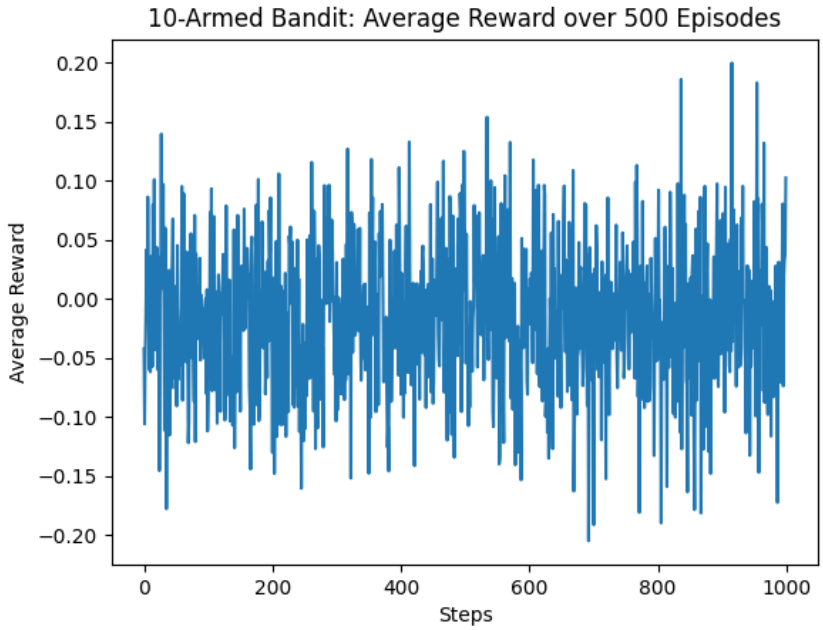
Borisov Timofei 580092

- Warum erhalten Sie das gezeigte Ergebnis für die zufällige Strategie vom Anfang?



In der ersten Aufgabe werden 500 Episoden mit 1000 Stichproben pro Episode durchgeführt. Natürlich geht es hier nicht um eine Auswahlstrategie. Es wird eine Zufallsauswahlmethode verwendet.

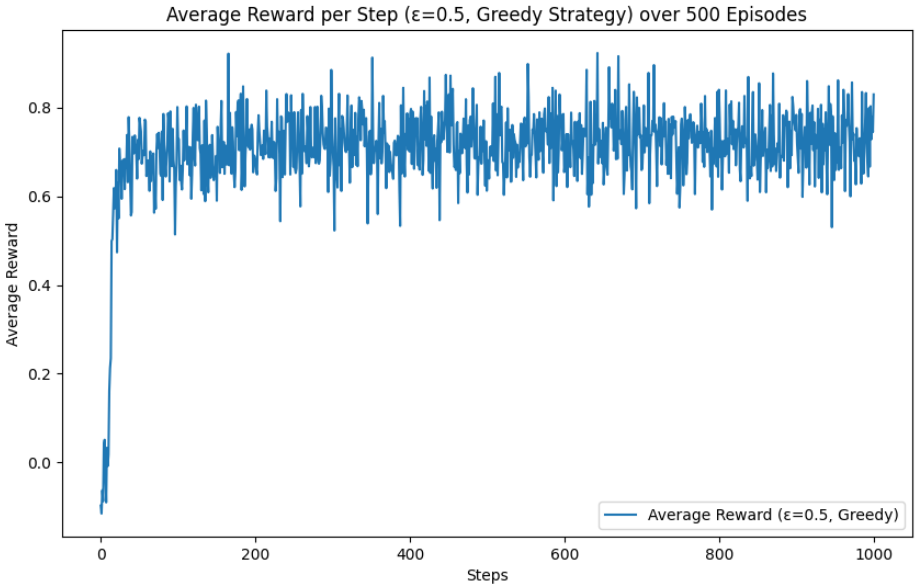
```
for episode in range(Ne):
    bandit_function = BanditFactory.get_normal_bandit_function(k)
    episode_rewards = np.zeros(N)

    for step in range(N):
        random_arm = np.random.randint(k)
        episode_rewards[step] = bandit_function.play_arm(random_arm)

    all_rewards += episode_rewards

average_rewards = all_rewards / Ne
```

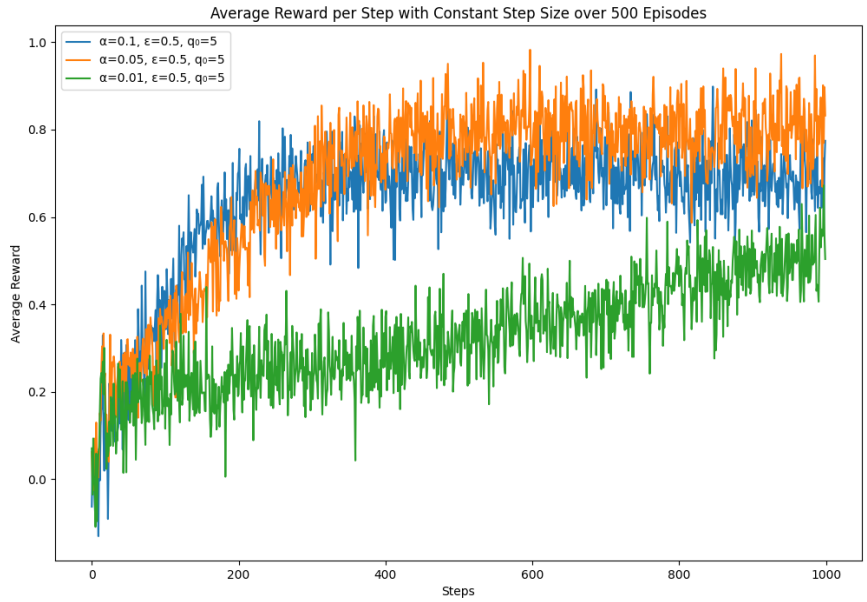
- Gehen Sie auf das Zusammenspiel zwischen Erkundung (exploration - neue Aktionen wählen) und Ausnutzen (exploitation - die beste verfügbare Aktion wählen) in Hinblick auf die folgenden Fragen ein:

- Mit welchen Methoden erhalten wir am schnellsten hohen rewards? - Welche Methoden liefert auf lange Sicht die besten rewards?

- Welche Methode scheint insgesamt die besten Resultate zu liefern und warum?

- Warum liefert die greedy Strategie mit Q0 = 5 besser Resultate als mit Q0 = 0 und was würden Sie erwarten, falls Q0 = −5 gewählt wird?





Der schnellste Weg zur höchsten Belohnung ist die Gier-Strategie. Im Falle einer konstanten Schrittweite ist die effizienteste Strategie eine große Schrittweite von 0,1 oder 0,5. Der schnellste Weg zur höchsten Belohnung ist die Gier-Strategie. Im Falle einer konstanten Teilung ist die effizienteste Strategie eine große Teilung von 0,1 oder 0,5. Interessant ist, dass ein nicht zu großer Schritt von 0,5 nicht auf das nicht-maximale Ergebnis fixiert ist, was die Möglichkeit schafft, ein höheres Ergebnis zu erzielen. Als stabilste und leistungsfähigste Strategie erweist sich die Strategie mit einer durchschnittlichen Schrittweite von 0,5, die auf lange Sicht bessere Ergebnisse erzielt.

Die Strategie mit Q0=0 neigt dazu, sich auf das durchschnittliche Ergebnis zu fixieren, während der Algorithmus mit Q0=5 aktiver nach einem optimalen Hebel sucht. Während Q0=-5 dem Algorithmus einen schlechten Start beschert, wenn er eine eher pessimistische Erwartung hat.

Im Fall alpha_small = [0.0000000001] gibt es eine starke Fixierung auf einen Wert nahe bei 0. Es gibt auch eine starke Schwankung des Ergebnisses, was auf ein instabiles Ergebnis hinweist.