An Introduction to
Uncertainty Quantification for Predictive Science
Part I

Simone Venturi
Ph.D. Candidate, NEQRAD Laboratory
University of Illinois at Urbana-Champaign, USA

Prof. Panesi's UQ Class, Fall 2019
September, 4th

Motivations

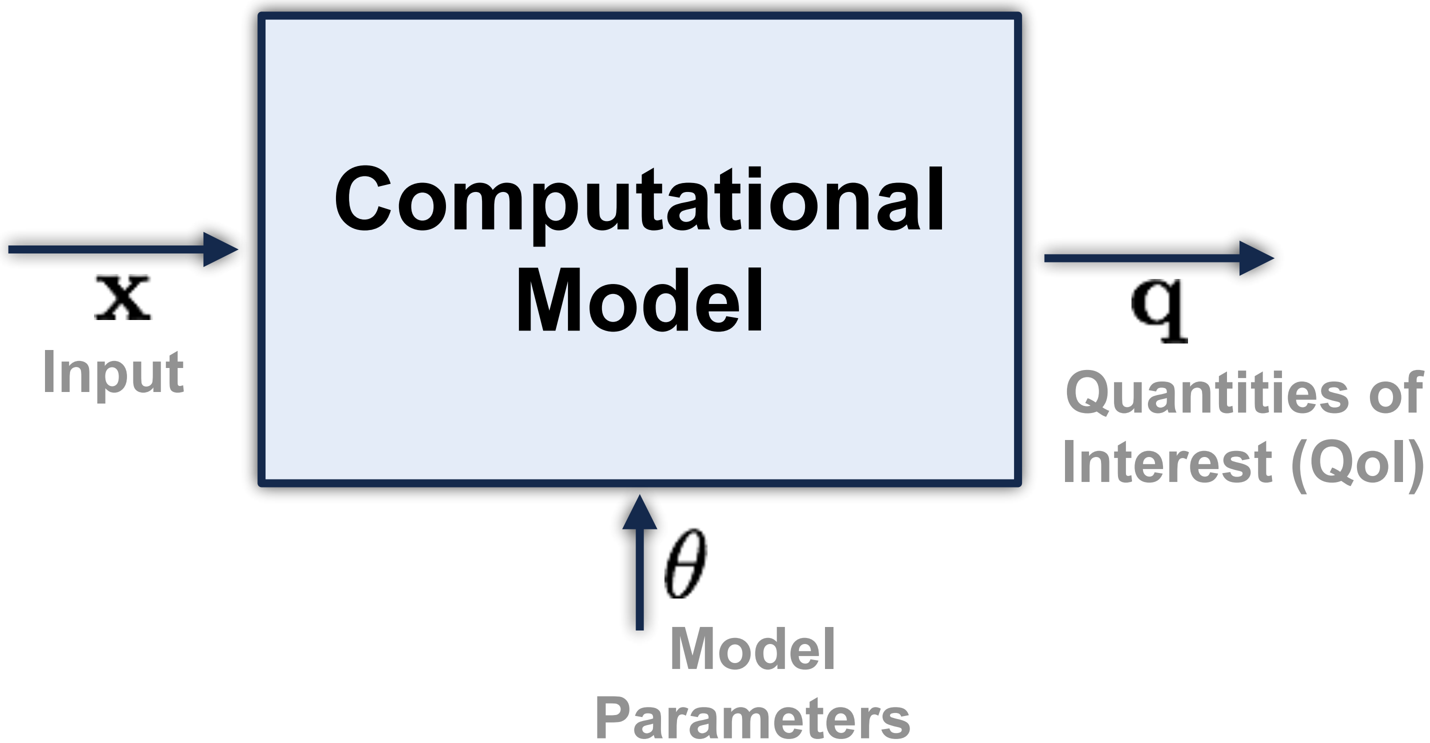
- Showing some UQ applications
  - Giving a road map of Predictive Science through examples;
- 2nd week of class … too early for applications?
  - Overview of the UQ class' building blocks;
  - Stimulating ideas for the final project.

Outline

- Introduction
  - What is Predictive Science and why is it important for my research?
- A 1st Example: Mass-Spring-Damper System

Introduction

"The ultimate purpose of most computational models is to make predictions, commonly in support of some decision-making process (e.g, for design of operation of some system)." [1]



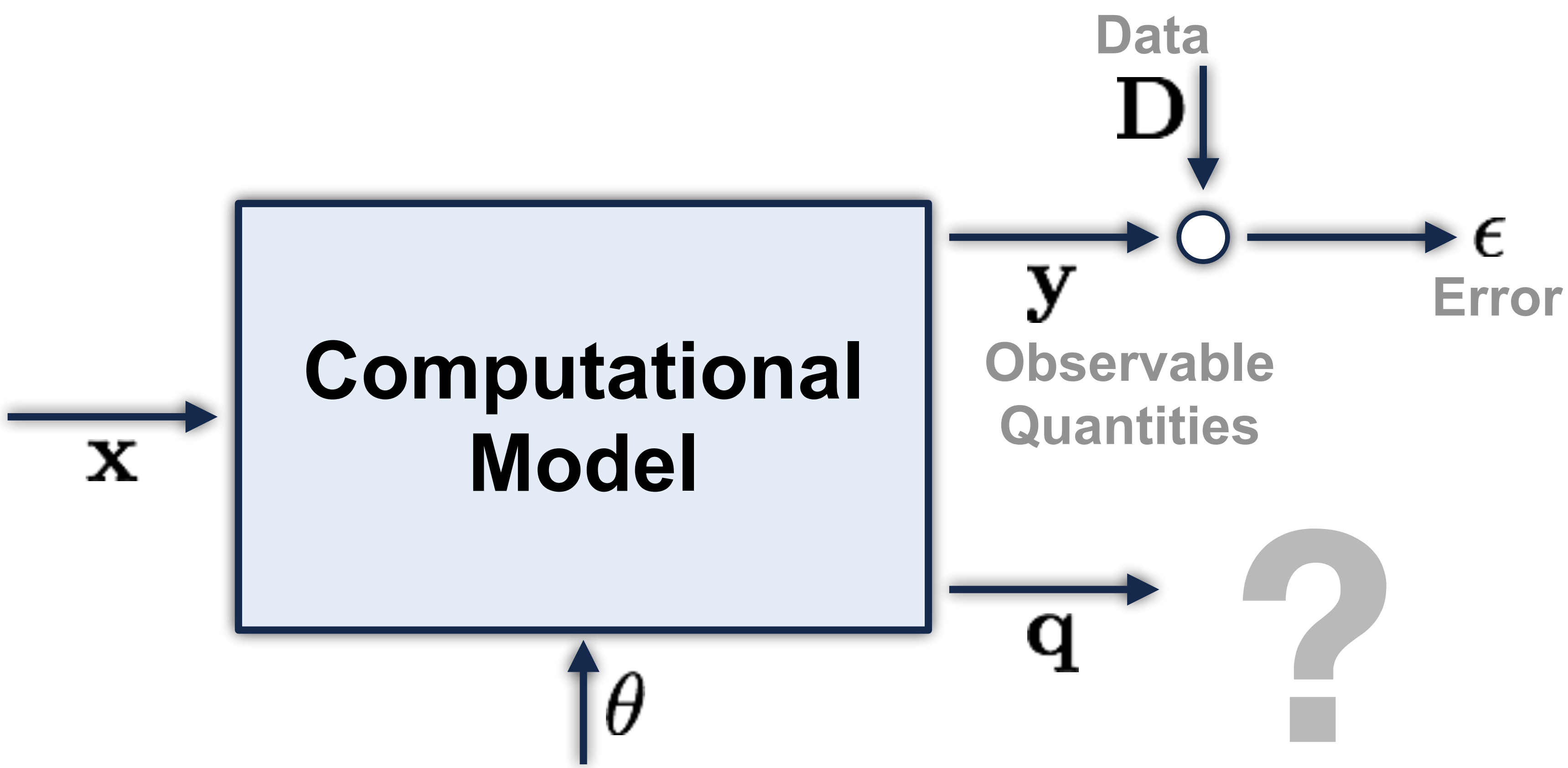
Reliability Assessment of the Computational Model

- **Verification**: Computer Simulation vs Mathematical Model
- **Uncertainty Quantification**: Determining uncertainties on the QoIs
- **Validation**: Deciding whether the model is a sufficient representation of reality for the purpose for which it will be used

[1] Oliver et al., "*Validating Predictions of Unobserved Quantities*", Computer Methods in Applied Mechanics and Engineering, Vol. 283, 2015.

Introduction

Generally, there is no observational data available for the QoIs for the scenarios of interest; this fact forces us to make extrapolative predictions.



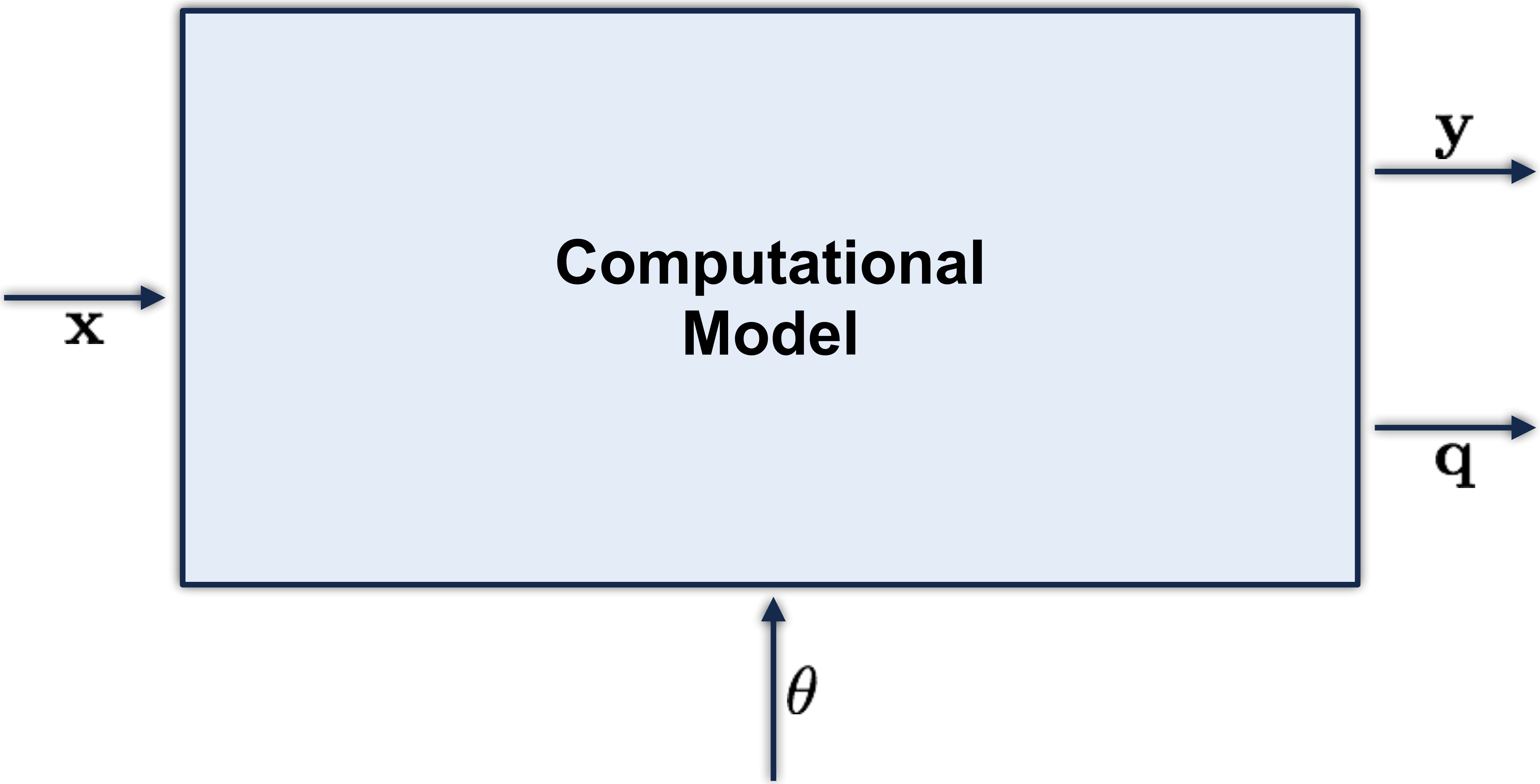
In order to asses the validity of the model, classical approaches to validation compare some observable outputs to observations.
This only ensures that the model can predict:

- the observable quantities,
- under the conditions of the observations,
- under the assumption of no observation error.

The need to extrapolate raises concerns about the reliability of predictions.
What entitles us to make such predictions?

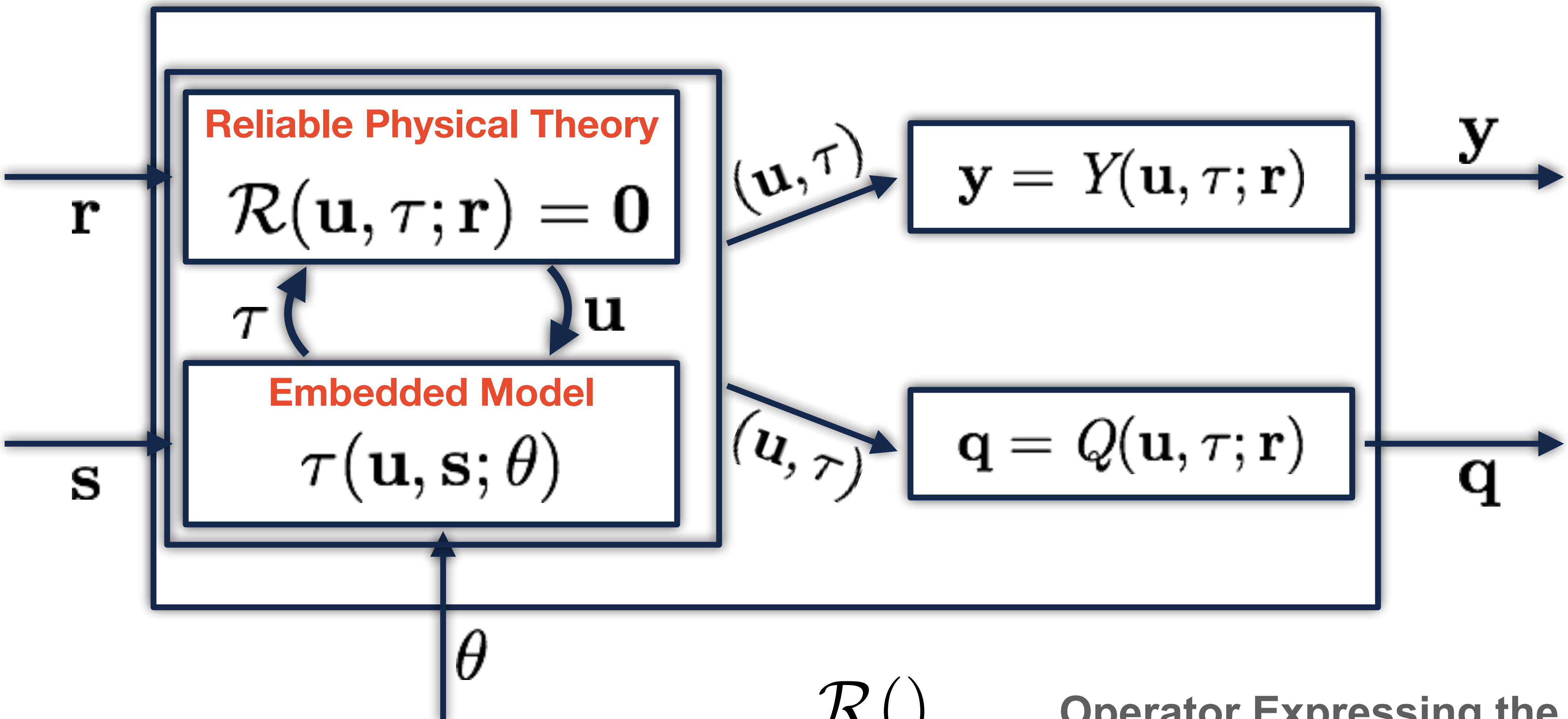
Introduction

The computational models are generally physics based, and they are constructed upon theories that are known to be highly reliable within well-defined domains of applicability.



Introduction

The computational models are generally physics based, and they are constructed upon theories that are known to be highly reliable within well-defined domains of applicability.



$\mathcal{R}()$	Operator Expressing the Theory
$\mathbf{u}$	State Variables
τ	Closure Variables
θ	Parameters of the Embedded Model
$\mathbf{r} \cup \mathbf{s} = \mathbf{x}$	Input for the Scenario

Introduction

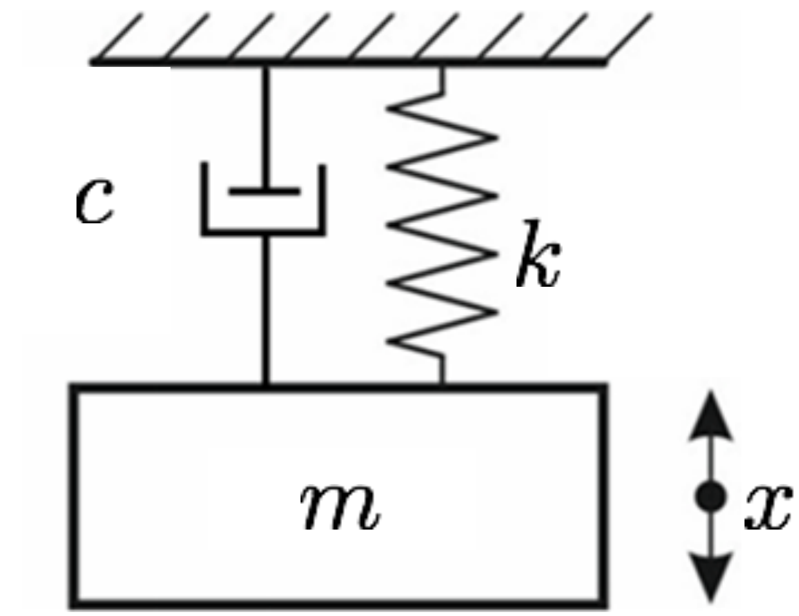
Example: Mass-Spring-Damper, Real World System

Reliable Physical Theory

$$\mathcal{R}(\mathbf{u}, \tau; \mathbf{r}) = \mathbf{0}$$

$$\boxed{m\ddot{x} + c\dot{x} + kx = 0}$$

$$\mathbf{u} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \qquad \tau = \begin{bmatrix} c \\ k \end{bmatrix} \qquad \mathbf{r} = \begin{bmatrix} m \\ x(0) \\ \dot{x}(0) \end{bmatrix}$$



Real World Embedded Model

$$\tau = (\mathbf{u}, \mathbf{v}, \mathbf{s}, \theta)$$

$$\boxed{\begin{aligned} k &= \text{const} \\ c(T) &= \exp\left(\frac{T_0}{T} - 1\right) \\ \dot{T} &= c(T)\dot{x}^2 - \frac{T - T_0}{t_T} \end{aligned}}$$

$$\mathbf{v} = T \qquad \mathbf{s} = \begin{bmatrix} T_0 \\ t_T \end{bmatrix} \qquad \theta = k$$

$$\boxed{\begin{aligned} \mathbf{y} &= x \\ \mathbf{q} &= \max(\dot{x}) \end{aligned}}$$

Introduction

Note: Displacement or velocity observations (e.g., from proximity sensor or vibrometer) can be represented by:

$$\mathbf{y} = \mathbf{C^T u}$$

where:

$$\mathbf{C^T} = [\mathbf{1}, \mathbf{0}] \quad \text{or} \quad \mathbf{C^T} = [\mathbf{0}, \mathbf{1}]$$

x

Real World Embedded Model

$$\tau = (\mathbf{u}, \mathbf{v}, \mathbf{s}, \theta)$$

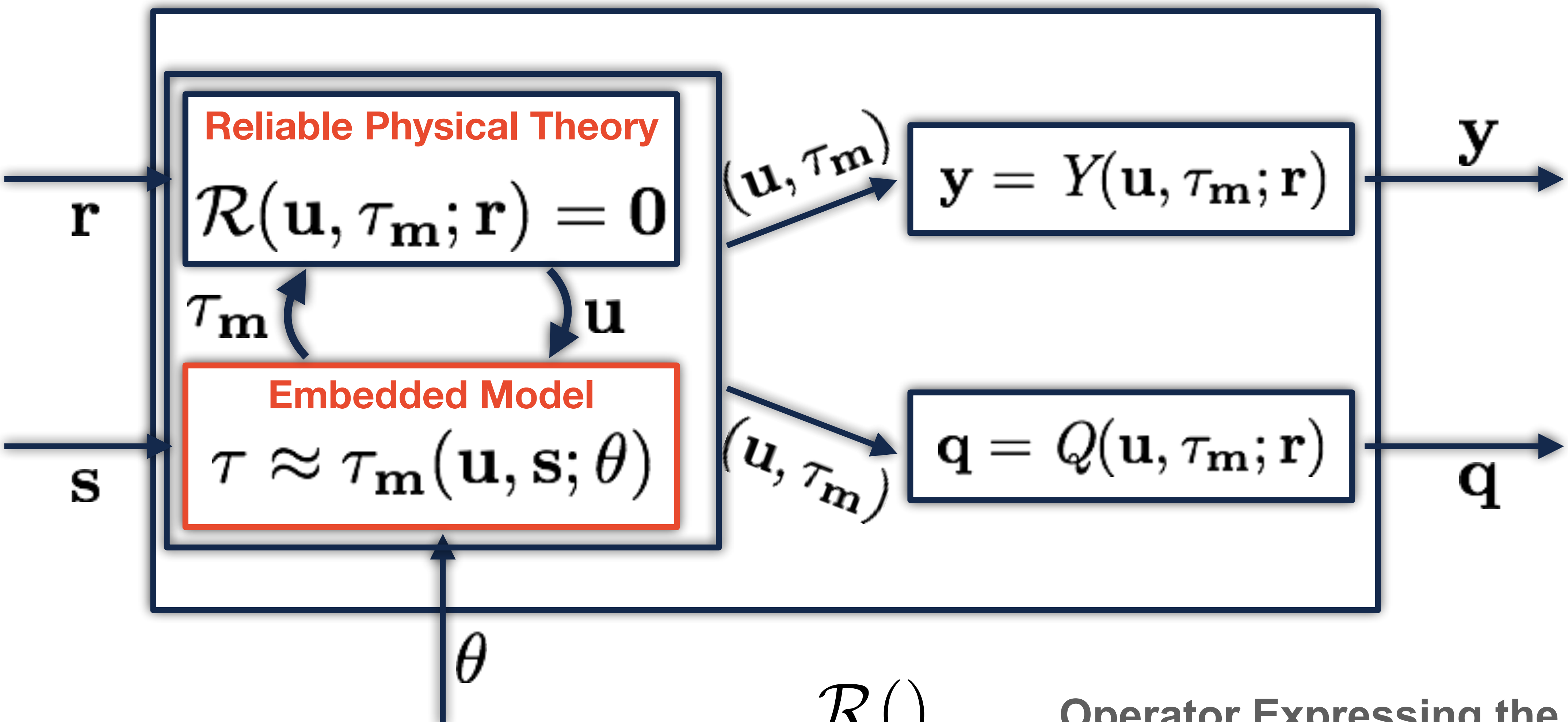
$$\boxed{\begin{aligned} k &= \text{const} \\ c(T) &= \exp\left(\frac{T_0}{T} - 1\right) \\ \dot{T} &= c(T)\dot{x}^2 - \frac{T - T_0}{t_T} \end{aligned}}$$

$$\mathbf{v} = T \qquad \mathbf{s} = \begin{bmatrix} T_0 \\ t_T \end{bmatrix} \qquad \theta = k$$

$$\boxed{\begin{aligned} \mathbf{y} &= x \\ \mathbf{q} &= \max(\dot{x}) \end{aligned}}$$

Introduction

If τ where known in terms of $(\mathbf{u}, \mathbf{r})$, the system would be closed. However, it is often the case that the required relationship between τ and $(\mathbf{u}, \mathbf{r})$ is unknown or does not exist.



$\mathcal{R}()$	Operator Expressing the Theory
$\mathbf{u}$	State Variable
$\tau_{\mathbf{m}}$	Closure Variables
θ	Parameters of the Embedded Model
$\mathbf{r} \cup \mathbf{s} = \mathbf{x}$	Input for the Scenario

τ

Introduction

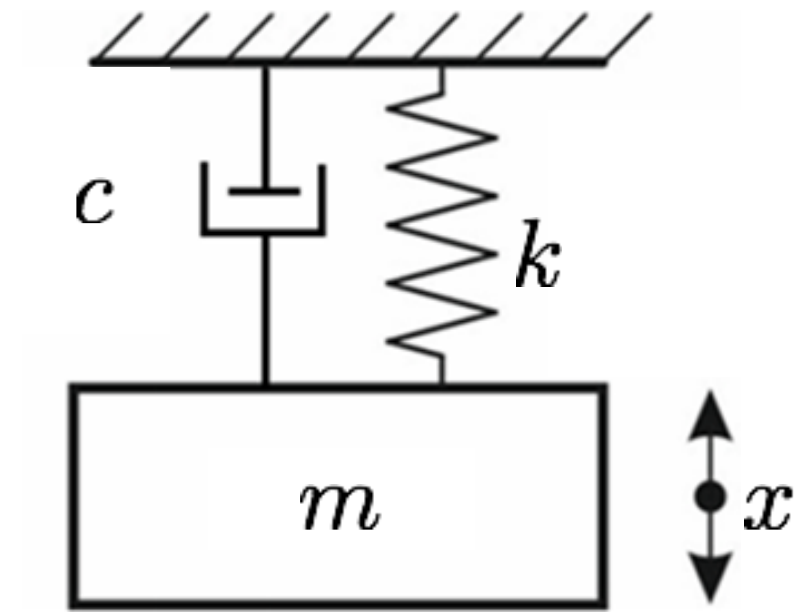
Reliable Physical Theory

Example: Mass-Spring-Damper, Approximated System

$$\mathcal{R}(\mathbf{u}, \tau_{\mathbf{m}}; \mathbf{r}) = \mathbf{0}$$

$$\boxed{m\ddot{x} + c\dot{x} + kx = 0}$$

$$\mathbf{u} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \qquad \tau = \begin{bmatrix} c \\ k \end{bmatrix} \qquad \mathbf{r} = \begin{bmatrix} m \\ x(0) \\ \dot{x}(0) \end{bmatrix}$$



Approximated Embedded Model

The true physics of the damper are not well-understood by the modeler

$$\tau_m = (\cancel{u}, \cancel{x}, \cancel{s}, \theta)$$

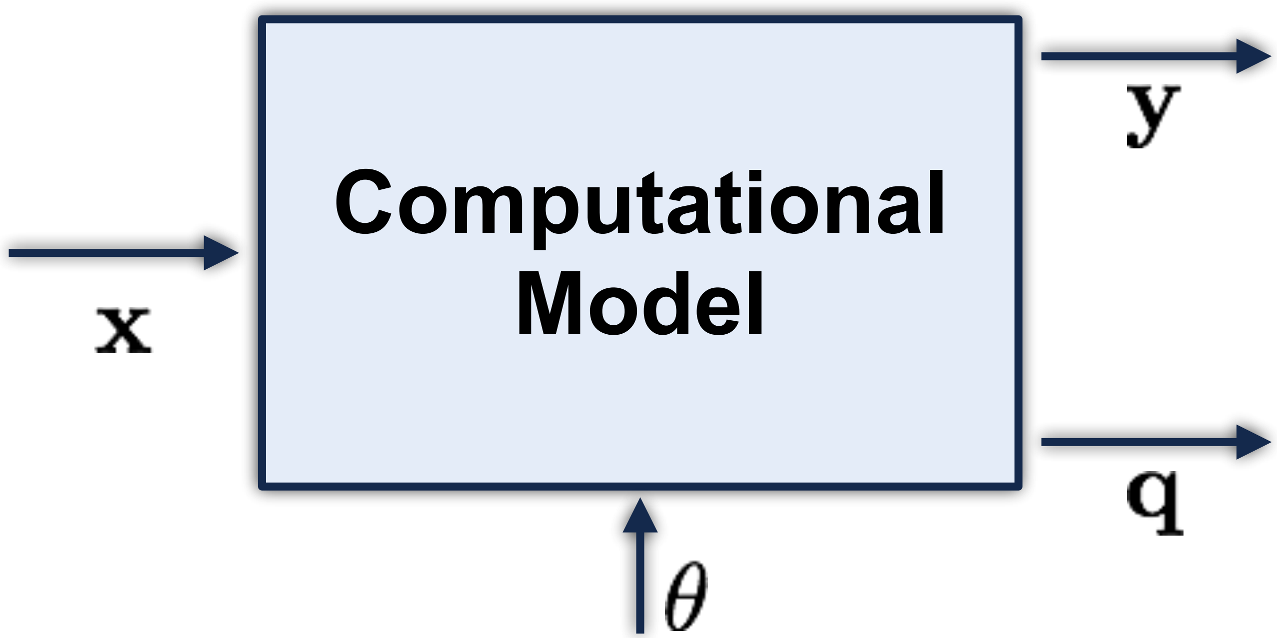
$$\boxed{\begin{aligned} k &= \text{const} \\ c &= \text{const} \end{aligned}}$$

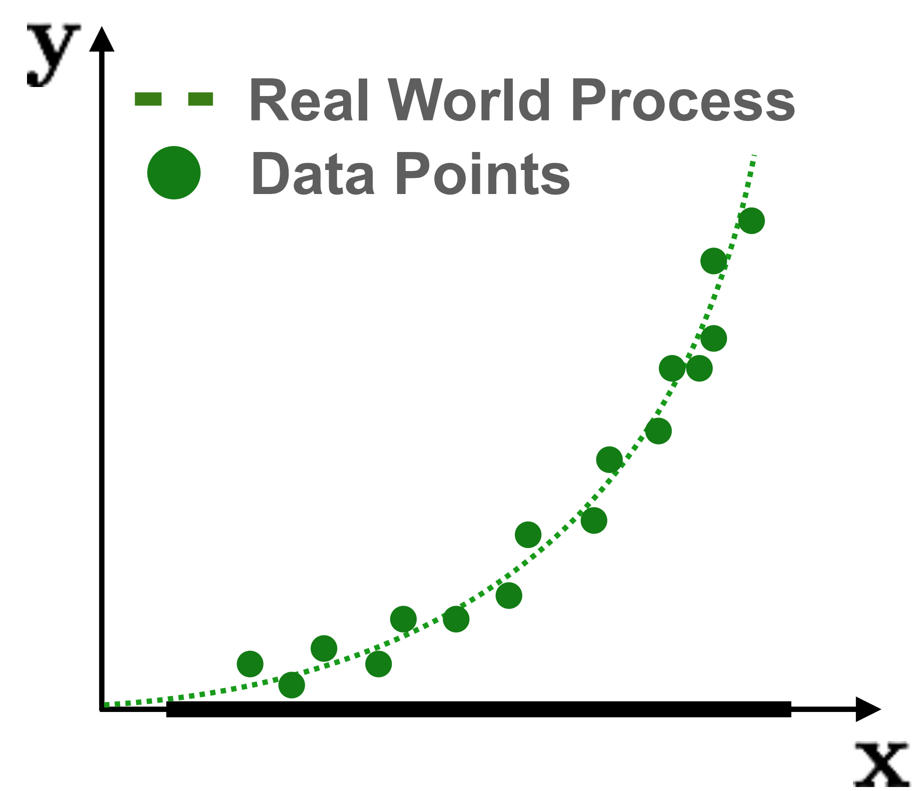
$$\theta = \begin{bmatrix} c \\ k \end{bmatrix}$$

$$\boxed{\begin{aligned} \mathbf{y} &= x \\ \mathbf{q} &= \max(\dot{x}) \end{aligned}}$$

Classic Approach to Validation

One approach [2] for taking into account the uncertainties on the predictions relies on "appending" a statistical model directly to the observable quantities:

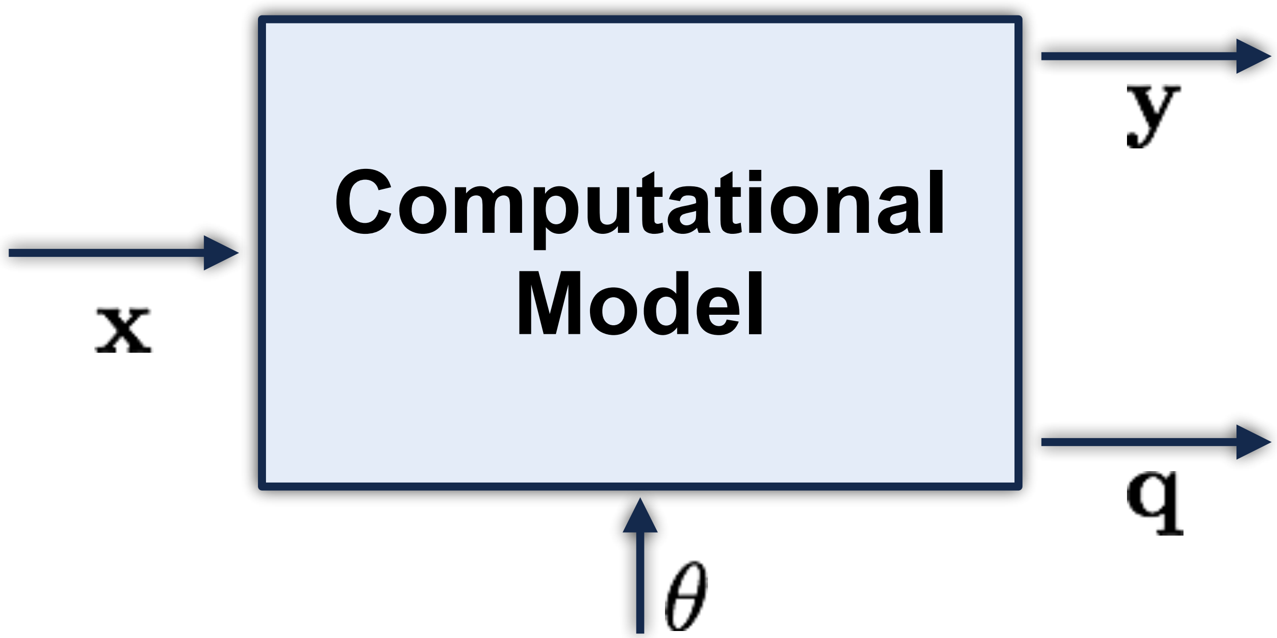




[2] Kennedy, M.C. and O'Hagan, A., "*Bayesian Calibration of Computer Models*", Journal of the Royal Statistical Society: Series B, Vol. 63, 2001.

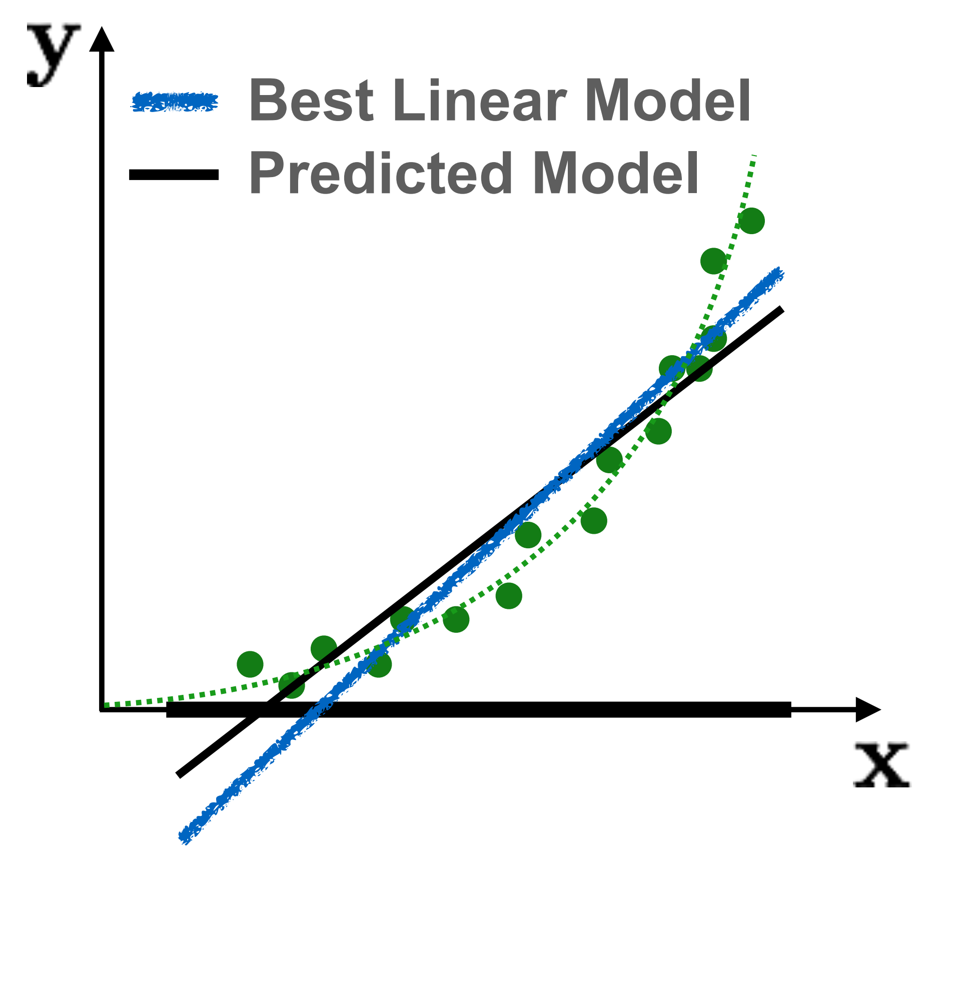
Classic Approach to Validation

One approach [2] for taking into account the uncertainties on the predictions relies on "appending" a statistical model directly to the observable quantities:



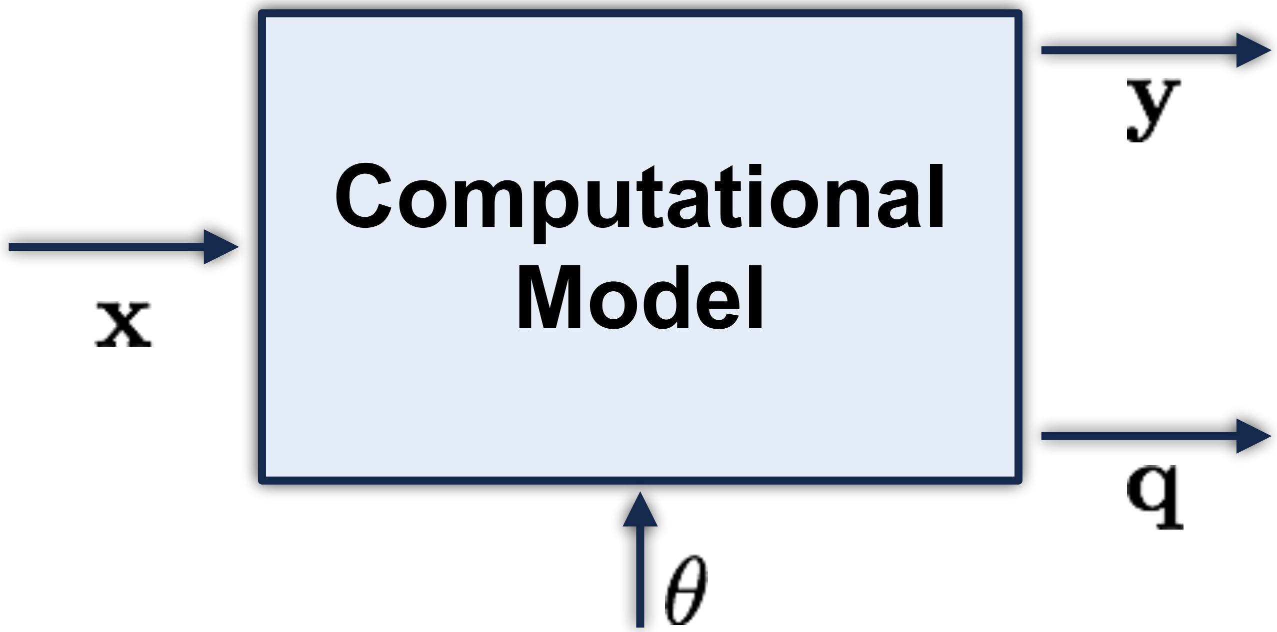
The causes of the difference between the true value of the real world process and the code-value are:

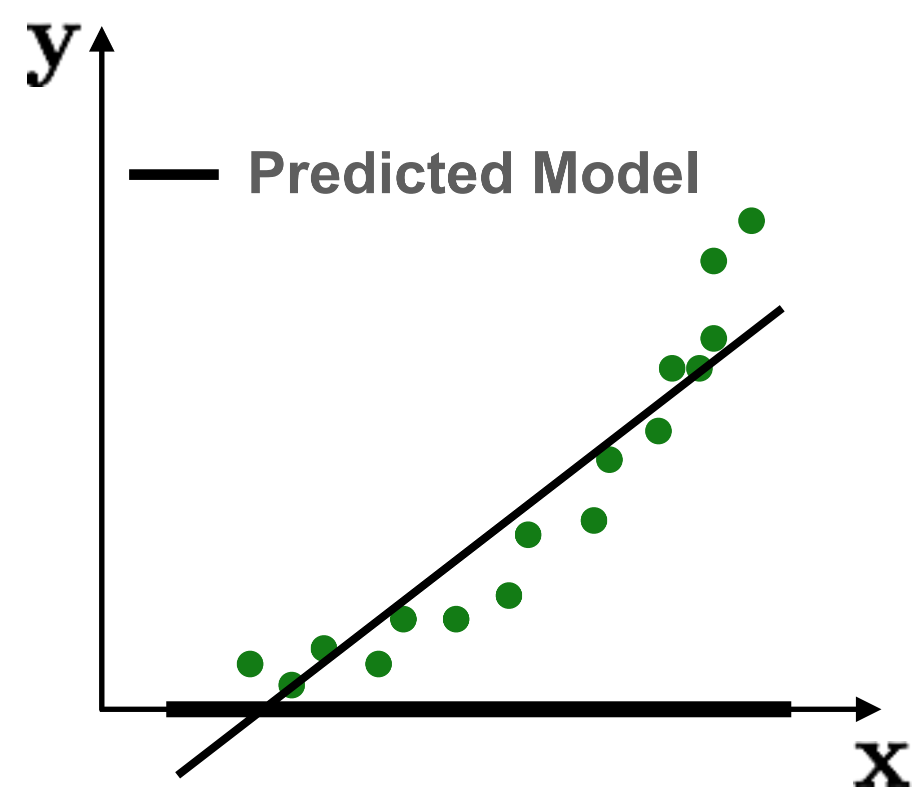
- Experimental or Measurement or Observation Error: ● vs – –
- Model Uncertainty or Structural Inadequacy: ▬ vs ●
- Parameter Uncertainty: ▬ vs ▬



Classic Approach to Validation

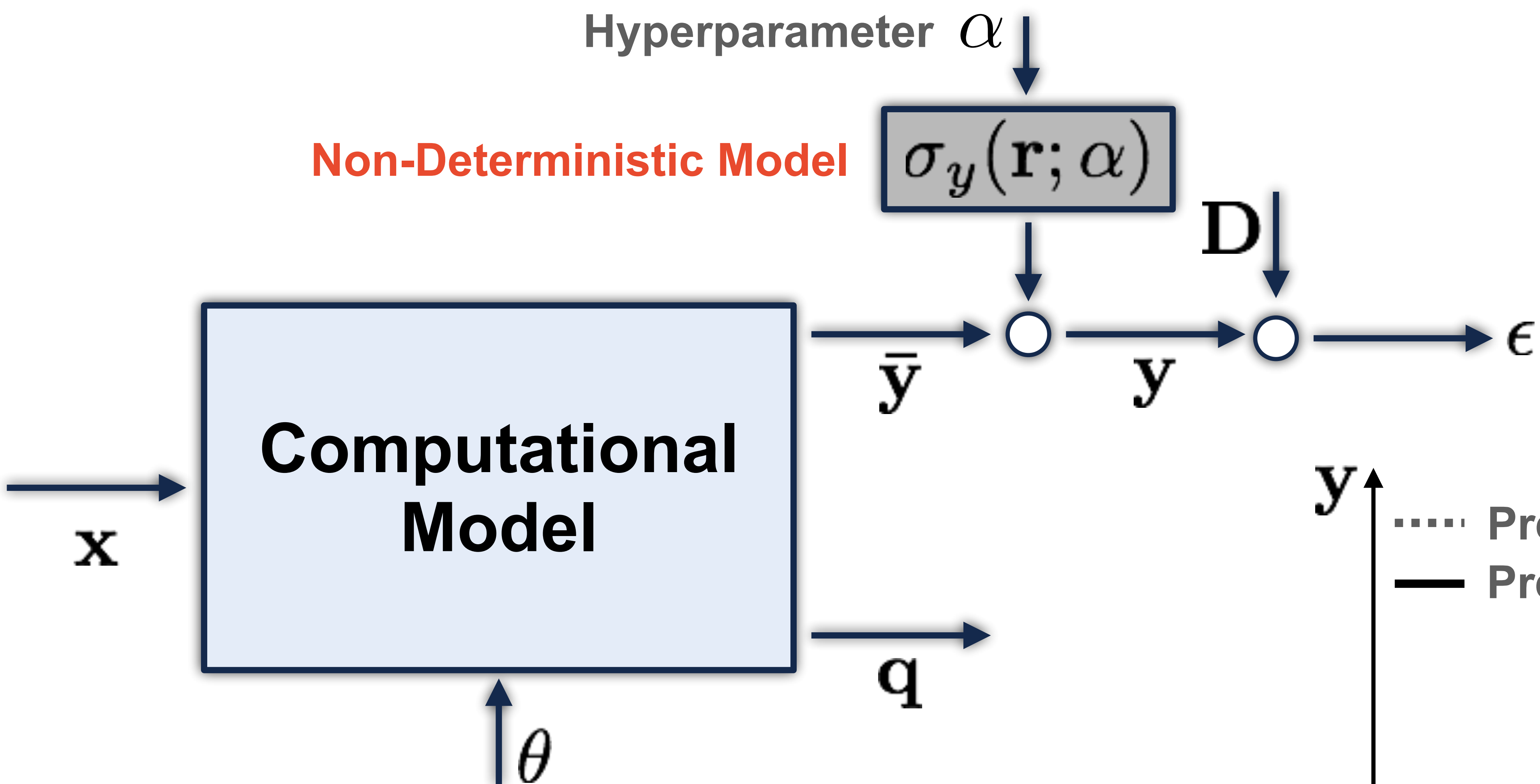
One approach [2] for taking into account the uncertainties on the predictions relies on "appending" a statistical model directly to the observable quantities:





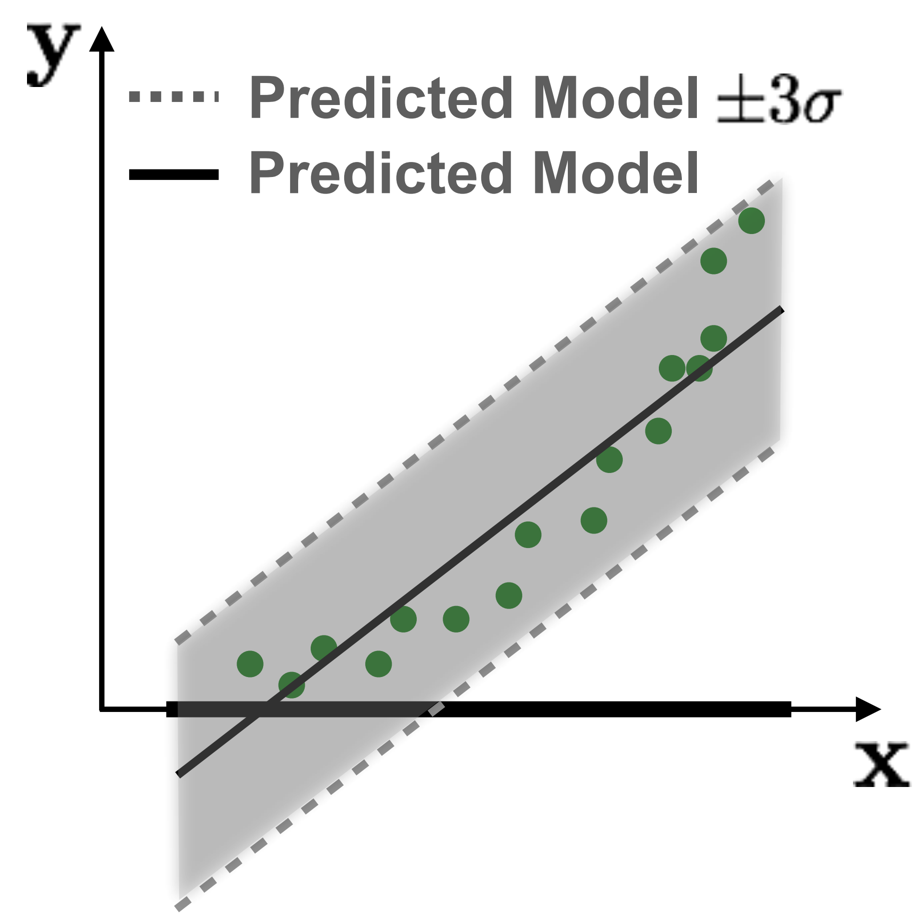
Classic Approach to Validation

One approach [2] for taking into account the uncertainties on the predictions relies on “appending” a statistical model directly to the observable quantities:



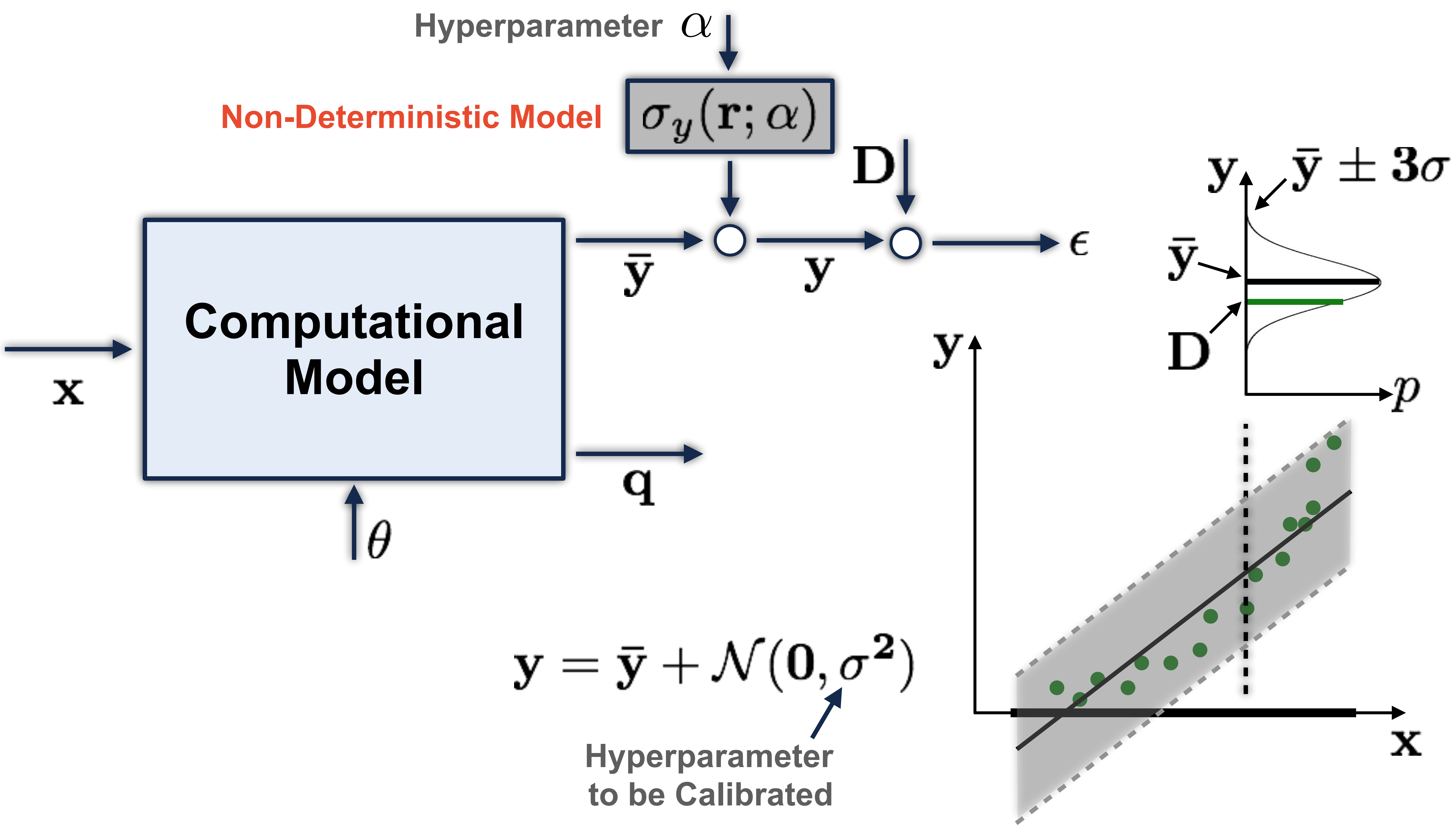
$$\mathbf{y} = \bar{\mathbf{y}} + \mathcal{N}(\mathbf{0}, \sigma^2)$$

Hyperparameter
to be Calibrated



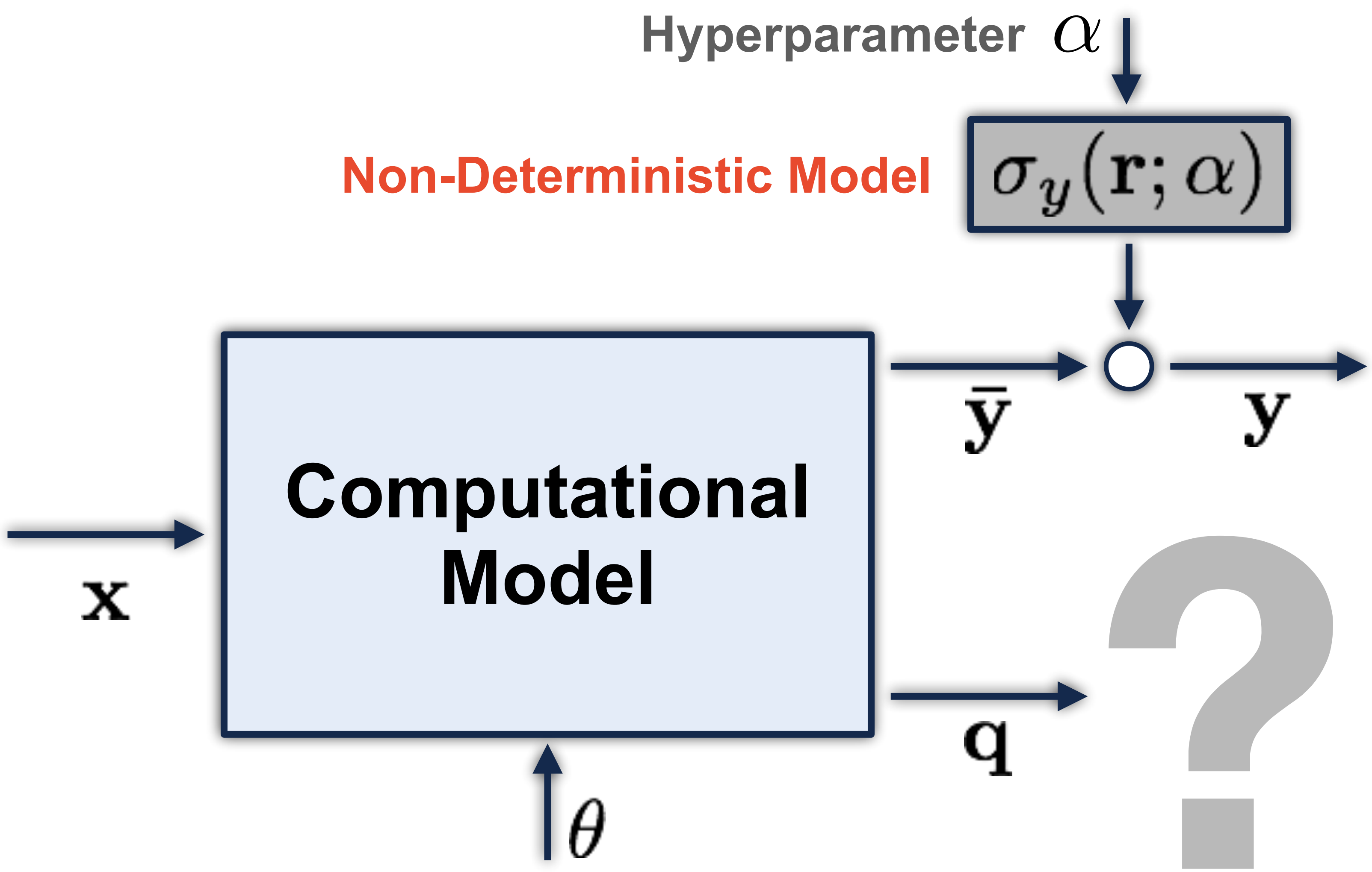
Classic Approach to Validation

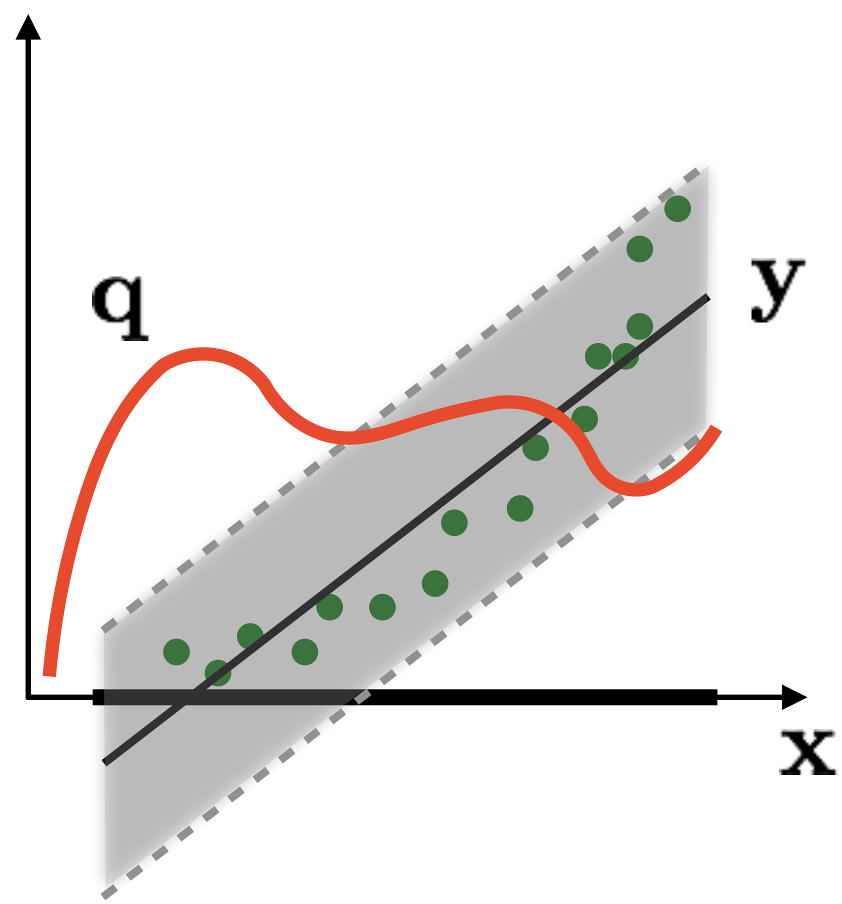
One approach [2] for taking into account the uncertainties on the predictions relies on “appending” a statistical model directly to the observable quantities:



Classic Approach to Validation

One approach [2] for taking into account the uncertainties on the predictions relies on "appending" a statistical model directly to the observable quantities:

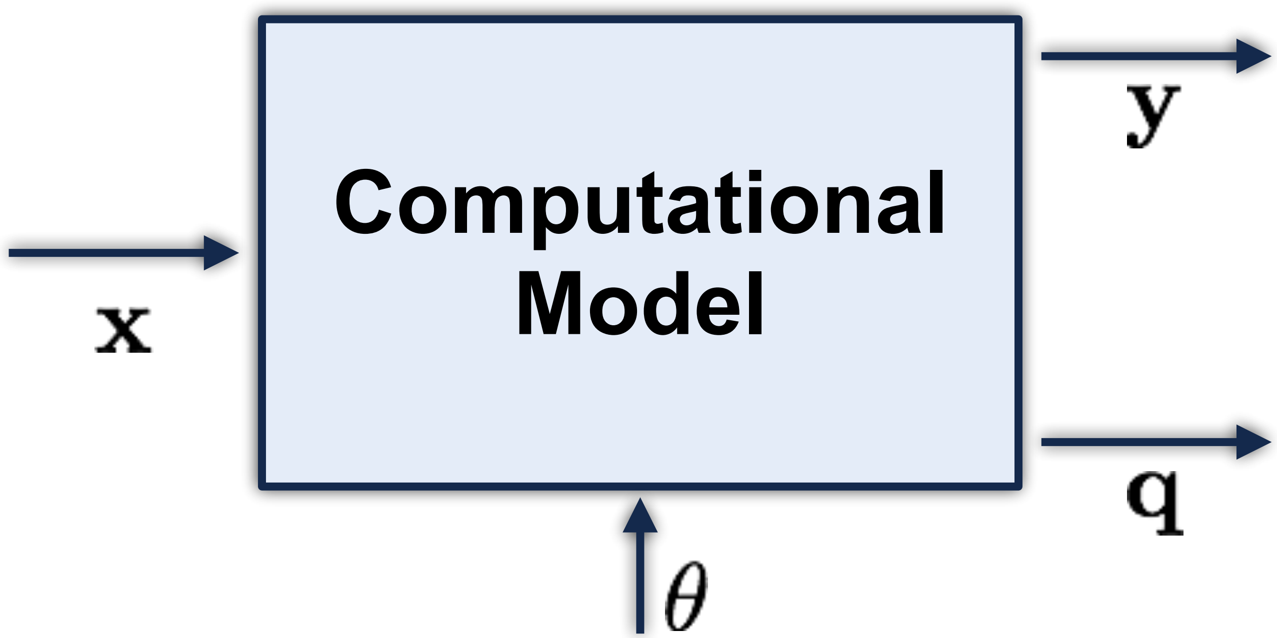


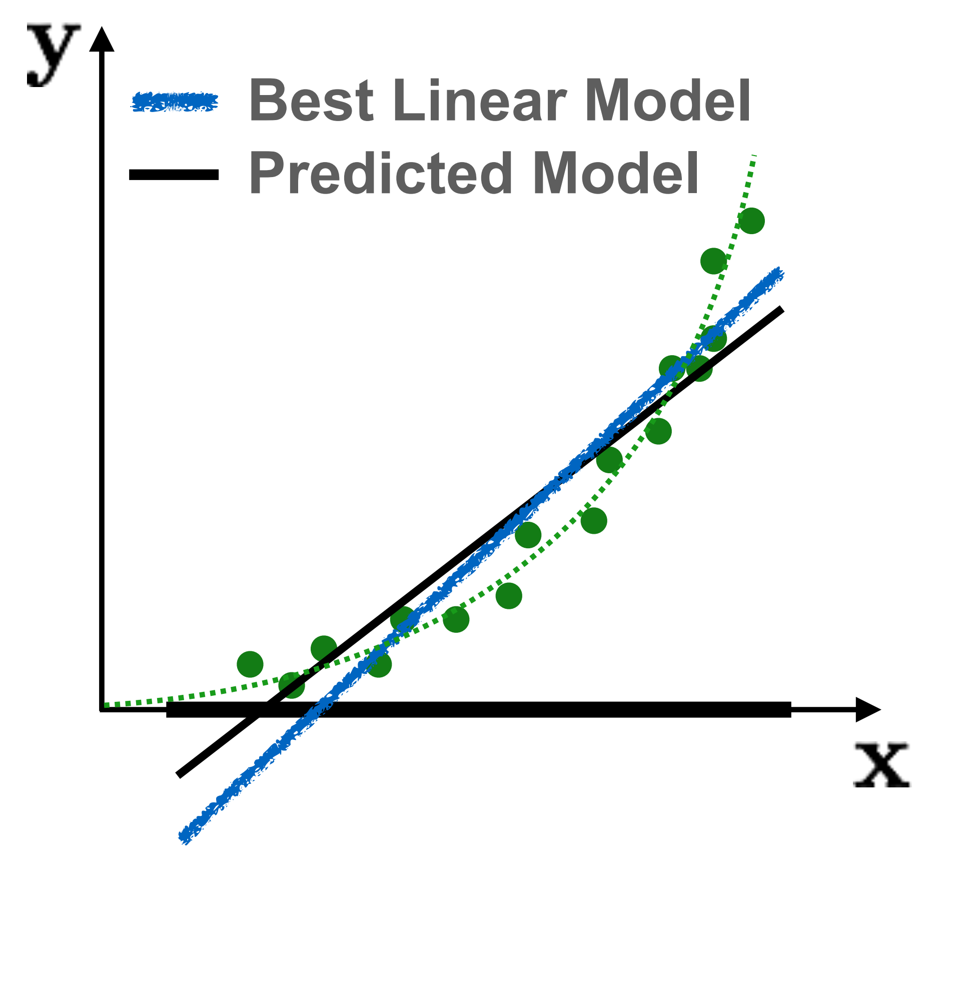


There is **no direct mapping between the observables to the QoIs.**
This treatment says nothing about the effect of the model discrepancy on the QoIs

Classic Approach to Validation

One approach [2] for taking into account the uncertainties on the predictions relies on "appending" a statistical model directly to the observable quantities:





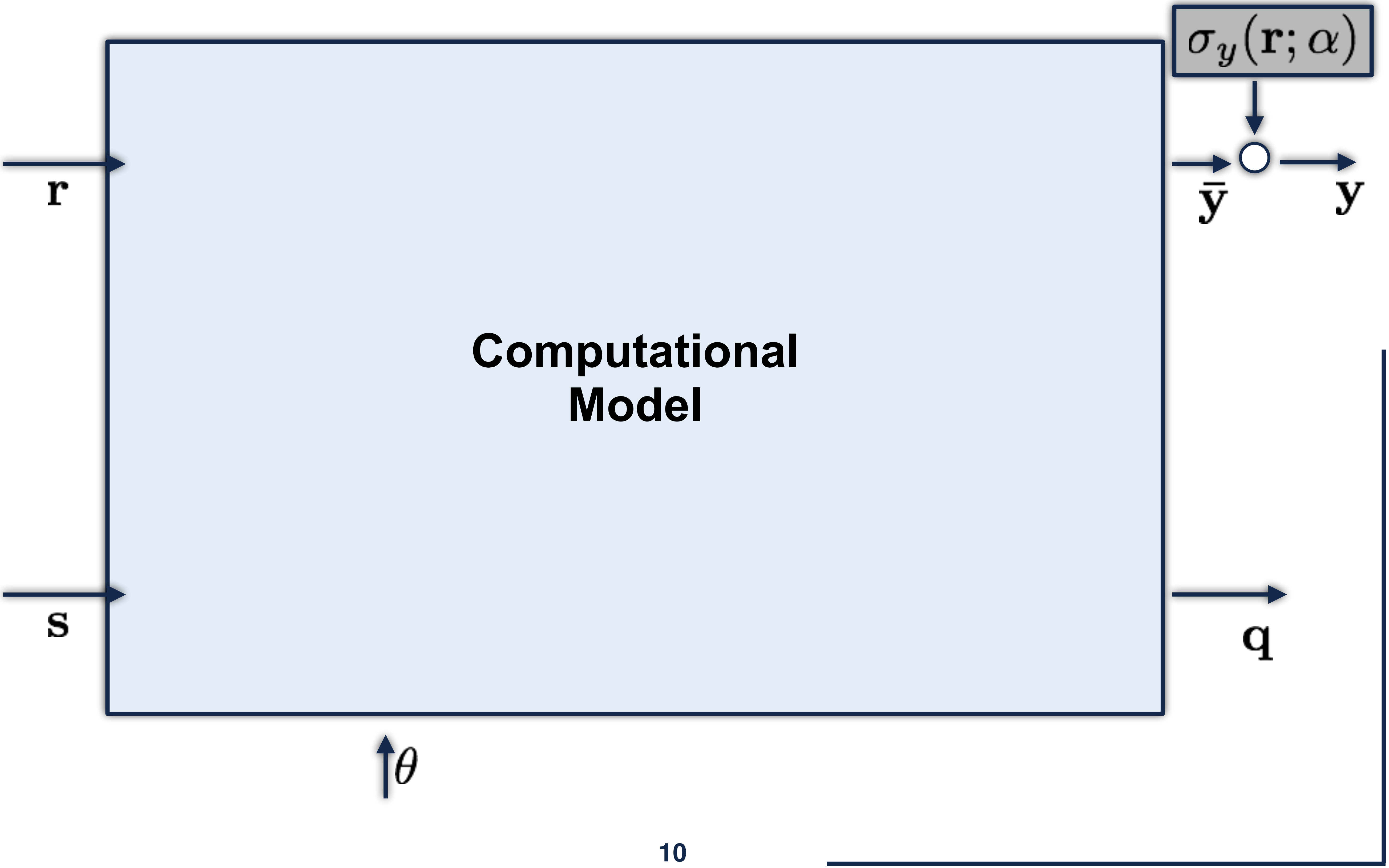
The causes of the difference between the true value of the real world process and the code-value are:

- Experimental or Measurement or Observation Error: ● vs - -
- Model Uncertainty or Structural Inadequacy: vs ●
- Parameter Uncertainty: —— vs

Due to Embedded Model!

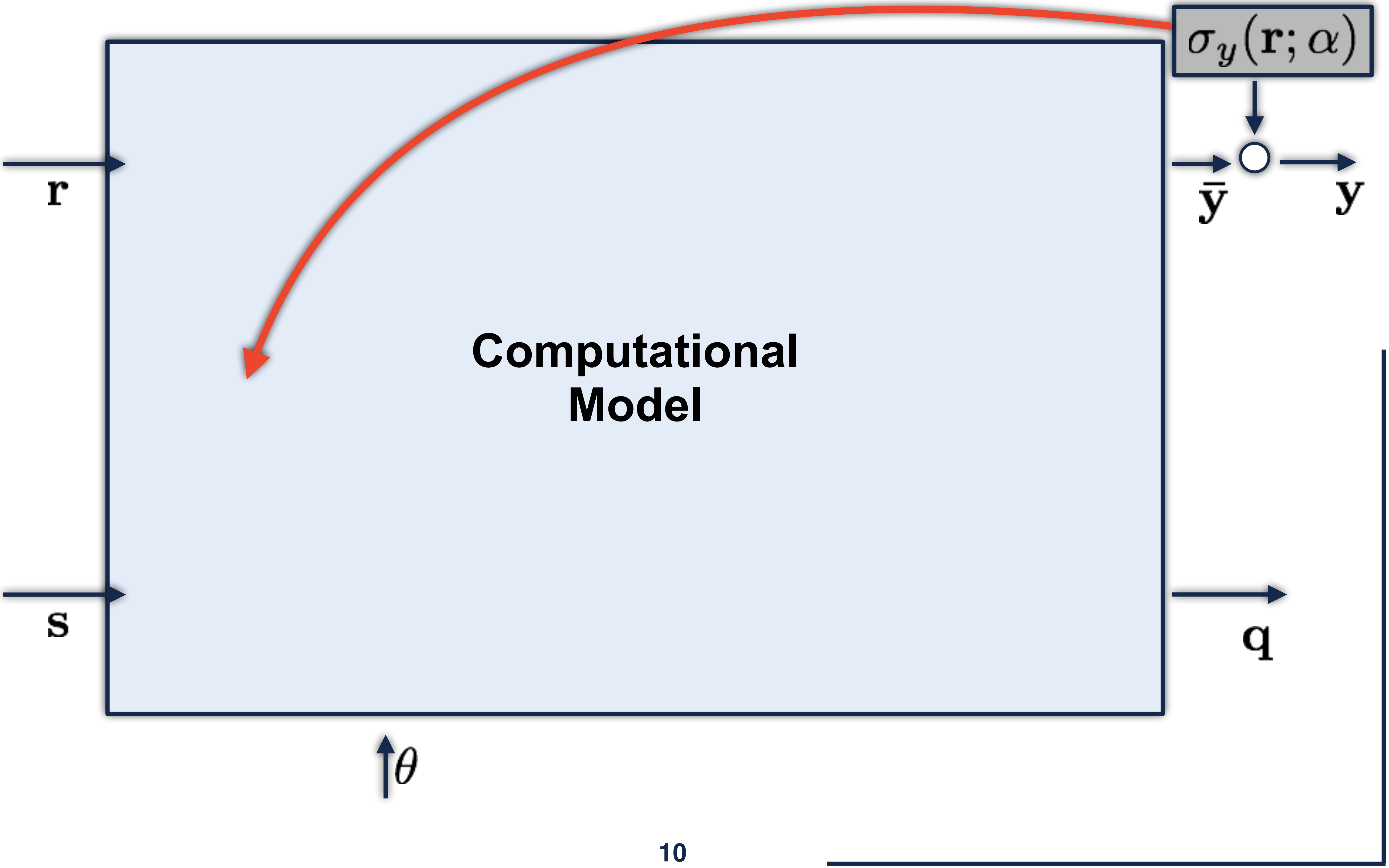
The Predictive Validation Process

Alternatively, we can move the non-deterministic model **upstream to the embedded component**, which is the main source of uncertainty (generally).



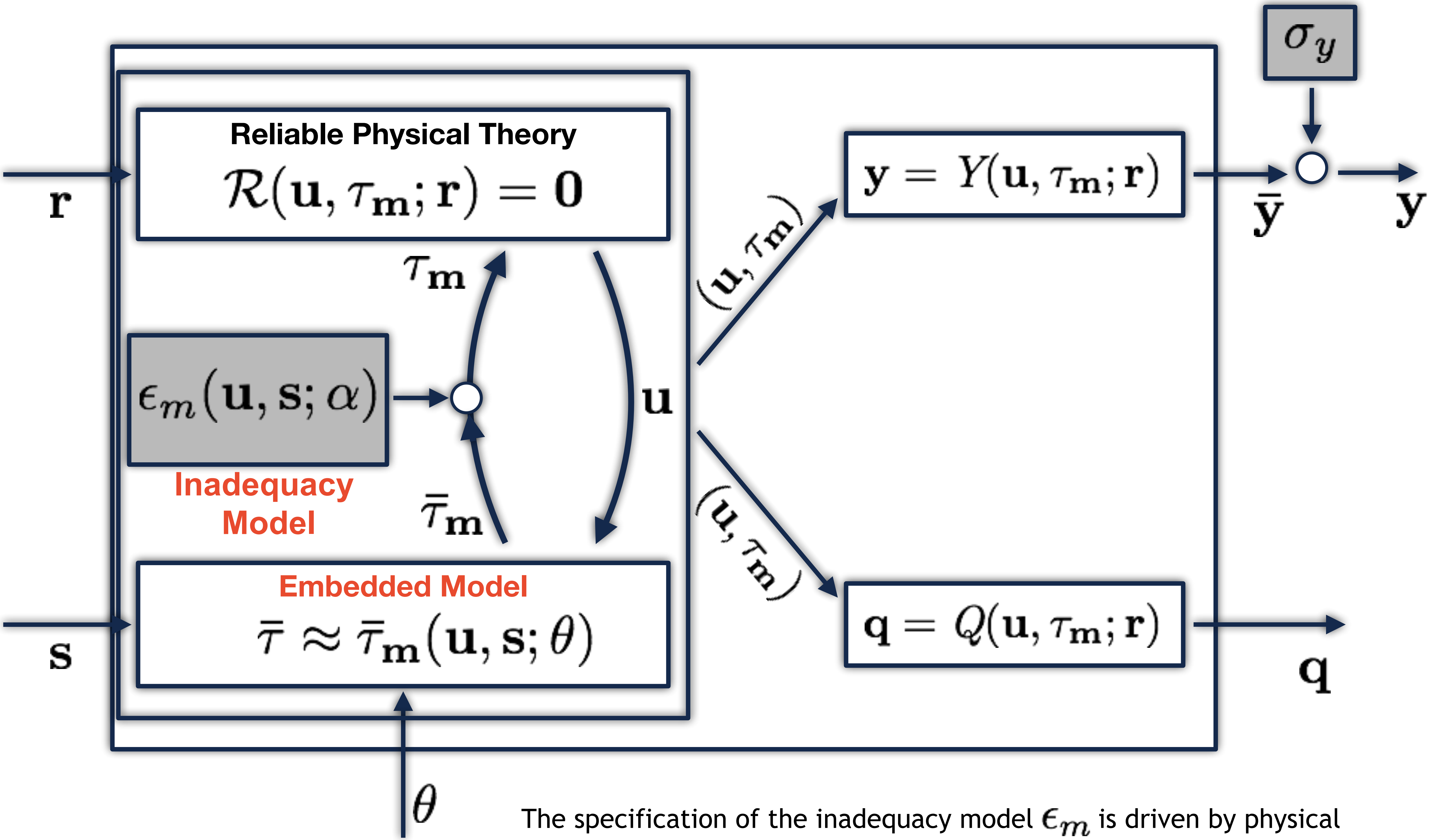
The Predictive Validation Process

Alternatively, we can move the non-deterministic model **upstream to the embedded component**, which is the main source of uncertainty (generally).



The Predictive Validation Process

Alternatively, we can move the non-deterministic model **upstream to the embedded component**, which is the main source of uncertainty (generally).



The specification of the inadequacy model ϵ_m is driven by physical knowledge about the nature of the error, and it is **problem-dependent**.

Introduction

Example: Mass-Spring-Damper, Approximated System

Reliable Physical Theory

$$\mathcal{R}(\mathbf{u}, \tau_{\mathbf{m}}; \mathbf{r}) = \mathbf{0}$$

$$\boxed{m\ddot{x} + c\dot{x} + kx = 0}$$

$$\mathbf{u} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \qquad \tau = \begin{bmatrix} c \\ k \end{bmatrix} \qquad \mathbf{r} = \begin{bmatrix} m \\ x(0) \\ \dot{x}(0) \end{bmatrix}$$

Approximated Embedded Model

$$\bar{\tau}_m = (\cancel{\mathbf{u}}, \cancel{\mathbf{v}}, \cancel{\mathbf{s}}, \theta)$$

$$\boxed{k = \text{const}}$$

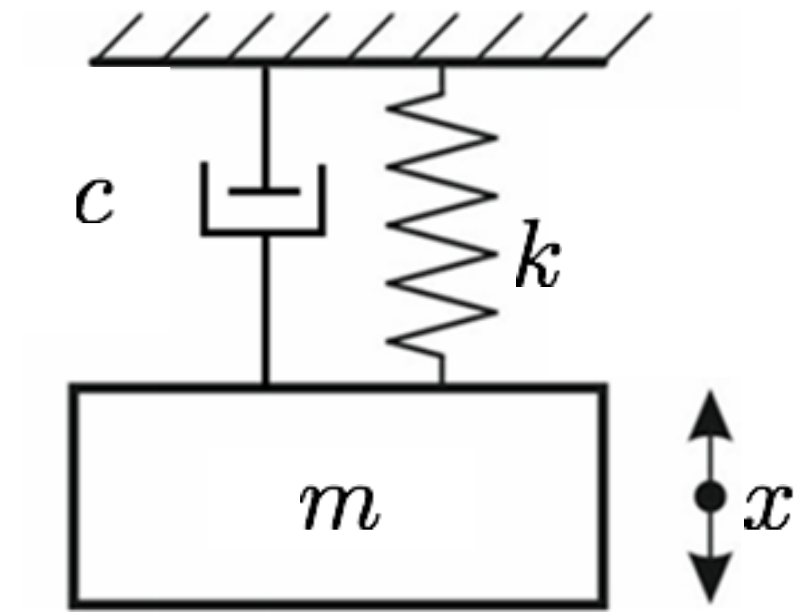
$$\theta = k$$

Inadequacy Model

$$\epsilon_m(\cancel{\mathbf{u}}, \cancel{\mathbf{s}}; \alpha)$$

$$\boxed{c \sim \mathcal{N}(\mu_c, \sigma_c^2)}$$

$$\alpha = \begin{bmatrix} \mu_C \\ \sigma_C \end{bmatrix}$$



$$\mathbf{y} = x$$
$$\mathbf{q} = \max(\dot{x})$$

Introduction

Note: It is critical that parameters be **<u>identifiable</u>** (i.e., parameters can be uniquely determined from observations). For:

$$m\ddot{x} + c\dot{x} + kx = 0$$

the parameter sets:

$$\theta = [m, c, k] \quad \text{and} \quad \theta = [1, c/m, k/m]$$

yield the same state values.

Approximated Embedded Mode

$$\bar{\tau}_m = (\cancel{u}, \cancel{v}, \cancel{s}, \theta)$$

$$\boxed{k = \text{const}}$$

$$\theta = k$$

Inadequacy Model

$$\epsilon_m(\cancel{u}, \cancel{s}; \alpha)$$

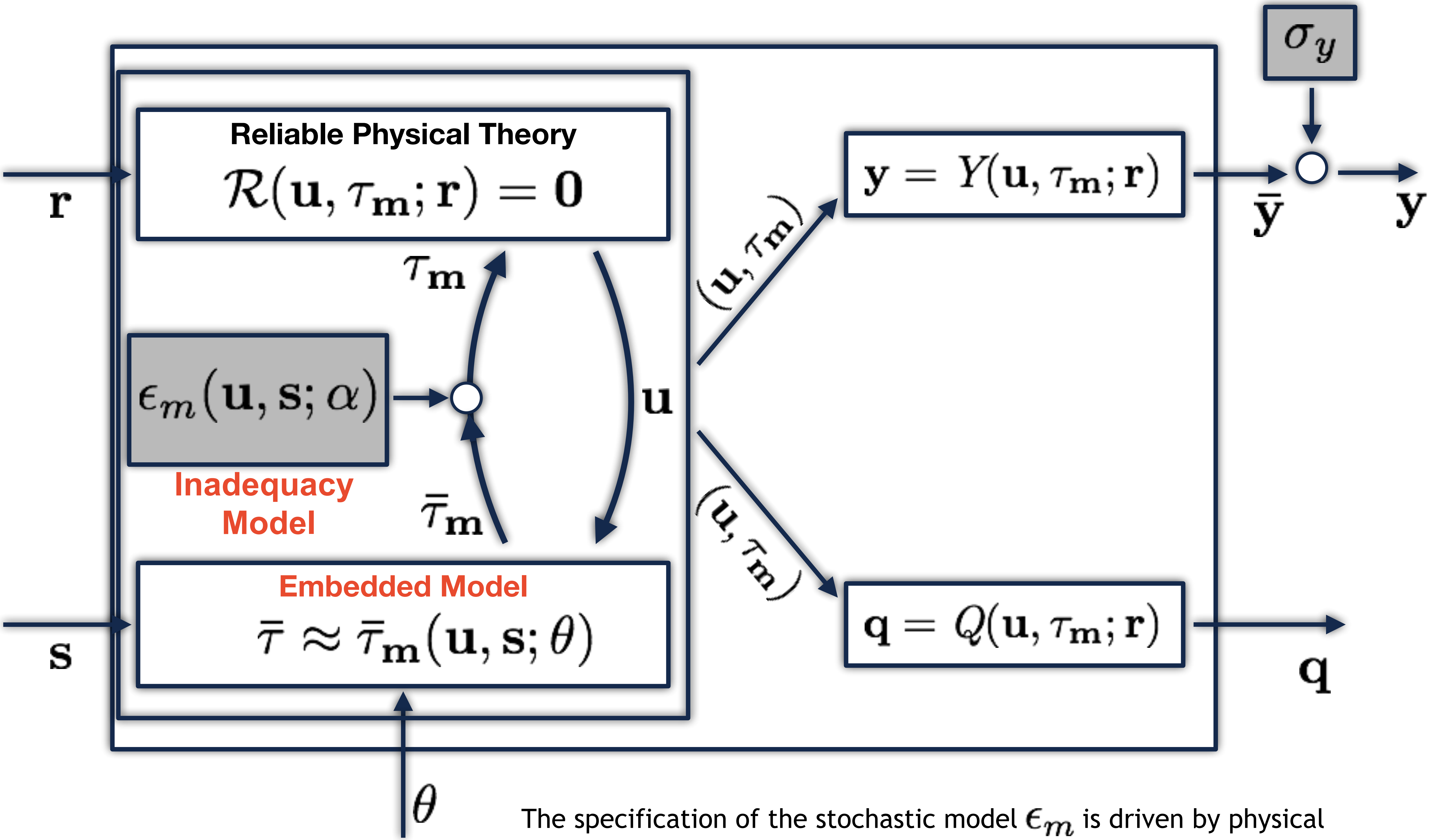
$$\boxed{c \sim \mathcal{N}(\mu_c, \sigma_c^2)}$$

$$\alpha = \begin{bmatrix} \mu_C \\ \sigma_C \end{bmatrix}$$

$$\boxed{\begin{aligned} \mathbf{y} &= x \\ \mathbf{q} &= \max(\dot{x}) \end{aligned}}$$

The Predictive Validation Process

Alternatively, we can move the non-deterministic model **upstream to the embedded component**, which is the main source of uncertainty (generally).



The specification of the stochastic model ϵ_m is driven by physical knowledge about the nature of the error, and it is **problem-dependent**.

The 3-Steps Reliability Assessment

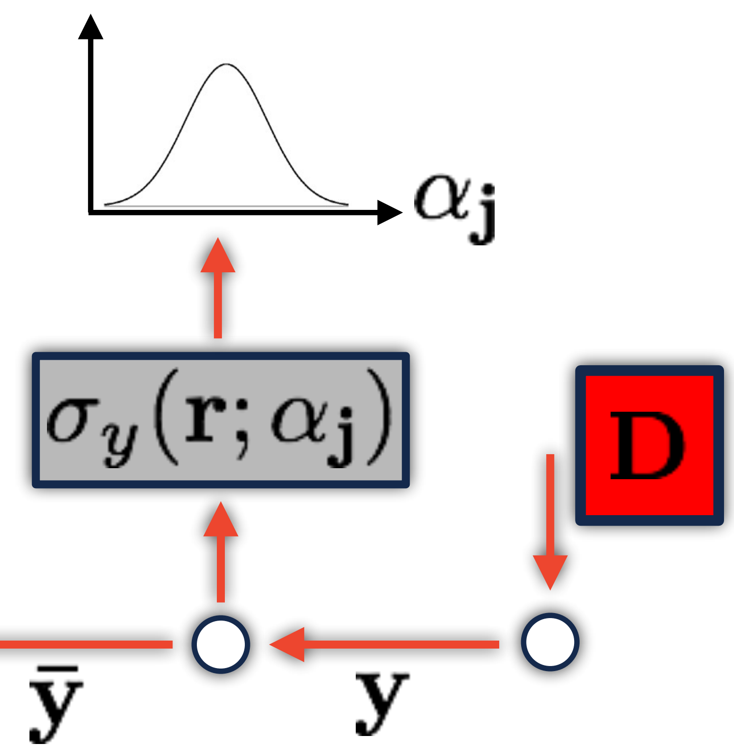
(Or Predictive Estimation): The probabilistic quantification of predicted experimental and computational outcomes with identified and quantified uncertainties.
It is comprised of 3 components:

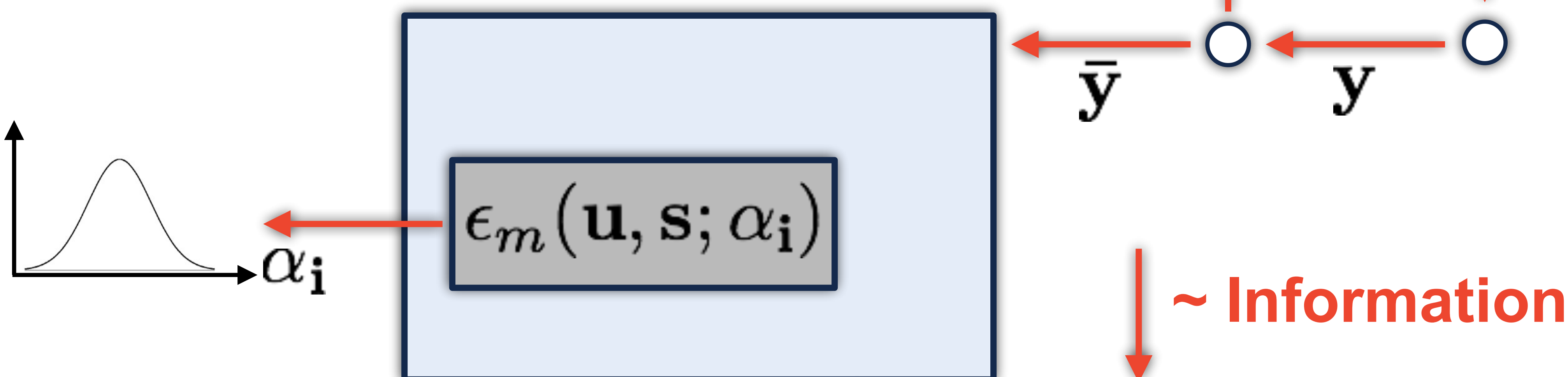
- **Model Calibration (**or **Inverse Problem)**
- **Validation (or Model Prediction)**
- **Predictive Assessment (or Estimation of the Validation Regime)**

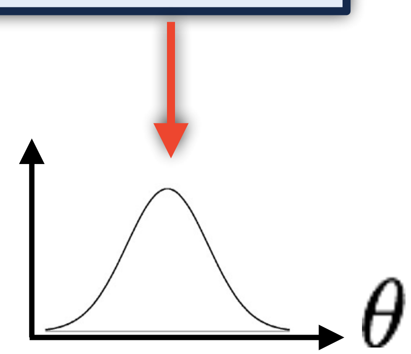
The 3-Steps Reliability Assessment

- **Model Calibration (or Inverse Problem)**
- **Validation**
- **Predictive Assessment**

The model is **informed by data**. Parameter values and their uncertainties are inferred from available observations by **solving an inverse problem.**







The use of probability to represent uncertainty naturally leads to the formulation of the calibration problem as Bayesian (**Bayesian Inference**):

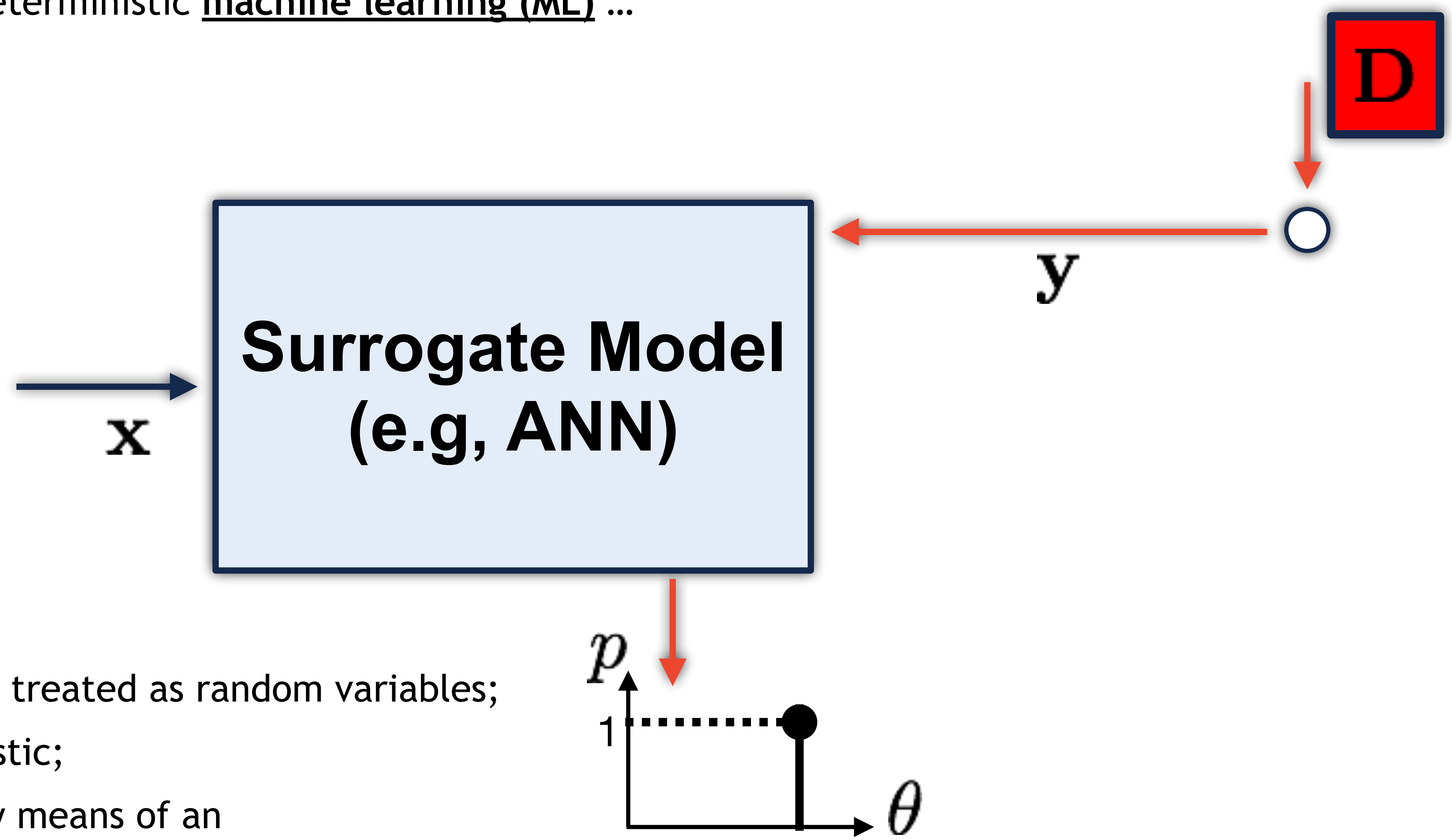
Posterior — Likelihood — Prior

$$p(\theta, \alpha | \mathbf{D}, \mathcal{M}) = \frac{\mathcal{L}(\theta, \alpha; \mathbf{D}, \mathcal{M}) \; \mathbf{p}(\theta, \alpha | \mathcal{M})}{\int \mathcal{L}(\theta, \alpha; \mathbf{D}, \mathcal{M}) \; \mathbf{p}(\theta, \alpha | \mathcal{M}) \; \mathbf{d}\theta \mathbf{d}\alpha}$$

The 3-Steps Reliability Assessment

- **Model Calibration (or Inverse Problem)**
- **Validation**
- **Predictive Assessment**

Similar to training process in deterministic **machine learning (ML)** …



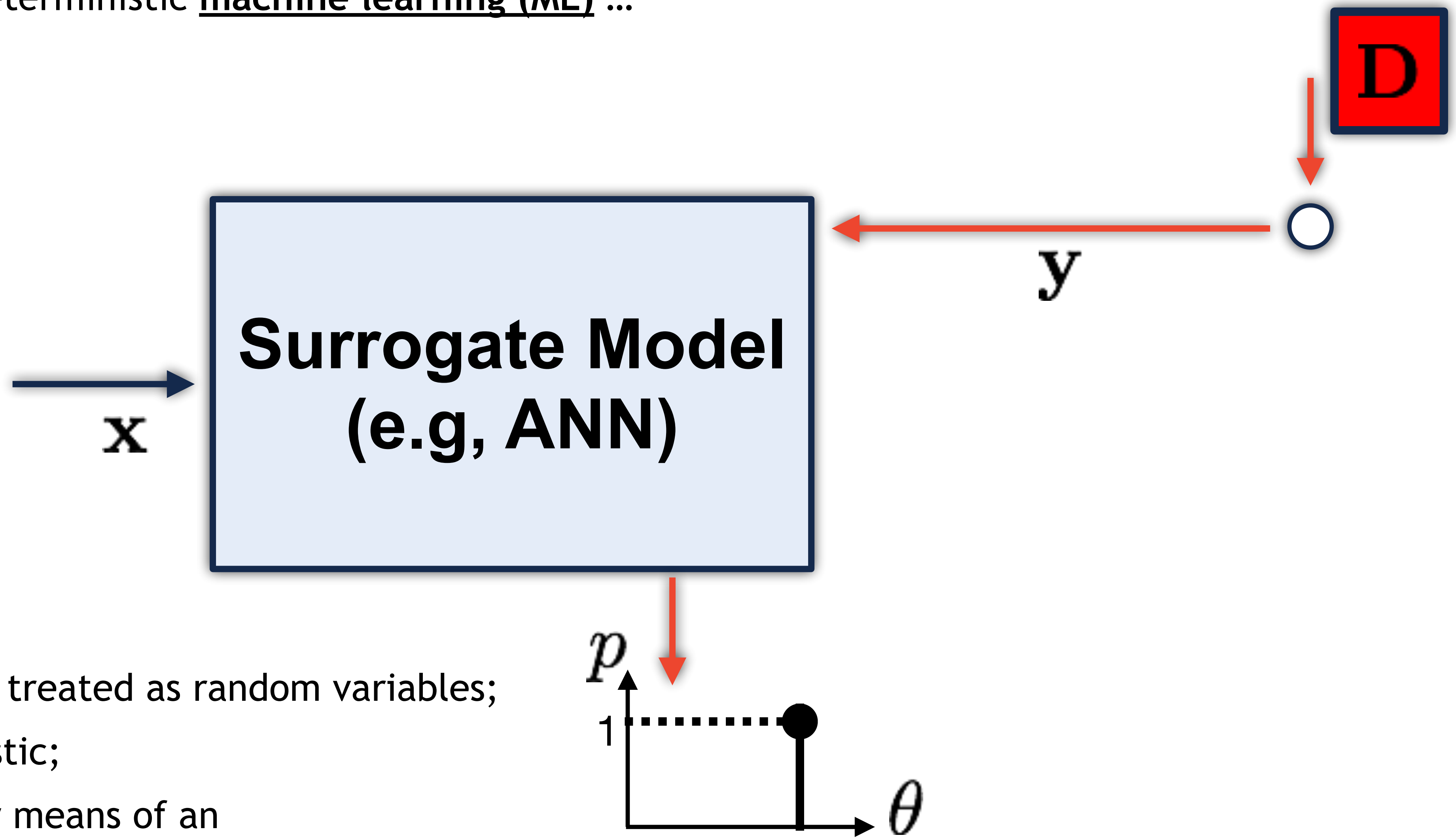
…, but in classic ML:

- The parameters are not treated as random variables;
- The model is deterministic;
- The training happens by means of an **optimization algorithm** (e.g., ADAMS);

The 3-Steps Reliability Assessment

- **Model Calibration (or Inverse Problem)**
- **Validation**
- **Predictive Assessment**

Similar to training process in deterministic **machine learning (ML)** …



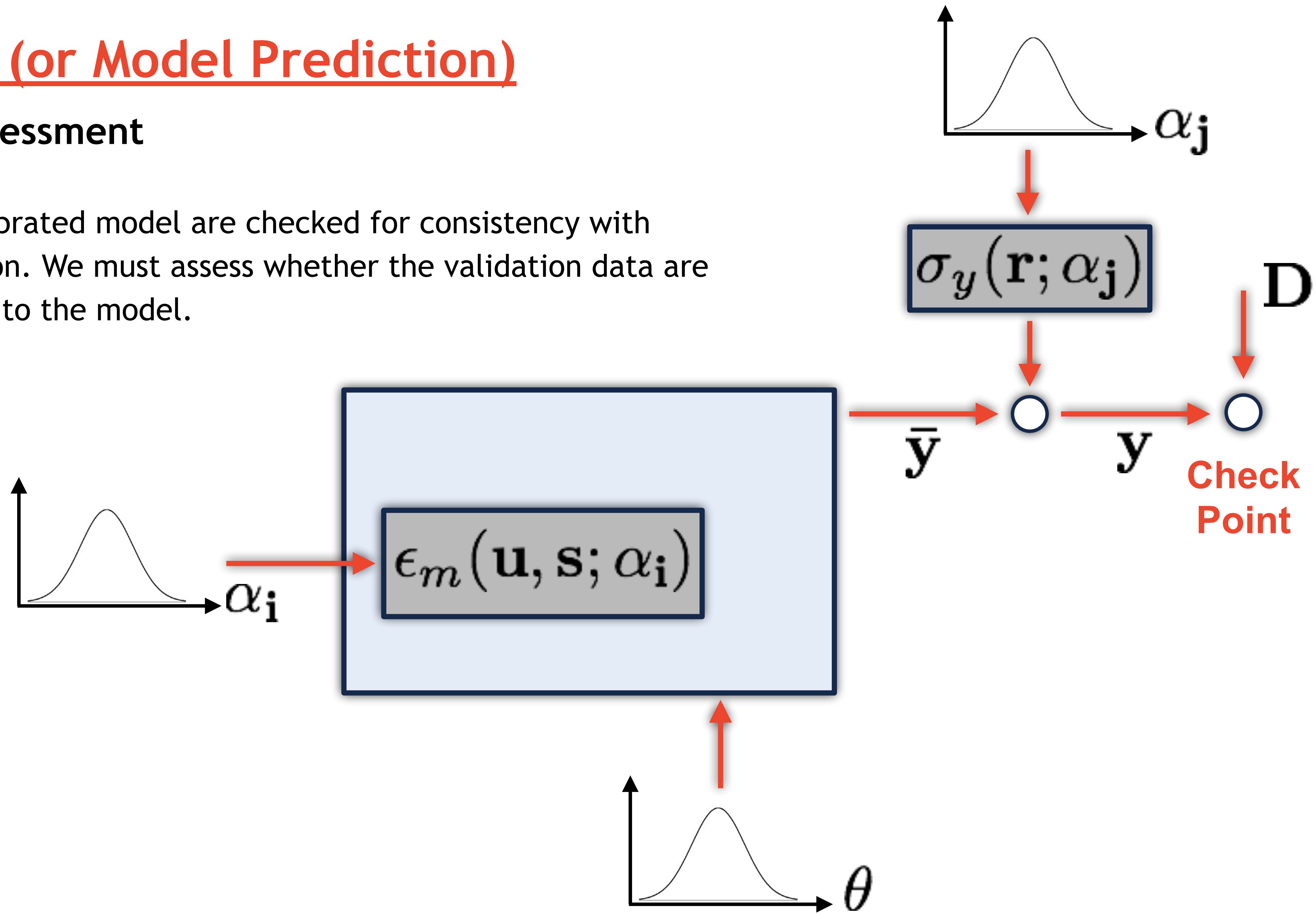
…, but in classic ML:

- The parameters are not treated as random variables;
- The model is deterministic;
- The training happens by means of an **optimization algorithm** (e.g., ADAMS);
- The computational model is generally not expensive;
- Plenty of data available.

The 3-Steps Reliability Assessment

- Calibration
- **Validation (or Model Prediction)**
- Predictive Assessment

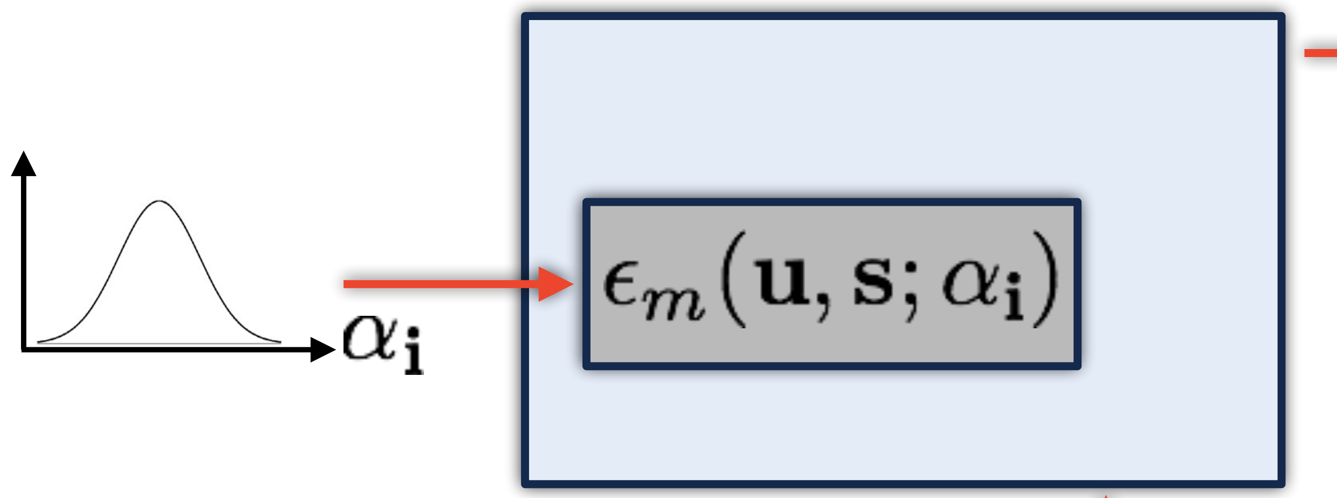
Outputs from a calibrated model are checked for consistency with available observation. We must assess whether the validation data are plausible according to the model.

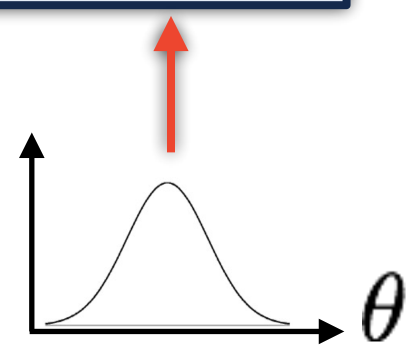


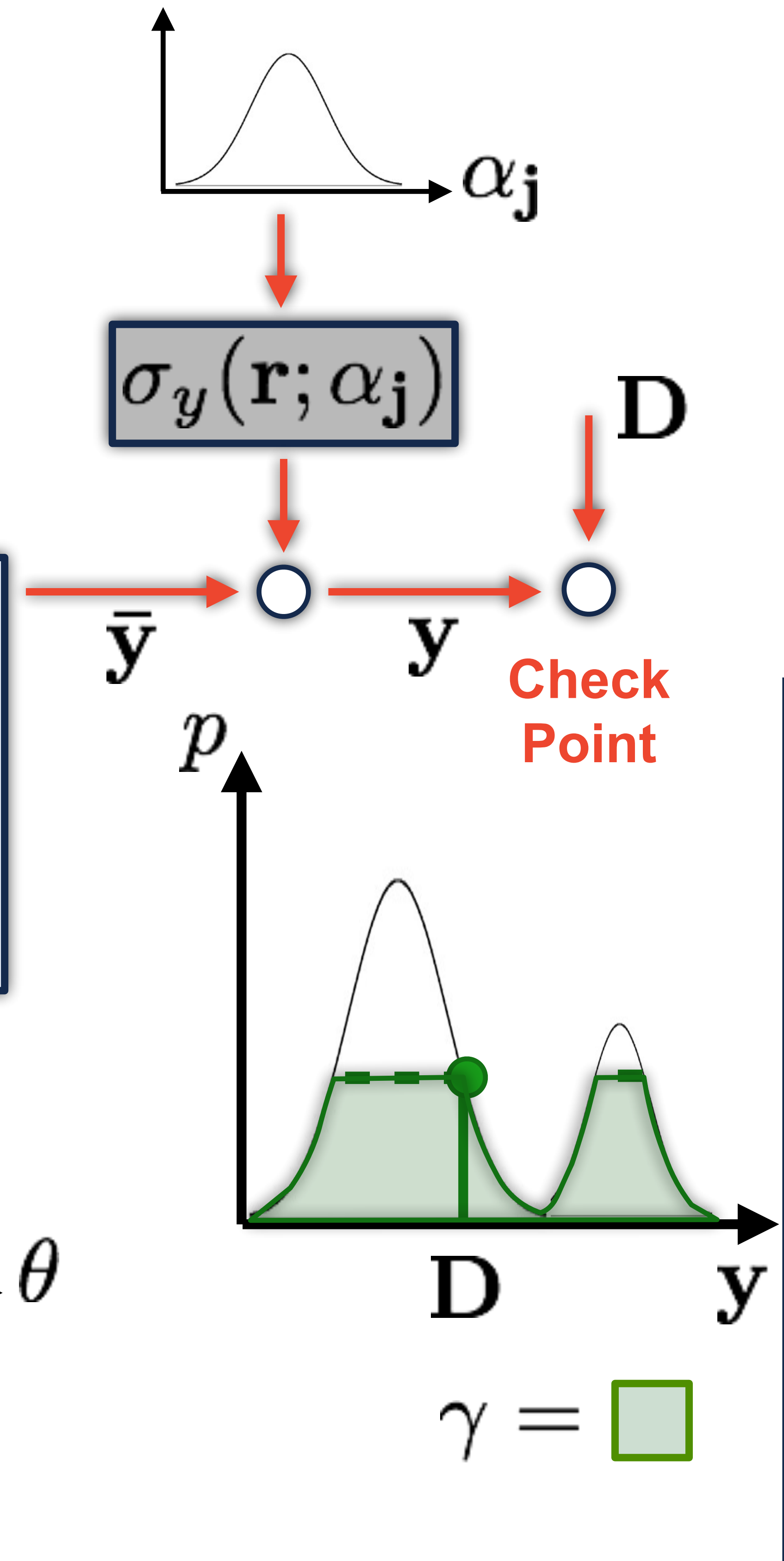
The 3-Steps Reliability Assessment

- **Calibration**
- **Validation (or Model Prediction)**
- **Predictive Assessment**

Outputs from a calibrated model are checked for consistency with available observation. We must assess whether the validation data are plausible according to the model.







There are number of possibilities for quantifying this plausibility, as the Bayesian credible intervals.

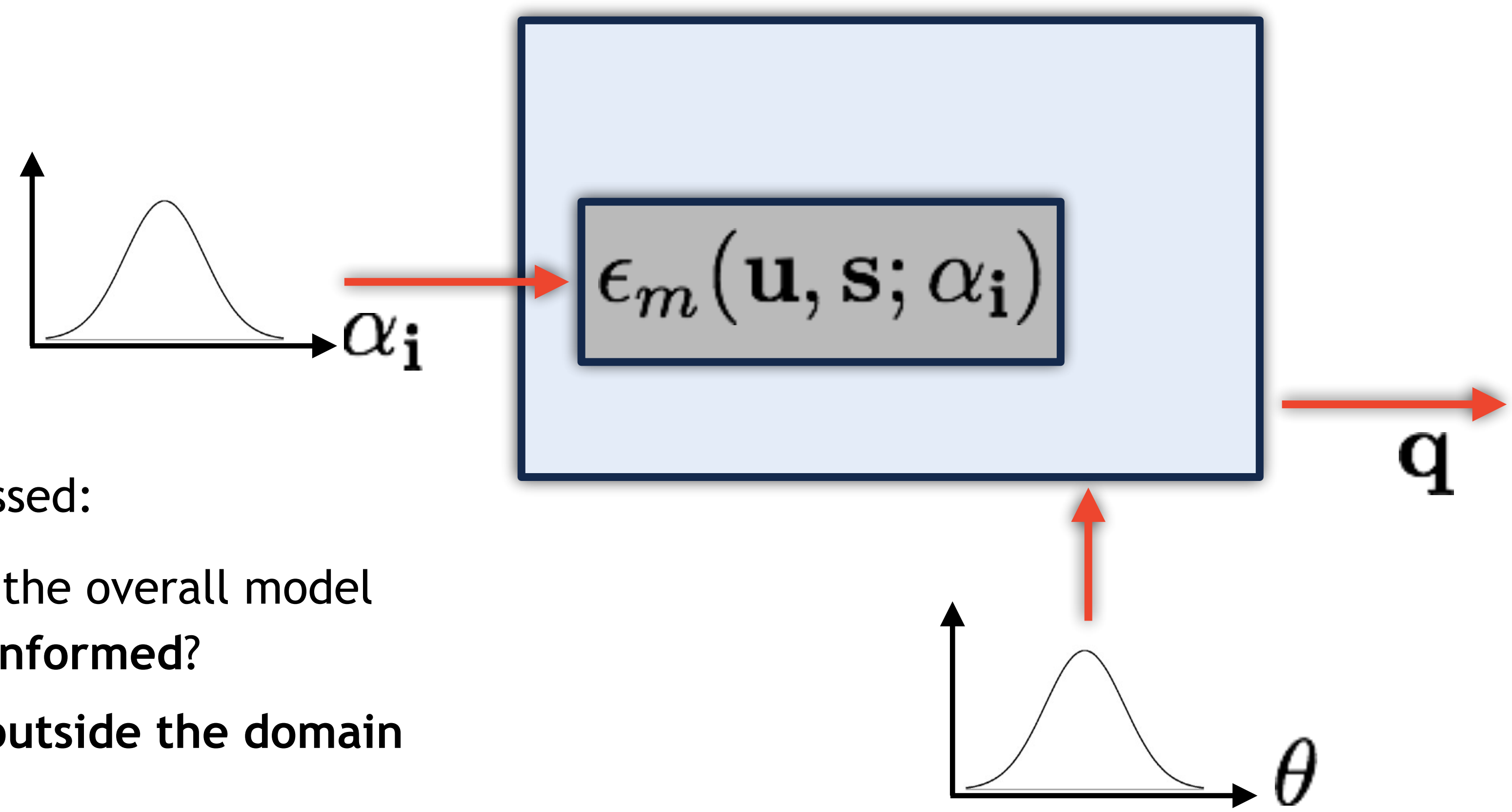
For example, we can use **highest posterior density (HPD)** credibility intervals.

When γ is smaller than some tolerance (e.g., 1.e-3), the data are considered an implausible outcome of the model.

The 3-Steps Reliability Assessment

- **Calibration** or **Inverse Problem**
- **Validation**
- **Predictive Assessment (or Estimation of the Validation Regime)**

Determines whether the calibration and validation phases were sufficiently informative and challenging to provide confidence in the reliability of the predictions of the QoIs.



Two primary questions need to be addressed:

- Are **QoIs sensitive to aspects** of the overall model that have **not been effectively informed**?
- Is the overall model being used **outside the domain of applicability**?

Moreover, is the prediction is determined to be credible, does it have **sufficiently small uncertainty** for our purposes?

Why is Predictive Science Fascinating?

- **The big picture stands out (System Engineering Point of View)**

- **The black boxes must be opened (Scientific Understanding)**

  Choices need to be made for:
  - Priors
  - Inadequacy Models
  - Domains of Applicability
  - Unknown Unknowns?
  - When to Ask for more Data, and What to Ask

    …

- **The curse of dimensionality is not an enemy (Recall to Efficiency)**
  - Sensitivity Analysis
  - Code Optimization
  - Model Reduction

    …

Probabilistic Programming

Codes:

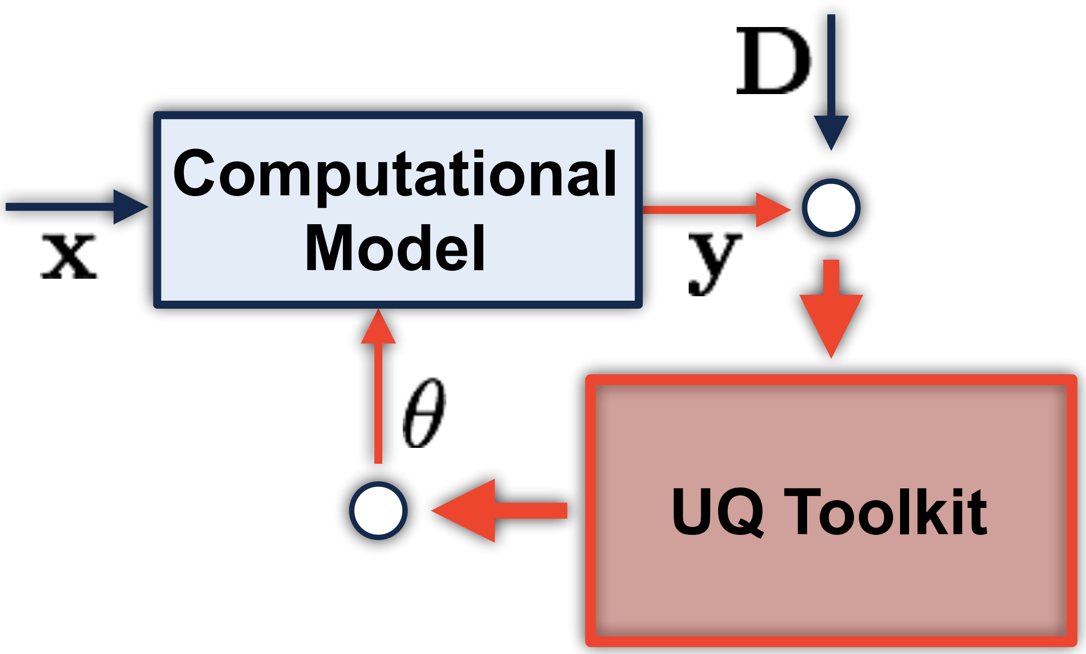
- SMUQ (UIUC), Fortran 2008

  https://www.politesi.polimi.it/bitstream/10589/118101/3/SimoneVenturi_MastersThesis.pdf

- Dakota (Sandia National Lab.s), C++

  https://dakota.sandia.gov/release-notes-headings/uncertainty-quantification-uq

Python Libraries:



- PyMC3

  https://docs.pymc.io/

- TensorFlow Probability

  https://www.tensorflow.org/probability

  Fusion Plasma Example: https://youtu.be/Bb1_zlrjo1c

Example 1:
Mass-Spring-Damper System

Case 1

Ideal Model:

$$m\ddot{x} + c\dot{x} + kx = 0$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \end{cases}$$

$$\begin{cases} m & = 1.0 \\ c & = 0.5 \\ k & = 3.0 \end{cases}$$

To the clean data points, we might add the noise $\epsilon \sim \mathcal{N}(0, \sigma_N^2)$

Computational Model:

$$m\ddot{x} + c\dot{x} + kx = 0$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \end{cases}$$

$$\begin{cases} m & = 1.0 \\ c & = \ ? \\ k & = \ ? \end{cases}$$

Likelihood Function is a Gaussian with σ_L

Case 1.1: $\sigma_N = 0.0 \quad \sigma_L = 0.01$

Case 1.2: $\sigma_N = 0.0 \quad \sigma_L = \ ?$

Case 1.3: $\sigma_N = 0.1 \quad \sigma_L = \ ?$

Case 2

Ideal Model:

$$\begin{cases} m\ddot{x} + c\dot{x} + kx = 0 \\ \dot{T} = c(T)\dot{x}^2 - \frac{T - T_0}{t_T} \end{cases}$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \\ T(0) = T_0 = 20.0 \end{cases}$$

$$\begin{cases} m & = 1.0 \\ c(T) & = \exp\left(\frac{T_0}{T} - 1\right) \\ k & = 3.0 \\ t_T & = 1.0 \end{cases}$$

To the clean data points, we might add the noise $\epsilon \sim \mathcal{N}(0, \sigma_N^2)$

Computational Model:

$$m\ddot{x} + c\dot{x} + kx = 0$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \end{cases}$$

$$\begin{cases} m & = 1.0 \\ c & = \ ? \\ k & = \ ? \end{cases}$$

Likelihood Function is a Gaussian with σ_L

Case 2.1: $\sigma_N = 0.0 \quad \sigma_L = 0.01$

Case 2.2: $\sigma_N = 0.0 \quad \sigma_L = \ ?$

Case 2.3: $\sigma_N = 0.1 \quad \sigma_L = \ ?$

Case 3

Ideal Model:

$$\begin{cases} m\ddot{x} + c\dot{x} + kx = 0 \\ \dot{T} = c(T)\dot{x}^2 - \frac{T-T_0}{t_T} \end{cases}$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \\ T(0) = T_0 = 20.0 \end{cases}$$

$$\begin{cases} m &= 1.0 \\ c(T) &= \exp\left(\frac{T_0}{T} - 1\right) \\ k &= 3.0 \\ t_T &= 1.0 \end{cases}$$

To the clean data points, we might add the noise $\epsilon \sim \mathcal{N}(0, \sigma_N^2)$

Computational Model:

$$m\ddot{x} + c\dot{x} + kx = 0$$

$$\begin{cases} x(0) = 4.0 \\ \dot{x}(0) = 0.0 \end{cases}$$

$$\begin{cases} m &= 1.0 \\ c &= \mathcal{N}(\mu_C, \sigma_C) \\ \mu_C &= \ ? \\ \sigma_C &= \ ? \\ k &= \ ? \end{cases}$$

Likelihood Function is a Gaussian with σ_L

Case 3.1: $\sigma_N = 0.0 \quad \sigma_L = 0.01$

Case 3.2: $\sigma_N = 0.0 \quad \sigma_L = \ ?$

Case 3.3: $\sigma_N = 0.1 \quad \sigma_L = \ ?$

Questions?

sventuri@illinois.edu
Talbot Lab, Room 302E