Description of code for estimating correlated error statistics in an ensemble transform Kalman filter

J. A. Waller[1]

[1]School of Mathematical and Physical Sciences, University of Reading, Reading, Berkshire, United Kingdom

July 15, 2014

1 Background

Data assimilation techniques combine physical observations of a dynamical system with a model prediction of the state of the system, known as the background state, weighted by their respective error covariance matrices, to obtain a best estimate of the current state of the system, known as the analysis. We denote here $\mathbf{x}_n^f \in \mathbb{R}^{Nm}$ as the background state, $\mathbf{x}_n^a \in \mathbb{R}^{Nm}$ as the analysis and $\mathbf{y}_n \in \mathbb{R}^{Np}$ as the observation at time t_n, and the covariance matrices of the errors in the background and observations by $\mathbf{R}_n$ and $\mathbf{P}_n^f$. The background at the next step of the assimilation process is generated by evolving the (possibly nonlinear) dynamical model of the system $\mathcal{M}_n$ forward from the analysis.

Until recently the observation error covariance matrix has been assumed uncorrelated. However, it has been shown that the error is correlated Waller et al. [2013] and that the inclusion of the correlated errors in the assimilation leads to: a more accurate analysis, the inclusion of more observation information content and an improvement in the NWP skill score Stewart et al. [2013], Weston et al. [2013].

This code describes a method developed in Waller [2013] and Waller et al. [2014] that allows spatially correlated and time-dependent observation error to be diagnosed and incorporated in an ensemble data assimilation system. The method combines an ensemble transform Kalman filter with a method that uses statistical averages of background and analysis innovations to provide an estimate of the observation error covariance matrix.

2 Method

2.1 The DBCP diagnostic

The DBCP diagnostic described in Desroziers et al. [2005] shows that the observation error covariance matrix can be calculated using

$$\mathbf{R} \approx E[(\mathbf{y} - \mathcal{H}(\mathbf{x}^a))(\mathbf{y} - \mathcal{H}(\mathbf{x}^f))^T]. \tag{1}$$

This is valid if the observation and forecast errors used in the gain matrix,

$$\mathbf{K} = \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}, \tag{2}$$

to calculate the analysis, are correct. However, provided that the correlation length-scales in $\mathbf{P}^f$ and $\mathbf{R}$ are sufficiently different, it has been shown that a reasonable estimate of $\mathbf{R}$ can be obtained even if the $\mathbf{R}$ and $\mathbf{P}^f$ used in $\mathbf{K}$ are not correctly specified. It has also been shown that the method can be iterated to estimate $\mathbf{R}$ [Mènard et al., 2009, Desroziers et al., 2009].

2.2 The ensemble transform Kalman filter with R estimation

The ETKFR uses the ETKF and the DBCP diagnostic to estimate a possibly non-uniform time varying observation error covariance matrix. After the filter is initialised the filter is split into two stages. The first is a spin-up stage that runs for a predetermined number of steps, N^s, and is an application of the standard ensemble transform Kalman filter. In the second stage at each assimilation step the observation error covariance matrix is updated using the DBCP diagnostic. We now present in detail the method that we have developed. Here the observation operator, $\mathbf{H}$, is chosen to be linear, but the method could be extended to account for a non-linear observation operator $\mathcal{H}$ (e.g. Evensen [2003]).

Initialisation - Begin with an initial ensemble $\{\mathbf{x}_0^i\}$ for $i = 1 \dots N$ at time $t = 0$ that has an associated initial covariance matrix $\mathbf{P}_0^f$. Also assume an initial estimate of the observation error covariance matrix $\mathbf{R}_0$; it is possible that this could just consist of the instrument error.

Step 1 - The first step is to use the full non-linear model, $\mathcal{M}_n$, to forecast each ensemble member, $\mathbf{x}_{n+1}^{f,i} = \mathcal{M}_n(\mathbf{x}_n^{a,i})$.

Step 2 - Calculate the ensemble mean,

$$\bar{\mathbf{x}}_n = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_n^i, \tag{3}$$

and covariance

$$\mathbf{P}_n = \mathbf{X}'_n{\mathbf{X}'_n}^T, \tag{4}$$

where $\mathbf{X}'_n$ is the matrix containing the ensemble perturbations.

Step 3 - Using the ensemble mean and the observations at time t_n, calculate and store $\mathbf{d}_n^b = \mathbf{y}_n - \mathbf{H}\bar{\mathbf{x}}_n^f$.

Step 4 - Update the ensemble mean using,

$$\bar{\mathbf{x}}_n^a = \bar{\mathbf{x}}_n^f + \mathbf{K}_n(\mathbf{y}_n - \mathbf{H}_n\bar{\mathbf{x}}_n^f), \tag{5}$$

where $\mathbf{K}_n$ is the Kalman gain.

Step 5 - Following Bishop et al. [2001] and Livings et al. [2008] calculate the analysis perturbations using

$${\mathbf{X}'}_n^a = {\mathbf{X}'}_n^f\boldsymbol{\Upsilon}_n, \tag{6}$$

where $\boldsymbol{\Upsilon}_n$ is the symmetric square root of $(\mathbf{I} - {\mathbf{Y}'}_n^{fT}\mathbf{S}_n^{-1}{\mathbf{Y}'}_n^f)$

Step 6 - The analysis mean is then used to calculate the analysis innovations, $\mathbf{d}_n^a = \mathbf{y}_n - \mathbf{H}\bar{\mathbf{x}}_n^a$.

Step 7 - If $n > N^s$, where N^s is the specified sample size, update $\mathbf{R}$ using

$$\mathbf{R}_{n+1} = \frac{1}{N^s - 1} \sum_{k=n-N^s+1}^{k=n} \mathbf{d}_k^a \mathbf{d}_k^{b\,T}. \tag{7}$$

Then symmetrise the matrix, $\mathbf{R}_{n+1} = \frac{1}{2}(\mathbf{R}_{n+1} + \mathbf{R}_{n+1}^T)$. Otherwise keep $\mathbf{R}_{n+1} = \mathbf{R}_0$.

Many of the steps in the proposed method are identical to the ETKF. Step 7, along with the storage of the background and analysis innovations in steps 3 and 6, are the additions to the ETKF that provide the estimate of the observation error covariance matrix.

In practice the number of samples available will be limited and therefore the estimated observation error covariance matrix will not be full rank. In this case it may be necessary to apply some form of regularisation to the estimated matrix.

We now describe the code that implements this method.

3 Code

The package Matlab is required to run the code. The code to run the ETKFR consists of a set of three functions. The main function that must be called is `ETKFR` and this in turn calls the functions `enkf_forecast` which forecasts the ensemble members and `etkf_analysis` which performs the analysis step. We also provide three additional codes that allow an example using the ETKFR to be run; these codes are a forecast model `KS` for the Kuromoto-Sivashinsky equations (with code adapted from Kassam and Trefethen [2005]) and a regularisation method `Homogeneous` that is used to regularise the estimated matrix.

We now give specific details of each function.

3.1 ETKFR

`[dob,doa,MEAN_XF,MEAN_XA] = ETKFR(model,t,kl,xf0,y,H,AllR,s,regularisation)` . Runs the ETKF assimilation and forecast; it calculates and stores the background and analysis innovations and uses them to calculate the correlated error covariance matrix $\mathbf{R}$.

Inputs:

- `model` - The forecast model. Should have form `x1 = model(t0, t1, x0)`. Inputs:
  - `t0` - Initial time.
  - `t1` - Vector of times when model solution required.
  - `x0` - State vector at initial time.

  Outputs:
  - `x1` - State vectors at times supplied in `t1`.
- `t` - A vector of forecast times.
- `kl` - A vector (length p) of forecast times at which the observations are available.

- `xf0` - The initial forecast ensemble as a matrix of column vectors ($N^m \times N$).
- `y` - Array ($p \times N^p$) of observation vectors.
- `H` - Observation operator ($N^p \times N^m$).
- `AllR` - A 3D array ($N^p \times N^p \times N^s$) containing the observation error covariance matrix to be used at each step of the assimilation (before the estimated matrix is used).
- `Ns` - The number of samples that will be used to calculate the observation error covariance matrix.
- `regularisation` - The method for regularising the estimated observation error covariance matrix. Should have the form `[RegR] = regularisation(R)`. Inputs:
  - `R` - matrix to be regularised.

  Outputs:
  - `RegR` - Regularised matrix.

Outputs:

- `dob` - The matrix ($p \times N^p$) of background innovations.
- `doa` - The matrix ($p \times N^p$) of analysis innovations.
- `MEAN_XF`- The forecast mean ($N^m \times p$) at each assimilation step.
- `MEAN_XA`- The analysis mean ($N^m \times p$) at each assimilation step.
- `EstR` - The estimated error covariance ($N^m \times p$) for assimilation steps after the first N^s assimilations.

3.2 enkf_forecast

`[xf] = enkf_forecast(model, t0, t1, xa)` Performs the forecast step of the ETKF.

Inputs:

- `model` - The forecast model.
- `t0` - Initial time.
- `t1` - Vector of times when model solution required.
- `xa` - Analysis state vectors to be forecast for each ensemble member.

Outputs:

- `xf` - Forecast state vectors for each ensemble member.

3.3 etfk_analysis

`[xa,mean_xf,mean_xa] = etkf_analysis(xf, y, H, rR)`

Inputs:

- `xf` - Forecast state vectors for each ensemble member.
- `y` - Vector $(1 \times N^p)$ of observations for the assimilation step
- `H` - Observation operator, a matrix of size $N^p \times N^m$
- `rR` - Analysis state vectors to be forecast for each ensemble member.

Outputs:

- `xa` - Forecast state vectors to be forecast for each ensemble member.
- `mean_xf` - Forecast state vectors to be forecast for each ensemble member.
- `mean_xa` - Forecast state vectors to be forecast for each ensemble member.

3.4 KS

`u = KS(t0, t1, x0)` The forecast model for the Kuromoto Sivashinsky equation.

Inputs:

- `t0` - Initial time.
- `t1` - Vector of times when model solution required.
- `x0` - State vector at initial time.

Outputs:

- `x1` - State vectors at times supplied in `t1`.

3.5 Homogeneous

`[RegR] = Homogeneous(R).` Code to regularise the estimated observation error matrix by making it isotropic and homogeneous.

Inputs:

- `R` - matrix to be regularised.

Outputs:

- `RegR` - Regularised matrix.

4 Example

Here we demonstrate the use of the code which is done in four stages:

- Enter Matlab and ensure that the Matlab directory is pointed at the folder containing the coded functions.
- Define the values to be used for t, kl, xf0, y, H, AllR and s .
- Call the function `ETKFR`

- Plot the results.

For this example we provide a matlab .mat file ETKFRexample that contains all the input matrices and vectors required. In this example we are forecasting and assimilating observations into the Kuromoto Sivashinsky model. This model has a solution on a periodic domain with $N^m = 256$ grid points and the assimilation is run until a final time of $T = 5000$. Direct observations are available every 10 time units and at 64 equally spaced points around the domain. Observations have an error from the distribution $\mathcal{N}(0, \mathbf{R})$ where the correlations in the observation error are the combination of a diagonal error with variance 0.1 and a correlated error determined by a SOAR function that is slowly varying in time, also with variance 0.1. The ETKFR is run with $N = 500$ ensemble members and initially the observation error matrix is assumed diagonal with an error variance of 0.1. We choose to estimate $\mathbf{R}$ using 250 samples. To run the example first load the .mat file,

```
>> load ETKFRexample.mat
```

then call the ETKFR function using the KS model and the homogeneous regularisation,

```
>> [dob,doa,MEAN_XF,MEAN_XA,EstR]=ETKFR(@KS,t,kl,xf0,y,H,AllR,s,@Homogeneous)
```

the code will then run (note this is not quick!) and the background and analysis innovations along with the mean background and analysis values and the estimated R matrices will be produced.

The .mat file also contains a matrix that represents the 'true' state and an array that contains the true observation error covariance matrices. This allows the results of the assimilation and observation error estimation to be plotted.

Plotting the forecast, analysis, observations and truth at the final time results in Figure 1.

```
>> figure
>> plot(xM,TrueState(:,20001))
>> hold on
>> plot(xM(4:4:256),y(500,:),'x')
>> plot(xM,MEAN_XF(:,500),'k')
>> plot(xM,MEAN_XA(:,500),'r')
```

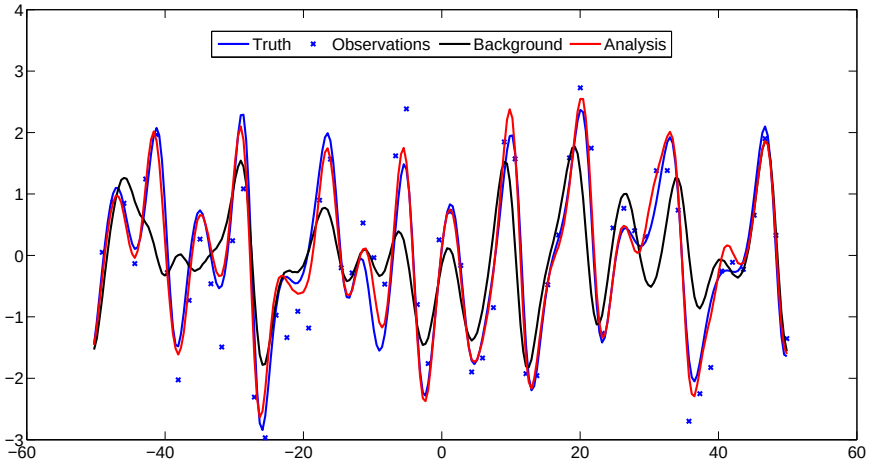

Figure 1 – True (blue), observations (crosses), background (black) and analysis (red) states of the Kuromoto Sivashinsky equation at the final assimilation time.

From Figure 1 we see that the analysis follows the truth more closely than the background. It also also possible to plot the estimated covariance matrices,

```
>> figure
>> subplot(2,1,1)
>> plot(xM(1:4:256),TrueR(33,:,251),'b')
>> hold on
>> plot(xM(1:4:256),EstR(33,:,251),'r')
>> subplot(2,1,2)
>> plot(xM(1:4:256),TrueR(33,:,500),'b')
>> hold on
>> plot(xM(1:4:256),EstR(33,:,500),'r')
```

In Figure 2 we plot the true and estimated matrices after the first estimation (250th assimilation step) and at the final estimation (500th assimilation step.

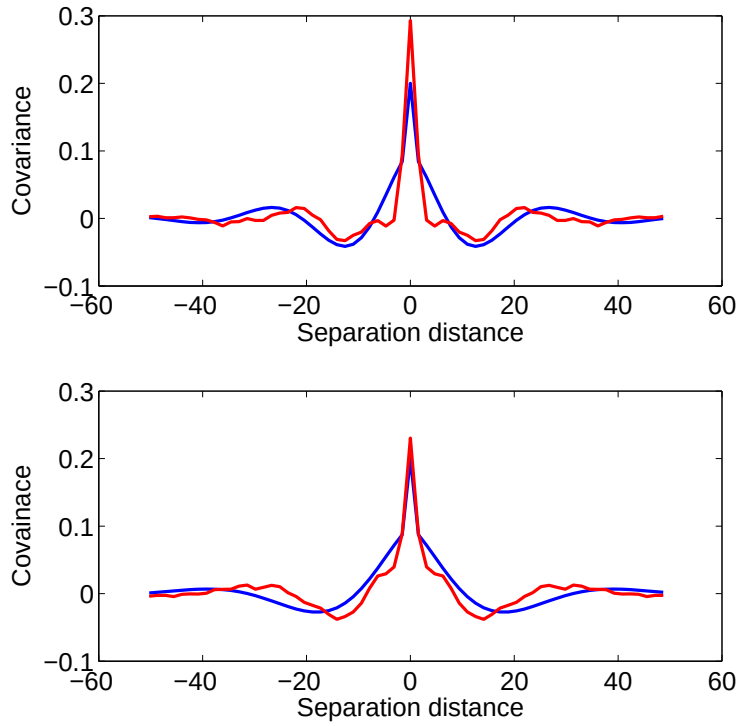

Figure 2 – Rows of the true (blue) and estimated (red) covariance matrices after the first 250 assimilation steps (top panel) and after the final assimilation step (bottom panel).

We see that the estimated covariances capture approximately the shape and variance of the true correlation function.

References

C. Bishop, B. Etherton, and S. Majumdar. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Monthly Weather Review*, 129:420–436, 2001.

G. Desroziers, L. Berre, B. Chapnik, and P. Poli. Diagnosis of observation, background and analysis-error statistics in observation space. *Quarterly Journal of the Royal Meteorological Society*, 131: 3385–3396, 2005.

G. Desroziers, L. Berre, and B. Chapnik. Objective validation of data assimilation systems: diagnosing sub-optimality. *In Proceedings of ECMWF Workshop on diagnostics of data assimilation system performance, 15-17 June 2009*, 2009.

G. Evensen. The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, 53:343–367, 2003.

A. Kassam and L. Trefethen. Fourth-order time-stepping for stiff pdes. *SIAM J. Sci. Computing*, 25:1214–1233, 2005.

D. M. Livings, S. L. Dance, and N. K. Nichols. Unbiased ensemble square root filters. *Physica D*, 237:1021–1028, 2008.

R. Mènard, Y. Yang, and Y. Rochon. Convergence and stability of estimated error variances derived from assimilation residuals in observation space. *In Proceedings of ECMWF Workshop on diagnostics of data assimilation system performance, 15-17 June 2009*, 2009.

L. M. Stewart, S. L. Dance, and N. K. Nichols. Data assimilation with correlated observation errors: experiments with a 1-D shallow water model. *Tellus A*, 65, 2013.

J. A. Waller. *Using observations at different spatial scales in data assimilation for environmental prediction*. PhD thesis, University of Reading, Department of Mathematics and Statistics, 2013. http://www.reading.ac.uk/maths-and-stats/research/theses/maths-phdtheses.aspx.

J. A. Waller, S. L. Dance, A. S. Lawless, N. K. Nichols, and J. R. Eyre. Representativity error for temperature and humidity using the Met Office UKV model. *Quarterly Journal of the Royal Meteorological Society*, 2013. Early View. DOI: 10.1002/qj.2207.

J. A. Waller, S. L. Dance, A. S. Lawless, and N. K. Nichols. Estimating correlated observation error statistics using an ensemble transform Kalman filter. *Tellus A*, 2014. Accepted.

P. P. Weston, W. Bell, and J. R. Eyre. Accounting for correlated error in the assimilaiton of high resolution sounder data. *Quarterly Journal of the Royal Meteorological Society*, 2013. Early View. DOI: 10.1002/qj.2306.