Model Selection Wrap-Up and Emulation

BEE 6940 Lecture 13

April 24, 2023

Table of Contents

Review of Model Selection

What Is The Goal of Model Selection?

Key Idea: Model selection consists of navigating the bias-variance tradeoff.

Model error (*e.g.* RMSE) is a combination of *irreducible error*, *bias*, and *variance*.

- Bias can come from under-dispersion (too little complexity) or neglected processes;
- Variance can come from over-dispersion (too much complexity) or poor identifiability.

Log-Likelihood as Predictive Fit Measure

The measure of predictive fit that we will use is the **log predictive density** or **log-likelihood** of a replicated data point/set, $p(y^{rep}|\theta)$.

For normally-distributed data, this is proportional to the mean-squared error.

Log-Likelihood as Predictive Fit Measure

Why use the log-likelihood density instead of the log-posterior?

- The likelihood captures the data-generating process;
- The posterior includes the prior, which is only relevant for parameter estimation.

Important: This means that the prior is still relevant in predictive model assessment, and should be thought of as part of the model structure!

Cross-Validation

Cross-validation is the gold standard for predictive accuracy: how well does the fitted model predict out of sample data?

The problems:

- Leave-one-out CV can be very computationally expensive!
- We often don't have a lot of data for calibration, so holding some back can be a problem.
- How to divide data with spatial or temporal structure? This can be addressed by partitioning the data more cleverly (*e.g.* leaving out future observations), but makes the data problem worse.

Expected Out-Of-Sample Predictive Accuracy

Instead, we will try to compute the *expected out-of-sample predictive accuracy*.

The out-of-sample predictive fit of a new data point $\tilde{y}_i$ is

$$\begin{aligned}\log p_{\text{post}}(\tilde{y}_i) &= \log \mathbb{E}_{\text{post}}\left[p(\tilde{y}_i|\theta)\right] \\ &= \log \int p(\tilde{y}_i|\theta) p_{\text{post}}(\theta)\, d\theta.\end{aligned}$$

Kullback-Leibler Divergence

K-L Divergence can be interpreted as the "surprise" when using Q (model approximation) as an approximation to P ("true" data-generating process).

We use "information criteria" as an approximation based on the existing data:

- compute $p(y|\theta)$ as expected log-predictive density for existing data;
- correct for bias from using calibration data twice

Information Criteria

Akaike Information Criterion

The AIC is defined as $-2\widehat{\mathrm{elpd}}_{\mathrm{AIC}}$ (for "historical" reasons; this is called the deviance scale).

Due to this convention, lower AICs are better (they correspond to a higher predictive skill).

Akaike Information Criterion

In the case of a normal model with independent and identically-distributed data and uniform priors, k is the "correct" bias term so that $\widehat{\text{elpd}}_{\text{AIC}}$ converges to the K-L divergence (there are corrections when the sample size is sufficiently small).

However, with more informative priors and/or hierarchical models, the bias correction k is no longer appropriate, as there is less "freedom" associated with each parameter.

AIC and Model Averaging

This gives rise to the idea of *Akaike weights*:

$$w_i = \frac{\exp(-\Delta_i/2)}{\sum_{m=1}^{M} \exp(-\Delta_m/2)}.$$

Model projections can then be weighted based on w_i, which can be interpreted as the probability that M_i is the K-L minimizing model in $\mathcal{M}$.

Deviance Information Criterion

The Deviance Information Criterion (DIC) is a more Bayesian generalization of AIC which uses the posterior mean

$$\hat{\theta}_{\text{Bayes}} = \mathbb{E}\left[\theta|y\right]$$

and a bias correction derived from the data.

Deviance Information Criterion

$$\widehat{\mathrm{elpd}}_{\mathrm{DIC}} = \log p(y|\hat{\theta}_{\mathrm{Bayes}}) - p_{\mathrm{DIC}},$$

where

$$p_{\mathrm{DIC}} = 2\left(\log p(y|\hat{\theta}_{\mathrm{Bayes}}) - \mathbb{E}_{\mathrm{post}}\left[\log p(y|\theta)\right]\right).$$

Then, as with AIC,

$$\mathrm{DIC} = -2\widehat{\mathrm{elpd}}_{\mathrm{DIC}}.$$

DIC: Effective Number of Parameters

What is the meaning of p_{DIC}?

- The difference between the average log-likelihood (across parameters) and the log-likelihood at a parameter average measures "degrees of freedom".
- Note that p_{DIC} can be negative if the posterior mean is far from the posterior mode: this suggests that the model is poorly constrained and may do better on new data than the existing data.
- The DIC adjustment assumes independence of residuals for fixed θ.

AIC vs. DIC

AIC and DIC often give similar results, but don't have to. For example, for the SLR example, with relatively uninformative priors,

$$\Delta_{\text{DIC}} \approx 4,$$

which based on the AIC scale suggests limited (but stronger) evidence for the quadratic model.

The key difference is the impact of priors on parameter estimation and model degrees of freedom.

Convergence of AIC and DIC

Both AIC and DIC converge to the K-L divergence.

Also valuable: they both also converge to expected leave-one-out cross-validation.

Watanabe-Akaike Information Criterion (WAIC)

WAIC is a *fully* Bayesian generalization of AIC.

$$\widehat{\mathrm{elpd}}_{\mathrm{WAIC}} = \sum_{i=1}^{n} \log \int p(y_i|\theta) p_{\mathrm{post}}(\theta)\, d\theta - p_{\mathrm{WAIC}},$$

where

$$p_{\mathrm{WAIC}} = \sum_{i=1}^{n} \mathrm{Var}_{\mathrm{post}} \left(\log p(y_i|\theta)\right).$$

Watanabe-Akaike Information Criterion (WAIC)

p_{WAIC} is an estimate of the number of "unconstrained" parameters in the model.

- A parameter counts as 1 if its estimate is "independent" of the prior;
- A parameter counts as 0 if it is fully constrained by the prior.
- A parameter gives a partial value if both the data and prior are informative.

Watanabe-Akaike Information Criterion (WAIC)

- WAIC can be viewed as an approximation to leave-one-out CV, and averages over the entire posterior, vs. AIC and DIC which use point estimates.
- But it doesn't work well with highly structured data; no real alternative to more clever uses of Bayesian cross-validation.

Bayesian LOO-CV

By default, Bayesian LOO-CV is extremely expensive:

$$\text{loo-cv} = \sum_{i=1}^{n} \log p_{\text{post}(-i)}(y_i),$$

which requires refitting the model without y_i for every data point.

Can think of the "number of effective parameters" as the difference between the log-predictive density of the data set minus the LOO-CV estimate.

Bayesian LOO-CV: More Advanced Methods

There are approximations to Leave-One-Out Cross-Validation which use *importance sampling* to avoid this, and these can be extended to time series.

See

- Vehtari et al (2015) on "Pareto-smoothed "
- Bürkner et al (2020) on time-series "leave-future-out CV".

"Bayesian" "Information" Criterion

$$\text{BIC} = -2 \log p(y|\hat{\theta}_{\text{MLE}}) + k \log n.$$

BIC:

- Is not Bayesian (it relies on the MLE);
- Has no relationship to information theory (unlike AIC/DIC);
- Assumes $\mathcal{M}$-closed (e.g. that the true model is under consideration);

"Bayesian" "Information" Criterion

BIC approximates the *prior* log-predictive likelihood and leave-k-out cross-validation (hence the extra penalization for additional parameters).

Differences between BIC values can therefore be interpreted as Bayes factors, which are ratios of marginalized likelihoods (see Kass & Raftery (1995) for more on Bayes factors).

This is why it's odd when model selection consists of examining both AIC and BIC: these are different quantities with different purposes!

Key Takeaways: Model Selection

Key Takeaways

- Model selection is a balance between bias (underfitting) and variance (overfitting).
- For predictive assessment, leave-one-out cross-validation is an ideal, but hard to implement in practice (particularly for time series).
- AIC and DIC can be used to approximate K-L divergence and leave-one-out cross-validation.
- BIC is an entirely different measure, approximating the prior predictive distribution.

Key Consideration

Model selection can result in significant biases when separated from hypothesis-driven model development.

- There's no free lunch: better off thinking about the scientific or engineering problem you want to solve and use domain knowledge/checks rather than throwing a large number of possible models into the machinery.
- Regularizing priors reduce potential for overfitting.
- Model averaging (Hoeting et al (1999)) and stacking (Yao et al (2018)) can combine multiple models as an alternative to selection.

Model Simplicity: Tradeoffs

Parsimony As A Modeling Virtue

Parsimony can reduce the chance of overfitting and increased variance, all else being equal.

Model simplicity has another advantange: **simpler models are less computationally expensive**.

Parsimony As A Modeling Virtue

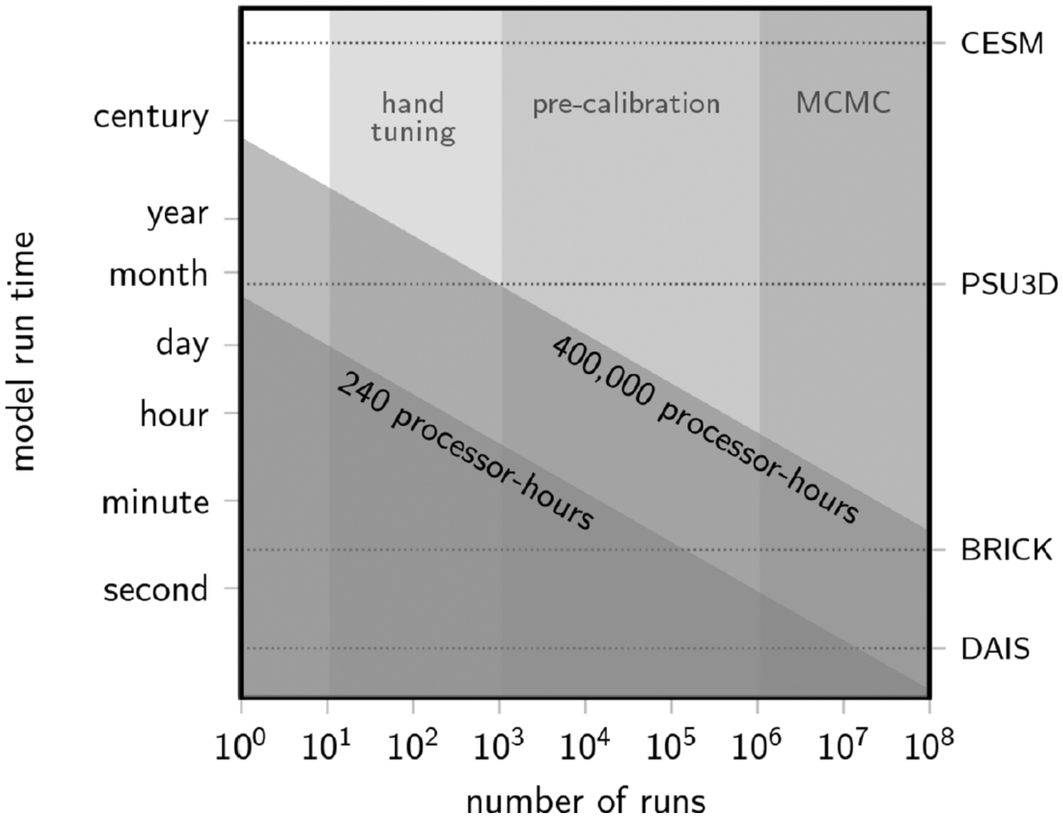
Parsimony can reduce the chance of overfitting and increased variance, all else being equal.

Model simplicity has another advantange: **simpler models are less computationally expensive**.

Why is this beneficial?

Benefits of Computational Simplicity

- More thorough representation of uncertainties
- Can focus on "important" characteristics for problem at hand
- Potential increase in generalizability



Source: Helgeson et al (2022)

Downsides to Computational Simplicity

- Potential loss of salience
- May miss important dynamics
- Parameter/dynamical compensation can result in loss of interpretability

Upshot of Simplicity Tradeoffs

Simple models can be epistemically and practically valuable.

But:

Need to carefully select which processes/parameters are included in the simplified representation, and at what resolution.

Emulation

Approximating Complex Models

Challenge: How do we simplify complex models to keep key dynamics but reduce computational expense?

Approximate (or **emulate**) the model response surface.

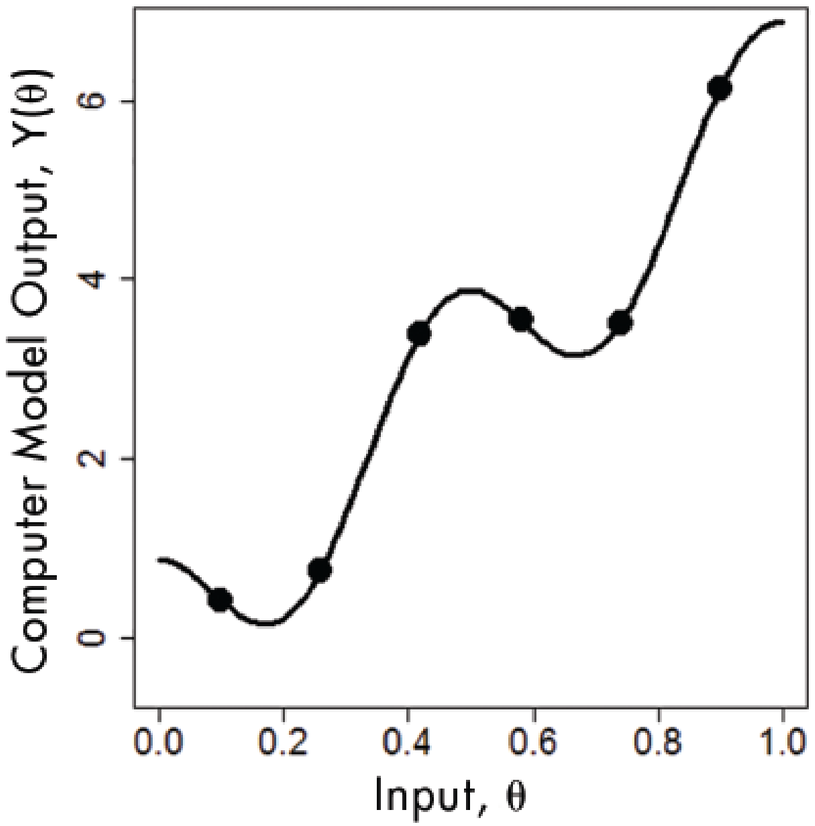
Approximating Complex Models

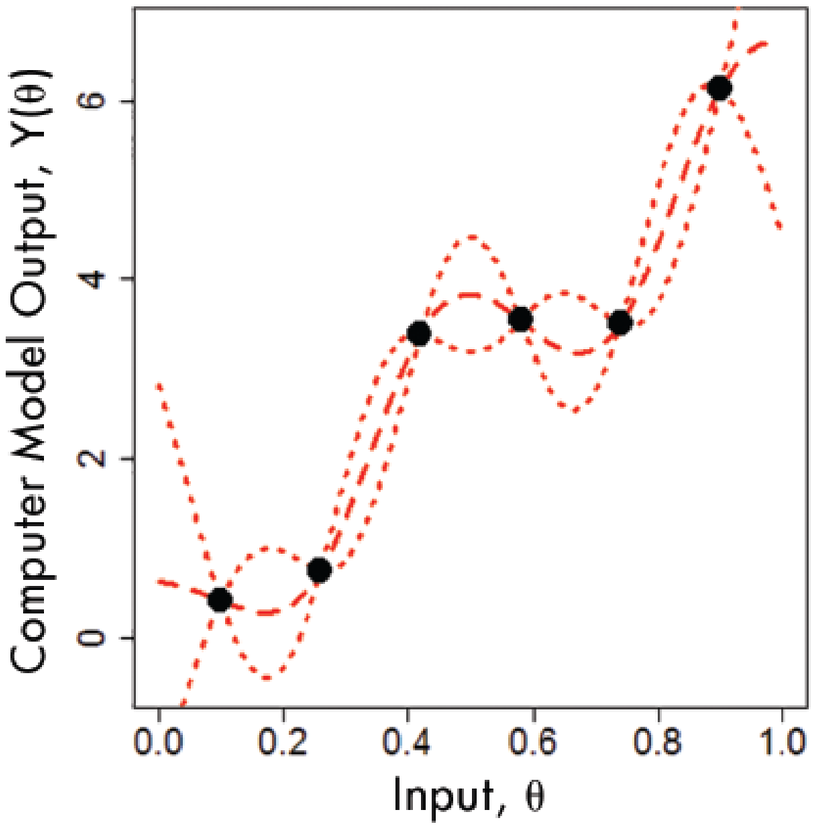
Challenge: How do we simplify complex models to keep key dynamics but reduce computational expense?

Approximate (or **emulate**) the model response surface.

1. Evaluate original model at an ensemble of points (design of experiment)
2. Calibrate emulator against those points.
3. Use emulator for UQ with MCMC or other methods.

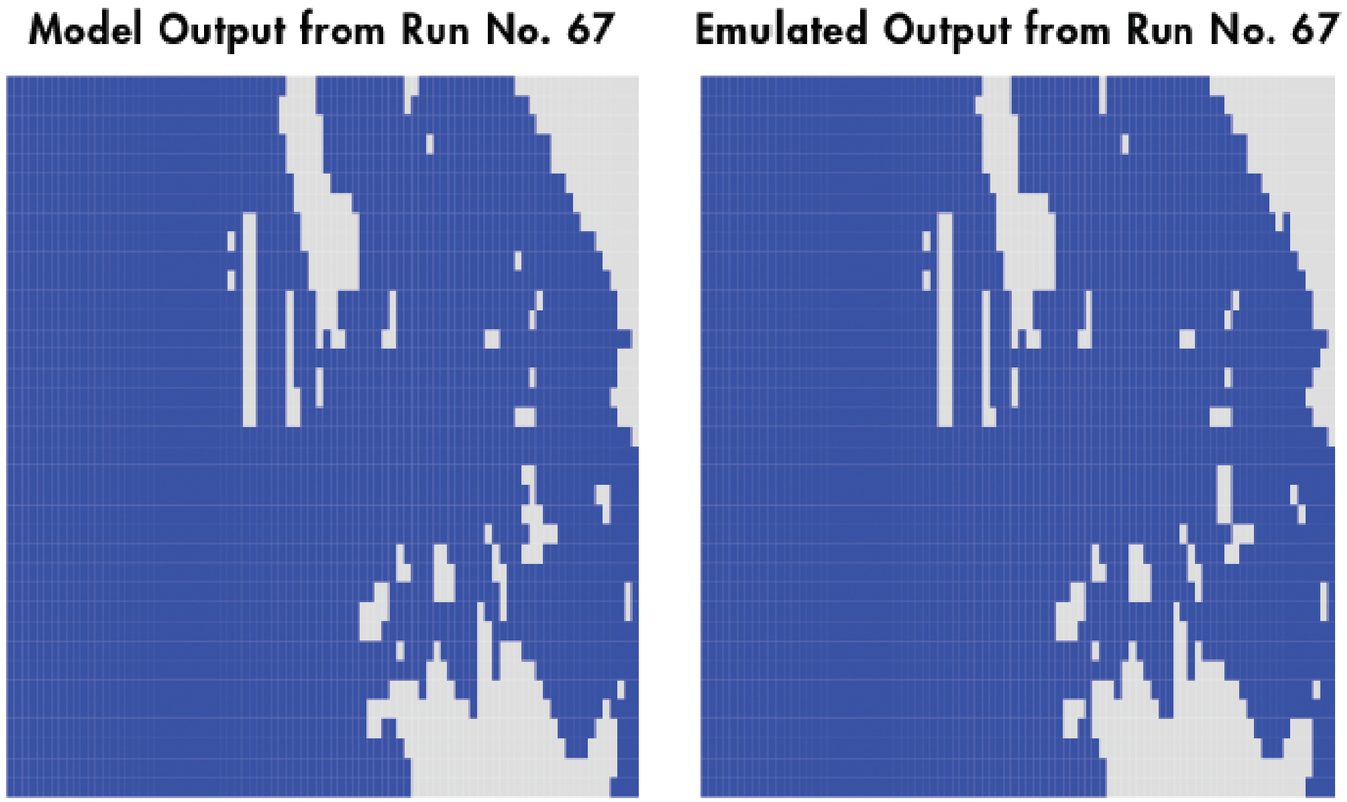
Emulation of a 1-D Toy Model





Source: Haran et al (2017)

Emulating Higher Dimensional Output



Source: Haran et al (2017)

Common Emulation Methods

- Gaussian Processes
- Polynomial Chaos Expansion
- Machine Learning

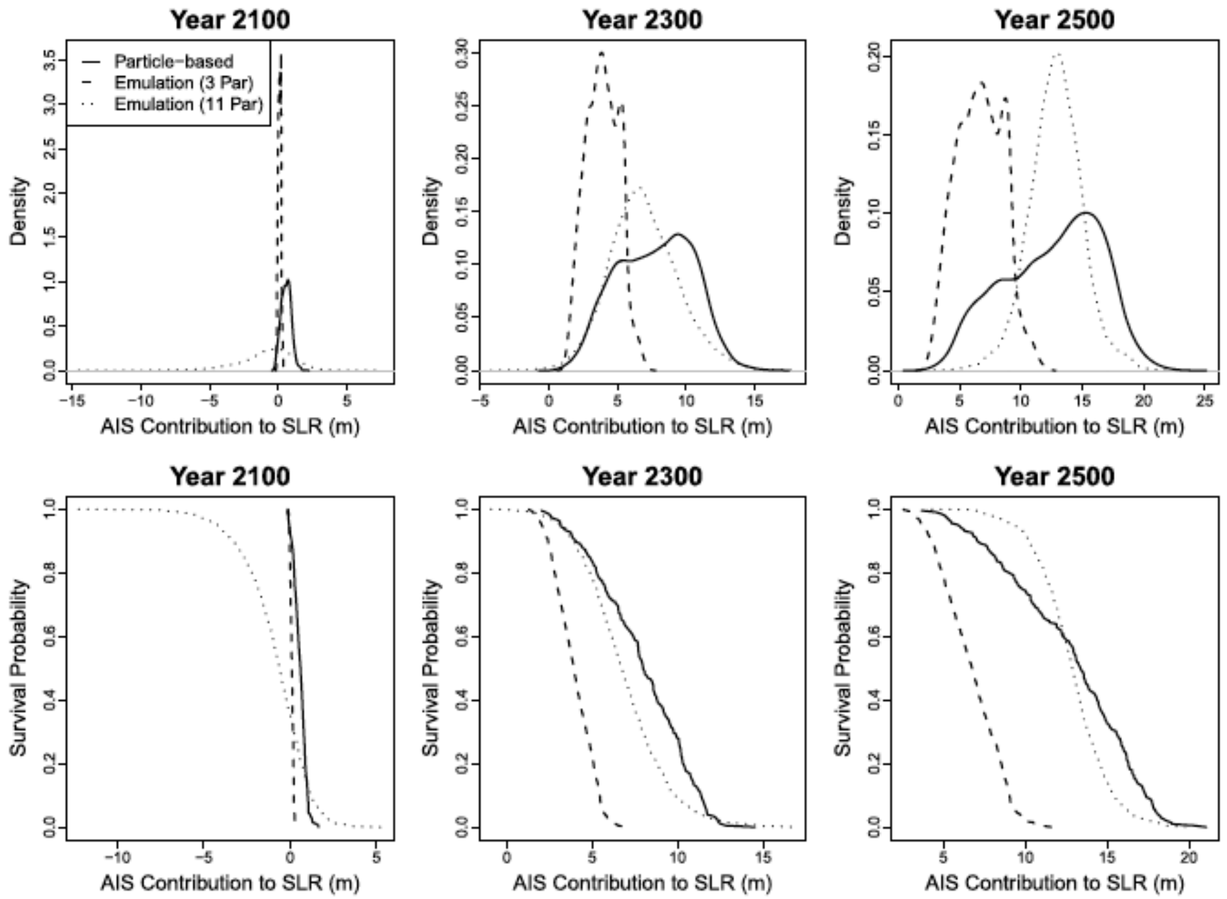
Experimental Design

Many of these methods (Gaussian processes in particular) only work for a few dimensions.

Need to:

1. Select key parameters with a sensitivity analysis.
2. Use an appropriate sampling of points (*e.g.* Latin Hypercube) for emulator training.
3. Select an appropriate metric for approximation.

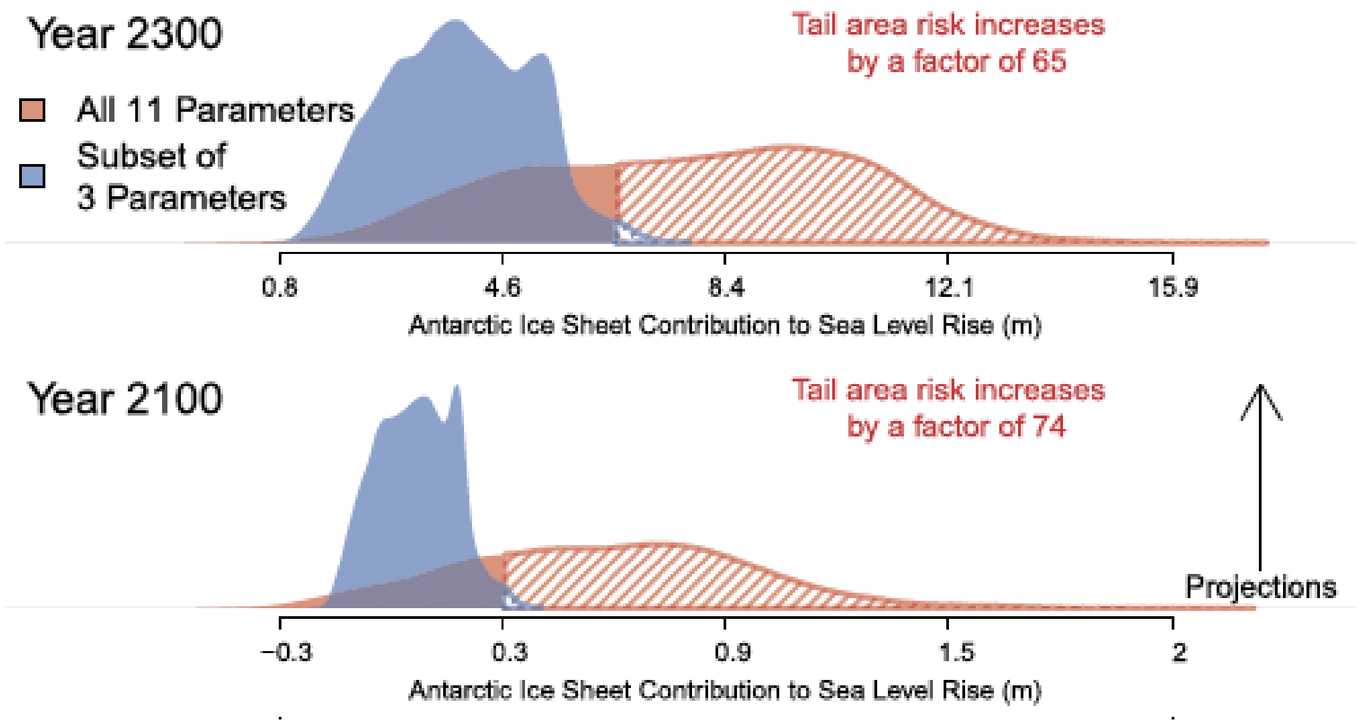
Is Emulation Always The Right Choice?



Source: Lee et al (2020)

Is Emulation Always The Right Choice?

This error can have large knock-on effects for risk analysis:



Source: Lee et al (2020)

Emulation: Key Takeaways

Emulation Takeaways

- Model simplicity can be valuable.
- Tradeoff between computational expense and fidelity of approximation.
- Emulation is a common approach.
- Emulator methods have different pros and cons which can make them more or less important.
- Emulator error can strongly influence resulting risk estimates.

Upcoming Schedule

Upcoming Schedule

Wednesday: Discussion of Helgeson et al (2022).

Monday: Modeling exposure, vulnerability, and response.