CS 6780 Research Project: Multi-armed Bandits with Dependent Arms

Bangrui Chen bc496@cornell.edu
Saul Toscano Palmerin st684@cornell.edu
Zhengdi Shen zs267@cornell.edu

1. Introduction

Exploration vs Exploitation has been studied extensively under the framework of multi-armed bandits problem. The multi-armed bandits problem was first studied by Robbins (Herbert, 1952). The problem is trying to find the optimal policy so that we can pull arms sequentially while maximizing the total expected reward. For the case each arm is independent, the Gittins Index (Gittins, 1979) provides an optimal policy for maximizing the expected reward. Further, Lai and Robbins (Lai, 1985) proved the regret under an arbitrary policy increases linearly with number of arms. Most policies that assume independence require each arm to be tried at least once, which are impractical in settings involving many or infinite arms.

Since then, a lot of research has been focused on the dependent case. In (Gabor Bartok, 2012) and (Chih-Chun Wang, 2004), they studied the multi-bandit problem with side information. In (Yasin Abbasi-Yadkori, 2009) and (Yasin Abbasi-Yadkori, 2011), people consider the reward for each bandit is the inner product between an single unknown parameter and the bandit. In (Paat Rusmevichientong, 2010), they modeled the expected reward of each arm linearly depends on a multivariate random variable. Furthermore in that paper, they showed that under arbitrary policy, the Bayes regret is $O(r\sqrt{T})$ where r is the dimensional of the multivariate random variable. They also provided an simple algorithm that reaches the $O(r\sqrt{T})$ regret bound.

Through PEGE has the optimal regret bound, it doesn't do well in practice. In this project, we modified the PEGE algorithm and proved the modified algorithms are still optimal in the sense their Bayes regret is still $O(r\sqrt{T})$. We perform numerical experiments using the Yelp academic data to compare our modified algorithms as well as the original PEGE algorithm. The results suggest our modified algorithm are much better. Further, we compare our modified algorithms with UCB and Exponential Gradient algorithm.

There are lots of application for this problem. For instance, a newly registered user for a website with no background information will typically face a cold start problem. Same thing would happen to a investment company who wants to invest money into emerging industries. Furthermore, this problem is closely related to the recommender systems, which are a subclass of information filtering system that seek to predict the "rating" or "preference" that user would give to an item. In these applications, one natural question would be how can we quickly learn user's preference from their feedback? All of these problems can be modeled as a multi-armed bandit problem and our heuristic policies can be applied.

2. Problem Formulation

We have a finite set $\mathcal{U}_r = \{\mathbf{u}_1, \cdots, \mathbf{u}_m\} \subset \mathbb{R}^r$ that corresponds to the set of arms, where $r \geq 2$. For any time $t = 1, 2, \cdots, T$, we are asked to pick one arm X_t. The reward Y_t of playing arm $X_t \in \mathcal{U}_r$ in period t is given by

$$Y_t = \theta \cdot X_t + \epsilon_t,$$

where $\epsilon_t \sim N(0, \sigma^2)$ is the measurement error with σ known. Here θ is an unknown random vector, which is drawn from a multivariate normal distribution with mean μ and variance Σ. We further assume μ and Σ are known.

For a fixed time period T, the goal of this problem is to find a strategy π to maximize the following expression

$$E^{\pi}\left[\sum_{t=1}^{T} Y_t\right]. \tag{1}$$

Or equivalently, we are trying to find a policy that can minimize the Bayes risk under π:

$$\text{Risk}(T, \pi) = E\left[\text{Regret}(\theta, T, \pi)\right], \tag{2}$$

where the cumulative regret is defined as the following:

$$\text{Regret}(\theta_0, T, \pi) = \sum_{t=1}^{T} E\left[\max_{X \in \mathcal{U}_r} X \cdot \theta_0 - X_t \cdot \theta_0 | \theta = \theta_0\right]. \tag{3}$$

3. PEGE

We are interested in the multi armed bandits problem with correlated arms. Theoretically this problem can be solved using dynamic programming, however it usually suffers from the curse of dimensionality when the dimension of the arm is high. Thus, it is desirable to develop some computational feasible heuristic algorithms.

In (Paat Rusmevichientong, 2010), they proved the following theorem which gives a lower bound for the bayes risk:

Theorem 1. *(Lower Bound) Consider a bandit problem where the set of arms is the unit sphere in $\mathbb{R}^r$, and ϵ_t has a standard normal distribution with mean 0 and variance one for all t and X_t. If θ has a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{I}_r/r$, then for all policies π and every $T \geq r^2$,*

$$\mathit{Risk}(T, \pi) \geq 0.006 r\sqrt{T}.$$

In their paper, they also provided a simple algorithm called Phased Exploration and Greedy Exploitation (PEGE) (Algorithm 3) that reaches the corresponding lower bound, i.e the bayes risk for PEGE is $O(r\sqrt{T})$. In order for the algorithm to work, it requires the following two assumptions:

Assumption 1.

- *There exists a positive constant σ_0 such that for any $r \geq 2$, $\boldsymbol{u} \in \mathcal{U}_r$, $t \geq 1$ and $x \in \mathbb{R}$, we have $E[e^{x\epsilon_t}] \leq e^{\frac{x^2\sigma_0^2}{2}}$.*
- *There exists positive constants $\bar{u}$ and λ_0 such that for any $r \geq 2$,*

  $$\max_{u \in \mathbb{U}^r} \|\boldsymbol{u}\| \leq \bar{u}$$

  and the set of arms $\mathcal{U}_r \subset \mathbb{R}^r$ has r linearly independent elements $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_r$ such that $\lambda_{\min}(\sum_{k=1}^r \boldsymbol{b}_k \boldsymbol{b}_k^{'}) \geq \lambda_0$.

Assumption 2. *We say that a set of arms $\mathcal{U}_r$ satisfies the smooth best arm response with parameter J (SBAR(J), for short) condition if for any nonzero vector $\boldsymbol{z} \in \mathbb{R}^r \setminus \{\boldsymbol{0}\}$, there is a unique set best arm $\boldsymbol{u}^*(\boldsymbol{z}) \in \mathcal{U}_r$ that gives the maximum expected reward, and for any two unit vectors $\boldsymbol{z} \in \mathbb{R}^r$ an $\boldsymbol{y} \in \mathbb{R}^r$ with $\|\boldsymbol{z}\| = \|\boldsymbol{y}\| = 1$, we have*

$$\|\boldsymbol{u}^*(\boldsymbol{z}) - \boldsymbol{u}^*(\boldsymbol{y})\| \leq J\|\boldsymbol{z} - \boldsymbol{y}\|$$

Algorithm 1 Phased Exploration and Greedy Exploitation

Description: For each cycle $c \geq 1$, complete the following two phases:

1. **Exploration (r periods)** For $k = 1, 2, \cdots, r$, play arm $\mathbf{b}_k \in \mathcal{U}_r$ given in Assumption 1(b), and observe the reward $Y^{b_k}(c)$. Compute the OLS estimate $\hat{\theta}(c) \in \mathbb{R}^r$, given by

   $$\begin{aligned}\hat{\theta}(c) &= \frac{1}{c}(\sum_{k=1}^{r} \mathbf{b}_k \mathbf{b}_k^{'})^{-1} \sum_{s=1}^{c}\sum_{k=1}^{r} \mathbf{b}_k Y^{b_k}(s) \\ &= \mathbf{Z} + \frac{1}{c}(\sum_{k=1}^{r} \mathbf{b}_k \mathbf{b}_k^{'})^{-1} \sum_{s=1}^{c}\sum_{k=1}^{r} \mathbf{b}_k \epsilon^{b_k}(s)\end{aligned}$$

   where for any k, $Y^{b_k}(s)$, and $\epsilon^{b_k(s)}$ denote the observed reward and the error random variable associated with playing arm $\mathbf{b}_k$ in cycle s.

2. **Exploitation (c periods)** Play the greedy arm $\mathbf{G}(c) = \arg\max_{v \in \mathcal{U}^r} \mathbf{v}^{'}\hat{\theta}(c)$ for c periods.

Under these two conditions, the Bayes risk for the PEGE algorithm is at most $O(r\sqrt{T})$.

Theorem 2. *Suppose that assumption 1 holds and that the set $\mathcal{U}_r$ satisfy the SBAR(J) condition. In addition, there exists a constant $M > 0$ such that for every $r \geq 2$ we have $E[\|\theta\|] \leq M$ and $E[1/\|Z\|] \leq M$. Then, there exist a positive constant a_1 that depends only on σ_0, $\bar{u}$, λ_0, J and M, such that for any $T \geq r$,*

$$\mathit{Risk}(T, \mathit{PEGE}) \leq a_1 r\sqrt{T}$$

The proof of the theorem is you calculate the bayes risk for the exploitation and exploration periods respectively and add them together. In order to prove the above mentioned theorem, it requires the following two lemmas:

Lemma 3. *Under Assumption 1, there exists a positive constant h_1 that depends only on σ_0, $\bar{u}$ and λ_0 such that for any $\boldsymbol{z} \in \mathbb{R}^r$ and $c \geq 1$,*

$$E\left[\|\hat{\theta}(c) - \theta_0\|^2 | \theta = \theta_0\right] \leq \frac{h_1 r}{c} \tag{4}$$

Lemma 4. *Suppose that Assumption 1 holds and the set $\mathbb{U}_r$ satisfy the SBAR(J) condition. Then, there exists a positive constant h_2 that depends only on σ_0, $\bar{u}$, λ_0 and J, such that for any $\boldsymbol{z} \in \mathbb{R}^r$ and $c \geq 1$,*

$$\begin{aligned}&E\left[\max_{\boldsymbol{u} \in \mathcal{U}_r} \theta_0^{'}(\boldsymbol{u} - \boldsymbol{G}(c)) | \theta = \theta_0\right] \\ \leq &\frac{2}{\|\theta_0\|} E\left[\|\hat{\theta}(c) - \theta_0\|^2 | \theta = \theta_0\right] \leq \frac{2h_1 r}{c\|\theta_0\|} = \frac{h_2 r}{c\|\theta_0\|}\end{aligned} \tag{5}$$

Regret of PEGE We analyse the convergence order of PEGE based on the above assumptions and lemmas:

In the kth step of an r-period exploration, we have the observation:

$$\mathrm{E}\left[\max_{X\in\mathcal{U}_r} X\cdot\theta_0 - \mathbf{b}_k\cdot\theta_0|\theta=\theta_0\right] \le 2\bar{u}\cdot\|\theta_0\|$$

Suppose there are C cycles in the learning (the last cycle may not be completed), then the number of steps is

$$T \ge r(C-1) + C(C-1)/2 \tag{6}$$

Thus, $C \le \sqrt{2T}$.

$$\begin{aligned}\mathrm{Regret}(\theta_0, T, PEGE) &\le \sum_{c=1}^{C}\left[2\bar{u}\|\theta_0\|\cdot r + \frac{rh_2}{c\|\theta_0\|}\cdot c\right]\\ =&(2\bar{u}\|\theta_0\| + \frac{h_2}{\|\theta_0\|})rC < \sqrt{2}(2\bar{u}\|\theta_0\| + \frac{h_2}{\|\theta_0\|})r\sqrt{T}\end{aligned} \tag{7}$$

Although the PEGE algorithm reaches the theoretical bayes risk lower bound, it doesn't perform well in practice. Thus, we proposed the following three modified PEGE algorithms, which still have $O(r\sqrt{T})$ bayes risk, but with a smaller coefficient.

Moditication 1. In the first modified algorithm, instead of doing c periods of exploitation in cycle c, we now do kc periods of exploitation, where k is a constant (Algorithm 1).

Algorithm 2 PEGE Modified 1

Description: For each cycle $c \ge 1$, complete the following two phases:

1. **Exploration (r periods)** For $k = 1, 2, \cdots, r$, play arm $\mathbf{b}_k \in \mathcal{U}_r$ given in Assumption 1(b), and observe the reward $Y^{b_k}(c)$. Compute the OLS estimate $\hat{\theta}(c) \in \mathbb{R}^r$, given by

$$\begin{aligned}\hat{\theta}(c) &= \frac{1}{c}(\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k^{'})^{-1}\sum_{s=1}^{c}\sum_{k=1}^{r}\mathbf{b}_k Y^{b_k}(s)\\ &= \mathbf{Z} + \frac{1}{c}(\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k^{'})^{-1}\sum_{s=1}^{c}\sum_{k=1}^{r}\mathbf{b}_k\epsilon^{b_k}(s)\end{aligned}$$

   where for any k, $Y^{b_k}(s)$, and $\epsilon^{b_k(s)}$ denote the observed reward and the error random variable associated with playing arm $\mathbf{b}_k$ in cycle s.

2. **Exploitation (kc periods)** Play the greedy arm $\mathbf{G}(c) = \arg\max_{v\in\mathcal{U}^r} \mathbf{v}^{'}\hat{\theta}(c)$ for kc periods.

Regret of Modification 1: In this case, we modify the regret bound in equation (6), (7), and get

$$T \ge r(C-1) + kC(C-1)/2 \tag{8}$$

When $k < 2r$, $C \le \sqrt{\frac{2T}{k}}$. Therefore

$$\begin{aligned}&\mathrm{Regret}(\theta_0, T, PEGE1) \le (2\bar{u}\|\theta_0\| + k\frac{h_2}{\|\theta_0\|})rC\\ \le&\sqrt{2}(2\bar{u}\|\theta_0\|/\sqrt{k} + \sqrt{k}\frac{h_2}{\|\theta_0\|})r\sqrt{T}\end{aligned} \tag{9}$$

A suitable k which minimises this upper bound is

$$k^* = \frac{2\bar{u}\|\theta_0\|^2}{h_2}$$

However, since we do not know $\|\theta_0\|$ in advance, we need to estimate it base on the prior distribution of θ and may also adjust our estimation adaptively based on the results we get in each step.

Moditication 2. Are you kidding me?

To be brief, we introduce notations

$$W(c) = (\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k')^{-1}\sum_{k=1}^{r}\mathbf{b}_k\epsilon^{\mathbf{b}_k}(c),$$

$$Z(c) = (\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k')^{-1}\sum_{k=1}^{r}\mathbf{b}_k Y^{\mathbf{b}_k}(c).$$

At cth cycle, conditional on $Z(1), \cdots, Z(c)$ (we will use $Z(1, \cdots, c)$ later), which are observable, we can get a more delicate estimation $E[\|\hat{\theta}(c) - \theta_0\|^2|Z(1, \cdots, c), \theta = \theta_0]$ instead of $E[\|\hat{\theta}(c) - \theta_0\|^2|\theta = \theta_0]$. In addition, we can estimate $\hat{\theta}(c)$ as a weighted summation instead of the average to make the above value even smaller.

It is easy to find that $W(1), \cdots, W(c) \overset{i.i.d.}{\sim} \mathcal{N}(\vec{0}, \Sigma)$. Here

$$\Sigma = \sigma^2(B^TB)^{-1}, \quad \text{for which } B = (\mathbf{b}_1, \cdots, \mathbf{b}_r).$$

We have

$$\begin{array}{ll} W(1) - W(2) & = Z(1) - Z(2)\\ W(1) - W(3) & = Z(1) - Z(3)\\ \cdots & \cdots\\ W(1) - W(c) & = Z(1) - Z(c)\end{array}$$

We can represent the relations by matrices:

$$A\vec{W} = \vec{Z}$$

$$A=\begin{pmatrix} 1 & -1 & 0 & \cdots & 0\\ 1 & 0 & -1 & \cdots & 0\\ \cdots & \cdots & \cdots & \cdots & \cdots\\ 1 & 0 & 0 & \cdots & -1\end{pmatrix}$$

$$\vec{W}=\begin{pmatrix} W'(1)\\ W'(2)\\ \cdots\\ W'(c)\end{pmatrix},\quad \vec{Z}=\begin{pmatrix} Z'(1)-Z'(2)\\ Z'(1)-Z'(3)\\ \cdots\\ Z'(1)-Z'(c)\end{pmatrix}$$

The dimension of A is $(c-1)$ by c. Using QR decomposition, we can get $A=LQ$, where L is $(c-1)$ by c lower triangular matrix, Q is c by c orthogonal matrix. Let $V=Q\vec{W}$. Then

$$LV=\vec{Z}$$

Further, we assume

$$L=(\tilde{L},\vec{0}),\quad V=\begin{pmatrix}\tilde{V}\\ \vec{v}_c\end{pmatrix},\quad Q=\begin{pmatrix}\tilde{Q}\\ \vec{q}_c\end{pmatrix},$$

where $\tilde{L}$ is $(c-1)\times(c-1)$ invertible matrix, $\tilde{V}$ is $(c-1)\times r$ matrix, $\tilde{Q}$ is $(c-1)\times c$ matrix, $\vec{0}$ is $c-1$ dimensional column vector, $\vec{v}_c$ is r dimensional row vector, and $\vec{q}_c$ is c dimensional row vector. Then

$$\vec{Z}=LV=\tilde{L}\tilde{V}\quad\Rightarrow\quad\tilde{V}=\tilde{L}^{-1}\vec{Z}.$$

$$\vec{W}=Q'V=\tilde{Q}'\tilde{V}+\vec{q}_c'\vec{v}_c=\tilde{Q}'\tilde{L}^{-1}\vec{Z}+\vec{q}_c'\vec{v}_c$$

Since $\vec{v}_c$ is independent of $\tilde{V}$, conditional on all the results in the explorations, the distribution of $\vec{v}_c$ is still $\mathcal{N}(\vec{0},\Sigma)$.

Suppose we use

$$\hat{\theta}(c)=(\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k')^{-1}\sum_{s=1}^{c}w_s\sum_{k=1}^{r}\mathbf{b}_kY^{b_k}(s)=\theta_0+\sum_{s=1}^{c}w_sW(s)$$

as the estimator of θ_0 in the cth cycle, where $\sum_{s=1}^{c}w_s=1$. Let $\vec{w}=(w_1,\cdots,w_c)'$. Conditional on $Z(1),\cdots,Z(c)$,

$$\begin{aligned}&E[\|\hat{\theta}(c)-\theta_0\|^2|Z(1,\cdots,c),\theta=\theta_0]\\ =&E[\vec{w}'\vec{W}\vec{W}'\vec{w}|Z(1,\cdots,c),\theta=\theta_0]\\ =&\vec{w}'\left(\tilde{Q}'\tilde{L}^{-1}\vec{Z}\vec{Z}'\tilde{L}'^{-1}\tilde{Q}+tr(\Sigma)\vec{q}_c'\vec{q}_c\right)\vec{w}\\ :=&\vec{w}'A(c)\vec{w}\end{aligned}$$

Since $tr(\Sigma)=\sigma^2(B^TB)^{-1}$, and σ is unknown to us, we use

$$\hat{\sigma}^2(c)=\frac{1}{r(c-1)}\left(\sum_{s=1}^{c}\|Z(s)\|^2-\frac{\|\sum_{s=1}^{c}Z(s)\|^2}{c}\right)$$

as an estimator of σ^2 for $c>1$.

The optimal $\vec{w}$ that minimizes this equation is

$$\vec{w}^*=\frac{A^{-1}(c)\vec{\mathbf{1}}}{\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}}},\quad\text{for which }\vec{\mathbf{1}}=(1,\cdots,1)'.$$

Thus,

$$\begin{aligned}&E[\|\hat{\theta}(c)-\theta_0\|^2|Z(1,\cdots,c),\theta=\theta_0]\\ \geq&(\vec{w}^*)'A(c)\vec{w}^*=\frac{\vec{\mathbf{1}}'A^{-1}(c)A(c)A^{-1}(c)\vec{\mathbf{1}}}{(\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}})^2}\\ =&\frac{\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}}}{(\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}})^2}=\frac{1}{\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}}}\end{aligned}$$

In cycle c, we denote $\frac{1}{\vec{\mathbf{1}}'A^{-1}(c)\vec{\mathbf{1}}}$ as $\delta(c)$. Then

Algorithm 3 PEGE Modified 2

Description: For each cycle $c\geq 1$, complete the following two phases:

1. **Exploration (r periods)** For $k=1,2,\cdots,r$, play arm $\mathbf{b}_k\in\mathcal{U}_r$ given in Assumption 1(b), and observe the reward $Y^{b_k}(c)$. Compute the OLS estimate $\hat{\theta}(c)\in\mathbb{R}^r$, given by

$$\begin{aligned}\hat{\theta}(c)&=\frac{1}{c}(\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k^{'})^{-1}\sum_{s=1}^{c}\sum_{k=1}^{r}\mathbf{b}_kY^{b_k}(s)\\ &=\theta+\frac{1}{c}(\sum_{k=1}^{r}\mathbf{b}_k\mathbf{b}_k^{'})^{-1}\sum_{s=1}^{c}\sum_{k=1}^{r}\mathbf{b}_k\epsilon^{b_k}(s)\end{aligned}$$

   where for any k, $Y^{b_k}(s)$, and $\epsilon^{b_k(s)}$ denote the observed reward and the error random variable associated with playing arm $\mathbf{b}_k$ in cycle s.

2. **Exploitation (c periods)** Play the greedy arm $\mathbf{G}(c)=\arg\max_{v\in\mathcal{U}^r}\mathbf{v}^{'}\hat{\theta}(c)$ for c periods.

Moditication 3. In the third modified algorithm, it further modified the second algorithm: during the exploitation period, it also updates the ordinary least estimate each time (Algorithm 3).

Regret of Modification 3: Since during the exploration periods, it will play the same arms as in algorithm 2, we just need to show the regret during the exploitation periods is smaller than the regret of algorithm 2. If we denote $\hat{\theta}(c,t)$ as the OLS estimator of θ during the t_{th} exploitation period in the c_{th} cycle, then to show algorithm 3 has smaller regret during exploitation is equivalent to show

$$E[\|\hat{\theta}(c,t)-\theta_0\|^2|\theta=\theta_0]<E[\|\hat{\theta}(c)-\theta_0\|^2|\theta=\theta_0].\tag{10}$$

Algorithm 4 PEGE Modified 3

Description: For each cycle $c \geq 1$, complete the following two phases:

1. **Exploration (r periods)** For $k = 1, 2, \cdots, r$, play arm $\mathbf{b}_k \in \mathcal{U}_r$ given in Assumption 1(b), and observe the reward $Y^{b_k}(c)$. Compute the OLS estimate $\hat{\theta}(c) \in \mathbb{R}^r$, given by

$$\begin{aligned}\hat{\theta}(c) &= \frac{1}{c}(\sum_{k=1}^{r} \mathbf{b}_k \mathbf{b}_k^{'})^{-1} \sum_{s=1}^{c}\sum_{k=1}^{r} \mathbf{b}_k Y^{b_k}(s) \\ &= \mathbf{Z} + \frac{1}{c}(\sum_{k=1}^{r} \mathbf{b}_k \mathbf{b}_k^{'})^{-1} \sum_{s=1}^{c}\sum_{k=1}^{r} \mathbf{b}_k \epsilon^{b_k}(s)\end{aligned}$$

   where for any k, $Y^{b_k}(s)$, and $\epsilon^{b_k(s)}$ denote the observed reward and the error random variable associated with playing arm $\mathbf{b}_k$ in cycle s.

2. **Exploitation (c periods)** Play the greedy arm $\mathbf{G}(c) = \arg\max_{v \in \mathcal{U}^r} \mathbf{v}'\hat{\theta}(c)$ for c periods.

Lemma 5. *Suppose X_1 is a $n_1 \times k$ matrix and X_2 is a $n_2 \times k$ matrix, then*

$$Trace(X_1^T X_1 + X_2^T X_2) < Trace(X_1^T X_1).$$

Proof. Denote $A = X_1^T X_1$, then use the Woodbury matrix identity, we get

$$\begin{aligned}&(X_1^T X_1 + X_2^T X_2)^{-1} \\ =&A^{-1} - A^{-1} X_2^T (I + X_2 X_2^T)^{-1} X_2 A^{-1}\end{aligned}$$

Thus, $Trace(X_1^T X_1 + X_2^T X_2) < Trace(X_1^T X_1)$ is equivalent to show $Trace(A^{-1} X_2^T (I + X_2 X_2^T)^{-1} X_2 A^{-1}) > 0$. However, it is easy to see that $A^{-1} X_2^T (I + X_2 X_2^T)^{-1} X_2 A^{-1}$ is a positive semidefinite matrix which means all of the eigenvalue is greater or equal to 0. Thus, our lemma holds. □

Theorem 6. *The algorithm 3 has bayes risk $O(r\sqrt{T})$.*

Proof. As we discussed before, we just need to show (10) holds true. Since we assume $\epsilon \sim N(0, \sigma^2)$, thus the OLS estimator $\hat{\beta}$ of $Y = X\beta + \epsilon$ follows a normal distribution $\hat{\beta} \sim N(\beta, \sigma^2 (X^T X)^{-1})$. Thus $E[\|\hat{\beta} - \beta\|^2] = \sigma^2 Trace(X^T X)^{-1}$. If we denote $X(c,t)$ is the X vector for the $\theta(\widehat{c,t})$ and $X(c)$ is the X vector for the $\theta(\hat{c})$, then $X(c,t)$ contains more rows than $X(c)$. Based on our previous lemma, we know (10) holds true. □

4. Heuristic Algorithms

There are two well known algorithms for this problem, which are Exponential Gradient algorithm and the upper confidence bound algorithm.

Notations:

x_i:	feature of the recommended restaurant at step i, binary vecto
y_i:	rate given by the user at step i, range is $(0, 5)$
θ:	user's preference
μ_0:	prior knowledge of $E[\theta]$
Σ_0:	prior knowledge of $\mathrm{Cov}(\theta)$

At each step, we assume the user's rating $y_i = \theta \cdot x_i + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

4.1. Upper Confidence Bound

At step $t + 1$, we have knowledge $\mu_0,\ \Sigma_0,\ (x_1, y_1), \cdots, (x_t, y_t)$. The distribution

$$\theta | [\mu_0,\ \Sigma_0,\ (x_1, y_1), \cdots, (x_t, y_t)] \sim \mathcal{N}(\mu_t, \Sigma_t)$$

where

$$\Sigma_t^{-1} = \Sigma_0^{-1} + \frac{1}{\sigma^2} X_t^T X_t \qquad = \Sigma_t^{-1} + \frac{1}{\sigma^2} x_t x_t^T \tag{11}$$

$$\mu_t = \Sigma_t \left(\Sigma_0^{-1} \mu_0 + \frac{1}{\sigma^2} X_t^T y \right) \quad = \Sigma_t \left(\Sigma_{t-1} \mu_{t-1} + \frac{1}{\sigma^2} x_t y_t \right) \tag{12}$$

where

$$X_t = \begin{pmatrix} x_1^T \\ \vdots \\ x_t^T \end{pmatrix}$$

Then, for each restaurant r, we suppose its feature vector is $x^{(r)}$. And its expected rating and variance are

$$E[x^{(r)} \cdot \theta] = x^{(r)} \cdot \mu_t, \quad Var(x^{(r)} \cdot \theta) = (x^{(r)})^T \Sigma_t x^{(r)}$$

The restaurant we will recommend at step $t + 1$ is

$$r_t = \arg\max_r E[x^{(r)} \cdot \theta] + 1.96\sqrt{Var(x^{(r)} \cdot \theta)}$$

4.2. Upper Confidence Bound Algorithms

Upper confidence bound algorithms (UCB) are widely used in multiarmed bandit problems because these algorithms usually have good empirical performance. However, they are not as fast as other methods like the EXP3 algorithm, and this may be a problem when the number of stages is very big.

UCB algorithms follow two steps. First, at time t, for each arm u we compute a upper confidence bound $U_t(u)$. Then, at time t we choose the arm u^* such that $u^* \in \arg\max_u U_t(u)$, i.e. u^* has the maximal upper confidence bound.

In our project, we start supposing that $\theta \sim N(\mu_0, \Sigma_0)$ where μ_0 and Σ_0 are computed using the available data. We have that $\theta \sim N(\mu_t, \Sigma_t)$ at step $t+1$, where μ_t, Σ_t are the paramaters of the posterior distribution of θ after we have seen $x_1, y_1, \ldots, x_t, y_t$. Specifically,

$$\begin{aligned}\mu_t &= \Sigma_t\left(\Sigma_0^{-1}\mu_0 + \frac{1}{\sigma^2}X_i^T y\right)\\ &= \Sigma_t\left(\Sigma_{t-1}\mu_{t-1} + \frac{1}{\sigma^2}x_t y_t\right)\\ \Sigma_t^{-1} &= \Sigma_0^{-1} + \frac{1}{\sigma^2}X_t^t X_t = \Sigma_t^{-1} + \frac{1}{\sigma^2}x_t x_t^t\end{aligned}$$

where

$$X_t = \begin{pmatrix} x_1^t \\ \vdots \\ x_t^t \end{pmatrix}.$$

Consequently an upper bound of the .95−confidence interval of the distribution of $y(u)$ is

$$E_{\mu_t,\Sigma_t}\left[x^{(u)}\cdot\theta\right] + 1.96\sqrt{\mathrm{Var}_{\mu_t,\Sigma_t}\left(x^{(u)}\cdot\theta\right)}$$

and then we will choose the arm $u^* = \arg\max_u \left(E_{\mu_t,\Sigma_t}\left[x^{(u)}\cdot\theta\right] + 1.96\sqrt{\mathrm{Var}_{\mu_t,\Sigma_t}\left(x^{(u)}\cdot\theta\right)}\right)$ at step $t+1$. In terms of our example, the arms are restaurants and the vector $x^{(u)}$ is a binary vector that represents the categories of the restaurant u.

4.3. Exponentiated Gradient Algorithm for Bandit Setting (EXP3)

This is a randomized algorithm. It keeps a vector of probabilities for each of the arms, and it chooses an arm according to this vector. The weight of the arm chosen is increased when the loss is small and decreased when the loss is high. Specifically, the algorithm is

1. Given $\gamma \in [0,1]$, initialize the weights $w_u(1) = 1$ for each arm u.
2. At each step t:
   (a) Set $p_u(t) = (1-\gamma)\frac{w_u(t)}{\sum_j w_j(t)} + \frac{\gamma}{K}$ for each arm u.
   (b) Draw the arm u_t choosen according to the distribution $p_{u_t}(t)$.
   (c) Observe the loss $y_{u_t}(t)$.
   (d) Set $w_{u_t}(t+1) = w_{u_t}(t)\exp\left(\gamma\frac{y_{u_t}(t)}{p_{u_t}(t)m}\right)$, and $w_j(t+1) = w_j(t)$ for all other arms.

This algorithm is usually fast, but if there are too many arms, the weights may be zero in most of the cases.

5. Numerical Experiment

In this simulation, we use the yelp academic dataset. The goal of this simulation is to find the favorite restaurant categories for a new user. There are 4596 restaurants in the dataset and each restaurant belongs to one or multiple categories. We first find the top twenty categories that has most restaurants, which are Pizza, Sandwiches, Food etc, and use those 20 categories as our feature. For each restaurant, if it belongs to certain category, then the corresponding element of its feature vector is 1 and 0 otherwise. So the feature vector of each restaurant is a 20 dimensional binary vector.

For each user, we calculated his user preference vector based on his rating and the restaurants' feature vectors that he rated using ridge regression (since there are not too many ratings, ordinary linear regression doesn't work here due to singularity). Then we calculated the sample mean and the sample variance of all users' preference vector and denote them as (μ, Σ). We further assume that for each user's preference vector $\theta \sim N(\mu, \Sigma)$ and generate new user from this distribution.

In our numerial examples, the reward function when the restaurant i is choosen is defined as

$$x_i \cdot \theta + \epsilon$$

where $\epsilon \sim N(0, 0.8)$, θ is the user's preference vector and x_i is the feature vector of the restaurant i.

In our first example, we use the EXP3 algorithm with $\gamma = 0.5$. The time horizon is 1000, and we simulated 100 user's preference vectors. In Figure 1, we can see that the reward is little if we use this algorithm.

In our second example, we use the UCB algorithm described in the previous section. The time horizon is 1000, and we simulated 100 user's preference vectors. In Figure 2, we can see that its performance is much better than the EXP3 algorithm.

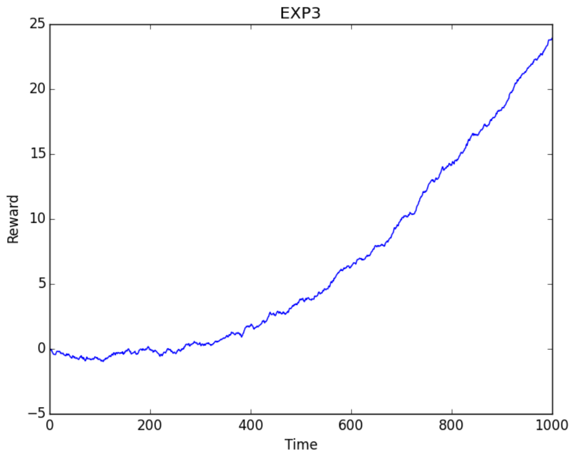

Figure 1. EXP3 algorithm.

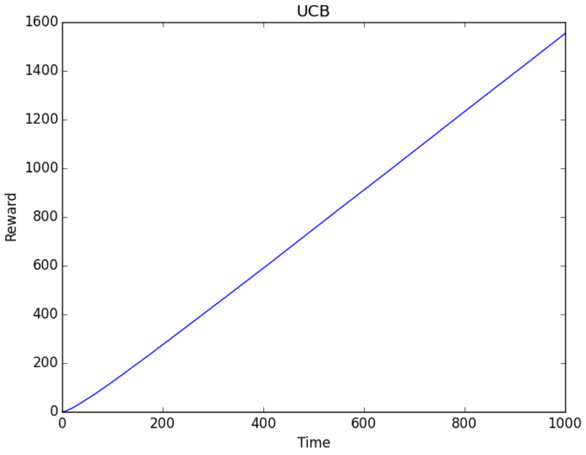

Figure 2. UCB algorithm.

6. Possible Applications

Recommender systems want to find the preference that a user would give to a subset of a finite set of items. They're widely applied to different problems. For example, they're used in Netflix where there are thousands of movies and TV episodes. The biggest challenge of these problems is that there are millions of objets and hundreds of million of users, and so it's necessary to find a model that performs well and be sufficiently fast.

References

Chih-Chun Wang, H. Vincent Poor. Bandit problems with side observations. *Journal of IEEE Transcations on Automatic Control*, 1(11), 11 2004.

Gabor Bartok, Csaba Szepesvari. Partial monitoring with side information. 2012.

Gittins, J.C. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society*, 41 (148177), 1979.

Herbert, Robbins. Some aspects of the sequential desgin of experiments. *Bulletin of American Mathematical Society*, 58(527535), 1952.

Lai, T. L., H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6, 1985.

Paat Rusmevichientong, John N. Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395, 5 2010.

Yasin Abbasi-Yadkori, Andras Antos, Csaba Szepesvari. Forced-exploration based algorithms for playing in stochastic linear bandits. 2009.

Yasin Abbasi-Yadkori, David Pal, Csaba Szepesvari. Improved algorithms for linear stochastic bandits. 2011.