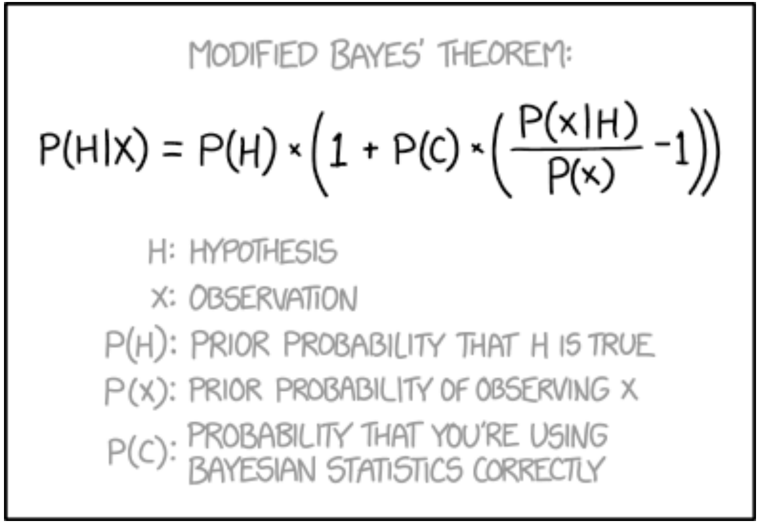
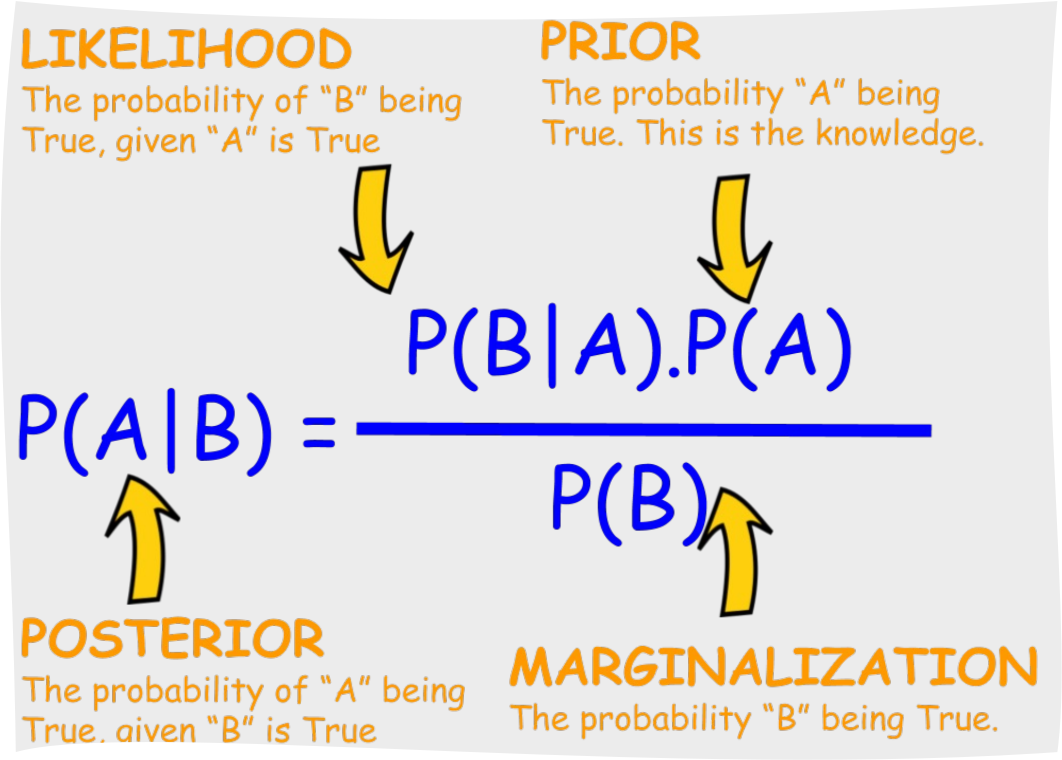
02. Introduction to Bayesian Optimization

Sep 7, 2022, MT ARD ST3 - pre-meeting Machine Learning Workshop

Chenran Xu



But Seriously... Bayes' Theorem



Rewrite with function f and observation X

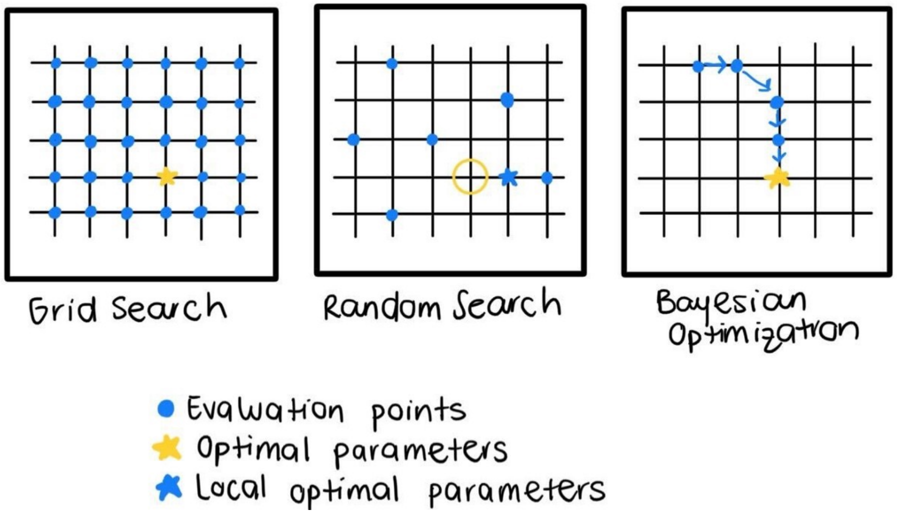
$$P(f|X) = \frac{P(X|f) * P(f)}{P(X)}$$

image: https://towardsdatascience.com/bayes-rule-with-a-simple-and-practical-example-2bce3d0f4ad0

What is Bayesian Optimization (BO)?

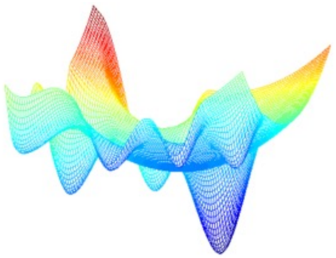
BO is a sequential algorithm for global optimization of an unknown function

It uses Bayes' theorem to update the posterior belief on the objective function and direct the optimization steps



Motivation: the Problem

Solve the global optimization problem



$$\max_{x \in A} f(x)$$

A is the optimization bounds

Unknown objective	f is not concave, no derivative information... so that gradient-based methods cannot be applied
Expensive evaluation	it takes a lot of time or has a monetary cost, number of function evaluations needs to be minimized
Continuous	the objective function can be approximated by a surrogate model
Moderate #inputs	dimension d of input x is not too high, usually $d \leq 20$

image: http://www.globaloptimization.org/

Bayesian Optimization (BO)

Pseudo Code

```
# Initialization
Build a statistical prior model (often Gaussian process, GP) on f
Observe n = n_0 initial points
# Optimization loop
while n ≤ max steps do
   Update GP posterior model using observed data
   Calculate an acquisition function α(x) based on GP model
   Choose next sample point x_n to maximize the acquisition function
   Observe y_n = f(x_n)
   n++
end while
```

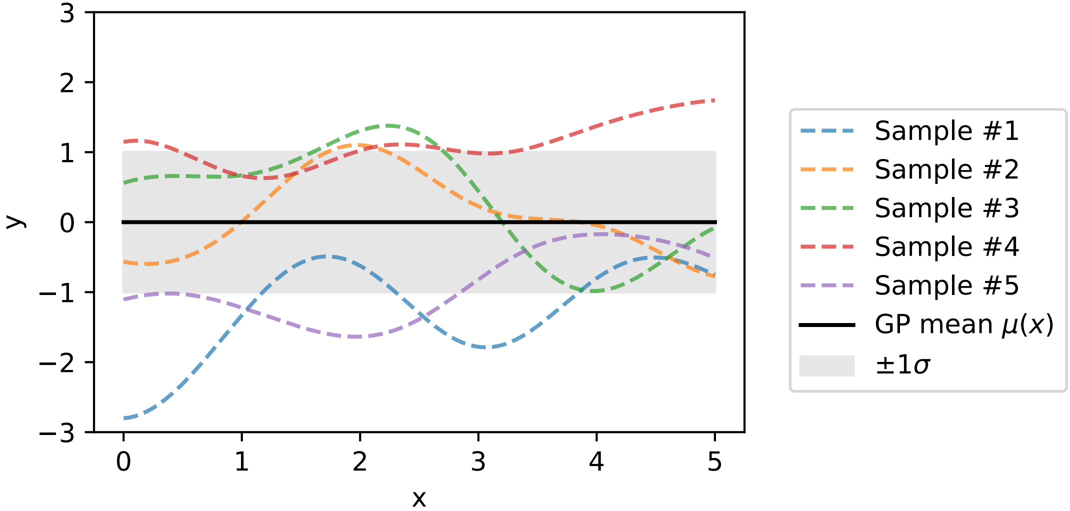
Components Explained: Gaussian Process (GP)

Often used in BO as a statistical surrogate model of the objective f

$$f \sim GP(\mu(x), k(x, x'))$$

mean

covariance (kernel)



Example of functions sampled from GP prior distribution

Note: usually the mean is set to $\mu(x) = 0$

* c.f. ***0.2.1*** *in the notebook*

GP: Covariance Function (Kernel)

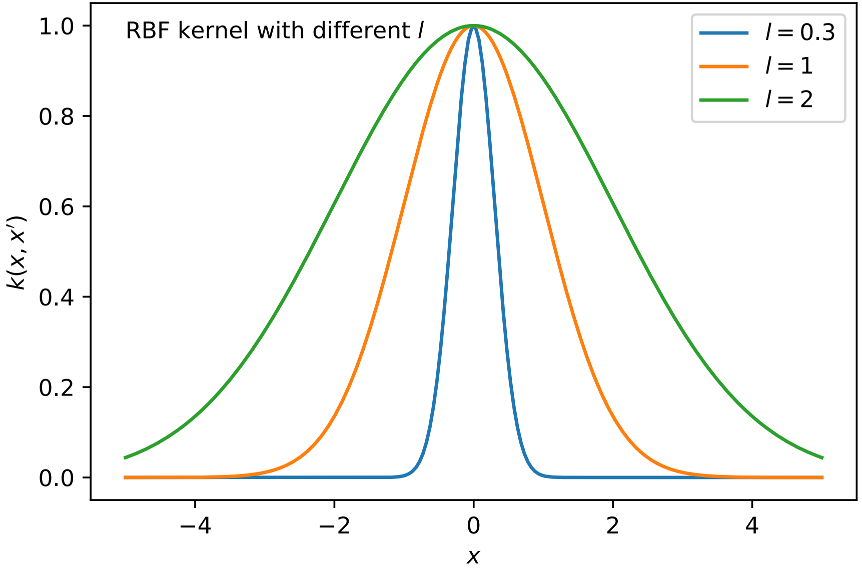
For a continuous function, nearby points x_i, x_j in the input space should have similar function values → large positive correlation.

Radial basis function

$$k_{RBF}(x, x') = \exp[-\frac{1}{2}\frac{|x - x'|^2}{l^2}]$$

$k \rightarrow 0$ for distant x, x'

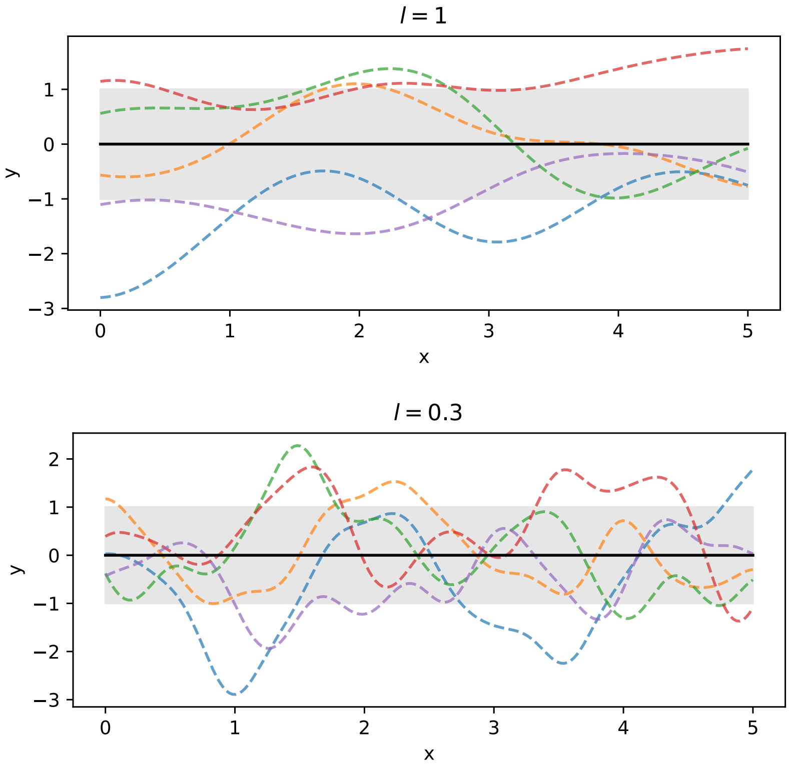
Lengthscale l controls the scaling of input
(~ how far GP can extrapolate from observed points)



GP with RBF kernel

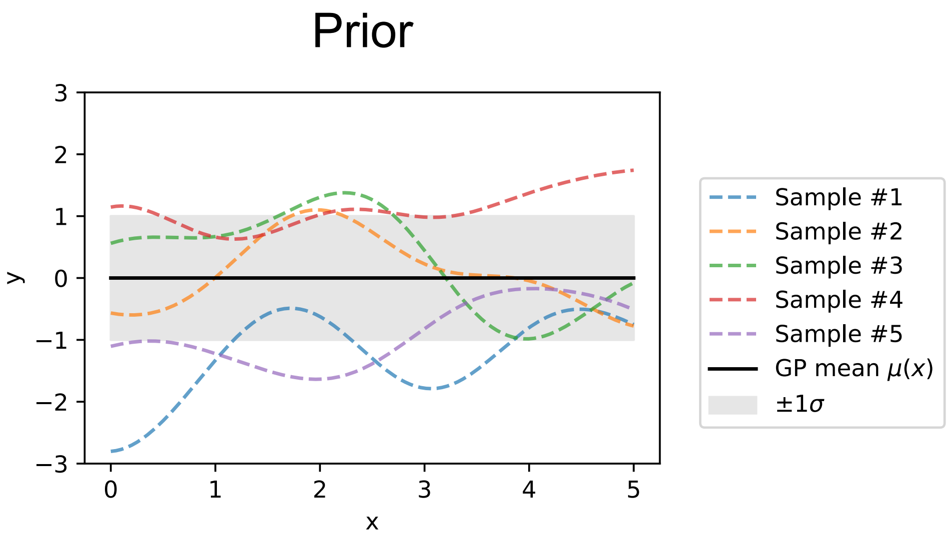
Functions sampled from GP with different lengthscales l

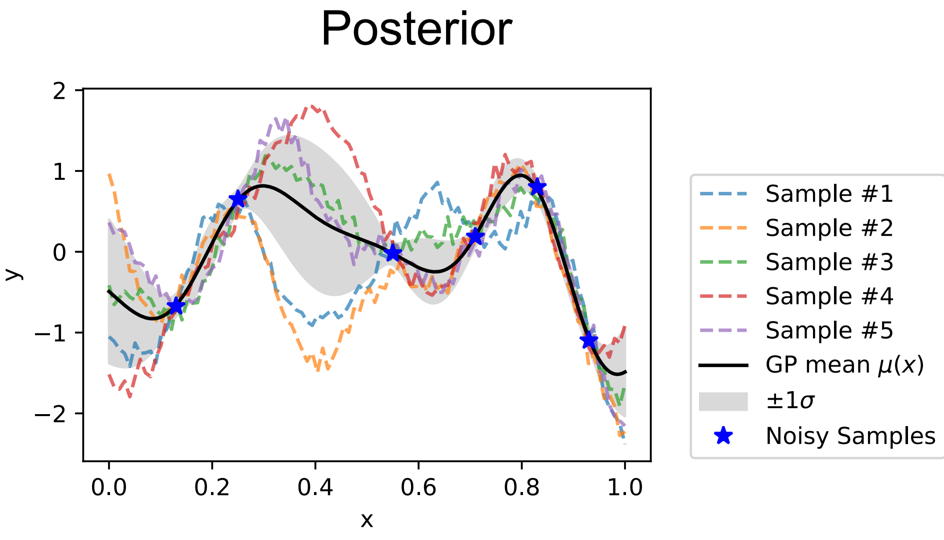
- Larger l → slow varying functions
- Small l → fast oscillating functions



GP Hyperparameters

A GP posterior distribution can be calculated after observation of the objective function





GP Hyperparameters

GP characteristics depend on the **hyperparameters** values and the **kernel choice**

parameter controlling the learning process;
often manually chosen beforehand,
or dynamically adapted during the optimization

RBF: $$k(x, x') = \sigma^2 \exp[-\frac{1}{2}\frac{|x - x'|^2}{l^2}] + \sigma_n^2$$

- Variance σ: scaling of the covariance function
- Noise σ_n : white noise of the observed signals
- Lengthscale l: scaling of input parameter space

** c.f. **0.2.2** in the notebook*

BO Components: Acquisition Function

An acquisition function α is built based on the GP posterior distribution.
The next point to be evaluated is chosen to maximise the acquisition function

→ efficient sampling of objective function

Common choice: upper confidence bound (UCB)

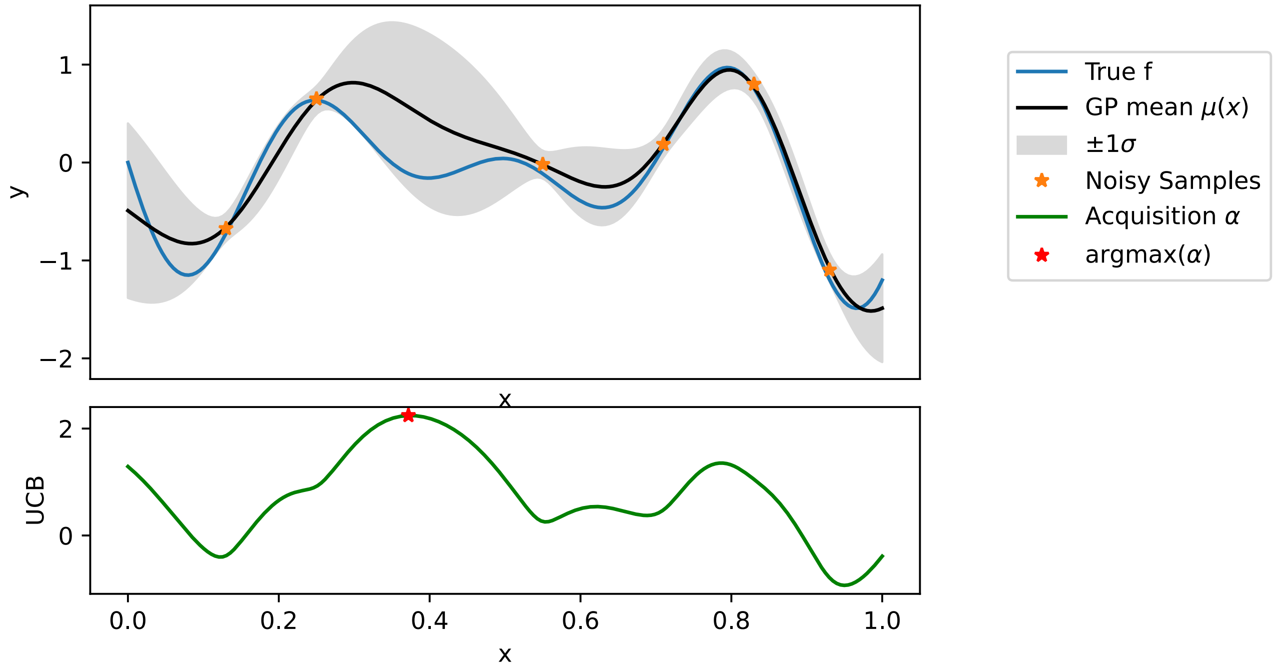
$$\alpha_{UCB}(x) = \mu(x) + \kappa\sigma(x)$$

mean and standard deviation
from GP posterior distribution

hyperparameter controlling the
exploitation-exploration trade-off

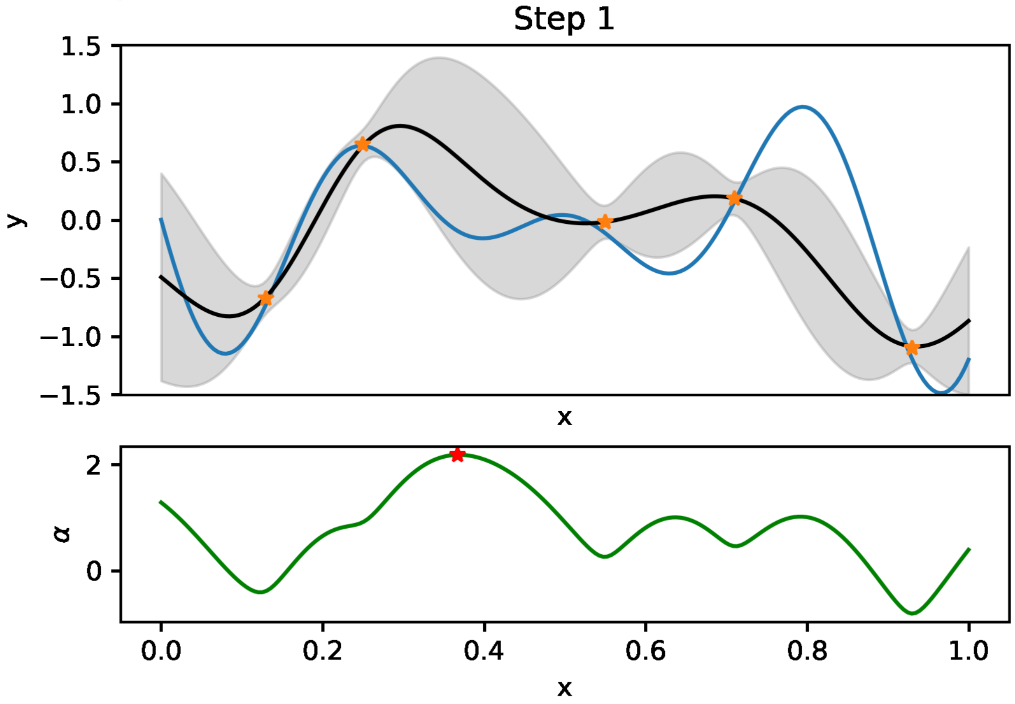
BO Components: Acquisition Function

Example: using UCB with $\kappa = 2$, corresponds to ~90% CL of normal distribution



Bayesian Optimization

Iteratively sample at $\operatorname{argmax} \alpha(x)$ and update GP model based on observations



Questions?

Let’s move on to the hands-on tutorial!

Literature & references are summarized at the end of the jupyter notebook.