Good Practices using Machine Learning

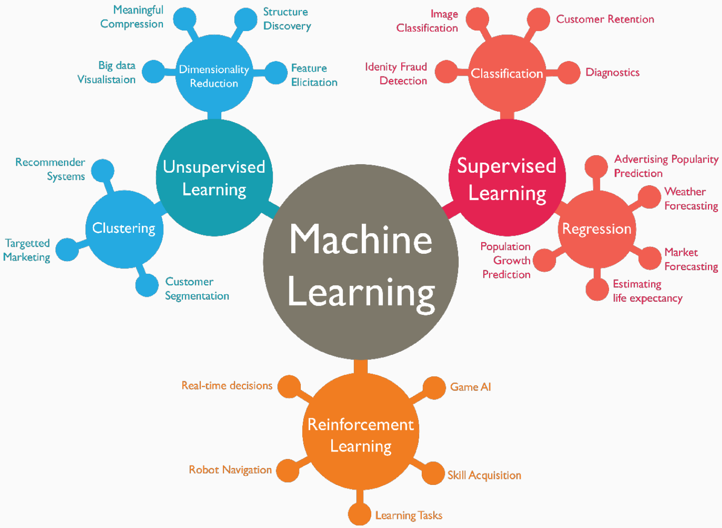
Carl Henrik Ek - che29@cam.ac.uk

September 29, 2021

http://carlhenrik.com

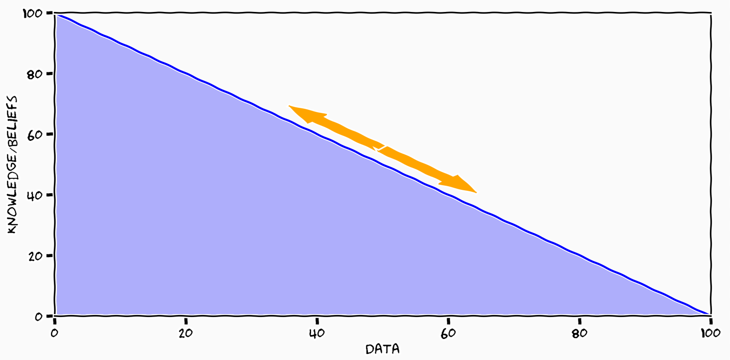
Introduction

Map of ML


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Handle

Pareto



Paradigms

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Active Learning
- Meta Learning

Reinforcement Learning

Machine Learning

Supervised Learning predict output from input

$$\mathcal{D} = \{y_i, x_i\}_{i=1}^{N}$$

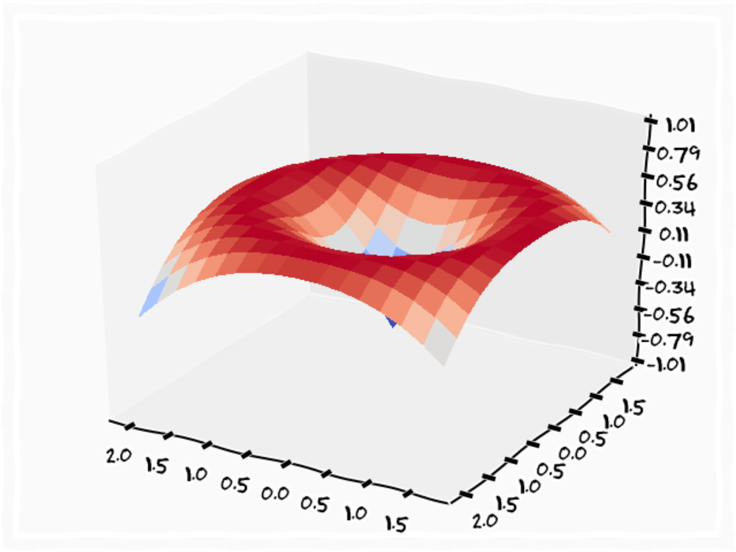
$$p(\mathbf{y}|\mathbf{x}, \theta)$$

Unsupervised Learning model the data

$$\mathcal{D} = \{y_i\}_{i=1}^{N}$$

$$p(\mathbf{y}|\theta)$$

Structure


1.01
0.79
0.56
0.34
0.11
-0.11
-0.34
-0.56
-0.79
-1.01
2.0
1.5
1.0
0.5
0.0
0.5
1.0
1.5
2.0
1.5
1.0
0.5
0.0
0.5
1.0
1.5

Reinforcement

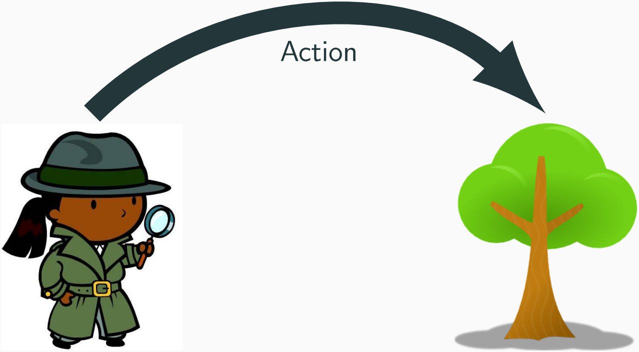
Reinforcement Learning

Can we learn without specifying how the task should be achieved by providing, rewards (positive) and punishment (negative)?

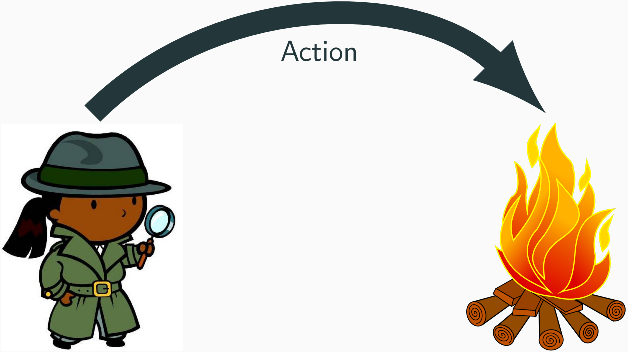
Formalism

Formalism

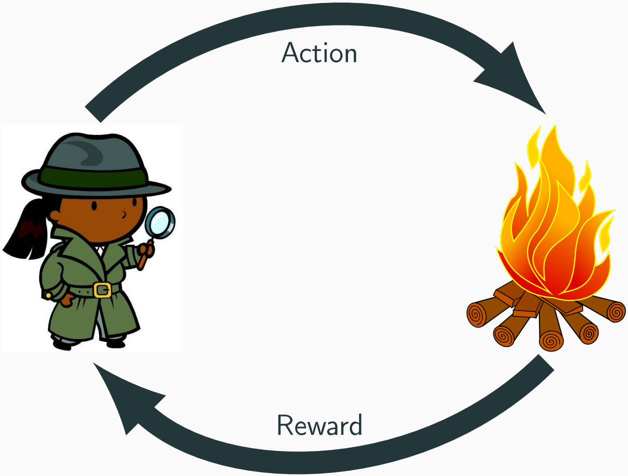
Formalism



Formalism



Formalism



Optimal Behaviour

- Finite time horizon

$$E\left[\sum_{t=0}^{h} r_t\right]$$

Optimal Behaviour

- Finite time horizon

$$E\left[\sum_{t=0}^{h} r_t\right]$$

- Average reward

$$\lim_{h\to\infty} E\left[\frac{1}{h}\sum_{t=0}^{h} r_t\right]$$

Optimal Behaviour

- Finite time horizon

$$E\left[\sum_{t=0}^{h} r_t\right]$$

- Average reward

$$\lim_{h\to\infty} E\left[\frac{1}{h}\sum_{t=0}^{h} r_t\right]$$

- Infinite horizon (discounted reward)

$$E\left[\sum_{t=0}^{\infty} \gamma^t r_t\right]$$

$$0 \leq \gamma \leq 1$$

Reinforcement Learning

- Dynamic model

$$p(\mathbf{s}_t|\mathbf{a}_{t-1}, \mathbf{s}_{t-1}, f)$$

- Policy

$$p(\mathbf{a}_t|\mathbf{s}_t, \pi)$$

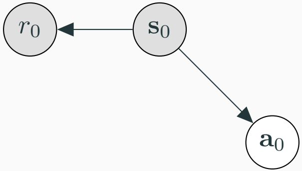
- Reward

$$p(r_t|\mathbf{s}_t)$$

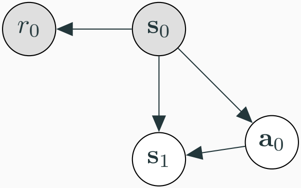
Reinforcement Learning

Reinforcement Learning

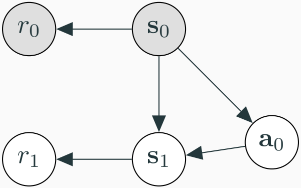
Reinforcement Learning



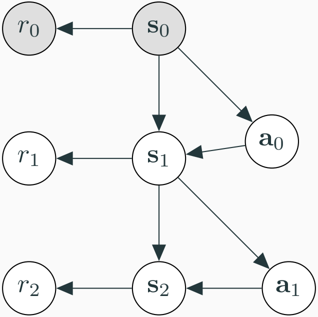
Reinforcement Learning

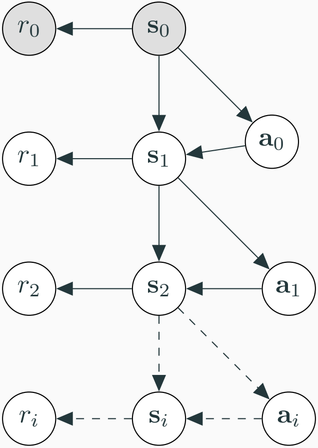


Reinforcement Learning

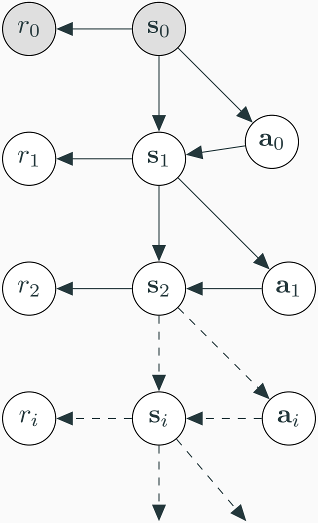


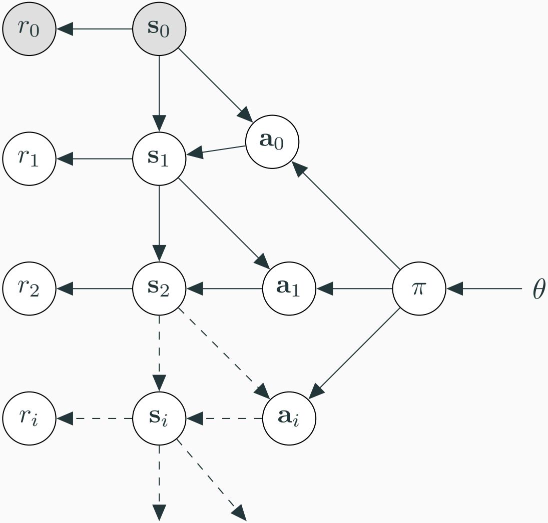
Reinforcement Learning

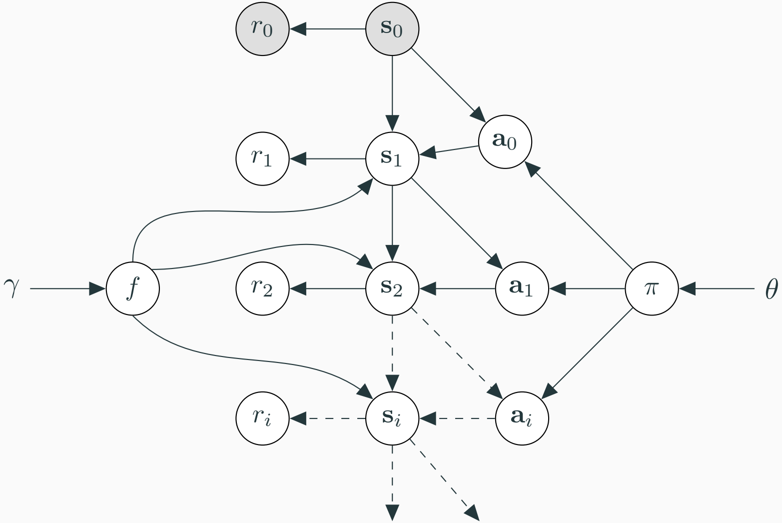



r_0
$\mathbf{s}_0$
$\mathbf{a}_0$
r_1
$\mathbf{s}_1$
r_2
$\mathbf{s}_2$
$\mathbf{a}_1$
r_i
$\mathbf{s}_i$
$\mathbf{a}_i$

Reinforcement Learning




r_0
$\mathbf{s}_0$
r_1
$\mathbf{s}_1$
$\mathbf{a}_0$
r_2
$\mathbf{s}_2$
$\mathbf{a}_1$
π
θ
r_i
$\mathbf{s}_i$
$\mathbf{a}_i$


r_0
$\mathbf{s}_0$
$\mathbf{a}_0$
r_1
$\mathbf{s}_1$
γ
f
r_2
$\mathbf{s}_2$
$\mathbf{a}_1$
π
θ
r_i
$\mathbf{s}_i$
$\mathbf{a}_i$

Reinforcment Learning : Rollout

$$
\begin{aligned}
p(\mathbf{s}_{0,\ldots,T}, \mathbf{a}_{0,\ldots,T-1}, r_{0,\ldots,T}, f, \pi, \gamma, \theta | \mathbf{s}_0) = \\
& p(\mathbf{s}_T | a_{T-1}, s_{T-1}, f) p(a_{T-1} | s_{T-1}, \pi) \ldots \\
& \qquad \vdots \\
& p(\mathbf{s}_2 | a_1, s_1, f) p(a_1 | s_1, \pi) \\
& p(\mathbf{s}_1 | a_0, s_0, f) p(a_0 | s_0, \pi) \\
& p(f | \gamma) p(\pi | \theta) p(\gamma) p(\theta)
\end{aligned}
$$

- if we want to learn dynamics and policy we need to marginalise out f, π, θ and γ
- very very hard problem as uncertainty propagated a long way

Reinforcement Learning

- Is not a model but an inference structure

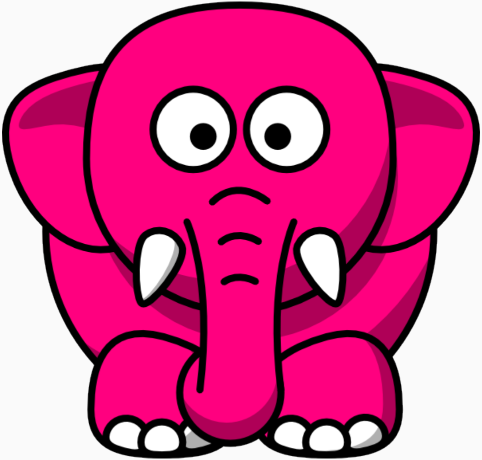
Reinforcement Learning

- Is not a model but an inference structure
- The data capture should be part of the model

Reinforcement Learning

- Is not a model but an inference structure
- The data capture should be part of the model
    - Exploration vs. Exploitation

Neural Networks

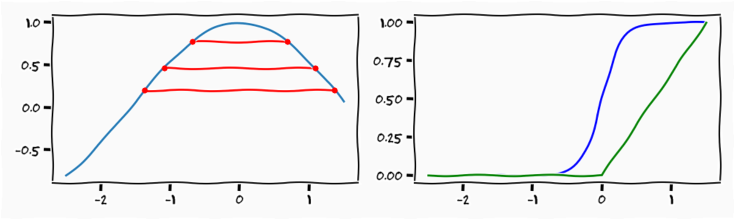


Neural Networks

$$x \to f_3 \to f_2 \to f_1 \to y$$

$$y = f_k(f_{k-1}(\dots f_0(x))) = f_k \circ f_{k-1} \circ \dots \circ f_1(x)$$

Composite Functions



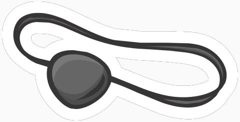
$$y = f_k(f_{k-1}(\dots f_0(x))) = f_k \circ f_{k-1} \circ \dots \circ f_1(x)$$

$$\mathsf{Kern}(f_1) \subseteq \mathsf{Kern}(f_{k-1} \circ \dots \circ f_2 \circ f_1) \subseteq \mathsf{Kern}(f_k \circ f_{k-1} \circ \dots \circ f_2 \circ f_1)$$

$$\mathsf{Im}(f_k \circ f_{k-1} \circ \dots \circ f_2 \circ f_1) \subseteq \mathsf{Im}(f_k \circ f_{k-1} \circ \dots \circ f_2) \subseteq \dots \subseteq \mathsf{Im}(f_k)$$

A piece of string

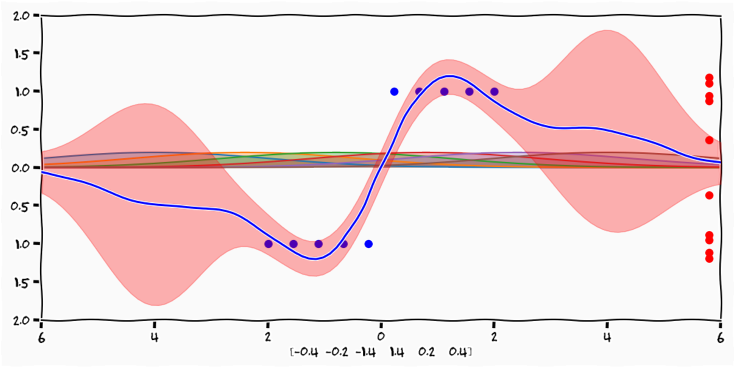
Assumptions: Algorithms

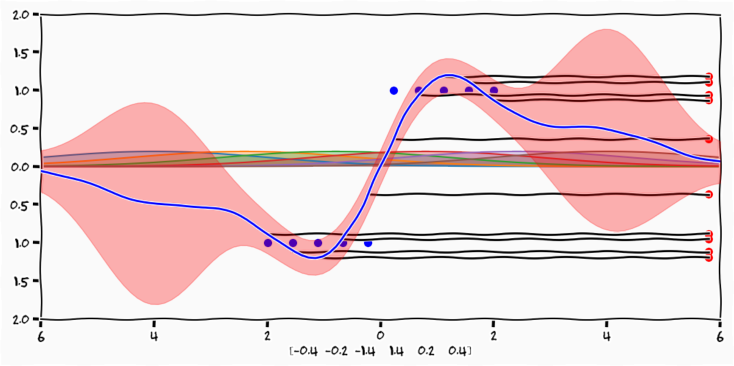


Statistical Learning

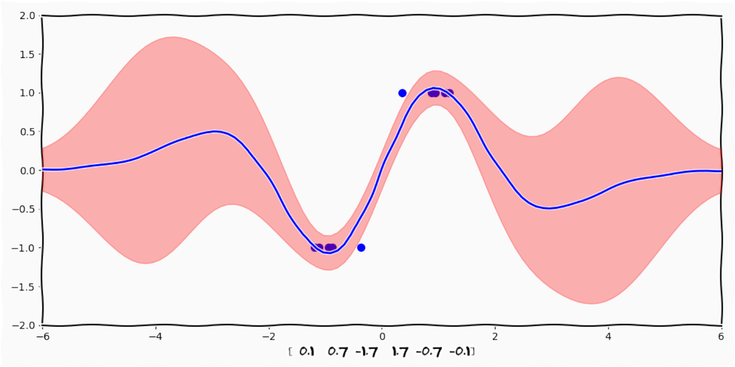
$$\color{red}{\mathcal{A}}_{\mathcal{F}}(\mathcal{S})$$

Composite Functions

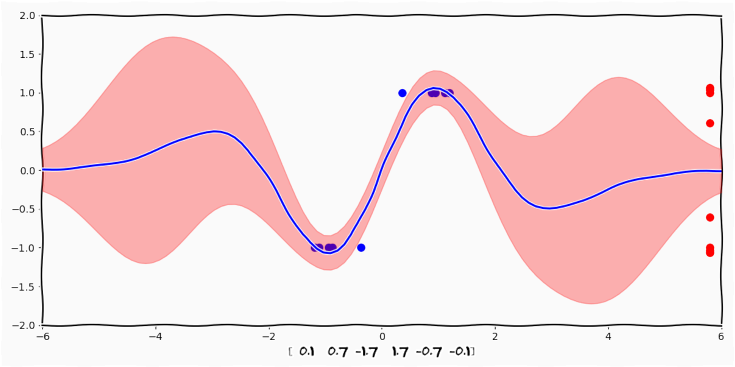



2.0
1.5
1.0
0.5
0.0
0.5
1.0
1.5
2.0
6
4
2
0
2
4
6
[-0.4 -0.2 -1.4 1.4 0.2 0.4]

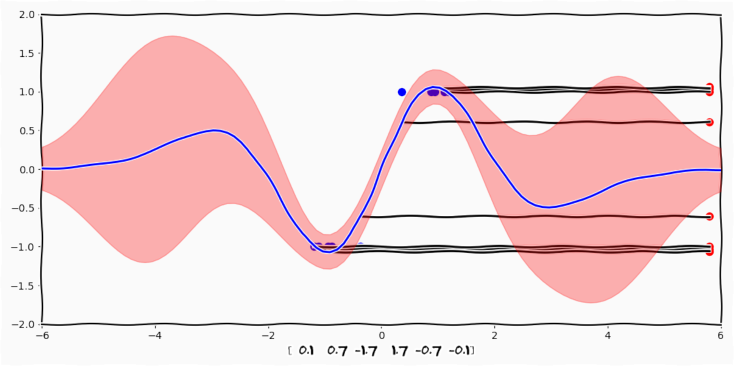
Composite Functions



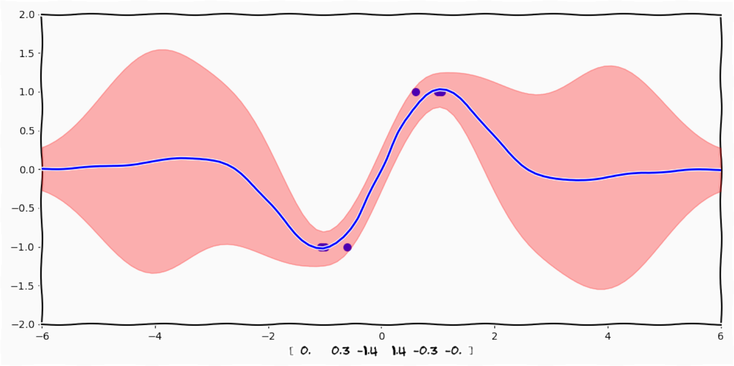
Composite Functions



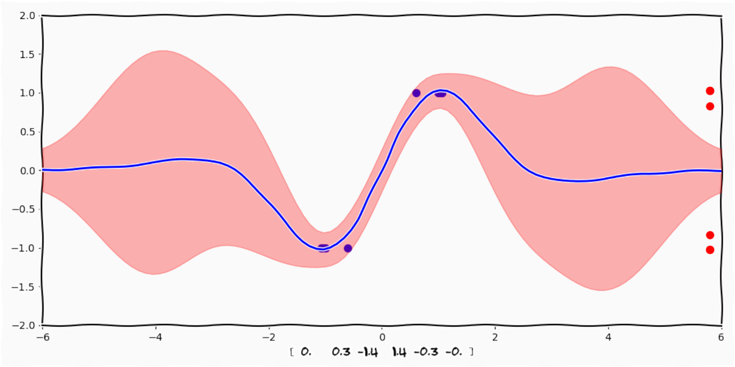
Composite Functions



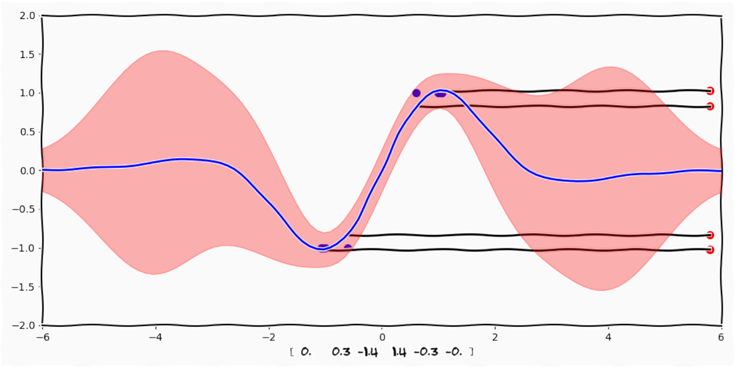
Composite Functions



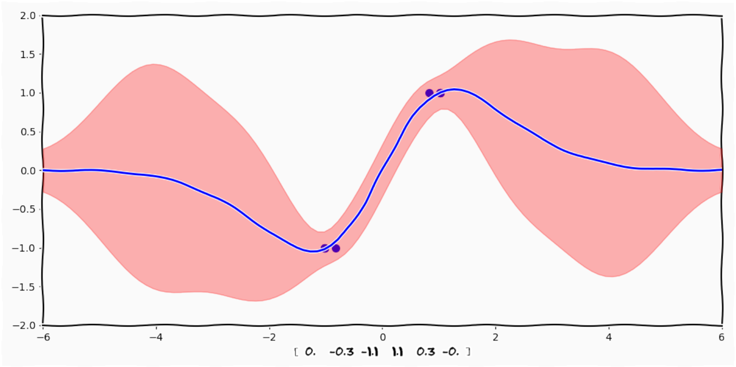
Composite Functions



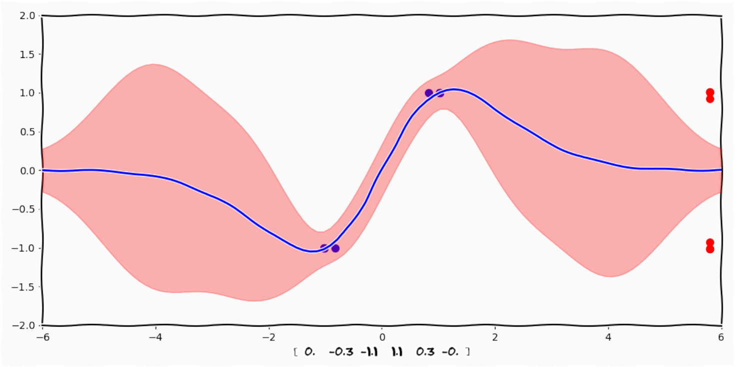
Composite Functions



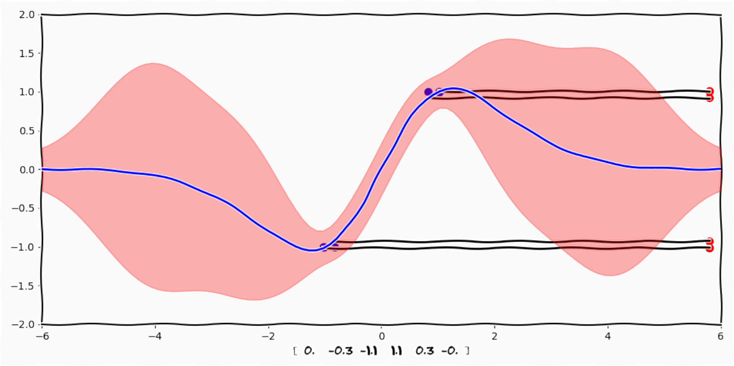
Composite Functions



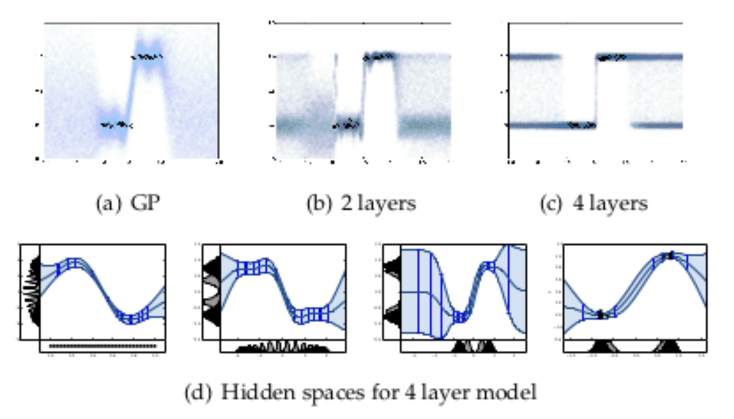
Composite Functions



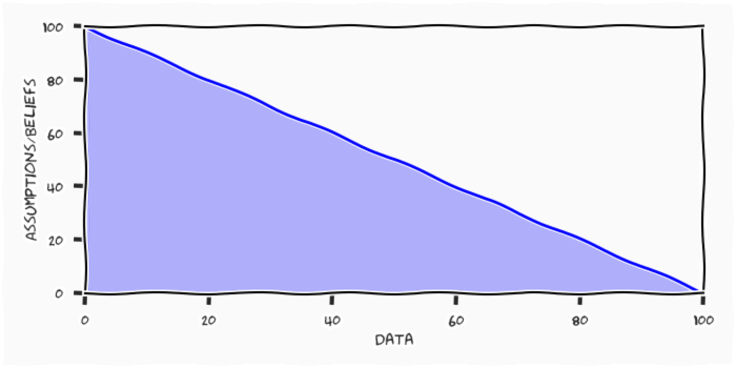
Composite Functions



The Final Composition



Data and Beliefs



Active Learning

Black-box Optimisation

$$x^{(*)} = \operatorname*{argmin}_{x \in \mathcal{X}} f(x)$$

- $\mathcal{X}$ is a bounded domain

Black-box Optimisation

$$x^{(*)} = \operatorname*{argmin}_{x \in \mathcal{X}} f(x)$$

- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown

Black-box Optimisation

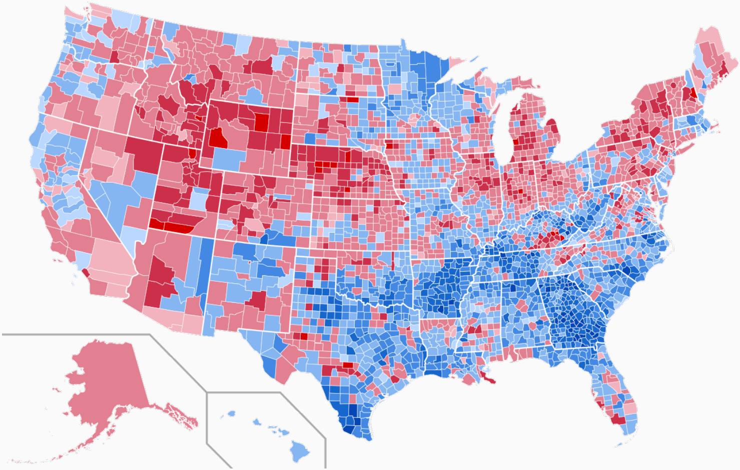
$$x^{(*)} = \operatorname*{argmin}_{x \in \mathcal{X}} f(x)$$

- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown
- Evaluations of f may be noisy

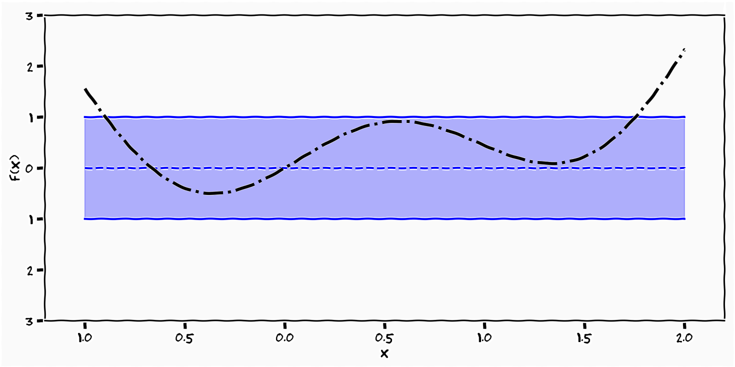
Black-box Optimisation

$$x^{(*)} = \operatorname*{argmin}_{x \in \mathcal{X}} f(x)$$

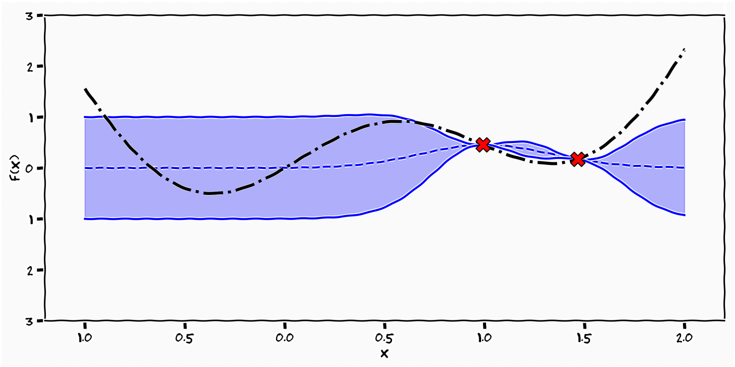
- $\mathcal{X}$ is a bounded domain
- f is explicitly unknown
- Evaluations of f may be noisy
- Evaluations of f is expensive



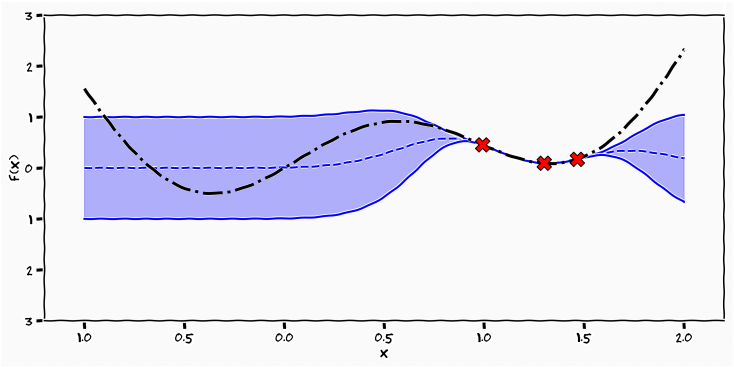
Posterior Search



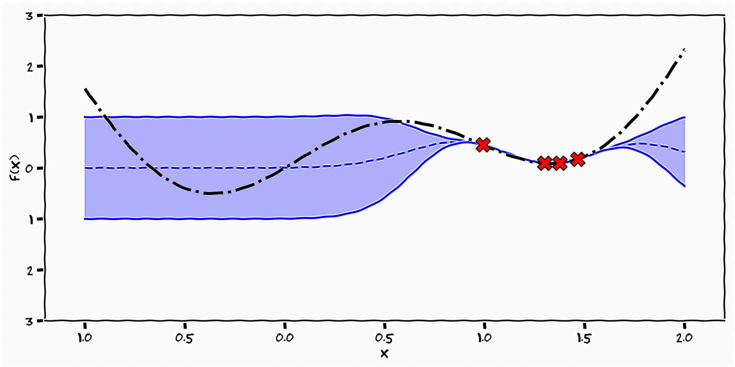
Posterior Search



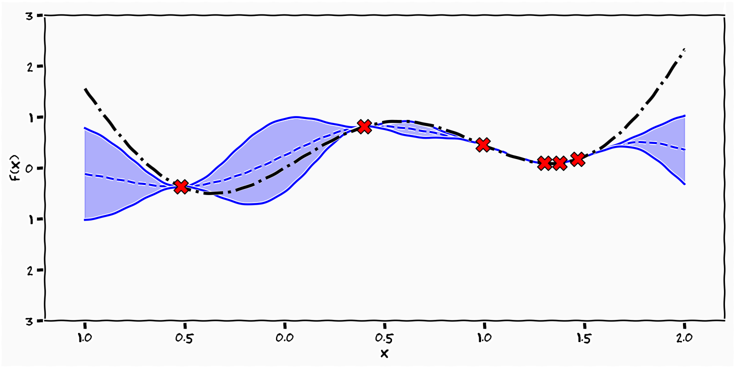
Posterior Search



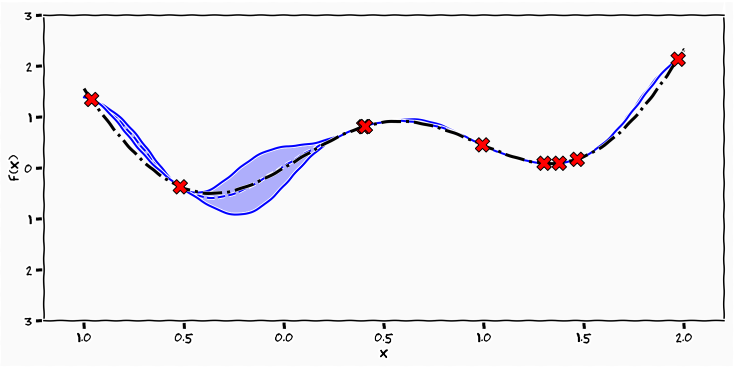
Posterior Search



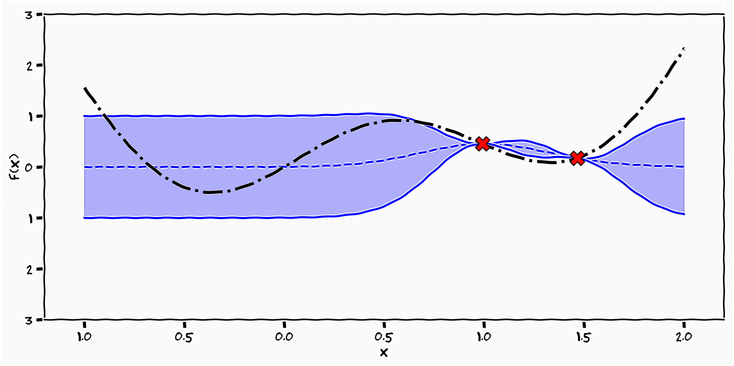
Posterior Search



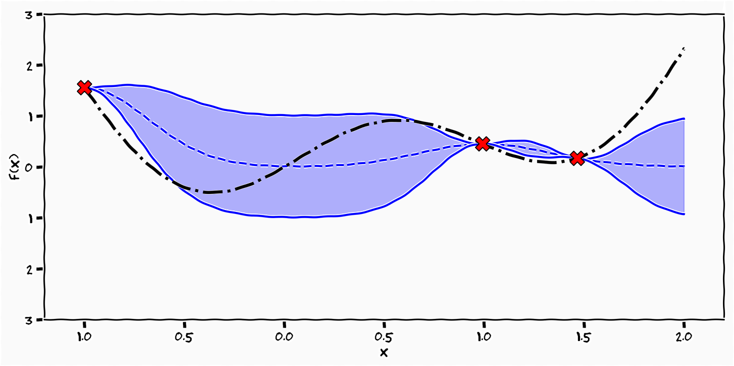
Posterior Search



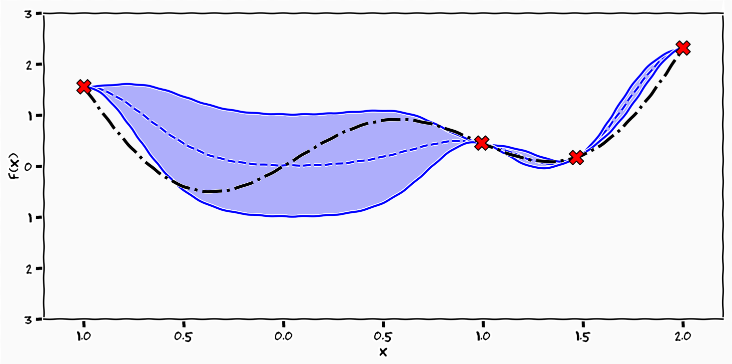
Posterior Search



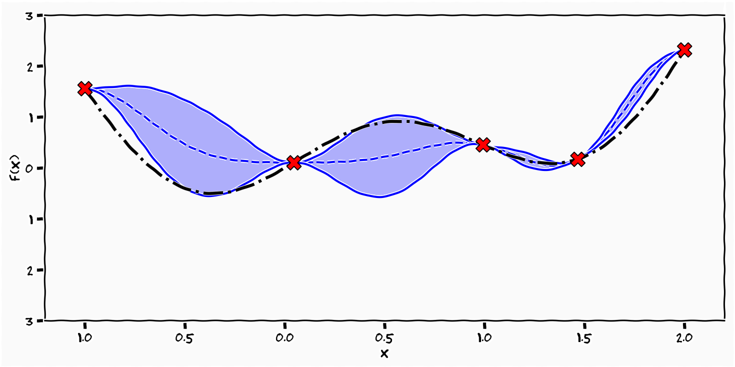
Posterior Search



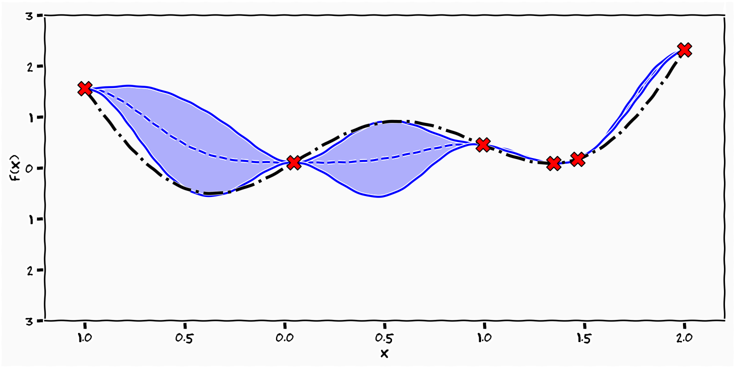
Posterior Search



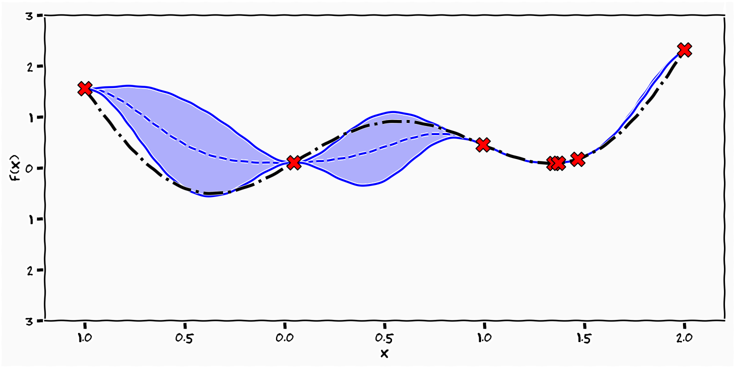
Posterior Search

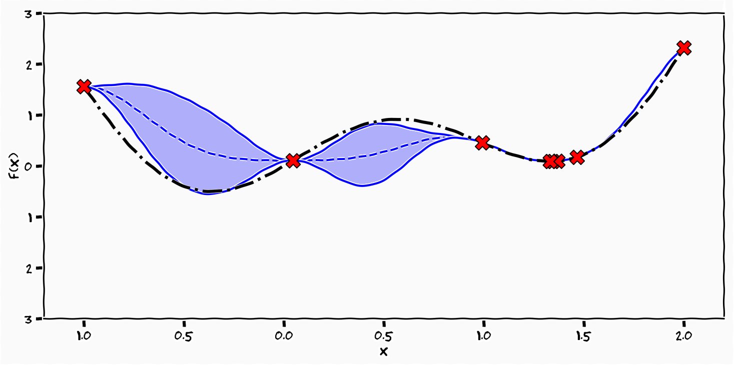


Posterior Search

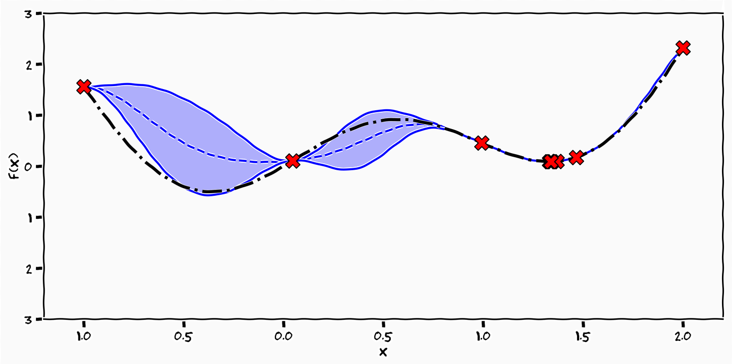


Posterior Search




3
2
1
0
1
2
3
f(x)
1.0
0.5
0.0
0.5
1.0
1.5
2.0
x

Posterior Search




STRONGARM
CAMERONS
TIMOTHY TAYLOR'S
LANDLORD
FURNACE
West Coast
Mosaic IPA

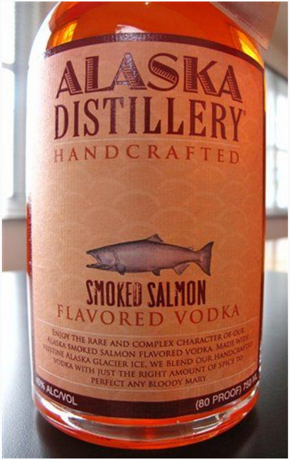

Coca-Cola


Coca-Cola

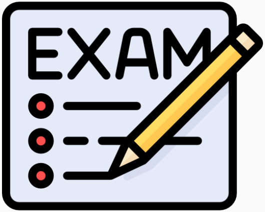

WYE VALLEY
BITTER
Superior Quality
ALC 3.7% VOL
WYE VALLEY
BREWERY
WYE VALLEY
BREWERY


Coca-Cola


WYE VALLEY
BITTER
Superior Quality
ALC 3.7% VOL
WYE VALLEY
BREWERY
WYE VALLEY
BREWERY


ALASKA
DISTILLERY
HANDCRAFTED
SMOKED SALMON
FLAVORED VODKA
ENJOY THE RARE AND COMPLEX CHARACTER OF OUR ALASKA SMOKED SALMON FLAVORED VODKA. MADE WITH PRISTINE ALASKA GLACIER ICE, WE BLEND OUR HANDCRAFTED VODKA WITH JUST THE RIGHT AMOUNT OF SPICE TO PERFECT ANY BLOODY MARY.
ALC/VOL
(80 PROOF)

Exploration and Explotation



Exploitation use the knowledge that we currently have

Exploration try to gain new knowledge by trying new things

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-1.png"

\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-2.png"

\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-3.png"

\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-4.png"

\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-10.png"

\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Surrogate Uncertainty

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/./bin/ass/bo-aq-int-gp-15.png"

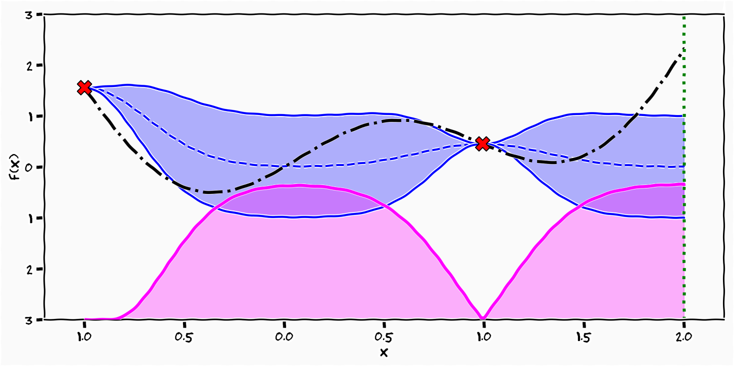
\unhbox \voidb@x \protect \penalty \@M \ /Dropbox/work/T

Acquisition Function

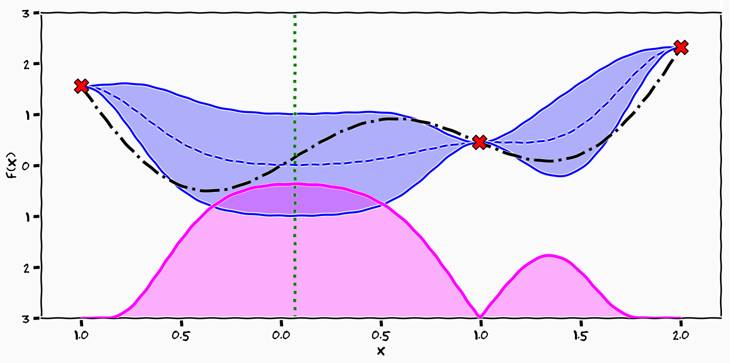
$$x_{n+1} = \underset{x \in \mathcal{X}}{\operatorname{argmax}}\, \alpha(x; \{x_i, y_i\}_{i=1}^{n}, \mathcal{M}_n)$$

- Formulate a sequential decision problem
- This will work well if $\alpha(x)$
  - is cheap to compute
  - balances *exploration* and *exploitation*

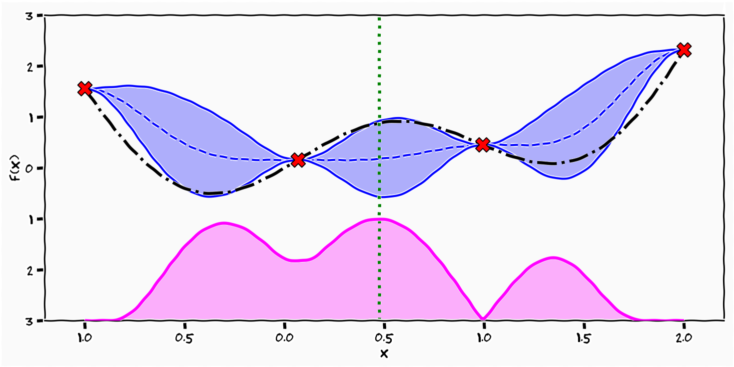
Expected Improvement



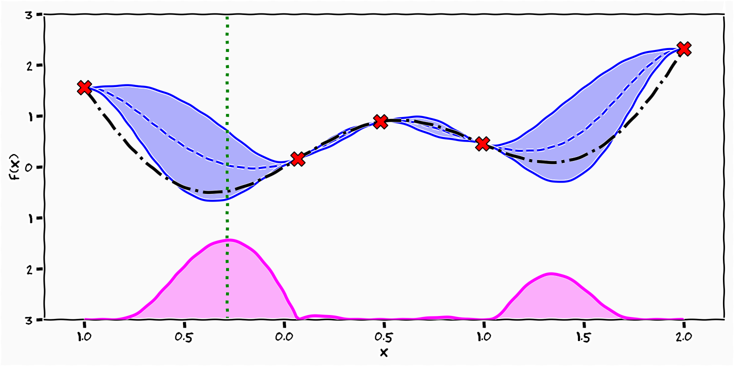
Expected Improvement



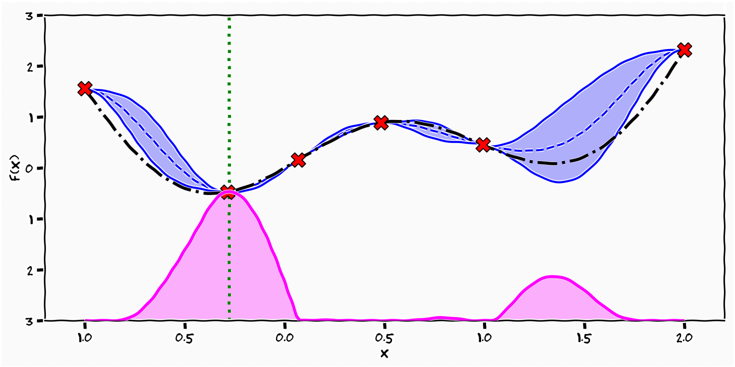
Expected Improvement



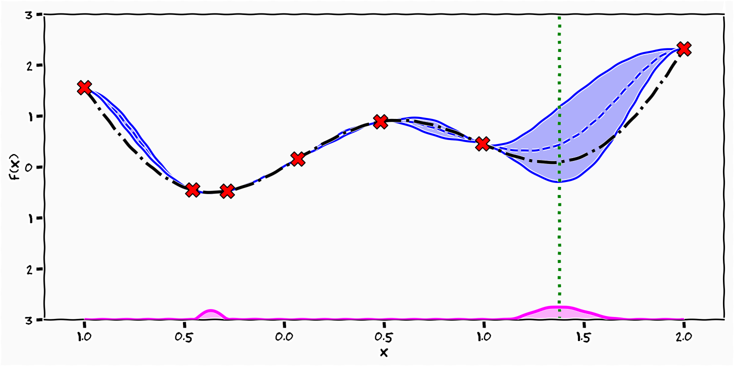
Expected Improvement



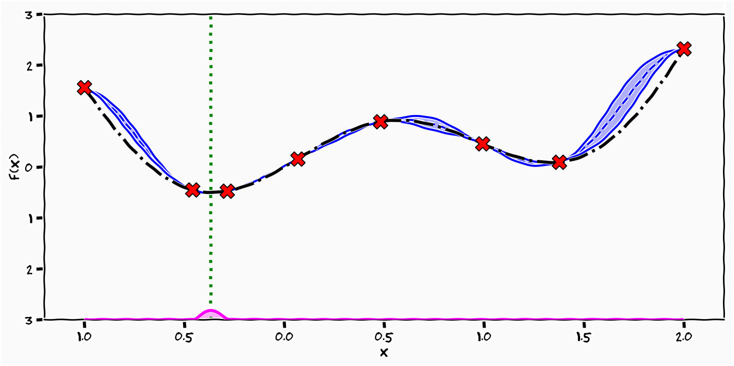
Expected Improvement

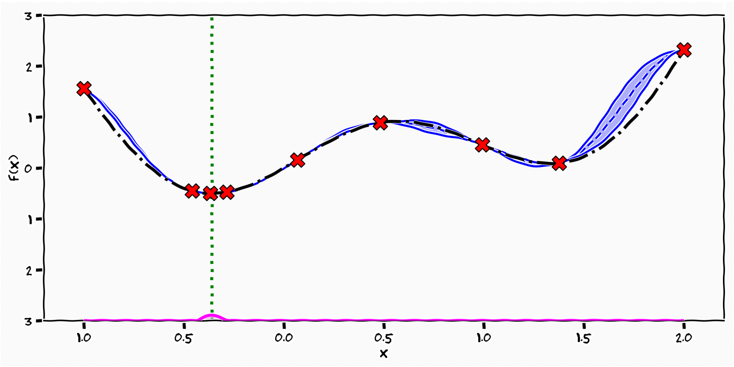


Expected Improvement



Expected Improvement




3
2
1
0
1
2
3
f(x)
1.0
0.5
0.0
0.5
1.0
1.5
2.0
x

Task 1 encode your knowledge about the function in the GP prior

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

[1]till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

Task 5 add new data to model and re-estimate hyperparameters

[1] till they open the door to the exam.

BO in Practice

Task 1 encode your knowledge about the function in the GP prior

Task 2 randomly sample some data

Task 3 specify your acquition function

Task 4 evaluate and maximise the acquisition function

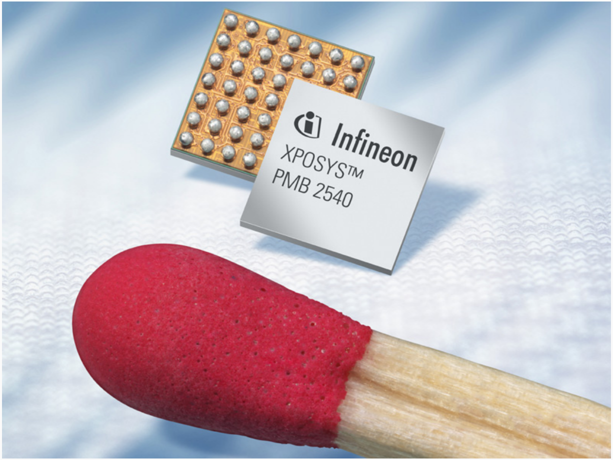
Task 5 add new data to model and re-estimate hyperparameters

Loop 4-5 till budget is gone[1]

[1] till they open the door to the exam.

How to Work on ML Problems


N90
NOKIA
Nseries
NOKIA

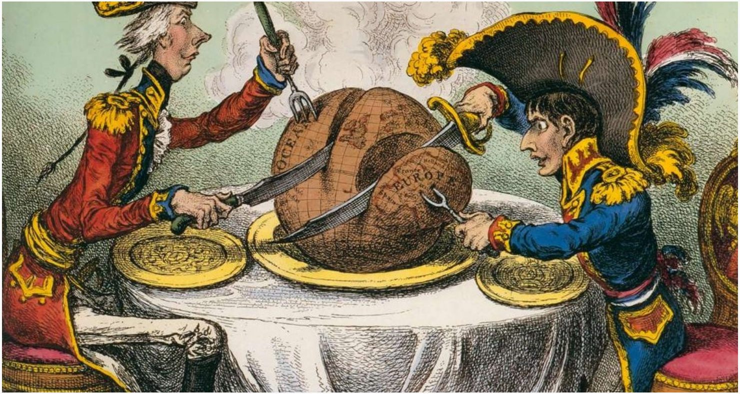

Infineon
XPOSYS™
PMB 2540

AI Replacement

/Dropbox/work/Teaching/L48-
MLPW/bin/lectures/06/bin/ass/ai-replacement.jpeg"

```
\unhbox \voidb@x \protect \penalty \@M \ /Dropb
```

ML Colonialism



The Three Asses [2]

[2] Neil Lawrence and Andrei Paleyes coined the term

Donkey's

- A*cce*ss - how to acquire data?
- *Ass*ess - what can you do with the data before you have a question?
- A*ddre*ss - what can you do when you have a question?

Infrastructure

Infrastructure

- Tensorflow
- PyTorch
- Torch
- NumpyTorch
- Edward
- Numpy
- Gluon
- Theano
- Tensorflow Probabilities
- JAX
- R
- .....

Benefits

$$p(f_*|\mathbf{x}_*, \mathbf{x}, \mathbf{y}, \boldsymbol{\theta}) = \mathcal{N}(k(\mathbf{x}_*, \mathbf{x})^{\mathrm{T}}(K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I}))^{-1}\mathbf{y},$$
$$k(\mathbf{x}_*, \mathbf{x}_*) - k(\mathbf{x}_*, \mathbf{x})^{\mathrm{T}}(K(\mathbf{x}, \mathbf{x}) + \sigma^2\mathbf{I})^{-1}K(\mathbf{x}, \mathbf{x}_*))$$

$$k(x_1, x_2) = \alpha e^{-\frac{1(x_1 - x_2)^2}{2\cdot\ell}}$$

AutoDiff Compute $\frac{\partial}{\partial x_1} p(f_*|\mathbf{x}_*, \mathbf{x}, \mathbf{y}, \boldsymbol{\theta})$

Operations $\log(e^x) = x$

Be careful!

- Compute the `log determinant` of a covariance matrix

Be careful!

- Compute the `log determinant` of a covariance matrix
- Covariance matrix $K \succeq 0$

$$K = L^{\mathrm{T}} L$$

Be careful!

- Compute the `log determinant` of a covariance matrix
- Covariance matrix $K \succeq 0$

$$K = L^{\mathrm{T}} L$$

- Simplification

$$log(|K|) = log(|L^{\mathrm{T}} L) = log(|L^{\mathrm{T}}| \cdot |L|) = 2 \cdot log(|L|)$$

Be careful!

- Compute the `log determinant` of a covariance matrix
- Covariance matrix $K \succeq 0$

$$K = L^{\mathrm{T}} L$$

- Simplification

$$log(|K|) = log(|L^{\mathrm{T}} L) = log(|L^{\mathrm{T}}| \cdot |L|) = 2 \cdot log(|L|)$$

- L is a triangular matrix $|A| = \prod \mathrm{diag}(A)$

$$log(|K|) = 2 \cdot log \left(\prod \mathrm{diag}(L) \right) = 2 \sum log \left(\mathrm{diag}(L) \right)$$

Gaussian processes

Code

```
from scipy.spatial.distance import cdist
x = np.linspace(-3,3,300).reshape(-1,1)
K = scip.exp(-cdist(X, X, 'sqeuclidean') / s**2)
y_samp = np.random.multivariate_normal(mu,K,Nsamp)
```

Summary

How to approach ML problems

- *How should I approach a machine learning problem?*

How to approach ML problems

- *How should I approach a machine learning problem?*
- Avoid doing machine learning for as long as possible

How to approach ML problems

- *How should I approach a machine learning problem?*
- Avoid doing machine learning for as long as possible
- 90% of the work in an ML project is done before you start doing ML

How to approach ML problems

- *How should I approach a machine learning problem?*
- Avoid doing machine learning for as long as possible
- 90% of the work in an ML project is done before you start doing ML
  - Focus on communication

How to approach ML problems

- *How should I approach a machine learning problem?*
- Avoid doing machine learning for as long as possible
- 90% of the work in an ML project is done before you start doing ML
  - Focus on communication
  - Clean, prepare, structure the data

How to approach ML problems

- *How should I approach a machine learning problem?*
- Avoid doing machine learning for as long as possible
- 90% of the work in an ML project is done before you start doing ML
    - Focus on communication
    - Clean, prepare, structure the data
    - Take control of the data gathering


If something's hard to do, then it's not worth doing
WWW.
INSPIRATION
POLICE.
NET

Courses

- Foundations of Data Science
- Advanced Data Science
- Probabilistic Machine Learning
- Machine Learning and the Physical World
- Deep Learning

eof