Heterogeneous Causal Learning for Effectiveness Optimization

key: 1°. combine multiple outcomes ⇐ (cost and gain) optimize jointly

2°. How to optimize [duality / Direct Ranking / Constrained Ranking — [quantile pooling (%/budget) / annealing

Algorithms:

retention treatment effect: $\tau^{*r}(x^{(i)}) = E(Y_1^r - Y_0^r \mid X^{(i)} = x^{(i)})$

cost " : $\tau^{*c}(x^{(i)}) = E(Y_1^c - Y_0^c \mid X^{(i)} = x^{(i)})$

maximize $\sum_{i=1}^{n} \tau^{*r}(x^{(i)}) z_i$ subject to $\sum_{i=1}^{n} \tau^{*c}(X^{(i)}) z_i \le B \quad z_i \in \{0,1\}$

1. Duality R-Learner:

• Lagrangian multipliers: $L(z,\lambda) = -\sum_{i=1}^{n} \tau^{*r}(x^{(i)}) z_i + \lambda \left(\sum_{i=1}^{n} \tau^{*c}(x^{(i)}) z_i - B\right) \rightarrow (6)$

[optimize z_i : maximize $\sum_{i=1}^{n} z_i s_i$ subject $0 \le z_i \le 1$

$s_i = \tau^{*r}(x^{(i)}) - \lambda \tau^{*c}(x^{(i)}) \rightarrow$ effectiveness score $\geq 0 \rightarrow z_i = 1$; $< 0 \rightarrow z_i = 0$

" " λ : $\frac{\partial g}{\partial \alpha} = B - \sum_{i=1}^{n} \tau^{*c}(x^{(i)}) z_i \rightarrow \lambda \Rightarrow \lambda + \alpha (B - \sum_{i=1}^{n} \tau^{*c}(x^{(i)}))$

α: learning rate

• fit a single scoring model:

$$s_i = \tau^{*E}(x^{(i)}) = \tau^{*r}(x^{(i)}) - \lambda \tau^{*c}(x^{(i)}) \longrightarrow (10)$$

$$= E(Y_1^r - Y_0^r \mid X = x) - \lambda E(Y_1^c - Y_0^c \mid X = x)$$

$$= E(cY_1^r - \lambda Y_1^c - c Y_0^r - \lambda Y_0^c \mid X = x)$$

$$= E(Y_1^E - Y_0^E \mid X = x)$$

2. Direct Ranking Model

unconstrained optimization problem: minimize $\frac{\tau^{*c}(x)}{\tau^{*r}(x)}$ cost per unit gain

loss function: $f(\cdot) = \arg\min_f \left\{ \frac{\tau^c}{\tau^r} + \Lambda_n(f(\cdot)) \right\} \longrightarrow (9)$

direct optimization is well suited for deep learning

3. Constrained Ranking Models

• Quantile pooling : a new pooling method for selecting a quantile of effectiveness (e.g. %/budget B) measures from the whole population using a sorting operator.

$S_i^{(k)} = \tanh(f^{(k)}(x^{(i)}))$ → input offset

fall-off function. (sigmoid)

$$V_i^{(k)} = S_i^{(k)} - d^{*(k)} \quad v_i^{(k)} = \sigma^*(V_i^{(k)}) = \frac{1}{1+\exp(-t^* V_i^{(k)})}$$

1°. Top quantile constraint:

$$d^{*(k)} = \Gamma_Q(s^{(k)}, q) = u(\overset{\text{sorting}}{\Psi(s^{(k)})}, n) \rightarrow \text{nth operator } n = N\frac{q}{100}$$

first sort user effectiveness scores then take the $q\%$ quantile value as offset $d^{(k)}$

2°. Fixed Cost constraint:

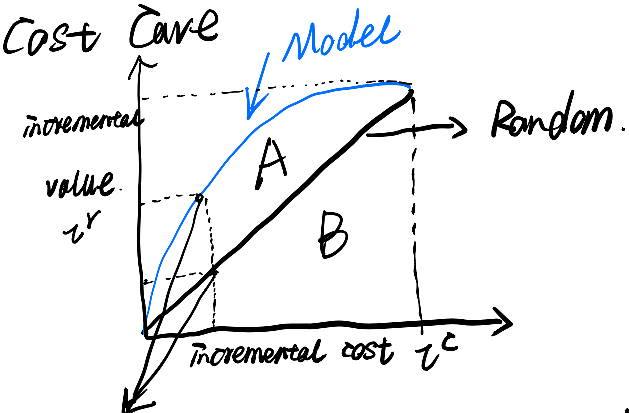
$$d^{*(k)} = \Gamma_Q(s^{(k)}, B) = \Phi(\overset{\text{cumulative sum}}{w}(\Psi(s^{(k)})), B, s^{(k)})$$

sort user effectiveness scores then take quantile value of $s^{(k)}$ as offset $d^{(k)}$ where the quantile value corresponds to the rank of user just before where the budget exceeds B.

- Annealing: t^* have a schedule of raising temperature.

Evaluation:

Cost Curve



Area Under Cost Curve (AUCC)

$$: \frac{A+B}{2B} \quad [0.5, 1)$$

$$\#\{T_i = 1 \mid S_i > S_i^{pth}\} \times ATE(x_i \mid S_i > S_i^{pth})$$

number of treatment samples at this point on the curve $\times$ ATE of this group.

Future work

1. Smart Explore/ Exploit (multi-arm bandit / Bayesian optimization). ↓ Similar what bidder wanted to do.

2. Deep Embedding → combine with OTVAF method?

Thoughts / Ideas / Learnings.

1. Budget constraint seems more suitable for bidder use case which we

can optimize on ROAS directly.

2. Evaluation: → ATE → segmented by TMLE
3. optimazation idea: 通常将 business problem → math → optimazation
   → deep learning framework.
4. Combined causal inference with deep learning:
   quantile pool (ranking).
5. Results shows really great → directly optimized by.
6. Used R-Learner duality set up here → somehow could we apply the similarly optimazation ideas / framework to other meta-learners (maybe not).

References:

Weak Duality and Strong Duality

- Lagrangian dual problem

$$\begin{array}{ll} \text{maximize} & g(\lambda, \nu) \\ \text{subject to} & \lambda \geq 0 \end{array} \qquad solution: d^*$$

- Weak duality and strong duality

$d^* \leq p^*$ (always true) $\qquad d^* = p^*$ (strong duality)

- Slater's constraint qualification
  - For optimization problem in forms of:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, i = 1, \ldots m \\ & Ax = b \end{array}$$

  - The strong duality usually true. Need additional constraint qualification: Slater's condition: $\exists x \in \mathcal{D}, f_i(x) < 0, i = 1, .., m$

Lagrangian Dual Function

- Consider the optimization problem in standard form:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, i = 1, \ldots, m \\ & h_i(x) = 0, i = 1, \ldots, p \end{array}$$

  - The domain $\mathcal{D} = \cap_i \textbf{dom} f_i \cap \cap_i \textbf{dom} h_i$
- The Lagrangian dual function is:
  - For $\lambda \in \mathbb{R}^m, \nu \in \mathbb{R}^p$

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right)$$

  - The Lagrangian dual function is concave
- Weak duality $\quad \forall \lambda \geq 0, \forall \nu \quad g(\lambda, \nu) \leq p^*$
  - Simple proof: $\quad \forall \lambda \geq 0: \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \leq 0$