Learning Decision-Focused Uncertainty Sets for Robust Optimization

Bartolomeo Stellato

Department of Operations Research and Financial Engineering

Department of Electrical and Computer Engineering

Department of Computer Science

PRINCETON UNIVERSITY

AAAI Workshop on Learnable Optimization — Feb 26 2024

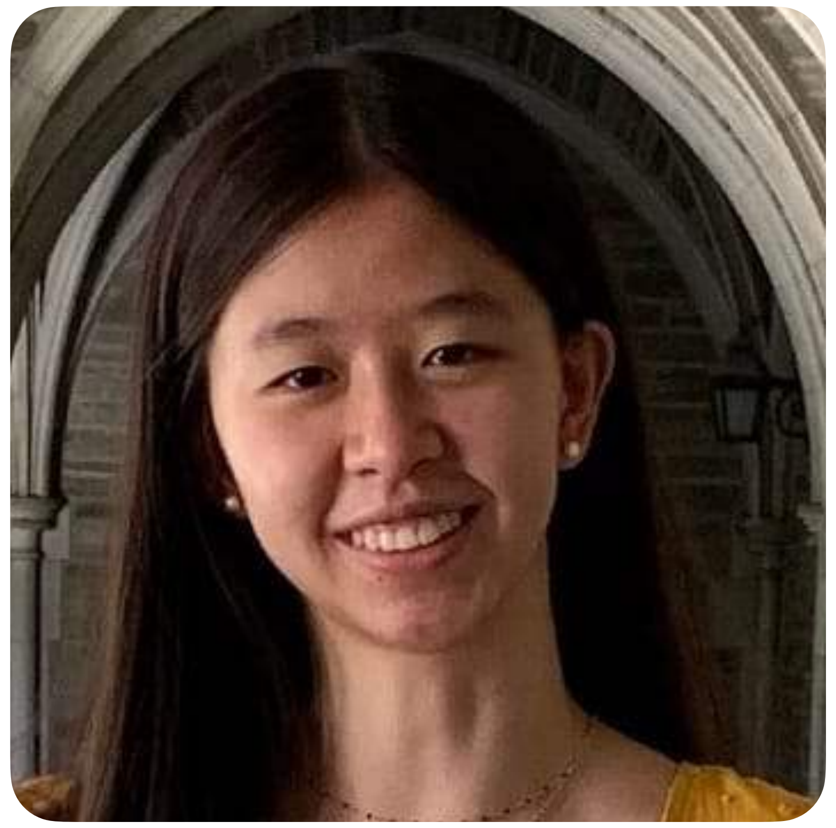
Joint work with

Irina Wang
Princeton ORFE

Bart Van Parys
MIT

Amit Solomon
Princeton OIT

Cole Becker
MIT



Annie Liang
Princeton ORFE

Intelligent systems make high-stakes decisions in an uncertain world

Intelligent systems make high-stakes decisions in an uncertain world

self-driving taxis

Intelligent systems make high-stakes decisions in an uncertain world

self-driving taxis

energy grids

Intelligent systems make high-stakes decisions in an uncertain world

self-driving taxis

energy grids

finance

Intelligent systems make high-stakes decisions in an uncertain world

self-driving taxis

energy grids

finance

robotics

Warm-up example

Portfolio optimization with penalty on trades

objective

$$-u^T x + \lambda \|x - x^{\mathrm{prev}}\|_1$$

Portfolio optimization with penalty on trades

uncertain returns

$$u \sim \mathcal{N}(\mu, \Sigma)$$

$$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \ \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$$

objective

$$-u^T x + \lambda \left\| x - x^{\text{prev}} \right\|_1$$

Portfolio optimization with penalty on trades

uncertain returns

$u \sim \mathcal{N}(\mu, \Sigma)$

$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \ \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$

objective

$-u^T x + \lambda \|x - x^{\text{prev}}\|_1$

previous holdings

$x^{\text{prev}} \sim \text{Dir}(\alpha)$

$\alpha = (2.5, 1)$

Portfolio optimization with penalty on trades

uncertain returns

$$u \sim \mathcal{N}(\mu, \Sigma)$$

$$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \; \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$$

objective

$$-u^T x + \lambda \|x - x^{\text{prev}}\|_1$$

investments decisions

previous holdings

$$x^{\text{prev}} \sim \text{Dir}(\alpha)$$
$$\alpha = (2.5, 1)$$

Portfolio optimization with penalty on trades

uncertain
returns

$$u \sim \mathcal{N}(\mu, \Sigma)$$

$$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$$

objective

$$-u^T x + \lambda \|x - x^{\text{prev}}\|_1$$

previous
holdings

$$x^{\text{prev}} \sim \mathrm{Dir}(\alpha)$$
$$\alpha = (2.5, 1)$$

investments decisions

robust problem
reformulation

$$\begin{array}{ll} \text{minimize} & t + \lambda \|x - x^{\text{prev}}\|_1 \\ \text{subject to} & -u^T x \le t \quad \forall u \in \mathcal{U}(\theta) \\ & \mathbf{1}^T x = 1, \quad x \ge 0 \end{array}$$

Portfolio optimization with penalty on trades

uncertain returns

$$u \sim \mathcal{N}(\mu, \Sigma)$$

$$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \ \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$$

objective

$$-u^T x + \lambda \|x - x^{\text{prev}}\|_1$$

investments decisions

previous holdings

$$x^{\text{prev}} \sim \mathrm{Dir}(\alpha)$$
$$\alpha = (2.5, 1)$$

robust problem reformulation

$$\begin{array}{ll} \text{minimize} & t + \lambda \|x - x^{\text{prev}}\|_1 \\ \text{subject to} & -u^T x \le t \quad \forall u \in \mathcal{U}(\theta) \\ & \mathbf{1}^T x = 1, \quad x \ge 0 \end{array}$$

uncertainty set

Portfolio optimization with penalty on trades

uncertain returns

$u \sim \mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} 0.5 & -0.3 \\ -0.3 & 0.4 \end{bmatrix}$$

objective

$$-u^T x + \lambda \|x - x^{\text{prev}}\|_1$$

previous holdings

$x^{\text{prev}} \sim \text{Dir}(\alpha)$

$\alpha = (2.5, 1)$

investments decisions

robust problem reformulation

$$\begin{array}{ll} \text{minimize} & t + \lambda \|x - x^{\text{prev}}\|_1 \\ \text{subject to} & -u^T x \le t \quad \forall u \in \mathcal{U}(\theta) \\ & \mathbf{1}^T x = 1, \quad x \ge 0 \end{array}$$

uncertainty set

how do we pick the uncertainty set?

Mean-variance vs reshaped uncertainty sets

parameters

$$\theta = (A, b)$$

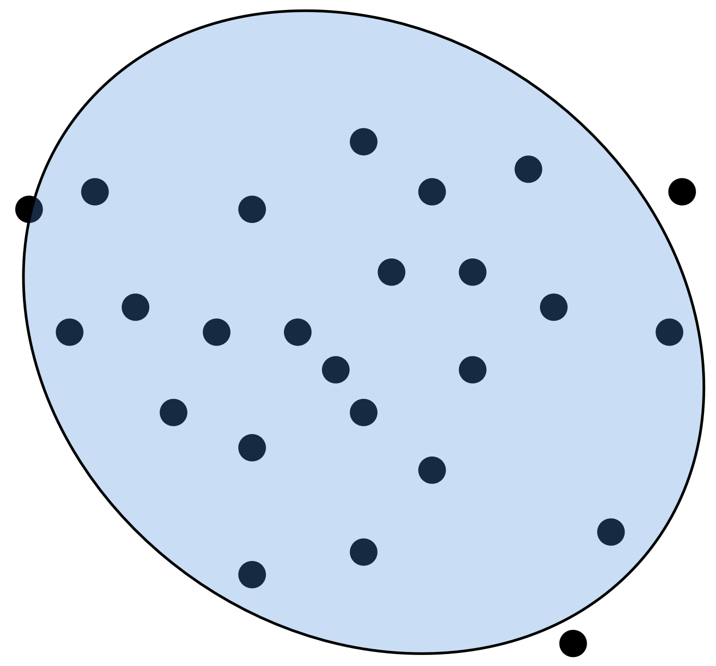
Mean-variance vs reshaped uncertainty sets

parameters
$$\theta = (A, b)$$

mean-variance set

$$\mathcal{U}^{\mathrm{mv}}(\theta) = \{u = \hat{\mu} + \hat{\Sigma}^{1/2} z \mid \|z\|_2 \le \rho\} = \{b^{\mathrm{mv}} + A^{\mathrm{mv}} z \mid \|z\|_2 \le \rho\}$$

empirical
mean and covariance



Mean-variance vs reshaped uncertainty sets

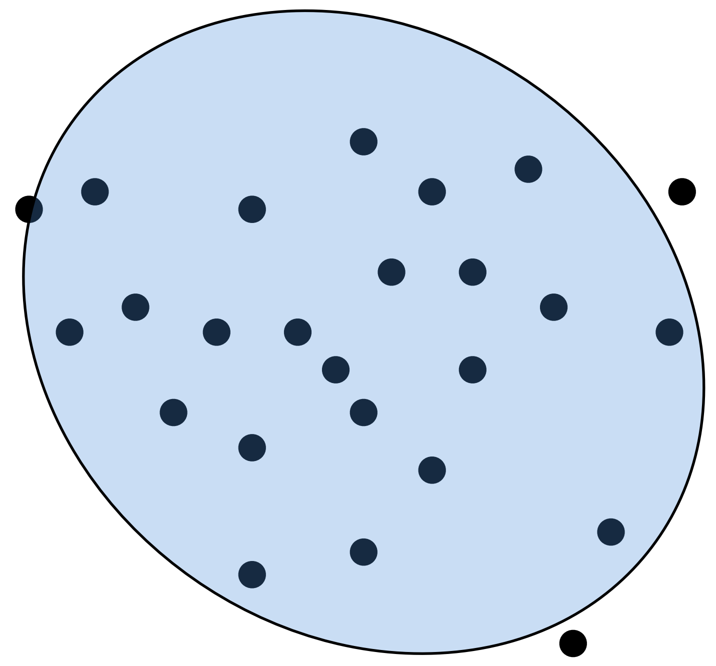
parameters

$$\theta = (A, b)$$

mean-variance set

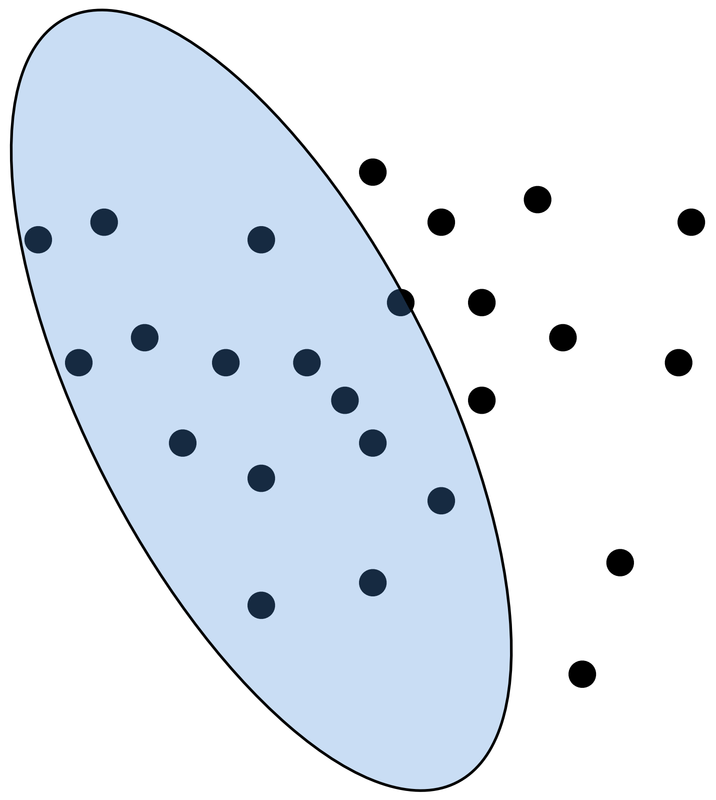
$$\mathcal{U}^{\mathrm{mv}}(\theta) = \{u = \hat{\mu} + \hat{\Sigma}^{1/2} z \mid \|z\|_2 \leq \rho\} = \{b^{\mathrm{mv}} + A^{\mathrm{mv}} z \mid \|z\|_2 \leq \rho\}$$

empirical
mean and covariance



reshaped uncertainty set

$$\mathcal{U}^{\mathrm{re}}(\theta) = \{u = b^{\mathrm{re}} + A^{\mathrm{re}} z \mid \|z\|_2 \leq \rho\}$$



Mean-variance vs reshaped uncertainty sets

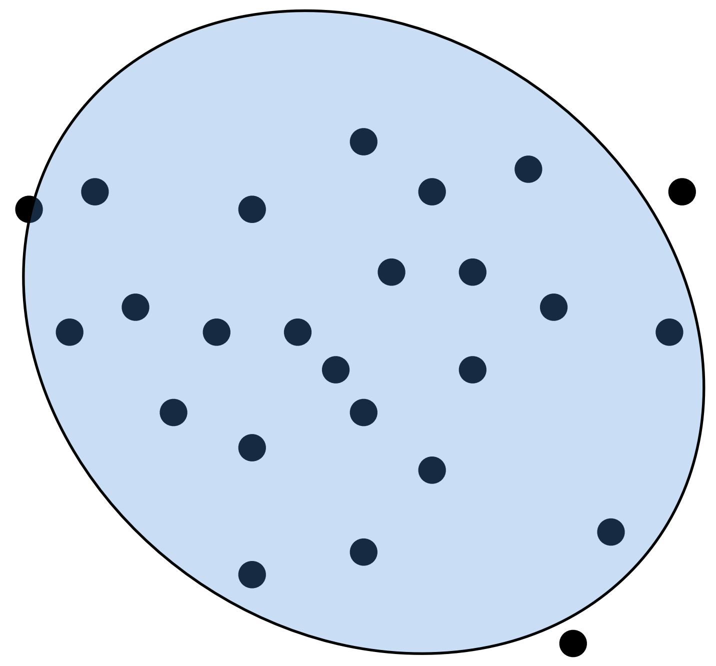
parameters

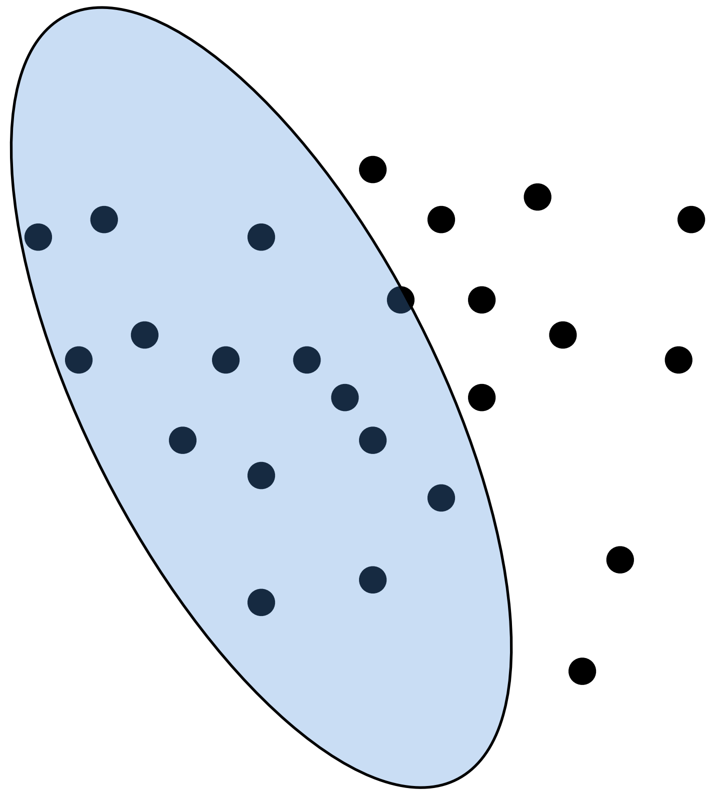
$$\theta = (A, b)$$

mean-variance set

$$\mathcal{U}^{\mathrm{mv}}(\theta) = \{u = \hat{\mu} + \hat{\Sigma}^{1/2} z \mid \|z\|_2 \leq \rho\} = \{b^{\mathrm{mv}} + A^{\mathrm{mv}} z \mid \|z\|_2 \leq \rho\}$$

empirical
mean and covariance





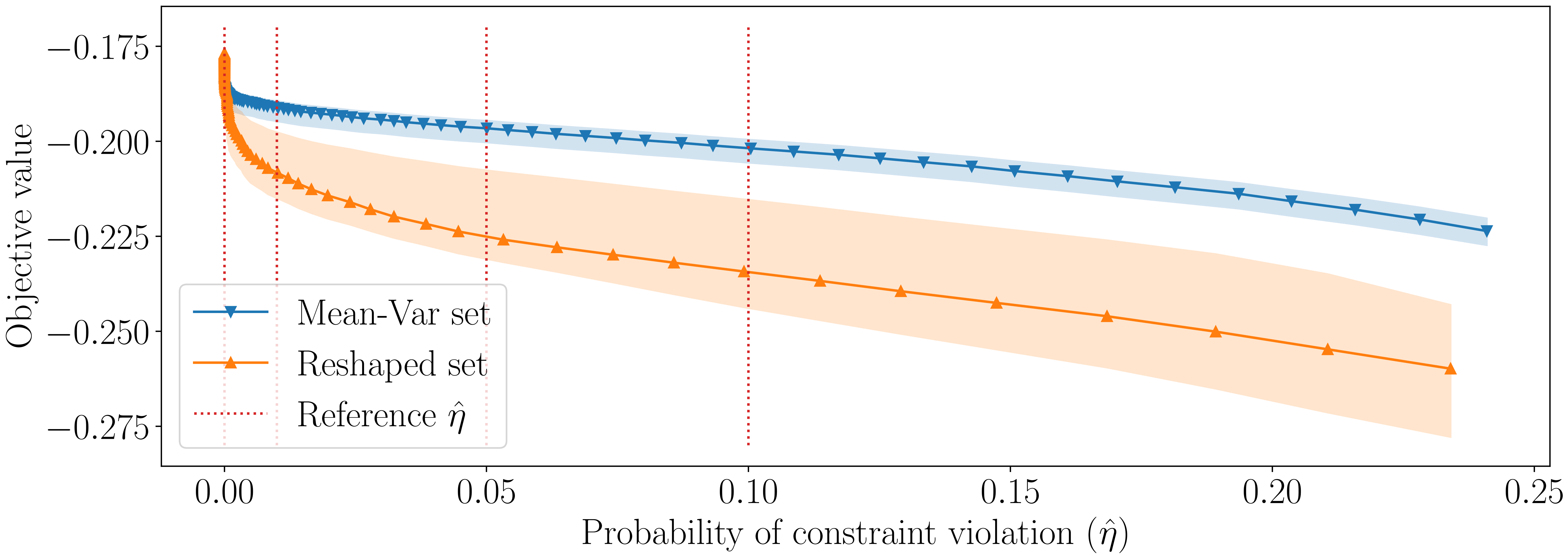
reshaped uncertainty set

$$\mathcal{U}^{\mathrm{re}}(\theta) = \{u = b^{\mathrm{re}} + A^{\mathrm{re}} z \mid \|z\|_2 \leq \rho\}$$

can the reshaped set
do better?

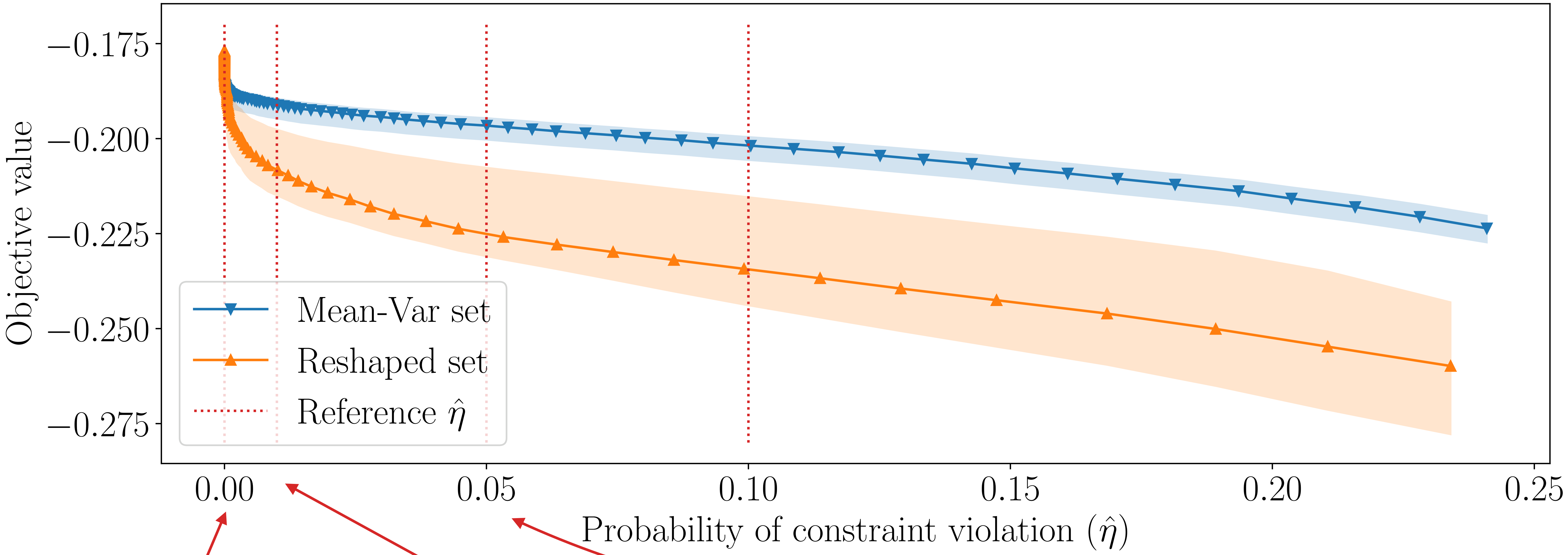
Reshaped set performs much better

pareto curves
for varying size
ρ



Reshaped set performs much better

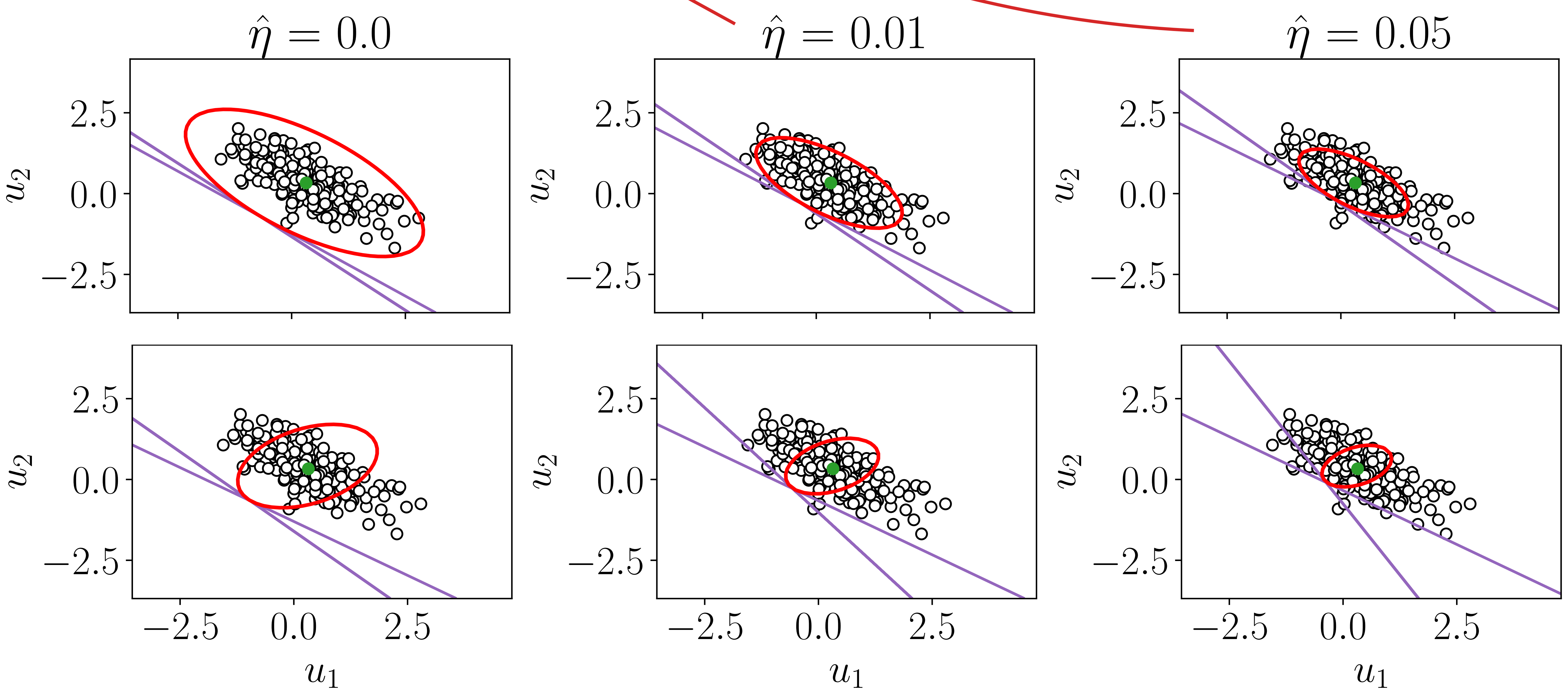
pareto curves for varying size ρ



level curves $-u^T x^\star - t^\star = 0$

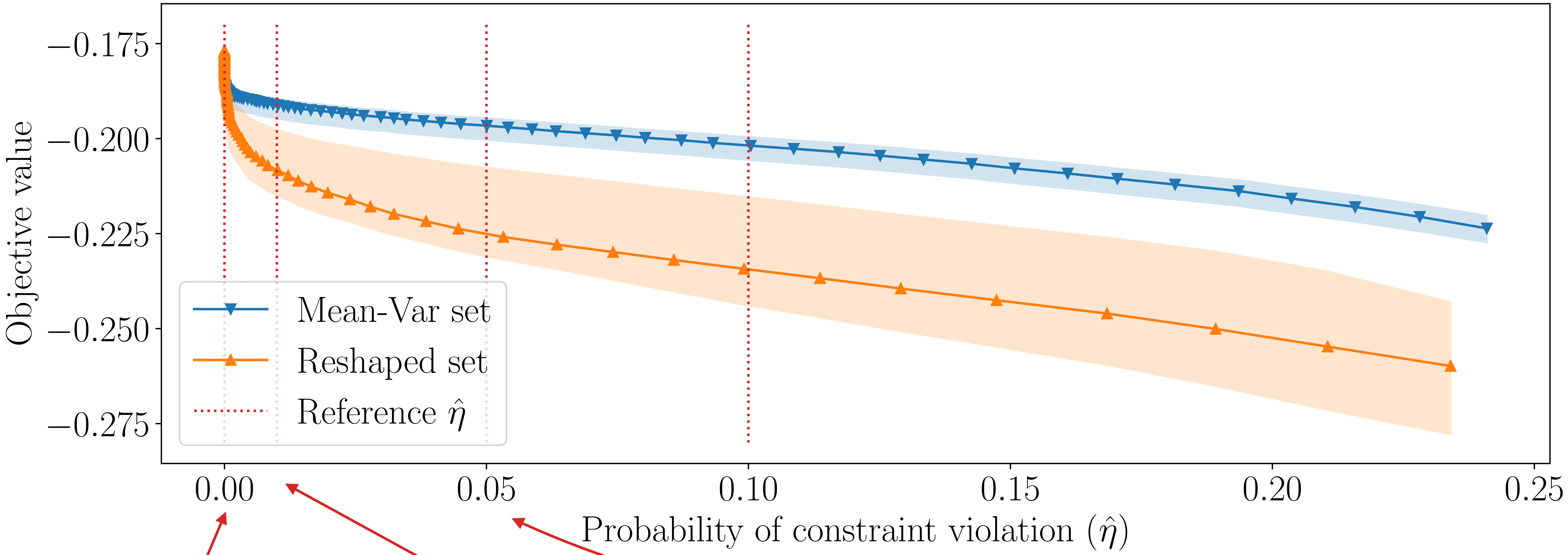
mean-variance $\mathcal{U}^{\rm mv}(\theta)$

reshaped $\mathcal{U}^{\rm re}(\theta)$



Reshaped set performs much better

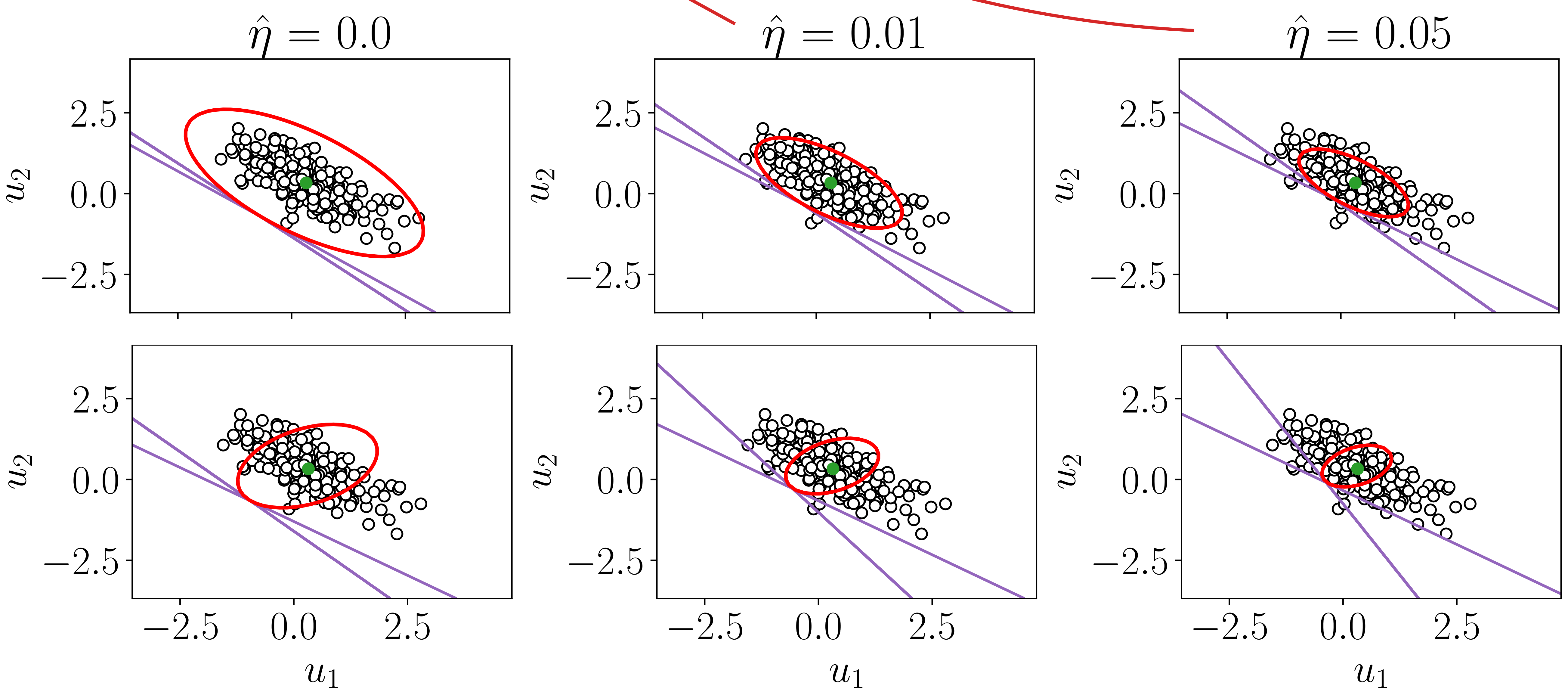
pareto curves
for varying size
ρ



level curves
$-u^T x^\star - t^\star = 0$

mean-variance
$\mathcal{U}^{\rm mv}(\theta)$

reshaped
$\mathcal{U}^{\rm re}(\theta)$



how can we find the reshaped set?

Problem setup

Decision-making with uncertain constraints

parametric robust optimization

$$
\begin{aligned}
x(\theta, y) \in \quad & \text{argmin} && f(x, y) \\
& \text{subject to} && g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{aligned}
$$

Decision-making with uncertain constraints

parametric robust optimization

$$
\begin{aligned}
x(\theta, y) \in \quad & \text{argmin} && f(x, y) \\
& \text{subject to} && g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{aligned}
$$

decisions

Decision-making with uncertain constraints

parametric robust optimization

$$
\begin{array}{rll}
x(\theta, y) \in & \text{argmin} & f(x, y) \\
& \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{array}
$$

decisions

uncertain parameter

Decision-making with uncertain constraints

parametric robust optimization

family/context parameter

$$
\begin{aligned}
x(\theta, y) \in \quad & \text{argmin} \quad && f(x, y) \\
& \text{subject to} \quad && g(x, u, y) \le 0 \quad \forall u \in \mathcal{U}(\theta)
\end{aligned}
$$

decisions

uncertain parameter

Decision-making with uncertain constraints

parametric robust optimization

family/context parameter

$$
\begin{aligned}
x(\theta, y) \in \quad & \text{argmin} && f(x, y) \\
& \text{subject to} && g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{aligned}
$$

decisions

uncertain parameter

probabilistic guarantees

$$
\mathbf{P}_{(u,y)}\big(g(x(\theta, y), u, y) \leq 0\big) \geq 1 - \eta
$$

Decision-making with uncertain constraints

parametric robust optimization

family/context parameter

$$x(\theta, y) \in \quad \text{argmin} \quad f(x, y)$$
$$\text{subject to} \quad g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)$$

decisions

uncertain parameter

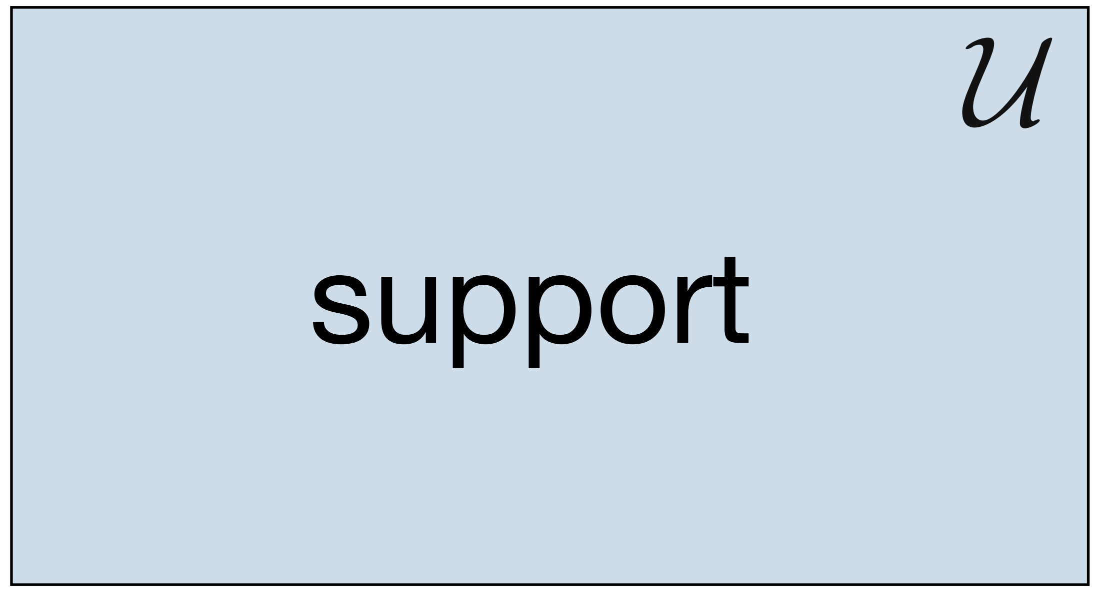
probabilistic guarantees

$$\mathbf{P}_{(u,y)}\big(g(x(\theta, y), u, y) \leq 0\big) \geq 1 - \eta$$

Can we construct a set that ensures the probabilistic guarantees?

Picking the uncertainty set is difficult

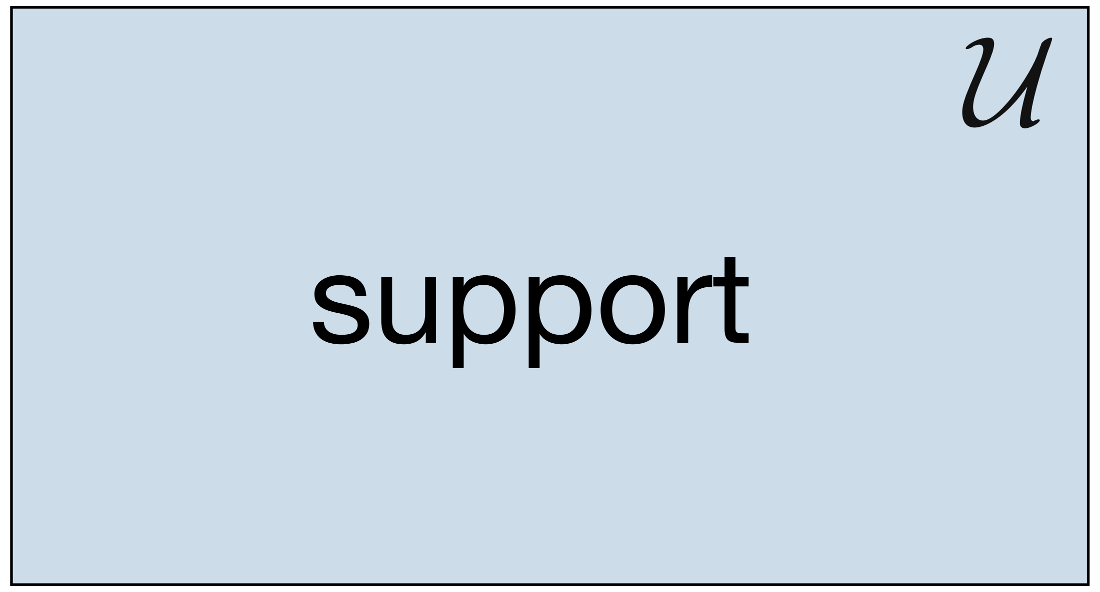
Worst-case approach



❌ Very conservative

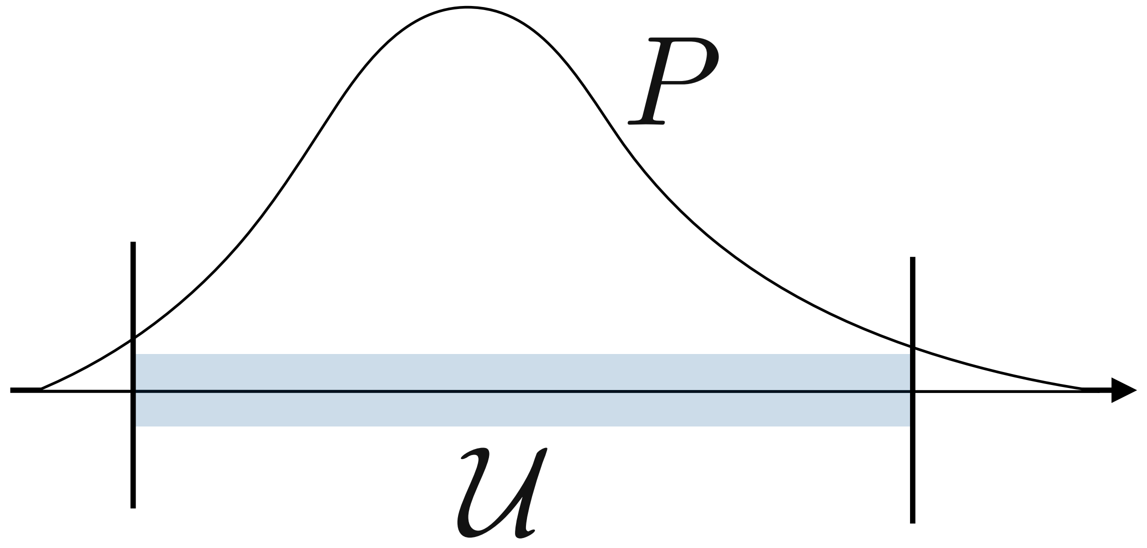
Picking the uncertainty set is difficult

Worst-case approach



❌ Very conservative

Probabilistic approach

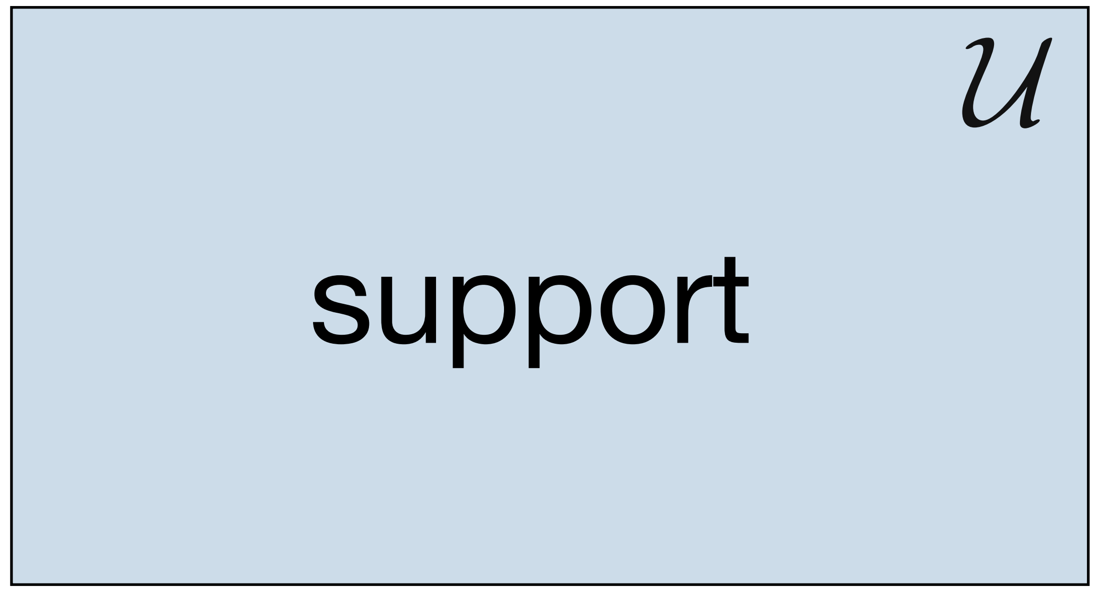


❌ nobody knows P

"Probabilistic Guarantees in Robust Optimization", D. Bertsimas, D. den Hertog, and J. Pauphilet (2019)

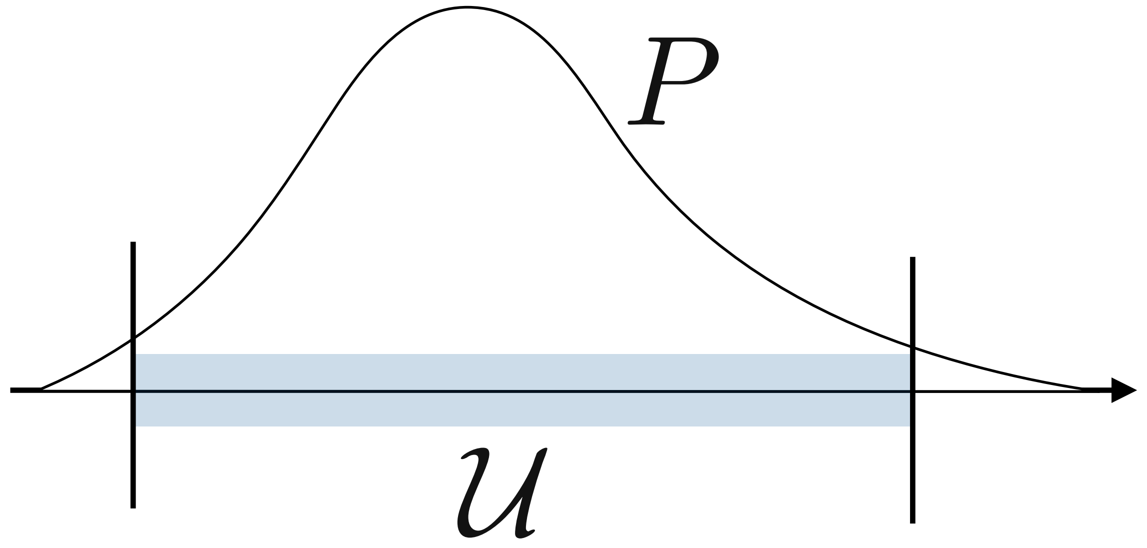
Picking the uncertainty set is difficult

Worst-case approach



❌ Very conservative

Probabilistic approach



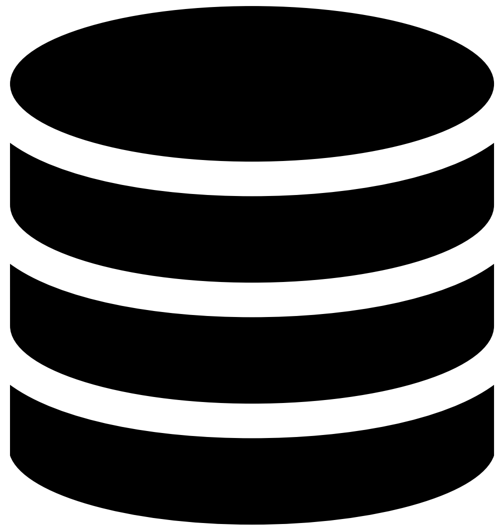
❌ nobody knows P

Data-driven approach

Can we use data to construct uncertainty sets?

“Probabilistic Guarantees in Robust Optimization”, D. Bertsimas, D. den Hertog, and J. Pauphilet (2019)

Data-driven methods for robust optimization



Hypothesis testing

D. Bertsimas, V. Gupta,
and N. Kallus (2014)

Data-driven methods for robust optimization

Hypothesis testing

D. Bertsimas, V. Gupta, and N. Kallus (2014)

Quantile estimation

L. Jeff Hong, Z. Huang, and H. Lam (2021)

Data-driven methods for robust optimization

Hypothesis testing

D. Bertsimas, V. Gupta, and N. Kallus (2014)

Quantile estimation

L. Jeff Hong, Z. Huang, and H. Lam (2021)

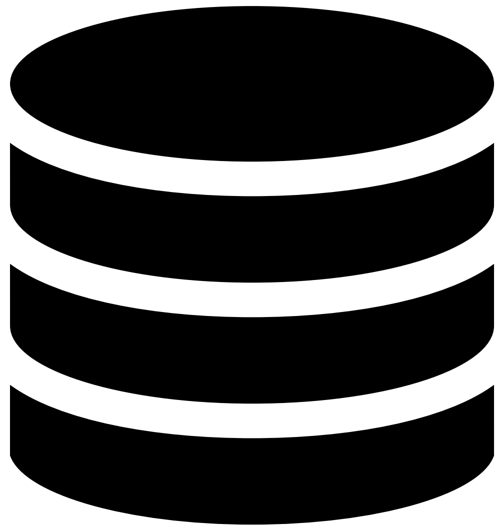
Wasserstein Distributionally Robust Optimization

P. M. Esfahani and D. Kuhn. (2018).

D. Bertsimas, S. Shtern, B. Sturt (2022)

I. Wang, C. Becker, B. Van Parys, and B. Stellato (2023)

Data-driven methods for robust optimization



Hypothesis testing

D. Bertsimas, V. Gupta,
and N. Kallus (2014)

Quantile estimation

L. Jeff Hong, Z. Huang, and H. Lam (2021)

Wasserstein Distributionally Robust Optimization

P. M. Esfahani and D. Kuhn. (2018).

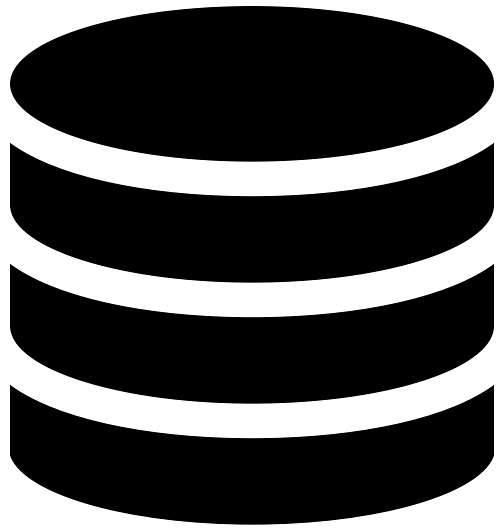
D. Bertsimas, S. Shtern, B. Sturt (2022)

I. Wang, C. Becker, B. Van Parys, and B. Stellato (2023)

Deep Learning

M. Goerigk, J. Kurtz (2023)

Data-driven methods for robust optimization



Hypothesis testing
D. Bertsimas, V. Gupta, and N. Kallus (2014)

Quantile estimation
L. Jeff Hong, Z. Huang, and H. Lam (2021)

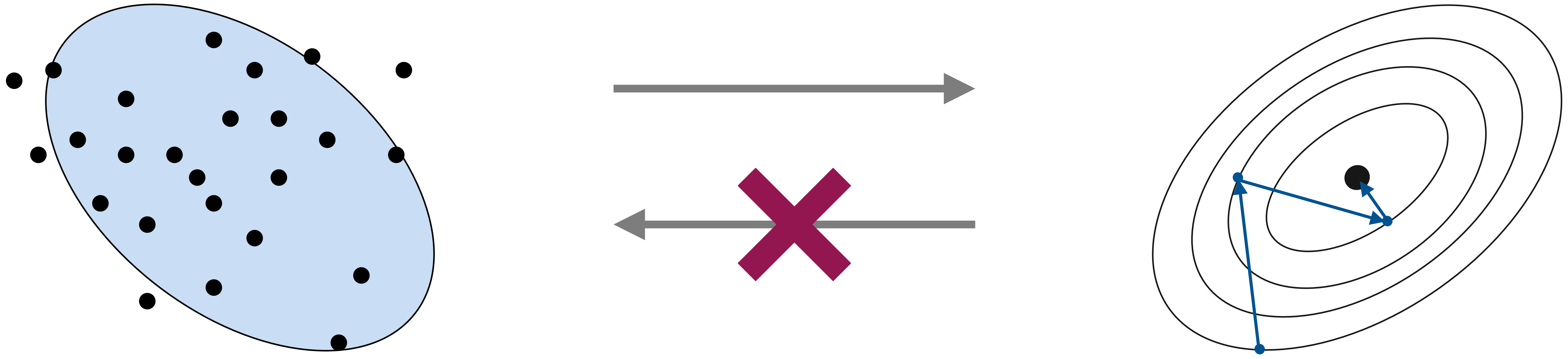
Wasserstein Distributionally Robust Optimization
P. M. Esfahani and D. Kuhn. (2018).
D. Bertsimas, S. Shtern, B. Sturt (2022)
I. Wang, C. Becker, B. Van Parys, and B. Stellato (2023)

Deep Learning
M. Goerigk, J. Kurtz (2023)

Most approaches decouple

Uncertainty set construction

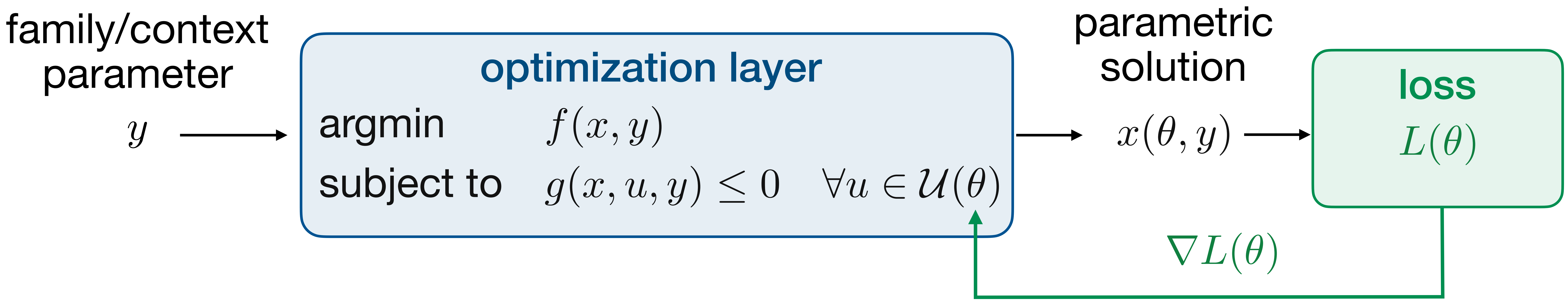
Downstream optimization task



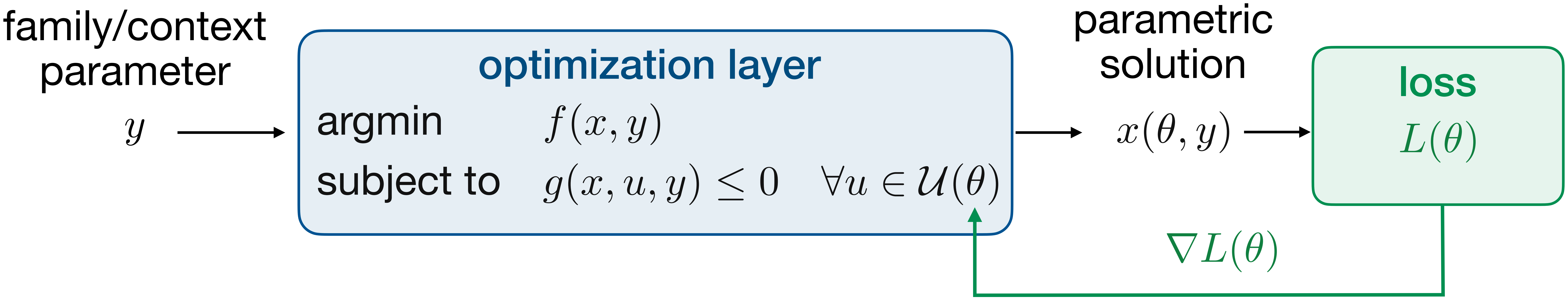
high coverage requirement

$$\mathbf{P}(u \in \mathcal{U}) \geq 1 - \epsilon$$

Leveraging the solution to tune the uncertainty sets



Leveraging the solution to tune the uncertainty sets



Main idea

Use differentiable optimization to automatically learn *shape* and *size*

Connections with Contextual Optimization

Contextual Optimization

$$x(y) \in \underset{x}{\text{argmin}}\, \mathbf{E}_{\mathbf{P}(u|y)}(f(x,u))$$

[D. Bertsimas and N. Kallus (2020)], [Elmachtoub and Grigas (2022)],
[H. Rahimian, B. Pagnoncelli (2022)]

“A Survey of Contextual Optimization Methods for Decision Making under Uncertainty”,
U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, T. Vidal (2023)

Connections with Contextual Optimization

Contextual Optimization

$$x(y) \in \operatorname*{argmin}_{x} \mathbf{E}_{\mathbf{P}(u|y)}(f(x,u))$$

[D. Bertsimas and N. Kallus (2020)], [Elmachtoub and Grigas (2022)],
[H. Rahimian, B. Pagnoncelli (2022)]

“A Survey of Contextual Optimization Methods for Decision Making under Uncertainty”,
U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, T. Vidal (2023)

Conditional Robust Optimization

$$x(y) \in \operatorname*{argmin}_{x} \max_{u \in \mathcal{U}(y)} f(x,u)$$

[A.R. Chenreddy, N. Bandi, E. Delage (2022)], [S. Ohmori (2021)],
[E. Persak, M. F. Anjos (2023)], [C.Sun, L. Liu, X. Li (2023)]

Connections with Contextual Optimization

Contextual Optimization

$$x(y) \in \underset{x}{\text{argmin}}\ \mathbf{E}_{\mathbf{P}(u|y)}(f(x,u))$$

[D. Bertsimas and N. Kallus (2020)], [Elmachtoub and Grigas (2022)],
[H. Rahimian, B. Pagnoncelli (2022)]

"A Survey of Contextual Optimization Methods for Decision Making under Uncertainty",
U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, T. Vidal (2023)

Conditional Robust Optimization

$$x(y) \in \underset{x}{\text{argmin}}\ \max_{u \in \mathcal{U}(y)} f(x,u)$$

[A.R. Chenreddy, N. Bandi, E. Delage (2022)], [S. Ohmori (2021)],
[E. Persak, M. F. Anjos (2023)], [C.Sun, L. Liu, X. Li (2023)]

Differences from our work

- Distribution/uncertainty set depends on y
- High coverage requirements $\mathbf{P}(u \in \mathcal{U}(y)) \geq 1 - \epsilon$
- Limited focus on uncertain constraints

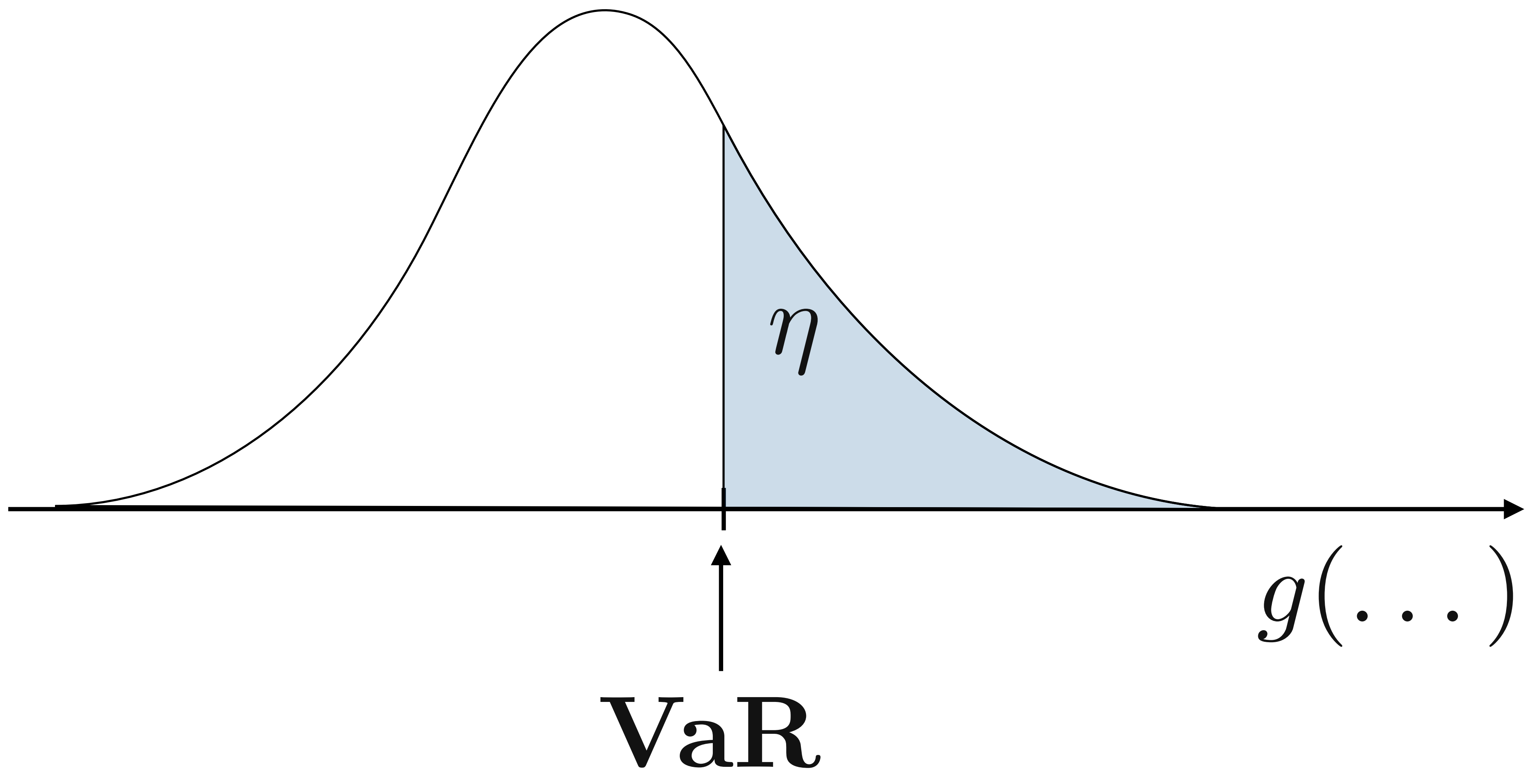
Learning problem formulation

Enforcing probabilistic guarantees with CVaR

probabilistic guarantees

$$\mathbf{P}_{(u,y)}\big(g(x(\theta,y),u,y) \leq 0\big) \geq 1-\eta$$

same as $\mathbf{VaR}(g(\dots),\eta) \leq 0$



Enforcing probabilistic guarantees with CVaR

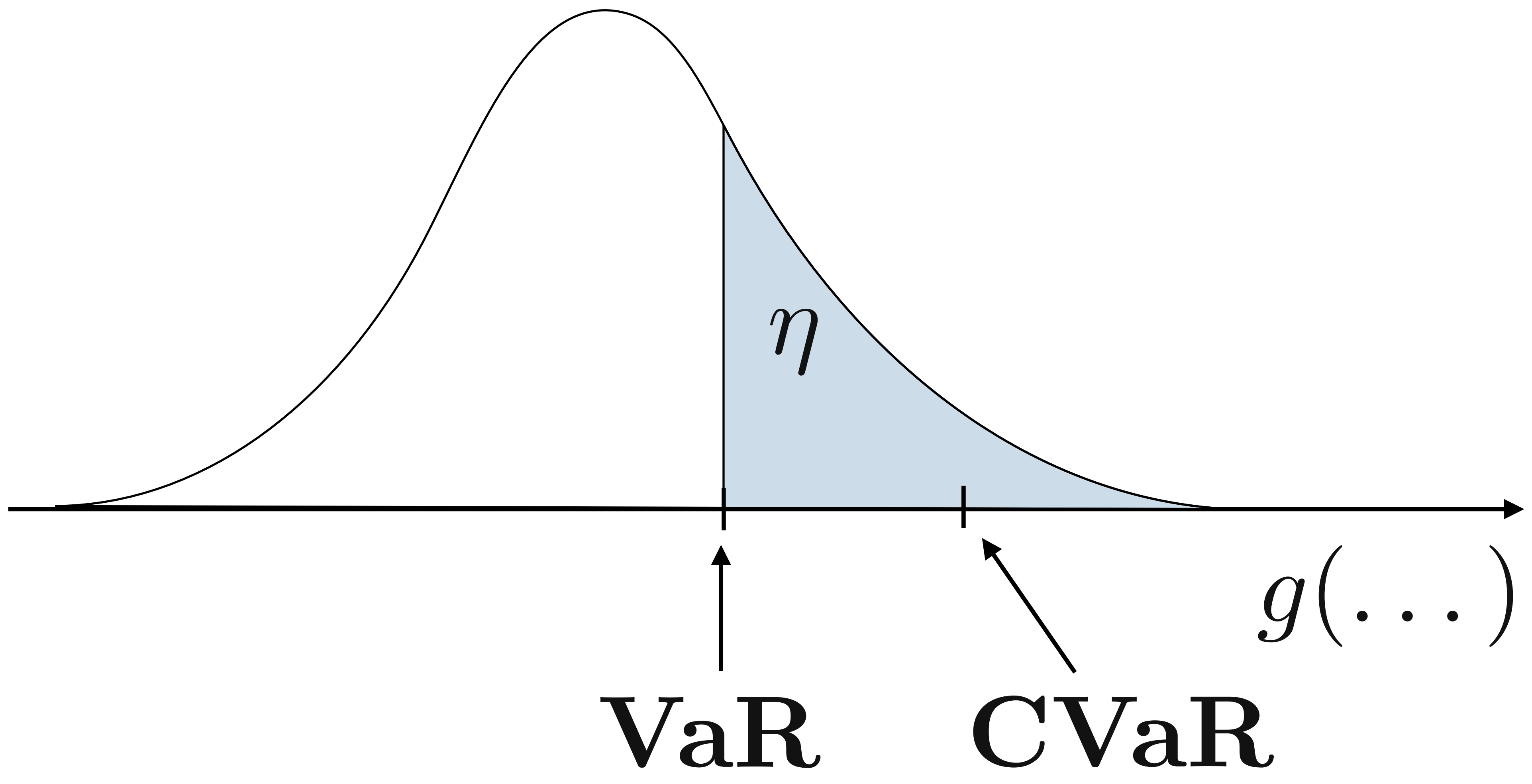
probabilistic guarantees

$$\mathbf{P}_{(u,y)}\big(g(x(\theta, y), u, y) \leq 0\big) \geq 1 - \eta$$

tractable approximation

$$\mathbf{CVaR}(g(x(\theta, y), u, y), \eta) \leq 0$$

same as $\mathbf{VaR}(g(\dots), \eta) \leq 0$

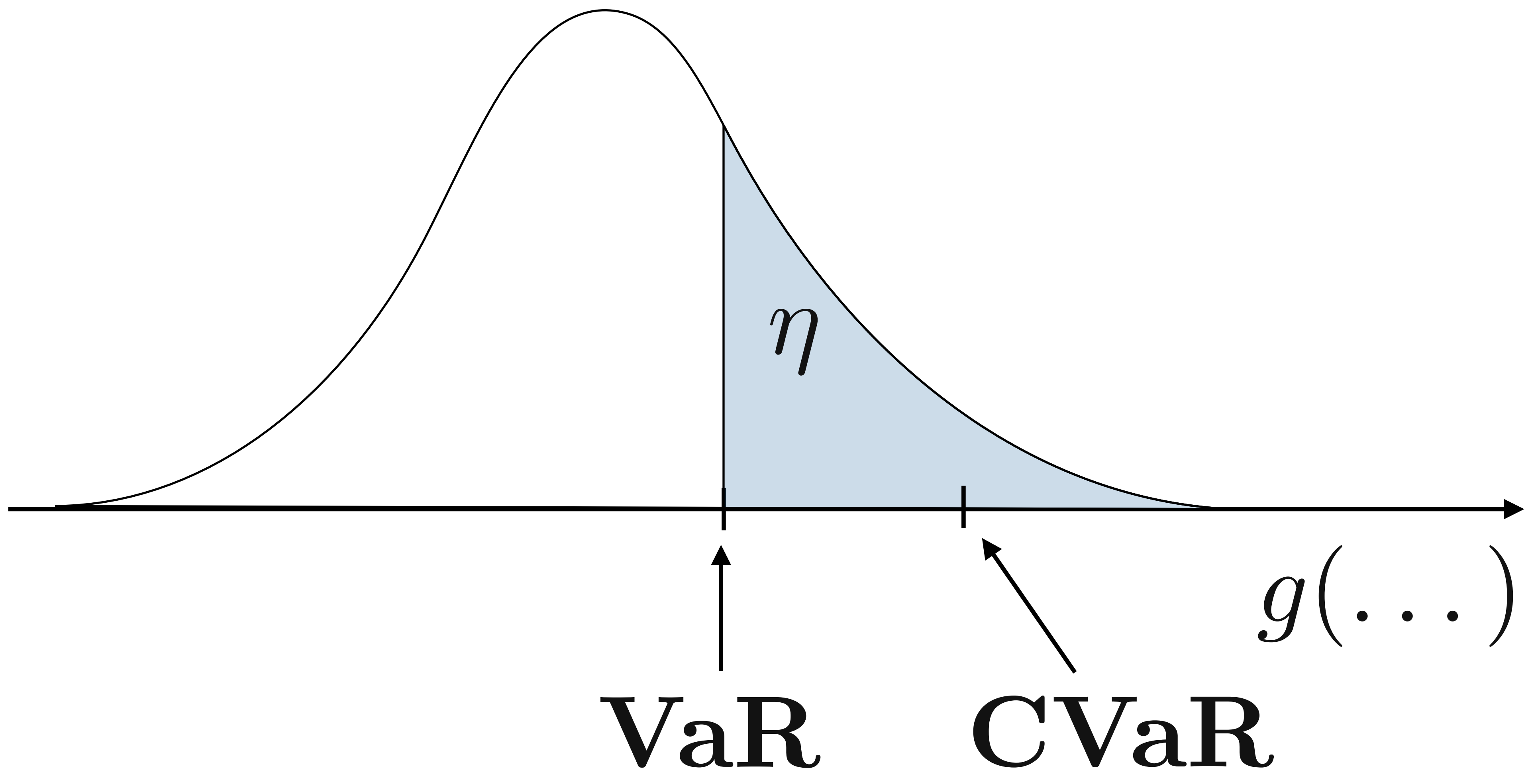


Enforcing probabilistic guarantees with CVaR

probabilistic guarantees

$$\mathbf{P}_{(u,y)}\big(g(x(\theta, y), u, y) \leq 0\big) \geq 1 - \eta$$

same as $\mathbf{VaR}(g(\dots), \eta) \leq 0$



tractable approximation

$$\mathbf{CVaR}(g(x(\theta, y), u, y), \eta) \leq 0$$

turn into constraint

$$\mathbf{E}_{(u,y)}\left(\frac{(g(x(\theta, y), u, y) - \alpha)_+}{\eta} + \alpha\right) \leq 0$$

Enforcing probabilistic guarantees with CVaR

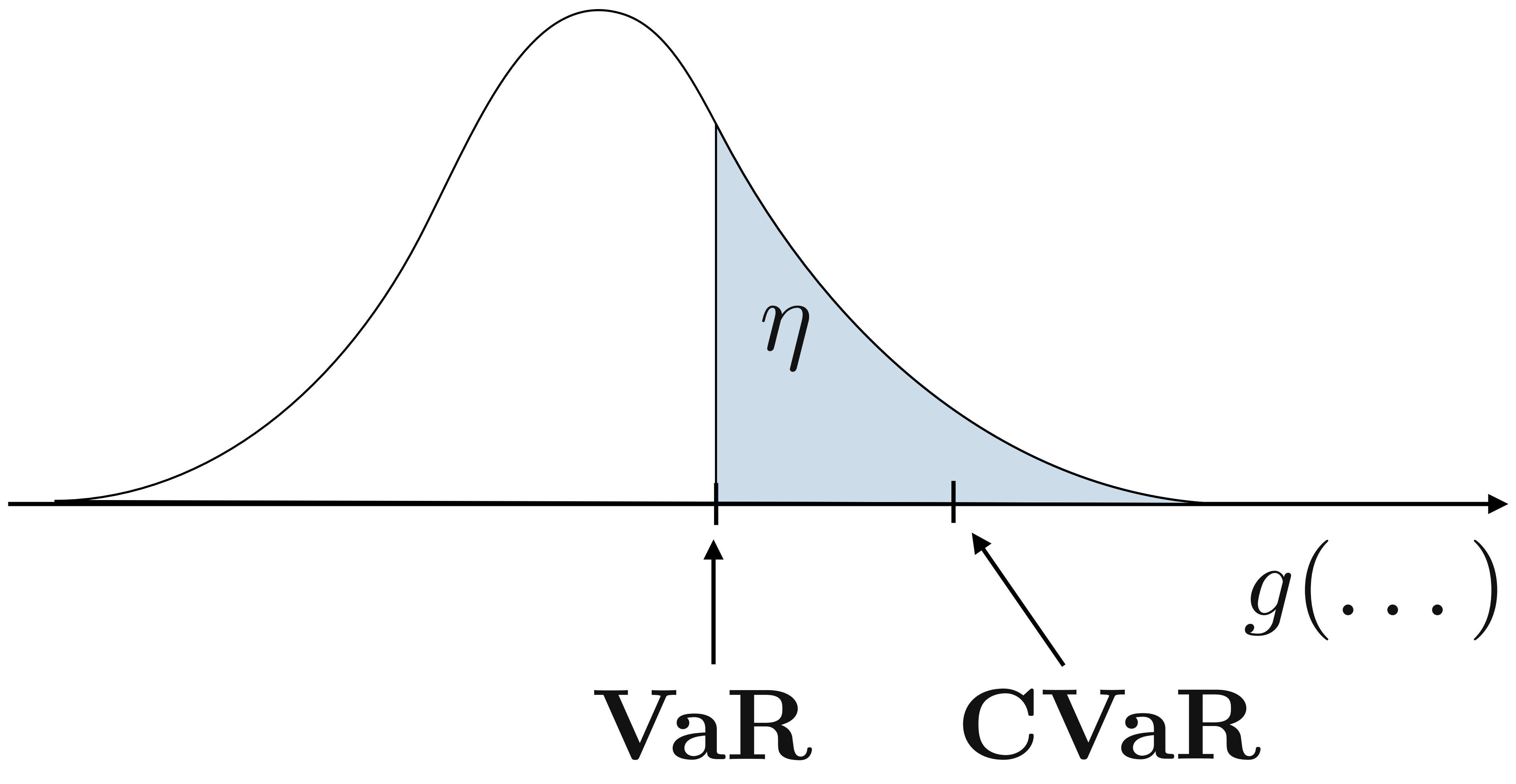
probabilistic guarantees

$$\mathbf{P}_{(u,y)}\big(g(x(\theta,y),u,y)\leq 0\big)\geq 1-\eta$$

same as $\mathbf{VaR}(g(\dots),\eta)\leq 0$

tractable approximation

$$\mathbf{CVaR}(g(x(\theta,y),u,y),\eta)\leq 0$$



turn into constraint

$$\mathbf{E}_{(u,y)}\left(\frac{(g(x(\theta,y),u,y)-\alpha)_+}{\eta}+\alpha\right)\leq 0 \xrightarrow{w=(u,y)} \mathbf{E}_w\left(h(\alpha,\theta,w)\right)\leq \kappa$$

threshold

Stochastic bilevel optimization to learn the uncertainty set

loss

$$\ell(\theta, w) = f(x(\theta, y), y)$$

training problem

$$\begin{array}{ll} \text{minimize} & \mathbf{E}_w[\ell(\theta, w)] \\ \text{subject to} & \mathbf{E}_w[h(\alpha, \theta, w)] \leq \kappa \end{array}$$

CVaR constraint

$$h(\alpha, \theta, w) = \frac{(g(x(\theta, y), u, y) - \alpha)_+}{\eta} + \alpha$$

decision variables
θ, α

random variables
$w = (u, y)$

Stochastic bilevel optimization to learn the uncertainty set

loss

$$\ell(\theta, w) = f(x(\theta, y), y)$$

training problem

$$\begin{array}{ll} \text{minimize} & \mathbf{E}_w[\ell(\theta, w)] \\ \text{subject to} & \mathbf{E}_w[h(\alpha, \theta, w)] \leq \kappa \end{array}$$

CVaR constraint

$$h(\alpha, \theta, w) = \frac{(g(x(\theta, y), u, y) - \alpha)_+}{\eta} + \alpha$$

decision variables
θ, α

random variables
$w = (u, y)$

inner robust problem

$$\begin{array}{lll} x(\theta, y) \in & \text{argmin} & f(x, y) \\ & \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array}$$

we must reformulate the infinite dimensional constraints

Uncertainty set parameters enter nicely in the reformulation

$$
\begin{array}{ll}
\text{minimize} & f(x, y) \\
\text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{array}
$$

Uncertainty set parameters enter nicely in the reformulation

$$
\begin{array}{ll}
\text{minimize} & f(x, y) \\
\text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)
\end{array}
$$

learned parameters

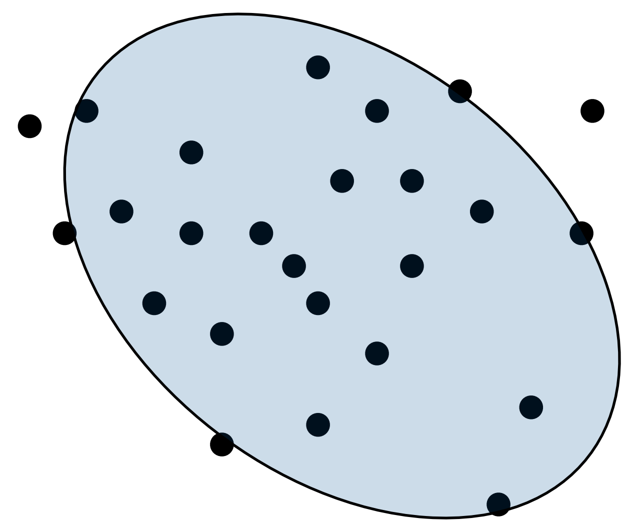
Uncertainty set parameters enter nicely in the reformulation

$$\begin{array}{ll} \text{minimize} & f(x, y) \\ \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array}$$

learned parameters

Example: ellipsoidal set

$$\mathcal{U}(\theta) = \{u = b + Az \mid \|z\|_2 \leq 1\}$$



Uncertainty set parameters enter nicely in the reformulation

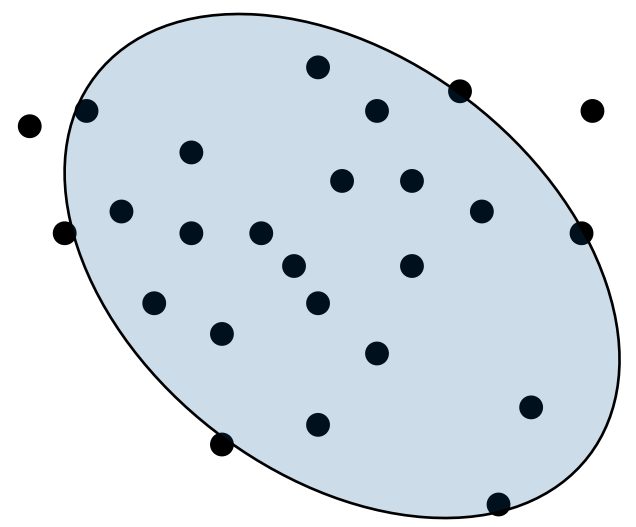
$$\begin{array}{ll} \text{minimize} & f(x, y) \\ \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array}$$

learned parameters

Example: ellipsoidal set

$$\mathcal{U}(\theta) = \{u = b + Az \mid \|z\|_2 \leq 1\}$$

$$\theta = (A, b)$$



Uncertainty set parameters enter nicely in the reformulation

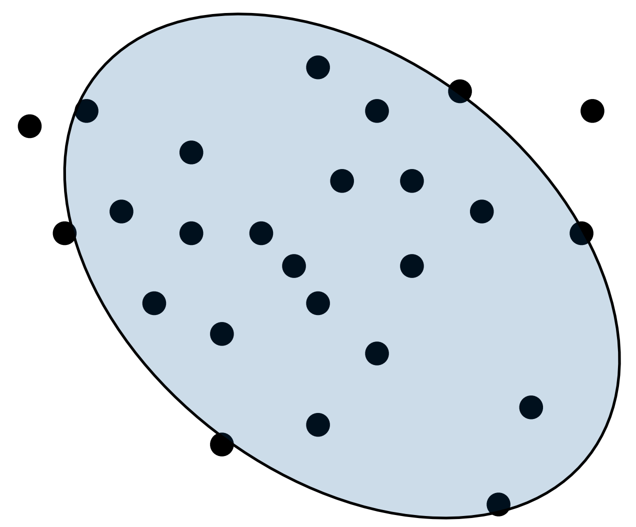
$$\begin{array}{ll} \text{minimize} & f(x, y) \\ \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array}$$

learned parameters

Example: ellipsoidal set

$$\mathcal{U}(\theta) = \{u = b + Az \mid \|z\|_2 \leq 1\}$$

$$\theta = (A, b)$$



linear constraint

$$g(x, u, y) = (y + Pu)^T x \leq 0, \quad \forall u \in \mathcal{U}(\theta)$$

Uncertainty set parameters enter nicely in the reformulation

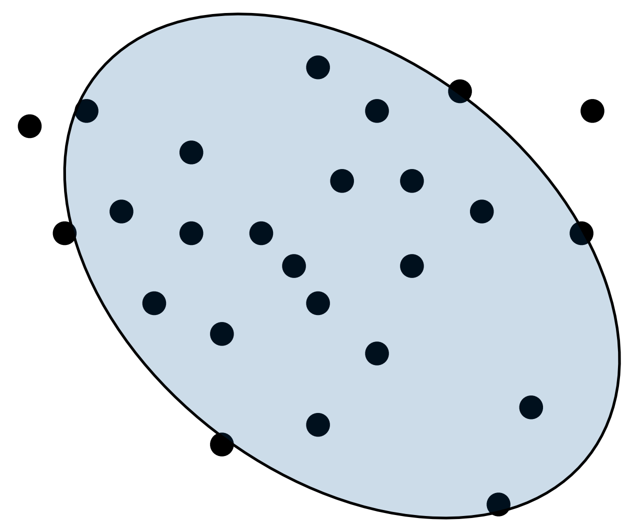
$$\begin{array}{ll} \text{minimize} & f(x, y) \\ \text{subject to} & g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array} \longleftarrow$$

learned parameters

Example: ellipsoidal set

$$\mathcal{U}(\theta) = \{u = b + Az \mid \|z\|_2 \leq 1\}$$

$$\theta = (A, b)$$



linear constraint

$$g(x, u, y) = (y + Pu)^T x \leq 0, \quad \forall u \in \mathcal{U}(\theta)$$

robust counterpart

$$y^T x + b^T P x + \|A^T P^T x\|_2 \leq 0$$

Solution algorithm

Constrained learning problem

$$F(\theta)$$

$$\begin{array}{ll} \text{minimize} & \boxed{\mathbf{E}_w[\ell(\theta, w)]} \\ \text{subject to} & \boxed{\mathbf{E}_w[h(\alpha, \theta, w)]} \leq \kappa \end{array}$$

$$H(\alpha, \theta)$$

Constrained learning problem

$$\begin{array}{ll} \text{minimize} & \underbrace{\mathbf{E}_w[\ell(\theta, w)]}_{F(\theta)} \\ \text{subject to} & \underbrace{\mathbf{E}_w[h(\alpha, \theta, w)]}_{H(\alpha,\theta)} \leq \kappa \end{array}$$

introduce slack variable $s \geq 0$

reformulated training problem

$$\begin{array}{ll} \text{minimize} & \mathbf{E}_w[\ell(\theta, w)] \\ \text{subject to} & \mathbf{E}_w[h(\alpha, \theta, w)] + s = \kappa \\ & s \geq 0 \end{array}$$

decision variables

θ, α, s

Stochastic augmented Lagrangian method

penalty

augmented Lagrangian

$$L(\alpha, \theta, s, \lambda, \mu) = F(\theta) + \lambda(H(\alpha, \theta) + s - \kappa) + \frac{\mu}{2}\|H(\alpha, \theta) + s - \kappa\|^2$$

multiplier

Stochastic augmented Lagrangian method

augmented Lagrangian

penalty

$$L(\alpha, \theta, s, \boxed{\lambda}, \boxed{\mu}) = F(\theta) + \boxed{\lambda}(H(\alpha, \theta) + s - \kappa) + \frac{\boxed{\mu}}{2}\|H(\alpha, \theta) + s - \kappa\|^2$$

multiplier

for $k = 1, \ldots, k_{\max}$ **do**

$G^k \leftarrow \hat{\nabla} L(\alpha^k, \theta^k, s^k, \lambda^k, \mu^k)$ **implicit differentiation**

$(\theta^{k+1}, \alpha^{k+1}, s^{k+1}) \leftarrow (\theta^k, \alpha^k, s^k) - tG^k$

$s^{k+1} \leftarrow (s^{k+1})_+$

if $\|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2 \leq \tau H_{\text{best}}$ **then**

$\lambda^{k+1} \leftarrow \lambda^k + \mu^k(H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa)$

$\mu^{k+1} \leftarrow \mu^k$

$H_{\text{best}} \leftarrow \|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2$

else

Choose $\mu^{k+1} > \mu^k$

$\lambda^{k+1} \leftarrow \lambda^k$

Stochastic augmented Lagrangian method

augmented Lagrangian

penalty

$$L(\alpha, \theta, s, \lambda, \mu) = F(\theta) + \lambda(H(\alpha, \theta) + s - \kappa) + \frac{\mu}{2}\|H(\alpha, \theta) + s - \kappa\|^2$$

multiplier

for $k = 1, \ldots, k_{\max}$ **do**
- $G^k \leftarrow \hat{\nabla} L(\alpha^k, \theta^k, s^k, \lambda^k, \mu^k)$ **implicit differentiation**
- $(\theta^{k+1}, \alpha^{k+1}, s^{k+1}) \leftarrow (\theta^k, \alpha^k, s^k) - tG^k$
- $s^{k+1} \leftarrow (s^{k+1})_+$

  update primal variables
- **if** $\|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2 \leq \tau H_{\text{best}}$ **then**
  - $\lambda^{k+1} \leftarrow \lambda^k + \mu^k(H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa)$
  - $\mu^{k+1} \leftarrow \mu^k$
  - $H_{\text{best}} \leftarrow \|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2$
- **else**
  - Choose $\mu^{k+1} > \mu^k$
  - $\lambda^{k+1} \leftarrow \lambda^k$

Stochastic augmented Lagrangian method

augmented Lagrangian

penalty

$$L(\alpha, \theta, s, \lambda, \mu) = F(\theta) + \lambda(H(\alpha, \theta) + s - \kappa) + \frac{\mu}{2}\|H(\alpha, \theta) + s - \kappa\|^2$$

multiplier

for $k = 1, \ldots, k_{\max}$ **do**
  $G^k \leftarrow \hat{\nabla} L(\alpha^k, \theta^k, s^k, \lambda^k, \mu^k)$ **implicit differentiation**
  $(\theta^{k+1}, \alpha^{k+1}, s^{k+1}) \leftarrow (\theta^k, \alpha^k, s^k) - tG^k$
  $s^{k+1} \leftarrow (s^{k+1})_+$
  if $\|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2 \leq \tau H_{\text{best}}$ **then**
    $\lambda^{k+1} \leftarrow \lambda^k + \mu^k(H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa)$
    $\mu^{k+1} \leftarrow \mu^k$
    $H_{\text{best}} \leftarrow \|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2$
  else
    Choose $\mu^{k+1} > \mu^k$
    $\lambda^{k+1} \leftarrow \lambda^k$

update primal variables

update multiplier
tighten tolerance

Stochastic augmented Lagrangian method

augmented Lagrangian

penalty

$$L(\alpha, \theta, s, \lambda, \mu) = F(\theta) + \lambda(H(\alpha,\theta) + s - \kappa) + \frac{\mu}{2}\|H(\alpha,\theta) + s - \kappa\|^2$$

multiplier

for $k = 1, \dots, k_{\max}$ **do**

$G^k \leftarrow \hat{\nabla} L(\alpha^k, \theta^k, s^k, \lambda^k, \mu^k)$ **implicit differentiation**

$(\theta^{k+1}, \alpha^{k+1}, s^{k+1}) \leftarrow (\theta^k, \alpha^k, s^k) - tG^k$

$s^{k+1} \leftarrow (s^{k+1})_+$

update primal variables

if $\|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2 \leq \tau H_{\text{best}}$ **then**

$\lambda^{k+1} \leftarrow \lambda^k + \mu^k (H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa)$

$\mu^{k+1} \leftarrow \mu^k$

$H_{\text{best}} \leftarrow \|H(\alpha^{k+1}, \theta^{k+1}) + s^{k+1} - \kappa\|_2$

update multiplier
tighten tolerance

else

Choose $\mu^{k+1} > \mu^k$

$\lambda^{k+1} \leftarrow \lambda^k$

update penalty

Stochastic gradients rely on KKT differentiation

family/context parameter

y $\longrightarrow$

optimization layer

argmin	$f(x, y)$
subject to	$g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)$

$\longrightarrow$ $x(\theta, y)$

parametric solution

Stochastic gradients rely on KKT differentiation

family/context parameter

$y \longrightarrow$

optimization layer

argmin $\quad f(x, y)$

subject to $\quad g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)$

parametric solution

$\longrightarrow x(\theta, y)$

Implicit differentiation

$\hat{\nabla}_\theta L$ *depends on jacobian* $J_x(\theta)$

Stochastic gradients rely on KKT differentiation

family/context parameter

$y \longrightarrow$

optimization layer

argmin	$f(x, y)$
subject to	$g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)$

parametric solution

$\longrightarrow x(\theta, y)$

Implicit differentiation

$\hat{\nabla}_\theta L$ *depends on jacobian* $J_x(\theta)$

(obtained by differentiating through the KKT optimality conditions)

“Differentiable Convex Optimization Layers”, A. Agrawal, B. Amos, S. Barratt, S. Boyd, S. Diamond, and J. Zico Kolter (2019)
“Differentiating Through a Conic Program”, A. Agrawal, S. Barratt, S. Boyd, E. Busseti, W. M. Moursi (2019)

Convergence analysis

(informal)

Convergence analysis

(informal)

Theorem (chain rule works)

if inner robust problem
- convex and conic
- has unique solution

$\longrightarrow$ $x(\theta, y)$ is path-differentiable with conservative jacobian $J_x(\theta) \neq \emptyset$

$\longrightarrow$ L is path-differentiable

Concepts of path-differentiability -> "Nonsmooth Implicit Differentiation for Machine Learning and Optimization", J. Bolte, T. Le, E. Pauwels, A. Silveti-Falls (2021)

Convergence analysis

(informal)

Theorem (chain rule works)

if inner robust problem
- convex and conic
- has unique solution

$\longrightarrow$ $x(\theta, y)$ is path-differentiable with conservative jacobian $J_x(\theta) \neq \emptyset$

$\longrightarrow$ L is path-differentiable

Theorem (locally optimal solutions)

if sequence of penalty parameters μ^k is bounded $\longrightarrow$ the algorithm converges almost surely to a feasible solution $(\alpha^\star, \theta^\star, s^\star)$

Concepts of path-differentiability -> "Nonsmooth Implicit Differentiation for Machine Learning and Optimization", J. Bolte, T. Le, E. Pauwels, A. Silveti-Falls (2021)

Convergence analysis

(informal)

Theorem (chain rule works)

if inner robust problem
- convex and conic
- has unique solution

→ $x(\theta, y)$ is path-differentiable with conservative jacobian $J_x(\theta) \neq \emptyset$

→ L is path-differentiable

Theorem (locally optimal solutions)

if sequence of penalty parameters μ^k is bounded → the algorithm converges almost surely to a feasible solution $(\alpha^\star, \theta^\star, s^\star)$

in addition, if $(\alpha^\star, \theta^\star, s^\star)$ satisfies:
- LICQ
- second-order optimality conditions

→ it is a locally optimal solution

Concepts of path-differentiability -> "Nonsmooth Implicit Differentiation for Machine Learning and Optimization", J. Bolte, T. Le, E. Pauwels, A. Silveti-Falls (2021)

Finite-sample probabilistic guarantees via threshold

optimality tolerance

target confidence

dimension of x

number of samples of u

Finite-sample probabilistic guarantees via threshold

optimality tolerance

target confidence

dimension of x

number of samples of u

threshold constraint

$$\mathbf{E}_w[h(\alpha^\star, \theta^\star, w)] \leq \kappa$$

Finite-sample probabilistic guarantees via threshold

optimality tolerance

target confidence

dimension of x

number of samples of u

threshold constraint

$$\mathbf{E}_w[h(\alpha^\star, \theta^\star, w)] \leq \kappa$$

Implies

Finite-sample probabilistic guarantee

$$\mathbf{P}^{N \times J}\left(\mathbf{P}_{(u,y)}(g(x, u, y) \leq 0) \geq 1 - \eta \quad \forall x \in \mathcal{X}\right) \geq 1 - \beta$$

Ingredients

- Tail bounds
- $\mathbf{CVaR} \geq \mathbf{VaR}$

Finite-sample probabilistic guarantees via threshold

optimality tolerance

target confidence

dimension of x

number of samples of u

threshold constraint

$$\mathbf{E}_w[h(\alpha^\star, \theta^\star, w)] \leq \kappa$$

Implies

Finite-sample probabilistic guarantee

$$\mathbf{P}^{N\times J}\left(\mathbf{P}_{(u,y)}(g(x, u, y) \leq 0) \geq 1 - \eta \quad \forall x \in \mathcal{X}\right) \geq 1 - \beta$$

$\Rightarrow$ it holds also for $x(\theta^\star, y)$

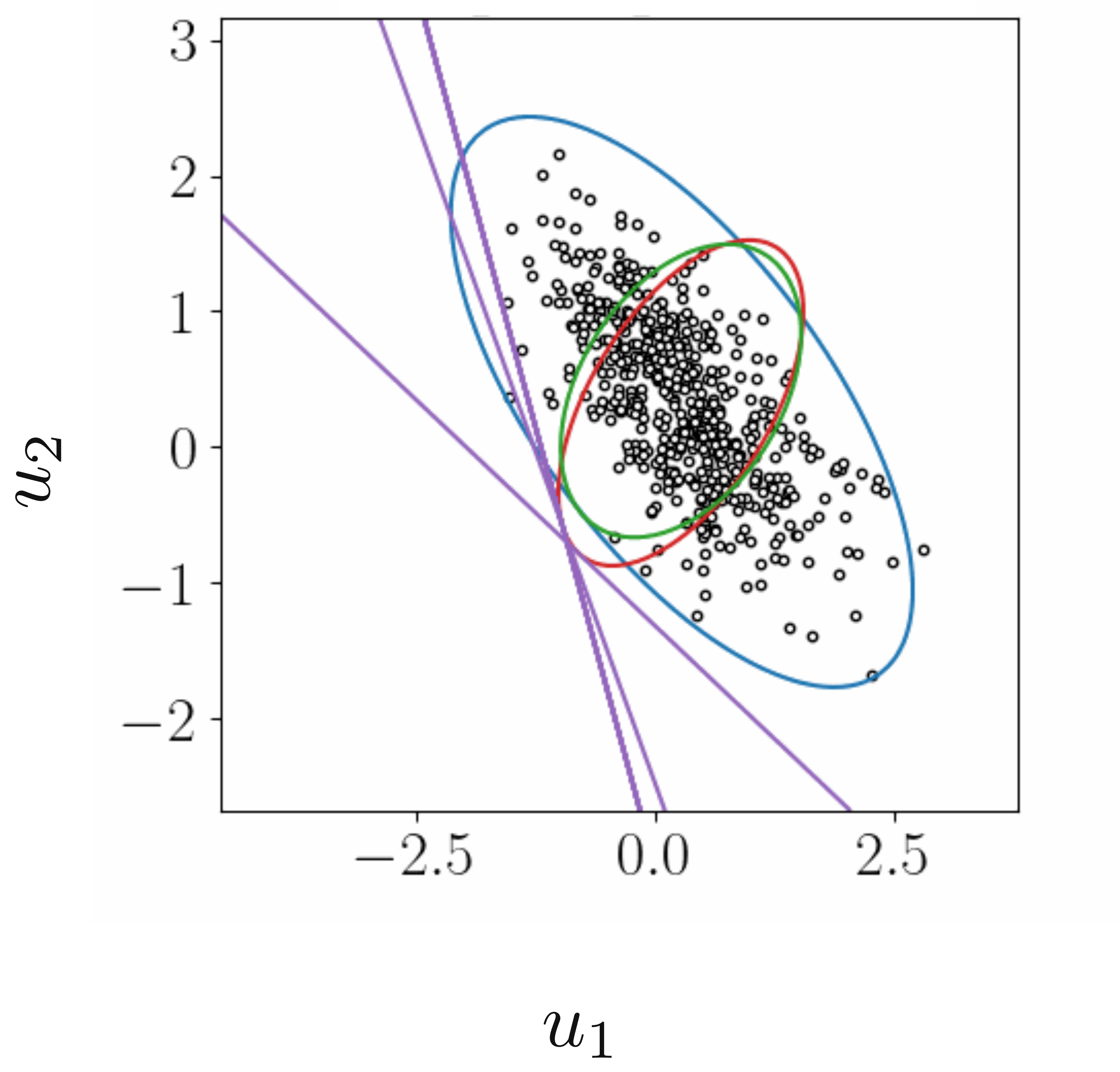
Ingredients

- Tail bounds
- $\mathbf{CVaR} \geq \mathbf{VaR}$

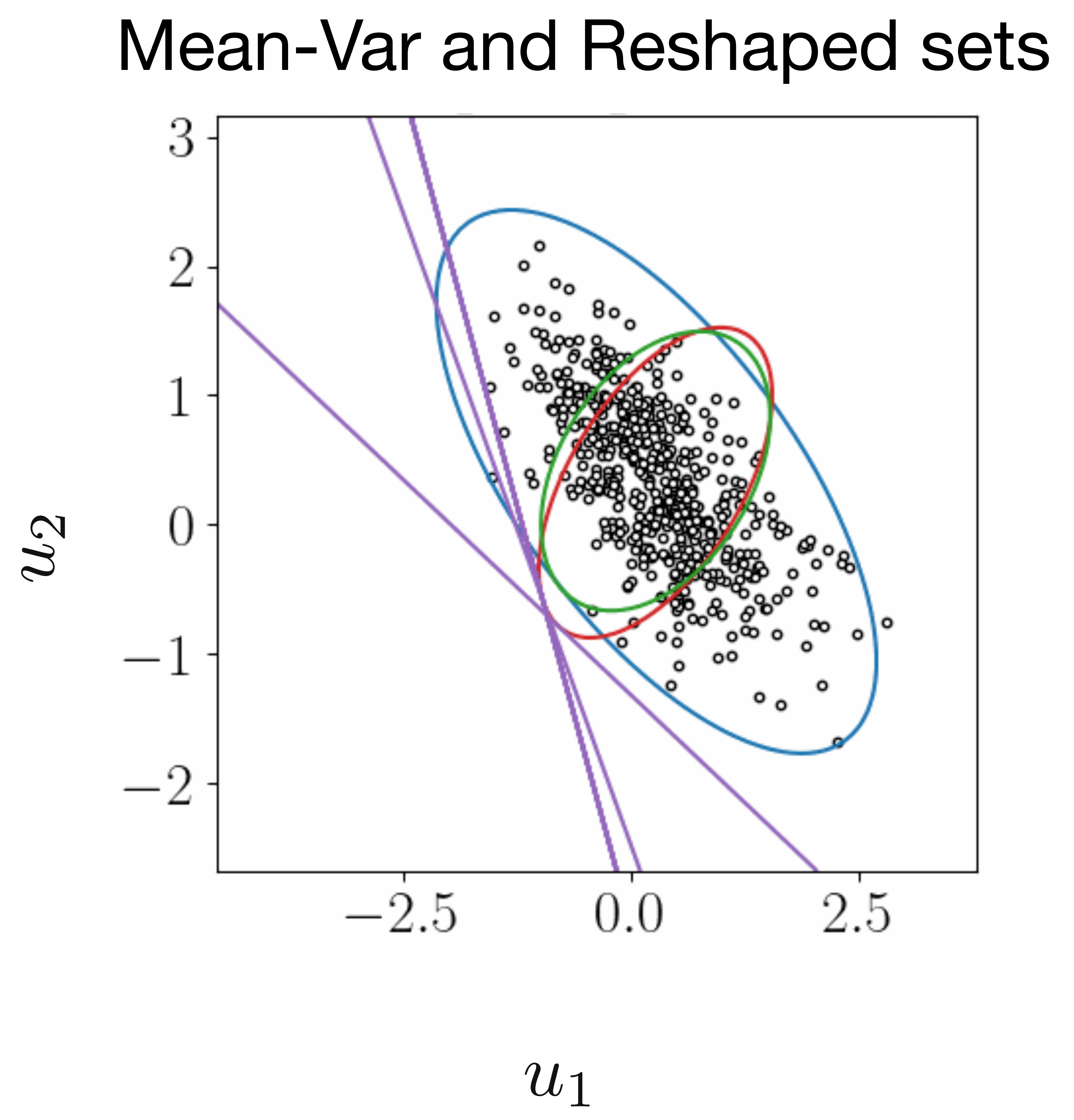
Numerical examples

Training the uncertainty set for the portfolio example

Mean-Var and Reshaped sets



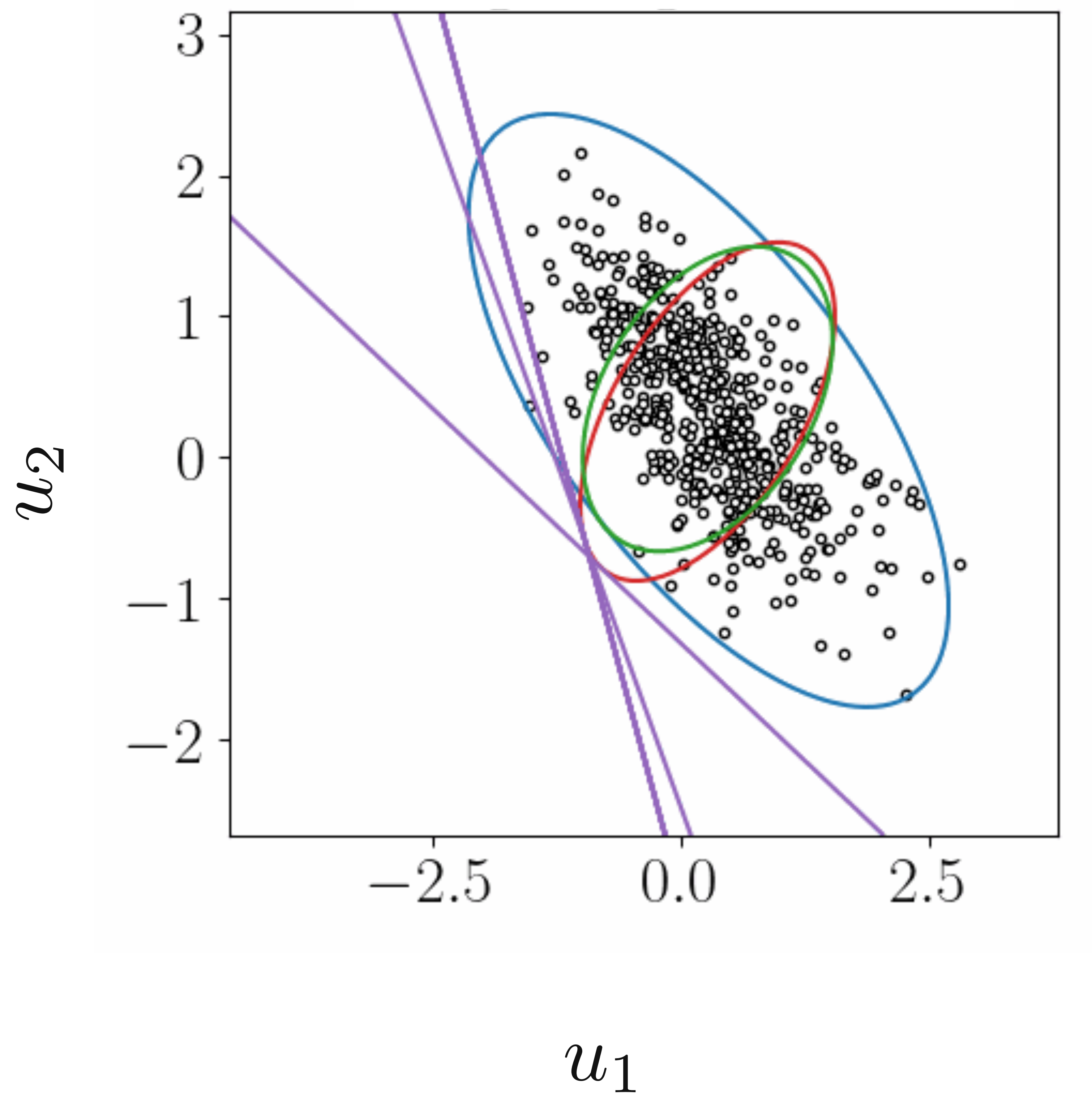
Training the uncertainty set for the portfolio example



Training the uncertainty set for the portfolio example

$$
\begin{array}{ll}
\text{minimize} & t + \lambda \|x - x^{\text{prev}}\|_1 \\
\text{subject to} & -u^T x \le t \quad \forall u \in \mathcal{U}(\theta) \\
& \mathbf{1}^T x = 1, \quad x \ge 0
\end{array}
$$

Mean-Var and Reshaped sets

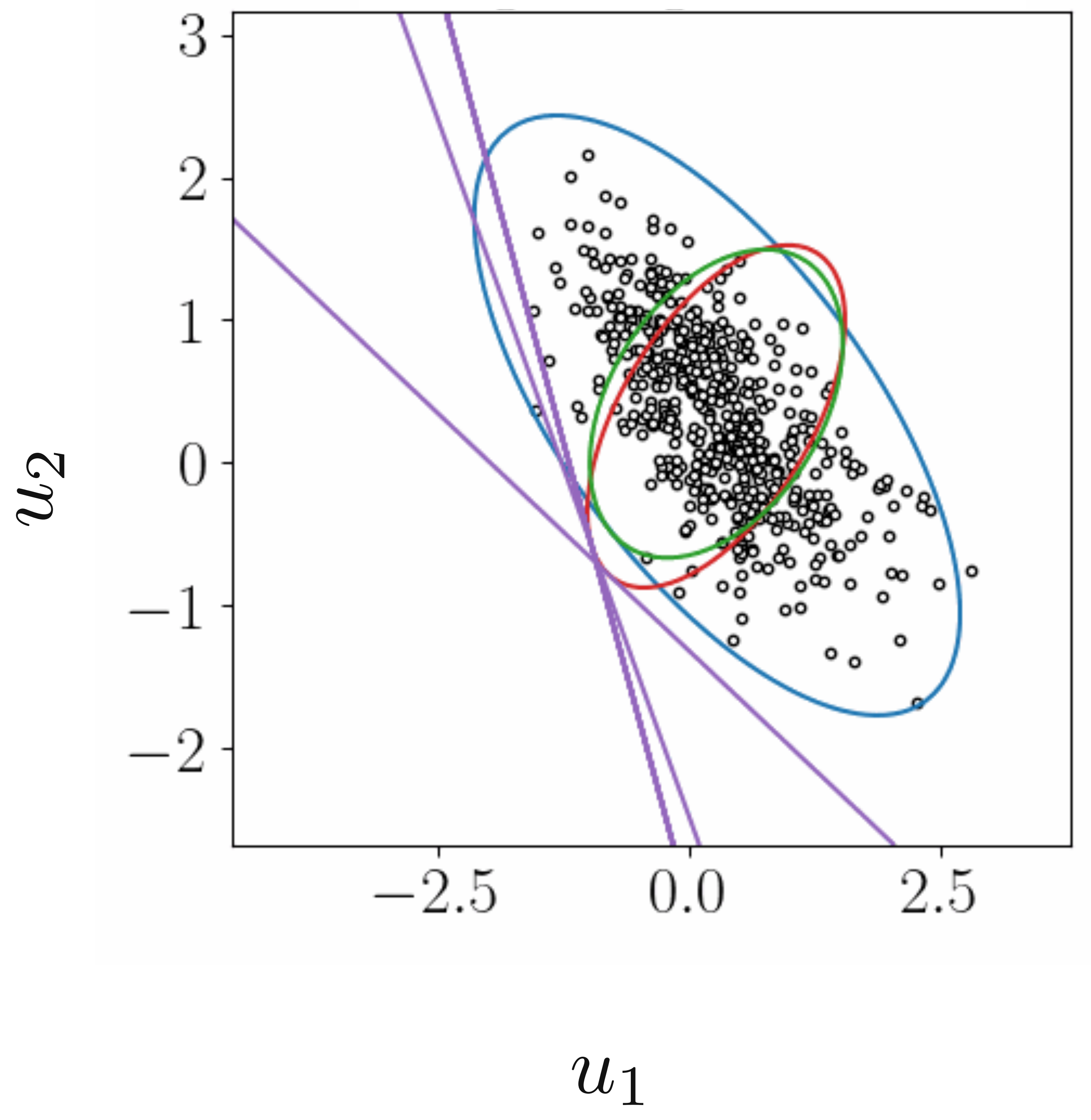


Training the uncertainty set for the portfolio example

$y = x^{\text{prev}}$
previous holdings

$$\begin{array}{ll} \text{minimize} & t + \lambda \|x - \boxed{x^{\text{prev}}}\|_1 \\ \text{subject to} & -u^T x \le t \quad \forall u \in \mathcal{U}(\theta) \\ & \mathbf{1}^T x = 1, \quad x \ge 0 \end{array}$$

Mean-Var and Reshaped sets

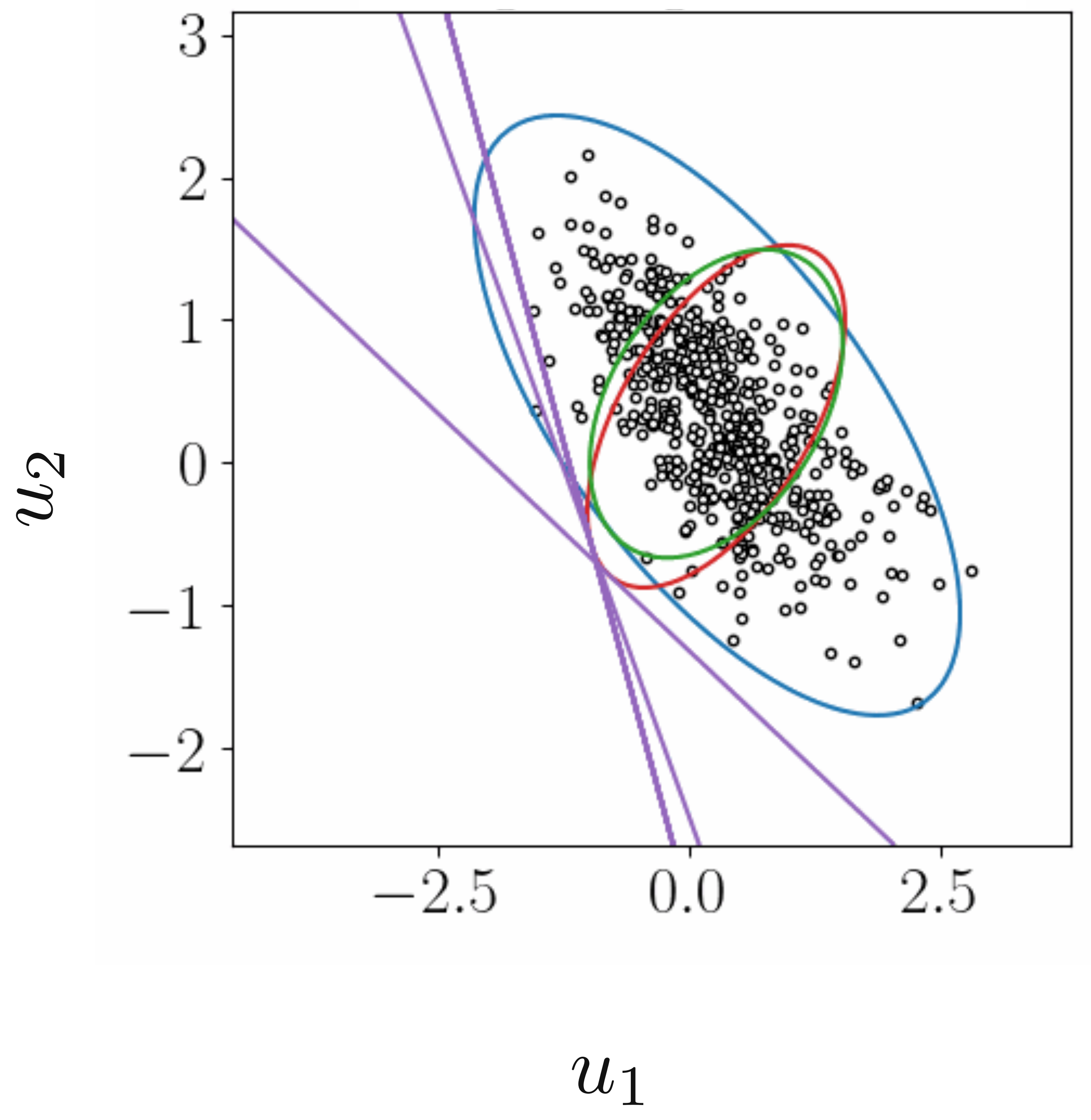


Training the uncertainty set for the portfolio example

$y = x^{\text{prev}}$
previous holdings

$$\begin{array}{ll} \text{minimize} & t + \lambda \|x - \boxed{x^{\text{prev}}}\|_1 \\ \text{subject to} & -u^T x \leq t \quad \forall u \in \mathcal{U}(\theta) \\ & \mathbf{1}^T x = 1, \quad x \geq 0 \end{array}$$

Mean-Var and Reshaped sets



constraint level curves
$-u^T x(\theta, y) - t(\theta, y) = 0$

Inventory management as two-stage robust optimization

n retail points

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

$$
\begin{array}{ll}
\text{minimize} & \tau \\
\text{subject to} & (t+h)^T s - y^T w(u) \le \tau, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \le s, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \le \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) \\
& \mathbf{1}^T s = C \\
& 0 \le s \le c
\end{array}
$$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

$$
\begin{array}{ll}
\text{minimize} & \tau \\
\text{subject to} & (t+h)^T s - y^T w(u) \le \tau, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \le s, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \le \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) \\
& \mathbf{1}^T s = C \\
& 0 \le s \le c
\end{array}
$$

"Pareto Efficiency in Robust Optimization" D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

sale prices

$$
\begin{array}{ll}
\text{minimize} & \tau \\
\text{subject to} & (t+h)^T s - y^T w(u) \leq \tau, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \leq s, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \leq \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) \\
& \mathbf{1}^T s = C \\
& 0 \leq s \leq c
\end{array}
$$

"Pareto Efficiency in Robust Optimization" D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

sale prices

$$
\begin{array}{ll}
\text{minimize} & \tau \\
\text{subject to} & (t+h)^T s - y^T w(u) \leq \tau, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \leq s, \quad \forall u \in \mathcal{U}(\theta) \\
& w(u) \leq \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) \\
& \mathbf{1}^T s = C \\
& 0 \leq s \leq c
\end{array}
$$

(minimize worst-case costs)

"Pareto Efficiency in Robust Optimization" D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

sale prices

$$
\begin{array}{lll}
\text{minimize} & \tau & \\
\text{subject to} & (t+h)^T s - y^T w(u) \leq \tau, \quad \forall u \in \mathcal{U}(\theta) & \textit{(minimize worst-case costs)} \\
& w(u) \leq s, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than stocked items)} \\
& w(u) \leq \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) & \\
& \mathbf{1}^T s = C & \\
& 0 \leq s \leq c &
\end{array}
$$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$ ← uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

sale prices

$$
\begin{array}{lll}
\text{minimize} & \tau & \\
\text{subject to} & (t+h)^T s - y^T w(u) \leq \tau, \quad \forall u \in \mathcal{U}(\theta) & \textit{(minimize worst-case costs)} \\
& w(u) \leq s, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than stocked items)} \\
& w(u) \leq \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than demand)} \\
& \mathbf{1}^T s = C & \\
& 0 \leq s \leq c &
\end{array}
$$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

sale prices

$$
\begin{array}{lll}
\text{minimize} & \tau & \\
\text{subject to} & (t+h)^T s - y^T w(u) \leq \tau, \quad \forall u \in \mathcal{U}(\theta) & \textit{(minimize worst-case costs)} \\
& w(u) \leq s, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than stocked items)} \\
& w(u) \leq \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than demand)} \\
& \mathbf{1}^T s = C & \textit{(total units of product available)} \\
& 0 \leq s \leq c &
\end{array}
$$

Inventory management as two-stage robust optimization

n retail points

uncertain demand $d = \bar{d} + Qu$

uncertain market factors $u \in \mathbf{R}^m$

two-stage decisions

stocking decisions $s \in \mathbf{R}^n$

sales decisions $w(d) \in \mathbf{R}^n$

transportation and holding costs

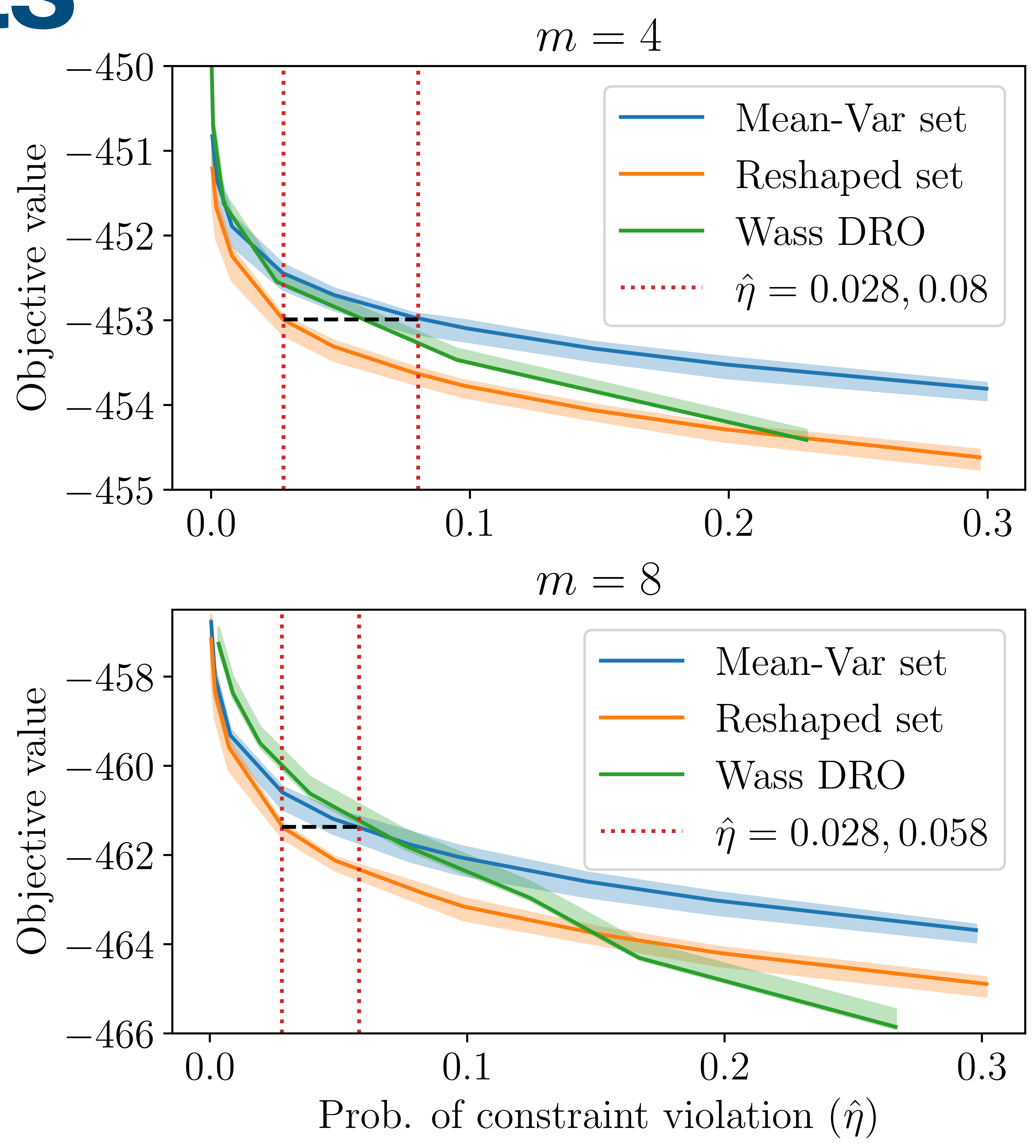
sale prices

$$
\begin{array}{lll}
\text{minimize} & \tau & \\
\text{subject to} & (t+h)^T s - y^T w(u) \le \tau, \quad \forall u \in \mathcal{U}(\theta) & \textit{(minimize worst-case costs)} \\
& w(u) \le s, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than stocked items)} \\
& w(u) \le \bar{d} + Qu, \quad \forall u \in \mathcal{U}(\theta) & \textit{(sell less than demand)} \\
& \mathbf{1}^T s = C & \textit{(total units of product available)} \\
& 0 \le s \le c & \textit{(maximum stock volume)}
\end{array}
$$

“Pareto Efficiency in Robust Optimization” D. A. Iancu, N. Trichakis (2014)

Inventory management results

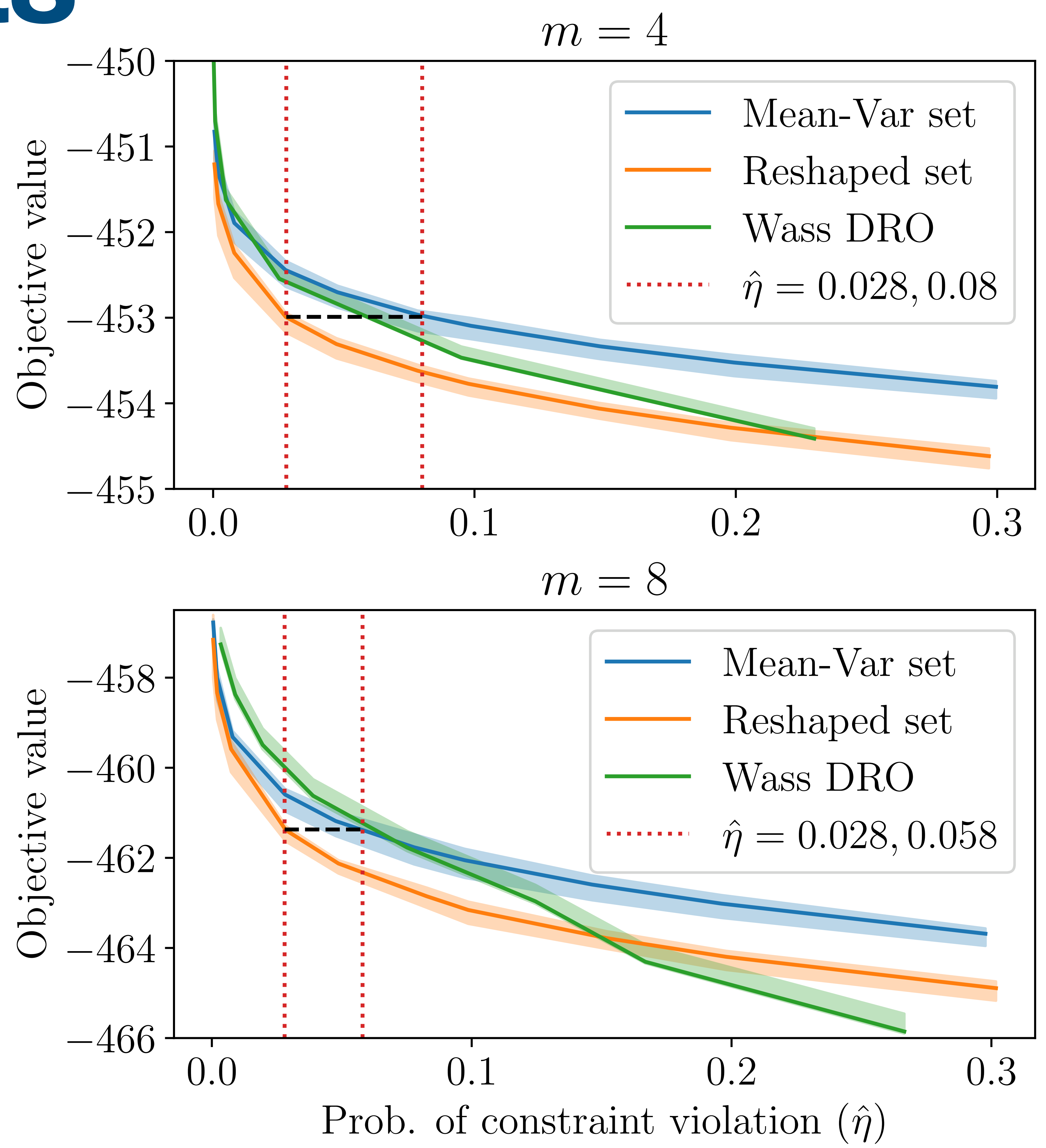
u size				$m = 4$			
Method	LROPT	LRO-T$_{0.03}$	LRO-T$_{0.05}$	MV-RO$_{0.03}$	MV-RO$_{0.05}$	W-DRO$_{0.03}$	W-DRO$_{0.05}$
Obj.	-451.67	-452.99	-453.31	-452.44	-452.70	-452.54	-452.77
$\hat{\eta}$	0.002	0.0278	0.0471	0.0277	0.0476	0.0252	0.0451
$\hat{\beta}$	0	0.2	0.3	0.3	0.25	0	0.1
t	0.00203	0.00212	0.00206	0.00201	0.00209	0.336	0.315
u size				$m = 8$			
Obj.	-459.49	-461.27	-462.06	-460.62	-461.18	-459.49	-460.62
$\hat{\eta}$	0.0068	0.0257	0.0458	0.0257	0.0477	0.0195	0.0390
$\hat{\beta}$	0	0	0	0	0.06	0.2	0.2
t	0.00623	0.00630	0.00613	0.00634	0.00619	0.910	0.932



Inventory management results

u size				$m=4$			
Method	LROPT	LRO-T$_{0.03}$	LRO-T$_{0.05}$	MV-RO$_{0.03}$	MV-RO$_{0.05}$	W-DRO$_{0.03}$	W-DRO$_{0.05}$
Obj.	-451.67	-452.99	-453.31	-452.44	-452.70	-452.54	-452.77
$\hat{\eta}$	0.002	0.0278	0.0471	0.0277	0.0476	0.0252	0.0451
$\hat{\beta}$	0	0.2	0.3	0.3	0.25	0	0.1
t	0.00203	0.00212	0.00206	0.00201	0.00209	0.336	0.315
u size				$m=8$			
Obj.	-459.49	-461.27	-462.06	-460.62	-461.18	-459.49	-460.62
$\hat{\eta}$	0.0068	0.0257	0.0458	0.0257	0.0477	0.0195	0.0390
$\hat{\beta}$	0	0	0	0	0.06	0.2	0.2
t	0.00623	0.00630	0.00613	0.00634	0.00619	0.910	0.932

better trade-off between *objective* and *probability of constraint violation*

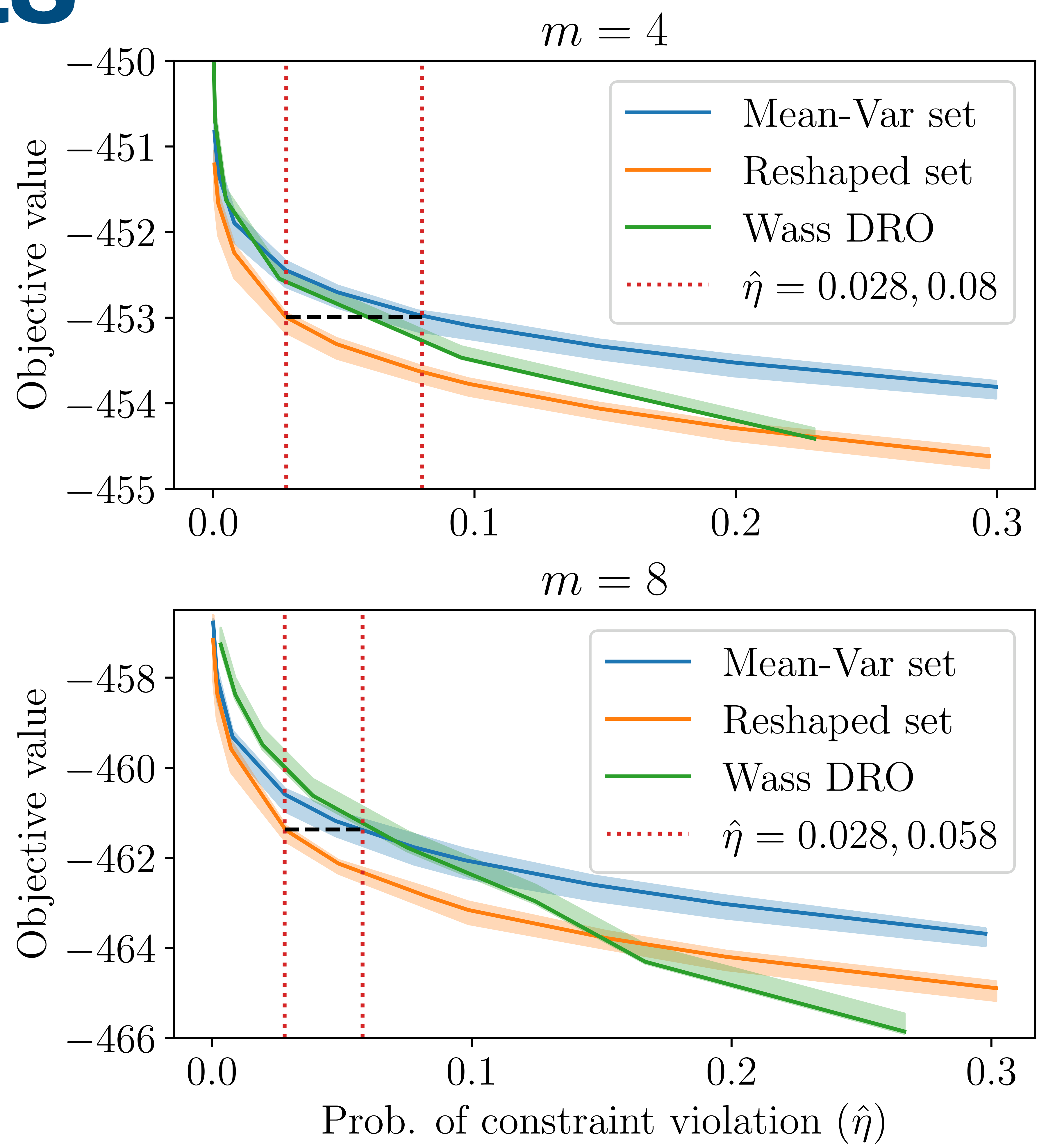


Inventory management results

u size				$m = 4$			
Method	LROPT	LRO-$T_{0.03}$	LRO-$T_{0.05}$	MV-$RO_{0.03}$	MV-$RO_{0.05}$	W-$DRO_{0.03}$	W-$DRO_{0.05}$
Obj.	-451.67	-452.99	-453.31	-452.44	-452.70	-452.54	-452.77
$\hat{\eta}$	0.002	0.0278	0.0471	0.0277	0.0476	0.0252	0.0451
$\hat{\beta}$	0	0.2	0.3	0.3	0.25	0	0.1
t	0.00203	0.00212	0.00206	0.00201	0.00209	0.336	0.315
u size				$m = 8$			
Obj.	-459.49	-461.27	-462.06	-460.62	-461.18	-459.49	-460.62
$\hat{\eta}$	0.0068	0.0257	0.0458	0.0257	0.0477	0.0195	0.0390
$\hat{\beta}$	0	0	0	0	0.06	0.2	0.2
t	0.00623	0.00630	0.00613	0.00634	0.00619	0.910	0.932

better trade-off between *objective* and *probability of constraint violation*

faster computation times than Wassertstein DRO



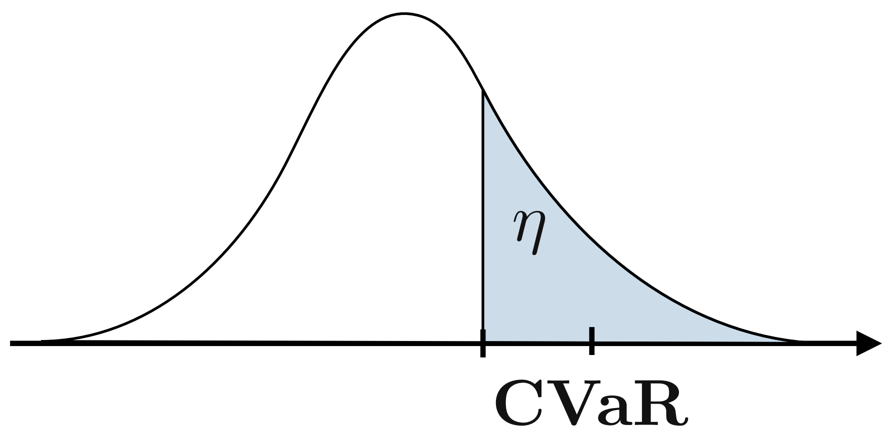
Learning decision-focused uncertainty sets for robust optimization

- Optimize **shape and size** of uncertainty sets
- **Bi-level optimization** formulation
  - CVaR constraint
  - Differentiable optimization to compute derivatives
  - Probabilistic guarantees
- **Improvements over RO and DRO formulations**

Learning decision-focused uncertainty sets for robust optimization

- Optimize **shape and size** of uncertainty sets

- **Bi-level optimization** formulation
  - CVaR constraint
  - Differentiable optimization to compute derivatives
  - Probabilistic guarantees



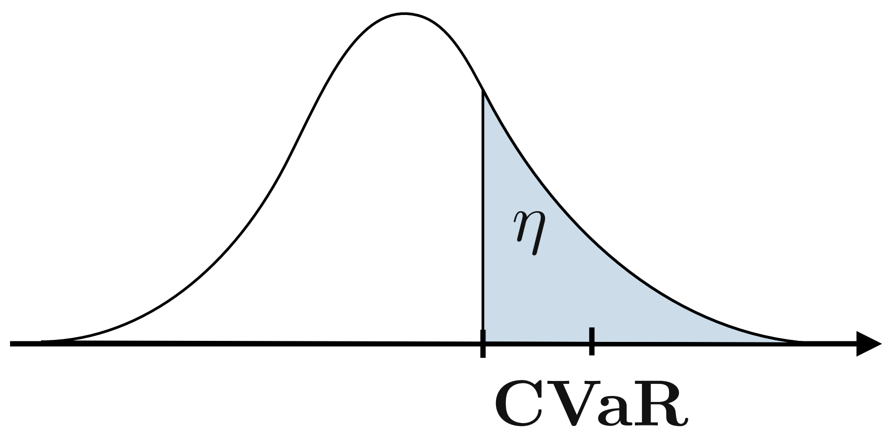
- **Improvements over RO and DRO formulations**

Learning decision-focused uncertainty sets for robust optimization

- Optimize **shape and size** of uncertainty sets

- **Bi-level optimization** formulation
  - CVaR constraint
  - Differentiable optimization to compute derivatives
  - Probabilistic guarantees

- **Improvements over RO and DRO formulations**



optimization layer

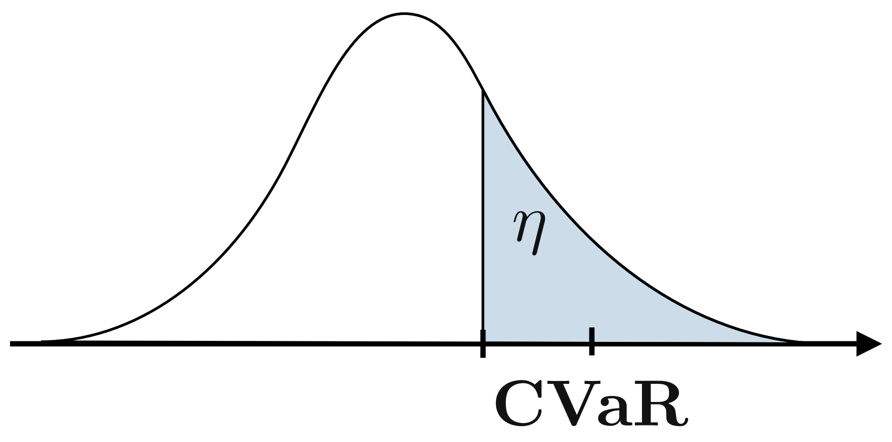
$$y \longrightarrow \begin{array}{ll} \text{argmin} & f(x,y) \\ \text{subject to} & g(x,u,y) \leq 0 \quad \forall u \in \mathcal{U}(\theta) \end{array} \longrightarrow x(\theta, y)$$

Learning decision-focused uncertainty sets for robust optimization

- Optimize **shape and size** of uncertainty sets

- **Bi-level optimization** formulation
  - CVaR constraint
  - Differentiable optimization to compute derivatives
  - Probabilistic guarantees

- **Improvements over RO and DRO formulations**



optimization layer

$y \longrightarrow$

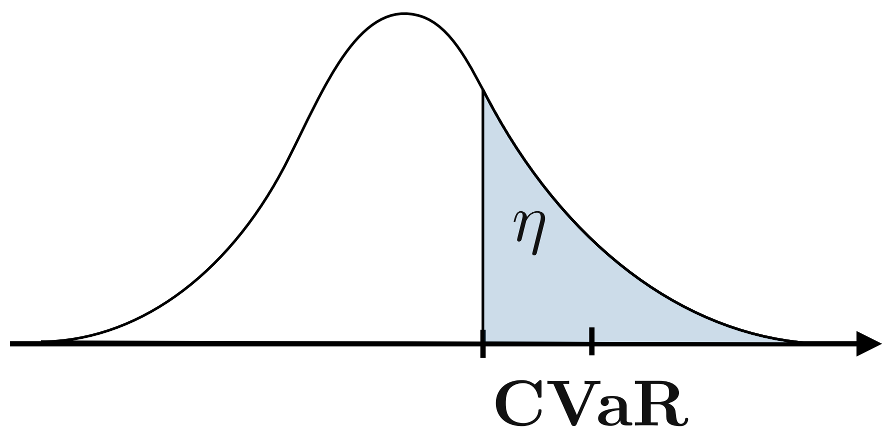
argmin $\quad f(x, y)$

subject to $\quad g(x, u, y) \leq 0 \quad \forall u \in \mathcal{U}(\theta)$

$\longrightarrow x(\theta, y)$

Learning decision-focused uncertainty sets for robust optimization

- Optimize **shape and size** of uncertainty sets
- **Bi-level optimization** formulation
  - CVaR constraint
  - Differentiable optimization to compute derivatives
  - Probabilistic guarantees
- **Improvements over RO and DRO formulations**



optimization layer

$y \longrightarrow$

argmin	$f(x, y)$	
subject to	$g(x, u, y) \leq 0$	$\forall u \in \mathcal{U}(\theta)$

$\longrightarrow x(\theta, y)$

https://github.com/stellatogrp/lropt

Learning for Robust Optimization
I. Wang, C. Becker, B. Van Parys, and B. Stellato
arxiv.org: 2305.19225, 2023

⚠️ will be updated soon!

LROPT software package (WIP)

It can be *hard to dualize* robust optimization problems

…not to mention *finding the right uncertainty set!*

LROPT software package (WIP)

It can be ***hard to dualize***
robust optimization problems

…not to mention ***finding the right uncertainty set!***

LROPT package

github.com/stellatogrp/lropt

1. Easily formulate and dualize robust optimization problems

2. Automatically tune uncertainty sets (using cvxpylayers)

LROPT software package (WIP)

It can be *hard to dualize* robust optimization problems

…not to mention *finding the right uncertainty set!*

LROPT package

github.com/stellatogrp/lropt

1. Easily formulate and dualize robust optimization problems
2. Automatically tune uncertainty sets (using cvxpylayers)

$$
\begin{array}{ll}
\text{minimize} & x^T P x + y^T x \\
\text{subject to} & (a + Bu)^T x \leq d, \quad \forall u \in \mathcal{U}
\end{array}
$$

$$\mathcal{U} = \{u = b + Az \mid \|z\|_2 \leq 1\}$$

```
unc_set = lropt.Ellipsoidal(u_data)
u = lropt.UncertainParameter(n,
                             uncertainty_set=unc_set)
x = cp.Variable(n)
y = cp.Parameter(n)
constraints = [(a + B@u) @ x <= d]
objective = cp.Minimize(cp.quad_form(P, x) + y @ x)
problem = lropt.RobustProblem(objective,
                              constraints)
problem.train()
```

Jun 27–28, 2024, Princeton University

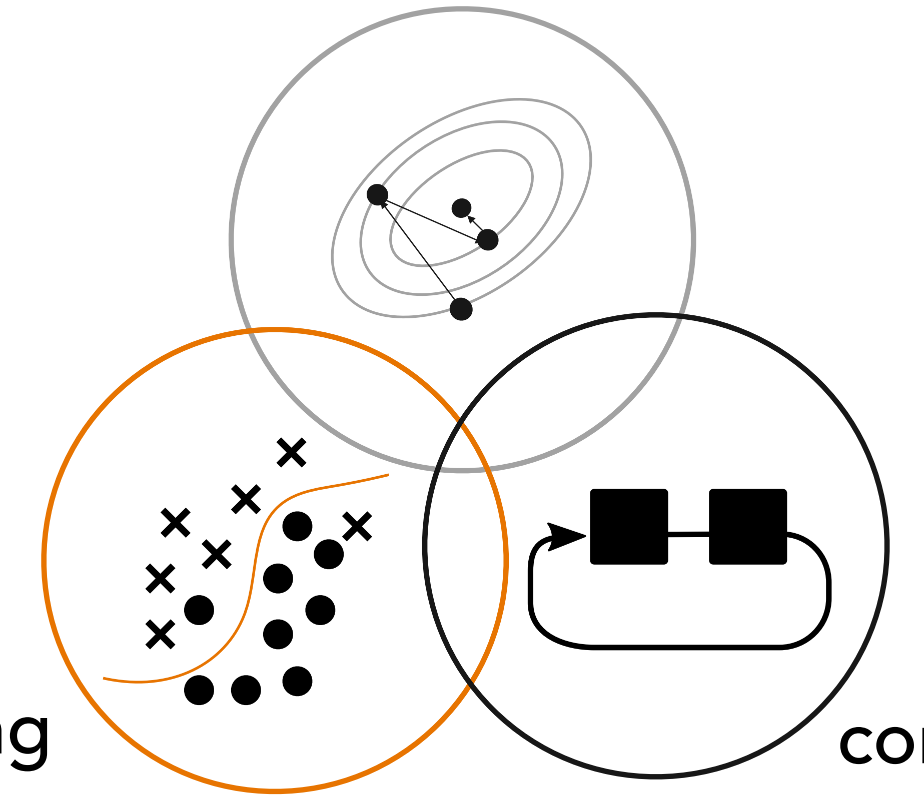
PRINCETON UNIVERSITY

Princeton Workshop on

Optimization, Learning, and Control

The Princeton Workshop on Optimization, Learning, and Control is a single-track workshop highlighting the latest research advances across these disciplines. Its main goal is to foster new interactions and lay the groundwork for new collaborations. The workshop will include a poster session for junior researchers.

optimization



learning

control

Important dates

- **March 3**: Deadline for poster abstract submission and travel support application
- **May 1**: Registration opens
- **June 1**: Registration deadline

Contacts

- **Website**: stellato.io/olc
- **Organizer**: Bartolomeo Stellato, Princeton University — stellato.io
- **Questions**: olc24@princeton.edu

Supported by

National Science Foundation

Center for Statistics and Machine Learning

ORFE Operations Research & Financial Engineering

GUROBI OPTIMIZATION

School of Engineering and Applied Science PRINCETON

IBM

Confirmed speakers

Shipra Agrawal
Columbia University

Anuradha Annaswamy
MIT

Francesco Borrelli
UC Berkeley

Sarah Dean
Cornell University

Paul Goulart
University of Oxford

Elad Hazan
Princeton University

Andrea Lodi
Cornell Tech

Robert Luce
Gurobi Optimization

Melanie Zeilinger
ETH Zurich

Anirudha Majumdar
Princeton University

Conclusion

Machine Learning tools
can help us
formulate optimization problems

Conclusion

Machine Learning tools
can help us
formulate optimization problems

↓

We should think
building robust optimization models
as an (automated)
training/validation procedure

⚙️ tune *train* → ✅ validate *certify* → 🚀 deploy

Conclusion

Machine Learning tools
can help us
formulate optimization problems

↓

We should think
building robust optimization models
as an (automated)
training/validation procedure

⚙️ tune *train* → ☑️ validate *certify* → 🚀 deploy

@b_stellato bstellato@princeton.edu stellato.io