Predict, then smart optimize with stochastic programming

Rohit Kannan

UW-Madison

February 28, 2020

Joint work with Güzin Bayraksan (OSU) and Jim Luedtke (UWM)

Acknowledgment: funding from the MACSER project (DOE)

Motivation

- Traditional stochastic programming (SP) formulation

$$\min_{z \in \mathcal{Z}} \mathbb{E}_Y[c(z, Y)]$$

Data-driven version: have access to (iid) samples $\{y^i\}_{i=1}^n$ of Y

$$\approx \min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, y^i)$$

Motivation

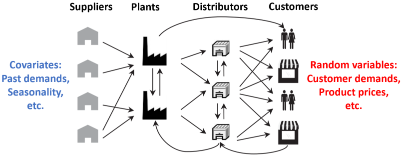
- Traditional stochastic programming (SP) formulation

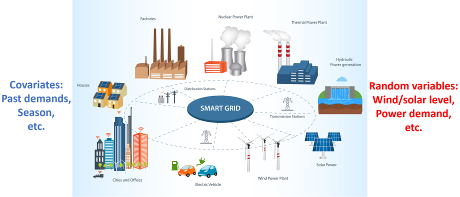
$$\min_{z \in \mathcal{Z}} \mathbb{E}_Y[c(z, Y)]$$

Data-driven version: have access to (iid) samples $\{y^i\}_{i=1}^n$ of Y

$$\approx \min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, y^i)$$

- Often concurrently observe data $\{x^i\}_{i=1}^n$ of covariates that can better characterize the distribution of Y relevant for SP





Problem setup

Given

- Data $\mathcal{D}_n := \{(y^i, x^i)\}_{i=1}^n$ on Y and X
- New random covariate observation $X = x$

Want to solve

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

Problem setup

Given

- Data $\mathcal{D}_n := \{(y^i, x^i)\}_{i=1}^n$ on Y and X
- New random covariate observation $X = x$

Want to solve

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

Assume

- True model: $Y = f^*(X) + \varepsilon$
- Known function class $\mathcal{F}$ such that $f^* \in \mathcal{F}$

$$\implies v^*(x) \equiv \min_{z \in \mathcal{Z}} \mathbb{E}_\varepsilon[c(z, f^*(x) + \varepsilon)]$$

Problem setup

Given

- Data $\mathcal{D}_n := \{(y^i, x^i)\}_{i=1}^n$ on Y and X
- New random covariate observation $X = x$

Want to solve

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

Assume

- True model: $Y = f^*(X) + \varepsilon$
- Known function class $\mathcal{F}$ such that $f^* \in \mathcal{F}$

$$\implies v^*(x) \equiv \min_{z \in \mathcal{Z}} \mathbb{E}_\varepsilon[c(z, f^*(x) + \varepsilon)]$$

If we know f^*, can solve

$$\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, f^*(x) + \varepsilon^i), \text{ where } \varepsilon^i := y^i - f^*(x^i)$$

Main ideas [1/2]

Estimate f^* from the data $\mathcal{D}_n$, and use this estimate as proxy for f^*.
Use its residuals on the data $\mathcal{D}_n$ as proxy for samples of the errors ε.

Main ideas [1/2]

Estimate f^* from the data $\mathcal{D}_n$, and use this estimate as proxy for f^*. Use its residuals on the data $\mathcal{D}_n$ as proxy for samples of the errors ε.

- For instance, estimate f^* using

$$\hat{f}_n(\cdot) \in \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y^i, f(x^i)\right)$$

- Use empirical residuals $\hat{\varepsilon}_n^i := y^i - \hat{f}_n(x^i)$ as proxy for samples of ε within a SAA framework

$$\hat{z}_n^{ER}(x) \in \arg\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, \hat{f}_n(x) + \hat{\varepsilon}_n^i) \qquad \text{(ER-SAA)}$$

Main ideas [1/2]

Estimate f^* from the data $\mathcal{D}_n$, and use this estimate as proxy for f^*. Use its residuals on the data $\mathcal{D}_n$ as proxy for samples of the errors ε.

- For instance, estimate f^* using

$$\hat{f}_n(\cdot) \in \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y^i, f(x^i)\right)$$

- Use empirical residuals $\hat{\varepsilon}_n^i := y^i - \hat{f}_n(x^i)$ as proxy for samples of ε within a SAA framework

$$\hat{z}_n^{ER}(x) \in \arg\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, \hat{f}_n(x) + \hat{\varepsilon}_n^i) \qquad \text{(ER-SAA)}$$

- Optimization step unaffected by complexity of prediction step
- Possible concerns: choice of $\mathcal{F}$, overfitting (if n is small)

Main ideas [2/2]

Can mitigate the effects of overfitting by using cross-validation residuals instead of empirical residuals

Main ideas [2/2]

Can mitigate the effects of overfitting by using cross-validation residuals instead of empirical residuals

- Estimate f^* using leave-one-out regression models

$$\hat{f}_{-j}(\cdot) \in \operatorname*{arg\,min}_{f \in \mathcal{F}} \frac{1}{n-1} \sum_{\substack{i=1 \\ i \neq j}}^{n} \ell\left(y^i, f(x^i)\right), \quad j = 1, \cdots, n$$

- Use leave-one-out residuals $\hat{\varepsilon}^i_{n,J} := y^i - \hat{f}_{-i}(x^i)$ within a SAA framework

$$\hat{z}^J_n(x) \in \operatorname*{arg\,min}_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, \hat{f}_n(x) + \hat{\varepsilon}^i_{n,J}) \qquad \text{(J-SAA)}$$

Main ideas [2/2]

Can mitigate the effects of overfitting by using cross-validation residuals instead of empirical residuals

- Estimate f^* using leave-one-out regression models

$$\hat{f}_{-j}(\cdot) \in \arg\min_{f \in \mathcal{F}} \frac{1}{n-1} \sum_{\substack{i=1 \\ i \neq j}}^{n} \ell\left(y^i, f(x^i)\right), \quad j = 1, \cdots, n$$

- Use leave-one-out residuals $\hat{\varepsilon}^i_{n,J} := y^i - \hat{f}_{-i}(x^i)$ within a SAA framework

$$\hat{z}^J_n(x) \in \arg\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, \hat{f}_n(x) + \hat{\varepsilon}^i_{n,J}) \qquad \text{(J-SAA)}$$

- Several other possible variants (K-fold CV, bootstrap, $\cdots$)

Flavor of theoretical results

Consider $$\min_{z \in \mathcal{Z}} c_z^\mathsf{T} z + \mathbb{E}_Y[Q(z, Y)],$$

where $$Q(z, Y) := \min_{v \in \mathbb{R}_+^{d_v}} \left\{ q_v^\mathsf{T} v : Wv = h(Y) - T(Y)z \right\}$$

Assumption

There is a constant $\alpha \in (0, 1]$ s.t. the prediction step satisfies

- Prediction error: $\|f^*(x) - \hat{f}_n(x)\|^2 = O_P\left(n^{-\alpha}\right)$
- Training estimation MSE: $\dfrac{1}{n}\displaystyle\sum_{i=1}^{n} \|f^*(x^i) - \hat{f}_n(x^i)\|^2 = O_P(n^{-\alpha})$

Under the above assumptions, we have

Theorem

$$\mathbb{E}\left[c(\hat{z}_n^{ER}(x), f^*(x) + \varepsilon)\right] = v^*(x) + O_P(n^{-\frac{\alpha}{2}})$$

Setup for computational experiments

Two-stage resource allocation LP model

- Meet demands of 30 customers for 20 resources
- Uncertain demands Y generated according to

$$Y_j = \alpha_j^* + \sum_{l=1}^{3} \beta_{jl}^*(X_l)^p + \varepsilon_j, \quad \forall j \in \{1, \cdots, 30\},$$

  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 $p \neq 1$)

$$Y_j = \alpha_j + \sum_{l=1}^{\dim(X)} \beta_{jl}(X_l)^p + \eta_j, \quad \forall j \in \{1, \cdots, 30\},$$

  where η_j are zero-mean errors
- Estimate optimality gap of solutions $\hat{z}_n^{ER}(x)$ and $\hat{z}_n^{J}(x)$

$$\hat{z}_n^{ER}(x) \in \operatorname*{arg\,min}_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^{n} c(z, \hat{f}_n(x) + \hat{\varepsilon}_n^i)$$

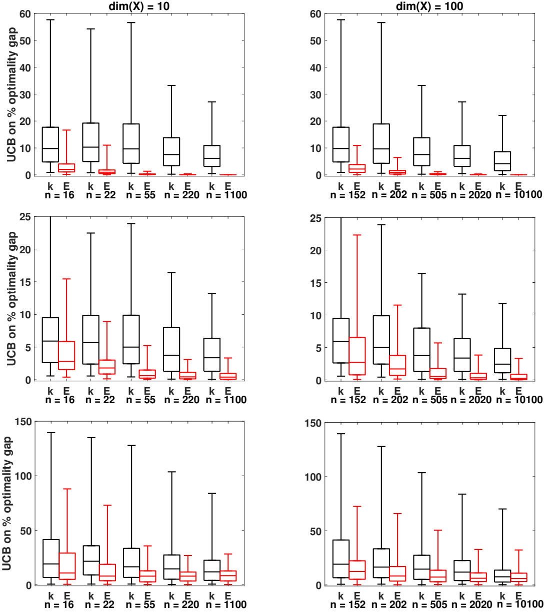
Advantage of using our data-driven formulation

degree

$\mathbf{p = 1}$

$\mathbf{p = 0.5}$

$\mathbf{p = 2}$



Red: ours
Black: benchmark

Advantage of the J-SAA formulation with limited data

Red: ER-SAA, Green: J-SAA