ML benchmarking of land models

Gab Abramowitz

UNSW SYDNEY

ACCESS National Research Infrastructure

tern Ecosystem Research Infrastructure

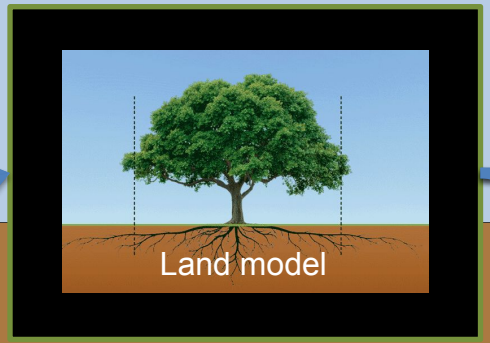
How land models work

Meteorology
SW radiation, LW radiation, temperature, precipitation, humidity, windspeed, air pressure, CO_2 concentration

Time t

Surface parameters
- Vegetation properties
- Soil properties

Time independent



Time t-1

Time t

Model states
- Soil moisture and temperature
- Carbon pools
- Vegetation state

Fluxes to atmosphere
- Evapotranspiration / latent heat flux
- Sensible heating of air
- Carbon absorption or emission
- Upward SW and LW radiation

Time t

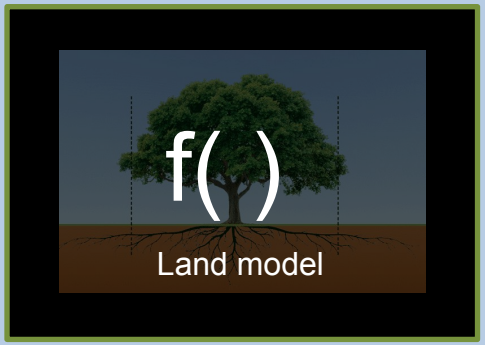
Fluxes to surface
- Surface runoff / streamflow
- Deep drainage / groundwater
- Carbon storage

How land models work

Meteorology
SW radiation, LW radiation, temperature, precipitation, humidity, windspeed, air pressure, CO_2 concentration

Surface parameters
- Vegetation properties
- Soil properties

Model states at time t-1
- Soil moisture and temperature
- Carbon pools
- Vegetation state

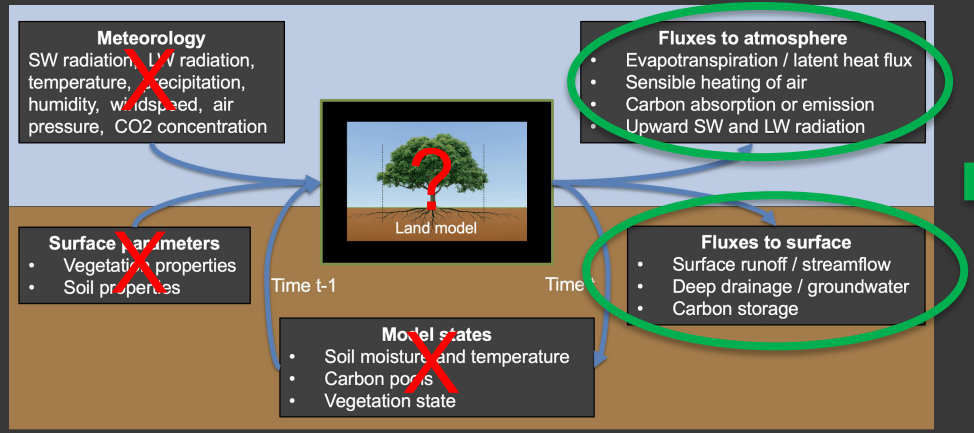


Fluxes to atmosphere
- Evapotranspiration / latent heat flux
- Sensible heating of air
- Carbon absorption or emission
- Upward SW and LW radiation

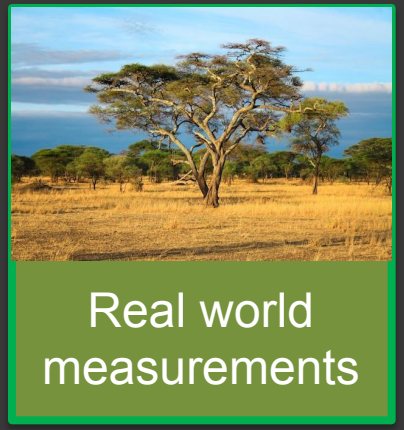
Fluxes to surface
- Surface runoff / streamflow
- Deep drainage / groundwater

Model states at time t
- Soil moisture and temperature
- Carbon pools
- Vegetation state

How do we evaluate a land model?



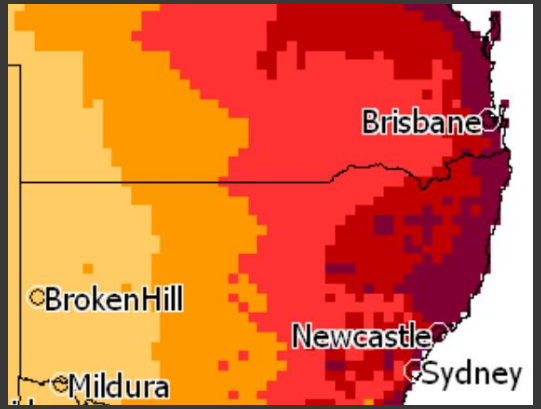
Do they agree?

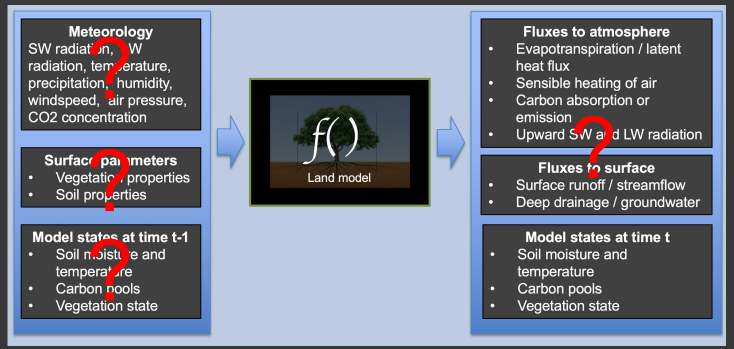


Not so simple!

Gridded land model simulations – poor constraint

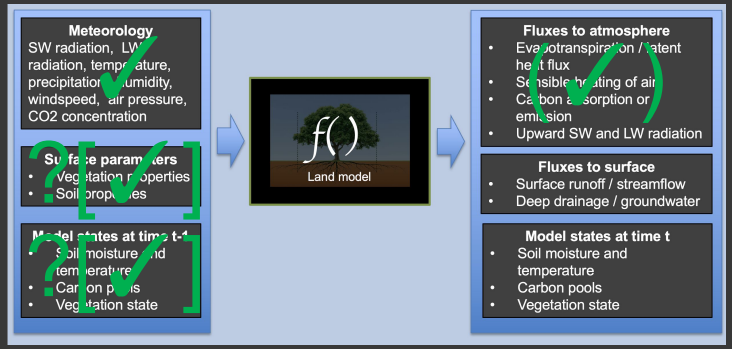
- Meteorological forcing is usually derived from a reanalysis product (model simulation with assimilated observations)
  - Forcing uncertainty is typically not known or quantified
- Model parameters are poorly constrained
  - Usually prescribed with vegetation type and soil type
- Most initial states cannot be measured
- Gridded evaluation products are typically very uncertain
- Evaluation time step is typically large (e.g. monthly)
  - Evaluating the result of many iterations of the model





Site-based land model simulations (flux towers)

- In-situ measurement of meteorology and atmospheric fluxes
- Same time step size as land models
- Many vegetation and soil parameters can be directly measured
- Soil moisture, temperature and carbon stores often measured
- Uncertainties relatively well quantified
- Hundreds of sites, millions of actual observations (large sample size)

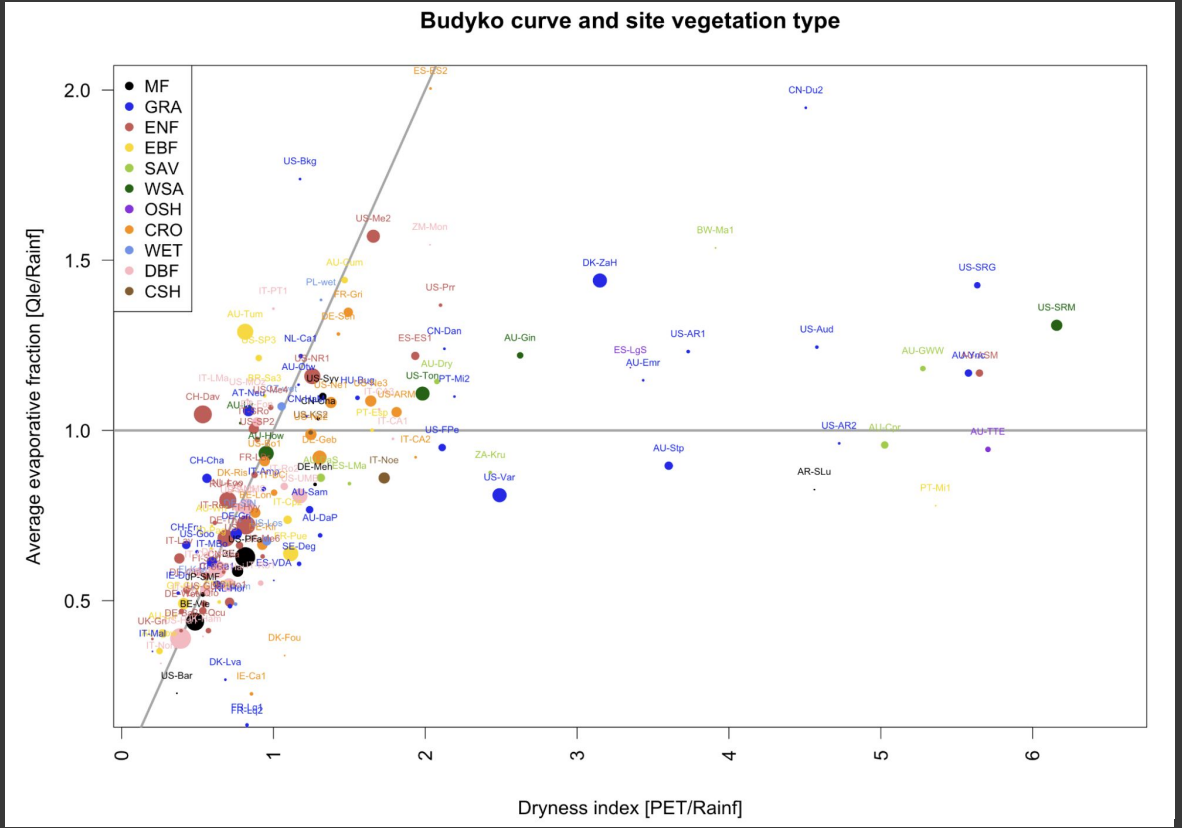


Elephants in the room

Flux tower data quality:

- Conservation issues
  - can look at both raw and corrected fluxes (using Fluxnet2015 EB closure approach)
- Low turbulence periods
  - Can filter out low wind speed / U* time steps

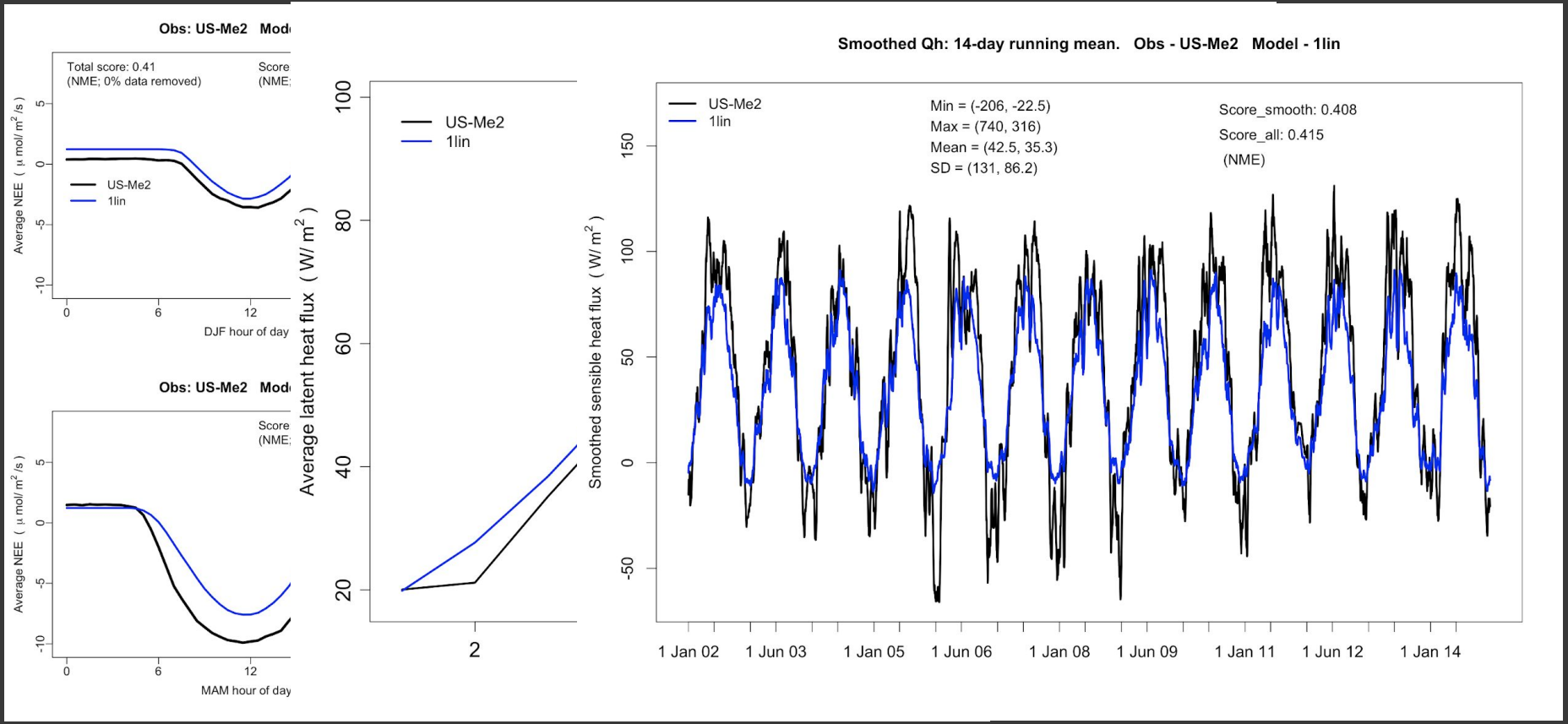
Inconsistencies between real world and model world…



Machine learning models as benchmarks

- Use ML to predict fluxes using meteorological variables as predictors, just like a land model
- Flux tower data is plentiful: 100s of sites, millions of samples, 30 minute data
- ML tells us how much information meteorology has about the fluxes – a proxy for predictability
- We can use ML models to benchmark land models – understand *a priori* how well we should expect them to perform
- By varying the set up of the ML models, and how we test them out-of-sample, we can create a hierarchy of performance levels to categorise land model performance. We can vary:
    - Which variables empirical models use as predictors
    - Which location & time period the empirical models are trained on (versus tested on)
    - How complicated empirical models are
    - Allow lagged variables as proxies for physical model states, or ML with states
- Allows differing performance expectation depending on complexity of conditions and location

Why is machine learning necessary?



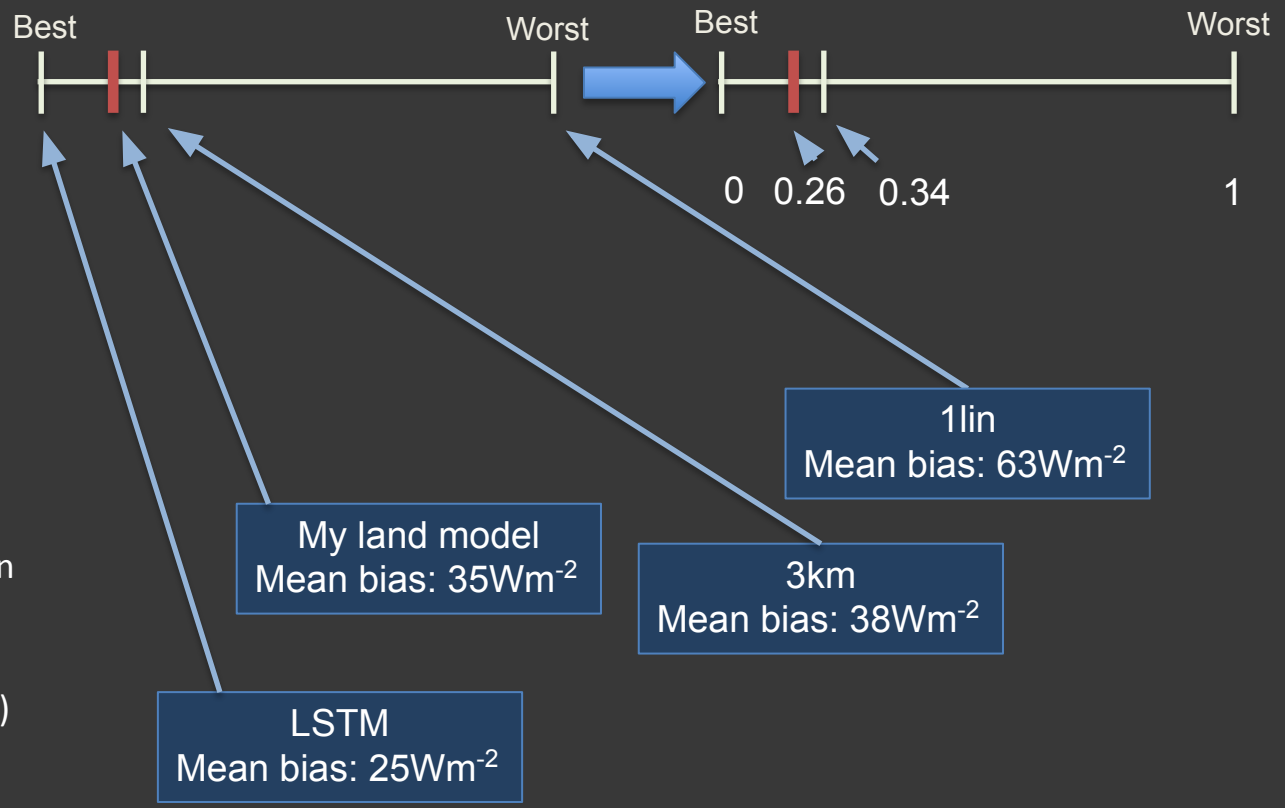
Different metric, different story: aggregate across metrics

Independent metric set:

mean bias, correlation, SD difference, normalized mean error, PDF overlap, 5th and 95th percentile difference.

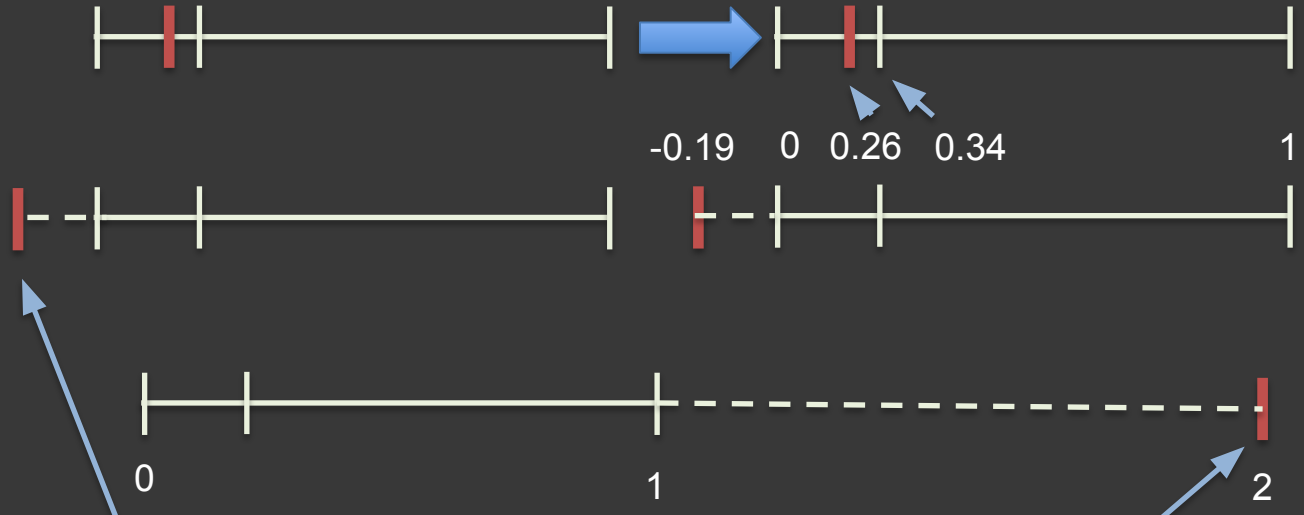
Simple hierarchy of empirical models:

1. Linear regression against SWdown (1lin)
2. Cluster+regression against SWdown, Tair and humidity (3km)
3. LSTM given all meteorology variables



Aggregating information from different metrics

1. Linear regression against SWdown (1lin)
2. Cluster+regression against SWdown, Tair and humidity (3km)
3. LSTM given all meteorology variables



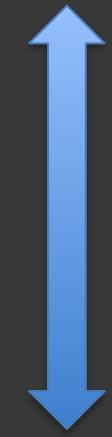
Take the mean of the normalised metric value over:

- many metrics
- many sites

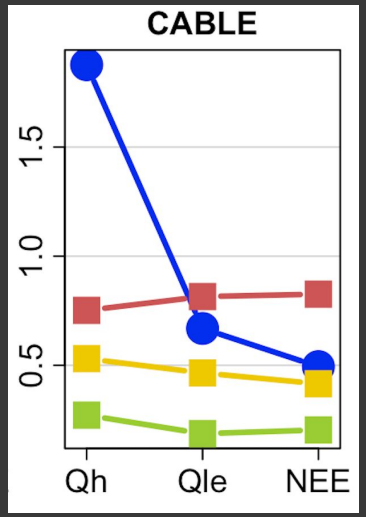
Independent normalized metric value (iNMV)

Aggregating information from different metrics

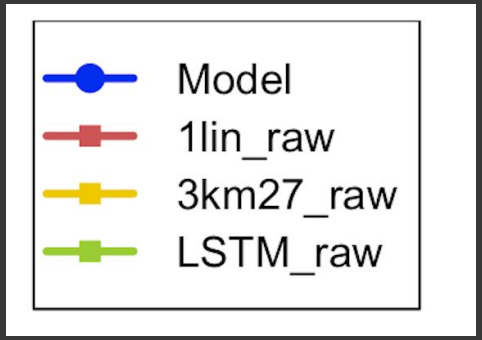
worse

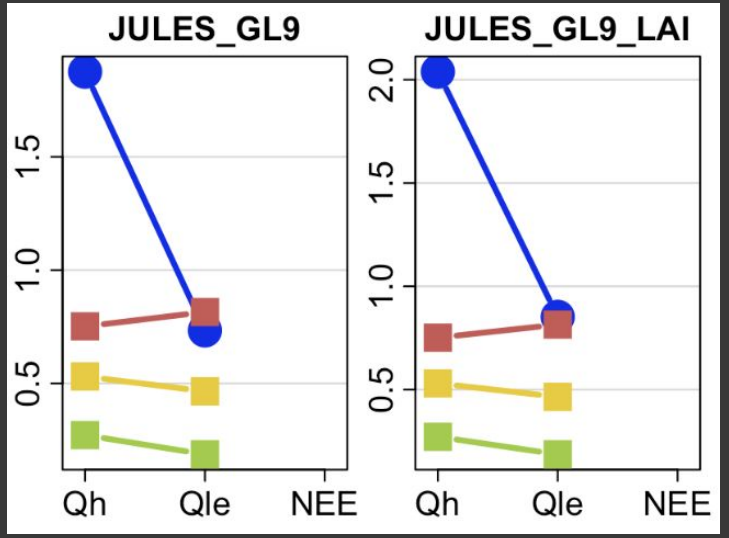


better



Independent normalized metric value (iNMV) averaged over 7 metrics and 154 sites

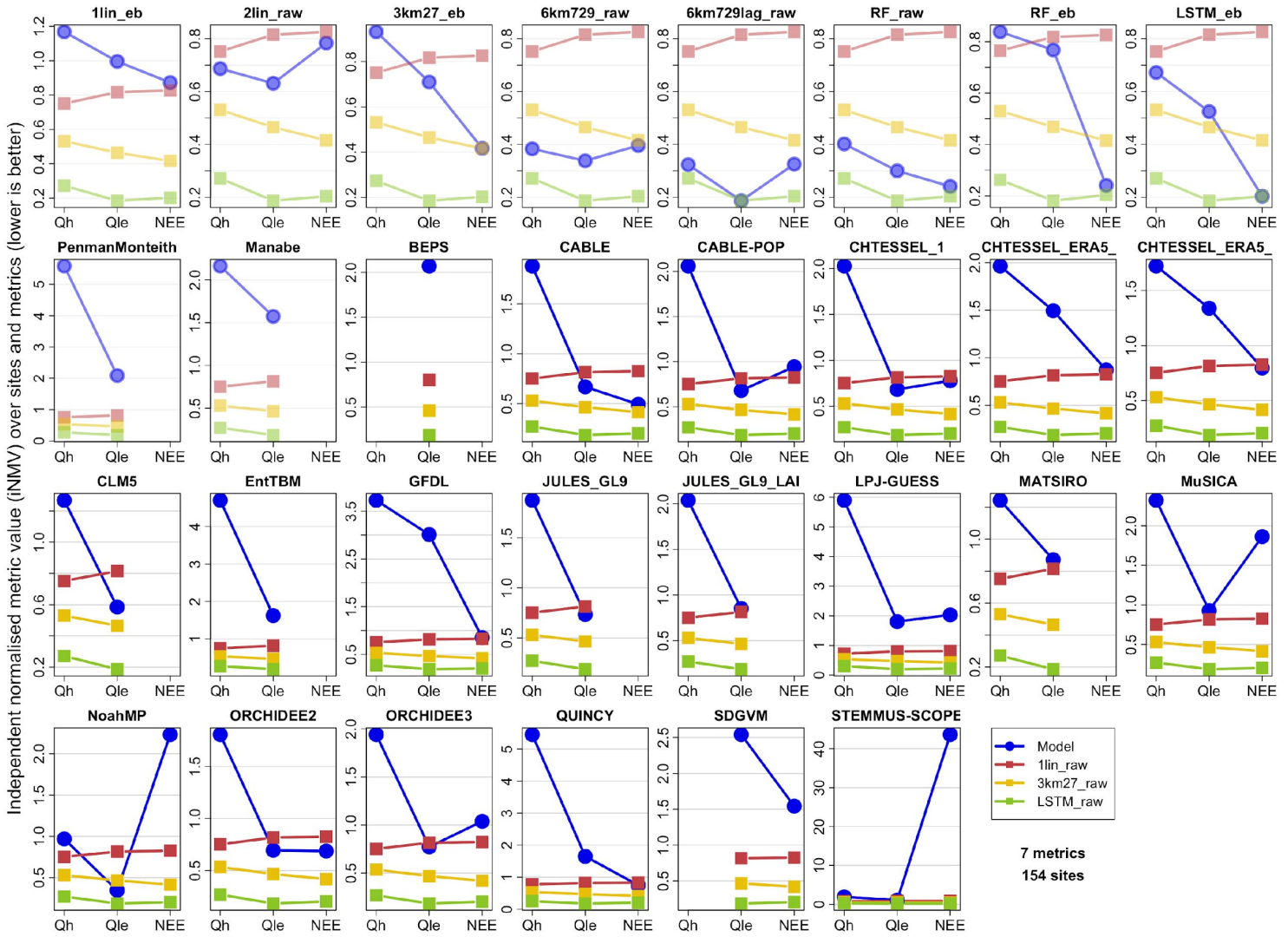




Independent normalized metric value (iNMV)

Only CABLE, CHTESSEL, CLM, JULES, ORCHIDEE, NoahMP ever beat the out-of-sample linear regression against shortwave

no model, averaged over all variables, beats linear regression

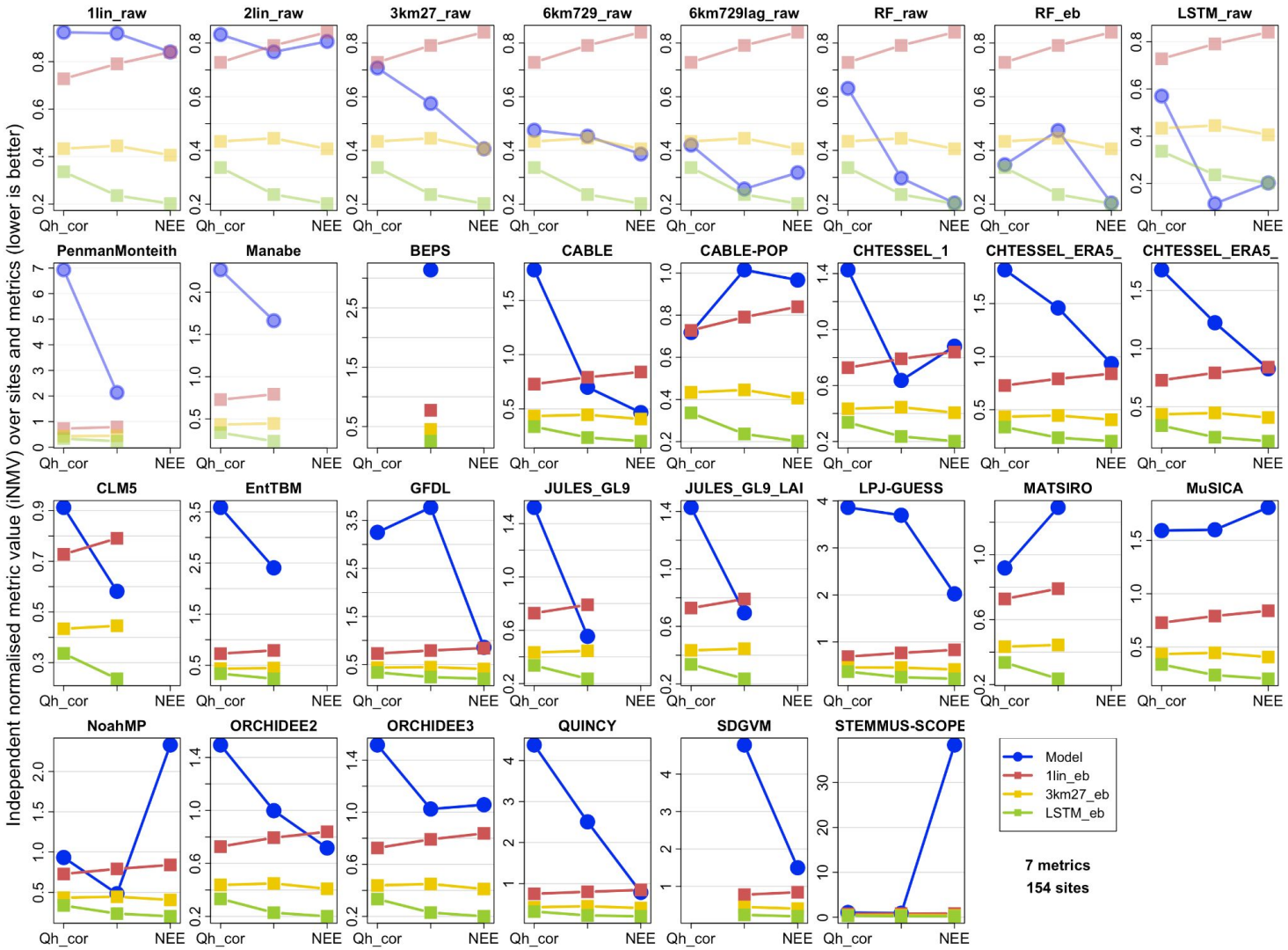


Independent normalized metric value (iNMV)

Only CABLE, CHTESSEL, CLM, JULES, ORCHIDEE, NoahMP ever beat the out-of-sample linear regression against shortwave

no model, averaged over all variables, beats linear regression

EB-corrected data

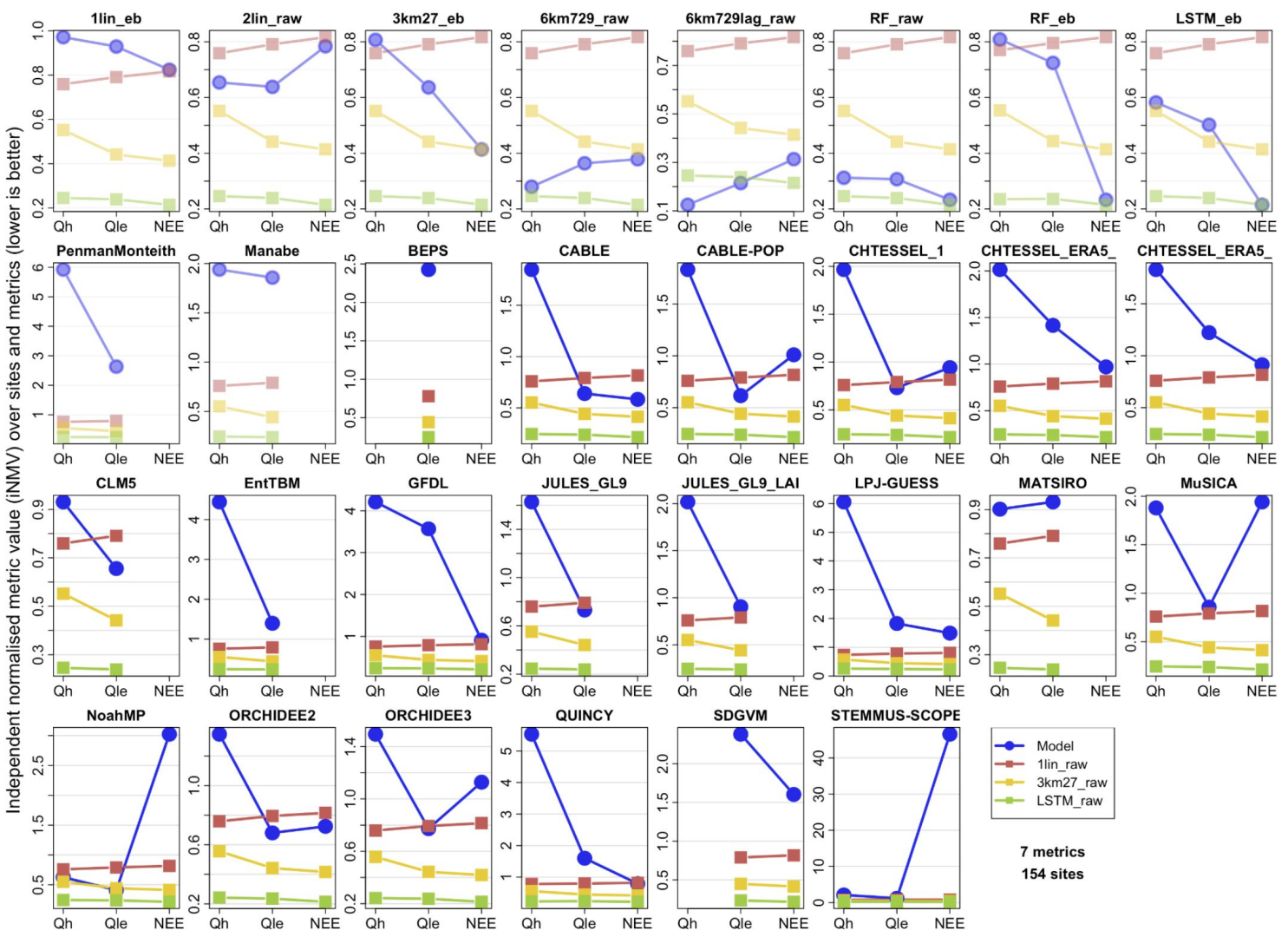


Independent normalized metric value (iNMV)

Only CABLE, CHTESSEL, CLM, JULES, ORCHIDEE, NoahMP ever beat the out-of-sample linear regression against shortwave

no model, averaged over all variables, beats linear regression

Wind speed > 2m/s



What could be going on here?

A few hypotheses:

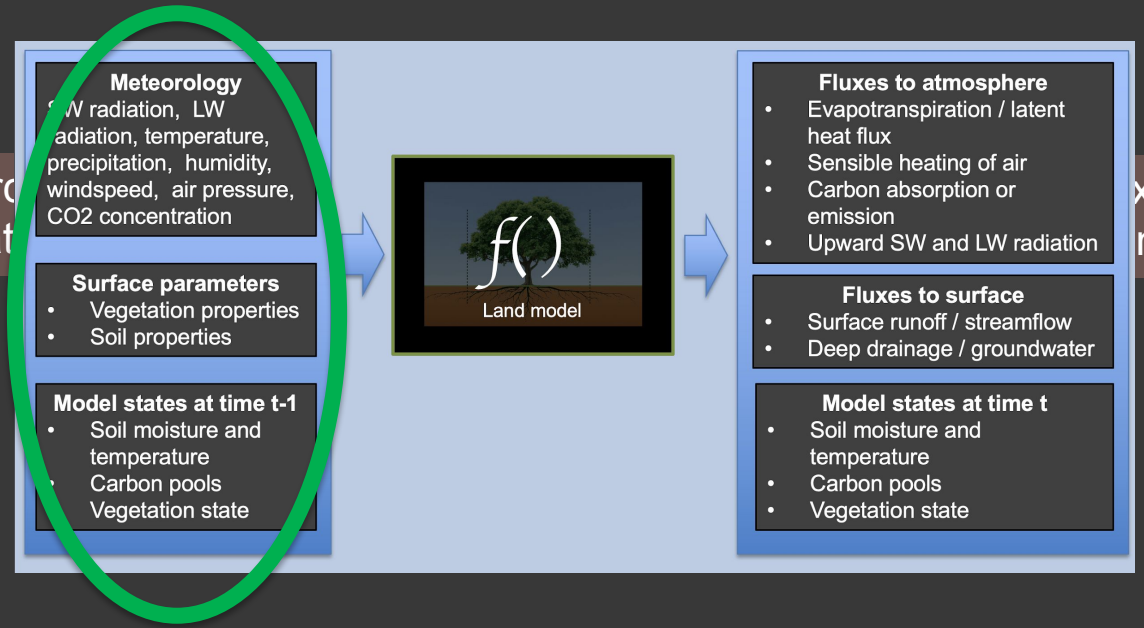
- Model is perfect but we can't prescribe the correct parameters
- Flux tower data is systematically wrong
  - Not just conservation, not just low turbulence, not just advection
- Coupling is key, it's about compensating biases – coupled SCM might be better?
- Parametrisations are good 90% of the time, but that is not good enough

How could we tell where the problem is?

Domain analysis:

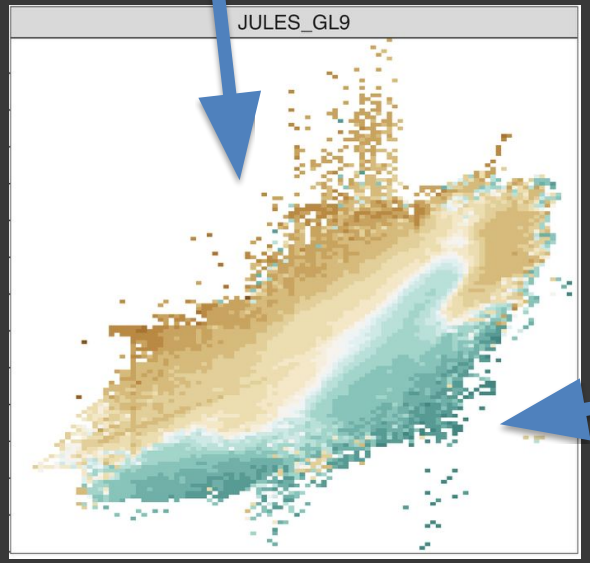
1. In which part of the forcing domain space (e.g. radiation, temperature, humidity) is poor LSM performance most common?

Look at met forc
variables two at
x
n each bin

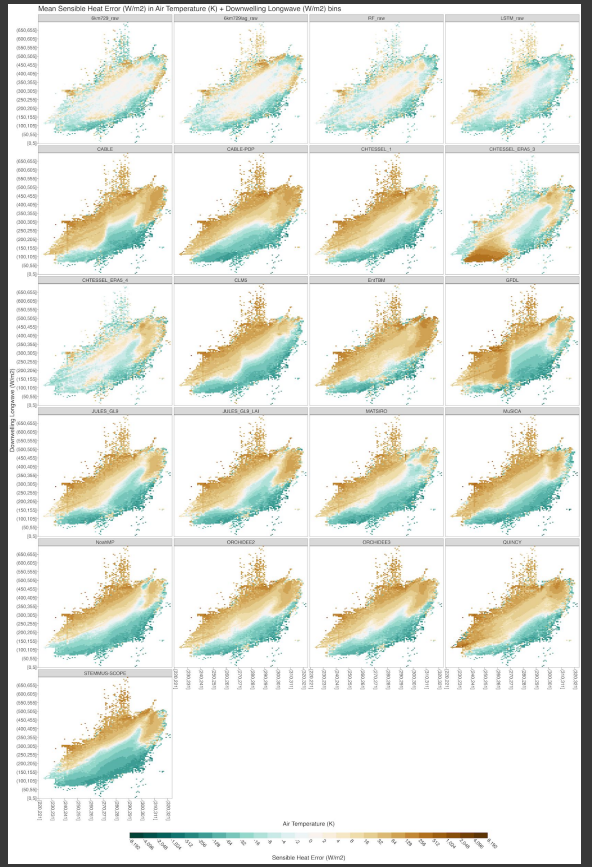


Mean flux error as a function of drivers, two at a time

High sensible heat bias



Low bias



Page et al, in review.

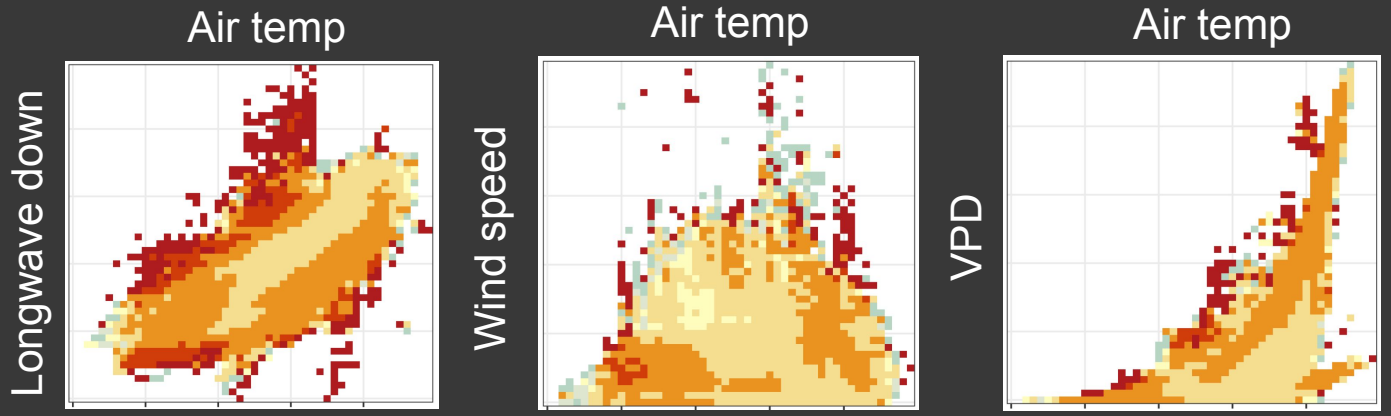
Potential for improvement
=> high proportion of time steps outperformed by 3 empirical models

JULES GL9 error > all (ML_1 error, ML_2 error, ML_3 error,)

⇒ JULES GL9 loss" (sensible heat)

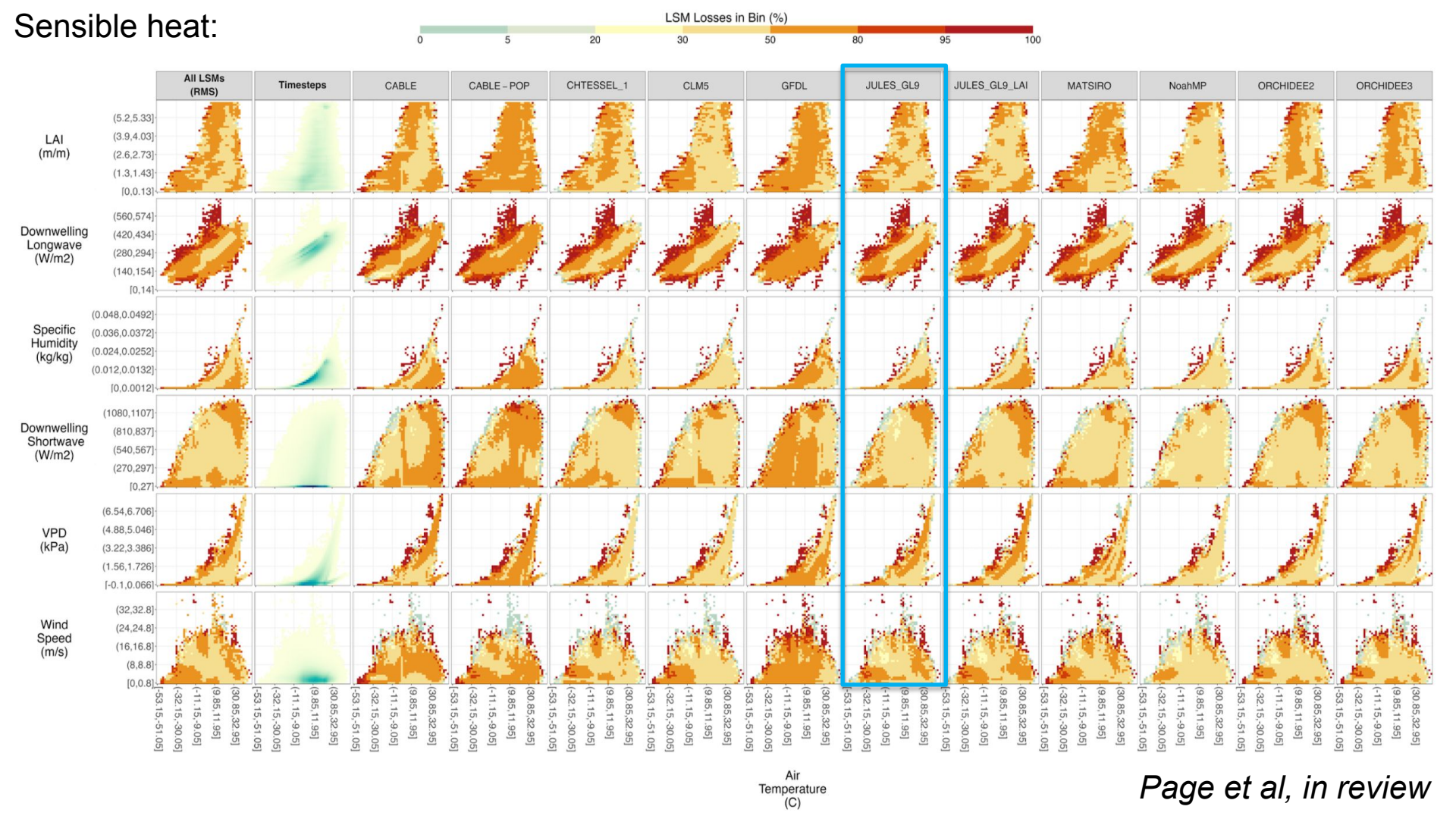
Null expectation is 25% of time steps

Yellow/ orange/ red
=>
We KNOW JULES performance can be improved

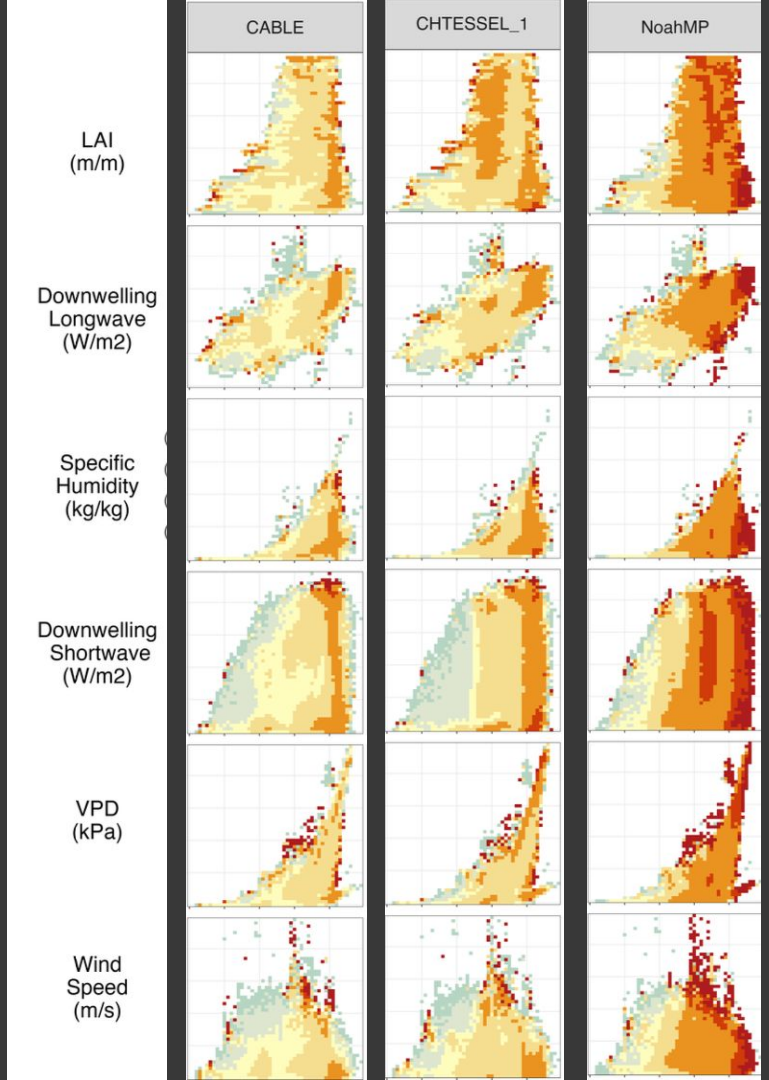


Page et al, in review

Sensible heat:



Page et al, in review

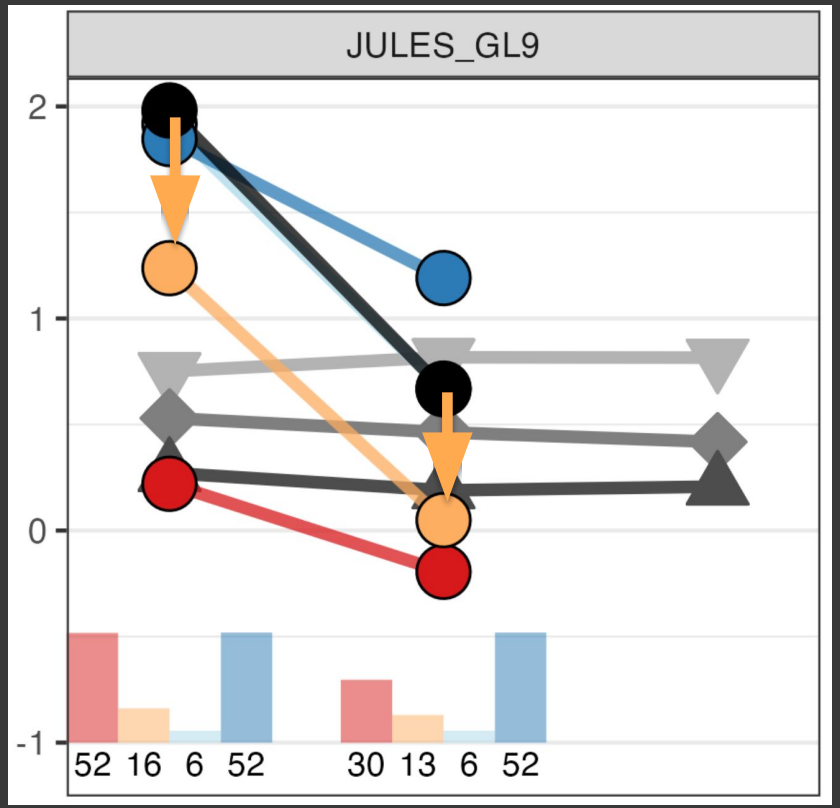


Temperature dependence of NEE performance

Page et al, in review

What happens if we remove these conditions from the analysis?

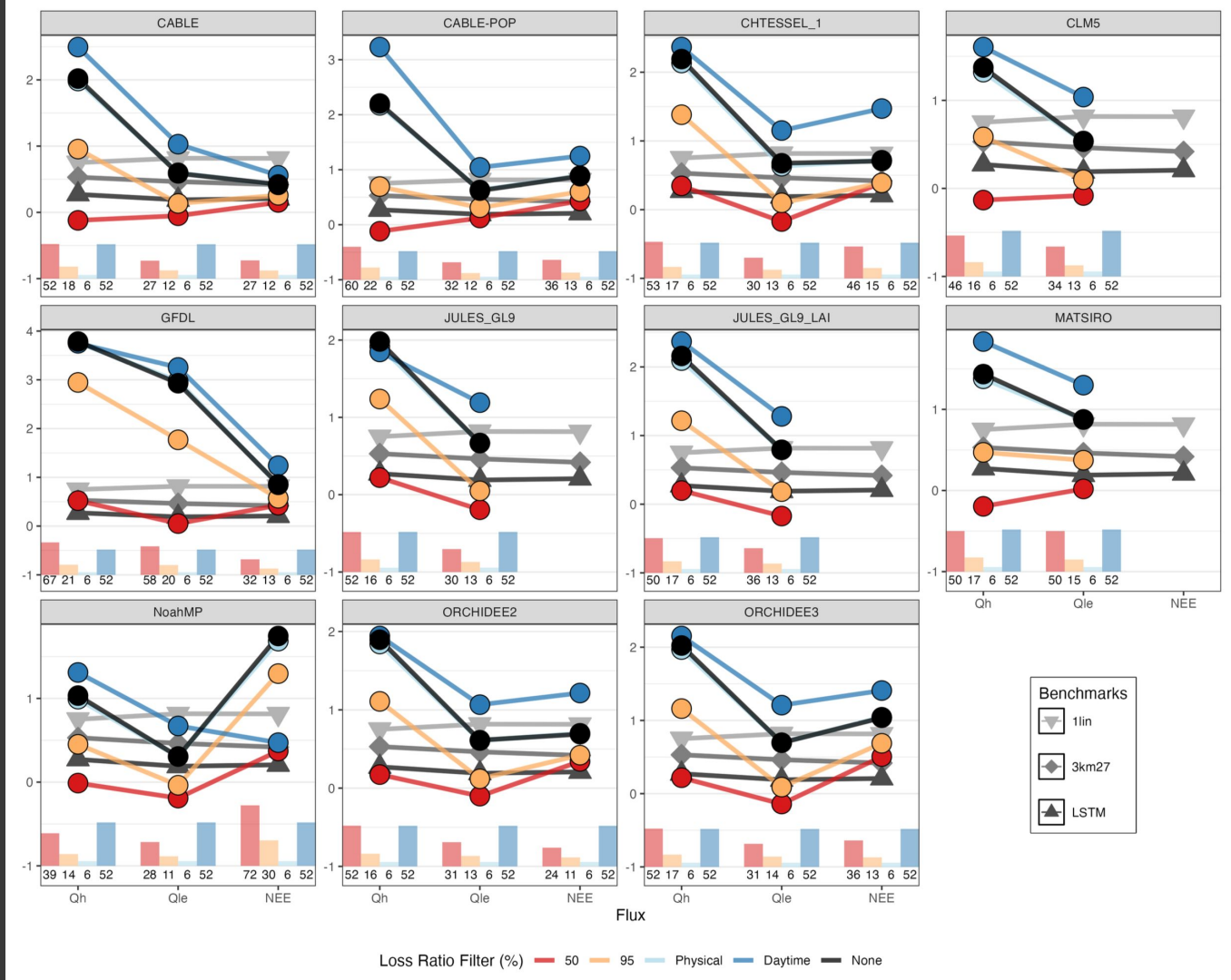
1. Remove met conditions where LSM loses 95%+
2. Remove conditions where LSM loses 50%+



Loss Ratio Filter (%) 50 95 Physical Daytime None

Page et al, in review

What happens if we remove these conditions from the analysis?



Page et al, in review

Solution?

We need an easy way for the community to access this kind of evaluation:

- Compare new model developments against a range of existing models
- Access to performance information *as capacity for improvement* (ML-based benchmarks)
- Broad range of metrics
- Evaluate multiple model use-cases at once
- Fast evaluation turnaround

Benchcab + modelevalution.org example

User wants to test development branch → Nominates test level: 1, 5, 42, 170 sites or ILAMB global → Nominates up to 3 comparison branches (typically trunk +)

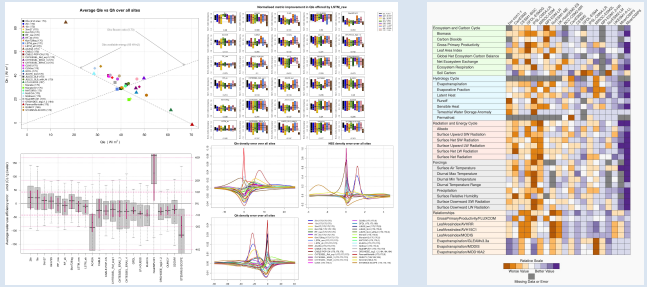
python benchcab

Automatic trunk head and branches built for e.g. 8 configurations → All 8 x 4 models run at N sites → Configuration, namelist characteristics integrated into netcdf global attributes

R

python ME.org

8 x 4 x N run simulation bundle pushed to LM benchmarking workspace, multiple site testing experiment → Analysis script decodes bundle, reports any discrepancies in conservation, configuration → URL of analyses returned to user



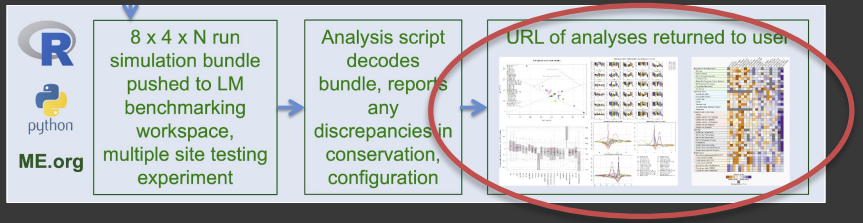
Current multi-site analysis page

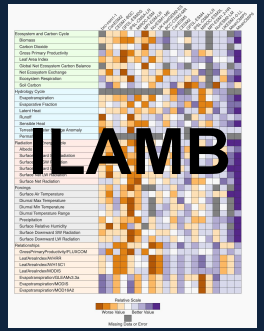
https://ncitest.modelevaluation.org/analysis/B8g9pFHq9RF5BSE2Y

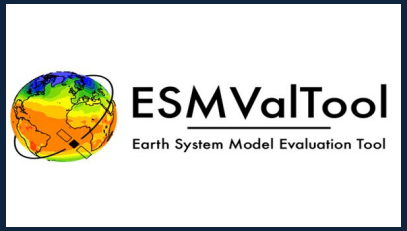
Analysis engines in modelevaluation.org

Analysis Package

python







METplus

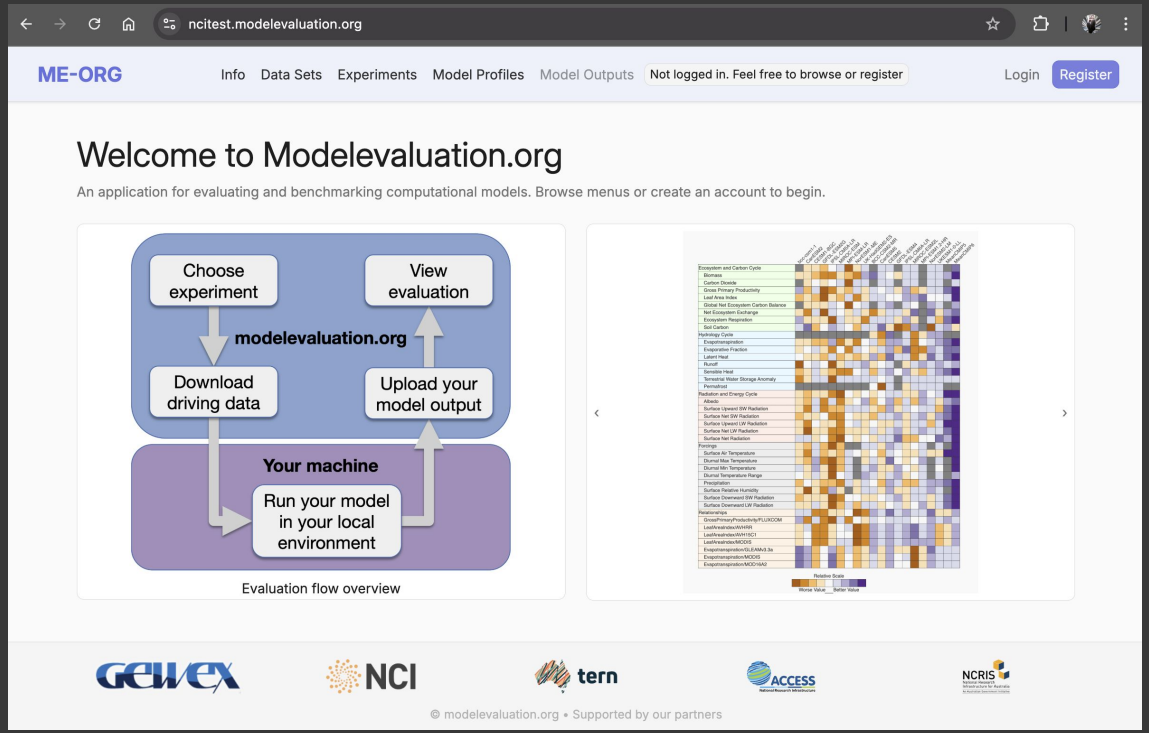
PLUMBER / PALS

CMEC

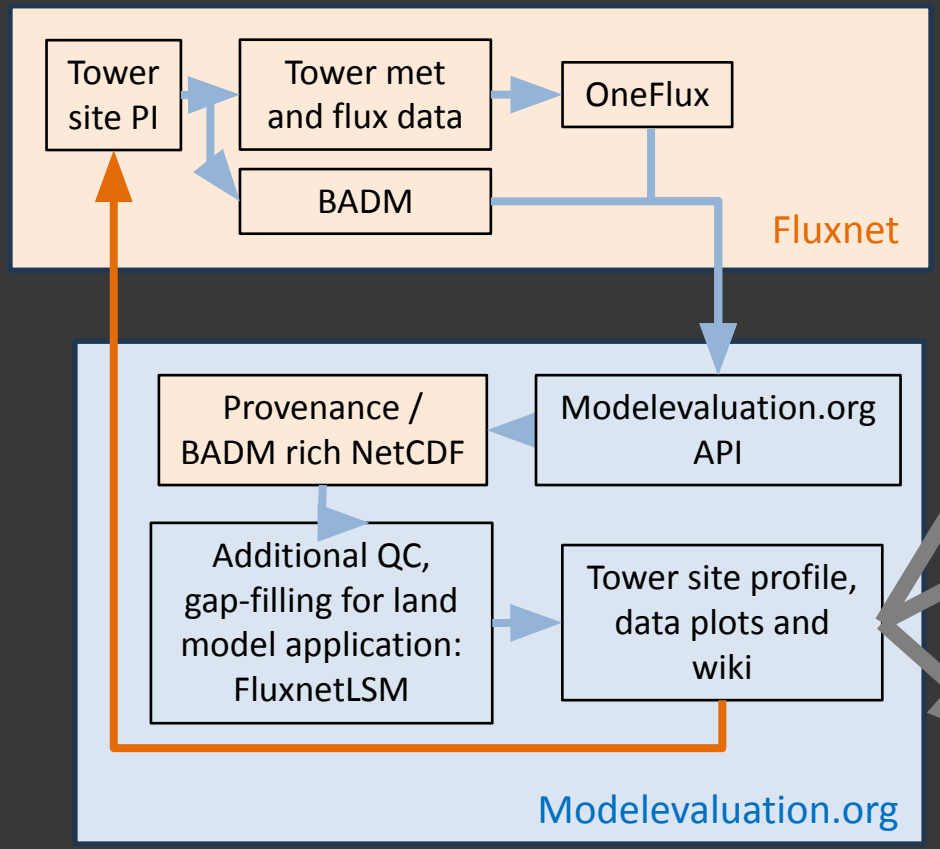
.....

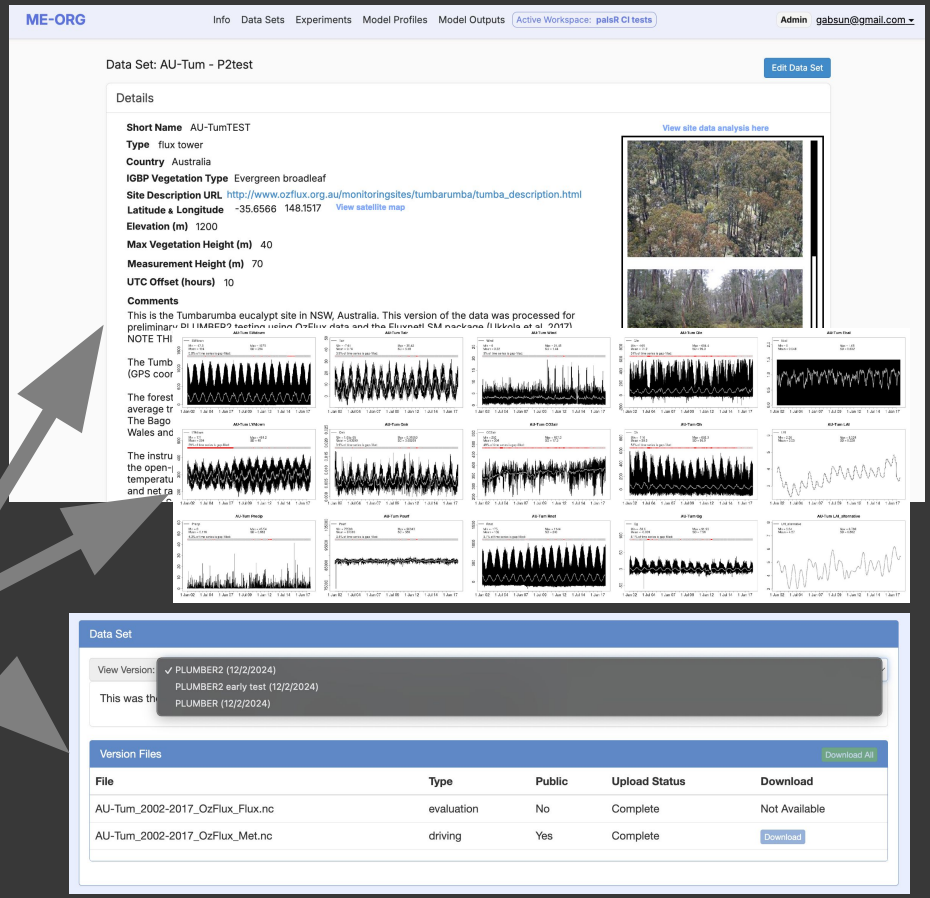
Modelevaluation.org

- Runs model analyses / benchmarks
- Manages relationship between data provenance / versioning, model repository version, model branches and configurations, multi-model comparisons and analysis results
- Hosted at Australian supercomputing facility (NCI) with HPC back-end capability; mirroring at other locations possible



Flux tower data pipelines in modelevaluation.org





Conclusions

- Mechanistic land models, including LSMs, are outperformed by relatively simple, out-of-sample empirical and ML models, in a broad range of metrics

- For now, no land model needs any representation of vegetation or soil to achieve its level of fidelity in flux prediction

- ML lets us understand exactly how much better we should expect land models to be, and in which circumstances – without it we would not know that there is a problem to solve.

- The FIRST step to resolving these issues is to build a (fast, comprehensive, interpretable) testing facility integrating these techniques. It is the only pathway to more robust land prediction (I think ☺️)