Imperial College London

Probabilistic Modeling for Fast Autonomous Learning

Marc Deisenroth

Department of Computing
Imperial College London

@mpd37
m.deisenroth@imperial.ac.uk

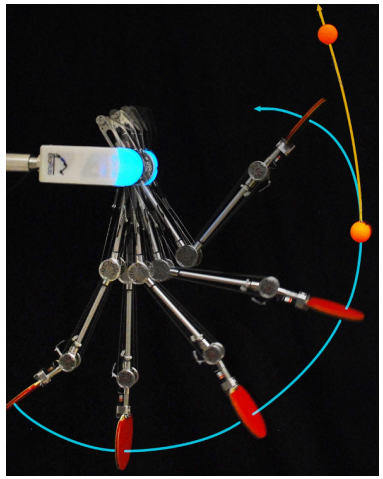
University of Nairobi, Kenya

August 29, 2019

Autonomous Robots: Key Challenges

- Three key challenges in autonomous systems:

  Modeling. Predicting. Decision making.



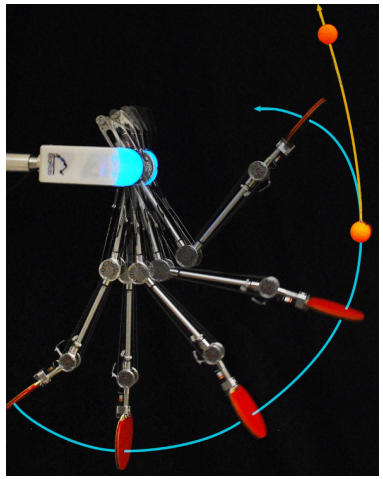
Robotics

Autonomous Robots: Key Challenges

- Three key challenges in autonomous systems:

  Modeling. Predicting. Decision making.

- No human in the loop ⏩ "Learn" from data
- Automatically extract information
- Data-efficient (fast) learning
- Uncertainty: sensor noise, unknown processes, limited knowledge, ...



Robotics

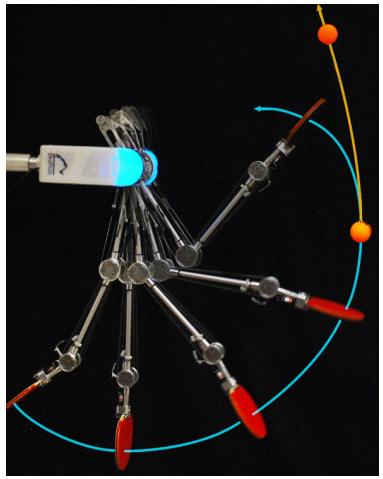
Autonomous Robots: Key Challenges

- Three key challenges in autonomous systems:

  Modeling. Predicting. Decision making.

- No human in the loop ⏩ "Learn" from data
- Automatically extract information
- Data-efficient (fast) learning
- Uncertainty: sensor noise, unknown processes, limited knowledge, ...

⏩ **Probabilistic machine learning**



Robotics

Overview

Model-based Reinforcement Learning

Safe Exploration

Meta Reinforcement Learning

Reinforcement Learning

$$x_{t+1} = f(x_t, u_t) + w, \qquad u_t = \pi(x_t, \theta)$$

State

Transition function

Control

Policy

Policy parameters

Reinforcement Learning

$$x_{t+1} = f(x_t, u_t) + w\,, \qquad u_t = \pi(x_t, \theta)$$

State — Transition function — Control — Policy — Policy parameters

Objective (Controller Learning)

Find policy parameters θ^* that minimize the expected long-term cost

$$J(\theta) = \sum_{t=1}^{T} \mathbb{E}[c(x_t)|\theta]\,, \qquad p(x_0) = \mathcal{N}(\mu_0, \Sigma_0)\,.$$

Instantaneous cost $c(x_t)$, e.g., $\|x_t - x_{\text{target}}\|^2$

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

Fast Reinforcement Learning

Objective

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

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

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

PILCO Framework: High-Level Steps

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

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

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

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification
2. Compute long-term predictions $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$
3. Policy improvement

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

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

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification
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, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

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

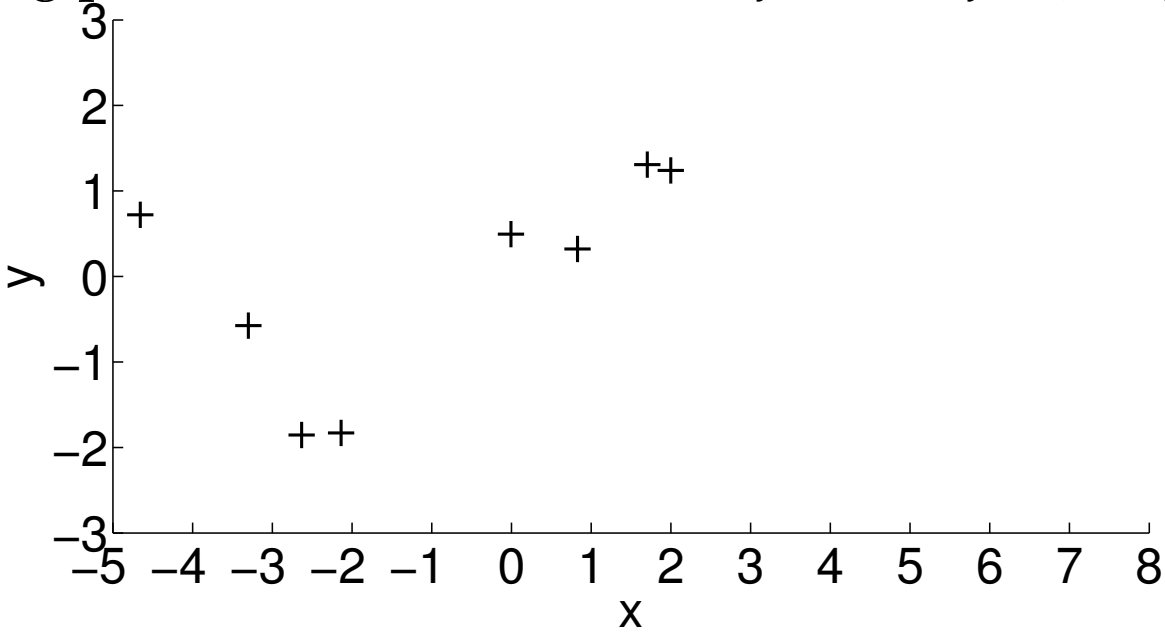
PILCO Framework: High-Level Steps

1. **Probabilistic model for transition function f**
   - ⏩ **System identification**
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, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Model Learning (System Identification)

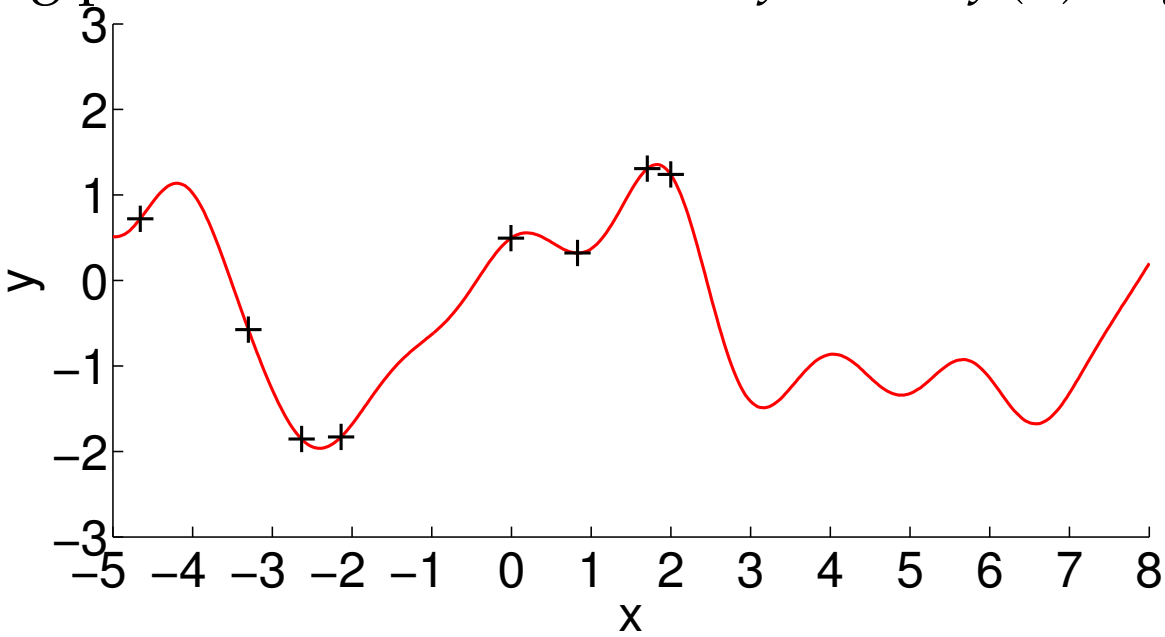
Model learning problem: Find a function $f : x \mapsto f(x) = y$



Observed function values

Model Learning (System Identification)

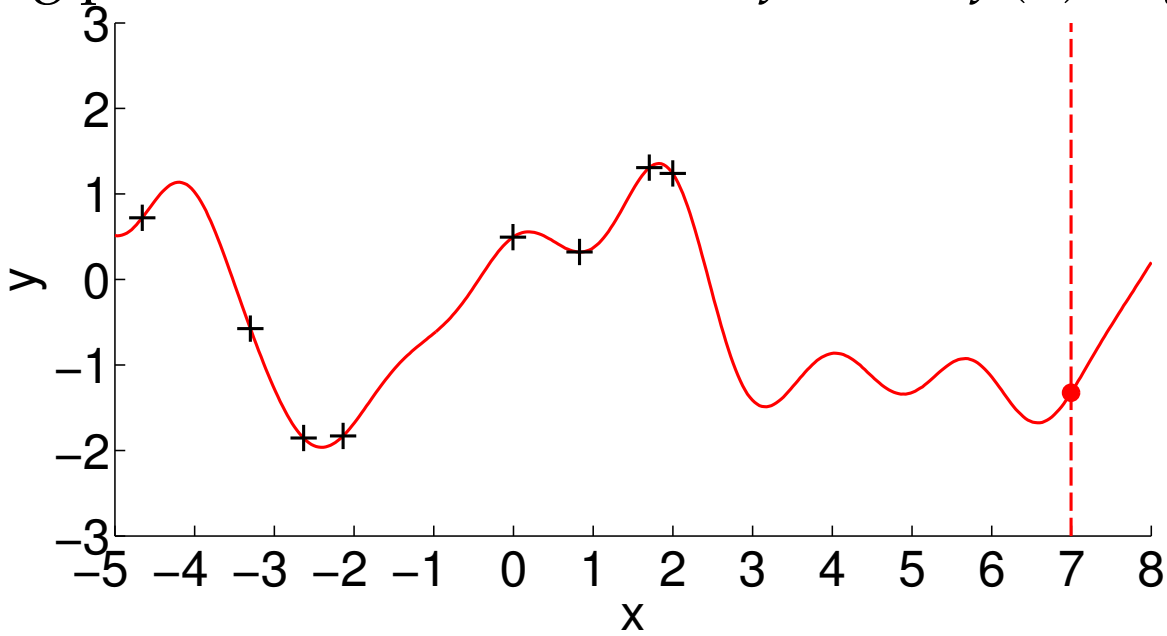
Model learning problem: Find a function $f : x \mapsto f(x) = y$



Plausible model

Model Learning (System Identification)

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

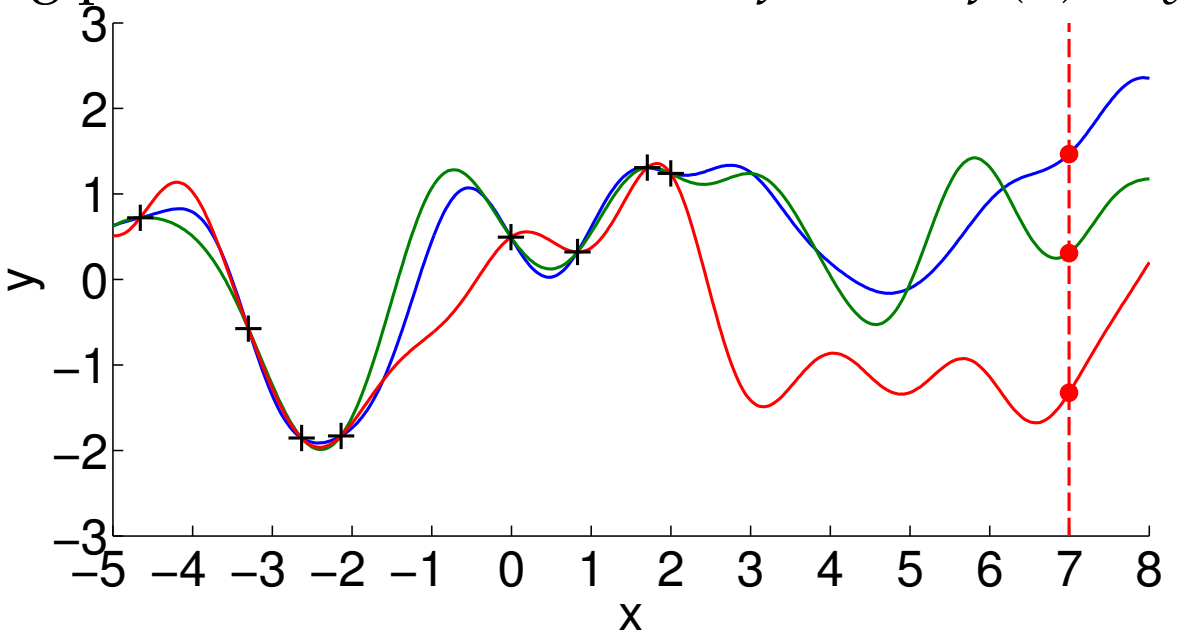


Plausible model

Predictions? Decision Making?

Model Learning (System Identification)

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

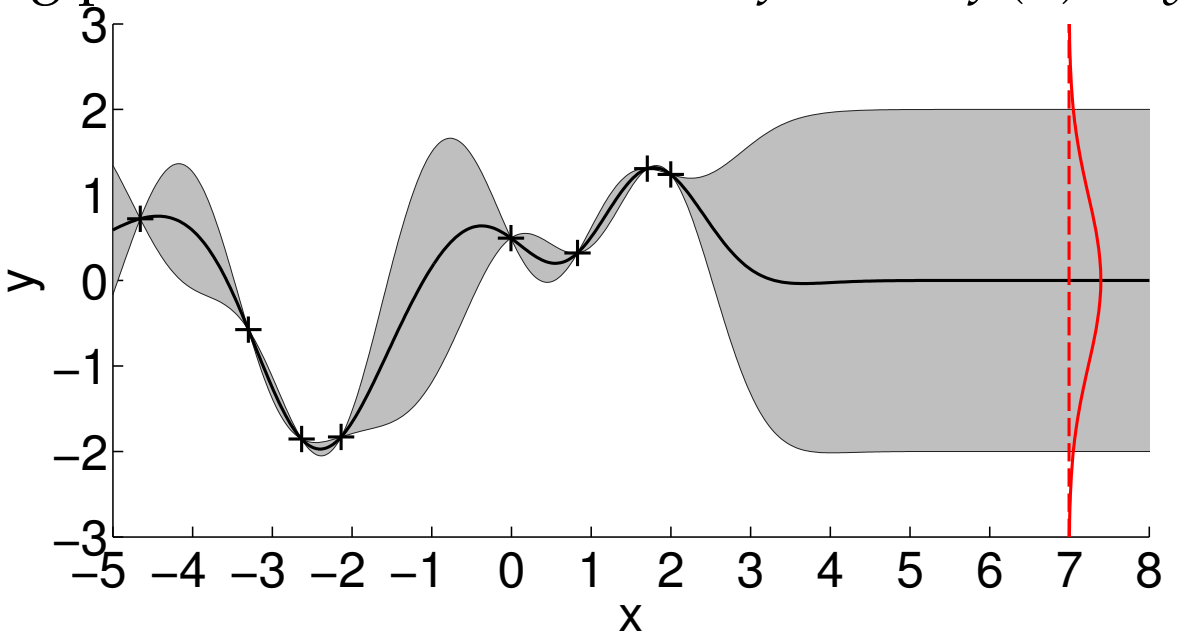


More plausible models

Predictions? Decision Making? Model Errors!

Model Learning (System Identification)

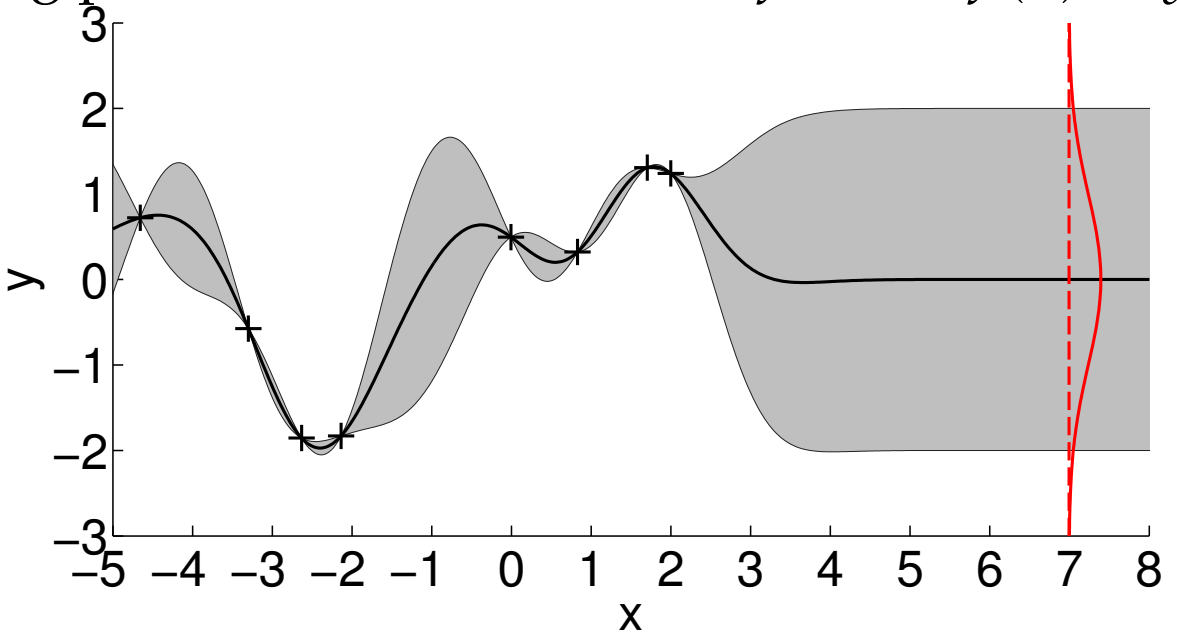
Model learning problem: Find a function $f : x \mapsto f(x) = y$



Distribution over plausible functions

Model Learning (System Identification)

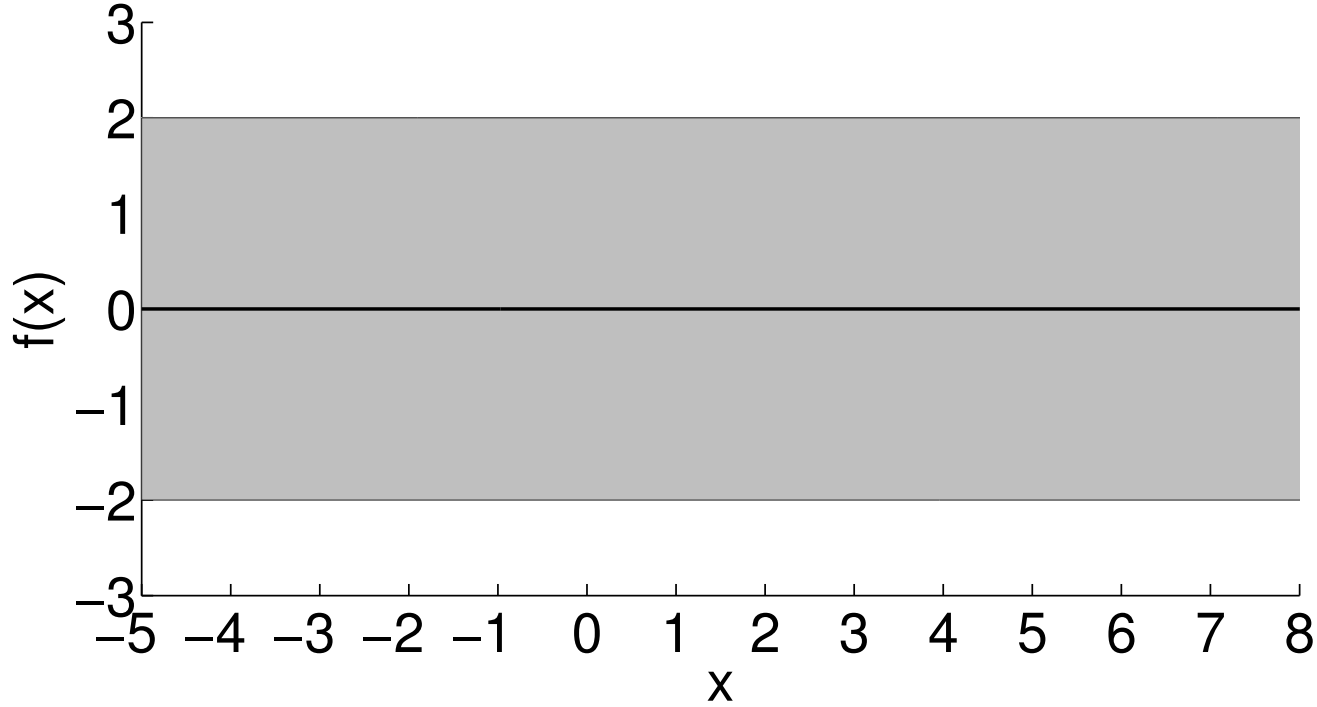
Model learning problem: Find a function $f : x \mapsto f(x) = y$



Distribution over plausible functions

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

Intuitive Introduction to Gaussian Processes

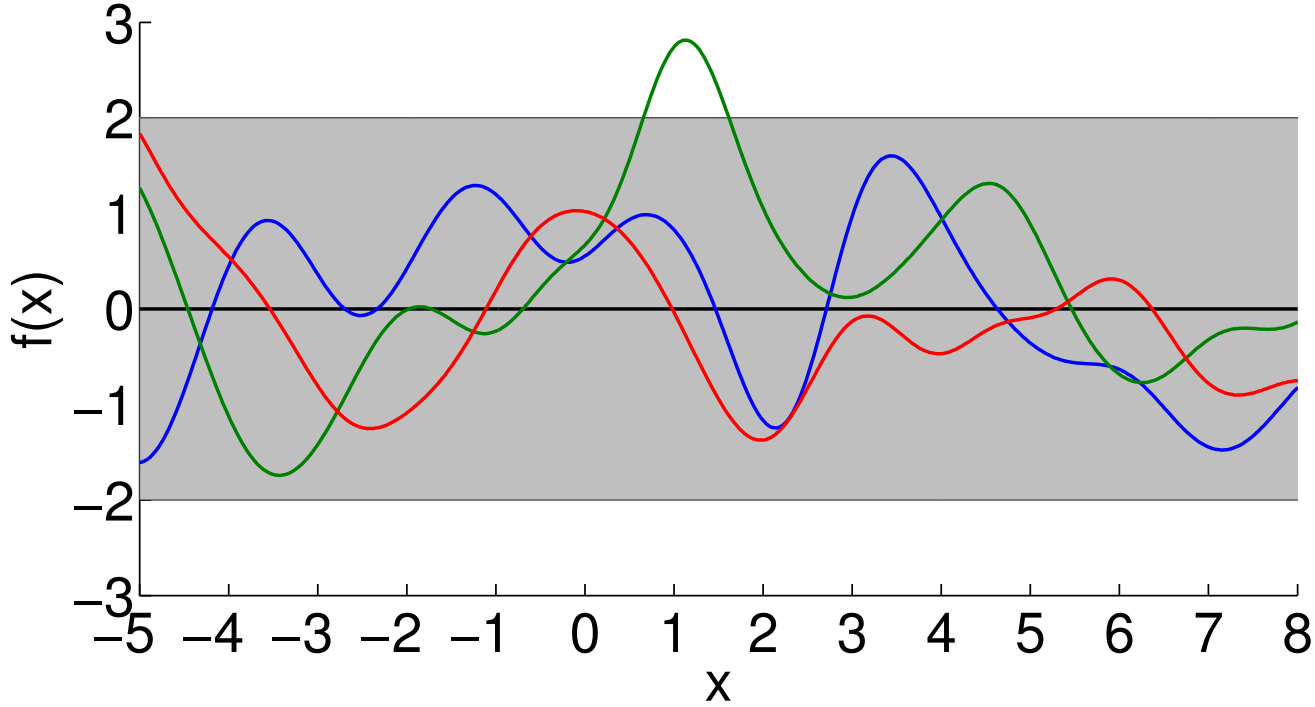


Prior belief about the function

Predictive (marginal) mean and variance:

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

Intuitive Introduction to Gaussian Processes

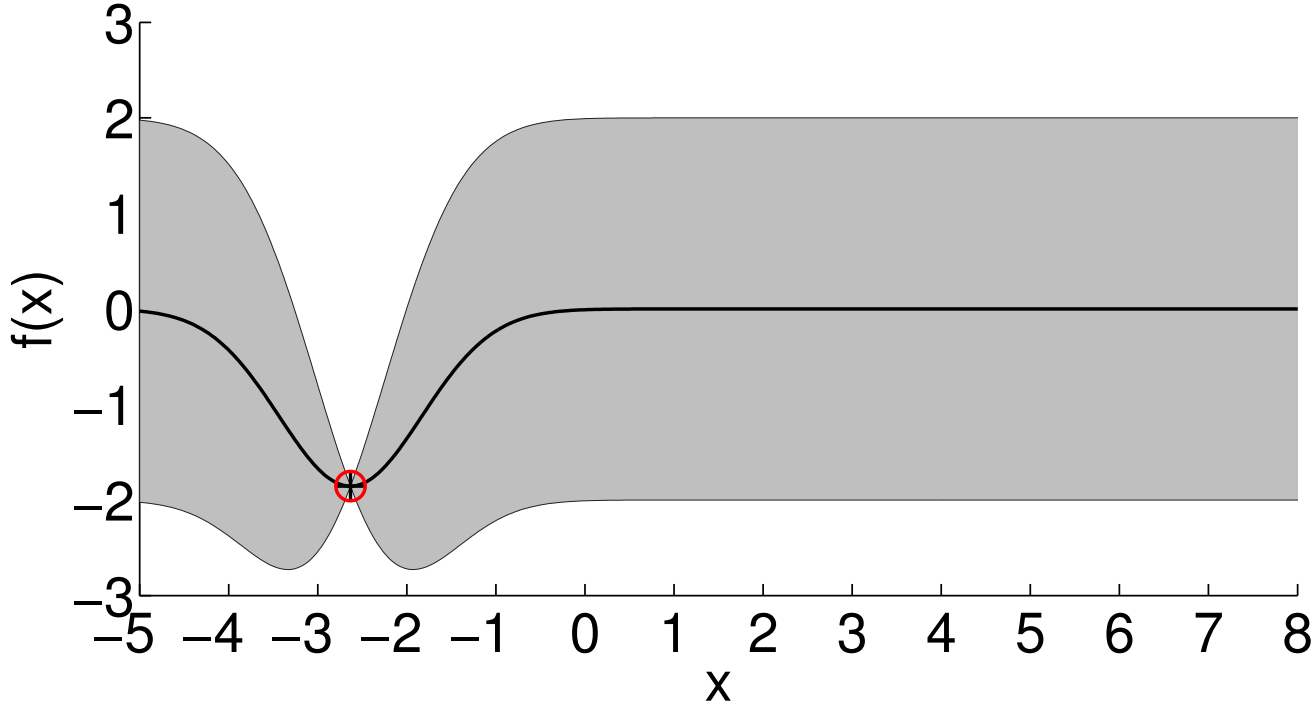


Prior belief about the function

Predictive (marginal) mean and variance:

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

Intuitive Introduction to Gaussian Processes



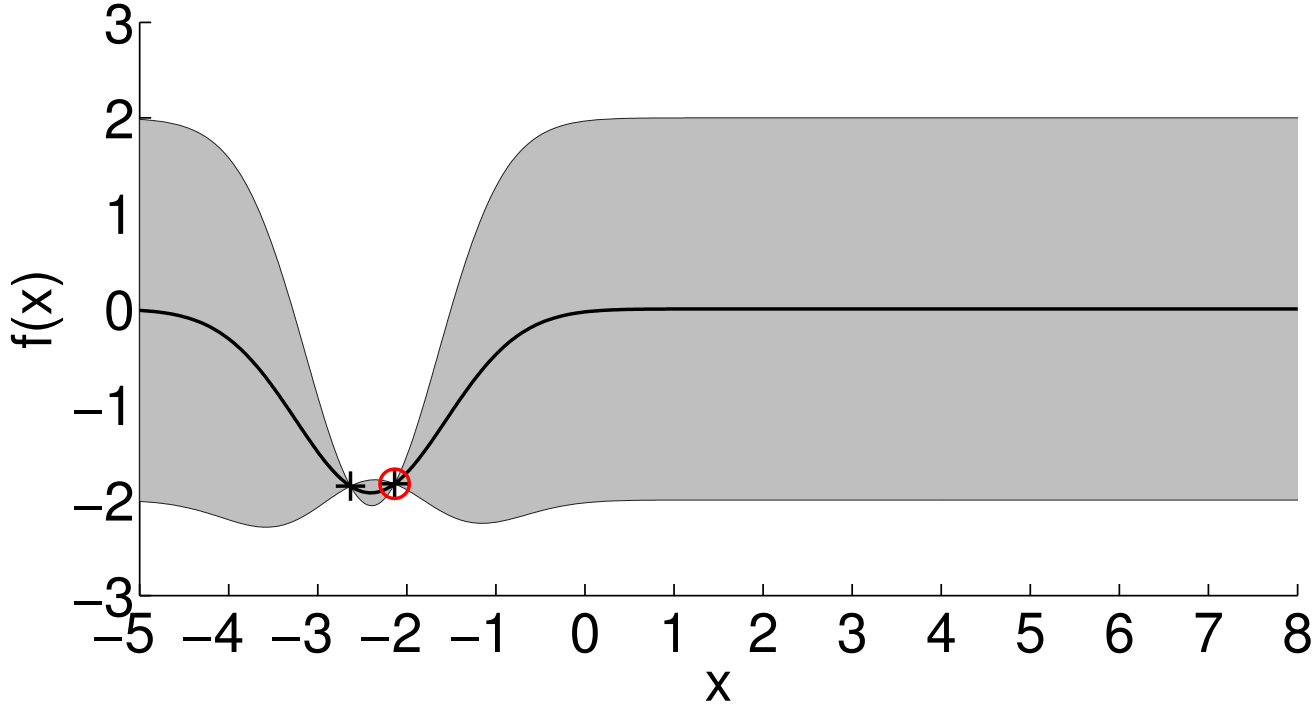
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



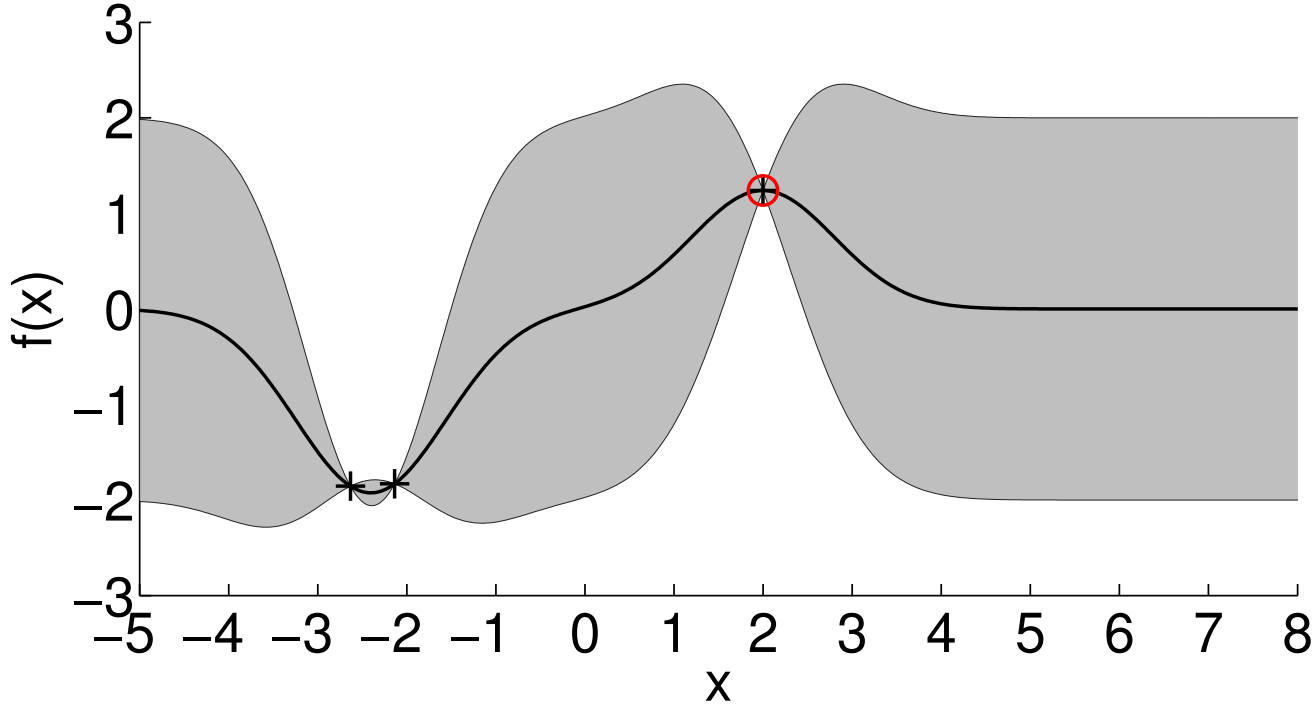
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



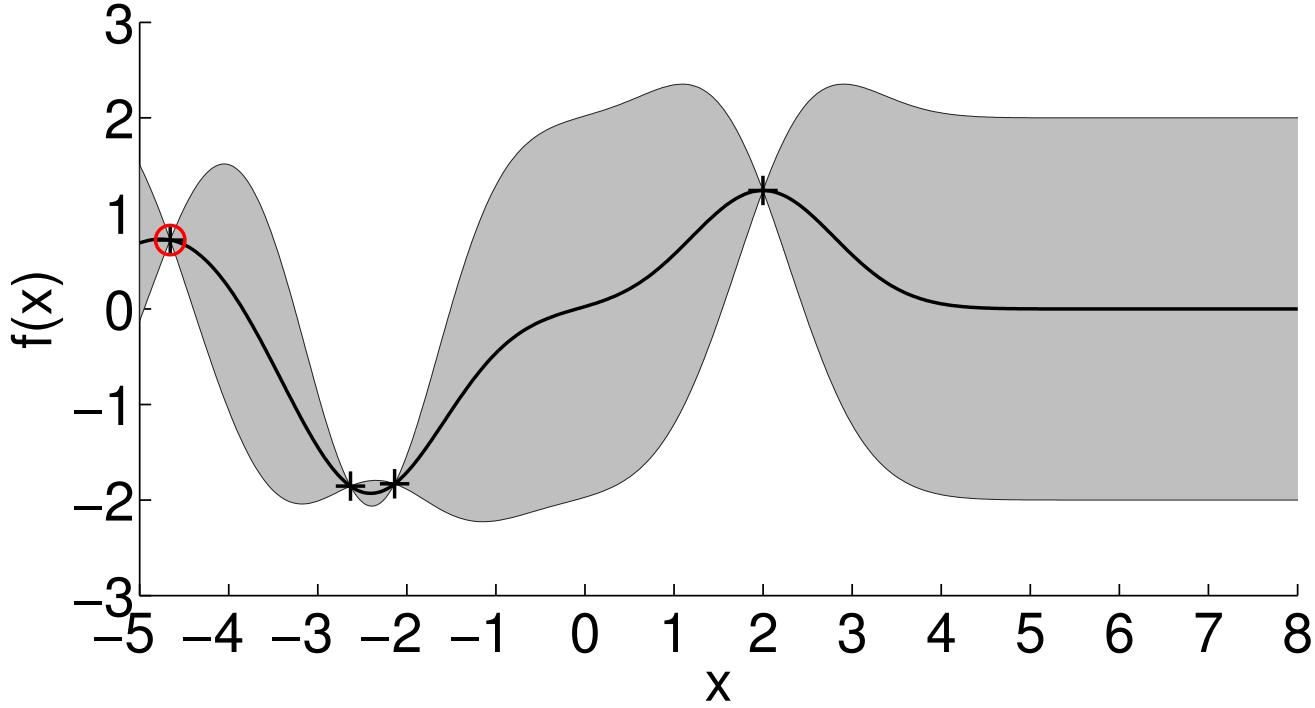
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



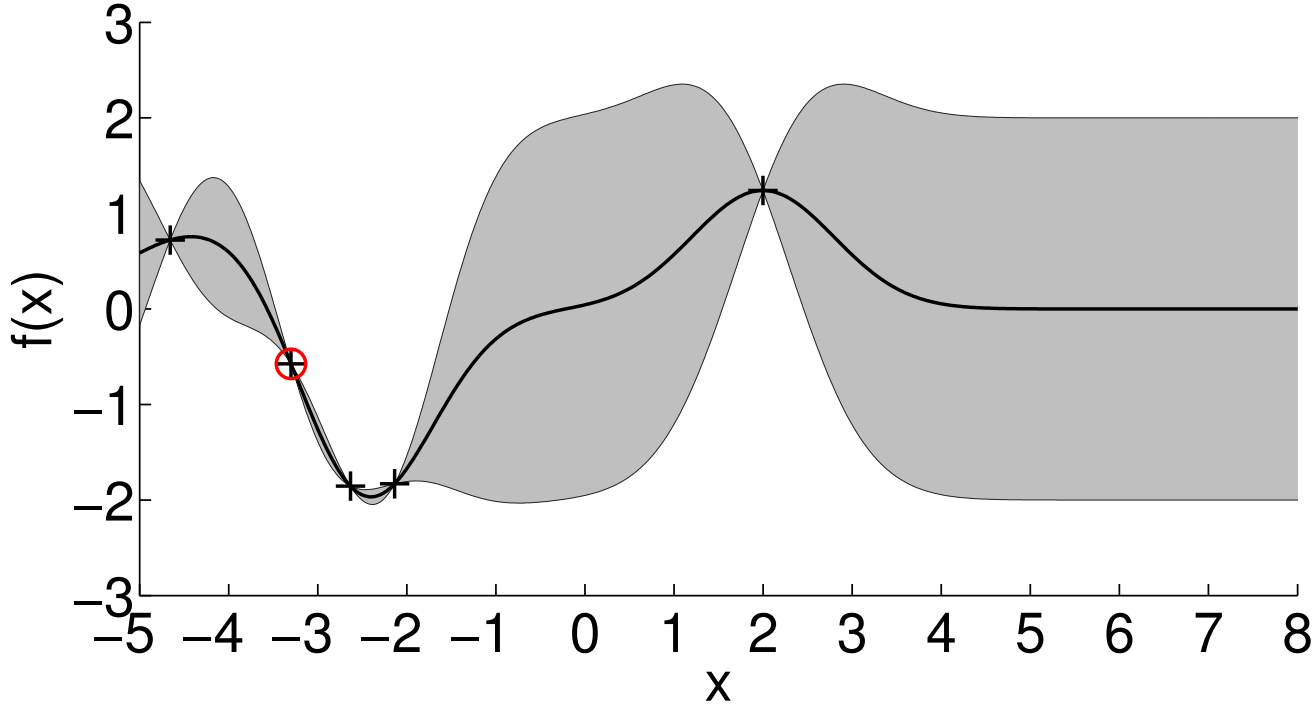
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



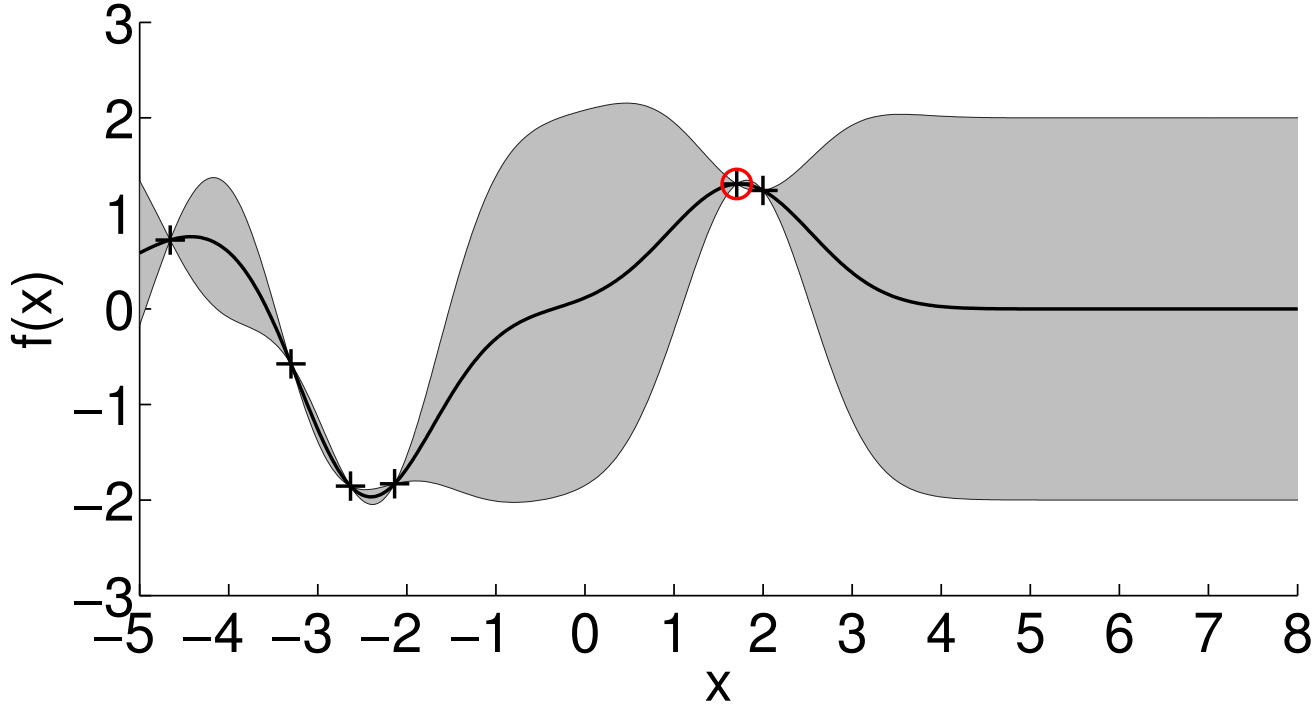
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



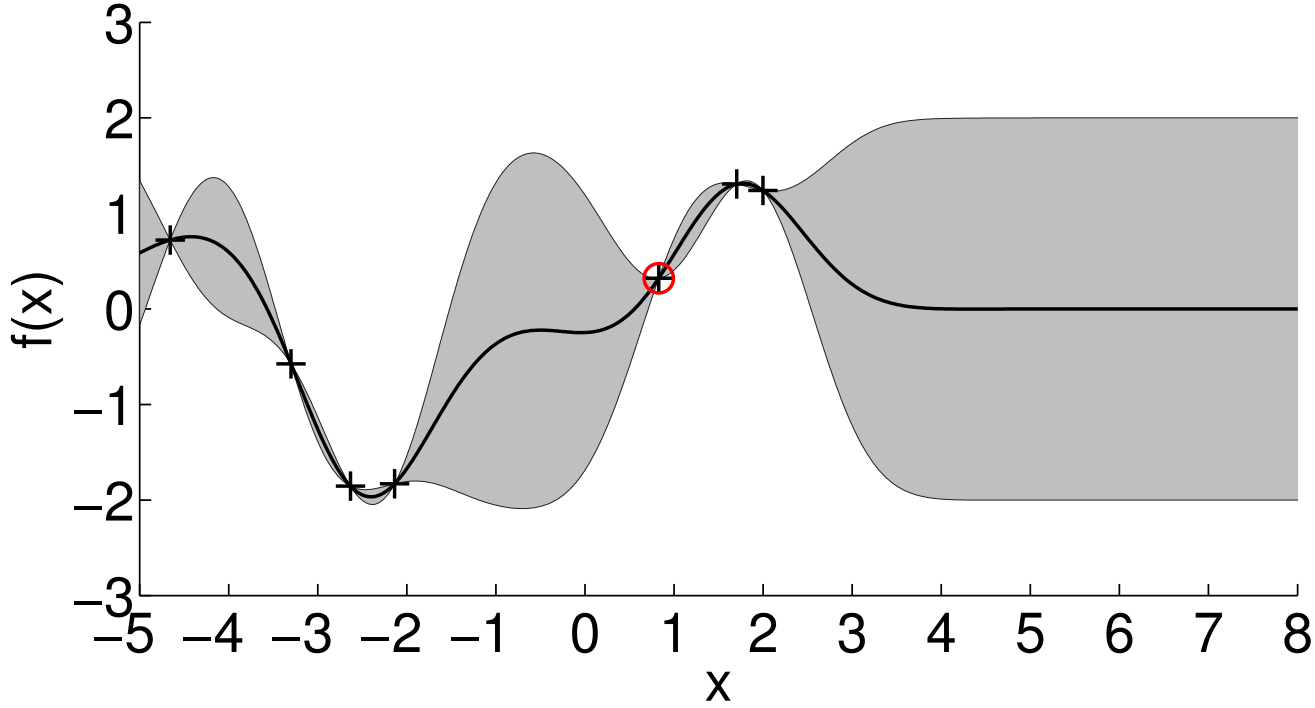
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



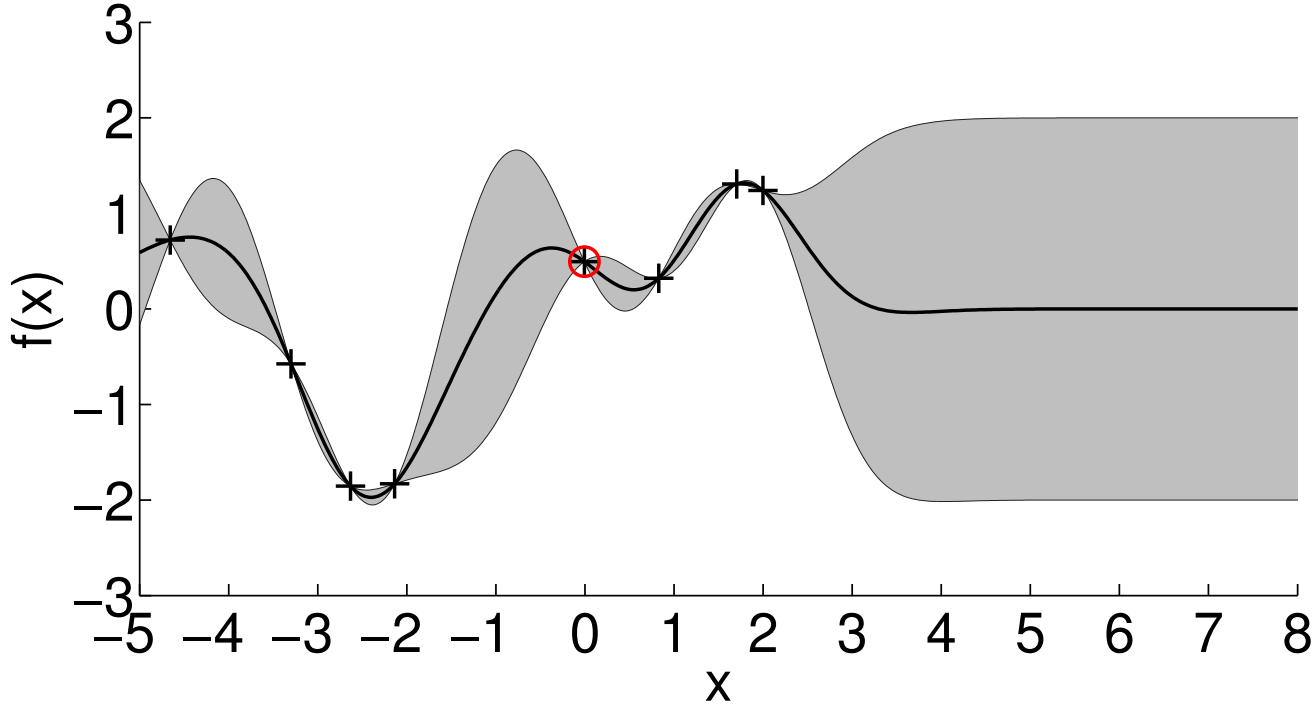
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



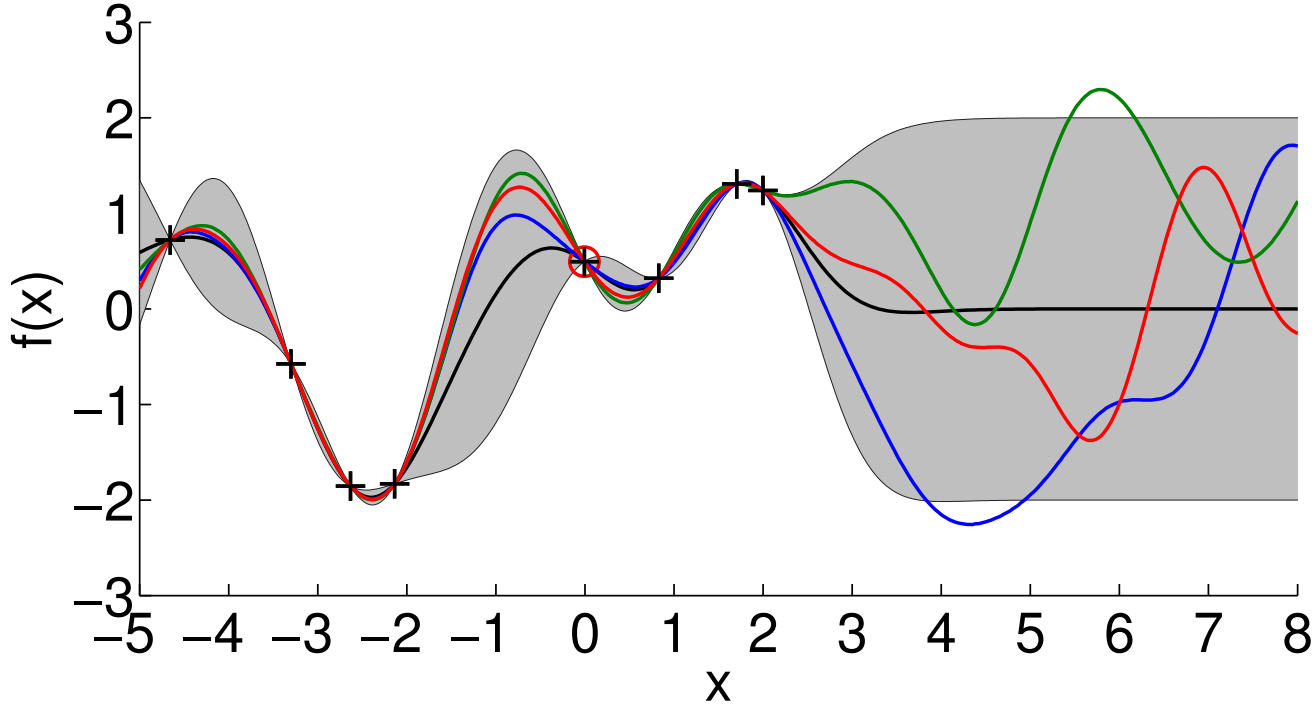
Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Intuitive Introduction to Gaussian Processes



Posterior belief about the function

Predictive (marginal) mean and variance:

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

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

Fast Reinforcement Learning

Objective

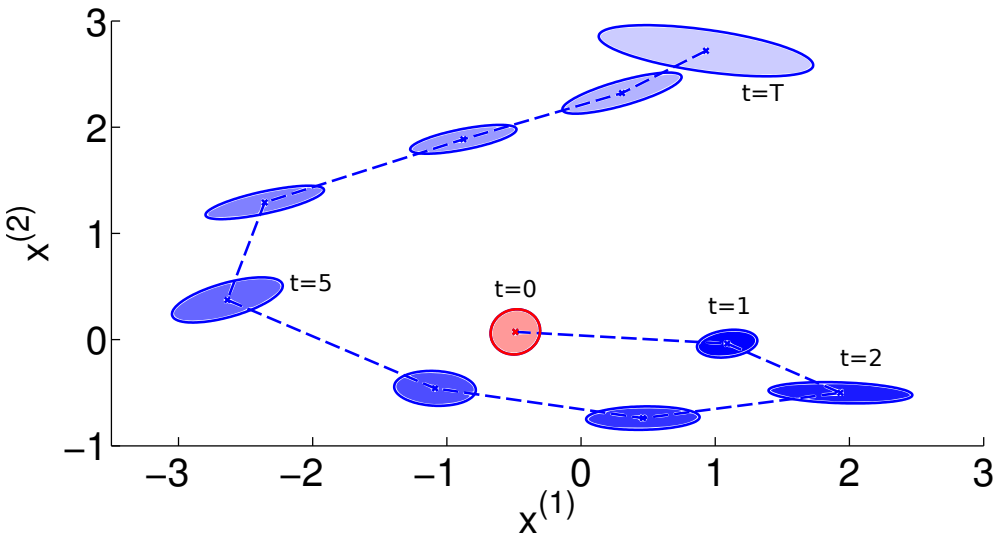
Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification
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, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

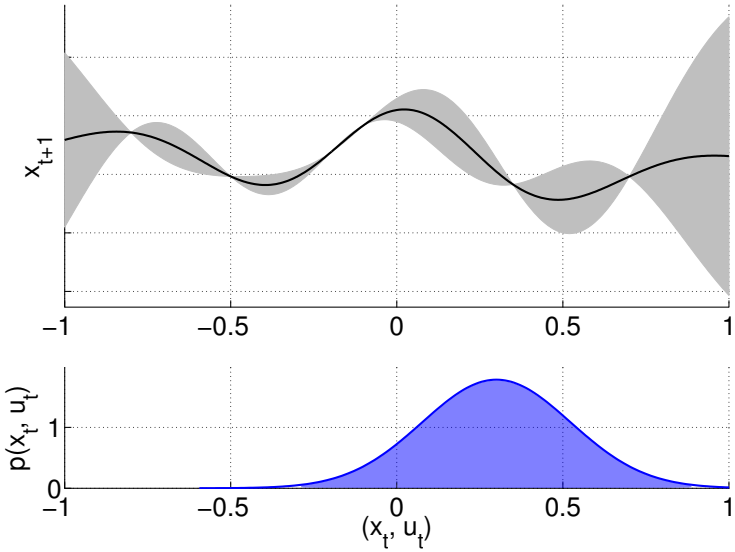
Long-Term Predictions



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

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Long-Term Predictions

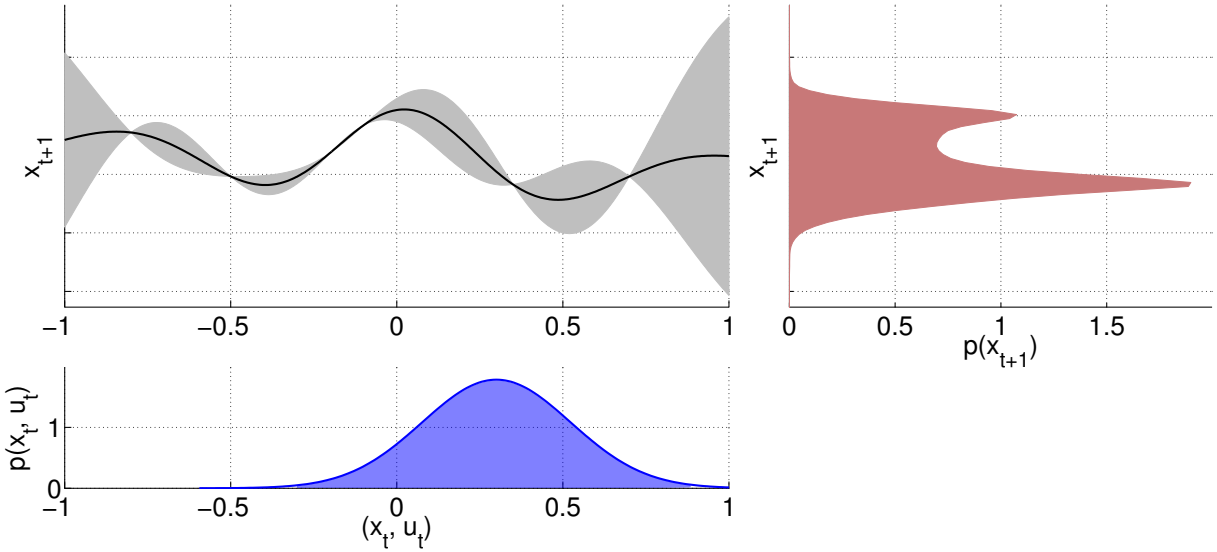


- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

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

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Long-Term Predictions

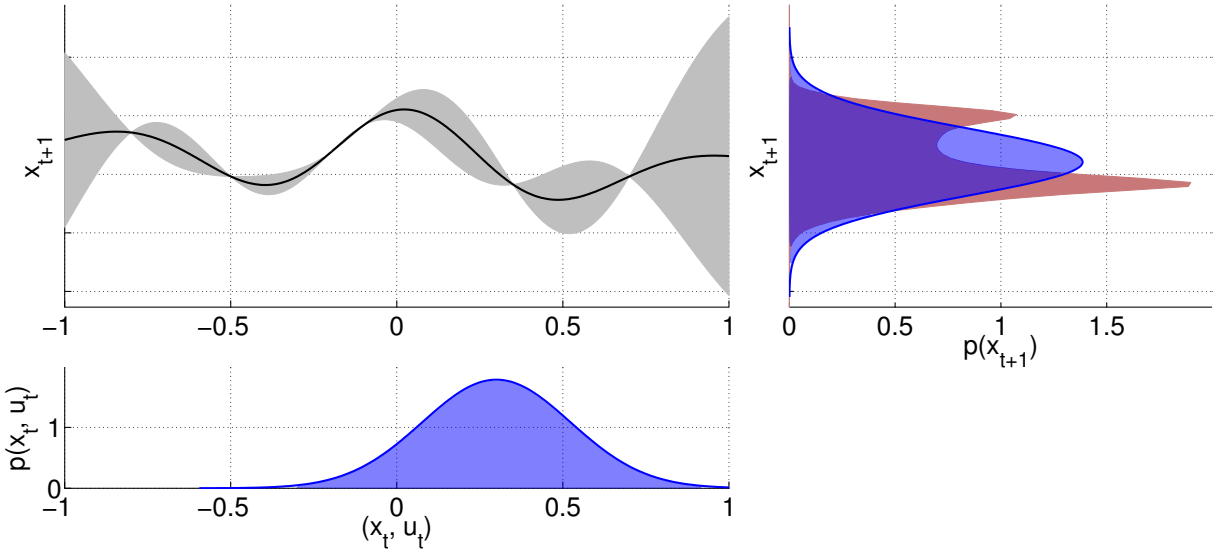


- Iteratively compute $p(\boldsymbol{x}_1|\boldsymbol{\theta}), \ldots, p(\boldsymbol{x}_T|\boldsymbol{\theta})$

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

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Long-Term Predictions



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

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

⏩ GP moment matching (Girard et al., 2002; Quiñonero-Candela et al., 2003)

Deisenroth et al. (IEEE-TPAMI, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

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

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification
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, 2015): *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, 2015): *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

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

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

Deisenroth et al. (IEEE-TPAMI, 2015): *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

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

  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, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Fast Reinforcement Learning

Objective

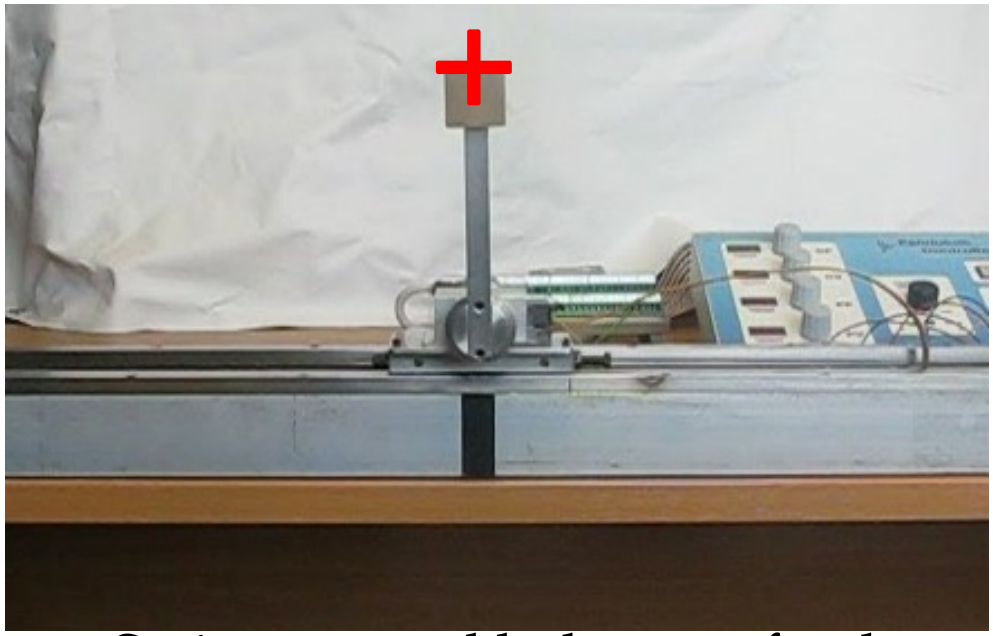
Minimize expected long-term cost $J(\boldsymbol{\theta}) = \sum_t \mathbb{E}[c(\boldsymbol{x}_t)|\boldsymbol{\theta}]$

PILCO Framework: High-Level Steps

1. Probabilistic model for transition function f
   - ⏩ System identification
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, 2015): *Gaussian Processes for Data-Efficient Learning in Robotics and Control*

Standard Benchmark Problem: Cart-Pole Swing-up

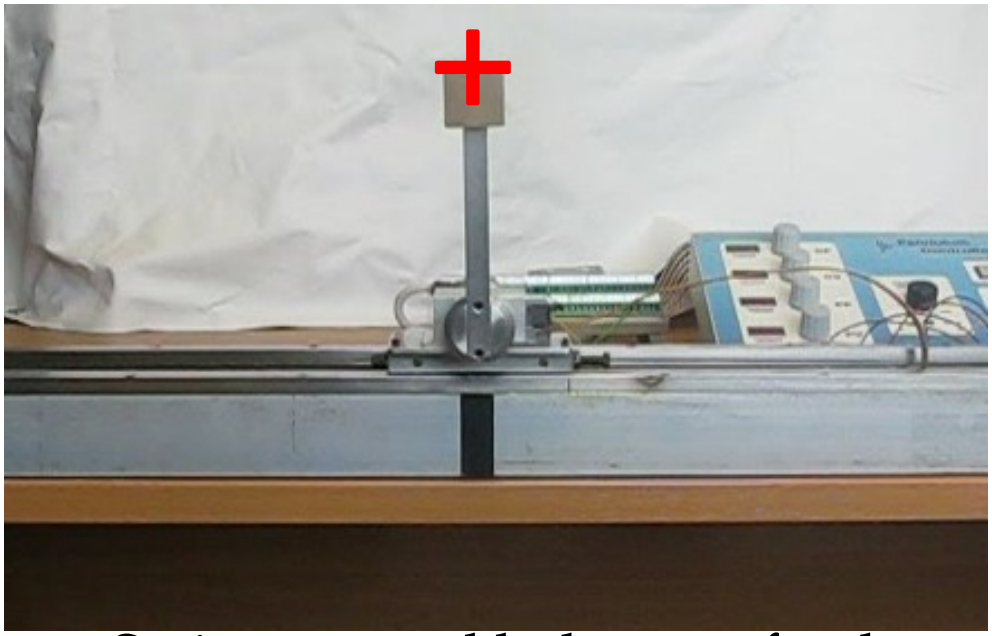


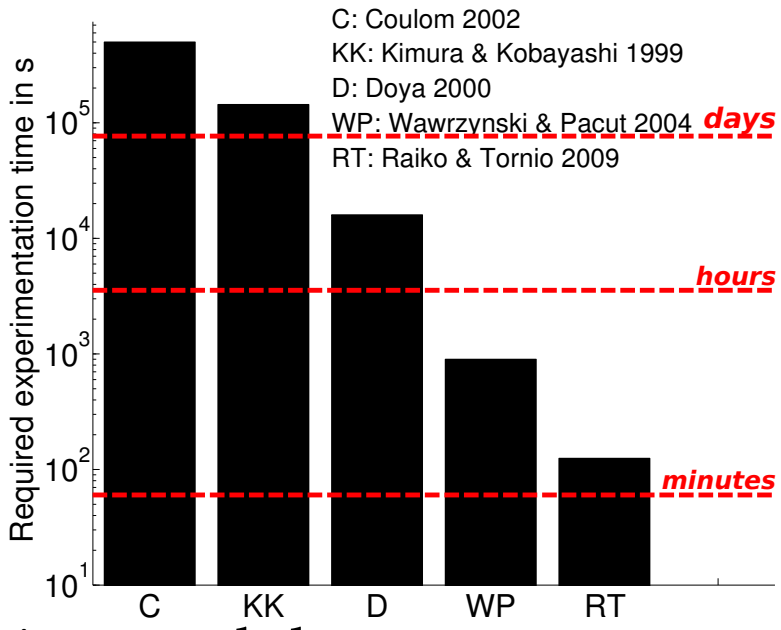
- Swing up and balance a freely swinging pendulum on a cart
- No knowledge about nonlinear dynamics ⏩ Learn from scratch
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$

- Code available at `https://github.com/ICL-SML/pilco-matlab`

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

Standard Benchmark Problem: Cart-Pole Swing-up



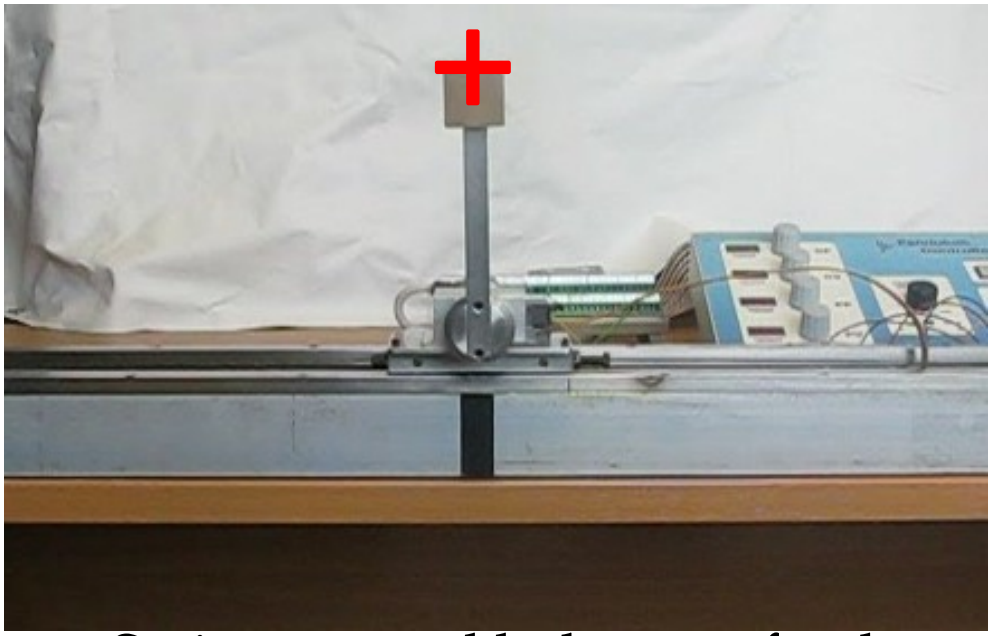


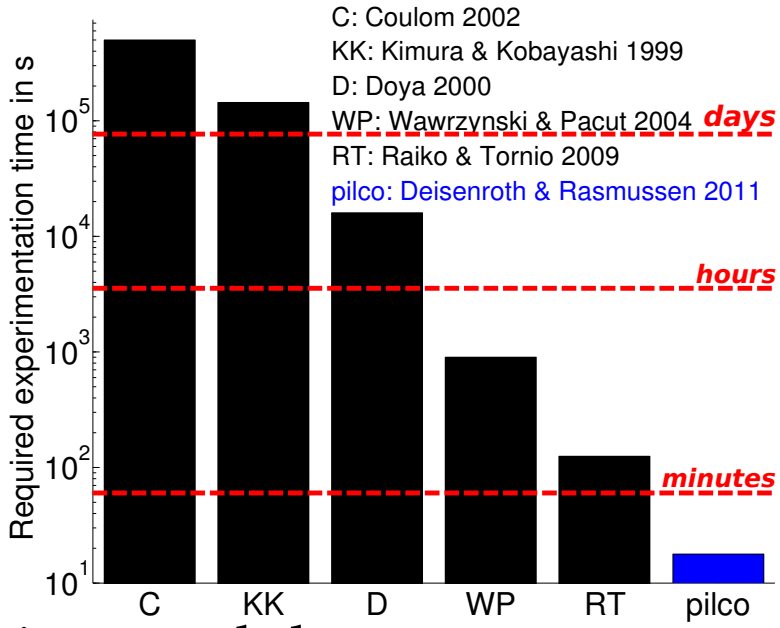
- Swing up and balance a freely swinging pendulum on a cart
- No knowledge about nonlinear dynamics ⏩ Learn from scratch
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$

- Code available at `https://github.com/ICL-SML/pilco-matlab`

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

Standard Benchmark Problem: Cart-Pole Swing-up

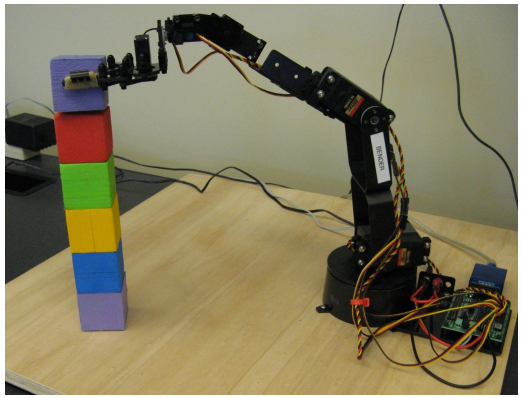




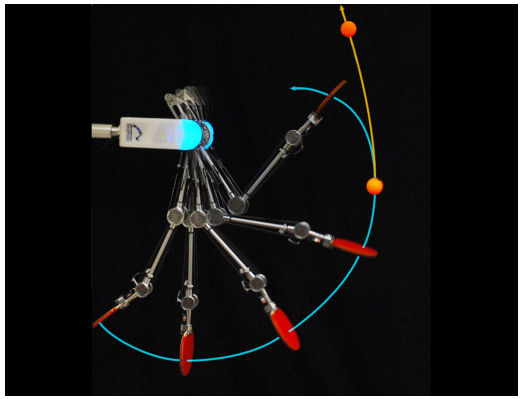
- Swing up and balance a freely swinging pendulum on a cart
- No knowledge about nonlinear dynamics ⏩ Learn from scratch
- Cost function $c(\boldsymbol{x}) = -\exp(-\|\boldsymbol{x} - \boldsymbol{x}_{\text{target}}\|^2)$
- **Unprecedented learning speed** compared to state-of-the-art
- Code available at `https://github.com/ICL-SML/pilco-matlab`

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

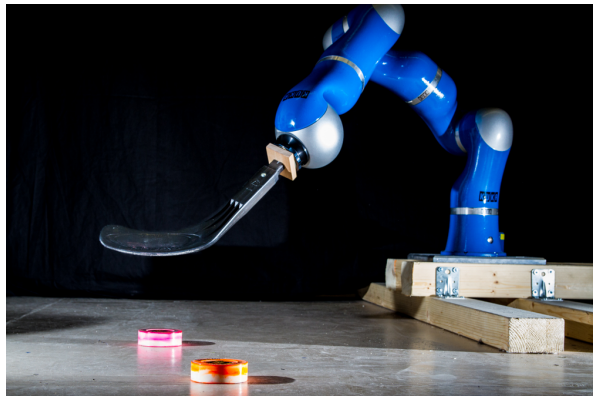
Wide Applicability



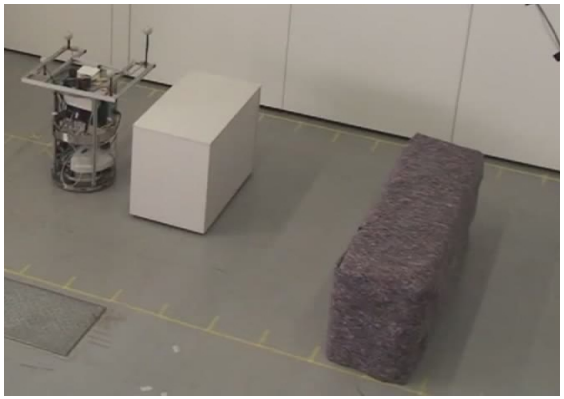
with D Fox



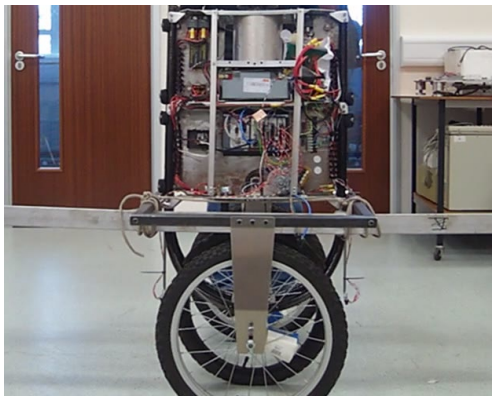
with P Englert, A Paraschos, J Peters



with A Kupcsik, J Peters, G Neumann



B Bischoff (Bosch), ESANN 2013

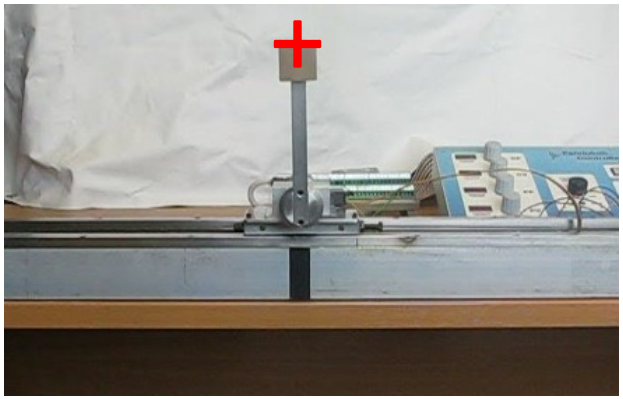


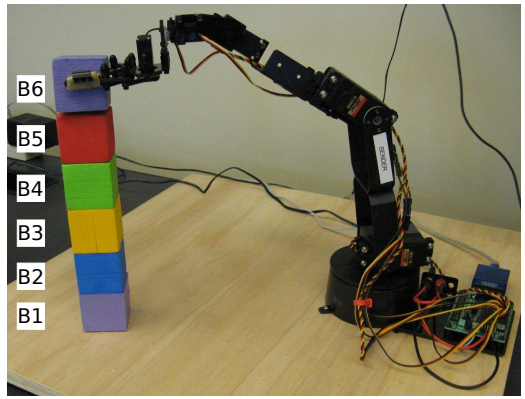
A McHutchon (U Cambridge)

- Application to a wide range of robotic systems

Deisenroth et al. (RSS, 2011): *Learning to Control a Low-Cost Manipulator using Data-efficient Reinforcement Learning*
Englert et al. (ICRA, 2013): *Model-based Imitation Learning by Probabilistic Trajectory Matching*
Deisenroth et al. (ICRA, 2014): *Multi-Task Policy Search for Robotics*
Kupcsik et al. (AIJ, 2017): *Model-based Contextual Policy Search for Data-Efficient Generalization of Robot Skills*

Summary (1)





- In robotics, data-efficient learning is critical
- Probabilistic, model-based RL approach
  - Reduce model bias
  - Unprecedented learning speed
  - Wide applicability

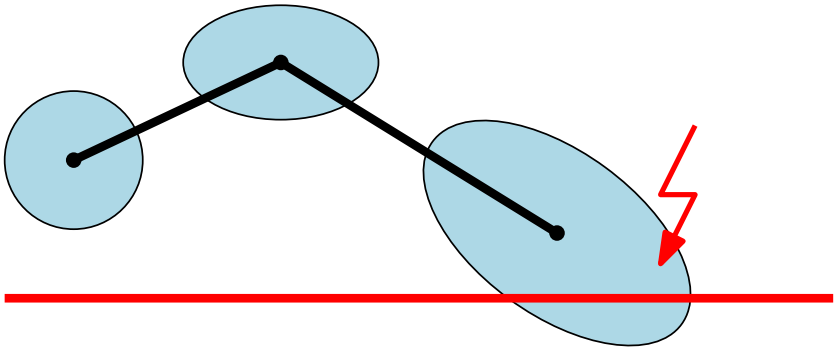
Overview

Model-based Reinforcement Learning

Safe Exploration

Meta Reinforcement Learning

Safe Exploration



- Deal with real-world safety constraints
- Use probabilistic model to predict whether constraints are violated (e.g., Sui et al., 2015; Berkenkamp et al., 2017)
- Adjust policy if necessary (during policy learning)

⏩ Safe exploration within an MPC-based RL setting

⏩ Optimize control signals $\boldsymbol{u}_t$ directly (no policy parameters)

Probabilistic MPC in RL

- GP model for transition dynamics
- Repeat (while executing the policy):
  1. In current state $\boldsymbol{x}_t$, determine optimal control sequence $\boldsymbol{u}_1^*, \ldots, \boldsymbol{u}_H^*$
  2. Apply first control $\boldsymbol{u}_1^*$ in state $\boldsymbol{x}_t$
  3. Transition to next state $\boldsymbol{x}_{t+1}$
  4. Update GP transition model

Kamthe & Deisenroth (AISTATS, 2018): *Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control*

Theoretical Results

- Uncertainty propagation is deterministic (GP moment matching)
  - ⏩ Re-formulate system dynamics:

$$\boldsymbol{z}_{t+1} = f_{MM}(\boldsymbol{z}_t, \boldsymbol{u}_t)$$

$$\boldsymbol{z}_t = \{\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t\} \quad \text{⏩ Collects moments}$$

Kamthe & Deisenroth (AISTATS, 2018): *Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control*

Theoretical Results

- Uncertainty propagation is deterministic (GP moment matching)
  - ⏩ Re-formulate system dynamics:

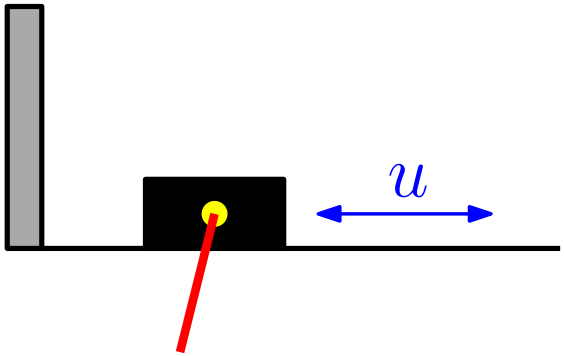
$$z_{t+1} = f_{MM}(z_t, u_t)$$

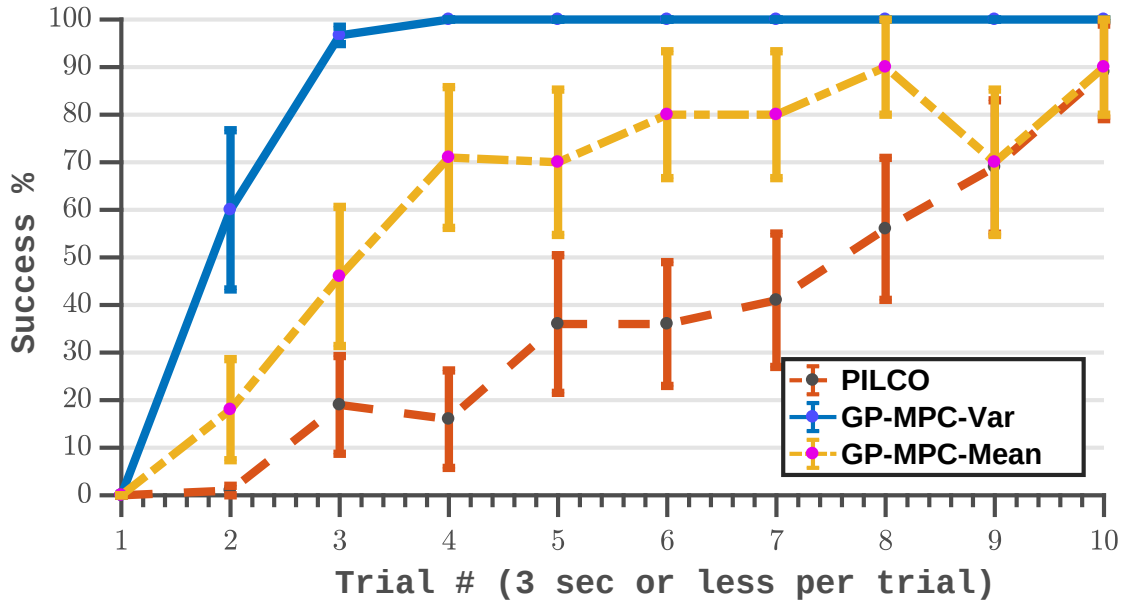
$$z_t = \{\mu_t, \Sigma_t\} \quad \text{⏩ Collects moments}$$

- Deterministic system function that propagates moments
- Lipschitz continuity (under mild assumptions) implies that we can apply Pontryagin's Minimum Principle
  - ⏩ Principled treatment of constraints on states and controls

Kamthe & Deisenroth (AISTATS, 2018): *Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control*

Experimental Results: Safety Constraints



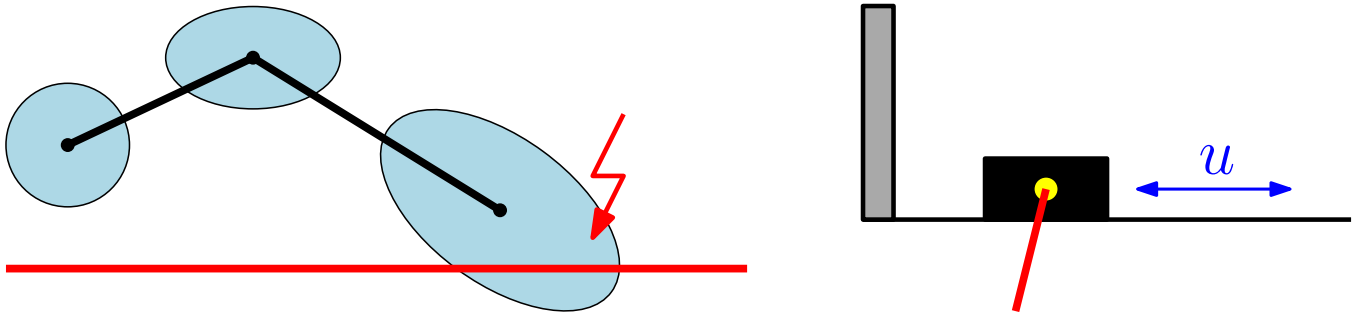


PILCO	16/100	constraint violations
GP-MPC-Mean	21/100	constraint violations
GP-MPC-Var	3/100	constraint violations

- **Propagating model uncertainty important for safety**

Kamthe & Deisenroth (AISTATS, 2018): *Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control*

Summary (2)



- Probabilistic prediction models for safe exploration
- Uncertainty propagation reduces violation of safety constraints
- MPC framework increases robustness to model errors
  - ⏩ Increased data efficiency

Overview

Model-based Reinforcement Learning

Safe Exploration

Meta Reinforcement Learning

Meta and Transfer Learning




- Objective: Learn predictive models (and controllers) for different robot arms

Meta and Transfer Learning




- Objective: Learn predictive models (and controllers) for different robot arms
- Overall the dynamics should not be too dissimilar
  - ⏩ Share some global properties

Meta and Transfer Learning




- Objective: Learn predictive models (and controllers) for different robot arms
- Overall the dynamics should not be too dissimilar
  - ⏩ Share some global properties
- Sightly different configurations (e.g., mass/link length)
  - ⏩ Differ locally

Meta and Transfer Learning



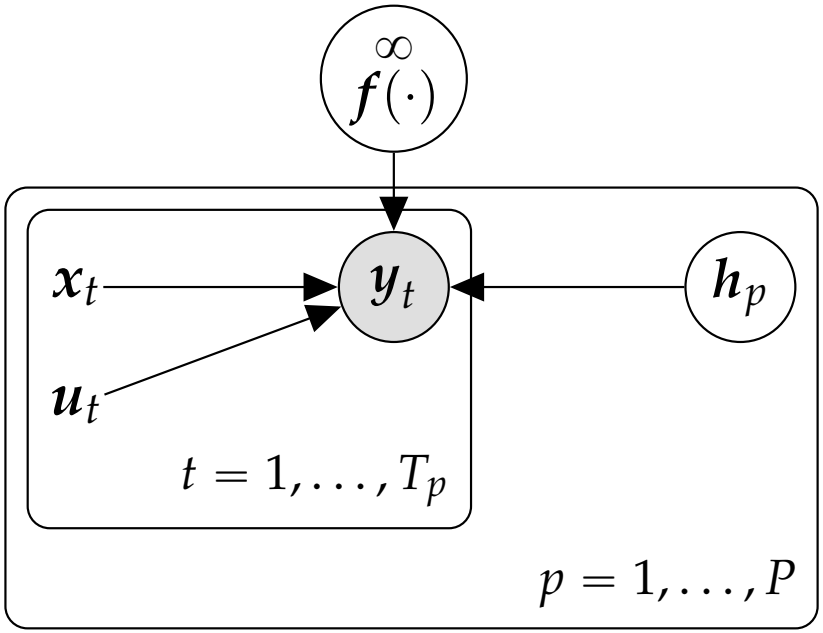

- Objective: Learn predictive models (and controllers) for different robot arms
- Overall the dynamics should not be too dissimilar
  - ⏩ Share some global properties
- Sightly different configurations (e.g., mass/link length)
  - ⏩ Differ locally
- Re-use experience gathered so far generalize learning to new dynamics that are similar
  - ⏩ **Accelerated learning**

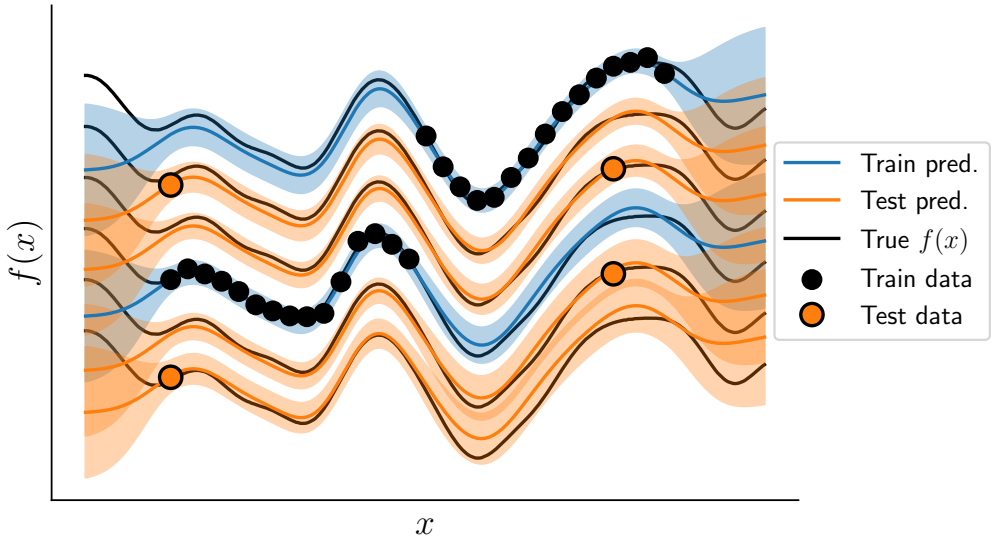
Approach

- Model (unknown) configurations with latent variable
- Separate global and task-specific properties
- Online inference of models of unseen configurations
- Few-shot model-based RL

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

Meta Model Learning with Latent Variables

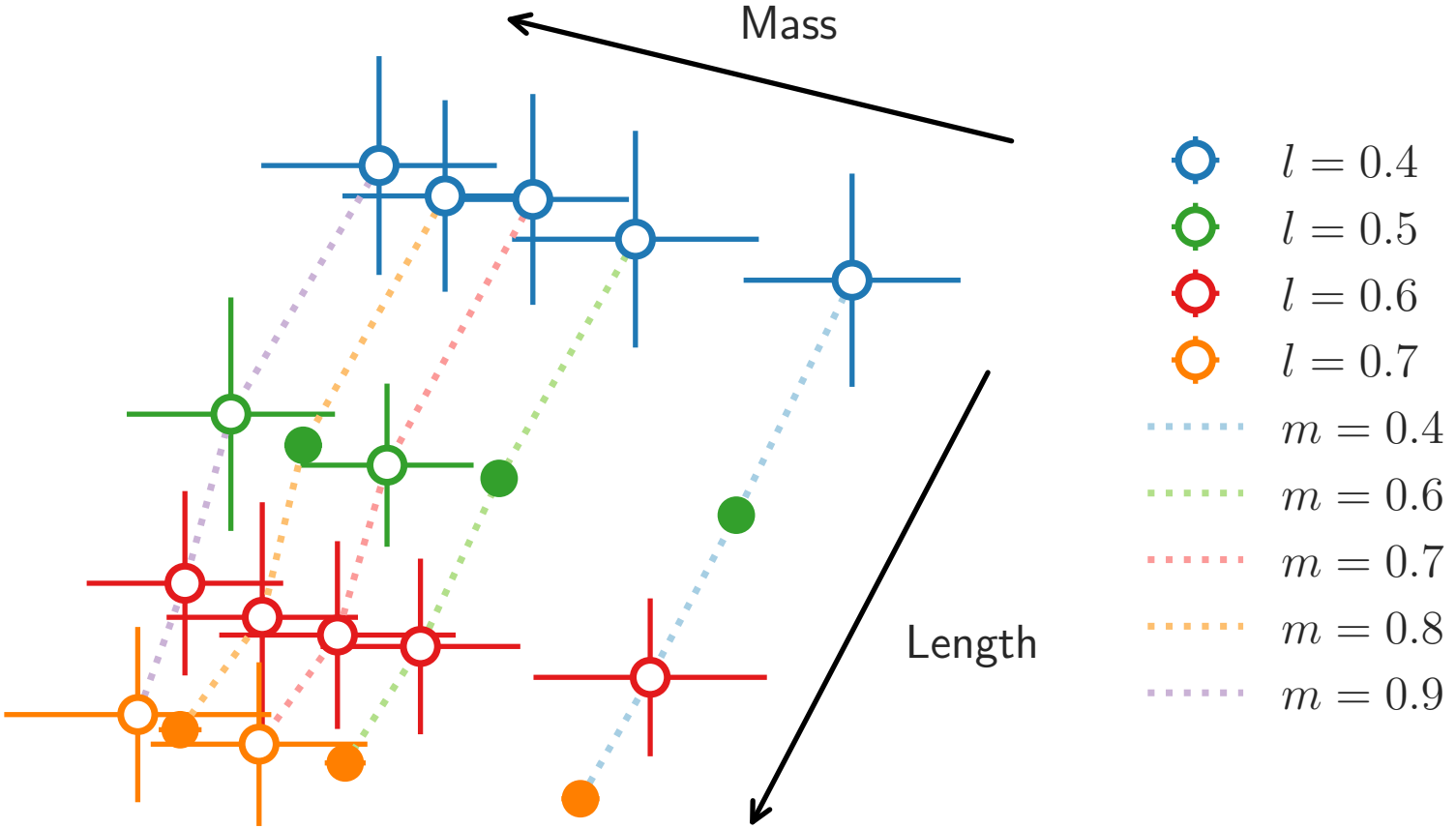




- GP captures global properties of the dynamics
- Latent variable h_p describes local configuration
  - ⏩ Variational inference to find a posterior on latent configuration
- Fast online inference of new configurations (no model re-training required)

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

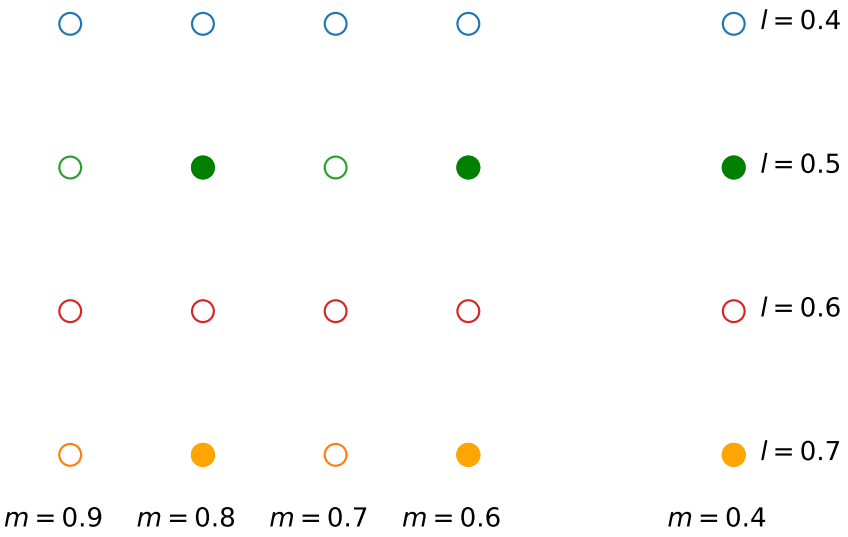
Probabilistic Latent Embeddings

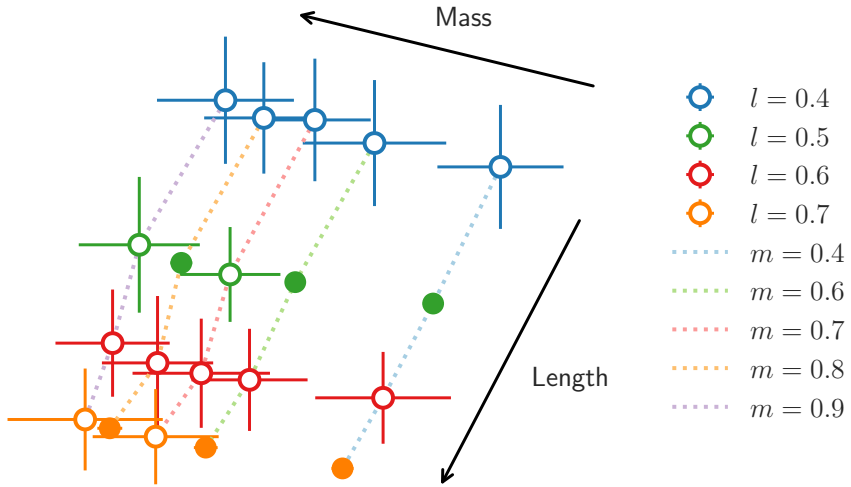


- Latent variable $\boldsymbol{h}$ encodes length l and mass m of the cart pole
- 6 training tasks, 14 held-out test tasks

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

Latent Embeddings (2)

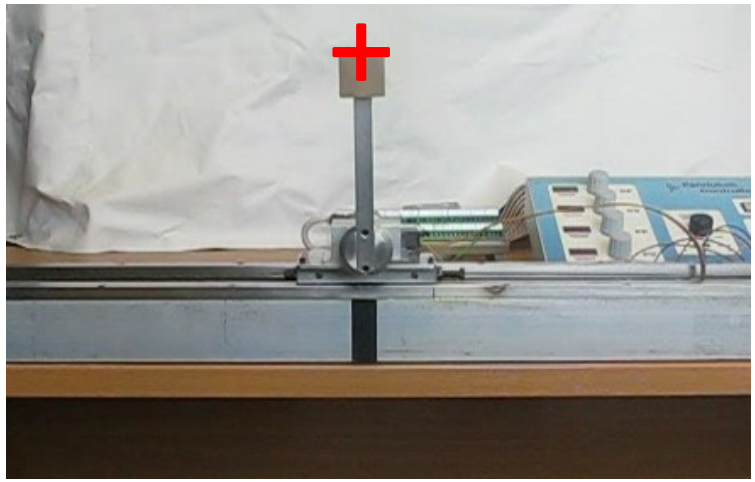




- Latent variable $\boldsymbol{h}$ encodes length l and mass m of the cart pole
- 6 training tasks, 14 held-out test tasks
- Left: True configurations; Right: corresponding embeddings

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

Meta-RL (Cart Pole): Training



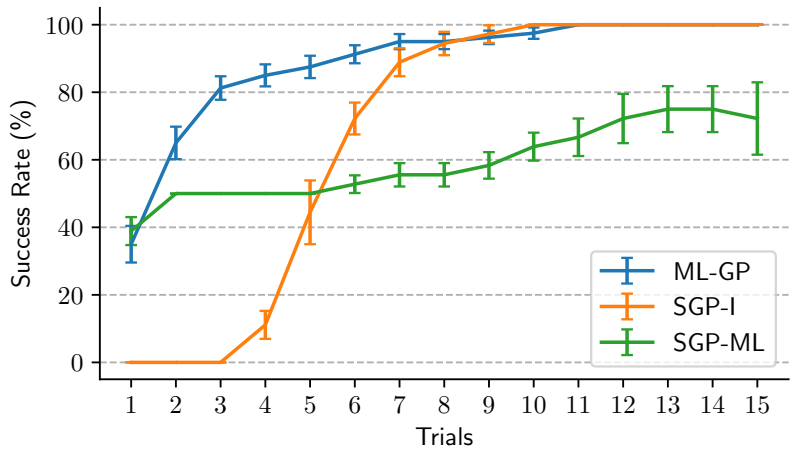
- Pre-trained on 6 training configurations until solved

Model	Training (s)	Description
Independent	16.1 ± 0.4	Independent GP-MPC
Aggregated	23.7 ± 1.4	Aggregated experience (no latents)
Meta learning	$\mathbf{15.1 \pm 0.5}$	Aggregated experience (with latents)

⏩ **Meta learning can help speeding up RL**

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

Meta-RL (Cart Pole): Few-Shot Generalization

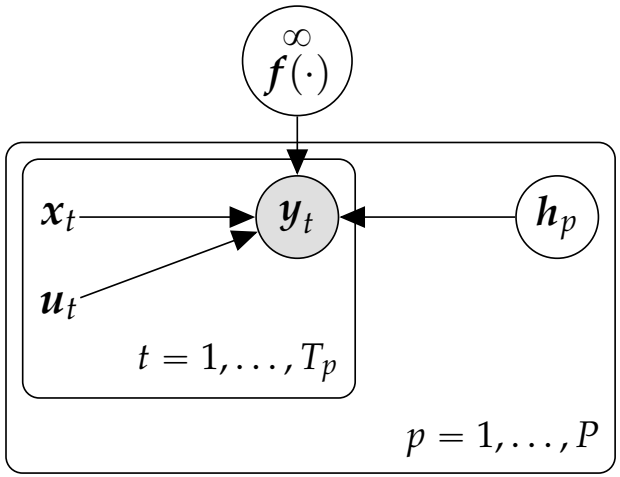


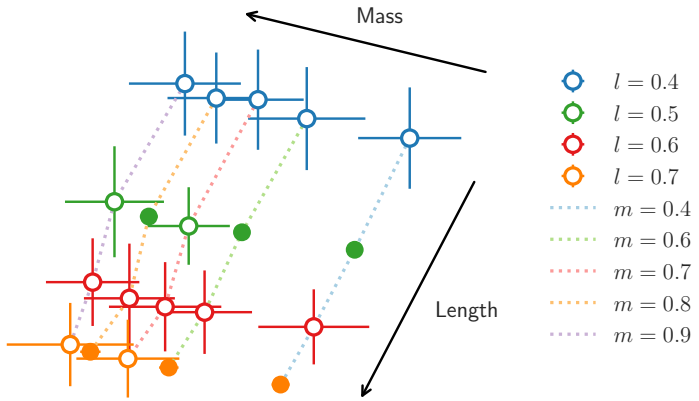
- Few-shot generalization on 4 unseen configurations
- Success: solve all 10 (6 training + 4 test) tasks
- Meta learning
- Independent (GP-MPC)
- Aggregated experience model (no latents)

⏩ Meta RL generalizes well to unseen tasks

Sæmundsson et al. (UAI, 2018): *Meta Reinforcement Learning with Latent Variable Gaussian Processes*

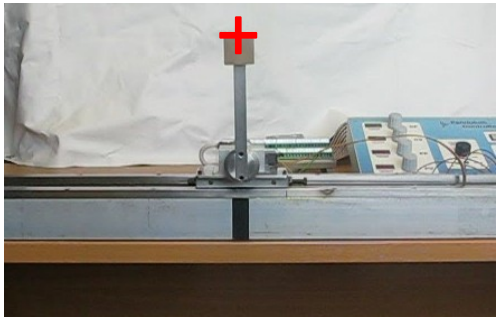
Summary (3)

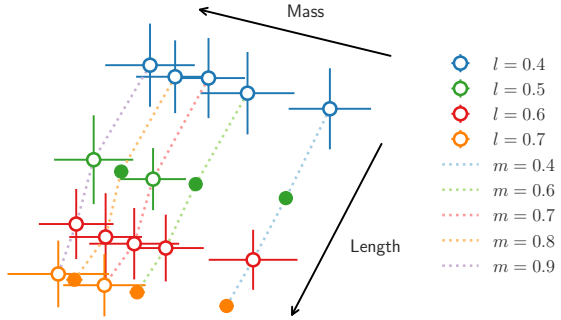




- Generalize knowledge from known task to unseen ones
- Latent variable describes how related tasks are
- Variational inference to get a posterior on these latent variables
- Significant speed-up in model learning and model-based RL

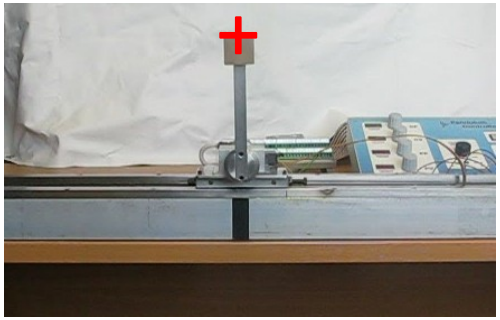
Wrap-up

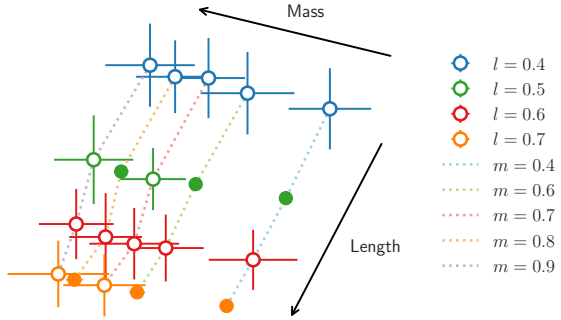




- Autonomous systems take humans out of the loop
- In robotics, data-efficient learning is critical
- Controller learning based on learned probabilistic models
  - Reinforcement learning
  - Safe exploration and MPC
  - Transfer and meta learning to generalize learned concepts to new situations
- Key to success: Probabilistic modeling and Bayesian inference

Wrap-up





- Autonomous systems take humans out of the loop
- In robotics, data-efficient learning is critical
- Controller learning based on learned probabilistic models
  - Reinforcement learning
  - Safe exploration and MPC
  - Transfer and meta learning to generalize learned concepts to new situations
- Key to success: Probabilistic modeling and Bayesian inference

Thank you for your attention

References I

[1] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause. Safe Model-based Reinforcement Learning with Stability Guarantees. In *Advances in Neural Information Processing Systems*, 2017.

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

[3] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2 of *Optimization and Computation Series*. Athena Scientific, Belmont, MA, USA, 3rd edition, 2007.

[4] 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.

[5] M. P. Deisenroth, P. Englert, J. Peters, and D. Fox. Multi-Task Policy Search for Robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2014.

[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*, 37(2):408–423, 2015.

[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*, 2011.

[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] P. Englert, A. Paraschos, J. Peters, and M. P. Deisenroth. Model-based Imitation Learning by Probabilistic Trajectory Matching. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2013.

[10] P. Englert, A. Paraschos, J. Peters, and M. P. Deisenroth. Probabilistic Model-based Imitation Learning. *Adaptive Behavior*, 21:388–403, 2013.

[11] A. Girard, C. E. Rasmussen, and R. Murray-Smith. Gaussian Process Priors with Uncertain Inputs: Multiple-Step Ahead Prediction. Technical Report TR-2002-119, University of Glasgow, 2002.

[12] S. Kamthe and M. P. Deisenroth. Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2018.

References II

[13] A. Kupcsik, M. P. Deisenroth, J. Peters, L. A. Poha, P. Vadakkepata, and G. Neumann. Model-based Contextual Policy Search for Data-Efficient Generalization of Robot Skills. *Artificial Intelligence*, 2017.

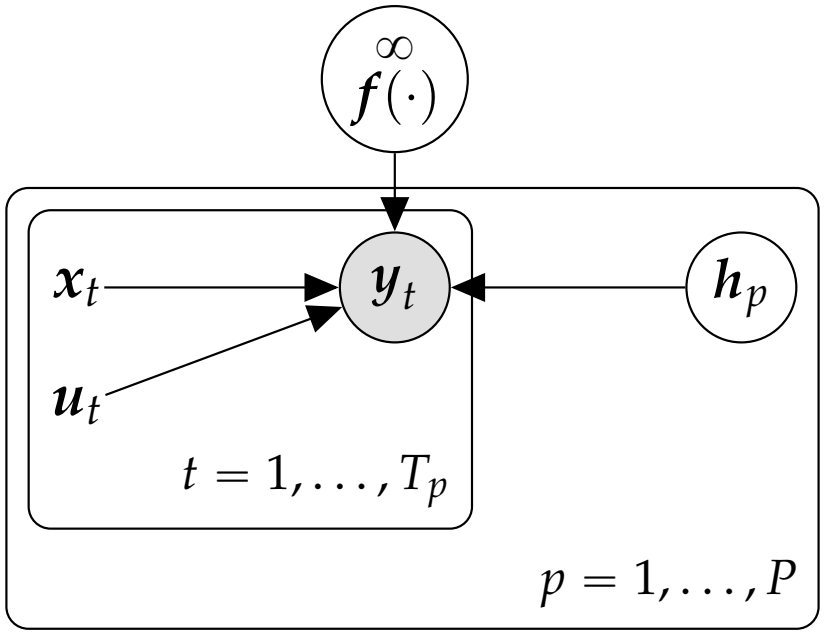
[14] J. Quiñonero-Candela, A. Girard, J. Larsen, and C. E. Rasmussen. Propagation of Uncertainty in Bayesian Kernel Models—Application to Multiple-Step Ahead Forecasting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 701–704, Apr. 2003.

[15] 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.

[16] S. Sæmundsson, K. Hofmann, and M. P. Deisenroth. Meta Reinforcement Learning with Latent Variable Gaussian Processes. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2018.

[17] Y. Sui, A. Gotovos, J. W. Burdick, and A. Krause. Safe Exploration for Optimization with Gaussian Processes. In *Proceedings of the International Conference on Machine Learning*, 2015.

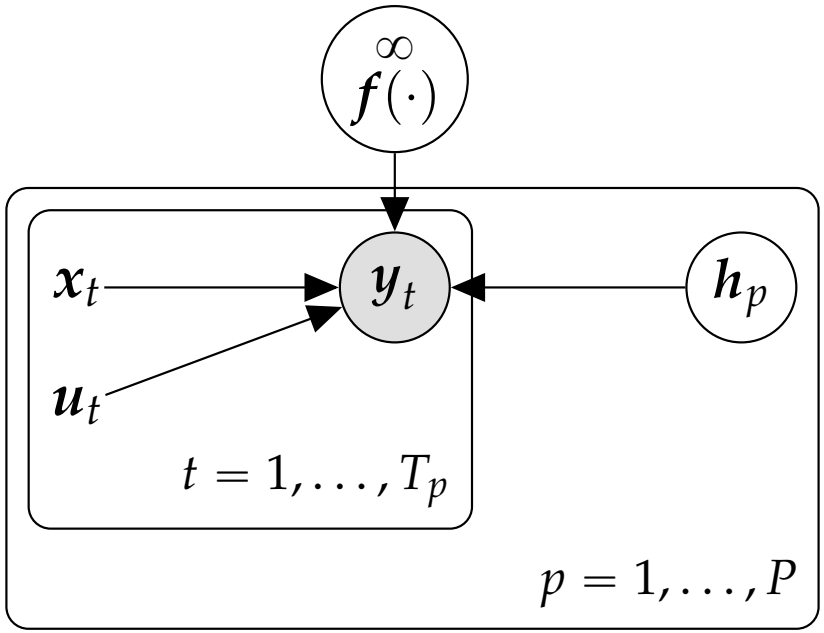
Meta-Learning Model



$$f(\cdot) \sim GP$$

$$p(\boldsymbol{H}) = \prod_p p(\boldsymbol{h}_p), \quad p(\boldsymbol{h}_p) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Meta-Learning Model



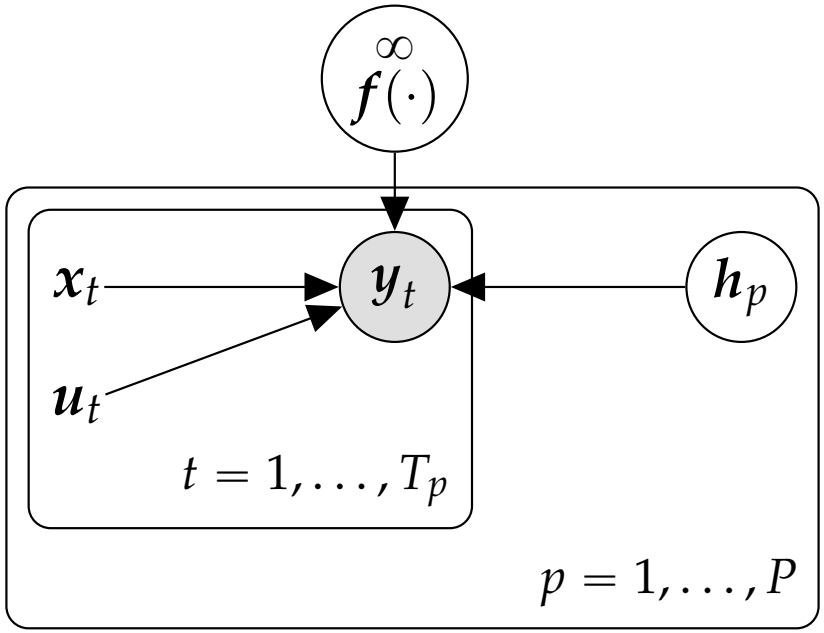
$$f(\cdot) \sim GP$$

$$p(\boldsymbol{H}) = \prod_p p(\boldsymbol{h}_p)\,, \quad p(\boldsymbol{h}_p) = \mathcal{N}\left(\boldsymbol{0},\, \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y}, \boldsymbol{H}, \boldsymbol{f}(\cdot)|\boldsymbol{X}, \boldsymbol{U}) = \prod_{p=1}^{P} p(\boldsymbol{h}_p) \prod_{t=1}^{T_p} p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{h}_p, \boldsymbol{f}(\cdot)) p(\boldsymbol{f}(\cdot))$$

$$\boldsymbol{y}_t = \boldsymbol{x}_{t+1} - \boldsymbol{x}_t$$

Variational Inference



Mean-field variational family:

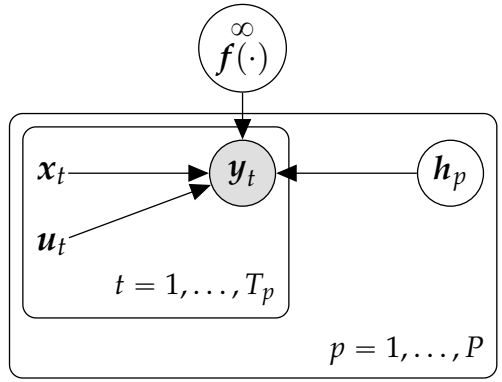
$$q(\boldsymbol{f}(\cdot), \boldsymbol{H}) = q(\boldsymbol{f}(\cdot))q(\boldsymbol{H})$$

$$q(\boldsymbol{H}) = \prod_{p=1}^{P} \mathcal{N}(\boldsymbol{h}_p | \boldsymbol{n}_p, \boldsymbol{T}_p)\,,$$

$$q(\boldsymbol{f}(\cdot)) = \int p(\boldsymbol{f}(\cdot)|\boldsymbol{f}_Z) q(\boldsymbol{f}_Z) d\boldsymbol{f}_Z$$ ⏩ SV-GP (Titsias, 2009)

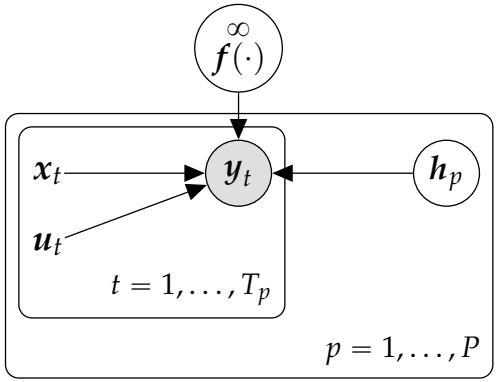
Evidence Lower Bound

$$ELBO = \mathbb{E}_{q(\boldsymbol{f}(\cdot),\boldsymbol{H})}\left[\log \frac{p(\boldsymbol{Y},\boldsymbol{H},\boldsymbol{f}(\cdot)|\boldsymbol{X},\boldsymbol{U})}{q(\boldsymbol{f}(\cdot),\boldsymbol{H})}\right]$$



Evidence Lower Bound

$$
\begin{aligned}
ELBO &= \mathbb{E}_{q(f(\cdot),\boldsymbol{H})}\left[\log \frac{p(\boldsymbol{Y},\boldsymbol{H},f(\cdot)|\boldsymbol{X},\boldsymbol{U})}{q(f(\cdot),\boldsymbol{H})}\right] \\
&= \sum_{p=1}^{P}\sum_{t=1}^{T_p} \mathbb{E}_{q(f_t|\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{h}_p)q(\boldsymbol{h}_p)}\left[\log p(\boldsymbol{y}_t|f_t)\right] \\
&\quad - \mathrm{KL}\left(q(\boldsymbol{H})||p(\boldsymbol{H})\right) - \mathrm{KL}\left(q(f(\cdot))||p(f(\cdot))\right)
\end{aligned}
$$



Evidence Lower Bound

$$
\begin{aligned}
ELBO &= \mathbb{E}_{q(f(\cdot),\boldsymbol{H})}\left[\log \frac{p(\boldsymbol{Y},\boldsymbol{H},f(\cdot)|\boldsymbol{X},\boldsymbol{U})}{q(f(\cdot),\boldsymbol{H})}\right] \\
&= \sum_{p=1}^{P}\sum_{t=1}^{T_p} \mathbb{E}_{q(f_t|\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{h}_p)q(\boldsymbol{h}_p)}\left[\log p(\boldsymbol{y}_t|f_t)\right] \\
&\quad - \mathrm{KL}\left(q(\boldsymbol{H})||p(\boldsymbol{H})\right) - \mathrm{KL}\left(q(f(\cdot))||p(f(\cdot))\right) \\
&= \sum_{p=1}^{P}\sum_{t=1}^{T_p} \overbrace{\mathbb{E}_{q(f_t|\boldsymbol{x}_t,\boldsymbol{u}_t,\boldsymbol{h}_p)q(\boldsymbol{h}_p)}\left[\log p(\boldsymbol{y}_t|f_t)\right]}^{\text{Monte Carlo estimate}} \\
&\quad \underbrace{-\mathrm{KL}\left(q(\boldsymbol{H})||p(\boldsymbol{H})\right) - \mathrm{KL}\left(q(\boldsymbol{F}_Z)||p(\boldsymbol{F}_Z)\right)}_{\text{closed-form solution}}
\end{aligned}
$$