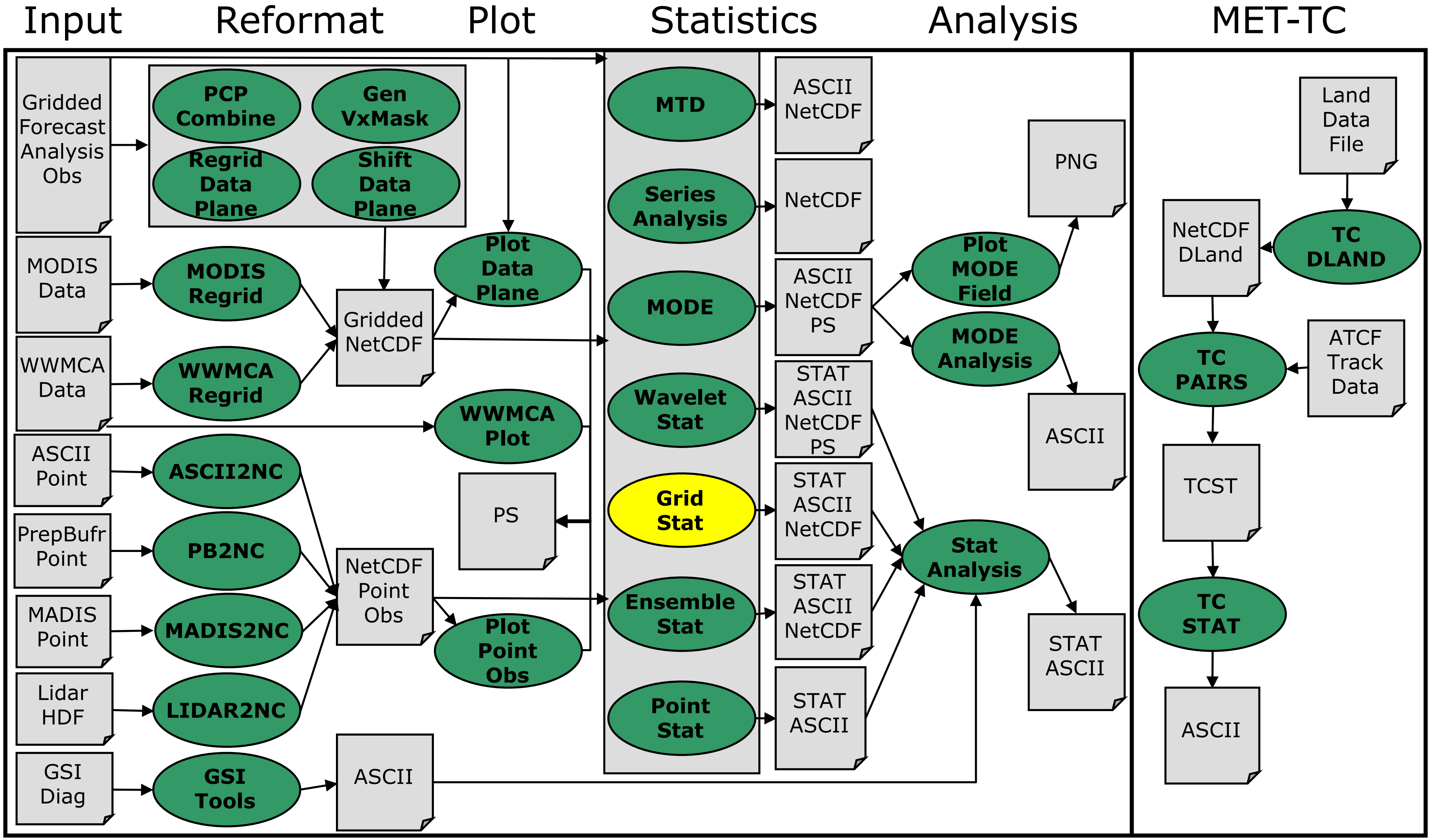
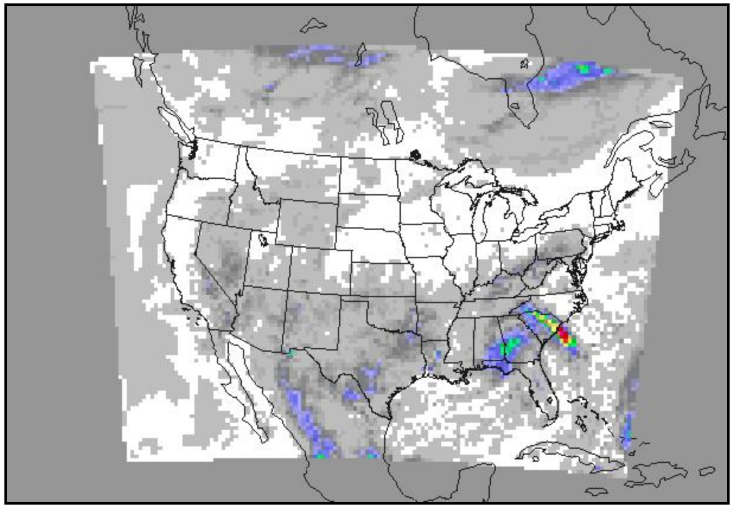
Grid-Stat Tool

Grid-Stat Tool

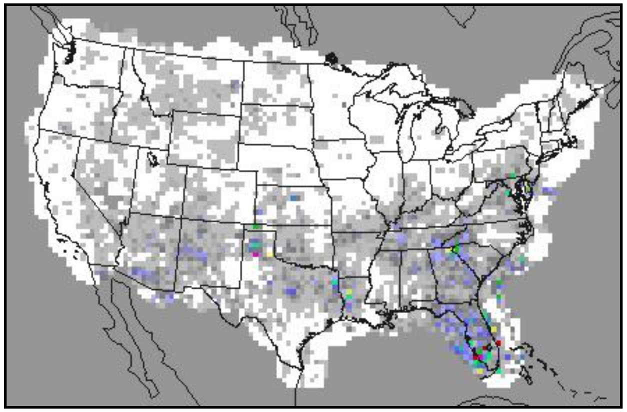


Grid-Stat: Overview



- Compare **gridded forecasts** to **gridded observations** on the same grid.
- 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.
  - Single and Multi-Category counts and statistics for thresholded fields.
  - Parametric and non-parametric confidence intervals for statistics.
  - Compute partial sums for raw fields.
  - Methods for probabilistic forecasts.
  - Economic Cost/Loss Value.
  - Neighborhood verification methods.
  - Fourier decomposition.

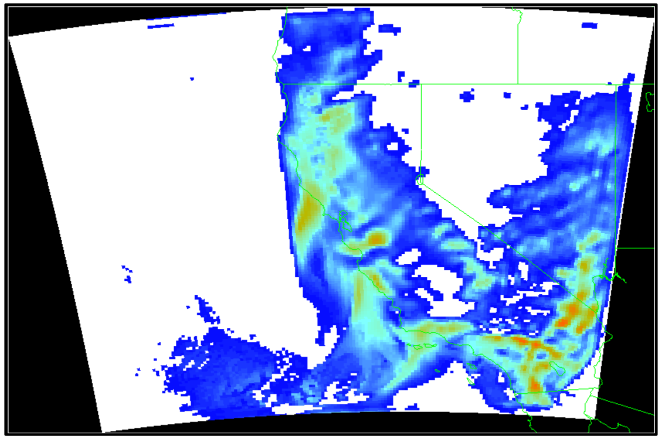


Grid-Stat: Input/Output

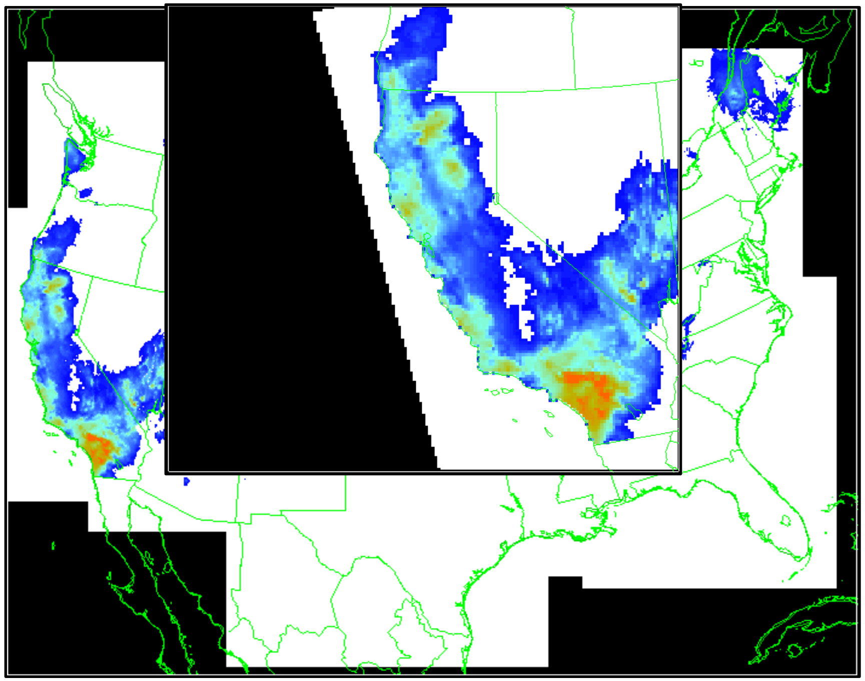
- Input Files
  - Gridded forecast and observation files
    - GRIB1 output of Unified Post-Processor (or other)
    - GRIB2 from NCEP (or other)
    - NetCDF from PCP-Combine, wrf_interp, or CF-compliant
  - ASCII configuration file
- Output Files
  - ASCII statistics file with all output lines (end with ".stat")
  - Optional ASCII files sorted by line type with a header row (ends with "_TYPE.txt")
  - Optional NetCDF matched pairs file

Grid-Stat: Common Grid

Model Forecast



StageIV Analysis

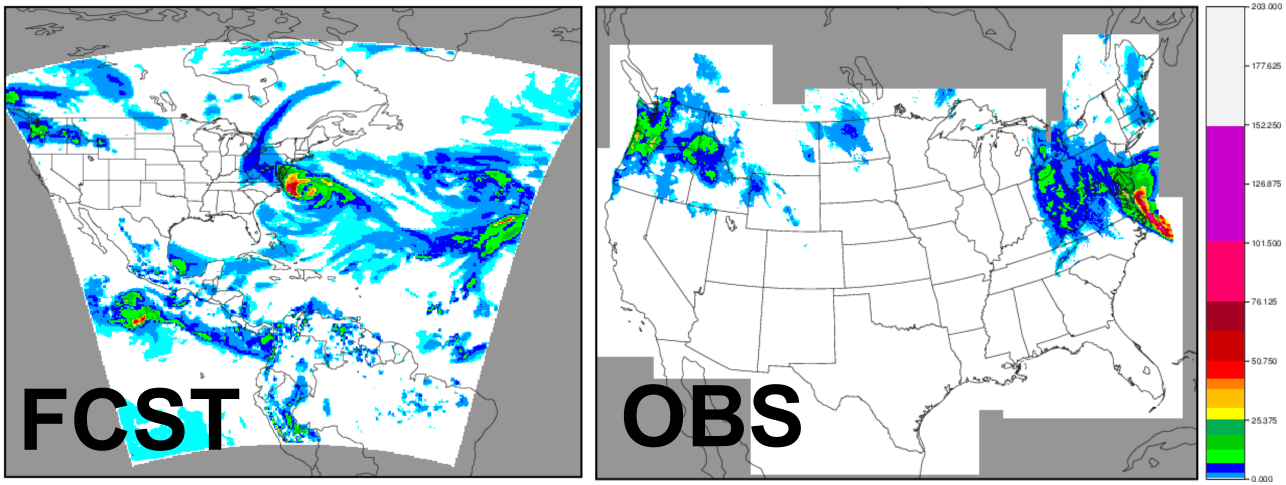


- Forecast and observations must be placed on a common grid.
- Regrid the StageIV Analysis (GRIB) to the model domain:

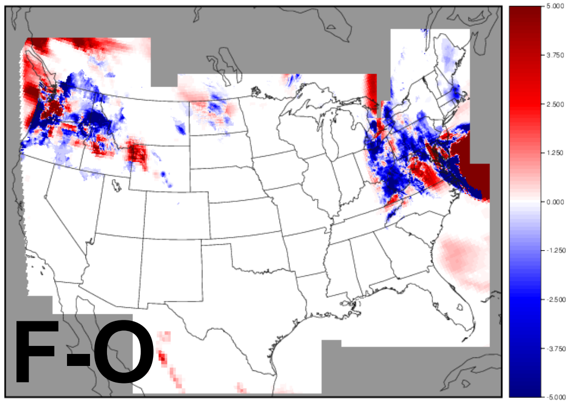
```
copygb –xg"255 5 169 154 31357 -129770 8 -120500 10395 10395 0 64" \
         ST4.2010122212.06h ST4.2010122212.06h_regrid
```

  - Practice running copygb in the practical session.
- Automated regridding in configuration file or use regrid_data_plane.

Grid-Stat: Automated Regridding

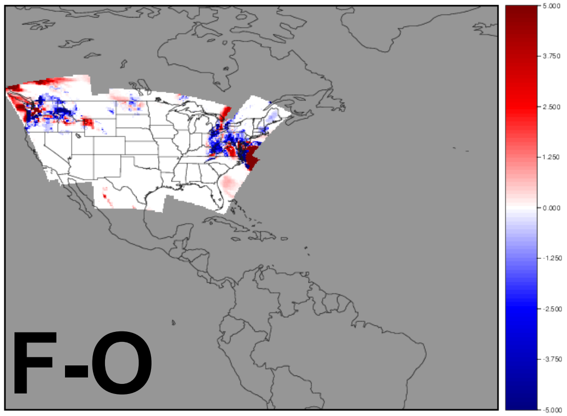


```
//
// Verification grid
//
regrid = {
   to_grid     = NONE;
   method      = BUDGET;
   width       = 2;
   vld_thresh  = 0.5;
   shape       = SQUARE;
}
```



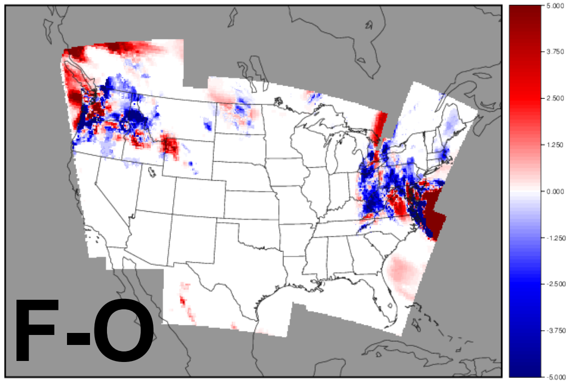
grid_stat_regrid_OBS_060000L_20121029_060000V_pairs.nc

`to_grid = FCST;`



grid_stat_regrid_FCST_060000L_20121029_060000V_pairs.nc

`to_grid = OBS;`

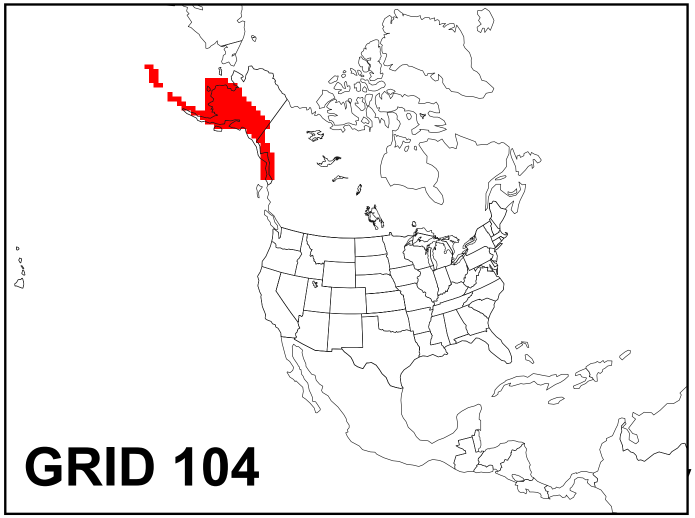


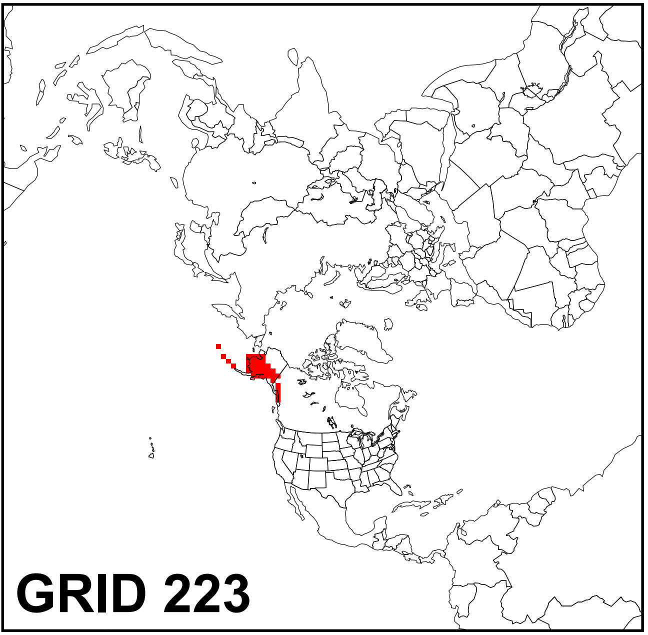
grid_stat_regrid_G130_060000L_20121029_060000V_pairs.nc

`to_grid = “G130”;`

Auto-Regridding of Masks

- 31 NCEP Verification Regions are defined for Grid 104.
  - CONUS, EAST, and WEST are groups of sub-regions.
- Added 34 NetCDF files to data/poly/NCEP_masks.
- **met-6.0:** Mask must be defined on verification domain:
  - ERROR : parse_poly_mask() -> The masking and verification grids do not match:
- **met-6.1:** Regrid mask (NEAREST neighbor) to verification domain:
  - DEBUG 2: Regridding mask grid to the verification grid using nearest neighbor interpolation:
- Most similar to VSDB logic.





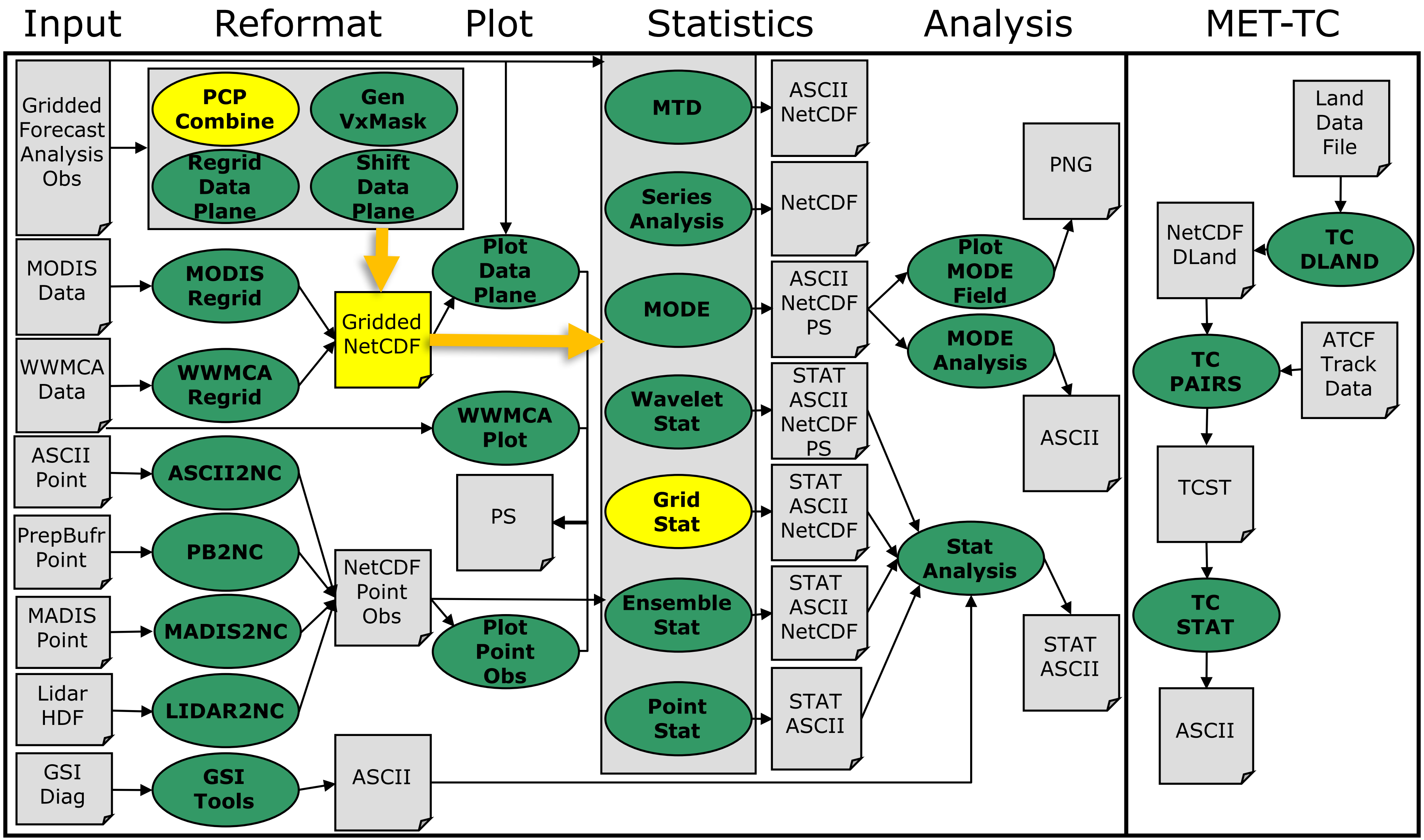
Grid-Stat: Usage

Usage: grid_stat

```
fcst_file
obs_file
config_file
[-outdir path]
[-log file]
[-v level]
[-compress level]
```

fcst_file	Gridded forecast file
obs_file	Gridded observation file
config_file	ASCII configuration file
-outdir	Output directory to be used
-log	Optional log file
-v	Level of logging

Grid-Stat: Example Data Flow



Grid-Stat: Configuration

- Many configurable parameters – only set a few:
  - Precipitation accumulated over 24 hours.
    - GRIB1 forecast
    - NetCDF observation
  - Threshold any rain and moderate rain (mm).
  - Accumulate stats over all the points in the domain and just the eastern United States.
  - Compute neighborhood statistics with two sizes.
  - Generate continuous, categorical, and neighborhood line types.

```
fcst = {
 field = [
  {
   name       = "APCP";
   level      = [ "A24" ];
   cat_thresh = [ >0.0, >20.0 ];
  }
 ];
}
```

```
obs = {
 field = [
  {
   name       = "APCP_24";
   level      = [ "(*,*)" ];
   cat_thresh = [ >0.0, >20.0 ];
  }
 ];
}
```

```
mask = {
 grid = [ "FULL" ];
 poly = [ "EAST.poly" ];
}
```

```
nbrhd = {
 vld_thresh = 1.0;
 width      = [ 3, 5 ];
 cov_thresh = [ >=0.5 ];
}
```

```
output_flag = {
 fho    = BOTH;
 ctc    = BOTH;
 cts    = BOTH;
 mctc   = BOTH;
 mcts   = BOTH;
 cnt    = BOTH;
 sl1l2  = BOTH;
 sal1l2 = NONE;
 vl1l2  = NONE;
 val1l2 = NONE;
 pct    = NONE;
 pstd   = NONE;
 pjc    = NONE;
 prc    = NONE;
 eclv   = NONE;
 nbrctc = BOTH;
 nbrcts = BOTH;
 nbrcnt = BOTH;
 grad   = NONE;
}
```

Grid-Stat: Field name and level

- GRIB1 and GRIB2 files
  - **name = “GRIB Abbreviation”;**
    - http://www.nco.ncep.noaa.gov/pmb/docs/on388/table2.html
    - TMP for Temperature, APCP for accumulated precipitation.
  - **level = [“string”];** Multiple values expand to multiple vx tasks
    - Level indicator followed by level value.
      - **A** for accumulation interval in HH[MMSS] format *(A06*).
      - **P** for pressure level (*P500*) or layer (*P500-600*).
      - **Z** for vertical level (*Z2* or *Z10*).
      - **L** for generic level type (*L100*).
      - **R** for a specific GRIB record number (*R225*).
- Gridded NetCDF files
  - **name = “string”;** Defines NetCDF variable name.
  - **level = [“string”];** Defines index into dimensions.
  - For APCP_06(lat,lon) from PCP-Combine output
    - *name = “APCP_06”; level = [“(*,*)”];*
  - For TT(Time, num_metgrid_levels, south_north, west_east) from p_interp
    - *name = “TT”; level = [“(0,0,*,*)”, “(0,1,*,*)”, “(0,2,*,*)”];*

Grid-Stat: Config File Defaults

- MET Statistics tools parse up to 4 configuration files:
  1. **MET_BASE/config/ConfigConstants** defines configuration file constants (e.g. NONE, STAT, BOTH) and should not be modified.
  2. **MET_BASE/config/ConfigMapData** defines default map data for all plots (map data files, line colors, widths, and types for Plot-Point-Obs, Plot-Data-Plane, Wavelet-Stat, and MODE).
  3. **MET_BASE/config/GridStatConfig_default** defines default settings for the specific tool.
  4. **User-specific configuration** file passed on the command line override default settings.

NOTE: **MET_BASE/config/README** describes config file options.

NOTE: When running a shared installation of MET, override default settings in the **user-specific configuration** file rather than modifying the system-wide defaults.

Grid-Stat: Run

- met-6.1/bin/grid_stat \
  sample_fcst.grb sample_obs.nc \
  GridStatConfig_APCP24 -outdir out -v 2

```
DEBUG 1: Default Config File: met-6.1/share/met/data/config/GridStatConfig_default
DEBUG 1: User Config File: GridStatConfig_APCP24
DEBUG 1: Forecast File: sample_fcst.grb
DEBUG 1: Observation File: sample_obs.nc
DEBUG 2: --------------------------------------------------------------------------------
DEBUG 2: Processing APCP/A24 versus APCP_A24, for interpolation method UW_MEAN(1), over region FULL, using 6412 pairs
DEBUG 2: Computing Categorical Statistics.
DEBUG 2: Computing Multi-Category Statistics.
DEBUG 2: Computing Continuous Statistics.
DEBUG 2: Processing APCP/A24 versus APCPA24, for interpolation method UW_MEAN(1), over region EAST, using 2582 pairs.
DEBUG 2: Processing APCP/A24 versus APCPA24, for interpolation method NBRHD(9), raw thresholds of >0.000 and >0.000,
over region EAST, using 5829 pairs.
DEBUG 2: Computing Neighborhood Categorical Statistics.
DEBUG 2: Computing Neighborhood Continuous Statistics.
… MORE NEIGHBORHOOD VERIFICATION TASKS LISTED …
DEBUG 2: --------------------------------------------------------------------------------
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V.stat
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_fho.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_ctc.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_cts.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_mctc.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_mcts.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_cnt.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_sl1l2.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_nbrctc.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_nbrcts.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_nbrcnt.txt
DEBUG 1: Output file: out/grid_stat_240000L_20050808_000000V_pairs.nc
```

Grid-Stat: ASCII Output Types

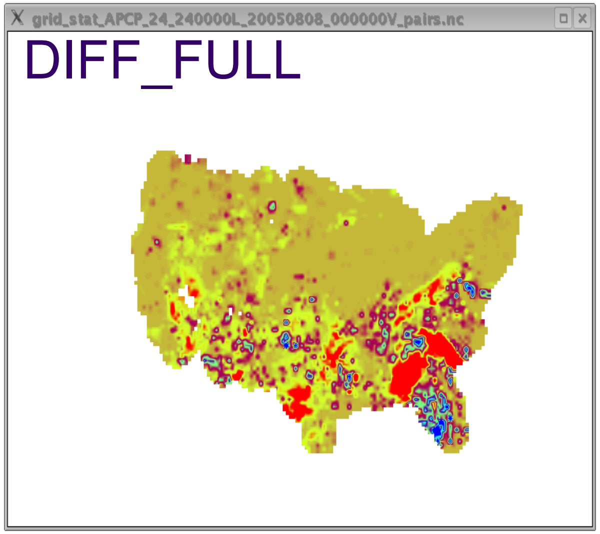
- Statistics line types: 19 possible
  - Same as Point-Stat (16)
    - FHO, CTC, CTS, MCTC, MCTS, CNT
    - SL1L2, SAL1L2, VL1L2 , VAL1L2
    - PCT, PSTD, PJC, PRC
    - ECLV
  - Neighborhood – apply threshold, define neighborhood
    - Neighborhood continuous statistics (NBRCNT)
    - Neighborhood contingency table counts (NBRCTC)
    - Neighborhood contingency table statistics (NBRCTS)
  - Gradient line type (GRAD)
- 22 header columns common to all line types
- Remaining columns specific to each line type

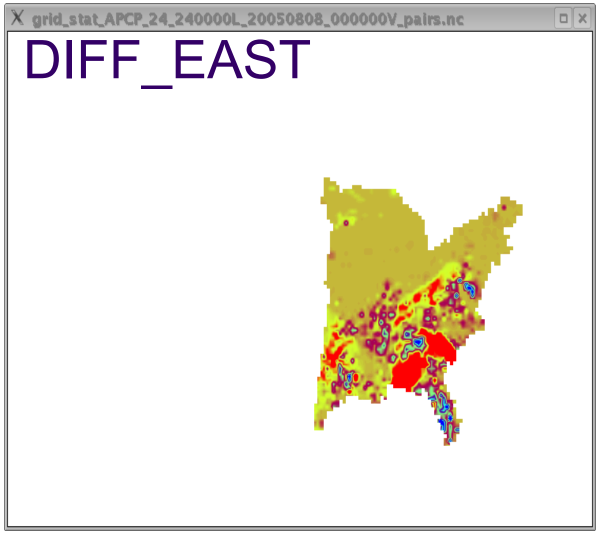
Grid-Stat: Sample Output

1. STAT file output for sample run:
   - 2 lines each for CNT, MCTC, MCTS, 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
2. Additional TXT files for each line type
3. NetCDF file containing matched pairs

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 fields:
  - FCST, OBS, and DIFF for FULL and EAST





```
// NetCDF matched
// pairs output file
nc_pairs_flag = {
   latlon     = TRUE;
   raw        = TRUE;
   diff       = TRUE;
   climo      = TRUE;
   weight     = FALSE;
   nbrhd      = FALSE;
   fourier    = FALSE;
   gradient   = FALSE;
   apply_mask = TRUE;
}
```

- Set **`apply_mask = FALSE;`** for only FULL domain.

Grid-Stat: CTC Output Line

VERSION	V6.1
MODEL	WRF
DESC	NA
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_24
FCST_LEV	A24
OBS_VAR	APCP_24
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	2582
FY_OY (hits)	5
FY_ON (f.a.)	104
FN_OY (miss)	70
FN_ON (c.n.)	2403

FHO vs CTC Line Types

- Grid-Stat, Point-Stat, and Stat-Analysis can output FHO and CTC line types.
  - Values are equivalent (*and redundant*).
  - CTC has integer counts for 4 cells of 2x2 table.
  - FHO has floating point rates.
  - FHO rounding issues for rare events.

LINE_TYPE	CTC
TOTAL	2582
FY_OY (hits)	5
FY_ON (false alarms)	104
FN_OY (misses)	70
FN_ON (correct neg)	2403

LINE_TYPE		FHO
TOTAL		2582
F_RATE (fcst rate)	**(5+104)/2582**	0.042215
H_RATE (hit rate)	**5/2582**	0.0019365
O_RATE (obs rate)	**(5+70)/2582**	0.029047

Comparing Different Fields

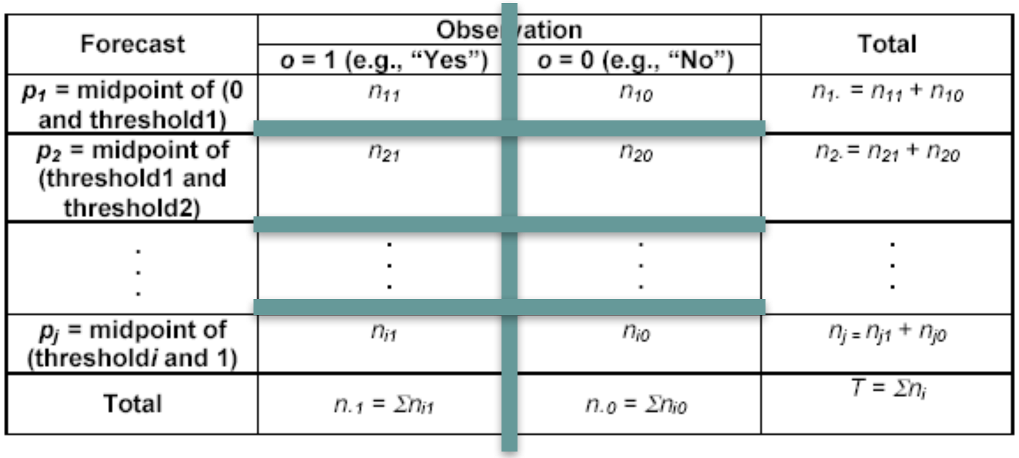
- Grid-Stat, Point-Stat, and all STAT tools may be used to compare different variables.
  - User must interpret results.
  - Example: Convective Precip vs. Total Precip
  - Configuration file settings:
    - Selecting variable/levels

```
fcst = {
 field = [
  {
   name       = "ACPCP";
   level      = [ "A24" ];
   cat_thresh = [ >0.0 ];
  }
 ];
};
```

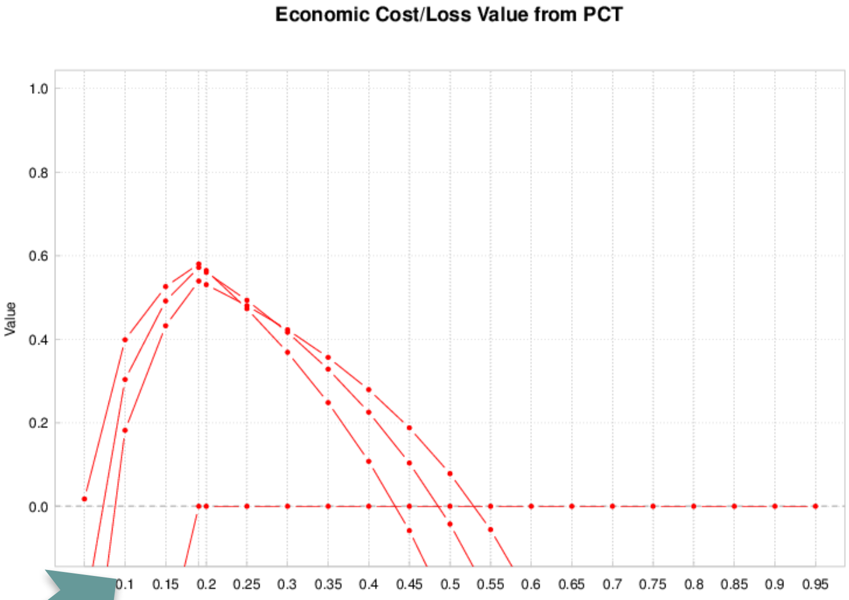
```
obs = {
 field = [
  {
   name       = "APCP";
   level      = [ "A24" ];
   cat_thresh = [ >0.0 ];
  }
 ];
};
```

Economic Cost/Loss Value

- Grid-Stat, Point-Stat, and Stat-Analysis can output the ECLV line type.
- Equivalent to the VSDB ECON line type, except…
  - ECON is only generated when evaluating ensemble probabilities.
  - ECLV from 2x2 CTC contingency table yields a single curve.
  - ECLV from Nx2 PCT probabilistic contingency table yields N curves.
- One ECLV line equals one curve on the plot.
  - Undefined at 0 and 1.
  - Maximized for the base rate.



Forecast	Observation		Total
	o = 1 (e.g., "Yes")	o = 0 (e.g., "No")	
p_1 = midpoint of (0 and threshold1)	n_{11}	n_{10}	$n_{1.} = n_{11} + n_{10}$
p_2 = midpoint of (threshold1 and threshold2)	n_{21}	n_{20}	$n_{2.} = n_{21} + n_{20}$
⋮	⋮	⋮	⋮
p_j = midpoint of (threshold*i* and 1)	n_{i1}	n_{i0}	$n_j = n_{j1} + n_{j0}$
Total	$n_{.1} = \Sigma n_{i1}$	$n_{.0} = \Sigma n_{i0}$	$T = \Sigma n_i$

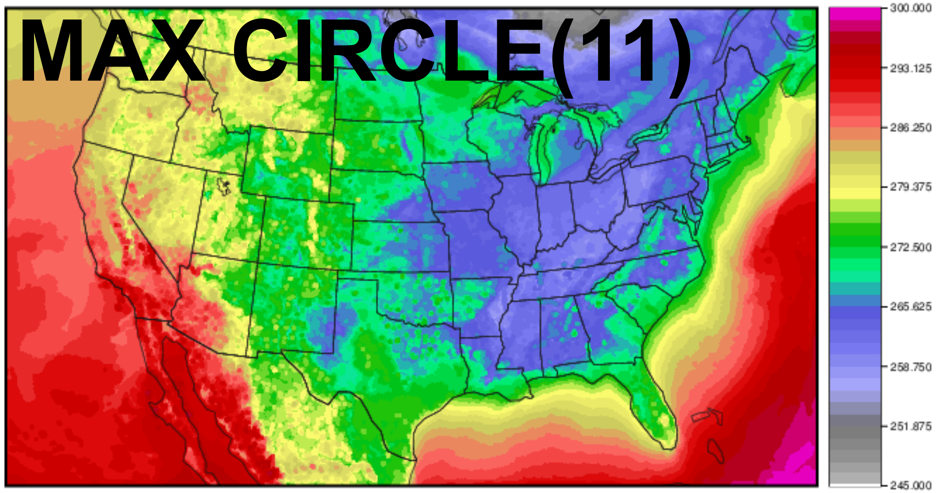


```
eclv_points = 0.05; // 0 to 1 every 0.05
eclv_points = [ 0.1, 0.2, 0.3, 0.4, 0.5, 0.65, 0.8, 0.95 ]; // non-equal
```

Grid-Stat: Data Smoothing

- Higher resolution forecasts typically score worse than lower resolution ones for traditional scores, like RMSE.
- Specify **interp** section to apply smoothing method(s) prior to computing statistics.
- Smoothing methods indicated in **INTERP_MTHD** and **INTERP_PNTS** columns.

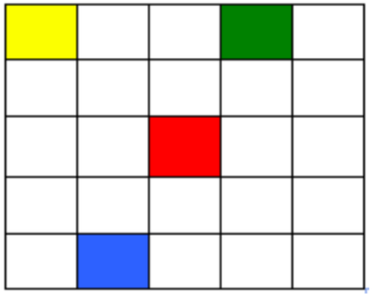
```
//
// Data smoothing methods
//
interp = {
  // FCST, OBS, or BOTH
  field      = BOTH;
  vld_thresh = 1.0;
  shape      = SQUARE;
  type = [
    // Default, no smoothing
    { method = NEAREST;
      width  = 1; },
    // Mean of 11x11 square
    { method = UW_MEAN;
      width  = 11; },
    // Max of circle diam 11
    { method = MAX;
      width  = 11;
      shape  = CIRCLE; }
  ];
}
```



hrrr_TMP_Z2_max_121.nc

Grid-Stat: Neighborhoods

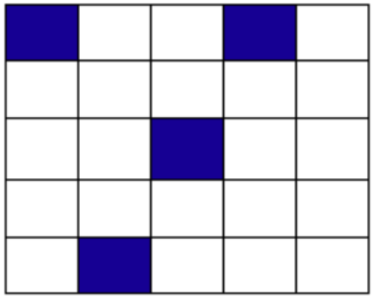
- As with all neighborhood methods (i.e. HiRA), allows for some spatial / temporal uncertainty in either model or observation by giving credit for being ‘close’.
- Apply categorical threshold and neighborhood width to convert gridded forecast and observation fields into fractional coverage fields.
- Select SQUARE or CIRCLE shape.
- Every permutation of **cat_thresh** and **nbrhd.width**.
  - **NBRCNT** statistics (**FBS**, **FSS**) computed directly from fractional coverage fields.
  - Apply **cov_thresh** thresholds to fractional coverage fields to compute **NBRCTC** and **NBRCTS** counts and statistics (i.e. like **CTC** and **CTS**).



Model Forecast

White boxes = 0

Colored boxes > 0



Threshold Forecast

Blue boxes = event

Fractional Proportion

- 1x1 Neighborhood: 1/1
- 3x3 Neighborhood: 1/9
- 5x5 Neighborhood: 4/25

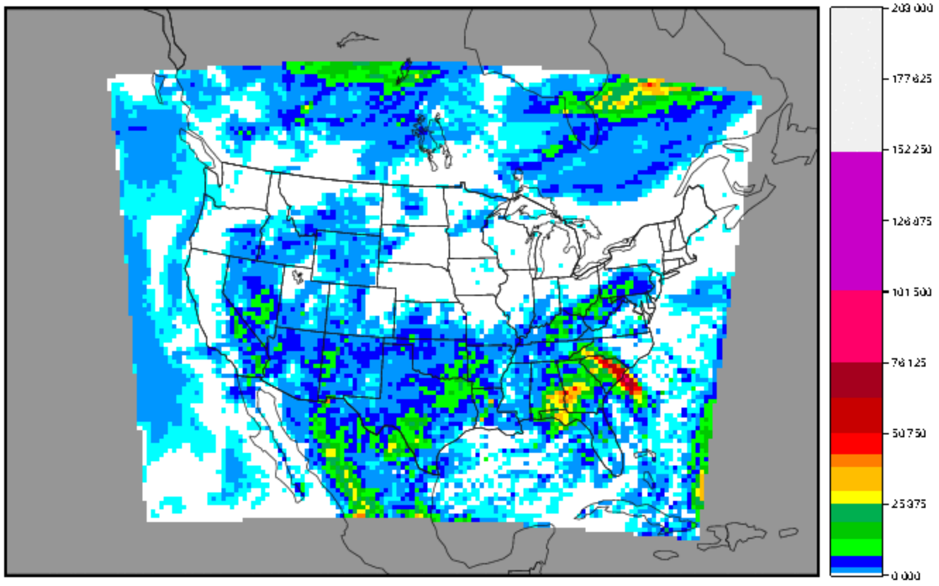
```
cat_thresh = [ >0.0, >=6.35 ];
...
nbrhd = {
   width      = [ 1, 3, 5, 7 ];
   cov_thresh = [ >=0.5 ];
   vld_thresh = 1.0;
   shape      = SQUARE;
}
```

Grid-Stat: Neighborhoods

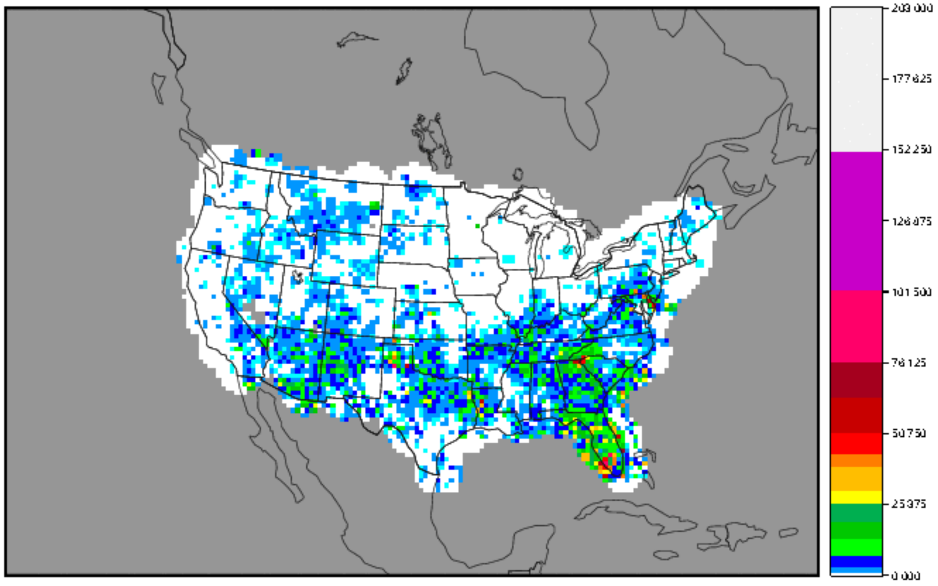
- Threshold 24-APCP >= 0.1”
- Edge effects as width increases.
- FSS increases as width increases:
  - 0.64287, 0.73593, 0.80247
  - 0.85106, 0.89191, 0.91487
  - 0.92632, 0.93536, 0.94517

```
cat_thresh = [ >=2.54 ];

nbrhd = {
   width = [ 3,  5,  7,  9, 11,
            13, 15, 17, 19 ];
}
nc_pairs_flag = {
   nbrhd = TRUE;
}
```



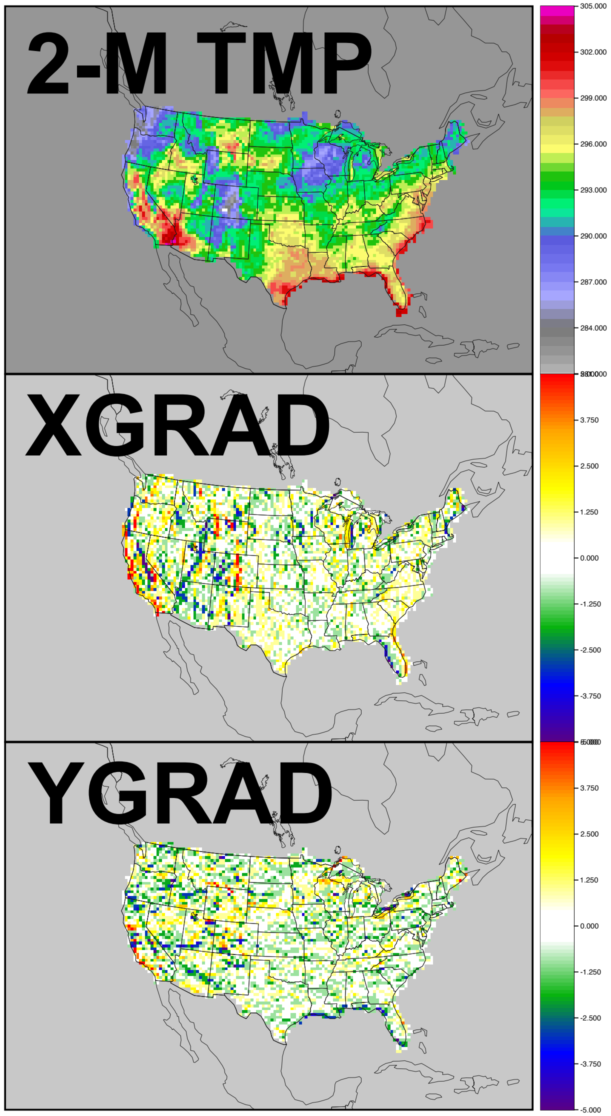
grid_stat_APCP_24_240000L_20050808_000000V_pairs.nc



grid_stat_APCP_24_240000L_20050808_000000V_pairs.nc

Grid-Stat: Gradients

- GRAD line type contains the S1 score and its components.
  - WMO-mandated statistic from 1954.
  - Computed over the gradients of forecast and observation fields computed in the X and Y grid direction.
  - Adapted from VSDB code:
    - FGBAR: mean forecast gradient
    - OGBAR: observed gradient
    - MGBAR: mean of maximum gradient
    - EGBAR: mean of gradient differences
    - S1 = 100 * EGBAR / MGBAR
    - S1_OG = 100 * EGBAR / OGBAR
    - FGOG_RATIO = FGBAR / OGBAR



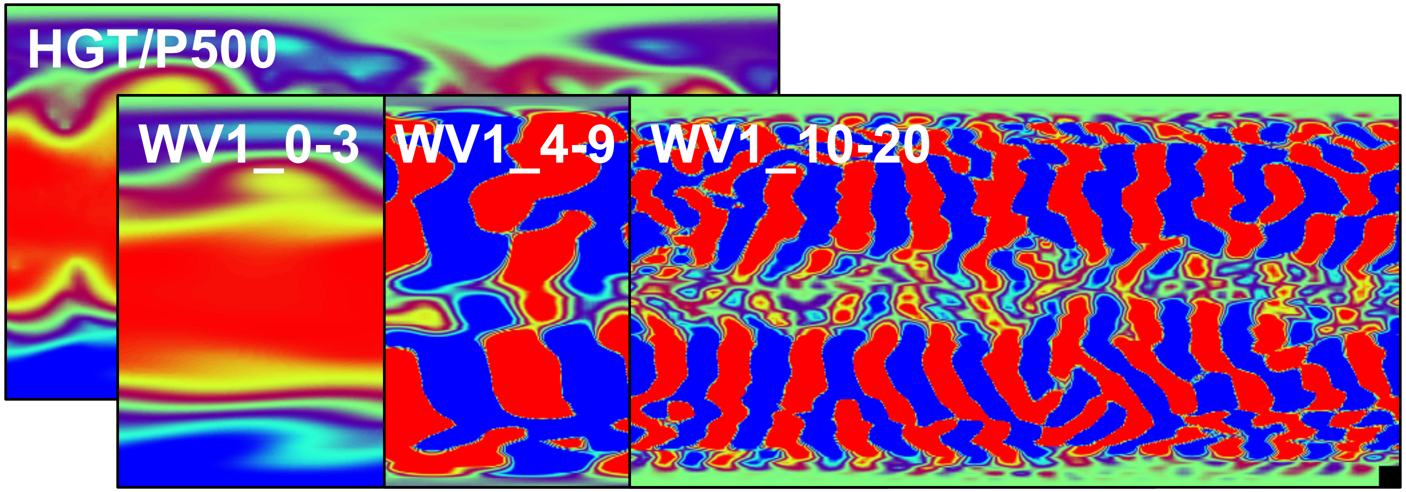
grid_stat_120000L_20050807_120000V_pairs.nc

Grid-Stat: Fourier Decomposition

- Added support to Grid-Stat for 1-Dimensional Fourier decompositions.
- Affects output for CNT, SL1L2, SAL1L2, VL1L2, and VAL1L2 line types.
- Configuration file option to specify the waves:

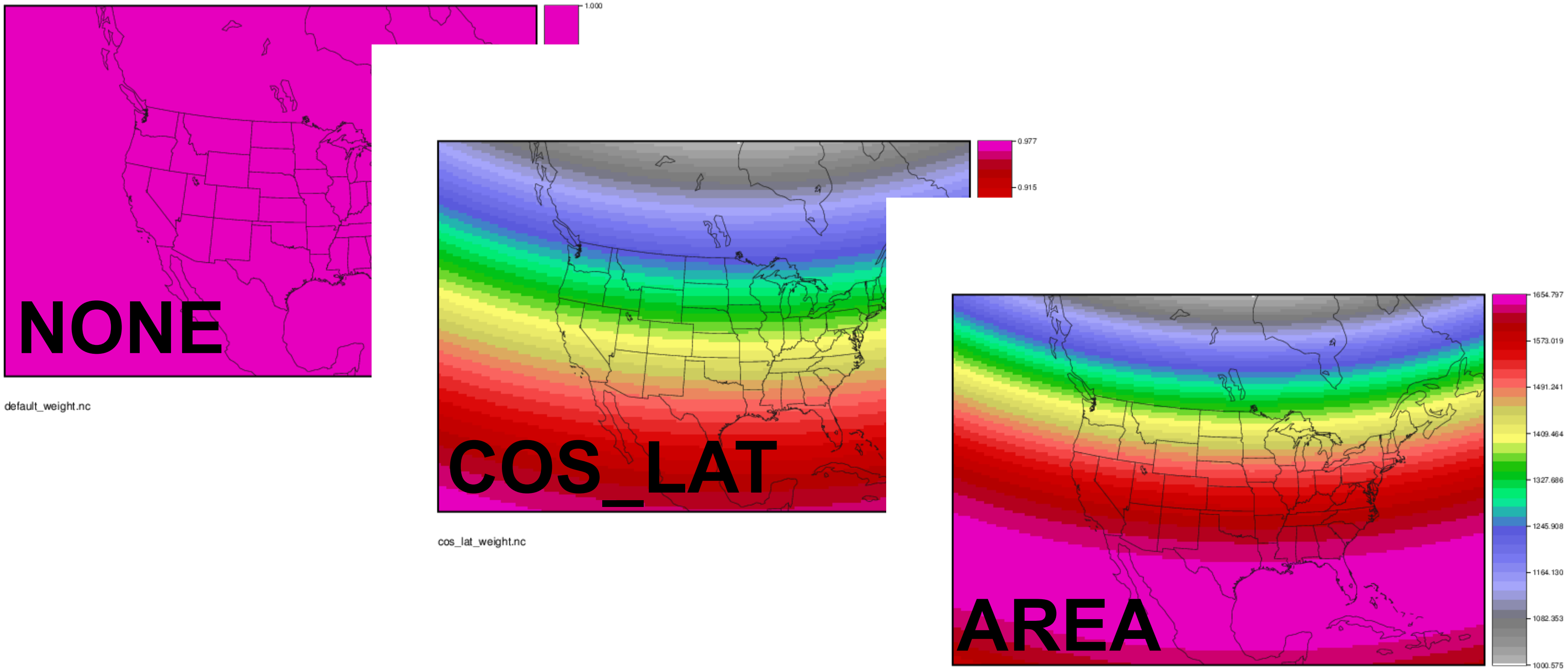
```
fourier = {
  wave_1d_beg = [  0, 0, 4, 10 ];
  wave_1d_end = [ 72, 3, 9, 20 ];
}
```

- Wave numbers indicated in the **INTERP_MTHD** column:
  - **WV1_0-72**
  - **WV1_0-3**
  - **WV1_4-9**
  - **WV1_10-20**



Grid-Stat: Grid Weighting

```
// The "grid_weight_flag" specifies how grid weighting should be applied…
//    - "NONE" to disable grid weighting using a constant weight (default).
//    - "COS_LAT" to define the weight as the cosine of the grid point latitude.
//      This an approximation for grid box area used by NCEP and WMO.
//    - "AREA" to define the weight as the true area of the grid box (km^2).
grid_weight_flag = NONE;
```



default_weight.nc

cos_lat_weight.nc

true_area_weight.nc

Config: Converting Data

- Enhance config file language to support functions of 1 variables.
- Use convert(x) function to define unit conversions:

```
   convert(x) = log10(x);
   convert(x) = sqrt(x);
```

- Common conversion functions pre-defined in share/met/config/ConfigConstants:

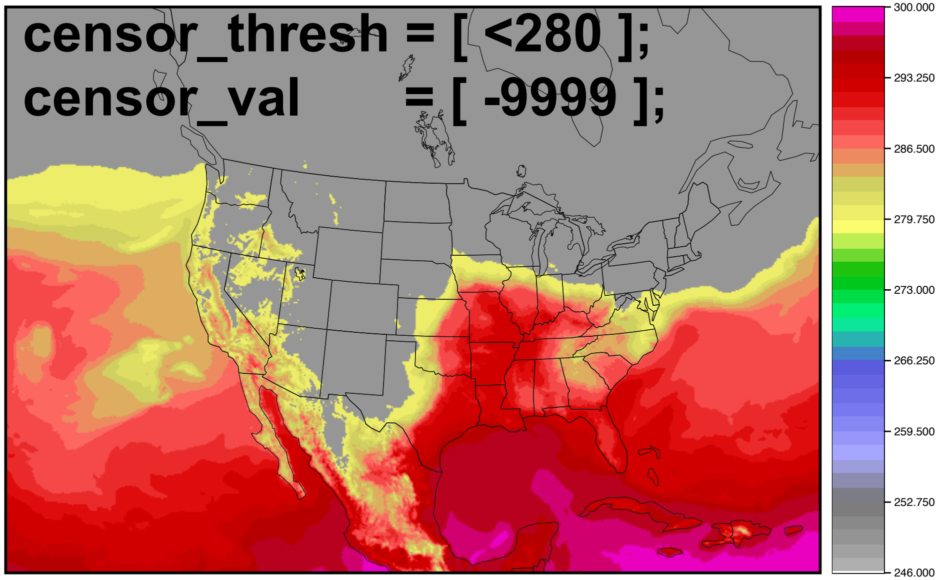
```
   K_to_C(t) = t - 273.15;
   C_to_K(t) = t + 273.15;
   C_to_F(t) = 1.8 * t + 32.0;
```

Config: Censoring Data

- Applying MET to wider range of data types reveals need for pre-processing.
- Censor logic is applied to raw data before any regridding is done.
- May be specified separately for each forecast and/or observation field:
  - The **censor_thresh** entry is an array of thresholds.
  - The **censor_val** entry is an array of replacement values.
- Reflectivity Example:
  - Forecast reflectivity values are 0 or >= 35 dBZ.
  - Observed reflectivity values are continuous less than 35 dBZ.
  - Define observation censor:

```
censor_thresh = [ <35 ];
censor_val    = [ 0 ];
```

- Used to make the forecast and observation data more comparable.
- Can be used for range checking data.



```
censor_thresh = [ <280 ];
censor_val     = [ -9999 ];
```

Config: Climatology Data

- Required for anomaly correlation (ANOM_CORR)
- NCEP monthly 2.5 degree
  - *match_day = FALSE*
- NCEP daily 1.0 degree
  - *match_day = TRUE*
- ERA daily 1.5 degree
  - *match_day = TRUE*
- Any other reference forecast.

```
climo_mean   = {

   file_name = [
    // List of file names
   ];
   field     = [
    // Same length as fcst.field
   ];
   regrid    = {
     method     = NEAREST;
     width      = 1;
     vld_thresh = 0.5;
   }
   time_interp_method = DW_MEAN;
   match_day          = FALSE;
   time_step          = 21600;
}
```

Config: Binned Climatologies

- Enhance Grid-Stat and Point-Stat to process climatological distributions (i.e. climo mean and standard deviation).
- Binned climatologies affect only the computation of probabilistic statistics.
- Config file options:

```
climo_mean     = { … };   // Climo Mean Fields
climo_stdev    = { … };   // Climo Standard Deviation
climo_cdf_bins = 10;      // Number of Climo Bins
                          // Or array of bin values
```

- For each observation value, use the climo mean and standard deviation and compute a CDF value between 0 and 1.
- Place that observation into the correct climo CDF bin.
- Compute stats for all pairs within each bin.
- When climo mean and standard deviation are provided, derive the climatological probability values when computing Brier Skill Score.
- Bin number appended to the **VX_MASK** output column:
  - **FULL_BIN1, FULL_BIN2, …, FULL_BIN10**

Verifying Winds

- Verify u, v, and speed, but not wind direction.
- Incremental support for wind direction:
  - Verify vector winds in Point-Stat and Grid-Stat:
    - Add wind speed thresholds to determine which u/v pairs are included in the vector partial sums (VL1L2).
  - Aggregate winds in Stat-Analysis:
    - Aggregate vector partial sum (VL1L2) or (***new in V5.0***) u/v pairs (MPR) and compute statistics for the wind direction errors.
      - Mean forecast and observation wind directions, mean error (F-O), and mean absolute error

Wind Direction: Example

Point-Stat: VL1L2 Lines

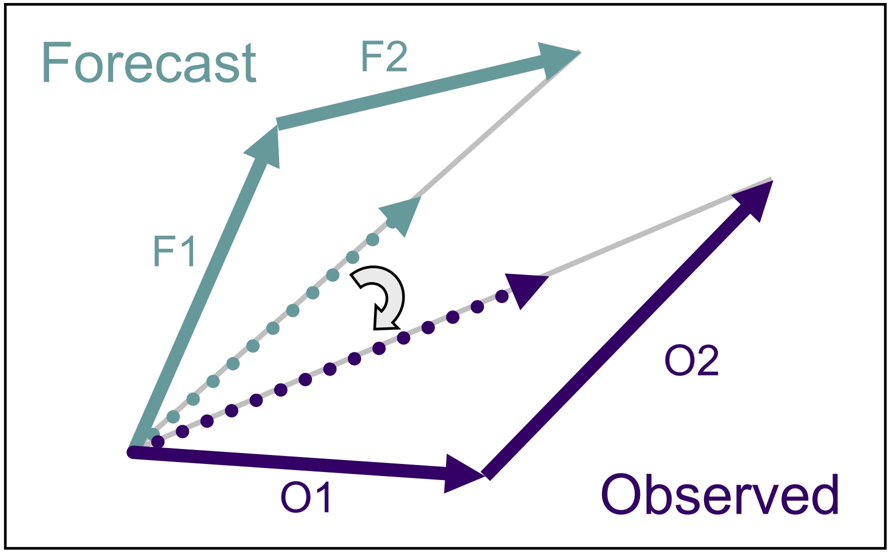
```
VX_MASK THRESH   LINE_TYPE TOTAL UFBAR    VFBAR    UOBAR    VOBAR    UVFOBAR  UVFFBAR  UVOOBAR
DTC_165 >=1.000 VL1L2     653   1.91117  0.07900  1.40658  -0.06126 13.01039 18.12575 20.31649
DTC_165 >=3.000 VL1L2     279   3.13561  -0.35096 2.87061  -0.30072 26.50472 30.03257 38.25362
DTC_165 >=5.000 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

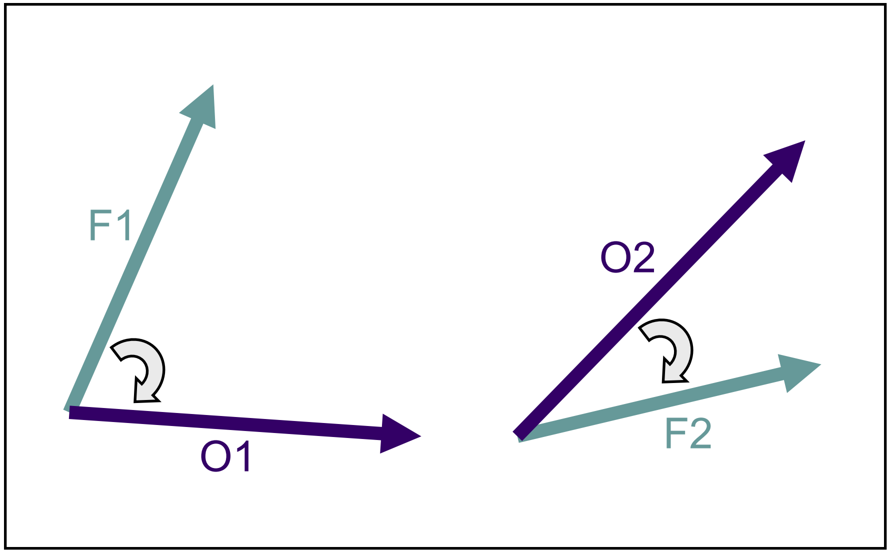
```
JOB_LIST:      -job aggregate_stat -fcst_thresh >=1.000 -line_type VL1L2 -out_line_type WDIR
     COL_NAME: TOTAL FBAR      OBAR     ME        MAE
ROW_MEAN_WDIR: 2     183.25038 0.22749  -3.02289  7.88372
    AGGR_WDIR: 3084  103.87238 85.96574 -17.90663 NA
------------------------------------------------------------------------------------------------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

Wind Direction: Output

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

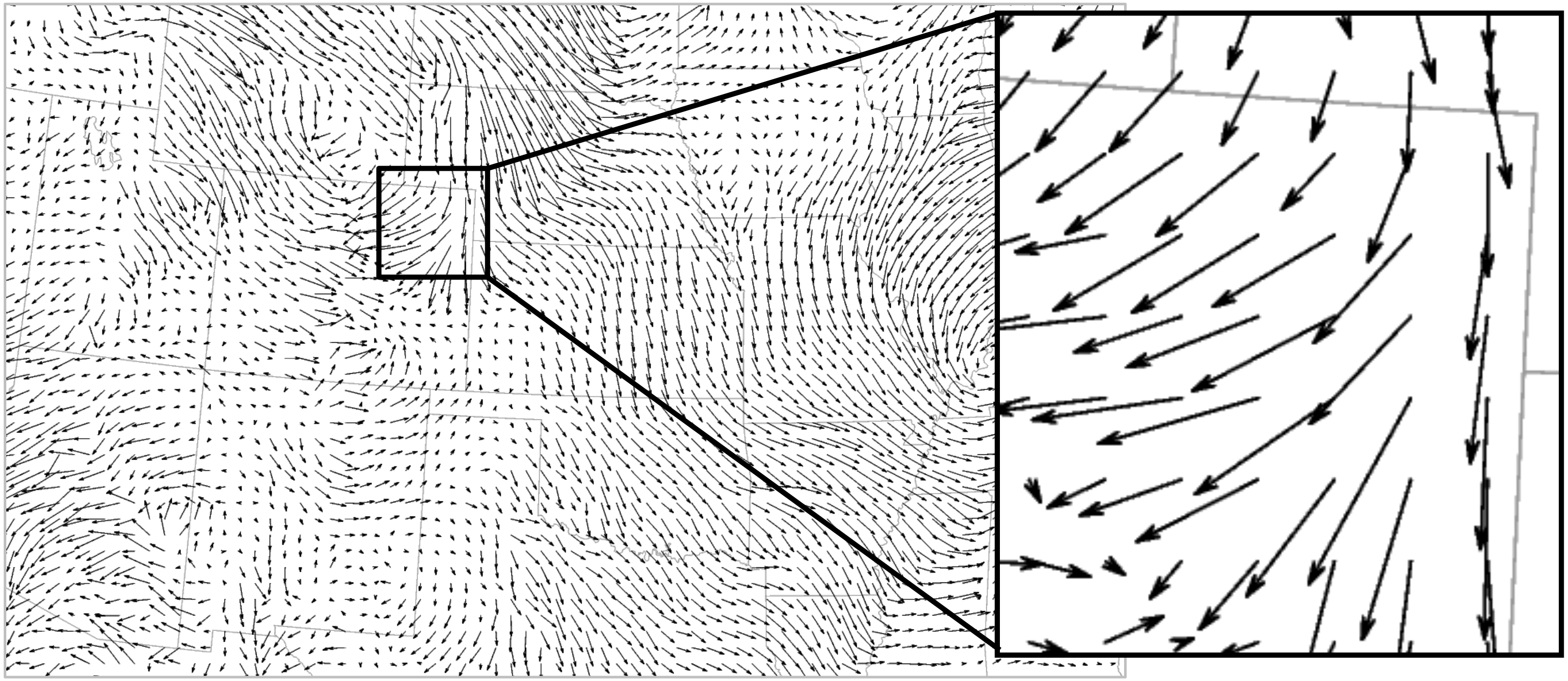


- ROW_MEAN_WDIR
  1. Derive wind directions and errors for each VL1L2 line
  2. Compute mean of errors



Wind Direction: Suggestions

- When aggregating, wind directions can cancel out.
  - Verify over regions with unimodal wind direction.
  - Verify u and v components separately.



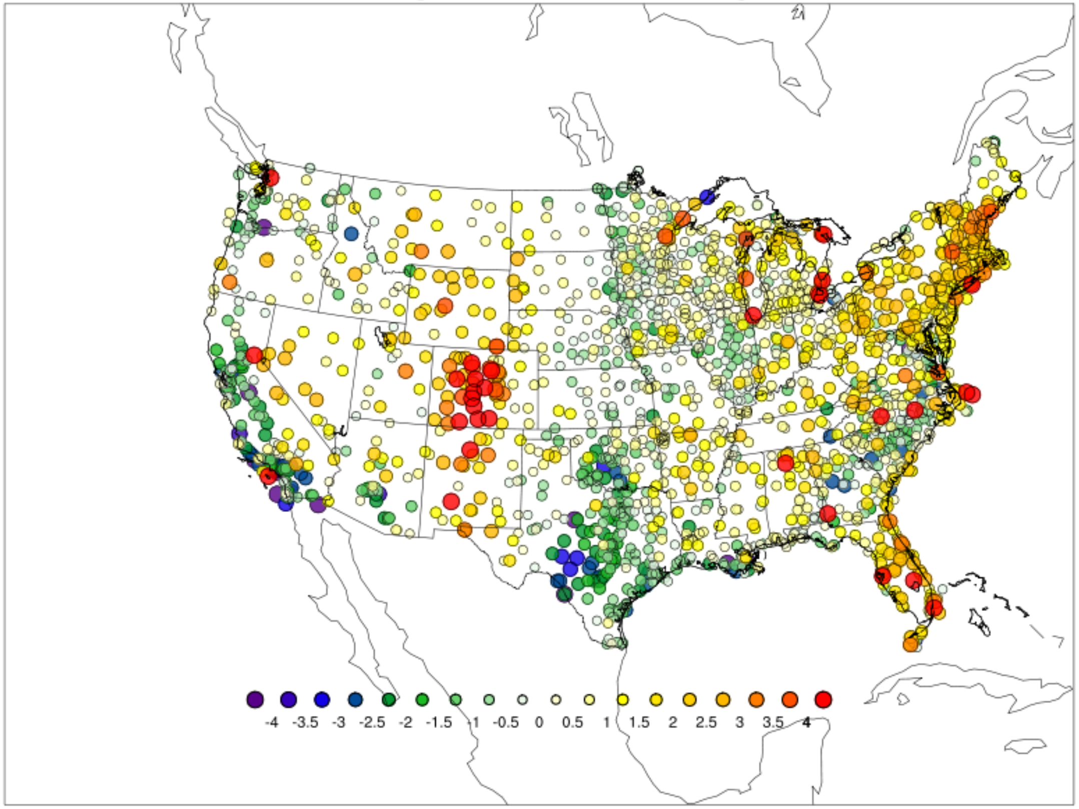
Stat-Analysis: -by case option

- Run the same analysis job for each unique set of values in one or more columns.
  - Compute aggregated continuous statistics for each forecast variable, masking region, and lead time:
    - stat_analysis –job aggregate_stat –line_type SL1L2 –out_line_type CNT –lookin out **–by fcst_var –by vx_mask –by fcst_lead**
  - Compute aggregated wind direction errors for each station:
    - stat_analysis –job aggregate_stat –line_type MPR –out_line_type WDIR –lookin out –fcst_lev Z2 **–by obs_sid**

Sample -by Case Plot

1. Run Point-Stat to generate MPR output.
2. Run Stat-Analysis to compute CNT stats for each station through time.
3. Plot Stat-Analysis output using NCL.

Dew Point Temperature Bias by Station ID



Config=AFWAOC_WRFv3.5 Season=WINTER Init=00UTC Fcst Hr=42h