RegTune: A regional tuner for global atmospheric models

Vincent Larson and Zhun Guo

AMWG meeting

1 Feb 2023

Outline of talk

- Why tune?
- How can we reduce the cost of tuning?
- Typical tuning problems: A day in the life of a model developer
- Example tuning results from a global model (E3SM)

Why tune?

Oftentime, when we improve a model's formulation (i.e., the model structure), the solutions worsen. When that happens, we want to

- Retune, in order to restore the accuracy; and
- Get hints about what part of the model structure to change next, i.e. which model equations are still wrong.

We want to re-tune as quickly as possible, so that we can resume working on the model structure.

How can we reduce the cost of tuning?

By reducing the number of global model simulations.

To do so, RegTune uses a crude approximation of the cost function's parameter dependence (namely, a parabola).

To find this parabola, RegTune expands the model's parameter dependence in a Taylor series.

RegTune attempts to match observations in multiple user-selected regions of the globe. We hope that this will give insight into model structural error.

$$f\left(Sc; p_1, p_2\right) \approx f_{\text{obs}}\left(Sc\right)$$

$$f\left(Cu; p_1, p_2\right) \approx f_{\text{obs}}\left(Cu\right)$$

$$f\left(WP; p_1, p_2\right) \approx f_{\text{obs}}\left(WP\right)$$

f = a model-simulated field, e.g., cloud cover
p_1 and p_2 are model parameters
Sc denotes a stratocumulus region
Cu denotes a cumulus region
WP denotes the Western Pacific warm pool region

RegTune uses a Taylor expansion of the parameter dependence of the global model about the default values, $p_{1,def}$ and $p_{2,def}$:

$$\begin{bmatrix} f(Sc; p_{1,def}, p_{2,def}) \\ f(Cu; p_{1,def}, p_{2,def}) \\ f(WP; p_{1,def}, p_{2,def}) \end{bmatrix} + \underbrace{\begin{bmatrix} \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Sc} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Sc} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Cu} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Cu} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=WP} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=WP} \end{bmatrix}}_{\textbf{Sensitivity matrix}} \begin{bmatrix} \delta p_1 \\ \delta p_2 \end{bmatrix} + \ldots \approx \begin{bmatrix} f_{\text{obs}}(Sc) \\ f_{\text{obs}}(Cu) \\ f_{\text{obs}}(WP) \end{bmatrix}$$

RegTune retains the diagonal part of the quadratic term, but we won’t describe it in this talk.

The goal of tuning is to approximate the bias vector as a sum of column sensitivity vectors

$$\begin{bmatrix} \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Sc} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Sc} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Cu} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Cu} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=WP} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=WP} \end{bmatrix} \begin{bmatrix} \delta p_1 \\ \delta p_2 \end{bmatrix} \approx - \begin{bmatrix} \delta b(Sc) \\ \delta b(Cu) \\ \delta b(WP) \end{bmatrix}$$

model bias = default - obs

Tuning can’t remove the part of the bias that is orthogonal to the plane spanned by the column vectors. That part is the result of model structural error. It leads to tuning trade-offs.

db and *df/dp* can be approximated by running global simulations.

A day in the life of a model developer . . .

. . . (or a week or a month or a year!)

What follows are some problems that are typically encountered in the course of tuning a model.

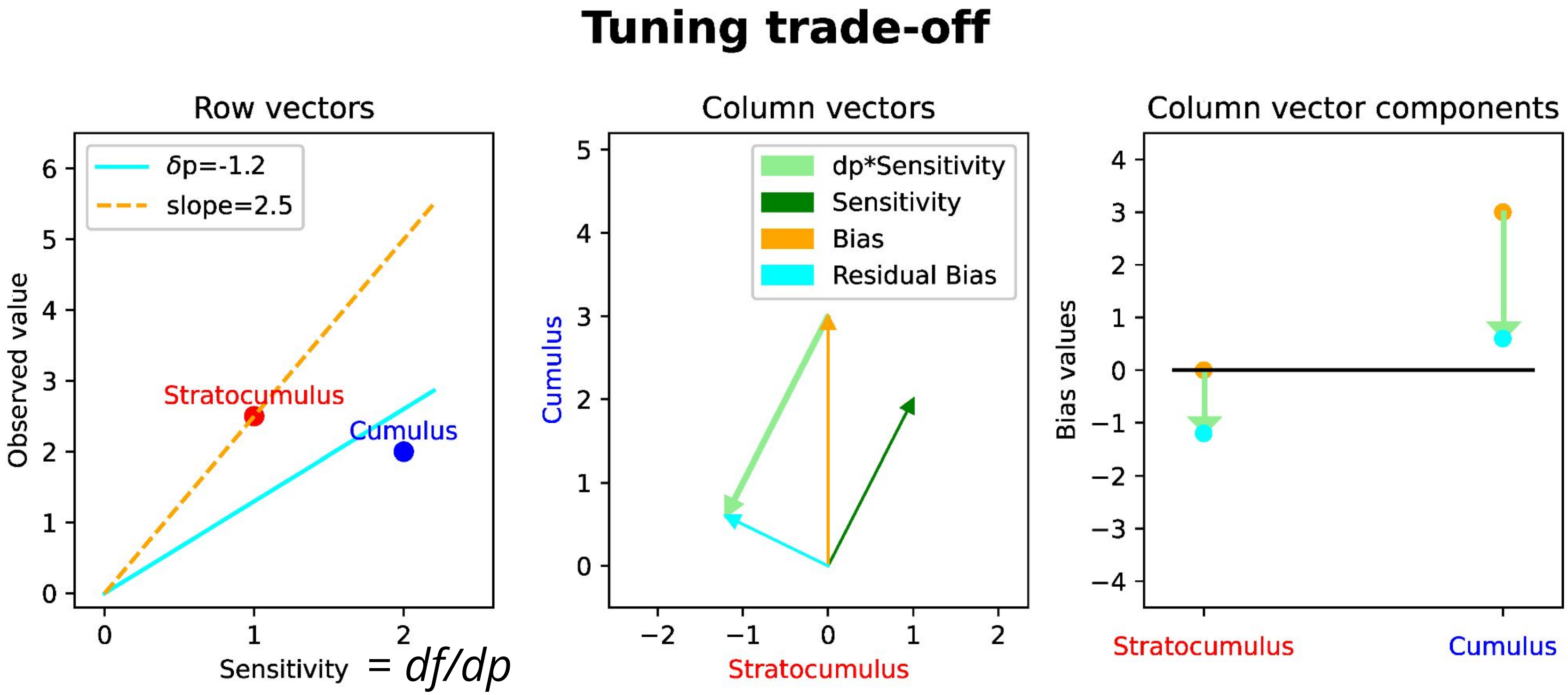
For simplicity, we reduce the problem to one dimension

$$\begin{bmatrix} \left.\frac{\partial f}{\partial p}\right|_{\mathbf{x}=Sc} \\ \\ \left.\frac{\partial f}{\partial p}\right|_{\mathbf{x}=Cu} \end{bmatrix} [\delta p] \approx - \begin{bmatrix} \delta b(Sc) \\ \\ \delta b(Cu) \end{bmatrix}$$

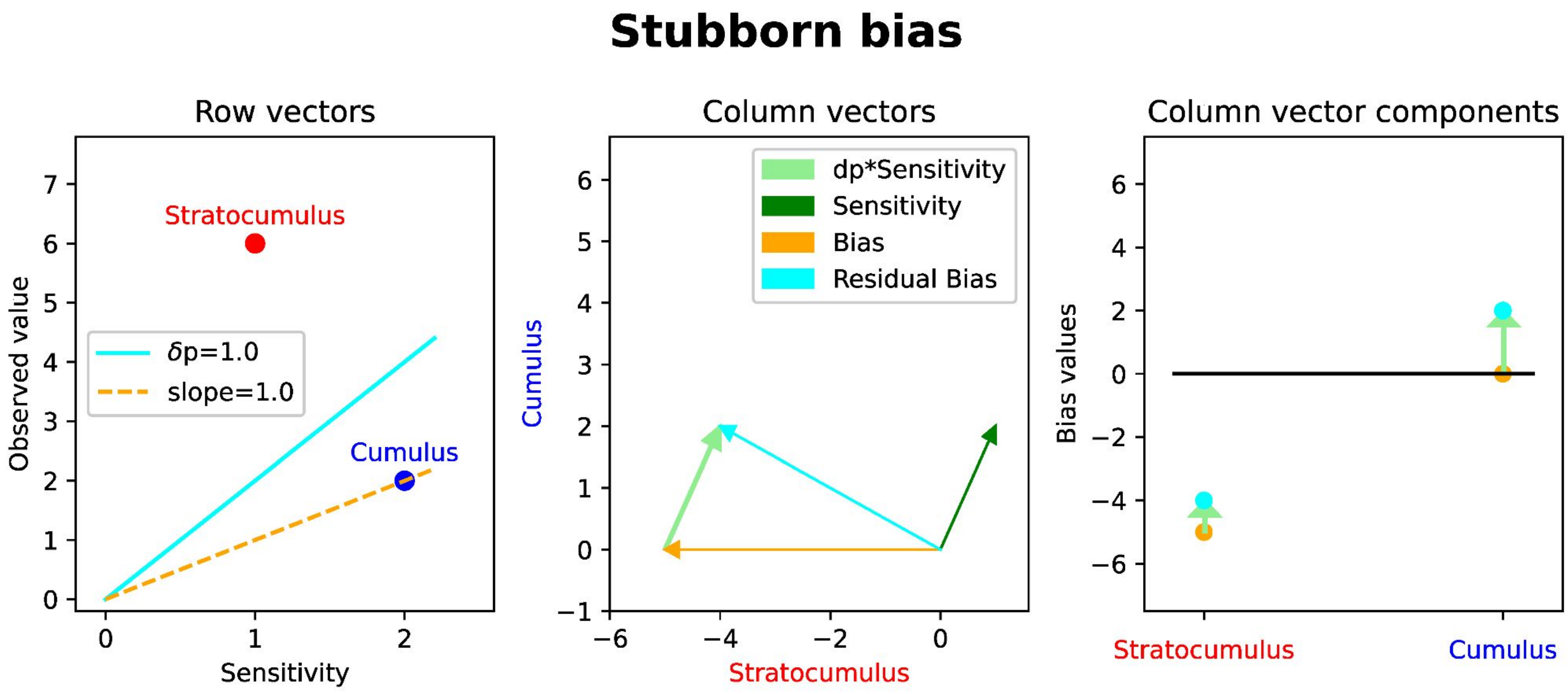
In this 1D problem, tuning means scaling the sensitivity column vector so that it best matches the bias vector.

A tuning trade-off occurs when the sensitivity vector is not parallel to the bias vector

$$\begin{bmatrix} \left.\frac{\partial f}{\partial p}\right|_{\mathbf{x}=Sc} \\ \\ \left.\frac{\partial f}{\partial p}\right|_{\mathbf{x}=Cu} \end{bmatrix} [\delta p] \approx - \begin{bmatrix} \delta b(Sc) \\ \\ \delta b(Cu) \end{bmatrix}$$



A stubborn bias is a large bias with little sensitivity



In this case, we need to find a new parameter or change the model structure.

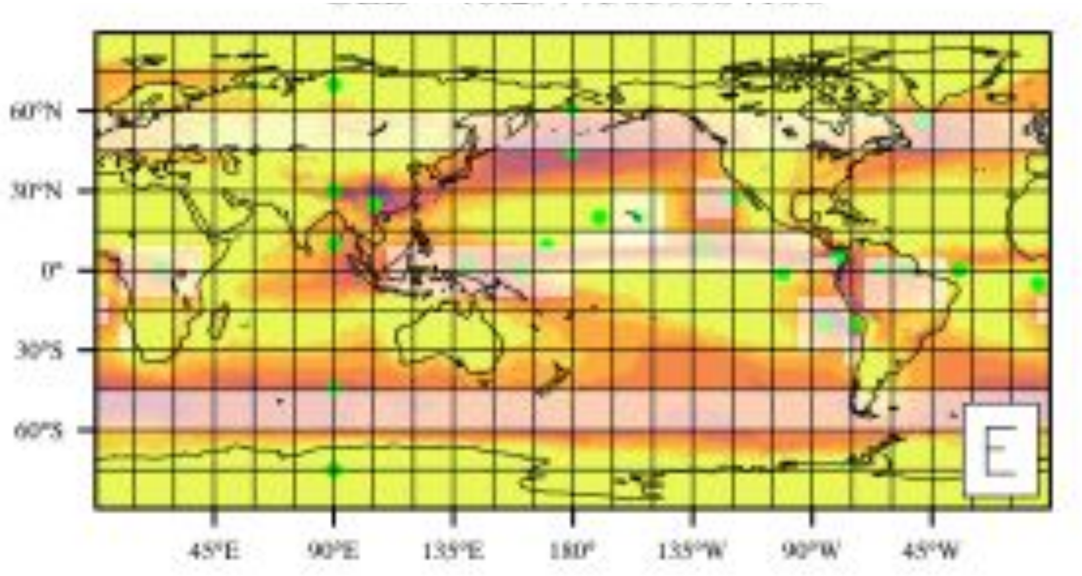
Example results from a global model, E3SM

We tune for 8 CLUBB parameters.

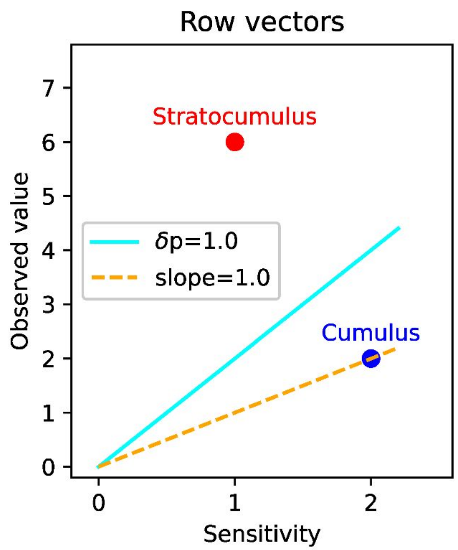
To form the sensitivity matrix, we need to run 2*8+1 global simulations. Each run lasts 14 months.

In this example, we attempt to match SWCF in various regions, plus globally averaged LWCF and PRECT.

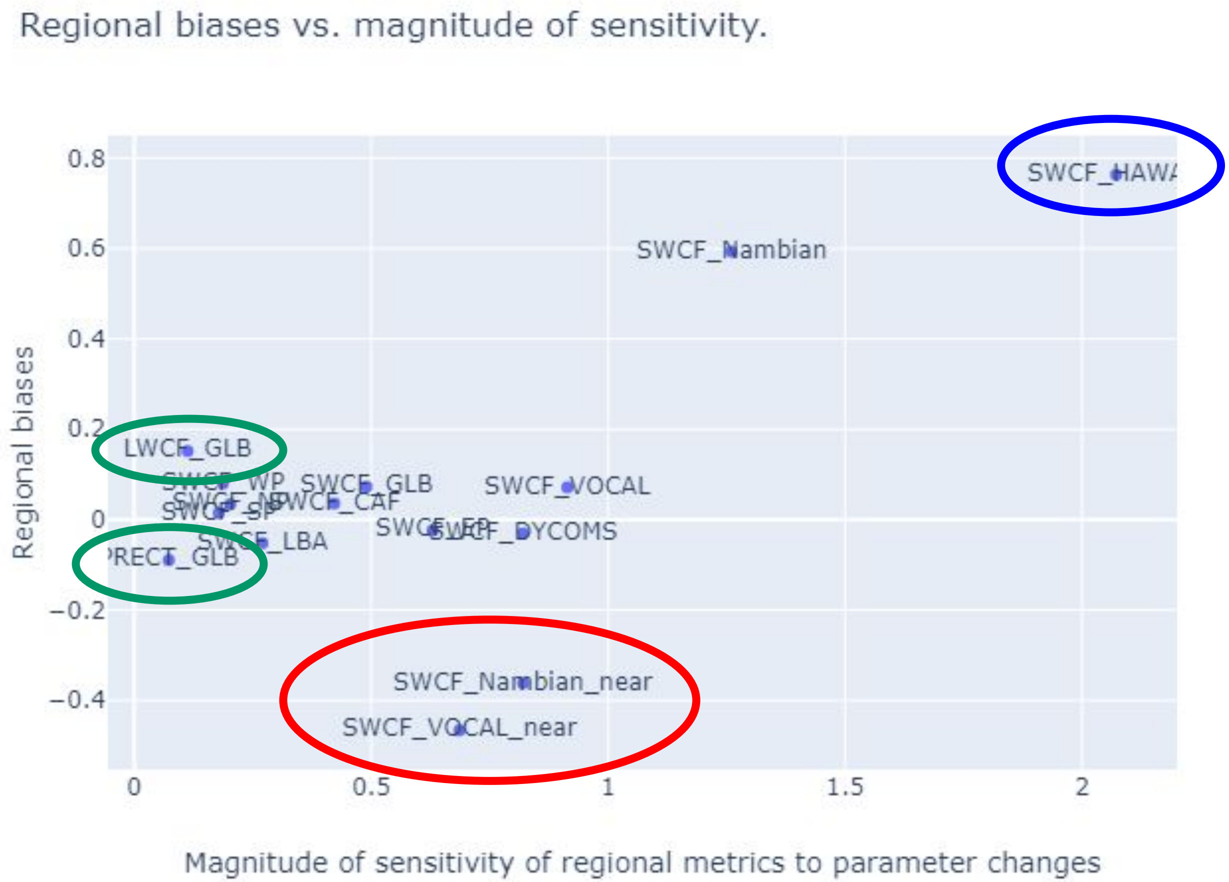
What follows are just some example plots for one particular configuration of flags. Please don't take the parameter recommend -ations seriously.



Even before we start tuning, we can see that LWCF and PRECT are stubborn biases; Hawaii and near-coastal Sc regions are trade-offs

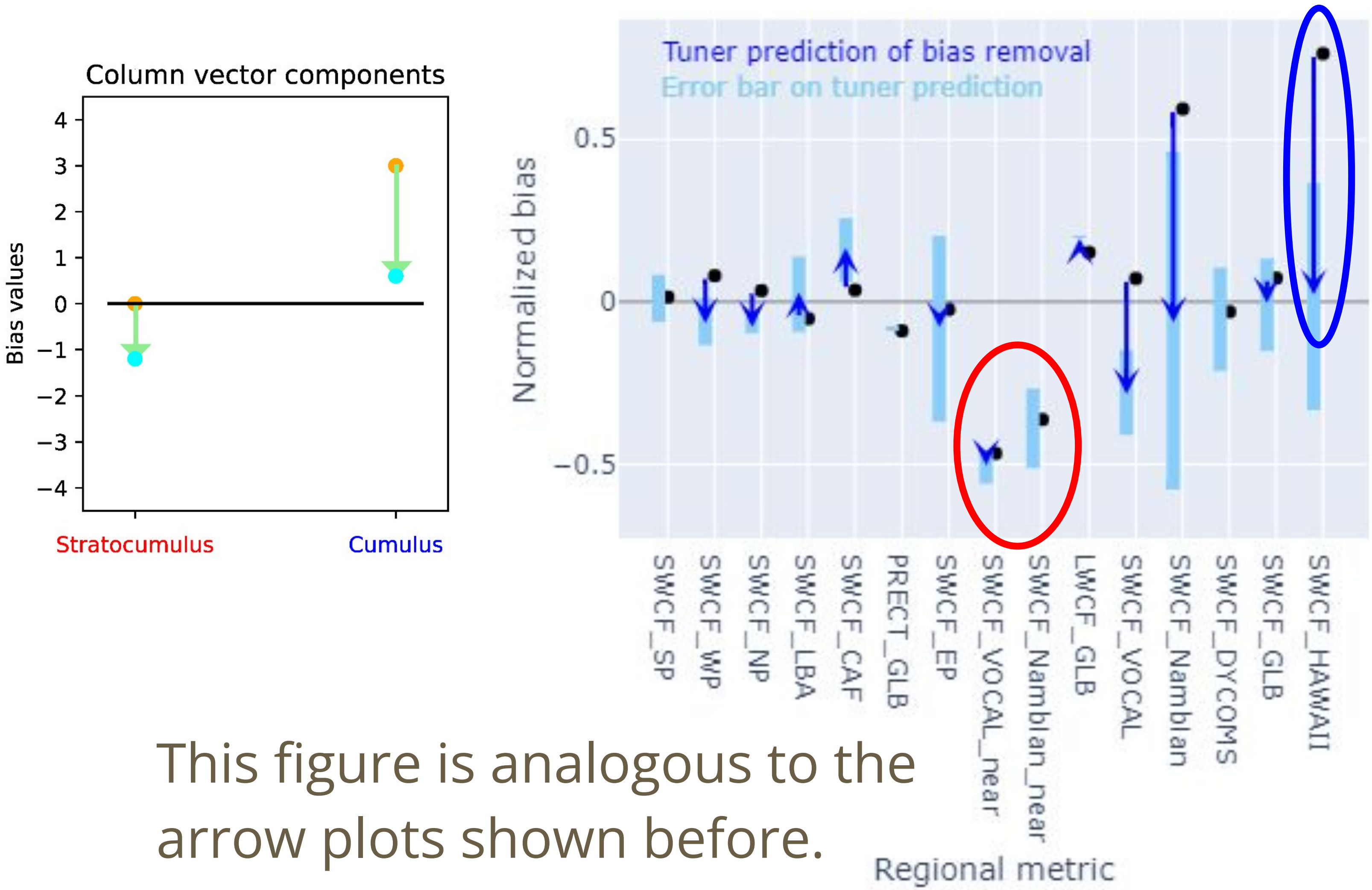


$$\begin{bmatrix} \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Sc} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Sc} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=Cu} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=Cu} \\ \left.\frac{\partial f}{\partial p_1}\right|_{\mathbf{x}=WP} & \left.\frac{\partial f}{\partial p_2}\right|_{\mathbf{x}=WP} \end{bmatrix} \begin{bmatrix} \delta p_1 \\ \delta p_2 \end{bmatrix} \approx - \begin{bmatrix} \delta b(Sc) \\ \delta b(Cu) \\ \delta b(WP) \end{bmatrix}$$



(This figure can be treated like the 1D case if the sensitivity vectors are positively correlated with each other.)

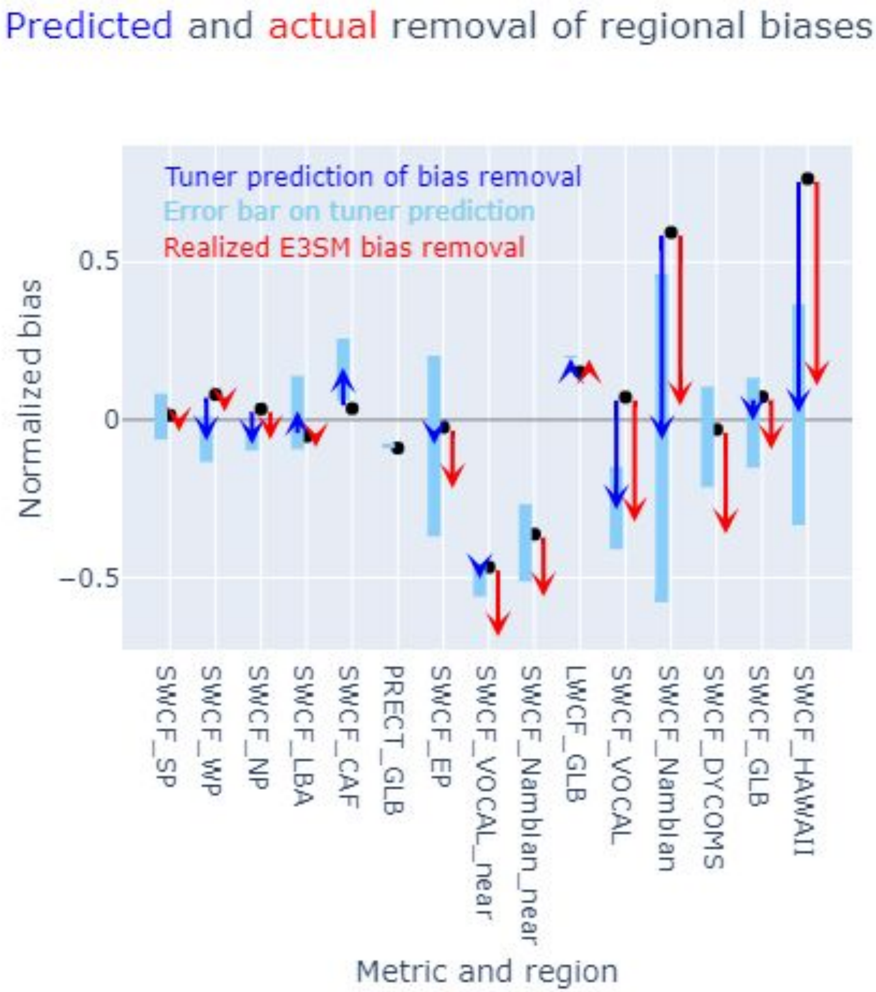
The tuner claims that the bias in Hawaii can be removed without damaging the near-coastal Sc regions



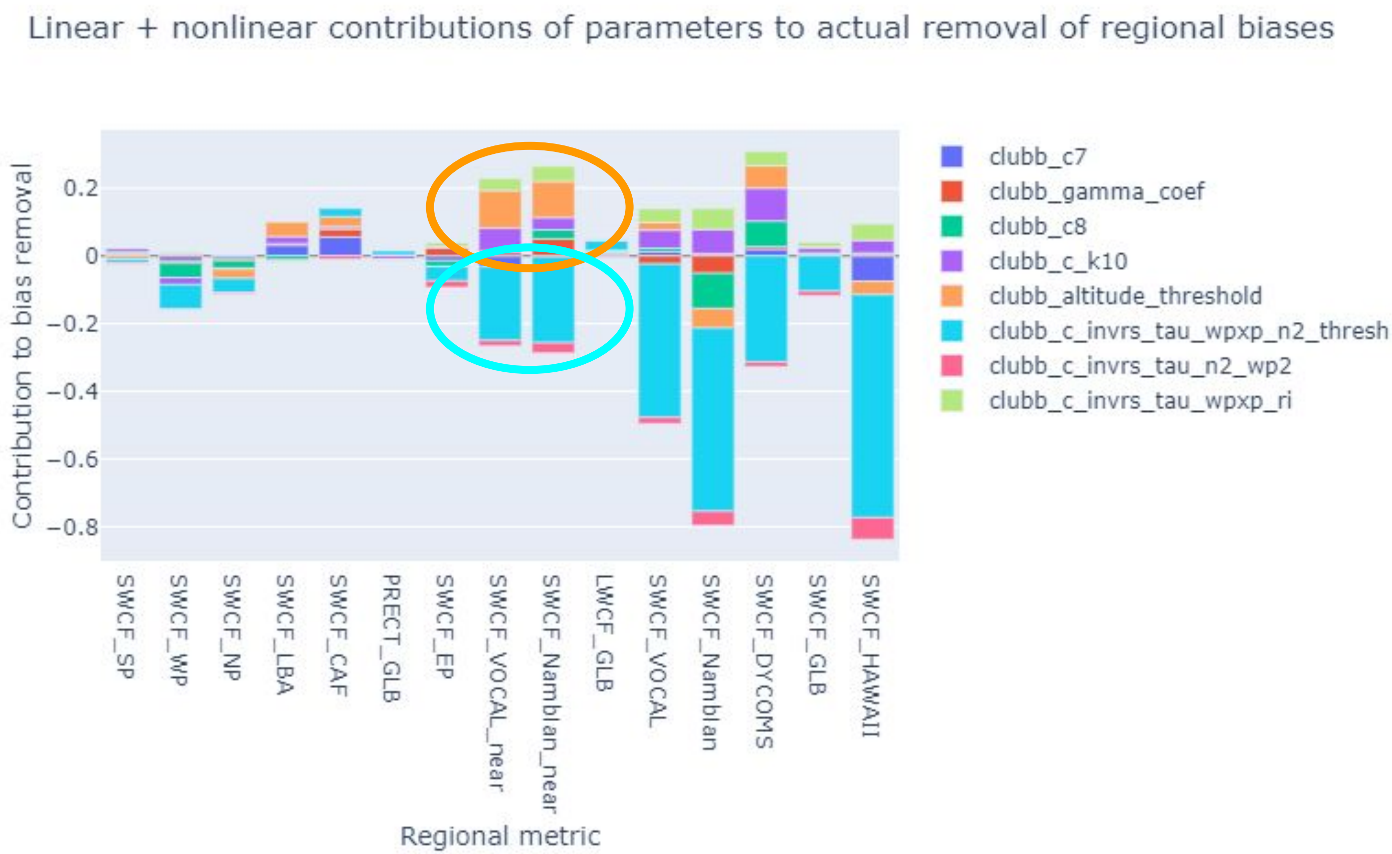
This figure is analogous to the arrow plots shown before.

Alas, the E3SM run does exhibit dimmed Sc. But the tuner's predictions are still useful

There are tuning trade-offs in the new solution, but overall it is improved, and we've saved time as compared to hand tuning.



RegTune assumes that altitude_thresh and n2_thresh will offset each other, but in fact they seem to interact. Does this indicate a problem with stably stratified layers near the surface?



Conclusions

- RegTune speeds up the process of tuning. It requires only 2n+1 global simulations.

- RegTune tells us which parameters matter for which regions (e.g., n2_thresh and altitude_thresh for near-coastal stratocumuli). This provides hints as to where to look for further potential structural model changes.