Building Recommenders and Search Engines by Re-using User Feedback

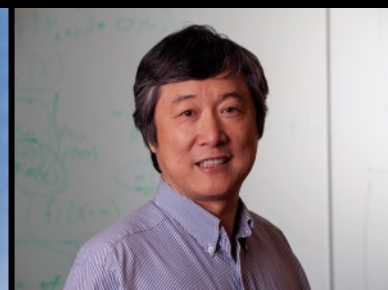
Adith Swaminathan

adswamin@microsoft.com

Joint work with Thorsten Joachims
and Tobias Schnabel (Cornell University)

Ack: NSF Grants

Bio



Counterfactual Evaluation and Learning

MSR - DLTC

Summary

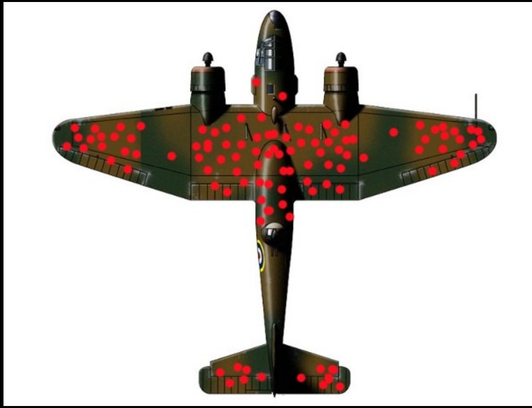
"Use logs collected from interactive systems to evaluate/train new interaction policies"

"Pay attention to feedback effects, and dis-entangle them" -- David

<u>Now:</u> Simple/pragmatic techniques to tackle biased user feedback

"Randomize cleverly to break confounding/feed back" -- Yisong

Wald's insight: What's missing?



- Where to add armor? Cover bullet-holes? (Survivor bias!)
- Beware: Confounding due to missing info

Overview

- "Use user ratings for collaborative filtering"
  - Project: MNAR (Schnabel et al, ICML 2016)

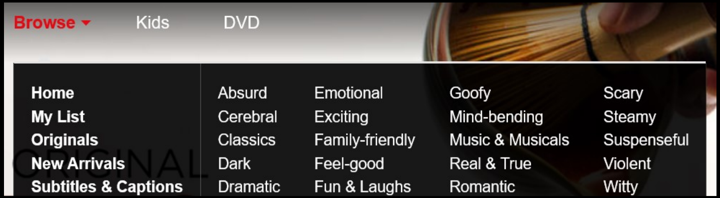
- "Use user clicks for search ranking"
  - Project: ULTR (Joachims et al, WSDM 2017)

Movie Recommendation

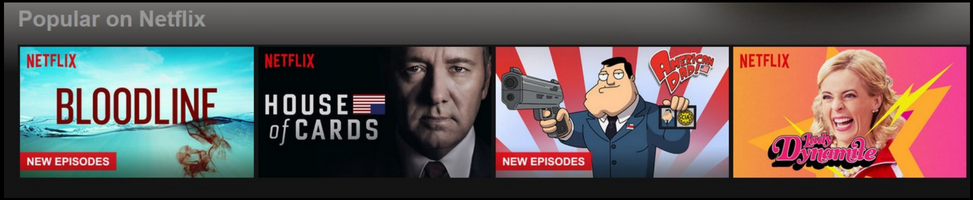
Example adapted from (Steck et al, 2010)

Selection Bias in Recommendations

- User-induced (e.g. browsing)

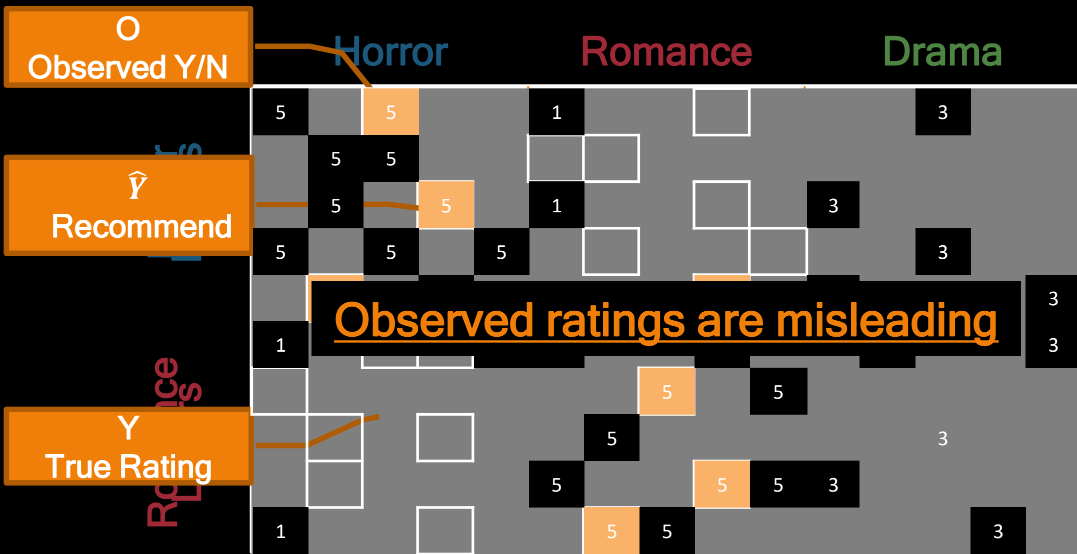


- System-induced (e.g. advertising)

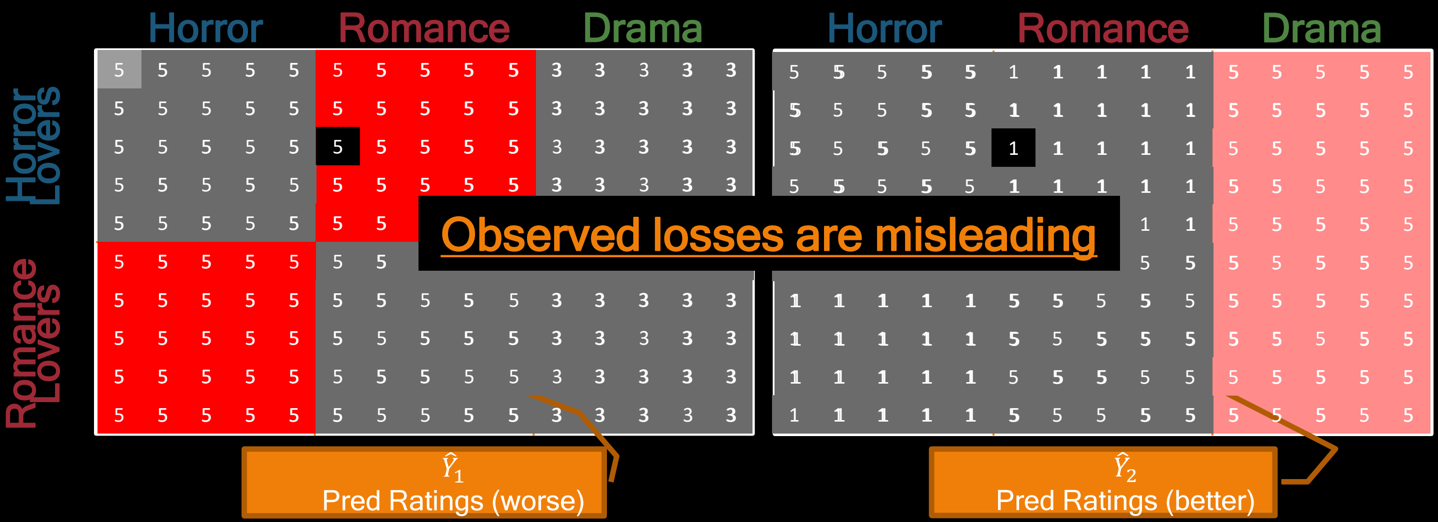


Question: What if we ignore these biases?

Evaluating recommendations under Selection Bias



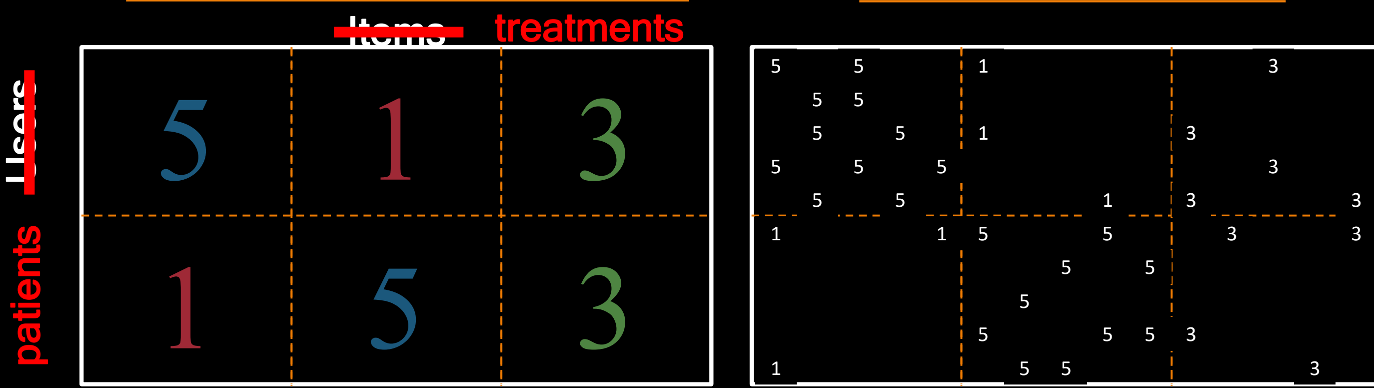
Evaluating rating predictions under Selection Bias



Recommendations as Treatments

Fix selection bias → potential outcomes framework

Counterfactual Outcomes Y | Factual Outcomes $\tilde{Y}$



⇒ Understand assignment mechanism

(Imbens & Ruben, 2015)

Assignment Mechanism for Recommendation

$$P_{u,i} = P\left(O_{u,i} = 1\right)$$

Inverse Propensity Scoring (IPS) is unbiased if $P_{u,i} > 0$:

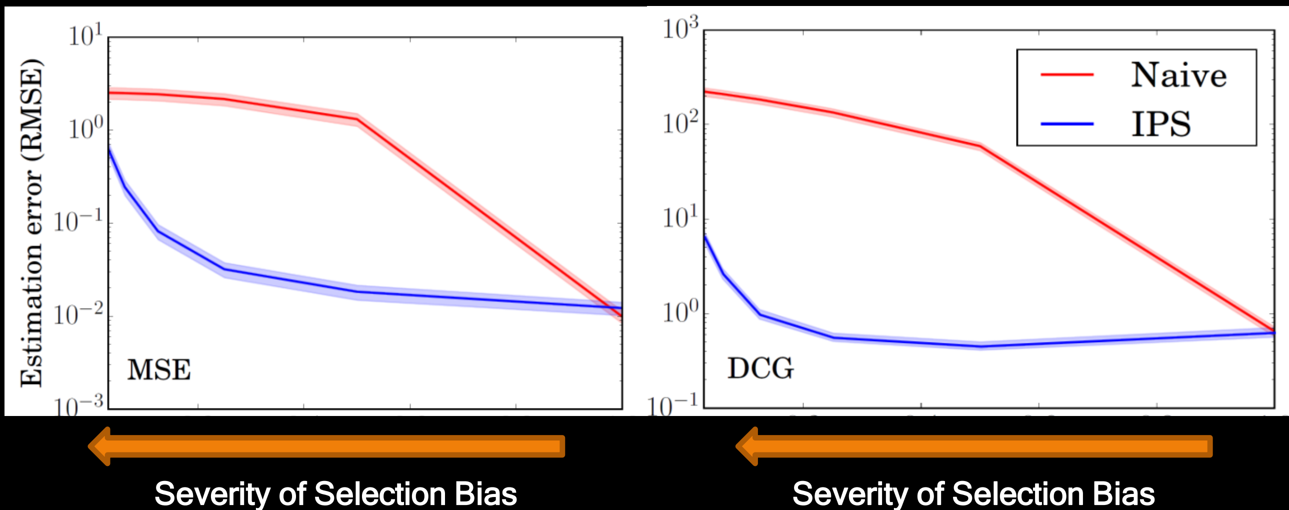
$$\hat{R}_{IPS} = \frac{1}{U \cdot I} \sum_{(u,i)} \frac{\mathbb{1}\{O_{ui}=1\}}{P_{u,i}} \left(Y_{u,i} - \hat{Y}_{u,i}\right)^2$$

Propensities P

Horror	Romance	Drama
p	$p/10$	$p/2$
$p/10$	p	$p/2$

(Horvitz & Thompson, 1952; Rosenbaum & Rubin, 1983; ...)

Debiasing Evaluation



IPS is robust to selection bias

Experimental vs. Observational

- Controlled Experiments
  - We control assignment mechanism (e.g. ad placement)
  - Propensities $P_{u,i} = P(O_{u,i} = 1)$ known [Just log propensities!]
  - Requirement: $P_{u,i} > 0$ (prob. assignment)

- Observational Study
  - Assignment mechanism not under our control (e.g. reviews/ratings)
  - Use features Z; $\hat{P}_{u,i} = P(O_{u,i} = 1 | Z)$ [Estimate propensity]
  - Requirement: $O_{u,i} \perp Y_{u,i} \mid Z$ (unconfounded)

Propensity Estimation

- Supervised Regression Problem

$$\hat{P}_{u,i} = P\left(O_{u,i} = 1 \mid Z\right)$$

- Off-the-shelf ML, e.g.,
  - Logistic regression
  - Naïve Bayes
  - Bernoulli Matrix Factorization
  - ...

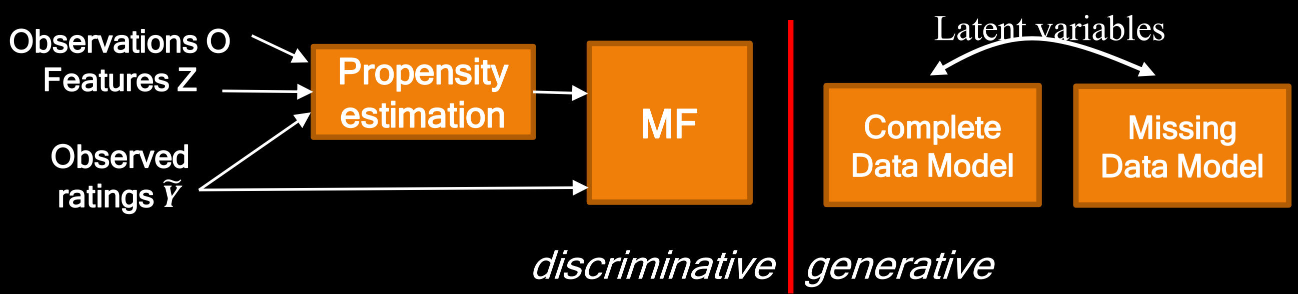
Observations O

Horror					Romance					Drama				
1	0	1	0	0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	1	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	1	0	1	0	0	0	0	1	0	1	0	0	0	1
1	0	0	0	1	1	0	0	1	0	0	1	0	0	1
0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	1	1	0	0	0	0
1	0	0	0	0	0	1	1	0	0	0	0	0	1	0

IPS is robust to inaccurate propensities

Debiased Collaborative Filtering

$$\hat{Y}^{ERM} = \underset{V,W}{\operatorname{argmin}} \left\{ \sum_{O_{u,i}=1} \frac{1}{P_{u,i}} \left(Y_{u,i} - V_u W_i\right)^2 + \lambda\left(\|V\|_F^2 + \|W\|_F^2\right) \right\}$$



(Marlin et al, 2007; Steck, 2011; ...)

Collaborative Filtering Results

- Two real-world MNAR datasets
  - YAHOO: Song ratings (15400 users; Marlin & Zemel, 2009)
  - COAT: Shopping ratings (300 users; new Schnabel et al, 2016)
- Report performance on MAR datasets

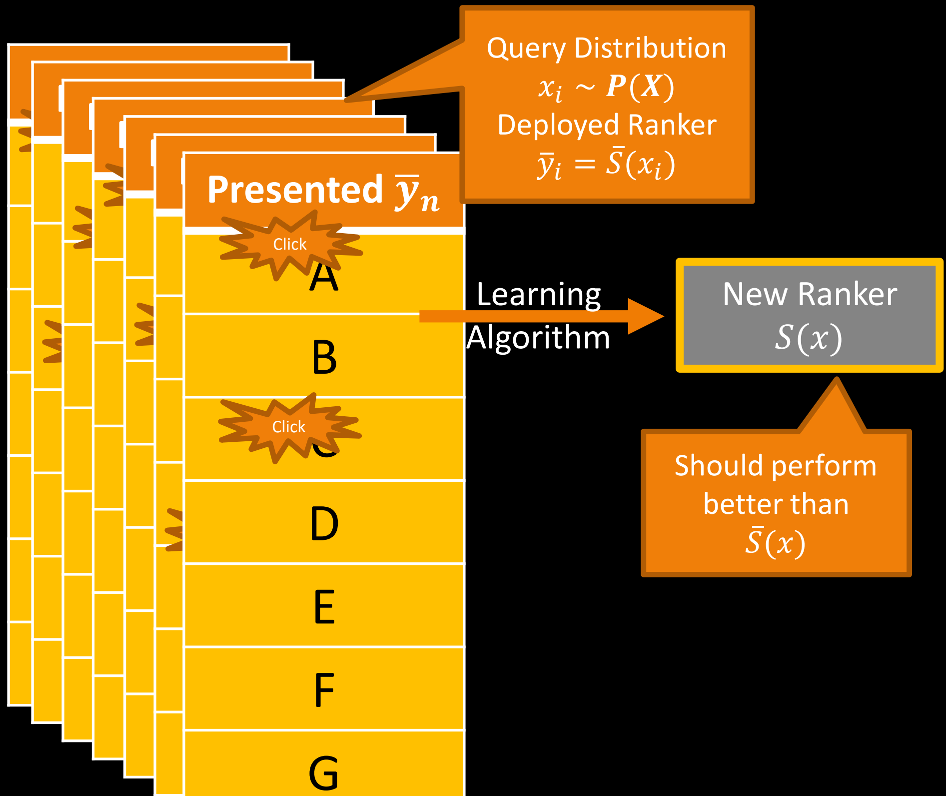
	YAHOO		COAT	
	MAE	MSE	MAE	MSE
MF-IPS	**0.810**	**0.989**	**0.860**	**1.093**
MF-Naive	1.154	1.891	0.920	1.202
HL MNAR	1.177	2.175	0.884	1.214
HL MAR	1.179	2.166	0.892	1.220

http://www.cs.cornell.edu/~schnabts/mnar/

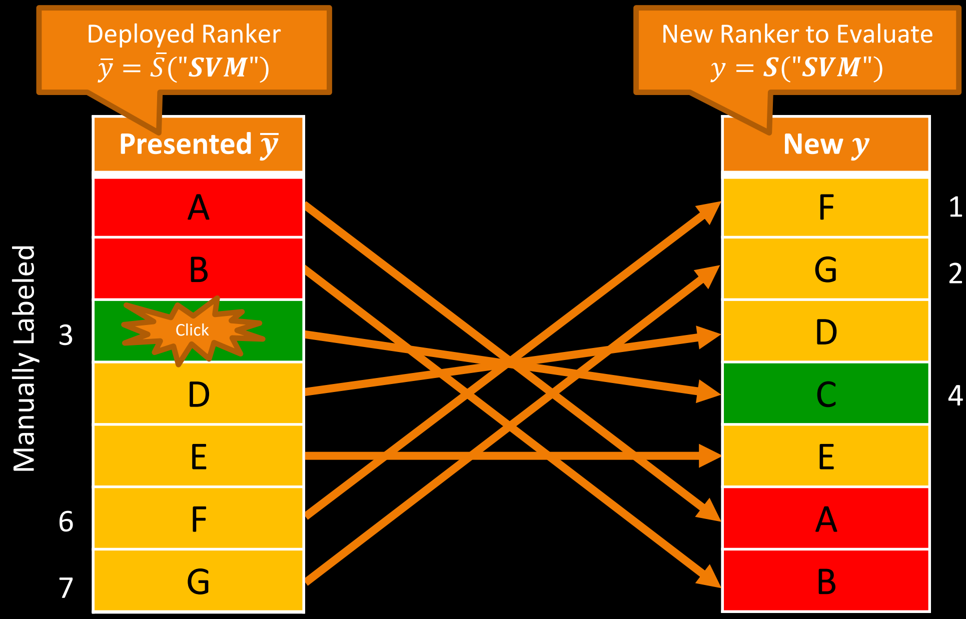
Overview

- "Use user ratings for collaborative filtering"
  - Project: MNAR (Schnabel et al, ICML 2016)
- "Use user clicks for search ranking"
  - Project: ULTR (Joachims et al, WSDM 2017)

Learning-to-Rank from Clicks



Evaluating Rankings



Evaluation with Missing Judgments

- Loss: $\Delta(y|r)$
  - Relevance labels $r_i \in \{0,1\}$
  - This talk: rank of relevant documents

$$\Delta(y|r) = \sum_i rank(i|y) \cdot r_i$$

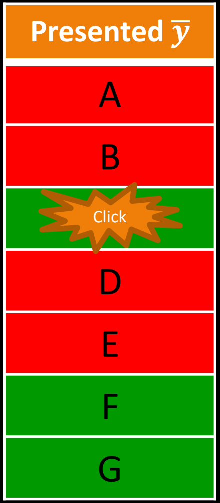
- Assume:
  - Click implies observed and relevant:

$$(c_i = 1) \leftrightarrow (o_i = 1) \wedge (r_i = 1)$$

- Problem:
  - No click can mean not relevant OR not observed

$$(c_i = 0) \leftrightarrow (o_i = 0) \vee (r_i = 0)$$

→ Understand observation mechanism



Inverse Propensity Score Estimator

- Observation Propensities $Q(o_i = 1|x, \bar{y}, r)$
  - Random variable $o_i \in \{0,1\}$ indicates whether relevance label r_i for is observed
- Inverse Propensity Score (IPS) Estimator:

$$\hat{\Delta}(y|r,o) = \sum_{i:c_i=1} \frac{rank(i|y)}{Q(o_i = 1|\bar{y}, r)}$$

New Ranking

- Unbiasedness: $E_o\left[\hat{\Delta}(y\,|\,r,o)\right] = \Delta(y|r)$

Presented $\bar{y}$	Q
A	1.0
B	0.8
C	0.5
D	0.2
E	0.2
F	0.2
G	0.1

[Horvitz & Thompson, 1952] [Rubin, 1983] [Zadrozny et al., 2003] [Langford, Li, 2009] [Swaminathan & Joachims, 2015]

ERM for Partial-Information LTR

- Unbiased Empirical Risk:

$$\hat{R}_{IPS}(S) = \frac{1}{N} \sum_{(x,\bar{y},c)\in S} \sum_{i:c_i=1} \frac{rank(i|y)}{Q(o_i = 1|\bar{y}, r)}$$

Consistent Estimator of True Error

- ERM Learning:

$$\hat{S} = \underset{S}{\operatorname{argmin}} \left[\hat{R}_{IPS}(S)\right]$$

Consistent ERM Learning

- Questions:
  - How do we optimize this empirical risk in a practical learning algorithm?
  - How do we define and estimate the propensity model $Q(o_i = 1|\bar{y}, r)$?

Propensity-Weighted SVM Rank

- Data: $S = \left(x_j, d_j, D_j, q_j\right)^n$

Query

Clicked

Others

Propensity

Optimizes convex upper bound on unbiased IPS risk estimate!

- Training QP:

$$w^* = \operatorname*{argmin}_{w,\xi \geq 0} \frac{1}{2} w \cdot w + \frac{C}{n} \sum_j \frac{1}{q_j} \sum_i \xi_j^i$$

$$\forall \bar{d}^i \in D_1 : w \cdot \left[\phi(x_1, d_1) - \phi\left(x_1, \bar{d}^i\right)\right] \geq 1 - \xi_1^i$$

$$\vdots$$

$$\forall \bar{d}^i \in D_n : w \cdot \left[\phi(x_n, d_n) - \phi\left(x_n, \bar{d}^i\right)\right] \geq 1 - \xi_n^i$$

- Loss Bound:

$$\forall w : rank(d, sort(w \cdot \phi(x, d))) \leq \sum_i \xi^i + 1$$

[Joachims et al., 2002]

Position-Based Propensity Model

- Model:

$$P\big(c_i = 1 | r_i, rank(i|\bar{y})\big) = q_{rank(i|\bar{y})} \cdot [r_i = 1]$$

- Assumptions
  - Examination only depends on rank
  - Click reveals relevance if rank is examined

Presented $\bar{y}$	Q
A	q_1
B	q_2
C	q_3
D	q_4
E	q_5
F	q_6
G	q_7

[Richardson et al., 2007] [Chuklin et al., 2015] [Wang et al., 2016]

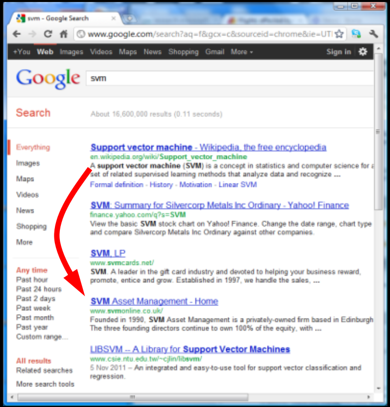
Estimating the Propensities

- Experiment:
  - Click rate at rank 1:

$$q_1 \cdot E\left(c_{S_1} = 1 \middle| o_{S_1} = 1\right)$$

- Intervention:
  - swap results at rank 1 and rank k
  - Click rate at rank k:

$$q_k \cdot E\left(c_{S_1} = 1 \middle| o_{S_1} = 1\right)$$



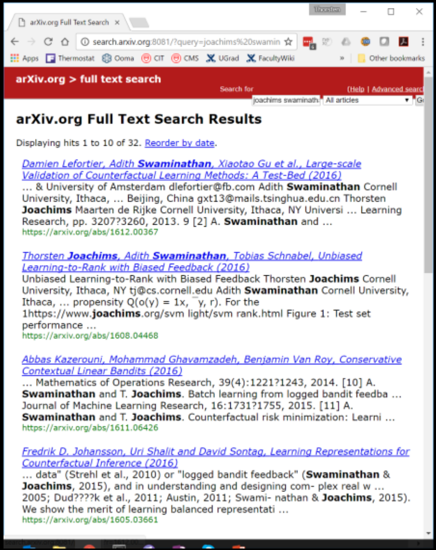
$$\Rightarrow \frac{q_1}{q_k} = \frac{Click\ rate\ at\ rank\ 1}{Click\ rate\ at\ rank\ k\ after\ swap}$$

[Langford et al., 2009; Wang et al., 2016]

Real-World Experiment

- Arxiv Full-Text Search
  - Run intervention experiment to estimate q_r
  - Collect training clicks using production ranker
  - Train naïve / propensity SVM-Rank (1000 features)
  - A/B tests via interleaving

Interleaving Experiment	Propensity SVM-Rank wins	Propensity SVM-Rank loses	Propensity SVM-Rank ties
against Prod	87	48	83
against Naive SVM-Rank	95	60	102



Overview

- "Use user ratings for collaborative filtering"
  - Project: MNAR (Schnabel et al, ICML 2016)
- "Use user clicks for search ranking"
  - Project: ULTR (Joachims et al, WSDM 2017)
- Discussion

Resources

- Randomized dataset:

http://www.cs.cornell.edu/~adith/Criteo/ [NIPS'16 workshop]

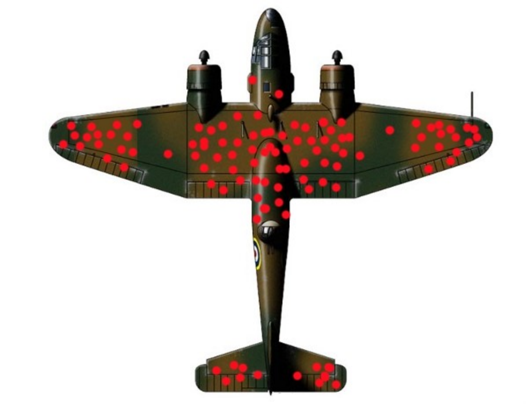
- Tutorial: Off-policy evaluation and optimization

http://www.cs.cornell.edu/~adith/CfactSIGIR2016 [SIGIR'16]

- Book: Causal Inference for Statistics, Social, and Biomedical Sciences, Imbens & Rubin, 2015.

- Many open questions!

Conclusion



Thanks!

Causality+ML

Simple/pragmatic techniques to tackle biased user feedback

adswamin@microsoft.com