Statistical Learning Group: Bayesian Optimization

Isaac J. Michaud

North Carolina State University
ijmichau@ncsu.edu

November 15, 2016

Overview

Optimization

Fundamental Problem:

For function f, find $x^{\star}$ such that $f(x^{\star}) \leq f(x)$ for all $x \in D$.

Optimization

Fundamental Problem:

For function f, find $x^\star$ such that $f(x^\star) \leq f(x)$ for all $x \in D$.

- Solve $f'(x) = 0$ and show $f''(x) > 0$

Optimization

Fundamental Problem:

For function f, find $x^\star$ such that $f(x^\star) \leq f(x)$ for all $x \in D$.

- Solve $f'(x) = 0$ and show $f''(x) > 0$
- Apply Newton's Method

$$x_{k+1} = x_k - \frac{f'(x)}{f''(x)}$$

Optimization

Fundamental Problem:

For function f, find $x^\star$ such that $f(x^\star) \leq f(x)$ for all $x \in D$.

- Solve $f'(x) = 0$ and show $f''(x) > 0$
- Apply Newton's Method

$$x_{k+1} = x_k - \frac{f'(x)}{f''(x)}$$

- Gradient Descent or Quasi-Newton (BFGS)

$$x_{k+1} = x_k - c_n f'(x), \text{ where } c_n \to 0$$

Optimization

Fundamental Problem:

For function f, find $x^\star$ such that $f(x^\star) \leq f(x)$ for all $x \in D$.

- Solve $f'(x) = 0$ and show $f''(x) > 0$
- Apply Newton's Method

$$x_{k+1} = x_k - \frac{f'(x)}{f''(x)}$$

- Gradient Descent or Quasi-Newton (BFGS)

$$x_{k+1} = x_k - c_n f'(x), \text{ where } c_n \to 0$$

- Random Search or Genetic Algorithm?

Bayesian Optimization

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

- Define a sample space of functions and build a probability model

Bayesian Optimization

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

- Define a sample space of functions and build a probability model
- Use the model to quantify uncertainty about the true value of the function

Bayesian Optimization

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

- Define a sample space of functions and build a probability model
- Use the model to quantify uncertainty about the true value of the function
- Use an **acquisition function** to decide where the minimum is most likely to be

Bayesian Optimization

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

- Define a sample space of functions and build a probability model
- Use the model to quantify uncertainty about the true value of the function
- Use an **acquisition function** to decide where the minimum is most likely to be
- Iteratively update the model by evaluating the function

Bayesian Optimization

Main Insight:

Let f be a realization of a stochastic process (even if it isn't).

- Define a sample space of functions and build a probability model
- Use the model to quantify uncertainty about the true value of the function
- Use an **acquisition function** to decide where the minimum is most likely to be
- Iteratively update the model by evaluating the function

The acquisition function drives an exploitation-exploration trade-off.

Applications in Statistics

Maximum Likelihood Estimation: For complicated models the likelihood may not have a closed form derivative and is expensive to evaluate (spatial model with $n > 10000$).

Applications in Statistics

Maximum Likelihood Estimation: For complicated models the likelihood may not have a closed form derivative and is expensive to evaluate (spatial model with $n > 10000$).

Tuning Parameter Selection/ Model Calibration: Optimize with respect to all tuning parameters simultaneously instead of individually.

Applications in Statistics

Maximum Likelihood Estimation: For complicated models the likelihood may not have a closed form derivative and is expensive to evaluate (spatial model with $n > 10000$).

Tuning Parameter Selection/ Model Calibration: Optimize with respect to all tuning parameters simultaneously instead of individually.

Optimal Bayesian Experimental Design: Let η be some design in a space of possible designs D, we can define the expected utility of η as

$$\Lambda(\eta) = \int u(\eta, \theta, y) p(y|\theta, \eta) p(\theta) dy d\theta,$$

find $\eta^\star = arg\max_{\eta \in D} \Lambda(\eta)$.

Applications in Statistics

Maximum Likelihood Estimation: For complicated models the likelihood may not have a closed form derivative and is expensive to evaluate (spatial model with $n > 10000$).

Tuning Parameter Selection/ Model Calibration: Optimize with respect to all tuning parameters simultaneously instead of individually.

Optimal Bayesian Experimental Design: Let η be some design in a space of possible designs D, we can define the expected utility of η as

$$\Lambda(\eta) = \int u(\eta, \theta, y) p(y|\theta, \eta) p(\theta) dy d\theta,$$

find $\eta^{\star} = arg\max_{\eta \in D} \Lambda(\eta)$.

Multi-armed Bandit Problems: For example A/B testing

Gaussian Processes

Definition: A random function f whose domain is $\mathbb{R}^n$ where every collection of points $\mathbf{x} = \{x_1, x_2, ..., x_k\} \subset \mathbb{R}^n$ the random vector $\{f(x_1), f(x_2), ..., f(x_k)\} \sim MVN(\mu(\mathbf{x}), \Sigma(\mathbf{x}))$.

Gaussian Processes

Definition: A random function f whose domain is $\mathbb{R}^n$ where every collection of points $\mathbf{x} = \{x_1, x_2, ..., x_k\} \subset \mathbb{R}^n$ the random vector $\{f(x_1), f(x_2), ..., f(x_k)\} \sim MVN(\mu(\mathbf{x}), \Sigma(\mathbf{x}))$.

Mean Function: Measures fixed, deterministic, trends and is often a linear combination of basis functions (think linear regression)

Gaussian Processes

Definition: A random function f whose domain is $\mathbb{R}^n$ where every collection of points $\mathbf{x} = \{x_1, x_2, ..., x_k\} \subset \mathbb{R}^n$ the random vector $\{f(x_1), f(x_2), ..., f(x_k)\} \sim MVN(\mu(\mathbf{x}), \Sigma(\mathbf{x}))$.

Mean Function: Measures fixed, deterministic, trends and is often a linear combination of basis functions (think linear regression)

Covariance Function: Measures the covariance between pairs of domain locations. Usually this is assumed to have the following properties:

- Stationary (same everywhere)
- Isotropic (same in all directions)
- Decays with distance

Gaussian Processes

Definition: A random function f whose domain is $\mathbb{R}^n$ where every collection of points $\mathbf{x} = \{x_1, x_2, ..., x_k\} \subset \mathbb{R}^n$ the random vector $\{f(x_1), f(x_2), ..., f(x_k)\} \sim MVN(\mu(\mathbf{x}), \Sigma(\mathbf{x}))$.

Mean Function: Measures fixed, deterministic, trends and is often a linear combination of basis functions (think linear regression)

Covariance Function: Measures the covariance between pairs of domain locations. Usually this is assumed to have the following properties:

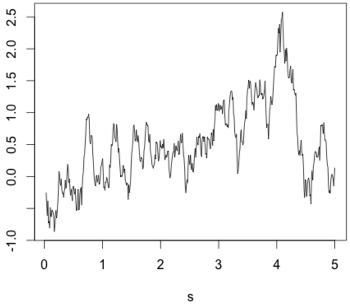
- Stationary (same everywhere)
- Isotropic (same in all directions)
- Decays with distance

Examples:

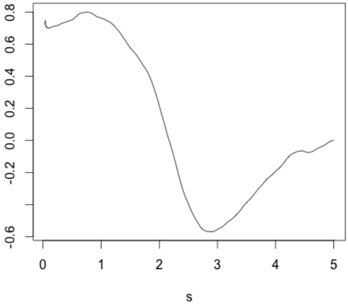
- Exponential - $C(x, y) = e^{\frac{1}{\rho}\|x-y\|}$
- Gaussian - $C(x, y) = e^{\frac{1}{2\rho^2}\|x-y\|^2}$

Differences in Covariance Functions

Here are random draws from mean zero Gaussian processes with different covariance functions:



Exponential Covariance



Gaussian Covariance

Kriging (Gaussian Process Regression)

Gaussian processes allow us to make predictions at unobserved locations. Let X_1 be the observed locations and X_2 be the unobserved locations. Before data is collected we have:

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

Kriging (Gaussian Process Regression)

Gaussian processes allow us to make predictions at unobserved locations. Let X_1 be the observed locations and X_2 be the unobserved locations. Before data is collected we have:

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

and using conditional expectations:

$$X_2|X_1 \sim MVN\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(X_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right).$$

Kriging (Gaussian Process Regression)

Gaussian processes allow us to make predictions at unobserved locations. Let X_1 be the observed locations and X_2 be the unobserved locations. Before data is collected we have:

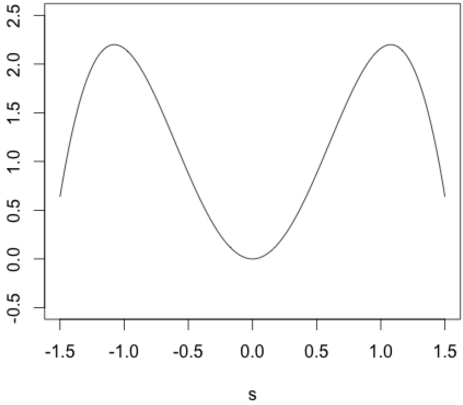
$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim MVN\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$

and using conditional expectations:

$$X_2|X_1 \sim MVN\left(\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(X_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right).$$

We need some initial data to fit a Gaussian process, usually collected by taking a Latin Hypercube Sample (space-filling design).

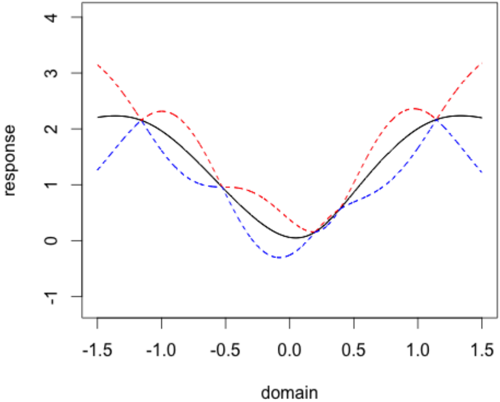
Example: Fit a GP to the function $f(x) = 4x^2 cos(x)$ on $[-1.5, 1.5]$



Kriging cont.

Example: Fit a GP to the function $f(x) = 4x^2 cos(x)$ on $[-1.5, 1.5]$

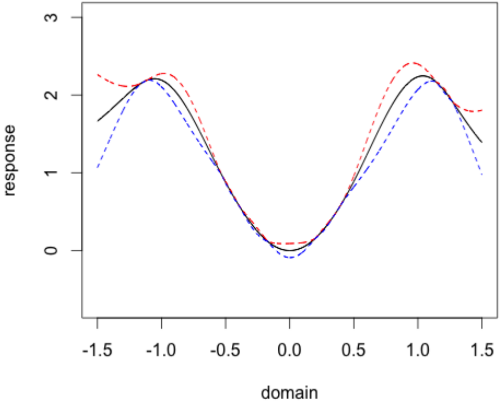
- Take 5 point Latin Hypercube Sample and fit GP model to data
- Red and Blue lines represent the 5^{th} and 95^{th} quantiles



Kriging cont.

Example: Fit a GP to the function $f(x) = 4x^2 cos(x)$ on $[-1.5, 1.5]$

- Augment with 5 more function evaluations and refit GP model to data
- Red and Blue lines represent the 5^{th} and 95^{th} quantiles



Where to evaluate next?

Goal: Find the minimum of f

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Definition: An *acquisition function* takes a model and tells us where the most promising locations are

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Definition: An *acquisition function* takes a model and tells us where the most promising locations are

Expected Improvement (EI): Let $a_n = min\{f(x_1), f(x_2), ..., f(x_n)\}$ be the smallest observed function value at stage n and Y_n be the Gaussian process fit to the observed data, then

$$EI(x) = E[max\{0, a_n - Y_n(x)\}].$$

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Definition: An *acquisition function* takes a model and tells us where the most promising locations are

Expected Improvement (EI): Let $a_n = min\{f(x_1), f(x_2), ..., f(x_n)\}$ be the smallest observed function value at stage n and Y_n be the Gaussian process fit to the observed data, then

$$EI(x) = E[max\{0, a_n - Y_n(x)\}].$$

- EI will be large where the function is known to be minimized or there is uncertainty about its value

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Definition: An *acquisition function* takes a model and tells us where the most promising locations are

Expected Improvement (EI): Let $a_n = min\{f(x_1), f(x_2), ..., f(x_n)\}$ be the smallest observed function value at stage n and Y_n be the Gaussian process fit to the observed data, then

$$EI(x) = E[max\{0, a_n - Y_n(x)\}].$$

- EI will be large where the function is known to be minimized or there is uncertainty about its value
- Maximize EI and sequentially update the GP

Where to evaluate next?

Goal: Find the minimum of f

Observation: It's unnecessary to evaluate where the minimum is unlikely

Definition: An *acquisition function* takes a model and tells us where the most promising locations are

Expected Improvement (EI): Let $a_n = min\{f(x_1), f(x_2), ..., f(x_n)\}$ be the smallest observed function value at stage n and Y_n be the Gaussian process fit to the observed data, then

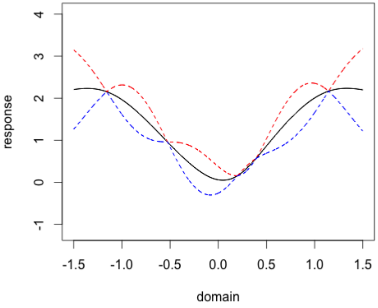
$$EI(x) = E[max\{0, a_n - Y_n(x)\}].$$

- EI will be large where the function is known to be minimized or there is uncertainty about its value
- Maximize EI and sequentially update the GP
- Converges to the minimum under regularity conditions

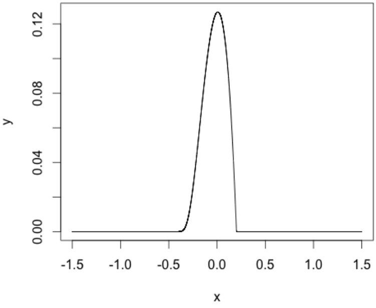
Expected Improvement

Example: Minimize $f(x) = 4x^2 cos(x)$ on $[-1.5, 1.5]$

- Take 5 point Latin Hypercube Sample and fit GP model to data
- Maximize EI (at $x = 0.008$)



Gaussian Process

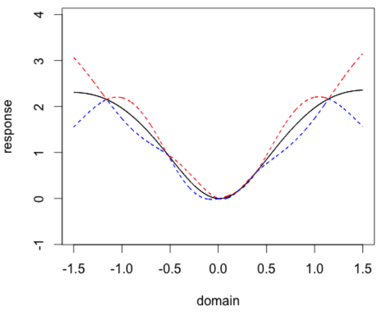


EI

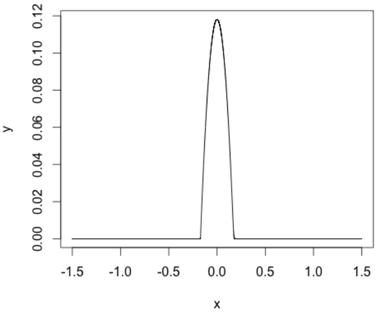
Expected Improvement

Example: Minimize $f(x) = 4x^2 cos(x)$ on $[-1.5, 1.5]$

- Augment data with $(0.008, f(0.008)$ and refit GP model
- Maximize EI (at $x = 0.0000765$)



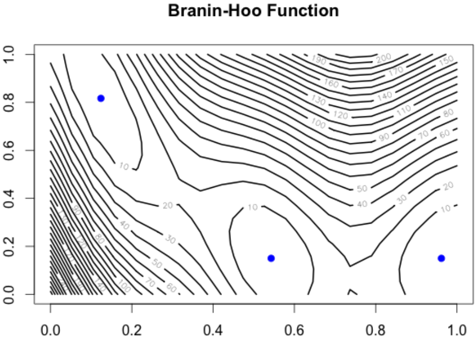
Gaussian Process



EI

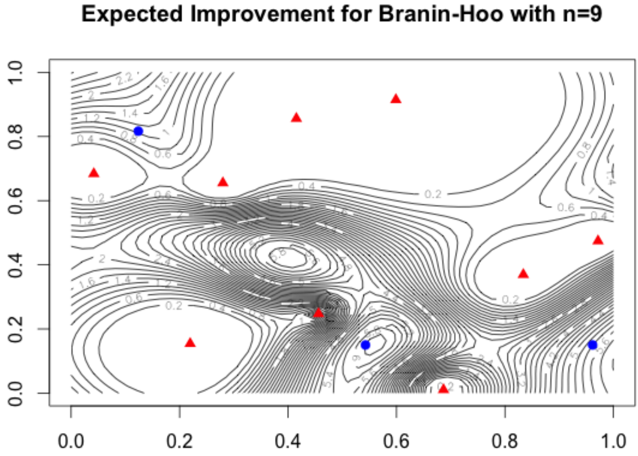
Branin-Hoo Function

$$f(x,y) = \left(-\frac{1.275x^2}{\pi} + \frac{5x}{\pi} + y - 6\right)^2 + \left(10 - \frac{5}{4\pi}\right) cos(x) + 10$$

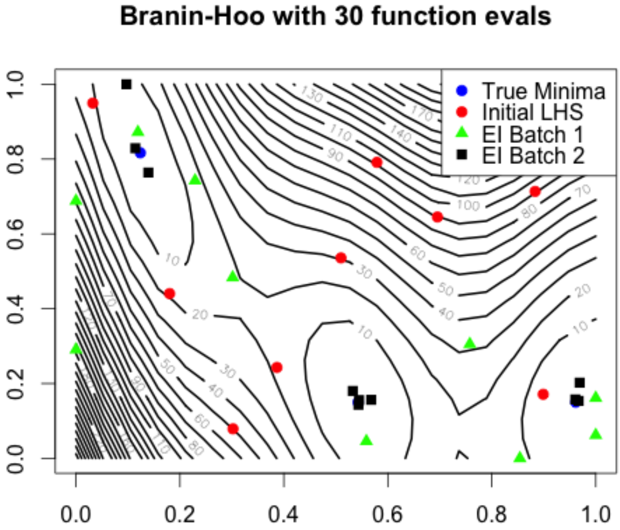


Branin-Hoo Function Expected Improvement

- Maximum at $x = 0.5610115$ and $y = 0.1523989$



Branin-Hoo Function Expected Improvement



Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes?

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes? Yes we can! Just add a nugget and....

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes? Yes we can! Just add a nugget and....

Expected Quantile Improvement:

$$EQI(x, \tau^2) = E[max(0, Q_{\beta,min} - Q_\beta(x)]$$

- Looks at the improvement of the β quantile

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes? Yes we can! Just add a nugget and....

Expected Quantile Improvement:

$$EQI(x, \tau^2) = E[max(0, Q_{\beta,min} - Q_{\beta}(x)]$$

- Looks at the improvement of the β quantile
- EI is a special case when $\beta = 0.5$

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes? Yes we can! Just add a nugget and....

Expected Quantile Improvement:

$$EQI(x, \tau^2) = E[max(0, Q_{\beta,min} - Q_\beta(x)]$$

- Looks at the improvement of the β quantile
- EI is a special case when $\beta = 0.5$
- τ^2 is a tuning parameter (future expected variance)

Noisy Optimization

What happens if f can only be simulated using Monte Carlo?

- Stochastic Gradient Decent (f' can be simulated)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Random Search

Can we use Gaussian Processes? Yes we can! Just add a nugget and....

Expected Quantile Improvement:

$$EQI(x, \tau^2) = E[max(0, Q_{\beta,min} - Q_\beta(x)]$$

- Looks at the improvement of the β quantile
- EI is a special case when $\beta = 0.5$
- τ^2 is a tuning parameter (future expected variance)

Optimal Bayesian Experimental Design:

$$\underset{\eta}{\operatorname{argmax}} \int u(\eta, \theta, y) p(y|\theta, \eta) p(\theta) dy d\theta$$

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients
- Parsimonious in the number of function evaluations

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients
- Parsimonious in the number of function evaluations

Bayesian optimization is not, as Steven Boyd would say, a mature technology. It needs a lot of work:

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients
- Parsimonious in the number of function evaluations

Bayesian optimization is not, as Steven Boyd would say, a mature technology. It needs a lot of work:

- Clustering and multi-extrema identification

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients
- Parsimonious in the number of function evaluations

Bayesian optimization is not, as Steven Boyd would say, a mature technology. It needs a lot of work:

- Clustering and multi-extrema identification
- Gaussian process validation (i.e. does the surrogate fit the function)

Conclusions

Bayesian optimization provides the following benefits:

- A probabilistic approach to optimization
- Good convergence without gradients
- Parsimonious in the number of function evaluations

Bayesian optimization is not, as Steven Boyd would say, a mature technology. It needs a lot of work:

- Clustering and multi-extrema identification
- Gaussian process validation (i.e. does the surrogate fit the function)
- Acquisition function optimization

References

Jones, Donald R., Matthias Schonlau, and William J. Welch. "Efficient global optimization of expensive black-box functions." Journal of Global optimization 13.4 (1998): 455-492.

Picheny, Victor, et al. "Quantile-based optimization of noisy computer experiments with tunable precision." Technometrics 55.1 (2013): 2-13.

Shahriari, Bobak, et al. "Taking the human out of the loop: A review of bayesian optimization." Proceedings of the IEEE 104.1 (2016): 148-175.