Sprawozdanie z Ćwiczenia 6 z WSI

Adam Szostek nr. indeksu 331443

8 stycznia 2025

1 Implementowany algorytm

1.1 Q-learning

Implementowany algorytm to q-learning, który jest jednym z najpopularniejszych, jeśli chodzi o uczenie ze wzmocnieniem. Głównym celem algorytmu jest skonstruowanie tabeli Q, przypisującej wartości każdej możliwej akcji dostępnej dla agenta w danym stanie.

Algorithm 1: Algorytm Q-learning

Data: t_{max}, γ, β, e_{max}
Result: Tabela Q
Funkcja `Q-learning`(t_{max}, γ, β, e_{max})**:**
  $Q_0 \leftarrow$ zainicjuj;
  $e \leftarrow 0$;
  while $e < e_{max}$ **do**
    $t \leftarrow 0$;
    $x_t \leftarrow$ zainicjuj;
    while $t < t_{max}$ ***and*** $x_t \notin$ *stany absorbujące* **do**
      $a_t \leftarrow$ wybierz akcję(x_t, Q_t);
      $r_t, x_{t+1} \leftarrow$ wykonaj akcję a_t;
      $\Delta \leftarrow r_t + \gamma \max_a Q_t(x_{t+1}, a) - Q_t(x_t, a_t)$;
      $Q_{t+1} \leftarrow Q_t + \beta\Delta$;
      $t \leftarrow t + 1$;
    $e \leftarrow e + 1$;

1.2 Strategie wyboru akcji

Zaimplementowano trzy strategie wyboru następnej akcji:

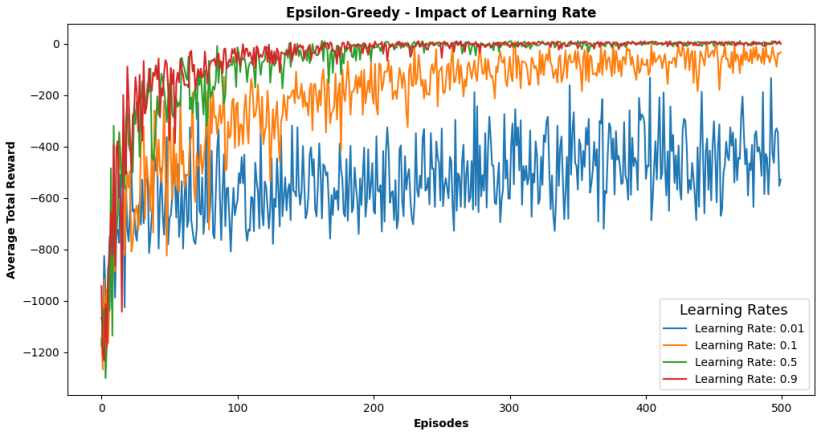
- **Epsilon-Greedy Strategy**: Strategia `Epsilon-Greedy` polega na wyborze losowej akcji z prawdopodobieństwem ϵ, a w przeciwnym przypadku (z prawdopodobieństwem $1 - \epsilon$) na wyborze akcji, która ma największą wartość w tabeli Q dla danego stanu. Jest to popularna metoda balansująca eksplorację (losowe działania) i eksploatację (wybór najlepszej znanej akcji).
- **Boltzmann Strategy**: Strategia `Boltzmann` (nazywana także softmax) wybiera akcję na podstawie rozkładu prawdopodobieństwa zależnego od wartości Q danej akcji i parametru temperatury. Im wyższa wartość Q dla danej akcji, tym większe prawdopodobieństwo jej wyboru. Temperatura (T) kontroluje stopień eksploracji: im wyższa temperatura, tym bardziej losowe stają się wybory.

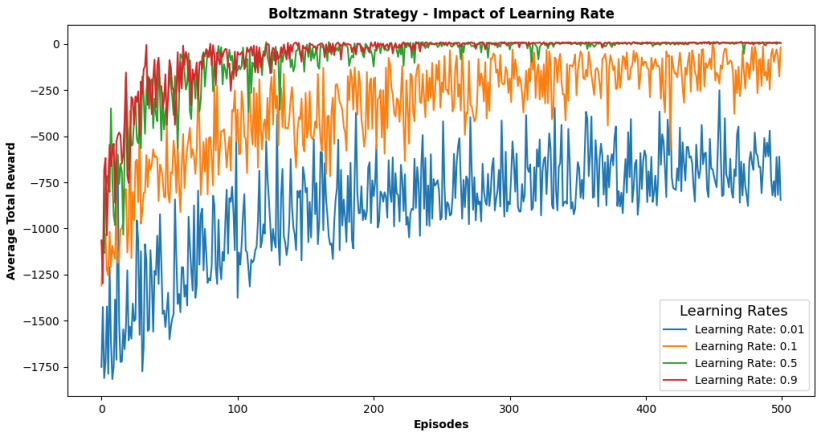
- **Count-Based Strategy**: Strategia `Count-Based` modyfikuje wartości Q w zależności od liczby odwiedzin danej akcji w określonym stanie. Działa na zasadzie premiowania rzadziej wybieranych akcji poprzez przyznawanie im dodatkowych punktów w tabeli Q, co motywuje agenta do ich eksploracji.

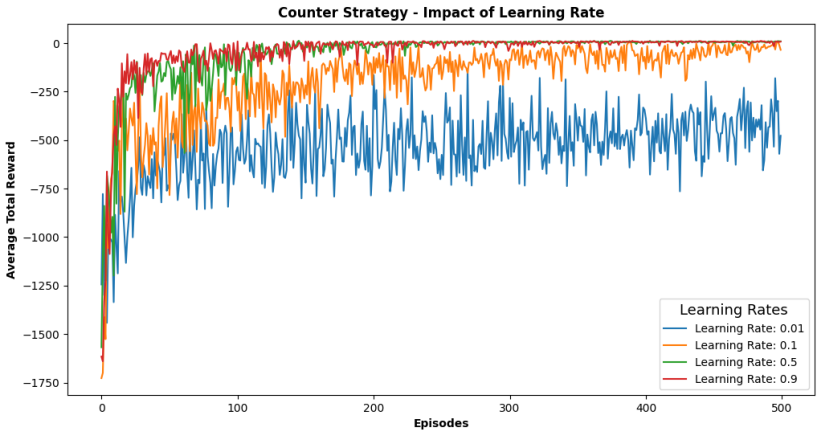
2 Planowane eksperymenty

Q-learning został przetestowany w środowisku Taxi-v3. Jest to środowisko z biblioteki OpenAI Gym, które symuluje problem transportu taksówki w mieście. Celem agenta (taksówki) jest dowożenie pasażerów z jednego punktu do innego, ucząc się optymalnej strategii na podstawie nagród. Agent otrzymuje pozytywne nagrody za dostarczenie pasażera do wyznaczonego celu, natomiast jest karany za nieudane próby oraz nieefektywne działania, takie jak poruszanie się bez wyraźnego celu. Przy użyciu tego środowiska zbadano efektywność trzech zaimplementowanych strategii wyboru akcji oraz wpływu współczynnika uczenia na algorytm. W tym celu sporządzono wykresy skumulowanych nagród za epizod dla agenta w zależności od ilości epizodów treningowych. Każda krzywa jest średnią 5 prób trenowania agenta.

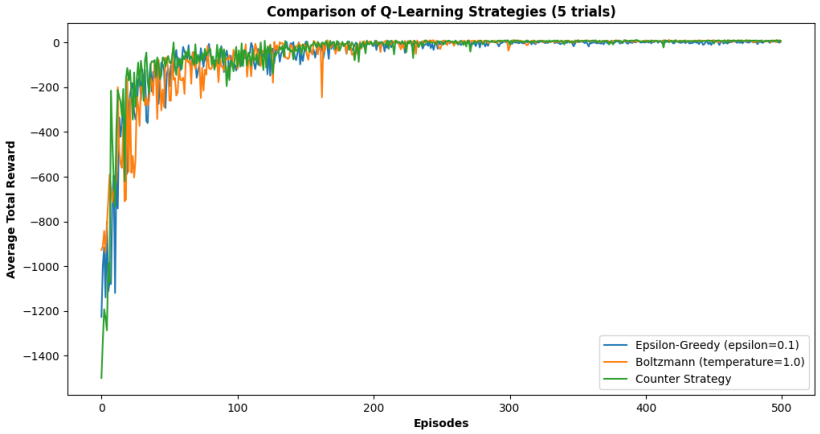
3 Uzyskane wyniki







Na wykresach badających wpływ współczynnika nauczania na algorytm q-learning widać, że dla każdej strategii najlepsze wyniki osiągają 0.9 i 0.5, natomiast pozostałe wartości mimo, iż początkowo również zdają się dążyć do optimum, zapewniają dużo mniej stabilne i ogółem gorsze wyniki. W dalszych badaniach używano Learning Rate = 0.9.



Na powyższym wykresie widać, że testowane strategie osiągają niemal identyczne wyniki. Jedyną zauważalną różnicą jest początkowa szybkość konwergencji krzywych do optimum – strategie licznikowa oraz epsilon-zachłanna osiągają je nieco szybciej niż strategia Boltzmanna.

4 Wnioski z wyników

Na podstawie przeprowadzonych eksperymentów numerycznych można stwierdzić, że współczynnik nauczania ma ogromny wpływ na działanie algorytmu q-learning. Dobór zbyt małego współczynnika sprawia, że agent osiąga dużo gorsze średnie nagrody na epizod, mimo zastosowania takiej samej strategii. Jednocześnie widać również, że dla wartości bliższych 0.9 krzywe są dużo stabilniejsze, tj. nie obserwujemy ich gwałtownej oscylacji, która ma miejsce przy mniejszych wartościach. Oscylacje te mogą wynikać z ograniczonej zdolności agenta do adaptacji na podstawie nowych informacji, co skutkuje wyborem suboptymalnych akcji, często prowadzących do ujemnych nagród.
Analizując porównanie strategii można natomiast stwierdzić, że wynik osiągnięty po ustalonej liczbie epizodów jest praktycznie identyczny. Jak zauważono przy opisie wyników,

początkowo strategia Boltzmanna wolniej konwerguje do optimum. Powodem takiego zachowania może być fakt, że pozostałe dwie strategie mają dużo prostsze mechanizmy eksploracji co pozwala agentowi szybciej zidentyfikować dobre dla danych stanów działania. W przypadku epsilon-zachłannej optymalne rozwiązanie może zostać znalezione szybciej, ale z uwagi na bardziej deterministyczne działanie akcje o wyższym potencjale mogą być przegapione. W środowisku testowym z większą liczbą stanów i akcji strategia licznikowa mogłaby okazać się najefektywniejsza ze względu na mechanizm premiujący rzadziej eksplorowane działania. Natomiast w środowiskach, gdzie kluczowa jest eksploracja i nagrody są trudniejsze do przewidzenia optymalny może być wybór podejścia Boltzmanna. Warto napomnieć, że dla strategii Boltzmanna i epsilon-zachłannej istotny jest dobór parametrów (temperatury t dla boltzmanna i epsilon dla epsilon zachłannej), który ma duży wpływ na ich działanie. Np. dla zbyt duzych wartości epsilon podejście to będzie wybierało zbyt często ruchy losowe co nie pozwoli agentowi osiągać optymalnych nagród.
Podsumowując, w długoterminowych eksperymentach wszystkie strategie konwergują do zbliżonego poziomu średnich nagród, co świadczy o ich skuteczności. Niemniej jednak istnieją scenariusze, w których jedna ze strategii może okazać się bardziej optymalna od pozostałych. Natomiast hiperparametr learning rate ma kluczowy wpływ na algorytm nie zależnie od wybranej strategii.