Dynamic Algorithms for Online Multiple Testing

Ziyu Xu (Neil)  Aaditya Ramdas
Dynamic Algorithms for Online Multiple Testing

Ziyu Xu (Neil)  Aaditya Ramdas
Problem: testing many different hypotheses

Our goal is to reject non-null hypotheses (make true discoveries)
Threshold for rejection/discovery output by algorithm

1st p-value resulting from experiment

\[ p_1 \leq \alpha_1 \]

2nd p-value

\[ p_2 \leq \alpha_2 \]

3rd p-value

\[ p_3 \leq \alpha_3 \]

\[ \ldots \]

infinite stream of p-values and hypotheses

= 

“online”

(Foster and Stine 2008)
Assumptions on p-values

\[ \Pr(p_k \leq s) \leq s \text{ for all } s \in (0, 1) \text{ if } H_k \text{ is null (superuniformity)} \]

\[ p_k \text{ potentially small if } H_k \text{ is non-null} \]

If we reject a null hypothesis, we make a “false discovery”
Key error metrics we wish to keep controlled

\[ FDP_k = \frac{\text{# of false discoveries by time } k}{\text{total # of discoveries by time } k} \]

\[ FDR = \sup_{k \in \mathbb{N}} \mathbb{E} \left[ FDP_k \right] \quad \text{error metric primarily considered in prior work} \]

\[ FDX^\epsilon_K = \Pr(\exists k \geq K : FDP_k > \epsilon) \quad \text{probabilistic bound} \]

\[ FDR = \sup_{\tau \in \mathcal{T}} \mathbb{E} \left[ FDP_{\tau} \right] \quad \text{Extend FDR to include data-dependent stopping times.} \]

Our algorithm controls these set of stopping times.
Alpha-investing: a method for selecting alpha values while maintaining error control

Nonzero wealth $\iff$ FDR or FDX is controlled

$$\text{Power} = \frac{\text{# of true discoveries}}{\text{# of non-null hypotheses}}$$

Goal: Design algorithms that have high power and ensure that $\text{FDX}_K^c \leq \delta$ or $\text{FDR} \leq \ell$

Lose wealth at each step based on size of alpha values.

And we continue to spend...
Contributions

1. First “practically” powerful algorithm with FDX control.
2. “Dynamic” algorithm for allocating alpha values that improves over prior methods.
3. First method that provides FDR control at stopping times. (see paper)
The estimator view of FDP (LORD)

\[ \text{FDP}_k = \frac{\sum_{i=1}^{k} \mathbf{1}\{p_i \leq \alpha_i \text{ and } H_i \text{ is null}\}}{\text{# of rejections at } k} \]

\[ \widehat{\text{FDP}}_k = \frac{\sum_{i=1}^{k} \alpha_i}{\text{# of rejections at } k} \]

\textbf{Theorem:} \( \mathbb{E}[\text{FDP}_k] \leq \mathbb{E}[\widehat{\text{FDP}}_k] \)

\( \text{LORD maintains } \widehat{\text{FDP}}_k \leq \ell \text{ for all } k \in \mathbb{N} \)

\textbf{Theorem:} L ORD ensures FDR \( \leq \ell \)

(Ramdas et al. 2017, Javanmard and Montanari 2018.)
The estimator view of FDP (SupLORD)

\[ \Pr(\exists k \in \mathbb{N} : \text{FDP}_k > \overline{\text{FDP}}_k) \leq \delta \]  

(Katsevich and Ramdas 2021)

\( \overline{\text{FDP}} \) upper bounds FDP with high probability

\[ \overline{\text{FDP}}_k = \log \left( \frac{1}{\delta} \right) \cdot \frac{1 + \sum_{i=1}^{k} \alpha_i}{1 + \# \text{ of rejections at } k} \]

Theorem: \( \overline{\text{FDP}}_k \leq \epsilon \) for all \( k \geq K \) where \( H_k \) is rejected \( \iff \text{FDX}^\epsilon_K \leq \delta \)

SupLORD (our method) ensures this
SupLORD surpasses prior methods empirically

\[ \text{p-values from 1-sided z-test on i.i.d. Gaussians.} \]

\[ \delta = 0.05, \epsilon = 0.15 \quad \text{(LORDFDX, SupLORD)} \]

\[ \ell = 0.05 \quad \text{(Bonferroni, LORD)} \]

\[ \pi = \text{rate of non-nulls} \]

\[ \mu = \text{mean of non-nulls} \]

\[ \mu = 2 \quad \mu = 3 \]

\[ \pi = \text{rate of non-nulls} \]

Our method

Prior FDX controlling method based on LORD

<table>
<thead>
<tr>
<th>Method</th>
<th>Rejections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bonferroni</td>
<td>879</td>
</tr>
<tr>
<td>LORD</td>
<td>12352</td>
</tr>
<tr>
<td>LORDFDX</td>
<td>8</td>
</tr>
<tr>
<td>SupLORD</td>
<td>18793</td>
</tr>
</tbody>
</table>

Real world dataset of p-values for mouse phenotypes from IMPC
Dynamic allocation of alpha values

Wealth for SupLORD: \( W(k) = \max \left\{ c \in \mathbb{R} : \log \left( \frac{1}{\delta} \right) \cdot \frac{c + 1 + \sum_{i=1}^{k} \alpha_i}{1 + \# \text{ of rejections at } k} \leq \epsilon \right\} \)

How much can I spend before \( \overline{\text{FDP}} \) exceeds \( \epsilon \)?

SupLORD accumulates too much wealth!

unused wealth \( \Downarrow \)

smaller alpha values \( \Downarrow \)

unexploited power
Solution: use larger alpha values that are more uniform in size.

Generally larger alpha values

Wealth no longer increasing

Dynamic allocation outperforms other methods
Takeaways

1. FDX control can be achieved using a high probability estimator of the FDP.
2. Previous algorithms underutilized wealth by not spending it fast enough. We can increase power by using larger alpha values.
Thanks!