Predict, then smart optimize with stochastic programming

Rohit Kannan\(^1\), Güzin Bayraksan\(^2\), and Jim Luedtke\(^1\)

\(^1\)Wisconsin Institute for Discovery, University of Wisconsin-Madison
\(^2\)Department of Integrated Systems Engineering, The Ohio State University

Setup

- Traditional SP formulation: \(\min_{z \in \mathbb{Z}} \mathbb{E} \{ c(z, Y) \} \)
- Data-driven SP: samples \(\{ y_i \}_{i=1}^n \) of random variables \(Y\)
- Often also have data \(\{ x_i \}_{i=1}^n \) of random features/covariates \(X\) that can be used to predict \(Y\)

Applications

- Shipment planning under demand uncertainty
- Smart grid operation under renewables uncertainty

Formulation

\[ v^*(x) = \min_{z \in \mathbb{Z}} \mathbb{E} \{ c(z, Y) \mid X = x \} \]

- \(X = x\) is a new random observation of the covariates
- Concurrent data \(D_n := \{ (y_i, x_i) \}_{i=1}^n \) of \(Y\) and \(X\)

Predict-then-smart-optimize frameworks

Learn model to predict \(Y\) given \(X = x\), use as proxy for \(f^*\).
Use the residuals of this model on the training data \(D_n\) as proxy for samples of the errors \(\varepsilon\).

- Estimate \(f^*\) using the data \(D_n\), e.g.,
  \[ \hat{f}_n(x) \in \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \ell(y_i, f(x_i)) \]

- Use empirical residuals \(\hat{\varepsilon}_n := y - \hat{f}_n(x)\) as proxy for samples of \(\varepsilon\) within a SAA framework
  \[ \varepsilon_{ER}^n(x) \in \arg \min_{z \in \mathbb{Z}} \frac{1}{n} \sum_{i=1}^n c(z, \hat{f}_n(x) + \hat{\varepsilon}_n) \quad \text{(ER-SAA)} \]

- Using leave-one-out residuals \(\hat{\varepsilon}_{n,j} := y_i - \hat{f}_{-i}(x_i)\) within the SAA could work better with limited data
  \[ \varepsilon_{J}^n(x) \in \arg \min_{z \in \mathbb{Z}} \frac{1}{n} \sum_{i=1}^n c(z, \hat{f}_n(x) + \hat{\varepsilon}_{n,j}) \quad \text{(J-SAA)} \]

Theoretical guarantees

- Conditions on the SP and the learning step for asymptotic optimality, rates of convergence of the ER-SAA, J-SAA solutions
- Applicable to two-stage stochastic linear programs (LPs)
- Can handle general learning frameworks and time series data \(D_n\)

For ER-SAA, the learning step must satisfy
  \[ \hat{f}_n(x) \xrightarrow{p} f^*(x) \quad \text{and} \quad \frac{1}{n} \sum_{i=1}^n \| f^*(x^i) - \hat{f}_n(x^i) \|^2 \xrightarrow{p} 0. \]

Setup for computational experiments

Two-stage resource allocation LP model
- Meet demands of 30 customers for 20 resources
- Uncertain demands \(Y\) generated according to
  \[ Y_j = \alpha^*_j + \sum_{i=1}^n \beta^*_j(X_i) p + \varepsilon_j, \quad \forall j \in \{1, \cdots, 30\}, \]
  where \(\varepsilon_j \sim \mathcal{N}(0, \sigma_j^2)\), \(p \in \{0.5, 1, 2\}\), \(\dim(X) \in \{10, 100\}\)
- Fit a linear model with OLS regression (even when \(p \neq 1\))

Numerical results

Legend: \(k\): kNN-based approach of Bertsimas & Kallus (2019),
\(\mathbb{E}\): ER-SAA + OLS, \(J\): J-SAA + OLS, \(UCB\): 99% upper confidence bound

- Advantage of using our data-driven formulations

- Advantage of the J-SAA formulation with limited data

Acknowledgment: Funding from the Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program under Contract Number DE-AC02-06CH11347.