Introduction to Bayesian Analysis

Irene Ji, Duke University

Simon Mak, Duke University

JETSCAPE Online Summer School 2022

(partially adapted from a BAND camp talk (ISNET 2020) by Simon Mak and Derek Everett;

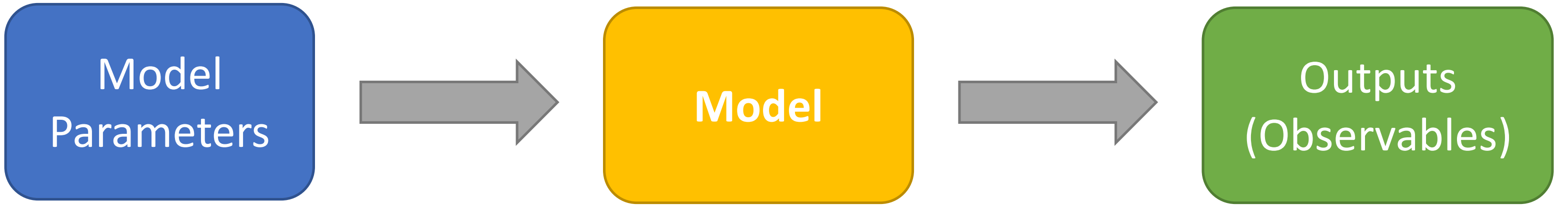
and a Bayesian Inference talk (JETSCAPE Summer School 2021) by Matthew Heffernan, source)

Section I.
The Bayesian Paradigm

Outline

- **Forward Problem**
- **Inverse Problem**
- Bayes Rule

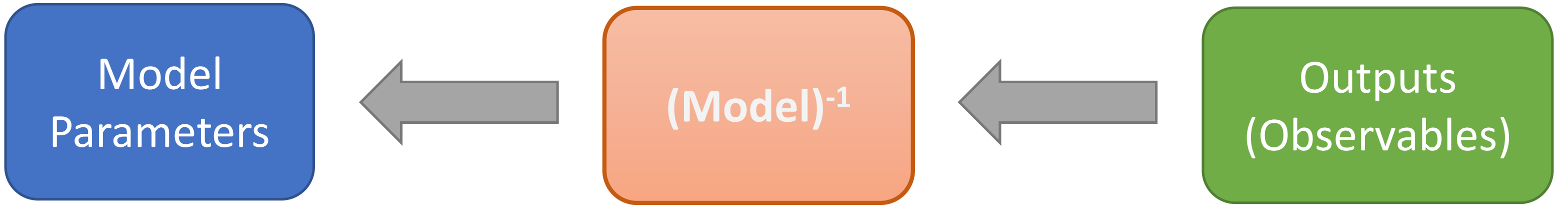
Forward Problem



- **Model**: theoretical descriptions of the relevant processes
- **Model Parameters**: inputs for the model
- **Outputs (Observables)**: outputs generated from the model

- **Forward problem:**
  - What are the **model outputs** for **given set of model parameters**?

Inverse Problem



Model Parameters

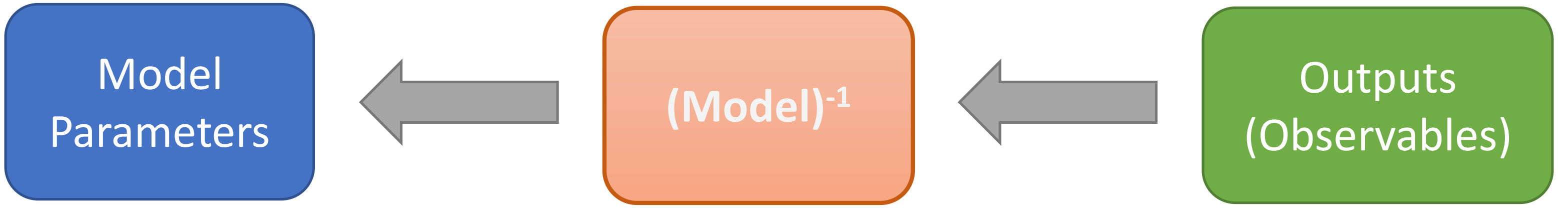
$(\text{Model})^{-1}$

Outputs (Observables)

- **Model**: theoretical descriptions of the relevant processes
- **Model Parameters**: inputs for the model
- **Outputs (Observables)**: outputs generated from the model

- **Inverse problem:**
  - What are the **model parameters** that result in **given set of model outputs**?

Inverse Problem

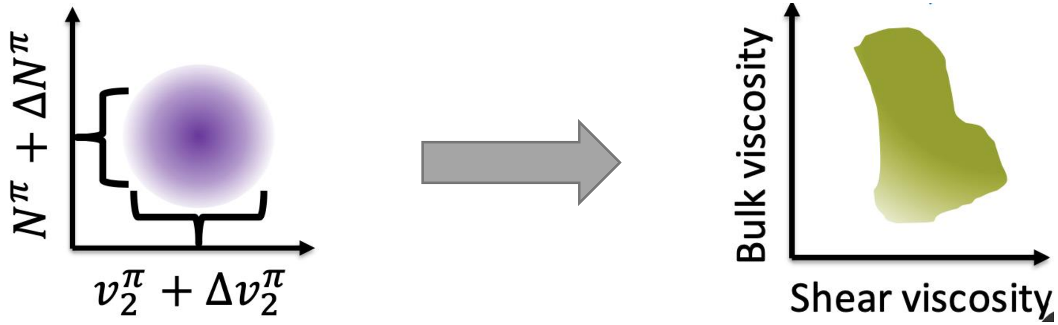


Model Parameters

(Model)$^{-1}$

Outputs (Observables)

- For **noisy** observables:



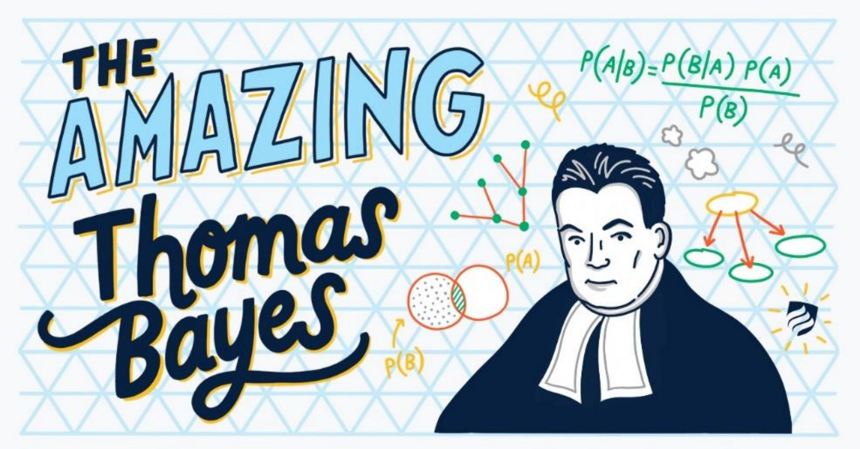
- Observed probability distribution ⇒ Distribution of parameters

Outline

- Forward Problem
- Inverse Problem
- **Bayes' Rule**

Bayes' Rule

$$\boldsymbol{p(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}}$$



https://www.elmhurst.edu/blog/thomas-bayes/

- $p(A)$: probability of A (or degree of belief in A)
- $p(B|A)$: probability of B given A
- $p(B)$: probability of B
- $p(A|B)$: probability of A given B (target)

Bayes' Rule: A Simple Example

Question:

- 40% of all rainy days have cloudy mornings. $p(B|A) = 0.4$
- In City X, probability of rainy days is 0.1. $p(A) = 0.1$
- In City X, probability of cloudy mornings is 0.2. $p(B) = 0.2$
- This morning in City X is cloudy, what's the probability of rain? $p(A|B) = ?$

Solution:

- Event A = "rain"
- Event B = "cloudy morning"
- $p(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)} = \frac{0.1 \cdot 0.4}{0.2} = 0.2$

Bayes' Rule for Parameter Estimation

- **Goal**: Learn unknown **model parameters** θ from **observables** y
- Two **ingredients**:
  - **Prior**: distribution $\pi(\theta)$ capturing **prior beliefs** on θ
  - **Likelihood**: function $f(y|\theta)$ describing probability (or density) of **observing** y given unknowns θ

Apply Bayes rule:

$$\pi(\theta|y) = \frac{f(y|\theta)\pi(\theta)}{f(y)} = \frac{f(y|\theta)\pi(\theta)}{\int f(y|\theta)\pi(\theta)d\theta}$$

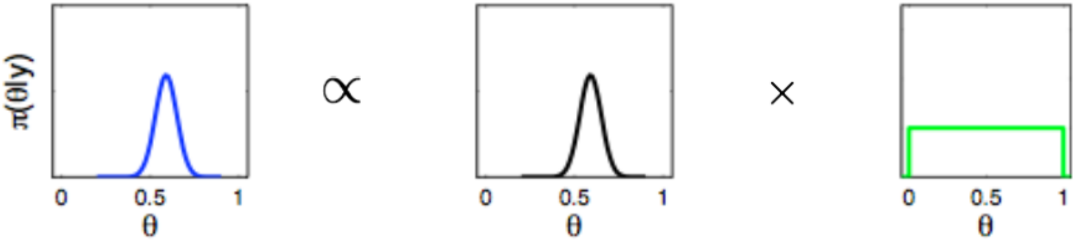
- $\pi(\theta|y)$: the **posterior** distribution, capturing our posterior **beliefs** on θ given **observed** data

Bayes' Rule for Parameter Estimation

Bayes rule:

$$\pi(\theta|y) = \frac{f(y|\theta)\pi(\theta)}{\int f(y|\theta)\pi(\theta)d\theta}$$

- **Normalizing constant** $\int f(y|\theta)\pi(\theta)d\theta$ typically **not known**
- Luckily, this is not necessary for **sampling** from $\pi(\theta|y)$ - we will use the following to perform **inference** on θ



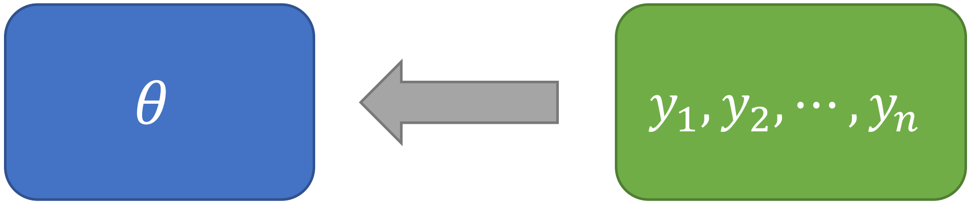
$$\pi\left(\theta|y\right) \propto f\left(y|\theta\right) \times \pi\left(\theta\right)$$

Section II.
Bayesian Parameter Estimation

Outline

- A Simple Example
- Markov Chain Monte Carlo (MCMC)
- Bayesian Parameter Estimation

A Simple Example



θ

$y_1, y_2, \cdots, y_n$

Suppose we observe **data** from the following **model**:

$$y_i = \theta + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

Two **ingredients**:

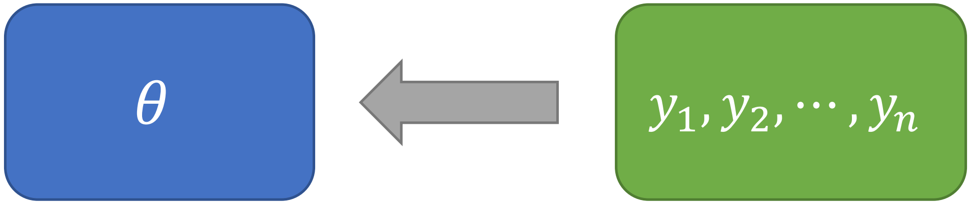
- **Prior** (prior belief on parameter θ before observing data):

$$\theta \sim N(\mu, \delta^2) \quad \Rightarrow \quad \pi(\theta) \propto \exp\left\{-\frac{1}{2\delta^2}(\theta - \mu)^2\right\}$$

- **Likelihood** (“probability” of observing data $y_1, \cdots, y_n$ given θ):

$$f(y_1, \cdots, y_n | \theta) = \prod_{i=1}^{n} f(y_i | \theta) \propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \theta)^2\right\}$$

A Simple Example



θ

$y_1, y_2, \cdots, y_n$

Suppose we observe **data** from the following **model**:

$$y_i = \theta + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

Then, by **Bayes' rule** (and some algebra), the **posterior** becomes:

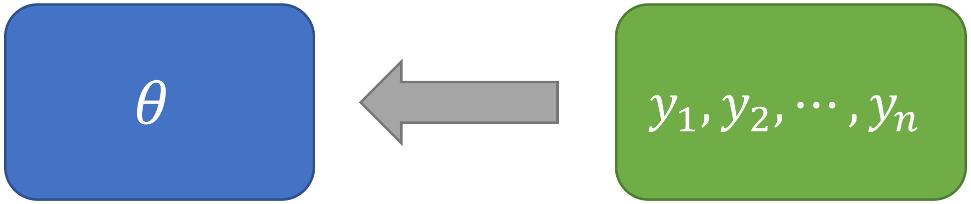
$$\pi(\theta | y_1, \cdots, y_n) \propto f(y_1, \cdots, y_n | \theta) \cdot \pi(\theta) \propto \exp\left\{-\frac{1}{2{\delta^*}^2}(\theta - \mu^*)^2\right\}$$

where:

$$\mu^* = w\bar{y} + (1 - w)\mu, \qquad w = \frac{n\sigma^{-2}}{n\sigma^{-2} + \delta^{-2}}$$

$${\delta^*}^2 = (n\sigma^{-2} + \delta^{-2})^{-1}$$

A Simple Example



θ

$y_1, y_2, \cdots, y_n$

Suppose we observe **data** from the following **model**:

$$y_i = \theta + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

Then, by **Bayes' rule** (and some algebra), the **posterior** becomes:

$$\theta | y_1, \cdots, y_n \sim N(\mu^*, {\delta^*}^2)$$

where:

$$\mu^* = w\bar{y} + (1 - w)\mu, \qquad w = \frac{n\sigma^{-2}}{n\sigma^{-2} + \delta^{-2}}$$

$${\delta^*}^2 = (n\sigma^{-2} + \delta^{-2})^{-1}$$

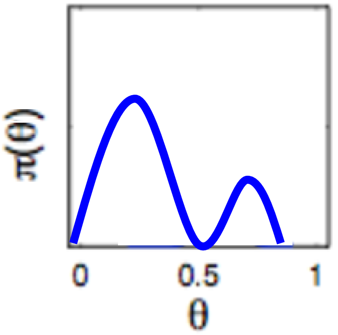
… so we can easily **sample** $\pi(\theta | y_1, \cdots, y_n)$ for parameter estimation!

Markov Chain Monte Carlo (MCMC)

Let's take the same observation **model**:

$$y_i = \theta + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

What happens if we use a more **general prior**?



- ... then the **posterior** $\theta | y_1, \cdots, y_n$ is **not** known and cannot be directly sampled **(no closed-form)**

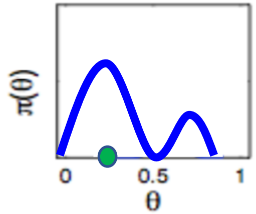
- Use **Markov Chain Monte Carlo (MCMC)** to sample:

$$\pi(\theta | y_1, \cdots, y_n) \propto f(y_1, \cdots, y_n | \theta) \cdot \pi(\theta)$$

Markov Chain Monte Carlo (MCMC)

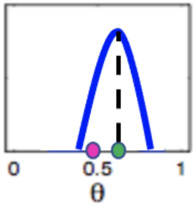
A popular **MCMC** sampling algorithm is **Metropolis-Hasting**:

1. **Initialization**: draw a sample θ_0 from **prior** distribution:



2. **At iteration *t***, draw a new point θ^* from **proposal** distribution:

- E.g., $\mathrm{p}(\theta^*|\theta_{t-1}) \sim N(\theta_{t-1}, \sigma^2)$

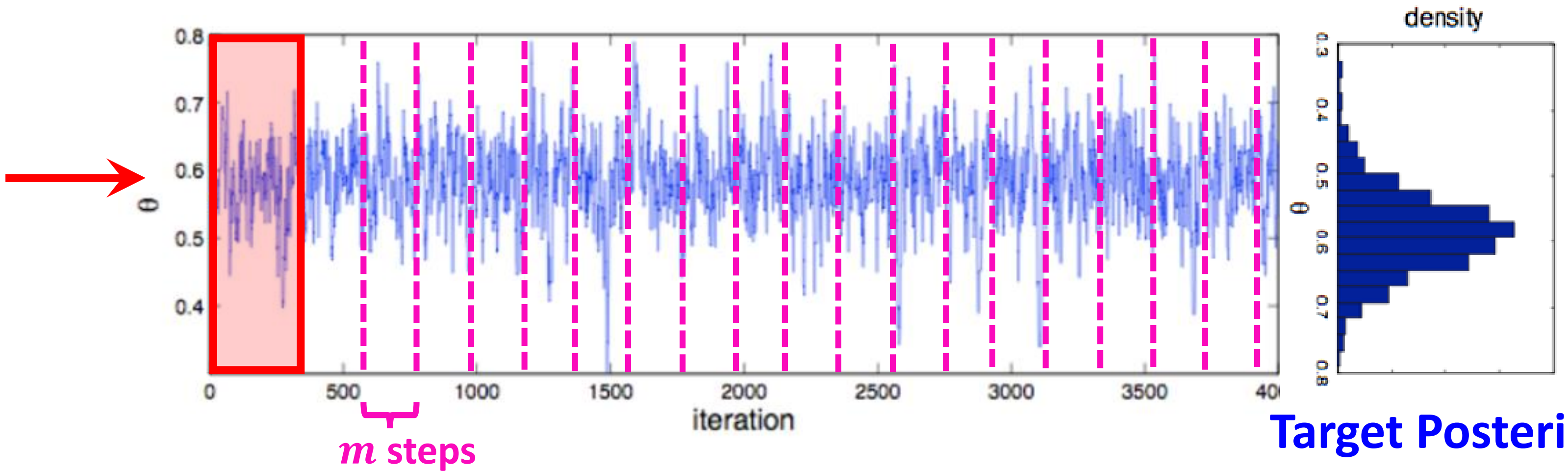


3. **Accept** the proposal $\theta_t = \theta^*$ with **probability:**

- $\alpha = min(1, \dfrac{\pi(\theta^*|y_1, \cdots, y_n)}{\pi(\theta_{t-1}|y_1, \cdots, y_n)} \cdot \dfrac{p(\theta_{t-1}|\theta^*)}{p(\theta^*|\theta_{t-1})})$
- Otherwise: $\theta_t = \theta_{t-1}$

Markov Chain Monte Carlo (MCMC)

4. **Steps 1-3** construct a **Markov chain** which samples the **posterior** using only **evaluations** of $\pi(\theta|y_1,\cdots,y_n)$:



m steps

Target Posterior

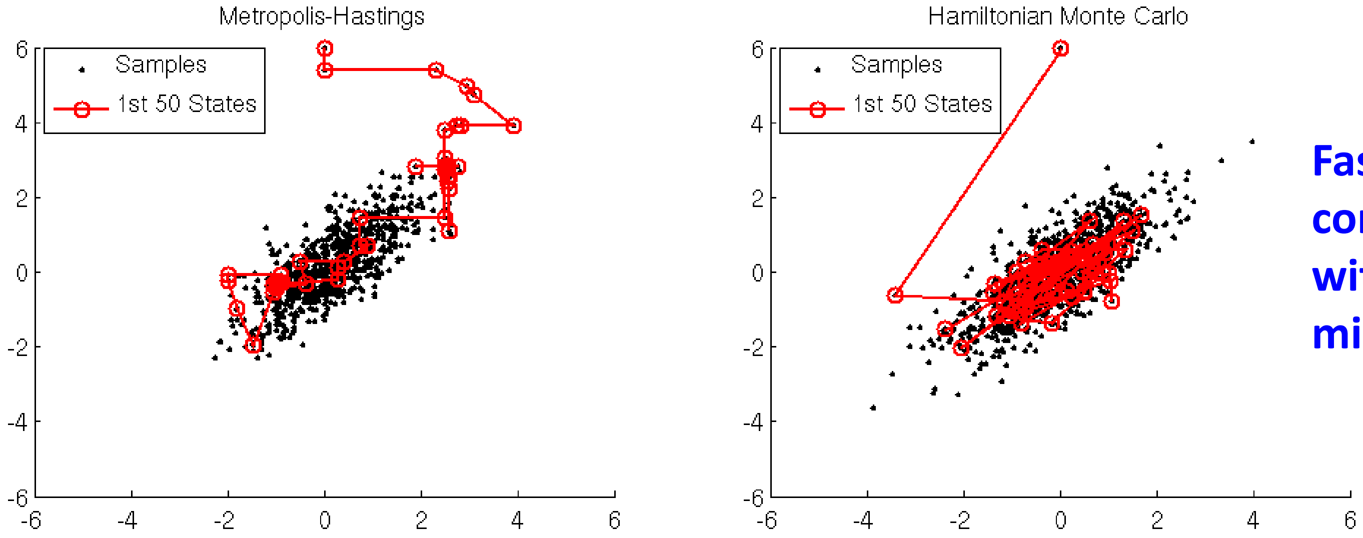
5. **Burn-in:** if initialized **poorly**, **remove** a few iterations at the start of the chain.

6. **Thinning (m-step):** if the samples are **highly correlated**, **keep** every **m-th** iteration and discard others.

Markov Chain Monte Carlo (MCMC)

Other MCMC sampling algorithms with better performance:

- **Hamiltonian Monte Carlo** (HMC)
  - Use **Hamiltonian dynamics** to create **proposal** distribution
  - $H(x,p) = U(x) + K(p)$, (Total energy = Potential + kinetic Energy)



Faster convergence with better mixing

  - Reference: https://www.mcmchandbook.net/HandbookChapter5.pdf

Figures: https://theclevermachine.wordpress.com/2012/11/18/mcmc-hamiltonian-monte-carlo-a-k-a-hybrid-monte-carlo/

Markov Chain Monte Carlo (MCMC)

Other MCMC sampling algorithms with better performance:

- **Affine-invariant** ensemble sampler for MCMC
  - Performs **affine transformation**
  - **Better performance** in general
  - Implemented in Python package "**emcee**" (Hands-on Session): https://emcee.readthedocs.io/en/stable/
  - Reference: https://arxiv.org/pdf/1202.3665.pdf

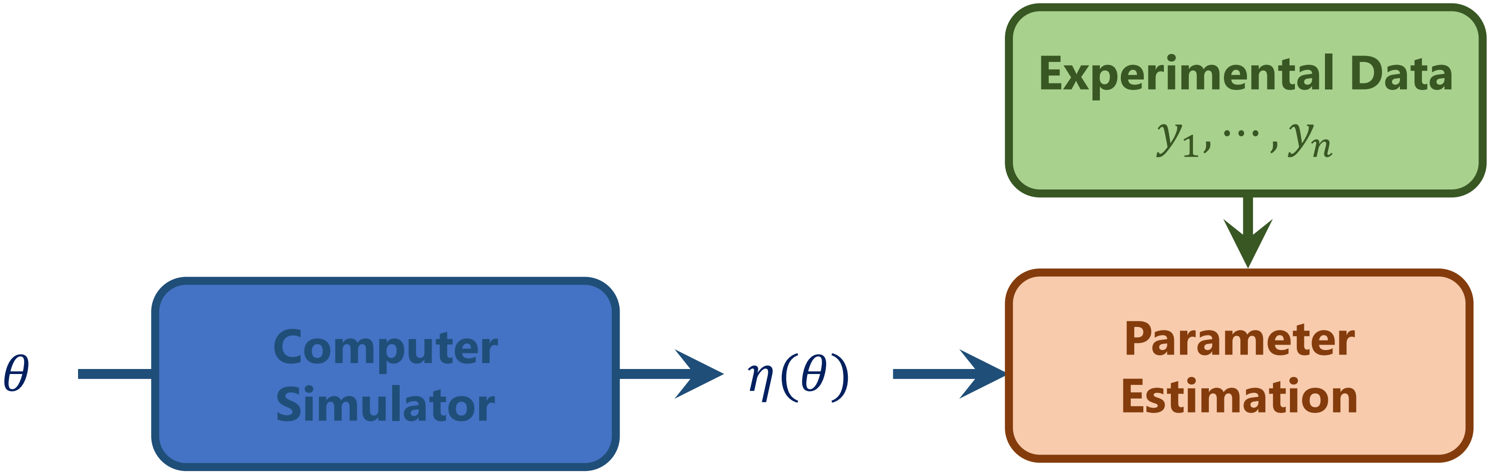
- **Parallel-tempered** MCMC
  - Good performance for **multi-modal** posterior distribution
  - Implemented in JETSCAPE-SIMS package, Weiyao will discuss this tomorrow
  - Reference: https://emcee.readthedocs.io/en/v2.2.1/user/pt/

Parameter Estimation

A more **realistic** observation model:

$$y_i = \eta(\theta) + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

where $\eta(\theta)$ is the **computer model** output with input **parameters** θ



Parameter Estimation

A more **realistic** observation model:

$$y_i = \eta(\theta) + \epsilon_i, \qquad \epsilon_i \sim^{i.i.d.} N(0, \sigma^2), \qquad i = 1, \cdots, n$$

- **Prior**: $\pi(\theta)$
- **Likelihood**:

$$f(y_1, \cdots, y_n | \theta) = \prod_{i=1}^{n} f(y_i|\theta) \propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \eta(\theta))^2\right\}$$

… then sample the **posterior** using MCMC:

$$\pi(\theta|y_1, \cdots, y_n) \propto f(y_1, \cdots, y_n|\theta) \cdot \pi(\theta) = \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i - \eta(\theta)\big)^2\right\}\pi(\theta)$$

A More Realistic Model

Posterior distribution:

$$\pi(\theta|y_1, \cdots, y_n) \propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\big(y_i - \eta(\theta)\big)^2\right\}\pi(\theta)$$

Computer model $\eta(\theta)$ **embedded** within **MCMC** sampler

- This is **fine** if the computer model $\eta(\theta)$ can be evaluated **quickly** for each θ (i.e., the simulator is computationally **cheap**)

- When the simulator is **expensive**, each evaluation of $\eta(\theta)$ is **time-intensive**. The MCMC would take a **long time** to run, since **each** sample requires an evaluation of $\eta(\theta)$!

Enter **model emulation!**

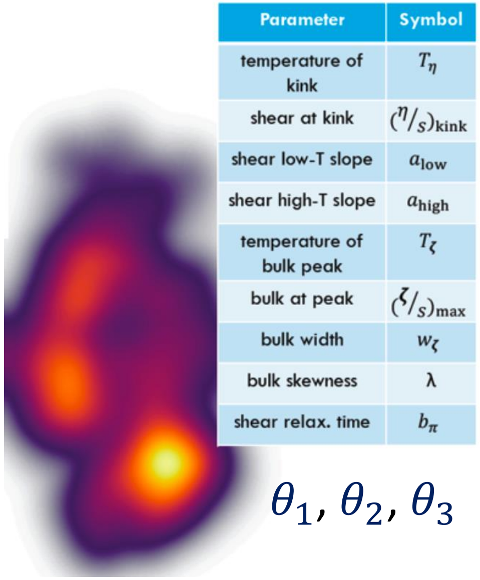
Section III.
Computer Model Emulation

Outline

- **Model Emulation**
  - **Gaussian Process**
- Multiple Observables
  - Principal Component Analysis
- Bayesian Inference Workflow

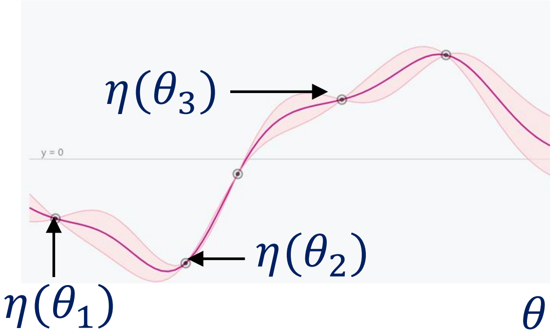
Model Emulation

One way to **speed up** computer simulations is via **model emulation**:

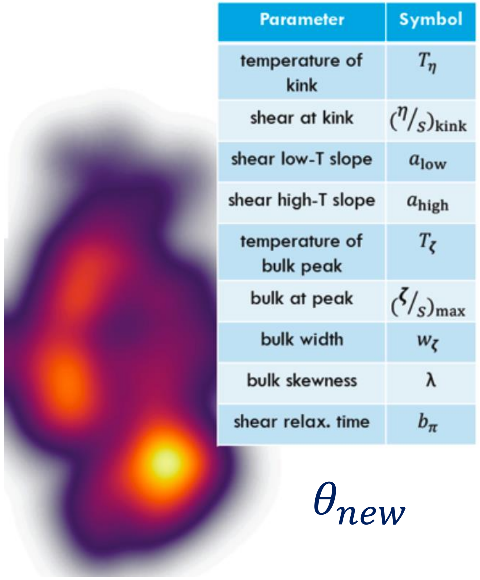


Parameter	Symbol
temperature of kink	T_η
shear at kink	$(^{\eta}/_{s})_{\text{kink}}$
shear low-T slope	a_{low}
shear high-T slope	a_{high}
temperature of bulk peak	T_ζ
bulk at peak	$(^{\zeta}/_{s})_{\text{max}}$
bulk width	w_ζ
bulk skewness	λ
shear relax. time	b_π

Run a few experiments at different engine parameters



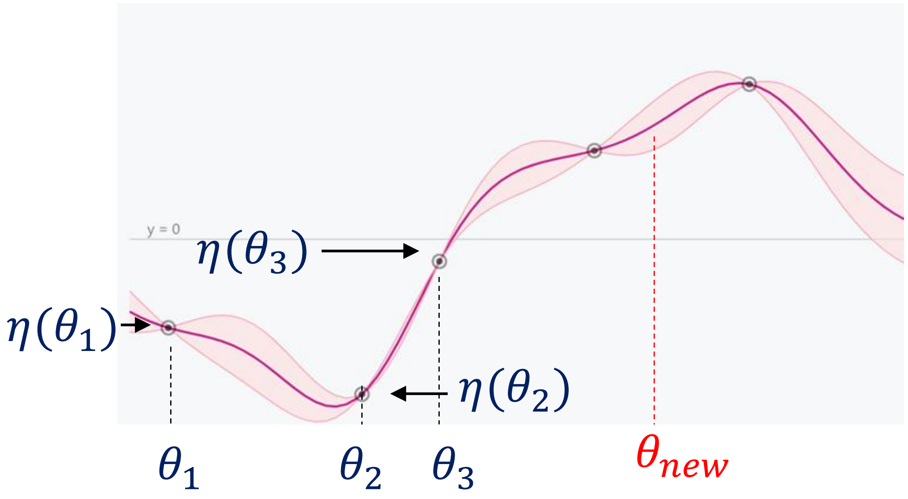
Train a predictive model using simulation data



Parameter	Symbol
temperature of kink	T_η
shear at kink	$(^{\eta}/_{s})_{\text{kink}}$
shear low-T slope	a_{low}
shear high-T slope	a_{high}
temperature of bulk peak	T_ζ
bulk at peak	$(^{\zeta}/_{s})_{\text{max}}$
bulk width	w_ζ
bulk skewness	λ
shear relax. time	b_π

Predict simulation output at a new engine parameter

Model Emulation



Simulation data:

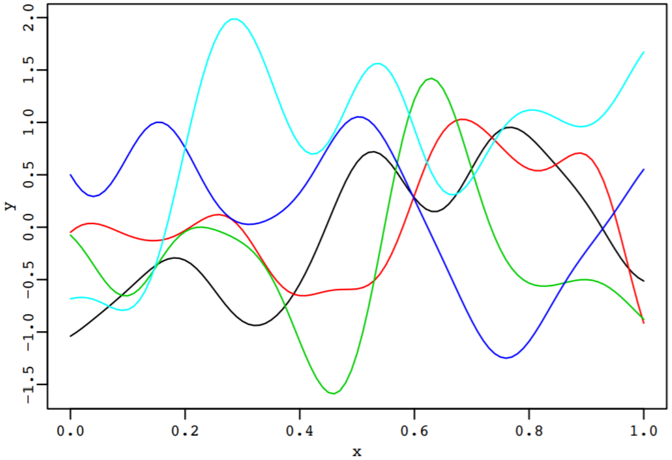
- **Input** parameters $\{\theta_1, \cdots, \theta_m\}$, $\theta_j \in [0,1]^d$
- Code **outputs** $\eta(\theta_1), \cdots, \eta(\theta_m) \in \mathbb{R}$, **expensive**

Objective:

- Predict **new** code evaluation $\eta(\theta_{new})$
- Quantify **uncertainty** of the prediction $\hat{\eta}(\theta_{new})$

Bayesian Predictive Modeling

- Assign to $\eta(\cdot)$ a **prior** stochastic process, which captures our **prior beliefs** on the unknown code output



- Condition on **observed** simulation data to obtain the **posterior** process $\eta(\cdot)|\mathrm{data}$, which can be used for **prediction** (**emulation**)

- **Gaussian processes** (Sacks et al. 1989): a flexible **Bayesian** nonparametric model widely used in machine learning, astrophysics, engineering, etc.

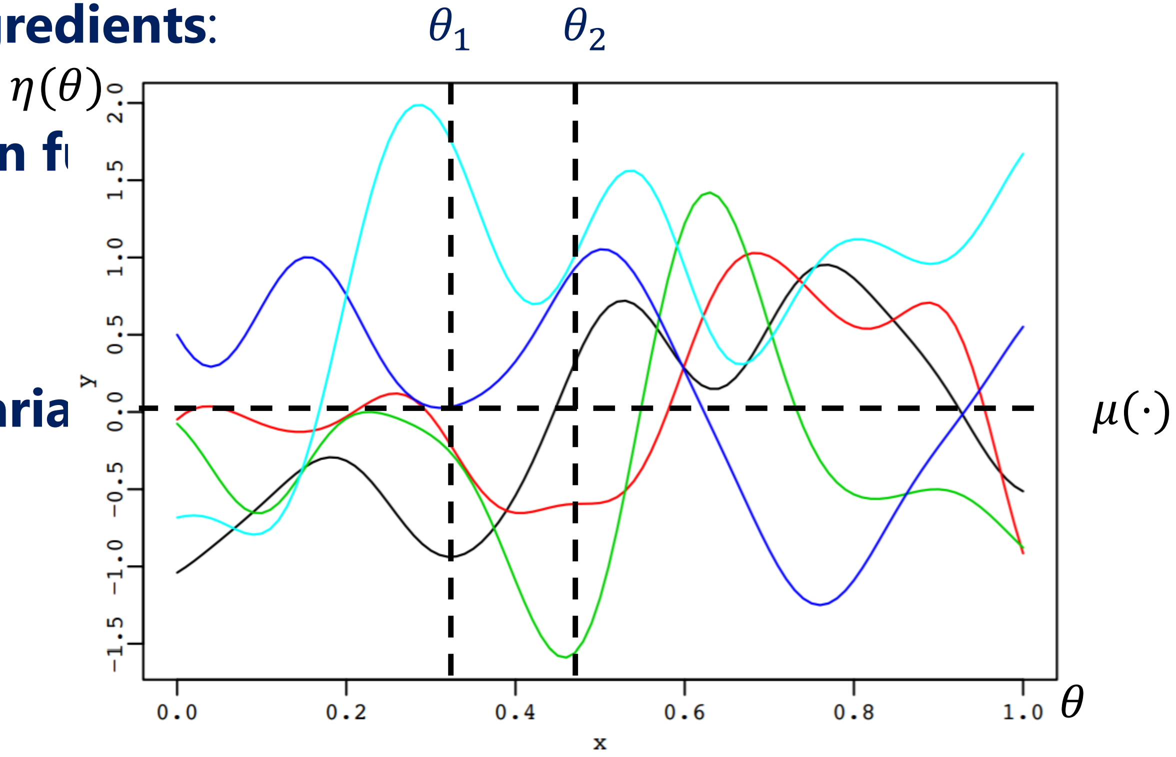
Gaussian Processes

Gaussian process (GP) prior model:

$$\eta(\cdot) \sim \mathrm{GP}\big(\mu(\cdot), k(\cdot,\cdot)\big)$$

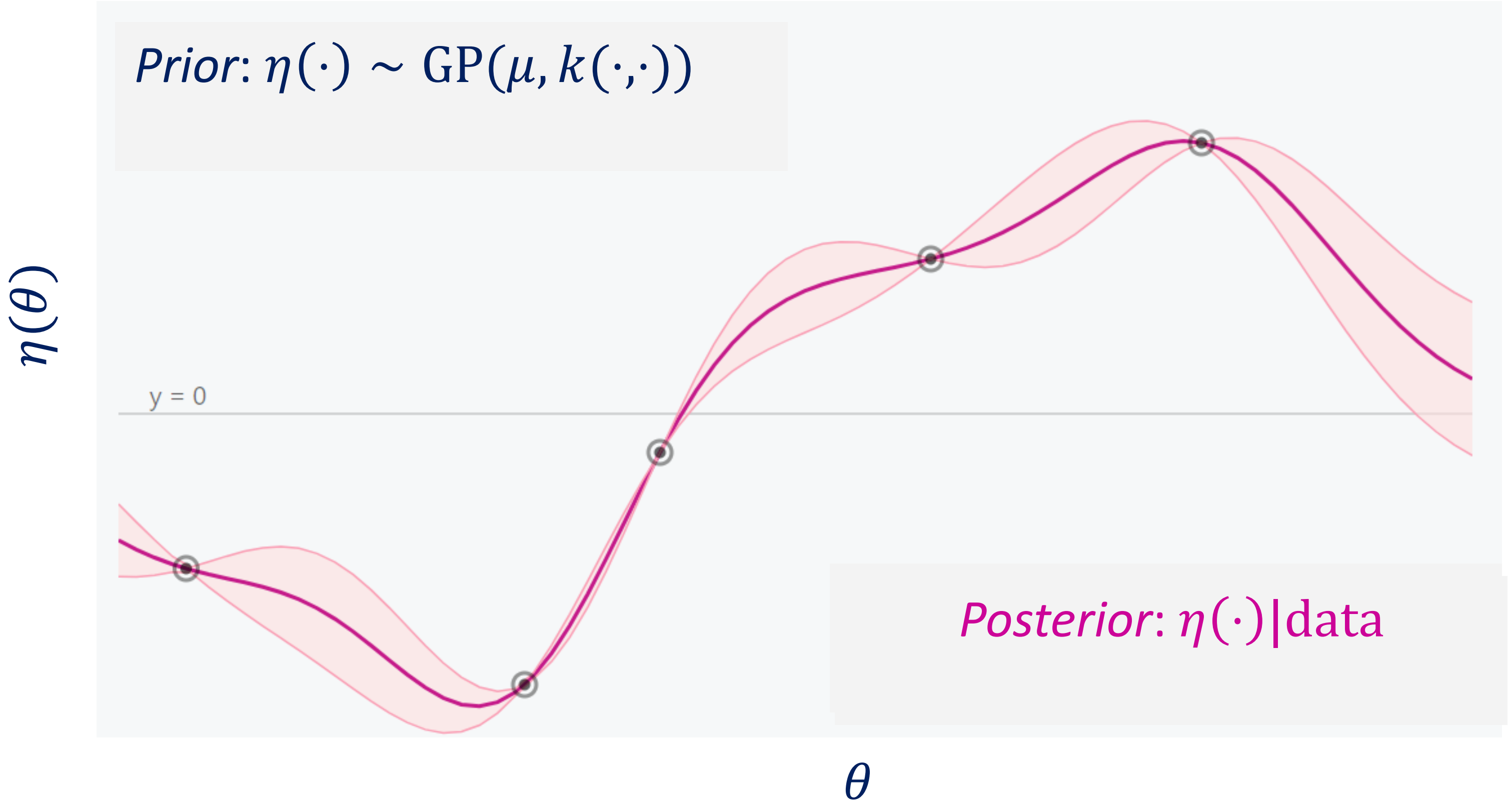
Two **ingredients**:

- **Mean f**
- **Covaria**



Gaussian Processes

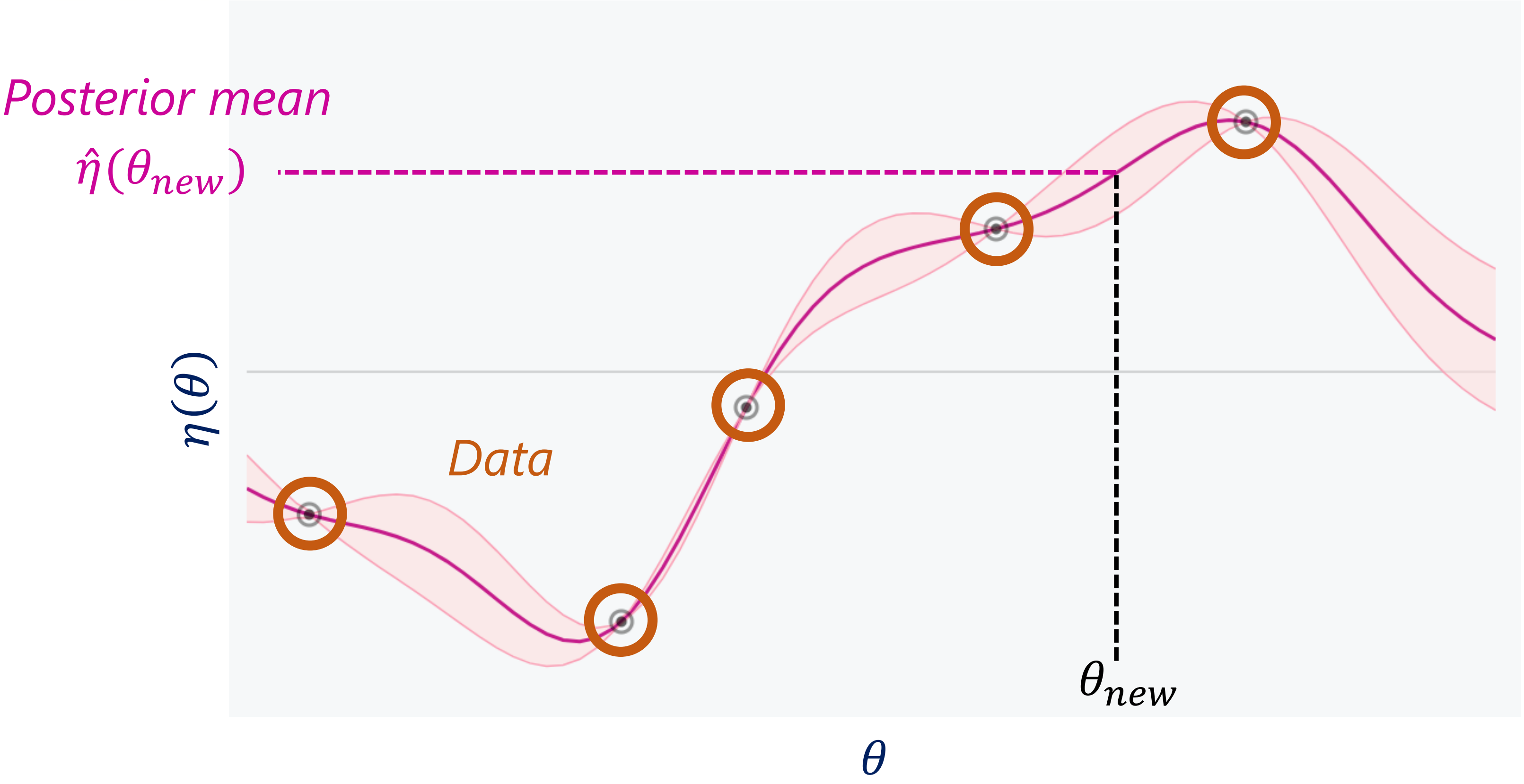
Conditioning on simulation data:



Prior: $\eta(\cdot) \sim \mathrm{GP}(\mu, k(\cdot,\cdot))$

Posterior: $\eta(\cdot)|\mathrm{data}$

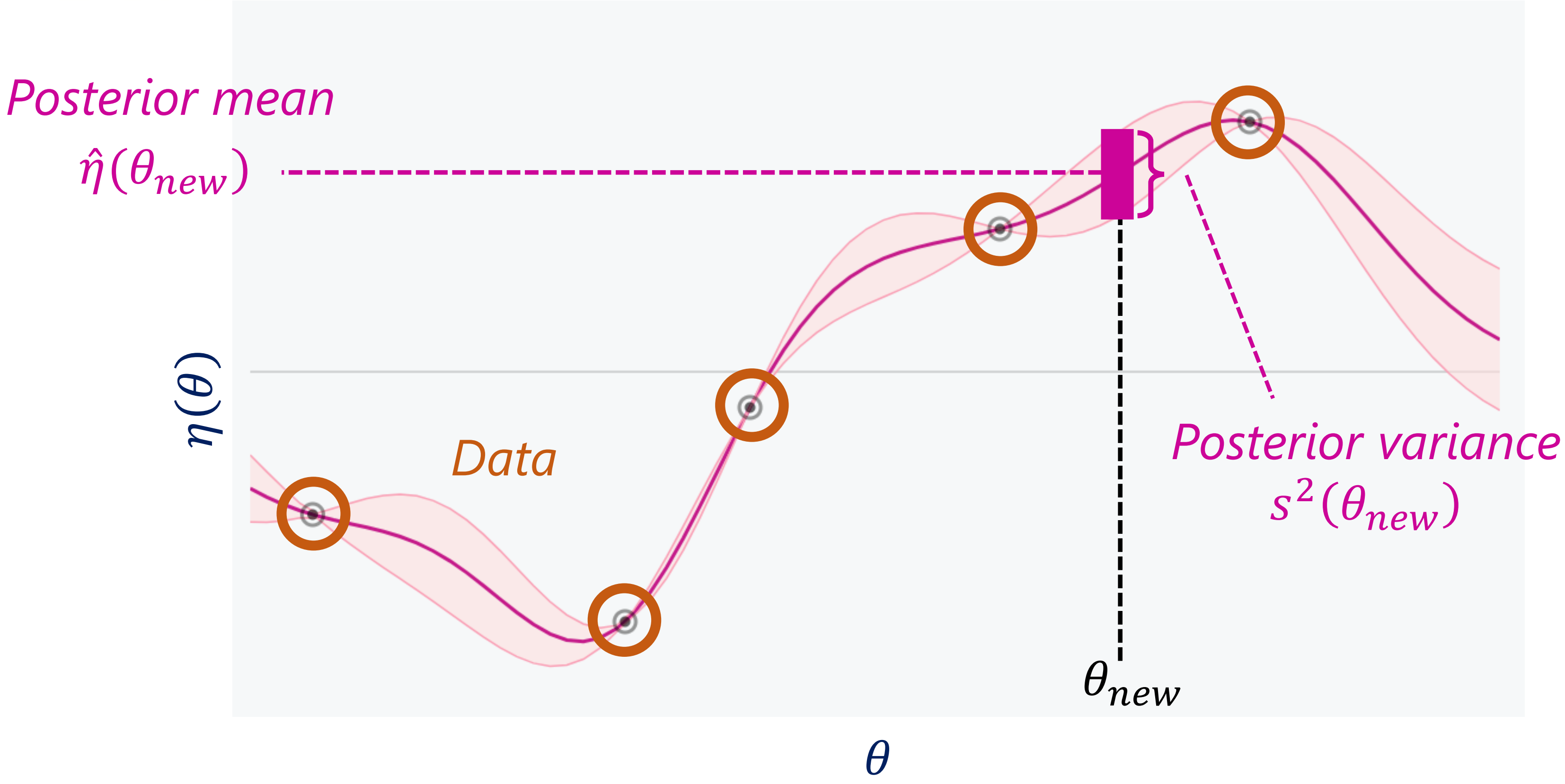
Gaussian Processes



Emulator (posterior process mean) – **closed form**:

$$\hat{\eta}(\theta_{new}) = \mathbb{E}[\eta(\theta_{new})|\text{data}] = \mu(\theta_{new}) + \boldsymbol{k}_{new}^{T}\boldsymbol{K}^{-1}(\boldsymbol{\eta} - \mu\mathbf{1})$$

Gaussian Processes



Uncertainty (posterior process variance) – also **closed form**:

$$s^2(\theta_{new}) = \mathrm{Var}[\eta(\theta_{new})|\mathrm{data}] = k(\theta_{new}, \theta_{new}) - \boldsymbol{k}_{new}^T \boldsymbol{K}^{-1} \boldsymbol{k}_{new}$$

Mean & Covariance Functions

Gaussian process (GP) prior model:

$$\eta(\cdot) \sim \mathrm{GP}\big(\mu(\cdot), k(\cdot,\cdot)\big)$$

- **Mean function** $\mu(\cdot)$ often taken to be a **constant** μ
- Popular **correlation functions** in the literature:
  - Squared-exponential correlation:
    - $k(\theta_1, \theta_2) = \gamma^2 \exp\left\{-\sum_{l=1}^{d} \phi_l \left(\theta_{1,l} - \theta_{2,l}\right)^2\right\}$
  - Matérn correlation (Cressie 1991)
  - Cubic correlation (Santner et al. 2013)

We will go over **GP fitting** in **Hands-on Session**

Parameter Estimation with Emulator

Let's integrate this **emulator** for **parameter estimation**:

Posterior distribution:

$$\pi(\theta|y_1,\cdots,y_n) \propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i-\hat{\eta}(\theta)\right)^2\right\}\pi(\theta)$$

$$\pi(\theta|y_1,\cdots,y_n) \propto \frac{1}{\sqrt{2\pi\sigma^{*2}(\theta)}^2}\exp\left\{-\frac{1}{2\sigma^{*2}(\theta)^2}\sum_{i=1}^{n}\left(y_i-\hat{\eta}(\theta)\right)^2\right\}\pi(\theta), \quad \sigma^{*2}(\theta)=\sigma^2+s^2(\theta)$$

- Simulator $\eta(\theta)$ **expensive**, want to replace with **emulator** $\hat{\eta}(\theta)$
- But the **emulator** has **predictive uncertainty** as well:
  - $s^2(\theta_{new}) = \text{Var}[\eta(\theta_{new})|\text{data}]$ from GP emulator
  - Integrate this predictive uncertainty within the **likelihood**

Parameter Estimation with Emulator

Let's integrate this **emulator** for **parameter estimation**:

Posterior distribution:

$$\pi(\theta|y_1,\cdots,y_n) \propto \frac{1}{\sqrt{2\pi{\sigma^*}^2(\theta)^2}} \exp\left\{-\frac{1}{2{\sigma^*}^2(\theta)^2}\sum_{i=1}^{n}\left(y_i-\hat{\eta}(\theta)\right)^2\right\}\pi(\theta), \quad {\sigma^*}^2(\theta)=\sigma^2+s^2(\theta)$$

- We can use this modified **posterior** (which **integrates** the emulator) within **MCMC** sampling

- Efficient **parameter estimation** for **expensive** computer models

Outline

- Model Emulation
  - Gaussian Process
- **Multiple Observables**
  - **Principal Component Analysis**
- Bayesian Inference Workflow

Principal Component Analysis (PCA)

- **Problem**: model M **correlated observables** jointly

- **One solution (dimension reduction)**:
  - convert to $\mathrm{k} \ll M$ **independent** (transformed) outputs
  - model them **independently**

- **Method: Principal Component Analysis (PCA)**
  - Finds **directions** with **maximum variances** (contain most information of data) and **project** data onto these directions

Principal Component Analysis (PCA)

- **Principal Components (PC):**
  - a few **linearly uncorrelated** coordinates
- **PCA:**
  - a **linear transformation** of observables that defines a new **coordinate** rule:
    - 1st PC: **highest** projected variance
    - 2nd PC: second highest projected variance, **orthogonal (perpendicular)** to 1st PC
    - …

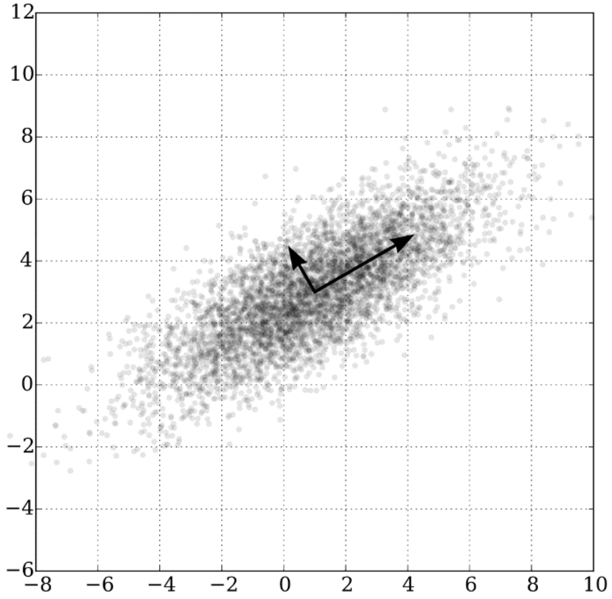
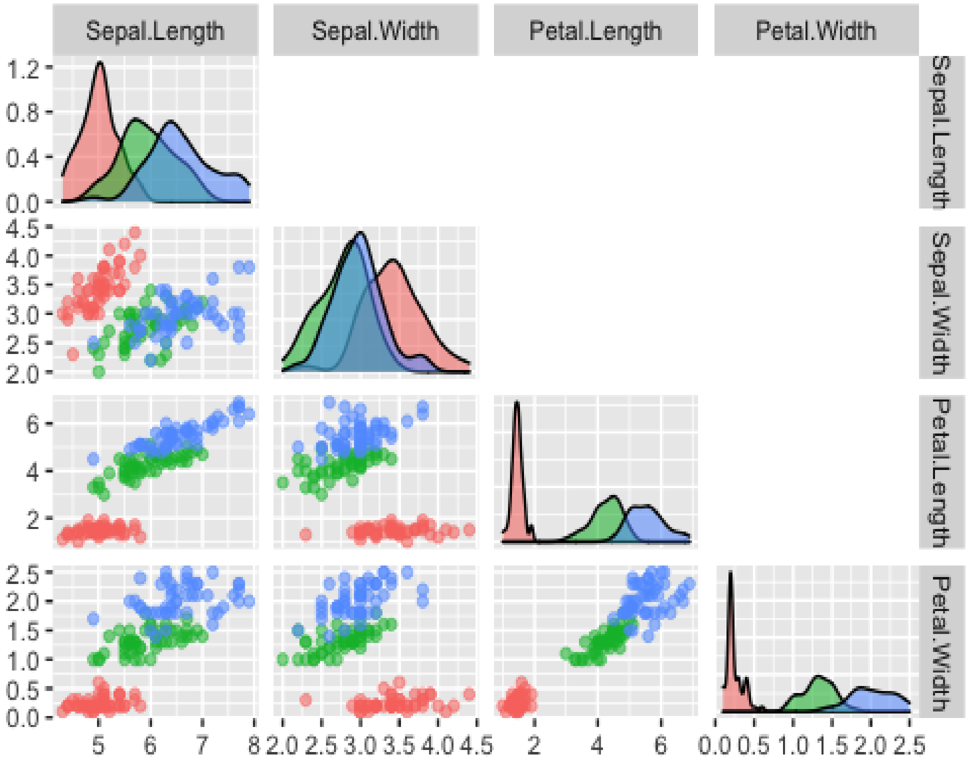


Figure source: https://en.wikipedia.org/wiki/Principal_component_analysis

A Simple Example (Iris Dataset)

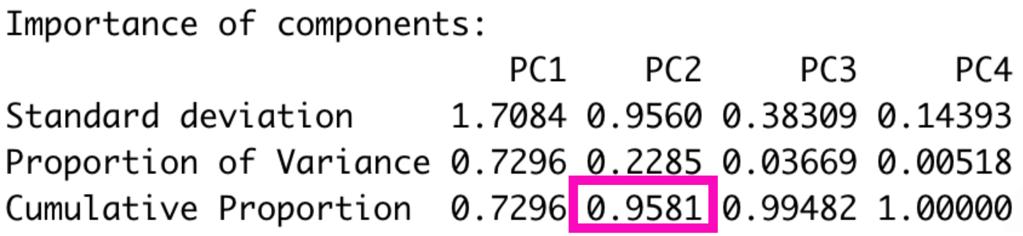
- **Iris data:**
  - $n = 150$ iris flowers, classified as 3 types (red, green, blue)
  - $M = 4$ measurements (observables) for each flower



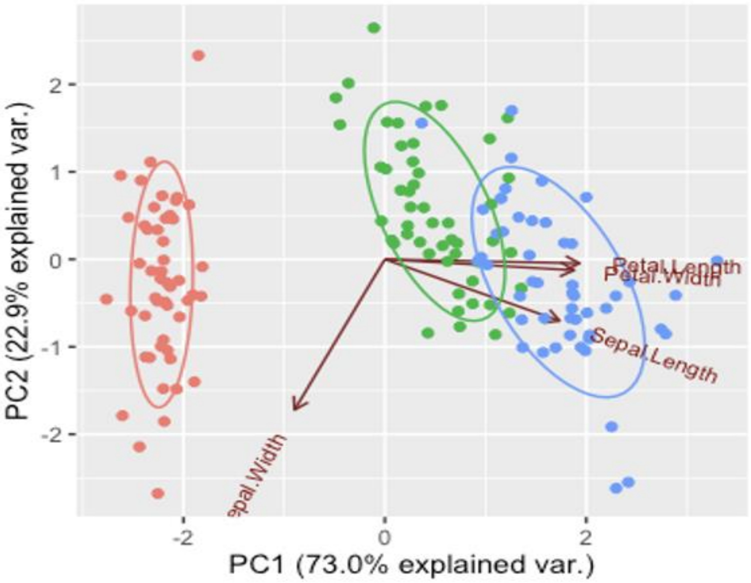
- Visualizing pairwise correlations of 4 observables

A Simple Example (Iris Dataset)

- **Perform PCA:**
  - First 2 PCs (PC1, PC2) explains **95.81%** of total variance
  - **Keep** PC1 and PC2, discard PC3 and PC4



```
Importance of components:
                          PC1    PC2     PC3     PC4
Standard deviation     1.7084 0.9560 0.38309 0.14393
Proportion of Variance 0.7296 0.2285 0.03669 0.00518
Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```



- First 2 PCs conveys almost all information contained in data
- ⇒ use for further analysis

  (e.g., classification, emulation,

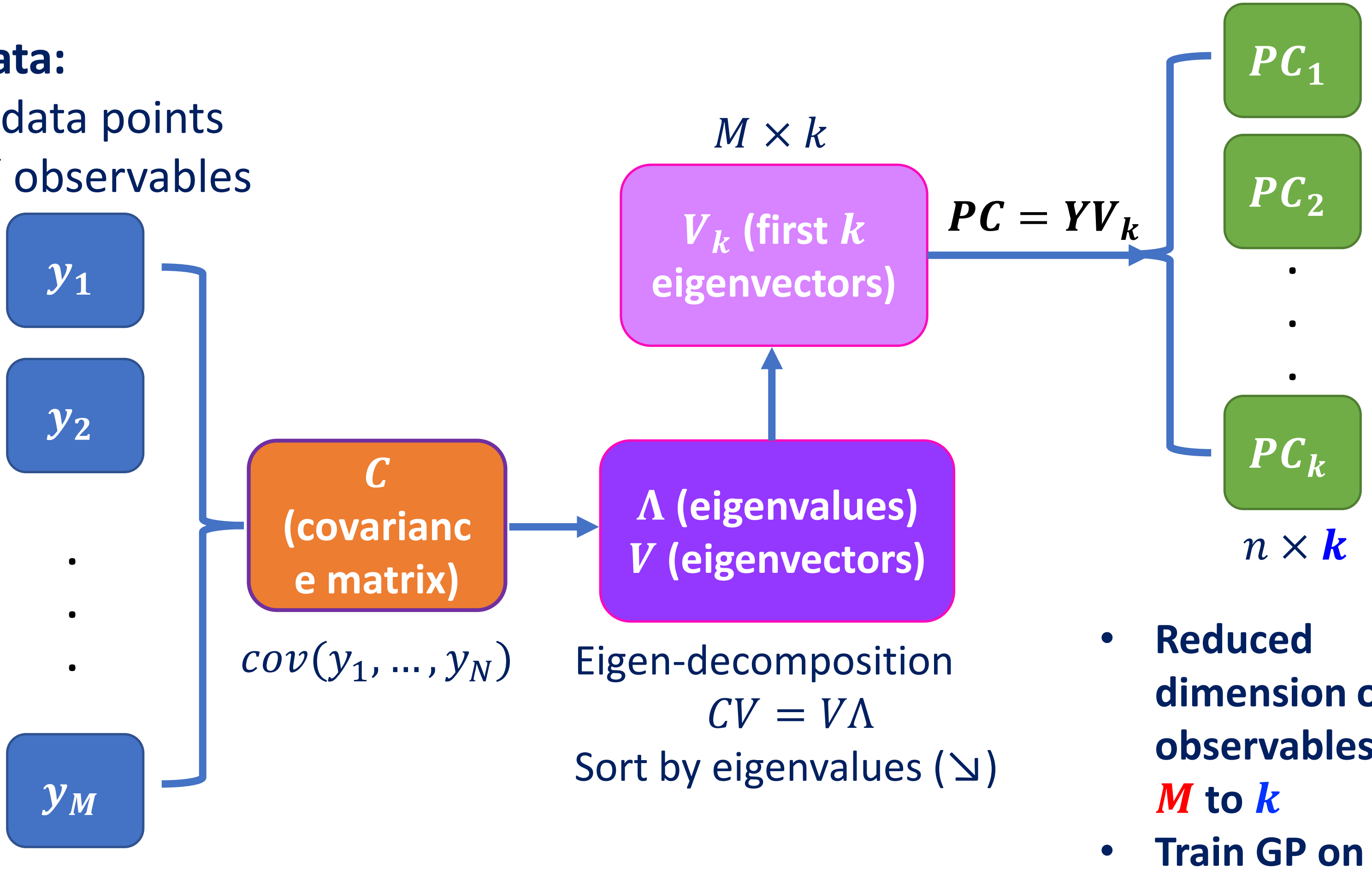
  etc.)

PCA for Parameter Estimation

Data:

$\boldsymbol{n}$ data points

$\boldsymbol{M}$ observables



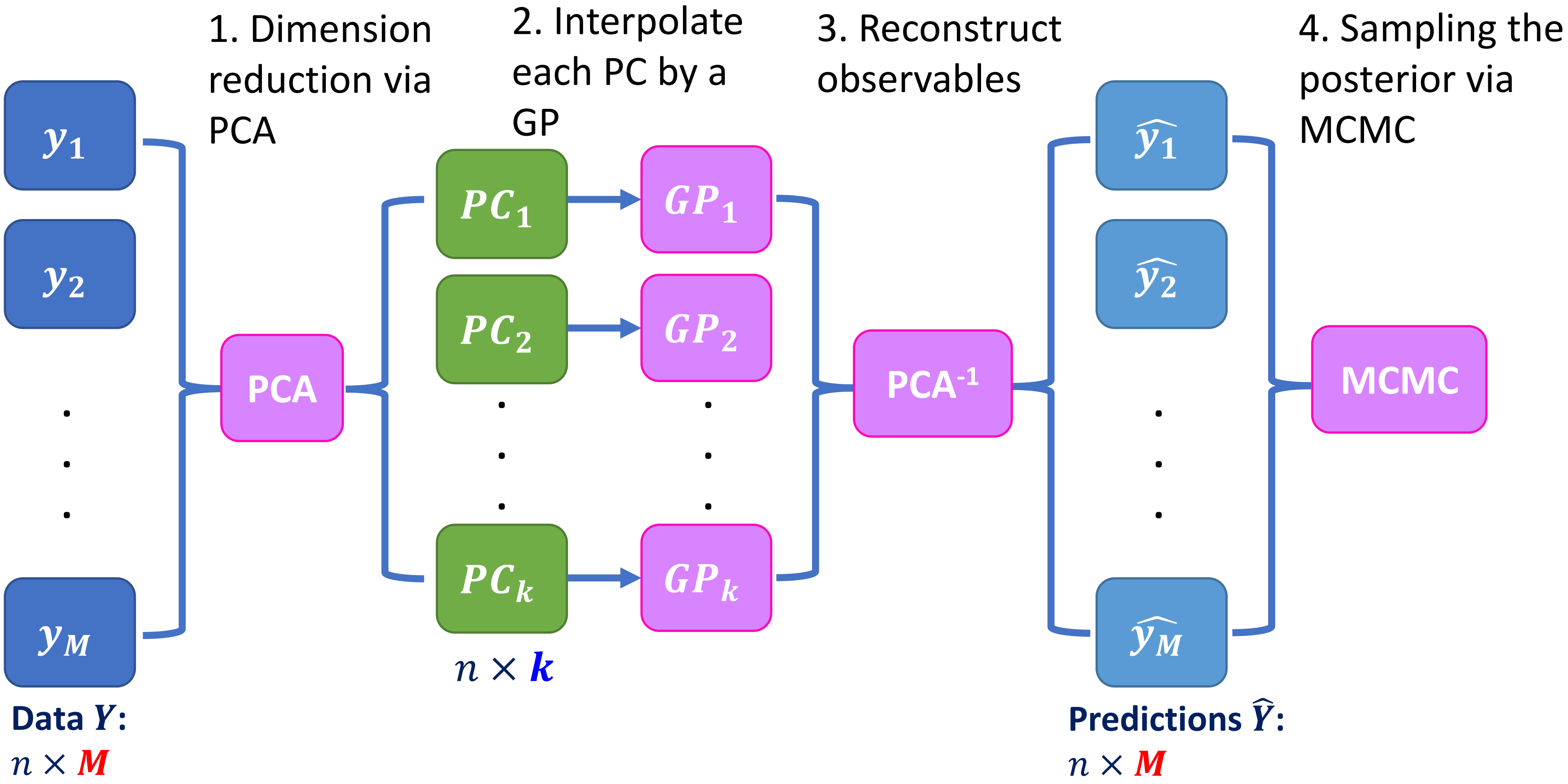
Matrix $\boldsymbol{Y}$:
$n \times \boldsymbol{M}$

- **Reduced dimension of observables from $\boldsymbol{M}$ to $\boldsymbol{k}$**
- **Train GP on PCs (independent)**

Outline

- Model Emulation
  - Gaussian Process
- Multiple Observables
  - Principal Component Analysis
- **Bayesian Inference Workflow**

Workflow



1. Dimension reduction via PCA

2. Interpolate each PC by a GP

3. Reconstruct observables

4. Sampling the posterior via MCMC

Data Y: $n \times M$

Predictions $\hat{Y}$: $n \times M$

Workflow

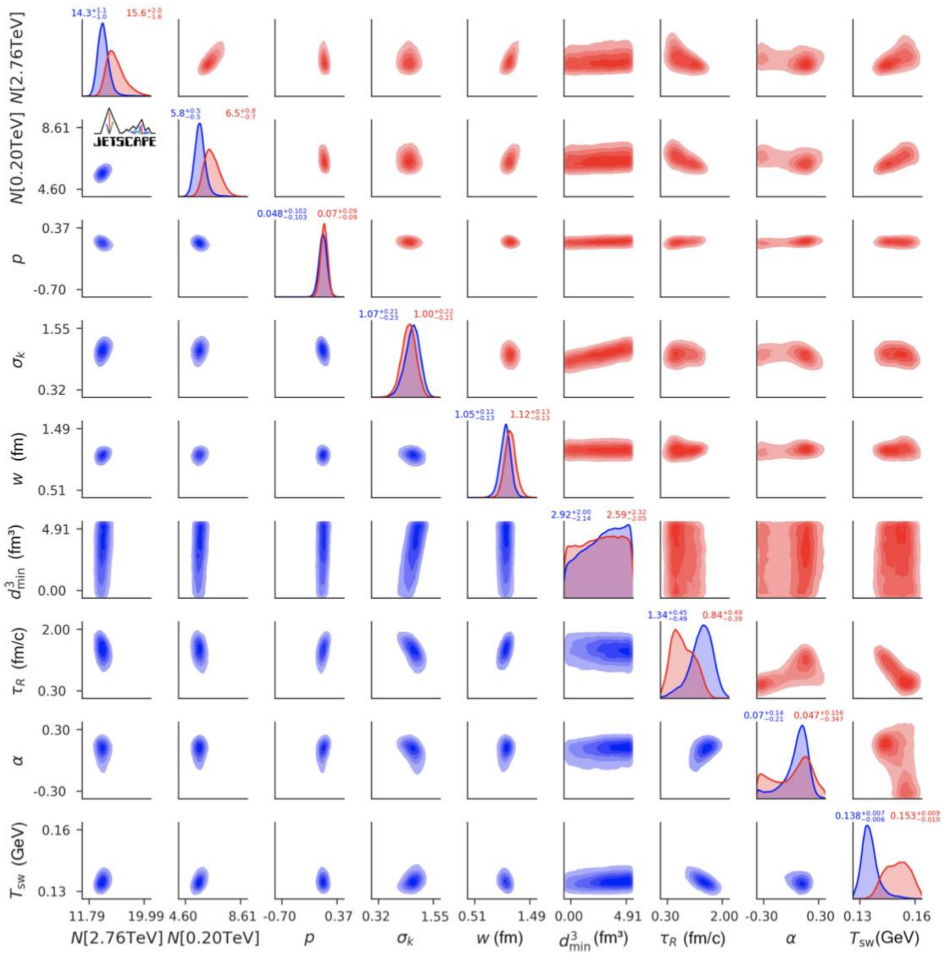


FIG. 10. The posterior for Grad (blue) and Chapman-Enskog (red) viscous corrections for select parameters related to the initial state, prehydrodynamic evolution, and switching temperature. The histograms on the diagonal are the marginal distributions for each parameter, with appended numbers denoting the median and the left and right limits of the 90% credible interval. Off-diagonal histograms display the joint posterior of each pair of parameters, marginalized over all others.

Posterior distribution of select parameters in recent **JETSCAPE** study

Everett et al. (2021): https://journals.aps.org/prc/pdf/10.1103/PhysRevC.103.054904

Questions

Hands-On Session

We will now go over a pedagogical example for **Bayesian parameter estimation** in the **hands-on session**.

Please follow along at: