Transfer Faster, Price Smarter: Minimax Dynamic Pricing under Cross-Market Preference Shift

Yi Zhang[♯], Elynn Chen[†], Yujun Yan[♭]

[♯]Columbia University, [†]New York University, [♭]Dartmouth College

October 21, 2025

- **Challenge**: In Dynamic pricing, new markets suffer from data scarcity, while mature markets generate abundant logs.
- **Gap**: Existing transfer methods assume identical utilities across markets, failing under preference shift.
- **Model**: A general random utility model for the market value of the product is given by

$$v_t^{(0)} = \mathring{g}^{(0)}(\boldsymbol{x}_t^{(0)}) + \varepsilon_t, \tag{1}$$

  In cross-market transfer learning, we observe additional samples from K sources markets indexed by superscript $^{(k)}$ for $k \in [K]$:

$$v_t^{(k)} = \mathring{g}^{(k)}(\boldsymbol{x}_t^{(k)}) + \varepsilon_t, \tag{2}$$

- **Goal**: Design a framework to leverage auxiliary markets while provably handling structured model shift.

The source and target markets operate concurrently; streaming data from large markets must be incorporated into the pricing decisions of small markets in real time.

Algorithm 1: CM-TDP-O2O$_{on}$

Input: Streaming source data $\{(p_t^{(k)}, \boldsymbol{x}_t^{(k)}, y_t^{(k)})\}_{t\ge1}$ for $k \in [K]$; streaming target contexts $\{\boldsymbol{x}_t^{(0)}\}_{t\ge1}$

Initialisation: $\ell_1 \leftarrow 1,\ \mathcal{T}_1 = \{1\},\ \widehat{g}_0^{(0)} = 0.$

for $m = 1, 2, \ldots$ **do** // episodes

  Compute $\ell_m = 2^{m-1}$, $\mathcal{T}_m = \{\ell_m, \ldots, \ell_{m+1} - 1\}$.

  // (i) aggregate previous episode's source data

  $\widehat{g}_m^{(\mathrm{ag})} \leftarrow \texttt{MLE\_or\_KRR}(\{(p_t^{(k)}, \boldsymbol{x}_t^{(k)}, y_t^{(k)})\}_{t\in\mathcal{T}_{m-1}, k\in[K]})$.

  // (ii) debias with previous episode's target data

  $\widehat{\delta}_m \leftarrow \texttt{Debias}(\widehat{g}_m^{(\mathrm{ag})}, \{(p_t^{(0)}, \boldsymbol{x}_t^{(0)}, y_t^{(0)})\}_{t\in\mathcal{T}_{m-1}})$.

  // For functions MLE_or_KRR and Debias, call Algorithm 2 for linear utility (or Algorithm 3 for non-parametric utility)

  Set $\widehat{g}_m^{(0)} \leftarrow \widehat{g}_m^{(\mathrm{ag})} + \widehat{\delta}_m$.

  for $t \in \mathcal{T}_m$ **do** // episode pricing

    Post price $\widehat{p}_t^{(0)} = h(\widehat{g}_m^{(0)}(\boldsymbol{x}_t^{(0)}))$; observe $y_t^{(0)}$ and store data.

Offline-to-Online ($\mathrm{O2O}_{\mathrm{off}}$)

The firm holds a fixed log of source-market data gathered before the target market opens, and this static information is used once the target goes live.

Algorithm 4: CM-TDP-$\mathrm{O2O}_{\mathrm{off}}$

Input: Offline source market data $\{(p_t^{(k)}, \boldsymbol{x}_t^{(k)}, y_t^{(k)})\}_{t \in \mathcal{H}^{(k)}}$ for $k \in [K]$; feature matrix $\{\boldsymbol{x}_t^{(0)}\}_{t \in \mathbb{N}}$ for the target market

```
/* ******* Phase 1:  Update with transfer learning *******          */
```

Call Algorithm 2 or 3 to calculate the initial aggregated estimate $\widehat{g}^{(\mathrm{ag})}$ using entire source market data $\{(\boldsymbol{p}_t^{(k)}, \boldsymbol{X}_t^{(k)}, \boldsymbol{y}_t^{(k)})\}_{t \in \mathcal{H}^{(k)}}$ for $k \in [K]$

Apply the price $\widehat{p}_1^{(0)} := h(\widehat{g}^{(\mathrm{ag})}(\boldsymbol{x}_1^{(0)}))$ and collect data $(\widehat{p}_1^{(0)}, \boldsymbol{x}_1^{(0)}, y_1^{(0)})$.

for *each episode* $m = 2, \ldots, m_0$ **do**

- Set the length of the m-th episode: $\ell_m := 2^{m-1}$
- Call Algorithm 2 or 3 to calculate the debiasing estimate $\widehat{\delta}_m$ using target market data $\{(p_t^{(0)}, \boldsymbol{x}_t^{(0)}, y_t^{(0)})\}_{t \in [2^{m-2}, 2^{m-1}-1]}$ and aggregated estimate $\widehat{g}^{(\mathrm{ag})}$.
- Set
$$\widehat{g}_m^{(0)} := \widehat{g}^{(\mathrm{ag})} + \widehat{\delta}_m.$$
- For each time t, apply price $\widehat{p}_t^{(0)} := h(\widehat{g}_m^{(0)}(\boldsymbol{x}_t^{(0)}))$ and collect data $(\widehat{p}_t^{(0)}, \boldsymbol{x}_t^{(0)}, y_t^{(0)})$.

```
/* ******* Phase 2:  Update without transfer learning *******       */
```

for *each* $m \geq m_0 + 1$ **do**

- Set the length of the m-th episode: $\ell_m := 2^{m-1}$
- Call Algorithm 2 or 3 to calculate $\widehat{g}_m^{(0)}$ using target market data $\{(p_t^{(0)}, \boldsymbol{x}_t^{(0)}, y_t^{(0)})\}_{t \in [2^{m-2}, 2^{m-1}-1]}$.
- For each time t, apply price and collect data.

Output: Offered price $\widehat{p}_t^{(0)}$, $t \geq 1$

Two Utility Models

Parametric Utility Model

- **Model**: Consider a linear model for the mean utility:

$$v_t^{(k)} = \mathbf{x}_t^{(k)} \cdot \boldsymbol{\beta}^{(k)} + \varepsilon_t, \quad k \in \{0\} \cup [K] \tag{3}$$

- **Similarity Characterization**: The maximum l_0-norm of the difference between target and source coefficients is bounded:

$$\max_{k \in [K]} \|\boldsymbol{\beta}^{(0)} - \boldsymbol{\beta}^{(k)}\|_0 \leq s_0.$$

- **Estimation**: Maximum Likelihood Estimation

Nonparametric Utility Model

- **Model**: Utilities in Reproducing kernel Hilbert Space:

$$v_t^{(k)} = g^{(k)}(\mathbf{x}_t^{(k)}) + \varepsilon_t, \quad g^{(k)} \in \mathcal{H}_k, \; k \in 0 \cup [K], \tag{4}$$

- **Similarity Characterization**: The discrepancy between target task and source task in the RKHS norm is uniformly bounded as

$$\max_{k \in [K]} \|g^{(0)} - g^{(k)}\|_K \leq H.$$

- **Estimation**: Kernel Logistic Regression

THM. Regret Bound (O2O$_{\text{on}}$, linear)

Upper Bound:

$$\mathrm{Regret}(T;\pi) = \mathcal{O}\Big(\frac{d}{K}\log d \log T + s_0 \log d \log T\Big).$$

Lower Bound:

$$\inf_{\pi}\ \sup\ \mathrm{Regret}(T;\pi)\ \ \geq\ \ c_1\,\frac{d}{K}\,\log T + c_2\, s_0 \log\frac{d}{s_0}\,\log T,$$

THM. Regret Bound (O2O$_{\text{on}}$, RKHS)

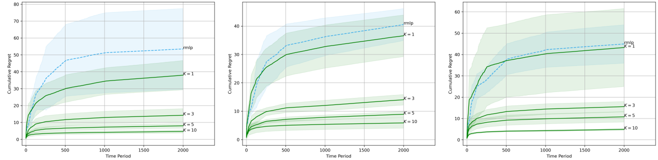
Upper Bound:

$$\mathrm{Regret}(T;\pi) = \mathcal{O}\left(K^{-\frac{2\alpha\beta}{2\alpha\beta+1}}\,T^{\frac{1}{2\alpha\beta+1}} + H^{\frac{2}{2\alpha+1}}\,T^{\frac{1}{2\alpha+1}}\right)$$

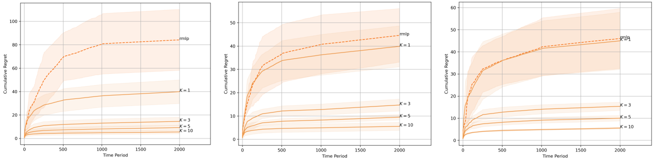
Lower Bound:

$$\inf_{\pi}\ \sup\ \mathrm{Regret}(T;\pi)\ \geq\ c\left\{K^{-\frac{2\alpha\beta}{2\alpha\beta+1}}\ T^{\frac{1}{2\alpha\beta+1}}\ +\ H^{\frac{2}{2\alpha+1}}\ T^{\frac{1}{2\alpha+1}}\right\}.$$

Configurations: (i) Identical Markets, (ii) Sparse difference Markets, (iii) Dense difference Markets



(a) Identical, $d = 10$ (b) Sparse, $d = 10$ (c) Dense, $d = 10$



(d) Identical, $d = 10$ (e) Sparse, $d = 10$ (f) Dense, $d = 10$

Figure 1: Cumulative regret across experimental conditions in $\mathrm{O2O_{on}}$ with linear (up) and RKHS (down) utility models.

Summary

- **Unified transfer pricing framework under utility shifts.** CM-TDP is *the first* dynamic pricing framework that allows *multiple* source markets whose utilities differ from the target by a structured shift, working in both Online-to-Online and Offline-to-Online regimes.
- **Minimax-optimal guarantees.** We establish minimax regret rate under linear mean utilities *and* RKHS-smooth utilities.
- **Bias-corrected aggregation architecture.** Our two-step *aggregate* $\rightarrow$ *debias* pipeline cleanly connects meta-learning, robust statistics, and exploration-driven bandits, and can plug in MLE, Lasso, or kernel ridge as well as black boxes.
- **Large empirical gains.** Simulations show up to $\underline{50\,\%}$ lower cumulative regret, $\underline{28\,\%}$ lower standard error and $\underline{5\times}$ faster learning relative to single-market learning, with the largest gains in data-scarce targets under Online-to-Online transfer.