Particle filters practical with the BV model

PA Browne
Email: p.browne@reading.ac.uk
Department of Meteorology, University of Reading,
RG6 6BB, UK

15 March 2016

In this practical we will investigate using different particle filters to assimilate data into the barotropic vorticity model. Recall that the model is solved in 2d on a square with periodic boundary conditions (i.e. a torus). There are 256 grid points in each direction, giving a total of 65,536 state variables.

The observations were created when you ran the version of the model that we consider the truth earlier in the course.

0 Getting started

Log into windows (somehow).

Once you are finally in - load an xterminal window from the following menus:

Start → All Programs → MobaXterm Personal Edition → MobaXterm Personal Edition

Click the button "Start local terminal" in the middle of the screen.

Log into ARCHER. Replace `USERNAME` below with the one you were given on a separate piece of paper.

```
:~> ssh -X USERNAME@login.archer.ac.uk
```

Enter your password. If a prompt box opens up, you can ignore it if you wish.

Move to your /work directory.

```
:~> cd /work/n02/n02/$USER/2016_dacourse
```

1 SIR filter

Set up the input files and submission scripts to run the SIR filter.

```
:~> ensemble sir 48
```

This will generate a submission file `pbs_jobscript` setup with 48 ensemble members.

Submit this to the queue on ARCHER using the command

```
:~> qsub -q course1 pbs_jobscript
```

You can watch the status of the queue with the command

```
:~> qstat
```

more specifically, just your own jobs in the queue can be shown with the command

```
:~> qstat -u $USER
```

Note the letter in the penultimate column. Q means queuing, R means running, E means ending.

To check the progress of the sequential method, run the following command.

```
:~> wc -l pf_out_00
```

The first number is one larger than the number of timesteps that have been completed. The model is set to run 1200 timesteps, so we have to wait for this number to reach 1201.

When the job has finished, we can now look at the results.

Use the python plotting routines from yesterday to analyse the results.

Another tool to analyse the ensemble is a rank histogram. You can plot a rank histogram with the command

```
:~> python python_histograms.py
```

Remember, if the rank histogram is flat, the truth is indistinguishable from any of the ensemble members. If the rank histogram is *hump shaped* then the ensemble is overdispersive. If the rank histogram is *U shaped* then the ensemble is underdispersive.

2 Equivalent weights particle filter

You are used to this now, set up the scripts to run the equivalent weights particle filter with the command

```
:~> ensemble ewpf 48
```

This will generate a submission file `pbs_jobscript` setup with 48 ensemble members.

Submit this to the queue on ARCHER using the command

```
:~> qsub -q course1 pbs_jobscript
```

When the job has finished, we can now look at the results.

There are a few parameters which you can play with in the equivalent weights particle filter scheme. These are the nudging factor, `nudgefac`, and the proportion of particles kept, `keep`, in the equivalent weights step.

Also, experiment with changing the number of ensemble members that you run.

As one last tool for analysing the results from the data assimilation, there is the tool `python_pdfs.py`. Recall that the mayor point of doing fully nonlinear data assimilation is to approximate the posterior pdf without assuming that it is Gaussian. Run the command

```
:~> python python_pdfs.py
```

and try to understand the results.