F-052

複利型深層強化学習における投資比率の最適化のための損失関数の提案

Proposed Loss Function for Optimizing Investment Ratio in Compound Deep Reinforcement Learning

位田 奨 *1　松井藤五郎 *2*3
Sho Inden　Tohgoroh Matsui

*1株式会社テクノプロ　テクノプロ・デザイン社
TechnoPro, Inc. TechnoPro Design Company

*2中部大学 生命健康科学部 臨床工学科
Department of Clinical Engineering, Colledge of Life and Health Sciences, Chubu University

*3中部大学 工学部 情報工学科
Department of Computer Science, College of Engineering, Chubu University

本論文では、複利型深層強化学習における投資比率の最適化のための損失関数を提案する。現在、強化学習を金融取引に応用する研究が盛んに行われている。複利型強化学習は、利益率の複利効果を最大化する行動を習得することを目的とした強化学習の枠組みであり、パラメータとして投資比率が存在する。この投資比率を最適化することで利益率の複利効果を最大化できる。また、深層強化学習を複利型に拡張した複利型深層強化学習が提案されている。先行研究において、複利型強化学習、複利型深層強化学習において投資比率を最適化する手法が提案されている。しかし、後者の手法は前者の手法をそのまま応用したものでありパラメータが増える、アルゴリズムが冗長であるといった問題が存在する。そこで本論文では、新たに損失関数を定義することによりそれらの問題点を解決する手法を提案する。

1. はじめに

現在、機械学習の発展に伴い、機械学習を様々な分野に応用する研究が盛んに行われている。その中で、強化学習を用いて金融取引戦略を習得する研究が行われている。複利型強化学習 [1] は、利益率の複利効果を最大化する行動を習得することを目的とした強化学習の枠組みである。また、深層教科学習を複利型に拡張した、複利型深層強化学習 [2] も提案されている。

複利型強化学習及び複利型深層強化学習には、新たなパラメータとして投資比率が追加されている。投資比率とは、自己資本の何 % を投資するかを設定するパラメータであり、この値を最適な値に設定することで、利益率の複利効果を最大化することができる。

先行研究にて、複利型強化学習においてオンライン勾配法によって投資比率を最適化する手法 [3]、複利型強化学習の手法を応用した複利型深層強化学習において投資比率を最適化する手法 [4] が提案されている。

しかし、複利型深層強化学習における投資比率を最適化する先行研究 [4] は、複利型強化学習における投資比率を最適化する手法 [3] をそのまま応用した手法であり、パラメータが増える、アルゴリズムが冗長であるといった問題が存在する。

そこで本研究では、それらの問題を解決する手法を提案する。

2. 先行研究

2.1 複利型強化学習

2.1.1 複利型強化学習の枠組み

従来の強化学習は、割引収益

$$r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$$

を最大化する行動を学習する。ここで r_t は時刻 t に観測された報酬、γ は割引率である。

それに対し複利型強化学習は二重指数関数による割引複利利益率

$$(1+R_{t+1}f)(1+R_{t+2}f)^{\gamma}(1+R_{t+3}f)^{\gamma^2}\cdots = \prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k}$$

の対数期待値を最大化する行動規則を学習する。ここで R_t は時刻 t に観測された利益率、f は投資比率である。

複利型強化学習における状態価値 $V^{\pi}(s)$ 及び行動価値 $Q^{\pi}(s,a)$ はそれぞれ式 (1)、式 (2) で表される。

$$\begin{aligned} V^{\pi}(s) &= E_{\pi}\left[\log\prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k}\,\middle|\, s_t = s\right] \\ &= \sum_{\alpha\in\mathcal{A}}\pi(s,a)\sum_{s'\in\mathcal{S}}\mathcal{P}^{\alpha}_{ss'}\left(\mathsf{R}^{a}_{ss'} + \gamma V^{\pi}(s')\right) \end{aligned} \tag{1}$$

$$\begin{aligned} Q^{\pi}(s,a) &= E_{\pi}\left[\log\prod_{k=0}^{\infty}(1+R_{t+k+1}f)^{\gamma^k}\,\middle|\, s_t = s,\, a_t = a\right] \\ &= \sum_{s'\in\mathcal{S}}\mathcal{P}^{\alpha}_{ss'}\left(\mathsf{R}^{a}_{ss'} + \gamma V^{\pi}(s')\right) \end{aligned} \tag{2}$$

ここで、$\pi(s,a)$ は行動規則 π の下で状態 s において行動 a が選択される確率、$\mathcal{P}^{\alpha}_{ss'}$ は状態 s において行動 a を行なった時に次の状態が s' になる確率、$\mathsf{R}^{a}_{ss'}$ は状態 s において行動 a を行なって次の状態が s' になったときに得られる利益率 Rf に 1 を加えたものの対数期待値

$$\mathsf{R}^{a}_{ss'} = E\left[\log(1+R_{t+1}f)\,|\, s_t = s,\, a_t = a,\, s_{t+1} = s'\right]$$

を表す。式 (1) 及び式 (2) は、従来の強化学習における V^{π} および Q^{π} の式の報酬期待値 $\mathcal{R}^{a}_{ss'}$ を $\mathsf{R}^{a}_{ss'}$ に置き換えたものに等しい。複利型強化学習では、この $Q^{\pi}(s,a)$ を最大化するような行動規則 π を学習する。

連絡先: 位田奨、株式会社テクノプロ　テクノプロ・デザイン社、inden.sho@technopro.com

アルゴリズム 1 投資比率を最適化する複利型 Q 学習

Input: 割引率 γ、学習率 α、学習率 η

Q 値を初期化
f を初期化
while 各エピソードに対して繰り返し **do**
  s を初期化
  while s が終状態になるまで **do**
    Q 値に従い状態 s における行動 a を選択
    行動 a を実行して利益率 R と次の状態 s' を観測
    $\delta \leftarrow \log(1+Rf) + \gamma \max_{a' \in \mathcal{A}} Q(s', a') - Q(s, a)$
    $Q(s, a) \leftarrow Q(s, a) + \alpha\delta$
    $f(s, a) \leftarrow f(s, a) + \eta \frac{R}{1+Rf(s,a)}$
    $s \leftarrow s'$
  end while
end while

2.1.2 複利型 Q 学習

複利型 Q 学習は、従来の Q 学習における報酬 r_{t+1} を $\log(1+R_{t+1}f)$ に置き換えたものに等しい。時刻 t の状態 s_t において行動 a_t を実行し、次の時刻 $t+1$ に利益率 R_{t+1} を受け取り、Q 値を以下の式で更新する。

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha\delta_t$$
$$\delta_t = \log(1 + R_{t+1}f) + \gamma \max_{a \in \mathcal{A}} Q_t(s_{t+1}, a) - Q_t(s_t, a_t)$$

ここで、α は学習率である。

また、複利型 Q 学習のアルゴリズムの中で、各状態・行動ごとの投資比率 f を式 (3) によって更新する手法 [3] が提案されている。

$$f_{t+1}(s_t, a_t) = f_t(s_t, a_t) + \eta \frac{R_t}{1 + R_t f_t(s_t, a_t)} \tag{3}$$

ここで η は学習率である。

投資比率を最適化する手法を取り入れた複利型 Q 学習のアルゴリズムをアルゴリズム 1 に示す。

2.1.3 複利型深層強化学習（複利型 DQN）

複利型 DQN [2] は、複利型 Q 学習と同様に DQN における報酬 r を $\log(1+Rf)$ に置き換えたものに等しい。

また、Q ネットワークとは別に投資比率 f を出力するネットワーク (以下、f ネットワークと表記) を追加し、複利型 DQN のアルゴリズムの中で式 (3) を f ネットワークの教師信号として学習し、投資比率を最適化する手法が提案されている [4]。

投資比率を最適化する従来手法 [4] を用いた複利型 DQN のアルゴリズムをアルゴリズム 2 に示す。

2.2 先行研究の問題点

複利型 DQN における投資比率を最適化する先行研究では、f ネットワークの教師信号として式 (3) を用いている。そのため、f ネットワークの学習率とは別に式 (3) における学習率 η も設定する必要がある。

また、ニューラルネットワークの更新は勾配法であり、式 (3) もまた勾配法である。式 (3) を教師信号として用いることは、言わば勾配法により勾配法の教師信号を求めている状態であり、式 (3) の勾配部分 $\frac{R_t}{1+R_t f_t}$ を直接 f ネットワークの更新に用いるべきと考えられる。

アルゴリズム 2 投資比率を最適化する複利型 DQN(従来手法)

Input: 割引率 γ、学習率 η、Q ネットワークの学習率、f ネットワークの学習率

Q ネットワークを初期化
f ネットワークを初期化
while 各エピソードに対して繰り返し **do**
  s を初期化
  while s が終状態になるまで **do**
    Q 値に従い状態 s における行動 a を選択
    行動 a を実行して利益率 R と次の状態 s' を観測
    $y \leftarrow \log(1+Rf) + \gamma \max_{a' \in \mathcal{A}} Q(s', a')$ を教師信号として Q ネットワークを更新
    $f(s, a) + \eta \frac{R}{1+Rf(s,a)}$ を教師信号として f ネットワークを更新
    $s \leftarrow s'$
  end while
end while

3. 提案手法

本論文では式 (4) を f ネットワークの損失関数として使用する手法を提案する。ここで、N は学習するデータ数である。

$$L = -\sum_{i=1}^{N} \log(1 + R_i f(s_i, a_i)) \tag{4}$$

式 (4) における $\log(1 + R_i f(s_i, a_i))$ は、式 (3) の勾配部分 $\frac{R_{t+1}}{1+R_{t+1}f_t(s_t,a_t)}$ を f で積分したものである。投資比率の最適化は、$\log(1 + R_i f(s_i, a_i))$ を総和したものを最大化する問題と同義である。そのため -1 倍することで、最小化問題としている。

式 (4) は誤差を求める式ではないが、最小値を持つ連続的な関数であるため損失関数として用いることができる。この手法により、勾配法によって勾配法の教師信号を求めている状態を解消することができ、また学習率 η を設定する必要がなくなる。

ただし、この手法を用いる場合には f ネットワークを行動ごとに別々の出力を持つ多出力モデルとして定義する。その上で、学習に用いるデータを行動ごとに分け、分けた行動ごとに式 (4) を用いて損失を求め、求めた損失から行動ごとに勾配を求め、各出力層に対応する勾配を適応する。

提案手法を用いた複利型 DQN のアルゴリズムをアルゴリズム 3 に示す。

4. 実験

提案手法が、投資比率の最適化に有効であることを確認する。学習環境として、先行研究 [4] にて使用されたバンディット環境を使用した。

本研究で使用するバンディット問題は、図 1 に示す Wheel から 1 つを選択する環境である。この選択が、行動 a に当たる。各 Wheel は、1 ドルの投資に対するリターンが 6 個書かれており、その中からランダムにリターンが返される。

各 Wheel の投資比率 f の最適値はケリー基準によって求められる。各 Wheel のケリー基準は、Wheel A が 69%、Wheel B が 50%、Wheel C が 22%、Wheel D が 150%である。

実験の結果として、f ネットワークの出力が各行動のケリー基準付近になることが望ましい結果である。

アルゴリズム 3 投資比率を最適化する複利型 DQN(提案手法)

Input: 割引率 γ、Q ネットワークの学習率、f ネットワークの学習率

Q ネットワークを初期化
f ネットワークを初期化
while 各エピソードに対して繰り返し **do**
  s を初期化
  while s が終状態になるまで **do**
    Q 値に従い状態 s における行動 a を選択
    行動 a を実行して利益率 R と次の状態 s' を観測
    $y \leftarrow \log(1+Rf) + \gamma \max_{a' \in \mathcal{A}} Q(s', a')$ を教師信号として Q ネットワークを更新
    $-\sum_{i=1}^{N} \log(1+R_i f(s_i, a_i))$ を使用し各行動の損失を求める
    求めた損失を f ネットワークの各出力に適用し更新
    $s \leftarrow s'$
  end while
end while

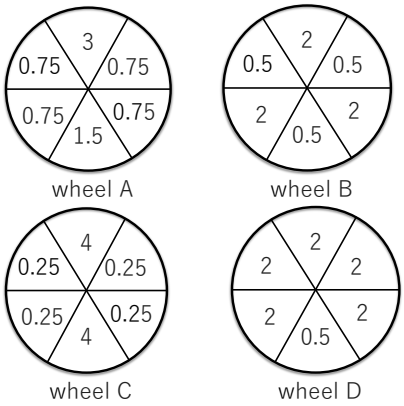

図 1: 投資比率の最適化の実験で使用するバンディット問題

学習にあたり、Q ネットワーク、f ネットワークの構造は共に入力層、中間層、出力層の 3 層とし、中間層のノード数は 16、それぞれのネットワークの学習率は 0.00005、バッチサイズは 16 とした。また、本実験で使用するバンディット環境には状態遷移が存在しないため、状態 s として常に 1 を入力した。

投資比率 f は、0 未満の値をとることは定義上無い。また、投資比率が 0 であることは 0 ドル投資することを示しており、不適切な値と言える。さらに今回の環境では問題ないが、Fx や株取引等の実際の環境においてはレバレッジの上限が法律に定められている。そのため、式 (5) に定義する Limit ReLU 関数を f ネットワークの出力層の活性化関数とする。

$$LimitReLU = \begin{cases} 0.01 & x \leq 0.01 \\ x & 0.01 < x < 2 \\ 2 & x \geq 2 \end{cases} \tag{5}$$

定義した Limit ReLU 関数をプロットしたものを図 2 に示す。Limit ReLU 関数は ReLU 関数を改造し、上限と下限を設けた関数である。本実験では上限を 2、下限を 0.01 としたが、実際はそれらはハイパーパラメータであり実際の法律や学習者の投資に対する考え等により決定される。

学習した結果を図 3 に示す。ここで、縦軸は f の値、横軸はステップ数を表す。

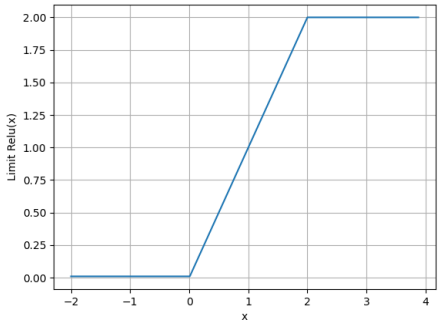

図 2: 定義した Limit ReLU 関数

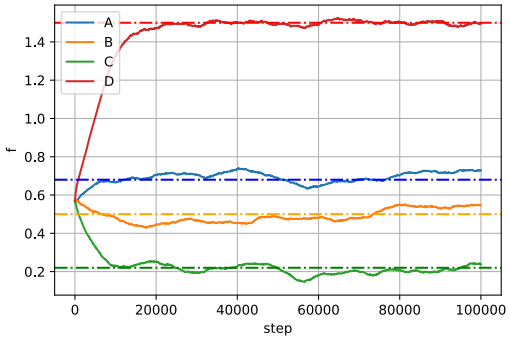

図 3: 複利型 DQN の各行動の投資比率

図 3 を見ると、Wheel A が 69%、Wheel B が 50%、Wheel C が 22%、Wheel D が 150%付近に収束していることが読み取れる。

5. 考察とまとめ

実験の結果から、f ネットワークの出力が最適な投資比率付近に推移しており、最適な投資比率を学習できていることが確認できた。このことから、提案手法は投資比率の最適化において有効であると考えられる。

参考文献

[1] 松井 藤五郎 (2013). 複利型強化学習—強化学習のファイナンスへの応用—. 計測と制御, 53(1):1022–1027.

[2] 松井 藤五郎, 片桐 雅浩 (2016). 金融取引戦略獲得のための複利型深層強化学習. 第 16 回人工知能学会金融情報学研究会 (SIG-FIN), SIG-FIN-016-01.

[3] 松井 藤五郎, 後藤 卓, 和泉 潔, 陳 ユ (2013). 複利型強化学習における投資比率の最適化. 人工知能学会論文誌, 28(3):267–272.

[4] 位田 奨, 松井 藤五郎 (2022). 複利型深層強化学習における投資比率の最適化及びリスクを考慮した行動の習得. 第 36 回人工知能学会 (JSAI), 1A5-GS-2-02.