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

©UCAR

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.
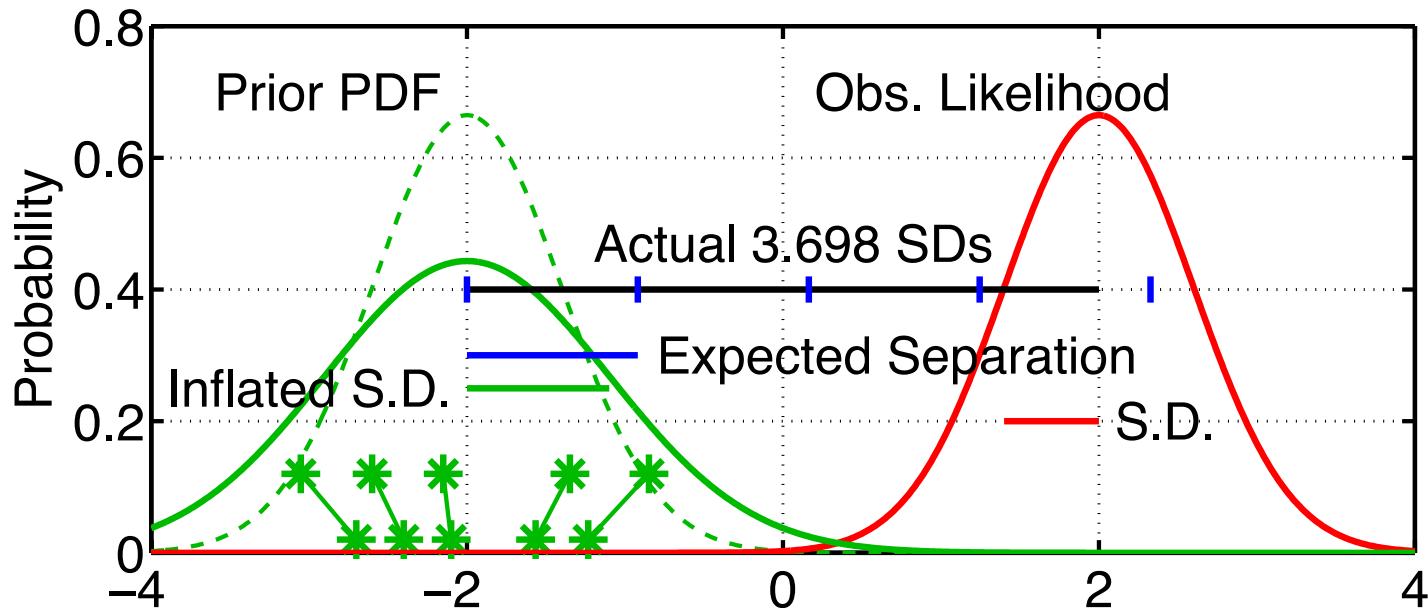
What to expect …

Recall that
Expected(prior_mean − observation) = $\sqrt{\sigma^2_{prior} + \sigma^2_{obs}}$
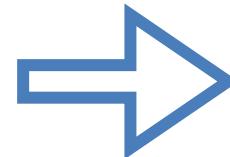
Error is dominated by observational noise if $\sigma^2_{obs} \gg \sigma^2_{prior}$

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

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

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 $\left|\overline{y^p} - y^o\right| \Big/ \sqrt{\sigma_{prior}^2 + \sigma_{obs}^2}$ > 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

```
&obs_diag_nml
   obs_sequence_name     = 'obs_seq.final',
   bin_width_days        = -1,
   bin_width_seconds     = -1,
   init_skip_days        = 0,         (Slightly different for 3D models.)
   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.
   /
```

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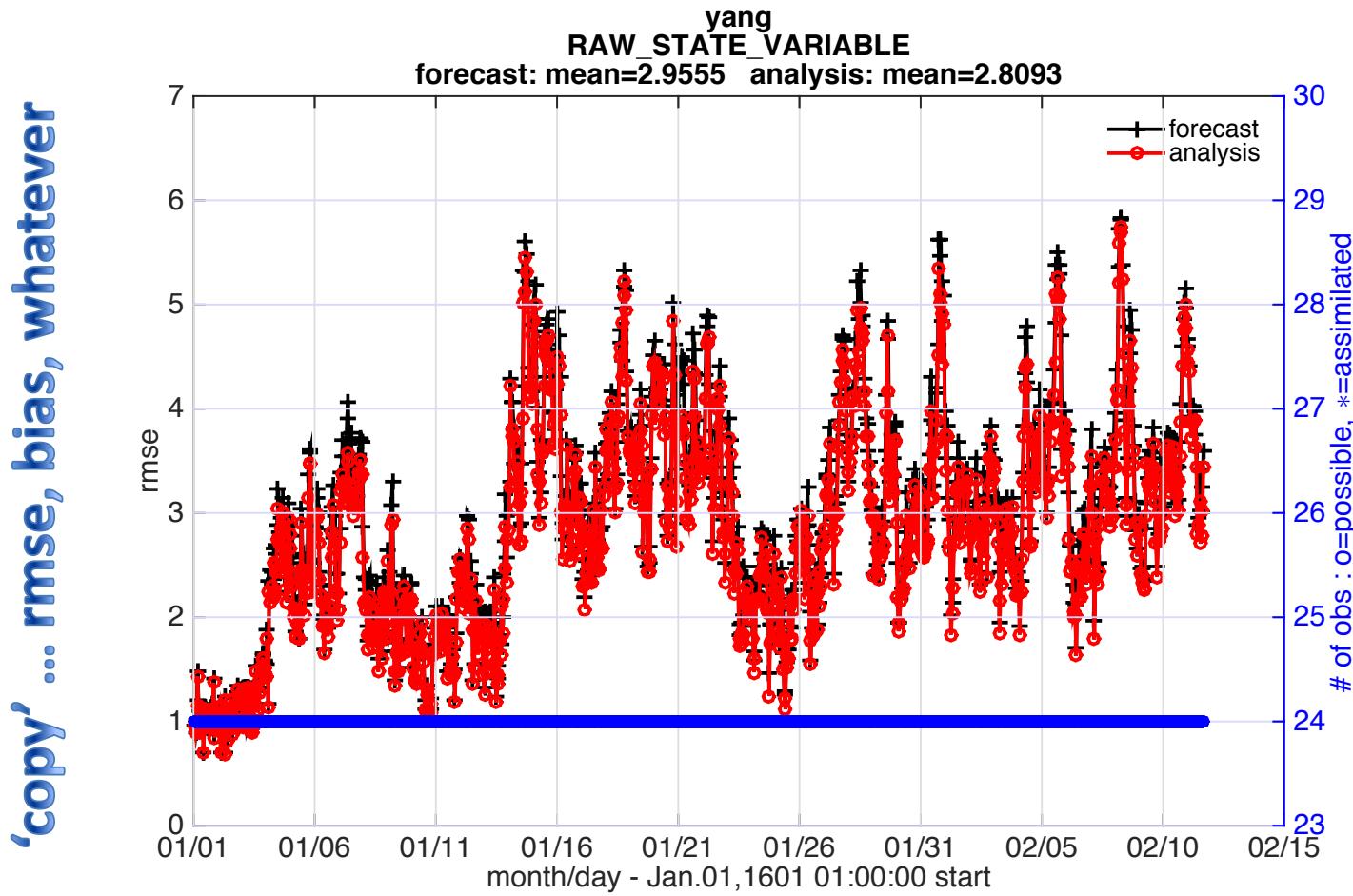
Here are a few of the Matlab functions available in *diagnostics/matlab*

- **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

'copy' ... rmse, bias, whatever

Observation use metrics

Time (aka 'evolution')

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 what-
   …                                                 ever you like.
 /


&quality_control_nml
   input_qc_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*!
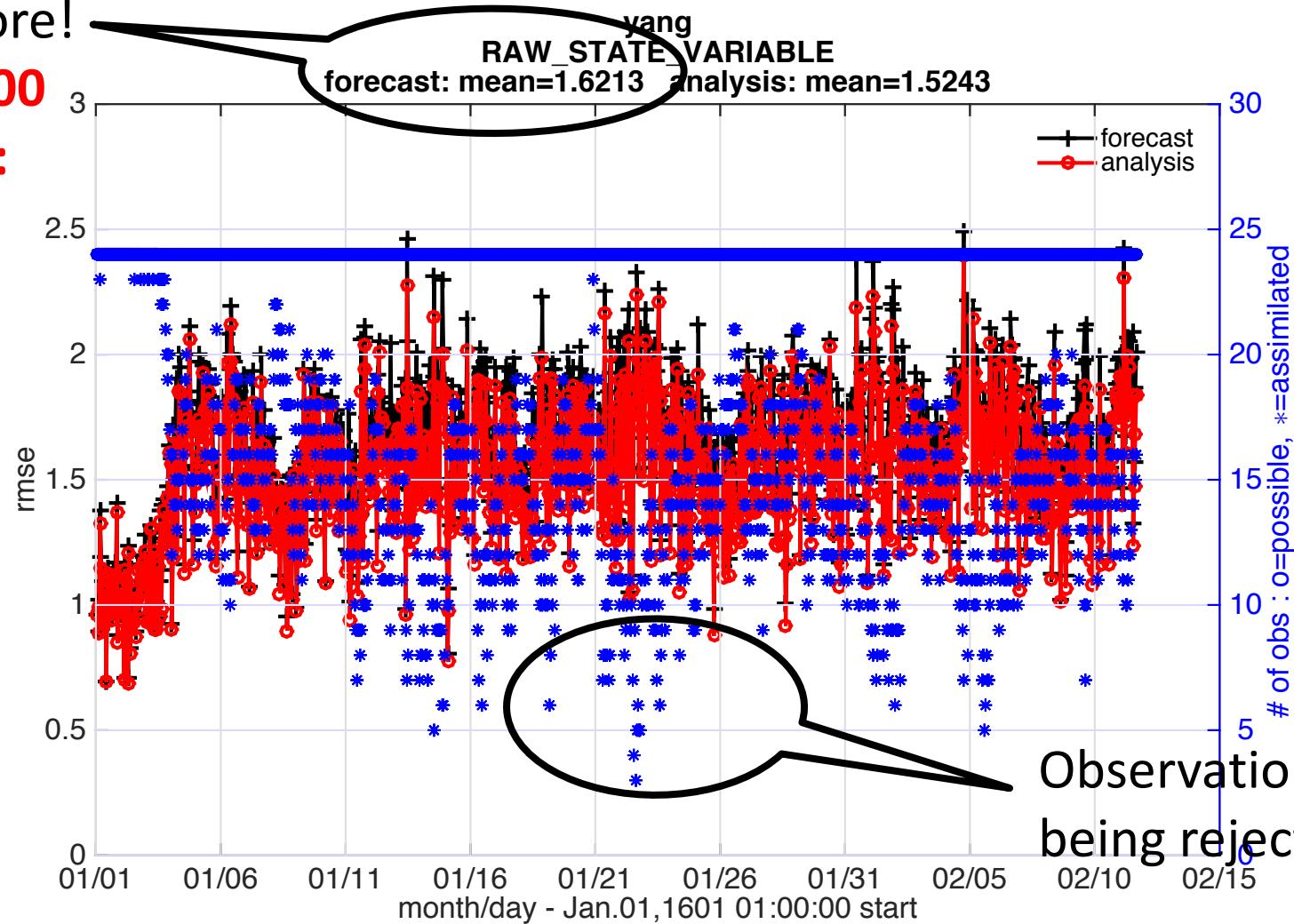
This is potentially **DANGEROUS**, but useful.
Rejecting 'good' observations can lead to inflated estimate of quality.

First Observation-space diagnostics:

Lower RMSE than before!

$1,000,000 question: Why?

outlier_threshold = 3.0

Observations being rejected!

data file: /Users/thoar/svn/DART/clean_lanai/models/lorenz_96/work/obs_diag_output.nc
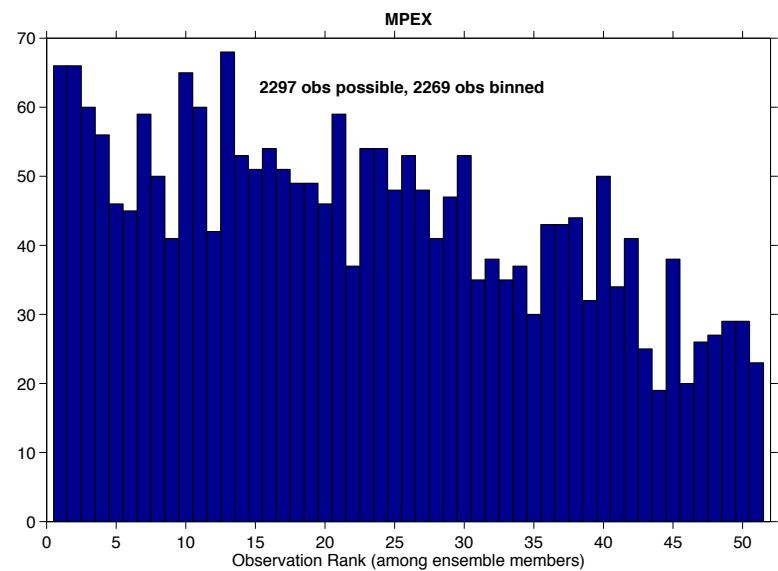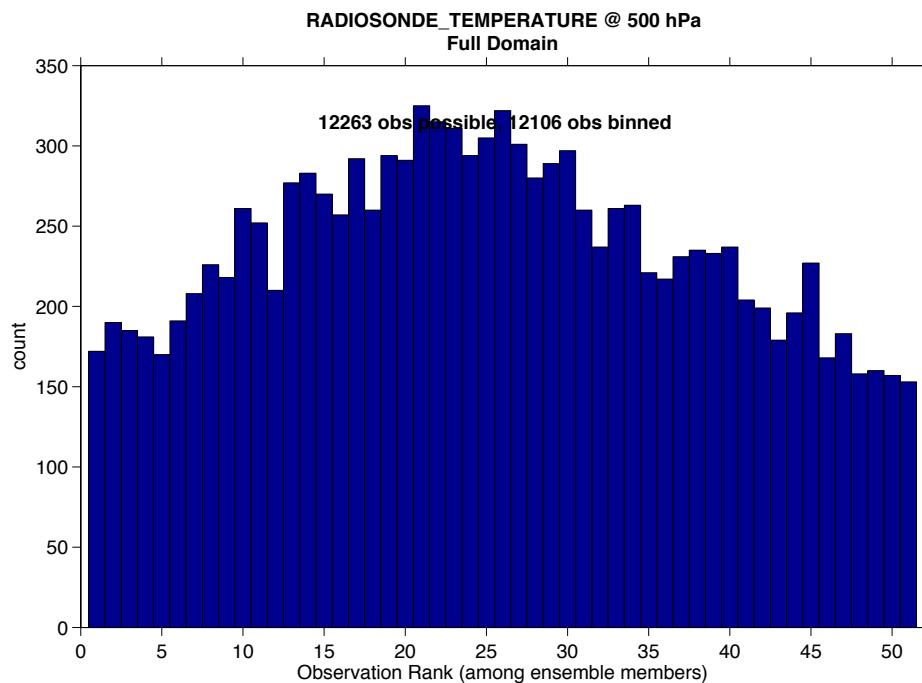
Lorenz_96 exercises:

- 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);
```


RADIOSONDE_TEMPERATURE @ 500 hPa
Full Domain

12263 obs possible, 12106 obs binned

MPEX

2297 obs possible, 2269 obs binned
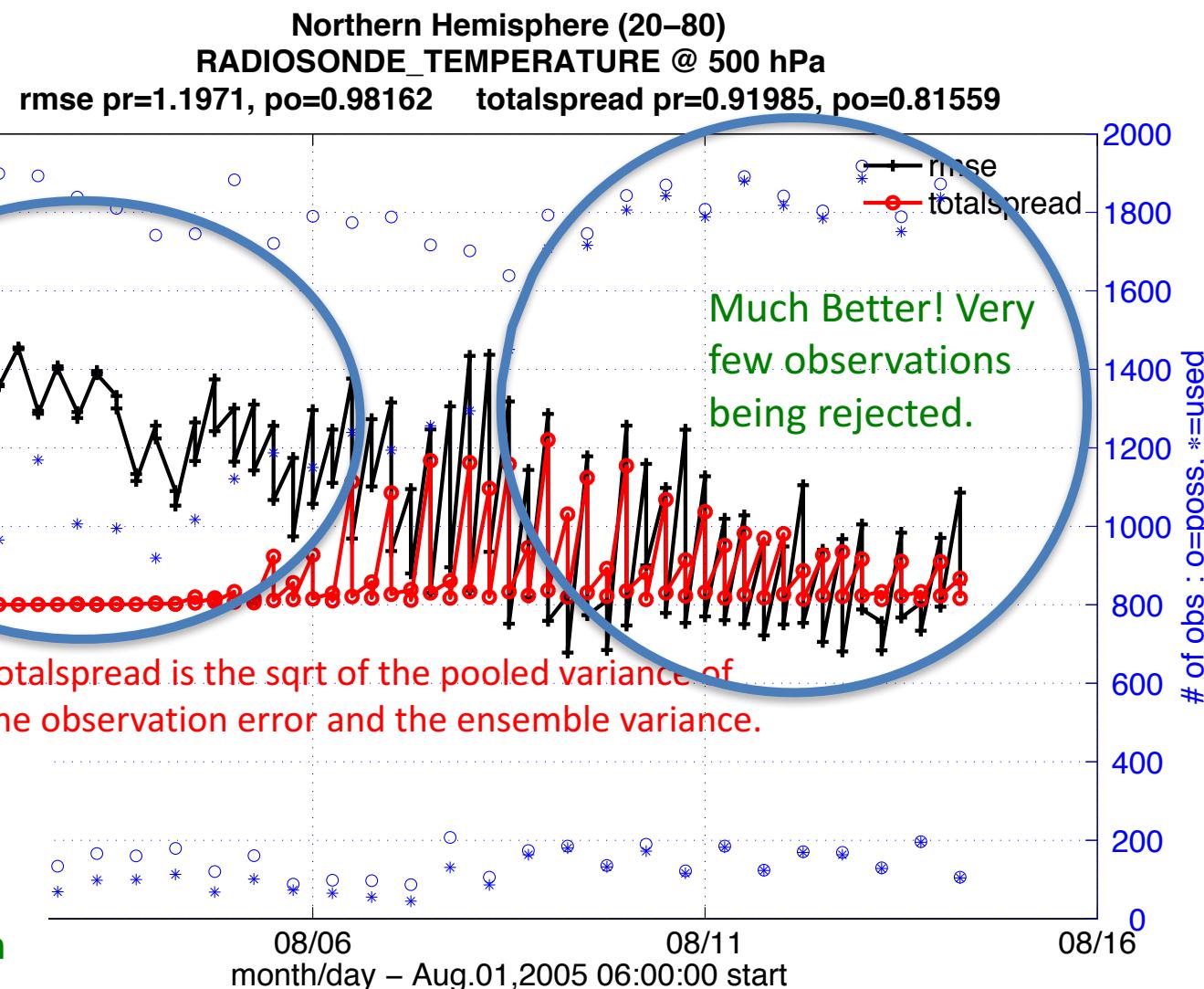
Results from WRF real-time forecasting.

May.16,2015 21:00:01 –– May.24,2015 03:00:00
data file: /Users/thoar/svn/DART/clean_lanai/models/wrf/work/obs_diag_output.nc

plot_rmse_xxx_evolution.m

plot_evolution.m
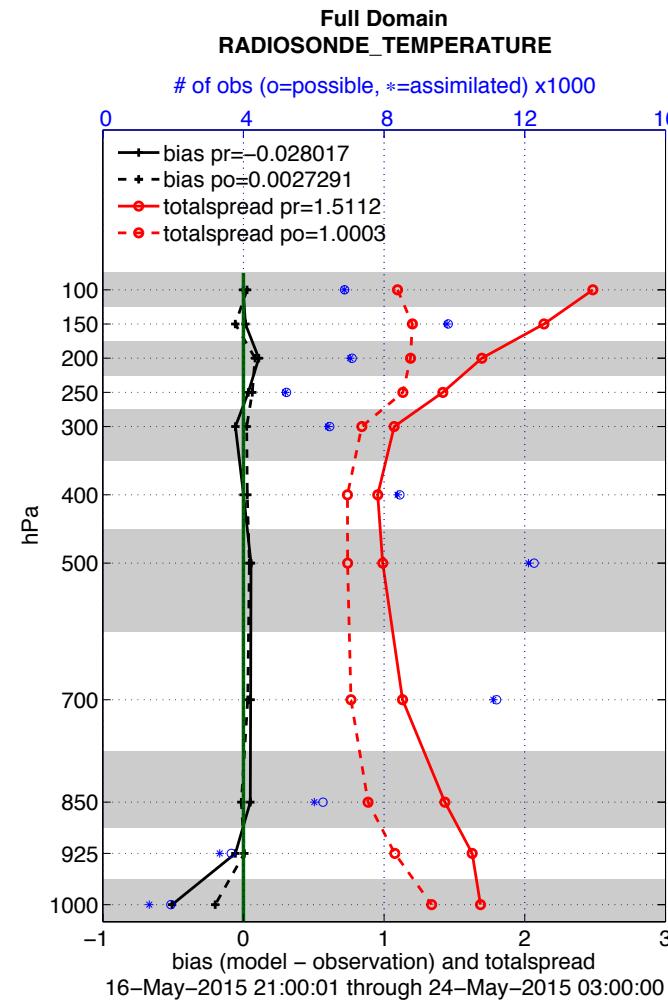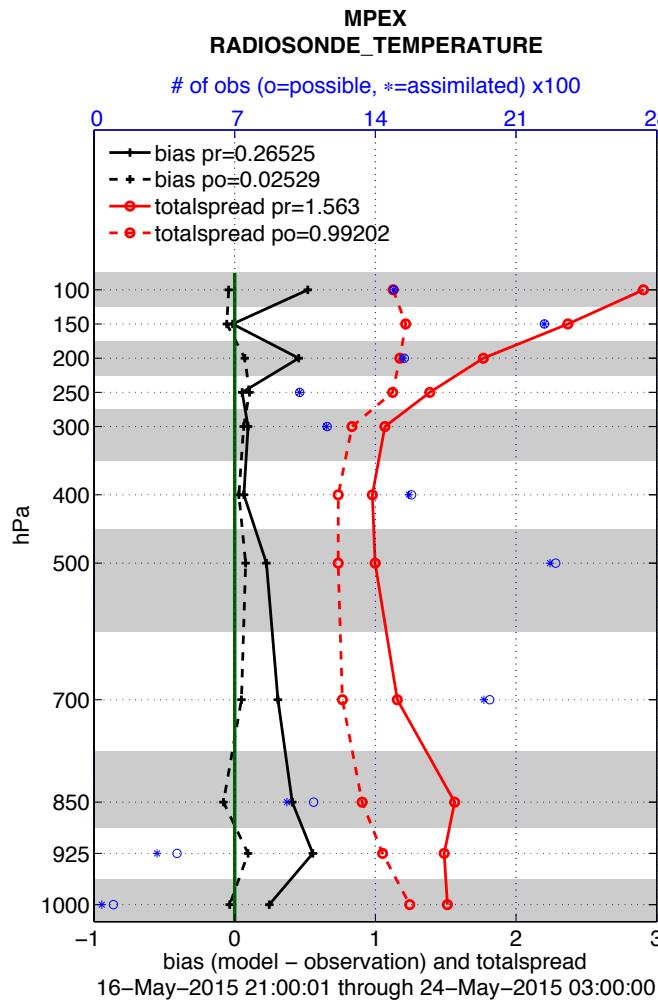
Northern Hemisphere (20–80)
RADIOSONDE_TEMPERATURE @ 500 hPa
rmse pr=1.1971, po=0.98162   totalspread pr=0.91985, po=0.81559

Much Better! Very few observations being rejected.

Totalspread is the sqrt of the pooled variance of the observation error and the ensemble variance.

Initially tiny spread and large observation rejection – system not performing well – yet!
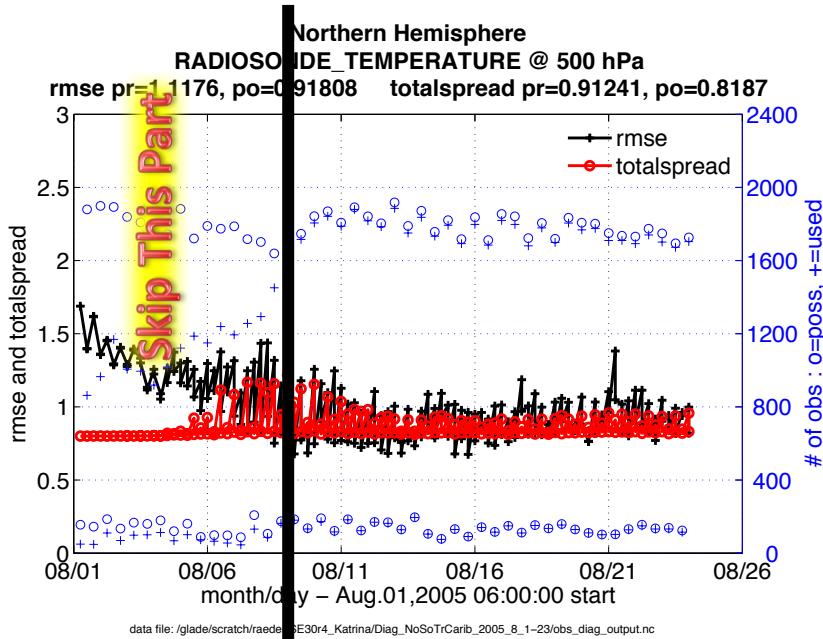
plot_profile.m   **plot_bias_xxx_profile.mplot_rmse_xxx_profile.m**

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

MPEX
RADIOSONDE_TEMPERATURE

# of obs (o=possible, *=assimilated) x100

bias pr=0.26525
bias po=0.02529
totalspread pr=1.563
totalspread po=0.99202

bias (model − observation) and totalspread
16−May−2015 21:00:01 through 24−May−2015 03:00:00

data file: /Users/thoar/svn/DART/clean_lanai/models/wrf/work/obs_diag_output.nc

Full Domain
RADIOSONDE_TEMPERATURE

# of obs (o=possible, *=assimilated) x1000

bias pr=−0.028017
bias po=0.0027291
totalspread pr=1.5112
totalspread po=1.0003

bias (model − observation) and totalspread
16−May−2015 21:00:01 through 24−May−2015 03:00:00

data file: /Users/thoar/svn/DART/clean_lanai/models/wrf/work/obs_diag_output.nc

A word of warning ...

Northern Hemisphere
RADIOSONDE_TEMPERATURE @ 500 hPa
rmse pr=1.1176, po=0.91808   totalspread pr=0.91241, po=0.8187

data file: /glade/scratch/raeder/SE30r4_Katrina/Diag_NoSoTrCarib_2005_8_1–23/obs_diag_output.nc

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.
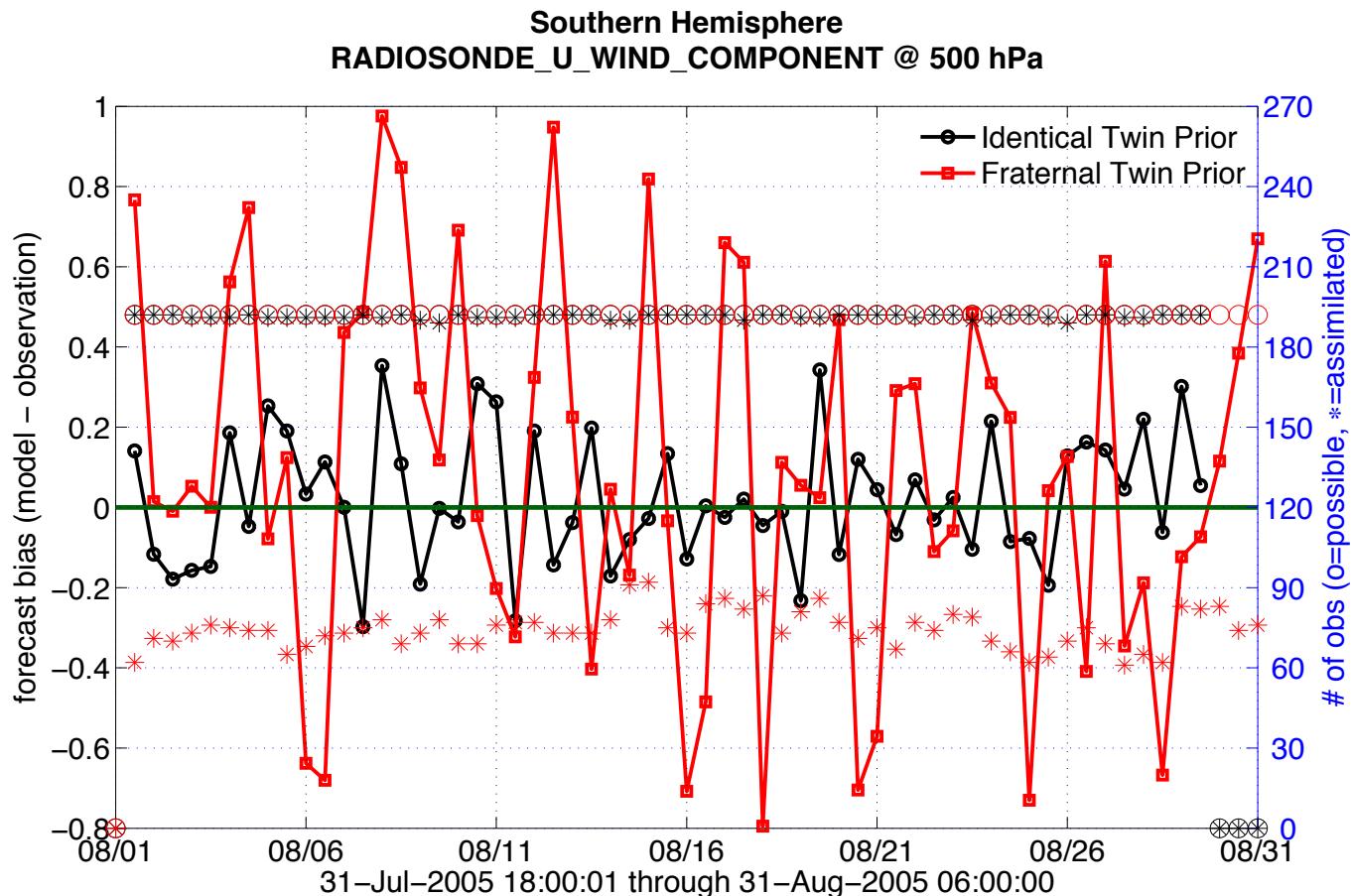
two_experiments_evolution.m

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

Southern Hemisphere
RADIOSONDE_U_WIND_COMPONENT @ 500 hPa
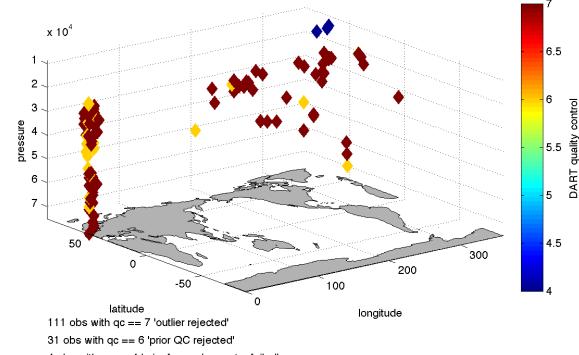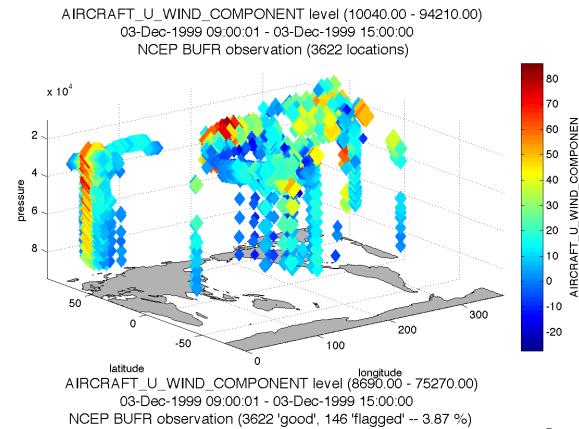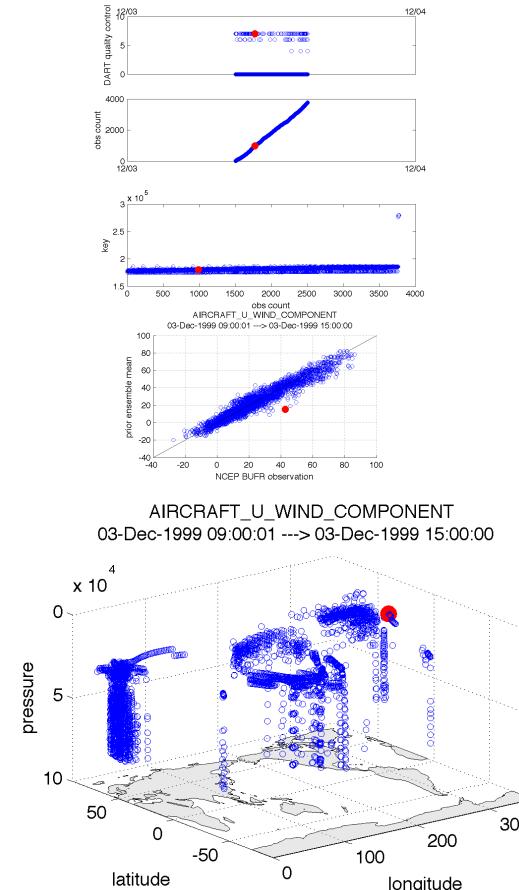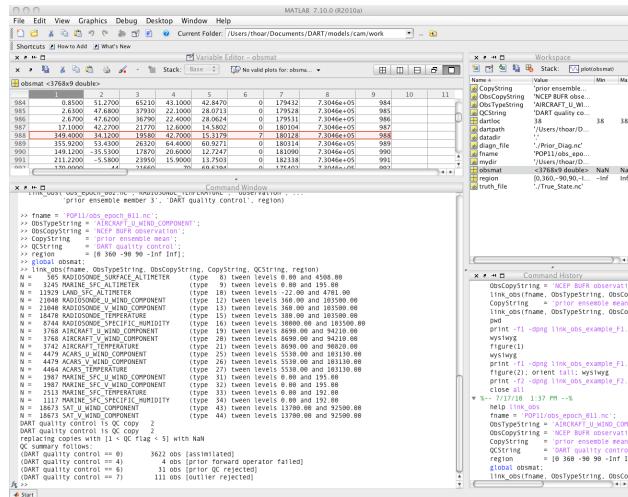
FYI:
'Identical' means the model that was used to generate the observations is also used for the assimilation.
'Fraternal' means the observations came from a different model.

SOME of the information in the observation sequence 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**

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

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.
  /
```
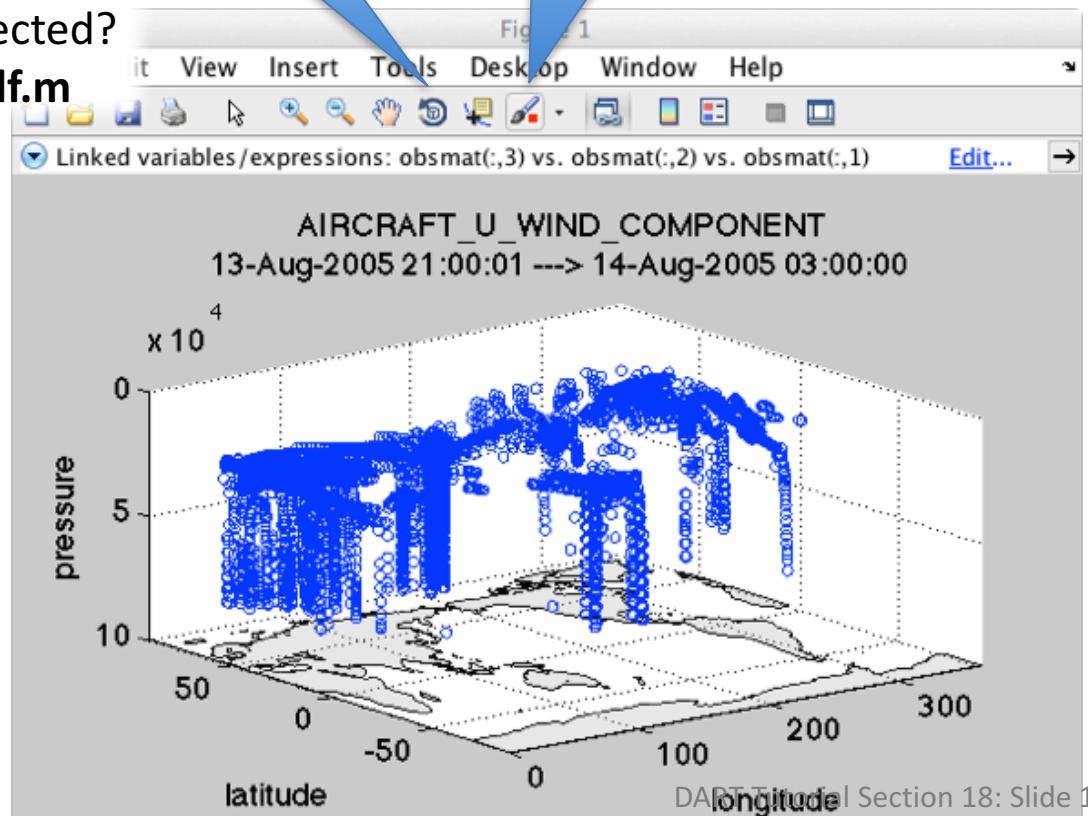
Matlab Hands-On: **link_obs** exploring observations

This enables rotation with the mouse.

'paintbrush' allows you to select observations for 'brushing'

Try different obs types,
Try to locate rejected obs,
Why were they rejected?
Try **plot_obs_netcdf.m**

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**