**Motivation**

- Distant goal: online isotonic regression on partial orders
- Current solution for linear orders does not scale

> New model and algorithms for linear case

**Random Permutation Model**

Random Permutation Model

- Adversary chooses data instances \( x_1 < \ldots < x_T, y_1, \ldots, y_T \)
- Sample UAR a permutation \( \sigma = (\sigma_1, \ldots, \sigma_T) \) of \( \{1, \ldots, T\} \)
- Round \( t \): covariate \( x_{\sigma_t} \), true label \( y_{\sigma_t} \), and loss \((\hat{y}_{\sigma_t} - y_{\sigma_t})^2\)

Learner minimizes expected regret,

\[
R_T := E_x \left[ \sum_{t=1}^T (y_{\sigma_t} - \hat{y}_{\sigma_t})^2 \right] - L_T = \sum_{t=1}^T r_t,
\]

where \( r_t := E_x [(y_{\sigma_t} - \hat{y}_{\sigma_t})^2 - L^*_t] \) is the per-round regret and \( L^*_t = L^t ((x_{\sigma_t}, y_{\sigma_t}), \ldots, (x_{\sigma_t}, y_{\sigma_t})) \) is the optimal loss of the first \( t \) labeled instances.

**Random Permutation Online Isotonic Regression**

Fit an isotonic (non-decreasing) function to the data:

\[
f^* = \arg\min_{f\text{ isotonic}} \sum_{i=1}^T (y_i - f(x_i))^2
\]

**Offline Isotonic Regression**

- Iteratively merge data into blocks until no violator of isotonic constraints exists
- Assign to data in each block the average of their labels \( y_i \)
- Blocks correspond to level sets of \( f^* \)

**Leave-One-Out Loss**

With Data \( D = \{(x_1, y_1), \ldots, (x_T, y_T)\} \), the \( \ell_{oo} \) of a \( t \) round game is

\[
\ell_{oo}(D) := \frac{1}{t} \left( \sum_{i=t+1}^T (y_i - \hat{y}_i (D \setminus (x_i, y_i)))^2 \right) - L^*(D).
\]

**Lemma 1.** \( r_t(D) \leq \ell_{oo}(D) \) for any \( t \) and any data set \( D = \{(x_1, y_1), \ldots, (x_T, y_T)\} \).

**Lower Bound**

Adversarial lower bound [Kotlowski, Koolen, and Malek, 2016] applies to random permutation model: \( \ell_{oo} = \Omega(t^{-2/3}) \).

**Matching Bounds**

Theorem 2. There is an algorithm for the random-permutation model with excess leave-one-out loss \( \ell_{oo} = \widetilde{O}(t^{-2}) \) and hence expected regret \( R_T \leq \sum_t \tilde{O}(t^{-2}) = \tilde{O}(T^2) \), which matches the lower bound of \( \ell_{oo} = \Omega(t^{-2/3}) \).

Caveat: algorithm is not efficient (on partial orders!)

**Forward Algorithm**

Two observations:

- PAVA is efficient and generalizes to partial orders
- Follow The Leader algorithms are common in practice

**Forward Algorithm:** To predict at \( x_t \), imagine \( y'_t \in [0, 1] \), compute \( f^* \) on \( \{(x_1, y'_1) \ldots (x_{t-1}, y'_{t-1})\} \cup \{(x_t, y'_t)\} \), and play \( \hat{y}_{t} = f^*(x_t) \).

**Regret Bounds**

- IR-Int: Compute \( f^* \) on past data. Predict with average of \( f^* \) at nearest \( x_t \).
- Interpolation \( \hat{y}_i = \lambda_1 y_1 + (1 - \lambda_1) y_T \) (where \( y_0^T \) and \( y_T^T \) are plug-in \( y'_0 = 0^T \) and plug-in \( y'_T = 1^T \))
- Last step minimax:

\[
\hat{y}_i = \arg\min_{y \in [0, 1]} \max_{y' \in [0, 1]} \left\{ (\hat{y}_i - y_i)^2 - L^t(y) \right\}
\]

- IVAP predictors [Vovk et al., 2015]:

\[
\hat{y}_i = \frac{y_i^2}{y_i^2 + 1 - y_i^2}, \quad \hat{y}_i = \frac{1 + (y_i^T)^2 - (1 - y_i^T)^2}{2}
\]

**Heavy-\( \gamma \)**

Parameters: Weight \( c > 0 \) and label \( \gamma \in [0, 1] \).

- Algorithm: To predict at \( x_t \)

  - Compute isotonic regression \( f^* \) on weighted dataset

\[
D' := \{(x_s, y_s, 1) \mid 1 \leq s \leq t \} \cup \{(x_t, \gamma, c)\}
\]

- Predict \( y_t = f^*(x_t) \)

Efficient weighted algorithms available [Kynig et al., 2015].

**Tuning Heavy-\( \gamma \)**

Any fixed label \( \gamma \) works. We like \( \gamma = 1 \).

(Not all adaptive labels work. Fixed point + lower bound.)

Theorem 5. Heavy-\( \gamma \) has sub-optimal \( \ell_{oo} \) loss unless \( c = \Theta(t^{1/2}) \).