# 内発的報酬

## Motivation

一般的な強化学習アルゴリズムでは、最初に環境内の各 state の価値関数を本来の評価値以外で初期化を行う。報酬を発見する前のエージェントは、適切でない価値関数をもとに行動するので、ランダムな探索を行っているに等しい。一度報酬が見つかれば、学習アルゴリズムによって各状態の評価に正確な報酬の値が加えられ、正しい評価値を用いた方策更新が可能になる。つまり最初の報酬発見が非常に重要であり、それまではランダムな探索を行うしかない。これが簡単なタスクであれば問題にならないが、報酬がスパースな問題設定の場合は最初の報酬発見に時間がかかり、いつまで経っても正しい方策更新が出来ない。
例えばゲームの非常に広大なダンジョンを考えよう。報酬はマップと比べて非常に疎な間隔に位置されている。このとき、通常のエージェントなら何も報酬が与えられていないなら分かれ道をランダムに決定するだろう。しかし人間なら既に進んで目的に達しなかったルートは除外して、行ったことがないルートを試すだろう。このような未知の部分への優先的探索を強化学習にも取り込むことができないか? というのが強化学習における内発的報酬の考え方である。これは好奇心に近く、一部手法は curiosity driven exploration (好奇心による探索) とも言われる。

## 内発的報酬

内発的報酬 (Intrinsic Reward) とは、外的な報酬に依存せず何らかの基準でエージェント自身が生成する報酬である。通常の報酬は環境で本来の目的を達成したときに獲得される外発的報酬 (Extrinsic Reward) である。

メモ: pseudo-count → ICM → RND

### Count based

状態への訪問回数を数えて、それに応じて訪問回数の少ない状態へ向かうような行動を取るようにすればよさそうである。ある状態 $s$ で選択した行動 $a$ の回数を $n(s,a)$ とする。$n(s,a)$ に反比例して内発的報酬を与えると、$n(s,a)$ が少ない(新規性が高い)状態遷移の価値評価が高まる。これを定式化すると以下のように成る。

$$\hat{Q}(s,a) = \hat{R}(s,a) + \gamma \sum_{s'} \hat{T}(s'|s,a) \max_{a'} \hat{Q}(s',a') + \frac{\beta}{\sqrt{n(s,a)}}$$

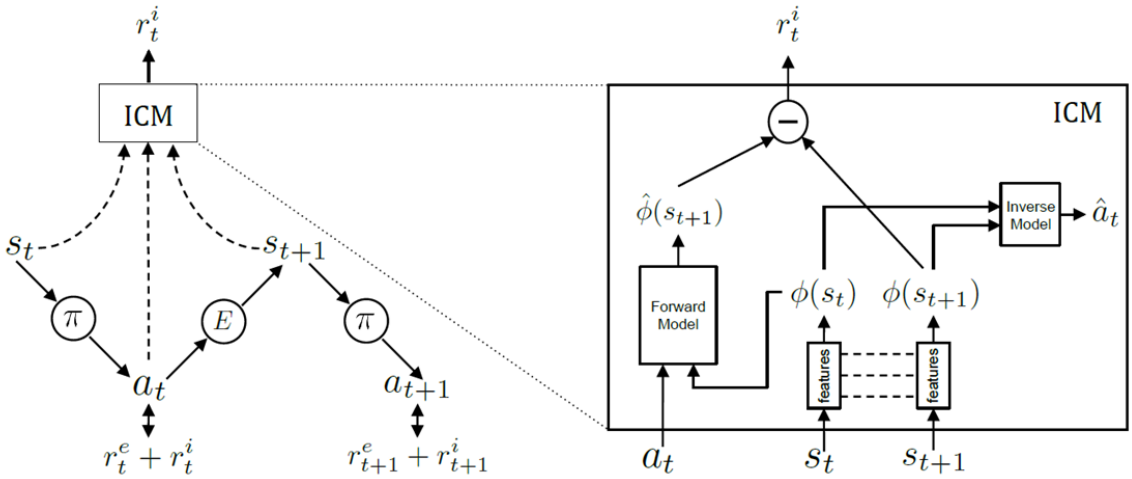
確かに今まで行ったことがない状態へ向かうことは出来そうだ。しかし、状態数があまりにも多すぎたり、状態空間が連続の場合は殆どの状態カウントが 0 になってしまい意味がない。
そこで、一つ一つの状態の訪問回数をカウントせず、画像ピクセル単位で見た画像の類似度、発生確率から擬似的に状態をカウントする Pseudo-Count という手法がある。

### 予測誤差

観測 $x_t$ と、その時選択する行動 $a_t$ から、次の観測 $x_{t+1}$ がどうなるか予測するモデル $f(x_t, a_t)$ を考える(順モデル)。モデルの出力と、実際に $x_t, a_t$ を選択した場合の次の観測 $x_{t+1}$ を用いて二乗誤差を計算し、NNを学習する。既に観測した遷移は予測誤差が高くなり(二乗誤差が小さくなる)、観測が少ない遷移は予測精度が低くなる(二乗誤差が大きい)。この予測誤差を内発的報酬とすれば、未知状態への探索を促進できそうである。

### ICM

ICM(Intrinsic Curiosity Module) では、エージェントの行動に関係があるもののみに注目する特徴抽出器を逆モデルを用いて学習し、予測誤差によって内発的報酬を生成する機構をつける。




順モデル (Forward Model) は、$\phi(s_t), a_t$ を入力し、次の状態の $\hat{\phi}(s_{t+1})$ を出力する。逆モデルは状態 $s_t, s_{t+1}$ から獲得された特徴 $\phi(s_t), \phi(s_{t+1})$ を入力すると $\hat{a}(t)$ を出力する。特徴空間で動作するので、高次元入力にも対応可能である。

## RND

RND(Random Network Distillation) では2つのDNNから内部報酬を生成する。

$$
\begin{aligned}
\mathbf{t} &= f_1(s) \\
\mathbf{y} &= f_2(s) \\
r^i &= ||\mathbf{t} - \mathbf{y}||^2
\end{aligned}
$$

$f_1$ は target network, $f_2$ は predictor network と呼ばれている。target network は学習せず、predictor network は target network の出力を教師データとして学習する。これによって、Noisy TV Problem を回避することができる。

## NGU

エピソード内で新規性を判定するモジュール (episodic novelity module) とエピソードをまたいで新規性を判定するモジュール (life-long novelty module) を導入して、各モジュールで生成した報酬の合計を最終的な内発的報酬とする。