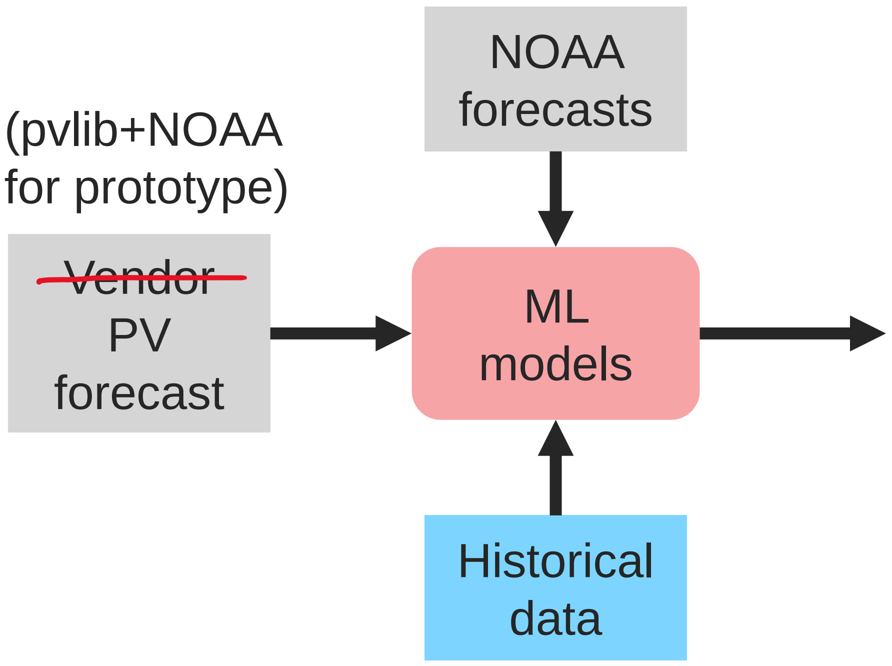
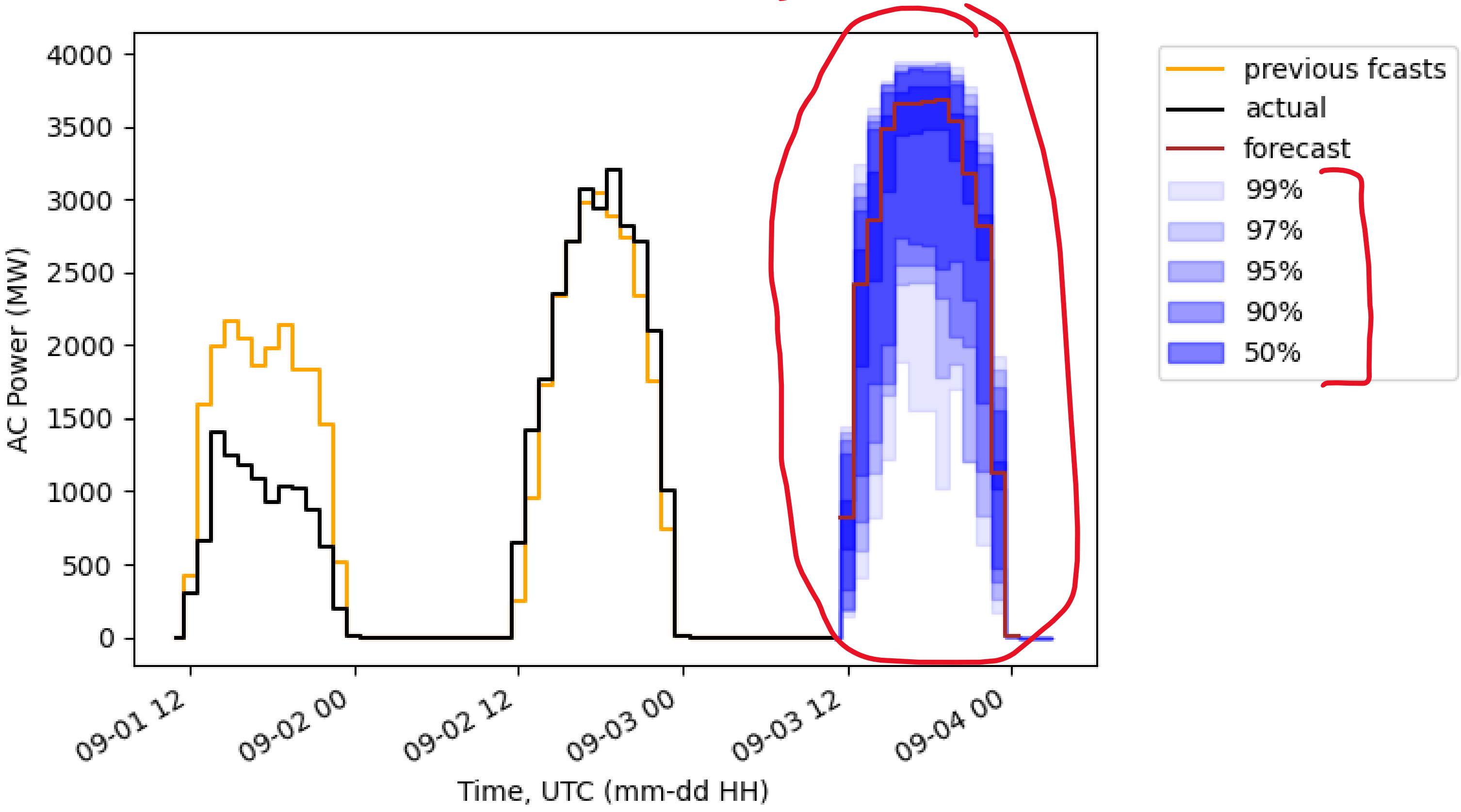
Making Regional Probabilistic Solar Forecasts from Deterministic Forecasts with Open-Source Tools and Data

William B. Hobbs, PE
Southern Company R&D
IEEE PVSC 2024

Overview

- Open-source tool to produce probabilistic forecast from an existing deterministic forecast (e.g., vendor)
- Focus on day-ahead for now





Motivation

- Lots of good deterministic forecasts are available
- None are perfect
- Grid operators have increasing need for uncertainty information
- Most probabilistic forecasts focus on single sites
- Errors across sites are not independent → can’t simply combine site-level probabilistic forecasts

Tools and data

- All in python, open-source (and free) tools
- `pvlib` for weather-to-power (forecast and synthetic actuals)
- `Herbie` to retrieve NOAA NWPs from AWS
  - A portion of this work used code generously provided by Brian Blaylock's Herbie python package (Version 2024.3.1) (https://doi.org/10.5281/zenodo.4567540)
- scikit-learn (`sklearn`) and `quantile-forest` (based on `sklearn`, published by Zillow) for ML regression models

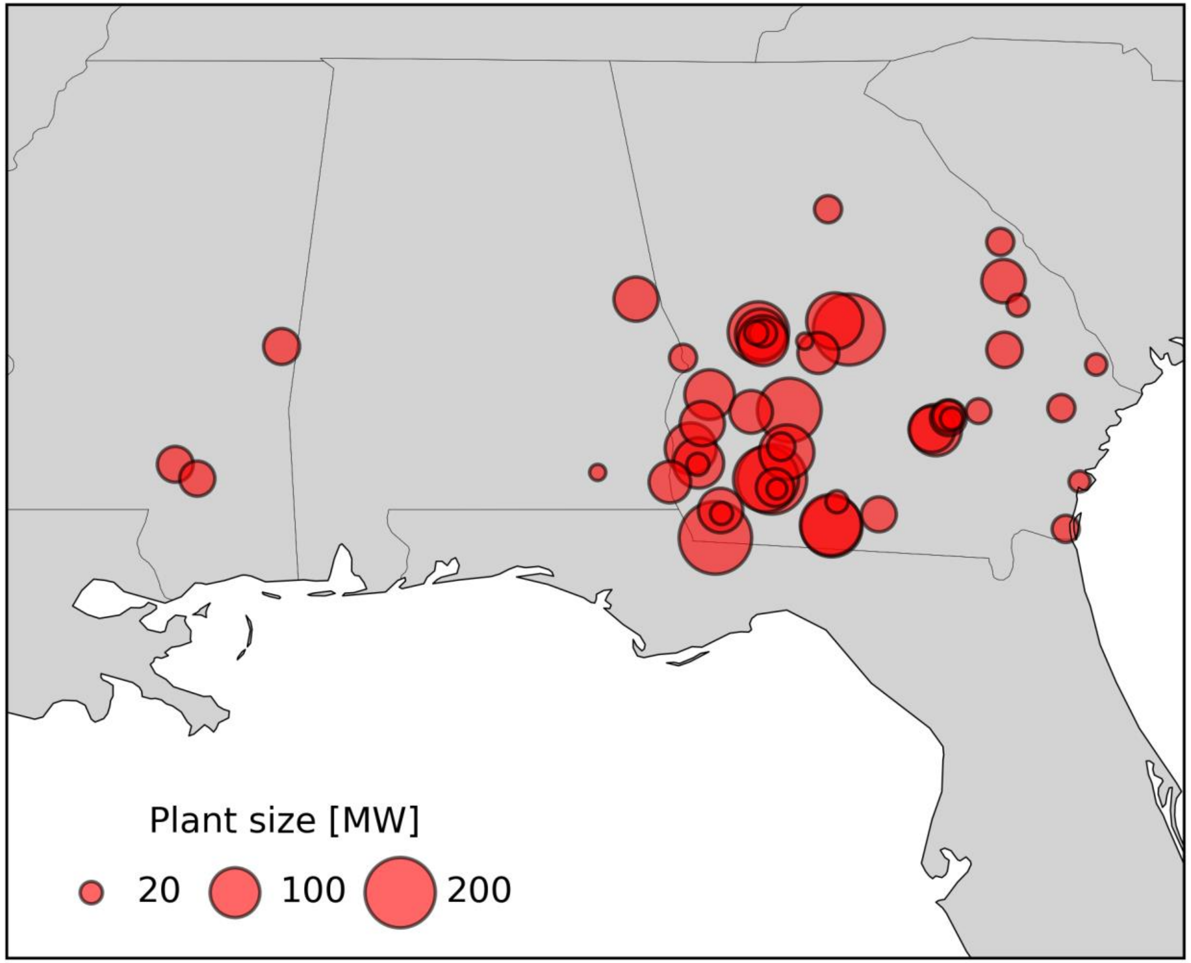
- Forecasted weather: NOAA GEFS and HRRR
- "Actual" weather: NOAA HRRR* f00
- Plant specs: EIA Form 860 data**

*Started with NREL NSRDB, but found an issue with cloud detection at low sun angles. Results were mostly the same, otherwise.

** All of SOCO BA

Solar Fleet

- About 3700 MW in Southern Company's Balancing Area



(Improved map by Adam Jensen)

Found two interesting indicators of forecast uncertainty:

1. **Ensemble spread for cloud cover**:
    - Variation in cloud cover across 30 GEFS ensemble members
        - *For each member, calculate average cloud cover over all solar plants, weighted by plant capacity, then calculate standard deviation of those averages across all 30 ensemble members*
2. **Spatial variation in irradiance**:
    - Variation of irradiance (and resulting calculated fleet power) across ~30km x 30km HRRR grid (10x10 native grid cells)
        - *(Max – Min)/Clear Sky*

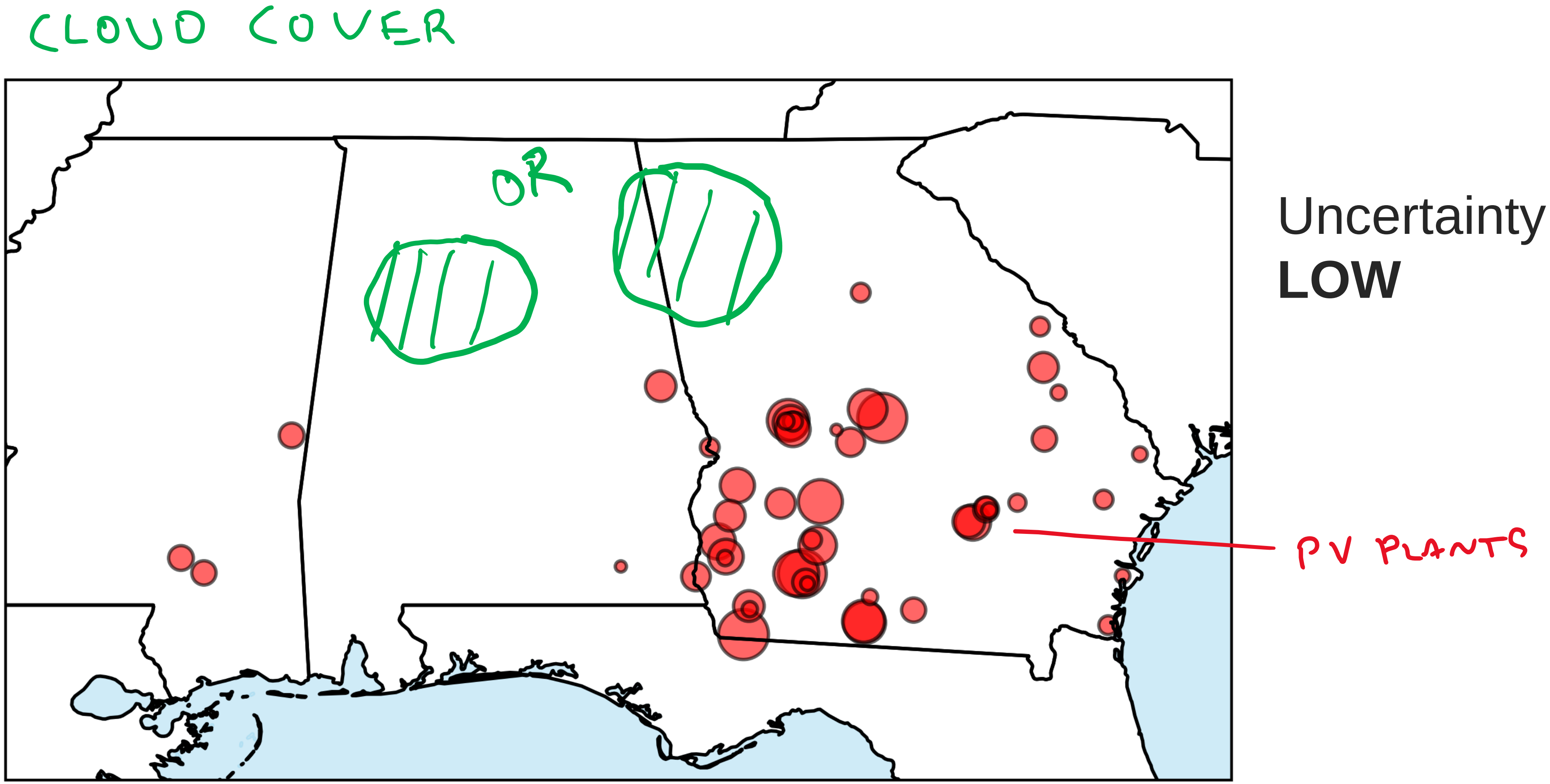
Timeout!

Why not just use GEFS for 30 power scenarios and calculate probabilities from that??

Great question! A few reasons:

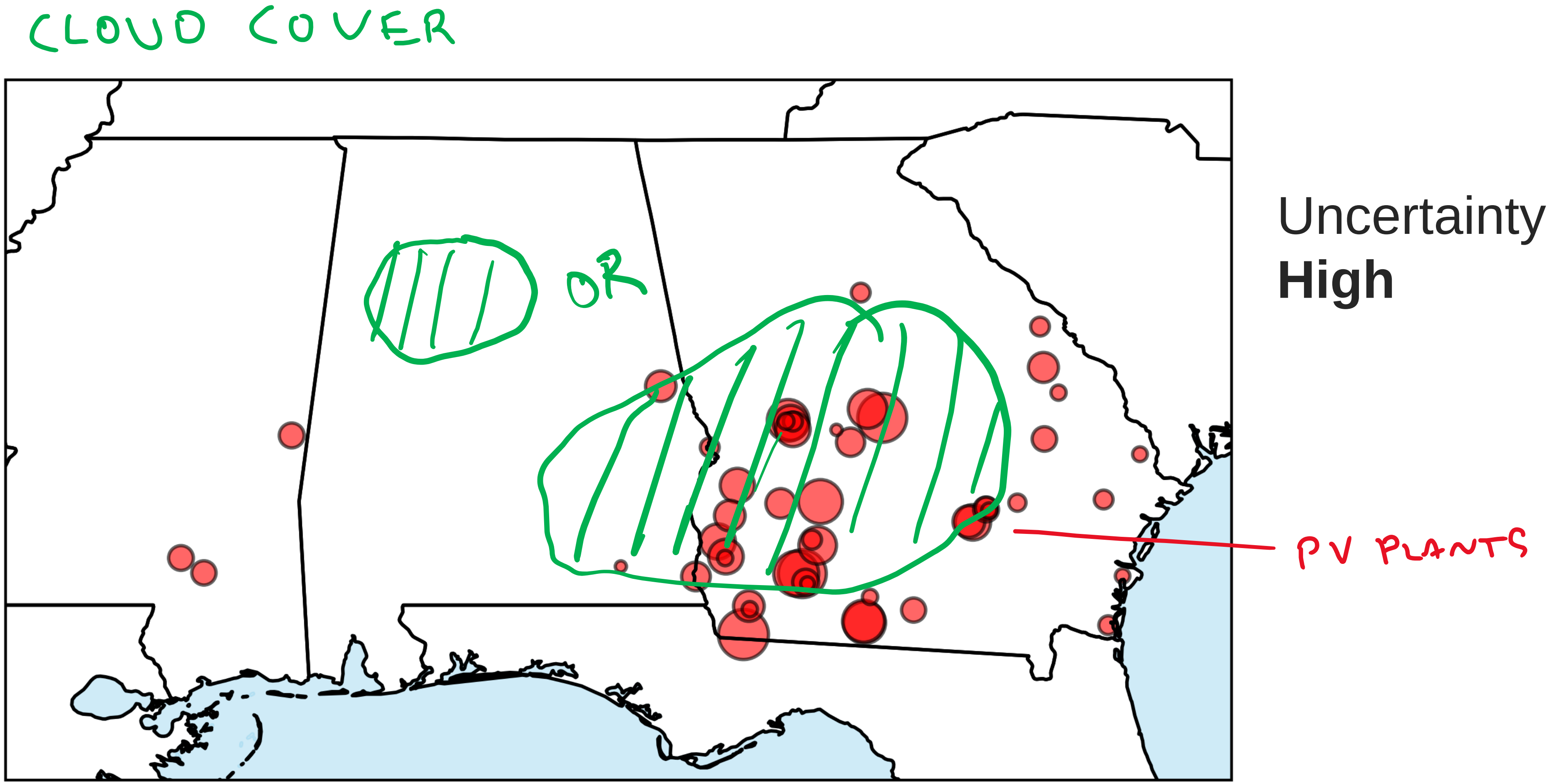
1. Ensembles are generally “underdispursed” (overconfident) for solar, i.e., what actually happens too often falls outside the range of all ensemble members
2. The GEFS model by itself may not be *great* for solar
3. I’m not trying to build a full forecast (hard to compete with the best vendors), I want to add probabilities to an *existing* good forecast

Cloud Cover Uncertainty Examples:



Uncertainty:
LOW

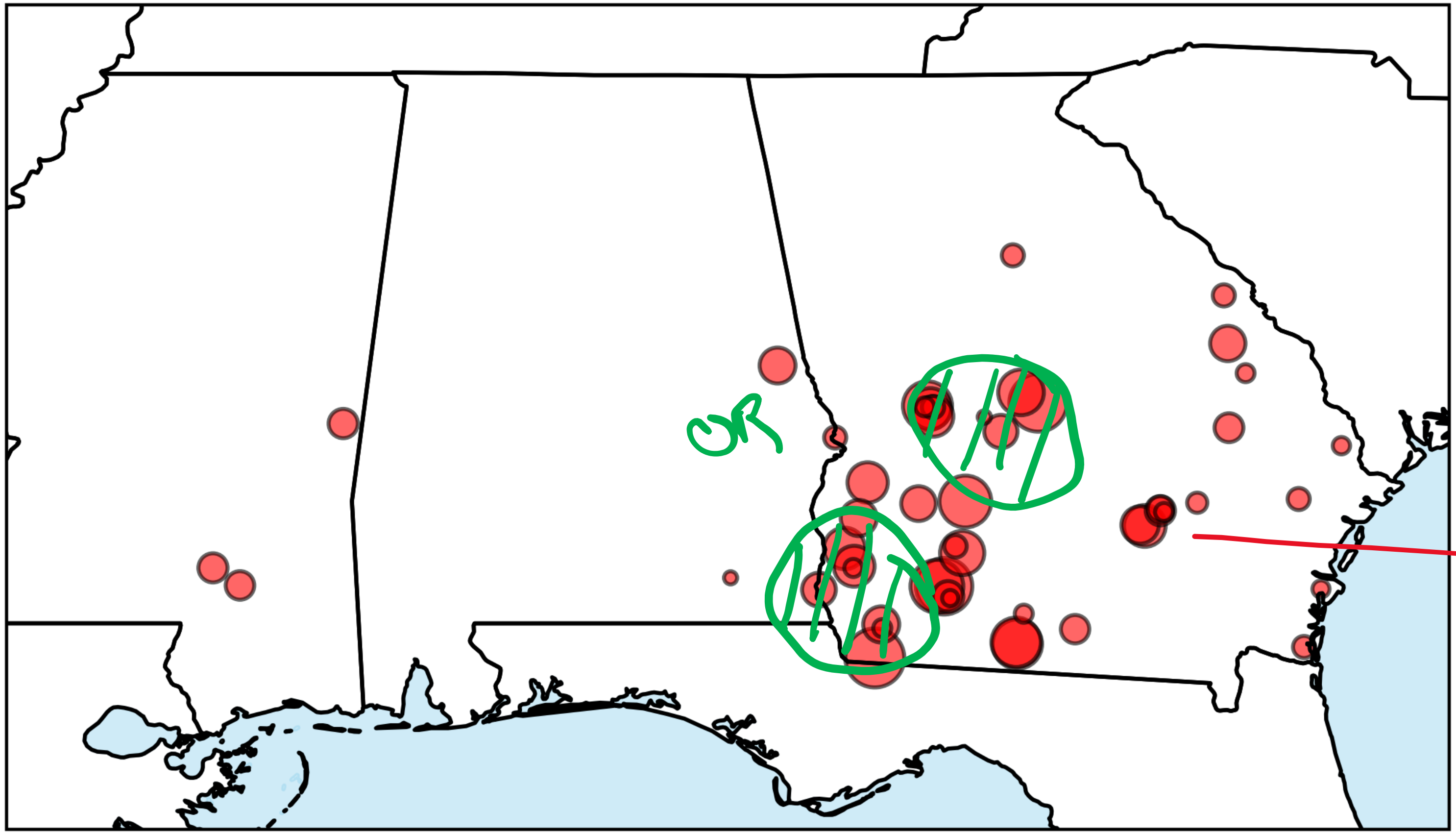
Cloud Cover Uncertainty Examples:



Uncertainty:
High

Cloud Cover Uncertainty Examples:

CLOUD COVER



PV PLANTS

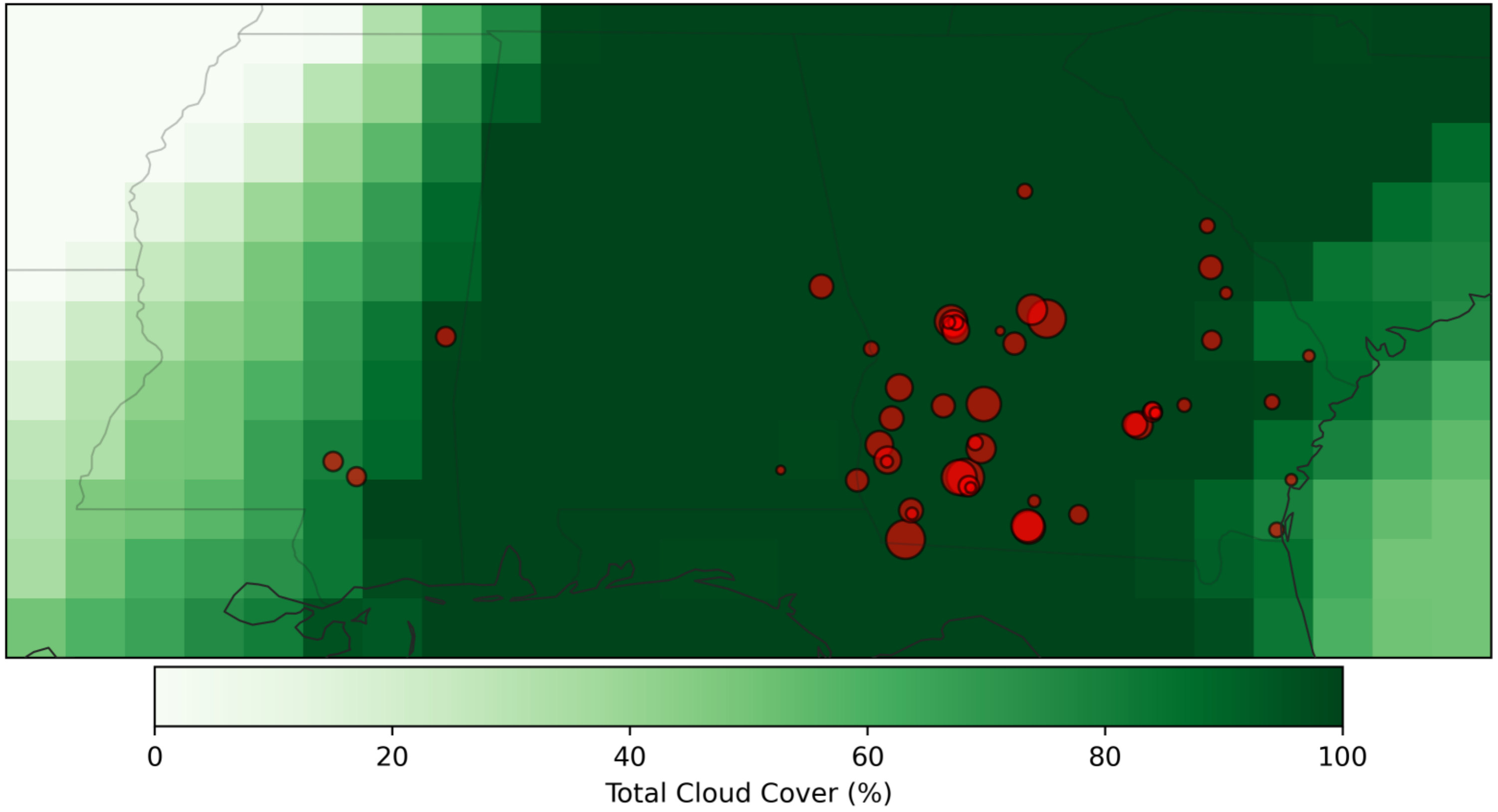
Uncertainty:
Low/Medium

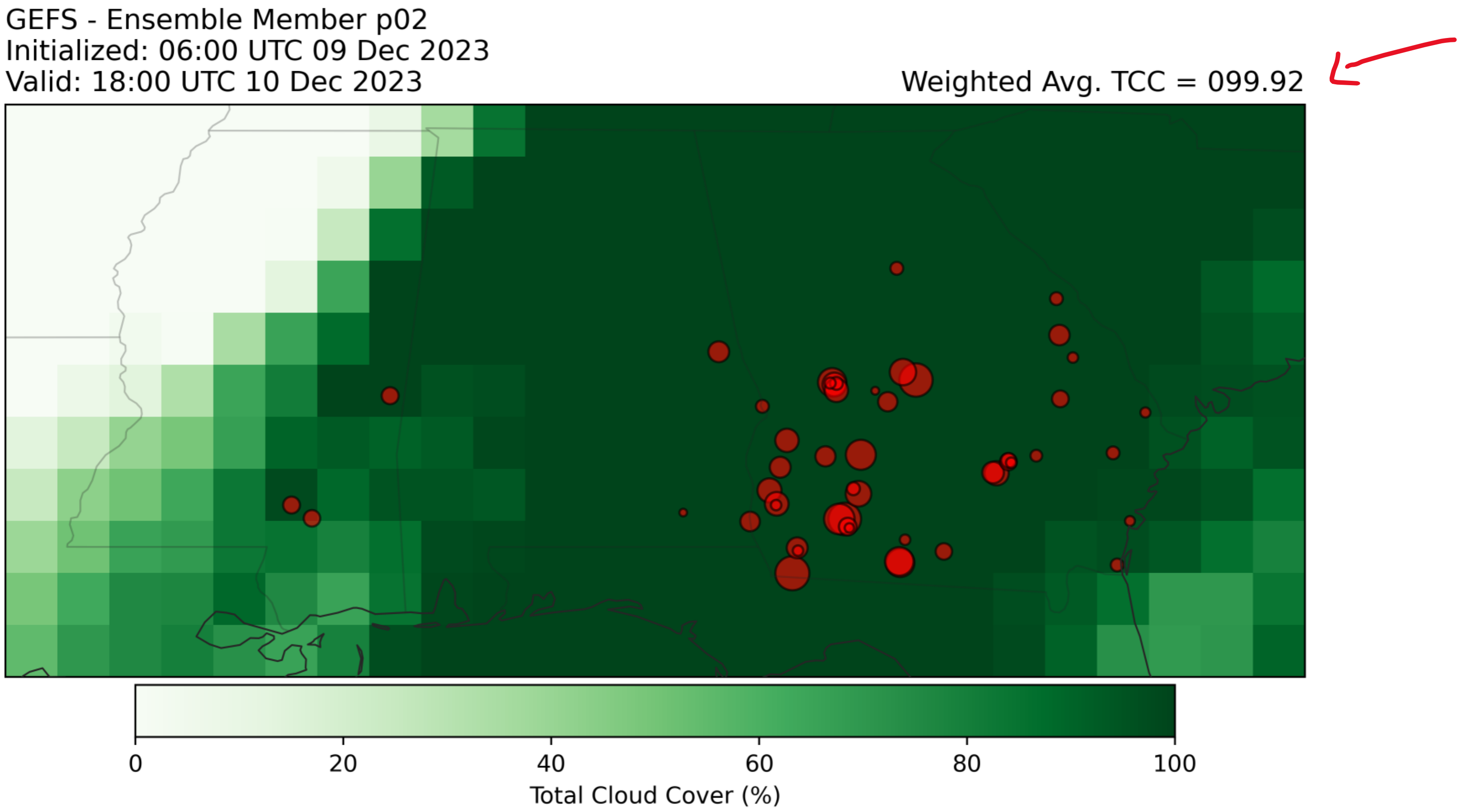
Real examples:

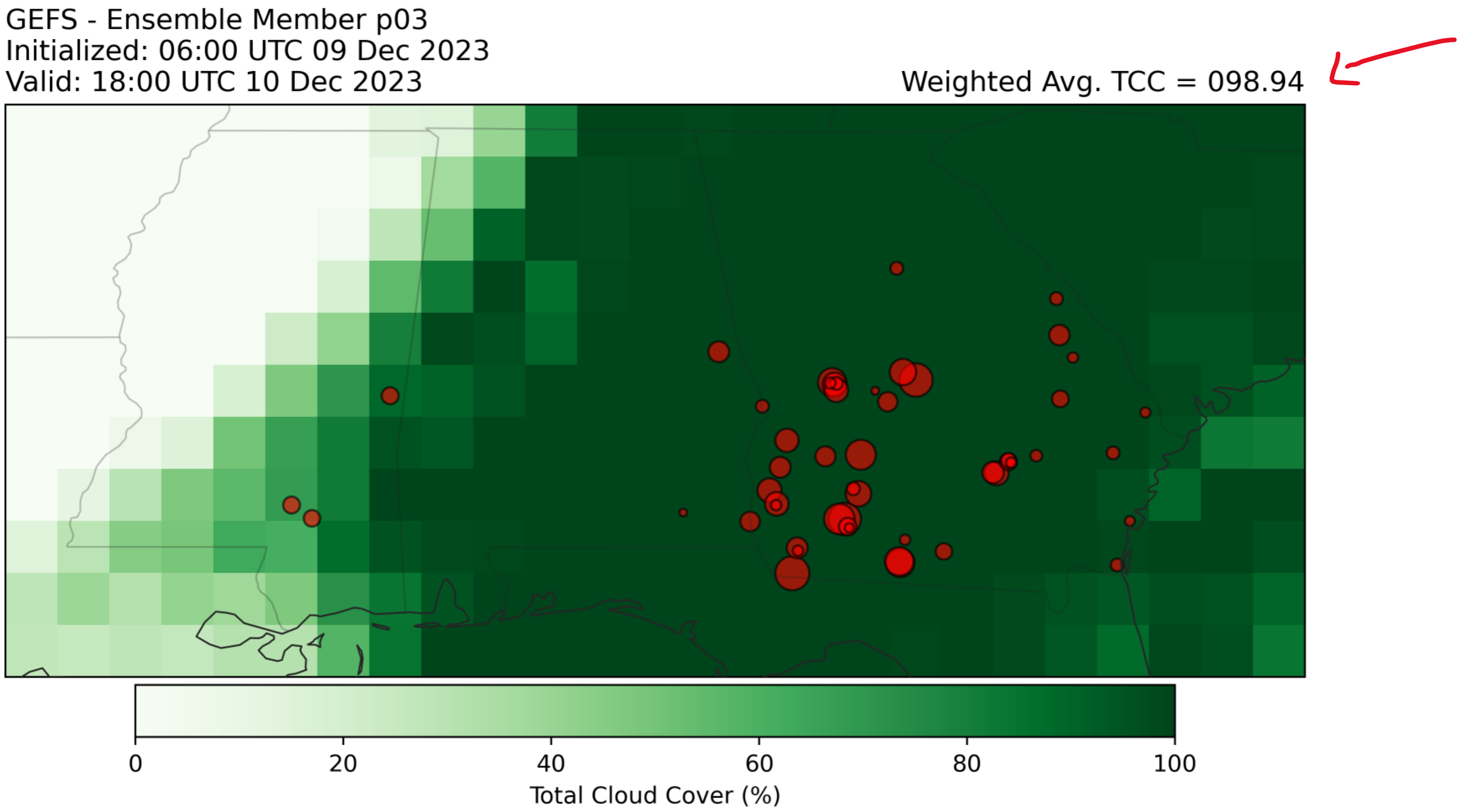
- Low solar, low uncertainty day
- (4 ensemble members shown, full analysis includes all 30 members)

GEFS - Ensemble Member p01
Initialized: 06:00 UTC 09 Dec 2023
Valid: 18:00 UTC 10 Dec 2023

Weighted Avg. TCC = 099.11



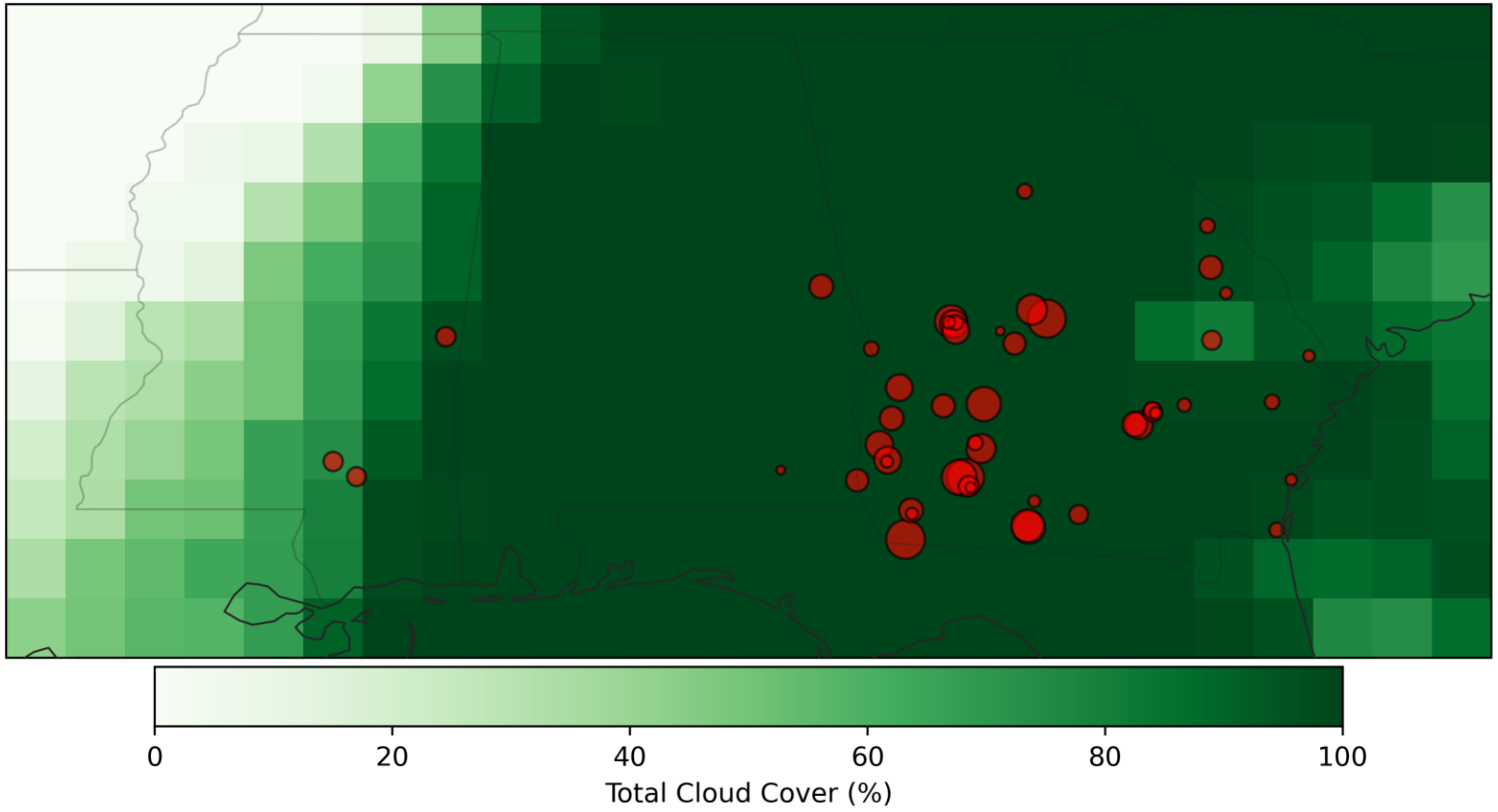

GEFS - Ensemble Member p02
Initialized: 06:00 UTC 09 Dec 2023
Valid: 18:00 UTC 10 Dec 2023
Weighted Avg. TCC = 099.92
0
20
40
60
80
100
Total Cloud Cover (%)


GEFS - Ensemble Member p03
Initialized: 06:00 UTC 09 Dec 2023
Valid: 18:00 UTC 10 Dec 2023
Weighted Avg. TCC = 098.94
0
20
40
60
80
100
Total Cloud Cover (%)

GEFS - Ensemble Member p04
Initialized: 06:00 UTC 09 Dec 2023
Valid: 18:00 UTC 10 Dec 2023

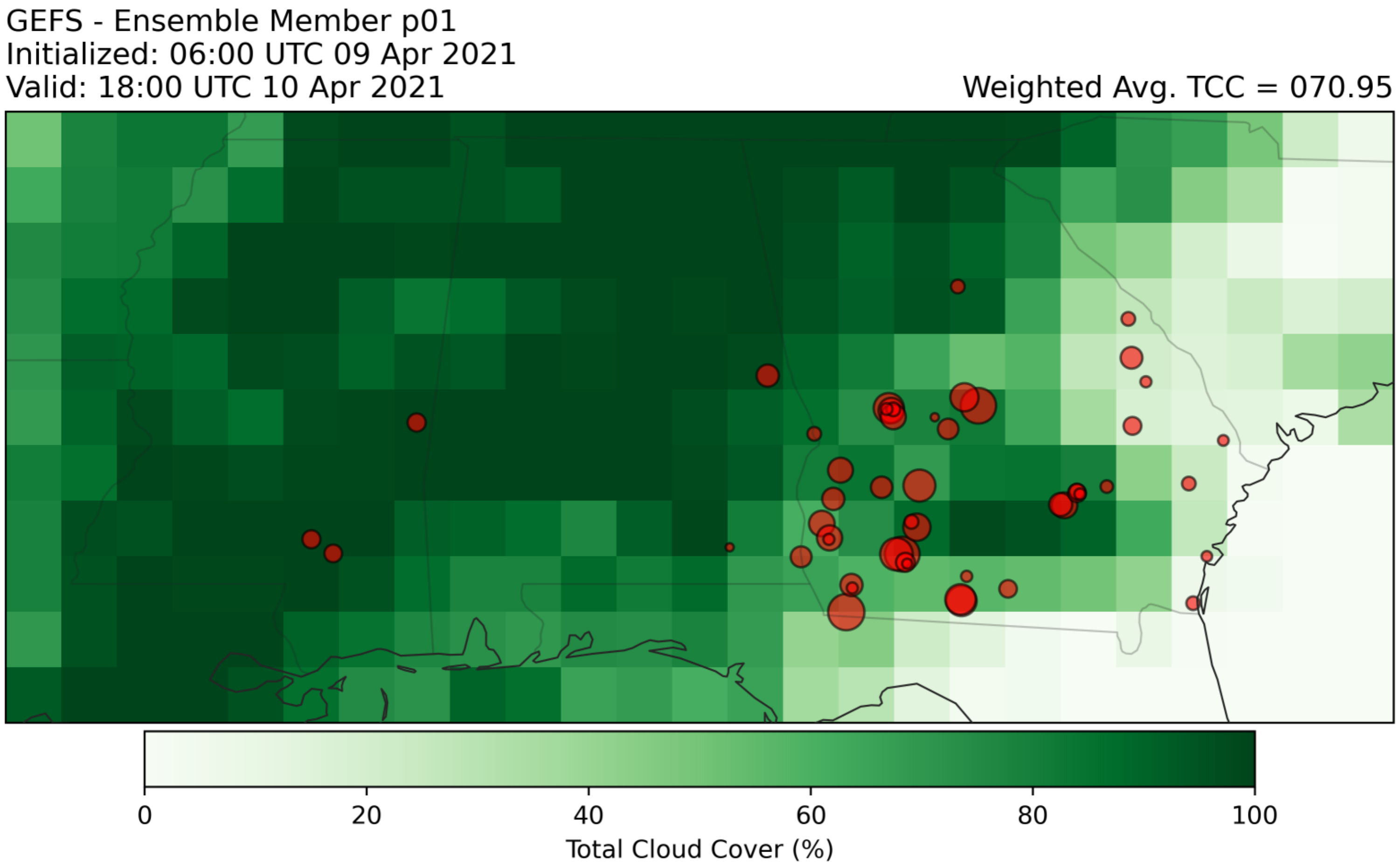
Weighted Avg. TCC = 098.82

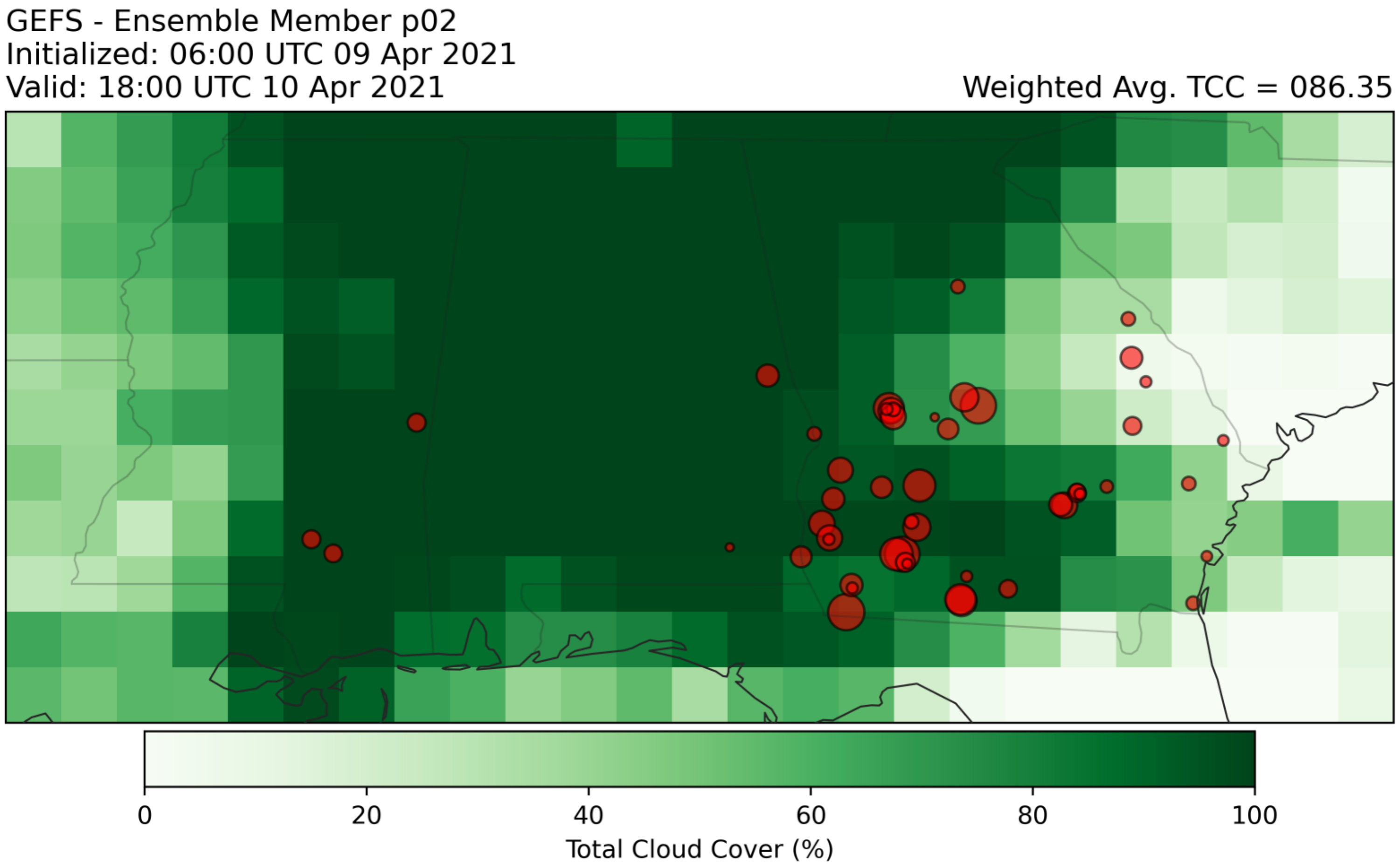
STD ≅ 1.0

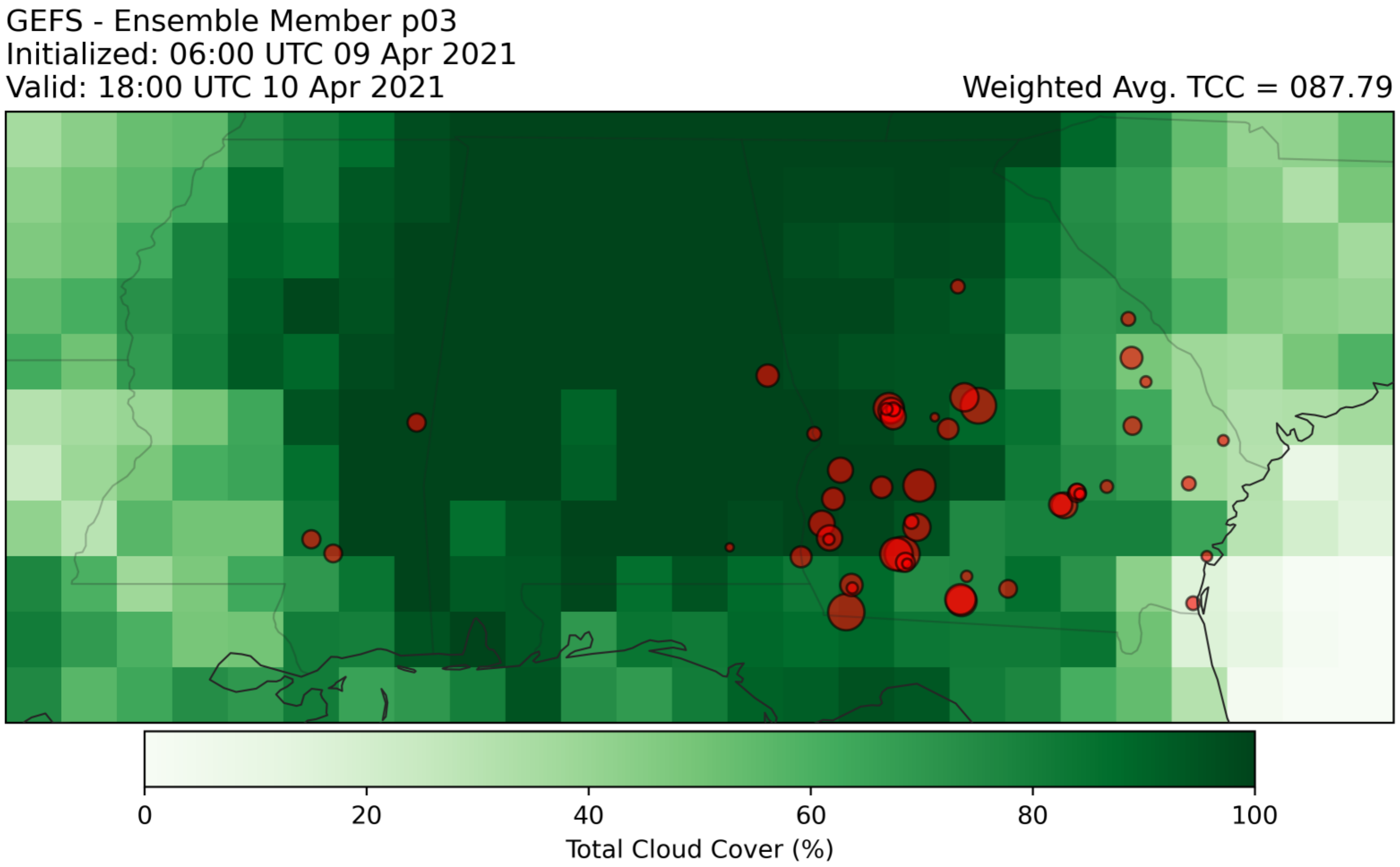


Real examples:

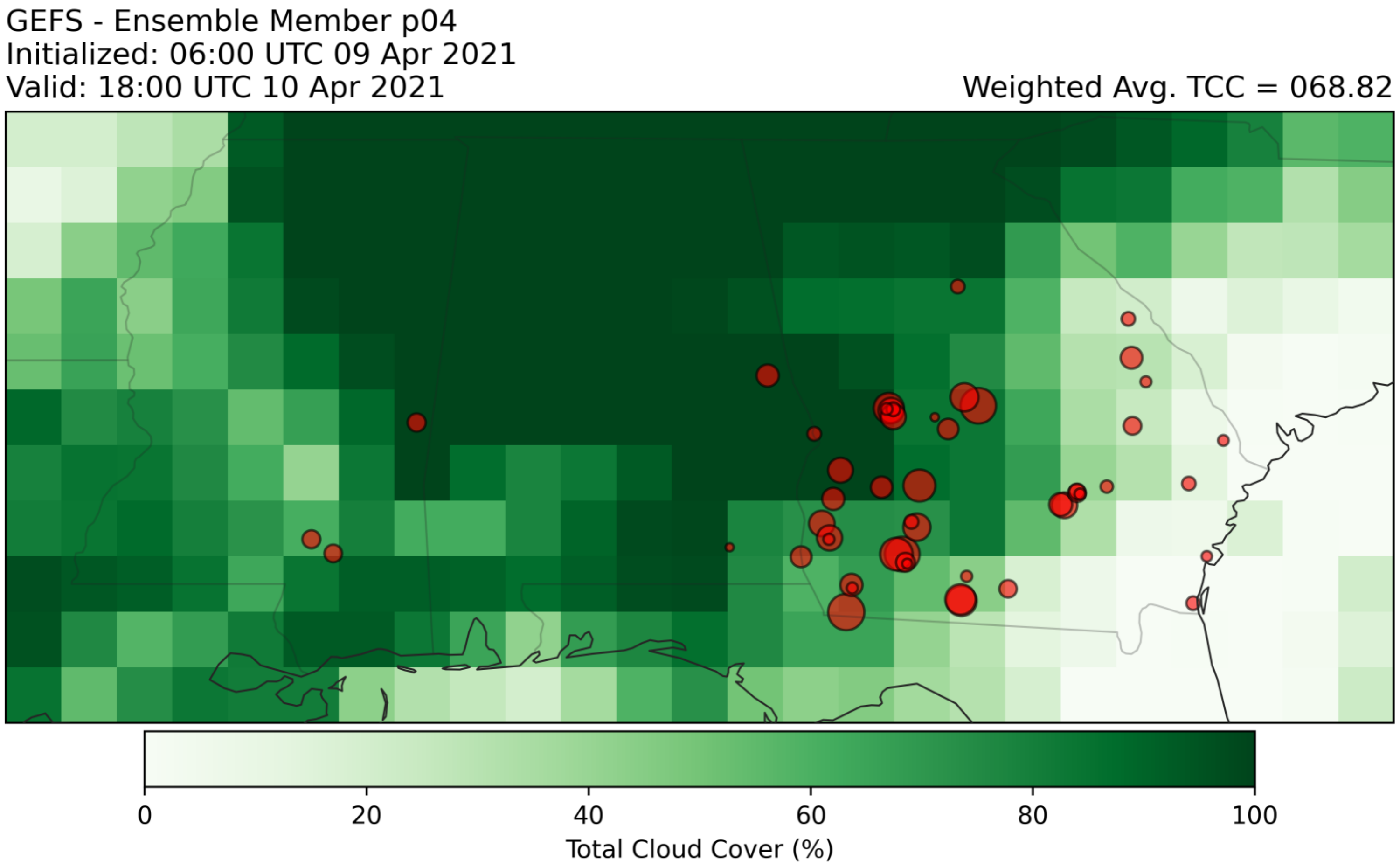
- Low solar, high uncertainty day
- (4 ensemble members shown, full analysis includes all 30 members)


GEFS - Ensemble Member p01
Initialized: 06:00 UTC 09 Apr 2021
Valid: 18:00 UTC 10 Apr 2021
Weighted Avg. TCC = 070.95
0
20
40
60
80
100
Total Cloud Cover (%)


GEFS - Ensemble Member p02
Initialized: 06:00 UTC 09 Apr 2021
Valid: 18:00 UTC 10 Apr 2021
Weighted Avg. TCC = 086.35
0
20
40
60
80
100
Total Cloud Cover (%)



GEFS - Ensemble Member p03
Initialized: 06:00 UTC 09 Apr 2021
Valid: 18:00 UTC 10 Apr 2021
Weighted Avg. TCC = 087.79
0
20
40
60
80
100
Total Cloud Cover (%)

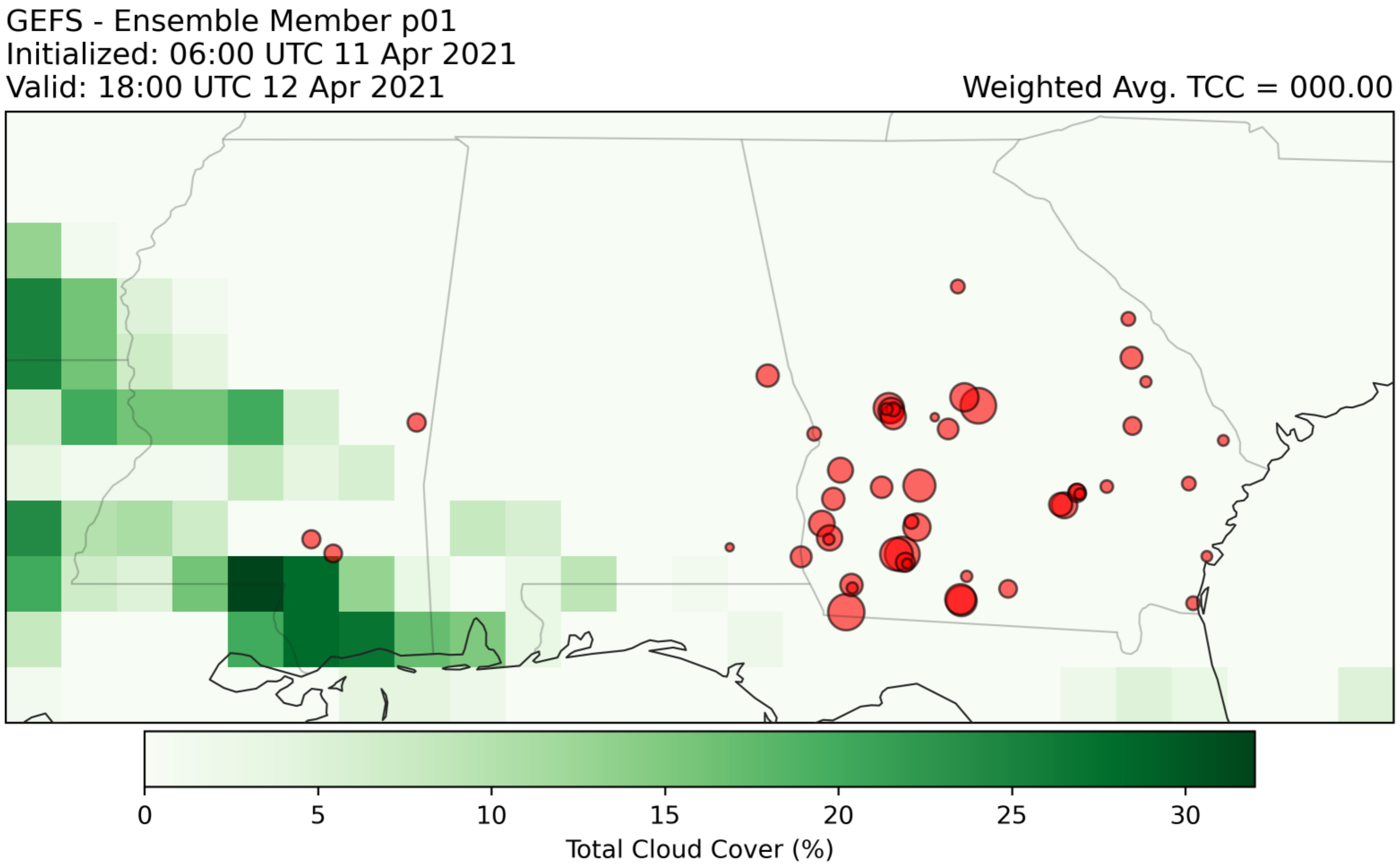


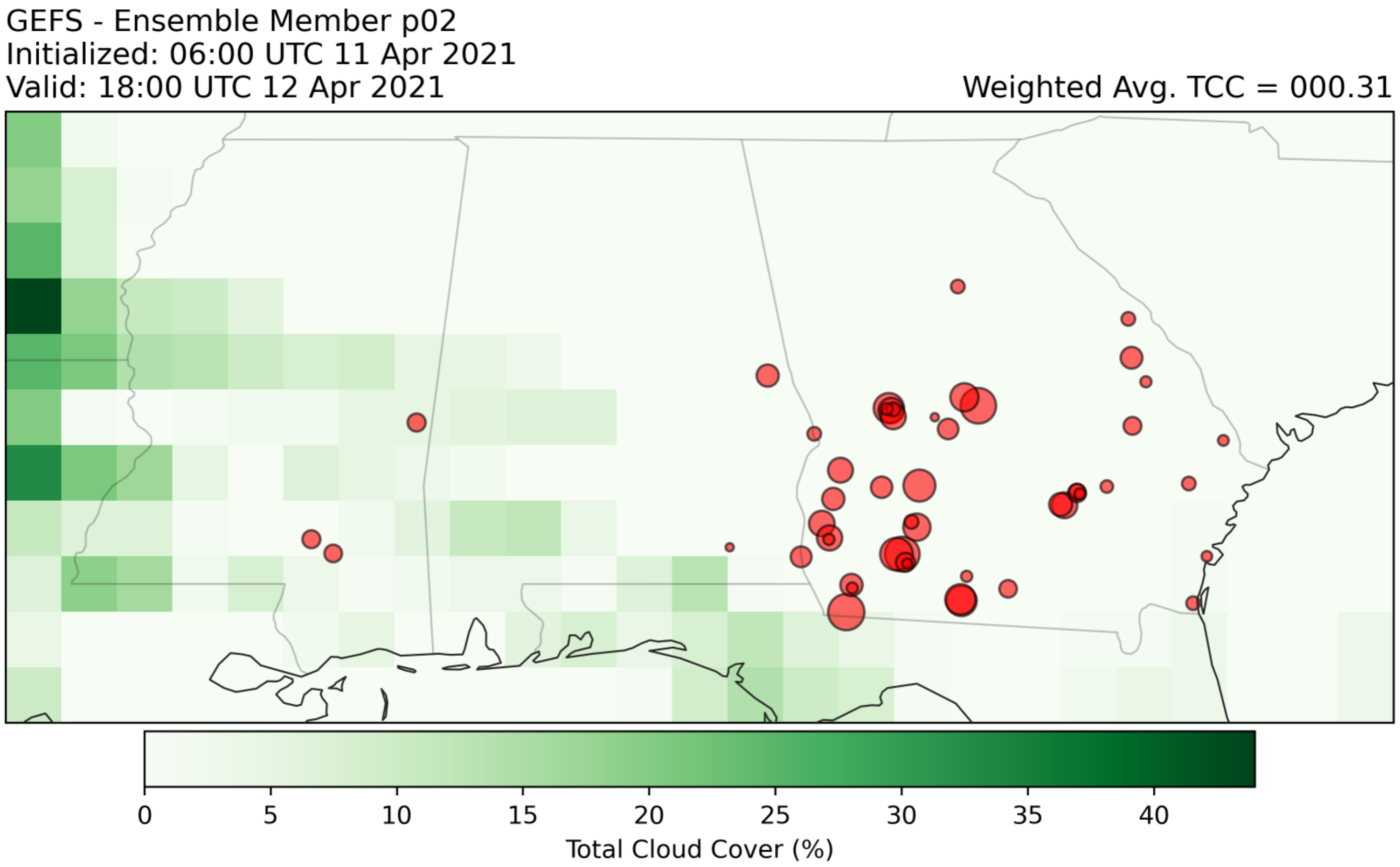
GEFS - Ensemble Member p04
Initialized: 06:00 UTC 09 Apr 2021
Valid: 18:00 UTC 10 Apr 2021
Weighted Avg. TCC = 068.82
0
20
40
60
80
100
Total Cloud Cover (%)

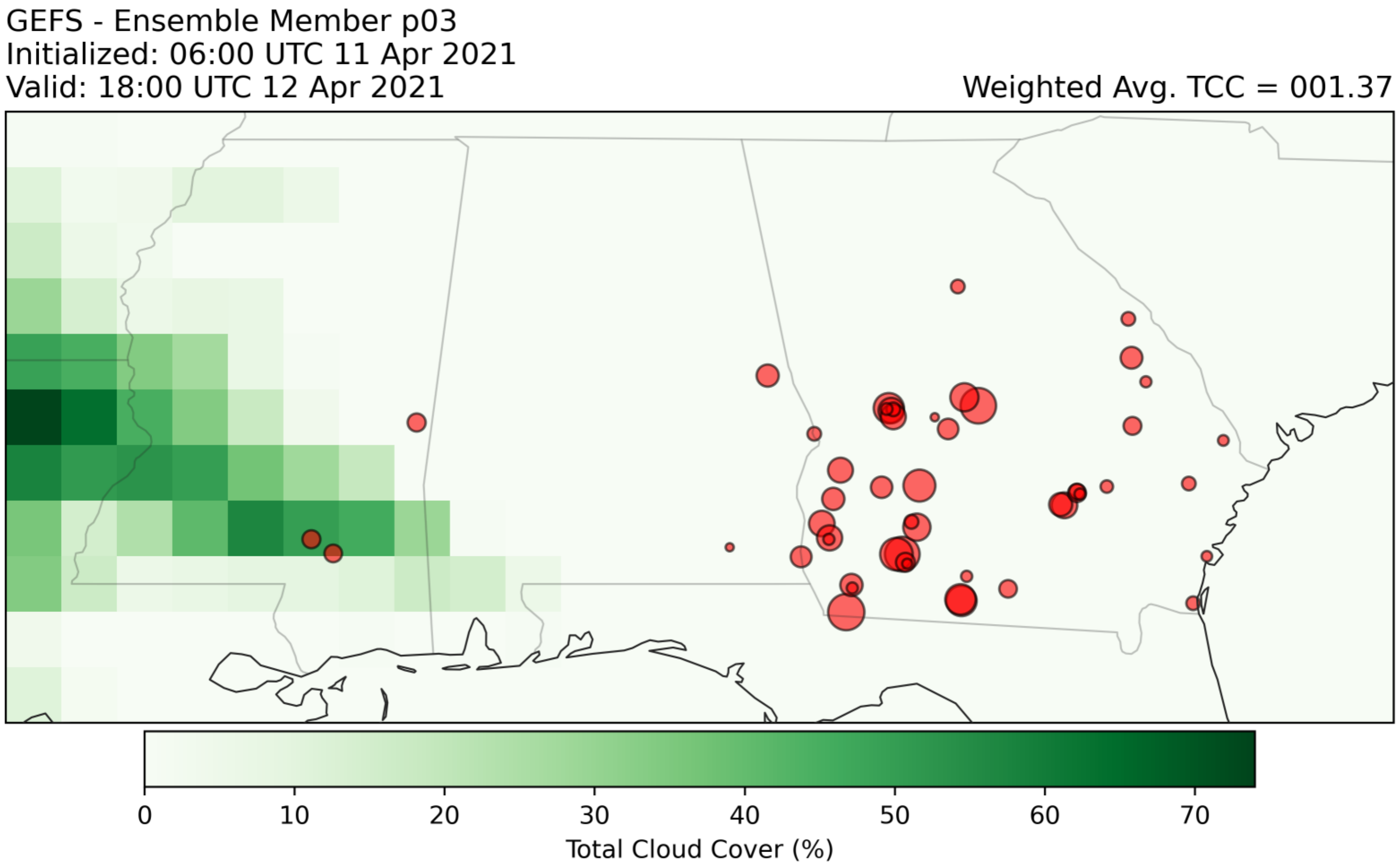
STD ≅ 10

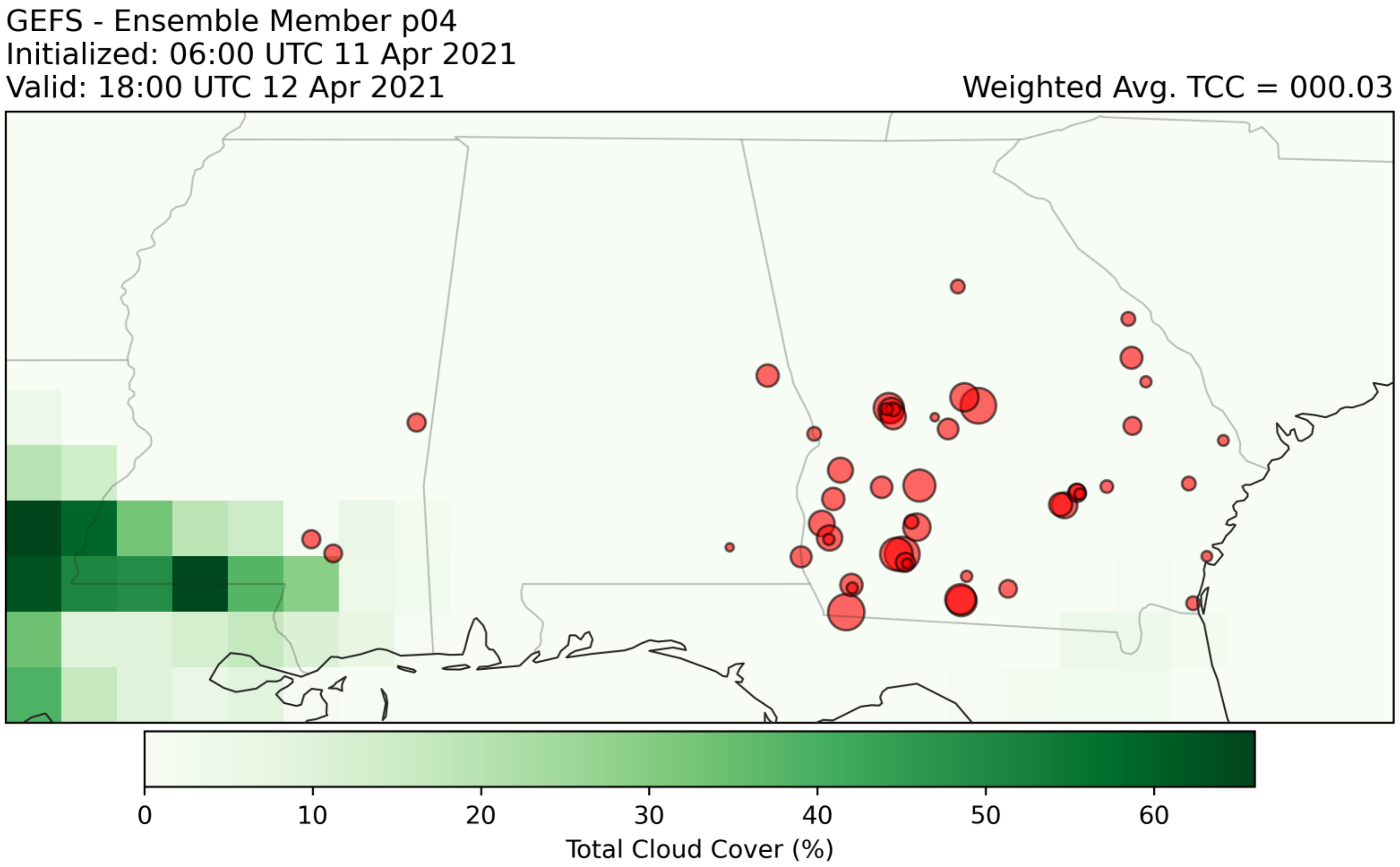
Real examples:

- High solar, low uncertainty day
- (4 ensemble members shown, full analysis includes all 30 members)


GEFS - Ensemble Member p01
Initialized: 06:00 UTC 11 Apr 2021
Valid: 18:00 UTC 12 Apr 2021
Weighted Avg. TCC = 000.00
0
5
10
15
20
25
30
Total Cloud Cover (%)


GEFS - Ensemble Member p02
Initialized: 06:00 UTC 11 Apr 2021
Valid: 18:00 UTC 12 Apr 2021
Weighted Avg. TCC = 000.31
0
5
10
15
20
25
30
35
40
Total Cloud Cover (%)

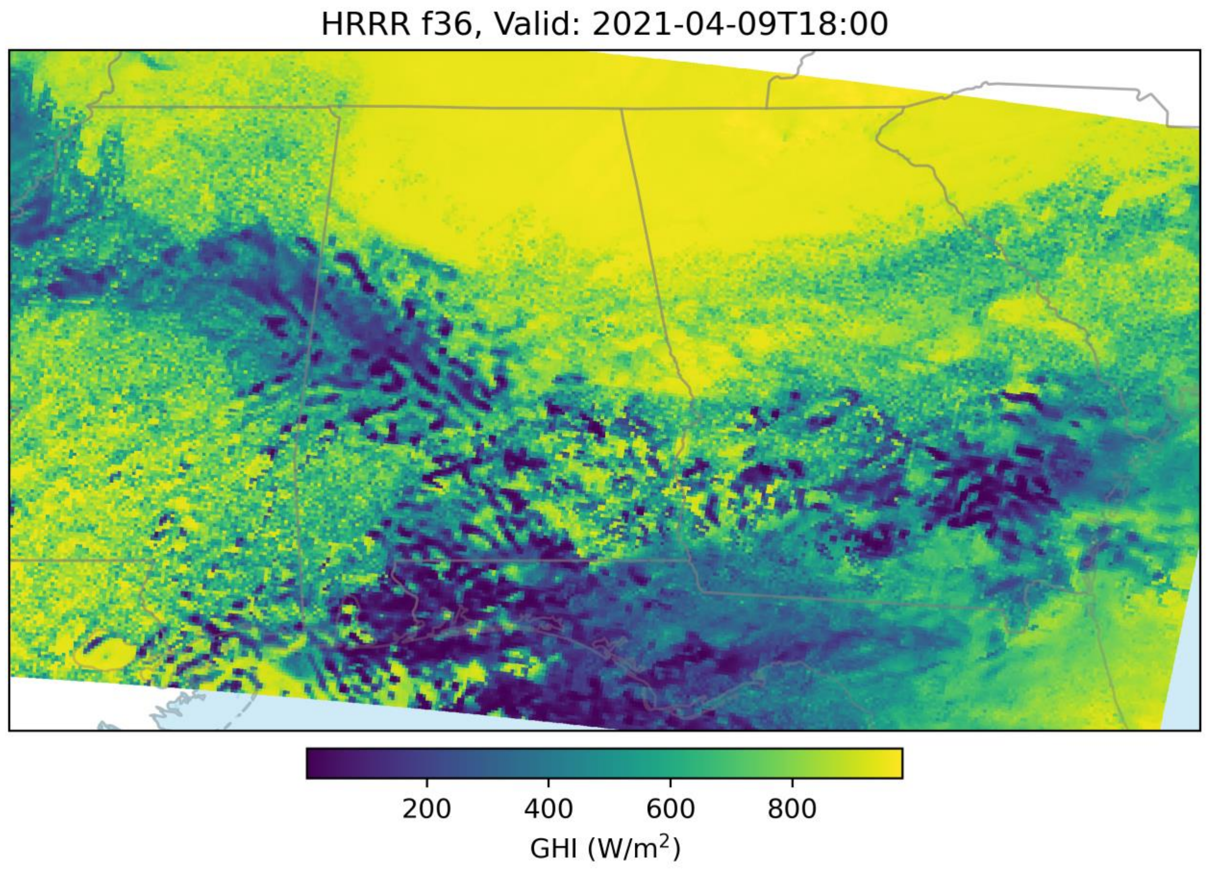

GEFS - Ensemble Member p03
Initialized: 06:00 UTC 11 Apr 2021
Valid: 18:00 UTC 12 Apr 2021
Weighted Avg. TCC = 001.37
0
10
20
30
40
50
60
70
Total Cloud Cover (%)

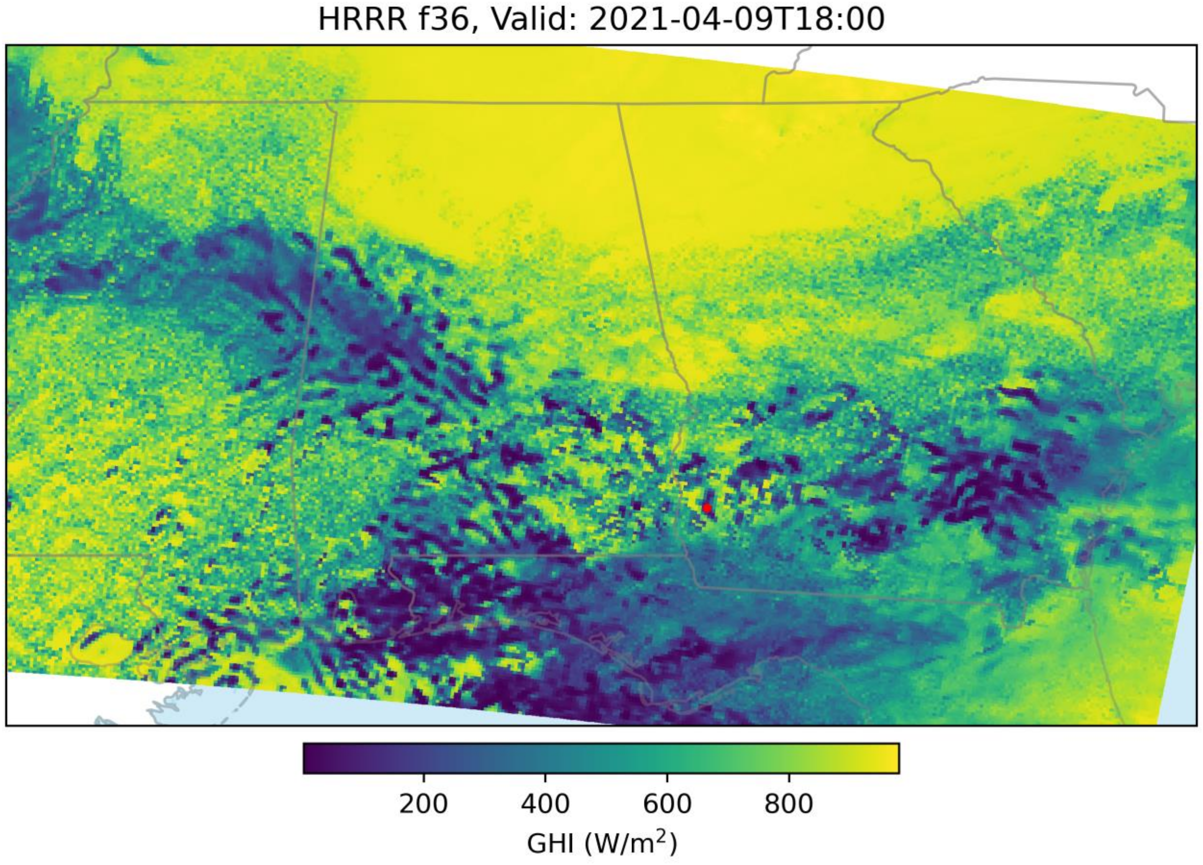

GEFS - Ensemble Member p04
Initialized: 06:00 UTC 11 Apr 2021
Valid: 18:00 UTC 12 Apr 2021
Weighted Avg. TCC = 000.03
0
10
20
30
40
50
60
Total Cloud Cover (%)

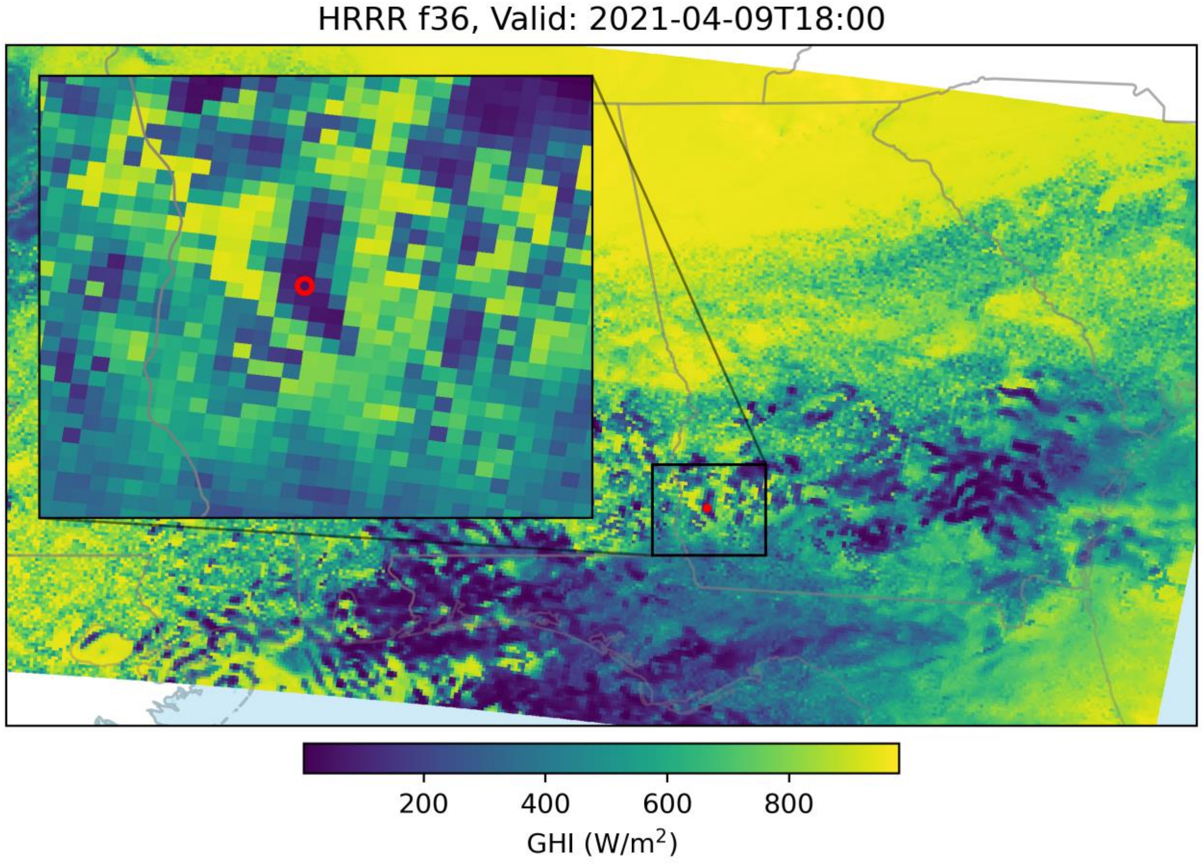
STD ≅ 0.7

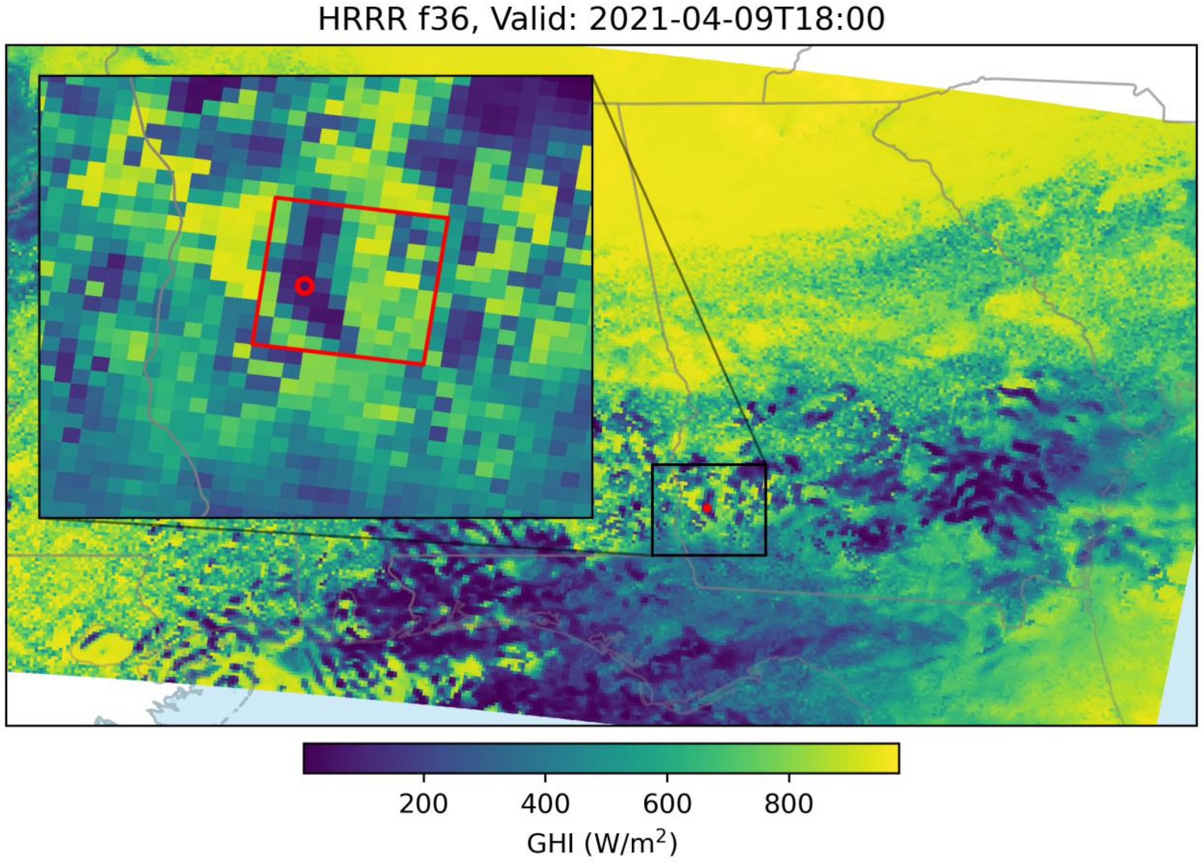
Spatial variation from HRRR

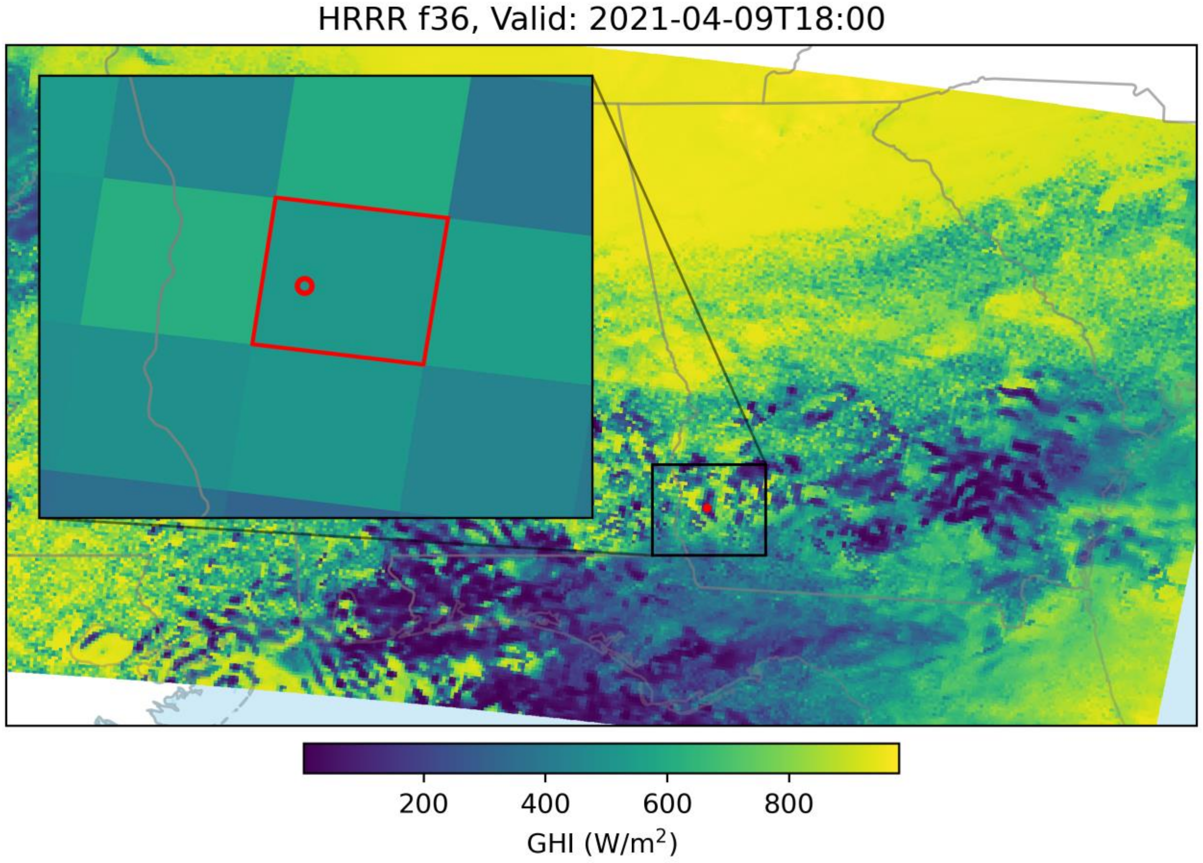
- Use NOAA's HRRR (high resolution rapid refresh) model

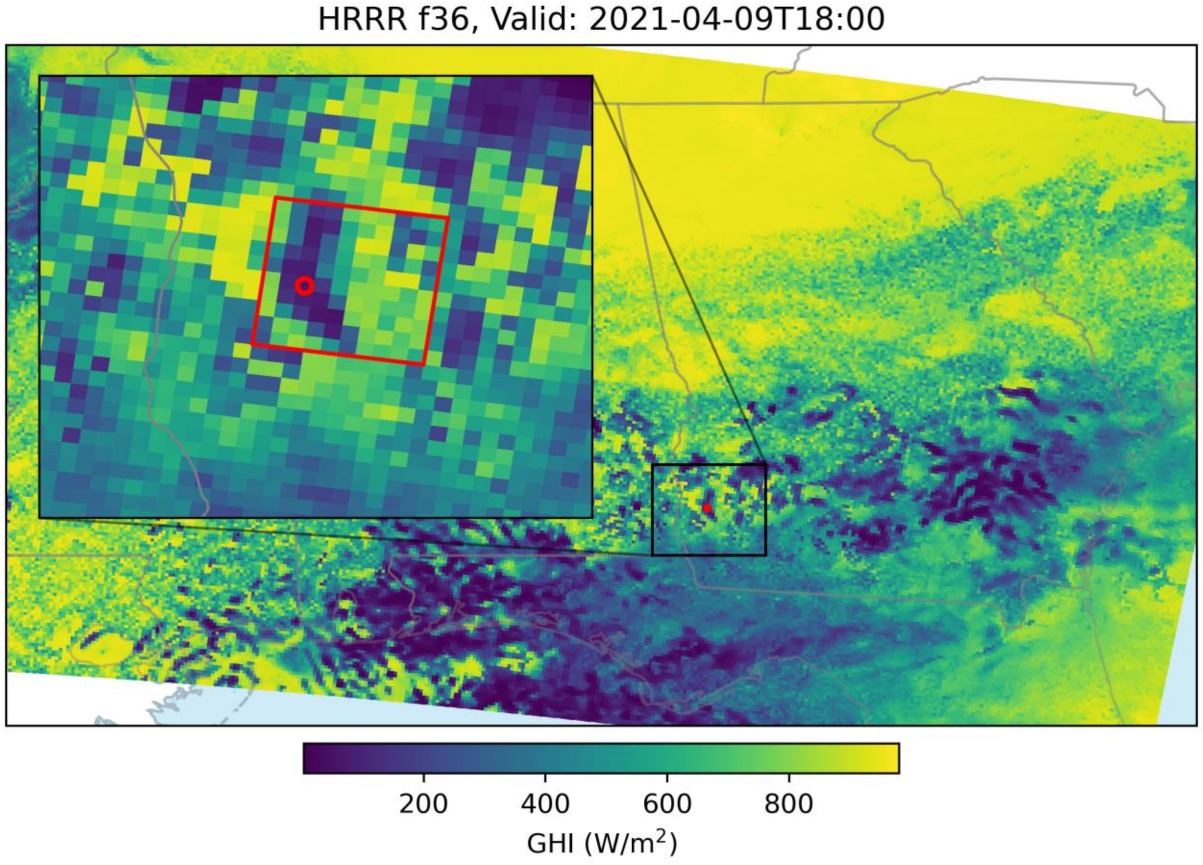

HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m2)

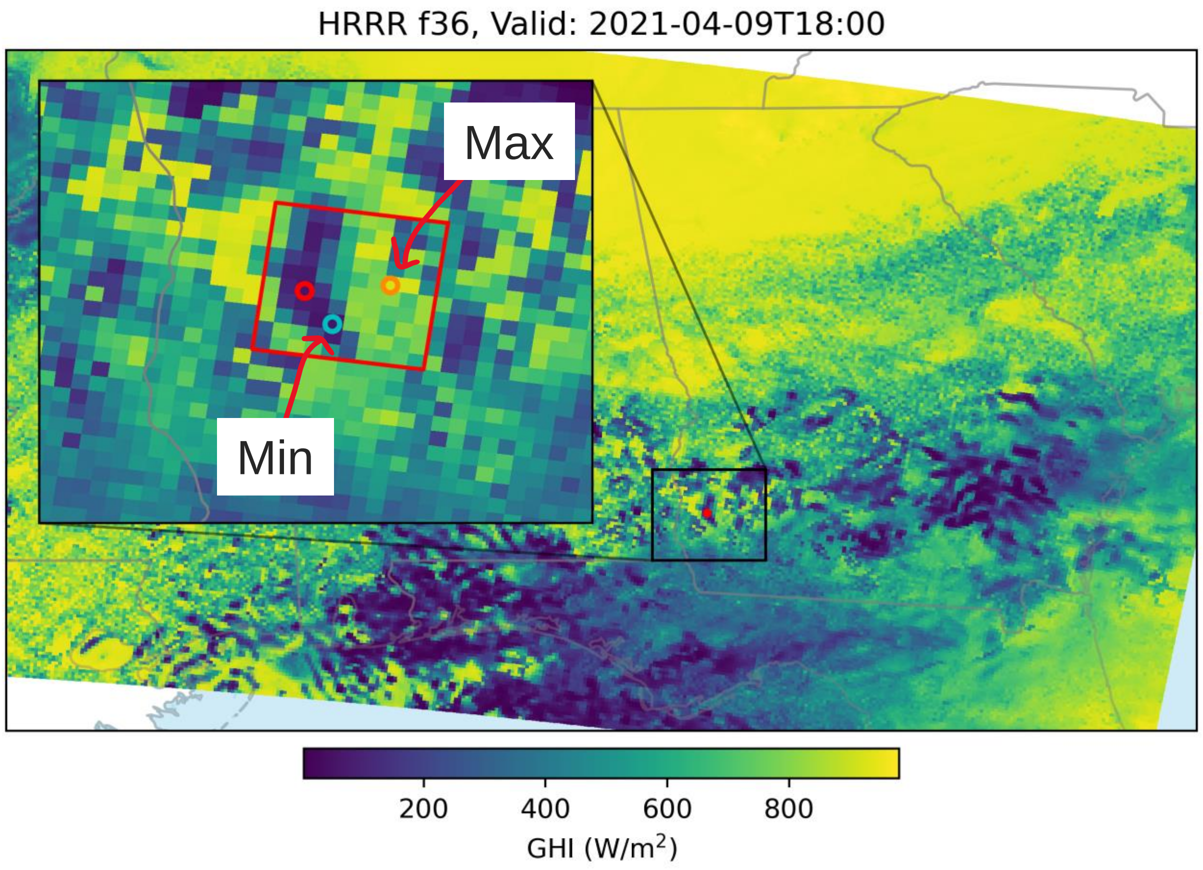

HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m^2)


HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m2)


HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m^2)


HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m2)

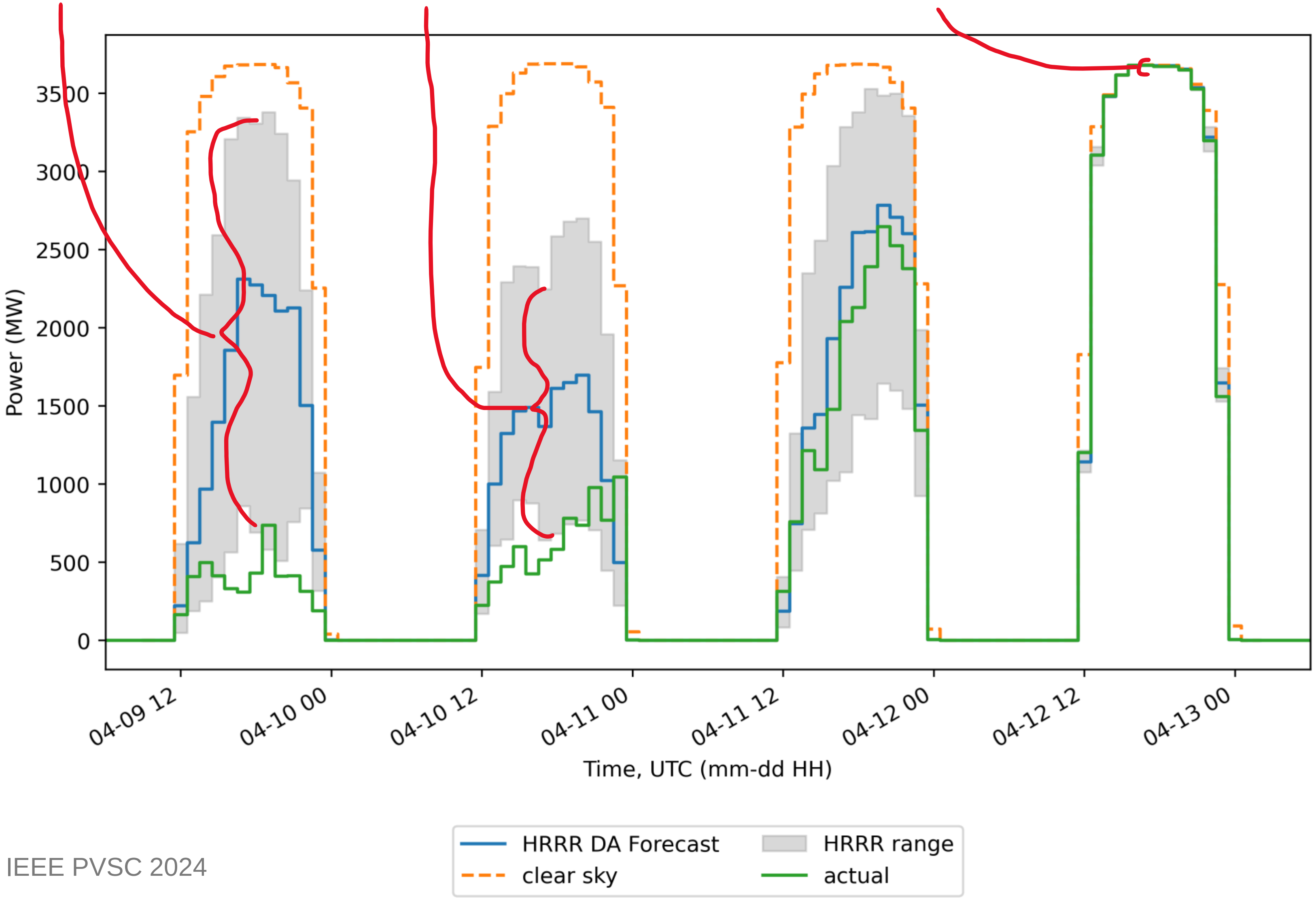

HRRR f36, Valid: 2021-04-09T18:00
200
400
600
800
GHI (W/m2)



…model power, repeat for all plants, sum mins, sum maxes

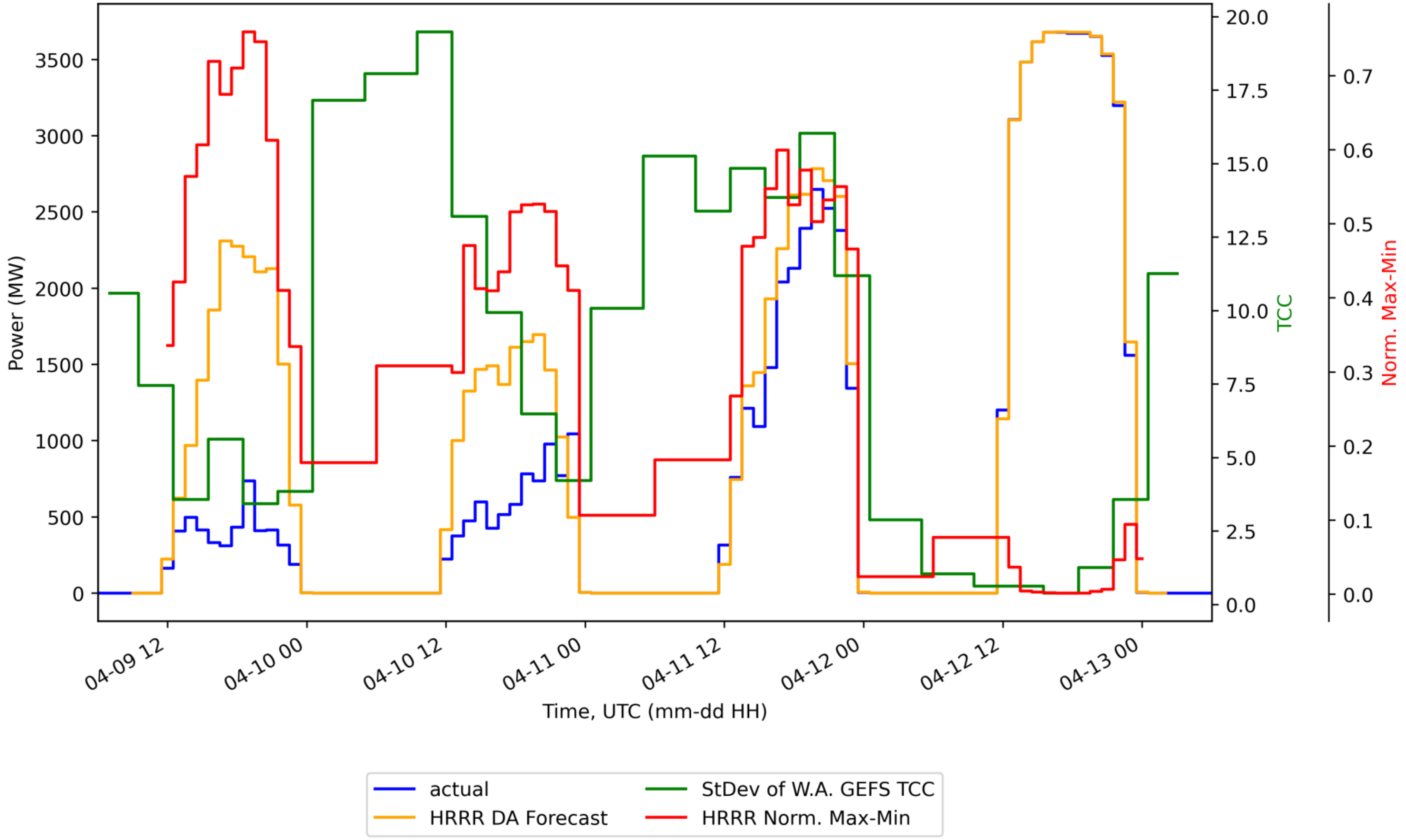
Example results:

High uncertainty Medium uncertainty Low uncertainty

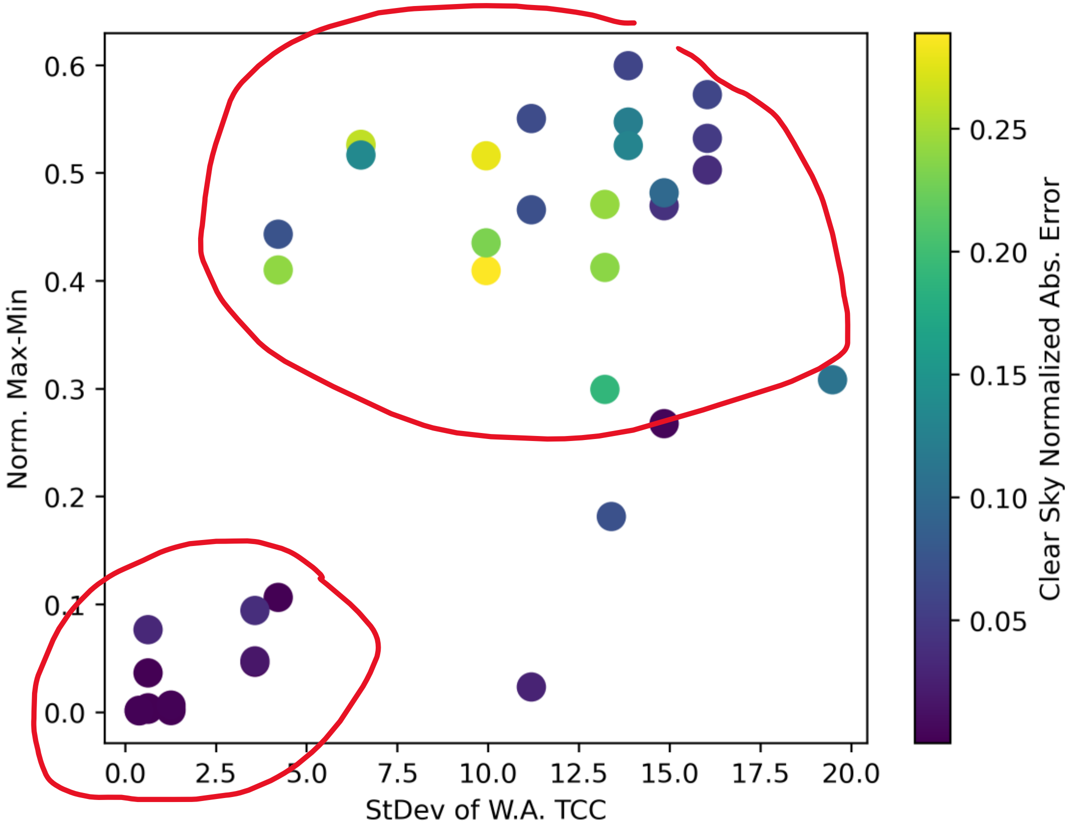


Preprint - presented at IEEE PVSC 2024

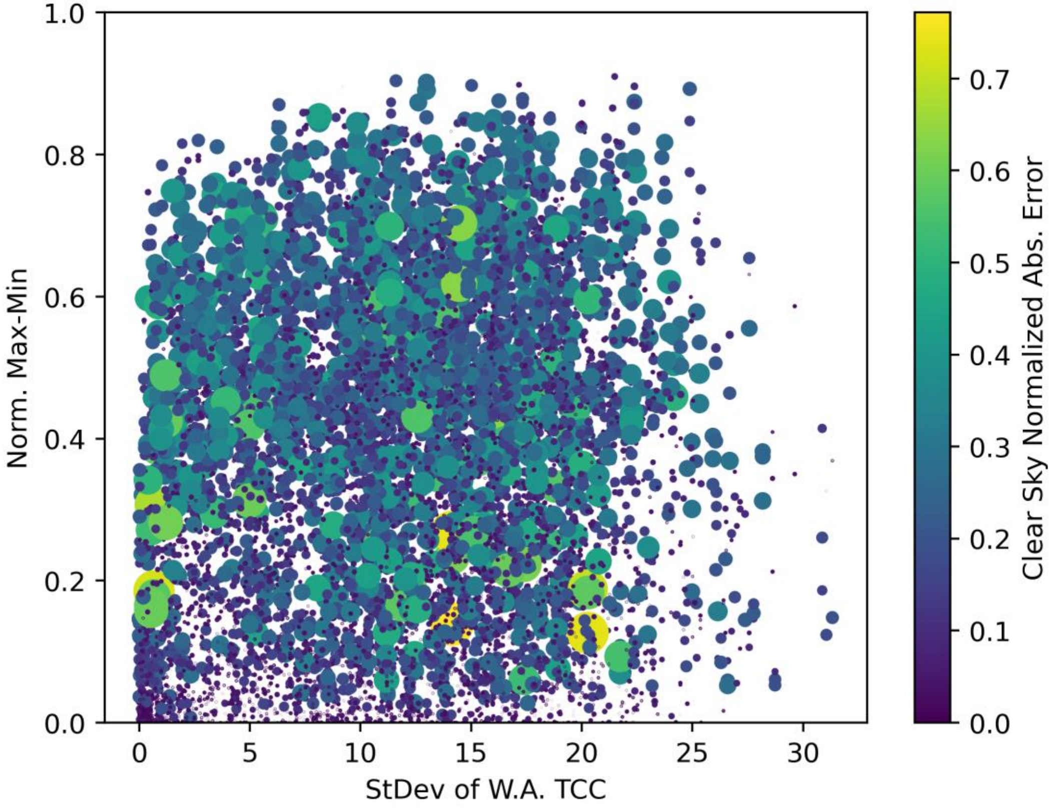
TCC and spatial variation for the days we've been looking at:



…now in a scatter plot:



… and for 2 full years:



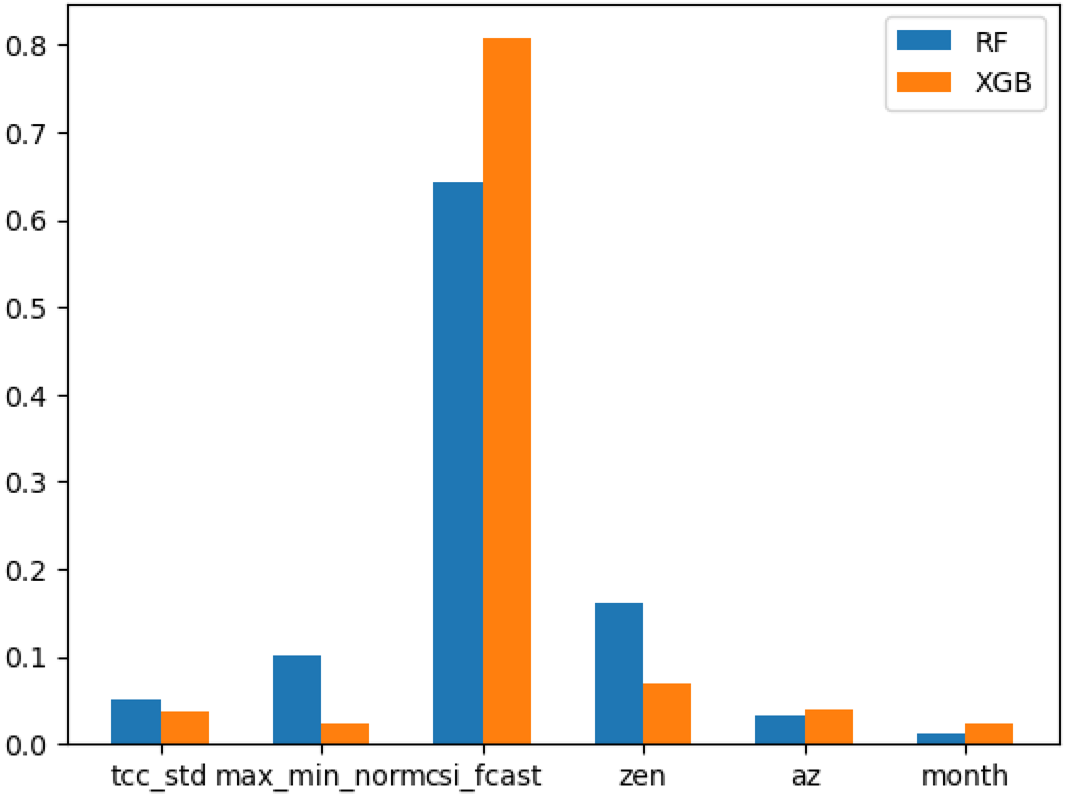
My conclusion: There's *something* there, but it's not a clear relationship

Solution:

- Machine learning quantile regression models: Random Forest, XGBoost
- **Inputs**:
  - **Existing forecast (normalized)**
  - **Historical forecast error (normalized)**
  - Std. of TCC
  - HRRR “spread”
  - Position of the sun in the sky
  - Month of year
- **Output**:
  - Updated forecast at different “quantiles”

Train with 2021/2022
Test with 2023

Feature importance

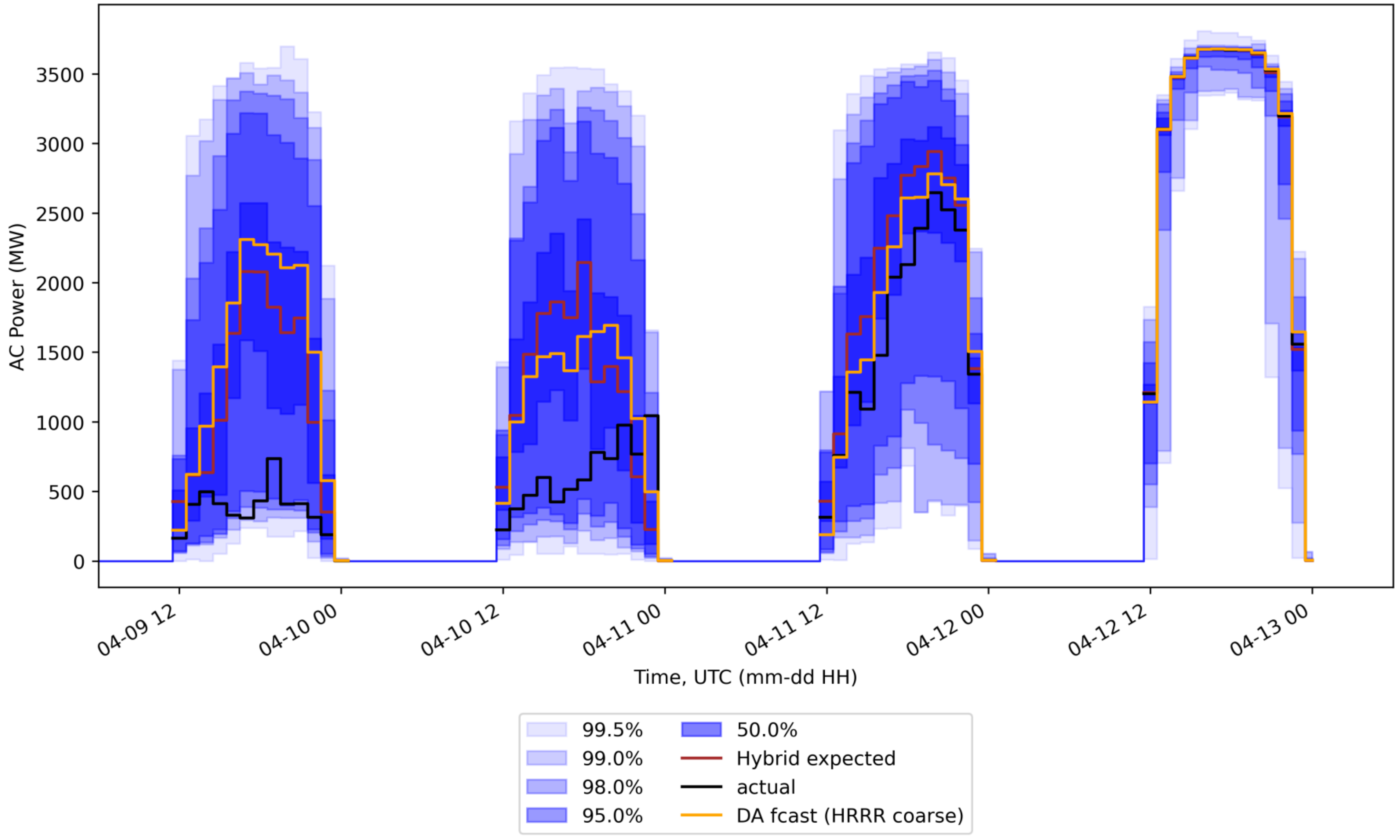


Novel indicators

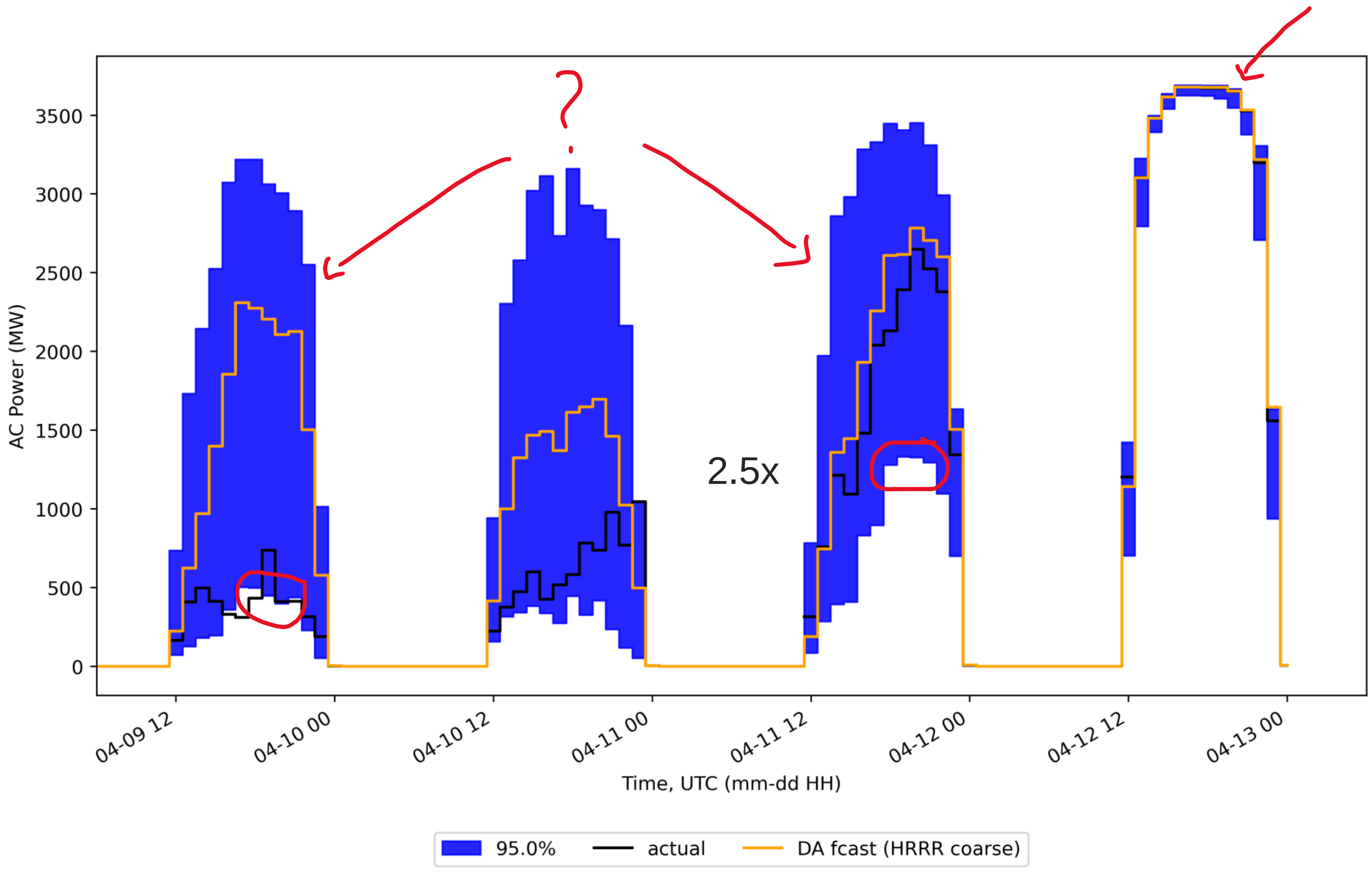
Solar position

Deterministic forecast

Sample results



Sample Results



Performance Statistics over 2023

- **PICP**: Prediction Interval Coverage Probability **"reliable"** ✓
  - *What fraction of observations fall in a prediction interval, e.g., 95% P.I.?*
- **PINAW**: Prediction Interval Normalized Average Width ← Normalized to clear sky
  - *Just because 95% of obs. fall in the 95% P.I. doesn't mean it's helpful…*

? **"sharp"**

Target PI	All Intervals	
	PICP	PINAW
0.995	0.993	0.59
0.99	0.989	0.54
0.98	0.981	0.46
0.95	0.957	0.37
0.5	0.510	0.12

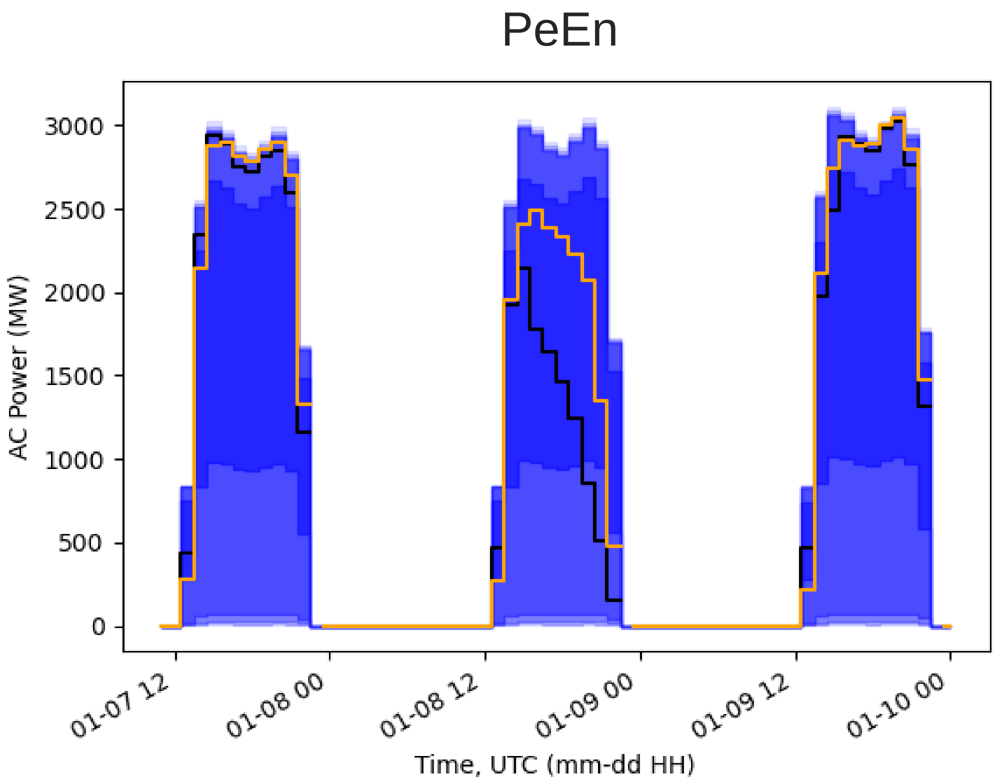
Comparison with Persistence Ensemble

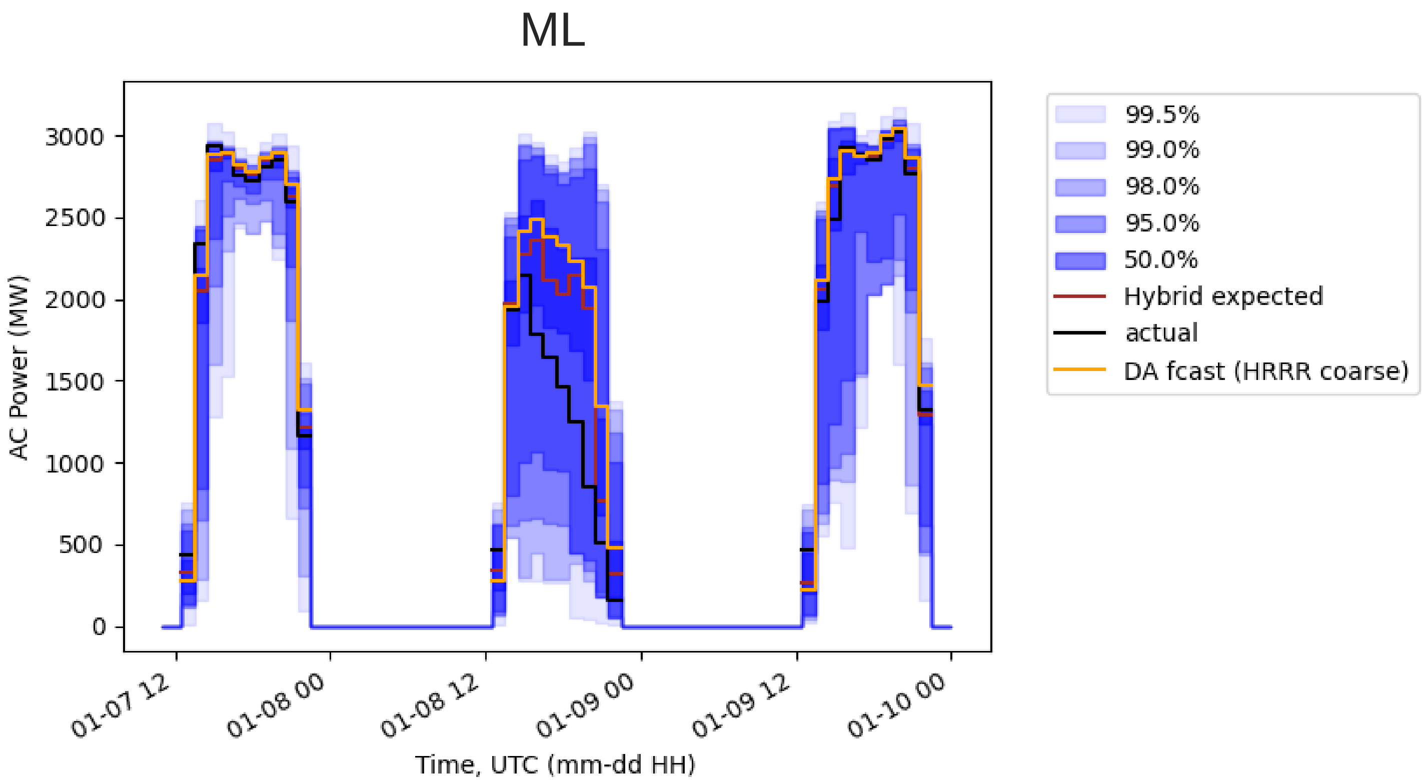
- Persistence Ensemble, PeEn: Like a historical PDF

(*removed intervals where obs. was 0% of clear sky*)

Target PI	ML Forecast		PeEn	
	PICP	PINAW	PICP	PINAW
0.995	0.993	0.59	0.997	0.81
0.99	0.969	0.54	0.991	0.80
0.98	0.950	0.46	0.979	0.79
0.95	0.904	0.37	0.943	0.77
0.5	0.440	0.12	0.494	0.45

Visual Comparison:





Timeout!

Of course it’s better than persistence! How do you know it’s actually *useful*??

Great question! I don’t *know* that, but:

- EPRI OPTSUN [1] showed potential for reliability improvements with no cost increase by using probabilistic forecasts
- These forecasts appear to be better than the ones used for Southern’s system in OPTSUN
- These forecasts only get better with a better deterministic forecast

[1] https://www.epri.com/optsun, see W. Hobbs, “Probabilistic Methods in Operations”, ESIG 2022 Spring Workshop for my summary (recorded: https://www.youtube.com/watch?v=1aO4kOoR2nc&t=1370s)

Operational Deployment Considerations

Runs in < 15 minutes on my 4-year-old laptop

Retraining with representative **observations** and **deterministic forecasts**

- Account for changes over time: increasing sites, **plant availability**
- Consider a reforecast (a.k.a., historical forecast, "hindcast") from the deterministic forecast provider
- Consider supplementing observations with satellite-based irradiance (plus power model) to build a longer history of newer fleet

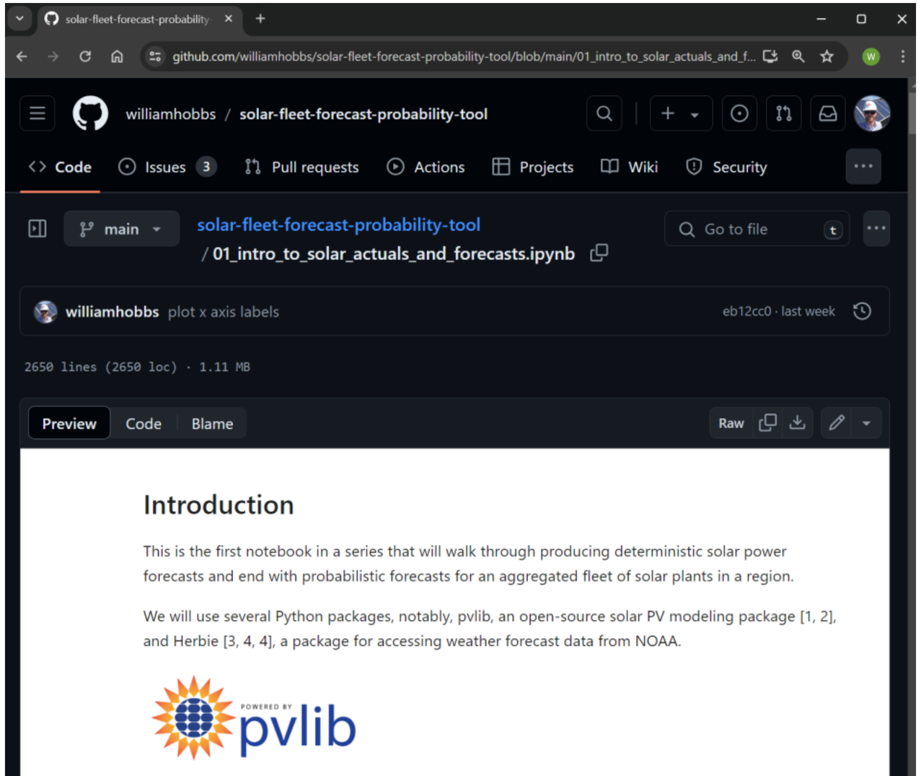
Especially if you *don't* want outages included in forecast uncertainty

Areas for Future Work

- Hyperparameter tuning
- Shorter and longer forecast horizons (than day-ahead)
- Extrapolation?
- Improved features
- Additional weather models
- Ramp rate probability forecasts
- Probabilistic net load

Available Code

- github.com/williamhobbs/solar-fleet-forecast-probability-tool
- BSD-3-Clause License
- Python, with Jupyter Notebooks to run everything yourself
- **Please let me know if you use it!**



Questions?

whobbs@southernco.com

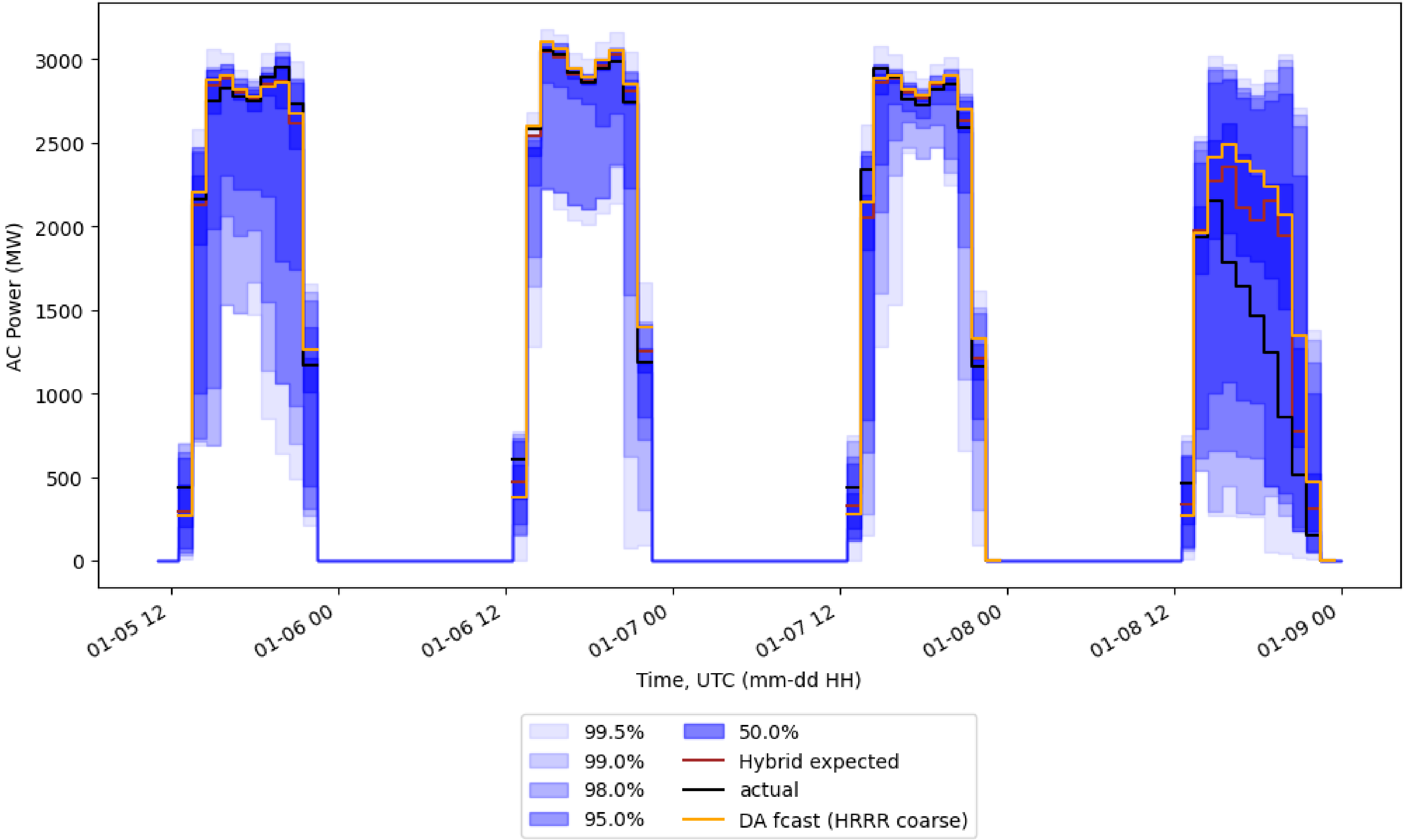
github.com/williamhobbs/solar-fleet-forecast-probability-tool

Southern Company

Preprint - presented at IEEE PVSC 2024

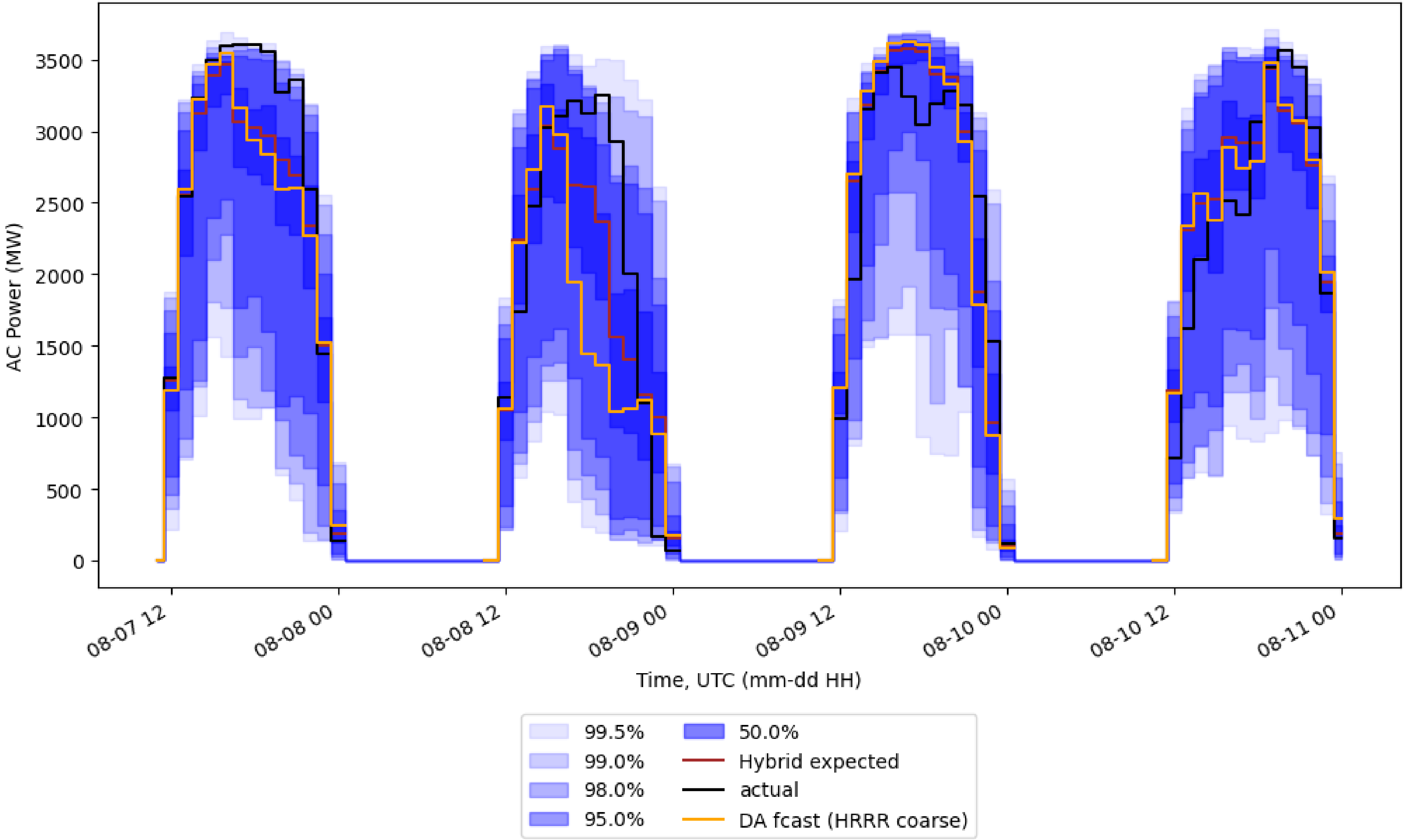
Sample data

- January



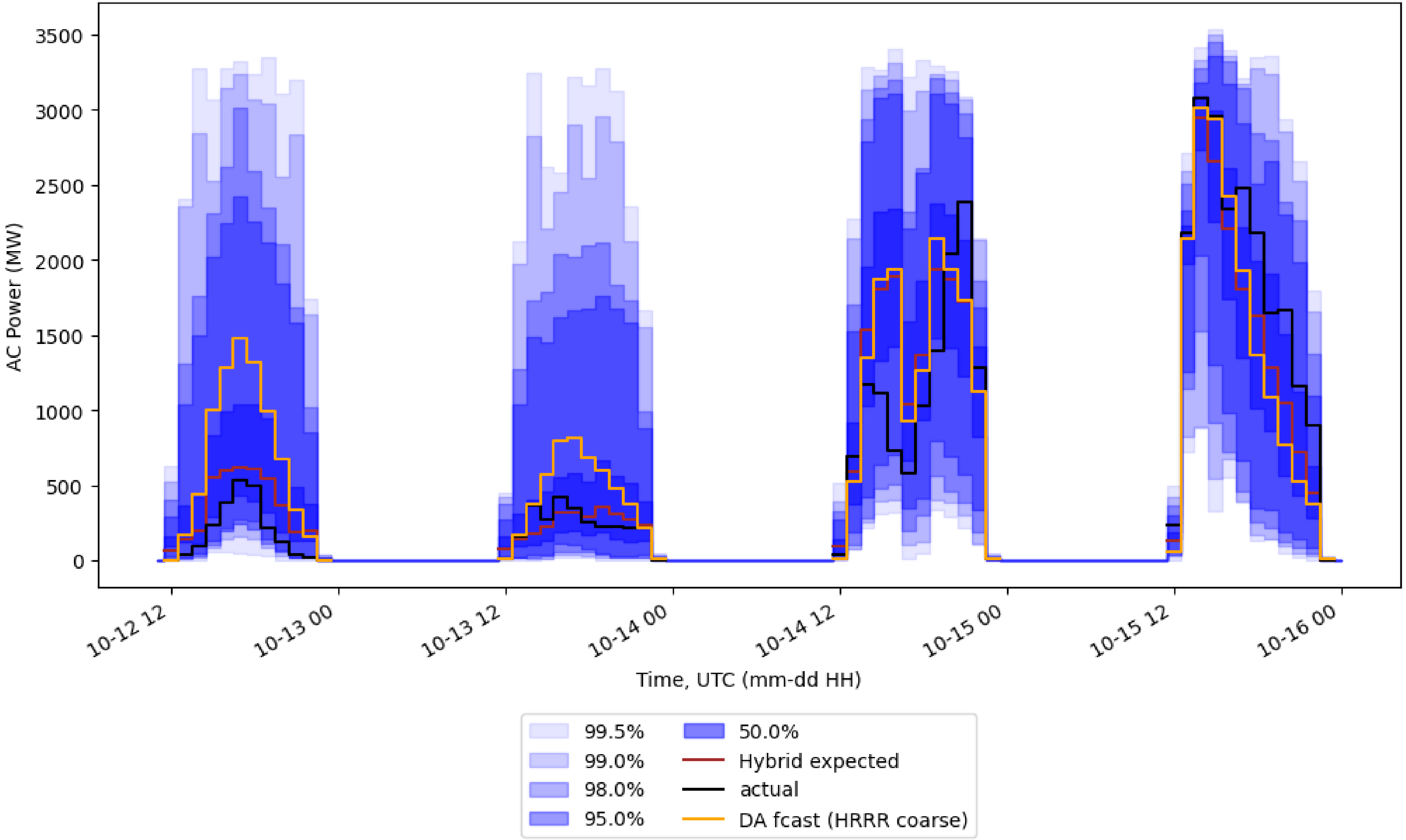
Sample data

- August



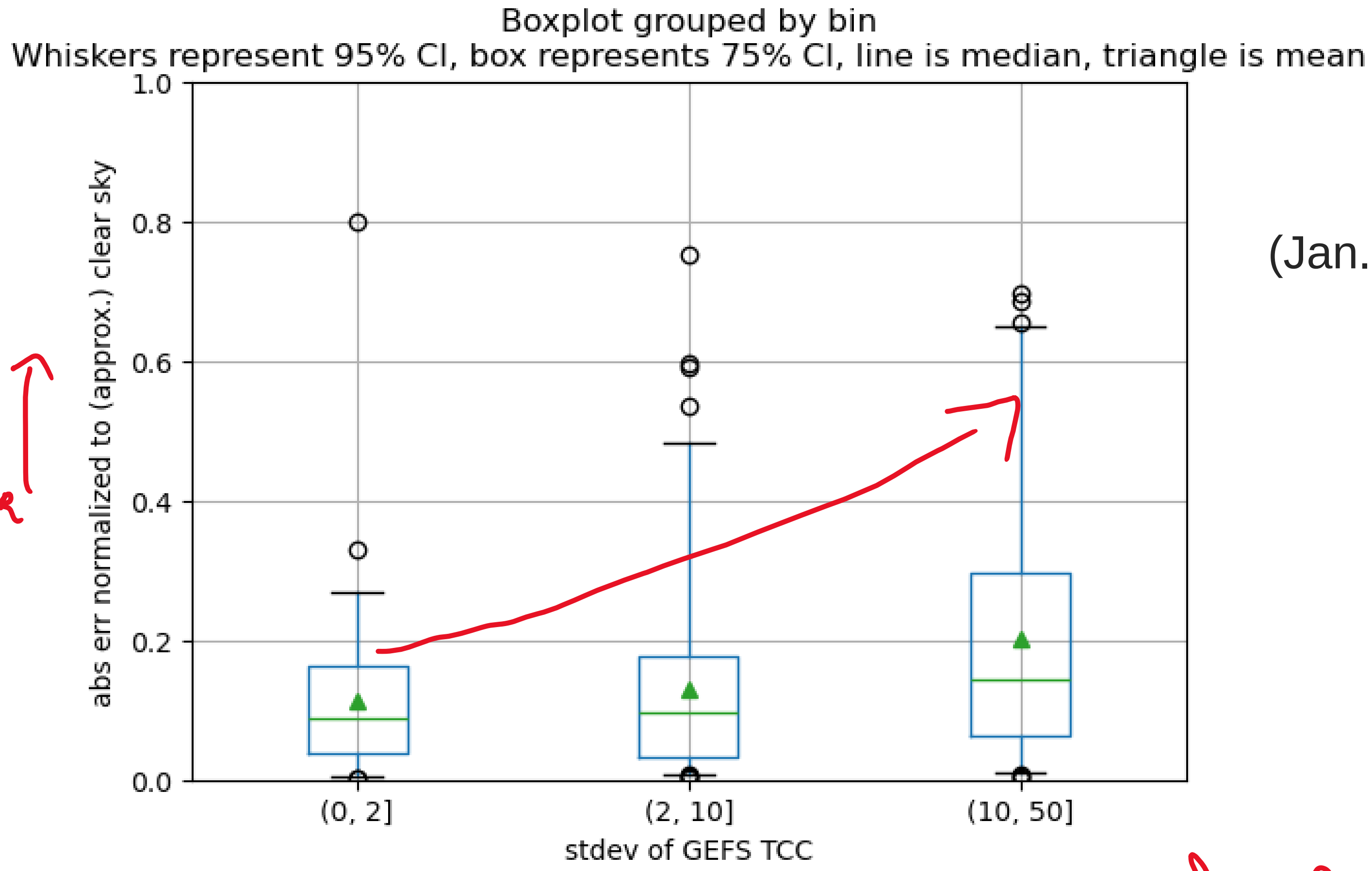
Sample data

- October



Using this uncertainty information

- Sometimes, there is a clear relationship between error and the GEFS ensemble spread (standard deviation of total cloud cover, or TCC):



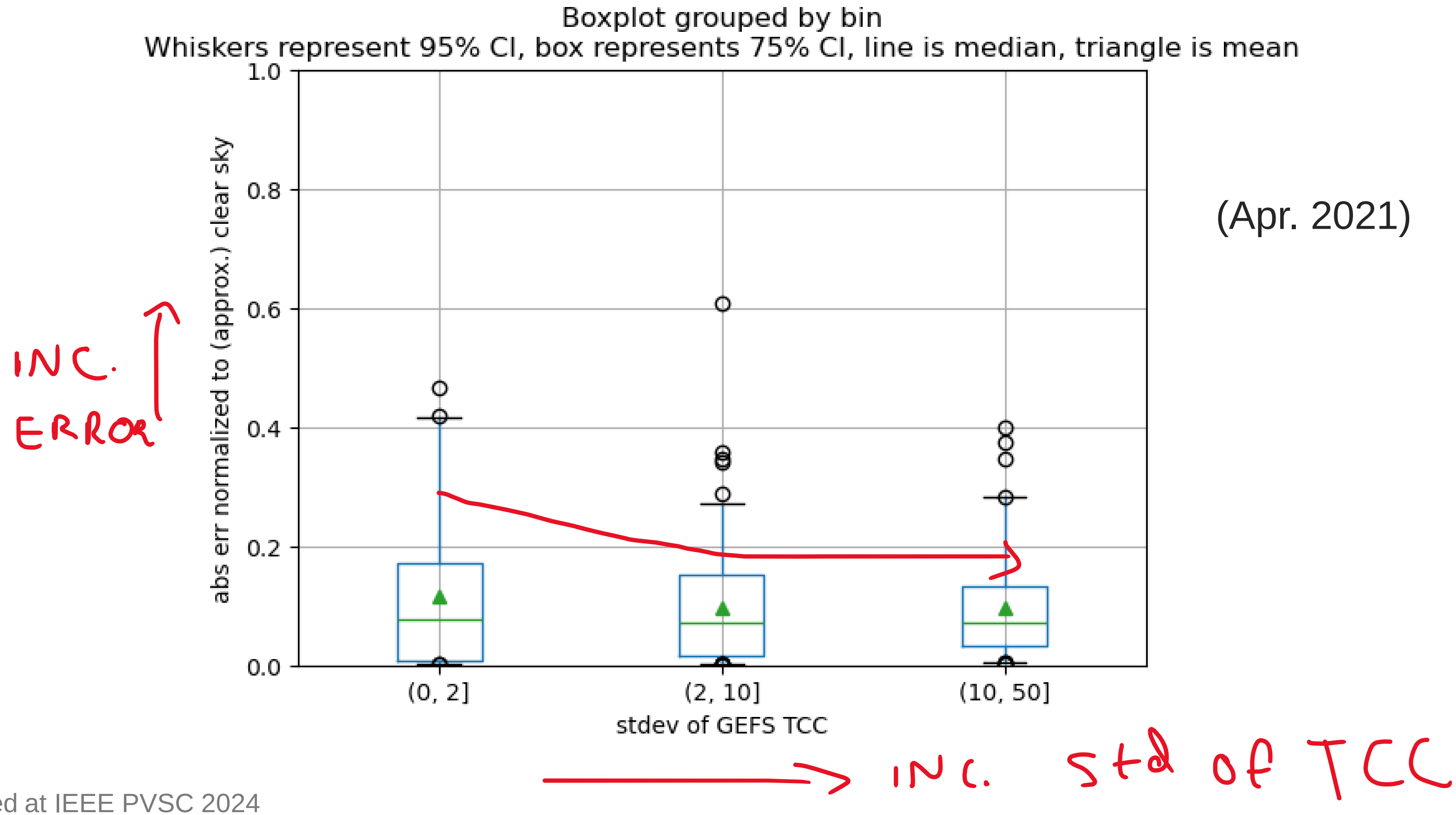
INC. ERROR

(Jan. 2021)

INC. std of TCC

Using this uncertainty information

- … but not always



(Apr. 2021)

Quantiles

- Quantile examples:
  - 0.5 is 50th percentile, or expected value
  - 0.95 is 95th percentile
  - Can be used for **prediction intervals**, e.g., 0.95 and 0.05 bound the 90% prediction interval (expect actual to fall in that range 90% of time)