MITOCARU IRINA

343C1

Tema2

Q-Learning

Contents

[Detalii de implementare 3](#_Toc38320336)

[1. Generare harta 3](#_Toc38320337)

[2. Strategia MaxFirst 4](#_Toc38320338)

[3. Strategia Random 4](#_Toc38320339)

[4. Strategia Exploration 4](#_Toc38320340)

[5. Strategia Balanced Exploration / Exploitation 4](#_Toc38320341)

[6. Recompense 5](#_Toc38320342)

[Vizualizarea hartilor folosite la grafice 6](#_Toc38320343)

[1.1. Harta mica (5 x 5) 6](#_Toc38320344)

[1.2. Harta medie (8 x 8) 6](#_Toc38320345)

[1.3. Harta mare (10 x 20) 6](#_Toc38320346)

[Harta mica (5 x 5) 6](#_Toc38320347)

[1. 500, 1000 si 3000 episoade 7](#_Toc38320348)

[1.1. MaxFirst 7](#_Toc38320349)

[1.2. Exploration 8](#_Toc38320350)

[1.3. Random 10](#_Toc38320351)

[1.4. Balanced Exploration / Exploitation 12](#_Toc38320352)

[2. Rata de invatare mica. Factor de discount mic – mediu – mare 13](#_Toc38320353)

[3. Rata de invatare mare. Factor de discount mediu – mare 15](#_Toc38320354)

[4. Factor de discount mic. Rata de invatare mica - medie - mare 16](#_Toc38320355)

[5. Factor de discount mare. Rata de invatare medie - mare 17](#_Toc38320356)

[6. Tabela de utilitati MaxFirst 18](#_Toc38320357)

[7. MaxFirst vs Random 20](#_Toc38320358)

[Harta medie (8 x 8) 22](#_Toc38320359)

[1. 500, 1000 si 3000 episoade 22](#_Toc38320360)

[1.1. MaxFirst 22](#_Toc38320361)

[1.2. Exploration 22](#_Toc38320362)

[1.3. Random 22](#_Toc38320363)

[1.4. Balanced Exploration / Exploitation 22](#_Toc38320364)

[2. Rata de invatare mica. Factor de discount mic – mediu – mare 22](#_Toc38320365)

[3. Rata de invatare mare. Factor de discount mediu – mare 22](#_Toc38320366)

[4. Factor de discount mic. Rata de invatare mica - medie - mare 22](#_Toc38320367)

[5. Factor de discount mare. Rata de invatare medie - mare 22](#_Toc38320368)

[6. Tabela de utilitati MaxFirst 22](#_Toc38320369)

[7. MaxFirst vs Random 22](#_Toc38320370)

[Harta mare (10 x 20) 22](#_Toc38320371)

[1. 500, 1000 si 3000 episoade 22](#_Toc38320372)

[1.1. MaxFirst 22](#_Toc38320373)

[1.2. Exploration 22](#_Toc38320374)

[1.3. Random 22](#_Toc38320375)

[1.4. Balanced Exploration / Exploitation 22](#_Toc38320376)

[2. Rata de invatare mica. Factor de discount mic – mediu – mare 22](#_Toc38320377)

[3. Rata de invatare mare. Factor de discount mediu – mare 22](#_Toc38320378)

[4. Factor de discount mic. Rata de invatare mica - medie - mare 22](#_Toc38320379)

[5. Factor de discount mare. Rata de invatare medie - mare 22](#_Toc38320380)

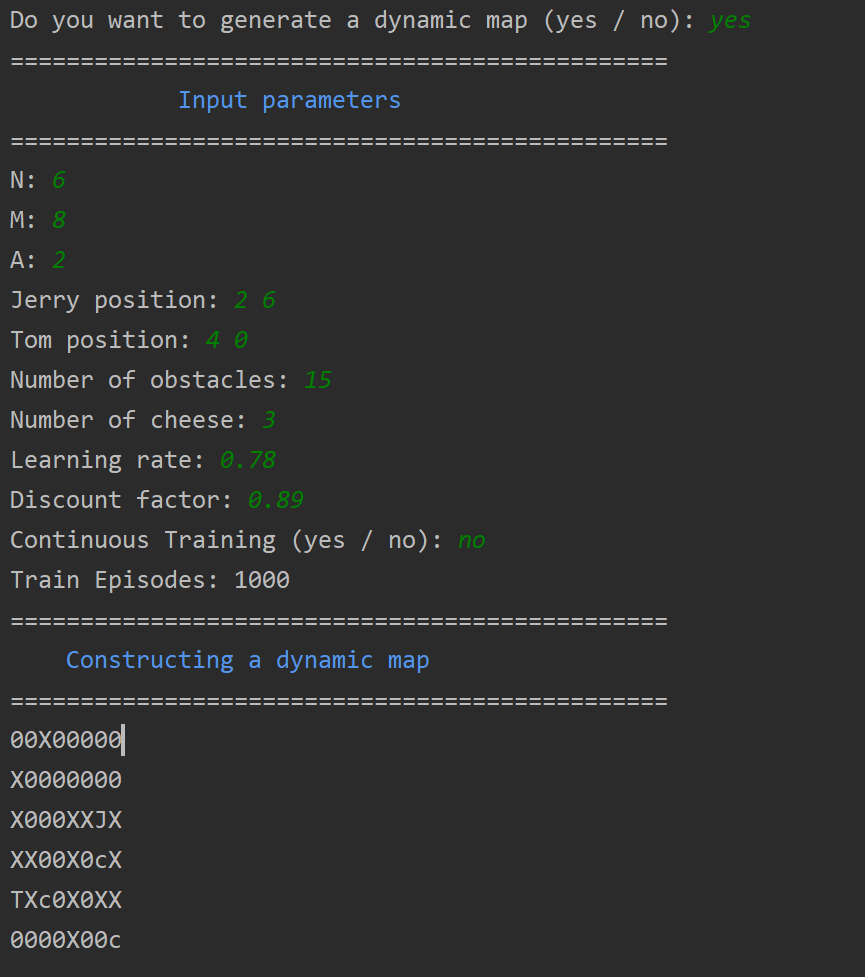
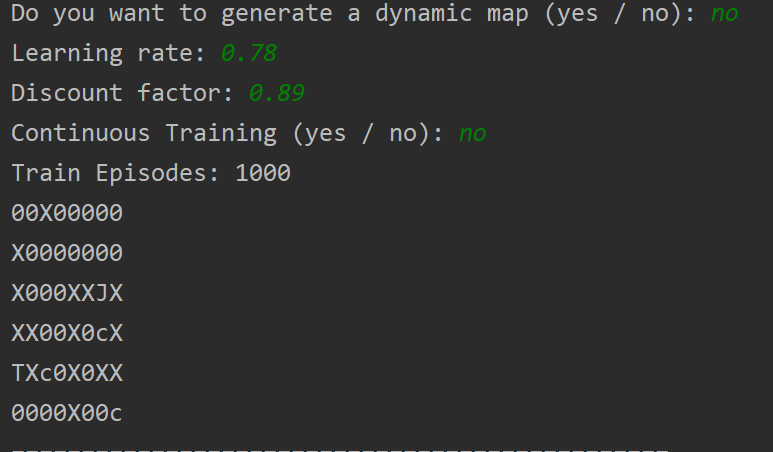
[6. Tabela de utilitati MaxFirst 22](#_Toc38320381)

[7. MaxFirst vs Random 22](#_Toc38320382)

[Concluzii 22](#_Toc38320383)

# Detalii de implementare

## Generare harta

Harta poate fi atat generata dinamic, folosind valorile de input primite ca argumente, cat si citita direct din fisier. Daca se genereaza dinamic, atunci este posibil ca tot procesul sa dureze mai mult deoarece probabilitatea ca obiectele (pisica, soarecele si bucatile de branza) sa fie blocate este mai mare. De asemenea, cu cat dimensiunea hartii este mai mare, cu atat generarea este mai costisitoare. Se aplica BFS pentru a verifica ca exista cale de la soarece la toate bucatile de branza si de la pisica la soarece.

Citire harta din fisier

Generare harta

## Strategia MaxFirst

Daca exista actiuni explorate:

* Alege una din actiunile maxime
* Verifica daca actiunea respectiva nu a fost deja luata de prea multe ori (agentul e blocat intr-o bucla), caz in care returneaza o actiune permisa oarecare
* Returneaza una din actiunile maxime

Daca nu exista actiuni explorate:

* Returneaza o actiune permisa oarecare

## Strategia Random

Daca exista mai mult de o actiune legala:

* Alege una din actiuni
* Verifica daca actiunea respectiva nu a fost deja luata de prea multe ori (agentul e blocat intr-o bucla), caz in care returneaza o alta actiune permisa

Altfel, returneaza aleator o actiune permisa

## Strategia Exploration

Daca exista mai mult de o actiune neexplorata:

* Alege una din actiuni
* Verifica daca actiunea respectiva nu a fost deja luata de prea multe ori (agentul e blocat intr-o bucla), caz in care returneaza o alta actiune permisa neexplorata

Altfel, returneaza o actiune explorata aleator

## Strategia Balanced Exploration / Exploitation

Se normalizeaza utilitatile (intre 0 si 1). Valorile negative sau actiunile neexplorate au valoarea 0.

Se alege actiunea cu valoarea cea mai apropriata de valoarea generate random.

Daca nu exista o actiune apropriata, se alege a actiune permisa aleator.

Pentru fiecare din cele 4 strategii, dupa antrenare, se poate alege rularea jocului pas cu pas, ca in figura de mai jos.



## Recompense

Daca pisica mananca soarecele sau soarcele intalneste pisica, episodul se termina si recompensa este de -10.

Pentru fiecare bucata de branza gasita de soarece, recompensa este de 10. Astfel, fiecare joc poate avea un castig maxim = nr bucati de branza \* 10 + alte recompense.

Pentru fiecare miscare aleatoare care nu aduce pisica prea aproape si care nu viziteaza de prea multe ori celulele deja vizitate, recompensa este de 0.1.

Daca pisica se aproprie mai putin de A pasi (inclusiv), se returneaza recompensa de -0.2. La fel si daca soarcele viziteaza de prea multe ori aceeasi celula (adica s-a blocat intr-un ciclu) sau daca pisica gaseste o bucata de branza.

# Vizualizarea hartilor folosite la grafice

## Harta mica (5 x 5)

5 5

0101020000101020000111000

2

0 4

3 0

0X0XJ

c0000

X0X0c

T000X

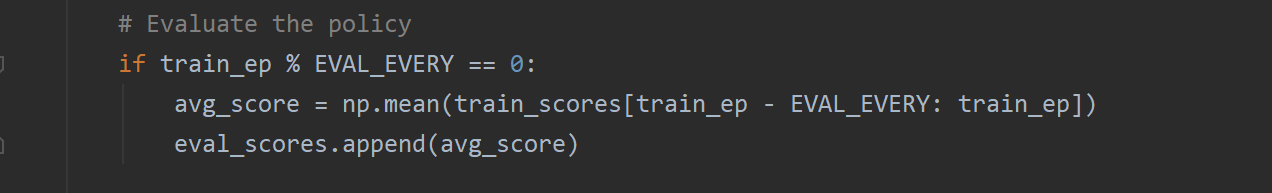
XX000

## Harta medie (8 x 8)

## Harta mare (10 x 20)

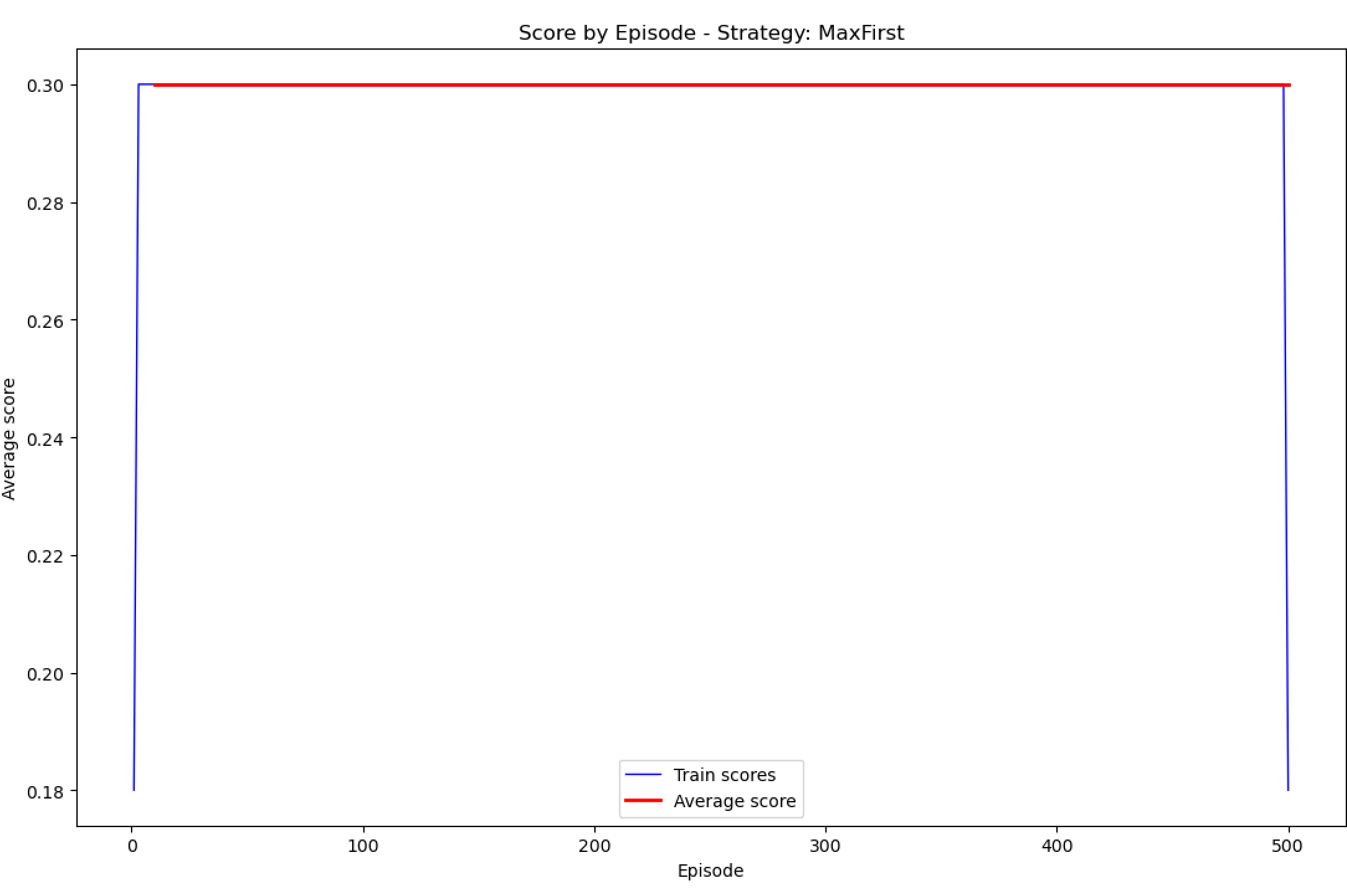
# Harta mica (5 x 5)

Graficele afiseaza atat scorul in timpul invatarii pentru fiecare episod (train scores), cat si media din 10 in 10 episoade a scorurilor (average score).

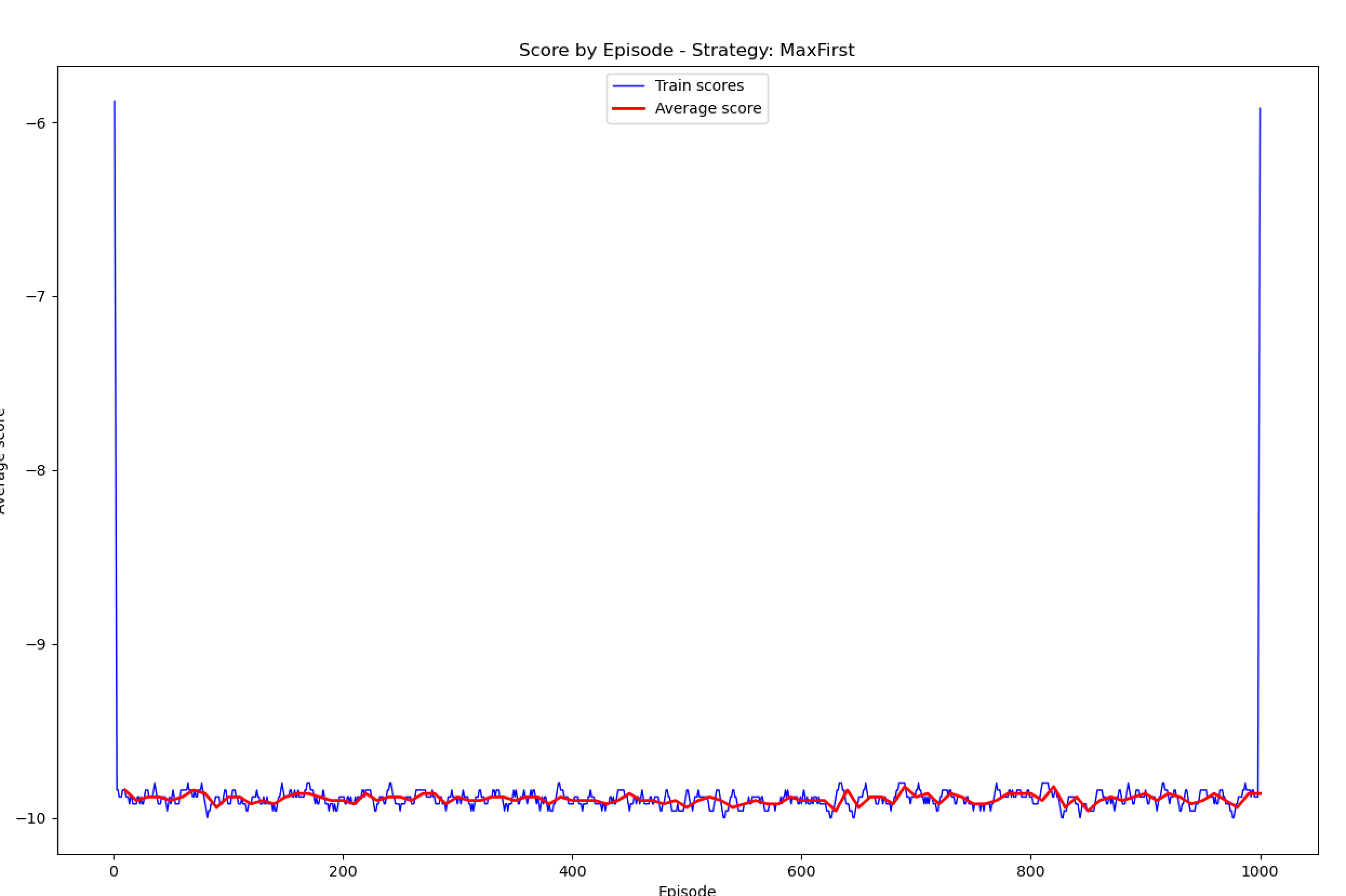


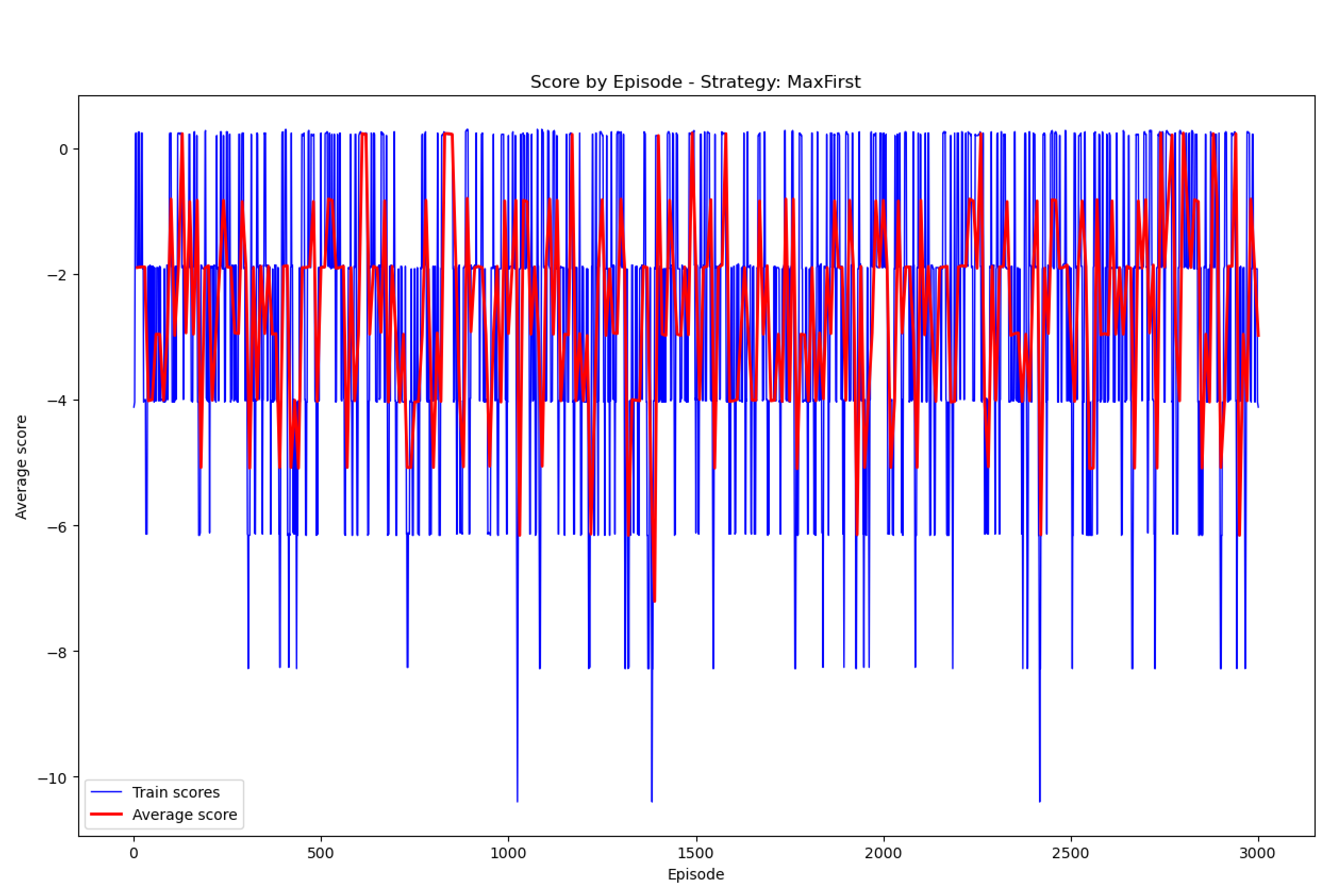
## 500, 1000 si 3000 episoade

### MaxFirst



Desi scorurile nu sunt negative, soarecele are o abordare constanta pe parcusul invatarii deoarece actiunile maxime nu il avantajeaza neaparat in configuratia actuala a hartii. Soarecele pare sa invete locatia primei recompense (cea de jos), dupa care incearca mai multe actiuni, insa pisica il ajunge din urma si e mancat. De aceea, recompensa de la bucata de branza e eliminata de cea data de pierderea jocului si scorurile vazute in graf sunt date de simplele miscari cu recompense de 0.1 fiecare.

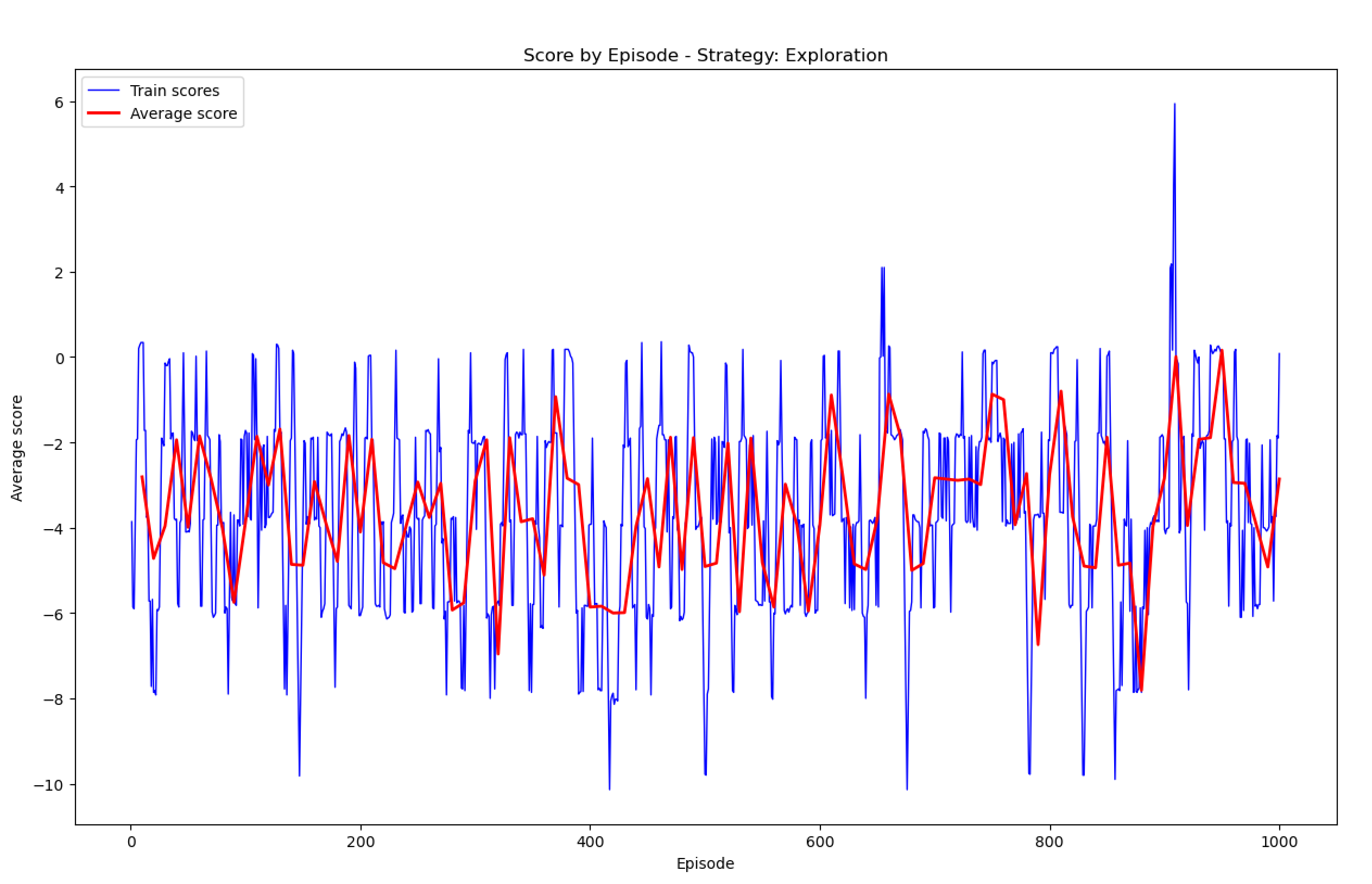




### Exploration

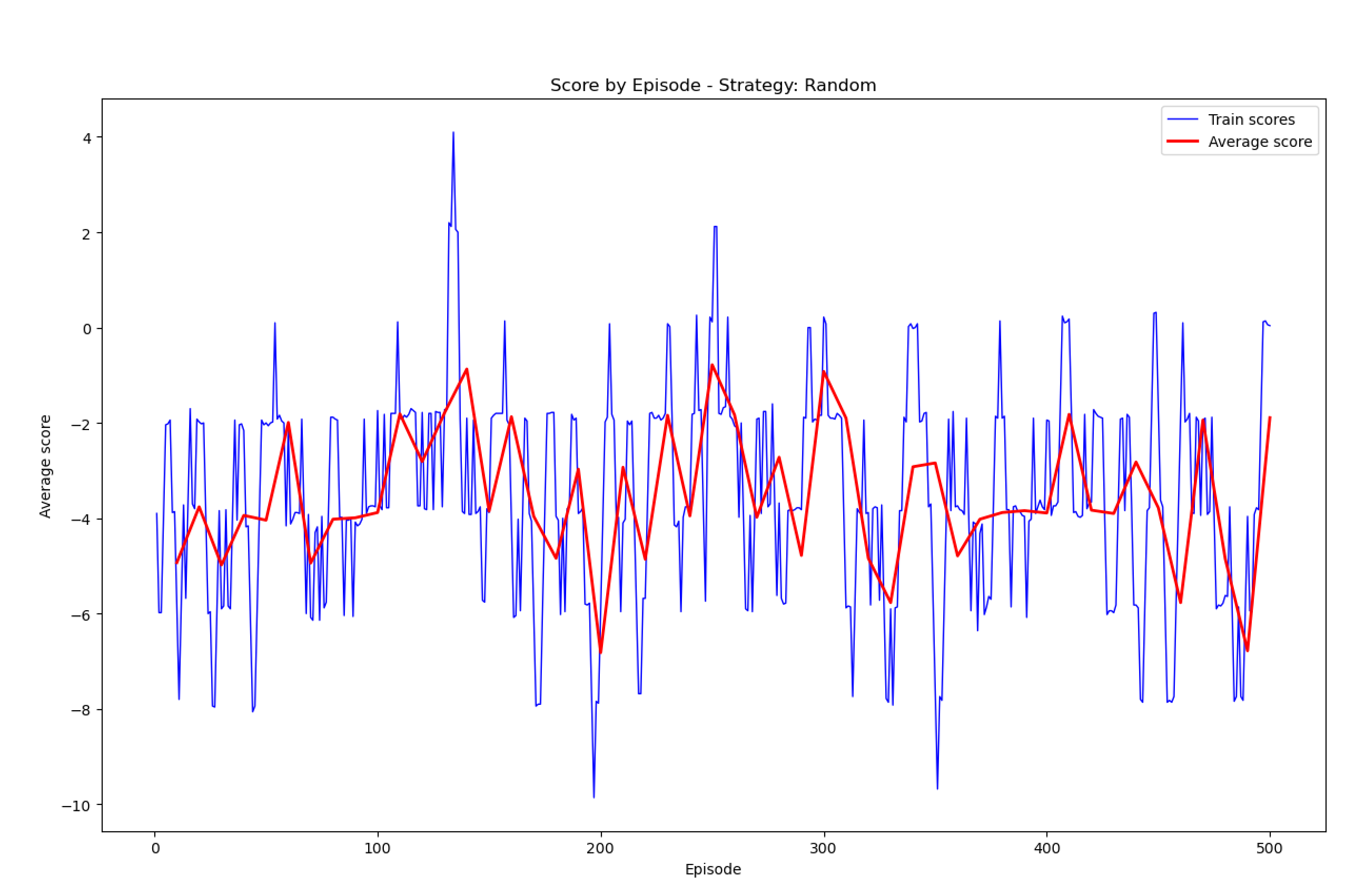


Explorarea in acest caz nu e deloc avantajoasa deoarece soarecele nu ia actiuni bune, evitand bucatile de branza in timp ce pisica se aproprie din ce in ce mai mult. Exista si episoade in care soarecele castiga, insa de cele mai multe ori pierde.



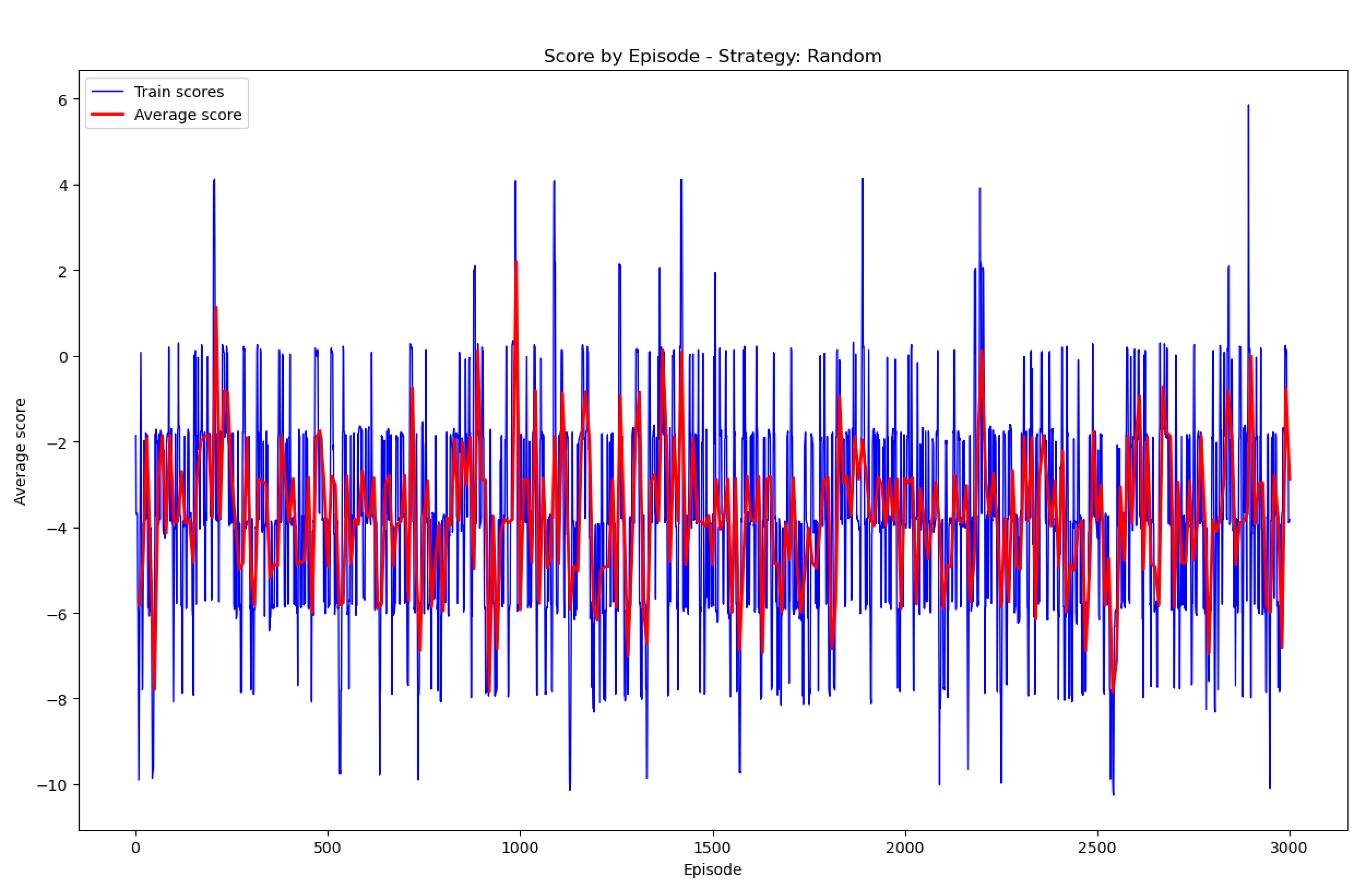


### Random

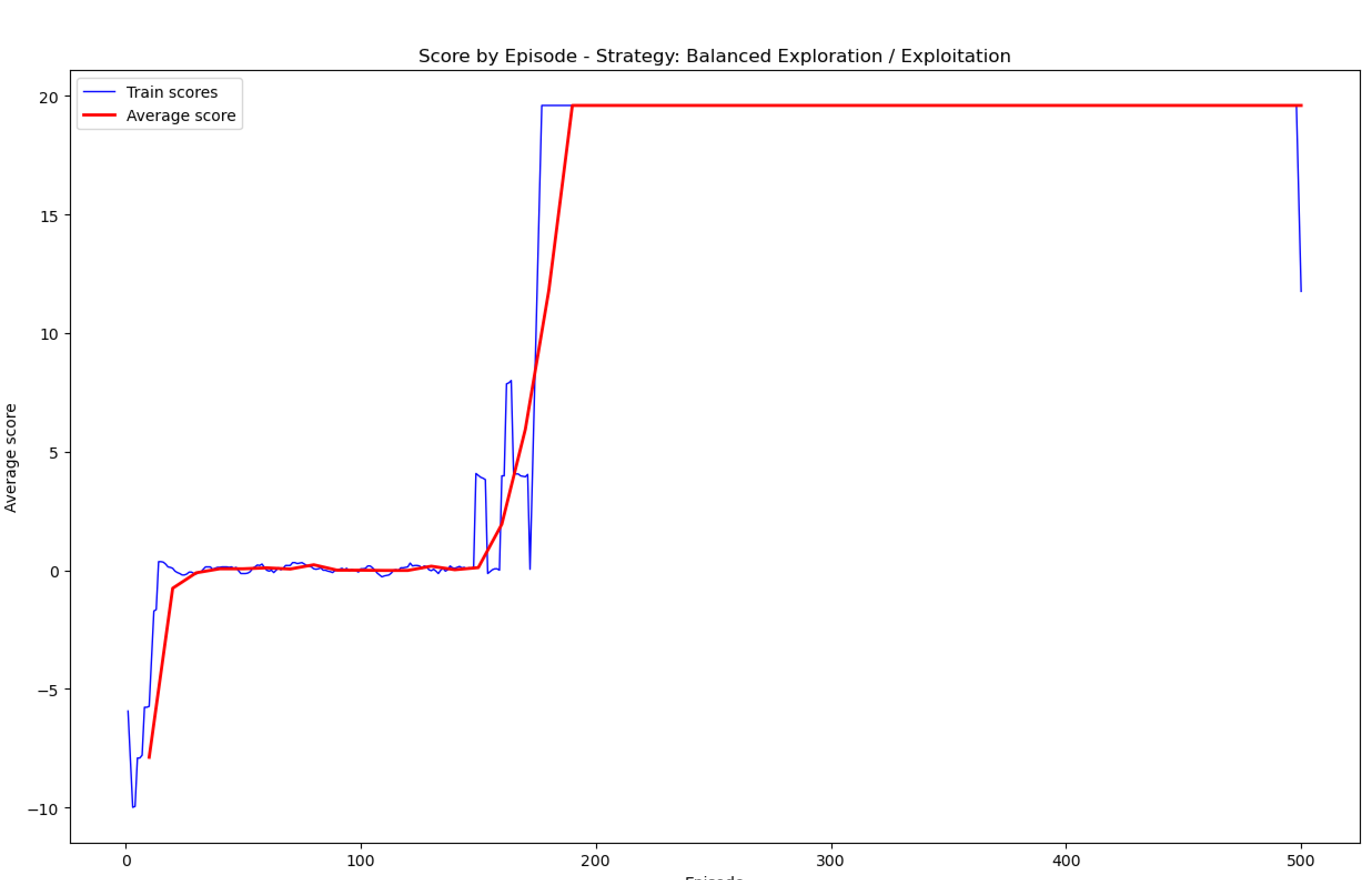


Nici in cazul alegerii aleatoare, netinand cont de tabela de utilitati, scorurile nu sunt bune. Ba chiar sunt mai proaste decat la strategia anterioara. Pare ca cu cat harta e mai mica, cu atat actiunile luate au o pondere mai mare (adica conteaza foarte mult ce actiune se face).

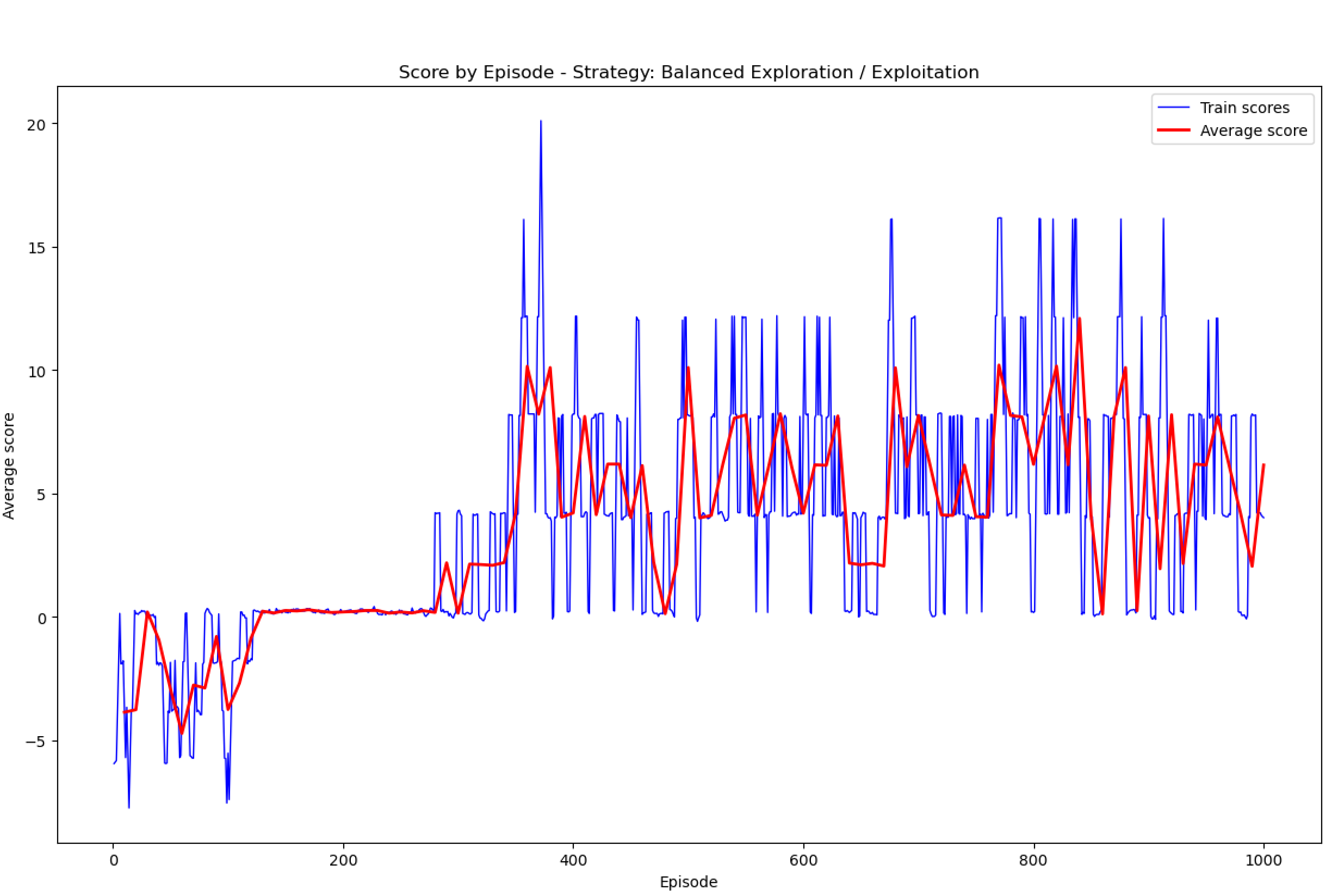


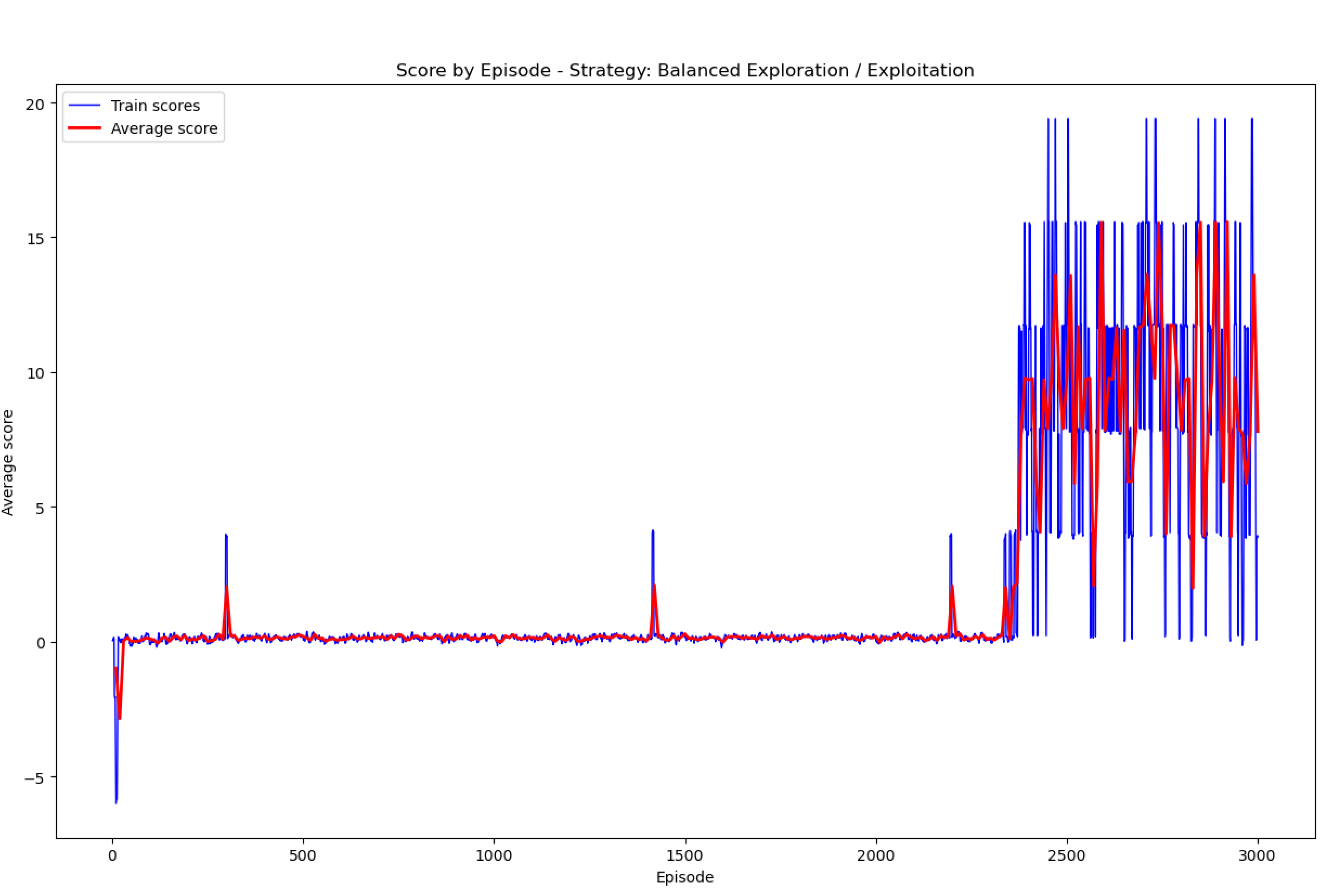


### Balanced Exploration / Exploitation

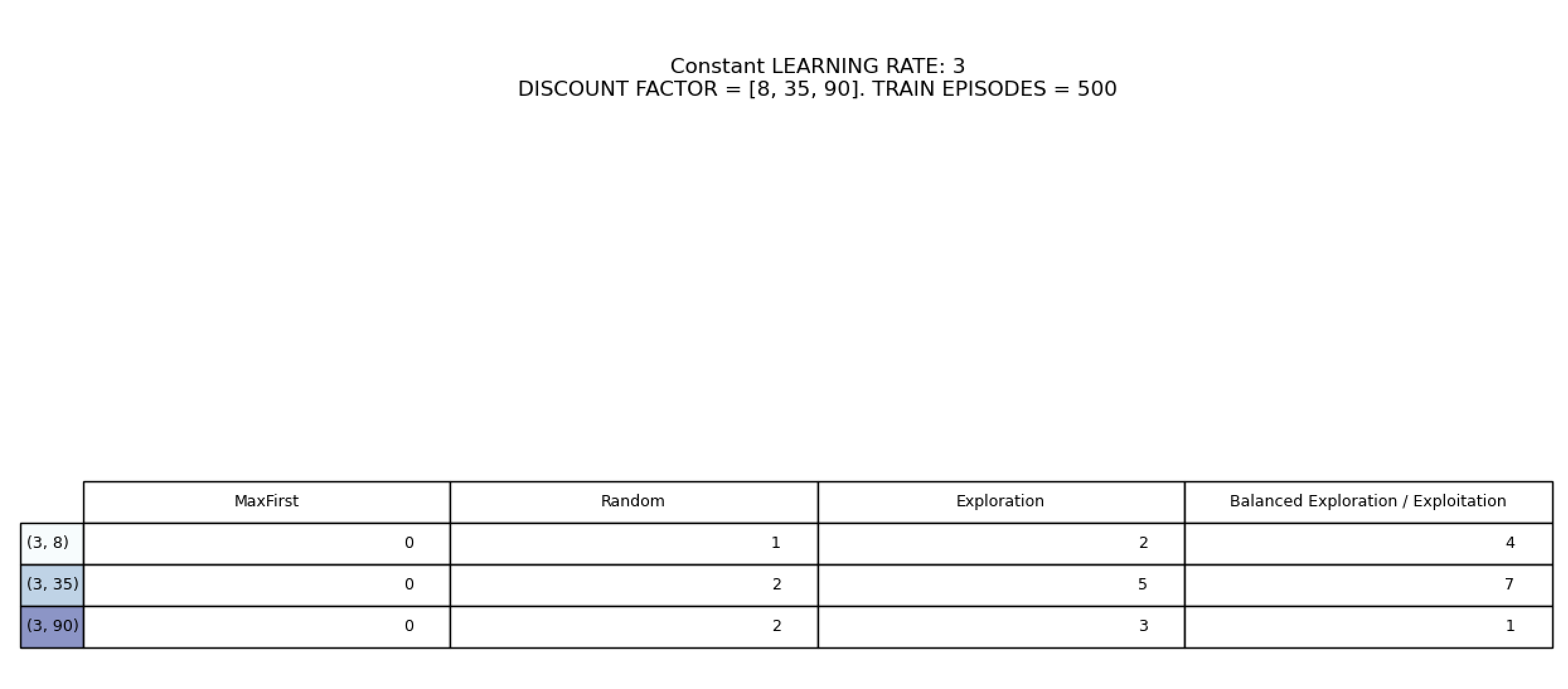


In cazul strategiei balansate, agentul pare sa invete din ce in ce mai bine, ajungand ca dupa 200 de episoade sa perfectioneze jocurile.



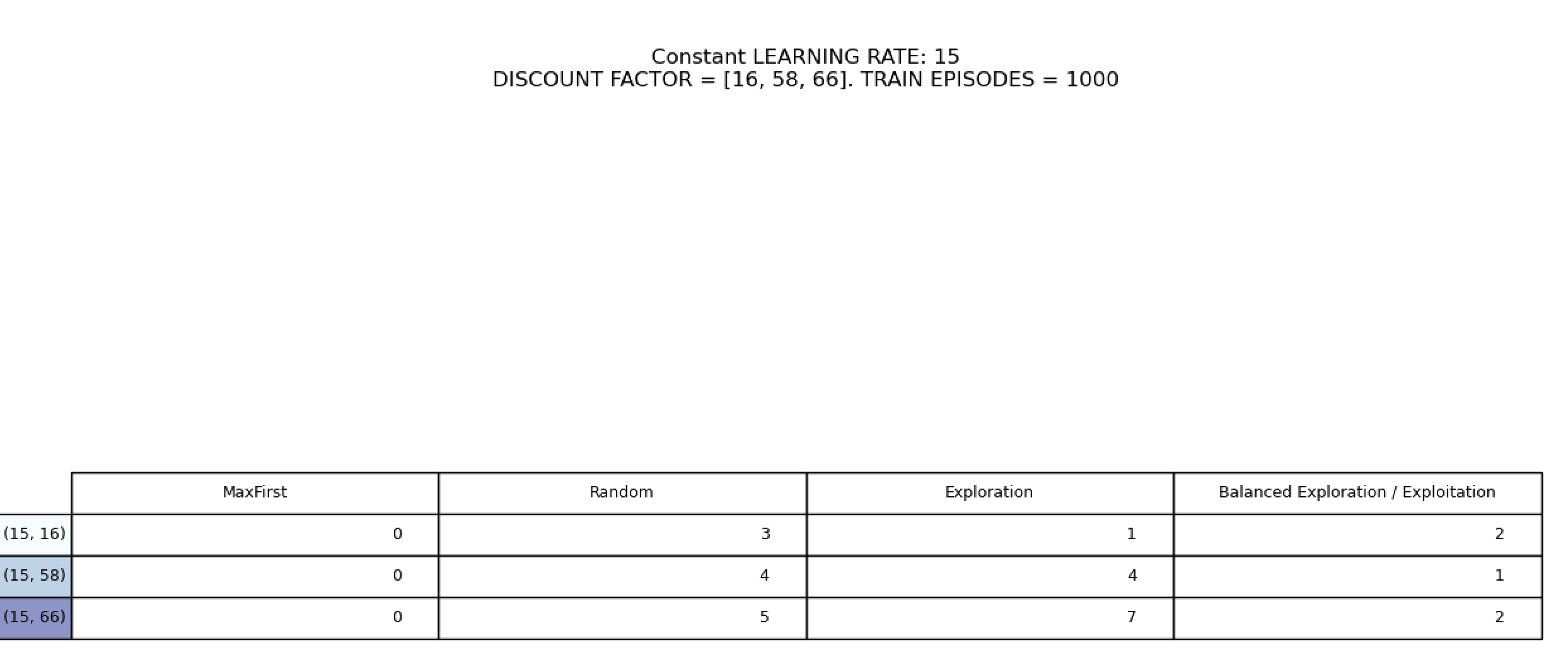


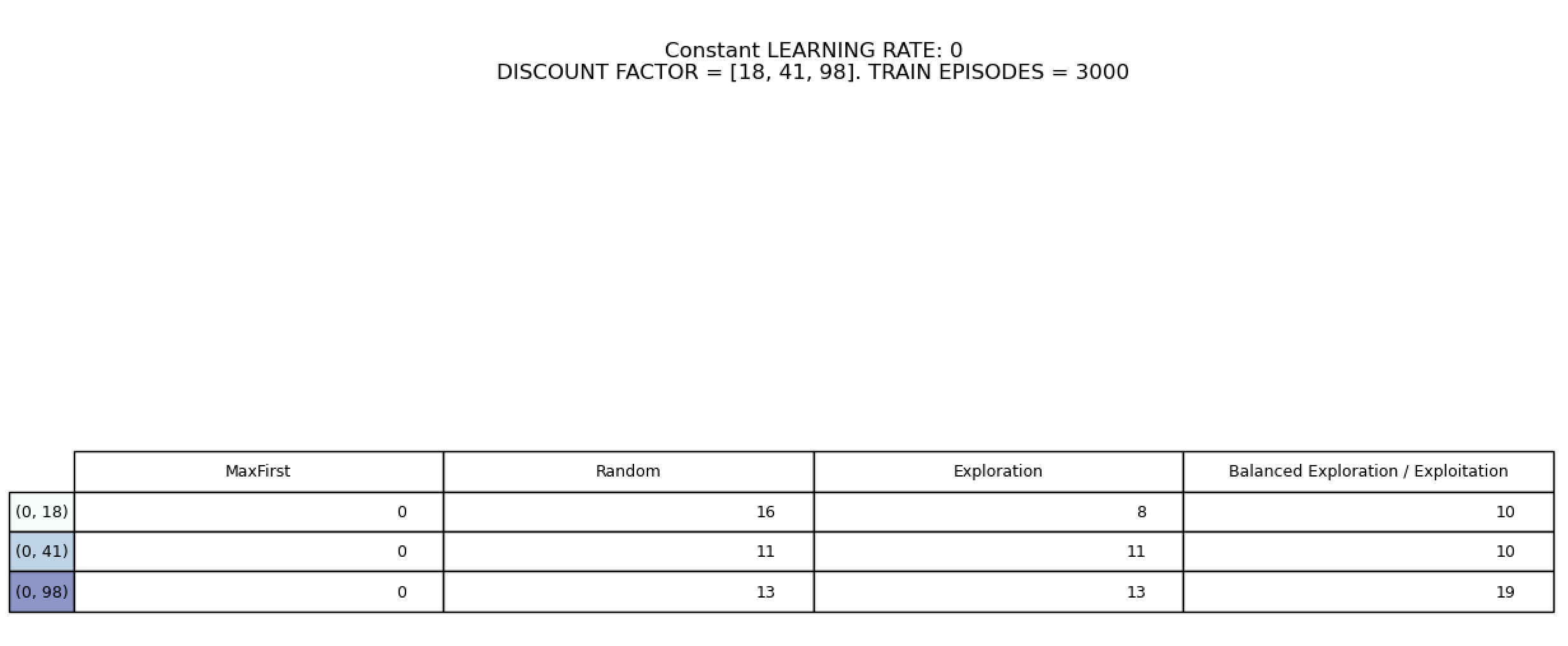
## Rata de invatare mica. Factor de discount mic – mediu – mare



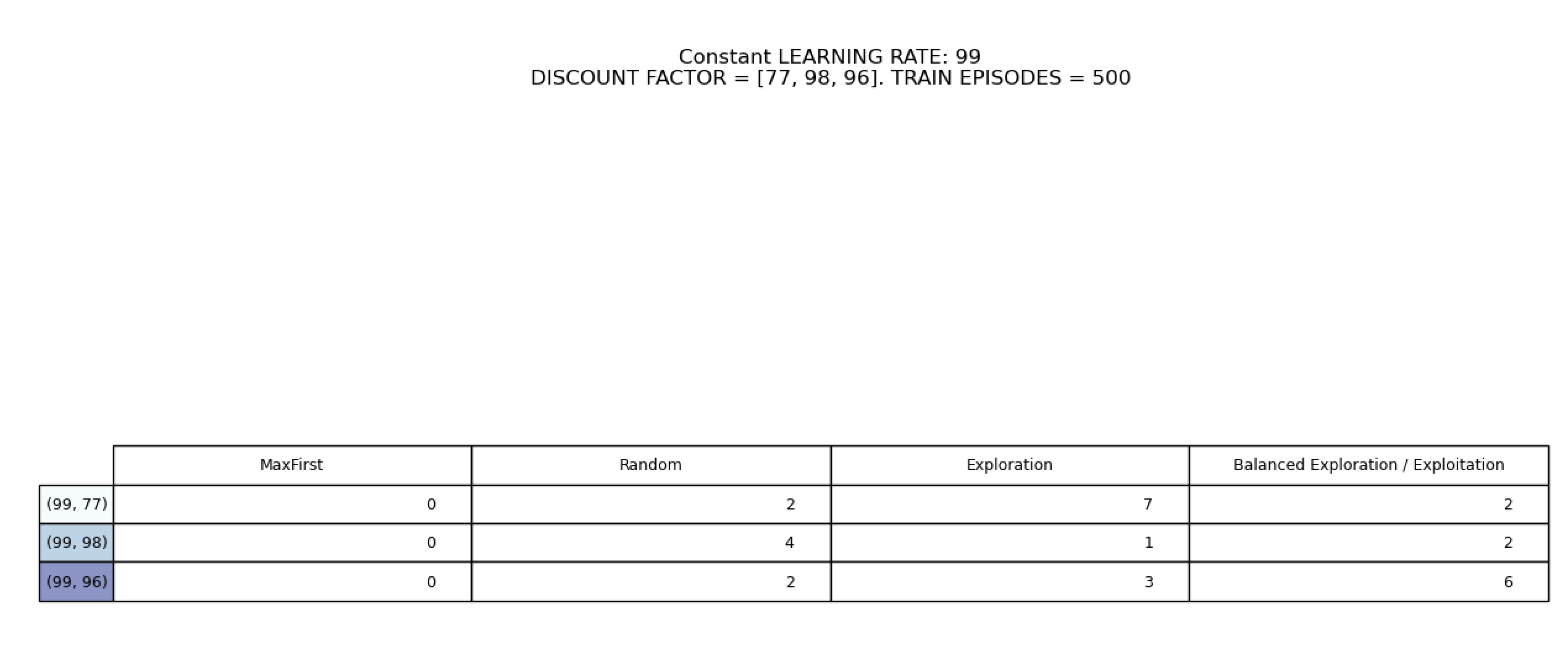
Strategia de MaxFirst nu este deloc avantajata de o rata de invatare mica deoarece informatia noua nu mai suprascrie informatia veche, lucru necesar pentru a deduce actiunea maxima – agentul nu invata mai nimic si tine cont doar de informatia acumulata pana atunci.

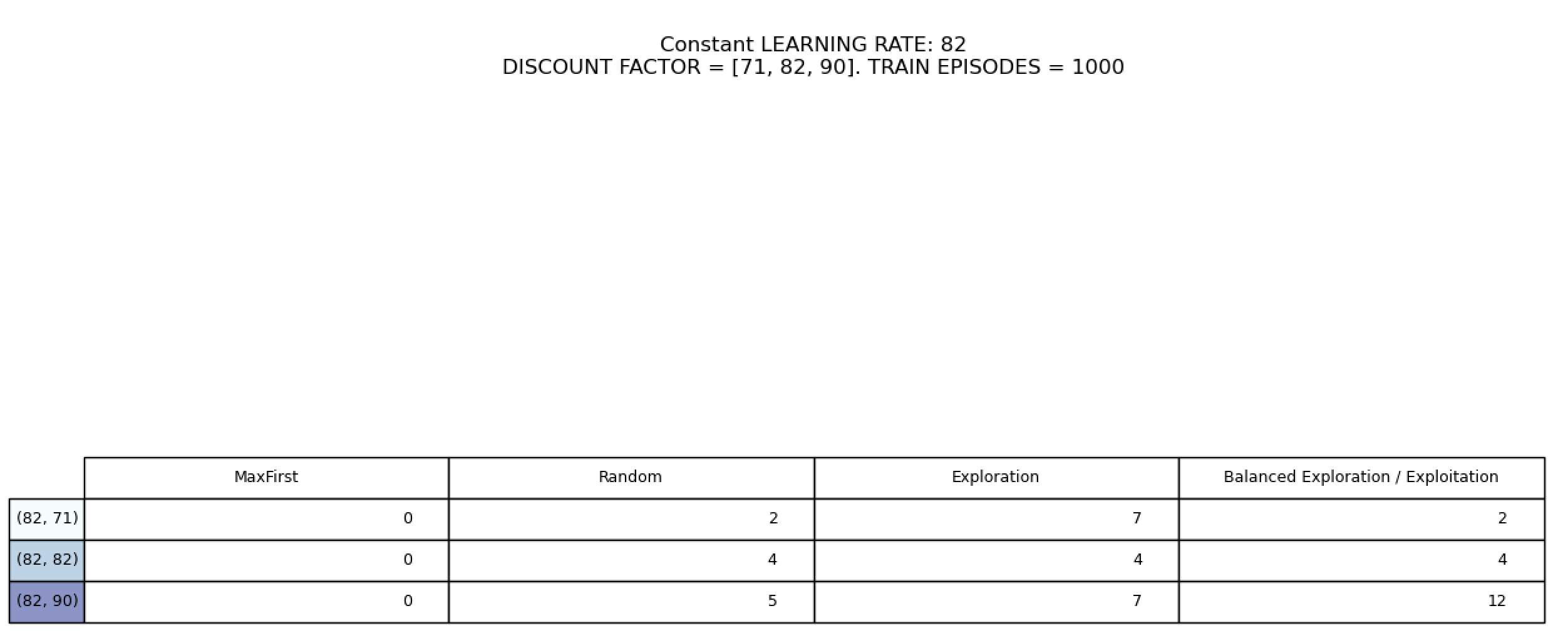
Nici celelalte strategii nu au rezultate exceptionale (maxim 10 jocuri castigate din 500), ceea ce inseamna ca indiferent de strategie, rata de invatare medie spre mare este foarte importanta pentru ca agentul sa invete.

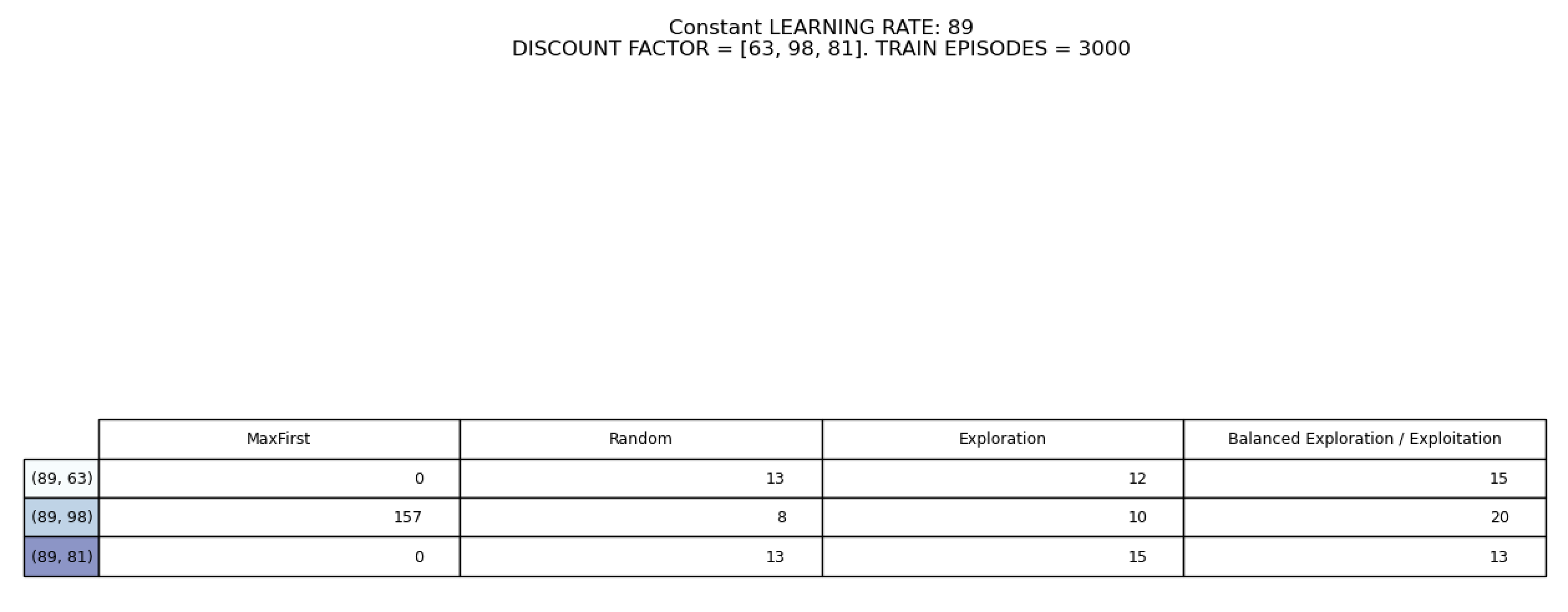




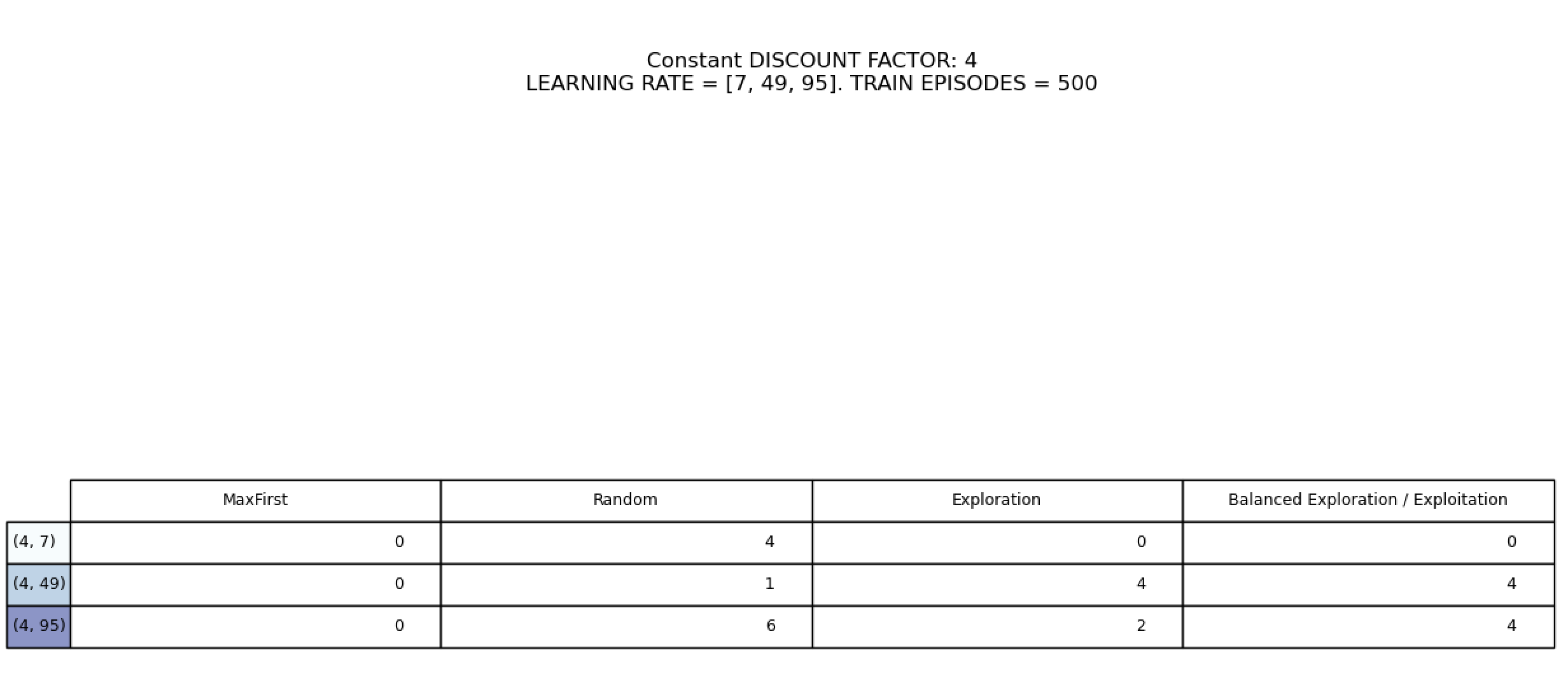
## Rata de invatare mare. Factor de discount mediu – mare

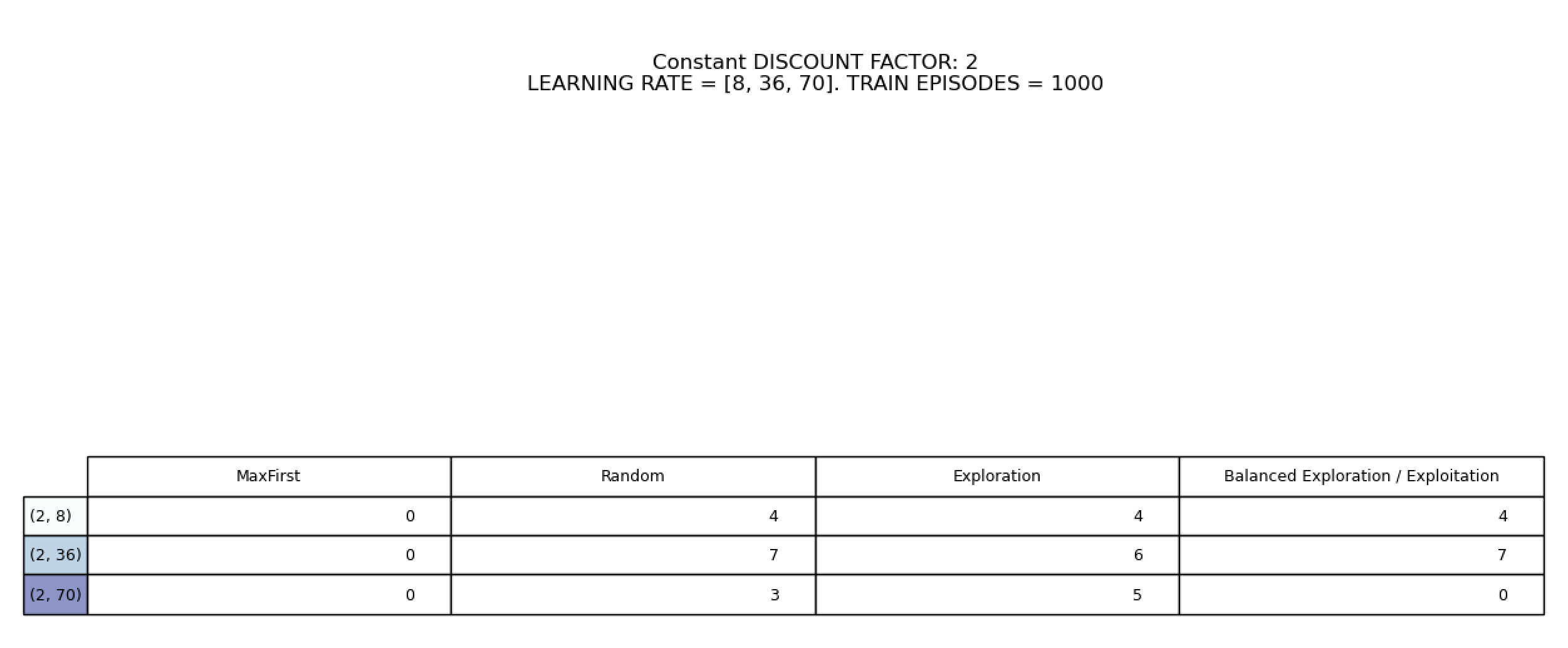


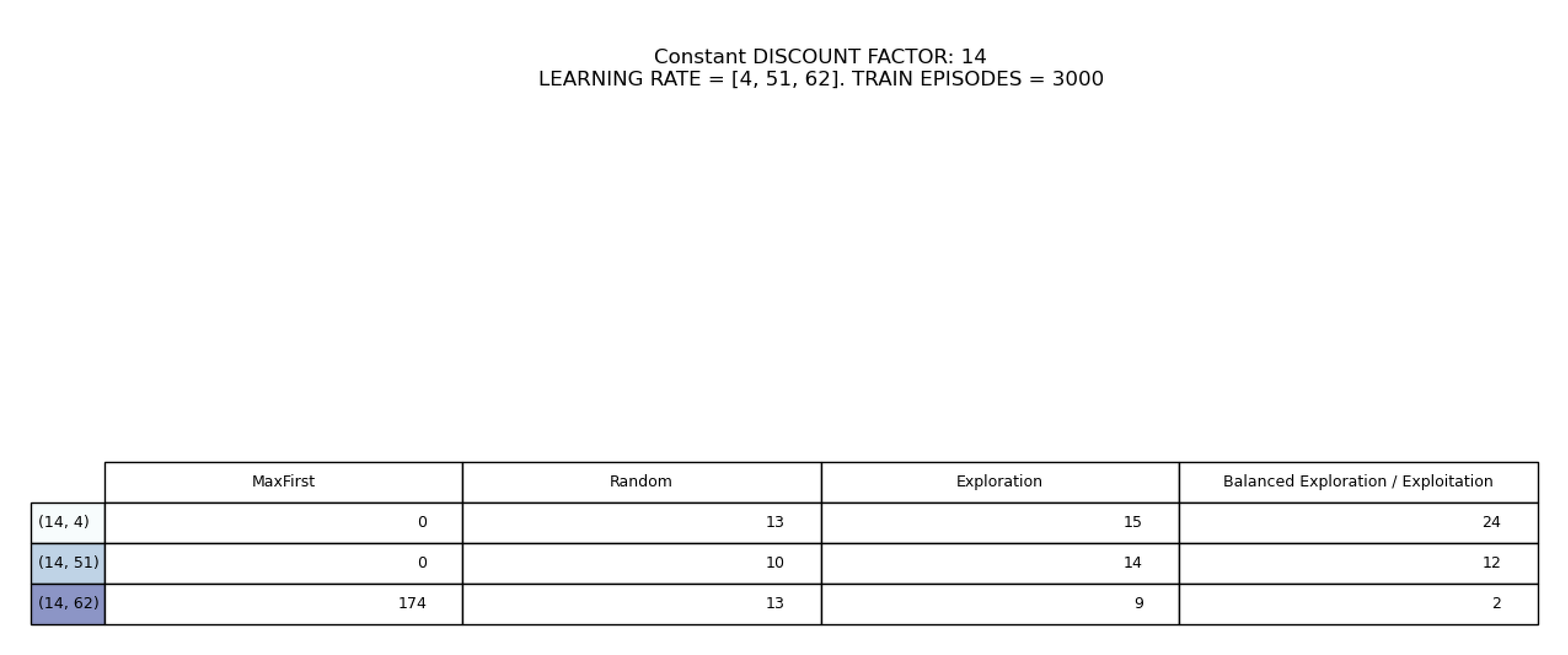




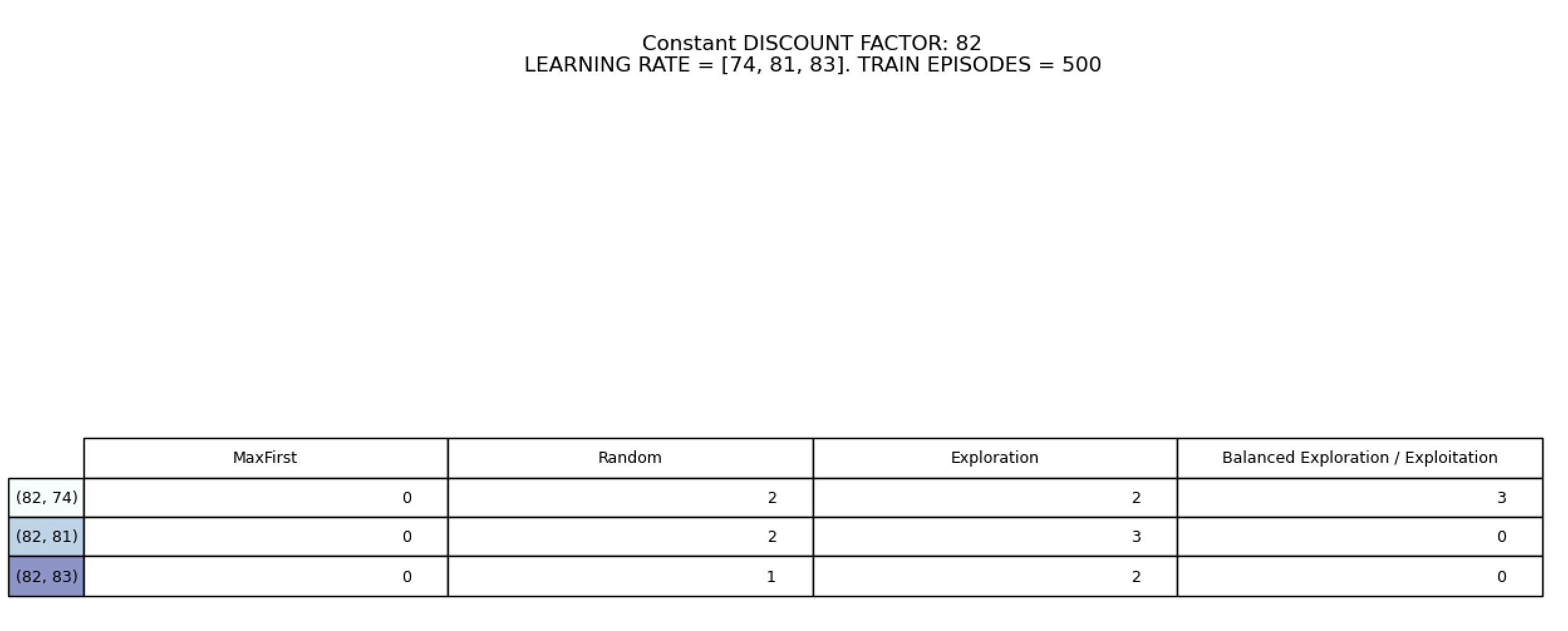
## Factor de discount mic. Rata de invatare mica - medie - mare

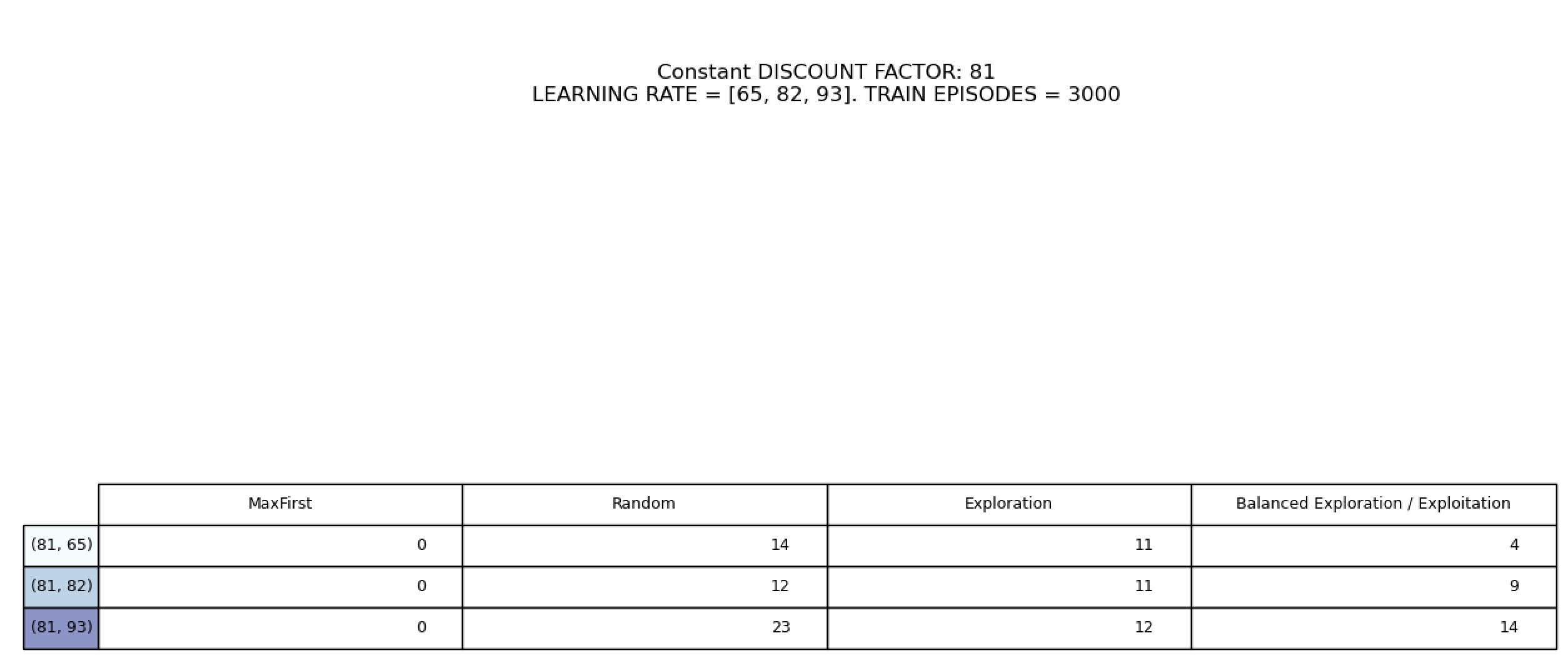






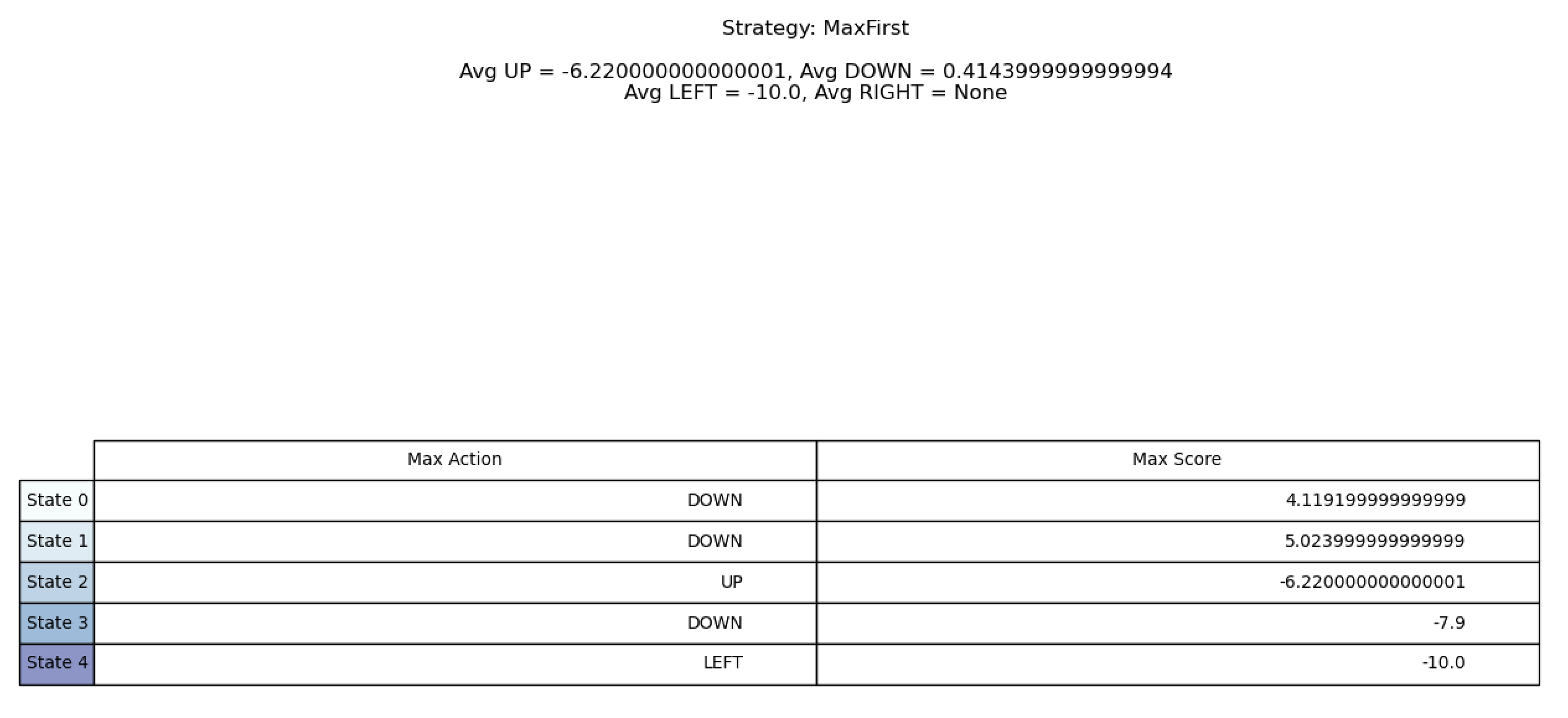
## Factor de discount mare. Rata de invatare medie - mare



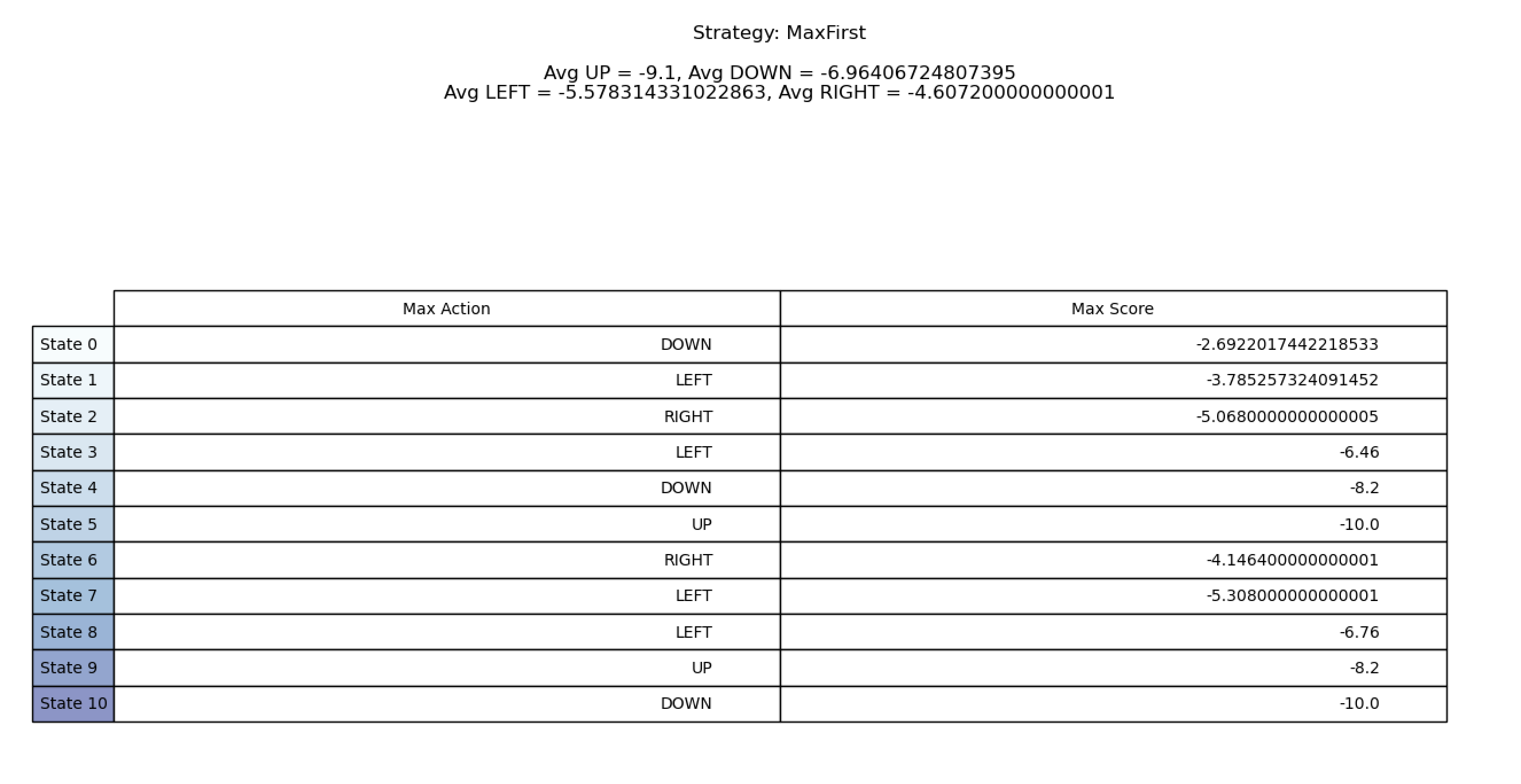


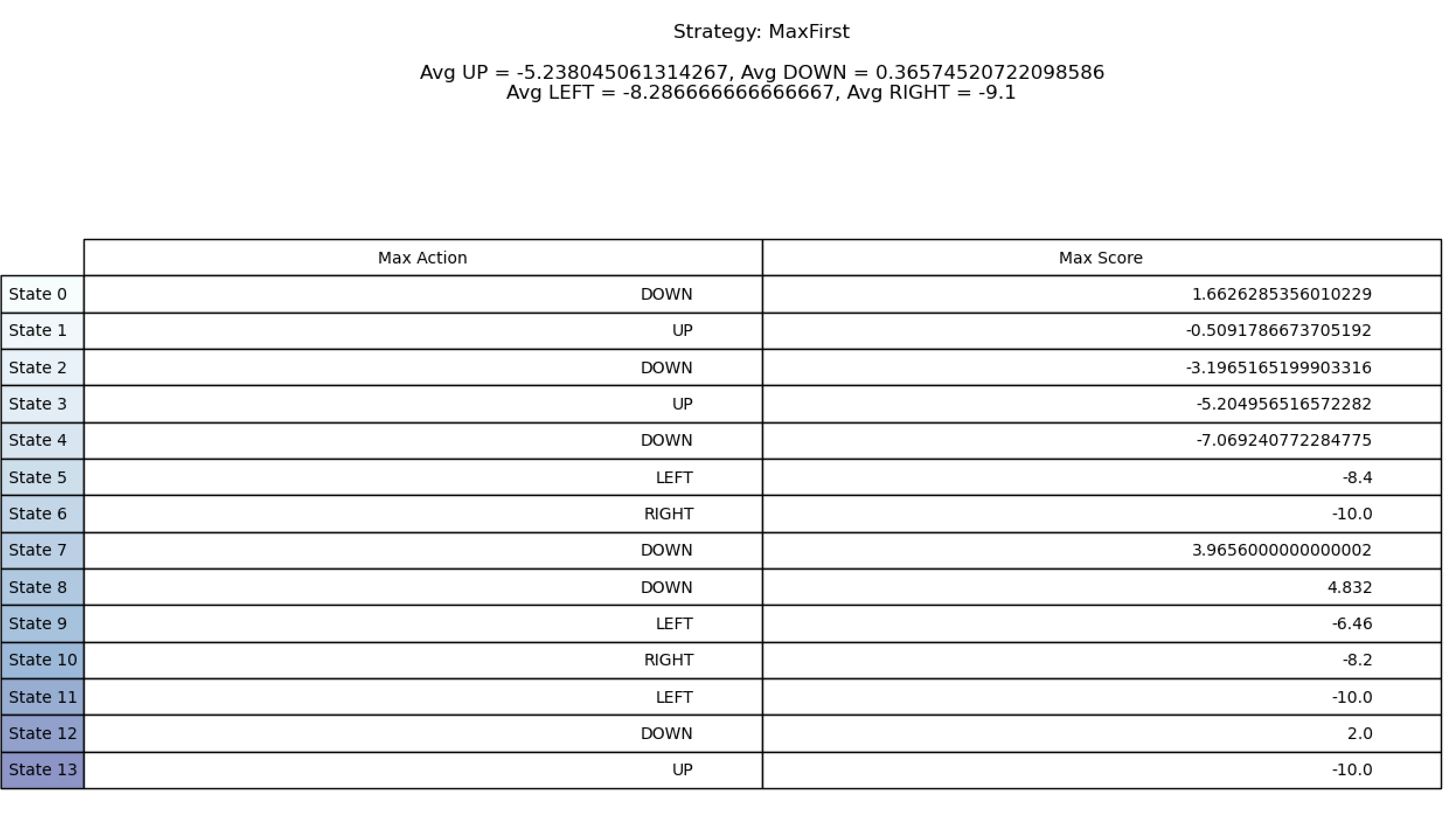
## Tabela de utilitati MaxFirst

Graficele cu tabela de utilitati afiseaza actiunea maxima si scorul acesteia pentru fiecare stare unica din Q. De asemenea, se afiseaza si media scorurilor fiecarei actiuni. Numarul maxim de stari afisate este 20.

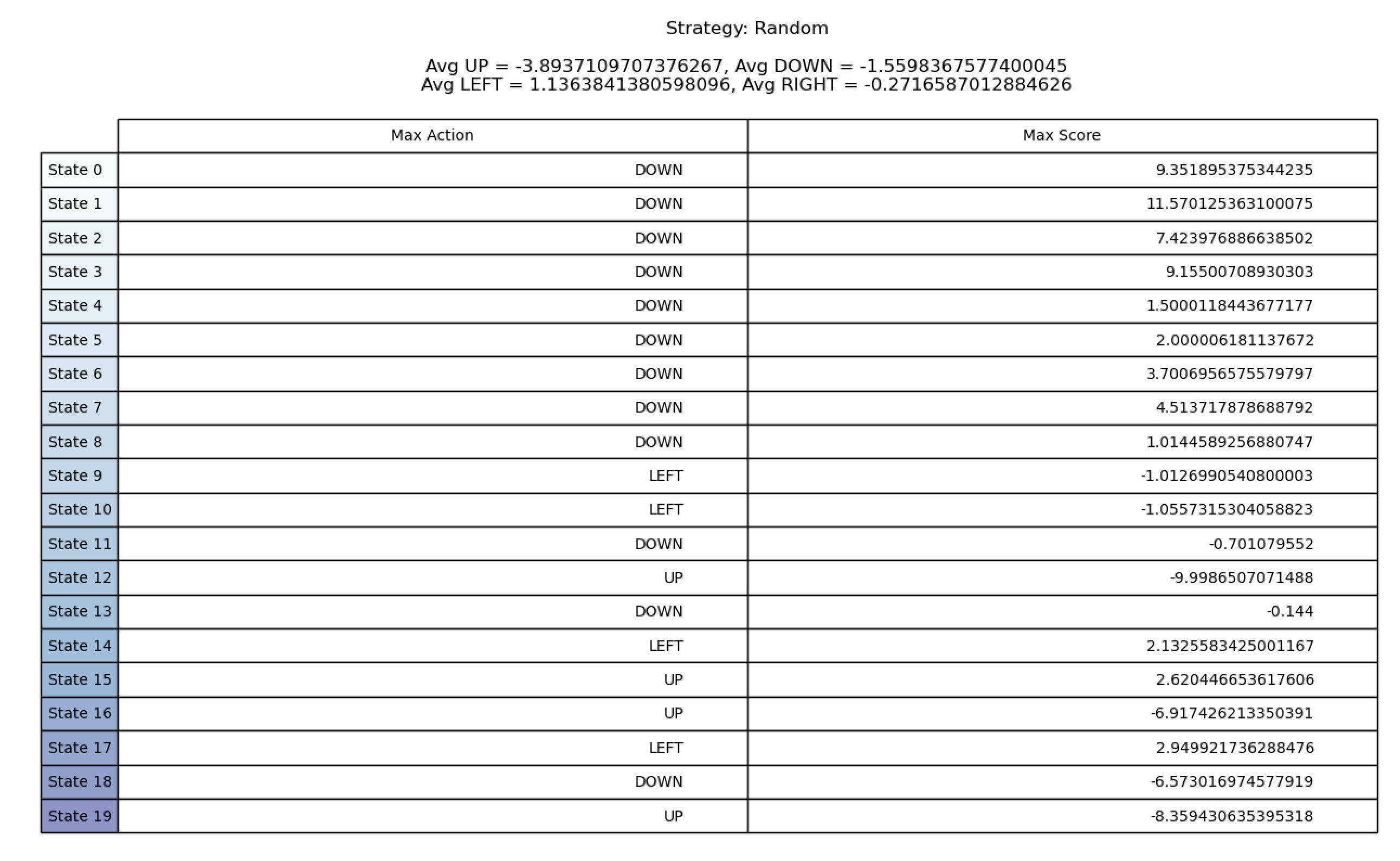


Asa cum se observa si din graficul cu scoruri, actiunea cea mai avantajoasa e cea de a merge in jos (spre prima bucata de branza), insa cand a incercat sa se aproprie si de a doua bucata, mergand la stanga, s-a apropriat si de pisica. De aceea, miscarea la stanga e cea mai dezavantajoasa.

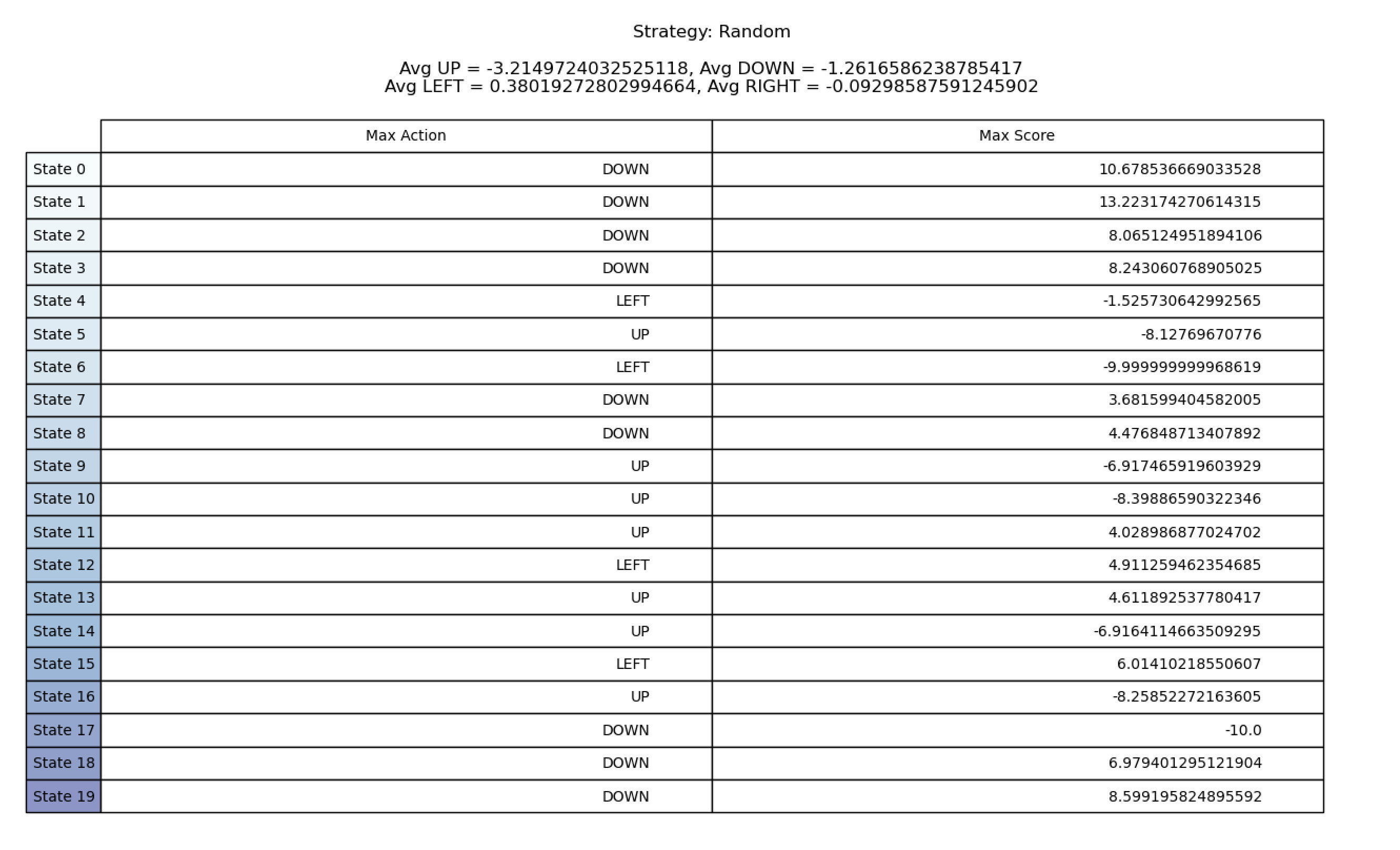


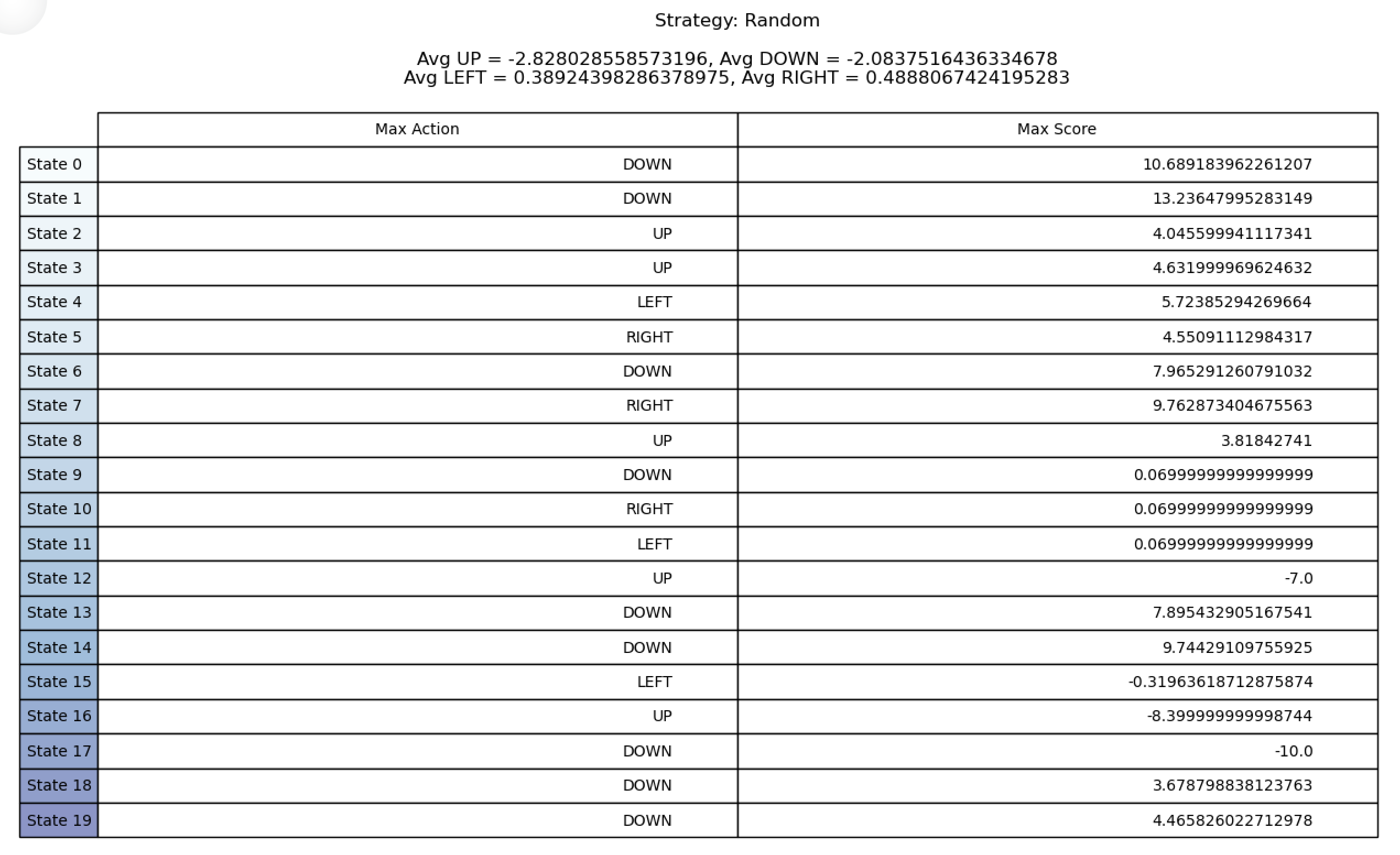


## MaxFirst vs Random



* Tabela de utilitati pentru strategia Random este mult mai mare decat cea pentru MaxFirst deoarece soarecele ia in calcul mai multe actiuni.
* Agentul este mai avantaj daca merge la stanga decat daca merge sus sau jos.
* Agentul incearca si actiunea de a merge la dreapta.





# Harta medie (8 x 8)

## 500, 1000 si 3000 episoade

### MaxFirst

### Exploration

### Random

### Balanced Exploration / Exploitation

## Rata de invatare mica. Factor de discount mic – mediu – mare

## Rata de invatare mare. Factor de discount mediu – mare

## Factor de discount mic. Rata de invatare mica - medie - mare

## Factor de discount mare. Rata de invatare medie - mare

## Tabela de utilitati MaxFirst

## MaxFirst vs Random

# Harta mare (10 x 20)

## 500, 1000 si 3000 episoade

### MaxFirst

### Exploration

### Random

### Balanced Exploration / Exploitation

## Rata de invatare mica. Factor de discount mic – mediu – mare

## Rata de invatare mare. Factor de discount mediu – mare

## Factor de discount mic. Rata de invatare mica - medie - mare

## Factor de discount mare. Rata de invatare medie - mare

## Tabela de utilitati MaxFirst

## MaxFirst vs Random

# Concluzii