the heating and moistening rates by convection, Like I like in case of these cloud resolving models such as the large eddy simulation and so on.

So, the same variables are also provided as inputs to the machine learning model. Now and this machine learning model provides it is using these inputs it provides this or it estimates this heating and moistening rates, it can also predict the other things like turbulence radiation waves shallow convection etcetera.

(Refer Slide Time: 09:20)



So, there is a bunch of papers on this matter. So, the first paper here the name is quite interesting here could machine learning break the convection parameterization deadlock. So, this paper appeared in 2018 when the researchers were just exploring this the possibility of using ML to emulate these CRMs. So, representing unresolved moist convection in coarse scale climate models remains one of the main bottlenecks of current climate simulations.

Many of the biases present with parameterized convection are strongly reduced when convection is explicitly resolved that is in cloud resolving models at high spatial resolution approximately a kilometer or so. We here present a novel approach to convective parameterization based on machine learning using an aquaplanet with prescribed sea surface temperature as a proof of concept.

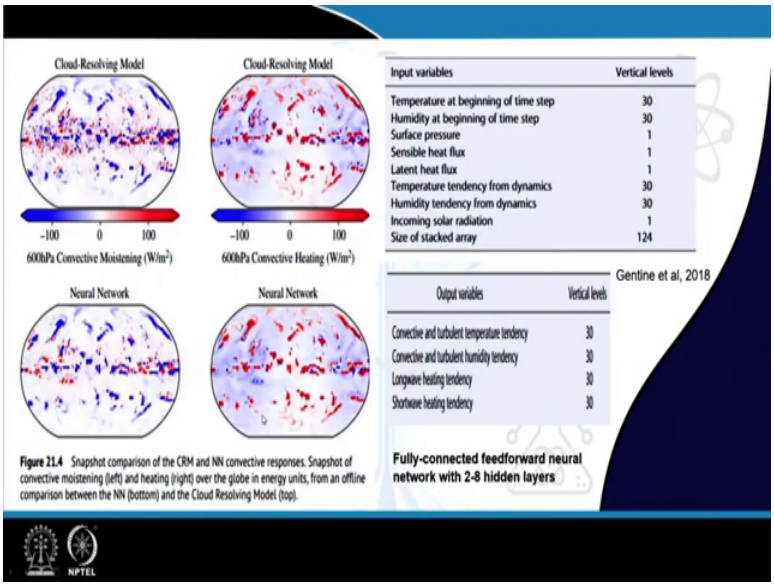
So, this aquaplanet it is basically I is a simplified model for the earth which is used in various earth system models. A deep neural network is trained with a super-parameterized version of a climate model. The super-parameterized version of a climate model means this like basically this the like this the this is the climate model and including this CRMA that is known as the super parameterized model.

So, the a deep neural network is trained with a super parameterized version of a climate model, in which the convection is resolved by thousands of embedded 2D clouds resolving models just as I mentioned earlier. The machine learning representation of convection which we call a cloud brain can skillfully predict many of the convective heating moistening and radiative features of super parameterization that are most important to climate simulation. Although an unintended side effect is to reduce some of the super parameterization inherent variance.

So, like this also I mentioned earlier the a simulation of the variance is often a big problem in regardless of what simulation methodology we are using. So, the like if we go for like very explicit process-based simulation we might be able to get the actual variance. But if we are going for any kind of like emulation or some less or I mean or typically any kind of statistical simulation.

In fact, the problem is in reproducing the variance since as few as three months high-frequency global training data prove sufficient to provide the skill, the approach presented here opens up a new possibility for a future class of convection parameterizations in climate models that are built top down. That is by learning salient features of convection from unusually explicit simulations.

(Refer Slide Time: 12:23)



So, these are the various input variables which they are providing to the machine learning model for emulating the cloud CRMs. So, the so like as you can see most of these are at different vertical levels that is the convection is of course as you can understand it is not something that happens only on the surface wave. But rather it is related to the evaporation of water from the surface rising there the evaporated water rising up and to a high altitude and forming clouds and so on.

So, like we have so like the vertical profile is of different variables is important for this. So, the input variables include the temperature at different vertical levels, the humidity at different vertical levels, the surface pressure, the sensible heat flux, the latent heat flux these are of course more relevant only for the surface.

The temperature tendency from the dynamic, the humidity tendency from the dynamics, the incoming solar radiation and so on and then of course there are the machine learning based parameters which are with the stack as the how many layers will be stacked and so on.

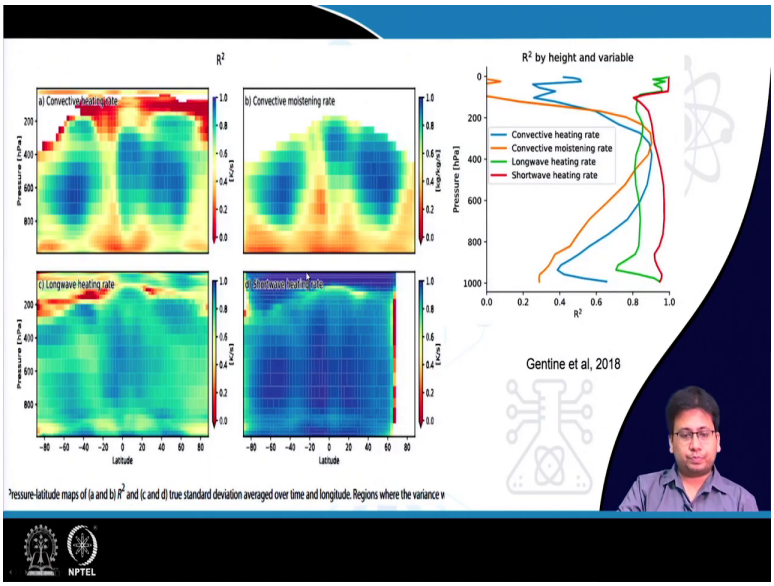
The outputs are going to be the convective and turbulent temperature tendency at all the 30 levels or 30 vertical levels; similarly the convective and turbulent humidity tendencies the long wave heating as well as the short wave heating all at the all at all vertical levels. So, like here like for

convective moistening at a given at 600 hPa means like at the at a. So, this basically specifies a particular altitude.

So, we know that hPa hectopascal is the unit of pressure. So, we know that pressure decreases as we go higher and higher up. So, 600 hPa is that altitude at which this kind of this pressure this particular pressure is achieved. So, at that particular altitude if you look at the convective moistening rate and as well as the convective heating rate, like we can like what we get something from the cloud resolving model and we get something from the neural network the emulating network.

So, here in both cases both for moistening and for heating this is the map. So, like this is basically the worlds map, worlds map at that particular altitude. So, here you can see the values produced by the CRM and here we you see the corresponding values emulated by the neural network. So, as you can as it appears like these 2 maps are near identical, the same thing for the convective heating also.

(Refer Slide Time: 15:09)



And similarly in case of like if you consider the vertical profile at different. So, like in this case the so here we can see here we are focusing on a particular altitude, but we are capturing

considering all latitudes and longitudes. Of course, the whole thing is 3 dimensional, so we are we cannot of course on the on the screen we cannot build a 3 dimensional map.

So, we are building 2 dimension by either keeping 1 dimension as fixed or by averaging over that dimension. So, in this case we are keeping the altitude dimension as fixed, in this case we are keeping like the longitude we are actually averaging over all the longitudes and also over time. We are just plotting latitude versus altitude and instead of like measuring altitude directly in terms of meters or something we are like measuring it in terms of the corresponding pressure.

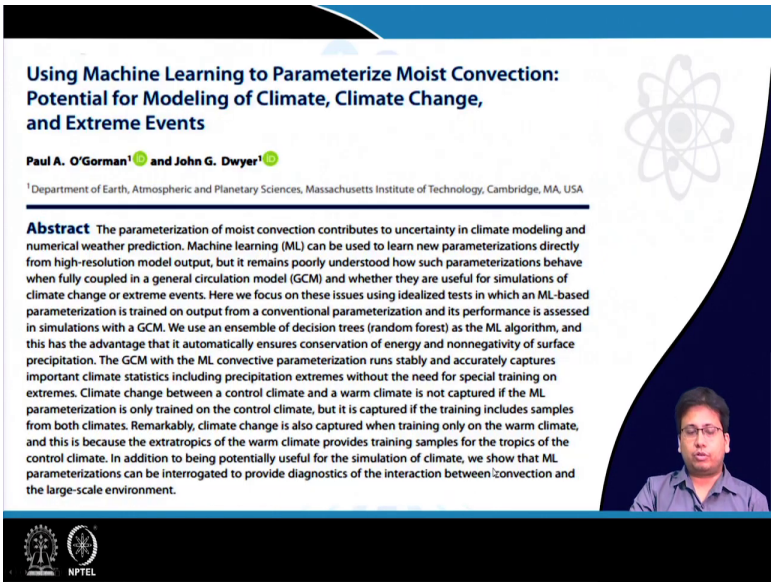
So, at every location which is a combination of latitude and altitude like we are again studying this convective heating rate and the convective moistening rate as we did here also. And so like these are the like these are not the actual values, but these are the what is known as the R^2 and the standard deviation. So, R^2 is basically it is like 1 minus the squared error divided by the standard deviation.

So, like basically the smaller this value is the better we can say the like so sorry the that is if the a the closer to 1 this value is we can say the better the simulation has been. So, like we can see that in all these places the value of this R^2 is quite close to 1 and which indicates that a like for the most part the emulation has been quite successful.

And like similarly we can also study the like the variation of R^2 over a different altitude levels for the different variables that are being emulated. So, we can see especially for this short wave heating rate the R^2 is always for all altitudes it is right from the ground to the top of the atmosphere it is like close to 1; which means that this is like in this case the simulation has been best.

In this case for the convective moistening rate unfortunately we see the R^2 value is actually quite poor near the surface, it improves a bit like as we go higher. But then beyond a particular altitude this again becomes bad, the others other 2 are like we can see in between.

(Refer Slide Time: 17:58)



So, this is one paper now another paper on roughly the same topic using Machine Learning to Parameterize Moist Convection Potential for Modeling of Climate, Climate Change and Extreme Events. So, in this case the model they have used is different in this case they have used random forest instead of neural network. By the way in this case the model they used with though I did not specify, it is basically a fully connected feed forward neural network with 2 to 8 hidden layers that is they have considered different versions.

So, one is a like a shallow neural network with only 2 hidden layers another is a deep 1 with 8 hidden layers, but it is fully connected no convolution or things. Like that the in so in this case the parameterization of moist convection contributes to uncertainty in climate modeling and numerical weather prediction. Machine learning can be used to learn new parameterizations directly from high resolution model output.

But it remains poorly understood how such parameterizations behave when fully coupled in a GCM and whether they are useful for simulations for climate change or extreme events. So, we have seen that they can emulate the these CRMs quite well, but how about say the climate change I mean we know that there is a secular increase of temperature over the over the years irrespective of local processes.

Then apart from that there is also there are also extreme events. So, how are these extreme events and the general climate change, how they are modeled how these can be in like estimate or how these can be captured if we go for emulation of this of these GCMs by the. I mean if we try to super parameterize these GCMs and then try to emulate them using the machine learning.

So, here we focus on these issues using idealized tests in which an ML based parameterization is trained on output from a conventional parameterization and its performance is assessed in simulations with a GCM. We use an ensemble of decision trees or random for as the ML algorithm and this has the advantage that it automatically ensures conservation of energy and non negativity of surface precipitation.

The GCM that so like this conservation of energy and non negativity of surface precipitation these are of course, physical like physical constraints which follow from physics. So, if we use the physics based neural network these are constraints which you would enforce on the values that are being predicted. So say for example the predicted value of rainfall can never be negative that is some a constraint which we would have to enforce if you are using some kind of using the PINN concept.

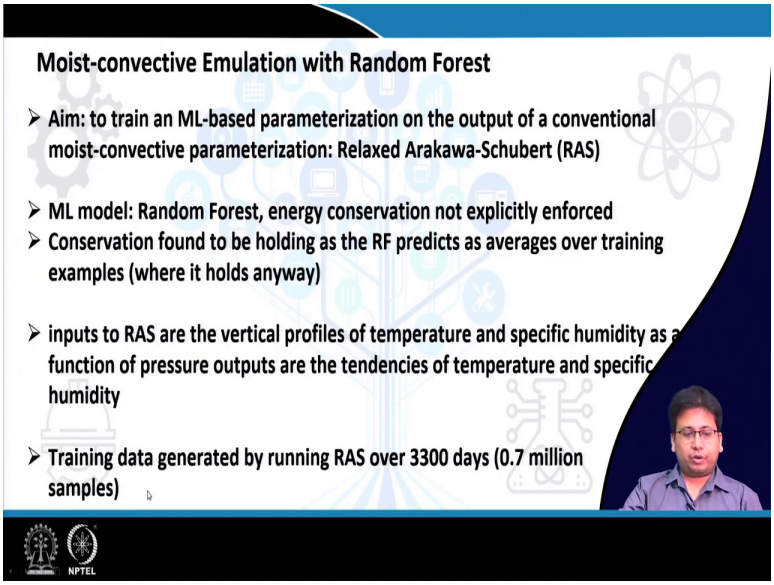
But in this case they are using random forest and like they are although they are saying that they these are not these are constraints are not explicitly imposed, but because the random forest predicts by taking averages over the training set. So, if in the as in the training set this constraint are satisfied anyway.

So, the idea is like when it makes a new prediction also this I will continue to be satisfied. The GCM with ML convective parameterization runs stably and accurately captures important climate statistics including precipitation extremes without the need for special training on extremes. Climate change between a control variable, control climate and the warm climate is not captured if the ML parameterization is only trained on controlled climate. But it is captured if the training includes samples from both climates.

Remarkably climate change is also captured by training only on warm climate and this is because the extratropics of the warm climate provides training samples for the tropics of the control climate. In addition to being potentially useful for the simulation of climate we show that ML

parameterizations can be interrogated to provide diagnostics of the interaction between convection and the large-scale environment.

(Refer Slide Time: 22:22)

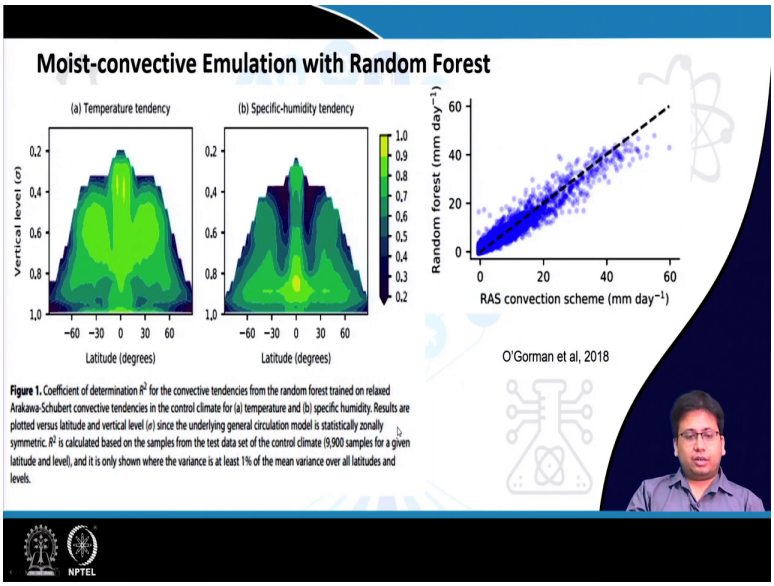


So, the aim, in this case, is to train an ML-based parameterization on the output of a like conventional or a physics-based moist convective parameterization model which they are called in which is called a RAS. The ML model in this case is the random forest and as I mentioned already the energy conservation or the non-negativity of precipitation these are not explicitly enforced.

But rather they are because it is a random forest which predicts by taking averages over the training set, these are considered to be holding anyway. The inputs to the RAS are the vertical profiles of temperature and specific humidity as a function of pressure output and the as well as the tendencies of temperature and specific humidity.

So, the so these are the inputs to the RASthe physics-based the with the physics-based model which we are planning to emulate using random forest. So, the training data is generated by running this RAS over three 3300 days that is roughly 9 years or so and this way some 0.7 million examples are generated.

(Refer Slide Time: 23:37)



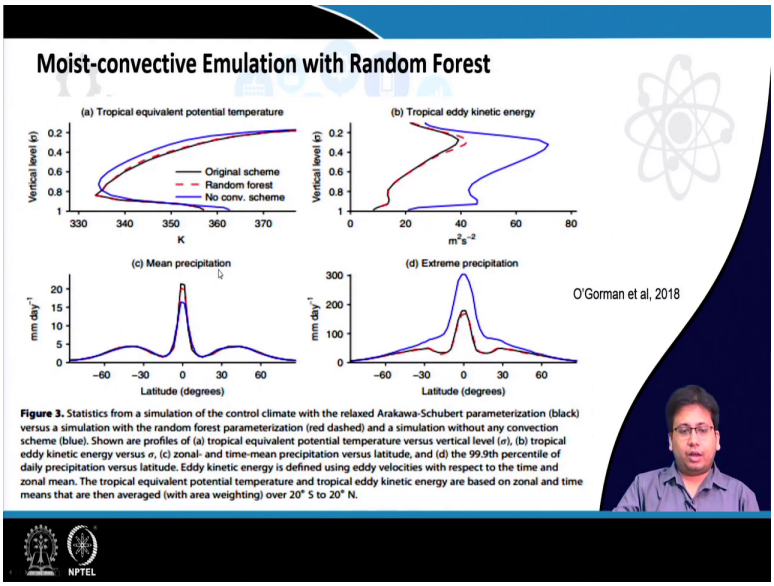
So, based on that the Random forest is trained, so like if you consider the different variables as simulated by the process based model as well as by the random forest. So, here you can see this is here we are considering the precipitation. So, the which is in measured in millimeters per day. So, here you can see that there is a near perfect agreement between what is obtained from the physics-based model and what is obtained from the ML model.

So, the daily precipitation is predicted with like almost perfectly. Now if you I mean this now if you consider the temperature tendency and the specific humidity tendency again we go for the latitude versus altitude plot as discussed earlier. So, like a like in this case also like we at the different latitude comma altitude values we compare the R^2 between of what is of the values that are predicted by the random forest model.

And like a high value of R^2 indicates that the error is nearly is nearly equal to 0 and as we can see for large parts of the of this profile we can see the R^2 values close to 1 in other parts also it is pretty high like above 0.7 so on. For both the temperature as well as specific humidity and it seems a like only at certain places very high or near the polar region. So, these are the place this is like we can say the South Pole region where the latitude is 90 degree South or here this is the North pole where latitude is 90 degree North.

So, these places the thing these things have not been captured properly, but then these can be considered as like these are outlier cases. But apart from that we can see for most of the like the 3D structure the tendency seem to be like the tendency as simulated by the RF model seems to be almost agreeing with what is obtained from the RAS model.

(Refer Slide Time: 25:48)



And these are like the other the other things or the other quantities that they have been simulated like for example Extreme precipitation. So, here the aim is that this can this RF model they it can predict the extreme values of precipitation quite well, even if it is not specifically trained to do so and here they are actually showing that.

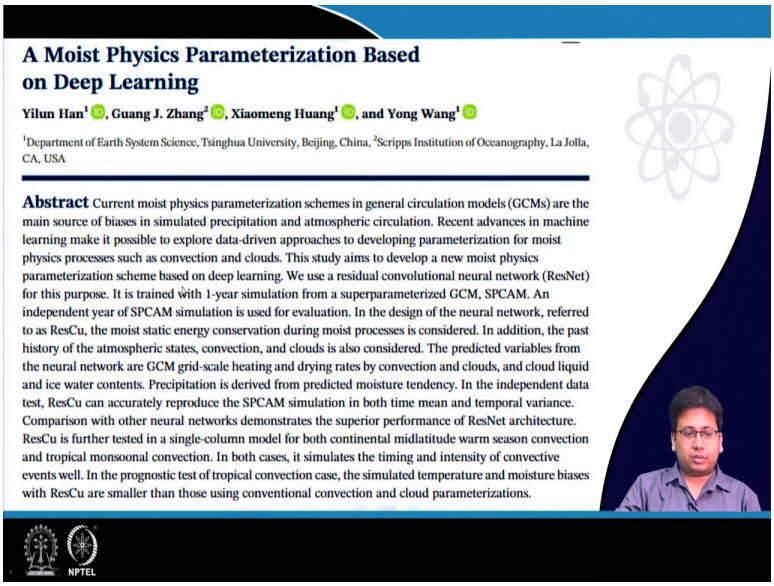
So, like so here the black is the what is the original scheme and the this red dash line is the random forest. So, here you can see that the original scheme that is RAS and the random forest these are like matching almost perfectly. So, here it is like the extreme values at the different latitude; by extreme we may mean the block maxima or something like that, like we already know the definitions of the different definitions of extreme.

So, like at different altitude sorry at different latitude, what is the maximum temperature I mean the maximum precipitation observed over this period. So, we see that it is almost matching in both cases similarly for mean precipitation also we see that they are all matching. The interesting

thing is the this kind of symmetric structure which shows the precipitation is maximum around the equator, it drops off and the 2 sides arises again near the tropics and then in the extra tropics it gradually falls off in both hemispheres.

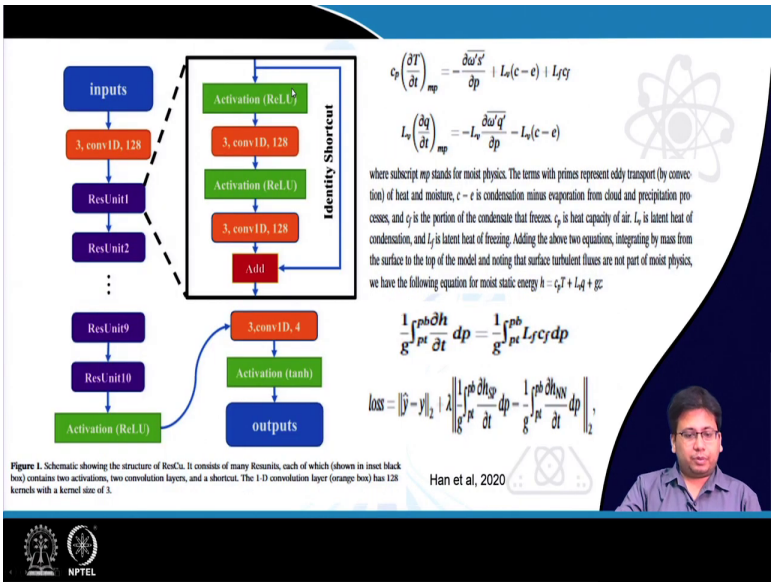
And this effect is captured almost equally by both the RAS as well as the random forest based thing.

(Refer Slide Time: 27:26)



Like similarly we have one more paper on roughly the same topic a moist physics parameterization based on deep learning.

(Refer Slide Time: 27:35)



So, here they have built a more sophisticated model like based on Resnets and so on. So, the current moist physics parameterization schemes in GCMs are the main source of biases in simulated precipitation and atmospheric circulation. Recent advances in machine learning make it possible to explore data-driven approaches to developing parameterization for moist physics processes, such as convection and clouds. This study aims to develop a new moist physics parameterization scheme based on deep learning.

We use a residual convolutional neural network or Resnet for this purpose, it is trained with one year simulation from a super parameterized GCM called SPCAM. This SPCAM is also the one which was discussed in the first paper, an independent year of SPCAM simulation is used for evaluation. In the design of neural networks referred to as ResCu the moist static energy conservation during the moist processes is considered.

In addition the past history of the atmospheric states convections and clouds these are also considered the predicted variables from the neural network are GCM grid scale heating and drying rates by convection and clouds and the cloud liquid and ice water contents. So, these are things that are predicted by the model by both the SPCAM as well as from the neural network. The precipitation is derived from predicted moisture tendency in the independent data test ResCu can accurately reproduce the SPCAM simulation in both time mean and temporal variance.

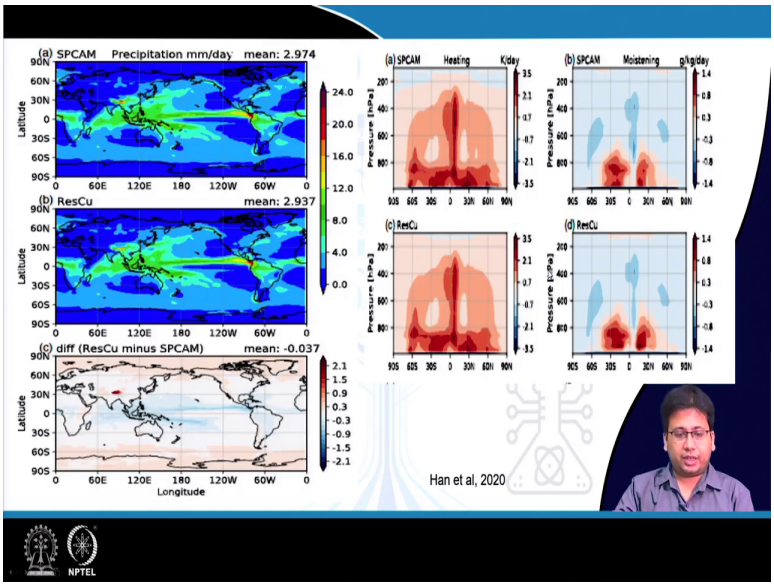
The comparison with other neural networks demonstrates the superior performance of Resnet architecture an so on. So, this is like the machine learning model. So, like as you can see it is called ResCu because it has so many of these of a residual units and then there are this. So, the each residual unit basically looks like this. So, it is a it goes through a like a series of convolutions but this identity mapping also.

So, there this skip each residual unit has this kind of skip connection, which is added to the simulation to the convolution result and this same process is repeated 10 times. So, these are some of the like the equations of the physics-based model which are to be which is to be emulated. So, in this case going back to the like PINN approach. So, like in this case you can see the loss function has been designed to actually see how much the or to what degree this equation is satisfied.

So, in like this becomes more difficult because to evaluate the loss function one has to do the integration also and to do the integration like it is not straight forward. So, like basically this derivative of the different things they have to be calculated as a function of time and then the integration over the different pressure levels. So, dp means these are the pressure levels from one level to another.

So, that is evaluating this loss function is a bit of a trouble, because so much of this automate differentiation and integration needs to be done that is one challenge challenging This model.

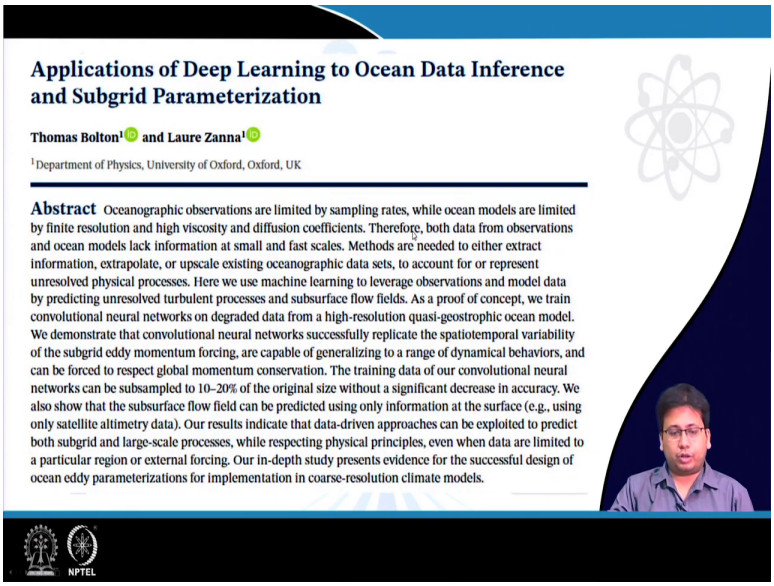
(Refer Slide Time: 31:09)



But the results are encouraging. So, here as you can see the precipitation in millimeter per day is like the or the world map of precipitation in as obtained from the SPCAM as and as simulated by the ResCu is are almost nearly identical. So, this is the difference map between the 2 things and as you can see the differences are almost minimal. Except for some small region here near the Himalayas or just in the Tibetan region I am not sure what is the reason for this problem here.

But apart from that simulation seems to be almost perfect. Similarly if you again consider those latitude altitude plots which we are talking about earlier, that is the heating and moistening tendencies. Once again we see like reasonable agreement between what is obtained from the SPCAM and what is obtained by the simulation by the ResCu model.

(Refer Slide Time: 32:12)

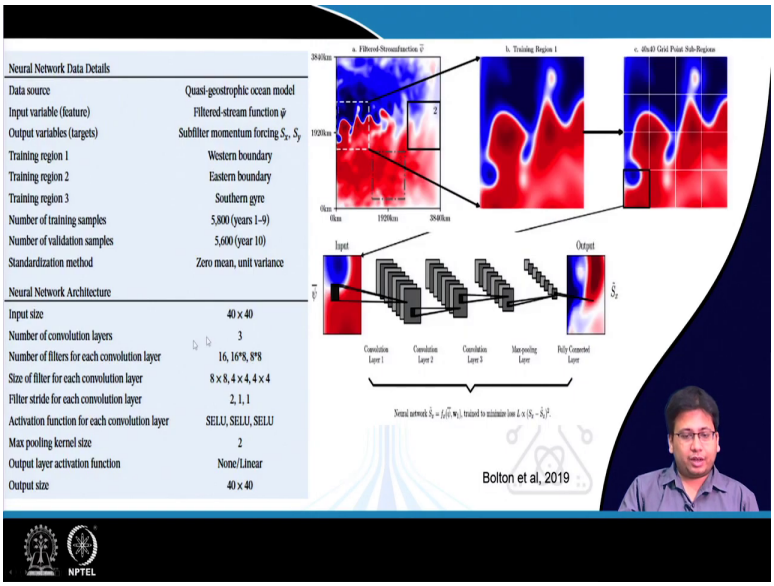


Now, we come to our last application of this topic the Applications of Deep Learning to Ocean Data Inference and Subgrid Parameterization. So, oceanographic observations are limited by sampling rates while ocean models are limited by finite resolution and high viscosity and diffusion coefficients. Therefore, both data from observations and ocean models lack information at small and fast scales, that is neither do we have very good high resolution observations nor do we have such good simulations from models.

Methods are needed to either extract information extrapolate or upscale existing oceanographic data sets to account for or represent the unresolved physical processes. Here we use machine learning to leverage observation and model data by predicting unresolved turbulent processes and subsurface flow fields as a proof of concept, we train convolutional neural networks on degraded data from a high resolution quasi geostrophic ocean model.

You do not know need to know what exactly is that but just assume that there is an ocean model like.

(Refer Slide Time: 33:27)



So, like whose aim here is the that a that is it has a some kind of something known as a streamed function which is calculated by the model at different or which can be estimated at different locations. And based on that the velocities or I mean the wave velocities at different locations and different depths has to be estimated.

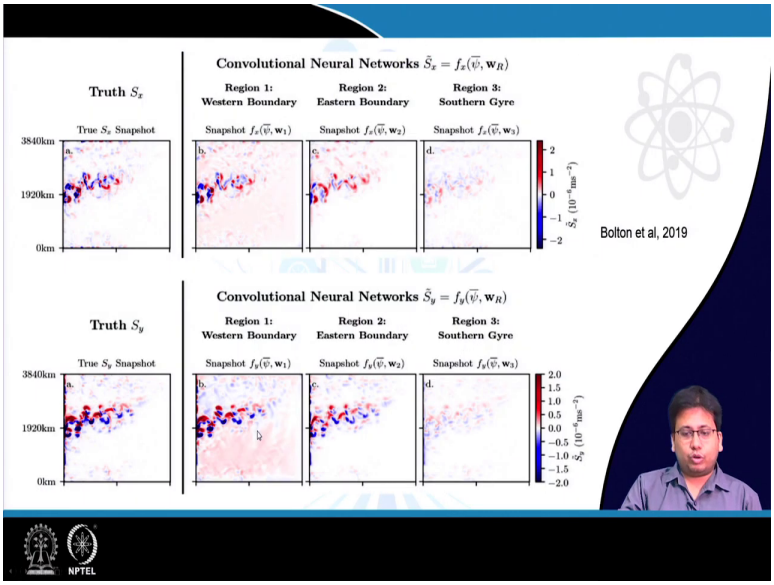
So, there are these physics-based models which like which can do this that is from the that is which take into account the this stream flow function and based on this. So, this stream flow function it can be estimated using various inputs and based on that it can like the that the ocean model can estimate these velocities. But it is at a coarse resolution.

So, what the aim is like we will try to learn a machine learning model which will be able to emulate this the results of this ocean model and then run that machine learning model at like high resolution. So, that we can get a more a high resolution map of the of these velocities.

So, these are the data source in this case is the basically the simulation results obtained from this quasi-geostrophic ocean variable. The input variable for the neural network is going to be the stream function Ψ and the output variables are going to be the like the sub filter momentum S_x and S_y along the 2 like along the x along the horizontal and along the vertical.

So, also like this is trained in one part of the model or in different part of the model and then trace sorry different parts of the ocean and then tested in other parts of the ocean. So, the and so it is trained on 9 years of data and tested on the 10th year and so on and so forth. So, this is the CNN architecture it is a pretty standard CNN using the like say 3 convolutional layers having this kind of specification using the selu. Like selu activation function.

(Refer Slide Time: 35:46)



So, like in the different cases so this is the ground truth of the S_x. So, that is the momentum and by ground truth we mean the simulation from the quasi geostrophic ocean model. So, you can see at different depths the like we have got a map of the horizontal velocity of the ocean, rather the horizontal component of the motion of the momentum of the ocean.

Similarly, we have the S_y also which is the like the if the I mean by S_x we mean the zonal component and by S_y we mean the meridional component. This is the East-West component and this is the north-south component and this is again calculated at different like at different altitudes and also along different locations of the ocean and by altitude again I mean that different depths of the ocean it can be calculated.

So, again so these are the predictions by the neural network, they the 3 things are depends on which region the neural network has been trained. If it is trained in one region we see like this is the map we get in the target region, if it is targeted if the CNA is trained in the second region. Then this is the map as we can see it is like more or less in both cases it is more or less the same. Though if we train the neural network in the third region then the predictions do not match so well.

The similar is the result in case of the S_y the meridional component of the momentum also. So, basically the idea here is that it or the takeaway message here is that it can a convolutional neural network can be used to emulate these kind of ocean models also and as a result get high dimension or high resolution maps for the ocean velocity at different locations.

(Refer Slide Time: 37:31)

And at different depths this paper has provided a proof of concept but hopefully it is possible to build on that. So, these are the different papers that we dealt with today. So, the so like the different paper these all these papers they give one message that like in different domains of earth sciences where parameterization of the process models is a problem.

It is possible to use the machine learning to emulate the processes the like high resolution process models and like the advantage of that is that while running those high resolution process

models can be very expensive. The machine learning can without requiring so much of computation can reproduce roughly the same results and this is true for different kinds of process models.

Like we saw the convective processes as well as the ocean oceanic processes there are other processes also and so this raises the hope that the large-scale earth system models like GCM they can benefit greatly by use like. If they outsource the different sub-grid process processes to machine learning models. So, that brings us to the end of this lecture thank you and in the following lectures we will see a couple of more applications of machine learning in earth system models. So, till then bye.