Reinforcement Learning & Dynamic Optimization

Assignment 1

Bandit Algorithms Analysis

ε-greedy & UCB1

Πέτρου Δημήτριος

Διδάσκων: Σπυρόπουλος Θρασύβουλος

Χανιά, Μάρτιος 2023

Πολυτεχνείο Κρήτης
Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

1. Εισαγωγή

Η ανάλυση αφορά στη μελέτη απόδοσης των αλγορίθμων UCB1 και ε-greedy οι οποίοι εφαρμόζονται σε προβλήματα multi-armed bandits με σκοπό την επίτευξη του μέγιστου reward μέσω βελτίωσης των κριτηρίων επιλογής ανά το χρόνο. Οι δύο αλγόριθμοι σκοπεύουν στην επίλυση του προβλήματος exploration-exploitation.

2. Υλοποίηση

Η υλοποίηση έγινε σε γλώσσα Python. Το πρόγραμμα εκτελεί τους δύο αλγορίθμους πάνω σε k bandits για T γύρους και διατηρεί για κάθε έναν ορισμένα στατιστικά καθώς και το συνολικό reward και regret ανά γύρο.

1. **ε-greedy**

   Η επιλογή του καλύτερου (μεγαλύτερο reward) bandit ανά τους γύρους γίνεται σύμφωνα με την παράμετρο ϵ_t η οποία συγκρίνεται με μια τυχαία επιλέγμενη τιμή. Το απότελεσμα της σύγκρισης ορίζει τον τρόπο επιλογής του επόμενου bandit που θα γίνει pull. Το tuning του αλγορίθμου, ώστε να βελτιωθούν οι επιλογές που πραγματοποιεί σε επόμενους γύρους, επιτυγχάνεται με την μείωση της παραμέτρου ϵ_t μέσω της σχέσης:

   $$\epsilon_{\mathbf{t}} = \mathbf{t}^{-\frac{1}{3}} \cdot (\mathbf{klog(t)})^{\frac{1}{3}}$$

   καθώς σύμφωνα με την θεωρία ο αλγόριθμος ε-greedy επιτυγχάνει sublinear regret $O(t^{\frac{2}{3}} \cdot (klog(t))^{\frac{1}{3}})$ για παράμετρο $\epsilon_t = O(t^{-\frac{1}{3}} \cdot (klog(t))^{\frac{1}{3}})$ Η παράμετρος ϵ_t αρχικοποιείται σε μια σχετικά μεγάλη τιμή ώστε κατά τον πρώτο γύρο να επιλεγεί καταχρηστικά ένα τυχαιο bandit. Στην συνέχεια και πριν από κάθε επόμενο γύρο επαναπροσδιορίζεται η παράμετρος σύμφωνα με τις μεταβλητές.

2. **UCB1**

   Ο αλγόριθμος UCB1 διατηρεί μια εκτίμηση του αναμενόμενου reward ανά bandit και υπολογίζει ενα διάστημα confidence ώστε να ισορροπήσει το exploration και το exploitation. Σε κάθε γύρο T ο αλγόριθμος επιλέγει το bandit με τη μεγαλύτερη τιμή του UCB η οποία υπολογίζεται ως:

   $$\mathbf{X_i} + \sqrt{\frac{\mathbf{lnT}}{\#\mathbf{PullsPerBandit}}}$$

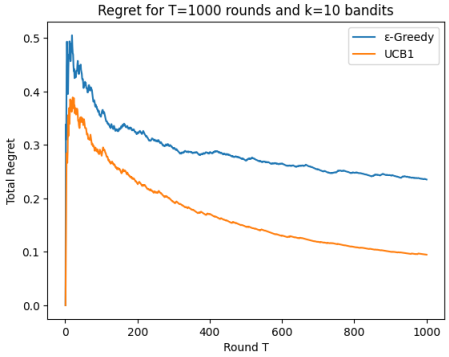
   όπου X_i είναι το cumulative reward για το bandit i και T ο συνολικός αριθμός των γύρων. Η ποσότητα της ρίζας αντιστοιχεί στο διάστημα confidence που εξασφαλίζει ότι τα λιγότερο εξερευνημένα bandits θα έχουν υψηλότερη προτεραιότητα.

3. Αποτελέσματα

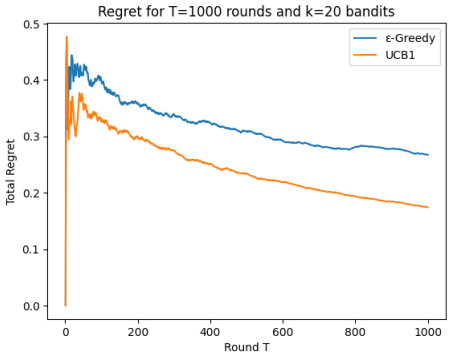
Η βασική παράμετρος σύμφωνα με την οποία αξιολογήθηκε η απόδοση κάθε αλγορίθμου είναι ο αριθμός του cumulative regret ανά τους γύρους. Η απεικόνιση του regret ανά τον χρόνο αποδεικνύει και την ταχύτητα μάθησης-βελτίωσης του αλγοριθμού αλλά και την αποτελεσματικότητα μεταξύ τους. Σημειώνεται πως οι δύο αλγόριθμοι δεν προσομοιώνονται ταυτόχρονα, πράγμα που σημαίνει ότι έρχονται αντιμέτωποι με μερικώς διαφορετικά rewards ανά bandit. Ωστόσο αυτό δεν επηρεάζει τον γενικό σκοπό της ανάλυσης καθώς οι διαφορές στα αποτελέσματα είναι αμελητεές και δεν επηρεάζουν ποιοτικά το διάγραμμα που προκύπτει.

Παρατίθενται αποτελέσματα εκτέλεσης των αλγορίθμων για:

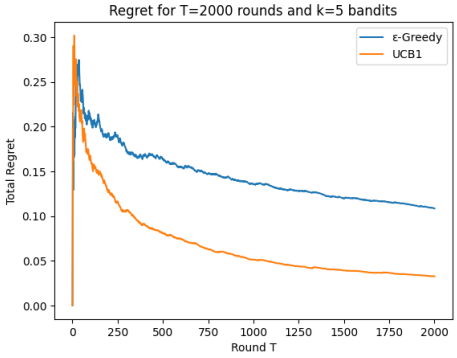
1. **k=10 & T=1000**



2. **k=20 & T=1000**



3. **k=5 & T=2000**



Οι δύο αλγόριθμοι πετυχαίνουν sublinear regret για κάθε σετ τιμών k,T. Όπως ήταν αναμενόμενο ο UCB1 αντιμετωπίζει καλύτερα το πρόβλημα από τον ε-greedy αφού σε όλες τις περιπτώσεις αφού φαίνεται πως είναι γρηγορότερος αλλά και καλύτερός στην επιτεύξη χαμηλού regret. Φαινομενικά σε αρχικά στάδια ο UCB1 φαίνεται να αποδίδει χειρότερα από τον e-greedy, ωστόσο αυτό εξαλείφεται με την πάροδο του χρόνου.

Στην περίπτωση 2 εκτελώντας τους αλγορίθμους για διπλάσιο αριθμό bandits k σε ίδιο χρόνο T παρατηρείται ότι αυτό δεν επηρεάζει την ήδη μέτρια απόδοση του ε-greedy αλλά καθυστερεί σημαντικά τον UCB1 ο οποίος καταλήγει σε σχεδόν διπλάσιο regret από ότι πριν.

Στην περίπτωση 3 για υποδιπλάσιο αριθμό bandits αλλά διπλάσιο χρόνο από τις αρχικές τιμές παρατηρείται ότι ο UCB1 σχεδόν εκμηδενίζει το regret σχετικά γρήγορα ενώ ο ε-greedy παρότι βελτιώνεται λόγω μικρότερου δείγματος παραμένει 2ος στην ιεραρχία τόσο σε ταχύτητα όσο και σε αποτελεσματικότητα.