Dept. of Statistics and Data Science
Machine Learning Dept.
Carnegie Mellon University

Dynamic Algorithms for Online Multiple Testing

Ziyu Xu (Neil)

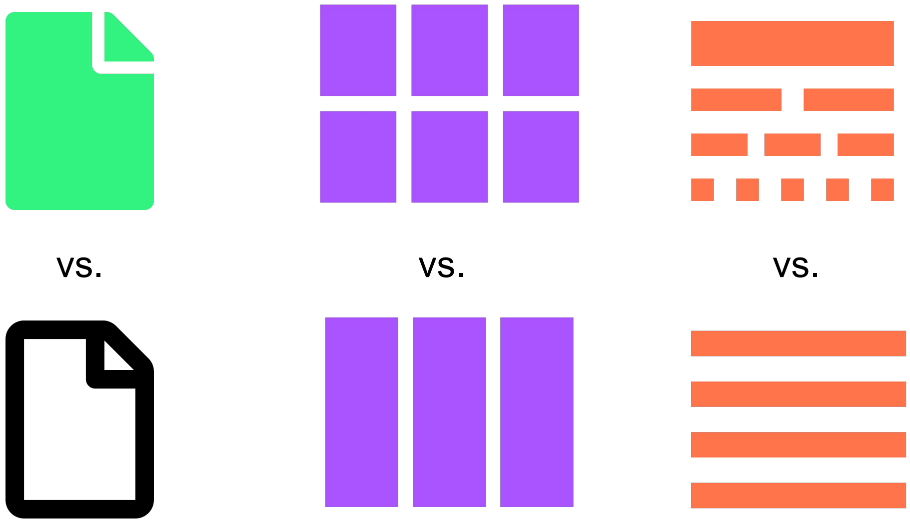
Aaditya Ramdas

Dept. of Statistics and Data Science
Machine Learning Dept.
Carnegie Mellon University

Dynamic Algorithms for Online Multiple Testing

Ziyu Xu (Neil)

Aaditya Ramdas



Color Amount Style

H_1 H_2 H_3

Problem: testing many different hypotheses

Our goal is to reject non-null hypotheses (make true discoveries)

Threshold for rejection output by algorithm

1st p-value resulting from experiment

2nd p-value

3rd p-value

$$p_1 \le \alpha_1 \qquad p_2 \le \alpha_2 \qquad p_3 \le \alpha_3$$

$$H_1 \qquad H_2 \qquad H_3 \qquad \ldots$$

infinite stream of p-values and hypotheses
=
"online"

(Foster and Stine 2008)

p-values are statistics supported on [0, 1] that "summarize evidence" for a hypothesis

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

p_k potentially small if H_k is non-null

$\boxed{\text{Smaller p-values} \Rightarrow \text{more evidence against the null}}$

If we reject a null hypothesis, we make a "false discovery"

Key error metrics we wish to keep controlled

$$\mathrm{FDP_k} = \frac{\text{\# of false discoveries by time } k}{\text{total \# of discoveries by time } k}$$

$$\mathrm{FDR} = \sup_{k \in \mathbb{N}} \mathbb{E}\left[\mathrm{FDP}_k\right]$$

error metric primarily considered in prior work

control at fixed times

expectation

$$\mathrm{FDX}_K^{\epsilon} = \mathbf{Pr}(\forall k \geq K : \mathrm{FDP}_k > \epsilon)$$

probabilistic bound

uniform over time

$$\mathrm{FDR} = \sup_{\tau \in \mathcal{T}} \mathbb{E}\left[\mathrm{FDP}_\tau\right]$$

Extend FDR to include data-dependent stopping times.

set of stopping times

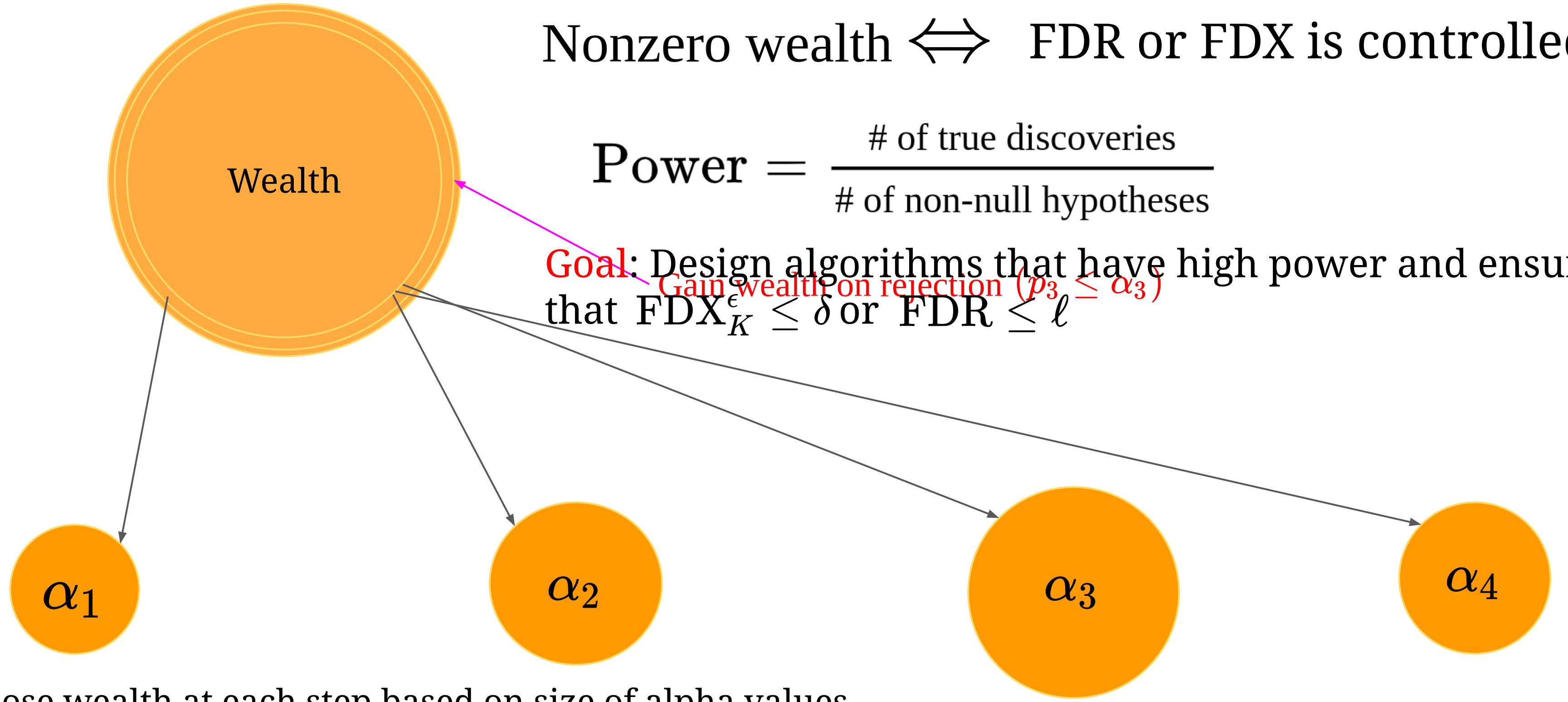
Our algorithm controls these

Alpha-investing: a method for selecting alpha values while maintaining error control

Nonzero wealth $\Longleftrightarrow$ 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 $\mathrm{FDX}_K^{\epsilon} \leq \delta$ or $\mathrm{FDR} \leq \ell$



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

$$\mathrm{FDP}_k = \frac{\sum_{i=1}^{k} \mathbf{1}\{p_i \le \alpha_i \text{ and } H_i \text{ is null}\}}{\text{\# of rejections at } k} \qquad \widehat{\mathrm{FDP}}_k = \frac{\sum_{i=1}^{k} \alpha_i}{\text{\# of rejections at } k}$$

Theorem: $\mathrm{FDP}_k \le \widehat{\mathrm{FDP}}_k$

LORD ensures $\widehat{\mathrm{FDP}}_k \le \ell$ for all $k \in \mathbb{N}$

Theorem: LORD ensures FDR $\le \ell$

(Ramdas et al. 2017, Javanmard and Montanari 2018.)

The estimator view of FDP

$\mathbf{Pr}(\exists k \in \mathbb{N} : \mathbf{FDP}_k > \overline{\mathbf{FDP}}_k) \leq \delta$ (Katsevich and Ramdas 2021)

$\overline{\mathrm{FDP}}$ upper bounds FDP with high probability

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

output alpha-value $\to$ increase $\overline{\mathrm{FDP}}$ $\to$ decrease wealth

make rejection $\to$ decrease $\overline{\mathrm{FDP}}$ $\to$ increase wealth

Theorem: $\overline{\mathrm{FDP}}_k \leq \epsilon$ for all $k \geq K$ where H_k is rejected $\Leftrightarrow \mathrm{FDX}_K^{\epsilon} \leq \delta$

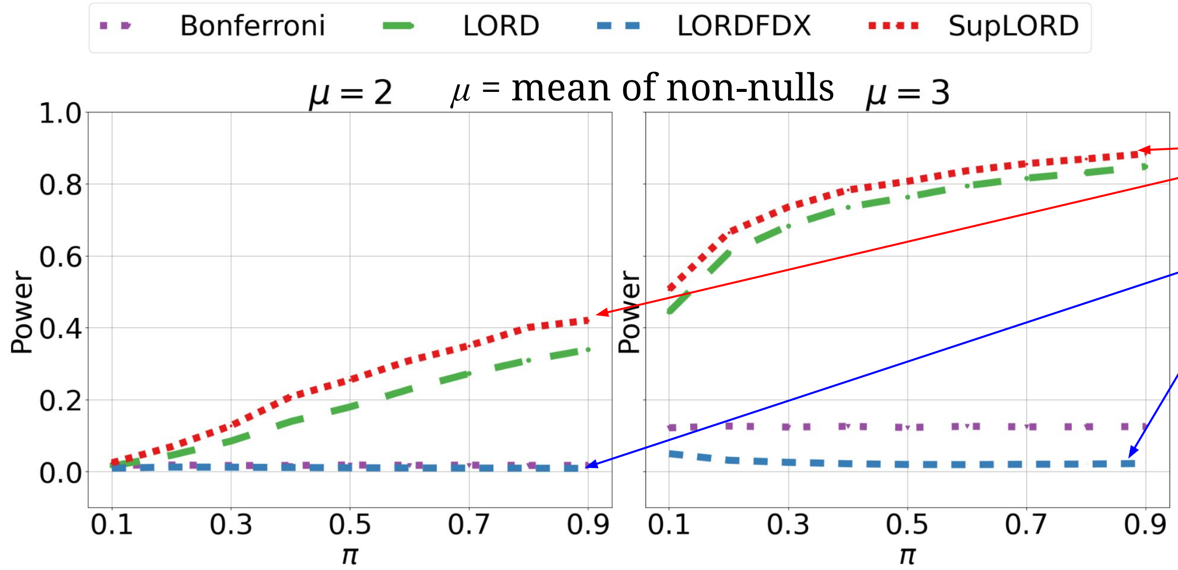
SupLORD (our method) ensures this

SupLORD surpasses prior methods empirically

p-values from 1-sided z-test on i.i.d. Gaussians.

$$\delta = 0.05, \epsilon = 0.15 \quad \text{(LORDFDX, SupLORD)}$$
$$\ell = 0.05 \quad \text{(Bonferroni, LORD)}$$

μ = mean of non-nulls



Our method

Prior FDX controlling method based on LORD

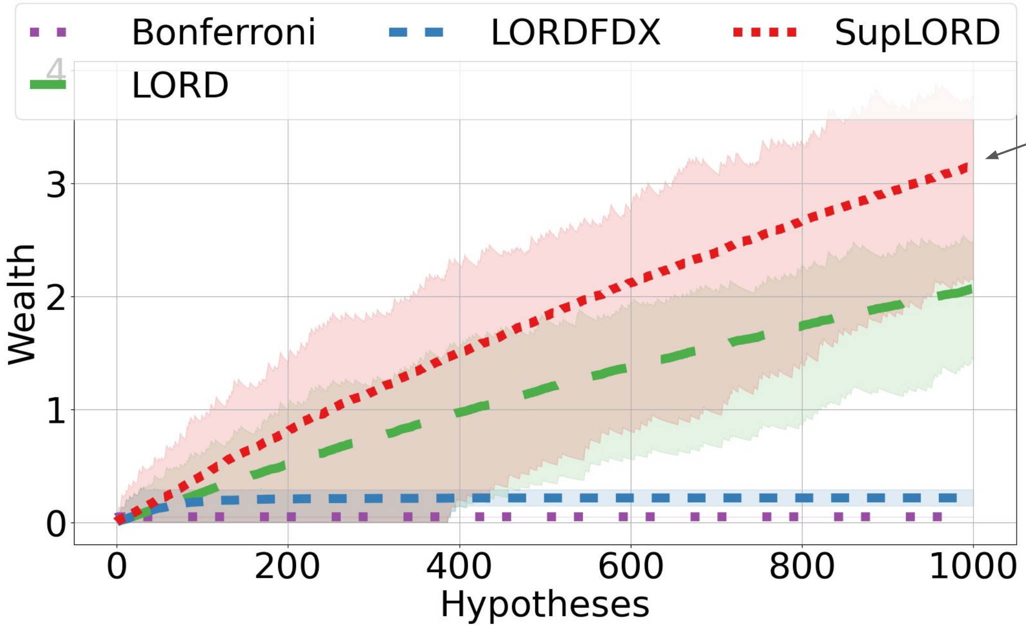
π = rate of non-nulls

Method	Rejections
Bonferroni	879
LORD	12352
LORDFDX	8
SupLORD	*18793*

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

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

How much can I spend before $\overline{\text{FDP}}$ exceeds ϵ ?



SupLORD accumulates too much wealth!

unused wealth

$\Downarrow$

smaller alpha values

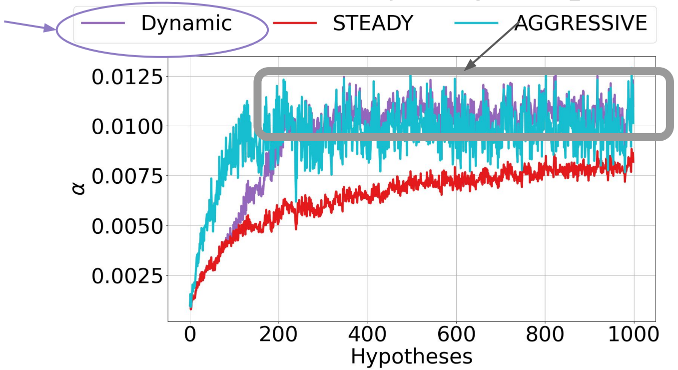
$\Downarrow$

unexploited power

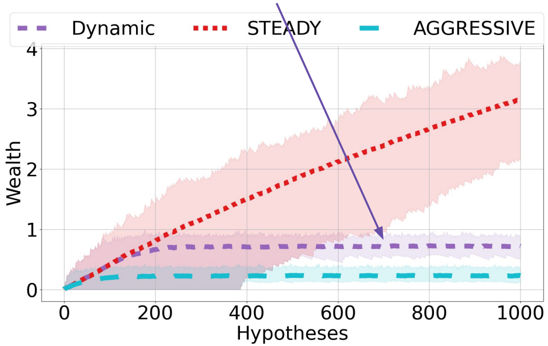
Solution: use larger alpha values that are more uniform in size.

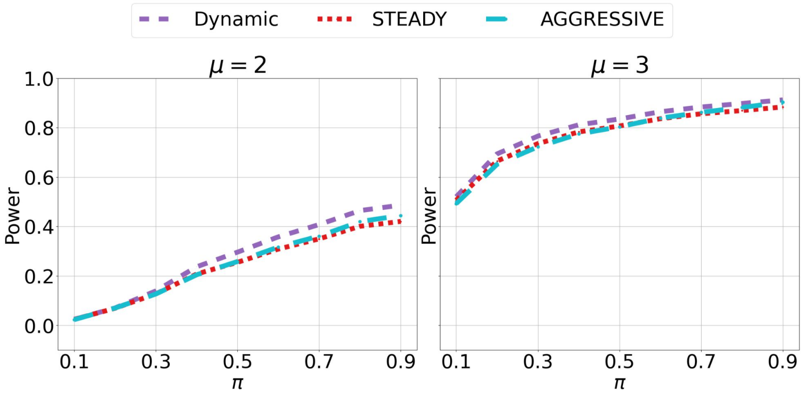
Ours

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 and we can simply use more of it to improve power.

Thanks!