Contextual Scenario Reduction for Two-Stage Stochastic Programming

David Islip [1] Sanghyeon Bae [2] Roy H. Kwon [1] Woo Chang Kim [2]

[1]Department of Mechanical and Industrial Engineering, University of Toronto

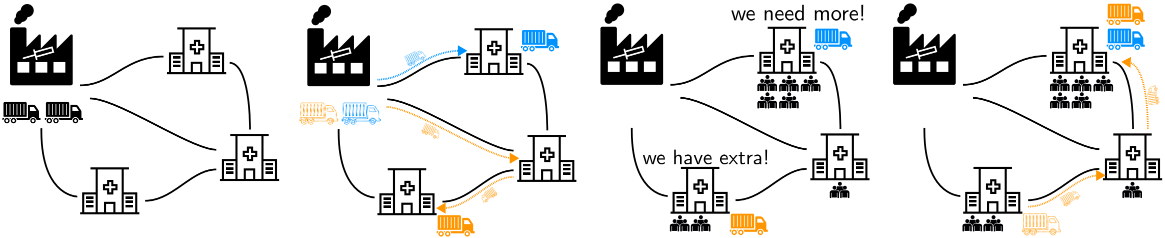
[2]Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology (KAIST)

October 20, 2023

Motivation

Two-Stage Stochastic Programming

- Stochastic programming is a decision-making framework that has succeeded in many areas, including finance, healthcare, and logistics.
- Two-stage stochastic programming makes here-and-now decisions while accounting for future uncertainties and available recourse actions



Motivation

Two-Stage Stochastic Programming

- The decision maker selects a first-stage decision $\boldsymbol{y}$ by solving:

$$\min_{\boldsymbol{y}} h(\boldsymbol{y}) + \mathbb{Q}(\boldsymbol{y}) \qquad \text{s.t.} \quad \boldsymbol{y} \in \mathcal{Y}, \quad \boldsymbol{y} \in \mathbb{R}^{s_1} \tag{2SP}$$

- where h is a function modeling the cost of the stage-I decision, $\mathcal{Y}$ is the feasible set for first-stage decisions, $\omega \in \Omega$ represents the uncertainty distributed according to the probability measure $\mathbb{P}$,
- $\mathbb{Q}(\boldsymbol{y}) = \mathbb{E}_{\mathbb{P}}[Q(\boldsymbol{y}, \omega)]$ is the expected recourse cost with $Q(\boldsymbol{y}, \omega)$ denoting the recourse cost of the first-stage decision $\boldsymbol{y}$ when uncertainty ω is realized:

$$Q(\boldsymbol{y}, \omega) = \min_{\boldsymbol{z}} g(\boldsymbol{z}, \omega) \quad \text{s.t.} \quad \boldsymbol{z} \in \mathcal{Z}(\boldsymbol{y}, \omega), \quad \boldsymbol{z} \in \mathbb{R}^{s_2} \tag{Stage II}$$

- where g models the recourse cost for uncertainty ω in sample space $\Omega \subseteq \mathbb{R}^d$, $\mathcal{Z}(\boldsymbol{y}, \omega)$ is the feasible set of recourse actions given the first-stage decision $\boldsymbol{y}$ and uncertainty ω

Motivation

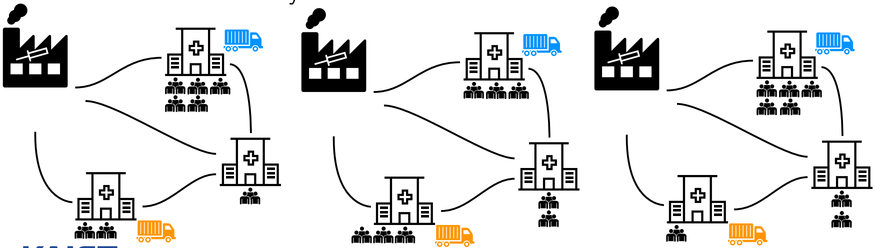
Scenario Reduction

- 2SP is intractable due to multi-dimensional integrals or exponentially many scenarios. Often, it requires considering a large finite subset Ξ of possible outcomes, and solving the 2SP defined on Ξ:

$$\min_{\boldsymbol{y}\in\mathcal{Y}\cap\mathbb{R}^{s_1}} \quad h(\boldsymbol{y}) + \frac{1}{|\Xi|}\sum_{j=1}^{|\Xi|} \min_{\boldsymbol{z}\in\mathcal{Z}(\boldsymbol{y},\xi^{(j)})} g(\boldsymbol{z},\xi^{(j)}) \tag{2SP-SAA}$$

- *scenario reduction* replaces Ξ with a set of K scenarios $\zeta_{1..K}$ such that $K << |\Xi|$ while maintaining solutions that perform well when evaluated on Ξ (Dupačová et al., 2003)

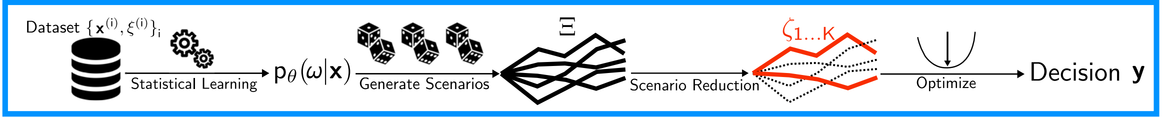
Many Possible Demand Scenarios



UNIVERSITY OF TORONTO KAIST

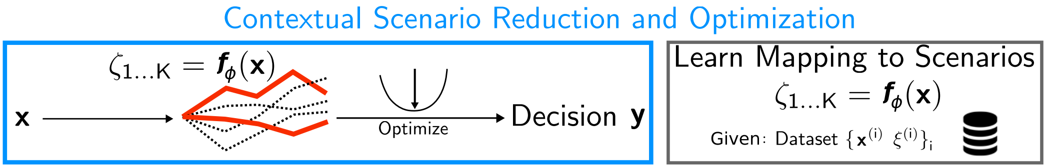
- Distribution of outcomes often depends on contextual information $\boldsymbol{x}$ known at decision time
- Dataset of context-scenario pairs $\tilde{\mathcal{D}} = \{(\boldsymbol{x}^{(i)}, \xi^{(i)})\}$ used to estimate conditional distribution $p_\theta(\omega|\boldsymbol{x})$

Optimization with Scenario Reduction



Proposed Approach
Overview

- Map contextual information $\boldsymbol{x}$ to surrogate scenario set $\zeta_{1..K}$ via some parametric mapping $\boldsymbol{f}_\phi$



- **How can we use a dataset of context-scenario pairs $\tilde{\mathcal{D}}$ to estimate $\boldsymbol{f}_\phi$ so that the $\zeta_{1..K}$ predicted by $\boldsymbol{f}_\phi(x)$ yield high quality first-stage decisions**

Proposed Approach

Method # 1: Distributional Contextual Scenario Reduction

- $\mathcal{D} = \{(\boldsymbol{x}^{(i)}, \Xi^{(i)})\}$ denote a large dataset of context and observed distributions conditional on the context

$$\min_{\phi} \mathcal{L}_{\text{distribution}}(\phi) := \frac{1}{|\mathcal{D}|} \sum_{i=1}^{N} d(\mathbb{P}(\phi, \boldsymbol{x}^{(i)}), \mathbb{Q}(\Xi^{(i)})) \tag{DCSR}$$

- where $\mathbb{P}(\phi, \boldsymbol{x}) = \frac{1}{K} \sum_{\zeta \in \boldsymbol{f}_{\phi}(\boldsymbol{x})}^{K} \delta_{\zeta}$ and $\mathbb{Q}(\Xi) = \frac{1}{|\Xi|} \sum_{\xi \in \Xi} \delta_{\xi}$ are the empirical measures associated with the sets $\boldsymbol{f}_{\phi}(\boldsymbol{x})$ and Ξ respectively and d is a distance metric between the distributions
- Stability bounds for 2SP (Römisch, 2003; Rachev & Römisch, 2002) $\implies$ consider integral probability metrics: $d_{\mathcal{F}}(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} |\mathbb{E}_{\mathbb{P}}[f(\omega)] - \mathbb{E}_{\mathbb{Q}}[f(\omega)]|$
- where $\mathcal{F}$ is a class of real-valued bounded measurable functions that correspond to difference distance (ex. 1-Wasserstein corresponds to 1-Lipshitz functions)

Proposed Approach

Method # 1: Distributional Contextual Scenario Reduction

- Maximum mean discrepancy (MMD), corresponds to $\mathcal{F} = \{f \in \mathcal{H} : \|f\|_\infty \leq 1\}$ where $\mathcal{H}$ denotes a Reproducing Kernel Hilbert Space (RKHS) with associated kernel $k : \Omega \times \Omega \to \mathbb{R}$ (Gretton et al., 2012)

- Sample complexity, computational complexity, ease of implementation, and sampling structure motivate choosing MMD as a distance measure

- let the dataset formed by the product of the contexts and associated observed distributions be denoted by $\tilde{\mathcal{D}} = \{(\boldsymbol{x}, \xi) \mid \xi \in \Xi,\ (\boldsymbol{x}, \Xi) \in \mathcal{D}\}$ then:

$$\mathcal{L}_{\text{MMD}}(\phi) := \frac{1}{|\tilde{\mathcal{D}}|} \sum_{j=1}^{|\tilde{\mathcal{D}}|} \Big(-\frac{2}{K} \sum_{i=1}^{K} k(\xi^{(j)}, \boldsymbol{f}_\phi^{(i)}(\boldsymbol{x}^{(j)})) + \frac{1}{K^2} \sum_{i=1}^{K} \sum_{i'=1}^{K} k(\boldsymbol{f}_\phi^{(i)}(\boldsymbol{x}^{(j)}), \boldsymbol{f}_\phi^{(i')}(\boldsymbol{x}^{(j)})) \Big) \qquad \text{(MMD Loss)}$$

- Sampling from the joint distribution of $(\boldsymbol{x}, \omega)$ is all that is required!

- The goal of scenario reduction is to select the surrogate scenarios $\zeta_{1\ldots K}$ so that the 2SP-SAA solution defined by $\zeta_{1\ldots K}$ performs well on the large set of scenarios Ξ
- This corresponds to the bi-level problem:

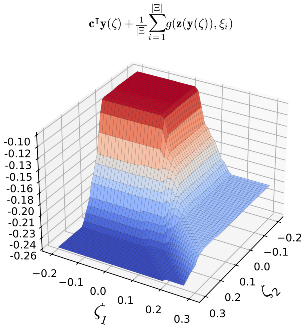
$$
\begin{aligned}
\min_{\zeta^{(1)},\ldots,\zeta^{(K)}} \quad & h(\boldsymbol{y}(\zeta_{1\ldots K})) + \frac{1}{|\Xi|}\sum_{j=1}^{|\Xi|} g(\boldsymbol{z}(\boldsymbol{y}(\zeta_{1\ldots K}), \xi^{(j)}), \xi^{(j)}) \\
\text{s.t.} \quad & \boldsymbol{z}(\boldsymbol{y}(\zeta_{1\ldots K}), \xi) \in \operatorname*{argmin}_{z \in \mathcal{Z}(\boldsymbol{y}(\zeta_{1\ldots K}),\xi)} g(\boldsymbol{z}, \xi) \quad \forall \xi \in \Xi \qquad \text{(SP)} \\
& \boldsymbol{y}(\zeta_{1\ldots K}) \in \operatorname*{argmin}_{\boldsymbol{y},\boldsymbol{z}_1,\ldots \boldsymbol{z}_K} h(\boldsymbol{y}) + \frac{1}{K}\sum_{i=1}^{K} g(\boldsymbol{z}_i, \zeta_i) \qquad (\zeta\text{-SAA}) \\
& \qquad \boldsymbol{y} \in \mathcal{Y},\ \boldsymbol{z}_i \in \mathcal{Z}(\boldsymbol{y}, \zeta)
\end{aligned}
\qquad \text{(BI-SAA)}
$$

UNIVERSITY OF TORONTO KAIST

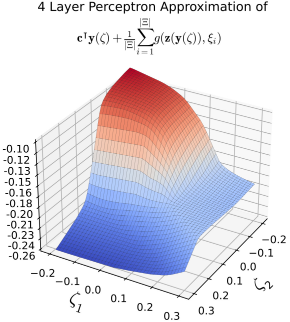
Proposed Approach

Bilevel Problem's Problems

- The bi-level problem can be heuristically solved via gradient descent: $\zeta \leftarrow \zeta - \eta\, \partial \bar{l}_\Xi / \partial \zeta$
- where the upper-level cost is given by $\bar{l}_\Xi(\zeta_{1\ldots K}) = h(\boldsymbol{y}(\zeta_{1\ldots K})) + \frac{1}{|\Xi|}\sum_{j=1}^{|\Xi|} g(\boldsymbol{z}(\boldsymbol{y}(\zeta_{1\ldots K}), \xi^{(j)}), \xi^{(j)})$



(a) The loss $\bar{l}_\Xi(\zeta)$ plotted against two components of the surrogate scenario ζ. The gradients are sparse.



(b) Approximating the loss surface with a neural network has a smoothing effect (Grigas, Qi, et al., 2021).

- The problem-driven approach aims to solve:

$$\min_{\phi} \mathcal{L}_{\text{task}}(\phi) := \frac{1}{|\mathcal{D}|} \sum_{i=1}^{N} \bar{l}_{\Xi^{(i)}}(\boldsymbol{f}_{\phi}(\boldsymbol{x}^{(i)})) \tag{PCSR}$$

- which is amenable to sampling context-scenario pairs since:

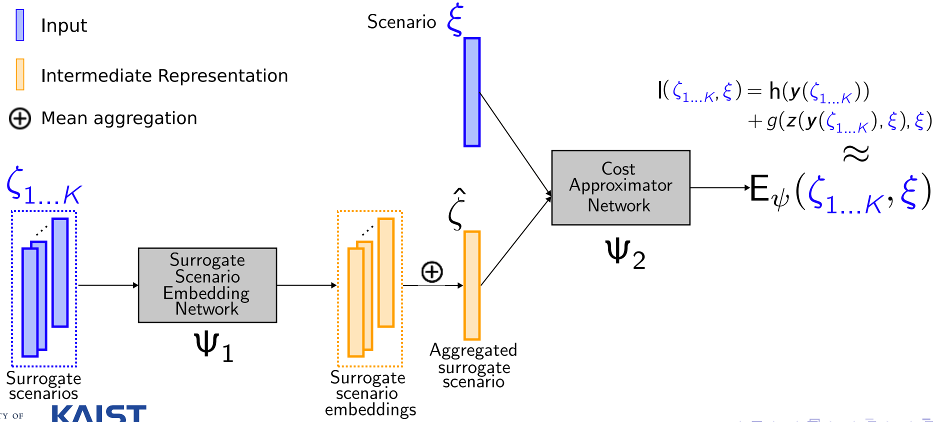
$$\mathcal{L}_{\text{task}}(\phi) = \frac{1}{|\tilde{\mathcal{D}}|} \sum_{i=1}^{|\tilde{\mathcal{D}}|} l(\boldsymbol{f}_{\phi}(\boldsymbol{x}^{(i)}), \xi^{(i)}) \tag{Task-Loss}$$

- where $l(\zeta_{1\ldots K}, \xi) = h(\boldsymbol{y}(\zeta_{1\ldots K})) + g(\boldsymbol{z}(\boldsymbol{y}(\zeta_{1\ldots K}), \xi), \xi)$
- Evaluating $l(\zeta_{1\ldots K}, \xi)$ requires solving (ζ-SAA) and a single subproblem (SP) to obtain $\boldsymbol{y}(\zeta_{1..K})$ and $\boldsymbol{z}(\boldsymbol{y}(\zeta_{1\ldots K}), \xi)$, respectively.

Proposed Approach

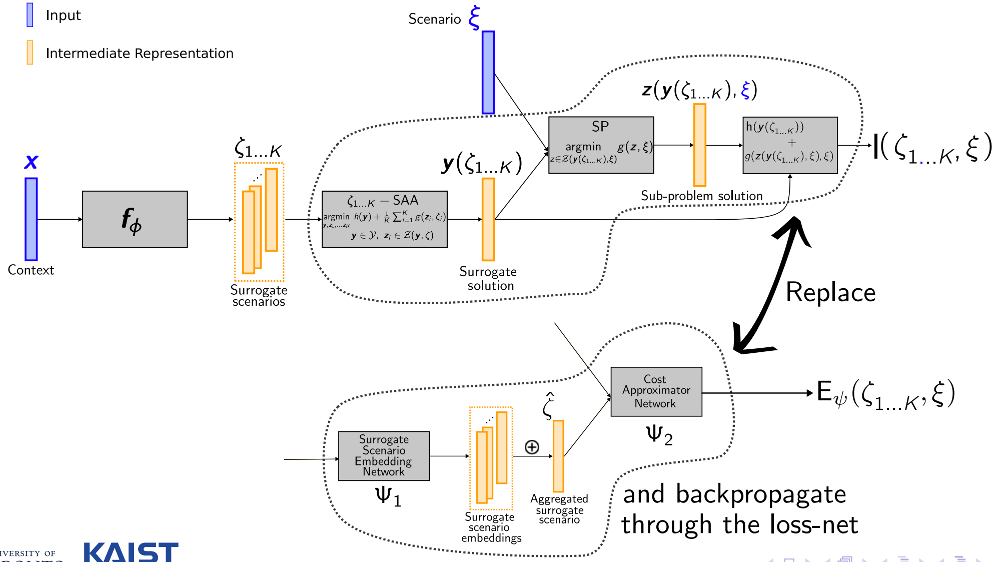
Problem Driven Contextual Scenario Reduction

- Motivated by the bi-level problem, a neural architecture is proposed to model the downstream loss
- Zharmagambetov et al. (2023) and Lee et al. (2022) also use neural architectures to approximate losses
- The proposed architecture is inspired by Dumouchelle et al. (2022)



Proposed Approach

Problem Driven Contextual Scenario Reduction



Proposed Approach

Method # 2: Static Problem Driven Contextual Scenario Reduction

- Globally approximate $I(\zeta_{1\ldots K}, \xi)$ by E_ψ via sampling $\zeta_{1\ldots K}, \xi$ over a specified input distribution. Then, the obtained approximation is used to guide the learning for $\boldsymbol{f}_\phi$

- The input samples and associated targets are denoted by $\mathcal{D}_{\text{loss}} = \{((\zeta_{1\ldots K}^{(i)}, \xi^{(i)}),\ I^{(i)}) \text{ where } I^{(i)} = I(\zeta_{1\ldots K}^{(i)}, \xi^{(i)})\}_{i=1}^{N_{\text{loss}}}$

$$\min_{\psi} \frac{1}{|\mathcal{D}_{\text{loss}}|} \sum_{i=1}^{N_{\text{loss}}} (E_\psi(\zeta_{1\ldots K}^{(i)}, \xi^{(i)}) - I^{(i)})^2 \quad ((\zeta_{1\ldots K}^{(i)}, \xi^{(i)}), I^{(i)}) \in \mathcal{D}_{\text{loss}}$$

- Given a trained loss-net, the approximate task loss is defined as:

$$\tilde{\mathcal{L}}_{\text{task}}(\phi) := \frac{1}{|\tilde{\mathcal{D}}|} \sum_{i=1}^{|\tilde{\mathcal{D}}|} E_\psi(\boldsymbol{f}_\phi(\boldsymbol{x}^{(i)}), \xi^{(i)}) \qquad \text{(Appx-Task-Loss)}$$

- directly minimizing the approximate task loss tends to maximize the error of the resulting task-net predictions $\Longrightarrow$

$$\min_{\phi} \lambda \tilde{\mathcal{L}}_{\text{task}}(\phi) + \mathcal{L}_{\text{MMD}}(\phi) \qquad \text{(Static-PCSR)}$$

Proposed Approach

Method # 3: Dynamic Problem Driven Contextual Scenario Reduction

- Zharmagambetov et al. (2023) and Lee et al. (2022) both note that learning networks to approximate a loss globally is a challenging task since the input space is high dimensional Ω^{K+1}.
- Instead, they propose using the loss-net to construct local approximations around the predictions from the task-net dynamically.
- Given a batch $B \subset \tilde{\mathcal{D}}$, dynamic training corresponds to the following dynamics at iteration t:

$$\begin{aligned} \psi_t &\leftarrow \psi_{t-1} - \nabla_\psi \frac{1}{|B|} \sum_{(\boldsymbol{x},\xi)\in B} \left(E_\psi(\boldsymbol{f}_{\phi_{t-1}}(\boldsymbol{x},\xi) - l(\boldsymbol{f}_{\phi_{t-1}}(\boldsymbol{x}),\xi)\right)^2 && \text{Loss step} \\ \phi_t &\leftarrow \phi_{t-1} - \nabla_\phi \frac{1}{|B|} \sum_{(\boldsymbol{x},\xi)\in B} \lambda E_{\psi_t}(\boldsymbol{f}_\phi(\boldsymbol{x}),\xi) + \mathcal{L}^{(B)}_{\text{MMD}}(\phi)) && \text{Task step} \end{aligned} \qquad \text{(Dynamic Training)}$$

Experimental Results

Two-Stage Portfolio Selection Problem

- Edirisinghe and Zhang (2013) consider a 2SP for asset selection that considers uncertain forecasts in the second stage.

$$
\begin{aligned}
\max \quad & (\boldsymbol{\mu}^{(0)})^\intercal \boldsymbol{y}_{\text{portfolio}} - \gamma ||\boldsymbol{y}_{\text{trade}}||_1 - \lambda (\boldsymbol{y}_{\text{portfolio}})^\intercal \boldsymbol{\Sigma}^{(0)} \boldsymbol{y}_{\text{portfolio}} + d \sum_{i=1}^{n_s} \pi_i Q_i(\boldsymbol{y}_{\text{portfolio}}, \boldsymbol{y}_{\text{trade}}) \\
\text{s.t} \quad & \boldsymbol{y}_{\text{trade}} = \boldsymbol{y}_{\text{portfolio}} - \boldsymbol{y}_{\text{initial position}}
\end{aligned}
$$

- where $\boldsymbol{\mu}^{(0)}$, $\boldsymbol{\Sigma}^{(0)}$ represents the expectation and covariance of asset returns, d represents a discount factor for expected second stage cost.

- Forecast $i \in [n_s]$ occurs with conditional probability $\pi_i | \boldsymbol{x}$ and profit $Q_i(\boldsymbol{y}_{\text{portfolio}}, \boldsymbol{y}_{\text{trade}})$, given by:

$$
\begin{aligned}
\max \quad & (\boldsymbol{\mu}^{(i)})^\intercal \boldsymbol{z}_{\text{portfolio}} - \gamma ||\boldsymbol{z}_{\text{trade}}||_1 - \lambda (\boldsymbol{z}_{\text{portfolio}})^\intercal \boldsymbol{\Sigma}^{(i)} \boldsymbol{z}_{\text{portfolio}} \\
\text{s.t.} \quad & \boldsymbol{z}_{\text{trade}} = \boldsymbol{z}_{\text{portfolio}} - \boldsymbol{y}_{\text{portfolio}}, \quad \boldsymbol{1}^\intercal \boldsymbol{z}_{\text{portfolio}} = 1
\end{aligned}
$$

- where superscripts (i) and (2) denote the regime-dependent information and second stage variables.

Experimental Results

What does CSRO look like for the Two-Stage Portfolio Problem?

- Setting: 50 Assets, a reasonable autoregressive model for asset returns (that is hidden), constant covariance

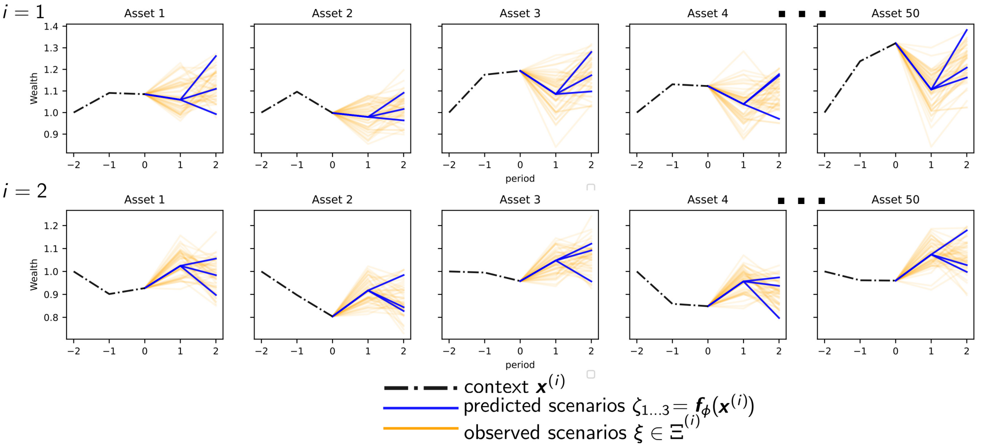


Figure: CSRO maps context to surrogate scenarios and bypasses conditional estimation, sampling, and scenario reduction

Experimental Results

Timing

- For evaluation purposes, sample 1000 different contexts and sample 50 scenarios per context $\mathcal{D} = \{\boldsymbol{x}^{(i)}, \Xi^{(i)}\}_{i=1}^{1000}$ with $|\Xi^{(i)}| = 50$ and $\tilde{\mathcal{D}} = \{(\boldsymbol{x}, \xi) \mid \xi \in \Xi, (\boldsymbol{x}, \Xi) \in \mathcal{D}\}$
- Set 800 instances as training and 200 instances as validation with $\mathcal{D}_{\text{train}}$, $\tilde{\mathcal{D}}_{\text{train}}$ and $\mathcal{D}_{\text{val}}$ and $\tilde{\mathcal{D}}_{\text{val}}$

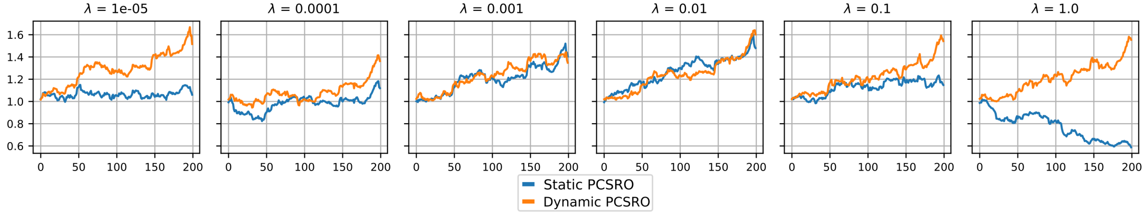
Method		DCSRO	Static PCSRO	Dynamic PCSRO
Data Generation Time (1000 instances 50 scenarios each)		2.6	729.5	729.5
Training Time	Task Net	1099.5	1610.5	4215.7
	Loss Net	0	12010.2	12010.2
	Total	1099.5	13620.7	16225.9
Solution Calculation Time (1000 instances 50 scenarios each)		40.5	43.49	46.15

Table: Training Times (seconds)

- Solving a single instance of 2SP via the deterministic equivalent takes 79.1 seconds
- Evaluating the $\zeta -$ SAA solutions over the scenarios takes ≈ 764 seconds

Experimental Results

Out of Sample Performance



Method	Mean		75th Percentile (Q3)		25th Percentile (Q1)	
λ	Dynamic PCSRO	Static PCSRO	Dynamic PCSRO	Static PCSRO	Dynamic PCSRO	Static PCSRO
0.00001	21.7	3.8	89.6	79.0	-50.8	-76.3
0.00010	16.4	6.6	78.8	81.0	-49.1	-54.9
0.00100	15.8	17.7	87.6	82.5	-51.7	-35.3
0.01000	24.2	20.4	79.6	92.5	-43.4	-47.5
0.10000	22.6	7.7	95.8	72.7	-42.3	-62.5
1.00000	22.9	-25.5	88.9	44.3	-30.8	-108.1

Table: PCSRO's excess returns over DCSRO statistics (bps)

Summary and Next Steps

- Summary
  - Contextual scenario reduction offers a fast and effective way to generate scenarios in a problem-driven manner
  - At decision-time, the proposed approach is independent of the number of scenarios and offers significant computational advantages
  - Problem-based contextual reduction offers improved out-of-sample performance over the proposed distribution-based approach
- Next Steps
  - Testing on other problems (i.e., discrete optimization problems)
  - Testing against standard non-contextual benchmarks (i.e., K-means and fast heuristics for Wasserstein scenario reduction)
  - Sensitivity analysis concerning various parameters (i.e., K)

Thank You!

We look forward to your questions

Dupačová, J., Gröwe-Kuska, N., & Römisch, W. (2003). Scenario reduction in stochastic programming. *Mathematical programming*, *95*, 493–511.

Römisch, W. (2003). Stability of stochastic programming problems. *Handbooks in operations research and management science*, *10*, 483–554.

Rachev, S. T., & Römisch, W. (2002). Quantitative stability in stochastic programming: The method of probability metrics. *Mathematics of Operations Research*, *27*(4), 792–818.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, *13*(25), 723–773. http://jmlr.org/papers/v13/gretton12a.html

Grigas, P., Qi, M., et al. (2021). Integrated conditional estimation-optimization. *arXiv preprint arXiv:2110.12351*.

Zharmagambetov, A., Amos, B., Ferber, A., Huang, T., Dilkina, B., & Tian, Y. (2023). Landscape surrogate: Learning decision losses for mathematical optimization under partial information. *arXiv preprint arXiv:2307.08964*.

Lee, J. Y., Patel, D., Goyal, P., Zhao, W., Xu, Z., & McCallum, A. (2022). Structured energy network as a loss. *Advances in Neural Information Processing Systems*, *35*, 20862–20875.

Dumouchelle, J., Patel, R., Khalil, E. B., & Bodur, M. (2022).Neur2sp: Neural two-stage stochastic programming. *arXiv preprint arXiv:2205.12006.*

Edirisinghe, C., & Zhang, X. (2013). Dynamic portfolio optimization under regime-based firm strength. In *Stochastic programming: Applications in finance, energy, planning and logistics* (pp. 129–154). World Scientific.