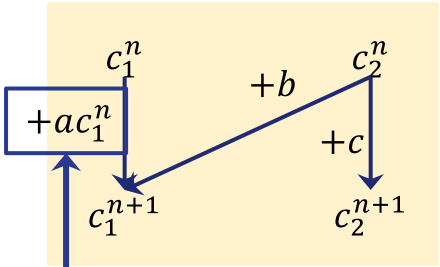
What is an adjoint model?
And why you should care about it?

University of Edinburgh

Fei Yao
Fei.Yao@ed.ac.uk

What is an Adjoint model?

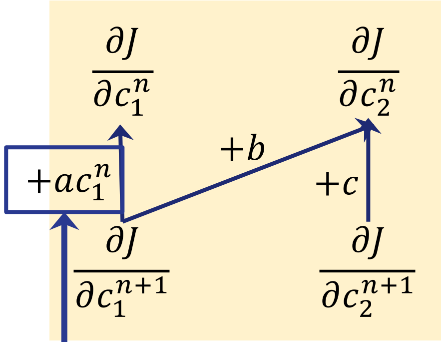
Forward



Assuming:

- $c_1^{n+1} = ac_1^n c_1^n + bc_2^n$
- $c_2^{n+1} = cc_2^n$
- J solely relies on $\boldsymbol{c^{n+1}}$

Backward



All the intermediate values computed in the forward run need to be stored for the adjoint run.

$$\boldsymbol{F_c^n} = \begin{bmatrix} \frac{\partial c_1^{n+1}}{\partial c_1^n} & \frac{\partial c_1^{n+1}}{\partial c_2^n} \\ \frac{\partial c_2^{n+1}}{\partial c_1^n} & \frac{\partial c_2^{n+1}}{\partial c_2^n} \end{bmatrix} = \begin{bmatrix} 2ac_1^n & b \\ 0 & c \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial J}{\partial c_1^n} \\ \frac{\partial J}{\partial c_2^n} \end{bmatrix} = \begin{bmatrix} \frac{\partial J}{\partial c_1^{n+1}} \frac{\partial c_1^{n+1}}{\partial c_1^n} + \frac{\partial J}{\partial c_2^{n+1}} \frac{\partial c_2^{n+1}}{\partial c_1^n} \\ \frac{\partial J}{\partial c_1^{n+1}} \frac{\partial c_1^{n+1}}{\partial c_2^n} + \frac{\partial J}{\partial c_2^{n+1}} \frac{\partial c_2^{n+1}}{\partial c_2^n} \end{bmatrix} = \begin{bmatrix} \frac{\partial c_1^{n+1}}{\partial c_1^n} & \frac{\partial c_2^{n+1}}{\partial c_1^n} \\ \frac{\partial c_1^{n+1}}{\partial c_2^n} & \frac{\partial c_2^{n+1}}{\partial c_2^n} \end{bmatrix} \begin{bmatrix} \frac{\partial J}{\partial c_1^{n+1}} \\ \frac{\partial J}{\partial c_2^{n+1}} \end{bmatrix}$$

$$= \begin{bmatrix} 2ac_1^n & 0 \\ b & c \end{bmatrix} \begin{bmatrix} \frac{\partial J}{\partial c_1^{n+1}} \\ \frac{\partial J}{\partial c_2^{n+1}} \end{bmatrix} = (\boldsymbol{F_c^n})^T \begin{bmatrix} \frac{\partial J}{\partial c_1^{n+1}} \\ \frac{\partial J}{\partial c_2^{n+1}} \end{bmatrix}$$

Adjoint/Transpose

Machine learning prefers to call it back-propagation using the *chain rule*.

What is an Adjoint model?

Let's expand dimensions:

$$J = \sum_{n=0}^{N} J^n(\boldsymbol{c^n}) + J_{\boldsymbol{p}}(\boldsymbol{p})$$

time

Space, species

A cost function may depend on a temporal subset (i.e., for some n, $J^n(c^n) = 0$) or whole of concentrations and may include a penalty term explicitly depending upon model parameters.

$$\boldsymbol{\nabla}_{c^n} J = \frac{\partial J}{\partial \boldsymbol{c}^n} = \frac{\partial J^n(\boldsymbol{c^n})}{\partial \boldsymbol{c}^n} + \frac{\partial J^{n+1}(\boldsymbol{c^{n+1}})}{\partial \boldsymbol{c}^n} + \cdots + \frac{\partial J^{N-1}(\boldsymbol{c^{N-1}})}{\partial \boldsymbol{c}^n} + \frac{\partial J^N(\boldsymbol{c^N})}{\partial \boldsymbol{c}^n} = \sum_{n'=n}^{N} \frac{\partial J^{n'}(\boldsymbol{c^{n'}})}{\partial \boldsymbol{c}^n}$$

A change in the current state vector will impact all subsequent state vectors and the associated cost functions.

$$\boldsymbol{\nabla}_{c^n} J = \frac{\partial J^n(\boldsymbol{c^n})}{\partial \boldsymbol{c^n}} + (\boldsymbol{F}_c^n)^T \frac{\partial J^{n+1}(\boldsymbol{c^{n+1}})}{\partial \boldsymbol{c^{n+1}}} + (\boldsymbol{F}_c^n)^T (\boldsymbol{F}_c^{n+1})^T \cdots (\boldsymbol{F}_c^{N-2})^T \frac{\partial J^{N-1}(\boldsymbol{c^{N-1}})}{\partial \boldsymbol{c^{N-1}}} + (\boldsymbol{F}_c^n)^T (\boldsymbol{F}_c^{n+1})^T \cdots (\boldsymbol{F}_c^{N-2})^T (\boldsymbol{F}_c^{N-1})^T \frac{\partial J^N(\boldsymbol{c^N})}{\partial \boldsymbol{c^N}}$$

$$= \frac{\partial J^n(\boldsymbol{c^n})}{\partial \boldsymbol{c^n}} + (\boldsymbol{F}_c^n)^T \left(\frac{\partial J^{n+1}(\boldsymbol{c^{n+1}})}{\partial \boldsymbol{c^{n+1}}} + \cdots + (\boldsymbol{F}_c^{n+1})^T \cdots (\boldsymbol{F}_c^{N-2})^T \frac{\partial J^{N-1}(\boldsymbol{c^{N-1}})}{\partial \boldsymbol{c^{N-1}}} + (\boldsymbol{F}_c^{n+1})^T \cdots (\boldsymbol{F}_c^{N-2})^T (\boldsymbol{F}_c^{N-1})^T \frac{\partial J^N(\boldsymbol{c^N})}{\partial \boldsymbol{c^N}} \right)$$

$$= \frac{\partial J^n(\boldsymbol{c^n})}{\partial \boldsymbol{c^n}} + (\boldsymbol{F}_c^n)^T \nabla_{c^{n+1}} J$$

Iteration

Adjoint forcings

$\boldsymbol{F}_c^n$ is defined as the Jacobian matrix between two consecutive state vectors, i.e., $\frac{\partial F^n(\boldsymbol{c^n})}{\partial \boldsymbol{c}^n}$.

Initialization

What is an Adjoint model?

From $\nabla_{c^n} J$ to $\nabla_{\boldsymbol{p}} J$:

- A change in a constant model parameter $\boldsymbol{p}$ will impact all state vectors (excluding initial conditions at $n = 0$) and the associated cost functions.
- $\boldsymbol{F}_{\boldsymbol{p}}^n$ can be similarly defined as the Jacobian matrix between state vectors and model parameters, i.e., $\frac{\partial F^n(\boldsymbol{c}^n)}{\partial \boldsymbol{p}}$.

$$\nabla_{p} J = (\boldsymbol{F}_{\boldsymbol{p}}^0)^T \nabla_{c^1} J + (\boldsymbol{F}_{\boldsymbol{p}}^1)^T \nabla_{c^2} J + \cdots + \left(\boldsymbol{F}_{\boldsymbol{p}}^{N-2}\right)^T \nabla_{c^{N-1}} J + \left(\boldsymbol{F}_{\boldsymbol{p}}^{N-1}\right)^T \nabla_{c^N} J + \boxed{\frac{\partial J_{\boldsymbol{p}}}{\partial \boldsymbol{p}}}$$

Initialization

$$\boxed{\nabla_{\boldsymbol{p}} J = \underline{(F_{\boldsymbol{p}}^{n-1})^T} \nabla_{c^n} J + \nabla_{\boldsymbol{p}} J}$$

Iteration

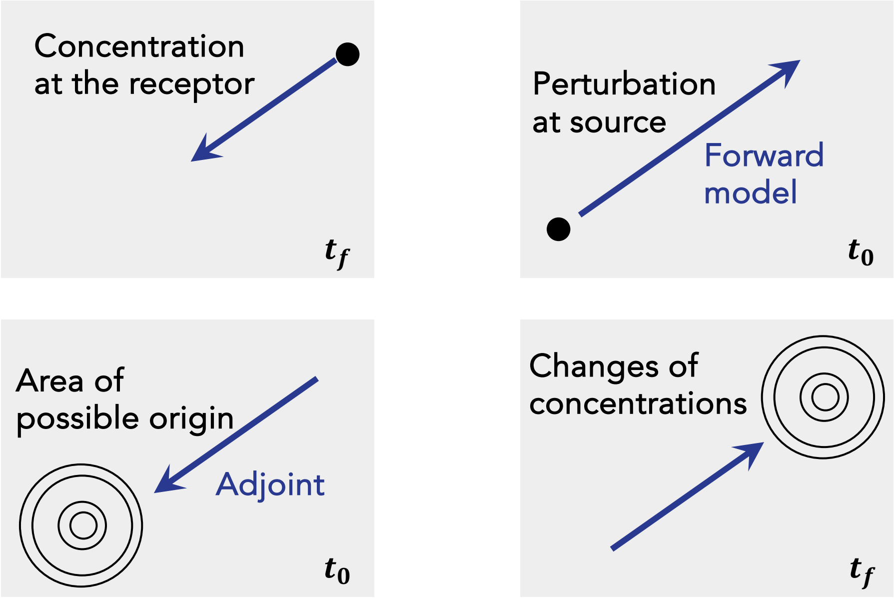
In a single step n, we simply need $\frac{\partial J^n(\boldsymbol{c}^n)}{\partial c^n}$, which are referred to as adjoint forcings as their role in the adjoint model is analogous to that of emissions in the forward model, as well as Jacobians $\boldsymbol{F}_c^{n-1}$ and $\boldsymbol{F}_{\boldsymbol{p}}^{n-1}$, which need NOT to be stored for more than this step.

Adjoint model for sensitivity studies

By re-writing $J = \sum_{c \in \Omega} \boldsymbol{c}$, where Ω is the domain of time, space, and species, we are interested in the sensitivity of a scalar (e.g., regional loads of multiple air pollutants over a specific period) with respect to many model parameters $\boldsymbol{p}$ (e.g., emissions).

Adjoint Model
(receptor-oriented)

Forward Model
(source-oriented)



Adjoint model for inversion studies

Linking the cost function to the Bayes' Theorem:

$$p(\boldsymbol{\sigma}) = \frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{S}_{\sigma}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)^T\boldsymbol{S}_{\sigma}^{-1}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)\}$$

$$p(\boldsymbol{c}_{obs}|\boldsymbol{\sigma}) = \frac{1}{(2\pi)^{\frac{q}{2}}|\boldsymbol{S}_{obs}|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(\boldsymbol{Hc}-\boldsymbol{c}_{obs})^T\boldsymbol{S}_{obs}^{-1}(\boldsymbol{Hc}-\boldsymbol{c}_{obs})\}$$

$$p(\boldsymbol{\sigma}|\boldsymbol{c}_{obs}) \propto \exp\{-\frac{1}{2}\{(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)^T\boldsymbol{S}_{\sigma}^{-1}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)+(\boldsymbol{Hc}-\boldsymbol{c}_{obs})^T\boldsymbol{S}_{obs}^{-1}(\boldsymbol{Hc}-\boldsymbol{c}_{obs})\}\}$$

$$J = \frac{1}{2}\sum_{c\in\Omega}(\boldsymbol{Hc}-\boldsymbol{c}_{obs})^T\boldsymbol{S}_{obs}^{-1}(\boldsymbol{Hc}-\boldsymbol{c}_{obs}) + \frac{1}{2}\gamma_r(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)^T\boldsymbol{S}_{\sigma}^{-1}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)$$

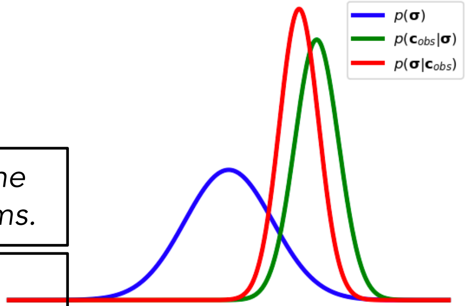
Posterior | Prior | Likelihood

$$p(\boldsymbol{\sigma}|\boldsymbol{c}_{obs}) = \frac{p(\boldsymbol{\sigma})p(\boldsymbol{c}_{obs}|\boldsymbol{\sigma})}{p(\boldsymbol{c}_{obs})}$$

Normalising constant

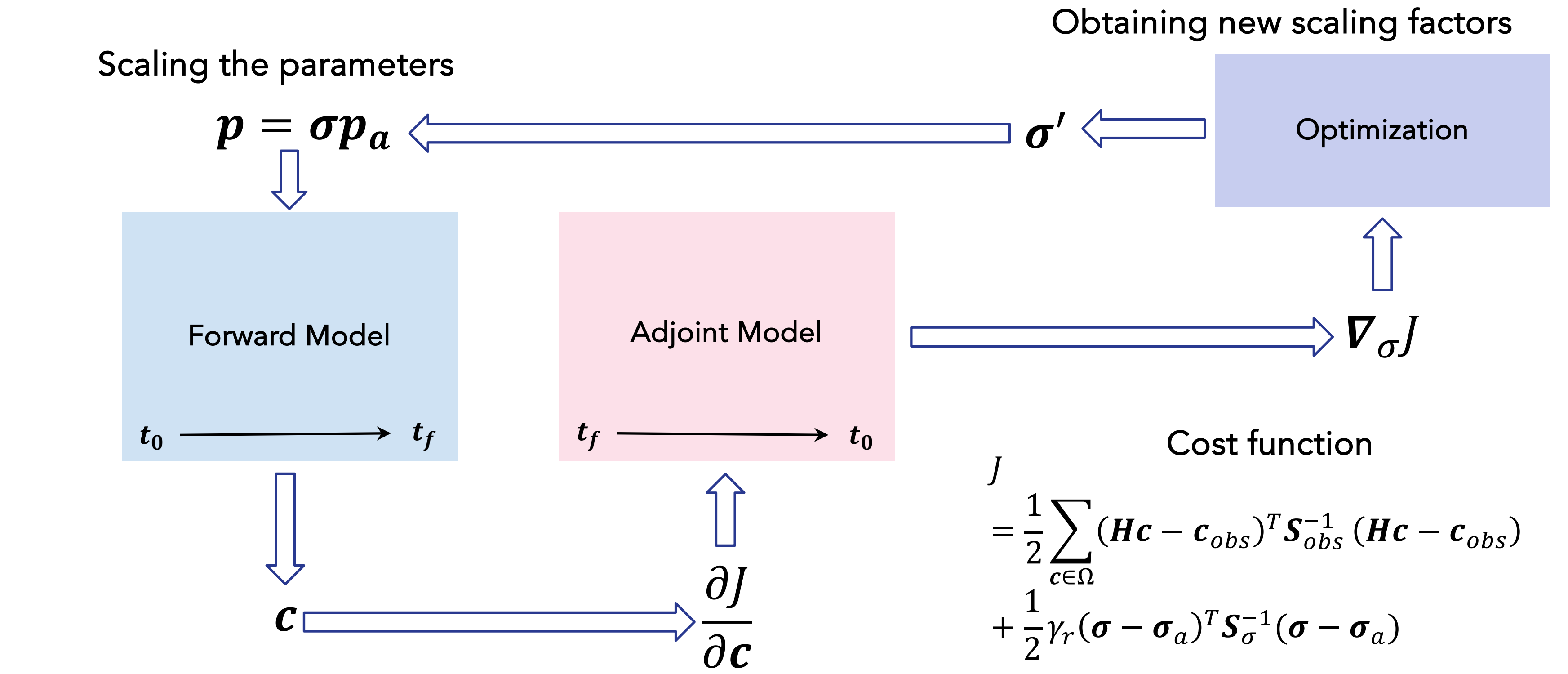
Additionally introducing a regularization parameter, which acts to control the weight given to the a priori relative to the observations, akin to specifying the strength of the priori in Bayesian terms.

p and q in $\frac{p}{2}$ and $\frac{q}{2}$ are the dimensionality of model and observation state vectors, respectively.



Adjoint model for inversion studies

Optimization flowchart



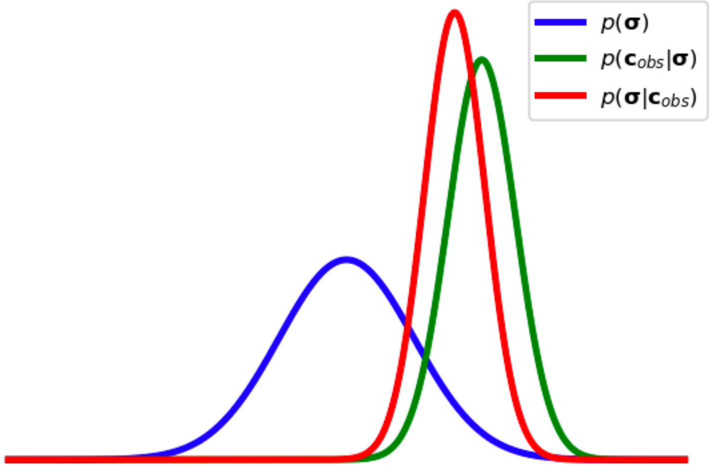
GEOS-Chem Adjoint model

Features and limitations

- Sensitivities: cheap, inversion: expensive
  - Sensitivities != Source apportionment
  - Adjoint requires an additional x2 CPU time
- Each application requires extra code development, some of which involves code validation (e.g., new emission inventories, chemistry, etc.)
  - The most recent adjoint model (v36) corresponds to GEOS-Chem v10
- Memory and I/O intensive
  - Memory usage ~x4 of standard
  - Forward model slower than standard owing to heavy I/O

Bayes' Theorem: 1-D example

The right-hand figure illustrates a univariate case, where a single parameter follows a Gaussian distribution, and we iteratively update its *a priori* distribution to its *a posterior* distribution given observed data. For a point estimate, we simply choose the parameter corresponding to the maximum *a posteriori* probability. For an interval estimate, we derive a region, such as $[a, b]$, that encompasses $1 - \alpha$ of the *a posteriori* probability.

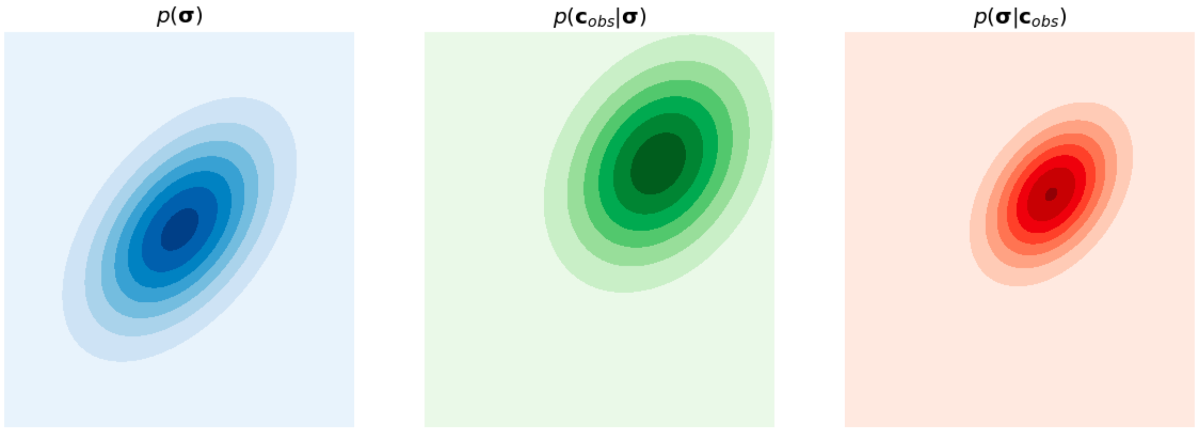


Bayesianism: Variation of beliefs about parameters in terms of fixed observed data.

Bayes' Theorem: 2/N-D example

The idea can be expanded to the multivariate joint distribution, where we have a series of parameter, each with its own distribution, and their **joint distribution** is given as $p(\boldsymbol{\sigma})$ — given a combination of each element in $\boldsymbol{\sigma}$, we have a probability. We similarly find the point/credible region corresponding to/surrounding the maximum probability in the *a posteriori* **hyperplane**.

An example of joint distributions (e.g., Gaussian) involving two parameters, with the joint probability shown in the colour space.



Kalman Filter and its ensemble variant

When the *a priori* and likelihood are Gaussian, the *a posteriori* is also Gaussian. The Kalman Filter and its ensemble variant aim to compute and estimate the mean and covariance of the *a posteriori*, respectively. It iteratively updates these estimates as new data become available, using the current *a posteriori* as the new *a priori* to drive the next iteration.

$$p(\boldsymbol{\sigma}|\boldsymbol{c}_{obs}) \propto \exp\{-\frac{1}{2}\gamma_r\{(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)^T\boldsymbol{S}_\sigma^{-1}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_a)+(\boldsymbol{Hc}-\boldsymbol{c}_{obs})^T\boldsymbol{S}_{obs}^{-1}(\boldsymbol{Hc}-\boldsymbol{c}_{obs})\}\}$$

$$p(\boldsymbol{\sigma}|\boldsymbol{c}_{obs}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{S}_{\sigma|c_{obs}}|^{1/2}}\exp\{-\frac{1}{2}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_{post})^T\boldsymbol{S}_{\sigma|c_{obs}}^{-1}(\boldsymbol{\sigma}-\boldsymbol{\sigma}_{post})\}$$

Reorganize in the standard Gaussian form

$$\boldsymbol{S}_{\sigma|c_{obs}} = (\boldsymbol{S}_\sigma^{-1}+\boldsymbol{H}^T\boldsymbol{S}_{obs}^{-1}\boldsymbol{H})^{-1}$$
$$\boldsymbol{\sigma}_{post} = \boldsymbol{S}_{\sigma|c_{obs}}(\boldsymbol{S}_{obs}^{-1}\,\boldsymbol{\sigma}_a+\boldsymbol{H}^T\,\boldsymbol{S}_{obs}^{-1}\,\boldsymbol{c}_{obs})$$

Given by GPT-4o, NOT manually verified, but in any case…

$$\boldsymbol{S}_{\sigma|c_{obs}} = (\gamma_r\boldsymbol{S}_\sigma^{-1}+\boldsymbol{H}^T\boldsymbol{S}_{obs}^{-1}\boldsymbol{H})^{-1}$$
$$\boldsymbol{\sigma}_{post} = \boldsymbol{S}_{\sigma|c_{obs}}(\gamma_r\boldsymbol{S}_{obs}^{-1}\,\boldsymbol{\sigma}_a+\boldsymbol{H}^T\,\boldsymbol{S}_{obs}^{-1}\,\boldsymbol{c}_{obs})$$

Frequentism versus Bayesianism

- Frequentism posits a single true parameter value, whereas Bayesianism views parameters as random variables with distributions, without claiming the existence of a single true posterior distribution. Instead, Bayesianism provides a framework for updating our beliefs about parameter distributions in light of new evidence.
- In some cases, as more data are observed, *the posterior* distribution may converge to a stable distribution. This stable distribution represents the limit of our updated beliefs, combining prior information with observed data, rather than a single "true" distribution in an absolute sense. Consequently, the credible region, derived from *the posterior* distribution ($p(a < \boldsymbol{\sigma} < b|\boldsymbol{c}_{obs}) = 1 - \alpha$), is simply a subset of the a posteriori distribution.

The frequentist/Bayesian divide is fundamentally a question of philosophy: the definition of probability.

MAP & 4DVar versus MLE & OLS

- The relationship between maximizing a posteriori (MAP) and minimizing the cost function in Bayesianism is analogous to the relationship between maximum likelihood estimation (MLE) and ordinary least squares (OLS) in frequentism. MLE is a special case of MAP when assuming a flat prior, where the prior does not influence the estimation. MAP is a generalization of MLE and reduces to MLE if we assume a non-informative (uniform/flat) prior. The discrepancies in the cost function definitions explain these differences.
- What about ridge regression and LASSO? Adding a Gaussian and Laplace prior on the regression coefficients?

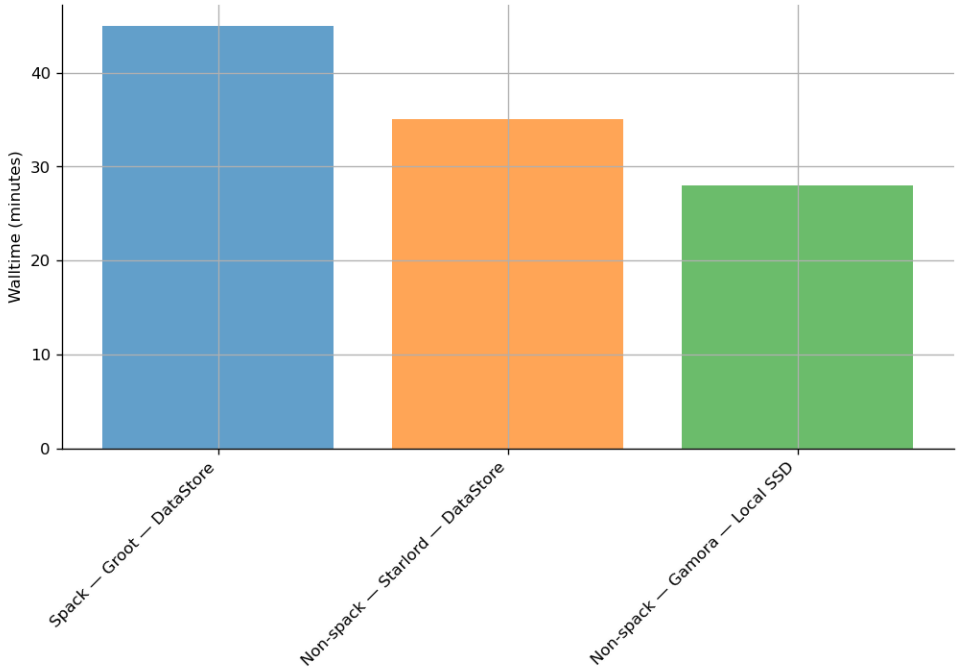
Data assimilation versus machine learning

- Outlook for Exploiting Artificial Intelligence in the Earth and Environmental Sciences (Boukabara et al., BAMS, 2021)

Machine learning		Data assimilation	
Concept	**Notation or example**	**Concept**	**Notation or example**
Labels	$\mathbf{y}$	Observations	$\mathbf{y}^o$
Features	$\mathbf{x}$	State	$\mathbf{x}$
Neural network or other learned models	$\mathbf{y}' = W(\mathbf{x})$	Physical forward model	$\mathbf{y} = H(\mathbf{x})$
Objective or loss function	$J = (\mathbf{y} - \mathbf{y}')^T(\mathbf{y} - \mathbf{y}') + J^w$	Cost function	$J = [\mathbf{y}^o - H(\mathbf{x})]^T\mathbf{R}^{-1}[\mathbf{y}^o - H(\mathbf{x})] + J^b$
Network weights ($\mathbf{w}$) regularization	$J^w = \mathbf{w}^T\mathbf{w}$	Background state ($\mathbf{x}^b$) term	$J^b = (\mathbf{x} - \mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}^b)$
		Error covariance matrices for observations and background state	**R, B**
Iterative gradient descent to find network weights $\mathbf{w}$	E.g., stochastic gradient descent; gradient computed with back propagation	For variational DA: Iterative gradient descent to find most probable state $\mathbf{x}$	E.g., conjugate gradient method; gradient computed with adjoint model

Improving the speed of running

- The time required to run a complete iteration, including both a forward and a backward run, has been reduced from 45 minutes to less than half an hour.



GC Adjoint Checkpointing files

- We generally do not save Jacobians, as this would require a large amount of space. Instead, as shown in previous slides, both F_c^n and F_p^n need <u>NOT</u> to be stored for more than a single step. However, we will need save all the intermediate values required for constructing Jacobians?

Sensitivity analysis of $\partial NO_2/\partial NO_x$