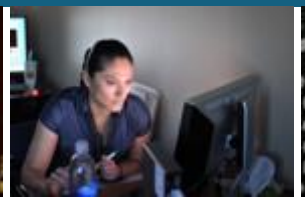
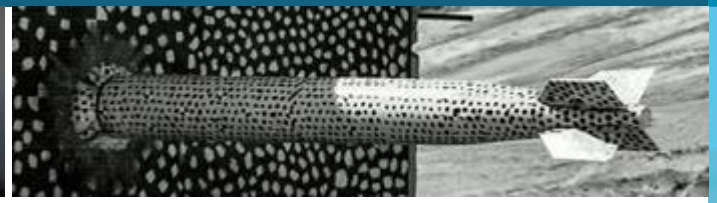
# Uncertainty Quantification for Cyber-Physical Pressurized Water Reactor Experiments






Dr. Taylor K. McKenzie (PI)

Co-authors: Christopher Lamb, Thomas D. Tarman



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

# Introduction

Ph.D. in Economics from University of Oregon in 2017

- Studied Industrial Organization, choice modeling, empirical models of competition
- Dissertation identified changes in pricing, competition, and technological change in US freight rail industry following deregulation

Joined Sandia National Laboratories in 2017

- Cyber risk and resilience analysis: Statistical analysis and uncertainty quantification to quantify risk to/resilience of cyber-physical systems using emulation and experimentation
- Interagency Nuclear Safety Review Board: Risk analyses of launches of spacecraft containing nuclear material, including Mars 2020/Perseverance
- Strategic Futures and Policy Analysis: Telecommuting/hybrid work posture, COVID-19 economic impact analyses, Global Futures studies on Economic Value & National Security and Knowledge Management

# Agenda

Uncertainty quantification for cyber-physical pressurized water reactor experiments (McKenzie et al. 2020) [1]

1. Background and system description
2. Research problem and uncertainty quantification approach
3. Kernel smoothing and Gaussian process approaches
4. Results
5. Discussion and impacts

# Background and System Description

# Background

Sandia has significant capabilities in cyber-physical emulation platforms/frameworks

- The SCEPTRE platform was developed to "analyze how cyber-initiated events affect the physical world," using a network emulation with hardware-in-the-loop capabilities to model, test, and validate cyber-physical systems [2]
- Used to assess risk and resilience of cyber-physical systems to natural and adversarial disruptions. Systems studied include critical infrastructures (e.g., power, oil/gas, chemical), space/satellite systems, industrial control systems (ICSs).

This project was funded by sponsors from DOE-NE, who were interested in the risk posed to cyber-physical nuclear power generation

- **Specifically interested in metrics to evaluate risk of catastrophic outcomes given a particular attack scenario**

Cyber-physical emulation is particularly well-suited for this analysis

- Nuclear reactor mechanics/dynamics are well understood and high-fidelity simulation models exist for these systems
- Cyber systems/components are less well-understood and more difficult to simulate with needed fidelity, so those components are emulated

# Emulation and Cyber Experiments

We want to evaluate how a critical system would respond to a given disruption

- Quantify impacts
- Identify and quantify effectiveness of mitigation strategies

Affecting operations of a critical system is infeasible

- Emulation provides ability to create a sufficient representation of a system, including hardware in the loop
- Emulation can be used to see how actual system would respond without affecting operations

Parameters of the system, emulation, or disruption may be uncertain

- Thus, outcomes are also uncertain and can be described by a distribution
- Uncertainty quantification (UQ) involves propagating input uncertainty to determine/approximate the distribution of outcomes

# Proof of Concept System: Pressurized Water Reactor

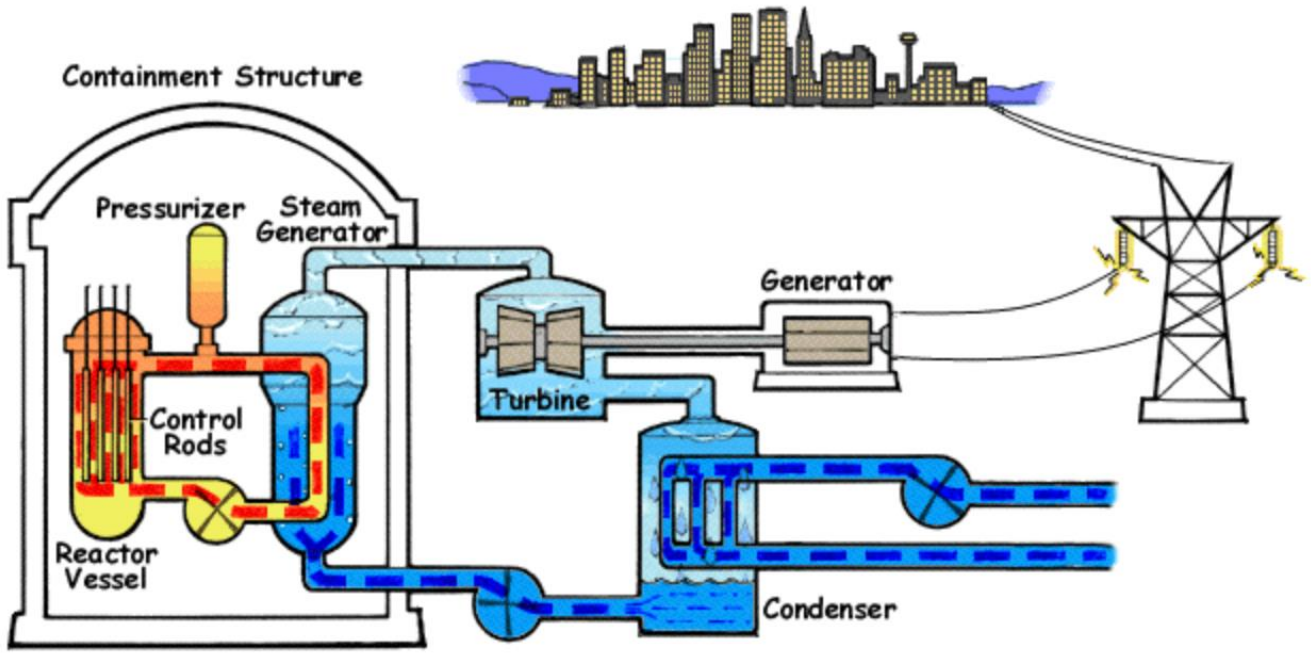
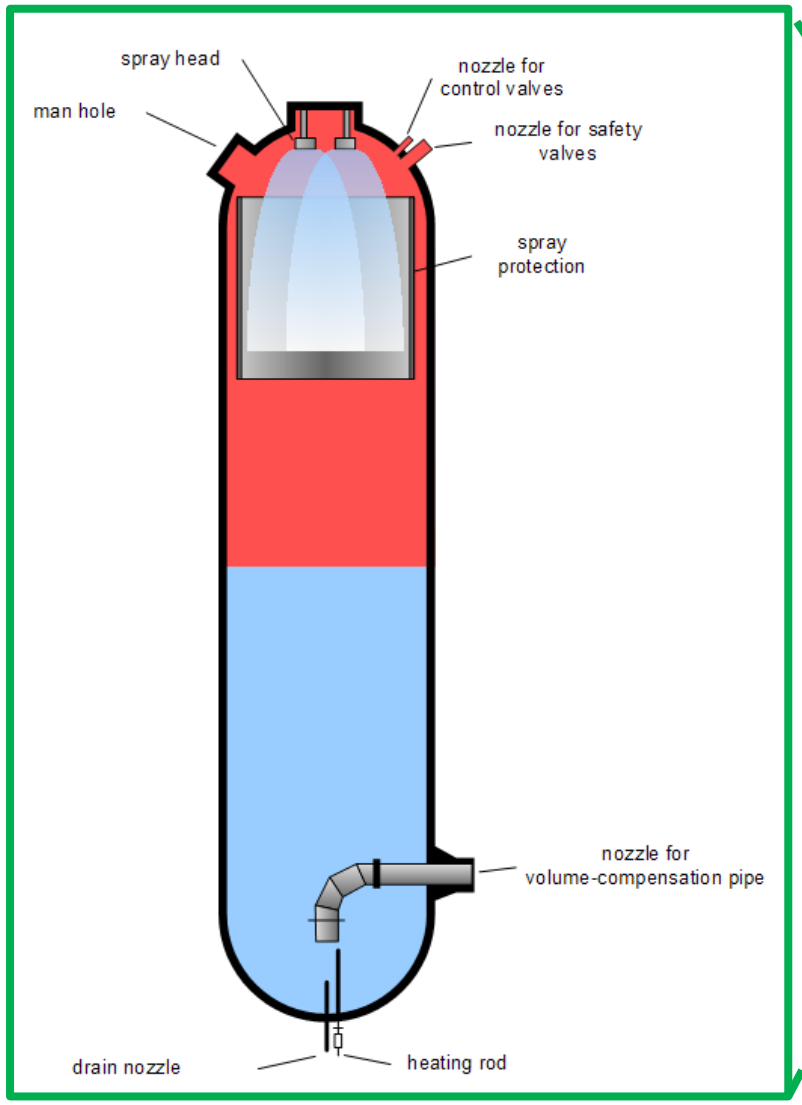



Image sources: NRC, Wikipedia

# Proof of Concept System: Pressurized Water Reactor



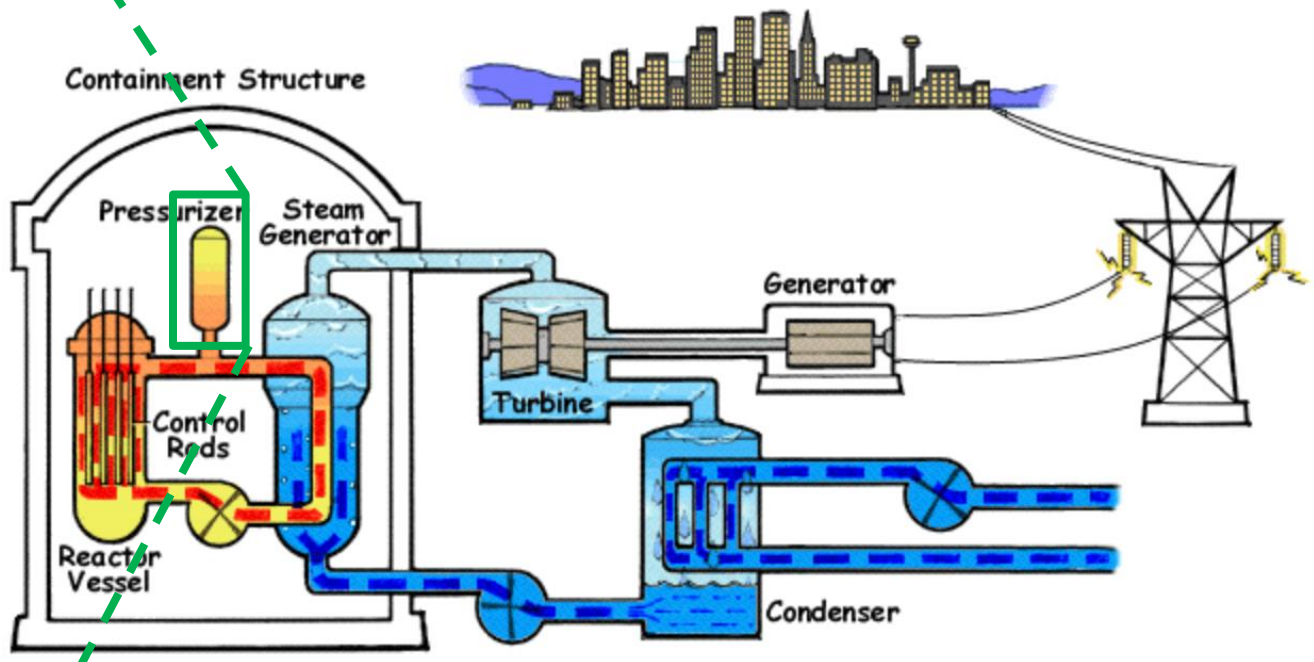




Image sources: NRC, Wikipedia

# Proof of Concept System: Pressurized Water Reactor

Pressurized water reactors (PWRs) are a nuclear reactor design with the following characteristics:

- Two coolant loops:
  - Primary loop where coolant cools and moderates the reactor
  - Secondary loop where coolant is heated by the primary and generates steam that drives turbines
- Pressurized primary reactor loop, which allows coolant to reach high temperatures while remaining liquid
  - Typical pressure: ~15-16 MPa
  - Typical temperature: ~345 °C

In a dual coolant loop design, only coolant in the primary loop requires significant radioactive containment

Pressurizer sits on the primary coolant loop and maintains appropriate pressure

- Too high could damage vessels/pipes/components. Too low could allow water to flash to steam.
- Electric heaters increase temperature/pressure of pressurizer, water sprayers reduce temperature/pressure

Pressurizer is a critical component of the system, and failures are highly consequential

**Dual loop design is unique to PWRs, motivating risk analysis specific to PWRs**

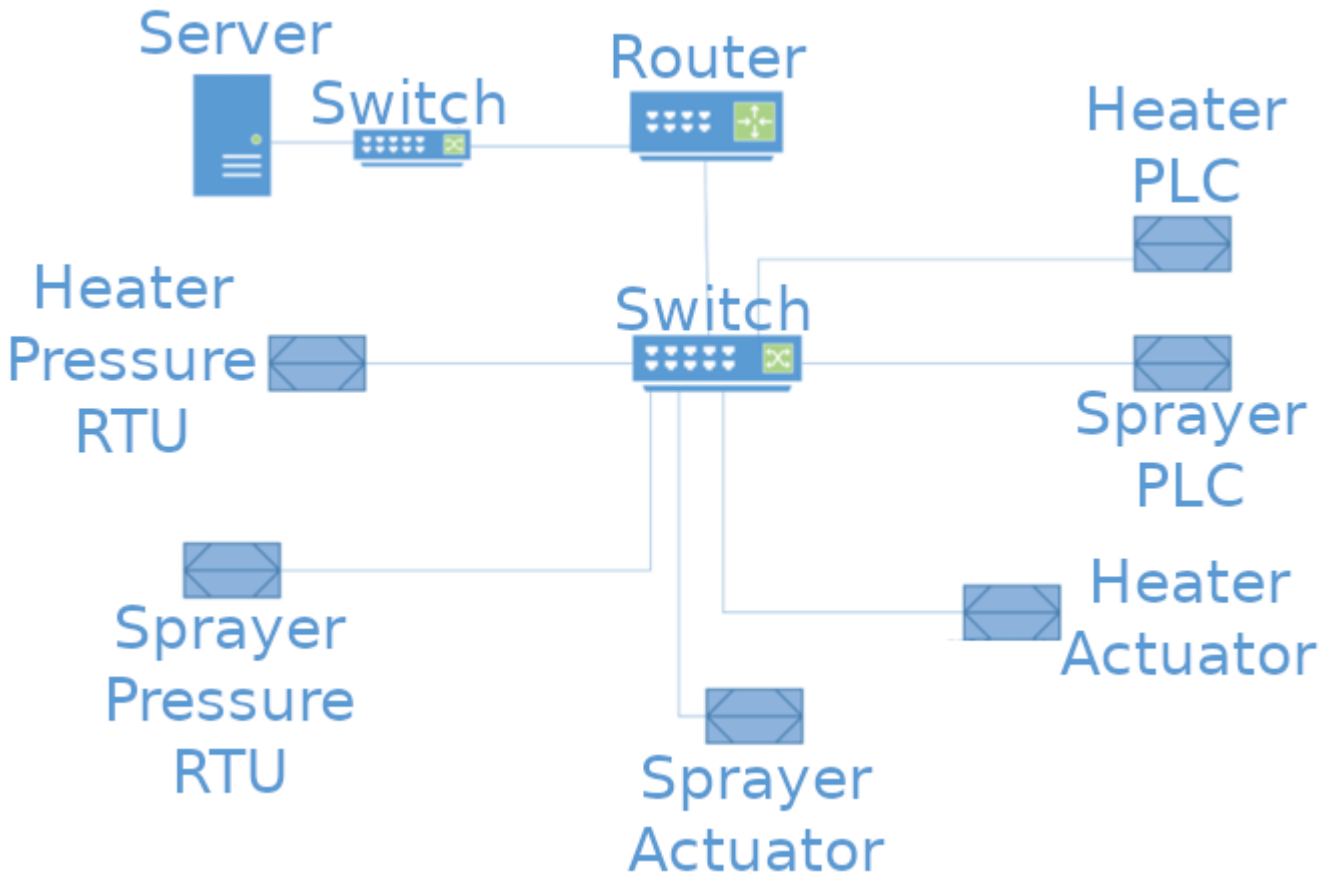
- PWRs are a very common design, accounting for 65% of nuclear power generation in the U.S. in 2021 [3]

# Proof of Concept System: Cyber-Physical PWR System

We draw on a cyber-physical emulation of a PWR system

**Abstract hardware**: Pressurizer (including electric variable heaters and spray valves to control pressure) described by numerical model implemented in Simulink

**Real software**: Supervisory control and data acquisition (SCADA) system, including server, router, network switches, remote terminal units (RTUs) to interface with pressure sensors, and programmable logic controllers (PLCs) to actuate heaters/sprayers



# Research Problem and Uncertainty Quantification Approach

# Research Problem

In general, previous risk/resilience analyses, red team assessments, and intelligence can be used to form a distribution of disruptions/attacks

- For this analysis, attacks will continually inject a specified control value to the pressurizer heater and/or sprayer. The injected values and which component(s) are targeted is assumed to be random.

Given a distribution of attacks, our goal is to estimate the probability that a catastrophic outcome will occur

- Severe consequences can occur if differential pressure on the pressurizer exceeds 1 MPa
- Emulation models for generic PWR systems were developed for previous projects. Those could be used to estimate failure probability using Monte Carlo methods.
- However, emulation is computationally expensive, making it infeasible to estimate probability of failure probability with acceptable level of uncertainty
- Need to use methods that can accommodate limited numbers of observations and correctly express uncertainty

Uncertainty quantification (UQ) methods can be used to estimate outcomes and evaluate uncertainty in those estimates when obtaining additional observations is expensive

# Disruption and Quantities of Interest

Disruption:

- An attacker is assumed to have gained access to the SCADA network and is able to (temporarily) override control values being fed to the heater and/or sprayer
- The SCADA network is assumed to not have any intrusion detection or mitigation system, so the attacker is free to generate network traffic with no risk of being discovered/stopped
- The attacker is assumed to target either heater, sprayer, or both, each with probability 1/3
- The attacker continuously injects false values to targeted actuator(s) for two minutes. Injected values are constant over the attack and are assumed to be uniformly distributed over [0, 100].

# UQ vs. Statistics

UQ methods were originally developed to serve as a proxy for deterministic computer simulations

- Sandia has a history of developing UQ methods and tools (e.g., Dakota) for simulations of nuclear weapons systems
- Simulations historically required major computing infrastructure and could only be feasibly evaluated a small number of times. However, analyses of those systems/simulations required many observations/evaluations.

UQ aims to describe the distribution of outcomes given uncertain parameters/inputs

How this is different than statistics:

- Statistics often focuses on estimating the relationship between inputs and outputs; this relationship is only important in UQ if it is useful for propagating uncertainty from inputs to outputs
- UQ applications utilize more machine learning methods (explainability is not necessarily a focus)
- In many UQ applications, the researcher is able to generate new data (in a smart way) at a cost

There are a few types of questions UQ analyses typically focus on

- In the upcoming slides
  - $X, x$ are inputs, $Y, y$ are outcomes
  - $X, Y$ are random variables, $x, y$ are specific values/realizations

# UQ Applications (1)

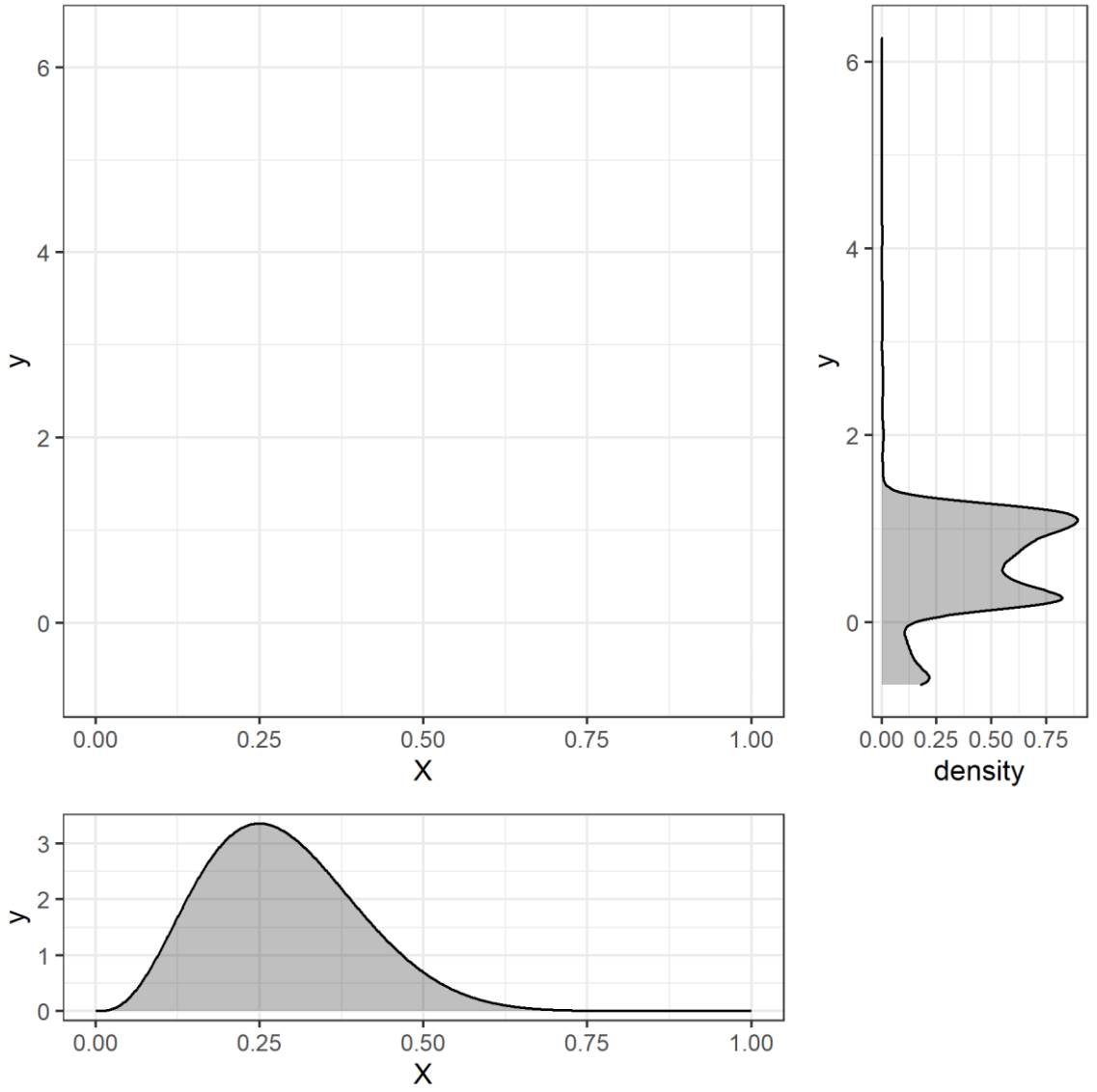
**Approach 1: Given a distribution of inputs $X$, describe the distribution of outcomes $Y$**

Common approach: Obtain a function $f$ such that $f(X) \equiv Y$

Since the distribution of $Y$ is the sole focus, $f$ need not describe any relationship that may exist between $X$ and $Y$

Polynomial chaos expansions are frequently employed in this application

- $f$ takes the form of an $n$th order polynomial
- New observations of $X$ can be generated in a way that minimizes expected distance between $f(X)$ and $Y$
- Useful as a proxy in Monte Carlo simulations when exact relationship between $X$ and $Y$ is not important



# UQ Applications (1)

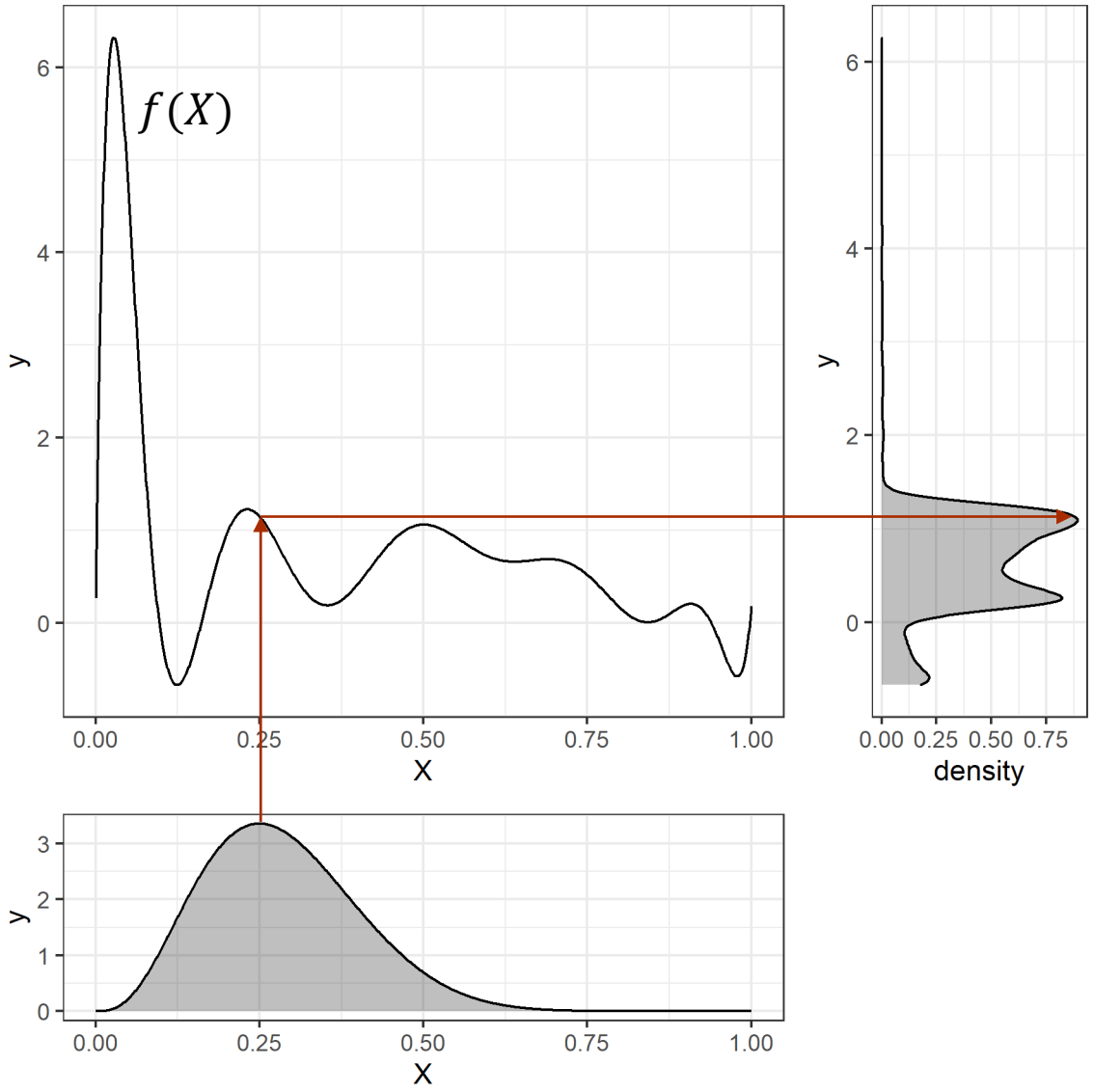
**Approach 1: Given a distribution of inputs $X$, describe the distribution of outcomes $Y$**

Common approach: Obtain a function $f$ such that $f(X) \equiv Y$

Since the distribution of $Y$ is the sole focus, $f$ need not describe any relationship that may exist between $X$ and $Y$

Polynomial chaos expansions are frequently employed in this application

- $f$ takes the form of an $n$th order polynomial
- New observations of $X$ can be generated in a way that minimizes expected distance between $f(X)$ and $Y$
- Useful as a proxy in Monte Carlo simulations when exact relationship between $X$ and $Y$ is not important



# UQ Applications (2)

**Approach 2: Given a value of inputs $x$, predict the deterministic outcome $y$**

There is some function $g$ such that conditional outcomes are described by $g(x)$

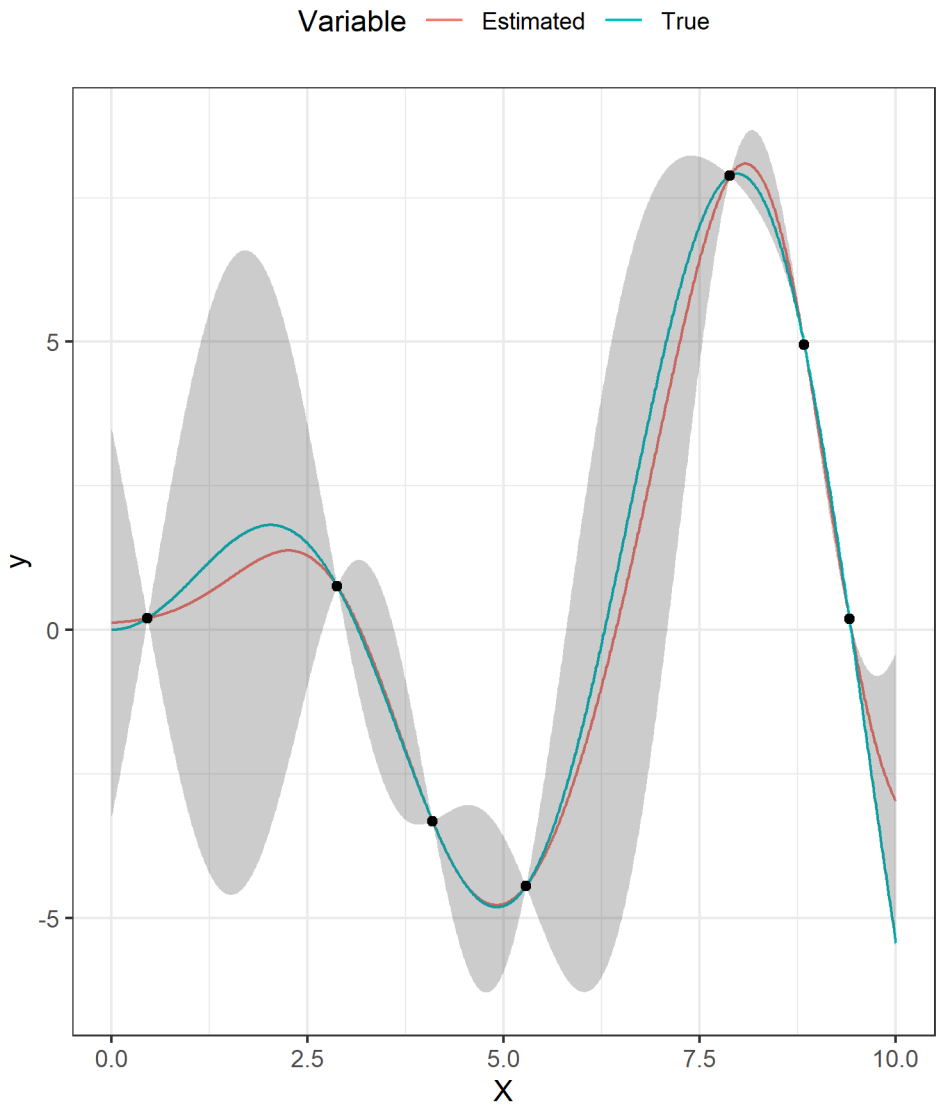
Want to construct an estimate $\hat{f}(x)$ such that

- $\hat{f}(x_i) = y_i$ for all observed $(x_i, y_i)$
- $E[\hat{f}(x)] = g(x)$ for all $x$
- Ideally, $\hat{f}(x)$ approximately follows a known distribution (e.g., $\hat{f}(x) \sim N(g(x), \sigma^2(x))$)

Interpolation and similar methods are popular: Splines, Gaussian processes, sometimes other ML methods

- Observed points are fit exactly, points between are estimated with a smooth curve

Useful as a proxy in simulations/computer codes when exact relationship between $X$ and $y$ is important



# UQ Applications (3)

**Approach 3: Given a value of inputs $x$, describe the distribution of the non-deterministic outcome $Y$**

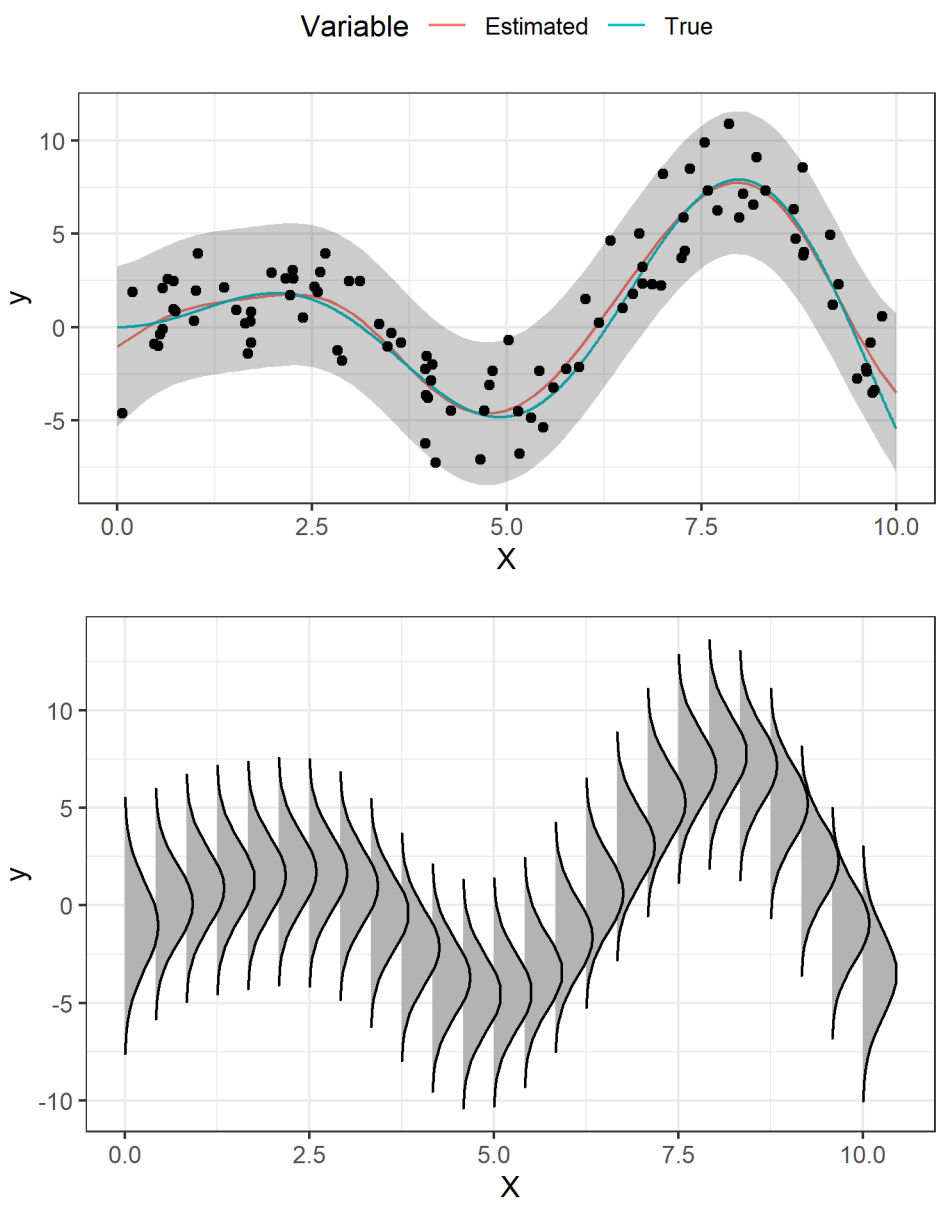
There is some function $H$ such that the conditional distribution of outcomes is described by $H(x)$

- $H$ describes uncertainty in $Y|x$ and uncertainty in the relationship between $y$ and $x$

Want to find $F$ such that $F(x) \equiv H(x)$ for all $x$

In practice, assumptions about distribution family and continuity/smoothness are made

Useful for when outcomes are not deterministic and more information is needed beyond mean, variance, etc.



# Research Objective and General Approach

Goal: Given a distribution of attacks, estimate the probability that differential pressure on the pressurizer will exceed 1 MPa

- It would also be useful to identify attack scenarios in which risk is high and estimate distributions of outcomes under particular attack scenarios

Approach 3 (estimate distribution of $Y$ given any value for $x$) provides insights into both overall risk **and** the relationship between attack characteristics and impacts

Data sneak peek:

- Inputs to the emulation are components targeted (heater and/or sprayer) and values injected to targeted components
- Output of emulation is binary indicator of whether differential pressure exceeded 1MPa at any point during the attack

It is not apparent *ex ante* whether the relationship between inputs and outcomes are simple/linear/monotonic/etc.

- Approach needs to be flexible (e.g., smoothing) **and** needs to provide conditional distributions of outcomes

# Kernel Smoothing and Gaussian Process Approaches

# A Failed Approach and Lessons Learned

**Kernel smoothing** was initially selected to analyze data

- Fits a smooth curve through noisy data; researcher specifies/selects "kernel" which controls level of smoothing
- Predictions of conditional outcomes are asymptotically normal, providing conditional distribution of outcomes
  - Result relies on assumptions similar to Central Limit Theorem (i.e., defined mean, finite variance)
- **Constrained kernel smoothing** can implement inequality constraints to accommodate binary outcomes, ensure predictions are in $[0, 1]$

However, the following issues arose:

- Asymptotic distributions for constrained kernel smoothing often not straightforward, involve additional distributional assumptions [4]
- Other real-valued outcomes from this emulation did not meet assumptions necessary for asymptotic distributions to be valid
  - E.g., log-time-to-threshold outcomes often showed evidence of infinite variance

Needed an alternative approach that didn't rely on asymptotics and could more reliably produce conditional outcome distributions for binary outcomes

# Gaussian Processes

Gaussian processes (GPs) are functionally similar to kernel smoothing

- Similar objective: Fit a smooth curve through noisy observations
- Major difference: Gaussian processes are likelihood-based, whereas kernel smoothers are not
  - Enables formal inference for smooth curve estimates without relying on asymptotics at the cost of needing to specify a generating distribution

Standard GP form:

$$f(x) \sim N\left(0, K_\theta(x)\right)$$
$$y \sim N(f(x), \sigma^2 I)$$

- $x$: Inputs, parameters to emulation/disruption
- $y$: Outcomes of emulation, conditional on $x$
- $f$: Emulation function mapping inputs to outcomes
- $K$: Kernel function describing degree and form of smoothing
- $\theta, \sigma^2$: Parameters of the GP to be fit
- $I$: Identity matrix with size equal to the number of observations used to fit the GP

GP details

# Gaussian Processes vs. Kernel Smoothers

Standard GP form:

$$f(x) \sim N\big(0, K_\theta(x)\big) \quad (1)$$

$$y \sim N(f(x), \sigma^2 I) \quad (2)$$

A few comments and comparisons with kernel smoothers:

- Eq. (1) is a distribution over functions. In estimation this describes observed data, but extends to all $x$ in the range of $X$. So, sampling methods will generate draws of entire functions.
- $K_\theta(x)$ is the "kernel" and plays a similar role as in kernel smoothers: Control how much smoothing occurs and fine-tune shape of estimated curves.
- Many parameters are sampled/evaluated in (1) (# of observations!), so the convolution $y \sim N(0, K + \sigma^2 I)$ is frequently used.
  - Lose the ability to directly evaluate functions/predicted values, but the distribution of function values for observed and hypothetical data can be inferred
  - Convolution isn't always possible/straightforward if eq. (2) is not a normal sampling statement; e.g., binary outcomes

GP details

# Gaussian Process for Binary Outcomes

The standard GP formulation assumes outcomes are real-valued, normally-distributed

- Not appropriate for modeling binary outcomes

GP can be extended to binary outcomes through use of a **link function** (denoted $\phi$)

$$f(x) \sim N\left(0, K_\theta(x)\right)$$

$$y \sim \text{Bernoulli}\left(\phi\left(f(x)\right)\right)$$

$$\phi: \mathbb{R} \to [0,1]$$

A popular choice for $\phi$ is the inverse logit function:

$$\phi(z) = \frac{1}{1+\exp(-z)}$$

To review, the binary outcome GP formulation:

- Fits a smooth curve through noisy data very similarly to kernel smoothing
- Does not rely on potentially invalid normality assumptions since focus is on a binary outcome
- Enables straightforward inference that leverages likelihoods explicitly specified by the model, even for constrained outcomes

# Sampling Procedure

Inputs to emulation: Which actuator is targeted, what value(s) are injected to targeted actuator(s)

Output of emulation: Indicator of whether differential pressure exceeded 1 MPa at any point during the attack

- Recall: Severe consequences can occur if differential pressure exceeds 1 MPa

We focus on estimating probability that differential pressure will exceed 1 MPa at any point in the two minute attack, conditional on which actuator is targeted and values injected to targeted actuator(s)

Input samples used to fit the GP were generated with Latin hypercube sampling

- Generates a sample of inputs that effectively spans the input space
- Sample contained 250 attack scenarios: 83 targeted heater, 83 targeted sprayer, and 84 targeted both

Emulation was run with each input sample and outcomes were recorded

- Emulation initialized then allowed to stabilize for 5 minutes, followed by a 2 minute attack
- Obtaining additional observations is relatively time-intensive, motivating

# Estimation Issues and Procedure

Potential estimation issue: Binary outcome models can suffer from parameter instability, especially when using more flexible relationships between inputs and outcomes [5]

- Can cause convergence issues for classical statistics estimation methods
- Bayesian statistics estimation methods offer a solution
  - Must specify priors over parameters, but even weakly-informative priors can improve stability
  - Convergence can be assessed more reliably (e.g., trace plots/diagnostics)

Following Gelman et al. [6], half-Cauchy priors were assumed for kernel parameters

- Relatively uninformative while maintaining a central tendency

$$f(x) \sim N\left(0, K_\theta(x)\right)$$
$$y \sim \text{Bernoulli}\left(\phi\left(f(x)\right)\right)$$
$$\phi(z) = \frac{1}{1+\exp(-z)}$$

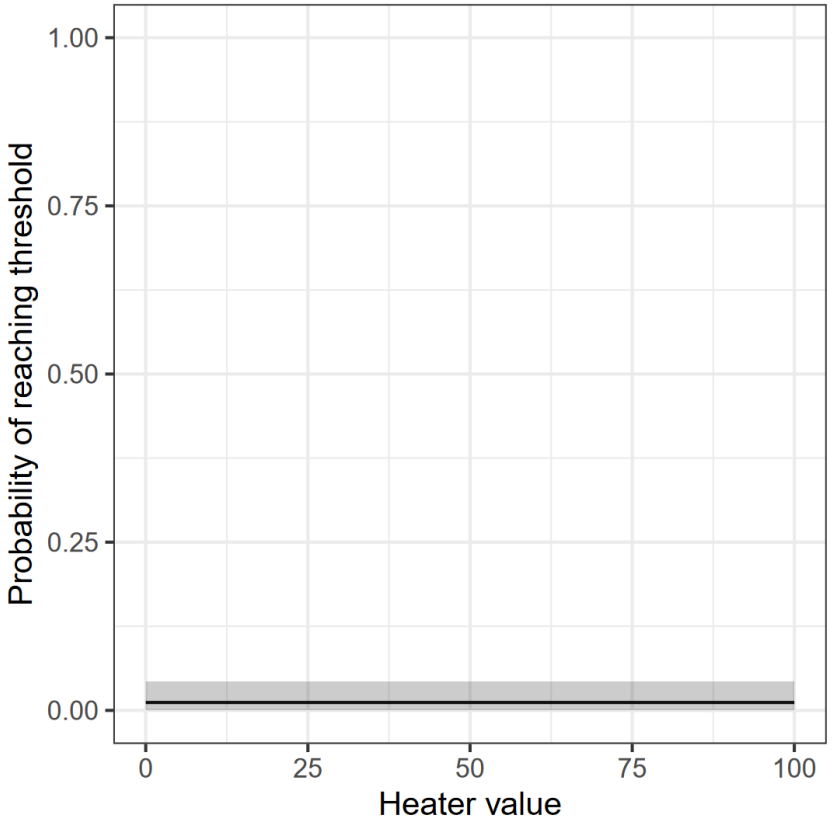
$$K_\theta(X)_{ij} = \alpha^2 \exp\left(-\frac{1}{2}\left(X_i - X_j\right)' P^{-1}\left(X_i - X_j\right)\right)$$
$$\alpha \sim \text{Cauchy}^+(0, 10)$$
$$P_{ii} \sim \text{Cauchy}^+(0, 10)$$
$$P_{ij} = 0 \quad \text{for } i \neq j$$

Model was estimated using No U-Turn Sampling as implemented in the Stan platform [7]

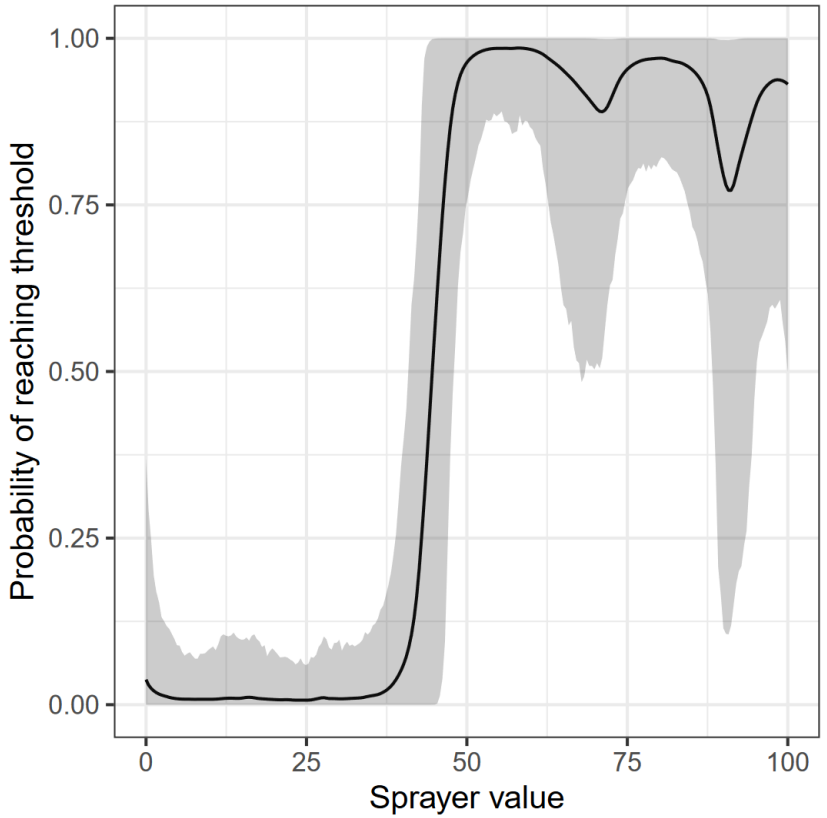
- Sampler used 1000 burn-in and 1000 sampling iterations
- Trace plots/diagnostics were used to assess convergence of Markov chain

# Results

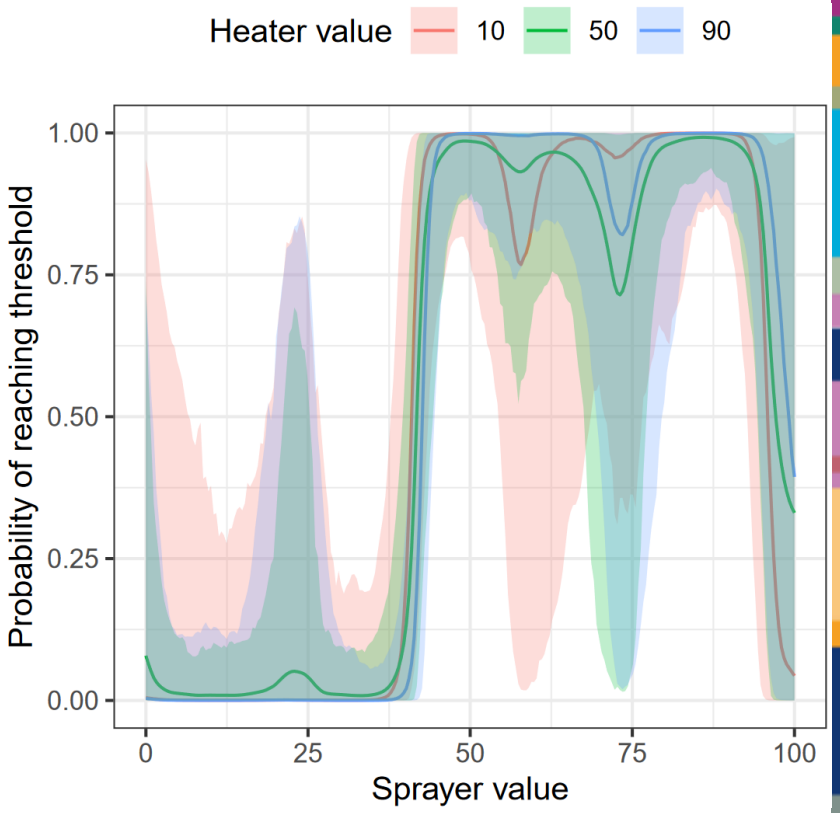
# Conditional Probability Estimates




Probability of reaching 1 MPa threshold when only the heater actuator is targeted, conditional on values injected to heater actuator




Probability of reaching 1 MPa threshold when only the sprayer actuator is targeted, conditional on values injected to sprayer actuator




Probability of reaching 1 MPa threshold when both actuators are targeted, conditional on values injected to both actuators

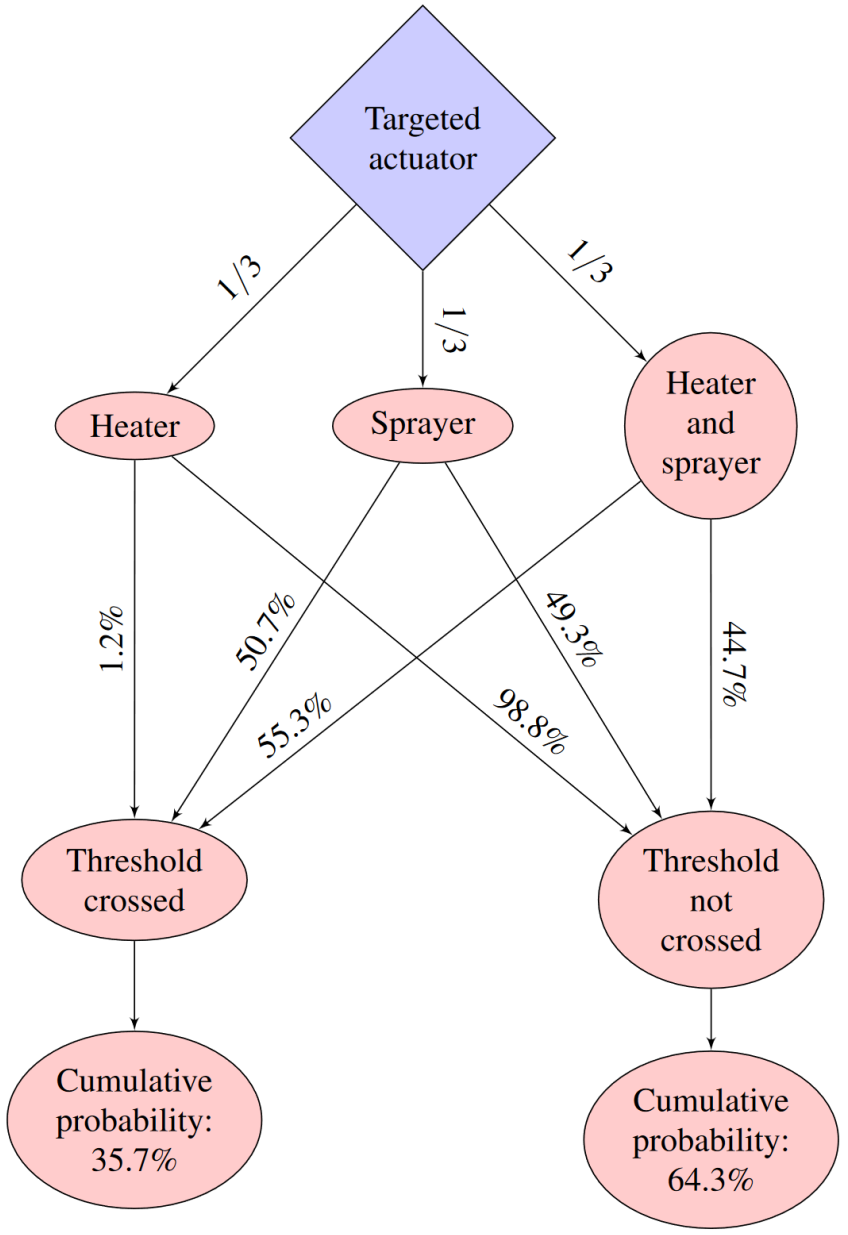
# Overall Probability Estimates

Fitted GP surrogate function then used to estimate overall probability of crossing 1 MPa threshold

- Heater actuator, sprayer actuator, and both actuators targeted with 1/3 probability each
- Values injected to targeted actuator(s) uniformly distributed on [0, 100]

GP surrogate function provides computationally feasible method of approximating these probabilities

- Each iteration of the emulation takes 5 minutes to set up/stabilize + 2 minutes for attack
- Each sample of the GP takes ~0.7 seconds (3.5 GHz processor)



# Discussion and Impacts

# Methodological Extensions

Investigation of analytical methods other than GPs

- Constrained kernel smoothing + bootstrapping for binary outcomes

Alternative forms of GP model

- **Problem**: Results show some evidence of overfitting
  - Indicates that estimate uncertainty may be under-quantified, increasing bias in estimates
- **Potential solutions**:
  - Adaptive sampling to focus on regions where overfitting appears to have occurred
  - Imposition of constraints such as monotonicity where justified
  - Alternative priors to "force" greater prediction uncertainty (last resort)

Alternative sampling methodologies to boost overall UQ/GP efficiency

Extension to real-valued outcomes

- **Problem**: Fat-tailed distributions for some real-valued outcomes complicate asymptotics
- Requires more explicit modeling of outcome distributions, which can be more difficult to work with computationally/numerically

# Impacts

Lessons learned and informing UQ cyber emulation analyses for other applications

- Assumptions supporting asymptotics may not be valid for real-valued outcomes. Methods that avoid asymptotics are preferable (possibly necessary). Candidates include Gaussian processes and bootstrapping.
- Reframing outcomes can be useful to avoid difficult distributional assumptions and simplify analysis
  - E.g., can real-valued outcomes be transformed into binary/categorical indicators and still capture important information?

This proof-of-concept model was extended to be more representative of actual PWR systems, directly useful to DOE-NE sponsor

- Incorporate intrusion detection/mitigation into SCADA network
- Utilize previous threat/vulnerability analyses and intelligence to develop more sophisticated/realistic distribution of attacks

# Acknowledgements

Thanks to Eric Vugrin, Robert Bruneau, Meg Galiardi, Amanda Gonzales, Jamie Thorpe, Ray Fasano, Tim Ortiz, and Ryan Kao

Work made possible by funding from DOE-NE Cyber Security for Nuclear Facilities

# References

[1] McKenzie, T., Tarman, T., and Lamb, C., 2020. "Uncertainty quantification for cyber-physical PWR experiments". *Proceedings of the 28th Annual International Conference on Nuclear Engineering*.

[2] Sandia National Laboratories, 2019. SCEPTRE: SCADA as a Platform.

[3] Nuclear Energy Institute, 2022. U.S. Nuclear Operating Plant Basic Information.

[4] Cai, T., Low, M., and Xia, Y., 2013. "Adaptive confidence intervals for regression functions under shape constraints". *The Annals of Statistics*, ***41***(2), pp. 722-750.

[5] Allison, P., 2009. "Fixed effects regression models". SAGE Publications.

[6] Gelman, A., et al., 2006. "Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper)". *Bayesian analysis*, ***1***(3), pp. 515-534.

[7] Stan Development Team, 2018. "Rstan: The R interface to Stan". R package version 2.18.2.

# Gaussian Process Details

Main model:

$$f(x) \sim N\big(0, K_\theta(x)\big) \qquad (1)$$
$$y \sim N(f(x), \sigma^2 I) \qquad (2)$$

$$K_\theta(X)_{ij} = \alpha^2 \exp\left(-\frac{1}{2}(X_i - X_j)' P^{-1}(X_i - X_j)\right)$$

Dimension reduction using convolution:

$$(1) + (2) \Rightarrow$$
$$y \sim N(0, K_\theta(x) + \sigma^2 I)$$

Inference:

$$y^* | x^*, y, x, \theta, \sigma \sim N(A, B)$$

$$A = K_\theta(X^*, X)\Sigma^{-1} y$$
$$B = K_\theta(X^*) - K_\theta(X^*, X)\Sigma^{-1} K_\theta(X^*, X)'$$
$$\Sigma = K_\theta(X) + \sigma^2 I$$

$$K_\theta(X^*)_{ij} = \alpha^2 \exp\left(-\frac{1}{2}(X_i^* - X_j^*)' P^{-1}(X_i^* - X_j^*)\right)$$

$$K_\theta(X^*, X)_{ij} = \alpha^2 \exp\left(-\frac{1}{2}(X_i^* - X_j)' P^{-1}(X_i^* - X_j)\right)$$

Return to main presentation