SOMETHING

Michael Byrd and Ross Darrow

May 30, 2020

Abstract

Something for JRPM

1 Introduction

Put why the JRPM audience would care here. I honestly don't know... should be a paragraph or two.

Minor lit review - focus mostly on empirical results, not theory.

1.1 Overview

In Section 2 we present the foundation for bandit algorithms. This includes reviewing Thompson sampling. Section **??** then further inspects a contextual bandit algorithm using Thompson sampling. Here, we review the algorithm and inspect improvements to ease computation **and better support robust exploration**. In Section **??** we implement a simulation study to investigate the potiential performance improvements for the contextual algorithm over bandit algorithms that ignore the context. Finally Section **??** offers final thoughts and future improvements.

2 Background

We consider the bandit problem with binary reward. For time step $t = 1, 2, \ldots$ suppose a recommendation is to be made from one of K possible options. Upon being received, the recommendation is either selected or not selected. The goal is to recommend the option that has the highest chance of success. Suppose, of the K options available to recommend, that each option has p_A features. We refer to each

option as an *arm*, where $\boldsymbol{a}_i \in \mathbb{R}^{p_A}$ denotes the set of features associated with arm i. Further suppose each instance has an additional p_I features, denoted $\boldsymbol{w}_t \in \mathbb{R}^{p_I}$. We use $\boldsymbol{x}_{t,i}$ to refer generally to the i^{th} *context* at time t as the collection of features composed of $\boldsymbol{w}_t$ and $\boldsymbol{a}_i$. The contextual features include the arm features, instance features, and any function of the two, such as interactions, and is assumed to be of dimension p.

Let $y_t \in \{0, 1\}$ be the reward from the recommendation at time t. We assume that

$$\mathbb{E}(y_t|\boldsymbol{x}_{t,i}) = f(\boldsymbol{x}_{t,i}; \boldsymbol{\Theta}), \tag{1}$$

where $\boldsymbol{\Theta}$ are learnable parameters. We wish to recommend the arm

$$\alpha_t^* = \underset{i}{\text{argmax}} \left\{\mathbb{E}(y_t|\boldsymbol{x}_{t,i})\right\}_{i=1}^{K}, \tag{2}$$

which corresponds to the arm with the highest expected reward; in the case of a binary reward, (2) corresponds to the arm with the highest probability of success. Let α_t denote the index corresponding to the arm selected at time t and $\boldsymbol{\Theta}_t$ be the parameters learned at time t. Define the cumulative regret at time t by

$$r_t = \sum_{j=1}^{t} f(x_{j,\alpha_j^*}, \boldsymbol{\Theta}) - f(x_{j,\alpha_j}, \boldsymbol{\Theta}). \tag{3}$$

We assume the parameters $\boldsymbol{\Theta}_t$ are updated as frequently as possible, preferably every iteration, though this may not always be the case in practice ??.

For instance... **Airline example here**

While tempting to always select the item that is estimated to give the highest expected reward, this could often lead to suboptimal results. Typically, bandit algorithms are used when historical data is not available. Hence, it is reasonable that the optimal solution has yet to have been learned due to having no data to incorporate into the estimated $\boldsymbol{\Theta}_t$. This has inspired the exploration-exploitation trade-off ??, which aims to incorporate the uncertainty about undersampled arms when choosing the arm to recommend. While many strategies exists ??, the general notion is that the more an arm is sampled, the more certain we are about the outcome. Hence, arms we are confident to perform worse than other arms will begin to never be picked with time, ultimately converging to the arm that minimizes the regret per each given instance.

2.1 Thompson Sampling for Bandit Problems

Many exploration policies exist in practice, but we focus on Thompson sampling ??. Thompson sampling puts a prior distribution on $\boldsymbol{\Theta}$, and then aims to iteratively

update the posterior distribution with each newly observed example. Access to the posterior distribution quantifies the uncertainity of the estimate of $\boldsymbol{\Theta}$ at time t. When making a recommendation, Thompson sampling algorithms generate $\boldsymbol{\Theta}_t^*$ from the from the current posterior distribution and use it in calculating the probability of success for each arm. This process incorporates the uncertainty of the estimate into the decision making to allow better exploration. As time progresses, the estimate of $\boldsymbol{\Theta}$ will become more certain and the posterior distribution will contract around $\boldsymbol{\Theta}$, which leads to $\boldsymbol{\Theta}_t^*$ often being close to $\boldsymbol{\Theta}$. Thompson sampling has been shown to efficiently trade-off exploring the set of available arms with offering arms that more probable to be successful ??. There has been considerable success achieved by Thompson sampling ??, and it has empircally outperformed other bandit approaches ??.

Formally, we assume that $\boldsymbol{\Theta} \sim \pi(\boldsymbol{\Theta})$ and are interested in iteratively computing

$$\pi(\boldsymbol{\Theta}|\boldsymbol{X}, \boldsymbol{y}) = \frac{\pi(\boldsymbol{\Theta})\pi(\boldsymbol{X}_t, \boldsymbol{y}_t|\boldsymbol{\Theta})}{\pi(\boldsymbol{X}_t, \boldsymbol{y}_t)}, \tag{4}$$

where $\boldsymbol{X}_t = (\boldsymbol{x}_{1,\alpha_1}, \ldots, \boldsymbol{x}_{t,\alpha_t})^T$ and $\boldsymbol{y}_t = (y_1, \ldots, y_t)^T$ denote the historical contexts for the arms recommended and their outcomes, respectively. The denominator is often referred to the normalizing constant, where

$$\pi(\boldsymbol{X}, \boldsymbol{y}) = \int_{\boldsymbol{\Theta}} \pi(\boldsymbol{X}_t, \boldsymbol{y}_t|\boldsymbol{\Theta})\pi(\boldsymbol{\Theta})d\boldsymbol{\Theta}. \tag{5}$$

Often (5) can be intractable, however $\pi(\boldsymbol{\Theta})$ and $\pi(\boldsymbol{X}_t, \boldsymbol{y}_t|\boldsymbol{\Theta})$ can be choosen to be conjugate which gives closed form updates for $\pi(\boldsymbol{\Theta}|\boldsymbol{X}, \boldsymbol{y})$??. Upon observing a new instance at time step $t+1$, $\boldsymbol{\Theta}_{t+1}^*$ is drawn at random from $\pi(\boldsymbol{\Theta}|\boldsymbol{X}_t, \boldsymbol{y}_t)$ and used to select the arm to be recommended as

$$\alpha_t = \underset{i}{\operatorname{argmax}} \left\{ f(\boldsymbol{x}_{t,i}; \boldsymbol{\Theta}_{t+1}^*) \right\}_{i=1}^K.$$

Finally, the posterior is recomputed with the new recomendation and the outcome as in (4).

2.2 Thompson Sampling for the Multi-Arm Bandit

To first illustrate Thompson sampling, we show it in the classic setting of the multi-arm bandit problem. The multi-arm bandit problem ignores the available context when making a recommendation at each instance. In a sense, the multi-arm bandit is exploring the marginal success rate for each available arm. A Thompson sampler

for a multi-arm bandit problem with binary reward puts a prior distribution on the probability of success and incorporates the empirical successes into the posterior update. This can easily be achieved using a beta prior and binomial likelihood, which is conjugate and gives closed form updates for (4). Then, at each iteration, each arm is randomly sampled from its corresponding posterior distribution, and the arm with the highest sample is recommended.

Formally, consider unknown success probabilities for each arm $\boldsymbol{\Theta} = (\theta_1, \ldots, \theta_K)^T$, where $\theta_k \sim Beta(a, b)$ for all $k = 1, \ldots, K$. Letting $\pi(y|\theta_k) \sim Bern(\theta_k)$, then, at time t, the posterior distribution for the success of arm k is

$$\pi(\theta_k|\boldsymbol{y}_t) \sim Beta(a + s_{t,k}, b + n_{t,k} - s_{t,k}), \tag{6}$$

where $n_{t,k} = \sum_{i=1}^{t} \mathbf{1}(\alpha_i = k)$ is the total number of times arm k was recommended at time t and $s_{t,k} = \sum_{i=1}^{t} \mathbf{1}(\alpha_i = k)y_i$ is the total number of successes for arm k at time t. Then, for the instance at time step $t+1$, possible arm success probabilities, $\boldsymbol{\Theta}^*_{t+1} = (\theta^*_1, \ldots, \theta^*_K)$, are generated according to ??. The recommended arm is selected by

$$\alpha^*_t = \underset{i}{\text{argmax}}\{\theta^*_1, \ldots, \theta^*_K\}. \tag{7}$$

As time progresses, the more the posterior distribution will reflect the potiential success rate for each arm. With more observations, the tighter the $Beta$ posterior will contract about the true success probability. Note that each arm's success probability is completely independent from on another. Hence, if contextual information gives insight into multi sets of arms, then the regret could better be minimized by more quickly finding optimal arm traits.

3 Thompson Sampling with Context

To illustrate the benefits of including contextual information, we inspect a contextual bandit algorithm using Thompson sampling. While incorporating additional information can help reduce the overall regret, it comes at the cost of complexity as conjugacy is not easily achieved for (4). For the sake of complexity, we assume a linear relationship between the context and success probability, such that

$$\mathbb{E}(y_t|\boldsymbol{x}_{t,i}) = h(\boldsymbol{x}^T_{t,i}\boldsymbol{\theta}) \tag{8}$$

for regression coefficients $\boldsymbol{\theta} \in \mathbb{R}^p$. The function h is assumed to be the logit function, $h(x) = 1/(1 + \exp(-x))$. Notably, any model that gives a posterior distribution, like

a Bayesian neural network ??, can be used, but this adds additional complexity to the update process.

With the assumed likelihood $y_t \sim Bern(h(\boldsymbol{x}_{t,i}^T\boldsymbol{\theta}))$, the choice for the prior on the regression coefficients is still needed. This, however, causes problems due to the intractability of (5) for most reasonable priors. Such issues are well known in Bayesian statistics, where heavy use of MCMC are common place. A recent advance showed a data augmentation technique can facilitate efficient sampling algorithms for (8) when $\pi(\boldsymbol{\theta}) \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ for mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$??. This technique was adapted by ?? in the context of Thompson sampling, and illustrated great improvements over other contextual bandit algorithms.

3.1 Pólya-Gamma Augmentation for Thompson Sampling

Here we briefly explore the data augmentation technique that facilitates the sampling of the posterior distribution for the linear contextual bandit. A random variable z is distributed according to a Pólya-Gamma distribution with parameters $b \in \mathbb{R}^+$ and $c \in \mathbb{R}$ if

$$z = \frac{1}{2\pi^2} \sum_{j=1}^{\infty} \frac{g_k}{(j - 1/2)^2 + c^2/(4\pi^2)}, \tag{9}$$

where $g_j \sim Gamma(b, 1)$ for $j \in \mathbb{N}$; if z is a Pólya-Gamma random variable with parameters b and c, then $z \sim PG(b, c)$. A useful identity was identified by ??, showing

$$\frac{(e^{\psi})^{\alpha}}{(1 + e^{\psi})^b} = 2^{-b} e^{\kappa\psi} \int_0^{\infty} e^{-z\psi^2/2} p(z) dz \tag{10}$$

for $z \sim PG(b, 0)$. Naturally (10) bares resemblence to the logit function, where we can set $\psi = \boldsymbol{x}_{t,i}^T\boldsymbol{\theta}$.

With some manipulation it can be shown that

$$\pi(\boldsymbol{\theta}|\boldsymbol{X}_t, \boldsymbol{y}_t, \boldsymbol{z}_t) \propto \pi(\boldsymbol{\theta}) \prod_{i=1}^{t} \exp(\frac{z_i}{2}(\boldsymbol{x}_{i,\alpha_i} - \kappa_i/z_i)^2) \tag{11}$$

for $\kappa_i = y_i - 1/2$ and Pólya-Gamma latent variables $\boldsymbol{z}_t = (z_1, \ldots, z_t)^T$. When $\pi(\boldsymbol{\theta}) \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ for mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, then (11) is proportional to a Normal distribution. Further, it can be shown that the full-conditional distribution for the latent variables $\boldsymbol{\theta}$ and $\boldsymbol{z}_t$ can be found in closed form

$$\boldsymbol{\theta} \sim N(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*) \tag{12}$$

$$\boldsymbol{z}_t \sim PG(1, \boldsymbol{X}_t\boldsymbol{\theta}), . \tag{13}$$

where $\boldsymbol{\Sigma}^* = (\boldsymbol{X}_t^T \boldsymbol{Z}_t \boldsymbol{X}_t + \boldsymbol{\Sigma}^{-1})^{-1}$ and $\boldsymbol{\mu}^* = \boldsymbol{\Sigma}^*(\boldsymbol{X}_t \kappa + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu})$

4 Simulation

5 Conclusion