Model Evaluation Tools (MET) tutorial

July, 2009

Thank You
to our sponsors

Support for MET is provided by

the Developmental Testbed Center (DTC)

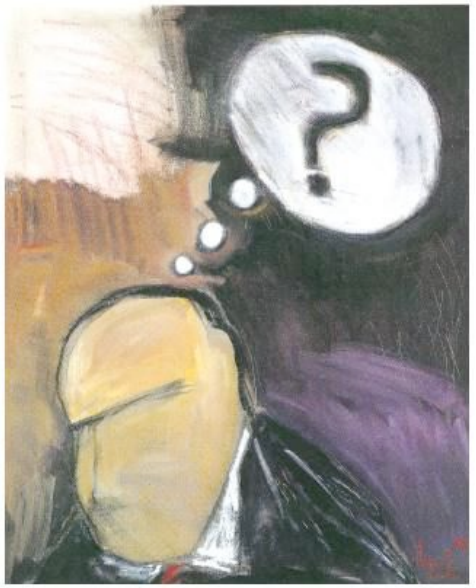
and

the Air Force Weather Agency (AFWA).

Who?

Presenting :
(In order of appearance)

- Tressa L. Fowler
- **John Halley Gotway**
- Tara Jensen
- Dave Ahijevych
- **Randy Bullock**
- Eric Gilleland



Involved but not present:

- Barbara G. Brown
- **Steve Sullivan**
- Lacey Holland (now at 3Tier)

- What?
  - Set of verification tools for evaluating forecasts via
    - standard statistics
    - object-based methods
    - scale decompositions

- Why?
  - Make verifiying easy.
  - Encourage verification.
  - Promote consistency across users.

- How?
  - A (unix like) package of software tools and scripts.
  - Community contributed methods, graphics, etc.

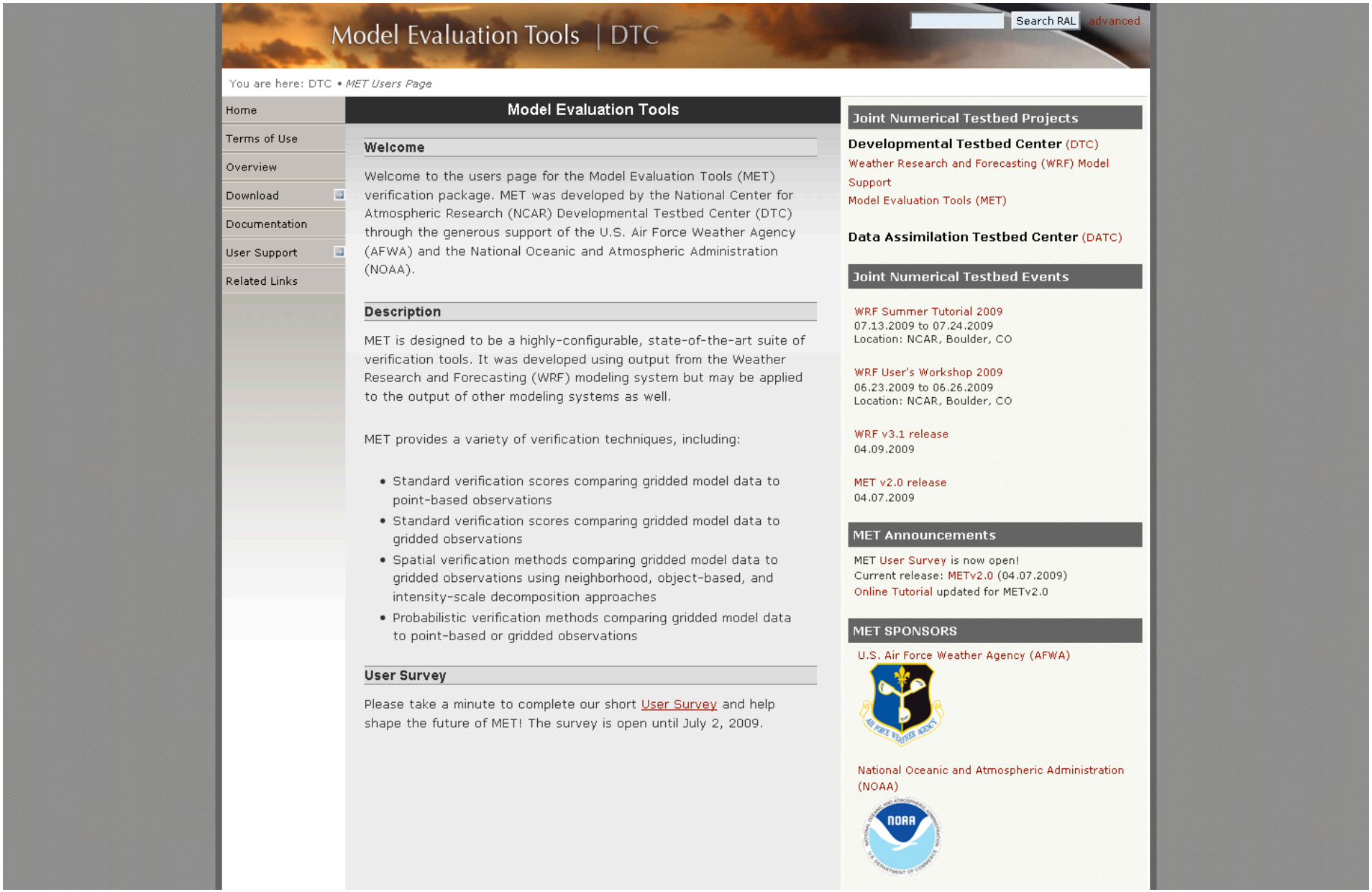
Release History

- **METv0.9**: Beta release – July, 2007
- **METv1.0**: First official release – January, 2008
- **METv1.1**: Incremental upgrades – July, 2008
- **METv2.0**: Current release – April, 2009
  - Pre-installed on tutorial machines
  - 500+ registered users from 66 countries
  - 50/50 University/Non-University users
  - On-line tutorial updated for METv2.0
  - Hands-on tutorial offered with the WRF-Tutorial

Downloading MET

- Download MET release and compile locally.
  - Register and download: **www.dtcenter.org/met/users**
- Language:
  - Primarily in C++ with calls to a Fortran library
- Supported Platforms and Compilers:
  1. Linux with GNU compilers
  2. Linux with Portland Group (PGI) compilers
  3. Linux with Intel compilers
  4. IBM machines with IBM compilers

www.dtcenter.org/met/users



Dependencies

- REQUIRED:
  - GNU Make Utility
  - C++/Fortran Compilers (GNU, PGI, Intel, or IBM)
  - NetCDF version 3 Library
  - BUFRLIB Library
  - GNU Scientific Library (GSL)
  - F2C Library (f2c or g2c, for some compilers)
- RECOMMENDED:
  - WRF Post-Processor
  - COPYGB (included with WRF-Post)
  - R statistics and graphics package

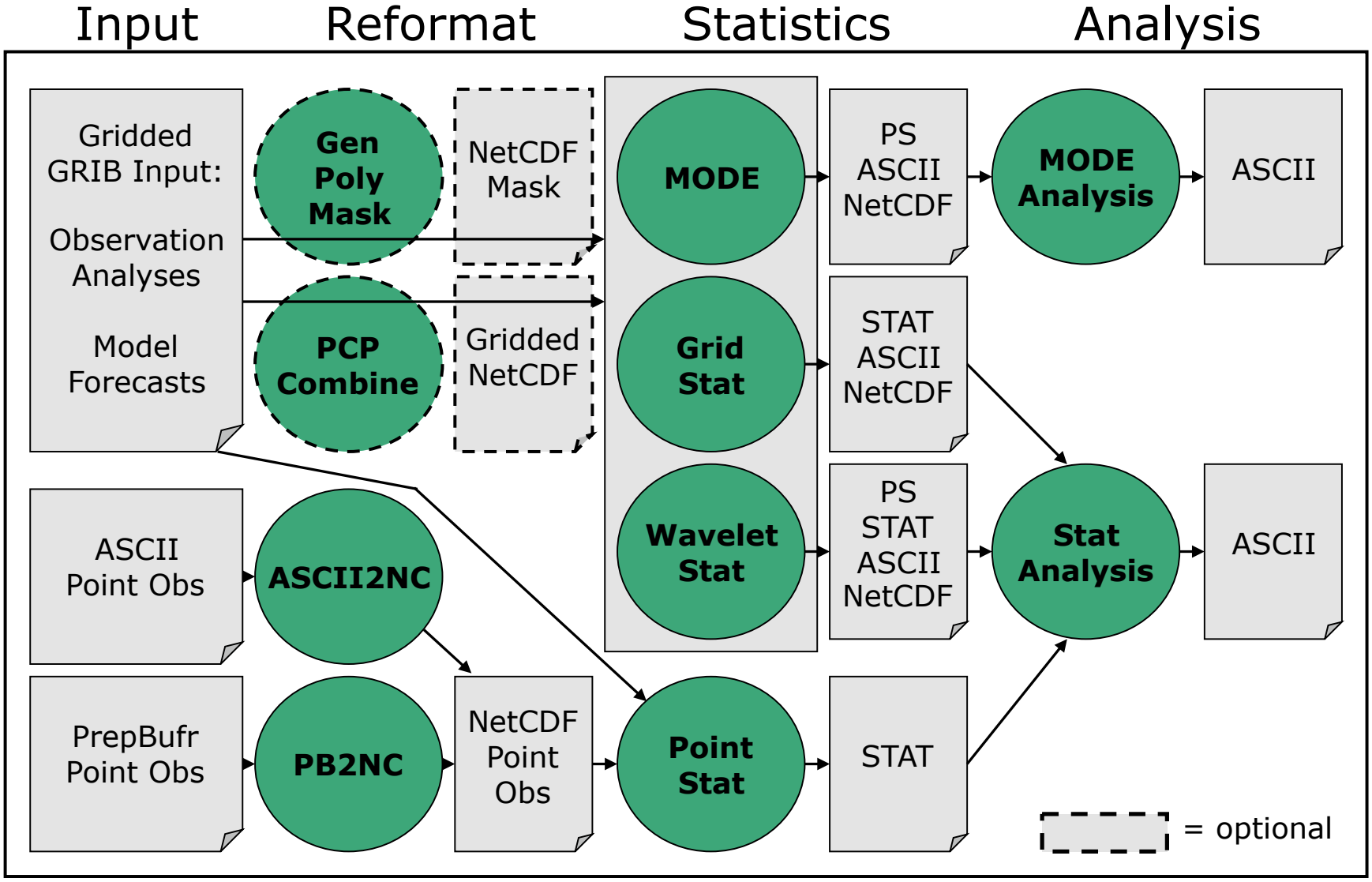
Directory Structure

File or Directory	Contents
README	Installation instructions and release notes.
Makefile_gnu (pgi, intel, ibm)	Top-level Makefile to be configured prior to building MET.
lib/	Source code for internal MET libraries.
src/	Source code for the MET applications.
doc/	MET User's Guide.
bin/	Built MET executables.
scripts/	Test scripts to be run after building MET.
data/	Sample data used by the test scripts.
out/	Output generated by the test scripts.

Building MET

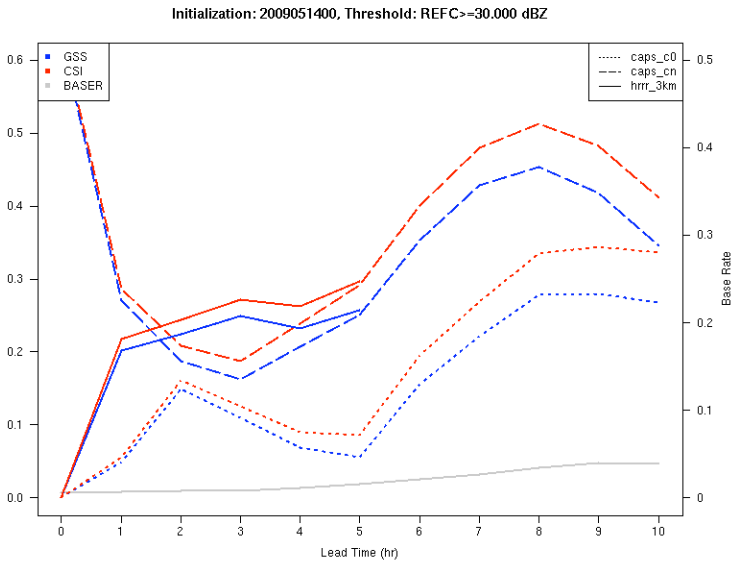
- Steps for building MET:
  1. Build required libraries.
     - Same family of compilers for MET
  2. Select the appropriate Makefile.
     - GNU, PGI, Intel, or IBM
  3. Configure the Makefile.
     - C++ and Fortran compilers
     - Paths for NetCDF, BUFRLIB, GSL, and F2C libraries
  4. Run Make to build all of the MET tools.
  5. Run the test script and check for runtime errors.
     - Runs each of the MET tools at least once.
     - Uses sample data distributed with the tarball.

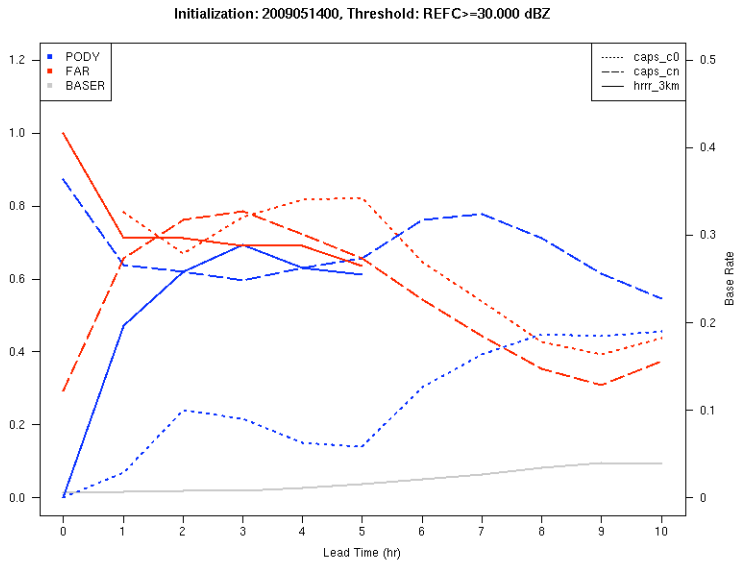
MET v2.0 Flowchart



Graphics

- Limited graphics incorporated into MET
- Options for plotting MET statistical output
  - R, NCL, IDL, GNUPlot, and many others
- Submit your own plotting and/or analysis scripts for posting to the MET website.





R Statistics and Graphics

- The R Project for Statistical Computing (www.r-project.org)
  - Powerful statistical analysis and plotting tools
  - Large and growing user community
  - Freely available and well supported for Linux/ Windows/Mac
- Sample R plotting and analysis scripts posted on the MET website

Configuration Files

- MET tools controlled using command line options and ASCII configuration files
  - Well commented and documented in MET User's Guide
  - Easy to modify
  - Distributed with the tarball
  - Web based config file creation for some tools.
- Configuration files control things such as:
  - Fields/levels to be verified
  - Thresholds to be applied
  - Interpolation methods to be used
  - Verification methods to be applied
  - Regions over which to accumulate statistics

Data Formats and Types

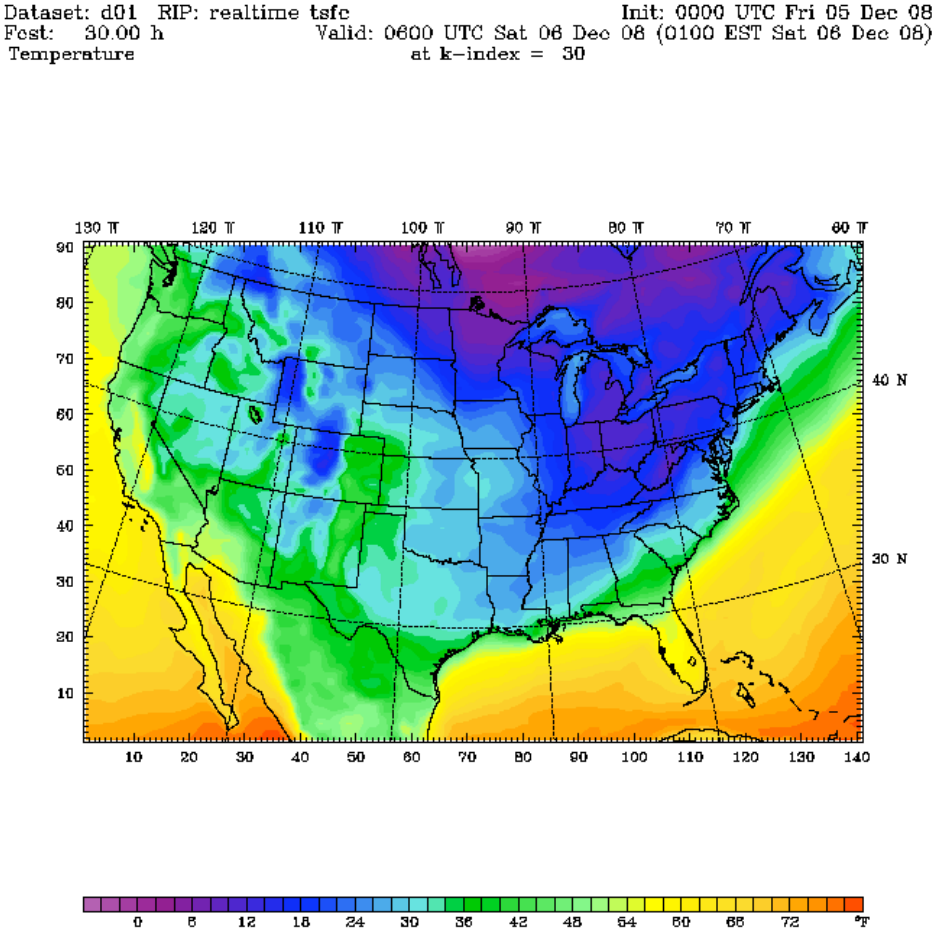
What you can do with MET depends on what type of data you have.

The **format** (grid, point) of your data determines your MET tool(s).

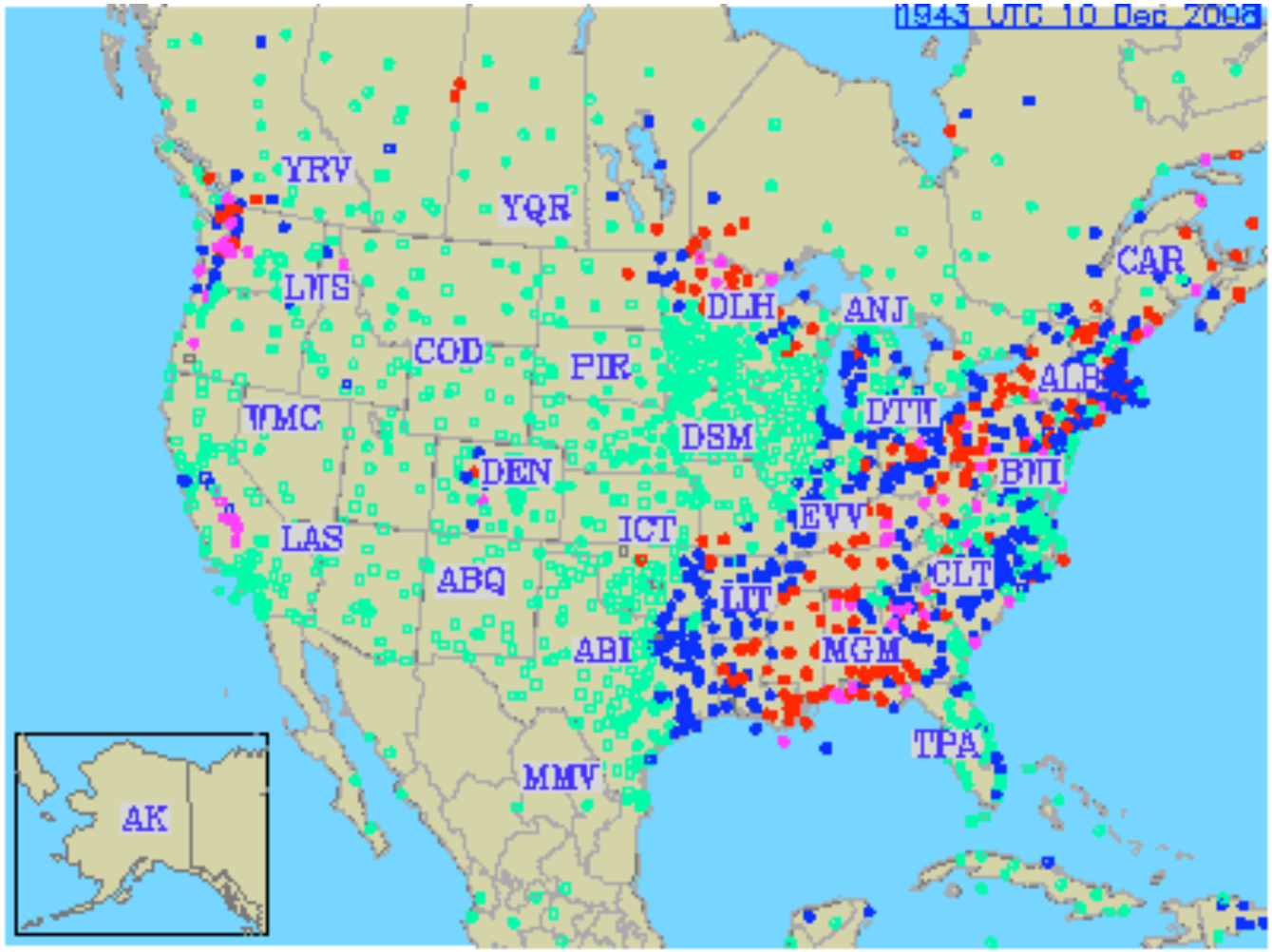
The **type** (continuous, binary) of your data determines the analyses to use within each tool.

Data Formats

Gridded Forecasts
(2D or 3D)



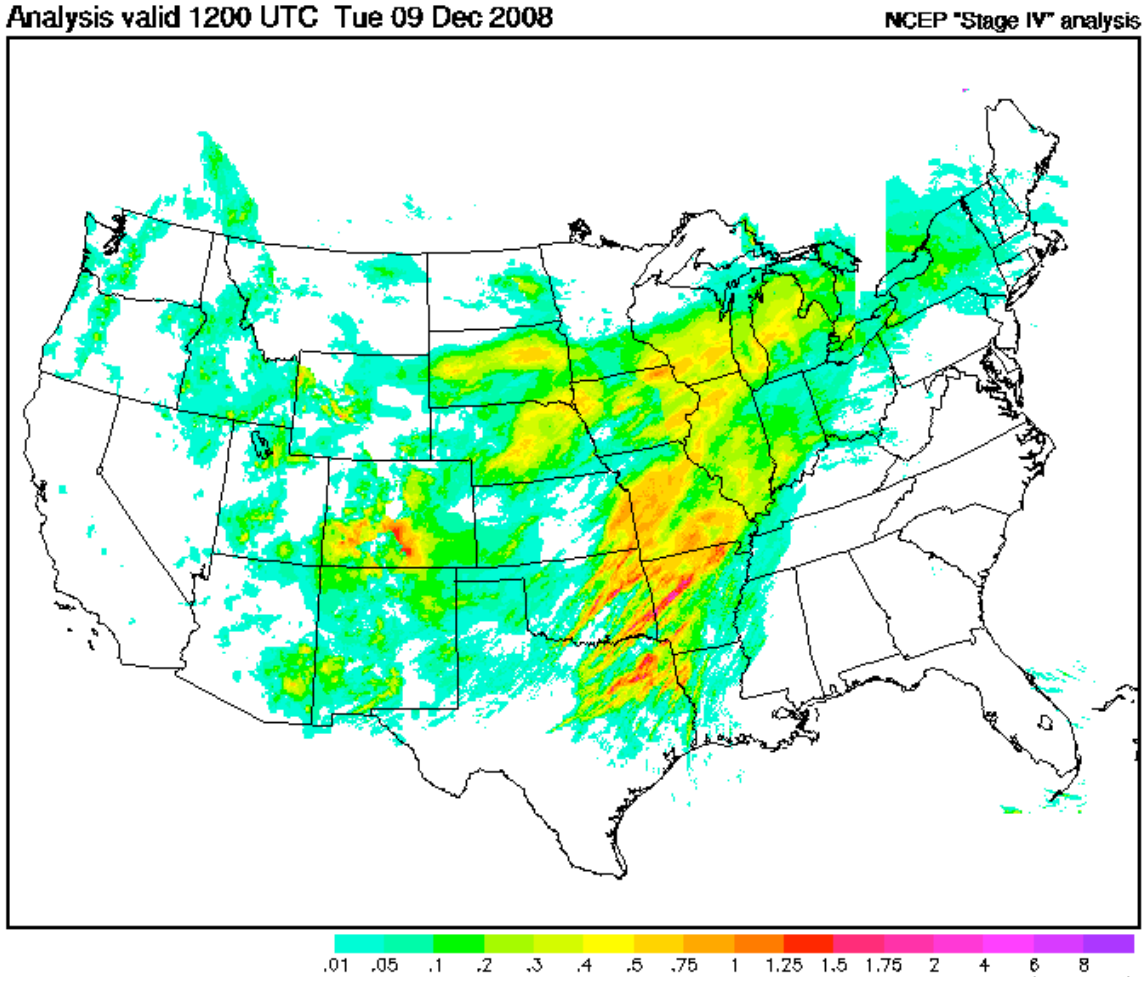
Point Observations
(2D or 3D)



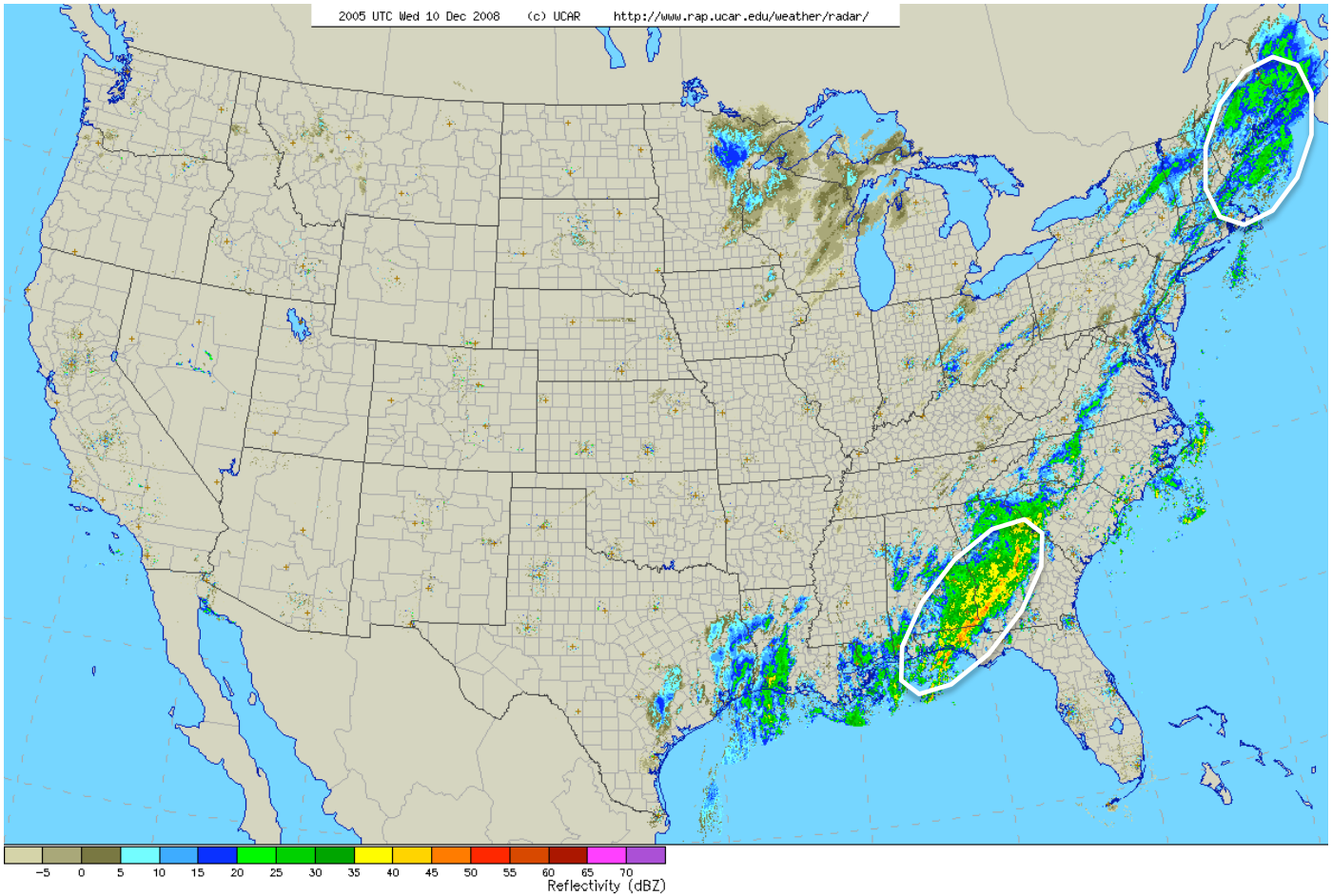
Gridded Observations

(2D or 3D)

Past 24-hour accumulated precip. (water equiv inches)



Gridded data
to transform into **Objects**



Data	MET Tool
Gridded Forecasts Gridded Observations	Grid stat Wavelet Stat MODE
Gridded Forecasts Point Observations	Point Stat

Data Types

Types of Forecasts

- Continuous
  - Wind speed
  - Temperature
- Categorical (includes Binary)
  - Rain / No Rain
  - Hurricane Category 1 - 5
- Probabilistic
  - Prob of freezing precip
- Ensembles

Types of Observations

- Continuous
  - Wind speed
  - Temperature
- Categorical (includes binary)
  - Rain / No Rain
  - No tornado, EF1, EF2, EF3, EF4, EF5

Data type	Analyses
Continuous forecasts Continuous observations	Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Bias
Continuous forecasts, Categorical observations	Receiver Operating Characteristic (ROC) curve, Kolmogorov-Smirnov
Categorical forecasts, Categorical observations	Contingency table statistics and skill scores
Probabilistic forecasts, Categorical observations	Brier score, ranked probability score (RPS), reliability diagram

Introduction to Standard Verification

Basics

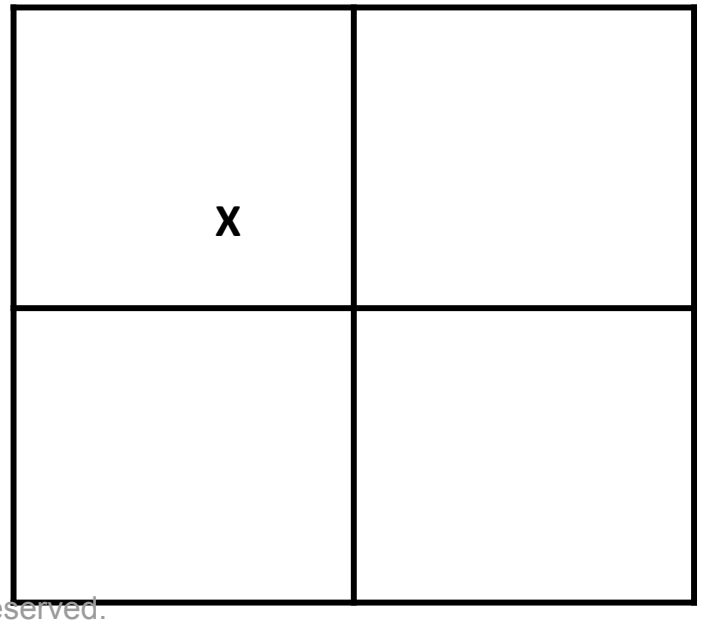
- Match up forecasts and observations at points.
- Calculate differences, sums, or counts over all the points.
- Summarize these things as statistics.

Matching Points to Grids

- Observation points are unlikely to fall exactly on forecast grid points.

- Match in horizontal space via choice of methods:
  - Closest
  - Interpolate
  - Function of surrounding points, e.g.
    - Min of closest 4
    - Median of closest 25

- Match in vertical by interpolating between level above and below.

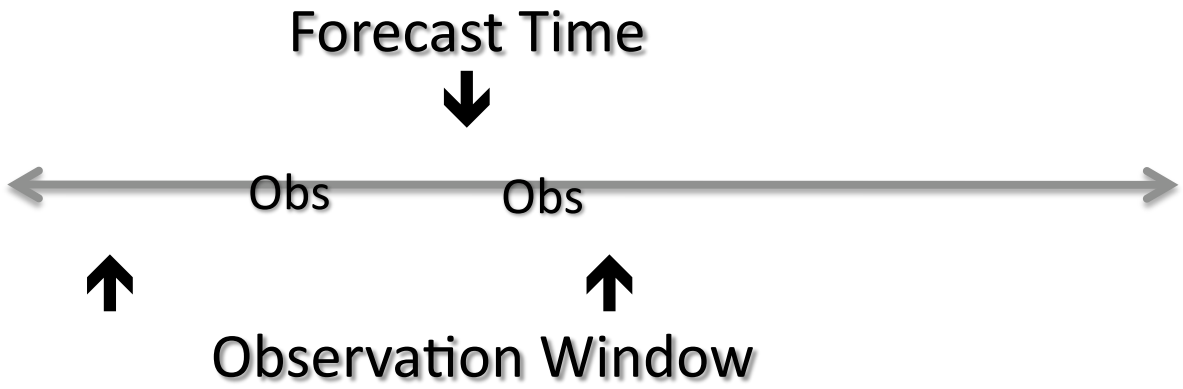


Matching Grids to Grids

- Must use some converter to put forecasts and observations on the same grid.
  - Example: copygb

Time

- If your forecasts and observations are not at the same time, you may need to define a time window for your observations.



Now you have a bunch of forecast / observation pairs

How well do they match?

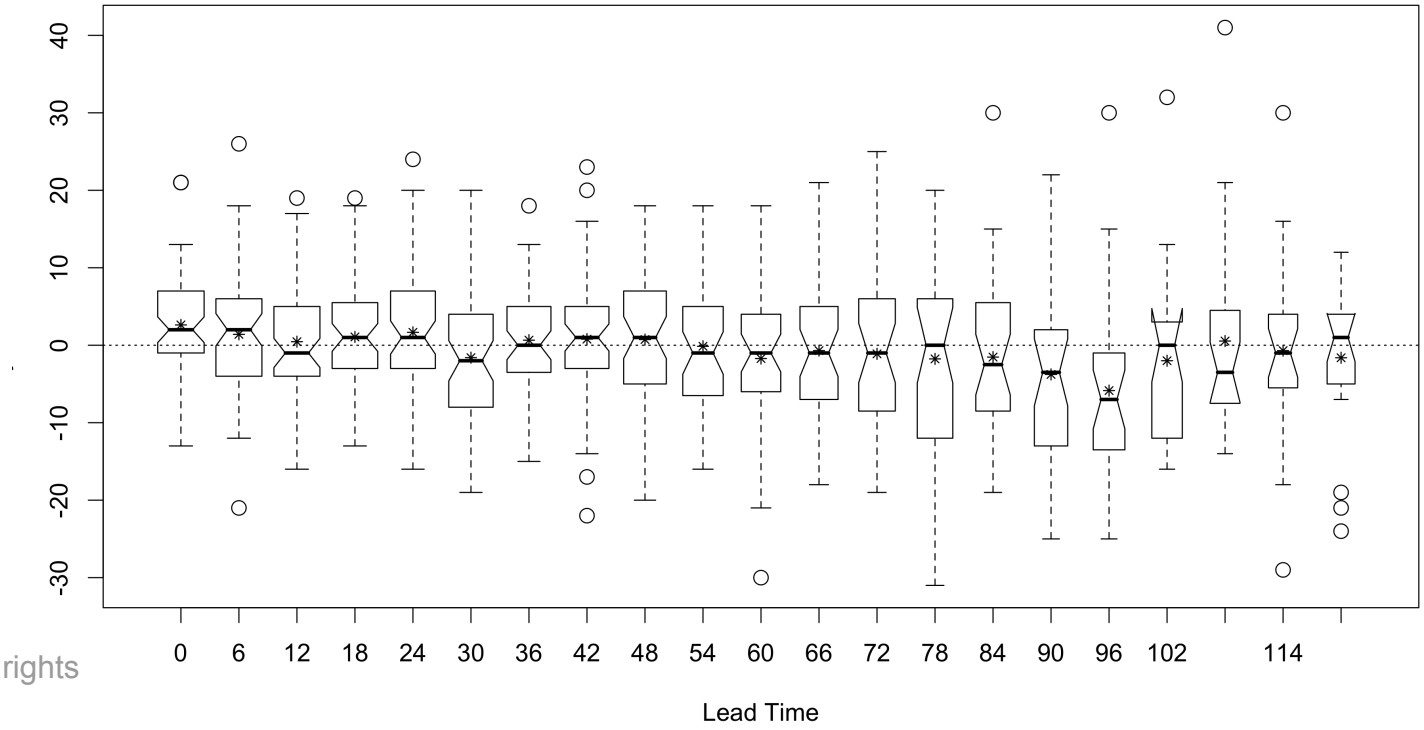
F	O	F	O
0.32	0.03	1	1
0. 51	0.0	0	0
0.42	0.48	0	1
0.08	0.14	1	0
0.20	0.23	0	1
0.75	0.33	1	1

How well do they match?Continuous Example

$$RMSE = \sqrt{\frac{\sum\left(F_i - O_i\right)^2}{n}}$$

$$r = \frac{\sum (F - \bar{F})(O - \bar{O})}{\sqrt{\sum (F - \bar{F})^2}\sqrt{\sum (O - \bar{O})^2}}$$

$$MAE = \frac{\sum\left|F_i - O_i\right|}{n}$$

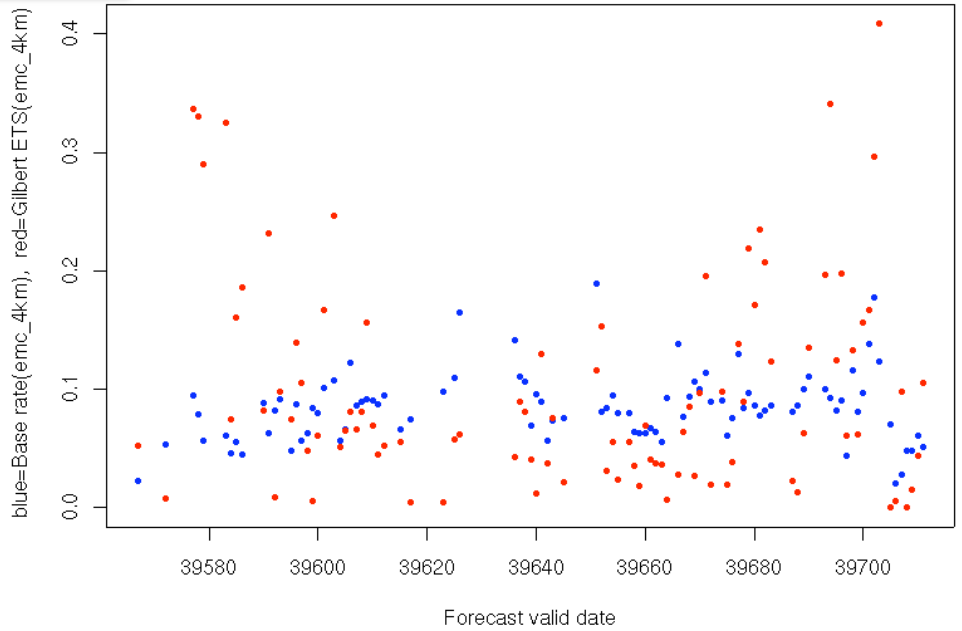


How well do they match? Categorical Example

	Observed Event	Observed Non-event
Forecast Event	Count = 532 (Hits)	Count = 219 (False Alarms)
Forecast Non-event	Count = 393 (Misses)	Count =1,627 (Correct No's)

$$GSS = \frac{hits - hits_{random}}{hits + misses + false\ alarms + hits_{random}}$$

$$POD = \frac{hits}{hits + misses}$$

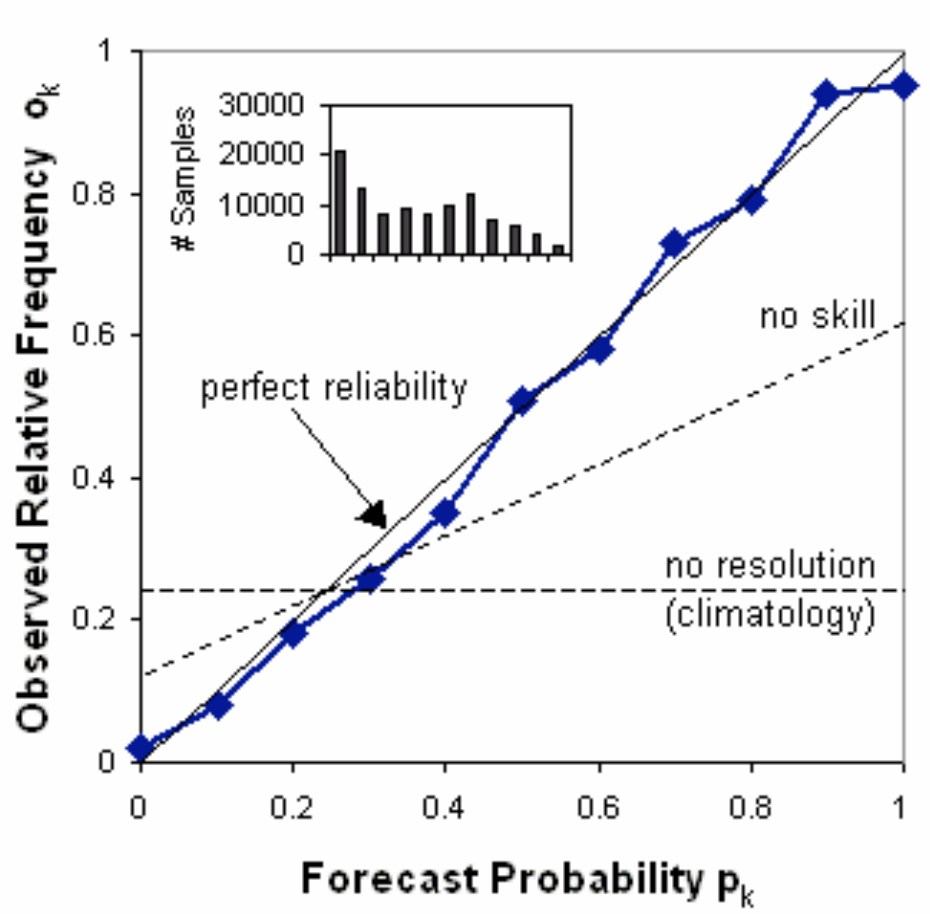


How well do they match? Probabilistic Example

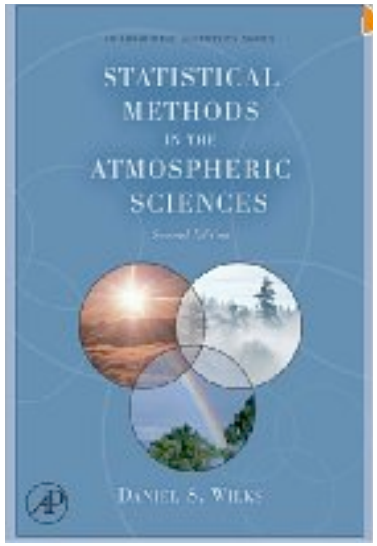
$$BS = \frac{1}{T}\sum (p_i - o_i)^2$$

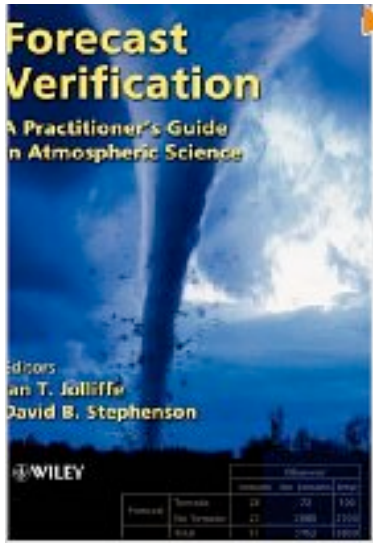
$$reliability_i = \frac{1}{T}\sum n_i\left(p_i - \overline{o}_i\right)^2$$

$$\text{Resolution} = \frac{1}{T}\sum n_{i.}(\overline{o}_i - \overline{o}\ )^2$$



References for information about verification





- http://www.bom.gov.au/bmrc/wefor/staff/eee/verif/verif_web_page.html
- Wilks, D.S., 2006. Statistical Methods in the Atmospheric Sciences. 2nd Ed, Academic Press.
- Jolliffe, Stephenson (eds.), 2003. Forecast Verification: A Practitioner's Guide in Atmospheric Science. Wiley.

File Formats and Pre-Processing

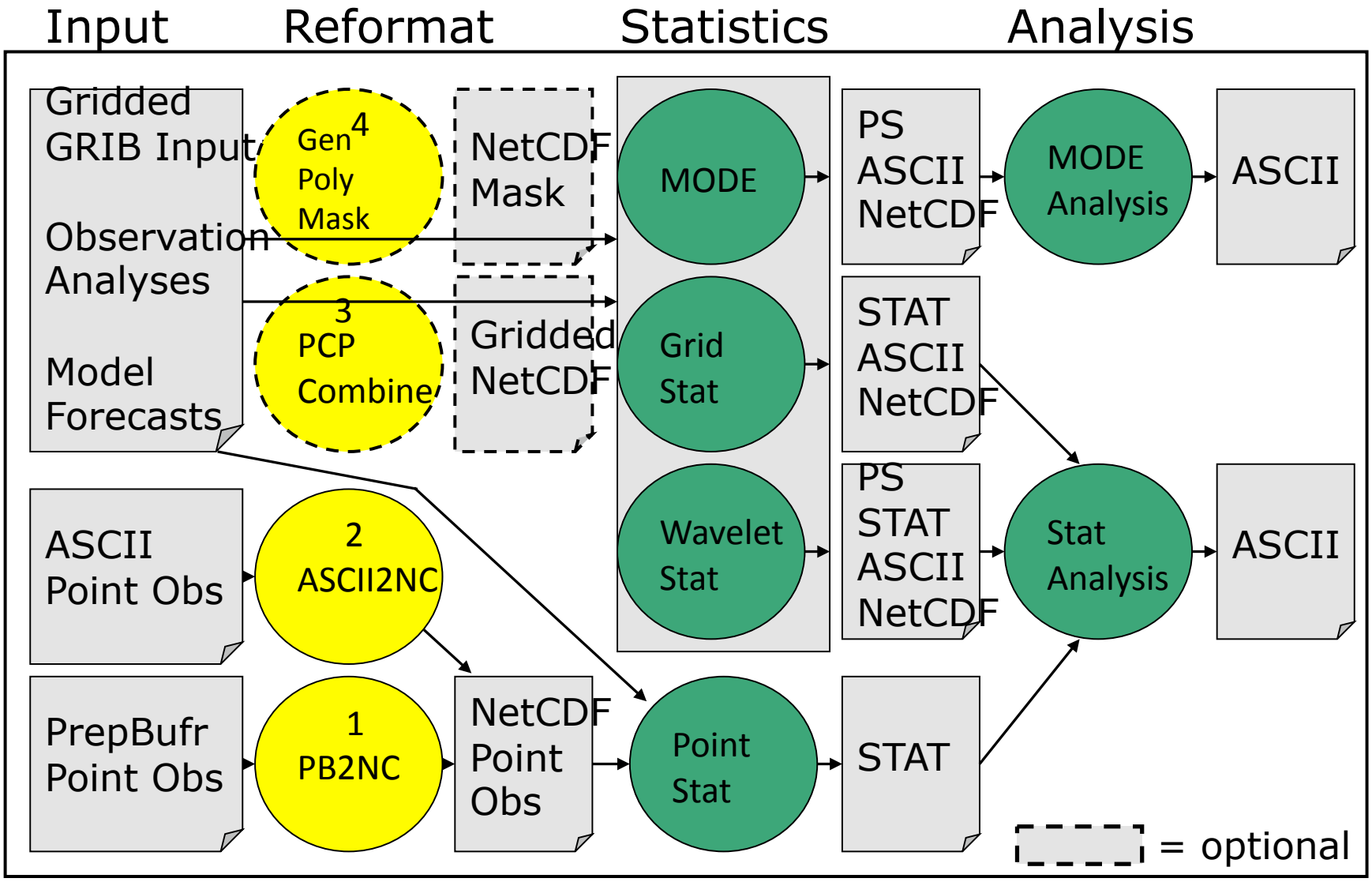
- File Formats
- Pre-processing Tools
- Useful Links

Presenter: Tara Jensen

Supported File Formats

- **Forecasts**
  - **GRIB** – GRIdded Binary file format (*version 1)
  - **NetCDF** – MET specific network Common Data Format
- **Observations**
  - **PREPBUFR** – binary dataset prepared by NCEP from varied data sources.
  - **ASCII** – MET specific format (10-cols x n-rows)
  - gridded **GRIB** – i.e. NEXRAD Level II or IV
  - **NetCDF** – MET specific NetCDF format

Pre-Processing / Reformatting



Data Reformating Tools

- **PB2NC and ASCII2NC**
  - Arrange observational data into the NetCDF point format expected by Point-Stat.
- **PCP_Combine (optional)**
  - Sum precipitation values across two or more time periods.
  - Subtract precipitation values to create values for finer subperiods.
  - Produces gridded NetCDF file that can be used as input grid for any Statistics tool.
- **Gen_Poly_Mask (optional)**
  - Used when more complex masking is needed.
  - Produces a NetCDF file of pre-defined mask.
  - May be used for masking in any Statistics tools.

1. PB2NC Tool

- **Stands for "PREPBUFR to NetCDF"**
- **Functionality:**
  - Filters and reformats PREPBUFR point observations into intermediate NetCDF format.
  - Configuration file specifies:
    - Observation types, variables, locations, elevations, quality marks, and times to retain or derive for use in Point-Stat.
- **Data formats:**
  - Reads PREPBUFR using NCEP's BUFRLIB.
  - Writes point NetCDF as input to Point-Stat.

  Note: v2.0 no longer requires CWORDSH to pre-process PREPBUFR files.

PREPBUFR

- **BUFR** is the World Meteorological Organization (WMO) standard binary code for the representation and exchange of observational data.
  - **http://www.nco.ncep.noaa.gov/sib/decoders/BUFRLIB/**
  - **http://www.ecmwf.int/products/data/software/**

- The **PREPBUFR** format is produced by NCEP for analyses and data assimilation. The system that produces this format:
  - Assembles observations dumped from a number of sources
  - Encodes
    - information about the observational error for each data type
    - background (first guess) interpolated to each data location
  - Performs both rudimentary multi-platform quality control and more complex platform-specific quality control

- **MET currently only supports PREPBUFR.** Let us know if you need to read in other BUFR format.

What is in PREPBUFR file?

METv2.0/data/sample_obs/prepbufr/ndas.t00z.prepbufr.tm12.20070401.nr

```
==> append : to filename to view the  data source
BUFR
230ADPUPA   UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS
231AIRCAR   MDCRS ACARS AIRCRAFT REPORTS                    232AIRCFT
AIREP/PIREP, AMDAR(ASDAR/ACARS), E-ADAS(AMDAR BUFR) ACF233SATWND
SATELLITE-DERIVED WIND REPORTS                  234PROFLR   WIND PROFILER
REPORTS                         235VADWND   VAD (NEXRAD) WIND REPORTS
236SATEMP   TOVS SATELLITE DATA (SOUNDINGS, RETRIEVALS, RADIANCES)
237ADPSFC   SURFACE LAND (SYNOPTIC, METAR) REPORTS
238SFCSHP   SURFACE MARINE (SHIP, BUOY, C-MAN PLATFORM) REPORTS
239SFCBOG   MEAN SEA-LEVEL PRESSURE BOGUS REPORTS
240SPSSMI   SSM/I RETRIEVAL PRODUCTS (REPROCESSED WIND SPEED, TPW)
241SYNDAT   SYNTHETIC TROPICAL CYCLONE BOGUS REPORTS
242ERS1DA   ERS SCATTEROMETER DATA (REPROCESSED WIND SPEED)
243GOESND   GOES SATELLITE DATA (SOUNDINGS, RETRIEVALS, RADIANCES)
244QKSWND   QUIKSCAT SCATTEROMETER DATA (REPROCESSED WIND SPEED)
245MSONET   MESONET SURFACE REPORTS (COOPERATIVE NETWORKS)
246GPSIPW   GLOBAL POSITIONING SATELLITE-INTEGRATED PRECIP. WATER
247RASSDA   RADIO ACOUSTIC SOUNDING SYSTEM (RASS) TEMP PROFILE
RPTSM063000BYTCNT…
```

Result of running linux “less” or “more” command:

>less \
ndas.t00z.prepbufr.tm12.20070401.nr

PB2NC: Usage

Usage: pb2nc

prepbufr_file

netcdf_file

config_file

[-pbfile prepbufr_file]

[-valid_beg time]

[-valid_end time]

[-nmsg n]

[-dump path]

[-v level]

prepbufr_file	**Input obs file in PrepBufr format**
netcdf_file	**Output name for NetCDF file**
config_file	**PB2NC configuration file**
-pbfile	**Additional input PrepBufr files**
-valid_beg	**Beginning of valid time window** [YYYYMMDD_[HH[MMSS]]
-valid_end	**End of valid time window** [YYYYMMDD_[HH[MMSS]]
-nmsg	**Number of PrepBufr messages to process**
-dump	**Dump entire contents o PrepBufr file to file in path**
-v	**Level of logging**

PB2NC: Run

- **METv2.0/bin/pb2nc** \ ndas.t00z.prepbufr.tm12.20070401.nr \
  out/sample_pb.nc PB2NCConfig_tutorial -v 2

```
Reading Config File:    PB2NCConfig_default
Creating NetCDF File:   out/sample_pb.nc
Processing PrepBufr File:       ndas.t00z.prepbufr.tm09.20070401.nr
Blocking PrepBufr file to:      /tmp/pb2nc_1705_0_blk.pb
PrepBufr Time Center:   20070331_150000
Searching Time Window:  20070331_133000 to 20070331_163000
Processing 70884 PrepBufr messages...
5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60% 65% 70% 75% 80% 85% 90% 95%
100%
Total PrepBufr Messages processed       = 70884
Rejected based on message type          = 0
Rejected based on station id            = 0
Rejected based on valid time            = 0
Rejected based on masking grid          = 0
Rejected based on masking polygon       = 0
Rejected based on elevation             = 0
Rejected based on pb report type        = 0
Rejected based on input report type     = 0
Rejected based on instrument type       = 0
Rejected based on zero observations     = 24153
Total PrepBufr Messages retained        = 46731
Total observations retained or derived  = 142709
```

This example takes a few minutes to run

Output NetCDF file is designated as out/sample_pb.nc

We will see the NetCDF format in a few slides

PREPBUFR - Known Issues

- CWORDSH blocking
  - In v1.0 and v1.1 it was needed to structure to data properly for reading by the MET fortran code.
  - Need eliminated in v2.0.
- Compilation of PREPBUFR on 64-bit OS
  - If you will be using PREPBUFR files, you should compile everything using 32-bit flags.
  - Investigation of this problem in ongoing.

2. ASCII2NC Tool

- **Stands for "ASCII to NetCDF"**
- **Functionality:**
  - Reformat ASCII point observations into intermediate NetCDF format.
  - One input ASCII format supported (10 columns).
  - No configuration file.
- **Data formats:**
  - Reads MET specific ASCII format with point obs.
  - Writes point NetCDF as input to Point-Stat.
  - ***Future: support additional standard ASCII formats.***

ASCII2NC: Usage

Usage: ascii2nc

ascii_file

netcdf_file

[-format ascii_format]

[-v level]

ascii_file	**Input obs file in ASCII format**
netcdf_file	**Output name for NetCDF file**
-format	**Override using MET specific Point format** (*Future Option*)
-v	**Level of logging**

MET specific ASCII Format

```
Msg      STID  ValidTime        Lat      Lon      Elev  GC Lvl    Hgt  Ob
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 7 837.0 1618 1618
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 11 837.0 1618 273.05
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 17 837.0 1618 271.85
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 52 837.0 1618 92
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 53 837.0 1618 0.00417
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 7 826.0 1724 1724
ADPUPA 72365 20070331_120000 35.03 -106.62 1618.0 11 826.0 1724 274.55
```

* *Use a value of "-9999" to indicate missing data*

Msg	**Message type**
STID	**WMO Station ID**
ValidTime	**Valid time for observation**
Lat	**Latitude** [North]
Lon	**Longitude** [East]
Elev	**Elevation** [m] *(Note: currently not used by MET code so can be filled with -9999.)*
GC	**GRIB code for variable** ***(i.e. AccPrecip = 61; MSLP = 2; Temp = 11, etc…)*** http://www.cpc.ncep.noaa.gov/products/wesley/opn_gribtable.html
Lvl	**Pressure** [mb] or **Accumulation Interval** [hr]
Hgt	**Height above Mean Sea Level** [m – MSL] *(Note: currently not used by MET code so can be filled with -9999.)*
Ob	**Observed value**

ASCII2NC: Run

- **METv2.0/bin/ascii2nc** sample_obs.txt \
  sample_ascii.nc -v 2

This example has 2140 obs and takes seconds to run

Output NetCDF file

Result of ncdump –h

Result of ncdump –v obs_arr

```
netcdf sample_ascii {
dimensions:
        mxstr = 15 ;
        hdr_arr_len = 3 ;
        obs_arr_len = 5 ;
        nhdr = 5 ;
        nobs = UNLIMITED ; // (2140 currently)
variables:
        char hdr_typ(nhdr, mxstr) ;
                hdr_typ:long_name = "message type" ;
        char hdr_sid(nhdr, mxstr) ;
                hdr_sid:long_name = "station identification" ;
        char hdr_vld(nhdr, mxstr) ;
                hdr_vld:long_name = "valid time" ;
                hdr_vld:units = "YYYYMMDD_HHMMSS UTC" ;
        float hdr_arr(nhdr, hdr_arr_len) ;
                hdr_arr:long_name = "array of observation station header values" ;
                hdr_arr:_fill_value = -9999.f ;
                hdr_arr:columns = "lat lon elv" ;
                … ;
        float obs_arr(nobs, obs_arr_len) ;
                obs_arr:long_name = "array of observation values" ;
                obs_arr:_fill_value = -9999.f ;
                obs_arr:columns = "hdr_id gc lvl hgt ob" ;
                obs_arr:hdr_id_long_name = "index of matching header data" ;
                … ;
```

```
obs_arr =
0, 7, 837, 1618, 1618,
1, 11, 837, 1618, 273.05,
2, 17, 837, 1618, 271.85,
3, 52, 837, 1618, 92,
4, 53, 837, 1618, 0.00417,
5, 7, 826, 1724, 1724,
6, 11, 826, 1724, 274.55,
7, 17, 826, 1724, 272.15,
8, 52, 826, 1724, 84,
9, 53, 826, 1724, 0.00432,
10, 7, 815.3, 1829, 1829,
11, 11, 815.3, 1829, 276.45,
12, 17, 815.3, 1829, 265.75,
13, 52, 815.3, 1829, 45,
14, 53, 815.3, 1829, 0.0027,
15, 7, 815, 1832, 1832,
16, 11, 815, 1832, 276.55,
17, 17, 815, 1832, 265.55,
18, 52, 815, 1832, 44,
19, 53, 815, 1832, 0.00266,
20, 7, 784.7, 2134, 2134,
21, 11, 784.7, 2134, 274.05,
22, 17, 784.7, 2134, 264.15,
23, 52, 784.7, 2134, 47,
…
```

3. PCP-Combine Tool

- **Stands for "Precip-Combine"**
- **Functionality:**
  - Mathematically combines precipitation fields across multiple files.
    - Add precipitation over 2 files
      - *2 NMM output files to go from 3-hr to 6-hr accumulation.*
    - Sum precipitation over more than 2 files
      - *12 WSR-88D Level II data to go from 5 min accumulation to 1-hr accumulation.*
    - Subtract precipitation in 2 files
      - *2 ARW output files to go from 12 hr accumulations to 6 hour accumulation*
  - Specify field name on the command line.
  - No configuration file.
- **Data formats:**
  - Reads GRIB format.
  - Writes gridded NetCDF as input to stats tools.

PCP-Combine: Usage

Usage: pcp_combine

[-sum sum_args]

or [-add add_args]

or [-subtract sub_args]

[-gc code]

[-v level]

-sum	**Accumulates data over multiple files.** ***Sum_args***: (init_time, in_accum, valid_time, out_accum, out_file, -pcpdir path, -pcprx reg_exp)
-add	**Accumulates data over two files.** ***Add_args:*** (in_file1, Accum1, in_file2, Accum2, out_file).
-subtract	**Subtracts data over two files.** ***Sub_args:*** (in_file1, Accum1, in_file2, Accum2, out_file).
-gc	**GRIB code for variable** *(i.e. ACPC = 61; NCPCP* =62*; ACPCP* =63*, etc…).*
-v	**Level of logging**

PCP-Combine: Run

Two example command lines

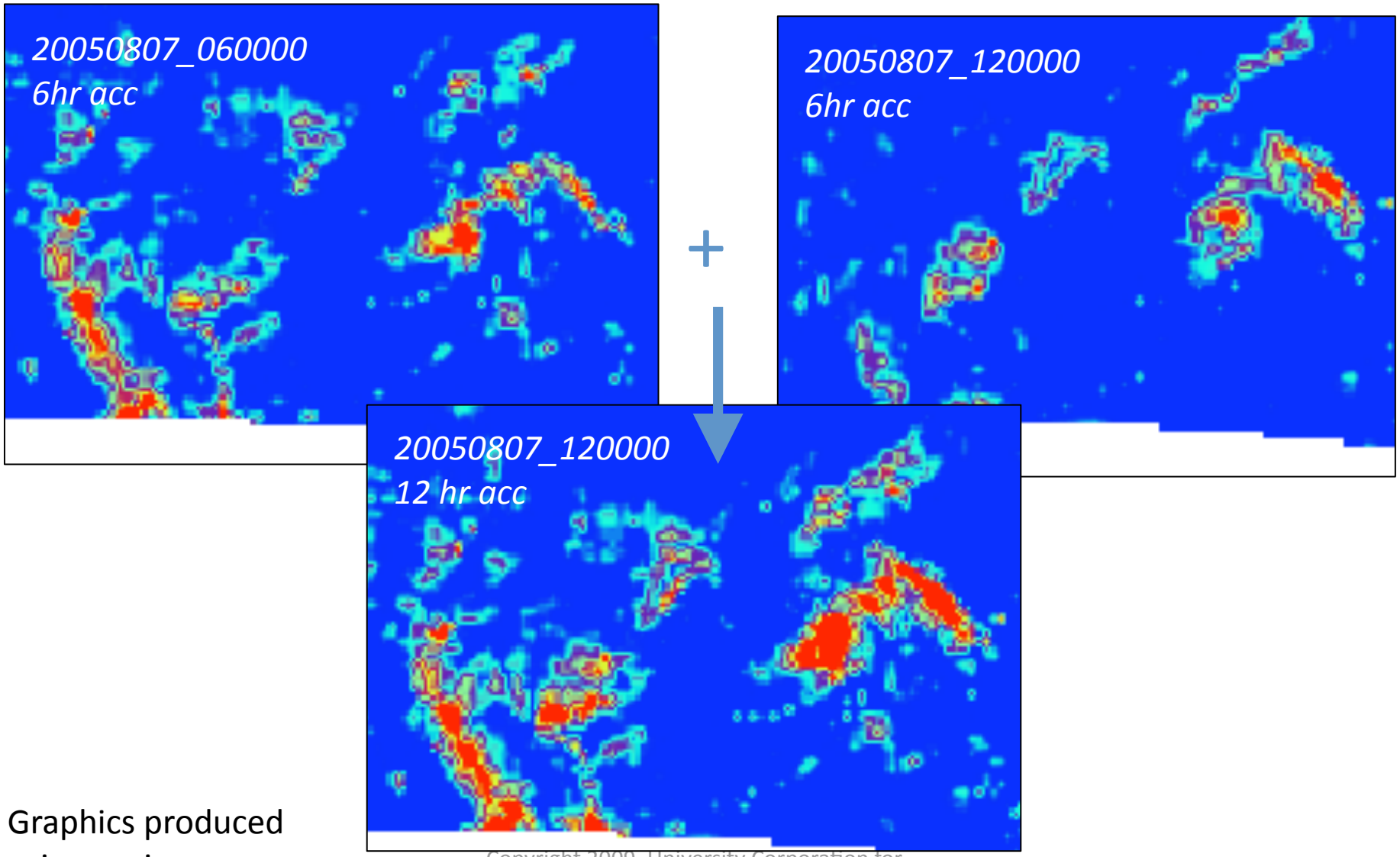
1) **Adding 2 (6-hourly) accumulation forecast files into 1 (12-hourly) accumulation forecast.**

```
METv2.0/bin/pcp_combine \
-add 20050807_060000.grb 6 \
20050807_120000.grb 6 \
sample_fcst.nc -pcpdir data/2005080700
```

2) **Summing 12 (1-hourly) accumulation observation files into 1 (12-hourly) accumulated observation file.**

```
METv2.0/bin/pcp_combine \
-sum 00000000_000000 1 \
20050807_120000 12 \
sample_obs.nc -pcpdir data/ST2ml
```

PCP-Combine: Example #1



Graphics produced using ncview

GRIB vs GRIB2

- **GRIB (or GRIB1): WRF postprocessor (WPP) produces GRIB1 format using copyGB tool**
  - Performs horizontal interpolation and destaggering (in the case of WRF-NMM) onto a defined grid.
  - Useful for both cores in creating an output grid not fixed by the model integration domain.
    - **http://www.dtcenter.org/wrf-nmm/users/downloads/**
- **GRIB2: NCEP and other WMO organizations have historical data in GRIB1 but now use GRIB 2 as the standard for gridded binary data.**
- **At least two GRIB2toGRIB1 converters available**
  - **NCEP cnvgrib**
    - **http://www.nco.ncep.noaa.gov/pmb/codes/GRIB2/**
  - **UCAR/CISL Grib Converter**
    - **http://dss.ucar.edu/libraries/grib/c.html**

Data Inventory Tools

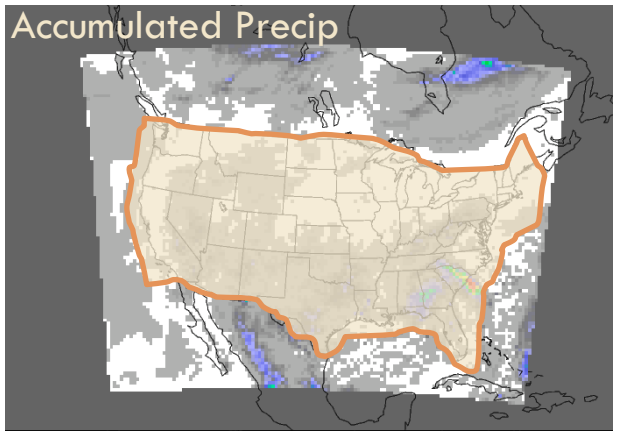
- **wgrib** – dumps GRIB1 headers and data.
  - http://www.cpc.ncep.noaa.gov/products/wesley/wgrib.html
- **wgrib2** – dumps GRIB2 headers and data.
  - http://www.cpc.ncep.noaa.gov/products/wesley/wgrib2/
- **ncdump -** dumps NetCDF headers and data.
  ncview – plots gridded NetCDF data.
  - http://www.unidata.ucar.edu/software/netcdf/
- **GrADS** – command line interface to produce plots.
  - http://www.iges.org/grads/downloads.html
- **NCL** – command line interface to produce plots.
  - http://www.ncl.ucar.edu/
- **IDV** – gui-driven visualization of many gridded and point datasets.
  - http://www.unidata.ucar.edu/software/idv/

4. Gen Poly Mask Tool

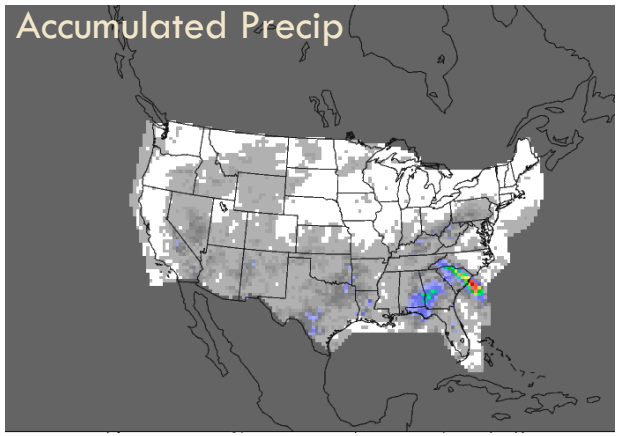
- **Stands for "Generate Polyline Mask"**

- **Functionality:**
  - Uses a lat/lon polyline to generate a 0/1 mask field to be applied to your data.
  - Applies this mask once – prior to running Point-Stat or Grid-Stat
  - No configuration file.

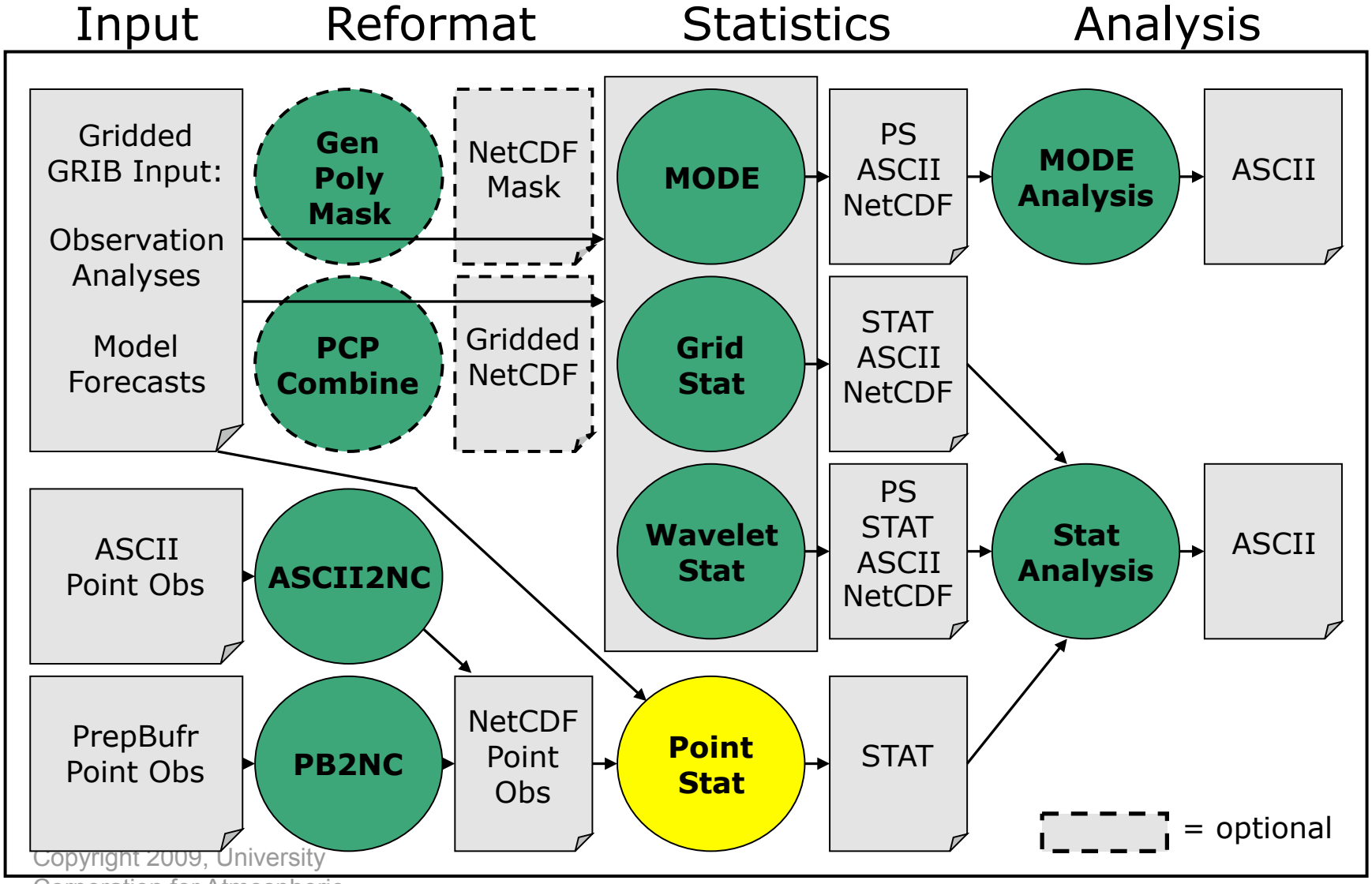
- **Data formats:**
  - Reads ASCII formatted polyline file.
  - Reads GRIB file.
  - Reads NetCDF files from PCP-Combine.
  - Writes gridded NetCDF file of 0/1 mask.



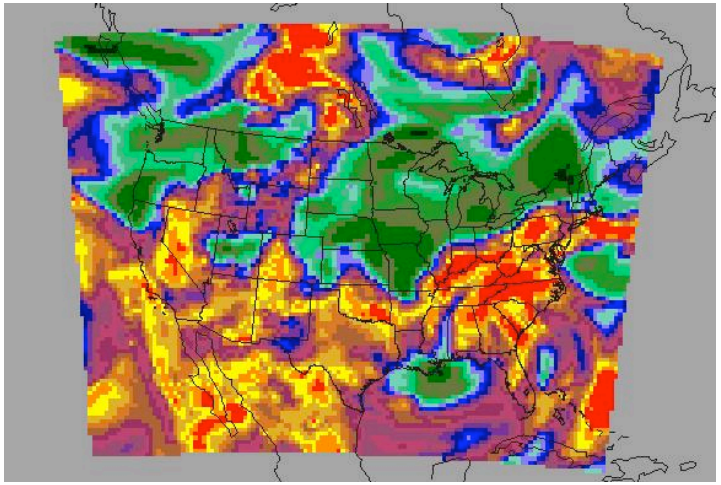

```
CONUS
31.1931 -120.4211
31.2291 -120.4976
31.2650 -120.5741
31.3009 -120.6123
31.3369 -120.6506
31.3728 -120.6888
31.4087 -120.6888
31.4447 -120.7270
...
```



Point-Stat Tool

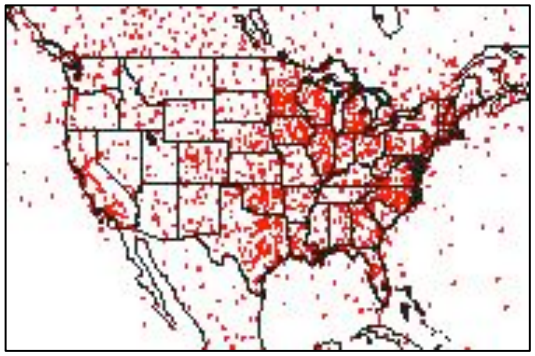


Point-Stat: Overview



- Compare **gridded forecasts** to **point observations**.
- Accumulate matched pairs over a defined area at a single point in time.
- Verify one or more variables/ levels.
- Analysis tool provided to aggregate through time.

- Verification methods:
  - Continuous statistics for raw fields.
  - Categorical counts and statistics for thresholded fields.
  - Parametric and non-parametric confidence intervals for statistics.
  - Compute partial sums for raw fields and/or the raw matched pair values.
  - Methods for probability forecasts.



Point-Stat: Input/Output

- Input Files
  - Gridded forecast file
    - GRIB output of WRF Post-Processor (or other)
    - NetCDF output of PCP-Combine
  - Point observation file
    - NetCDF output of PB2NC
    - NetCDF output of ASCII2NC
  - ASCII configuration file
- Output Files
  - ASCII statistics file with all output lines (end with ".stat")
  - Optional ASCII files sorted by line type with a header row (ends with "_TYPE.txt")

Point-Stat: Usage

Usage: point_stat

fcst_file

obs_file

config_file

[-climo climo_file]

[-ncfile netcdf_file]

[-valid_beg time]

[-valid_end time]

[-outdir path]

[-v level]

fcst_file	Forecast file in GRIB or NetCDF
obs_file	Point observation file in NetCDF (PB2NC or ASCII2NC)
config_file	ASCII configuration file
-climo	Climatological file for computing anomaly partial sums
-ncfile	Additional point observation files in NetCDF
-valid_beg	Beginning of valid time window for matching
-valid_end	End of valid time window for matching
-outdir	Output directory to be used
-v	Level of logging

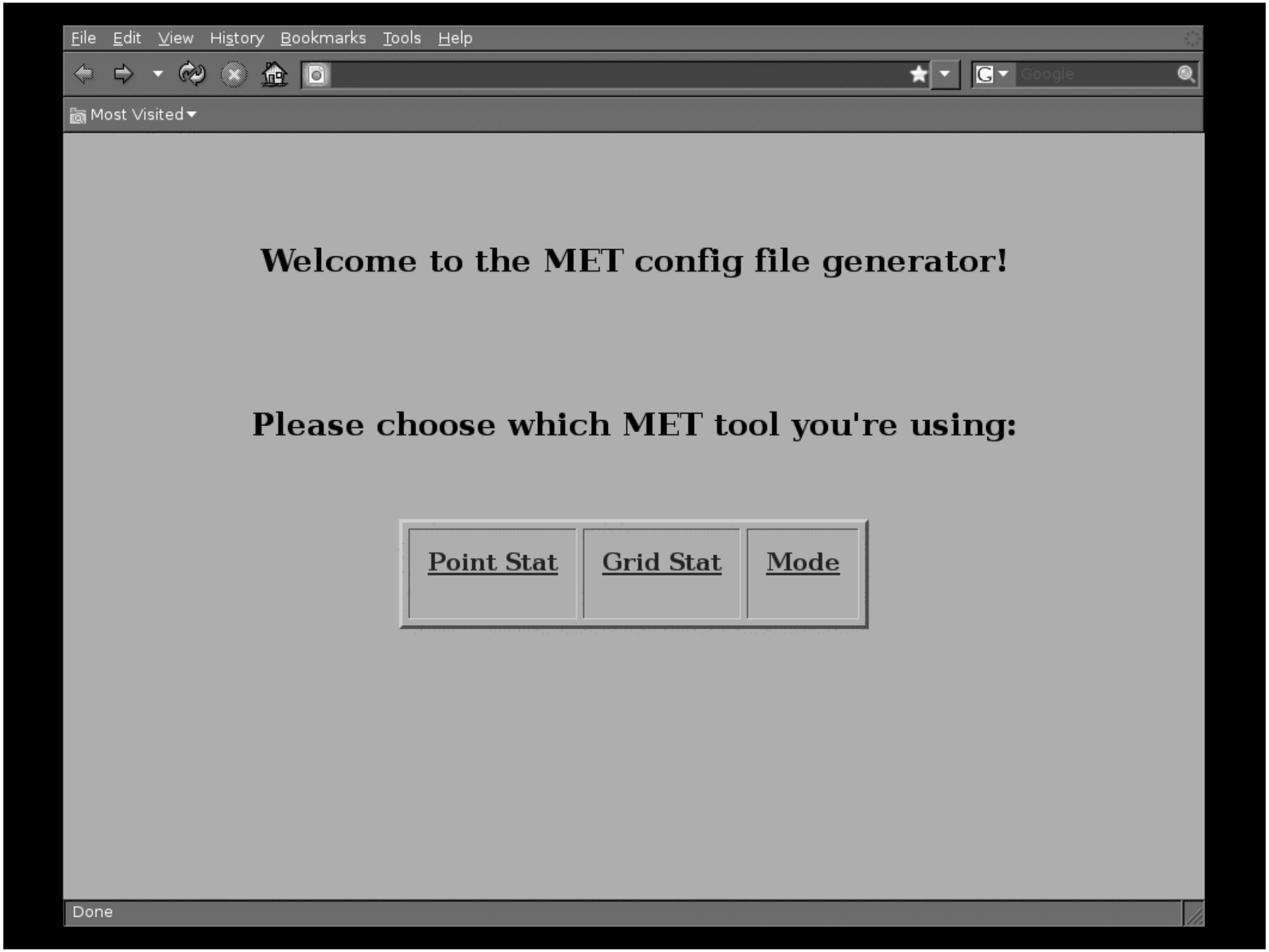
Point-Stat: Configuration

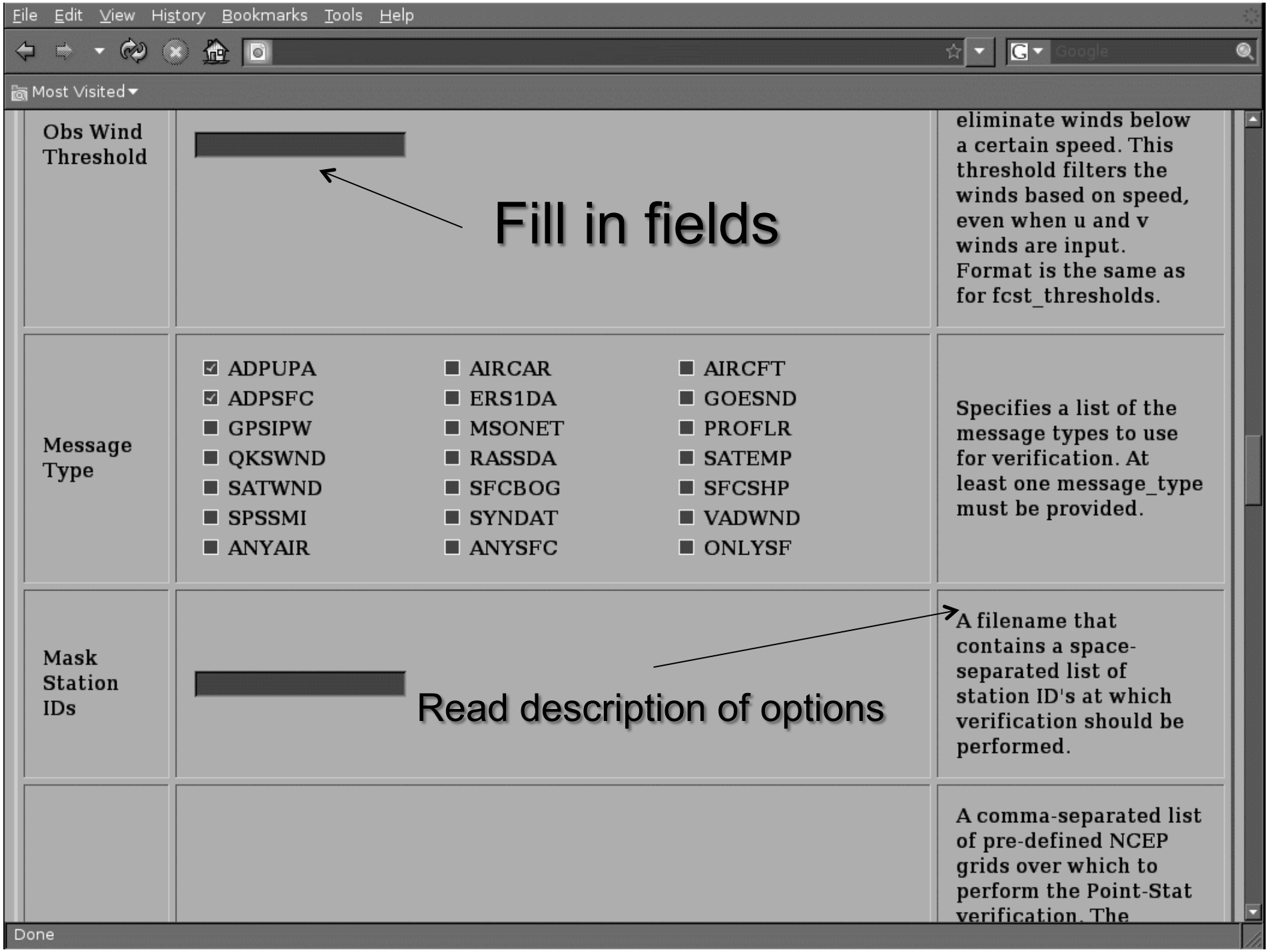
- 28 configurable parameters – only set a few:
  - Temperature at the surface (2-meter).
    - **fcst_field[] = ["TMP/Z2“];**
  - Temperature below freezing.
    - **fcst_thresh[] = [“le273“];**
  - Match to observations at the surface.
    - **message_type[] = ["ADPSFC"];**
  - Look at all the points in my domain.
    - **mask_grid[] = ["FULL"];**
  - Match observation to the nearest forecast value.
    - **interp_wdth[] = [1];**
  - Generate all possible statistic types, except probabilistic.
    - **output_flag[] = [2, 2, 2, 2, 2, 2, 2, 2, 0, 0, 0, 0, 2];**

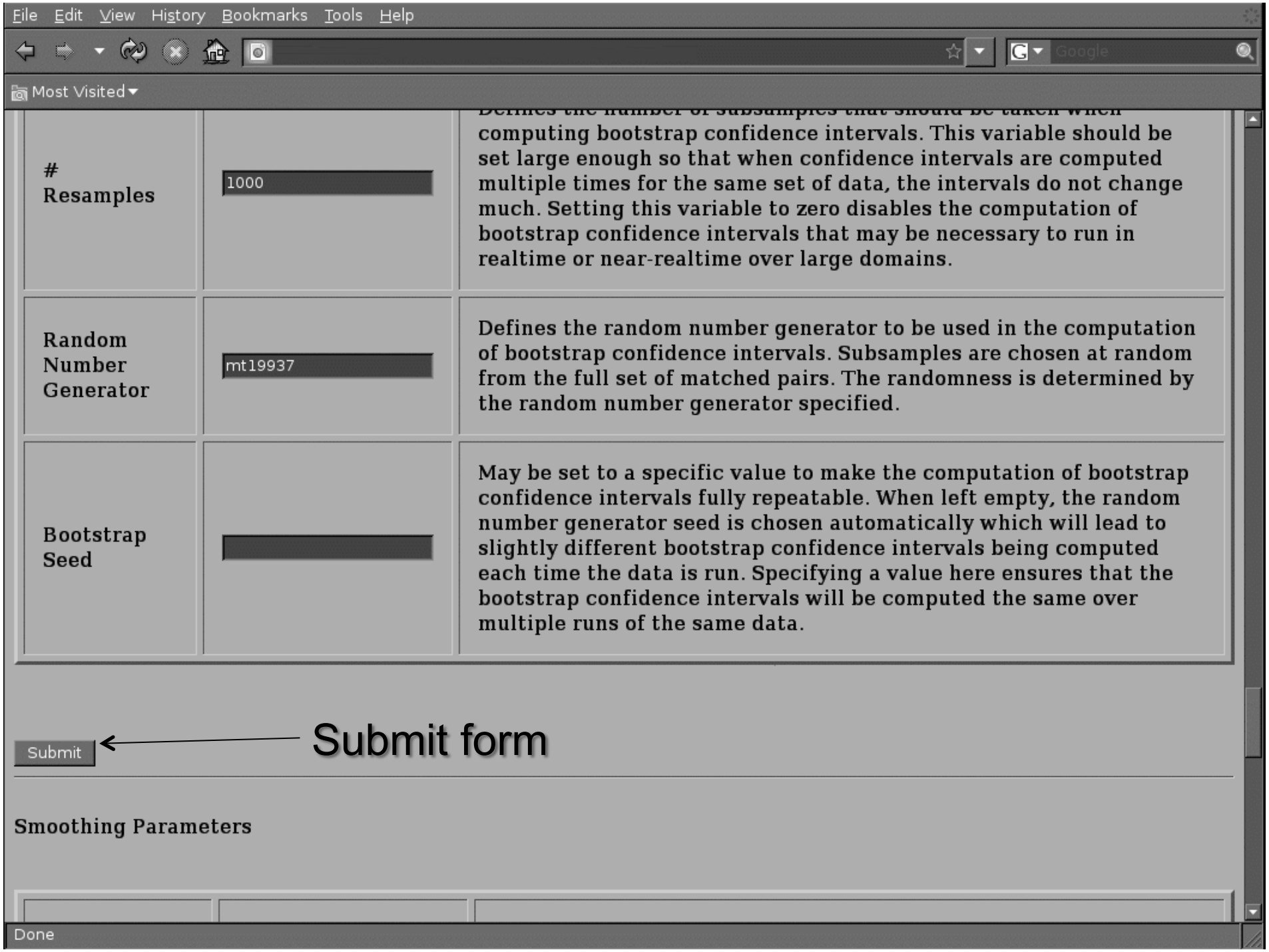
MET Config File GUI

- Many MET utilities require a config file to specify options.
- Now a web GUI exists to assist in creation of some of these files.
  - http://verif.rap.ucar.edu/cgi/metgui/base.cgi
- All fields are described.
- Fill in blank fields.
- Submit to get ascii config file, then save from browser to your machine.

```
////////////////////////////////////////////////////////////////////////
//
// Point Stat Config File
//
//   Generated by MET Config File Web Utility
//
//   June 23, 2009   12:33 pm  MDT
//
////////////////////////////////////////////////////////////////////////

model = "WRF";

beg_ds = -5400;

end_ds = -5400;

fcst_field [] = [ "TMP/P750-900", "UGRD/Z10", "VGRD/Z10" ];

obs_field [] = [];

fcst_thresh [] = [ "lt273 ge273", "ge5", "ge5" ];

obs_thresh [] = [];

fcst_wind_thresh [] = [ "NA" ];

obs_wind_thresh [] = [];

message_type [] = [ "ADPUPA", "ADPSFC" ];

mask_grid [] = [ "DTC165", "DTC166" ];

mask_poly [] = [ "MET_BASE/data/poly/LMV.poly" ];

mask_sid = "";

ci_alpha [] = [ 0.10, 0.05 ];

boot_interval = 1;

boot_rep_prop = 1.00;

n_boot_rep = 1000;

boot_rng = "mt19937";
```


File Edit View History Bookmarks Tools Help
Google
Most Visited
Welcome to the MET config file generator!
Please choose which MET tool you're using:
Point Stat
Grid Stat
Mode
Done


File Edit View History Bookmarks Tools Help
Most Visited
Obs Wind Threshold
Fill in fields
eliminate winds below a certain speed. This threshold filters the winds based on speed, even when u and v winds are input. Format is the same as for fcst_thresholds.
Message Type
ADPUPA
ADPSFC
GPSIPW
QKSWND
SATWND
SPSSMI
ANYAIR
AIRCAR
ERS1DA
MSONET
RASSDA
SFCBOG
SYNDAT
ANYSFC
AIRCFT
GOESND
PROFLR
SATEMP
SFCSHP
VADWND
ONLYSF
Specifies a list of the message types to use for verification. At least one message_type must be provided.
Mask Station IDs
Read description of options
A filename that contains a space-separated list of station ID's at which verification should be performed.
A comma-separated list of pre-defined NCEP grids over which to perform the Point-Stat verification. The
Done


File Edit View History Bookmarks Tools Help
Most Visited
Resamples
1000
computing bootstrap confidence intervals. This variable should be set large enough so that when confidence intervals are computed multiple times for the same set of data, the intervals do not change much. Setting this variable to zero disables the computation of bootstrap confidence intervals that may be necessary to run in realtime or near-realtime over large domains.
Random Number Generator
mt19937
Defines the random number generator to be used in the computation of bootstrap confidence intervals. Subsamples are chosen at random from the full set of matched pairs. The randomness is determined by the random number generator specified.
Bootstrap Seed
May be set to a specific value to make the computation of bootstrap confidence intervals fully repeatable. When left empty, the random number generator seed is chosen automatically which will lead to slightly different bootstrap confidence intervals being computed each time the data is run. Specifying a value here ensures that the bootstrap confidence intervals will be computed the same over multiple runs of the same data.
Submit
Submit form
Smoothing Parameters
Done

```
////////////////////////////////////////////////////////////////////////
//
// Point Stat Config File
//
//   Generated by MET Config File Web Utility
//
//   June 23, 2009   12:33 pm  MDT
//
////////////////////////////////////////////////////////////////////////

model = "WRF";

beg_ds = -5400;

end_ds = -5400;

fcst_field [] = [ "TMP/P750-900", "UGRD/Z10", "VGRD/Z10" ];

obs_field [] = [];

fcst_thresh [] = [ "lt273 ge273", "ge5", "ge5" ];

obs_thresh [] = [];

fcst_wind_thresh [] = [ "NA" ];

obs_wind_thresh [] = [];

message_type [] = [ "ADPUPA", "ADPSFC" ];

mask_grid [] = [ "DTC165", "DTC166" ];

mask_poly [] = [ "MET_BASE/data/poly/LMV.poly" ];

mask_sid = "";

ci_alpha [] = [ 0.10, 0.05 ];

boot_interval = 1;

boot_rep_prop = 1.00;

n_boot_rep = 1000;
```

Resulting file

Save to your computer.

Point-Stat: Run

- METv2.0/bin/point_stat \
  sample_fcst.grb sample_pb.nc \
  PointStatConfig_TMPZ2 -outdir out -v 2

```
Forecast File: sample_fcst.grb
Climatology File: none
Configuration File: PointStatConfig_TMPZ2
Observation File: sample_pb.nc
----------------------------------------
Reading records for TMP/Z2.
For TMP/Z2 found 1 forecast levels and 0 climatology levels.
----------------------------------------
Searching 179772 observations from 44076 PrepBufr messages.
----------------------------------------
Processing TMP/Z2 versus TMP/Z2, for observation type ADPSFC, over region FULL, for interpolation method
UW_MEAN(1), using 11370 pairs.
Computing Categorical Statistics.
Computing Continuous Statistics.
Computing Scalar Partial Sums.
----------------------------------------
Output file: out/point_stat_360000L_20070331_120000V.stat
Output file: out/point_stat_360000L_20070331_120000V_fho.txt
Output file: out/point_stat_360000L_20070331_120000V_ctc.txt
Output file: out/point_stat_360000L_20070331_120000V_cts.txt
Output file: out/point_stat_360000L_20070331_120000V_cnt.txt
Output file: out/point_stat_360000L_20070331_120000V_sl1l2.txt
Output file: out/point_stat_360000L_20070331_120000V_sal1l2.txt
Output file: out/point_stat_360000L_20070331_120000V_vl1l2.txt
Output file: out/point_stat_360000L_20070331_120000V_val1l2.txt
Output file: out/point_stat_360000L_20070331_120000V_mpr.txt
```

Point-Stat: ASCII Output Types

- Statistics line types: 13 possible
  - Categorical - apply threshold
    - Contingency table counts and stats (FHO, CTC, CTS)
  - Continuous - raw fields
    - Continuous statistics (CNT)
    - Partial Sums (SL1L2, SAL1L2, VL1L2, VAL1L2)
  - Probabilistic
    - Contingency table counts and stats (PCT, PSTD)
    - Continuous statistics and ROC curve (PJC, PRC)
  - Matched pairs
    - Raw matched pairs – a lot of data! (MPR)
- 21 header columns common to all line types
- Remaining columns specific to each line type

Point-Stat: Sample Output

- STAT file output for sample run:
  - 1 line each for FHO, CTC, CTS, CNT, and SL1L2
  - 11,370 lines for MPR!
- Additional TXT files for each line type

Point-Stat: CTC Output Line

VERSION	V2.0
MODEL	WRF
FCST_LEAD	360000
FCST_VALID_BEG	20070331_120000
FCST_VALID_END	20070331_120000
OBS_LEAD	000000
OBS_VALID_BEG	20070331_103000
OBS_VALID_END	20070331_133000
FCST_VAR	TMP
FCST_LEV	Z2
OBS_VAR	TMP
OBS_LEV	Z2
OBTYPE	ADPSFC

VX_MASK	FULL
INTERP_MTHD	UW_MEAN
INTERP_PNTS	1
FCST_THRESH	<273.000
OBS_THRESH	<273.000
COV_THRESH	NA
ALPHA	NA
LINE_TYPE	CTC
TOTAL	11370
FY_OY	1635
FY_ON	380
FN_OY	438
FN_ON	8917

Point-Stat: Matched Pairs

- **Matched Pair (MPR) line type contains raw matched pairs.**
  - **Data overload!**

TOTAL	INDEX	OBS_LAT	OBS_LON	OBS_LVL	OBS_ELV	FCST	OBS	CLIMO
11370	1	43.93000	-60.01000	1010.79999	4.01053	271.87788	271.54999	NA
11370	2	46.43000	-71.93000	1016.09998	102.04903	268.50255	269.45001	NA
11370	3	44.23000	-78.36000	1004.50000	191.44466	272.94013	272.35001	NA
11370	4	51.67000	-124.40000	916.50000	872.82202	263.69020	264.95001	NA
11370	5	58.61000	-117.16000	973.90002	337.50449	272.37757	270.95001	NA
11370	6	52.18000	-122.04000	906.50000	938.08594	271.37738	264.35001	NA
11370	7	50.68000	-127.36000	1020.20001	22.03931	275.44020	275.04999	NA
11370	8	50.45000	-100.59000	949.09998	562.38477	272.18978	271.75000	NA
11370	9	57.13000	-61.47000	899.70001	834.87476	258.06464	254.64999	NA
11370	10	47.56000	-59.16000	1000.90002	40.06803	272.06486	269.54999	NA
11370	11	47.41000	-72.79000	1006.90002	169.37592	266.37724	265.95001	NA
11370	12	45.76000	-62.68000	1014.00000	1.99518	268.94018	268.64999	NA
11370	13	49.24000	-65.33000	1014.90002	28.96468	264.25276	267.25000	NA
11370	14	43.29000	-79.79000	1017.79999	77.03765	273.56474	275.85001	NA
11370	15	48.78000	-123.04000	1015.70001	23.93772	278.12724	280.25000	NA

Comparing Different Fields

- Grid-Stat and Point-Stat may be used to compare two different variables.
  - User must interpret results.
  - Example: Total Precip vs. Convective Precip
  - Configuration file settings:
    - Selecting variable/levels:
      - **fcst_field[] = [“APCP/A24"];**
      - **obs_field[] = [“ACPCP/A24”];**
    - Selecting thresholds:
      - **fcst_thresh[] = ["gt0.0 ge20.0"];**
      - **obs_thresh[] = []; (leave blank to use fcst setting)**

Verifying Winds

- Verify speed pre-METv2.0, but no direction.
- Incremental support for verification of winds:
  - Enhancements for Point-Stat and Grid-Stat:
    - Add wind speed thresholds to determine which U/V pairs are included in the vector partial sums (VL1L2).
  - Enhancements for Stat-Analysis:
    - Support new job to aggregate one or more vector partial sum lines and compute statistics for the wind direction errors.
      - Mean forecast and observation wind directions, mean error (F-O), and mean absolute error

Verifying Winds: Example

Point-Stat: VL1L2 Lines

VX_MASK	THRESH	LINE_TYPE	TOTAL	UFBAR	VFBAR	UOBAR	VOBAR	UVFOBAR	UVFFBAR	UVOOBAR
DTC_165	>=1.000	VL1L2	653	1.91117	0.07900	1.40658	-0.06126	13.01039	18.12575	20.31649
DTC_165	>=3.000	VL1L2	279	3.13561	-0.35096	2.87061	-0.30072	26.50472	30.03257	38.25362
DTC_165	>=5.000	VL1L2	96	5.21268	-2.74580	5.47813	-2.01667	49.90791	51.10427	70.78802
DTC_166	>=1.000	VL1L2	2431	-1.62742	0.25391	-1.23402	-0.04393	18.48309	29.70179	21.89615
DTC_166	>=3.000	VL1L2	1610	-1.84581	0.16061	-1.47491	-0.11217	24.45214	36.67400	29.36032
DTC_166	>=5.000	VL1L2	520	-0.93518	-0.45435	-0.25923	-0.49558	37.21821	52.51917	47.26483

Stat-Analysis: aggregate_stat jobs

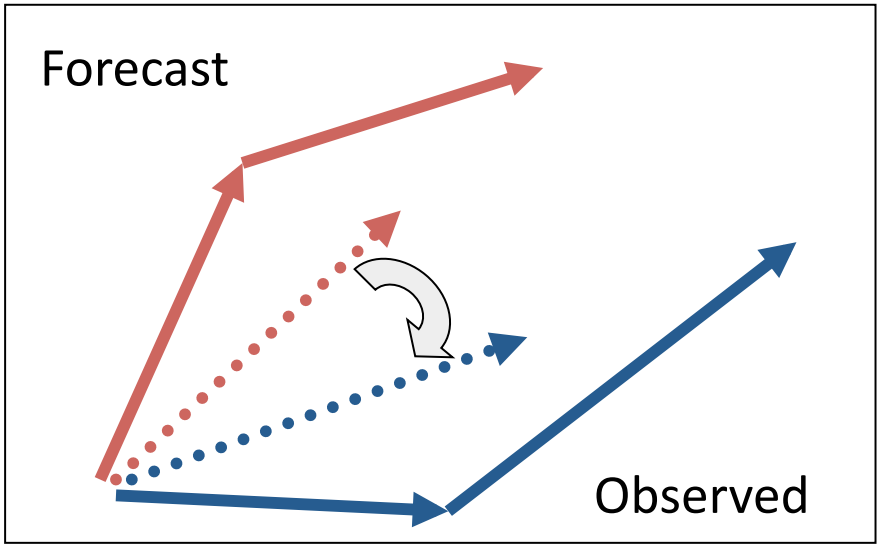
```
JOB_LIST:      -job aggregate_stat -fcst_thresh >=1.000 -line_type VL1L2 -out_line_type WDIR
     COL_NAME: TOTAL FBAR      OBAR     ME        MAE
ROW_MEAN_WDIR: 2     183.25038 0.22749  -3.02289  7.88372
    AGGR_WDIR: 3084  103.87238 85.96574 -17.90663 NA
```

```
JOB_LIST:      -job aggregate_stat -fcst_thresh >=3.000 -line_type VL1L2 -out_line_type WDIR
     COL_NAME: TOTAL FBAR     OBAR     ME        MAE
ROW_MEAN_WDIR: 2     5.67967  0.81565  -4.86402  4.86402
    AGGR_WDIR: 1889  94.38140 80.45939 -13.92200 NA
```

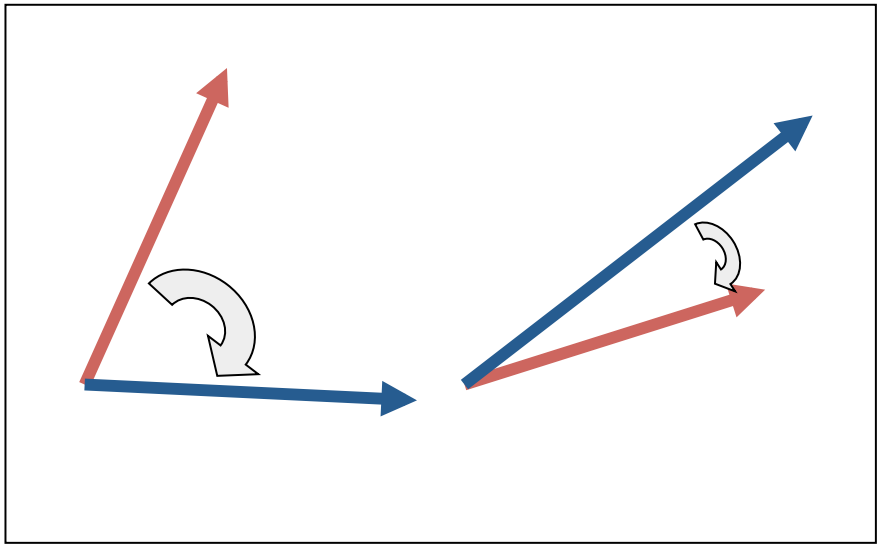
```
JOB_LIST:      -job aggregate_stat -fcst_thresh >=5.000 -line_type VL1L2 -out_line_type WDIR
     COL_NAME: TOTAL FBAR      OBAR      ME        MAE
ROW_MEAN_WDIR: 2     0.93288   338.91179 -22.02109 22.02109
    AGGR_WDIR: 616   358.38152 319.08761 -39.29391 NA
```

Verifying Winds: Output

- AGGR_WDIR
  1. Aggregate VL1L2 partial sums lines
  2. Derive wind directions and errors



- ROW_MEAN_WDIR
  1. Derive wind directions and errors for each VL1L2 line
  2. Compute mean of errors



Stat Analysis Tool

- Filtering
- Summarizing
- Aggregating

of Grid-Stat, Point-Stat,
& Wavelet-Stat output

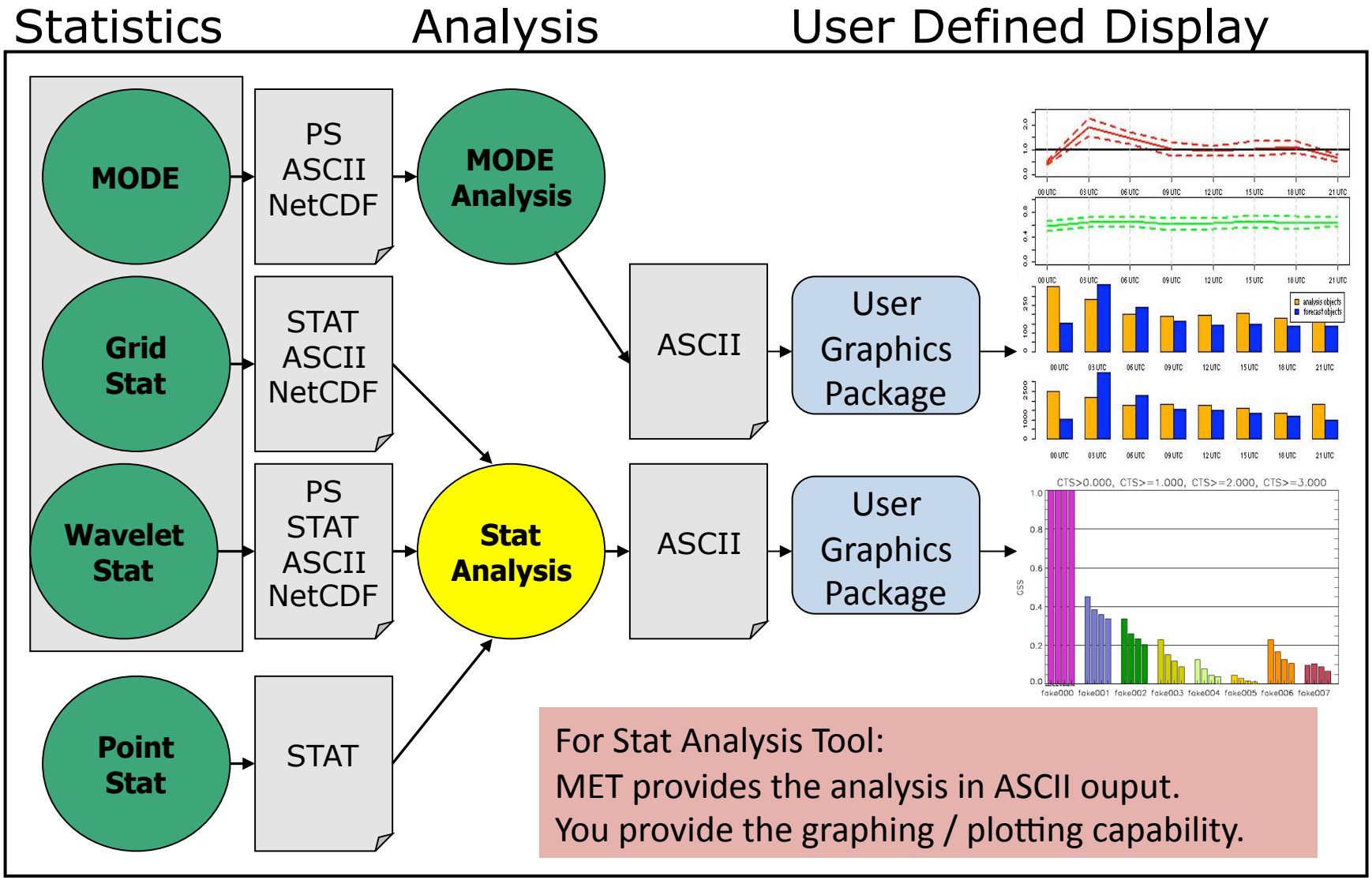
Presenter: Tara Jensen

What can Stat Analysis do for you?

Can I get…

- ✓ **Q:** Overall statistics for all gridded observations compared to the forecasts for hours 0 through 24 together?
  A: Yes - using Stat Analysis Tool on Grid-Stat output
- ✓ **Q:** Long-term statistics at individual sites (e.g., mean absolute error or RMS error for daily forecasts for a month).?
  A: Yes - using Stat Analysis Tool on Point-Stat output
- ✓ **Q:** My contingency table statistics aggregated over multiple runs?
  A: Yes – using Stat Analysis Tool on any output
- ✓ **Q:** Statistics aggregated for a large number (N) of individual stations in one simultaneous run?
  A: Yes – but it would be cumbersome. You would have to configure Stat Analysis Tool to run (N) number of jobs.

Stat Analysis Tool



Stat Analysis Jobs

- **Filtering**
  - **filter -** filters out lines from one or more stat files based on user-specified filtering options.
- **Summarizing**
  - **summary -** produces summary information from a single column of data including:

    mean, standard deviation, min, max, and the 10th, 25th, 50th, 75th, and 90th percentiles.
- **Customized tool for AFWA**
  - **stat_job_go_index -** computes the GO Index, a performance statistic used primarily by the United States Air Force.

Stat Analysis Jobs

- **Aggregation**
  - **aggregate -** aggregates stat data across multiple time steps or masking regions. Output line type is the same as input line type.
  - **aggregate_stat –** also aggregates across multiple time steps or masking regions. Output line sis typically different from input line types.

Valid line type combinations include:

-line_type FHO, CTC, -out_line_type CTS
-line_type SL1L2, SAL1L2, -out_line_type CNT
-line_type VL1L2, VAL1L2, -out_line_type WDIR (wind direction)
-line_type PCT, -out_line_type PSTD, PJC, PRC
-line_type NBRCTC, -out_line_type NBRCTS
-line_type MPR, -out_line_type FHO, CTC, CTS, CNT SL1L2, SAL1L2,PCT, PSTD, PJC, PRC

Stat Analysis Tool: Usage

Usage: stat_analysis

-lookin path

[-out filename]

[-v level]

-config config_file

or –jobs command line options with associated arguments

[filter]

[summary]

[aggregate]

[aggregate_stat]

[stat_job_go_index]

-lookin	Path to *.stat files – this can be a directory or a single file name (Use one or more times)
-out	Output name for ASCII file
-v	Level of logging
-config	StatAnalysisConfig file
filter	See previous 2 slides
summary	See previous 2 slides
aggregate	See previous 2 slides
aggregate_stat	See previous 2 slides
stat_job_go_index	See previous 2 slides

Stat Analysis Tool: Configuration

- 22 configurable parameters – only set a few:
  - Apply NAM G212 mask
    - **vx_mask[ ] = [“G212”];**
  - Using only the Temperature variable
    - **fcst_var[ ] = [“TMP”];**
      obs_var[ ] = [];
  - Filter on CTC lines in which fcst_var[ ] > 278 ºK
    - **line_type[ ] = [“CTC"];**
    - **fcst_thresh[ ] = [“>278”];**
      obs_thresh[ ] = [];
  - Dump the filtered stat data to a file
    AND sum contingency table count (CTC) lines of data
    for pressure levels between 850 and 750
    - **jobs[ ] = *["-job filter -dump_row out/filter_job.stat", ***
      ***"-job aggregate –line_type CTC ***
      ***-dump_row out/aggr_ctc_job.stat –fcst_lev P850-750"*];**

Stat Analysis Tool: Run stat_job_aggr

"-job aggregate –line_type CTC -dump_row out/aggr_ctc_job.stat \ -fcst_lev P850-750"

Point Stat Output *(i.e. pointstat_out.stat)*

```
V2.0     WRF    … ADPUPA G212 … TMP
    P850-750 … >278.00 CTC
    401      192      11      24      174
    UW_MEAN 1
```

```
V2.0     WRF    … ADPSFC G212 … TMP
    P850-750 … >278.00 CTC
    167      25      23      0      119
    UW_MEAN  1
```

(NOTE: header modified to show only pertinent info)

		OBS		
FCST		Y	N	
	Y	192	11	203
	N	24	174	198
		216	185	401

		OBS		
FCST		Y	N	
	Y	25	23	48
	N	0	119	119
		25	142	167

Stat Analysis Tool: Run stat_job_aggr

Stat Analysis Output *(i.e. aggr_ctc_job.stat)*

```
FILTER:  -job aggregate
    -vx_mask G212 -line_type CTC
    -fcst_thresh >278.000 -var TMP
    -dump_row out/filter_job.stat

JOB_LIST: -job aggregate
-vx_mask G212 -line_type CTC
    -fcst_thresh >278.000 -var TMP
    -level P850-750 -dump_row out/
    aggr_ctc_job.stat
COL_NAME:        TOTAL
    FY_OY            FY_ON
    FN_OY          FN_ON
    INTERP_MTHD      INTERP_PNTS
CTC:             568               217
    34              24          293

    -9999                      -9999
```

		OBS		
FCST		Y	N	
	Y	217	34	251
	N	24	293	317
		241	327	568

Stat Analysis Tool: Run stat_job_summary

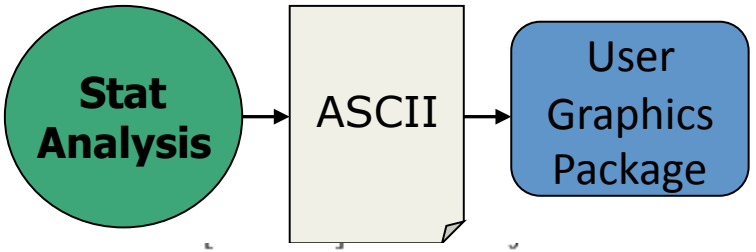
***"-job summary –line_type CNT -fcst_var TMP ***
–dump_row out/job_summary_RMSE.stat –column RMSE"

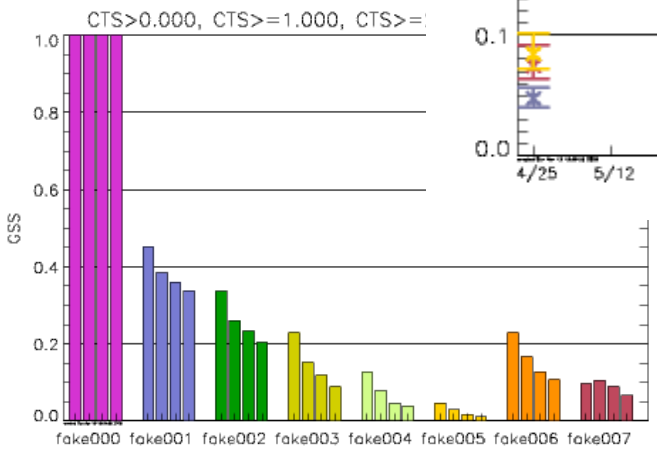
Column Number	Description
1	Summary (job type)
2	Total
3-7	Mean* *Includes normal and bootstrap upper and lower confidence limits*
8-10	Standard deviation** *Includes bootstrap upper and lower confidence limits*
11	Minimum value
12	10th percentile
13	25th percentile
14	Median (50th percentile)
15	75th percentile
16	90th percentile
17	Maximum value

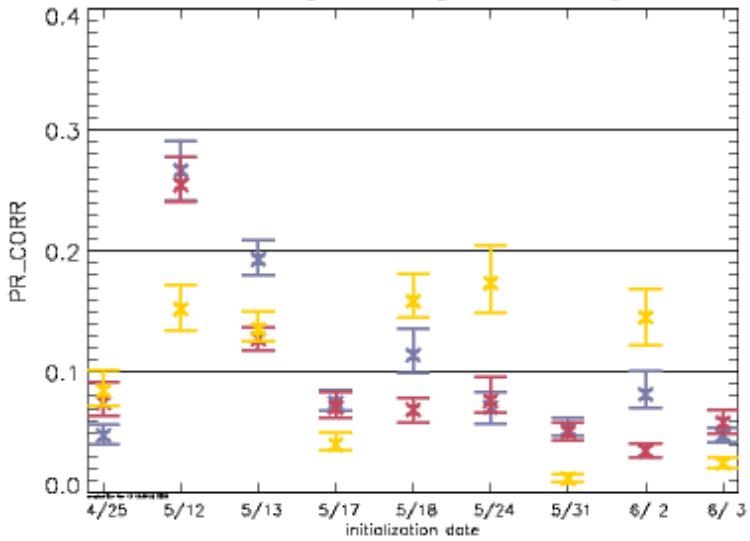
(stat_analysis.out cont.)

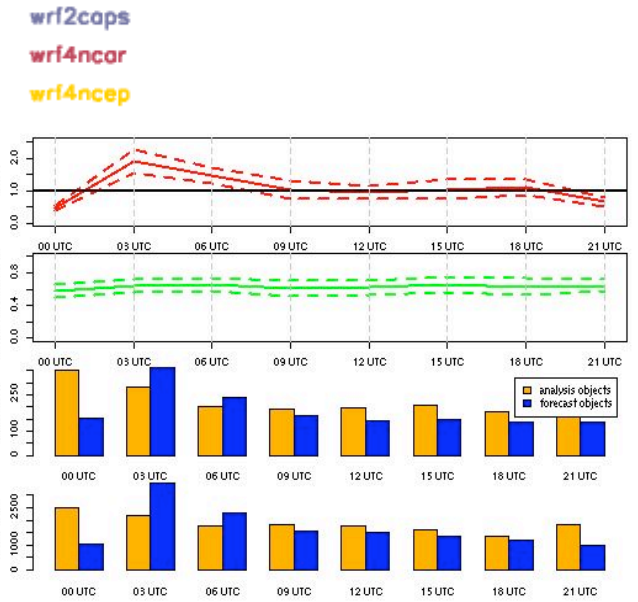
```
JOB_LIST: -job
stat_job_summary -line_type
CNT ….
COL_NAME: TOTAL MEAN
MEAN_NCL MEAN_NCU MEAN_BCL
MEAN_BCU STDEV STDEV_BCL
STDEV_BCU MIN  P10
P25 P50 P75
P90 MAX
SUMMARY: 4  1.98438
1.33219 2.63656  1.58837
2.29289 0.40986 0.04574
0.55950 1.41291 1.59671
1.87241 2.07130 2.18328
2.18328 2.30251
```

Use your favorite plotting software



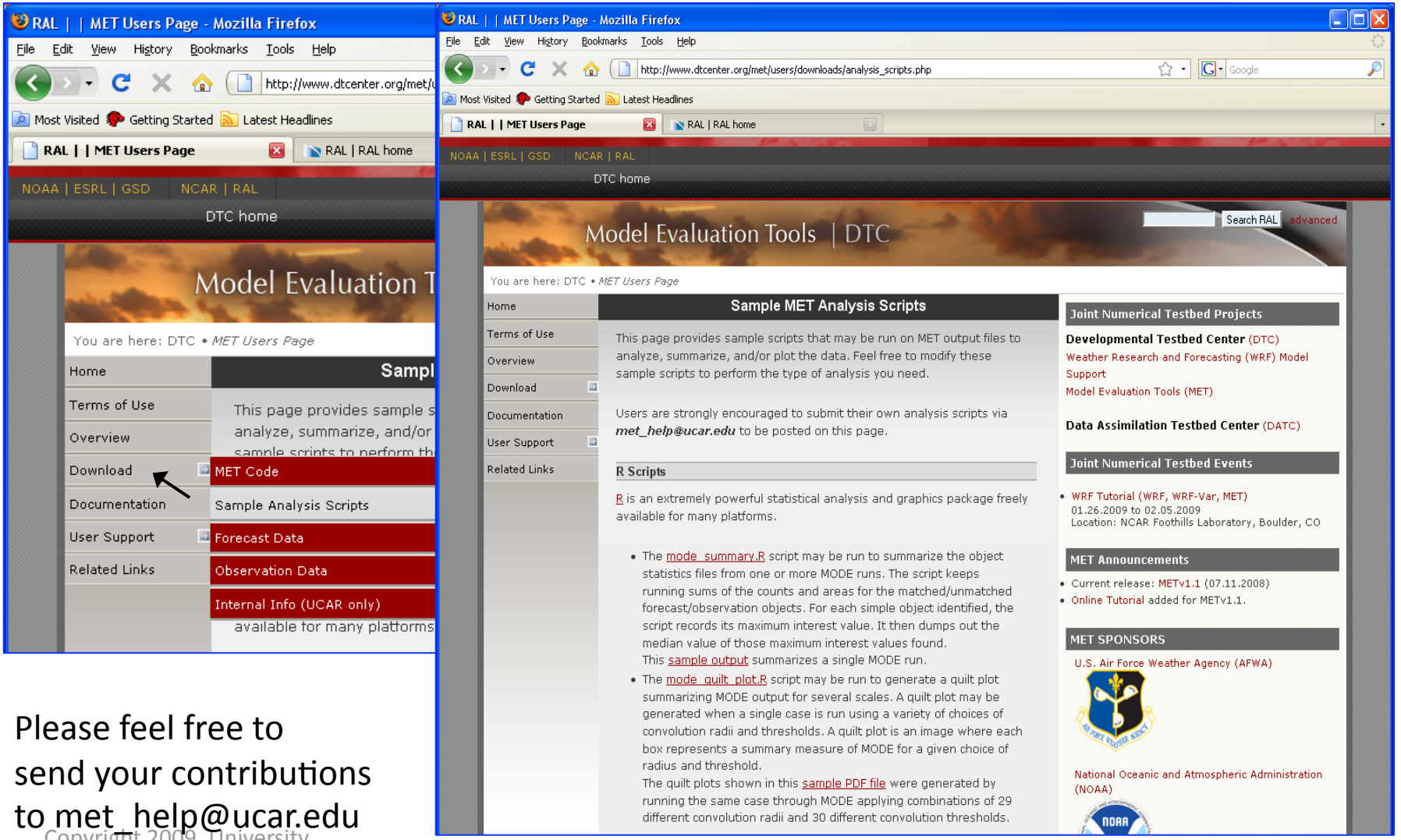






MET provides the analysis in ascii ouput.
You provide the graphing / plotting capability.

User Contributed Plotting Scripts

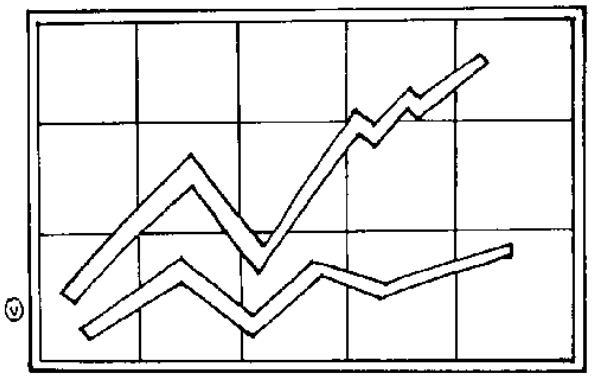


Please feel free to send your contributions to met_help@ucar.edu

Thanks - Any Questions?

Now you know how to:
Filter it.
Aggregate it.
and
Plot it...

Next up is:
Analyzing it.
and
Customizing it.



Interpretation of Output – Traditional Statistics

Point and Grid Stat Output Lines

- CTC - Contingency Table Counts
- CTS - Contingency Table Statistics
- CNT - Continuous Statistics
- FHO - Forecast, Hit, Observation Rates
- PCT - Contingency Table counts for Probabilistic forecasts
- PSTD – Contingency table Statistics for Probabilistic forecasts with Dichotomous outcomes
- PJC - Joint and Conditional factorization for Probabilistic forecasts
- PRC - Receiver Operating Characteristic for Probabilistic forecasts
- SL1L2 - Scalar L1L2 Partial Sums
- SAL1L2 - Scalar Anomaly L1L2 Partial Sums when climatological data is supplied
- VL1L2 - Vector L1L2 Partial Sums
- VAL1L2 - Vector Anomaly L1L2 Partial Sums when climatological data is supplied
- MPR - Matched Pair data

Point and Grid stat output example

Header Line:

TOTAL BASER BASER_NCL BASER_NCU BASER_BCL BASER_BCU FMEAN FMEAN_NCL FMEAN_NCU FMEAN_BCL FMEAN_BCU ACC ACC_NCL ACC_NCU ACC_BCL ACC_BCU FBIAS FBIAS_BCL FBIAS_BCU PODY PODY_NCL PODY_NCU PODY_BCL PODY_BCU PODN PODN_NCL PODN_NCU PODN_BCL PODN_BCU POFD POFD_NCL POFD_NCU POFD_BCL POFD_BCU FAR FAR_NCL FAR_NCU FAR_BCL FAR_BCU CSI CSI_NCL CSI_NCU CSI_BCL CSI_BCU GSS GSS_BCL GSS_BCU HK HK_NCL HK_NCU HK_BCL HK_BCU HSS HSS_BCL HSS_BCU ODDS ODDS_NCL ODDS_NCU ODDS_BCL ODDS_BCU

Data Line:

CTS 5 1.00000 0.56552 1.00000 NA NA 0.40000 0.11762 0.76928 NA NA 0.40000 0.11762 0.76928 NA NA 0.40000 NA NA 0.40000 0.11762 0.76928 NA NA NA NA NA NA NA NA NA NA NA NA 0.00000 0.00000 0.43448 NA NA 0.40000 0.11762 0.76928 NA NA 0.00000 NA NA NA NA NA NA NA 0.00000 NA NA NA NA NA NA NA

Example Contingency Table Statistics

See Appendix C of MET documentation for equations and details

- TOTAL – Count of total pairs included in the stats for this line
- BASER – base rate (e.g. sample climatological rate of event)
- ACC - accuracy
- FBIAS – frequency bias
- PODY – probability of detection (events)
- PODN – probability of detection (non-events)
- POFD – probability of false detection
- FAR – False Alarm Ratio
- CSI – Critical success index
- GSS – Gilbert Skill Score
- HK - Hanssen-Kuipers Discriminant
- HSS - Heidke Skill Score
- ODDS – Odds Ratio
- Etc., etc., etc. . .

Confidence Intervals for some measures are included with the following suffixes:
_NCL = Normal Confidence Lower
_NCU = Normal Confidence Upper
_BCL = Bootstrap Confidence Lower
_BCU = Bootstrap Confidence Upper

Example Continuous Statistics

See Appendix C of MET documentation for equations and details

Confidence Intervals for some measures are included with the following suffixes:
_NCL = Normal Confidence Lower
_NCU = Normal Confidence Upper
_BCL = Bootstrap Confidence Lower
_BCU = Bootstrap Confidence Upper

- FBAR – Forecast average
- OBAR– Observation average
- ME – Mean Error
- ESTDEV– Estimated standard deviation of the error
- MAE – Mean absolute error
- MSE – Mean squared error
- BCMSE – Bias corrected mean squared error
- RMSE – Root mean squared error
- E50 – Median (50% percentile) of the error.
- Etc., etc., etc. . .

Example Probability Forecast Statistics

See Appendix C of MET documentation for equations and details

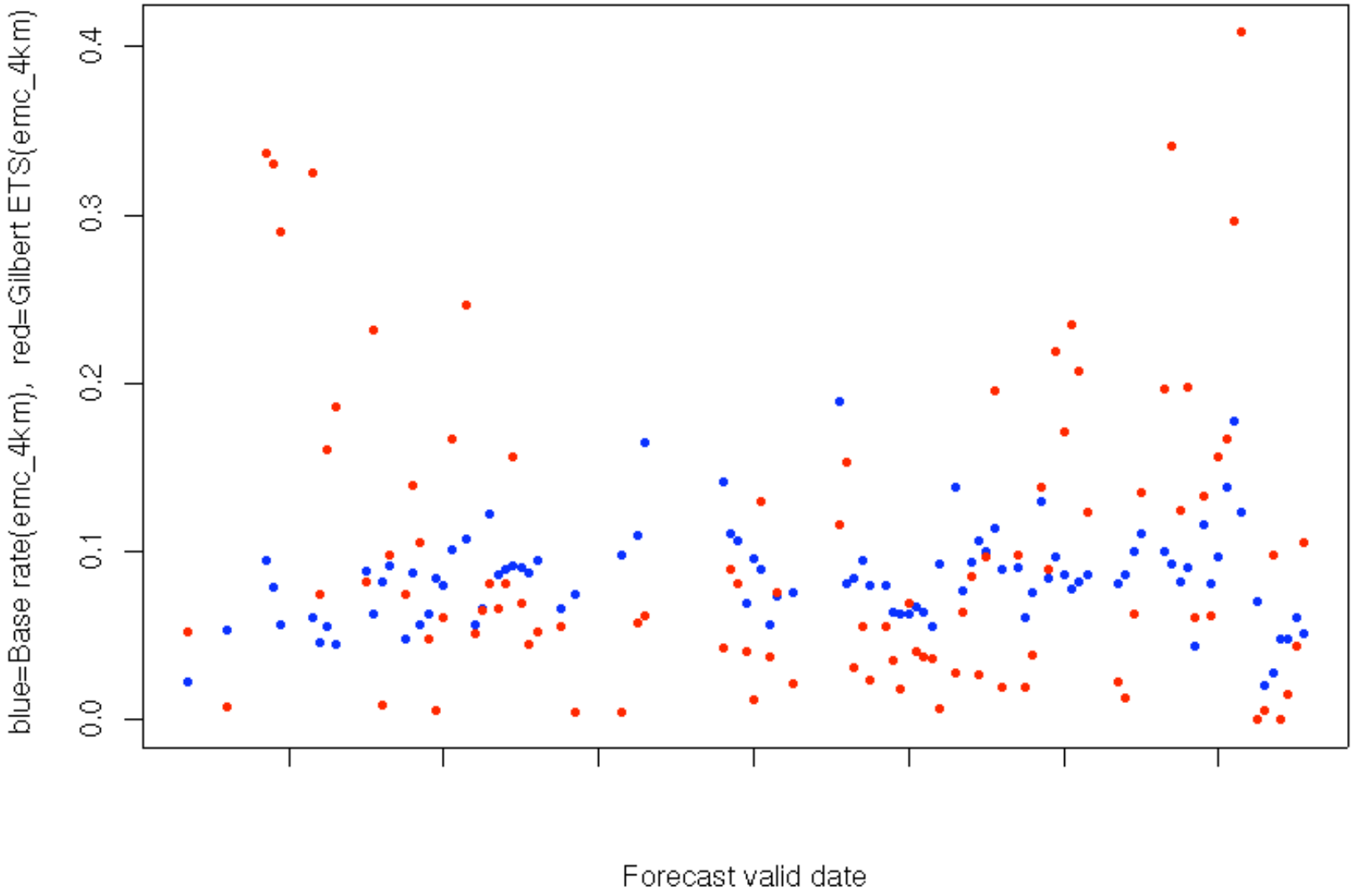
- RELIABILITY
- RESOLUTION
- UNCERTAINTY
- ROC_AUC - Area under the receiver operating characteristic curve
- BRIER – Brier Score
- CALIBRATION_i - conditional probability of an event given each probability forecast
- REFINEMENT_i – probability of each forecast category
- LIKELIHOOD_i - conditional probability for each forecast category given the event
- BASER_i - probability of an event for each forecast category

Example of Appendix C info

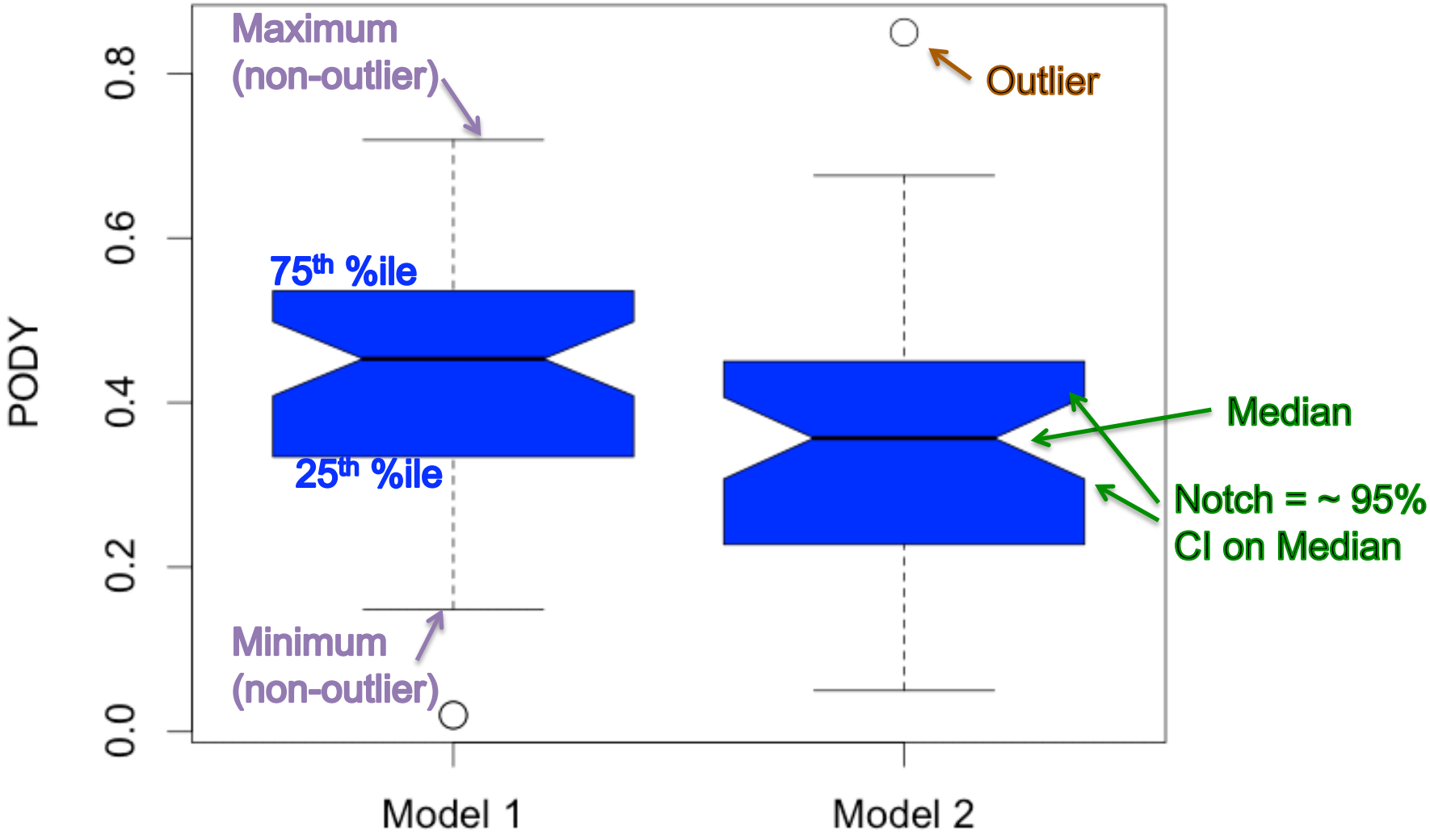
- **Accuracy -** for a 2x2 contingency table is defined as $\frac{n_{11} + n_{00}}{T}$

- That is, it is the proportion of forecasts that were either hits or correct rejections – the fraction that were correct.
- Accuracy ranges from 0 to 1; a perfect forecast would have an accuracy value of 1.
- Accuracy should be used with caution, especially for rare events, because it can be strongly influenced by large values of n_{00}.

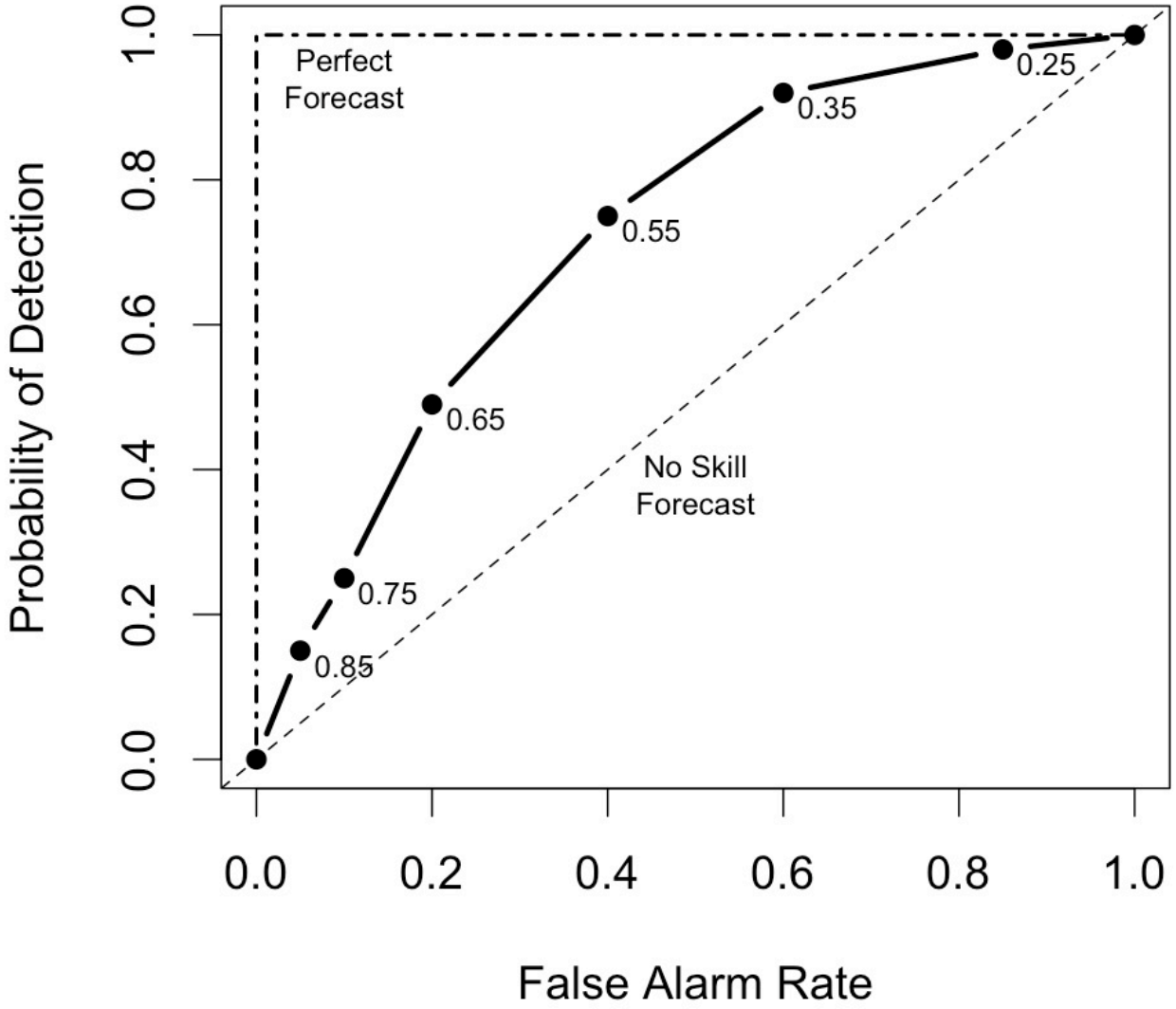
Example Scatter Plot



Example Box (and Whisker) Plot

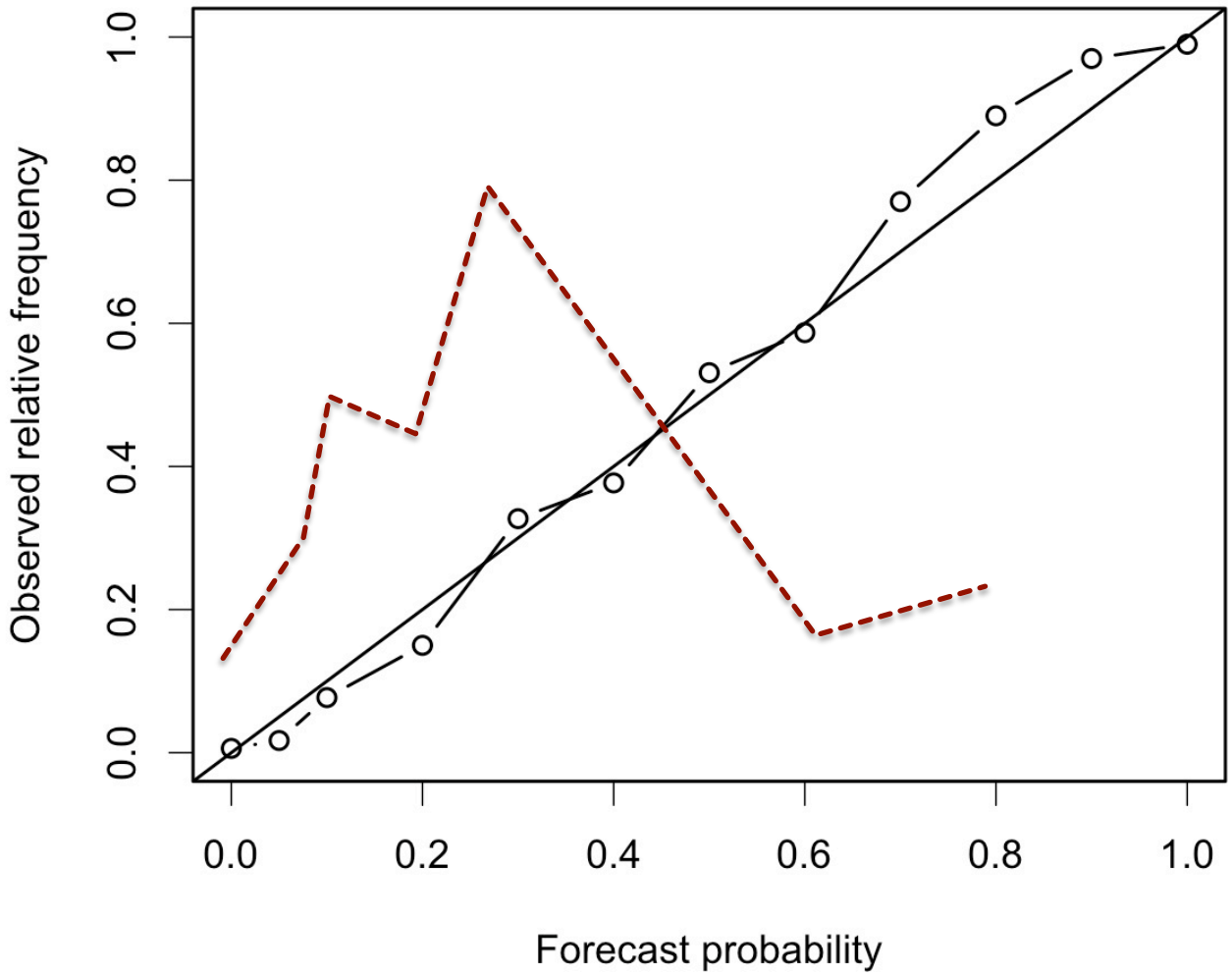


Example Receiver Operating Characteristic Plot



Create with points from PRC line type.

Example Reliability Diagram



A measure of conditional bias.

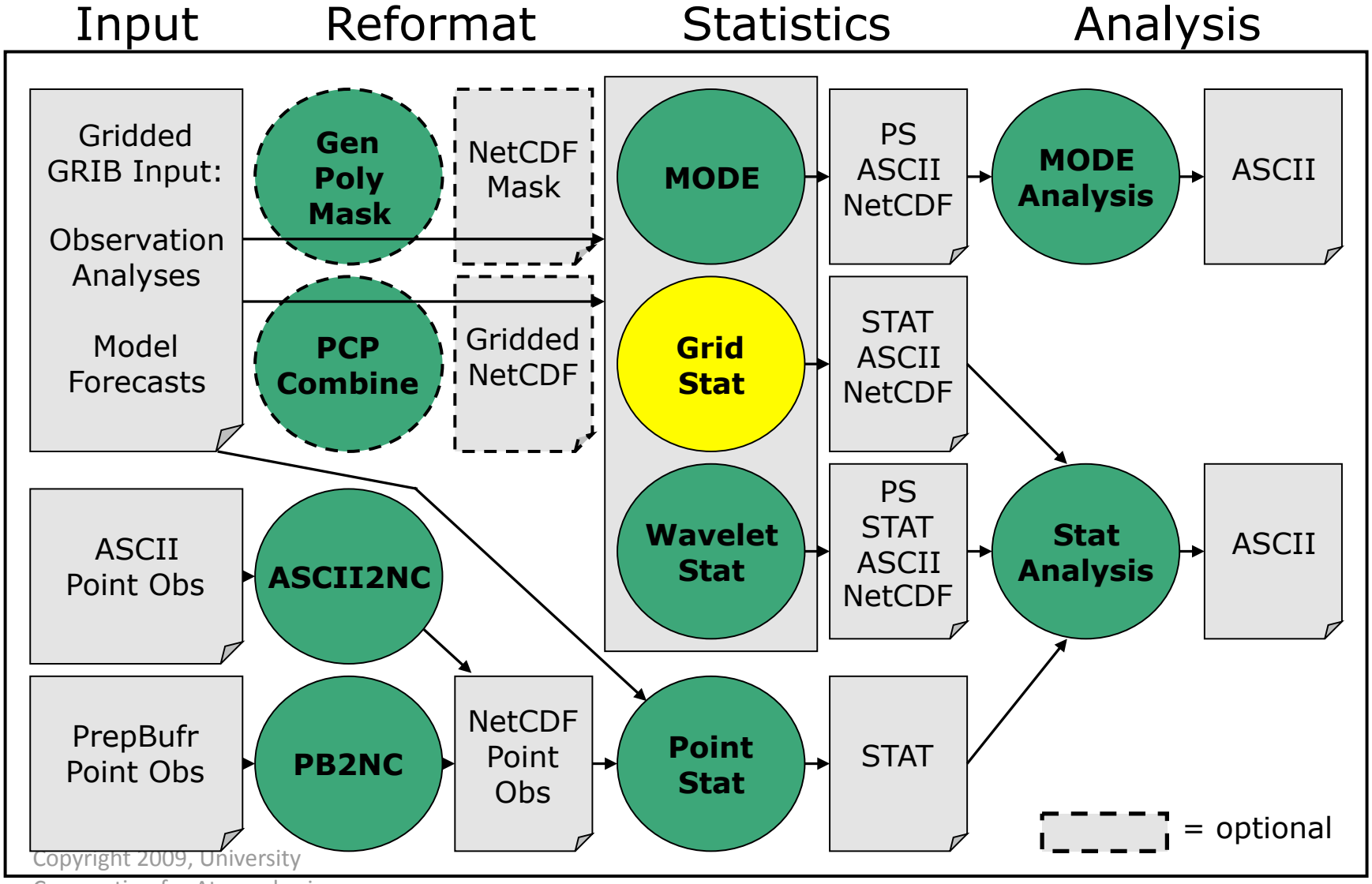
Do the forecast and observed probabilities match?

One to one line (solid) shows perfect reliability.

Forecast 1 (the line with dots) has good reliability.

Forecast 2 (the dashed line) has very poor reliability.

Grid-Stat Tool



Grid-Stat: Configuration

- 27 configurable parameters – only set a few:
  - Precipitation accumulated over 24 hours.
    - **fcst_field[] = [“APCP/A24“];**
  - Any rain and moderate rain.
    - **fcst_thresh[] = [“gt0.00, gt20.00“]; (mm)**
  - Look at all the points and only the eastern United States.
    - **mask_grid[] = ["FULL"];**
    - **mask_poly[] = [“EAST.poly"];**
  - Compute neighborhood statistics with two sizes.
    - **nbr_width[] = [3, 5];**
  - Generate all possible statistic types, except probabilistic.
    - **output_flag[] = [2, 2, 2, 2, 2, 2, 0, 0, 0, 0, 2, 2, 2, 1]**

Grid-Stat: Run

- METv2.0/bin/grid_stat \
  sample_fcst.grb sample_obs.nc \
  GridStatConfig_APCP24 -outdir out -v 2

```
Forecast File: sample_fcst.grb
Observation File: sample_obs.nc
Configuration File: GridStatConfig_APCP24
----------------------------------------
Processing APCP/A24 versus APCP/A24, for interpolation method UW_MEAN(1), over region FULL, using 6412 pairs.
Computing Categorical Statistics.
Computing Continuous Statistics.
Processing APCP/A24 versus APCP/A24, for interpolation method UW_MEAN(1), over region EAST, using 2586 pairs.
Computing Categorical Statistics.
Computing Continuous Statistics.
Processing APCP/A24 versus APCP/A24, for interpolation method NBRHD(9), raw thresholds of >0.000 and >0.000,
over region EAST, using 5829 pairs.
... MORE NEIGHBORHOOD COMPUTATIONS ...
----------------------------------------
Output file: out/grid_stat_240000L_20050808_000000V.stat
Output file: out/grid_stat_240000L_20050808_000000V_fho.txt
Output file: out/grid_stat_240000L_20050808_000000V_ctc.txt
Output file: out/grid_stat_240000L_20050808_000000V_cts.txt
Output file: out/grid_stat_240000L_20050808_000000V_cnt.txt
Output file: out/grid_stat_240000L_20050808_000000V_sl1l2.txt
Output file: out/grid_stat_240000L_20050808_000000V_nbrctc.txt
Output file: out/grid_stat_240000L_20050808_000000V_nbrcts.txt
Output file: out/grid_stat_240000L_20050808_000000V_nbrcnt.txt
Output file: out/grid_stat_240000L_20050808_000000V_pairs.nc
```

Grid-Stat: ASCII Output Types

- Statistics line types: 8 possible
  - Same as Point-Stat
    - FHO, CTC, CTS, CNT, and SL1L2
    - PCT, PSTD, PJC, and PRC
  - Omitted for Grid-Stat
    - *SAL1L2*, *VL1L2*, or *VAL1L2*
  - Neighborhood – define neighborhood, apply threshold
    - Neighborhood continuous statistics (NBRCNT)
    - Neighborhood contingency table counts (NBRCTC)
    - Neighborhood contingency table statistics (NBRCTS)
- 21 header columns common to all line types
- Remaining columns specific to each line type

Grid-Stat: Sample Output

- STAT file output for sample run:
  - 2 lines each for CNT and SL1L2
    - = 2 verification regions (FULL and EAST)
  - 4 lines each for FHO, CTC, and CTS
    - = 2 regions * 2 thresholds
  - 8 lines each for NBRCNT, NBRCTC, NBRCTS
    - = 2 regions * 2 thresholds * 2 neighborhood sizes
- Additional TXT files for each line type
- NetCDF file containing matched pairs

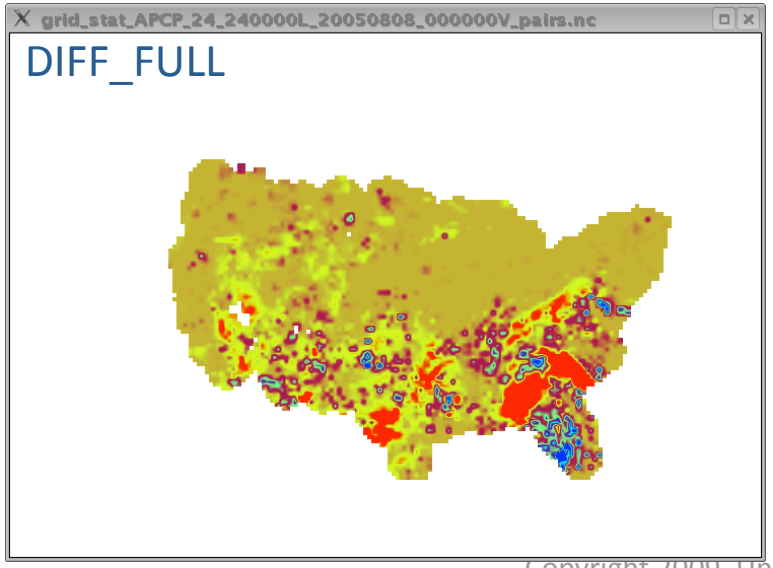
Grid-Stat: CTC Output Line

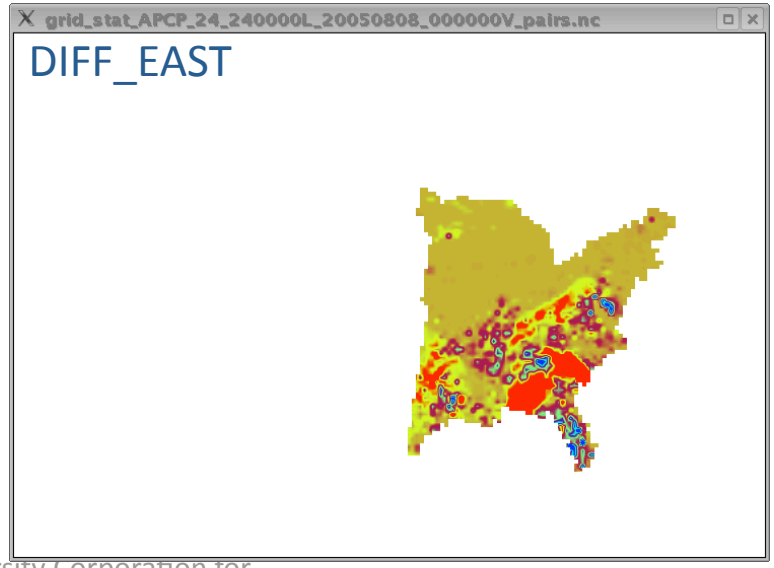
VERSION	V2.0
MODEL	WRF
FCST_LEAD	240000
FCST_VALID_BEG	20050808_000000
FCST_VALID_END	20050808_000000
OBS_LEAD	000000
OBS_VALID_BEG	20050808_000000
OBS_VALID_END	20050808_000000
FCST_VAR	APCP
FCST_LEV	A24
OBS_VAR	APCP
OBS_LEV	A24
OBTYPE	MC_PCP

VX_MASK	EAST
INTERP_MTHD	UW_MEAN
INTERP_PNTS	1
FCST_THRESH	>=20.000
OBS_THRESH	>=20.000
COV_THRESH	NA
ALPHA	NA
LINE_TYPE	CTC
TOTAL	2586
FY_OY	5
FY_ON	104
FN_OY	70
FN_ON	2407

Grid-Stat: NetCDF Matched Pairs

- Forecast, observation, and difference fields for each combination of…
  - Variable, level, masking region, and interpolation method (smoothing)
- Sample output contains 6 variables:
  - FCST, OBS, and DIFF for FULL and EAST





Customizing Point-Stat and Grid-Stat Output

We'll restrict this discussion to an explanation of masking and interpolation.

What does masking do?

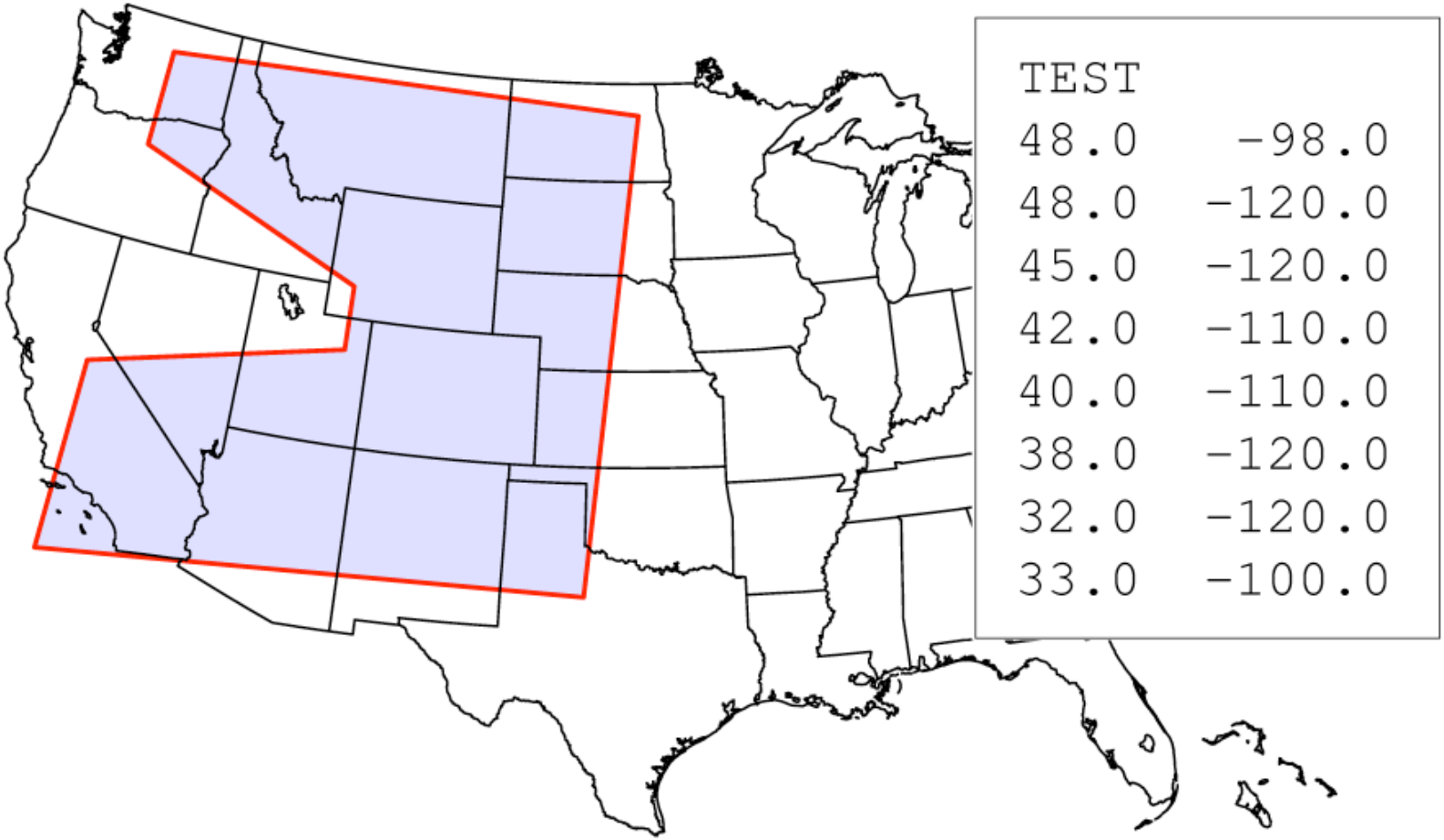
- Defines a region for verification.
- Eliminates all other areas from consideration.

Masking Methods

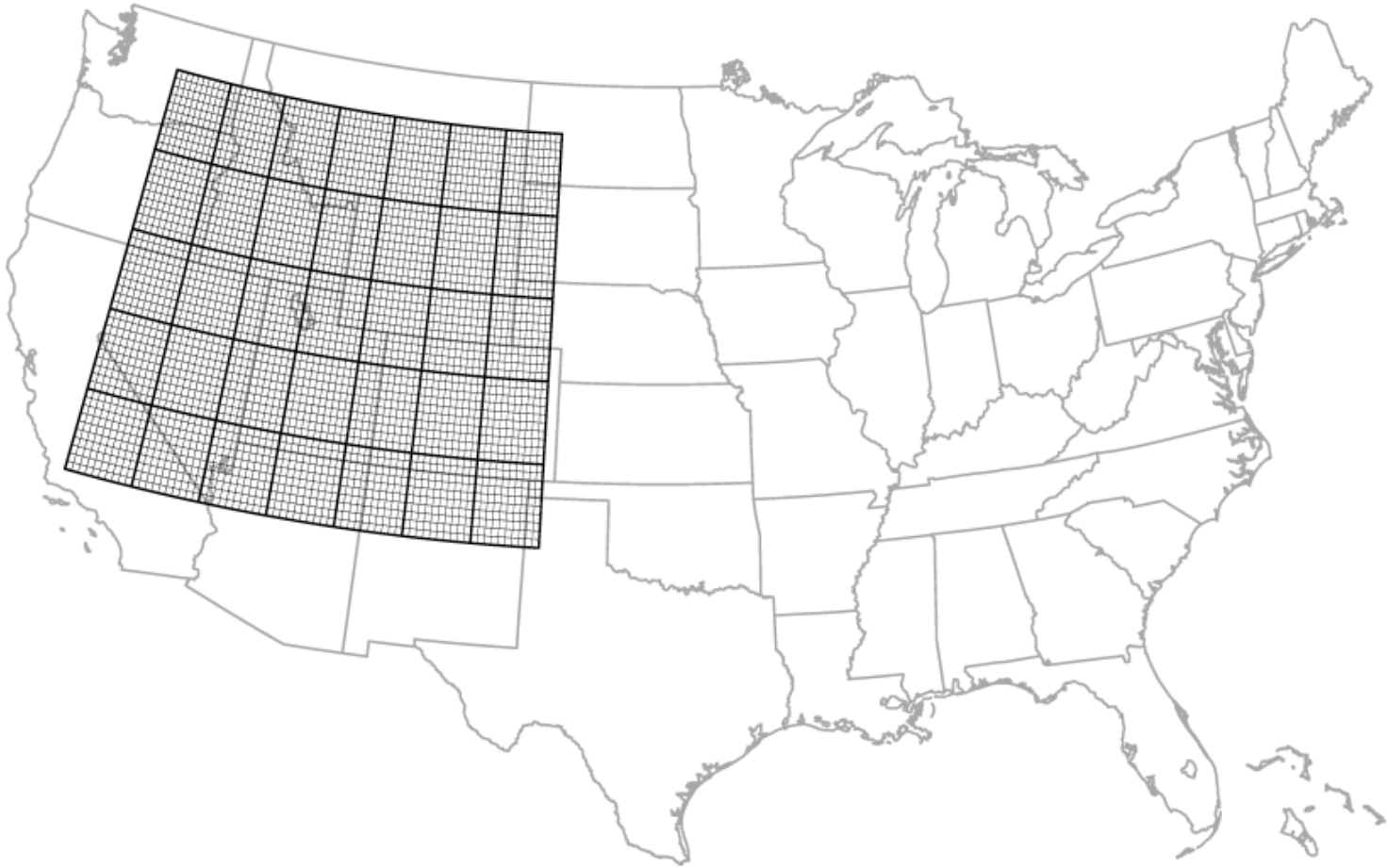
	Polyline	Grid	Stations	On/Off Bit Map
Point Stat	✓	✓	✓	✓
Grid Stat	✓	✓	N/A	✓

Masking by stations doesn't apply to Grid Stat.

Polyline Masking

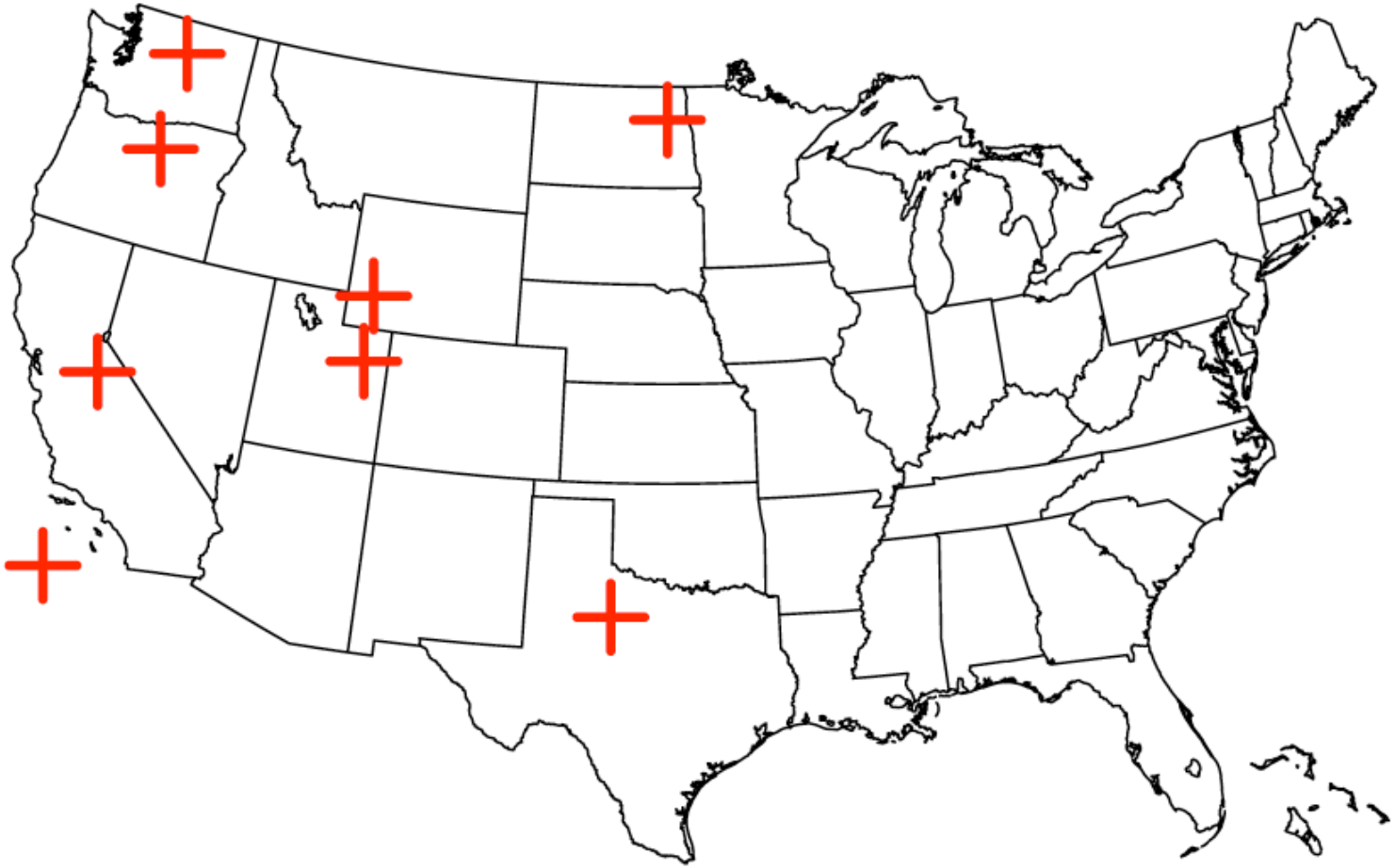


Grid Masking

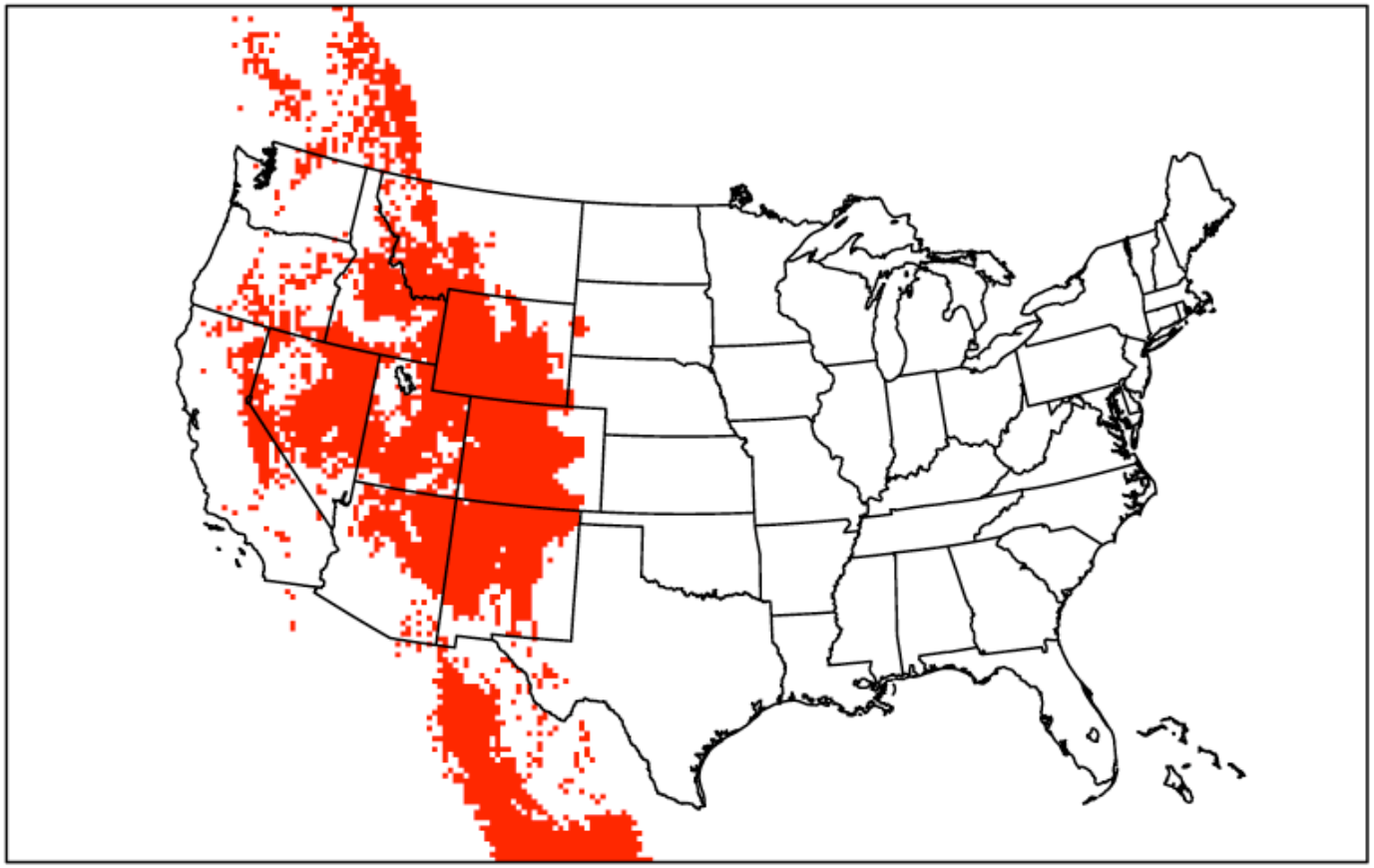


NCEP grids link: www.nco.ncep.noaa.gov/pmb/docs/on388/tableb.html

Station Masking



Data Threshold Mask



Topography > 5000 feet

What does interpolating do?

- Matches gridded forecast data to point observations.
- Smoothes gridded information.

Interpolation Methods

	Min	Max	Median	UW Mean	DW Mean	Nearest Nbr	Least Squares
Point Stat	✓	✓	✓	✓	✓	✓	✓
Grid Stat	✓	✓	✓	✓	N/A	N/A	N/A

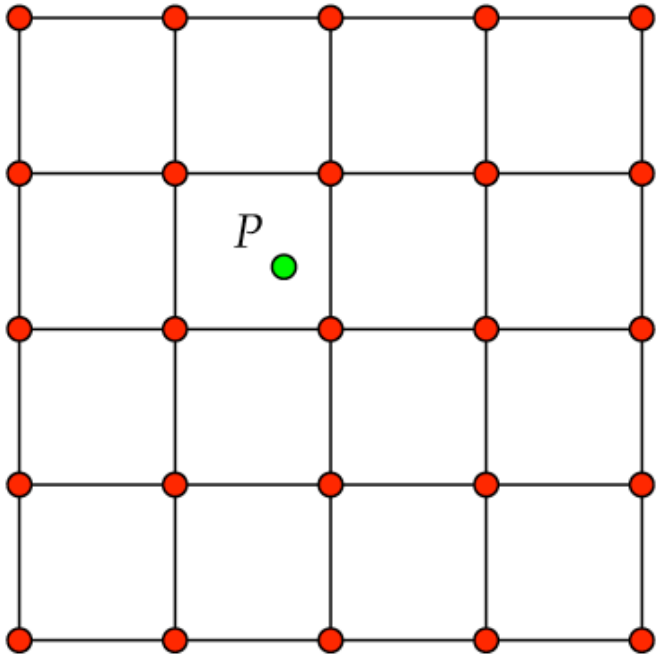
For Grid Stat, these are smoothing methods.

Interpolation

Need to Choose:

(1) Method

(2) Width



Min, Max, Median

Takes minimum, maximum or median of values in interpolation square.

Median separates the upper half of data values from the lower half. This is different from the mean, which is an average.

Nearest Neighbor

Essentially, no interpolation is performed.

Value at interpolation point is simply the data value at the closest grid point.

Unweighted Mean
Distance-Weighted Mean

Unweighted Mean is the average.

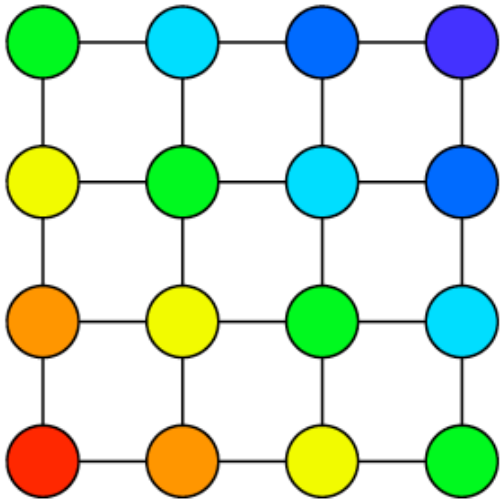
Distance-Weighted Mean is an average weighted according to distance from nearby grid points.

Least Squares

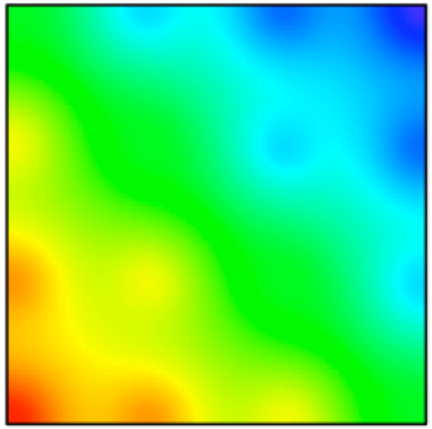
Performs a local Least-Squares linear fit in interpolation square.

$$z = Ax + By + C$$

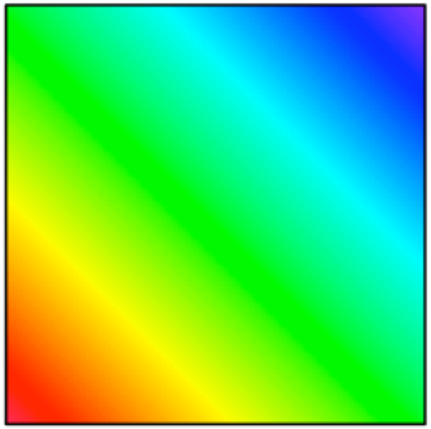
Interpolation Examples



Nearest Neighbor



Distance Weighted Mean



Least Squares

Least Squares

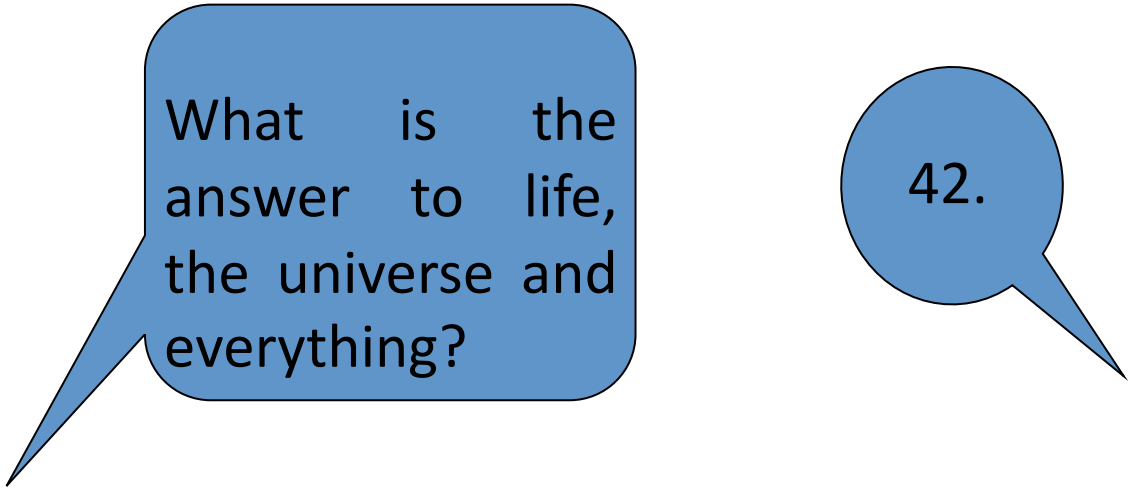
Performs a local Least-Squares linear fit in interpolation square.

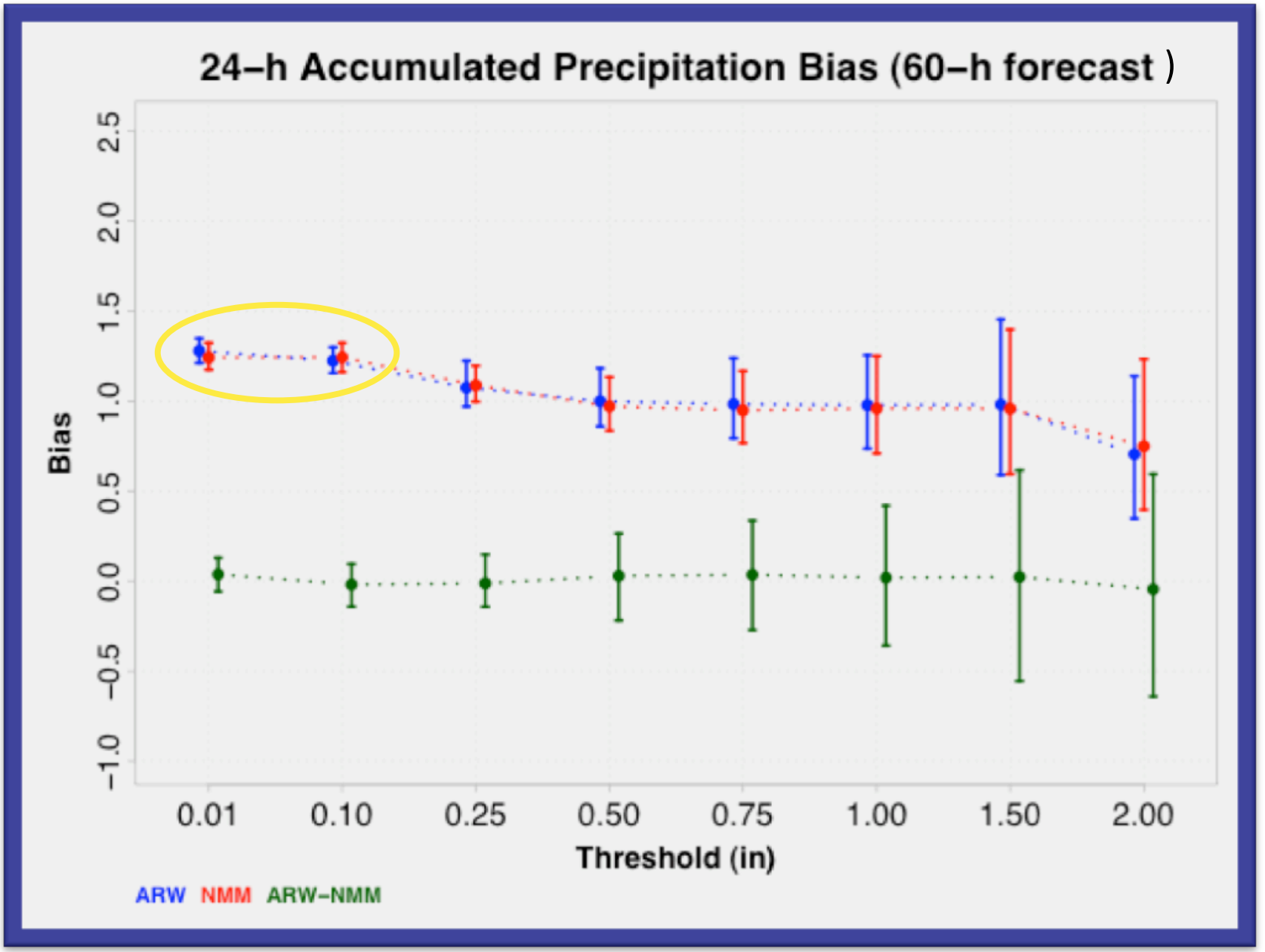
$$z = Ax + By + C$$

- MET is highly configurable.
- Masking and interpolating are two ways to customize MET to your verification needs.

- There are many more, see documentation for details . . .

Giving meaning to your forecast verification results with Confidence Intervals





Accounting for Uncertainty

- Observational
- Model
  - Model parameters
  - Physics
  - Verification scores
- Sampling
  - Verification statistic is a realization of a random process.
  - What if the experiment were re-run under identical conditions?

Hypothesis Testing and Confidence Intervals

- Hypothesis testing
  - Given a null hypothesis (e.g., no bias), is there enough evidence to reject it?
  - One- or two-sided, but test is against a single null hypothesis.
- Confidence intervals
  - Related to hypothesis tests, but more useful.
  - How confident are we that the true value of the statistic (e.g., bias) is different from a particular value?
  - Interpretation for most *frequentist* intervals is a bit awkward.

Hypothesis Testing and Confidence Intervals

"If we re-run the experiment 100 times, and create 100 (1-α)100% CI's, then we expect the true value of the parameter to fall inside (1-α)100 of the intervals."

Hypothesis Testing and Confidence Intervals

Example: The difference in bias between two models is 0.01.

Hypothesis test: Is this different from zero?

Confidence interval: Does zero fall within the interval? Does 0.5 fall within the interval?

Confidence Intervals (CI's)

- Parametric
  - Assume the observed sample is a realization from a known *population* distribution with possibly unknown parameters (e.g., normal).
  - Normal approximation CI's are most common.
  - Quick and easy.

- Nonparametric
  - Assume the distribution of the observed sample is representative of the *population* distribution.
  - Bootstrap CI's are most common.
  - Can be computationally intensive, but easy enough.

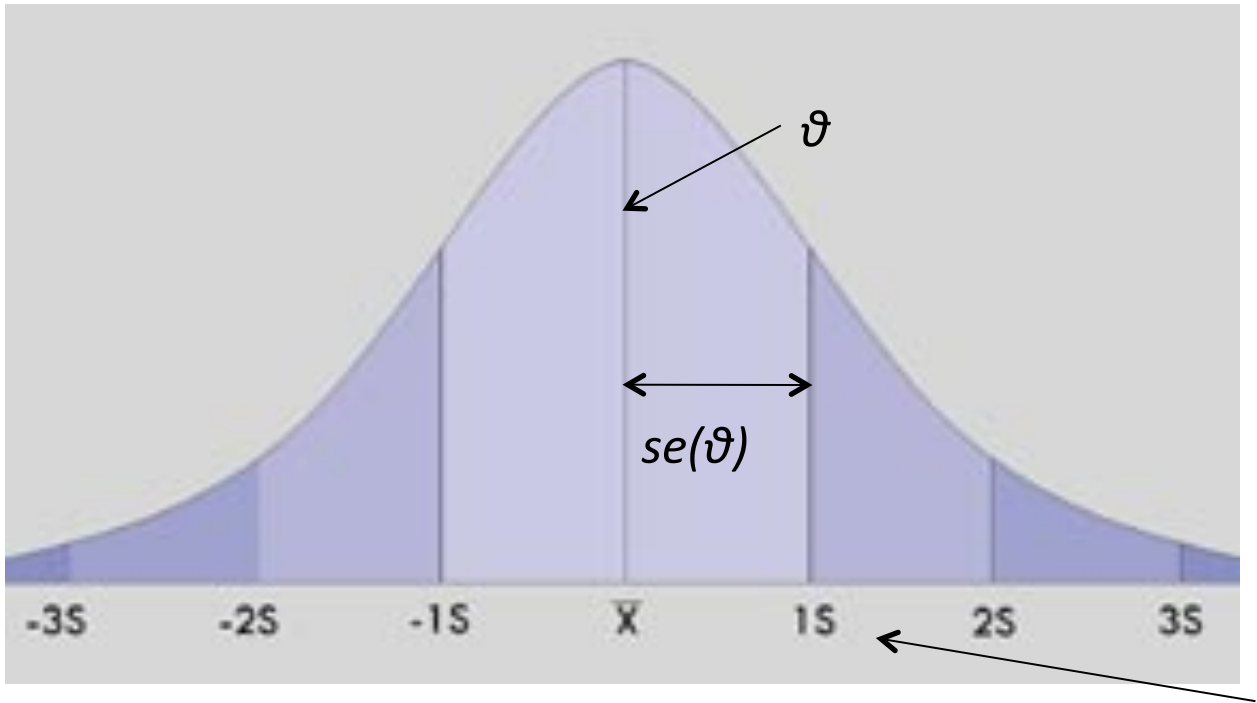
Normal Approximation CI's

$$\hat{\theta} \pm z_{\alpha/2} se(\theta)$$

Is a (1-α)100% Normal CI for Θ, where Θ is the statistic of interest (e.g., the forecast mean), se(Θ) is the standard error for the statistic, and z_v is the v–th quantile of the standard normal distribution.

Normal Approximation CI's

$$\hat{\theta} \pm z_{\alpha/2} se(\theta)$$



Normal Approximation CI's

Example: Let $X_1,\ldots,X_n$ be independent and identically distributed (iid) sample from a normal distribution with variance σ_X^2 .

Then, $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ is an estimate of the mean

of the sample. And a (1-α)100% CI is given by

$$\bar{X} \pm z_{\alpha/2}\frac{\sigma_X}{\sqrt{n}}$$

Normal Approximation CI's

- Numerous verification statistics can take this approximation in some form or another. In other cases (e.g., forecast/observation variance, linear correlation), different parametric CI's can be used that still rely on the underlying sample's being iid normal.

- Contingency table verification scores such as probability of detection (POD) and false alarm ratio (FAR) also have normal approximation CI's (for large enough sample sizes).

Normal Approximation CI's

- Check the validity of the independence assumption.
- Check the validity of the normal distribution (e.g., qq-plots, automatic tests exist too).

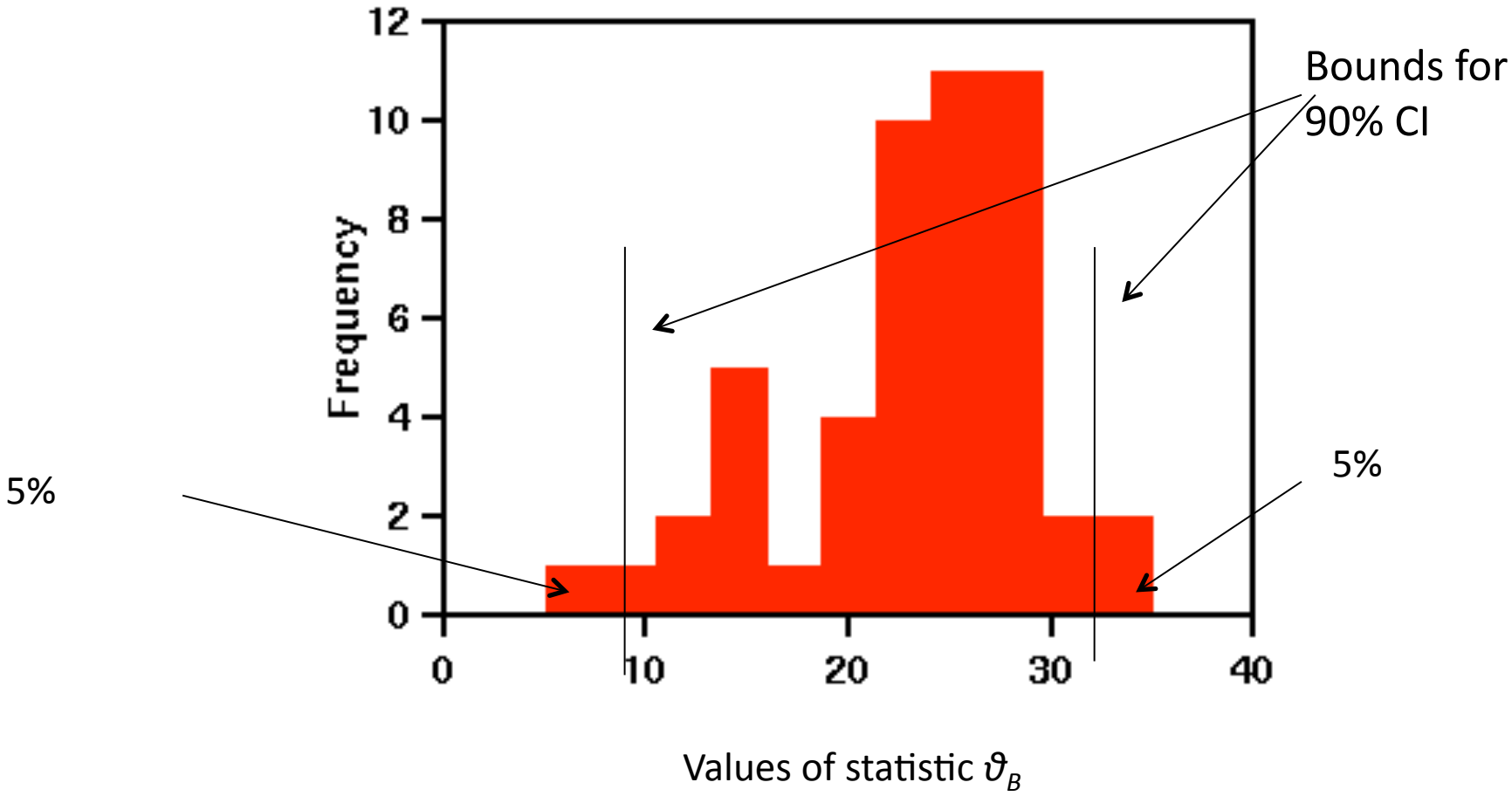
(cf. Gilleland, 2008)

Bootstrap CI's

IID Bootstrap Algorithm

1. Resample *with replacement* from the sample, $X_1,\ldots,X_n$
2. Calculate the verification statistic(s) of interest from the resample in step 1.
3. Repeat steps 1 and 2 many times, say B times, to obtain a sample of the verification statistic(s) ϑ_B .
4. Estimate (1-α)100% CI's from the sample in step 3.

Empirical Distribution (Histogram) of statistic calculated on repeated samples



Bootstrap CI's

IID Bootstrap Algorithm: Types of CI's

1. Percentile Method CI's*
2. Bias-corrected and adjusted (BCa)*
3. ABC
4. Basic bootstrap CI's
5. Normal approximation
6. Bootstrap-t

*1 and 2 are available In MET

Bootstrap CI's

Sample size is a configurable parameter for the bootstrap in MET.

Usually it is appropriate to use a sample the same size as the original sample.

However, there are cases where it is better to take smaller samples (e.g., if the population distribution is heavy-tailed; see Gilleland, 2008).

Practical Considerations

- Point-stat is quicker than Grid-stat, so bootstrap is quicker with Point-stat.
- May be prohibitively computationally inefficient to bootstrap over an entire field (i.e., over several thousand points), but can also bootstrap the statistics for each field over time. Measures the (between-field) uncertainty of the estimates over time, rather than the within field uncertainty.
- Normal approximation intervals are quick, and generally accurate. Check the normality assumption!
- Check whether the samples (for either type of interval) are independent or not.

Thank you. Questions?

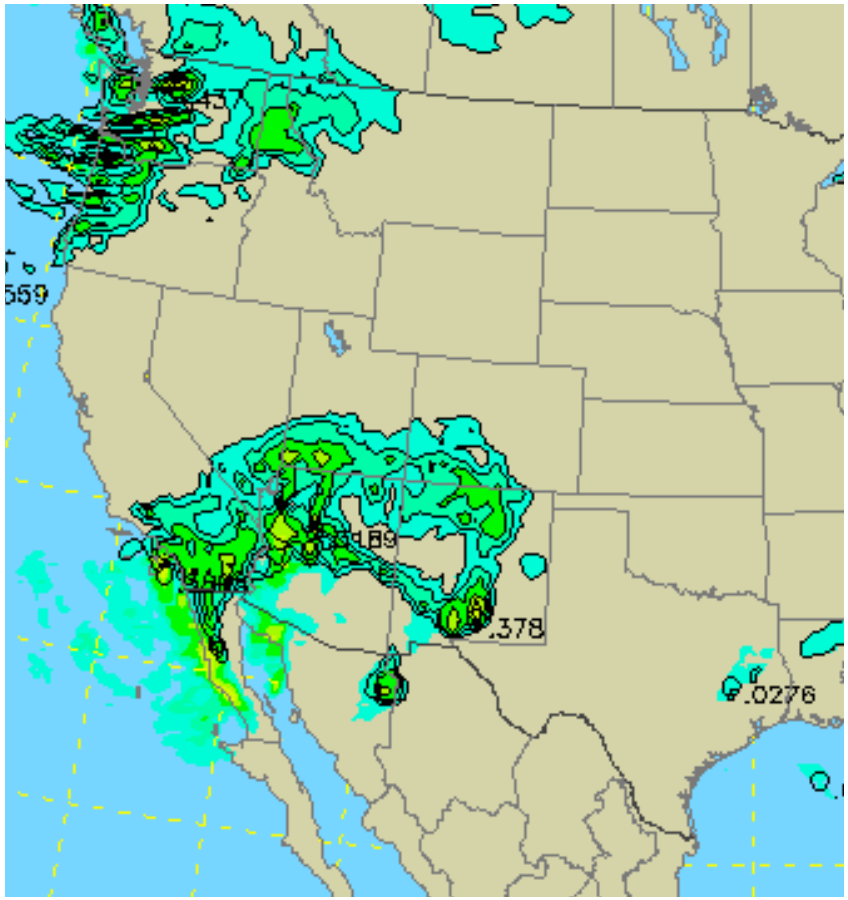
For more information, see:

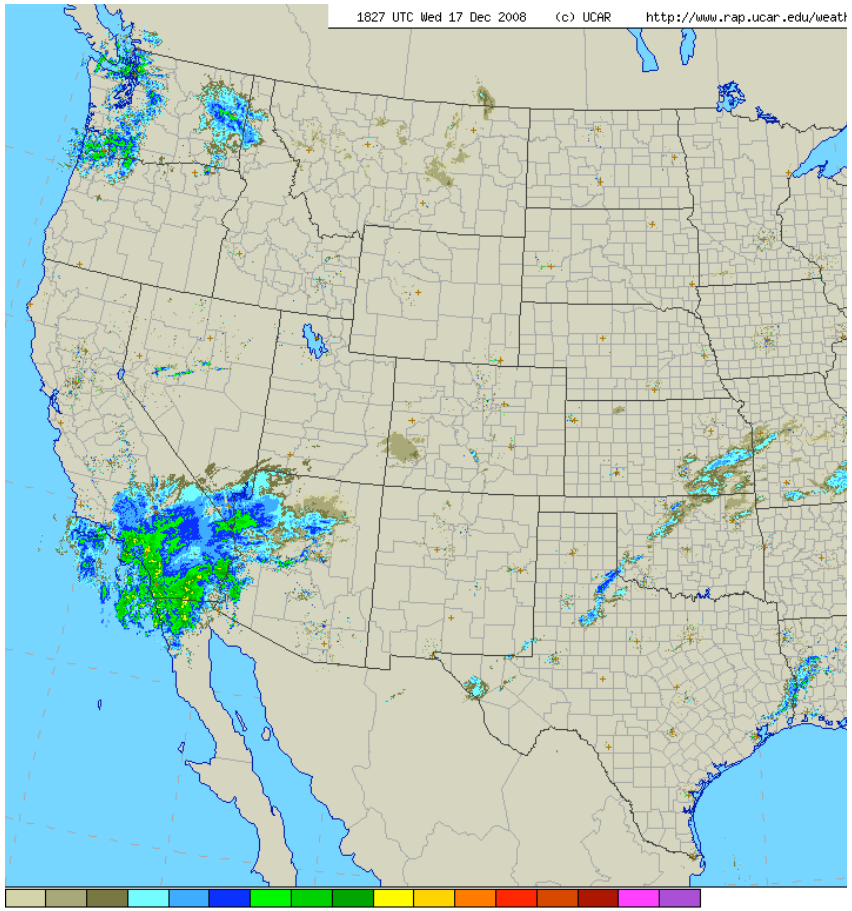
Developmental Testbed Center, 2009. Model Evaluation Tools User's Guide. Available at: http://www.dtcenter.org/met/

Gilleland E, 2008. Confidence intervals for forecast verification. *Submitted* as an NCAR Technical Note. Available at: http://www.ral.ucar.edu/~ericg/Gilleland2008.pdf

Why Use Objects?

Typical situation





Forecast

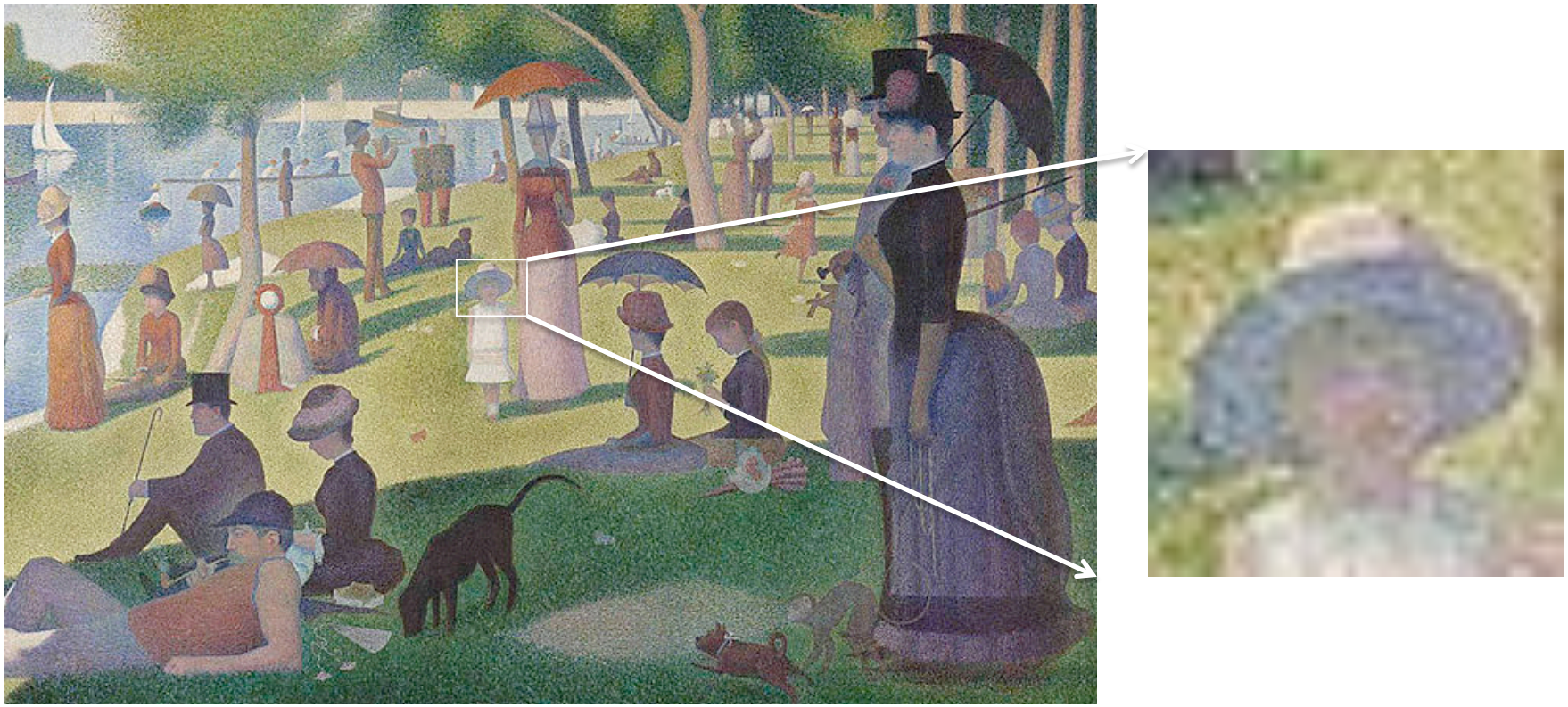
Observation

Traditional verification matches up points, then sums them up.

Many forecasts are more than the sums of their parts.

This method fails when forecasts are distorted, displaced, too weak, too strong, mis-sized.

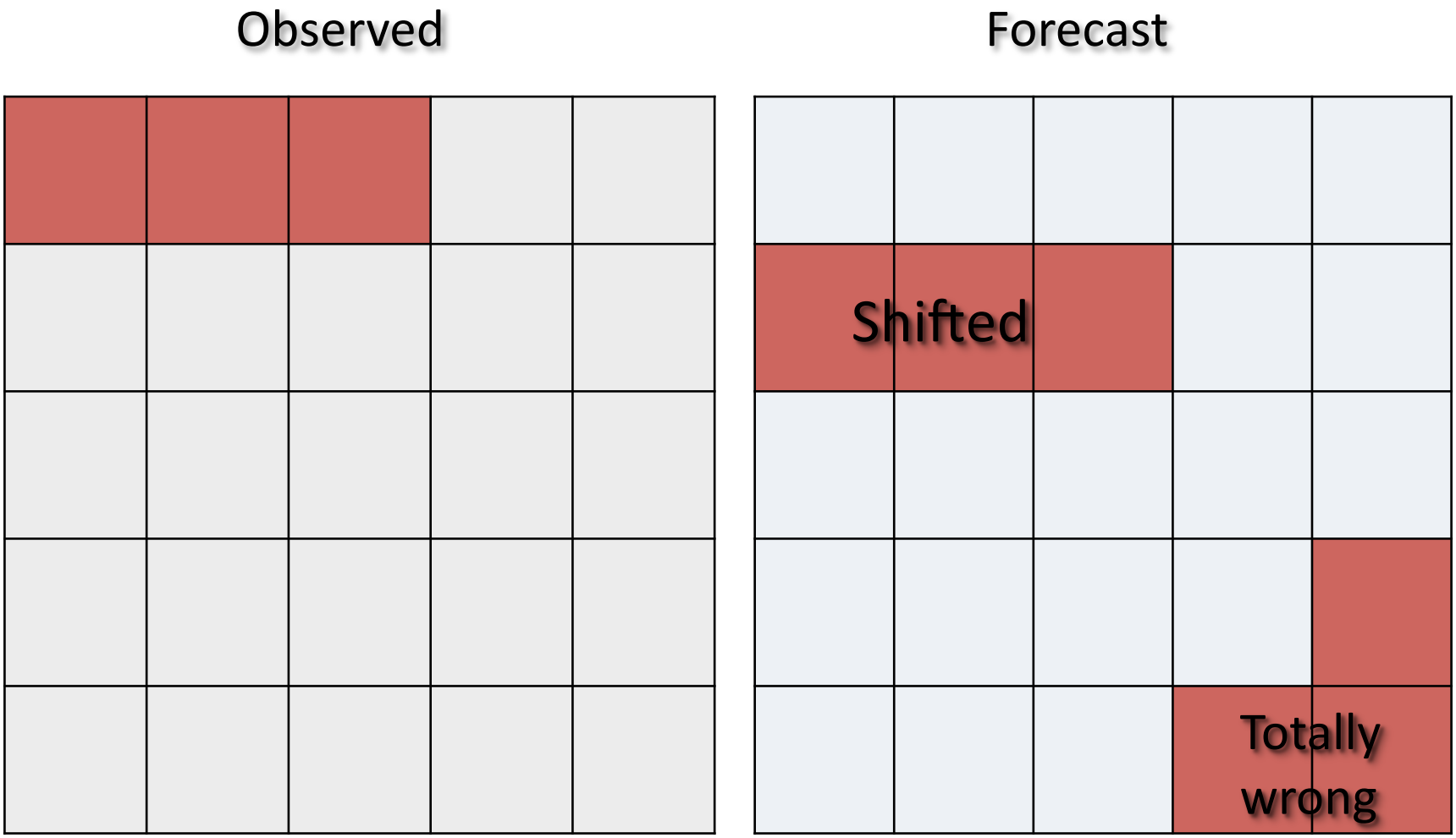
Pixels or Pictures?



Object verification
is more like what
humans
do.

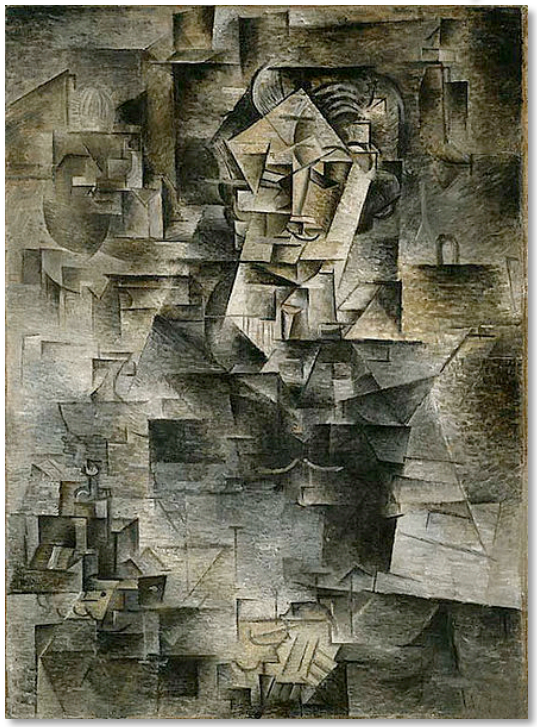
Objects recognize the relationship
between “things” even when they are
distorted, displaced, mis-sized, too strong, too
weak . . .

Simple example



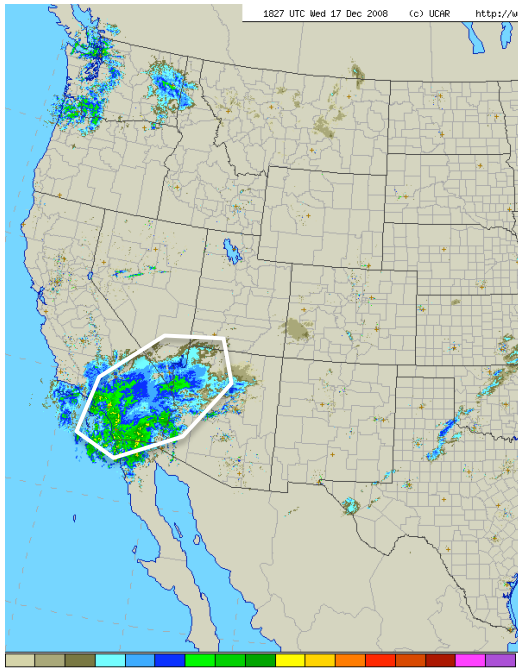
REAL - observed

Forecast 1 –
Distorted view of reality

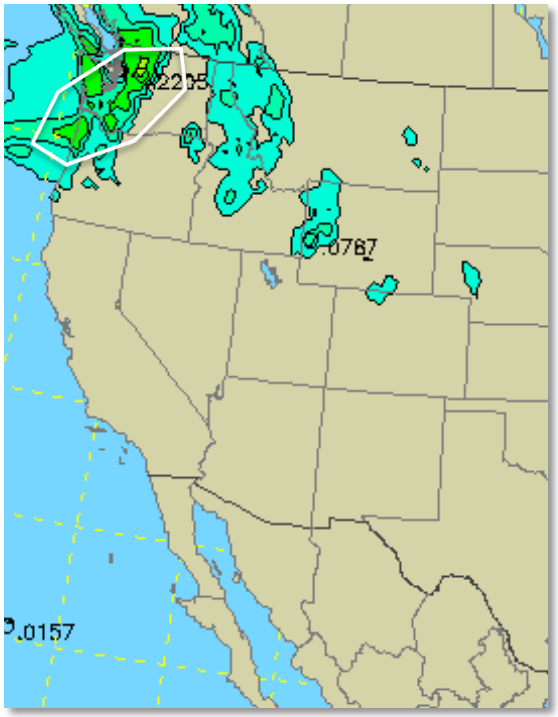


Forecast 2 –
Another distorted
view of reality

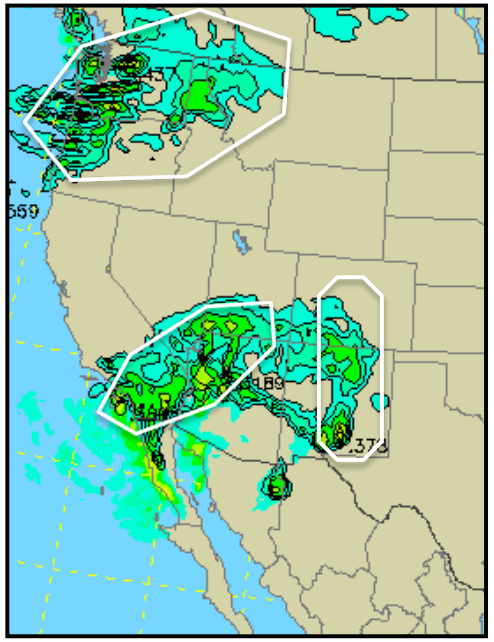
REAL - observed



Forecast 1 –
Distorted view of reality



Forecast 2 –
Another distorted
view of reality



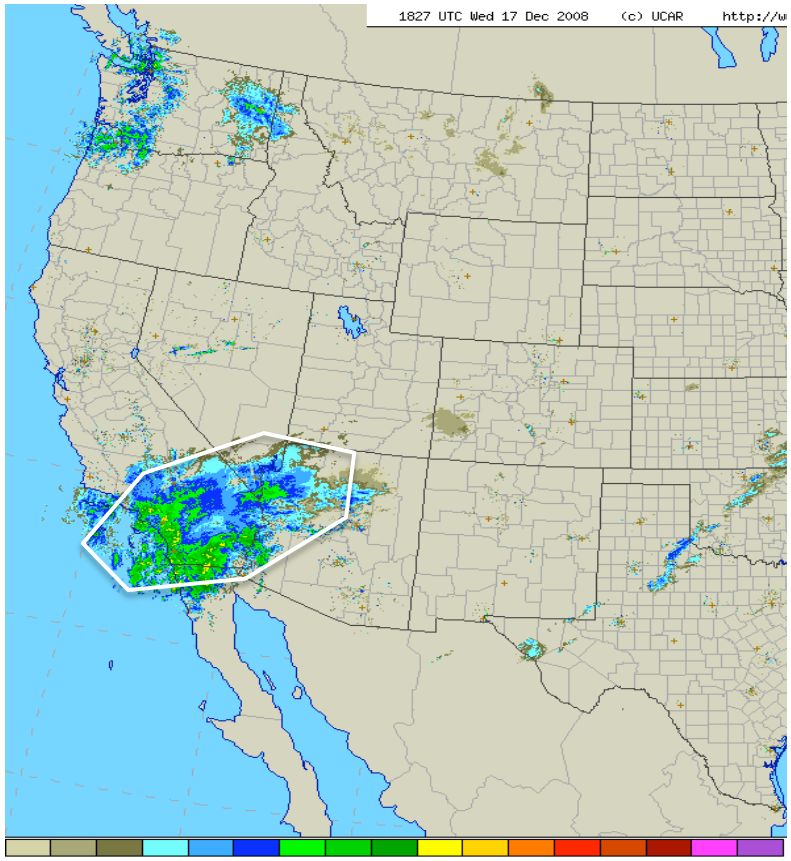
We can compare attributes of forecast and observations even when they are not in the same place!

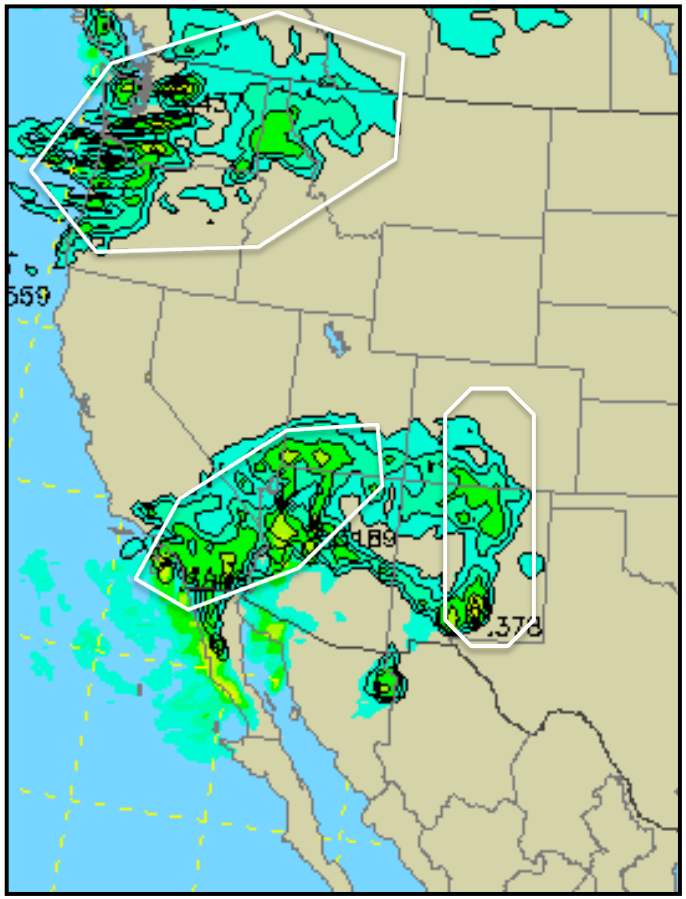
- Is the object in the right place?
- Does the size of these objects match?
- Is the intensity within the objects similar?

Centroid distance = 25

Area ratio = 85%

50 DBz vs 40 DBz





This is not really a new idea . . .

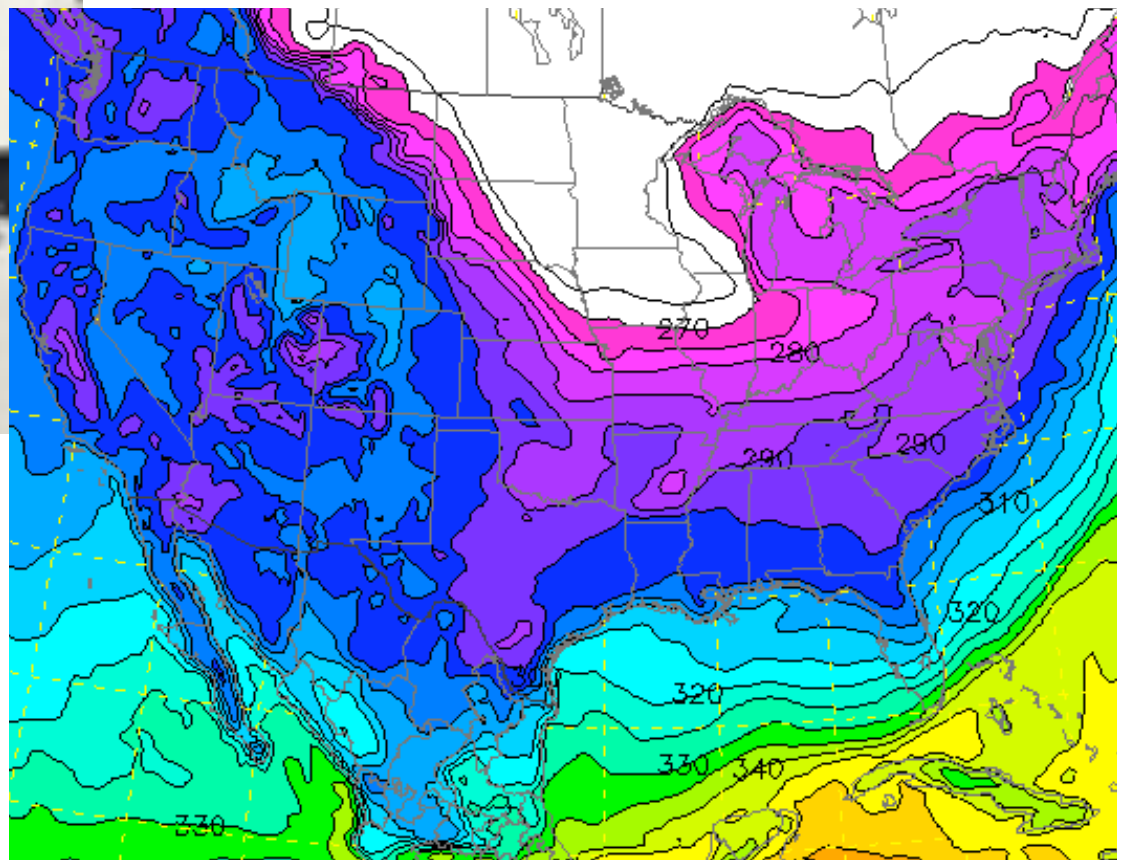
Analytic cubists "analyzed" natural forms and reduced the forms into basic geometric parts on the two-dimensional picture plane.

Analytic cubism was developed between 1908 and 1912 . . .

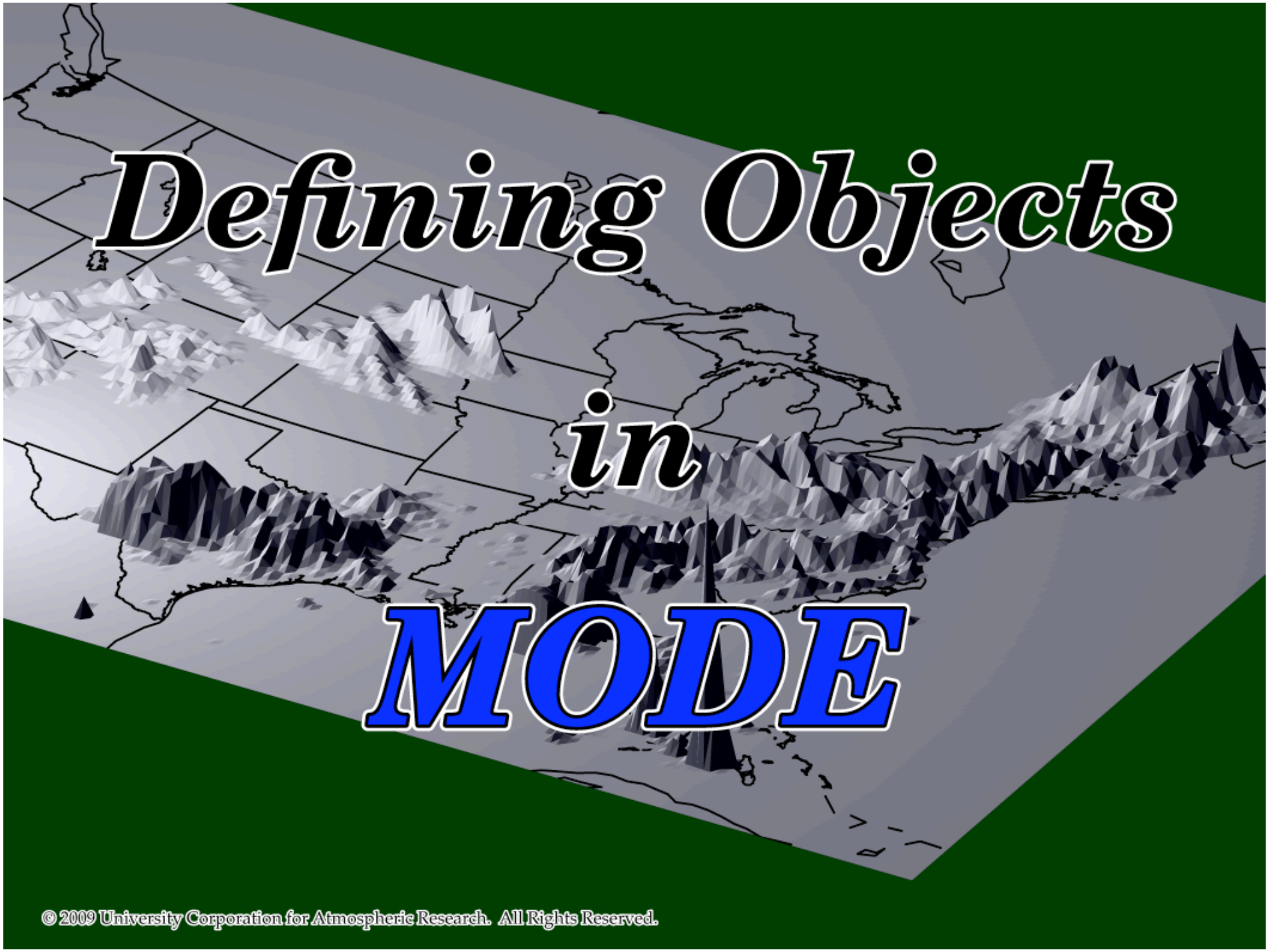
Comparing objects can tell you things about your forecast like . . .

This:	Instead of this:
30% Too Big	POD = 0.35
Shifted west 1 km	FAR = 0.7235
Rotated 15°	CSI = 0.1587
Peak Rain 1/2” too much	

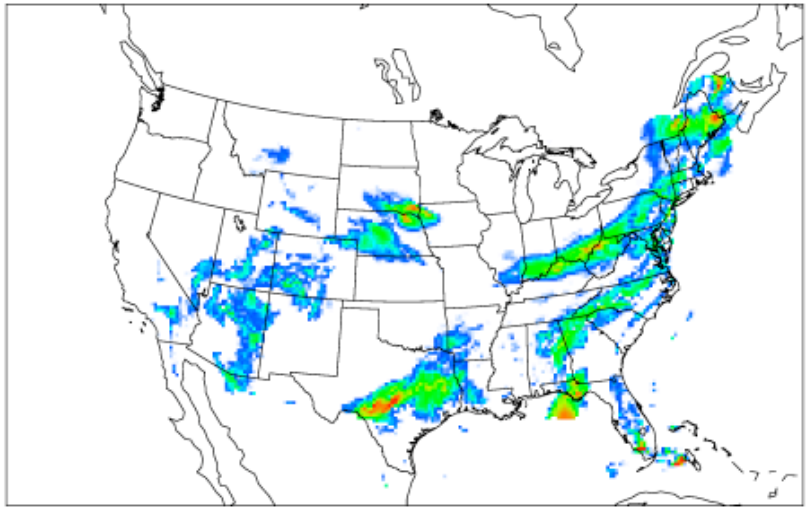
Verifying with objects doesn't always make sense . . .



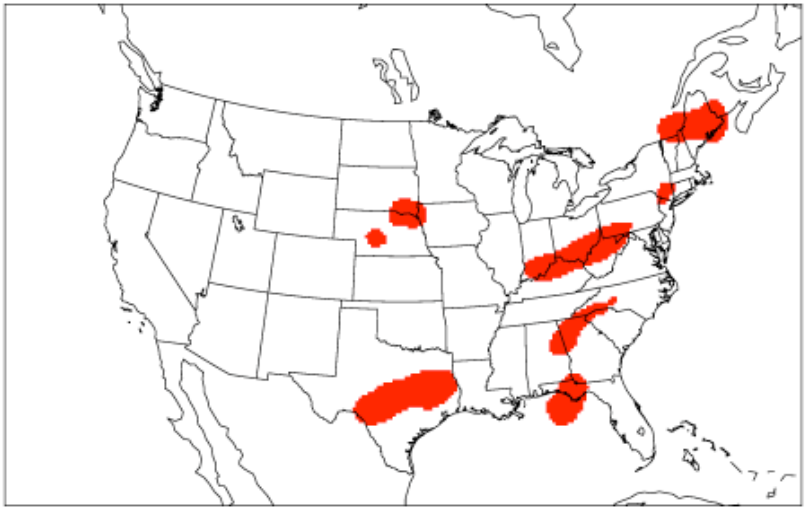
- In MET, object based verification is done using the MODE (Method for Object-Based Diagnostic Evaluation) tool.
  - Define objects
  - Compute attributes (e.g. area, centroid, axis angle, intensity)
  - Merge objects (e.g. thunderstorm cells merge into line)
  - Match forecast and observed objects
  - Compare attributes between matches
  - Output summary statistics



What are Objects?

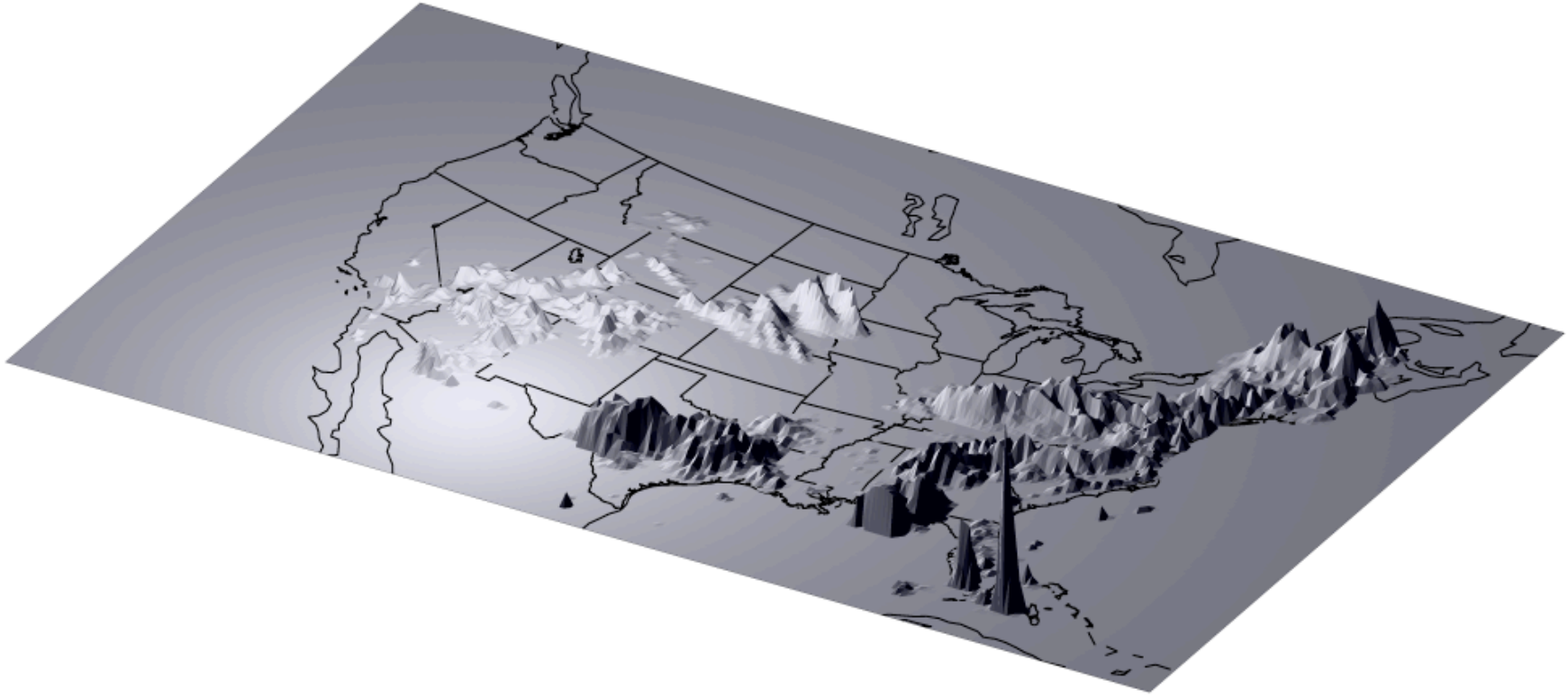


Raw Field



Object Field

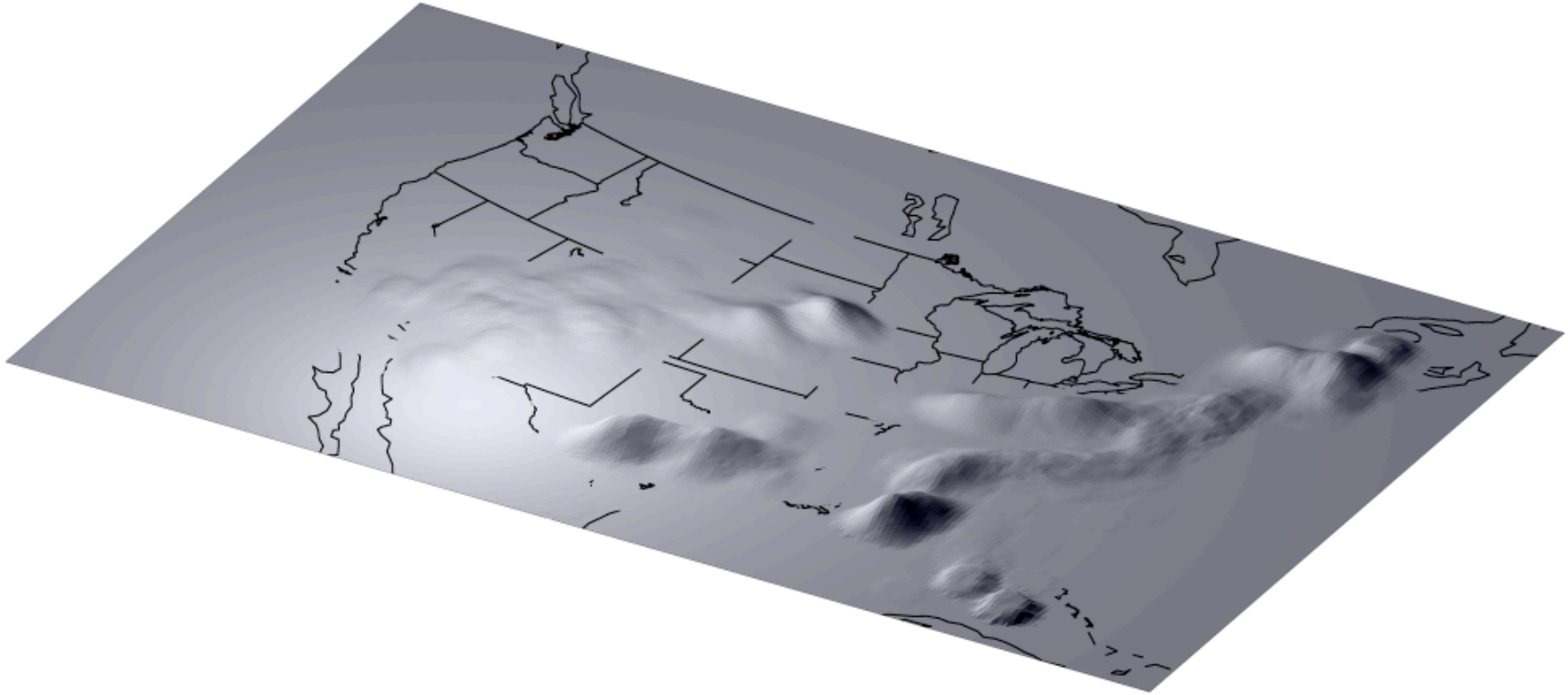
Objects are Regions of Interest



Step #1

Start with the raw data field.

In this case, a precipitation field.

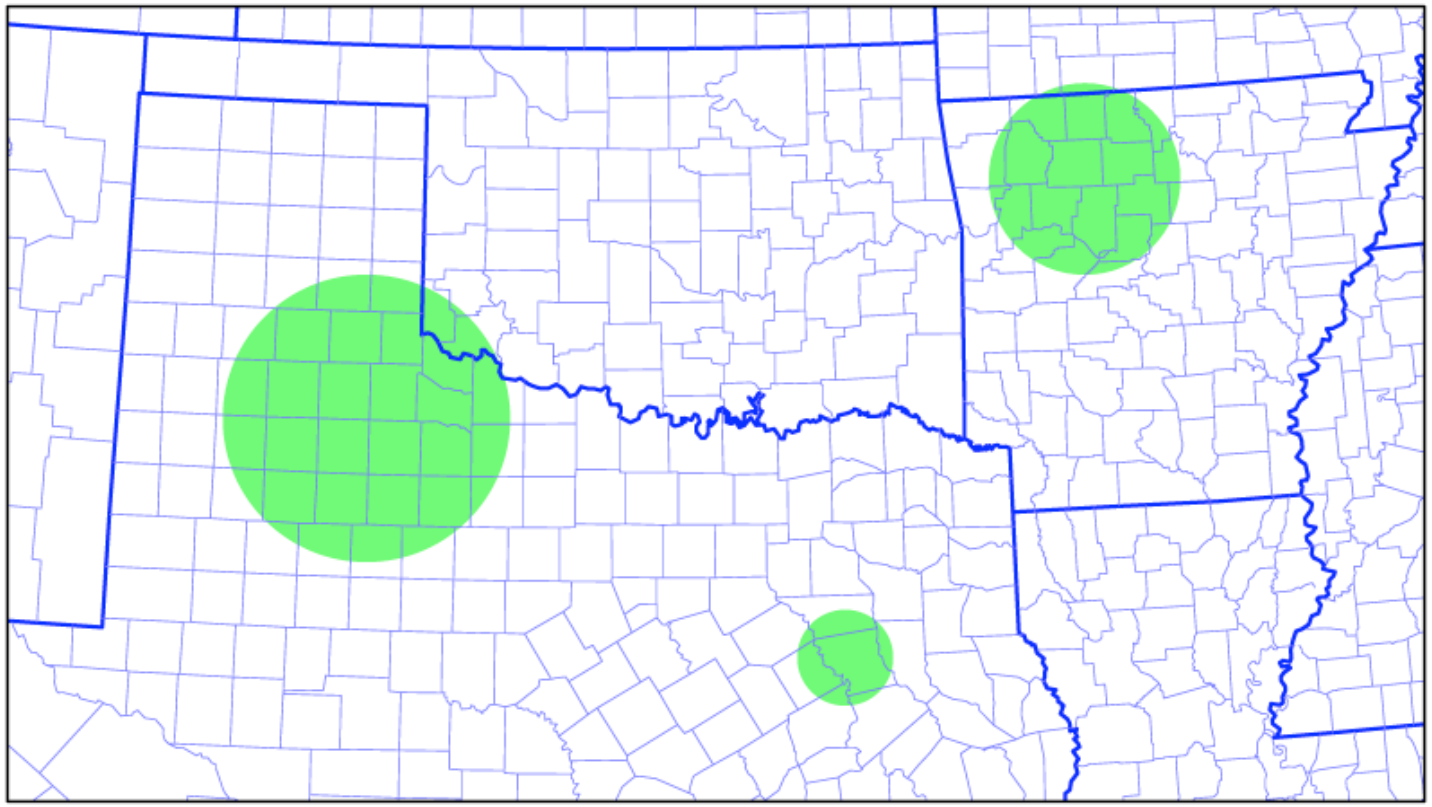


Step #2

Apply convolution operator.

This is basically a smoothing operation.

Convolution Radius



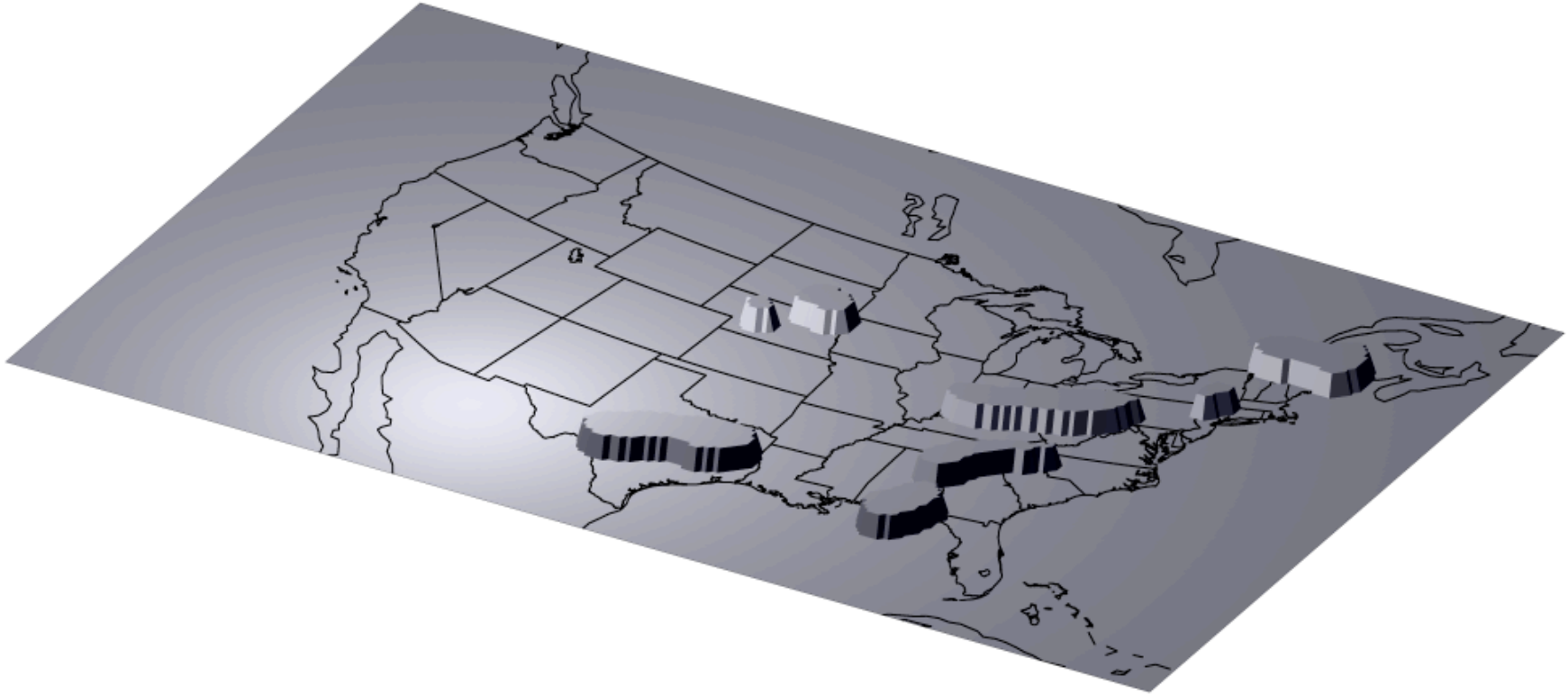
Radius of Influence

Convolution

Uses raw field $f(x, y)$ and

filter function $\phi(x, y)$

$$C(x, y) = \sum_{(\hat{x},\hat{y})\in G} \phi(\hat{x}, \hat{y})\, f(x - \hat{x}, y - \hat{y})$$



Step #3

Threshold the smoothed field.

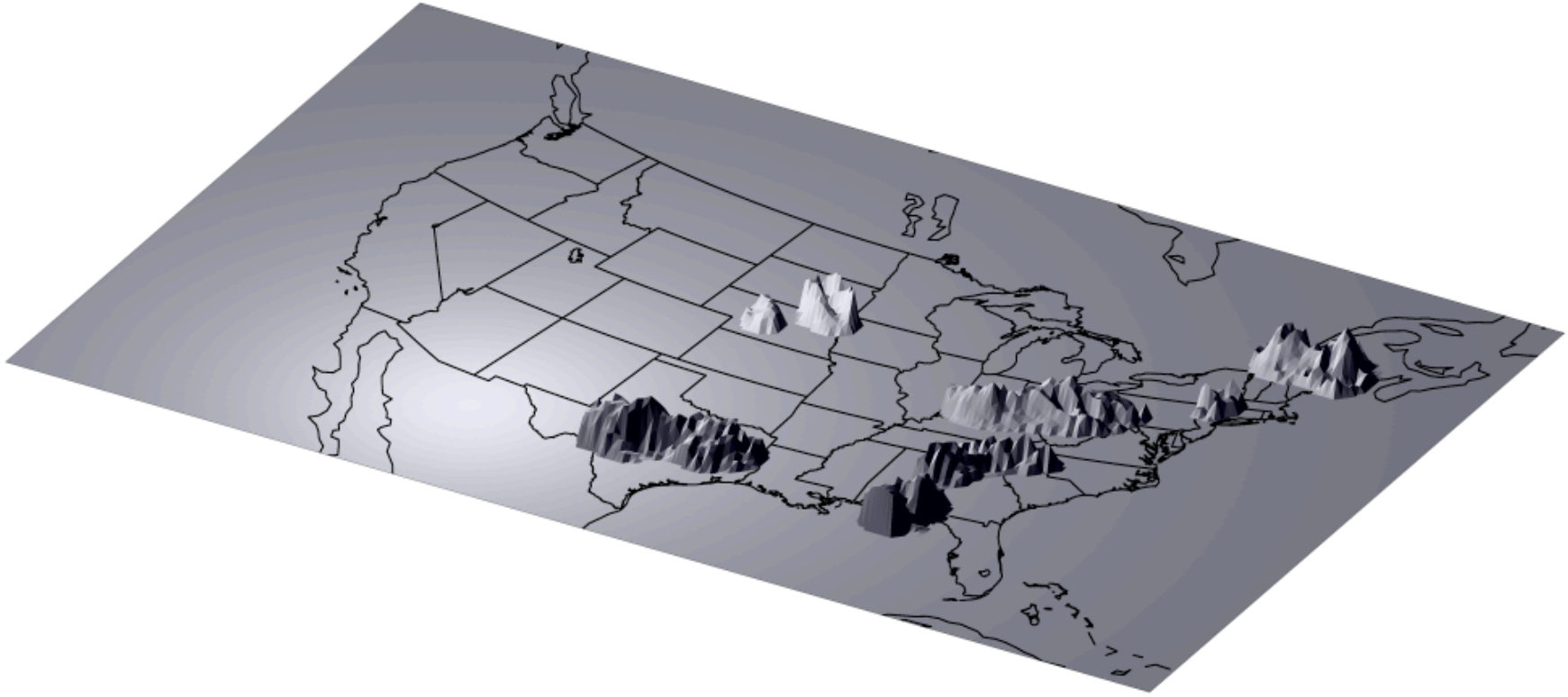
This produces an on/off mask field.

Masking

Uses convolved field $C(x, y)$

and threshold T

$$M(x,y) = \begin{cases} 1 & \text{if } C(x,y) \geq T \\ \\ 0 & \text{else} \end{cases}$$

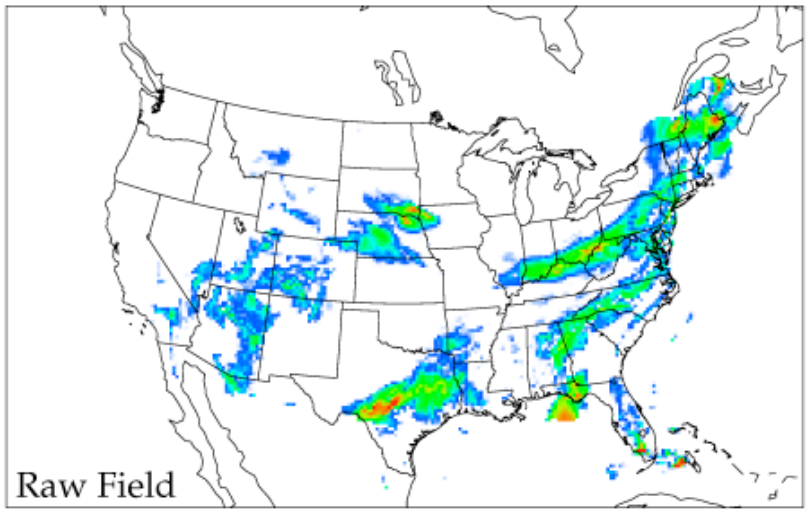


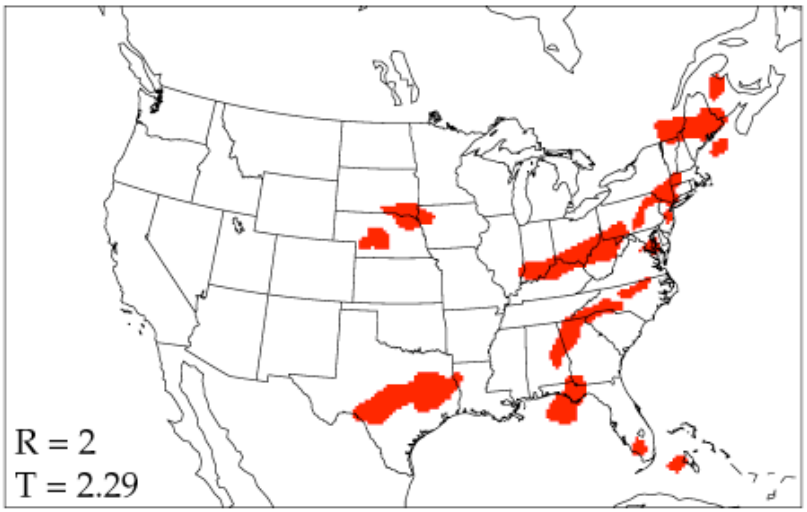
Step #4

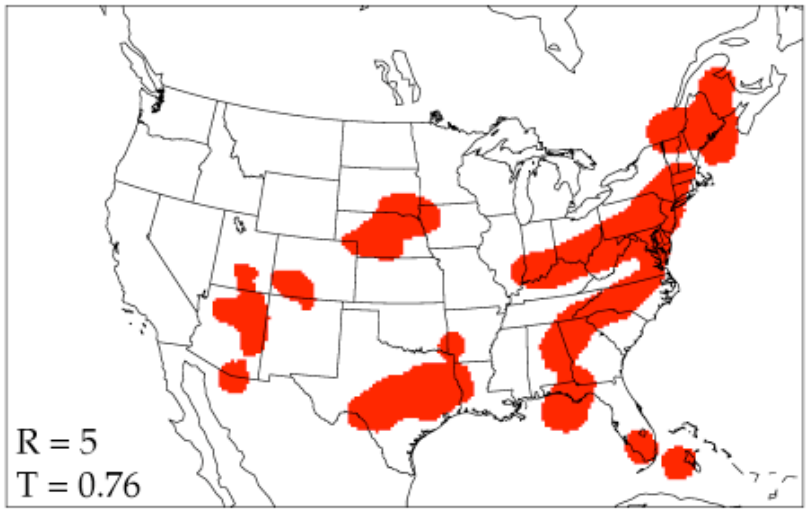
Restore original data to object interiors.

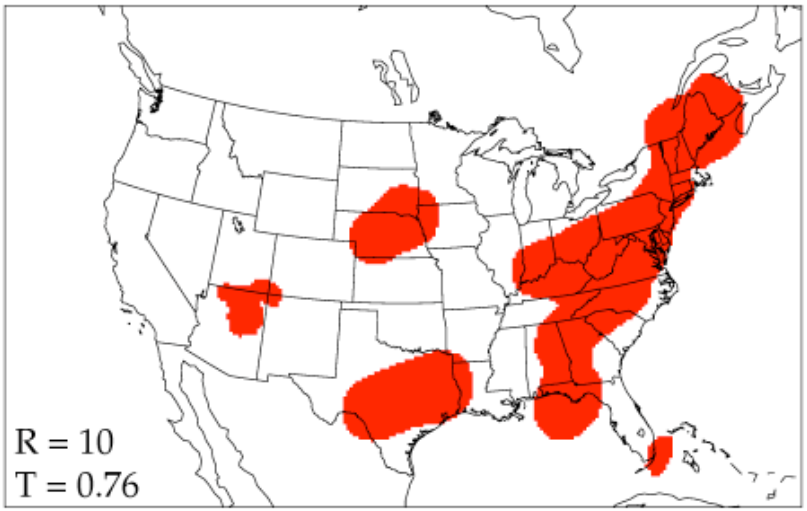
This gives us our objects.

Changing Object-Definition Parameters









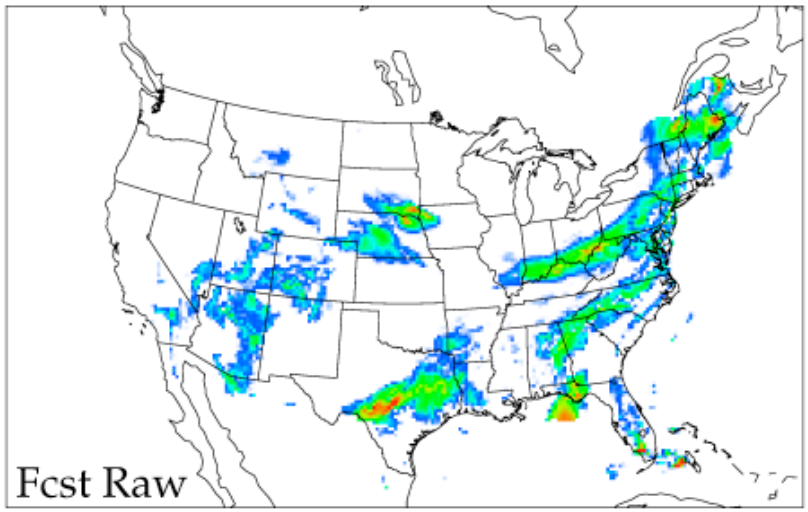
Matching & Merging

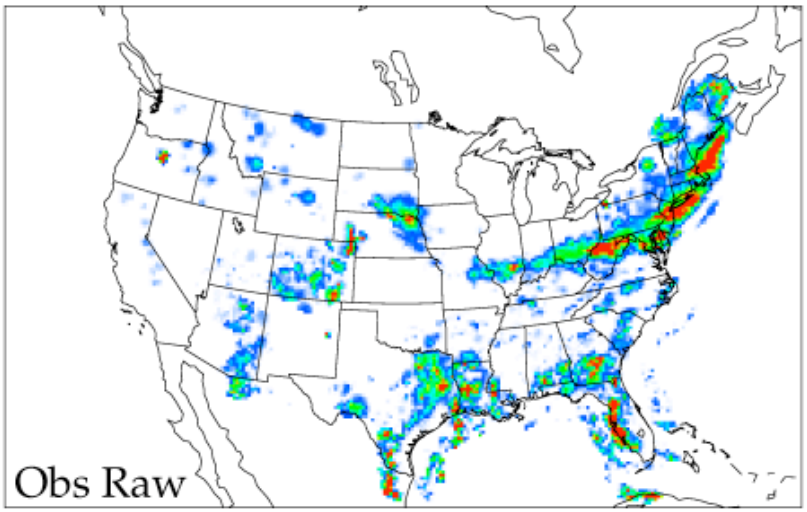
Merging: Associating objects in the same field.

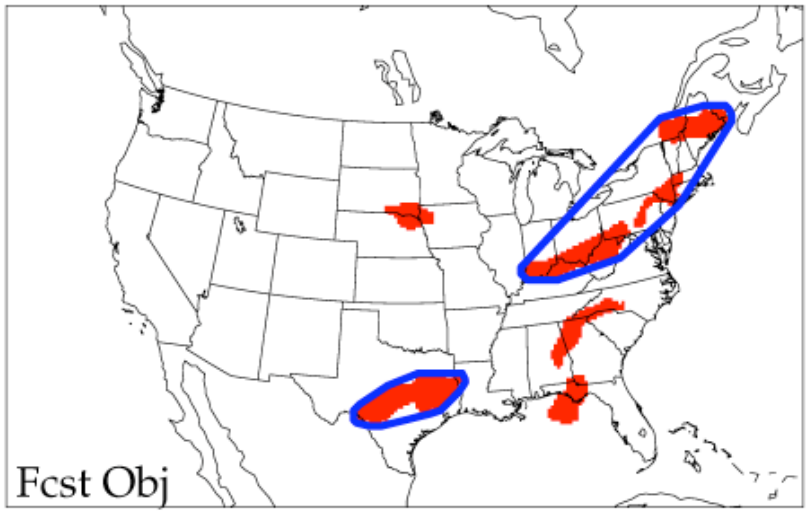
Matching: Associating objects in different fields.

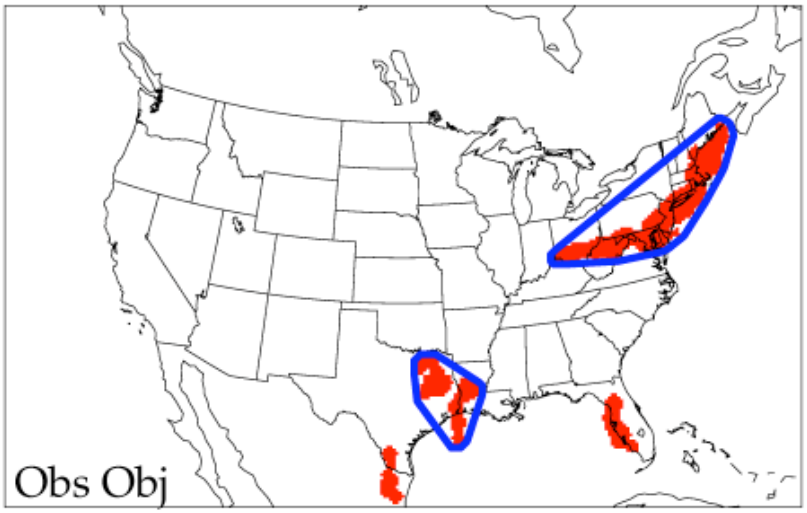
MODE does this using a Fuzzy-Logic engine.

Example of Matching & Merging









Object Attributes

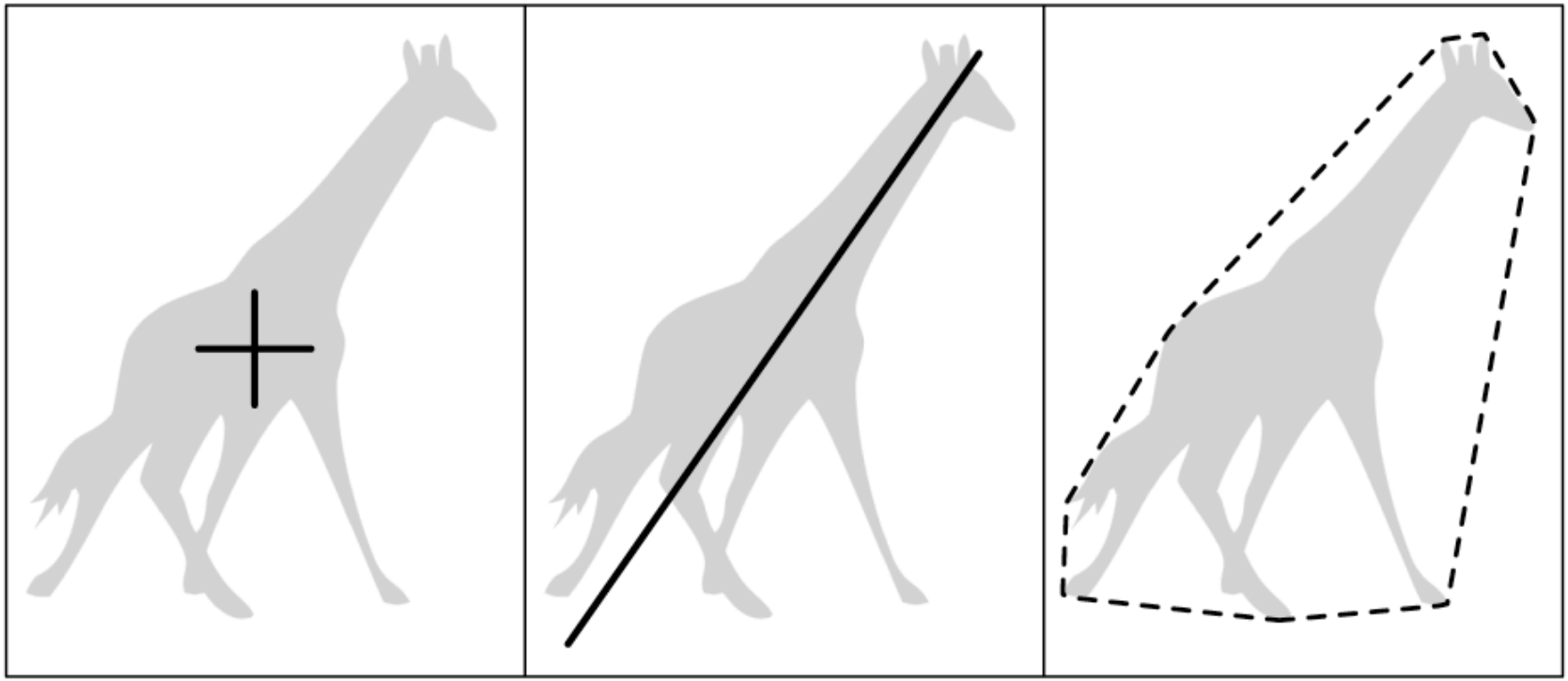
Single:

- Area
- Centroid
- Axis Angle
- Median Intensity
- Complexity
- Aspect Ratio
- Curvature

Pair:

- Centroid Distance
- Angle Difference
- Median Intensity Ratio
- Intersection Area
- Convex Hull Distance
- Boundary Distance
- Area Ratio

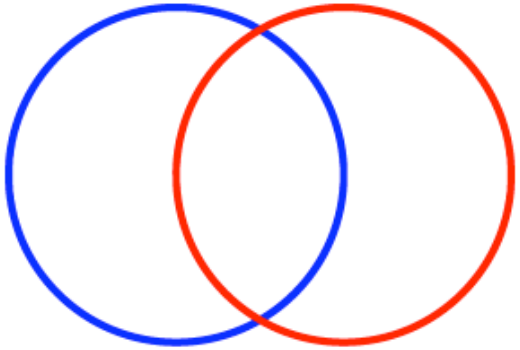
Example Single Attributes



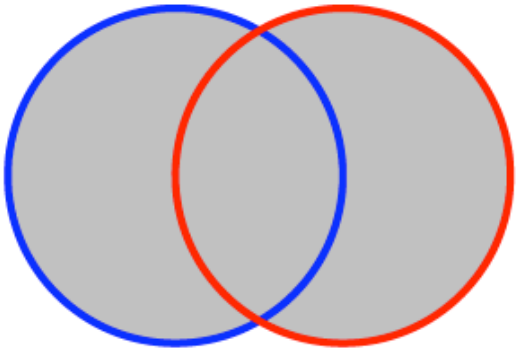
Centroid Axis Convex Hull

Example Pair Attributes

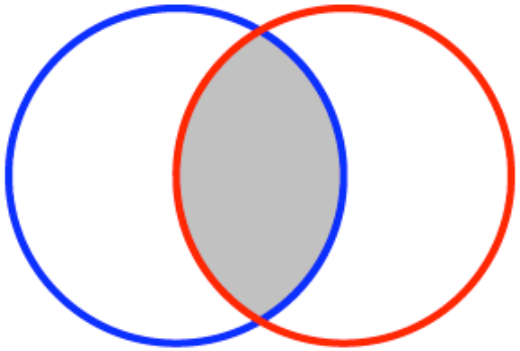
Forecast Object



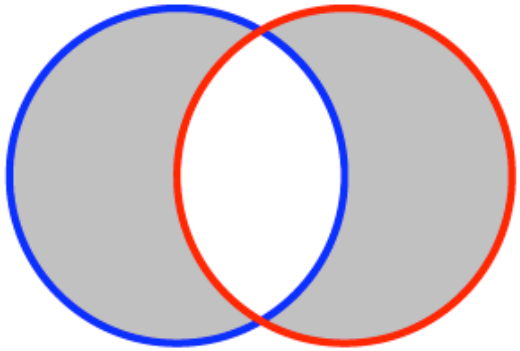
Observed Object



Union



Intersection

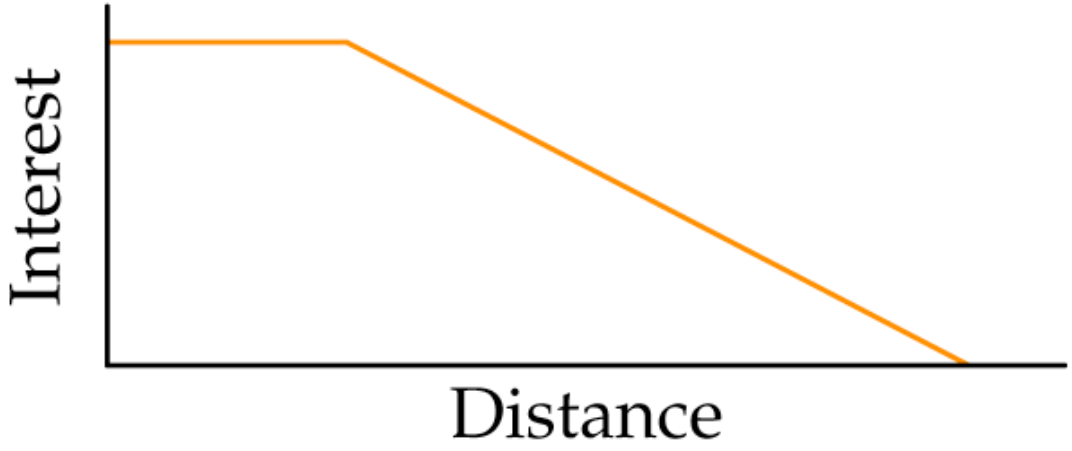


Symmetric Difference

Interest Maps

Map attributes to interest values.

Example: Centroid Distance



All interest maps can be changed in the config file.

Weights

Express relative importance of different attributes in matching and merging.

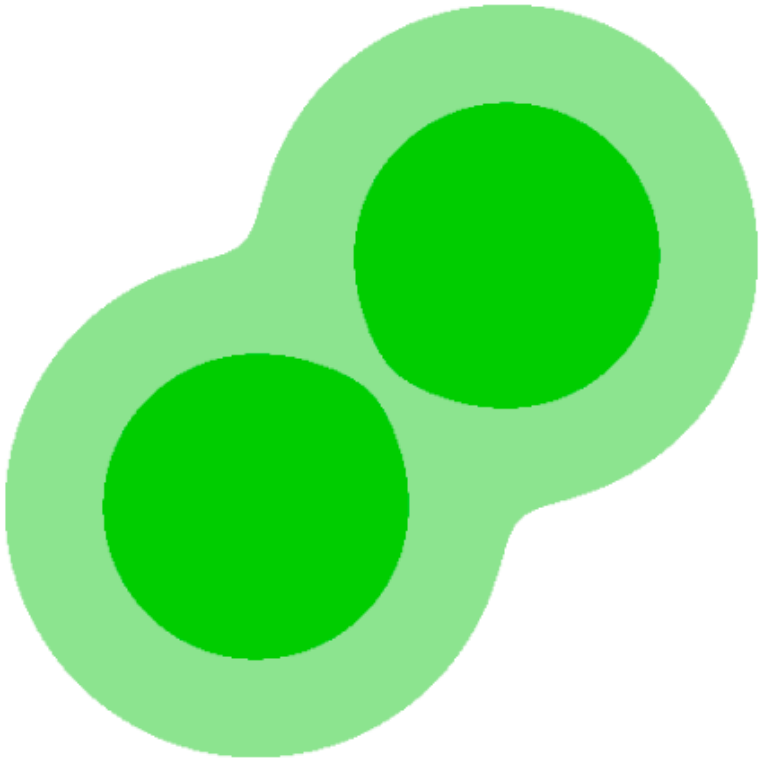
All weights can be changed in the config file.

Total Interest

Calculated from weights, attributes, and interest maps.

$$T(\alpha) = \frac{\sum_i w_i \, C_i(\alpha) \, I_i(\alpha)}{\sum_i w_i \, C_i(\alpha)}$$

Alternative Merging Method

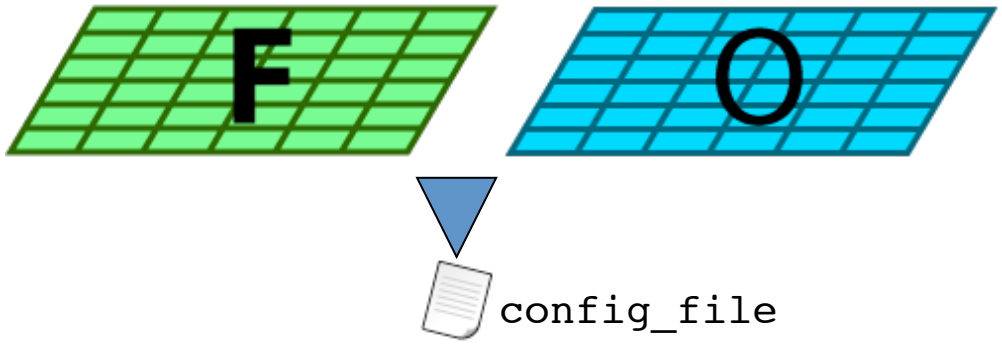


Double Thresholding

Verifying WRF with Objects

Running MODE and MODE analysis tool

Running MODE - Input



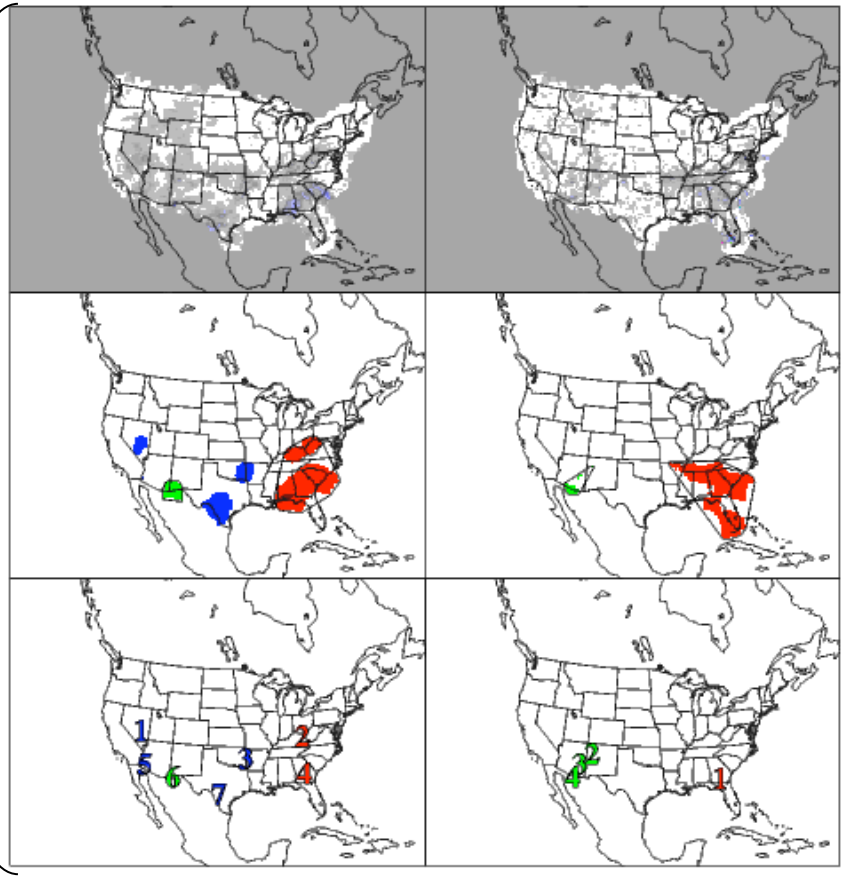
- 2 fields on the same grid
  - forecast (fcst_file) and observation (obs_file)
- configuration settings (config_file)

- syntax
  - `>mode fcst_file obs_file config_file`

Running MODE - Output

- PostScript
  - object pictures
  - parameter summary
  - total interest for each object pair

MODE: APCP_12 at SFC vs APCP_12 at SFC



Fcst	Obs	Interest
4	1	0.9173
6	4	0.8301
2	1	0.5589
6	3	0.5027
5	4	0.4328
5	3	0.2430
6	2	0.2384
5	2	0.1111
1	3	0.0898
1	2	0.0123
4	2	NA
1	1	NA
5	1	NA
7	2	NA
6	1	NA
2	3	NA
3	3	NA
4	3	NA
7	1	NA
3	1	NA
7	3	NA
1	4	NA
2	4	NA
3	4	NA
4	4	NA
2	2	NA
3	2	NA
7	4	NA

	Forecast	Observation
Model:	WRF	
Field:	APCP_12	APCP_12
Level:	SFC	SFC
Units:	kg/m^2	kg/m^2
Initial:	20050807 00:00:00	20050807 12:00:00
Valid:	20050807 12:00:00	20050807 12:00:00
Accum:	12:00:00	12:00:00

	Forecast	Observation
Mask M/G/P:	on/off/off	on/off/off
Raw Thresh:	>=0.00	>=0.00
Conv Radius:	5 gs	5 gs
Conv Thresh:	>=2.50	>=2.50
Area Thresh:	>=0 gs	>=0 gs
Inten Thresh:	p100>=0.00	p100>=0.00
Merge Thresh:	>=1.25	>=1.25
Merging:	thresh	thresh
Matching:	match/merge	
Simple/M/U:	7/3/4	4/4/0
Area:	944 gs	654 gs
Area M/U:	628/316	654/0
Cluster:	2	2
MMI:	0.4328	0.6664
MMI (F+O):	0.5027	

Centroid/Boundary:	2.00	4.00
Convex Hull/Angle:	0.00	1.00
Area/Intersection Area:	1.00	2.00
Complexity/Intensity:	0.00	0.00
Total Interest Thresh:	0.70	

object pictures

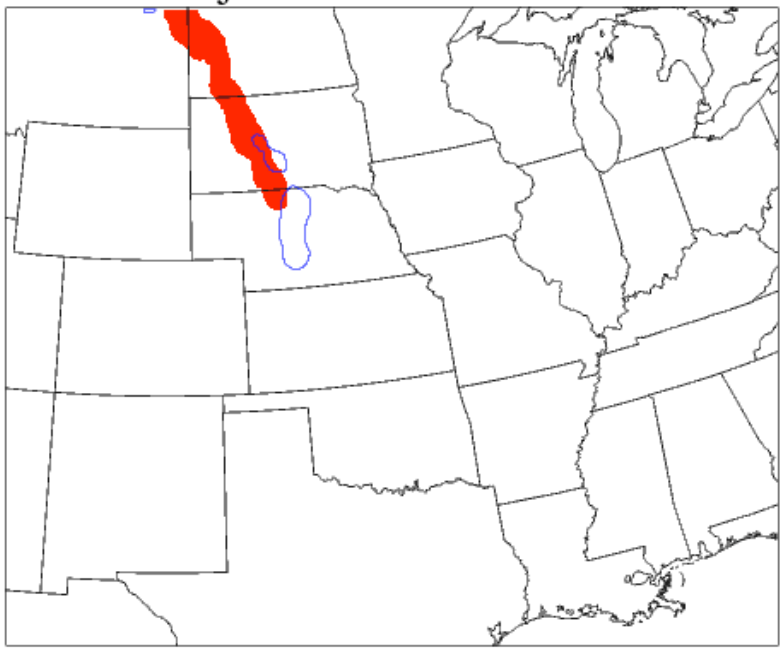
object-pairs total interest

<u>model info</u>
name, times

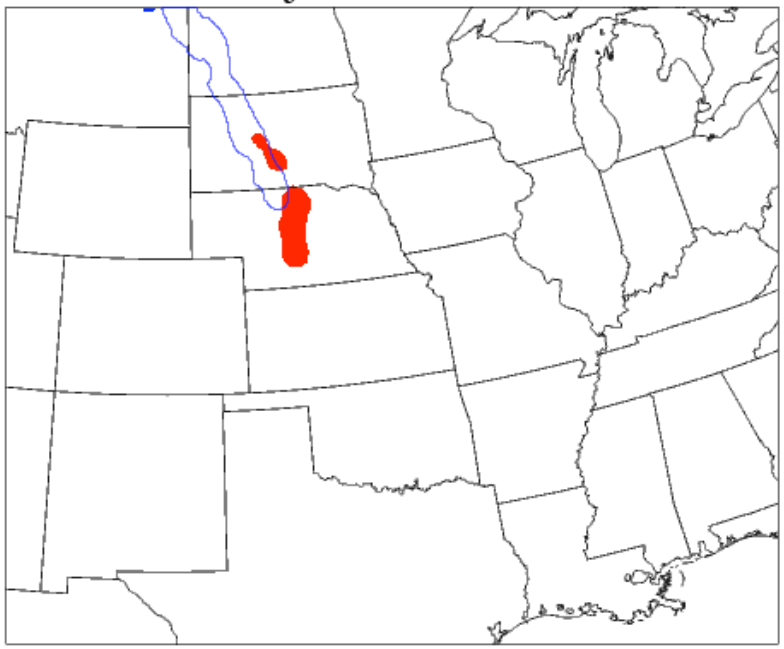
<u>parameter summary</u>
threshold
smoothing radius
match/merge
stats
MMI

object-attribute weights

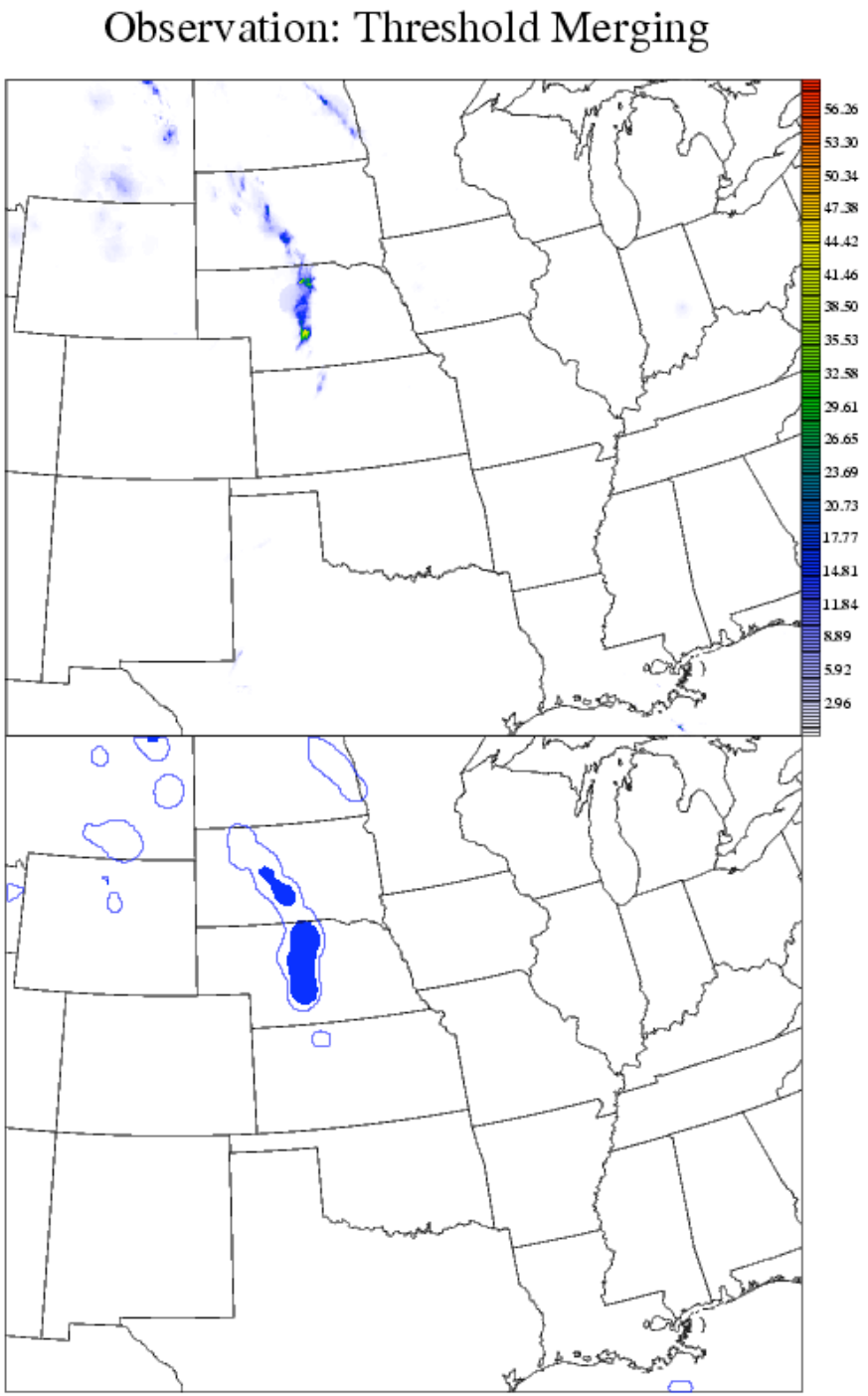
Forecast Objects with Observation Outlines



Observation Objects with Forecast Outlines



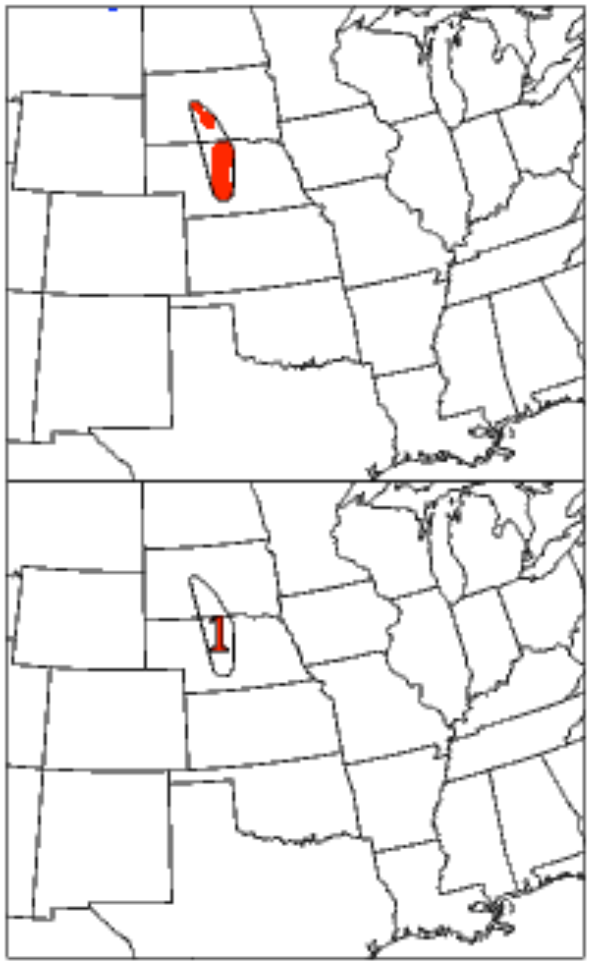
Merging Objects with Double Threshold



Cluster Object Information

Forecast

Observation



CLUS PAIR	CEN DIST	ANG DIFF	FCST AREA	OBS AREA	INTER AREA	UNION AREA	SYM DIFF	FCST INT50	OBS INT50	FCST INT90	OBS INT90	TOT INTR
1	97.32	12.16	4184	1762	358	5588	5230	5.33	4.06	16.51	15.57	0.7622

Running MODE - Output

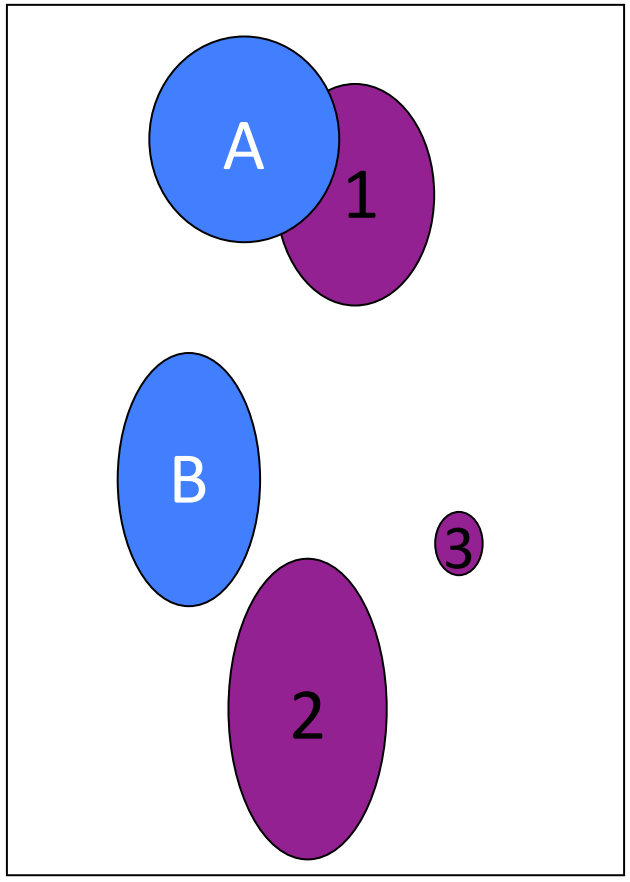
- PostScript
  - object pictures
  - parameter summary
  - total interest for each object pair
- ASCII
  - object sizes, shapes, positions
  - stats for simple, paired objects and clusters
  - standard contingency table stats on smoothed and thresholded fields
- netCDF
  - gridded object fields

Interpretation of MODE output

- How well do objects match?
  - median of the maximum interest (MMI)*
- What % of objects are matched?
- Displacement of matched objects
- mode_analysis tool

* Davis et al., 2009: The Method for Object-based Diagnostic Evaluation (MODE) Applied to WRF Forecasts from the 2005 SPC Spring Program. Weather and Forecasting

Median of the Max. Interest (MMI)



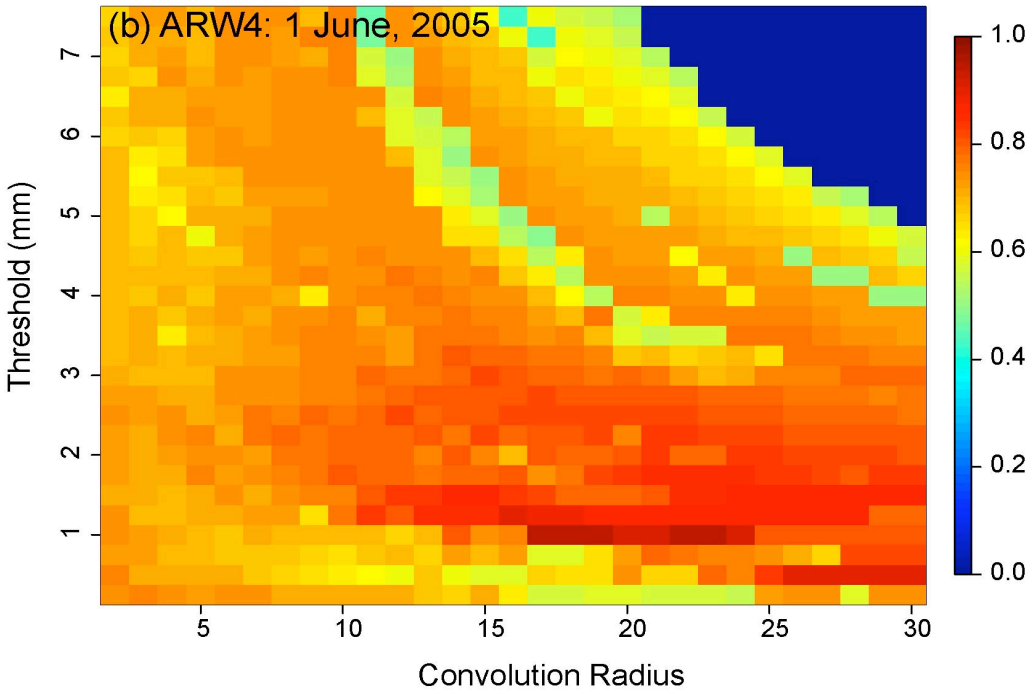
Interest Matrix

observed

forecast	A	B
1	0.90	0.65
2	0.50	0.80
3	0.40	0.55

MMI = median { 0.55, 0.80, 0.80, 0.90, 0.90 } = 0.80

Median of the Max. Interest (MMI) Quilt Plot



MMI as a function of convolution radius (grid squares) and threshold (mm) for 24-h forecast of 1-h rainfall

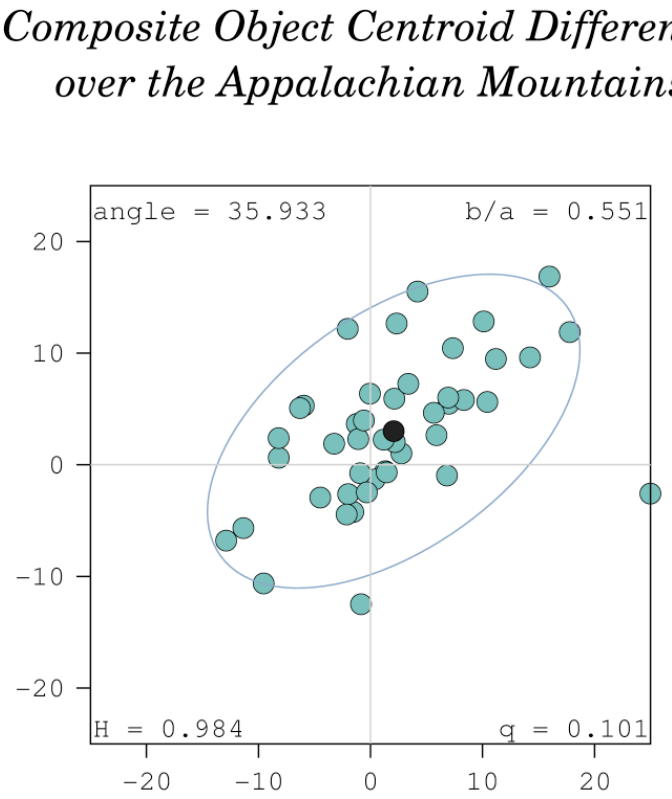
What % of Objects are Matched?

- Hit rate, Threat Score, Heidke Skill Score
- Other standard contingency table scores

	Forecasted object	Unforecasted object
Observed object	Matched	Missed
Unobserved object	False alarm	Correct null

MODE Analysis

- displacement of forecast clusters from matched observed clusters



MODE Analysis Tool
SUMMARY Example

Command Line

```
mode_analysis -summary
              -mask_file ttt -config config/mode_test_config    \
              -dump_lines out -lookin /d1/score/mode_files  \
              -fcst -composite -area_min 3000    \
              -centroid_x_min 600
              -centroid_x_max 1100    \
              -column CENTroid_x   \
              -column centroid_y    \
              -column centroid_lat   \
              -column centroid_lon    \
              -column area   \
              -column axis_ang   \
              -column length
```

Output

```
Total mode lines read =  73,330
Total mode lines kept =     539
```

Field	Min	Max	Mean	StdDev	P10	P25	P50	P75	P90
centroid_x	600.23	914.61	779.36	97.98	626.36	687.96	804.30	866.75	894.24
centroid_y	55.22	560.08	335.55	113.08	189.48	240.58	333.51	421.06	496.82
centroid_lat	22.00	40.14	32.08	4.06	26.83	28.66	32.00	35.15	37.87
centroid_lon	-107.03	-95.01	-100.18	3.75	-106.03	-103.67	-99.23	-96.84	-95.79
area	3210.00	85486.00	12680.96	9931.67	4935.00	6256.00	9445.00	16106.00	23678.00
axis_ang	-88.84	89.90	13.54	44.82	-55.80	-16.36	17.28	48.23	71.27
length	100.57	494.54	200.08	82.11	112.44	133.62	179.96	249.06	315.33

MODE Analysis Tool
By Case Example

Command Line

```
mode_analysis -bycase
              -dump_lines out
              -mask_file ttt -config config/mode_test_config
              -fcst_valid_min 20070702 -fcst_valid_max 20070702_12
              -area_min 3000
              -centroid_x_min 600 -centroid_x_max 1100
               /d1/score/mode_files/ncwf2_vs_ncwdp/* /d1/score/mode_files/rcpf_vs_ncwdp/*
```

Output

Fcst Valid Time	Area Matched	Area Unmatched	# Fcst Matched	# Fcst Unmatched	# Obs Matched	# Obs Unmatched
Jul 2, 2007 00:00:00	12392	20786	0	1	1	1
Jul 2, 2007 01:00:00	6706	11038	0	0	1	2
Jul 2, 2007 02:00:00	7507	18696	0	0	1	3
Jul 2, 2007 03:00:00	19401	32268	2	3	1	2
Jul 2, 2007 04:00:00	0	16551	0	2	0	1
Jul 2, 2007 05:00:00	15311	29730	1	2	1	2
Jul 2, 2007 06:00:00	4730	8182	0	0	1	2
Jul 2, 2007 07:00:00	3733	13285	0	1	1	2
Jul 2, 2007 08:00:00	6994	6994	0	0	1	1
Jul 2, 2007 09:00:00	15981	15981	0	0	2	2
Jul 2, 2007 10:00:00	51501	53427	2	2	4	4
Jul 2, 2007 11:00:00	15779	21089	1	1	1	2
Jul 2, 2007 12:00:00	31339	40665	1	2	2	2

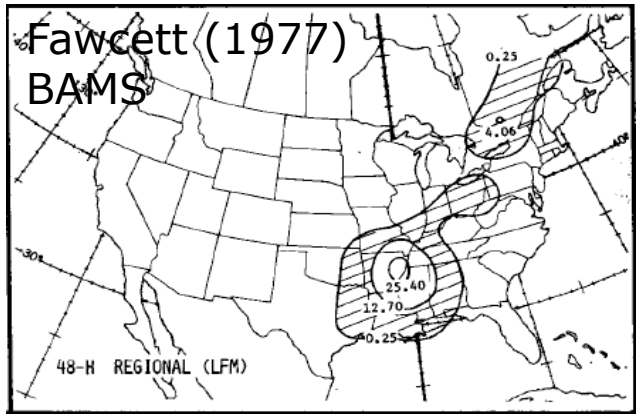
Why Use Spatial Scales?

Neighborhood and Scale-separation Approaches

Challenge of Higher Resolution Models

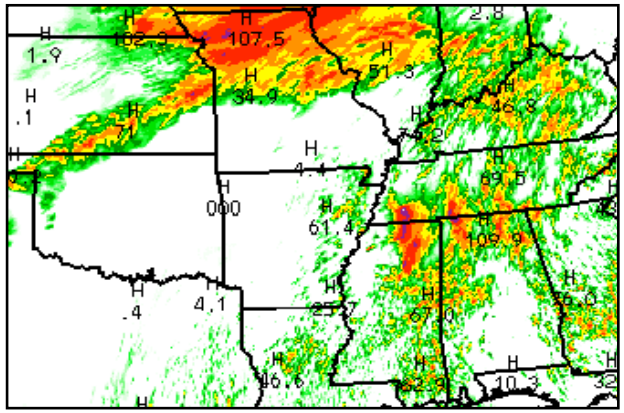
- Scales have shrunk
- A *one* grid-point displacement error often treated same as *ten* grid-point error

THEN



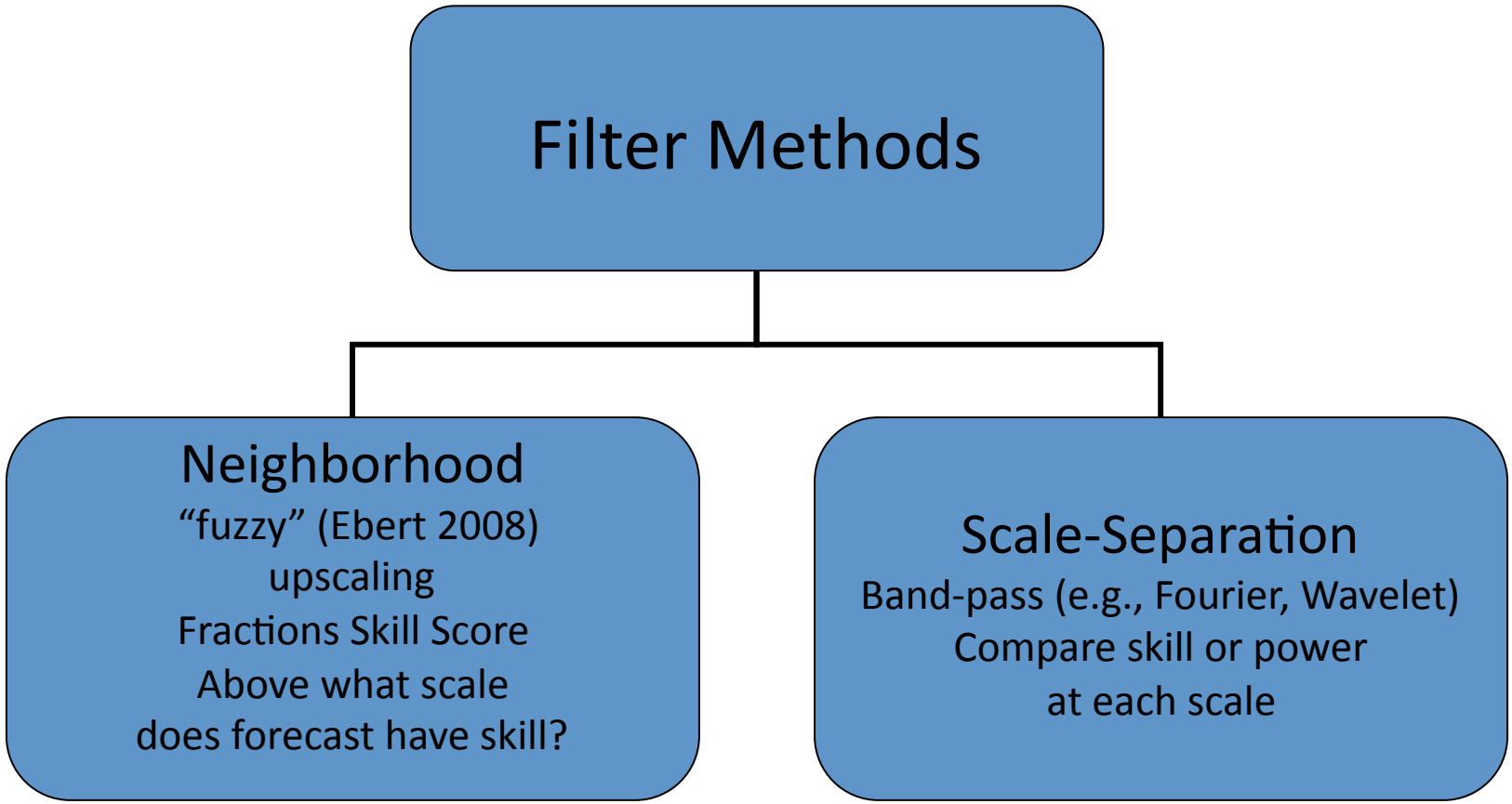
190-km LFM

NOW



3-km WRF, 2009

Giving Credit for a Close Forecast

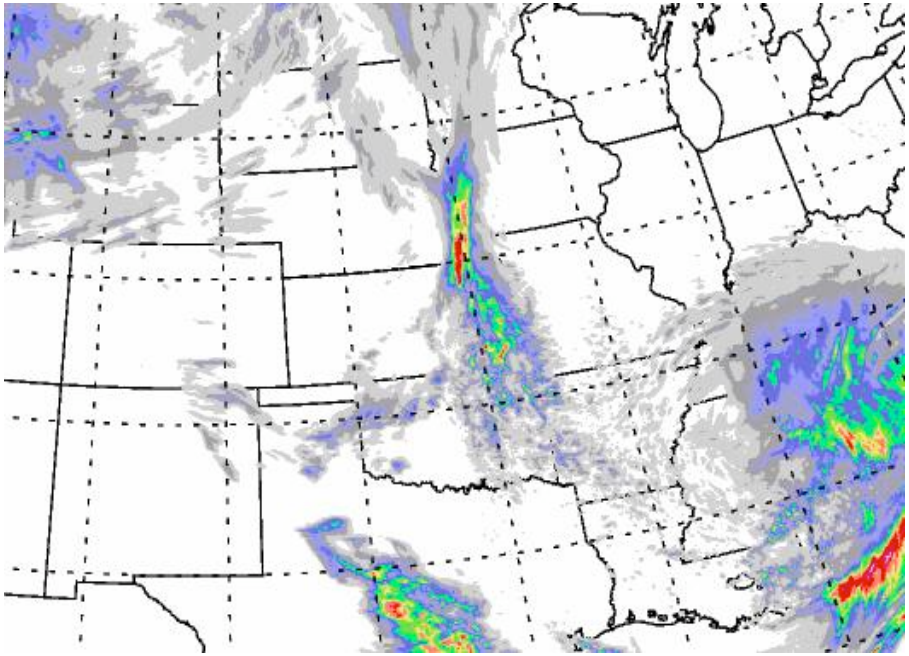


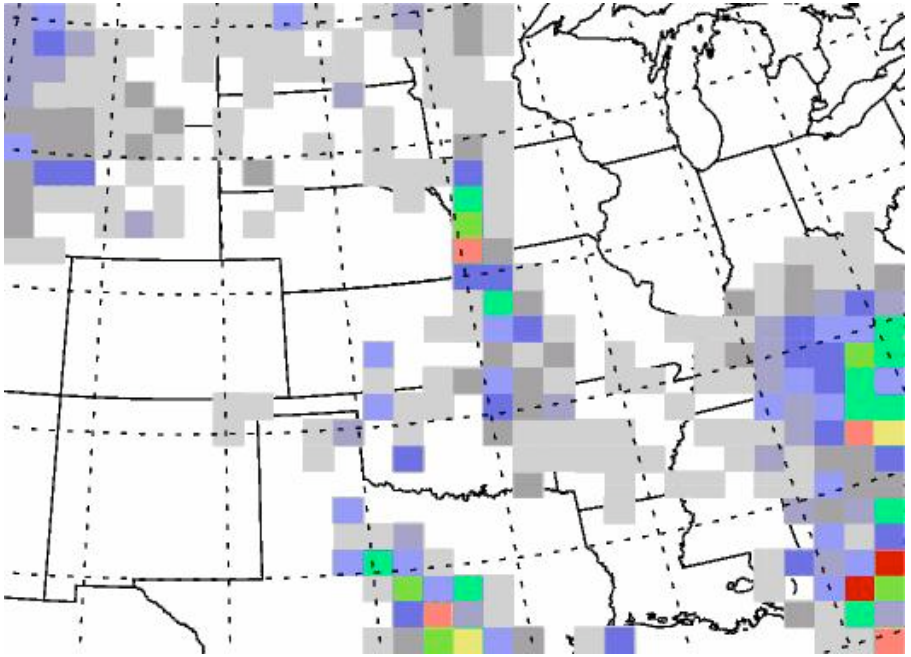
Neighborhood Methods

- Upscaling or coarsening the grid
- Filter options
  - shape (***square***, round)
  - size
  - type (***unweighted-mean, min, median, max***)

*MET options in **italic***

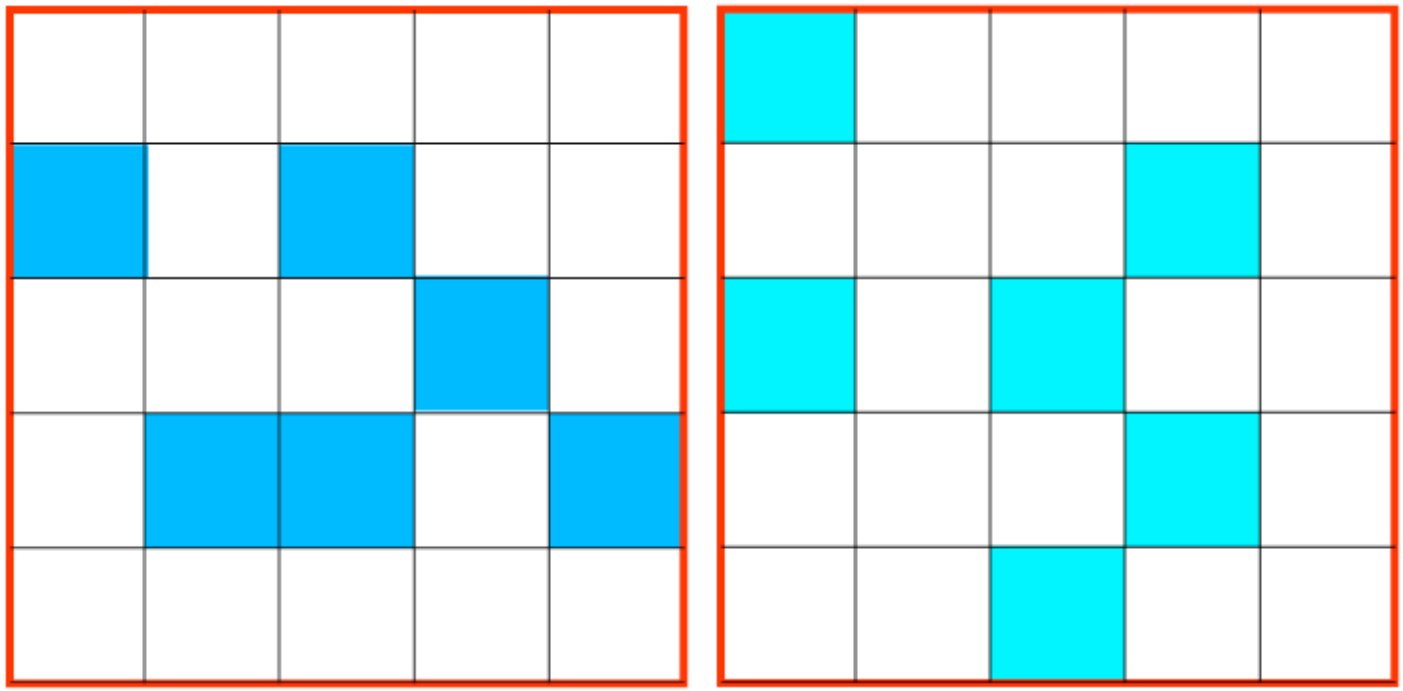
Neighorhood Methods





upscaling

Neighborhood Methods
Comparing Fractional Coverage

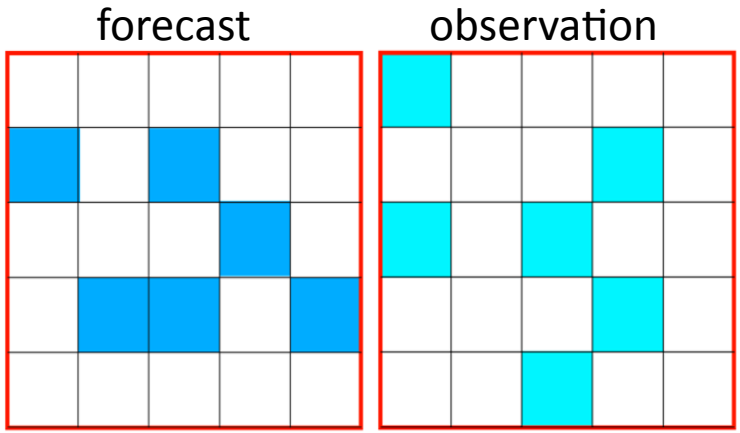


Fraction = 6/25 = 0.24

Fraction = 6/25 = 0.24

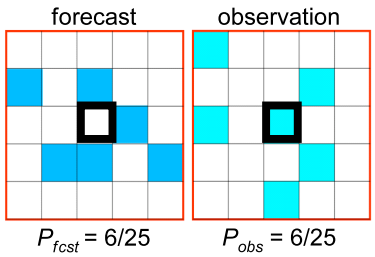
Threshold exceeded where squares are blue

Neighborhood Methods



P is the event frequency
within the neighborhood

Neighborhood Methods



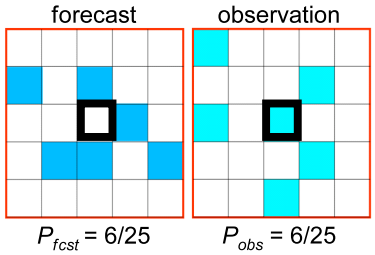
n = number of points in the whole domain

$$\frac{1}{n}\sum_{i=1}^{n}\left(P_{f\,cst}-P_{obs}\right)^2$$

Fractions Brier Score

Roberts and Lean (2008)

Neighborhood Methods



$$1 - \frac{\frac{1}{n}\sum_{i=1}^{n}\left(P_{f\,cst} - P_{obs}\right)^2}{\frac{1}{n}\sum_{i=1}^{n}\left(P_{f\,cst}\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(P_{obs}\right)^2}$$

Fractions Skill Score

Roberts and Lean (2008)

Neighborhood Methods

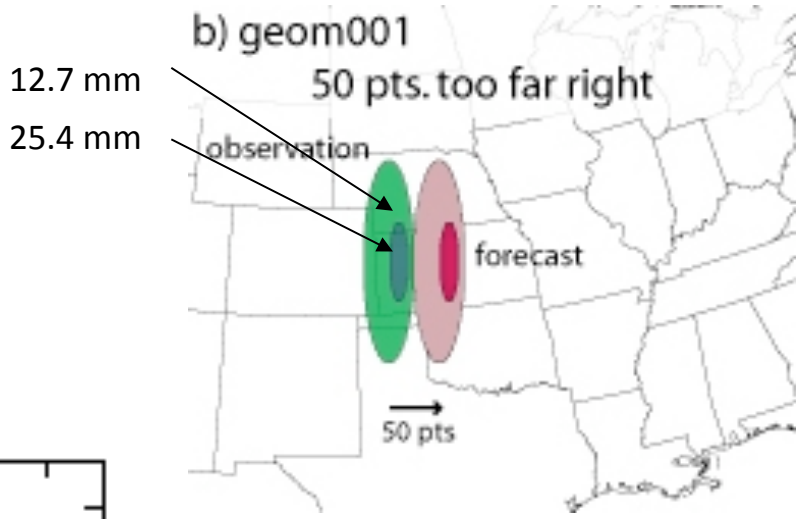


Table courtesy of E Ebert.

Fractions skill score — FSS

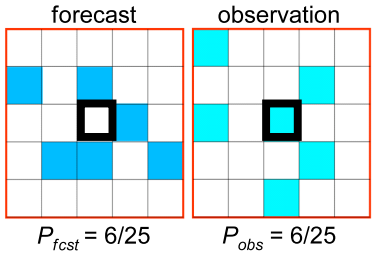
Spatial scale (pts) \ Threshold (mm)	1	2	5	10	20	50	100	200
129	0.69	0.69	0.69	0.69	0.64			
65	0.34	0.34	0.34	0.34	0.25			
33	0.09	0.09	0.09	0.09	0.00			
17	0.03	0.03	0.03	0.03	0.00			
9	0.01	0.01	0.01	0.01	0.00			
5	0.00	0.00	0.00	0.00	0.00			
3	0.00	0.00	0.00	0.00	0.00			
1	0.00	0.00	0.00	0.00	0.00			

good

bad

1
0.9
0.8
0.7
0.60
0.5
0.4
0.3
0.20
0.1
0
−0.1

Neighborhood Methods



$$FSS_{usef\ ul} = 0.5 + \frac{\overline{P_{obs}}}{2}$$

Fractions Skill Score must exceed this to be better than random forecast.

$\overline{P_{obs}}$ is the observed frequency of the event over the full domain.

Neighborhood Methods

- In MET, Neighborhood methods are in grid_stat tool.
- Neighborhood filters in MET:
  - Minimum
  - Maximum
  - Median
  - Mean
  - Fractions Brier Score
  - Fractions Skill Score
- See Ebert (2008) for a good summary and comparison of these techniques (and references).

Neighborhood Methods

Slide from E Ebert. See Ebert (2008) for full references.

Fuzzy method	Matching strategy*	Decision model for useful forecast
Upscaling (Zepeda-Arce et al. 2000; Weygandt et al. 2004)	NO-NF	**Resembles obs when averaged to coarser scales**
Minimum coverage (Damrath 2004)	NO-NF	**Predicts event over minimum fraction of region**
Fuzzy logic (Damrath 2004), joint probability (Ebert 2002)	NO-NF	**More correct than incorrect**
Fractions skill score (Roberts and Lean 2007)	NO-NF	**Similar frequency of forecast and observed events**
Area-related RMSE (Rezacova et al. 2006)	NO-NF	**Similar intensity distribution as observed**
Practically perfect hindcast (Brooks et al. 1998)	NO-NF	**Resembles a forecast based on perfect knowledge of observations**
Pragmatic (Theis et al. 2005)	SO-NF	**Can distinguish events and non-events**
CSRR (Germann and Zawadzki 2004)	SO-NF	**High probability of matching observed value**
Multi-event contingency table (Atger 2001)	SO-NF	**Predicts at least one event close to observed event**

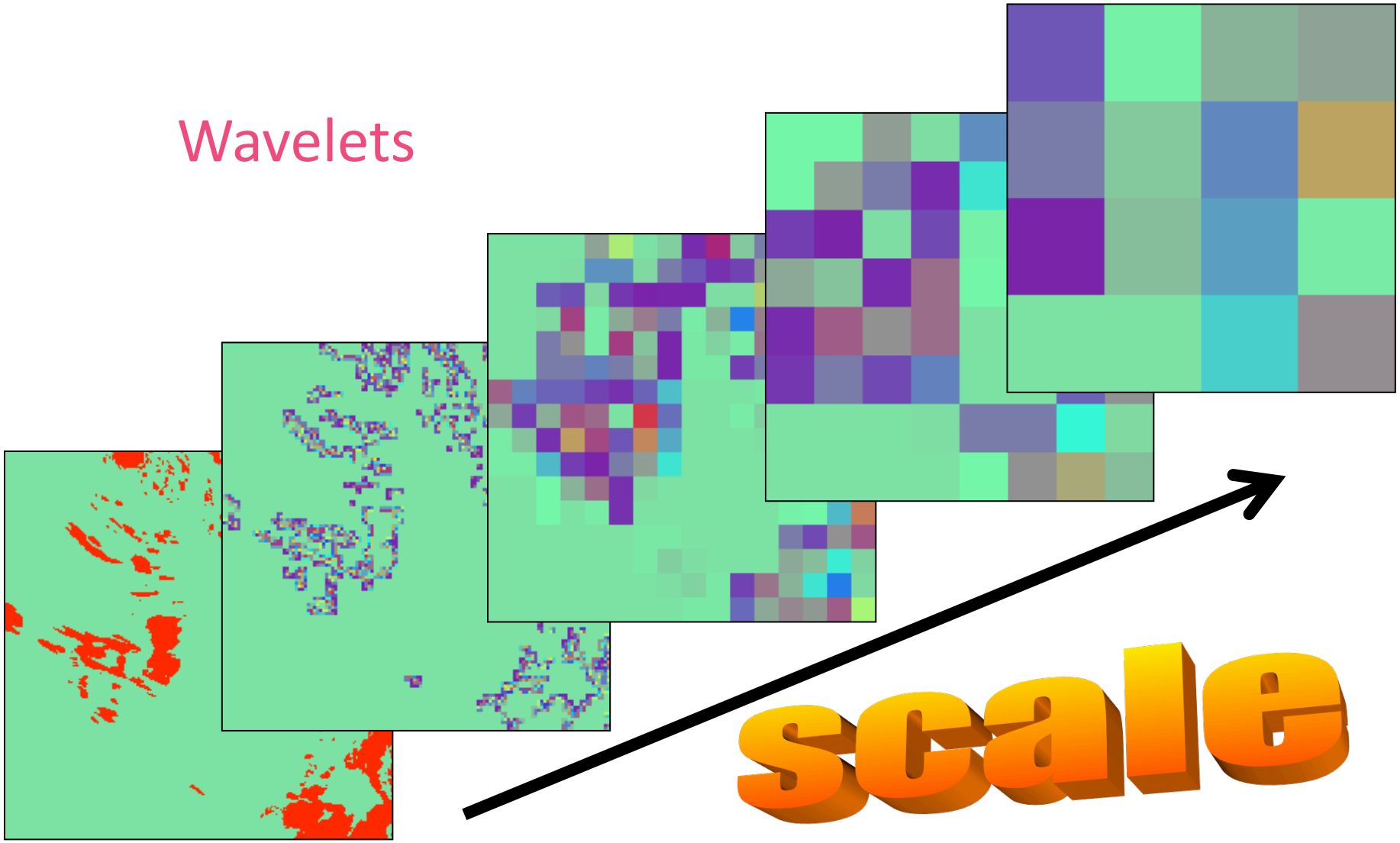
*NO-NF = neighborhood observation-neighborhood forecast,
SO-NF = single observation-neighborhood forecast

Scale-Separation Methods

- Fourier
  - Skamarock (2004), MWR **132**:3019-3032
  - Harris *et al*. (2001), J Hydrometeorol. **2**:406-418
  - Tustison *et al*. (2001), JGR **106**(D11):11775-11784
  - *and many more…*
- Wavelet
  - Briggs and Levine (1997), MWR **125**:1329-1341
  - Casati *et al*. (2004). [In MET wave_stat tool]

Scale-Separation Methods

Wavelets



Scale Separation Methods

Intensity Scale (IS) (Casati *et al.*, 2004)

1. Create binary fields for a threshold, and calculate MSE (denote, MSE(t)).
2. Apply wavelet decomposition to binary fields.
3. Calculate MSE for each threshold (t) and scale (j) (denote, MSE(t,j)), and find MSE%(t,j)=MSE(t,j)/MSE(t).
4. Find $MSE(t)_{random}$ = FBI*Br*(1-Br) + Br*(1-FBI*Br), where FBI=(a+b)/(a+c) is the frequency bias index and Br=(a+c)/n is the sample climatology from the contingency table.
5. IS SkillScore(t,j)=1-MSE(t,j)*(n+1)/$MSE(t)_{random}$.

Scale-Separation Methods

Wavelets: Practical considerations

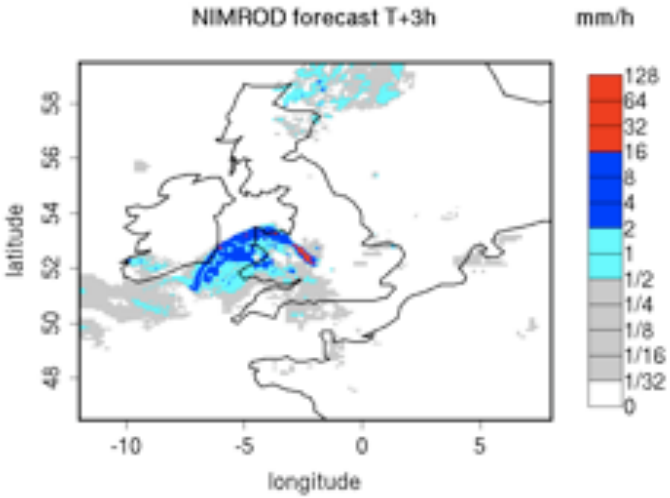
- Applicable only on square grids of size, 2^n.
- If grid does not meet this, then must alter the grid in some way. For example,
  - Pad with zeros,
  - Cut the domain down.
- Several choices for wavelets. Haar Wavelets are a good choice for precipitation (because of their discrete nature), but other choices might be better for other fields.

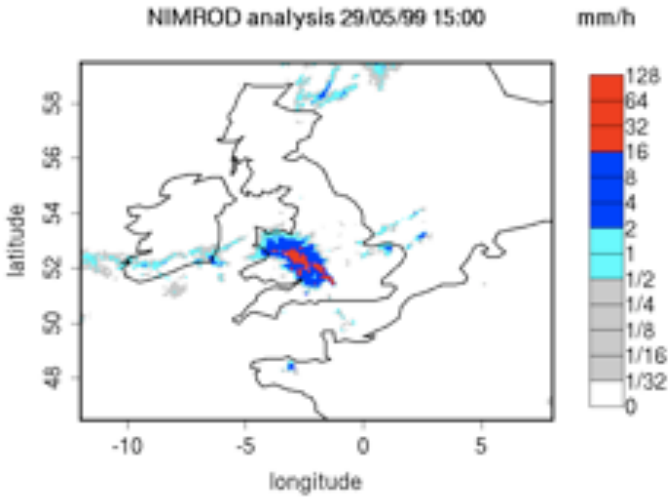
Casati *et al*.(2004)

Forecast

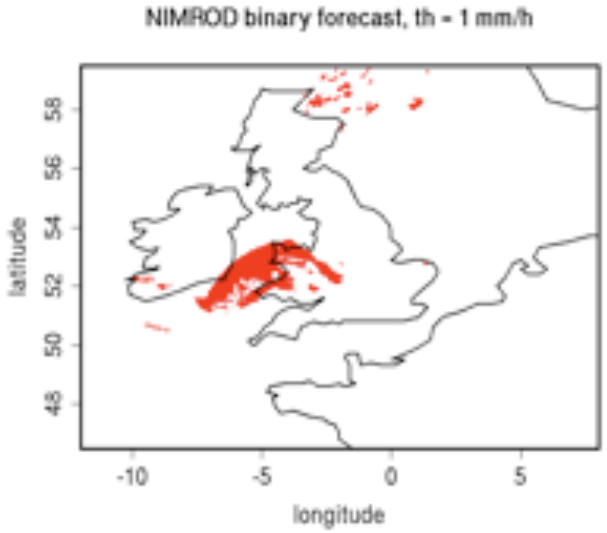
Observation

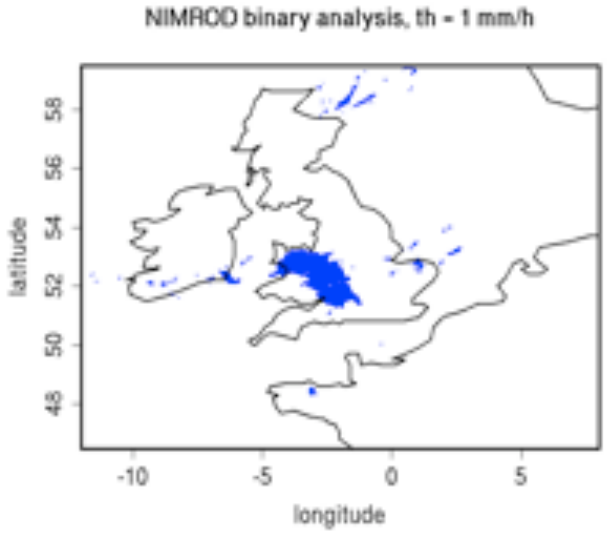
Raw Data





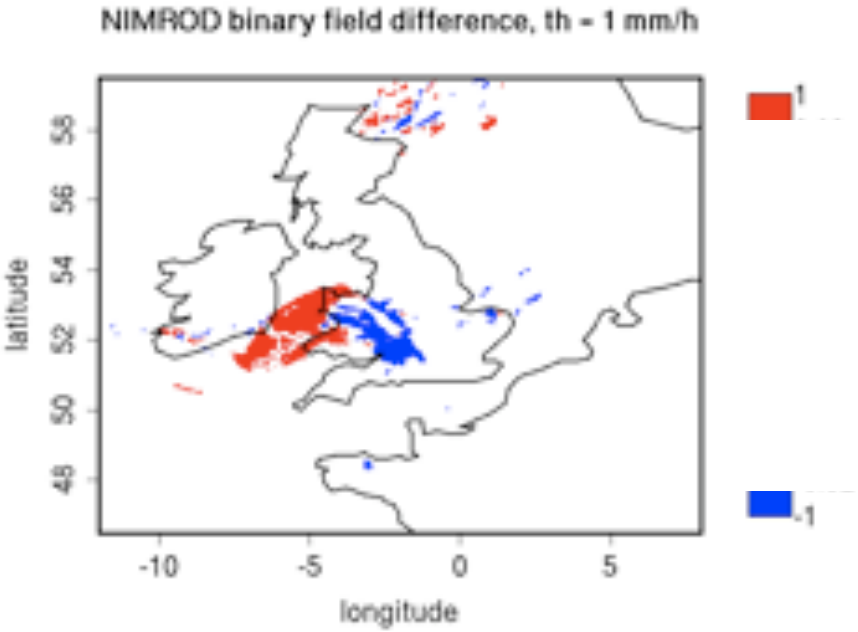
Binary Data



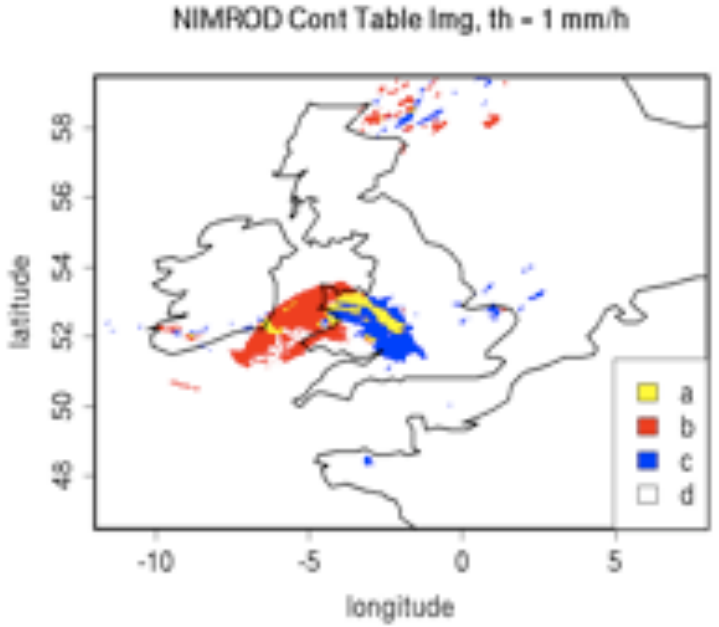


Casati *et al*.(2004)

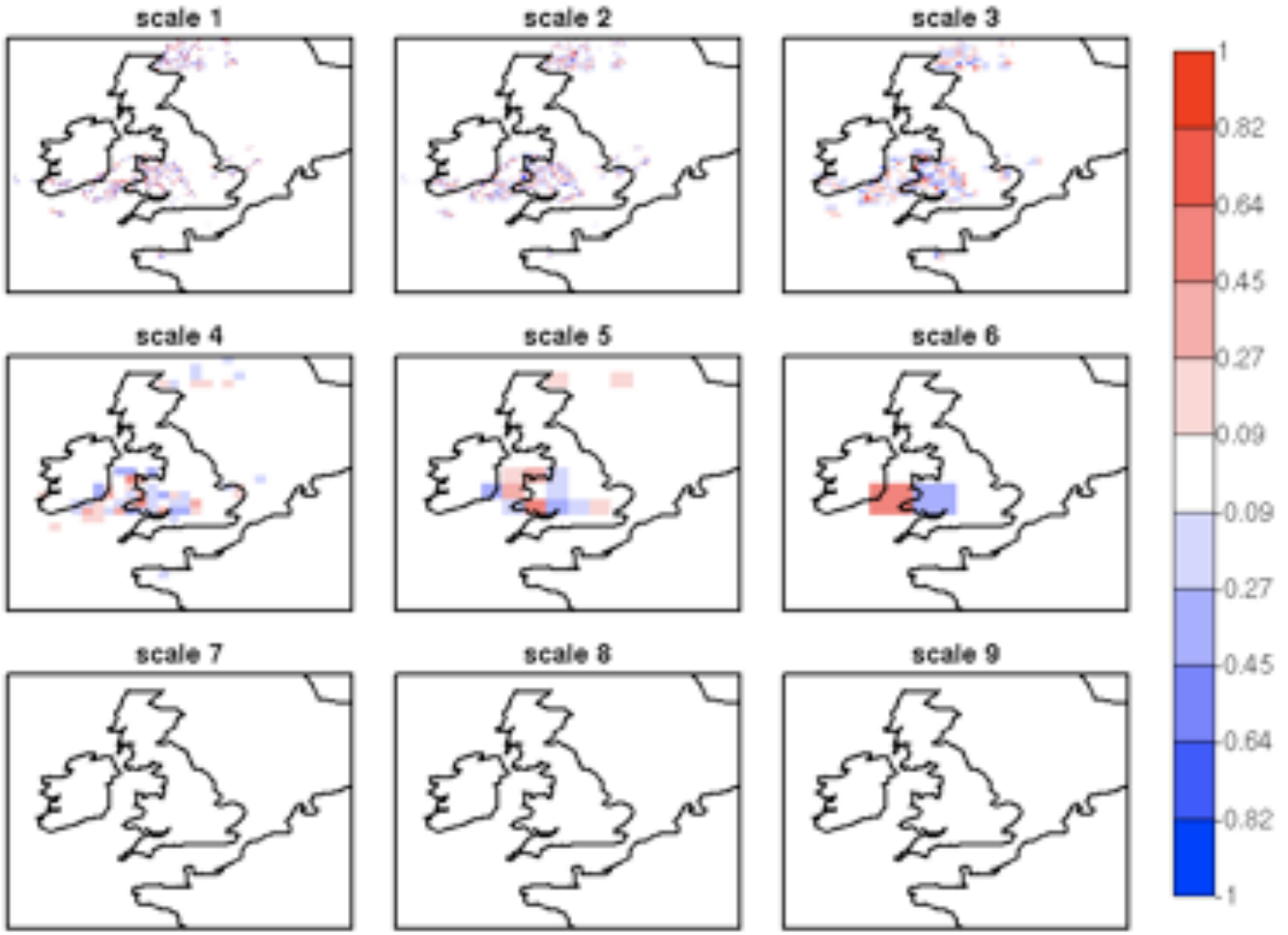
Binary Difference Field



Contingency Table Contributions

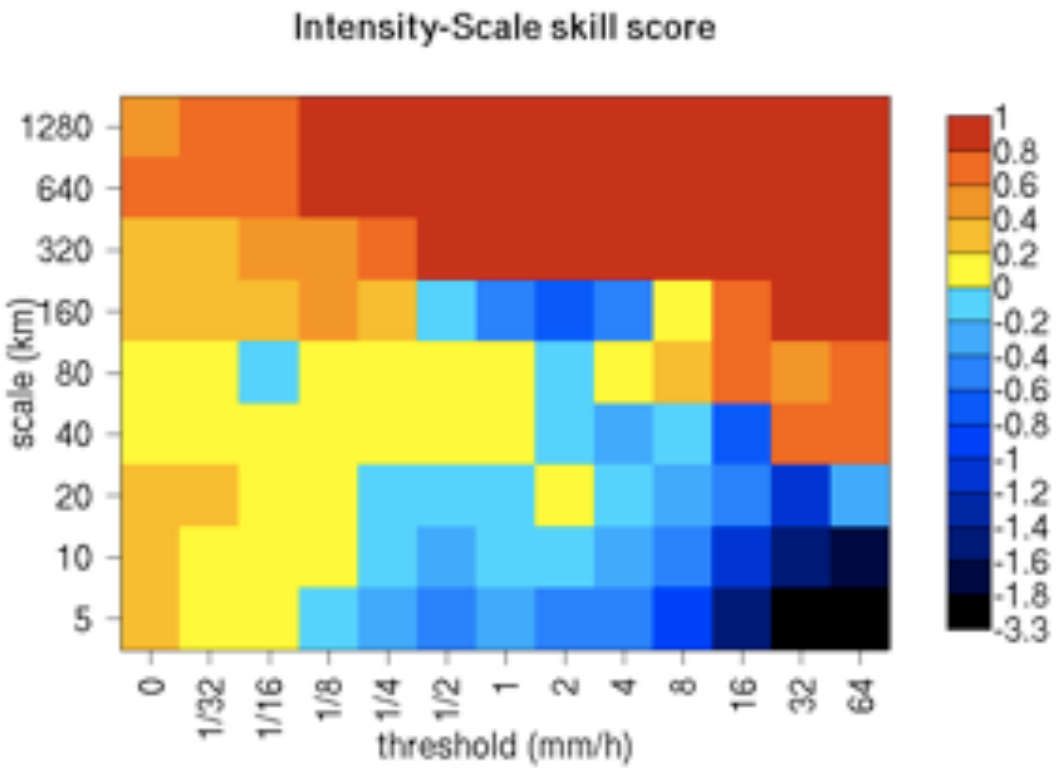


Casati *et al*.(2004)



Casati *et al*.(2004)

Intensity Skill Score (ISS) is the Heidke Skill Score (HSS), but calculated at each threshold and wave number.



HSS is an accuracy score corrected for random chance.

Thank you…Questions?

References

Casati B, G Ross, and DB Stephenson, 2004. A new intensity-scale approach for the verification of spatial precipitation forecasts. *Meteorol. Appl*. **11**:141--154.

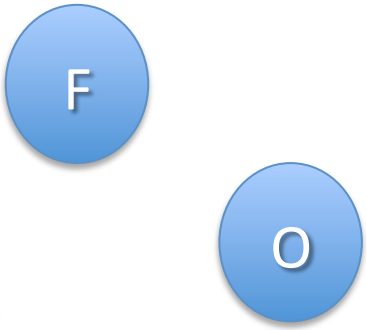
Ebert EE, 2008. Fuzzy verification of high resolution gridded forecasts: A review and proposed framework. *Meteorol. Appl*., **15**:51--64. DOI: 10.1002/met.25.

See also, http://www.ral.ucar.edu/projects/icp

Customizing and Understanding Wavelet Stats

Options for Handling Missing data

- Points with valid forecasts.
- Points with valid observations.
- Points with both valid forecasts and observations. $F \cap O$
- All points. $F \cup O$



Thresholds

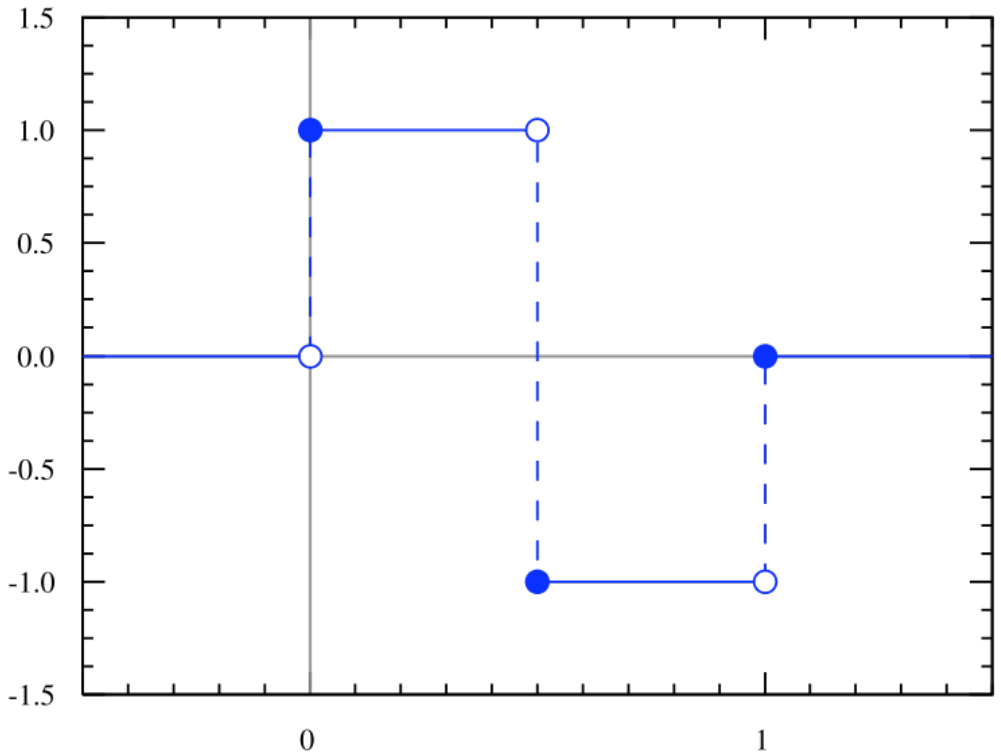
Forecast		Threshold	Event
0.05	<	0.2	0
0.17	<	0.2	0
0.45	>	0.2	1
2.15	>	0.2	1
0.05	<	1	0
0.17	<	1	0
0.45	<	1	0
2.15	>	1	1

Tiles – Grid must be 2^n x 2^n

- Cut down – user selected subset (square)
- Tiles – automated selection of subset(s)
- Pad with zeros – not recommended unless adds very small number of points.

Wavelets

- Haar
- Centered Haar
- Daubechies
- Centered Daubechies
- B spline
- Centered B spline

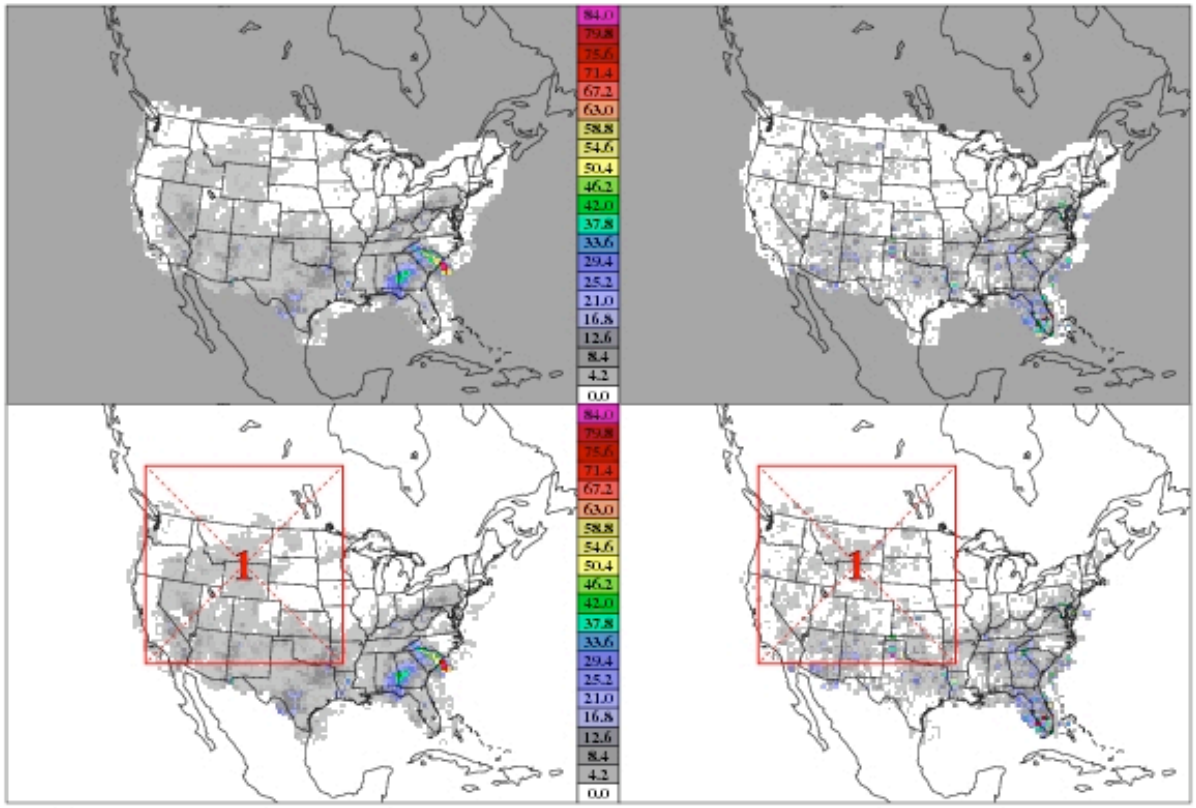


Output

- Text files
  - Options
  - Statistics
  - Can be read into stat analysis tool.
- Postscript files
  - Graphics
  - Options
  - Statistics

Wavelet-Stat: APCP at A24

Forecast | Observation



Model Name: WRF

Init Time: Aug 7, 2005 00:00:00
Valid Time: Aug 8, 2005 00:00:00
Lead Time: 24:00:00
Accum Time: 24:00:00

Tile Method: User-Defined
Tile Count: 1
Tile Dim: 64 x 64
Tile Corner: (45, 45)

Mask Missing: Fcst/Obs
Wavelet(k): Haar (2)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Binary

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.37311
Base Rate:	0.14233	Fcst Energy Squared (%):	0.22949 (100.00)
Mean-Squared Error (%):	0.19214 (100.00)	Obs Energy Squared (%):	0.14233 (100.00)

Overall forecast has skill (ISS > 0)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 1

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	-0.95852
Base Rate:	0.14233	Fcst Energy Squared (%):	0.04102 (17.87)
Mean-Squared Error (%):	0.08575 (44.63)	Obs Energy Squared (%):	0.05170 (36.32)

Errors at this scale account for nearly half of the MSE.

At this scale, forecast does not have skill (ISS < 0)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 2

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	-0.03746
Base Rate:	0.14233	Fcst Energy Squared (%):	0.03201 (13.95)
Mean-Squared Error (%):	0.04543 (23.64)	Obs Energy Squared (%):	0.02391 (16.80)

Forecast transitions from no skill to skill at scale 3.

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 3

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.44982
Base Rate:	0.14233	Fcst Energy Squared (%):	0.01656 (7.22)
Mean-Squared Error (%):	0.02409 (12.54)	Obs Energy Squared (%):	0.01278 (8.98)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 4

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.70019
Base Rate:	0.14233	Fcst Energy Squared (%):	0.02332 (10.16)
Mean-Squared Error (%):	0.01313 (6.83)	Obs Energy Squared (%):	0.00954 (6.70)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 5

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.87362
Base Rate:	0.14233	Fcst Energy Squared (%):	0.02745 (11.96)
Mean-Squared Error (%):	0.00553 (2.88)	Obs Energy Squared (%):	0.01600 (11.24)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 6

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.75764
Base Rate:	0.14233	Fcst Energy Squared (%):	0.03646 (15.89)
Mean-Squared Error (%):	0.01061 (5.52)	Obs Energy Squared (%):	0.00815 (5.72)

Wavelet-Stat: APCP at A24, Tile 1, >1.000, Scale 7

Difference (F-0)

Frequency Bias:	1.61235	Intensity Skill Score:	0.82650
Base Rate:	0.14233	Fcst Energy Squared (%):	0.05267 (22.95)
Mean-Squared Error (%):	0.00760 (3.95)	Obs Energy Squared (%):	0.02026 (14.23)

Summary

- Wavelet tool provides a flexible method for decomposing spatial fields into different scales.
- Once decomposed, verification measures at each *physical* scale can be examined and compared.

Tutorial Wrap Up

- Download MET at:

http://www.dtcenter.org/met/users/downloads/

- Email: met_help@ucar.edu with questions and suggestions.