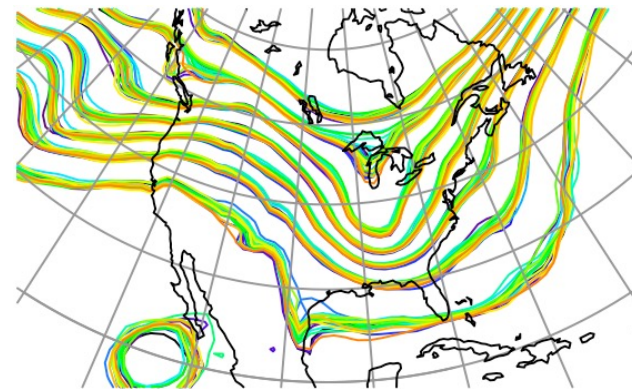


Data  
Assimilation  
Research  
Testbed



## DART Tutorial Section 18: Lost in Phase Space: The Challenge of Not Knowing the Truth.



©UCAR



The National Center for Atmospheric Research is sponsored by the National Science Foundation. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

NCAR | National Center for  
UCAR | Atmospheric Research

# Reality Strikes

In real applications, **the truth is unknown.**

All that we have are observations.

Having the truth available has been convenient,  
but also misleading.

Much less information is available from the observations.

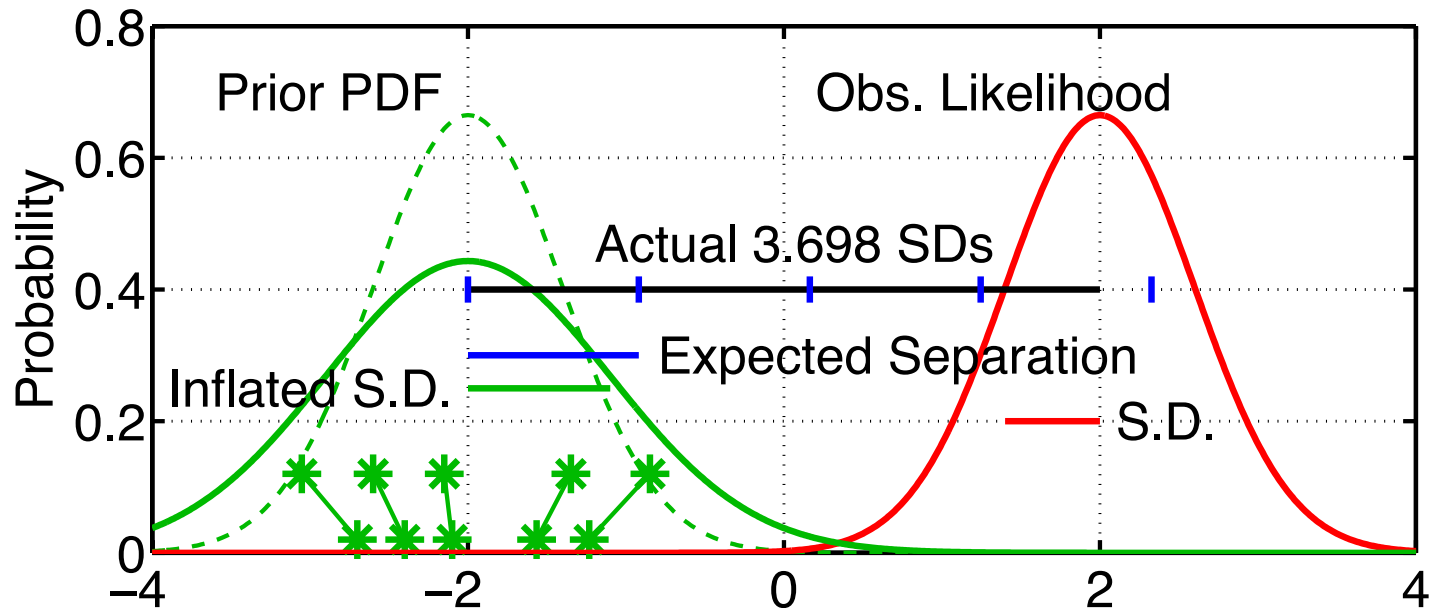
They are generally functions of the state variables.

They are always contaminated with observational errors.

# What to expect ...

Recall that

$$\text{Expected}(\text{prior\_mean} - \text{observation}) = \sqrt{\sigma_{\text{prior}}^2 + \sigma_{\text{obs}}^2}$$



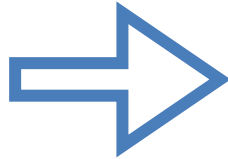
Error is dominated by observational noise if  $\sigma_{\text{obs}}^2 \gg \sigma_{\text{prior}}^2$

Suppose  $\sigma_{\text{obs}} = 1.0$ ,  $\sigma_{\text{prior}} = 0.1$ , then  $E(\text{RMS}) = 1.005$ .

Halving  $\sigma_{\text{prior}}$  to 0.05  $\Rightarrow E(\text{RMS}) = 1.001$ ; only a 0.4% reduction!

# First Observation-space diagnostics:

Whether or not to assimilate or reject observations based on their Expected Separation is controlled during **filter** based on namelist settings in *input.nml*.

If  $\left| \overline{y^p} - y^o \right| / \sqrt{\sigma_{prior}^2 + \sigma_{obs}^2} > \text{outlier\_threshold}$   Observation rejected!  
(DART QC ==7)

```
&quality_control_nml  
  input_qc_threshold      = 3.0  
  outlier_threshold      = -1.0  
/
```

**Negative value  
means USE ALL** 

The program **obs\_diag** post-processes *obs\_seq.final*, calculates metrics like RMSE, bias, ensemble spread, totalspread, # of observations used or rejected ... Start with the *lorenz\_96* model.

# Observation-space diagnostics

The observation sequence file is not in a particularly user-friendly format.

To aid in the evaluation and interpretation, a program named ***obs\_diag*** must be run to produce a netCDF file with results that can be plotted in a manner of your choosing. DART has Matlab functions/scripts that create high-quality graphics.

For up-to-date information on the latest, greatest diagnostics, go to:

[http://www.image.ucar.edu/DAReS/DART/DART2\\_Diagnostics.php#obs\\_diagnostics](http://www.image.ucar.edu/DAReS/DART/DART2_Diagnostics.php#obs_diagnostics)

```
&obs_diag_nml
  obs_sequence_name      = 'obs_seq.final',
  bin_width_days        = -1,
  bin_width_seconds     = -1,
  init_skip_days        = 0,
  init_skip_seconds     = 0,
  Nregions              = 3,
  trusted_obs           = 'null',
  lonlim1               = 0.00, 0.00, 0.50, -1.0
  lonlim2               = 1.01, 0.50, 1.01, -1.0
  reg_names             = 'whole', 'yin', 'yang', 'bogus',
  create_rank_histogram = .true.,
  outliers_in_histogram = .true.,
  use_zero_error_obs    = .false.,
  verbose               = .false.
/
```

**(Slightly different for 3D models.)**

# Observation-space diagnostics

The observation sequence file is not in a particularly user-friendly format. To aid in the evaluation and interpretation, a program named ***obs\_diag*** must be run to produce a netCDF file with results that can be plotted in a manner of your choosing. DART has Matlab functions/scripts that create high-quality graphics. For up-to-date information on the latest, greatest diagnostics, go to:

[http://www.image.ucar.edu/DAReS/DART/DART2\\_Diagnostics.php#obs\\_diagnostics](http://www.image.ucar.edu/DAReS/DART/DART2_Diagnostics.php#obs_diagnostics)

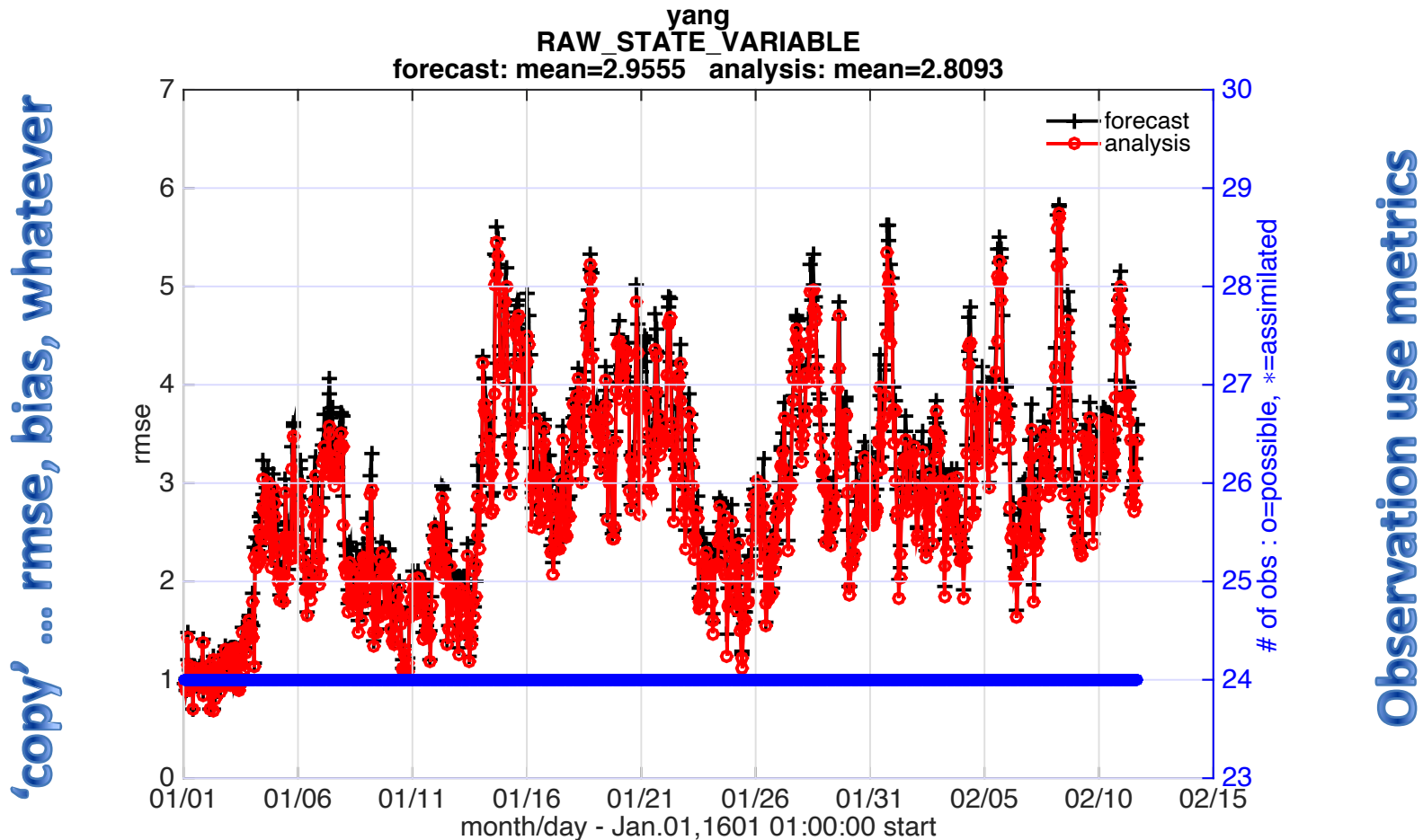
Here are a few of the Matlab functions available in *diagnostics/matlab*

- **plot\_rank\_histogram.m**
- **plot\_evolution.m**
- **plot\_rmse\_xxx\_evolution.m**
- **two\_experiments\_evolution.m**
- **plot\_profile.m**
- **plot\_bias\_xxx\_profile.m**
- **plot\_rmse\_xxx\_profile.m**
- **two\_experiments\_profile.m**

These work with ANY *obs\_seq.final* from ANY experiment with ANY model!

# Lorenz 96 observation diagnostic example

outlier\_threshold = -1.0



data file: /Users/thoar/svn/DART/clean\_lanai/models/lorenz\_96/work/obs\_diag\_output.nc

# First Observation-space diagnostics:

Try setting the rejection threshold to a small positive number and rerunning *filter*, and then rerunning *obs\_diag* on the new output file.

```
&filter_nml
...
obs_sequence_in_name      = "obs_seq.out"
obs_sequence_out_name    = "obs_seq.final"
...
/
```

Change to whatever you like.

```
&quality_control_nml
input_gc_threshold       = 3.0,
outlier_threshold        = -1.0,
/
```

Change to 3.0

Don't forget to rerun *filter*!

Don't forget to rerun *obs\_diag*!

Don't forget to use the right filename in *obs\_diag\_nml*!

This is potentially **DANGEROUS**, but useful.

Rejecting 'good' observations can lead to inflated estimate of quality.

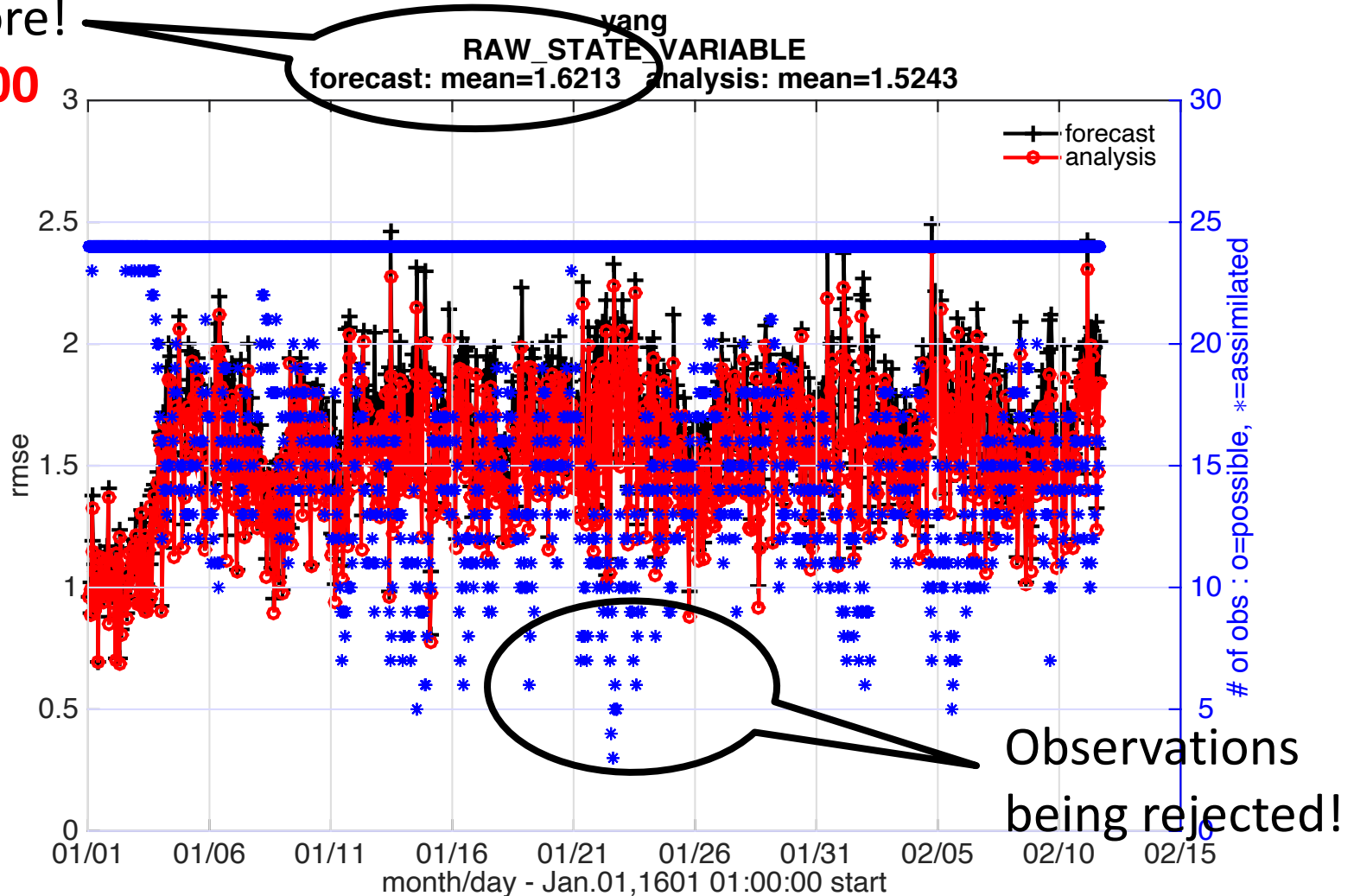


# First Observation-space diagnostics:

Lower RMSE  
than before!

**\$1,000,000**  
**question:**  
**Why?**

outlier\_threshold = 3.0

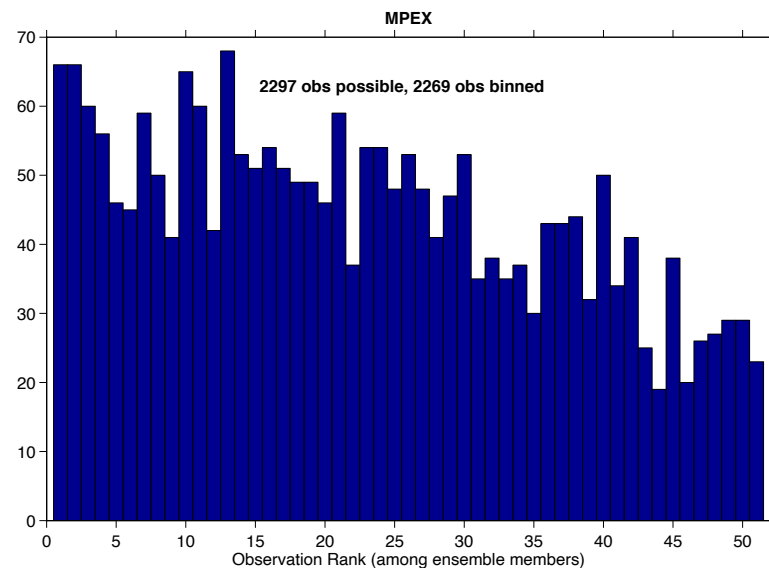
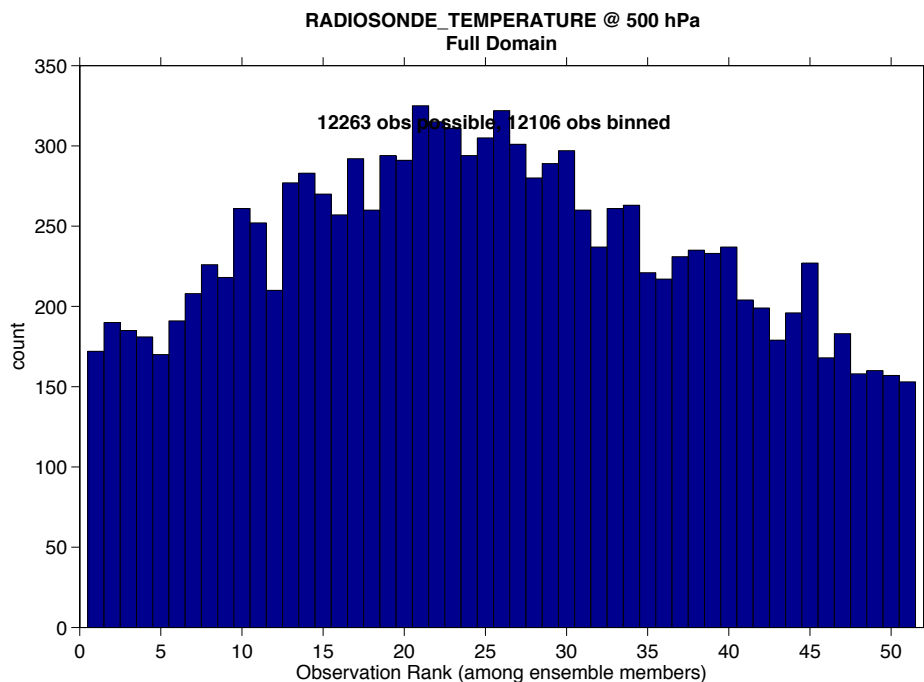


# Lorenz\_96 exercises:

- Pick a case that works relatively well and look at observation-space diagnostics.
- Pick a case that is similar, but clearly different, with physical-space diagnostics.
- See if you can detect the difference with observation-space diagnostics.
- Rerun ***obs\_diag*** with different *bin\_widths*.

# Observation-space diagnostics: rank histograms

```
>> fname = 'obs_diag_output.nc';  
>> timeindex = -1;  
>> varname = 'RADIOSONDE_TEMPERATURE';  
>> plot_rank_histogram(fname, timeindex, varname);
```



Results from WRF real-time forecasting.

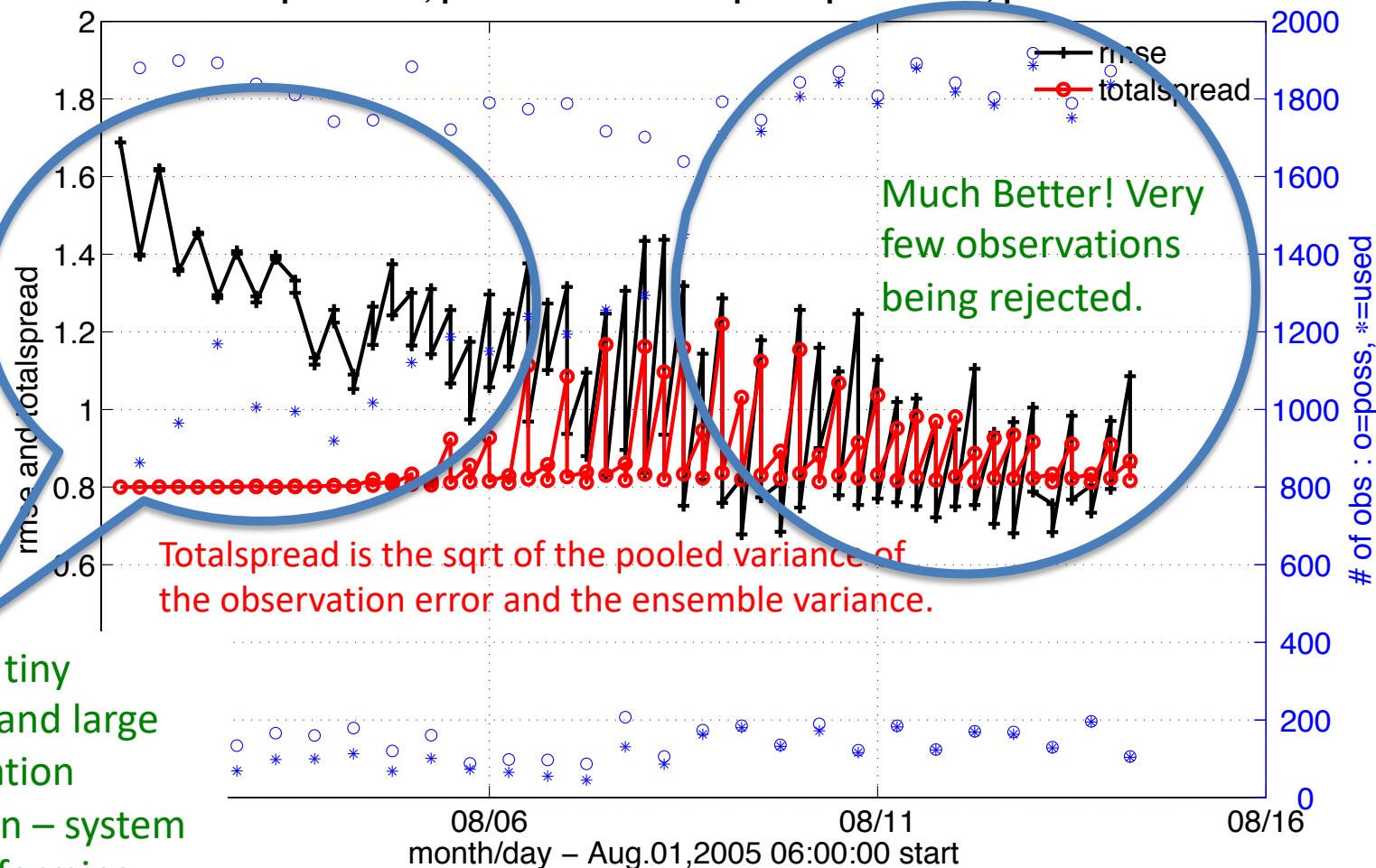
# Observation-space diagnostics: time evolution (by level)

plot\_rmse\_xxx\_evolution.m

plot\_evolution.m

Northern Hemisphere (20–80)  
RADIOSONDE TEMPERATURE @ 500 hPa

rmse pr=1.1971, po=0.98162    totalspread pr=0.91985, po=0.81559

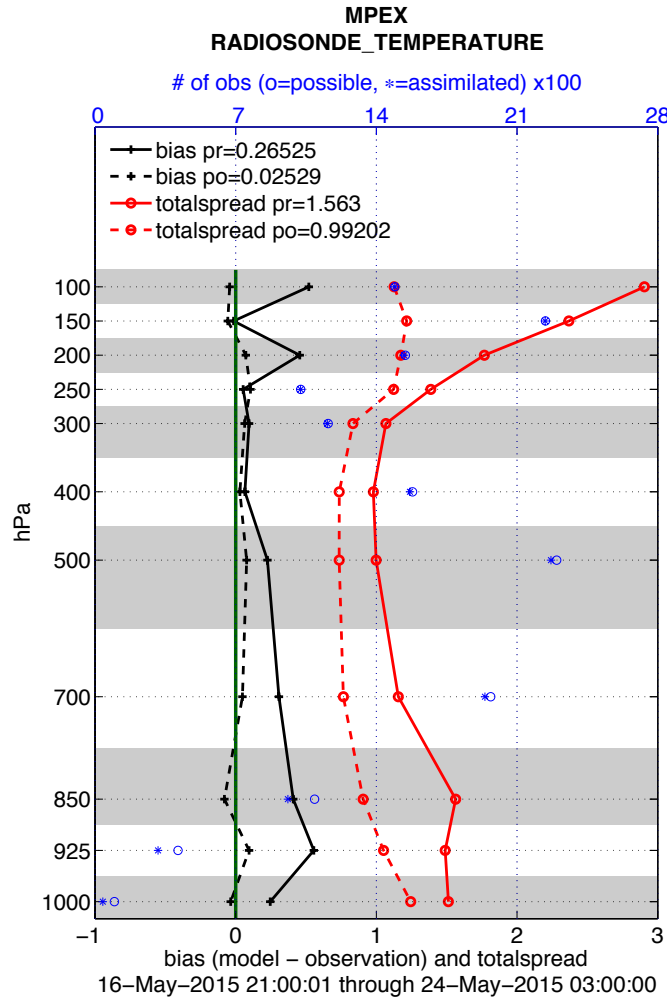


# Observation-space diagnostics: time-averaged profiles

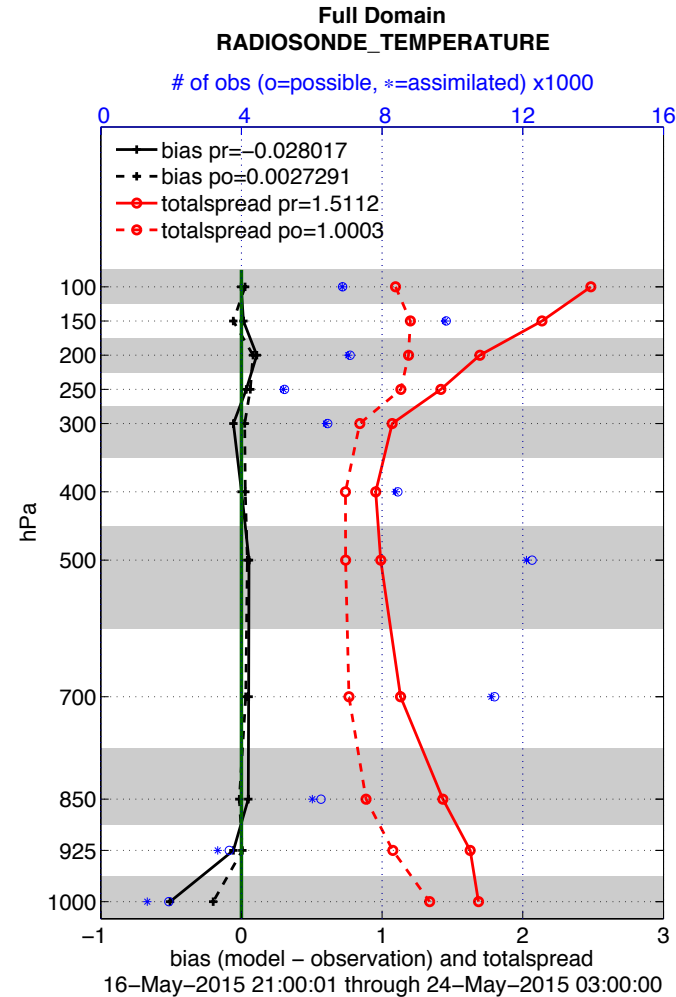
plot\_profile.m

plot\_bias\_xxx\_profile.m plot\_rmse\_xxx\_profile.m

Note: These are much more informative for models with levels! (i.e. the 1D models are not very interesting this way)

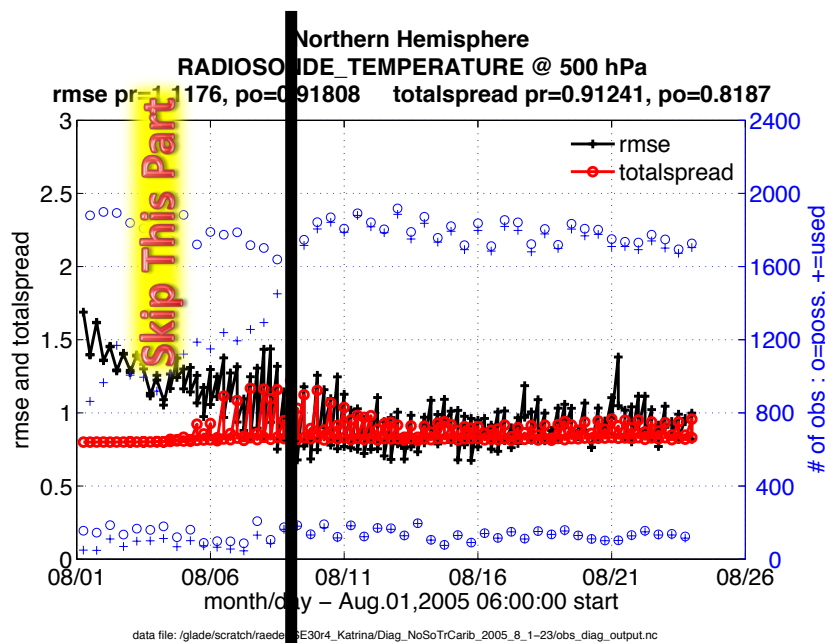


data file: /Users/thoar/svn/DART/clean\_lanai/models/wrf/work/obs\_diag\_output.nc



data file: /Users/thoar/svn/DART/clean\_lanai/models/wrf/work/obs\_diag\_output.nc

# A word of warning ...



NOTE: The `&obs_diag_nml` is different for low-order models and realistic models. Check carefully when using.

```
&obs_diag_nml
  obs_sequence_name = ''
  obs_sequence_list = 'file_list.txt'
  first_bin_center = 2005, 8, 1, 6, 0, 0
  last_bin_center = 2005, 8, 26, 0, 0, 0
  bin_separation = 0, 0, 0, 6, 0, 0
  bin_width = 0, 0, 0, 6, 0, 0
  time_to_skip = 0, 0, 10, 0, 0, 0
  max_num_bins = 1000
  trusted_obs = 'null'
  ...
/
```

***obs\_diag*** `'time_to_skip'` setting will allow you to ignore the spinup before starting the time-averaging for the vertical profiles while still calculating metrics for the entire period of record for the time-evolution products.

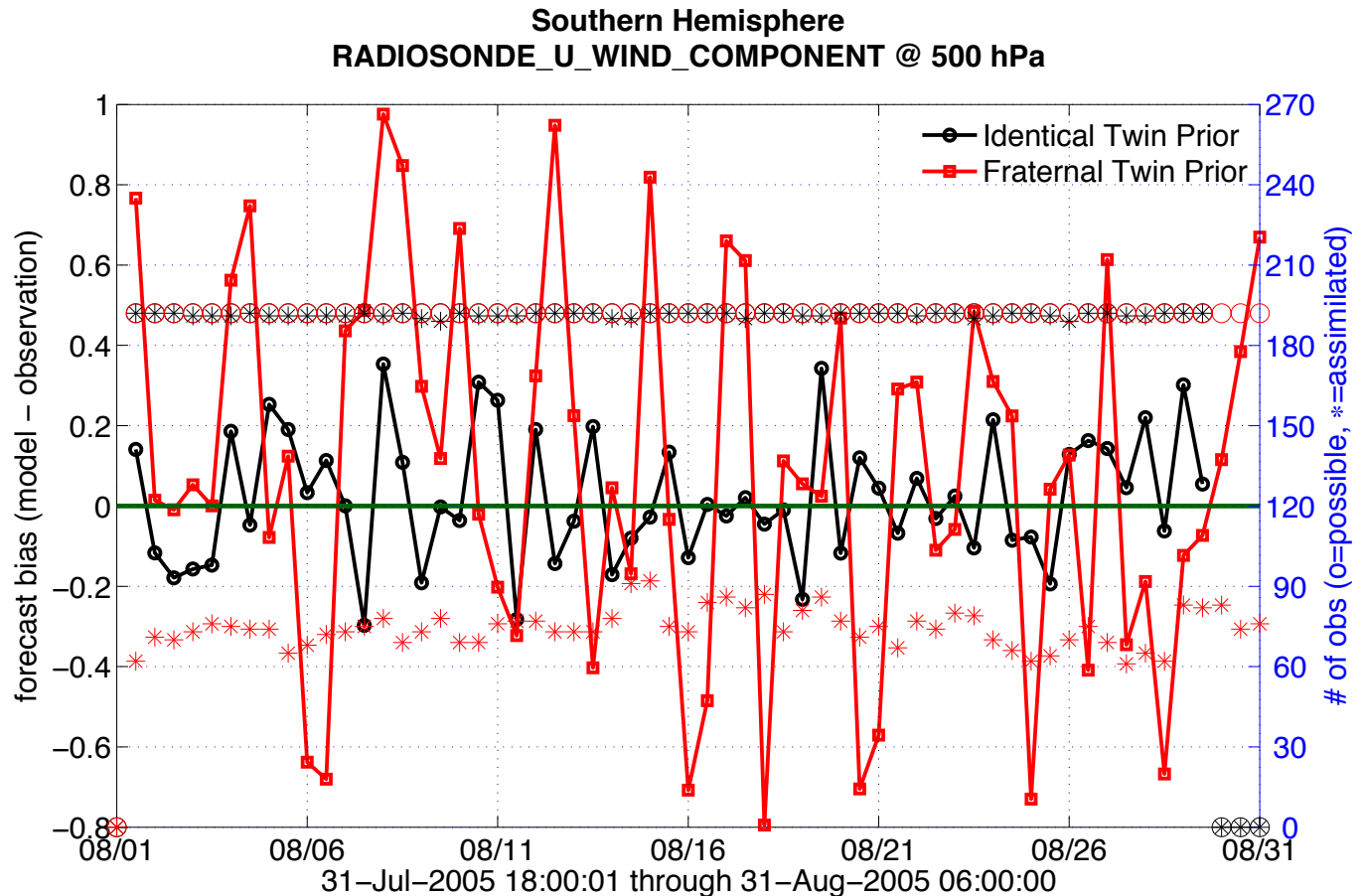
# Observation-space diagnostics: comparing experiments

`two_experiments_evolution.m`

`two_experiments_profile.m`

This is useful for quick comparisons. Really 'fair' comparisons require more processing to compare the same set of observations across experiments.

[assimilation\\_code/programs\\_obs\\_common\\_subset.html](#)  
[obs\\_seq\\_coverage.html](#)  
[obs\\_selection.html](#)  
[obs\\_seq\\_verify.html](#)



FYI:

'Identical' means the model that was used to generate the observations is also used for the assimilation.

'Fraternal' means the observations came from a different model.

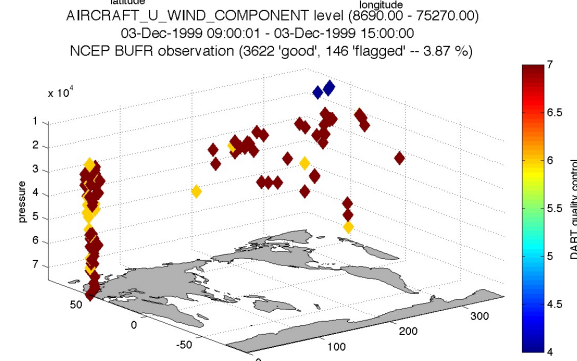
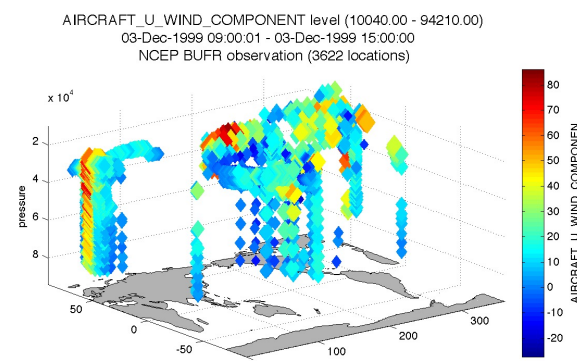
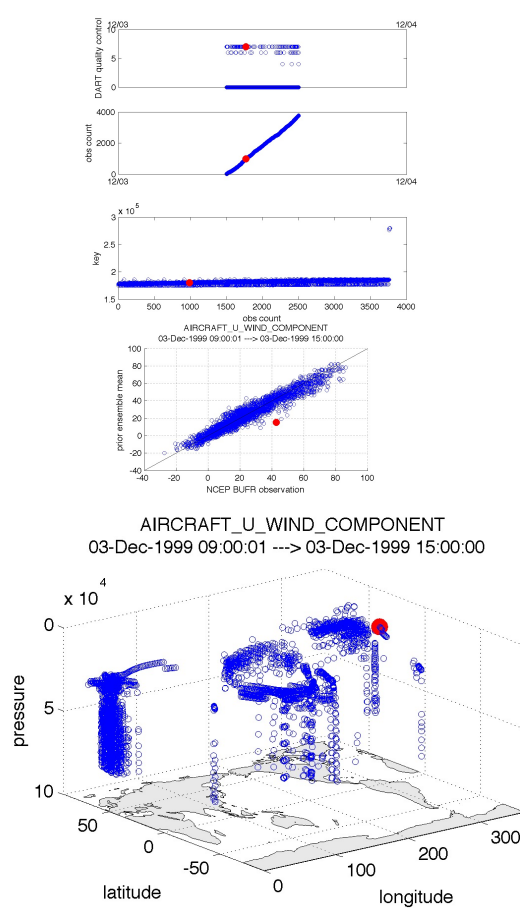
# Observation-space diagnostics: netCDF

SOME of the information in the observation files can be converted to netCDF and easily plotted. A program named **obs\_seq\_to\_netcdf** must be run to produce the netCDF.

Here are a few of the Matlab functions available in *diagnostics/matlab*.

- link\_obs.m
- plot\_obs\_netcdf.m
- plot\_obs\_netcdf\_diffs.m
- plot\_coverage.m

```
MATLAB 7.3.0 (R2007a)
File Edit View Graphics Debug Desktop Window Help
Current Folder: /Users/rhour/Documents/DART/models/can/work
Workspace:
Variable Editor - obsmat
obsmat: 3768x9 doubles
1 2 3 4 5 6 7 8 9 10 11
984 0.8500 5.2700 65210 43.1000 42.8470 0 1.7942 7.3046e+05 984
985 2.6100 47.6800 37930 22.1000 28.0713 0 1.7924 7.3046e+05 985
986 2.4000 47.9200 38790 22.4000 28.0624 0 1.7913 7.3046e+05 986
987 17.1000 42.2700 31720 12.6000 14.5302 0 1.8004 7.3046e+05 987
988 165.6000 34.3000 12060 20.6000 13.1179 7 1.8023 7.3046e+05 988
989 155.8000 33.4000 28220 44.4000 40.9271 0 1.8033 7.3046e+05 989
990 149.1000 35.3000 17670 20.6000 12.7447 0 1.8109 7.3046e+05 990
991 211.2000 55.8000 23950 15.9000 13.7103 0 1.8234 7.3046e+05 991
992 170.0000 44.6000 17600 15.6000 13.4665 0 1.8160 7.3046e+05 992
Command Window
>> link_obs('obs', 'prior ensemble', 'DART quality control - region')
name = POP11/obs epoch 011.nc :
obsTypeString = AIRCRAFT_U_WIND_COMPONENT :
obsCopyString = NCEP BUFR observation :
OCString = NCEP BUFR observation :
copyString = prior ensemble mean :
copyString = DART quality control :
region = (0 360 -90 90 -Inf Inf) :
>> plot_obs_netcdf('obs', 'prior ensemble', 'DART quality control - region')
line_obsName = obsTypeString_obsCopyString_OCString_region :
N = 100 MAXLINE_SPEC_TEMPERATURE (type 5) been levels 0.00 and 4500.00
N = 13528 LAND_SFC_ALTITUDE (type 18) been levels 12.00 and 4781.00
N = 1248 RADIOSONDE_U_WIND_COMPONENT (type 12) been levels 0.00 and 181500.00
N = 10488 RADIOSONDE_U_WIND_COMPONENT (type 13) been levels 0.00 and 181500.00
N = 10478 RADIOSONDE_TEMPERATURE (type 15) been levels 300.00 and 181500.00
N = 8748 RADIOSONDE_SPECIFIC_HUMIDITY (type 16) been levels 38000.00 and 181500.00
N = 3768 AIRCRAFT_U_WIND_COMPONENT (type 20) been levels 8000.00 and 94210.00
N = 3768 AIRCRAFT_V_WIND_COMPONENT (type 28) been levels 8000.00 and 94210.00
N = 3742 AIRCRAFT_TEMPERATURE (type 23) been levels 5300.00 and 181510.00
N = 4478 ACARS_U_WIND_COMPONENT (type 25) been levels 5300.00 and 181510.00
N = 4464 ACARS_TEMPERATURE (type 27) been levels 5300.00 and 181510.00
N = 1392 MAXLINE_SFC_ALTITUDE (type 18) been levels 0.00 and 4781.00
N = 1392 MAXLINE_SFC_U_WIND_COMPONENT (type 12) been levels 0.00 and 193.00
N = 1312 MAXLINE_SFC_TEMPERATURE (type 15) been levels 0.00 and 193.00
N = 13872 SAT_U_WIND_COMPONENT (type 43) been levels 13700.00 and 92500.00
N = 13872 SAT_V_WIND_COMPONENT (type 44) been levels 13700.00 and 92500.00
DART quality control: 11 OC copy 2
DART quality control: 11 OC copy 3
DART quality control: 11 OC copy 4
DART quality control: 11 OC copy 5
OC summary follow:
DART quality control == 4) 3622 obs (assimilated)
DART quality control == 4) 4 obs (prior forward operator failed)
DART quality control == 4) 11 obs (prior QC rejected)
DART quality control == 7) 113 obs (outlier rejected)
```



111 obs with qc == 7 'outlier rejected'  
31 obs with qc == 6 'prior QC rejected'  
4 obs with qc == 4 'prior forward operator failed'



# Complicated observation-space diagnostics.

The program ***obs\_seq\_to\_netcdf*** converts much of the information in an observation sequence file to a netCDF file. For now, we're going to explore a pre-computed file available at:

[www.image.ucar.edu/pub/DART/Tutorial\\_Datasets/obs\\_epoch\\_SE30r4\\_Katrina.nc](http://www.image.ucar.edu/pub/DART/Tutorial_Datasets/obs_epoch_SE30r4_Katrina.nc)

It was generated with the following input:

```
&schedule_nml
  calendar          = 'Gregorian'
  first_bin_start   = 2005, 8, 13, 21, 0, 0
  first_bin_end     = 2005, 8, 14, 03, 0, 0
  last_bin_end      = 2005, 8, 14, 03, 0, 0
  bin_interval_days = 10000
  bin_interval_seconds = 0
  max_num_bins      = 1000
  print_table       = .true.
/
&obs_seq_to_netcdf_nml
  obs_sequence_name = 'cam_obs_seq.2005-08-14-00000.final'
  obs_sequence_list = ''
  lonlim1 = 160.
  lonlim2 = 40.
  latlim1 = 10.
  latlim2 = 65.
/
```



# DART Tutorial Index to Sections

1. Filtering For a One Variable System
2. The DART Directory Tree
3. DART Runtime Control and Documentation
4. How should observations of a state variable impact an unobserved state variable?  
Multivariate assimilation.
5. Comprehensive Filtering Theory: Non-Identity Observations and the Joint Phase Space
6. Other Updates for An Observed Variable
7. Some Additional Low-Order Models
8. Dealing with Sampling Error
9. More on Dealing with Error; Inflation
10. Regression and Nonlinear Effects
11. Creating DART Executables
12. Adaptive Inflation
13. Hierarchical Group Filters and Localization
14. Quality Control
15. DART Experiments: Control and Design
16. Diagnostic Output
17. Creating Observation Sequences
18. Lost in Phase Space: The Challenge of Not Knowing the Truth
- 19. DART-Compliant Models and Making Models Compliant**
- 20. Model Parameter Estimation**
- 21. Observation Types and Observing System Design**
- 22. Parallel Algorithm Implementation**
23. Location module design (not available)
24. Fixed lag smoother (not available)
- 25. A simple 1D advection model: Tracer Data Assimilation**