Can Probabilistic Feedback Drive User Impacts in Online Platforms?

Jessica Dai*, Bailey Flanigan**, Nika Haghtalab*, Meena Jagadeesan*, Chara Podimata*** (alphabetical order)

*University of California, Berkeley **Carnegie Mellon University ***Massachusetts Institute of Technology*

arXiV

Recommendation systems rely on **feedback** from users to learn about their preferences over content.

Can some societal impacts of online platforms be attributed to **differential rates of feedback** across pieces of content?

Probabilistic feedback on an online platform: a simple model

Bandit algorithm	Platform's recommendation algorithm
K arms	K pieces of content items
ℓ_i: "loss" of an arm i	"quality/utility" of content i
$\boldsymbol{f_i}$: "feedback probability" of arm i	likelihood of observing ℓ_i when item i is recommended

For rounds t = 1, 2, … T:

- The algorithm picks one item i_t from the set of items [K] to recommend
- Incur loss ℓ_i
- With probability $\boldsymbol{f_i}$, observe the loss ℓ_i

Standard measure of performance is **regret**:

$$R(T) = \mathbb{E}\left[\sum_{t\in[T]} \ell_{i_t,t}\right] - \min_{j\in[K]}\mathbb{E}\left[\sum_{t\in[T]} \ell_{j,t}\right]$$

Formalizing "user impacts": APC & FOC

APC (Arm Pull Count) for i : $\mathbb{E}\left[\sum_{t\in[T]} \mathbf{1}[i_t = i]\right]$

"How often is content shown to users?"

FOC (Feedback Obs. Ct.) for i : $\mathbb{E}\left[\sum_{t\in[T]} \mathbf{1}[i_t = i] \cdot X_{i_t,t}\right]$

"How often do users give feedback?"

Monotonicity: does increasing an arm's $\boldsymbol{f_i}$ increase or decrease APC/FOC? Need a precise way to evaluate this.

- *Fix an instance $\mathfrak{I}$. Consider instance $\tilde{\mathfrak{I}}$, which is identical except for $\boldsymbol{f_i}$, which is increased on $\tilde{\mathfrak{I}}$.*
- *The algorithm is (e.g.)* ***positive monotonic in APC*** *if APC($\tilde{\mathfrak{I}}$) > APC($\mathfrak{I}$).*

Example: "own-group" content and APC

- $\boldsymbol{f_i}$ is higher for content that is produced by "similar" people (demographics, ideology)
- Positive monotonicity in APC means users see content from "similar" people more often – related to problems like "echo chambers"

Insights for platform design

- Identify relationships between content and feedback – and what kinds of monotonicities are desirable
- More generally, should formalize & track measures of performance beyond "loss"/utility; we do this for impact of probabilistic feedback

Three black-box transformations for all achievable monotonicity guarantees

For any no-regret (stochastic) bandit algorithm with regret R_{ALG}:

Transfor-mation	High-level idea	Regret	APC	FOC
BBDivide	Divide T into equally-sized blocks	$R_{ALG}\left(\frac{Tf^*}{\ln T}\right)\cdot\frac{\ln T}{f^*}$	$\approx$	$+$
BBPull	Pull the same arm until the first time feedback is observed	$R_{ALG}(T)\cdot\frac{1}{\min_i f_i}$	$\approx, -$	$\approx, +$
BBDivAdj	Pull each arm a prespecified number of times, increasing with $\boldsymbol{f_i}$	$R_{ALG}\left(\frac{Tf^*}{3\ln T}\right)\cdot\frac{6\ln T}{f^*}$	$\approx, +$	$\approx, +$

f^* is a tunable parameter between $(0, \min_i f_i]$. The $\approx$ symbol indicates approx. balance.

+ Improved regret for BBPull+AAE/UCB: $O\left(\sqrt{T\ln(T)\,\sum_{i\in[K]}\frac{1}{f_i}}\right)$

+ Strict monotonicity for BBPull+AAE and BBDA+AAE

Takeaway: wide range of monotonicity properties are achievable while preserving low regret!

3-Phase EXP3: adversarial losses + no-regret at the cost of monotonicity control

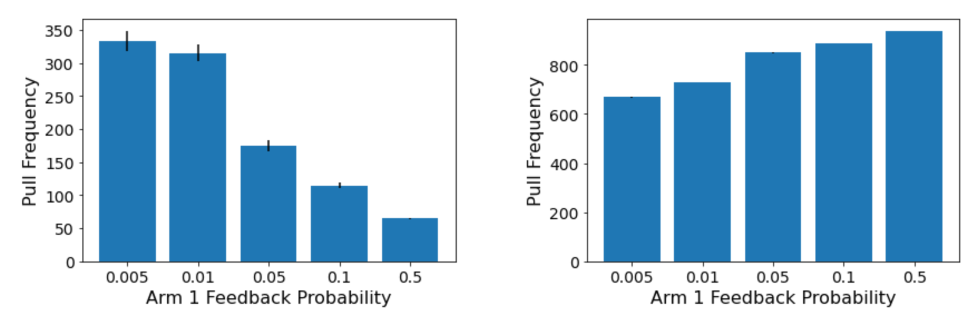
Standard EXP3 incurs linear regret when $\boldsymbol{f_i} \neq 1$.

3-phase EXP3 (below) achieves regret $O\left(\sqrt{T\ln K\,\sum_{i\in[K]}\frac{1}{f_i}}\right)$.

3-phase EXP3

Phase 1: Obtain high-probability estimate of $\boldsymbol{f_i}$
Phase 2: Obtain unbiased estimate of $\boldsymbol{f_i}$
Phase 3: Run standard EXP3, with hp est. to set learning rate and unbiased est. to create unbiased loss estimator

Improvement over previous work on MAB + feedback graphs: Esposito et.al. 2022 achieve $O\left(\sqrt{TK\min_i f_i}\right)$.



Lacks clean monotonicity properties. K=2, T=1000. Left: ℓ_1= 0.9, ℓ_2= 0.1. Right: ℓ_1= 0.1, ℓ_2= 0.9