Experiments with variational DA in L96

NCEO Intensive Course on Data Assimilation

University of Reading, 2022

The objective of this practical is to perform state estimation using 3DVar and 4DVar in the Lorenz 1996 system. Furthermore, some methods to generate climatological covariances are explored.

1 Description of the code

1.1 Files

These are the python files used in this part of the activity:

- *ControlL96Var.py*. This is the control file - it is one of two that you will be running and modifying.
- *ControlsL96covs.py*: This file explores different ways to generate a climatological background error covariance matrix using the Lorenz 1996 system. You will be running and modifying it.
- *L96model.py*. This file contains the instructions for running the L96 model.
- *L96misc.py*. This file generates different observation operators, creates the observations, and generates a simple background error covariance matrix.
- *L96var.py*. This file contains the routines to perform 3D-Var and strong constraint 4D-Var. This includes computing the tangent linear model and transition matrices.
- *L96plots.py*. This file has instructions for producing different output plots.

1.2 Instructions

You will run different sections of the file *ControlL96Var.py*. These are numbered as comments of the file (recall that in python # is used for comments). To run **only** a section of a file you can highlight the desired lines of code with the mouse, and then press F9.

- The first lines of the file import the different packages that the file uses: numpy, matplotlib, and the functions we have created for this activity.
- ***Section 1: The Nature Run***. This section generates the **nature** run of the experiment, i.e. what we consider to be the true system trajectory. You can change the initial conditions for the nature run, the final time (the model time step is fixed at 0.025 time units), and the background initial guess from which the assimilation will start. For speed of computations and to display figures in an easier manner, we have selected $N_x = 12$ variables. This model can be run from a given initial condition, but the default is to spin it up from a perturbation around the unstable fixed point of

the system. You will get a Hovmoller diagram (a contour plot showing the time evolution of the different variables in a circle of latitude), as well as a figure with $N_x = 12$ panels.

- ***Section 2: The observations***. This section generates the observations, the observation error covariance matrix and the observation operator. You can select to observe different variables with three options: $'all'$ corresponds to observing all variables, $'1010'$ corresponds to observing every other variable, and $'landsea'$ corresponds to observing only half of the domain (a challenging setting). Different choices will create the appropriate observation error covariance matrix. The $\mathbf{R}$ matrix is designed to be diagonal (common assumption), but you can adjust the observation error variance. You can also choose the frequency of the observations (in number of model steps). In this model the time auto-correlation is quite small, so we recommend experimenting with observation frequencies no larger than 4 time steps.
- ***Section 3: Data assimilation***. This short section creates the climatological $\mathbf{B}$ matrix for this model. There is a scaling tuning parameter that can be varied depending on the observation frequency.
- ***Section 3a: 3DVar***. This section runs the 3D-Var assimilation. It also computes the background and analysis RMSE with respect to the truth. The trajectories and the RMSE's are plotted.
- ***Section 3b: 4DVar***. This section runs the 4D-Var assimilation. You can select the length of the assimilation window, which is expressed in terms of the number of observation times per window. It also computes the background and analysis RMSE with respect to the truth. The trajectories and the RMSE's are plotted.

2 Experiments

2.1 State estimation

1. Run the Lorenz 1996 model to a maximum time $t = 4$ and plot the trajectory. This nature run will be the basis of our experiments. Generate a synthetic observational set.
2. 3DVar. In this case we will fix the observational standard deviation to $\sqrt{2}$. The following parameters can vary: the frequency of observations in time and the density of observations in space. Do experiments with the combinations indicated in the following table:

Obs frequency	Obs density
2	all
””	1010
””	landsea
4	all
””	1010
””	landsea

   What can you say this time about the different observational densities and the observation frequency?

3. 4DVar. As before we vary the length of the assimilation window. Try the following number of observational times per window: $\{1, 2, 4\}$ for some of the configurations of the previous exercise. What is the influence of the length of the assimilation window in the performance of the DA? How does this change with respect to the observational network?

2.2 Generation of climatological background covariances

So far the code has provided a $\mathbf{B}_c$ matrix for us, i.e. a climatological error covariance matrix. In this section we will explore how to generate it in two ways.

1. The first way is to look at the dynamics of the systems alone, with no knowledge about the observatioal process. We will follow Polavaru et al (2005) for this exercise with two implementations. The first (and more primitive one) one looks at long runs of a model and samples the state at different time periods. This is labeled *simple* in the exercises. A more sofisticated method, the so-called *Canadian quick method* looks at the growth of perturbations over a given forecast time. Perform experiments with the following options:

Method	Sampling period
Simple	1
Simple	2
Simple	4
Canadian	1
Canadian	2
Canadian	4

   For each case: Can you interpret the 4 plots that result in each case? What happens to the features in them as you increase the sampling period?

   How do the two methods compare? Do you understand the difference between them?

2. The observational set-up has an impact in the resulting $\mathbf{B}_c$. Think of a real set-up: observed grid points will have smaller forecast errors than unobserved ones. Now we use a diagnostic method described in Yang et al (2006) and used in Kalnay et al (2007). The method does the following: it starts with a trial $\mathbf{B}$ e.g. $\mathbf{I}$ and performs a series of variational experiments. From the results it diagnoses an updated $\mathbf{B}$ which is fed back into the system. This is iterated until convergence. In this experiment you have to vary:

obs network
'all'
'1010'
'landsea'

   In this experiment we are diagnosing $\mathbf{B}$ for a 3DVar system. How is this diagnosed at each step? How would it be different in 4DVar? What do the different panels in the resulting figure mean? Also, you will notice some weights α in the code, what do they mean?