Introduction to Uncertainty Quantification for Bayesian Inverse Problems

Babak Maboudi - day 1 - Jyväskylä summer school 2025

What are inverse problems?

- When we want to understand hidden causes from indirect measurements.
- It is best understood by examples!

Examples of Inverse Problems

X-ray Computed Tomography (CT) or CAT scan

Anna Bertha Ludwig's hand

X-ray by Wilhelm Röntgen

1895

First X-ray image in space

2025

Examples of Inverse Problems

X-ray Computed Tomography (CT) or CAT scan

X-ray radiography

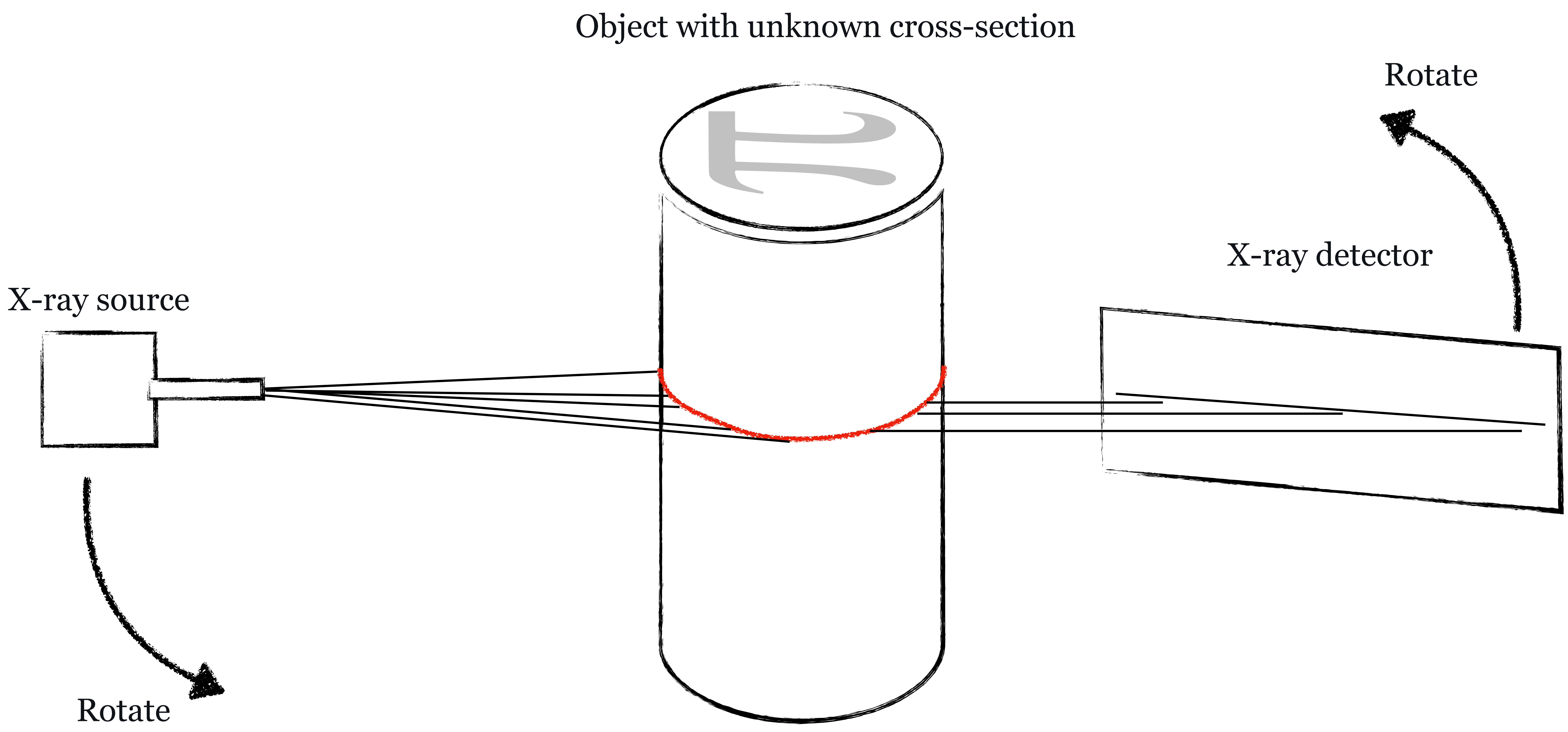
One X-ray image

X-ray computed tomography (CT)

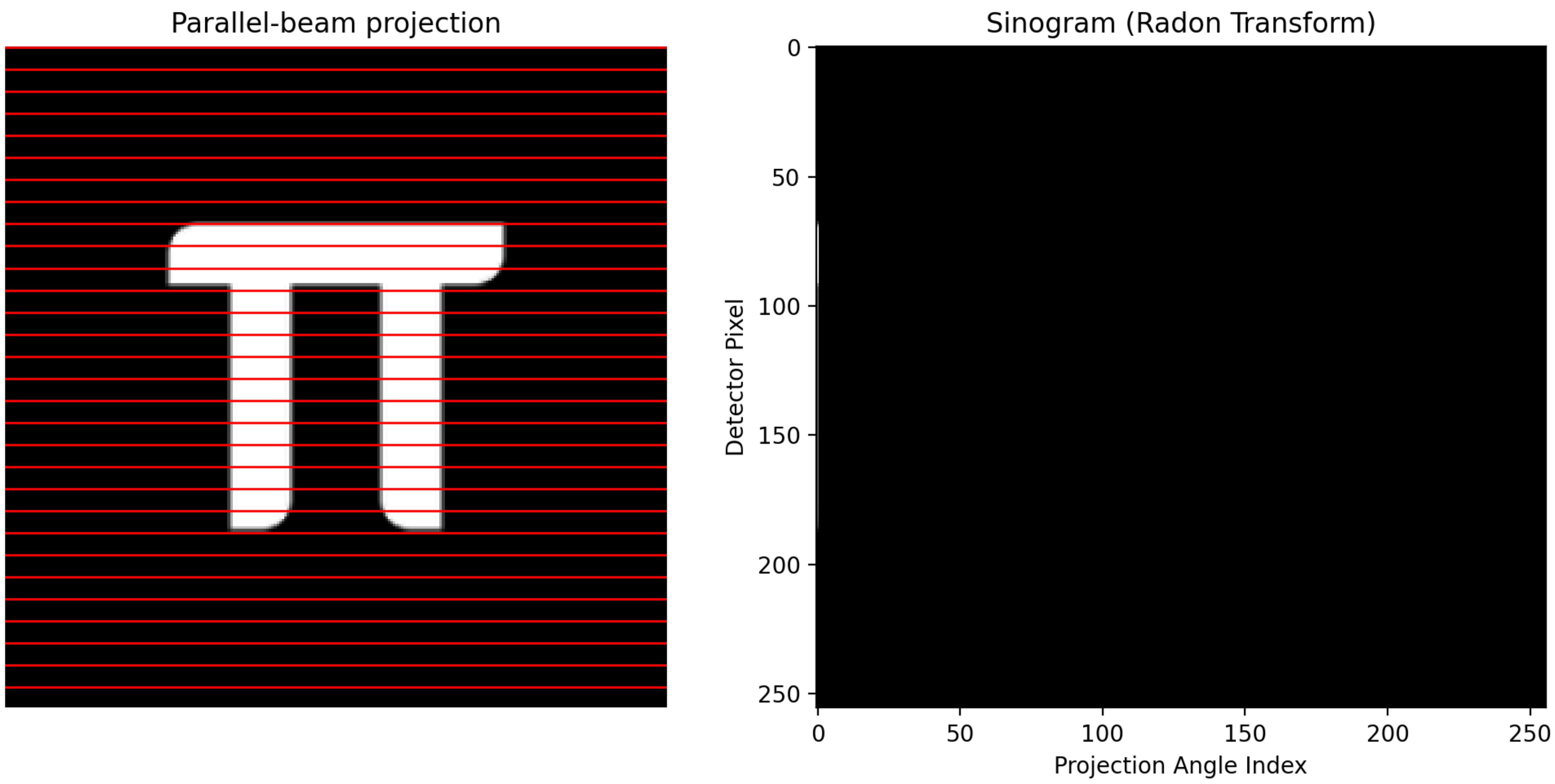
Sequence of X-ray images

Examples of Inverse Problems

X-ray CT, a 2D example

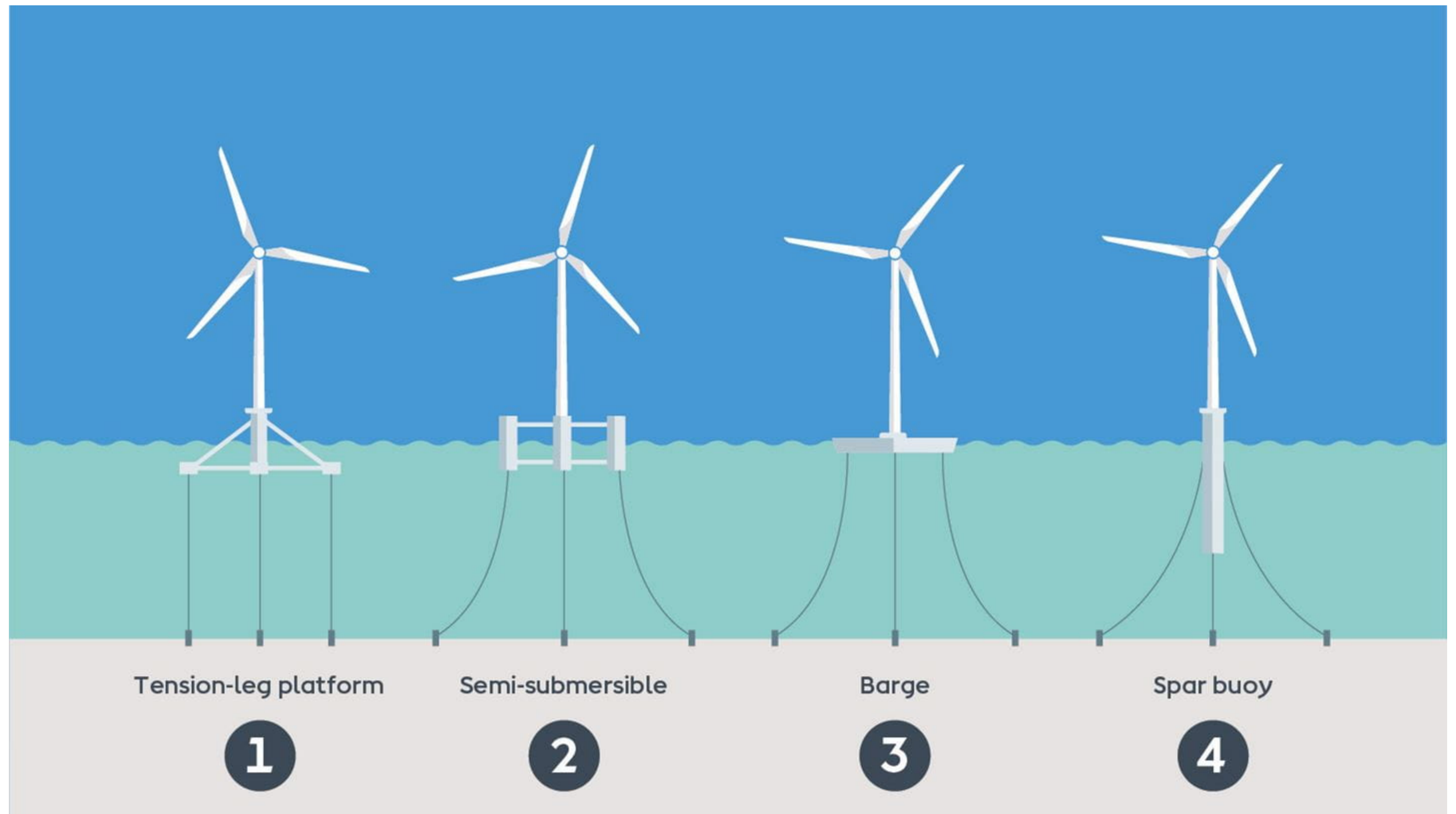


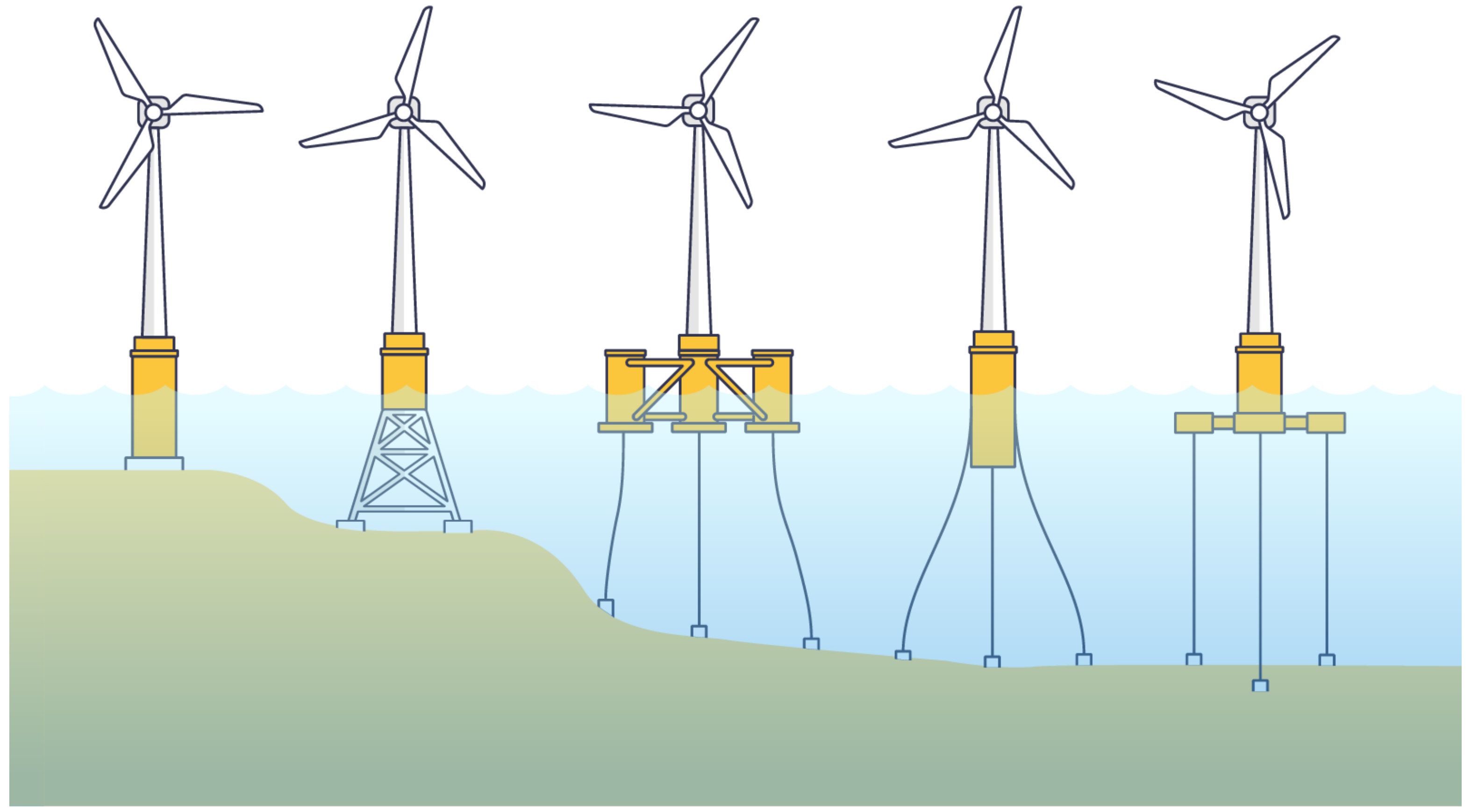
Examples of Inverse Problems
X-ray CT, a 2D example



Examples of Inverse Problems

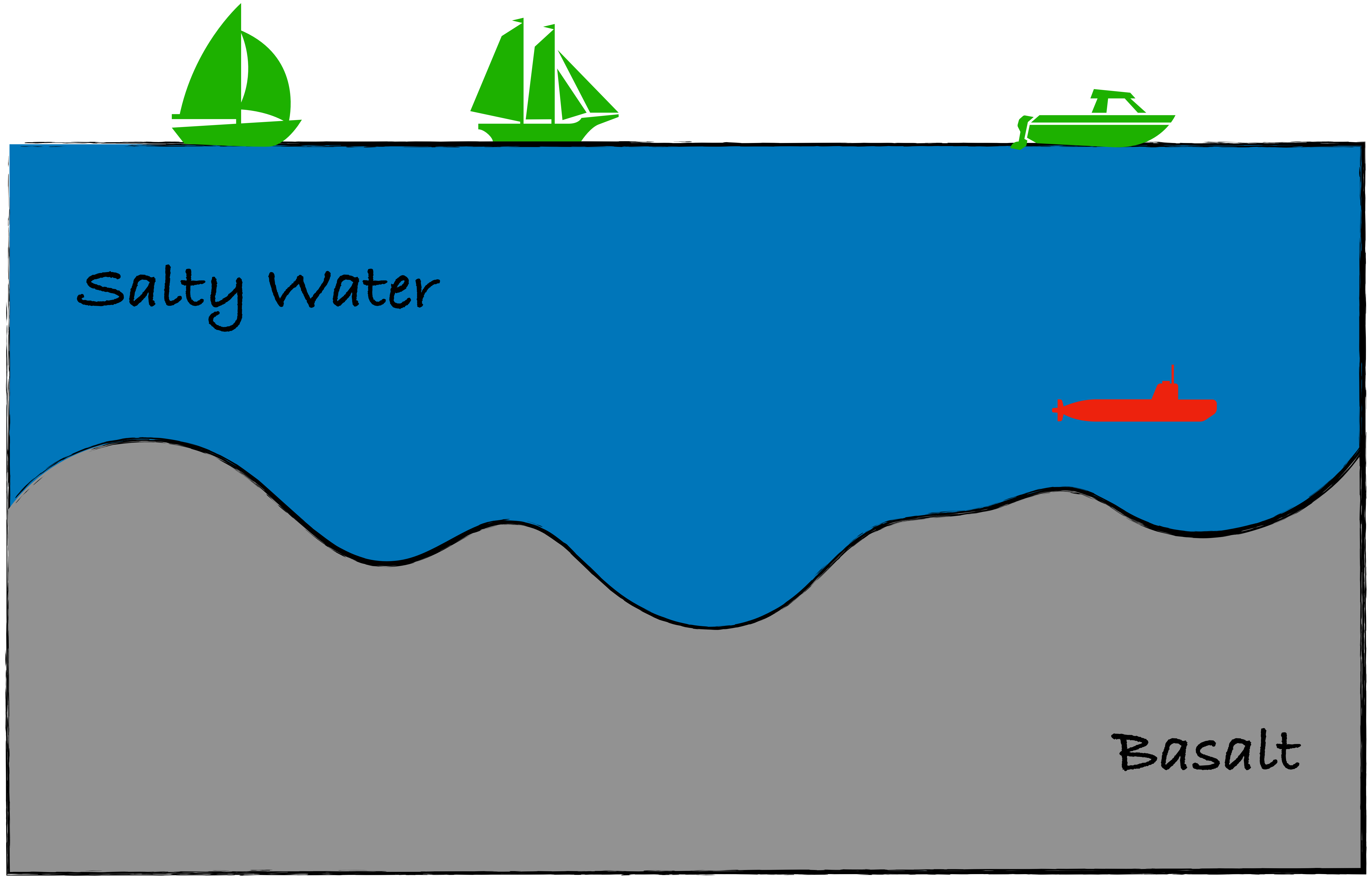
Ocean Floor Detection/Exploration





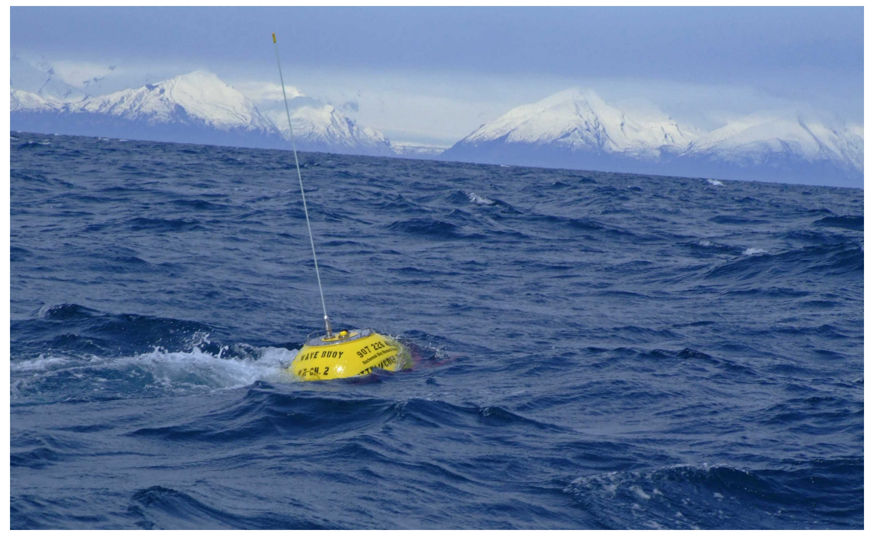
Examples of Inverse Problems

Ocean Floor Detection/Exploration

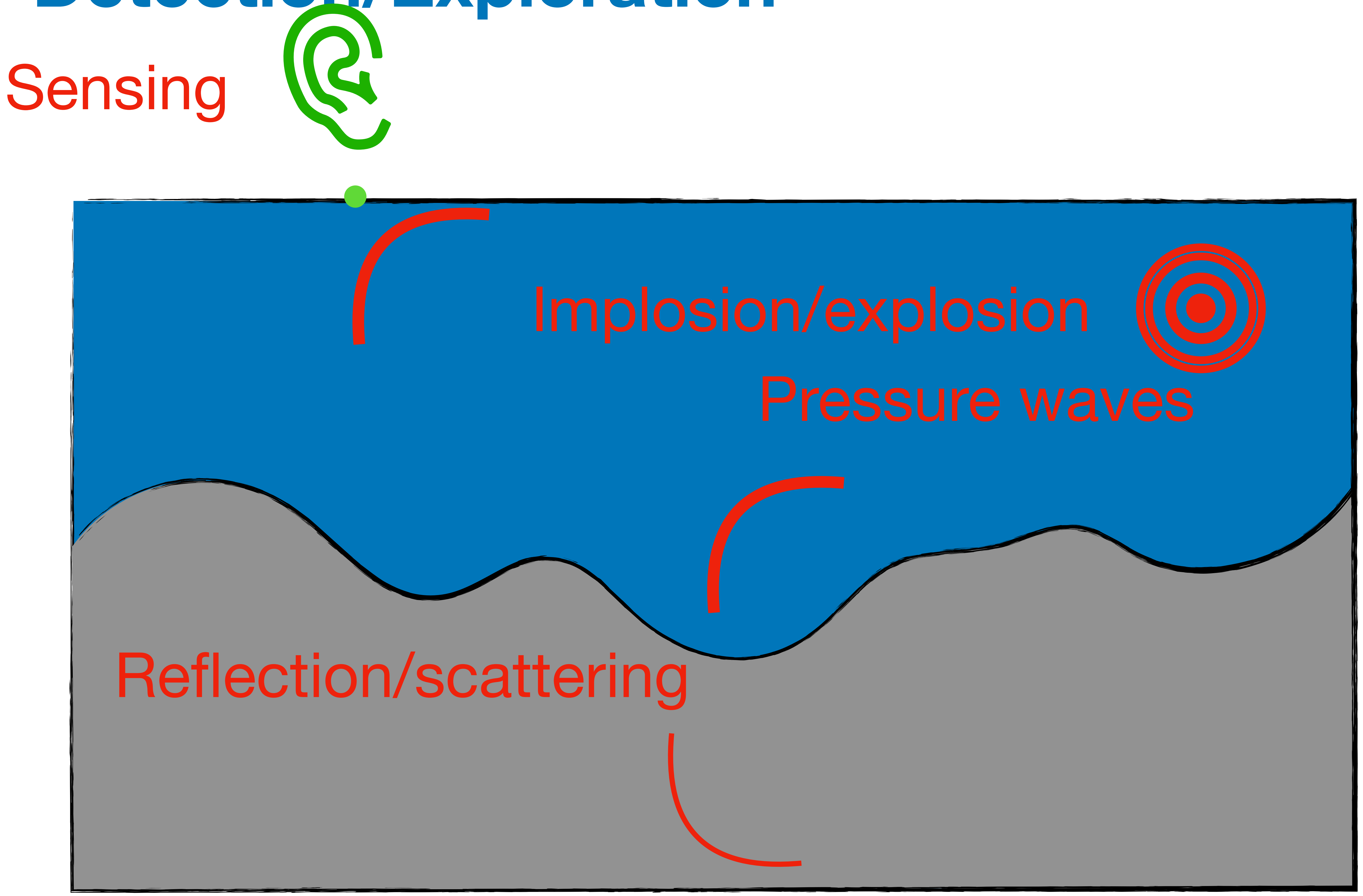


Examples of Inverse Problems

Ocean Floor Detection/Exploration

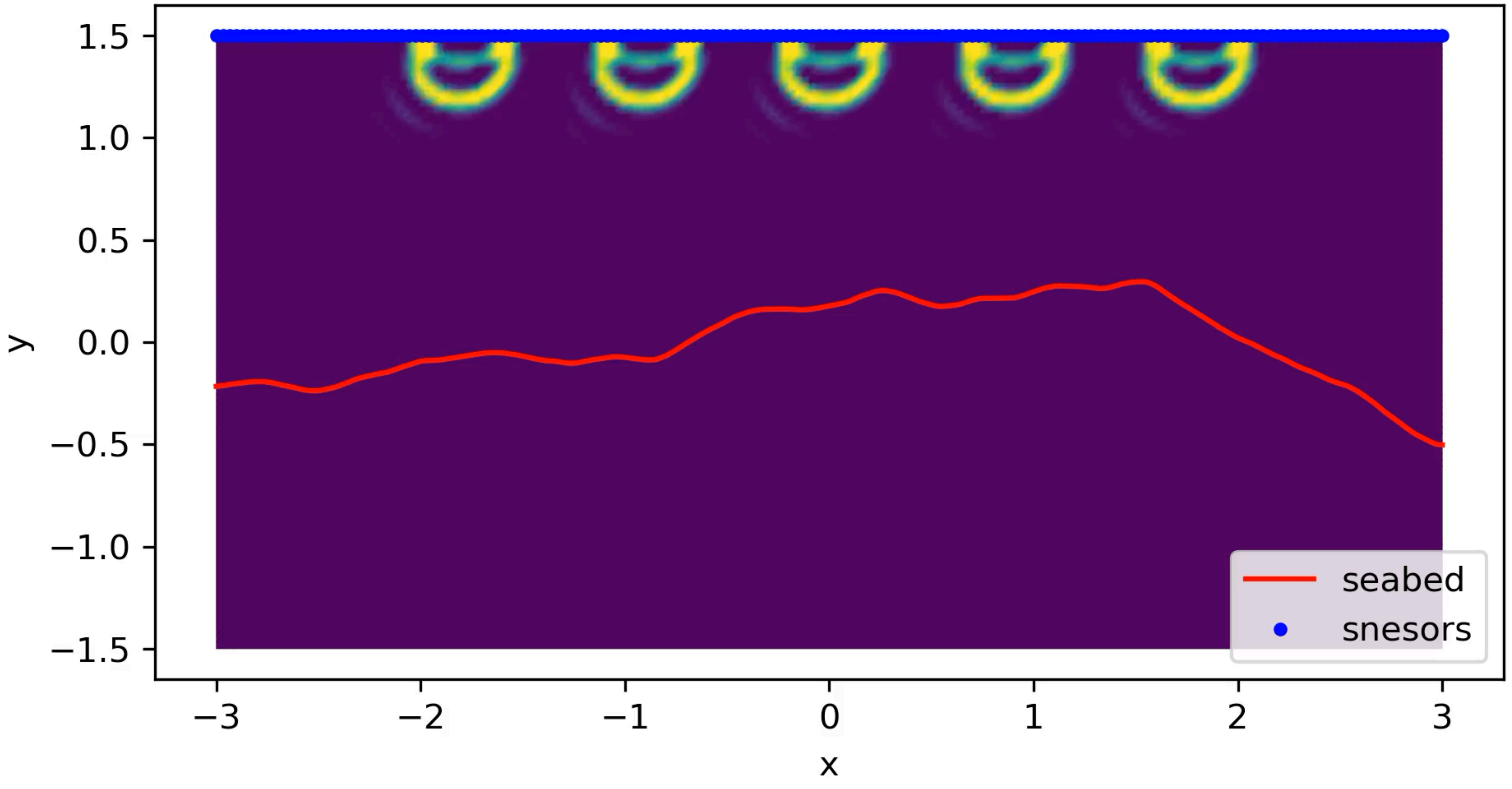


Wave Buoys



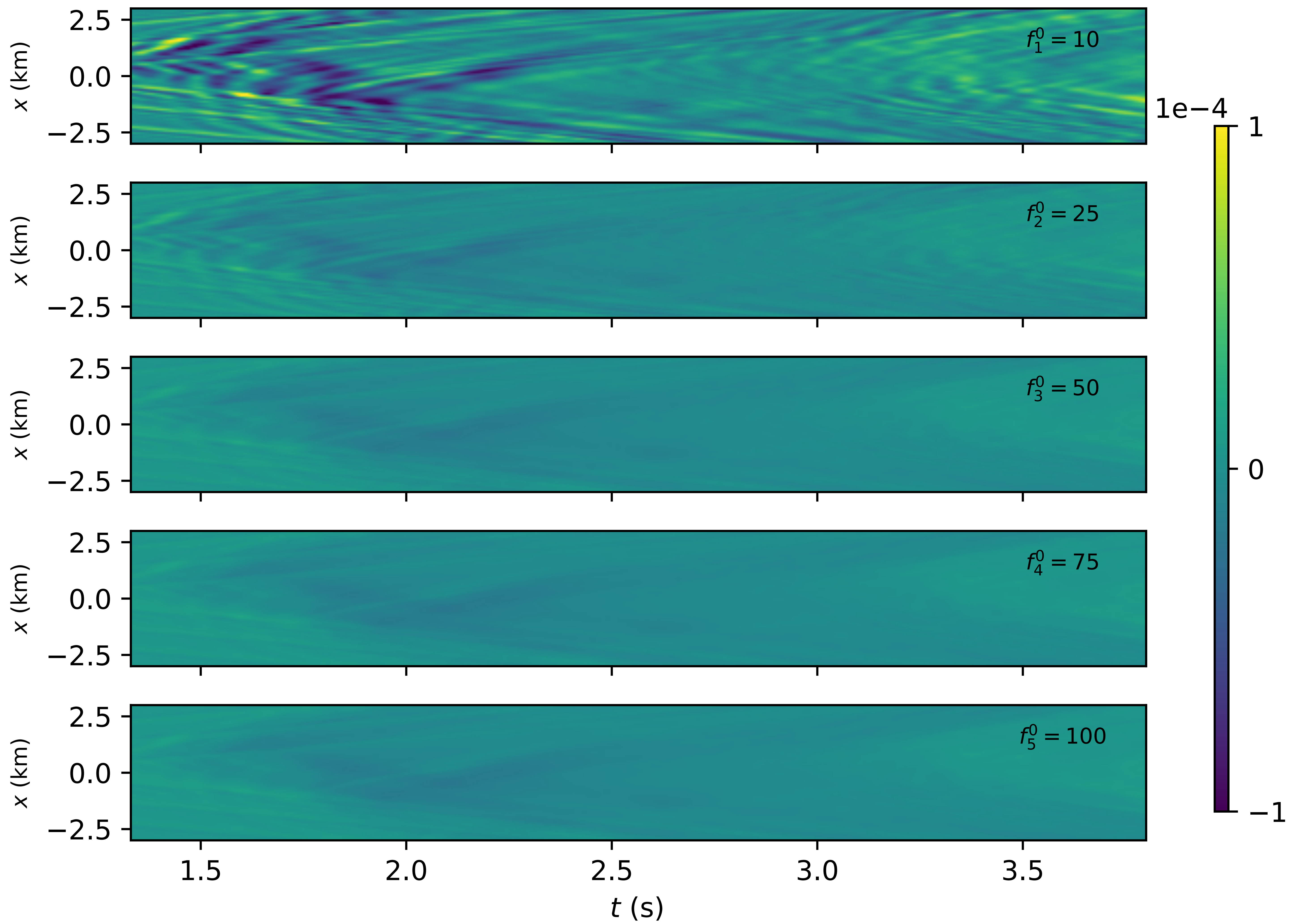
Examples of Inverse Problems

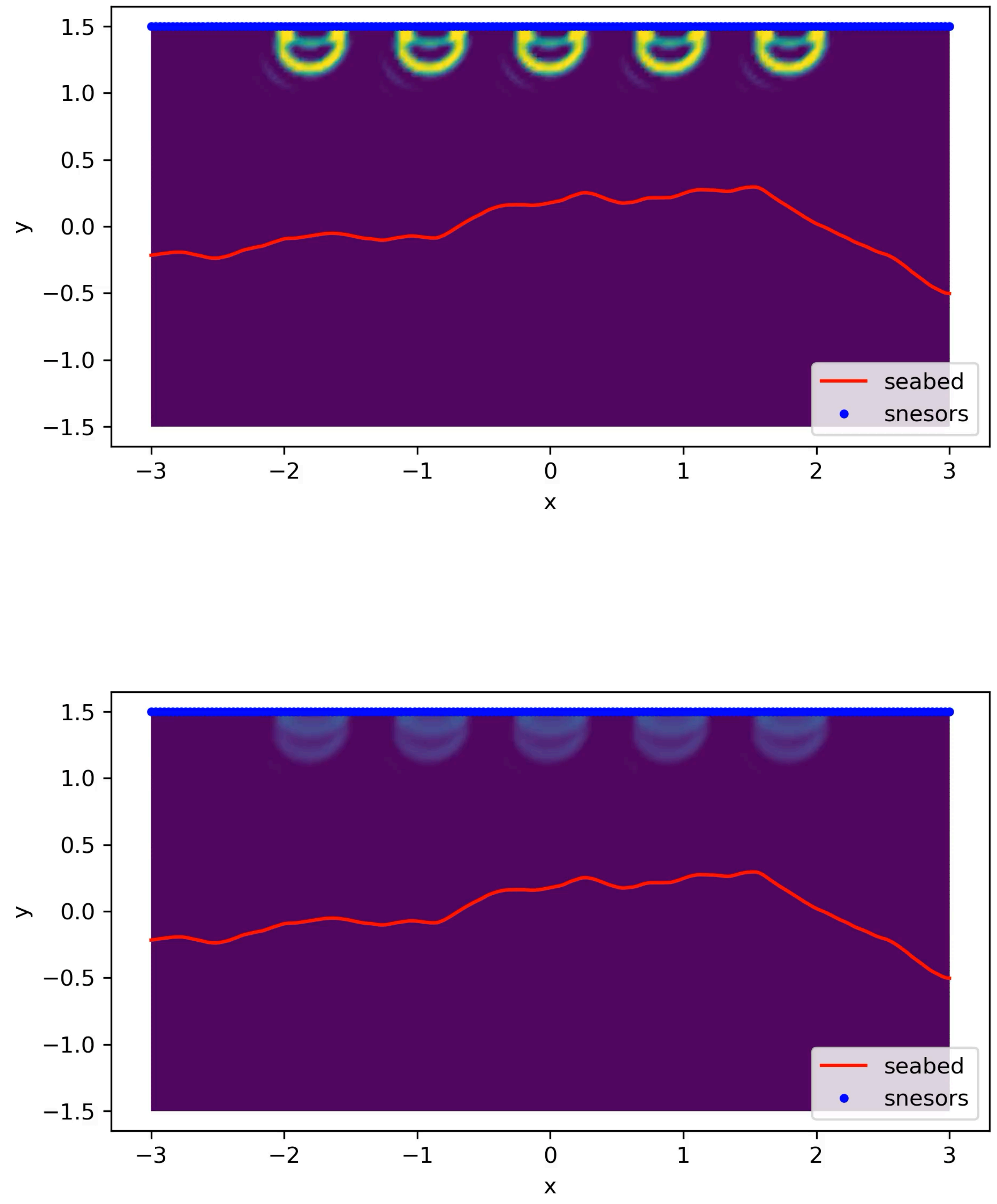
Ocean Floor Detection/Exploration



Examples of Inverse Problems

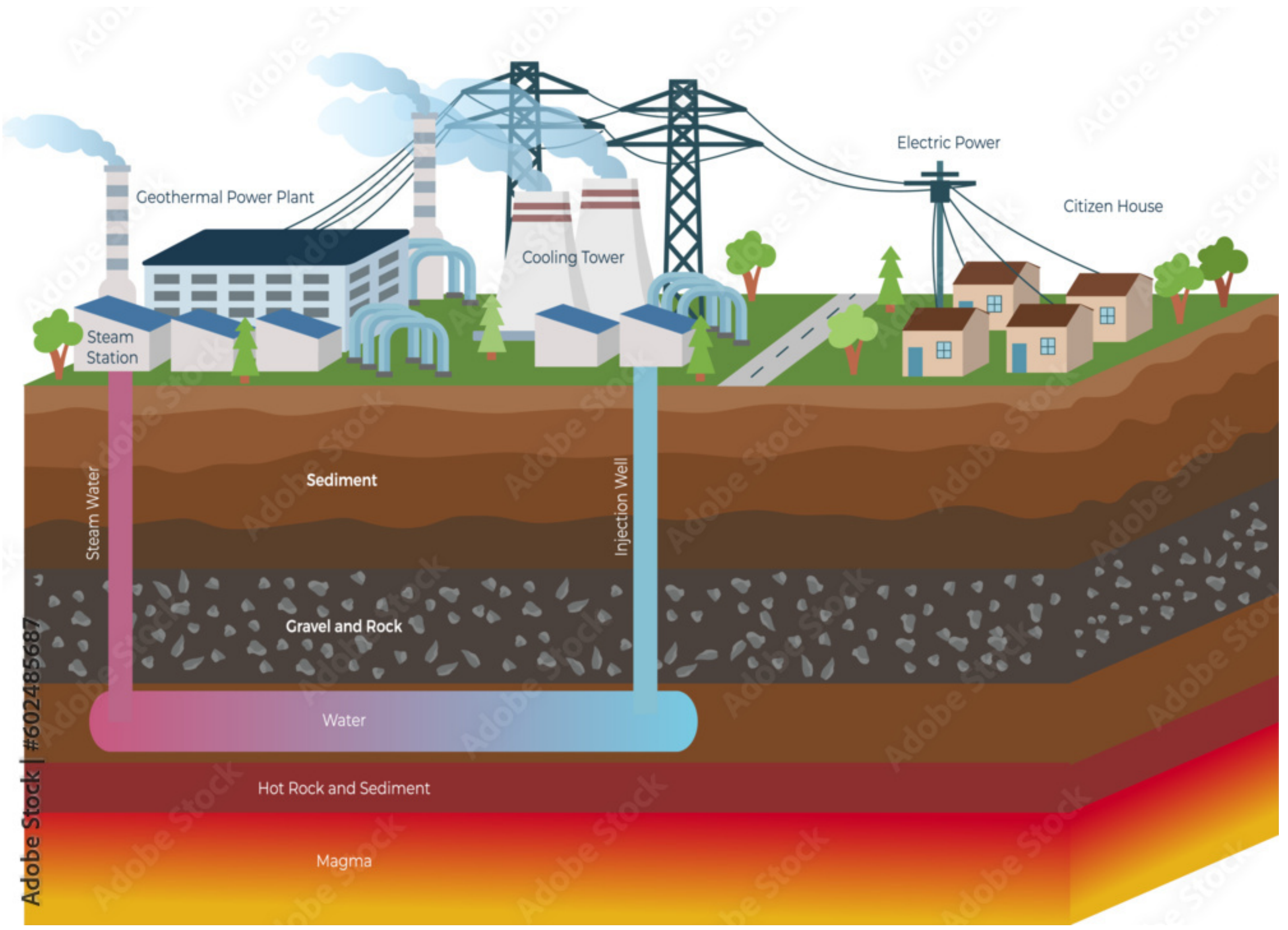
Ocean Floor Detection/Exploration





Examples of Inverse Problems

Geo-thermal Power stations (Motivation for Course Project)



Examples of Inverse Problems

- A typical formulation of an inverse problem
$$\mathbf{y} = F(\mathbf{x}) + \varepsilon$$
  - Here $\mathbf{x}$ is a mathematical representation of the unknown. It can be a parameter in $\mathbb{R}^+$, a vector of parameters in $\mathbb{R}^d$, a function in a function space, …
  - F is called the forward operator and is the (physical or approximate) process that creates noise-free measurements from a known $\mathbf{x}$.
  - ε is the noise in the measurement instruments.
  - $\mathbf{y}$ is the raw measurement.
- Discuss in groups what are $\mathbf{x}, \mathbf{y}, \varepsilon$ and F in the examples we discussed previously.

Uncertainty in Inverse Problems

Exercise 1

- In each of these examples investigate what are sources of uncertainty?
  - X-ray CT
  - Ocean floor detection with waves
  - Geo-thermal power station (both for exploration and monitoring)
- What are the consequences of uncertainties in each case?

Uncertainty in Inverse Problems

Exercise 2

- Choose an inverse problem of your choice. Investigate what are the sources of uncertainties.
- What are the consequences of uncertainty in your example?

Teaching Objectives of This Course

By the end of this course:

- You can formulate an inverse problem in a statistical setting.
- You can incorporate prior knowledge into the statistical problem in terms of a prior distribution.
- You can use the Bayes' theorem to formulate the solution to an inverse problem as the posterior distribution.
- You can write an algorithm and a Python code that can explore the posterior distribution with a sampling method.
- You can interpret samples of the posterior distribution as level of uncertainty (quantified uncertainty).
- You can perform uncertainty quantification for both linear and non-linear inverse problems.
- You will deliver outputs through teamwork.

What to expect in this course?

- You must work in groups of 3.
- You need to write codes in Python. You need numpy, matplotlib and scipy.
- We will have an active learning teaching method (appose to in-active students).
- Every session will have “homework” which we will do during the lectures.
- Ideally all homework will be finished in class.
- You will hand-in selected homework as your course report in groups of 3. You will be evaluated based on the final report.
- Advice: Start your Latex report as we go through the week.

Course Overview

- Day 1: Introduction to Inverse Problems, Uncertainty Quantification and revision on basic probability theory.
- Day 2: Markov chain and acceptance/rejection sampling.
- Day 3: The famous random-walk Metroplis-Hastings algorithm.
- Day 4: Choices of priors for Bayesian inverse problems.
- Day 5: Continuous and differentiable priors for Bayesian inverse problems.

Bayesian vs. Frequentist Debate

Source for the coin analogy:

Cassie Kozyrkov

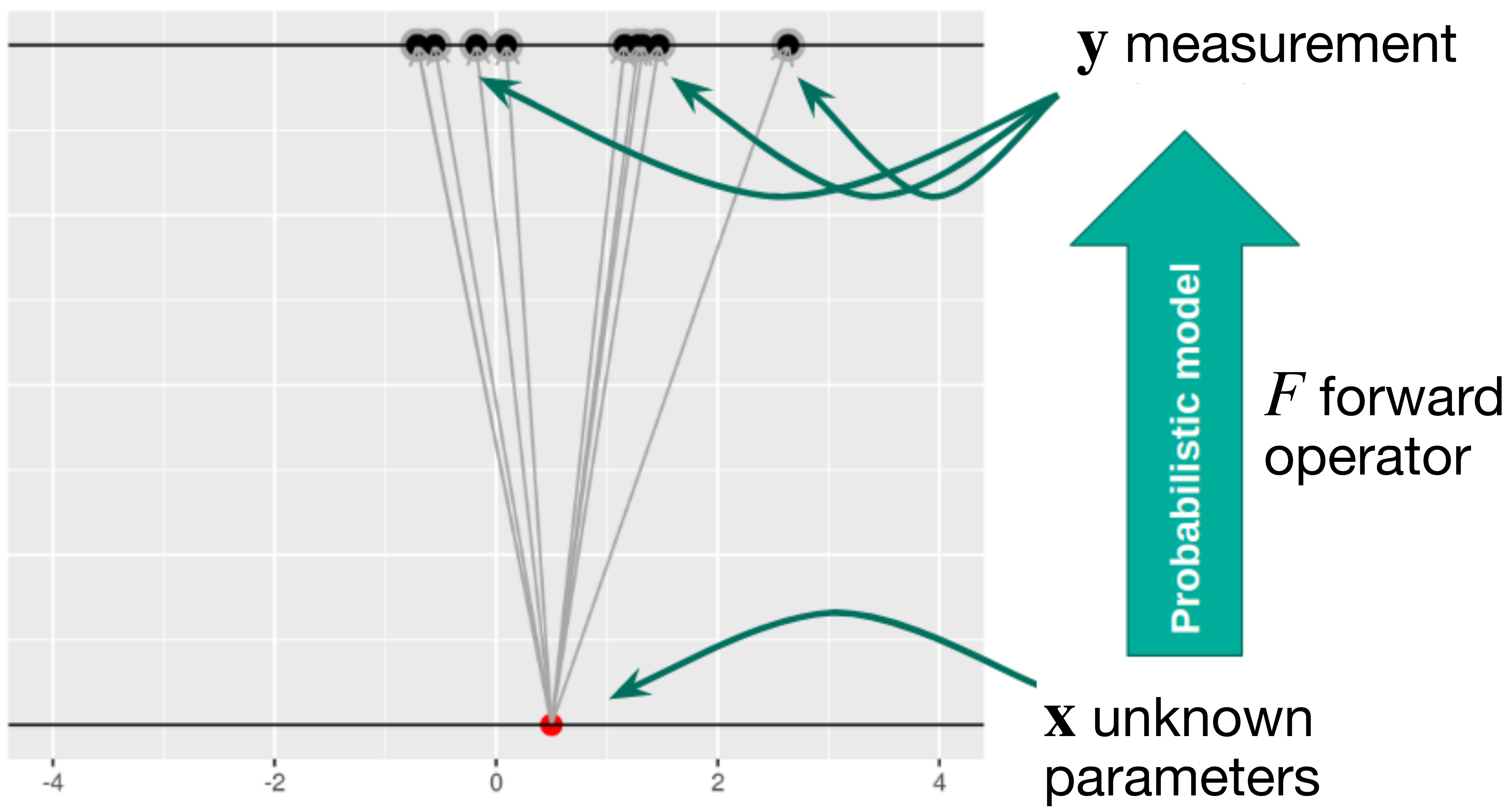
Are you Bayesian or Frequentist?

https://www.youtube.com/watch?v=GEFxFVESQXc&t=2s

Inverse Problems

A generic view

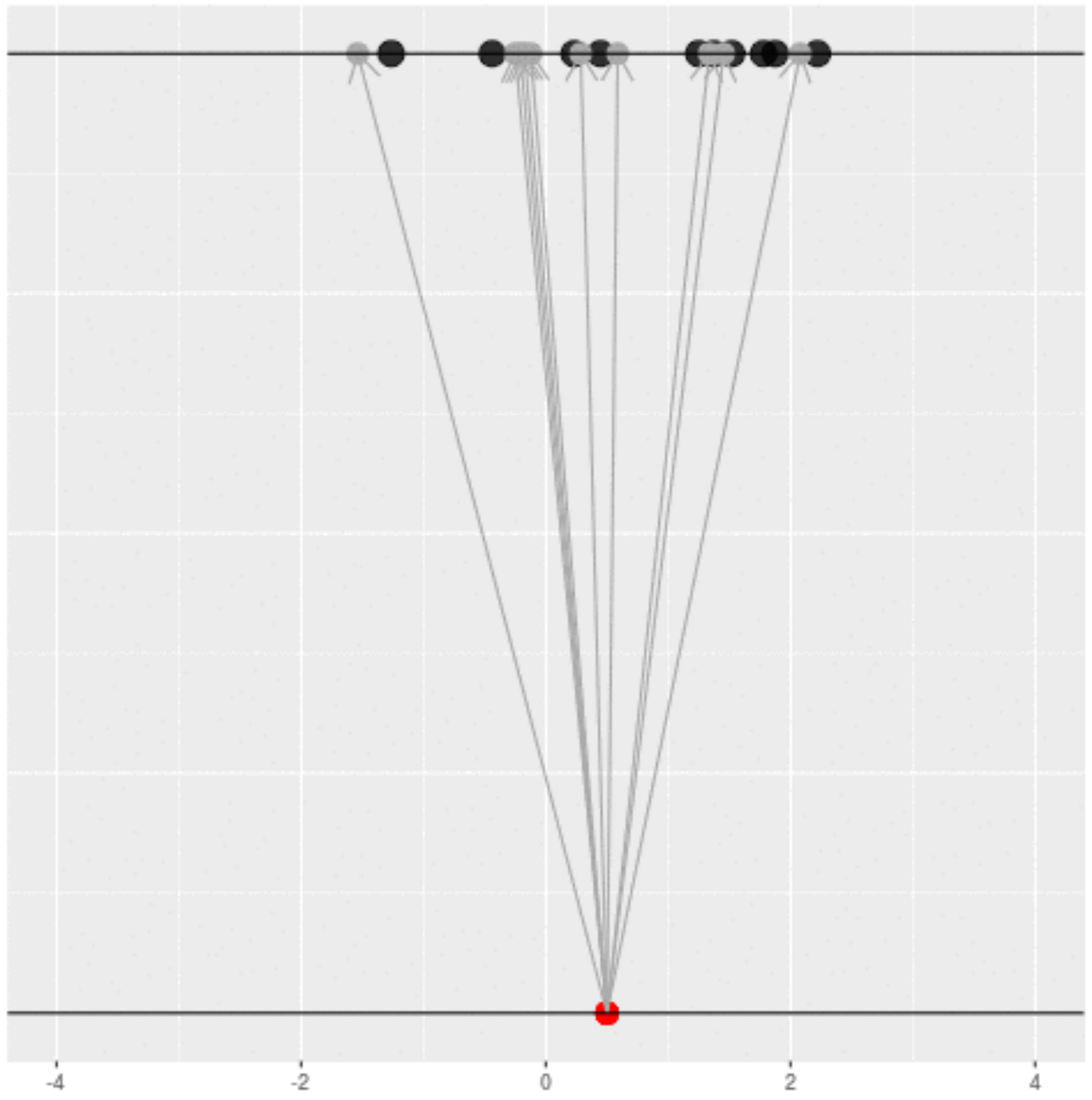
- Recall
  - $\mathbf{x}$ is the unknown
  - $\mathbf{y}$ is the measurement
  - F is the forward operator



Inverse Problems

A frequentist approach

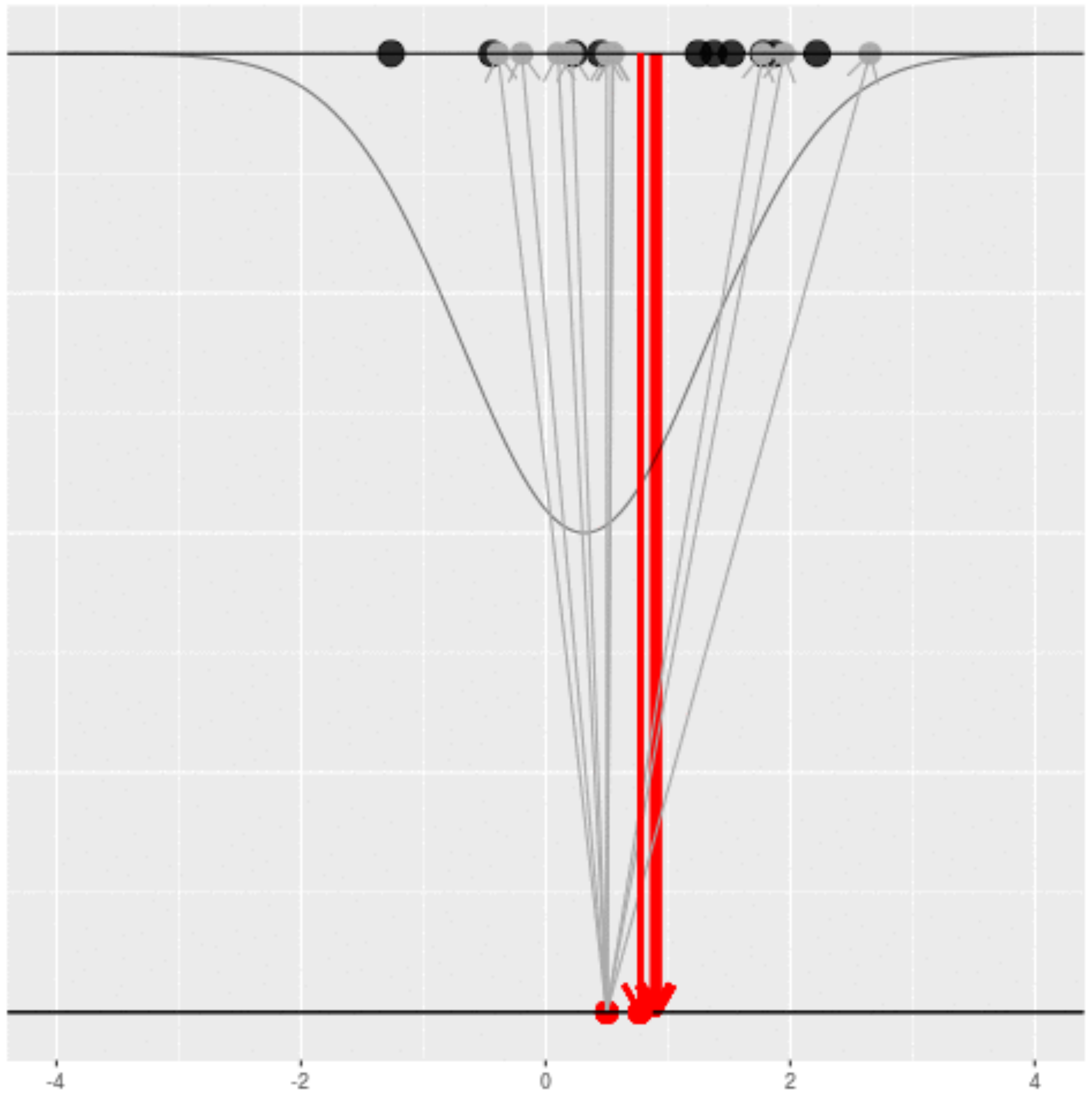
- There is a true parameter $\mathbf{x}$
- Due to noise, every measurement is different, i.e., same parameter can result in different measurement data.



Inverse Problems

A frequentist uncertainty quantification

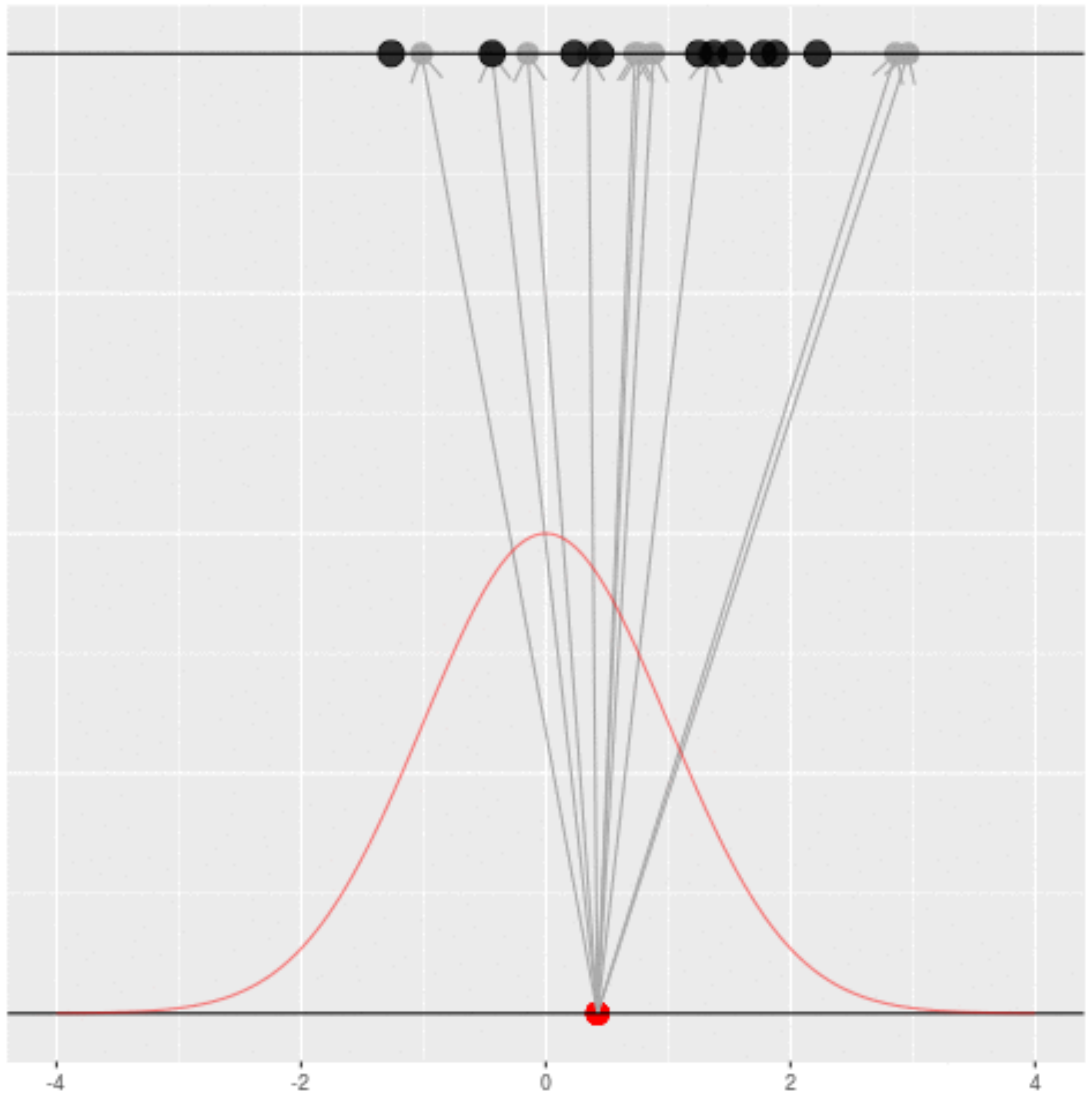
- Pick an estimate strategy.
- The range these estimates create, if we repeated the experiment, quantifies the uncertainty.



Inverse Problems

A Bayesian approach

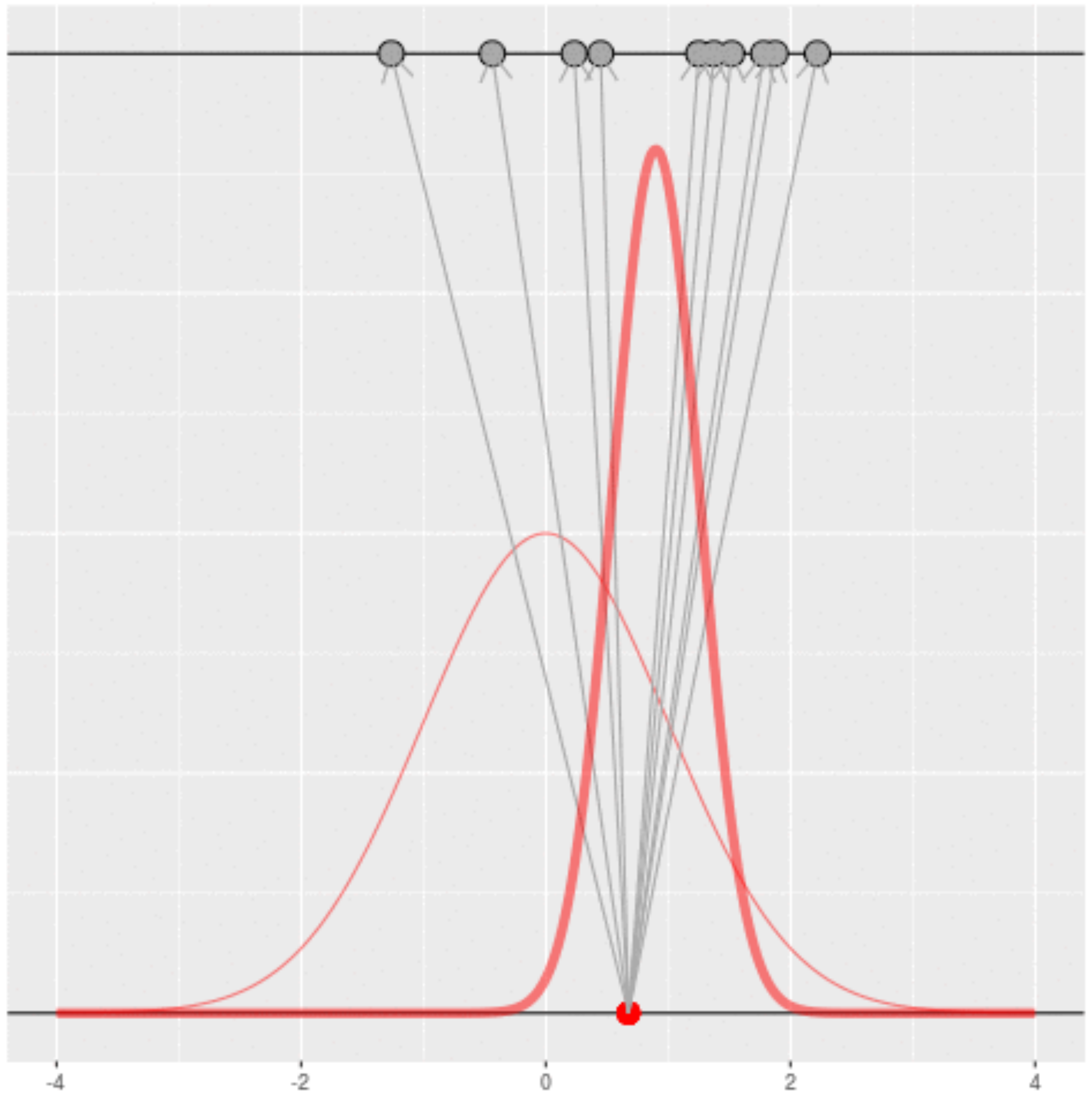
- There is a no true parameter **x**
- We have an opinion, or a prior belief on what the value of **x** is.
- Different parameters can create different measurement data.



Inverse Problems

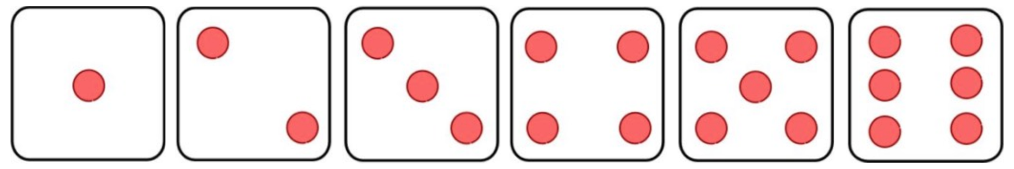
A Bayesian uncertainty quantification

- We delete the parameters that does not match with data.
- What remains is called the posterior.
- The uncertainty is then interpreted as the shape of the posterior.



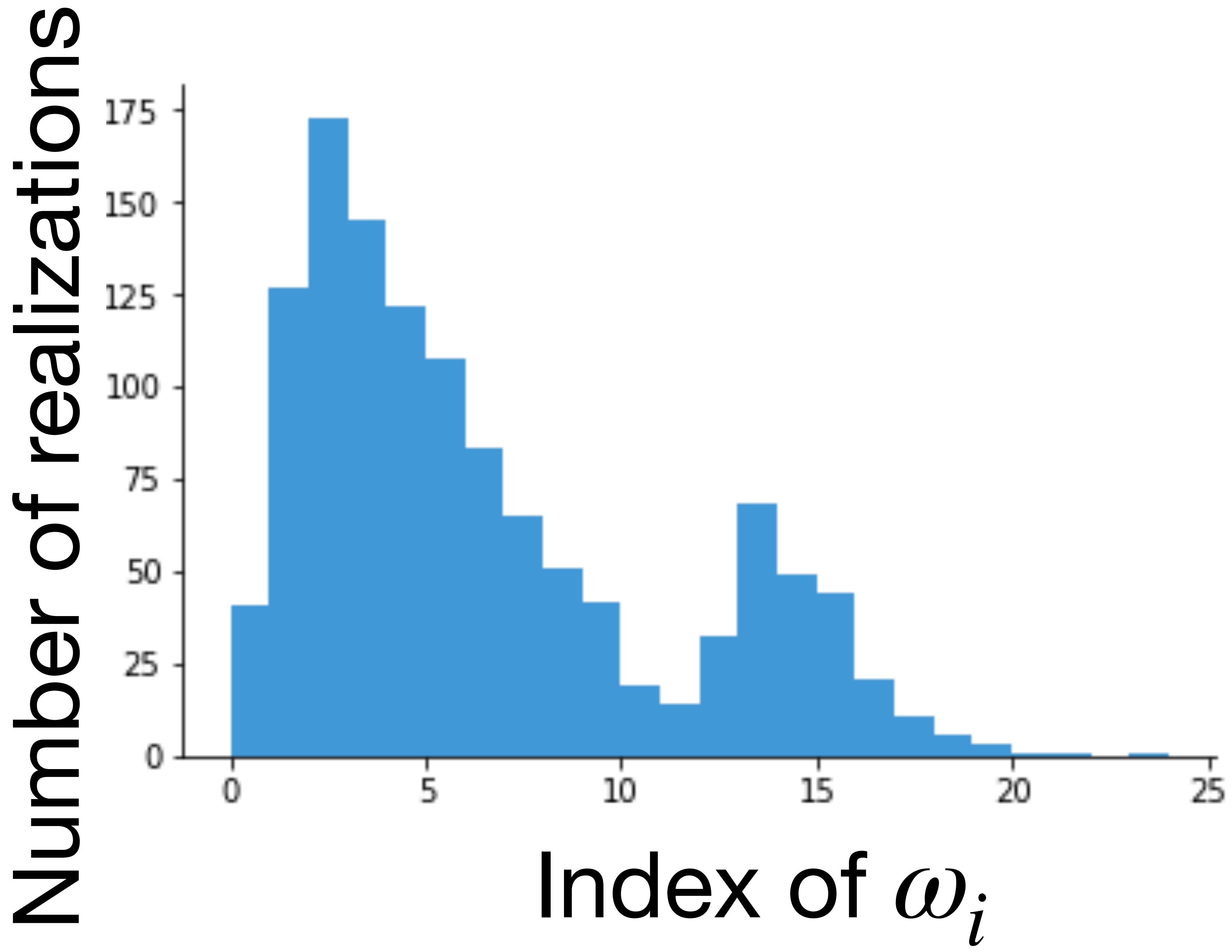
Recap

Random variables

- We model random events that takes values in a set Ω as Ω-valued random variables and write them with capital letters, e.g., $X, Y, \ldots$
  - Let $A = \{$$\}$, then a dice roll is an A-valued random variable.
  - Let $B = \{$$\}$, then a flip of a (Finnish) 1 Euro coin is a B-valued random variables.
  - Let $C = [0{,}3]$ meters, then measuring hight of people is a C-valued random variable.
  - Let D be the set of continuous and finite paths in 3D space. Then the path of a mosquito in the air is a D-valued random variable.

Recap
Histogram

- Let X be an Ω-valued random variables. Then under repeated realizations of X we record the outcomes in a sequence $\omega_1, \omega_2, \omega_1, \omega_4 \ldots$ A histogram is then a frequency plot.



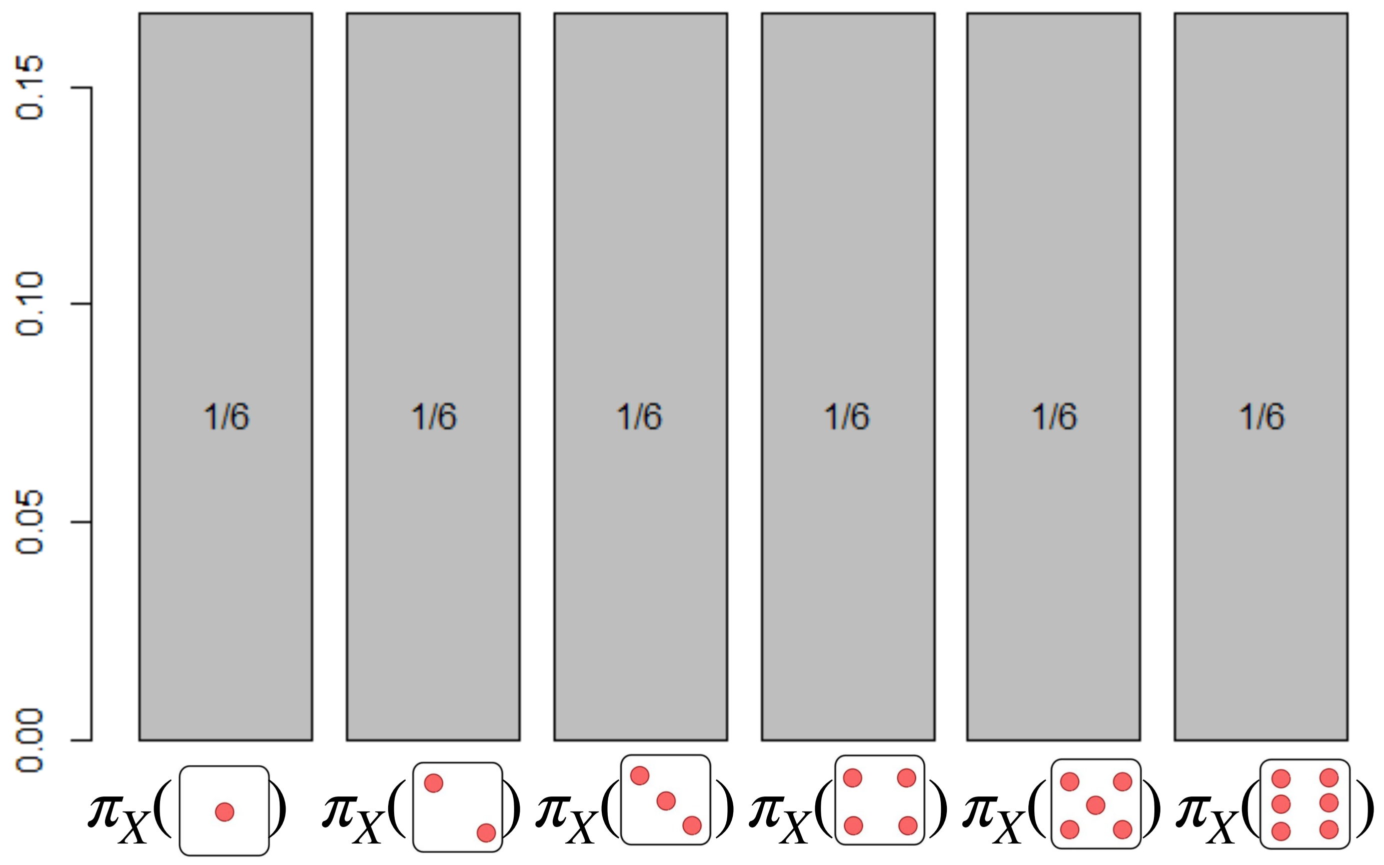
Recap
Density function

- The law, under which, a random variable behaves is called a distribution of a random variable.
  - For X being a fair dice roll, the distribution assigns equal probability of seeing each number.
- We can (sometimes) express a distribution using a density function. Which contains the “probability” of each event.
  - We represent a density as $\pi_X(\mathbf{x})$: This means the probability of the Ω-valued random variable X for the outcome $\mathbf{x} \in \Omega$.

Recap

Examples of Density function

- Let X be a random variable of a fair dice roll. Then we can represent its density function as



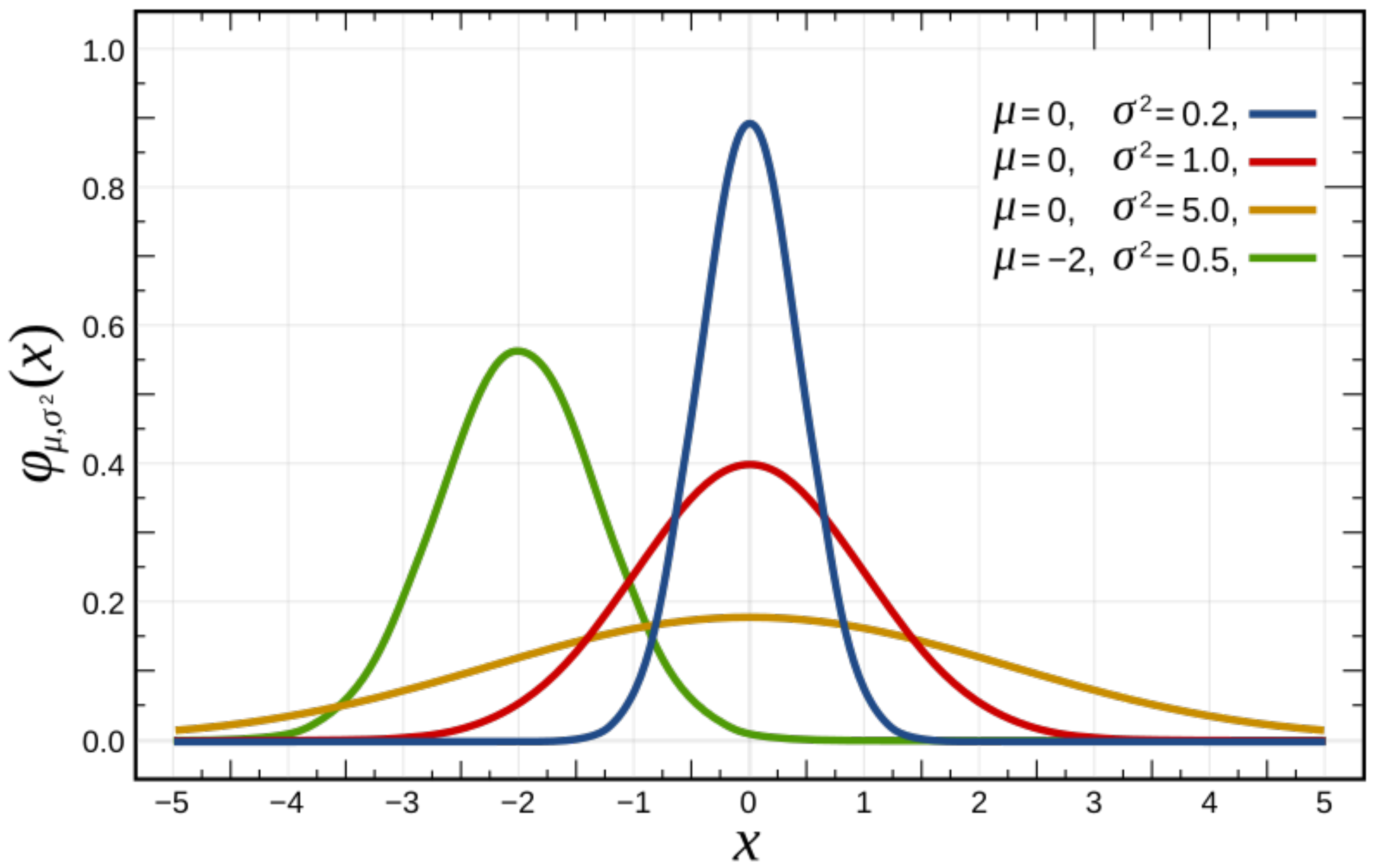
Recap

Examples of Density function

- Let X be an $\mathbb{R}$-valued random variable. Then the wrong way to think about its density function is to think of a function that assigns probability to each point $\mathbf{x} \in \mathbb{R}$. Although this analogy is wrong, we can use it for the purpose of this course.

- Normal distribution is a classic example. In this case:

$$\pi_X(\mathbf{x}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\mathbf{x}-\mu)^2}{2\sigma^2}\right)$$



Recap

Examples of Density function - multivariate Gaussian

- Let X be an $\mathbb{R}^n$-valued Gaussian random variable. Then there is a vector $\mathbf{m} \in \mathbb{R}^n$ (the mean) and a symmetric and positive-definite matrix $\mathscr{C}$ (the covariance matrix) such that the density function of X is

$$\pi_X(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |\mathscr{C}|}} \exp(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathscr{C}^{-1}(\mathbf{x} - \mathbf{m}))$$

  and we write $X \sim \mathscr{N}(\mathbf{m}, \mathscr{C})$.

- In this course we only consider zero-mean random variables.

Recap

Exercise

- What is the difference between a density function of random variable X and a histogram of a random variable X?

Statistical Formulation of Inverse Problems

- Recall formulation of an inverse problem.

$$\mathbf{y} = F(\mathbf{x}) + \varepsilon$$

- We define random variables to replace the components of the inverse problem.
  - define X to be the random variable of the unknown $\mathbf{x}$.
  - Define Y to be the random variable of the measurement $\mathbf{y}$.
  - Similarly E is the random variable of ε.
  - Note that randomness in F comes from $\mathbf{x}$, and F itself is not necessarily random.

Statistical Formulation of Inverse Problems

- The statistical modeling of the inverse problem is:

$$Y = F(X) + E$$

- What we need to define now is $\pi_X(\mathbf{x}), \pi_E(\mathbf{e})$:
  - $\pi_X(\mathbf{x})$ is called the prior density. It shows our belief (the probability) of any value $\mathbf{x}$ the.
  - $\pi_E(\mathbf{e})$ is the distribution of noise. Every sensor comes with a description of how precise measurements are.

- The solution to the inverse problem is then the conditional random variable $X \mid Y$, a.k.a. *the posterior*. We can also describe the posterior in terms of its density function.

$$\pi_{X \mid Y=\mathbf{y}}(\mathbf{x})$$

Statistical Formulation of Inverse Problems

Bayes' rule

- We now use the Bayes' rule to simplify the posterior density function:

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{\pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})}{\pi_Y(\mathbf{y})}$$

  - $\mathbf{y}$ is measurement data and we have it!
  - $\pi_{X|Y=\mathbf{y}}(\mathbf{x})$ is the posterior density function. This is what we want to compute.
  - $\pi_X(\mathbf{x})$ is the prior density function. This is something that we have.
  - $\pi_{Y|X}(\mathbf{y})$ is called the likelihood density function and is easy to evaluate (next slide).
  - $\pi_Y(\mathbf{y})$ is the probability of data. This is very hard to evaluate but generally unimportant. We find a way to deal with this term in the next days.

Statistical Formulation of Inverse Problems

Likelihood distribution $\pi_{Y|X}(\mathbf{y})$

- Recall a statistical inverse problem
$$Y = F(X) + E$$

- We want to compute $\pi_{Y|X=\mathbf{x}}(\mathbf{y})$. This means that we have $\mathbf{x}$:
$$Y = F(\mathbf{x}) + E$$

- Suppose that E is a Gaussian, i.e., $E \sim \mathcal{N}(0,\sigma^2)$, or $\pi_E(\mathbf{e}) = \dfrac{1}{\sqrt{2\pi\sigma^2}} \exp(-\mathbf{e}^2/(2\sigma^2))$.

- $F(\mathbf{x})$ is not random! Take it to the other side of the equation and write:
$$Y - F(\mathbf{x}) = E$$

- Can you guess what is the density function of the left-hand-side? $F(\mathbf{x})$ can be a mean of a distribution.

Statistical Inversion

Linear problem with Gaussian noise

- Recall a statistical inverse problem

$$Y = F(X) + E,$$

  Let

  - $X \sim \mathcal{N}(0,1)$, with

$$\pi_X(\mathbf{x}) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\mathbf{x}^2}{2}\right)$$

  - $E \sim \mathcal{N}(0,\sigma^2)$, with

$$\pi_E(\varepsilon) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\varepsilon^2}{2\sigma^2}\right)$$

  - $\pi_{Y|X=\mathbf{x}} \sim \mathcal{N}(F(\mathbf{x}), \sigma^2)$, with

$$\pi_{Y|X=\mathbf{x}}(\mathbf{y}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\mathbf{y} - F(\mathbf{x}))^2}{2\sigma^2}\right)$$

  - Since $\mathbf{y}$ is already measured, then $\pi_Y(\mathbf{y})$ is a constant.

Statistical Inversion

Linear problem with Gaussian noise

- Now we can put everything into the Bayes' rule:

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{\pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})}{\pi_Y(\mathbf{y})}$$

- This gives us the relation:

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{1}{c}\exp\left(-\frac{(\mathbf{y}-F(\mathbf{x}))^2}{2\sigma^2}\right)\exp\left(-\frac{\mathbf{x}^2}{2}\right)$$

- The constant c is referred to as the normalization constant.

Statistical Inversion

Linear problem with Gaussian noise - multivariate case $\mathbf{x} \in \mathbb{R}^n$

- Now we can put everything into the Bayes' rule:

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{\pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})}{\pi_Y(\mathbf{y})}$$

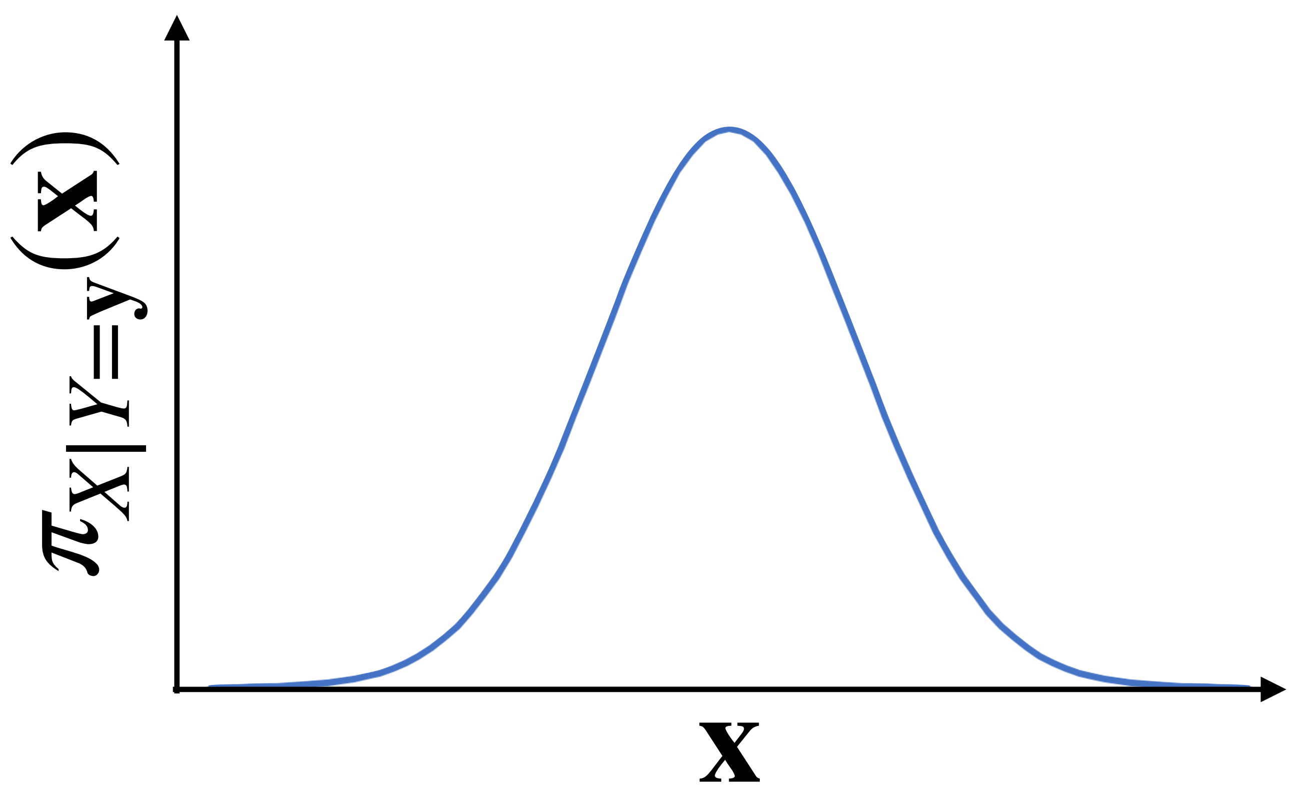
- This gives us the relation:

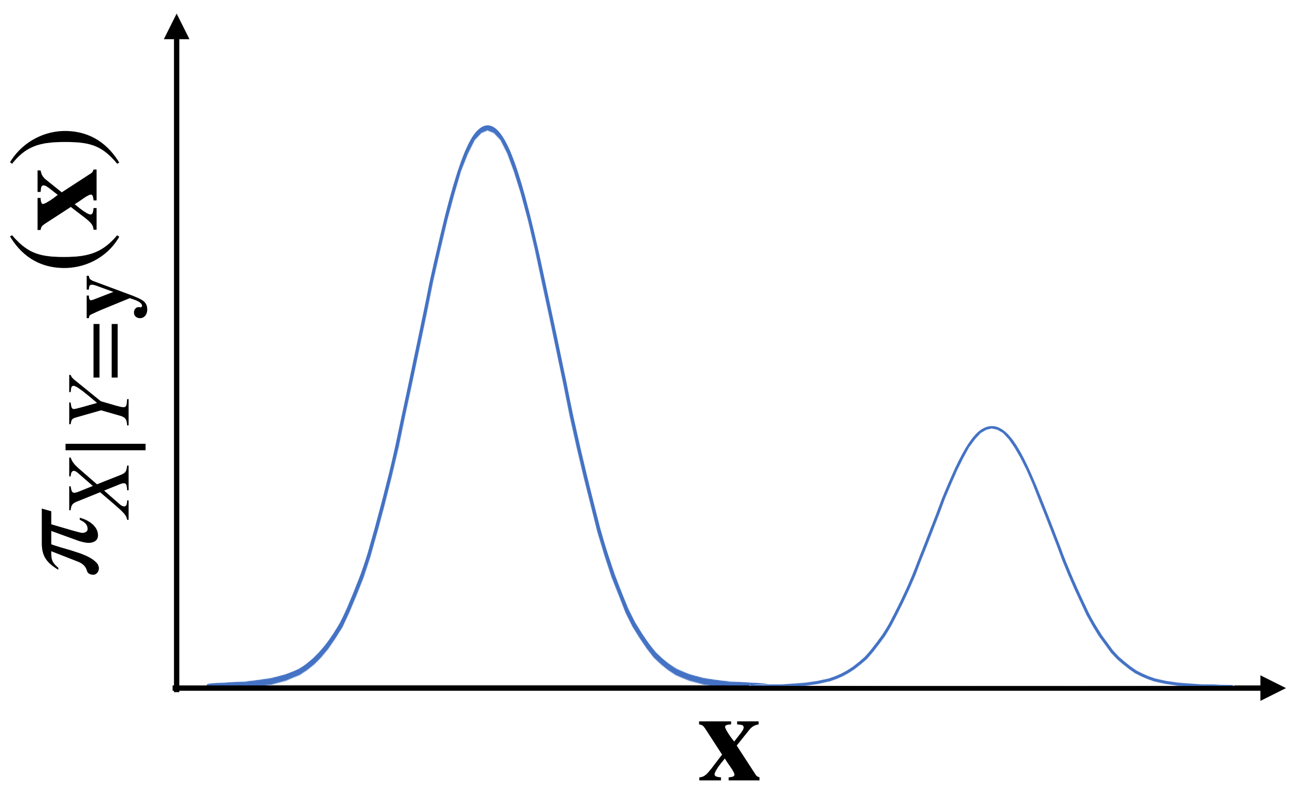
$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{1}{c}\exp\left(-\frac{\|\mathbf{y} - F(\mathbf{x})\|_2^2}{2\sigma^2}\right)\exp\left(-\frac{\mathbf{x}^T\mathscr{C}^{-1}\mathbf{x}}{2}\right)$$

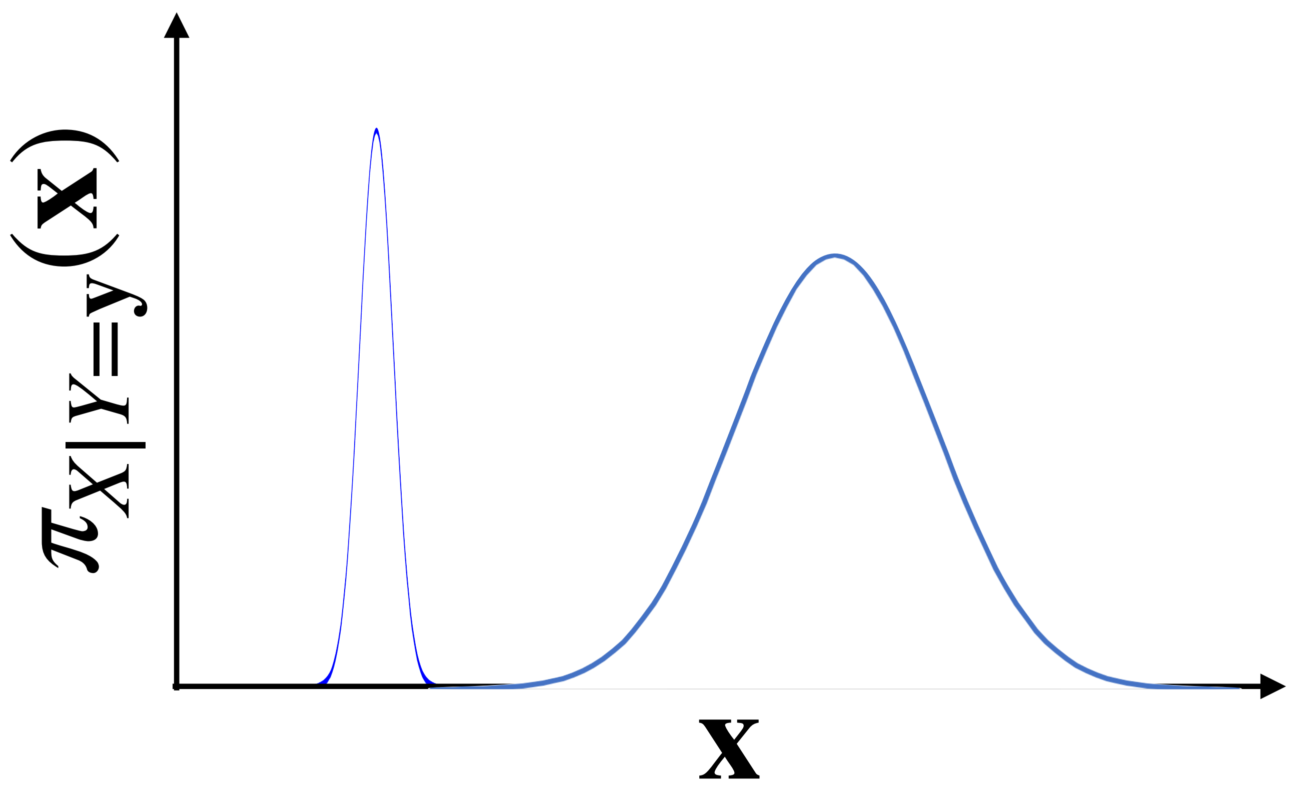
- The constant c is referred to as the normalization constant.

Point Estimations of the Posterior

- Now that we have a distribution $\pi_{X|Y=\mathbf{y}}$, what should we report as the solution to the inverse problem.
- Some examples of possible posterior distributions:







Point Estimations of the Posterior

- The maximum a posterior (MAP) estimation

$$\mathbf{x}_{\text{MAP}} := \arg\max_{\mathbf{x}} \quad \pi_{X|Y=\mathbf{y}}(\mathbf{x})$$

- The posterior mean (or sometimes conditional mean)

$$\mathbf{x}_{\text{mean}} := \mathbb{E}(X\,|\,Y = \mathbf{y}) = \int_{\Omega} \mathbf{x}\pi_{X|Y=\mathbf{y}}(\mathbf{x})\ d\mathbf{x}$$

- However, there is no right or wrong point estimators!

Connection to the Tikhonov regularization

Exercise (HW1)

- For the inverse problem:

$$\mathbf{y} = F(\mathbf{x}) + \mathbf{e}$$

- A classic solution to inverse problems are Tikhonov regularized optimization:

$$\mathbf{x}_{\text{Tyk}} := \arg\min_{\mathbf{x}} \|F(\mathbf{x}) - \mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_2^2$$

- Show that for the posterior

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{1}{c}\exp\left(-\frac{\|\mathbf{y} - F(\mathbf{x})\|_2^2}{2\sigma^2}\right)\exp\left(-\frac{\|\mathbf{x}\|_2^2}{2}\right),$$

  i.e., linear inverse problem with Gaussian noise and standard-normal prior ,i.e., $X \sim \mathcal{N}(0,I_n)$, we have

$$\mathbf{x}_{\text{Tik}} = \mathbf{x}_{\text{MAP}}$$

- What is the regularization parameter λ in this case?

Connection to the Tikhonov regularization

Exercise (HW1)

- Hints: Start with the MAP minimization problem.
- In the arg-min problem, you can take $\log$ of and/or multiply the argument with a positive constant without effecting its results.
- Same is true when dropping constants.
- When multiplying an arg-min problem with a negative value, the problem becomes arg-max, and vice-versa.

Markov chain Monte-Carlo

Sampling Complex distributions

Babak Maboudi - day 2 - Jyväskylä summer school 2025

Markov chains

Monte Carlo Integration

- Let X be an $\mathbb{R}^n$-valued random variable, i.e., a random variable which takes values in $\mathbb{R}^n$, and f be an integrable function. Then

$$\mathbb{E}(f(X)) = \int_{\mathbb{R}^n} f(\mathbf{x})\pi_X(\mathbf{x})\, d\mathbf{x} \approx \sum_{j=1}^{N} w_j f(\mathbf{x}_j),$$

- In Monte Carlo Integration, $\mathbf{x}_j$ are i.i.d. realization of π_X, then the approximator becomes the ergodic average:
  - Mean approximation

$$\mathbf{m} = \mathbb{E}(X) \approx \sum_{j=1}^{N} \frac{1}{N}\mathbf{x}_j$$

  - Variance approximation

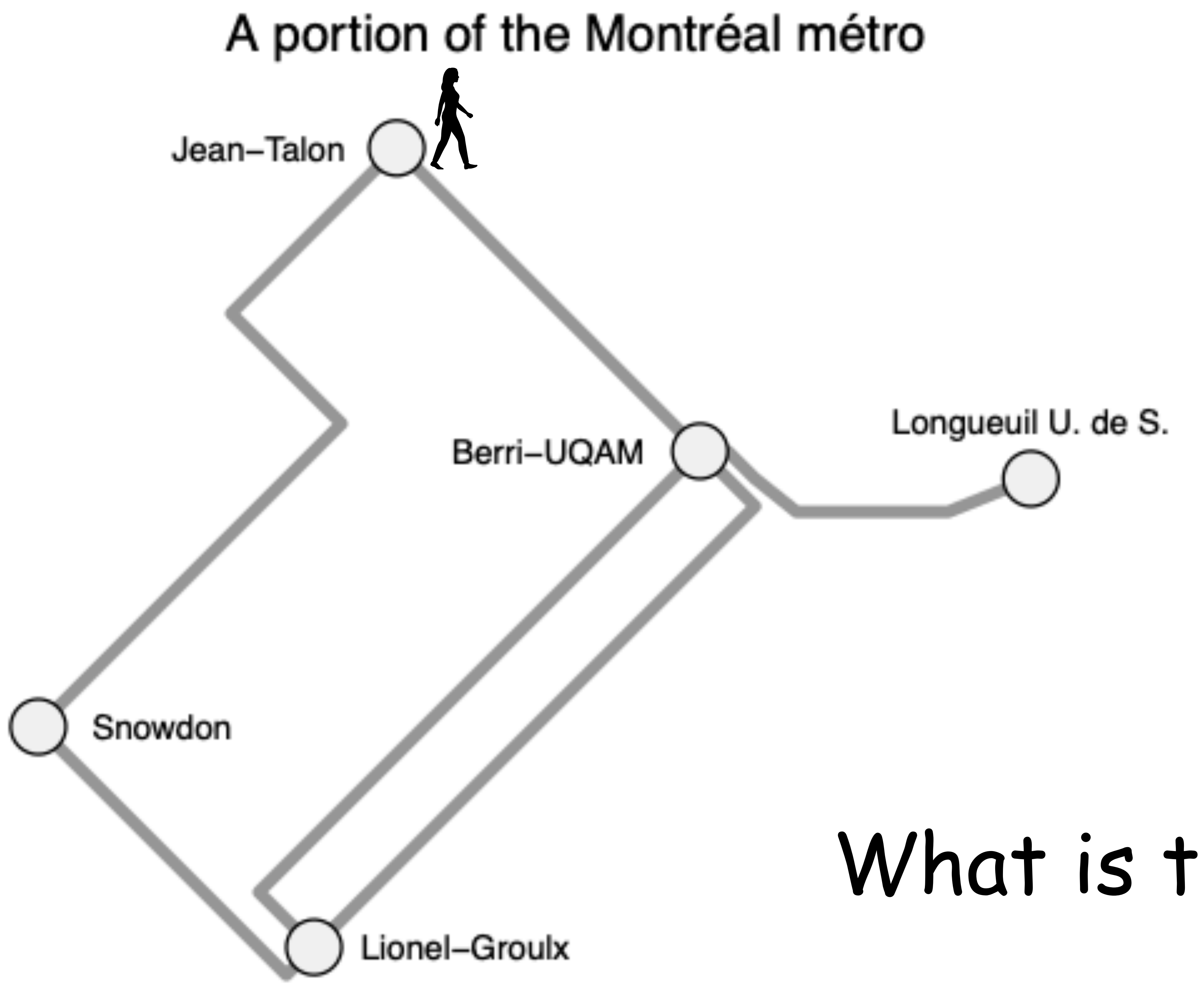
$$v = \mathrm{Var}(X) = \mathbb{E}(\|X - m\|_2^2) \approx \sum_{j=1}^{N} \frac{1}{N-1}\|\mathbf{x}_j - \mathbf{m}\|_2^2$$

Monte Carlo Integration

- However, the foundation for Monte Carlo estimation is independent realizations of the distribution of X.
- In Inverse Problems, we rarely have access to the distribution of X, this requires a complete knowledge of the density function.
- However, dependent sampling is possible! This is the principle idea of Markov-chain Monte Carlo methods.

Montréal Metro Map

Exercise from Art B. Owen (2013)



What is the distribution of

Alice's location?

Markov chains

Introducing notations

- Let $\Omega = \{\omega_1, \ldots, \omega_M\}$ be the *State Space*.
- Let X be a Ω*-valued random variable*.

Markov chains
Introducing notations

- Definition: A *Markov chain* is a sequence $X_0, X_1, X_2, \ldots, X_N$ (or $\{X_i\}_{i\leq N}$) of random variables with Markov property:
  - $\mathbb{P}(X_{i+1} \in A \,|\, X_j = x_j,\ 0 \leq j \leq i) = \mathbb{P}(X_{i+1} \in A \,|\, X_i = x_i)$
  - Here A is a set of states.
  - This is referred to being memoryless.

Markov chains

Further conditions

- A Markov chain is *(time-)homogeneous* if
$$\mathbb{P}(X_{i+1} = y \,|\, X_i = x) = \mathbb{P}(X_1 = y \in A \,|\, X_0 = x)$$

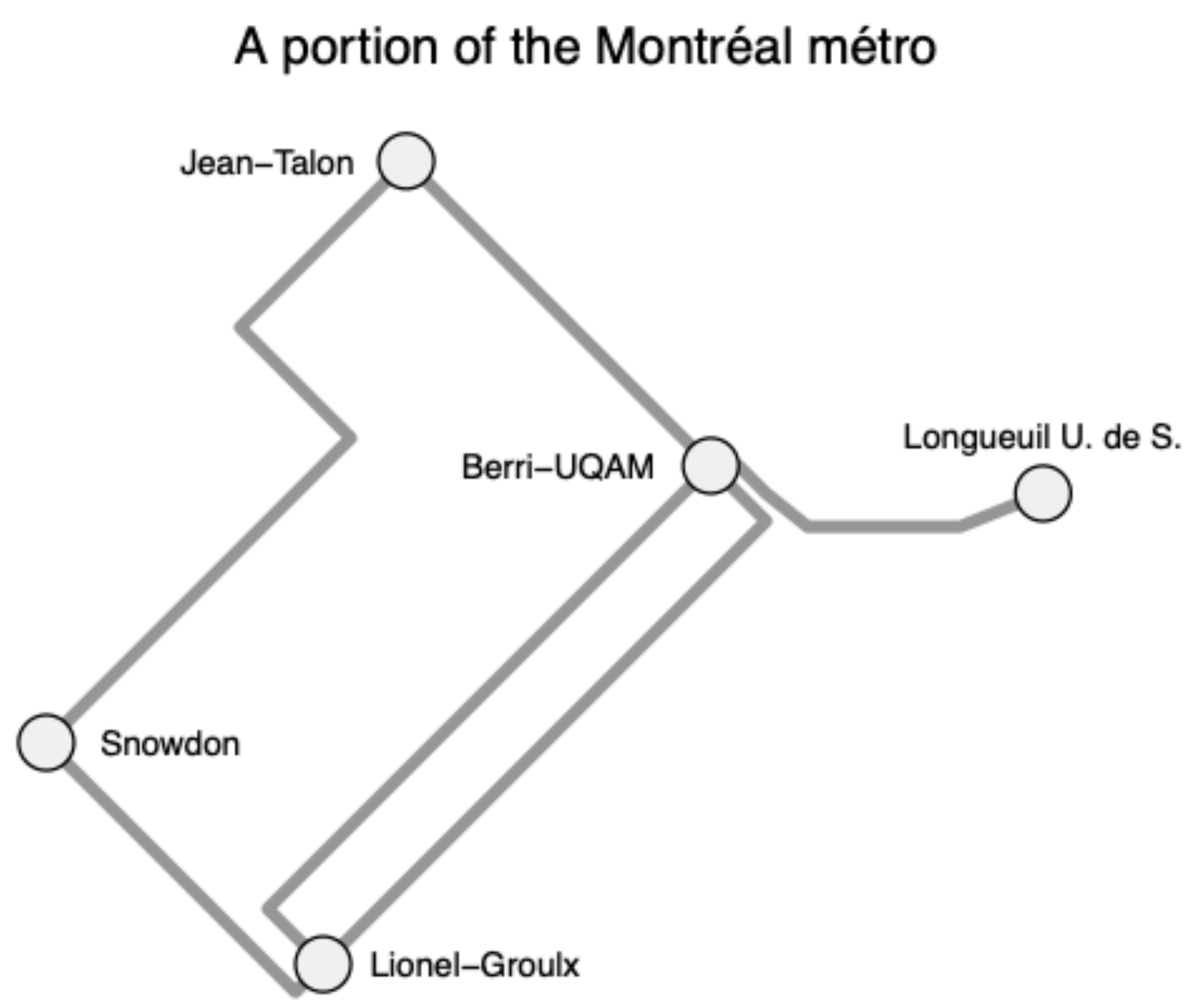
- A *transition probability* is the probability of going from state ω_i to ω_j:
$$p_{i\leftarrow j} = p_{ij} = \mathbb{P}(X_1 = \omega_i \in A \,|\, X_0 = \omega_j)$$

- A *transition matrix* is when you collect all transition probabilities in a matrix.
$$P = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1M} \\ p_{21} & p_{22} & \cdots & p_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1M} & p_{M2} & \cdots & p_{MM} \end{pmatrix}$$

Markov chains

Exercise 1 - `exercise_1.py`

- Choose a starting station
- Apply the function `roam` to move to the next station
- Draw 10000 samples from Montréal métro problem.
- Plot the histogram of the samples.
- Which station is the most likely destination?



Markov chains

Exercise 2 - `exercise_2.py`

- Now suppose that the initial state is not deterministic:
  $p_0(\omega_j)$: the probability of being at the jth station on the first step
- Similarly we define:
  $p_n(\omega_j)$: the probability of being at the jth station on the nth step
- What is the probability of being in the second station after 1 step?
  $$p_1(\omega_2) = p_0(\omega_1)p_{21} + p_0(\omega_2)p_{22} + \ldots + p_0(\omega_M)p_{2M} = \sum_j p_{2j}p_0(\omega_j)$$
- Complete the python code `exercise_2.py` to compute $p_1(\omega_2)$ when you are initially at any given station with equal probability, i.e.,
  $$p_0(\omega_j) = 1/5, \quad j = 1,\ldots,5,$$
- Can you write an operation between P and p_0 that gives you all the probabilities $p_1(\omega_j)$, for $j = 1,\ldots,5$?
- What is the sum of the elements in p_1? Why?

Markov chains

- We have
$$p_1 = Pp_0,$$
then,
$$p_2 = Pp_1,$$
and
$$p_n = Pp_{n-1}.$$

- What is the transition matrix Q for doing 2 steps? i.e., what is the matrix Q that gives you:
$$p_2 = Qp_0$$
Hint: look at the Markovian principle (the recursive definitions above).

Markov chains

Exercise 3

- Compute the transition matrix, P^2, for 2 steps?
- Compute the transition matrix, P^{200}, for 200 steps?
- What does the pattern in P^{200} mean?
  Hint: the component $[P^{200}]_{ij}$, i.e., the element on the ith row and the jth column of P^{200}, means the probability of starting at the station j and after 200 steps of the Markov chain arriving at station i.

Sampling using a Markov chain

Explanation,

- What ever value X_0 has it will be almost forgotten (independent) in X_{100}.
- What ever value X_{100} has, it will be forgotten (independent) in X_{200}.
- If we take a widely separated sequence of equi-spaced samples we should get a nearly i.i.d. samples.
- Repeat the Markov chain sampling in `exercise_1.py`, but this time select only every 100 samples. Create a histogram and normalize it.
- Find the distribution p_{1000}, i.e., $P^{1000} p_0$. Compare the histogram with p_{1000}.

Markov Chain

Stationary distribution

- We say π is a stationary distribution when:
  $$P\pi = \pi$$
  In other words, the transition matrix doesn't change the distribution.

Irreducible and periodic transition kernels

- What can you say about these transition matrices? Do they have a unique stationary distribution?

$$P_1 = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \end{pmatrix}, \qquad P_1 = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \end{pmatrix}.$$

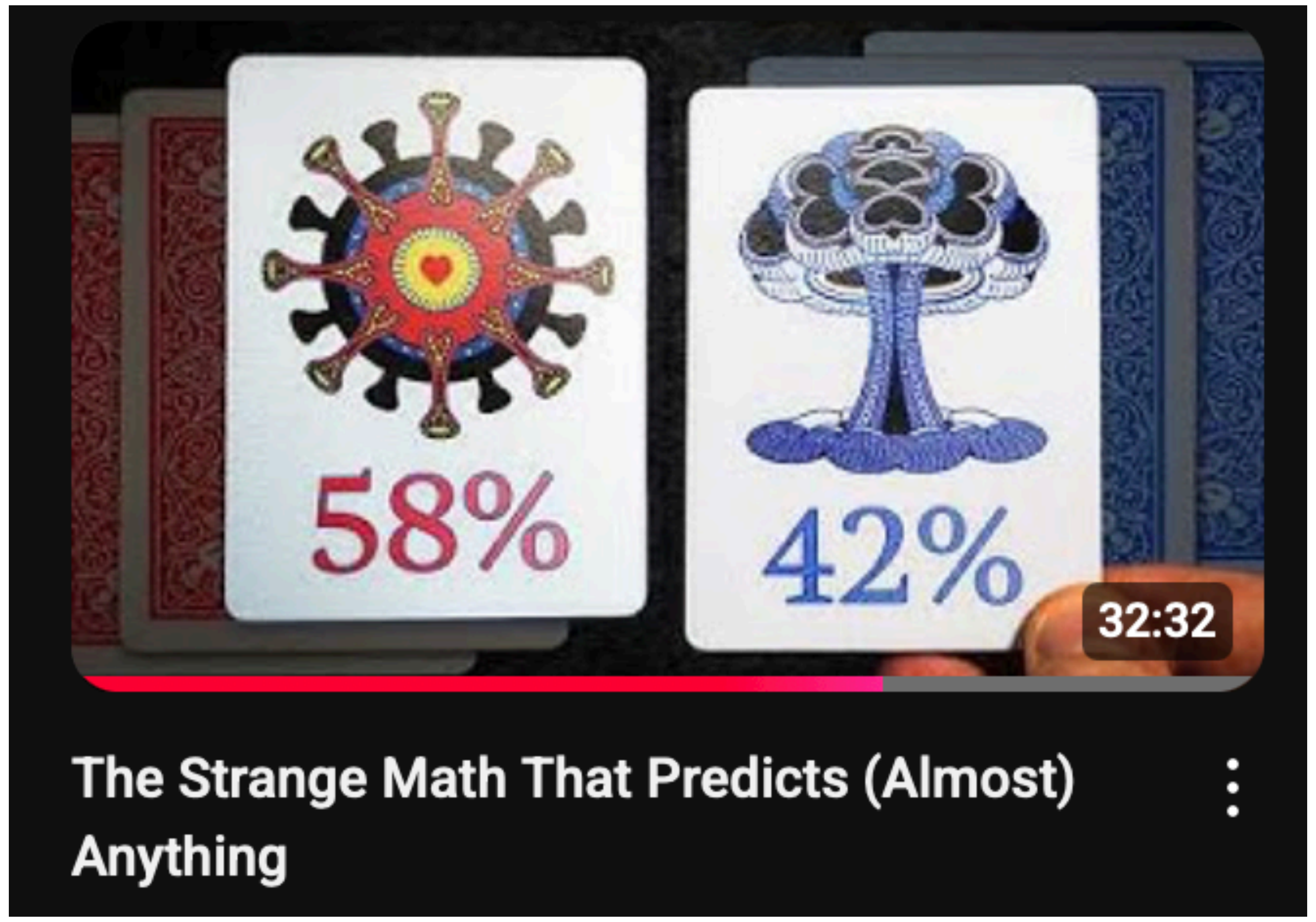
Irreducible and aperiodic transition kernels

- Theorem: If a transition matrix P is irreducible and aperiodic, and has a stationary distribution π then:

$$\lim_{n\to\infty} \mathbb{P}_{\omega_0}(X_n = \omega) = \pi(\omega)$$

- This “means” that the Markov chain method can arrive at the stationary distribution.

Veritasium video on Markov and Markov chains

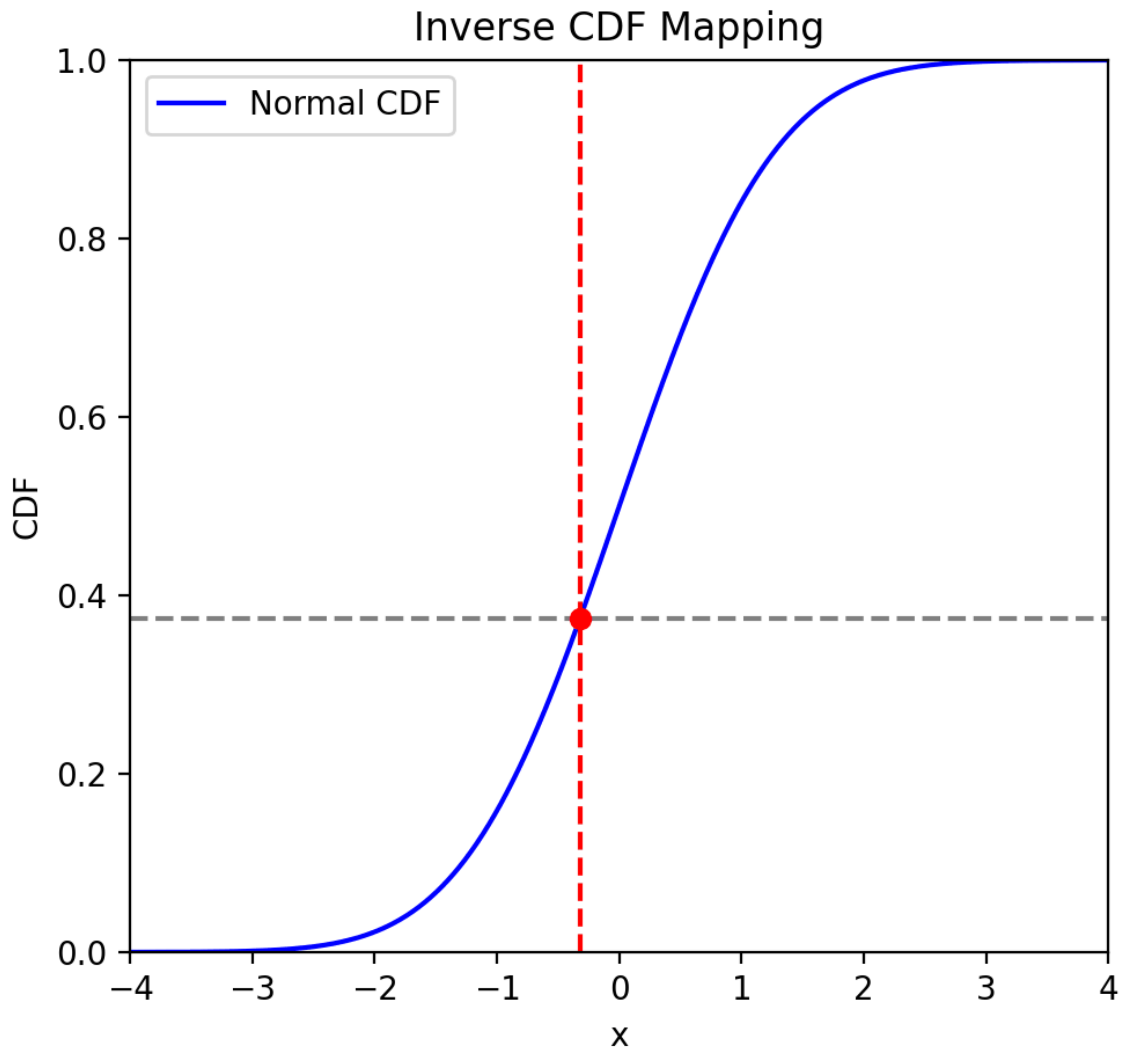


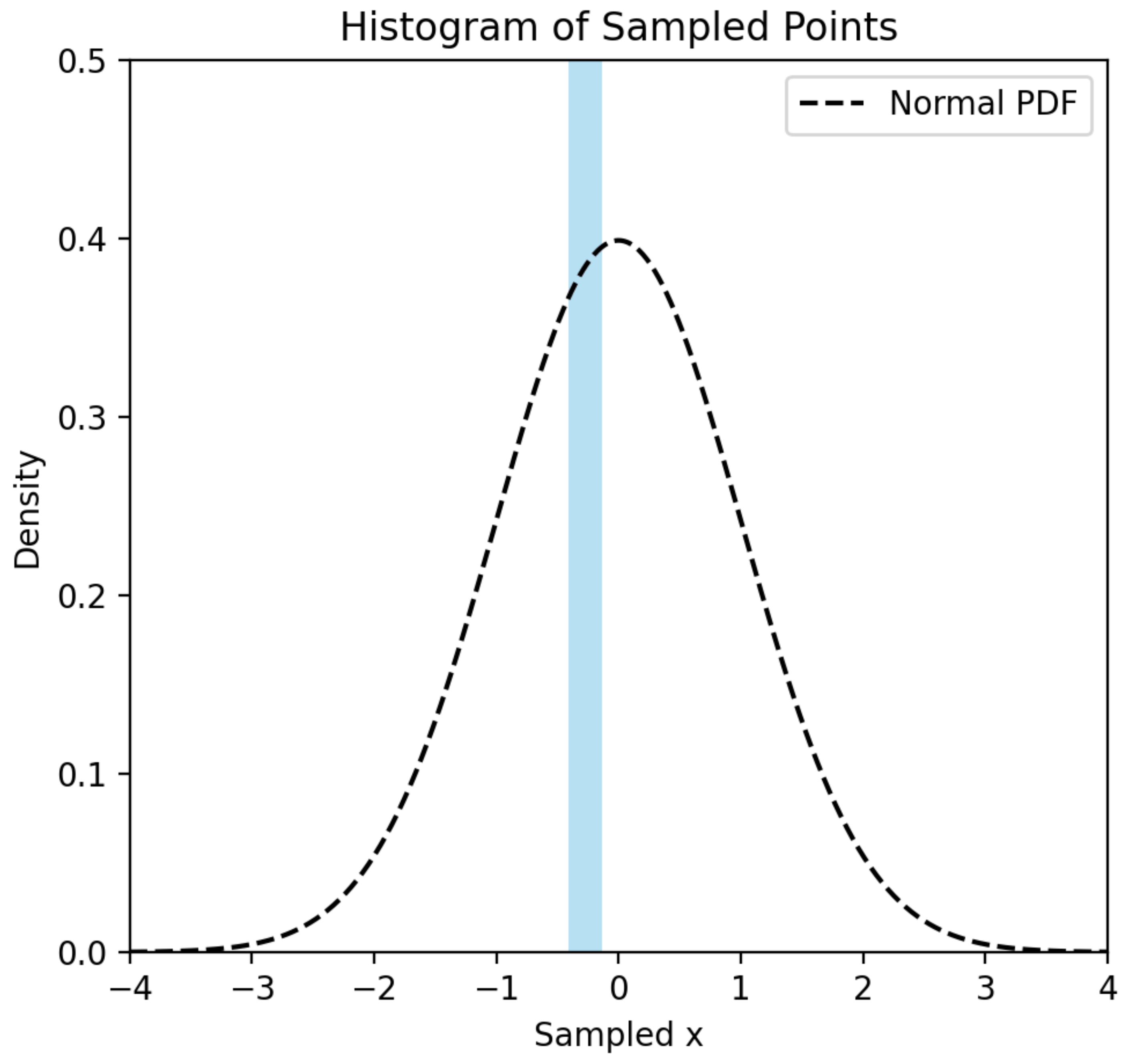
Acceptance/Rejection Sampling

Sampling from a Distribution

The inverse CDF method

$$F_X(x) = \mathbb{P}(X < x) = \int_{-\infty}^{x} \pi_X(x)\ dx$$

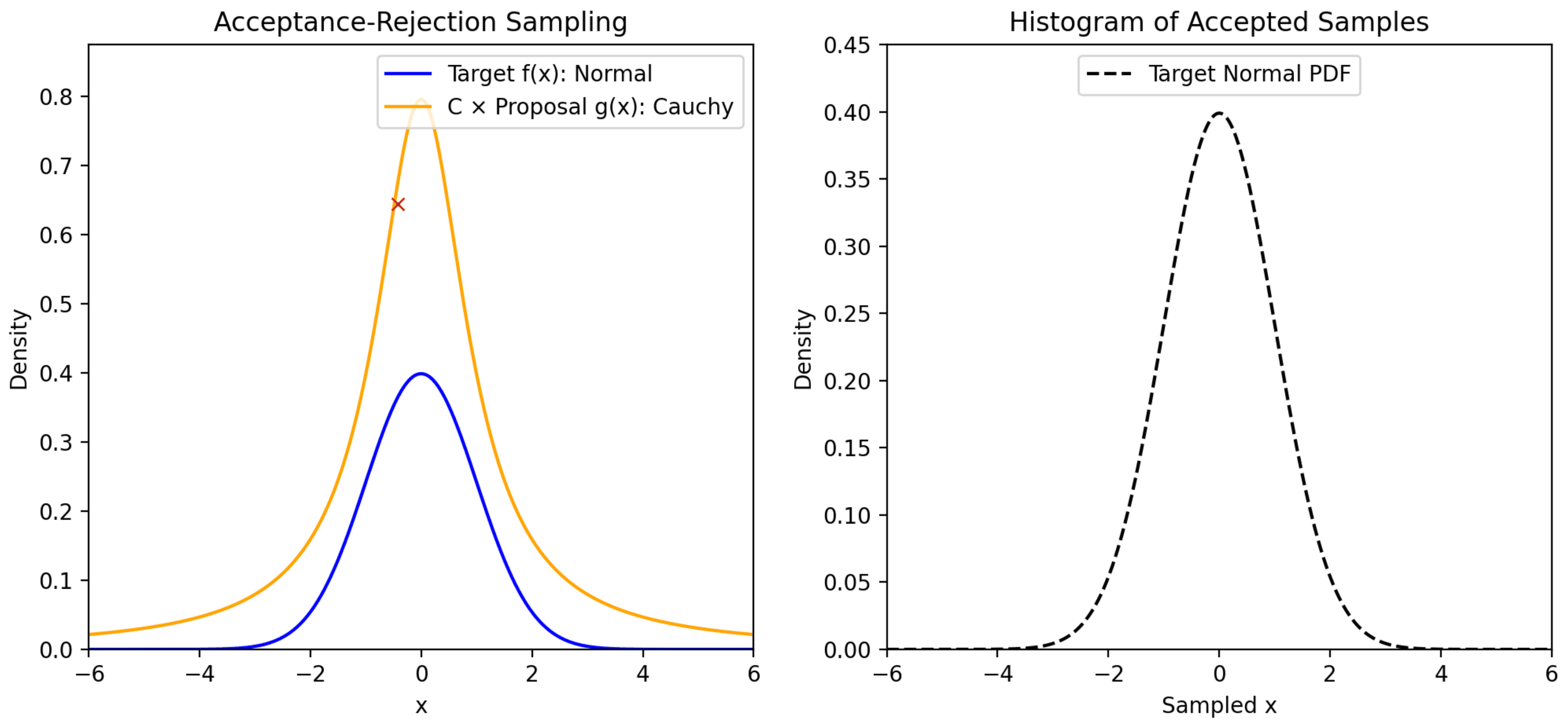




Sampling from a Distribution

Acceptance/Rejection method

- f is the density of the target distribution
- g is a density of a distribution that is easy t sample from (e.g. using the inverse CDF method).
- Algorithm:
    1. Sample y according to g
    2. Sample u according to $U(0,1)$
    3. If $u \leq f(y)/(Cg(y))$ accept, otherwise reject
    4. Repeat until desired samples achieved.



Acceptance/Rejection Sampling
Exercise (HW2)

- Get the Python script `exercise_4.py` from day 2 folder.
- Write Python functions `f(x)` and `g(x)` that computes the density functions of target Gaussian distribution and proposal Cauchy distribution:

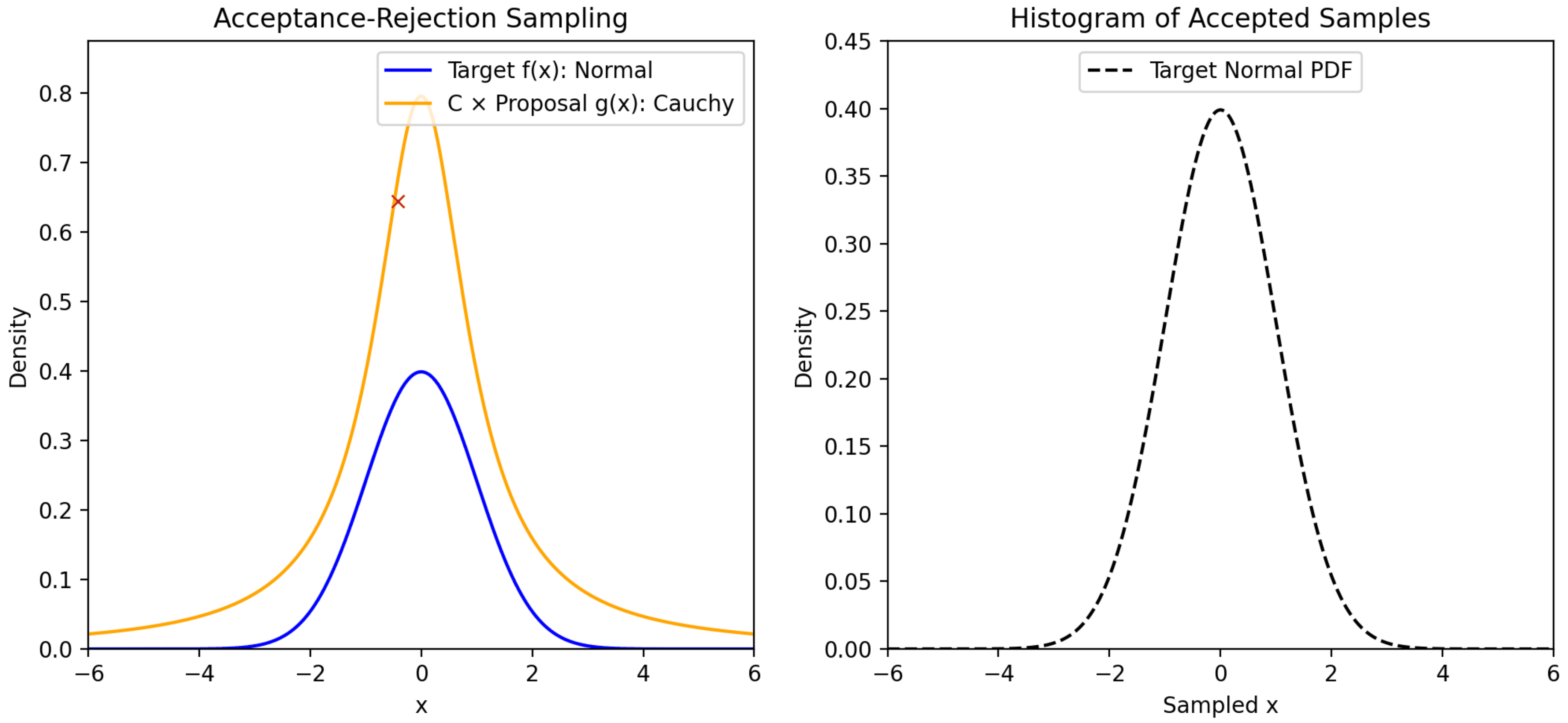
$$f(x) = \frac{1}{\sqrt{(2\pi)}} \exp(-\frac{x^2}{2}), \qquad g(x) = \frac{1}{\pi(1+x^2)}$$

- Set $c = 2$ and perform acceptance/rejection to draw 2000 samples from the distribution of f, i.e.,
  - draw a sample $x^\star$ from the proposal distribution g.
  - Draw a number from the uniform distribution $u \sim U(0,1)$
  - If $cg(x)u \leq f(x)$ accept $x^\star$ as a sample from f, otherwise reject.
- Plot the histogram of the samples and show that they approximate a standard-normal distribution.
- Choose the "step-size" $c = 1,1.52,$ and 2 and repeat the sampling. Compute the number of accepted samples. Which value is the best and why?

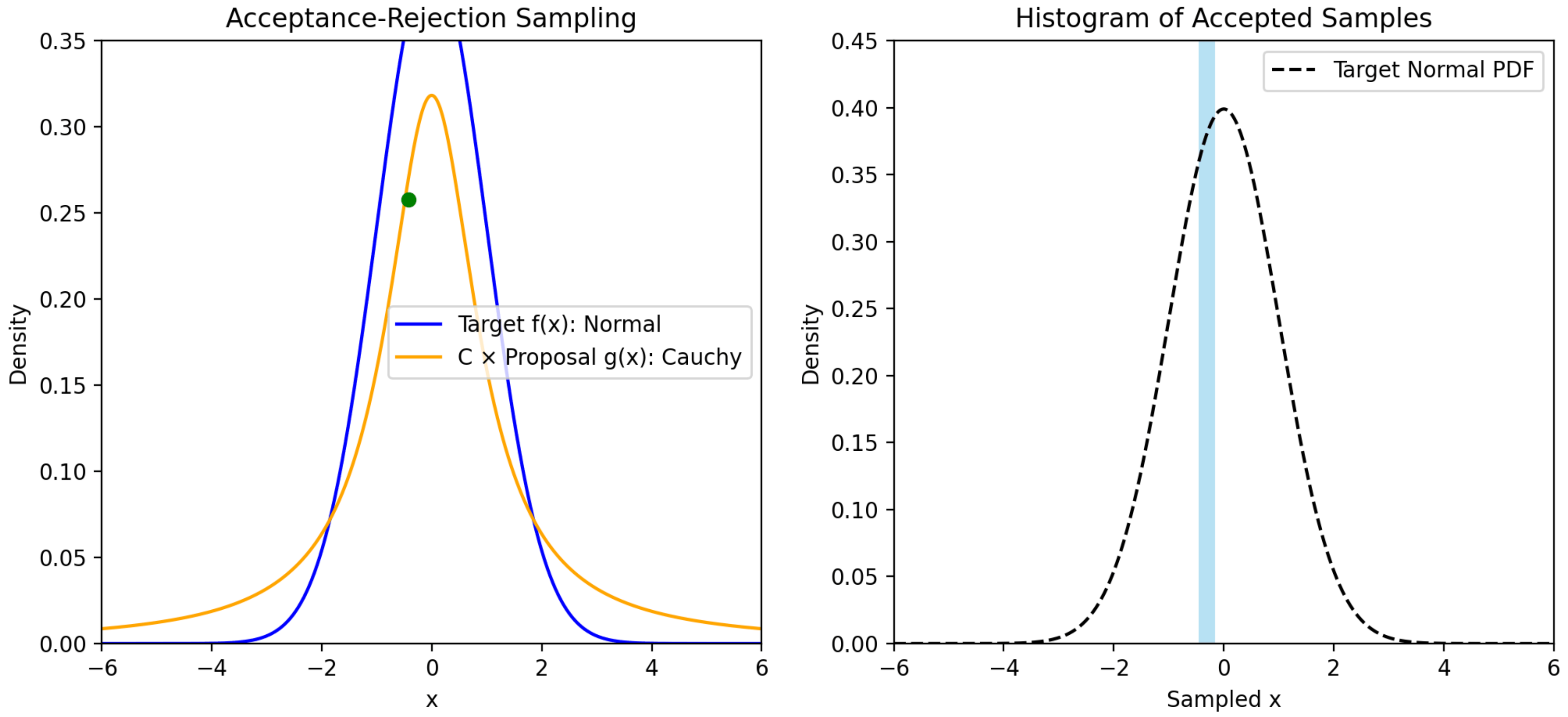
Sampling from a Distribution

Acceptance/Rejection method

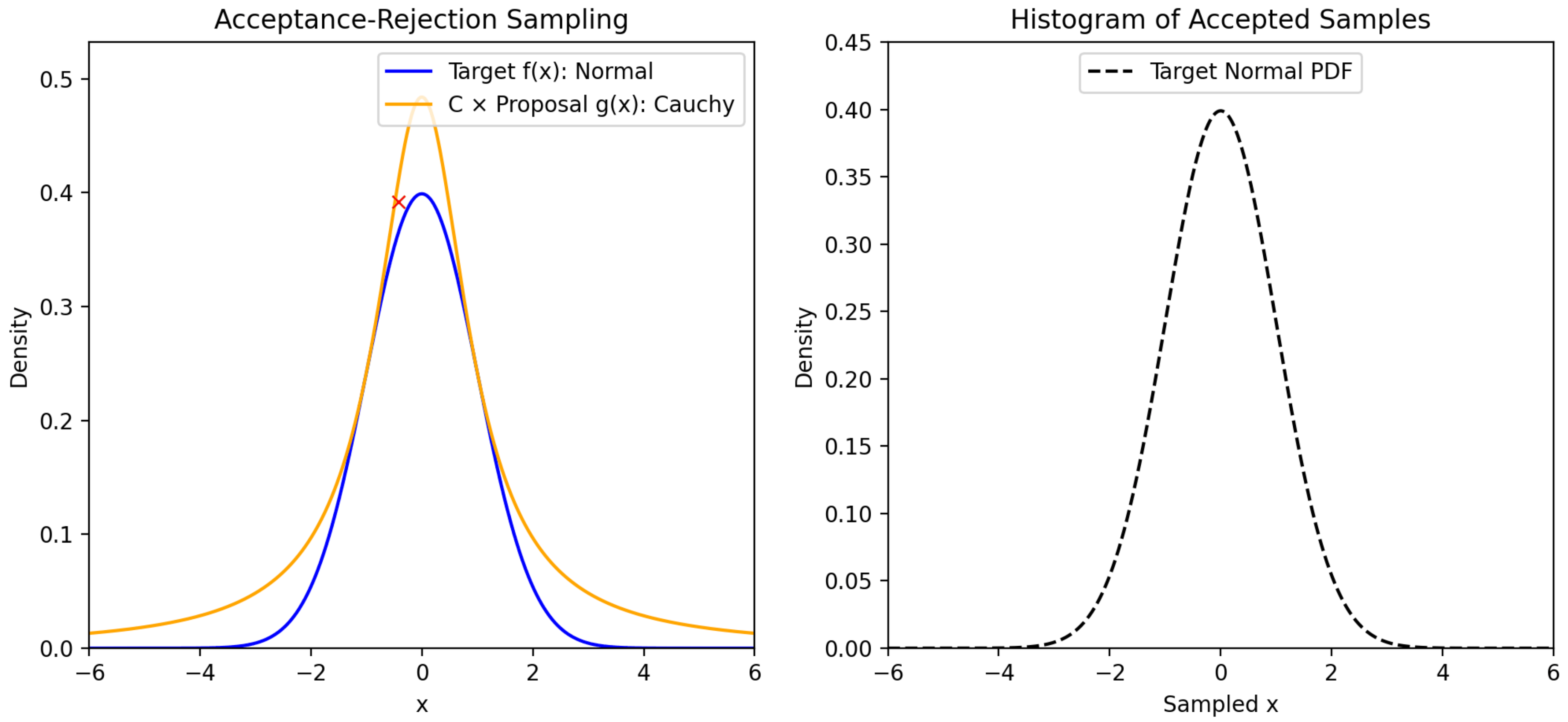
$c = 2.5$ acc. rate $= 0.39$



$c = 1$ acc. rate $= 0.71$



$c = 1.52$ acc. rate $= 0.63$



Metropolis-Hastings Algorithm

Sampling from complex distributions

Babak Maboudi - day 3 - Jyväskylä summer school 2025

New notation

- From now on will consider continuous state space
  before: $\Omega = \{\omega_1, \ldots, \omega_M\}$ now $\Omega = [0,1]$ or $\mathbb{R}$
- Transition probability:
  before: $[P]_{ij} = p_{ij}$ now $P(y \leftarrow x)$
- Transition strategy:
  before: matrix now transition kernel
- We still may use sum for an easier visual interpretation (to avoid introducing differentials and probability measures)

Intention

- We want to find the transition matrix/kernel P that results in a stationary distribution π, where π is the posterior distribution of out inverse problem.
- Recall that the Bayes' rule indicates:

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) = \frac{\pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})}{\pi_Y}(\mathbf{y}) \propto \pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})$$

  - $\pi_{X|Y}$ is the posterior
  - $\pi_{Y|X}$ is the likelihood
  - π_X is the prior distribution
  - $\pi_Y(\mathbf{y})$ is the data distribution, which is difficult to compute and is independent of X.

Detailed Balance

Exercise 1

- Recall that when arriving at the stationary distribution we will have
$$P\pi = \pi$$
  - Here, P is the transition matrix/kernel.
  - π is the stationary distribution.
- We want to design a P with a stationary distribution $\pi_{X|Y=\mathbf{y}}$.

Balance condition

- Recall that in a stationary distribution π of a transition probability matrix P we have that:

$$\sum_{\mathbf{x}\in\Omega} P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = \pi(\mathbf{y})$$

$$= \int_{\Omega} P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x})\ d\mathbf{x}$$

- Now multiply right side with $1 = \sum_{x\in\Omega} P(\mathbf{x} \leftarrow \mathbf{y})$

$$\sum_{\mathbf{x}\in\Omega} P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = \sum_{\mathbf{x}\in\Omega} P(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y})$$

or

$$\sum_{\mathbf{x}\in\Omega,\mathbf{x}\neq\mathbf{y}} P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = \sum_{\mathbf{x}\in\Omega,\mathbf{x}\neq\mathbf{y}} P(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y})$$

Detailed Balance Condition

- Recall the balance condition:

$$\sum_{\mathbf{x}\in\Omega,\mathbf{x}\neq\mathbf{y}} P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = \sum_{\mathbf{x}\in\Omega,\mathbf{x}\neq\mathbf{y}} P(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y})$$

- A sufficient condition for balance condition is detailed balance condition:

$$P(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = P(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y}), \quad \text{for all } \mathbf{x}, \mathbf{y} \in \Omega$$

- We already have π (the posterior) how can we choose P?

- It would be very nice to choose any P that we want 😈😅

Metropolis-Hastings Acceptance Probability

- Let us design a transition matrix/kernel $Q(\mathbf{y} \leftarrow \mathbf{x})$ and plug-into the detailed balance relation:
$$Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) \neq Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y}),$$
- We can correct the imbalance using acceptance/rejection (Hastings) strategy:
$$A(\mathbf{y} \leftarrow \mathbf{x})Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x}) = A(\mathbf{x} \leftarrow \mathbf{y})Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y}),$$
  - We know π: the posterior
  - We know Q: generic (aperiodic, irreducible and time-reversible) transition kernel
  - We only need to know A: The probability of accepting $\mathbf{y}$ moving from $\mathbf{x}$.
  - All terms are ≤ 1

Metropolis-Hastings Acceptance ratio

- Let us reformulate the final equation

$$A(\mathbf{y} \leftarrow \mathbf{x}) = \frac{Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y})}{Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x})} A(\mathbf{x} \leftarrow \mathbf{y}),$$

- Without loss of generality we assume:

$$A(\mathbf{x} \leftarrow \mathbf{y}) = \lambda Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{x})$$

- Then

$$A(\mathbf{y} \leftarrow \mathbf{x}) = \lambda Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{y})$$

- So

$$\lambda \times \max \left\{ Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{x}), Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{y}) \right\} = 1$$

- Substituting into above:

$$A(\mathbf{x} \leftarrow \mathbf{y}) = \min \left(1, \frac{Q(\mathbf{x} \leftarrow \mathbf{y})\pi(\mathbf{x})}{Q(\mathbf{y} \leftarrow \mathbf{x})\pi(\mathbf{y})} \right)$$

Random Walk Metropolis-Hastings algorithm

- Goal: To create a Markov chain with stationary distribution of the posterior distribution.
- Take a state $X_0 = \mathbf{x}$ with probability $\pi_X(\mathbf{x}_0) \neq 0$ as your first element in your Markov chain.
- Choose a transition kernel Q of your choice!
- Propose a new sample $\mathbf{x}^\star \sim Q_\mathbf{x}$.
- Form the Metropolis-Hasting acceptance ratio

$$a = \min\left(1, \frac{Q(\mathbf{x}^\star \leftarrow \mathbf{x})\pi_{X|Y}(\mathbf{x}^\star)}{Q(\mathbf{x} \leftarrow \mathbf{x}^\star)\pi_{X|Y}(\mathbf{x})}\right)$$ or for a symmetric Q, $a = \min\left(1, \frac{\pi(\mathbf{x}^\star)}{\pi(\mathbf{x})}\right)$

- Draw a random number $u \sim U(0,1)$
- If $\alpha > u$ accept the proposal and set $X_n = \mathbf{x}^\star$, else reject the sample and set $X_n = \mathbf{x}$.
- Repeat until the desired samples are collected.

Example (most common) transition kernel

Brownian motion - a symmetric transition strategy

Your (probably) First Statistical Inversion

- Consider the inverse problem:
$$Y = \mathscr{A}X + E$$
  - $X \in \mathbb{R}^2$, $Y, E \in \mathbb{R}^3$ and
$$\mathscr{A} = \begin{pmatrix} 1 & -1 \\ 1 & -2 \\ 2 & 1 \end{pmatrix}.$$
  - Assume that
$$\pi_E(e) \propto \exp(-\frac{1}{2\sigma^2}\|e\|_2^2), \text{ with } \sigma^2 = 0.09.$$
  - Assume that the prior distribution for X is standard normal distribution, i.e.,
$$\pi_X(x) \propto \exp(-\frac{1}{2}\|x\|_2^2)$$

Your (probably) First Statistical Inversion

Exercise (HW3)

- Use the Bayes' theorem to write the density function of the posterior distribution, $\pi_{X|Y=\mathbf{y}}(\mathbf{x})$, up to proportionality constant,

$$\pi_{X|Y=\mathbf{y}}(\mathbf{x}) \propto \cdots$$

  i.e., drop terms that are not a function of $\mathbf{x}$.

Your (probably) First Statistical Inversion

Exercise (HW3)

- Get the Python code `exercise.py` for day 3.
- Write a Python function `prior` that computes the prior density $\pi_X(\mathbf{x})$ without the proportionality constant
- Write a Numpy matrix to define the matrix $\mathscr{A}$
- Use the noise standard deviation `sigma` and the measurement `y_obs` in the code

Your (probably) First Statistical Inversion

Exercise (HW3)

- Write a Python function `likelihood` that computes the likelihood density $\pi_{Y|X=\mathbf{x}}(\mathbf{y})$ without the proportionality constant. Here, $\mathbf{y}$ is `y_obs`.
- Write a Python function `posterior` that computes the posterior density $\pi_{Y|X=\mathbf{x}}(\mathbf{y})\pi_X(\mathbf{x})$ without the proportionality constant. Use `prior` and `likelihood` functions.

Your (probably) First Statistical Inversion

Exercise (HW3)

- Complete the code for the Metropolis-Hastings random walk algorithm:
  - Set the dimension of X
  - Choose an initial guess $\mathbf{x}$ for the Markov chain process
  - Choose a step size c for the Metropolis proposal. $c = 1$ is a good step size for this problem

Your (probably) First Statistical Inversion

Exercise (HW3)

- In the acceptance-rejection loop:
  - Choose a Gaussian proposal kernel. I.e., propose a point $x^\star$ according to
    $$\mathbf{x}^\star = \mathcal{N}(\mathbf{x}, c^2 I_2) \quad \text{or} \quad \mathbf{x}^\star = \mathbf{x} + c\mathbf{z}$$
    with $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $\mathbf{e} \sim \mathcal{N}(0, I_2)$
  - Compute the acceptance probability
    $$a = \min\left(1, \frac{\text{posterior}(\mathbf{x}^\star)}{\text{posterior}(\mathbf{x})}\right)$$
  - Draw a sample $u \sim U([0,1])$
  - Accept $\mathbf{x}^\star$ if $u < \alpha$ and set $\mathbf{x} = \mathbf{x}^\star$ and else reject the proposed sample.

Your (probably) First Statistical Inversion

Exercise (HW3)

- Draw 50000 samples from the posterior distribution with step size $c = 1$ in the random walk Metropolis-Hasting algorithm.
- Sub-sample (skip every 10) equi-spaced samples to achieve near i.i.d. samples of the posterio.
- Draw a 2D histogram of the posterior and mark the posterior mean on it.

Your (probably) First Statistical Inversion

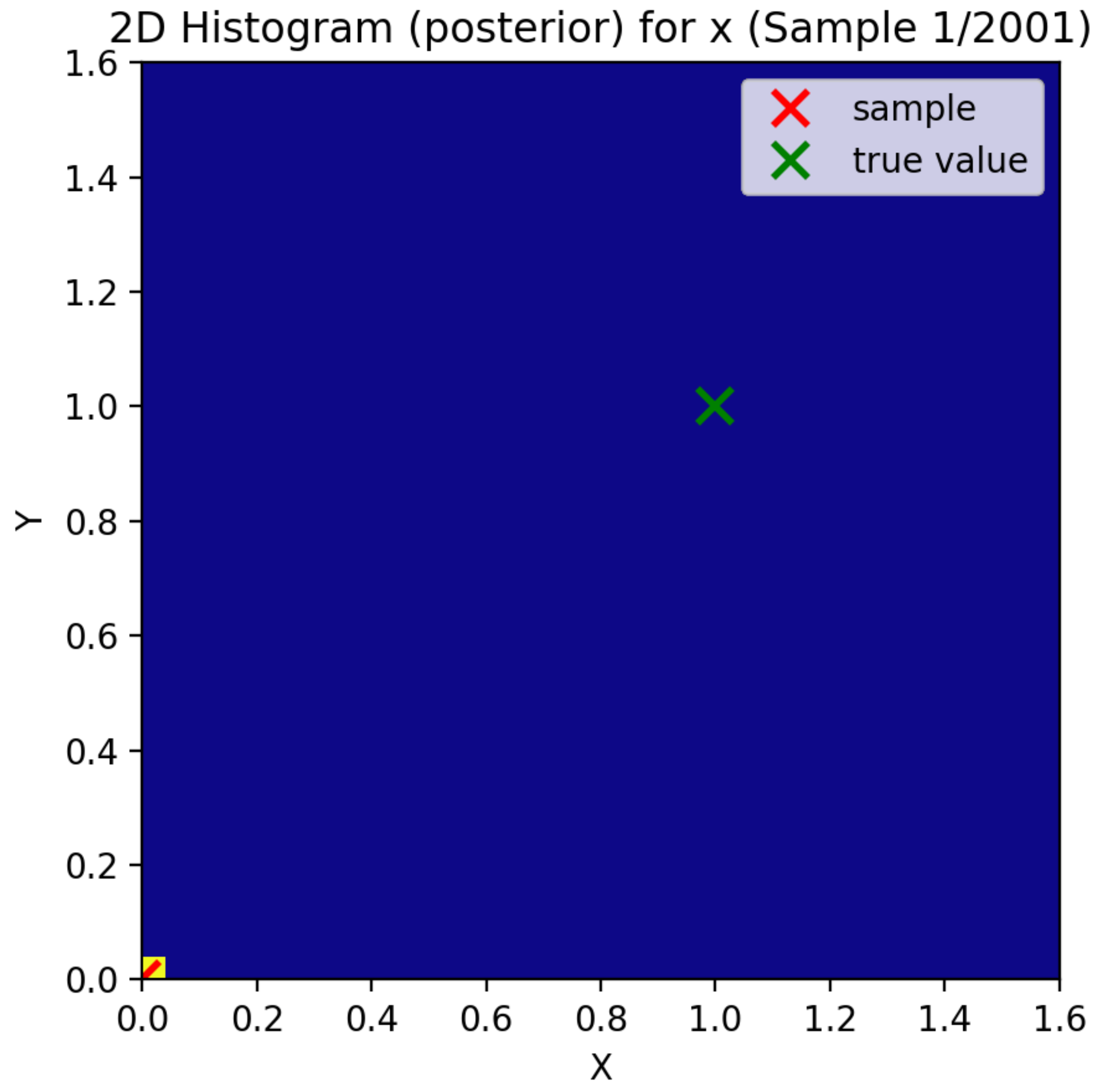
Exercise (HW3)

- Let the step-size $c \in \{0.001, 0.1, 10\}$. Plot the posterior 2D histogram for each step size and explain the differences. Which value of c is better for this problem, in your opinion, and why?

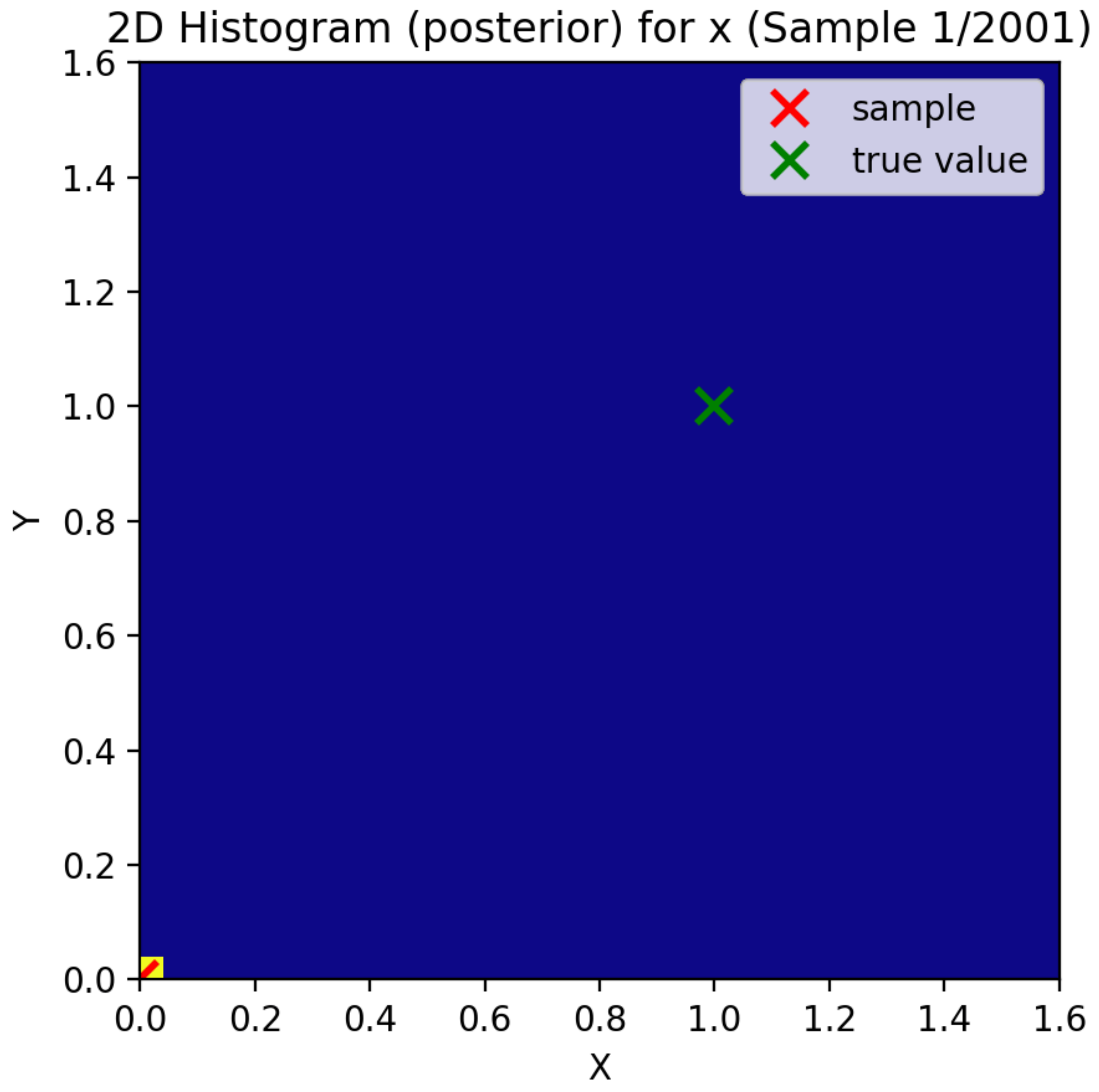
- Let noise variance be $\sigma^2 \in \{0.01, 0.1, 1\}$. Plot the posterior 2D histogram for each noise variance. What differences do you observe? Discuss uncertainty in the posterior mean estimation.

The effect of step size in random walk MH

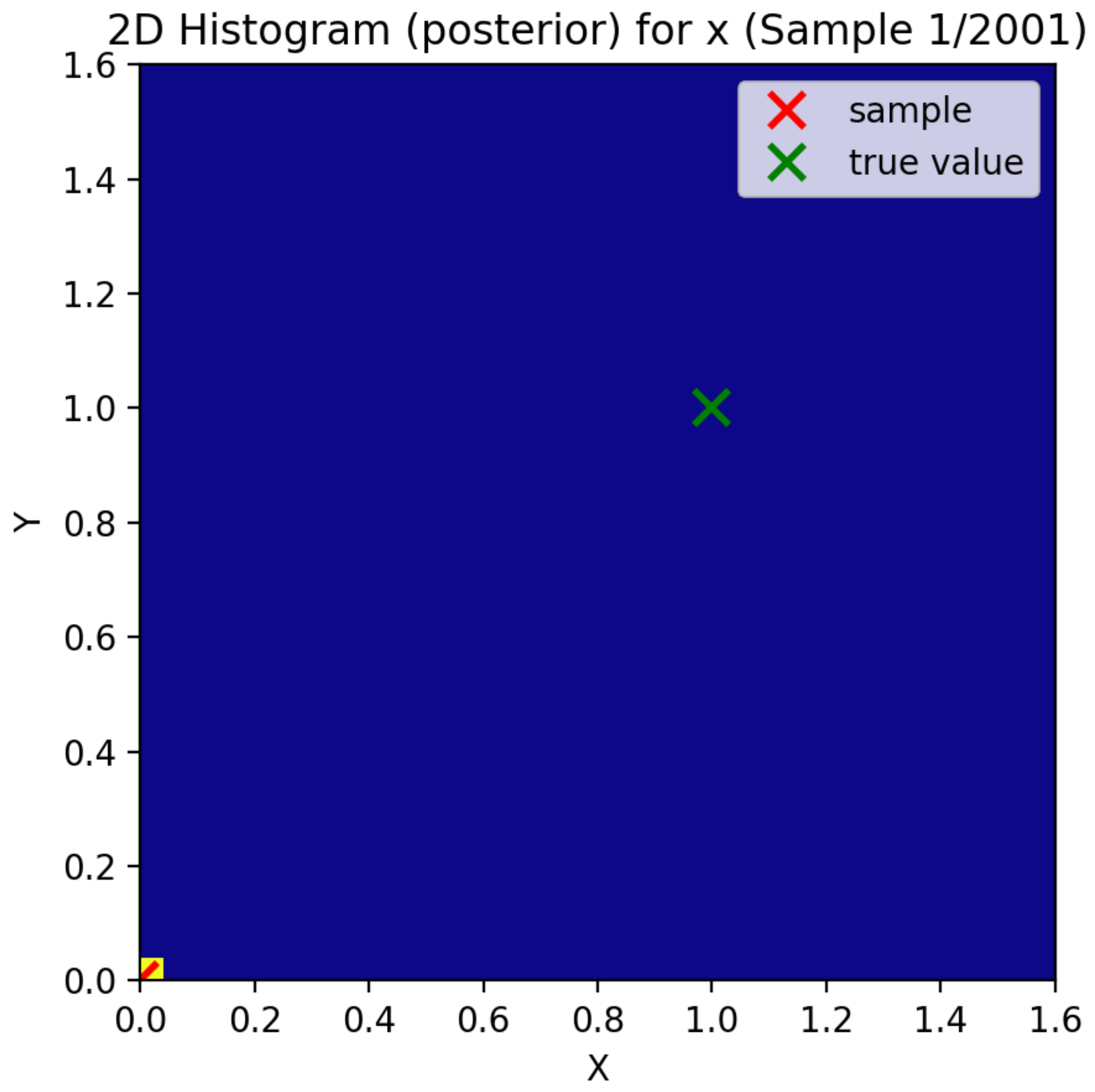
$c = 0.01$



$c = 0.1$



$c = 1$



Announcement

- 1 Postdoc Position (Jan. 1, 2026)
- 1 PhD position (Mar. 1, 2026)
- 1 Postdoc Position (2027)

Reach out to me:
babak.maboudi@oulu.fi

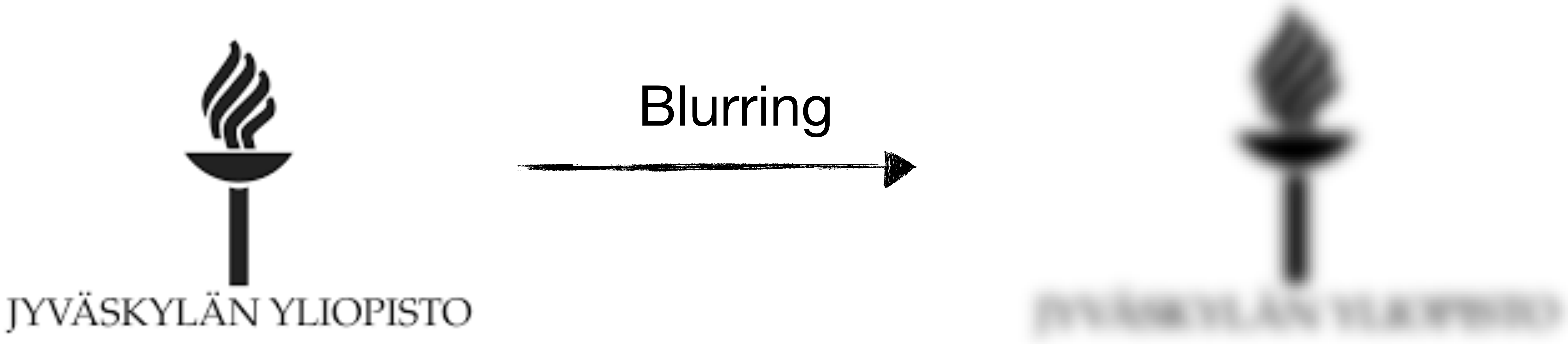
UNIVERSITY OF OULU

Prior Engineering

How to choose the right prior for your Bayesian inverse problems

Babak Maboudi - day 4 - Jyväskylä summer school 2025

The De-blurring Problem



De-blurring inverse problem in 1D

- Let $x(s)$ represent (continuous) pixel intensities or an image in 1D, with $s \in [0,1)$.

- We can express blurring using a convolution operation:

$$y(s) = (g * x)(s) := \int_0^1 g(t - s)x(s) \; dt$$

  g is called a kernel and a typical kernel is a Gaussian kernel:

$$g(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{z^2}{2\sigma^2}\right)$$

De-blurring Inverse Problem in 1D

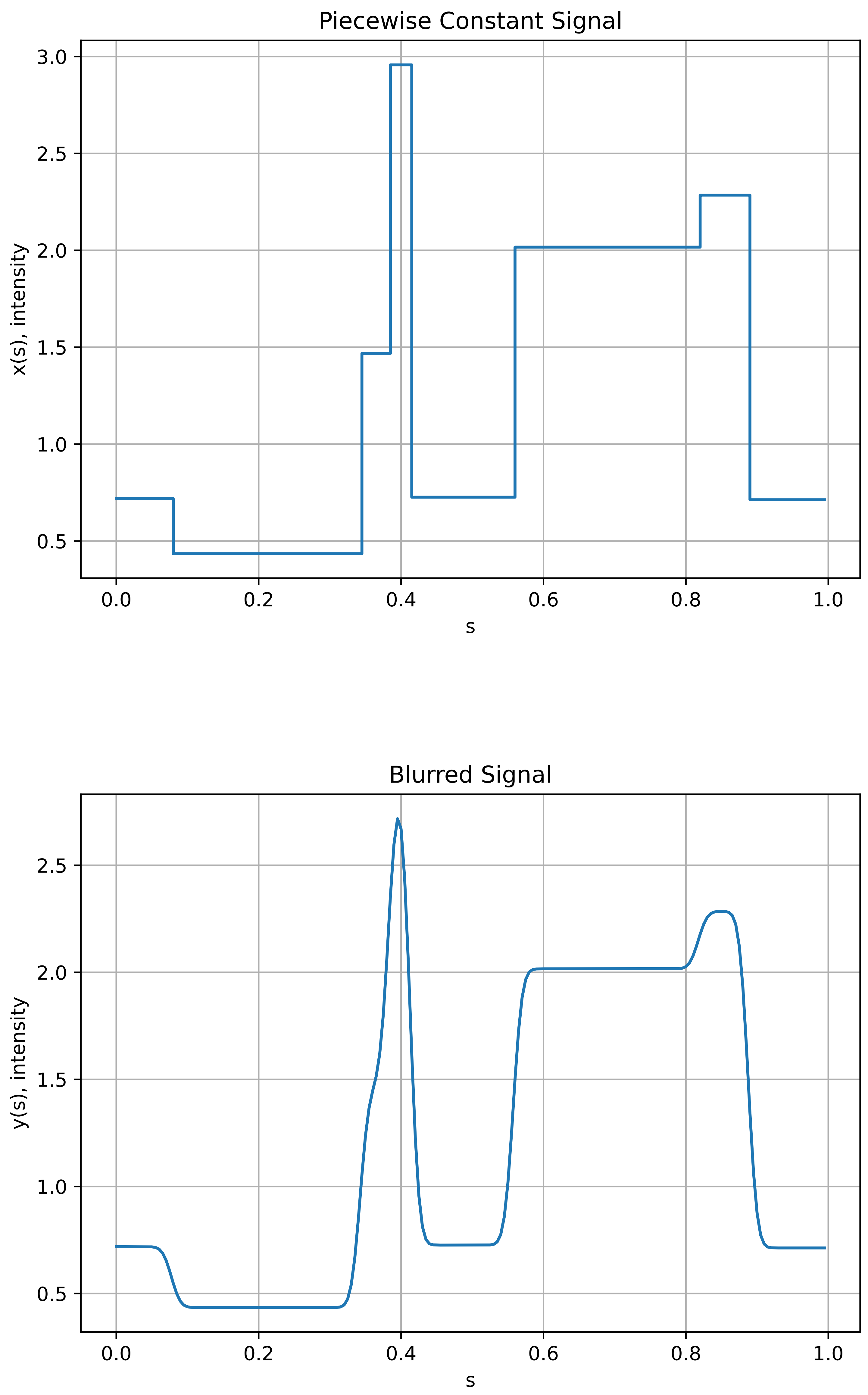
- Let $x(s)$ represent (continuous) pixel intensities or an image in 1D, with $s \in [0,1)$.

- We can express blurring using a convolution operation:

$$y(s) = (g * x)(s) := \int_0^1 g(t-s)x(t)\ dt$$

g is called a kernel and a typical kernel is a Gaussian kernel:

$$g(t) = \frac{1}{\sqrt{2\pi\delta^2}} \exp\left(-\frac{t^2}{2\delta^2}\right)$$



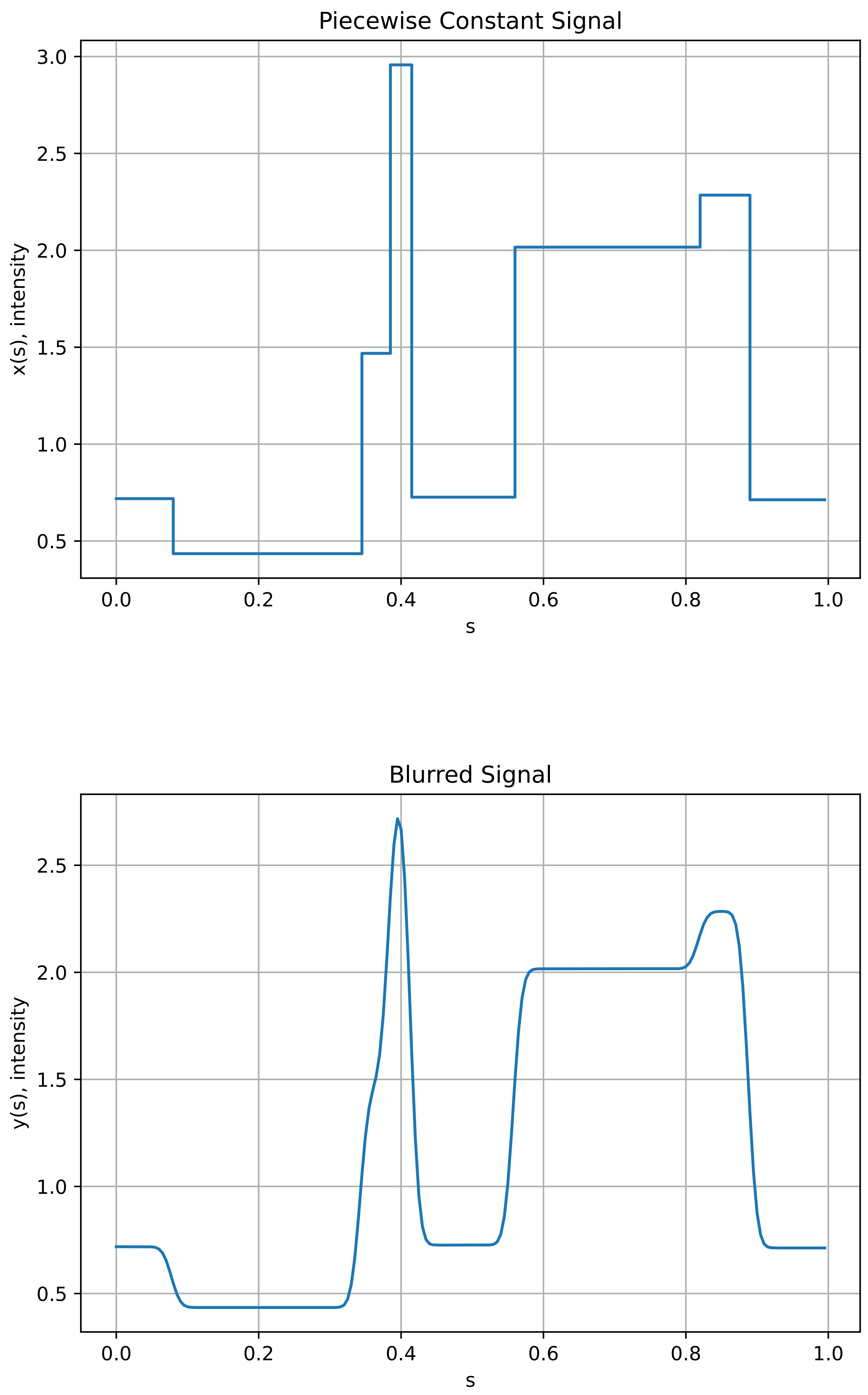
Discrete De-blurring problem

- We discretize $x(s)$ with a vector $\mathbf{x} \in \mathbb{R}^N$ with $x_i = x(i\Delta s)$, and $\Delta s = 1/N$.

- We can express discrete blurring operation as an (implicit) matrix vector multiplication:

$$\mathbf{y} = A\mathbf{x}$$

$$A \in \mathbb{R}^{N \times N}, \mathbf{y} \in \mathbb{R}^N.$$



Statistical De-Blurring

Exercise 1 (HW4)

- Complete the code in `exercise_1.py` in day 4 folder

- Discretize the unit interval with a vector s that contains the elements $i\Delta s$ with $i = 1,\ldots,100$ and $\Delta s = 1/100$. This is already in the code!

- Create a Numpy vector `x` that approximates the step signal

$$x(s) = \begin{cases} 1, & 0.2 \leq s \leq 3.5, \\ 2, & 0.5 \leq s \leq 0.7, \\ 0 & \text{otherwise.} \end{cases}$$

- Use the Python function `A` to add blurr to `x` and create blurred measurement `y`. Use blurring standard deviation δ (in the code `delta`) to be 1. Furthermore, use the suggested code to add noise to the blurred signal to create a noisy measurement `y_obs`.

- Plot the original signal `x` and the blurred measurement `y` and noisy measurements `y_obs`.

Formulating the de-blurring problem as an inverse problem

Exercise 2

- Formulate the problem as an inverse problem:
$$\mathbf{y} = \mathscr{A}\mathbf{x} + \mathbf{e}$$

- Here $\mathbf{e} \sim \mathscr{N}(0, \sigma^2 I_N)$, i.e., noise is independent and standard normal for each component of $\mathbf{y}$.

Bayesian formulation of the De-blurring problem

Posing a Bayesian inverse problem

- Define parameters as random variables:

$$Y = \mathscr{A}\mathbf{X} + E$$

  - $E \in \mathbb{R}^N$ is noise random variable.
  - $Y \in \mathbb{R}^N$ is measurement random variable.
  - $X \in \mathbb{R}^N$ is the unknown random variable.

Posing a Bayesian inverse problem

- Now we need to define the distribution for each component $E, Y|X,$, and X to use the Bayes' formula:

$$\pi_{X|Y=\mathbf{y}} \propto \pi_{Y|X}(\mathbf{y})\pi_X(\mathbf{x}).$$

  - E is a Gaussian noise with, therefore,

$$E \sim \mathcal{N}(0,\sigma^2 I_N), \qquad \pi_E(e) \propto \exp(-\frac{\|e\|_2^2}{2\sigma^2})$$

  - $Y|X$ is translation of E with AX, i.e.,

$$Y|X = \mathcal{N}(\mathcal{A}X, \sigma^2 I_N), \quad \pi_{Y|X=x}(y_{\text{obs}}) \propto \exp(-\frac{\|y_{\text{obs}} - \mathcal{A}x\|_2^2}{2\sigma^2})$$

  - The art of Bayesian inversion is to choose π_X relevant to the problem.

Gaussian i.i.d. priors

- The most basic prior we can choose is that each pixel is independent from the other pixels and follows a standard normal distribution.

- Therefore,

$$X \sim \mathcal{N}(0, I_N), \qquad \pi_X(\mathbf{x}) \propto \exp(-\frac{\|\mathbf{x}\|_2^2}{2})$$

Exercise 3

- Complete `exercise_2.py` from day4:
- Create a Python function that computes the log of (un-normalized) prior density.
- Create a Python function that computes the log of (un-normalized) likelihood density.
- Combine the two to create a Python function that computes the log of (un-normalized) posterior density.
- Use the random-walk Metropolis-Hastings algorithm to compute 10000 posterior samples. Choose the step-size c such that you have acceptance rate of 23%. [The Metropolis-Hastings must be adjusted to log]
- Plot the mean.
- For each pixel plot the variance of the samples, i.e., restrict all samples to a specific pixel and then estimate its variance.
- Plot the pixel-wise variance as a measure for uncertainty. Explain what you see.

Prior on Increments

JYVÄSKYLÄN YLIOPISTO

Prior on Increments

- Instead of specifying a prior on pixel intensities, we want to specify their relation to their neighbors, i.e,
$$\mathbf{x}_{i+1} - \mathbf{x}_i \sim \mathcal{N}(0, \alpha^2), \quad i = 1, \dots, N-1.$$
- There are 2 approaches to implement this:

1. We can define the differences matrix:
$$D = \begin{pmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -1 & 1 \end{pmatrix} \text{ and } \pi_X(\mathbf{x}) \propto \exp\left(-\frac{\|D\mathbf{x}\|_2^2}{2\sigma_{\text{prior}}^2}\right)$$
Remember boundary condition in D and prior parameters in σ_{prior}.

2. Or we can define increment variables $\mathbf{z}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$ and insert an integration matrix T in the likelihood:
$$T = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix}, \ \pi_Z(z) \propto \exp\left(-\frac{\|\mathbf{z}\|_2^2}{2\sigma_{\text{prior}}^2}\right), \ \pi_{Y|Z}(\mathbf{y}_{\text{obs}}) \propto \exp\left(-\frac{\|AT\mathbf{z} - \mathbf{y}_{\text{obs}}\|}{2\sigma^2}\right),$$
Remember boundary condition in T and prior parameters in σ_{prior}.

Prior on Increments

Exercise (HW4)

- Fill-in the exercise in `exercise_3.py` for day 4.
- Implement the Gaussian prior on increments and repeat the sampling and uncertainty quantification like the previous exercise. What difference do you observe and why?
- Where are you the most confident in reconstruction?

Prior on Increments

Sparsity in Jump

- If we know that there are only a few changes in intensity we can modify the prior on the jumps:
  - Gaussian distribution:

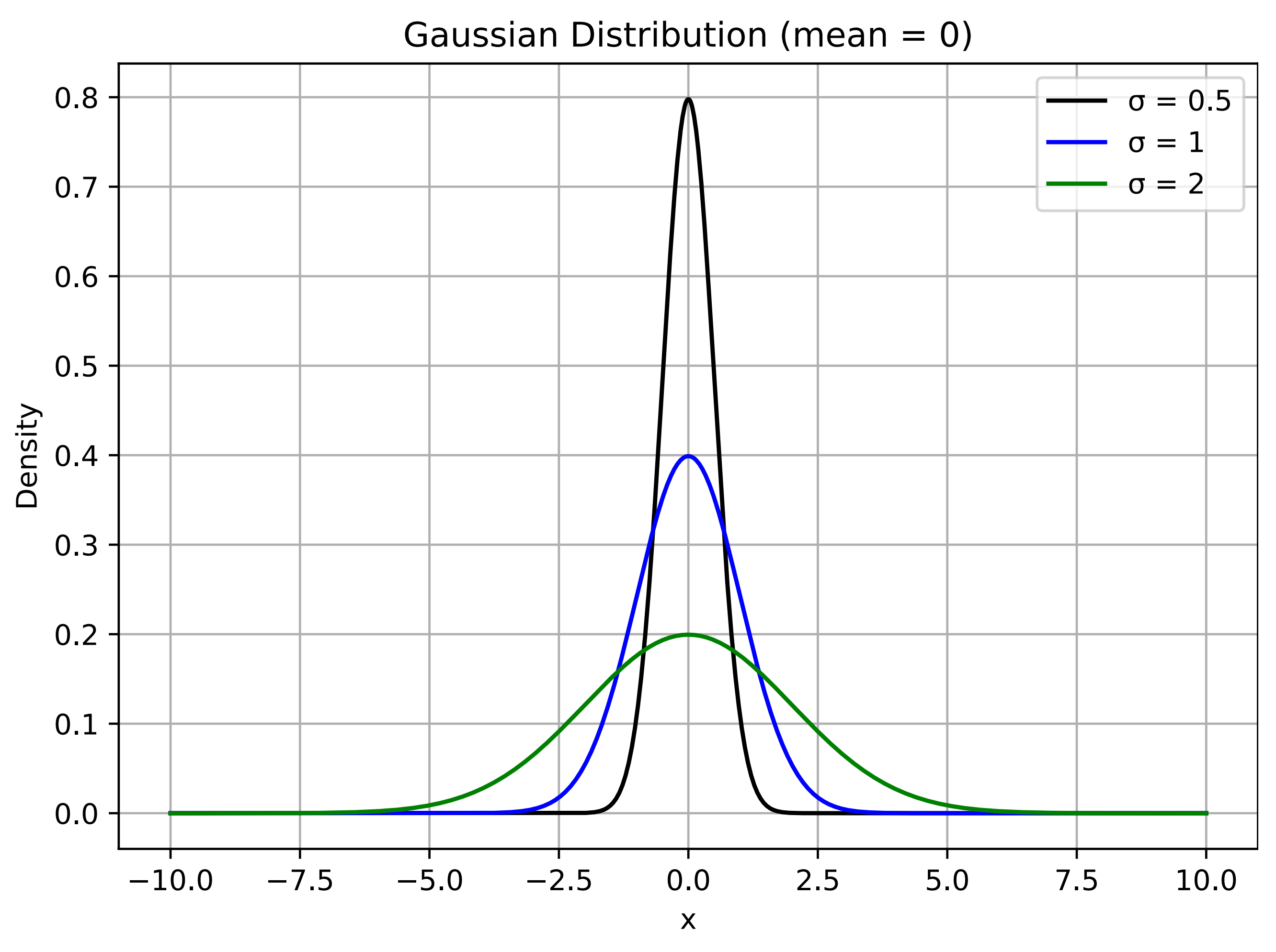
$$\pi_Z(\mathbf{z}) \propto \exp\left(-\frac{\|\mathbf{z}\|_2^2}{2\sigma_{prior}^2}\right)$$

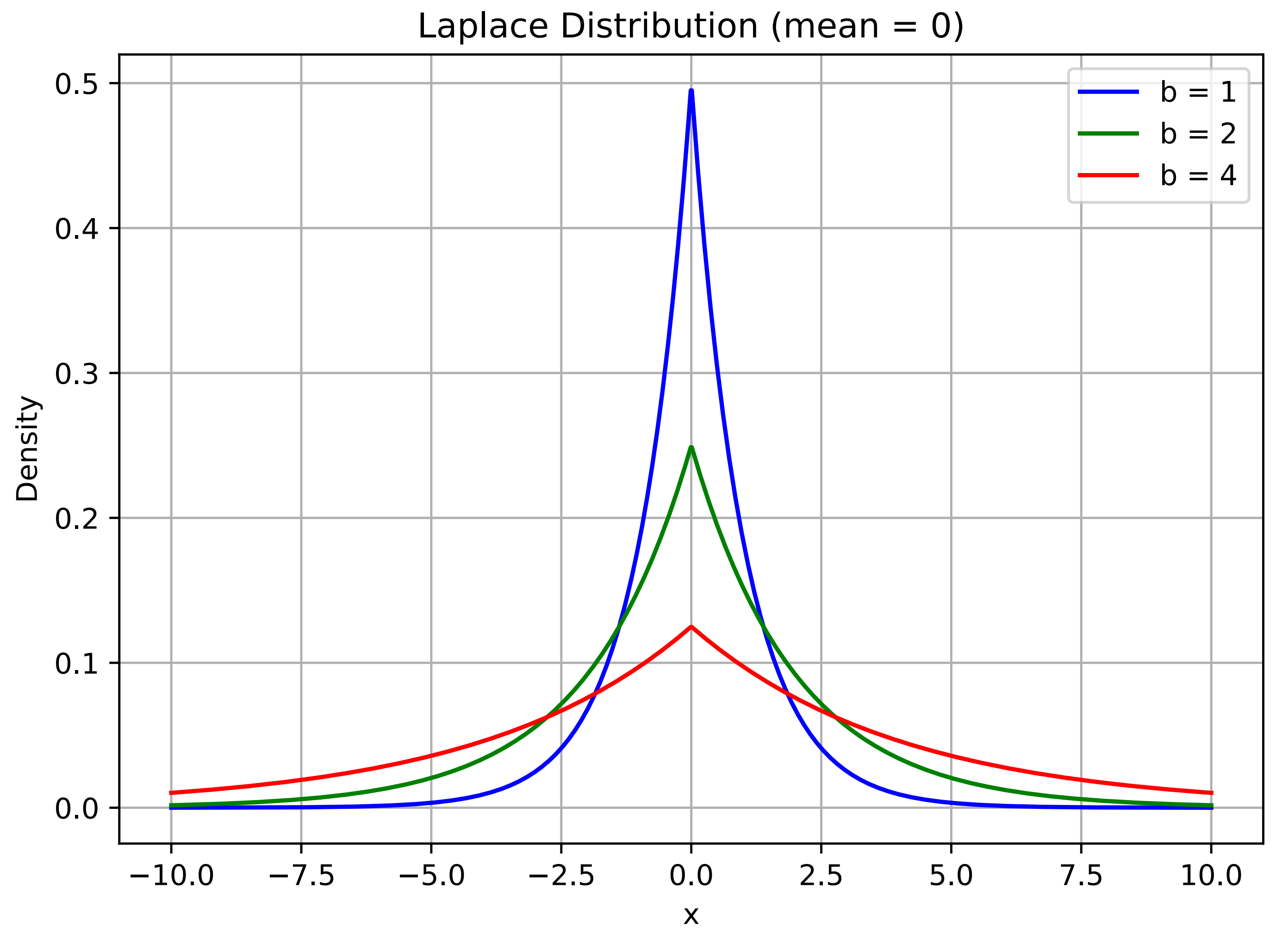
  - Laplace distribution:

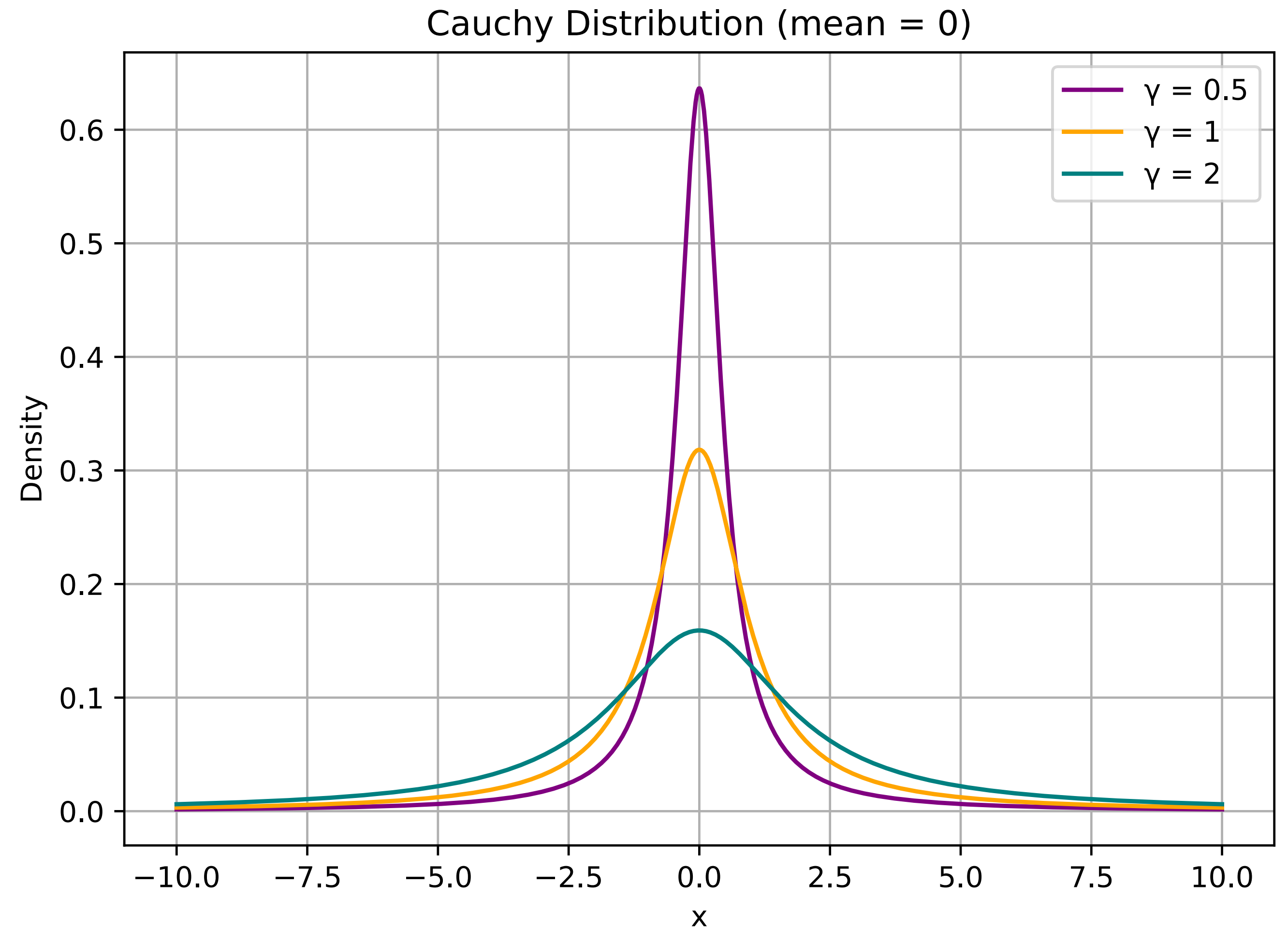
$$\pi_Z(\mathbf{z}) \propto \exp\left(-\frac{\|\mathbf{z}\|_1}{b}\right)$$

  - Cauchy distribution:

$$\pi_Z(\mathbf{z}) \propto \prod_i \frac{1}{1+\frac{\mathbf{z}_i^2}{\gamma^2}}$$







Prior on Increments

Exercise (HW4)

- Repeat the experiments with sparsity promoting Laplace and Cauchy priors. Tune the prior parameters to obtain desired characteristics in the reconstruction.

- Compare with the previous results.

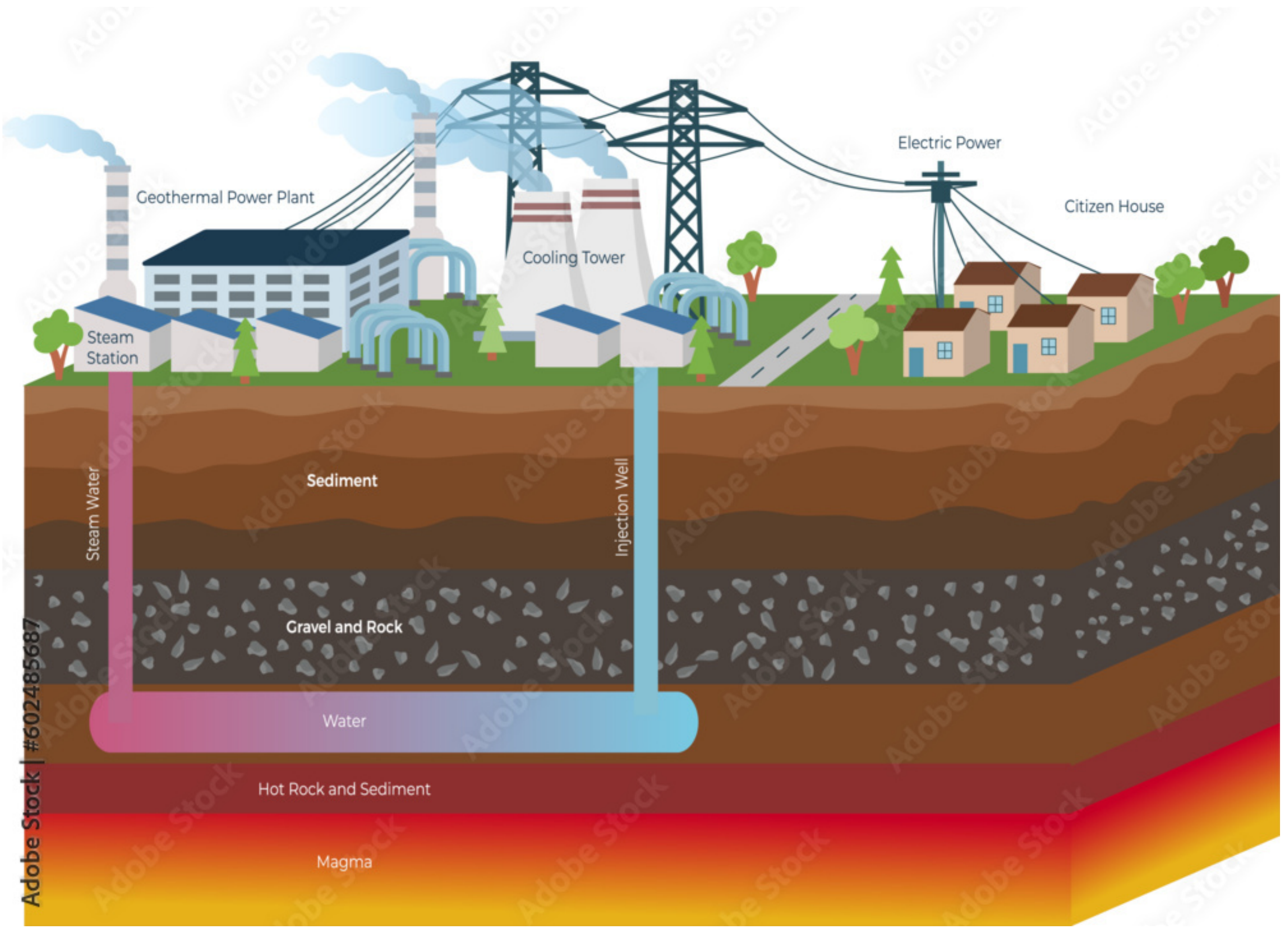
Smooth Priors and Random Fields

Gaussian and Exponential priors

Geo-thermal Power Stations

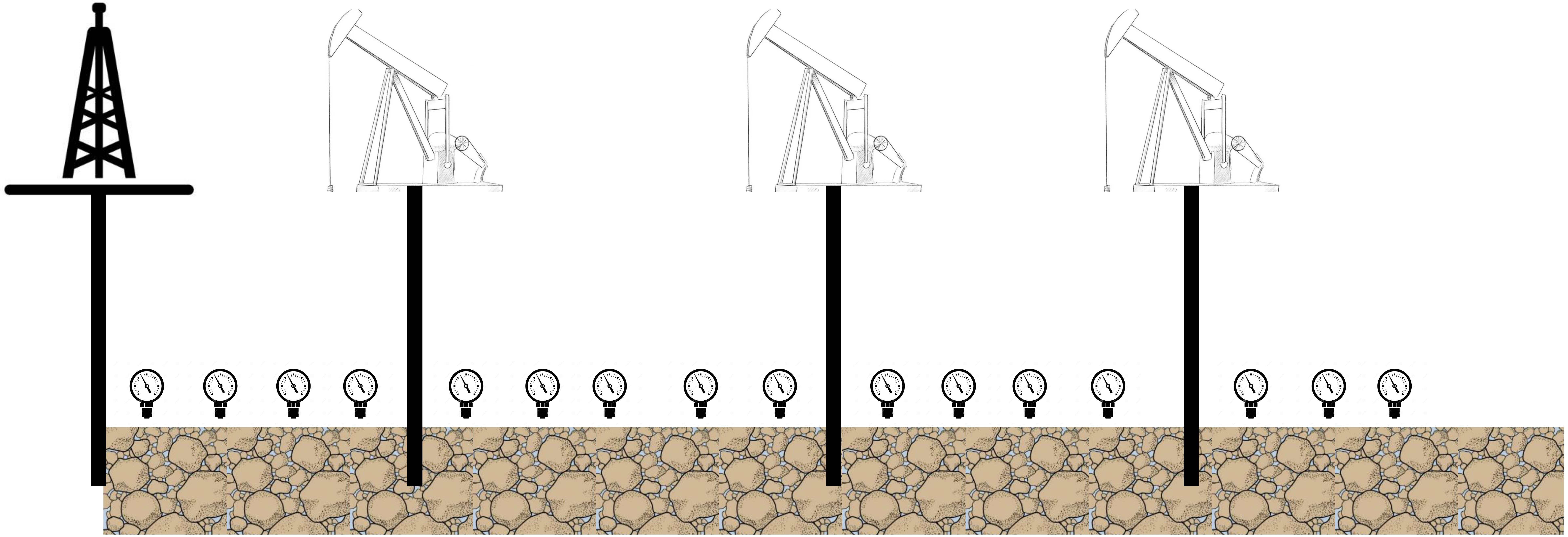
Motivation for today’s project

Krafla Power Station, Iceland



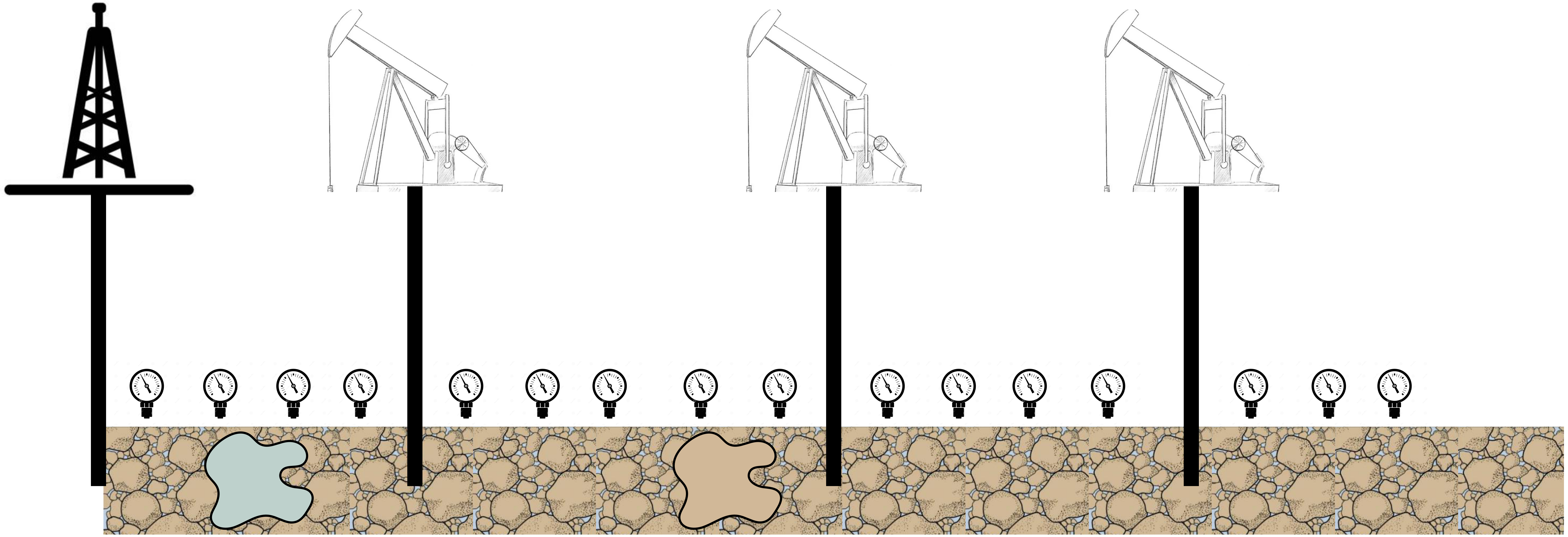
Porosity Estimation

1D flow in porous medium



Porosity Estimation

1D flow in porous medium



Smooth Priors for Bayesian Inverse Problems

The covariance function method

- We want to create a random function $X(s)$, with $s \in [0,1)$.
- We want realizations of X be *continuous* and *differentiable*.

Smooth Priors for Bayesian Inverse Problems

The covariance function method

- Let us first discretize $x(s)$, by choosing step-size $\Delta s = 1/N$.

$$x(s) \approx \mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N), \quad \mathbf{x}_i = x(i\Delta s)$$

- We can now specify how $\mathbf{x}_i$ are correlated using a covariance function:
  - Gaussian covariance function (infinitely many times differentiable):

$$f(\mathbf{x}_i, \mathbf{x}_j) = c_{ij} := \exp(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{2\ell^2})$$

  - Exponential covariance function (1-time differentiable):

$$f(\mathbf{x}_i, \mathbf{x}_j) = c_{ij} := \exp(-\frac{|\mathbf{x}_i - \mathbf{x}_j|}{\ell})$$

- ℓ is called the *length scale* and controls how far apart points have strong correlation.

Smooth Priors for Bayesian Inverse Problems

The covariance function method

- Once we have the correlations, we can create a covariance matrix:

$$C_X = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1,N} \\ c_{21} & c_{22} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N1} & c_{N2} & \cdots & c_{N,N} \end{pmatrix}$$

- We can then create a Gaussian prior for X following

$$X \sim \mathcal{N}(0, C_X), \quad \text{with} \qquad \pi_X(\mathbf{x}) \propto \exp(-\frac{1}{2}\mathbf{x}^T C_X^{-1}\mathbf{x})$$

Independent Components of X

- Since we enforced spatial correlation for components of $X(s)$ then, by definition, it's components, i.e., $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ are correlated. This makes inference really challenging!

- Theorem: If the covariance matrix C of a multi-variate random variable X is symmetric and positive-definite, then the random variable
$$Z := C^{-1/2}X$$
has independent and standard normal components, where $C^{-1/2}$ is the precision matrix, or the inverse or the Cholesky factor $C^{1/2}$

- The formulation of $X = C^{1/2}Z$ for a random function is referred to as the *Karhunen-Loève expansion of X.*

Exercise 1

- Create a covariance matrix following once the Gaussian cov. function, and once following the exponential cov. function.

- Sample random functions from the distribution $\mathcal{N}(0, C_X)$. For different length scales 0.05, 0.1, and 0.2. Explain what you see.

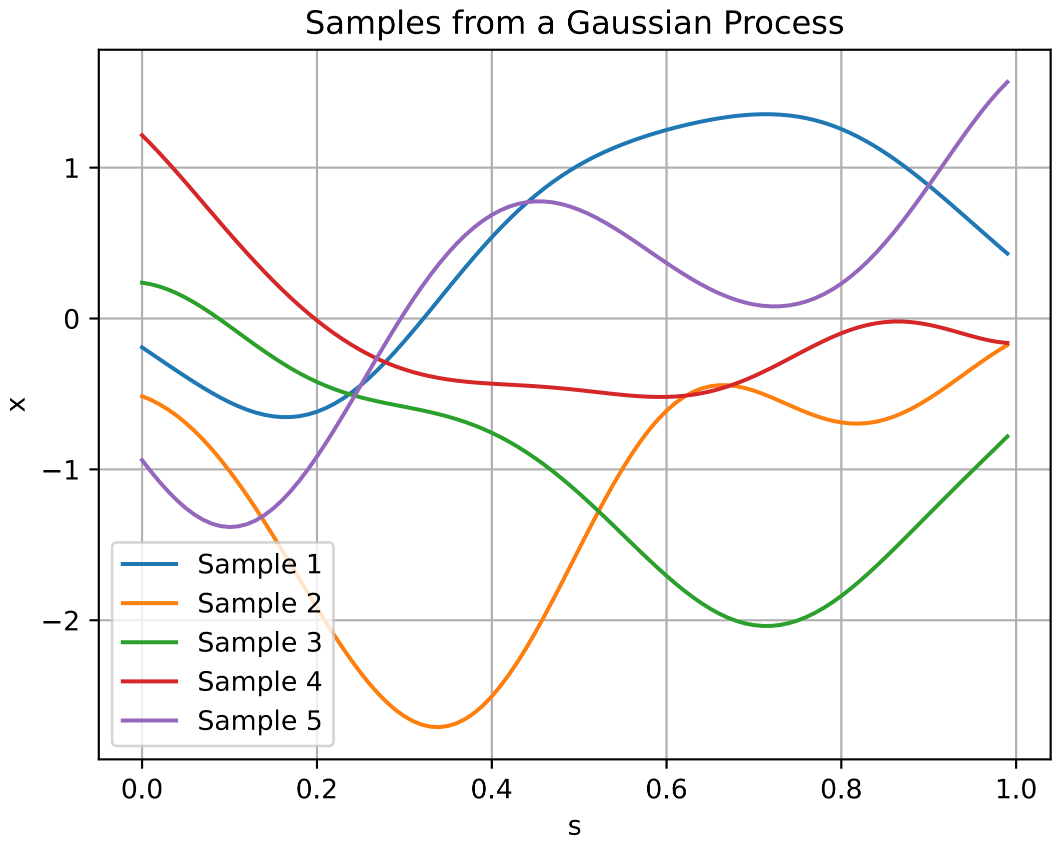
- Compute the Cholesky factor $C_X^{1/2}$.

- For the Gaussian covariance kernel, set the length scale to 0.1 and plot 5 samples from $C^{1/2}Z$, where $Z \sim \mathcal{N}(0, I_N)$.

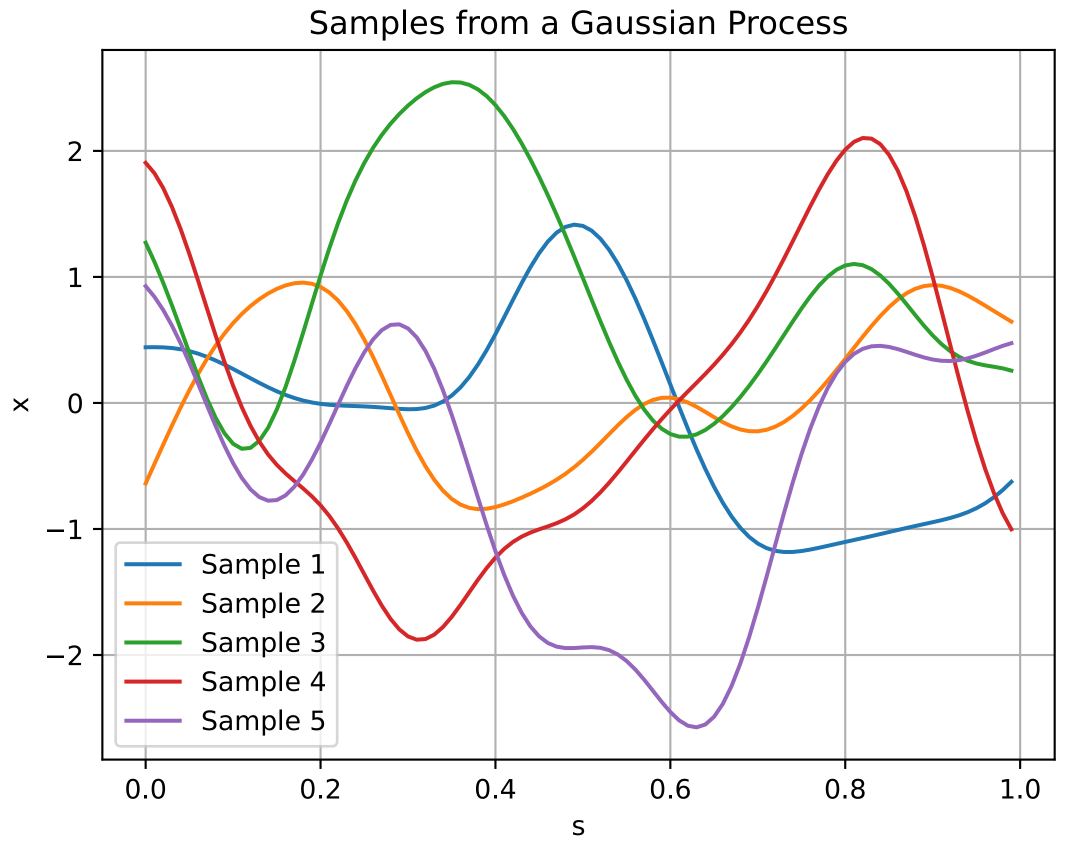
Samples from a Gaussian Random Field

Gaussian kernel

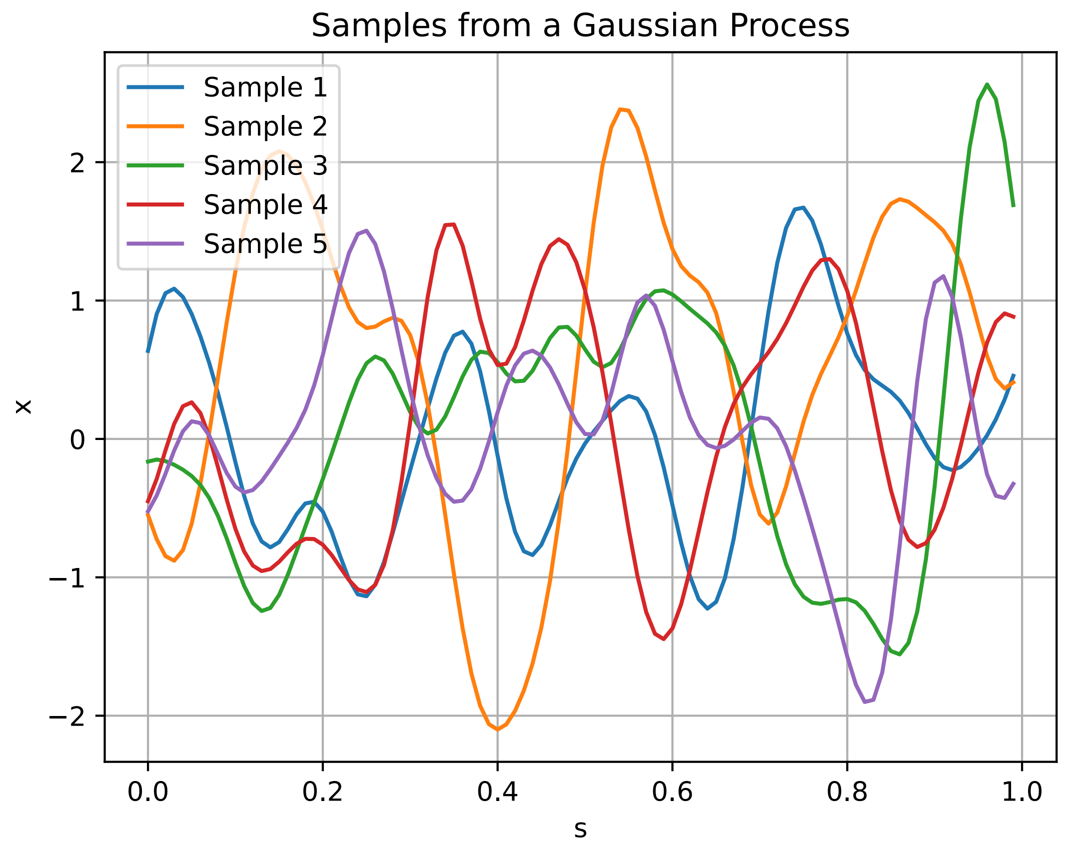
$\ell = 0.2$



$\ell = 0.1$

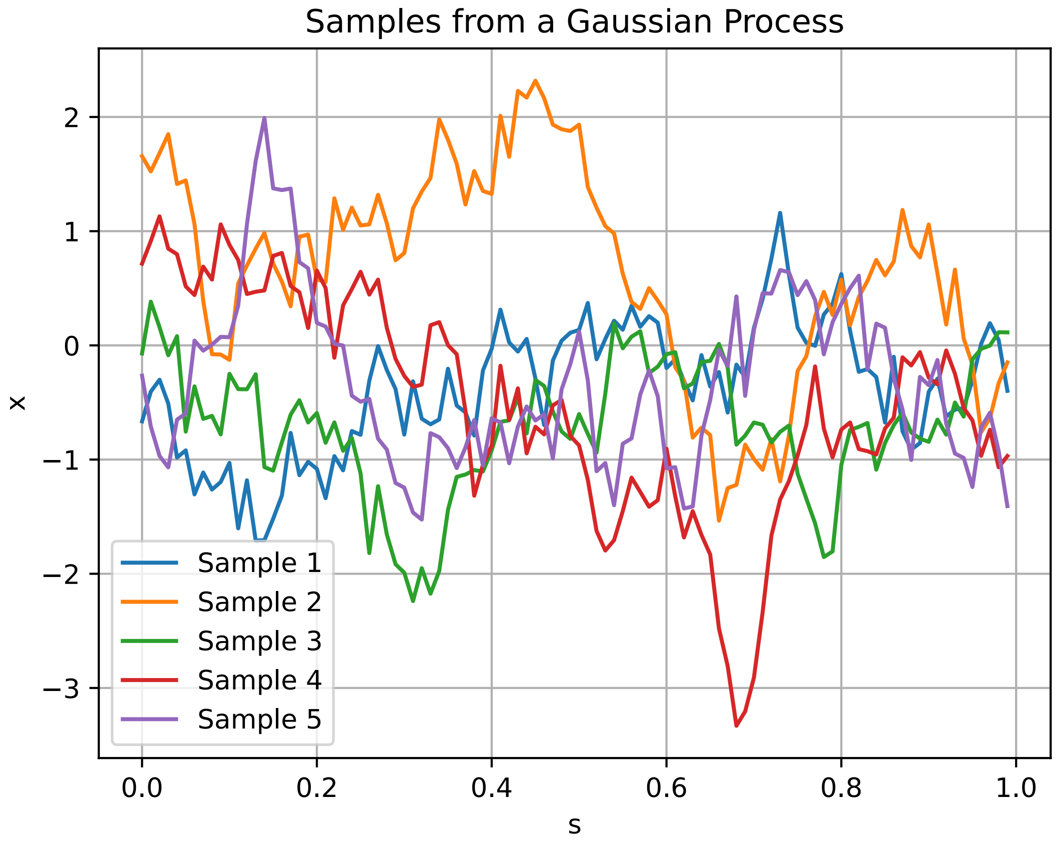


$\ell = 0.05$

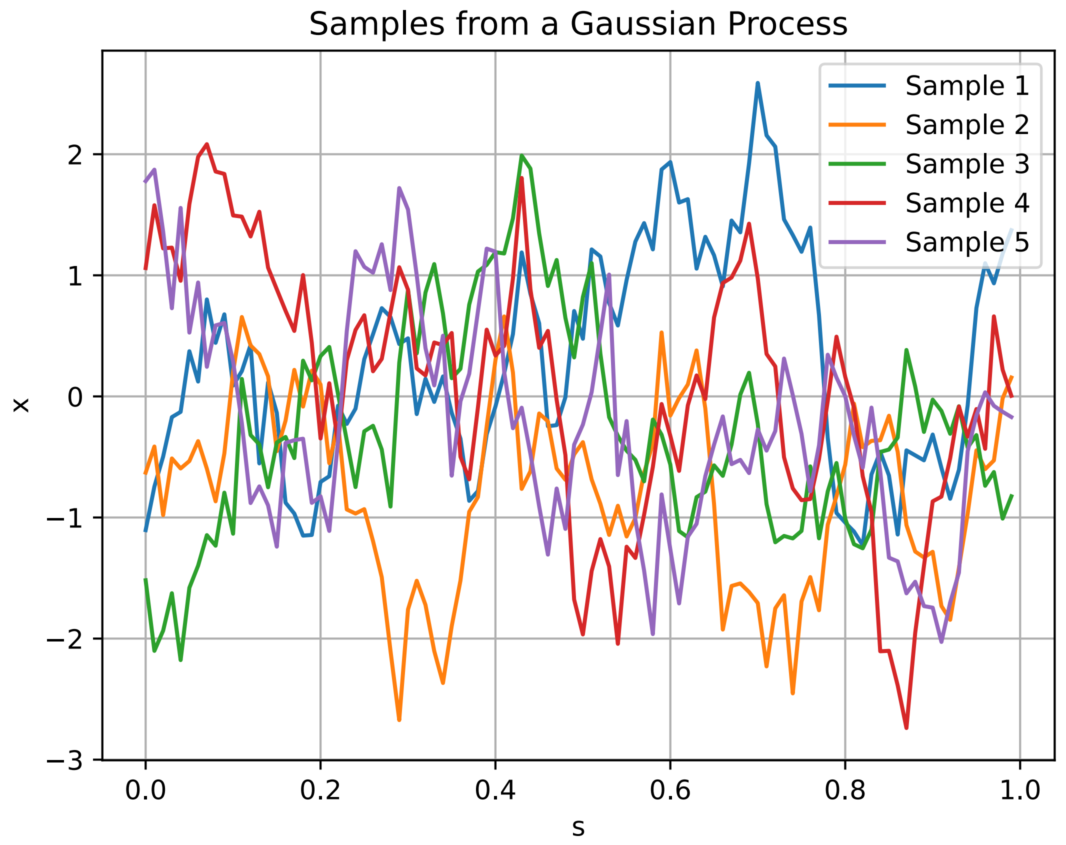


Exponential kernel

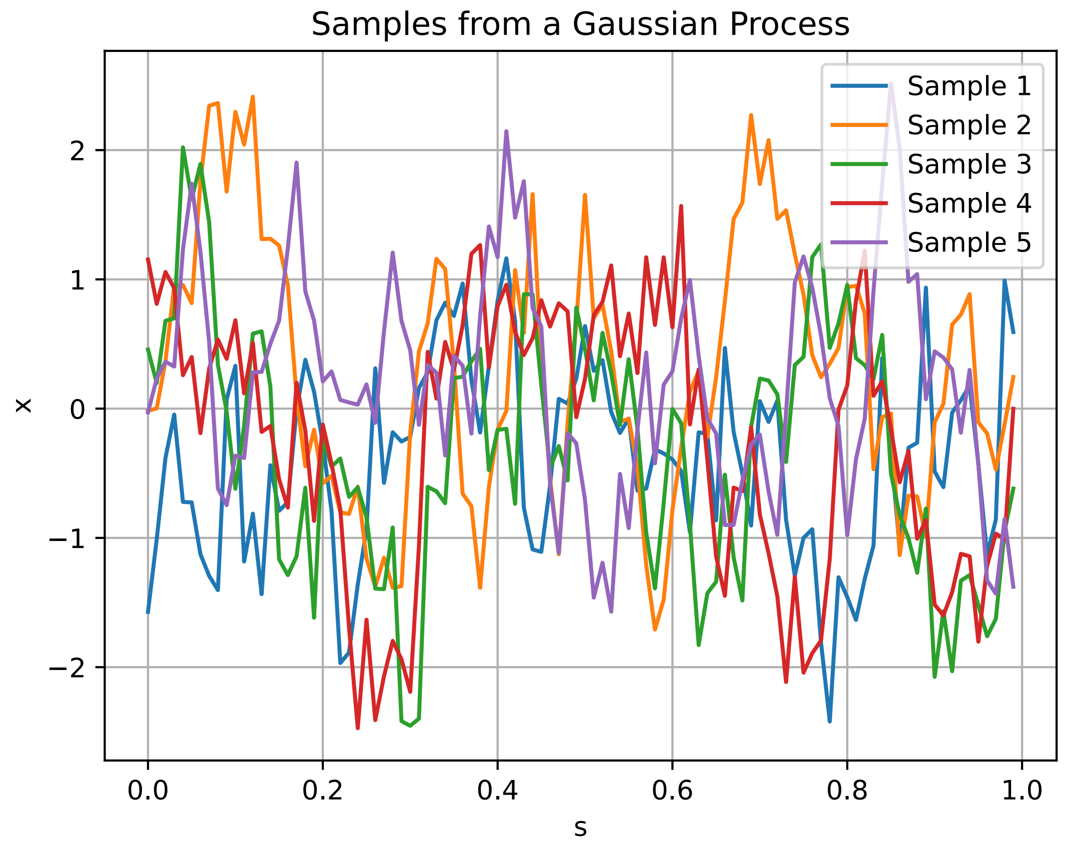
$\ell = 0.2$



$\ell = 0.1$



$\ell = 0.05$



Hydraulic Imaging

Problem formulation

- The hydraulic imaging can be formulated in terms of the PDE problem:

$$\frac{\partial}{\partial s}\left(\exp(x(s))\frac{\partial p_i(s)}{\partial s}\right) = q_i\delta(s_i^{\text{well}}), \qquad i = 1,\ldots,N_{\text{well}}$$

  - $p_i(s)$ is the pressure profile when the ith injector is active
  - q_i is the injection rate at the location of ith well
  - $\exp(x)$ is the porosity profile.
  - x is a continuous and possibly differentiable function.

Hydraulic Imaging

Notation

- We refer to $\mathbf{x}$ to be the unknown in the inverse problem.
- We refer to $\mathbf{p}_i = F_i(\mathbf{x})$ to refer to the experiment of injecting in the ith well and recording the pressure profile $\mathbf{p}_i$ (solving 1 PDE).
- We can write the inverse problem as

$$\mathbf{y}_i = F_i(\mathbf{x}) + \varepsilon_i, \quad \text{for } i = 1,\ldots,5$$

Hydraulic Imaging
Run the hydraulic problem

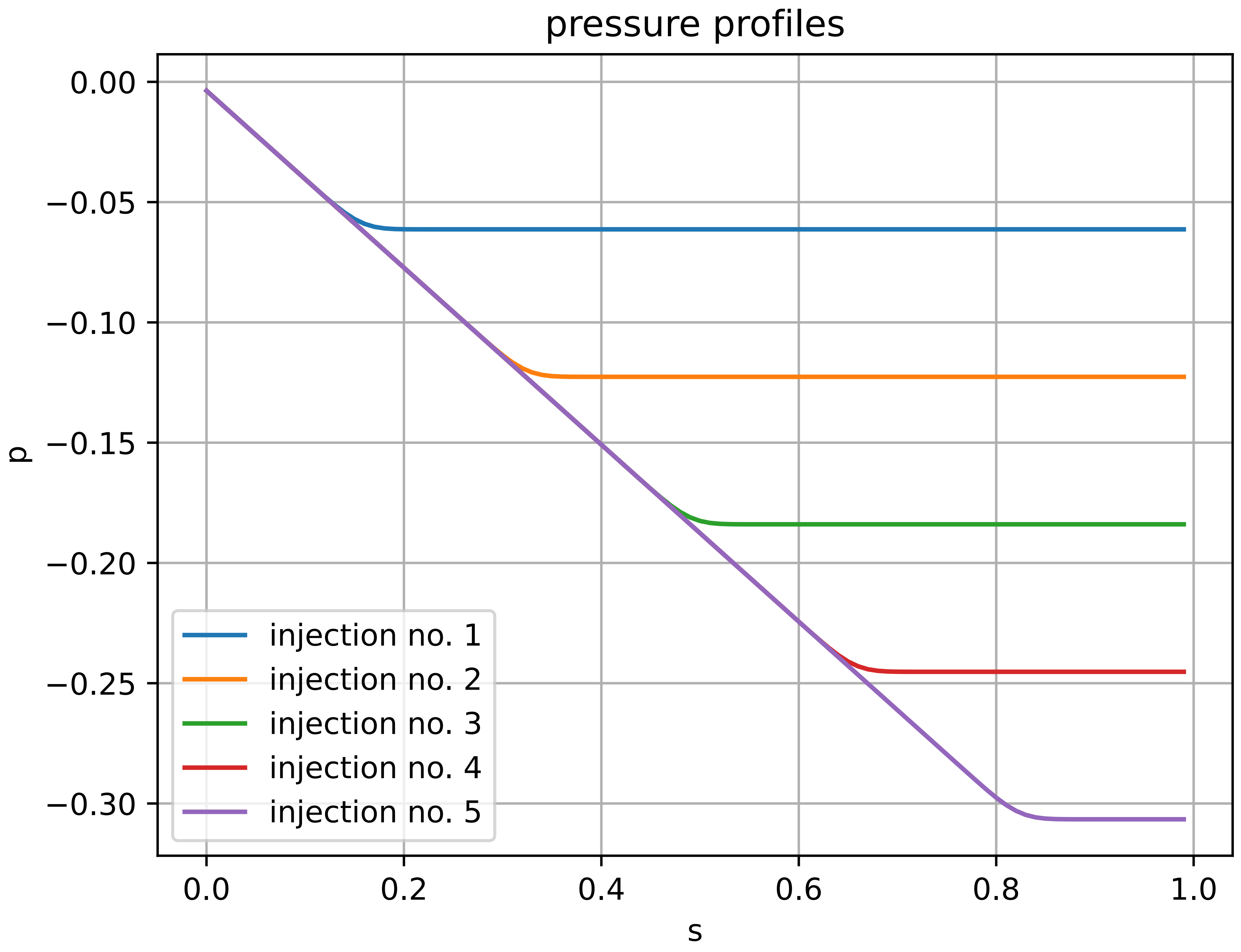
```
import numpy as np
import matplotlib.pyplot as plt

from hydraulic import hydraulic_class

N_points = 128

hydraulic = hydraulic_class(N_points) # initiate hydraulic problem with N discretization points

X = np.ones(N_points) # constant porosity profile
p = hydraulic.forward(X) # returns the pressure profile for 5 injections
```



Hydraulic Imaging

Bayesian Formulation

- First we introduce random variables:

$$\mathbf{Y}_i = F_i(\mathbf{X}) + E_i$$

- Y_i, $i = 1,\ldots,5$, are random variables of pressure profile for ith injection.

- E_i, $i = 1,\ldots,5$, are random variables of noise for ith injection.

- X is the random variable for log-porosity.

Hydraulic Imaging

Bayesian Formulation

- We now formulate the Bayes' rule

$$\pi_{X|Y_1,\ldots,Y_5} \propto \pi_{Y_1,\ldots,Y_5|X}\pi_X$$

- Under independence of measurements we can further break down the posterior

$$\pi_{X|Y_1,\ldots,Y_5} \propto \pi_{Y_1|X}\cdots\pi_{Y_5|X}\pi_X$$

- Now we need to specify each distribution on the right hand side.

Hydraulic Imaging

Bayesian Formulation

- Let us break down the terms in Bayes' rule:

$$\pi_{X|Y_1,\ldots,Y_5} \propto \pi_{Y_1|X} \cdots \pi_{Y_5|X} \pi_X$$

  - $\pi_{Y_i|X}$ is the likelihood density for the ith injection/measurement:

$$\pi_{Y_i|X}(\mathbf{y}_i^{\text{obs}}) \propto \exp(-\frac{\|F_i(\mathbf{x}) - \mathbf{y}_i^{\text{obs}}\|_2^2}{2\sigma_i^2}),$$

    where σ_i is the noise standard deviation for the ith experiment.

  - $\pi_X(\mathbf{x})$ is the prior density for a smooth function

$$\pi_X(\mathbf{x}) \propto \exp\left(-\frac{\mathbf{x}C_X^{-1}\mathbf{x}}{2}\right)$$

Hydraulic Imaging

Bayesian Formulation with Reparameterization

- Let us break down the terms in Bayes' rule:

$$\pi_{Z|Y_1,\ldots,Y_5} \propto \pi_{Y_1|Z} \cdots \pi_{Y_5|Z} \pi_Z$$

  - $\pi_{Y_i|X}$ is the likelihood density for the ith injection/measurement:

$$\pi_{Y_i|Z}(\mathbf{y}_i^{\text{obs}}) \propto \exp\left(-\frac{\|F_i(C_X^{1/2}\mathbf{z}) - \mathbf{y}_i^{\text{obs}}\|_2^2}{2\sigma_i^2}\right),$$

    where σ_i is the noise standard deviation for the ith experiment.

  - $\pi_Z(\mathbf{z})$ is the prior density for a smooth function

$$\pi_Z(\mathbf{z}) \propto \exp\left(-\frac{\|\mathbf{z}\|_2^2}{2}\right)$$

Exercise 2

- Write a Python function that computes the log-prior density corresponding to a Gaussian covariance function with correlation length $\ell = 0.1$.
- Write a Python function that computes the log-likelihood density function which combines the 5 experiments for the hydraulic imaging problem.
- Combine the two to write a Python function that computes the log-posterior of the hydraulic imaging problem.
- Use the random-walk Metropolis-Hasting algorithm to sample from the posterior distribution.
- Plot the posterior mean and also point-wise variance.