Projects 1, 2 and 3
Nonlinear model dynamics: Lorenz model

In the course, we have only considered linear model dynamics. In reality, we will be dealing with nonlinear models. What kind of complication does this pose to our data assimilation schemes? To find out, in these projects, you will implement a data assimilation scheme (extended Kalman Filter (ExKF), ensemble Kalman filter (EnKF) or 4DVAR) for a highly nonlinear model: Lorenz' three component, highly truncated model of convection that displays chaotic behaviour. If you plot the trajectory of model states in phase space, you get the famous butterfly picture of the attractor.

The nonlinear system of ordinary differential equations is

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y-x), \\ \frac{dy}{dt} &= \rho x - y - xz, \\ \frac{dz}{dt} &= xy - \beta z, \end{aligned} \tag{1}$$

with the parameters:

$$\sigma = 10, \qquad \rho = 28 \qquad \beta = 8/3.$$

σ is the Prandtl number, ρ is a normalized Rayleigh number and β is a nondimensional wavenumber.

This model has been used as a prototype of nonlinear atmospheric models. It is small enough to be able to fully implement complex data assimilation scheme and its chaotic behaviour is challenging to all assimilation schemes. A good reference (and I believe the first to use the Lorenz model for data assimilation) is Miller et al. (1994) in the Journal of the Atmospheric Sciences (JAS Vol. 51, No. 8, p1037-1056). We will use the same data assimilation parameters that they use. The model time step is 0.01 (dimensionless time units). The model integration scheme is a 4th order Runge-Kutta scheme (see Numerical Recipes ch. 15.1 for more details). For testing purposes, try a time limit of [0,4.0]. For a challenge try assimilating data from [0,8.0].

All variables will be observed during an observation time. Thus, $\mathbf{H} = \mathbf{I}$. The default observation frequency is every 25 time steps. Allow the user to choose this at run time. The observation error is assumed normally distributed with mean 0 and variance 2. There is no correlation between variables for observation error. Thus $\mathbf{R} = (\sigma^r)^2\mathbf{I}$.

The initial state error is also assumed unbiased and Gaussian with variance 2. Again, there is assumed correlation between variables for the initial state error. Thus, $\mathbf{P}_0 = (\sigma_0)^2\mathbf{I}$.

You can assume no model error, $\mathbf{Q} = 0$, both when generating observations and when propagating the error covariances.

You will be provided with a MATLAB routine for the model dynamics, the tangent linear and adjoint models.

Project 4
Nonlinear models: Double well problem

The double well problem is governed by the following nonlinear scalar equation:

$$\frac{dx}{dt} = -4x(x^2 - 1). \tag{2}$$

It is easy to see that stationary solutions exist for $x = 0, 1, -1$. The solutions at $x = 1, -1$ are stable while that at $x = 0$ is unstable. Thus with stochastic forcing, the position (of say a marble) can follow one attractor or well for a long time, then suddenly shift to the other well. The challenge of the assimilation scheme will be to track these state transitions. Thus, for the true system, the model equation will have a stochastic forcing:

$$\frac{dx}{dt} = -4x(x^2 - 1) + \sigma db.$$

where db is $\mathcal{N}(0, 1)$ and σ=0.24. This simple model was used to examine the effectiveness of data assimilation schemes for highly nonlinear problems by Miller et al. (1994).

The Tangent Linear Model (TLM) is very simply:

$$\frac{d\delta x}{dt} = [-4x(x^2 - 1)]\delta x.$$

Because the TLM is a scalar, the transpose or adjoint is the same thing. Thus the adjoint model is the same as the TLM. Or you can say that the TLM is self-adjoint.

The forecast model (and its tangent linear or adjoint models) will not include forcing. The forcing of the truth represents random, unknown errors of the forecast model. The forecast model and its derivatives are deterministic.

For this system, you will develop an extended Kalman filter, and a 4DVAR system. If time permits, you should also consider implementing an ensemble Kalman Filter.

Since there is only 1 variable to observed, $\mathbf{H} = \mathbf{I}$. The default observation frequency is every 20 time steps. Allow the user to choose this at run time. The observation error is assumed normally distributed with mean 0 and variance 0.01. Thus $\mathbf{R} = (\sigma^r)^2\mathbf{I} = 0.01$. The initial state error is also assumed unbiased and Gaussian with variance $\mathbf{P}_0 = (\sigma_0)^2\mathbf{I} = 0.01$. You can assume model error, $\mathbf{Q} = \mathbf{P}_0$.

You will be provided with a MATLAB routine for the model dynamics and the Tangent Linear Model (TLM).

Projects 5, 6 and 7
40-component Lorenz model

The following toy model was defined by Lorenz (1995) as a simple representation of the atmosphere. There is a single variable defined on a 1-D spatial grid with K points. The following equation governs the evolution of X at the k^{th} grid point:

$$\frac{dX_k}{dt} = -X_{k-2}X_{k-1} + X_{k-1}X_{k+1} - X_k + F \tag{3}$$

where F is a constant forcing. Although this is a very simple model compared to real weather forecast models, there are some similarities between the two. The quadratic terms are like the advection terms in weather forecast models. The linear term is like an addition diffusion term. The forcing term can represent real forcings such as sea surface termperature, physical forcings such as convection, or radiation.

The Tangent Linear Model (TLM) is given by

$$\frac{d\delta X_k}{dt} = -\delta X_{k-2}X_{k-1} - X_{k-2}\delta X_{k-1} + \delta X_{k-1}X_{k+1} + X_{k-1}\delta X_{k+1} - \delta X_k. \tag{4}$$

The Adjoint model was derived line-by-line.

Imagine that the grid is defined on a latitude circle and X is a variable such as temperature. Use $K = 36$ and F=8. The time step is 0.05 units or 6 hours. When running the model only, you can integrate forward for 14400 time steps or 10 years. With the assimilation, everything slows down a bit, but try to assimilate for a period of 240 days. When testing your scheme, reduce the number of gridpoints to speed up the code. Allow observations to vary in space and in time at regular intervals. (For a challenge you could try irregular observation locations.) Set the observation and initial state error standard deviations to 1. You should of course play with these values to test your scheme.

When developing any assimilation scheme, the first step is to understand the model. To do this, explore the model's parameter space, i.e. run the model changing the parameters (here F) to see how the model behaves. Here try F=0.2, 2.0, 3.0, 8.0. You will see that the model is stable for small F, periodic for medium F and chaotic for large F. Of course, the challenge is to use a large F, but for testing your assimilation scheme you may also want to try other values. Also, to see how nonlinear the model is, try running it twice. The first run is a control run. The second run has the same initial condition as the first but with a small random perturbation added to it (say 10%). By seeing how long the two solutions stay close for different model parameters, you can learn more about the nonlinearity of the model.

An interesting aspect of this model (and the shallow water model) is that there is a spatial dimension. Therefore, you can see how observations influence spreads in space.

Projects 8, 9, 10 and 11
1D linear shallow water model

The shallow water model is often used as a prototype for atmospheric models for testing data assimilation or initialization schemes. In addition, to relevance for oceanographic applications, the primitive equations when decomposed into Normal modes result in a series of shallow water equations for each vertical mode (see Daley 1991 for details). Thus understanding how assimilation schemes work with this model is a first step towards understanding the more complex weather forecasting models. Although the model is linear, the challenge will be dealing with the multivariate state (with variables, u, v and height or geopotential) and the spatial dimension.

The goal will be to develop one of the following data assimilation schemes: (1) KF, (2) Ensemble KF, (3) OI, (4) 4DVAR.

The Hinkelmann-Phillips model is a one dimensional linear shallow water model with the following governing equations

$$\frac{\partial u}{\partial t} + \bar{U}\frac{\partial u}{\partial x} - fv + \frac{\partial \phi}{\partial x} = 0, \tag{5}$$

$$\frac{\partial v}{\partial t} + \bar{U}\frac{\partial v}{\partial x} + fu = 0, \tag{6}$$

$$\frac{\partial \phi}{\partial t} + \bar{U}\frac{\partial \phi}{\partial x} - f\bar{U}v + \Phi_0\frac{\partial u}{\partial x} = 0. \tag{7}$$

where Φ_0 is the (constant) mean depth of fluid, $\bar{U}$ is (constant) basic zonal flow. f is the (constant) Coriolis parameter. All variables in the above equations are independent of y. The model grid is periodic (say a latitude circle) with 80 grid points, or a grid spacing of 4.5 degrees. The time step is 6 hours. The model integration is for 30 time steps, but when an assimilation scheme is added, the program will slow down considerably and you may want to reduce the integration length or increase the grid spacing. However, be careful to obey the CFL criterion $\Delta t < \Delta x/\bar{U}$ to avoid computational instability of the model. In fact, to be safe, make sure that $\Delta t < 0.1\Delta x/\bar{U}$.

Because the model is linear, the Tangent Linear Model is identical to the original model. The adjoint model is then the adjoint of the original linear model.

Because the model is linear, the difficulty here is the multivariate aspect. Try to assimilate observations with a regular spatial and temporal distribution. Since ϕ is O(1) but the winds are O(10), try an observation and background error standard deviations of 0.1 for ϕ and 1 for u and v. You can choose which variable(s) to assimilate. You could also try to introduce some nonlinearity in this problem but assuming observations of wind speed ($\sqrt{u^2 + v^2}$). Another possibility is to try linear observations such as averaged or mean wind. In this case, the observation is similar to say assimilating total column ozone in the sense that the observation is a weighted sum of a model variable over a number of gridpoints.

SUGGESTIONS

To create the assimilation scheme, here are some suggested steps.

1. First understand your model but running it for various initial conditions and exploring the parameter space. Also, if you have a nonlinear model, for what time range is the Tangent Linear Model valid? You can determine this by comparing the difference between two model runs with the evolution of the difference obtained with the TLM.

2. Start with the code for the passive tracer equation: kf.m or var4d.m. Then make the necessary modifications.

3. Set the true initial conditions.

4. Set the initial state error covariance matrix, $\mathbf{P}_0$. For 4DVAR, $\mathbf{P}_0 = \mathbf{P}^b$.

5. Define the initial state by perturbing the truth by $\mathbf{P}_0$.

6. Define the model error covariance matrix.

7. Define the observing network by defining $\mathbf{H}$ and $\mathbf{R}$ and setting the observing frequency.

8. For the ExKF or EnKF: In a time loop, update the state and covariance matrix. Then, if observations are available, produce an analysis and its error covariance matrix. For 4DVAR: write a subroutine that provides the cost given an arbitrary model state. Linearize this to get the subroutine for the gradient. Test the gradient routine. Choose a packaged minimization routine, or modify the Newton's method for nonquadratic cost functions by making it iterative.

9. Plot results in at least 2 forms. First plot the truth, analysis and observations as a function of time for x, y and z. Plot the analysis error variance as a function of time for each variable separately, or for all variables together.

Test your scheme and see that it works. Note that if you have a highly nonlinear model, your scheme may not work well. Your mark is not based on the success of your scheme but in how well you attempt to implement your scheme and in how you understand and diagnose what is happening. The plots will be helpful for debugging. Try experimenting with different observation errors, frequencies, initial state errors, assimilation lengths, etc. Describe what you/we can learn from your various experiments.

Notes: The ExKF and EnKF are the same except for the propagation of the error covariance matrices in time. A good reference for the ensemble KF is Evensen (2001). Copies of Evensen (2001) and of Miller et al. (1994) are available in pdf form on the course website.

REFERENCES

1. Daley, 1991: Atmospheric Data Analysis. *Cambridge Univ. Press*, 457 pp.

2. Evensen, Geir, 2001: Sequential data assimilation for Nonlinear Dynamics: The Ensemble Kalman Filter. In *Ocean Forecasting: Conceptual basis and applications*, N. Pinardi and J. D. Woods, ed., Springer-Verlag Berlin Heidelberg.

3. Lorenz, E. N., 1995: Predictability - A problem partly solved. *ECMWF seminar on predictability*, 4-8 Sept. 1995. Copies availabel from me.

4. Miller, R. N., M. Ghil, and F. Gauthiez, 1994: Advanced data assimilation in strongly nonlinear dynamical systems, *J. Atmos. Sci.*, **51**(8):1037-1056.