

# Gaussian Processes For Global Optimization

Ludwig Winkler

Methods of Artificial Intelligence and Machine Learning  
TU Berlin

March 25, 2018

# Outline

## Gaussian Processes

- Intuition

- Observation

- Derivative Observations

## Optimization

- Optimization with Gaussian Processes

- Acquisition Function

- Example

# Problem Setting

- Optimization of a function
- Function is computationally expensive
- Trying to find optimum with few evaluations
- Do so with the help of Gaussian Processes

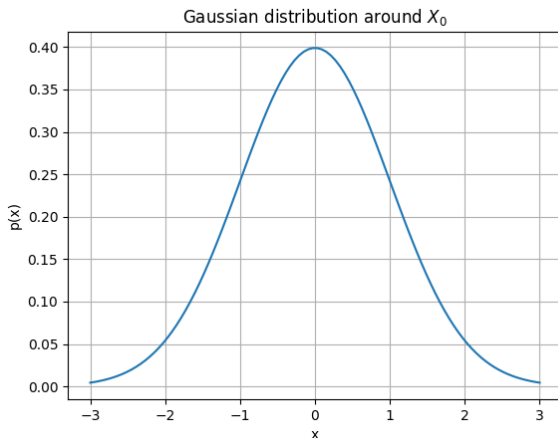
# Gaussian Processes

"A GP is defined as a **distribution over** the infinite number of **possible outputs** of our function, such that the distribution over any finite number of them is a **multivariate Gaussian**."

- Michael A. Osborne

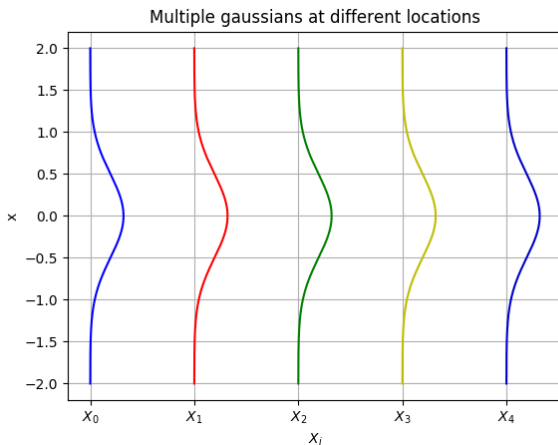
# Gaussian Processes

- Random variable with distribution we can sample from



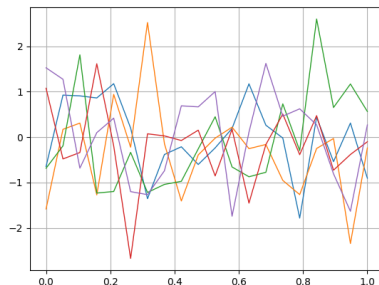
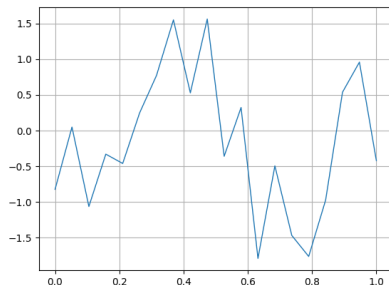
# Gaussian Processes

- Distributions at different locations  $X_i$



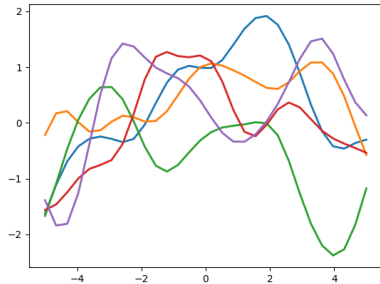
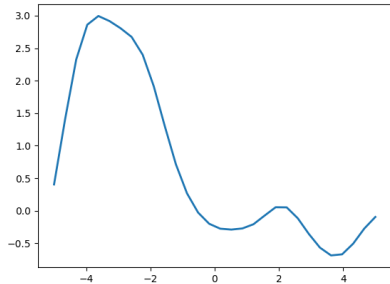
# Gaussian Processes

- Sample simultaneously from all distributions
- Each distribution is independent from all other distributions



# Gaussian Processes

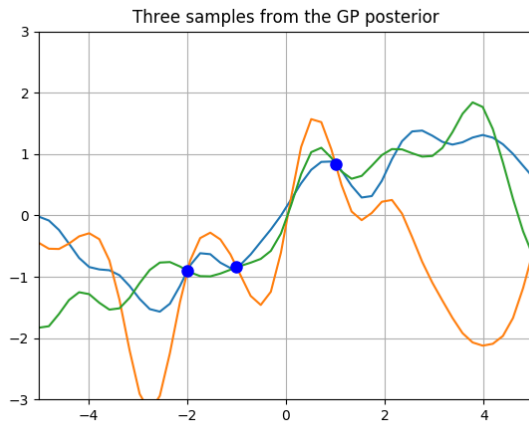
- Similar sample points should have similar values
- Introduce covariance between distributions
- Covariance determined by kernel





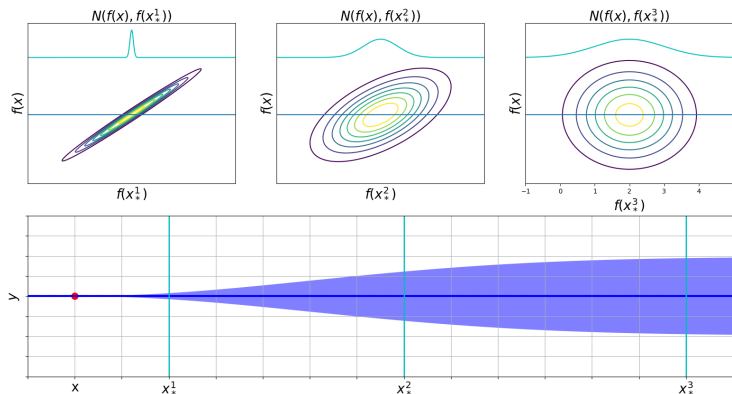
# Gaussian Processes

- Condition joint distribution over all sample points on observations



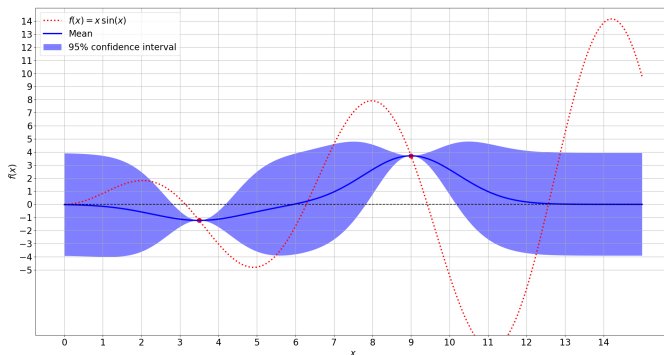
# Covariance Matrix

- Variance is reduced as observations are approached



# Gaussian Process in a Nutshell

- Observations  $\mathcal{D} = \{(x_n, y_n)_{n=0}^N\} = (\mathbf{X}, \mathbf{y})$  for regression task
- Predicted  $y_*$  should be similar to  $y_i$  if  $x_*$  is similar to  $x_i$
- Uncertainty for  $y_*$  if  $x_*$  is far away from any  $x_i$



# Gaussian Process (GP)

- Multivariate Gaussian over observations and predictions
- Kernel  $k(x_i, x_j)$  measures similarity between  $x_i$  and  $x_j$
- Kernel matrix  $K_{ij} = k(x_i, x_j)$  as covariance matrix of GP

$$\begin{bmatrix} \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}(\cdot | \mathbf{0}, \mathbf{K})$$

- Kernel matrix  $\mathbf{K}$  from observations  $\mathbf{X}$  and predictions  $\mathbf{x}_*$

$$\mathbf{K} = \begin{bmatrix} k_{\mathbf{y}, \mathbf{y}}(\mathbf{X}, \mathbf{X}) & k_{\mathbf{y}, y_*}(\mathbf{X}, x_*) \\ k_{y^*, \mathbf{y}}(x_*, \mathbf{X}) & k_{y^*, y^*}(x_*, x_*) \end{bmatrix} = \begin{bmatrix} K_{XX} & K_{Xx_*} \\ K_{x_*X} & K_{x_*x_*} \end{bmatrix}$$

# Gaussian Processes

- GP's compute predictive distributions

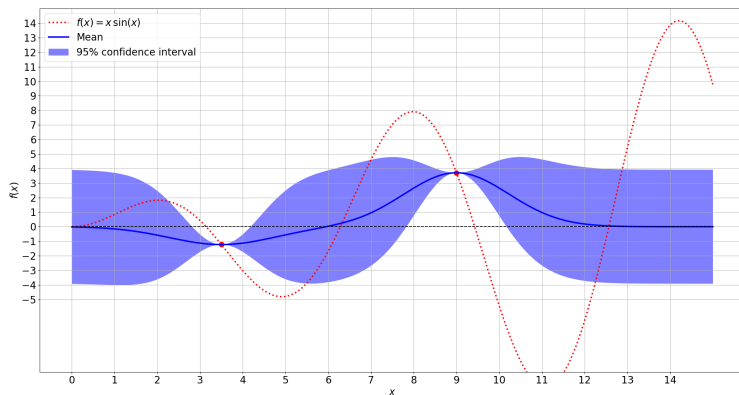
$$p(y_*|x_*, \mathbf{X}, \mathbf{y}) = \mathcal{N}\left(\underbrace{K_{x_*X}K_{XX}^{-1}\mathbf{y}}_{\mu}, \underbrace{K_{x_*x_*} - K_{x_*X}K_{XX}^{-1}K_{Xx_*}}_{\Sigma}\right)$$

- Bayesian inference for  $y_*|x_*$  based on observations  $(\mathbf{X}, \mathbf{y})$

$$\begin{aligned}\mu(x_*) &= K_{x_*X}K_{XX}^{-1}\mathbf{y} \\ \sigma^2(x_*) &= \text{diag}\left[K_{x_*x_*} - K_{x_*X}K_{XX}^{-1}K_{Xx_*}\right]\end{aligned}$$

# Observations

## GP with Observations

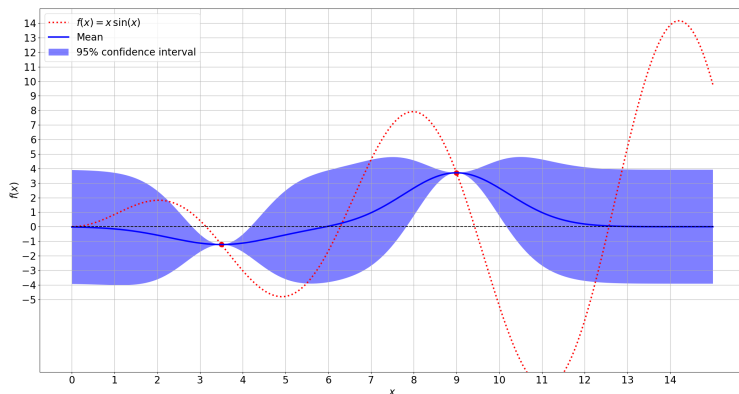


# Derivative Observations

- Points close to observations should have similar derivative
- Inclusion of derivative observations
- Better predictive distributions with lower variance

# Derivative Observations

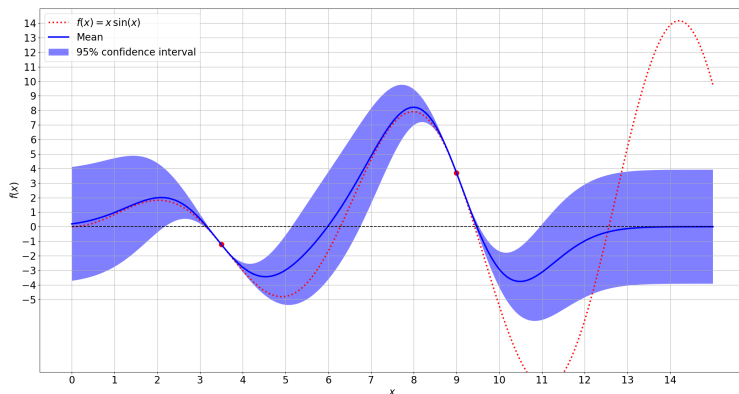
## GP with Observations





# Derivative Observations

## GP with Derivative Observations



# Derivative Observations

- GP over observations and derivative observations

$$\begin{bmatrix} \mathbf{y} \\ \nabla \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}(\cdot | \mathbf{0}, \mathbf{K}^\nabla)$$

- Expanded covariance matrix with derivative observations

$$\mathbf{K}^\nabla = \begin{bmatrix} K_{XX}^{\nabla, \nabla \nabla} & K_{Xx_*}^\nabla \\ K_{x_*X}^\nabla & K_{x_*x_*} \end{bmatrix}$$

# Derivative Observations

- Expanded kernel matrices with derivative observations

$$K_{XX}^{\nabla, \nabla \nabla} = \begin{bmatrix} k_{\mathbf{y}, \mathbf{y}}(\mathbf{X}, \mathbf{X}) & k_{\mathbf{y}, \nabla \mathbf{y}}(\mathbf{X}, \mathbf{X}) \\ k_{\nabla \mathbf{y}, \mathbf{y}}(\mathbf{X}, \mathbf{X}) & k_{\nabla \mathbf{y}, \nabla \mathbf{y}}(\mathbf{X}, \mathbf{X}) \end{bmatrix}$$

$$K_{Xx_*}^{\nabla} = \begin{bmatrix} k_{\mathbf{y}, y_*}(\mathbf{X}, x_*) \\ k_{\nabla \mathbf{y}, y_*}(\mathbf{X}, x_*) \end{bmatrix}$$

$$K_{x_*X}^{\nabla} = \begin{bmatrix} k_{y_*, \mathbf{y}}(x_*, \mathbf{X}) & k_{y_*, \nabla \mathbf{y}}(x_*, \mathbf{X}) \end{bmatrix}$$

- Modified kernels for kernel matrix

$$\text{cov}[y, \nabla_{x'} y'] = k_{y, \nabla y'}(x, x')$$

$$\text{cov}[\nabla_x y, y'] = k_{\nabla y, y'}(x, x')$$

$$\text{cov}[\nabla_x y, \nabla y'] = k_{\nabla_x y, \nabla y'}(x, x')$$

# Derivative Observations

- Modified kernels for covariance with derivative observations

$$\text{cov}[y, y'] = \frac{1}{N} \sum_{i=0}^N y_i \cdot y'_i = k(x, x')$$

$$\begin{aligned}\text{cov}[y, \nabla_{x'} y'] &= \frac{1}{N} \sum_{i=0}^N y_i \cdot \nabla_{x'} y'_i = \nabla_{x'} \frac{1}{N} \sum_{i=0}^N y_i \cdot y'_i \\ &= \nabla_{x'} \text{cov}[y, y'] = \nabla_{x'} k(x, x')\end{aligned}$$

$$\begin{aligned}\text{cov}[\nabla_x y, \nabla_{x'} y'] &= \frac{1}{N} \sum_{i=0}^N \nabla_x y_i \cdot \nabla_{x'} y'_i = \nabla_x \nabla_{x'} \frac{1}{N} \sum_{i=0}^N y_i \cdot y'_i \\ &= \nabla_x \nabla_{x'} \text{cov}[y, y'] = \nabla_x \nabla_{x'} k(x, x')\end{aligned}$$

# Derivative Observations

- Expanded kernel matrix  $\mathbf{K}_\nabla$

$$\begin{aligned}\mathbf{K}^\nabla &= \begin{bmatrix} \begin{bmatrix} k_{\mathbf{y},\mathbf{y}}(\mathbf{X}, \mathbf{X}) & k_{\mathbf{y},\nabla\mathbf{y}}(\mathbf{X}, \mathbf{X}) \\ k_{\nabla\mathbf{y},\mathbf{y}}(\mathbf{X}, \mathbf{X}) & k_{\nabla\mathbf{y},\nabla\mathbf{y}}(\mathbf{X}, \mathbf{X}) \end{bmatrix} & \begin{bmatrix} k_{\mathbf{y},y_*}(\mathbf{X}, x_*) \\ k_{\nabla\mathbf{y},y_*}(\mathbf{X}, x_*) \end{bmatrix} \\ \begin{bmatrix} k_{y_*,\mathbf{y}}(x_*, \mathbf{X}) & k_{y_*,\nabla\mathbf{y}}(x_*, \mathbf{X}) \end{bmatrix} & \begin{bmatrix} k_{y_*,y_*}(x_*, x_*) \end{bmatrix} \end{bmatrix} \\ &= \begin{bmatrix} K_{XX}^{\nabla,\nabla} & K_{Xx_*}^\nabla \\ K_{x_*X}^\nabla & K_{x_*x_*} \end{bmatrix}\end{aligned}$$

# Optimization

- Best element  $x^*$  w.r.t. to some criterion from set of elements  $\mathcal{X}$
- Minimization/maximization of objective function  $f(x)$

$$f(x^*) = \min_{x \in \mathcal{X}} f(x) = \max_{x \in \mathcal{X}} -f(x)$$

# Optimization

- Best element  $x^*$  w.r.t. to some criterion from set of elements  $\mathcal{X}$
- Minimization/maximization of objective function  $f(x)$

$$f(x^*) = \min_{x \in \mathcal{X}} f(x) = \max_{x \in \mathcal{X}} -f(x)$$

- Successive queries  $x_1, x_2, \dots \in \mathcal{X}$
- Leverage existing information to optimally select query  $x_i$
- Maximization used as exemplary optimization

# Optimization with GP

- Optimization problem as GP over set  $\mathcal{X}$
- Select next query  $x_{n+1}$  from GP with previous queries  $x_1, \dots, x_n$



# Optimization with GP

- Optimization problem as GP over set  $\mathcal{X}$
- Select next query  $x_{n+1}$  from GP with previous queries  $x_1, \dots, x_n$
- Acquisition function  $\Lambda(x \mid x_1, \dots, x_n)$  as improvement criterion

$$x_{n+1} = \operatorname{argmax}_{x \in \mathcal{X}} \Lambda(x \mid x_1, \dots, x_n)$$

# Optimization with GP

- Optimization problem as GP over set  $\mathcal{X}$
- Select next query  $x_{n+1}$  from GP with previous queries  $x_1, \dots, x_n$
- Acquisition function  $\Lambda(x \mid x_1, \dots, x_n)$  as improvement criterion

$$x_{n+1} = \operatorname{argmax}_{x \in \mathcal{X}} \Lambda(x \mid x_1, \dots, x_n)$$

- Improvement relativ to optimal solution  $x^+$  from observations  $x_{1:n}$

$$x^+ = \operatorname{argmax}_{x_i \in x_{1:n}} f(x_i)$$

# Optimization with GP

- Optimization problem as GP over set  $\mathcal{X}$
- Select next query  $x_{n+1}$  from GP with previous queries  $x_1, \dots, x_n$
- Acquisition function  $\Lambda(x \mid x_1, \dots, x_n)$  as improvement criterion

$$x_{n+1} = \operatorname{argmax}_{x \in \mathcal{X}} \Lambda(x \mid x_1, \dots, x_n)$$

- Improvement relativ to optimal solution  $x^+$  from observations  $x_{1:n}$

$$x^+ = \operatorname{argmax}_{x_i \in x_{1:n}} f(x_i)$$

- Rescaling with respect to optimal solution  $x^+$

$$z(x) = \frac{\mu(x) - f(x^+)}{\sigma(x)}$$

# Acquisition Function

- Upper confidence bound

$$\text{UCB}[x] = \mu(x) + \kappa\sigma(x) - f(x^+)$$

# Acquisition Function

- Upper confidence bound

$$\text{UCB}[x] = \mu(x) + \kappa\sigma(x) - f(x^+)$$

- Probability of improvement

$$\text{PI}[x] = P(f(x) \geq f(x^+)) = \Phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right)$$

# Acquisition Function

- Upper confidence bound

$$\text{UCB}[x] = \mu(x) + \kappa\sigma(x) - f(x^+)$$

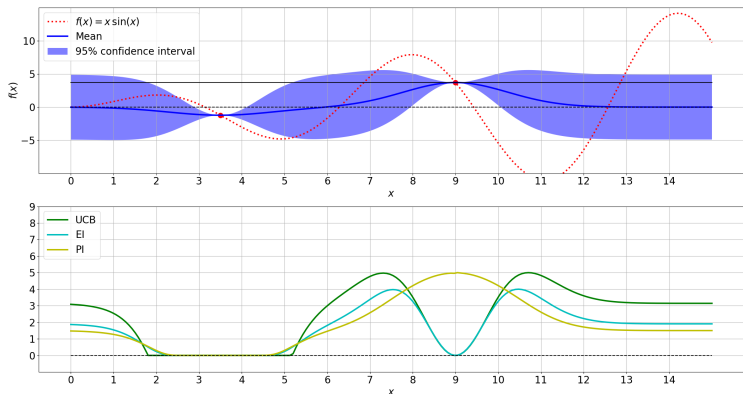
- Probability of improvement

$$\text{PI}[x] = P(f(x) \geq f(x^+)) = \Phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right)$$

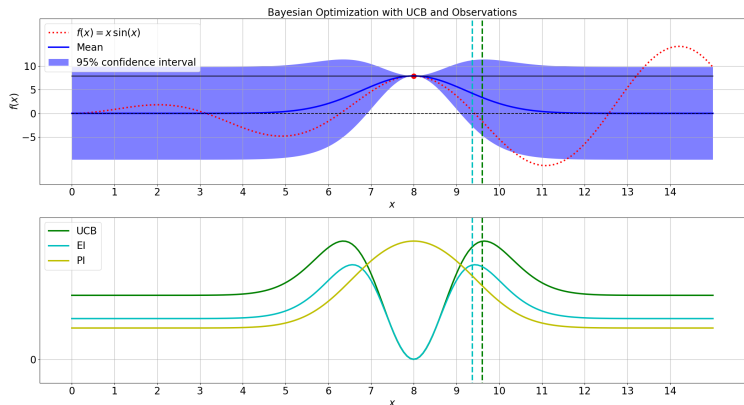
- Expected improvement

$$\text{EI}[x] = \sigma(x) \left( \frac{\mu(x) - f(x^+)}{\sigma(x)} \Phi\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right) + \mathcal{N}_{0,1}\left(\frac{\mu(x) - f(x^+)}{\sigma(x)}\right) \right)$$

# Acquisition Function

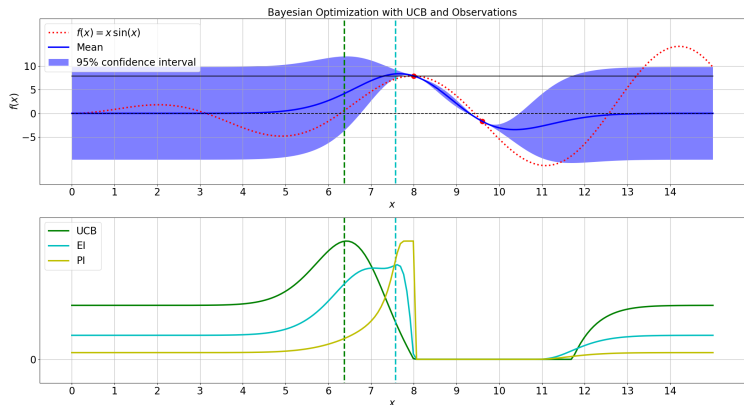


# Bayesian Optimization

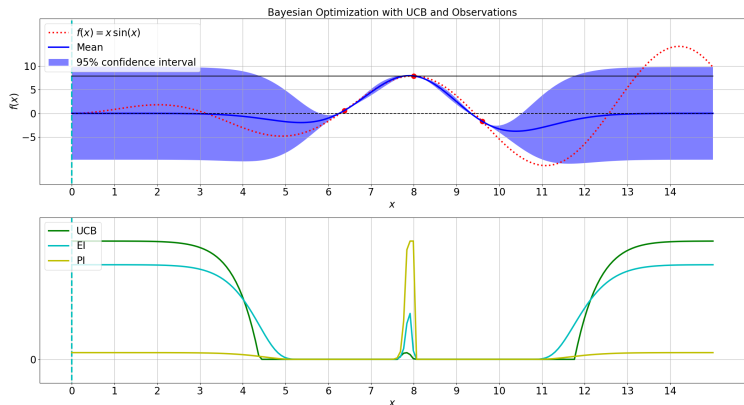




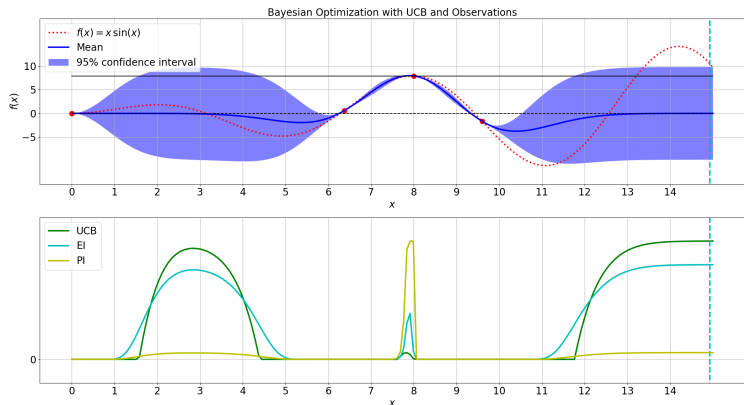
# Bayesian Optimization



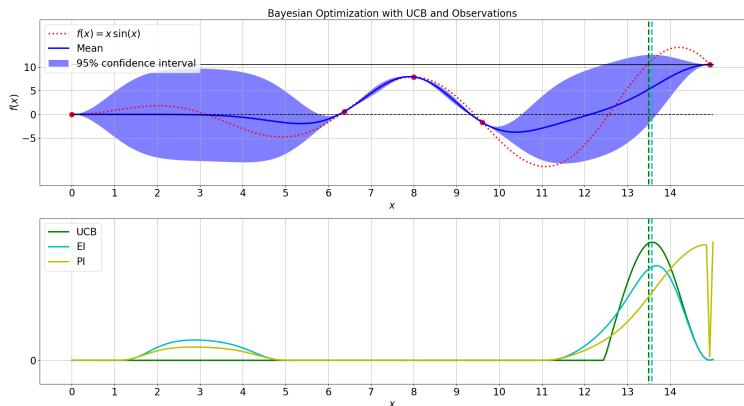
# Bayesian Optimization



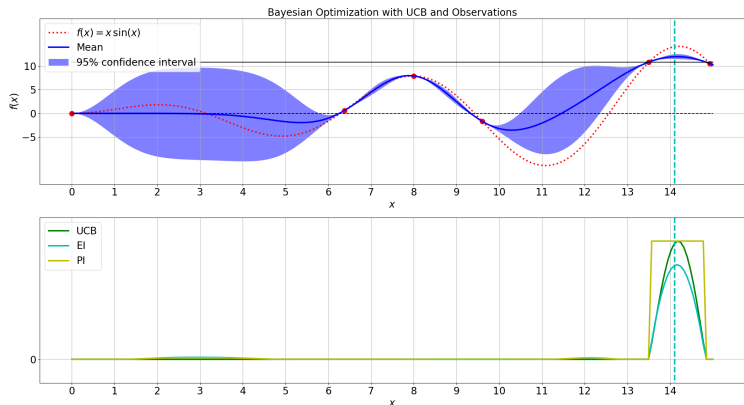
# Bayesian Optimization



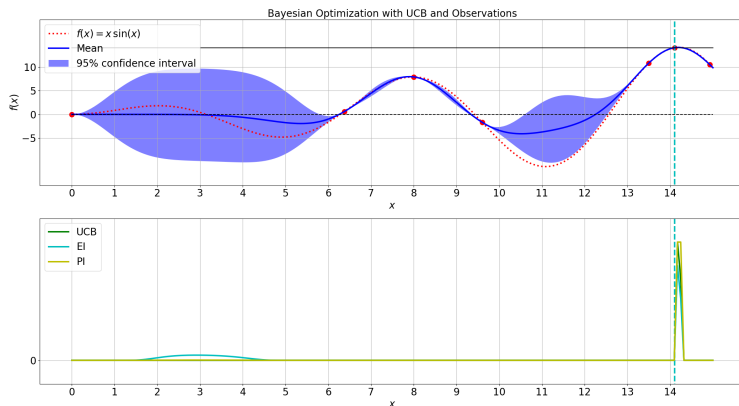
# Bayesian Optimization



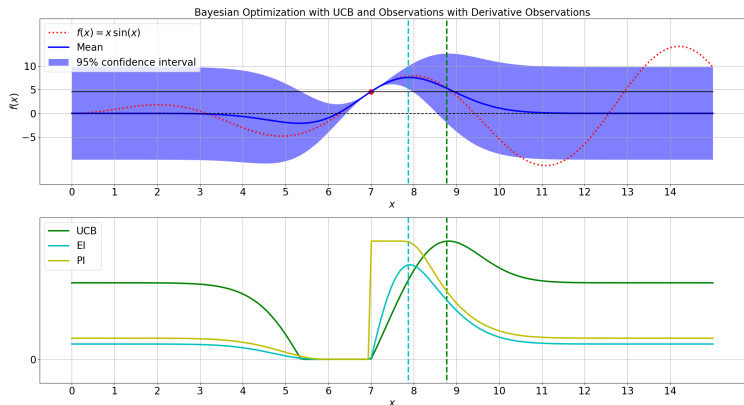
# Bayesian Optimization



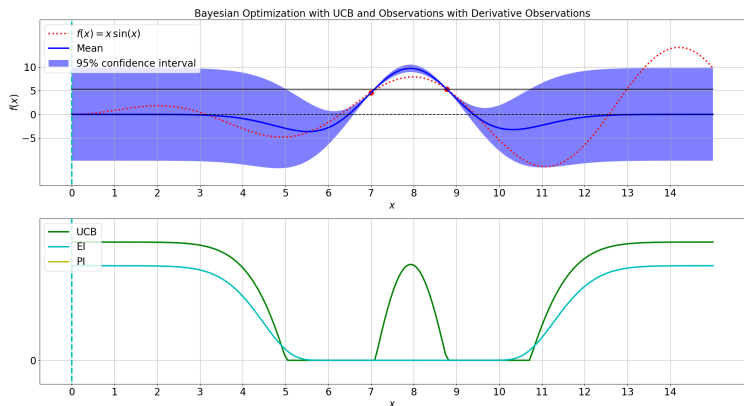
# Bayesian Optimization



# Bayesian Optimization

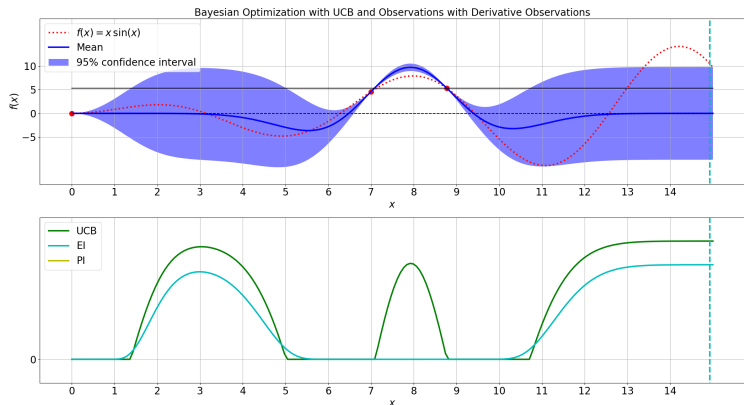


# Bayesian Optimization

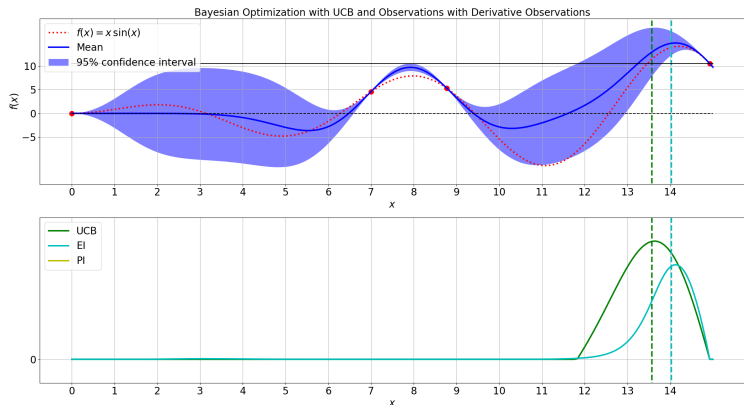




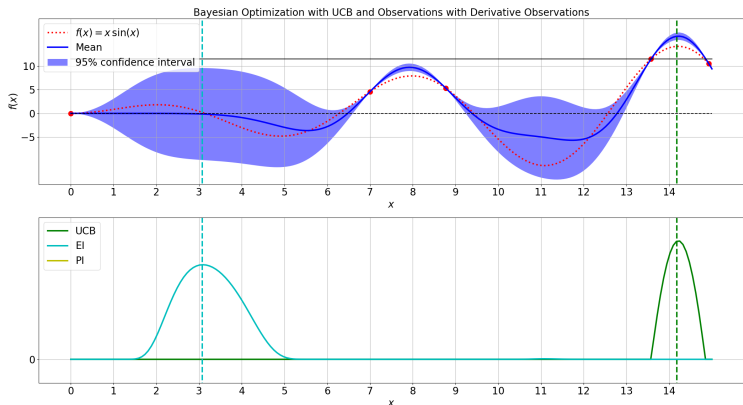
# Bayesian Optimization



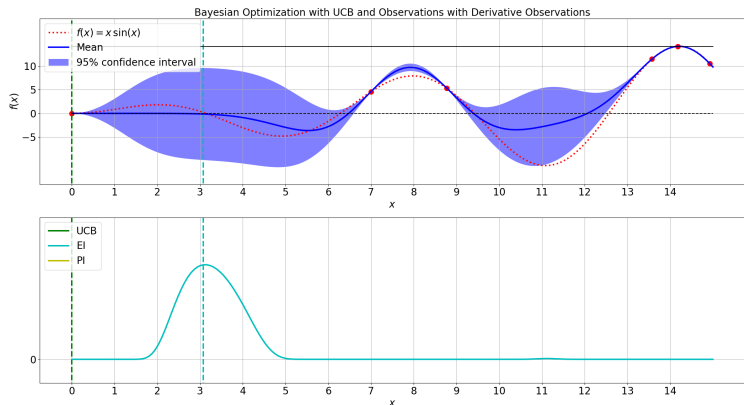
# Bayesian Optimization



# Bayesian Optimization



# Bayesian Optimization



# Outlook

- Multi-dimensional input feature space
- Noisy observations
- Multi-Step search
- Kernel hyperparameter optimization
- Bayesian Optimization for hyperparameter search

# Thank you

# Sources

- Osborne et al. - Gaussian Processes for Global optimization
- Brochu et al. - A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning
- Wu et al. - Exploiting gradients and Hessians in Bayesian optimization and Bayesian quadrature
- Solak et al. - Derivative observations in Gaussian Process Models of Dynamic Systems