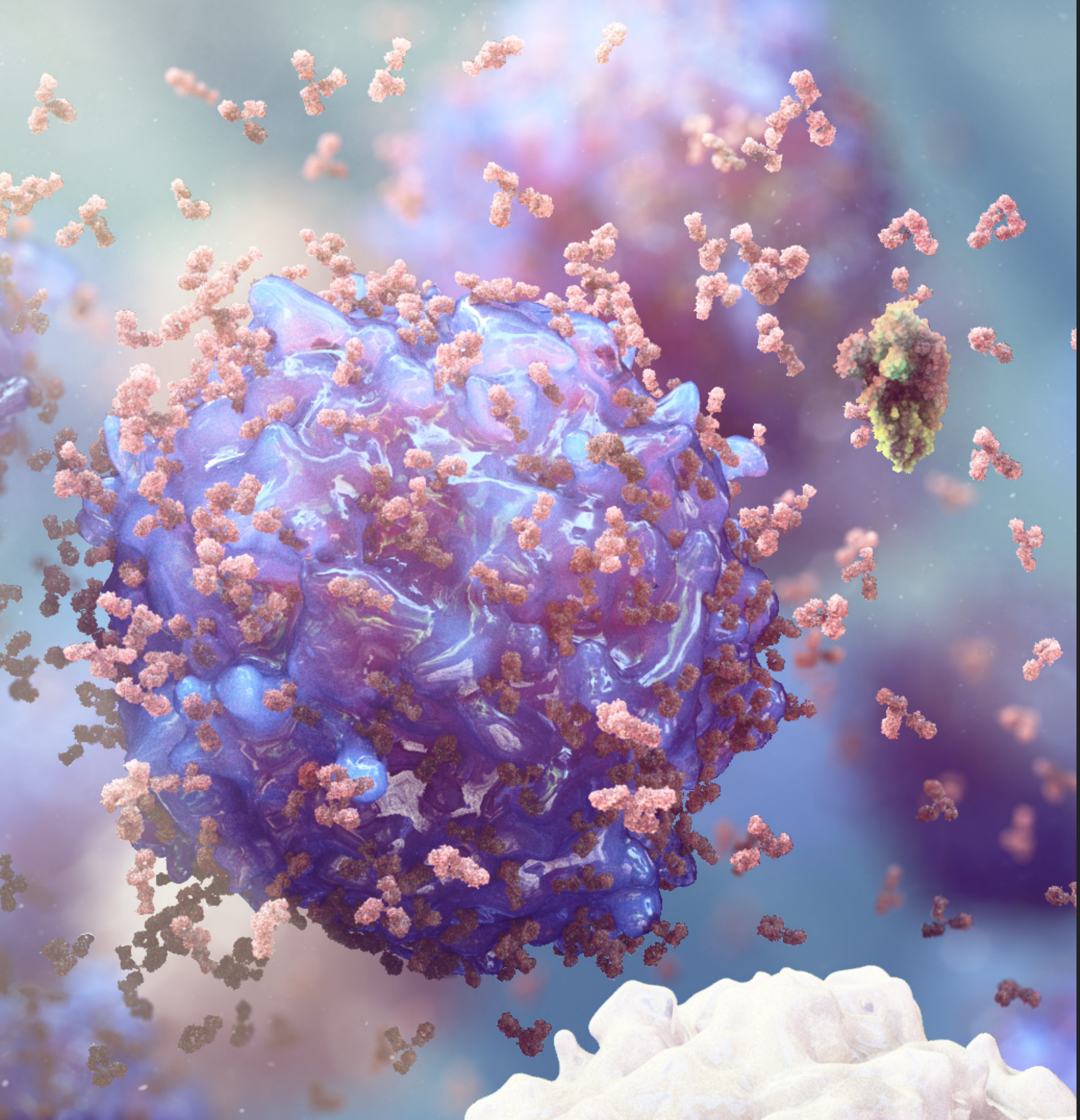
Robust Multi-Objective Bayesian Optimization Under Input Noise

Sam Daulton[*1,2], Sait Cakmak[*,1,3], Max Balandat[1],
Mike Osborne[2], Enlu Zhou[3], Eytan Bakshy[1]

ICML 2022

*Equal Contribution
[1] Meta
[2] University of Oxford
[3] Georgia Institute of Technology



AstraZeneca

UNIVERSITY OF
OXFORD

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)
  - Shelf temperature

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)
  - Shelf temperature
  - Chamber pressure

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)
  - Shelf temperature
  - Chamber pressure
- Objectives:

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)
  - Shelf temperature
  - Chamber pressure
- Objectives:
  - Efficiency of drying step (maximize)

Optimizing Vaccine Freeze Drying Process

- Goal: Tune experimental conditions (parameters)
  - Shelf temperature
  - Chamber pressure
- Objectives:
  - Efficiency of drying step (maximize)
  - Product quality (maximize)

Multi-Objective Bayesian Optimization

$$\max_{x \in \mathcal{X}} f(x)$$

Goal: optimize a **vector-valued** black-box function that:

- Is expensive to evaluate ($, time)
- Does not provide gradients

$$f(x) = (f^{(1)}(x), \dots, f^{(M)}(x)) \subset \mathbb{R}^M$$

Multi-Objective Bayesian Optimization

$x \rightarrow$ [?] $\rightarrow f(x)$

$\max_{x \in \mathcal{X}} f(x)$

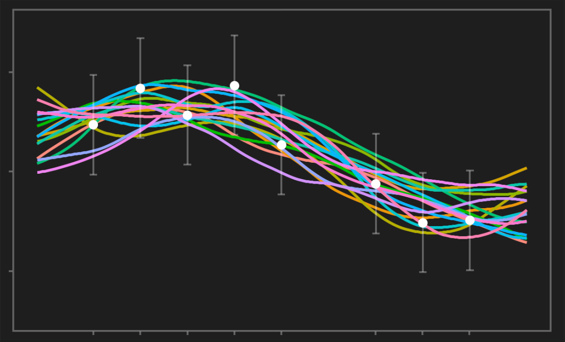
Goal: optimize a **vector-valued** black-box function that:

- Is expensive to evaluate ($, time)
- Does not provide gradients

$$f(x) = (f^{(1)}(x), \ldots, f^{(M)}(x)) \subset \mathbb{R}^M$$

$\mathcal{D} = \{x_i, f(x_i)\}$

$x \rightarrow$ [Surrogate Model] $\rightarrow \hat{f}(x) \rightarrow$ [Acquisition Function] $\rightarrow \alpha(x)$



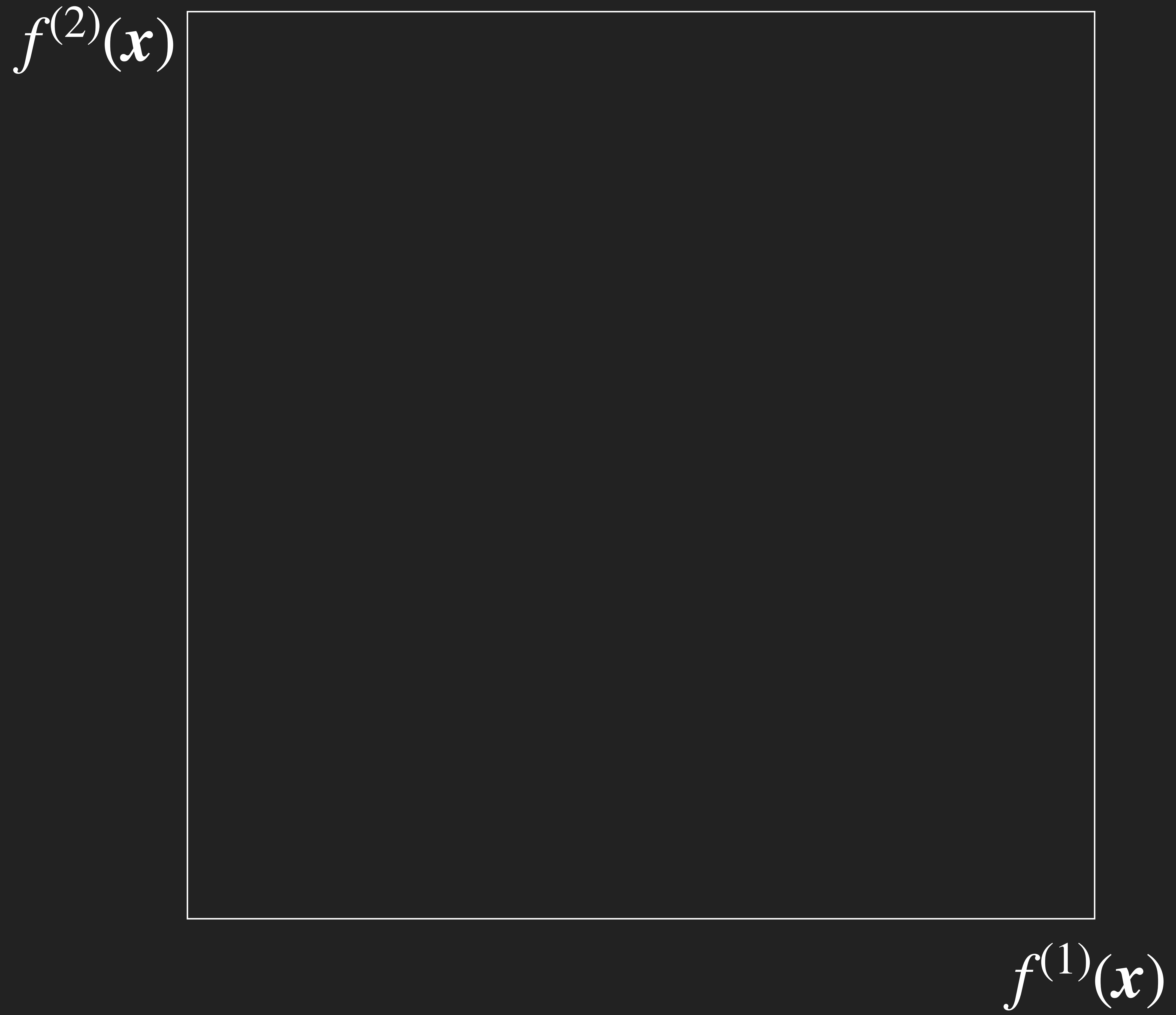
- Use a surrogate model that is fast to evaluate and provides gradients
- Use acquisition functions to perform explore/exploit

[Numerical Optimization] $\rightarrow x^*$

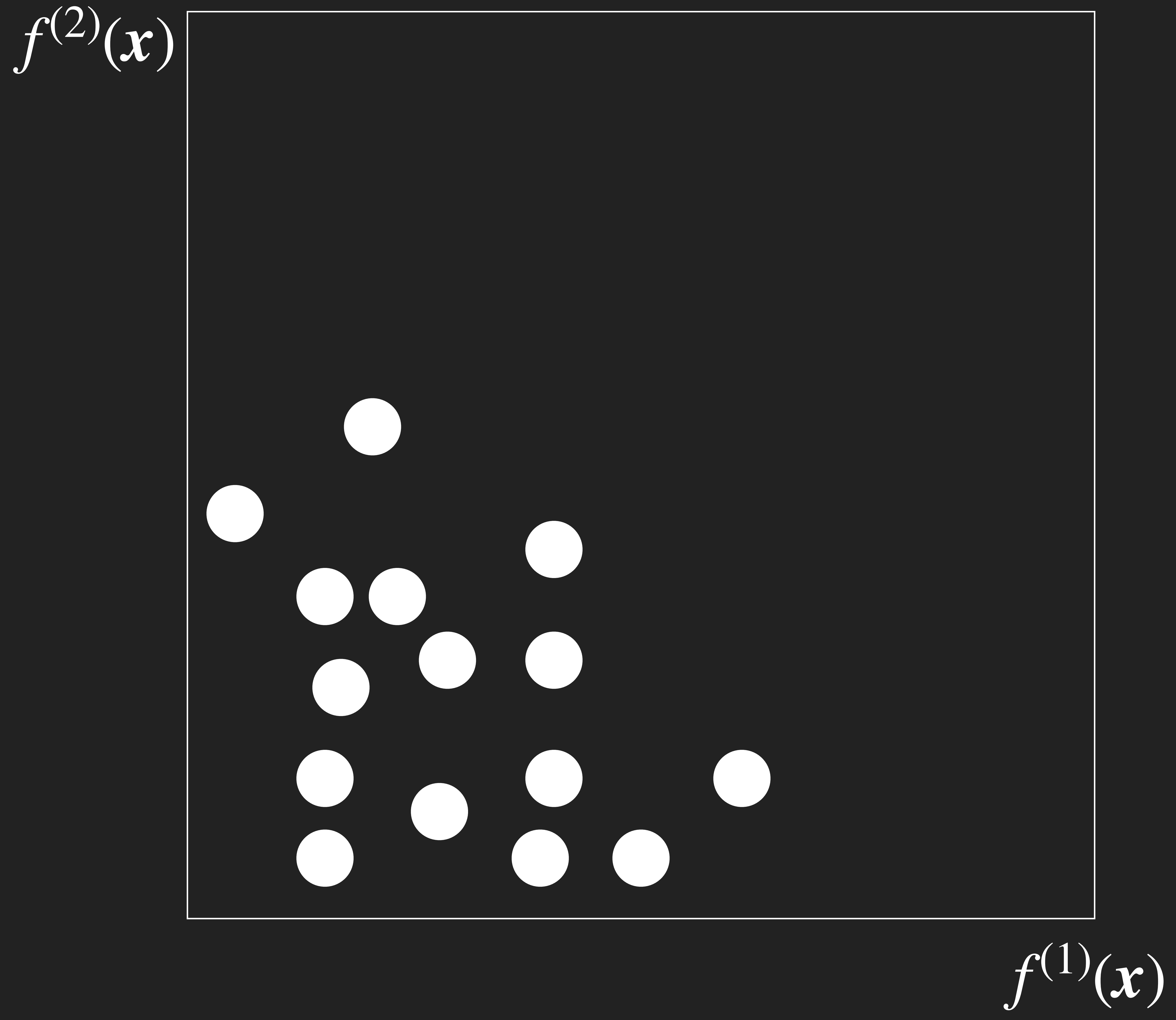
$\max_{x \in \mathcal{X}} \alpha(x)$

- Generate candidate points to evaluate next

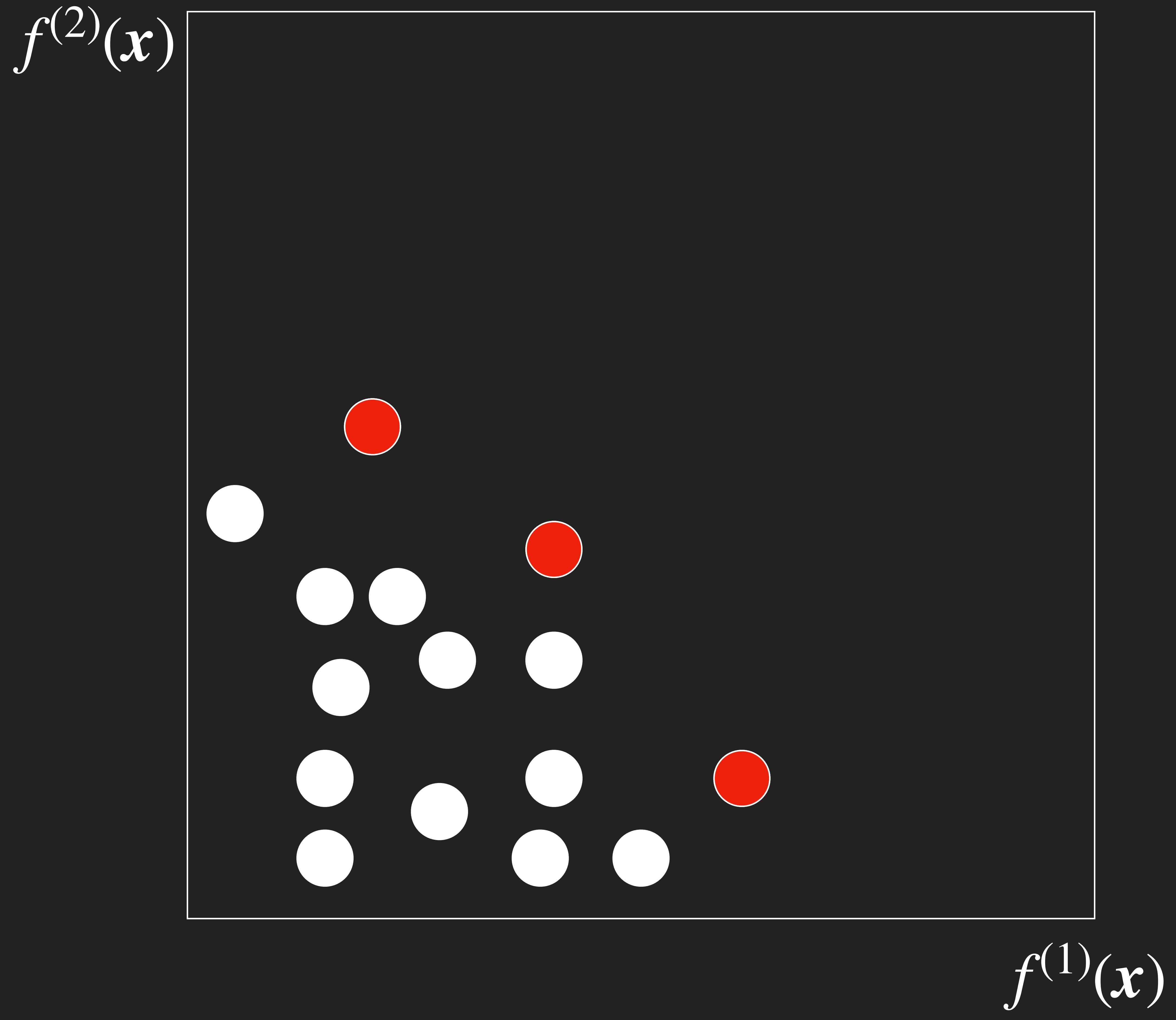
Pareto Frontier



Pareto Frontier



Pareto Frontier



Many Manufacturing Processes are Subject to Input Noise

- The realized parameters are different than intended

Many Manufacturing Processes are Subject to Input Noise

- The realized parameters are different than intended
- The performance metrics might degrade

Many Manufacturing Processes are Subject to Input Noise

- The realized parameters are different than intended
- The performance metrics might degrade
  - Sometimes the effects are catastrophic

Many Manufacturing Processes are Subject to Input Noise

- The realized parameters are different than intended
- The performance metrics might degrade
  - Sometimes the effects are catastrophic
- Example: In vaccine freeze drying,

Many Manufacturing Processes are Subject to Input Noise

- The realized parameters are different than intended
- The performance metrics might degrade
  - Sometimes the effects are catastrophic
- Example: In vaccine freeze drying,
  - Higher temperatures are more efficient, but too high of a temperature can ruin the product

Problem Formulation

- We consider the setting where:

Problem Formulation

- We consider the setting where:
    - We can sample from the input noise process $P(\boldsymbol{\xi}; \boldsymbol{x})$

Problem Formulation

- We consider the setting where:
  - We can sample from the input noise process $P(\boldsymbol{\xi}; \boldsymbol{x})$
  - We have access to a simulator during optimization, without input noise

Problem Formulation

- We consider the setting where:
  - We can sample from the input noise process $P(\boldsymbol{\xi}; \boldsymbol{x})$
  - We have access to a simulator during optimization, without input noise
  - Input noise is present at implementation time

Problem Formulation

- We consider the setting where:
  - We can sample from the input noise process $P(\boldsymbol{\xi}; \boldsymbol{x})$
  - We have access to a simulator during optimization, without input noise
  - Input noise is present at implementation time
  - The way in which input noise affects the input parameters $\boldsymbol{x} \diamond \boldsymbol{\xi}$ is known (e.g. additive, multiplicative, etc).

Quantifying Risk

Quantifying Risk

Example risk measures

Quantifying Risk

Example risk measures

- Expected Bayes Risk: $\mathbb{E}_{\boldsymbol{\xi} \sim P(\boldsymbol{\xi})}[f(\boldsymbol{x} \diamond \boldsymbol{\xi})]$

Quantifying Risk

Example risk measures

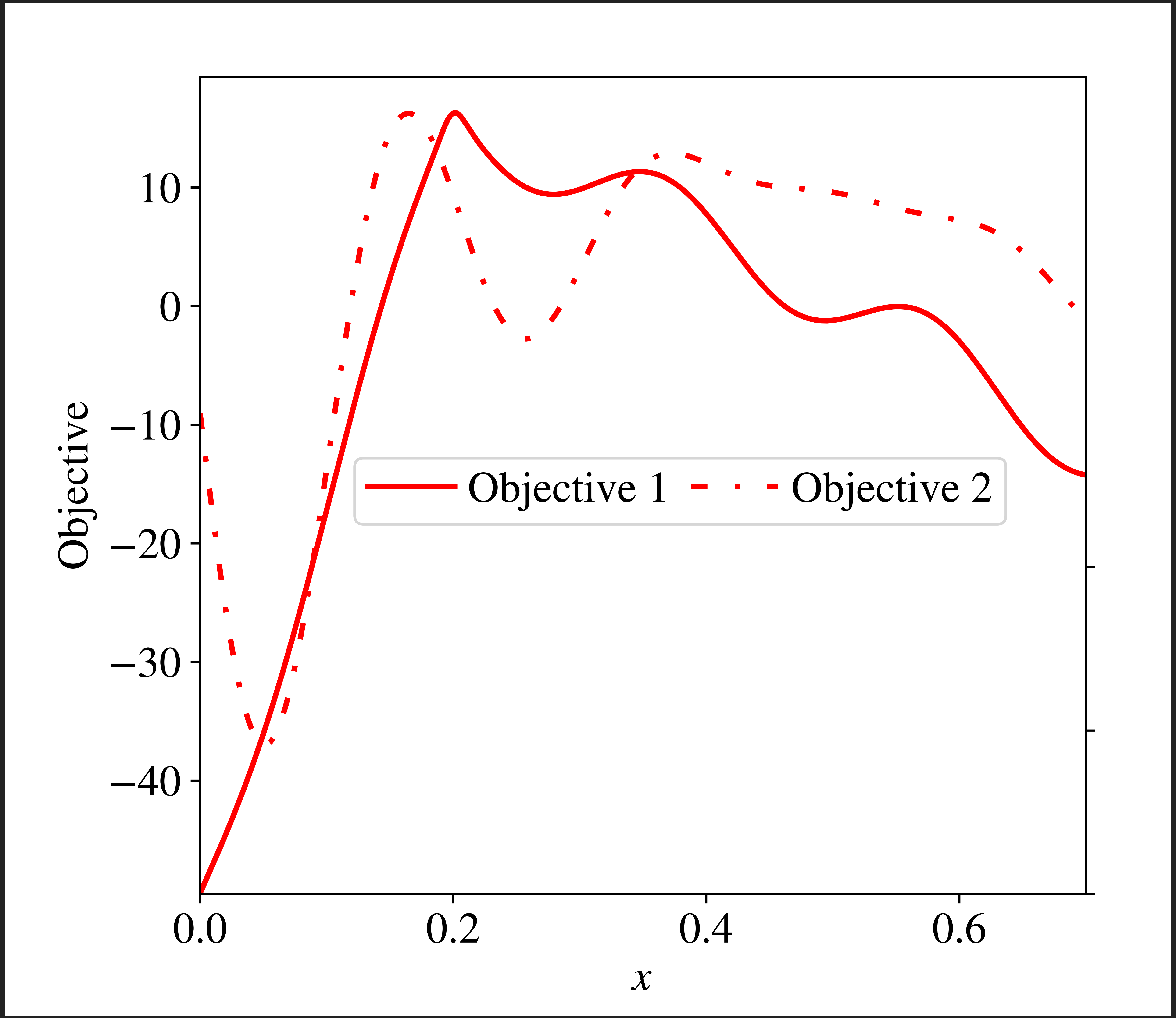
- Expected Bayes Risk: $\mathbb{E}_{\boldsymbol{\xi} \sim P(\boldsymbol{\xi})}[f(\boldsymbol{x} \diamond \boldsymbol{\xi})]$
- Worst case: $\min_{\boldsymbol{\xi} \sim P(\boldsymbol{\xi})} [f(\boldsymbol{x} \diamond \boldsymbol{\xi})]$

Quantifying Risk

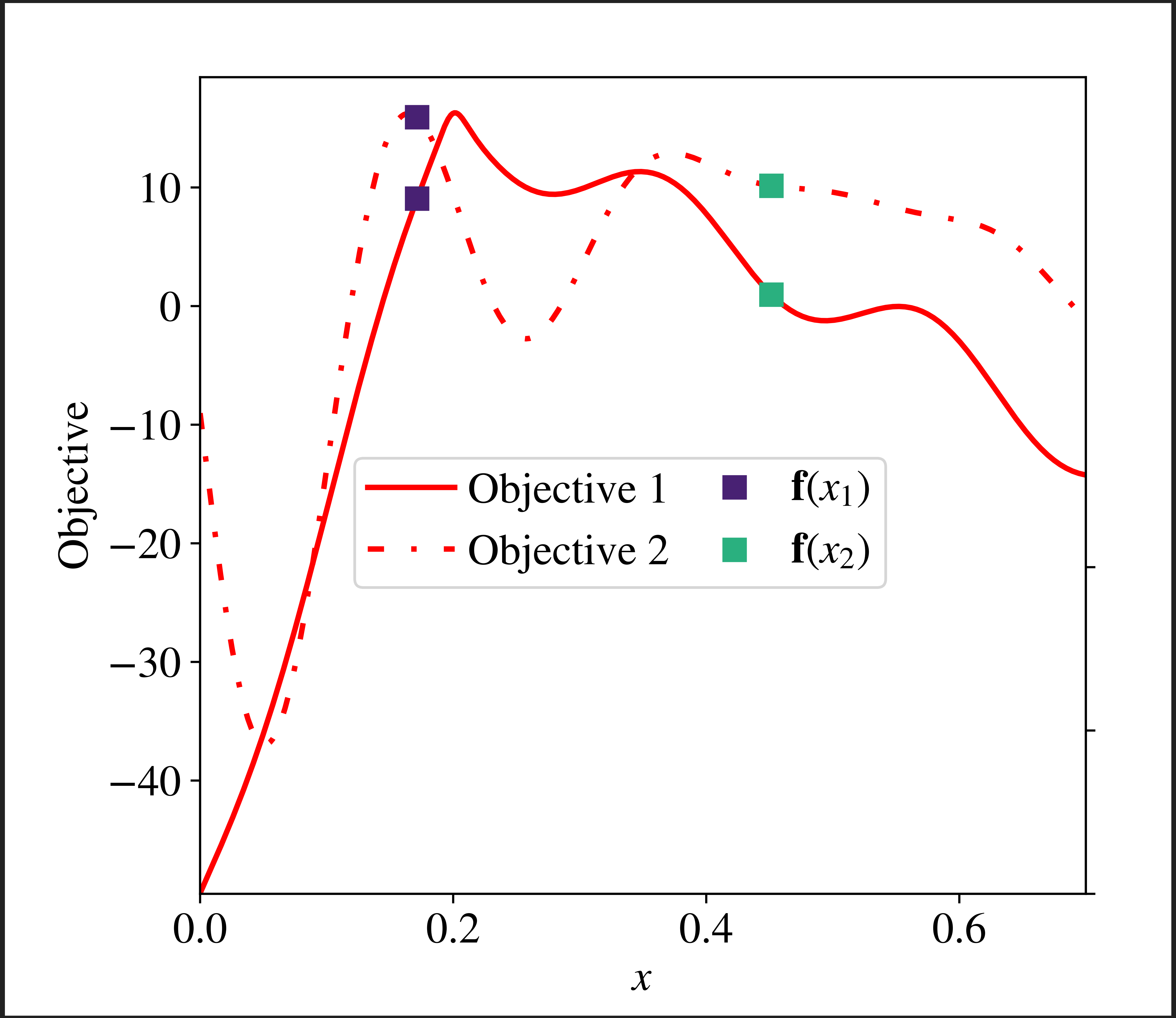
Example risk measures

- Expected Bayes Risk: $\mathbb{E}_{\boldsymbol{\xi} \sim P(\boldsymbol{\xi})}[f(\boldsymbol{x} \diamond \boldsymbol{\xi})]$
- Worst case: $\min_{\boldsymbol{\xi} \sim P(\boldsymbol{\xi})} [f(\boldsymbol{x} \diamond \boldsymbol{\xi})]$
- Value-At-Risk: lower bound on $f(\boldsymbol{x} \diamond \boldsymbol{\xi})$ with probability α

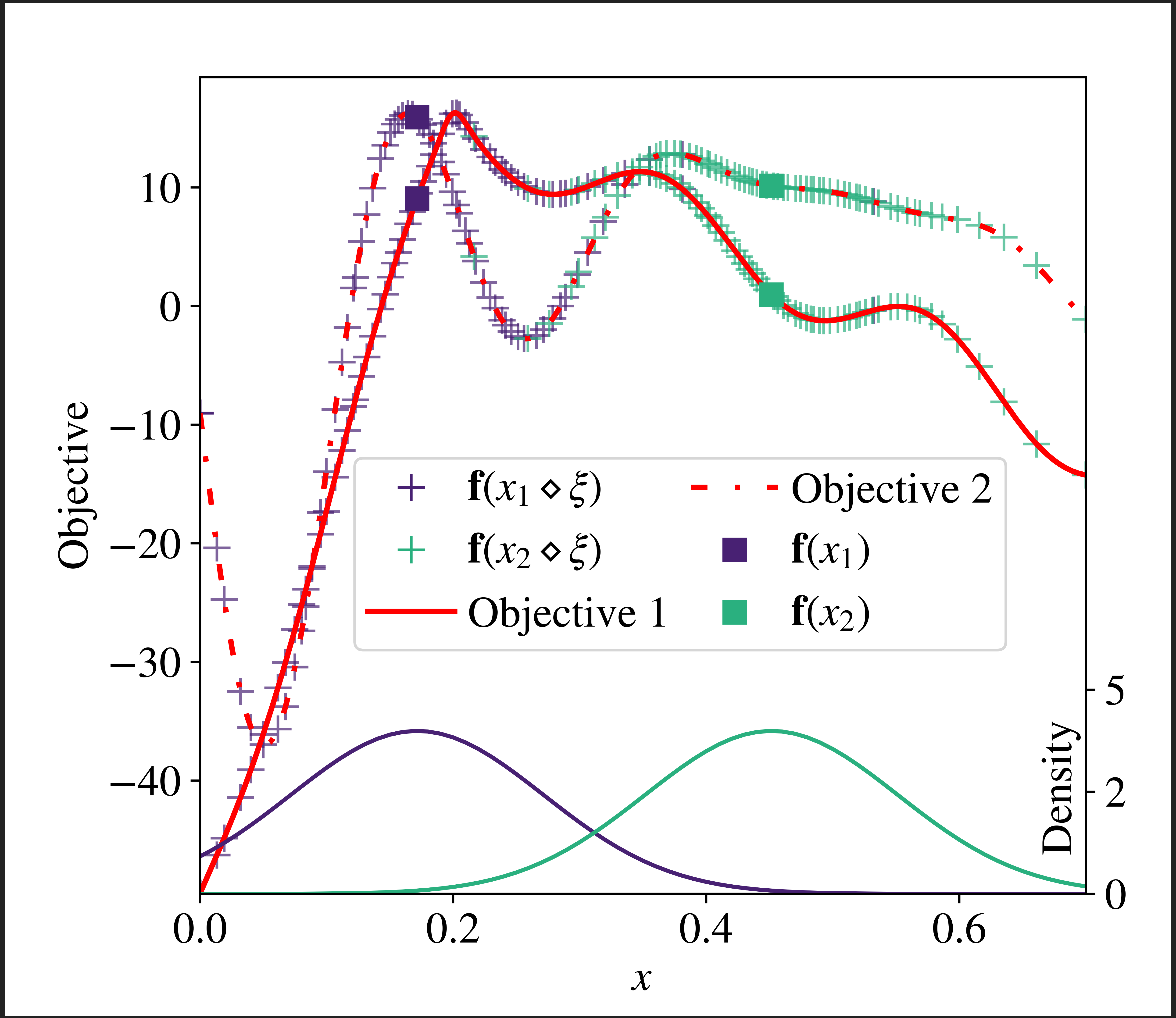
Toy Illustration



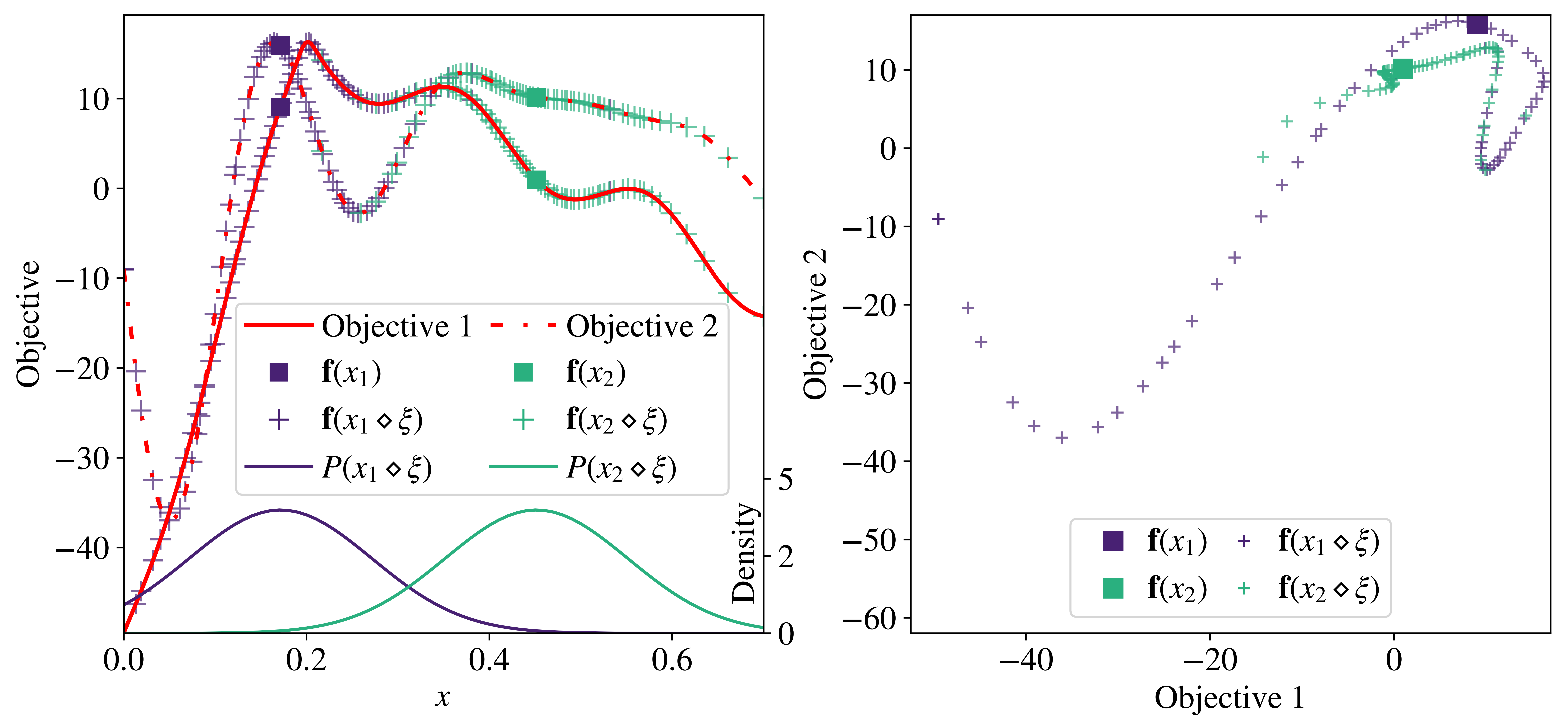
Toy Illustration



Toy Illustration



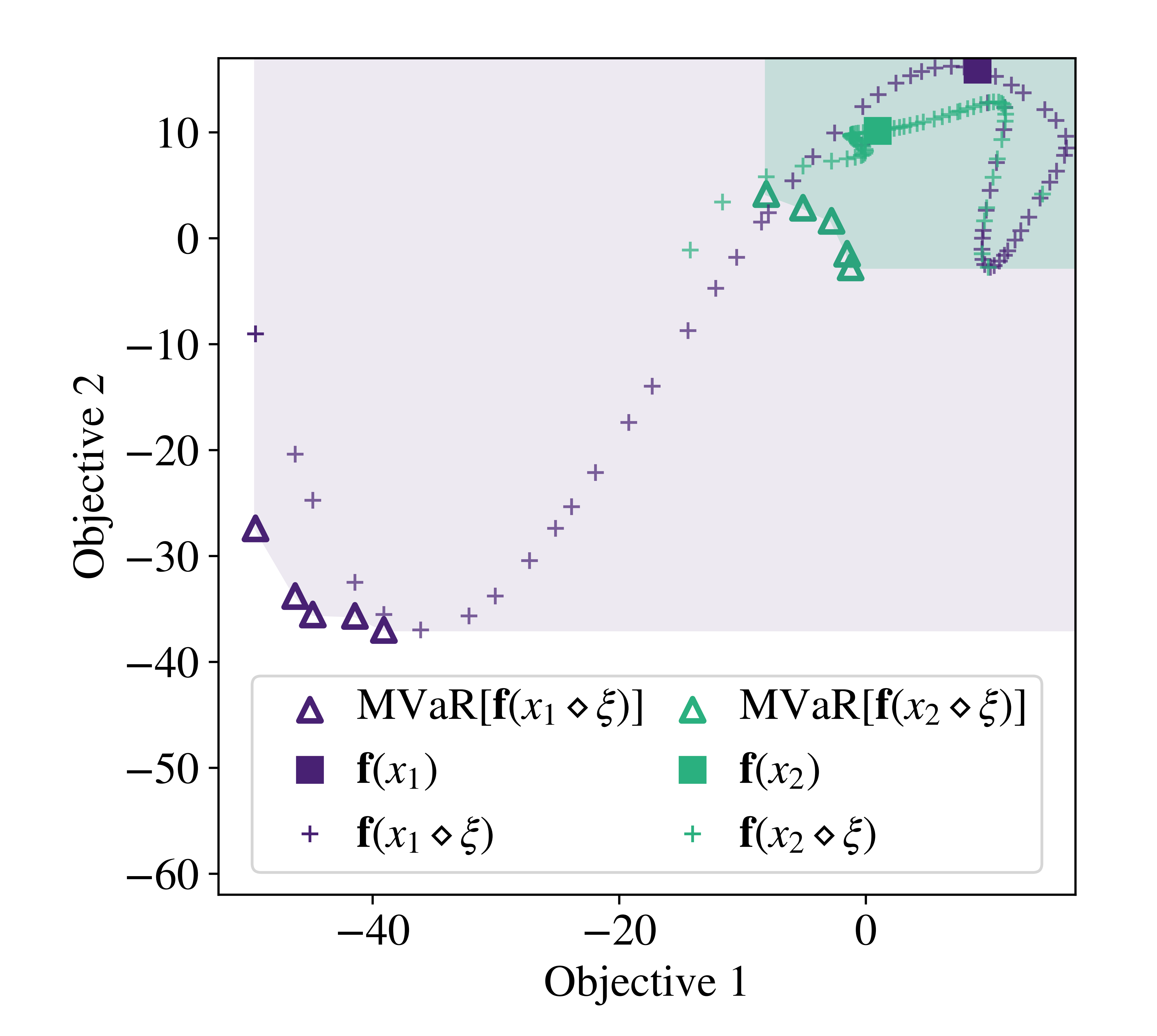
Toy Illustration



Multivariate Value-At Risk (MVaR)

Definition 4.2. The MVAR of $\boldsymbol{f}$ for a given point $\boldsymbol{x}$ and confidence level $\alpha \in [0, 1]$ is:

$$\text{MVAR}_\alpha\left[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi})\right] = \text{PARETO}\left(\left\{\boldsymbol{z} \in \mathbb{R}^M : P\left[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi}) \geq \boldsymbol{z}\right] \geq \alpha\right\}\right).$$

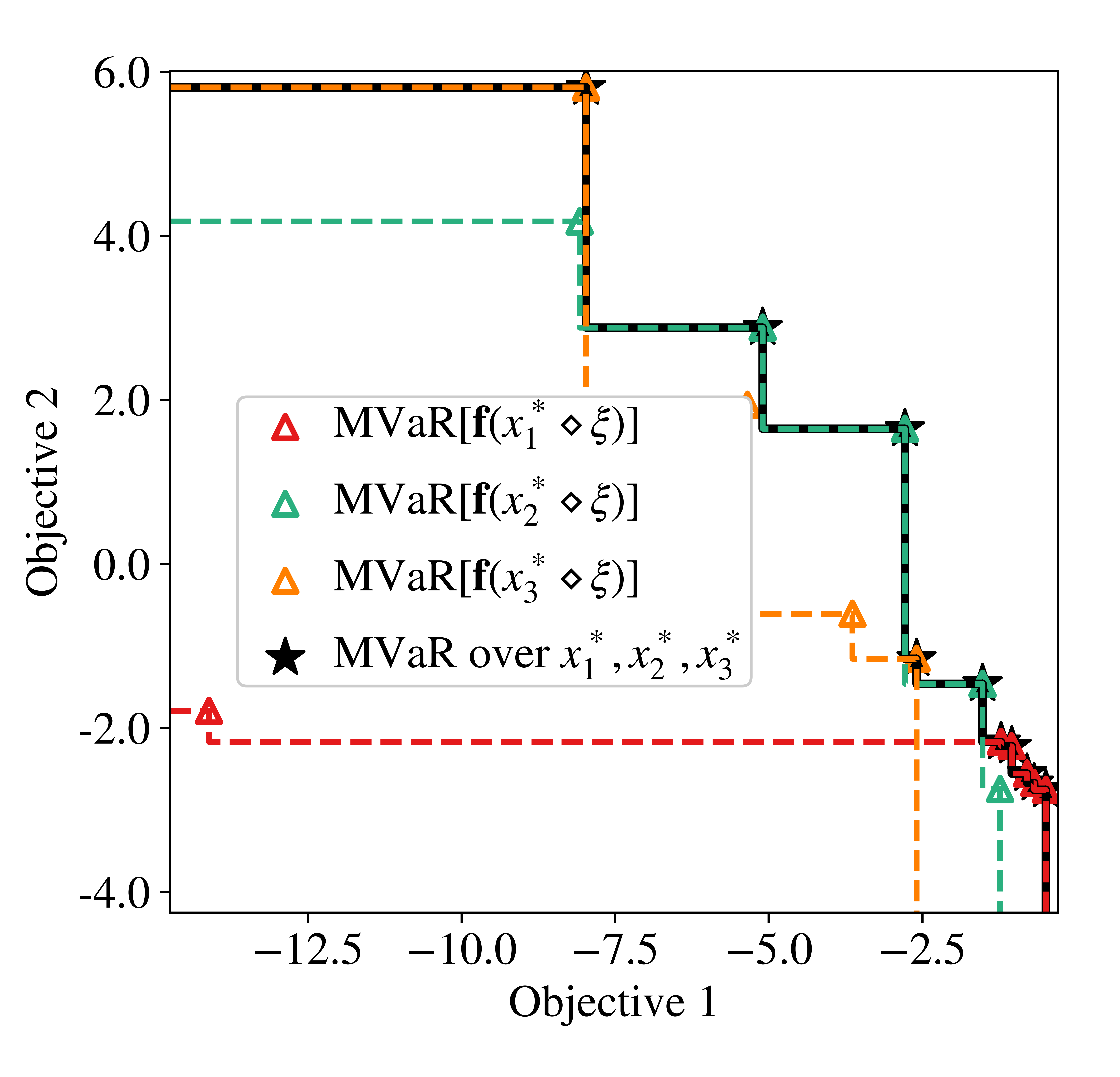


Global MVaR

Definition 4.3. The MVaR for a set of points X is:

$$\mathrm{MVaR}_{\alpha}\left[\{\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi})\}_{\boldsymbol{x} \in X}\right] = \mathrm{Pareto}\left(\bigcup_{\boldsymbol{x} \in X} \mathrm{MVaR}_{\alpha}\left[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi})\right]\right).$$

Our optimization goal is to optimize global MVaR



Issues with Direct MVaR Optimization

Issues with Direct MVaR Optimization

- MVaR is expensive to compute

Issues with Direct MVaR Optimization

- MVaR is expensive to compute
    - Requires approximating multivariate CDFs

Issues with Direct MVaR Optimization

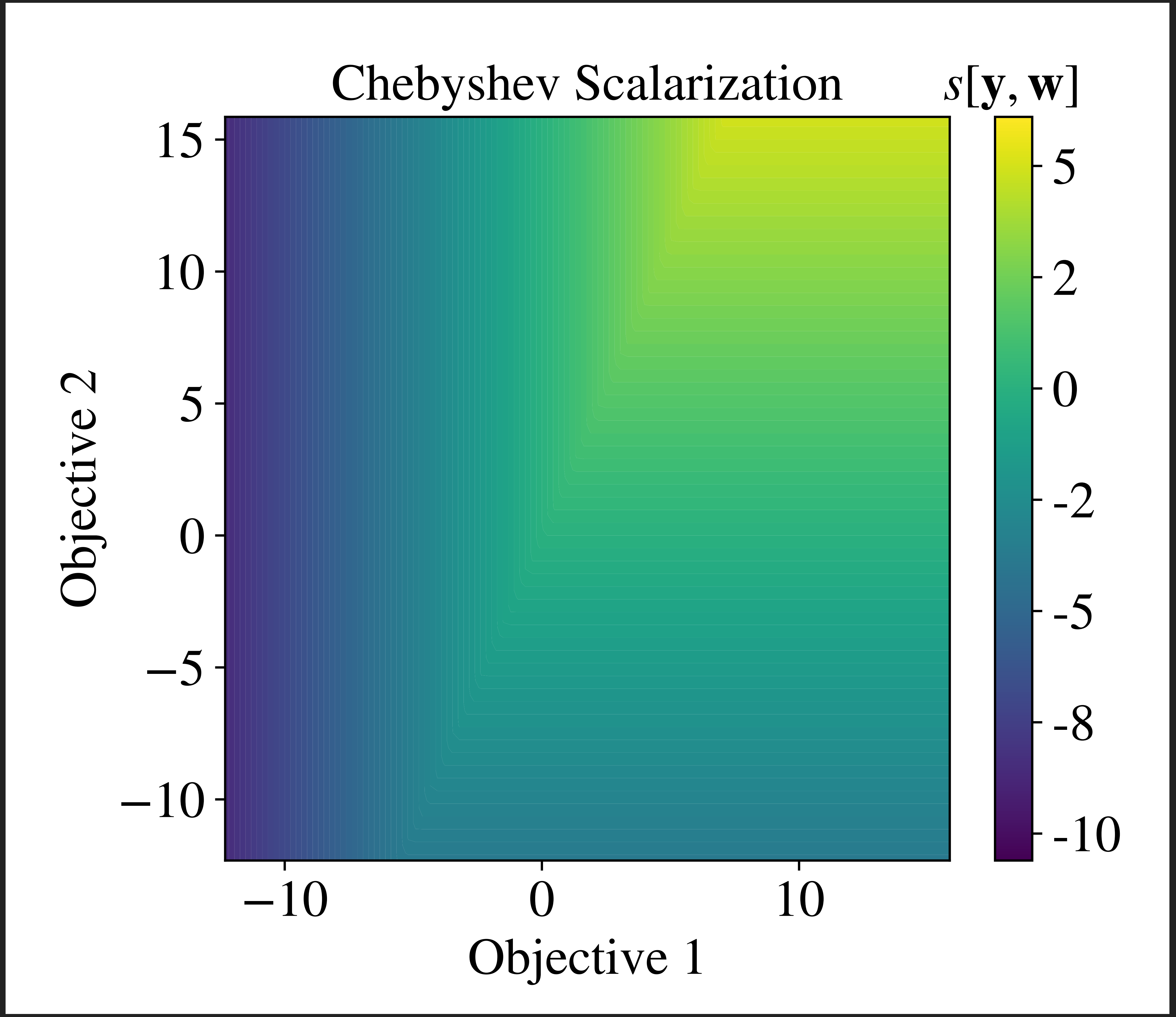
- MVaR is expensive to compute
  - Requires approximating multivariate CDFs
  - Exponential in the number of objectives

Relationship Between MVaR and Scalarizations

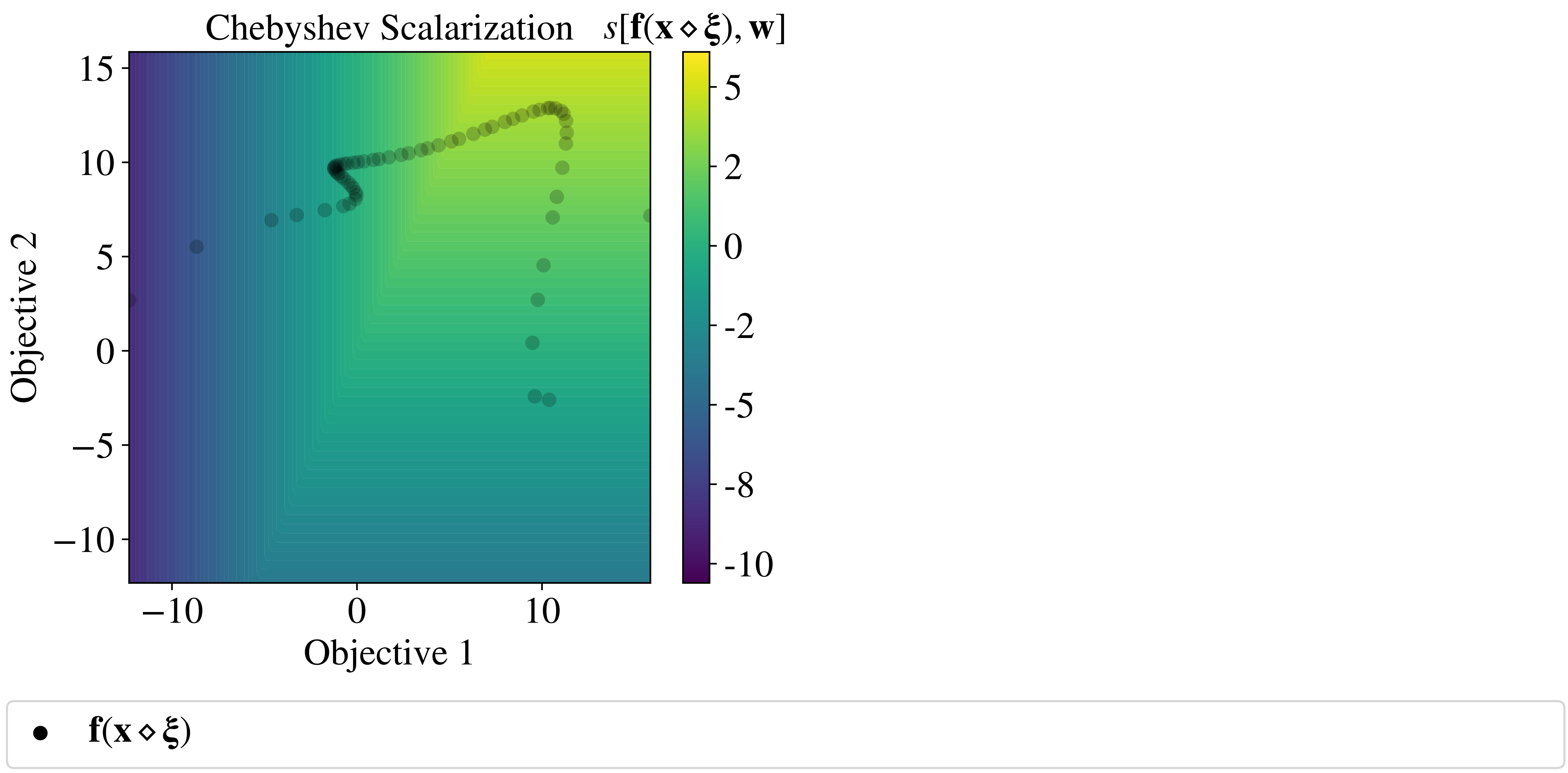
Chebyshev Scalarization:

$$s[\boldsymbol{y}, \boldsymbol{w}] = \min_i w_i y_i$$

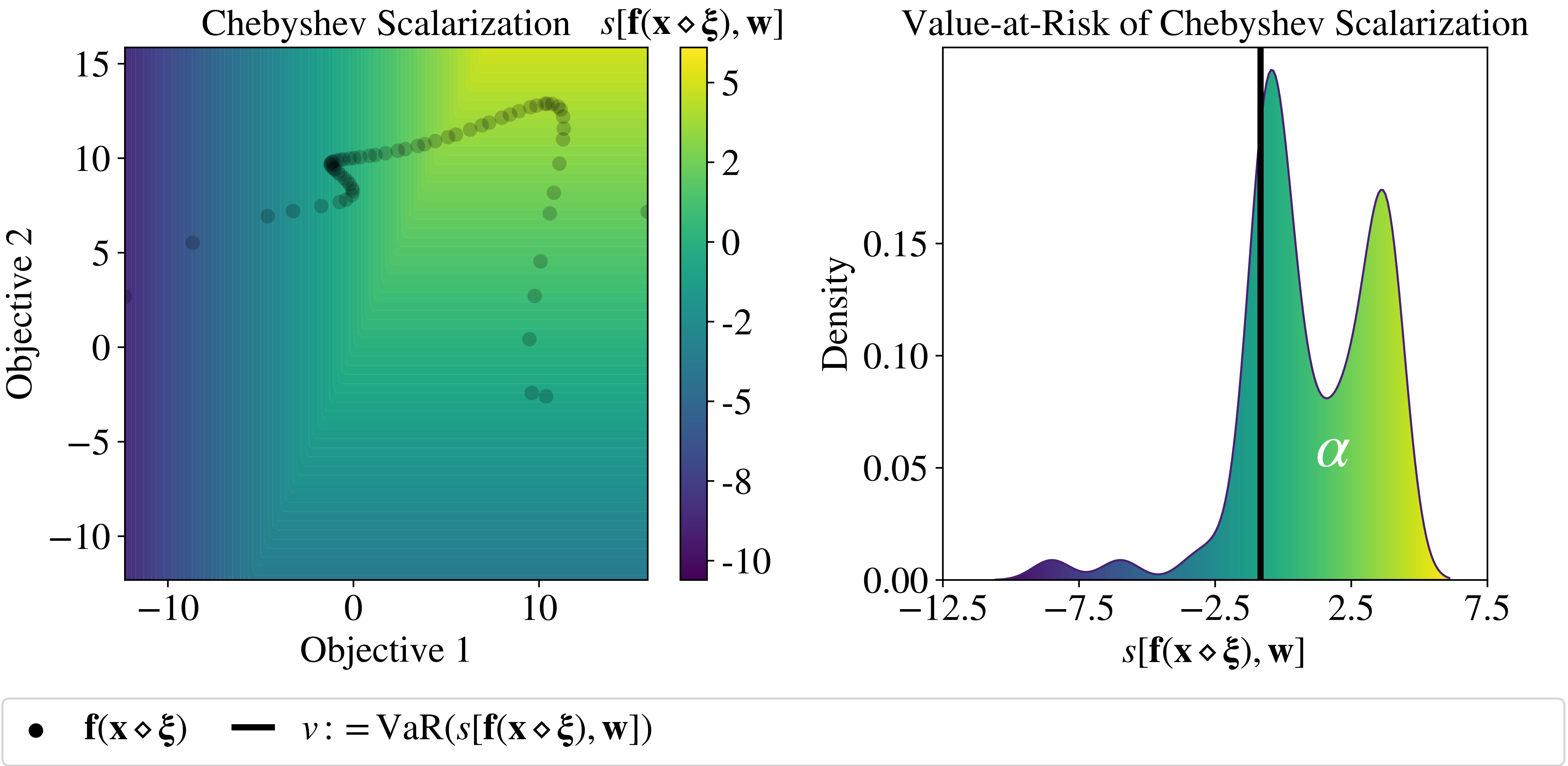
$$\boldsymbol{w} \in \Delta_+^{M-1}$$



VaR of Scalarization



VaR of Scalarization



VaR of Scalarization

VaR of Scalarization

VaR of Scalarization

$$\mathrm{VaR}_\alpha\left[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}), \boldsymbol{w}]\right] = \sup\{z \in \mathbb{R} : P\left[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}), \boldsymbol{w}] \geq z\right] \geq \alpha\}.$$

VaR of Scalarization

VaR of Scalarization

$$\mathrm{VAR}_\alpha\left[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}),\boldsymbol{w}]\right] = \sup\{z \in \mathbb{R} : P\left[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}),\boldsymbol{w}] \geq z\right] \geq \alpha\}.$$

Bijection (Main Result)

MVaR

$$\mathrm{MVaR}_\alpha\left[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi})\right] = \mathrm{Pareto}\left(\left\{\boldsymbol{z} \in \mathbb{R}^M : P\left[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi}) \geq \boldsymbol{z}\right] \geq \alpha\right\}\right).$$

VaR of Scalarization

$$\mathrm{VaR}_\alpha\left[s[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi}), \boldsymbol{w}]\right] = \sup\{z \in \mathbb{R} : P\left[s[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi}), \boldsymbol{w}] \geq z\right] \geq \alpha\}.$$

Bijection (Main Result)

MVaR

$$\mathrm{MVaR}_\alpha\left[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi})\right] = \mathrm{Pareto}\left(\{\boldsymbol{z}\in\mathbb{R}^M : P[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}) \geq \boldsymbol{z}] \geq \alpha\}\right).$$

VaR of Scalarization

$$\mathrm{VaR}_\alpha\left[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}),\boldsymbol{w}]\right] = \sup\{z\in\mathbb{R} : P[s[\boldsymbol{f}(\boldsymbol{x}\diamond\boldsymbol{\xi}),\boldsymbol{w}] \geq z] \geq \alpha\}.$$

MARS: MVaR Approximation via Random Scalarizations

- The bijection motivates a generative process for optimizing different MVaR trade-offs using Bayesian Optimization

Algorithm 1 MARS

Input: input noise distribution $P(\boldsymbol{\xi})$, search space $\mathcal{X}$, black-box objectives $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^M$, confidence level α

Initialize $\mathcal{D}_0 \leftarrow \emptyset$, $\mathrm{GP}_0 \leftarrow \mathrm{GP}(\mathbf{0}, k)$

for $n = 1$ **to** N **do**

  Sample $\boldsymbol{w} \sim \Delta_+^{M-1}$

  Set objective to be $l(\boldsymbol{x}) = \mathrm{VaR}_\alpha(s[\boldsymbol{f}(\boldsymbol{x} \diamond \boldsymbol{\xi}), \boldsymbol{w}])$

  $\boldsymbol{x}_n \leftarrow \arg\max_{\boldsymbol{x} \in \mathcal{X}} \mathrm{acq}(\boldsymbol{x}, l)$

  Evaluate $\boldsymbol{f}(\boldsymbol{x}_n), \mathcal{D}_n \leftarrow \mathcal{D}_{n-1} \cup \{\boldsymbol{x}_n, \boldsymbol{f}(\boldsymbol{x}_n)\}$

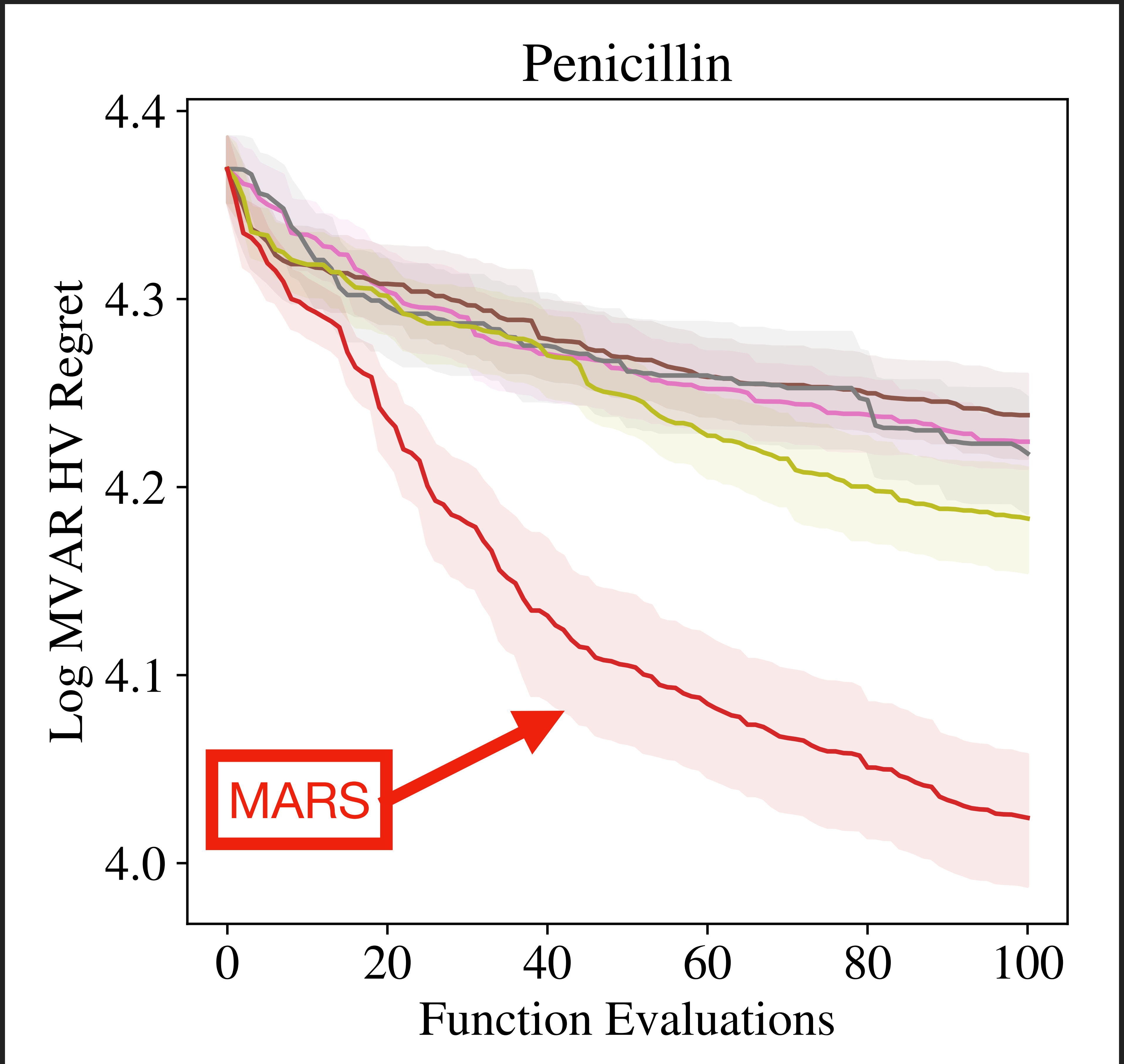
  Update posterior GP_n conditional on $\{\boldsymbol{x}_n, \boldsymbol{f}(\boldsymbol{x}_n)\}$

end for

Experiment: Penicillin Production

- Objectives:
  - Penicillin yield (maximize)
  - CO2 output (minimize)
  - Time-to-ferment (minimize)
- 7 Parameters:

Parameter	Noise Level
Culture Volume	3%
Biomass Concentration	3%
Temperature	0.5%
Glucose Concentration	2%
Substrate Feed Rate	1%
Substrate Feed Concentration	1%
H^+ Concentration	1%



Paper and Open Source Code

Paper: https://arxiv.org/abs/2202.07549

Code: github.com/facebookresearch/robust_mobo

BoTorch