A Gentle Introduction to Uncertainty Quantification

Ilias Bilionis
Assistant Professor
Predictive Science Laboratory
School of Mechanical Engineering, Purdue University
www.predictivesciencelab.org
ibilion@purdue.edu

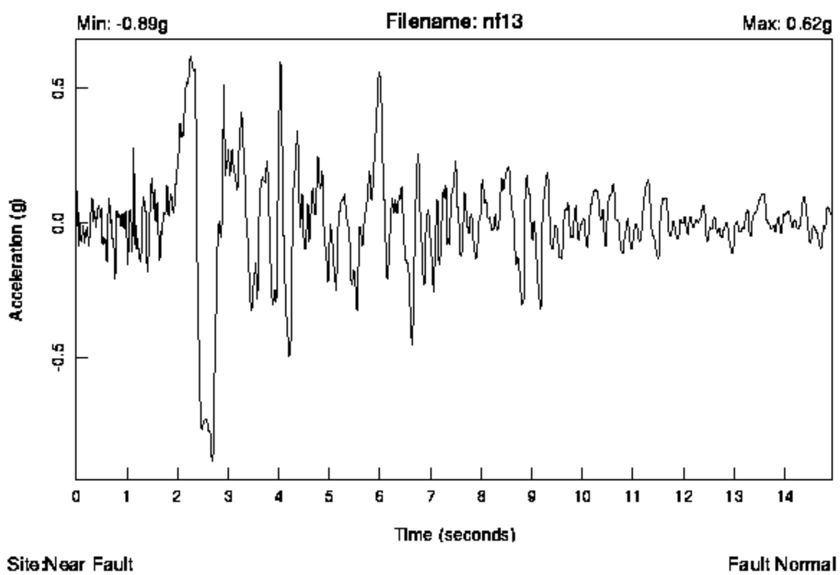
PREDICTIVE
SCIENCE LABORATORY

Objectives

- Learn what is uncertainty quantification (UQ) and why it is important.
- Be able to distinguish between an **error** and an **uncertainty**.
- Be able to distinguish between **aleatory** and **epistemic** uncertainties.
- Be able to use **probability theory** to represent both aleatory and epistemic uncertainties.
- Be able to compute the **probability of failure** using **Monte Carlo** simulations.

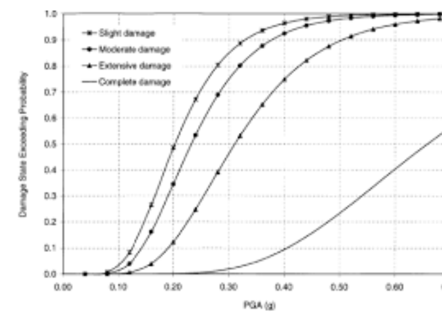
PREDICTIVE
SCIENCE LABORATORY

Where is UQ needed?
Building reliability



Uncertainty in external forcing

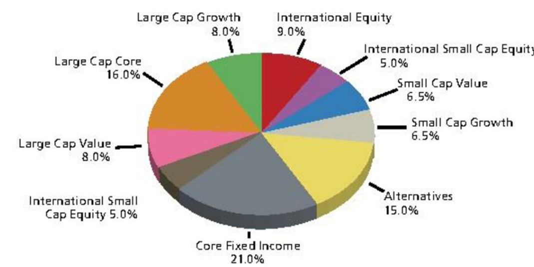
Simulation



Fragility Curve

PREDICTIVE
SCIENCE LABORATORY

Where is UQ needed?
Stock/bond portfolio allocation

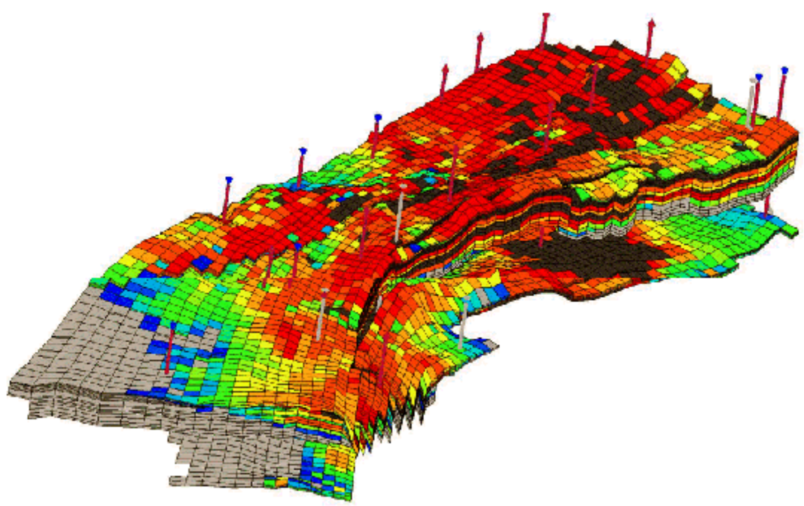


Portfolio Risk

Uncertainty in external forcing

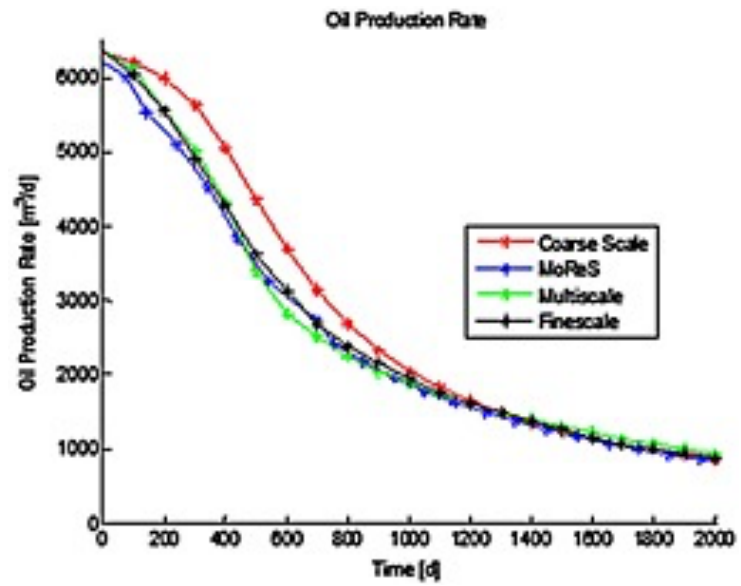
PREDICTIVE
SCIENCE LABORATORY

Where is UQ needed?
Oil reservoir operation



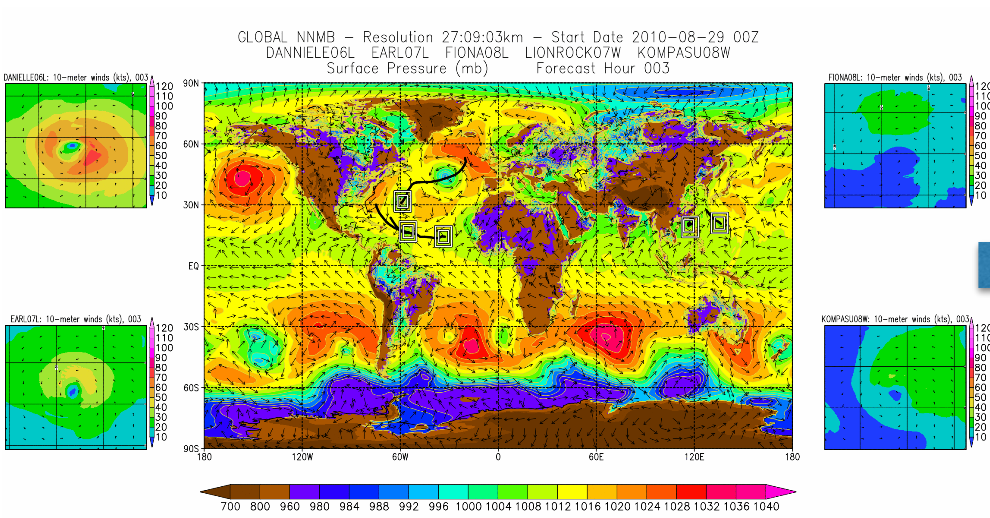
Uncertainty in field parameters

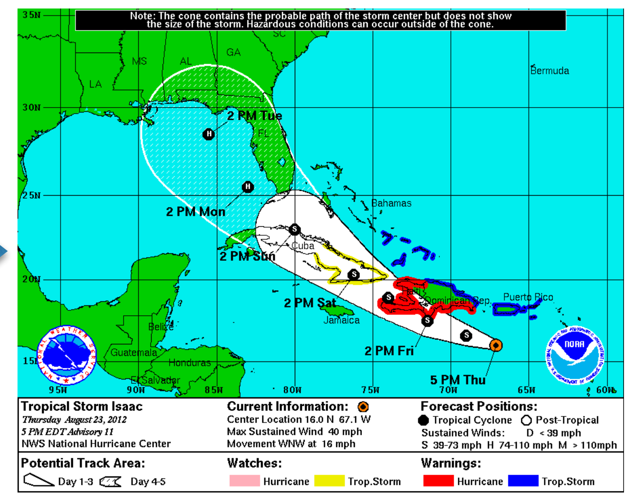
Simulation



Oil produced over time

Where is UQ needed? Prediction of extreme weather





Hurricane path

Uncertainty in initial conditions

PREDICTIVE SCIENCE LABORATORY

http://hwrf.aoml.noaa.gov/

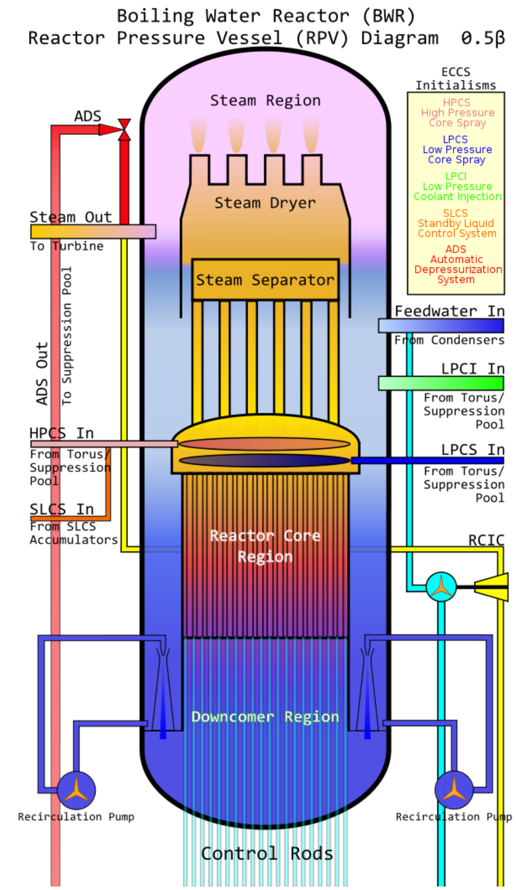
What is Uncertainty Quantification?

- Fukushima, Japan, March 11 2011
- After major earthquake, a 15 meter tsunami disables the power supply and cooling of three reactors. All three cores melted in the first 3 days.
- 100,000 people were evacuated.
- It took about a year to cool down the reactors.

PREDICTIVE
SCIENCE LABORATORY

What is Uncertainty Quantification?

- What is the probability of a core meltdown?
- What do we need to know in order to compute it?
- How can we reduce it?
- What if we have missed something…



Cooling systems of a boiling water reactor

PREDICTIVE
SCIENCE LABORATORY

Formal Definition of Uncertainty Quantification

"Uncertainty quantification (UQ) is the science of quantitative characterization and reduction of uncertainties in applications. It tries to determine how likely certain outcomes are if some aspects of the system are not exactly known."

–Wikipedia

In plain words…

Models

Experiments

Predictions about
the real world

and then…

Optimize engineering systems under this uncertainty!

Errors & Uncertainties

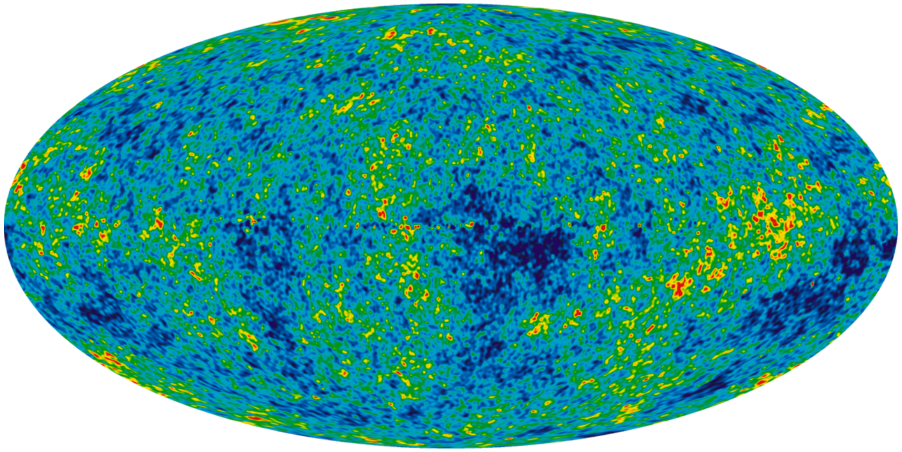
- **Errors:** are associated with the translation of math into computer code. Examples of errors:
  - round-off errors, convergence issues
  - implementation bugs…
- **Uncertainties:** are associated with the specification of the physical model:
  - values of various parameters
  - initial & boundary conditions, external forcing
  - constitutive laws (i.e., the physics themselves)

PREDICTIVE
SCIENCE LABORATORY

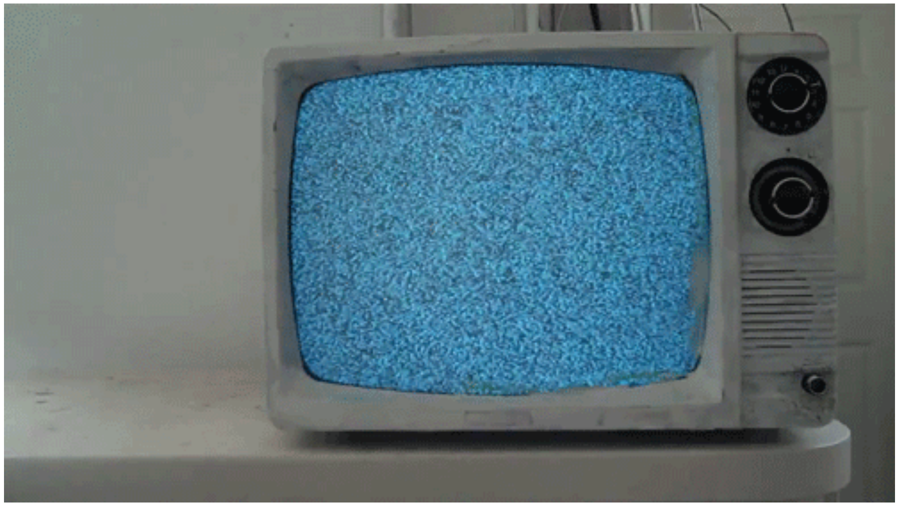
Aleatory vs Epistemic Uncertainty

- **Aleatory:** naturally occurring randomness that we cannot (or do not know how to) reduce.
- **Epistemic:** uncertainty due to lack of knowledge that we can reduce by paying a price.

Aleatory Uncertainty Example: Cosmic Microwave Background

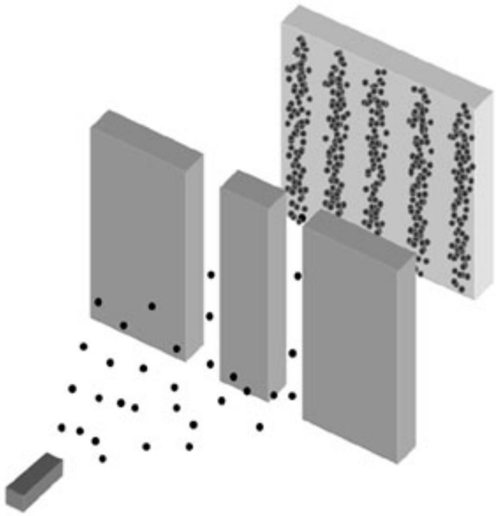


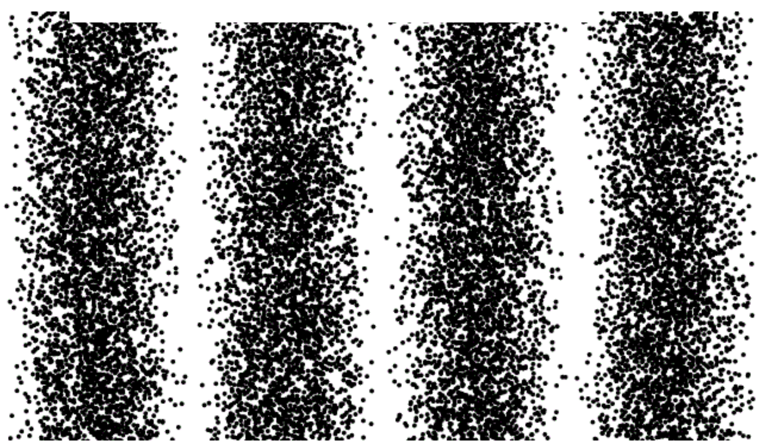
Thermal radiation left over from the Big Bang.
Arno Penzias, Rober Wilson, 1978 Nobel Prize



PREDICTIVE
SCIENCE LABORATORY

Aleatory Uncertainty Example: Double Slit Experiment





"[The quantum slit experiment] is a phenomenon which is impossible […] to explain in any classical way, and which has in it the heart of quantum mechanics. In reality it contains, the *only* mystery of [quantum mechanics]."

-Richard Feynman, (1965)

PREDICTIVE
SCIENCE LABORATORY

Aleatory Uncertainty Example: Turbulence

PREDICTIVE
SCIENCE LABORATORY

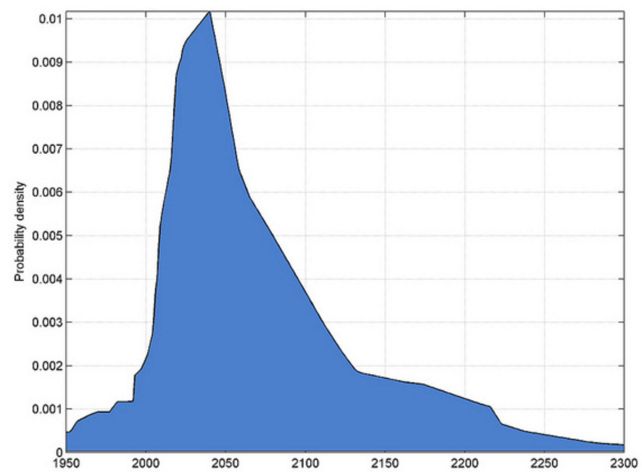
How to deal with aleatory uncertainty?

- Consider an aleatory variable ***s***.
- The intrinsic randomness of ***s*** is described by a *probability density p(**s**).*

$$\mathcal{D} = \{s_1, \ldots, s_n\}$$

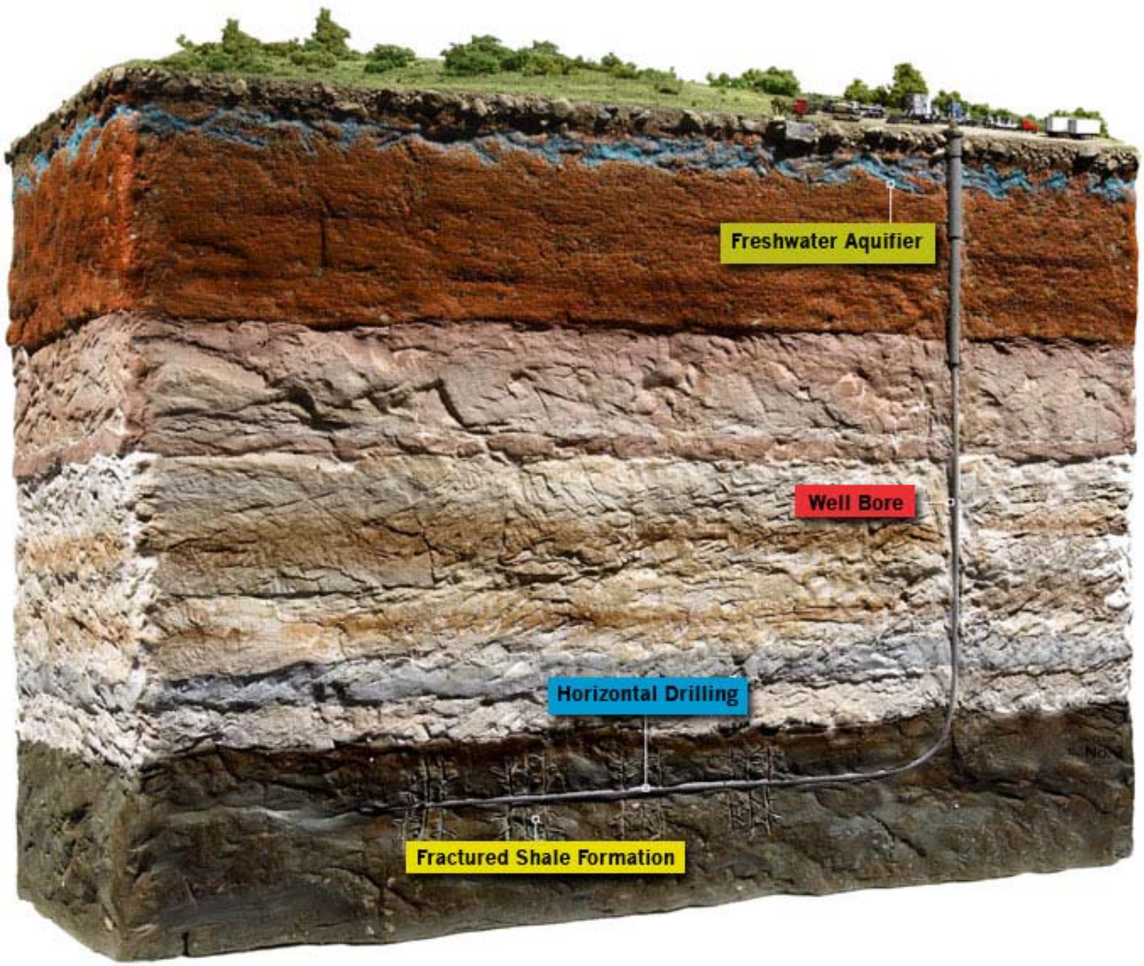
uncertainty quantification

$$p(s \mid \mathcal{D})$$



PREDICTIVE
SCIENCE LABORATORY

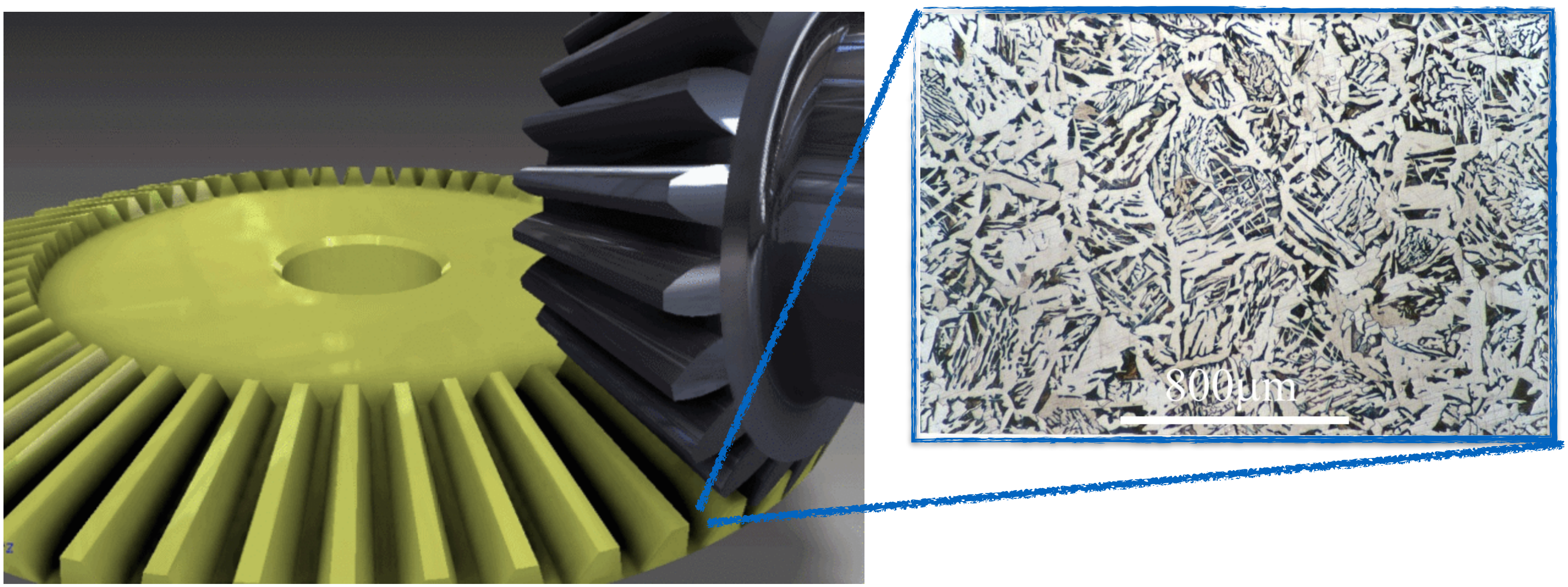
Epistemic Uncertainty Example: Ground Contamination from Fracking



The ground is not random…
but we don't really know how it looks like… unless we drill everywhere!

PREDICTIVE
SCIENCE LABORATORY

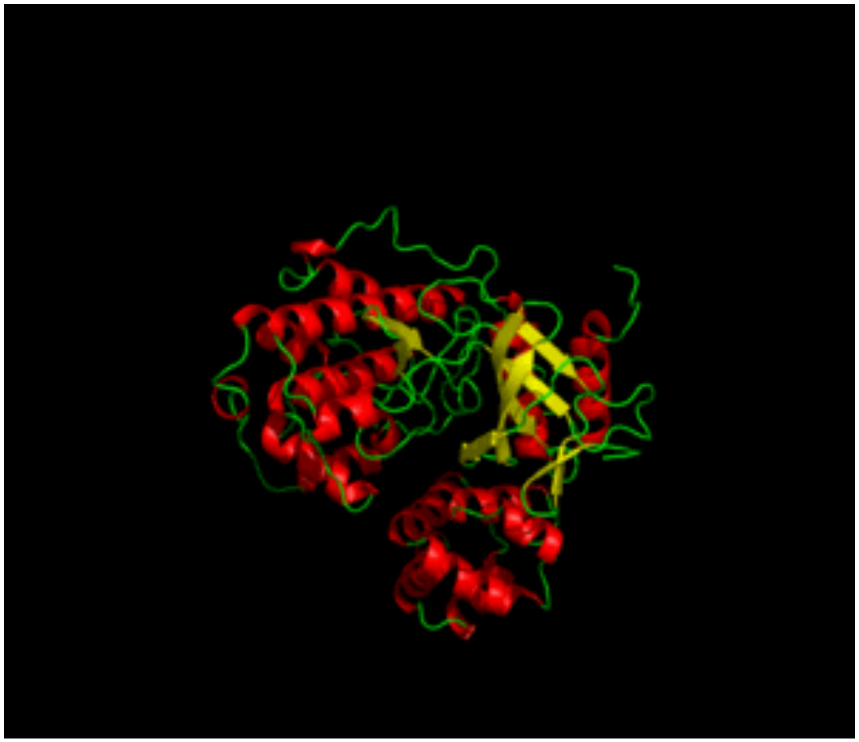
Epistemic Uncertainty Example: Microstructure of a Specific Object



Microstructure is not random, but we don't know exactly how it looks like…

PREDICTIVE
SCIENCE LABORATORY

Epistemic Uncertainty Example: Unknown Physical Law



Simulation of the interaction of two biomolecules

Statistical mechanics:

$$p(\mathbf{q}) \propto \exp\left\{-\frac{V(\mathbf{q})}{k_B T}\right\}$$

Positions of all atoms

Boltzmann constant

Temperature

Empirical potential (energy of the stem). We are not exactly sure about its form…

PREDICTIVE
SCIENCE LABORATORY

How to deal with epistemic uncertainty?

- Consider an epistemic variable ***s***.
- The uncertainty of ***s*** is described by a *probability density p(**s**).*
- But now, *p(**s**),* measures our degree of *belief* about ***s*** getting a specific value (Bayesian approach to probability).

Prior $p(s)$

$\mathcal{D}$ Bayes Rule

Posterior $p(s \mid \mathcal{D}) \propto p(\mathcal{D} \mid s) p(s)$

PREDICTIVE
SCIENCE LABORATORY

So, what is UQ?

probability theory

Models

Experiments

Predictions about the real world

and then…

Optimize engineering systems under this uncertainty!

PREDICTIVE
SCIENCE LABORATORY

References

- Prof. Paul Constantines' uq homework.
- Prof.'s Gianluca Iaccarino's lecture notes to uncertainty quantification.
- Dr. Ben Kenney's finite difference code in Python.
- Wikipedia's page on Monte Carlo.
- The guys at www.nanohub.org.
- Too many Google searches to refer to…

PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code

- You are given a computer code of extreme importance to national security.
- The code has works with two parameters:
  - n: the grid size that controls the accuracy of its approximation.
  - s: a physical parameter about which you are uncertain.
  - an expert physicist tells you that s can be anything between -1 and 1.

PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code

- The code works as follows:

```
x = solver(s, n)
```

- The result x is a vector of size n - 1. You have no idea what it means…

- An expert engineer tells you that the following quantity is of at most importance:

$$y = \max_i x_i$$

- If it gets above 1.2, we will have a catastrophic failure.

- They want you compute the probability that this happens:

$$p_{\text{fail}} = P[y > 1.2] = ?$$

Example: Extremely Important Computer Code

- In our example:

$$x = \text{solver}(s, n)$$

- Is the uncertainty in s aleatory or epistemic?
- We don't know and we don't care…
- Using probability theory, we treat all cases in the same manner.

Example: Extremely Important Computer Code

- Go to www.nanohub.org and login using your username.
- Open the "Workspace" tool and launch it.
- We will need some Python packages. Run this to load them:

```
use -e anaconda-2.3.0
```

Example: Extremely Important Computer Code

- Download the code from the svn repository:

  svn checkout https://nanohub.org/tools/mcprobf/svn/trunk mcprobf

- Change working directory: cd mcprobf
- Open your favorite editor (e.g., geany), and open the file "extremely_important_solver.py".
- Read the documentation at the very top if you like.
- Now, let's run the code for a grid size n = 11 an for randomly picked s's:

  python extremely_important_solver.py | less

PREDICTIVE
SCIENCE LABORATORY

Sample output

```
================================================================================
                             VERY IMPORTANT SOLVER
================================================================================
This program runs the solver a couple of times for
demonstration purposes.
PARAMETERS:
--------------------------------------------------------------------------------
grid size:  11
--------------------------------------------------------------------------------
> starting demo
Solver run 001
--------------------------------------------------------------------------------
input s = -0.747
output x:
[ 0.01677629  0.02771111  0.03462954  0.03868628  0.04055841  0.04055841
  0.03868628  0.03462954  0.02771111  0.01677629]
critical parameter y = max(x_i) = 0.041
Solver run 002
--------------------------------------------------------------------------------
input s = -0.435
output x:
[ 0.02089752  0.03547578  0.04525622  0.05125958  0.0541124   0.0541124
  0.05125958  0.04525622  0.03547578  0.02089752]
critical parameter y = max(x_i) = 0.054
Solver run 003
--------------------------------------------------------------------------------
input s = 0.852
output x:
[-0.04906068 -0.09957009 -0.14482012 -0.17880114 -0.19700014 -0.19700014
 -0.17880114 -0.14482012 -0.09957009 -0.04906068]
critical parameter y = max(x_i) = -0.049
Solver run 004
--------------------------------------------------------------------------------
```

```
Solver run 018
----------------------------------------------------------------------------
-----
input s = 0.680
output x:
[-0.12533044 -0.24579867 -0.34901066 -0.42434774 -0.46405908 -0.46405908
 -0.42434774 -0.34901066 -0.24579867 -0.12533044]
critical parameter y = max(x_i) = -0.125
Solver run 019
----------------------------------------------------------------------------
-----
input s = 0.387
output x:
[ 0.10083067  0.18840496  0.25828789  0.30694042  0.33189865  0.33189865
  0.30694042  0.25828789  0.18840496  0.10083067]
critical parameter y = max(x_i) = 0.332
    *** CATASTROPHIC FAILLURE ***

Solver run 020
----------------------------------------------------------------------------
-----
input s = 0.011
output x:
[ 0.03456338  0.06146756  0.08121665  0.09418066  0.10060251  0.10060251
  0.09418066  0.08121665  0.06146756  0.03456338]
critical parameter y = max(x_i) = 0.101
```

Example: Extremely Important Computer Code

- Now change the value of “n” at line 56 to “10001”.
- The solver becomes more accurate, but also slower.
- Do you see any “CATASTROPHIC FAILURE” event?

Example: Extremely Important Computer Code

- In our example:

$$x = \text{solver}(s, n)$$

- What causes the “error” and what the “uncertainty”?

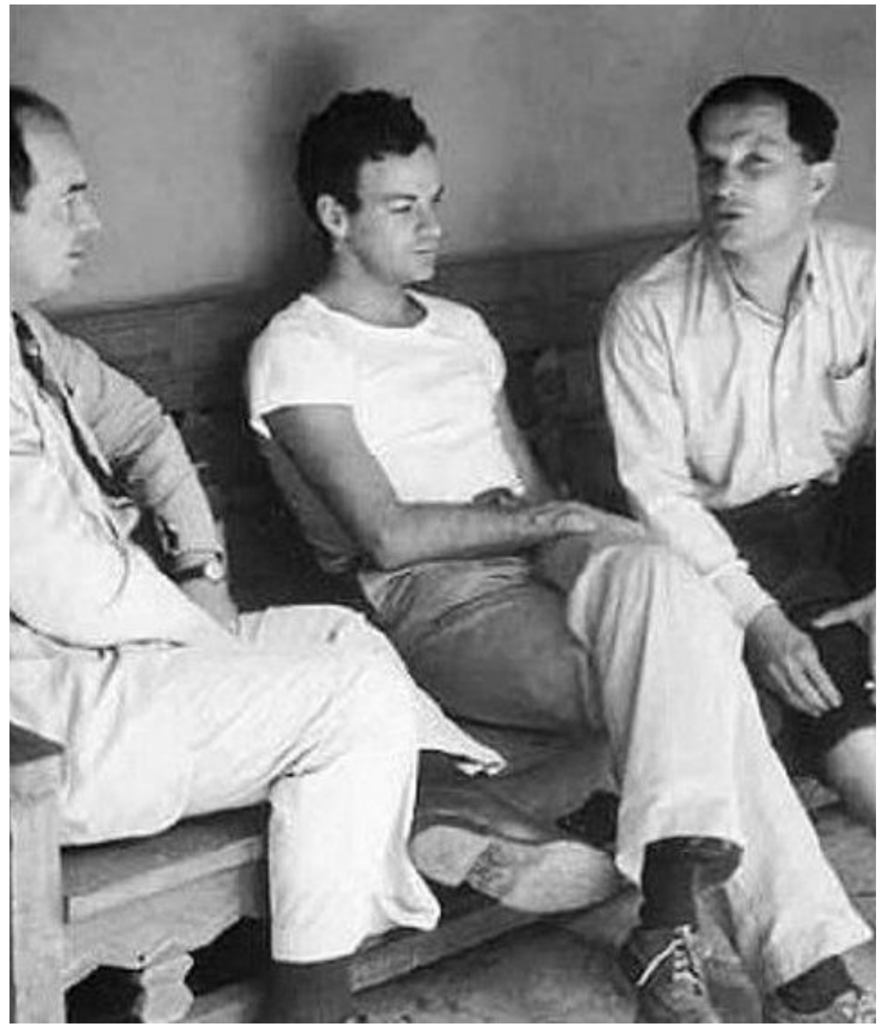
Monte Carlo Simulations

- When we are doing UQ, we usually have to compute expectations of the form:

$$\mathbb{E}[f(s)] = \int f(s)p(s)ds.$$

- When $\boldsymbol{s}$ is high-dimensional, this is a difficult problem.

- Monte Carlo popularized during WWII at Los Alamos mainly by Stanislaw Ulam and John von Neumann.



Von Neumann, Feynman, Ulam

PREDICTIVE
SCIENCE LABORATORY

Monte Carlo Simulations

- We want to compute:

$$I := \mathbb{E}[f(s)] = \int f(s)p(s)ds.$$

- The idea is simple:
  - Sample the random parameter repeatedly:

$$s_1, \ldots, s_N \sim p(s)$$

  - Use the *empirical average* to approximate the expectation:

$$I \approx \hat{I}_N = \frac{f(s_1) + \ldots + f(s_N)}{N} = \frac{1}{N}\sum_{i=1}^{N} f(s_i)$$

PREDICTIVE
SCIENCE LABORATORY

Monte Carlo Simulations

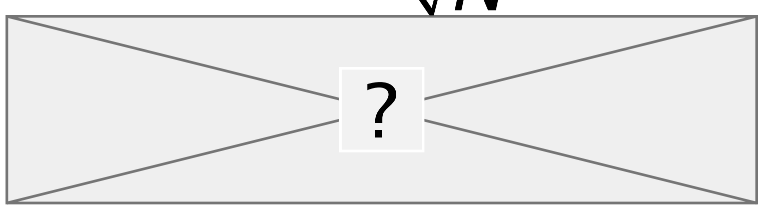
- Wrapping it up:

$$I \approx \hat{I}_N = \frac{f(s_1)+\ldots+f(s_N)}{N} = \frac{1}{N}\sum_{i=1}^{N} f(s_i)$$

- The fact at the estimate converges to the true value is known as the law of large numbers.

- Using the central limit theorem, it is also possible to get estimates of the **error** of the MC estimator:

$$\delta I_N = \frac{\hat{\sigma}_N}{\sqrt{N}},$$



PREDICTIVE
SCIENCE LABORATORY

In our example…

We want to compute the probability of failure:

$$p_{\text{fail}} = P[y > 1.2] = ?$$

Can we express it as an expectation so that we can use Monte Carlo?

PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code

- In our example:

$$f(s) = \chi_{\{y(s;n)>\alpha\}}(s) := \begin{cases} 1, & \text{if } y(s;n) > \alpha, \\ 0, & \text{otherwise.} \end{cases}$$

- since we have:

$$\mathbb{E}[f(s)] := \int \chi_{\{y(s;n)>\alpha\}}(s)p(s)ds := P[y > \alpha] = p_{\text{fail}}.$$

- and it is a simple algebra exercise to develop the estimators

$$p_{\text{fail}} \approx \hat{p}_N \pm 1.96\hat{\sigma}_N \quad \hat{p}_N = \frac{1}{N}\sum_{i=1}^{N}\chi(s_i) \quad \hat{\sigma}_N = \frac{\hat{p}_N(1-\hat{p}_N)}{\sqrt{N}}$$

PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code

- Go back to www.nanohub.org and use geany to open the file "example.py". Read the documentation.

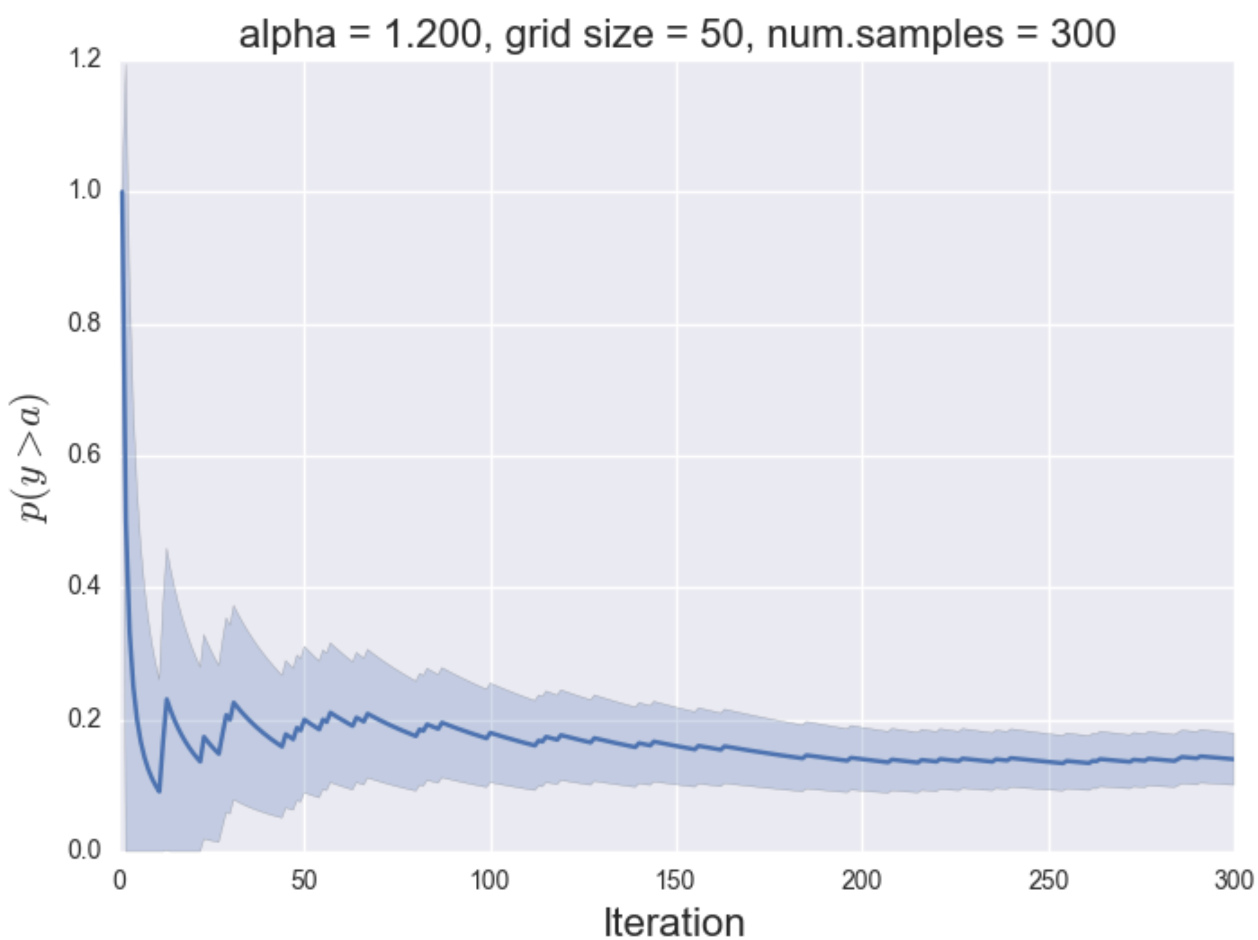
- Skim through the code to see the implementation.

- Run the code with (you'll have to press enter):

```
python example.py
```

PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code



PREDICTIVE
SCIENCE LABORATORY

Example: Extremely Important Computer Code

- Try changing the grid size “n” at line 48 to “2”. What do you observe?
- Try changing the grid size “n” at line 48 to 100. What do you observe (Ctrl-C to exit)?
- Would you trust the results of our computation?
- Repeat the same exercise for different alphas (line 45).

PREDICTIVE
SCIENCE LABORATORY

Thank you!