Week 8: Bayesian Optimization

AIM-5014-1A: Experimental Optimization

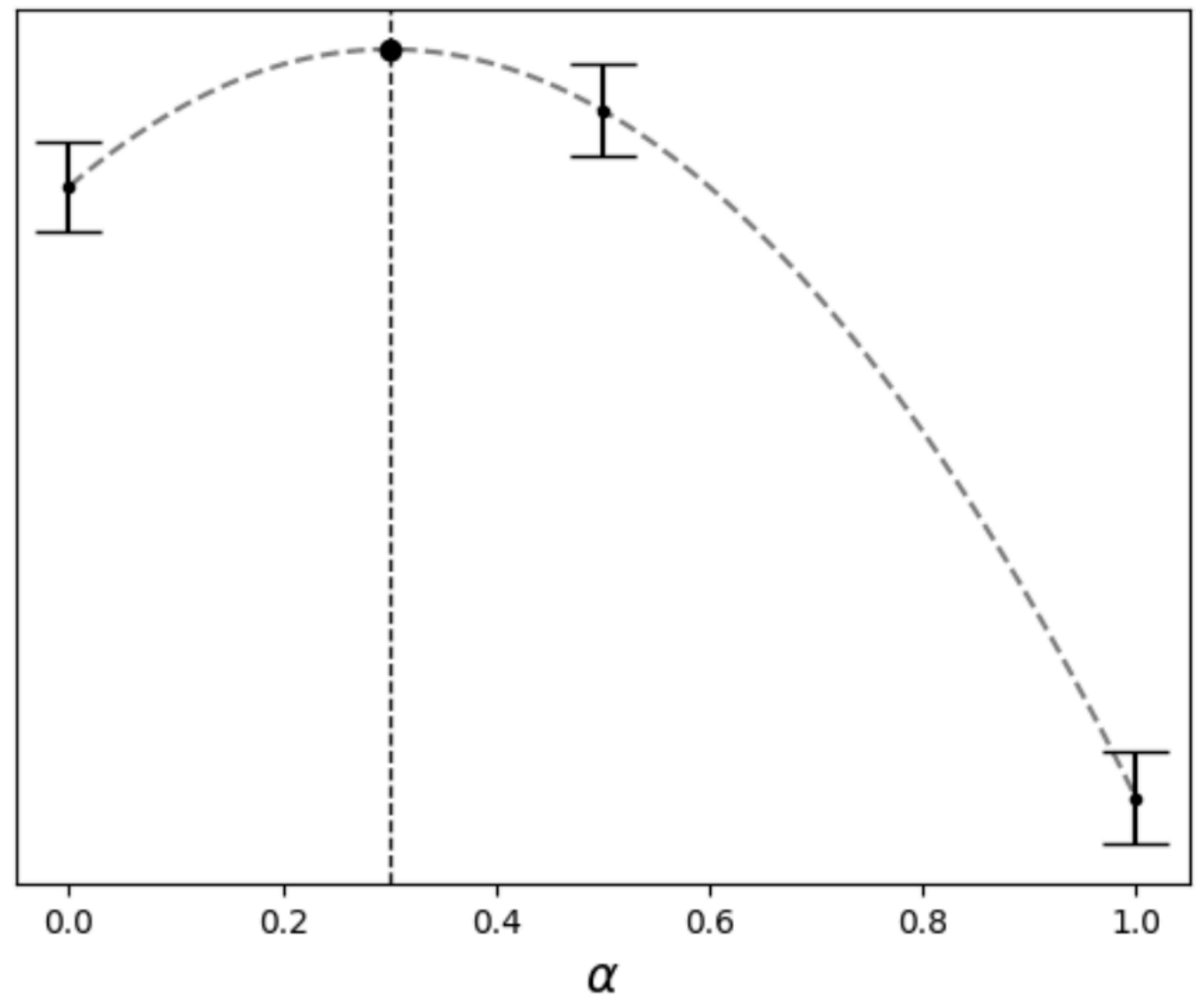
David Sweet // 20230720

Review: LLN, CLT, A/B Testing

- As $N \to \infty$, $\bar{y} \to E[BM]$ (LLN)
  - CLT: $\bar{y} \sim \mathcal{N}(E[BM], \sigma^2)$, “measured BM is gaussian”
- **Design**: $N \geq \left(\frac{2.5\hat{\sigma}_\delta}{PS}\right)^2$
- **Measure**: Randomize, $\bar{\delta} = \bar{y}_B - \bar{y}_A$, $se = \sigma_\delta/\sqrt{N}$
- **Analyze**: Accept B if $\bar{\delta} > PS$ and $\frac{\bar{\delta}}{se} \geq 1.64$ (check guardrails)
- **False Positive Traps**: Early stopping, multiple comparisons (use Bonferroni)

Review: Response Surface Methodology

- Parameters:
  - categorial: discrete unordered, strings; ex: A/B
  - ordinal: discrete ordered, integers; ex: 1, 2, 3, …
  - continuous: double; ex., [0,1]   <== RSM
- Surrogate, $y(x)$, models response surface, $E[y(x)]$
- Find optimum, $x^* = \arg\max_x y(x)$, and validate by A/B test

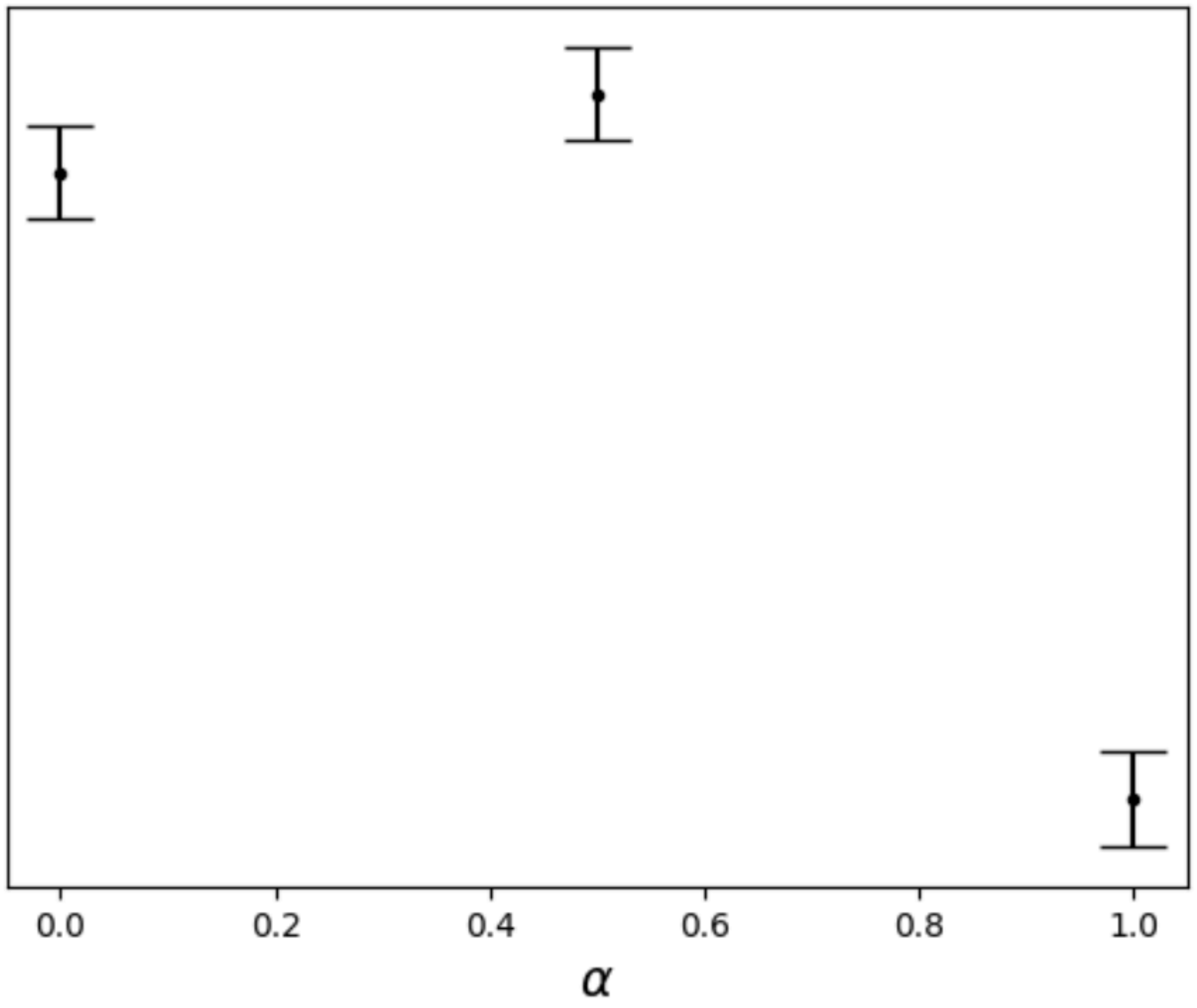


Case: Song recommender (again)

- In prod (A): Ranking songs by p_{listen} = P{user will listen until the end}
- In dev (B): Ranking songs by p_{like} = P{user will click song's like button}
- Rank by: $score = \alpha p_{\text{listen}} + (1 - \alpha) p_{\text{like}}$
- Today: Try Bayesian Optimization instead of RSM

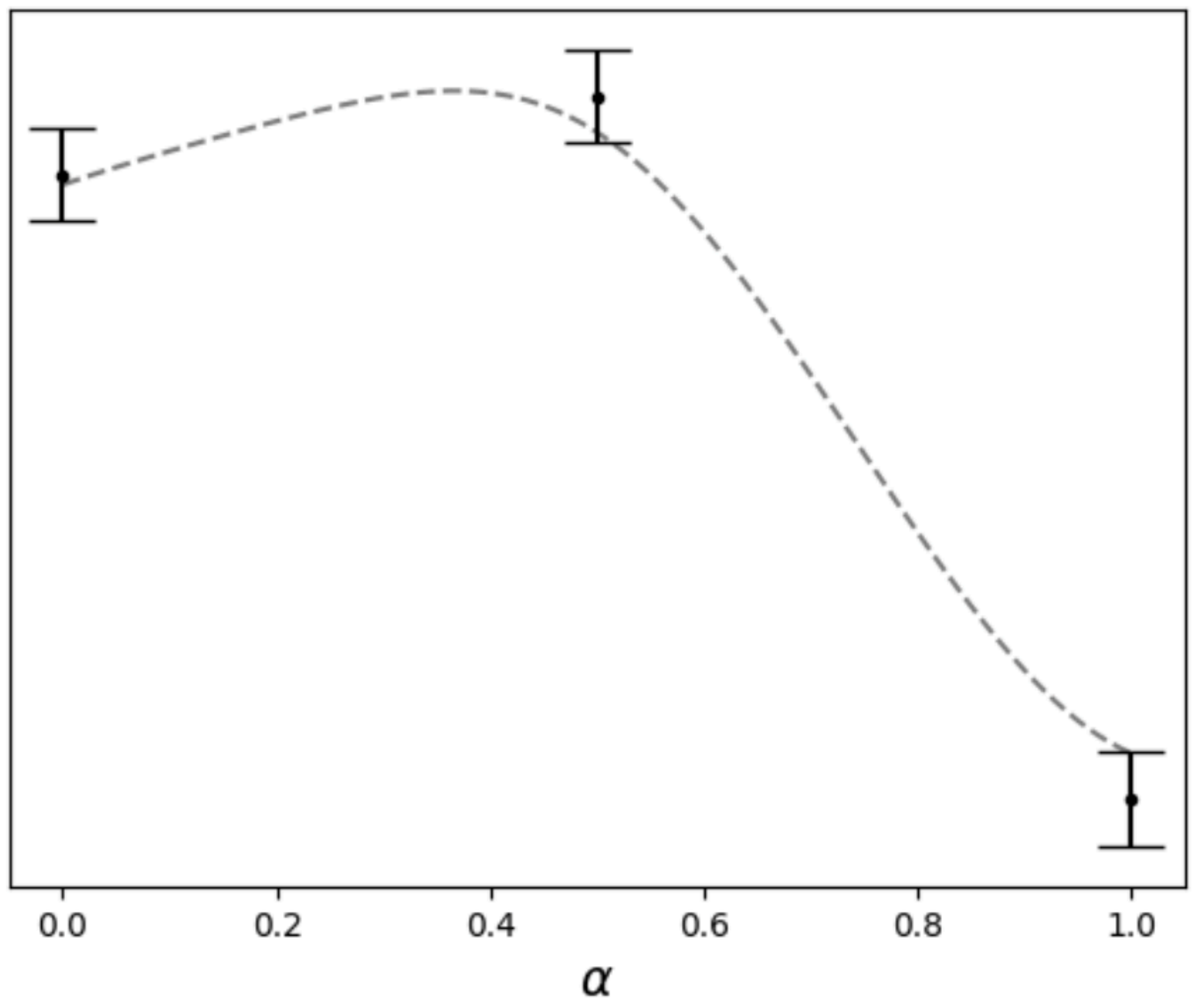
BO: Initialization

- Start with same initial design
- Measure BM at
  - $\alpha \in \{0.0, 0.5, 0.1\}$



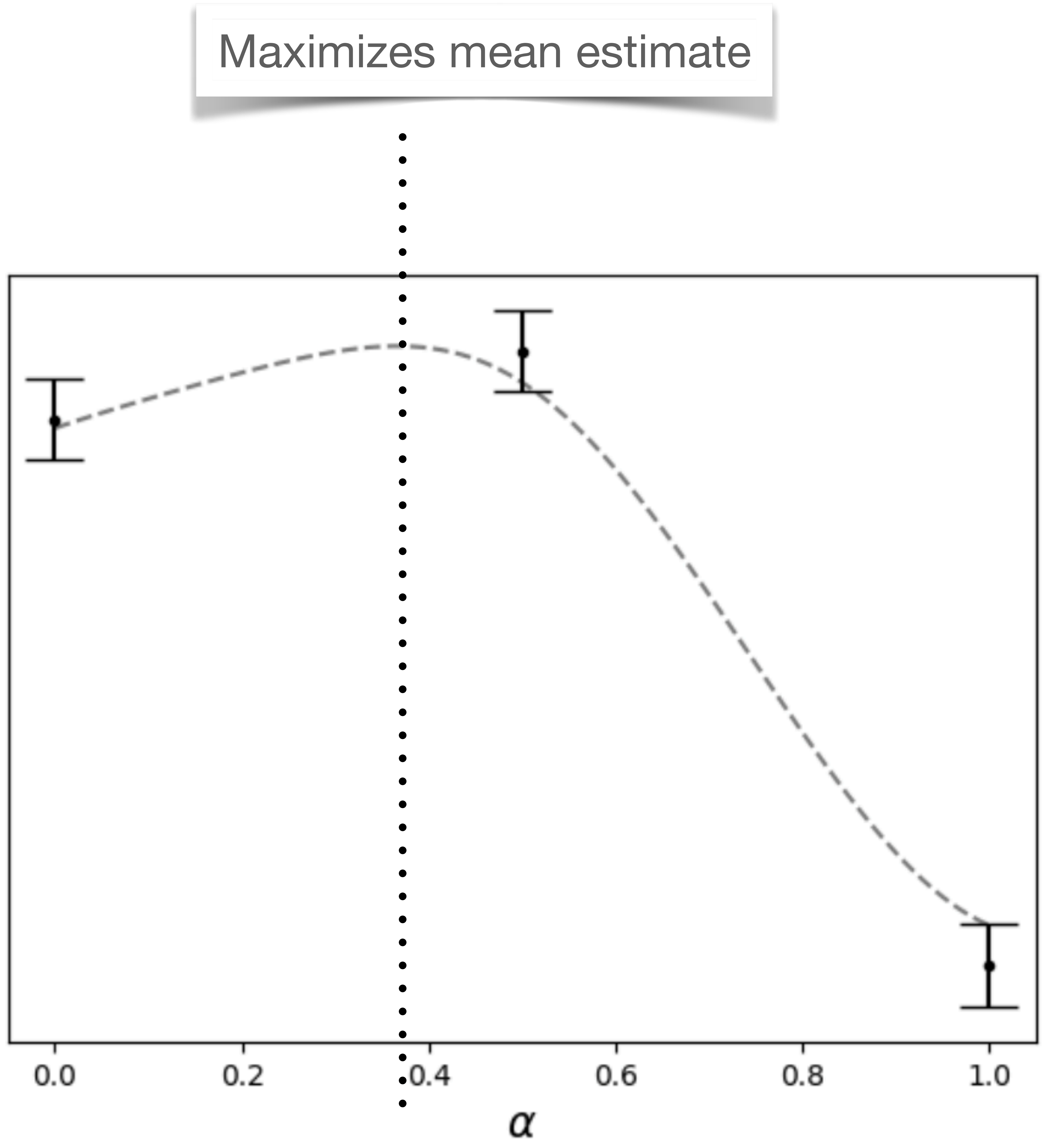
BO: Surrogate model

- BO surrogate
- GPR: Gaussian Process Regression
  - Not quite a parabola, but fits
- Model of response surface, $BM(\alpha)$



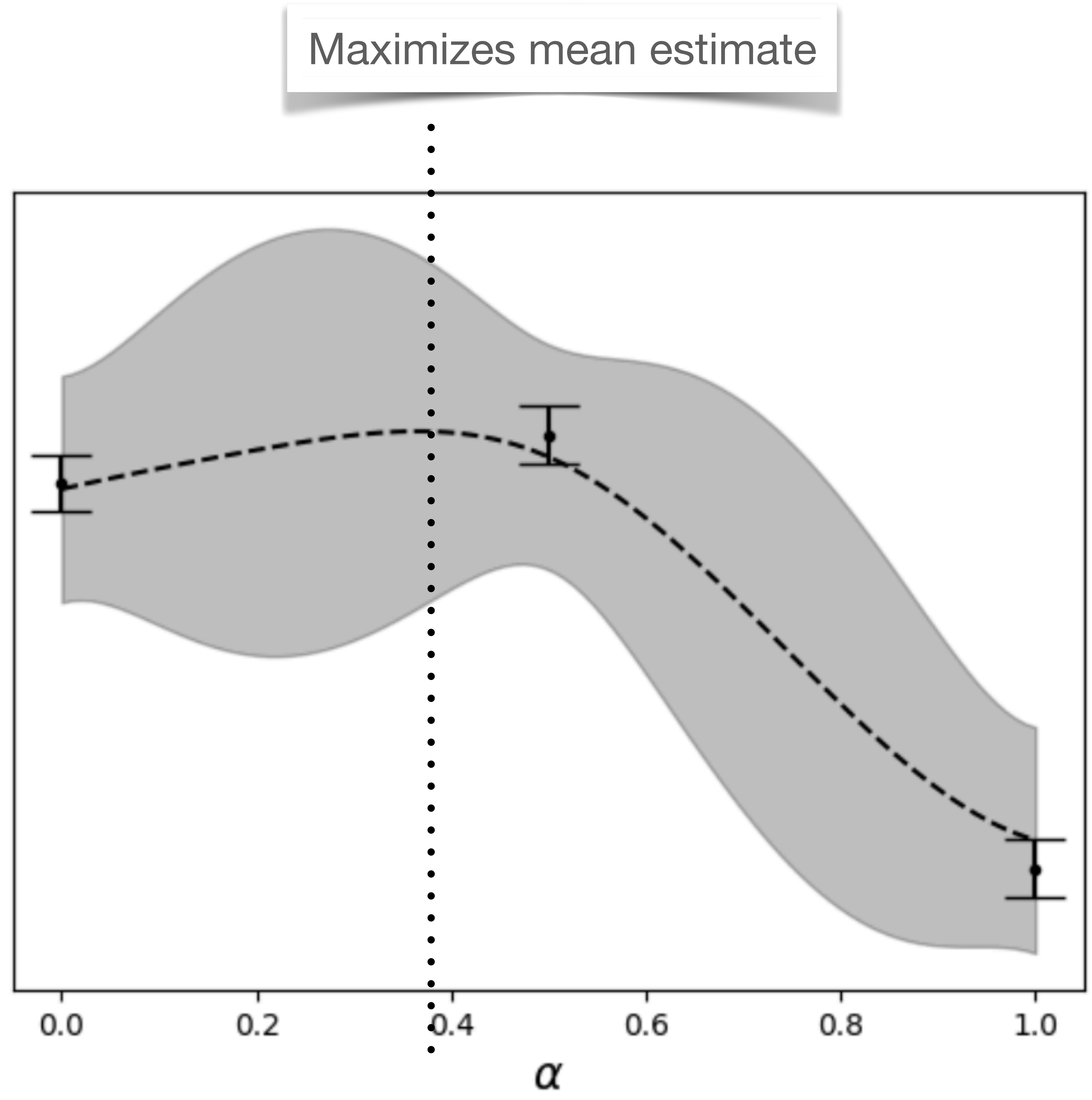
BO: Surrogate model

- Find α that maximizes dashed curve (mean estimate)
- What is true shape of response surface?
  - Could be anything
  - Uncertain about shape



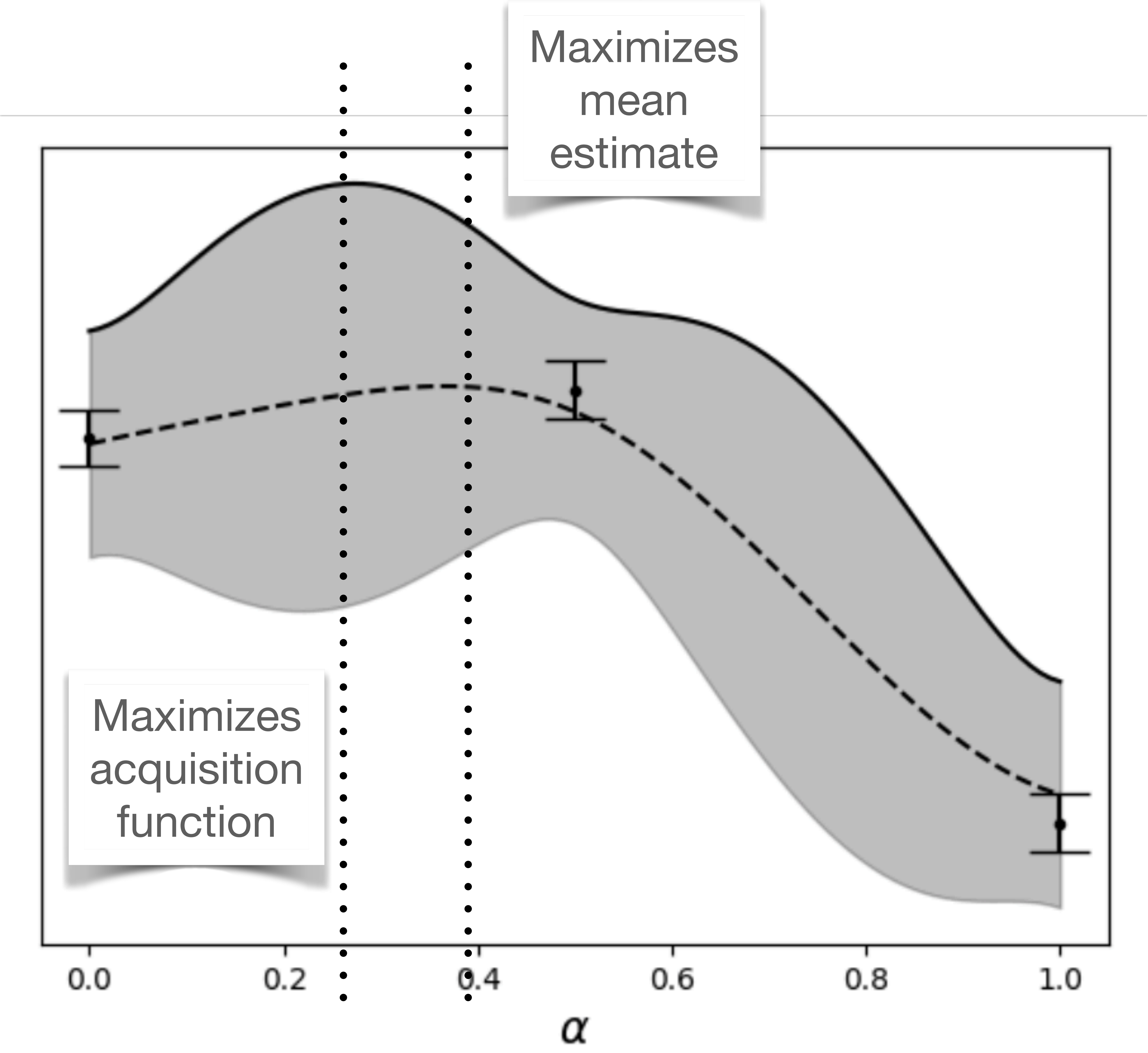
BO: Surrogate model

- GPR also models uncertainty
- Left of dashed line is very uncertain
- Could fix that by measuring there



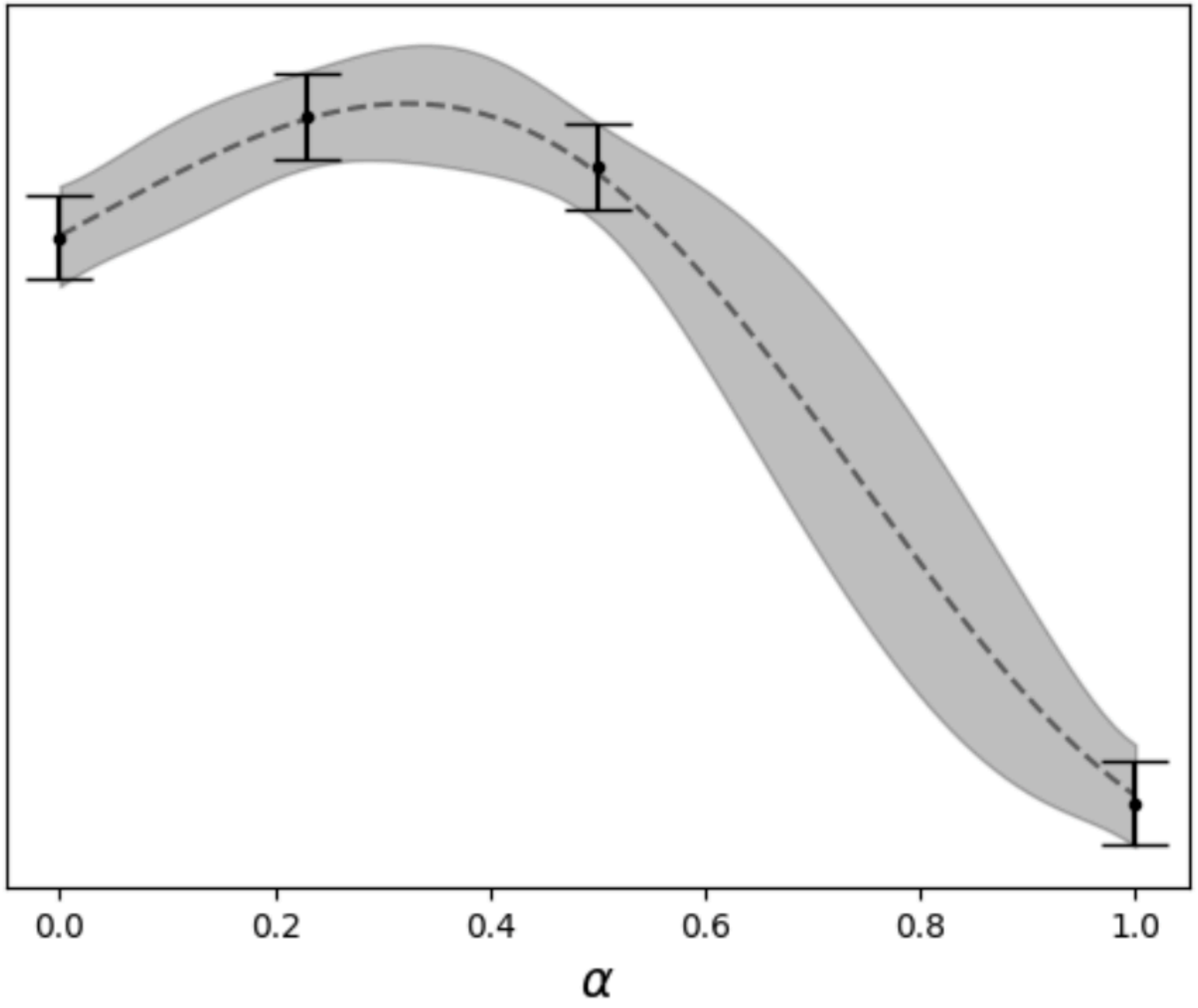
BO: Acquisition Function

- BO maximizes top of gray area
  - dark line is *acquisition function*
- High mean *and* high uncertainty



BO: Surrogate model

- More certainty about shape of response surface now
- More likely that surrogate optimum matches response surface optimum

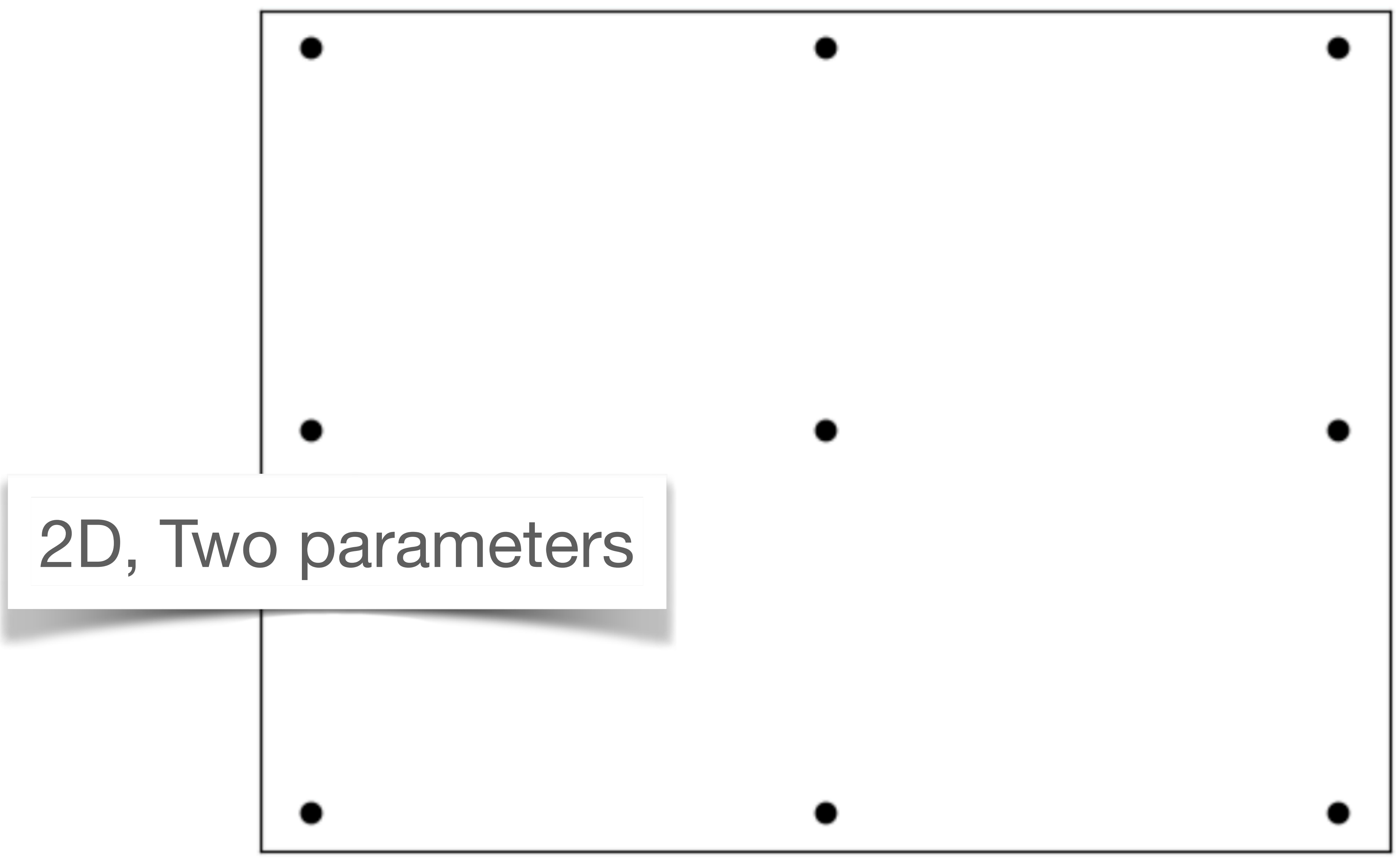


Bayesian Optimization Overview

- **Initialization**: Spread measurements out in parameter space
- **Surrogate**: Gaussian Process Regression (GPR)
  - Models mean and se
- **Design**: Optimize acquisition function
  - Determines next parameter value(s) to measure

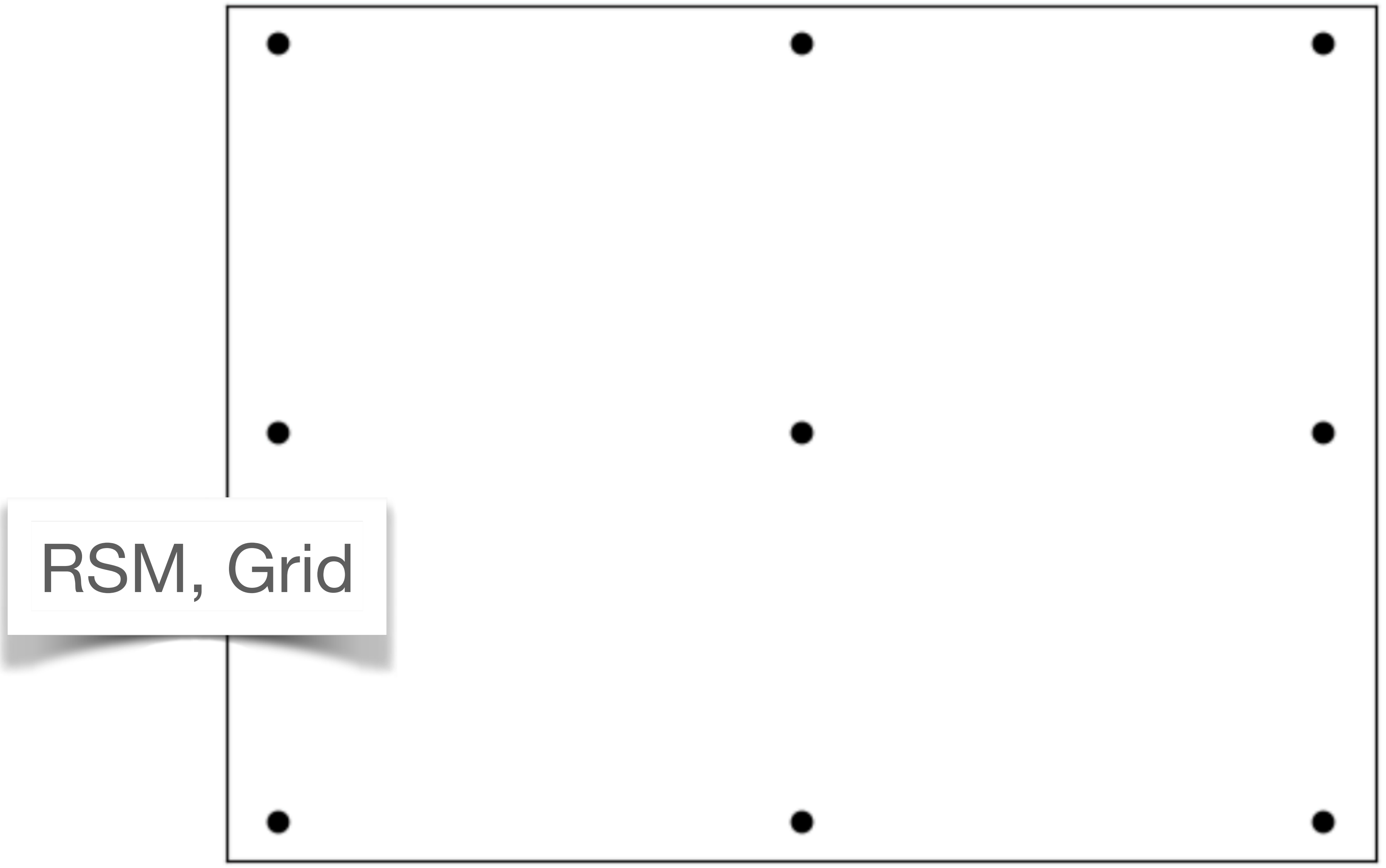
BO: Initialization

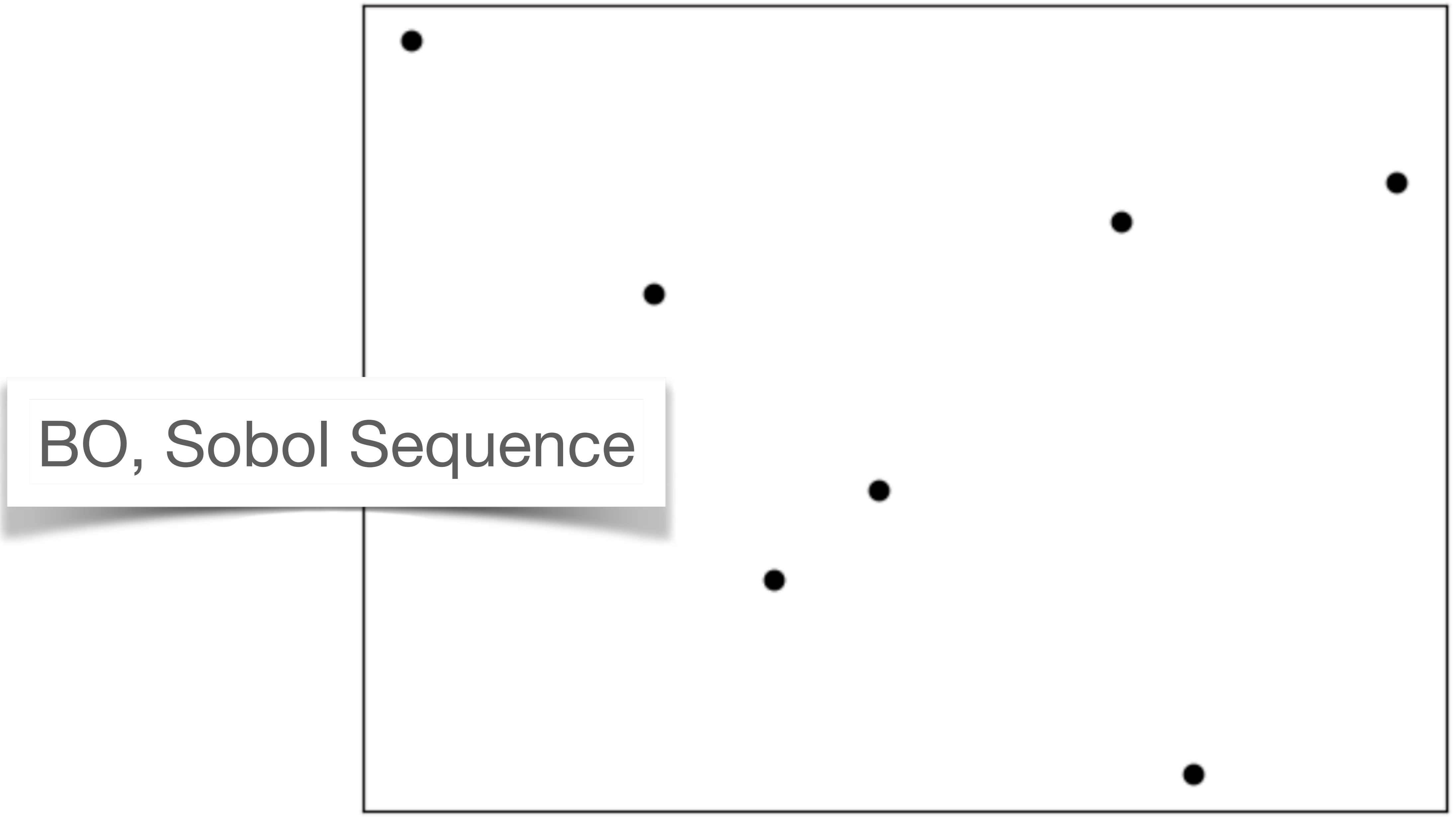
- Spread points out in parameter space
    - 1D: 3 points
    - 2D: $3 \times 3 = 3^2 = 9$ points
    - 3D: $3 \times 3 \times 3 = 3^3 = 27$ points
    - … dD: $3^d =$ too many points
- Curse of dimensionality



BO: Initialization

- K arms; $K = 3^d$
- Exponential in (curse of) d
- Solution: Space-filling sequence
  - K independent of d
- Choose K based on capacity to experiment

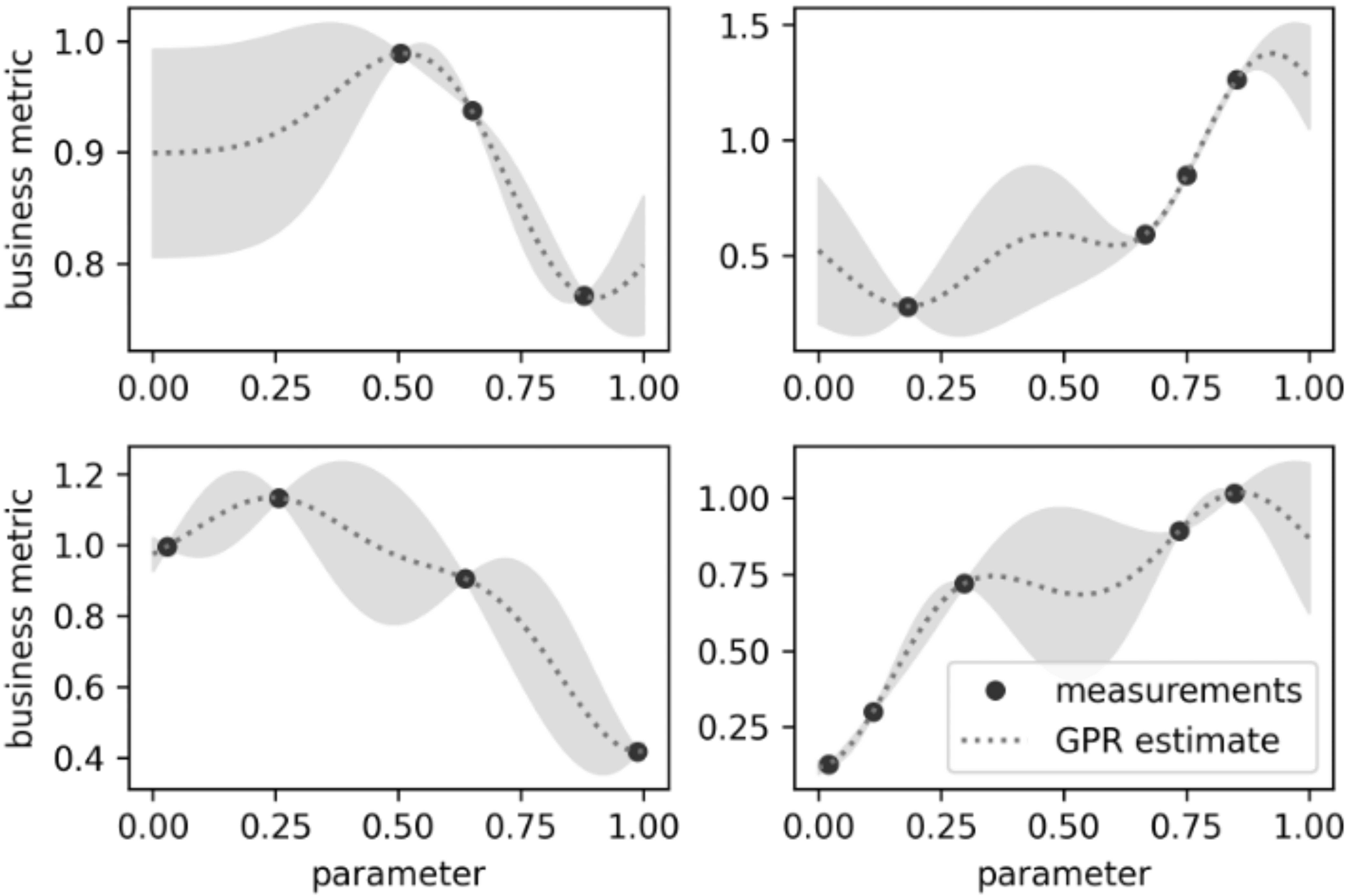




BO: Surrogate (GPR)

- Recall, RSM uses linear model
  - $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$
  - Engineer decides regressors
- Gaussian Process Regression (GPR)
  - Estimates are weighted averages of all measurements
  - “Fancy KNN”: Nearer neighbors are given more weight
  - No fitting, no betas; GPR is *non-parametric*

BO: Surrogate (GPR)

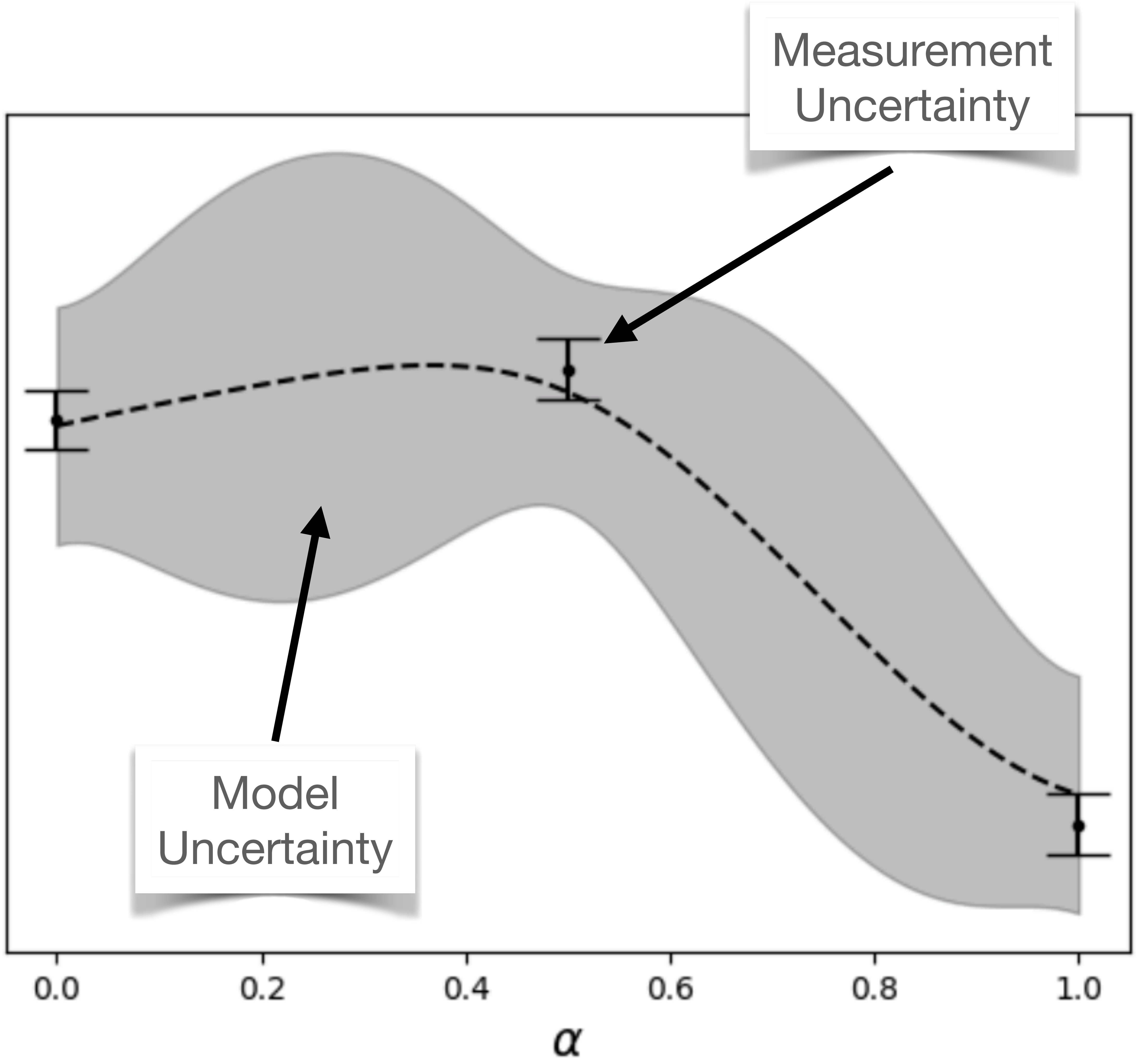


BO: Surrogate (GPR)

- Two types of uncertainty
  - Aleatoric: *measurement uncertainty*
  - Epistemic: *model uncertainty*
- Measurement uncertainty: Familiar se; Noise in your system
- Model uncertainty: Parameters where we haven’t measured yet

BO: Surrogate (GPR)

- Measurement uncertainty
  - Error bars
  - Decrease by increasing N
- Model uncertainty
  - Gray areas
  - Decrease by measuring a new parameter value



BO: GPR Equations

$$w(x, x_i) = e^{-(x-x_i)^2/(2s^2)},\ (K_x)_i = w(x, x_i),\ (K_{xx})_{ij} = w(x_i, x_j)$$

$$\hat{y}(\hat{x}) = K_x^T K_{xx}^{-1} y$$

$$\hat{\sigma}_y{}^2 = 1 - K_x^T K_{xx}^{-1} K_x$$

See Appendix C of *Experimentation for Engineers*

- x, y are vectors of measured parameters and BMs
- s is a hyperparameter, tuned to the measurements
- $\hat{x}$ is a query value, $\hat{y}$, $\hat{\sigma}_y$ are estimates

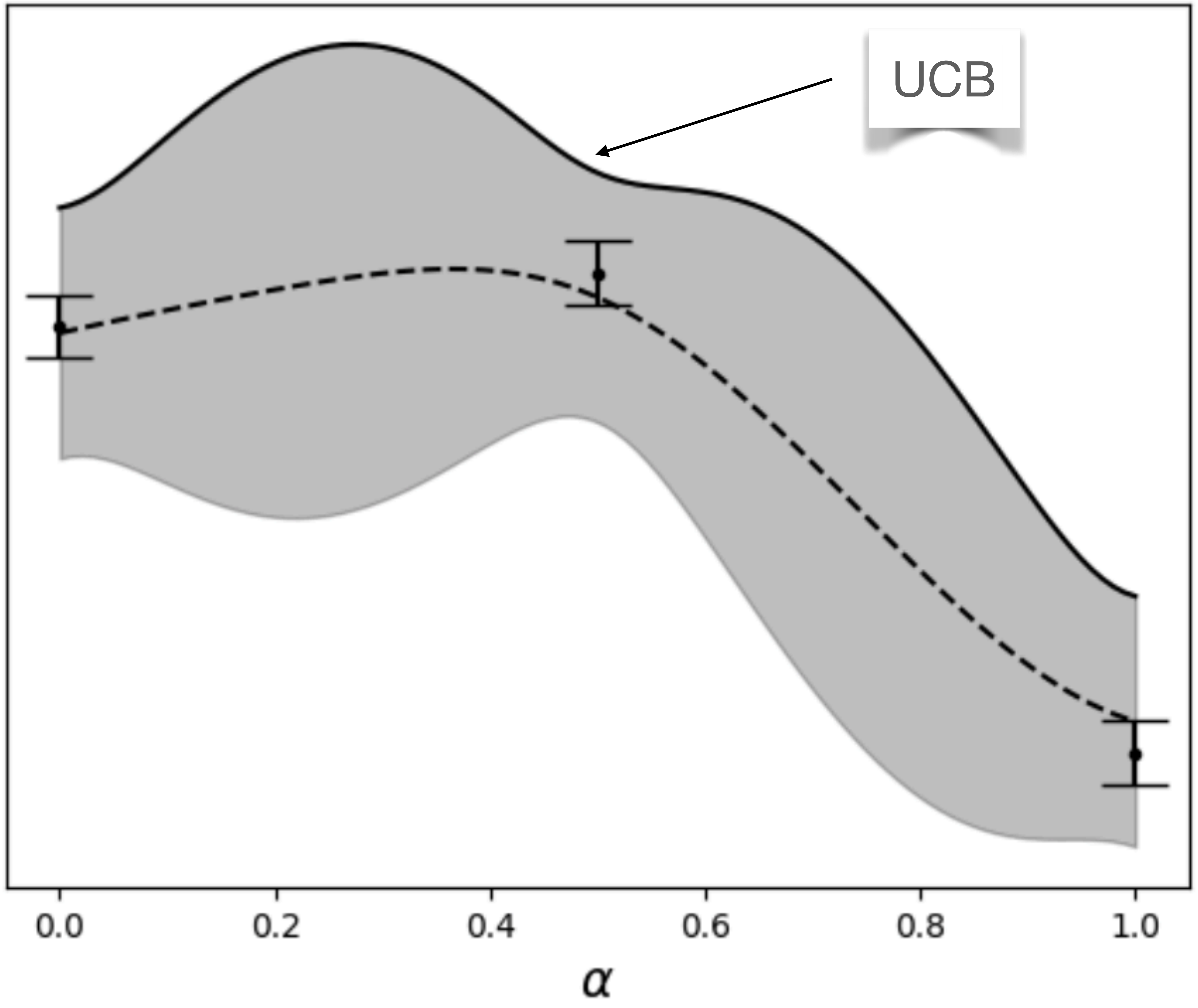
What puts the G in GPR?

And how is it a “process”?

- Model each value $y(x)$ as a gaussian distribution
- Model any collection of $\{y(x)\}$ as a multivariate gaussian distribution
  - x is continuous, so really an infinite-dimension gaussian distribution
- First considered as $y(t)$, where t is time. A process is something that changes over time. A gaussian process is one where y has a gaussian distribution that changes over time. Ex: a Brownian motion (continuous random walk)
- Change t to x and you have a machine learning tool, GP regression

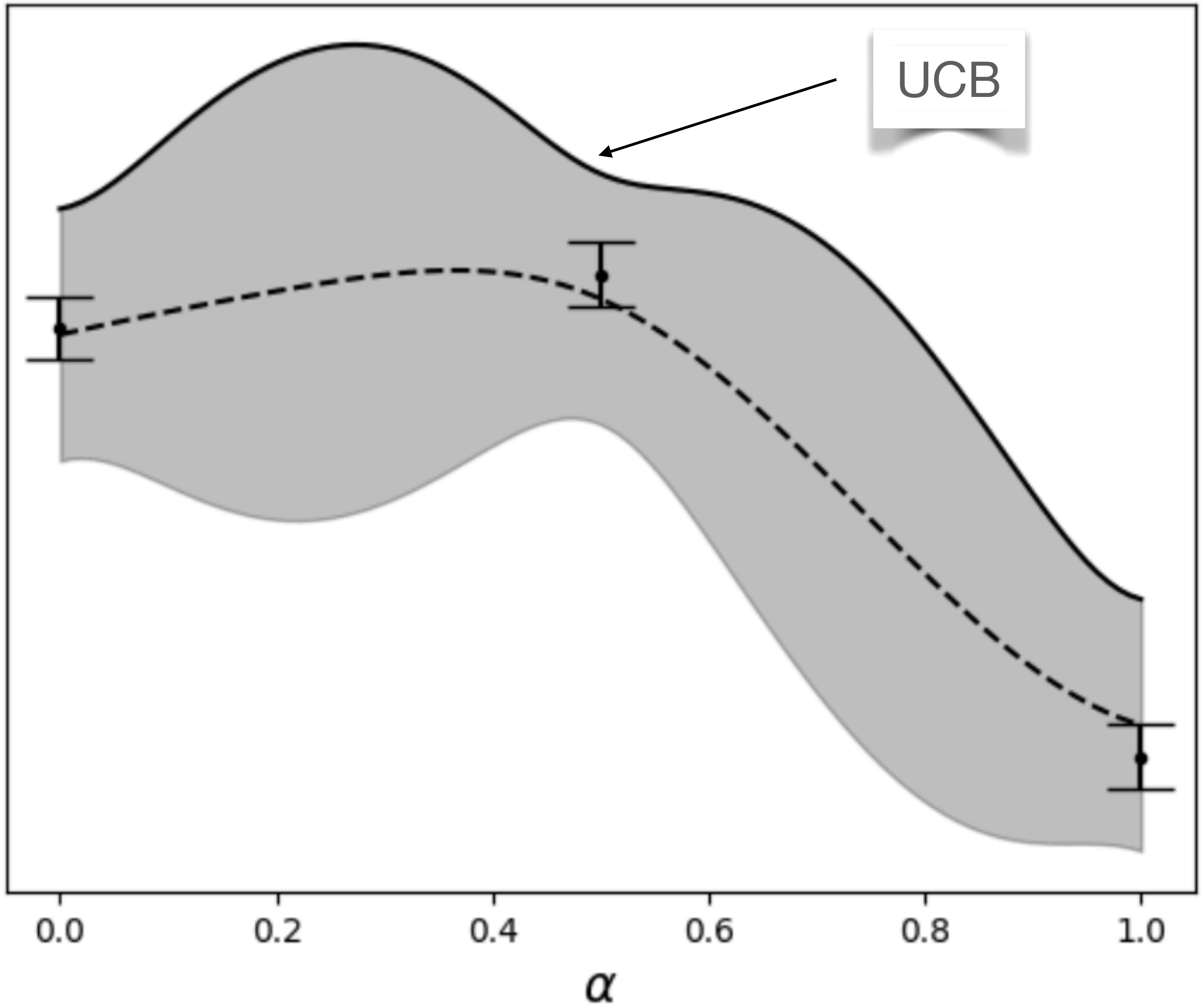
BO: Acquisition Function

- Acquisition function determines experiment design
  - Determines next parameter value to measure
- Ex: Upper confidence bound (UCB)
  - $af_{\text{ucb}} = \mu + \sigma$
  - Dark line



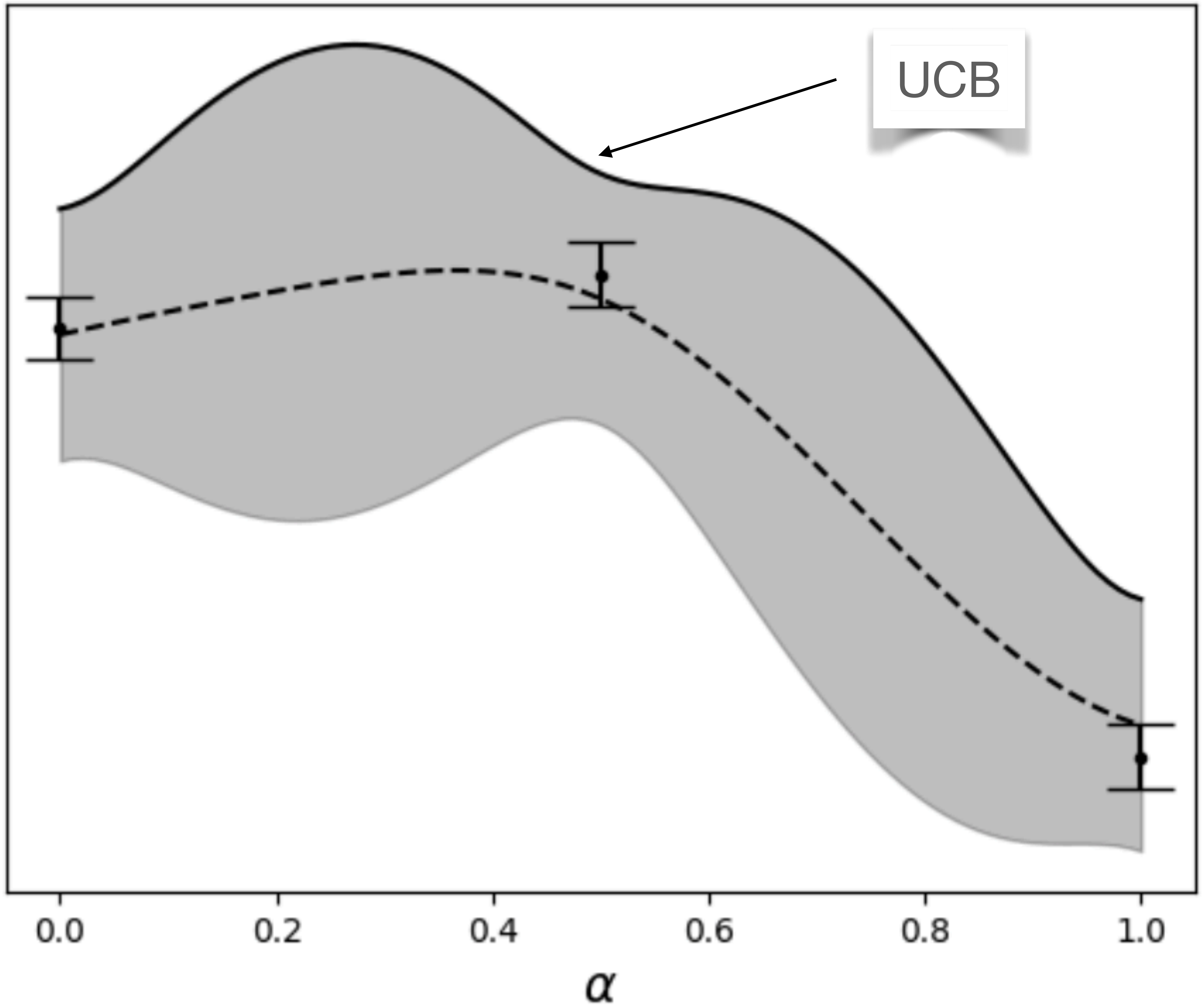
BO: Acquisition Function

- $af_{\text{ucb}} = \mu + \sigma$
- Seeks higher μ, i.e., more BM
  - Measuring here exploits current measurements
  - Yields more $/clicks/etc. in next measurement
- Also, …



BO: Acquisition Function

- $af_{\text{ucb}} = \mu + \sigma$
- Seeks higher σ, more uncertainty
    - Measuring here explores parameter space
    - Improves the *next* surrogate, thus *next* design
- Trading some BM now for more BM later



BO: Acquisition Function

- Many other acquisition functions
  - Expected Improvement (EI, qNEI)
  - Probability of Improvement (PI)
  - Thompson Sampling
  - Entropy Search (ES, PES, MES, GIBBON)
  - TuRBO, EBO, …
- No “best answer”, although qNEI is a good default

https://arxiv.org/pdf/1706.07094.pdf

BO: Acquisition Function

- Optimize A.F. w/numerical optimizer
  - BFGS, req. gradient
    - `scipy.optimize.minimize`
    - `from botorch.optim import optimize_acqf`
  - CMA-ES, no gradient
    - Black Box Optimizer (BBO)
    - `pip install pycma`

Bayesian optimization is also a BBO

BO: Connections

- BO builds on
  - A/B testing: Take a low-se, low-bias measurement
  - MAB: Balance exploration (μ) & exploitation (σ) in design
  - RSM: Build a surrogate and optimize over it

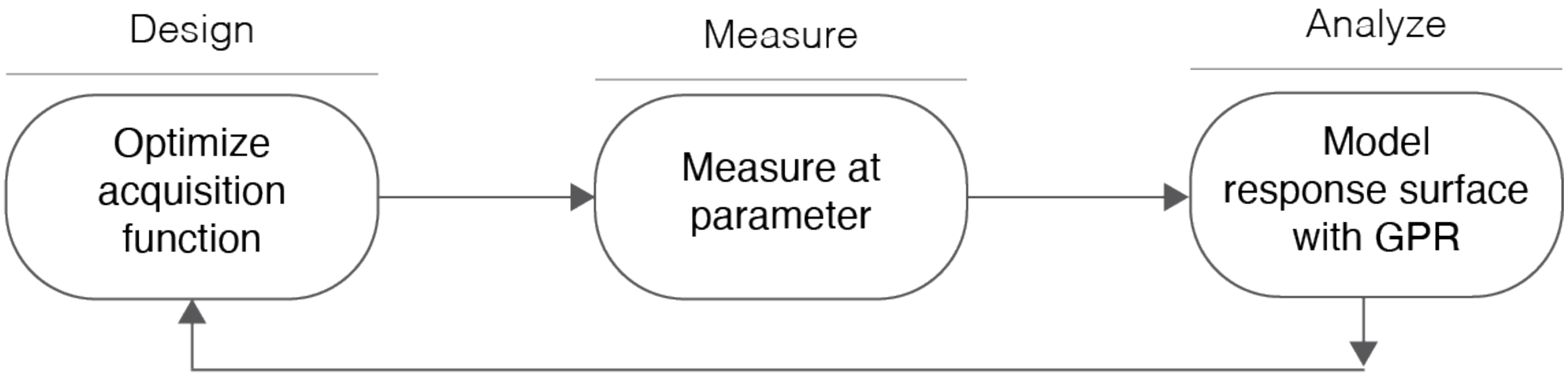
BO: Connections

- Advances technique
- GPR instead of linear regression for surrogate
  - More flexible, more automated
- Acquisition function over continuous parameters
  - MAB’s Thompson Sampling, eps-greedy are acquisition functions over categorical parameters

BO: Connections

		+ Exploration/exploitation
	A/B testing (measurement)	Multi-armed bandits
+ Surrogate function	Response surface modeling	Bayesian optimization

BO Iteration



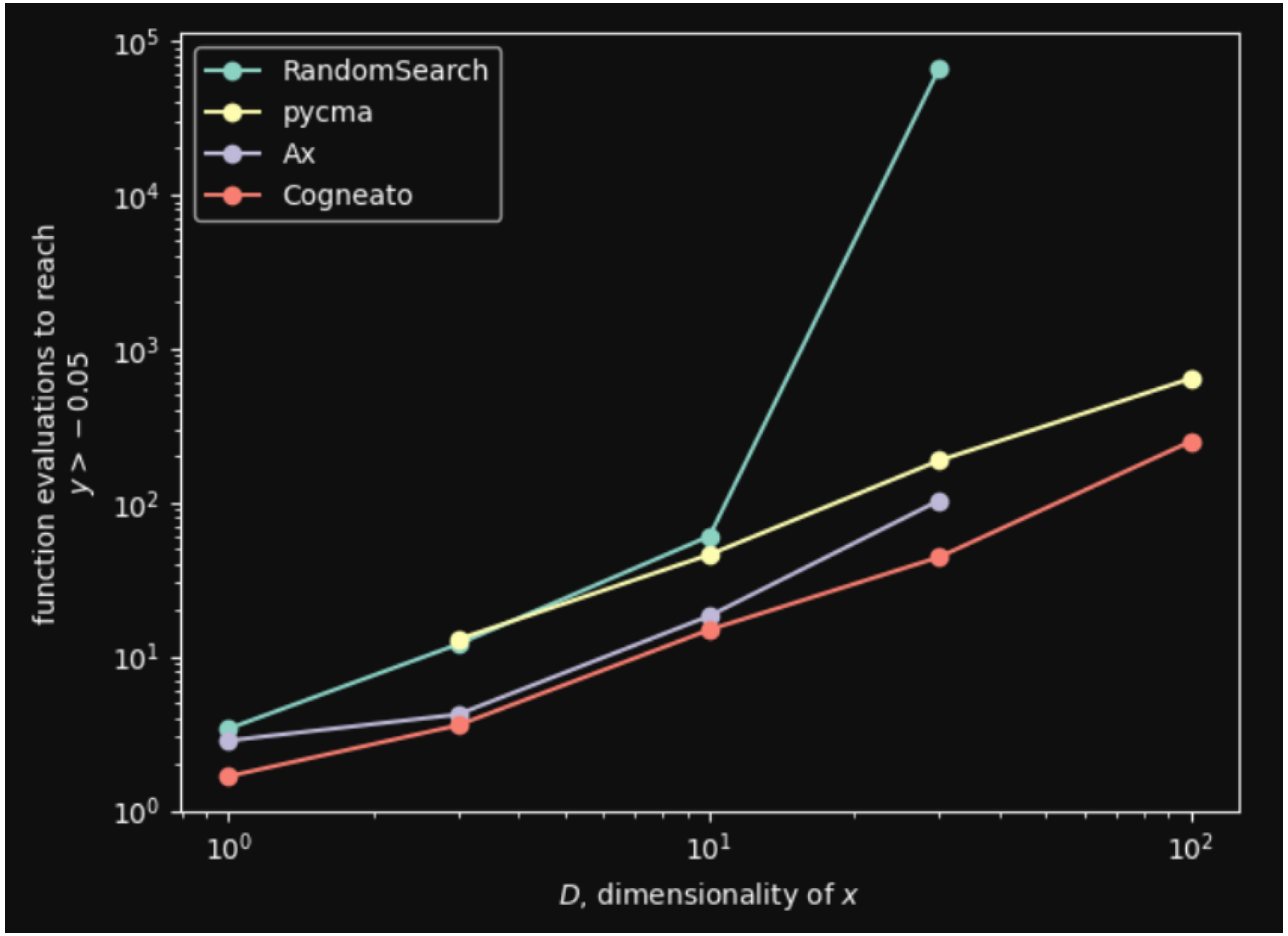
BO: Flexible & Feature-Rich

- **Mixed variable types**: Continuous, ordinal (integer), and categorical (boolean)
- **Multiple metrics**: PnL, risk, volume, order rate, … simultaneously
- **Multiple fidelities**: Combine simulator results w/live results
- **Constraints**: Limit risk, capital, market participation
- **Arbitrary measurements**: Build surrogate from all available measurements
- Research is ongoing into higher dimensions, better initialization, better acquisition functions, surrogates for more complex systems

Cogneato

cogneato.xyz

- GPyTorch, BoTorch, Ax
- Simplified interface, custom algorithm code
- Scales:100D+
- Mixed categorical, ordinal, and continuous parameters
- Multi-arm (batch) designs



Cogneato

cogneato.xyz

intensity:[0,1]	num_objects:{0..3}	scale:[1,3]	version:old,new	color:red,green,blue	position:{1..5}	views:mean	views:se
1	0	1.15	old	red	2	100	30
0.5	3	2.2	old	blue	1	110	35
0.22	1	2.5	new	blue	5	60	22

- Create a table for your measurements in a spreadsheet
- Copy & paste table to Cogneato
- Cogneato returns next experiment design
- Go measure, come back when you’re done (hours/days later)

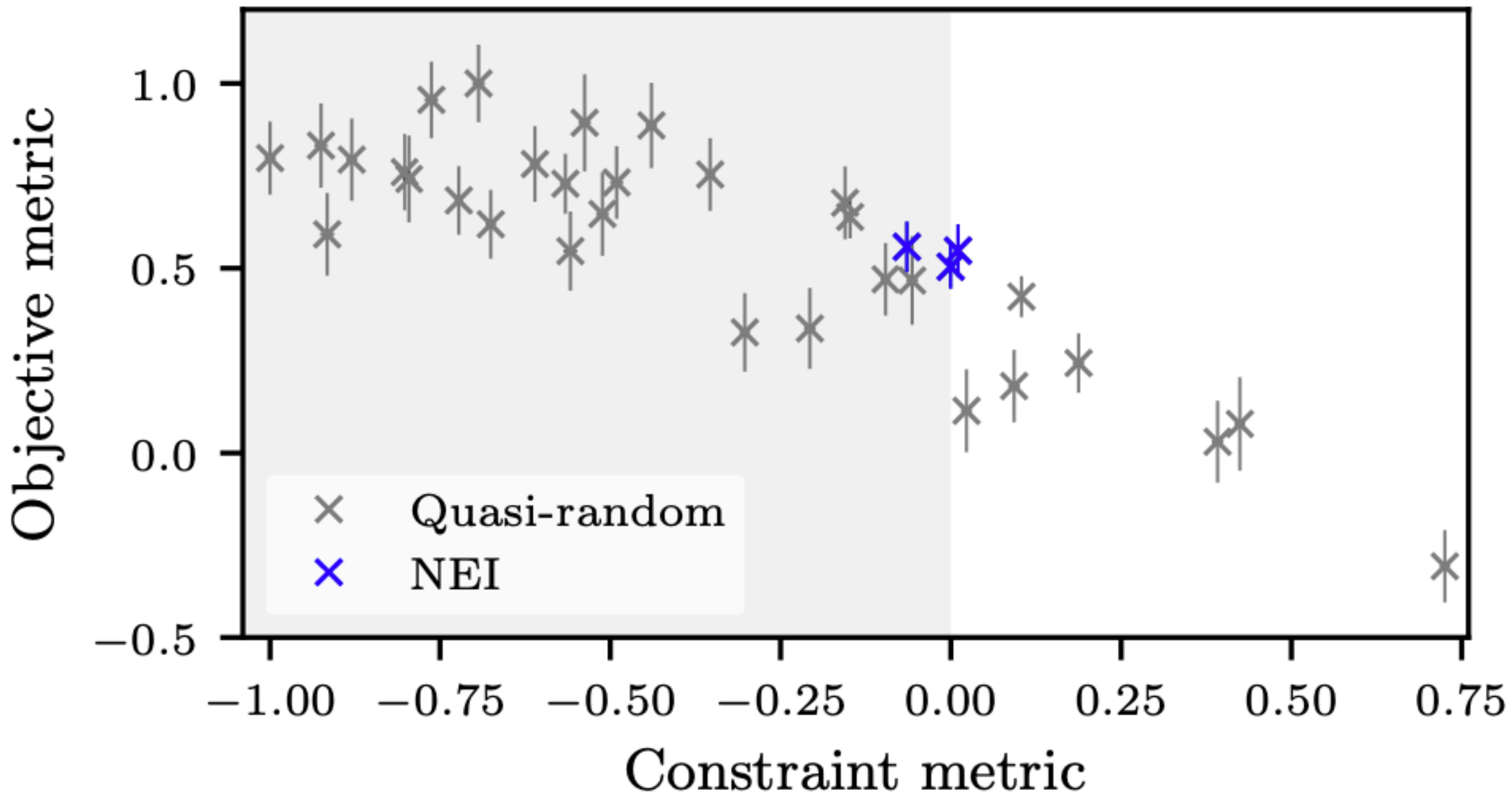
Expected Improvement (qNEI)

http://andamooka.org/~dsweet/EOCourse/EI.mov

DoE-Inspired Acquisition Function
J. Ren & D Sweet
http://andamooka.org/~dsweet/EOCourse/ITS.mov

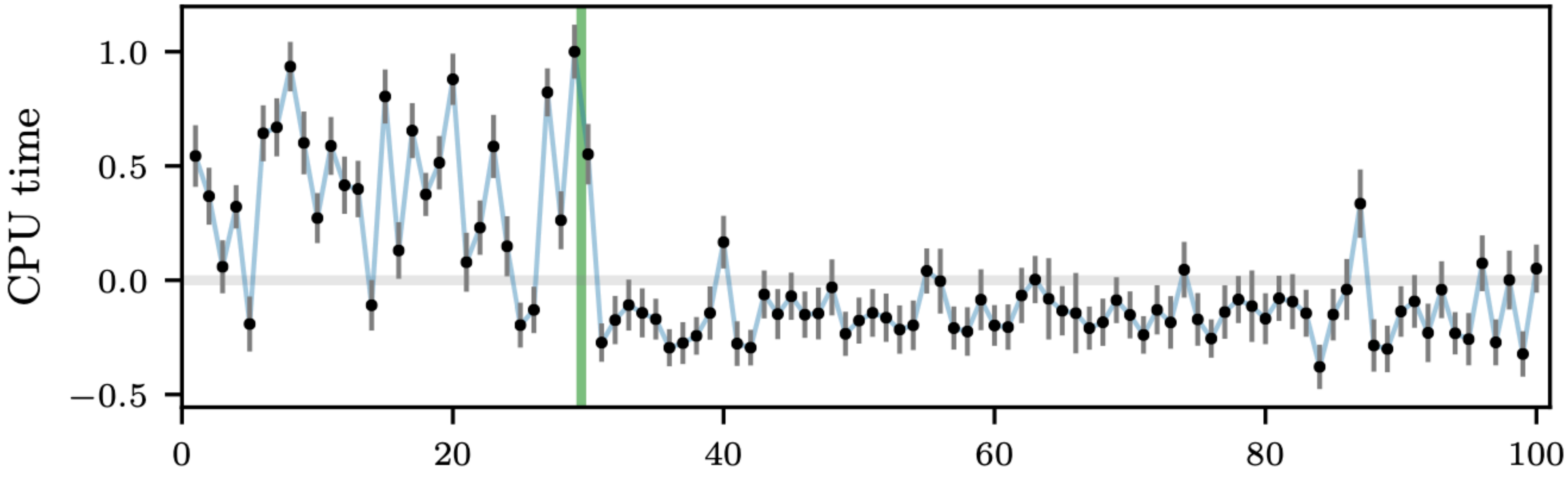
Case: Ranking System

- **Constrained Bayesian Optimization with Noisy Experiments**
  https://arxiv.org/pdf/1706.07094.pdf
- Production/Meta
- Proprietary metric
- 6 parameters
- 31 arms in first pass
- 3 arms in second pass



Case: HipHop Virtual Machine

- **Constrained Bayesian Optimization with Noisy Experiments**
  https://arxiv.org/pdf/1706.07094.pdf
- Offline runs of PHP/Hack software
- CPU time
- 7 parameters
- 1 arm/batch
- 100 batches



Case: GPU Kernels

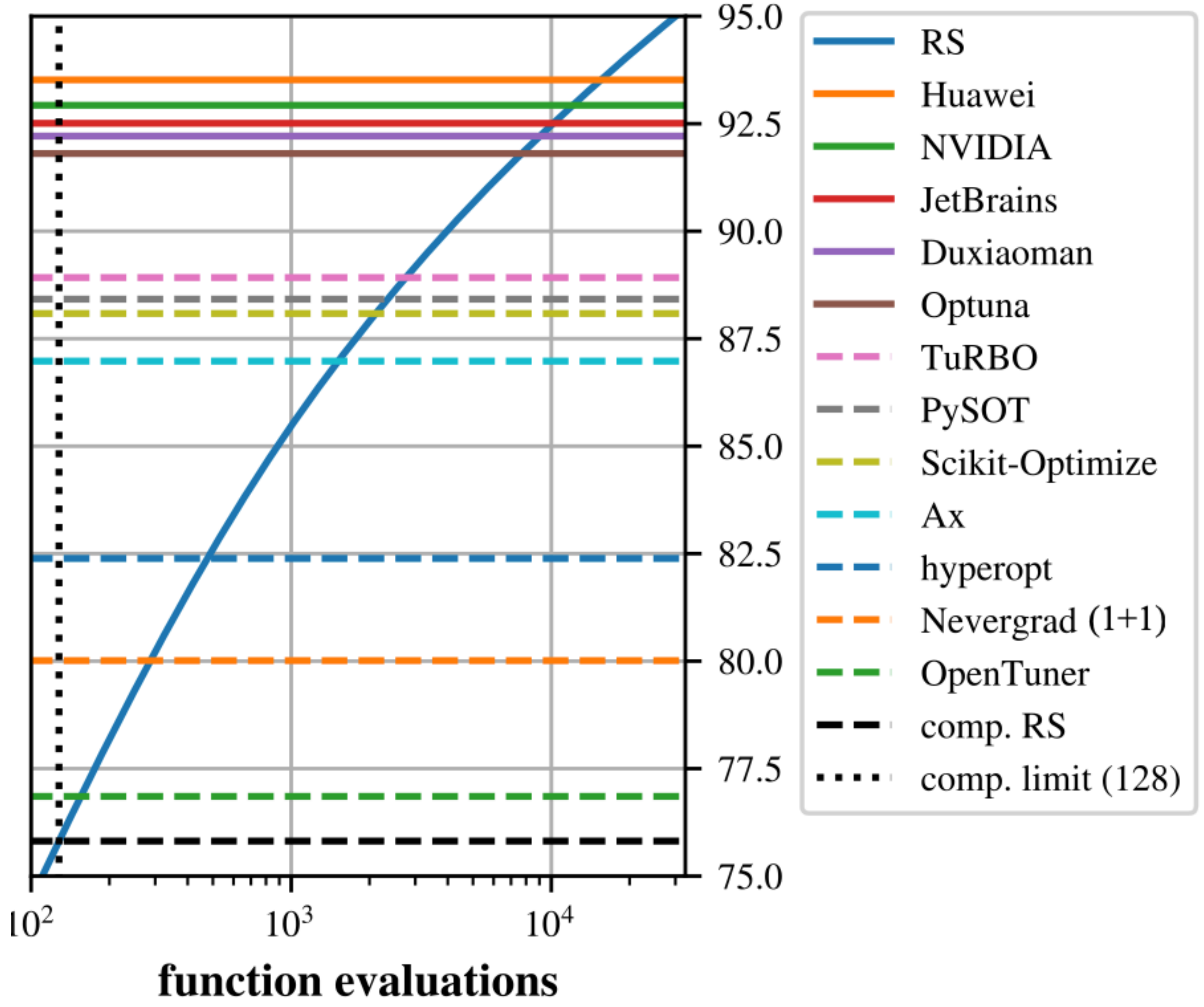
- **Bayesian Optimization for auto-tuning GPU kernels**
  https://arxiv.org/pdf/2111.14991.pdf

Kernel	Configurations	Invalid	Minimum	Tunable parameters
GEMM	17956	0 (0%)	28.307	M_{wg}, N_{wg}, K_{wg}, M_{dimC}, N_{dimC}, M_{dimA}, N_{dimB}, K_{WI}, V_{WM}, V_{WN}, S_{TRM}, S_{TRN}, S_A, S_B, PRECISION
Convolution	9400	3624 (38.5%)	1.625	filter_width, filter_height, block_size_x, block_size_y, tile_size_x, tile_size_y, use_padding, read_only
PnPoly	8184	323 (3.9%)	26.968	block_size_x, tile_size, between_method, use_precomputed_slopes, use_method

TABLE II: Specifications of tunable kernels for the GTX Titan X. Minimum execution time is given in milliseconds.

Case: Hyperparameter Optimization

- **Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of …**
  https://arxiv.org/pdf/2104.10201.pdf
- Fitting / training of supervised learning models
  - GBDT, logistic regression, MLP
- Out-of-sample loss
- 8 arms/batch
- 16 batches



Reading

- **An Intuitive Tutorial to Gaussian Processes Regression**
  https://arxiv.org/pdf/2009.10862.pdf
- **10 Things to Know About Covariate Adjustment**
  https://egap.org/resource/10-things-to-know-about-covariate-adjustment/
- Chapter 5 from **Experimentation for Engineers**

Summary

- **Initialization**: Space-filling sequence, Sobol
- **Surrogate**: GPR
  - Non-parametric
  - Models mean and uncertainty
- **Acquisition function**: Determines next experiment design
  - Balances exploration (μ) with exploitation (σ)