

**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 - 01**  
**Spatio - Temporal Statistics**  
**Lecture - 10**  
**Data Assimilation**

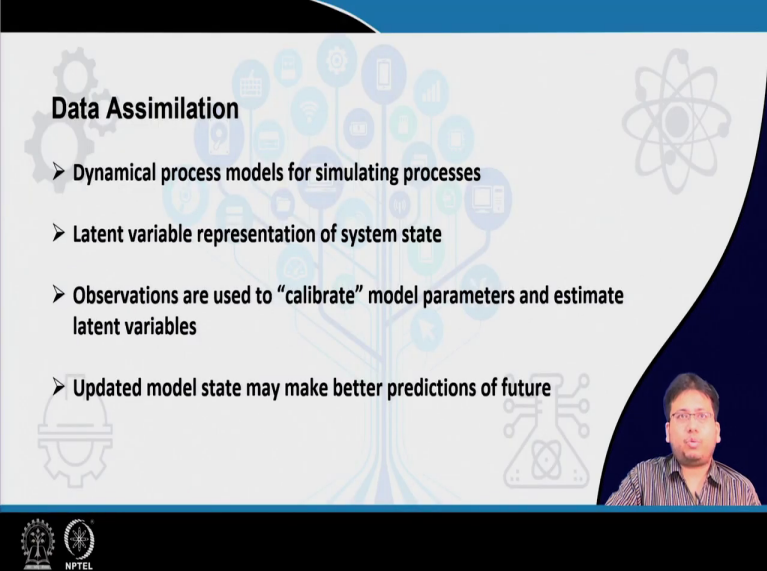
Hello everyone, welcome to lecture 10 of this course on, Machine Learning for Earth System Science. So, we are still in the module 1 of Spatio-Temporal Statistics, but this is going to be the last lecture of this module. So, today we are going to talk about the topic of Data Assimilation.

(Refer Slide Time: 00:43)



Now, what is the concepts we are going to cover today are data assimilation, Kalman filter, and their applications in earth system science.

(Refer Slide Time: 00:50)



**Data Assimilation**

- Dynamical process models for simulating processes
- Latent variable representation of system state
- Observations are used to “calibrate” model parameters and estimate latent variables
- Updated model state may make better predictions of future

The slide features a background with various icons related to data and technology, including a gear, a tree of nodes, a smartphone, a Wi-Fi symbol, a bar chart, a document, a network diagram, and a molecular structure. In the bottom right corner, there is a small video inset showing a man with glasses speaking. The NPTEL logo is visible in the bottom left corner.

So, now what is data assimilation? Now, we have already in some previous lectures when we are defining the Gaussian process and things like that. What we were doing is, we were basically building these kinds of stochastic models for simulation of processes. We had we said at that time that the those models will be used for like simulating or generating artificial data corresponding to the different processes.

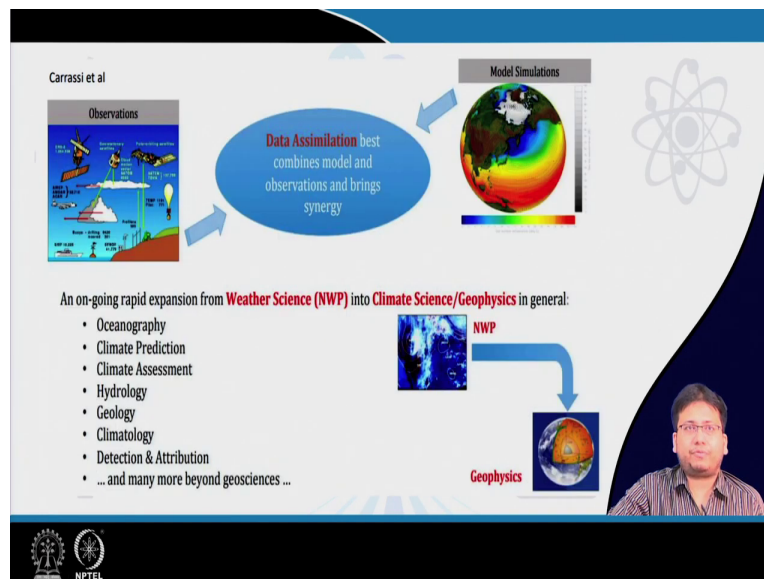
Now, these dynamical process models are there which are based on like latent variables. So, these latent variables we can say they represent the system state in some sense that is like we may have it need not the system state may not be a single variable but it can be like a collection of different latent variables. And the and typically in case of most models what happens is that these latent variables they drive the observations that is what also happened in the stochastic models which we had discussed earlier.

Now, the observations are used to calibrate the model parameters and estimate the latent variables. So, like we had already discussed in briefly about how Monte Carlo Markov chain Monte Carlo gives sampling parameter estimation and such things are used to or to estimate the parameters of these models. But suppose that like there is some ongoing process which we are trying to predict or like one step ahead using this kind of a model. Now, in this case what happens is that the observations they come in sequentially that is we do not have a big set of

observations to begin with. So, that we can like calibrate the model by estimating the parameters accordingly, but the data keeps coming one step at a time.

So, here what we do is we keep like we keep on updating the model as and when the data comes, and the reason why we want to do the update is that; so, that we can make the better predictions of the future. It is also possible that like with the various parameters in the model they may also be time varying, it need not happen that the all the parameters will be fixed. So, like as the more and more data comes we also have to keep on updating those parameters accordingly.

(Refer Slide Time: 03:25)

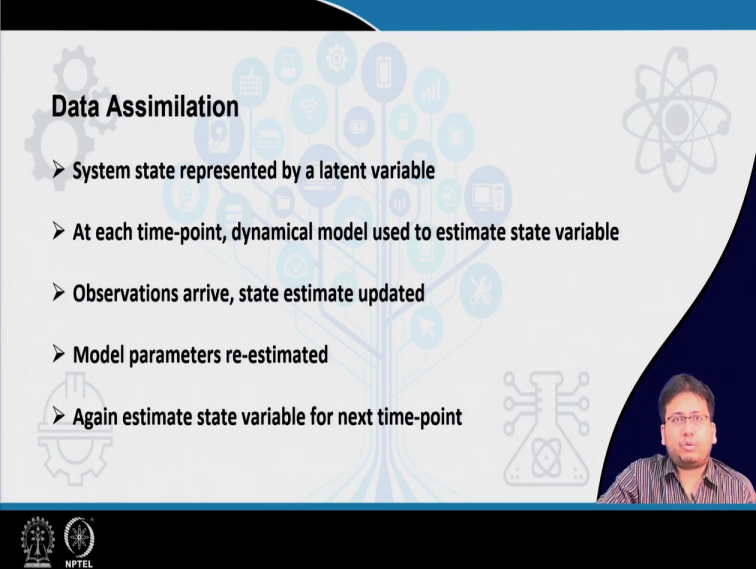


So, that is the basic idea of data assimilation that is data keeps coming and we like assimilate that data into our model to make the our model more realistic and more closely connected to the data. So this is like a slide from Carrassi who has an excellent tutorial on this Kalman filter on data assimilations and Kalman filters.

So, like in the like various domains of earth system science like this concept of this data assimilation is quite heavily used in all these things. So, like the basic thing is that we have model simulations and we have observations. So, the idea here is to somehow like assimilate the observations into the geophysical model simulations.

So, it data assimilation best combines the model and observations and brings synergy between the two. What is the purpose of bringing synergy? The purpose is to make the model more or to like bring the model as close to the data as possible; so, that it can make predictions or forecasts more accurately.

(Refer Slide Time: 04:44)



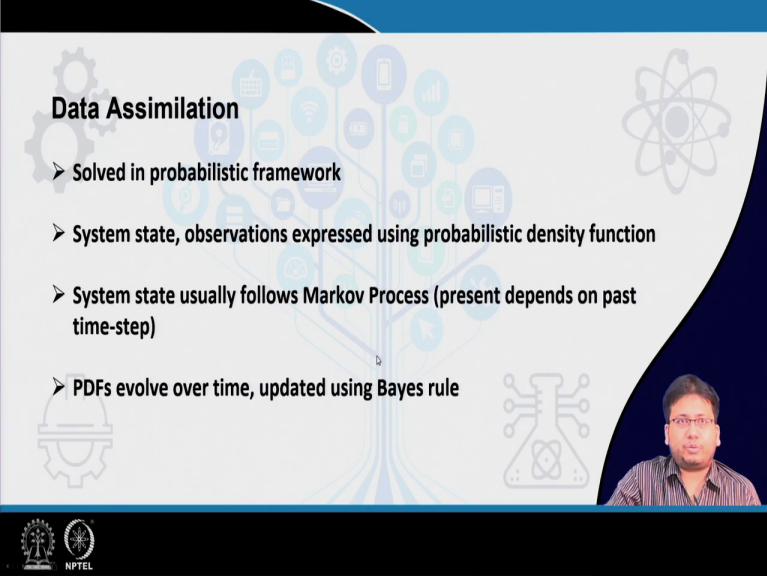
**Data Assimilation**

- System state represented by a latent variable
- At each time-point, dynamical model used to estimate state variable
- Observations arrive, state estimate updated
- Model parameters re-estimated
- Again estimate state variable for next time-point

The slide features a background with various icons representing technology and science, including gears, a tree of nodes, a molecular structure, and a circuit board. A video inset in the bottom right corner shows a man with glasses speaking. The NPTEL logo is visible in the bottom left corner.

So, as I said in case of data assimilation the system state is represented by a latent variable at each time point the dynamical model is used to estimate the state variable. Now, the observations keep arise and the state estimate is also updated the model parameters are reestimated, and then again the whole process repeats in the next times time point.

(Refer Slide Time: 05:07)



**Data Assimilation**

- Solved in probabilistic framework
- System state, observations expressed using probabilistic density function
- System state usually follows Markov Process (present depends on past time-step)
- PDFs evolve over time, updated using Bayes rule

The slide features a background with various icons related to data and technology, including a gear, a tree of nodes, a smartphone, a Wi-Fi symbol, a document, a laptop, and a network diagram. A small video inset in the bottom right corner shows a man with glasses speaking. The NPTEL logo is visible in the bottom left corner.

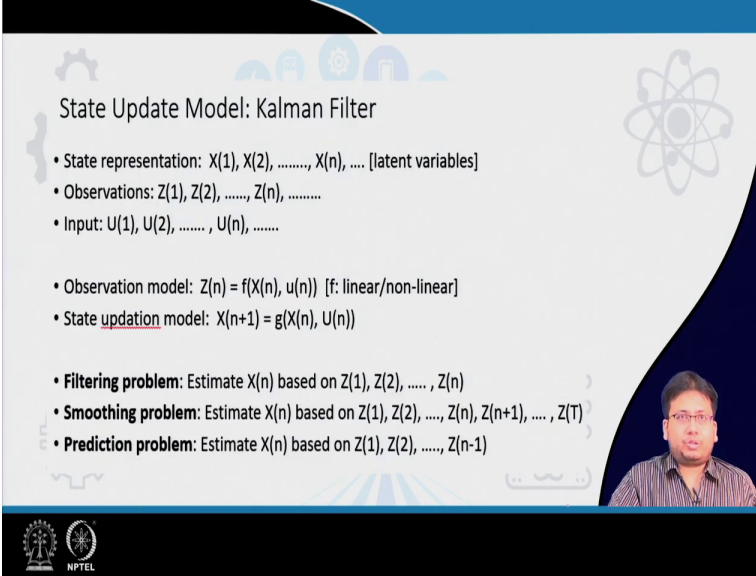
So, that is how it data assimilation usually works and it is solved in a probabilistic framework, that is, we like we use the probabilistic framework to take into account the various uncertainties. I mean, even when we are building the model of a particular process we will not be able to like include all aspects of that process in it. So, those aspects of the I mean there will be various influencing variables which we are not aware of.

So, what we usually do is those unknown influencing variables we just consider them to be some kind of random noise which follow some particular probability distribution. So, in this case also the system state as well as the various observation they are expressed using the probability density function, the latent state which we just talked about of course, it cannot be measured directly. So, like it. In fact, it might happen that different values of that latent variable may result in the same observation.

So, from the observations, we may never be certain about what value the latent state variable has. So, in such a situation we may want to define some kind of a probability distribution over the range of possible values of it ok. And in such models, we also usually assume that the system state follows the Markov process where the present depends where the value at present depends only on the past value not in the I mean the previous time step not on like say 10-time steps away or things like that.

So, the uncertainties which we mentioned they are represented as probability density functions and these PDFs these also keep evolving over time and like the we have to update these PDFs using the Bayes theorem.

(Refer Slide Time: 07:05)



State Update Model: Kalman Filter

- State representation:  $X(1), X(2), \dots, X(n), \dots$  [latent variables]
- Observations:  $Z(1), Z(2), \dots, Z(n), \dots$
- Input:  $U(1), U(2), \dots, U(n), \dots$
- Observation model:  $Z(n) = f(X(n), u(n))$  [f: linear/non-linear]
- State update model:  $X(n+1) = g(X(n), U(n))$
- **Filtering problem:** Estimate  $X(n)$  based on  $Z(1), Z(2), \dots, Z(n)$
- **Smoothing problem:** Estimate  $X(n)$  based on  $Z(1), Z(2), \dots, Z(n), Z(n+1), \dots, Z(T)$
- **Prediction problem:** Estimate  $X(n)$  based on  $Z(1), Z(2), \dots, Z(n-1)$

The slide features a blue header, a light blue background with faint icons (gears, a person, a house, a network), and a dark blue curved border on the right. A small video inset in the bottom right corner shows a man with glasses and a striped shirt. The NPTEL logo is in the bottom left corner.

So, now one of the most popular and successful approaches to this problem of data assimilation is the so called Kalman Filter. So, in this frame; so, Kalman Filter is a kind of framework. So, let us say that the state representations they are like denoted by  $X(1), X(2), \dots, X(n), \dots$ ; so, these  $X$  these are the latent variables; so, 1, 2, ...,  $n$ ,... these denote the various time steps.

Now, corresponding to those system states which are of course, latent we have observations which we denote by  $Z(1), Z(2), \dots, Z(n), \dots$ , and we also have inputs  $U(1), U(2), \dots, U(n), \dots$ . So, these inputs can be like if you remember we had like when we had discussed the geostatistical equation and the corresponding spatio temporal stochastic model, we had considered the covariates.

So, the covariates are non random variables they are exogenous variables which are in a sense they are input to the network. So, these  $U(1), U(2), \dots$  they can be considered as the exogenous covariates of the process. Now, the observe we have like this kind of models they have two

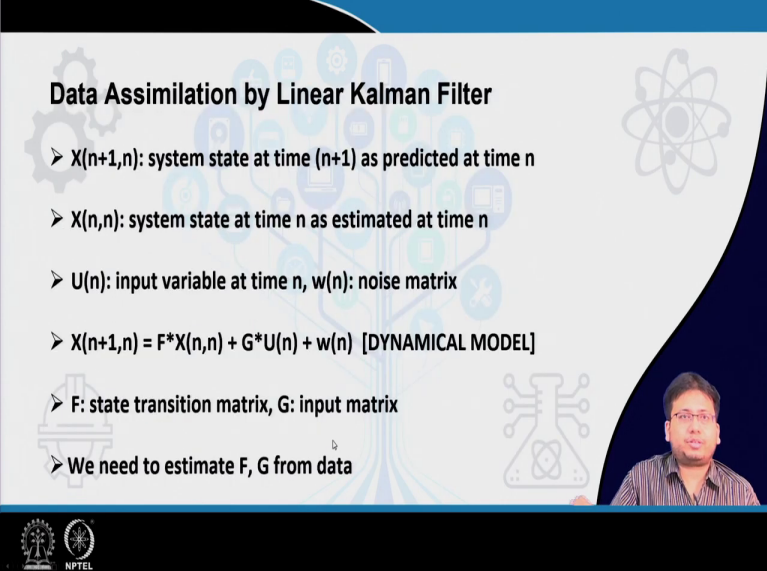
components; one is the observation model which depends on the system state  $X$  as well as the those covariates  $U$ .

So, like in a sense this is something which we had discussed in those lectures and then there is a state updation model also which basically says how  $X(n + 1)$  depends on  $X(n)$  and  $U(n)$ . That is given the system state and the covariates at this point of time how will the state variable be at the next point of time ok; so, that is the state updation model.

So, observation model and state updation model these are the two fundamental parts of this kind of dynamical process model. Now, then we usually are interested in three kinds of problems related to this. One is the filtering problem, that is suppose you have observations from time  $1, \dots, n$  and you want to estimate  $X(n)$  that is the current time the current system state based on the current and all the past observations that is we call what we call as the filtering problem.

Next comes the smoothing problem; what is that? Let us say we have observed the full time series from  $T = 1, \dots, T = T$  we have all the observations. So, based on that at any intermediate time step we are interested in knowing the system step that is the smoothing problem. And finally, the prediction problem that is you have observation till time  $(n - 1)$  and based on that you are trying to predict or forecast what the system state is next going to be at  $T = n$ . So, these are the three main problems which we usually try to solve.

(Refer Slide Time: 10:14)



**Data Assimilation by Linear Kalman Filter**

- $X(n+1,n)$ : system state at time  $(n+1)$  as predicted at time  $n$
- $X(n,n)$ : system state at time  $n$  as estimated at time  $n$
- $U(n)$ : input variable at time  $n$ ,  $w(n)$ : noise matrix
- $X(n+1,n) = F \cdot X(n,n) + G \cdot U(n) + w(n)$  [DYNAMICAL MODEL]
- $F$ : state transition matrix,  $G$ : input matrix
- We need to estimate  $F$ ,  $G$  from data

The slide features a blue header, a light blue background with technical icons (gears, atom, circuit), and a video inset of a speaker in the bottom right corner. The NPTEL logo is in the bottom left.

Now, let us denote by  $X(n + 1, n)$  this denotes the system step  $X$  at time  $(n + 1)$  as predicted as time  $n$ . So, like  $X(n)$  basically means the actual system state at  $T = n$ , but the system step we are system state we are not able to measure directly. So, we like always have to have some kind of estimate of it and those estimates which we make they also vary over time.

So, if I am making the estimate of the present at present, I like I will it will take some value or if I am estimating the future from the present or the past from the present, they will have different values. So, by  $X(n + 1, n)$  or in general  $X(i, j)$ , i mean the estimate of the system step at time  $i$  as estimated from time  $j$  ok.

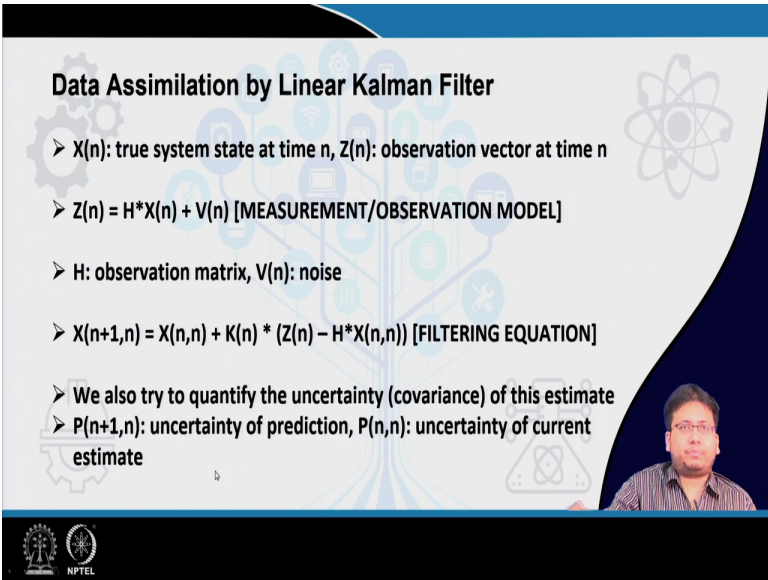
Similarly,  $X(n, n)$  will mean the system state at time  $n$  as estimated at that same time  $n$  and of course,  $U(n)$  is the input variable and  $w$  is what we call as the noise. So, like we already said why these processes should have some like we should consider some noise in the process. So, this is what the dynamical model of the Kalman filter looks like, that is,  $X(n + 1, n)$  that is the my estimate of the time of the system state at the next step of time is going to be or can be represented like this.  $X(n + 1, n) = F * X(n, n) + G * U(n) + w(n)$ . So, here  $X(n, n)$  is of course, the current estimate of the current state and  $F$  is the like transformation matrix or you can call it as the state transition matrix.  $G * U(n)$ ;  $U(n)$  is the covariates at this point of



time and they can be mapped to the system state in future by this matrix called  $G$  and then we have the noise  $w(n)$ .

So, note that this  $F$  and  $G$  matrices like they you can you may be able to relate them to the  $A$  and  $B$  matrices which we had discussed like in lecture 3 or lecture 4 when we are dealing with the geo statistical equation. Now,  $F$  is the state transition matrix and  $G$  is called the input matrix; so, of course, we need to estimate these matrices  $F$  and  $G$  from the data.

(Refer Slide Time: 12:57)



**Data Assimilation by Linear Kalman Filter**

- $X(n)$ : true system state at time  $n$ ,  $Z(n)$ : observation vector at time  $n$
- $Z(n) = H * X(n) + V(n)$  [MEASUREMENT/OBSERVATION MODEL]
- $H$ : observation matrix,  $V(n)$ : noise
- $X(n+1,n) = X(n,n) + K(n) * (Z(n) - H * X(n,n))$  [FILTERING EQUATION]
- We also try to quantify the uncertainty (covariance) of this estimate
- $P(n+1,n)$ : uncertainty of prediction,  $P(n,n)$ : uncertainty of current estimate

The slide features a background with a blue and white geometric pattern. A small video inset in the bottom right corner shows a man with glasses speaking. The NPTEL logo is visible in the bottom left corner.

Now, apart from that apart from this dynamical model there are many other equations in this related to the Kalman Filter; say, most importantly the measurement of the observation model. So, let us say  $X(n)$  is the true state of the system at  $T = n$ ; so, note that here we are talking about  $X(n)$  as opposed to  $X(n, n)$  or  $X(n + 1, n)$ .

So, these like whenever we have two indices two time, indices of  $X$  like this what we are talking about is estimates of  $X$ . In this case we are talking about the true value of  $X$  which is of course, unknown, but if it were known like we like then it would follow this kind of a relation. Where  $Z(n)$  is the observation which is of course, known and then  $V(n)$  is the like measurement error and then  $H$  is the observation matrix.

$$Z(n) = H * X(n) + V(n)$$

So, now it can be shown that this  $X$  this expression for  $X(n + 1, n)$  that is like here we are basically talking about the this problem the prediction problem. That is I have observation till a particular time point I am trying to predict the estimate the next the time step at the next I mean the system state at the next time step.

So,  $X(n + 1, n)$  this can be represented in the dynamical model using this equation.

$$X(n + 1, n) = X(n, n) + K(n) * (Z(n) - H * X(n, n))$$

But it can also be like rewritten in the following way using this quantity called  $K(n)$  which is known as the Kalman gain and this is also called the filtering equation. So, here you can see the estimate of the next state or as observed from the current state is the current state plus the Kalman gain times this thing.

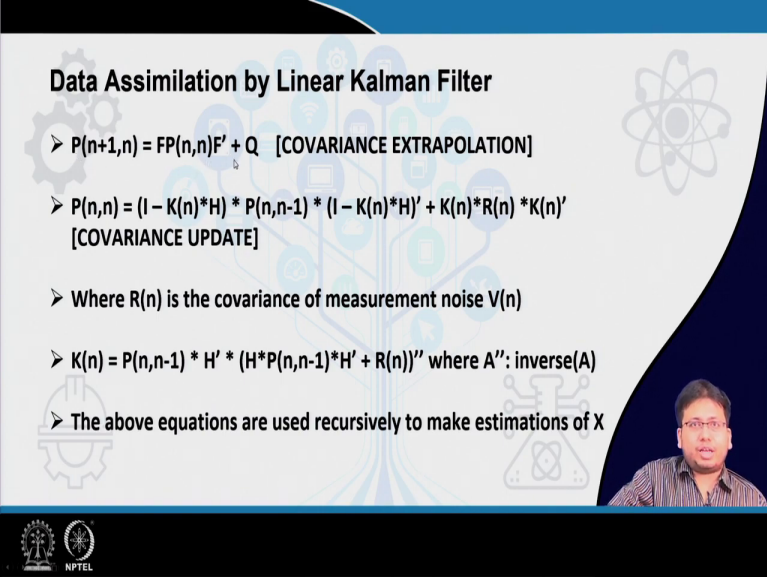
What is this thing? This thing is nothing but the like you like you can say the difference between what is observed and what is predicted by the model. Note that  $X(n, n)$  is what the is my estimate of the current state; so, if that was like if that was indeed the correct the if that were indeed the correct state of the system. Then the corresponding observation would have been  $H$  times that value plus the noise, but the noise we may consider to be zero mean.

So, the expected value of the observation at this current observation should be just  $H * X(n, n)$  but the actual observation is  $Z(n)$ ; so, these two may not be exactly equal to each other. So, the Kalman gain basically like transforms the this error of my of observe of the current observation and it transforms the or that is based on the current error of observation which I make it tries to estimate the next step.

So, this that is this is this  $K(n)$  this is a very crucial quantity of this Kalman filter this is known as the Kalman gain. Now, apart from predicting the system step at the next time step, we also try to quantify the uncertainty of this estimate. That is for this  $X$  we like this the  $X$  which we are talking about here in a sense it is like the expected value or the mean value.

But it there are like I mean like instead of making a point forecast we should also keep in mind that it could like there can be an error of the forecasting also, that is instead of predicting exactly this value it can predict a nearby value also. So, that covariance is called  $P(n + 1, n)$ ; so, this is the uncertainty of prediction of  $X(n + 1, n)$ . And  $P(n, n)$  is the that is the uncertainty of the current estimate ok, that is it is the variance of  $X(n, n)$ .

(Refer Slide Time: 17:12)



**Data Assimilation by Linear Kalman Filter**

- $P(n+1, n) = F P(n, n) F' + Q$  [COVARIANCE EXTRAPOLATION]
- $P(n, n) = (I - K(n) H) P(n, n-1) (I - K(n) H)' + K(n) R(n) K(n)'$  [COVARIANCE UPDATE]
- Where  $R(n)$  is the covariance of measurement noise  $V(n)$
- $K(n) = P(n, n-1) H' (H P(n, n-1) H' + R(n))^{-1}$  where  $A^{-1}$ : inverse(A)
- The above equations are used recursively to make estimations of  $X$

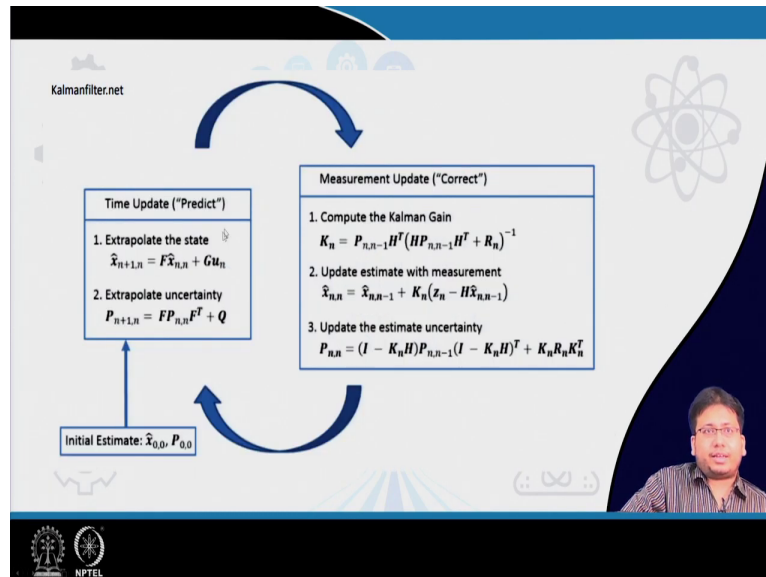
Now, like we it is possible to like calculate these variances in terms of each other. So, like as you can see here like this actually follows or this equation can be derived from the from these equations here with making some appropriate assumptions; so, this is known as the covariance extrapolation.

So, here basically we are estimating the uncertainty of the next state prediction in terms of the current state prediction right. And this  $F$  this is like this matrix is like what we considered like earlier this  $F$  matrix right the state transition matrix and  $Q$  is like is the like is the it is the noise covariance. Now, this  $P$  of  $n$  comma; so, here we are estimating  $P(n + 1, n)$  in terms of  $P(n + 1, n)$ .

But, this  $P(n + 1, n)$  itself can be expressed in terms of  $P(n, n - 1)$ , like and that estimate again involves this quantity of Kalman gain. So, all these quantities the Kalman gain the

$P(n + 1, n)$  and  $P(n)$  all these can be written recursively one in terms of the other. That is we first get an like expression of  $K(n)$ ; using  $K(n)$  or let us say using  $K(n)$ , I and  $K$  and  $P(n, n - 1)$  I can get  $P(n, n)$ . Now, once I have  $P(n, n)$ , then I can have  $P(n + 1, n)$  using which I can again derive  $K(n + 1)$  and so on and so forth.

(Refer Slide Time: 19:10)



So, the whole thing goes around in a cycle and that is illustrated in this diagram which I have taken from this excellent tutorial called Kalmanfilter.net. So, here it is like saying that we have an initial estimate of the system state and the corresponding covariance; so, this is the initial condition. Now, based on this first we extrapolate the state the what the next state is going to be and its corresponding uncertainty right.

So, they are given as  $X(n + 1, n)$  and  $P(n + 1, n)$ . Now, so this is you can say this is my prediction of what the system step is going to be in the next step. Now, at the next step I get the data, I get my observation; so, what I do now first of all I calculate the Kalman gain. So, the Kalman gain I can calculate using the equation which we have already talked about, next we can update our estimation of the system state using this kind of an equation.

So, note the  $X(n, n - 1)$  is like it is the estimate of the current state which I had earlier I mean. So, when I; so, when I made this kind of prediction; so, this prediction in a sense it becomes

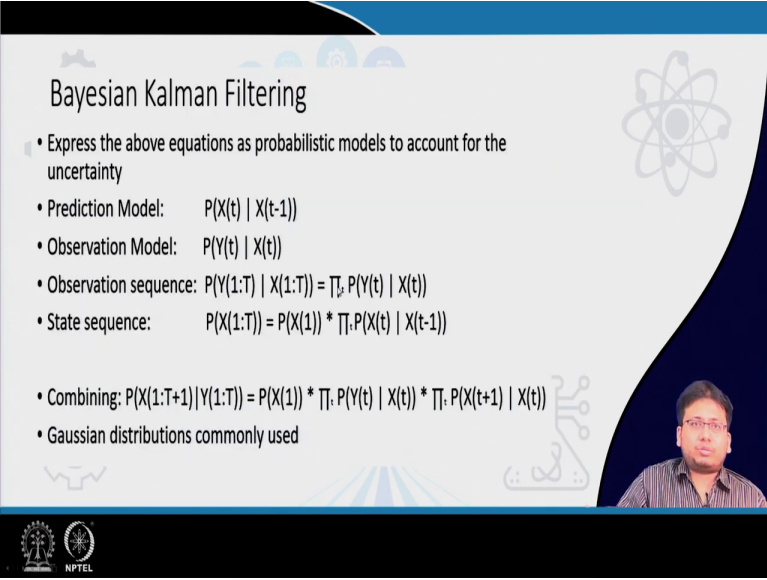
something like this is the this is the same thing. But here now we are talking about at time  $T = n + 1$  I am trying to update this thing this prediction by using the Kalman gain.

Now, that I have obtained  $Z(n)$ ; so, I can now calculate this and the Kalman gain for the Kalman gain I already have this equation in terms of  $P(n, n - 1)$ , but  $P(n, n - 1)$  is again something which I had already obtained I mean I had already calculated.

Now, when I get this updated estimate then I can of the current state then I can also estimate the uncertainty or the covariance of the current state, that is  $P(n, n)$  that can be calculated in terms of the quantities that are already known. And then once I have  $P(n, n)$ ; now, I go back to the this state. Now, I will calculate  $P(n + 1, n)$  in terms of  $P(n, n)$  and I will also calculate  $X(n + 1, n)$  using  $X(n, n)$ .

Because, I have got I already know this quantity and I will again pass it on. So, these two steps will they will keep on like going around in loop; it is like I will in this step I am predicting, in this step I am correcting that is once I get the data, I correct my predictions I, reestimate the system step and then again I predict what the next system step will be. Again I get new data I like make corrections of my estimate and then again predict the next step and this just goes on; so, this is how the Kalman filtering works.

(Refer Slide Time: 22:05)



**Bayesian Kalman Filtering**

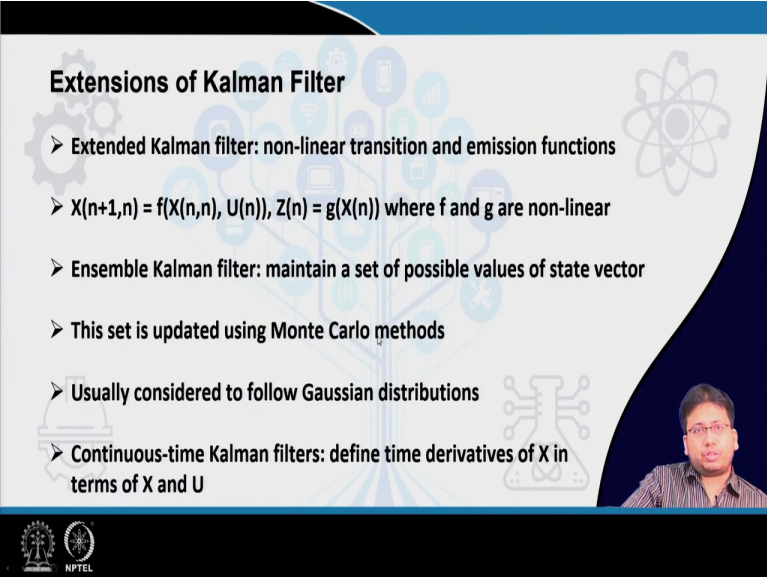
- Express the above equations as probabilistic models to account for the uncertainty
- Prediction Model:  $P(X(t) | X(t-1))$
- Observation Model:  $P(Y(t) | X(t))$
- Observation sequence:  $P(Y(1:T) | X(1:T)) = \prod_t P(Y(t) | X(t))$
- State sequence:  $P(X(1:T)) = P(X(1)) * \prod_t P(X(t) | X(t-1))$
- Combining:  $P(X(1:T+1) | Y(1:T)) = P(X(1)) * \prod_t P(Y(t) | X(t)) * \prod_t P(X(t+1) | X(t))$
- Gaussian distributions commonly used

The slide features a blue header, a white body with a blue atom-like graphic on the right, and a black footer with the NPTEL logo. A video inset in the bottom right corner shows a man with glasses speaking.

So, like, as you like the equations we were discussing so far all those equations we had some kind of a noise variable this kind of a random variable right, these  $Q$ 's or like this  $V$ 's the  $W$ 's these were all the random variables. But we had written these as equations involving a random variable, but instead of writing them an equation we can directly go into some kind of a probability distribution that is we can; so, the prediction model is become something like this.

Given the current state what is the probability distribution over the next state or given the current system state what is the probability distribution over the observation. Now, given a sequence of like system states what is going to be the sequence of the observations and things like that. So, we can actually define these kinds of probability distributions and in the case of Kalman filters we usually use the Gaussian distributions for all these things, that is each of these distributions is considered to be a Gaussian distribution.

(Refer Slide Time: 23:23)



**Extensions of Kalman Filter**

- **Extended Kalman filter:** non-linear transition and emission functions
- $X(n+1,n) = f(X(n,n), U(n)), Z(n) = g(X(n))$  where  $f$  and  $g$  are non-linear
- **Ensemble Kalman filter:** maintain a set of possible values of state vector
- This set is updated using Monte Carlo methods
- Usually considered to follow Gaussian distributions
- **Continuous-time Kalman filters:** define time derivatives of  $X$  in terms of  $X$  and  $U$

The slide features a blue header and footer. The footer contains the NPTEL logo and a small video inset of a man speaking. The background has a faint pattern of gears and a stylized atom.

If we do so, if we use the Gaussian distribution then we are actually able to calculate the in the closed form expressions for these functions like  $F$  and so on which we are talking about. Now, different extensions of the Kalman filters are possible; so, one is; so, first of all we can have the extended Kalman filters.

So, in the Kalman filter which we are talking about this is actually a linear Kalman filter in the sense that the equations we are considering the observation model or the state transition model etcetera. They are all as you can see they are all linear models, but we can make them non-linear by including some kind of non-linear transition functions  $F$  or observation function  $G$  etcetera.

Then there is a possibility of using the ensemble Kalman filters where instead of having a single estimate of the state vector, we can have a set of possible values of them. And this set at every stage is like is updated using the Monte Carlo methods and this is usually done to it is usually considered to follow the Gaussian distributions.

We can also have the continuous time Kalman filters in which we like instead of defining of  $X(n + 1, n)$  and things like that we actually define the time derivatives of  $X(n)$  in terms of say let say the current values of  $X(n)$   $U(n)$  and things like that.

(Refer Slide Time: 24:53)

**Environmental Science & Technology**

**In Situ Monitoring of Groundwater Contamination Using the Kalman Filter**

Franziska Schmidt,<sup>1</sup> Haruko M. Wainwright,<sup>1,2</sup> Boris Faybishenko,<sup>3</sup> Miles Denham,<sup>4</sup> and Carol Eddy-Dilek<sup>1</sup>

<sup>1</sup>Department of Nuclear Engineering, University of California Berkeley, Etchevery Hall, 2521 Hearst Avenue, Berkeley, California 94709, United States  
<sup>2</sup>Climate and Ecosystem Sciences Division, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, MS 74R-316C, Berkeley, California 94720-8126, United States  
<sup>3</sup>Energy Geosciences Division, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, California 94720-8126, United States  
<sup>4</sup>Panoramic Environmental Consulting, LLC, P.O. Box 906, Aiken, South Carolina 29802, United States  
<sup>5</sup>Savannah River National Laboratory, Savannah River Site, Aiken, South Carolina 29808, United States

**ABSTRACT:** This study presents a Kalman filter-based framework to establish a real-time *in situ* monitoring system for groundwater contamination based on *in situ* measurable water quality variables, such as specific conductance (SC) and pH. First, this framework uses principal component analysis (PCA) to identify correlations between the contaminant concentrations of interest and *in situ* measurable variables. It then applies the Kalman filter to estimate contaminant concentrations continuously and in real-time by coupling data-driven concentration-decay models with the previously identified data correlations. We demonstrate our approach with historical groundwater data from the Savannah River Site F-Area. We use SC and pH data to estimate tritium and uranium concentrations over time. Results show that the developed method can estimate these contaminant concentrations based on *in situ* measurable variables. The estimates remain reliable with less frequent or no direct measurements of the contaminant concentrations, while capturing the dynamics of short- and long-term contaminant concentration changes. In addition, we show that data mining, such as PCA, is useful to understand correlations in groundwater data and to design long-term monitoring systems. The developed *in situ* monitoring methodology is expected to improve long-term groundwater monitoring by continuously confirming the contaminant plume's stability and by providing an early warning system for unexpected changes in the plume's migration.

**Supporting Information**

Contaminant concentration evolution model  
 In situ water quality data  
 Kalman Filter  
 Anomaly detection  
 Continuous contaminant concentration estimation

NPTEL

So, this Kalman filter is a concept which has often been used in the like in the domain of earth system sciences in various kinds of problems. So, for example, this one is talking about the *in situ* monitoring of groundwater contamination. So, we like let we are focusing on the contamination of groundwater, but we have observations at sparse point sparse time intervals.

And like we have what is known as the in situ observations that is only at certain points certain particular locations we have say something like well where we have the it is possible to measure the purity of ground water using sensors and things like that. But, in other locations and we do not have these kinds of measurements.

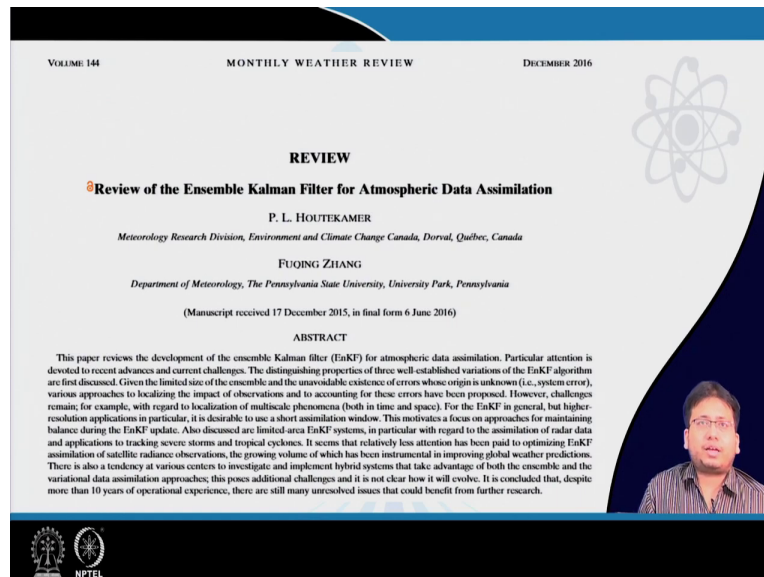
So, what we do is we use a Kalman filter; so, at like at any given location we try to estimate. So, like in this case actually rather than time we are doing the like we are actually moving over space and at any given location let us say I have some kind of model for the ground water contamination. So, let us say in this case the system state may represent the purity of the ground water.

And like at and we see that is the model basically tries to predict how in other locations the I mean the and also at as a time series that is also possible. At other locations and at other points of time how the purity of ground water will change.

Now, at we have when we have in situ measurements that is at a single location at a and at a single point of time then I will once I have that observations that then I can modify my system state accordingly. And then again go on predicting at other locations and other time points then again, I will have some again some in situ observations again I will update and. So, on the whole process will go on ok.

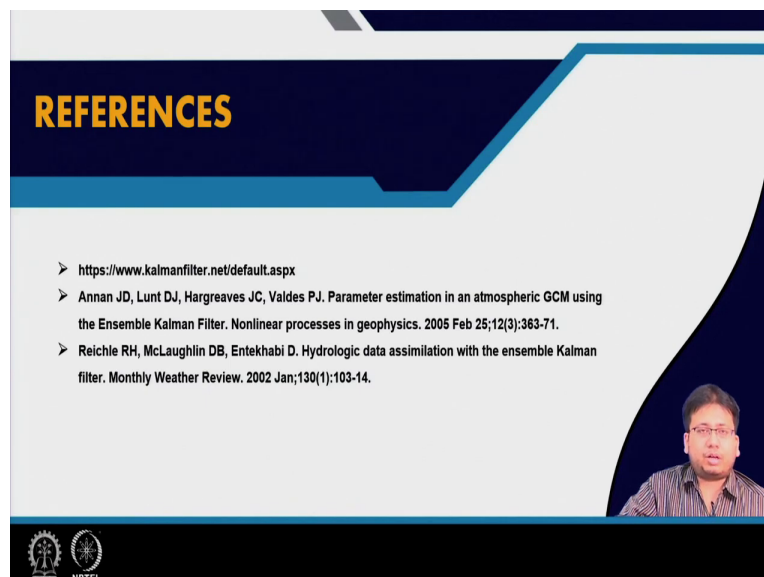


(Refer Slide Time: 27:05)



And similarly, I can also do the review like of the ensemble Kalman filters of the atmospheric for atmospheric data simulation and so on.

(Refer Slide Time: 27:17)



So, like regarding references I will actually direct you to this excellent tutorial called Kalman filter dot net. So, like here all the various equations related to the Kalman filters and their actual

values assuming the Gaussian distribution and so on they are derived in great details. I actually encourage you to go through all those equations and try to understand the all the derivations etcetera.

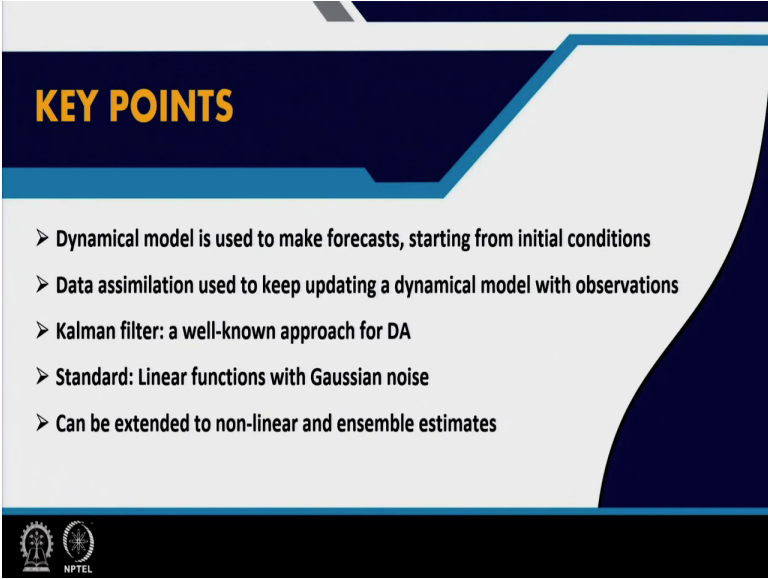
And then there are also several papers where these concepts have been like used in various domains of earth sciences; for example, there are GCMS these are the global climate models. Now, in case of the these global climate models, this data assimilation is a very important thing because like these models once they start once we run them with some initial conditions.

They may go along a entirely different trajectory which have nothing to do with the like what has actually been observed. So, it is necessary to ground those simulations to reality and that is achieved with the help of data assimilations.

But this GC when did these GCMS are assimilated with data, then they actually give us like what is known as reanalysis data. So, when we have these data sets of let us sea surface temperature from all over the world and things like that, that is like such data is actually what is known as reanalysis data that is it is not pure observations. We cannot actually measure the sea surface temperature at all parts of the world because we do not have so much of sensing power.

But we can like you can say marry the observations with some kind of model that is the purpose of data assimilation and then we can get the estimated data like from all over the world. And then there like in various other hydrological and other processes these concepts of data assimilation is often used and this Kalman filter or the various extension of the Kalman filter that we discussed they are all very useful.

(Refer Slide Time: 29:34)



**KEY POINTS**

- Dynamical model is used to make forecasts, starting from initial conditions
- Data assimilation used to keep updating a dynamical model with observations
- Kalman filter: a well-known approach for DA
- Standard: Linear functions with Gaussian noise
- Can be extended to non-linear and ensemble estimates

NPTEL

So, the key points here is that the dynamical models are used to make forecast starting from some initial conditions. The data assimilation is used to keep updating the dynamical model with the observations as and when they arrive and to keep; so, the these updates include the state variables as well as the various parameters.

Now, a well-known approach to data assimilation is Kalman filter its standard form of Kalman filter is using linear functions for the measurement as well as for state transitions using the Gaussian noise. And this can, but it can also be extended to non-linear and ensemble estimates.

So, with this we come to the end of not only this lecture, but also this module 1 on spatio-temporal statistics. So, the next module will be the machine learning methods for earth system science; so, which will start from the next lecture; so, till then good bye.