Mateusz Kołacz, 336360

**WSI – Ćwiczenie 6**

Zadanie

Proszę zaimplementować algorytm Q-Learning i użyć go do wyznaczenia polityki decyzyjnej dla problemu [FrozenLake8x8](https://gymnasium.farama.org/environments/toy_text/frozen_lake/). W problemie tym celem agenta jest przedostanie się przez zamarznięte jezioro z domu do celu, unikając dziur (zawsze rozpoczynamy epizod z górnego lewego rogu mapy, który ma współrzędne 0). Symulator dla tego problemu można pobrać w sposób typowy dla pythona:

Obraz zawierający tekst, Czcionka, zrzut ekranu, biały

Opis wygenerowany automatycznie

Na początku proszę się zająć wersją bez poślizgu:

Obraz zawierający tekst, Czcionka, zrzut ekranu, linia

Opis wygenerowany automatycznie

W tym scenariuszu wystarcza 1000 epizodów, maksymalnie 200 kroków na epizod.

Dla każdego epizodu, dla minimum 25 niezależnych uruchomień uczenia Q-learning, wyliczyć i zwizualizować na wykresie średnie wartości oryginalnych nagród (tych z gym) w funkcji numeru epizodu. Poniżej szkic kodu:

Obraz zawierający tekst, zrzut ekranu, Czcionka, algebra

Opis wygenerowany automatycznie

Obliczenia należy powtórzyć dla zmienionej na potrzeby eksperymentu wersji algorytmu, po czym należy narysować wyniki (2 krzywe na jednym wykresie):

Obraz zawierający tekst, Czcionka, zrzut ekranu, numer

Opis wygenerowany automatycznie

Powinno wyjść coś takiego:

Obraz zawierający diagram, Wykres, zrzut ekranu, linia

Opis wygenerowany automatycznie

Dane zwizualizowane na niebiesko powstały przy identycznych ustawieniach jak te pokazane na czerwono, widać więc znaczny wpływ losowości pomimo uśrednienia.

Zwracam uwagę, że na wykresie każda z osi ma określoną skalę i interpretację, a zatem pomimo stosowania wewnątrz algorytmu własnej funkcji wykres musi powstać w oparciu o funkcję oryginalną.

Należy porównać wyniki domyślnego sposobu nagradzania (1 za dojście do celu, 0 w przeciwnym przypadku) z dwoma własnymi systemami nagród i kar. Propozycje te powinny częściej niż oryginalny system dawać niezerową informację zwrotną agentowi. Można łatwo dowiedzieć się co się stało na podstawie oryginalnych informacji z gym, np. if done and r == 0: wpadł do dziury; if state == next\_state: poszedł w ścianę.

W drugiej części badań należy włączyć poślizg (is\_slippery=True), należy zwiększyć liczbę epizodów do 10000. Jakie są zmiany w stosunku do wersji bez poślizgu? Jak często udaje się dość do celu? Jak zaproponowane wcześniej systemy nagród wpływają na wyniki w odniesieniu do wyników podstawowego systemu oceny? Spisać wnioski.

**Wyniki**

[ TODO ]