7R-03

報酬獲得を汎化可能とした確率的認知的満足化

吉井 佑輝 *1　南 朱音 *1　甲野 佑 *1　高橋 達二 *1

*1東京電機大学大学院

1. はじめに

深層強化学習は, 状態行動空間が広大である場合, 最適な方策を学習するために必要な探索量が膨大となる課題を抱えている. そこで我々は, 効率的な探索を実現することを目的とし, 現実の環境下における人間の意思決定傾向をモデル化した認知的満足化価値関数 (Risk-Sensitive Satisficing: RS)[高橋 16] に着目した. 現在, 決定論的な方策としての RS に対し, 確率的方策として拡張した Stochastic RS (SRS) [加藤 21] が提案されている. SRS は RS と比較し, 非定常環境といったより複雑な環境に対して特に有用であることが示されている. しかし, SRS は離散空間しか扱えず, 類似状態に対する汎化が行うことができないことから, 適用できない状況が存在する. そこで, 本論文では SRS を線形近似可能とした Linear SRS(LinSRS) を提案する. その後, 環境に線形空間を持つ文脈付きバンディットシミュレーションでその性能を検証し, さらに既存手法と比較する.

2. 文脈付きバンディット問題

本研究では, 現在の状態に関する特徴ベクトルが明確に与えられることを前提とした, ワンショットの意思決定課題である文脈付きバンディット問題を用いて実験を行う. 各選択肢が期待値を保持しているのに加え, 特徴ベクトル $x_t \in \mathbb{R}^d$ を保持している. ここで, t は時刻, d は各選択肢が持つ特徴ベクトルの次元数である. さらに, 各行動の特徴量が時刻 t により異なる値を取るという設定がされている. このような設定の元, 時刻 t 毎に異なる最適な選択肢を選び出し, 得られる報酬の最大化を目的としたシミュレーションタスクが文脈付きバンディットシミュレーションである.

またバンディット問題では, 性能指標として後悔の度合いを表す regret が多く使用され, 以下のように表される.

$$\text{regret} = \sum_{t=1}^{T} (\max p_{t,a} - p_{t,a^{\text{select}}}) \tag{1}$$

ここで変数 T は現在の試行回数, $p_{t,a}$ は時刻 t に行動 a を選択した時の選択肢の報酬確率, $p_{t,a^{\text{select}}}$ は時刻 t に実際に選択した選択肢の報酬確率である.

3. 認知的満足化価値関数 RS

人間は意思決定において, ある基準値を定め, 基準値を超える価値を持つ行動が見つかるまで探索を続け, 発見したら探索を止めてその行動に満足するという傾向を持つ. この意思決定傾向を満足化と呼ぶ. この満足化原理を意思決定手法として反映した満足化価値関数が考案されている.

$$RS_a = \frac{n_a}{N}\delta_a = \frac{n_a}{N}(E_a - \aleph) \tag{2}$$

ここで n_a は行動 a を試行した回数, N は総試行回数, E_a は行動 a の経験期待値, $\aleph$ は満足化基準値である. また, 最も大きい報酬分布と 2 番目に大きい報酬分布の間に満足化基準値を設定することで, 最適な満足化基準値 $\aleph_{\text{opt}}$ となり, 満足化による最適化の表現が可能となる.

∗ Stochastic Risk-sensitive Satisficing with generalizable rewards. Yuki Yoshii, Akane Minami, Yu Kono, Tatsuji Takahashi : Graduate School of Tokyo Denki University

4. 確率的満足化方策 SRS

バンディット問題において, 選択肢の経験期待値 E_a がいずれも $E_a < \aleph$ である場合, これを非満足状況であると定義する. このとき, 特定の選択肢の試行量割合 $\rho_a = \frac{n_a}{N}$ が高ければ高いほど, その選択肢から算出される RS 値は低下していく. 対してそれ以外の選択肢から算出される RS 値は上昇する. そのため総試行回数 N が充分大きくなると全ての RS 値は一定の値 $-Z$ となり, これを RS 均衡値と呼ぶ. それに伴い N が充分大きい際の試行量割合 ρ_a を以下のように逆算することができる.

$$\text{RS}_a = -Z \tag{3}$$

$$\rho_a = \frac{n_a}{N} = \frac{Z}{\aleph - E_a} \tag{4}$$

$$Z = \frac{1}{\sum_{i=1}^{K} \frac{1}{\aleph - E_a}} \tag{5}$$

K は選択肢の数である. このように RS 均衡値 $-Z$ を求めることができるので, 試行量割合 ρ_a は基準値 $\aleph$ と経験期待値 E_a から定義できる. この RS 均衡値 $-Z$ を用いて求められる試行量割合を理想試行量割合 ${\rho_a}^Z$ とし, SRS 値の算出にはこれを用いる.

SRS では現在の試行量割合 ρ_a と理想試行量割合 ${\rho_a}^Z$ との差分から確率分布を生成し, それに従って行動を選択していく. また負の割合の発生を防ぐために調整パラメータ b を用意する. パラメータ b の更新方法, 確率分布 SRS の定式, 選択確率 π_i を微少の定数 ε を用いて以下に示す.

$$b_a = \frac{n_a}{{\rho_a}^Z} - N + \varepsilon \tag{6}$$

$$\text{SRS}_a = (N + b_{\max}){\rho_a}^Z - n_a > 0 \tag{7}$$

$$\pi_a = \frac{\text{SRS}_a}{\text{SRS}_1 + \text{SRS}_2 + \cdots + \text{SRS}_K} \tag{8}$$

5. 提案手法 LinSRS

SRS を線形近似関数へと拡張した Linear SRS(LinSRS) を提案する. 報酬期待値の不偏推定量 $\hat{\theta}_a$ は特徴入力数 $\mathbf{A}_a$, 累積報酬 $\mathbf{b}_a$, 特徴ベクトル $\mathbf{x}_{t,a}$ を用いて定義される.

$$\hat{\theta}_a = (\mathbf{A}_a^{-1}\mathbf{b}_a)^{\mathrm{T}}\mathbf{x}_{t,a} \tag{9}$$

また, 新たに類似度に基づいた試行回数の擬似的な量, 擬似試行回数 $\hat{\phi}_a$ の更新方法を, 信頼度の推定値 n_a と, ベースラインを含む教師信号 y_a を用いて定義した.

$$\hat{\phi}_a = \hat{\phi}_a + \eta(y_a - n_a)\mathbf{x}_{t,a} \tag{10}$$

$$n_a = \text{softmax}(\hat{\phi}_a^T \mathbf{x}_{t,a}) \tag{11}$$

$$y_a = \frac{w u_{t,a} + \rho_{t,a}}{w + 1} \tag{12}$$

定数 η は学習率を表す. $u_{t,a}$ は各行動の選択の有無を表す one-hot ベクトル, $\rho_{t,a}$ は時刻 t までに行動 a を選択した割合を用いる. w は現時刻 t の試行に対する重みを表す. パラメータの更新はミニバッチ学習で行う.

また, 式 (5) と同様に試行量割合 $\hat{\rho}_a$ は以下のように逆算することができる.

$$\hat{\rho}_a = \frac{n_a}{N} = \frac{Z}{\aleph - \hat{\theta}_a} \quad (13)$$

$$Z = \frac{1}{\sum_{i=1}^{K} \frac{1}{\aleph - \hat{\theta}_a}} \quad (14)$$

これらの推定量と基準値 $\aleph$ を用いて, 式 (2) と同じ構成で, 線形関数に拡張された SRS の確率分布 SRS や選択確率 π_a を定義する.

$$\mathrm{SRS}_a = (N + b_{\max})\hat{\rho}_a{}^{Z} - \hat{\phi}_a \quad (15)$$

$$\pi_a = \frac{\mathrm{SRS}_a}{\mathrm{SRS}_1 + \mathrm{SRS}_2 + \cdots + \mathrm{SRS}_K} \quad (16)$$

LinSRS では各 step において選択確率 π_a を基に行動を選択する. 各変数の更新方法を以下に示す. 変数 $r_{t,a}$ は, 各 step で実際に行動 a を選択して与えられた 0 か 1 の報酬値を表す. 特徴入力数 $\mathbf{A}_a$ の初期値は単位行列 $\mathbf{I}$ とした. 累積報酬 $\mathbf{b}_a$ の各次元の初期値は全て 0 とした.

$$\mathbf{A}_a = \mathbf{I} + \sum_{t=1}^{T} \mathbf{x}_{t,a}\mathbf{x}_{t,a}^{\mathrm{T}} \quad (17)$$

$$\mathbf{b}_a = \sum_{t=1}^{T} \mathbf{x}_{t,a} r_{t,a} \quad (18)$$

$$b > \frac{\hat{\phi}_a}{\hat{\rho}_a^{Z}} - N \quad (19)$$

6. 実験

本研究では, 人工的な生成分布からサンプリングしたデータセット (以後, 人工データセットとする) を用いた文脈付きバンディット問題で , 提案手法 LinSRS の性能を評価する. LinSRS と既存の最適化アルゴリズムを同一の評価指標で比較するために, LinSRS の最適な満足化基準値 $\aleph_{\mathrm{opt}}$ が常に一定となる人工データセットを用いた. 最適な満足化基準値を与えた LinSRS と既存の最適化アルゴリズムは, どちらも最適行動の発見を目指すアルゴリズムとして定義できる . 目標を同じにすることで, LinSRS と既存の最適化アルゴリズムを同一の評価指標で比較することが可能となる.

6.1 特徴ベクトルの分布と報酬関数

報酬分布のパラメータ $\boldsymbol{\theta}$ は, 平均 $\boldsymbol{\mu}$ がゼロベクトルであり, 分散共分散行列が対角線のみ $\sigma = 0.01$ の対角行列を要素に持つ, 多変量正規分布から独立にサンプリングされると定義した.

$$\boldsymbol{\theta}_a \sim \mathcal{N}(\boldsymbol{\mu},\ \sigma \times \mathbf{I}) \quad (20)$$

6.2 常に最適な基準値が一定であるデータセット

本実験では, 最適な満足化基準値 $\aleph_{\mathrm{opt}} = 0.7$ とし, 人工データセットを以下の手順で生成した.

1. 式 (20) で任意のパラメータ $\boldsymbol{\theta}_{\boldsymbol{a}}^{*}$ を生成.
2. 任意のパラメータ $\boldsymbol{\theta}_{\boldsymbol{a}}^{*}$ に対して, 特徴 $\mathbf{x}_{\mathbf{t},\mathbf{a}}$ を大量に生成.
3. その中から $p_{\mathrm{first}} > \aleph_{\mathrm{opt}} > p_{\mathrm{second}}$ となる, 特徴ベクトル $\mathbf{x}_{\mathbf{t},\mathbf{a}}$ と報酬分布 $p_{t,a}$ のみを抽出.

6.3 実験設定

本実験において選択肢 a の数を 8, 次元数 d を 128 と設定した. また, 各シミュレーションの最初に全ての選択肢をそれぞれ 10 回ずつ選択することでパラメータの初期更新を行うとともに, 各種アルゴリズムでバッチサイズを 20 に統一した. アルゴリズムの方策に従って行動選択を行う総回数は 100,000 step, これを 100 シミュレーション繰り返した際の regret の平均を算出し, 結果として示す.

性能比較検証のために用いた 4 つのアルゴリズムは LinUCB[Li 10], LinTS[Riquelme 18], LinRS, LinSRS である. それぞれのパラメータは最適なものを設定しており, LinUCB がスケーリングパラメータ $\alpha = 0.1$, LinTS がスケーリングパラメータ $\lambda = 0.25$, 逆ガンマ分布の引数の初期値 $\alpha = \beta = 6$. 各人工データセットにおける最適な満足化基準値 $\aleph_{\mathrm{opt}} = 0.7$ に対して, LinRS は基準値 $\aleph = 0.65$, LinSRS は基準値 $\aleph = 0.6$ とした. 加えて, 基準値 $\aleph$ による挙動の差異を調べるために基準値 $\aleph = [0.55,0.65]$ を追加して実験を行った.

7. 結果・考察

実験結果として各アルゴリズムにおける regret の推移を表したものを図 1 に示す. 図に関して, 線と同じ色で薄く塗られた範囲はそのアルゴリズムの regret の振れ幅を表しており, 上端・下端はそれぞれ全シミュレーション中の最大値・最小値と一致する.

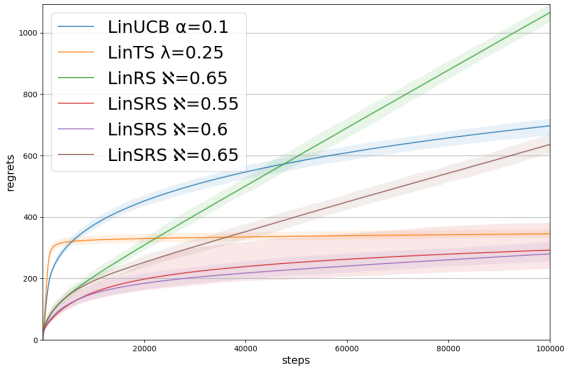

図 1: 最適な満足化基準値 $\aleph_{\mathrm{opt}}$=0.7 の人工データセットを用いたシミュレーションにおける regret の推移

LinSRS は基準値 $\aleph$ が最適である時, 全体を通して regret を最も抑えることができている. また, 分散に関しても基準値を最も低く設定したもの以外は分散を抑えられており, 性能の良さが見て取れる. さらに, LinSRS はシミュレーション中の初期状態において早い段階から学習傾向が見られ, regret の伸びが抑えられている. 満足化基準値と確率的な方策により, パラメータの正確な推定や最適な行動を選択するために必要となる探索が効率良く行われることが, 成績の良さに表れていると考えられる.

一方で, 各シミュレーションの最適な満足化基準値が $\aleph_{\mathrm{opt}} = 0.7$ であるデータセットに対して, 実際は $\aleph_{\mathrm{opt}} = 0.6$ であった. この結果から RS における最適基準値の設け方と異なっていることが分かる. これは, 線形近似によるパラメータの近似誤差や試行量割合の更新方法が正確な推定に影響を及ぼしていると推測される.

8. おわりに

本研究では, 確率的満足化方策を線形近似可能とすることで, 文脈付きバンディットシミュレーションにおいて, 既存手法と比較し結果が同等以上となることを示した. 今後の課題としては, 試行量割合や満足状況下におけるパラメータの更新方法を改善すること, また, 実世界におけるデータセットを用いてより複雑な環境下における効果を検証することが挙げられる.

参考文献

[高橋 16] 高橋 達二, 甲野 佑, 浦上 大輔. 認知的満足化 - 限定合理性の強化学習における効用, 人工知能学会論文誌, Vol. 31, No. 6, pp. 1-11, (2016).

[Li 10] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. WWW, pp. 661–670, (2010).

[Riquelme 18] Carlos Riquelme, George Tucker, Jasper Snoek. Deep Bayesian Bandits Showdown An Empirical Comparison of Bayesian Deep Networks for Thompson Sampling, ICLR 2018, (2018).

[加藤 21] 加藤 暦雄; 甲野 佑; 高橋 達二. 満足化方策における非満足均衡を用いた確率的方策の検証. 人工知能学会全国大会論文集 2021, JSAI 2021, (2021).