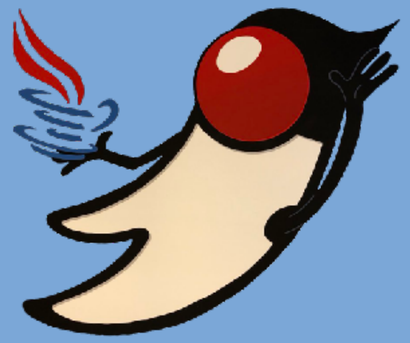
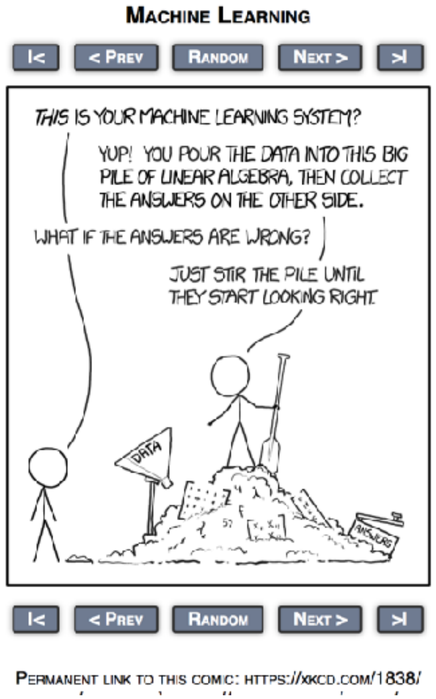
Bayesian Optimization of Gaussian Processes applied to Performance Tuning

Ramki Ramakrishna
@ysr1729
#TwitterVMTeam

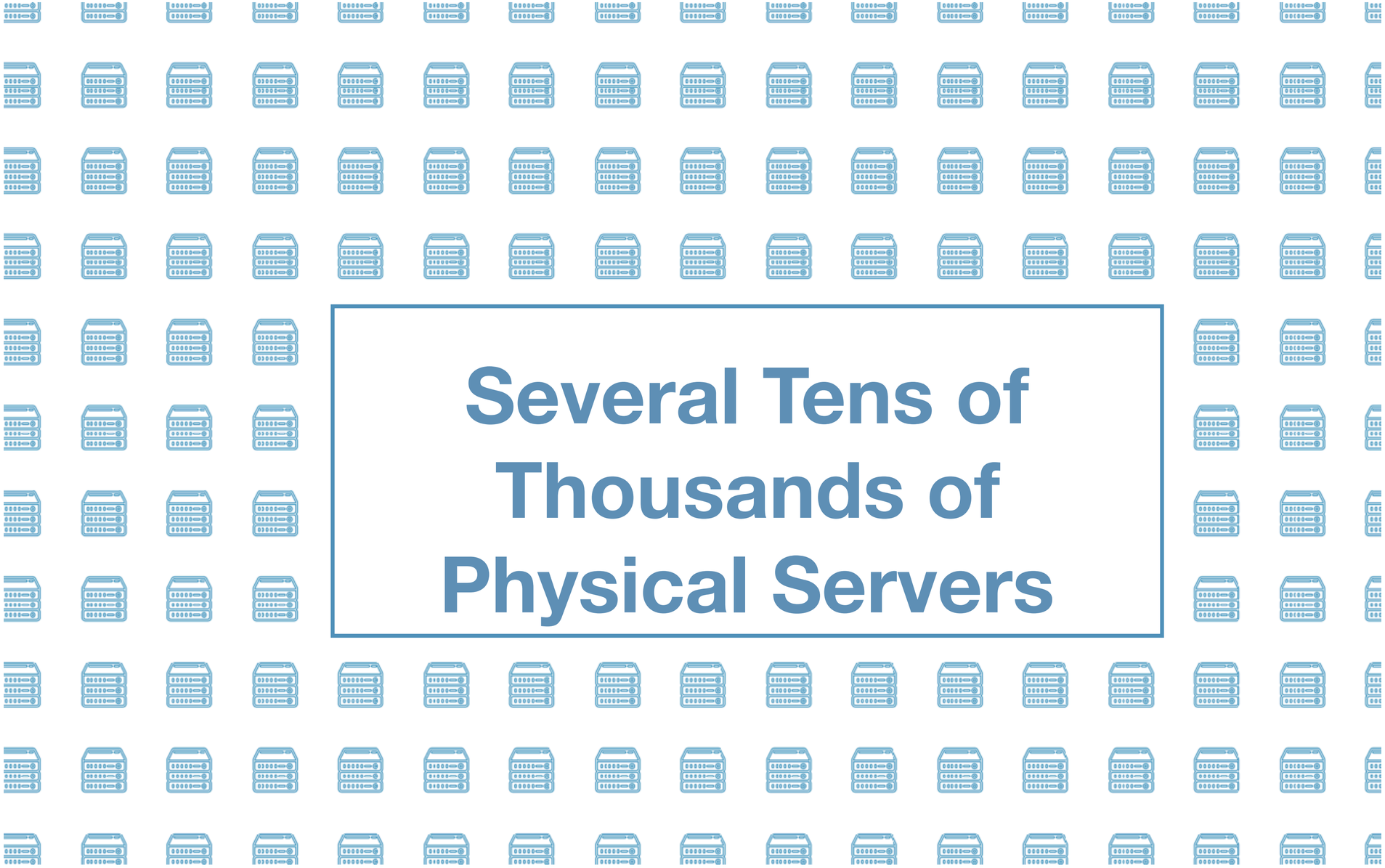
QCon Sao Paulo, 2019

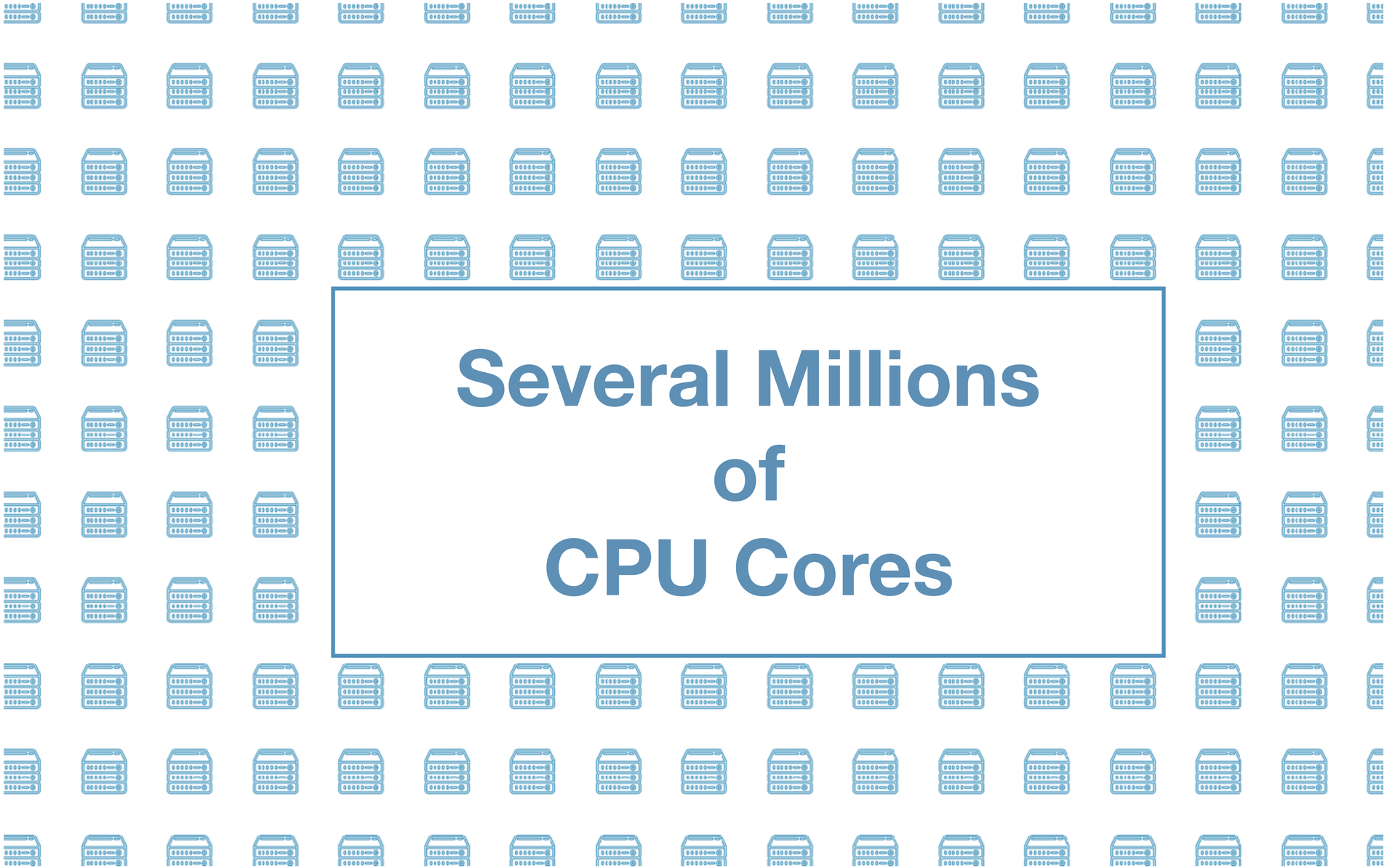


A JVM Engineer talks to a Data Scientist

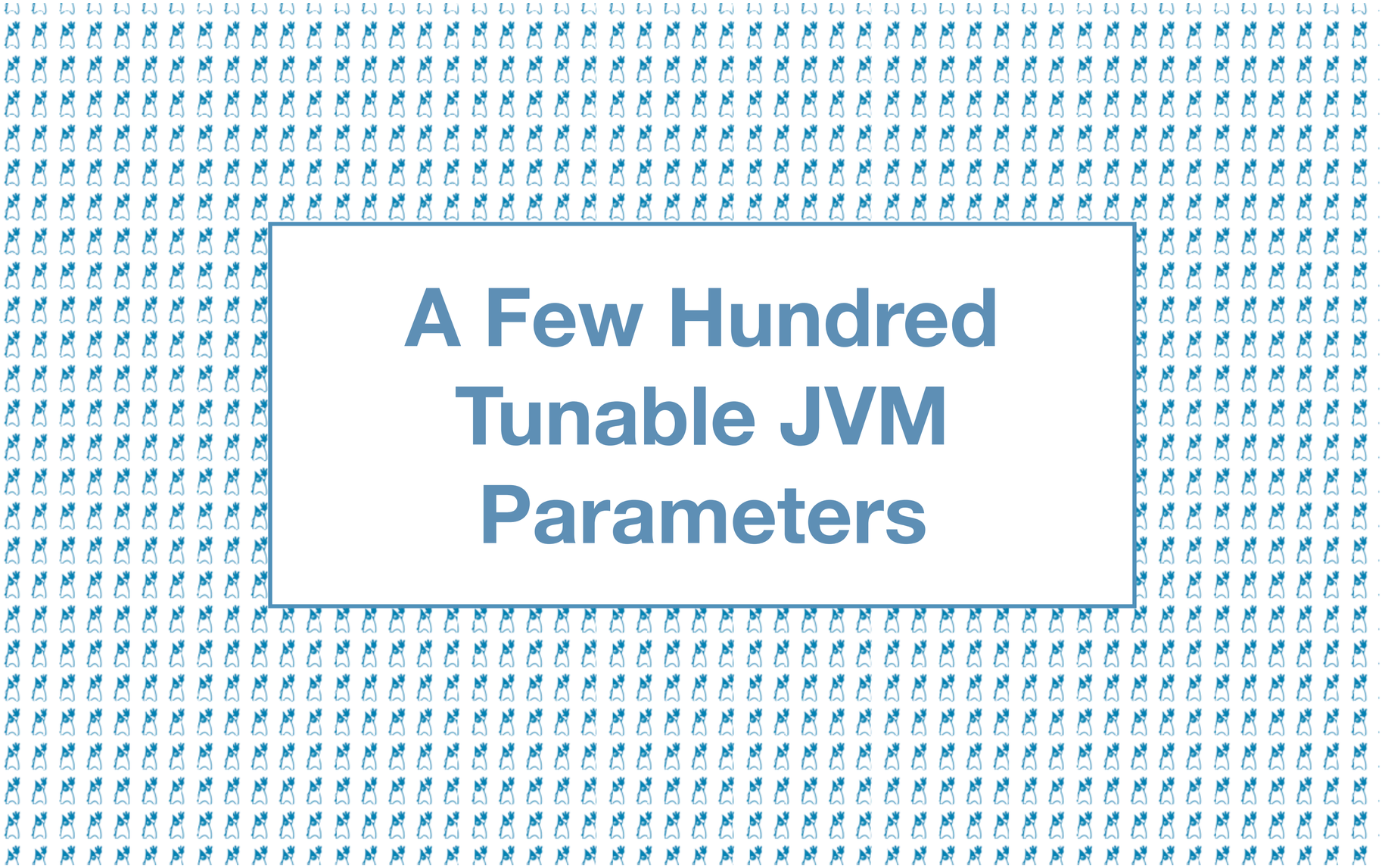


Many Hundreds of Services


Several Tens of
Thousands of
Physical Servers


Several Millions
of
CPU Cores


Several Hundreds of Thousands of Twitter JVMs


A Few Hundred
Tunable JVM
Parameters


Service
Finagle
Twitter JVM
Mesos Container
Kernel & OS
Hardware


f(
Service
Finagle
Twitter JVM
Mesos Container
Kernel & OS
Hardware
)
x 10
x 10
x 100
x 10
x 10
x 10

Andrew Ng

@AndrewYNg

If you're trying to understand AI's near-term impact, don't think "sentience." Instead think "automation on steroids."

11:04 AM - 1 May 2017 from Stanford, CA

1,397 Retweets **2,085** Likes

38 1.4K 2.1K

Mining for Gold

- 1930’s South Africa
- Prospecting for gold and other minerals
- Daniel Krige, 1951: “Kriging” in geostatistics
- Jonas Mockus 70’s
- Jones et al. 80’s
- Rasmussen & Williams 90’s : Gaussian Processes

Applications

- design of expensive experiments
- optimal designs
- optimization of engineered materials
- hyperparameter tuning (architectural parameters) of neural networks

Engineering as Optimization

- linear or non-linear objective function
- finite convex or non-convex space, rectangular, linear (affine) or non-linear constraints
- black box objective function
- black box constraints
- noisy objective function
- noisy constraints

Black Box Modeling

- Model the unknown objective function
- Model the unknown constraints
- Model is a “surrogate”
- Evaluations are expensive

Models and Model Parameters

- Parametric models
- Non-parametric models

Probabilistic Models

- A measure of our uncertainty
- A measure of measurement/observation noise

Gaussian Process

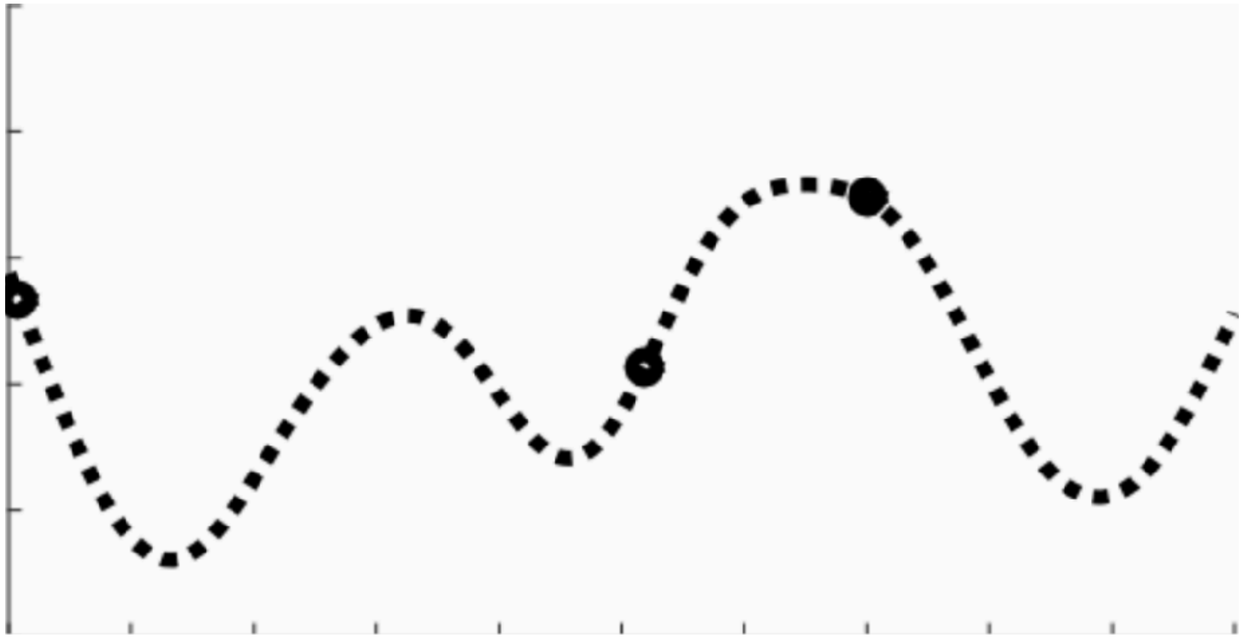
$$GP(\mu, \kappa)$$

- $\mu(x)$: mean function
- $\kappa(x, x')$: covariance function

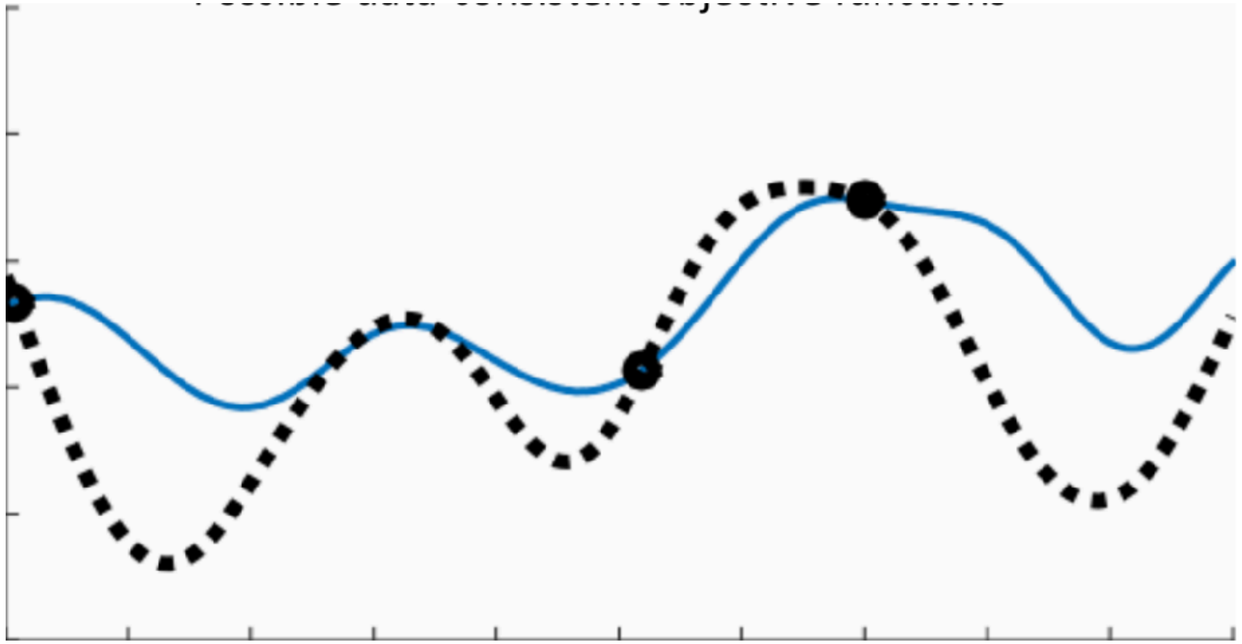
Gaussian Process

- Two different views
  - a vector of possibly uncountably many Gaussian variables with given mean and a joint covariate distribution
  - a Gaussian distribution over functions

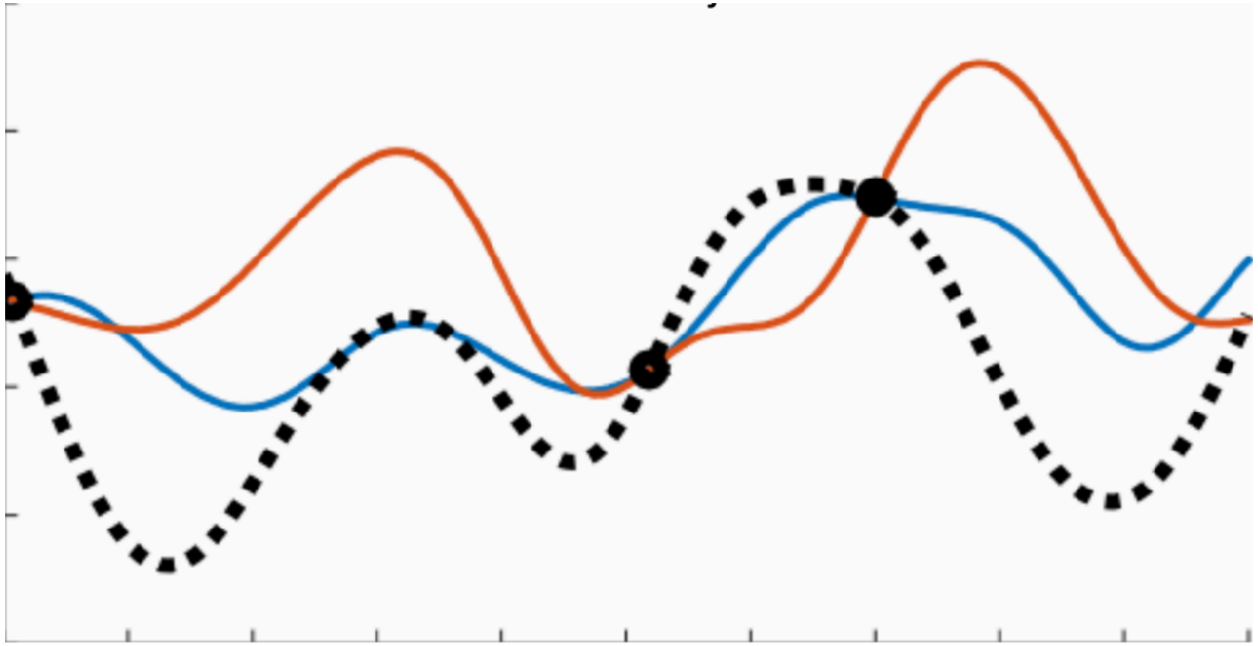
Gaussian Process



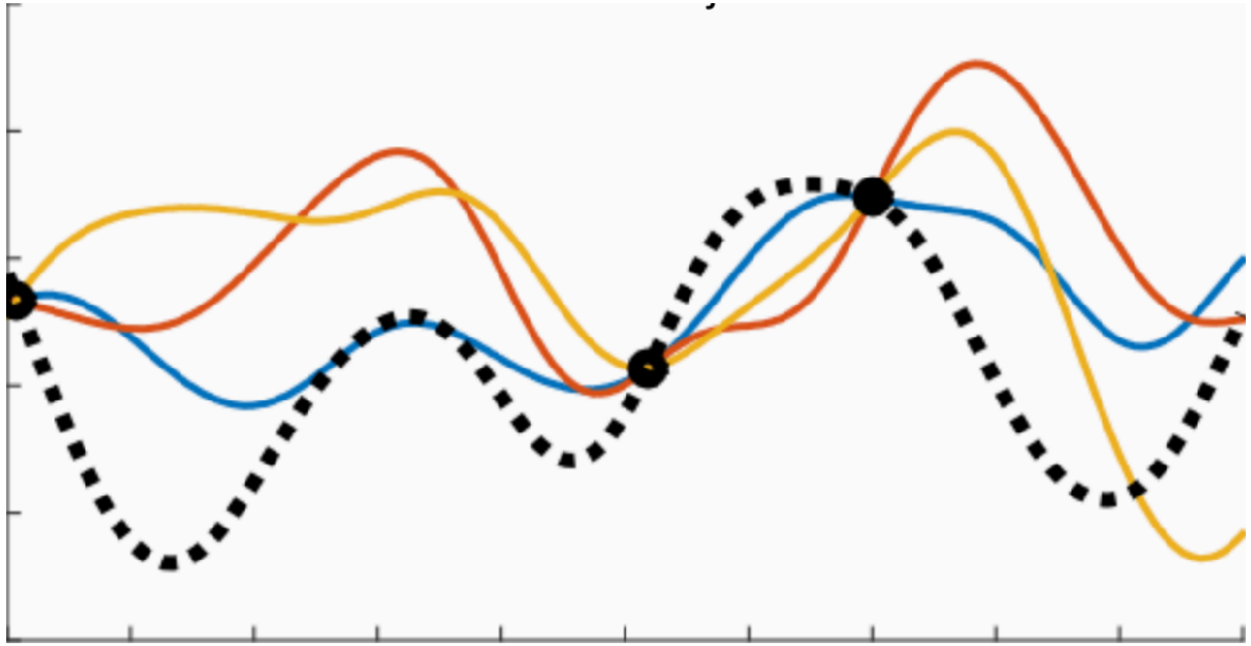
Gaussian Process



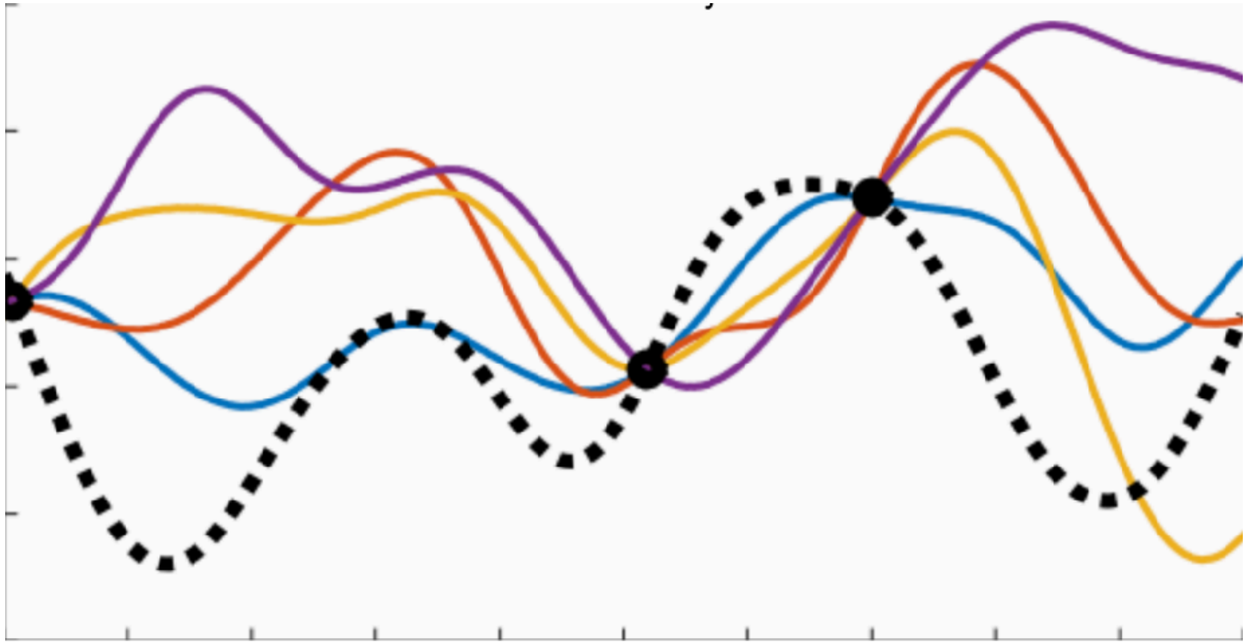
Gaussian Process



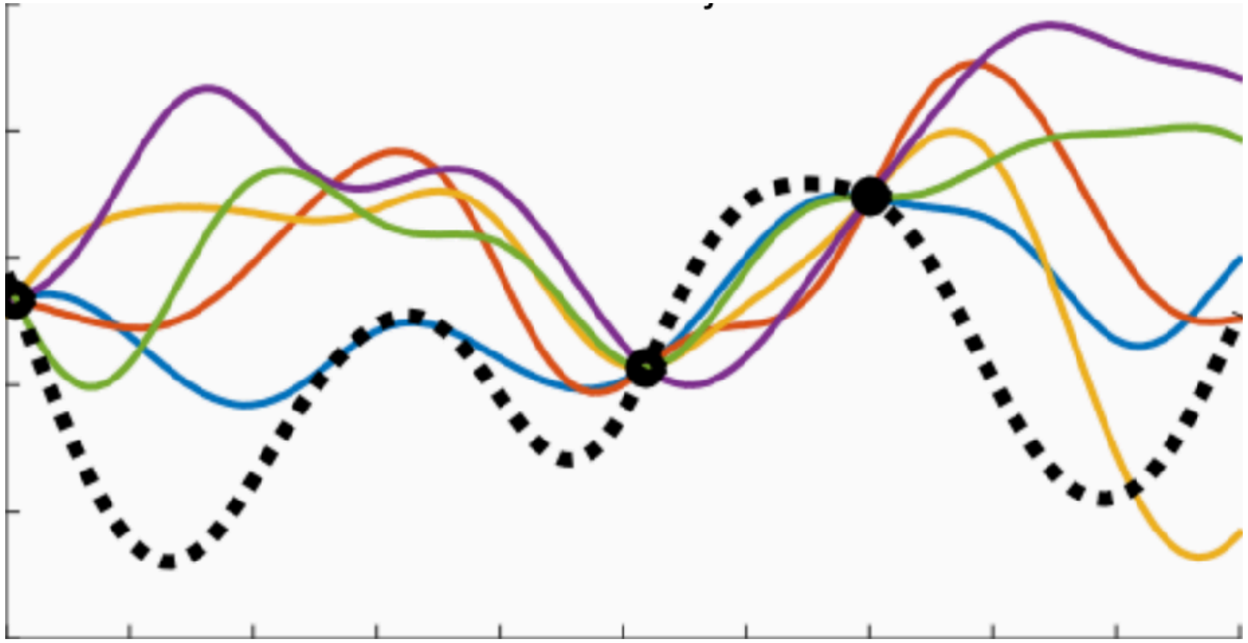
Gaussian Process



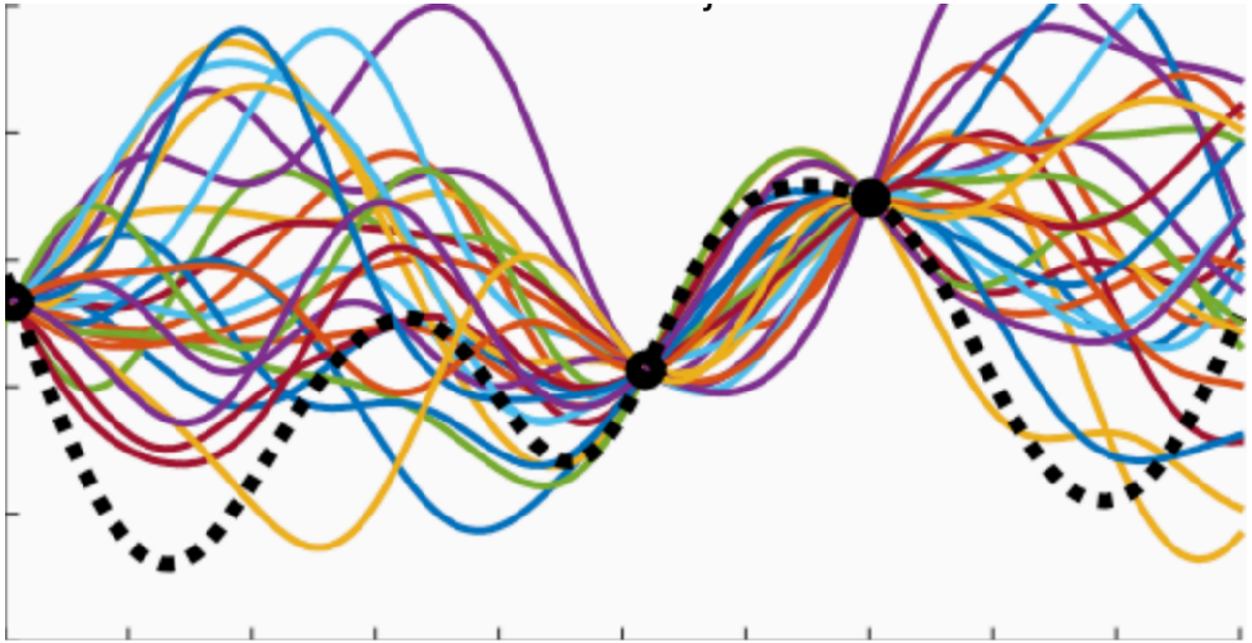
Gaussian Process



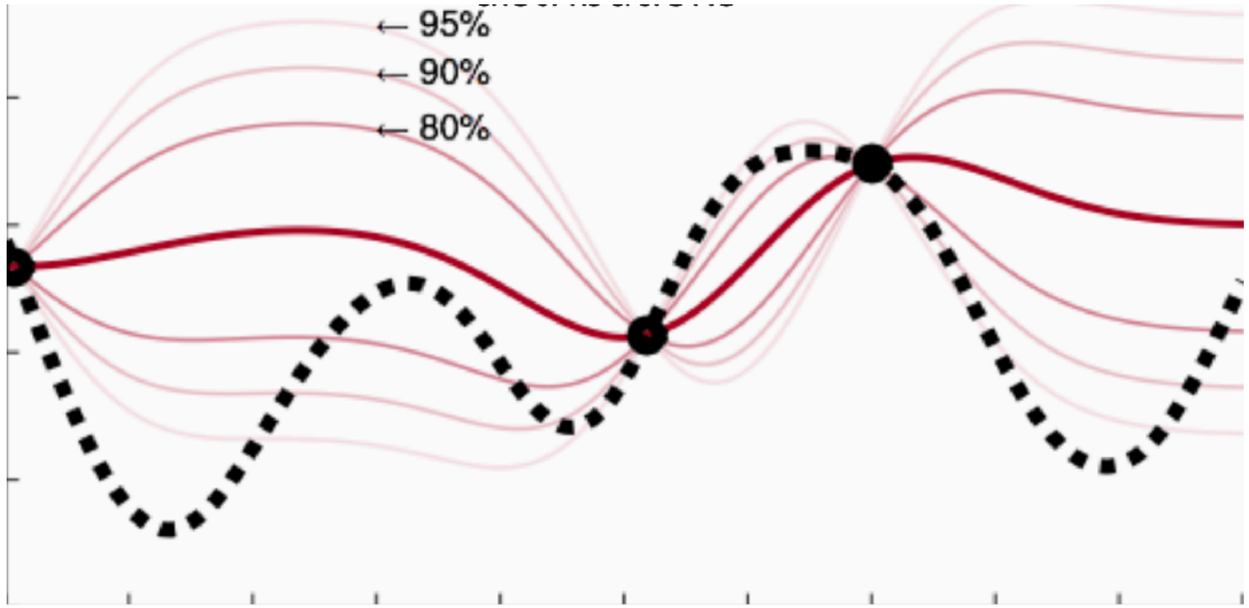
Gaussian Process



Gaussian Process



Gaussian Process



GP_n

$$\mu_n(x) = \kappa^T(\mathbf{K} + \sigma^2_{noise}\mathbf{I})^{-1}Y$$

$$\kappa_n(x, x') = \kappa(x, x') - \kappa^T(\mathbf{K} + \sigma^2_{noise}\mathbf{I})^{-1}\kappa'$$

Covariance Kernel Function

- Squared exponentials (SE)
- “n/2" Matern kernels

Covariance Kernel Functions

- Squared Exponential:

$$k_i(x_i - x_i') = e^{-\left(\frac{x_i - x_i'}{\theta_i}\right)^2}$$

- Matern Function $\nu = \frac{3}{2}$:

$$k_i(x_i - x_i') = \left(1 + \sqrt{3}\left|\frac{x_i - x_i'}{\theta_i}\right|\right) e^{-\sqrt{3}\left|\frac{x_i - x_i'}{\theta_i}\right|}$$

- Matern Function $\nu = \frac{5}{2}$:

$$k_i(x_i - x_i') = \left(1 + \sqrt{5}\left|\frac{x_i - x_i'}{\theta_i}\right| + \frac{5}{3}\left|\frac{x_i - x_i'}{\theta_i}\right|^2\right) e^{-\sqrt{5}\left|\frac{x_i - x_i'}{\theta_i}\right|}$$

Covariance Kernel Functions

- Squared Exponential:

$$k_i(x_i - x_i') = e^{-\left(\frac{x_i - x_i'}{\theta_i}\right)^2}$$

- Matern Function $\nu = \frac{3}{2}$:

$$k_i(x_i - x_i') = \left(1 + \sqrt{3}\left|\frac{x_i - x_i'}{\theta_i}\right|\right) e^{-\sqrt{3}\left|\frac{x_i - x_i'}{\theta_i}\right|}$$

- Matern Function $\nu = \frac{5}{2}$:

$$k_i(x_i - x_i') = \left(1 + \sqrt{5}\left|\frac{x_i - x_i'}{\theta_i}\right| + \frac{5}{3}\left|\frac{x_i - x_i'}{\theta_i}\right|^2\right) e^{-\sqrt{5}\left|\frac{x_i - x_i'}{\theta_i}\right|}$$

Acquisition Function

$$GP_{prior} + Data_n \rightarrow^{Bayes} GP_n \rightarrow^{?} x_{n+1}$$

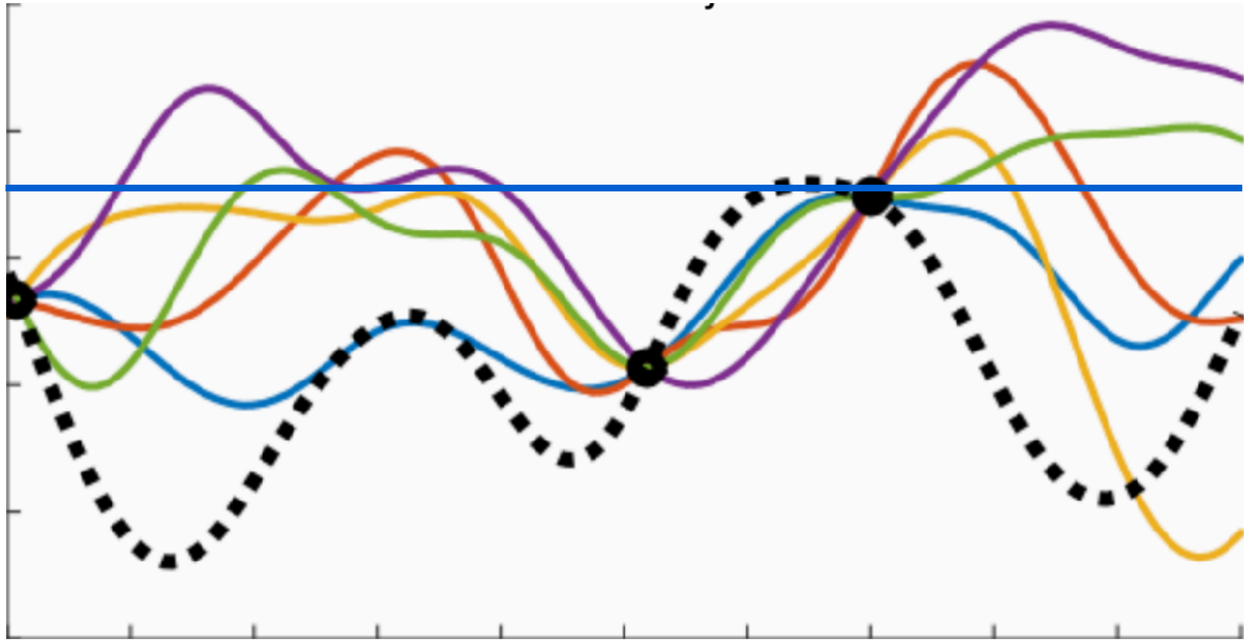
Acquisition Function

$$GP_{prior} + Data_{n+1} \rightarrow^{Bayes} GP_{n+1} \rightarrow^{?} x_{n+2}$$

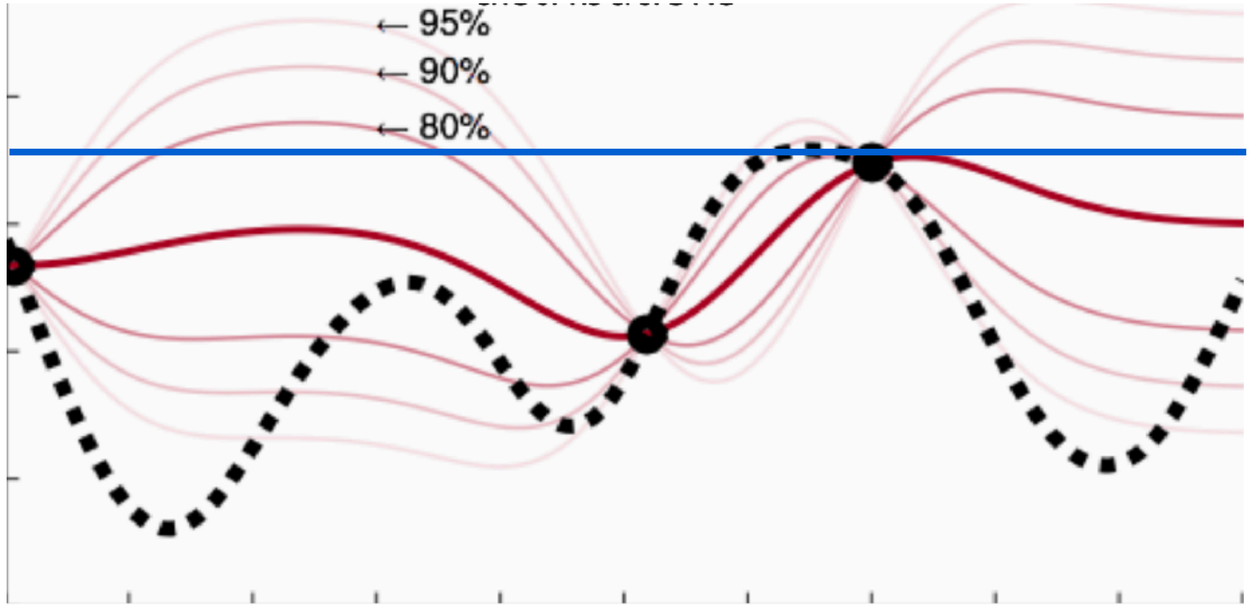
Acquisition Functions

- Thompson Sampling from the posterior GP (TS)
- Probability of Improvement (PI)
- Upper Confidence Bound (UCB)
- Expected Improvement (EI)

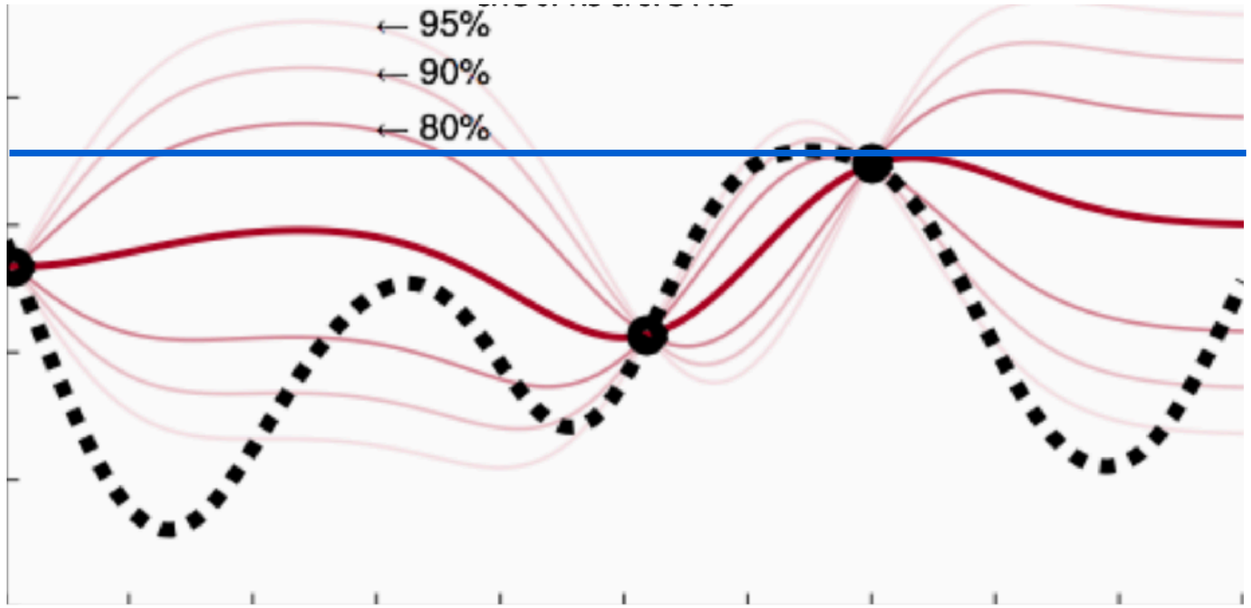
Thompson Sampling



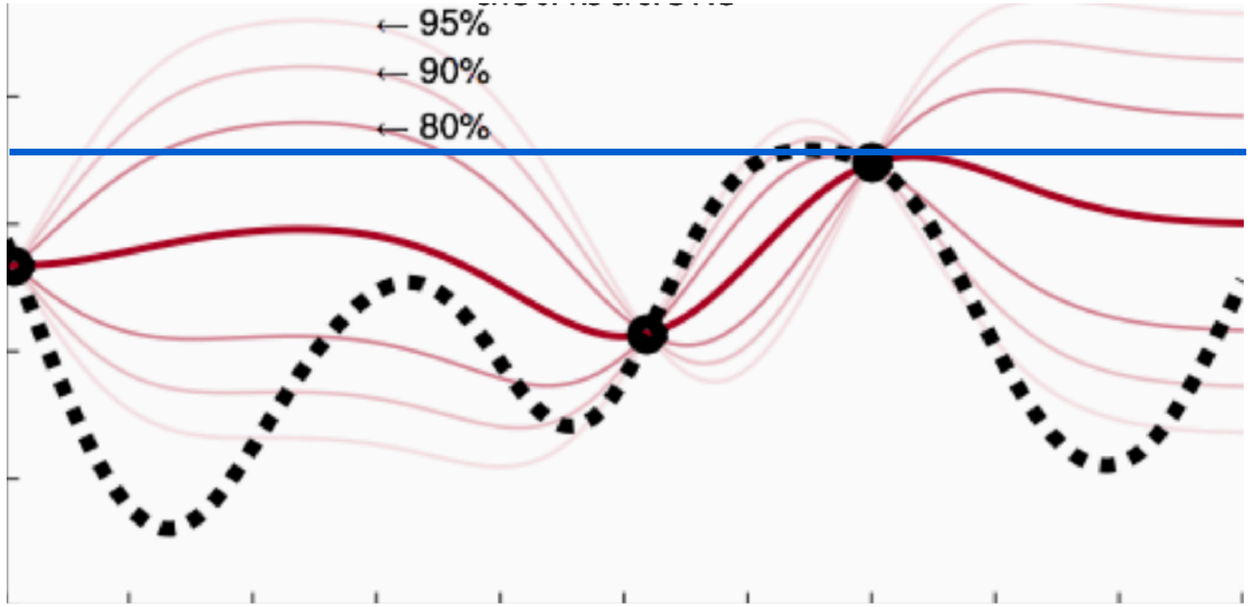
Probability of Improvement



Upper Confidence Bound



Expected Improvement



Acquisition Function

- Thompson Sampling from the posterior GP (TS)
- Probability of Improvement (PI)
- Upper Confidence Bound (UCB)
- **Expected Improvement (EI)**

Maximizing the Acquisition Function

- piecewise infinitely smooth
- gradient-based techniques work
- modified Monte-Carlo techniques are typically used

Optimizing Performance Parameters

```
myExpt = Optimizer.declareDevice(Parm1: {Int, Min1, Max1},
                                 Parm2: {Real, Min2, Max2},
                                 Parm3: {Enum, enum1, enum2, enum3}
                                 …)

myExpt.setSLA(…)                                    // set performance SLA
myExpt.setTerminationCriteria(…)                    // set termination criteria

while (!myExpt.shouldTerminate()) {
   parmSuggestion = myExpt.suggest()                // get another test suggestion
   newRun = myDevice.test(parmSuggestion)           // test device at given setting
   if (myExpt.isValid(newRun)) {                    // is SLA met?
     myExpt.update(parmSuggestion, newRun)          // update w/new result
   }
}

return myExpt.bestConfig()
```

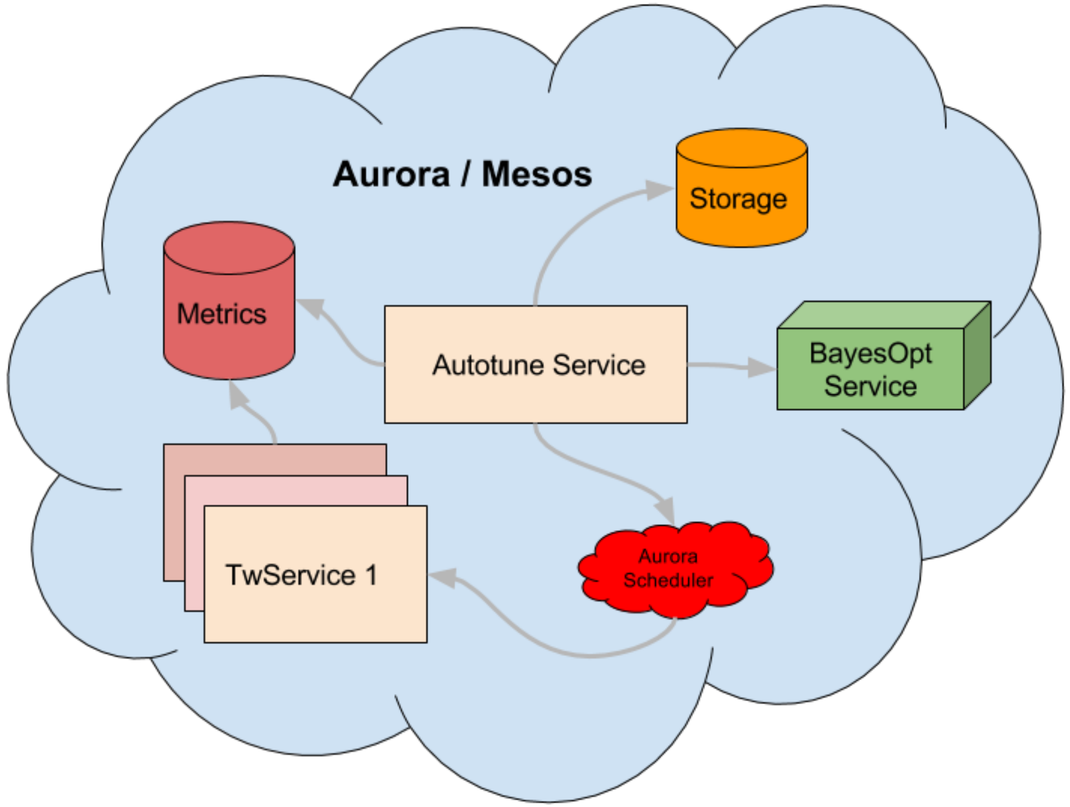
Bayesian Optimization

```
myExpt = Optimizer.declareDevice(Parm1: {Int, Min1, Max1},
                                 Parm2: {Real, Min2, Max2},
                                 Parm3: {Enum, enum1, enum2, enum3}
                                 …)

myExpt.setSLA(…)                                  // set performance SLA
myExpt.setTerminationCriteria(…)                  // set termination criteria

while (!myExpt.shouldTerminate()) {
   parmSuggestion = myExpt.suggest()              // get another test suggestion
   newRun = myDevice.test(parmSuggestion)         // test device at given setting
   if (myExpt.isValid(newRun)) {                  // is SLA met?
     myExpt.update(parmSuggestion, newRun)        // update w/new result
   }
}

return myExpt.bestConfig()
```

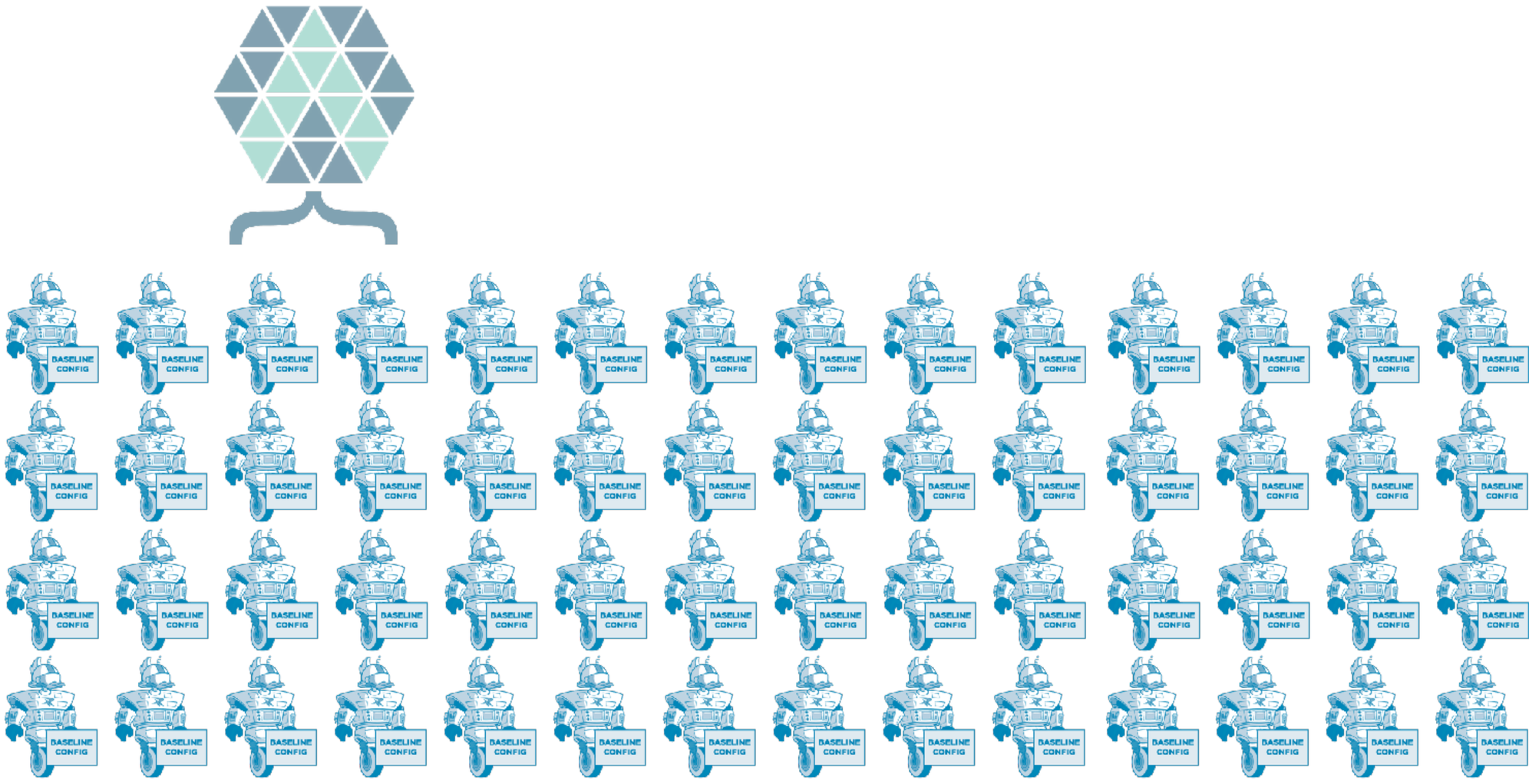
Constraints

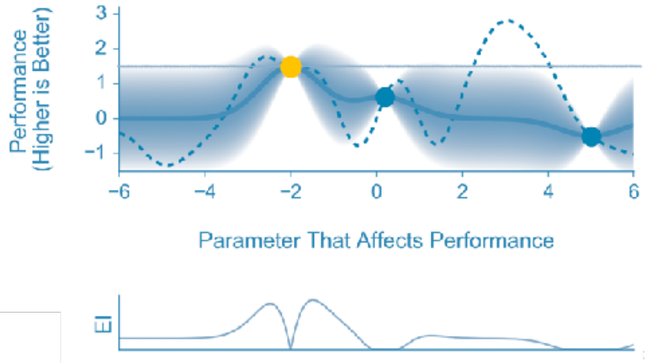
```
myExpt = Optimizer.declareDevice(Parm1: {Int, Min1, Max1},
                                 Parm2: {Real, Min2, Max2},
                                 Parm3: {Enum, enum1, enum2, enum3}
                                 …)

myExpt.setSLA(…)                                  // set performance SLA
myExpt.setTerminationCriteria(…)                  // set termination criteria

while (!myExpt.shouldTerminate()) {
   parmSuggestion = myExpt.suggest()              // get another test suggestion
   newRun = myDevice.test(parmSuggestion)         // test device at given setting
   if (myExpt.isValid(newRun)) {                  // is SLA met?
     myExpt.update(parmSuggestion, newRun)        // update w/new result
   }
}

return myExpt.bestConfig()
```

AUTOTUNE AS A SERVICE

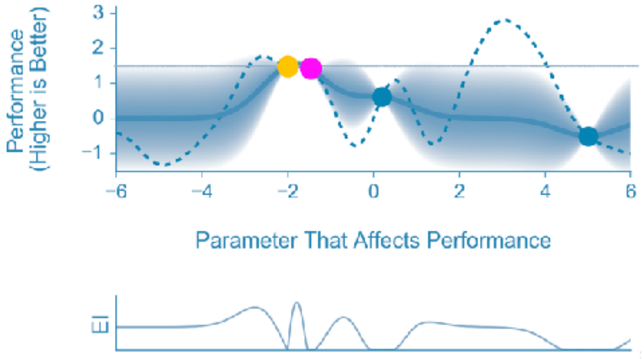



BASELINE
CONFIG

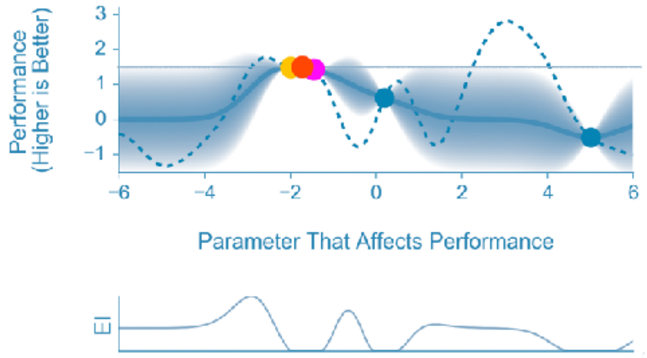

BASELINE
CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

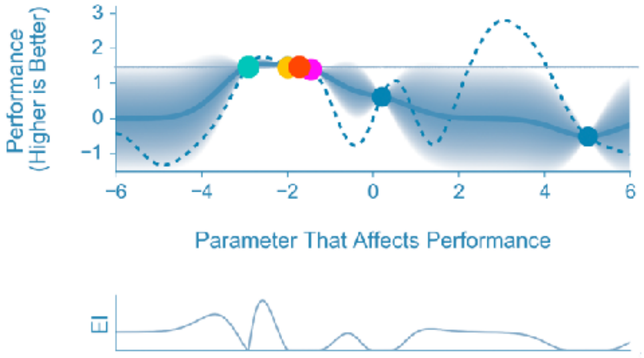

BASELINE CONFIG
EVAL 001 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

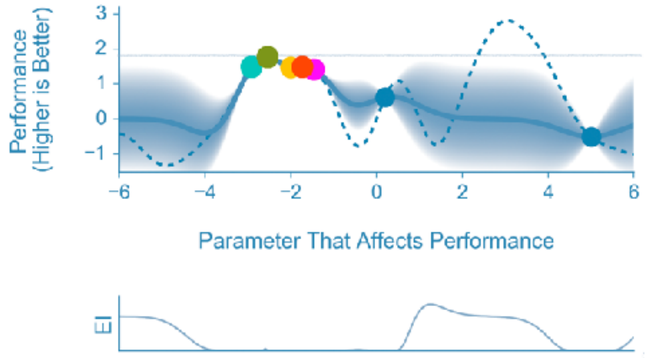

BASELINE CONFIG
EVAL 002 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

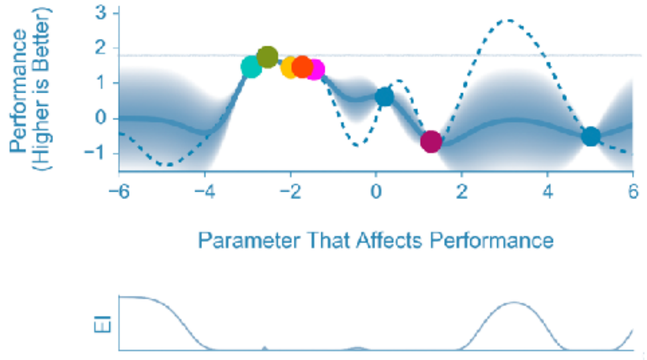

BASELINE CONFIG
EVAL 005 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

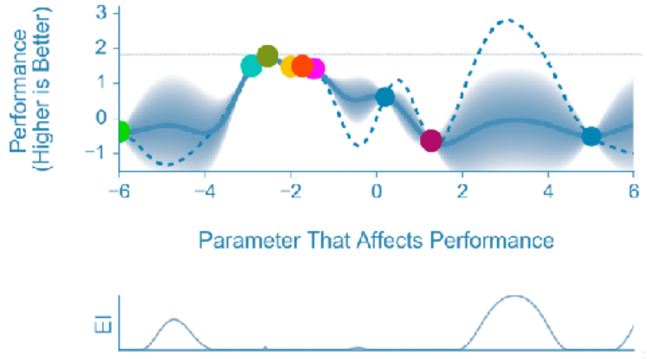

BASELINE CONFIG
EVAL 004 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

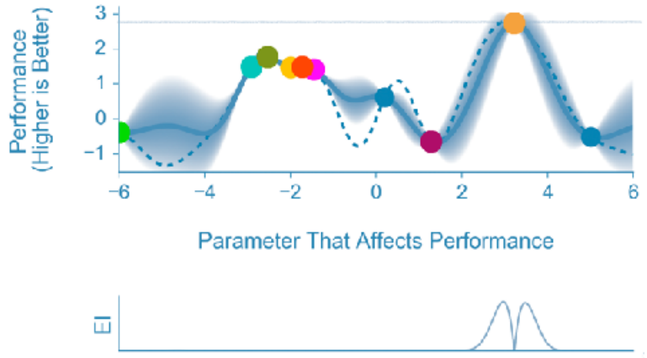

BASELINE CONFIG
EVAL 005 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

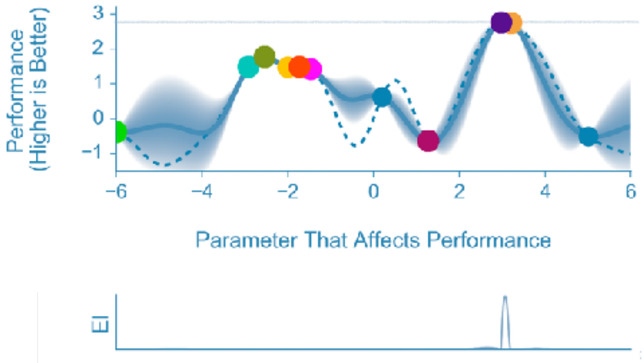

BASELINE CONFIG
EVAL 006 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

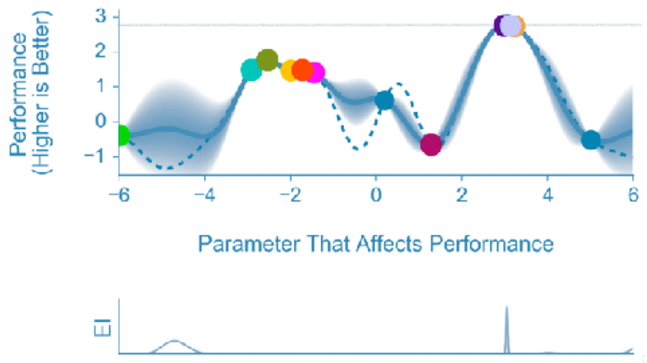

BASELINE CONFIG
EVAL 007 CONFIG


Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

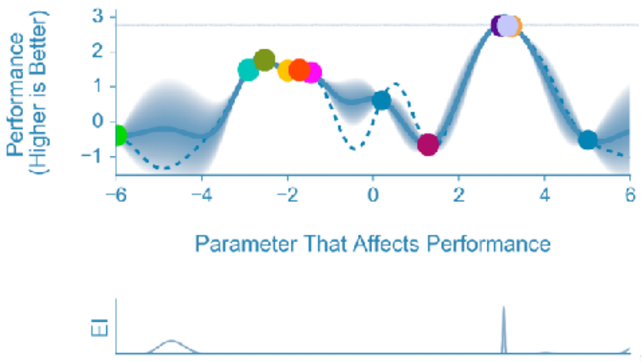

BASELINE CONFIG
EVAL 006 CONFIG

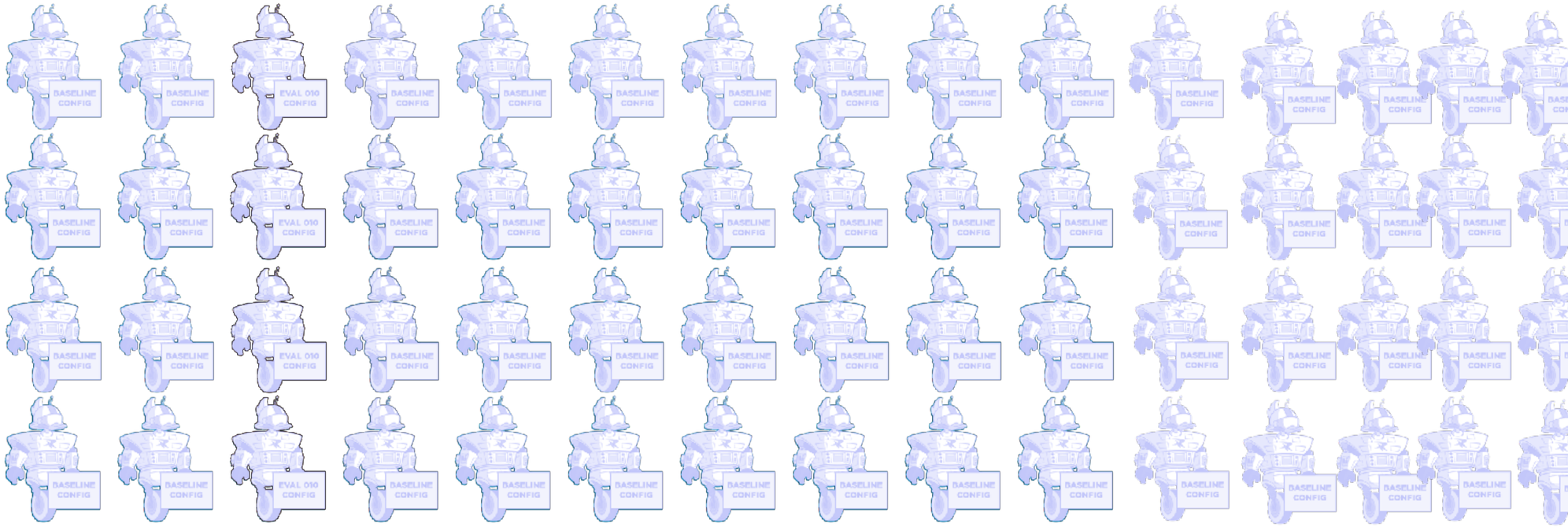

Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI


BASELINE CONFIG
EVAL 009 CONFIG

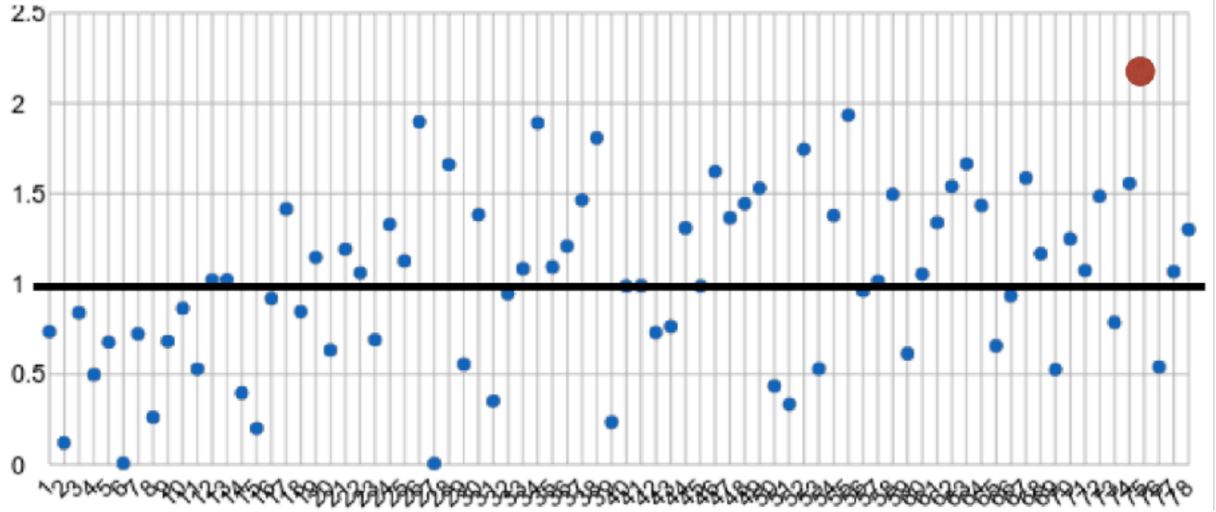

Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI

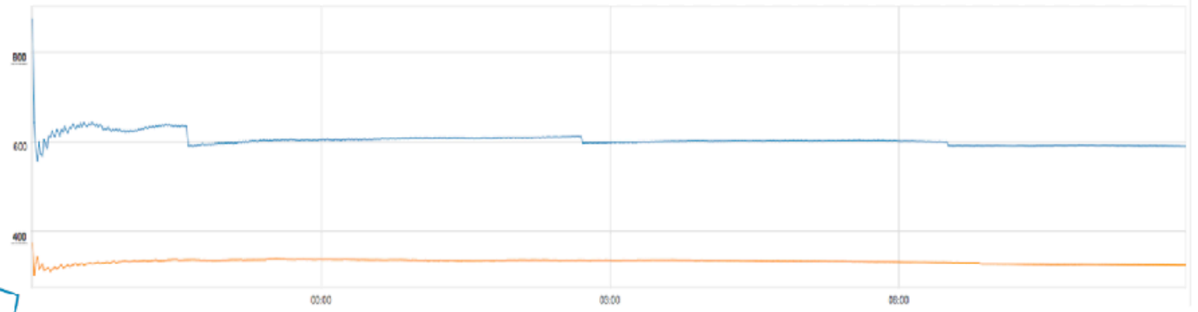

BASELINE CONFIG
EVAL 010 CONFIG

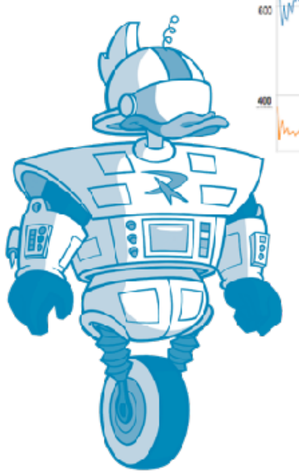

Performance
(Higher is Better)
3
2
1
0
−1
−6
−4
−2
0
2
4
6
Parameter That Affects Performance
EI


BASELINE CONFIG
EVAL 010 CONFIG

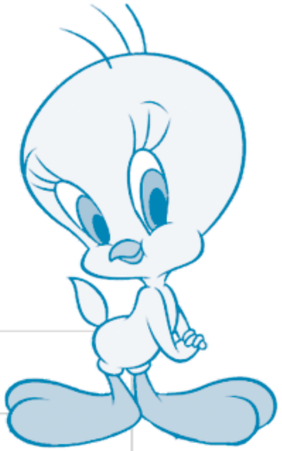
GizmoDuck & Garbage Collection Overhead via Tuning JVM Parameters

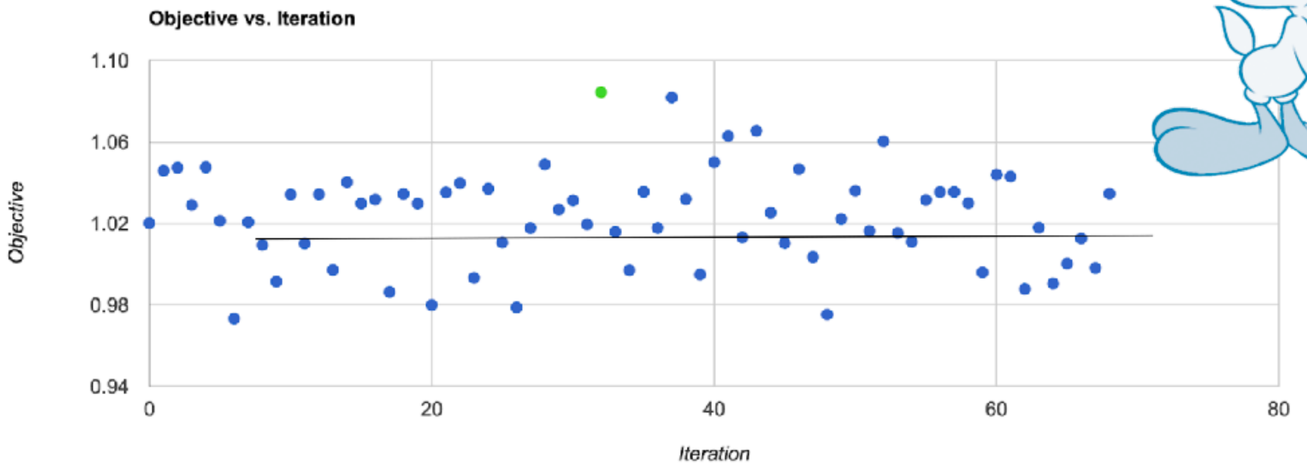


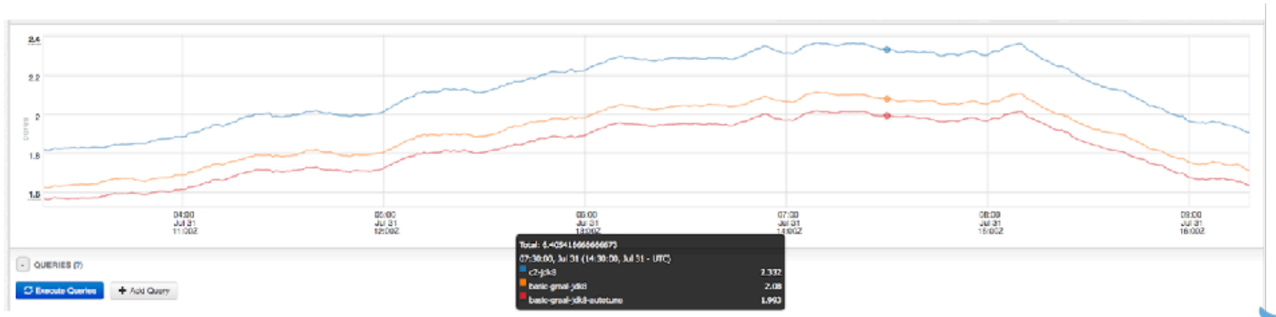




TweetyPie & CPU Utilization via Tuning Graal JIT Parameters








OBSERVABILITY