Improving Then Rotting Bandit

谷本　悠斗　　総合研究大学院大学　統計科学専攻　博士課程(5年一貫制)2年

【多腕バンディット問題とは】

バンディット問題は、スロットマシンにおけるアームの選択に由来する。学習者は各ラウンドごとに報酬が未知である多数のアームの中から1つ選択し、そのアームから報酬を得る（図１）。以上の手順を決められた期間繰り返した後の累積報酬の最大化を目的とする。

報酬を最大化する際に、アームに関する情報をどの程度探索するかが問題となる。探索ばかりしているとさまざまなアームに関する情報は集まるが、報酬の低いアームもたくさん引くため、累積報酬を最大化できない。一方、探索をあまりしないと限られた情報のみで最適なアームを選ぼうとするため、最適なアームを選べない可能性が高くなる。以上のような「探索と活用のトレードオフ」を考慮することが累積報酬最大化の鍵となる。以上のような設定は広告配信やインターネット広告などさまざまな分野で応用されている。

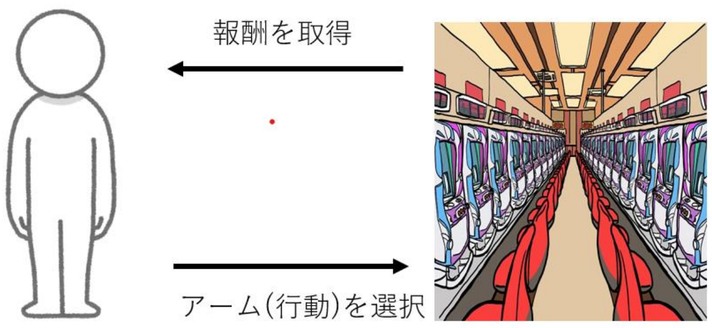

図１　バンディット問題での学習者と環境（スロットマシン）の関係

【報酬の非定常性】

上記の問題に対し、さまざまなアルゴリズム(UCBやトンプソンサンプリングなど)が提案されているが、大半のアルゴリズムはアームを引く回数にかかわらず報酬が一定であるという仮定を置いている。この仮定は応用上必ずしも満たされているとは限らない。

例えば広告配信においてクリック率に対し従来のバンディットアルゴリズムを適用すると、同じ広告（アーム）ばかりが同じユーザーに配信される。これは最初は商品の認知にプラスの影響を与えるが、やがて広告に飽きてクリック率にマイナスの影響を与える可能性がある(Two-factor model)（図２）。この例はニュース記事の推薦でも同様であると考えられる。

以上のような報酬が引く回数に依存するような問題はRested Banditとよばれ、近年研究が進んでいる。

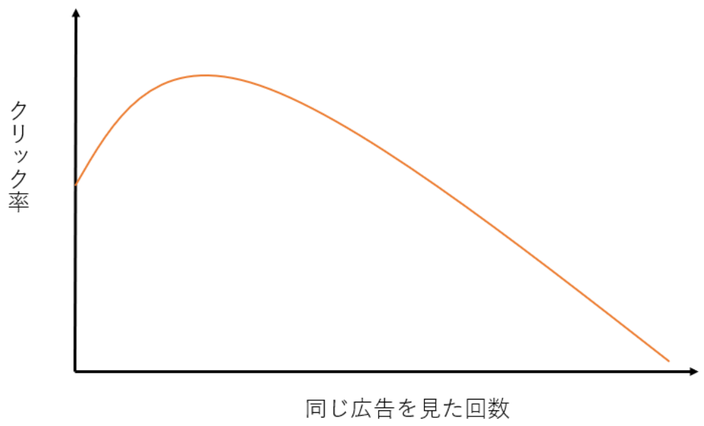

図２　広告を見た回数とクリック率との関係

【問題設定】

総ラウンド数をT、総アーム数をKとする。学習者がラウンドtでアームi(t)を引くとき、学習者が得られる利得$r_{i(t),t}$は、引く回数に依存する平均($\mu_{i(t)}(N_{i(t),t}) \in [0,1]$（$N_{i(t),t}$:ラウンドtまでにアームiを引いた回数）とσ^2-subGaussian であるノイズϵ_t からなる。ここで、ラウンドt-1までのアームの選択と報酬の歴史H_{t-1}で条件付けた時のノイズϵ_tの期待値は0とする。さらに図２のような報酬の推移を考えるために、$(\mu_{i(t)}(N_{i(t),t})$は$N_{i(t),t} < N_i^*$の時、単調増加であるが、$N_{i(t),t} > N_i^*$で単調減少であると仮定する。

$$r_{i(t),t} = \underbrace{\mu_{i(t)}(N_{i(t),t})}_{\text{mean}} + \underbrace{\epsilon_t}_{\sigma^2-\text{subgaussian noise}}$$

$\mu_{i(t)}(N_{i(t),t})$は$N_{i(t),t} < N_i^*$で単調増加、 $N_{i(t),t} > N_i^*$で単調減少

図3　報酬の構造

【オラクル政策】

各アームの報酬について全て事前に知っている場合、資源配分問題と同様にして動的計画法を用いて最適な累積報酬の期待値を計算することができる。$f_i(n)$をアームiをn回引いた時の累積報酬(即ち、$f_i(n) = \sum_{k=0}^{n} \mu_{i(t)}(k)$ ただし、$f_i(0) = 0$)、$F_i(t)$をアーム１からアームkのみを用いた時のラウンドtにおける累積報酬の最大値(即ち、$F_k(t) = \max\{\sum_{i=1}^{k} f_i(N_{i,t}) : \sum_{i=1}^{k} N_{i,t} = t\}$)とする。この時$F_1(t), F_2(t), \cdots, F_K(t)$を順番に計算することで最適解を導出できる（図4）。まず、各tに対して $F_1(t)$を計算する。これはアーム１のみをt回引いた時の累積報酬に等しい。次に$F_2(t)$を計算するためにt回のうち、何回アーム2に割当てを行えば累積報酬を最大化すればよいかを考える。$F_3(t)$ 以降に関しても$F_3(t)$ 以降に関しても同様に解くことで逐次的に最適解を導くことができる。

$$\begin{aligned} &For\ 1 \le t \le T, \\ F_1(t) &= f_1(t) \\ F_2(t) &= \max_{0 \le N_{2,t} \le t} (f_2(N_{2,t}) + F_1(t - N_{2,t})) \\ &\vdots \\ F_K(t) &= \max_{0 \le N_{K,t} \le t} (f_K(N_{K,t}) + F_{K-1}(t - N_{K,t})) \end{aligned}$$

図4　動的計画法によるオラクル政策

【今後の課題】

報酬が未知の時にオラクル政策との誤差が理論的に小さくなるようなアルゴリズムを作ることが今後の目標である。また、実際に数値実験を行い、既存のアルゴリズムより優れていることも示したい。