Artificial Intelligence for Aerospace Engineering: Bridging the Gap Between Data-Driven and Physics-Based Modeling

James Warner

NASA Langley Research Center

Hampton, Virginia

Ann and H.J. Smead Aerospace Engineering Sciences Seminar Series

University of Colorado Boulder

December 1st, 2023

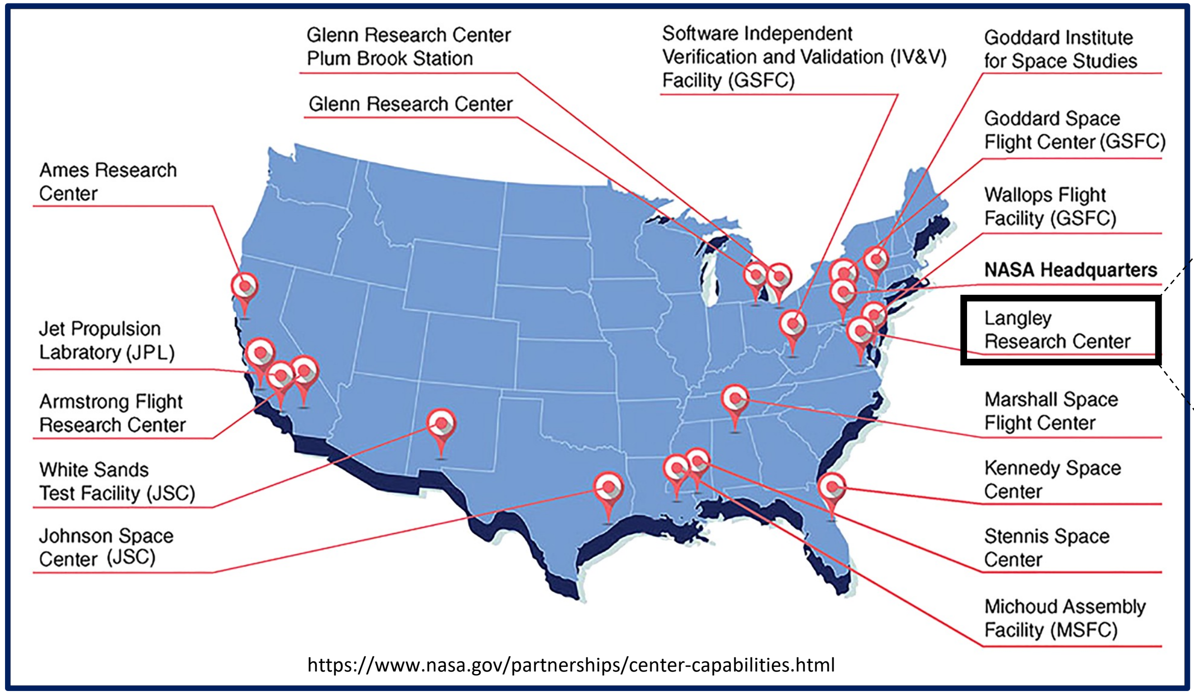
*Team Members: Geoffrey Bomarito, Patrick Leser, Paul Leser

About Me

- **Hometown:** Warwick, NY
- **Binghamton University** *B.S. Mechanical Engineering*
- **N.I.S.T** *Undergraduate Researcher*
- **Cornell University** MS/*PhD Civil Engineering*
- **Duke University** *Research Associate*
- **NASA** *Senior Computational Scientist*
    - Langley Research Center, Hampton, VA
    - Durability, Damage Tolerance, and Reliability Branch
    - 2013 - Present

NASA Langley Research Center

NASA Facilities and Centers Across the US



Langley Research Center, Hampton, VA

- The oldest of the NASA Centers (est. 1917!) specializing in a range of aeronautics and space research:
  - On-Demand Mobility Technologies
  - Commercial Air Transport Technologies
  - Earth's Atmospheric Composition
  - Entry, Descent, and Landing Systems
  - Advanced Space Structures
  - Autonomous In-Space Assembly and Manufacturing

Research Group Overview

Skills

Applied Mathematics

Software Development

Probability

Statistics

Applied Physics

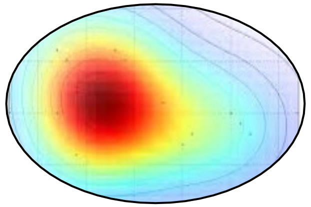
Mechanical Testing

Aerospace Engineering

Numerical Methods

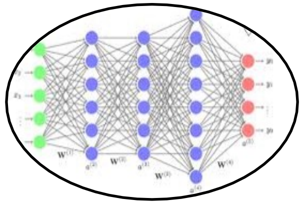
Solid Mechanics

Research Areas



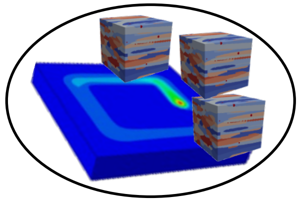
Uncertainty quantification (UQ)

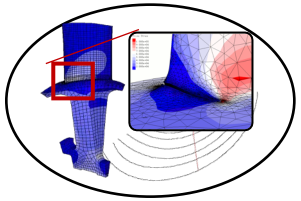
Machine Learning (ML)



High Performance Computing

Damage Science

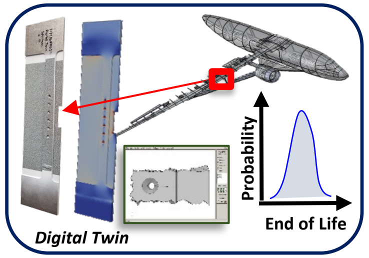




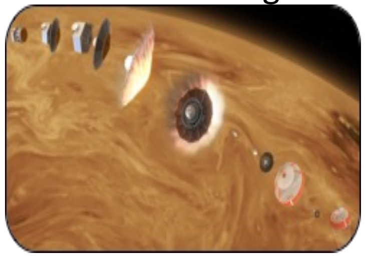
Fatigue and Fracture

Applications

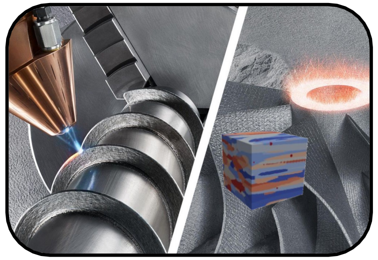
Structural Health Management



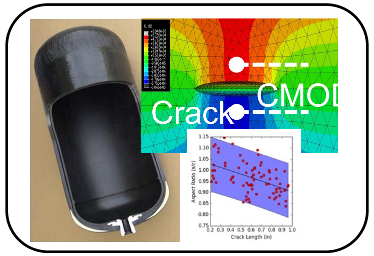
Entry, Descent, and Landing



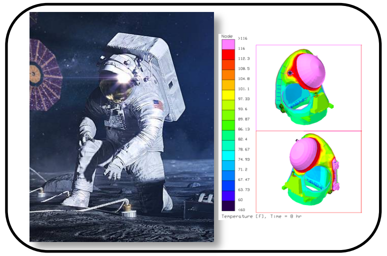
Additive Manufacturing



Composite Overwrapped Pressure Vessel Liner Fracture Mechanics

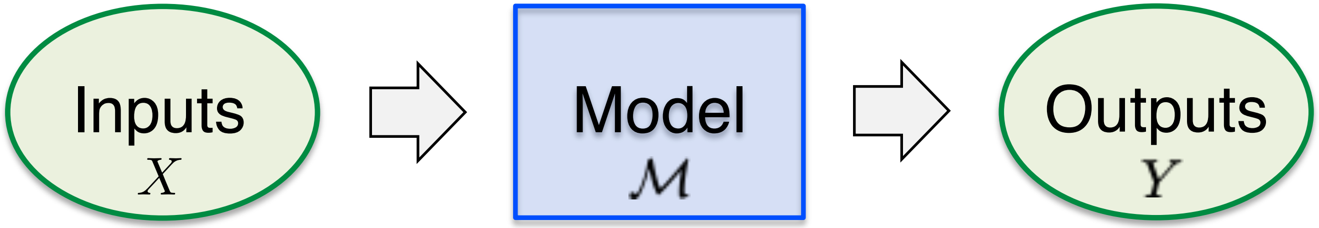


xEMU Spacesuit Certification



Frangible Joint Reliability

Modeling and Simulation for Aerospace Engineering

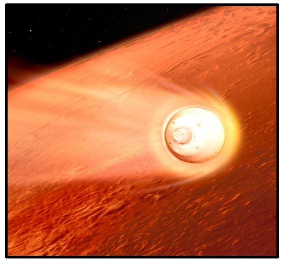


Modeling and Simulation for Aerospace Engineering

Inputs X → Model $\mathcal{M}$ → Outputs Y

Trajectory Simulation

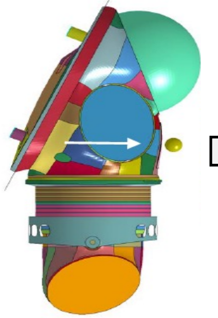
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

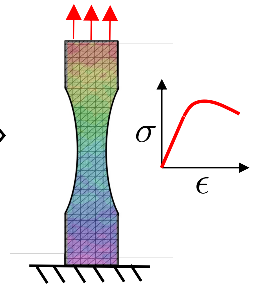
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

- Applied load
- Material properties



Deformation, Stress, Strain

Motivation: Outer Loop Applications

- Examples: optimization, inverse problems, design, …

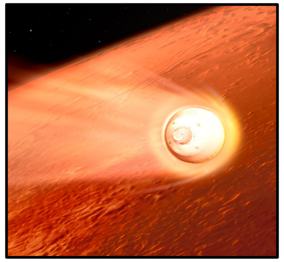
Repeated Evaluations

Inputs X ⇨ Model $\mathcal{M}$ ⇨ Outputs Y

Trajectory Simulation

- Atmospheric conditions
- Mass properties

⇨

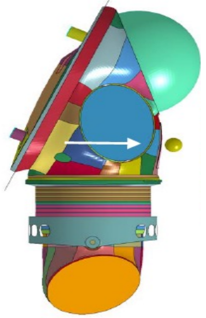


⇨ Landing location

Spacesuit Impact Model

- Impact velocity
- Lunar regolith properties

⇨

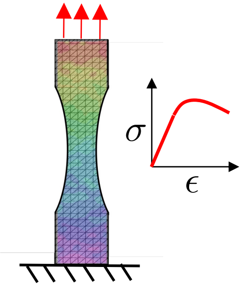


⇨ Max contact force

Mechanical Testing Simulation

- Applied load
- Material properties

⇨



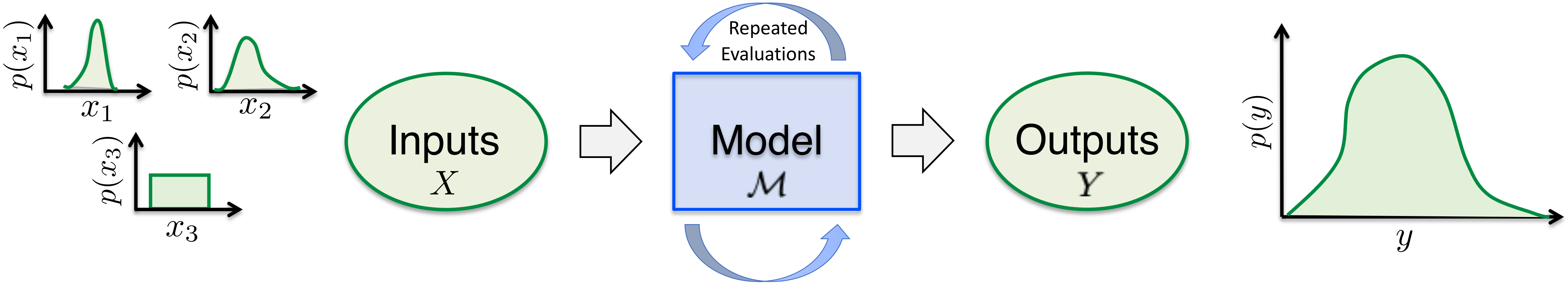
⇨ Deformation, Stress, Strain

➢ Trajectory optimization/guidance

➢ Structural design optimization

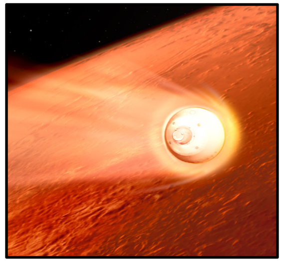
➢ Optimal experimental design

UQ



Trajectory Simulation

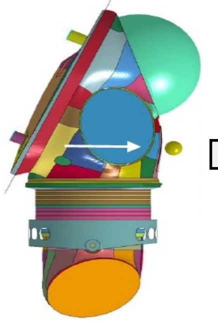
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

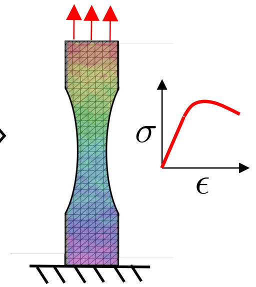
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

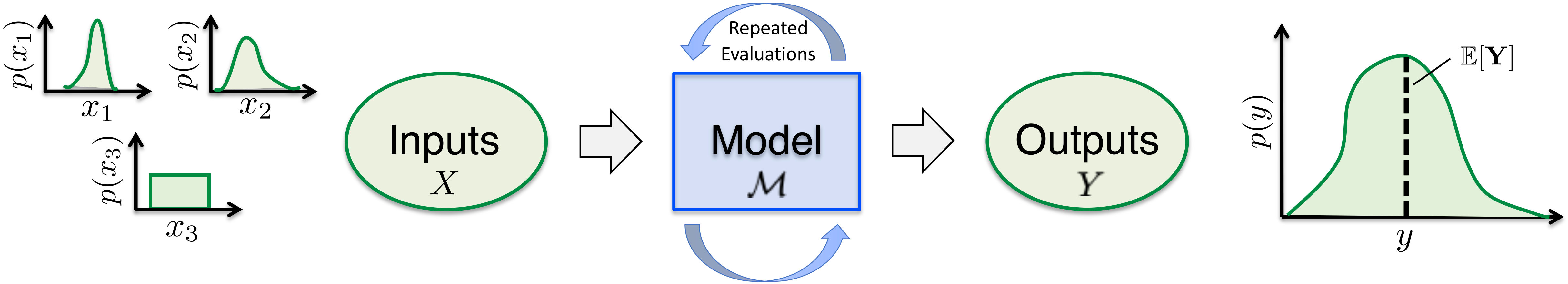
- Applied load
- Material properties



Deformation, Stress, Strain

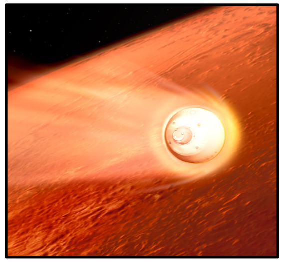
UQ

Statistical estimators



Trajectory Simulation

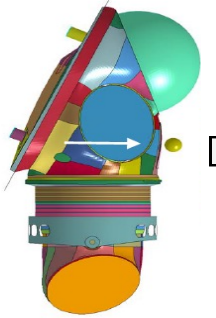
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

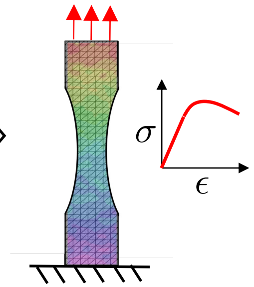
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

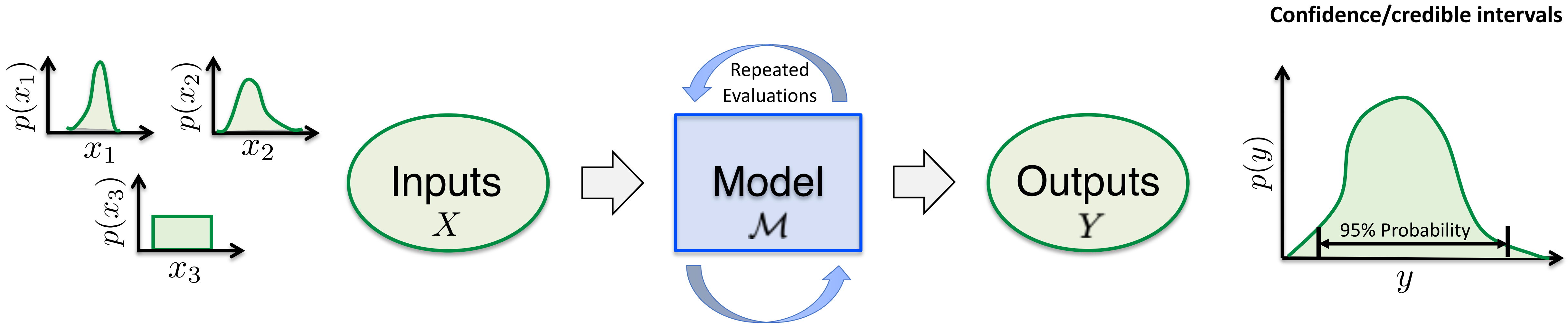
- Applied load
- Material properties



Deformation, Stress, Strain

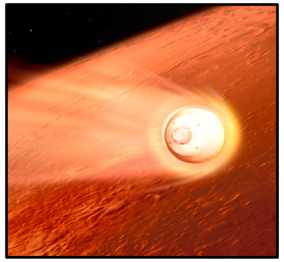
➢ **Expected landing location**

UQ



Trajectory Simulation

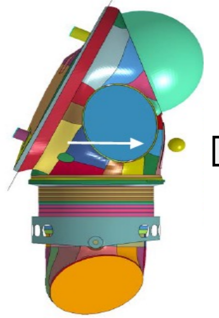
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

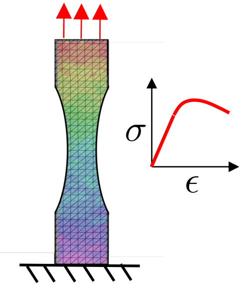
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

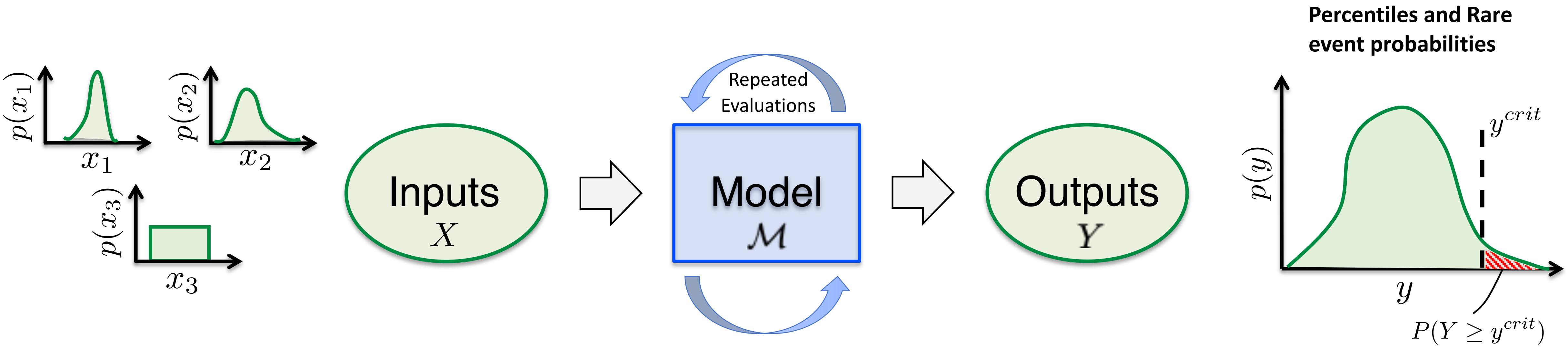
- Applied load
- Material properties



Deformation, Stress, Strain

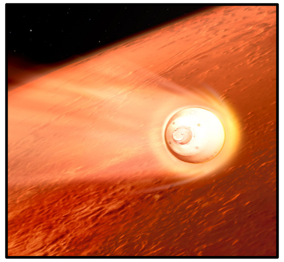
- ➢ Expected landing location
- ➢ **95% confidence landing ellipse**

UQ



Trajectory Simulation

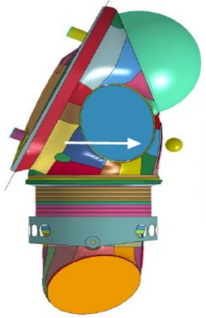
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

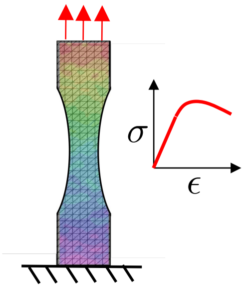
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

- Applied load
- Material properties

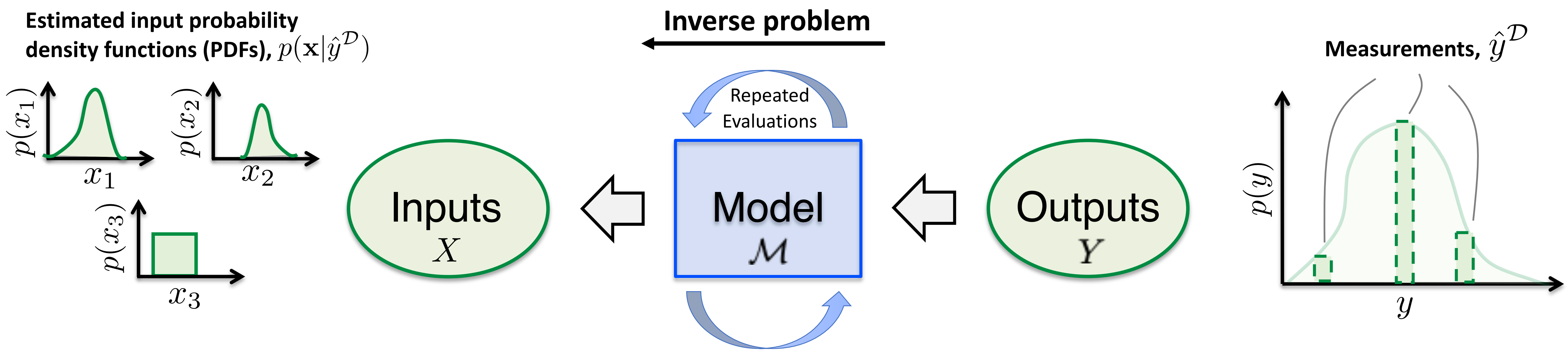


Deformation, Stress, Strain

- ➢ Expected landing location
- ➢ 95% confidence landing ellipse

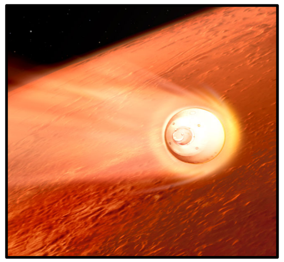
- ➢ **Probability of failure**

UQ



Trajectory Simulation

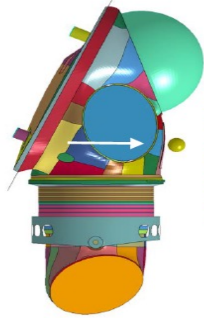
- Atmospheric conditions
- Mass properties



Landing location

Spacesuit Impact Model

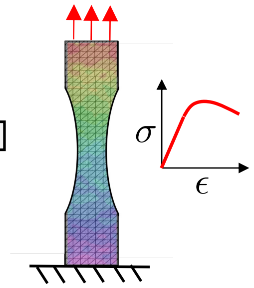
- Impact velocity
- Lunar regolith properties



Max contact force

Mechanical Testing Simulation

- Applied load
- Material properties



Deformation, Stress, Strain

➢ Expected landing location
➢ 95% confidence landing ellipse

➢ Probability of failure

➢ **Calibrated material parameter PDFs**

UQ

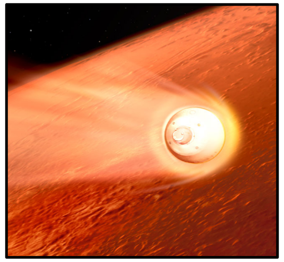
UQ with high-fidelity (Hi-Fi), physics-based models can be computationally intractable

Number of Model Evaluations O(10^3-10^6)

Inputs X → Model $\mathcal{M}$ → Outputs Y

Trajectory Simulation

- Atmospheric conditions
- Mass properties

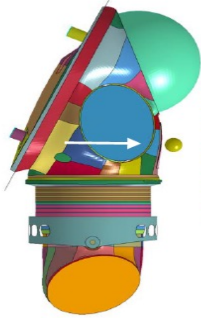


Landing location

Computation Time: O(mins-hours)

Spacesuit Impact Model

- Impact velocity
- Lunar regolith properties

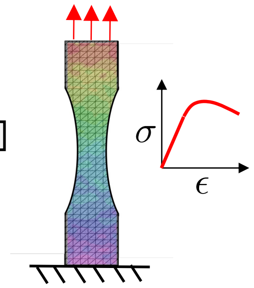


Max contact force

Computation Time: O(hours-days)

Mechanical Testing Simulation

- Applied load
- Material properties

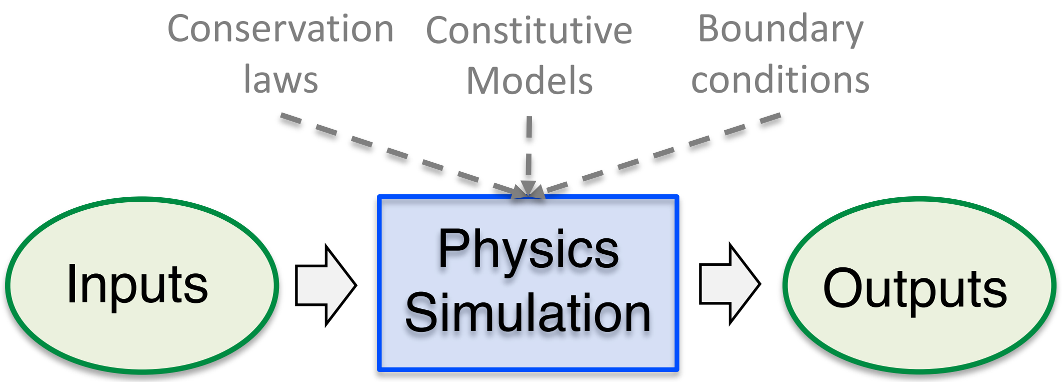


Deformation, Stress, Strain

Computation Time: O(sec-mins)

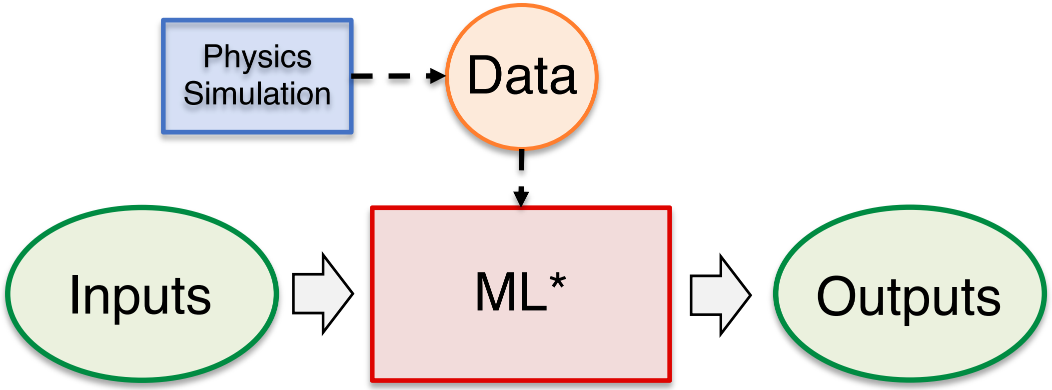
Two Modeling Paradigms

Physics-Based Modeling



Interpretable predictions
Theoretically rigorous
Historically-proven legacy codes
Computationally intensive
Slow development and deployment

Data-Driven Modeling



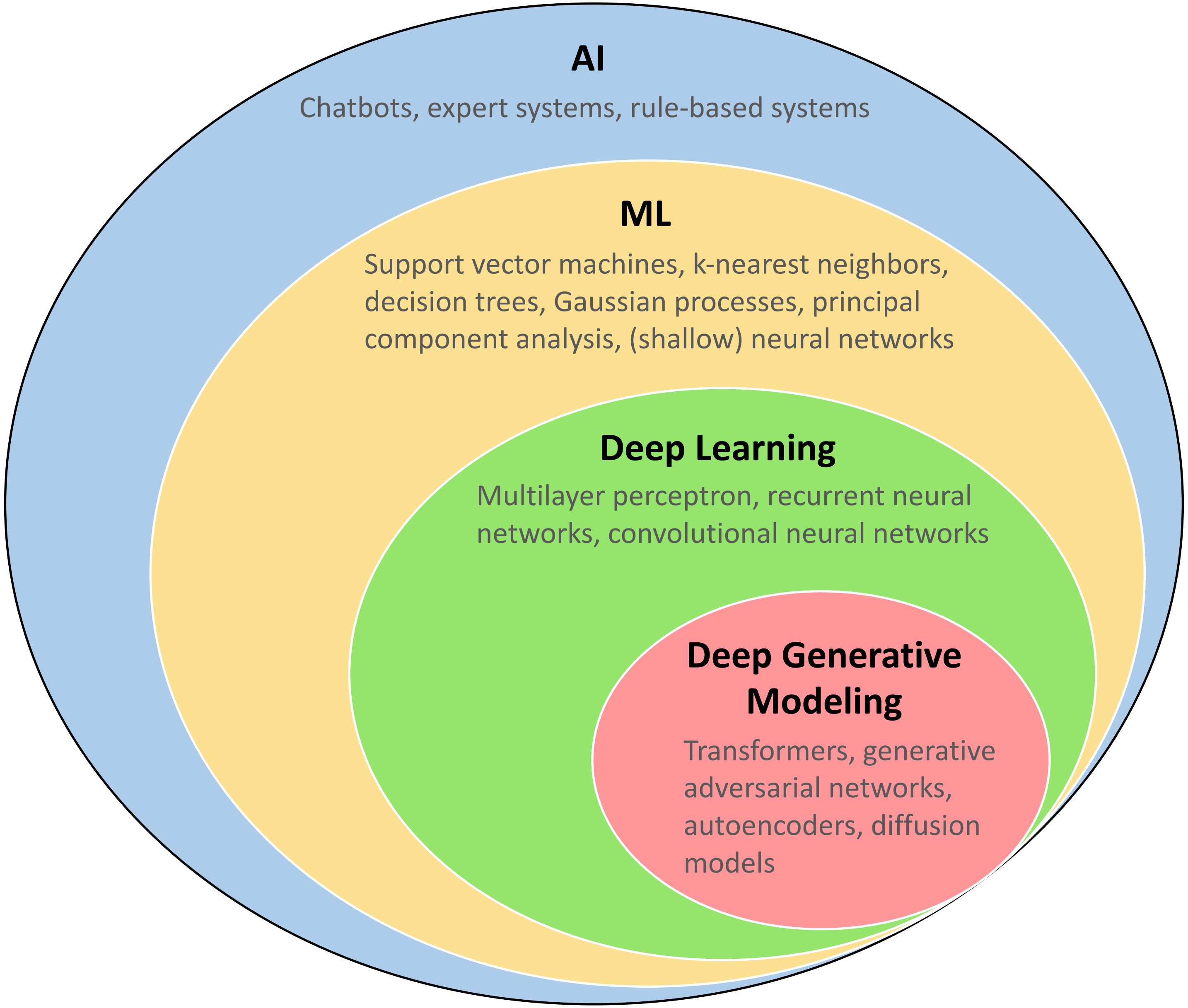
Efficient predictions
Fast deployment
Widely available libraries
Lack interpretability ("Black Box")
Ineffective for extrapolation/ sparse datasets

*Surrogate model, response surface, reduced-order model, kriging, metamodel, …….

Outline: Artificial Intelligence for Aerospace Engineering

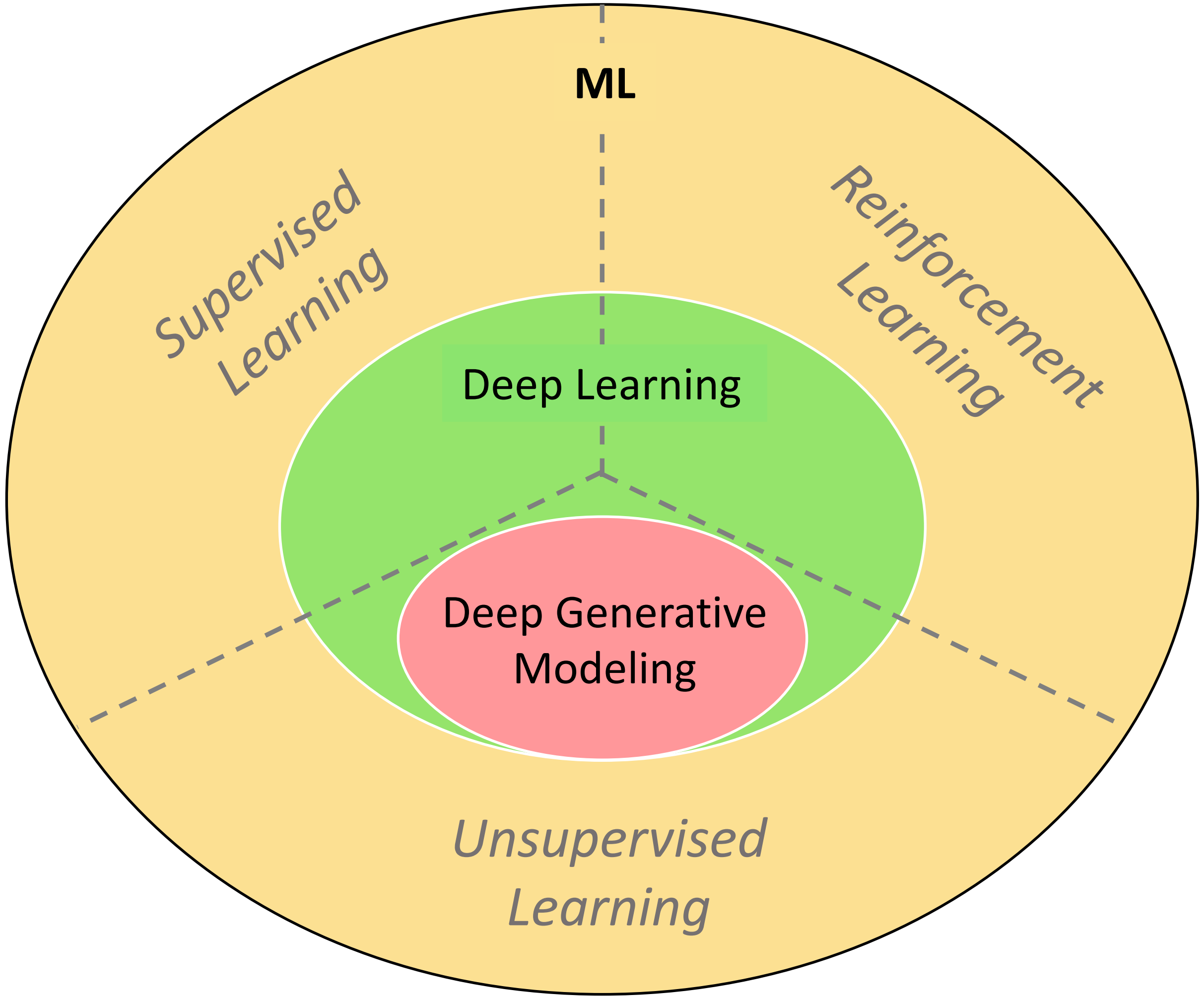
1. Artificial intelligence (AI) / ML primer
2. *Replacing* physics-based with data-driven model (surrogate modeling)
3. *Combining* physics-based and data-driven modeling
   - Physics-informed generative adversarial networks (PI-GANs)
     - Application: Material identification for certification by analysis
   - Multi-model Monte Carlo simulation
     - Application: Trajectory simulation for entry, descent, and landing (EDL)
   - Active learning with Gaussian Process models
     - Application: Spacesuit reliability analysis
4. Summary

AI and its Subsets



- Simulating intelligent behavior
- Learning from experience/data instead of being explicitly programmed
- Neural networks with multiple layers of processing to extract progressively high-level features from data
- Approximating high-dimensional probability distributions from data

ML: Three Categories

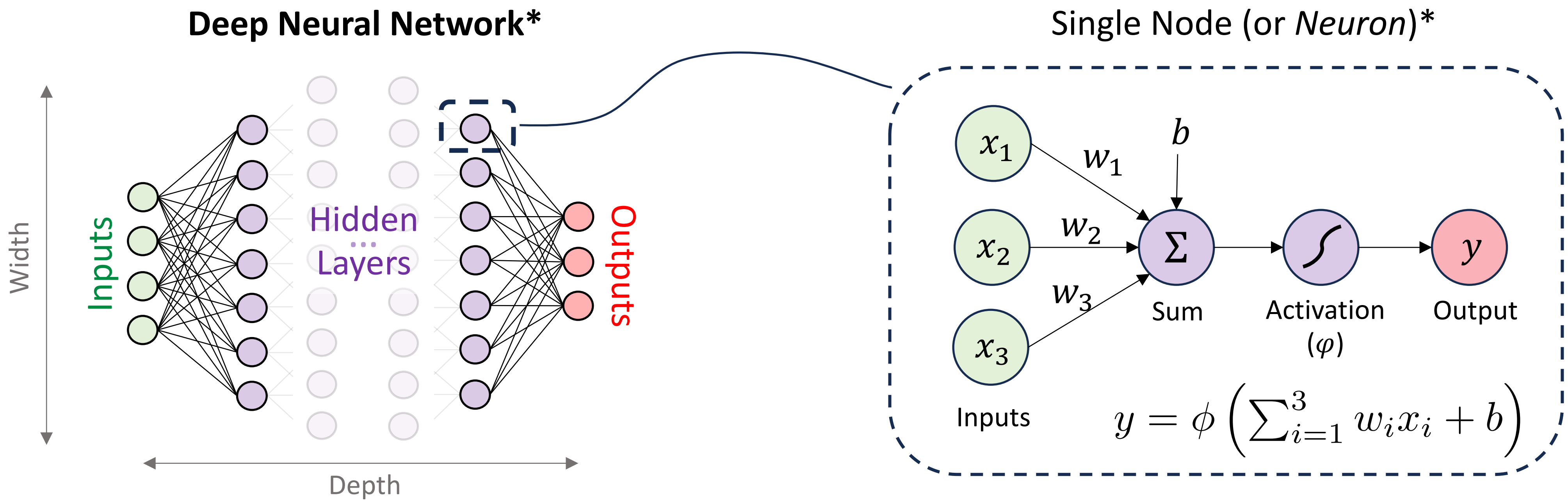


- *Supervised learning*:
  - Learns mapping between labeled data - inputs (features) and outputs (targets)
    1. Classification: discrete outputs
    2. Regression: continuous outputs

- *Unsupervised learning*:
  - Learns patterns from unlabeled dataset – inputs only
    - Example algorithms: Clustering, dimensionality reduction, density estimation

- *Reinforcement learning*:
  - Learns a task by interacting with environment through trial and error
    - Example applications: robotics, autonomous cars, gaming

Deep Learning Terminology



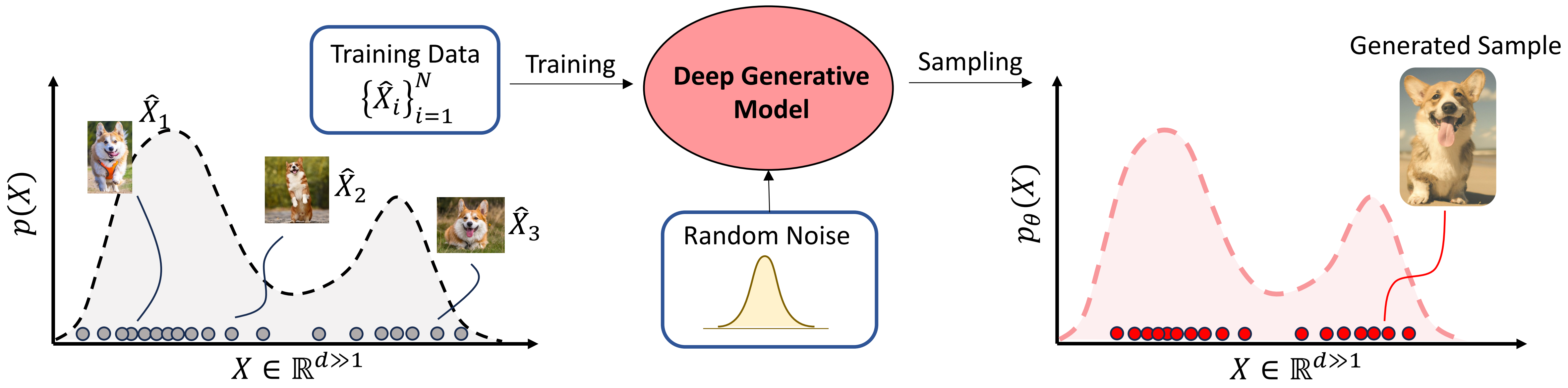
- Neural network parameters, $\theta = \{\vec{w}, \vec{b}\}$, are learned by optimizing a *loss function*, e.g., mean squared error (MSE) for supervised learning:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(y(\hat{X}_i;\theta) - \hat{y}_i\right)^2$$

$\{\hat{\vec{X}}, \hat{\vec{y}}\}$ are labeled training data

*For *full-connected* or *multilayer perceptron* architecture

Deep Generative Modeling Overview



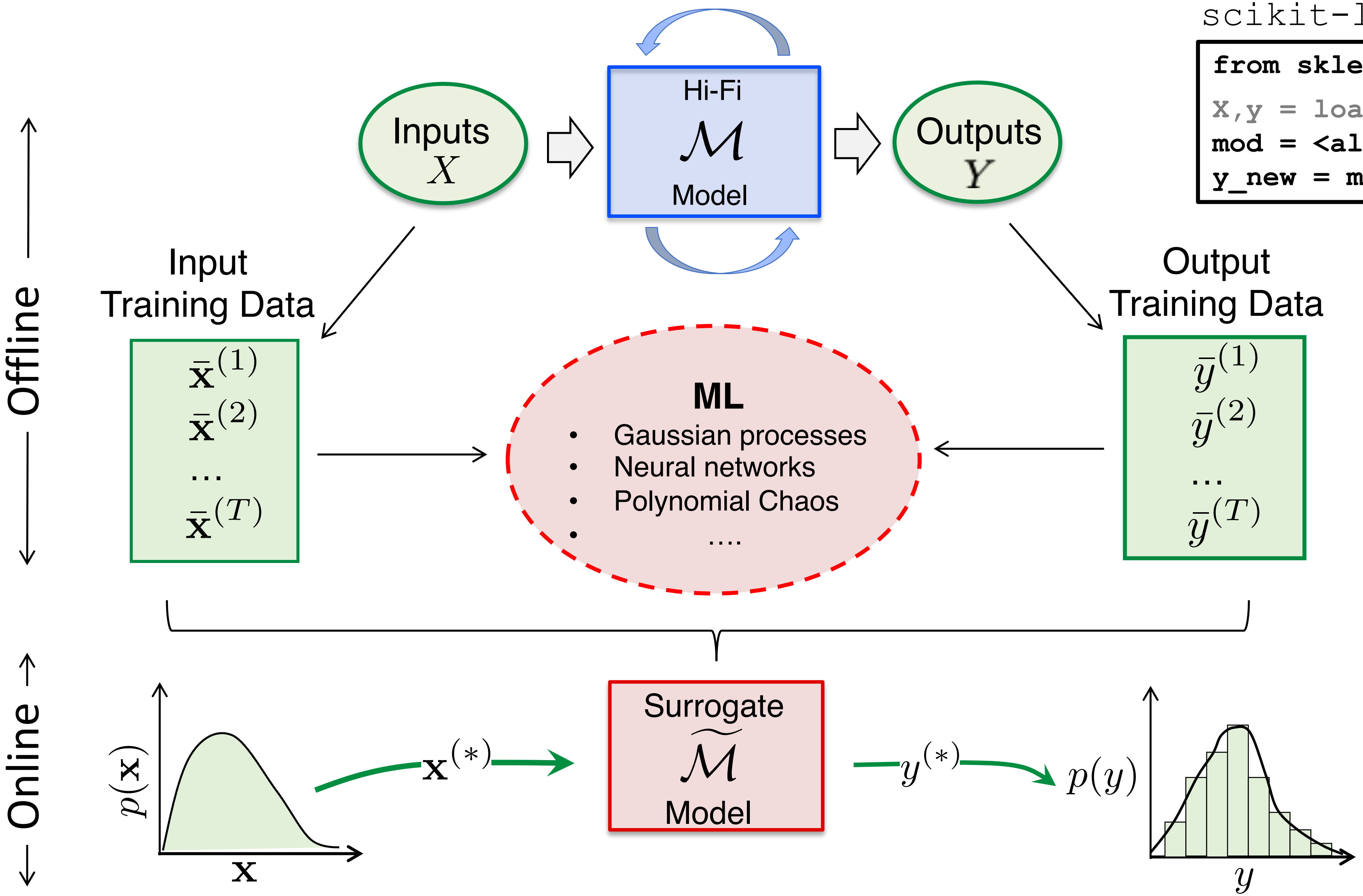
Goal: $p_\theta(X) \approx p(X)$

Training Data	Generative Model Architectures	Examples
Images	Generative adversarial networks (GANs), diffusion models, variational autoencoders (VAEs)	DALL-E, Midjourney
Text, Audio, Video	Transformers, long short term memory (LSTM) networks, Recurrent neural networks (RNNs)	ChatGPT, Bard, Jukebox

Surrogate Modeling

`scikit-learn` with Python:

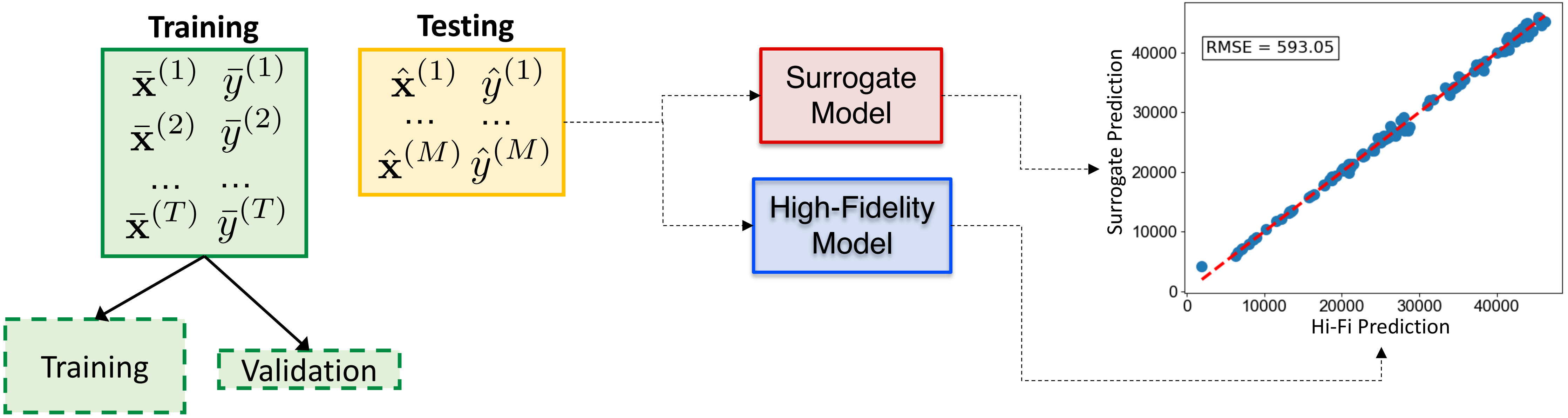
```
from sklearn import <alg>
X,y = load_data()
mod = <alg>.fit(X, y)
y_new = mod.predict(X_new)
```



Orders of magnitude computational speedup possible

Surrogate Model Validation

- **Always** create a **separate** dataset for testing that is **not** used for training
  - Guards against the problem of ***overfitting***
- A third *validation* dataset is often used during training to tune machine learning *hyperparameters* (e.g., # layers, nodes in neural network)
- If surrogate model error is non-negligible, it can be factored into total uncertainty when making probabilistic predictions

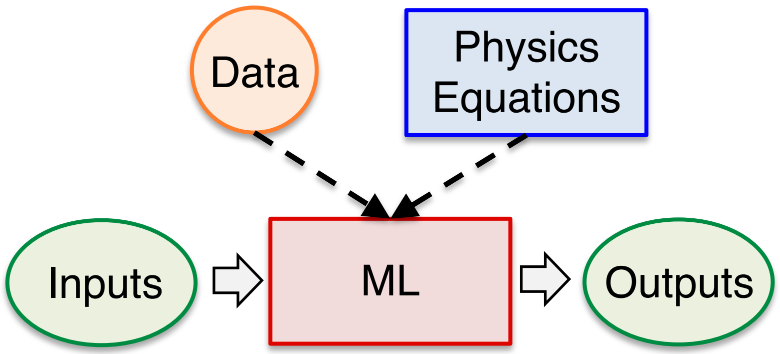


Surrogate Modeling Pitfalls and Challenges

1) Not validating the effectiveness of surrogate prior to deployment
    - ***Or validating the model using the training data!***
2) Not accounting for (or correcting) surrogate errors when they are non-negligible
    - Surrogate models yield **biased** estimators
3) Using surrogate model to extrapolate outside the range of training data
4) Training surrogate models from sparse training data
    - Common when high-fidelity model is expensive to run

Combining Physics and ML

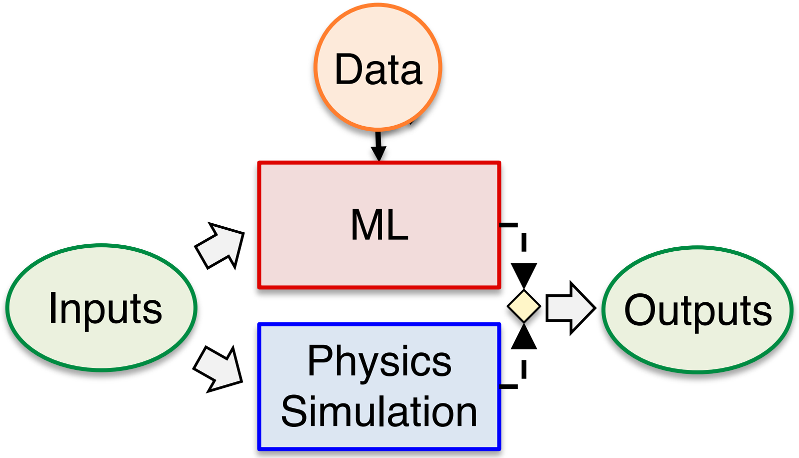
Physics-informed deep learning



➢ ***Integrate*** physical laws into the training of ML models

1. Physics-informed neural networks. Raissi et al. 2019.
2. Physics-informed generative adversarial networks Yang et. al. 2018

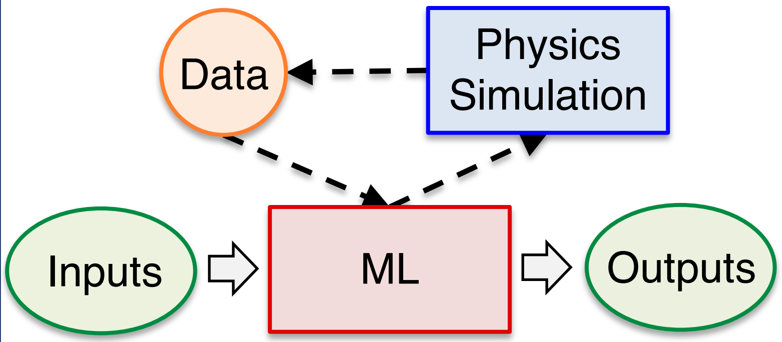
Multi-model Monte Carlo



➢ ***Fuse*** predictions from ML and physics-based models

3. Multilevel Monte Carlo. Giles et al. 2015
4. Multifidelity Monte Carlo. Peherstorfer et al. 2016
5. Approximate Control Variates. Gorodetsky et al. 2019

Active Learning



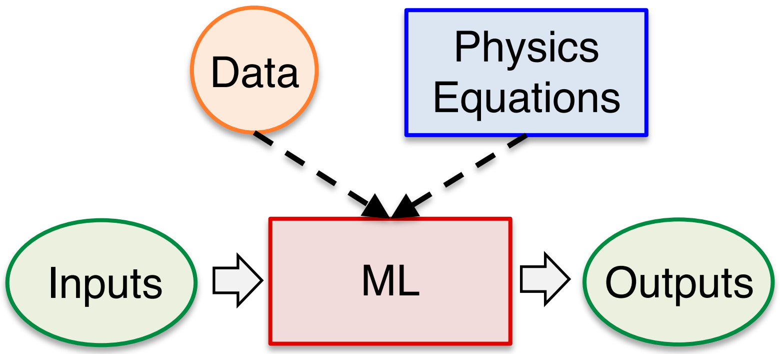
➢ ***Guide*** physics-based training data generation using ML Model

6. Active learning in practice. Settles 2011.
7. Active learning for reliability analysis. Bichon et al. 2008.

- **Application**: Material identification for certification by analysis
- **Application*:*** Trajectory simulation for EDL
- **Application:** Spacesuit reliability analysis

Combining Physics and ML

Physics-informed deep learning

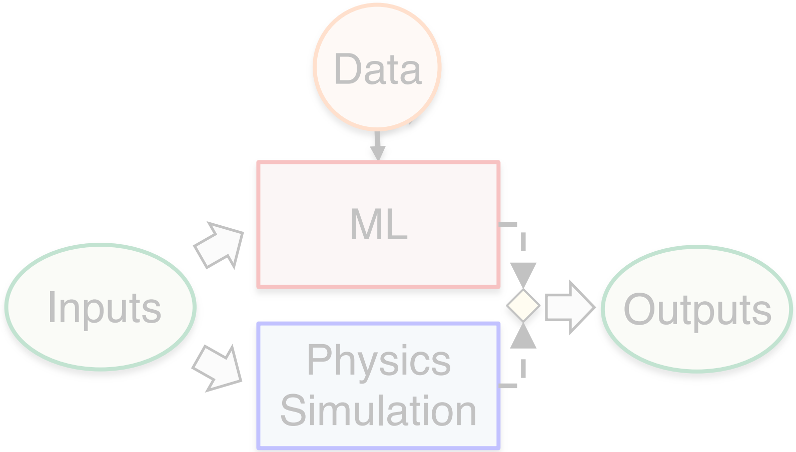


➢ ***Integrate*** physical laws into the training of ML models

1. Physics-informed neural networks. Raissi et al. 2019.
2. Physics-informed generative adversarial networks Yang et. al. 2018

- **Application**: Material identification for certification by analysis

Multi-model Monte Carlo

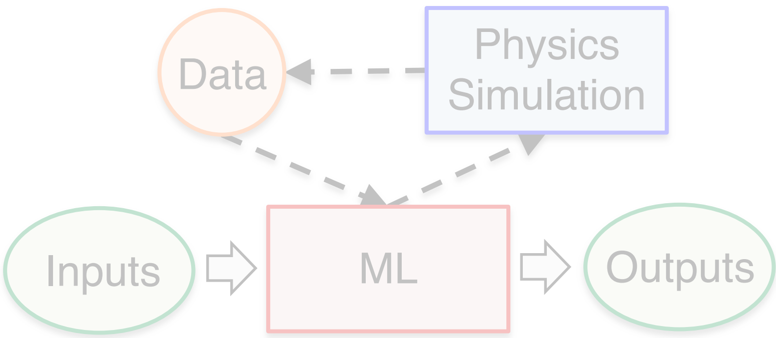


➢ ***Fuse*** predictions from ML and physics-based models

3. Multilevel Monte Carlo. Giles et al. 2015
4. Multifidelity Monte Carlo. Peherstorfer et al. 2016
5. Approximate Control Variates. Gorodetsky et al. 2019

- **Application:** Trajectory simulation for EDL

Active Learning



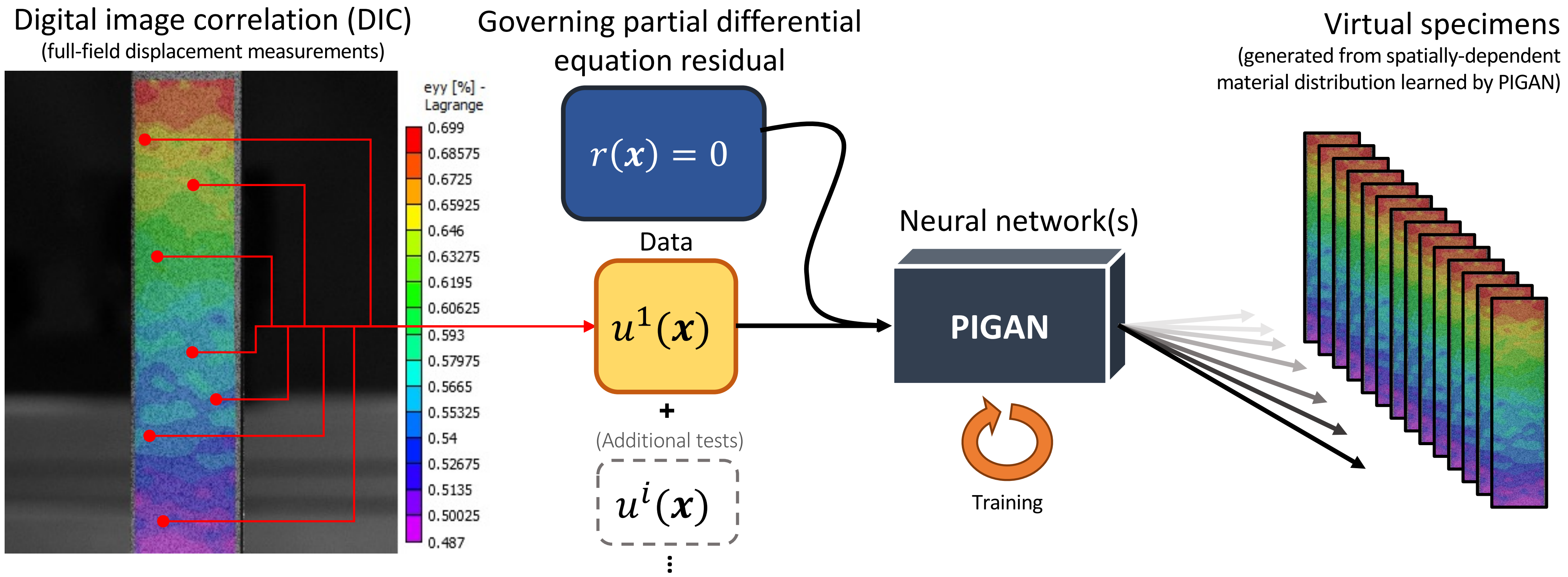
➢ ***Guide*** physics-based training data generation using ML Model

6. Active learning in practice. Settles 2011.
7. Active learning for reliability analysis. Bichon et al. 2008.

- **Application:** Spacesuit reliability analysis

Physics-Informed ML for Material Identification

- **Motivation**: Material identification for certification by analysis
- **Challenge**: solving high-dimensional, probabilistic inverse problems
- **Approach**: Implement physics-informed generative adversarial network (PIGAN) capable of estimating random material property fields from limited test data



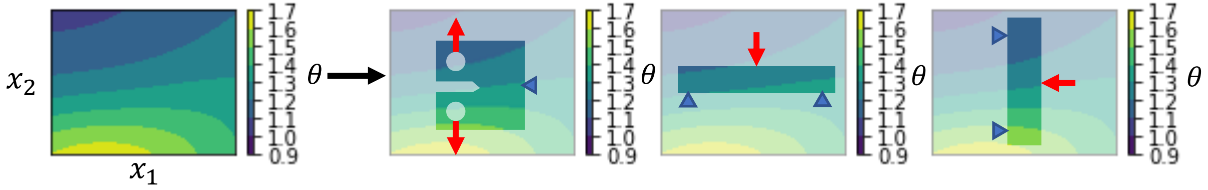
Collaborators: Will Jenkins (University of Utah), Jacob Hochhalter (University of Utah), Sean Current (Ohio State), Mohannad Elhamod (Virginia Tech), Marten Thompson (University of Minnesota)

Virtual Test Specimens

- Learned material fields can be mapped to different scenarios (boundary conditions, geometries, etc.)

Generated material field

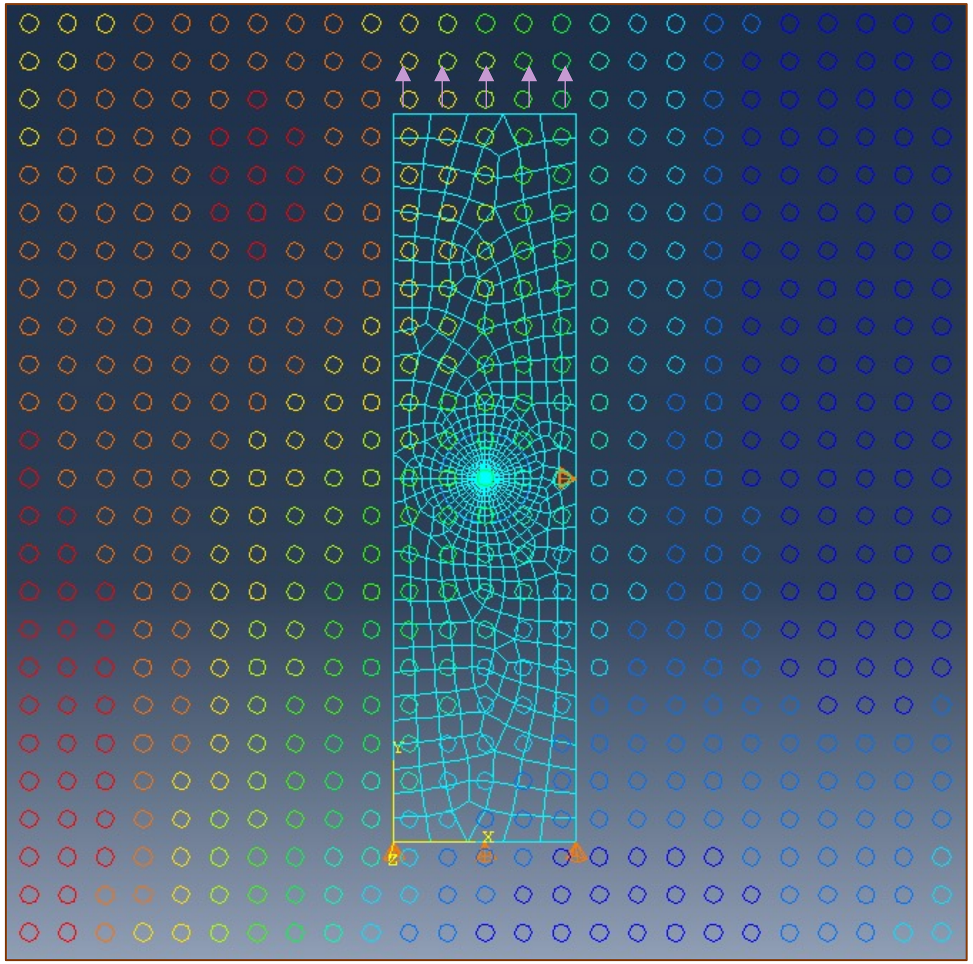
Example specimen configurations



- **Example**: Use finite element analysis (FEA) and Monte Carlo simulation to estimate probability distributions for crack mouth opening displacement (CMOD) due to spatial material variability

*The use of vendor and manufacturer names does not imply an endorsement by the U.S. Government, nor does it imply that the specified software is the best available

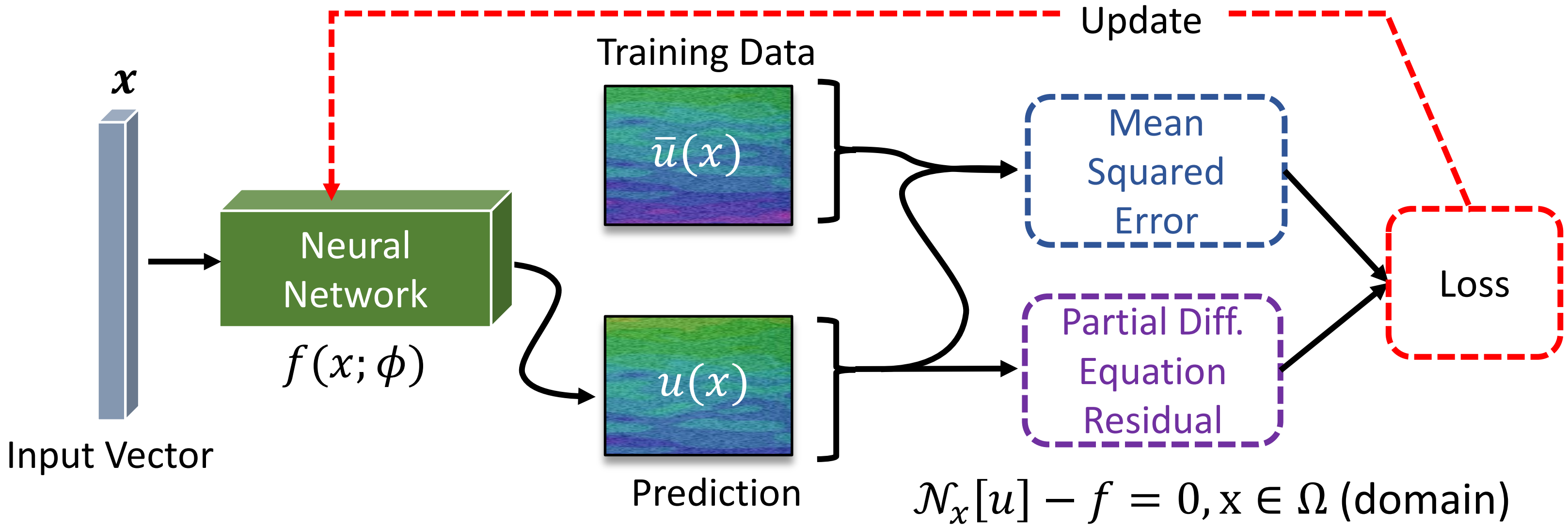
Cracked mesh overlaid on a sampled elastic modulus field (Abaqus* screenshot)



Higher modulus   Lower modulus

Physics-Informed Neural Networks

- Neural network trained to produce solution $u(x)$ to a partial differential equation
- Trained using two-part objective:
  1. Minimize mean squared error with respect to data
  2. Output physically admissible solutions throughout the problem domain

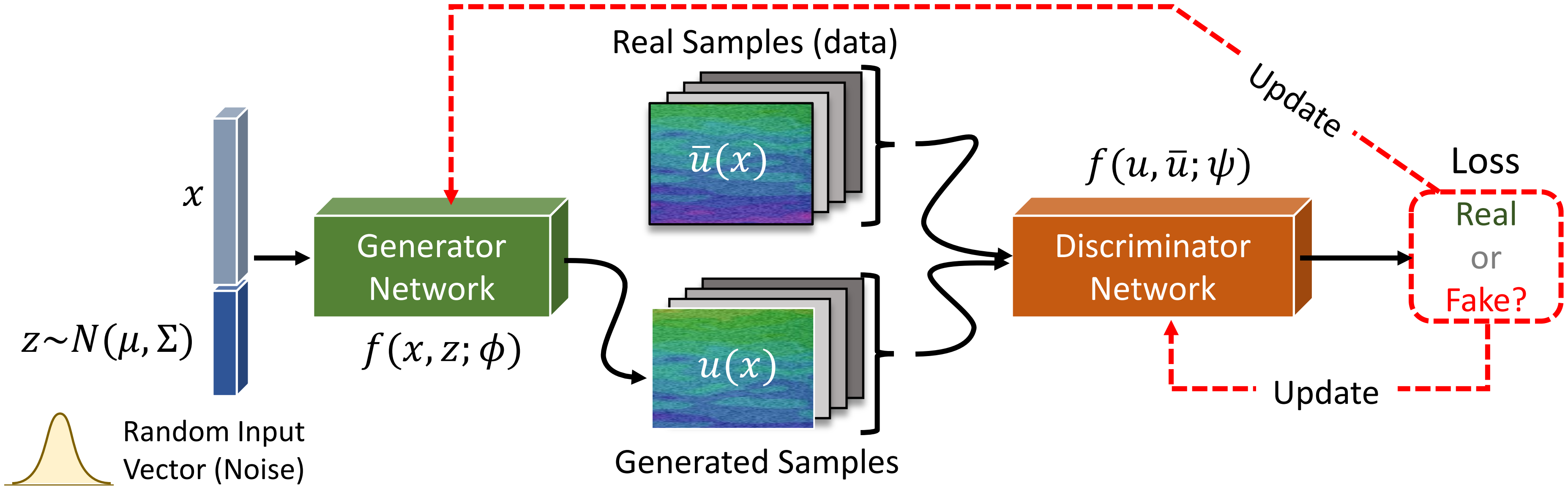


$\mathcal{N}_x[u] - f = 0, \mathrm{x} \in \Omega$ (domain)
$B_x[u] - b = 0, x \in \Gamma$ (boundary)

Raissi et al. "Physics-informed neural networks: A deep learning framework for solving forward and inverse..." *J. Comp. Phys.* (2019)

Generative Adversarial Networks

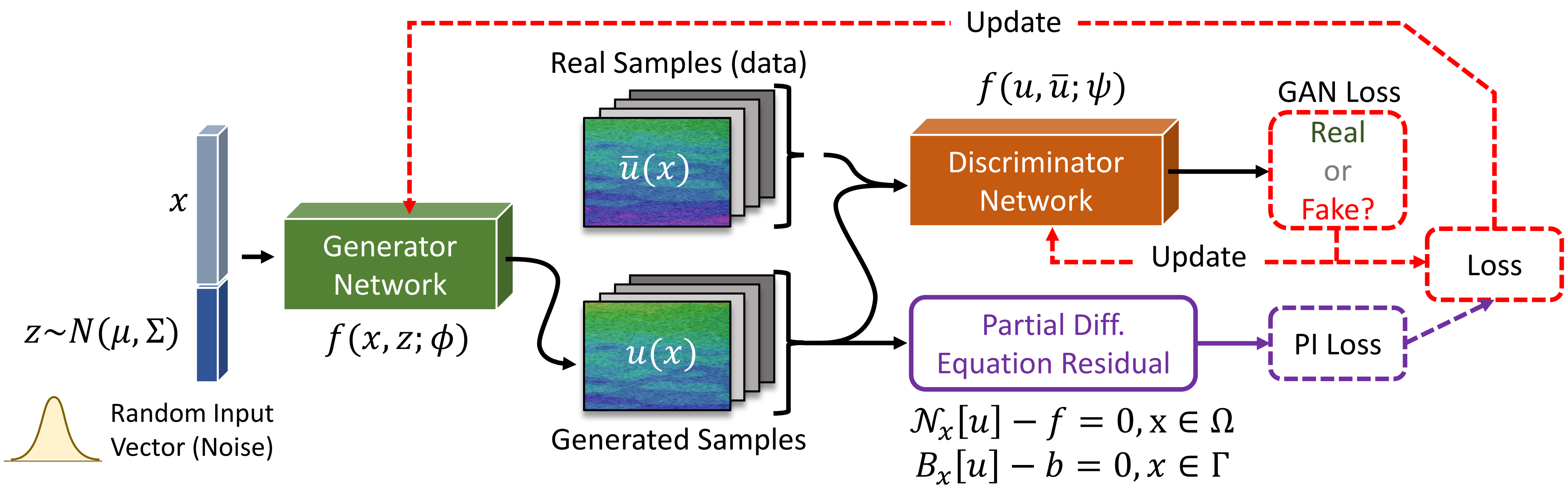
- Deep generative models that learn high-dimensional probability distributions
- Training objective: *discriminator network* tries to discern real from fake while *generator network* tries to fool the discriminator



Goodfellow, Ian J., et al. "Generative adversarial networks." arXiv preprint arXiv:1406.2661 (2014)

Physics-Informed GANs

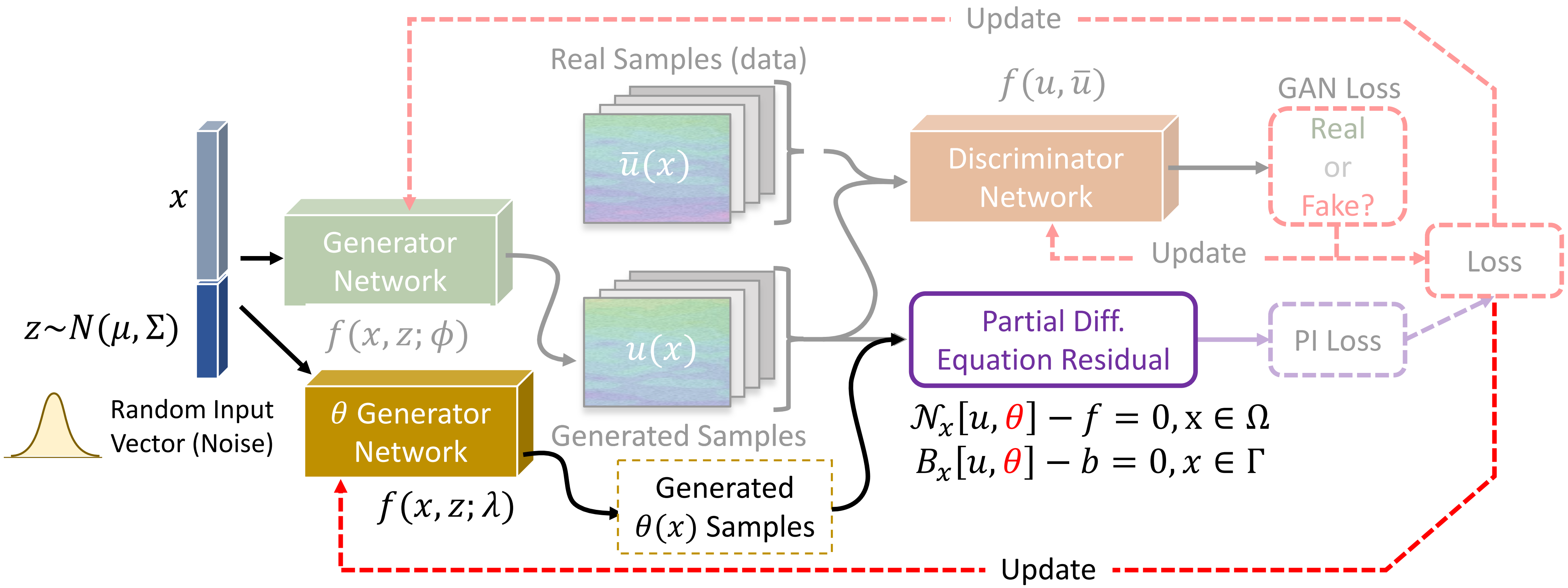
- Combination of PINNs and GANs to learn physically-admissible, probabilistic solution, $\boldsymbol{u}(x)$



Yang et al. "Physics-informed generative adversarial networks for stochastic differential equations." arXiv preprint arXiv:1811.02033 (2018)

Physics-Informed GANs

- Additional parameters, θ, can be incorporated by introducing additional generators
- Enables solving complex, probabilistic inverse problems (no direct θ observations)



Yang et al. "Physics-informed generative adversarial networks for stochastic differential equations." arXiv preprint arXiv:1811.02033 (2018)

Physics-informed GANs (PIGANs)

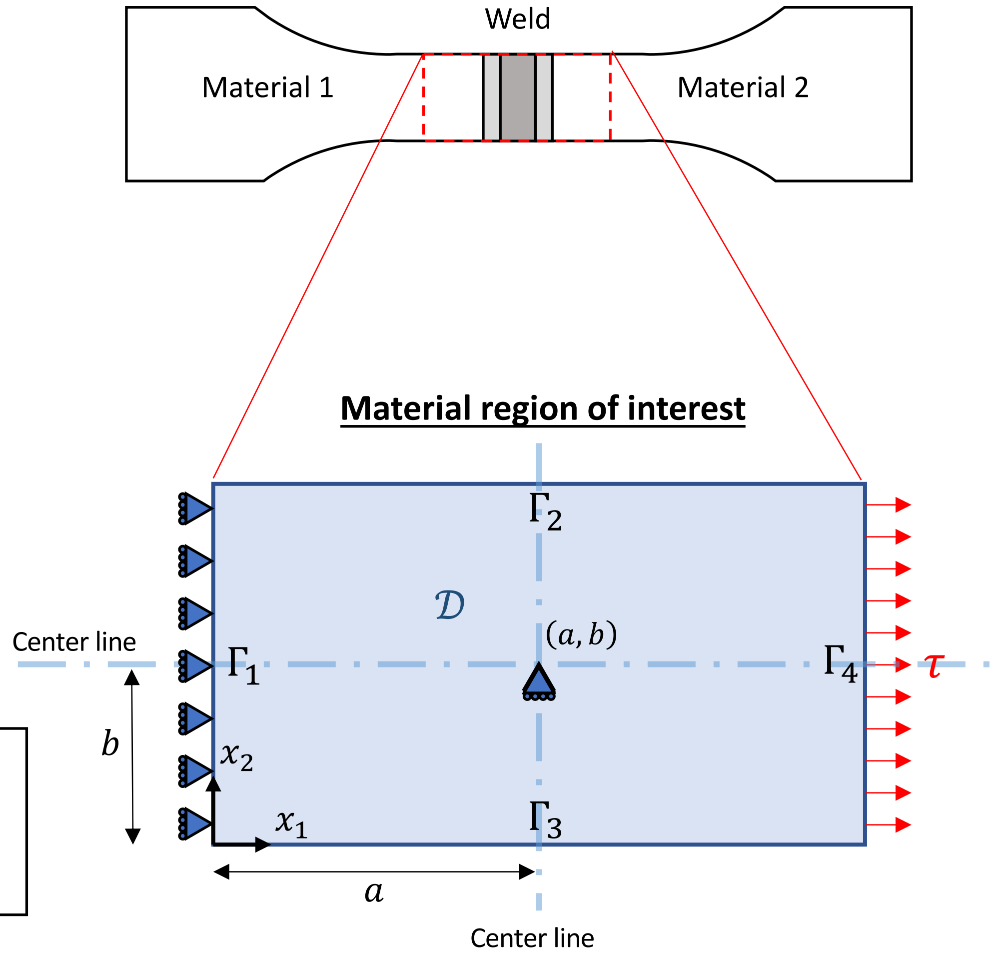
PIGANs Demonstration Overview

- **Objective**: Quantify uncertainty in the elastic modulus field, E, across a bi-material weld given surface displacement measurements from 85 tensile tests
- **Physics:**
  - Measured quantity:
    - $\mathbf{u}(\mathbf{x}, \omega)$ for $\mathbf{x} \in \mathcal{D}, \ \omega \in \Omega$
  - Governing partial differential equation (assuming two-dimensional plane stress, isotropic elasticity, small strains):
    - $\nabla \cdot \boldsymbol{\sigma}(\mathbf{u}, E) = 0$
  - Boundary conditions:
    - $u_1 = 0, \ \mathbf{x} \in \Gamma_1$
    - $u_2 = 0, \ \mathbf{x} = [a, b]$
    - $\boldsymbol{\sigma}(\mathbf{u}, E) \cdot \mathbf{n} = [\tau, 0], \ \ \mathbf{x} \in \Gamma_4$ (traction free Γ_2 and Γ_3)

Definitions

$\mathbf{x}$ – position vector	E– elastic modulus field; i.e., $E(\mathbf{x}, \omega)$
$\mathbf{u}$ – displacement vector	$\mathbf{n}$ – surface outward normal vector
ω – random event in sample space, Ω	τ – applied traction
$\boldsymbol{\sigma}$ – second order stress tensor	

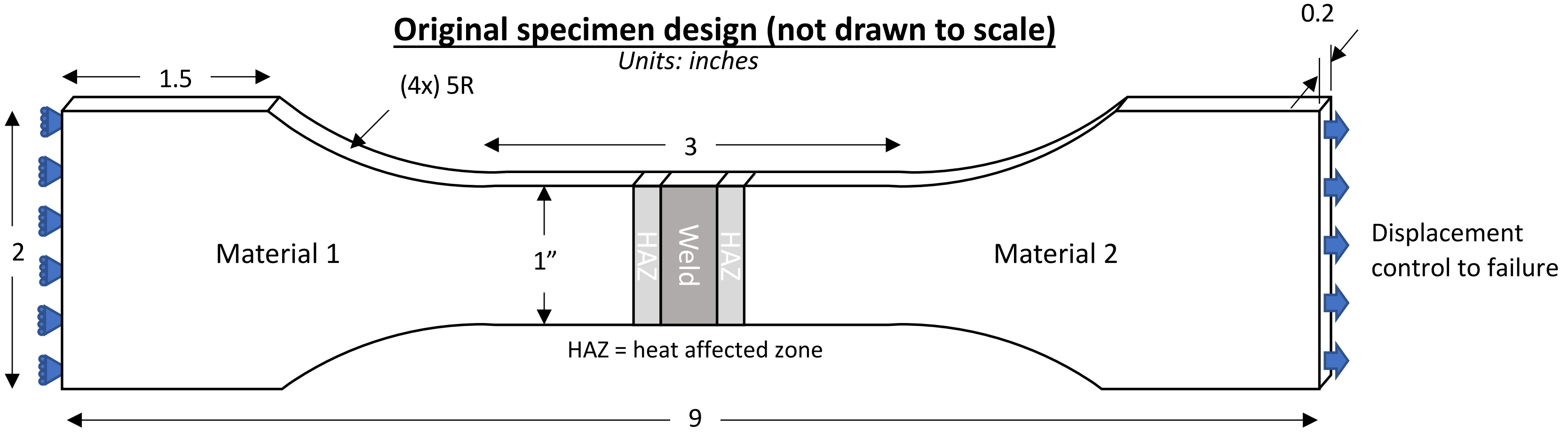
Test specimen schematic (not to scale)



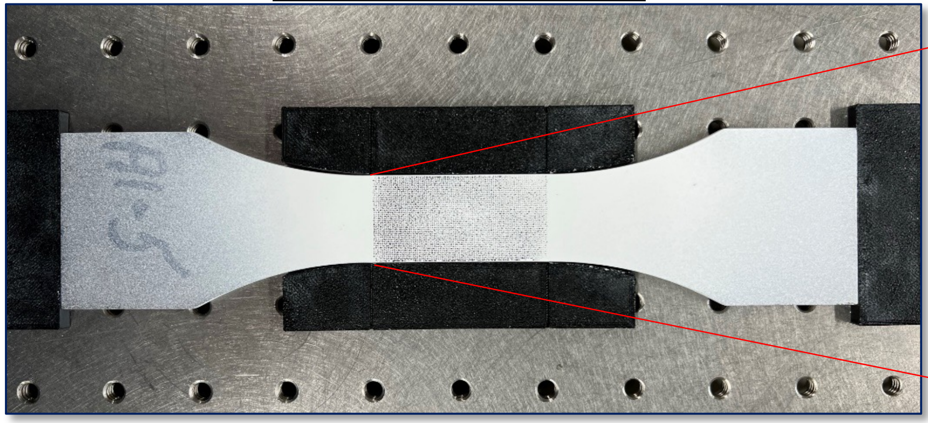
Experimental Setup

Original specimen design (not drawn to scale)

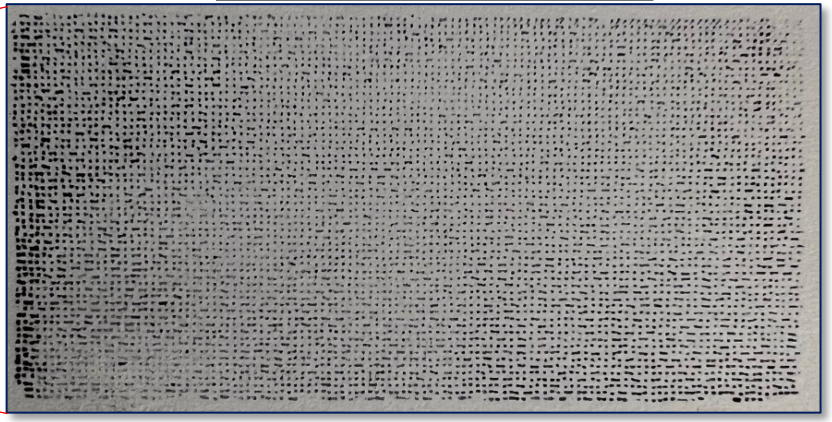
Units: inches



DIC pattern stamping jig

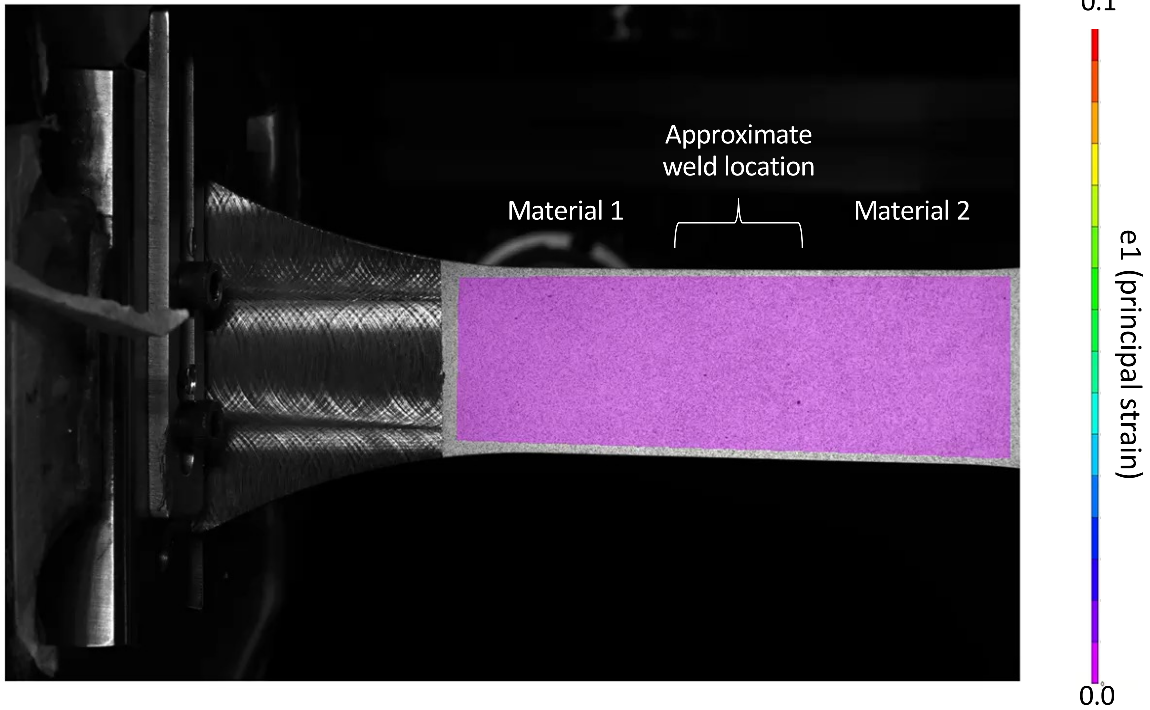


Close up view of DIC pattern

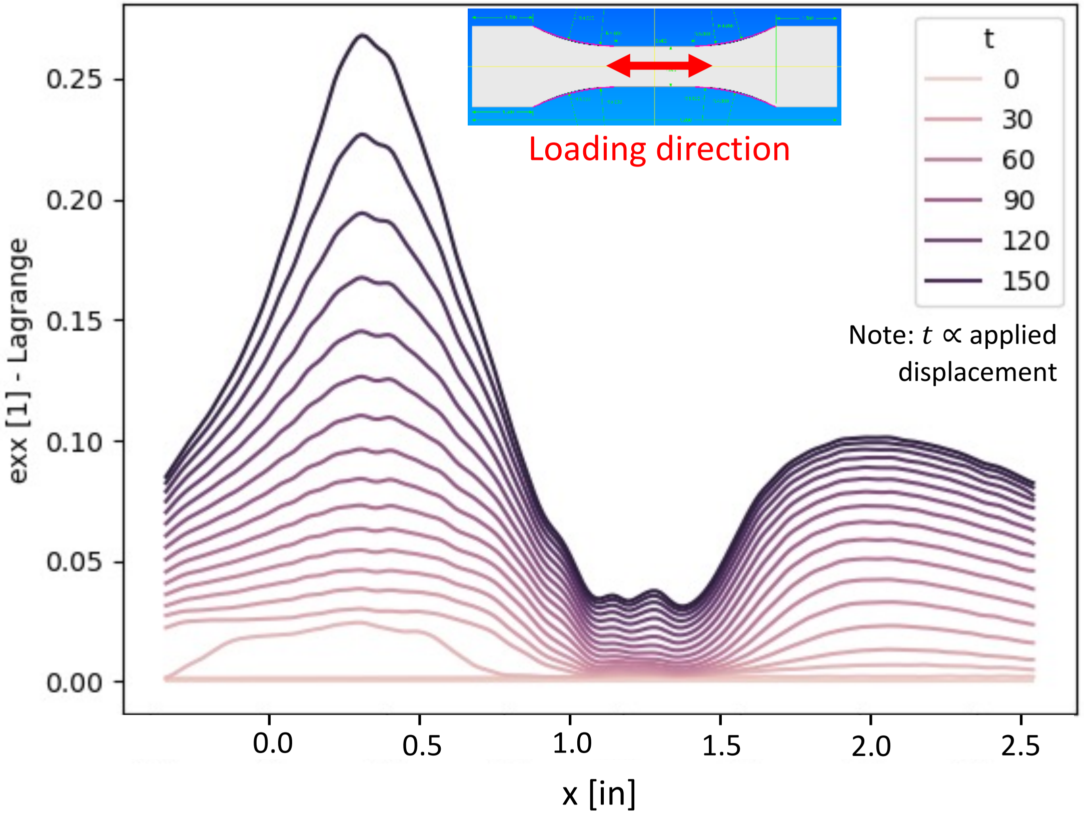


DIC

1/8 inch thick coupon, strain in loading direction

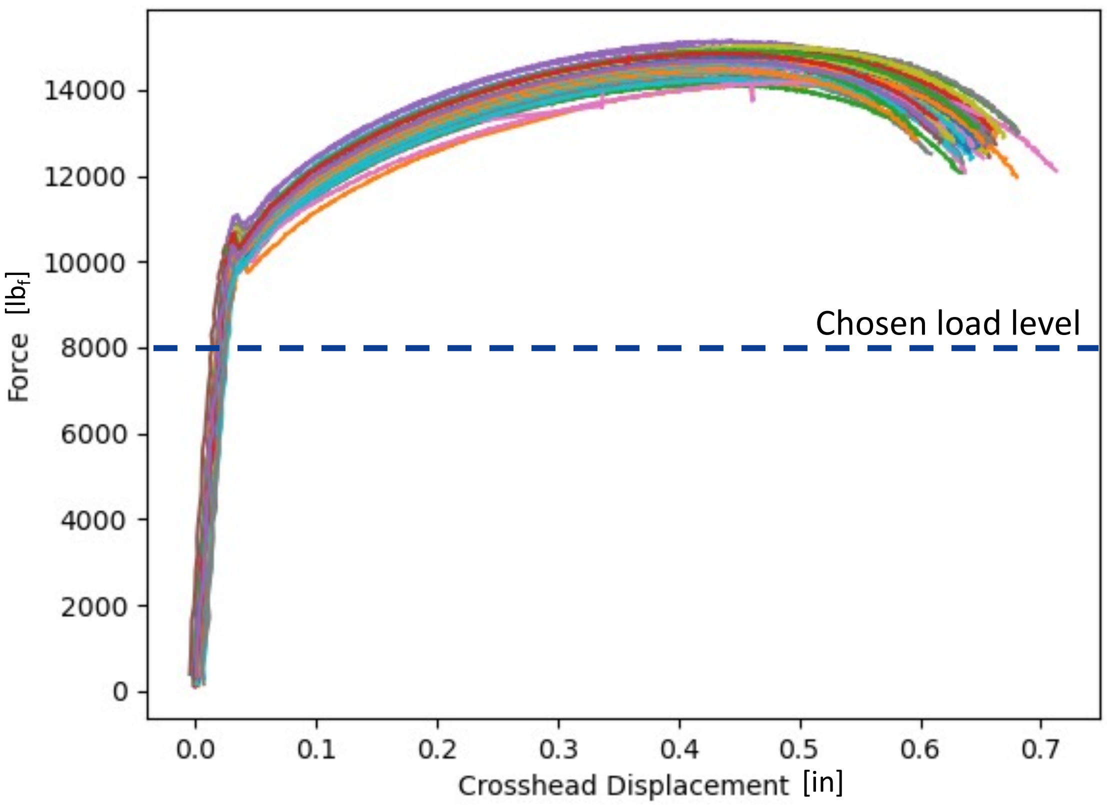


Strain along center of coupon in loading direction

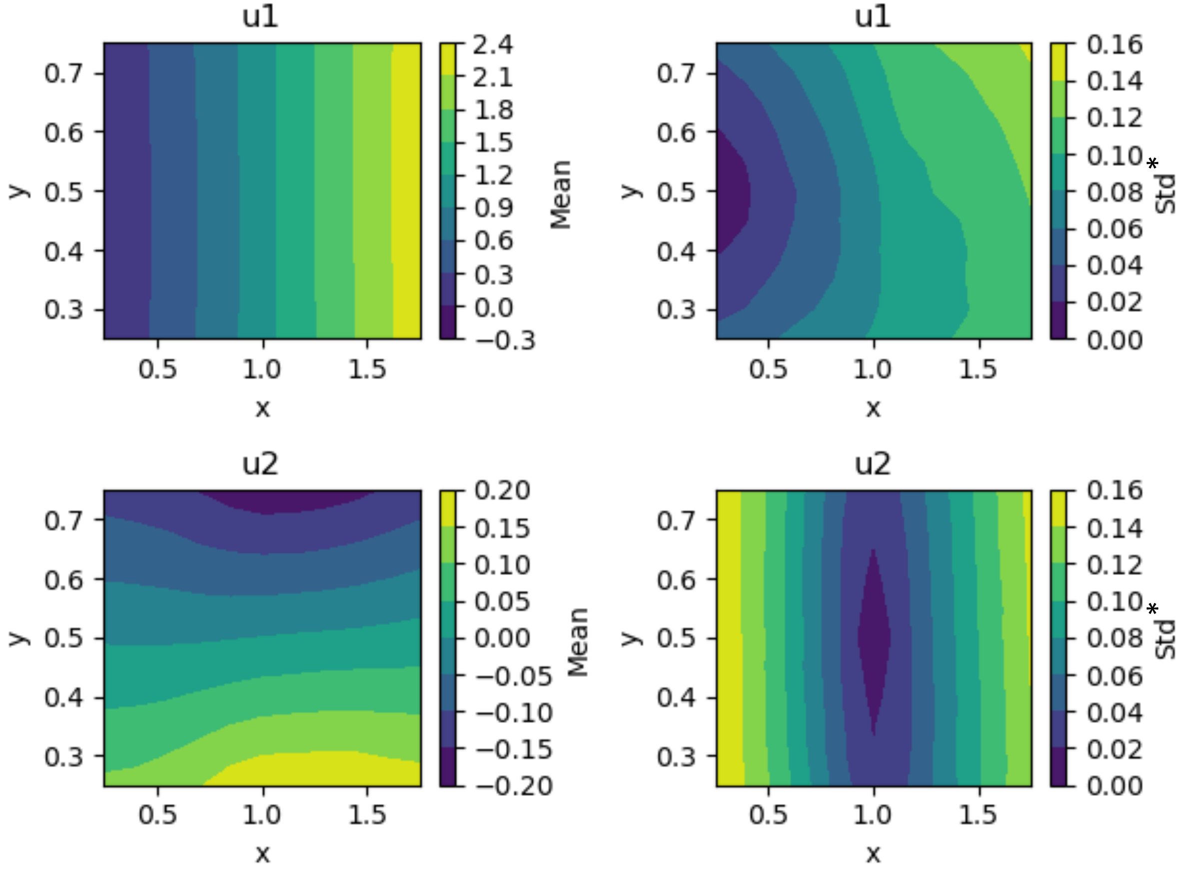


Summary of Test Data

Global force versus displacement response for all 85 tests



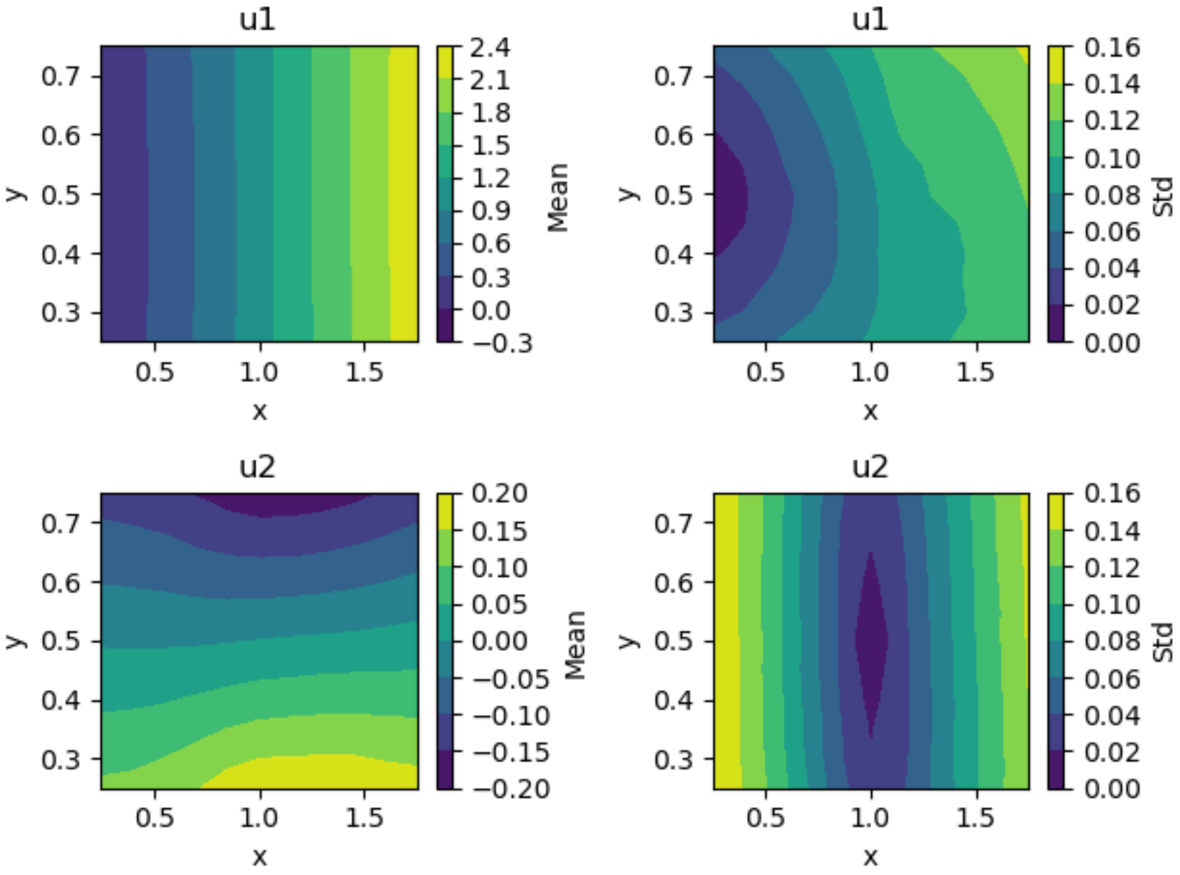
Normalized displacement statistics for all 85 tests



*Std = Standard deviation

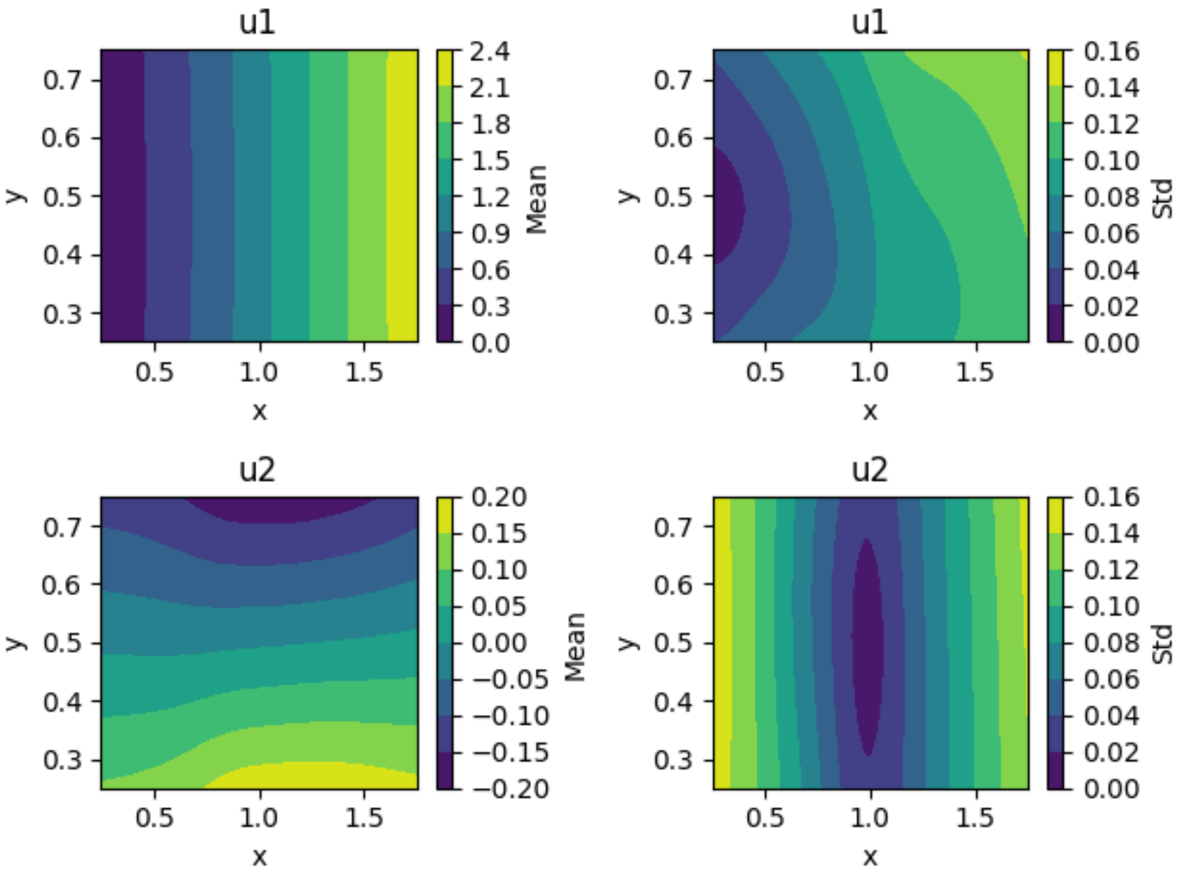
Results: Displacement Statistics

Normalized displacement statistics from all 85 tests



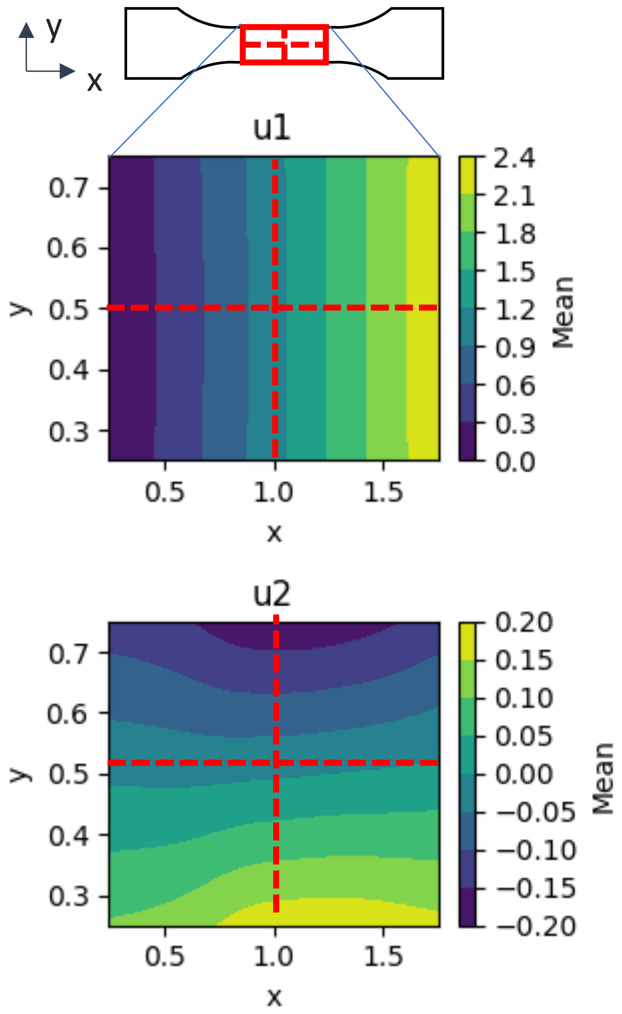
Predicted normalized displacement statistics

(1000 PIGAN-generated samples)



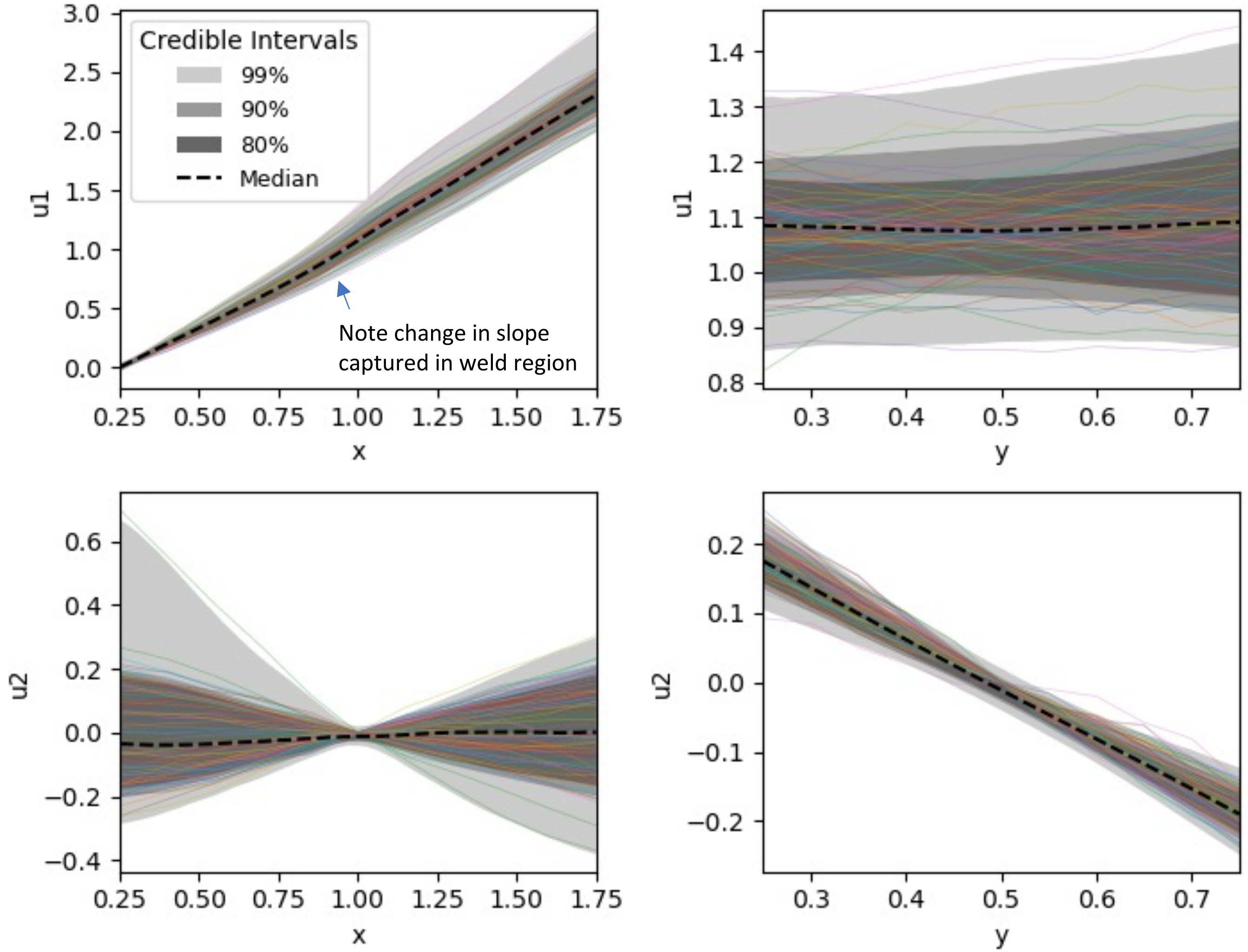
Results: Displacement Distributions

Mean displacement fields for reference



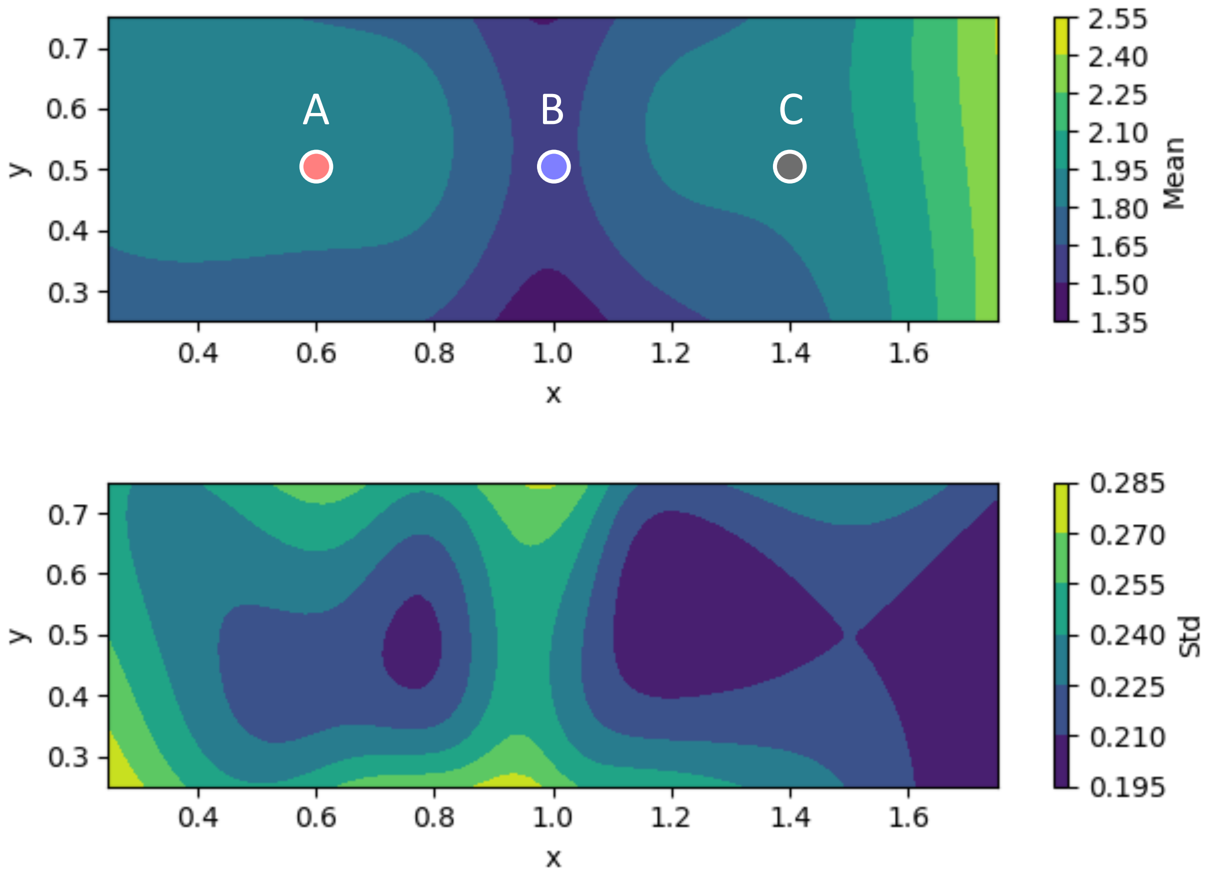
Cross-sectional displacement distributions compared to test data

(10,000 PIGAN-generated samples, colored lines indicate DIC data)

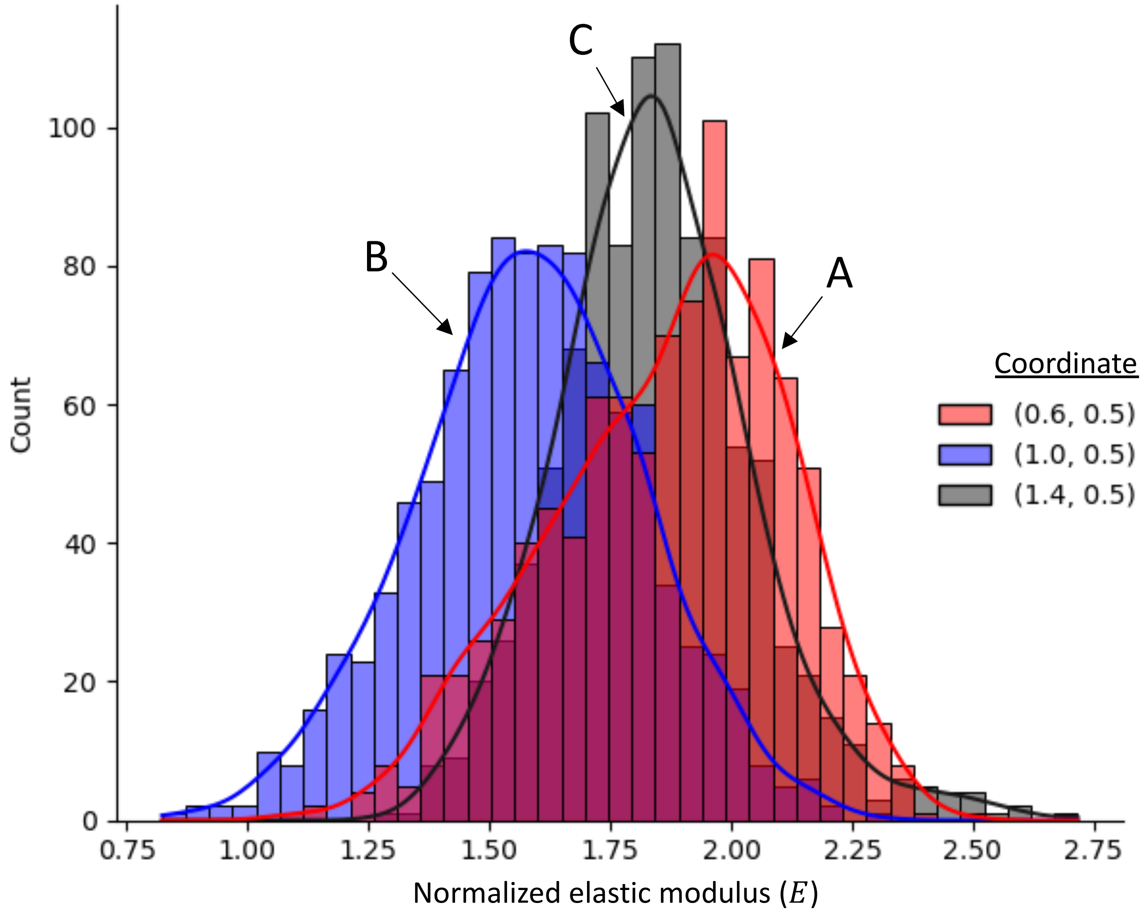


Results: Predicting Elastic Modulus

Predicted normalized elastic modulus statistics
(1000 PIGAN-generated samples)

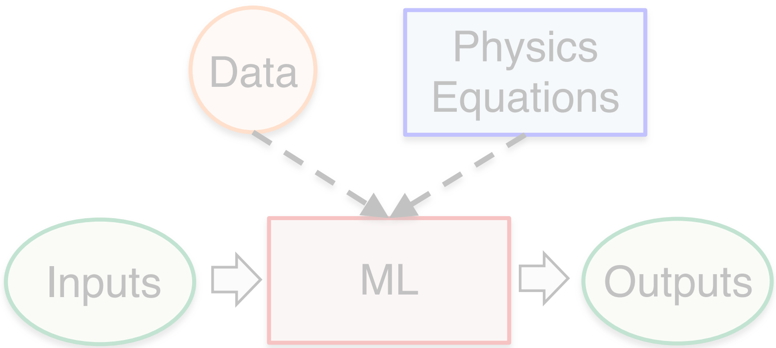


Predicted normalized elastic modulus at three points
(1000 PIGAN-generated samples)



Combining Physics and ML

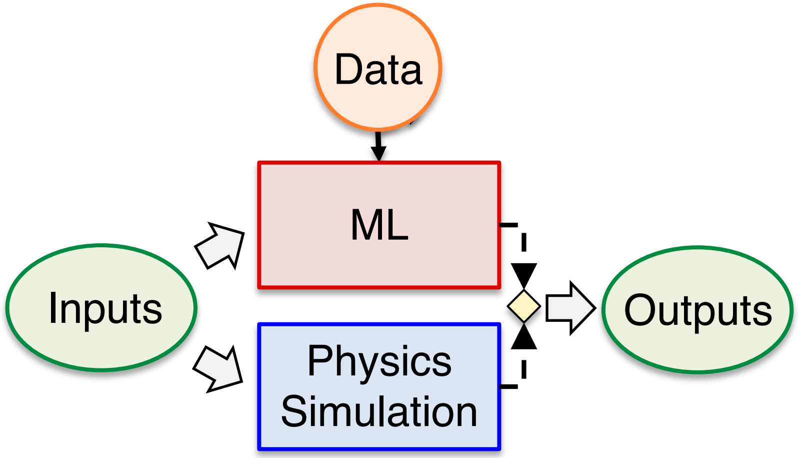
Physics-informed deep learning



➢ ***Integrate*** physical laws into the training of ML models

1. Physics-informed neural networks. Raissi et al. 2019.
2. Physics-informed generative adversarial networks Yang et. al. 2018

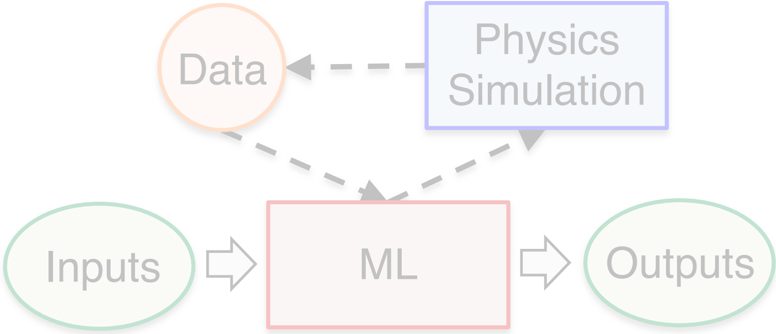
Multi-model Monte Carlo



➢ ***Fuse*** predictions from ML and physics-based models

3. Multilevel Monte Carlo. Giles et al. 2015
4. Multifidelity Monte Carlo. Peherstorfer et al. 2016
5. Approximate Control Variates. Gorodetsky et al. 2019

Active Learning



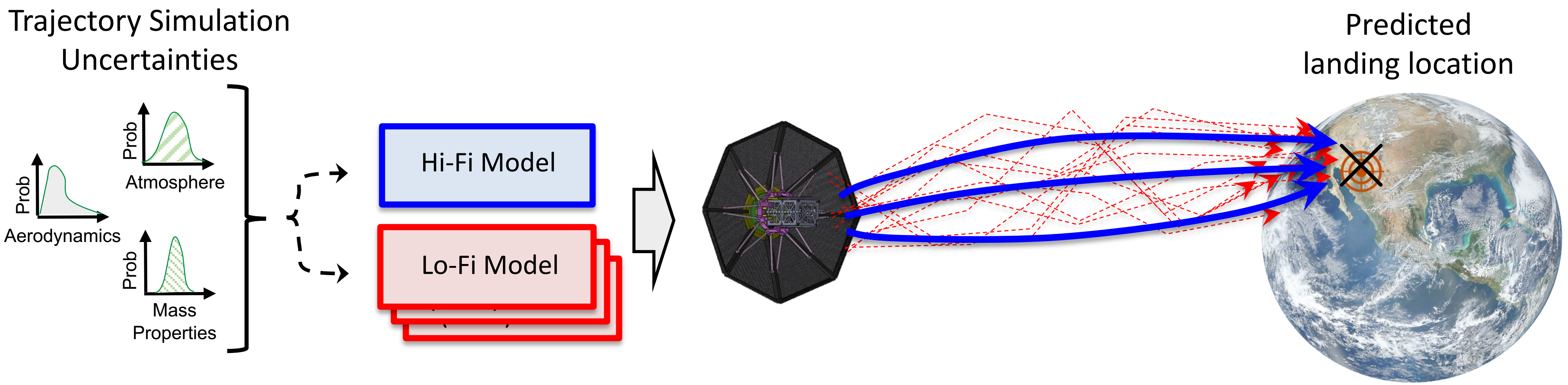
➢ ***Guide*** physics-based training data generation using ML Model

6. Active learning in practice. Settles 2011.
7. Active learning for reliability analysis. Bichon et al. 2008.

- **Application**: Material identification for certification by analysis
- **Application*:*** Trajectory simulation for EDL
- **Application:** Spacesuit reliability analysis

Multi-Model Monte Carlo for Trajectory Simulation

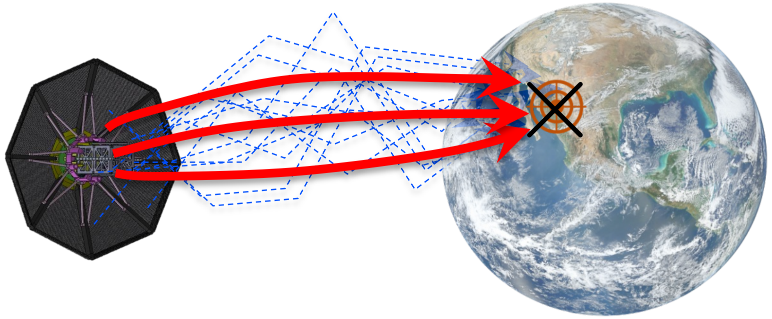
- **Motivation**: high-precision guidance software for EDL
- **Challenge**: enabling fast and accurate uncertainty propagation
  - Using Hi-Fi trajectory simulations can be too slow
  - Using low-fidelity (Lo-Fi) surrogate models lead to biased estimators
- **Approach**: Combine predictions from a Hi-Fi model with Lo-Fi model(s) using multi-model Monte Carlo (MC) to estimate trajectory state statistics



Collaborators: Anthony Williams (NASA), Som Dutta (NASA), Justin Green (NASA), Luke Morrill (NASA), Sam Nieomoeller (UCLA)

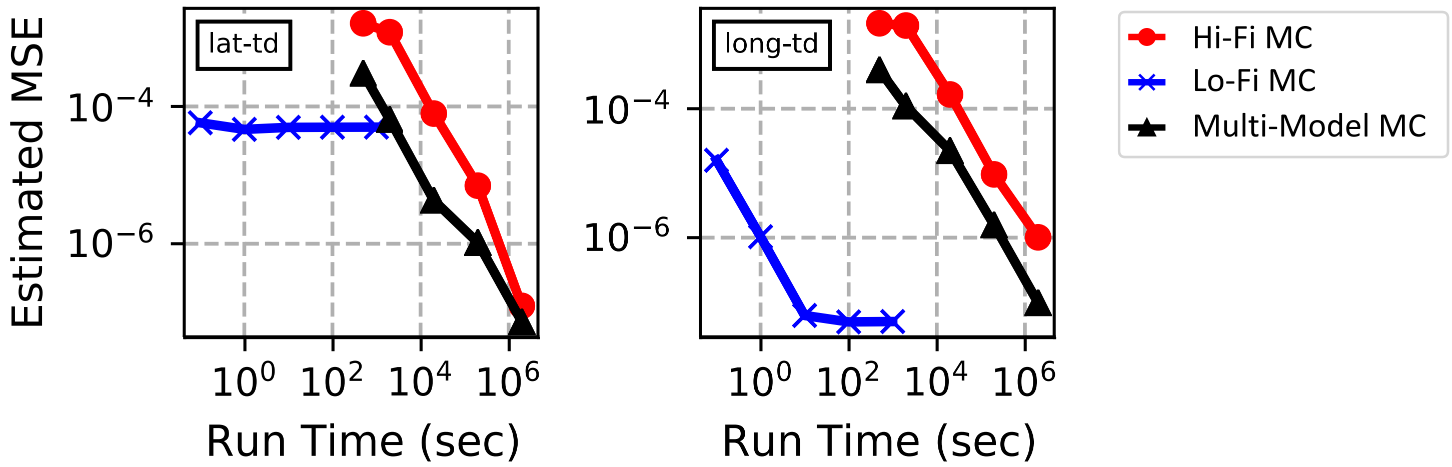
Multi-Model Monte Carlo for Trajectory Simulation

- **Example:** predicting the expected landing location for an umbrella heatshield reentering Earth atmosphere*



Models:
Hi-Fi: Program to Optimize Simulated Trajectories II (POST2)
Lo-Fi: Support vector machine (SVM) surrogate model

Error versus Run Time for Expected Landing Location

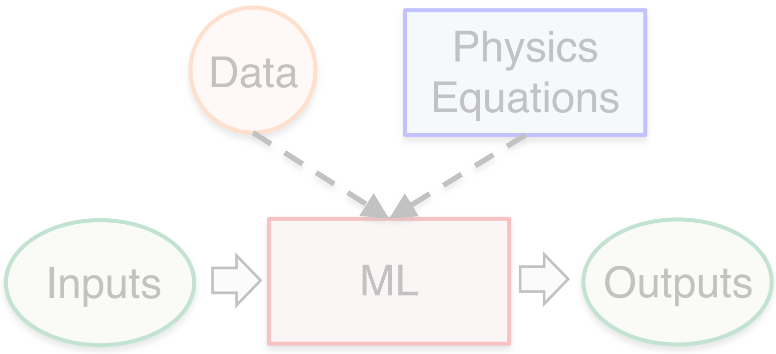


- ➢ High-fidelity MC requires the most run time for accurate estimators
- ➢ Low-fidelity MC has irreducible error due to estimator bias
- ➢ Multi-model MC is **more efficient** than high-fidelity MC and is **unbiased**

*J. E. Warner et al. *Multi-Model Monte Carlo Estimators for Trajectory Simulation*. AIAA SciTech. 2021.

Combining Physics and ML

Physics-informed deep learning

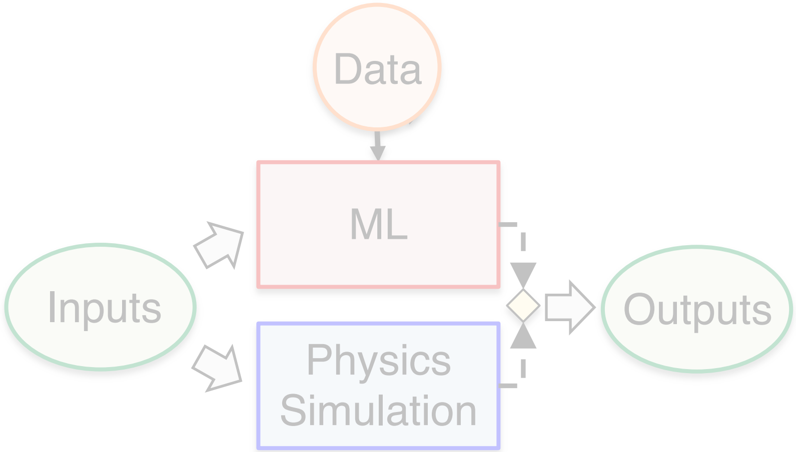


➢ ***Integrate*** physical laws into the training of ML models

1. Physics-informed neural networks. Raissi et al. 2019.
2. Physics-informed generative adversarial networks Yang et. al. 2018

- **Application**: Material identification for certification by analysis

Multi-model Monte Carlo

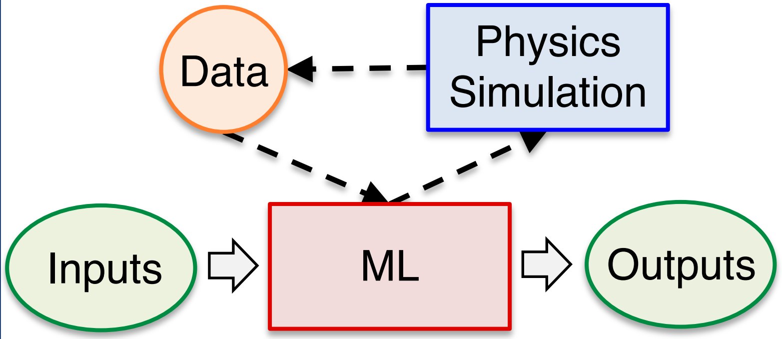


➢ ***Fuse*** predictions from ML and physics-based models

3. Multilevel Monte Carlo. Giles et al. 2015
4. Multifidelity Monte Carlo. Peherstorfer et al. 2016
5. Approximate Control Variates. Gorodetsky et al. 2019

- **Application:** Trajectory simulation for EDL

Active Learning



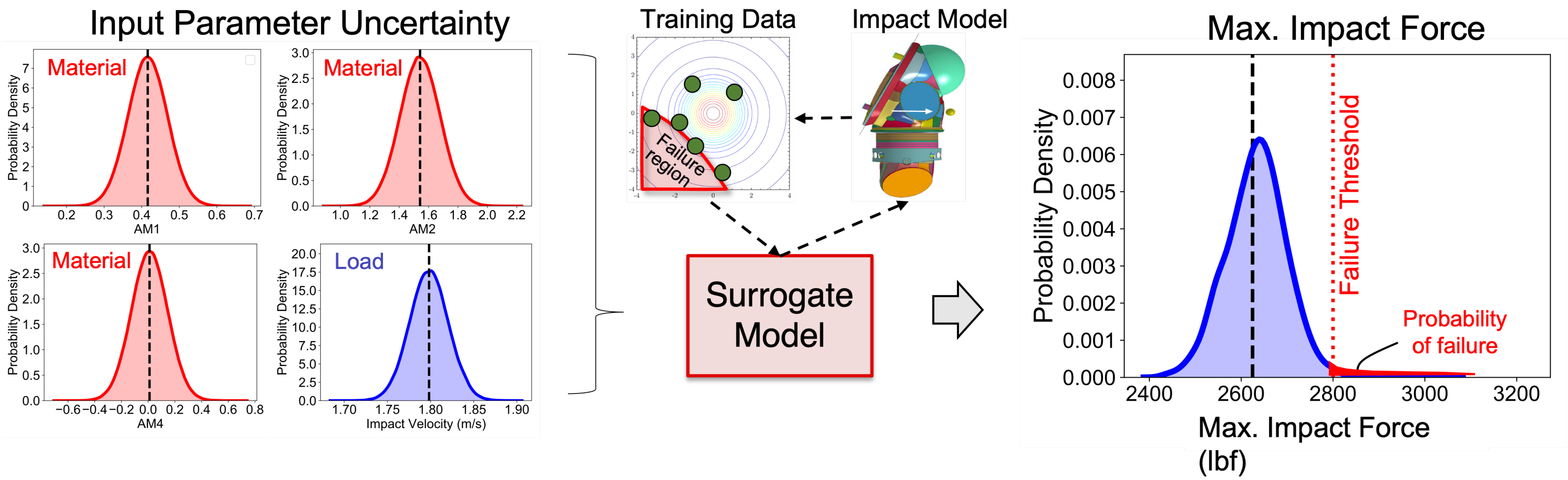
➢ ***Guide*** physics-based training data generation using ML Model

6. Active learning in practice. Settles 2011.
7. Active learning for reliability analysis. Bichon et al. 2008.

- **Application:** Spacesuit reliability analysis

Active Learning for Reliability Analysis

- **Motivation**: reliability analysis of the xEMU spacesuit
- **Challenge**: predicting small failure probabilities with expensive damage simulations
  - Standard surrogate models are inaccurate for predicting extreme behavior (failure) with limited training data
- **Approach**: use *active learning* to train Gaussian Process (GP) surrogate models that are tailored for accurate failure predictions

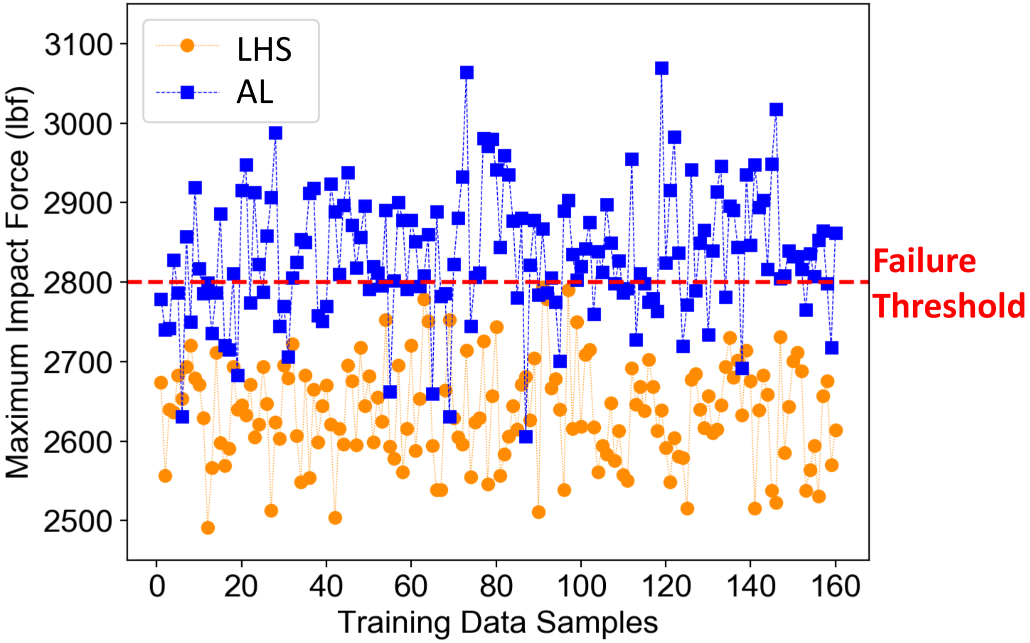


Collaborators: Robert Gramacy, Austin Cole, Annie Sauer (Virginia Tech)

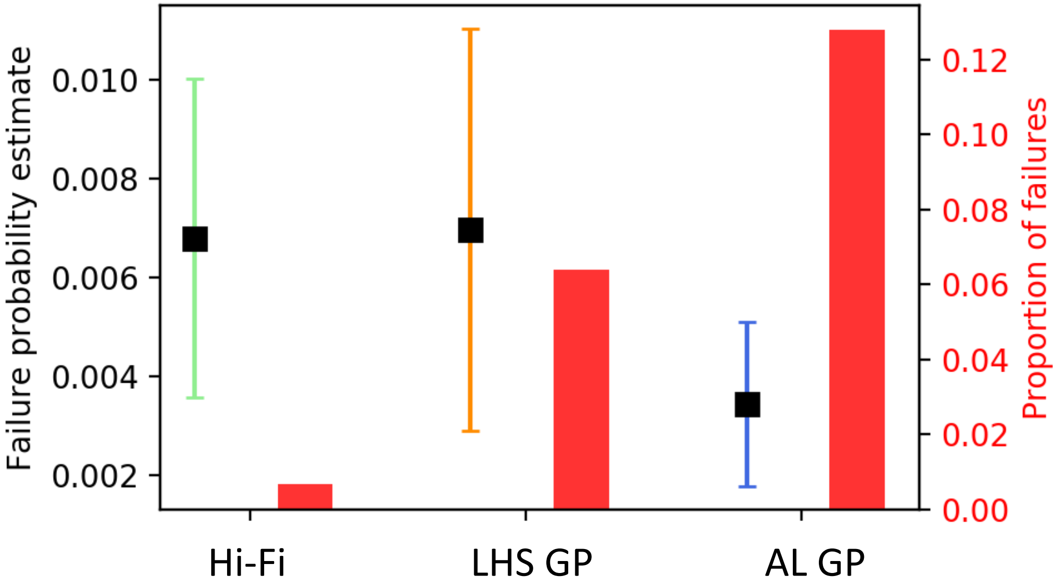
Active Learning for Reliability Analysis

- **Example:** Estimate the probability of impact failure in the spacesuit under material/impact velocity uncertainty*
- Compare surrogate-based solutions from:
  - **LHS GP** – trained from random **L**atin **H**yper**c**ube samples
  - **AL GP** – trained using **A**ctive **L**earning targeting failure region

Training Data Selection: LHS vs. AL



Probability of Failure Estimate Comparison



2500 model evaluations | 450 model evaluations | 450 model evaluations

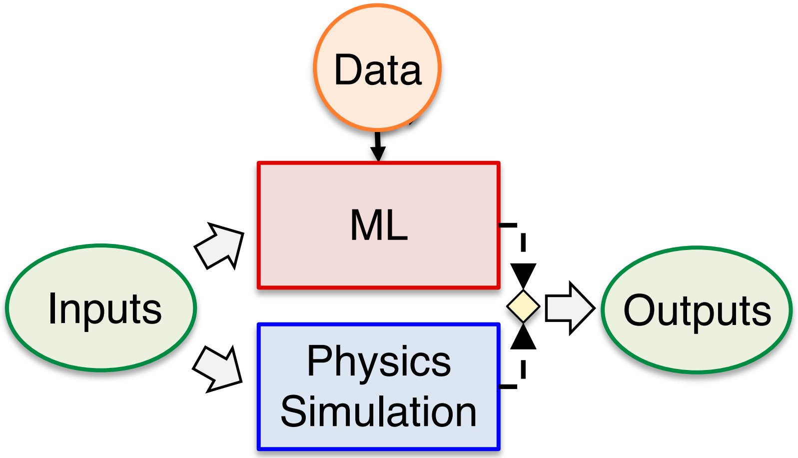
* D. A. Cole et al. *Entropy-Based Adaptive Design for Contour Finding and Estimating Reliability*. Journal of Quality Technology. 2023.

Summary: AI for Aerospace Engineering

- Replacing physics-based models with supervised machine learning models (surrogate modeling):
    1. Always validate using a separate testing dataset
    2. Avoid extrapolation outside training data range
    3. Account for non-negligible surrogate errors in analysis
- Combining physics-based and data-driven modeling:
    1. **Multi-model Monte Carlo**: fuse predictions from ML with physics-based models for unbiased *trajectory simulation* estimators
    2. **Physics-informed ML**: integrate physical laws into the training of ML models for *material identification*
    3. **Active learning**: guide physics-based training data generation using ML models for *spacesuit reliability analysis*

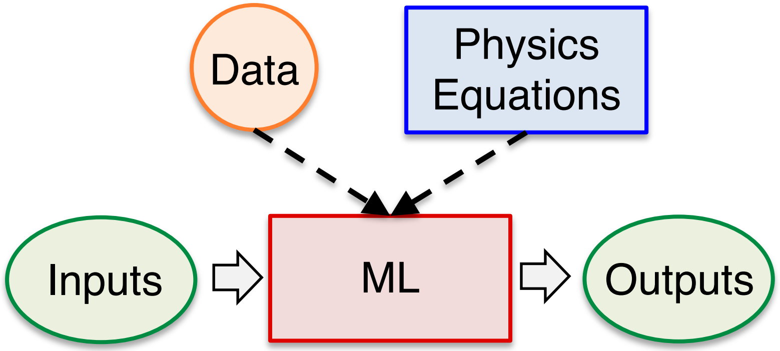
References

Multi-model Monte Carlo



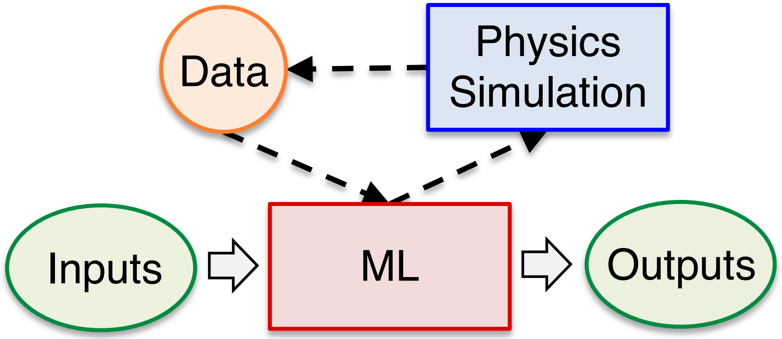
- J. E. Warner, G. F. Bomarito, L. Morrill, P. E. Leser, W. P. Leser, R. A. Williams, S. Dutta, S. Nieomoeller. *Multi-Model Monte Carlo Estimators for Trajectory Simulation*. 2021 AIAA SciTech Conference.
- G. F. Bomarito, P. E. Leser, J. E. Warner, W. P. Leser. *On the Optimization of Approximate Control Variates with Parametrically Defined Estimators*. Journal Paper. Journal of Computational Physics. February 2022.
- G. F. Bomarito, J. E. Warner, P. E. Leser, W. P. Leser, L. Morrill. Multi Model Monte Carlo with Python (MXMCPy). NASA TM 2020-220585. May 2020.

Physics-informed Deep Learning



- J. E. Warner, J. Cuevas, P. E. Leser, G. F. Bomarito, W. P. Leser. *Inverse Estimation of Elastic Modulus Using Physics-Informed Adversarial Networks*. NASA TM 2020-5001300. May 2020.

Active Learning



- D. A. Cole, R. B. Gramacy, J. E. Warner, G. F. Bomarito, P. E. Leser, W. P. Leser. *Entropy-Based Adaptive Design for Contour Finding and Estimating Reliability*. Journal of Quality Technology. 2023.
- B. K. Stanford, K. E. Jacobson, A. Sauer, J. E. Warner. *Gradient-Enhanced Reliability Analysis of Transonic Aeroelastic Flutter*. 2022 AIAA Scitech Proceedings.

- **NASA Internships**: https://stemgateway.nasa.gov/public/s/explore-opportunities/internships
- **Email:** james.e.warner@nasa.gov