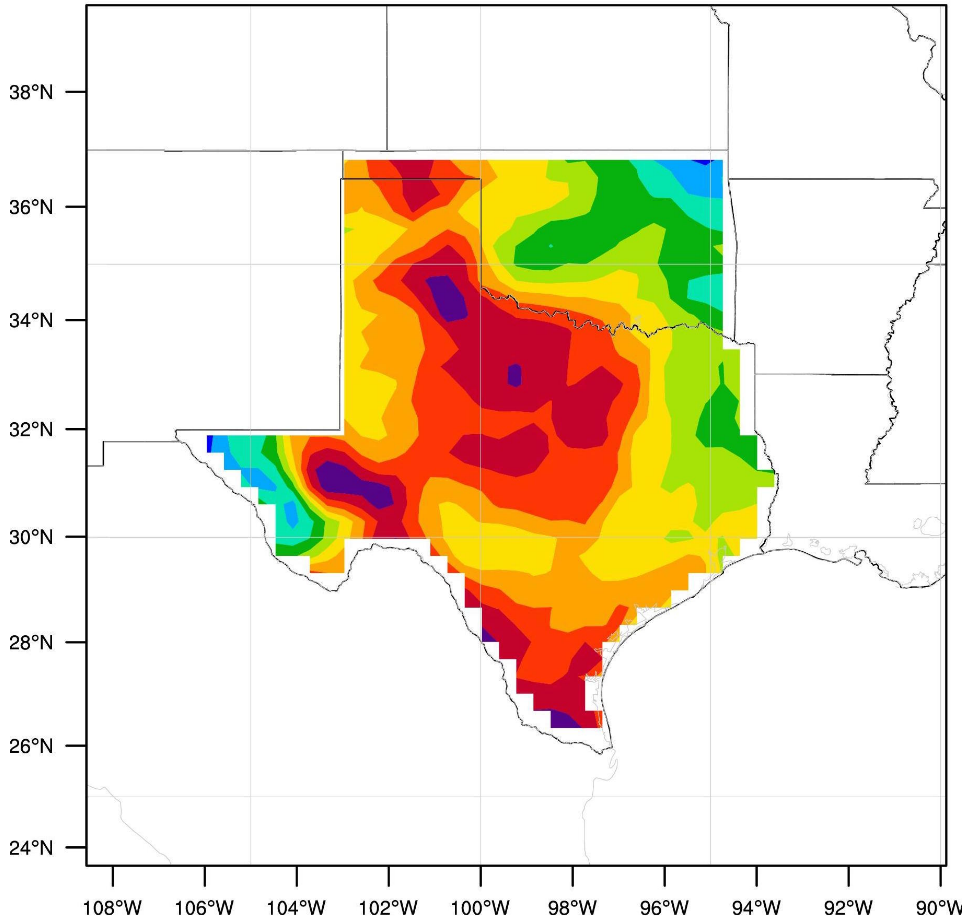


Data Manipulation and Evaluation From an Extremes Perspective

Abby Jaye

July 2019

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

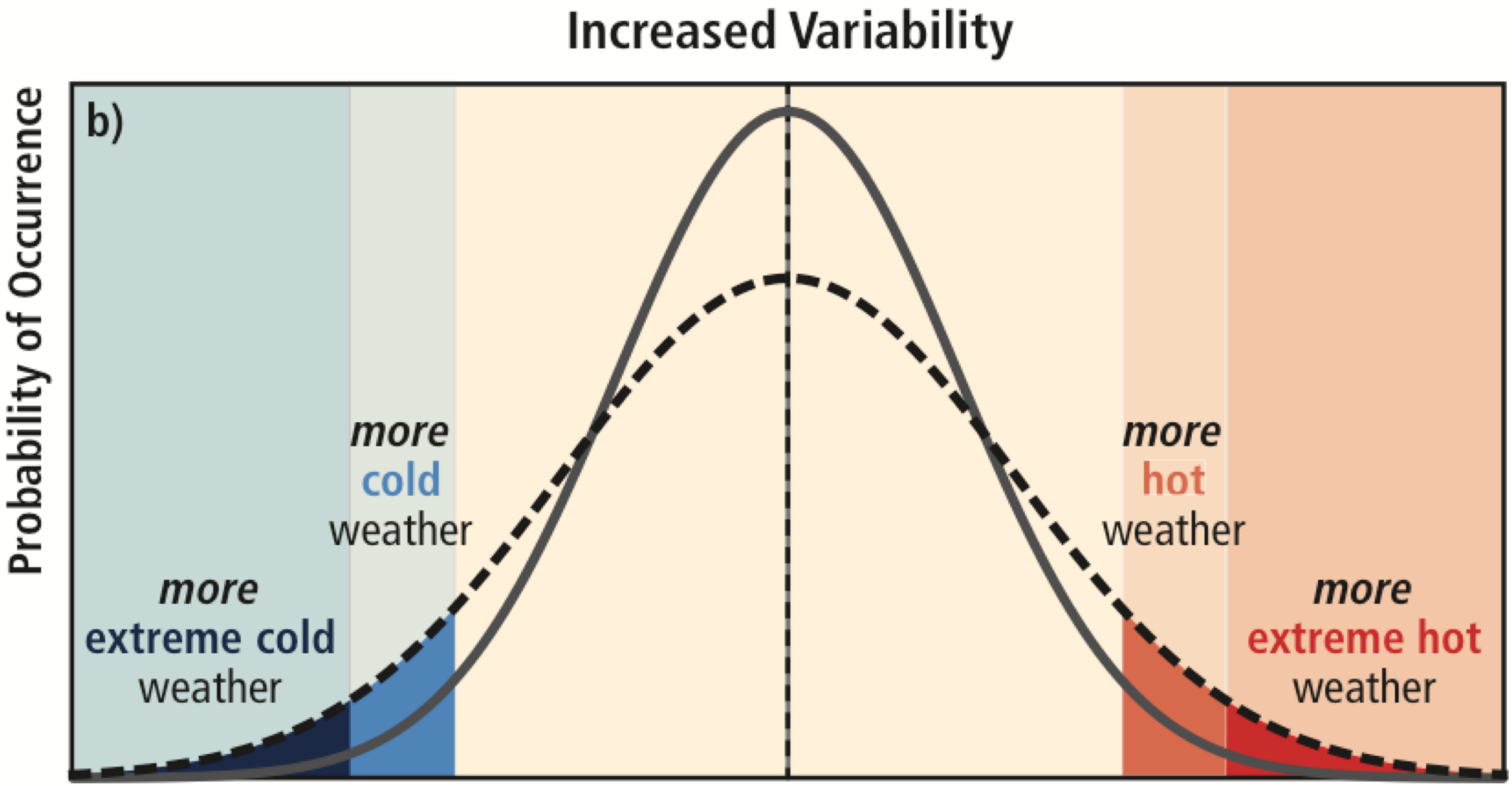
Dial Down for Extremes

- Does the model perform well at the large scale?
- Let's look at extremes from a more localized level.
  - Point data – What's going on in Boulder?
  - States, countries, counties, river basins, zip codes, etc

What are Extremes?

- Regular model evaluation: avg, bias, rmse, etc
- Extremes are typically a measure of exceedance above a threshold and statistics
- To look at extremes you need to be able to:
  1. Read your data
  2. Regrid your data in order to compare
  3. Extract the data in time and space (regular and irregular)
  4. Calculate statistics
  5. Plot data

What are Extremes?



Variables to be Evaluated

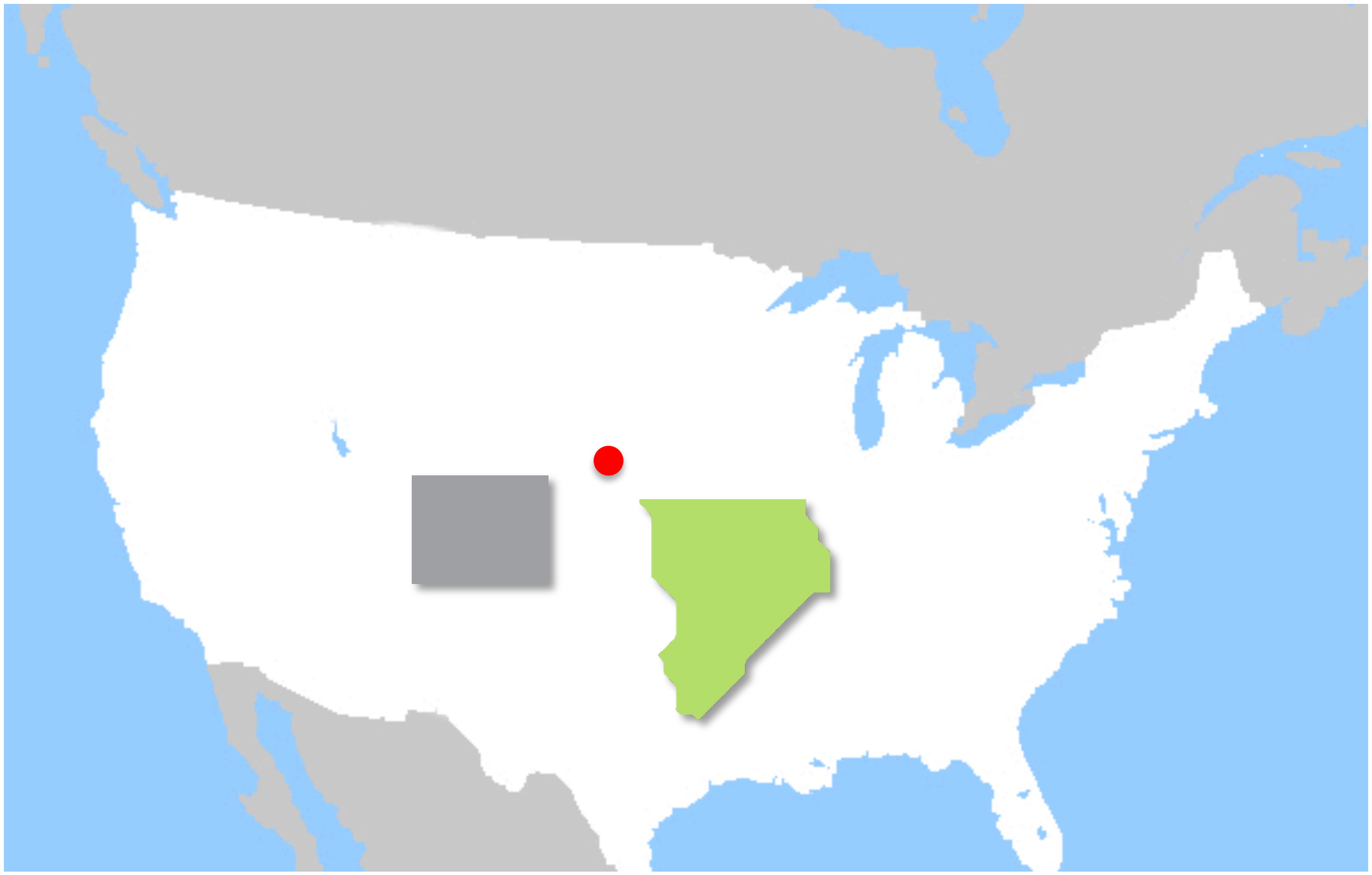
- Local variables –
  - Temperature
  - Precipitation
  - Specific Humidity (10m, 500hPa, 850hPa)
  - Wind (10m, 500hPa, 850hPa)
- Large-scale environments
  - Mean Sea Level Pressure
  - Sea Surface Temperature
  - Geopotential Height

Extreme Indicators

Zhang, et al. (2017)

TXx	Max Tmax	Monthly Max of daily max temps
TNx	Max Tmin	Monthly Max of daily min temps
TXn	Min Tmax	Monthly Min of daily max temps
TNn	Min Tmin	Monthly Min of daily min temps
TN10p	Cool nights	% time when daily min temps < 10^{th} percentile
TX10p	Cool days	% time when daily max temps < 10^{th} percentile
TN90p	Warm nights	% time when daily min temps > 90^{th} percentile
TX90p	Warm days	% time when daily max temps > 90^{th} percentile
DTR	Temp Range	Mean difference between daily min and max temp
GSL	Growing Season	Number of days between set temp range
FD0	Frost Days	Number of days below 0 °C
SU25	Summer days	Number of days above 25 °C
TR20	Tropical nights	Number of min temp days above 20 °C
WSDI	Warm spells	At least 6 consecutive days above a threshold
CSDI	Cold Spells	At least 6 consecutive days below a threshold
RX1day	Max 1-day rain	Max 1-day precipitation
RX5day	Max 5-day rain	Max consecutive 5-day precipitation
SDII	Intensity Index	Total precipitation / number of wet days
R10	Heavy rain days	Days with precipitation > 10 mm
R20	Very heavy rain days	Days with precipitation > 20 mm
CDD	Consecutive Dry days	Number of consecutive days with < 1mm of rain
CWD	Consecutive Wet days	Number of consecutive days with > 1 mm of rain
R95p	Very Wet days	Number of days with precipitation > 95^{th} percentile
R99p	Extremely Wet days	Number of days with precipitation > 99^{th} percentile

What Would You Like to Examine



Full model domain

Regular area

A point

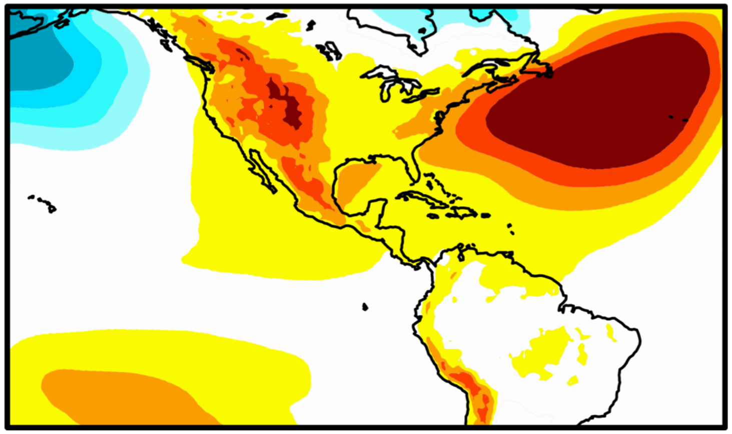
Climate region or river basin

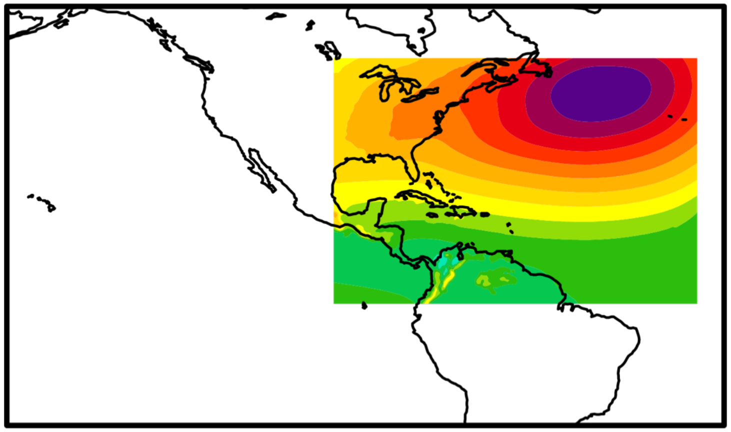
Comparisons with:

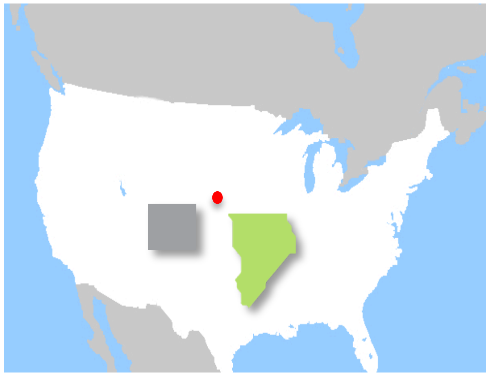
- Other models
- Re-analyses
- Observations

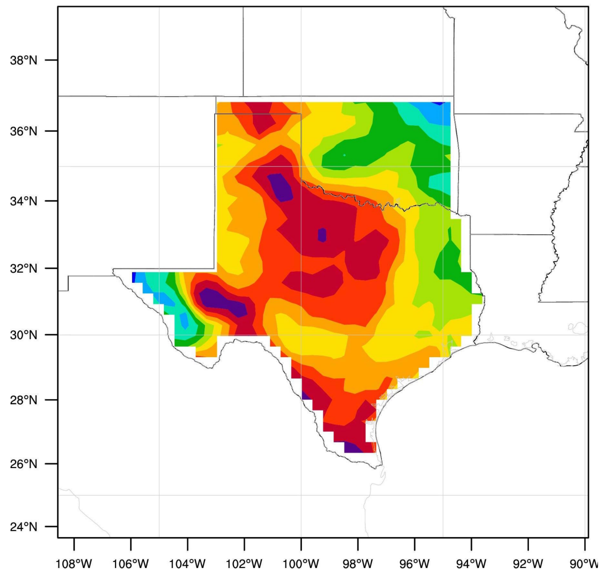
Might require regridding

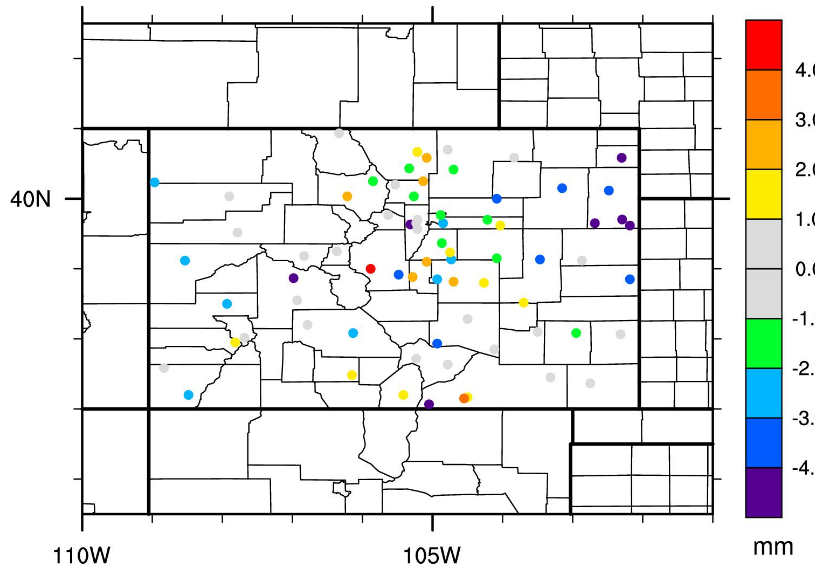
Display Spatial Information



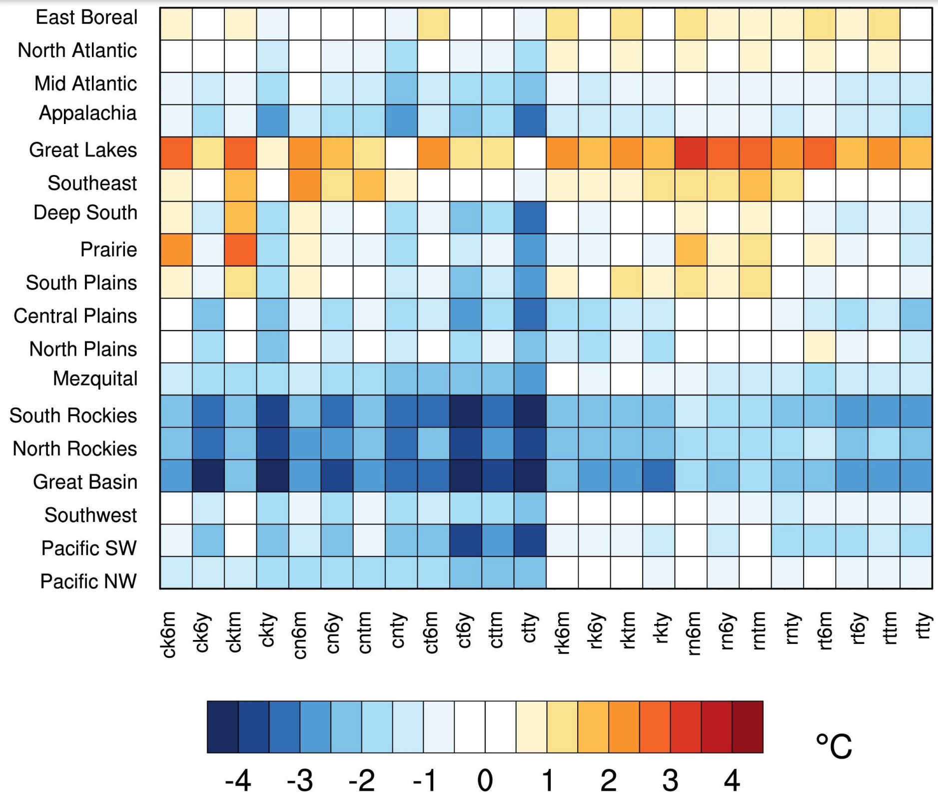


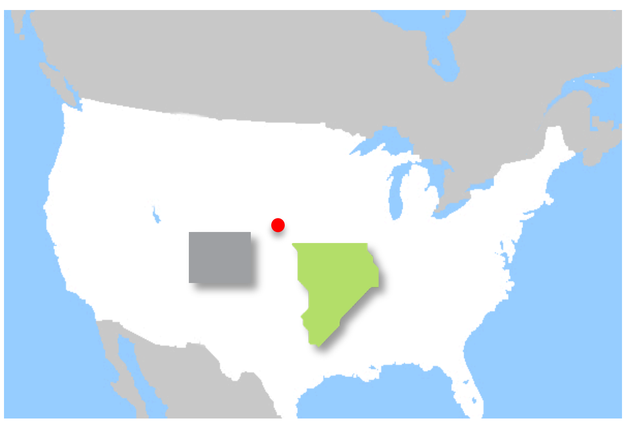


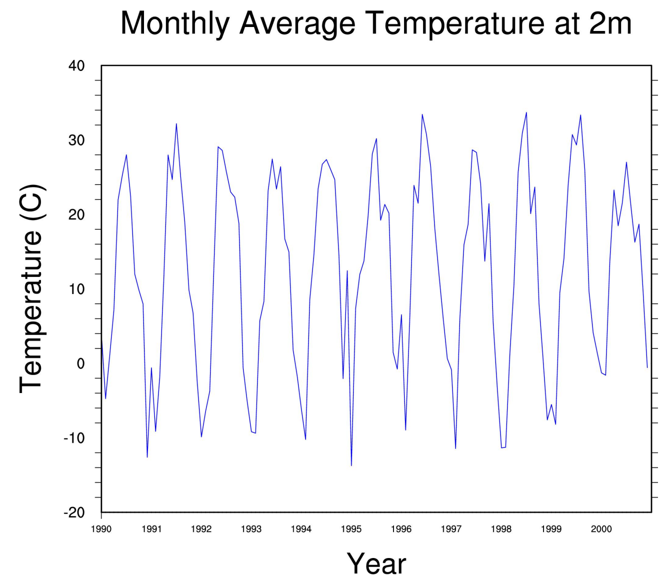


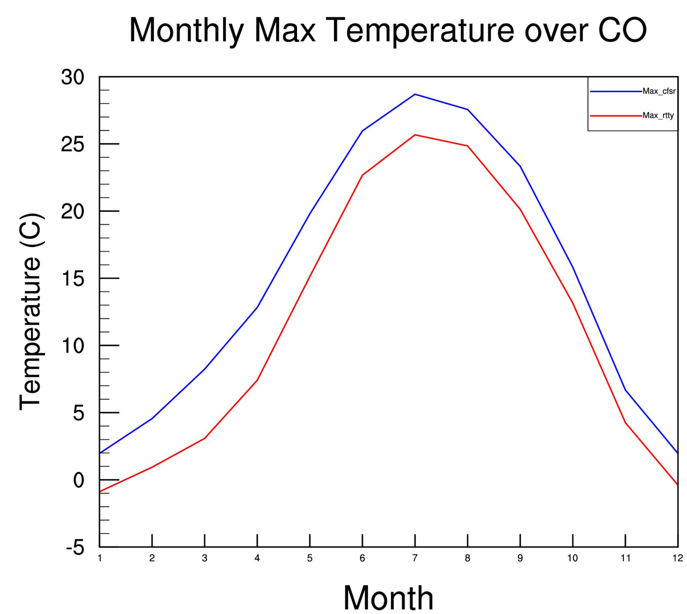


Display Aggregated Information



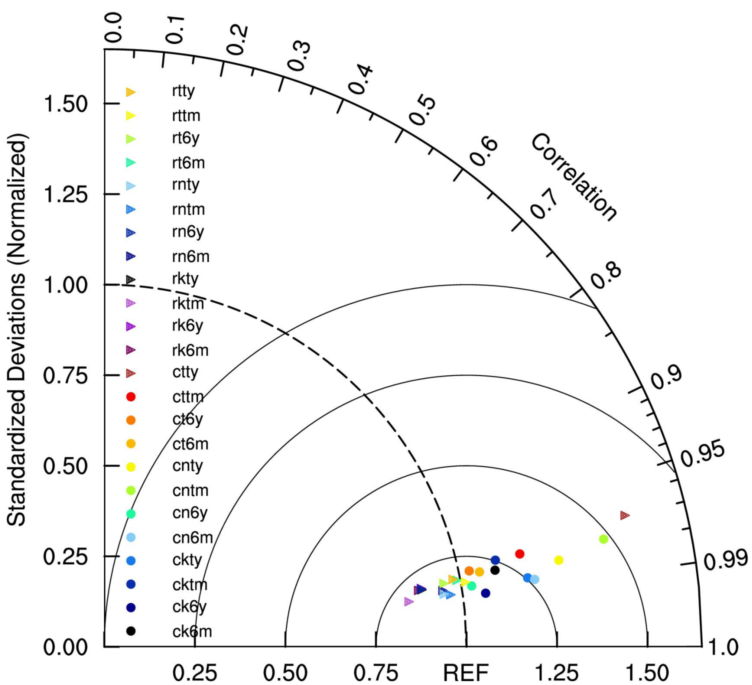




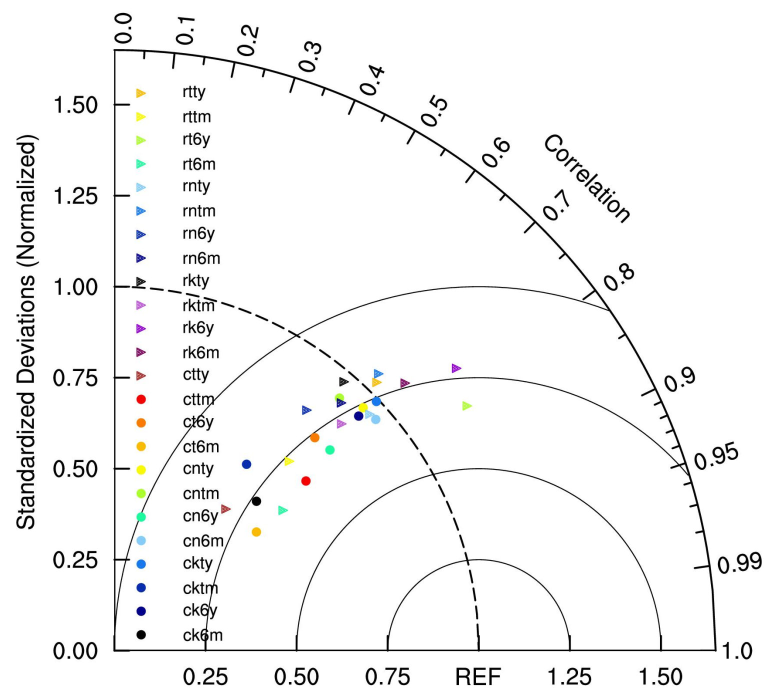


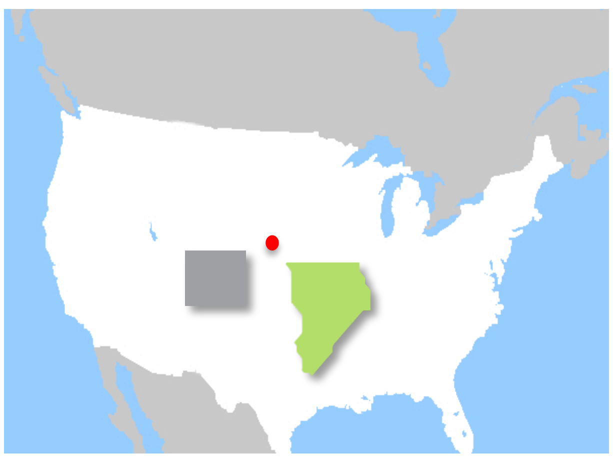
Display Statistical Information

Taylor Diagram for T2MAX JJA

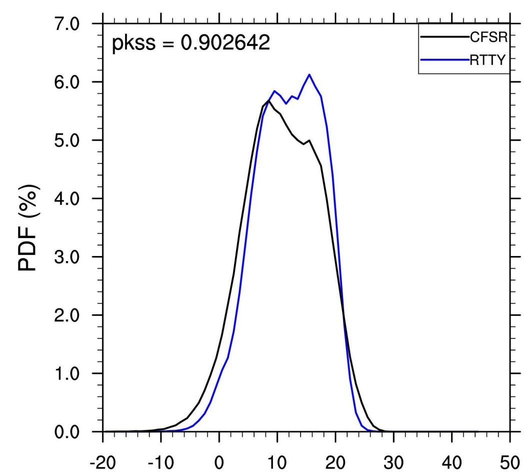


Taylor Diagram for T2MAX JJA - Colorado





T2MIN JJA Colorado



Which Tools Are Best?

I have my model runs… Now what?

- What science questions are you assessing?
- How large is your data set?
- How do you want to present your results?
  - *Visually e.g. maps of differences in variables*
  - *Tabulated e.g. statistical significance analyses*
  - *Correlations and dependencies*
- What tools are available for data analysis?

Questions to Consider

- Can it read your data
  - netCDF (CF), GRIB, ASCII, shapefiles
- Do you have to first pre-process the data
- Can it handle big datasets
- Is it purely a visualization tool, or can you do post-processing as well
- Which diagnostic / statistical functions does it have
- 3D or 2D visualization
- Can it deal with staggered grids
- How easy is it to add diagnostics
- How is data below ground handled
- Vertical grids
- How are model time stamps handled
- Ease of use
- Cost of package
- Support

**** Maybe more than one tool might be the solution ****

Using NCL and Python

- Traditionally we have used NCL for our analysis and plotting
- Are shifting some scripting to Python
- Lab will have a combination of NCL and Python

www.ncl.ucar.edu

- Python libraries: PyNGL, xarray, NumPy, matplotlib, wrf, netCDF4

WRF-Python

- A collection of diagnostic and interpolation routines for use with WRF-ARW
- Functionality is very similar to what is provided by the WRF NCL functions
- When coupled with either matplotlib or PyNGL you can create plots very similar to what you make with NCL

https://github.com/NCAR/wrf-python

Data Manipulation

- What format do we want the data in?
- What do we want to see?
- Example:
  - *We want to investigate heat waves over many different climate regions in a domain*
  1. *To look at extreme temperatures we want daily max and mins.*
  2. *We also want to pull out specific climate regions from these files using shapefiles.*
- We probably want to:
  1. *Manipulate the data (create averages, max/min, etc)*
  2. *Make a plot*

NCL Data Manipulation Python

NCL

```
out = addfile("t2_dailymax.nc","c") ; Create new
netCDF file

a = addfile("wrfout_d01.nc","r") ;24 hrs in file
t = a->T2-273.15
landmask = a->LANDMASK(0,:,:)
lat = a->XLAT(0,:,:)
lon = a->XLONG(0,:,:)


tland = mask(t,landmask,1) ; Mask out the ocean

tlandmax = dim_max_n(tland,0) ; Daily max

; Write attributes for the variable
tlandmax!0 = "Time"
tlandmax!1 = "south_north"
tlandmax!2 = "west_east"
tlandmax@units = "C"
tlandmax@coordinates = "XLONG XLAT"
tlandmax@description = "DAILY MAX TEMP at 2 M"



; Write out data
out->XLAT = lat
out->XLONG = lon
out->T2MAX = tlandmax
```

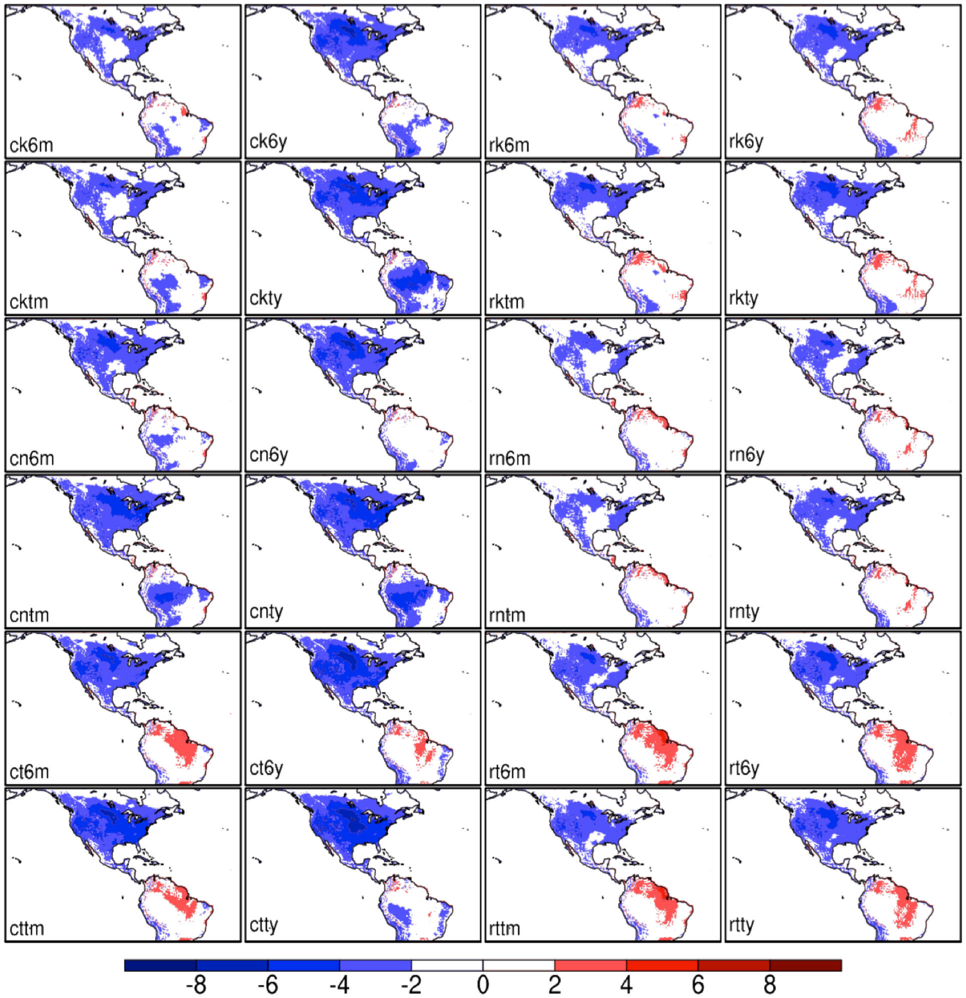
Python

```
out = Dataset("t2_dailymax.nc) ; Create new
netCDF file

a = Dataset("wrfout_d01.nc","r") ;24hrs in file
t = a.variables['T2']-273.15
landmask = a.variables['LANDMASK'][0,:,:]
lat = a.variables['XLAT'][0,:,:]
lon = a.variables['XLONG'][0,:,:]

tland = np.copy(t)
tland[:,landmask == 0] = np.nan ; Mask out ocean

tlandmax = np.nanmax(tland,axis=0) ; Daily max

; Write attributes for the variable
out.createDimension('Time',None)
out.createDimension('south_north',279)
out.createDimension('west_east',395)
T2MAX = out.createVariable('T2MAX',float,\
        ('Time','south_north','west_east'))
T2MAX.setncatts({'description': u'DAILY MAX TEMP
AT 2M','coordinates': u'XLONG XLAT','units':
u'C'})

; Write out data
out.variables['XLAT'][:]=lat[:]
out.variables['XLONG'][:]=lon[:]
out.variables['T2MAX'][:]=tlandmax[:]
out.close()
```

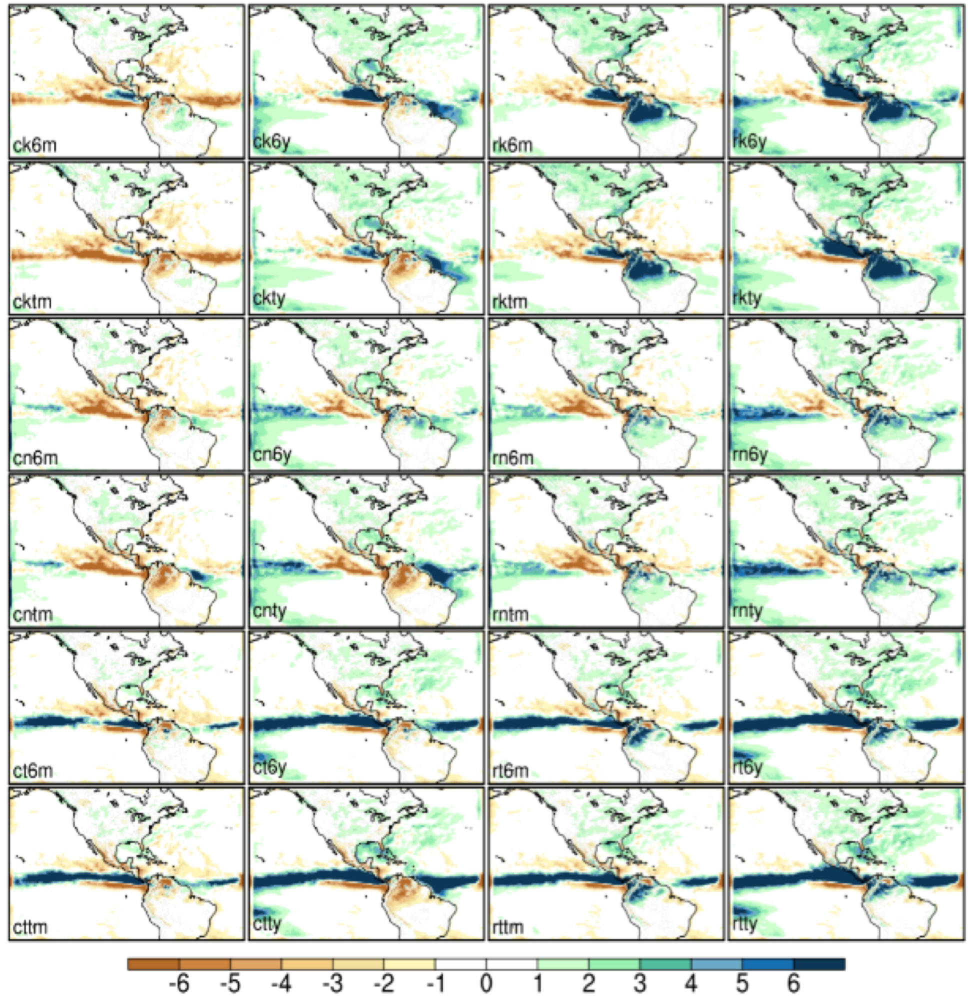
Spatial Examples

Science question – I have an ensemble of model runs – how do they compare?

Bias Map



Annual maximum surface temperature anomaly (K) over the period 1990-2010 for each ensemble member.



June-July-August (JJA) precipitation anomaly (mm/day) over the period 1990-2010 for each ensemble member.

Bruyere et al. 2017

Spatial Examples

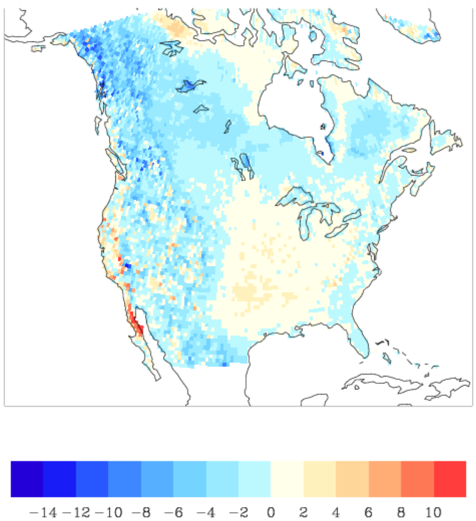
Science question – How well does my model compare with other models or reanalysis?

Bias Map

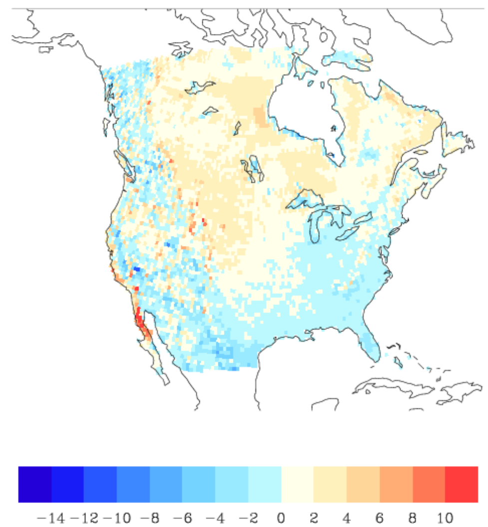
Summer [June–August (JJA)] temperature bias

NARCCAP

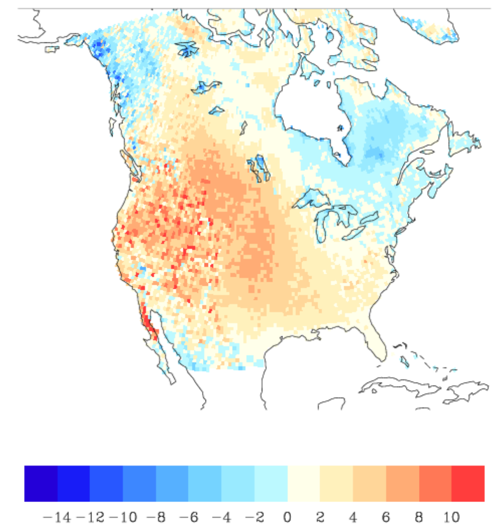
CRCM-UDEL Summer Temps Regrided .5 degree



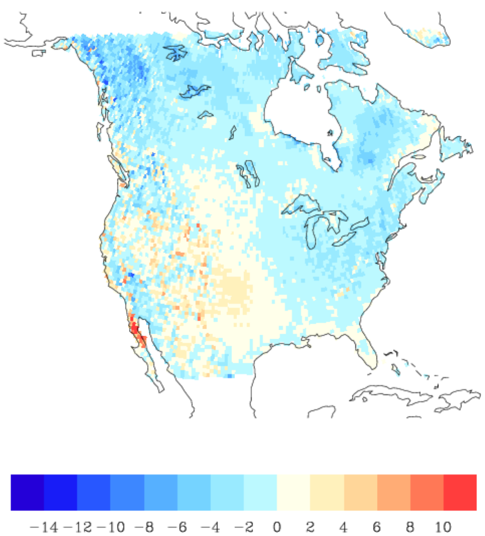
ECPC-UDEL Summer Temps Regrided .5 degree



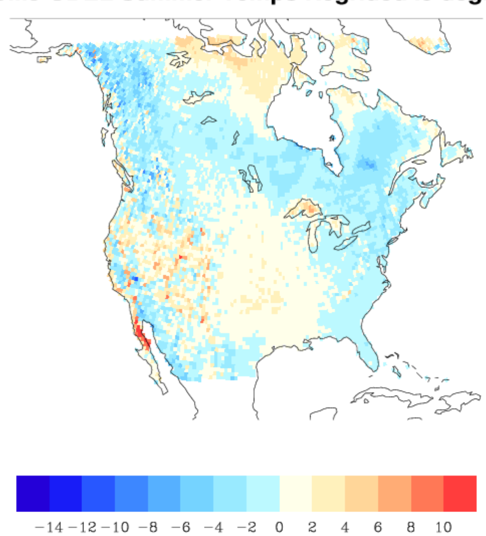
HRM3-UDEL Summer Temps Regrided .5 degree



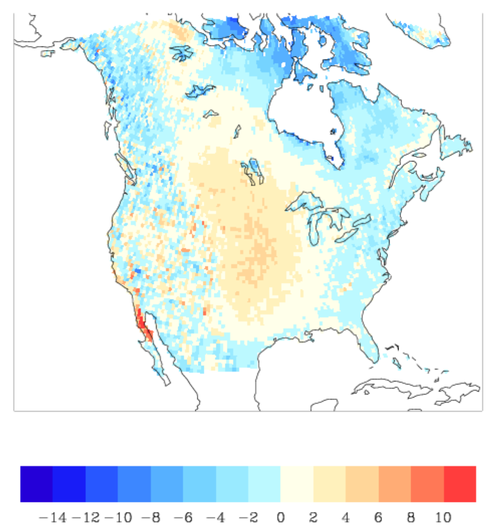
MM5I-UDEL Summer Temps Regrided .5 degree



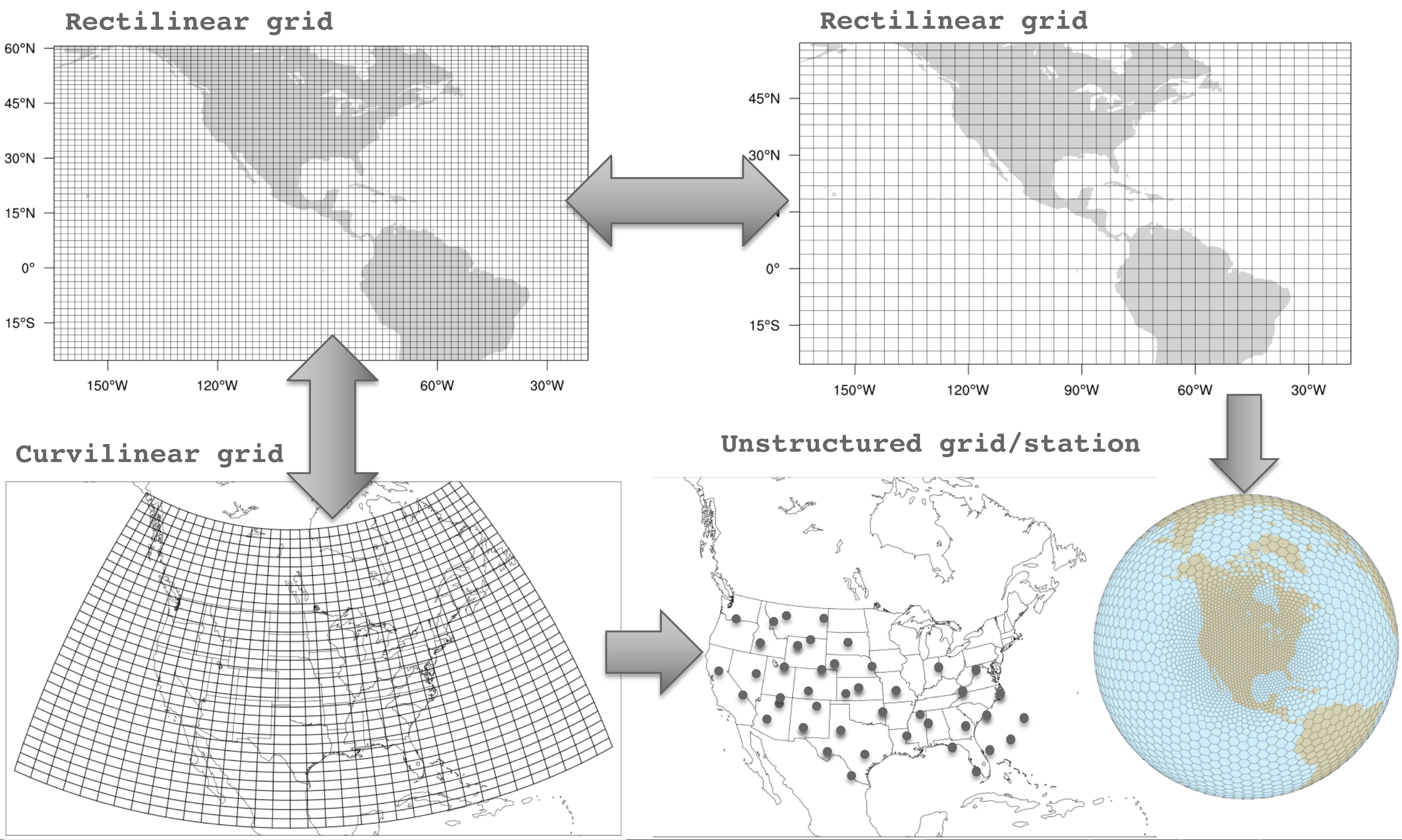
RCM3-UDEL Summer Temps Regrided .5 degree



WRFP-UDEL Summer Temps Regrided .5 degree



Regridding/Interpolation



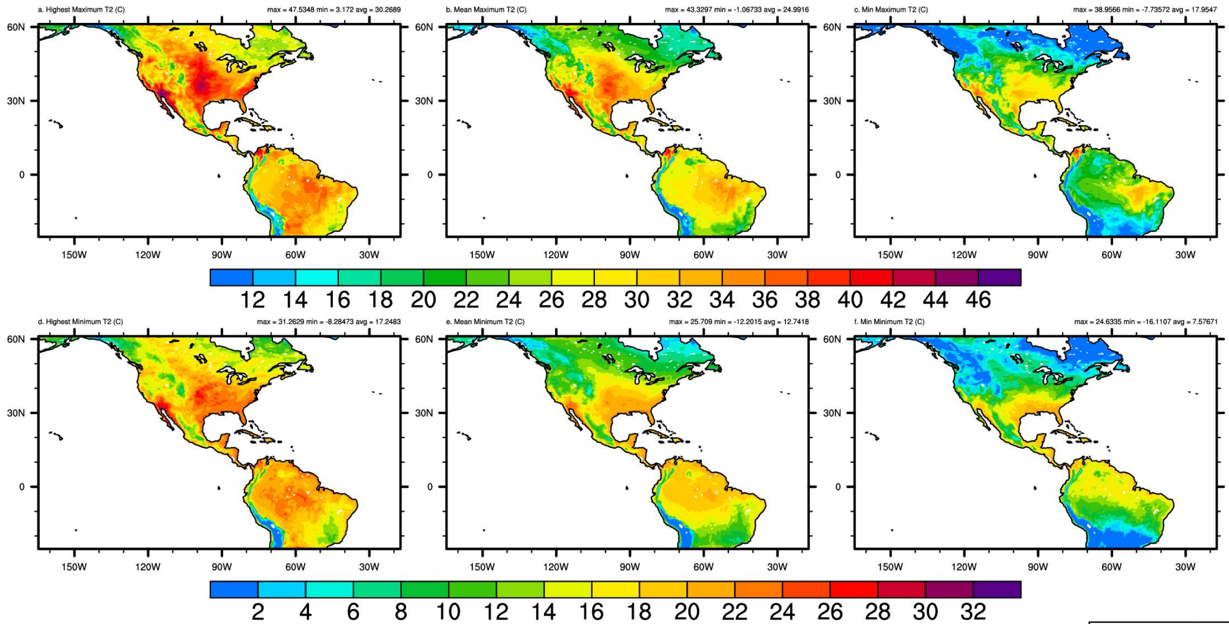
Regridding Tools in NCL and Python

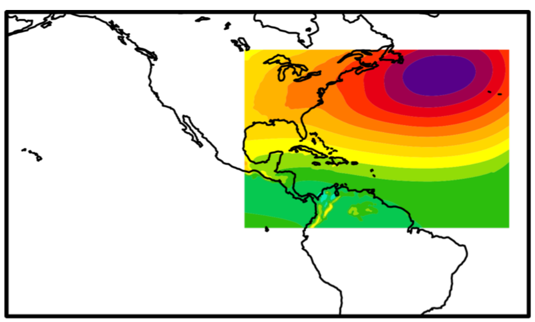
- ESMF (Earth System Modeling Framework)
  - Provides regridding on rectilinear, curvilinear, and unstructured grids using bilinear patch, or conservative interpolation
- Conventional bilinear interpolation
  - linit2, linit2_Wrap
- Rectilinear to curvilinear (and the other way)
  - rgrid2rcm, rgrid2rcm_Wrap
  - rcm2rgrid, rcm2rgrid_Wrap
- Conservative remapping
  - area_conserve_remap, area_conserve_remap_Wrap
- NCL website provides many great examples!

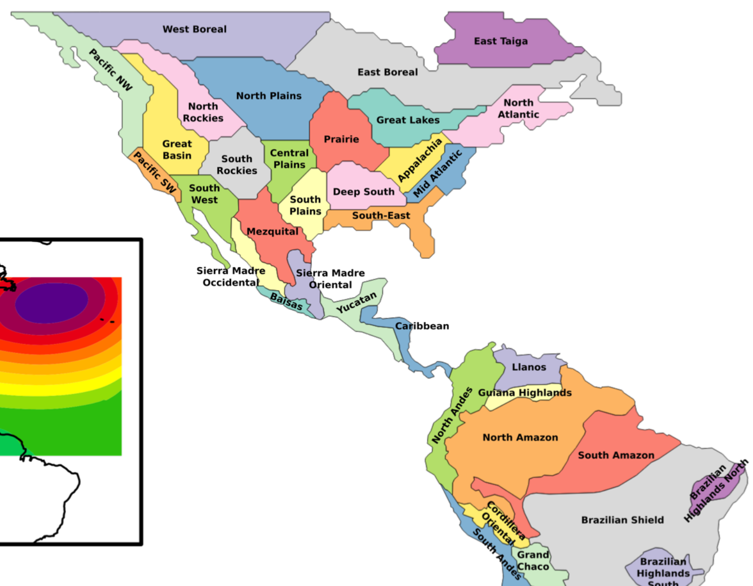
Working With Subsets of Data

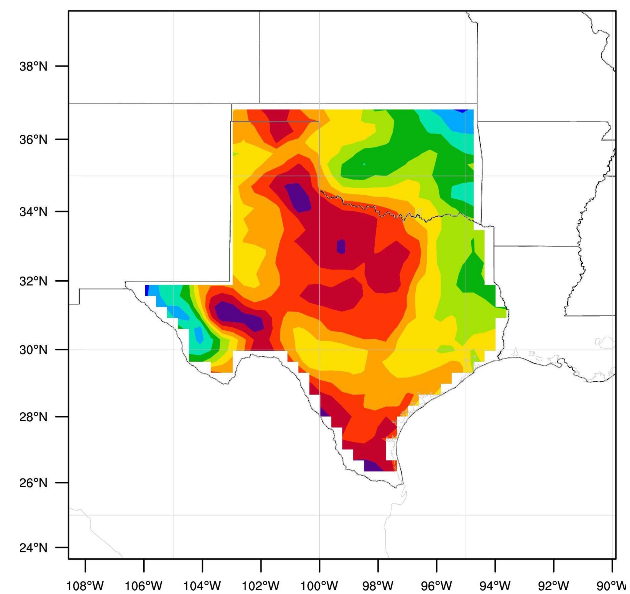
For subsetting, you might only be interested in the performance of the model over land, over Colorado or a climate zone

Data Manipulation









Shapefiles and WRF

- A geospatial vector data format for GIS systems software
- We can use it to mask data to specific regional or state borders, rather than drawing a box over an area
- Shapefiles can have three different types of data:
  - Point (locations of cities or places of interest, population data, election data)
  - Polyline (non-closed boundaries like rivers and roads)
  - Polygon (closed geographic boundaries like countries, states, provinces, territories, and lakes)
  - Only one data type per shapefile!

Shapefiles and WRF

```
shp_filename = "cb_2014_us_state_20m.shp"
f = addfile(shp_filename, "r")
print(f)        ; print shapefile metadata

id = f->NAME

print(id)
```

```
Variable: f
Type: file
filename:       cb_2014_us_state_20m
path:   /glade/p/p66770001/DNV_Ensembles/T2/ShapeFiles/ShapeFiles_US/cb_2014_us_state_20m.shp
   file global attributes:
      layer_name : cb_2014_us_state_20m
      geometry_type : polygon
      geom_segIndex : 0
      geom_numSegs : 1
      segs_xyzIndex : 0
      segs_numPnts : 1
   dimensions:
      geometry = 2
      segments = 2
      num_features = 52  // unlimited
      num_segments = 132
      num_points = 13785
   variables:
      integer geometry ( num_features, geometry )

      integer segments ( num_segments, segments )

      double x ( num_points )

      double y ( num_points )

      double z ( num_points )

      string STATEFP ( num_features )

      string STATENS ( num_features )

      string AFFGEOID ( num_features )

      string GEOID ( num_features )

      string STUSPS ( num_features )

      string NAME ( num_features )

      string LSAD ( num_features )

      double ALAND ( num_features )

      double AWATER ( num_features )
```

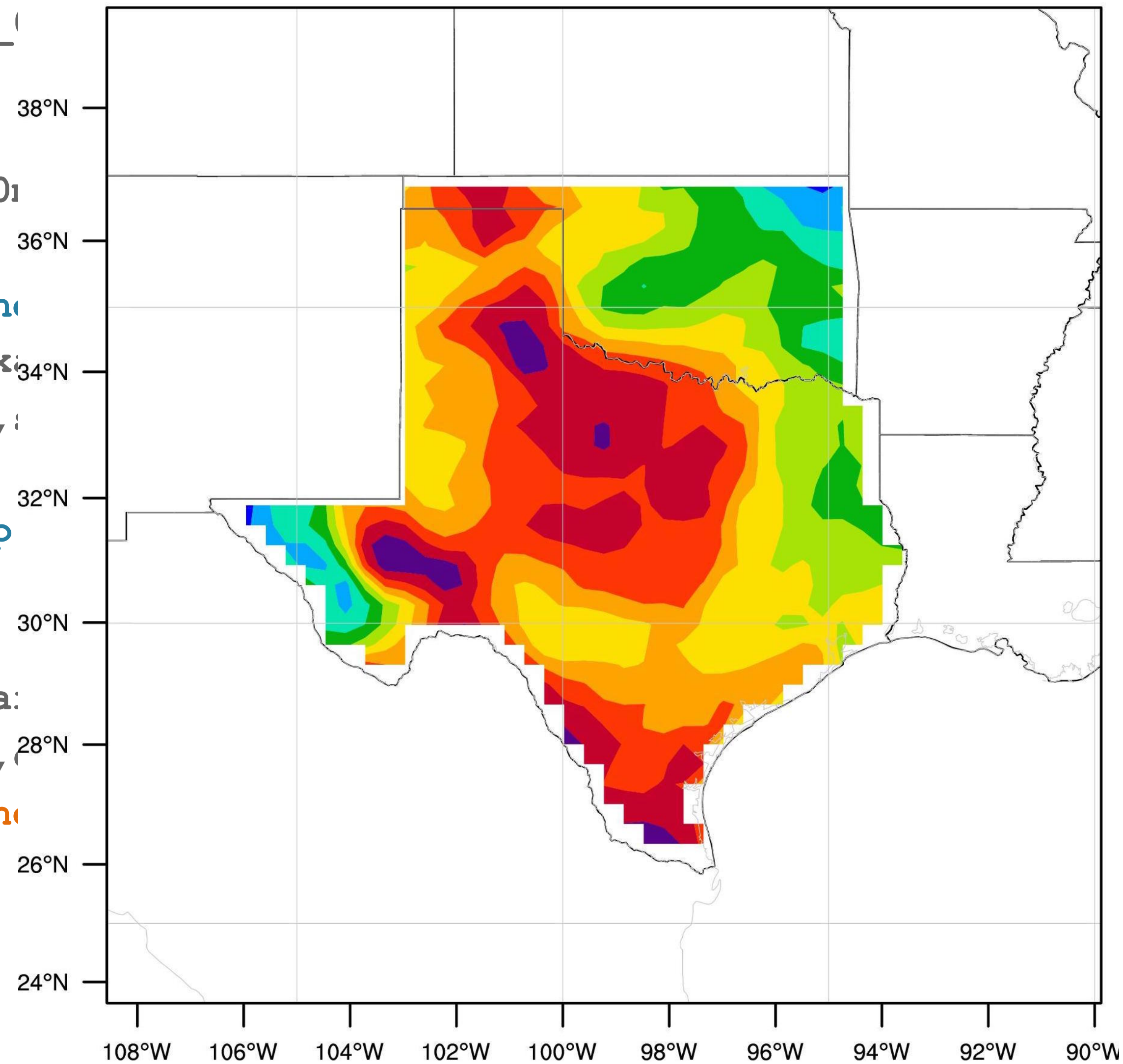
```
Variable: id
Type: string
Total Size: 416 bytes
            52 values
Number of Dimensions: 1
Dimensions and sizes:   [num_features | 52]
Coordinates:
Number Of Attributes: 0
(0)     California
(1)     District of Columbia
(2)     Florida
(3)     Georgia
(4)     Idaho
(5)     Illinois
(6)     Iowa
(7)     Kentucky
(8)     Louisiana
(9)     Maryland
(10)    Michigan
(11)    Minnesota
(12)    Missouri
(13)    New York
(14)    Oregon
(15)    Tennessee
(16)    Texas
(17)    Virginia
(18)    Wisconsin
(19)    Alaska
(20)    Arizona
(21)    Arkansas
(22)    Colorado
(23)    Indiana
(24)    Connecticut
(25)    Hawaii
(26)    Nebraska
(27)    New Mexico
(28)    North Carolina
(29)    Ohio
(30)    Maine
(31)    Massachusetts
(32)    Mississippi
(33)    Montana
(34)    Oklahoma
(35)    South Carolina
(36)    South Dakota
(37)    Utah
(38)    Washington
(39)    West Virginia
(40)    Wyoming
(41)    Delaware
(42)    Rhode Island
(43)    Alabama
(44)    North Dakota
(45)    Pennsylvania
(46)    Vermont
(47)    Puerto Rico
(48)    Kansas
(49)    Nevada
(50)    New Hampshire
(51)    New Jersey
```

Shapefiles and WRF

```
; Read in file and variable
a = addfile("wrfout_d01_2000-07-17_00:00:00.nc","r")
var = a->T2(0,:,:)

shp_filename = "cb_2014_us_state_20m.shp" ; Shapefile from internet
opt = True
opt@shape_var = "NAME" ; We know the variable name
opt@shape_names = (/"Oklahoma","Texas"/) ; The states we want to mask
var_mask = shapefile_mask_data(var,shp_filename,opt) ; Mask the data

; Make contours, draw them on a map, then draw the outline of the polygons
opts = True
opts@cnFillOn = True
contour_mask = wrf_contour(a,wks,var_mask,opts)
plot_mask = wrf_map_overlays(a,wks,contour_mask,pltres,mpres)
id_mask = gsn_add_shapefile_polylines(wks,plot_mask,shp_filename,True)
draw(plot_mask)
```

Shapefiles and WRF

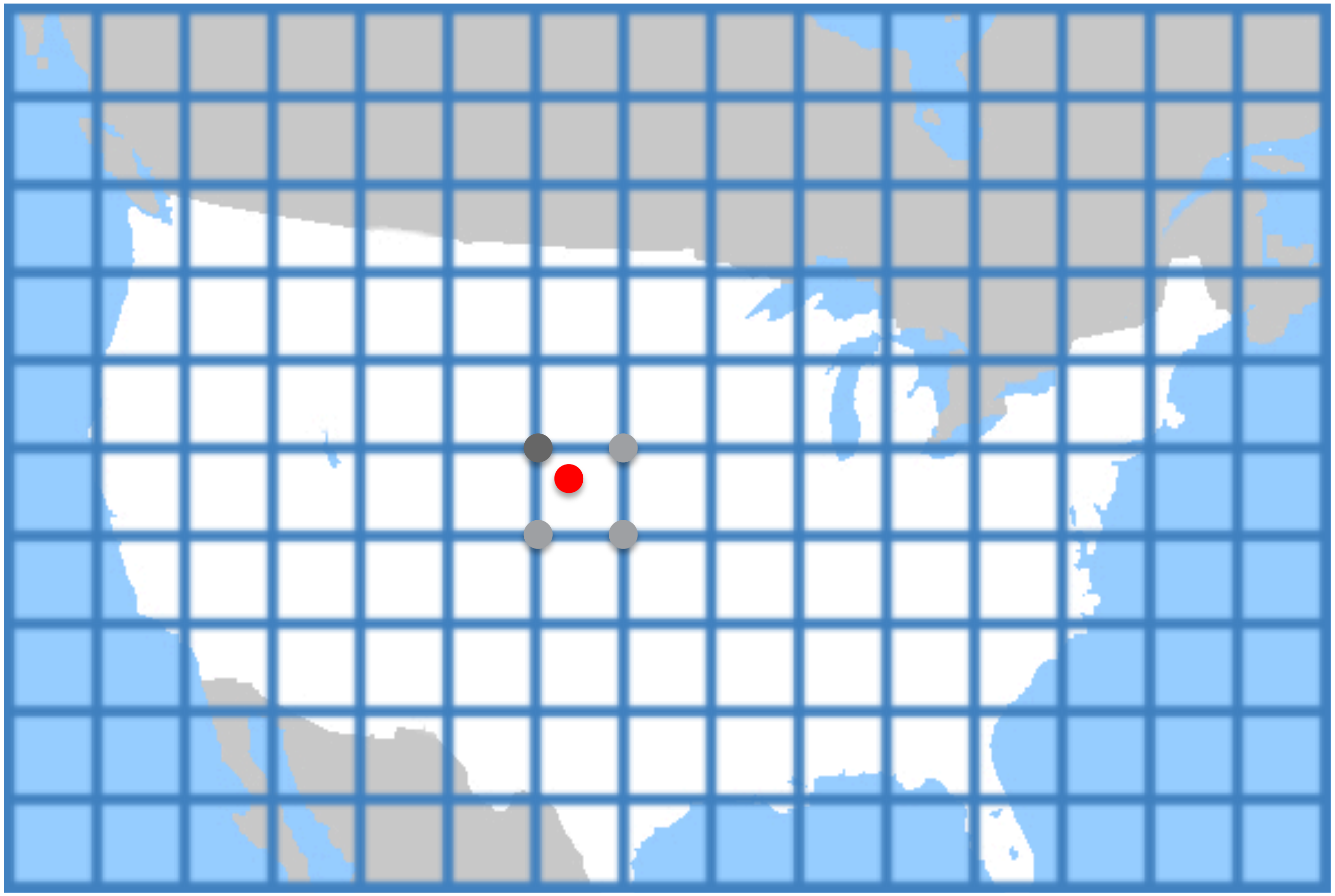
```
; Read in file and variable
a = addfile("wrfout_d01_2000-07-17_(
var = a->T2(0,:,:)

shp_filename = "cb_2014_us_state_20
opt = True
opt@shape_var = "NAME"  ; We know th
opt@shape_names = (/"Oklahoma","Tex
var_mask = shapefile_mask_data(var,

; Make contours, draw them on a map
opts = True
opts@cnFillOn = True
contour_mask = wrf_contour(a,wks,va
plot_mask = wrf_map_overlays(a,wks,
id_mask = gsn_add_shapefile_polylin
draw(plot_mask)
```



Working With Point Data

Typically when you want to compare your model results with observations

Thinking about extracting a point



- Your point of interest will likely not coincide with a model grid point
- Most researchers pick the closest point
- Alternatively, pick all points around point of interest and interpolate

NCL:	locij = wrf_user_ll_to_ij(a, lon, lat, True)
Python:	locij = ll_to_xy(a, lat, lon, True)

Find the Closest Grid Box to Lat/Lon

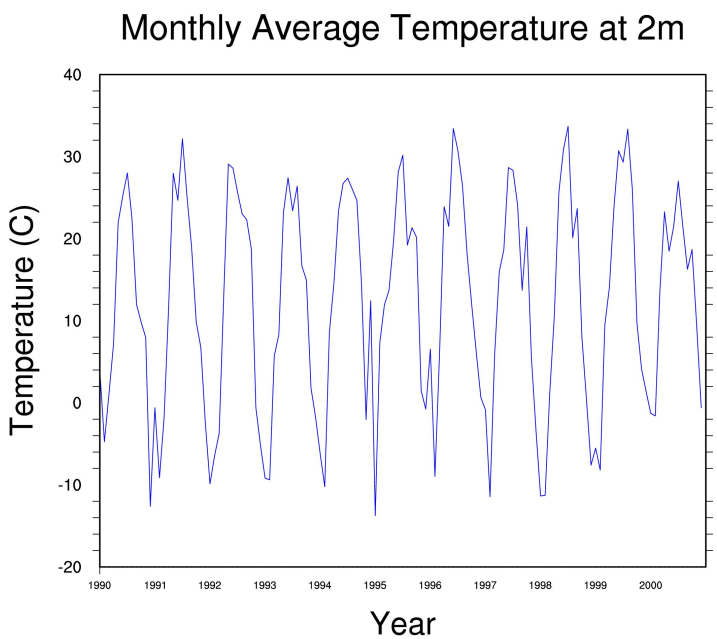
NCL:

```
locij = wrf_user_ll_to_ij(a,105.27,40.02,True)
locY = locij(0)-1
locX = locij(1)-1
t2_point = a->T2(:,locY,locX)
```

Python:

```
locij = ll_to_xy(a,40.02,105.27,True)
locY = locij(0)
locX = locij(1)
t2_point = a.variables['T2'][:,locY,locX]
```

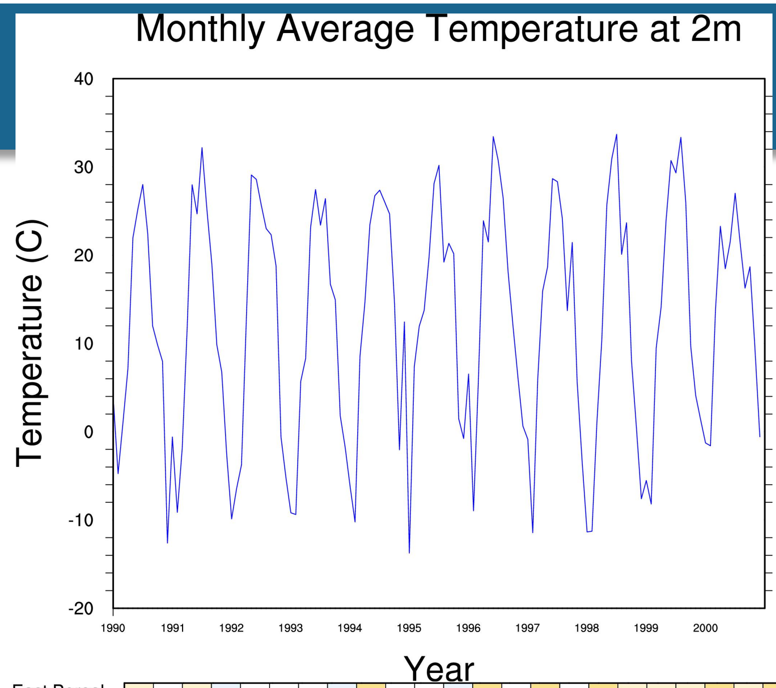
```
T2 taken from
a lat/lon
point in
Boulder
```

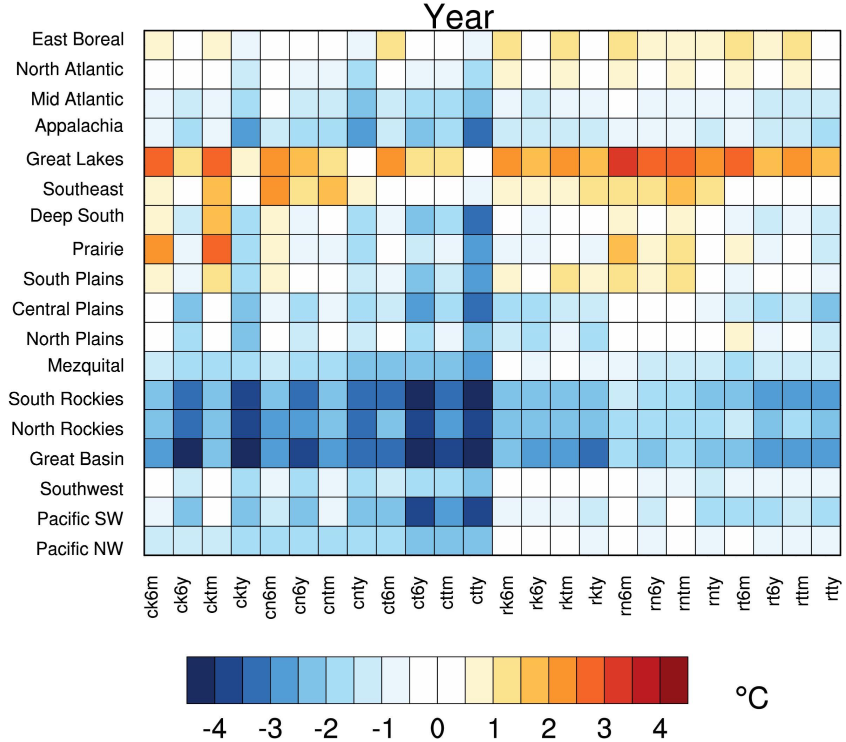


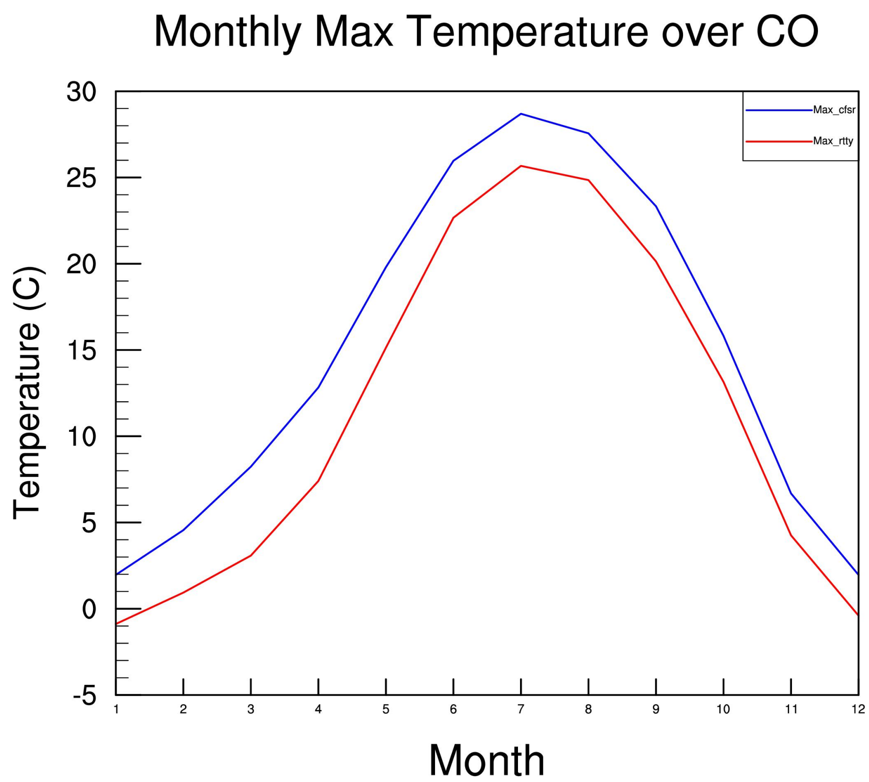
Aggregated Ways to Display Data

Does the model get year to year variability of temperature correct?

Does the model capture the annual and diurnal cycles?


Monthly Average Temperature at 2m
Temperature (C)
40
30
20
10
0
-10
-20
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
Year


East Boreal
North Atlantic
Mid Atlantic
Appalachia
Great Lakes
Southeast
Deep South
Prairie
South Plains
Central Plains
North Plains
Mezquital
South Rockies
North Rockies
Great Basin
Southwest
Pacific SW
Pacific NW
ck6m
ck6y
cktm
ckty
cn6m
cn6y
cntm
cnty
ct6m
ct6y
cttm
ctty
rk6m
rk6y
rktm
rkty
rn6m
rn6y
rntm
rnty
rt6m
rt6y
rttm
rtty
-4
-3
-2
-1
0
1
2
3
4
°C


Monthly Max Temperature over CO
Temperature (C)
30
25
20
15
10
5
0
-5
Max_cfsr
Max_rtty
1
2
3
4
5
6
7
8
9
10
11
12
Month

Difference Map

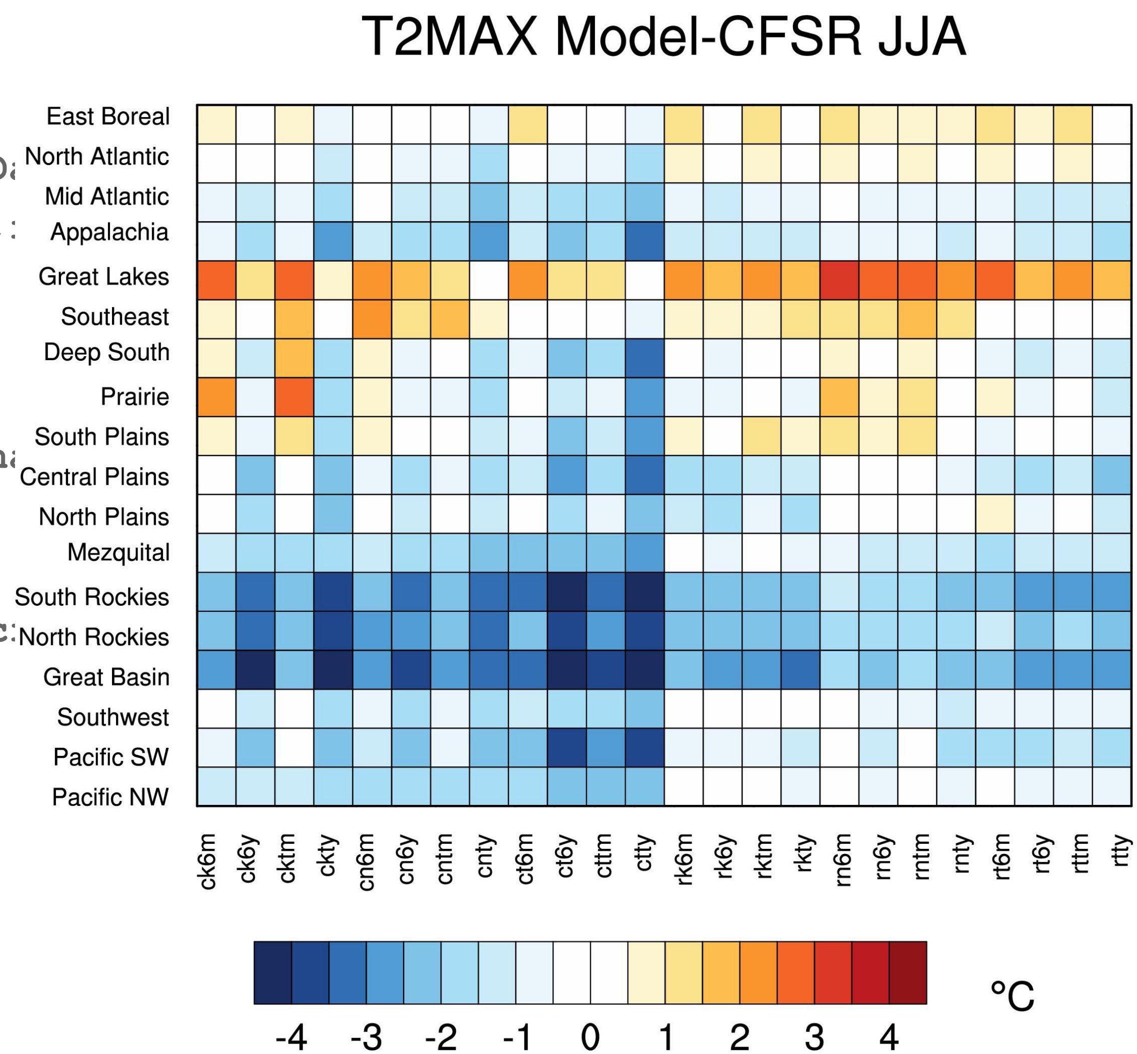
- First we want to subset our data
  - Input files are daily (365 days) for eleven years
  - We only care about June, July and August
- We will use a netCDF operator (NCO) called ncea to pull out the days we want

```
#! /bin/tcsh -f
#

foreach mem (cfsr ck6m ck6y cktm ckty ... rttm rtty)
  echo mem=$mem
  ncea -F -d JulianDay,152,243 t2_daily_Regional_Data_$mem.nc t2_daily_Regional_Data_jja_$mem.nc
end
```

nco.sourceforge.net

Difference Map

```
column = (/"ck6m","ck6y","cktm",...,"rttm",rtty"/)   ; Ensemble members
numColumn = dimsizes(column)
row = (/"PacificNW","PacificSW",...
numRow = dimsizes(row)
cfsr = addfile("t2_daily_Regional_Da
t2_diff = new((/numRow,numColumn/),:
do i=0,numColumn-1
  do j=0,numRow-1
    var = "T2MAX_"+row(j)
    data = addfile("t2_daily_Regiona
    t2 = data->$var$
    t2_cfsr = cfsr->$var$-273.15
    t2_diff(j,i) = avg(t2)-avg(t2_c
  end do
end do
```



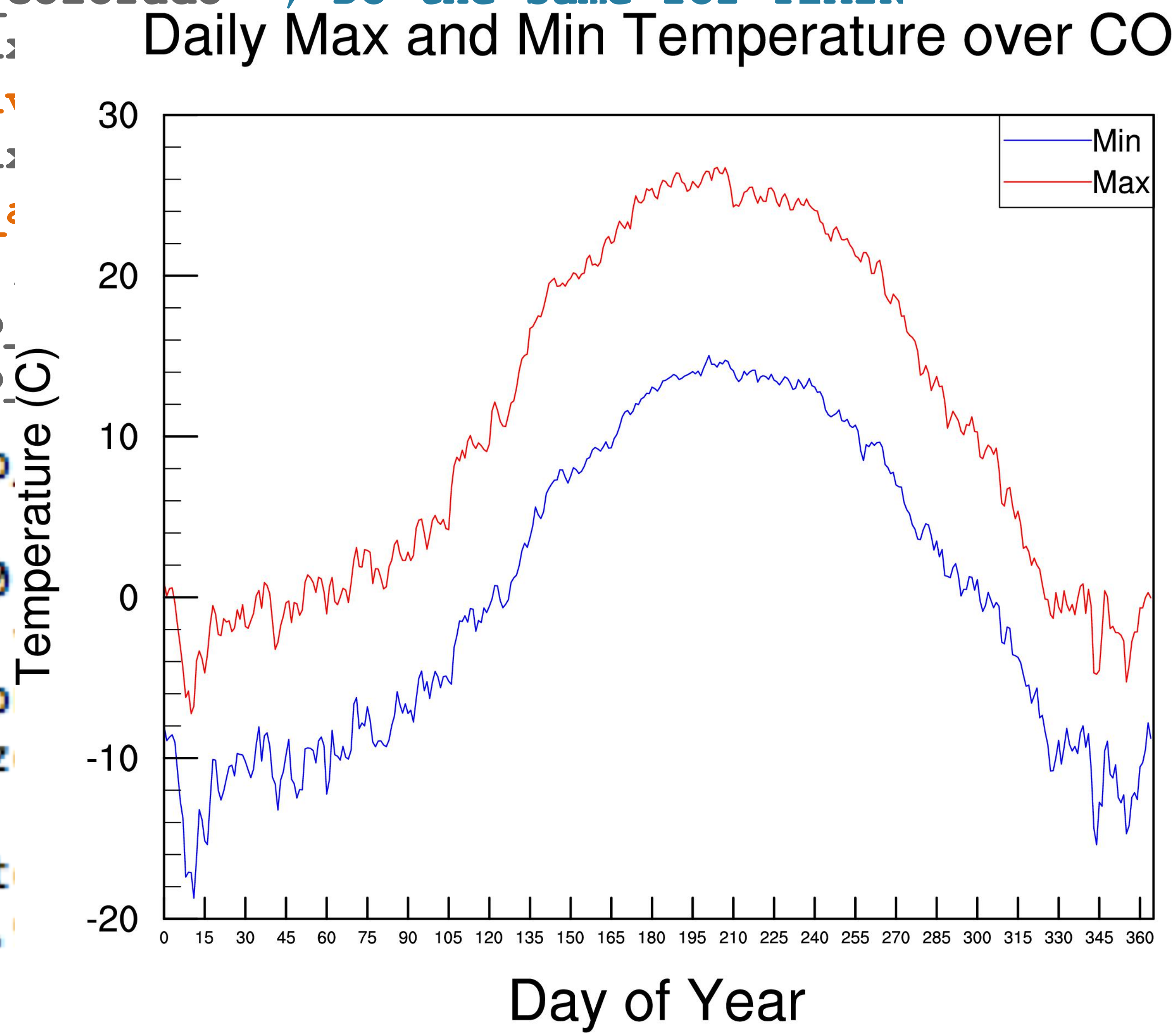
Time Series

- Annual cycle
  - T2Max and T2Min
  - We can plot it using daily data, or we can make monthly averages
- Diurnal cycle
  - Precipitation
  - We will look at station data over Colorado
  - Compare model data to observations spatially and by amount, number of occurrences and intensity.

Time Series

```
t2max_co = a->T2MAX_Colorado   ; Do the same for T2MIN
printVarSummary(t2ma
t2max_co_avg = dim_av
printVarSummary(t2ma
t2max_co_avg2 = dim_a
data = new((/2,365/)
data(0,:) = t2max_co_
data(1,:) = t2min_co_
```

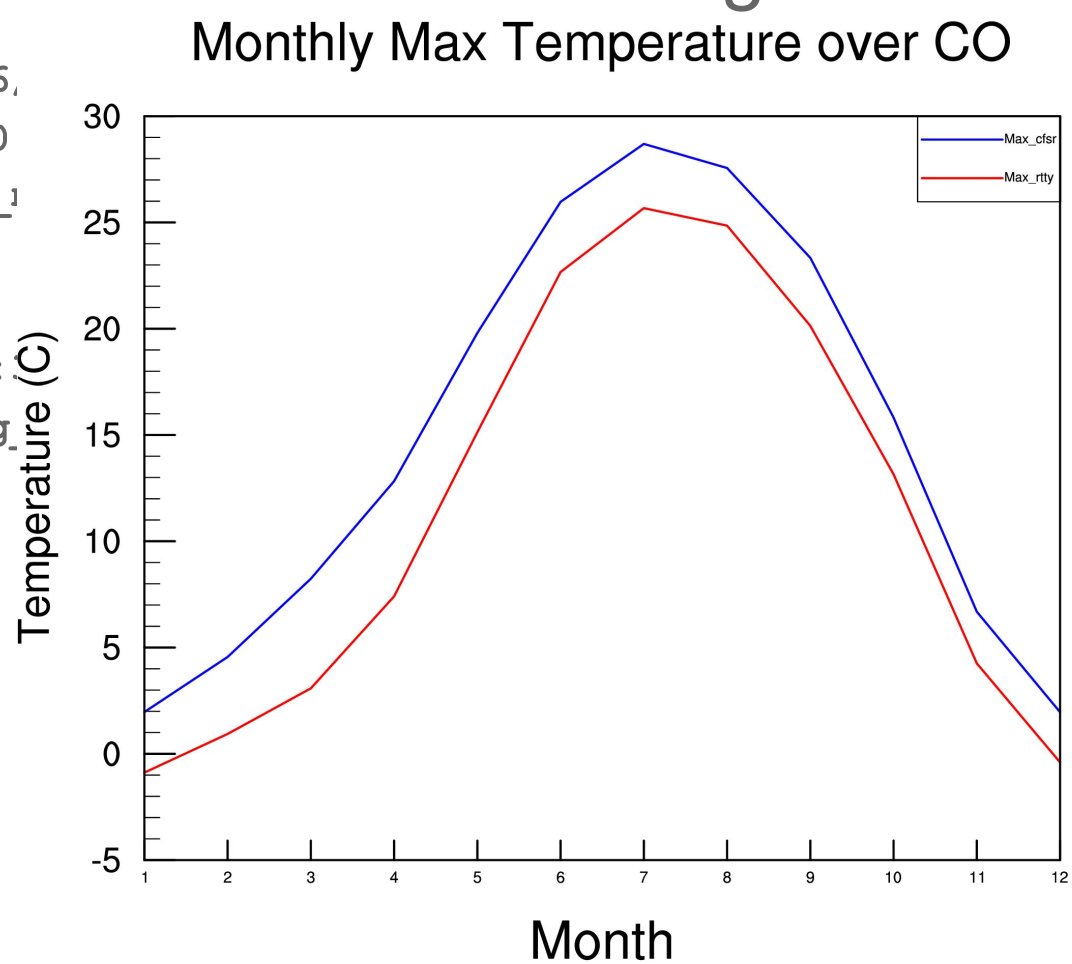
```
Variable: t2max_co_
Type: float
Total Size: 388360
            97090
Number of Dimensio
Dimensions and siz
Coordinates:
Number Of Attribut
  _FillValue :  9.
```



Time Series

- Let's try it with monthly averages
  - The main difference is with the following lines:

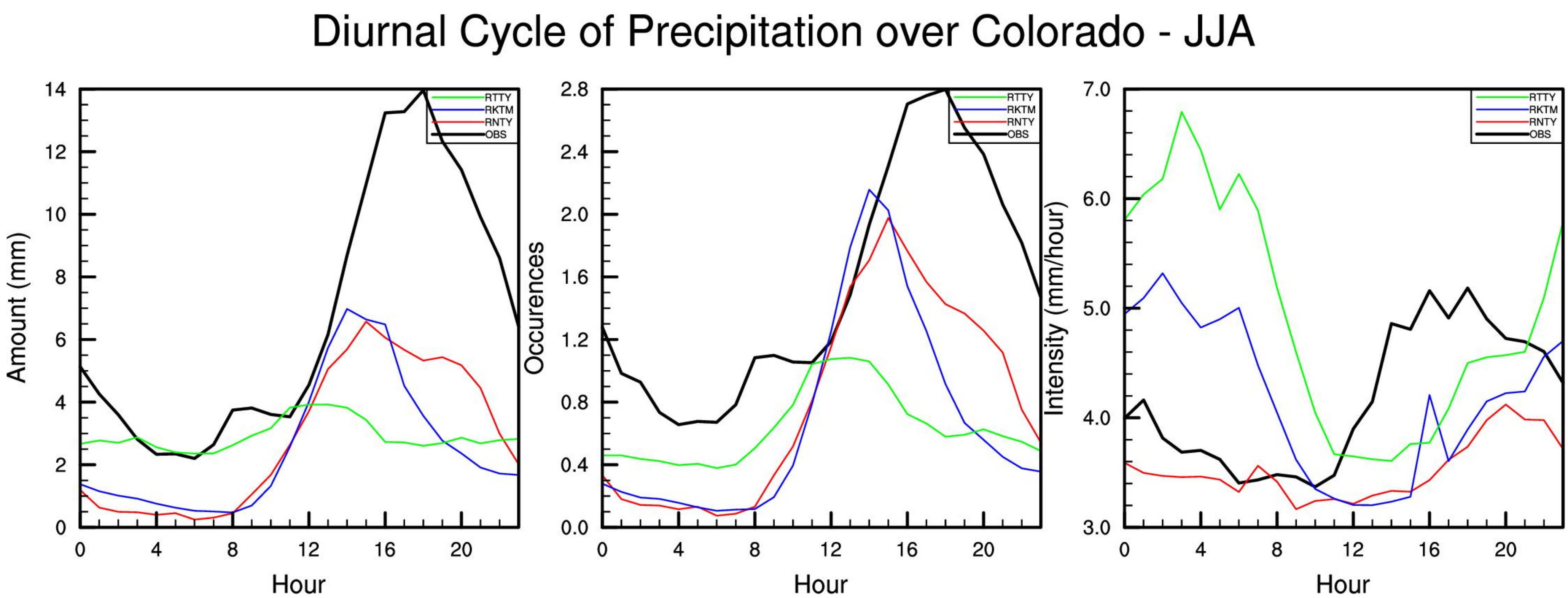
```
t2maxmon = new((/11,12,266,
t2maxjan = t2max_co(:,0:30
t2maxmon(:,0,:) = dim_avg_1
                  :
                  :
t2maxdec = t2max_co(:,334:
t2maxmon(0,11,:) = dim_avg_
```



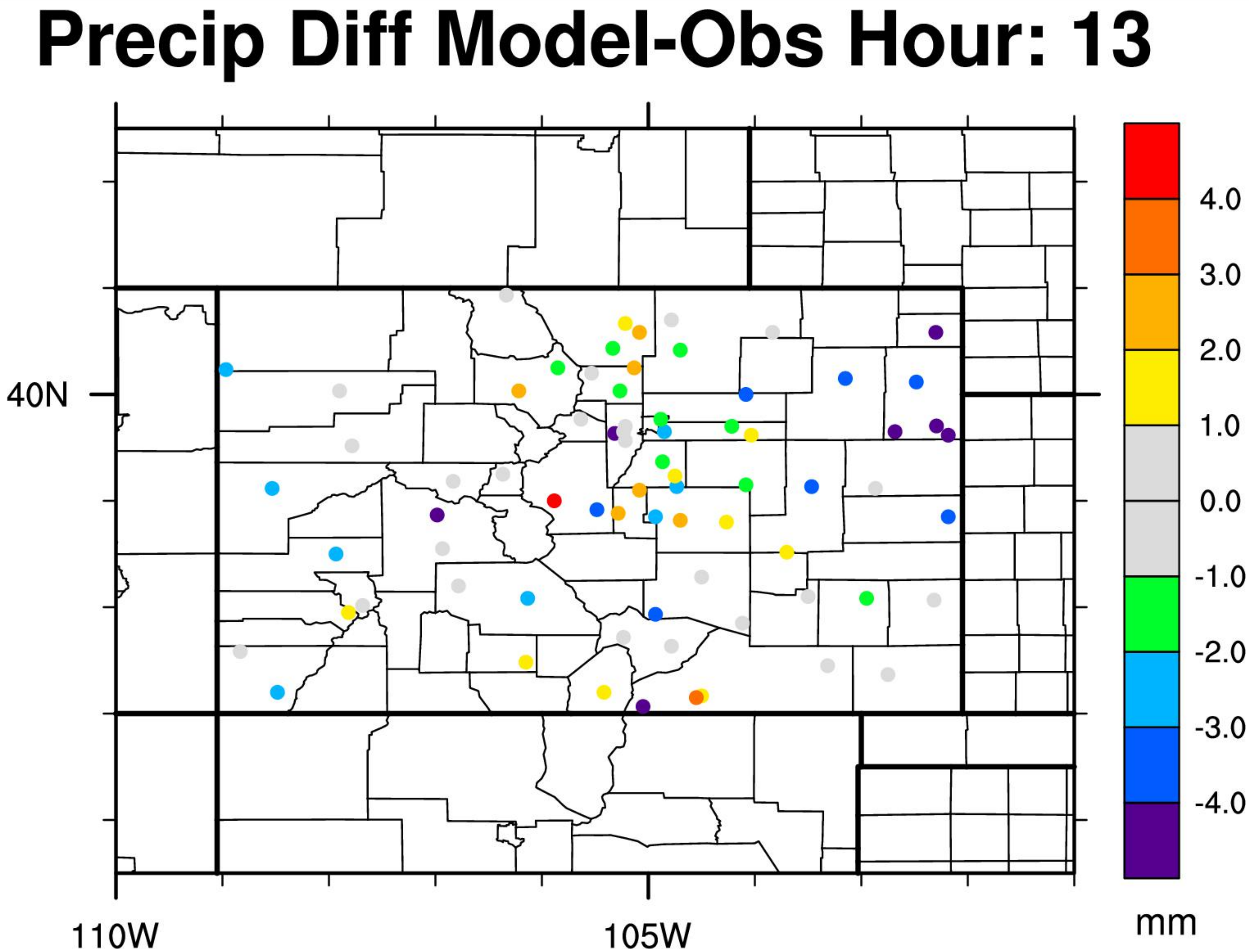
Time Series

- To look at precipitation diurnally we want amount, times of occurrence, and intensity
  - To be counted, precip must be over 2.5mm
- We can either average over a domain, or an area pulled out by shapefiles
- We can also compare by gridpoint/latlon using wrf_user_ll_to_ij
- Make sure of the timezone!
  - Most obs come in at local time
  - Most reanalysis and model runs are UTC

Time Series


Diurnal Cycle of Precipitation over Colorado - JJA
RTTY
RKTM
RNTY
OBS
Amount (mm)
Occurences
Intensity (mm/hour)
Hour
0
4
8
12
16
20
14
12
10
8
6
4
2
0
2.8
2.4
2.0
1.6
1.2
0.8
0.4
0.0
7.0
6.0
5.0
4.0
3.0

Time Series



Precip Diff Model-Obs Hour: 13
40N
110W
105W
4.0
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-4.0
mm

Statistical Methods

Does the model get extremes and spread?

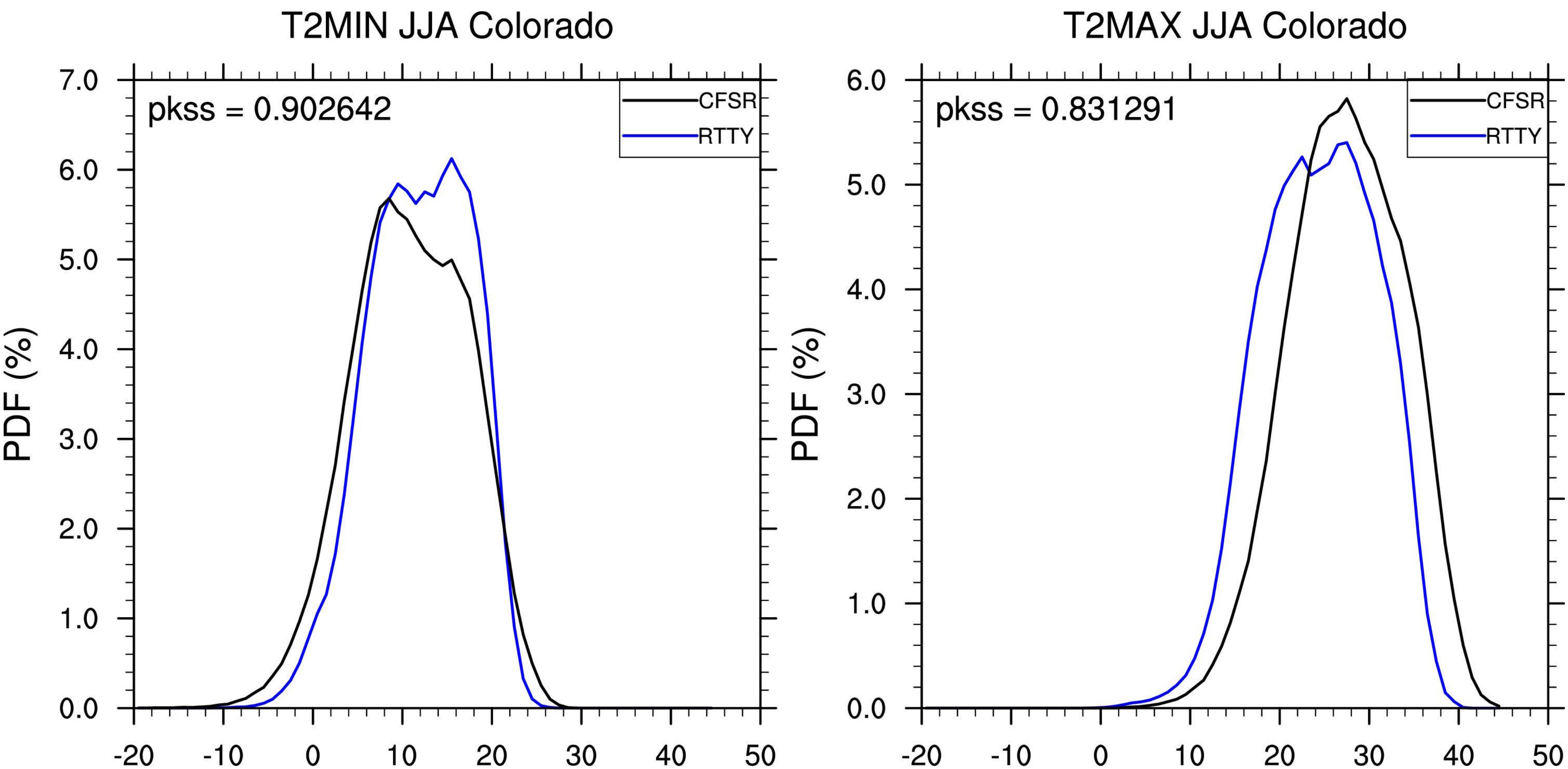
Probability Density Function (PDF)

- A function where the integral across of a variable give the probability that a value of the variable lies within that integral
- Perkins Skill Score
  - Measure of the overlap between observed and modeled PDF
  - Score = 1 -> Perfect simulation!
  - Score = 0 -> Bad simulation!
  - To calculate, use the binned data from the PDFs.
  - Get the minimum value of the model and reference data and sum the values.

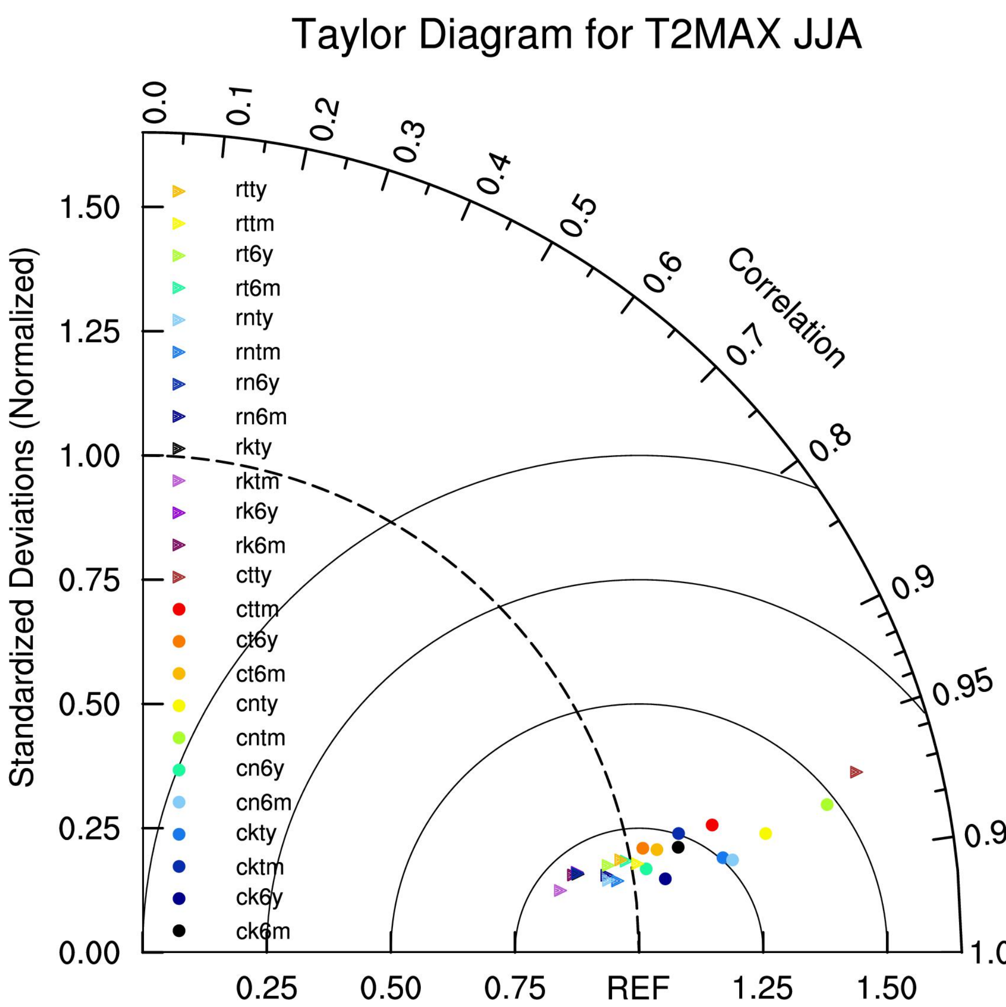
Probability Density Function (PDF)

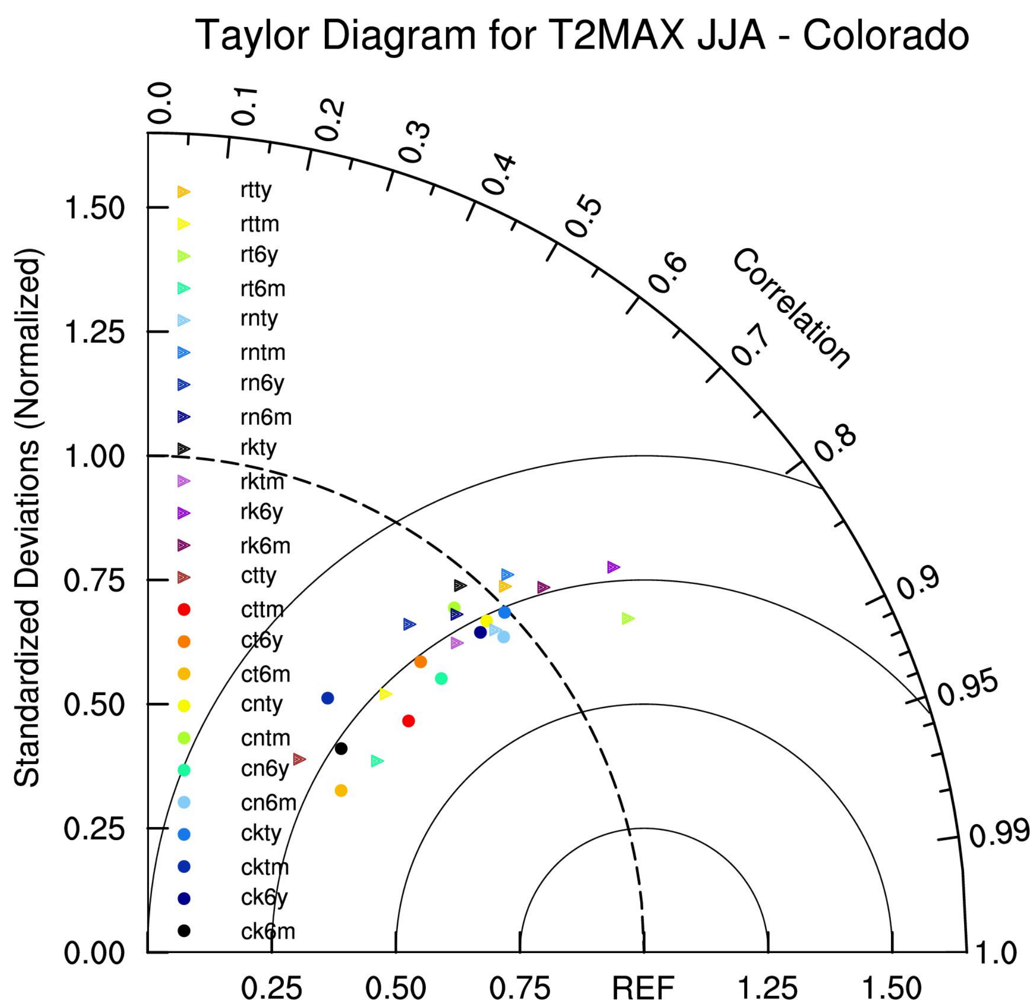
```
pdf_mod = pdfx(mod,nBin,True)
pdf_obs = pdfx(obs,nBin,True)

; Perkins Skill Score
t2_perkins = new((/nBin/),float)
do i=0,nBin-1
  t2_perkins_temp = new((/2/),float)
  t2_perkins_temp(0) = pdf_mod(i)/100
  t2_perkins_temp(1) = pdf_obs(i)/100
  t2_perkins(i) = min(t2_perkins_temp)
  delete(t2_perkins_temp)
end do
; Sum the minimum bins to get the PSS
sum_t2_perkins = sum(t2_perkins)
; Print Perkins Skill Score to screen
print("PKSS = " +flt2string(sum_t2_perkins))
```

Probability Density Function (PDF)



Taylor Diagram





Feature Tracking

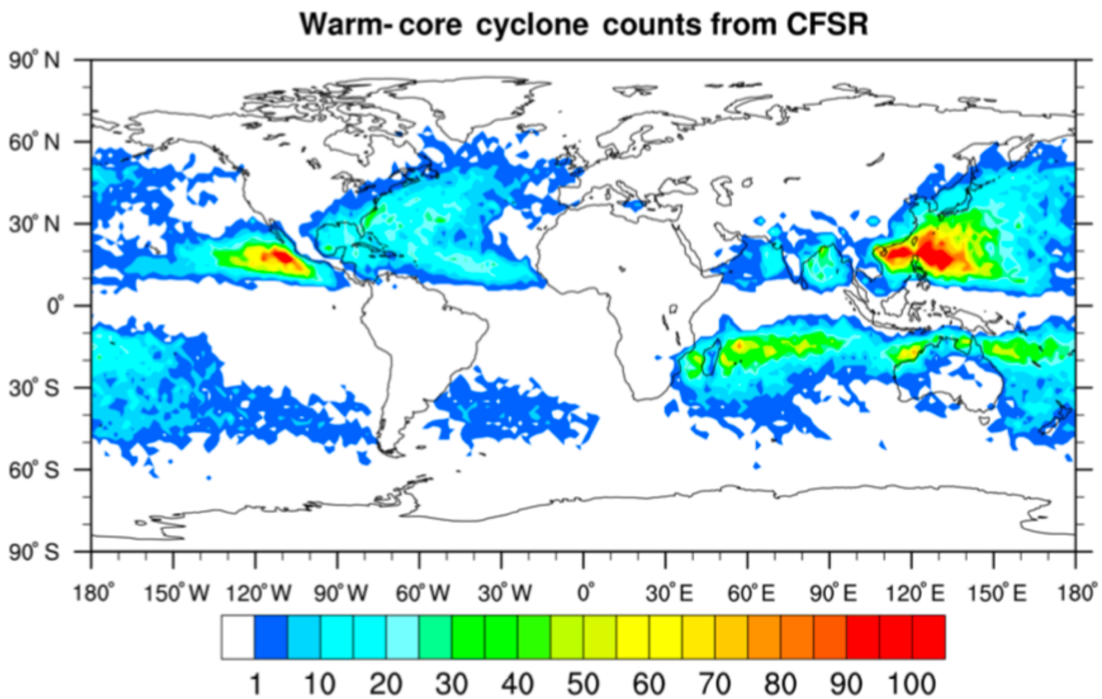
An automated feature tracker can be very useful to see if your model data has extremes such as tropical storms and strong winter storms

TempestExtremes

- **TempestExtremes** is an open-source software framework for automated feature tracking that can be used on a wide array of structured or unstructured grids
- Pointwise tracking can track:
  - Tropical cyclones
  - Extra-tropical cyclones
  - Easterly waves

TempestExtremes

- What data do we need?
  - PSL, USFC, VSFC, U850, V850, T400, Z300, Z500
  - Ideally, we want to pull out only these variables from the model data



Ullrich and Zarzycki 2017

1. Locates candidate points of interest of local extremes
2. Eliminates candidate points that do not satisfy a prescribed set of thresholds
3. Connects candidate points in time
4. Eliminates tracks based on a duration criteria

Summary

- Post-Processing
  - Extract data for location
  - Reading different format data
  - Shapefile
- Evaluation
  - Annual and Diurnal Cycles
  - PDFs and Perkins Skill Score
  - Regridding
  - Spatial evaluation
  - Temporal evaluation
  - Feature tracking