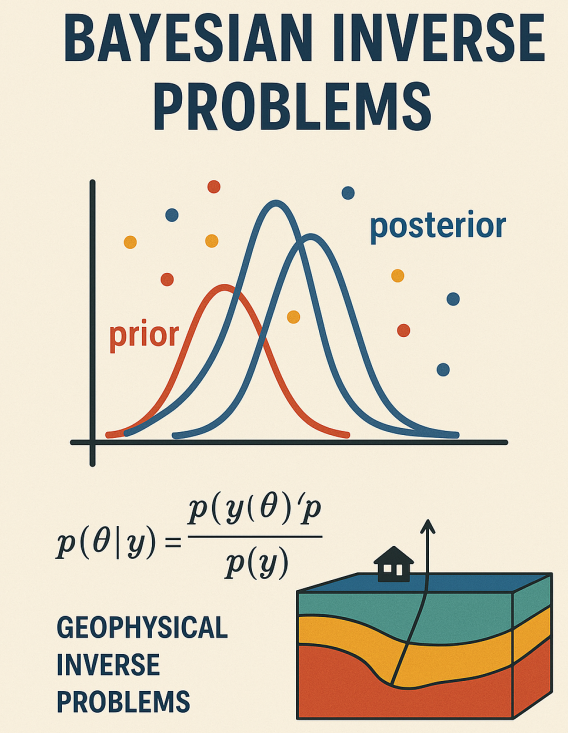
Inverse Problems and Data Assimilation

Mark Asch - MAKUTU/2025



Outline of the course

1. Introduction to inverse problems and data assimilation: overview, setting, history, definitions, examples.

2. Bayesian inverse problems.

   (a) Bayesian inference.
   (b) Bayesian/Statistical inversion theory.
   (c) Full wave inversion example.
   (d) Point and interval estimates.

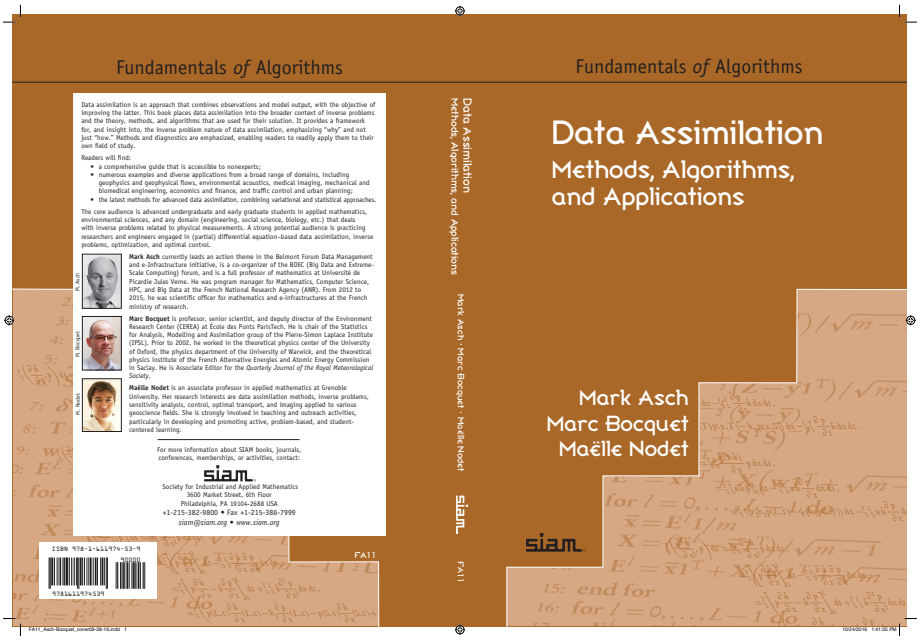
3. Posterior Estimation methods.

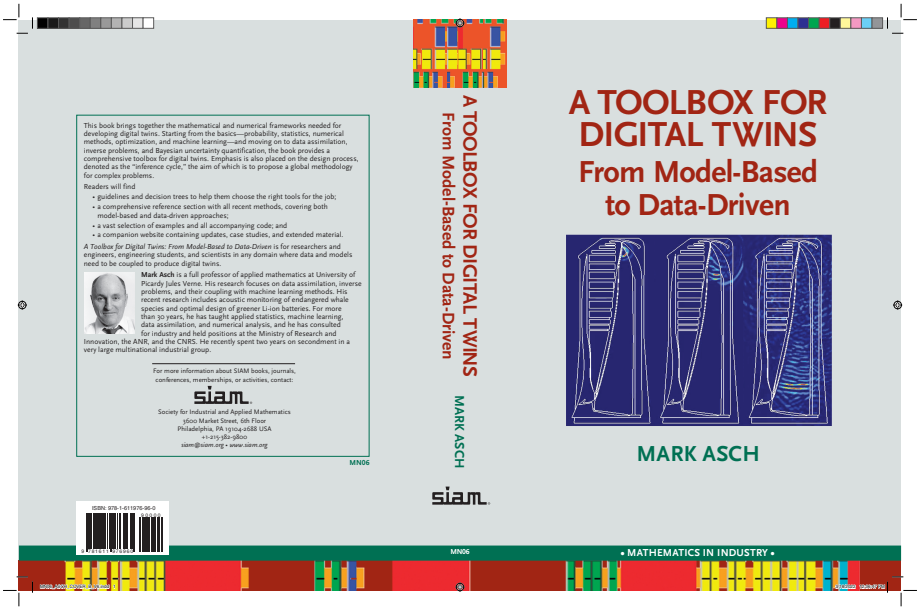
   (a) Monte Carlo methods.
   (b) Rejection Sampling. Importance Sampling.
   (c) McMC and variants for posterior estimation.
   (d) Metropolis Hastings, Gibbs and Hamiltonian McMC.
   (e) Introduction to Variational Inference (VI) for posterior estimation.

4. Statistical estimation, Kalman filters and sequential data assimilation.

   (a) Introduction to statistical DA.
   (b) The Kalman filter and Ensemble KF.
   (c) Ensemble Kalman Inversion (EKI).

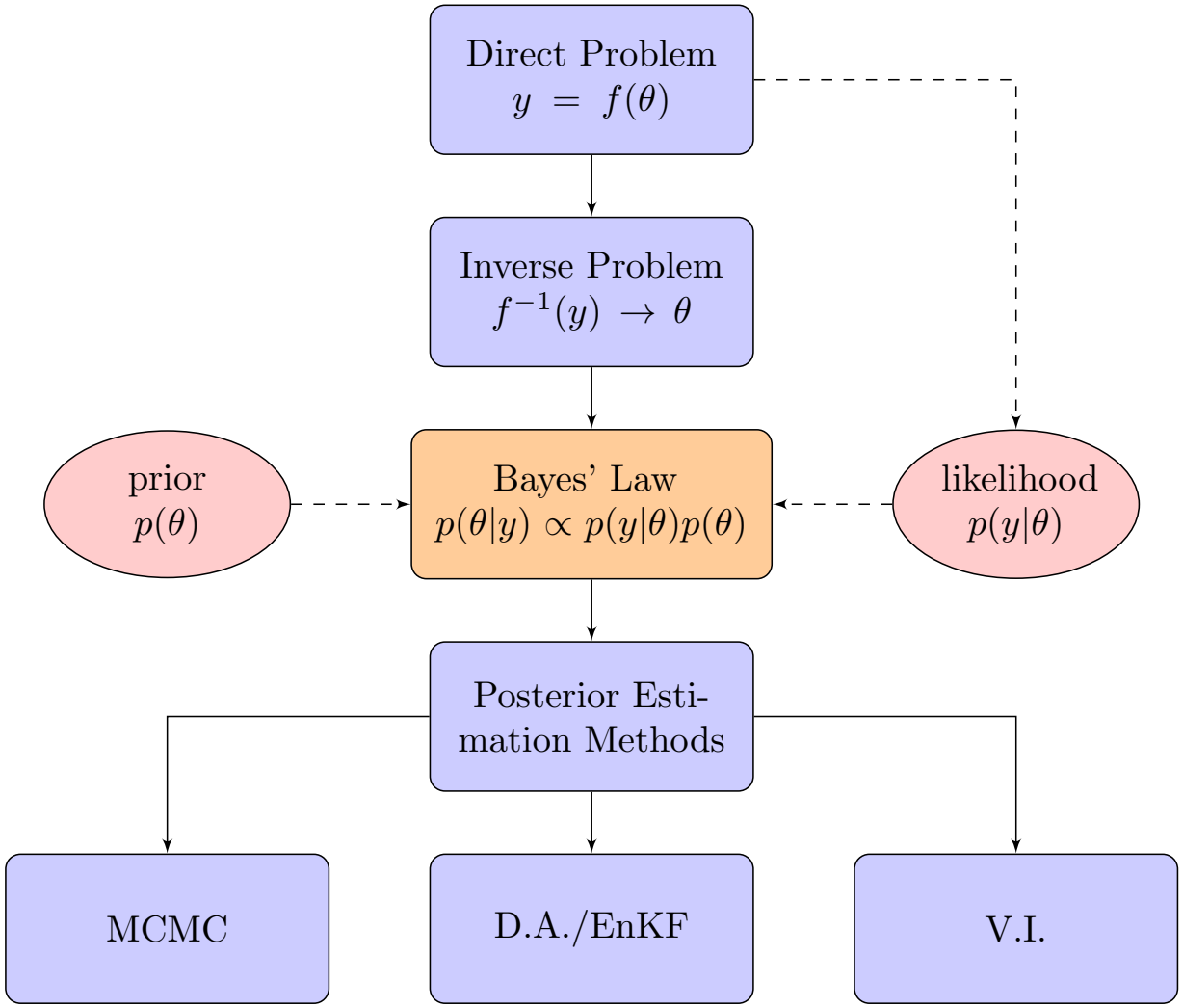
2 Reference Textbooks





INTRODUCTION

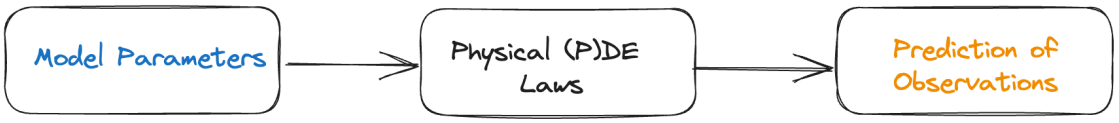
OVERVIEW


Direct Problem
$y = f(\theta)$
Inverse Problem
$f^{-1}(y) \rightarrow \theta$
prior
$p(\theta)$
Bayes' Law
$p(\theta|y) \propto p(y|\theta)p(\theta)$
likelihood
$p(y|\theta)$
Posterior Estimation Methods
MCMC
D.A./EnKF
V.I.

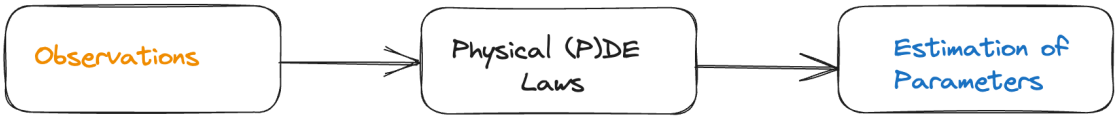
FORWARD AND INVERSE PROBLEMS

Classical Forward and Inverse Problems

Direct Problem:



Inverse Problem:



- Consider a parameter-dependent dynamical system,

$$\frac{dz}{dt} = g(t, z; \theta), \qquad z(t_0) = z_0,$$

with g known, $\theta \in \Theta$, $z(t) \in \mathbb{R}^k$.

Forward: Given θ, z_0, find $z(t)$ for $t \geq t_0$.

Inverse: Given $z(t)$ for $t \geq t_0$, find $\theta \in \Theta$.

Stochastic Forward and Inverse Problems

- All real-life systems are subject to noise—in the "physics", in the measurements, in the model, in the simulations.

- Bayes' Theorem provides a posterior probability density that captures ALL the uncertainty.

Observations

- Observation equation:

$$f(t,\theta) = \mathcal{H} z(t,\theta),$$

  where $\mathcal{H}$ is the observation operator—to account for the fact that observations are never completely known (in space-time).

- Usually we have a finite number of discrete (space-time) observations

$$\{\tilde{y}_j\}_{j=1}^{n},$$

  where

$$\tilde{y}_j \approx f(t_j,\theta).$$

Model-driven and data-driven inverse problems

- Model-driven:

$$\tilde{y}_j = f(t_j, \theta)$$

- Data-driven:

$$\tilde{y}_j = f(t_j, \theta) + \varepsilon_j,$$

where ε_j is error and requires that we introduce variability/uncertainty into the modeling and analysis.

Well-posedness

1. Existence

2. Uniqueness

3. Continuous dependence of solutions on observations.

✔ The existence and uniqueness together are also known as "identifiability".

✔ The continuous dependence is related to the "stability" of the inverse problem.

Well-posedness (mathematical)

Definition 1. Let X and Y be two normed spaces and let $K : X \to Y$ be a linear or nonlinear map between the two. The problem of finding x given y such that

$$Kx = y$$

is well-posed if the following three properties hold:

WP1 Existence—for every $y \in Y$ there is (at least) one solution $x \in X$ such that $Kx = y$.

WP2 Uniqueness—for every $y \in Y$ there is at most one $x \in X$ such that $Kx = y$.

WP3 Stability—the solution x depends continuously on the data y in that for every sequence $\{x_n\} \subset X$ with $Kx_n \to Kx$ as $n \to \infty$, we have that $x_n \to x$ as $n \to \infty$.

- This concept of ill-posedness will help us to understand and distinguish between direct and inverse

models.

- It will provide us with basic comprehension of the methods and algorithms that will be used to solve inverse problems.

- Finally, it will assist us in the analysis of "what went wrong?" when we attempt to solve the inverse problems.

Inverse Problems: General Formulation

- All inverse problems share a common formulation.

- Let the model parameters[1] be a vector (in general, a multivariate random variable), $\mathbf{m}$, and the data be $\mathbf{d}$,

$$\mathbf{m} = (m_1, \ldots, m_p) \in \mathcal{M},$$
$$\mathbf{d} = (d_1, \ldots, d_n) \in \mathcal{D},$$

  where

  $\Rightarrow$ $\mathcal{M}$ and $\mathcal{D}$ are the corresponding model parameter space and data space.

[1] Applied mathematicians often call the equation $G(m) = d$ a mathematical model and m the parameters. Other scientists call G the forward operator and m the model. We will adopt the more mathematical convention, where m will be referred to as the model parameters, G the model and d the data.

- The mapping $G\colon \mathcal{M} \to \mathcal{D}$ is defined by the direct (or forward) model

$$\mathbf{d} = g(\mathbf{m}), \tag{1}$$

where

$\Rightarrow$ $g \in G$ is an operator that describes the "physical" model and can take numerous forms, such as algebraic equations, differential equations, integral equations, or linear systems.

- Then we can add the observations or predictions, $\mathbf{y} = (y_1, \ldots, y_r)$, corresponding to the mapping from data space into observation space, $H : \mathcal{D} \to \mathcal{Y}$, and described by

$$\mathbf{y} = h(\mathbf{d}) = h\left(g(\mathbf{m})\right),$$

where

$\Rightarrow$ $h \in H$ is the observation operator, usually some projection into an observable subset of $\mathcal{D}$.

- Note that, in addition, there will be some random noise in the system, usually modeled as additive noise, giving the more realistic, stochastic direct model

$$\mathbf{d} = g(\mathbf{m}) + \epsilon, \tag{2}$$

where

$\Rightarrow$ ϵ is a random vector.

Inverse Problems: Classification

- We can now classify inverse problems as:

  $\Rightarrow$ deterministic inverse problems that solve (1) for $\mathbf{m}$,

  $\Rightarrow$ statistical/stochastic inverse problems that solve (2) for $\mathbf{m}$.

- The first class will be treated by linear algebra and optimization methods.

- The latter can be treated by a Bayesian (filtering) approach, and also by

  $\Rightarrow$ weighted least-squares,

  $\Rightarrow$ maximum likelihood,

  $\Rightarrow$ Data Assimilation,

  $\Rightarrow$ optimal transport techniques.

- Both classes can be further broken down into:

⇒ Linear inverse problems, where (1) or (2) are linear equations. These include linear systems—that are often the result of discretizing (partial) differential equations—and integral equations.

⇒ Nonlinear inverse problems where the algebraic or differential operators are nonlinear.

- Finally, since most inverse problems cannot be solved explicitly, computational methods are indispensable for their solution—see [Asch2022]

- Also note that we will be inverting here between the model and data spaces, that are usually both of high dimension and thus this model-based inversion will invariably be computationally expensive.

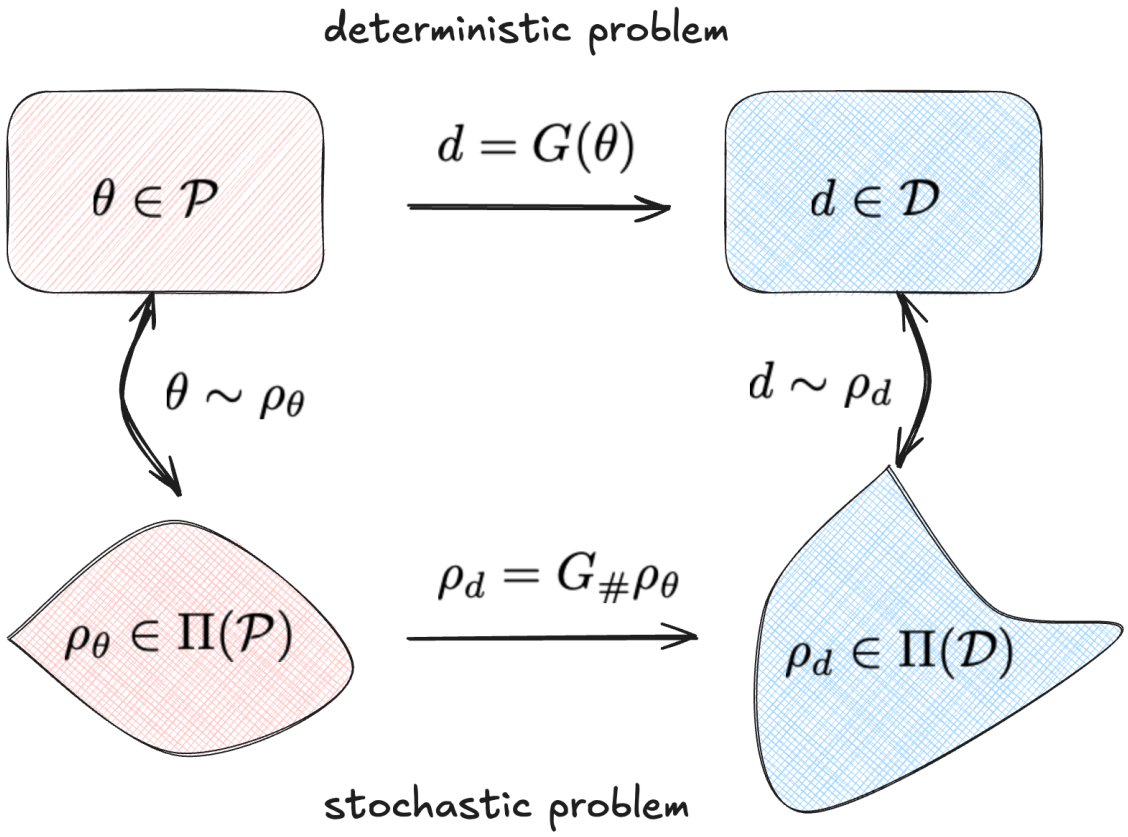
- This will motivate us to employ

⇒ reduced order methods based on suitable projections,

⇒ inversion between the data and observation spaces in a purely data-driven approach, using

machine learning methods...

Deterministic vs. Stochastic Inverse Problems



Remark 1. The stochastic inverse case could be considered as an optimal transport problem!

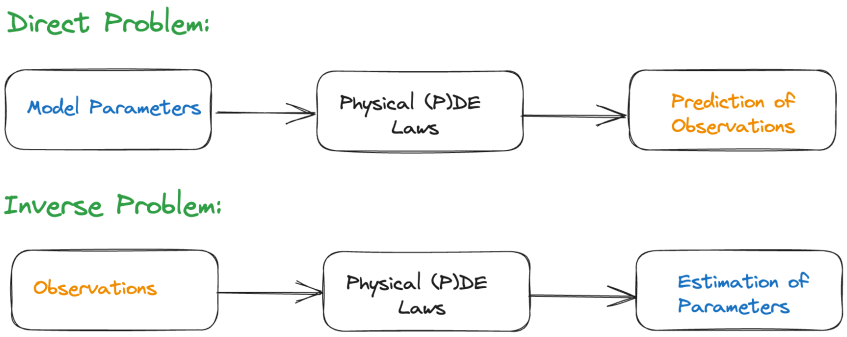
DETERMINISTIC INVERSE PROBLEMS FOR DE's

(DIP)

DIP Overview

Definition 2. *Deterministic inverse problems* (DIP) involve recovering unknown parameters, coefficients, or initial/boundary conditions of a differential equation from observations of its solution.

- Unlike forward (direct) problems where we compute solutions form a given, complete problem specification, inverse problems work "backwards" from observed data to infer the underlying model parameters.



DIP Mathematical Framework

- The Direct (Forward) Problem

Let u denote the state variable (solution) and θ denote parameters/coefficients. The direct problem is:

Given: Parameter vector $\theta \in \Theta \subset \mathbb{R}^p$

Find: Solution $u \in U$ satisfying

$$\mathcal{L}(u;\theta) = f \quad \text{in } \Omega,$$

$$\mathcal{B}(u;\theta) = g \quad \text{on } \partial\Omega,$$

where

- $\mathcal{L}$ is a differential operator (ODE or PDE) ,
- $\mathcal{B}$ represents boundary/initial conditions,

- Ω is the spatial domain (could be space-time),
- U is an appropriate function space (e.g., $H^1(\Omega)$, $C^2(\Omega)$).

We denote the **parameter-to-solution map** (forward operator)

$$\mathcal{G} : \Theta \to U, \quad \theta \mapsto u(\theta).$$

- The Inverse Problem

Given: Observations $d \in \mathbb{R}^m$ of the solution

Find: Parameter $\theta^* \in \Theta$ such that the solution matches observations that are related to the solution through an **observation operator** $\mathcal{H} : U \to \mathbb{R}^m$

$$d = \mathcal{H}(u(\theta^*)) + \eta,$$

where η represents measurement noise, taken as zero for the DIP case.

- The inverse problem seeks to determine θ^* from knowledge of d, which amounts to inverting the composed operator

$$\mathcal{M} = \mathcal{H} \circ \mathcal{G} : \Theta \to \mathbb{R}^m.$$

DIP Optimization Formulation

Since inverse problems are typically **ill-posed** (in Hadamard's sense: existence, uniqueness, or stability may fail), we reformulate as an optimization problem.

- Least Squares Formulation

  $\Rightarrow$ Minimize the **data misfit functional**

$$J(\theta) = \frac{1}{2}\|d - \mathcal{G}(u(\theta))\|^2,$$

  where $\|\cdot\|$ is typically the ℓ^2 norm in $\mathbb{R}^m$.

- Tikhonov Regularization

  $\Rightarrow$ To ensure well-posedness and incorporate prior information

$$J_\alpha(\theta) = \frac{1}{2}\|d - \mathcal{G}(u(\theta))\|^2 + \frac{\alpha}{2}\|\mathcal{R}(\theta - \theta_0)\|^2,$$

  where

- $\alpha > 0$ is the **regularization parameter** (see APPENDIX for details),
- $\mathcal{R}$ is a regularization operator (e.g., identity, gradient, Laplacian),
- θ_0 is a prior estimate.

- Optimization Methods: Gradient-Based

1. **Steepest Descent:**

$$\theta^{k+1} = \theta^k - \beta_k \nabla J(\theta^k)$$

2. **Gauss-Newton Method:** Linearize $\mathcal{M}(\theta) \approx \mathcal{M}(\theta^k) + \mathcal{M}'(\theta^k)(\theta - \theta^k)$ and solve

$$\left(\mathcal{M}'(\theta^k)^T \mathcal{M}'(\theta^k) + \alpha I\right) \delta\theta^k = \mathcal{M}'(\theta^k)^T (d - \mathcal{M}(\theta^k))$$

$$\theta^{k+1} = \theta^k + \delta\theta^k$$

3. **Adjoint Method (for PDE-constrained optimization):** Compute gradients efficiently by solving an adjoint equation (avoids computing the full Jacobian)—see below and Practical #2.

- Optimization Methods: Derivative-Free

  $\Rightarrow$ Genetic algorithms.
  - $\rightarrow$ Ensemble Kalman Filter (EnKF).
  - $\rightarrow$ Bayesian optimization.

Example: ODE Inverse Problem

- Direct Problem: Consider a first-order decay model

$$\frac{du}{dt} = -\theta u(t), \quad t \in [0, T]$$

$$u(0) = u_0$$

where $\theta > 0$ is the unknown decay rate. The analytical solution is

$$u(t; \theta) = u_0 e^{-\theta t}$$

- Inverse Problem

  $\Rightarrow$ **Given:** Measurements d_i at times t_i for $i = 1, \ldots, m$

$$d_i = u(t_i; \theta^*) + \eta_i, \quad \eta_i = 0.$$

  $\Rightarrow$ **Find:** θ^*

- Optimization Problem

$$\min_{\theta>0} J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(d_i - u_0 e^{-\theta t_i})^2$$

- Gradient Computation

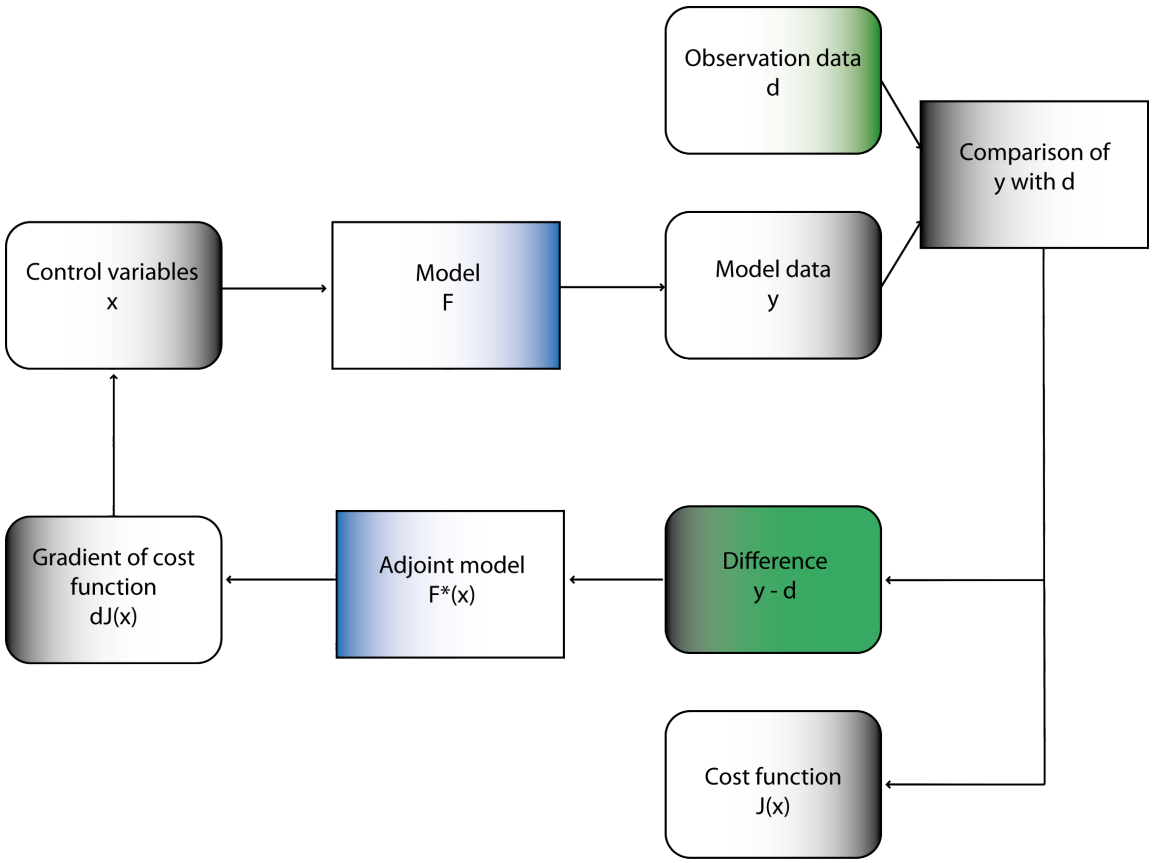
$$\frac{dJ}{d\theta} = -\sum_{i=1}^{m}(d_i - u_0 e^{-\theta t_i}) \cdot u_0 t_i e^{-\theta t_i}$$

- Solution: Set to zero and solve, gives a nonlinear equation for θ, typically solved iteratively (e.g., modified Newton's method).

ADJOINT METHOD FOR DIP

Adjoint Methods (I)

- A very general approach for solving inverse problems... including Machine Learning!
- Variational DA is based on an adjoint approach.



Adjoint Methods (II) - definition

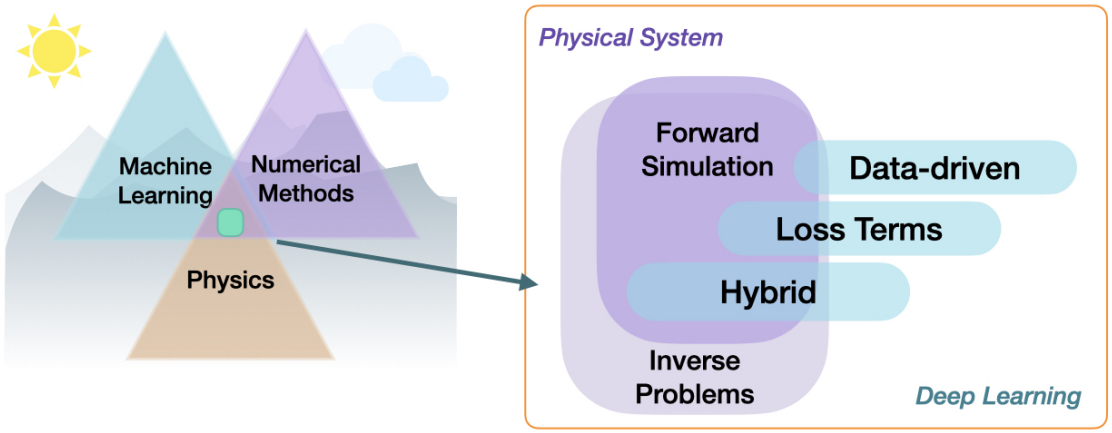
Definition 3. *An* **adjoint method** *is a general mathematical technique, based on variational calculus, that enables the computation of the gradient of an objective, or cost functional with respect to the model parameters in a very efficient manner.*

- This method will be addressed in the Practical session.
- See also the lecture material in `11_DA_adj.pdf`
- An adaptation of the adjoint method, known as "backpropagation" forms the basis for almost all machine learning algorithms!

Backprop, Differentiable Physics, Adjoints & Machine Learning

- Reference: `https://physicsbaseddeeplearning.org/` from Nils Thuerey's group at TUM.

- Backprop $\Leftrightarrow$ Adjoint Method $\Leftrightarrow$ Differentiable Physics



- Example workflow:

  1. Parameterize unknown quantities (ϑ).
  2. Run simulation with current ϑ estimate.
  3. Compute data misfit: L = ||simulation(ϑ) - observations||^2

4. Backpropagate $\partial \mathrm{L}/\partial \vartheta$ through simulator.
5. Update ϑ via gradient-based optimization (Adam, L-BFGS, etc.)

- Advantages over traditional adjoint methods:

  $\Rightarrow$ Automatic differentiation eliminates hand-derived adjoint equations.

  $\Rightarrow$ Easy to incorporate neural network components (hybrid models).

  $\Rightarrow$ Handles complex, non-linear physics naturally.

- Differentiable physics keeps the (coarse) solver in the training loop, whereas PINN is based on augmenting the loss function with the physics.

EXAMPLES

Inverse Problems - Application Domains

- *Computational imaging*—recovering the true image from a blurred and noisy observation.
- *Geophysics*—inferring the conductivity of the sub-surface from measurements at wells; inferring the geological substructure from measurements on the earth's surface, or the seafloor surface.
- *Machine learning*—building an underlying model from observed data points.
- *Others*: non-destructive testing, astrophysics, medicine, weather prediction (DA), . . .

Inverse Problem Examples

Shared Mathematical Formulation

Inverse Problem

Given observation data $y \in Y$, determine the unknown $u \in U$ such that

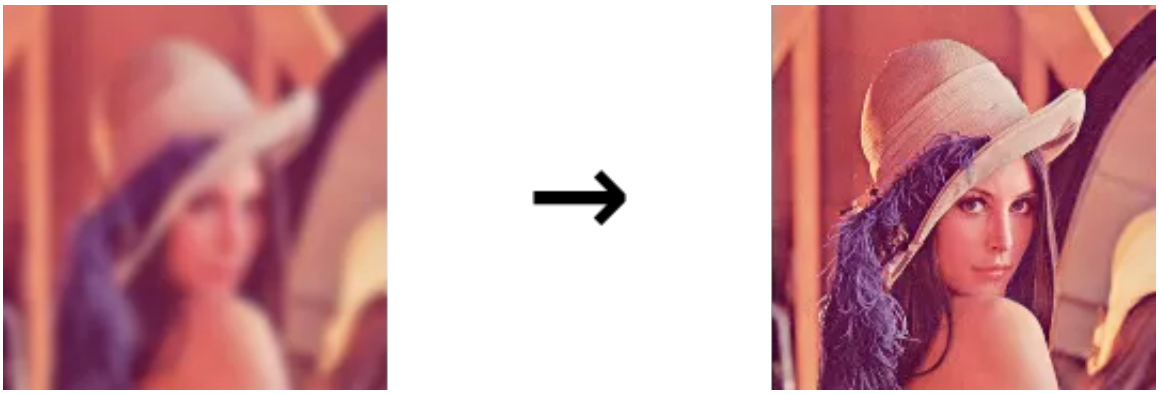
$$y = \mathcal{G}(u) + \eta,$$

where $\mathcal{G}$ is a mathematical model of a process and η is random (observational) noise.

Inverse Problem Examples

Example 1: Imaging

- **Goal**: reconstruct an image u given a noisy, partial observation y.



Inverse Problem Examples

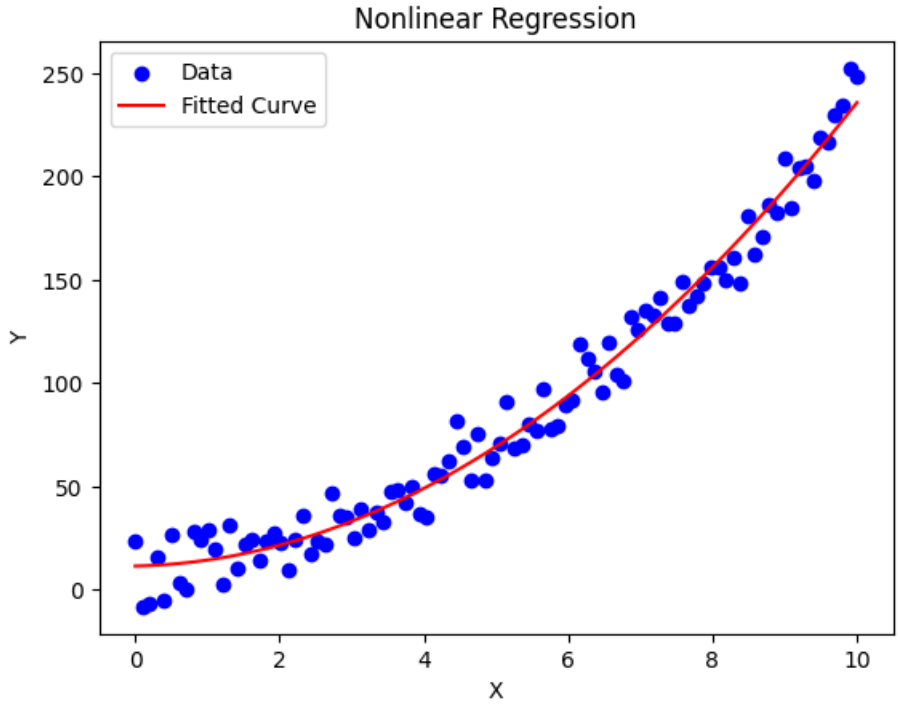
Example 1: Imaging

- **Goal**: reconstruct an image u given a noisy, partial observation y.
- **Unknown**: $u \in \mathbb{R}^{d_u}$, the pixel values of the image.
- **Map**: $\mathcal{G} = G \in \mathbb{R}^{d_y \times d_u}$, linear in many imaging problems, where G incorporates mechanisms such as
  - $\Rightarrow$ blurring ($\rightarrow$ averaging over a neighbourhood of pixels)
  - $\Rightarrow$ Fourier transform ($\rightarrow$ observations in frequency domain)
  - $\Rightarrow$ mask ($\rightarrow$ partial observations)
- Observations: $y = \mathcal{G}(u) + \eta \in \mathbb{R}^{d_y}$.

Inverse Problem Examples

Example 2: Regression

- **Goal**: reconstruct a function f given noisy point values (measurements) $\{f(x_i)\}$.



Inverse Problem Examples

Example 2: Regression

- **Goal**: reconstruct a function f given noisy point values (measurements) $\{f(x_i)\}$.

- **Unknown**: $u \in \mathbb{R}^{d_u}$, coefficients in a (polynomial) basis expansion

$$f(x;u) = \sum_{j=1}^{d_u} u_j \phi_j(x),$$

  where $\{\phi_j\}_{j=1}^{d_u}$ are linearly independent, eg. $\phi_j(x) = x^{j-1}$.

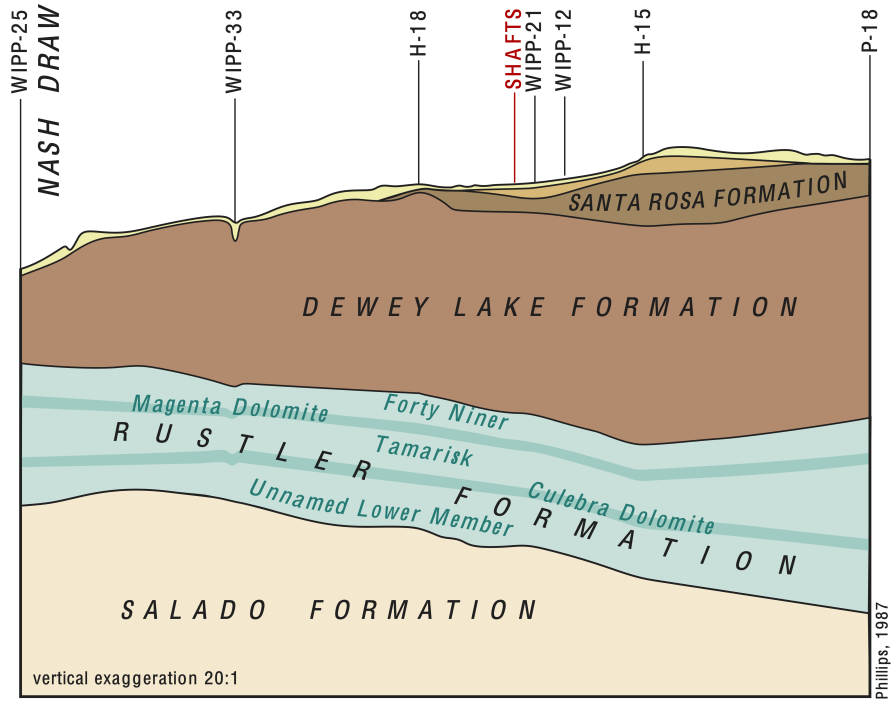
- **Map**: $\mathcal{G}$ implicitly defined by $u \mapsto \{f(x_i;u)\}_{i=1}^{d_y}$ and $\mathcal{G} = G$ is linear (Vandermonde matrix) with $a_{ij} = \phi_j(x_i)$.

- Observations: $y = \{f(x_i;u) + \eta_i\}_{i=1}^{d_y} \in \mathbb{R}^{d_y}$.

Inverse Problem Examples

Example 3: Porous Medium Flow

- **Goal**: reconstruct the hydraulic conductivity k of the subsurface given noisy measurements of the water pressure $\{p(x_i)\}$.



Cross-section at WIPP

Inverse Problem Examples

Example 3: Porous Medium Flow

- **Goal**: reconstruct the hydraulic conductivity k of the subsurface given noisy measurements of the water pressure $\{p(x_i)\}$.
- **Unknown**: $u \in \mathbb{R}^{d_u}$, coeff's in a basis expansion

$$f(x;u) = \phi_0(x) + \sum_{j=1}^{d_u} u_j \phi_j(x),$$

  where $\{\phi_j\}_{j=1}^{d_u}$ are linearly independent and ϕ_0 ensures positivity of k.

- **Map**: $\mathcal{G}$ implicitly defined by $u \mapsto \{p(x_i;u)\}_{i=1}^{d_y}$, where p is the solution of (Darcy)

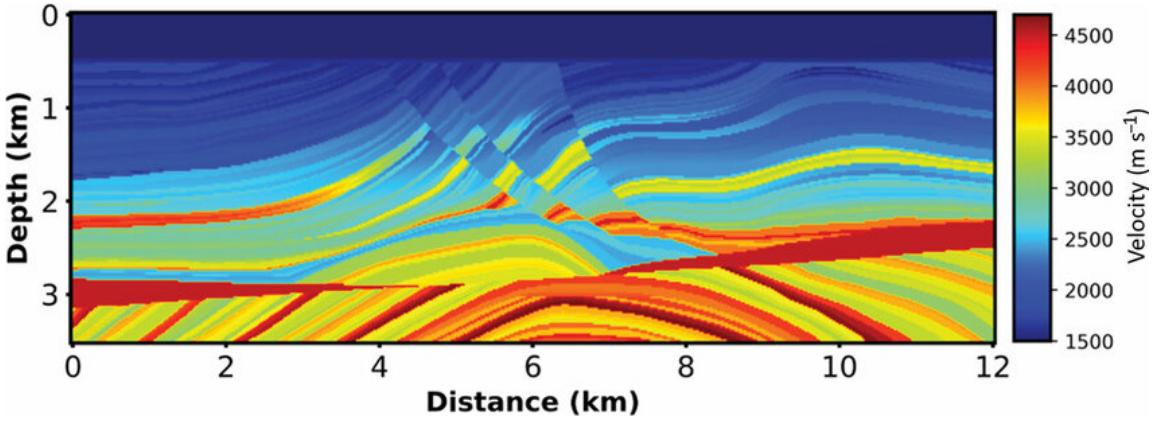
$$-\nabla \cdot (k(x;u)\nabla p(x;u)) = h(x)$$

- Observations: $y = \{p(x_i;u) + \eta_i\}_{i=1}^{d_y} \in \mathbb{R}^{d_y}$.

Inverse Problem Examples

Example 4: Full Wave Inversion (FWI)

- **Goal**: reconstruct the subsurface properties ρ, κ given noisy measurements of the acoustic/elastic pressure at given times $\{p(x_i, t_j)\}$.



Inverse Problem Examples

Example 4: Full Wave Inversion (FWI)

- **Goal**: reconstruct the subsurface properties ρ, κ given noisy measurements of the acoustic/elastic pressure at given times $\{p(x_i, t_j)\}$.

- **Unknown**: $u \in \mathbb{R}^{d_u}$, coeff's in a basis expansion

$$f(x; u) = \phi_0(x) + \sum_{j=1}^{d_u} u_j \phi_j(x),$$

- **Map**: $\mathcal{G}$ implicitly defined by $u \mapsto \{p(x_i, t_n; u)\}_{i=1}^{d_y}$, where p is the solution of

$$\rho(x; u)\frac{\partial^2 p}{\partial t^2} - \nabla \cdot (\kappa(x; u)\nabla p(x; u)) = h(x)$$

- Observations: $y_n = \{p(x_i, t_n; u) + \eta_i\}_{i=1}^{d_y} \in \mathbb{R}^{d_y}$.

DIP vs BIP

Deterministic/Bayesian IP Standoff

Our test problem is the estimation of an unknown (or badly known) coefficient, m, in an elliptic partial differential equation. This is a protoypical problem setting, and can be extended to a large number of other pde's, contexts and problems.

- Let $\Omega \subset \mathbb{R}^n$, $n \in \{1, 2, 3\}$ be an open, bounded domain, and consider the following deterministic inverse problem (DIP):

$$\min_m J(m) := \frac{1}{2}\int_\Omega (u - u_{\rm d})^2 \, {\rm d}x + \frac{\gamma}{2}\int_\Omega |\nabla m|^2 \, {\rm d}x, \tag{3}$$

where u is the solution of a Poisson, boundary-value problem,

$$\begin{aligned} -\nabla \cdot (\exp(m)\nabla u) &= f \text{ in } \Omega, \\ u &= 0 \text{ on } \partial\Omega. \end{aligned} \tag{4}$$

$\Rightarrow$ $m \in \mathcal{M}_{\mathrm{ad}} := \{m \in H^1(\Omega) \bigcap L^\infty(\Omega)\}$ the unknown coefficient field,

$\Rightarrow$ u_{d} denotes (possibly noisy) measured data,

$\Rightarrow$ $f \in H^{-1}(\Omega)$ a given force, and

$\Rightarrow$ $\gamma \geq 0$ the regularization parameter.

DIP

Numerical Solution

- Simulations will show the computational cost as function of problem size N of

  $\Rightarrow$ (1) direct simulations, and
  $\Rightarrow$ (2) deterministic inverse probem.

- Direct solution could be seen as a random differential equation (a differential equation with random coefficients) approach, where the parameter is perturbed randomly, but

  $\Rightarrow$ there is no propagation of uncertainty;
  $\Rightarrow$ we have no guarantee that the ensemble is representative;
  $\Rightarrow$ we require a very large number of simulations to obtain reliable (but, see above 2 points) statistics.

- Some advantages of the RDE approach:

$\Rightarrow$ extremely simple and non-intrusive (requires no modification of the direct code);

$\Rightarrow$ can aid falsification (in a Bayesian approach) where we would like to ascertain whether realizations fall within a given distribution or not.

DIP

Computational Scaling

- Computational cost as a function of the problem size[2].

N	#dof	Time Fwd (s)	Time DIP (s)
32	1024	0.0277	3.2
64	4096	0.0446	10.9
128	16384	0.1336	61.9
256	65536	0.6138	368.1
512	262144	3.6356	≈ 3000
1024	1×10^6	27.112	

Table 1: CPU time for deterministic Poisson equation (left), deterministic inverse problem (right) - factor of 500X between direct and inverse.

[2] Villa, U. and Petra, N. and Ghattas, O. *hIPPYlib: an Extensible Software Framework for Large-scale Deterministic and Bayesian Inversion.* 2016. `http://hippylib.github.io`

- The results (Table 1 and Figure 1) show a cubic (power of 3) exponential growth.

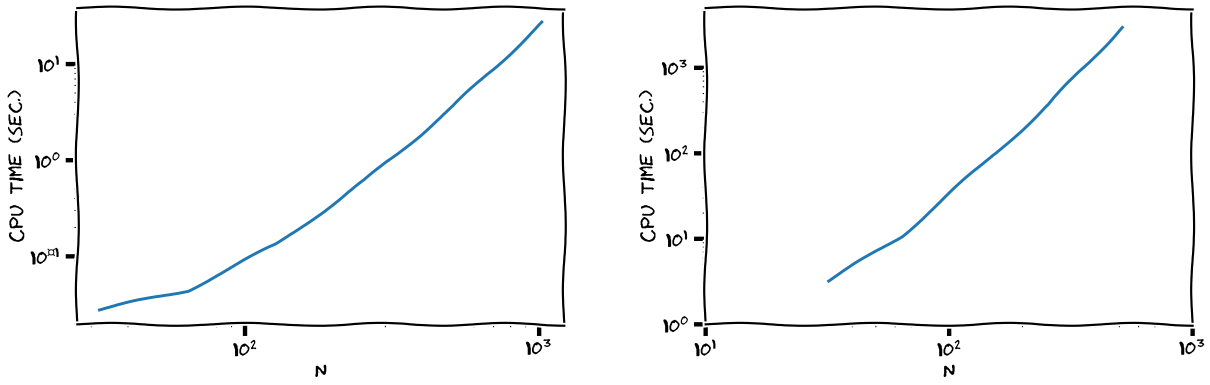


Figure 1: CPU time as function of problem size: direct problem (left), inverse problem (right).

- The computational cost for a direct solution scales as $\mathcal{O}\left(N^2\right)$.

- The computational cost for an inverse problem solution is approximately $500\times$ that for the direct problem, i.e. $\mathcal{O}\left(N^3\right)$.

BIP

Problem Formulation

- Determining the permeability of an unknown medium (subsurface rock, battery cell, etc.) is enormously important in a range of different applications. Among these applications are:

  $\Rightarrow$ the prediction of transport of radioactive waste from underground waste repositories,
  $\Rightarrow$ the forecast of geothermal production,
  $\Rightarrow$ the optimization of oil recovery from underground fields,
  $\Rightarrow$ the electrochemical behaviour of battery cells.

- Darcy's law is an excellent model of the pressure field as a function of the permeability,

$$-\nabla \cdot (k \nabla p) = 0, \quad x \in D,$$
$$p = h, \quad x \in \partial D.$$

- The inverse problem is: find the permeability k from observations/measurements of the pressure at points in the interior of D.

BIP

Numerical Solution

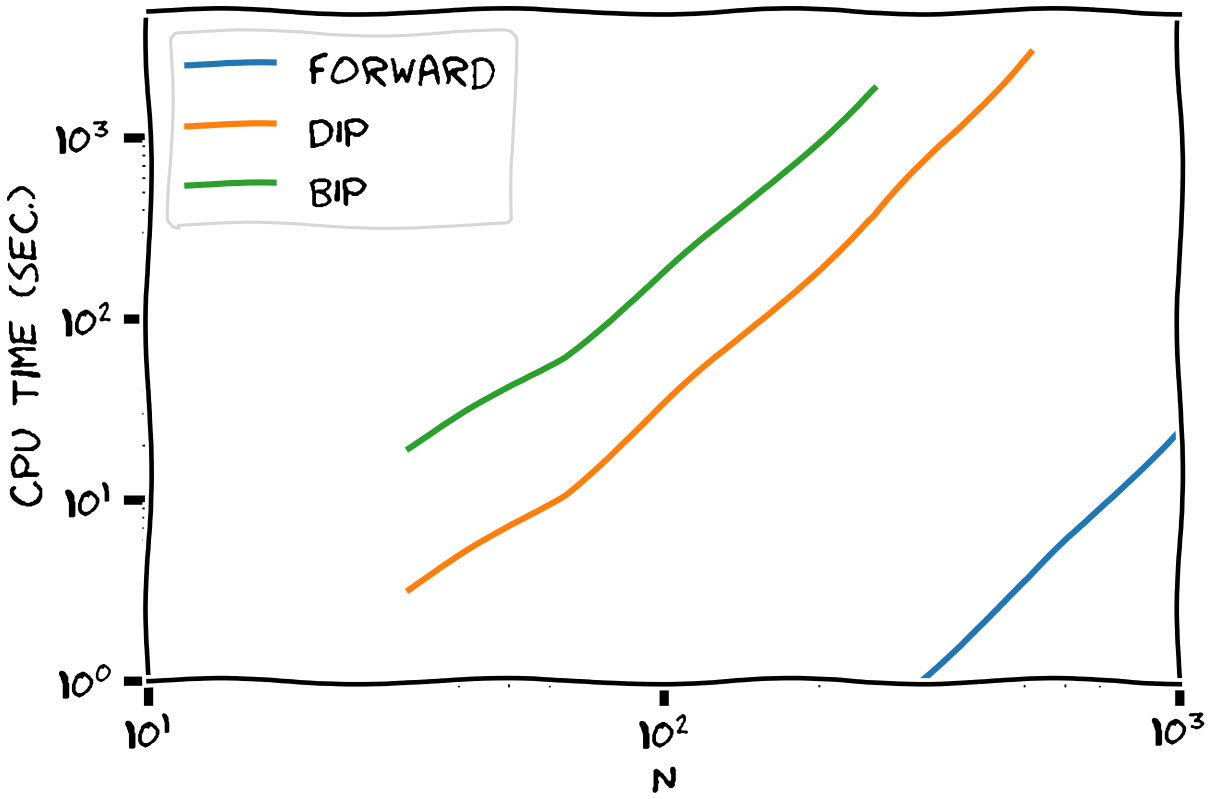
N	#dof	Time Fwd	Time DIP	Time BIP (s)
32	1024	0.03	3.2	19.3
64	4096	0.04	10.9	63.1
128	16384	0.13	61.9	317.0
256	65536	0.61	368.1	1865.8
512	262144	3.64	≈ 3000	
1024	1×10^6	27.1		

Table 2: CPU times for MAP solution of Bayesian inverse problem

- Method used here [Ghattas, et al] is state-of-the-art, scalable, adjoint-based algorithm, based on a stochastic Newton McMC method with MAP-based Hessian.

- Comparison of CPU times for forward and inverse

problems.



- **Remark**: Cost of BIP is 5-50X DIP, which is 2 500-25 000X the direct problem (depending on the required posterior statistics).

CONCLUSIONS

IP: conclusions

- Ill-posedness must always be addressed.
- Deterministic inversion produces a difficult optimization problem, requiring delicate regularization.
- Statistical inversions, based on Bayesian analysis are one approach for bypassing the ill-posedness, but introduce new theoretical and computational challenges.
- Kalman filters have a long tradition and a strong theoretical basis and they can solve the Bayesian inverse problem. Ensemble Kalman filters can be used as a basis for the formulation and solution of more general inverse problems.

References

1. M. Asch. *A Toolbox for Digital Twins—From Model-Based to Data-Driven*. SIAM, 2022.

2. A. Tarantola. *Inverse problem theory and methods for model parameter estimation.* SIAM, 2005.

3. J.-L. Lions. Exact controllability, stabilization and perturbations for distributed systems. *SIAM Rev.*, **30**(1):1–68, 1988.

4. J. Nocedal, S.J. Wright. *Numerical Optimization*. Springer, 2006.

5. F. Tröltzsch. *Optimal Control of Partial Differential Equations*. AMS, 2010.

6. Ghattas Group. `http://users.ices.utexas.edu/~omar/`

APPENDIX

Tikhonov Regularization

Tikhonov Regularization: Introduction

- Tikhonov regularization (TR) is probably the most widely used method for regularizing ill-posed, discrete and continuous inverse problems.

- Note that the LASSO and ridge regression methods—see ML Lectures—are special cases of TR,

- The theory is the subject of entire books...

- Recall:

  $\Rightarrow$ the objective of TR is to reduce, or remove, ill-posedness in optimization problems by modifying the objective function.

  $\Rightarrow$ the three sources of ill-posedness: non-existence, non-uniqueness and sensitivity to perturbations.

  $\Rightarrow$ TR, in principle, addresses and alleviates all three sources of ill-posedness and is thus a vital tool for the solution of inverse problems.

Tikhonov Regularization: Formulation

- The most general TR objective function is

$$\mathcal{T}_\alpha(\mathbf{m};\mathbf{d}) = \rho\left(G(\mathbf{m}),\mathbf{d}\right) + \alpha J(\mathbf{m}),$$

  where

  $\Rightarrow$ ρ is the *data discrepancy functional* that quantifies the difference between the model output and the measured data;

  $\Rightarrow$ J is the *regularization functional* that represents some desired quality of the sought for model parameters, usually smoothness;

  $\Rightarrow$ α is the *regularization parameter* that needs to be *tuned*, and determines the relative importance of the regularization term.

- Each domain, each application and each context will require *specific choices* of these three items,

and often we will have to rely either on previous experience, or on some sort of numerical experimentation (trial-and-error) to make a good choice.

- In some cases there exist empirical algorithms, in particular for the choice of α.

Tikhonov Regularization: Discrepancy

The most common *discrepancy functions* are:

- *least-squares*,

$$\rho_{\mathrm{LS}}(\mathbf{d}_1, \mathbf{d}_2) = \frac{1}{2}\|\mathbf{d}_1 - \mathbf{d}_2\|_2^2,$$

- *1-norm*,

$$\rho_1(\mathbf{d}_1, \mathbf{d}_2) = |\mathbf{d}_1 - \mathbf{d}_2|,$$

- *Kullback-Leibler distance*,

$$\rho_{\mathrm{KL}}(d_1, d_2) = \langle d_1, \log(d_1/d_2) \rangle \,,$$

where d_1 and d_2 are considered here as probability density functions. This discrepancy is valid in the Bayesian context.

Tikhonov Regularization: Regularization

The most common *regularization functionals* are derivatives of order one or two.

- *Gradient smoothing*:

$$J_1(\mathbf{m}) = \frac{1}{2}\|\nabla \mathbf{m}\|_2^2,$$

  where ∇ is the gradient operator of first-order derivatives of the elements of $\mathbf{m}$ with respect to each of the independent variables.

- *Laplacian smoothing*:

$$J_2(\mathbf{m}) = \frac{1}{2}\|\nabla^2 \mathbf{m}\|_2^2,$$

  where $\nabla^2 = \nabla \cdot \nabla$ is the Laplacian operator defined as the sum of all second-order derivatives of $\mathbf{m}$ with respect to each of the independent variables.

TR: Computing the Regularization Parameter

- Once the data discrepancy and regularization functionals have been chosen, we need to tune the regularization parameter, α.

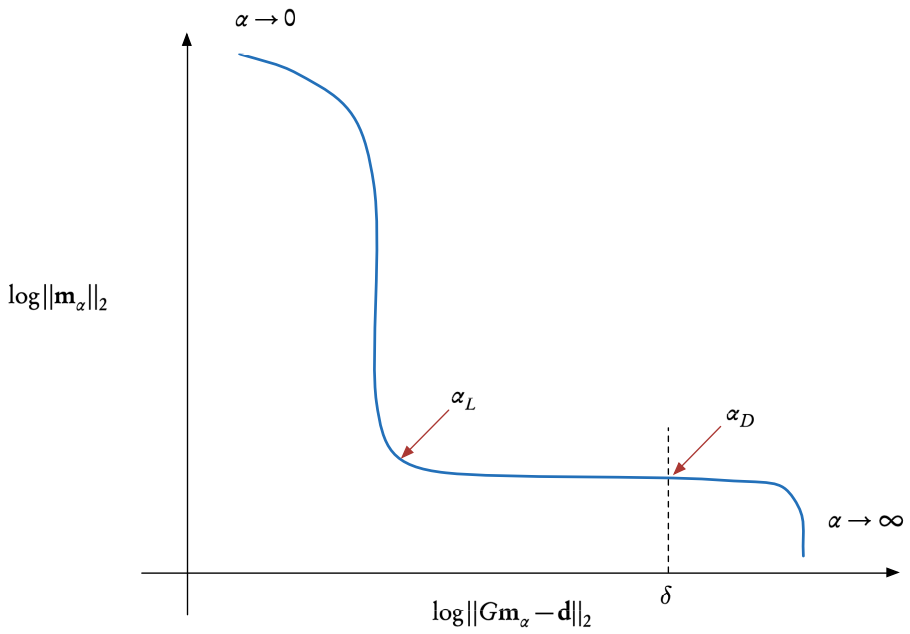
- We have here, similarly to the bias-variance trade-off of ML Lectures, a competition between the discrepancy error and the magnitude of the regularization term.

  $\Rightarrow$ We need to choose, the best compromise between the two.

- We will briefly present three frequently used approaches:

  1. L-curve method.
  2. Discrepancy principle.
  3. Cross-validation and LOOCV.

TR: Computing the Regularization Parameter (II)



- The L-curve criterion is an empirical method for picking a value of α.

  $\Rightarrow$ Since $e_m(\alpha) = \|\mathbf{m}\|_2$ is a strictly decreasing function of α and $e_d(\alpha) = \|G\mathbf{m} - \mathbf{d}\|_2$ is a strictly increasing one,

  $\Rightarrow$ we plot $\log e_m$ against $\log e_d$ we will always obtain an L-shaped curve that has an "elbow" at

the optimal value of $\alpha = \alpha_L$, or at least at a good approximation of this optimal value—see Figure .

$\Rightarrow$ This trade-off curve gives us a visual recipe for choosing the regularization parameter, reminiscent of the bias-variance trade off

$\Rightarrow$ The range of values of α for which one should plot the curve has to be determined by either trial-and-error, previous experience, or a balancing of the two terms in the TR functional.

- The **discrepancy principle**

$\Rightarrow$ choose the value of $\alpha = \alpha_D$ such that the residual error (first term) is equal to an *a priori* bound, δ, that we would like to attain.

$\Rightarrow$ On the L-curve, this corresponds to the intersection with the vertical line at this bound, as shown in Figure.

$\Rightarrow$ A good approximation for the bound is to put $\delta = \sigma\sqrt{n}$, where σ^2 is the variance and n the number of observations.[3] This can be thought

[3] This is strictly valid under the hypothesis of i.i.d. Gaussian noise.

of as the noise level of the data.

$\Rightarrow$ The discrepancy principle is also related to regularization by the truncated singular value decomposition (TSVD), in which case the truncation level implicitly defines the regularization parameter.

- **Cross-validation**, as we explained in ML Lectures, is a way of using the observations themselves to estimate a parameter.

  $\Rightarrow$ We then employ the classical approach of either LOOCV or k-fold cross validation, and choose the value of α that minimizes the RSS (Residual Sum of Squares) of the test sets.

  $\Rightarrow$ In order to reduce the computational cost, a generalized cross validation (GCV) method can be used.

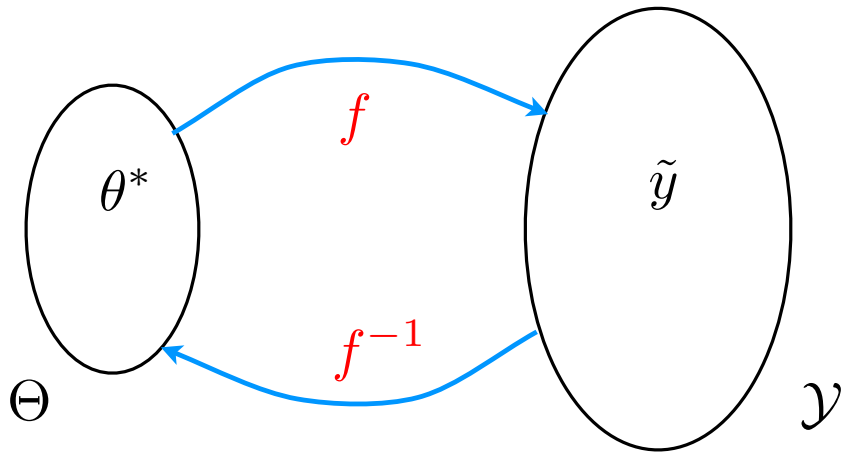
Ill-posedness of inverse problems

All inverse problems are notoriously ill-posed!!!

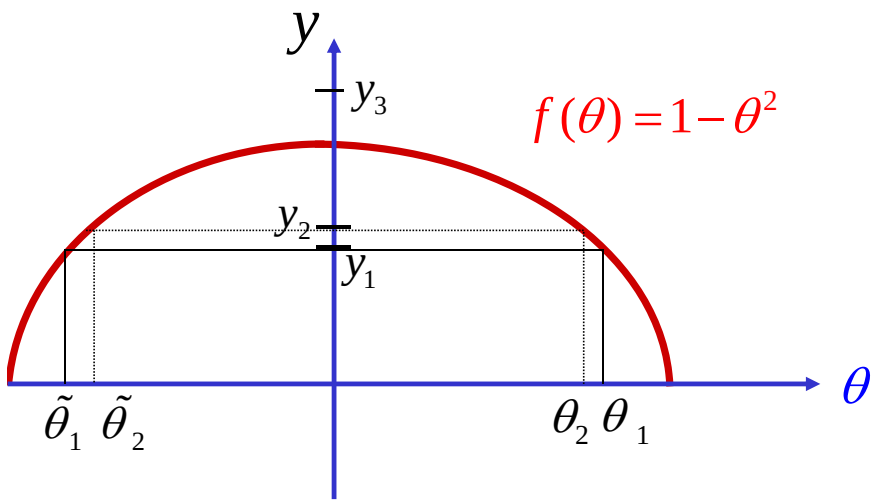
Simplest case: one observation $\tilde{y}$ for $f(\theta)$ and we need to find the pre-image

$$\theta^* = f^{-1}(\tilde{y})$$

for a given $\tilde{y}$.



Simplest case



- ✘ Non-existence: there is no θ_3 such that $f(\theta_3) = y_3$
- ✘ Non-uniqueness: $y_j = f(\theta_j) = f(\widetilde{\theta}_j)$ for $j = 1, 2$.
- ✘ Lack of continuity of inverse map: $|y_1 - y_2|$ small $\not\Rightarrow \left|f^{-1}(y_1) - f^{-1}(y_2)\right| = \left|\theta_1 - \tilde{\theta}_2\right|$ small.

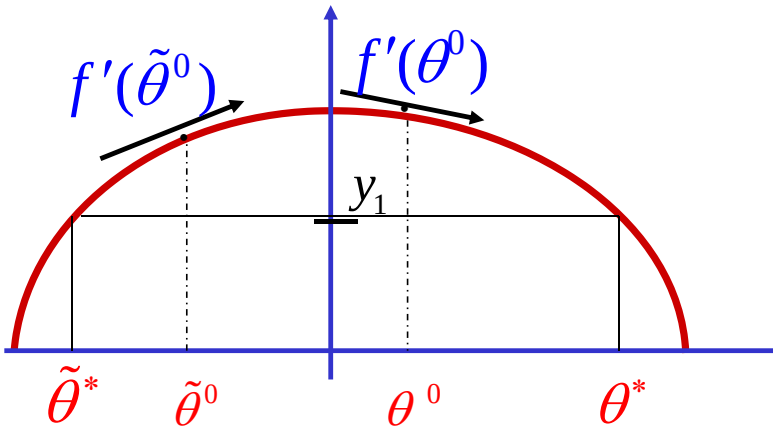
Why is this so important???

Couldn't we just apply a good least squares algorithm (for example) to find the best possible solution?

- Define $J(\theta) = |y_1 - f(\theta)|^2$ for a given y_1
- Apply a standard iterative scheme, such as direct search or gradient-based minimization, to obtain a solution
- Newton's method:

$$\theta^{k+1} = \theta^k - \left[J'(\theta^k)\right]^{-1} J(\theta^k)$$

Newton iterations



- $\theta^{k+1} = \theta^k - \left[J'(\theta^k)\right]^{-1} J(\theta^k), \quad J(\theta) = |y_1 - f(\theta)|^2$
- $J'(\theta) = 2\,(y_1 - f(\theta))\,(-f'(\theta))$

✘ $J'(\theta^0) = 2(-)(--) < 0 \quad \Rightarrow \quad \theta^1 > \theta^0$, etc.

✘ $J'(\tilde{\theta}^0) = 2(-)(-+) > 0 \quad \Rightarrow \quad \tilde{\theta}^1 < \tilde{\theta}^0$, etc.

What went wrong?

✘ This behavior is not the fault of steepest descent algorithms.

✘ It is a manifestation of the inherent ill-posedness of the problem.

✘ How to fix this problem is the subject of much research over the past 50 years!!!

✔ Many remedies (fortunately) exist....

- ✔ explicit and implicit constrained optimizations
- ✔ regularization and penalization
- ✔ machine learning...

Example: Tikhonov regularization

Idea is to replace the ill-posed problem for $J(\theta) = |y_1 - f(\theta)|^2$ by a "nearby" problem for

$$J_\beta(\theta) = |y_1 - f(\theta)|^2 + \beta\,|\theta - \theta_0|^2$$

where β is "suitably chosen" regularization/penalization parameter—see below for details.

✔ When it is done correctly, TR provides convexity and compactness.

✘ Even when done correctly, it *modifies the problem* and new solutions may be far from the original ones.

✘ It is not trivial to regularize correctly or even to know if you have succeeded...

Examples of ill-posedness

We illustrate these with 2 numerical examples.

- ✔ Duffing's equation.
- ✔ Estimation of seismic travel time.

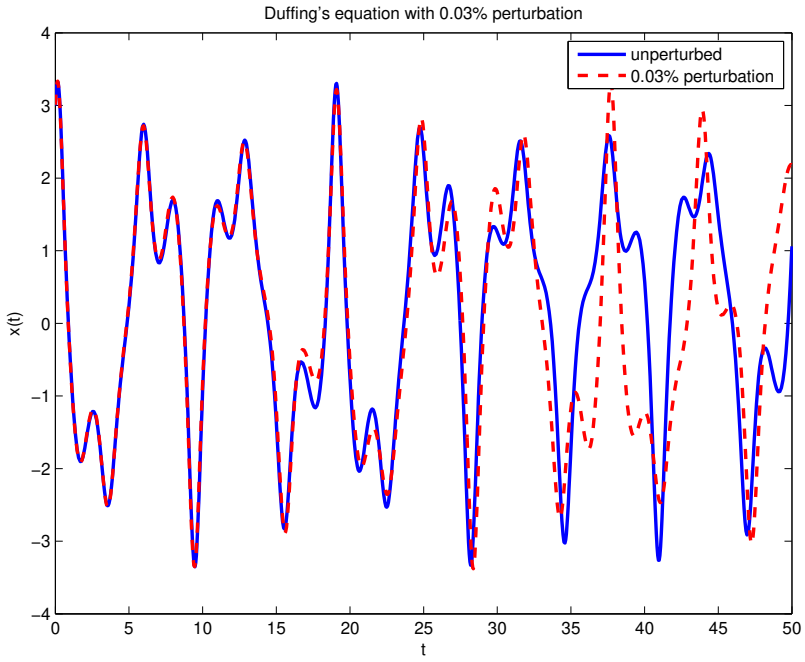
Nonlinearity: Duffing

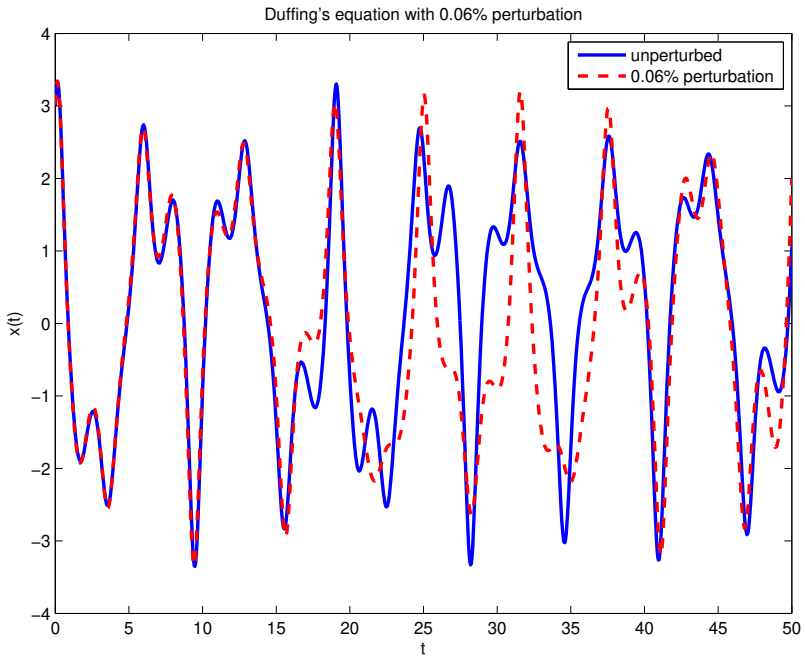
The highly nonlinear Duffing's equation,

$$\ddot{x} + 0.05\dot{x} + x^3 = 7.5\cos t$$

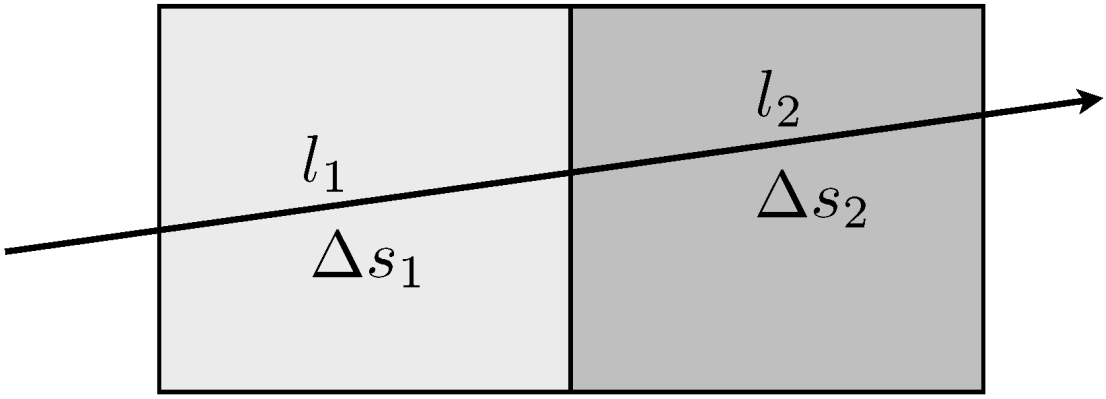
exhibits great sensitivity to the initial conditions. We can observe that two very closely spaced initial states lead to a large discrepancy in the trajectories.

- let $x(0) = 3$ and $\dot{x}(0) = 4$ be the true initial state;
- introduce an error of 0.03% - here we have an accurate forecast until $t = 35$;
- introduce an error of 0.06% - - here we only have an accurate forecast until $t = 20$.


Duffing's equation with 0.03% perturbation
unperturbed
0.03% perturbation
x(t)
t
4
3
2
1
0
−1
−2
−3
−4
0
5
10
15
20
25
30
35
40
45
50


Duffing's equation with 0.06% perturbation
unperturbed
0.06% perturbation
x(t)
t
4
3
2
1
0
−1
−2
−3
−4
0
5
10
15
20
25
30
35
40
45
50

Non uniqueness: seismic travel-time tomography



A signal seismic ray passes through a 2-parameter block model.

- unknowns are the 2 block slownesses (inverse of seismic velocity) $(\Delta s_1, \Delta s_2)$

- data is the observed travel time of the ray, Δt_1

- model is the linearized travel time equation,

$$\Delta t_1 = l_1 \Delta s_1 + l_2 \Delta s_2$$

  where l_j is the length of the ray in the j-th block.

Clearly we have one equation for two unknowns and hence there is no unique solution.

DATA ASSIMILATION

What is data assimilation?

- Simplest view: a method of combining observations with model output.

- Why do we need data assimilation? Why not just use the observations? (cf. Regression)

  $\Rightarrow$ We want to predict the future!
  - $\rightarrow$ For that we need models.
  - $\rightarrow$ But when models are not constrained periodically by reality, they are of little value.
  - $\rightarrow$ Therefore, it is necessary to fit the model state as closely as possible to the observations, before a prediction is made.

Definition 4. Data assimilation (DA) is the approximation of the true state of some physical system at a given time, by combining time-distributed observations with a dynamic model in an optimal way.

Data assimilation methods

There are two major classes of methods:

1. Variational methods where we explicitly minimize a cost function using optimization methods.

2. Statistical methods where we compute the best linear unbiased estimate (BLUE) by algebraic computations using the Kalman filter.

- They provide the same result in the linear case, which is the only context where their optimality can be rigorously proved.

- They both have difficulties in dealing with non-linearities and large problems.

- The error statistics that are required by both, are in general poorly known.

DA: approaches

- DA is an approach for solving a specific class of inverse, or parameter estimation problems, where the parameter we seek is the initial condition.

- Assimilation problems can be approached from many directions (depending on your background/preferences):

  $\Rightarrow$ control theory;
  $\Rightarrow$ variational calculus;
  $\Rightarrow$ statistical estimation theory;
  $\Rightarrow$ probability theory,
  $\Rightarrow$ stochastic differential equations.

- Newer approaches: nudging methods, reduced methods, ensemble methods and hybrid methods that combine variational and statistical approaches, Machine/Deep Learning based approaches.

DA: applications

1. Navigation: important application of the Kalman filter.

2. Remote sensing: satellite data.

3. Geophysics: seismic exploration, geo-prospection, earthquake prediction.

4. Air and noise pollution, source estimation.

5. Weather forecasting.

6. Climatology. Global warming.

7. Epidemiology.

8. Forest fire evolution.

9. Finance.

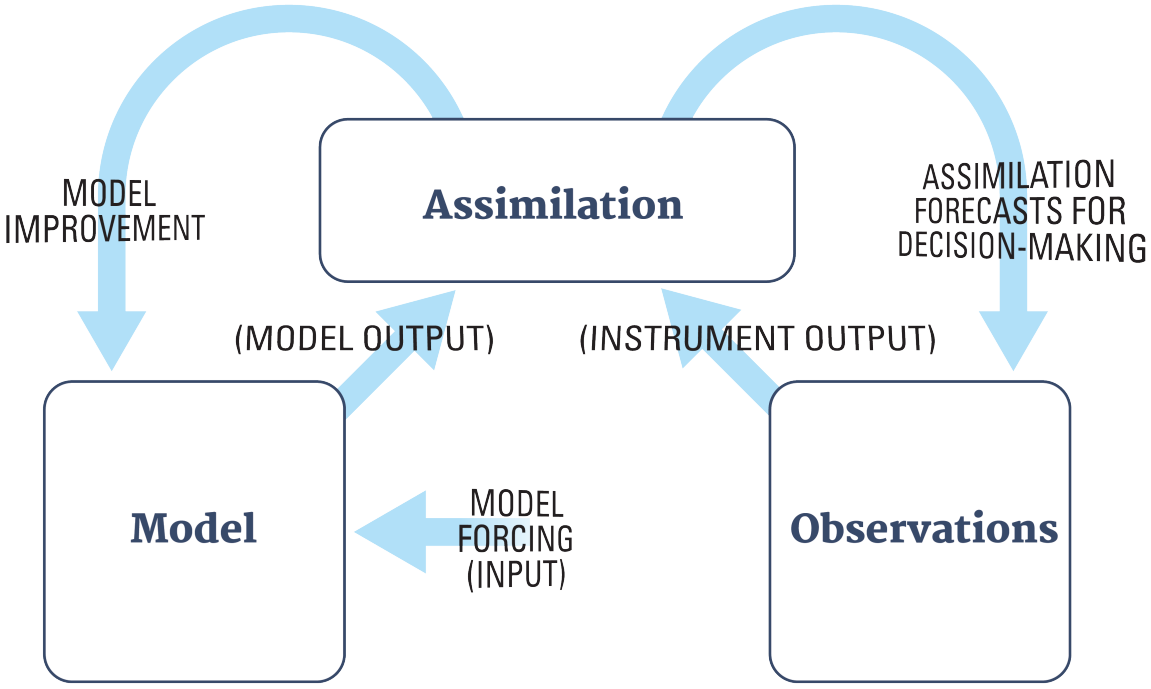
DA & IP: nonlinearity...

The problems of data assimilation (in particular) and inverse problems in general arise from:

1. The nonlinear dynamics of the physical model equations.

2. The nonlinearity of the inverse problem.

Introduction: iterative process...



- Closely related to

  $\Rightarrow$ the inference cycle

  $\Rightarrow$ machine learning...

DA: Definitions and notation

- ✔ Analysis is the process of approximating the true state of a physical system at a given time
- ✔ Analysis is based on:
  - ✔ observational data,
  - ✔ a model of the physical system,
  - ✔ background information on initial and boundary conditions.
- ✔ An analysis that combines time-distributed observations and a dynamic model is called data assimilation.

Standard notation

- A discrete model for the evolution of a physical (atmospheric, oceanic, etc.) system from time t_k to time t_{k+1} is described by a dynamic, state equation

$$\mathbf{x}^f(t_{k+1}) = M\left[\mathbf{x}^f(t_k),\right] \tag{5}$$

  $\Rightarrow$ $\mathbf{x}$ is the model's state vector of dimension n,
  $\Rightarrow$ M is the corresponding dynamics operator (finite difference or finite element discretization).

- The error covariance matrix associated with $\mathbf{x}$ is given by $\mathbf{P}$ since the true state will differ from the simulated state (5) by random or systematic errors.

- Observations, or measurements, at time t_k are defined by

$$\mathbf{y}_k^{\mathrm{o}} = H_k\left[\mathbf{x}^t(t_k)\right] + \varepsilon_k,$$

$\Rightarrow$ H is an observation operator

$\Rightarrow$ ε is a white noise process zero mean and covariance matrix $\mathbf{R}$ (instrument errors and representation errors due to the discretization)

$\Rightarrow$ observation vector $\mathbf{y}_k^{\mathrm{o}} = \mathbf{y}^{\mathrm{o}}(t_k)$ has dimension p_k (usually $p_k \ll n.$)

- Subscripts are used to denote the discrete time index, the corresponding spatial indices or the vector with respect to which an error covariance matrix is defined.

- Superscripts refer to the nature of the vectors/matrices in the data assimilation process:

  $\Rightarrow$ "a" for analysis,

  $\Rightarrow$ "b" for background (or 'initial/first guess'),

  $\Rightarrow$ "f" for forecast,

  $\Rightarrow$ "o" for observation and

  $\Rightarrow$ "t" for the (unknown) true state.

Standard notation - continuous system

- Now let us introduce the continuous system. In fact, continuous time simplifies both the notation and the theoretical analysis of the problem. For a finite-dimensional system of ordinary differential equations, the sate and observation equations become

$$\dot{\mathbf{x}}^{\mathrm{f}} = \mathcal{M}(\mathbf{x}^{\mathrm{f}}, t)$$

and

$$\mathbf{y}^{\mathrm{o}}(t) = \mathcal{H}(\mathbf{x}^{\mathrm{t}}, t) + \boldsymbol{\epsilon},$$

where $\dot{(\,)} = \mathrm{d}/\mathrm{d}t$, $\mathcal{M}$ and $\mathcal{H}$ are nonlinear operators in continuous time for the model and the observation respectively.

- This implies that $\mathbf{x}$, $\mathbf{y}$, and $\boldsymbol{\epsilon}$ are also continuous-in-time functions.

- For PDEs, where there is in addition a dependence on space, attention must be paid to the function spaces, especially when performing variational analysis.

- With a PDE model, the field (state) variable is commonly denoted by $\boldsymbol{u}(\mathbf{x}, t)$, where $\mathbf{x}$ represents the space variables (no longer the state variable as above!), and the model dynamics is now a nonlinear partial differential operator,

$$\mathcal{M} = \mathcal{M}\left[\partial_{\mathbf{x}}^{\alpha}, \boldsymbol{u}(\mathbf{x}, t), \mathbf{x}, t\right]$$

  with $\partial_{\mathbf{x}}^{\alpha}$ denoting the partial derivatives with respect to the space variables of order up to $|\alpha| \leq m$ where m is usually equal to two and in general varies between one and four.

DA: conclusions

Data assimilation requires not only the observations and a background, but also knowledge of:

- ✔ error statistics (background, observation, model, etc.)
- ✔ physics (forecast model, model relating observed to retrieved variables, etc.).

The challenge of data assimilation is in combining our stochastic knowledge with our physical knowledge.

DA Codes

Various open-source repositories and codes are available for both academic and operational data assimilation.

1. DARC: `https://research.reading.ac.uk/met-darc/` from Reading, UK.

2. DAPPER: `https://github.com/nansencenter/DAPPER` from Nansen, Norway.

3. DART: `https://dart.ucar.edu/` from NCAR, US, specialized in ensemble DA.

4. OpenDA: `https://www.openda.org/`.

5. Verdandi: `http://verdandi.sourceforge.net/` from INRIA, France.

6. PyDA: `https://github.com/Shady-Ahmed/PyDA`, a Python implementation for academic use.

7. Filterpy: `https://github.com/rlabbe/filterpy`, dedicated to KF variants.

8. EnKF; `https://enkf.nersc.no/`, the original Ensemble KF from Geir Evensen.

BAYESIAN INFERENCE:
An Introduction

Bayes' Law

- **Independence and Conditioning**:

Definition 5 (Independence). Two events, A and B, are independent if

$$\mathrm{P}(A \cap B) = \mathrm{P}(A)\mathrm{P}(B).$$

Definition 6 (Conditional Probability). The conditional probability of A on B is the probability that A occurs provided (or knowing) that B has occurred, and is given by

$$\mathrm{P}(A \mid B) = \frac{\mathrm{P}(A \cap B)}{\mathrm{P}(B)}.$$

- It follows that if A and B are mutually independent, then

$$\mathrm{P}(A \mid B) = \mathrm{P}(A), \quad \mathrm{P}(B \mid A) = \mathrm{P}(B).$$

- We can now state **Bayes' Theorem**

  $\Rightarrow$ in discrete form, for events

  $\Rightarrow$ in continuous form, for probability distributions

Theorem 1 (Bayes' Theorem for Two Events). *Let A and B be two events in Ω, then*

$$\mathrm{P}(A \mid B) = \frac{\mathrm{P}(B \mid A)\mathrm{P}(A)}{\mathrm{P}(B)}$$

or

$$\mathrm{P}(B \mid A) = \frac{\mathrm{P}(A \mid B)\mathrm{P}(B)}{\mathrm{P}(A)}.$$

- A more general situation is where Ω is subdivided into a partition of events, such as blood types in a human population.

Theorem 2 (Bayes' Theorem for a Partition). *Let $A_1, A_2, \ldots, A_k$ be a partition of Ω with $\mathrm{P}(A_i) > 0$ for each i. If $\mathrm{P}(B) > 0$, then, for each $i = 1, \ldots, k$,*

$$\mathrm{P}(A_i \mid B) = \frac{\mathrm{P}(B \mid A_i)\mathrm{P}(A_i)}{\mathrm{P}(B)},$$

where the total probability

$$\mathrm{P}(B) = \sum_{j=1}^{k} \mathrm{P}(B \mid A_j)\mathrm{P}(A_j).$$

Bayesian Inference

In the most general case, where we want to perform Bayesian inference for the estimation of parameters (an inverse problem!), we simply replace the probabilities by the corresponding density functions.

- Then Bayesian inference is performed in three steps:

1. Choose a probability density $f(\theta)$, called the prior distribution, that expresses our beliefs, or prior experimental or historical knowledge, about a parameter θ before we see any data.
2. Choose a statistical model $f(x \mid \theta)$ that reflects our beliefs about x given θ. Notice that this is expressed as a conditional probability, called the likelihood function, and not as a joint probability function.
3. After observing data $x_1, \ldots, x_n$, update our beliefs and calculate the posterior distribution $f(\theta \mid x_1, \ldots, x_n)$.

- Let us look more closely at the three components of Bayes' Law.

Definition 7 (Prior Distribution)**.** For a given statistical model that depends on a parameter θ, considered as random, the distribution assigned to θ before observing the other random variables of interest is called the *prior distribution*. This is just the marginal distribution of the parameter.

Definition 8. [Posterior Distribution] For a statistical inference problem, with parameter θ and random sample $X_1, \ldots, X_n$, the conditional distribution of θ given $X_1 = x_1, \ldots, x_n = X_n$ is called the *posterior distribution* of θ.

Definition 9 (Likelihood Function)**.** Suppose that $X_1, X_2, \ldots, X_n$ have a joint density function

$$f(X_1, X_2, \ldots, X_n \mid \theta).$$

Given the observations $X_1 = x_1$, $X_2 = x_2$, $\ldots$,

$X_n = x_n$, the likelihood function of θ is

$$L(\theta) = L(\theta \mid x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_n \mid \theta).$$

If the X_i are i.i.d. with density $f(X_i \mid \theta)$, , then the joint density is a product and

$$L(\theta \mid x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} f(x_i \mid \theta).$$

- We point out the following properties of the likelihood:

  $\Rightarrow$ The likelihood is not a probability density function and can take values outside the interval $[0, 1]$.

  $\Rightarrow$ Likelihood is an important concept in both frequentist and Bayesian statistics.

  $\Rightarrow$ Likelihood is a measure of the extent to which a sample provides support for particular values of a parameter in a parametric model—this will be very important when we will deal with parameter estimation, and inverse problems in general.

$\Rightarrow$ The likelihood measures the support (evidence) provided by the data for each possible value of the parameter. This means that if we compute the likelihood function at two points, $\theta = \theta_1$, $\theta = \theta_2$, and find that $L(\theta_1 \mid x) > L(\theta_2 \mid x)$, then the sample observed is more likely to have occurred if $\theta = \theta_1$. We say that θ_1 is a more plausible value for θ than θ_2.

$\Rightarrow$ For i.i.d. random variables, the log-likelihood is usually used, since it reduces the product to a sum.

Bayes' Theorem

- We now formulate the general version of **Bayes' Theorem**.

Theorem 3. *Suppose that n random variables, $X_1, \ldots, X_n$, form a random sample from a distribution with density, or probability function in the case of a discrete distribution, $f(x \mid \theta)$. Suppose also that the unknown parameter, θ, has a prior pdf $f(\theta)$. Then the posterior pdf of θ is*

$$f(\theta \mid x) = \frac{f(x_1 \mid \theta) \cdots f(x_n \mid \theta) f(\theta)}{f_n(x)}, \quad (6)$$

where $f_n(x)$ is the marginal joint pdf of $X_1, \ldots, X_n$.

- In this theorem,

  $\Rightarrow$ the *prior*, $f(\theta)$, represents the credibility of, or belief in the values of the parameters

we seek, without any consideration of the data/observations;

$\Rightarrow$ the *posterior*, $f(\theta \mid x)$, is the credibility of the parameters with the data taken into account;

$\Rightarrow$ $f(x \mid \theta)$, considered as a function of θ, is the *likelihood* function, which is the probability that the data/observation could be generated by the model with a given value of the parameter;

$\Rightarrow$ the denominator, called the *evidence*, $f_n(x)$, is the total probability of the data taken over all the possible parameter values, also called the *marginal likelihood*, or the marginal, and can be considered as a normalization factor;

$\Rightarrow$ the posterior distribution is thus proportional to the product of the likelihood and the prior distribution, or, in applied terms,

$$f(\text{parameter} \mid \text{data}) \propto f(\text{data} \mid \text{parameter})\, f(\text{parameter}).$$

- What can one do with the posterior distribution thus obtained? The answer is a lot of things, in fact a complete quantification of the incertitude of the parameter's estimation is possible. We can compute:

$\Rightarrow$ Point estimates by summarizing the center of the posterior. Typically, these are the posterior mean or the posterior mode.

$\Rightarrow$ Interval estimates for a given level α—see below.

$\Rightarrow$ Estimates of the probability of an event, such as $\mathrm{P}(a < \theta < b)$ or $P(\theta > b)$.

$\Rightarrow$ Posterior quantiles.

Recap of Bayesian Setting

We seek the posterior density of the parameters θ

$$\pi^y(\theta) := p(\theta|y) \propto \mathcal{L}(y, f(\theta))p(\theta).$$

Ingredients:

- Parameters $\theta \in \mathbb{R}^d$, data $y \in \mathbb{R}^n$.
- Prior density $p(\theta)\colon \mathbb{R}^d \to \mathbb{R}^+$.
- Forward model $g\colon \mathbb{R}^d \to \mathbb{R}^n$.
  - $\Rightarrow$ Can be a black-box function.
  - $\Rightarrow$ Each evaluation is expensive.
- Likelihood function $\mathcal{L}\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$.
  - $\Rightarrow$ $\mathcal{L}(y, g(\theta)) = p(y|\theta)$
  - $\Rightarrow$ Each evaluation requires an evaluation of g.