CSE 8803RS: Recommendation Systems

Lecture 19: Cold-Start CF and Contextual Bandit Problems

Hongyuan Zha

School of Computational Science & Engineering
College of Computing
Georgia Institute of Technology

Cold-Start Problems in CF

- New users
  — making use of meta-features: e.g., regression prior
  — interview process: the set of items relatively stable, user-centric
    - Static sets of interview questions
    - Adaptive interview questions
    - Binary vs. sequential search
- New items
    - Making use of meta-features: e.g., regression prior
    - The set of items changes rapidly: Yahoo!'s Today Module
      — Little historical data per user
    - Computational advertising: placing sets of ads on web pages
    - System-centric: maximizing click-through rate (CTR)
- Several angles to consider the trade-off/problem formulation

Yahoo!'s Today Module

Contextual Multi-Armed Bandit Problem

- To draw users' attention, rank article in the pool, and highlight the most attractive in the story position (F_1)
- Each user visits and their click probabilities on articles *iid*
- Articles in the pool $\Leftrightarrow$ arms
- The payoff is one if F_1 is clicked, otherwise zero
  — the expected payoff is the click through rate (CTR)
- User/article features help to select the article

Multi-Armed Bandit Problem

Context-free bandit problem

- Proposed by H. Robbins (1952)
  — together with T.L. Lai proved logrithmic low-bound for expected regret (1985)
- Similar to a slot machine (one-armed bandit) but with K levers (arms)
- When pulled, each lever provides a reward drawn from a distribution specific to that lever
- Initially, we know little of the levers
  — through repeated trials, we can eventually focus on the most rewarding lever
- Trade-off: Exploitation vs. exploration

Multi-Armed Bandit Problem

- $\mathcal{A} = \{1, 2, \ldots, K\}$ the set of arms
- Multi-armed bandit algorithm A proceeds in discrete trials $t = 1, 2, \ldots$. In trial t:
  1. Associated with each arm a is a real-valued payoff/reward $r_{t,a}$ — $[r_{t,1}, \ldots, r_{t,K}] \sim \mathcal{D}$
  2. Based on observed payoffs in previous trials, A chooses $a_t \in \mathcal{A}$ and receive payoff r_{t,a_t}
  3. The new information (a_t, r_{t,a_t}) is incorporated into A's arm-selection strategy

Some Definitions

- *Total T-trial payoff* of algorithm/strategy A

$$G_A(T) \equiv \mathcal{E}_{\mathcal{D}} \left\{ \sum_{t=1}^{T} r_{t,a_t} \right\}$$

- Let $\mu^* = \max_{1 \le a \le K} \mu_a, \mu_a \equiv \mathcal{E}_{\mathcal{D}} r_a$.
- T-trial regret of A,

$$R_A(T) \equiv T\mu^* - G_A(T)$$

- Per-trial payoff $g_A(T) \equiv G_A(T)/T$ and per-trial regret $\rho_A(T) \equiv R_A(T)/T$
- Zero-regret algorithm

$$Pr(\rho_A(T) \to 0) \to 1, \quad T \to \infty$$

Greedy Algorithm

- Suppose at trial t, arm a has been chosen k_a times with payoff $r^{(1)}, \ldots, r^{(k_a)}$, use

$$Q_t(a) = \frac{1}{k_a}(r^{(1)} + \cdots + r^{(k_a)})$$

  to estimate μ_a, the mean of r_a
- Choose a_t at trial t if

$$Q_t(a_t) = \max_{1 \le a \le K} Q_t(a)$$

ϵ-Greedy Algorithm

- Behave greedily most of the time, but once in a while, say with probability ϵ, randomly select one arm *a*
  — with probability $(1 - \epsilon)$ choose the greedy arm
  — with probability ϵ, randomly select one arm *a*
- Balance/trade-off between exploration and exploitation
  — Exploit the past experience to select the arm that appears to be the best
  — Explore by choosing seemingly sub-optimal arms to gather more information about the arms

Upper Confidence Bound (UBC) Algorithm

- ϵ-greedy algorithm $\Rightarrow$ *unguided* exploration
- Estimate $Q_t(a)$ as well as a *confidence interval*, with high probability

$$|Q_t(a) - \mu_a| \le c_{t,a}$$

- Choose a_t at trial t if

$$a_t = \mathrm{argmax}_{1 \le a \le K} \; (Q_t(a) + c_{t,a})$$

Contextual Bandit Problem

- $\mathcal{A} = \{1, 2, \ldots, K\}$ the set of arms
- Multi-armed bandit algorithm A interact with the world in discrete trials $t = 1, 2, \ldots$. In trial t:
  1. The world chooses a feature vector x_t. Associated with each arm a is a real-valued payoff/reward $r_{t,a}$
     — $[x_t, r_{t,1}, \ldots, r_{t,K}] \sim \mathcal{D}$
  2. Based on observed payoffs in previous trials h_{t-1} and x_t, A chooses $a_t \in \mathcal{A}$ and receive payoff r_{t,a_t}
  3. The new information $h_t = h_{t-1} \cup (x_t, a_t, r_{t,a_t})$ is incorporated into A's arm-selection strategy

Contexts

For Yahoo! Today Module: news article recommendation

- Users and articles can be represented by features
  — users: demographic features, historical activities
  — articles: BOW, category labels
- Contextual Bandit: bandits with co-variates, side information etc.
- Many algorithms proposed in the past

LinUBC Algorithm

- Linearity assumption (Auer 2002, JMLR):

$$\mathcal{E}(r_a|x_a) = x_a^T \theta_a, \quad a = 1, \ldots, K, \qquad x_t = [x_{t,1}, \ldots, x_{t,K}]$$

- For a, let m be the number of times arm a was selected *before* trial t. Collect data $[D_a, b_a] \in R^{m\times(d+1)}$, where D_a the m d-dimensional feature vectors, and b_a the corresponding payoffs
- Linear/ridge regression problem to estimate θ_a,

$$\hat{\theta}_a = (D_a^T D_a + I_d)^{-1} D_a^T b_a$$

LinUBC Algorithm

- Confidence Bound: with probability $> 1 - \delta$,

$$|x_{t,a}^T \hat{\theta}_a - \mathcal{E}(r_{t,a}|x_{t,a})| \leq \alpha \left(x_{t,a}^T A_a^{-1} x_{t,a} \right)^{1/2}$$

  here $A_a = D_a^T D_a + I_d, \alpha = 1 + (\log(2/\delta)/2)^{1/2}$

- LinUBC:

$$a_t = \mathrm{argmax}_{1 \leq a \leq K} \left(x_{t,a}^T \hat{\theta}_a + \alpha \left(x_{t,a}^T A_a^{-1} x_{t,a} \right)^{1/2} \right)$$

LinUBC

Algorithm 1 LinUCB with disjoint linear models.

```
Inputs: $\alpha \in \mathbb{R}_+$
for $t = 1, 2, 3, \dots, T$ do
    Observe features of all arms $a \in \mathcal{A}_t$: $\mathbf{x}_{t,a} \in \mathbb{R}^d$
    for all $a \in \mathcal{A}_t$ do
        if $a$ is new then
            $\mathbf{A}_a \leftarrow \mathbf{I}_d$ ($d$-dimensional identity matrix)
            $\mathbf{b}_a \leftarrow \mathbf{0}_{d \times 1}$ ($d$-dimensional zero vector)
        end if
        $\hat{\boldsymbol{\theta}}_a \leftarrow \mathbf{A}_a^{-1}\mathbf{b}_a$
        $p_{t,a} \leftarrow \hat{\boldsymbol{\theta}}_a^\top \mathbf{x}_{t,a} + \alpha\sqrt{\mathbf{x}_{t,a}^\top \mathbf{A}_a^{-1} \mathbf{x}_{t,a}}$
    end for
    Choose arm $a_t = \arg\max_{a \in \mathcal{A}_t} p_{t,a}$ with ties broken arbitrarily, and observe a real-valued payoff $r_t$
    $\mathbf{A}_{a_t} \leftarrow \mathbf{A}_{a_t} + \mathbf{x}_{t,a_t}\mathbf{x}_{t,a_t}^\top$
    $\mathbf{b}_{a_t} \leftarrow \mathbf{b}_{a_t} + r_t \mathbf{x}_{t,a_t}$
end for
```

LinUBC with Hybrid Linear Models

- LinUBC: training of different arms are separate
- $x_t = [x_{t,1}, \dots, x_{t,K}]$ is supposed to capture the context which involves both users and article
  — set aside part of the paramters that are common to all arms

$$\mathcal{E}(r_a | x_a) = z_a^T \beta + x_a^T \theta_a$$

- In the case of two arms, the regression problem is

$$\begin{bmatrix} Z_1 & D_1 & 0 \\ Z_2 & 0 & D_2 \end{bmatrix} \begin{bmatrix} \beta \\ \theta_1 \\ \theta_2 \end{bmatrix} \approx \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

Experiments

- Data collection: random bucket in May 2009
  — view-based randomization
  — F_1 article used
  — 4.7M events on May 1 (training/tuning)
  — 36M events on May 3-9 for testing
- Each event:
  — the random article shown to the user at F_1
  — user/article information
  — user click (yes/no) at F_1

Features

- User features: 1193 categorical features
  — demographic info: gender; age
  — geographic info: 200 metro and US states
  — behavioral categories: consumption history within Yahoo! properties
- Article features: 83 categorical features
  — URL categories
  — Editor topic categories

Features: Dimension Reduction

- Fit a bilinear logistic regression model to CTR: $\phi_u^T W \phi_a$
- With the weight matrix $W = C^T D$, $C\phi_u$ and $D\phi_a$ can be considered as the k-dimensional projected features
- Quantizing the k-dimensional features using K-means
  — each user and article is represented by a five dimensional vector, degree of membership to each of the five clusters
  — add 1 to each feature vector
  — $z_{t,a}$ outer-product of the user and article features, and $x_{t,a}$ article features alone

Offline Evaluation

- Logged data available for a different policy/algorithm
  — off-policy evaluation in reinforcement learning
- Offline data: $\mathcal{S}$, a stream of events where arms are selected uniformly at random

Algorithm 1 `Policy_Evaluator` (with infinite data stream).

Inputs: $T > 0$; bandit algorithm A; stream of events S
$h_0 \leftarrow \emptyset$ {An initially empty history}
$\hat{G}_{\mathsf{A}} \leftarrow 0$ {An initially zero total payoff}
for $t = 1, 2, 3, \ldots, T$ **do**
  repeat
    Get next event $(\mathbf{x}, a, r_a)$ from S
  until $\mathsf{A}(h_{t-1}, \mathbf{x}) = a$
  $h_t \leftarrow \textsc{concatenate}(h_{t-1}, (\mathbf{x}, a, r_a))$
  $\hat{G}_{\mathsf{A}} \leftarrow \hat{G}_{\mathsf{A}} + r_a$
end for
Output: $\hat{G}_{\mathsf{A}}/T$