Μια Εισαγωγή στην Ενισχυτική Μάθηση

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Μαθηματικών

Διπλωματική Εργασία

Καπετανάκης Βασίλειος

Μεταπτυχιακό Πρόγραμμα Σπουδών στα Μαθηματικά
Ειδίκευση Στατιστικής και Επιχειρησιακής Έρευνας

Επιβλέπων: Αντώνης Οικονόμου

Αθήνα
Ιούνιος 2021

Στην οικογένεια μου
και τον φίλο μου Παναγιώτη.

Περίληψη

Η Ενισχυτική Μάθηση αποτελεί μια από τις σπουδαιότερες και πιο ανερχόμενες κατηγορίες Μηχανικής Μάθησης, λόγω της μεγάλης ευελιξίας που διαθέτουν οι αλγόριθμοι της, στην διαχείριση μεγάλων χώρων καταστάσεων και άγνωστων πιθανοτήτων μετάβασης, σε προβλήματα που μοντελοποιούνται ως Μαρκοβιανές Διαδικασίες Αποφάσεων. Στόχος της παρούσας εργασίας είναι η παρουσίαση των βασικών αρχών την Ενισχυτικής Μάθησης, δίνοντας έμφαση τόσο στο απαραίτητο μαθηματικό πλαίσιο στο οποίο είναι δομημένη, όσο και σε αλγορίθμους, πολλοί εκ των οποίων υλοποιούνται στο λογισμικό R για την καλύτερη κατανόηση τους. Αν και οι πολύ τεχνικές μαθηματικές αποδείξεις, απουσιάζουν υπό το πρίσμα μιας εισαγωγής, έγινε προσπάθεια ένταξης εκείνων που κατά κύριο λόγο βασίζονται σε επιχειρήματα της Θεωρίας Πιθανοτήτων που συναντάει κανείς και σε προπτυχιακό επίπεδο.

Η παρούσα εργασία είναι δομημένη σε 3 Κεφάλαια. Στο Κεφάλαιο 1, γίνεται μια σύντομη επισκόπηση στα 3 βασικότερα είδη Μηχανικής Μάθησης που είναι: η Επιβλεπόμενη Μάθηση, η Μη-Επιβλεπόμενη Μάθηση, και η Ενισχυτική Μάθηση, με σκοπό ο αναγνώστης της εργασίας να μπορεί να διαχωρίζει ποιος είναι ο στόχος του κάθε είδους, και σε ποιες περιπτώσεις το καθένα από αυτά είναι καταλληλότερο. Στο Κεφάλαιο 2, γίνεται μια εισαγωγή σε μια απλουστευμένη υποκατηγορία προβλημάτων Ενισχυτικής Μάθησης γνωστή και ως Multi-Armed Bandits, βασικό χαρακτηριστικό της οποίας είναι ότι η δέσμη των δυνατών αποφάσεων σε κάθε βήμα παραμένει σταθερή. Επίσης στο τέλος του Κεφαλαίου, γίνεται εφαρμογή των αλγορίθμων που διατυπώνονται στο λογισμικό R, με σκοπό την πειραματική επαλήθευση των θεωρητικών τους ιδιοτήτων.

Τέλος, το Κεφάλαιο 3 είναι αφιερωμένο στο γενικότερο πλαίσιο της Ενισχυτικής Μάθησης όπου κάθε κατάσταση χαρακτηρίζεται από το δικό της σύνολο αποφάσεων. Αφού διατυπωθούν με σαφήνεια κρίσιμες έννοιες όπως οι Εξισώσεις του Bellman, οι βέλτιστες συναρτήσεις αξίας, και οι οι βέλτιστες πολιτικές, θα προχωρήσουμε στη διατύπωση κάποιων από τους σημαντικότερους Αλγορίθμους Ενισχυτικής Μάθησης που προσεγγίζουν βέλτιστες πολιτικές, τόσο στην περίπτωση που οι πιθανότητες μετάβασης είναι γνωστές, όσο και στην περίπτωση που είναι άγνωστες.

Abstract

Reinforcement Learning is one of the most important and up-and-coming categories of Machine Learning, due to the great flexibility of its algorithms, in managing large state spaces and unknown transition probabilities, to problems modeled as Markov Decision Processes. The aim of this postgraduate Thesis is to present the basic principles of Reinforcement Learning, emphasizing both the necessary mathematical framework in which it is structured, and algorithms, many of which are implemented in R software for better understanding. Although the highly technical mathematical proofs are absent in the light of an introduction, an attempt has been made to include those which are mainly based on arguments of Probability Theory, that one can encounter in an undergraduate level of studies.

This Thesis is structured in 3 Chapters. Chapter 1, is a brief overview of the 3 main types of Machine Learning: Supervised Learning, Unsupervised Learning, and Reinforcement Learning, so that the reader can distinguish what the goal of each type is, and in which cases each of them is more appropriate. In Chapter 2, an introduction is made to a simplified subset of the Reinforcement Learning Problem also known as the Multi-Armed Bandit Problem, a key feature of which is that the set of possible actions at each time step remains unchanged. In addition, at the end of the Chapter, the algorithms formulated are implemented in the R software, in order to experimentally verify their theoretical properties.

Lastly, Chapter 3 is devoted to the general context of Reinforcement Learning where each state is characterized by its own set of actions. Having clearly articulated critical concepts such as the Bellman Equations, optimal value functions, and optimal policies, we will proceed to formulate some of the most important Reinforcement Learning Algorithms that approach optimal policies, both in the case of known and unknown transition probabilities.

Ευχαριστίες

Θα ήθελα αρχικά να εκφράσω τις θερμότερες ευχαριστίες μου στον επιβλέποντα Καθηγητή μου, κύριο Αντώνη Οικονόμου, του οποίου η καθοδήγηση και η συνεχής παρότρυνση, συνέβαλαν τα μέγιστα στη διαμόρφωση της παρούσας Διπλωματικής εργασίας. Η πόρτα του γραφείου του ήταν πάντα ανοιχτή καθ΄ όλη την διάρκεια των σπουδών μου για να συζητήσουμε και να με στηρίξει σε οποιοδήποτε θέμα με απασχολούσε.

Θα ήθελα επίσης να εκφράσω την ευγνωμοσύνη μου στους Καθηγητές Απόστολο Μπουρνέτα και Σάμη Τρέβεζα για την συμμετοχή τους στην κριτική επιτροπή. Η συμβολή τους καθ΄ όλη την διάρκεια των σπουδών μου ήταν ιδιαιτέρως σημαντική καθώς με τα μαθήματα τους «Προσομοίωση» και «Απαραμετρική Στατιστική» αντίστοιχα, έλαβα ερεθίσματα τα οποία αποτέλεσαν έμπνευση για τη συγγραφή της παρούσας εργασίας.

Ένα μεγάλο ευχαριστώ, στα μέλη της μεταπτυχιακής επιτροπής επιλογής φοιτητών, Καθηγητές Λουκία Μελιγκοτσίδου, Κωστή Μηλολιδάκη και Σάμη Τρέβεζα που μου επέτρεψαν να συμμετάσχω στο Μεταπτυχιακό Πρόγραμμα. Στον συνάδελφο και φίλο μου Νίκο Σταμάτη, με τον οποίο περάσαμε αρκετές ώρες να συζητάμε και να ανταλλάσσουμε ιδέες σχετικά με το περιεχόμενο της εργασίας ακόμα και πριν την έναρξη της συγγραφής της. Στον επίσης συνάδεφλο και φίλο μου, Περικλή Βασιλείου από το Πρόγραμμα των Θεωρητικών Μαθηματικών, ο οποίος διατηρούσε από την πρώτη στιγμή αμείωτο το ενδιαφέρον του να τη διαβάσει και να τη συζητήσουμε σε όλα τα στάδια εξέλιξης της.

Τέλος, δεν θα μπορούσαν να λείπουν από τις ευχαριστίες μου, όλα τα μέλη της οικογενείας μου και οι φίλοι μου. Χωρίς την ανιδιοτελή τους στήριξη, υπομονή και κατανόηση σε όλα τα στάδια των σπουδών μου, δεν θα τα είχα καταφέρει.

Περιεχόμενα

Κεφάλαιο 1

Γνωριμία με τη Μηχανική Μάθηση και τα είδη της

Η Μηχανική Μάθηση (Machine Learning) έχει κάνει αισθητή την παρουσία της τα τελευταία χρόνια, τόσο στην εξέλιξη της τεχνολογίας, όσο και στις ζωές μας γενικότερα. Αναγνώριση εικόνας και ήχου, αυτόματη μετακίνηση ηλεκτρονικών μέσων, εντοπισμός κακόβουλου λογισμικού και διαδικτυακής απάτης αποτελούν απλά ένα μικρό δείγμα από το συνεχώς αναπτυσσόμενο πεδίο εφαρμογών της.

Η κεντρική ιδέα της Μηχανικής Μάθησης αφορά τη μελέτη και κατασκευή αλγορίθμων που μαθαίνουν από τα δεδομένα για την επίτευξη ενός στόχου. Ο στόχος συχνά συμπίπτει με τη λήψη μιας απόφασης ή την πρόβλεψη μιας μελλοντικής κατάστασης.

Η ιδέα ότι ένας υπολογιστής μπορεί να κάνει υπολογισμούς μαθαίνοντας και όχι μόνο με τον ρητό προγραμματισμό του, είναι τουλάχιστον τόσο παλιά όσο και η ανάπτυξη των πρώτων ηλεκτρονικών υπολογιστών. Παρά το γεγονός ότι υπήρξαν φιλότιμες προσπάθειες από διάφορους ερευνητές για την υλοποίηση αυτής της ιδέας[1] δεν υπήρξε ουσιαστική πρόοδος μέχρι και τις αρχές της δεκαετίας του 80′ με ένα εγχειρίδιο ορόσημο για τον κλάδο, το Machine Learning: The AI Approach[2] (1983). Το συγκεκριμένο εγχειρίδιο περιείχε μεγάλο όγκο ερευνητικών εργασιών με αντικείμενο τη Μηχανική Μάθηση και έμελλε να πυροδοτήσει το παγκόσμιο ενδιαφέρον της επιστημονικής κοινότητας θέτοντας τα θεμέλια της ανερχόμενης εξέλιξης της μέχρι και τη μορφή που έχει σήμερα.

Η Μηχανική Μάθηση έχει πολλές επιρροές και συσχετίσεις με διάφορους επιστημονικούς τομείς. Από την περιοχή της Επιστήμης των Υπολογιστών έχει στενούς δεσμούς με την Τεχνητή Νοημοσύνη, ενδεικτικά, ενώ από τον κλάδο των Μαθηματικών αξίζει να αναφερθούν οι ισχυρές συνδέσεις της με τη Στατιστική και τον Μαθηματικό Προ-

[1] Πολύ σημαντική θεωρείται η εφεύρεση του Perceptron, ενός είδους τεχνητού νευρωνικού δικτύου, από τον Frank Rosenblatt, (1957).

[2] Συντάχθηκε από τους R. Michalski, J. Carbonell, and T. Mitchell.

γραμματισμό. Από τη μια μεριά η Στατιστική παρέχει όλο το απαραίτητο θεωρητικό και αλγοριθμικό υπόβαθρο για την αποτελεσματικότητα μιας πρόβλεψης και από την άλλη ο Μαθηματικός Προγραμματισμός παρέχει τη θεωρία και τις μεθόδους για τη βέλτιστη λήψη μιας απόφασης. Η κατάλληλη εφαρμογή της θεωρίας από όλους τους εμπλεκόμενους τομείς, είναι αυτή που τελικά θα δώσει την λύση σε ένα απαιτητικό πρόβλημα Μηχανικής Μάθησης.

Στις ενότητες που ακολουθούν θα γίνει μια πολύ συνοπτική παρουσίαση των 3 πολύ βασικών τύπων μηχανικής μάθησης:

- Επιβλεπόμενη Μάθηση (Supervised Learning)
- Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)
- Ενισχυτική Μάθηση (Reinforcement Learning))

Για την Ενισχυτική Μάθηση, που είναι και το αντικείμενο μελέτης της Εργασίας, θα αναπτυχθούν ορισμένες εισαγωγικές έννοιες στο τέλος του κεφαλαίου ως βάση για τη μετάβαση του αναγνώστη στα επόμενα κεφάλαια.

1.1 Επιβλεπόμενη Μάθηση

Η Επιβλεπόμενη Μάθηση (Supervised Learning) είναι μια κατηγορία Μηχανικής Μάθησης της οποίας σκοπός είναι ο χαρακτηρισμός δεδομένων και η ένταξη τους σε κάποιες κατηγορίες που ονομάζονται **κλάσεις (classes)**. Για την επίτευξη αυτού του στόχου είναι απαραίτητη η ύπαρξη ενός συνόλου διαθέσιμων παραδειγμάτων, καθένα από τα οποία θα έχει ένα **διάνυσμα χαρακτηριστικών (attributes vector)** και μια **ετικέτα (label)** η οποία θα υποδηλώνει την κατηγορία στην οποία ανήκει το παράδειγμα. Στην γενική περίπτωση, το σύνολο των διαθέσιμων παραδειγμάτων χωρίζεται σε 2 υποσύνολα: **Το σύνολο εκπαίδευσης (training set)** και το **σύνολο δοκιμής (testing set)**. Στο σύνολο εκπαίδευσης γίνεται αναζήτηση ενός μηχανισμού-κανόνα ο οποίος στόχο έχει να ταξινομεί μελλοντικά δεδομένα στα οποία είναι άγνωστη η κλάση στην οποία ανήκουν. Στο σύνολο δοκιμής γίνεται έλεγχος κατά πόσο ο προηγούμενος κανόνας ή **ταξινομητής (classifier)** που δημιουργήθηκε από το σύνολο εκπαίδευσης είναι αποτελεσματικός, αν δηλαδή έχει την τάση να ταξινομεί σωστά τα παραδείγματα δοκιμής, το οποίο μπορούμε να το συμπεράνουμε εφόσον γνωρίζουμε ήδη την κλάση στην οποία ανήκουν. Φυσικά, η τελική αποτελεσματικότητα του ταξινομητή θα αξιολογηθεί και από μελλοντικά παραδείγματα. Είναι δυνατόν για τα παραδείγματα δοκιμής ο ταξινομητής να έχει μεγάλο ποσοστό επιτυχίας (λόγου χάρη επειδή είναι λίγα σε πλήθος) αλλά αυτό να αρχίσει να μειώνεται σημαντικά με την εμφάνιση νέων παραδειγμάτων. Όλη η διαδικασία είναι δυναμική και στην πράξη χρειάζεται η εκ νέου ανανέωση του ταξινομητή ώστε να προσαρμοστεί καλύτερα στα νέα παραδείγματα και να γίνει αποδοτικότερος.

Στις δύο υποενότητες που ακολουθούν θα γίνει μια σύντομη επισκόπηση για 2 από τα βασικότερα είδη ταξινομητών, τους **γραμμικούς ταξινομητές (Linear Classifiers)** και τους **ταξινομητές k-πλησιέστερων γειτόνων (k- nearest neighbors classifiers)**

1.1.1 Γραμμικοί ταξινομητές

Αν κάνουμε γεωμετρική αναπαράσταση των στοιχείων ενός συνόλου εκπαίδευσης με n χαρακτηριστικά, θα παρατηρήσουμε ότι αυτά αποτελούν σημεία στον $\mathbb{R}^n$. Σε πολλές περιπτώσεις η θέση τους στο χώρο $\mathbb{R}^n$ θα «μαρτυρήσει» και σε ποιά κλάση ανήκουν. Ας υποθέσουμε για απλότητα ότι κάθε στοιχείο μπορεί να ταξινομηθεί σε μια από 2 διαθέσιμες κλάσεις, τη «θετική» (+) και την «αρνητική» (-), ενώ κάθε χαρακτηριστικό αποτελεί μια πραγματική μεταβλητή.

Ορισμός 1.1 Γραμμικός ταξινομητής στον $\mathbb{R}^n$ καλείται ένα **υπερεπίπεδο** της μορφής

$$w^T x = 0$$

όπου

$$w = (w_0, w_1, ..., w_n)^T \text{ και } x = (1, x_1, x_2, ..., x_n)^T.$$

Στόχος του γραμμικού ταξινομητή είναι η διαμέριση του $\mathbb{R}^n$ σε 2 υποσύνολα: θετικών (+) και αρνητικών (-) παραδειγμάτων. Έτσι ένα νέο δεδομένο σημείο x_0 που δεν έχει ετικέτα κλάσης θα χαρακτηρίζεται ως θετικό (+) αν $w^T x_0 > 0$ και αρνητικό (-) αν $w^T x_0 < 0$. Αν υπάρχει $w \in \mathbb{R}^{n+1}$ με την ιδιότητα $w^T x > 0$ για κάθε $x \in (+)$ και $w^T x < 0$ για κάθε $x \in (-)$ τότε οι δυο κλάσεις ονομάζονται **γραμμικά διαχωρίσιμες**.

Έτσι, η παραπάνω διαμέριση δεν είναι εφικτή για όλα τα δυνατά παραδείγματα εκπαίδευσης, και αυτό αποτελεί και μια από τις αδυναμίες του συγκεκριμένου ταξινομητή. Συνεπώς η υπόθεση ότι οι 2 κλάσεις είναι γραμμικά διαχωρίσιμες για τα δεδομένα παραδείγματα εκπαίδευσης είναι απαραίτητη για την ανάλυση των γραμμικών ταξινομητών.

Ο έλεγχος για την γραμμική διαχωρισιμότητα 2 κλάσεων μπορεί να επιτευχθεί με διάφορους τρόπους, τόσο θεωρητικούς, όσο και αλγοριθμικούς. Χωρίς να υπεισέλθουμε σε περαιτέρω λεπτομέρειες οι οποίες θα ξέφευγαν από τους σκοπούς της παρούσας Εργασίας, αξίζει να προχωρήσουμε στη διατύπωση ενός κριτηρίου που προέρχεται από την **Κυρτή Ανάλυση**.

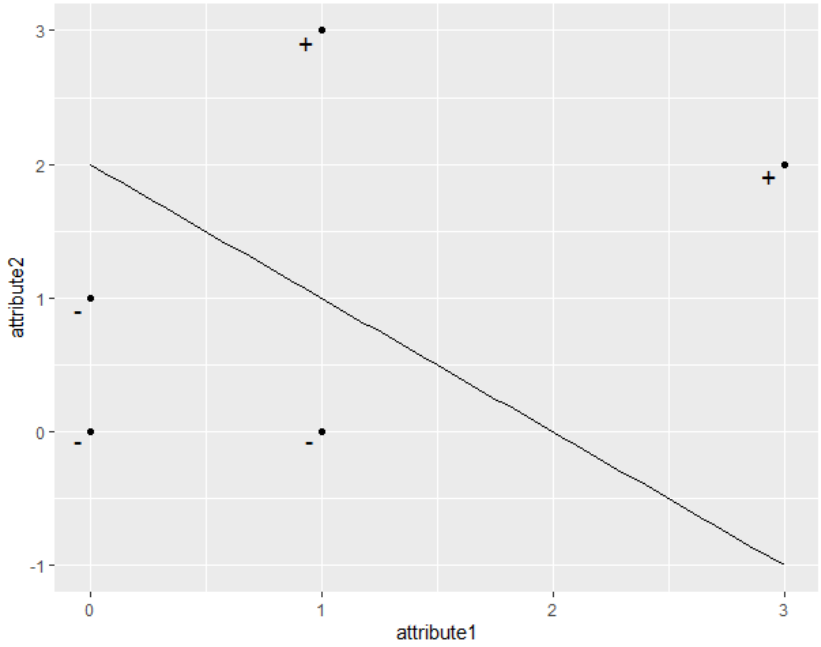
Κριτήριο Γραμμικής Διαχωρισιμότητας

Δυο κλάσεις σημείων στον $\mathbb{R}^n$, C_1 και C_2 είναι γραμμικά διαχωρίσιμες αν οι κυρτές τους θήκες $conv(C_1)$ και $conv(C_2)$ είναι **ξένες**, αν δηλαδή ισχύει:

$$conv(C_1) \cap conv(C_2) = \varnothing \tag{1.1}$$

Παράδειγμα 1.1 Ας υποθέσουμε ότι έχουμε στη διάθεση μας τα ακόλουθα 5 παραδείγματα εκπαίδευσης καθένα από τα οποία αποτελεί ένα διάνυσμα 2 χαρακτηριστικών:

Παραδείγματα εκπαίδευσης	Κλάση
$(0, 0)$	$-$
$(1, 0)$	$-$
$(0, 1)$	$-$
$(1, 3)$	$+$
$(3, 2)$	$+$



Σχήμα 1.1: Η ευθεία $y + x - 2 = 0$ ως γραμμικός ταξινομητής.

Σε αυτό το σημείο πρέπει να έχει γίνει αντιληπτός ο επακριβής ρόλος του ταξινομητή. Ένα μελλοντικό δεδομένο (x_0, y_0) θα χαρακτηρίζεται ως θετικό $(+)$ αν και μόνο αν $y_0 + x_0 - 2 > 0$, ενώ θα χαρακτηρίζεται ως αρνητικό αν και μόνο αν $y_0 + x_0 - 2 < 0$. Δεν πρέπει βέβαια να παραλείψουμε και την περίπτωση όπου για ένα μελλοντικό δεδομένο (x_0, y_0) ισχύει $y_0 + x_0 - 2 = 0$ (δηλαδή αν το δεδομένο βρίσκεται ακριβώς πάνω στον ταξινομητή). Σε αυτή την περίπτωση το δεδομένο συνηθίζεται είτε να ταξινομείται σε κάποια από τις κλάσεις με ένα τυχαίο τρόπο, είτε δίνεται προτίμηση στην κλάση με τα περισσότερα παραδείγματα. Από εδώ και στο εξής, θα ενσωματώνουμε την περίπτωση της ισότητας στην αρνητική κλάση. ■

Ο Αλγόριθμος Perceptron

Ας δούμε τώρα πώς μπορούμε να οδηγηθούμε στον πρώτο αλγόριθμο μηχανικής μάθησης της Εργασίας για τον εντοπισμό ενός Γραμμικού Ταξινομητή. Με $c(x)$ θα συμβολίζουμε την κλάση ενός παραδείγματος $x \in \mathbb{R}^n$ και πιο συγκεκριμένα αν $c(x) = 1$ θα θεωρούμε ότι ανήκει στη θετική κλάση, ενώ αν $c(x) = 0$ ότι ανήκει στην αρνητική κλάση. Με $h(x)$ θα συμβολίζουμε την κλάση που ο ταξινομητής **υποθέτει** ότι ανήκει το παράδειγμα x. Υπενθυμίζουμε ότι ο ταξινομητής υποθέτει ότι το παράδειγμα είναι θετικό αν $w^T x > 0$ και αρνητικό αν $w^T x \leq 0$. Επιπλέον θα υποθέσουμε ότι οι συνιστώσες

των παραδειγμάτων εκπαίδευσης ανήκουν στο διάστημα $[0, 1]$ τα οποία ενδεχομένως να έχουν προκύψει από μετασχηματισμό κάποιων άλλων αρχικών παραδειγμάτων εκπαίδευσης. Αυτό είναι κάτι αρχικά σύνηθες στη Μηχανική Μάθηση, ειδικά όταν θέλουμε τα χαρακτηριστικά των παραδειγμάτων να είναι συγκρίσιμα και απαλλαγμένα από μονάδες μέτρησης. Επιπροσθέτως, θα εργαστούμε υποθέτοντας ότι οι δυο κλάσεις είναι γραμμικά διαχωρίσιμες, ή με άλλα λόγια, ότι υπάρχει γραμμικός ταξινομητής για τον οποίον ισχύει $c(x) = h(x)$ για όλα τα παραδείγματα εκπαίδευσης. Στόχος μας είναι η εύρεση ενός διανύσματος συντελεστών w με την προηγούμενη ιδιότητα.

Για να καταλάβουμε το μηχανισμό από τον οποίο θα προκύψουν οι συντελεστές, ας υποθέσουμε ότι έχουμε στη διάθεση μας μια προσωρινή (ακόμα και ατελή) έκδοση του ταξινομητή. Αν του δοθεί ένα παράδειγμα x τότε εκείνος θα επιστρέψει την υπόθεση του $h(x)$. Αν αυτή διαφέρει από την πραγματική κλάση $c(x) \neq h(x)$ τότε οι συντελεστές w_i δεν είναι τέλειοι και για αυτό πρέπει να τροποποιηθούν. Η τροποποίηση αυτή είναι λογικό να γίνει ως εξής:

- Αν $c(x) = 1$ και $h(x) = 0$ τότε αυτό συμβαίνει μόνο όταν $w^T x = \sum_{i=0}^{n} w_i x_i \leq 0$, και αυτό αποτελεί ένδειξη ότι οι συντελεστές είναι μικροί. Αν λοιπόν αυξήσουμε τους συντελεστές τότε το άθροισμα $\sum_{i=0}^{n} w_i x_i$ ενδέχεται να ξεπεράσει το 0 και έτσι να ταξινομηθεί ως θετικό.

- Ομοίως, αν $c(x) = 0$ και $h(x) = 1$ οι συντελεστές πρέπει να μειωθούν ώστε το άθροισμα $\sum_{i=0}^{n} w_i x_i$ να γίνει αρνητικό.

Μπορούμε τώρα, με ιδιαίτερο ενδιαφέρον να διαπιστώσουμε ότι τα παραπάνω μπορούν να συνοψιστούν με τη χρήση του παρακάτω κανόνα ανανέωσης:

$$w_i \leftarrow w_i + \eta[c(x) - h(x)]x_i, \tag{1.2}$$

όπου $\eta \in (0, 1]$, μια σταθερά που καθορίζει το μέγεθος επιρροής στούς συντελεστές. Ας δούμε τώρα τις βασικές πτυχές του κανόνα ανανέωσης :

1. *Σωστή Δράση.* Αν $c(x) = h(x)$ τότε $[c(x)-h(x)] = 0$ και άρα όλοι οι συντελεστές παραμένουν αμετάβλητοι. Αν $c(x) = 1$ και $h(x) = 0$ τότε $[c(x) - h(x)] = 1$ που σημαίνει ότι οι συντελεστές θα οδηγηθούν σε **αύξηση**. Αν από την άλλη μεριά, $c(x) = 0$ και $h(x) = 1$ τότε $[c(x) - h(x)] = -1$ που σημαίνει ότι οι συντελεστές θα οδηγηθούν σε **μείωση**.

2. *Κατάλληλη επιρροή στους συντελεστές.* Οι συντελεστές που θα επηρεαστούν περισσότερο είναι εκείνοι που διαθέτουν υψηλότερες τιμές στα χαρακτηριστικά τους.

3. *Ελεγχόμενη αλλαγή των συντελεστών.* Η ποσότητα αλλαγής των συντελεστών ελέγχεται από τον **λόγο εκμάθησης** η του οποίου η τιμή ανήκει στο διάστημα $(0, 1]$ και επιλέγεται από το χρήστη.

Ο κανόνας ανανέωσης (1.1) είναι βάση του Αλγορίθμου Perceptron ο οποίος παρουσιάζεται στον Πίνακα 1.1. Ο αλγόριθμος ξεκινά με την αρχικοποίηση των συντελεστών σε μικρές τυχαίες τιμές. Έπειτα τα παραδείγματα εκπαίδευσης παρουσιάζονται ένα προς ένα. Μετά από κάθε παρουσίαση, κάθε συντελεστής του ταξινομητή τροποποιείται με βάση τον κανόνα (1.1). Η διαδικασία τελειώνει μόνο σε περίπτωση όπου όλα τα παραδείγματα εκπαίδευσης έχουν ταξινομηθεί σωστά, ενώ σε αντίθετη περίπτωση επαναλαμβάνεται ξεκινώντας από το πρώτο παράδειγμα κ.ο.κ.

Πίνακας 1.1 Ο αλγόριθμος εκμάθησης Perceptron

Υπόθεση: Οι δυο κλάσεις $c(x) = 0$ και $c(x) = 1$ είναι γραμμικά διαχωρίσιμες.
Είσοδος: N παραδείγματα εκπαίδευσης $x_i = (1, x_{i1}, x_{i2}, \dots, x_{in})$, $i \in \{1, 2, ..., N\}$ με ετικέτες κλάσης $c(x_i)$
Βήμα 1: Αρχικοποίησε όλους τους συντελεστές w_j, $j \in \{0, 1, 2, ..., n\}$ σε μικρούς τυχαίους αριθμούς και επίλεξε τον λόγο εκμάθησης $\eta \in (0, 1]$.
Βήμα 2: Για κάθε παράδειγμα εκπαίδευσης x_i, $i \in \{1, 2, \dots, N\}$ με κλάση $c(x_i)$:

- Θέσε $h(x_i) = 1$ αν $\sum_{j=0}^{n} w_j x_{ij} > 0$, και $h(x_i) = 0$ διαφορετικά.
- Ανανέωσε κάθε συντελεστή με τον κανόνα, $w_j \leftarrow w_j + \eta[c(x_i) - h(x_i)]x_{ij}$

Βήμα 3: Αν $c(x_i) = h(x_i)$ για κάθε $i \in \{1, 2, ..., N\}$, τερμάτισε, διαφορετικά επίστεψε στο βήμα 2.
Έξοδος: Οι συντελεστές w_i του γραμμικού ταξινομητή $w^T x = 0$.

Παρατήρησεις

1. **Κριτήριο τερματισμού.** Αν τα παραδείγματα εκπαίδευσης είναι γραμμικά διαχωρίσιμα, ο παραπάνω αλγόριθμος θα τερματίσει σε πεπερασμένο αριθμό βημάτων ανεξάρτητα από πλήθος των χαρακτηριστικών, τις αρχικές τιμές των συντελεστών και τον λόγο εκμάθησης. Σε αντίθετη περίπτωση ο αλγόριθμος δεν τερματίζει εφόσον είναι αδύνατο η πραγματική και η υποθετική κλάση να συμπίπτουν για όλα τα παραδείγματα εκπαίδευσης.
2. **Ευαισθησία στην αρχικοποίηση.** Για διαφορετικές τιμές των αρχικών συντελεστών w_i και του λόγου εκμάθησης η, ο αλγόριθμος ενδέχεται να τερματίσει σε διαφορετικό αριθμό επαναλήψεων και να εξάγει διαφορετικούς ταξινομητές.
3. **Μοναδικότητα λύσης.** Αν τα παραδείγματα εκπαίδευσης είναι γραμμικά διαχωρίσιμα τότε υπάρχουν άπειροι γραμμικοί ταξινομητές. Ποιος όμως είναι ο "καλύτερος" και με ποιό κριτήριο; Για περισσότερες πληροφορίες παραπέμπουμε τον αναγνώστη σε **αλγόριθμους μηχανής διανυσμάτων υποστήριξης (support vector machines)** [**2**].

1.1.2 Ταξινομητές k-πλησιέστερων γειτόνων (k-nearest neighbors)

Η λογική του είδους ταξινόμησης με βάση τους k-πλησιέστερους γείτονες είναι να αποδοθεί σε ένα μελλοντικό δεδομένο η κλάση στην οποία βρίσκονται στην πλειονότητα τους τα k πιο κοντινά του παραδείγματα. Ας υποθέσουμε ότι έχουμε στη διάθεση μας ένα πλήθος από παραδείγματα εκπαίδευσης, τα οποία ανήκουν είτε στην θετική (+), είτε στην αρνητική κλάση (-). Αν λόγου χάρη, θέλουμε να ταξινομήσουμε ένα μελλοντικό δεδομένο σημείο που το πλησιέστερο παράδειγμα σε αυτό είναι θετικό και χρησιμοποιούμε τον **1-πλησιέστερο γείτονα** ως ταξινομητή, τότε το δεδομένο θα χαρακτηριστεί και αυτό ως θετικό (+). Αν από την άλλη πλευρά γνωρίζουμε ότι από τα 3 πλησιέστερα παραδείγματα είναι 2 αρνητικά (-) και ένα θετικό (+), τότε το δεδομένο θα χαρακτηριστεί ως αρνητικό σε περίπτωση που χρησιμοποιούμε τον ταξινομητή **3-πλησιέστερων γειτόνων**.

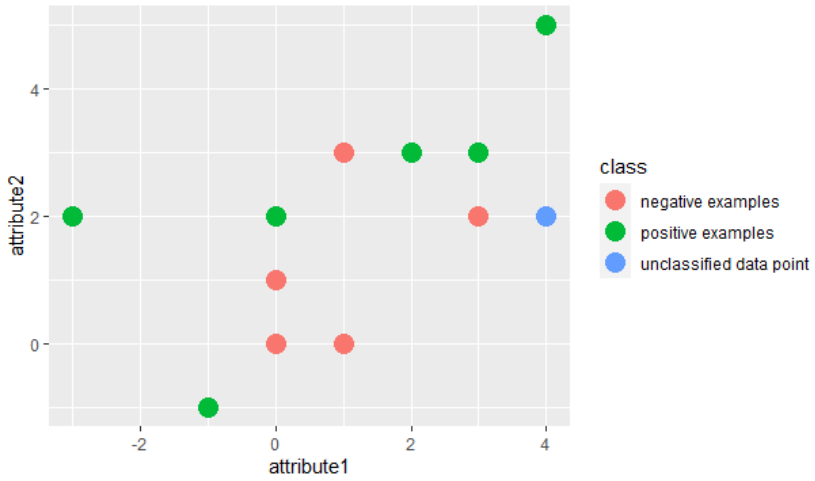
Φυσικά η μέτρηση των αποστάσεων σε έναν Ευκλείδιο χώρο όπως ο $\mathbb{R}^n$ δεν αποτελεί πρόβλημα. Έστω $x = (x_1, x_2, ..., x_n)$ και $y = (y_1, y_2, ..., y_n)$, δύο διανύσματα στον $\mathbb{R}^n$. Η απόσταση του x από το y ως προς την **ευκλείδια μετρική** είναι:

$$d(x, y) = ||x - y|| = \left(\sum_{i=1}^{n} (x_i - y_i)^2 \right)^{\frac{1}{2}} \tag{1.3}$$

Παράδειγμα 1.2 Ας υποθέσουμε ότι έχουμε στη διάθεση μας τα ακόλουθα 10 παραδείγματα εκπαίδευσης καθένα από τα οποία αποτελεί ένα διάνυσμα 2 χαρακτηριστικών:

Παραδείγματα εκπαίδευσης	Κλάση
$(0, 2)$	$+$
$(0, 0)$	$-$
$(4, 5)$	$+$
$(1, 0)$	$-$
$(0, 1)$	$-$
$(1, 3)$	$-$
$(2, 3)$	$+$
$(3, 2)$	$-$
$(-1, -1)$	$+$
$(3, 3)$	$+$

Ας υποθέσουμε επιπλέον ότι έχουμε στη διάθεση μας το **αταξινόμητο δεδομένο** $\mathbf{u} = (\mathbf{4}, \mathbf{2})$ το οποίο επιθυμούμε να ταξινομήσουμε. Στο Σχήμα 1.2 δίνεται η γεωμετρική εποπτεία του Παραδείγματος 1.2:



Σχήμα 1.2: Αναπαράσταση των παραδειγμάτων εκπαίδευσης και του αταξινόμητου δεδομένου στο επίπεδο.

Ταξινόμηση του u με τον 1-πλησιέστερο γείτονα
Εύκολα μπορούμε να επαληθεύσουμε ότι το πλησιέστερο παράδειγμα από το u είναι το $(3,2) \in (-)$ με απόσταση $d(u,(3,2)) = \sqrt{(4-3)^2+(2-2)^2} = 1$. Συνεπώς με βάση τον 1-πλησιέστερο γείτονα το u θα χαρακτηριστεί ως **αρνητικό**.

Ταξινόμηση του u με τους 3-πλησιέστερους γείτονες
Εύκολα μπορούμε να επαληθεύσουμε ότι τα 3 πλησιέστερα παραδείγματα στο u είναι:

1. Το $(3,2) \in (-)$ με απόσταση $d(u,(3,2)) = \sqrt{(4-3)^2+(2-2)^2} = 1$.
2. Το $(3,3) \in (+)$ με απόσταση $d(u,(3,3)) = \sqrt{(4-3)^2+(2-3)^2} = \sqrt{2}$
3. Το $(2,3) \in (+)$ με απόσταση $d(u,(2,3)) = \sqrt{(4-2)^2+(2-3)^2} = \sqrt{5}$.

Οι δύο θετικοί ψήφοι έναντι της μιας αρνητικής θα έχει ως συνέπεια το u να χαρακτηριστεί ως **θετικό**.

Ταξινόμηση του u με τους 5-πλησιέστερους γείτονες
Εύκολα μπορούμε να επαληθεύσουμε ότι τα 5 πλησιέστερα παραδείγματα στο u είναι:

1. Το $(3,2) \in (-)$ με απόσταση $d(u,(3,2)) = \sqrt{(4-3)^2+(2-2)^2} = 1$.
2. Το $(3,3) \in (+)$ με απόσταση $d(u,(3,3)) = \sqrt{(4-3)^2+(2-3)^2} = \sqrt{2}$.
3. Το $(2,3) \in (+)$ με απόσταση $d(u,(2,3)) = \sqrt{(4-2)^2+(2-3)^2} = \sqrt{5}$.

4. Το $(4,5) \in (+)$ με απόσταση $d(u,(4,5)) = \sqrt{(4-4)^2+(2-5)^2} = 3$.

5. Το $(1,3) \in (-)$ με απόσταση $d(u,(1,3)) = \sqrt{(4-1)^2+(2-3)^2} = \sqrt{10}$.

Οι 3 θετικοί ψήφοι έναντι των 2 αρνητικών θα έχει ως συνέπεια το u να χαρακτηριστεί ως **θετικό**. ■

Παρατηρήσεις

1. Γίνεται αντιληπτό πως για διαφορετικές τιμές του k οι ταξινομητές k-πλησιέστερων γειτόνων ενδέχεται να βρίσκονται σε «διαφωνία», δηλαδή να ταξινομούν σε διαφορετικές κλάσεις τα ίδια δεδομένα σημεία. Στην πράξη το επιθυμητό θα ήταν η εξεύρεση εκείνου του αριθμού k που θα έχει το **ελάχιστο ποσοστό λανθασμένης ταξινόμησης** σε μελλοντικά δεδομένα. Αυτό όμως δεν είναι πάντα εύκολο. Μικρές τιμές του k οδηγούν σε έλλειψη πληροφορίας και άρα είναι σχετικά αναξιόπιστες, και από την άλλη πλευρά μεγάλες τιμές του k τείνουν να ταξινομήσουν το δεδομένο με βάση την κλάση που έχει τα περισσότερα παραδείγματα, το οποίο είναι εμφανώς προβληματικό. Επομένως στην προσπάθεια εξεύρεσης εκείνου του k που ελαχιστοποιεί το ποσοστό λανθασμένης ταξινόμησης θα πρέπει να λαμβάνεται υπόψιν η δυσλειτουργική συμπεριφορά του στις ακραίες τιμές.

2. Σε περιπτώσεις που το πλήθος των κλάσεων είναι 2 όπως και στο Παράδειγμα 1.2 το k πρέπει να είναι αναγκαστικά περιττός αριθμός ώστε να αποφεύγονται οι «ισοπαλίες». Γενικά αν το πλήθος των κλάσεων είναι n, με $n > 2$, και υπάρχουν «ισοπαλίες» μεταξύ κάποιων κλάσεων τότε ο αποφασίζων θα πρέπει να δώσει ένα μηχανισμό που θα επιλέγει σε ποια κλάση θα εντάξει το δεδομένο σημείο.

Στον Πίνακα 1.2 δίνεται η πιο απλή έκδοση του αλγόριθμου ταξινόμησης με βάση τους k πλησιέστερους γείτονες.

Πίνακας 1.2 Η απλούστερη έκδοση του ταξινομητή k πλησιέστερων γειτόνων [**2**]

Είσοδος: N παραδείγματα εκπαίδευσης $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $i \in \{1,2,\ldots,N\}$ με ετικέτες κλάσης $c(x_i)$, ένα δεδομένο σημείο x_0 (χωρίς ετικέτα κλάσης) προς ταξινόμηση και ο αριθμός k.

Βήμα 1: Υπολόγισε $d(x_0, x_i)$ για κάθε i και βρες τα k πλησιέστερα παραδείγματα στο x_0.
Βήμα 2: Βρες την κλάση c που ανήκουν τα περισσότερα από το k πλησιέστερα παραδείγματα.
Βήμα 3: θέσε $c(x_0) = c$ ως την κλάση του δεδομένου σημείου x_0.
Έξοδος: Η κλάση $c(x_0)$ του δεδομένου σημείου x_0.

1.2 Μη-Επιβλεπόμενη Μάθηση

Ενώ η Επιβλεπόμενη Μάθηση εστιάζει στην εύρεση ταξινομητών από κατηγοριοποιημένα παραδείγματα, η Μη-Επιβλεπόμενη Μάθηση επικεντρώνει το ενδιαφέρον της στην ανακάλυψη κάποιας πιθανής δομής που κρύβεται πίσω από μη κατηγοριοποιημένα δεδομένα. Ένα από τα σημαντικότερα ζητήματα της Μη-Επιβλεπόμενης Μάθησης, μεταξύ άλλων[3], είναι η αναζήτηση κάποιων ομάδων που ονομάζονται **συστάδες (clusters)** στις οποίες ανήκουν όμοια παραδείγματα. Σε αυτό το κεφάλαιο θα γίνει μια σύντομη περιγραφή της **Ανάλυσης Συστάδων (Cluster Analysis)**, καθώς και του βασικότερου αλγορίθμου ανίχνευσης συστάδων, γνωστού και ως **αλγορίθμου των k-μέσων (k-means algorithm)**.

1.2.1 Ανάλυση Συστάδων

Όπως και στην Επιβλεπόμενη Μάθηση υποθέτουμε πως έχουμε στη διάθεση μας N παραδείγματα τα οποία περιγράφονται από n χαρακτηριστικά που όμως **δεν** συνοδεύονται και από ετικέτες κλάσης. Στόχος είναι η δημιουργία m συστάδων ($2 \leq m < N$) από όμοια παραδείγματα. Στο Σχήμα 1.3 μπορούμε να παρατηρήσουμε μερικά παραδείγματα 2 χαρακτηριστικών στα οποία φαίνεται εύλογη η ομαδοποίηση τους σε 3 ή περισσότερες συστάδες. Ασφαλώς, η οπτική ένταξη των παραδειγμάτων σε συστάδες ήταν εφικτή λόγω της διδιάστατης αναπαράστασης τους. Σε παραδείγματα με 4 ή και περισσότερα χαρακτηριστικά, που είναι ανέφικτη η οπτικοποίηση τους, είναι αδύνατος ο οπτικός εντοπισμός των συστάδων. Σε αυτή την περίπτωση, που άλλωστε είναι και η πιο συχνή στην πράξη, το πρόβλημα αντιμετωπίζεται με αλγόριθμους ανάλυσης συστάδων.

Περιγραφή Συστάδων

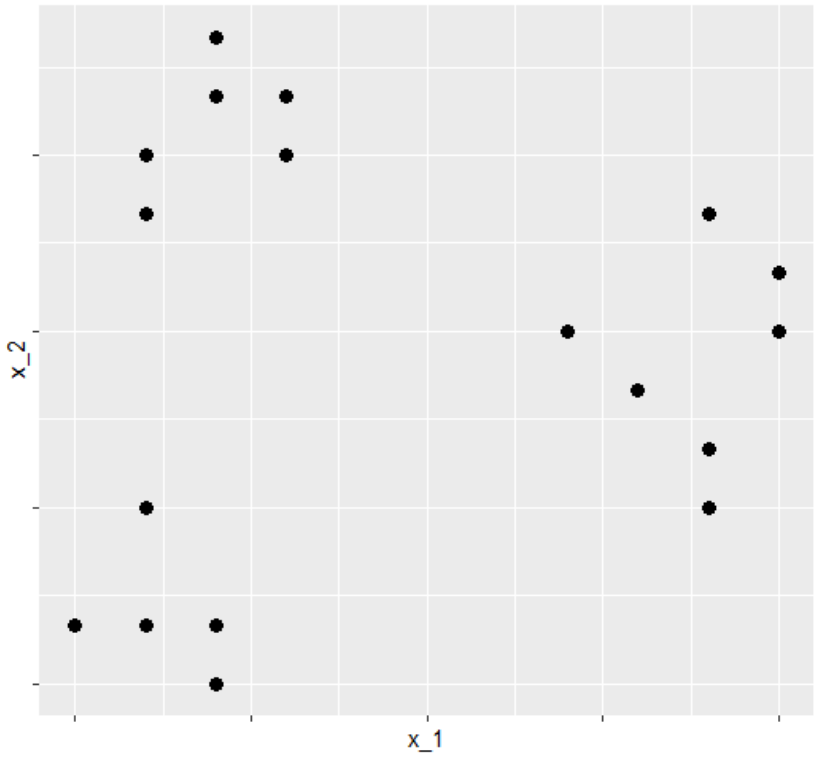
Έχοντας δώσει μια περισσότερο διαισθητική προσέγγιση στην έννοια της συστάδας, σε αυτό το σημείο είναι απαραίτητο να δώσουμε τον ορισμό του **κεντροειδούς (centroid)** που αποτελεί και τον απλούστερο τρόπο περιγραφής μιας συστάδας.

Ορισμός

Έστω $x_1, x_2, ..., x_N$ παραδείγματα n χαρακτηριστικών στον $\mathbb{R}^n$ με $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ $\forall\ i = 1, 2, ..., N$. **Κεντροειδές** των παραδειγμάτων $x_i,\ i \in \{1, 2, ..., N\}$ καλείται το διάνυσμα

$$c := \frac{1}{N}\left(\sum_{i=1}^{n} x_{i1}, \sum_{i=1}^{n} x_{i2}, \dots, \sum_{i=1}^{n} x_{in}\right) \tag{1.4}$$

[3]Ενδεικτικά αναφέρουμε την **Ανάλυση Κυρίων Συνιστωσών (Principal Component Analysis)** και την **Εκτίμηση Πυκνότητας (Density Estimation).**



Σχήμα 1.3: Συστάδες παραδειγμάτων στο επίπεδο.

Ο ρόλος του κεντροειδούς είναι απλός. Λειτουργεί ως «αντιπρόσωπος» μιας ολόκληρης συστάδας. Ένα νέο παράδειγμα που θέλουμε να εντάξουμε σε ήδη καθορισμένες συστάδες θα ανήκει στη συστάδα με την ελάχιστη απόσταση από το κεντροειδές της.

Δημιουργία Συστάδων

Στην επόμενη ενότητα παρουσιάζεται ο αλγόριθμος των k-μέσων ο οποίος διαθέτει τα εξής χαρακτηριστικά:

- Κάθε συστάδα που δημιουργείται δεν επικαλύπτει κάποια άλλη. Έτσι ένα παράδειγμα θα ανήκει σε μια, και μόνο μια, συστάδα.
- Ο επιθυμητός αριθμός συστάδων προς δημιουργία δίνεται ως είσοδος από το χρήστη.
- Στόχος είναι η δημιουργία συστάδων από παραδείγματα με το **ελάχιστο άθροισμα τετραγώνων** από το κεντροειδές τους. Πολλές φορές αυτό υλοποιείται επιτυχώς, χωρίς προβλήματα αλλά δυστυχώς δεν είναι πάντα εφικτό. Ο αλγόριθμος ενδέχεται να τερματίσει και σε κάποια υποβέλτιστη λύση.

1.2.2 Ο Αλγόριθμος των k-μέσων ($k-$means algorithm)

Έχοντας δώσει το εννοιολογικό πλαίσιο μια συστάδας και ορίσει το κεντροειδές, που αποτελεί τον απλούστερο τρόπο περιγραφής της, μπορούμε να προχωρήσουμε στην ανάπτυξη του αλγορίθμου των k-μέσων. Το «k» στο όνομα του συγκεκριμένου αλγόριθμου ταυτίζεται με το πλήθος των επιθυμητών συστάδων και δίνεται από τον χρήστη.

Το πρώτο βήμα του αλγορίθμου δημιουργεί k αρχικές συστάδες όπου κάθε παράδειγμα ανήκει αυστηρά σε 1 από αυτές. Έπειτα γίνεται ο υπολογισμός των κεντροειδών $c_i,\ i \in \{1, 2, ..., k\}$. Στο δεύτερο βήμα επιλέγεται ένα παράδειγμα x και υπολογίζονται οι αποστάσεις του από όλα τα κεντροειδή c_i. Αν το πλησιέστερο κεντροειδές είναι της ίδιας συστάδας στην οποία ανήκει το x, τότε μένουν όλα ως έχουν και ο αλγόριθμος προχωρά στο τελευταίο βήμα. Διαφορετικά το x μετατοπίζεται από την τρέχουσα συστάδα του και εντάσσεται σε αυτή με το πλησιέστερο κεντροειδές. Εν συνεχεία γίνεται και επανυπολογισμός των κεντροειδών c_i και ο αλγόριθμος προχωρά στο τελευταίο βήμα. Κατά το τελευταίο βήμα γίνεται ο εξής έλεγχος: αν για όλα τα παραδείγματα ισχύει ότι βρίσκονται σε ελάχιστη απόσταση από τις τρέχουσες συστάδες τους τότε ο αλγόριθμος σταματά. Διαφορετικά γίνεται επιστροφή στο δεύτερο βήμα.

Όλη η παραπάνω διαδικασία συνοψίζεται στον ψευδοκώδικα του Πίνακα 1.3. Εφόσον το κριτήριο του βήματος 4 επιτυγχάνεται πάντα, υπάρχει εγγύηση ότι ο αλγόριθμος θα τερματίσει σε πεπερασμένο αριθμό βημάτων.

Πίνακας 1.3 Ο αλγόριθμος των k-μέσων

Είσοδος: N παραδείγματα εκπαίδευσης $x_i = (x_{i1}, x_{i2}, ... , x_{in}),\ i \in \{1, 2, ..., N\}$ χωρίς ετικέτες κλάσης και ο αριθμός των συστάδων k.

Βήμα 1: Αρχικοποίησε[4] k διακεκριμένες συστάδες και υπολόγισε τα κεντροειδή c_j για κάθε $j \in \{1, 2, ..., k\}$.
Βήμα 2: Επίλεξε παράδειγμα $x_i,\ i \in \{1, 2, ..., N\}$ και υπολόγισε $d(x_i, c_j)$ για κάθε $j \in \{1, 2, ..., k\}$. Έστω $cluster(x_i)$ η τρέχουσα συστάδα του x_i και $m(x_i) := argmin_j\{d(x_i, c_j)\}$, η συστάδα με την μικρότερη απόσταση από το x_i.
Βήμα 3: Αν $cluster(x_i) = m(x_i)$ τότε συνέχισε στο Βήμα 4. Διαφορετικά αν $cluster(x_i) \neq m(x_i)$ τότε θέσε $cluster(x_i) \leftarrow m(x_i)$.
Βήμα 4: Αν για όλα τα παραδείγματα υπάρχει ταύτιση με τις τρέχουσες και πλησιέστερες συστάδες, αν δηλαδή $cluster(x_i) = m(x_i)\ \forall\ i \in \{1, 2, ..., N\}$, τότε τερμάτισε. Διαφορετικά επίστρεψε στο Βήμα 2.

Έξοδος: k διακεκριμένες συστάδες παραδειγμάτων.

---

[4] Η αρχικοποίηση μπορεί να γίνει είτε με κάποιο τυχαίο μηχανισμό, είτε ρητά από το χρήστη.

Στο Παράδειγμα 1.3 θα γίνει μια αριθμητική υλοποίηση του αλγορίθμου των k- μέσων.

Παράδειγμα 1.3 Ας υποθέσουμε ότι έχουμε στη διάθεση μας τα ακόλουθα 9 παραδείγματα (χωρίς ετικέτες κλάσης) 2 χαρακτηριστικών και επιθυμούμε να τα εντάξουμε σε **k=3 συστάδες**.

Παραδείγματα εκπαίδευσης
$(1, 2)$
$(4, 2)$
$(7, 2)$
$(2, 1)$
$(5, 2)$
$(8, 4)$
$(3, 3)$
$(2, 0)$
$(9, 3)$

Έστω ότι ξεκινάμε με τις παρακάτω αρχικές συστάδες:

Συστάδα 1	Συστάδα 2	Συστάδα 3
$x_1 = (1, 2)$	$x_4 = (4, 2)$	$x_7 = (7, 2)$
$x_2 = (2, 1)$	$x_5 = (2, 0)$	$x_8 = (8, 4)$
$x_3 = (5, 2)$	$x_6 = (3, 3)$	$x_9 = (9, 3)$

Για κάθε συστάδα υπολογίζουμε το κεντροειδές της:

	Συστάδα 1	Συστάδα 2	Συστάδα 3
	$x_1 = (1, 2)$	$x_4 = (4, 2)$	$x_7 = (7, 2)$
	$x_2 = (2, 1)$	$x_5 = (2, 0)$	$x_8 = (8, 4)$
	$x_3 = (5, 2)$	$x_6 = (3, 3)$	$x_9 = (9, 3)$
κεντροειδή	$c_1 = (2.66, 1.66)$	$c_2 = (3, 1.66)$	$c_3 = (8, 3)$

Στη συνέχεια επιλέγουμε παραδείγματα και υπολογίζουμε τις αποστάσεις τους από όλα τα κεντροειδή:

1. Για $x_1 = (1, 2)$
   - $d(x_1, c_1) = \sqrt{(1 - 2.66)^2 + (2 - 1.66)^2} \cong 1.69$
   - $d(x_1, c_2) = \sqrt{(1 - 3)^2 + (2 - 1.66)^2} \cong 2.03$
   - $d(x_1, c_3) = \sqrt{(1 - 8)^2 + (2 - 3)^2} \cong 7.07$

Παρατηρούμε ότι η ελάχιστη απόσταση είναι του x_1 από το c_1, δηλαδή $m(x_1) = 1$. Άρα $cluster(x_1) = m(x_1) = 1$ και συνεπώς προχωράμε σε επόμενο παράδειγμα.

2. Για $x_2 = (2, 1)$

   - $d(x_2, c_1) = \sqrt{(2 - 2.66)^2 + (1 - 1.66)^2} \cong 0.93$
   - $d(x_2, c_2) = \sqrt{(2 - 3)^2 + (1 - 1.66)^2} \cong 1.2$
   - $d(x_2, c_3) = \sqrt{(2 - 8)^2 + (1 - 3)^2} \cong 6.32$

   Παρατηρούμε ότι η ελάχιστη απόσταση είναι του x_2 από το c_1, δηλαδή $m(x_2) = 1$. Άρα $cluster(x_2) = m(x_2) = 1$ και συνεπώς προχωράμε σε επόμενο παράδειγμα.

3. Για $x_3 = (5, 2)$

   - $d(x_3, c_1) = \sqrt{(5 - 2.66)^2 + (2 - 1.66)^2} \cong 2.36$
   - $d(x_3, c_2) = \sqrt{(5 - 3)^2 + (2 - 1.66)^2} \cong 2.1$
   - $d(x_3, c_3) = \sqrt{(5 - 8)^2 + (2 - 3)^2} \cong 3.16$

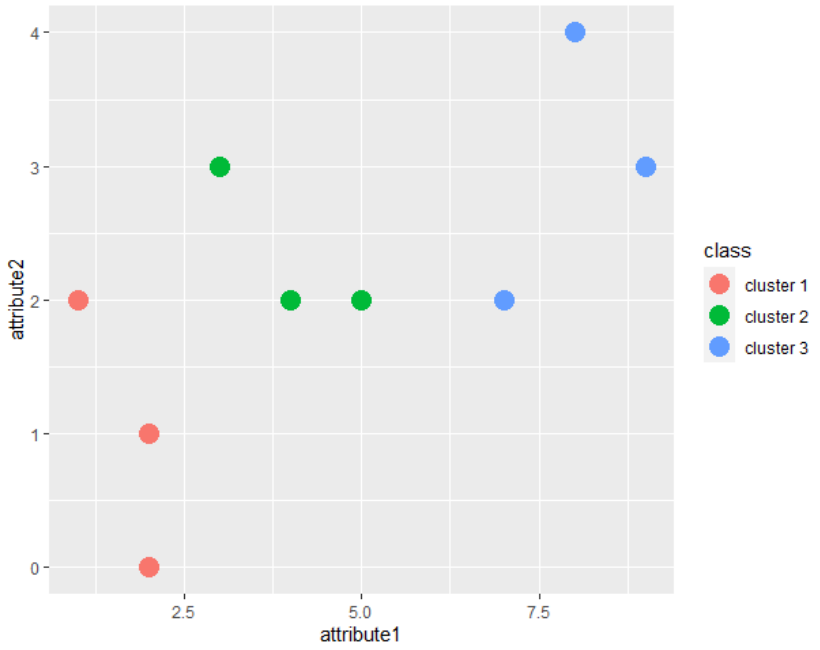
   Παρατηρούμε ότι η ελάχιστη απόσταση είναι του x_3 από το c_2, δηλαδή $m(x_3) = 2$. Άρα $1 = cluster(x_3) \neq m(x_3) = 2$. Επομένως το x_3 «μεταφέρεται» στη Συστάδα 2 και γίνεται επανυπολογισμός των κεντροειδών:

	Συστάδα 1	Συστάδα 2	Συστάδα 3
	$x_1 = (1, 2)$	$x_4 = (4, 2)$	$x_7 = (7, 2)$
	$x_2 = (2, 1)$	$x_5 = (2, 0)$	$x_8 = (8, 4)$
		$x_6 = (3, 3)$	$x_9 = (9, 3)$
		$x_3 = (5, 2)$	
κεντροειδή	$c'_1 = (1.5, 1.5)$	$c'_2 = (3.5, 1.75)$	$c'_3 = (8, 3)$

Σε αυτό το σημείο πρέπει να υπολογίσουμε εκ νέου τις αποστάσεις $d(x_i, c'_j)$ ξεκινώντας από το x_1, όπως και πριν. Ακολουθώντας την ίδια διαδικασία θα διαπιστώσουμε τώρα ότι τα παραδείγματα x_1, x_2, x_4 θα παραμείνουν στην τρέχουσα συστάδα τους ενώ το x_5 θα πρέπει να μετακινηθεί στη πρώτη συστάδα. Έτσι οδηγούμαστε στην ακόλουθη ομαδοποίηση:

	Συστάδα 1	Συστάδα 2	Συστάδα 3
	$x_1 = (1, 2)$	$x_4 = (4, 2)$	$x_7 = (7, 2)$
	$x_2 = (2, 1)$	$x_6 = (3, 3)$	$x_8 = (8, 4)$
	$x_5 = (2, 0)$	$x_3 = (5, 2)$	$x_9 = (9, 3)$
κεντροειδή	$c''_1 = (1.66, 1)$	$c''_2 = (4, 2.33)$	$c''_3 = (8, 3)$

Επαναλαμβάνοντας την ίδια διαδικασία θα διαπιστώνουμε ότι η τρέχουσα κλάση κάθε παραδείγματος $x_i,\ i = 1, 2, ..., 9$ ταυτίζεται με την πλησιέστερη και έτσι ο αλγόριθμος τερματίζει. Στο Σχήμα 1.4 φαίνεται η τελική συσταδοποίηση των αρχικών παραδειγμάτων.



Σχήμα 1.4: Συσταδοποίηση των διανυσμάτων του Παραδείγματος 1.3 με τον αλγόριθμο των k-μέσων. ■

1.3 Βασικές Αρχές Ενισχυτικής Μάθησης

Η Ενισχυτική Μάθηση αποτελεί το τρίτο βασικό είδος Μηχανικής Μάθησης. Ενώ η Επιβλεπόμενη Μάθηση και η Μη-Επιβλεπόμενη Μάθηση εστίαζαν σε προβλήματα δημιουργίας ταξινομητών ή στην ανακάλυψη κάποιας πιθανής δομής των δεδομένων αντίστοιχα, το ζητούμενο της Ενισχυτικής Μάθησης είναι διαφορετικό. Εδώ ο **μαθητής** [5] **(learner)** δεν γνωρίζει εκ των προτέρων ποιές αποφάσεις θα του αποφέρουν το μέγιστο δυνατό όφελος, και έτσι πρέπει ο ίδιος να το ανακαλύψει μόνος του δοκιμάζοντας τες. Θα λέγαμε με ευκολία πως η Ενισχυτική Μάθηση είναι εκείνο το είδος Μηχανικής

[5] Μαθητής στη Μηχανική μάθηση καλείται οποιοδήποτε πρόγραμμα μαθαίνει ένα μοντέλο μηχανικής μάθησης από την διαθέσιμη πληροφορία.

Μάθησης που προσεγγίζει καλύτερα τον τρόπο με τον οποίο μαθαίνουν και οι ζωντανοί οργανισμοί: **δοκιμή και σφάλμα**.

Στο πλαίσιο της Ενισχυτικής Μάθησης πρωταρχικό ρόλο λοιπόν παίζουν το **υποκείμενο ή πράκτορας**[6] (**agent**) και το **περιβάλλον** του (**enviroment**), τα οποία έρχονται σε συνεχή αλληλεπίδραση για ένα **πεπερασμένο ή άπειρο χρονικό ορίζοντα** N. Σε κάθε χρονική στιγμή το υποκείμενο βρίσκεται σε μια **κατάσταση** $S_t \in S$[7] η οποία συνοψίζει όλη την διαθέσιμη πληροφορία που κατέχει στο σύστημα που βρίσκεται. Στη συνέχεια, το υποκείμενο λαμβάνει στοχαστικά μια **απόφαση** A_t μέσα από ένα **σύνολο δυνατών αποφάσεων** $A(s_t)$[8] , για να ακολουθήσει μια **τυχαία αμοιβή ή ποινή** $R_t \in R$[9] από το περιβάλλον η οποία εξαρτάται στοχαστικά από την απόφαση που έλαβε το υποκείμενο. Αφού φανερωθεί η αμοιβή $R_t = r_t$ από το περιβάλλον, το υποκείμενο μεταβαίνει στην επόμενη κατάσταση $S_{t+1} = s_{t+1}$ και η διαδικασία επαναλαμβάνεται με τον ίδιο τρόπο. Η κατάσταση S_{t+1} εξαρτάται και αυτή στοχαστικά τόσο από την προηγούμενη κατάσταση S_t όσο και από την απόφαση A_t του προηγούμενου βήματος.

Σχήμα 1.5: Αλληλεπίδραση του υποκειμένου με το περιβάλλον του στο χρόνο.

Στόχος του υποκειμένου είναι να βελτιστοποιήσει το **μέσο αναμενόμενο άθροισμα των αμοιβών του** $\mathbb{E}[\sum_{t=1}^{N} R_t]$ (σε προβλήματα πεπερασμένου ορίζοντα) ή το **μέσο αποπληθωρισμένο άθροισμα των αμοιβών του** $\mathbb{E}[\sum_{t=1}^{\infty} R_t \gamma^{t-1}]$ (συνήθως σε προβλήματα άπειρου ορίζοντα). Ο αριθμός γ ονομάζεται **συντελεστής αποπληθωρισμού** και ανήκει στο διάστημα [0,1]. Ο συντελεστής αποπληθωρισμού καθορίζει την παρούσα αξία των μελλοντικών αμοιβών: μια αμοιβή που θα εισπραχθεί t χρονικά βήματα αργότερα θα αξίζει γ^{t-1} επί το πόσο που θα άξιζε αν είχε εισπραχθεί άμεσα.

Πολιτική στην Ενισχυτική Μάθηση ονομάζεται ο τρόπος συμπεριφοράς του υποκειμένου, βρισκόμενο σε οποιαδήποτε κατάσταση S_t. Φορμαλιστικά, αποτελεί την οικογένεια δεσμευμένων συναρτήσεων πιθανότητας[10] $\pi(A_t = a_t \mid S_t = s_t)$, $t \in \{1, 2, ..., N\}$

[6] Στην Ενισχυτική Μάθηση ο μαθητής καλείται πιο εξειδικευμένα υποκείμενο ή πράκτορας.

[7] Με S θα συμβολίζουμε το σύνολο όλων των δυνατών καταστάσεων.

[8] Με $A(s_t)$ θα συμβολίζουμε την σύνολο των δυνατών αποφάσεων που μπορούν να ληφθούν στην κατάσταση s_t.

[9] Με R θα ονομάζουμε το σύνολο όλων των πιθανών αμοιβών.

[10] Αν το σύνολο των δυνατών αποφάσεων $A(s) := \bigcup_{s_t \in S} A(s_t)$ είναι υπεραριθμήσιμο τότε η πολιτική

που εκφράζει την πιθανότητα το υποκείμενο βρισκόμενο στην κατάσταση s_t να πάρει την απόφαση a_t.

Αξία μιας πολιτικής π είναι η αναμενόμενη αμοιβή που θα εισπράξει το υποκείμενο ακολουθώντας την πολιτική π. Μαθηματικά συμβολίζεται με $u_\pi(s_t)$ και ορίζεται από την παρακάτω έκφραση:

$$u_\pi(s_t) := \mathbb{E}_\pi \left[\sum_{k=0}^{N} \gamma^k R_{t+k+1} \mid S_t = s_t \right], \; s_t \in S$$

Ενώ η αμοιβή r_t εκφράζει την **άμεση αξία** που θα εισπραχθεί σε ένα βήμα, η συνάρτηση βέλτιστης τιμής $u_\pi(s_t)$ εκφράζει την **μακροπρόθεσμη αξία** που αναμένεται να συλλεχθεί αν στη συνέχεια ακολουθηθεί η πολιτική π. Είναι δυνατόν σε πολλές περιπτώσεις λοιπόν να προτιμηθεί μια απόφαση με μικρή αμοιβή για το παρόν, προς όφελος μιας μεγαλύτερης στο μέλλον.

Το **βασικό δίλημμα** σε όλα τα προβλήματα Ενισχυτικής Μάθησης είναι να βρεθεί ισορροπία ανάμεσα στην **εξερεύνηση νέων επιλογών** και στην **εκμετάλλευση της ήδη αποκτηθείσας γνώσης**. Η εξερεύνηση νέων επιλογών δίνει μακροπρόθεσμα την ευκαιρία για εύρεση μιας πολιτικής η οποία θα αποφέρει ενδεχομένως καλύτερες αμοιβές σε σχέση με μια τρέχουσα πολιτική, όμως βραχυπρόθεσμα μπορεί να είναι επιζήμια καθώς πολλές διαφορετικές επιλογές οδηγούν στην μη επιλογή αυτής (ή αυτών) που είναι βέλτιστη. Από την άλλη μεριά, η εκμετάλλευση της ήδη αποκτηθείσας γνώσης, οδηγεί σε εντελώς αντιδιαμετρικές συνέπειες. Βραχυπρόθεσμα, αν δεν υπάρχει αλλαγή πολιτικής, γίνεται πλήρης εκμετάλλευση της πληροφορίας που έχει συλλεχθεί ως τώρα η οποία ενδεχομένως να αποφέρει σε κάθε βήμα κάποια ικανοποιητική αμοιβή, αλλά μακροπρόθεσμα αν η τρέχουσα πολιτική αποφάσεων δεν είναι βέλτιστη τότε σίγουρα μελλοντικά δεν θα εισπραχθεί συγκεντρωτικά η μέγιστη δυνατή συνολική αμοιβή. Το παραπάνω δίλημμα είναι γνωστό στη βιβλιογραφία ως δίλημμα **εξερεύνησης εναντίον εκμετάλλευσης (exploration vs exploitation dillema)**.

Στο **Κεφάλαιο 2** θα περιοριστούμε σε προβλήματα που η δέσμη των δυνατών αποφάσεων $A(s_t)$ παραμένει σταθερή για κάθε s_t. Τα προβλήματα που ανήκουν στην παραπάνω κατηγορία εντάσσονται σύμφωνα με τη διεθνή βιβλιογραφία στο πλαίσιο των ***Multi-Armed-Bandits*** ή εν συντομία **MAB's** , αν και για την ελληνική ακόμα παραμένει ασαφής. Στην παρούσα εργασία θα χρησιμοποιείται και ο όρος **Στοχαστικές Μηχανές Επιβράβευσης** ως ισοδύναμος, όπου αυτό θεωρείται σκόπιμο. Ο λόγος που αφιερώνεται αυτοτελές κεφάλαιο στην συγκεκριμένη κατηγορία προβλημάτων στην παρούσα εργασία είναι διττός. Αρχικά δεν θα μπορούσε να λείπει από μια εισαγωγική εργασία Ενισχυτικής Μάθησης ένας τόσο σημαντικός υποκλάδος της και πλούσιος σε θεωρία και εφαρμογές (Χρηματοοικονομικά, Marketing κλπ.). Ο δεύτερος λόγος είναι

π αποτελεί οικογένεια συναρτήσεων πυκνότητας πιθανότητας αλλά είναι εκτός πλαισίου της παρούσας εργασίας.

για την ομαλή μετάβαση του αναγνώστη στο Κεφάλαιο 3 όπου το πλαίσιο γενικεύεται σε δέσμες αποφάσεων που εξαρτώνται από την εκάστοτε κατάσταση.

Στο **Κεφάλαιο 3**, που κάθε κατάσταση χαρακτηρίζεται και από το δικό της σύνολο αποφάσεων, το ευρύτερο πρόβλημα εντάσσεται στις ***Μαρκοβιανές Διαδικασίες Αποφάσεων (Markov Decision Processes)***, ή εν συντομία στις **ΜΔΑ**. Ο τελευταίος όρος είναι γνωστός και στην Επιχειρησιακή Έρευνα, καθώς η επίλυση προβλημάτων υπό αυτό το πρίσμα έχει μελετηθεί μέσω του **Δυναμικού Προγραμματισμού**. Για ποιο λόγο λοιπόν η Ενισχυτική Μάθηση να διαπραγματεύεται ένα αυτοτελές, κατά τα άλλα, μαθηματικό αντικείμενο στο οποίο έχουν διατυπωθεί η θεωρία και οι μέθοδοι επίλυσης; Η απάντηση σε αυτό το ερώτημα βρίσκεται στην καρδιά της Ενισχυτικής Μάθησης και μπορεί να αποδοθεί σε 2 βασικά προβλήματα τα οποία εντοπίζονται σε πραγματικές εφαρμογές:

1. ***Κατάρα της μοντελοποίησης (curse of modeling)*** Σε πολλά προβλήματα είναι μη ρεαλιστικό να καθορίσουμε εκ των προτέρων τη συνάρτηση πιθανότητας (ή πυκνότητας) των αμοιβών του περιβάλλοντος καθώς δεν έχουμε την πλήρη γνώση της συμπεριφοράς του. Έτσι ενδέχεται σε πολλες περιπτώσεις ο καθορισμός των παραπάνω πιθανοτήτων να είναι δεσμευτικός.

2. ***Κατάρα της διάστασης (curse of dimensionality)*** Σε πολλά προβλήματα στα οποία μπορεί να υπάρχει και αυστηρό μοντέλο για το περιβάλλον, τα σύνολα των δυνατών αποφάσεων και καταστάσεων είναι τόσο μεγάλα που καθιστούν αδύνατη την εύρεση της ακριβούς βέλτιστης λύσης ακόμα και από τους πιο σύγχρονους υπολογιστές.

Ενώ λοιπόν στην Επιχειρησιακή Έρευνα γίνεται προσπάθεια εύρεσης της ακριβούς βέλτιστης λύσης της εκάστοτε εφαρμογής, στην Ενισχυτική Μάθηση γίνεται προσπάθεια εύρεσης μιας προσεγγιστικά βέλτιστης λύσης η οποία θα αντιμετωπίζει όμως με επιτυχία τα παραπάνω βασικά προβλήματα.

Κεφάλαιο 2

Multi-Armed Bandits

Το σπουδαιότερο χαρακτηριστικό που διαχωρίζει την Ενισχυτική Μάθηση από άλλα είδη Μηχανικής Μάθησης είναι ότι χρησιμοποιεί την πληροφορία των παραδειγμάτων εκπαίδευσης για να εκτιμήσει τις βέλτιστες δράσεις αντί να εκπαιδεύεται γνωρίζοντας ποιες είναι οι «σωστές». Αυτό είναι άρρηκτα συνδεδεμένο με το ένα εκ των δυο δομικών στοιχείων της Ενισχυτικής Μάθησης που είναι η **εξερεύνηση**. Με τον όρο εξερεύνηση στην Ενισχυτική Μάθηση εννοούμε τη συνεχή απόπειρα του υποκειμένου να δοκιμάζει όλες τις διαθέσιμες επιλογές για να ανακαλύψει ποια είναι «προικισμένη» με την μεγαλύτερη μέση αμοιβή. Το παραπάνω δομικό στοιχείο έρχεται να συμπληρώσει η **εκμετάλλευση** της ήδη αποκτηθείσας γνώσης, η οποία θα αξιοποιήσει τη συγκεντρωμένη πληροφορία με σκοπό να επιλέγεται συχνότερα εκείνη η επιλογή για την οποία υπάρχουν ενδείξεις ότι είναι η βέλτιστη. Παρά το γεγονός ότι τα δυο παραπάνω δομικά στοιχεία αποτελούν αντίθετα από άποψη στρατηγικής για την επίλυση ενός προβλήματος, ο κατάλληλος συνδυασμός τους είναι τελικά εκείνος που θα δώσει τη λύση σε ένα πρόβλημα Ενισχυτικής Μάθησης.

Σε αυτό το Κεφάλαιο θα εστιάσουμε σε προβλήματα που η δέσμη των δυνατών αποφάσεων παραμένει σταθερή για κάθε χρονική στιγμή. Ωστόσο, στα πλαίσια της παρούσας Εργασίας, θα περιοριστούμε σε προβλήματα στα οποία:

- Οι συναρτήσεις πιθανότητας (ή πυκνότητας) του περιβάλλοντος θα θεωρούνται άγνωστες, αλλά στάσιμες.

- Οι συναρτήσεις πιθανότητας (ή πυκνότητας) του περιβάλλοντος δεν υποθέτουμε ότι διαθέτουν κάποια παραμετρική δομή.

Βέβαια αξίζει να αναφερθεί πως καθεμία από τις παραπάνω υποθέσεις μόνο αμελητέες δεν πρέπει να θεωρούνται στη γενική μορφή του προβλήματος. Στην πράξη είναι πολύ συχνό οι κατανομές των αμοιβών του περιβάλλοντος να μεταβάλλονται με το χρόνο. Λόγου χάρη μπορεί σε κάθε βήμα ένας επενδυτής να πρέπει να αποφασίσει σε ποια από

n μετοχές να επενδύσει τα χρήματα του. Όμως είναι γνωστό ότι οι αποδόσεις των μετοχών χαρακτηρίζονται από μη στασιμότητα. Έτσι η παράβλεψη αυτής της υπόθεσης θα δημιουργούσε προβλήματα. Όσον αφορά το δεύτερο χαρακτηριστικό, πολλές είναι και οι περιπτώσεις όπου το πρόβλημα από τη «φύση» του διακατέχεται από κάποια παραμετρική δομή. Ας υποθέσουμε το εξής παράδειγμα: Μια εταιρεία Digital Marketing θέλει να αποφασίσει σε ποια από k θέσεις σε μια ιστοσελίδα θα τοποθετήσει μια διαφήμιση. Έτσι, πειραματίζεται βάζοντας τη διαφήμιση σε όλες τις δυνατές θέσεις για να αποφανθεί ποια είναι εκείνη που προτιμάται κατά κόρον από το εμπορικό της κοινό. Οι αμοιβές που είναι λογικό να υποθέσουμε ότι εκλαμβάνει είναι:

$$\begin{cases} 0 & \text{, αν πελάτης δεν έκανε «click» στη διαφήμηση} \\ 1 & \text{, αν ο πελάτης έκανε «click» στη διαφήμηση} \end{cases}$$

Γίνεται αντιληπτό πως σε αυτό το παράδειγμα οι αμοιβές του περιβάλλοντος για κάθε θέση $i = 1, 2, \dots k$ ακολουθούν την κατανομή Bernoulli(p_i). Φυσικά αυτή η πληροφορία πρέπει οπωσδήποτε να αξιοποιηθεί καθώς θα οδηγήσει ενδεχομένως σε ισχυρότερα συμπεράσματα σε σχέση με κάποιο αλγόριθμο ο οποίος είναι πιο γενικός και δεν το λαμβάνει υπόψην.

2.1 Το Πρόβλημα k-Armed Bandit

Ας υποθέσουμε ότι ερχόμαστε αντιμέτωποι με το ακόλουθο πρόβλημα. Βρισκόμαστε συνεχώς σε μια κατάσταση στην οποία πρέπει να επιλέγουμε μεταξύ k διαθέσιμων επιλογών ή δράσεων. Μετά από κάθε απόφαση μας φανερώνεται μια αριθμητική αμοιβή η οποία προέρχεται από κάποια στάσιμη κατανομή πιθανότητας. Σκοπός μας είναι να μεγιστοποιήσουμε το αναμενόμενο άθροισμα των αμοιβών σε ένα καθορισμένο χρονικό ορίζοντα. Προφανώς αν ξέραμε εκ των προτέρων ποια δράση διαθέτει την μεγαλύτερη μέση αμοιβή τότε το καλύτερο που θα μπορούσαμε να κάνουμε θα ήταν να επιλέγουμε πάντοτε εκείνη. Το ενδιαφέρον λοιπόν είναι με ποιο τρόπο πρέπει να δράσουμε όταν δεν γνωρίζουμε a-priori ποια είναι η βέλτιστη. Έτσι είναι φυσιολογικό να προσπαθήσουμε να δώσουμε μια απάντηση στα εξής ερωτήματα: Πώς πρέπει να κινηθούμε από άποψη στρατηγικής ώστε να έχουμε ελπίδες να εξασφαλίσουμε μια μέση αμοιβή «κοντά» στη βέλτιστη δυνατή σε ένα βάθος χρόνου; Θα πρέπει να επιλέγουμε συνεχώς εκείνη η οποία διαθέτει τη μεγαλύτερη μέση δειγματική αμοιβή και άρα αυτή που έχει τις περισσότερες ενδείξεις ότι είναι η βέλτιστη, ή πρέπει να αφιερώνουμε το χρόνο μας στο να τις δοκιμάζουμε όλες για χάρη της αβεβαιότητας και σε τι ρυθμό;

Παρά το γεγονός ότι τέτοια προβλήματα τύπου **εξερεύνησης-εκμετάλλευσης** είχαν διατυπωθεί αρκετές δεκαετίες πίσω, μελετήθηκαν αναλυτικά και από μαθηματική σκοπιά από το 1930 και μετά με κύριο πρωτεργάτη τον **Robbins (1952)** κάτω από

την ομπρέλα του προβλήματος **Multi-Armed Bandit**[1] (**MAB**) ή **k-Armed Bandit** (όταν θέλουμε να δηλώσουμε με σαφήνεια το πλήθος των διαθέσιμων επιλογών). Η προέλευση της ονομασίας είναι λόγω της μεταφορικής συσχέτισης του προβλήματος με τους κουλοχέρηδες[2] των καζίνο με τη διαφορά ότι διαθέτει k λαβές αντί για μία[3]. Κατά τη συγκεκριμένη μεταφορά το πρόβλημα είναι ίδιου τύπου με αυτό που αντιμετωπίζει ένας παίχτης σε ένα καζίνο ο οποίος δοκιμάζοντας πολλούς κουλοχέρηδες προσπαθεί να ανακαλύψει ποιος από αυτούς του αποδίδει κατά μέσο όρο τα μεγαλύτερα χρηματικά έπαθλα. Όσον αφορά την Ελληνική απόδοση της συγκεκριμένης ονομασίας στην παρούσα Εργασία θα χρησιμοποιείται ο όρος **Στοχαστικές Μηχανές Επιβράβευσης (ΣΜΕ) με k λαβές** .

Πριν προχωρήσουμε στη διατύπωση του αυστηρού ορισμού για το k Armed Bandit, παρουσιάζουμε ενδεικτικά κάποιες εφαρμογές τους με σκοπό την καλύτερη κατανόηση του διπόλου εξερεύνησης-εκμετάλλευσης, με το οποίο όλοι μας, λίγο ή πολύ, έχουμε βρεθεί αντιμέτωποι.

- **Εφαρμογή 1**: Τοποθέτηση διαφήμισης από Συντάκτη ιστότοπου
  - **εκμετάλλευση**: Τοποθέτηση δημοφιλών άρθρων στην αρχική σελίδα.
  - **εξερεύνηση**: Τοποθέτηση νέων άρθρων στην αρχική σελίδα.
- **Εφαρμογή 2**: Χορήγηση Φαρμάκου από γιατρό σε ασθενή
  - **εκμετάλλευση**: Χορήγηση δοκιμασμένου φαρμάκου από τον γιατρό στον ασθενή.
  - **εξερεύνηση**: Χορήγηση ενός πειραματικού αλλά πολλά υποσχόμενου φαρμάκου από τον γιατρό στον ασθενή.
- **Εφαρμογή 3**: Εξόρυξη πετρελαίου από πετρελαϊκή εταιρεία
  - **εκμετάλλευση**: Εξόρυξη στην καλύτερη έως τώρα τοποθεσία.
  - **εξερεύνηση**: Εξόρυξη σε νέα τοποθεσία.
- **Εφαρμογή 4**: Μαθαίνοντας ένα παιχνίδι
  - **εκμετάλλευση**: Επιλογή κίνησης που ο παίχτης θεωρεί ως καλύτερη με βάση την εμπειρία του.
  - **εξερεύνηση**: Επιλογή μιας νέας, πειραματικής κίνησης.

[1] Η πρώτη επιστημονική αναφορά σε προβλήματα Bandit έγινε από τον William R. Thompson το 1933 σε ένα άρθρο στο περιοδικό Biometrica, το οποίο αφορούσε κλινικές δοκιμές.

[2] Οι κουλοχέρηδες είναι γνωστοί διεθνώς και ως bandits, δηλαδή «ληστές» επειδή έχουν την τάση να ζημιώνουν τον παίχτη.

[3] Η ισοδύναμη ονομασία k **One-Armed Bandits** ίσως αποτύπωνε καλύτερα τη συγκεκριμένη μεταφορά.

2.2 Το Λεξιλόγιο των Bandits

Ορισμός 2.1. Multi armed Bandit with k arms (MAB) ή **Στοχαστική Μηχανή Επιβράβευσης με k λαβές (ΣΜΕ)** ονομάζεται μια διατεταγμένη k-άδα κατανομών πιθανότητας της μορφής $(P_1, P_2, ..., P_k)$. Υποθέτουμε ότι όλες οι κατανομές έχουν πεπερασμένες μέσες τιμές που θα συμβολίζουμε με $\mu_1, \mu_2, ...\mu_k$, ενώ την μέγιστη θα τη συμβολίζουμε με μ^*, δηλαδή $\mu^* = max\{\mu_1, \mu_2, ..., \mu_k\}$ Τέλος, θα υποθέσουμε ότι υπάρχει $\mu' \in \{\mu_1, \mu_2, ...\mu_k\}$ τέτοιο ώστε $\mu^* - \mu' \neq 0$ διότι διαφορετικά το πρόβλημα είναι τετριμμένο.[4]

Όπως και στο γενικό πλαίσιο της Ενισχυτικής μάθησης είναι σημαντικό να αποσαφηνιστούν οι ακόλουθες έννοιες-κλειδιά προσαρμοσμένες στο ειδικότερο πλαίσιο των **MAB's**:

- Η **δέσμη των δυνατών αποφάσεων (ή επιλογών,ή δράσεων, ή λαβών)** Α στο πλαίσιο των MAB's παραμένει σταθερή και ταυτίζεται με το σύνολο $\{1, 2, ..., k\}$. Με A_t θα συμβολίζουμε την απόφαση που λήφθηκε στο γύρο t και με a^* αυτή που αντιστοιχεί στη λαβή με τη μέγιστη μέση τιμή[5], δηλαδή $a^* = argmax_{a \in A}\{\mu_a\}$.

- Η **Κατάσταση** s_t του υποκειμένου στο γύρο t χαρακτηρίζεται τόσο από τη χρονική στιγμή του γύρου όσο και από πληροφορίες και στατιστικά που έχουν συγκεντρωθεί έως τώρα από το υποκείμενο.

- **Πολιτική** π ονομάζεται η πιθανότητα, το υποκείμενο βρισκόμενο σε οποιαδήποτε κατάσταση s_t να επιλέξει κάποια απόφαση $a \in A$.

- Η **αμοιβή** του υποκειμένου X_t στον γύρο t, είναι τυχαία μεταβλητή και μάλιστα $(X_t|\ A_t = a) \sim P_a$ αφού η κατανομή της X_t εξαρτάται αποκλειστικά από ποια λαβή θα επιλέξει το υποκείμενο.

- Η **Αξία** $Q(a)$, μιας απόφασης a, είναι η μέση τιμή της αμοιβής που θα λάβουμε αν επιλέξουμε a, δηλαδή $Q(a) = E[X_t|A_t = a] = \mu_a$.

- **Μέγιστη Αξία** ορίζεται η ποσότητα $Q(a^*) = \max_{a \in A} Q(a) = \mu^*$.

- **Απώλεια** (**Regret**) στο γύρο t ορίζεται η συνάρτηση $r(t) = E[\mu^* - X_t|A_t = a_t] = \mu^* - Q(a_t)$ η οποία ουσιαστικά αποτυπώνει τη μέση αμοιβή που «χάνεται» επιλέγοντας την απόφαση a_t έναντι της βέλτιστης a^* στον γύρο t.

[4] Αν δεν ισχύει αυτή η υπόθεση τότε $\mu_1 = \mu_2 = ... = \mu_\kappa$ και άρα όποια πολιτική και αν ακολουθήσει το υποκείμενο θα είναι βέλτιστη.

[5] Σε περίπτωση που υπάρχουν τουλάχιστον 2 μέσες τιμές που είναι μέγιστες θέτουμε με μ^* αυθαίρετα μια εξ' αυτών.

- **Συνολική Απώλεια** (**Total Regret**) μέχρι και τον γύρο T ορίζεται η συνάρτηση:

$$R(T) = \sum_{t=1}^{T} r(t) = \sum_{t=1}^{T} (\mu^* - Q(A_t)) = T\mu^* - \sum_{t=1}^{T} Q(A_t) = T\mu^* - E\left[\sum_{t=1}^{T} X_t\right] \geq 0$$

Ο **στόχος** του υποκειμένου είναι η εύρεση εκείνης της πολιτικής που οδηγεί στη **μεγιστοποίηση** του αναμενόμενου αθροίσματος των αμοιβών του $E\left[\sum_{t=1}^{T} X_t\right]$ ή ισοδύναμα στην **ελαχιστοποίηση** της συνολικής του απώλειας $R(T)$. Η τελευταία, αποτελεί και το βασικότερο **μέτρο απόδοσης** μιας πολιτικής π. Τα μέτρα απόδοσης παίζουν καθοριστικό ρόλο στη σύγκριση των πολιτικών, και κατ΄ επέκταση των αλγορίθμων, που θα δούμε στις επόμενες ενότητες. Η **βέλτιστη πολιτική** σε αυτό το πρόβλημα είναι η απόφαση σε κάθε βήμα να ταυτίζεται με τη βέλτιστη δυνατή, δηλαδή αυτή για την οποία ισχύει $P(A_t = a^*) = 1 \ \forall \ t \ \in \mathbb{N}$. Πράγματι, αν για μια πολιτική ισχύει η παραπάνω ιδιότητα έχουμε:

$$R(T) = T\mu^* - \sum_{t=1}^{T} Q(A_t) = T\mu^* - \sum_{t=1}^{T} E[X_t | A_t = a^*] = T\mu^* - T\mu^* = 0$$

Ασφαλώς μια τέτοια πολιτική είναι αδύνατον να βρεθεί με βεβαιότητα καθώς το υποκείμενο δεν διαθέτει a-priori πληροφορία για τις μέσες τιμές $\mu_i, \ i = 1, ..., k$. Έτσι ο **ρεαλιστικός στόχος** του υποκειμένου μεταφέρεται στην εξεύρεση πολιτικών για τις οποίες

$$\lim_{t \to +\infty} P(A_t = a^*) \approx 1$$

2.3 Μέτρα Απόδοσης

Τα μέτρα απόδοσης αποτελούν δείκτες αξιολόγησης των διάφορων πολιτικών για την επίτευξη του στόχου του υποκειμένου. Στα πλαίσια της παρούσας εργασίας θα εστιάσουμε στα ακόλουθα μέτρα απόδοσης:

1. Συνάρτηση Απώλειας: $R(T)$
2. Οριακή πιθανότητα λήψης της βέλτιστης απόφασης: p_l
3. Δειγματικός μέσος των αμοιβών: $\overline{X_t}$

Η συνάρτηση απώλειας $R(T)$

Όπως θα δούμε και αναλυτικότερα στις ενότητες που ακολουθούν καλύτεροι αλγόριθμοι από την άποψη των μέτρων απόδοσης θα είναι εκείνοι που εκμεταλλεύονται και εξερευνούν για άπειρο αριθμό βημάτων **κάθε** διαθέσιμη επιλογή. Στην επόμενη πρόταση θα αποδείξουμε ότι κάτω από μια πολιτική π, η οποία εξερευνά σε άπειρο πλήθος γύρων κάθε διαθέσιμη επιλογή, η συνάρτηση συνολικής απώλειας συγκλίνει στο $+\infty$ με πιθανότητα 1.

Πρόταση 2.1 Σε ένα MAB με πολιτική άπειρης εξερεύνησης κάθε διαθέσιμης επιλογής ισχύει:

$$R(T) \rightarrow \infty \text{ για } T \rightarrow \infty \text{ , με πιθανότητα 1} \tag{2.1}$$

Απόδειξη: Θέτουμε $\mu' = min\{\mu_1, \mu_2, ..., \mu_k\}$ και a', μια απόφαση που οδηγεί σε μέση αμοιβή μ'. Η πολιτική άπειρης εξερεύνησης εγγυάται την ύπαρξη δεικτών $t_1, t_2, ... \subseteq \{1, 2, ...\}$ τέτοιους ώστε $a_{t_j} = a' \neq a^* \ \forall \ j \in \mathbb{N}$ με πιθανότητα 1. Ισοδύναμα με πιθανότητα 1, θα ισχύει ότι $\mu_{t_j} = \mu' \neq \mu^* \ \forall \ j \in \mathbb{N}$ και ειδικότερα $\mu_{t_j} < \mu^* \ \forall \ j \in \mathbb{N}$ αφού η μ^* είναι η μέγιστη. Θέτουμε με T_1 το σύνολο με όλους τους δείκτες με την προηγούμενη ιδιότητα. Τότε με πιθανότητα 1 έχουμε:

$$R(T) = \sum_{t=1}^{T} (\mu^* - Q(A_t))$$

άρα

$$\lim_{T\to\infty} R(T) = \sum_{t=1}^{\infty} (\mu^* - Q(A_t)) = \sum_{t\in T_1} (\mu^* - Q(A_t)) + \sum_{t\in T_1^c} (\mu^* - Q(A_t))$$

Για το πρώτο άθροισμα έχουμε ότι $\forall \ t \ \in T_1 \ : \ Q(A_t) = \mu'$ και επομένως $\mu^* - Q(A_t) = \mu^* - \mu' > 0$. Για το δεύτερο άθροισμα θα ισχύει ότι $\forall \ t \ \in T_2 \ : \ Q(A_t) \leq \mu^*$ και επομένως $\mu^* - Q(A_t) \geq 0$. Συνεπώς,

$$\lim_{T\to\infty} R(T) = \sum_{t\in T_1} (\mu^* - \mu') + \sum_{t\in T_1^c} (\mu^* - Q(A_t)) \geq \sum_{t\in T_1} (\mu^* - \mu')$$

Επειδή η ποσότητα $\mu^* - \mu'$ είναι σταθερά και το T_1 άπειρα αριθμήσιμο έχουμε,

$$\sum_{t\in T_1} (\mu^* - \mu') = +\infty$$

Τελικά,

$\lim_{T\to\infty} R(T) = +\infty$, με πιθανότητα 1. ■

Η παραπάνω Πρόταση μας λέει διαισθητικά πως είναι αναπόφευκτο η συνολική απώλεια που έχουμε σε βάθος χρόνου να είναι άπειρη[6]. Πρακτικά όμως αυτό δεν αποτελεί πρόβλημα. Ρεαλιστικά σε ένα πρόβλημα MAB θα θέλαμε να προσεγγίσουμε την βέλτιστη λύση σε πεπερασμένο χρόνο και όσο το δυνατόν «ταχύτερα» γίνεται. Είναι φανερό λοιπόν ότι σε ένα πρόβλημα MAB επιθυμητή ιδιότητα είναι η εξής:

$$\lim_{T\to\infty} \frac{R(T)}{T} \to 0 \text{ , με πιθανότητα 1.}$$

ή με άλλα λόγια η $R(T)$ να είναι **υπογραμμική** με πιθανότητα 1. Η παραπάνω ιδιότητα μας εξασφαλίζει ότι σε βάθος χρόνου δεν υπάρχουν μεγάλες αποκλίσεις της μέσης αμοιβής που λαμβάνει το υποκείμενο σε κάθε βήμα από τη μέγιστη δυνατή.

Το επόμενο ερώτημα που γεννάται φυσιολογικά είναι κατά πόσο υπάρχουν πολιτικές για τις οποίες η συνολική απώλεια $R(T)$ μπορεί να αποδώσει ακόμα καλύτερα από άποψη ταχύτητας σύγκλισης, η με άλλα λόγια αν υπάρχουν πολιτικές για τις οποίες:

$$\lim_{T\to\infty} \frac{R(T)}{T^a} \to 0 \text{ , με πιθανότητα 1 και } \alpha < 1 \text{ αυθαίρετα μικρό.}$$

Η απάντηση σε αυτό το ερώτημα ήρθε από τους Lai και Robbins και είναι **αρνητική**.

Θεώρημα 2.1 (Θεώρημα των Lai-Robbins) Σε ένα πρόβλημα MAB χωρίς κάποια πρότερη γνώση του Περιβάλλοντος η μέση τιμή της συνολική απώλειας $R(T)$ είναι ασυμπτωτικά τουλάχιστον **λογαριθμική** κάτω από οποιαδήποτε πολιτική άπειρης εξερεύνησης π στο πλήθος των γύρων T. Πιο συγκεκριμένα, ισχύει ότι:

$$\lim_{T\to\infty} \frac{E_\pi\left[R(T)\right]}{\log T} \geq \sum_{a|\Delta_a>0} \frac{1}{\Delta_a} \tag{2.2}$$

όπου, $\Delta_a := \mu^* - \mu_a$, το **χάσμα υποβελτιστότητας** μια απόφασης a.

Απόδειξη: Η απόδειξη παραλείπεται καθώς είναι εκτός πλαισίου της παρούσας Εργασίας. ■

Συνοψίζοντας τα παραπάνω αποτελέσματα τονίζουμε πως στην προσπάθεια εξεύρεσης πολιτικών που προσεγγίζουν τη βέλτιστη πρέπει να λαμβάνονται υπόψιν τα ακόλουθα:

- Η συνάρτηση απώλειας $R(T)$ αποτελεί βασικό μέτρο απόδοσης οποιασδήποτε πολιτικής καθώς μας πληροφορεί για την ταχύτητα προσέγγισης της βέλτιστης λύσης. Παρά το γεγονός ότι αυξάνεται απεριόριστα σε άπειρο ορίζοντα όσο πιο «μικρή» είναι ασυμπτωτικά σαν συνάρτηση τόσο λιγότερη μέση αμοιβή χάνεται σε κάθε βήμα.

[6] Σε πολιτικές άπειρης εξερεύνησης όλων των διαθέσιμων επιλογών.

- Μια επιθυμητή ιδιότητα συνεπώς της $R(T)$ είναι η υπογραμμικότητα εφόσον η τελευταία εξασφαλίζει ότι σε βάθος χρόνου χάνεται λιγότερη μέση αμοιβή σε κάθε βήμα.

- Η $R(T)$ παρ' όλα αυτά είναι ασυμπτωτικά τουλάχιστον λογαριθμική σε πολιτικές άπειρης εξερεύνησης.

Οριακή Πιθανότητα λήψης της Βέλτιστης Απόφασης p_l

Είδαμε πως ο ρεαλιστικός στόχος του υποκειμένου είναι η εξεύρεση πολιτικών για τις οποίες ισχύει:

$$\lim_{t \to +\infty} P(A_t = a^*) \approx 1$$

Όταν το παραπάνω όριο υπάρχει θέτουμε

$$p_l := \lim_{t \to +\infty} P(A_t = a^*) \tag{2.3}$$

το οποίο ταυτίζεται με την **οριακή πιθανότητα λήψης της βέλτιστης απόφασης** a^* η ισοδύναμα με το **μακροπρόθεσμο ποσοστό του χρόνου που λαμβάνεται η βέλτιστη απόφαση** a^*. Μια Πολιτική π, θα καλείται **ασυμπτωτικά βέλτιστη** αν και μόνο αν:

$$p_l = \lim_{t \to +\infty} P(A_t = a^*) = 1 \tag{2.4}$$

Ο Δειγματικός Μέσος των αμοιβών $\overline{X_T}$

Ο δειγματικός μέσος των αμοιβών $\overline{X_T}$ αποτελεί τον γνωστό μέσο όρο των αμοιβών που έχουν συλλεχθεί έως και το βήμα T :

$$\overline{X_T} = \frac{1}{T} \sum_{t=1}^{T} X_t \tag{2.5}$$

Μια πολιτική που σε βάθος χρόνου θα εξασφάλιζε ότι ο $\overline{X_T}$ είναι «κοντά» στο μ^* με μεγάλη πιθανότητα θα σήμαινε ισοδύναμα ότι η συγκεκριμένη πολιτική οδηγεί σε βέλτιστες αποφάσεις σε βάθος χρόνου, που είναι και το επιθυμητό σενάριο. Έτσι για τον δειγματικό μέσο ιδανικά θα θέλαμε:

$$\overline{X_T} \xrightarrow{p} \mu^*$$

όπου $\xrightarrow{p}$, η γνωστή, κατά πιθανότητα σύγκλιση.

2.4 Ο Αλγόριθμος greedy

Εκτίμηση των $Q(a)$

Τα παραδείγματα εκπαίδευσης στο ειδικό πλαίσιο των MAB'S αποτελούν οι αμοιβές του περιβάλλοντος των οποίων οι κατανομές εξαρτώνται αποκλειστικά από την επιλογή που κάνει σε κάθε γύρο και όχι από όλο το υπάρχον ιστορικό. Το υποκείμενο **δεν** γνωρίζει εκ των προτέρων ποια απόφαση θα το οδηγήσει στη μεγαλύτερη μέση αμοιβή μ^*. Παρ΄ όλα αυτά μπορεί να **εκτιμήσει** τη μέση αμοιβή κάθε απόφασης $a \in A$ έως και τον γύρο t ως εξής:

$$\hat{Q}_t(a) = \frac{\sum_{i=1}^{t} X_i \cdot \mathbb{1}_{\{A_i=a\}}}{\sum_{i=1}^{t} \mathbb{1}_{\{A_i=a\}}} \tag{2.6}$$

όπου η δείκτρια $\mathbb{1}_{\{A_i=a\}}$ είναι τυχαία μεταβλητή για την οποία ισχύει:

$$\mathbb{1}_{\{A_i=a\}} = \begin{cases} 1, \text{ αν } A_i = a \\ 0, \text{ αν } A_i \neq a \end{cases} \tag{2.7}$$

Σε περίπτωση που ο παρονομαστής είναι 0 και άρα δεν έχει επιλεγεί μέχρι και τον γύρο t η απόφαση a τότε θέτουμε την $\hat{Q}_t(a)$ ως μια αυθαίρετη σταθερά όπως το 0. Καθώς ο παρονομαστής της εκτιμήτριας $\hat{Q}_t(a)$ πηγαίνει στο $+\infty$ τότε από τον **Ισχυρό Νόμο των Μεγάλων Αριθμών**:

$$\hat{Q}_t(a) \to Q(a) = \mu_a \text{ , με πιθανότητα } 1 \tag{2.8}$$

Συνεπώς για «μεγάλο» t οι εκτιμήτριες $\hat{Q}_t(a)$ θα είναι κοντά στις μέσες τιμές μ_a αρκεί να υπάρχει κατάλληλα μεγάλο δείγμα από όλες τις δυνατές αποφάσεις του υποκειμένου.

Η Πολιτική greedy

Ο πιο απλός κανόνας επιλογής αποφάσεων είναι να επιλέγεται συνεχώς η επιλογή που έχει τη μεγαλύτερη εκτίμηση $\hat{Q}_t(a)$. Η παραπάνω επιλογή θα ονομάζεται **πλεονεκτική** ή **greedy** καθώς εκμεταλλεύεται πλήρως την υπάρχουσα γνώση για εξασφάλιση υψηλής μέσης αμοιβής. Φορμαλιστικά μια επιλογή θα ονομάζεται πλεονεκτική στον γύρο t εαν:

$$A_t = argmax_a\{\hat{Q}_t(a)\} \tag{2.9}$$

Έτσι, μια πολιτική θα ονομάζεται **πλεονεκτική** ή **greedy** αν για ένα προκαθορισμένο αριθμό γύρων T_0 εξερευνά όλες τις δυνατές επιλογές (συνήθως ομοιόμορφα) και

εν συνεχεία επιλέγεται για όλους τους υπολειπόμενους γύρους η πλεονεκτική επιλογή του γύρου T_0. Η παραπάνω πολιτική εκμεταλλεύεται, από έναν προκαθορισμένο αριθμό γύρων και μετά, πάντα την πληροφορία που έχει συλλεχθεί κατά την «περίοδο εξερεύνησης» αδιαφορώντας για τις φαινομενικά υποδεέστερες δράσεις που στην πραγματικότητα όμως μπορεί να είναι καλύτερες. Στον Πίνακα 2.1 παρουσιάζεται η Πολιτική **greedy** σε μορφή αλγορίθμου.

Πίνακας 2.1 Ο αλγόριθμος **greedy**

---

1. Αρχικοποίησε $Q(a) \leftarrow 0$ για κάθε $a = 1, 2, ..., k$
2. (Περίοδος εξερεύνησης) Επίλεξε κάθε λαβή N φορές:

$$A_t \leftarrow 1, \text{ για } t = 1, 2, ..., N$$

$$A_t \leftarrow 2, \text{ για } t = N+1, N+2, ..., 2N$$

$$\dots$$

$$A_t \leftarrow k, \text{ για } t = (k-1)N+1, (k-1)N+2, ..., kN$$

3. Όρισε ως $\hat{a}$ τη λαβή με τη μεγαλύτερη εκτίμηση στους πρώτους kN γύρους (σπάζοντας τις ισοπαλίες αυθαίρετα):

$$\hat{a} \leftarrow argmax_a\{\hat{Q}_{kN}(a)\}$$

4.(Περίοδος εκμετάλλευσης) Επίλεξε την λαβή $\hat{a}$ για όλους τους εναπομείναντες γύρους:

$$A_t \leftarrow \hat{a}, \text{ για } t > kN$$

Στην Πρόταση που ακολουθεί παρουσιάζονται 3 βασικά αποτελέσματα της πλεονεκτικής πολιτικής ως προς τα μέτρα απόδοσης.

Πρόταση 2.2 Έστω ένα πρόβλημα **MAB** με k λαβές και η πολιτική αποφάσεων

greedy όπως διατυπώθηκε στον Πίνακα 2.1. Έστω επιπλέον q_i η πιθανότητα η λαβή i να έχει και τη βέλτιστη δειγματική αμοιβή στους πρώτους kN γύρους:

$$q_i := P\left(\hat{Q}_{kN}(i) = \max_a\{\hat{Q}_{kN}(a)\}\right) \tag{2.10}$$

Τέλος, ας υποθέσουμε, χωρίς βλάβη της γενικότητας, ότι $a^* = k$. Τότε ισχύουν τα ακόλουθα :

1. Για κάθε $T \geq kN + 1$:

$$R(T) = \begin{cases} N(k\mu^* - \sum_{i=1}^{k} \mu_i) \ \text{, με πιθανότητα } q_{a^*} \\ (\mu^* - \mu_1)T + N(k\mu_1 - \sum_{i=1}^{k} \mu_i), \text{ με πιθανότητα } q_1 \\ \vdots \\ (\mu^* - \mu_{k-1})T + N(k\mu_{k-1} - \sum_{i=1}^{k} \mu_i), \text{ με πιθανότητα } q_{k-1} \end{cases}$$

2. $$\lim_{t\to+\infty} P(A_t = a^*) = \begin{cases} 1 \ \text{, με πιθανότητα } q_{a^*} \\ 0 \ \text{, με πιθανότητα } 1 - q_{a^*} \end{cases}$$

3. $$\overline{X_T} \xrightarrow{p} \begin{cases} \mu^*, \text{ με πιθανότητα } q_{a^*} \\ \mu_1, \text{ με πιθανότητα } 1 - q_1 \\ \vdots \\ \mu_{k-1}, \text{ με πιθανότητα } 1 - q_{k-1} \end{cases}$$

Απόδειξη:

1. Σύμφωνα με την πολιτική **greedy** του πίνακα 2.1 το υποκείμενο δοκιμάζει κάθε απόφαση για N γύρους κατά την περίοδο εξερεύνησης. Συνεπώς μέχρι και τον γύρο kN έχουν συλλεχθεί από αυτό τα στατιστικά: $\hat{Q}_{kN}(1), \hat{Q}_{kN}(2), ..., \hat{Q}_{kN}(k)$. Τα στατιστικά αυτά πριν την εμφάνιση τους, αποτελούν τυχαίες μεταβλητές εφόσον είναι μέσοι όροι αμοιβών που φανερώνονται από το περιβάλλον και συνεπώς η διάταξη τους είναι στοχαστική. Έστω $\hat{Q}_{kN}(a^*)$ η δειγματική εκτίμηση του μ^*. Τότε αν $\hat{Q}_{kN}(a^*) = \max_a\{\hat{Q}_{kN}(a)\}$ τότε αυτό θα εξασφαλίσει ότι για όλους τους υπολειπόμενους γύρους $T \geq kN+1$ θα επιλέγεται η βέλτιστη απόφαση a^* δηλαδή:

$$A_t = a^* \text{ για κάθε T } \geq kN + 1$$

και άρα

$$Q(A_t) = \mu^* \text{ για κάθε T } \geq kN + 1$$

Συνεπώς με πιθανότητα $q_{a^*} = P\left(\hat{Q}_{kN}(a^*) = \max_a\{\hat{Q}_{kN}(a)\}\right)$ και για κάθε T $\geq kN+1$ έχουμε:

$$\begin{aligned} R(T) &= T\mu^* - \sum_{t=1}^{T} Q(A_t) \\ &= T\mu^* - \sum_{t=1}^{kN} Q(A_t) - \sum_{t=kN+1}^{T} Q(A_t) \\ &= T\mu^* - N\sum_{i=1}^{k}\mu_i - (T-kN)\mu^* \\ &= T\mu^* - N\sum_{i=1}^{k}\mu_i - T\mu^* + kN\mu^* \\ &= N(k\mu^* - \sum_{i=1}^{k}\mu_i). \end{aligned}$$

Έστω τώρα ότι επιλέγεται η λαβή 1, το οποίο συμβαίνει με πιθανότητα $q_1 = P\left(\hat{Q}_{kN}(1) = \max_a\{\hat{Q}_{kN}(a)\}\right)$. Τότε :

$$A_t = 1 \text{ για κάθε T } \geq kN+1$$

και άρα

$$Q(A_t) = \mu_1 \leq \mu^* \text{ για κάθε T } \geq kN+1$$

Έτσι με πιθανότητα q_1 και για κάθε $T \geq kN+1$ έχουμε:

$$\begin{aligned} R(T) &= T\mu^* - \sum_{t=1}^{T} Q(A_t) \\ &= T\mu^* - \sum_{t=1}^{kN} Q(A_t) - \sum_{t=kN+1}^{T} Q(A_t) \\ &= T\mu^* - N\sum_{i=1}^{k}\mu_i - (T-kN)\mu_1 \\ &= T\mu^* - N\sum_{i=1}^{k}\mu_i - T\mu_1 + kN\mu_1 \\ &= (\mu^* - \mu_1)T + N(k\mu_1 - \sum_{i=1}^{k}\mu_i). \end{aligned}$$

Επαναλαμβάνοντας την ίδια διαδικασία για τις λαβές $2, 3, \dots k-1$ καταλήγουμε στο ζητούμενο.

2. Προκύπτει άμεσα:

   - Με πιθανότητα q_{a^*} θα ισχύει:

     $$P(A_t = a^*) = 1 \text{ για κάθε } t \geq kN + 1$$

     και άρα

     $$\lim_{t \to +\infty} P(A_t = a^*) = 1$$

   - Με πιθανότητα $1 - q_{a^*}$ θα υπάρχει $a' \neq a^*$ τέτοιο ώστε:

     $$P(A_t = a') = 1 \text{ για κάθε } t \geq kN + 1$$

     και άρα

     $$P(A_t = a^*) = 0 \text{ για κάθε } t \geq kN + 1$$

     και επομένως,

     $$\lim_{t \to +\infty} P(A_t = a^*) = 0$$

3. Έστω $\overline{X_T}$, ο δειγματικός μέσος των αμοιβών για $T \geq kN+1$. Τότε με πιθανότητα q_{a^*} θα έχουμε:

   $$\begin{aligned} \overline{X_T} &= \frac{X_1 + X_2 + \cdots + X_T}{T} \\ &= \frac{\sum_{t=1}^{kN} X_t}{T} + \frac{\sum_{t=kN+1}^{T} X_t}{T} \end{aligned}$$

   όπου,

   $$X_{kN+1}, X_{kN+2}, \cdots, X_T \sim P_{a^*}$$

   Έστω τώρα ότι έχουμε στην διάθεση μας τις επιπλέον ανεξάρτητες και ισόνομες παρατηρήσεις: $X'_1, X'_2, ..., X'_{kN} \sim P_{a^*}$. Θέτουμε επιπλέον $X'_t = X_t \ \forall \ t = kN + 1, ..., T$. Τότε ο δειγματικός μέσος των αμοιβών γράφεται ισοδύναμα:

   $$\overline{X_T} = \frac{\sum_{t=1}^{kN}(X_t - X'_t)}{T} + \frac{\sum_{t=1}^{T} X'_t}{T}$$

   Θέτοντας,

   $$S_1 := \sum_{t=1}^{kN}(X_t - X'_t) \text{ και } S_2 := \sum_{t=1}^{T} X'_t$$

   έχουμε:

- $\frac{S_1}{T} \xrightarrow{p} 0$ αφού $\sum_{t=1}^{kN}(X_t - X'_t) < +\infty$
- $\frac{S_2}{T} \xrightarrow{p} \mu^*$ από τον **Ασθενή Νόμο των Μεγάλων Αριθμών**[7]

Συνεπώς από το **Θεώρημα του Slutsky** έπεται ότι:

$$\overline{X_T} \xrightarrow{p} 0 + \mu^* = \mu^*$$

Τέλος, οι περιπτώσεις ότι με πιθανότητα $1 - q_i$, $i \in A\backslash\{a^*\}$:

$$\overline{X_T} \xrightarrow{p} \mu_i$$

αποδεικνύονται όμοια και η απόδειξη φτάνει στο τέλος της. ■

Παρατηρήσεις

1. Η πιθανότητα εντοπισμού της βέλτιστης δράσης κατά την περίοδο εξερεύνησης $q_{a^*} = P\left(\hat{Q}_{kN}(a^*) = \max_a\{\hat{Q}_{kN}(a)\}\right)$ γενικά ανήκει στο διάστημα $[0, 1]$ και η τιμή της εξαρτάται από τις κατανομές $P_1, P_2, ..., P_k$.

2. Παρατηρούμε πως **με πιθανότητα** q_{a^*} ο αλγόριθμος greedy αποδίδει εξαιρετικά ως προς και τα 3 μέτρα απόδοσης:

- Η συνάρτηση συνολικής απώλειας $R(T) < \infty$
- $p_l = \lim_{t\to+\infty} P(A_t = a^*) = 1$
- $\overline{X_t} \xrightarrow{p} \mu^*$

Όμως, **με πιθανότητα** $1 - q_{a^*}$ οι αποδόσεις του αλγορίθμου είναι απογοητευτικές:

- Η συνάρτηση συνολικής απώλειας $R(T)$ είναι **γραμμική**
- $p_l = \lim_{t\to+\infty} P(A_t = a^*) = 0$
- $\overline{X_t} \xrightarrow{p} \mu'$, όπου $\mu' \leq \mu^*$

Τα παραπάνω συμφωνούν απόλυτα με τη διαίσθηση μας. Αν το υποκείμενο εντοπίσει τη βέλτιστη δράση κατά την περίοδο εξερεύνησης, τότε για όλους τους υπολειπόμενους γύρους θα επιλέγει τη βέλτιστη απόφαση a^* και συνεπώς από το γύρο $kN + 1$ δεν θα χάνεται μέση αμοιβή στο χρόνο. Σε περίπτωση που όμως «αστοχήσει» οι συνέπειες θα είναι καταστροφικές εφόσον θα επιλέγεται από τον γύρο $kN + 1$ και έπειτα μια απόφαση που δεν είναι βέλτιστη.[8]

3. Το γεγονός ότι τελικά η απόδοση της πολιτικής greedy εξαρτάται από την πιθανότητα εύρεσης της βέλτιστης δράσης, q_{a^*}, την κάνει αρκετά αναποτελεσματική και γι΄ αυτό το λόγο δεν χρησιμοποιείται ποτέ στην πράξη, παρά μόνο παραλλαγές της.

[7] Ο Ασθενής Νόμο των Μεγάλων Αριθμών εφαρμόζεται άμεσα εφόσον η ακολουθία $\{X'_t\}_{t=1}^{+\infty}$ είναι ανεξάρτητη και ισόνομη και $E[X'_1] = \mu^* < +\infty$.

[8] Εκτός από την περίπτωση που $\mu' = \mu^*$.

2.5 Ο Αλγόριθμος ε-greedy

Το πρόβλημα με την πολιτική **greedy** ήταν ότι ήταν **υπερβολικά σίγουρη** για την εκτίμηση του μ^* καθιστώντας την εξερεύνηση αναποτελεσματική. Μια απλή εναλλακτική, είναι το υποκείμενο να συμπεριφέρεται **πλεονεκτικά** σχεδόν σε κάθε γύρο αλλά μια στο τόσο, ας πούμε με μικρή πιθανότητα ε, να επιλέγει τυχαία, αλλά **ομοιόμορφα**, μια απόφαση μεταξύ όλων των δυνατών, και ανεξάρτητα από τις εκτιμήσεις $\hat{Q}_t(a)$. Το πλεονέκτημα της παραπάνω εναλλακτικής είναι ότι καθώς το πλήθος των γύρων αυξάνει απεριόριστα τότε, με πιθανότητα 1, κάθε απόφαση θα παρθεί για άπειρο πλήθος φορών και κατά συνέπεια όλες οι εκτιμήσεις $\hat{Q}_t(a)$ θα συγκλίνουν στα $Q(a)$.

Πριν διατυπώσουμε τον αλγόριθμο ε-greedy θα δώσουμε έναν αναδρομικό τύπο υπολογισμού μιας εκτίμησης σε σχέση με την προηγούμενη. Η παραπάνω είναι ιδιαίτερα χρήσιμη καθώς κατά την εκτέλεση της απαιτεί λιγότερη μνήμη σε σχέση με το να διατηρούσαμε αρχείο για όλες τις αμοιβές των προηγούμενων γύρων. Για να απλοποιήσουμε το συμβολισμό παρακάτω, με $T-1$ δηλώνουμε το πλήθος των φορών που έχει ληφθεί ως τώρα μια απόφαση a και με $X_1, X_2, ..., X_{T-1}$ τις αντίστοιχες αμοιβές της a . Αν τέλος απαλλάξουμε και την εξάρτηση του $\hat{Q}_t(a)$ από το a θα έχουμε:

$$\hat{Q}_{T-1} = \frac{X_1 + X_2 + \cdots + X_{T-1}}{T-1} \tag{2.11}$$

Έστω τώρα ότι η ίδια απόφαση λαμβάνεται ακόμα μια φορά και X_T η συνακόλουθη αμοιβή. Τότε:

$$\begin{aligned}
\hat{Q}_T &= \frac{X_1 + X_2 + \cdots + X_{T-1} + X_T}{T} \\
&= \frac{1}{T}\left(X_T + \sum_{t=1}^{T-1} X_t\right) \\
&= \frac{1}{T}\left(X_T + (T-1)\frac{1}{T-1}\sum_{t=1}^{T-1} X_t\right) \\
&= \frac{1}{T}\left(X_T + (T-1)\hat{Q}_{T-1}\right) \\
&= \frac{1}{T}\left(X_T + T\hat{Q}_{T-1} - \hat{Q}_{T-1}\right) \\
&= \hat{Q}_{T-1} + \frac{1}{T}\left(X_T - \hat{Q}_{T-1}\right)
\end{aligned} \tag{2.12}$$

Παρατηρήσεις

1. Ο κανόνας ανανέωσης (2.12) απαιτεί μνήμη μόνο για τα $\hat{Q}_{T-1}$ (παλαιά εκτίμηση), T (πλήθος γύρων που έχει ληφθεί η απόφαση) και X_T (νέα αμοιβή)

2. Ο παραπάνω κανόνας ανανέωσης μιας εκτίμησης είναι της εξής γενικής μορφής που συναντάται συχνά στα πλαίσια της Ενισχυτικής Μάθησης:

$$\text{Νέα Εκτίμηση} \leftarrow \text{Παλαιά Εκτίμηση} + \text{Μέγεθος Βήματος}\,[\text{Στόχος} - \text{Παλαιά Εκτίμηση}]$$

Η έκφραση [Στόχος – Παλαιά Εκτίμηση] εκφράζει ένα σφάλμα της εκτίμησης σε σχέση με ένα «Στόχο». Το πρόσημο της έκφρασης υποδηλώνει της επιθυμητή κατεύθυνση κίνησης της νέας εκτίμησης: Αν είναι θετική, αυτό σημαίνει ότι υπάρχουν ενδείξεις ότι η τελευταία πρέπει να αυξηθεί ενώ σε αντίθετη περίπτωση πρέπει να υποστεί μείωση. Το Μέγεθος Βήματος καθορίζει σε τι βαθμό πρέπει να λαμβάνεται υπόψιν ως προς το χρόνο το σφάλμα της εκτίμησης. Στο παράδειγμα μας θεωρούμε κάθε νέα αμοιβή που μας φανερώνεται ότι είναι «ίσης αξίας» με κάθε προηγούμενη και άρα είναι λογικό το Μέγεθος Βήματος να ταυτίζεται με $\frac{1}{T}$.

Στον πίνακα που ακολουθεί παρουσιάζεται ο αλγόριθμος **ε-greedy**. Με $Bandit(A_t)$ συμβολίζουμε μια τυχαία αμοιβή προερχόμενη από την λαβή A_t, και με $DUnif(1, 2, \ldots, k)$ μια τυχαία μεταβλητή προερχόμενη από την διακριτή ομοιόμορφη κατανομή με παραμέτρους $1, 2, \ldots, k$.

Πίνακας 2.2 Ο αλγόριθμος **ε-greedy**

---

1. Αρχικοποίησε για κάθε $a = 1, 2, ..., k$

$$\hat{Q}_0(a) \leftarrow 0$$
$$N_0(a) \leftarrow 0$$

2. Για κάθε $t = 1, 2, ...$:

$$A_t \leftarrow \begin{cases} argmax_a\{\hat{Q}_{t-1}(a)\}, & \text{με πιθανότητα 1-ε (σπάζοντας τις ισοπαλίες αυθαίρετα)} \\ DUnif(1, 2, ..., k), & \text{με πιθανότητα ε} \end{cases}$$

$$X_t \leftarrow \text{Bandit}(A_t)$$
$$N_t(A_t) \leftarrow N_{t-1}(A_t) + 1$$
$$\hat{Q}_t \leftarrow \hat{Q}_{t-1} + \frac{1}{N_t(A_t)}\left(X_t - \hat{Q}_{t-1}\right)$$

Στην Πρόταση που ακολουθεί παρουσιάζονται 3 βασικά αποτελέσματα της πολιτικής ε-greedy ως προς τα μέτρα απόδοσης.

Πρόταση 2.3 Έστω ένα πρόβλημα **MAB** με k λαβές και η πολιτική αποφάσεων **ε-greedy** όπως διατυπώθηκε στον Πίνακα 2.2 ($0 < \varepsilon < 1$). Τότε ισχύουν τα ακόλουθα:

1. Η $R(T)$ είναι ασυμπτωτικά **γραμμική** με πιθανότητα 1.

2. $$\lim_{t\to+\infty} P(A_t = a^*) = 1 - \varepsilon + \frac{\varepsilon}{k}$$

3. $$\overline{X_T} \xrightarrow{p} (1-\varepsilon)\cdot\mu^* + \frac{\varepsilon}{k}\sum_{i=1}^{k}\mu_i$$

Απόδειξη: Η απόδειξη παραλείπεται. ■

Ο αλγόριθμος **ε-greedy** είναι μια σαφώς βελτιωμένη έκδοση του αλγορίθμου **greedy** από την άποψη ότι εξερευνεί όλες τις διαθέσιμες επιλογές για άπειρο αριθμό γύρων και έτσι η σύγκλιση των μέτρων απόδοσης δεν εξαρτάται από κάποιο στατιστικό μέτρο που έχει συλλεχθεί σε πεπερασμένα βήματα. Επίσης, θέτοντας το ε αυθαίρετα κοντά στο 0 μπορούμε να εξασφαλίσουμε ότι ασυμπτωτικά το υποκείμενο θα λαμβάνει τη βέλτιστη απόφαση σε ποσοστό πολύ κοντά στο 1 (ισοδύναμα και ο δειγματικός μέσος θα είναι πολύ κοντά στο μ^*). Βέβαια το τελευταίο, λίγη ή καθόλου σημασία μπορεί να έχει σε μια εφαρμογή στην οποία το επιθυμητό αποτέλεσμα είναι η εξασφάλιση υψηλής μέσης αμοιβής σε σχετικά γρήγορο χρόνο. Πράγματι, αν θέσουμε το ε πολύ κοντά στο 0 αυτό θα σημαίνει ότι το υποκείμενο με τεράστια πιθανότητα θα δρα πλεονεκτικά και έτσι θα επιλέγει τη δράση με το μεγαλύτερο δειγματικό μέσο. Όμως, στα αρχικά βήματα της διαδικασίας το υποκείμενο έχει πολύ λίγη πληροφορία για τις εκτιμήσεις που έχει κάνει και κατά συνέπεια θα αργήσει ενδεχομένως «αρκετά» να ανακαλύψει τη βέλτιστη απόφαση. Η αργή ταχύτητα σύγκλισης των 2 τελευταίων μέτρων απόδοσης του αλγορίθμου αποτυπώνεται και στη συνάρτηση συνολικής απώλειας που παραμένει γραμμική. Ο λόγος που ουσιαστικά οφείλεται αυτό είναι ότι το ε όσο κοντά και αν είναι στο 0, παραμένει σταθερό και έτσι κάθε απόφαση λαμβάνεται με σταθερό ποσοστό σε βάθος χρόνου.

2.6 Ο Αλγόριθμος UCB

Ένα από τα βασικότερα μειονεκτήματα του αλγορίθμου ε-greedy είναι ότι η εξερεύνηση πραγματοποιείται ομοιόμορφα ακόμα και σε αποφάσεις που ενδέχεται να οδηγούν επανειλημμένα σε χαμηλές αμοιβές, μη διαχωρίζοντας τις με άλλες για τις οποίες υπάρχει η δυναμική να είναι βέλτιστες. Για να αποφευχθεί η παραπάνω αναποτελεσματική εξερεύνηση, μια προσέγγιση είναι, το υποκείμενο να επιλέγει πάντα με βάση ένα άνω φράγμα της μέσης τιμής $Q(a)$ αντί της άμεσης εκτίμησης $\hat{Q}(a)$. Διαισθητικά, αν έχουμε στη διάθεση μας ένα άνω φράγμα για τη μέση τιμή κάθε λαβής το οποίο ταυτόχρονα:

- μικραίνει όσο επιλέγεται μια συγκεκριμένη απόφαση a, και άρα πλησιάζει τη πραγματική αξία της a και
- μεγαλώνει όσο δεν επιλέγεται η a αλλά με σχετικά **αργό ρυθμό**

τότε το δίπολο εξερεύνησης εκμετάλλευσης επιβραβεύει τόσο εκείνες τις λαβές οι οποίες αποδίδουν μεγάλες μέσες αμοιβές αλλά ταυτόχρονα και εκείνες οι οποίες έχουν επιλεγεί λιγότερο και άρα χαρακτηρίζονται από περισσότερη αβεβαιότητα. Τελικά, αν μια λαβή είναι επανειλημμένα «κακή», δηλαδή αποφέρει μη ικανοποιητικές αμοιβές και ταυτόχρονα έχει επιλεγεί αρκετές φορές δεν θα υπάρχει κίνητρο στο να επιλέγεται παρά μόνο όταν το άνω φράγμα της ξεπεράσει όλες τις υπόλοιπες λόγω του ότι αναπόφευκτα θα αυξάνει και κάποια στιγμή θα είναι μεγαλύτερο από τα υπόλοιπα. Η διαισθητική αυτή προσέγγιση θα γίνει περισσότερο αντιληπτή αφού διατυπώσουμε την έκφραση για το άνω φράγμα.

Οι **Αλγόριθμοι τύπου UCB (Upper Confidence Bound)**, που μια ειδική κατηγορία των οποίων θα διατυπωθεί στη συνέχεια, ουσιαστικά μετράνε τη δυναμική που έχει μια απόφαση στο να είναι βέλτιστη με ένα άνω φράγμα της αξίας της αμοιβής, $\hat{Q}_t(a) + \hat{U}_t(a)$, έτσι ώστε η πραγματική αξία να είναι κάτω από αυτό το άνω φράγμα με μεγάλη πιθανότητα:

$$Q(a) \leq \hat{Q}_t(a) + \hat{U}_t(a) \text{ , με πιθανότητα κοντά στο 1}$$

Ασφαλώς, η ποσότητα $\hat{U}_t(a)$ θα είναι συνάρτηση του αριθμού των δοκιμών $N_t(a)$ και καθώς το τελευταίο αυξάνει θα πρέπει το άνω φράγμα να μειώνεται. Κατά τον αλγόριθμο UCB πλεονεκτική θα είναι εκείνη η απόφαση για την οποία:

$$A_t^{UCB} = argmax_a\{\hat{Q}_t(a) + \hat{U}_t(a)\} \tag{2.13}$$

Άλλο ένα ερώτημα που πρέπει να απαντήσουμε είναι πως μπορούμε να οδηγηθούμε σε κάποια έκφραση για το $\hat{U}_t(a)$ δεδομένου ότι δεν διαθέτουμε κάποια πληροφορία για τις κατανομές των αμοιβών. Σε αυτό το ερώτημα θα δώσει απάντηση η ανισότητα του Hoeffding της οποίας ο μόνος περιορισμός είναι ότι οι αμοιβές πρέπει να λαμβάνουν τιμές στο $[0, 1]$.

Θεώρημα 2.2 (Ανισότητα του Hoeffding) Έστω $X_1, X_2, ..., X_t$ ανεξάρτητες και ισόνομες τυχαίες μεταβλητές φραγμένες στο [0,1] και $\overline{X_t}$ ο δειγματικός μέσος:

$$\overline{X_t} = \frac{1}{t}\sum_{i=1}^{t} X_i$$

Τότε,

$$P\left(E[X] > \overline{X_t} + u\right) \leq e^{-2tu^2} \tag{2.14}$$

Απόδειξη: Η απόδειξη παραλείπεται. ■

Επανερχόμενοι τώρα στο πλαίσιο των MAB's αν $X_1, X_2, ..., X_{N_t(a)}$ ανεξάρτητες και ισόνομες τυχαίες μεταβλητές από μια λαβή $a \in A$, θέτοντας $u = \hat{U}_t(a)$ η ανισότητα παίρνει την μορφή:

$$P\left(Q(a) > \hat{Q}_t(a) + \hat{U}_t(a)\right) \leq e^{-2N_t(a)\hat{U}_t(a)^2} \tag{2.15}$$

Αν επιπλέον θέσουμε,

$$p := e^{-2N_t(a)\hat{U}_t(a)^2} \tag{2.16}$$

και λύνοντας ως προς $\hat{U}_t(a)$ παίρνουμε την έκφραση,

$$\hat{U}_t(a) = \sqrt{\frac{-lnp}{2N_t(a)}} \tag{2.17}$$

Για τις διάφορες τιμές p υπάρχουν διάφοροι αλγόριθμοι που μπορούν να προκύψουν με βάση το παραπάνω κατώφλι. Ένας από τους δημοφιλέστερους αποτελεί ο **αλγόριθμος UCB1** των Auer, Cesa-Bianchi και Fisher ο οποίος χρησιμοποιεί την τιμή $p = t^{-4}$. Έτσι, κατά τον συγκεκριμένο αλγόριθμο η τιμή για το άνω φράγμα, παίρνει τη μορφή:

$$\hat{Q}_t(a) + \hat{U}_t(a) = \hat{Q}_t(a) + \sqrt{\frac{2lnt}{N_t(a)}} \tag{2.18}$$

Ας επανέλθουμε τώρα στη συζήτηση περί αποτελεσματικότερης εξερεύνησης του άνω φράγματος. Ο όρος $\hat{Q}_t(a)$ εκφράζει την άμεση εκτίμηση που έχουμε για την αξία της απόφασης a. Όσο πιο μεγάλη τιμή έχει, τόσο πιο πιθανό είναι να την επιλέξουμε. Από την άλλη μεριά ο όρος $\hat{U}_t(a)$ είναι ο όρος που εκφράζει την **αβεβαιότητα**. Κάθε φορά που επιλέγεται η a ο παρονομαστής του κλάσματος μεγαλώνει και έτσι ο όρος αβεβαιότητας μικραίνει. Επίσης κάθε φορά που η a **δεν** επιλέγεται, ο παρονομαστής παραμένει σταθερός αλλά επειδή το t εμφανίζεται στο αριθμητή, η εκτίμηση της αβεβαιότητας μεγαλώνει. Η παρουσία του φυσικού λογαρίθμου σημαίνει ότι η αύξηση γίνεται όλο και μικρότερη στο πέρας του χρόνου αλλά παραμένει μη φραγμένη ωθώντας όλες τις δράσεις να επιλεχθούν. Έτσι, δράσεις με χαμηλές εκτιμήσεις, ή δράσεις που έχουν επιλεγεί έως τώρα σχετικά συχνά θα επιλέγονται με φθίνουσα συχνότητα στο πέρας του χρόνου.

Πίνακας 2.3 Ο **αλγόριθμος UCB1**

1. Επίλεξε κάθε λαβή από μια φορά. Δηλαδή για κάθε $t = 1, 2, ..., k$ θέσε:

$$A_t \leftarrow t$$

$$X_t \leftarrow \text{Bandit}(A_t)$$

$$N_t(A_t) \leftarrow 1$$

$$\hat{Q}_t(A_t) \leftarrow X_t$$

$$\hat{U}_t(A_t) \leftarrow \sqrt{\frac{2lnt}{N_t(A_t)}}$$

2. Για τους υπολειπόμενους γύρους επίλεξε την λαβή με το μέγιστο άνω φράγμα. Δηλαδή για κάθε $t \geq k + 1$ θέσε:

$$A_t \leftarrow argmax_a\{\hat{Q}_{t-1}(a) + \hat{U}_{t-1}(a)\}$$

$$X_t \leftarrow \text{Bandit}(A_t)$$

$$N_t(A_t) \leftarrow N_{t-1}(A_t) + 1$$

$$\hat{Q}_t(A_t) \leftarrow \hat{Q}_{t-1} + \frac{1}{N_t(A_t)}\left(X_t - \hat{Q}_{t-1}\right)$$

$$\hat{U}_t(A_t) \leftarrow \sqrt{\frac{2lnt}{N_t(A_t)}}$$

Τέλος, για τον αλγόριθμο UCB1 ισχύει η ακόλουθη πρόταση:

Πρόταση 2.4 Έστω ένα πρόβλημα **MAB** με k λαβές και η πολιτική αποφάσεων **UCB1** όπως διατυπώθηκε στον Πίνακα 2.3 Τότε ισχύουν τα ακόλουθα:

1. Η $R(T)$ είναι ασυμπτωτικά **λογαριθμική** με πιθανότητα 1
2. $\lim_{t\to+\infty} P(A_t = a^*) = 1$
3. $\overline{X_t} \xrightarrow{p} \mu^*$

Απόδειξη: Η απόδειξη παραλείπεται. ■

Σύνοψη

Συγκρίνοντας τους 3 αλγορίθμους ως προς την οριακή συμπεριφορά τους μπορούμε με βεβαιότητα να πούμε ότι ο ucb1 είναι αποτελεσματικότερος ως προς τα και τα 3 μέτρα απόδοσης. Ο λόγος που συμβαίνει αυτό είναι ότι εξερευνεί με τρόπο «εξυπνότερο» από τον ε-greedy και επίσης δεν σταματάει την διαδικασία εξερεύνησης όπως ο Αλγόριθμος greedy. Προφανώς αναλόγως την εφαρμογή τους και το πλήθος των διαθέσιμων γύρων η επιλογή των παραμέτρων κάθε αλγορίθμου μπορεί να παίξει καθοριστικό ρόλο. Έτσι δεν αποκλείεται σε ορισμένες περιπτώσεις οποισδήποτε από του 3 να αποδώσει καλύτερα έναντι των άλλων 2, ειδικά αν μιλάμε για μικρό αριθμό στιγμιοτύπων ή **επεισοδίων**. Ο όρος επεισόδιο στην Ενισχυτική Μάθηση χρησιμοποιείται όταν αναφερόμαστε στην εξέλιξη μιας συγκεκριμένης αλληλεπίδρασης του υποκειμένου με το περιβάλλον του. Έτσι γίνεται αντιληπτό, πως οποιαδήποτε απόπειρα σύγκρισης, είναι εύλογο να γίνεται σε μεγάλο αριθμό επεισοδίων και κατά συνέπεια ως προς τις μέσες τιμές των μέτρων απόδοσης στην εξέλιξη του χρόνου.

2.7 Εφαρμογή Αλγορίθμων στην R με Κατανομές Βήτα

Το ενδιαφέρον με τους αλγορίθμους που αναπτύχθηκαν στις προηγούμενες ενότητες είναι ότι υποθέτουν ελάχιστη a-priori πληροφορία για τις κατανομές των αμοιβών του περιβάλλοντος (όπως ότι οι μέσες τιμές είναι πεπερασμένες και για τον ucb1 ότι είναι επιπλέον φραγμένες στο [0.1]). Αυτή είναι άλλωστε και η φιλοσοφία τους: να οδηγούμαστε σε προσεγγιστικά βέλτιστες πολιτικές ανεξαρτήτως κάποιας γνώσης για τη φύση των κατανομών. Παρ΄ όλα αυτά εάν θέλουμε εξακριβώσουμε τα θεωρητικά αποτελέσματα που παρουσιάστηκαν με αυτά που θα προκύψουν στα πλαίσια μιας εφαρμογής είναι αρκετά εύλογο να ξεκινήσουμε με υποθέσεις για τις κατανομές του περιβάλλοντος.

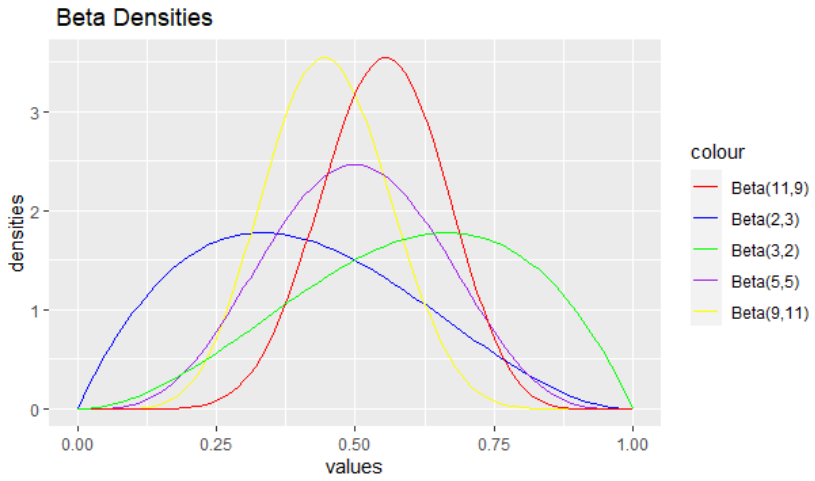
Σε αυτήν την ενότητα θα γίνει εφαρμογή των αλγορίθμων που περιγράφηκαν στην Γλώσσα Προγραμματισμού R με σκοπό την καλύτερη κατανόηση τους, τον σχολιασμό των μέτρων απόδοσης, καθώς και την σύγκριση τους ως προς διάφορα κριτήρια. Επιπλέον θα υποθέσουμε ότι όλες οι αμοιβές προέρχονται από κατανομές Βήτα.

Υλοποίηση Αλγορίθμων

Ας υποθέσουμε ότι ερχόμαστε αντιμέτωποι με ένα πρόβλημα MAB με $k = 5$ λαβές και επιπλέον ότι:

- $X|A_t = 1 \sim Beta(2, 3)$
- $X|A_t = 2 \sim Beta(9, 11)$
- $X|A_t = 3 \sim Beta(5, 5)$
- $X|A_t = 4 \sim Beta(11, 9)$
- $X|A_t = 5 \sim Beta(3, 2)$

Στο Σχήμα 2.1 απεικονίζεται οι συναρτήσεις πυκνότητας πιθανότητας των αμοιβών κάθε δράσης:



Σχήμα 2.1: Συναρτήσεις πυκνότητας πιθανότητας των αμοιβών κάθε δράσης $a = 1, ..., 5$.

Γνωρίζουμε ότι αν μια τυχαία μεταβλητή $X \sim Beta(a, b)$ τότε:

$$E[X] = \frac{a}{a+b}$$

και,

$$Var[X] = \frac{ab}{(a+b)^2(a+b+1)}$$

Συνεπώς το διάνυσμα των μέσω τιμών είναι:

$$(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5) = (0.4, 0.45, 0.5, 0.55, 0.6)$$

και επομένως η βέλτιστη δράση είναι η $a^* = 5$ με μέση αμοιβή $\mu^* = 0.6$.

Αλγόριθμος greedy

Υλοποιήσαμε τον Αλγόριθμο *greedy_ beta* (Παράρτημα) για 2 διαφορετικές τιμές της παραμέτρου M_Exploration = πλήθος γύρων που επιλέγεται κάθε δράση κατά την περίοδο εξερεύνησης για σταθερό M= πλήθος γύρων.

Πρώτη Υλοποίηση

Επιλογή Παραμέτρων

- M= 10000
- M_Exploration= 2

Οι εκτιμήσεις που βρήκε και αλγόριθμος για τα $\mu_i,\ i = 1, 2, ..5$, μέχρι και τον γύρο 10 κατά την περίοδο εξερεύνησης ήταν:

$$(\hat{Q}_{10}(1), \hat{Q}_{10}(2), \hat{Q}_{10}(3), \hat{Q}_{10}(4), \hat{Q}_{10}(5)) = (0.337, 0.328, 0.664, 0.375, 0.563)$$

Συνεπώς ο αλγόριθμος για τους εναπομείναντες γύρους θα επιλέγει την λαβή $\hat{a} = 3$.

Μέτρα απόδοσης μετά το πέρας M γύρων

Μέτρα απόδοσηςς	Τιμή
R(M)	1000
$\overline{p}_M$	0.0002
$\overline{X}_M$	0.4996

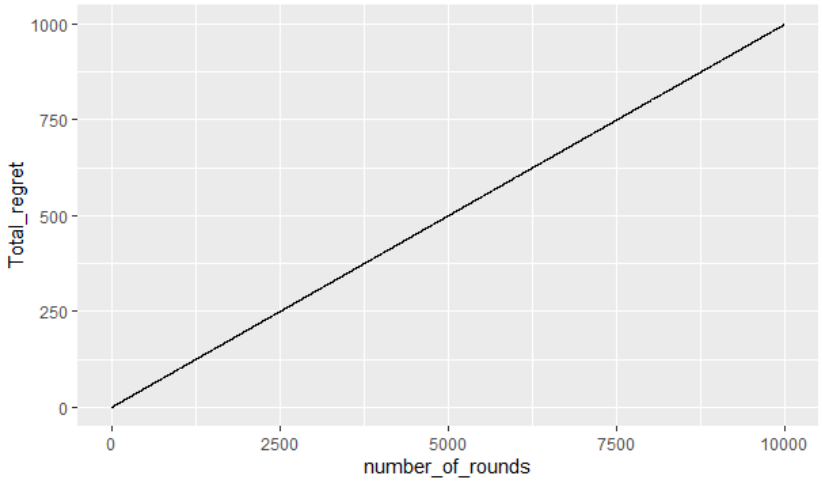
Σχολιασμός Μέτρων Απόδοσης

1. Λόγω της χαμηλής τιμής του M_Exploration ο αλγόριθμος δεν έχει καλές εκτιμήσεις των μ_i και θεωρεί πως βέλτιστη απόφαση είναι η $a = 3$, με αποτέλεσμα για τους εναπομείναντες 9990 γύρους να λαμβάνεται συνεχώς εκείνη.
2. Η τιμή της Συνάρτησης Συνολικής απώλειας είναι γραμμική στο πλήθος των γύρων και μάλιστα μπορούμε να επαληθεύσουμε την τιμή της από την Πρόταση 2.2:

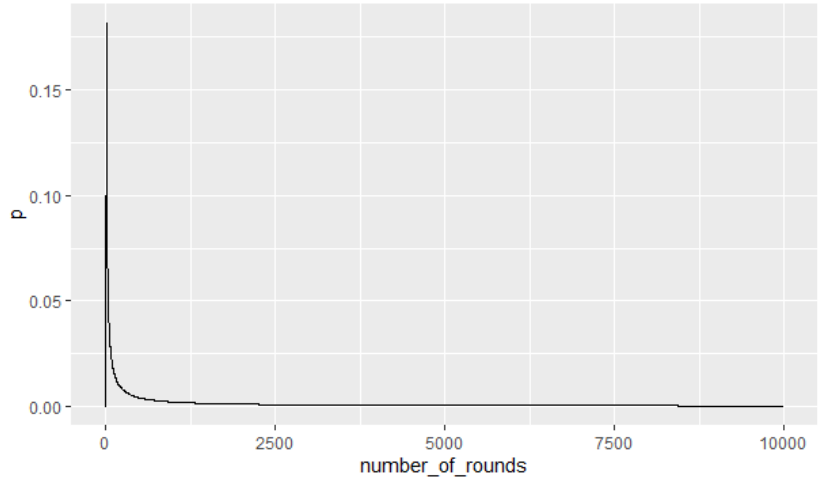
$$\begin{aligned} R(M) &= (\mu^* - \mu')M + \text{M\_Exploration}(k\mu' - \sum_{i=1}^{k} \mu_i) \\ &= (0.6 - 0.5) \cdot 10000 + 2(5 \cdot 0.5 - 2.5) \\ &= 1000 \end{aligned}$$

3. Η τιμή του δειγματικού ποσοστού που λαμβάνεται βέλτιστη απόφαση είναι 0.0002 εφόσον επιλέγεται αποκλειστικά 2 φορές (κατά την περίοδο εξερεύνησης) σε σύνολο 10000 γύρων. Όσο το M αυξάνει τόσο το $\overline{p}_M$ θα συγκλίνει (ντετερμινιστικά μάλιστα) στο 0.
4. Λόγω της μεγάλης τιμής του M η τιμή του δειγματικού μέσου είναι πολύ κοντά στο $\mu_3 = 0.5$.

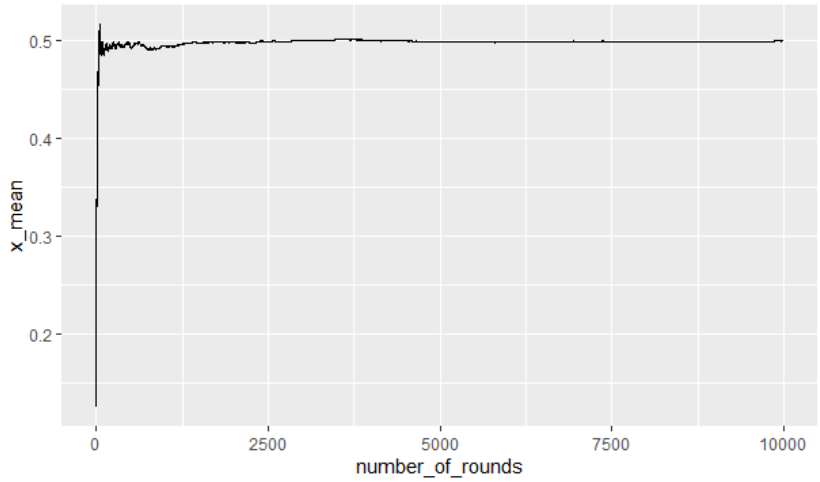
Ακολουθούν οι γραφικές παραστάσεις των μέτρων απόδοσης στο πλήθος των γύρων M.



Σχήμα 2.2: Η συνάρτηση συνολικής Απώλειας.



Σχήμα 2.3: Το ποσοστό του χρόνου που λαμβάνεται η a^*.



Σχήμα 2.4: Ο Δειγματικός Μέσος.

Δεύτερη Υλοποίηση

Επιλογή Παραμέτρων

- M= 10000
- M_Exploration= 200

Οι εκτιμήσεις που βρήκε και αλγόριθμος για τα $\mu_i,\ i = 1, 2, ..5$, μέχρι και τον γύρο 1000 κατά την περίοδο εξερεύνησης ήταν:

$$(\hat{Q}_{1000}(1), \hat{Q}_{1000}(2), \hat{Q}_{1000}(3), \hat{Q}_{1000}(4), \hat{Q}_{1000}(5)) = (0.399, 0.447, 0.511, 0.539, 0.579)$$

Επειδή η τιμή του M_Exploration είναι σχετικά μεγάλη (100 φορές μεγαλύτερη σε σχέση με την πρώτη υλοποίηση) παρατηρούμε ότι οι εκτιμήσεις είναι πολύ κοντά στα πραγματικά μ_i. Αυτό έχει ως αποτέλεσμα ο αλγόριθμος να εντοπίζει ότι η βέλτιστη απόφαση είναι η $a = 5$ με αποτέλεσμα να επιλέγεται συνεχώς αυτή για τους εναπομείναντες γύρους της περιόδου εκμετάλλευσης.

Μέτρα απόδοσης μετά το πέρας M γύρων

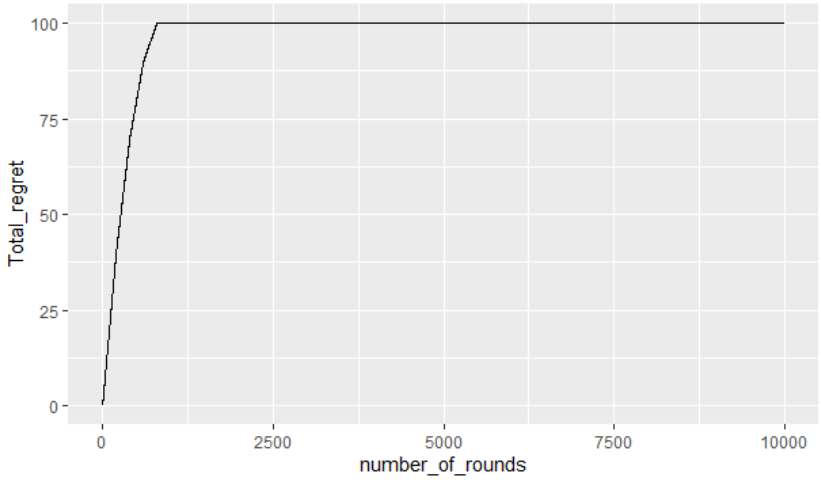
Μέτρα απόδοσηςς	Τιμή
R(M)	100
$\overline{p}_M$	0.92
$\overline{X}_M$	0.5875

Σχολιασμός Μέτρων Απόδοσης

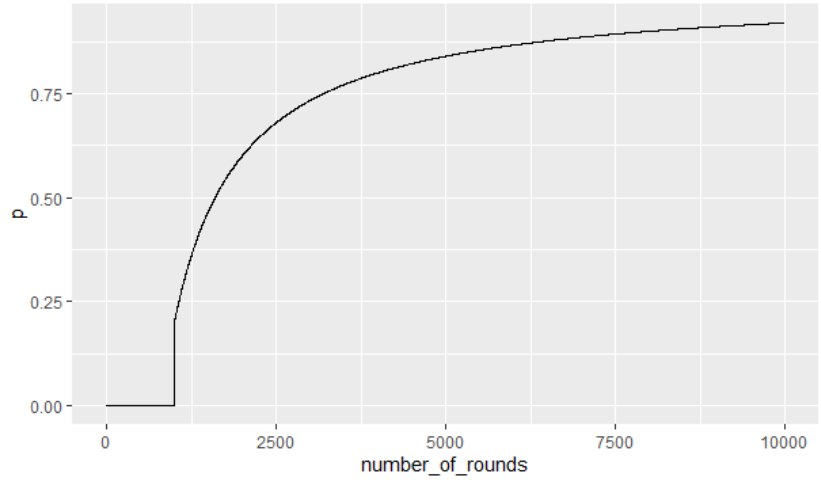
1. Η τιμή της Συνάρτησης Συνολικής απώλειας θα είναι σταθερή και για κάθε $M \geq 800$ και μάλιστα μπορούμε να επαληθεύσουμε την τιμή της από την Πρόταση 2.2:

$$\begin{aligned} R(M) &= \text{M\_Exploration}(k\mu^* - \sum_{i=1}^{k} \mu_i) \\ &= 200 \cdot (5 \cdot 0.6 - 2.5) \\ &= 100 \end{aligned}$$

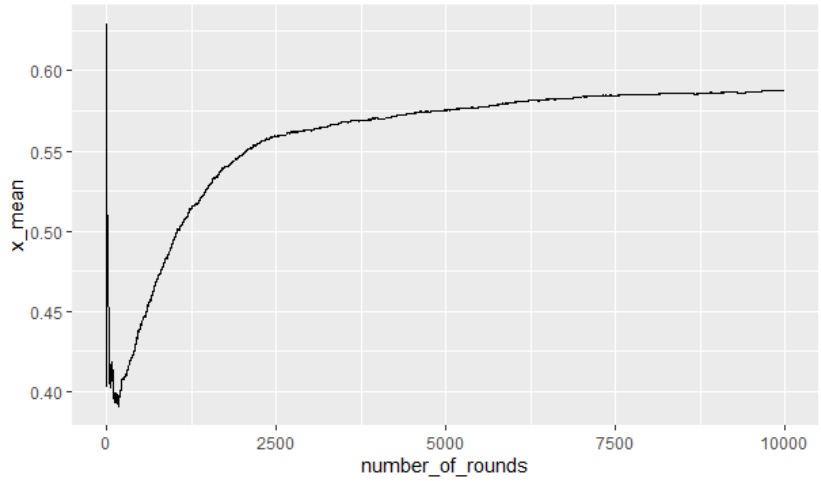
2. Η τιμή του δειγματικού ποσοστού που λαμβάνεται βέλτιστη απόφαση είναι 0.92 εφόσον επιλέγεται 9200 φορές (200 κατά την περίοδο εξερεύνησης και 9000 κατά την περίοδο εκμετάλλευσης) σε σύνολο 10000 γύρων. Όσο το M αυξάνει τόσο το $\overline{p}_M$ θα συγκλίνει στο 1. Επιπλέον, λόγω της μεγάλης τιμής του M η τιμή του δειγματικού μέσου είναι πολύ κοντά στο $\mu^* = 0.6$.



Σχήμα 2.5: Η συνάρτηση συνολικής Απώλειας.



Σχήμα 2.6: Το ποσοστό του χρόνου που λαμβάνεται η a^*.



Σχήμα 2.7: Ο Δειγματικός Μέσος.

Αλγόριθμος ε-greedy

Υλοποιήσαμε τον Αλγόριθμο *epsilon_ greedy_ beta* (Παράρτημα) για 2 διαφορετικές τιμές της παραμέτρου *epsilon* ($\varepsilon = 0.1,\ \varepsilon = 0.01$) για σταθερό πλήθος γύρων $M = 2000$. Η ταχύτητα σύγκλισης των μέτρων απόδοσης στα αρχικά βήματα του αλγόριθμου ε-greedy εξαρτάται σε μεγάλο βαθμό από την ταχύτητα που ο αλγόριθμος θέτει ως πλεονεκτική δράση την a^*. Για **διαφορετικές υλοποιήσεις** της αλληλεπίδρασης υποκειμένου- περιβάλλοντος ο αλγόριθμοι στα αρχικά στάδια μπορεί να οδηγηθούν σε ικανοποιητικές ή μη αμοιβές, αναλόγως και με την τιμή της παραμέτρου ε. Έτσι για να μπορέσουμε να συγκρίνουμε αποτελεσματικά 2 εκτελέσεις του αλγορίθμου για δυο διαφορετικές του ε αυτό θα πρέπει να γίνει ως προς την **μέση** συμπεριφορά τους. Για αυτό το σκοπό προσομοιώσαμε M_sim= 1000 υλοποιήσεις για κάθε τιμή τιμή της παραμέτρου ε. Οι πίνακες που ακολουθούν αφορούν μέσους όρους των μέτρων απόδοσης ως προς τις 1000 προσομοιώσεις.

Υλοποίηση με ε=0.1

Μέτρα απόδοσης μετά το πέρας M γύρων

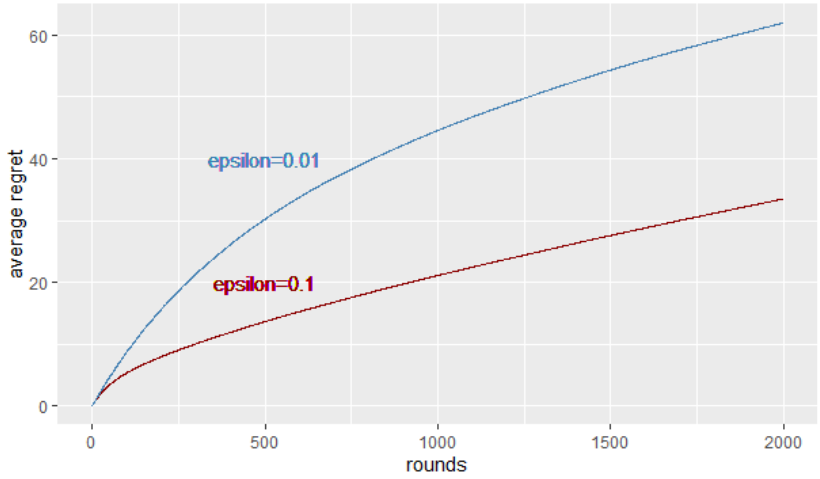
Μέτρα απόδοσης	Τιμή
R(M)	33.4925
$\overline{p}_M$	0.8065
$\overline{X}_M$	0.5832

Υλοποίηση με ε=0.01

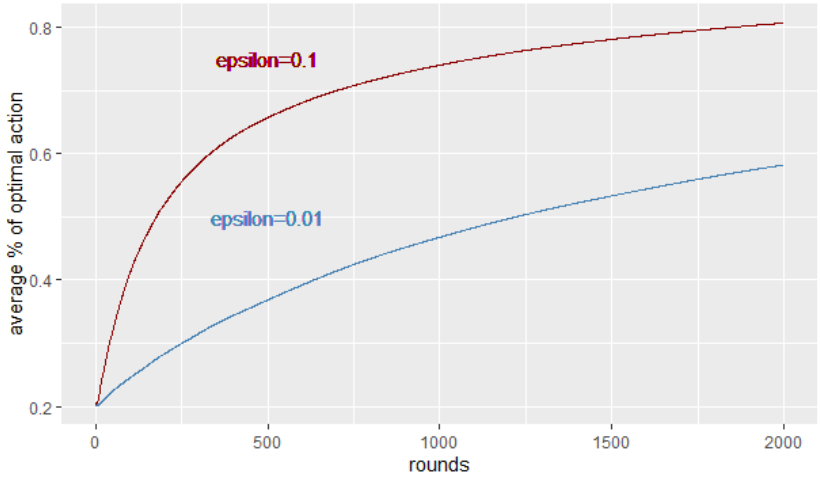
Μέτρα απόδοσης μετά το πέρας M γύρων

Μέτρα απόδοσης	Τιμή
R(M)	61.9439
$\overline{p}_M$	0.5819
$\overline{X}_M$	0.5691

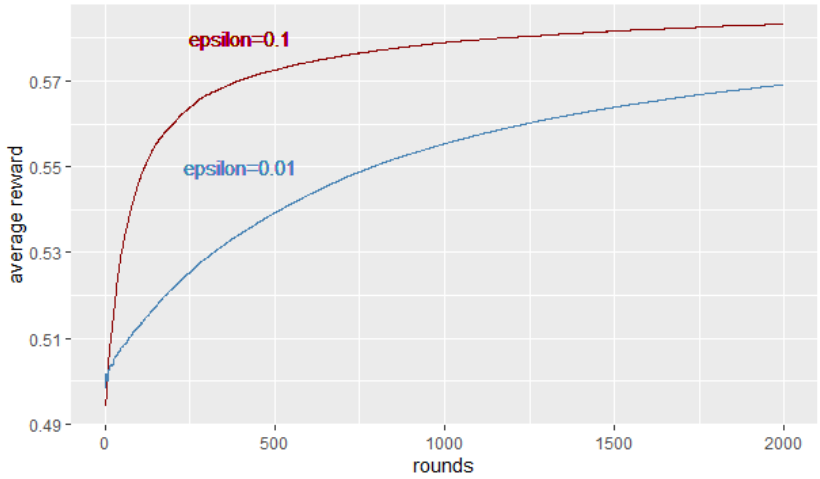
Παρατηρούμε ότι η υλοποίηση με $\varepsilon = 0.1$ υπερτερεί ως προς όλα τα μέτρα απόδοσης έναντι εκείνης με $\varepsilon = 0.01$ στα πρώτους 2000 γύρους. Αυτό συμβαίνει καθώς η πρώτη εντοπίζει κατά μέσο όρο γρηγορότερα την a^* με αποτέλεσμα οδηγείται το υποκείμενο σε καλύτερες αμοιβές. Βέβαια σε βάθος χρόνου αυτή η εικόνα θα αντιστραφεί αφού από την Πρόταση 2.3 το μακροπρόθεσμου ποσοστό του χρόνου που θα λαμβάνεται η a^* θα είναι μεγαλύτερη για μικρότερο ε.



Σχήμα 2.8: Η συνάρτηση μέσης Συνολικής Απώλειας.



Σχήμα 2.9: Το μέσο ποσοστό του χρόνου που λαμβάνεται η a^*.



Σχήμα 2.10: Η Μέση αμοιβή.

Αλγόριθμος UCB1

Υλοποιήσαμε τον Αλγόριθμο *ucb1_ beta* για $M = 10000$ και $M = 100000$:

Υλοποίηση με $M = 10^4$

Μέτρα απόδοσης μετά το πέρας M γύρων

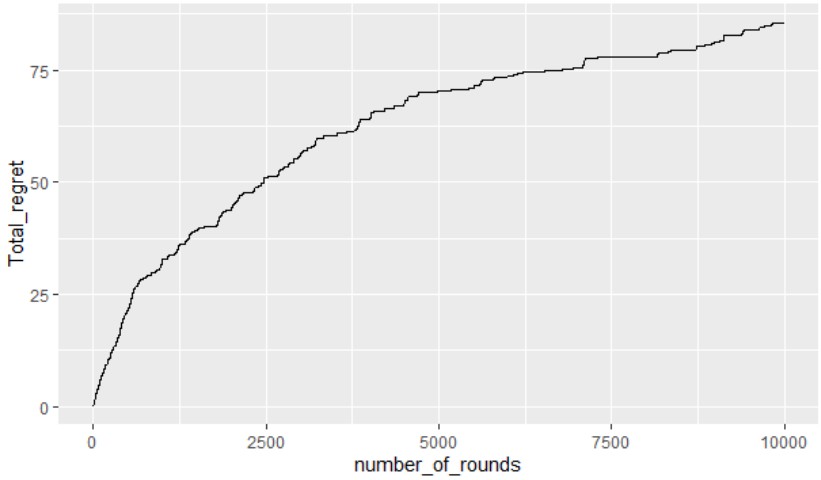
Μέτρα απόδοσης	Τιμή
R(M)	85.55
$\overline{p}_M$	0.8808
$\overline{X}_M$	0..5910

Υλοποίηση με $M = 10^5$

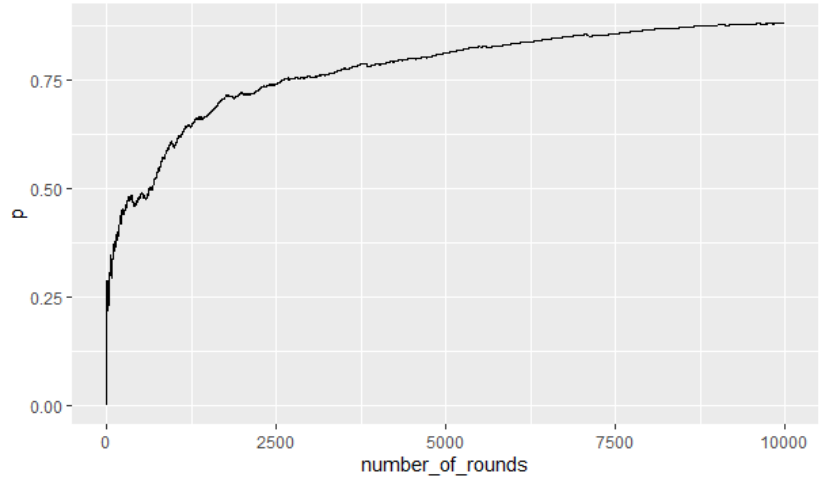
Μέτρα απόδοσης μετά το πέρας M γύρων

Μέτρα απόδοσης	Τιμή
R(M)	89.85
$\overline{p}_M$	0.9858
$\overline{X}_M$	0.5997

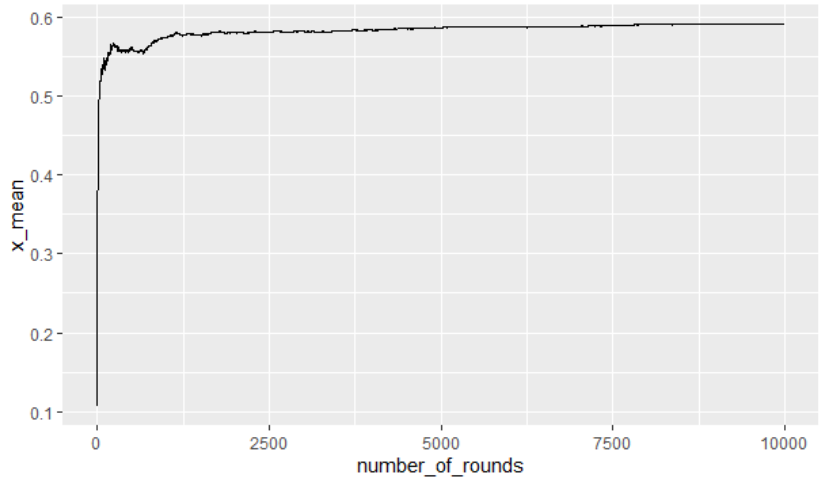
Παρατηρώντας τα μέτρα απόδοσης των 2 υλοποιήσεων μπορούμε να επιβεβαιώσουμε την αποτελεσματικότητα του Αλγορίθμου ucb1. Αρχικά για $M = 10^4$ ο Αλγόριθμος έχει επιλέξει την βέλτιστη δράση a^* σε ποσοστό περίπου 88% ενώ ο δειγματικός μέσος έχει τιμή 0.591 που είναι πολύ κοντά στην $\mu^* = 0.6$. Φυσικά, όπως είναι αναμενόμενο, μεγαλώνοντας το μέγεθος των γύρων για $M = 10^5$, το ποσοστό λήψης της βέλτιστης απόφασης γίνεται περίπου 98.5% (πολύ κοντά στο 100%) ενώ ο δειγματικός μέσος βρίσκεται ακόμα πλησιέστερα στο 0.6 με τιμή 0.5997. Ιδιαίτερο ενδιαφέρον παρουσιάζει η ελάχιστη αύξηση της συνολικής απώλειας κατά την δεύτερη υλοποίηση της τάξης του 5% παρά το γεγονός ότι το μέγεθος των γύρων M δεκαπλασιάστηκε. Αυτό οφείλεται στην **λογαριθμική** αύξηση της συνάρτησης συνολικής απώλειας που μας εξασφαλίζει ο Αλγόριθμος ucb1. Τα διαγράμματα που ακολουθούν αφορούν την εξέλιξη των μέτρων απόδοσης της υλοποίησης με $M = 10^4$.



Σχήμα 2.11: Η συνάρτηση Συνολικής Απώλειας.



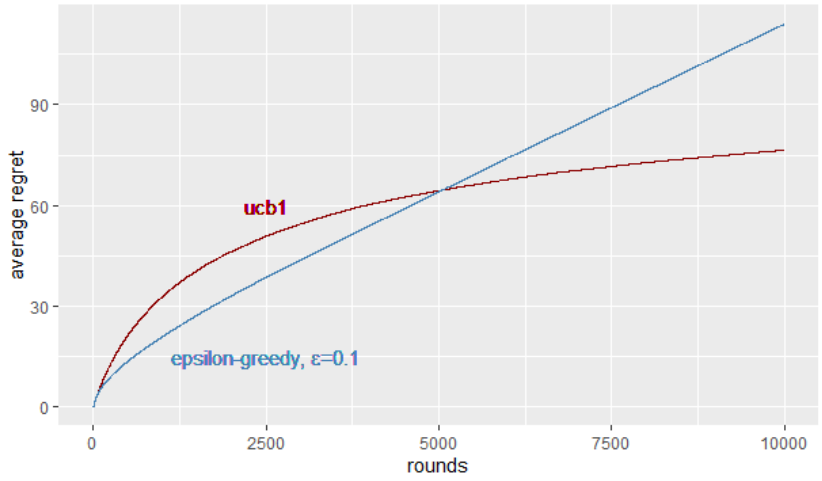
Σχήμα 2.12: Το ποσοστό του χρόνου που λαμβάνεται η a^*.



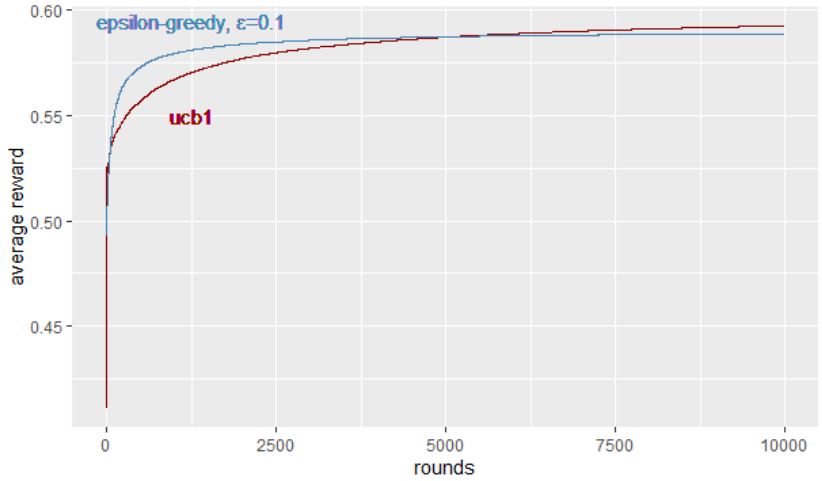
Σχήμα 2.13: Ο Δειγματικός Μέσος.

Σύγκριση UCB1 και ε-greedy

Τέλος συγκρίναμε τους αλγορίθμους UCB1 και ε-greedy (με ε= 0.01) ως προς τις μέσες επιδόσεις τους. Για αυτό το σκοπό προσομοιώσαμε 1000 υλοποιήσεις ή **επεισόδια** για κάθε αλγόριθμο σε σύνολο 10000 γύρων. Όπως μπορούμε να διαπιστώσουμε και από τα διαγράμματα που ακολουθούν ο αλγόριθμος ε-greedy φαίνεται να αποδίδει καλύτερα περίπου μέχρι τους πρώτους 5000 γύρους ενώ στη συνέχεια λόγω του μεγάλου πλήθους γύρων ο αλγόριθμος ucb1 είναι αποτελεσματικότερος.



Σχήμα 2.14: Η συνάρτηση μέσης Συνολικής Απώλειας.



Σχήμα 2.15: Μέση αμοιβή.

Κεφάλαιο 3

Ενισχυτική Μάθηση

Σε αυτό το Κεφάλαιο, θα γίνει μια εισαγωγή στο γενικότερο πλαίσιο εφαρμογής της Ενισχυτικής Μάθησης, όπου κάθε κατάσταση χαρακτηρίζεται από το δικό της σύνολο αποφάσεων. Στο Κεφάλαιο 2, εκτιμούσαμε την αξία κάθε απόφασης $Q(a)$, το οποίο ήταν αρκετό εφόσον οι καταστάσεις δεν έπαιζαν κάποιο ουσιαστικό ρόλο. Ωστόσο, το ενδιαφέρον μας τώρα θα μεταφερθεί στην εκτίμηση της αξίας της κάθε κατάστασης $u(s)$ ή στην αξία του κάθε ζεύγους κατάστασης-απόφασης $Q(s, a)$.

Αρχικά, θα γίνει μια επισκόπηση στις Μαρκοβιανές Διαδικασίες Αποφάσεων, στις αρχές του Δυναμικού Προγραμματισμού, και στις εξισώσεις του Bellman οι οποίες αποτελούν τον πυρήνα για την δημιουργία Αλγορίθμων που προσεγγίζουν βέλτιστες πολιτικές. Στη συνέχεια θα διατυπώσουμε κάποιους αναδρομικούς αλγόριθμους που προσεγγίζουν βέλτιστες πολιτικές σε περιπτώσεις που η **δυναμική**[1] του περιβάλλοντος είναι γνωστή. Τέλος, θα αναπτύξουμε κάποιους από τους σημαντικότερους αλγόριθμους που προσεγγίζουν βέλτιστες πολιτικές όπως η **Μέθοδος Monte Carlo**, ο **Αλγόριθμος Sarsa**, και η **Q-Μάθηση** που υλοποιούνται σε περιπτώσεις που η δυναμική του περιβάλλοντος είναι άγνωστη.

3.1 Πεπερασμένες Μαρκοβιανές Διαδικασίες Αποφάσεων

Στο τέλος του Κεφαλαίου 1, περιγράψαμε τις βασικές πτυχές της αλληλεπίδρασης του υποκειμένου με τον περιβάλλον του, ενώ ορίσαμε κάποιες κάποιες έννοιες-κλειδιά οι οποίες παίζουν καθοριστικό ρόλο στην Ενισχυτική Μάθηση. Στην παρούσα ενότητα θα προχωρήσουμε με περισσότερη λεπτομέρεια στη μορφή του γενικότερου προβλήματος

[1] Ο όρος Δυναμική του περιβάλλοντος παραπέμπει στην κατανομή πιθανότητας των αμοιβών του, η οποία καθορίζεται εξ' ολοκλήρου από την κατάσταση που βρίσκεται το υποκείμενο και την απόφαση που έλαβε. Θα διατυπωθεί προσεχώς ο αυστηρός ορισμός που θα χρησιμοποιείται στην παρούσα Εργασία.

της Ενισχυτικής Μάθησης που είναι οι Μαρκοβιανές Διαδικασίες Αποφάσεων. Επίσης θα ορίσουμε και θα μελετήσουμε νέες ποσότητες οι οποίες απουσίαζαν από το απλουστευμένο πλαίσιο των Multi-Armed Bandits.

Ας θυμηθούμε, εν συντομία, την ακολουθιακή δομή που χαρακτηρίζει την Ενισχυτική Μάθηση. Το υποκείμενο και το Περιβάλλον του αλληλεπιδρούν σε μια σειρά από διακριτά χρονικά βήματα $t = 0, 1, 2, 3, \ldots$.[2] Σε κάθε χρονική στιγμή το υποκείμενο βρίσκεται σε μια κατάσταση S_t, και με βάση αυτή, καλείται να επιλέξει μια απόφαση A_t μέσα από ένα σύνολο δυνατών αποφάσεων $A(S_t)$, που εξαρτάται από την εκάστοτε κατάσταση. Στη συνέχεια λαμβάνει μια αμοιβή R_t ως συνέπεια της προηγούμενης απόφασης, και έτσι βρίσκεται εκ νέου στην επόμενη κατάσταση S_{t+1}. Όλα τα παραπάνω συνθέτουν την δημιουργία μιας **τροχιάς** ως εξής:

$$S_0, A_0, R_1, S_1, A_1, R_2, \ldots \tag{3.1}$$

Αν μάλιστα μας ενδιαφέρει η εξέλιξη μιας τροχιάς μέχρι μια συγκεκριμένη χρονική στιγμή t μπορούμε να ορίσουμε την **ιστορία** της αλληλεπίδρασης μέχρι και την χρονική στιγμή t ως εξής:

$$H_t := S_0, A_0, R_1, S_1, A_1, R_2, \ldots, R_t, S_t, A_t \tag{3.2}$$

Ας εστιάσουμε τώρα στον τρόπο με τον οποίο κατανέμονται από κοινού οι αμοιβές με τις επόμενες καταστάσεις. Στις **Μαρκοβιανές Διαδικασίες Αποφάσεων** είναι αρκετή η γνώση των πληροφοριών του προηγούμενου βήματος και **όχι** όλου του υπάρχοντος ιστορικού. Επιπλέον θα υποθέσουμε ότι οι καταστάσεις, οι αποφάσεις και οι αμοιβές είναι διακριτές τυχαίες μεταβλητές με αποτέλεσμα τα σύνολα $S, A(S), R$ να είναι πεπερασμένα.[3] Έτσι ορίζουμε ως **δυναμική** του περιβάλλοντος την ακόλουθη πιθανότητα:

$$p(s', r|s, a) := P(S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a), \tag{3.3}$$

για κάθε $s, s' \in S,\ r \in R$ και $a \in A(s)$. Η γνώση της p καθορίζει μονοσήμαντα την πιθανότητα μετάβασης από το ζεύγος (s, a) στο (s', r). Ενώ η απουσία εξάρτησης από όλη την υπάρχουσα ιστορία, μπορεί να μοιάζει περιοριστική, στην πραγματικότητα είναι πολλές οι εφαρμογές οι οποίες μοντελοποιούνται ως ΜΔΑ. Η ιδέα είναι η μοντελοποίηση να γίνεται με τέτοιο τρόπο ώστε όλη η διαθέσιμη πληροφορία που το υποκείμενο έχει λάβει έως τώρα λόγω της αλληλεπίδρασης, να συνοψίζεται στην υπάρχουσα κατάσταση. Σε αυτή την περίπτωση λέμε ότι οι καταστάσεις χαρακτηρίζονται από τη **Μαρκοβιανή Ιδιότητα**.

[2] Περιοριζόμαστε στο πλαίσιο του διακριτού χρόνου, το οποίο όμως μπορεί να επεκταθεί και για συνεχή χρόνο.

[3] Οι Μαρκοβιανές Διαδικασίες Αποφάσεων με Πεπαρασμένα $S, A(S)$ και R καλούνται **πεπερασμένες**.

Η (3.3) για συγκεκριμένες τιμές των $s \in S$ και $a \in A(s)$ καθορίζει μια διδιάστατη κατανομή πιθανότητας και κατά συνέπεια θα πρέπει να ισχύει:

$$\sum_{s' \in S} \sum_{r \in R} p(s', r|s, a) = 1, \text{ για κάθε } s \in S \text{ και } a \in A(s). \tag{3.4}$$

Ας επανέλθουμε τώρα στην περιγραφή του στόχου του υποκειμένου. Έστω λοιπόν ότι το υποκείμενο βρίσκεται στο χρονικό βήμα t. Σε προβλήματα πεπερασμένου ορίζοντα, που η αλληλεπίδραση σταματά μετά το πέρας ενός βήματος T, ο στόχος του είναι η μεγιστοποίηση της μέσης τιμής της ποσότητας:

$$G_t := R_{t+1} + R_{t+2} + \ldots R_T \tag{3.5}$$

Το G_t είναι το **συνολικό κέρδος** του υποκειμένου που θα εισπράξει από την επόμενη χρονική στιγμή μέχρι και το τέλος της αλληλεπίδρασης. Η (3.5), όπως προαναφέρθηκε, έχει νόημα μόνο σε προβλήματα που χαρακτηρίζονται από κάποια **τερματική κατάσταση**. Η μετάβαση στην τερματική κατάσταση σηματοδοτεί την λήξη ενός **επεισοδίου**. Μετά την λήξη του επεισοδίου η διαδικασία συνήθως επανεκκινείται σε κάποια νέα αρχική κατάσταση και **ανεξάρτητα** από την έκβαση της προηγούμενης. Έτσι τα προβλήματα πεπερασμένου ορίζονται δύνανται να σπάσουν σε διαφορετικές εκβάσεις, με διαφορετικές καταστάσεις, αποφάσεις και αμοιβές που καλούνται **επεισόδια ή στιγμιότυπα** της αλληλεπίδρασης. Τέλος να σημειώσουμε, πως ο χρόνος τερματισμού T είναι τυχαία μεταβλητή η οποία δύναται να διαφέρει από επεισόδιο σε επεισόδιο.

Τα προβλήματα άπειρου ορίζοντα, αντιθέτως δεν χαρακτηρίζονται από τη διαμέριση τους σε ανεξάρτητα επεισόδια. Αν από την άλλη μεριά, ο ορισμός του συνολικού κέρδους ήταν όμοιος με την (3.5) τότε αυτό θα δημιουργούσε προβλήματα απειρισμού της G_t. Για την αποφυγή του προηγούμενου προβλήματος, σε προβλήματα άπειρου ορίζοντα, ο στόχος του υποκειμένου μεταφέρεται στην μεγιστοποίηση της μέσης τιμής της ποσότητας:

$$G_t := R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{3.6}$$

Η ποσότητα G_t, τώρα είναι το **συνολικό αποπληθωρισμένο κέρδος** του υποκειμένου στον άπειρο ορίζοντα και ο αριθμός $\gamma \in [0, 1)$ ονομάζεται συντελεστής αποπληθωρισμού. Όσο το γ πλησιάζει την τιμή 0 τόσο πιο πολύ το υποκείμενο ενδιαφέρεται για τις άμεσες αμοιβές, ενώ όσο το γ προσεγγίζει το 1, τόσο το ενδιαφέρον του είναι εξίσου έντονο για άμεσες αμοιβές όπως και για τις αμοιβές που θα εισπράξει στο μέλλον. Να σημειώσουμε πως αν η ακολουθία των αμοιβών είναι φραγμένη, τότε ότι η σειρά στην (3.6) θα συγκλίνει σε πεπερασμένο αριθμό. Πράγματι, αν η ακολουθία των αμοιβών είναι φραγμένη τότε υπάρχει $R \in \mathbb{R}$ τέτοιο ώστε $R_{t+k+1} \leq R$ για κάθε $k = 0, 1, 2, \ldots$.

Συνεπώς,

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \leq R \sum_{k=0}^{\infty} \gamma^k = \frac{R}{1-\gamma} < +\infty.$$

Φυσικά, για την επίτευξη του στόχου του υποκειμένου είναι απαραίτητη η εύρεση πολιτικών οι οποίες θα του εξασφαλίζουν όσο το δυνατόν καλύτερες αμοιβές. Υπενθυμίζουμε ότι η **πολιτική** είναι ο τρόπος συμπεριφοράς του, βρισκόμενο σε οποιαδήποτε κατάσταση. Αποτελεί δηλαδή συνάρτηση, από όλες τις καταστάσεις σε πιθανότητες να επιλέξει κάθε δυνατή απόφαση. Αν το υποκείμενο ακολουθεί μια συγκεκριμένη πολιτική, τότε πρέπει να είναι σε θέση να αξιολογεί το πόσο «καλή» ή «κακή» είναι οποιαδήποτε κατάσταση. Αυτό γίνεται μέσω της **συνάρτησης αξίας** η οποία εκφράζει το αναμενόμενο συνολικό κέρδος που περιμένει εισπράξει, βρισκόμενο σε μια τωρινή κατάσταση και ακολουθώντας την συγκεκριμένη πολιτική. Υπενθυμίζουμε ότι για Μαρκοβιανές Διαδικασίες Αποφάσεων η συνάρτηση αξίας ορίζεται ως εξής[4]:

$$u_\pi(s) := \mathbb{E}_\pi \left[G_t \mid S_t = s \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s \right], \text{ για κάθε } s \in S \tag{3.7}$$

Αν επιθυμούμε να προσδιορίσουμε την αξία μιας κατάστασης s σε συνδυασμό με τη λήψη μιας απόφασης a τότε ορίζουμε ως **συνάρτηση αξίας του ζεύγους κατάστασης-απόφασης** την ποσότητα:

$$Q_\pi(s,a) := \mathbb{E}_\pi \left[G_t \mid S_t = s, A_t = a \right] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a \right] \tag{3.8}$$

για κάθε $s \in S$ και $a \in A(s)$. Η (3.7) μας πληροφορεί για την αξία μιας κατάστασης ανεξάρτητα από το ποια απόφαση θα ληφθεί στο βήμα t, ενώ η (3.8) μας πληροφορεί για την αξία μιας κατάστασης σε συνδυασμό με την επιλογή συγκεκριμένης απόφασης στο βήμα t. Αν μάλιστα χρησιμοποιήσουμε το **Θεώρημα Διπλής Μέσης Τιμής** στην (3.7) δεσμεύοντας ως προς την A_t, οδηγούμαστε στην ακόλουθη έκφραση που συνδέει την συνάρτηση αξίας μιας κατάστασης με τις συναρτήσεις αξίας κατάστασης- απόφασης στο βήμα t ως εξής:

$$u_\pi(s) = \sum_{a \in A(s)} \pi(a|s)\ Q_\pi(s,a) \tag{3.9}$$

[4]Θεωρούμε ενοποιημένο συμβολισμό για τα προβλήματα πεπερασμένου και άπειρου ορίζοντα για το G_t, εκείνο που εμφανίζεται στην (3.6). Πράγματι αν το πρόβλημα είναι πεπερασμένου ορίζοντα, με τερματική κατάσταση T, τότε θέτοντας $R_t = 0,\ \forall\ t \geq T$ και επιτρέποντας $\gamma = 1$ επιστρέφουμε στην (3.5).

3.2 Εξισώσεις του Bellman

Ας εστιάσουμε τώρα στο πώς μπορούμε να υπολογίσουμε τη συνάρτηση αξίας κάθε κατάστασης ως προς συγκεκριμένες πολιτικές στην περίπτωση που η δυναμική του περιβάλλοντος **είναι γνωστή**. Σε αυτές θα μας δώσουν απάντηση, οι γνωστές από τον Δυναμικό Προγραμματισμό, **εξισώσεις του Bellman**. Οι τελευταίες, βασίζονται σε μια θεμελιώδη ιδιότητα των συναρτήσεων αξίας, που είναι η αναδρομική τους έκφραση με βάση όλες τις συναρτήσεις αξίας του επόμενου βήματος. Για να μπορέσουμε να οδηγηθούμε στην διατύπωση των εξισώσεων του Bellman, αρχικά χρειαζόμαστε μια αναδρομική έκφραση για το συνολικό κέρδος G_t:

$$\begin{aligned} G_t &= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} = R_{t+1} + \sum_{k=1}^{\infty} \gamma^k R_{t+k+1} \\ &= R_{t+1} + \gamma \sum_{k=1}^{\infty} \gamma^{k-1} R_{t+1+k} = R_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k R_{t+1+k+1} \\ &= R_{t+1} + \gamma G_{t+1} \end{aligned} \tag{3.10}$$

Με βάση την (3.10) μπορούμε να οδηγηθούμε σε μια αναδρομική έκφραση για την αξία $u_\pi(s)$ ως εξής:

$$\begin{aligned} u_\pi(s) &= \mathbb{E}_\pi \left[G_t \mid S_t = s \right] \\ &= \mathbb{E}_\pi \left[R_t + \gamma G_{t+1} \mid S_t = s \right] \\ &= \sum_a \pi(a|s)\, \mathbb{E}_\pi \left[R_t + \gamma G_{t+1} \mid S_t = s, A_t = a \right] && (3.11) \\ &= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \Big[\mathbb{E}_\pi \left[R_t + \gamma G_{t+1} \mid R_t = r, S_{t+1} = s' \right] \Big] && (3.12) \\ &= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \Big[r + \gamma\, \mathbb{E}_\pi \left[G_{t+1} \mid S_{t+1} = s' \right] \Big] && (3.13) \\ &= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u_\pi(s') \right] && (3.14) \end{aligned}$$

Η ισότητα στην (3.11) προκύπτει από το Θεώρημα Διπλής Μέσης Τιμής δεσμεύοντας ως προς όλες τις δυνατές τιμές της απόφασης A_t που μπορούν να ληφθούν κατάσταση s. Στη συνέχεια εφαρμόζουμε εκ νέου το Θεώρημα Διπλής Μέσης Τιμής δεσμεύοντας αυτή τη φορά ως προς όλα τα δυνατά ζεύγη επόμενων καταστάσεων και αμοιβών που μπορούν να προκύψουν από την προηγούμενη κατάσταση s και απόφαση a. Να σημειώσουμε πως στην (3.12) παραλείπουμε από την δέσμευση στην μέση τιμή τις δεσμεύσεις $S_t = a$ και $A_t = a$, δεδομένου ότι για τον υπολογισμό της αρκούν οι δεσμεύσεις $R_t = r$ και $S_{t+1} = a$. Χρησιμοποιώντας την γραμμικότητα της μέσης τιμής και απαλοίφοντας

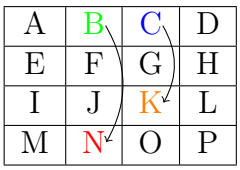
την εξάρτηση της R_t από την μέση τιμή του G_{t+1} οδηγούμαστε στην (3.13). Για την μετάβαση στην (3.14) αρκεί να παρατηρήσουμε ότι η $\mathbb{E}_\pi\left[G_{t+1} \mid S_{t+1} = s'\right]$ ταυτίζεται εξ΄ ορισμού με την $u_\pi(s')$. Η σχέση (3.14) για όλες τις τιμές του $s \in S$ συνθέτει τις εξισώσεις του Bellman για τον υπολογισμό της αξίας των καταστάσεων για μια **συγκεκριμένη πολιτική π**.

Με παρόμοιο σκεπτικό, μπορούμε να οδηγηθούμε στις εξισώσεις του Bellman για τις **συναρτήσεις αξίας των ζευγών κατάστασης-απόφασης**:

$$\begin{aligned}
Q_\pi(s,a) &= \mathbb{E}_\pi\left[G_t \mid S_t = s, A_t = a\right] \\
&= \mathbb{E}_\pi\left[R_t + \gamma G_{t+1} \mid S_t = s, A_t = a\right] \\
&= \sum_{s'}\sum_{r} p(s', r|s, a)\Big[\,\mathbb{E}_\pi\left[R_t + \gamma G_{t+1} \mid R_t = r, S_{t+1} = s'\right]\Big] \\
&= \sum_{s'}\sum_{r} p(s', r|s, a)\Big[r + \gamma\,\mathbb{E}_\pi\left[G_{t+1} \mid S_{t+1} = s'\right]\Big] \qquad (3.15) \\
&= \sum_{s'}\sum_{r} p(s', r|s, a)\Big[r + \gamma\sum_{a'}\pi(a'|s')\,\mathbb{E}_\pi\left[G_{t+1} \mid S_{t+1} = s', A_{t+1} = a'\right]\Big] \qquad (3.16) \\
&= \sum_{s'}\sum_{r} p(s', r|s, a)\Big[r + \gamma\sum_{a'}\pi(a'|s')Q_\pi(s', a')\Big] \qquad (3.17)
\end{aligned}$$

Να σημειώσουμε ότι οι ισότητες μέχρι την (3.15) προκύπτουν με την ίδια λογική όπως για τον υπολογισμό της αναδρομικής έκφρασης του $u_\pi(s)$. Στη συνέχεια, θέλοντας να εμφανίσουμε την επόμενη απόφαση A_{t+1} μέσα στην δέσμευση, χρησιμοποιούμε το Θεώρημα Διπλής Μέσης Τιμής δεσμεύοντας ως προς όλες τις δυνατές τιμές της A_{t+1}, οι οποίες θα προκύπτουν με πιθανότητα $\pi(a'|s')$ λόγω της πολιτικής π. Έτσι φτάνουμε στην σχέση (3.16), στην οποία αντικαθιστώντας την $\mathbb{E}_\pi\left[G_{t+1} \mid S_{t+1} = s', A_{t+1} = a'\right]$ με βάση τον ορισμό της, $Q_\pi(s', a')$, καταλήγουμε στην (3.17), η οποία, για όλες τις τιμές των $s' \in S$ και $a' \in A(s')$ συνοψίζει τις **εξισώσεις του Bellman για τις αξίες όλων των δυάδων κατάστασης-απόφασης**.

Παράδειγμα 3.1 Στο Σχήμα 3.1 απεικονίζεται ένα τετραγωνικό πλέγμα, το οποίο αποτελεί αναπαράσταση μιας απλής πεπερασμένης Μαρκοβιανής Διαδικασίας Αποφάσεων. Τα κελιά του πλέγματος αντιστοιχούν στις καταστάσεις του υποκειμένου. Σε κάθε κελί υπάρχουν 4 δυνατές αποφάσεις: **αριστερά**($\leftarrow$), **δεξιά**($\rightarrow$), **πάνω**($\uparrow$), **κάτω**($\downarrow$) και η επιλογή καθεμιάς οδηγεί ντετερμινιστικά το υποκείμενο στην επιθυμητή κατεύθυνση. Οι αποφάσεις οι οποίες θα οδηγούσαν το υποκείμενο εκτός πλέγματος αφήνουν την τοποθεσία του αμετάβλητη αλλά επίσης και σε αμοιβή -2 μονάδων, ως ποινή για την έξοδο του από το πλέγμα. Όλες οι υπόλοιπες δράσεις οδηγούν σε αμοιβή 0, εκτός από εκείνες που μετακινούν το υποκείμενο εκτός των ειδικών καταστάσεων B και C. Στην κατάστα-



Σχήμα 3.1: Το παράδειγμα του πλέγματος.

ση B και οι 4 δράσεις οδηγούν στην μετακίνηση του υποκειμένου στην κατάσταση N με αμοιβή +20, ενώ από την κατάσταση C όλες οι δράσεις οδηγούν στην μετακίνηση του υποκειμένου στην κατάσταση K με αμοιβή +10. Ας υποθέσουμε επιπλέον ότι το υποκείμενο διαλέγει και τις 4 αποφάσεις με ίση πιθανότητα, σε όλες τις καταστάσεις. Τέλος, επειδή πρόκειται για πρόβλημα άπειρου ορίζοντα, θα επιλέξουμε συντελεστή αποπληθωρισμού $\gamma = 0.9$. Σκοπός μας είναι η εύρεση της αξίας κάθε κατάστασης κάτω από την **ομοιόμορφη τυχαία πολιτική του υποκειμένου**.

Αρχικά ας δούμε ποια είναι τα σύνολα καταστάσεων ,αποφάσεων και αμοιβών:

- **Σύνολο Καταστάσεων**: $S = \{A, B, C, D, \ldots, O, P\}$
- **Σύνολα Αποφάσεων**: $A(s) = \{\leftarrow, \rightarrow, \uparrow, \downarrow\}$ για κάθε $s \in S$
- **Σύνολο Αμοιβών**: $R = \{-2, 0, +10, +20\}$

Η δυναμική του περιβάλλοντος σε αυτό το παράδειγμα είναι **ντετερμινιστική**. Αν για παράδειγμα το υποκείμενο βρίσκεται στην κατάσταση F και αποφασίσει " $\rightarrow$ " τότε αυτό θα οδηγηθεί με βεβαιότητα στην κατάσταση G με αμοιβή 0 σύμφωνα με τις προδιαγραφές του παραδείγματος. Δηλαδή θα ισχύει:

$$p(s', r|F, \rightarrow) = \begin{cases} 1, \text{αν } s' = G \text{ και } r = 0 \\ 0, \text{ αλλιώς} \end{cases}$$

Όσον αφορά την πολιτική π, θα ισχύει:

$$\pi(a|s) = \frac{1}{4}, \text{ για κάθε } s \in S \text{ και } a \in A(s).$$

Ας δούμε τώρα την μορφή που θα πάρουν οι εξισώσεις του Bellman σε αυτό το παράδειγμα. Ξεκινάμε από την κατάσταση A. Η εξίσωση του Bellman για την κατάσταση A θα έχει ως εξής:

$$\begin{aligned} u_\pi(A) = &\frac{1}{4}\Big(-2 + 0.9u_\pi(A)\Big) + \frac{1}{4}\Big(-2 + 0.9u_\pi(A)\Big) \\ &+ \frac{1}{4}\Big(+0 + 0.9u_\pi(B)\Big) + \frac{1}{4}\Big(+0 + 0.9u_\pi(E)\Big) \end{aligned}$$

Ο πρώτος όρος αφορά την περίπτωση που το υποκείμενο επιλέξει " ← ". Τότε επειδή θα ευρεθεί εκτός πλέγματος θα εισπράξει αμοιβή -1 και θα ξαναγυρίσει στην Α. Ο δεύτερος όρος προκύπτει όμοια, αφού και η επιλογή " ↑ " θα επιφέρει το ίδιο αποτέλεσμα. Ο τρίτος όρος αφορά στην περίπτωση όπου το υποκείμενο θα επιλέξει " → ". Τότε δεν θα εισπράξει κάποια αμοιβή και θα μεταφερθεί στην κατάσταση Β. Τέλος, αν το υποκείμενο επιλέξει " ↓ " δεν θα εισπράξει πάλι κάποια άμεση αμοιβή και θα μετακινηθεί στην κατάσταση Ε. Όσον αφορά τώρα την αξία της κατάστασης Β, επειδή όλες οι επιλογές καταλήγουν στην κατάσταση Ν οδηγούμαστε στην εξίσωση:

$$u_\pi(B) = 20 + 0.9u_\pi(N)$$

Εφαρμόζοντας τις εξισώσεις του Bellman και για τις υπολειπόμενες καταστάσεις θα προκύψει ένα **γραμμικό σύστημα 16 εξισώσεων και 16 αγνώστων:**

$$\begin{cases} 0.55u_\pi(A) - 0.225u_\pi(B) - 0.225u_\pi(E) = -1 \\ u_\pi(B) - 0.9u_\pi(N) = 20 \\ u_\pi(C) - 0.9u_\pi(K) = 10 \\ 0.55u_\pi(D) - 0.225u_\pi(C) - 0.225u_\pi(H) = -1 \\ \vdots \\ 0.55u_\pi(M) - 0.225u_\pi(I) - 0.225u_\pi(N) = -1 \\ 0.775u_\pi(N) - 0.225u_\pi(M) - 0.225u_\pi(J) - 0.225u_\pi(0) = -0.5 \\ 0.775u_\pi(O) - 0.225u_\pi(N) - 0.225u_\pi(K) - 0.225u_\pi(P) = -0.5 \\ 0.775u_\pi(P) - 0.225u_\pi(O) - 0.225u_\pi = -1 \end{cases}$$

Στο Σχήμα 3.2 παρουσιάζονται οι αξίες κάθε κατάστασης οι οποίες προέκυψαν, λύνοντας το παραπάνω γραμμικό σύστημα. Παρατηρούμε αρχικά, ότι η αξία φθίνει όσο πλησιάζουμε το κάτω μέρος του πλέγματος, και ειδικότερα στις άκρες του. Αυτό αντικατοπτρίζει αφενός την μεγάλη πιθανότητα που υπάρχει να βρεθεί το υποκείμενο εκτός πλέγματος και άρα να ζημιωθεί με αμοιβή -2 μονάδων, και αφετέρου ότι βρίσκονται σχετικά μακριά από τις καταστάσεις B και C, οι οποίες είναι οι μόνες που επιφέρουν κάποια θετική αμοιβή. Η κατάσταση B, είναι η καλύτερη κατάσταση στην οποία μπορεί να βρεθεί το υποκείμενο, παρ΄ όλα αυτά η αξία της, 18.55, είναι μικρότερη από την άμεση αμοιβή της 20. Αυτό συμβαίνει, διότι από την κατάσταση B, το υποκείμενο θα βρεθεί με βεβαιότητα στη κατάσταση N στην οποία είναι πιθανό από κει και πέρα να βρεθεί εκτός πλέγματος. Από την άλλη μεριά, η αξία της κατάστασης C είναι 10.44, μεγαλύτερη από τη άμεση αμοιβή της που είναι 10, καθώς η κατάσταση K, που θα μεταβεί στην συνέχεια, δεν βρίσκεται σε τόσο μειονεκτική θέση. Αντιθέτως, η αξία της κατάστασης K είναι ίση με 0.49, που είναι θετική. Η θετικότητα στην αξία της K υποδηλώνει ότι, το όφελος του να βρεθούμε από την K, στις καταστάσεις B ή C υπερτερεί έναντι του ρίσκου του να βρεθούμε εκτός πλέγματος από τη θέση αυτή.

7.03	18.55	10.44	2.89
3.09	6.02	4.05	1.06
-0.17	1.06	0.49	-1.06
-2.54	-1.61	-1.84	-3.00

Σχήμα 3.2: Αξίες των καταστάσεων του Παραδείγματος 3.1 κάτω από την τυχαία ομοιόμορφη πολιτική. ■

Συμπερασματικά, σε προβλήματα που η δυναμική του περιβάλλοντος είναι γνωστή, και για καθορισμένες πολιτικές, μπορούμε να υπολογίσουμε την αξία κάθε κατάστασης με την βοήθεια των εξισώσεων του Bellman, λύνοντας ένα **γραμμικό σύστημα** ($\mathbf{n} \times \mathbf{n}$), όπου n το πλήθος όλων των δυνατών καταστάσεων του προβλήματος. Αξίζει βέβαια να σημειώσουμε πως ακόμα και σε ένα τόσο εξιδανικευμένο πλαίσιο προβλημάτων (σε προβλήματα δηλαδή που η δυναμική του περιβάλλοντος είναι γνωστή), αν το πλήθος των καταστάσεων είναι πολύ μεγάλο τόσο η δυσκολία στη μοντελοποίηση, όσο και οι περιορισμοί στην υπολογιστικές μας δυνατότητες, μπορούν να οδηγήσουν σε προβλήματα στον υπολογισμό της αξίας κάθε κατάστασης κάτω από καθορισμένες πολιτικές.

3.3 Βέλτιστες Συναρτήσεις Αξίας και Βέλτιστες Πολιτικές

Στην προηγούμενη ενότητα, είδαμε πως μπορούμε να υπολογίσουμε τις συναρτήσεις αξίας κάθε κατάστασης που βρίσκεται το υποκείμενο στην περίπτωση που η Δυναμική του Περιβάλλοντος είναι **γνωστή** και για **συγκεκριμένες πολιτικές**. Έτσι, αν έχουμε στη διάθεση μας διάφορες πολιτικές, η συνάρτηση αξίας κάτω από κάθε πολιτική μας βοηθάει στο να τις αξιολογήσουμε και να αποφασίσουμε ποια είναι η «καλύτερη». Πότε όμως θα πρέπει να θεωρούμε ότι μια πολιτική υπερέχει έναντι μιας άλλης; Απάντηση σε αυτό το ερώτημα θα μας δώσει ο Ορισμός 3.1.

Ορισμός 3.1 (Διάταξη Πολιτικών)
Μια πολιτική π λέγεται ότι είναι καλύτερη ή ίση μιας άλλης πολιτικής π' αν και μόνο αν η συνάρτηση αξίας κάτω από την πολιτική π, είναι μεγαλύτερη ή ίση της συνάρτησης αξίας της κάτω από την πολιτική π':

$$\pi \geq \pi' \text{ ανν } u_\pi(s) \geq u_{\pi'}(s) \text{ για κάθε } s \in S. \tag{3.18}$$

Σύμφωνα με τον παραπάνω ορισμό, μια πολιτική θα θεωρείται καλύτερη ή ίση έναντι μιας άλλης, αν και μόνο αν η συνάρτηση αξίας της είναι μεγαλύτερη ή ίση **σε κάθε κατάσταση**. Σε Πεπερασμένες Μαρκοβιανές Διαδικασίες Αποφάσεων αποδεικνύεται

ότι **υπάρχει τουλάχιστον 1 πολιτική η οποία είναι καλύτερη ή ίση από όλες τις άλλες**. Μια τέτοια πολιτική θα ονομάζεται **βέλτιστη**.

Θεώρημα 3.1 (Ύπαρξη Βέλτιστης Πολιτικής)
Σε Πεπερασμένες Μαρκοβιανές Διαδικασίες Αποφάσεων υπάρχει **ντετερμινιστική πολιτική**, π^*, για την οποία ισχύει:

$$u_{\pi^*}(s) \geq u_\pi(s) \text{ για κάθε } s \in S \tag{3.19}$$

Απόδειξη: Η απόδειξη παραλείπεται. ■

Το Θεώρημα 3.1 μας εξασφαλίζει την ύπαρξη τουλάχιστον μιας πολιτικής, που είναι βέλτιστη. Επειδή ενδέχεται να υπάρχουν παραπάνω από μια με την ίδια ιδιότητα, με π^* θα συμβολίζουμε οποιαδήποτε πολιτική είναι βέλτιστη. Πάντως όλες οι βέλτιστες πολιτικές μοιράζονται την **ίδια** συνάρτηση αξίας, που καλείται **βέλτιστη συνάρτηση αξίας**, συμβολίζεται με u^*, και ορίζεται ως:

$$u^*(s) := \max_\pi u_\pi(s), \text{ για κάθε } s \in S \tag{3.20}$$

Επίσης, οι βέλτιστες πολιτικές μοιράζονται και τις ίδιες **βέλτιστες αξίες κατάστασης-απόφασης**, που συμβολίζονται με Q^*, και ορίζονται ως ακολούθως:

$$Q^*(s,a) := \max_\pi Q_\pi(s,a), \text{ για κάθε } s \in S \text{ και } a \in A(s). \tag{3.21}$$

Η (3.21) δίνει το αναμενόμενο κέρδος που θα εισπράξει το υποκείμενο λαμβάνοντας την απόφαση a, στην κατάσταση s και ακολουθώντας από κει και πέρα μια βέλτιστη πολιτική. Έτσι, η $Q^*(s,a)$ μπορεί να γραφεί εναλλακτικά ως εξής:

$$Q^*(s,a) = \mathbb{E}\left[R_t + \gamma u^*(S_{t+1}) \mid S_t = s, A_t = a\right] \tag{3.22}$$

Εξισώσεις του Bellman για βέλτιστες πολιτικές

Επειδή η u^* είναι συνάρτηση αξίας κάποιας πολιτικής (πιο συγκεκριμένα, κάποιας βέλτιστης πολιτικής π^*), θα πρέπει να ικανοποιεί τις εξισώσεις του Bellman, καθώς οι τελευταίες ισχύουν για οποιαδήποτε πολιτική. Συνεπώς η (3.14), αντικαθιστώντας την συνάρτηση αξίας, με την u^*, μπορεί να πάρει την ακόλουθη μορφή:

$$u^*(s) = \sum_a \pi^*(a|s) \sum_{s'} \sum_r p(s', r|s, a)\left[r + \gamma u^*(s')\right] \tag{3.23}$$

Λόγω του Θεωρήματος 3.1, η σχέση (3.23) μπορεί να πάρει μια ακόμα πιο ειδική μορφή. Αρχικά να επισημάνουμε ότι η (3.23) θα ισχύει για οποιαδήποτε βέλτιστη πολιτική

εφόσον όλες τους, μοιράζονται την ίδια συνάρτηση αξίας. Συνεπώς θα ισχύει και για τη βέλτιστη ντετερμινιστική πολιτική που μας εγγυάται το Θεώρημα 3.1. Σύμφωνα με αυτή η $\pi^*(a|s)$, θα ταυτίζεται με τη μονάδα σε κάποια απόφαση που οδηγεί την αξία κατάστασης- απόφασης σε μεγιστοποίηση, και 0 σε όλες τις υπόλοιπες. Έτσι καταλήγουμε στην παρακάτω σχέση:

$$u^*(s) = \max_a \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u^*(s')\right] \tag{3.24}$$

Η (3.24) για όλες τις τιμές του $s \in S$ συνοψίζει τις **εξισώσεις βελτιστότητας του Bellman για την u***. Το ιδιαίτερο ενδιαφέρον που παρουσιάζουν οι τελευταίες, είναι ότι έχει χαθεί η εξάρτηση τους ως προς οποιαδήποτε συγκεκριμένη πολιτική. Επιπλέον η παρουσία του max, σημαίνει πως για να βρούμε τις βέλτιστες αξίες κάθε κατάστασης, πρέπει να λύσουμε ένα **μη γραμμικό σύστημα εξισώσεων**, (σε αντίθεση με το γραμμικό σύστημα που προέκυπτε όταν υπολογίζαμε τις αξίες των καταστάσεων κάτω από συγκεκριμένες πολιτικές). Η δυσκολία στην επίλυση του παραπάνω συστήματος είτε επειδή η δυναμική του περιβάλλοντος ενδέχεται να μην είναι γνωστή (κατάρα της μοντελοποίησης), είτε επειδή το πλήθος των καταστάσεων σε πολλά προβλήματα είναι πολύ μεγάλο (κατάρα της διάστασης), θα μας οδηγήσει στην ανάπτυξη των προσεγγιστικών μεθόδων που θα δούμε αναλυτικότερα στις επόμενες ενότητες.

Ομοίως μπορούν να προκύψουν και οι **εξισώσεις βελτιστότητας του Bellman για το Q***:

$$Q^*(s, a) = \sum_{s'} \sum_r p(s', r|s, a) \Big[r + \gamma \max_{a'} Q^*(s', a')\Big] \tag{3.25}$$

Παράδειγμα 3.2 Ας θεωρήσουμε το ακόλουθο πρόβλημα. Ένας πράκτορας μπορεί να κινείται σε ένα τετραγωνικό 3×3 πλέγμα, αυτή τη φορά όμως μόνο δεξιά ($\rightarrow$) ή πάνω ($\uparrow$) (βλέπε Σχήμα 3.3). Επίσης υποθέτουμε τον επιπλέον περιορισμό ότι δεν του επιτρέπεται να επιλέξει κάποια απόφαση που θα τον οδηγήσει εκτός του πλέγματος (έτσι στις καταστάσεις A,B,F,I υπάρχει μόνο μια διαθέσιμη επιλογή). Σε αυτό το πρόβλημα υποθέτουμε επιπροσθέτως, την ύπαρξη τερματικής κατάστασης C στην οποία η μετάβαση του πράκτορα σε αυτή σηματοδοτεί την λήξη της αλληλεπίδρασης. Σκοπός μας είναι η εύρεση της συνάρτησης βέλτιστης αξίας των καταστάσεων μέσω των εξισώσεων βελτιστότητας του Bellman και στη συνέχεια μέσω αυτών, όπως θα δούμε, θα μπορούμε να υπολογίσουμε και βέλτιστες πολιτικές. Επειδή πρόκειται για πρόβλημα πεπερασμένου ορίζοντα θεωρούμε, $\gamma = 1$. Στο Σχήμα 3.3 απεικονίζονται τα σύνολα καταστάσεων (Σχήμα 3.3 (α′)), αποφάσεων για όλες τις δυνατές καταστάσεις (Σχήμα 3.3 (β′)) και αμοιβών για κάθε ζεύγος κατάστασης απόφασης (Σχήμα 3.3 (γ′)). Παραδείγματος χάριν, αν ο πράκτορας βρίσκεται στην κατάσταση D, μπορεί να κινηθεί είτε πάνω (προς την κατάσταση A), με άμεση αμοιβή $+3$ μονάδες, είτε δεξιά (προς την κατάσταση E),

A	B	C
D	E	F
G	H	I

(α′) Καταστάσεις

$\rightarrow$	$\rightarrow$	
$\uparrow$,$\rightarrow$	$\uparrow$,$\rightarrow$	$\uparrow$
$\uparrow$,$\rightarrow$	$\uparrow$,$\rightarrow$	$\uparrow$

(β′) Αποφάσεις

+10	+12	
+3,+2	+1,+8	+14
+9,+7	+8,+5	+11

(γ′) Αμοιβές

Σχήμα 3.3: Το παράδειγμα του πλέγματος με τερματική κατάσταση

λαμβάνοντας αμοιβή +2 μονάδων. Αν επίσης, λόγου χάρη, βρεθεί στην κατάσταση F, τότε θα μπορεί να μετακινηθεί μόνο πάνω (προς την κατάσταση C) λαμβάνοντας κέρδος +14 μονάδων, καθώς η απόφαση δεξιά θα το οδηγούσε εκτός πλέγματος, που δεν επιτρέπεται από τις προδιαγραφές του προβλήματος. Να σημειώσουμε επιπλέον ότι επειδή η κατάσταση C είναι τερματική, η βέλτιστη αξία της θα είναι μηδενική, δεδομένου, ότι από κει και πέρα σταματάει η αλληλεπίδραση και κατά συνέπεια οποιαδήποτε μελλοντική είσπραξη από τον πράκτορα. Λαμβάνοντας υπόψιν όλα τα παραπάνω, οι εξισώσεις βελτιστότητας του Bellman για το παραπάνω πρόβλημα μπορούν να πάρουν την ακόλουθη μορφή:

$$\begin{cases} u^*(C) = 0 \\ u^*(B) = 12 + 1 \cdot u^*(C) = 12 + 0 = 12 \\ u^*(A) = 10 + 1 \cdot u^*(B) = 10 + 12 = 22 \\ u^*(F) = 14 + 1 \cdot u^*(C) = 14 + 0 = 14 \\ u^*(I) = 11 + 1 \cdot u^*(F) = 11 + 14 = 25 \\ u^*(E) = \max\{1 + u^*(B), 8 + u^*(F)\} = \max\{13, 22\} = 22 \\ u^*(D) = \max\{3 + u^*(A), 2 + u^*(E)\} = \max\{25, 24\} = 25 \\ u^*(H) = \max\{8 + u^*(E), 5 + u^*(I)\} = \max\{30, 30\} = 30 \\ u^*(G) = \max\{9 + u^*(D), 7 + u^*(H)\} = \max\{34, 37\} = 37 \end{cases}$$

Ας δούμε τώρα πως, έχοντας τις βέλτιστες αξίες κάθε κατάστασης, μπορούμε με σχετική ευκολία να υπολογίσουμε και βέλτιστες πολιτικές. Αρκεί να δούμε σε κάθε κατάσταση, ποια ή ποιες αποφάσεις οδήγησε την συνάρτηση αξίας σε μεγιστοποίηση. Οι αποφάσεις στις καταστάσεις A, B, F και I είναι ήδη βέλτιστες εφόσον είναι οι μοναδικές. Για την κατάσταση E, η μεγιστοποίηση προέκυψε αποφασίζοντας " $\rightarrow$ ", για την κατάσταση D, αποφασίζοντας " $\uparrow$ ", ενώ για την G, αποφασίζοντας " $\rightarrow$ ". Για την κατάσταση H, παρατηρούμε ότι οποιαδήποτε από τις δυο διαθέσιμες αποφάσεις οδηγεί σε μεγιστοποίηση της u. Αυτό σημαίνει ότι **υπάρχει τουλάχιστον μια ντετερμινιστική πολιτική η οποία είναι βέλτιστη**. Στο συγκεκριμένο πρόβλημα, επειδή η H είναι η μόνη κατάσταση με την προηγούμενη ιδιότητα **υπάρχουν ακριβώς 2 ντετερμινιστικές πολιτικές που είναι βέλτιστες**. Στο Σχήμα 3.4 απεικονίζεται η συνάρτηση

22	12	0
25	22	14
37	30	25

(α′) u^*

$\rightarrow$	$\rightarrow$	
$\uparrow$	$\rightarrow$	$\uparrow$
$\rightarrow$	$\uparrow$,$\rightarrow$	$\uparrow$

(β′) π^*

Σχήμα 3.4: Συνάρτηση Βέλτιστης Αξίας και Βέλτιστες Ντετερμινιστικές Πολιτικές του Παραδείγματος 3.2

βέλτιστης τιμής και οι 2 βέλτιστες ντετερμινιστικές πολιτικές. Τέλος μπορούμε με ευκολία να αποδείξουμε πως οποιαδήποτε στοχαστική πολιτική, η οποία αποδίδει κάποια θετική πιθανότητα p στην επιλογή " $\uparrow$ " στην κατάσταση H, και $1-p$ στην επιλογή " $\rightarrow$ " στην ίδια κατάσταση, (και ταυτόχρονα ταυτίζεται με τις βέλτιστες ντετερμινιστικές πολιτικές του Σχήματος 3.4(β′) σε όλες τις υπόλοιπες καταστάσεις) είναι και εκείνη βέλτιστη. Πράγματι έστω π' μια τέτοια πολιτική, και u' η συνάρτηση αξίας της. Τότε:

- $u'(s) = u^*(s)$, για κάθε $s \in S \setminus \{H\}$
- $u'(H) = p(8+22) + (1-p)(8+22) = 30 = u^*(H)$

Συνεπώς, $u'(s) = u^*(s)$, για κάθε $s \in S$ και επομένως η στοχαστική πολιτική π′ είναι επίσης βέλτιστη. ■

Όπως θα έγινε και περισσότερο αντιληπτό μέσω του Παραδείγματος 3.2, η γνώση της u^* μπορεί να μας οδηγήσει με σχετική ευκολία στην εξεύρεση βέλτιστων πολιτικών, ο οποίος είναι άλλωστε ο ύψιστος στόχος των προβλημάτων Ενισχυτικής Μάθησης. Δυστυχώς η απλότητα με την οποία οδηγηθήκαμε στην εύρεση της u^* στο παράδειγμα μέσω των εξισώσεων Βελτιστότητας του Bellman δεν γενικεύεται στα περισσότερα προβλήματα που συναντάμε στην πράξη. Αντιθέτως θα λέγαμε με ευκολία, ότι αποτελεί πρόκληση ακόμα και σε περιπτώσεις ακριβούς μοντέλου για το περιβάλλον. Αυτό συμβαίνει κυρίως, λόγω της περιορισμένης υπολογιστικής μας δυνατότητας να διαχειριστούμε μεγάλα σύνολα καταστάσεων. Το γνωστό σε όλους, παιχνίδι τάβλι, είναι ένα αρκετά χαρακτηριστικό παράδειγμα για τον παραπάνω ισχυρισμό. Στην προσπάθεια μας να λύσουμε τις εξισώσεις βελτιστότητας του Bellman για την u^* θα ερχόμασταν αντιμέτωποι με περίπου 10^{20} δυνατές καταστάσεις. Θα χρειαζόντουσαν χιλιάδες χρόνια ακόμα και από τους πιο σύγχρονους υπολογιστές για έναν τόσο απαιτητικό υπολογισμό. Πάντως, η εύρεση της u^* όταν η δυναμική του περιβάλλοντος είναι γνωστή, μπορεί να γίνει, έστω θεωρητικά μέσω μεθόδων από τον **Δυναμικό Προγραμματισμό**. Σκοπός της παρούσας εργασίας, ωστόσο, δεν είναι η ανάπτυξη των κλασσικών αναλυτικών μεθόδων που συναντάμε στον Δυναμικό Προγραμματισμό. Αντιθέτως, σκοπός μας είναι η ανάπτυξη **προσεγγιστικών μεθόδων** τόσο όταν η δυναμική του περιβάλλοντος είναι γνωστή (Ενότητα 3.4) όσο και άγνωστη (Ενότητες 3.5-3.8).

3.4 Προσεγγιστικές Μέθοδοι στον Δυναμικό Προγραμματισμό

Ο όρος Δυναμικός Προγραμματισμός παραπέμπει σε όλες εκείνες τις μεθόδους, που σχετίζονται με τον υπολογισμό βέλτιστων συναρτήσεων αξίας και πολιτικών όταν η δυναμική του περιβάλλοντος είναι γνωστή. Παρά το γεγονός ότι υπάρχουν προβλήματα στα οποία είναι δυνατή ακόμα και η εύρεση αναλυτικής λύσης για τις βέλτιστες συναρτήσεις αξίας, οι κλασσικοί αλγόριθμοι του Δυναμικού Προγραμματισμού είναι περιορισμένης χρησιμότητας στην Ενισχυτική Μάθηση λόγω της πολύ ισχυρής υπόθεσης ενός τέλειου μοντέλου. Παρ΄ όλα αυτά η θεωρητική τους αξία είναι κομβικής σημασίας καθώς οργανώνουν τις βασικές αρχές και το πλαίσιο που χρειάζεται το πρόβλημα της προσέγγισης βέλτιστων λύσεων μέσω της χρήσης των συναρτήσεων αξίας. Στην παρούσα ενότητα θα εστιάσουμε σε κάποιους προσεγγιστικούς αλγόριθμους για την εξεύρεση βέλτιστων πολιτικών παραμένοντας στην ισχυρή υπόθεση ενός τέλειου μοντέλου.

3.4.1 Αξιολόγηση Πολιτικής

Είδαμε σε προηγούμενη ενότητα πως για συγκεκριμένη πολιτική π, και σε προβλήματα που η δυναμική του περιβάλλοντος είναι γνωστή, μπορούμε να υπολογίσουμε την συνάρτηση αξίας της πολιτικής λύνοντας ένα γραμμικό σύστημα $|S| \times |S|$. Ας δούμε τώρα έναν εναλλακτικό τρόπο που μπορούμε να υπολογίσουμε την συνάρτηση αξίας της πολιτικής, προσεγγιστικά, **χρησιμοποιώντας τις εξισώσεις του Bellman ως κανόνα ανανέωσης**. Για τους σκοπούς μας, αυτός ίσως είναι πιο κατάλληλος, καθώς θα αναδείξει κάποια βασικά στοιχεία που χρειαζόμαστε και στην περίπτωση όπου η δυναμική του περιβάλλοντος είναι άγνωστη.

Έστω ότι έχουμε στη διάθεση μας μια ακολουθία από διαδοχικές προσεγγίσεις της u_π, $u_0, u_1, \ldots$ καθεμία από τις οποίες αποτελεί συνάρτηση από το S στο $\mathbb{R}$, οι οποίες ικανοποιούν τις ακόλουθες ιδιότητες:
1. Η αρχική προσέγγιση u_0 επιλέγεται αυθαίρετα (με εξαίρεση ότι στις τερματικές καταστάσεις, αν υπάρχουν, πρέπει να μηδενίζεται).
2. Για $k \geq 1$ ακολουθούν τον κανόνα ανανέωσης:

$$u_{k+1}(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u_k(s') \right] \text{ για κάθε } s \in S. \tag{3.26}$$

Τότε μπορεί να αποδειχθεί ότι κάτω από τις ίδιες συνθήκες ύπαρξης της u_π, η ακολουθία συναρτήσεων $\{u\}_{k=0}^{\infty}$ συγκλίνει στην u_π. Ο παραπάνω αλγόριθμος καλείται **επαναληπτική αξιολόγηση πολιτικής** (**iterative policy evaluation**) και συνοψίζεται στον ψευδοκώδικα του Πίνακα 3.1. Ο τρόπος λειτουργίας του αλγορίθμου είναι ο ακόλουθος: Σε κάθε βήμα η συνάρτηση αξίας ανανεώνεται σύμφωνα με την εξίσωση

του Bellman χρησιμοποιώντας τις προσεγγίσεις του προηγούμενου βήματος. Όταν ανανεωθεί η αξία όλων των καταστάσεων ολοκληρώνεται μια **σάρωση** του χώρου καταστάσεων. Ενδεχομένως να χρειαστούν αρκετές σαρώσεις μέχρι η προσέγγιση να είναι «κοντά» στην u_π. Ένας τρόπος να ελέγξουμε ότι η τρέχουσα προσέγγιση είναι ικανοποιητική είναι να μην υπάρχουν μεγάλες αποκλίσεις μεταξύ των δυο τελευταίων διαδοχικών προσεγγίσεων. Επειδή η προσεγγίσεις κάθε βήματος αποτελούν συναρτήσεις, ο πιο φυσιολογικός τρόπος στο να μετράμε διαφορές μπορεί να γίνει μέσω της **άπειρο νόρμας** $||u_{k+1} - u_k||_\infty := \max_{s \in S} |u_{k+1}(s) - u_k(s)|$. Επειδή, γενικά, η σύγκλιση επιτυγχάνεται μόνο στο όριο, η διαδικασία επιθυμούμε να τερματίσει όταν η μέγιστη απόλυτη διαφορά των αξιών είναι μικρότερη από μια καθορισμένη σταθερά θ που ρυθμίζει το επίπεδο ακρίβειας της εκτίμησης.

Πίνακας 3.1 Ο Αλγόριθμος Επαναληπτικής Αξιολόγησης Πολιτικής

Είσοδος: Μια πολιτική π, προς αξιολόγηση, και η παράμετρος $\theta > 0$, η οποία καθορίζει την ακρίβεια της εκτίμησης.

1. Αρχικοποίησε την πρώτη προσέγγιση $V(s)$ για κάθε $s \in S$, αυθαίρετα, εκτός από τις τερματικές καταστάσεις στις οποίες θέσε $V(s_{term}) \leftarrow 0$.
2. Θέσε $\Delta \leftarrow 0$, όπου Δ η μέγιστη απόλυτη διαφορά των 2 διαδοχικών προσεγγίσεων
3. Για όλες τις καταστάσεις $s \in S$ επανέλαβε:
$u \leftarrow V(s)$
$V(s) \leftarrow \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma u(s')]$
$\Delta \leftarrow \max\{\Delta, |u - V(s)|\}$
4. Αν $\Delta \geq \theta$ επίστρεψε στο 3, διαφορετικά τερμάτισε.

Έξοδος: Μια συνάρτηση αξίας $V \approx u_\pi$

Παράδειγμα 3.3 Ας υποθέσουμε ότι ένας πράκτορας κινείται στο τετραγωνικό πλέγμα του Σχήματος 3.5. Αυτή τη φορά θεωρούμε 2 τερματικές καταστάσεις (σημειωμένες με κόκκινο). Το υποκείμενο σε όλες τις καταστάσεις (εκτός των τερματικών) έχει την ευχέρεια να κινηθεί **αριστερά** ($\leftarrow$), **δεξιά** ($\rightarrow$), **πάνω** ($\uparrow$) ή **κάτω** ($\downarrow$) και η του κάθε απόφαση οδηγεί **ντετερμινιστικά** τον πράκτορα στις αντίστοιχες καταστάσεις. Θεωρούμε επιπλέον αμοιβή $R = -2$, για όλες τις μεταβάσεις μέχρι να φτάσει σε κάποια από τις 2 τερματικές καταστάσεις. Ο συντελεστής αποπληθωρισμού είναι και για αυτό το πρόβλημα $\gamma = 1$ (λόγω της ύπαρξης τερματικών καταστάσεων). Υποθέτοντας ότι ο πράκτορας ακολουθεί την **ομοιόμορφη τυχαία πολιτική** θα εκτιμήσουμε την αξία της συγκεκριμένης πολιτικής με βάση τον Αλγόριθμο Επαναληπτικής Αξιολόγησης Πολιτικής του Πίνακα 3.1. Για αυτό το σκοπό υλοποιήσαμε τον iterative_policy_evaluation

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

Σχήμα 3.5: Το παράδειγμα του πλέγματος με 2 τερματικές καταστάσεις

στη Γλώσσα R (βλέπε Παράρτημα), και καταλήξαμε στις ακόλουθες προσεγγίσεις:

+0.00	+0.00	+0.00	+0.00
+0.00	+0.00	+0.00	+0.00
+0.00	+0.00	+0.00	+0.00
+0.00	+0.00	+0.00	+0.00

(α′) $k = 0$

+0.00	−2.00	−2.00	−2.00
−2.00	−2.00	−2.00	−2.00
−2.00	−2.00	−2.00	−2.00
−2.00	−2.00	−2.00	+0.00

(β′) $k = 1$

+0.00	−3.50	−4.00	−4.00
−3.50	−4.00	−4.00	−4.00
−4.00	−4.00	−4.00	−3.50
−4.00	−4.00	−3.50	+0.00

(γ′) $k = 2$

+0.00	−4.87	−5.87	−6.00
−4.87	−5.75	−6.00	−5.87
−5.87	−6.00	−5.75	−4.87
−6.00	−5.87	−4.80	+0.00

(δ′) $k = 3$

+0.00	−12.2	−16.7	−17.9
−12.2	−15.4	−16.8	−16.7
−16.7	−16.8	−15.4	−12.2
−17.9	−16.7	−12.2	+0.00

(ε′) $k = 10$

+0.00	−28.0	−40.0	−44.0
−28.0	−36.0	−40.0	−40.0
−40.0	−40.0	−36.0	−28.0
−44.0	−40.0	−28.0	+0.00

(ϝ′) $k \geq 200$

Σχήμα 3.6: Προσεγγίσεις τις συνάρτησης αξίας του Παραδείγματος 3.3 για διάφορες τιμές του k. Για $k \geq 200$ ο αλγόριθμος συγκλίνει στην πραγματική αξία u_π της τυχαίας ομοιόμορφης πολιτικής.

3.4.2 Βελτίωση Πολιτικής

Ο λόγος που υπολογίζουμε την συνάρτηση αξίας μιας πολιτικής, είναι για να μας βοηθήσει να βρούμε καλύτερες. Έστω ότι έχουμε προσδιορίσει τη συνάρτηση αξίας u_π, μιας αυθαίρετης ντετερμινιστικής πολιτικής π. Για κάποια κατάσταση $s \in S$, θα θέλαμε να ξέρουμε κατά πόσο μια απόφαση $a \neq \pi(s)$ θα οδηγούσε ή όχι, σε βελτίωση της πολιτικής. Ένας τρόπος να το ελέγξουμε είναι συγκρίνοντας τις ποσότητες $Q_\pi(s, a)$ και $u_\pi(s)$. Η ποσότητα $Q_\pi(s, a)$ εκφράζει την αξία που θα λάβει το υποκείμενο επιλέγοντας

την απόφαση a στην κατάσταση s και από κει και πέρα ακολουθώντας την υπάρχουσα πολιτική π. Από την άλλη μεριά η $u_\pi(s)$ εκφράζει την αξία που θα λάβει το υποκείμενο ακολουθώντας την πολιτική π από την κατάσταση s. Διαισθητικά περιμένουμε ότι αν $Q_\pi(s,a) > u_\pi(s)$ τότε είναι καλύτερο το υποκείμενο στην κατάσταση s να επιλέξει την απόφαση a και στη συνέχεια να «παραμείνει πιστό» στην πολιτική π παρά να ακολουθούσε την πολιτική π από την αρχή. Στην πραγματικότητα αυτό αποτελεί ειδική περίπτωση ενός γενικότερου αποτελέσματος που καλείται **Θεώρημα Βελτίωσης Πολιτικής (Policy Improvement Theorem)**.

Θεώρημα 3.1 (Θεώρημα Βελτίωσης Πολιτικής)
Έστω πολιτικές π, και π' για τις οποίες ισχύει:

$$Q_\pi(s,\pi'(s)) \geq u_\pi(s) \text{ για κάθε } s \in S. \tag{3.27}$$

Τότε η πολιτική π', είναι καλύτερη ή ίση της πολιτικής π, δηλαδή:

$$u_{\pi'}(s) \geq u_\pi(s) \text{ για κάθε } s \in S. \tag{3.28}$$

Απόδειξη:

$$\begin{aligned}
u_\pi(s) &\leq Q_\pi(s,\pi'(s)) \\
&= \mathbb{E}_\pi\left[R_{t+1} + \gamma G_{t+1} \mid S_t = s, A_t = \pi'(s)\right] \\
&= \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma u_\pi(S_{t+1}) \mid S_t = s, A_t = \pi'(s)\right] \\
&= \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma u_\pi(S_{t+1}) \mid S_t = s\right] \\
&\leq \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma Q_\pi(S_{t+1},\pi'(S_{t+1})) \mid S_t = s\right] \\
&= \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma\, \mathbb{E}_{\pi'}\left[R_{t+2} + \gamma u_\pi(S_{t+2}) \mid S_t, A_{t+1} = \pi'(S_{t+1})\right] \mid S_t = s\right] \\
&= \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 u_\pi(S_{t+2}) \mid S_t = s\right] \\
&\vdots \\
&\leq \mathbb{E}_{\pi'}\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 u_\pi(S_{t+3}) \mid S_t = s\right] \\
&\vdots \\
&\leq \mathbb{E}_{\pi'}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s\right] \\
&= u_{\pi'}(s)
\end{aligned}$$

■

Να σημειώσουμε επίσης ότι αν υπάρχει κατάσταση $s_0 \in S$ η οποία ικανοποιεί την (3.27) με γνήσια ανισότητα, τότε για την ίδια κατάσταση θα πρέπει να ικανοποιείται και η (3.28) με γνήσια ανισότητα. Σε αυτή τη περίπτωση, αν δηλαδή για μια πολιτική π' ισχύει

$u_{\pi'}(s) \geq u_\pi(s)$ για κάθε $s \in S$. και υπάρχει s_0 για την οποία $u_{\pi'}(s_0) > u_\pi(s_0)$ λέμε ότι η πολιτική π΄ είναι **(γνησίως) καλύτερη** από την π και γράφουμε $\pi' > \pi$.

Ας δούμε τώρα πως μπορεί να εφαρμοστεί το Θεώρημα 3.1 για τις 2 πολιτικές που αναπτύξαμε στις αρχές της ενότητας. Αν π, μια αυθαίρετη ντετερμινιστική πολιτική, και π' η πολιτική που ταυτίζεται με την π σε όλες τις καταστάσεις, εκτός μιας κατάστασης s_0, για την οποία ισχύει $\pi'(s_0) = a \neq \pi(s_0)$, και επιπλέον $Q_\pi(s_0, \pi'(s_0)) > u_\pi(s_0)$, τότε:

- Αν $s \neq s_0$ τότε $Q_\pi(s, \pi'(s)) = Q_\pi(s, \pi(s)) = u_\pi(s) \geq u_\pi(s)$, και συνεπώς η (3.27) ικανοποιείται (και μάλιστα με ισότητα).
- Αν $s = s_0$ τότε $Q_\pi(s, \pi'(s)) = Q_\pi(s_0, \pi'(s_0)) > u_\pi(s_0) = u_\pi(s)$

Επομένως, η (3.27) ικανοποιείται για κάθε $s \in S$ και μάλιστα επειδή για μια κατάσταση αυτό συμβαίνει με γνήσια ανισότητα αυτό σημαίνει πως η π' θα είναι γνησίως καλύτερη από την π. Με αυτόν τον τρόπο μπορούμε, έχοντας μια συγκεκριμένη πολιτική π και την αξία της, να την βελτιώσουμε παρεμβαίνοντας **σε μια μόνο** κατάσταση και αποφασίζοντας μια δράση με μεγαλύτερη αξία (όταν αυτή υπάρχει).

Η φυσική επέκταση του τελευταίου είναι η παρέμβαση να γίνει σε **όλες τις δυνατές καταστάσεις** και για όλες τις **δυνατές αποφάσεις** επιλέγοντας την δράση **με τη μεγαλύτερη αξία του ζεύγους κατάστασης απόφασης** $Q_\pi(s, a)$. Μια τέτοια νέα πολιτική π' θα ονομάζεται **πλεονεκτική** ή **greedy ως προς την αρχική πολιτική π**. Πιο συγκεκριμένα για την πλεονεκτική πολιτική π' θα ισχύει:

$$\pi'(s) := \operatorname*{argmax}_{a \in A(s)} Q_\pi(s, a) \text{ για κάθε } s \in S. \tag{3.29}$$

$$= \operatorname*{argmax}_{a \in A(s)} \mathbb{E}\left[R_{t+1} + \gamma u_\pi(S_{t+1}) \mid S_t = s, A_t = \pi'(s)\right] \text{ για κάθε } s \in S. \tag{3.30}$$

$$= \operatorname*{argmax}_{a \in A(s)} \sum_{s'} \sum_{r} p(s', r|s, a) \left[r + \gamma u_\pi(s')\right] \text{ για κάθε } s \in S. \tag{3.31}$$

Η πλεονεκτική πολιτική π' ικανοποιεί εκ κατασκευής τις προϋποθέσεις του Θεωρήματος 3.1, και έτσι γνωρίζουμε ότι η π' θα είναι καλύτερη ή ίση της αρχικής πολιτικής π. Η διαδικασία κατά την οποία μια νέα πολιτική βελτιώνεται σε σχέση με μια αρχική πολιτική, κάνοντας την πλεονεκτική ως προς την συνάρτηση αξίας της αρχικής, ονομάζεται **βελτίωση πολιτικής**. Να σημειώσουμε πως η κατασκευή της πλεονεκτικής πολιτικής μπορεί να επεκταθεί και στην περίπτωση όπου η αρχική πολιτική είναι **στοχαστική**.

Έστω τώρα ότι η νέα πολιτική π' είναι καλύτερη ή ίση αλλά όχι γνησίως καλύτερη της αρχικής πολιτικής π. Τότε θα ισχύει $u_\pi = u'_\pi$, και επομένως για κάθε $s \in S$:

$$u_{\pi'}(s) = \max_{a \in A(S)} \sum_{s'} \sum_{r} p(s', r|s, a) \left[r + \gamma u_\pi(s')\right] \quad \text{(από (3.31))}$$

$$= \max_{a \in A(S)} \sum_{s'} \sum_{r} p(s', r|s, a) \left[r + \gamma u_{\pi'}(s')\right]$$

Η τελευταία εξίσωση, ταυτίζεται με την εξίσωση βελτιστότητας του Bellman, και συνεπώς θα πρέπει να ισχύει $u_\pi = u_{\pi'} = u^*$. Με άλλα λόγια, τόσο η π, όσο και η π' είναι ήδη βέλτιστες πολιτικές. Έτσι συμπεραίνουμε ότι η βελτίωση πολιτικής πρέπει να μας δίνει μια αυστηρή βελτίωση στην πολιτική εκτός από την περίπτωση όπου η αρχική πολιτική είναι ήδη βέλτιστη.

Παράδειγμα 3.4 Ας επιστρέψουμε στο τετραγωνικό πλέγμα του Παραδείγματος 3.3. Εφαρμόζοντας τον Αλγόριθμο Επαναληπτικής Αξιολόγησης Πολιτικής υπολογίσαμε ότι η συνάρτηση αξίας της ομοιόμορφης τυχαίας πολιτικής είναι αυτή που παρουσιάζεται στο Σχήμα 3.7. Ας δούμε τώρα πως μπορούμε να υπολογίσουμε με βάση την (3.29) κάποια

$+0.00$	-28.0	-40.0	-44.0
-28.0	-36.0	-40.0	-40.0
-40.0	-40.0	-36.0	-28.0
-44.0	-36.0	-28.0	$+0.00$

Σχήμα 3.7: Η συνάρτηση αξίας της τυχαίας ομοιόμορφης πολιτικής του Παραδείγματος 3.3

πλεονεκτική πολιτική π', ως προς την τυχαία ομοιόμορφη πολιτική π. Ξεκινάμε με την κατάσταση 1 υπολογίζοντας όλες τις αξίες κατάστασης-απόφασης:

$$Q_\pi(1, \leftarrow) = -2 + u(0) = -2 + 0 = -2$$
$$Q_\pi(1, \uparrow) = -2 + u(1) = -2 - 28 = -30$$
$$Q_\pi(1, \rightarrow) = -2 + u(2) = -2 - 40 = -42$$
$$Q_\pi(1, \downarrow) = -2 + u(5) = -2 - 36 = -38$$

Συνεπώς, $\pi'(1) = \underset{a}{\text{argmax}}\ Q_\pi(1, a) =$ " $\leftarrow$ ". Δηλαδή το καλύτερο που μπορεί να κάνει να κάνει ο πράκτορας βρισκόμενος στην κατάσταση 1, είναι να επιλέξει αριστερά. Αυτό είναι απολύτως λογικό εφόσον αυτή η απόφαση θα τον οδηγήσει στην τερματική κατάσταση 0, και έτσι από κει και πέρα δεν θα λαμβάνει αρνητικά κέρδη περιπλανόμενος στις υπόλοιπες καταστάσεις του πλέγματος. Συνεχίζοντας με την κατάσταση 2, βρίσκουμε:

$$Q_\pi(2, \leftarrow) = -2 + u(1) = -2 - 28 = -30$$
$$Q_\pi(2, \uparrow) = -2 + u(2) = -2 - 40 = -42$$
$$Q_\pi(2, \rightarrow) = -2 + u(3) = -2 - 44 = -46$$
$$Q_\pi(2, \downarrow) = -2 + u(6) = -2 - 40 = -42$$

Επομένως, αφού $\pi'(2) = \underset{a}{\text{argmax}}\ Q_\pi(2, a) =$ " $\leftarrow$ ", το καλύτερο που μπορεί να κάνει

ο πράκτορας βρισκόμενος στη κατάσταση 2 είναι πάλι να κινηθεί προς τα αριστερά. Για την κατάσταση 3, ομοίως βρίσκουμε:

$$Q_\pi(3,\leftarrow) = -2 + u(2) = -2 - 40 = -42$$
$$Q_\pi(3,\uparrow) = -2 + u(3) = -2 - 44 = -46$$
$$Q_\pi(3,\rightarrow) = -2 + u(3) = -2 - 44 = -46$$
$$Q_\pi(3,\downarrow) = -2 + u(7) = -2 - 40 = -42$$

Παρατηρούμε τώρα ότι η αξία κατάστασης απόφασης $Q_\pi(3,a)$, οδηγείται σε μεγιστοποίηση. τόσο επιλέγοντας την δράση αριστερά, όσο και την δράση κάτω. Αυτό σημαίνει ότι η πλεονεκτική πολιτική π' δεν είναι μοναδική. Συνεχίζοντας και με τις υπόλοιπες καταστάσεις μπορούμε να οδηγηθούμε στον Σχήμα 3.8 όπου παρουσιάζονται όλες οι πλεονεκτικές πολιτικές του πράκτορα ως προς την τυχαία ομοιόμορφη πολιτική. Ας δο-

	$\leftarrow$	$\leftarrow$	$\leftarrow,\downarrow$
$\uparrow$	$\leftarrow,\uparrow$	$\leftarrow,\downarrow$	$\downarrow$
$\uparrow$	$\uparrow,\rightarrow$	$\downarrow,\rightarrow$	$\downarrow$
$\uparrow,\rightarrow$	$\rightarrow$	$\rightarrow$	

Σχήμα 3.8: Πλεονεκτικές πολιτικές ως προς την τυχαία ομοιόμορφη πολιτική στο παράδειγμα του πλέγματος με 2 τερματικές καταστάσεις.

ύμε τώρα, τι επιρροή είχε η βελτίωση πολιτικής στην συνάρτηση αξίας. Υπολογίζοντας την συνάρτηση αξίας οποιασδήποτε πλεονεκτικής πολιτικής π' (λύνοντας, λόγου χάρη, το γραμμικό σύστημα που προκύπτει μέσω των εξισώσεων του Bellman) καταλήγουμε στον πίνακα του Σχήματος 3.9. Παρατηρούμε η συνάρτηση αξίας $u_{\pi'}$ μιας οποιασδήποτε

$+0$	-2	-4	-6
-2	-4	-6	-4
-4	-6	-4	-2
-6	-4	-2	$+0$

Σχήμα 3.9: Συνάρτηση Αξίας οποιασδήποτε πλεονεκτικής πολιτικής π'

τέτοιας πολιτικής, παίρνει τις τιμές $-2, -4$ ή -6 για κάθε $s \in S$, σε αντίθεση με την u_π η οποία είναι το πολύ -28. Συνεπώς μπορούμε να επαληθεύσουμε εύκολα ότι η συνθήκη $u_{\pi'}(s) \geq u_\pi(s)$ για κάθε $s \in S$ που μας εγγυάται το Θεώρημα Βελτίωσης Πολιτικής ικανοποιείται.

Αν τώρα επιχειρήσουμε να βρούμε μια νέα πολιτική π'', που αυτή τη φορά είναι πλεονεκτική ως προς την π' (με σκοπό να τη βελτιώσουμε ακόμα παραπάνω), θα παρατηρήσουμε ότι δεν θα υπάρχει κάποια αυστηρή βελτίωση στην συνάρτηση αξίας. Αυτό σημαίνει πως

η πολιτική π' **είναι ήδη βέλτιστη**. Παρά το γεγονός ότι στη συγκεκριμένη περίπτωση χρειάστηκε 1 μόνο εφαρμογή της βελτίωσης πολιτικής για να οδηγηθούμε σε κάποια βέλτιστη, γενικά αυτό που είναι πάντα σίγουρο, είναι απλώς μια βελτίωση σε σχέση με την αρχική. ■

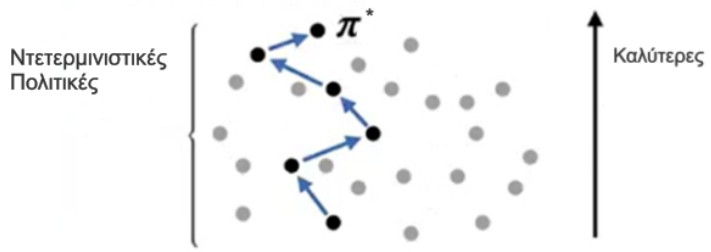
3.4.3 Επανάληψη Πολιτικής

Έχοντας μια πολιτική π, και εφαρμόζοντας την βελτίωση πολιτικής, βρίσκουμε μια καλύτερη (ή ίση) πολιτική π'. Στη συνέχεια μπορούμε να υπολογίσουμε την $u_{\pi'}$, και με βάση αυτή να εφαρμόσουμε εκ νέου την βελτίωση πολιτικής στην π', με σκοπό να βρούμε μια ακόμα καλύτερη πολιτική π''. Συνεχίζοντας καθ΄ αυτόν τρόπο, μπορούμε να φτιάξουμε μια μονότονη ακολουθία με βελτιωμένες πολιτικές και συναρτήσεις αξίας ως εξής:

$$\pi_0 \xrightarrow{A} u_{\pi_0} \xrightarrow{B} \pi_1 \xrightarrow{A} u_{\pi_1} \xrightarrow{B} \pi_2 \xrightarrow{A} u_{\pi_2} \xrightarrow{B} \ldots \xrightarrow{B} \pi^* \xrightarrow{A} u^*,$$

όπου το σύμβολο $\xrightarrow{A}$ δηλώνει μια αξιολόγηση πολιτικής, ενώ το σύμβολο $\xrightarrow{B}$ δηλώνει μια βελτίωση πολιτικής. Κάθε πολιτική θα αποτελεί μια γνήσια βελτίωση της προηγούμενης, εκτός από την περίπτωση όπου είναι ήδη βέλτιστη. Επειδή μια Πεπερασμένη Μαρκοβιανή Διαδικασία Αποφάσεων έχει πεπερασμένο αριθμό ντετερμινιστικών πολιτικών (το πλήθος τους ταυτίζεται με το γινόμενο των διαθέσιμων επιλογών για όλες τις δυνατές καταστάσεις) η διαδικασία θα πρέπει να συγκλίνει σε κάποια βέλτιστη πολιτική και κάποια βέλτιστη συνάρτηση αξίας σε πεπερασμένο αριθμό βημάτων.

Ο παραπάνω τρόπος στο να βρίσκουμε μια βέλτιστη πολιτική ονομάζεται **επανάληψη πολιτικής** (**policy iteration**). Στον Πίνακα 3.2 παρουσιάζεται ο Αλγόριθμος Επανάληψης Πολιτικής σε μορφή ψευδοκώδικα.



Σχήμα 3.10: Απλοποιημένη γεωμετρική αναπαράσταση της μονότονης εύρεσης καλύτερων πολιτικών στον χώρο όλο των ντετερμινιστικών πολιτικών μιας Πεπερασμένης ΜΔΑ σύμφωνα με την επανάληψη πολιτικής.

Πίνακας 3.2 Ο Αλγόριθμος Επανάληψης Πολιτικής

---

1. Αρχικοποίηση: Αρχικοποίησε αυθαίρετα $V(s) \in \mathbb{R}$ (εκτός αν υπάρχουν τερματικές καταστάσεις όπου θέσε $V(s_{term}) \leftarrow 0$) και $\pi(s) \in A(s)$ για κάθε $s \in S$.

2. Επιλογή παραμέτρου: Επίλεξε την παράμετρο θ, η οποία ρυθμίζει την επιθυμητή ακρίβεια στο στάδιο της επαναληπτικής αξιολόγησης.

3. Στάδιο Επαναληπτικής **Αξιολόγησης** Πολιτικής
i. Θέσε $\Delta \leftarrow 0$, όπου Δ η μέγιστη απόλυτη διαφορά των 2 διαδοχικών προσεγγίσεων.
ii. Για όλες τις καταστάσεις $s \in S$ επανέλαβε:
$u \leftarrow V(s)$
$V(s) \leftarrow \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u(s')\right]$
$\Delta \leftarrow \max\{\Delta, |u - V(s)|\}$
iii. Αν $\Delta \geq \theta$ επίστρεψε στο *i*, διαφορετικά συνέχισε στο 4.

4. Στάδιο **Βελτίωσης** Πολιτικής
i. Θέσε *σταθερότητα-πολιτικής* $\leftarrow$ ΑΛΗΘΗΣ
ii. Για όλες τις καταστάσεις $s \in S$ επανέλαβε:
παλαιά-δράση$(s) \leftarrow \pi(s)$
$\pi(s) \leftarrow \underset{a}{\text{argmax}} \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma V(s')\right]$
iii Αν *παλαιά-δράση* $\neq \pi$ τότε *σταθερότητα-πολιτικής* $\leftarrow$ ΨΕΥΔΗΣ
iv Αν *σταθερότητα-πολιτικής* τερμάτισε, διαφορετικά πήγαινε πίσω στο 3.

Έξοδος: Μια συνάρτηση αξίας $V \approx u^*$, και μια πολιτική $\pi \approx \pi^*$.

Παράδειγμα 3.5 Ας υποθέσουμε ότι ένας πράκτορας κινείται στο τετραγωνικό πλέγμα του Σχήματος 3.11, αυτή τη φορά με 1 τερματική κατάσταση (σημειωμένη με κόκκινο) και επιπλέον 6 ειδικές καταστάσεις (σημειωμένες με γκρι). Πιο συγκεκριμένα η μετάβαση του πράκτορα σε κάποια από τις ειδικές καταστάσεις επιφέρει άμεση αμοιβή $R = -10$ μονάδων, ενώ για όλες τις υπόλοιπες, η άμεση αμοιβή που θα εισπράξει είναι $R = -1$.

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

Σχήμα 3.11: Το παράδειγμα του πλέγματος με 1 τερματική και 6 ειδικές καταστάσεις.

Τα υπόλοιπα στοιχεία του προβλήματος παραμένουν ίδια με αυτά που συναντήσαμε και στο Παράδειγμα 3.4. Σκοπός μας τώρα είναι ξεκινώντας από μια αρχική πολιτική π, και εφαρμόζοντας την επανάληψη πολιτικής να καταλήξουμε σε κάποια βέλτιστη. Ας θεωρήσουμε ότι ο πράκτορας ξεκινάει με την ομοιόμορφη τυχαία πολιτική. Υλοποιώντας τον αλγόριθμο policy_iteration στην Γλώσσα R βρίσκουμε διαδοχικά:

$u_1 =$

0	−137.5	−209.6	−239.4
−129.5	−181.0	−220.7	−238.3
−194.4	−214.3	−232.0	−241.8
−217.6	−227.0	−238.2	−242.0

$\xrightarrow{B} \pi_1 =$

	←	←	←
↑	←	←	←
↑	↑	↑	↑
↑	←	←	←

$\xrightarrow{A}$

$u_2 =$

0	−1.000	−11,00	−21.00
−1.000	−2.000	−3.000	−4.000
−2.000	−3.000	−4.000	−5.000
−12.00	−13.00	−14.00	−15.00

$\xrightarrow{B} \pi_2 =$

	←	↓	↓
↑	←	←	←
↑	↑	↑	↑
↑	←	←	↑

$\xrightarrow{A}$

$u_3 =$

0	−1.000	−4,000	−5.000
−1.000	−2.000	−3.000	−4.000
−2.000	−3.000	−4.000	−5.000
−12.00	−13.00	−14.00	−6.000

$\xrightarrow{B} \pi_3 =$

	←	↓	↓
↑	←	←	←
↑	↑	↑	↑
↑	←	→	↑

$\xrightarrow{A}$

$u_4 =$

0	−1.000	−4,000	−5.000
−1.000	−2.000	−3.000	−4.000
−2.000	−3.000	−4.000	−5.000
−12.00	−13.00	−7.000	−6.000

$\xrightarrow{B} \pi_4 =$

	←	↓	↓
↑	←	←	←
↑	↑	↑	↑
↑	→	→	↑

$\xrightarrow{A}$

$u_5 =$

0	−1.000	−4,000	−5.000
−1.000	−2.000	−3.000	−4.000
−2.000	−3.000	−4.000	−5.000
−12.00	−8.000	−7.000	−6.000

$\xrightarrow{B} \pi_5 =$

	←	↓	↓
↑	←	←	←
↑	↑	↑	↑
→	→	→	↑

$\xrightarrow{A}$

$u_6 =$

0	−1.000	−4,000	−5.000
−1.000	−2.000	−3.000	−4.000
−2.000	−3.000	−4.000	−5.000
−9.000	−8.000	−7.000	−6.000

$\xrightarrow{B} \pi_6 =$

	←	↓	↓
↑	←	←	←
↑	↑	↑	↑
→	→	→	↑

$= \pi^*$

Ο αλγόριθμος ξεκινάει υπολογίζοντας την συνάρτηση αξίας, u_1, της τυχαίας ομοιόμορφης πολιτικής. Έπειτα βρίσκει την πλεονεκτική πολιτική π_1 ως προς τη u_1 εφαρμόζοντας ένα βήμα **βελτίωσης πολιτικής**. Η συνάρτηση u_2 προκύπτει μετά από **αξιολόγηση**

της πολιτικής π_1. Μπορούμε να παρατηρήσουμε την εμφανώς μεγάλη αύξηση στις τιμές u_2 σε σχέση με τη u_1 ύστερα από μόλις ένα βήμα αξιολόγησης. Ο αλγόριθμος σταματά στην προσέγγιση π_6 εφόσον η τελευταία δεν παρουσίασε κάποια μεταβολή σε σχέση με την προηγούμενη (την π_5). Αυτό, όπως είδαμε σε προηγούμενη ενότητα, σημαίνει ότι $\pi_6 = \pi^*$, δηλαδή ότι η π_6 είναι **βέλτιστη**. Αυτό άλλωστε φαίνεται και στο ταμπλό της τελευταίας: Ο πράκτορας μένοντας πιστός στην πολιτική π_6 αποφεύγει την είσοδο του σε ζημιογόνες καταστάσεις, ακολουθώντας ένα «ασφαλές» μονοπάτι.

Η συνάρτηση αξίας u_6 ταυτίζεται με την βέλτιστη συνάρτηση αξίας u^*. Η ερμηνεία της σε αυτό το παράδειγμα είναι άμεση: για κάθε κατάσταση εκφράζει τον ακριβή αριθμό βημάτων που απαιτείται για την είσοδο του πράκτορα στην τερματική κατάσταση πολλαπλασιασμένη επί -1 που είναι το κόστος μετάβασης σε μη επιζήμιες καταστάσεις (τις οποίες θα αποφύγει εφόσον θα ακολουθήσει την βέλτιστη πολιτική π_6). ■

3.4.4 Επανάληψη Αξίας

Η επανάληψη πολιτικής αποτελεί έναν αποτελεσματικό τρόπο, ξεκινώντας από μια αρχική πολιτική, και συνεχίζοντας διαρκώς με βήματα αξιολόγησης και βελτίωσης , να προσεγγίσουμε κάποια βέλτιστη ύστερα από έναν εύλογο αριθμό επαναλήψεων. Ένα μειονέκτημα της , ωστόσο, είναι ότι σε κάθε επανάληψη της περιλαμβάνει την αξιολόγηση πολιτικής η οποία μπορεί να αποβεί υπολογιστικά ακριβή, λόγω την πολλαπλών σαρώσεων του χώρου καταστάσεων που γενικά απαιτεί μέχρι να φτάσει σε μια ικανοποιητική σύγκλιση. Στην πραγματικότητα, σε πολλές περιπτώσεις, το βήμα βελτίωσης είναι δυνατόν να περικοπεί με πολλούς τρόπους χωρίς να χάνονται οι εγγυήσεις για την σύγκλιση του αλγορίθμου επανάληψης πολιτικής.

Στην παρούσα υποενότητα θα εστιάσουμε την ειδική περίπτωση όπου η αξιολόγηση πολιτικής **σταματά μετά από 1 μόλις σάρωση**. Ο Αλγόριθμος κατά τον οποίο σε κάθε επανάληψη πραγματοποιείται ένα βήμα αξιολόγησης και ένα βελτίωσης, ονομάζεται **επανάληψη αξίας (value iteration)** και συνοψίζεται στον ψευδοκώδικα του Πίνακα 3.3. Κατά τον συγκεκριμένο αλγόριθμο η συνάρτηση αξίας ανανεώνεται σύμφωνα με τον κανόνα:

$$u_{k+1}(s) = \max_a \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u_k(s') \right], \tag{3.32}$$

για κάθε $s \in S$. Μπορεί να αποδειχθεί ότι για αυθαίρετη επιλογή της αρχικής προσέγγισης u_0, (με εξαίρεση ότι αν υπάρχουν τερματικές καταστάσεις πρέπει σε αυτές να μηδενίζεται), η ακολουθία συναρτήσεων $\{u\}_{k=0}^{\infty}$ συγκλίνει στην u^*, κάτω από τις ίδιες συνθήκες που εγγυώνται την ύπαρξη της u^*. Αν κοιτάξουμε προσεκτικά την (3.32), θα παρατηρήσουμε ότι αυτή ουσιαστικά μπορεί να προκύψει απλώς **μετατρέποντας την εξίσωση βελτιστότητας του Bellman σε κανόνα ανανέωσης**[5]. Επιπλέον,

[5] Αυτό μας θυμίζει και τον κανόνα ανανέωσης του Αλγορίθμου Αξιολόγησης Πολιτικής η οποία όμως

στον Πίνακα 3.3 μπορούμε να παρατηρήσουμε, ότι τα βήματα αξιολόγησης και βελτίωσης έχουν συγχωνευτεί σε **μια μόνο δομή επανάληψης**. Αυτό μπορεί να γίνει εφόσον η συνάρτηση αξίας ανανεώνεται συνεχώς με βάση την πλεονεκτική δράση, δηλαδή με εκείνη η οποία την οδηγεί σε μεγιστοποίηση. Ανανεώνοντας την συνάρτηση αξίας πάντα με αυτό τον τρόπο, η αναφορά στην πλεονεκτική πολιτική κάθε βήματος μπορεί να παραλειφθεί. Σε αντίθεση με τον αλγόριθμο επανάληψης πολιτικής, δεν υπάρχει αναφορά ούτε σε κάποια αρχική πολιτική, στην είσοδο του αλγορίθμου, αλλά ούτε σε οποιασδήποτε άλλη στον κανόνα ανανέωσης της συνάρτησης αξίας. Τέλος, όπως και στους αλγορίθμους που διατυπώθηκαν, λόγω του ότι η σύγκλιση θεωρητικά επιτυγχάνεται στο όριο, η ανανέωση της συνάρτησης αξίας σταματά όταν η μέγιστη μεταβολή της ύστερα από 2 διαδοχικές σαρώσεις, είναι μικρότερη από μια πρόκαθορισμένη παράμετρο θ.

Πίνακας 3.3 Ο Αλγόριθμος Επανάληψης Αξίας

Είσοδος: Η παράμετρος $\theta > 0$, η οποία καθορίζει την ακρίβεια της εκτίμησης.

1. Αρχικοποίησε την πρώτη προσέγγιση $V(s)$ για κάθε $s \in S$, αυθαίρετα, εκτός από τις τερματικές καταστάσεις στις οποίες θέσε $V(s_{term}) \leftarrow 0$.
2. Θέσε $\Delta \leftarrow 0$, όπου Δ η μέγιστη απόλυτη διαφορά των 2 διαδοχικών προσεγγίσεων.
3. Για όλες τις καταστάσεις $s \in S$ επανέλαβε:
$u \leftarrow V(s)$
$V(s) \leftarrow \max_a \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u(s')\right]$
$\Delta \leftarrow \max\{\Delta, |u - V(s)|\}$
4. Αν $\Delta \geq \theta$ επίστρεψε στο 3, διαφορετικά συνέχισε στο 5.
5. Για κάθε $s \in S$ θέσε : $\pi(s) \leftarrow \underset{a}{\text{argmax}} \sum_{s'} \sum_r p(s', r|s, a) \left[r + \gamma u(s')\right]$

Έξοδος: Μια πολιτική $\pi \approx \pi^*$.

3.4.5 Περιορισμοί και Επεκτάσεις

Ασύγχρονος Δυναμικός Προγραμματισμός

Συνοψίζοντας την Ενότητα 3.4, είδαμε διάφορους αλγορίθμους δυναμικού προγραμματισμού με τους οποίους μπορούμε να αξιολογούμε, να βελτιώνουμε υπάρχουσες πολιτικές, καθώς και να προσεγγίζουμε βέλτιστες, στην περίπτωση όπου η δυναμική του περιβάλλοντος είναι γνωστή. Όλοι οι αλγόριθμοι που αναπτύχθηκαν είχαν το χαρακτη-

χρησιμοποιούσε την εξίσωση του Bellman μιας καθορισμένης προς αξιολόγηση πολιτικής, ως κανόνα ανανέωσης.

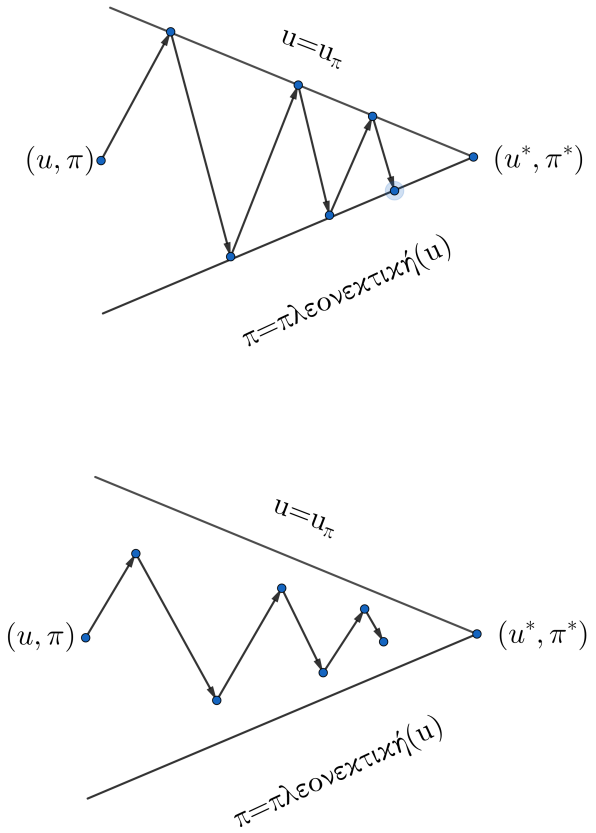
ριστικό ότι **σε κάθε σάρωση του χώρου καταστάσεων ανανεώνονταν οι αξίες όλων των καταστάσεων**. Τι γίνεται όμως στην περίπτωση όπου ο χώρος καταστάσεων είναι πολύ μεγάλος; Σίγουρα τότε η ανανέωση των αξιών για όλες τις καταστάσεις θα είναι τόσο απαιτητική υπολογιστικά που θα δημιουργήσει προβλήματα στην υλοποίηση των αλγορίθμων. Χωρίς να επεκταθούμε σε πολλές λεπτομέρειες οι οποίες θα ξέφευγαν από τα πλαίσια μιας εισαγωγής, αξίζει να αναφέρουμε πως υπάρχουν μέθοδοι προσέγγισης οι οποίες δεν απαιτούν την ανανέωση των αξιών κάθε κατάστασης σε κάθε σάρωση. Τέτοιες μέθοδοι καλούνται **Ασύγχρονες** (**Asynchronous**) και η υλοποίηση τους γίνεται με αλγόριθμους **Ασύγχρονου Δυναμικού Προγραμματισμού (Asynchronous Dynamic Programming**). Οι παραπάνω αλγόριθμοι λοιπόν, έχουν το χαρακτηριστικό, ότι σε κάθε επανάληψη η συνάρτηση αξίας ανανεώνεται για ένα υποσύνολο του χώρου καταστάσεων και σε οποιαδήποτε σειρά. Έτσι, οι αξίες κάποιων καταστάσεων μπορούν να ανανεωθούν πολλές φορές, πριν η αξία άλλων, ανανεωθεί μια φορά. Φυσικά, για να συγκλίνουν σωστά, θα πρέπει να συνεχίζουν να ανανεώνουν σε βάθος χρόνου τις αξίες όλων των καταστάσεων. Ο λόγος που τέτοιοι αλγόριθμοι είναι σημαντικοί στην Ενισχυτική Μάθηση είναι λόγω της μεγάλης ευελιξίας που έχουν στο να επιλέγουν καταστάσεις προς ανανέωση. Με αυτόν τον τρόπο, δεν χρειάζεται η εξονυχιστική ανανέωση όλου του χώρου καταστάσεων σε κάθε ανανέωση για να επιτευχθεί κάποια πρόοδος. Η προηγούμενη ιδέα είναι ιδιαιτέρως σημαντική και σε αλγορίθμους Ενισχυτικής Μάθησης όπου η δυναμική του περιβάλλοντος είναι άγνωστη.

Γενικευμένη Επανάληψη Πολιτικής

Ο Αλγόριθμος Επανάληψης Πολιτικής που συναντήσαμε στην υποενότητα 3.4.3 περιλάμβανε 2 ταυτόχρονες αλληλεπιδραστικές διαδικασίες: η μια έκανε την συνάρτηση αξίας συνεπή ως προς την τρέχουσα πολιτική (αξιολόγηση πολιτικής), και η άλλη έκανε την πολιτική, πλεονεκτική, ως προς την τρέχουσα συνάρτηση αξίας. Στην επανάληψη πολιτικής, οι δυο διαδικασίες εναλλάσσονται με τέτοιο τρόπο, ώστε η μια να ολοκληρώνεται πριν την έναρξη της άλλης. Αυτό όμως δεν είναι πάντα απαραίτητο. Για παράδειγμα, στον Αλγόριθμο Επανάληψης Αξίας, είδαμε ότι μόνο μια επανάληψη αξιολόγησης πολιτικής μεσολαβούσε για κάθε βελτίωση πολιτικής. Στην πραγματικότητα, η επανάληψη αξίας, αποτελεί απλώς μια ειδική περίπτωση μιας ευρύτερης οικογένειας αλγορίθμων που ονομάζεται **Γενικευμένη Επανάληψη Πολιτικής** (**Generalized Policy Iteration**) Οι παραπάνω αλγόριθμοι έχουν το χαρακτηριστικό, ότι επιτρέπουν στις 2 διαδικασίες να αλληλεπιδρούν σε ένα ακόμα πιο γενικό πλαίσιο. Η ευελιξία τους είναι τόσο μεγάλη που επιτρέπει τόσο τη χρήση σύγχρονων, όσο και ασύγχρονων μεθόδων. Λόγω αυτής της μεγάλης ευελιξίας, σχεδόν όλες οι μέθοδοι Ενισχυτικής Μάθησης μπορούν να περιγραφούν καλά με τη χρήση της Γενικευμένης Επανάληψης Πολιτικής.

Η βασική διαφορά της Γενικευμένης Επανάληψης Πολιτικής σε σχέση με την Επανάληψη πολιτικής είναι ότι η τελευταία δεν χρησιμοποιεί απαραίτητα ακριβείς προσεγ-

γίσεις για την συνάρτηση αξίας για να μεταβεί στο στάδιο της βελτίωσης. Έτσι κατά το στάδιο της βελτίωσης η πολιτική γίνεται πλεονεκτική ως προς μια συνάρτηση αξίας που αποτελεί προσέγγιση της πραγματικής. Με αυτόν τον τρόπο και η πολιτική του βήματος βελτίωσης αποτελεί προσέγγιση της πλεονεκτικής πολιτικής, που θα κατασκευαζόταν αν και η συνάρτηση αξίας ήταν ακριβής. Φυσικά, ο τελικός στόχος που είναι η αποτελεσματική προσέγγιση βέλτιστων πολιτικών παραμένει ίδιος. Αν πάντως οι διαδικασίες αξιολόγησης και βελτίωσης σταθεροποιηθούν αυτό σημαίνει ότι οι τρέχουσα συνάρτηση αξίας και η πολιτική πρέπει να είναι βέλτιστες. Στο Σχήμα 3.12 απεικονίζεται μια απλοποιημένη γεωμετρική αναπαράσταση για τα 2 είδη επανάληψης πολιτικής.



Σχήμα 3.12: Απλοποιημένη γεωμετρική αναπαράσταση της εναλλαγής των σταδίων αξιολόγησης και βελτίωσης πολιτικής για την προσέγγιση κάποιας βέλτιστης, κατά την Επανάληψη Πολιτικής (πάνω) και Γενικευμένη Επανάληψη Πολιτικής (κάτω).

3.5 Μέθοδοι Monte Carlo

Εκτίμηση Αξιών με Monte Carlo

Οι μέθοδοι που αναπτύχθηκαν στην προηγούμενη ενότητα είχαν το χαρακτηριστικό ότι θεωρούσαν δεδομένο ότι η δυναμική $p(s', r|s, a)$ είναι γνωστή. Προφανώς όμως, αυτό δεν είναι πάντα εφικτό σε στοχαστικά φαινόμενα. Στην πραγματικότητα, στις περισσότερες φορές το υποκείμενο δεν γνωρίζει a-priori την κατανομή πιθανότητας που του φανερώνονται τα κέρδη, και έτσι ο μοναδικός τρόπος να εκτιμήσει όλες τις ποσότητες ενδιαφέροντος είναι με βάση την **εμπειρία** του. Η εμπειρία του μπορεί να προκύψει είτε με **πραγματική**, είτε με **προσομοιωμένη** αλληλεπίδραση (όποτε είναι αυτό εφικτό) με το περιβάλλον του.

Όπως και στις μεθόδους όπου η δυναμική ήταν γνωστή, η πρώτη σημαντική ποσότητα που το υποκείμενο πρέπει να μπορεί να εκτιμά είναι η συνάρτηση αξίας μιας πολιτικής. Αυτό, όμως, τώρα πρέπει να γίνει παρά την απουσία ενός τέλειου μοντέλου. Για να δούμε πώς αυτό μπορεί να υλοποιηθεί, υπενθυμίζουμε τον ορισμό της αξίας μιας κατάστασης:

$$u_\pi(s) = \mathbb{E}_\pi \left[G_t \mid S_t = s \right]$$

Σε περίπτωση όπου η αλληλεπίδραση χαρακτηρίζεται από την ύπαρξη κάποιας τερματικής κατάστασης, (και άρα η αλληλεπίδραση είναι δυνατόν να σπάσει σε ανεξάρτητα επεισόδια) τότε το υποκείμενο αλληλεπιδρώντας με το περιβάλλον του σε ανεξάρτητα επεισόδια μπορεί να λάβει ένα τυχαίο δείγμα από την κατανομή που ακολουθεί η τυχαία μεταβλητή $Y(s) := G_t \mid S_t = s$. Τότε αν η ακολουθία των αμοιβών είναι φραγμένη, και η κατανομή των $Y_n(s)$ είναι **στάσιμη**, από τον Ισχυρό Νόμο των Μεγάλων Αριθμών θα έχουμε:

$$\overline{Y_n(s)} := \frac{Y_1(s) + Y_2(s) + \cdots Y_n(s)}{n} \overset{\sigma.\beta}{\to} u_\pi(s), \tag{3.33}$$

όπου $\overset{\sigma.\beta}{\to}$, η γνωστή, σχεδόν βέβαιη σύγκλιση. Έτσι λοιπόν για έναν αρκετά μεγάλο αριθμό επεισοδίων, n, αναμένουμε ότι η δειγματική αξία της κατάστασης s, $\overline{Y_n(s)}$, θα είναι αρκετά κοντά στην πραγματική, $u_\pi(s)$. Εκτιμώντας την αξία **κάθε** κατάστασης $s \in S$ με αυτόν τον τρόπο, θα έχουμε μια εκτίμηση της συνάρτησης αξίας u_π. Επειδή η εκτίμηση της u_π, έγινε αποκλειστικά παίρνοντας μέσους όρους τυχαίων δειγμάτων ανήκει στην ευρεία οικογένεια των **Μεθόδων Monte Carlo.**[6]

Φυσικά, η επίσκεψη σε μια κατάσταση $s \in S$, μπορεί να συμβαίνει περισσότερο από 1 φορές στο ίδιο επεισόδιο. Η **Μέθοδος Monte Carlo πρώτης επίσκεψης** εκτιμά

[6]Οι Μέθοδοι Monte Carlo αποτελούν μια ευρεία κλάση υπολογιστικών αλγορίθμων που βασίζονται στην τυχαία δειγματοληψία για την εύρεση αριθμητικών αποτελεσμάτων. Ο όρος Monte Carlo χρησιμοποιείται συχνά για οποιαδήποτε μέθοδο εκτίμησης που η λειτουργία της περιλαμβάνει μια σημαντική τυχαία συνιστώσα.

την $u_\pi(s)$ προσμετρώντας την αξία που θα λάβει το υποκείμενο αποκλειστικά από την πρώτη που φορά που διέρχεται από την s, σε αντίθεση με την **Μέθοδο Monte Carlo κάθε επίσκεψης** η οποία συνυπολογίζει τις αξίες που προκύπτουν από κάθε επίσκεψη στην s. Οι δυο μέθοδοι είναι αρκετά όμοιες αλλά παρουσιάζουν κάποιες ελαφρώς διαφορετικές θεωρητικές ιδιότητες. Σε αυτή την ενότητα, ωστόσο, θα εστιάσουμε στην δεύτερη. Στον Πίνακα 3.4 παρουσιάζεται η Μέθοδος Monte Carlo Κάθε Επίσκεψης για την εκτίμηση της συνάρτηση αξίας μιας πολιτικής π (αξιολόγηση πολιτικής), σε μορφή ψευδοκώδικα. Αξίζει να σημειώσουμε πως για την ανανέωση των μέσων όρων έχουμε χρησιμοποιήσει τον κανόνα:

$$\text{Νέα Εκτίμηση} \leftarrow \text{Παλαιά Εκτίμηση} + \text{Μέγεθος Βήματος}\left[\text{Στόχος} - \text{Παλαιά Εκτίμηση}\right],$$

που συναντήσαμε στο Κεφάλαιο των Multi-Armed-Bandits, καθώς όπως είδαμε, είναι υπολογιστικά ευνοϊκότερος.

Πίνακας 3.4 Ο Αλγόριθμος Monte Carlo Κάθε Επίσκεψης για την Αξιολόγηση Πολιτικής

Είσοδος: Μια πολιτική π, προς αξιολόγηση, ο αριθμός των επιθυμητών επεισοδίων, n, και ο συντελεστής αποπληθωρισμού, γ.

1. Αρχικοποίησε $V(s) \leftarrow 0$ για κάθε $s \in S$, και $N(s) \leftarrow 0$ για κάθε $s \in S$, όπου $N(s)$, το πλήθος των φορών που το υποκείμενο έχει περάσει από την κατάσταση s.
2. Επανάλαβε n φορές τα ακόλουθα:
i. Δημιούργησε ένα επεισόδιο που ακολουθεί την π: $S_0, A_0, R_1.S_1, A_1, R_2, \ldots, R_{T-1}, S_{T-1}, A_{T-1}, R_T$
ii. Θέσε $G_T \leftarrow 0$
iii. Για κάθε βήμα $t = T-1, T-2, \ldots, 0$ επανάλαβε:
$G_t \leftarrow R_{t+1} + \gamma G_{t+1}$
$N(S_t) \leftarrow N(S_t) + 1$
$V(S_t) \leftarrow V(S_t) + \frac{1}{N(S_t)}(G_t - V(S_t))$

Έξοδος: Μια συνάρτηση αξίας $V \approx u_\pi$.

Όπως είδαμε και στις προσεγγιστικές μεθόδους Δυναμικού Προγραμματισμού που αναπτύξαμε, είναι ιδιαιτέρως σημαντικό, πέρα από την συνάρτηση αξίας $u_\pi(s)$, να μπορούμε και να εκτιμάμε τη συνάρτηση αξίας για τα ζεύγη κατάστασης-απόφασης $Q_\pi(s,a)$. Η τελευταία, είναι ιδιαιτέρως χρήσιμη για την διαδικασία της βελτίωσης μιας υπάρχουσας πολιτικής, και κατ΄ επέκταση στην προσέγγιση κάποιας βέλτιστης. Η Μέθοδος Monte Carlo για την εκτίμηση της Q_π είναι ουσιαστικά ίδια με αυτή που διατυπώθηκε και για

την u_π. Η διαφορά τώρα είναι όταν χρησιμοποιούμε τον όρο επίσκεψη, αυτός θα αφορά ένα ζεύγος κατάστασης-απόφασης (s, a). Παίρνοντας πάντως ένα αρκετά μεγάλο δείγμα από τις τυχαίες μεταβλητές $Y(s, a) := G_t \mid S_t = s, A_t = a$ και κατασκευάζοντας τους αντίστοιχους μέσους όρους μπορούμε πάλι να προσεγγίσουμε την Q_π, υπό την εγγύηση του I.N.M.A.

Η μόνη περιπλοκή που ενδεχομένως συναντήσουμε τώρα, είναι ότι κάποια ζεύγη κατάστασης-απόφασης μπορεί να μην τα επισκεφθεί ποτέ το υποκείμενο, κάτω από συγκεκριμένες πολιτικές. Αν για παράδειγμα η π, είναι ντετερμινιστική τότε οι αξίες $Q_\pi(s, a)$ θα ανανεώνονται για μοναδικά $a \in A(s)$. Αυτό αποτελεί σημαντικό πρόβλημα καθώς θα υπάρχουν αξίες ζευγών κατάστασης και απόφασης για τα οποία δεν θα υπάρχει προσέγγιση. Μια λύση στο παραπάνω πρόβλημα είναι να απαιτήσουμε κάθε επεισόδιο να ξεκινά από οποιοδήποτε ζεύγος $(s, a) \in (S, A(S))$ με θετική πιθανότητα. Η παραπάνω απαίτηση θα καλείται υπόθεση της **Τυχαιοποιημένης Αρχικοποίησης** (**Exploring Starts Hypothesis**) του πρώτου ζεύγους κατάστασης-απόφασης κάθε επεισοδίου. Η παραπάνω υπόθεση εξασφαλίζει ότι οριακά όλα τα ζεύγη κατάστασης απόφασης, θα επισκέπτονται άπειρες φορές, με συνέπεια να οδηγούμαστε σε αποτελεσματική προσέγγιση της Q_π.

Πίνακας 3.5 Ο Αλγόριθμος Monte Carlo Κάθε Επίσκεψης για την Προσέγγιση της Q_π (με Τυχαιοποιημένη Αρχικοποίηση)

Είσοδος: Μια πολιτική π, ο αριθμός των επιθυμητών επεισοδίων, n, και ο συντελεστής αποπληθωρισμού, γ.

1. Αρχικοποίησε $Q(s, a) \leftarrow 0$ για κάθε $s \in S$, $a \in A(s)$ και $N(s, a) \leftarrow 0$ για κάθε $s \in S$, όπου $N(s, a)$, το πλήθος των φορών που το υποκείμενο έχει επισκεφθεί το ζεύγος (s, a).
2. Επανέλαβε n φορές τα ακόλουθα:
i. Επίλεξε $(S_0, A_0) \in (S, A(S_0))$ έτσι ώστε η πιθανότητα επιλογής $P(S = S_0,\ A = A(S_0)) > 0$ για κάθε $(S, A) \in (S, A(S))$
ii. Δημιούργησε ένα επεισόδιο που ακολουθεί την π: $S_0, A_0, R_1.S_1, A_1, R_2, \ldots, R_{T-1}, S_{T-1}, A_{T-1}, R_T$
iii. Θέσε $G_T \leftarrow 0$
iv. Για κάθε βήμα $t = T - 1, T - 2, \ldots, 0$ επανέλαβε:
$G_t \leftarrow R_{t+1} + \gamma G_{t+1}$
$N(S_t, A_t) \leftarrow N(S_t, A_t) + 1$
$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}(G_t - Q(S_t, A_t))$

Έξοδος: Μια συνάρτηση αξίας κατάστασης απόφασης $Q \approx Q_\pi$.

Προσέγγιση Βέλτιστων Πολιτικών με Monte Carlo

Έχοντας δει πώς μπορούμε να εκτιμούμε αξίες για τα ζεύγη κατάστασης-απόφασης με τη Μέθοδο Monte Carlo μπορούμε να προχωρήσουμε στη διατύπωση ενός Αλγορίθμου Monte Carlo για την προσέγγιση κάποιας βέλτιστης πολιτικής. Ο Αλγόριθμος του Πίνακα 3.6 ουσιαστικά αποτελεί παράδειγμα Αλγορίθμου Γενικευμένης Επανάληψης Πολιτικής: Η συνάρτηση $Q_\pi(s,a)$ ανανεώνεται τώρα για κάθε επεισόδιο με την Μέθοδο Monte Carlo, και το βήμα βελτίωσης γίνεται ως προς την πλεονεκτική πολιτική, αυτή που δηλαδή σε κάθε κατάσταση επιλέγει την δράση με την μεγαλύτερη αξία. Λόγω της Τυχαιοποιημένης Αρχικοποίησης οι προσεγγίσεις της $Q_\pi(s,a)$, γίνονται όλο και καλύτερες σε βάθος χρόνου με αποτέλεσμα η ακολουθία των πολιτικών που παράγει ο αλγόριθμος να συγκλίνει σε κάποια π^*.

Πίνακας 3.6 Ο Αλγόριθμος Monte Carlo για προσέγγιση κάποιας βέλτιστης πολιτικής π^* (με Τυχαιοποιημένη Αρχικοποίηση)

Είσοδος: Ο αριθμός των επιθυμητών επεισοδίων, n, και ο συντελεστής αποπληθωρισμού γ.

1. Αρχικοποίησε:
i. $\pi(s) \in A(S)$ για κάθε $a \in A(s)$
ii. $Q(s,a) \leftarrow 0$ για κάθε $s \in S$, $a \in A(s)$
iii. $N(s,a) \leftarrow 0$ για κάθε $s \in S$
2. Επανάλαβε n φορές τα ακόλουθα:
i. Επίλεξε $(S_0, A_0) \in (S, A(S_0))$ έτσι ώστε η πιθανότητα επιλογής $P(S = S_0,\ A = A(S_0)) > 0$ για κάθε $(S,A) \in (S, A(S))$
ii. Δημιούργησε ένα επεισόδιο που ακολουθεί την π: $S_0, A_0, R_1.S_1, A_1, R_2, \ldots, R_{T-1}, S_{T-1}, A_{T-1}, R_T$
iii. Θέσε $G_T \leftarrow 0$
iv. Για κάθε βήμα $t = T-1, T-2, \ldots, 0$ επανάλαβε:
$G_t \leftarrow R_{t+1} + \gamma G_{t+1}$
$N(S_t, A_t) \leftarrow N(S_t, A_t) + 1$
$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}(G_t - Q(S_t, A_t))$
$\pi(S_t) \leftarrow \underset{a \in A(S_t)}{\text{argmax}}\ Q(S_t, a)$

Έξοδος: Μια πολιτική $\pi \approx \pi^*$.

3.6 Μέθοδος Προσωρινών Διαφορών

Ένα βασικό χαρακτηριστικό το οποίο διέθετε μέθοδος Monte Carlo για την εκτίμηση της συνάρτησης αξίας u_π, ήταν ότι η ανανέωση τον καταστάσεων που επισκέφθηκε το υποκείμενο κατά τη διάρκεια ενός επεισοδίου γινόταν μετά τη λήξη του επεισοδίου, ώστε να μπορούν να είναι γνωστά όλα τα συνολικά κέρδη G_t. Αυτό όμως δεν είναι αναγκαστικά απαραίτητο. Ας θυμηθούμε την αναδρομική έκφραση του G_t, ως προς το συνολικό κέρδος της επόμενης κατάστασης:

$$G_t = R_{t+1} + \gamma G_{t+1}$$

Συνεπώς,

$$\begin{aligned} u_\pi(s) &= \mathbb{E}_\pi \left[G_t \mid S_t = s \right] \\ &= \mathbb{E}_\pi \left[R_{t+1} + \gamma G_{t+1} \mid S_t = s \right] \\ &= \mathbb{E}_\pi \left[R_{t+1} + \gamma u_\pi(S_{t+1}) \mid S_t = s \right] \end{aligned}$$

Αν λοιπόν έχουμε στη διάθεση μας μια προσέγγιση $V_\pi(S_{t+1})$, της $u_\pi(S_{t+1})$, θα αναμένουμε $G_t \approx R_{t+1} + V_\pi(S_{t+1})$. Ο κανόνας ανανέωσης των αξιών με τη Μέθοδο Monte Carlo για στάσιμα προβλήματα[7] ήταν:

$$V(S_t) \leftarrow V(S_t) + \frac{1}{N(S_t)}(G_t - V(S_t)) \tag{3.34}$$

Αν αντικαταστήσουμε στην (3.34) την G_t, με την προσέγγιση της, $R_{t+1} + V_\pi(S_{t+1})$, θα προκύψει ο κανόνας:

$$V(S_t) \leftarrow V(S_t) + \frac{1}{N(S_t)}(R_{t+1} + V(S_{t+1}) - V(S_t)) \tag{3.35}$$

Ο κανόνας ανανέωσης των αξιών με βάση την (3.35), ονομάζεται **Μέθοδος Προσωρινών Διαφορών** ή **Εκμάθηση Προσωρινών Διαφορών** (**Temporal Difference Learning**). Το βασικό της χαρακτηριστικό είναι ότι η ανανέωση της κάθε κατάστασης γίνεται με βάση την αμοιβή του επόμενου βήματος και την εκτίμηση της αξίας της επόμενης κατάστασης. Με αυτόν τον τρόπο δεν χρειάζεται να αναμένουμε την λήξη του επεισοδίου για να ανανεωθούν οι εκτιμήσεις των αξιών όπως γινόταν με την Αξιολόγηση Monte Carlo, και αυτό το στοιχείο την καθιστά αμεσότερη. Όσον αφορά την σύγκλιση της $V(S_t)$ στην u_π με τον κανόνα (3.35) αποδεικνύεται ότι και αυτή συμβαίνει με πιθανότητα 1 όπως ακριβώς συνέβαινε και με τον (3.34). Παρά το γεγονός ότι μέχρι

[7] Ο κανόνας ανανέωσης μπορεί να επεκταθεί και για μη στάσιμα προβλήματα επιλέγοντας σταθερό μέγεθος βήματος $a \in [0, 1)$, αλλά για λόγους απλότητας θα παραμείνουμε στην στάσιμη περίπτωση.

και σήμερα δεν έχει αποδειχτεί με ποια από τις 2 μεθόδους επιτυγχάνεται γρηγορότερη σύγκλιση, στην πράξη έχει βρεθεί ότι η αμεσότητα που έχει η Ε.Π.Δ στην ανανέωση των αξιών, οδηγεί συνήθως σε γρηγορότερη σύγκλιση από την κλασσική Μ.C.

Στον Πίνανα (3.7) παρουσιάζεται η Μέθοδος Προσωρινών Διαφορών σε μορφή για την εκτίμηση της u_π σε μορφή ψευδοκώδικα (για στάσιμα προβλήματα). Αυτός ο Αλγόριθμος καλείται σύμφωνα με την διεθνή βιβλιογραφία TD(0), καθώς αποτελεί ειδική περίπτωση ενός γενικότερου Αλγορίθμου Ενισχυτικής Μάθησης που ονομάζεται TD(λ) ο οποίος δημιουργήθηκε από τον Richard Sutton[8].

Πίνακας 3.7 Ο Αλγόριθμος TD(0) για την Εκτίμηση της u_π (για στάσιμα προβλήματα)

Είσοδος: Μια πολιτική π, προς αξιολόγηση, ο αριθμός των επιθυμητών επεισοδίων, n, και ο συντελεστής αποπληθωρισμού, γ.

1. Αρχικοποίησε $V(s) \leftarrow 0$ για κάθε $s \in S$, και $N(s) \leftarrow 0$ για κάθε $s \in S$, όπου $N(s)$, το πλήθος των φορών που το υποκείμενο έχει περάσει από την κατάσταση s.
2. Επανάλαβε n φορές τα ακόλουθα:
i. Παρατήρησε μια αρχική κατάσταση $State$ που προκύπτει από την πολιτική π
ii Όσο η κατάσταση $State$ δεν είναι τερματική επανάλαβε:
Παρατήρησε μια κατάσταση A, που προκύπτει από εφαρμογή της πολιτικής π, στην κατάσταση $State$
Παρατήρησε αμοιβή R, και νέα κατάσταση $State'$, οι οποίες προκύπτουν από την λήψη της A
$N(State) \leftarrow N(State) + 1$
$V(State) \leftarrow V(State) + \frac{1}{N(State)}(R + V(State') - V(State))$
$State \leftarrow State'$

Έξοδος: Μια συνάρτηση αξίας $V \approx u_\pi$.

3.7 Ο Αλγόριθμος Sarsa

Ας δούμε τώρα πως μπορούμε να χρησιμοποιήσουμε την Μέθοδο Προσωρινών Διαφορών, τώρα όμως για το πρόβλημα της προσέγγισης κάποιας βέλτιστης πολιτικής. Όπως και στον Αλγόριθμο του Πίνακα 3.6 τώρα θα χρειαστούμε αρχικά έναν κανόνα ανανέωσης για τις αξίες των ζευγών κατάστασης-απόφασης ο οποίος είναι απαραίτητος για το

[8]Τόσο ο Richard Sutton, όσο και ο Andrew Barto θεωρούνται 2, εκ των πατέρων την μοντέρνας Ενισχυτικής Μάθησης με πολύ σημαντική συνεισφορά στη περιοχή.

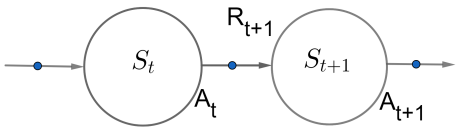
στάδιο της αξιολόγησης πολιτικής. Εφαρμόζοντας ακριβώς τον κανόνα ανανέωσης της (3.35), τώρα όμως για τις $Q(S_t, A_t)$, θα προκύψει ο κανόνας ανανέωσης:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}\Big[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\Big] \qquad (3.36)$$

Όπως και ο Αλγόριθμος TD(0), μπορεί να αποδειχθεί ότι όσο εξασφαλίζεται ότι η ανανέωση για όλα τα ζεύγη δεν θα σταματά σε βάθος χρόνου, όλες οι ποσότητες Q θα συγκλίνουν στις Q^*.

Για να προκύψει ένας Αλγόριθμος Γενικευμένης Επανάληψης Πολιτικής χρειαζόμαστε επιπλέον να καθορίσουμε πώς θα πραγματοποιείται και το στάδιο Βελτίωσης Πολιτικής. Αυτό μπορεί να γίνει παίρνοντας πάντα την πλεονεκτική δράση (όπως είδαμε και σε προηγούμενους Αλγορίθμους) ή εναλλακτικά αποφασίζοντας με βάση την ε-πλεονεκτική δράση όπως είδαμε και στο κεφάλαιο των Multi-Armed-Bandits. Το πλεονέκτημα της τελευταίας είναι, ότι ανανεώνοντας την πολιτική με βάση την ε-πλεονεκτική δράση μπορούμε να πετύχουμε η τυχαιοποιημένη αρχικοποίηση να γίνεται μόνο για την αρχική κατάσταση και όχι για το αρχικό ζεύγος-κατάστασης απόφασης. Για σταθερή βέβαια τιμή του ε, όπως είδαμε και στο Κεφάλαιο 2, δεν θα επιτυγχάνεται ακριβής σύγκλιση στο Q^*, στο όριο, και συνεπώς, αν επιθυμούμε την ασυμπτωτική σύγκλιση θα πρέπει να επιλέξουμε κάποια άλλη τιμή όπως για παράδειγμα $\varepsilon = \frac{1}{n}$, όπου n ο αριθμός των επεισοδίων.

Συνοψίζοντας όλες τις παραπάνω ιδέες μπορεί να προκύψει ο Αλγόριθμος Γενικευμένης Επανάληψης Πολιτικής **Sarsa** που συνοψίζεται στον ψευδοκώδικα του Πίνακα 3.8. Η ονομασία του τελευταίου, έχει προκύψει αφού σε κάθε ανανέωση είναι απαραίτητη η γνώση της πεντάδας $(S_t, A_t, R_t, S_{t+1}, A_{t+1})$.



Σχήμα 3.13: Η πεντάδα των στοιχειών που χρειάζεται ο Αλγόριθμος Sarsa για την ανανέωση των αξιών των ζευγών κατάστασης-απόφασης.

Πίνακας 3.8 Ο Αλγόριθμος Sarsa για προσέγγιση κάποιας βέλτιστης πολιτικής π^* (με Τυχαιοποιημένη Αρχικοποίηση της αρχικής κατάστασης και πολιτική ε-greedy για την επιλογή αποφασεων)

Είσοδος: Ο αριθμός των επιθυμητών επεισοδίων, n, ο συντελεστής αποπληθωρισμού γ, και ο συντελεστής ε.

1. Αρχικοποίησε:
i. $Q(s,a) \leftarrow 0$ για κάθε $s \in S$
ii. $N(s,\alpha) \leftarrow 0$ για κάθε $s \in S$, $a \in A(s)$
2. Επανάλαβε n φορές τα ακόλουθα:
i. Αρχικοποίησε την πρώτη κατάσταση την αλληλεπίδρασης $State$ (ώστε να κάθε κατάσταση να επιλέγεται με θετική πιθανότητα)
ii Όσο η κατάσταση $State$ δεν είναι τερματική επανάλαβε:

$$A \leftarrow \begin{cases} \underset{a \in A(State)}{\text{argmax}}\ Q(State,a) & \text{, με πιθανότητα 1-ε} \\ DUnif(a_1, a_2, ..., a_{|A(State)|}) & \text{, με πιθανότητα ε} \end{cases}$$

Παρατήρησε αμοιβή R και νέα κατάσταση $State'$

$$A' \leftarrow \begin{cases} \underset{a \in A(State')}{\text{argmax}}\ Q(State',a) & \text{, με πιθανότητα 1-ε} \\ DUnif(a_1, a_2, ..., a_{|A(State')|}) & \text{, με πιθανότητα ε} \end{cases}$$

$N(State,A) \leftarrow N(State,A) + 1$
$Q(State,A) \leftarrow Q(State,A) + \frac{1}{N(State,A)}\Big[R + \gamma Q(State',A') - Q(State,A)\Big]$
$State \leftarrow State'$
$A \leftarrow A'$
3. Για κάθε $s \in S$ θέσε:
$\pi(s) \leftarrow \underset{a \in A(s)}{\text{argmax}}\ Q(s,a)$

Έξοδος: Μια πολιτική $\pi \approx \pi^*$.

3.8 Q-Μάθηση

Μέχρι τώρα έχουμε διατυπώσει 2 Αλγορίθμους για τη εύρεση μιας προσεγγιστικά βέλτιστης πολιτικής, χωρίς γνώση της δυναμικής του Περιβάλλοντος (υποθέτοντας βέβαια

ότι αυτή δεν αλλάζει με την πάροδο του χρόνου): τον κλασσικό Αλγόριθμο Monte Carlo, και τον Αλγόριθμο Sarsa. Ένας τρίτος Αλγόριθμος ο οποίος αποτελεί έναν εκ των σημαντικότερων και των πιο συχνά χρησιμοποιούμενων είναι αυτός της **Q-Μάθησης**, ο οποίος ανακαλύφθηκε από τον Watkins το 1989. Ο τελευταίος είναι πολύ κοντά στην λογική του Αλγόριθμου Sarsa με τη διαφορά ότι η ανανέωση των ζευγών κατάστασης απόφασης γίνεται με τον κανόνα:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}\Big[R_{t+1} + \gamma \max_{a \in A(S_{t+1})} Q(S_{t+1}, a) - Q(S_t, A_t)\Big] \quad (3.37)$$

Στον Πίνακα 3.9 παρουσιάζεται ο Αλγόριθμος **Q-Μάθησης** για την προσέγγιση κάποιας π^* με μέγεθος βήματος $\frac{1}{N(S_t,A_t)}$. Η σύγκλιση του (3.37) στο Q^* αποδείχθηκε από τους Watkins και Dayan, το 1992, για ένα ακόμα γενικότερο μέγεθος βήματος από αυτό που παρατηρούμε στον κανόνα (3.37), στο οποίο όμως δεν θα εμβαθύνουμε περαιτέρω.

Πίνακας 3.9 Ο Αλγόριθμος Q-Μάθησης για την εκτίμηση του Q^* (με Τυχαιοποιημένη Αρχικοποίηση της αρχικής κατάστασης και πολιτική ε-greedy για την επιλογή αποφασεων)

Είσοδος: Ο αριθμός των επιθυμητών επεισοδίων, n, ο συντελεστής αποπληθωρισμού γ, και ο συντελεστής ε.

1. Αρχικοποίησε:
i. $Q(s, a) \leftarrow 0$ για κάθε $s \in S$
ii. $N(s, \alpha) \leftarrow 0$ για κάθε $s \in S$, $a \in A(s)$
2. Επανάλαβε n φορές τα ακόλουθα:
i. Αρχικοποίησε την πρώτη κατάσταση την αλληλεπίδρασης $State$ (ώστε να κάθε κατάσταση να επιλέγεται με θετική πιθανότητα)
ii Όσο η κατάσταση $State$ δεν είναι τερματική επανάλαβε:

$$A \leftarrow \begin{cases} \underset{a \in A(State)}{\text{argmax}}\ Q(State, a) & \text{, με πιθανότητα 1-ε} \\ DUnif(a_1, a_2, ..., a_{|A(State)|}) & \text{, με πιθανότητα ε} \end{cases}$$

Παρατήρησε αμοιβή R και νέα κατάσταση $State'$
$N(State, A) \leftarrow N(State, A) + 1$
$Q(State, A) \leftarrow Q(State, A) + \frac{1}{N(State,A)}\Big[R + \gamma \max_{a \in A(S_{t+1})} Q(State', a) - Q(State, A)\Big]$
$State \leftarrow State'$

Έξοδος: $Q \approx Q^*$.

Παράρτημα

Κώδικες στο Λογισμικό R

Σε αυτό το παράρτημα περιλαμβάνονται οι κώδικες που χρησιμοποιήθηκαν στο Κεφάλαιο 2.7 των Multi-Armed-Bandits καθώς και σε παραδείγματα στο Κεφάλαιο 3.

1 greedy_beta

```
# greedy_beta

#Bandit problem with k=5 arms with

#arm 1 ~ Beta (2,3)
#arm 2 ~ Beta (9,11)
#arm 3 ~ Beta (5,5)
#arm 4 ~ Beta (11,9)
#arm 5 ~ Beta (3,2)

#package for discrete uniform function rdunif "purrr"
#package for argmax "ramify"
#max.col(a) returns the position if the max value of matrix
#(1 x k) a breaking ties randomly by default
#ggplot2 package for nice visualization

library("purrr")
library("ramify")
library("ggplot2")
```

```
# beta function

BETA<- function(i)
{if (i==1)
{BETA=rbeta(1,shape1 = 2, shape2 = 3)}
  else if (i==2)
  {BETA=rbeta(1,shape1 = 9, shape2 = 11)}
  else if (i==3)
  {BETA=rbeta(1,shape1 = 5, shape2 = 5)}
  else if (i==4)
  {BETA=rbeta(1,shape1 = 11, shape2 = 9)}
  else if (i==5)
  {BETA=rbeta(1,shape1 = 3, shape2 = 2)}
  else {BETA=0}
}

# beta parameters
SHAPE_1=mat("2,9,5,11,3")
SHAPE_2=mat("3,11,5,9,2")
arms_means=SHAPE_1/(SHAPE_1+SHAPE_2)
m_star=max(arms_means)
a_star=argmax(arms_means)


M=10000
k=length(SHAPE_1)
M_exploration=2
q=matrix(1:k,nrow=1,ncol=k)*0
N=matrix(1:k,nrow=1,ncol=k)*0
a=matrix(1:M,nrow=1,ncol=M)*0
p=matrix(1:M,nrow=1,ncol=M)*0
x_mean=matrix(1:M,nrow=1,ncol=M)*0
Total_regret=matrix(1:M,nrow=1,ncol=M)*0
for (i in 1:k) {
  for (j in ((i-1)*M_exploration+1):(i*M_exploration)) {
    {a[j]=i
    N[i]=N[i]+1
    q[i]=q[i]+(1/N[i])*(BETA(i)-q[i])
    x_mean[j]=sum(N*q)/j
    if (j==1)
    {Total_regret[j]=m_star-arms_means[j]}
    else {Total_regret[j]=Total_regret[j-1]
```

```
    +m_star-arms_means[i]}
    }
    if (i==a_star)
    {p[i]=1/j}
    else {p[i]=0}
  }
}
#print estimations

print(paste0("The estimation of mean_1 is : ", q[1]))
print(paste0("The estimation of mean_2 is : ", q[2]))
print(paste0("The estimation of mean_3 is : ", q[3]))
print(paste0("The estimation of mean_4 is : ", q[4]))
print(paste0("The estimation of mean_5 is : ", q[5]))

I=max.col(q)
for (j in ((k*M_exploration)+1):M) {
  a[j]=I
  N[I]=N[I]+1
  q[I]=q[I]+(1/N[I])*(BETA(I)-q[I])
  #percenage of the optimal action chosen
  p[j]=N[a_star]/j
  #sample mean
  x_mean[j]=(sum(N*q)/j)
  #total regret
  Total_regret[j]=Total_regret[j-1]+m_star-arms_means[I]
}


number_of_rounds=1:M
p_dataframe=data.frame(p,number_of_rounds)
x_mean_dataframe=data.frame(x_mean,number_of_rounds)
Total_regret_dataframe=data.frame(Total_regret,number_of_rounds
   )

#plots of perfomance measures
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=p))
+geom_line()
ggplot(data=x_mean_dataframe,aes(x=number_of_rounds,y=x_mean))
+geom_line()
ggplot(data=Total_regret_dataframe,aes(x=number_of_rounds,
y=Total_regret))+geom_line()
```

```
#outputs
print(paste0("The percentage of optimal action chosen is :
", p[M]))
print(paste0("The Total Regret is : ", Total_regret[M]))
print(paste0("The Sample Mean is : ", x_mean[M]))
```

2 epsilon_greedy_beta

```
#epsilon_greedy_beta

#Bandit problem with k=5 arms with

#arm 1 ~ Beta (2,3)
#arm 2 ~ Beta (9,11)
#arm 3 ~ Beta (5,5)
#arm 4 ~ Beta (11,9)
#arm 5 ~ Beta (3,2)

#package for discrete uniform function rdunif "purrr"

#package for argmax "ramify"

#max.col(a) returns the position if the max value of matrix
#(1 x k) a breaking ties randomly by default
#ggplot2 package for nice visualization

library("purrr")
library("ramify")
library("ggplot2")

# beta function

BETA<- function(i)
{if (i==1)
{BETA=rbeta(1,shape1 = 2, shape2 = 3)}
  else if (i==2)
  {BETA=rbeta(1,shape1 = 9, shape2 = 11)}
  else if (i==3)
  {BETA=rbeta(1,shape1 = 5, shape2 = 5)}
```

```
  else if (i==4)
  {BETA=rbeta(1,shape1 = 11, shape2 = 9)}
  else if (i==5)
  {BETA=rbeta(1,shape1 = 3, shape2 = 2)}
  else {BETA=0}
}

# beta parameters
SHAPE_1=mat("2,9,5,11,3")
SHAPE_2=mat("3,11,5,9,2")
arms_means=SHAPE_1/(SHAPE_1+SHAPE_2)
m_star=max(arms_means)
a_star=argmax(arms_means)

M_sim=1000
M=2000
k=length(SHAPE_1)
q=matrix(1:k,nrow=M_sim,ncol=k)*0
N=matrix(1:k,nrow=M_sim,ncol=k)*0
epsilon=0.1
a=matrix(1:M,nrow=M_sim,ncol=M)*0
p=matrix(1:M,nrow=M_sim,ncol=M)*0
x_mean=matrix(1:M,nrow=M_sim,ncol=M)*0
Total_regret=matrix(1:M,nrow=M_sim,ncol=M)*0
p_average=matrix(1:M,nrow=1,ncol=M)*0
Total_regret_average=matrix(1:M,nrow=1,ncol=M)*0
x_mean_average=matrix(1:M,nrow=1,ncol=M)*0

for (z in 1:M_sim)
{
y=rdunif(1,k,1)
for (i in 1:k) {
  if (y==i)
  {q[z,i]=q[z,i]+BETA(i)
  x_mean[z,1]=q[z,i]
  Total_regret[z,1]=m_star-arms_means[i]
  N[z,i]=N[z,i]+1
  a[z,1]=y}
}

if (y==a_star) {p[z,1]=1
}
```

```
for (j in 1:(M-1))
{
  u=runif(1)
  if (u <= 1-epsilon)
  {I=max.col(t(q[z,]))
  a[z,j+1]=I}
  else
  { I=rdunif(1,k,1)
  a[z,j+1]=I
  }
  N[z,I]=N[z,I]+1
  q[z,I]=q[z,I]+(1/N[z,I])*((BETA(I)-q[z,I]))
  #percenage of the optimal action chosen
  p[z,j+1]=N[z,a_star]/(j+1)
  #sample mean
  x_mean[z,j+1]=(sum(N[z,]*q[z,])/(j+1))
  #total regret
  Total_regret[z,j+1]=Total_regret[z,j]+m_star-arms_means[I]
}
}

#average perfomance of measures in round i, i=1,...M_sim
for (i in 1:M)
 { p_average[i]=sum(p[,i])/M_sim
  x_mean_average[i]=sum(x_mean[,i])/M_sim
  Total_regret_average[i]=sum(Total_regret[,i])/M_sim
}

number_of_rounds=1:M
p_dataframe=data.frame(p_average,number_of_rounds)
x_mean_dataframe=data.frame(x_mean_average,number_of_rounds)
Total_regret_dataframe=data.frame(Total_regret_average,
number_of_rounds)

#plots of perfomance measures
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=p_average))
+geom_line()
ggplot(data=x_mean_dataframe,aes(x=number_of_rounds,
y=x_mean_average))+geom_line()
ggplot(data=Total_regret_dataframe,aes(x=number_of_rounds,
y=Total_regret_average))+geom_line()
```

```
# M rounds
print(paste0("The percentage of optimal action chosen is
: ", p_average[M]))
print(paste0("The Total Regret is : ",
Total_regret_average[M]))
print(paste0("The Sample Mean is : ", x_mean_average[M]))
```

3 ucb1_beta

```
#ucb_1_beta

#Bandit problem with k=5 arms with

#arm 1 ~ Beta (2,3)
#arm 2 ~ Beta (9,11)
#arm 3 ~ Beta (5,5)
#arm 4 ~ Beta (11,9)
#arm 5 ~ Beta (3,2)

#package for discrete uniform function rdunif "purrr"
#package for argmax "ramify"
#max.col(a) returns the position if the max value of matrix
#(1 x k) a breaking ties randomly by default
#ggplot2 package for nice visualization

library("purrr")
library("ramify")
library("ggplot2")

#beta function

BETA<- function(i)
{if (i==1)
{BETA=rbeta(1,shape1 = 2, shape2 = 3)}
  else if (i==2)
  {BETA=rbeta(1,shape1 = 9, shape2 = 11)}
  else if (i==3)
  {BETA=rbeta(1,shape1 = 5, shape2 = 5)}
  else if (i==4)
  {BETA=rbeta(1,shape1 = 11, shape2 = 9)}
```

```
  else if (i==5)
  {BETA=rbeta(1,shape1 = 3, shape2 = 2)}
  else {BETA=0}
}

#beta parameters
SHAPE_1=mat("2,9,5,11,3")
SHAPE_2=mat("3,11,5,9,2")
arms_means=SHAPE_1/(SHAPE_1+SHAPE_2)
m_star=max(arms_means)
a_star=argmax(arms_means)


M=10000
k=length(SHAPE_1)
q=matrix(1:k,nrow=1,ncol=k)*0
N=matrix(1:k,nrow=1,ncol=k)*0  every arm
U=matrix(1:k,nrow=1,ncol=k)*0
Upper_Bound=q+U
a=matrix(1:M,nrow=1,ncol=M)*0
p=matrix(1:M,nrow=1,ncol=M)*0
x_mean=matrix(1:M,nrow=1,ncol=M)*0
Total_regret=matrix(1:M,nrow=1,ncol=M)*0
for (i in 1:k) {
  N[i]=N[i]+1
  q[i]=BETA(i)
  U[i]=sqrt(2*log(i)/N[i])
  Upper_Bound[i]=q[i]+U[i]
  x_mean[i]=sum(N*q)/i
  a[i]=i
  if (i==1)
  {Total_regret[i]=m_star-arms_means[i]}
  else {Total_regret[i]=Total_regret[i-1]
  +m_star-arms_means[i]}
  if (i==a_star)
  {p[i]=1/a_star}
  else {p[i]=0}
}

for (j in (k+1):M)
{
  I=max.col(Upper_Bound)
```

```
  a[j]=I
  N[I]=N[I]+1
  q[I]=q[I]+(1/N[I])*(BETA(I)-q[I])
  U[I]=sqrt(2*log(i)/N[I])
  Upper_Bound=q+U
  #percentage of the optimal action chosen
  p[j]=N[a_star]/j
  #sample mean
  x_mean[j]=(sum(N*q)/(j))
  #total regret
  Total_regret[j]=Total_regret[j-1]+m_star-arms_means[I]
}
number_of_rounds=1:M
p_dataframe=data.frame(p,number_of_rounds)
x_mean_dataframe=data.frame(x_mean,number_of_rounds)
Total_regret_dataframe=
data.frame(Total_regret,number_of_rounds)

#plots of perfomance measures
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=p))
+geom_line()
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=x_mean))
+geom_line()
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=Total_regret))
+geom_line()

#outputs
print(paste0("The percentage of optimal action chosen is
: ", p[M]))
print(paste0("The Total Regret is : ", Total_regret[M]))
print(paste0("The Sample Mean is : ", x_mean[M]))
```

4 ucb1_beta _sim

```
#ucb1_beta_sim
#ucb1 with simulation for comparison with epsilon-greedy

library("purrr")
library("ramify")
library("ggplot2")
```

```
# beta function

BETA<- function(i)
{if (i==1)
{BETA=rbeta(1,shape1 = 2, shape2 = 3)}
  else if (i==2)
  {BETA=rbeta(1,shape1 = 9, shape2 = 11)}
  else if (i==3)
  {BETA=rbeta(1,shape1 = 5, shape2 = 5)}
  else if (i==4)
  {BETA=rbeta(1,shape1 = 11, shape2 = 9)}
  else if (i==5)
  {BETA=rbeta(1,shape1 = 3, shape2 = 2)}
  else {BETA=0}
}

# beta parameters
SHAPE_1=mat("2,9,5,11,3")
SHAPE_2=mat("3,11,5,9,2")
arms_means=SHAPE_1/(SHAPE_1+SHAPE_2)
m_star=max(arms_means)
a_star=argmax(arms_means)

M_sim=1000
M=10000
k=length(SHAPE_1)
q=matrix(1:k,nrow=M_sim,ncol=k)*0
N=matrix(1:k,nrow=M_sim,ncol=k)*0
U=matrix(1:k,nrow=M_sim,ncol=k)*0
Upper_Bound=q+U
a=matrix(1:M,nrow=M_sim,ncol=M)*0
p=matrix(1:M,nrow=M_sim,ncol=M)*0
x_mean=matrix(1:M,nrow=M_sim,ncol=M)*0
Total_regret=matrix(1:M,nrow=M_sim,ncol=M)*0
p_average=matrix(1:M,nrow=1,ncol=M)*0
Total_regret_average=matrix(1:M,nrow=1,ncol=M)*0
x_mean_average=matrix(1:M,nrow=1,ncol=M)*0

for (z in 1:M_sim)
{
```

```
for (i in 1:k) {
  N[z,i]=N[z,i]+1
  q[z,i]=BETA(i)
  U[z,i]=sqrt(2*log(i)/N[z,i])
  Upper_Bound[z,i]=q[z,i]+U[z,i]
  x_mean[z,i]=sum(N[z,]*q[z,])/i
  a[z,i]=i
  if (i==1)
  {Total_regret[z,i]=m_star-arms_means[i]}
  else {Total_regret[z,i]=Total_regret[z,i-1]
  +m_star-arms_means[i]}
  if (i==a_star)
  {p[z,i]=1/a_star}
  else {p[z,i]=0}
}

for (j in (k+1):M)
{
  I=max.col(t(Upper_Bound[z,]))
  a[z,j]=I
  N[z,I]=N[z,I]+1
  q[z,I]=q[z,I]+(1/N[z,I])*(BETA(I)-q[z,I])
  U[z,I]=sqrt(2*log(i)/N[z,I])
  Upper_Bound[z,I]=q[z,I]+U[z,I]
  #percenage of the optimal action chosen
  p[z,j]=N[z,a_star]/j
  #sample mean
  x_mean[z,j]=(sum(N[z,]*q[z,])/(j))
  #total regret
  Total_regret[z,j]=Total_regret[z,j-1]+m_star-arms_means[I]
}
}

#average perfomance of measures in round i, i=1,...M_sim
for (i in 1:M)
{ p_average[i]=sum(p[,i])/M_sim
x_mean_average[i]=sum(x_mean[,i])/M_sim
Total_regret_average[i]=sum(Total_regret[,i])/M_sim
}

number_of_rounds=1:M
p_dataframe=data.frame(p,number_of_rounds)
```

```
x_mean_dataframe=data.frame(x_mean,number_of_rounds)
Total_regret_dataframe=
data.frame(Total_regret,number_of_rounds)

#plots of perfomance measures
ggplot(data=p_dataframe,aes(x=number_of_rounds,y=p_average))
+geom_line()
ggplot(data=x_mean_dataframe,
aes(x=number_of_rounds,y=x_mean_average))+geom_line()
ggplot(data=Total_regret_dataframe,
aes(x=number_of_rounds,y=Total_regret_average))+geom_line()

#M rounds
print(paste0("The percentage of optimal action chosen is :
", p_average[M]))
print(paste0("The Total Regret is : ",
Total_regret_average[M]))
print(paste0("The Sample Mean is : ", x_mean_average[M]))


#comparison with epsilon_greedy

#regrets
ggplot(data, aes(x=number_of_rounds)) +
  geom_line(aes(y = Total_regret_1), color = "darkred") +
  geom_line(aes(y = Total_regret_2), color="steelblue") +
  labs(y="average regret", x = "rounds")+
  geom_text(x=2500, y=60, label="ucb1",colour="darkred")+
  geom_text(x=2500, y=15, label="epsilon-greedy,
  epsilon=0.1",colour="steelblue")

#xmeans
ggplot(data_x, aes(x=number_of_rounds)) +
  geom_line(aes(y =x_mean_1), color = "darkred") +
  geom_line(aes(y =x_mean_2), color="steelblue")+
  labs(y="average reward", x = "rounds")+
  geom_text(x=1250, y=0.55, label="ucb1",colour="darkred")+
  geom_text(x=1250, y=0.595, label="epsilon-greedy,
  epsilon=0.1",colour="steelblue")
```

5 iterative _policy _evaluation

```
#iterative policy evaluation for example 3.3

V=matrix(0,4,4)
D=0
theta=0.00001
iterations=0
k=10^5
termination=F
while (termination==F) {
  u=V
  V[1,1]=0
  V[1,2]= 0.25*(-2+u[1,1])+0.25*(-2+u[1,2])
        + 0.25*(-2+u[1,3])+0.25*(-2+u[2,2])
  V[1,3]= 0.25*(-2+u[1,2])+0.25*(-2+u[1,3])
        + 0.25*(-2+u[1,4])+0.25*(-2+u[2,3])
  V[1,4]= 0.25*(-2+u[1,3])+0.25*(-2+u[1,4])
        + 0.25*(-2+u[1,4])+0.25*(-2+u[2,4])

  V[2,1]= 0.25*(-2+u[2,1])+0.25*(-2+u[1,1])
        + 0.25*(-2+u[2,2])+0.25*(-2+u[3,1])
  V[2,2]= 0.25*(-2+u[2,1])+0.25*(-2+u[1,2])
        + 0.25*(-2+u[2,3])+0.25*(-2+u[3,2])
  V[2,3]= 0.25*(-2+u[2,2])+0.25*(-2+u[1,3])
        + 0.25*(-2+u[2,4])+0.25*(-2+u[3,3])
  V[2,4]= 0.25*(-2+u[2,3])+0.25*(-2+u[1,4])
        + 0.25*(-2+u[2,4])+0.25*(-2+u[3,4])

  V[3,1]= 0.25*(-2+u[3,1])+0.25*(-2+u[2,1])
        + 0.25*(-2+u[3,2])+0.25*(-2+u[4,1])
  V[3,2]= 0.25*(-2+u[3,1])+0.25*(-2+u[2,2])
        + 0.25*(-2+u[3,3])+0.25*(-2+u[4,2])
  V[3,3]= 0.25*(-2+u[3,2])+0.25*(-2+u[2,3])
        + 0.25*(-2+u[3,4])+0.25*(-2+u[4,3])
  V[3,4]= 0.25*(-2+u[3,3])+0.25*(-2+u[2,4])
        + 0.25*(-2+u[3,4])+0.25*(-2+u[4,4])

  V[4,1]= 0.25*(-2+u[4,1])+0.25*(-2+u[3,1])
        + 0.25*(-2+u[4,2])+0.25*(-2+u[4,1])
  V[4,2]= 0.25*(-2+u[4,1])+0.25*(-2+u[3,2])
        + 0.25*(-2+u[4,3])+0.25*(-2+u[4,2])
```

```
  V[4,3]= 0.25*(-2+u[4,2])+0.25*(-2+u[3,3])
        + 0.25*(-2+u[4,4])+0.25*(-2+u[4,3])
  V[4,4]=0
  D=max(abs(u-V))
  iterations=iterations+1
  if ((D<theta) || (iterations >=k)){termination=T}
}
V
```

6 policy_iteration

```
#policy iteration for example 3.5

V=matrix(0,4,4)
policy=matrix(0.25,15,4)
D=0
theta=0.00001
k=10^5
policy_stable=T
while (policy_stable==T){
termination=F
iterations=0
while (termination==F) {
  u=V
  V[1,1]=0
  V[1,2]= policy[1,1]*(-1+u[1,1])+policy[1,2]*(-10+u[1,2])
        + policy[1,3]*(-10+u[1,3])+policy[1,4]*(-1+u[2,2])
  V[1,3]= policy[2,1]*(-10+u[1,2])+policy[2,2]*(-10+u[1,3])
        + policy[2,3]*(-10+u[1,4])+policy[2,4]*(-1+u[2,3])
  V[1,4]= policy[3,1]*(-10+u[1,3])+policy[3,2]*(-10+u[1,4])
        + policy[3,3]*(-10+u[1,4])+policy[3,4]*(-1+u[2,4])

  V[2,1]= policy[4,1]*(-1+u[2,1])+policy[4,2]*(-1+u[1,1])
        + policy[4,3]*(-1+u[2,2])+policy[4,4]*(-10+u[3,1])
  V[2,2]= policy[5,1]*(-1+u[2,1])+policy[5,2]*(-10+u[1,2])
        + policy[5,3]*(-1+u[2,3])+policy[5,4]*(-10+u[3,2])
  V[2,3]= policy[6,1]*(-1+u[2,2])+policy[6,2]*(-10+u[1,3])
        + policy[6,3]*(-1+u[2,4])+policy[6,4]*(-10+u[3,3])
  V[2,4]= policy[7,1]*(-1+u[2,3])+policy[7,2]*(-10+u[1,4])
        + policy[7,3]*(-1+u[2,4])+policy[7,4]*(-1+u[3,4])
```

```
  V[3,1]= policy[8,1]*(-10+u[3,1])+policy[8,2]*(-1+u[2,1])
        + policy[8,3]*(-10+u[3,2])+policy[8,4]*(-1+u[4,1])
  V[3,2]= policy[9,1]*(-10+u[3,1])+policy[9,2]*(-1+u[2,2])
        + policy[9,3]*(-10+u[3,3])+policy[9,4]*(-1+u[4,2])
  V[3,3]= policy[10,1]*(-10+u[3,2])+policy[10,2]*(-1+u[2,3])
        + policy[10,3]*(-1+u[3,4])+policy[10,4]*(-1+u[4,3])
  V[3,4]= policy[11,1]*(-10+u[3,3])+policy[11,2]*(-1+u[2,4])
        + policy[11,3]*(-1+u[3,4])+policy[11,4]*(-1+u[4,4])

  V[4,1]= policy[12,1]*(-1+u[4,1])+policy[12,2]*(-10+u[3,1])
        + policy[12,3]*(-1+u[4,2])+policy[12,4]*(-1+u[4,1])
  V[4,2]= policy[13,1]*(-1+u[4,1])+policy[13,2]*(-10+u[3,2])
        + policy[13,3]*(-1+u[4,3])+policy[13,4]*(-1+u[4,2])
  V[4,3]= policy[14,1]*(-1+u[4,2])+policy[14,2]*(-10+u[3,3])
        + policy[14,3]*(-1+u[4,4])+policy[14,4]*(-1+u[4,3])
  V[4,4]= policy[15,1]*(-1+u[4,3])+policy[15,2]*(-1+u[3,4])
        + policy[15,3]*(-1+u[4,4])+policy[15,4]*(-1+u[4,4])
  D=max(abs(u-V))
  iterations=iterations+1
  if ((D<theta) || (iterations >=k)){termination=T}
}

old_policy=policy
#greedy policy in state 1
for (j in 1:4){
if (j==max.col(matrix(c(-1+u[1,1],-10+u[1,2],-10+u[1,3],
-1+u[2,2]),1,4),ties.method = "first"))
policy[1,j]=1
else policy[1,j]=0
}

#greedy policy in state 2
for (j in 1:4){
  if (j==max.col(matrix(c(-10+u[1,2],-10+u[1,3],
  -10+u[1,4],-1+u[2,3]),1,4),ties.method = "first"))
    policy[2,j]=1
  else policy[2,j]=0
}

#greedy policy in state 3
for (j in 1:4){
```

```
  if (j==max.col(matrix(c(-10+u[1,3],-10+u[1,4],-10+u[1,4],
  -1+u[2,4]),1,4),ties.method = "first"))
    policy[3,j]=1
  else policy[3,j]=0
}

#greedy policy in state 4
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[2,1],-1+u[1,1],-1+u[2,2],
  -10+u[3,1]),1,4),ties.method = "first"))
    policy[4,j]=1
  else policy[4,j]=0
}

#greedy policy in state 5
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[2,1],-10+u[1,2],-1+u[2,3],
  -10+u[3,2]),1,4),ties.method = "first"))
    policy[5,j]=1
  else policy[5,j]=0
}

#greedy policy in state 6
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[2,2],-10+u[1,3],-1+u[2,4],
  -10+u[3,3]),1,4),ties.method = "first"))
    policy[6,j]=1
  else policy[6,j]=0
}

#greedy policy in state 7
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[2,3],-10+u[1,4],-1+u[2,4],
  -1+u[3,4]),1,4),ties.method = "first"))
    policy[7,j]=1
  else policy[7,j]=0
}

#greedy policy in state 8
for (j in 1:4){
  if (j==max.col(matrix(c(-10+u[3,1],-1+u[2,1],-10+u[3,2],
  -1+u[4,1]),1,4),ties.method = "first"))
```

```
    policy[8,j]=1
  else policy[8,j]=0
}

#greedy policy in state 9
for (j in 1:4){
  if (j==max.col(matrix(c(-10+u[3,1],-1+u[2,2],-10+u[3,3],
  -1+u[4,3]),1,4),ties.method = "first"))
    policy[9,j]=1
  else policy[9,j]=0
}

#greedy policy in state 10
for (j in 1:4){
  if (j==max.col(matrix(c(-10+u[3,2],-1+u[2,3],-1+u[3,4],
  -1+u[4,3]),1,4),ties.method = "first"))
    policy[10,j]=1
  else policy[10,j]=0
}

#greedy policy in state 11
for (j in 1:4){
  if (j==max.col(matrix(c(-10+u[3,3],-1+u[2,4],-1+u[3,4],
  -1+u[4,4]),1,4),ties.method = "first"))
    policy[11,j]=1
  else policy[11,j]=0
}

#greedy policy in state 12
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[4,1],-10+u[3,1],-1+u[4,2],
  -1+u[4,1]),1,4),ties.method = "first"))
    policy[12,j]=1
  else policy[12,j]=0
}

#greedy policy in state 13
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[4,1],-10+u[3,2],-1+u[4,3],
  -1+u[4,2]),1,4),ties.method = "first"))
    policy[13,j]=1
  else policy[13,j]=0
```

```
}

#greedy policy in state 14
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[4,2],-10+u[3,3],-1+u[4,4],
  -1+u[4,3]),1,4),ties.method = "first"))
    policy[14,j]=1
  else policy[14,j]=0
}

#greedy policy in state 15
for (j in 1:4){
  if (j==max.col(matrix(c(-1+u[4,3],-1+u[3,4],-1+u[4,4],
  -1+u[4,4]),1,4),ties.method = "first"))
    policy[15,j]=1
  else policy[15,j]=0
}

if (identical(old_policy,policy)==T)
  policy_stable=F
print(V)
print(policy)
}
```

Βιβλιογραφία

[1] Richard S. Sutton, Andrew G. Barto, Reinforcement Learning: An Introduction, 2nd edition, The MIT Press (2018)

[2] Miroslav Kubat, An Introduction to Machine Learning, 2nd edition, Springer (2017)

[3] Tor Lattimore, Csaba Szepesvari, Bandit Algorithms, Cambridge University Press

[4] Dimitri P. Bertsekas, Dynamic Programming and Optimal Control, 3rd edition, Athena Scientifica (2010)

[5] Dimitri P. Bertsekas and John Tsitsiklis, Neuro-Dynamic Programming, Athena Scientifica (1996)

[6] Peter Auer, Nicolo Cesa-Bianchi, Paul Fisher, Finite-time Analysis of the Multiarmed Bandit Problem, Kluwer Academic Publishers (2002)

[7] Frank Rosenblatt, The Perceptron: A Perceiving and Recognizing Automaton, Cornell Aeronautical Laboratory, INC. (1957)

[8] Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell, Machine Learning: An Artificial Intelligence Approach, Elsevier Science (1983)

[9] William R. Thompson, On The Likelihood That One Unknown Probability Exceeds Another In View Of The Evidence of Two Samples, Biometrica (1933)

[10] T. Lai, H. Robbins, Asymptotically efficient adaptive allocation rules, Advances in Applied Mathematics 6 (1985)

[11] Nikos Stamatis, Mathematical properties of the Stochastic Approximation and the multi-armed bandit problem (2020)