Counterfactual Evaluation and Learning

Part 2

Adith Swaminathan, Thorsten Joachims

Department of Computer Science & Department of Information Science

Cornell University

Website: http://www.cs.cornell.edu/~adith/CfactSIGIR2016/

Funded in part through NSF Awards IIS-1247637, IIS-1217686, IIS-1513692.

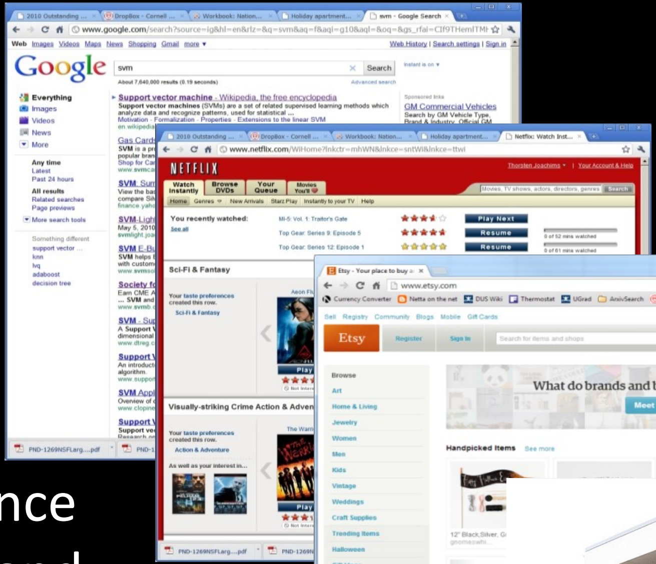
User Interactive Systems

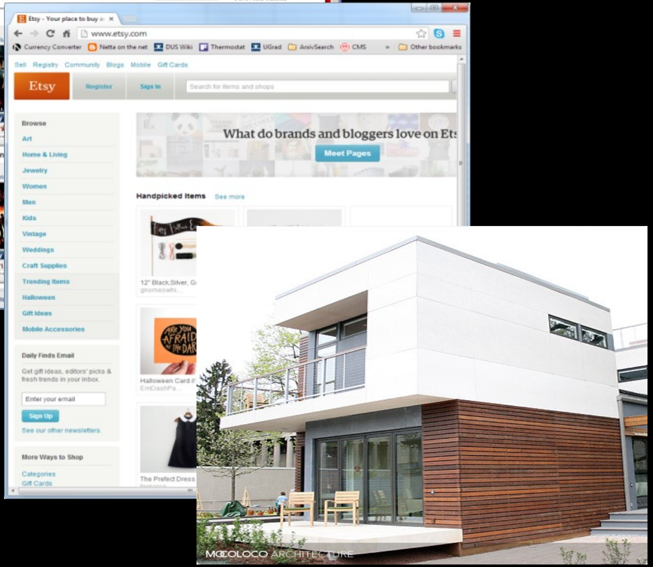
Examples

- Search engines
- Entertainment media
- E-commerce
- Smart homes, robots, etc.

→ Logs of User Behavior for

- Evaluating system performance
- Learning improved systems and gathering knowledge
- Personalization





Log Data from Interactive Systems

- Data

context　π_0 action　reward / loss　propensity

$$S = \left((x_1, y_1, \delta_1, p_1), \dots, (x_n, y_n, \delta_n, p_n)\right)$$

→ Partial Information (aka "Contextual Bandit") Feedback

- Properties
  - Contexts x_i drawn i.i.d. from unknown $P(X)$
  - Actions y_i selected by existing system $\pi_0(Y|X)$
  - Feedback δ_i from unknown function $\delta: X \times Y \to \Re$

[Zadrozny et al., 2003] [Strehl et al., 2010], [Bottou, et al., 2014]

Goals for this Tutorial

- Use interaction log data

$$S = \big((x_1, y_1, \delta_1, p_1), \ldots, (x_n, y_n, \delta_n, p_n)\big)$$

  for
  - Evaluation:
    - Estimate online measures of some system π offline.
    - System π is typically different from π_0 that generated log.
  - Learning:
    - Find new system π that improves performance.
    - Do not rely on interactive experiments like in online learning.

SIGIR 2016 Tutorial
Counterfactual Evaluation and Learning

PART 2: LEARNING

Learning: Outline

- Optimizing online metrics offline
- Approach 1: “Model the world”
  - Derive policy from predicted rewards
- Approach 2: “Model the bias”
  - ERM via IPS: Reduction to weighted multi-class classification
- Revisiting the variance issue
  - ERM via Slates: Modeling feedback for combinatorial actions
  - CRM via POEM: Variance regularized ERM for stochastic rules
  - CRM via Norm-POEM: Self-normalized IPS for equivariance
- Case study
- Summary & Code samples

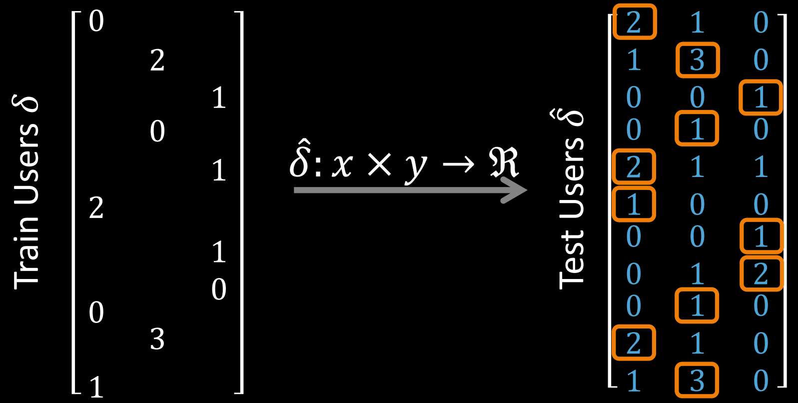
Goal of Learning

- Given:
  - Log data $S = \left((x_1, y_1, \delta_1, p_1), \dots, (x_n, y_n, \delta_n, p_n)\right)$
  - Hypothesis space H of possible policies π
- Find: Policy $\pi \in H$ that has maximum utility

$$\mathrm{U}(\pi) = \int \int \delta(x, y)\pi(y|x)P(x)\, dx\, dy$$

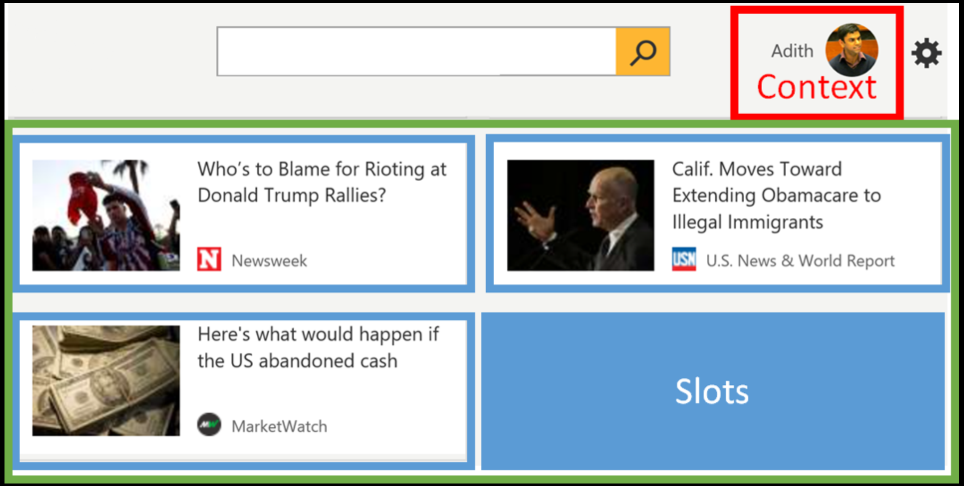
Approach "Model the World"
Reward Predictor

- Given:
  - Log $S = \left((x_1, y_1, \delta_1, p_1), \ldots, (x_n, y_n, \delta_n, p_n)\right)$ from π_0
  - Assumptions about reward model $\hat{\delta}: x \times y \rightarrow \Re$ (e.g., regression, click model)
- Algorithm:
  - Train reward predictor $\hat{\delta}: x \times y \rightarrow \Re$ using S
  - Derive policy $\hat{\pi}(x) \equiv \underset{y}{\operatorname{argmax}}\{\hat{\delta}(x, y)\}$



News Recommender: Exp Setup

- Context x: User profile
- Action y: Ranking
  - Pick from 7 candidates to place into 3 slots
- Reward δ: "Revenue"
  - Complicated hidden function



- Logging policy π_0: Non-uniform randomized logging system
  - Placket-Luce "explore around current production ranker" (see case study)

News Recommender: Results

- Reward Predictor:
  - Features: Stacked features of three articles
  - Regression method: selected best via CV from {Ridge, Lasso, Least Squares, Decision Trees}

Approach	True Revenue
Production ranker	224.00
Randomized π_0	214.00
Reward predictor	175.71

Issues with Reward Predictor

Issue 1:

- Model bias + selection bias = biased and not consistent

Can be remedied via propensity weighting
→ e.g. [Li et al., 2014] [Schnabel et al., 2016a].

Issue 2:

- First solves hard problem (reward prediction) in order to solve easier problem (find good policy)
  - Predict correct rewards $\longrightarrow$ optimal policy
  - Optimal policy $\not\longrightarrow$ predict correct rewards

Learning: Outline

- Optimizing online metrics offline
- Approach 1: “Model the world”
  - Derive policy from predicted rewards
- Approach 2: “Model the bias”
  - ERM via IPS: Reduction to weighted multi-class classification
- Revisiting the variance issue
  - ERM via Slates: Modeling feedback for combinatorial actions
  - CRM via POEM: Variance regularized ERM for stochastic rules
  - CRM via Norm-POEM: Self-normalized IPS for equivariance
- Case studies
- Summary & Code samples

Empirical Risk Minimization

Empirical Risk Minimization (ERM) with Regularization:

Given hypothesis space H of rules (or policies) $\pi: X \rightarrow Y$

$$\hat{\pi} = \underset{\pi \in H}{\operatorname{argmax}} \left[\widehat{U}(\pi) - Reg(\pi) \right]$$

→ SVMs, Neural Nets, Boosted Trees, etc

Questions for learning from log data:

- What estimator to use for $\widehat{U}(\pi)$?
- What regularizer $Reg(\pi)$ to use?
- Deterministic vs. Stochastic policies π?
- How to solve argmax?

ERM with IPS Estimator

- Given:
  - Log $S = \left((x_1, y_1, \delta_1, p_1), \ldots, (x_n, y_n, \delta_n, p_n)\right)$ from π_0
  - Deterministic prediction rules $\pi \in H$: $\mathrm{y} = \pi(x)$
- Training:

$$\hat{\pi} := \operatorname{argmax}_{\pi \in H} \left\{ \frac{1}{n} \sum_{i}^{n} \frac{I\{y_i = \pi(x_i)\}}{p_i} \delta_i \right\}$$

[Zadrozny et al., 2003] [Dudik et al., 2011], [Bottou, et al., 2014]

Deterministic π → Multi-class ERM

- Treat π as a classifier with weighted loss

$$(x, y, \delta, p) \rightarrow (x, y, w); w = {}^{\delta}/_{p}$$

- Policy utility is same as weighted accuracy!

$$U(\pi) = E_{x,y}[wI\{\pi(x) = y\}]$$

➔Use weighted multi-class algorithms to pick π.
Implemented in Vowpal Wabbit
https://github.com/JohnLangford/vowpal_wabbit/wiki

[Zadrozny et al., 2003]

Summary: ERM via IPS

- Empirical Risk Minimization (ERM) with Regularization:
  - What estimator to use for $\widehat{U}(\pi)$?
    - VW: IPS or Doubly Robust
  - What regularizer $Reg(\pi)$ to use?
    - Standard regularizers to prevent overfitting
  - Deterministic vs. stochastic π?
    - Deterministic
  - How to solve argmax?
    - Reduce to multi-class classification, use off-the-shelf algos

News Recommender: Results

- VW: Reduce to multi-class filter tree, doubly robust estimator with ridge regression, default parameters, 4 epochs via CV

Approach	Revenue
Production ranker	224.00
Randomized π_0	214.00
Reward predictor	175.71
ERM via IPS (VW)	177.93

Adith takes over