

Computer Model Calibration for Studying the West Antarctic Ice Sheet

Murali Haran

Department of Statistics, Pennsylvania State University

Meeting of the International Statistical Institute (ISI)
Marrakech, Morocco. July 2017.

Collaborators:

Won Chang (Dept of Statistics, University of Chicago)

Patrick Applegate (Penn State Institutes for Energy and Environment)

David Pollard (Penn State Institutes for Energy and Environment)

The Loneliness of Polar Bears



Photo from mrwallpaper.com

Ice Sheets

- ▶ Enormous mass of glacial land ice
- ▶ Antarctic ice sheet is over 14 million km².
 - ▶ ≈ United States + Mexico combined.
- ▶ Greenland ice sheet over 1.7 million km²
- ▶ Combine for over 99% of the freshwater ice in the world
- ▶ Can have major impacts on sea level rise:
 - ▶ Melting entire Antarctic ice sheet: sea level rise ≈ 57 m.
 - ▶ Melting entire Greenland ice sheet: sea level rise ≈ 7 m
- ▶ Even a modest contribution to sea level rise can have a major impact on humans, e.g. on low-lying areas and areas prone to storm surges.

SCIENCE

Climate Model Predicts West Antarctic Ice Sheet Could Melt Rapidly

By JUSTIN GILLIS MARCH 30, 2016

For half a century, climate scientists have seen the West Antarctic ice sheet, a remnant of the last ice age, as a sword of Damocles hanging over human civilization.

The great ice sheet, larger than Mexico, is thought to be potentially vulnerable to disintegration from a relatively small amount of global warming, and capable of raising the sea level by 12 feet or more should it break up. But researchers long assumed the worst effects would take hundreds — if not thousands — of years to occur.

Now, new research suggests the disaster scenario could play out much sooner.

Continued high emissions of heat-trapping gases could launch a disintegration of the ice sheet within decades, according to a study published Wednesday, heaving enough water into the ocean to raise the sea level as much as three feet by the end of this century.

Statistics and Ice Sheets

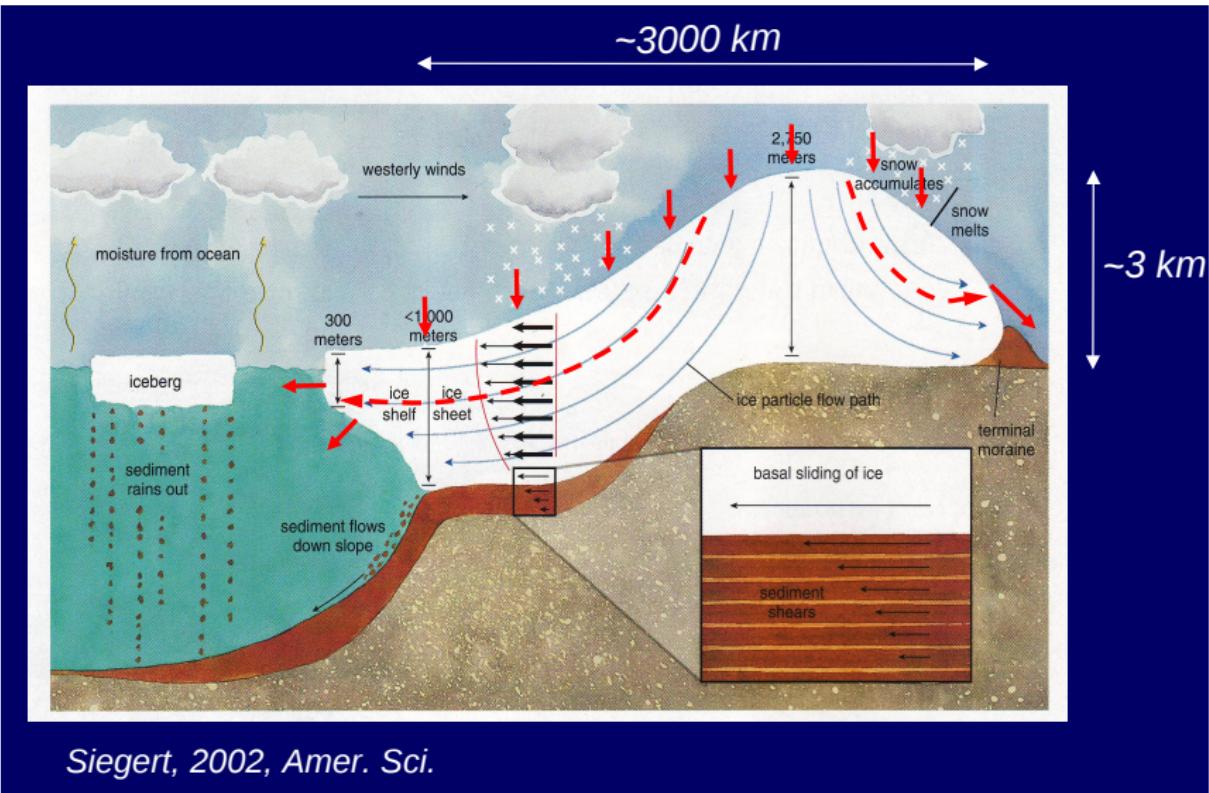
Formulating reasonable hypotheses regarding climatic change requires physical insight and ingenuity, but subsequently testing these hypotheses demands quantitative computation. – Edward N. Lorenz (1970)

- ▶ My translation: to study future climate we need physical models, data, and statistics. Innovation on all three fronts.
- ▶ Focus here: The West Antarctic Ice Sheet (WAIS).
- ▶ WAIS has the potential to be a significant contributor to future sea level change.

Talk Summary

- ▶ How can we project the future behavior of the West Antarctic Ice Sheet?
 - ▶ Ice sheet model: PSU3D-ICE (Pollard and DeConto, 2009).
- ▶ Key model input parameters are uncertain
- ▶ Observations:
 1. Satellite data on the modern ice sheet.
 2. Paleo reconstructions of ice sheet from 25,000 years ago to present time.
- ▶ Our research: methods to use observations of the ice sheet to infer important parameters of the ice sheet model.
- ▶ Challenges
 1. Two sets of data: spatial and temporal binary/non-Gaussian data. “Data” = Observations and computer model output.
 2. High-dimensional spatial data
- ▶ I will describe new methods to address these issues.

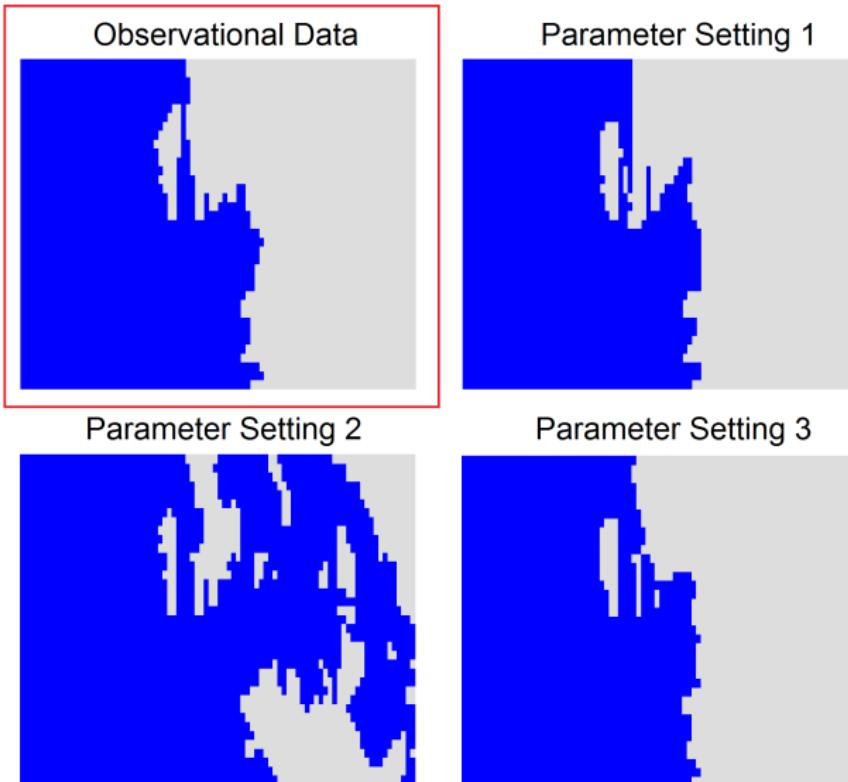
Ice Sheet Physics



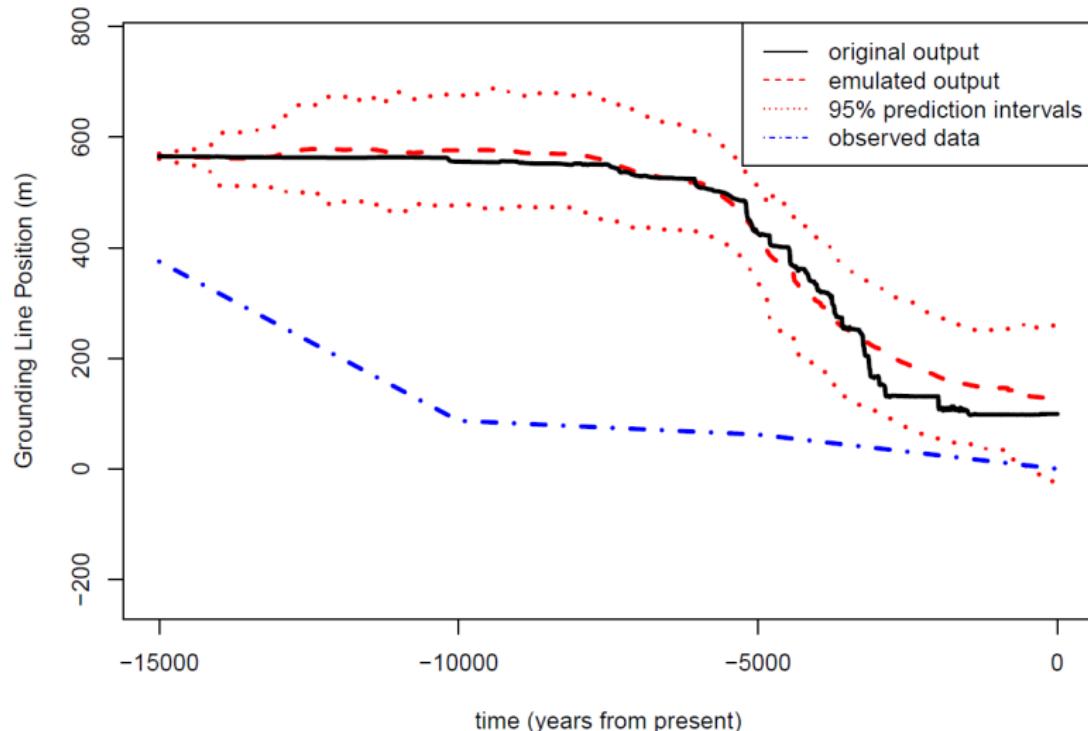
Ice Sheet Model Parameters

- ▶ The ice sheet's behavior is complex.
- ▶ Model equations predict ice flow, thickness, temperatures, and bedrock elevation, through thousands to millions of years.
- ▶ Examples of key model parameters:
 - ▶ Ocean melt coefficient: sensitivity of ice sheet to temperature change in the surrounding ocean
 - ▶ Strength of the “calving” process. Calving = where ice breaks off and transitions from attached to floating
 - ▶ “Slipperiness” of the ocean floor

Satellite Observations versus Model Output

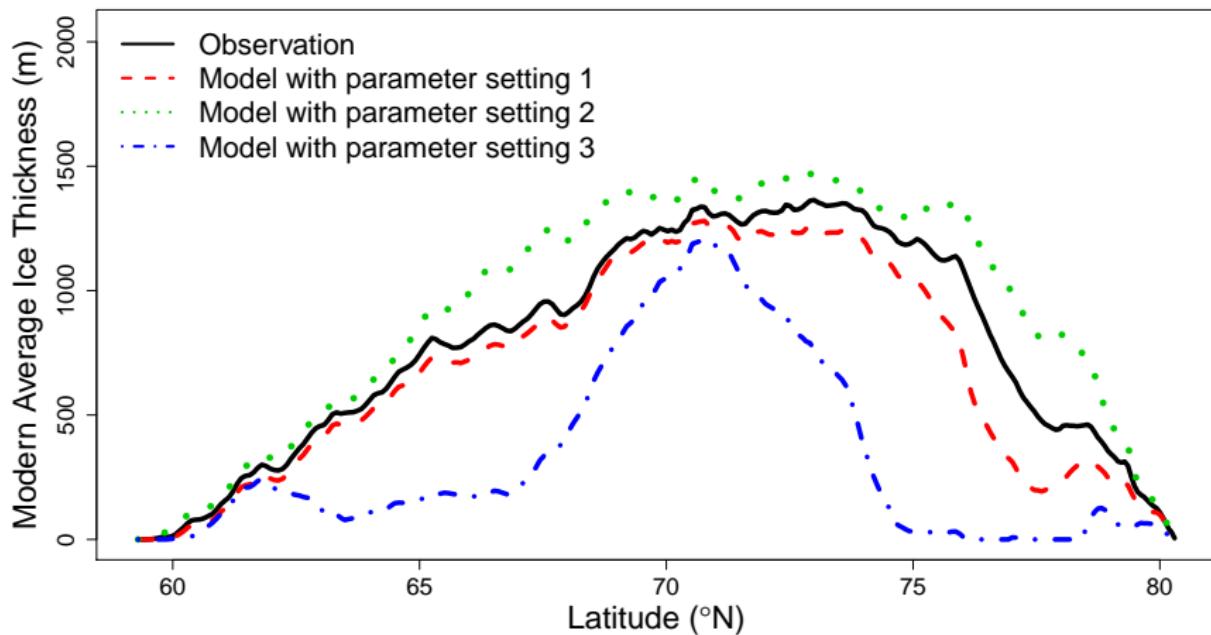


Paleo Data



Aggregated Ice Sheet Data: Example

- To avoid binary spatial data: aggregate across longitude.



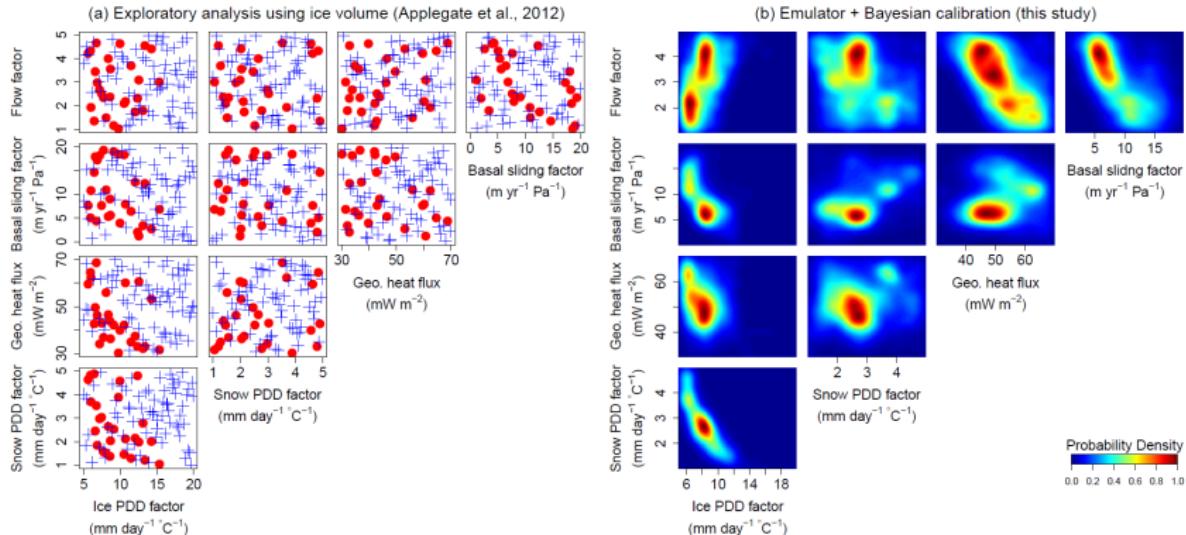
What is the Effect of Data Aggregation?

- ▶ Common practice: Calibration using aggregated data (e.g. zonal aggregate)
 - ▶ Avoids inferential and computational issues
 - ▶ Allow for limited skill of climate model in reproducing spatial patterns
- ▶ Using unaggregated data may result in
 - ▶ perhaps less uncertainty due to using more information?
 - ▶ perhaps more uncertainty due to poor model skill?
- ▶ These questions are largely unanswered due to the inability of existing methods to handle unaggregated data.

How Does Statistical Rigor Help Scientists?

1. We account for (epistemic) uncertainties in emulation
2. We provide *real* probability distributions, very important for impacts/risk quantification.
3. We use all available information (no aggregation): often reduces uncertainties.
4. We provide sharper/more useful results.

Example of Sharper Results



Left: previous ad-hoc methods. Right: statistical calibration
Chang, Haran, Olson, Keller (2014), *Annals of Applied Stats*

Two-stage Approach to Emulation-Calibration

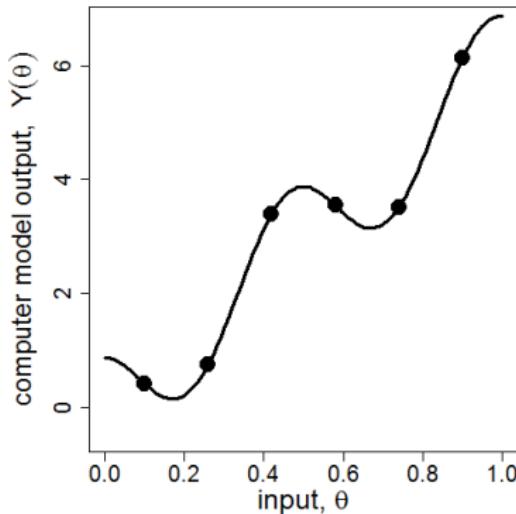
1. Emulation step: Find fast approximation for computer model using a Gaussian process (GP).
2. Calibration step: Infer climate parameter using emulator and observations, while accounting for data-model discrepancy

Modularization

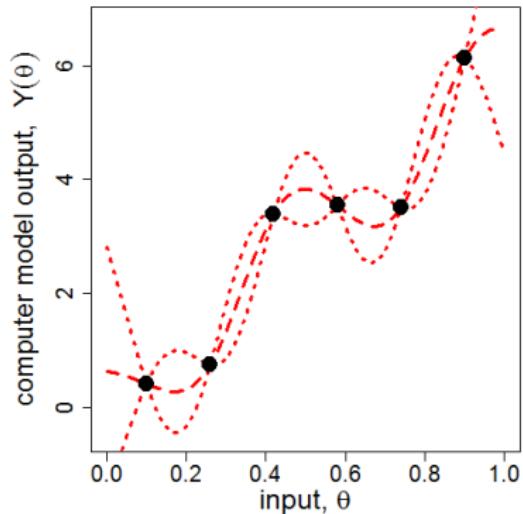
- ▶ Liu, Bayarri and Berger (2009)
- ▶ Bhat, Haran, Olson, Keller (2012)
- ▶ Chang, Haran, Applegate, Pollard (2016a; 2016b)

Emulation Step

Toy example: model output is a scalar, and continuous.



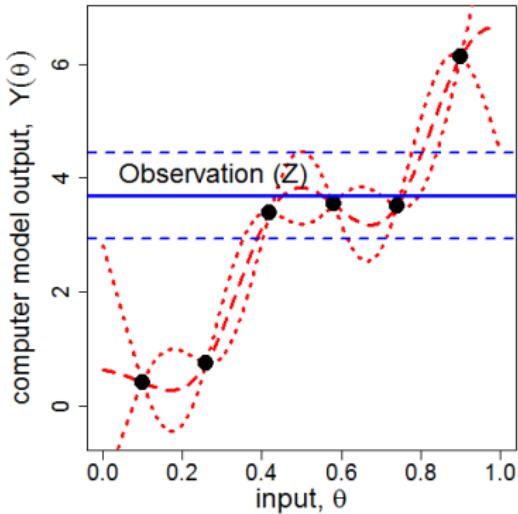
Computer model output (y-axis)
vs. input (x-axis)



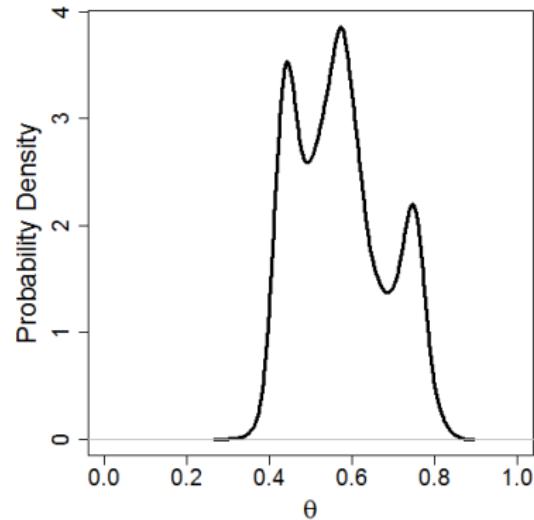
Emulation (approximation)
of computer model using GP

Calibration Step

Toy example: model output, observations are scalars



Combining observation
and emulator



Posterior PDF of θ
given model output and observation

Summary of Statistical Problem

► **Goal:** Learn about θ based on two sources of information:

► **Observations:**

1. Observed time series of past grounding line positions reconstructed from paleo records: $\mathbf{Z}_1 = (Z_1(t_1), \dots, Z_1(t_n))^T$, t_1, \dots, t_n are time points locations.
2. Observed modern ice-no ice from satellite data:
 $\mathbf{Z}_2 = [Z_2(\mathbf{s}_1), \dots, Z_2(\mathbf{s}_m)]$, locations $\mathbf{s}_1, \dots, \mathbf{s}_m$.

► **Model output**

1. $\mathbf{Y}_1(\theta_1), \dots, \mathbf{Y}_1(\theta_p)$, where each
 $\mathbf{Y}_1(\theta_i) = (Y_1(\theta_i, t_1), \dots, Y_1(\theta_i, t_n))^T$ is a time series of grounding line positions at parameter setting θ_i .
2. $\mathbf{Y}_2(\theta_1), \dots, \mathbf{Y}_2(\theta_p)$, where each
 $\mathbf{Y}_2(\theta_i) = (Y_2(\theta_i, \mathbf{s}_1), \dots, Y_2(\theta_i, \mathbf{s}_m))^T$ is a vector of spatial data at parameter setting θ_i .

Step 1: Computer Model Emulation Basics

- ▶ Fit Gaussian process model for computer model output \mathbf{Y} to interpolate the values at the parameter settings $\theta_1, \dots, \theta_p$ and the spatial locations $\mathbf{s}_1, \dots, \mathbf{s}_n$

$$\text{vec}(\mathbf{Y}) \sim N(\mathbf{X}\boldsymbol{\beta}, \boldsymbol{\Sigma}(\xi_y)),$$

$\text{vec}(\cdot)$ concatenates columns into one vector

- ▶ $\boldsymbol{\beta}$ and ξ_y estimated by maximum likelihood, $\hat{\boldsymbol{\beta}}, \hat{\xi}_y$.
- ▶ Covariance interpolates across spatial surface and input space.

Result: Obtain a probability model (from predictive distribution) for model output at any input parameter θ , $\eta(\theta, \mathbf{Y})$.

Step 2: Calibration Basics

- ▶ Discrepancy \approx mismatch between computer model output and data when parameters are perfectly calibrated and there is no observational error.
- ▶ Probability model for observations \mathbf{Z} is then

$$\mathbf{Z} = \eta(\boldsymbol{\theta}, \mathbf{Y}) + \boldsymbol{\delta},$$

where n -dimensional spatial field $\boldsymbol{\delta}$ is model-observation discrepancy with covariance parameter ξ_δ .

- ▶ Inference for $\boldsymbol{\theta}$ based on posterior distribution

$$\pi(\boldsymbol{\theta}, \xi_\delta | \mathbf{Z}, \mathbf{Y}, \hat{\xi}_y) \propto \underbrace{\hat{\mathcal{L}}(\mathbf{Z} | \mathbf{Y}, \boldsymbol{\theta}, \xi_\delta, \hat{\xi}_y)}_{\text{likelihood given by above}} \times \underbrace{p(\boldsymbol{\theta}) \times p(\xi_\delta)}_{\text{priors for } \boldsymbol{\theta} \text{ and } \xi_\delta}$$

with emulator parameter $\hat{\xi}_y$ fixed at MLE.

Challenge: Two sets of non-Gaussian, dependent data.

Emulation-Calibration with Binary Spatial Output

- ▶ Now $Y(\theta, \mathbf{s})$ is binary (0-1) model output, $Z(\mathbf{s})$ is data.
- ▶ Let $\boldsymbol{\Gamma}_{p \times n}$ be matrix of natural parameters for model output:
$$\gamma_{ij}^Y = \log \left(\frac{p_{ij}}{1-p_{ij}} \right)$$
 is logit for i th parameter setting at j th spatial location and $p_{ij} = P(Y(\theta_i, \mathbf{s}_j) = 1)$.
- ▶ Given $\boldsymbol{\Gamma}$, $Y(\theta_i, \mathbf{s}_j)$'s are conditionally independent Bernoulli.
- ▶ Approach (sketch):
 1. Assume it is possible to estimate $\boldsymbol{\Gamma}$ from the $n \times p$ matrix of computer model output.
 2. Emulate computer model by *interpolating natural parameters* using a Gaussian process across input parameter space and spatial locations.
 3. Calibration by using fitted Gaussian process $\eta(\theta, \mathbf{Y})$ + discrepancy δ to obtain a likelihood function for the *natural parameter vector for observations*.

Challenges

- ▶ Step 1 (obtaining Γ) is ill-posed: np parameters for np data points.
- ▶ Step 2 (emulation) is computationally infeasible: Cholesky factorization has computational cost of
$$\frac{1}{3} \times p^3 \times n^3 = \frac{1}{3} \times 499^3 \times 3,182^3 = 1.33 \times 10^{18} \text{ flops}$$
- ▶ Step 3 (calibration): involves having to perform a high-dimensional integration + expensive matrix operations.

We propose dimension-reduction to address both ill-posedness and computational issues.

Principal Components for Emulation: Basic Idea

- ▶ Consider model outputs at $\theta_1, \dots, \theta_p$ as if they were replicates of a multivariate process, thereby obtaining their PCs

$$\begin{pmatrix} Y(\mathbf{s}_1, \theta_1) & \dots & Y(\mathbf{s}_n, \theta_1) \\ \vdots & \ddots & \vdots \\ Y(\mathbf{s}_1, \theta_p) & \dots & Y(\mathbf{s}_n, \theta_p) \end{pmatrix}_{p \times n} \Rightarrow \begin{pmatrix} Y_1^R(\theta_1) & \dots & Y_{J_y}^R(\theta_1) \\ \vdots & \ddots & \vdots \\ Y_1^R(\theta_p) & \dots & Y_{J_y}^R(\theta_p) \end{pmatrix}_{p \times J_y}$$

- ▶ PCs pick up characteristics of model output that vary most across input parameters $\theta_1, \dots, \theta_p$.

Surprisingly flexible approach, allowing for non-separable covariance.

This works very well for Gaussian spatial data. (Chang, Haran, Olson, Keller, 2014).

Efficient Emulation: Outline

- ▶ Rewrite Γ in terms of logistic principal components (Lee et al., 2010).
- ▶ Use maximum likelihood to perform logistic principal components. Non-trivial, requires majorization-minimization (MM) algorithm (Lange et al. 2000; Hunter and Lange, 2004).
- ▶ Emulate each column of (reduced-dimensional) principal component matrix with an independent Gaussian process. Very fast and easy to do.
- ▶ We can obtain an emulator for Γ by emulating these principal components.

Dimension-reduction

- ▶ Consider Γ the $p \times n$ matrix of natural parameters for model output. Using logistic principal components (Lee et al., 2010), rewrite as:

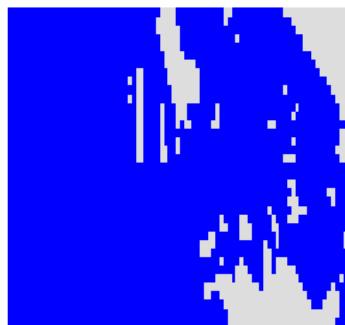
$$\Gamma = \mathbf{1}_p \otimes \boldsymbol{\mu}^T + \mathbf{W} \mathbf{K}_y^T, \quad (1)$$

where \mathbf{K}_y is an $n \times J_y$ orthogonal basis matrix, \mathbf{W} is the $p \times J_y$ principal component matrix with (i, j) th element $w_j(\theta_i)$, and $\boldsymbol{\mu}$ is the $n \times 1$ mean vector.

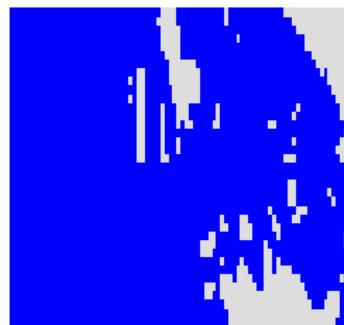
- ▶ Non-trivial and computationally challenging optimization to obtain matrices \mathbf{W} , \mathbf{K}_y by maximizing log-likelihood. Use majorization-minimization (MM) algorithm (Lange et al. 2000; Hunter and Lange, 2004).
- ▶ Emulate each column of \mathbf{W} using a separate Gaussian process.
- ▶ (Analogous to Gaussian emulation) By emulating these principal components we can emulate the original process.

Emulation Examples

Model Output from Run No.67



Emulated Output for Run No.67



Model Output from Run No.491



Emulated Output for Run No.491

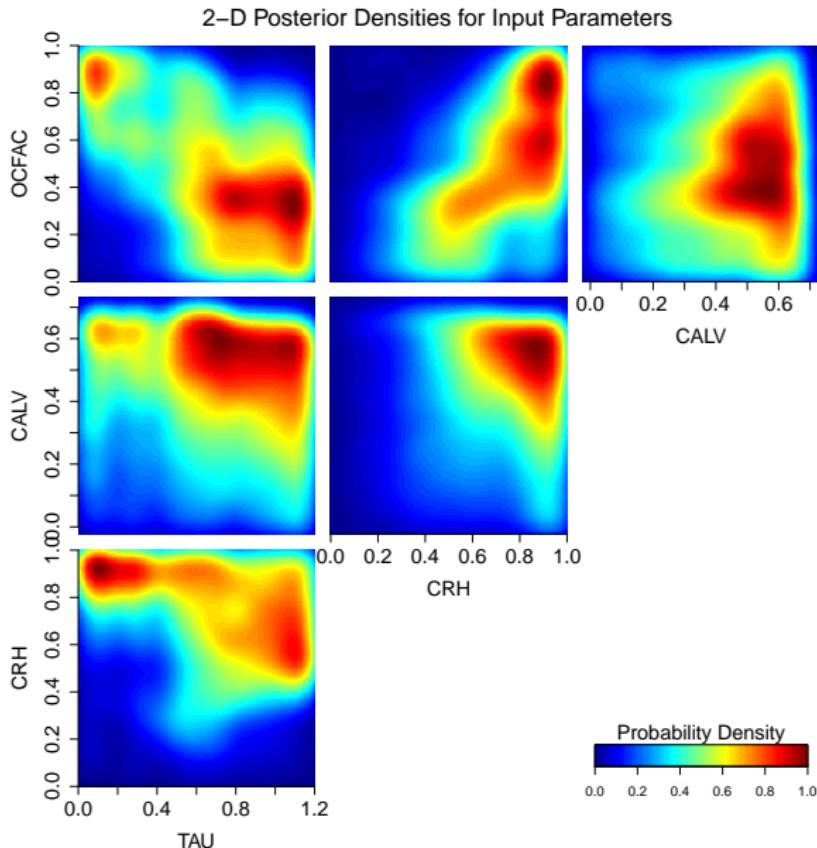


Efficient Calibration

Main ideas:

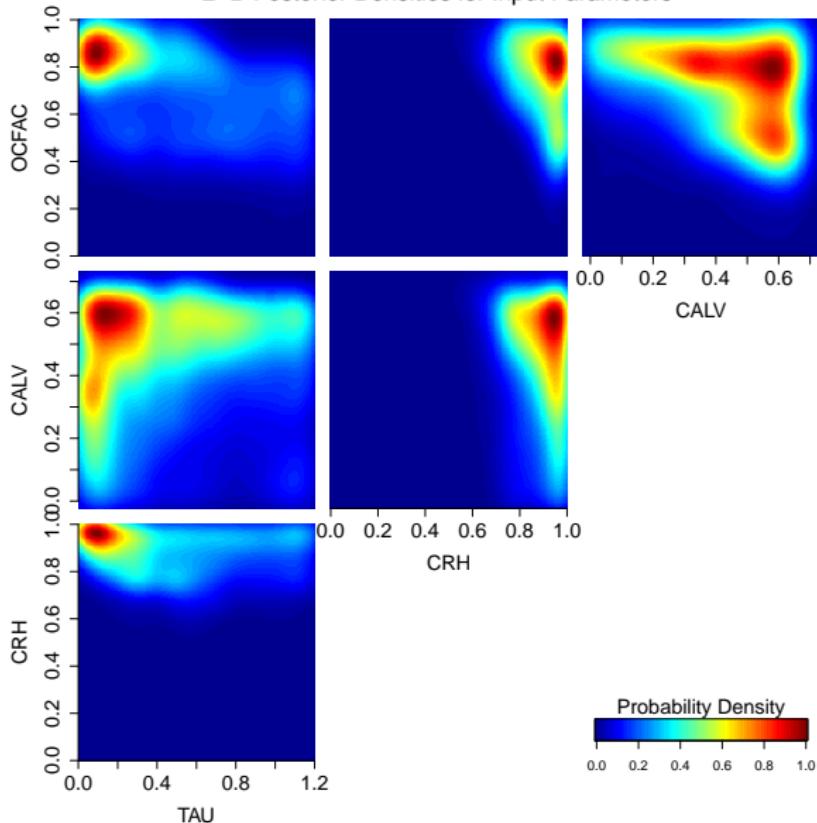
- ▶ Utilize the data to derive information about the model-data discrepancy.
- ▶ Use a basis representation for model-data discrepancy term.
- ▶ Markov chain Monte Carlo

Calibration Results with Modern Data



Calibration Results with Modern and Paleo Data

2-D Posterior Densities for Input Parameters

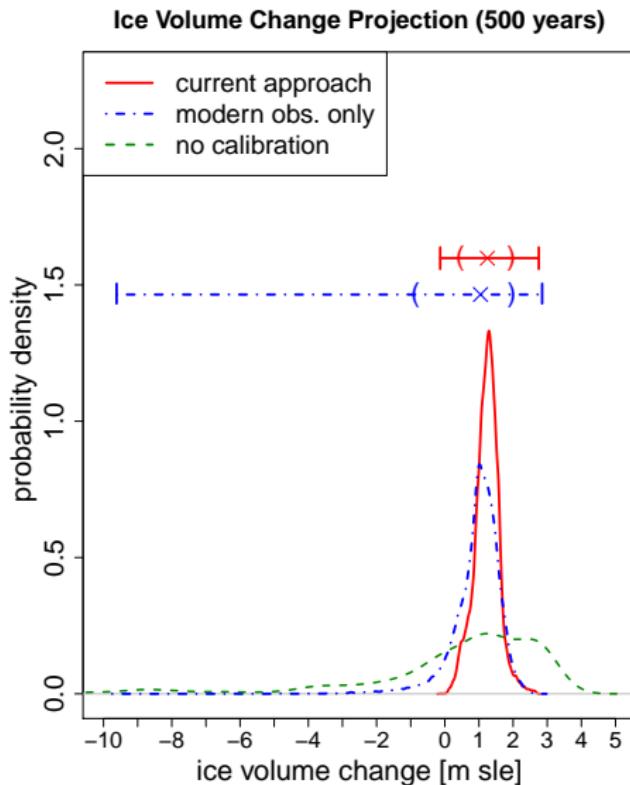


Paleo data eliminates unrealistic trajectories of the ice sheet.

Why are the Results Different?

- ▶ Unrealistic simulations with overshoots in past ice retreat and projected future regrowth are eliminated
- ▶ This “constrains” the probability distribution of the parameters

Ice Volume Projections: Paleo Cuts Off Left Tail



Possibility of “no sea level rise” is virtually eliminated.

Summary Remarks

- ▶ Principle components-based approach to computer model calibration with two sources of data.
 - ▶ Driven by computer model output at various settings ⇒ we choose a basis that is well suited to emulation and calibration.
 - ▶ Sidesteps issues with choice of basis, number of knots.
- ▶ Method tested extensively using simulated data examples, multiple data sets.
- ▶ Using paleo data results in an important change in sea level rise projections.
- ▶ Ongoing work: detailed understanding the current state of ice sheet thickness using simpler physical models.

Conclusion

I have a good brain and I have said a lot of things.

– Donald J. Trump

Acknowledgments

Collaborators:

- ▶ [Won Chang](#), University of Cincinnati
- ▶ [Yawen Guan](#), Penn State Statistics
- ▶ David Pollard, Earth and Environmental Systems Institute (EESI), Penn State U.
- ▶ Patrick Applegate, EESI, Penn State U.
- ▶ Klaus Keller, Geosciences, Penn State U.
- ▶ Roman Olson, The University of New South Wales

This work was partially supported by the following grants:

- ▶ The Network for Sustainable Climate Risk Management (SCRiM), **NSF GEO-1240507**.
- ▶ **NSF CDSE/DMS-1418090** Statistical Methods for Ice Sheet Projections

Relevant Manuscripts (online/arxiv.org)

- ▶ Haran, M, Chang, Keller, K., Nicholas, R., Pollard, D. (2016): *Statistics and the Future of Ice Sheets*, submitted.
- ▶ Chang, W., Haran, M, Applegate, P., Pollard, D. (2016): Improving Ice Sheet Model Calibration Using Paleoclimate and Modern Data, *Annals of Applied Statistics*.
- ▶ Chang, W., Haran, M, Applegate, P., Pollard, D. (2016): Calibrating an Ice sheet model using high-dimensional non-Gaussian spatial data, *J of Amer Stat Association*
- ▶ Chang, W., Applegate, P., Haran, M. and Keller, K. (2016) Large Ensemble modeling of last deglacial retreat of the West Antarctic Ice Sheet: Comparison of simple and advanced statistical techniques, *Geoscientific Model Development*.
- ▶ Chang, W., M. Haran, R. Olson, and K. Keller (2014): Fast dimension-reduced climate model calibration, *Annals of Applied Statistics*

Appendix

BEGIN APPENDIX

Efficient Calibration Outline

- ▶ Model for n -dimensional vector of natural parameters λ for observational data using emulator $\mathbf{K}_y \boldsymbol{\eta}(\theta, \mathbf{W})$:

$$\lambda = \mu + \mathbf{K}_y \boldsymbol{\eta}(\theta^*, \mathbf{W}) + \delta,$$

δ is discrepancy term that represents structural error between the model output and observational data.

- ▶ To get around the challenges in integrating out δ , use a basis representation.
- ▶ Discrepancy basis obtained from common discrepancy pattern across parameter settings.

Efficient Calibration: Details

- ▶ Calibration model for n -dimensional vector of natural parameters λ for observational data using the emulator $\mathbf{K}_y \boldsymbol{\eta}(\theta, \mathbf{W})$:

$$\lambda = \mu + \mathbf{K}_y \boldsymbol{\eta}(\theta^*, \mathbf{W}) + \delta,$$

where θ^* is the “best fit” value for the observational data, δ is discrepancy term that represents structural error between the model output and observational data.

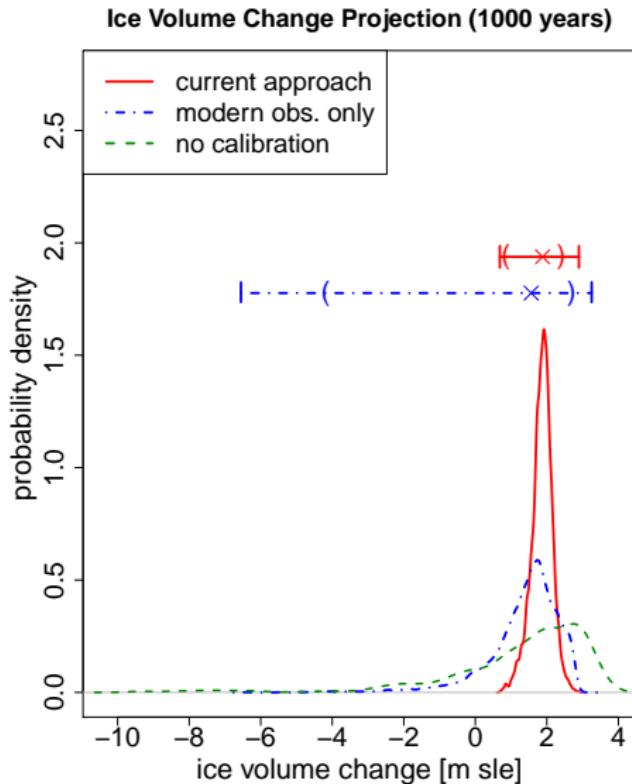
- ▶ To get around the challenges in integrating out δ described above we use a basis representation for the discrepancy term such that

$$\delta = \mathbf{K}_d \mathbf{v},$$

with the $n \times J_d$ basis matrix \mathbf{K}_d and the J_d -dimensional random coefficient vector $\mathbf{v} \sim N(\mathbf{0}, \sigma_d^2 \mathbf{I}_{J_d})$.

- ▶ Discrepancy basis obtained from common discrepancy pattern across parameter settings, proportion mismatch at each location.

Ice Volume Change Projections: Modern + Paleo



Appendix Step 1: Emulation (Approximating Computer Model)

- ▶ Find MLE for covariance parameter ξ , denoted by $\hat{\xi}$
- ▶ Get $\eta(\theta_{NEW}, \mathbf{Y})$ for prediction at any $\theta_{NEW} \in \Theta$:
 - ▶ GP gives

$$\begin{pmatrix} \mathbf{Y} \\ \mathbf{Y}(\theta_{NEW}) \end{pmatrix} \sim N \left(\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}_{n(p+1) \times 1}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}_{n(p+1) \times n(p+1)} \right)$$

- ▶ Emulator:

$$\eta(\theta_{NEW}, \mathbf{Y}) = \mathbf{Y}(\theta_{NEW}) | \mathbf{Y} \sim N \left(\Sigma_{21} \Sigma_{11}^{-1} \mathbf{Y}, \Sigma_{22} - \Sigma_{21} \Sigma_{11}^{-1} \Sigma_{12} \right)$$

Appendix: Computational Challenges for Gaussian Emulation-Calibration

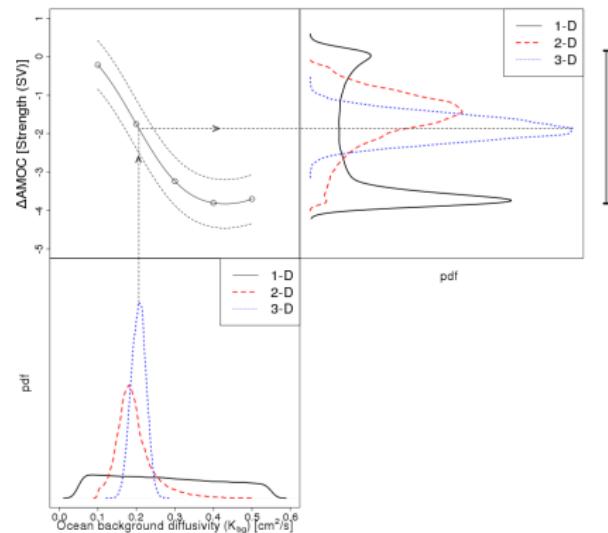
- ▶ Emulation requires dealing with $np \times np$ covariance matrix of \mathbf{Y} . E.g. if $n = 61,051$, $p = 250$
 - ▶ Cholesky decomposition costs $\frac{1}{3}n^3p^3 = 1.185 \times 10^{21}$ flops.
 - ▶ Covariance matrix is of size $8 \times \frac{250^2 \times 61051^2}{1024^3} = 1,735,624$
- ▶ Calibration faces similar challenges for dealing with $n \times n$ covariance matrix.

Our fast reduced-dimension approach: Fast computation using PC and Kernel Convolution

Appendix: Results

Computational efficiency allows us to calibrate using unaggregated data.

- ▶ We compare 1D (depth profile) and 2D (zonal average) with 3D (unaggregated) data.
- ▶ Inference with 3D data leads to sharper inference for θ .
- ▶ Inference using 3D data is more robust to changes in prior specifications for discrepancy parameters.



Appendix: Cross-Validation for Emulator

- ▶ Example of leave-10%-out cross validation result:

