Frontier Informatics 2022 (English Class)

Graduate School of Science and Technology

Degree Programs in System and Information Engineering

Contents of the Lecture ：

Students will learn basics about 4 dimensional data analysis in meteorology. First, an introductory lecture about data assimilation. Then, students will lean optimal interpolation, 3D variation method, 4D variation method, and Kalman filter which is known as one of machine learnings.

Time and Date： 12:15-15:00, 6 and 13 July, 2022 Spring C
Instructor： Hiroshi L. Tanaka, Professor （CCS-302, Tel: 853-6482）
Theme of Lecture： Meteorological data analysis using Kalman filter
Place： CCS WS Room (and Zoom), Exercise using icho server
Text：R. Daley 1991 Atmospheric Data Analysis Cambridge Univ. Press
F. Bouttier and P. Courtier 1999 Data assimilation concepts and methods ECMWF Internal Report
E. Kalnay 2003 Atmospheric Modeling, Data Assimilation and Predictability Cambridge Univ. Press
and a Handout by H. L. Tanaka: Data Assimilation

Schedule：

1. **Weather forecasting** Dynamical system
2. **Discovery of chaos** Future prediction、
3. **Exercise 1** Chaos model
4. **Data assimilation** Optimal interpolation
5. **Kalman filter** Machine learning
6. **Exercise 2** Kalman filter

Score： Attendance 0%, Report 100%

Lecture 2 :
Data Assimilation using Kalman Filter

Data Assimilation

1. A simple model of data assimilation

As a simple model of data assimilation, let's consider an observation of temperature using two different thermometers. Let T_1 and T_2 be temperature observations by thermometer 1 and 2 with different accuracy. Suppose T_t be a true temperature (unknown), then $\epsilon_1 = T_1 - T_t$ and $\epsilon_2 = T_2 - T_t$ are observation errors. For simplicity, we assume there is no bias in observations. So, $\overline{\epsilon_1} = \overline{\epsilon_2} = 0$, where the over bar denotes time mean or expectation. Then the variance of observation error is expressed as $\overline{\epsilon_1^2} = \sigma_1^2$、$\overline{\epsilon_2^2} = \sigma_2^2$, and we assume there is no correlation between observation errors, i.e., $\overline{\epsilon_1 \epsilon_2} = 0$.

With these preparations, the goal is to evaluate the best estimate of temperature T_a which may be close to the truth. We call it as analysis of the best temperature estimate. In the linear framework, best estimate of the analysis is expressed using two coefficients a_1 and a_2 as:

$$T_a = a_1 T_1 + a_2 T_2.$$

Since there is no bias in observations, we have $\overline{T_1} = \overline{T_2}$. Also since there is no correlation between errors, the weight coefficients must satisfy $a_1 + a_2 = 1$. Then, the error variance of the analysis would be described as:

$$\sigma_a^2 = \overline{(T_a - T_t)^2} = \overline{[a_1(T_1 - T_t) + a_2(T_2 - T_t)]^2} = a_1^2\sigma_1^2 + a_2^2\sigma_2^2.$$

We eliminate a_2 from $a_1 + a_2 = 1$, and differentiate the variance by a_1 to look for the minimum of the analysis error variance. As a result, we can obtain the following relation for the best estimate of the temperature T_a by:

$$\frac{\partial \sigma_a^2}{\partial a_1} = \frac{\partial}{\partial a_1}[a_1^2\sigma_1^2 + (1 - a_1)^2\sigma_2^2] = 0.$$

By solving it for coefficients, we obtain

$$a_1 = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}, \quad a_2 = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2},$$

or

$$a_1 = \frac{\frac{1}{\sigma_1^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}, \quad a_2 = \frac{\frac{1}{\sigma_2^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}},$$

This completes the best estimation of the analyzed temperature $T_a = a_1 T_1 + a_2 T_2$.

Here, the inverse of error variance is called precision. Then the weighting coefficient has large value for accurate observation. If the precisions are exactly same, the weights will be $a_1 = a_2 =$

0.5. This is a simple average of the two. This method of obtaining the best estimate of the temperature with known error variances (or precision) is called an optimal interpolation (OI).

By substituting the weight in the analysis variance $\sigma_a^2 = a_1^2\sigma_1^2 + a_2^2\sigma_2^2$, we obtain

$$\sigma_a^2 = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2} = (1-a_2)\sigma_1^2, \quad or \quad \frac{1}{\sigma_a^2} = \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2},$$

and the analysis may be written with a known coefficient as

$$T_a = a_1T_1 + a_2T_2 = T_1 + a_2(T_2 - T_1).$$

Now, we replace the name of $T_1 = T_b$ as background data (or first guess or prediction data), and $T_2 = T_o$ as observation data. Then

$$T_a = T_b + K(T_o - T_b).$$

$$K = \frac{\sigma_b^2}{\sigma_b^2+\sigma_o^2}.$$

This equation is referred to as data assimilation equation, and constructs the basis of the data assimilation for a complex system, such as weather forecasting. From this equation, we find that the model prediction T_b become the analysis T_a if the model prediction is accurate enough to have σ_b^2 is 0, so K=0. Conversely, the observation T_o become the analysis T_a if the observation is accurate enough to have σ_o^2 is 0, so K=1. In the data assimilation equation, the best estimate of the weight coefficient K is somewhere between 0 and 1.0 depending of the magnitudes of the observation error and prediction error.

2. Observation operator

Consider the observation is not by thermometer but by radiometer measuring emissivity y. The emissivity is related to temperature by Stefan-Bolzmann's law.

$$y = h(T) \approx \sigma T^4,$$

and the assimilation equation becomes:

$$T_a = T_b + K[y - h(T_b)].$$

The weight is noted here by K. The second term in the right hand side is called analysis increment. The optimal weight K is expressed using the tangent linear operator $H = \frac{\partial h}{\partial T}$ for h, and is given by

$$K = \sigma_b^2 H(\sigma_o^2 + H\sigma_b^2 H)^{-1}.$$

In this simple example of data assimilation, a scalar variable of temperature is used. When the variable is a vector, the weight K becomes a matrix as shown later.

3. Nudging

A simple data assimilation is called nudging when the analysis data x_a is forced to the observation $y(i)$ by a prescribed weight matrix $K(i,j)$, which is called as a gain matrix.

$$x_a(j) = x_b(j) + K(i,j)[y(i) - x_b(i)].$$

Here, the analysis is located at a grid j and the observation and prediction is given at a grid i. The gain matrix $K(i,j)$ is given subjectively in nudging, and is often a function of the distance between the grids i and j. When the gain is large, the analysis is strongly adjusted to the observation.

4. Optimal Interpolation (OI)

The optimal interpolation (OI) is a simple data assimilation method used for the numerical weather prediction in early age. It is a vector extension of the simple data assimilation described in previous section for a scalar variable. Suppose that the observation of state variables of the atmosphere is denoted by y, the back ground data (i.e., prediction or first guess firld) by x_b, and the analysis by x_a, with those true values by y_t and x_t. Those are vectors with dimension M and N. Then we have observation error $\epsilon_o = y - y_t$, prediction error $\epsilon_b = x_b - x_t$, and analysis error $\epsilon_a = x_a - x_t$. We assume that the errors have no bias, i.e., $\overline{\epsilon_o} = 0$、$\overline{\epsilon_b} = 0$、$\overline{\epsilon_a} = 0$.

For each error vector, we can construct a covariance matrix, such as $R = \overline{\epsilon_o \epsilon_o^T}$, $B = \overline{\epsilon_b \epsilon_b^T}$, and $A = \overline{\epsilon_a \epsilon_a^T}$. Here the superscript T denotes transpose of the matrix. The matrix dimensions become $M \times M$, $N \times N$ and $N \times N$, and we assume N is larger than M. We also assume that the cross-correlation is 0 for different variables such as $\overline{\epsilon_a \epsilon_b^T} = 0$.

With these preparations, the data assimilation by OI may be described by a vector and matrix form of the data assimilation equation.

$$x_a = x_b + K(y - Hx_b).$$

Here, matrix dimension is $N \times M$ for the gain K, and $M \times N$ for the observation operator H. In general, the interpolation from observation point to analysis grids is a part of the observation operator.

In nudging, the gain matrix K for the weight was given empirically. In contrast, the gain matrix for OI is obtained by an optimization so that the analysis error is minimized as demonstrated in the previous section for a scalar variable.

By subtracting true values from the assimilation equation, we obtain a relation for errors;

$$\epsilon_a = \epsilon_b + K(\epsilon_o - H\epsilon_b).$$

With this analysis error ϵ_a, the covariance matrix for analysis error becomes

$$A = \overline{\epsilon_a \epsilon_a^T} = \overline{[\epsilon_b + K(\epsilon_o - H\epsilon_b)][\epsilon_b + K(\epsilon_o - H\epsilon_b)]^T}.$$

By calculation the right hand side of the equation, we obtain:

$$A = B - KHB - BH^TK^T + KRK^T + KHBH^TK^T.$$

Since the diagonal element of A represents a variance of the variable, the optimal gain matrix K may be obtained by minimizing the trace of $Tr(A)$. The minimum can be found by differentiating the trace by K and setting 0 for the right hand side.

$$\frac{\partial Tr(A)}{\partial K} = 0.$$

Here the trace is a scalar and K is a matrix of $N \times M$, so the differentiation yields $N \times M$ matrix. The differentiation is actually a variation method of a functional. The trace in matrix form can be expressed in a tensor form using subscripts with a standard summation rule when the same symbol appears.

$$Tr(A) = A_{ii} = B_{ii} - K_{ij}H_{jk}B_{ki} - B_{ij}H_{kj}K_{ik} + K_{ij}R_{jk}K_{ik} + K_{ij}H_{jk}B_{kl}H_{ml}K_{im}.$$

Here, the matrix transpose is expressed by changing the turn of the subscripts. By the calculus of variation, we obtain the next equation:

$$\frac{\partial Tr(A)}{\partial K_{in}} = -H_{nj}B_{ji} - B_{ij}H_{nj} + R_{nj}K_{ij} + K_{ij}R_{jn} + H_{nj}B_{jk}H_{lk}K_{il} + K_{ij}H_{jk}B_{kl}H_{nl}.$$

For example, $K_{ij}H_{jk}B_{ki}$ is subject to the variation only when j is n. So the variation of this term becomes $H_{nj}B_{ji}$.

We can rewrite the tensor form of the relation in a matrix form as

$$\frac{\partial Tr(A)}{\partial K} = 2(-BH^T + KR + KHBH^T) = 0.$$

In the proof, we use the fact that matrices B and R are symmetric. For symmetric matrices, a relation $H_{nj}B_{ji} = B_{ji}H_{nj}$ may be expressed as $B^T H^T = BH^T$, which has the same dimension as K_{in}. Using this relation, we can obtain the gain matrix as

$$K = BH^T(R + HBH^T)^{-1}.$$

We can notice that the relation in matrix form is same as that in a scalar form in the previous section.

$$K = \sigma_b^2 H(\sigma_o^2 + H\sigma_b^2 H)^{-1}.$$

For simplicity, we can assume the observation operator H is just a unit matrix I when all observations are on the analysis points. Then, we can confirm that K is 0 if prediction is perfect, and K is I if observation is accurate. In the assimilation equation, the optimal K is somewhere between 0 and I depending on the magnitude of the observation error and prediction error.

In general, the analysis covariance matrix is expressed in a lengthy form:

$$A = B - KHB - BH^TK^T + KRK^T + KHBH^TK^T = (I - KH)B(I - KH)^T + KRK^T.$$

But it may be reduced to a simple relation when the gain matrix is optimized as

$$A = (I - KH)B.$$

This relation is same with $\sigma_a^2 = (1 - a_2)^2\sigma_b^2 + a_2^2\sigma_o^2$ and $\sigma_a^2 = (1 - a_2)\sigma_b^2$ in the scalar form of the optimization in the previous section.

Finally, the equations for the data assimilation for OI may be summarized as

$$x_a = x_b + K(y - Hx_b),$$
$$K = BH^T(R + HBH^T)^{-1}.$$

The first equation is the general form of the assimilation equation, and the optimal gain K can be obtained by the second equation from the prescribed observation error and prediction error in matrix forms.

In OI, setting observation error R and prediction error B is a problem. Mostly they are given empirically with some statistical calculation. Since the computation is simple, the OI has been used for long time for the operational data assimilation by Japan Meteorological Agency (JMA) in early time of 1980s. The method is a kind of interpolation in 3D without a use of dynamical model. In the next section, the data interpolation is extended to 4D using a dynamical model.

5. Kalman filter (KF)

Kalman filter was proposed by Kalman (1960) and is regarded as the most advanced ultimate algorism of the 4D data assimilation in the weather forecast. In the algorism, the observed data is assimilated to the running general circulation model (GCM) of the atmosphere so that the model atmosphere is adjusted to the observation and converged ultimately to the atmosphere of the truth. The algorithm is considered as one of the machine learning technologies reading the big weather data in the past.

Suppose that a dynamical model is linear and the discrete numerical model is integrated in time to predict one-time step of the future. The projection mapping of the dynamical system converts the initial data of the state variable x^a_{i-1} to one time step future of the prediction x^b_i by a system matrix M.

$$x^b_i = Mx^a_{i-1}, \tag{1}$$

where the time step of the numerical model is of the order of 1 minute and the time interval to data assimilation is of the order of 6 hours for the GCM. If the model is not perfect, the mapping may be expressed with a model error ϵ^m using the truth x^t_i.

$$x^t_i = Mx^t_{i-1} + \epsilon^m.$$

By subtracting the equation with truth, we obtain an equation for errors

$$\epsilon^b_i = M\epsilon^a_{i-1} - \epsilon^m,$$

where ϵ^a_i is analysis error, and ϵ^b_i is prediction error. Using the errors, we can compute covariance matrices for the analysis error $A = \overline{\epsilon^a\epsilon^{aT}}$ and for prediction error $B = \overline{\epsilon^b\epsilon^{bT}}$ by taking an expectation (i.e., time mean). Therefore, the error matrices should have a next relation by substitution.

$$B_i = MA_{i-1}M^T + Q. \tag{2}$$

Here, $Q = \overline{\epsilon^m\epsilon^{mT}}$ is a covariance matrix for the model error. We have assumed that the cross-covariance is 0 between different kinds of errors.

On the other hand, we already have an assimilation equation and the optimal weight of gain matrix as derived from the OI.

$$x_i^a = x_i^b + K_i(y_i - Hx_i^b), \tag{3}$$

$$K_i = B_i H^T (H B_i H^T + R)^{-1}. \tag{4}$$

In Kalman filter the gain matrix of K_i is called Kalman gain. Using the Kalman gain, we already know that the analysis error A_i is related to the prediction error B_i as

$$A_i = (I - K_i H) B_i. \tag{5}$$

If both Kalman gain and observation operator are I, analysis error is 0, and if Kalman gain is 0, analysis error is same as prediction error.

In summary, the algorithm of Kalman filter can be described by a set of equations (1) to (5). In order to run the Kalman filter we need to prepare the initial conditions for analysis x_{i-1}^a and its covariance matrix A_{i-1} at $(i-1)$-th step. The covariance matrices for observation error R and model error Q must be prescribed. If the model is perfect, we know Q =0. The observation error is given by a diagonal matrix with no cross-correlation between errors. Then the inverse of the diagonal matrix can be computed easily by setting inverse for the diagonal elements.

The time integration of the Kalman filter starts from the model prediction by (1) to get a prediction x_i^b and (2) to get B_i. Once B_i is computed, Kalman gain K_i is computed by (4). For convenience, we may assume $H = I$. Then the assimilation equation (3) combines the observation y_i and the prediction x_i^b to get the optimal analysis x_i^a using the optimal weight K_i. Finally, the covariance matrix A_i is updated from B_i using (5), which completes the algorithm of the Kalman filter. With the updated analysis x_i^a and its covariance matrix A_i at i-th step, we can advance the time to $(i+1)$-th step to repeat the loop of the time integration. The loop is referred to as analysis-prediction cycle. As inferred from (5), the analysis error tends to decrease ultimately to zero for the linear model by assimilating new observations by the analysis-prediction cycle. That means the model atmosphere tends to converge to the true atmosphere by learning a big data of past observations. How nice is it! We can predict the future perfectly.

6. Example of 4D data assimilation

As an example of simple Kalman filter, let's introduce a simple advection equation of temperature T with a constant wind U as the dynamical system:

$$\frac{\partial T}{\partial t} = -U \frac{\partial T}{\partial x}.$$

The solution to this partial differential equation is a transport of initial distribution of $T(x, y)$ by the wind speed U. The computational domain is a periodic boundary, and a half domain is land with observation, and the rest half is ocean with no observation. Let's start from uniform temperature, with repeated data assimilation over the land. Since we know the observed distribution of T moves to ocean with the same shape by the dynamical model, Kalman filter keeps assimilating the observation into the model atmosphere. Hence, when the shape of temperature moves around one periodic cycle of the domain, all shape of temperature is accumulated (assimilated) into the model atmosphere. In other word, the model atmosphere has converged to the true atmosphere. This is a process of machine learning. Once the model atmosphere has converged to the true atmosphere, we can predict the future using the perfect model atmosphere.

7. Extended Kalman filter (EKF)

Above description of the Kalman filter is for a linear dynamical model. However, the dynamical model in most physical application is nonlinear. So, we write the nonlinear mapping m from x^a_{i-1} to x^b_i:

$$x^b_i = m(x^a_{i-1}).$$

For a true state variable, we need to add model error ϵ^m

$$x^t_i = m(x^t_{i-1}) + \epsilon^m.$$

By subtracting the truth, we obtain a equation for errors:

$$\begin{aligned}
\epsilon^b_i &= m(x^t_{i-1} + \epsilon^a_{i-1}) - m(x^t_{i-1}) - \epsilon^m \\
&= m(x^t_{i-1}) + \frac{\partial m}{\partial x}\epsilon^a_{i-1} - m(x^t_{i-1}) - \epsilon^m \\
&= M_{i-1}\epsilon^a_{i-1} - \epsilon^m
\end{aligned}$$

Here, the nonlinear model of i-th time step m_i has been linearized by its tangent linear model $M_i = \frac{\partial m_i}{\partial x}$ represented by Jacobian matrix. The new algorithm for the nonlinear dynamical system is referred to as extended Kalman filter (EKF). In EKF, the model time integration is conducted by the nonlinear model, but the time integration of the covariance matrix is conducted by the linear model with the Jacobian matrix. More over, we need to introduce a covariance inflation ρ to amplify the error covariance matrix for the model prediction, such as:

$$B_i = (1+\rho)(M_{i-1}A_{i-1}M^T_{i-1} + Q).$$

The matrix transpose of the tangent linear model is called adjoint model. The inflation reflects the contributions from higher order terms in the Tayler series expansion of the nonlinear model when it is approximated by the tangent linear model. It is well known that the Kalman filter would diverge without this inflation. Since the filter ultimately converged to the truth to becomes a perfect model. Then the model becomes over confidence to reject a new observation because K has converged to 0. Without the data assimilation of the real world, the model would diverge to its one world. The value of inflation is of the order of 0.1. We can construct the optimal estimate of the inflation using a Kalman filter algorithm in the course of the time integration of the main model.

The following is the summary of the EKF. When we refer to the Kalman filter KF, it often means EKF in most of the applications.

$$x^b_i = m(x^a_{i-1}),$$

$$B_i = (1+\rho)(M_{i-1}A_{i-1}M_{i-1}^T + Q),$$

$$K_i = B_i H^T (H B_i H^T + R)^{-1},$$

$$A_i = (I - K_i H) B_i,$$

$$x_i^a = x_i^b + K_i(y_i - H x_i^b).$$

Kalman filter is called as the ultimate 4D data assimilation algorithm. Unfortunately, the EKF is not used in the numerical weather prediction because the matrix inverse computation is involved in the equation to get Kalman gain. Numerical weather prediction model usually has a dimension of $N = 10^9$ for the state variables of temperature, wind, pressure etc. at 3D x-y-z grids over the entire globe for the GCM. Matrix inversion for 10^9 by 10^9 is impossible by present computer power. For this reason, alternative approach called ensemble Kalman filter has been developed in numerical weather prediction. Because of chaos in the numerical model prediction, we have developed an ensemble prediction with various initial conditions for the state variable. In the ensemble Kalman filter, the model prediction error B_i is approximated by the spread of the ensemble model predictions with M members of the order 100. Therefore, the rank of B_i is degenerated to M by M for this method. So we can compute the matrix inversion of M by M to get the Kalman gain. The detail of the modern and advanced technique of the Kalman filter is reserved to the other class in Dynamic Meteorology. However, the concept of the original Kalman filter may be very important for many applications in physics and engineer.

An alternative 4D data assimilation method has been developed based on a variational method. The famous one is 3D variational method (3D VAR) which is an application and extension of the OI. In the variational method, a cost function is minimized to get the optimal gain and optimal analysis. The idea of the 3D VAR has been extended to 4D VAR using a dynamical model to minimize the cost function which includes the time integration during the analysis-prediction cycle. The 4D VAR is equivalent to Kalman filter within the interval of the analysis-prediction cycle. The Kalman filter can accumulate all information in the past into one model, that is a unique point compared with the 4D VAR. This assimilation method will be described in the class of Dynamic Meteorology.

END

○ Excercise： (ubuntu : icho)

Log into ubuntu (icho) by using VNC Viewer.
Copy a subdirectory named kalman and bgcm from Tanaka's home

pwd : present working directory, /home/student-number

ls -la : list the directory,

cp -r /home/tanaka.hiroshi.fw/kalman . : copy all directory

cp -r /home/tanaka.hiroshi.fw/bgcm . : copy all directory

tcsh : tcshell activate

cd kalman : change directory to kalman

more 000readme : read me instructions

more kalman-code.f : Daley (1991)

csh kalman.sh

csh plot0.sh plot of data and obs-error

csh plot1.sh plot of obs-err and fct-error

csh plot2.sh plot of fct-error and anal-error

csh plot3.sh plot of obs, fst, and anl Coval

IOBS-1,DATA(3,IOBS),ST(3,IOBS),SF(3),SA(3),PF(3,3),PA(3,3),RR(3,3)

cd : return to top directory

cd bgcm : change directory to barotropic-Smodel

there are many subdirectories: source grads wave3 wave6 kal-smodel

Follow the instruction in 000readme there.

END