信頼領域法，TRPO，GAE

菱沼 徹

2020 年 3 月 6 日

- 信頼領域法・・・一般的な連続凸最適化問題における勾配法の一種.
- TRPO・・・信頼領域法のアイデアを，強化学習における方策改善に適用.
- GAE・・・関数近似を改善する方法.

1 Trust Region 法（[2] の 4 節）

非線形最適化問題では，非線形目的関数 $f(x)$ を最小化することを目指す．そのままでは解くことは難しいため，現在の探索点周りで近似最適化問題を設定し，その解を用いて次の探索点を見つける，ということの反復を考える．k 反復目の近似最適化問題は，現在の探索点 x_k の周りで f をある次数まで Taylaor 展開したものを近似問題の目的関数として扱い，これを（n 次）モデル関数と呼び，m_k で表す.

現在の探索点から次の探索点までのステップの決め方には，2 つのアプローチがある（[2] の 65 ページ）.

- line search ... f のモデル関数を用いて勾配を導出し，その方向に沿って適切なステップ幅 α を決める.
- trust region ... まずモデル関数が f を適切に表現していると信頼できる領域を定義し，その後この領域におけるモデルを近似的に最小にするようにステップを選ぶ.

実際のアルゴリズムでは，trust region のサイズを，以前の反復の間のアルゴリズムの性能に従って決める．もしモデルが信頼できて目的関数の挙動を正確に予測するなら，trust region のサイズは大きくなり大きなステップがとられる．逆に，失敗したステップの場合には，現在のモデル関数が現在の trust region 上の目的関数を適切に表現していないことが分かり，trust region のサイズを小さくしてもう一度ステップを計算しなおす.

1.1 概要

次の 2 次モデル関数を考える（[2] の式 (4.1)）.

$$m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T B_k p \tag{1}$$

ここで，$f_k = f(x_k)$，$\nabla f_k = \nabla f(x_k)$，$B_k$ は対称行列とする．Taylor の定理より，$t \in (0,1)$ に対して，

$$f(x_k + p) = f_k + \nabla f_k p + \frac{1}{2} p^T \nabla^2 (x_k + tp) p \tag{2}$$

であるため，$m_k(p)$ と $f(x_k + p)$ の間の差は $\mathcal{O}(||p||^2)$ である（[2] の式 (4.2)）.

B_k が真のヘッセ行列 $\nabla^2 f(x_k)$ に等しい場合，近似誤差は $\mathcal{O}(||p||^3)$ である．$B_k = \nabla^2 f(x_k)$ である場合は trust-region Newton 法と呼ばれるが，ここでは一般に（対称で一様有界な）B_k を用いる場合を述べるとする.

各ステップを得るために，次の部分問題の解を探す（[2] の式 (4.3)）.

$$\min_p m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T B_k p, \quad s.t.\ ||p|| \leq \Delta_k \tag{3}$$

ここで，$\Delta_k > 0$ は，trust-region 半径である．$||\cdot||$ をユークリッドノルムとして，問題 (3) の解 p_k^* は半径 Δ_k のボールの中で m_k を最小化する．したがって，trust-region 法は，目的関数と制約の両方が 2 次である部分問題 (3) の系列を解くことを要求する．もし B_k が正定で $||B_k^{-1}\nabla f_k|| \leq \Delta_k$ なら，問題 (3) の解を見つけることは簡単である．それは，単純な制約されていない最小値 $p_k^B = -B_k^{-1}\nabla f_k$ である．他の場合には問題 (3) の解は明らかではないが，通常は，特別に大きい努力をせずとも見つけることができる．いずれの場合においても，収束と良い実用性を得るために必要なのは，近似解のみである．

trust region 法を定義する最初の課題は，各反復において trust region 半径 Δ_k を選ぶ方法である．我々は，この方法を，各反復におけるモデル関数 m_k と目的関数 f の間の合意（agreement）に基づかせる．与えられたステップ p_k に対して，次の比を定義する（[2] の式 (4.4)）．

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)} \tag{4}$$

分子は actual reduction（実際の減少），分母は predicted reduction（予測される減少）と呼ばれる．$p = 0$ を含む領域にわたるモデル m_k を最小化することにより p_k が得られるため，predicted reduction は常に非負である．そのため，次のことが考えられる．

- もし ρ_k が負なら，新しい目的関数の値 $f(x_k + p_k)$ は現在の目的関数の値 $f(x_k)$ よりも大きくなってしまうため，ステップ p_k は排除されるべきである（Algorthm 1 の 17 行目）．
- 逆に，もし ρ_k が 1 に近いなら，このステップについてモデル m_k は関数 f の良い近似をしているため，次の反復に対する trust region を安全に拡大する（Algorthm 1 の 9 行目）．
- もし ρ_k が正だが 1 に近くないなら，trust region を変化させない（Algorthm 1 の 11 行目）．
- しかし，もし ρ_k がゼロや負に近いなら，trust region を小さくする（Algorthm 1 の 6 行目）．

Algorthm 1 は trust region 法の概要を説明する．

1.2 自分のための補足

■1 変数の Rolle の定理． f が $[a, b]$ で連続，(a, b) で微分可能，$f(b) = f(a)$ のとき，ある $\xi \in (0, 1)$ で $f'(\xi) = 0$ となる．

■1 変数の Taylor の定理.

$$f(x + p) = f(x) + f'(x)p + \frac{1}{2}mp^2$$

とおく．さらに，

$$F(y) = f(x + p) - f(y) + f'(x)(x + p - y) + \frac{1}{2}m(x + p - y)^2$$

とおく．このとき，$F(x) = F(x + p) = 0$ が成り立つ．また，Rolle の定理より，ある $\xi \in (x, x + p)$ に対して，$F'(\xi) = 0$ が成り立つため，次式が得られる．

$$(-f''(\xi) + 2m)(x + p - \xi) = 0$$

したがって，$m = \frac{f''(\xi)}{2}$ が成り立つ．$\xi \in (x, x + p)$ を $\xi = x + tp$ と書き直せば（$t \in (0, 1)$），

$$f(x + p) = f(x) + f'(x)p + \frac{1}{2}f''(x + tp)p^2$$

■多変数の Taylor の定理. 導出は省略するが，上式を拡張して次のように書ける．

$$f(x + p) = f(x) + \nabla f(x)p + \frac{1}{2}p^T\nabla^2(x + tp)p$$

Algorithm 1 Trust Region 法

$\bar{\Delta} > 0$，$\Delta_0 \in (0, \bar{\Delta})$，$\eta \in [0, \frac{1}{4})$ を与える．
for $k = 0, 1, \cdots$ **do**
  問題 (3) を解いて，p_k を得る．
  式 (4) より ρ_k を計算する．
  if $\rho_k < \frac{1}{4}$ **then**
    $\Delta_{k+1} = \frac{1}{4}||p_k||$（小さくする）
  else
    if $\rho_k > \frac{3}{4}$ and $||p_k|| = \Delta_k$ **then**
      $\Delta_{k+1} = \min(2\Delta_k, \bar{\Delta})$（大きくする）
    else
      $\Delta_{k+1} = \Delta_k$（変化させない）
    end if
  end if
  if $\rho_k > \eta$ **then**
    $x_{k+1} = x_k + p_k$（更新する）
  else
    $x_{k+1} = x_k$（更新せずやり直す）
  end if
end for

2 TRPO: Trust Region Policy Optimization [3]

2.1 準備

初期状態 s_0 は，分布 ρ_0 からサンプルされる．軌道 $(s_0, a_0, s_1, a_1, \cdots)$ は，終端状態にたどり着くまで，方策 $a_t \sim \pi(\cdot|s_t)$ に従って行動をサンプリングし，またダイナミクス $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$ に従って状態をサンプリングすることにより生成される．報酬 $r(s_t)$ を，各時間ステップにおいて受け取る．割引率を $\gamma \in (0, 1)$ とする．確率的定常方策 π で表し，$a_t \sim \pi(a_t|s_t)$ とする．期待割引報酬 $\eta(\pi)$ を次のように定義する．

$$\eta(\pi) = \mathbb{E}_{s_0, a_0, \cdots \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t)\right]$$

状態行動価値関数 Q_π，価値関数 V_π，アドバンテージ関数 A_π を次のように定義する．

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}, a_{t+1}, \cdots \sim \pi}\left[\sum_{\ell=0}^{\infty} \gamma^\ell r(s_{t+\ell})\right]$$
$$V_\pi(s_t) = \mathbb{E}_{a_t, s_{t+1}, \cdots \sim \pi}\left[\sum_{\ell=0}^{\infty} \gamma^\ell r(s_{t+\ell})\right]$$
$$A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s)$$

他の方策 $\tilde{\pi}$ の期待リターンは，次のように書ける（[3] 付録に導出あり）．

$$\eta(\tilde{\pi}) = \eta(\pi) + \mathbb{E}_{s_0, a_0, \cdots \sim \tilde{\pi}}\left[\sum_{t=0}^{\infty} \gamma^t A_\pi(s_t, a_t)\right]$$

ここで，$\mathbb{E}_{s_0,a_0,\cdots\sim\tilde{\pi}}[\cdot]$ は，$a_t \sim \tilde{\pi}(\cdot|s_t)$ を意味する．ρ_π を，（規格化していない）割引訪問頻度とする．

$$\rho_\pi(s) = \sum_{t=0}^{\infty} \gamma^t P(s_t = s)$$

これを用いて，次のように書ける．

$$\eta(\tilde{\pi}) = \eta(\pi) + \sum_s \rho_{\tilde{\pi}}(s) \sum_a \tilde{\pi}(a|s) A_\pi(s,a) \tag{5}$$

もし全ての s に対して $A_\pi(s,a) \geq 0$ となるなら，方策は必ず改善される（確定的方策で方策反復をやることを言っている）．しかしながら，近似設定では，推定と近似誤差のせいで，ある状態 s に対するアドバンテージ関数が負になることが避けられない．

式 (6) 中において，ρ が $\tilde{\pi}$ に依存しているため，これを直接的に最適化することが難しい．その代わりに，この論文では次の近似を用いる．

$$L_\pi(\tilde{\pi}) = \eta(\pi) + \sum_s \rho_\pi(s) \sum_a \tilde{\pi}(a|s) A_\pi(s,a) \tag{6}$$

この式は，ρ に関して方策の変化を無視していることに注意する．もしパラメトライズされた方策 $\pi_\theta(a|s)$ を用いるなら，L_π と η は一次まで一致する（つまり次式が成り立つ）．

$$L_{\pi_{\theta_0}}(\pi_\theta)|_{\theta=\theta_0} = \eta(\pi_\theta)|_{\theta=\theta_0}, \quad [\nabla_\theta L_{pi_\theta}(\pi_\theta)]_{\theta=\theta_0} = [\nabla_\theta \eta(\pi_\theta)]_{\theta=\theta_0}$$

この式は，もし十分小さいステップ幅で $L_{\pi_{\theta_{old}}}$ を改善すれば，η も改善されるだろう，ということを意味している．

この問題を扱うために，Kakade and Langford は，η の改善の lower bound を与えた．π_{old} を現在の方策とし，また $\pi' = \arg\max_\pi L_{\pi_{old}}(\pi')$ とする．そして，新しい方策を

$$\pi_{new}(a|s) = (1-\alpha)\pi_{old}(a|s) + \alpha\pi'(a|s) \tag{7}$$

とする．Kakade and Langford は，以下の lower bound を導いた．

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - \frac{2\epsilon\gamma}{(1-\gamma)^2}\alpha^2, \quad \epsilon = \max_s |\mathbb{E}_{a\sim\pi'(a|s)}[A_\pi(s,a)]| \tag{8}$$

（この論文では少しだけ弱いがよりシンプルな形に修正した）しかしながら，この bound は，式 (7) により生成される混合方策にしか適用することができない．

2.2 一般の確率的方策に対する単調改善保証

この論文の理論的結果は，π と $\tilde{\pi}$ の間の距離測度で式 (8) 中の α を置き換え，定数 ϵ を任意に変化させることである．この論文で特に陥る距離測度は，total variation divergence であり，離散確率分布 p, q に対しては $D_{TV}(p||q) = \frac{1}{2}\sum_i |p_i - q_i|$ である．$D_{TV}^{\max}(\pi, \tilde{\pi})$ を次のように定義する．

$$D_{TV}^{\max}(\pi, \tilde{\pi}) = \max_s D_{TV}(\pi(\cdot|s)||\tilde{\pi}(\cdot|s)) \tag{9}$$

■定理 1． 次の bound が成り立つ．

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - C\left[D_{TV}^{\max}(\pi_{old}, \pi_{new})\right]^2, \quad \frac{4\gamma \max_s |\mathbb{E}_{a\sim\pi'(a|s)}[A_\pi(s,a)]|}{(1-\gamma)^2} \tag{10}$$

証明は付録にある．

total variance divergence と KL divergence の間の関係として，$D_{TV}(p||q)^2 \leq D_{KL}(p||q)$ が成り立つことに注意する．定理 1 より，以下の bound が成り立つ．

$$\eta(\pi_{new}) \geq L_{\pi_{old}}(\pi_{new}) - CD_{TV}^{\max}(\pi_{old}, \pi_{new}) \tag{11}$$

なお，$\pi_{new} = \pi_{old}$ のときには等式が成り立つ．

i 番目の方策改善ステップにおいて，評価関数 $M_i(\pi) = L_{\pi_i}(\pi) - CD_{TV}^{\max}(\pi_i, \pi)$ を最大化するように π_{i+1} を選ぶとする．このとき，$\eta(\pi_{i+1}) - \eta(\pi_i) \geq M_i(\pi_{i+1}) - M_i(\pi_i) \geq 0$ であることから，i 番目の方策改善ステップでは η は減少しない．

2.3 パラメトライズされた方策の最適化

簡単のため，$\eta(\theta) = \eta(\pi_\theta)$，$L_\theta(\tilde{\theta}) = L_{\pi_\theta}(\tilde{\pi}_\theta)$，$D_{KL}(\theta||\tilde{\theta}) = D_{KL}(\pi_\theta||\tilde{\pi}_\theta)$ の表記を用いる．次の最大化をすることにより，真の目的 η の改善を保証することができる．

$$\max_\theta [L_{\theta_{old}}] - CD_{KL}^{\max}(\theta_{old}, \theta)$$

もし上述の理論により推薦される定数 C を用いるなら，実際にはステップサイズは非常に小さくなる．より大きなステップをロバストに取る一つの方法は，新しい方策と古い方策の間の KL ダイバージェンスの拘束を用いることである．

$$\begin{array}{ll} \min_\theta & L_{\theta_{old}}(\theta) \\ s.t. & D_{KL}^{\max}(\theta_{old}, \theta) \leq \delta \end{array} \tag{12}$$

この問題は，状態空間の全ての点において bounded な KL divergence を要求する．これは理論に動機づけられるが，この問題は拘束の数の多さのせいで解くには実際的ではない．その代わりに，平均 KL divergence を考えるヒューリスティック近似を用いる．

$$\bar{D}_{KL}^{\rho}(\theta_1, \theta_2) = \mathbb{E}_{s\sim\rho}\left[D_{KL}(\pi(\cdot|s)||\tilde{\pi}(\cdot|s))\right]$$

結局，次の問題を解く．

$$\begin{array}{ll} \min_\theta & L_{\theta_{old}}(\theta) \\ s.t. & \bar{D}_{KL}^{\rho\theta_{old}}(\theta_{old}, \theta) \leq \delta \end{array} \tag{13}$$

同様の方策更新は先行研究において提案されている．この論文の 7,8 節において，この論文のアプローチと先行研究を比較する．

2.4 目的関数と拘束のサンプルベース推定

問題 (13) を期待値により書き表すと，次のようになる（アドバンテージ関数を Q 値で置き換えるが，定数の変化なので解には影響しない）．

$$\begin{array}{ll} \min_\theta & \mathbb{E}_{s\sim\rho\theta_{old}, a\sim q}\left[\frac{\pi_\theta(a|s)}{q(a|s)} Q_{\theta_{old}}(s_n, a)\right] \\ s.t. & \mathbb{E}_{s\sim\rho\theta_{ald}}[D_{KL}(\theta_{old}, \theta)] \leq \delta \end{array} \tag{14}$$

2.5 注意

[3] の付録 C で次のように言っている．

This section describes how to efficiently approximately solve the following constrained optimization problem, which we must solve at each iteration of TRPO:

$$\max L(\theta) \text{ s.t. } \bar{D}_{KL}(\theta_old, \theta) \leq \delta \tag{55}$$

The method we will describe involves two steps: (1) compute a search direction, using a linear approximation to objective and quadratic approximation to the constraint; and (2) perform a line search in that direction, ensuring that we improve the nonlinear objective while satisfying the nonlinear constraint.

（省略）

Having computed the search direction $s \approx A^{-1}g$, we next need to compute the maximal step length β such that $\theta + \beta s$ will satisfy the KL divergence constraint. To do this, let $\delta = \bar{D}_{KL} \approx \frac{1}{2}(\beta s)^T A(\beta s) = \frac{1}{2}\beta^2 s^T As$. From this, we obtain $\beta = \sqrt{2\delta / s^T As}$, where δ is the desired KL divergence. （省略）

Last, we use a line search to ensure improvement of the surrogate objective and satisfaction of the KL divergence constraint, both of which are nonlinear in the parameter vector θ (and thus depart from the linear and quadratic approximations used to compute the step). We perform the line search on the objective $L_{\theta_{old}} - \mathcal{X}[\bar{D}_{KL}(\theta_{old}, \theta) \leq \delta]$, where $\mathcal{X}[...]$ equals zero when its argument is true and $+\infty$ when it is false. Starting with the maximal value of the step length β computed in the previous paragraph, we shrink β exponentially until the objective improves. （省略）

この操作は，Algorthm 1 とは若干異なる．具体的には，制約付き最適化問題の近似解を得て精度を評価するという部分は Algorthm 1 と同じであるが，Algorthm 1 の 14-18 行目のように p_k（TRPO の勾配 g に相当）を計算した後に更新する／しないの 2 択をしていない部分が異なる．つまり，Algorthm 1 では精度が悪い場合には更新を全くしないのだが，TRPO では精度が悪い場合にはステップ幅を縮小して更新している．そして，このステップ幅を選ぶ動作が，line search である．

[2] では line search vs trust region という文脈で書かれていたため，混乱するかもしれない（私は最初混乱した）が，TRPO の解釈をまとめると，

- ステップ p_k（TRPO の勾配 g に相当）を得る動作は制約付き最適化問題の解として得ていて，これは trust region 法に特有のアイデアである（つまり，line search には無いアイデアである）．
- ステップ p_k を使って更新する部分は TRPO において拡張されていて（更新する／しないの 2 択→ステップ幅を選んで更新），その際に line search を導入している．

（余談：TRPO 論文は（の旧版？）を引用しているものの，付録においてヘッセベクトル積やフィッシャーベクトル積の部分だけ参照しているだけである）．

2.6 まとめ

- 方策勾配法を trust region 法でやる．
- 一般に，trust region 法では，ステップを評価した後に式 (4) に基づいて更新するため，trust region 自体はどの空間で考えても（accept ratioη には依るが）ロバストに更新できるはずである．TRPO の場合には，RL の問題構造に動機づけられた確率的方策の間の KL divergence に基づいて trust region を考える．これが，なんか効率化にいいらしい．
- 勾配を得るための局所近似に対してロバストなのであって，サンプル近似に対してロバストなわけではない．
- TRPO の論文 [3] の時点では，アドバンテージ関数の代わりに Q 値を用いて勾配をサンプル推定しているアルゴリズムが示されている．ただし，その後の GAE の論文 [4] においてアドバンテージ関数のサンプル推定が可能になったため，いま TRPO と名がついている実装例のほとんどは，GAE と組み合わされたものである（Q 値よりアドバンテージ関数の方が良い）．

3 GAE: Generalized Advantage Function [4]

3.1 準備

初期状態 s_0 は，分布 ρ_0 からサンプルされる．軌道 $(s_0, a_0, s_1, a_1, \cdots)$ は，終端状態にたどり着くまで，方策 $a_t \sim \pi(a_t|s_t)$ に従って行動をサンプリングし，またダイナミクス $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$ に従って状態をサンプリングすることにより生成される．報酬 $r_t = r(s_t, a_t, s_{t+1})$ は，各時間ステップにおいて受け取る．目標は，期待総報酬 $\sum_{t=0}^{\infty} r_t$ を最大化すことであり，全ての方策に対して有限であると仮定する．なお，問題設定の一部として割引を定義しない．以下において，バイアス-バリアンス（分散）トレードオフを調整するアルゴリズムパラメータとして，割引率が現れる．

方策個賠法は，勾配 $g := \nabla_\theta \mathbb{E}[\sum_{t=0}^{\infty} r_t]$ を繰り返し推定することにより，期待総報酬を最大化する．方策勾配に対する複数の関連する表現が存在する．

$$g = \mathbb{E}_{s_{0:\infty},\, a_{0:\infty}} \left[\sum_{t=0}^{\infty} \Psi_t \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

ここで，Ψ_t は，次のものがあり得る．

1. $\sum_{\ell=0}^{\infty} r_\ell$：初期時刻からの軌道の総報酬.
2. $\sum_{\ell=0}^{\infty} r_{t+\ell}$：時刻 t において (s_t, a_t) から始まる軌道の総報酬.
3. $\sum_{\ell=0}^{\infty} r_{t+\ell} - b(s_t)$：上からベースラインを引いたもの.
4. $Q^\pi(s_t, a_t)$：Q 値.
5. $A^\pi(s_t, a_t) := Q^\pi(s_t, a_t) - V^\pi(s_t)$：アドバンテージ関数.
6. $r_t + V^\pi(s_{t+1}) - V^\pi(s_t)$：TD 残差.

ここで，価値関数は次のように定義される．

$$V^\pi(s_t) := \mathbb{E}_{s_{t+1:\infty},\, a_{t:\infty}} \left[\sum_{\ell=0}^{\infty} r_{t+\ell} \right], \quad Q^\pi(s_t, a_t) := \mathbb{E}_{s_{t+1:\infty},\, a_{t+1:\infty}} \left[\sum_{\ell=0}^{\infty} r_{t+\ell} \right]$$

$\Psi_t = A^\pi(s_t, a_t)$ とすることにより分散をほぼ最小にすることができるが，実際にはアドバンテージ関数は未知で推定されなければならない．

パラメータ γ を導入する．これにより，バイアスを導入してしまうことを引き換えにして，遠い報酬の重みを小さくすることにより分散を低くすることができる．（注意：我々の言葉で言うとすれば，「割引を大きくするとJ値は小さくなり，またその結果としてサンプル推定の分散も小さくなるが，元々解きたい問題からどんどん離れていく」となる）このパラメータは，MDP の割引定式において用いられる割引率に対応するが，この論文では割引問題における分散低減パラメータとして扱う．割引価値関数は，次により与えられる．

$$V^{\pi,\gamma}(s_t) := \mathbb{E}_{s_{t+1:\infty},\, a_{t:\infty}} \left[\sum_{\ell=0}^{\infty} \gamma^\ell r_{t+\ell} \right], \quad Q^{\pi,\gamma}(s_t, a_t) := \mathbb{E}_{s_{t+1:\infty},\, a_{t+1:\infty}} \left[\sum_{\ell=0}^{\infty} \gamma^\ell r_{t+\ell} \right]$$

また，割引アドバンテージ関数 $A^{\pi,\gamma}(s_t, a_t) := Q^{\pi,\gamma}(s_t, a_t) - V^{\pi,\gamma}(s_t)$ とする．方策勾配の割引近似は，次のように定義される．

$$g^\gamma = \mathbb{E}_{s_{0:\infty},\, a_{0:\infty}} \left[\sum_{t=0}^{\infty} A^{\pi,\gamma}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

以下では，バイアスされた（しかしバイアスされすぎてはいない）$A^{\pi,\gamma}$ に対する推定をどのように得るかを議論する．

3.2 アドバンテージ関数推定

関数 V の TD 残差を，$\delta_t := -V(s_t) + [r_t + \gamma V(s_{t+1})]$ で定義する．もし価値関数が $V = V^{\pi,\gamma}$ という条件を満たす（この論文では γ-just と呼ぶ．割引問題において方策勾配がバイアスしない／不偏であることの条件）なら，

$$\mathbb{E}_{s_{t+1}}\left[\delta_t\right] = -V^{\pi,\gamma}(s_t) + \mathbb{E}_{s_{t+1}}\left[r(s_t, a_t, s_{t+1}) + \gamma V^{\pi,\gamma}(s_{t+1})\right] = -V^{\pi,\gamma}(s_t) + Q^{\pi,\gamma}(s_t, a_t) = A^{\pi,\gamma}(s_t, a_t)$$

となる．よって，もし $V = V^{\pi,\gamma}$ なら（このときのみであるが），δ_t はアドバンテージ関数の不偏推定量である．δ_t の k 個の割引和は，次のように書ける．

$$\hat{A}_t^{(k)} := \sum_{\ell=0}^{k-1} \gamma^\ell \delta_t = -V(s_t) + \left[\sum_{\ell=0}^{k-1} \gamma^\ell r_t + \gamma^k V(s_{t+k})\right]$$

この式は，k ステップリターンの推定からベースライン $-V(s_t)$ を引いたものである．$\delta_t = \hat{A}_t^{(1)}$ との類似で，$\hat{A}_t^{(k)}$ をアドバンテージ関数の推定として考えよう．ただし，$k \to \infty$ とするにつれて，バイアスは大きくなる．なお，$k \to \infty$ では，$\hat{A}_t^{(\infty)} = -V(s_t) + \sum_{\ell=0}^{\infty} \gamma^\ell r_t$ となり，経験リターンから価値関数ベースラインを引いたものが得られる．

一般化アドバンテージ関数 (GAE) を，次のように定義する．

$$\hat{A}_t^{GAE(\gamma,\lambda)} := (1-\lambda) \sum_{k=0}^{\infty} \lambda^k \hat{A}_t^{(k)} = \sum_{t=0}^{\infty} (\gamma\lambda)^\ell \delta_{t+\ell} \tag{15}$$

これは，価値関数の推定のための TD(λ) を，アドバンテージ関数に対応させたものである．$\lambda = 0,\ 1$ の時には，次が得られる．

$$\hat{A}_t^{GAE(\gamma,0)} = -V(s_t) + r_t + \gamma V(s_{t+1})$$
$$\hat{A}_t^{GAE(\gamma,1)} = -V(s_t) + \sum_{\ell=0}^{\infty} \gamma^\ell r_t$$

$\hat{A}_t^{GAE(\gamma,0)}$ は，どのような V に対してもバイアスを引き起こさないが，分散が大きい．$\hat{A}_t^{GAE(\gamma,1)}$ は，$V = V^{\pi,\gamma}$ の時のみバイアスせず，それ以外ではバイアスを引き起こすが，低い分散を持つ．

3.3 価値関数近似

価値関数を表すための非線形関数近似器を用いる場合，最も単純なアプローチは非線形回帰問題を解くことである．

$$\min_{\phi} \sum_{n=1}^{N} ||V_\phi(s_n) - \hat{V}_n||^2$$

ここで，$\hat{V}_t = \sum_{\ell=0}^{\infty} \gamma^\ell r_{t+\ell}$ は，報酬の割引総和であり，n は軌道のバッチにおける全ての時間ステップにわたる添え字である（ここでの n はバッチの添え字ではないことに注意！つまり，エピソードについての $t = 1, \cdots, \infty$ を縦に並べて順番付けしたものがここでの n である）．

論文 [4] の実験では，バッチ最適化の各反復法において価値関数を，trust region 法を用いて最適化する．trust region 法は，データの一番最近のバッチにオーバーフィッチングすることを避ける助けとなる．trust

Algorithm 2 TRPO with GAE

方策パラメータ θ_0 と価値観すパラメータ ϕ_0 を初期化する．
for $i = 0, 1, \cdots$ **do**
　N 時間ステップが得られるまで，現在の方策 π_θ をシミュレートする．
　$V = V_\phi$ を用いて，全ての時刻 $t \in \{1, 2, \cdots\}$ における δ_t^V を計算する．
　$\hat{A}_t = \sum_{\ell=0}^{\infty} (\gamma\lambda)^\ell \delta_{t+\ell}^V$ を全ての時間ステップに対して計算する．
　式 (18) の TRPO 更新により θ_{i+1} を計算する．
　式 (17) により ϕ_{i+1} を計算する．
end for

region 問題を定式化するために，まず $\sigma^2 = \frac{1}{N}\sum_{n=1}^{N} ||V_{\phi_{old}}(s_n) - \hat{V}_n||^2$ を計算する．ここで，ϕ_{old} は，最適化前のパラメータである．そうして，次の制約付き最適化問題を解く．

$$\begin{array}{ll} \min_\phi & \sum_{n=1}^{N} ||V_\phi(s_n) - \hat{V}_n||^2 \\ s.t. & \frac{1}{N}\sum_{n=1}^{N} \frac{||V_\phi(s_n) - \hat{V}_{\phi_{old}}(s_n)||^2}{2\sigma^2} \leq \epsilon \end{array} \tag{16}$$

この拘束は，価値関数は平均 $V_\phi(s)$ で分散 σ^2 を持つ条件付きガウス分布をパラメトライズするように取るとした場合の，以前の価値関数と新しい価値関数の間の平均 KL ダイバージェンスが ϵ よりも小さくなる拘束と等価である．

共役勾配アルゴリズムを用いて trust region 問題の近似解を計算する．具体的には，次の 2 次計画問題を解く．

$$\begin{array}{ll} \min_\phi & g^T(\phi_- \phi_{old}) \\ s.t. & (\phi_- \phi_{old})^T \left[\frac{1}{N}\sum_n j_n j_n^T\right]_{\phi_{old}} (\phi_- \phi_{old}) \leq \epsilon \end{array} \tag{17}$$

ここで，g は目的関数 $\sum_{n=1}^{N} ||V_\phi(s_n) - \hat{V}_n||^2$ の勾配であり，また $j_n = \nabla_\phi V_\phi(s_n)$ である．（式 (16) から式 (17) の導出は後述）

3.4　方策最適化アルゴリズム

GAE は様々な方策勾配法に対して用いることができるが，この論文では TRPO[3] を用いて方策を更新する．TRPO は，各反復において次の制約付き最適化問題を解く．

$$\begin{array}{ll} \min_\theta & \frac{1}{N}\sum_{n=1}^{N} \frac{\pi_\theta(a_n|s_n)}{\pi_{\theta_{old}}(a_n|s_n)} \hat{A}_n \\ s.t. & \frac{1}{N}\sum_{n=1}^{N} D_{KL}\left(\pi_{\theta_{old}}(\cdot|s_n)||\pi_\theta(\cdot|s_n)\right) \leq \epsilon \end{array} \tag{18}$$

TRPO の論文 [3] に示されているように，目的関数を線形化・制約を 2 次化することによりこの問題を近似的に解く．これは，$\theta - \theta_{old} \propto -F^{-1}g$ の方向のステップを得て，ここで F は平均 Fisher 情報行列，g は方策勾配推定である．この方策更新は，自然方策勾配や natural actor-critic と同じステップ方向を得るが，異なるステップサイズ決定スキームと計算手法を用いる．

方策と価値関数を反復的に更新する全体的なアルゴリズムは, Algorithm 2 に与えられる．方策更新 $\theta_i \to \theta_{i+1}$ は価値関数 V_{ϕ_i} を用いて行われて，$V_{\phi_{i+1}}$ ではないことに注意する．もし価値関数を先に更新すると，追加的なバイアスが導入されるだろう．これを見るために，価値関数がオーバーフィットした極端な状況を考えると，Bellman 残差 $r_t + \gamma V(s_{t+1}) - V(s_t)$ は全ての時間ステップにおいてゼロになり，つまり方策勾配もゼロになる．

3.5 自分のための補足

■KL ダイバージェンス． 連続確率変数の場合，一般に，KL ダイバージェンスは次のように定義される．

$$KL(q(\boldsymbol{u})||p(\boldsymbol{u})) = \int q(\boldsymbol{u}) \ln \frac{q(\boldsymbol{u})}{p(\boldsymbol{u})} d\boldsymbol{u} = \int q(\boldsymbol{u})[\ln q(\boldsymbol{u}) - \ln p(\boldsymbol{u})] d\boldsymbol{u}$$

正規分布 $q(u) = \mathcal{N}(\boldsymbol{u}|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$，$p(u) = \mathcal{N}(\boldsymbol{u}|\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ とすると，

$$\begin{aligned}
&KL(q(\boldsymbol{u})||p(\boldsymbol{u})) = \int q(\boldsymbol{u})[\ln q(\boldsymbol{u}) - \ln p(\boldsymbol{u})] d\boldsymbol{u} \\
&= \frac{1}{2} \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} + \int q(\boldsymbol{u}) \left[-\frac{1}{2} (\boldsymbol{u} - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_q^{-1} (\boldsymbol{u} - \boldsymbol{\mu}_q) - \frac{1}{2} (\boldsymbol{u} - \boldsymbol{\mu}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{u} - \boldsymbol{\mu}_p) \right] d\boldsymbol{u} \\
&= \frac{1}{2} \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - \frac{1}{2} \int q(\boldsymbol{u}) \mathrm{tr} \left[\boldsymbol{\Sigma}_q^{-1} (\boldsymbol{u} - \boldsymbol{\mu}_q)(\boldsymbol{u} - \boldsymbol{\mu}_q)^T - \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{u} - \boldsymbol{\mu}_p)(\boldsymbol{u} - \boldsymbol{\mu}_p)^T \right] d\boldsymbol{u} \\
&= \frac{1}{2} \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - \frac{1}{2} \mathrm{tr}[\boldsymbol{\Sigma}_q^{-1} \boldsymbol{\Sigma}_q] + \frac{1}{2} \int q(\boldsymbol{u}) \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (\boldsymbol{u}\boldsymbol{u}^T - 2\boldsymbol{\mu}_p \boldsymbol{u}^T + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) \right] d\boldsymbol{u} \\
&= \frac{1}{2} \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - \frac{D}{2} + \frac{1}{2} \mathrm{tr} \int q(\boldsymbol{u}) \left[\boldsymbol{\Sigma}_p^{-1} \boldsymbol{u}\boldsymbol{u}^T \right] d\boldsymbol{u} + \frac{1}{2} \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (-2\boldsymbol{\mu}_p \boldsymbol{\mu}_q^T + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) \right] d\boldsymbol{u} \\
&= \frac{1}{2} \left\{ \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - D + \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (-2\boldsymbol{\mu}_p \boldsymbol{\mu}_q^T + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) \right] + \int q(\boldsymbol{u}) \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} [(\boldsymbol{u} - \boldsymbol{\mu}_q)(\boldsymbol{u} - \boldsymbol{\mu}_q)^T + 2\boldsymbol{\mu}_q \boldsymbol{u}^T - \boldsymbol{\mu}_q \boldsymbol{\mu}_q^T] \right] d\boldsymbol{u} \right\} \\
&= \frac{1}{2} \left\{ \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - D + \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (-2\boldsymbol{\mu}_p \boldsymbol{\mu}_q^T + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T) \right] + \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\Sigma}_q + 2\boldsymbol{\mu}_q \boldsymbol{\mu}_q^T - \boldsymbol{\mu}_q \boldsymbol{\mu}_q^T) \right] \right\} \\
&= \frac{1}{2} \left\{ \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - D + \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\Sigma}_q + \boldsymbol{\mu}_p \boldsymbol{\mu}_p^T - 2\boldsymbol{\mu}_p \boldsymbol{\mu}_q^T + \boldsymbol{\mu}_q \boldsymbol{\mu}_q^T) \right] \right\} \\
&= \frac{1}{2} \left\{ \ln \frac{\det \boldsymbol{\Sigma}_p}{\det \boldsymbol{\Sigma}_q} - D + \mathrm{tr} \left[\boldsymbol{\Sigma}_p^{-1} \boldsymbol{\Sigma}_q \right] + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) \right\}
\end{aligned}$$

（注：トレースの性質 $\mathrm{tr}(X + Y) = \mathrm{tr}(X) + \mathrm{tr}(Y)$ と $\mathrm{tr}(XY) = \mathrm{tr}(YX$ を使った）（よく使う式で，例えばノート [1] の最後の節に導出が乗っている）

1 変数の場合は，$q(u) = \mathcal{N}(u|\mu_q, \sigma_q^2)$ と $p(u) = \mathcal{N}(u|\mu_p, \sigma_p^2)$ として，

$$\begin{aligned}
D_{KL}(q(u)||p(u)) &= \int_{-\infty}^{\infty} q(u) \ln \frac{q(u)}{p(u)} du \\
&= \int_{-\infty}^{\infty} q(u) \left(-\frac{(u - \mu_q)^2}{2\sigma_q^2} - \frac{1}{2} \ln(2\pi\sigma_q^2) + \frac{(u - \mu_p)^2}{2\sigma_p^2} + \frac{1}{2} \ln(2\pi\sigma_p^2) \right) du \\
&= -\left(\frac{(\mu_q^2 + \sigma_q^2) - 2\mu_q^2 + \mu_q^2}{2\sigma_q^2} \right) + \left(\frac{(\mu_q^2 + \sigma_q^2) - 2\mu_q \mu_p + \mu_p^2}{2\sigma_p^2} \right) + \frac{1}{2} \ln \frac{\sigma_p^2}{\sigma_q^2} \\
&= -\frac{1}{2} + \left(\frac{\sigma_q^2 + (\mu_q - \mu_p)^2}{2\sigma_p^2} \right) + \frac{1}{2} \ln \frac{\sigma_p^2}{\sigma_q^2} \\
&= \frac{1}{2} \left\{ \ln \frac{\sigma_p^2}{\sigma_q^2} - 1 + \left(\frac{\sigma_q^2}{\sigma_p^2} \right) + \left(\frac{(\mu_p - \mu_q)^2}{\sigma_p^2} \right) \right\}
\end{aligned}$$

平均 μ_i を更新することだけを考える場合には，KL ダイバージェンスの中で $\frac{(\mu_q - \mu_p)^2}{2\sigma_p^2}$ の項以外は関係ない．式 (16) は，更新するパラメータに関係ない項を ϵ の中に押し込んで書いてある．

■式 (16) 拘束条件の Taylor 展開． $\lambda = 1$ としたアドバンテージ関数の推定は，論文 [4] の Eq.18 より，$\hat{A}_t^{GAE(\gamma,1)} = \sum_{\ell=0}^{\infty} \gamma^\ell r_{t+\ell} - V(s_t) = \hat{V}_t - V(s_t)$ と得られる．ここで，$V(s) = V^{\pi,\gamma}(s)$ として，現在の価値関

数近似器 $V_{\phi_{old}}(s)$ を用いると，

$$\sum_{n=1}^{N} ||V_\phi(s_n) - \hat{V}_n||^2 = \sum_{n=1}^{N} ||V_\phi(s_n) - \left[\hat{A}_n^{GAE(\gamma,1)} + V_{\phi_{old}}(s_n)\right]||^2$$

となる（この論文内では $\lambda = 1$ の場合を実装したが，一般化して $\hat{V}_t^\lambda = \sum \ell = 0^\infty (\gamma\lambda)^\ell r_{t+\ell} = \hat{A}_t^{GAE(\gamma,1)} + V_{\phi_{old}}(s_n)$ でも良いとのこと）.

3.6 まとめ

- パラメトライズされた方策表現と，非線形価値関数近似表現を用いる.
- TD 残差を用いて，一般化アドバンテージ関数のサンプル近似が式 (15) で得られる.
- TD 残差を用いて，非線形価値関数近似器を更新する（with trust region 法）
- Algorithm 2.

参考文献

[1] John Duchi. Derivations for linear algebra and optimization. *Berkeley, California*, 3, 2007.

[2] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

[3] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897, 2015.

[4] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *International Conference on Learning Representation*, 2016.