Imperial College London

# Bayesian Machine Learning for Controlling Autonomous Systems

**Marc Deisenroth**
Department of Computing
Imperial College London

Department of Computer Science
TU Darmstadt

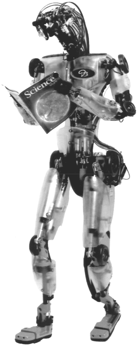
http://wp.doc.ic.ac.uk/sml/
m.deisenroth@imperial.ac.uk

Talk at University of Sheffield
January 17, 2014

# Motivation

- Three key challenges in autonomous systems:

**Modeling. Predicting. Decision making.**



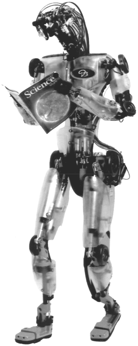

**Robotics**

# Motivation

- Three key challenges in autonomous systems:

  **Modeling. Predicting. Decision making.**

- Data-efficient learning
- Noisy signals and processes



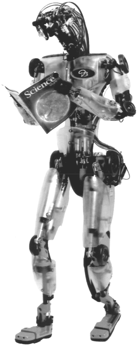

**Robotics**

# Motivation

- Three key challenges in autonomous systems:

  **Modeling. Predicting. Decision making.**

- Data-efficient learning
- Noisy signals and processes




**Robotics**

Increase autonomy: deal with uncertainty

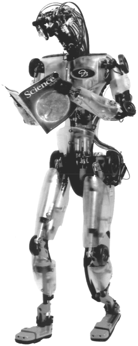
▸▸ **Bayesian machine learning**

# Motivation

- Three key challenges in autonomous systems:

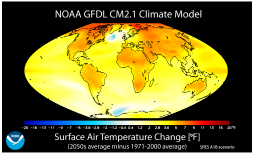
  **Modeling. Predicting. Decision making.**

- Data-efficient learning
- Noisy signals and processes



**Robotics**

Increase autonomy: deal with uncertainty
▸▸ **Bayesian machine learning**



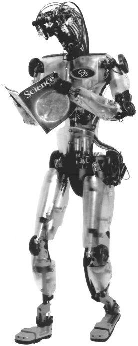

Climate Science

# Motivation

- Three key challenges in autonomous systems:

  **Modeling. Predicting. Decision making.**

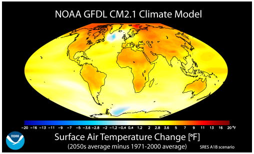
- Data-efficient learning
- Noisy signals and processes



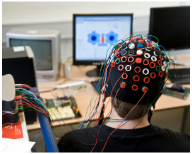

**Robotics**

Increase autonomy: deal with uncertainty

▶▶ **Bayesian machine learning**




Climate Science



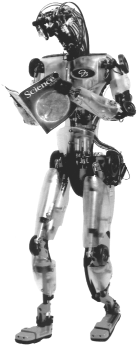
Brain-Computer Interface

# Motivation

- Three key challenges in autonomous systems:

  **Modeling. Predicting. Decision making.**

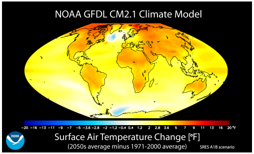
- Data-efficient learning
- Noisy signals and processes



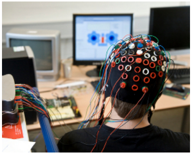

Robotics

Increase autonomy: deal with uncertainty

▶▶ **Bayesian machine learning**



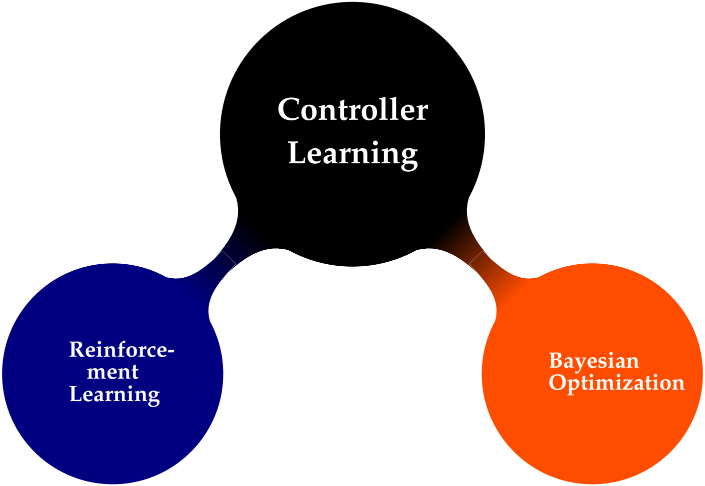

Climate Science



Brain-Computer Interface

Prosthetics

# Outline


Controller Learning
Reinforcement Learning
Bayesian Optimization

# Reinforcement Learning Set-up

$$\boxed{\boldsymbol{x}_{t+1}} = f(\boxed{\boldsymbol{x}_t}, \boxed{\boldsymbol{u}_t}) + \boldsymbol{w}\,, \qquad \boxed{\boldsymbol{u}_t} = \boxed{\pi}(\boxed{\boldsymbol{x}_t}, \boxed{\boldsymbol{\theta}})$$

State Control Policy Policy parameters

# Reinforcement Learning Set-up

$$\boxed{\boldsymbol{x}_{t+1}} = f(\boxed{\boldsymbol{x}_t}, \boxed{\boldsymbol{u}_t}) + \boldsymbol{w}\,, \qquad \boxed{\boldsymbol{u}_t} = \boxed{\pi}(\boxed{\boldsymbol{x}_t}, \boxed{\boldsymbol{\theta}})$$

State Control Policy Policy parameters

## Objective

Find policy parameters $\boldsymbol{\theta}^*$ that minimize the expected long-term cost

$$J(\boldsymbol{\theta}) = \sum_{t=1}^{T} \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]\,, \qquad p(\boldsymbol{x}_0) = \mathcal{N}\big(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0\big)\,.$$

Instantaneous cost $c(\boldsymbol{x}_t)$, e.g., $\|\boldsymbol{x}_t - \boldsymbol{x}_{\text{target}}\|^2$

- Typical objective in optimal control and reinforcement learning (Bertsekas, 2005; Sutton & Barto, 1998)

# Model-based Policy Search

## Objective

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \dots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. Policy improvement

---

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. Policy improvement
4. Apply controller

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

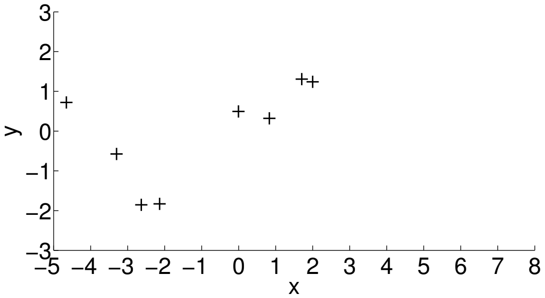
**High-Level Steps:**

1. **Probabilistic model for transition function $f$ to be robust to model errors**
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. Policy improvement
4. Apply controller

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model Learning

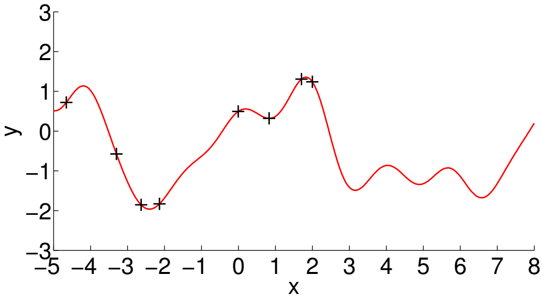
Model learning problem: Find a function $f : x \mapsto f(x) = y$




Observed function values

# Model Learning

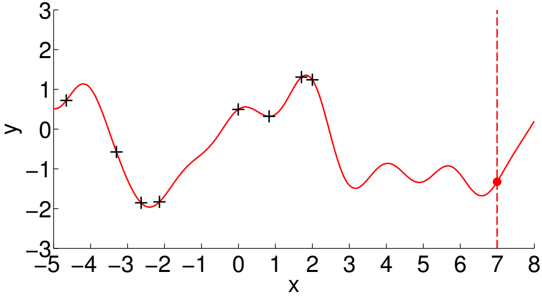
Model learning problem: Find a function $f : x \mapsto f(x) = y$




Plausible function approximators

# Model Learning

Model learning problem: Find a function $f : x \mapsto f(x) = y$

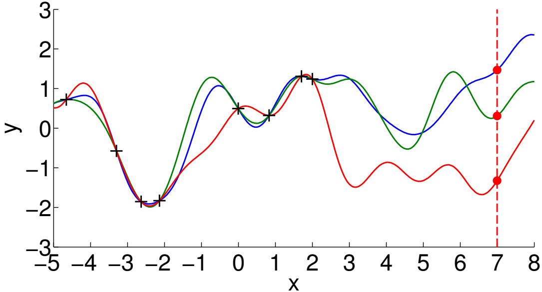



Plausible function approximators

**Predictions? Decision Making?**

# Model Learning

Model learning problem: Find a function $f : x \mapsto f(x) = y$

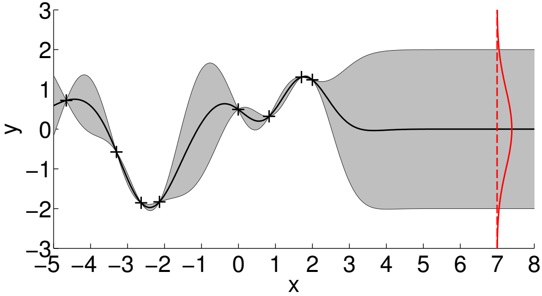



Plausible function approximators

**Predictions? Decision Making? Model Errors!**

# Model Learning

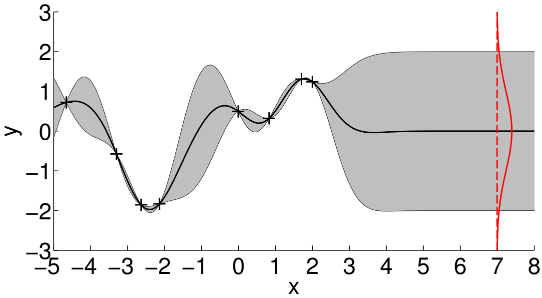
Model learning problem: Find a function $f : x \mapsto f(x) = y$




Distribution over plausible functions

# Model Learning

Model learning problem: Find a function $f : x \mapsto f(x) = y$




Distribution over plausible functions

- Express uncertainty about the underlying function
- Gaussian process for model learning (Rasmussen & Williams, 2006)

# Introduction to Gaussian Processes

- State-of-the-art nonparametric Bayesian regression method
- Probability distribution over functions
- Fully specified by
  - Mean function $m$ (average function)
  - Covariance function $k$ (assumptions on structure)

$$\mathrm{Cov}[f(\boldsymbol{x}_p), f(\boldsymbol{x}_q)] \quad ⏩ \quad k(\boldsymbol{x}_p, \boldsymbol{x}_q)$$

# Introduction to Gaussian Processes

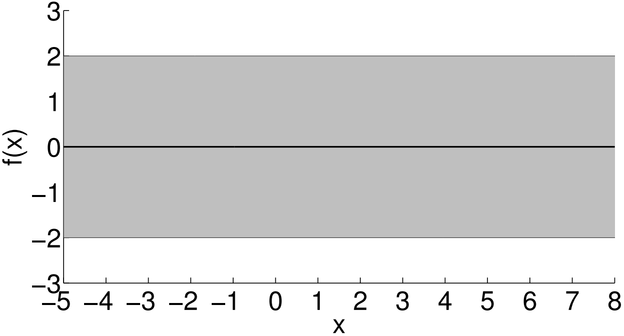
- State-of-the-art nonparametric Bayesian regression method
- Probability distribution over functions
- Fully specified by
  - Mean function $m$ (average function)
  - Covariance function $k$ (assumptions on structure)

$$\mathrm{Cov}[f(\boldsymbol{x}_p), f(\boldsymbol{x}_q)] \quad \blacktriangleright\!\blacktriangleright \quad k(\boldsymbol{x}_p, \boldsymbol{x}_q)$$

- Posterior predictive distribution at $\boldsymbol{x}_*$ is Gaussian:

$$p(f(\boldsymbol{x}_*)|\underbrace{\boldsymbol{x}_*}_{\text{Test input}}, \underbrace{\boldsymbol{X}, \boldsymbol{y}}_{\text{Training data}}) = \mathcal{N}\big(f(\boldsymbol{x}_*)\,|\,m(\boldsymbol{x}_*), \sigma^2(\boldsymbol{x}_*)\big)$$

# Intuitive Introduction to Gaussian Processes

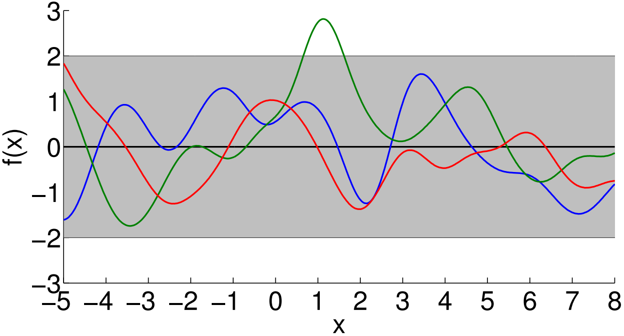



Prior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\varnothing] = m(\boldsymbol{x}_*) = 0$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\varnothing] = \sigma^2(\boldsymbol{x}_*) = \mathrm{Cov}[f(\boldsymbol{x}_*), f(\boldsymbol{x}_*)] = k(\boldsymbol{x}_*, \boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

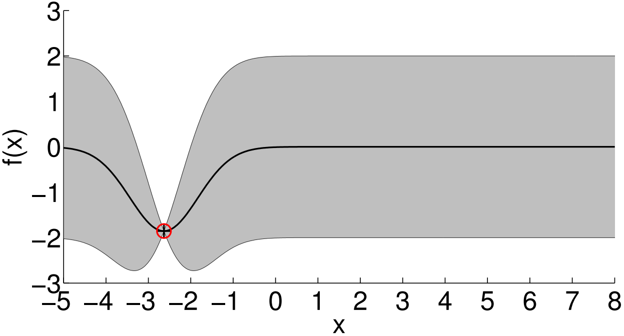



Prior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\varnothing] = m(\boldsymbol{x}_*) = 0$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\varnothing] = \sigma^2(\boldsymbol{x}_*) = \mathrm{Cov}[f(\boldsymbol{x}_*), f(\boldsymbol{x}_*)] = k(\boldsymbol{x}_*, \boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

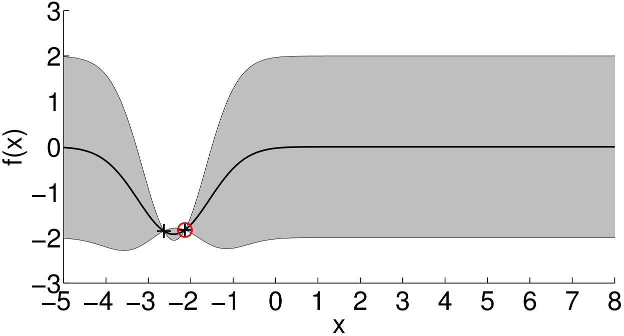



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

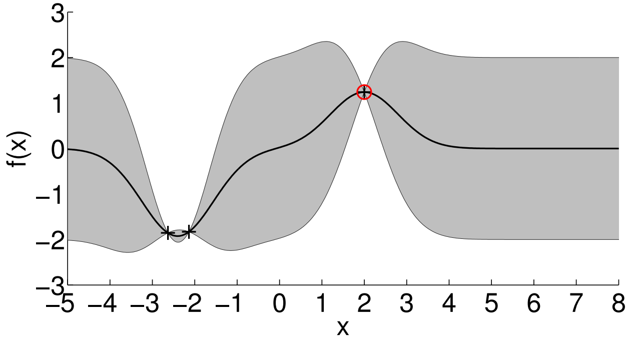



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

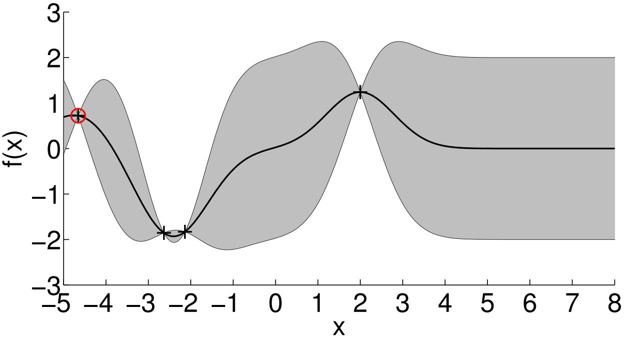



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

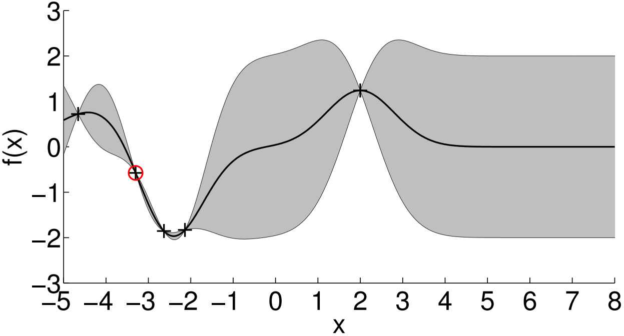



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

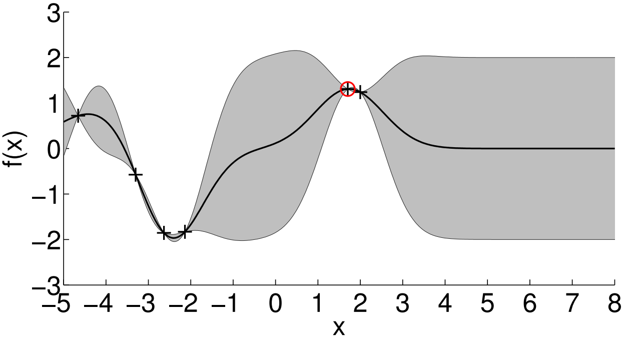



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

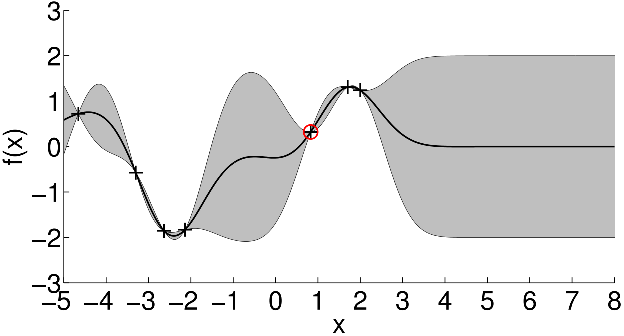



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

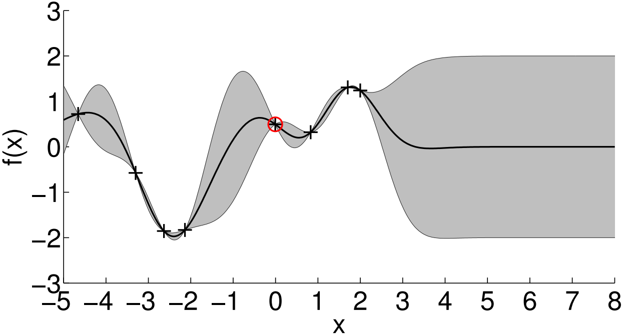



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes

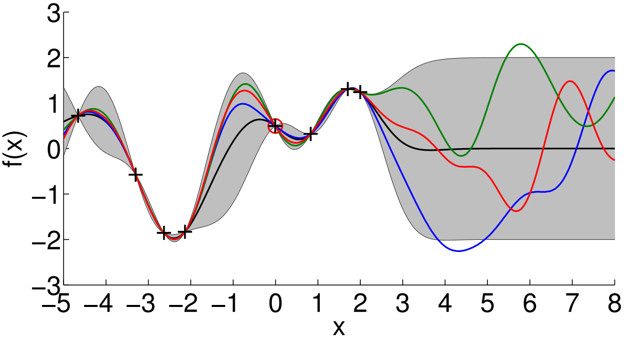



Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Intuitive Introduction to Gaussian Processes




Posterior belief about the function

Predictive (marginal) mean and variance:

$$\mathbb{E}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = m(\boldsymbol{x}_*) = k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}\boldsymbol{y}$$
$$\mathbb{V}[f(\boldsymbol{x}_*)|\boldsymbol{X},\boldsymbol{y}] = \sigma^2(\boldsymbol{x}_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - k(\boldsymbol{X},\boldsymbol{x}_*)^\top k(\boldsymbol{X},\boldsymbol{X})^{-1}k(\boldsymbol{X},\boldsymbol{x}_*)$$

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

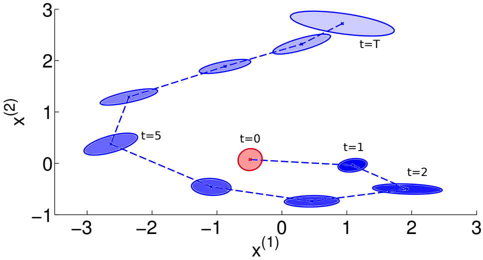
**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors
2. **Compute long-term predictions** $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \dots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. Policy improvement
4. Apply controller

---

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

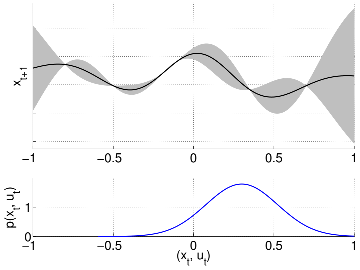
# Long-Term Predictions




- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Long-Term Predictions

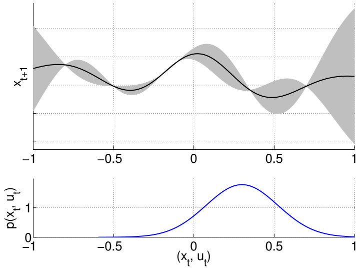



- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

$$\underbrace{p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{u}_t)}_{\text{GP prediction}} \; \underbrace{p(\boldsymbol{x}_t, \boldsymbol{u}_t|\boldsymbol{\theta})}_{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})}$$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Long-Term Predictions

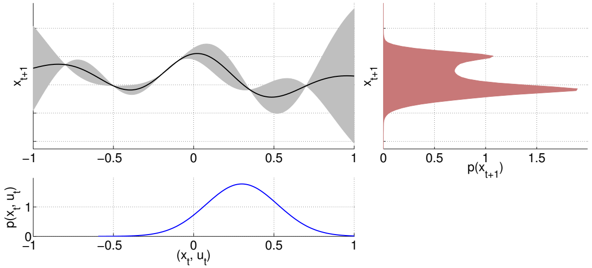



- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

$$\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{\theta})} = \iiint \underbrace{\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{u}_t)}}_{\text{GP prediction}} \underbrace{\boxed{p(\boldsymbol{x}_t, \boldsymbol{u}_t|\boldsymbol{\theta})}}_{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})} \, \mathrm{d}f \, \mathrm{d}\boldsymbol{x}_t \, \mathrm{d}\boldsymbol{u}_t$$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Long-Term Predictions

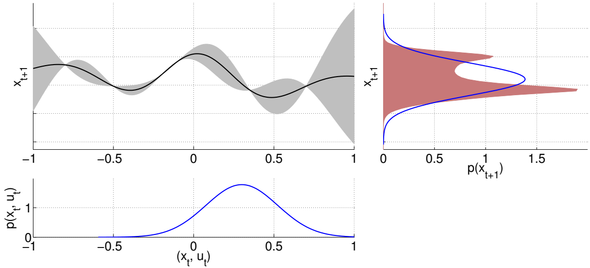



- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

$$\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{\theta})} = \iiint \underbrace{\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{u}_t)}}_{\text{GP prediction}} \underbrace{\boxed{p(\boldsymbol{x}_t, \boldsymbol{u}_t|\boldsymbol{\theta})}}_{\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})} \, \mathrm{d}f \, \mathrm{d}\boldsymbol{x}_t \, \mathrm{d}\boldsymbol{u}_t$$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Long-Term Predictions




- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

$$\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{\theta})} = \iiint \underbrace{\boxed{p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{u}_t)}}_{\text{GP prediction}} \underbrace{\boxed{p(\boldsymbol{x}_t, \boldsymbol{u}_t|\boldsymbol{\theta})}}_{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})} \, \mathrm{d}f \, \mathrm{d}\boldsymbol{x}_t \, \mathrm{d}\boldsymbol{u}_t$$

▸▸ Moment matching (Quiñonero-Candela et al., 2003)

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Model-based Policy Search

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

**High-Level Steps:**

1. Probabilistic model for transition function $f$ to be robust to model errors
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. **Policy improvement**
   - Compute expected long-term cost $J(\boldsymbol{\theta})$
   - Find parameters $\boldsymbol{\theta}$ that minimize $J(\boldsymbol{\theta})$
4. Apply controller

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Policy Improvement

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

- Know how to predict $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Policy Improvement

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

- Know how to predict $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
- Compute

$$\mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}] = \int c(\boldsymbol{x}_t)\mathcal{N}\big(\boldsymbol{x}_t \,|\, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t\big)\mathrm{d}\boldsymbol{x}_t \,, \quad t = 1, \ldots, T \,,$$

and sum them up to obtain $J(\boldsymbol{\theta})$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Policy Improvement

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

- Know how to predict $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
- Compute

$$\mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}] = \int c(\boldsymbol{x}_t)\mathcal{N}\big(\boldsymbol{x}_t \,|\, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t\big)\mathrm{d}\boldsymbol{x}_t\,, \quad t = 1, \ldots, T\,,$$

  and sum them up to obtain $J(\boldsymbol{\theta})$
- Analytically compute gradient $\mathrm{d}J(\boldsymbol{\theta})/\mathrm{d}\boldsymbol{\theta}$
- Standard gradient-based optimizer (e.g., BFGS) to find $\boldsymbol{\theta}^*$

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

# Policy Improvement

**Objective**

Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

- Know how to predict $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \dots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
- Compute

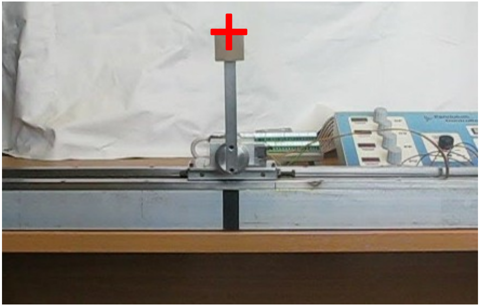
$$\mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}] = \int c(\boldsymbol{x}_t)\mathcal{N}\left(\boldsymbol{x}_t \,|\, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t\right)\mathrm{d}\boldsymbol{x}_t\,, \quad t = 1, \dots, T\,,$$

  and sum them up to obtain $J(\boldsymbol{\theta})$
- Analytically compute gradient $\mathrm{d}J(\boldsymbol{\theta})/\mathrm{d}\boldsymbol{\theta}$
- Standard gradient-based optimizer (e.g., BFGS) to find $\boldsymbol{\theta}^*$

▶▶ PILCO framework for controller learning

Deisenroth et al. (IEEE-TPAMI, 2014): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

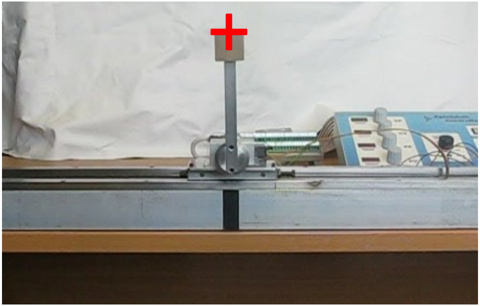
# Standard Benchmark Problem: Cart-Pole Swing-up

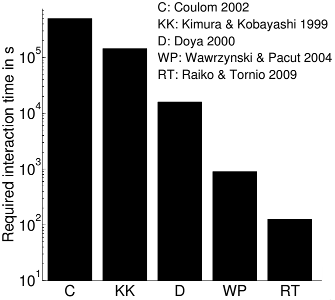


- Swing up and balance a freely swinging pendulum on a cart
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$
- Code available at http://mloss.org/software/view/508/

Deisenroth & Rasmussen (ICML, 2011): *PILCO: A Model-based and Data-efficient Approach to Policy Search*

# Standard Benchmark Problem: Cart-Pole Swing-up

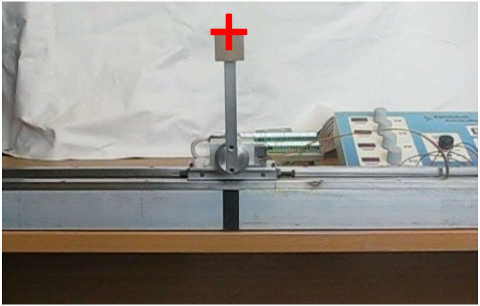


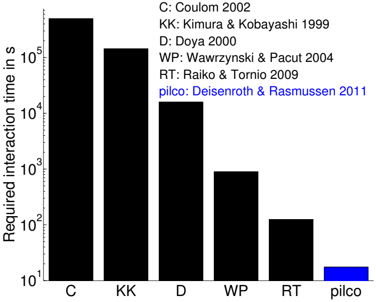



- Swing up and balance a freely swinging pendulum on a cart
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$
- Code available at http://mloss.org/software/view/508/

Deisenroth & Rasmussen (ICML, 2011): *PILCO: A Model-based and Data-efficient Approach to Policy Search*

# Standard Benchmark Problem: Cart-Pole Swing-up

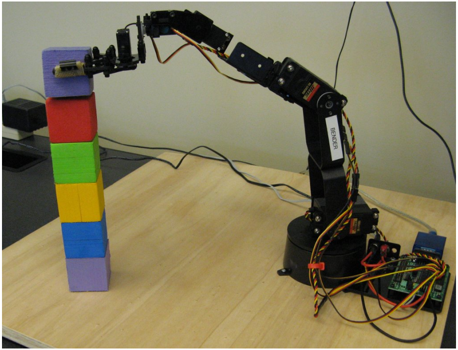





- Swing up and balance a freely swinging pendulum on a cart
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$
- **Unprecedented learning speed** compared to state-of-the-art
- Code available at http://mloss.org/software/view/508/

Deisenroth & Rasmussen (ICML, 2011): *PILCO: A Model-based and Data-efficient Approach to Policy Search*

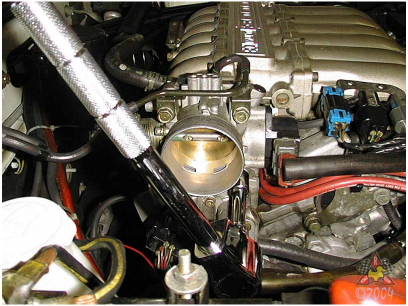
# Learning to Control an Off-the-Shelf Robot




- Autonomously learn block-stacking with a low-cost robot
- Robot very noisy
- Learn forward model and controller from scratch

Deisenroth et al. (RSS, 2011): *Learning to Control a Low-Cost Manipulator using Data-efficient Reinforcement Learning*

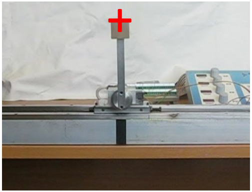
# Controlling Throttle Valves in Combustion Engines

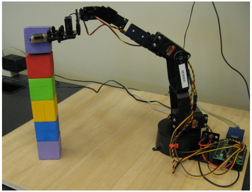


Bischoff et al., ECML 2013

▶▶ More videos at `http://www.youtube.com/user/PilcoLearner`

# Summary (1)

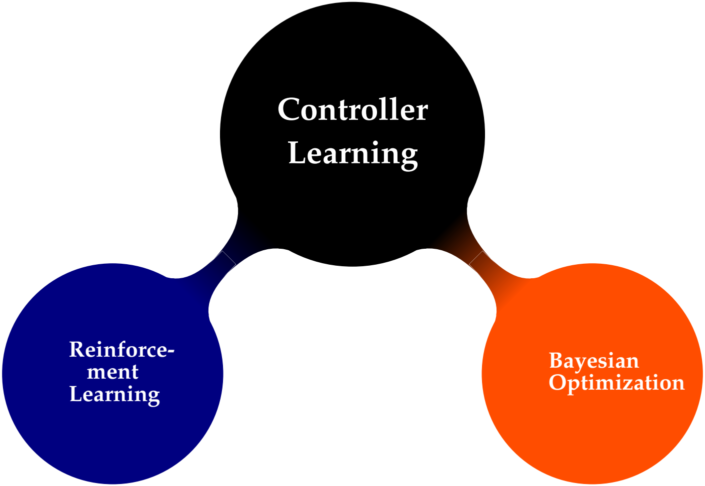




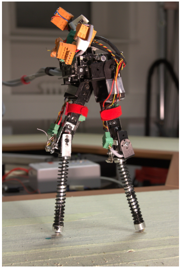
## Practical Framework for Autonomous Learning

- Key: Explicit incorporation of model uncertainty into long-term predictions and decision making
- Applied to real systems

# Outline


Controller Learning
Reinforce-ment Learning
Bayesian Optimization

# Bayesian Optimization for Learning Controllers

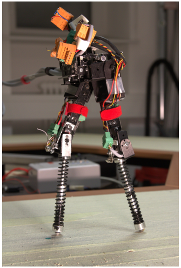


- Learning forward models is not always easy
- Legged locomotion: ground contacts

**Objective**

Find parameters $\boldsymbol{\theta}$ of controller $\pi(\boldsymbol{\theta})$

# Bayesian Optimization for Learning Controllers



- Learning forward models is not always easy
- Legged locomotion: ground contacts

**Objective**

Find parameters $\boldsymbol{\theta}$ of controller $\pi(\boldsymbol{\theta})$

**Challenges:**

- No forward model
- No analytic cost function, no demonstrations
- Still need to be data efficient (fragile robot)
- Manual parameter search can be tedious

▶▶ Bayesian optimization (e.g., Jones 1998; Osborne et al., 2009)

# Bayesian Optimization

**Objective**

Minimize an objective function $g$, which is very expensive to evaluate

# Bayesian Optimization

**Objective**

Minimize an objective function $g$, which is very expensive to evaluate

Key Idea:

1. Build a model $\tilde{g}$ of the objective function
2. Find $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \tilde{g}(\boldsymbol{\theta})$
3. Evaluate true objective $g$ at $\boldsymbol{\theta}^*$
4. Update the model $\tilde{g}$

# Bayesian Optimization

**Objective**

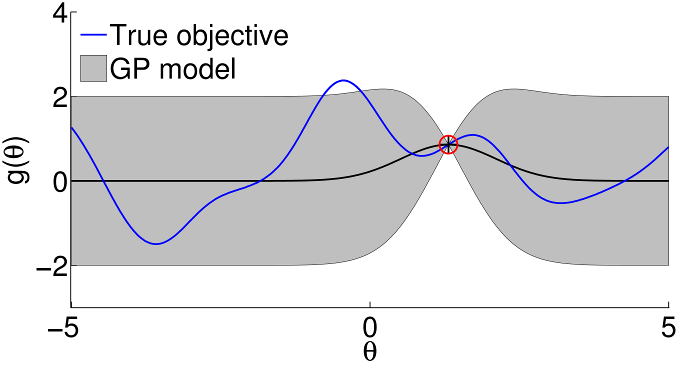
Minimize an objective function $g$, which is very expensive to evaluate

Key Idea:

1. Build a model $\tilde{g}$ of the objective function
2. Find $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \tilde{g}(\boldsymbol{\theta})$
3. Evaluate true objective $g$ at $\boldsymbol{\theta}^*$
4. Update the model $\tilde{g}$

- Standard model $\tilde{g}$ is a Gaussian process
- Standard assumption:
  Computations are cheap compared to evaluating true objective $g$

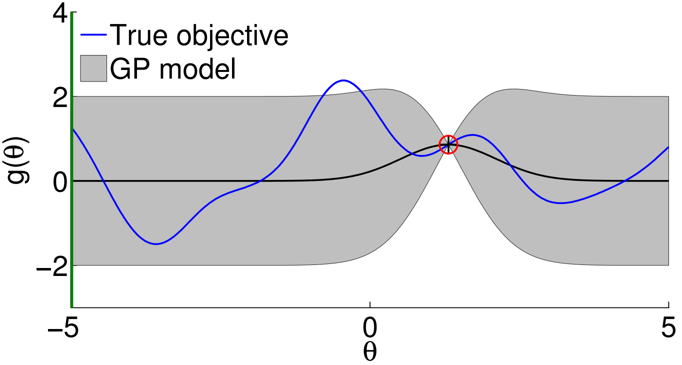
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

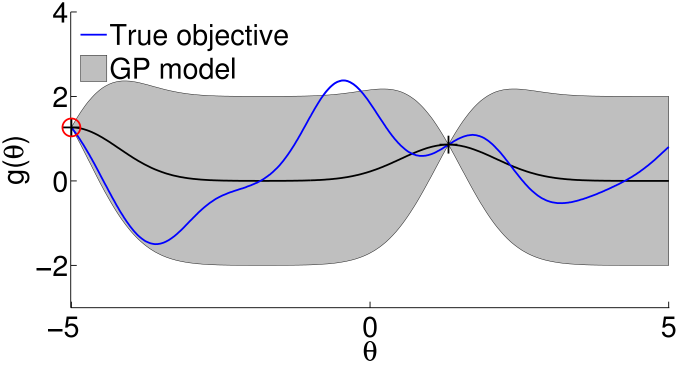
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

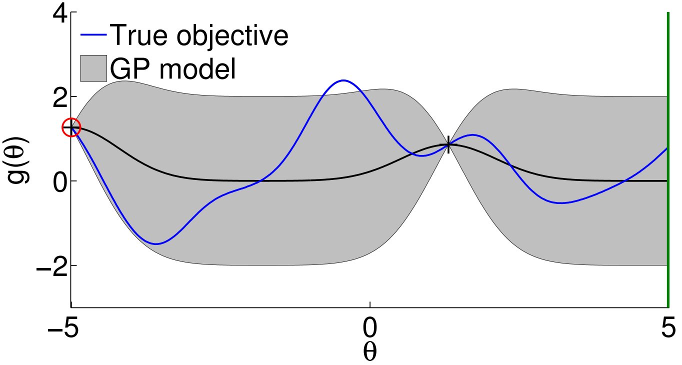
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

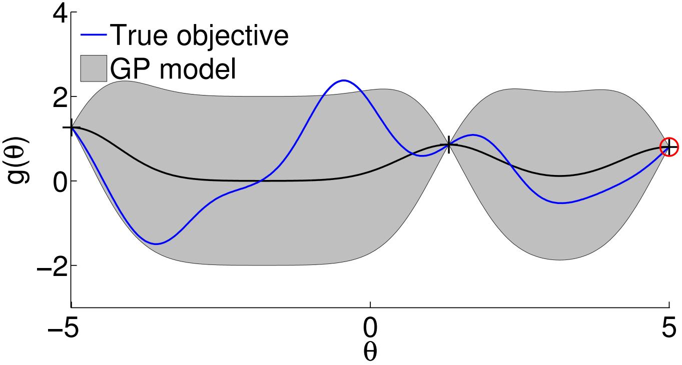
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

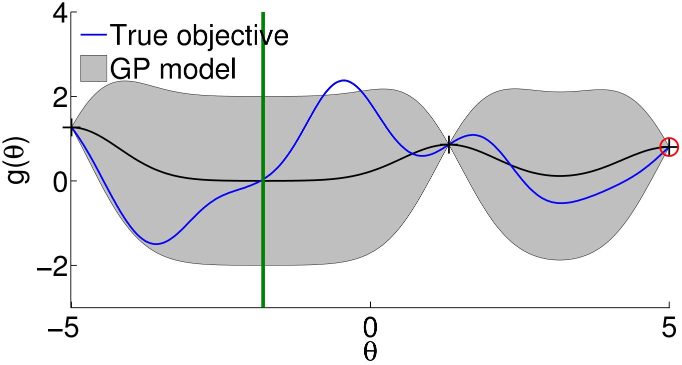
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

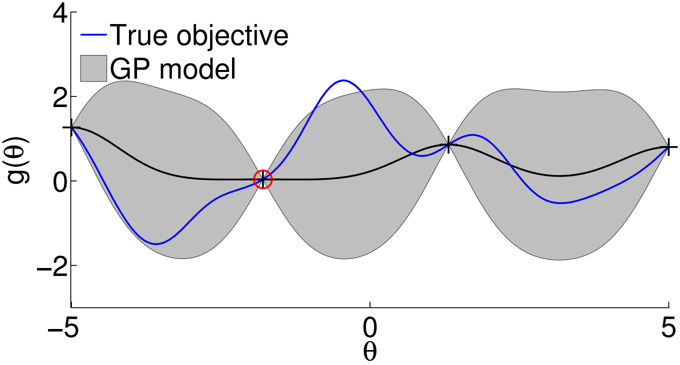
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

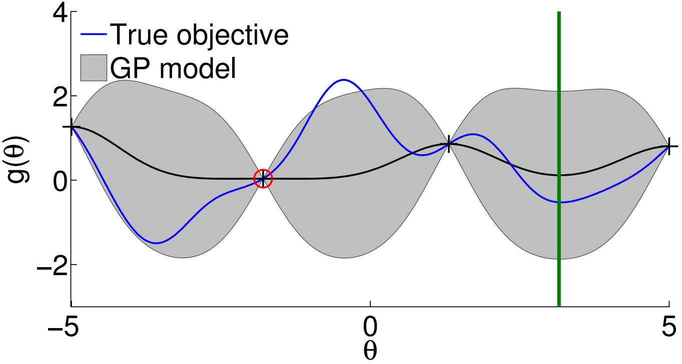
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

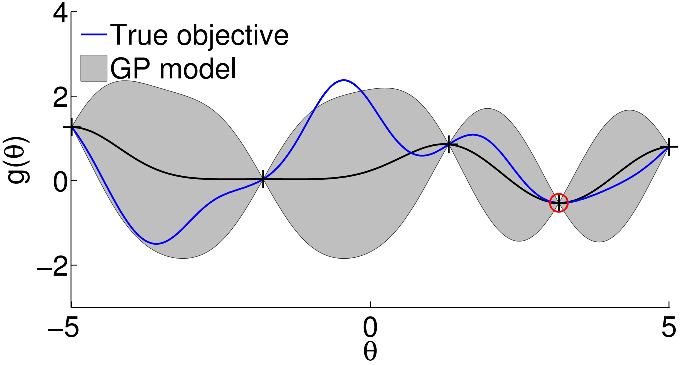
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

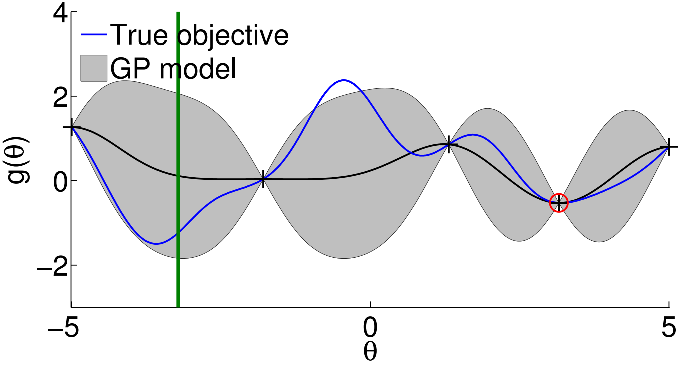
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

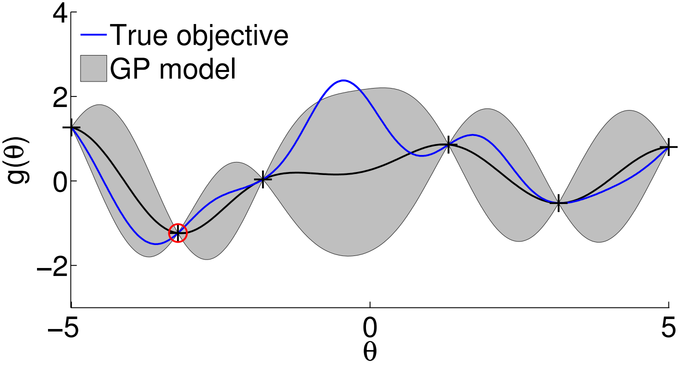
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

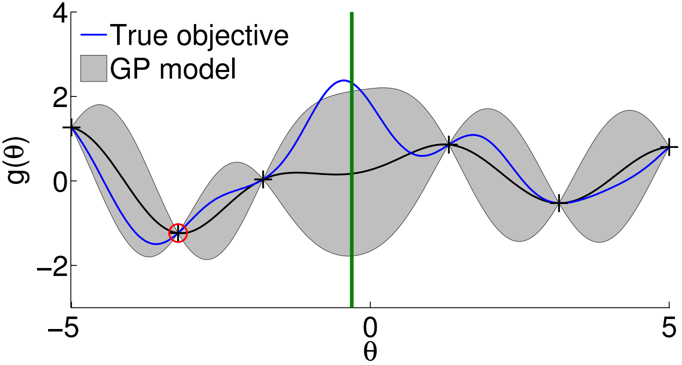
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

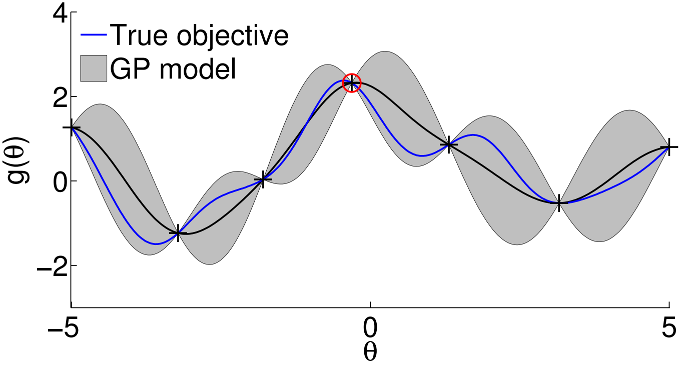
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

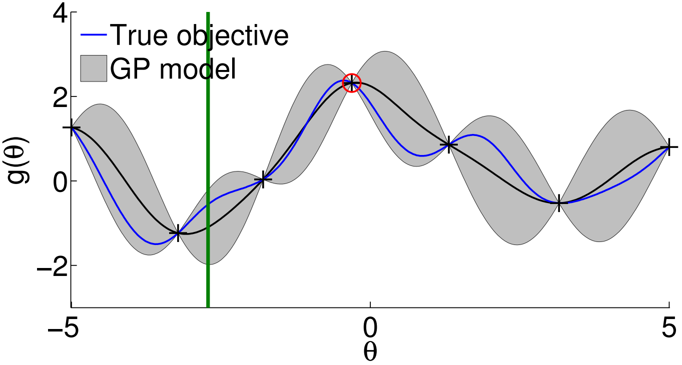
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

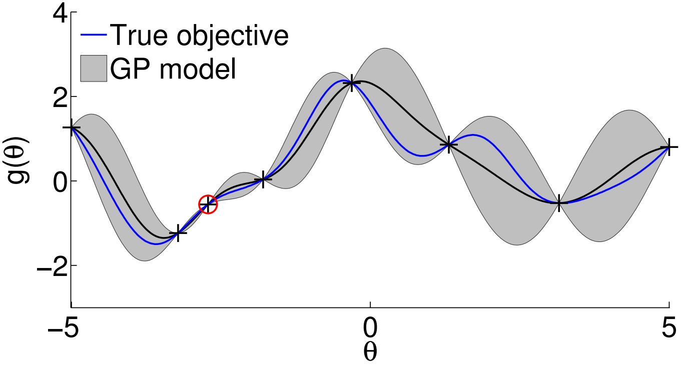
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

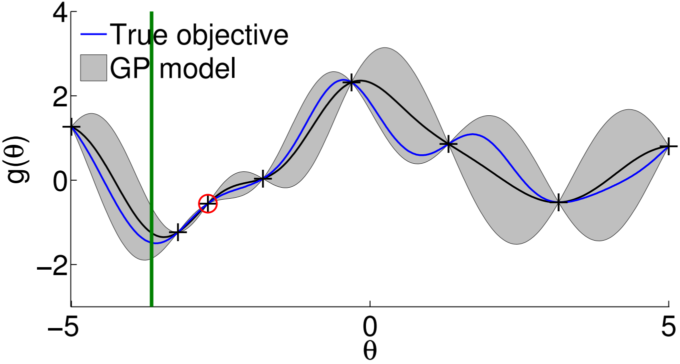
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

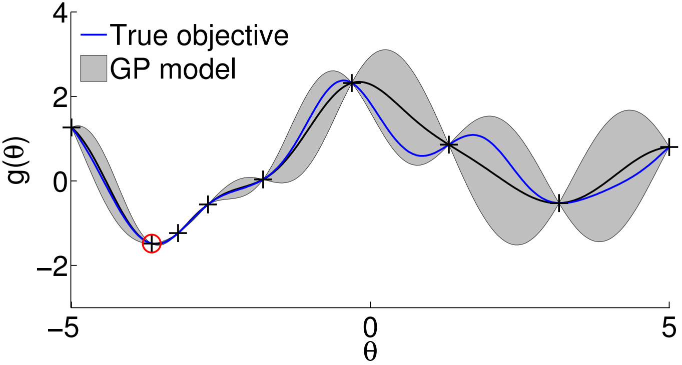
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_\theta \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

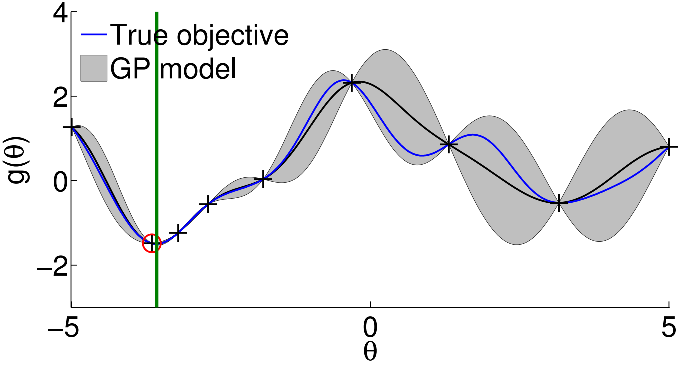
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

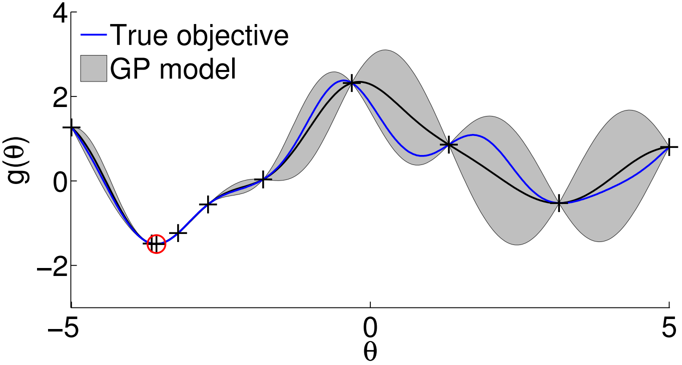
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

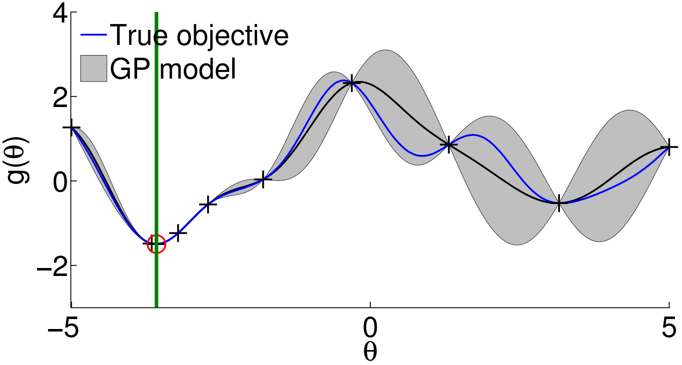
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

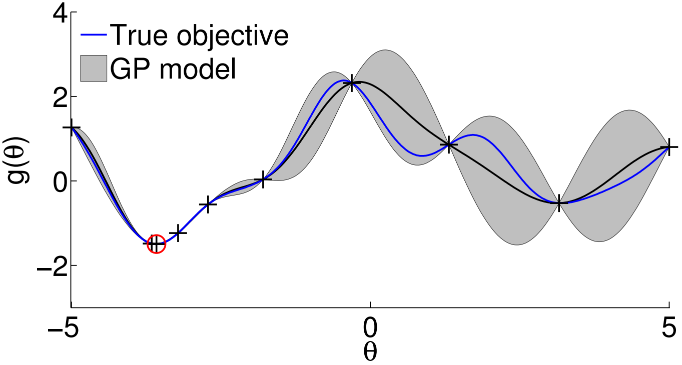
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

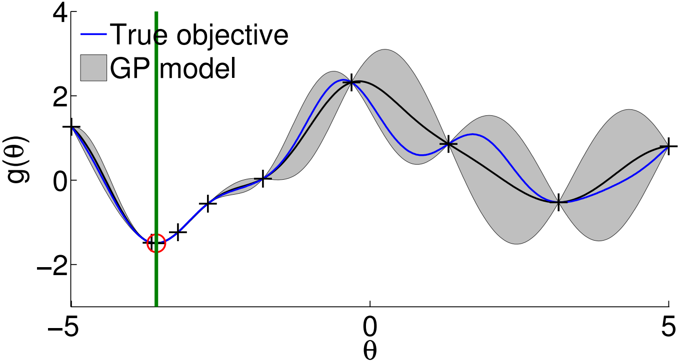
# Bayesian Optimization: Illustration




- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

# Bayesian Optimization: Illustration



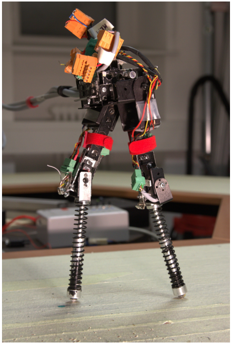

- Upper-Confidence-Bound (UCB) criterion to select next point

$$\theta^* \in \arg\min_{\theta} \quad \mathbb{E}[\tilde{g}(\theta)] - 2\sqrt{\mathbb{V}[\tilde{g}(\theta)]}$$

- Global minimum found after 10 function evaluations

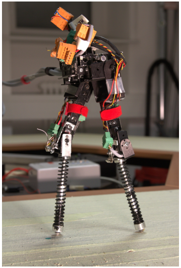
# Bayesian Gait Optimization for Legged Locomotion

- Fragile biped
  ⏩ Only few experiments feasible
- Maximize robustness and walking speed
- 4 motors:
  2 actuated hips + 2 actuated knees
- Controller implemented as a finite-state-machine (8 parameters)
- Good parameters found after 100 experiments
- Substantial speed-up compared to manual parameter search



Calandra et al. (LION, 2014): *Bayesian Gait Optimization for Bipedal Locomotion*

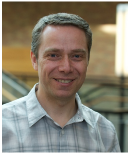
# Summary (2)



## Bayesian Gait Optimization

- Bayesian optimization for learning controllers in a few experiments
- General framework
  (no assumptions on dynamics, no explicit cost required)
- Limited to few parameters ($\approx$ 10–20)

## Collaborators

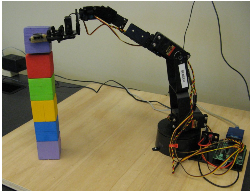
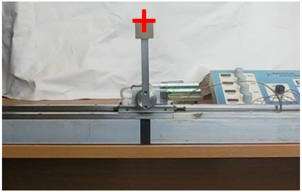
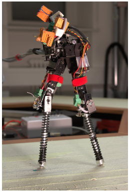
CE Rasmussen (U Cambridge)
D Fox (U Washington)



R Calandra (TU Darmstadt)
J Peters (TU Darmstadt)
A Seyfarth (TU Darmstadt)

## Funding

# Wrap-up

- Data-efficient controller learning for autonomous systems
    - Reinforcement learning
    - Bayesian optimization
- Key to success: Probabilistic modeling and Bayesian inference

`m.deisenroth@imperial.ac.uk`

**Thank you for your attention**

# References


[1] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1 of *Optimization and Computation Series*. Athena Scientific, Belmont, MA, USA, 3rd edition, 2005.

[2] B. Bischoff, D. Nguyen-Tuong, T. Koller, H. Markert, and A. Knoll. Learning Throttle Valve Control Using Policy Search. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, 2013.

[3] R. Calandra, N. Gopalan, A. Seyfarth, J. Peters, and M. P. Deisenroth. Bayesian Gait Optimization for Bipedal Locomotion. In *Proceedings of the International Conference on Learning and Intelligent Optimization*, 2014.

[4] R. Calandra, J. Peters, A. Seyfarth, and M. P. Deisenroth. An Experimental Evaluation of Bayesian Optimization on Bipedal Locomotion. In *Proceedings of the International Conference on Robotics and Automation*, 2014.

[5] M. P. Deisenroth. *Efficient Reinforcement Learning using Gaussian Processes*, volume 9 of *Karlsruhe Series on Intelligent Sensor-Actuator-Systems*. KIT Scientific Publishing, November 2010. ISBN 978-3-86644-569-7.

[6] M. P. Deisenroth, D. Fox, and C. E. Rasmussen. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[7] M. P. Deisenroth and C. E. Rasmussen. PILCO: A Model-Based and Data-Efficient Approach to Policy Search. In *Proceedings of the International Conference on Machine Learning*, pages 465–472, New York, NY, USA, June 2011. ACM.

[8] M. P. Deisenroth, C. E. Rasmussen, and D. Fox. Learning to Control a Low-Cost Manipulator using Data-Efficient Reinforcement Learning. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2011.

[9] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, 13(4):455–492, December 1998.

[10] M. A. Osborne, R. Garnett, and S. J. Roberts. Gaussian Processes for Global Optimization. In *Proceedings of the International Conference on Learning and Intelligent Optimization*, 2009.

[11] J. Quiñonero-Candela, A. Girard, J. Larsen, and C. E. Rasmussen. Propagation of Uncertainty in Bayesian Kernel Models—Application to Multiple-Step Ahead Forecasting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 701–704, April 2003.

[12] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, MA, USA, 2006.

[13] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, MA, USA, 1998.