SAMA-IPSL

History Matching for the tuning of climate models: Lessons from the L96 model

Redouane Lguensat
Research Engineer IPSL
Paris, France

@redouanelg

EGU General Assembly 2022

MOPGA project: HRMES

Using machine learning to help tuning climate models

Modeling groups tune by hand models to make them match observations

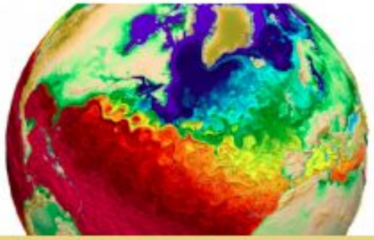
Very expensive, for example it took 5 years to find an acceptable tuning of IPSL model

Our goal :

- Use ML-based emulators to replace the expensive climate model
- find one or many good tunings
- run the expensive model with these “good” tunings

$\boldsymbol{x}$

Parameters



Climate model simulation code

$\boldsymbol{f}(\boldsymbol{x})$

Metrics, ex:

* Meridional Heat Transport
* AMOC
* Temperature profiles
* Etc…

Compare with

$\boldsymbol{z}$

Observations

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \|z - \boldsymbol{f}(\boldsymbol{x})\|_f$$

But ! tuning to a handful of metrics may risk achieving improved performance in those metrics at the expense of unphysical behavior in metrics or processes that were not used in tuning ——> **Overtuning**

« Overtuning is a real concern and the raison d'être for Bayesian UQ methods » Hourdin et al. 2017

History Matching:

* Established technique used in particle physics, molecular dynamics, population genetics, neuroscience, epidemiology, ecology, astrophysics and recently climate science
* Has always benefited from ML advances

Instead of looking for THE best set of parameters that solves an optimization problem. History Matching uses observed data to rule-out any parameter settings which are ``**implausible**".

History Matching and ML

Ideally we would individually check every possible parameter setting for the input:

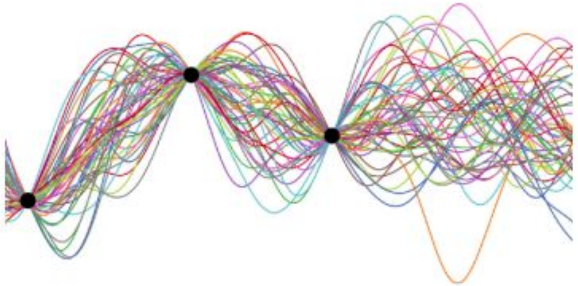
Impossible (climate models are **expensive** to run)

Need for **space-filling designs** to cover the space of parameter search

Ex: using Latin Hypercube Sampling

Need for replacing the expensive simulator with a rapid and cheap **emulator**

Surrogate modeling

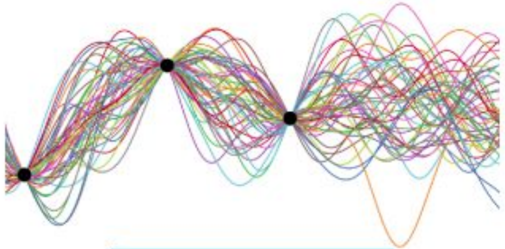


History Matching pipeline

Initial guess of parameter space

↓

Space-filling design → Calculate metrics → Training an Emulator → Calculate Implausibility



$$I(\boldsymbol{\theta}) = \frac{|E[f(\boldsymbol{\theta})] - z|}{\sqrt{\mathrm{Var}[f(\boldsymbol{\theta})]}}$$

Ex: using Latin Hypercube Sampling

Ex: using Gaussian Process Regression

↓

Rule out if I >3

(Pukelsheim's 3-sigma rule)

Set of plausible parameters

HM is an iterative process: done in *waves*

HM for climate model tuning

<u>Tuning atmospheric models:</u>

HM was used to tune **atmospheric models**, ex: LMDZ (Hourdin et al. 2020, Couvreux et al. 2020):

* Using single-column models (SCMs) they afford to run several simulations with different set of parameters
* Short timescales

JAMES Journal of Advances in Modeling Earth Systems

Research Article | Open Access

Process-based climate model development harnessing machine learning: I. a calibration tool for parameterization improvement

Fleur Couvreux, Frédéric Hourdin, Daniel Williamson, Romain Roehrig ... See all authors

First published: 07 December 2020 | https://doi.org/10.1029/2020MS002217

JAMES Journal of Advances in Modeling Earth Systems

Research Article | Open Access

Process-based climate model development harnessing machine learning: II. model calibration from single column to global

Frédéric Hourdin, Daniel Williamson, Catherine Rio, Fleur Couvreux ... See all authors

First published: 04 December 2020 | https://doi.org/10.1029/2020MS002225

<u>Tuning ocean models:</u>

HM was used to tune **ocean models**, ex: NEMO ORCA 2° (Williamson et al. 2017):

* Using an available ensemble of 400 NEMO simulations ran for 150 years.
* Long timescales

Geosci. Model Dev., 10, 1789–1816, 2017
www.geosci-model-dev.net/10/1789/2017/
doi:10.5194/gmd-10-1789-2017
© Author(s) 2017. CC Attribution 3.0 License.

Geoscientific Model Development

Tuning without over-tuning: parametric uncertainty quantification for the NEMO ocean model

Daniel B. Williamson[1], Adam T. Blaker[2], and Bablu Sinha[2]

[1]College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, UK
[2]National Oceanography Centre, Southampton, SO14 3ZH, UK

Correspondence to: Daniel B. Williamson (d.williamson@exeter.ac.uk)

Received: 20 July 2016 – Discussion started: 30 August 2016
Revised: 24 November 2016 – Accepted: 30 January 2017 – Published: 27 April 2017

The Lorenz96 as a toy model

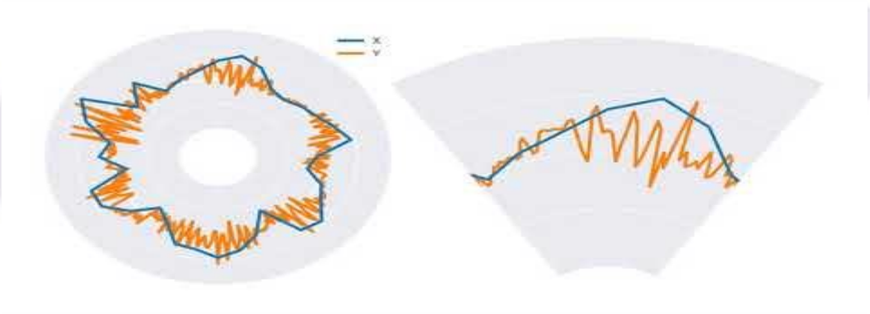
* Periodic system of K (k=1,…,K) ODEs
* **Two-level version:** add periodic variable Y with its own set of ODEs.
* The X and Y ODEs are linked through coupling terms. Each X has J Y variables associated with it.

$$\frac{dX_k}{dt} = \underbrace{-X_{k-1}\left(X_{k-2} - X_{k+1}\right)}_{\text{Advection}} \underbrace{-X_k}_{\text{Diffusion}} \underbrace{+F}_{Forcing} \underbrace{-\frac{hc}{b}\sum_{j=1}^{J} Y_{j,k}}_{Coupling}$$

$$\frac{dY_{j,k}}{dt} = \underbrace{-cbY_{j+1,k}\left(Y_{j+2,k} - Y_{j-1,k}\right)}_{\text{Advection}} \underbrace{-cY_{j,k}}_{\text{Diffusion}} \underbrace{+\frac{hc}{b}X_k}_{\text{Coupling}}$$

Analogy with coupled ocean-atmosphere models:

Atmosphere	Fast dynamics	Y
Ocean	Slow dynamics	X



Credits: Stephan Rasp

Experiment: HM for the tuning of parameters (F, h, c, b)

HM on the Lorenz96

* **<u>Metrics</u>**: long-term time means to mimic climatological quantities
* **<u>Ground Truth</u>:** perfect setting K=36 **X** variables each coupled with J=10 **Y** variable. F=10, h=1, c=10, b=10, chaotic behavior.
* **<u>HM code:</u>** R + Python code
  ! Parallel computation + GPR models can be trained on **GPU**

$$f(X,Y) = \begin{pmatrix} X \\ \bar{Y} \\ X^2 \\ X\bar{Y} \\ \bar{Y}^2 \end{pmatrix}$$

Justified by energy conservation constraints, check Schneider et al. 2017 for details (ESM 2.0 paper)

180-dimensional vector

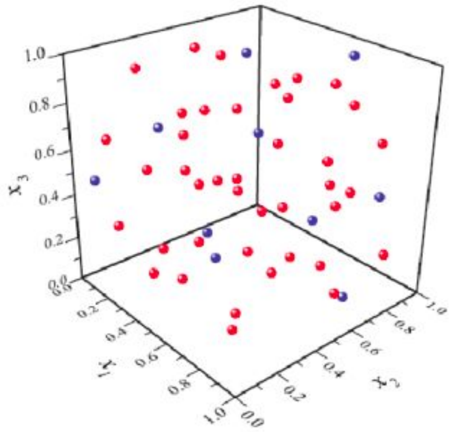
HM on the Lorenz96

Initial guess of parameter space

Params	Prior	True
F	[-20,20]	10
h	[-2,2]	1
c	[0,20]	10
b	[-20,20]	10

40 samples from a LHS →

Space filling design



Run the L96 model

Build a training database for the emulator:

x_train.size=(40,4)

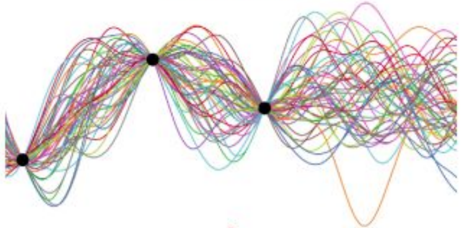
y_train.size=(40,180)

Use PCA

y_train.size=(40,8)

← Here, one GP Per output

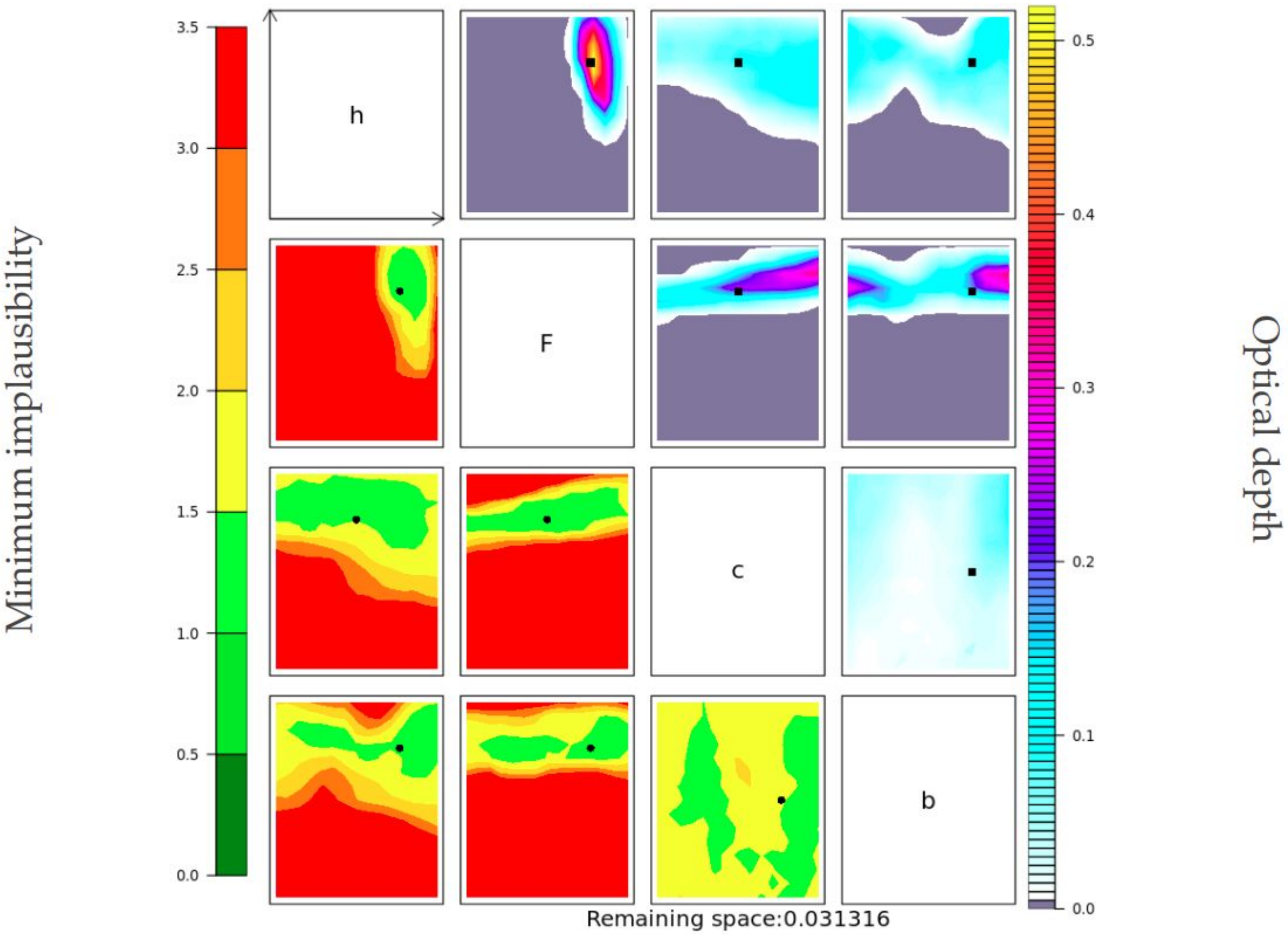
Train the emulator then use it for inference on a large number of samples



Calculate Implausibility

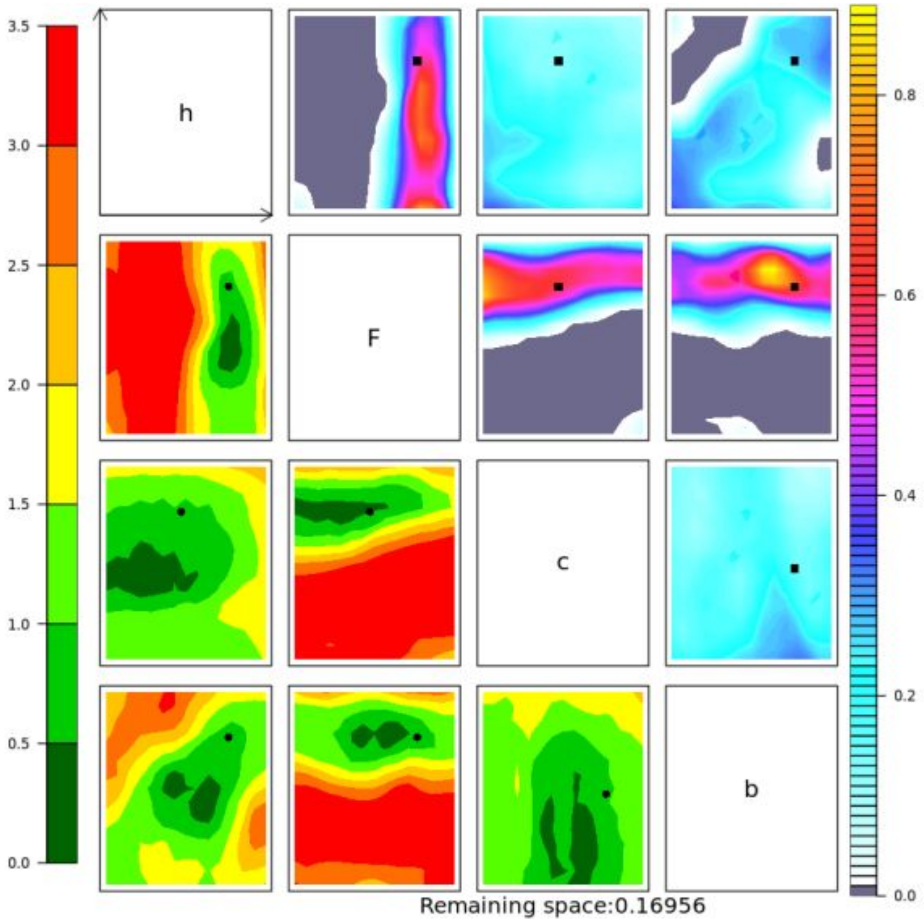
$$I(\boldsymbol{\theta}) = \frac{|E[f(\boldsymbol{\theta})] - z|}{\sqrt{\mathrm{Var}[f(\boldsymbol{\theta})]}}$$

explaining NROY plots

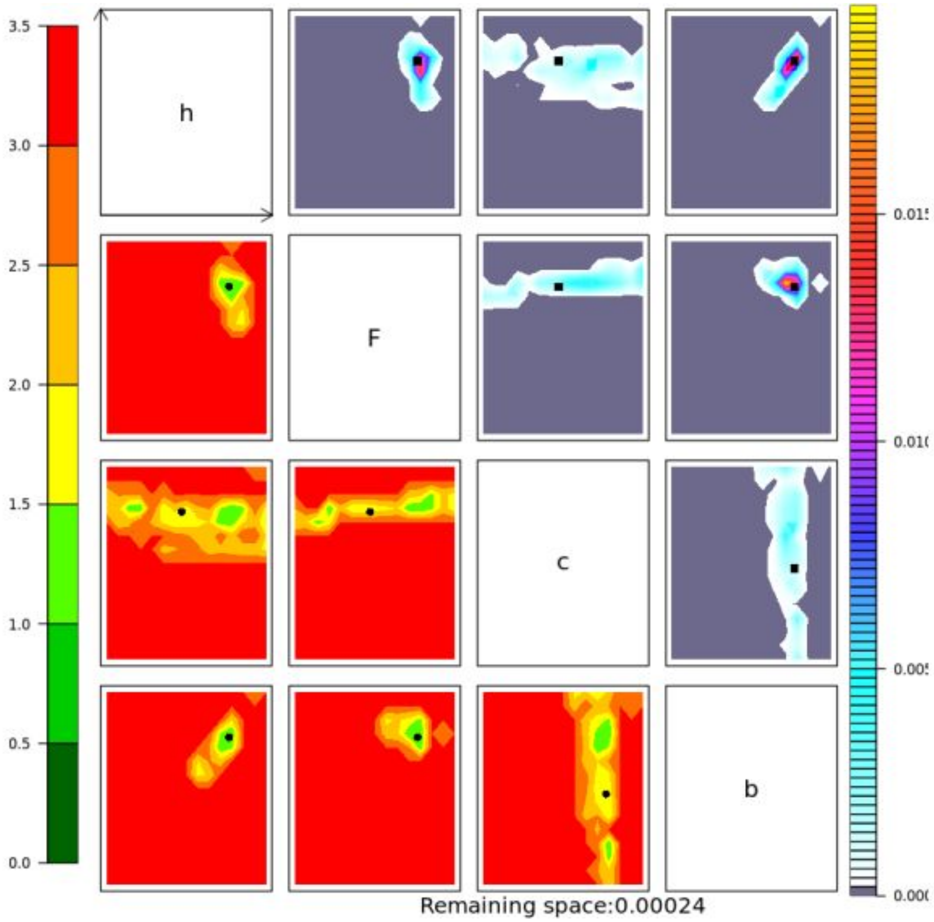


HM on the Lorenz96

wave	NROY (%)	NbSim
1	16.96	40
2	7.91	40
3	5.55	40
4	2.31	40
5	0.94	40
6	0.02	40
		Total = 240



wave 1



wave 6

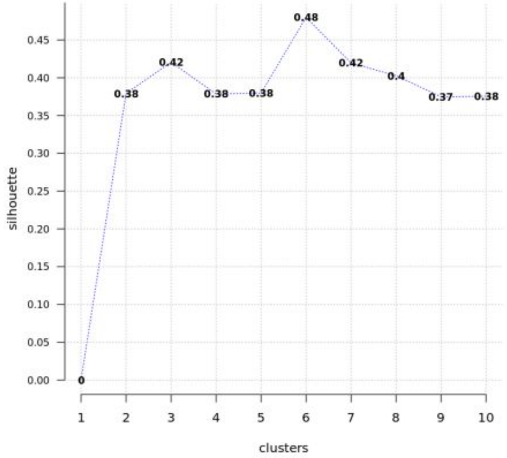
what to do with the NROY?

Open question, depends heavily on the computational budget

In case of a very short budget, an idea can be the use of **clustering** and selecting cluster centers

Here we use ***K-means*** and find six clusters

verification step where we make sure the six cluster centers belong to the last NROY

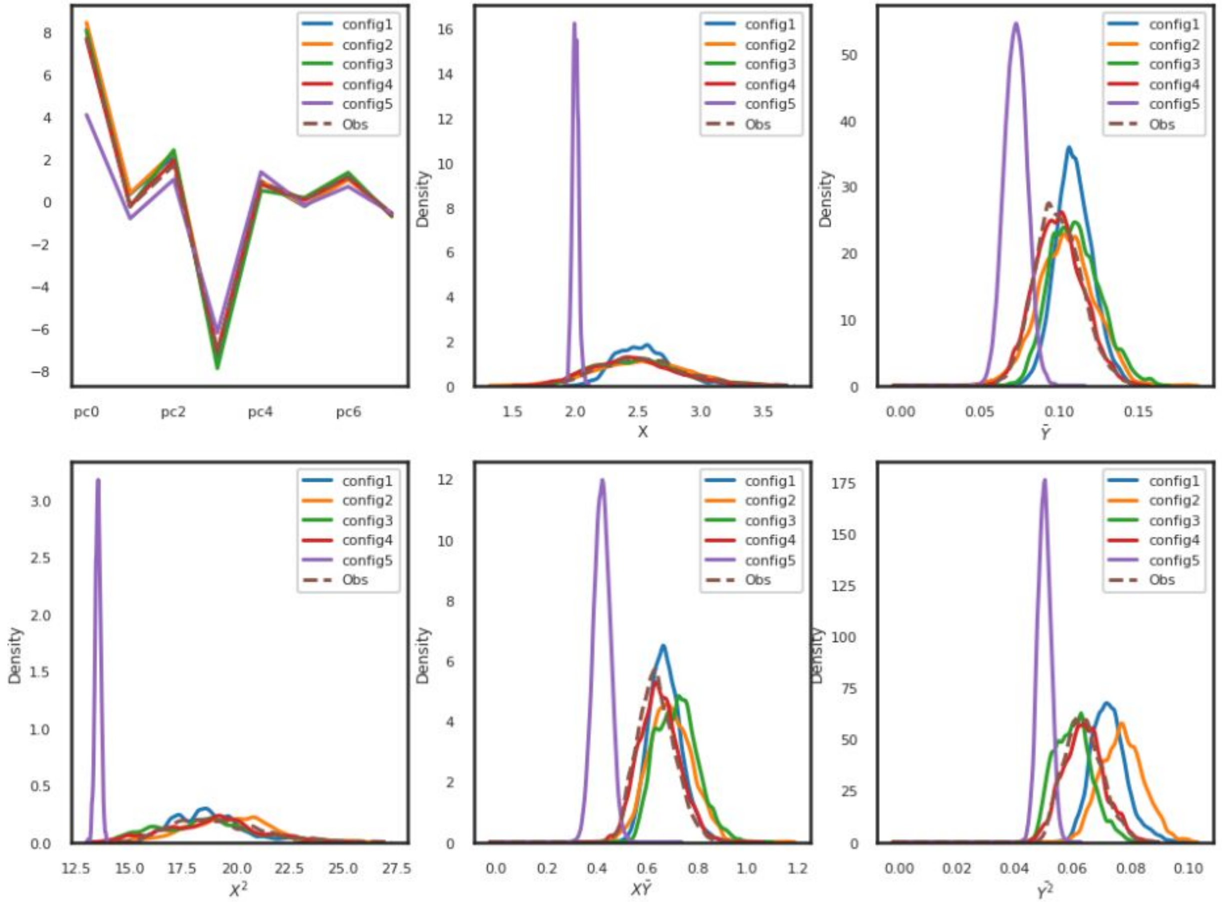


	h	F	c	b
1	0.99	11.90	16.21	9.17
2	1.15	8.94	3.94	10.33
3	0.58	10.31	11.90	6.89
4	0.92	10.31	10.28	9.35
5	1.87	11.52	19.03	15.65

centers of clusters

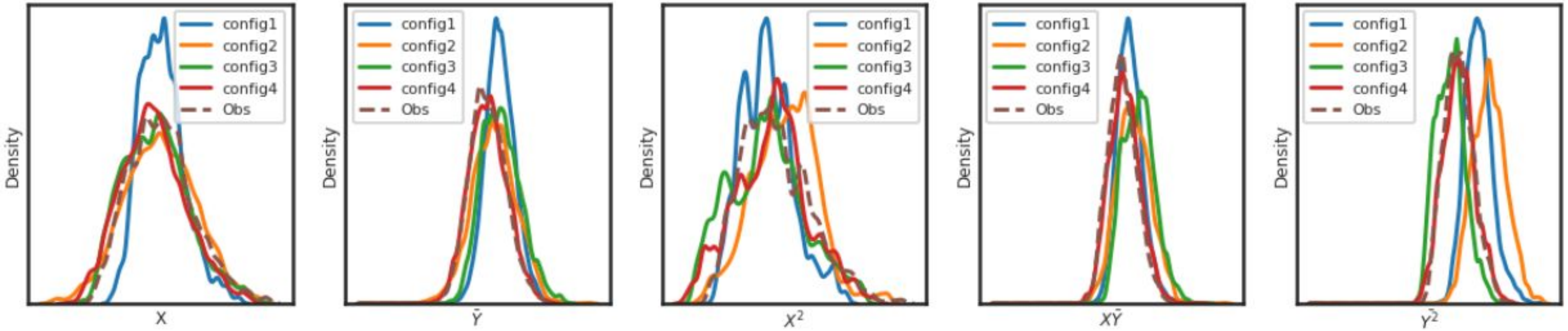
HM on the Lorenz96

We take the set of parameters and **run the real Lorenz96 simulation then check the metrics and compare them with observations**



HM on the Lorenz96

We take the set of parameters and **run the real Lorenz96 simulation then check the metrics and compare them with observations**



	h	F	c	b	KL-div
1	0.99	11.90	16.21	9.17	0.13
2	1.15	8.94	3.94	10.33	0.09
3	0.58	10.31	11.90	6.89	0.15
4	0.92	10.31	10.28	9.35	**0.01**

Lessons learned

- History Matching is a powerful tool that can **assist (and not replace)** the expertise of domain-based tuning

- **Setting a computational budget in advance** is important and plays an important role in the final result

- Incorporate as much **domain expertise** as you can in the History Matching process

- There are still many open questions that could make History Matching more efficient for climate models (dimensionality reduction, non-convex NROY clusters, etc.)

Our review paper on Machine Learning for ocean science

ENVIRONMENTAL RESEARCH
LETTERS

TOPICAL REVIEW

Bridging observations, theory and numerical simulation of the ocean using machine learning

Maike Sonnewald[1,2,3,9,*], Redouane Lguensat[4,5], Daniel C Jones[6], Peter D Dueben[7], Julien Brajard[5,8] and V Balaji[1,2,4]

1 Princeton University, Program in Atmospheric and Oceanic Sciences, Princeton, NJ 08540, United States of America
2 NOAA/OAR Geophysical Fluid Dynamics Laboratory, Ocean and Cryosphere Division, Princeton, NJ 08540, United States of America
3 University of Washington, School of Oceanography, Seattle, WA, United States of America
4 Laboratoire des Sciences du Climat et de l'Environnement (LSCE-IPSL), CEA Saclay, Gif Sur Yvette, France
5 LOCEAN-IPSL, Sorbonne Université, Paris, France
6 British Antarctic Survey, NERC, UKRI, Cambridge, United Kingdom
7 European Centre for Medium Range Weather Forecasts, Reading, United Kingdom
8 Nansen Center (NERSC), Bergen, Norway
9 Present address: Princeton University, Program in Atmospheric and Oceanic Sciences, 300 Forrestal Rd., Princeton, NJ 08540, United States of America
* Author to whom any correspondence should be addressed.

E-mail: maikes@princeton.edu

Keywords: ocean science, physical oceanography, observations, theory, modelling, supervised machine learning, unsupervised machine learning