Gaussian processes for decision making under uncertainty

Javier González
Amazon Cambridge, UK

Oxford, UK

November 2018
Uncertainty quantification

*UQ is the science of quantitative characterization and reduction of uncertainties in both computational and real world applications (Wikipedia).*

- how to characterize uncertainty?
- how to reduce and *use* uncertainty?
In essence...

We just need to identify ‘where’ we are ignorant and act on it!
Simulations

Simulators are great but:

- Are often slow and expensive to run.
- Can only simulate just what it has been programmed to simulate.
- Simulators are black boxes hard to interpret.
Basic idea of surrogate modelling/emulation
[O'Hagan 2013; O' Hagan, 2006; Conti and O'Hagan, 2010]

Replace (or complement) the simulator with and emulator.

Emulator: probabilistic model fitted on simulation runs.

- Predictions are inexpensive.
- Predictions come with a level of uncertainty (GP emulators).

An emulator is a 'model of a model'
Need to quantify all sources of uncertainty

UQ deals with the end-to-end study of the impact of all forms of error and uncertainty in the models that we use to analyse or build a system of interest.
Decisions under uncertainty

**Statistical inference:**

\[ \text{model} + \text{data} \rightarrow \text{prediction} \]

- Many ways to do this: we focus on Gaussian processes.
- Semi-mechanistic models are key in practice.
- Machine learning promises automatic decision making.

**Decision making:**

\[ \text{predictions} \rightarrow \text{decisions} \]

- The models we use need to tell us when they don’t know.
- We need calibrated uncertainties in decision making.
Decisions under uncertainty

**Inference**

- **Things that I know:** 
  \[ y \]

- **Things that I don’t know:** 
  \[ y^* \]

- **Description of the world:** 
  \[ p(y^*, y) \]

- **What I need:** 
  \[ p(y^* | y) \]

**Decisions**

- **Actions I can take:** 
  \[ a \in A \]

- **Reward I gain:** 
  \[ R(a | y, y^*) \]

- **‘Optimal’ decision:** 
  \[ a^* = \arg \max_{a \in A} \alpha(a; R, p) \]

Example:

\[ \alpha(a; R, p) = \mathbb{E}_p R(a | y, y^*) \]
Decisions under uncertainty

**Inference**

- Things that I know: \( y \)
- Things that I don’t know: \( y^* \)
- Description of the world: \( p(y^*, y) \)
- What I need: \( p(y^* | y) \)

**Decisions**

- Actions I can take: \( a \in \mathcal{A} \)
- Reward I gain: \( R(a | y, y^*) \)
- ‘Optimal’ decision:
  \[
  a^* = \arg \max_{a \in \mathcal{A}} \alpha(a; R, p)
  \]
  Example:
  \[
  \alpha(a; R, p) = \mathbb{E}_p R(a | y, y^*)
  \]
Outer loop applications

Situations where a probabilistic model can be used in decision making
Decisions problems associated to outer loop applications

In origin are 'deterministic problems'

- **Optimization:**
  \[ x^* = \arg \min_{x} f(x). \]

- **Quadrature:**
  \[ Z = \int_{\mathcal{X}} f(x)p(x)dx. \]

- **Active learning/Experimental design.**
  \[ \{x_1^*, \ldots, x_n^*\} = \arg \min_{\mathcal{X}^n} \| f - \hat{f}_{\{x_1, \ldots, x_n\}} \|_H. \]

- **Control/ Reinforcement learning.**
  \[ \min \mathbb{E} \left[ \int_{0}^{T} g(t, X^u_t, u_t)dt \right] + G(X^u_T) \]
Decisions problems associated to outer loop applications

In origin are 'deterministic problems'

▶ Optimization:

\[ x^* = \arg \min_{\mathcal{X}} f(x). \]

▶ Quadrature:

\[ Z = \int_{\mathcal{X}} f(x)p(x)dx. \]

▶ Active learning/Experimental design.

\[ \{x_1^*, \ldots, x_n^*\} = \arg \min_{\mathcal{X}^n} \| f - \hat{f}_{\{x_1, \ldots, x_n\}} \|_H. \]

▶ Control/ Reinforcement learning.

\[ \min \mathbb{E} \left[ \int_0^T g(t, X_t^u, u_t)dt \right] + G(X_T^u) \]
All these problems can be solved under a common sequential decision making framework:

- Use some form of belief of the environment (model of $f$).
- Make sequential decisions using some form of reward $\alpha(a; R, p(f))$.
- Decisions influence rewards.
- Described as ‘Exploration/Exploitation’ problems.
Model of $f$: Gaussian process emulators

[Rasmussen and Williams, 2006]

\[
Y = f(X) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2 I)
\]

\[
f(X) \sim \mathcal{GP}(0, k(X, X'))
\]

- Multivariate Gaussian under linear restrictions.
- Posterior mean and variance have closed form.
- Semi-mechanistic: prior in the covariance function.
Model of $f$: Deep Gaussian processes

[Damianou and Lawrence, 2013]

- Handle non linearities.
- Uncertainty propagation across complex pipelines.
- Intractable, requires approximations.
Multi-fidelity emulators

- We have access to other lower fidelities that we can sort.
- Costs are also available.
Model of $f$: Multi-fidelity Gaussian process

Single fidelity vs. multiple fidelities
Emulators with multi-fidelity outputs
[OHagan, 2000], [Perdikaris et al. 2017], [Forrester, 2007]

**Linear multi-fidelity:**

\[ f_i (x) = \rho f_{i-1} (x) + \delta_d (x) \]

- \( f_{d-1} (x) \): low fidelity simulation of the problem of interest.
- \( f_d (x) \): higher fidelity simulation.
- \( \delta_d (x) \): additive difference between the lower and higher fid.

\[ f_0 (x) \text{ and } \{ \delta_d (x) \}_{d=1}^{D} \text{ are all Gaussian processes.} \]

**Extensions:**

\[ f_i (x) = \rho(x) f_{i-1} (x) + \delta_i (x) \]

\[ f_i (x) = g_i (f_{i-1} (x)) + \delta_i (x), \]
Toy example

[Perdikaris et al. 2017]
Black-box optimization

Consider a ‘well behaved’ function $f : \mathcal{X} \to \mathbb{R}$ where $\mathcal{X} \subseteq \mathbb{R}^D$ is a bounded domain.

$$x_M = \arg \min_{x \in \mathcal{X}} f(x).$$

- $f$ is explicitly unknown and multi-modal.
- Evaluations of $f$ are expensive.

**Applications:**

- Robotics, control, reinforcement learning.
- Model calibration.
- Compilers, hardware, software, industrial design.
- Intractable likelihoods.
Surrogate models in optimization problems

[Jones et al., 1998]

Build an acquisition function to collect more data.

\[ \alpha_{EI}(x; \theta, D) = \int_y \max(0, y_{\text{best}} - y) p(y|x; \theta, D) dy \]

Exploration/Exploitation
Surrogate models in optimization problems

[Jones et al., 1998]

Build an acquisition function to collect more data.

\[
\alpha_{EI}(x; \theta, \mathcal{D}) = \int_y \max(0, y_{\text{best}} - y) p(y|x; \theta, \mathcal{D}) dy
\]

Exploration/Exploitation
Surrogate models in optimization problems
[Jones et al., 1998]

Build an acquisition function to collect more data.

\[ \alpha_{EI}(\mathbf{x}; \theta, \mathcal{D}) = \int_y \max(0, y_{\text{best}} - y) p(y|x; \theta, \mathcal{D}) \, dy \]

Exploration/Exploitation
Surrogate models in optimization problems
[Jones et al., 1998]

Build an acquisition function to collect more data.

\[ \alpha_{EI}(x; \theta, D) = \int_y \max(0, y_{best} - y) p(y|x; \theta, D) dy \]

Exploration/Exploitation
Surrogate models in optimization problems
[Jones et al., 1998]

Build an acquisition function to collect more data.

\[ \alpha_{EI}(\boldsymbol{x}; \theta, \mathcal{D}) = \int_y \max(0, y_{best} - y) p(y|\boldsymbol{x}; \theta, \mathcal{D}) dy \]

Exploration/Exploitation
Surrogate models in optimization problems

[Jones et al., 1998]

Build an acquisition function to collect more data.

\[ \alpha_{EI}(x; \theta, D) = \int_y \max(0, y_{\text{best}} - y) p(y|x; \theta, D) dy \]

Exploration/Exploitation
Multi-fidelity experimental design
Recipe to use multi-fidelity to optimize a system

1. Collect a few low fidelity and high fidelity observations.

2. Build a multi-fidelity emulator of the system.

3. Improve the model with more observations: Max. information gain per unit of cost.

4. Optimize the emulator instead of the original system.
Cost per simulation: 1u.

Coste per experiment: 5u.
Example multi-fidelity experimental design
Example multi-fidelity experimental design
Example multi-fidelity experimental design

Experiments and simulations - Joint model

Variance reduction for a new sample
Example multi-fidelity experimental design

Experiments and simulations - Joint model

Variance reduction for a new sample
Example multi-fidelity experimental design

Experiments and simulations - Joint model

$f(x)$

Experiments
f-experiment
Simulations

Variance reduction for a new sample

Var. reduccion experiment
Var. reduccion simulation
Example multi-fidelity experimental design
Example multi-fidelity experimental design
Example multi-fidelity experimental design
Conclusions

- Big room for uncertainty quantification in climate science.

- Semi-mechanistic models + good calibrations of the uncertainty are key to build good decisions systems.

- GPs are a fundamental tool for emulation but there are others.

- Climate sciences is about ‘big data’ but thinking on what we can do with ‘small data’ can help to drive new directions.