ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Δημιουργία Υποσυστήματος Εκμάθησης των Ρυθμιστικών Παραμέτρων ενός Νευρωνικού Δικτύου με χρήση Ενισχυτικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΓΕΡΑΣΙΜΟΣ Ε. ΣΠΑΝΑΚΗΣ

Επιβλέπων : Ανδρέας-Γεώργιος Σταφυλοπάτης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2006

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Δημιουργία Υποσυστήματος Εκμάθησης των Ρυθμιστικών Παραμέτρων ενός Νευρωνικού Δικτύου με χρήση Ενισχυτικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΓΕΡΑΣΙΜΟΣ Ε. ΣΠΑΝΑΚΗΣ

Επιβλέπων : Ανδρέας-Γεώργιος Σταφυλοπάτης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10η Οκτωβρίου 2006.

.....	.....	.....
Ανδρέας–Γεώργιος Σταφυλοπάτης Καθηγητής Ε.Μ.Π.	Στέφανος Κόλλιας Καθηγητής Ε.Μ.Π.	Παναγιώτης Τσανάκας Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2006

.....

ΓΕΡΑΣΙΜΟΣ Ε. ΣΠΑΝΑΚΗΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η παρούσα διπλωματική εργασία μελετά την εφαρμογή της ενισχυτικής μάθησης -και πιο συγκεκριμένα της μάθησης Q- στην προσαρμογή των παραμέτρων μάθησης ενός νευρωνικού δικτύου-ταξινομητή, με στόχο τη βελτίωση της απόδοσης και της ταχύτητάς του. Η μάθηση Q είναι μια μορφή ενισχυτικής μάθησης χωρίς μοντέλο και δίνει έναν απλό τρόπο επίλυσης προβλημάτων, ο οποίος βασίζεται στη συνάρτηση δράσης-αποτίμησης Q, η οποία απεικονίζει ζεύγη κατάστασης-δράσης σε αναμενόμενες τιμές ανταμοιβής.

Σε έναν ταξινομητή, καθώς περνάει ο χρόνος, μόνο η διαδικασία εκπαίδευσης είναι αυτή που αλλάζει, συνεπώς η χρησιμότητα του ελέγχου των παραμέτρων της μάθησης ώστε να προσαρμόζονται κάθε φορά στις νέες συνθήκες θα βοηθούσε στη δημιουργία ενός καλύτερου συστήματος ταξινόμησης.

Μελετήθηκε η προσαρμογή του ρυθμού μάθησης του συστήματος ενώ με ανάλογο τρόπο μπορεί να επεκταθεί η μέθοδος και σε άλλες κρίσιμες παραμέτρους της μάθησης (όπως ο αριθμός των εποχών). Η μέθοδος εφαρμόστηκε σε διάφορα σύνολα δεδομένων και έγινε σύγκριση των αποτελεσμάτων με σύστημα που δε χρησιμοποιεί την προσαρμογή του ρυθμού μάθησης.

Λέξεις Κλειδιά: <<Ενισχυτική μάθηση, μάθηση Q, προσαρμογή ρυθμού μάθησης, μάθηση μηχανής>>

Abstract

This diploma thesis deals with the application of reinforcement learning –and especially Q-learning- to the adaptation of learning parameters of a neural network, aiming at improving both its performance and its speed. Q-learning is a form of model-free reinforcement learning and provides us with a simple way of solving problems. It is based on the Q "action-evaluation" function that maps pairs of states and actions to estimated values of reward.

Given a classifier based on neural networks, the only thing that changes during time, is the iterative learning procedure. Thus, the control of the learning parameters, aiming to adapt them to the changing conditions of the environment, would be very useful for the development of a better classification system.

This thesis focuses on the adaptation of the learning rate parameter, but the method can be applied to other critical learning parameters in a similar fashion. The method was tested upon different data sets and was compared with the results of a system that does not use learning rate adaptation.

Keywords: <<Reinforcement Learning, Q-learning, learning rate adaptation, machine learning>>

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τα ακόλουθα πρόσωπα :

- Τον **καθηγητή Ανδρέα-Γεώργιο Σταφυλοπάτη** για την ανάθεση αυτής της διπλωματικής εργασίας και για τη βοήθεια και καθοδήγηση του τόσο κατά τη διάρκεια εκπόνησης της εργασίας όσο και κατά τη διάρκεια των φοιτητικών μου χρόνων.

- Τον **υποψήφιο διδάκτορα Μηνά Περτσελάκη** για τις γνώσεις που μου μετέδωσε σχετικά με το αντικείμενο της διπλωματικής εργασίας, για το χρόνο που διέθεσε καθώς και για τη σημαντική βοήθεια και συνεργασία ώστε να επιτευχθεί το καλύτερο αποτέλεσμα για την εργασία αυτή.

- Τους **γονείς μου** για την κάθε είδους υποστήριξη και συμπαράσταση που μου παρείχαν σε όλη τη διάρκεια της φοιτητικής μου διαδρομής.

- Τους **φίλους μου** και τα **αγαπημένα μου πρόσωπα** για τη στήριξη τους.

Περιεχόμενα

Σχήματα

Πίνακες

1 *Εισαγωγή*

1.1 Αντικείμενο της διπλωματικής

Η διαδικασία εκπαίδευσης ενός δικτύου περιλαμβάνει συνήθως τις ελεύθερες παραμέτρους του συστήματος, που αλλιώς ονομάζονται και βάρη. Διαδικασία εκμάθησης των παραμέτρων που ρυθμίζουν τη λειτουργία εκπαίδευσης και που συνήθως προκαθορίζονται από τον δημιουργό του συστήματος, όπως π.χ. ο ρυθμός μάθησης ή ο αριθμός των εποχών, δεν υφίσταται.

Η ενισχυτική μάθηση αποτελεί μια μορφή μη-επιβλεπόμενης εκπαίδευσης ενός συστήματος που βασίζεται στο ενισχυτικό σήμα που δέχεται από το περιβάλλον. Το σήμα αυτό χαρακτηρίζει το πόσο «καλό» ήταν το αποτέλεσμα και όχι αν ήταν σωστό ή λάθος. Παράλληλα, η ενισχυτική μάθηση απαιτεί τα δεδομένα της να μεταβάλλονται συναρτήσει του χρόνου και έχει την δυνατότητα να μαθαίνει ακόμα και αν ένα ανεπιθύμητο αποτέλεσμα εμφανιστεί πολλά βήματα μετά από την πράξη που το δημιούργησε. Για τους δύο παραπάνω λόγους, η ενισχυτική μάθηση έχει βρει εφαρμογή κυρίως στη ρομποτική και τα προβλήματα ελέγχου, και όχι τόσο στον τομέα της μάθησης μηχανής και της αναγνώρισης προτύπων.

Στην παρούσα διπλωματική εργασία φιλοδοξούμε να αναστρέψουμε την κατάσταση αυτή και να παρουσιάσουμε έναν τρόπο με τον οποίο η ενισχυτική μάθηση μπορεί να συνεισφέρει στο κομμάτι της εκπαίδευσης ενός ταξινομητή. Επιλέγουμε το κομμάτι της εκπαίδευσης καθώς είναι το μόνο στάδιο της δημιουργίας ενός ταξινομητή που περιλαμβάνει χρονική μεταβολή (βελτίωση μάθησης συναρτήσει του χρόνου). Με άλλα λόγια, στόχος είναι η «εκμάθηση του τρόπου μάθησης», με εύρεση της βέλτιστης πολιτικής για τη διαδικασία αυτή. Για την εύρεση της βέλτιστης πολιτικής

χρησιμοποιούμε τεχνικές ενισχυτικής μάθησης και πιο συγκεκριμένα τη μάθηση Q. Το αποτέλεσμα που επιδιώκουμε είναι η επιλογή των κατάλληλων παραμέτρων εκπαίδευσης, όπως και η επιτάχυνση της διαδικασίας μάθησης.

Η συγκεκριμένη εργασία επέλεξε να εφαρμόσει τα παραπάνω και να μελετήσει τα αντίστοιχα αποτελέσματα, στο ρυθμό μάθησης, παράμετρο ιδιαίτερα βασική για την εκπαίδευση του ταξινομητή. Παρόλα αυτά, αισιοδοξούμε πως η όλη ιδέα μπορεί να βρει εφαρμογή και σε άλλες παραμέτρους του συστήματος όπως ο αριθμός των κρυμμένων κόμβων, ο όρος ορμής, κλπ

1.2 Διάρθρωση της εργασίας

Η διπλωματική εργασία αποτελείται από 5 ακόμη κεφάλαια.

Στο δεύτερο κεφάλαιο περιγράφονται θέματα που σχετίζονται με την ενισχυτική μάθηση και τις ως τώρα τεχνικές που έχουν αναπτυχθεί με έμφαση στη χρησιμοποιούμενη από την εργασία μάθηση Q.

Στο τρίτο κεφάλαιο αναλύεται η λειτουργία ενός ταξινομητή και περιγράφονται οι παράμετροι οι οποίες θέλουμε να ελέγξουμε και να τροποποιήσουμε κατά τη διάρκεια της μάθησης. Παρουσιάζονται επίσης υπάρχουσες τεχνικές για τη μεταβολή του ρυθμού μάθησης.

Στο τέταρτο κεφάλαιο περιγράφεται πιο συγκεκριμένα η μεθοδολογία που ακολουθήθηκε, καθώς και οι δυσκολίες που εμφανίστηκαν κατά την ανάπτυξη της εφαρμογής, ενώ παράλληλα παρουσιάζεται η διαδικασία υλοποίησής της.

Στο πέμπτο κεφάλαιο παρουσιάζεται η πειραματική μελέτη που περιλαμβάνει εφαρμογή της μεθόδου σε συγκεκριμένα προβλήματα ταξινόμησης, όπως και συγκριτικά αποτελέσματα σε σχέση με τυπικές τεχνικές εκπαίδευσης.

Τέλος, στο έκτο κεφάλαιο συγκεντρώνονται και παρουσιάζονται τα συμπεράσματα που προέκυψαν από την εργασία αυτή, με στόχο την αξιολόγηση της επίδοσης της μεθόδου που μελετήσαμε και αναπτύξαμε, αλλά και την ενδεχόμενη μελλοντική βελτίωσή της.

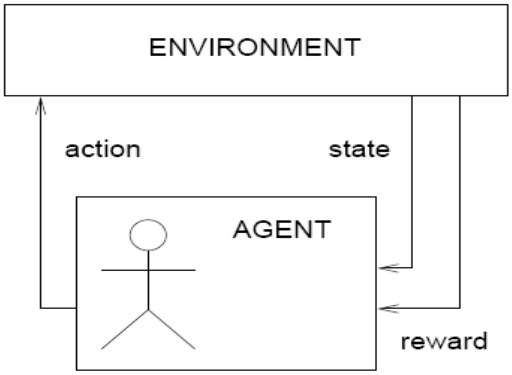
Στο παράρτημα παρατίθεται ο κώδικας της εφαρμογής.

2 *Ενισχυτική Μάθηση*

2.1 Εισαγωγή

Ο όρος ενισχυτική μάθηση αναφέρθηκε πρώτη φορά στην τεχνητή νοημοσύνη από τον Minski (1961) και ανεξάρτητα στη θεωρία ελέγχου από τους Waltz και Fu το ίδιο έτος [1]. Έχει τις αρχές της στις πρώτες μέρες ανάπτυξης του κυβερνοχώρου με τις έρευνες στην στατιστική, ψυχολογία, νευροεπιστήμη και επιστήμη των υπολογιστών. Τα τελευταία πέντε με δέκα χρόνια έχει προσελκύσει ραγδαία αυξανόμενο ενδιαφέρον στην κοινότητα της μηχανικής μάθησης και της τεχνητής νοημοσύνης [4]. Η υπόσχεση που δίνει η ενισχυτική μάθηση είναι δελεαστική – ένας τρόπος προγραμματισμού πρακτόρων (agents) με ανταμοιβή ή τιμωρία σε κάθε ενέργειά τους, χωρίς να χρειάζεται να προσδιοριστεί ο τρόπος που πρέπει να επιτευχθεί ο στόχος. Όπως είναι επακόλουθο, υπάρχουν διάφορα υπολογιστικά προβλήματα και εμπόδια στην επίτευξη αυτού του προγραμματισμού.

Η ενισχυτική μάθηση είναι το πρόβλημα που αντιμετωπίζεται από ένα πράκτορα, ο οποίος πρέπει να μάθει τη σωστή συμπεριφορά μέσα από δοκιμές με ένα δυναμικό περιβάλλον [7]. Το γενικό σχήμα της ενισχυτικής μάθησης φαίνεται στο σχήμα 1.



ΣΧΗΜΑ 1 : ΓΕΝΙΚΟ ΣΧΗΜΑ ΕΝΙΣΧΥΤΙΚΗΣ ΜΑΘΗΣΗΣ

Υπάρχουν δύο βασικές στρατηγικές για την αντιμετώπιση των προβλημάτων ενισχυτικής μάθησης. Η πρώτη είναι να ψάξει ο πράκτορας στο χώρο των συμπεριφορών που έχουμε ορίσει στο σύστημα και να βρει μία που να λειτουργεί καλά μέσα στο περιβάλλον. Η προσέγγιση αυτή έχει βρει πρόσφορο έδαφος στους γενετικούς αλγορίθμους και στο γενετικό προγραμματισμό. Η δεύτερη είναι να γίνει χρήση στατιστικών τεχνικών και μεθόδων δυναμικού προγραμματισμού ώστε να υπολογιστεί η χρησιμότητα της επιλογής όλων των δράσεων στις καταστάσεις του περιβάλλοντος που έχουμε ορίσει. Ακόμα δεν έχει καθοριστεί ποια τεχνική αποδίδει καλύτερα για τις διάφορες συνθήκες.

Η ενισχυτική μάθηση διαφέρει από το πιο διαδεδομένο και μελετημένο πρόβλημα της επιβλεπόμενης μάθησης σε πολλά σημεία. Η πιο σημαντική διαφορά είναι πως δεν υπάρχει παρουσίαση των ζευγών εισόδου/εξόδου. Αντίθετα, ο πράκτορας αφού επιλέξει μία δράση ενημερώνεται για την άμεση ανταμοιβή του και την επόμενη κατάσταση, αλλά δεν ενημερώνεται για το ποια είναι η καλύτερη δράση όσον αφορά το μακροπρόθεσμο ενδιαφέρον του στο πρόβλημα. Είναι πολύ σημαντικό για τον πράκτορα να συγκεντρώσει χρήσιμη εμπειρία για τις πιθανές καταστάσεις του συστήματος, μεταβάσεις μεταξύ αυτών και τις ανταμοιβές ώστε να λειτουργήσει βέλτιστα. Άλλη μια διαφορά από την επιβλεπόμενη μάθηση είναι πως η επί γραμμής λειτουργία (on-line performance) είναι σημαντική: Η επιβεβαίωση του συστήματος γίνεται κάποιες φορές ταυτόχρονα με τη μάθηση.

Η ενισχυτική μάθηση και ο δυναμικός προγραμματισμός είναι στενά συσχετισμένες έννοιες, μιας και οι δύο προσεγγίσεις χρησιμοποιούνται για την επίλυση αλυσίδων Markov. Η βασική ιδέα και των δύο είναι να μαθαίνεις τις συναρτήσεις τιμών, οι

οποίες μπορούν να χρησιμοποιηθούν για την εύρεση της βέλτιστης πολιτικής. Παρόλη αυτή τη στενή σχέση, υπάρχει μια βασική διαφορά μεταξύ ενισχυτικής μάθησης και δυναμικού προγραμματισμού. Στην πρώτη, ο πράκτορας δε γνωρίζει απαραίτητα τη συνάρτηση ανταμοιβής και τις συναρτήσεις μετάβασης καταστάσεων. Τόσο η ανταμοιβή όσο και η νέα κατάσταση από την επιλογή μίας δράσης καθορίζονται από το περιβάλλον και οι συνέπειες μιας δράσης πρέπει να εξαχθούν από την αλληλεπίδραση με το περιβάλλον. Με άλλα λόγια, οι πράκτορες της ενισχυτικής μάθησης δε χρειάζεται να γνωρίζουν ένα μοντέλο του δικού τους περιβάλλοντος. Το γεγονός αυτό διαχωρίζει την ενισχυτική μάθηση από το δυναμικό προγραμματισμό, στον οποίο η πλήρης γνώση της συνάρτησης ανταμοιβής και της συνάρτησης μετάβασης καταστάσεων είναι απαραίτητη. Για τη σύγκλιση σε μία βέλτιστη πολιτική, απαιτείται ένας αριθμός πεπερασμένων επαναλήψεων και όχι μετάβαση στο πραγματικό περιβάλλον και παρατήρηση των αποτελεσμάτων.

Κάποιες πλευρές της ενισχυτικής μάθησης συνδέονται στενά με θέματα έρευνας και προγραμματισμού στην τεχνητή νοημοσύνη. Οι ευρετικοί αλγόριθμοι της τεχνητής νοημοσύνης δημιουργούν μια επιτυχή πορεία μέσα από ένα γράφημα καταστάσεων. Ο προγραμματισμός λειτουργεί παρόμοια, αλλά ο γράφος καταστάσεων κατασκευάζεται πιο πολύπλοκα και σε αυτόν οι καταστάσεις αναπαριστώνται σαν συνθέσεις λογικών εκφράσεων αντί ατομικών συμβόλων. Από την άλλη πλευρά, η ενισχυτική μάθηση (τουλάχιστον για τις διακριτές περιπτώσεις που έχει αναπτυχθεί το θεωρητικό υπόβαθρο), υποθέτει πως ολόκληρος ο χώρος καταστάσεων μπορεί να απαριθμηθεί και να αποθηκευτεί στη μνήμη – μία υπόθεση από την οποία δε δεσμεύονται οι συμβατικοί ευρετικοί αλγόριθμοι.

2.2 Εκμετάλλευση ή εξερεύνηση : Το πρόβλημα της μιας κατάστασης

Η πιο απλή περίπτωση προβλήματος ενισχυτικής μάθησης είναι το γνωστό σαν το πρόβλημα των «*k*-κουλοχέρηδων», το οποίο έχει τροφοδοτήσει την έρευνα της στατιστικής και των εφαρμοσμένων μαθηματικών. Ο πράκτορας βρίσκεται σε ένα δωμάτιο με ένα σύνολο *k*-κουλοχέρηδων (τα γνωστά μηχανήματα τζόγου). Ο πράκτορας έχει το δικαίωμα να τραβήξει το μοχλό μόνο σε *h* μηχανήματα.

Οποιοδήποτε μηχάνημα μπορεί να επιλεγεί κάθε φορά. Το μόνο κόστος του πράκτορα είναι η σπατάλη μιας από τις *h* προσπάθειες εφόσον επιλέξει κάποιο μη-βέλτιστο μηχάνημα. Όταν το μηχάνημα *i* επιλεγεί, τότε αυτό δίνει σαν επιστροφή 1 ή 0 σύμφωνα με κάποια πιθανότητα p_i, με τα ενδεχόμενα των επιστροφών να είναι ασυμβίβαστα και τα $p_i s$ είναι άγνωστα. Το ερώτημα είναι ποια θα πρέπει να είναι η πολιτική του πράκτορα;

Το πρόβλημα αυτό παρουσιάζει τη βασική ανάγκη ισορρόπησης μεταξύ της εκμετάλλευσης και της εξερεύνησης [2]. Ο πράκτορας μπορεί να πιστεύει πως ένας συγκεκριμένος μοχλός έχει μία σημαντικά υψηλή πιθανότητα επιστροφής. Θα πρέπει να επιλέγει αυτό το μοχλό συνέχεια ή θα πρέπει να διαλέξει κάποιον άλλο για τον οποίο έχει λιγότερες πληροφορίες; Οι απαντήσεις σε αυτές τις ερωτήσεις εξαρτώνται από το πόσο χρόνο αναμένεται ο πράκτορας να παίξει το παιχνίδι. Όσο περισσότερο διαρκεί το παιχνίδι, τόσο χειρότερες είναι οι συνέπειες της προσκόλλησης σε κάποιο μη-βέλτιστο μηχάνημα και τόσο το περισσότερο πρέπει να εξερευνήσει ο πράκτορας.

Υπάρχουν πολλές στρατηγικές για την επίλυση του συγκεκριμένου προβλήματος. Ο όρος «δράση» θα χρησιμοποιηθεί για την επιλογή μοχλού του πράκτορα. Είναι σημαντικό να σημειωθεί πως το συγκεκριμένο πρόβλημα αποτελεί και τον ορισμό ενός περιβάλλοντος ενισχυτικής μάθησης με μία κατάσταση και με μεταβάσεις μόνο μέσα σε αυτήν. Το πρόβλημα επιδέχεται καταρχάς λύσεις με αποτελέσματα τα οποία είναι τυπικώς ορθά (δυναμικός προγραμματισμός, αυτόματα μάθησης, προσέγγιση Gittins με ευρετήρια κατανομής). Παρότι οι λύσεις αυτές μπορούν να επεκταθούν σε προβλήματα με πραγματικές τιμές ανταμοιβών, δε μπορούν να εφαρμοστούν άμεσα στα προβλήματα ενισχυτικής μάθησης με πολλές καταστάσεις. Επίσης, υπάρχουν λύσεις που δεν είναι τυπικά επιβεβαιωμένες και ορισμένες αλλά έχουν μεγάλη χρήση και μπορούν να εφαρμοστούν (με μία επιφυλακτικότητα ως προς την ασφάλεια των αποτελεσμάτων) στη γενική περίπτωση. Τέτοιες λύσεις είναι οι : «άπληστη» τεχνική, τυχαία επιλογή βάσει πιθανοτήτων, τεχνικές βασισμένες στα διαστήματα.

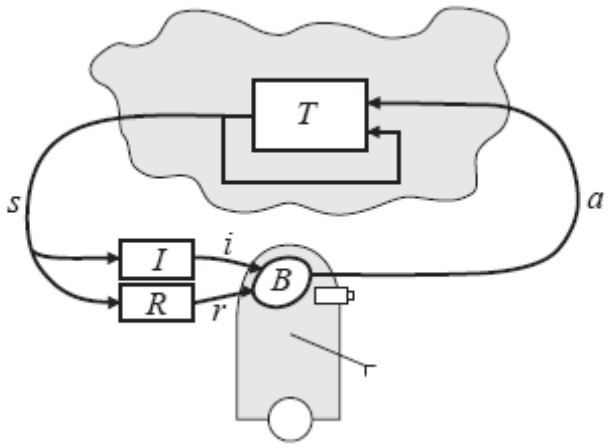
2.3 Γενικότερα προβλήματα – Καθυστερημένη ανταμοιβή

Στη γενική περίπτωση των προβλημάτων ενισχυτικής μάθησης, οι δράσεις του πράκτορα ορίζουν όχι μόνο την άμεση ανταμοιβή του αλλά και την (τουλάχιστον πιθανοτικά) επόμενη κατάσταση του περιβάλλοντος. Τέτοια περιβάλλοντα μπορούν

να θεωρηθούν ως δίκτυα προβλημάτων κουλοχέρηδων αλλά ο πράκτορας πρέπει να λάβει υπόψιν την επόμενη κατάσταση όπως επίσης και την άμεση ανταμοιβή, όταν αποφασίσει ποια δράση θα επιλέξει. Ο πράκτορας πρέπει να είναι σε θέση να μάθει από την καθυστερημένη ανταμοιβή : μπορεί να χρειαστεί μια μεγάλη σειρά από δράσεις που θα δίνουν ασήμαντη ενίσχυση αλλά τελικά θα καταλήγει σε μία κατάσταση με μεγάλη ενίσχυση. Ο πράκτορας πρέπει να είναι ικανός να μάθει ποιες από τις δράσεις είναι επιθυμητές βάσει της ανταμοιβής που μπορεί να λάβει χώρα αυθαίρετα στο μέλλον.

2.4 Το μοντέλο ενισχυτικής μάθησης

Στο κλασσικό μοντέλο ενισχυτικής μάθησης, ένας πράκτορας συνδέεται στο περιβάλλον μέσω της αντίληψης και της δράσης, όπως απεικονίζεται στο σχήμα 2 [11].



ΣΧΗΜΑ 2 : ΚΛΑΣΣΙΚΟ ΜΟΝΤΕΛΟ ΕΝΙΣΧΥΤΙΚΗΣ ΜΑΘΗΣΗΣ

Σε κάθε βήμα της αλληλεπίδρασης ο πράκτορας λαμβάνει σαν είσοδο *i* κάποια ένδειξη της παρούσας κατάστασης *s* του περιβάλλοντος. Έπειτα, ο πράκτορας επιλέγει μία δράση *α* για να δημιουργήσει έξοδο. Η δράση αλλάζει την κατάσταση του περιβάλλοντος και η αποτίμηση αυτής της αλλαγής κατάστασης δίδεται στον πράκτορα σαν ενίσχυση *r*. Η συμπεριφορά του πράκτορα *B* θα έπρεπε να επιλέγει δράσεις που τείνουν να αυξάνουν το μακροπρόθεσμο άθροισμα των τιμών του

σήματος ενίσχυσης. Μπορεί να μάθει να το κάνει αυτό, έπειτα από συστηματικές προσπάθειες, καθοδηγούμενες από μία μεγάλη ποικιλία αλγορίθμων.

Τυπικά το μοντέλο αποτελείται από :

- Ένα διακριτό σύνολο καταστάσεων του περιβάλλοντος, *S*,
- Ένα διακριτό σύνολο δράσεων του πράκτορα, *A*,
- Ένα σύνολο γραμμικών σημάτων ενίσχυσης, συνήθως {0,1} ή πραγματικών αριθμών

Στο σχήμα επίσης φαίνεται μία συνάρτηση εισόδου *I*, η οποία καθορίζει πως ο πράκτορας βλέπει την εκάστοτε κατάσταση του περιβάλλοντος. Συνήθως θεωρείται πως αυτή είναι και η ταυτοτική συνάρτηση, ότι δηλαδή ο πράκτορας έχει αντίληψη της ακριβούς κατάστασης του περιβάλλοντος.

Ένας διαισθητικός τρόπος για την κατανόηση της σχέσης μεταξύ περιβάλλοντος και πράκτορα είναι το ακόλουθο παράδειγμα διαλόγου :

Περιβάλλον : Είσαι στην κατάσταση 65. Έχεις 4 πιθανές δράσεις

Πράκτορας : Θα διαλέξω τη δράση 2

Περιβάλλον : Έλαβες ενίσχυση 7 μονάδων. Τώρα είσαι στην κατάσταση 15 και έχεις 2 πιθανές δράσεις

Πράκτορας : Θα διαλέξω τη δράση 1

Περιβάλλον : Έλαβες ενίσχυση -4 μονάδων. Τώρα είσαι στην κατάσταση 65 και έχεις 4 πιθανές δράσεις

Πράκτορας : Θα διαλέξω τη δράση 2

Περιβάλλον : Έλαβες ενίσχυση 5 μονάδων. Τώρα είσαι στην κατάσταση 44 και έχεις 5 πιθανές δράσεις

... ...

Η δουλειά του πράκτορα είναι να βρει μία πολιτική π, αντιστοιχίζοντας καταστάσεις σε δράσεις, η οποία να μεγιστοποιεί, το μακροπρόθεσμο ποσό της ενίσχυσης. Γι' αυτό ακριβώς το λόγο, ο πράκτορας δύναται να θυσιάσει την άμεση ενίσχυση με στόχο να λάβει μεγαλύτερη ανταμοιβή αργότερα. Γενικά αναμένεται, ότι το

περιβάλλον θα είναι μη-ντετερμινιστικό, το οποίο σημαίνει ότι επιλέγοντας την ίδια δράση στην ίδια κατάσταση για δύο διαφορετικές περιστάσεις, μπορεί να οδηγήσει σε διαφορετικές επόμενες καταστάσεις και / ή διαφορετικές τιμές του σήματος ενίσχυσης. Αυτό συμβαίνει και στο διάλογο που περιγράφεται παραπάνω. Από την κατάσταση 65 και επιλέγοντας τη δράση 2, έχουμε δύο διαφορετικές περιπτώσεις επόμενων καταστάσεων και σημάτων ενίσχυσης. Παρόλα αυτά, υποθέτουμε πως το περιβάλλον είναι στάσιμο, δηλαδή οι πιθανότητες μετάβασης καταστάσεων ή λήψης σημάτων ενίσχυσης δεν αλλάζουν στο πέρασμα του χρόνου.

Η αναμενόμενη τιμή της συσσωρευμένης ενίσχυσης που επιτυγχάνεται από μία αυθαίρετη πολιτική π, ξεκινώντας από μία αυθαίρετη αρχική κατάσταση s_t δίνεται από τον τύπο :

$$V^{\pi}(s_t) = E\left[\sum_{i=0}^{\infty} \gamma^i r_{t+1}\right] \tag{2.1}$$

Όπου το r_{t+1} είναι η ανταμοιβή που λαμβάνεται επιλέγοντας μία δράση στην κατάσταση s_{t+1} χρησιμοποιώντας την πολιτική π και $\gamma \in [0,1)$ είναι ο παράγοντας έκπτωσης που καθορίζει τη σχετική τιμή της καθυστερημένης έναντι της άμεσης ανταμοιβής [9]. Η αναμενόμενη τιμή (συμβολισμός *E*) είναι απαραίτητη γιατί οι ανταμοιβές μπορεί να είναι μη-ντετερμινιστικές. Ανταμοιβές που λαμβάνονται *i* χρονικά βήματα μετά, υπόκεινται σε μία έκπτωση γ^i . Αν $\gamma=0$, μόνο οι άμεσες ανταμοιβές λαμβάνονται υπόψιν. Όσο το γ πλησιάζει στο 1, η έμφαση δίνεται περισσότερο στις μελλοντικές ανταμοιβές σε σχέση με τις άμεσες. Η συνάρτηση V^{π} καλείται συνάρτηση κατάστασης-τιμής (state-value function) για την πολιτική π. Στη βιβλιογραφία επίσης, αναφέρεται συχνά σαν συνάρτηση χρησιμότητας (utility function).

Χρησιμοποιώντας τη συνάρτηση V^{π} η διαδικασία της μάθησης μπορεί να οριστεί ως ακολούθως. Ο πράκτορας πρέπει να μάθει μία βέλτιστη πολιτική, δηλαδή μια πολιτική π^* η οποία μεγιστοποιεί την V^{π} για όλες τις καταστάσεις s :

$$\pi^* = \arg\max_{\pi} V^{\pi}(s) \tag{2.2}$$

Προς απλοποίηση του συμβολισμού, ορίζουμε τη συνάρτηση $V^{\pi^*}(s)$ μιας τέτοιας βέλτιστης πολιτικής σαν $V^*(s)$. Η συνάρτηση αυτή καλείται και συνάρτηση βέλτιστης τιμής.

Η διαδικασία μάθησης, όπως αντιμετωπίζεται από έναν πράκτορα ενισχυτικής μάθησης, συνήθως λαμβάνεται σαν μία αλυσίδα αποφάσεων Markov (Markov decision process, MDP). Σε μία τέτοια αλυσίδα, και οι μεταβάσεις μεταξύ καταστάσεων και οι ανταμοιβές εξαρτώνται αποκλειστικά από την παρούσα κατάσταση και την παρούσα δράση. Δεν υπάρχει εξάρτηση από προηγούμενες δράσεις ή καταστάσεις. Αυτό είναι γνωστό και σαν αρχή Markov ή αρχή ανεξαρτησίας των δρόμων. Έτσι, η ανταμοιβή και η νέα κατάσταση ορίζονται ως εξής :

$$\begin{aligned} r_t &= r(s_t, a_t) \\ s_{t+1} &= \delta(s_t, a_t) \end{aligned} \tag{2.3}$$

Η συνάρτηση ανταμοιβής r και η συνάρτηση μετάβασης καταστάσεων δ μπορεί να είναι μη-ντετερμινιστικές.

Αν ο πράκτορας ήξερε τη συνάρτηση βέλτιστης τιμής V^*, τις πιθανότητες μετάβασης μεταξύ καταστάσεων και τις αναμενόμενες ανταμοιβές, θα μπορούσε εύκολα να ορίσει τη βέλτιστη δράση, εφαρμόζοντας την αρχή της μέγιστης αναμενόμενης τιμής, δηλαδή μεγιστοποιώντας το άθροισμα της αναμενόμενης άμεσης ανταμοιβής και της αναμενόμενης τιμής της επόμενης κατάστασης, η οποία τιμή δείχνει τις αναμενόμενες ανταμοιβές από το σημείο αυτό και έπειτα.

$$\begin{aligned} \pi^*(s) &= \arg\max_{a \in A} E[r(s,a) + \gamma V^*(\delta(s,a))] \\ &= \arg\max_{a \in A} \left(E[r(s,a)] + \gamma \sum_{s' \in S} T(s,a,s') V^*(s') \right) \end{aligned} \tag{2.4}$$

όπου το *T(s,a,s')* συμβολίζει την πιθανότητα μετάβασης από την κατάσταση *s* στην *s'* επιλέγοντας τη δράση *α*.

Ας σημειωθεί πως οι τιμές μιας κατάστασης και της επόμενής της σχετίζονται ως ακολούθως:

$$\begin{aligned} V^*(s) &= E[r(s, \pi^*(s)) + \gamma V^*(\delta(s, \pi^*(s)))] \\ &= \max_{a \in A} \left(E[r(s,a)] + \gamma \sum_{s' \in S} T(s,a,s') V^*(s') \right) \end{aligned} \tag{2.5}$$

Οι τελευταίες εξισώσεις είναι γνωστές σαν εξισώσεις Bellman. Λύνοντας αυτές τις εξισώσεις (μία για κάθε κατάσταση) παίρνουμε μία μοναδική τιμή για κάθε κατάσταση. Δυστυχώς, λόγω της παρουσίας του τελεστή «*max*», οι εξισώσεις είναι μη-γραμμικές και επομένως δύσκολο να λυθούν. Ο συνήθης τρόπος για την επίλυση

αυτών των εξισώσεων είναι με τεχνικές δυναμικού προγραμματισμού όπως επανάληψη τιμής (value iteration) και επανάληψη πολιτικής (policy iteration).

2.5 Επανάληψη τιμής (Value Iteration) και Επανάληψη πολιτικής (Policy Iteration)

Η επανάληψη πολιτικής και η επανάληψη τιμής είναι δύο δημοφιλείς αλγόριθμοι δυναμικού προγραμματισμού. Καθεμιά από αυτές τις μεθόδους μπορεί να χρησιμοποιηθεί για τον αξιόπιστο υπολογισμό μιας βέλτιστης πολιτικής για μία πεπερασμένη αλυσίδα Markov (όταν υπάρχει πλήρη γνώση των χαρακτηριστικών της δηλαδή γνωρίζουμε το μοντέλο).

2.5.1 Επανάληψη τιμής (Value Iteration)

Ένας τρόπος για την εύρεση της βέλτιστης πολιτικής είναι η εύρεση της βέλτιστης τιμής συνάρτησης [5]. Αυτό μπορεί να καθοριστεί από έναν απλό επαναληπτικό αλγόριθμο, και εύκολα μπορεί να αποδειχθεί πως συγκλίνει στη σωστή τιμή V^*.

Αρχικοποίησε την V(s) αυθαίρετα

Επανέλαβε μέχρι η πολιτική να είναι αρκετά καλή

  Επανέλαβε για όλα τα s που ανήκουν στο S

    Επανέλαβε για όλα τα α που ανήκουν στο A

$$Q(s,a) := R(s,a) + \gamma \sum_{s' \in S} T(s,a,s')V(s')$$

$$V(s) := \max_a Q(s,a)$$

  τέλος επανάληψης

τέλος επανάληψης

ΣΧΗΜΑ 3 : ΑΛΓΟΡΙΘΜΟΣ ΕΠΑΝΑΛΗΨΗΣ ΤΙΜΗΣ (VALUE ITERATION)

Δεν είναι προφανές όμως πότε πρέπει να σταματήσει ο αλγόριθμος. Αποτελεσματικά κριτήρια για το σταμάτημα του αλγορίθμου έχουν βρεθεί κατά καιρούς (Bellman

residual, Williams & Baird, 1993b). Ο αλγόριθμος είναι πολύ ευέλικτος. Οι αναθέσεις τιμών στην V δε χρειάζεται να γίνονται με αυστηρή σειρά όπως φαίνεται στον αλγόριθμο αλλά μπορούν να γίνουν ασύγχρονα και παράλληλα με την προϋπόθεση πως η τιμή κάθε κατάστασης ανανεώνεται ατελείωτα σε ένα ατελείωτο τρέξιμο. Ενημερώσεις και ανανεώσεις τιμών της μορφής :

$$Q(s,a) := Q(s,a) + \alpha(r + \gamma \max_{a'} Q(s',a') - Q(s,a)) \tag{2.6}$$

μπορούν επίσης να χρησιμοποιηθούν εφόσον κάθε ζευγάρι *α* και *s* ενημερώνεται απείρως συχνά, το *s'* λαμβάνεται από τη συνάρτηση *T(s,a,s')*, το *r* λαμβάνεται από τη συνάρτηση *R(s,a)* και ο ρυθμός μάθησης *α* μειώνεται αργά και σταδιακά.

2.5.2 Επανάληψη πολιτικής (Policy Iteration)

Ο αλγόριθμος αυτός χειρίζεται απευθείας την πολιτική, αντί να ψάχνει να βρει τη βέλτιστη, έμμεσα από τη βέλτιστη τιμή της συνάρτησης. Λειτουργεί ως ακολούθως :

Διάλεξε μία πολιτική π′

Επανέλαβε

π:=π′

υπολόγισε τη συνάρτηση τιμής της πολιτικής π

λύσε τις γραμμικές εξισώσεις

$$V_\pi(s) = R(s,\pi(s)) + \gamma \sum_{s' \in S} T(s,\pi(s),s') V_\pi(s')$$

βελτίωσε την πολιτική σε κάθε κατάσταση

$$\pi'(s) := \arg\max_a (R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') V_\pi(s'))$$

μέχρι π=π′

ΣΧΗΜΑ 4 : ΕΠΑΝΑΛΗΨΗ ΠΟΛΙΤΙΚΗΣ (POLICY ITERATION)

Η συνάρτηση τιμής μιας πολιτικής είναι η αναμενόμενη άπειρη, με την έκπτωση, ανταμοιβή η οποία μπορεί να αποκτηθεί σε κάθε κατάσταση, επιλέγοντας αυτή την πολιτική. Μπορεί να καθοριστεί λύνοντας ένα σύνολο γραμμικών εξισώσεων. Όταν γνωρίζουμε την τιμή κάθε κατάστασης υπό την παρούσα πολιτική, μπορούμε να αποφανθούμε για το αν η τιμή αυτή μπορεί να βελτιωθεί αλλάζοντας την πρώτη

δράση που επιλέχτηκε. Αν αυτό είναι δυνατό, τότε αλλάζουμε την πολιτική ώστε να διαλέξουμε τη νέα δράση, οποτεδήποτε και αν είναι σε αυτή την κατάσταση. Όταν δεν είναι δυνατές περαιτέρω βελτιώσεις, τότε η πολιτική είναι σίγουρο πως είναι βέλτιστη.

Από τη στιγμή που οι πιθανές διαφορετικές πολιτικές είναι $|A|^{|S|}$ και η ακολουθία των πολιτικών βελτιώνεται σε κάθε βήμα, ο αλγόριθμος τερματίζει το πολύ σε εκθετικό αριθμό επαναλήψεων (Puterman, 1994). Ωστόσο, ένα ανοιχτό ερώτημα είναι πόσες επαναλήψεις χρειάζεται ο αλγόριθμος στη χειρότερη περίπτωση.

2.5.3 Βελτιώσεις στην επανάληψη τιμής και επανάληψη πολιτικής

Στην πράξη, η επανάληψη τιμής είναι πολύ γρηγορότερη από την επανάληψη πολιτικής αλλά η τελευταία συνήθως χρειάζεται λιγότερες επαναλήψεις. Σχετικά με το ποια τακτική είναι καλύτερη για μεγάλης κλίμακας προβλήματα, έχουν υπάρξει πολλά επιχειρήματα. Ο τροποποιημένος αλγόριθμος επανάληψης πολιτικής του Puterman δίνει μία μέθοδο για την αλλαγή του χρόνου επαναλήψεων με τη βελτίωση των επαναλήψεων με ένα πιο ομαλό τρόπο. Η βασική ιδέα είναι ότι το «ακριβό» κομμάτι της επανάληψης πολιτικής είναι να λύσεις ως προς την ακριβή τιμή του V_π. Έτσι, αντί να βρεθεί μία ακριβή τιμή για το V_π, μπορούν να επαναληφθούν μερικά βήματα μιας τροποποιημένης επανάληψης τιμής όπου η πολιτική κρατείται συγκεκριμένη για επιτυχείς επαναλήψεις. Αυτό μπορεί να αποδειχθεί πως παράγει μία προσέγγιση της τιμής V_π που συγκλίνει γραμμικά στο γ. Στην πράξη, αυτό οδηγεί σε ουσιαστική επιτάχυνση του αλγορίθμου.

Διάφορες μέθοδοι αριθμητικής ανάλυσης, οι οποίες επιταχύνουν τη σύγκλιση δυναμικού προγραμματισμού μπορούν να χρησιμοποιηθούν για την επιτάχυνση της επανάληψης τιμής και πολιτικής (πολυπλεγματικές μέθοδοι, συγχώνευση καταστάσεων).

2.6 Μαθαίνοντας μια βέλτιστη πολιτική

Στα προηγούμενα αναφέρθηκαν μέθοδοι για την εύρεση μίας βέλτιστης πολιτικής για μία αλυσίδα markov, υποθέτοντας πως έχουμε έτοιμο μοντέλο. Το μοντέλο αυτό στην

ουσία περιλαμβάνει τη συνάρτηση πιθανότητας μετάβασης *T(s,α,s')* και τη συνάρτηση ενίσχυσης *R(s,α)*. Η ενισχυτική μάθηση πρωτίστως ασχολείται με το πως θα βρεθεί μία βέλτιστη πολιτική, χωρίς να υπάρχει γνώση του μοντέλου αυτού. Ο πράκτορας πρέπει να αλληλεπιδράσει άμεσα με το περιβάλλον ώστε να αποκτήσει πληροφορίες, οι οποίες με τη βοήθεια κατάλληλου αλγορίθμου, μπορούν να χρησιμοποιηθούν για την εύρεσης μιας βέλτιστης πολιτικής.

Σε αυτό το σημείο υπάρχουν δύο μέθοδοι για την επίτευξη του στόχου :

- Χωρίς-μοντέλο : ο πράκτορας μαθαίνει έναν ελεγκτή χωρίς να μάθει κάποιο μοντέλο
- Βασισμένη σε μοντέλο : ο πράκτορας μαθαίνει ένα μοντέλο και το χρησιμοποιεί για να φτιάξει έναν ελεγκτή.

Ποια προσέγγιση είναι η καλύτερη; Το ερώτημα αυτό έχει δημιουργήσει έντονη διαμάχη στην κοινότητα της ενισχυτικής μάθησης. Ένας σημαντικός αριθμός αλγορίθμων έχει προταθεί και από τις δύο πλευρές. Η ερώτηση αυτή εμφανίζεται επίσης και σε άλλους τομείς όπως ο προσαρμοζόμενος έλεγχος, όπου υπάρχει η διχογνωμία μεταξύ άμεσου και έμμεσου ελέγχου.

Οι αλγόριθμοι της πρώτης κατηγορίας συνήθως εγγυώνται μικρό χρόνο υπολογισμού για κάθε εμπειρία αλλά κάνουν συνήθως αναποτελεσματική χρήση των δεδομένων που συλλέγουν και χρειάζονται αρκετή εμπειρία για να επιτύχουν καλή επίδοση. Αντίθετα, οι αλγόριθμοι της δεύτερης κατηγορίας είναι ιδιαίτερα σημαντικοί σε εφαρμογές που ο υπολογισμός θεωρείται φτηνός και η εμπειρία του πραγματικού κόσμου ακριβή.

2.6.1 Μέθοδοι βασισμένες σε κάποιο μοντέλο

Παραδείγματα αλγορίθμων της κατηγορίας αυτής είναι:

Μέθοδος ισοδυναμίας βεβαιότητας : Πρώτα μαθαίνεις τις συναρτήσεις *T* και *R* εξερευνώντας το περιβάλλον και κρατώντας στατιστικά για τα αποτελέσματα κάθε δράσης. Έπειτα υπολογίζεις μία βέλτιστη πολιτική σύμφωνα με κάποια από τις μεθόδους που αναφέρθηκαν στην παράγραφο 2.5. Οι ενστάσεις που έχουν τεθεί σε αυτή τη μέθοδο είναι αρκετές :

- Κάνει μία αυθαίρετη διάκριση μεταξύ της φάσης μάθησης και της φάσης δράσης
- Πως θα συγκεντρώσει δεδομένα για το περιβάλλον αρχικά; Η τυχαία εξερεύνηση μπορεί να είναι επικίνδυνη και σε μερικά περιβάλλοντα είναι μία υπερβολικά αναξιόπιστη μέθοδος συλλογής δεδομένων
- Η πιθανότητα αλλαγών στο περιβάλλον είναι επίσης προβληματική. Σπάζοντας τη ζωή του πράκτορα σε φάση αμιγούς μάθησης και φάση αμιγούς δράσης εμφανίζεται ο κίνδυνος ο βέλτιστος ελεγκτής να είναι μη-βέλτιστος εφόσον το περιβάλλον αλλάξει.

Μια παραλλαγή της μεθόδου αυτής περιλαμβάνει τη συνεχή μάθηση του μοντέλου σε όλη τη διάρκεια ζωής του πράκτορα και σε κάθε βήμα, το παρόν μοντέλο χρησιμοποιείται για να υπολογίζεται η βέλτιστη πολιτική και η συνάρτηση τιμής. Αυτή η μέθοδος κάνει αρκετά αποτελεσματική χρήση των δεδομένων αλλά εξακολουθεί να αγνοεί την ερώτηση της εξερεύνησης και είναι ιδιαίτερα απαιτητική υπολογιστικά, ακόμα και για μικρούς χώρους καταστάσεων.

<u>*Μέθοδος Dyna :*</u> Η αρχιτεκτονική του Dyna χρησιμοποιεί στρατηγικές που είναι και πιο αποτελεσματικές τόσο από τη μάθηση χωρίς μοντέλο αλλά και πιο υπολογιστικά αποδοτικές από τον προηγούμενο αλγόριθμο. Χρησιμοποιεί την εμπειρία τόσο για να φτιάξει το μοντέλο (συναρτήσεις T και R) όσο και για να προσαρμόσει την πολιτική ενώ χρησιμοποιεί το μοντέλο για να προσαρμόζει την πολιτική.

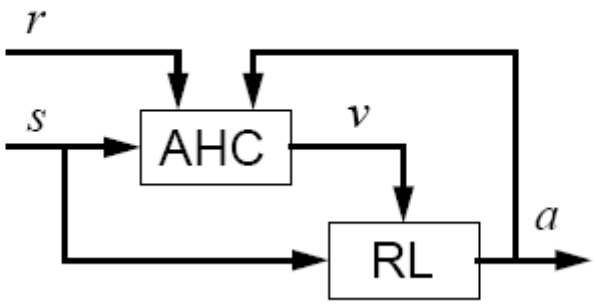
Πέραν των παραπάνω μεθόδων έχουν προταθεί και άλλες, βελτιωμένες, όπως η ουρά-Dyna, η εκκαθάριση με προτεραιότητα κλπ

2.6.2 Μέθοδοι που είναι ανεξάρτητες από κάποιο μοντέλο

Το μεγαλύτερο πρόβλημα που αντιμετωπίζεται στην ενισχυτική μάθηση είναι η ανάθεση προσωρινής ανταμοιβής. Πως μπορούμε να γνωρίζουμε αν μία δράση που επιλέχτηκε είναι καλή αν έχει μακροπρόθεσμα αποτελέσματα; Μία στρατηγική είναι να περιμένουμε ως το τέλος και να ανταμείψουμε τις δράσεις ανάλογα με το αν το αποτέλεσμα ήταν καλό ή κακό. Σε διάφορες όμως διεργασίες είναι δύσκολο να γνωρίζουμε πότε έρχεται το τέλος και αυτό μπορεί να χρειαστεί μεγάλο ποσοστό μνήμης. Έτσι χρησιμοποιούμε τη γνώση από την επανάληψη τιμής για να προσαρμόσουμε την αναμενόμενη τιμή μιας κατάστασης βασισμένη στην άμεση

ανταμοιβή και στην αναμενόμενη τιμή της επόμενης κατάστασης. Αυτή η τάξη αλγορίθμων είναι γνωστοί σαν μέθοδοι χρονικής διαφοράς (temporal difference methods/TD) [6].

Προσαρμοστική Ευρετική Κριτική (Adaptive Heuristic Critic/AHC) και TD(λ) : Ο αλγόριθμος προσαρμοστικής ευρετικής κριτικής είναι μία προσαρμοσμένη έκδοση της επανάληψης πολιτικής στην οποία ο υπολογισμός της συνάρτησης τιμής δεν υλοποιείται από ένα σύνολο γραμμικών εξισώσεων αλλά υπολογίζεται από έναν αλγόριθμο που καλείται *TD(0)*. Ένα μπλοκ διάγραμμα του αλγορίθμου αυτού φαίνεται στο ακόλουθο σχήμα :



ΣΧΗΜΑ 5 : ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΑΛΓΟΡΙΘΜΟΥ ΠΡΟΣΑΡΜΟΣΤΙΚΗΣ ΕΥΡΕΤΙΚΗΣ ΚΡΙΤΙΚΗΣ

Στην ουσία έχουμε δύο συνιστώσες : η μία είναι ο κριτής (συμβολίζεται με *AHC*) και η άλλη η συνιστώσα της ενισχυτικής μάθησης (συμβολίζεται με *RL*). Η συνιστώσα *RL* θα δρα όχι για να μεγιστοποιήσει τη στιγμιαία ανταμοιβή αλλά για να μεγιστοποιήσει την ευρετική τιμή *v*, η οποία υπολογίζεται από τον κριτή. Ο κριτής χρησιμοποιεί το πραγματικό εξωτερικό σήμα της ενίσχυσης για να μάθει να αντιστοιχίζει τις καταστάσεις με τις αναμενόμενες –με την έκπτωση- τιμές, με δεδομένο ότι η πολιτική που ακολουθείται είναι αυτή που υπάρχει τη δεδομένη στιγμή στη συνιστώσα *RL*.

Η αναλογία με την προσαρμοσμένη επανάληψη πολιτικής φαίνεται αν φανταστούμε τις δύο συνιστώσες να λειτουργούν εναλλακτικά. Η πολιτική π, που υλοποιείται από την *RL*, τροποποιείται και ο κριτής μαθαίνει τη συνάρτηση τιμής V_π για αυτή την πολιτική. Τώρα τροποποιείται ο κριτής και η συνιστώσα *RL* μαθαίνει μία νέα πολιτική π' η οποία μεγιστοποιεί τη νέα συνάρτηση τιμής κ.ο.κ. Στις περισσότερες υλοποιήσεις παρόλα αυτά, οι δύο συνιστώσες λειτουργούν παράλληλα. Μόνο όμως η

εναλλακτική υλοποίηση μπορεί να εγγυηθεί τη σύγκλιση στη βέλτιστη πολιτική, κάτω από τις κατάλληλες συνθήκες.

Το μόνο που μένει είναι να εξηγηθεί πως ο κριτής μπορεί να μάθει την τιμή μιας πολιτικής. Ορίζουμε σαν $\langle s, a, r, s' \rangle$ μία πλειάδα εμπειρίας η οποία περικλείει μία απλή μετάβαση στο περιβάλλον. Εδώ, *s* είναι η κατάσταση του πράκτορα πριν τη μετάβαση, *α* είναι η επιλογή δράσης, *r* η άμεση ανταμοιβή και *s'* η νέα κατάσταση. Η τιμής μιας πολιτικής μαθαίνεται με χρήση του αλγόριθμου *TD(0)* του Sutton, που χρησιμοποιεί τον εξής κανόνα ενημέρωσης :

$$V(s) := V(s) + \alpha(r + \gamma V(s') - V(s)) \tag{2.7}$$

Όποτε ο πράκτορας βρεθεί σε μία κατάσταση *s*, η αναμενόμενη της τιμή ενημερώνεται ώστε να είναι πιο κοντά στο *r+γV(s')* αφού το *r* είναι η άμεση ανταμοιβή και το *V(s')* η αναμενόμενη τιμή της επόμενης κατάστασης. Αυτό είναι παρόμοιο με τον απλό κανόνα της επανάληψης τιμής με τη μόνη διαφορά ότι εδώ η γνώση προέρχεται από τον πραγματικό κόσμο και όχι από κάποιο γνωστό μοντέλο. Η ιδέα-κλειδί είναι ότι το *r+γV(s')* είναι ένα δείγμα της τιμής της *V(s)* και ότι είναι αρκετά πιθανό να είναι σωστή τιμή επειδή ενσωματώνει την πραγματική ενίσχυση r. Αν ο ρυθμός μάθησης α προσαρμόζεται κατάλληλα (πρέπει να μειώνεται αργά) και η πολιτική τροποποιείται κατάλληλα, ο αλγόριθμος *TD(0)* εγγυάται σύγκλιση στη βέλτιστη συνάρτηση τιμής.

Ο αλγόριθμος *TD(0)*, στην πραγματικότητα είναι ένα στιγμιότυπο μιας πιο γενικής τάξης αλγορίθμων που καλούνται *TD(λ)* (στην προκείμενη περίπτωση λ=0). Ο *TD(0)* κοιτάζει μόνο ένα βήμα μπροστά όταν προσαρμόζει τις αναμενόμενες τιμές. Αν και λογικά θα καταλήξει στη σωστή απάντηση, εντούτοις μπορεί να χρειαστεί αρκετό χρόνο. Ο γενικότερος *TD(λ)* κανόνας είναι παρόμοιος με τον *TD(0)* :

$$V(u) := V(u) + \alpha(r + \gamma V(s') - V(s))e(u) \tag{2.8}$$

αλλά εφαρμόζεται σε κάθε κατάσταση ανάλογα με την καταλληλότητα της *e(u)*. Ένας ορισμός της καταλληλότητας e θα μπορούσε να είναι :

$$e(s) = \sum_{k=1}^{t} (\lambda\gamma)^{t-k} \delta_{s,s_k}, \quad \text{όπου} \quad \delta_{s,s_k} = \begin{cases} 1, s = s_k \\ 0, \text{αλλού} \end{cases} \tag{2.9}$$

Η καταλληλότητα της κατάστασης *s* είναι ο βαθμός στον οποίο έχει βρεθεί ο πράκτορας στο πρόσφατο παρελθόν. Όταν μία ενίσχυση ληφθεί, τότε χρησιμοποιείται για να ενημερωθούν όλες οι καταστάσεις που έχει περάσει ο πράκτορας πρόσφατα,

σύμφωνα με την καταλληλότητά τους. Όταν $\lambda=0$, τότε έχουμε τον αλγόριθμο *TD(0)*. Όταν $\lambda=1$, τότε ενημερώνουμε όλες τις καταστάσεις ανάλογα με τον αριθμό των φορών που έχει βρεθεί εκεί ο πράκτορας στο τέλος ενός τρεξίματος. Η ενημέρωση της καταλληλότητας μπορεί να γίνει επί της γραμμής (online) ως ακολούθως :

$$e(s) := \begin{cases} \gamma\lambda e(s)+1, s = \pi\alpha\rho ούσα \\ \gamma\lambda e(s), \alpha\lambda\lambda ιώς \end{cases} \tag{2.10}$$

Είναι υπολογιστικά πιο ακριβό να εκτελεστεί ο *TD(λ)*, αν και συχνά συγκλίνει πολύ γρήγορα για μεγάλα λ.

Οι δουλειές των δύο συνιστωσών του αλγορίθμου μπορούν να εκτελεστούν με ενοποιημένο τρόπο από τον αλγόριθμο της μάθησης *Q* του Watkins [8], ο οποίος αναλύεται στα επόμενα

2.7 Εισαγωγή στη μάθηση Q

Η μάθηση Q είναι ένας αλγόριθμος ενισχυτικής μάθησης που μαθαίνει τις τιμές μιας συνάρτησης *Q(s,α)* ώστε να βρει μία βέλτιστη πολιτική π. Οι τιμές της συνάρτησης *Q(s,α)* δείχνουν πόσο καλό είναι να επιλεγεί μία συγκεκριμένη δράση σε μία συγκεκριμένη κατάσταση. Η συνάρτηση *Q(s,α)* ορίζεται ως η ανταμοιβή που δίνεται άμεσα εκτελώντας τη δράση α από την κατάσταση s, επαυξημένη κατά την τιμή (με έκπτωση) των ανταμοιβών που θα ληφθούν στο μέλλον αν ακολουθηθεί μια βέλτιστη πολιτική.

$$Q(s,\alpha) = E\left[r(s,\alpha) + \gamma V^*(\delta(s,\alpha)) \right] \tag{2.11}$$

Αν η συνάρτηση Q είναι γνωστή τότε μία βέλτιστη πολιτική π δίνεται από τον τύπο :

$$\pi^*(s) = \arg\max_{\alpha \in A} Q(s,\alpha) \tag{2.12}$$

Η εξίσωση αυτή δείχνει πως αν ένας πράκτορας γνωρίζει τη συνάρτηση Q, δε χρειάζεται να γνωρίζει τη συνάρτηση ανταμοιβής *r(s,α)* και τη συνάρτηση μετάβασης *δ(s,α)* για να ορίσει μία βέλτιστη πολιτική π^*, σε αντίθεση με την επανάληψη πολιτικής και την επανάληψη τιμής.

Ας σημειωθεί πως η *V*(s)* και η Q(s,α) σχετίζονται ως εξής :

$$V^*(s) = \max_{a \in A} Q(s,a) \tag{2.13}$$

Ένας αναδρομικός ορισμός της εξίσωσης Q μπορεί να δοθεί αν αφαιρέσουμε τις εξισώσεις (2.11) και (2.13).

$$Q(s,\alpha) = E\left[r(s,\alpha) + \gamma \max_{\alpha' \in A} Q(\delta(s,a),\alpha') \right] \tag{2.14}$$

2.8 Αλγόριθμος μάθησης Q

Ο αλγόριθμος μάθησης Q βασίζεται στον ορισμό της συνάρτησης Q. Ένας πράκτορας, επαναληπτικά, υπολογίζει τις τιμές της συνάρτησης Q. Σε κάθε επανάληψη του αλγορίθμου ο πράκτορας παρατηρεί την παρούσα κατάσταση *s*, επιλέγει μία δράση *α*, εκτελεί αυτή την πράξη *α* και έπειτα παρατηρεί την ανταμοιβή *r=r(s,α)* και τη νέα κατάσταση *s'=δ(s,α)*. Στη συνέχεια ενημερώνει την αναμενόμενη τιμή της συνάρτησης *Q*, που συμβολίζεται με $\hat{Q}$, σύμφωνα με τον εξής κανόνα εκπαίδευσης :

$$\hat{Q}(s,a) \leftarrow (1-\alpha)\hat{Q}(s,a) + \alpha\left(r + \gamma \max_{a' \in A} \hat{Q}(s,a') \right) \tag{2.15}$$

όπου $\alpha \in [0,1)$ είναι η παράμετρος του ρυθμού μάθησης. Ο αλγόριθμος φαίνεται στο σχήμα 6.

```
Για όλες τις καταστάσεις s στο S και για όλες τις δράσεις α
στο A αρχικοποίησε την Q(s,α) σε αυθαίρετη τιμή.
ΕΠΑΝΕΛΑΒΕ (για κάθε προσπάθεια)
      Αρχικοποίησε την παρούσα κατάσταση s
      ΕΠΑΝΕΛΑΒΕ (για κάθε βήμα της προσπάθειας)
            Παρατήρησε την παρούσα κατάσταση s
            Διάλεξε μία δράση α ακολουθώντας μια πολιτική π
            Εκτέλεσε τη δράση α
            Λάβε μια άμεση ανταμοιβή α
            Παρατήρησε τη νέα κατάσταση s'
            Ανανέωσε την Q(s,α) σύμφωνα με την εξίσωση (2.15)
            s←s'
      ΜΕΧΡΙ η s να είναι τελική κατάσταση
```

ΣΧΗΜΑ 6 : ΑΛΓΟΡΙΘΜΟΣ ΜΑΘΗΣΗΣ Q

Οι Watkins και Dayan απέδειξαν πως οι αναμενόμενες τιμές Q του πράκτορα θα συγκλίνουν στις πραγματικές τιμές, με πιθανότητα 1, με τις εξής προϋποθέσεις

- το περιβάλλον είναι μία σταθερή αλυσίδα Markov με ορισμένες ανταμοιβές *r(s,α)*
- οι αναμενόμενες τιμές της συνάρτησης Q αποθηκεύονται σε έναν πίνακα αναζήτησης και αρχικοποιούνται σε πεπερασμένες αυθαίρετες τιμές
- κάθε δράση εκτελείται σε μία κατάσταση άπειρες φορές
- $\gamma \in [0,1), \alpha \in [0,1)$ και το α μειώνεται σταδιακά ως το 0 καθώς περνάει ο χρόνος

Όταν οι τιμές της Q συγκλίνουν στις βέλτιστες τιμές τους, είναι θεμιτό για τον πράκτορα να δράσει άπληστα και να επιλέξει σε κάθε κατάσταση τη δράση με τη μεγαλύτερη τιμή Q. Κατά τη διάρκεια της μάθησης, υπάρχει ένα μεγάλο δίλημμα μεταξύ της εκμετάλλευσης και της εξερεύνησης, στο οποίο πρέπει να δοθεί λύση. Στη γενική περίπτωση, δεν υπάρχουν προσεγγίσεις, τυπικά επιβεβαιωμένες που να υποστηρίζουν τη μία ή την άλλη λύση.

Η μάθηση Q είναι ανεπηρέαστη από την εξερεύνηση : Οι τιμές της Q θα συγκλίνουν στις βέλτιστες τιμές, ανεξάρτητα από το πως ο πράκτορας συμπεριφέρεται όταν τα

δεδομένα συλλέγονται (όσο βέβαια όλα τα ζεύγη καταστάσεων-δράσεων δοκιμαστούν αρκετά συχνά). Αυτό σημαίνει πως αν και το θέμα εκμετάλλευσης-εξερεύνησης πρέπει να διευθετηθεί στη μάθηση Q, οι λεπτομέρειες της στρατηγικές εξερεύνησης δε θα επηρεάσουν τη σύγκλιση του αλγορίθμου μάθησης. Για αυτούς τους λόγους, η μάθηση Q είναι η πιο διάσημη και φαίνεται πως είναι ο πιο αποτελεσματικός αλγόριθμος για μάθηση με καθυστερημένη ενίσχυση για συστήματα χωρίς χρήση μοντέλου (άμεσα). Το μειονέκτημά της είναι πως δεν διευθετεί θέματα που αφορούν τη γενίκευση πάνω σε μεγάλο χώρο καταστάσεων ή/και δράσεων ενώ μπορεί να συγκλίνει αργά προς μία βέλτιστη πολιτική.

2.9 Υλοποίηση μοντέλου μάθησης Q

2.9.1 Υπολογισμός συνάρτησης Q

Οι εκτιμώμενες τιμές της συνάρτησης Q πρέπει να αποθηκευτούν κάπου κατά τη διάρκεια του υπολογισμού και μετά. Ο απλούστερος τρόπος είναι ένας πίνακας αναζήτησης με μία ξεχωριστή καταχώρηση για κάθε ζεύγος κατάστασης-δράσης. Το πρόβλημα της μεθόδου αυτής είναι η πολυπλοκότητα του χώρου. Προβλήματα με μεγάλο χώρο καταστάσεων ή/και δράσεων οδηγούν σε αργή μάθηση και μεγάλους πίνακες με τιμές Q, τα οποία στη χειρότερη περίπτωση δε μπορούν να αποθηκευτούν στη μνήμη του υπολογιστή.

Για την αντιμετώπιση του προβλήματος των μεγάλων χώρων καταστάσεων/δράσεων, χρησιμοποιείται μία μέθοδος προσέγγισης της συνάρτησης (όπως ένα νευρωνικό δίκτυο ή ένα δέντρο απόφασης) για την «αποθήκευση» των τιμών Q.

Κατά τη διάρκεια της μάθησης Q, το νευρωνικό δίκτυο μαθαίνει μια αντιστοίχηση από περιγραφές καταστάσεων σε τιμές Q. Αυτό γίνεται με τον υπολογισμό μιας τιμής Q-στόχου σύμφωνα με την εξίσωση (2.15) και χρησιμοποιώντας τον αλγόριθμο της ανάστροφης διάδοσης (backpropagation) για την ελαχιστοποίηση της ασυμφωνίας μεταξύ της τιμής Q-στόχου και της αναμενόμενης τιμής Q που υπολογίζεται από το νευρωνικό δίκτυο. Ολόκληρος ο αλγόριθμος φαίνεται στο σχήμα 7.

```
Αρχικοποίησε όλα τα βάρη του νευρωνικού δικτύου (ΝΔ) σε μικρούς τυχαίους αριθμούς
ΕΠΑΝΕΛΑΒΕ (για κάθε προσπάθεια)
    Αρχικοποίησε την παρούσα κατάσταση s
        ΕΠΑΝΕΛΑΒΕ (για κάθε βήμα κάθε προσπάθειας)
            Παρατήρησε την παρούσα κατάσταση s
            Για όλες τις δράσεις α' στην s χρησιμοποίησε
                το ΝΔ για τον υπολογισμό της Q(s,α)
            Διάλεξε μία δράση α ακολουθώντας μία πολιτική π
            Q^output←Q(s,α)
            Εκτέλεσε τη δράση α
            Λάβε μία άμεση ανταμοιβή r
            Παρατήρησε τη νέα κατάσταση s'
            Για όλες τις δράσεις α' στην s' χρησιμοποίησε
                το ΝΔ για τον υπολογισμό της Q(s',α')
            Σύμφωνα με την εξίσωση (2.15) υπολόγισε
                το Q^target←Q(s,α)
            Προσάρμοσε το ΝΔ με ανάστροφη διάδοση του
                σφάλματος (Q^target-Q^output)
            s←s'
        ΜΕΧΡΙ s να είναι τελική κατάσταση
```

ΣΧΗΜΑ 7 : ΑΛΓΟΡΙΘΜΟΣ ΜΑΘΗΣΗΣ Q ΜΕ ΧΡΗΣΗ ΝΕΥΡΩΝΙΚΟΥ ΔΙΚΤΥΟΥ

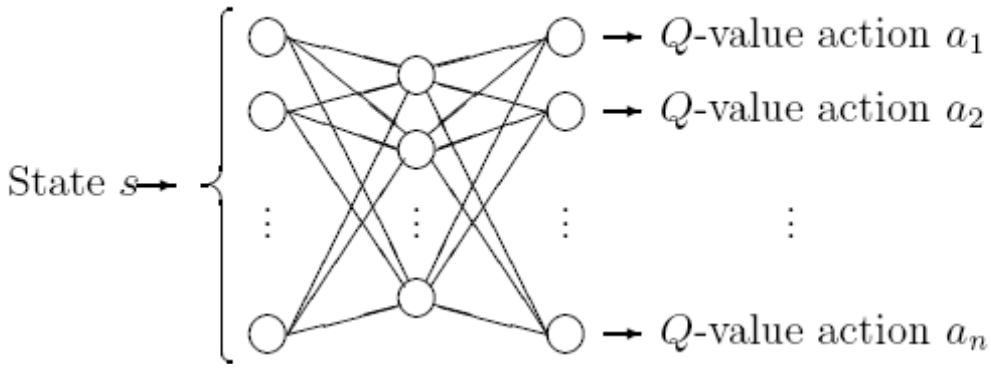
Είναι σημαντικό να σημειωθεί πως σε προβλήματα με μεγάλο χώρο καταστάσεων/δράσεων, το νευρωνικό δίκτυο εκπαιδεύεται βάσει των «επισκέψεων» σε ένα μικρό μόνο μέρος του χώρου καταστάσεων/δράσεων. Η χρήση του νευρωνικού δικτύου κάνει πιθανή τη γενίκευση σε καταστάσεις και δράσεις. Το νευρωνικό δίκτυο, βασισμένο στην εμπειρία από προηγούμενα ζεύγη καταστάσεων-δράσεων είναι ικανό να δώσει μία αναμενόμενη τιμή Q για ένα αυθαίρετο ζεύγος κατάστασης-δράσης.

Πως μπορεί όμως να αντιμετωπίσει ένα νευρωνικό δίκτυο αυτή την περίπτωση; Είναι πιθανό πως τα εσωτερικά στρώματα του νευρωνικού δικτύου μαθαίνουν να εξάγουν

γνωρίσματα που είναι χρήσιμα στον υπολογισμό των τιμών Q για ζεύγη καταστάσεων-δράσεων. Για να διευκολυνθεί η δουλειά του νευρωνικού δικτύου κάποιος μπορεί να προσφέρει πρόσθετα γνωρίσματα στην πλευρά εισόδου εκτός από την περιγραφή της κατάστασης, το οποίο θα οδηγήσει πιθανότατα στη μάθηση μιας καλύτερης πολιτικής.

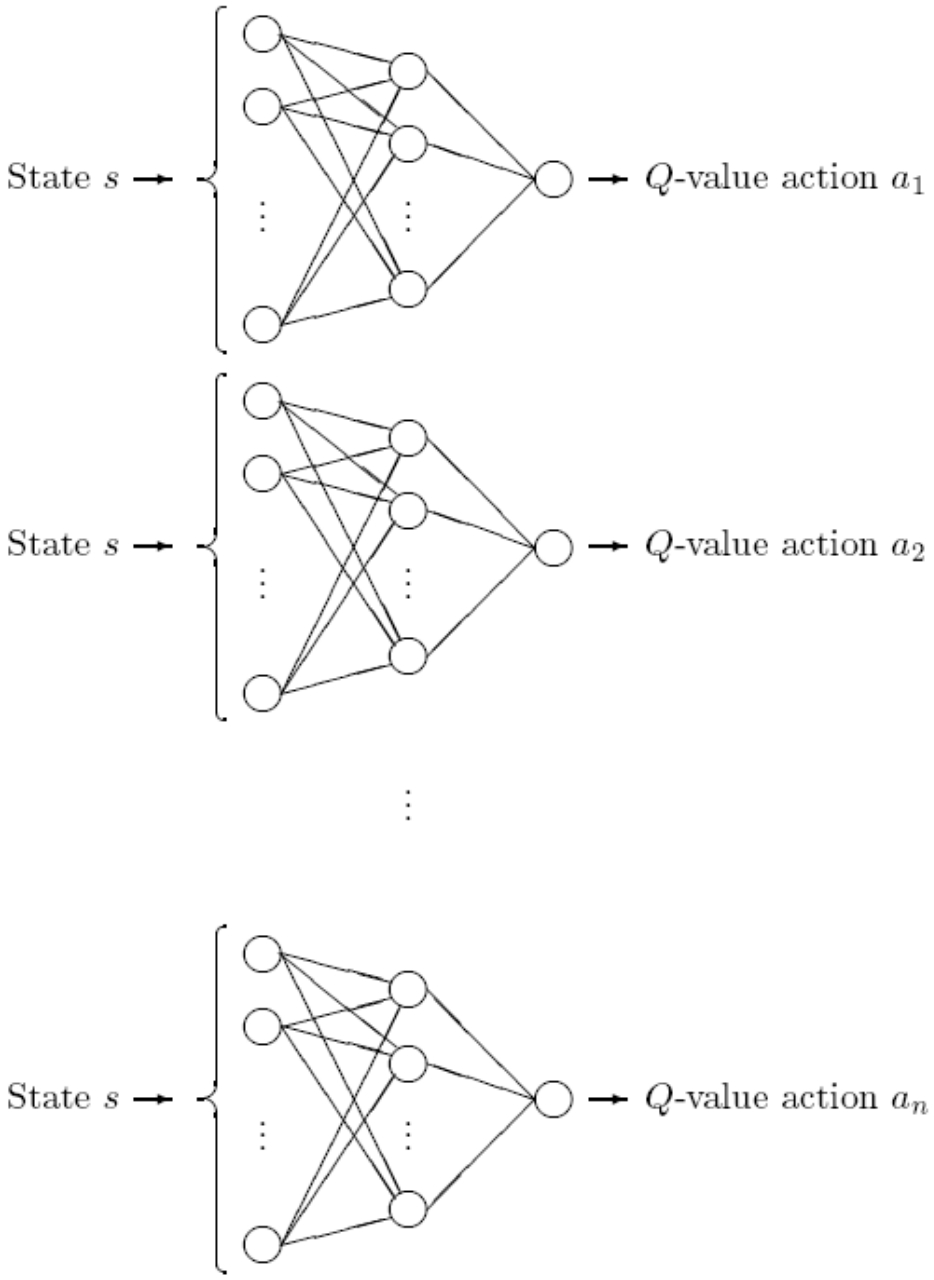
Όταν η συνάρτηση Q εξάγεται από ένα πολυστρωματικό νευρωνικό δίκτυο, είναι πιθανό να χρησιμοποιηθεί ένα ξεχωριστό δίκτυο για κάθε δράση ή ένα απλό δίκτυο με μία ξεχωριστή έξοδο για κάθε δράση ή ένα απλό δίκτυο με εισόδους και την κατάσταση και τη δράση και την τιμή Q σαν έξοδο, με συνηθέστερες τις δύο πρώτες προσεγγίσεις.

Η είσοδος ενός απλού δικτύου με μία ξεχωριστή έξοδο για κάθε μία από τις δράσεις αποτελείται από μία ή περισσότερες μονάδες που αναπαριστούν την κατάσταση. Η έξοδος του δικτύου αποτελείται από τόσες μονάδες όσες είναι οι δράσεις που μπορούν να επιλεγούν. Το σχήμα 8 αναπαριστά ένα τέτοιο δίκτυο. Όταν ένα απλό δίκτυο χρησιμοποιείται, η γενίκευση πάνω και από καταστάσεις και δράσεις είναι δυνατή.



ΣΧΗΜΑ 8 : ΈΝΑ ΑΠΛΟ ΝΕΥΡΩΝΙΚΟ ΔΙΚΤΥΟ ΜΕ ΞΕΧΩΡΙΣΤΗ ΕΞΟΔΟ ΓΙΑ ΚΑΘΕ ΔΡΑΣΗ

Αν υπάρχει ένα ξεχωριστό δίκτυο για κάθε δράση, η είσοδος κάθε δικτύου αποτελείται από μία ή περισσότερες μονάδες που αναπαριστούν μία κατάσταση. Κάθε δίκτυο έχει μόνο μία έξοδο, που θεωρείται η τιμή της Q που σχετίζεται με την κατάσταση που δίνεται σαν είσοδος στο δίκτυο και με τη δράση που αντιπροσωπεύεται από το δίκτυο. Το σχήμα 9 αναπαριστά πολλαπλά τέτοια δίκτυα που σχετίζονται με τις δράσεις $\alpha_1,\alpha_2,...,\alpha_n$. Σε αυτή την περίπτωση, είναι δυνατή γενίκευση μόνο πάνω από καταστάσεις.



ΣΧΗΜΑ 9 : ΞΕΧΩΡΙΣΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΓΙΑ ΚΑΘΕ ΔΡΑΣΗ

Η μάθηση Q με χρήση νευρωνικών δικτύων για την αποθήκευση των τιμών Q μπορεί να λύσει μεγαλύτερα προβλήματα από τη μάθηση Q με χρήση πίνακα αλλά δεν εγγυάται τη σύγκλιση. Το πρόβλημα που σχετίζεται με τη χρήση των νευρωνικών δικτύων, είναι πως αυτά τα δίκτυα πραγματοποιούν μη-τοπικές αλλαγές στη συνάρτηση Q, ενώ η μάθηση Q απαιτεί οι ανανεώσεις στη συνάρτηση Q να είναι τοπικές. Όταν ανανεώνεται η τιμή για ένα ζεύγος κατάστασης-δράσης, το δίκτυο μπορεί να σβήσει την τιμή κάποιων άλλων ζευγών. Αυτός είναι και ένας από τους

λόγους που η μέθοδος των νευρωνικών δικτύων δεν εγγυάται σύγκλιση στις πραγματικές τιμές Q.

2.9.2 Επιλογή δράσεων

Οι αλγόριθμοι που περιγράφηκαν στα σχήματα 6 και 7 δεν καθορίζουν πως επιλέγονται οι δράσεις από τον πράκτορα. Μία από τις προκλήσεις που προκύπτουν εδώ, είναι η ισορρόπηση ανάμεσα στην επιλογή της εκμετάλλευσης και αναζήτησης. Ένας πράκτορας εξαντλεί τη γνώση του για τη συνάρτηση Q όταν επιλέγει μία δράση με τη μεγαλύτερη αναμενόμενη τιμή Q. Αντίθετα, αν ο πράκτορας επιλέξει μία από τις άλλες δράσεις, τότε εξερευνά επειδή βελτιώνει την εκτίμηση της τιμής Q της δράσης.

Ο πράκτορας θέλει να εξαντλήσει τη γνώση που έχει με στόχο να πάρει την ανταμοιβή αλλά επίσης θέλει και να εξερευνήσει ώστε να κάνει καλύτερες επιλογές δράσεων στο μέλλον. Δεν είναι δυνατό να εξερευνήσει και να εξαντλήσει ταυτόχρονα γιατί θα υπάρξει σύγκρουση ανάμεσα στις δύο τακτικές.

Υπάρχουν πολλές μέθοδοι για την ισορρόπηση μεταξύ εκμετάλλευσης και εξερεύνησης. Μία από αυτές είναι η αποκαλούμενη "softmax" επιλογή δράσης, όπου οι πράκτορες επιλέγουν δράσεις πιθανοτικά βασιζόμενοι στις αναμενόμενες τιμές Q χρησιμοποιώντας την κατανομή του Boltzmann. Δοσμένης μιας κατάστασης s, ένας πράκτορας επιλέγει μία δράση α με πιθανότητα :

$$P(a) = \frac{\exp(\hat{Q}(s,a)/T)}{\sum_{a' \in A} \exp(\hat{Q}(s,a')/T} \tag{2.16}$$

όπου T είναι μία θετική παράμετρος που καλείται θερμοκρασία και ελέγχει το ποσοστό της εξερεύνησης. Μία πολύ χαμηλή θερμοκρασία τείνει σε μία άπληστη επιλογή δράσεων δηλαδή επιλέγεται η δράση με τη μεγαλύτερη αναμενόμενη τιμή Q. Μία πολύ υψηλή θερμοκρασία οδηγεί σε σχεδόν τυχαία επιλογή δράσεων. Η θερμοκρασία συνήθως πέφτει σταδιακά με το χρόνο, οπότε οδηγούμαστε σε σταδιακή μετάβαση από την εξερεύνηση στην εκμετάλλευση.

2.9.3 Περιβάλλοντα με πολλούς πράκτορες

Η μάθηση Q δεν εγγυάται σύγκλιση σε μη-στάσιμα περιβάλλοντα. Υπάρχουν πολλοί λόγοι για να χαρακτηριστεί ένα περιβάλλον ως μη-στάσιμο. Μία πιθανότητα είναι να υπάρχουν πολλοί πράκτορες ενεργοί στο ίδιο περιβάλλον. Κάθε πράκτορας λαμβάνει τις δράσεις του άλλου σαν μέρος του περιβάλλοντος, μαθαίνει και έτσι αλλάζει και προσαρμόζει την πολιτική του. Αυτό λαμβάνεται σαν ένα μη-στάσιμο περιβάλλον.

Η μάθηση Q αναπτύχθηκε για περιβάλλοντα στάσιμα. Λόγω μη-στασιμότητας, η σύγκλιση σε μία βέλτιστη πολιτική δε μπορεί να εγγυηθεί σε μη-στάσιμα περιβάλλοντα. Μια άλλη προσέγγιση σε ένα τέτοιο περιβάλλον θα ήταν να χρησιμοποιηθεί ένας αλγόριθμος ενισχυτικής μάθησης που έχει προσαρμοστεί ώστε να βρεθούν βέλτιστες πολιτικές σε περιβάλλοντα με πολλούς πράκτορες.

2.9.4 Μερικώς παρατηρήσιμες καταστάσεις

Ως τώρα έχουμε υποθέσει πως η κατάσταση που βρίσκεται το σύστημα ταυτίζεται με την πραγματική κατάσταση του περιβάλλοντος. Όταν τα αποτελέσματα της εισόδου προέρχονται από μία συσκευή αντίληψης του πράκτορα, τότε δεν υπάρχει λόγος να υποθέσουμε πως η συσκευή αυτή ταυτοποιείται μοναδικά με την κατάσταση του περιβάλλοντος. Εξαιτίας των αναπόφευκτων αντιληπτικών περιορισμών, αρκετές διαφορετικές καταστάσεις μπορούν να αντιστοιχιστούν στην ίδια είσοδο [10]. Το φαινόμενο αυτό αναφέρεται σαν *αντιληπτική ταύτιση* (perceptual aliasing). Λόγω αυτού του φαινομένου, δεν μπορούμε να εγγυηθούμε πως η μάθηση Q θα καταλήξει σε χρήσιμες πολιτικές δράσης, πόσο μάλλον σε βέλτιστες.

Διάφοροι μελετητές προσπάθησαν να αντιμετωπίσουν το πρόβλημα προσπαθώντας να μοντελοποιήσουν «κρυμμένες» καταστάσεις με τη χρήση εσωτερικής μνήμης. Αυτό σημαίνει, πως αν κάποιο κομμάτι του περιβάλλοντος μπορεί να ανιχνευθεί τώρα, ίσως είχε ανιχνευθεί και στο παρελθόν και μπορεί να μνημονευθεί από τον πράκτορα. Όταν υπάρχει αυτή η περίπτωση δεν υφίσταται πλέον πρόβλημα Markov, καθώς η επόμενη κατάσταση (προερχόμενη από οποιαδήποτε δράση) μπορεί να εξαρτάται από μια ακολουθία προηγούμενων παρά άμεσα από την προηγούμενη. Ο επανορισμός μιας δομής Markov ίσως είναι δυνατός σε αυτή την περίπτωση, αν πέραν από την αντίληψη υπάρχει και η δυνατότητα μνήμης.

2.9.5 Κλιμάκωση προβλημάτων

Διάφορες δυσκολίες εμπόδισαν μέχρι στιγμής την ευρεία εφαρμογή της ενισχυτικής μάθησης σε μεγάλα προβλήματα. Κάποια έχουν αντιμετωπιστεί. Παρακάτω αναφέρονται κάποια από αυτά μαζί με αναφορές για τις λύσεις που έχουν προταθεί.

i. εκμετάλλευση έναντι εξερεύνησης
   - Τυχαία χρήση δράσεων
   - Αγαπημένες καταστάσεις δεν επισκέπτονται συχνά
   - Διαχωρισμός φάσης μάθησης από τη φάση της χρήσης
   - Χρήση «δασκάλου» για την οδήγηση της εξερεύνησης

ii. Αργός χρόνος προς τη σύγκλιση
   - Συνδυασμός μάθησης με προηγούμενη γνώση : χρησιμοποίηση αναμενόμενων τιμών Q (αντί τυχαίων τιμών) στην αρχή
   - Χρήση ιεραρχίας δράσεων : μάθηση των πρωταρχικών δράσεων πρώτα και πάγωμα των χρήσιμων ακολουθιών σε μία μακρο-δράση και ύστερα εκμάθηση χρήσης αυτών των μακρο-δράσεων.
   - Χρήση «δασκάλου» : διαβαθμισμένα μαθήματα που ξεκινάνε από τις ανταμοιβές και χρήση παραδειγμάτων καλής συμπεριφοράς
   - Χρήση πιο αποτελεσματικών υπολογισμών όπως συχνή ανανέωση σε κάθε προσπάθεια

iii. Μεγάλος χώρος καταστάσεων [12]
   - Χρήση μη αυτόματης κωδικοποίησης (hand-coded) γνωρισμάτων
   - Χρήση νευρωνικών δικτύων
   - Χρήση μεθόδου του «πλησιέστερου γείτονα»

iv. Πρόσκαιρα προβλήματα έκπτωσης : Η χρήση μικρού συντελεστή μάθησης του πράκτορα (γ) μπορεί να τον κάνει άπληστο για πρόσκαιρες ανταμοιβές και αδιάφορο για το μέλλον αλλά η χρήση μεγάλου γ επιβραδύνει τη μάθηση
   - Χρήση μεθόδου μάθησης βασισμένη σε μέσες ανταμοιβές

v. Όχι «μεταφορά» της μάθησης. Το τι μαθαίνεται εξαρτάται από τη δόμηση της ανταμοιβής. Αν οι ανταμοιβές αλλάξουν, η μάθηση πρέπει να ξεκινήσει ξανά

- Χωρισμός της μάθησης σε δύο μέρη : μάθηση ενός «μοντέλου δράσεων» που προβλέπει πως οι δράσεις αλλάζουν καταστάσεις (σταθερό στα διάφορα προβλήματα) και ύστερα μάθηση των «τιμών» των καταστάσεων από την ενισχυτική μάθηση για κάθε διαφορετική περίπτωση ανταμοιβών. Μερικές φορές το κομμάτι της ενισχυτικής μάθησης μπορεί να αντικατασταθεί από έναν «προγραμματιστή» που χρησιμοποιεί το μοντέλο των δράσεων ώστε να φτιάχνει σχέδια για την επίτευξη των στόχων.

3 *Προσαρμογή παραμέτρων μάθησης ενός ταξινομητή*

3.1 Το πρόβλημα της ταξινόμησης / κατηγοριοποίησης

Το έργο της ταξινόμησης λαμβάνει χώρα σε ένα ευρύ πεδίο της ανθρώπινης δραστηριότητας. Στην ευρύτερη έννοια του ο όρος μπορεί να καλύψει κάθε δραστηριότητα όπου μία απόφαση ή πρόβλεψη λαμβάνεται στη βάση των διαθέσιμων ως τώρα πληροφοριών και μία διαδικασία ταξινόμησης είναι μία τυπική μέθοδος για να λαμβάνονται και άλλες κρίσεις για νέες καταστάσεις στο μέλλον [16].

Θέματα που πρέπει να εξεταστούν και να ληφθούν σοβαρά υπόψιν σε έναν ταξινομητή είναι :

- Ακρίβεια : Το σύστημα το οποίο ταξινομεί πρέπει να είναι αξιόπιστο, πράγμα το οποίο σημαίνει πως πρέπει να ελέγχεται με κάποιο τρόπο ο ρυθμός σφαλμάτων
- Ταχύτητα : Σε μερικές περιπτώσεις η ταχύτητα είναι ένα σημαντικό θέμα. Ένας ταξινομητής με 90% ακρίβεια μπορεί να είναι προτιμότερος από κάποιον που έχει 95% ακρίβεια αν είναι 100 φορές γρηγορότερος.

Όπως έχει ήδη αναφερθεί η διαδικασία εκπαίδευσης ενός νευρωνικού δικτύου περιλαμβάνει μόνο τις ελεύθερες παραμέτρους που ονομάζονται βάρη. Διαδικασία για την προσαρμογή και την εκμάθηση των παραμέτρων που ρυθμίζουν τη λειτουργία της εκπαίδευσης δεν υπάρχει και συνήθως προκαθορίζονται από την αρχή του συστήματος. Γίνεται λοιπόν κατανοητό πως σε ένα σύστημα όπως είναι ο

ταξινομητής, η λειτουργία της τροποποίησης και προσαρμογής των παραμέτρων της εκπαίδευσης (όπως ο ρυθμός μάθησης, ο αριθμός των εποχών κλπ) θα οδηγούσε σε σαφώς καλύτερα αποτελέσματα τόσο από άποψη ακρίβειας και ποιότητας όσο και από άποψη ταχύτητας.

3.2 Επηρεάζοντας το ρυθμό μάθησης

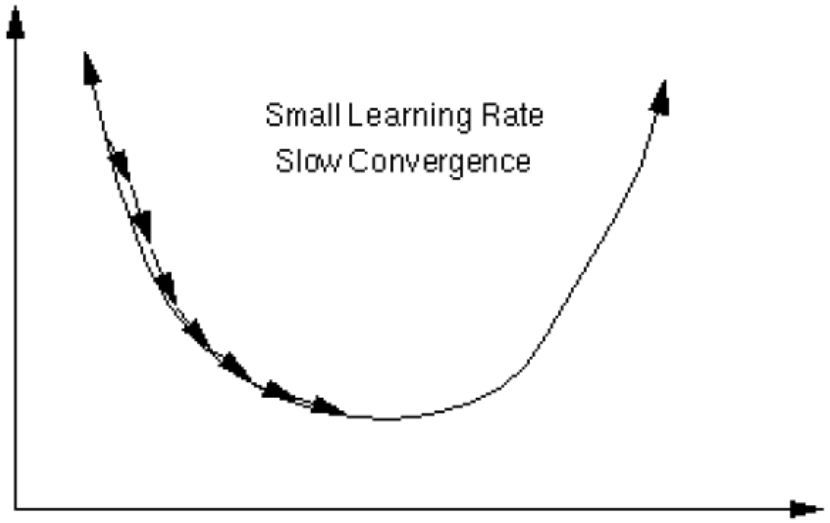
Ο κανόνας μάθησης της ανάστροφης διάδοσης είναι ένα ισχυρό εργαλείο γα την προσαρμογή των βαρών στα νευρωνικά δίκτυα [13]. Σε κάθε χρονική στιγμή t, κάθε βάρος ανανεώνεται σύμφωνα με τον κανόνα :

$$\Delta w(t) = -\eta \frac{\partial E}{\partial w}(t) \tag{3.1}$$

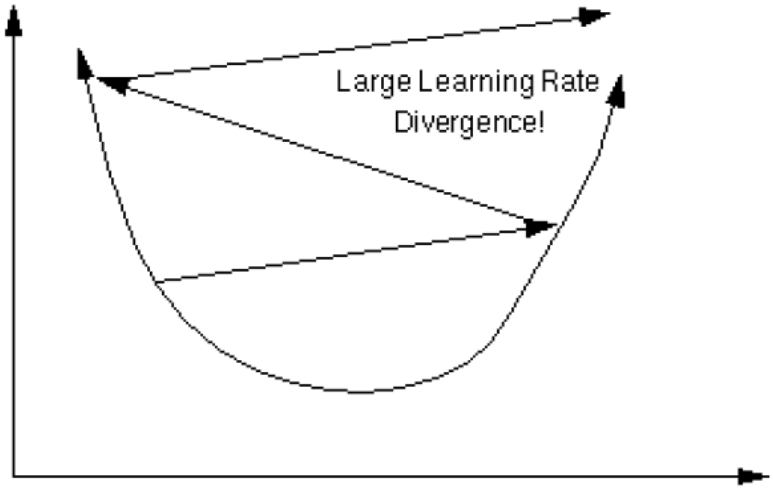
όπου η είναι ο ρυθμός μάθησης και $E(t)$ είναι το σφάλμα στην εποχή t. Η δυσκολία στη χρήση αυτού του αλγορίθμου είναι στο να βρεθεί ένας καλός ρυθμός μάθησης, δεδομένου ότι δεν υπάρχει κάποιος μαθηματικός τύπος που να τον ορίζει και η βέλτιστη τιμή του για κάθε πρόβλημα ποικίλλει.

Επίσης, πολλές φορές αντιμετωπίζεται το πρόβλημα της υπερεκπαίδευσης (overtraining) ενός δικτύου σε ένα συγκεκριμένο σετ δεδομένων. Ένα άλλο συχνό πρόβλημα που παρουσιάζεται είναι ακριβώς το αντίθετο : το δίκτυο δηλαδή, δεν μαθαίνει σχεδόν καθόλου σωστά! Αυτό συνήθως οφείλεται σε κακή ανάθεση των παραμέτρων του δικτύου.

Δεν πρέπει να ξεχνά κανείς πως ο κανόνας της κατάβασης κλίσης (gradient descent), βάσει του οποίου λειτουργεί ο αλγόριθμος ανάστροφης διάδοσης (back propagation) πρέπει να έχει μία λογική τιμή για να λειτουργεί καλά [17]: Αν είναι πολύ μικρός (σχήμα 10) η σύγκλιση θα είναι πολύ αργή ενώ αν είναι πολύ μεγάλος, το δίκτυο θα αποκλίνει (σχήμα 11).



ΣΧΗΜΑ 10 : ΜΙΚΡΟΣ ΡΥΘΜΟΣ ΜΑΘΗΣΗΣ



ΣΧΗΜΑ 11 : ΜΕΓΑΛΟΣ ΡΥΘΜΟΣ ΜΑΘΗΣΗΣ

Δυστυχώς ο καλύτερος ρυθμός μάθησης είναι τυπικά διαφορετικός για κάθε βάρος του δικτύου. Μερικές φορές οι διαφορές αυτές είναι μικρές και αμελητέες, ώστε ένας καθολικός συμβιβαστικός ρυθμός μάθησης να λειτουργήσει σωστά, άλλες φορές δεν συμβαίνει αυτό. Για να αντιμετωπίσουμε αυτό το πρόβλημα καταφεύγουμε σε διάφορες τεχνικές, οι οποίες περιλαμβάνουν αρχικά την προεπεξεργασία των δεδομένων (κανονικοποίηση εισόδων, αρχικοποίηση βαρών κτλ) και κατόπιν τις διάφορες μεθόδους προσαρμογής του ρυθμού μάθησης.

Υπάρχει μία ποικιλία τεχνικών για την προσαρμογή του ρυθμού μάθησης [15]. Η αδυναμία αυτών των μεθόδων είναι πως απαιτούν άλλες βοηθητικές παραμέτρους οι οποίες απαιτούν ρύθμιση. Ακολουθεί περιγραφή των μεθόδων αυτών

3.2.1 Όρος ορμής (momentum MOM)

Η χρήση του όρου ορμής δεν προσαρμόζει άμεσα το ρυθμό μάθησης. Αντίθετα, προσθέτει έναν όρο στην εξίσωση προσαρμογής των βαρών. Αυτός ο επιπλέον όρος είναι ένα κλάσμα της προηγούμενης ανανέωσης βάρους, που εξομαλύνει τις αλλαγές που γίνονται στο βάρος. Έτσι, ο κανόνας μάθησης γίνεται :

$$\Delta w(t) = -\eta \frac{\partial E}{\partial w}(t) + \alpha \Delta w(t-1) \tag{3.2}$$

όπου το α είναι ο παράγοντας ορμής με τιμές μεταξύ 0 και 1.

Ο παράγοντας αυτός θα πρέπει να καθοριστεί μέσα από δοκιμές. Όταν η παράγωγος εξακολουθεί να δείχνει προς την ίδια κατεύθυνση, αυτό θα αυξήσει τον αριθμό των βημάτων που χρειάζονται μέχρι το ελάχιστο. Είναι επίσης συχνά αναγκαίο να μειωθεί ο καθολικός ρυθμός μάθησης η όταν χρησιμοποιείται μεγάλος όρος ορμής (α κοντά στο 1). Αν γίνει χρήση μεγάλου ρυθμού μάθησης και μεγάλου όρου ορμής, ο αλγόριθμος θα ξεπεράσει το ελάχιστο με μεγάλα βήματα.

Όταν η παράγωγος αλλάξει κατεύθυνση, ο όρος της ορμής θα εξομαλύνει τις διαφορές. Αυτό είναι ιδιαίτερα χρήσιμο σε περίπτωση που το νευρωνικό δίκτυο δεν έχει αρχικοποιηθεί σωστά.

3.2.2 Μέθοδος Delta-Bar-Delta (DBD)

Η μέθοδος DBD προσαρμόζει άμεσα το ρυθμό μάθησης σε κάθε εποχή. Χρησιμοποιώντας αυτή τη μέθοδο, κάθε βάρος έχει το δικό του ρυθμό μάθησης που ποικίλει ανά το χρόνο ($\eta(t)$). Αν η διόρθωση βάρους ακολουθεί την ίδια πορεία με τις προηγούμενες διορθώσεις, ο ρυθμός μάθησης αυξάνεται κατά μία σταθερά (η^+). Αν δεν ακολουθεί την ίδια πορεία, τότε ένα κλάσμα (η^-) του ρυθμού μάθησης αφαιρείται από αυτόν. Έτσι έχουμε :

$$\bar{\delta}(t) = (1-\theta)\frac{\partial E}{\partial w}(t) + \theta\bar{\delta}(t-1)$$

$$\Delta\eta(t) = \begin{cases} \eta^{+}, & \alpha\nu \quad \bar{\delta}(t-1)\dfrac{\partial E}{\partial w}(t) > 0 \\ -\eta^{-}\eta(t), & \alpha\nu \quad \bar{\delta}(t-1)\dfrac{\partial E}{\partial w}(t) < 0 \\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma \end{cases} \tag{3.3}$$

Έτσι, όταν η ανανέωση του βάρους γίνεται προς την ίδια κατεύθυνση, ο ρυθμός μάθησης αυξάνεται γραμμικά. Αν όμως, η ανανέωση του βάρους ταλαντεύεται, ο ρυθμός μάθησης μειώνεται εκθετικά.

3.2.3 Μέθοδος Super-SAB (SSAB)

Υπάρχουν δύο βασικές διαφορές μεταξύ της SSAB και της DBD. Πρώτον, η SSAB αυξάνει και μειώνει το ρυθμό μάθησης εκθετικά. Δεύτερον, τα βήματα που προκαλούν την παράγωγο του βάρους να αλλάξει πρόσημο δε γίνονται. Τα επαναληπτικά βήματα της SSAB είναι τα ακόλουθα :

- Πραγματοποίησε ένα βήμα MOM:

$$\Delta w(t) = -\eta\frac{\partial E}{\partial w}(t) + \alpha\Delta w(t-1)$$

- Αν το πρόσημο της παραγώγου του δοσμένου βάρους w παραμένει το ίδιο ($\frac{\partial E}{\partial w}(t)\frac{\partial E}{\partial w}(t-1) > 0$), τότε ο ρυθμός μάθησης αυξάνεται ($\eta(t) = \eta^{+}\eta(t-1)$) και το βάρος ανανεώνεται. ($w(t) = w(t-1) + \Delta w(t)$). Αν η παράγωγος αλλάξει πρόσημο, αυτό υπονοεί ότι το τελευταίο βήμα ήταν πολύ μεγάλο και πρέπει να έχει «χαθεί» κάποιο ελάχιστο. Σε αυτές τις περιπτώσεις, ο ρυθμός μάθησης μειώνεται ($\eta(t) = \eta^{-}\eta(t-1)$), το προηγούμενο βήμα θεωρείται ανεκτέλεστο ($w(t) = w(t-1) - \Delta w(t-1)$) και το *Δw(t)* τίθεται στο 0 για να εξουδετερώσει την επιρροή του όρου ορμής σε επόμενες ανανεώσεις βαρών.

3.2.4 Resilient Propagation (RPROP)

Η RPROP είναι παρόμοια με την SSAB ως προς το ότι ο ρυθμός μάθησης αυξάνεται και μειώνεται εκθετικά και τα βήματα που προκαλούν την αλλαγή πρόσημου της παραγώγου θεωρούνται ως ανεκτέλεστα. Διαφέρει όμως, από τις περισσότερες μεθόδους σε δύο σημεία [14]. Πρώτον, μόνο το πρόσημο της παραγώγου και όχι το μέγεθος της, επηρεάζει την διόρθωση του βάρους. Δεύτερον, οι ρυθμοί μάθησης είναι φραγμένοι μεταξύ των τιμών η_{min} και η_{max}.

Το πρώτο βήμα για τον επαναληπτικό βρόχο αποτελείται από την ενημέρωση του ρυθμού μάθησης ανάλογα με το αν άλλαξε ή όχι το πρόσημο της παραγώγου :

$$\eta(t)=\begin{cases}\min(\eta^{+}\eta(t-1),\eta_{\max}) & ,\alpha\nu \quad \dfrac{\partial E(t)}{\partial w}\dfrac{\partial E(t-1)}{\partial w}>0\\ \max(\eta^{-}\eta(t-1),\eta_{\min}) & ,\alpha\nu \quad \dfrac{\partial E(t)}{\partial w}\dfrac{\partial E(t-1)}{\partial w}<0\\ \eta(t-1) & ,\alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases} \tag{3.4}$$

Αν η παράγωγος αλλάξει πρόσημο ($\frac{\partial E(t)}{\partial w}\frac{\partial E(t-1)}{\partial w}<0$), το προηγούμενο βήμα ακυρώνεται ($\Delta w(t)=-\Delta w(t-1)$) και ο όρος $\frac{\partial E(t)}{\partial w}$ τίθεται στο 0 για να εμποδίσει μία δεύτερη διαδοχική μείωση του ρυθμού μάθησης. Αλλιώς, ($\frac{\partial E(t)}{\partial w}\frac{\partial E(t-1)}{\partial w}\geq 0$), το βάρος ανανεώνεται σύμφωνα με το πρόσημο της παραγώγου :

$$\Delta w(t)=\begin{cases}-\eta(t) & ,\alpha\nu \quad \dfrac{\partial E(t)}{\partial w}>0\\ +\eta(t) & ,\alpha\nu \quad \dfrac{\partial E(t)}{\partial w}<0\\ 0 & ,\alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases} \tag{3.5}$$

με $w(t)=w(t-1)+\Delta w(t)$.

3.2.5 Γενικευμένη χωρίς-μείωση προσαρμοστική μέθοδος (Generalized no-decrease adaptive method, GNDAM)

Η GNDAM δε δίνει σε κάθε βάρος το δικό του ρυθμό μάθησης. Αντίθετα, τα βάρη χωρίζονται σε διάφορες ομάδες και ένας ρυθμός μάθησης συσχετίζεται με μία ομάδα ($\eta_g(t)$ είναι ο ρυθμός μάθησης για την ομάδα g). Σε κάθε χρονική στιγμή ανανεώνεται ένας ρυθμός μάθησης. Η βασική της διαφορά από τις υπόλοιπες μεθόδους είναι πως δε χρησιμοποιεί το πρόσημο της παραγώγου ενός δοσμένου βάρους για να προσαρμόσει το ρυθμό μάθησης αλά βασίζεται σε ευριστική προσπάθεια, όπου ο καθολικός ρυθμός μάθησης προσαρμόζεται ανάλογα με το ρυθμό σφαλμάτων που παράγεται από δύο ταυτοτικά δίκτυα που χρησιμοποιούν διαφορετικούς ρυθμούς μάθησης.

Στο πρώτο βήμα γίνεται η δημιουργία ενός ταυτοτικού δικτύου όπου μόνο ο ρυθμός μάθησης της ομάδας g αλλάζει :

$$\begin{aligned} w_i'(t) &= w_i(t), \forall i \\ \eta_i'(t) &= \eta_i(t), \forall i \neq g \\ \eta_g'(t) &= \eta_g(t) \cdot m_g(t-1) \end{aligned} \tag{3.6}$$

Στο επόμενο βήμα εφαρμόζεται η ανάστροφη διάδοση για $N_g(t)$ επαναλήψεις. Το τελευταίο βήμα διαλέγει ποιο δίκτυο ($w(t)$ και $\eta(t)$ ή $w'(t)$ και $\eta'(t)$) θα διατηρηθεί ($w(t+1)$ και $\eta(t+1)$) και ποιο θα απορριφθεί. Αν ο μεγαλύτερος ρυθμός μάθησης παρήγαγε το μικρότερο ρυθμό σφάλματος ή αν η διαφορά μεταξύ των ρυθμών σφαλμάτων των δύο δικτύων είναι αμελητέα, το δίκτυο με το μεγαλύτερο ρυθμό μάθησης επιλέγεται ($w(t+1)$ και $\eta(t+1)$ και $m_g(t)=K$). Αλλιώς, το δίκτυο με το μικρότερο ρυθμό μάθησης επιλέγεται και $m_g(t)=1/K$.

Αν και η γενική αρχή αυτής της προσέγγισης είναι σχετικά απλή, η υλοποίηση είναι πολύ πιο πολύπλοκη με δεδομένο ότι ο αριθμός των επαναλήψεων ($N_g(t)$) που χρειάζονται για τη σύγκριση δύο ρυθμών μάθησης είναι σημαντικός. Αν είναι πολύ μικρός, η διαφορά σπάνια θα είναι σημαντική και ο ρυθμός μάθησης θα γίνει πολύ μεγάλος. Αν είναι πολύ μεγάλος, τότε η διαφορά θα είναι συχνά σημαντική και οι προσομοιώσεις δείχνουν πως ο ρυθμός μάθησης θα γίνει μικρός. Για να διατηρηθεί ο αριθμός $N_g(t)$ σε μία τιμή τέτοια ώστε ο ρυθμός μάθησης να μη μειώνεται ή αυξάνεται συνέχεια, η GNAM χρησιμοποιεί μία FQUEST. Αυτή η διαδικασία είναι μία τροποποίηση της μεθόδου προσαρμογής QUEST που χρησιμοποιείται στην

ψυχοφυσική για να επιβεβαιώσει την εκ των υστέρων συνάρτηση πυκνότητας πιθανότητας (ΣΠΠ) ενός κατωφλίου. Η FQUEST δοκιμάζει να υπολογίσει το $N_g(t)$ ώστε η πιθανότητα της αύξησης του ρυθμού μάθησης να είναι 50%. Για να το πετύχει αυτό, χρησιμοποιεί μία ΣΠΠ ($PDF_g(t)$). Για την ανανέωση της ΣΠΠ, η FQUEST χρειάζεται να γνωρίζει την πιθανότητα ο μεγαλύτερος ρυθμός μάθησης να προκαλεί το μικρότερο ρυθμό σφαλμάτων. Αυτή η πιθανότητα υπολογίζεται ως εξής :

$$p_g(t) = \alpha \cdot p_g(t-1) + (1-\alpha) \cdot I(t) \tag{3.7}$$

όπου $0<\alpha<1$ και $I(t)$ ίσο με 1 αν ο μεγαλύτερος ρυθμός μάθησης προκαλεί το μικρότερο ρυθμό σφαλμάτων τη χρονική στιγμή t ή ίσο με 0 αλλιώς. Η $p_g(t)$ χρησιμοποιείται για να ορίσει την επιθυμητή πιθανότητα να υπάρχει μία σημαντική διαφορά μεταξύ των ρυθμών σφαλμάτων που δημιουργούνται από τα δύο δίκτυα, που εξαρτάται άμεσα από το $N_g(t)$. Ο στόχος της FQUEST είναι να δώσει στο $N_g(t)$ μία τιμή ώστε η πιθανότητα ο ρυθμός μάθησης να αυξάνεται, να είναι 50%. Εφόσον η πιθανότητα ο ρυθμός μάθησης να αυξηθεί (0.5) είναι ίση με την πιθανότητα ο υψηλότερος ρυθμός μάθησης να προκαλεί το μικρότερο ρυθμό σφαλμάτων ($p_g(t)$) πολλαπλασιασμένη με την πιθανότητα να έχουν μία σημαντική διαφορά, έχουμε τελικά πως η τελευταία αυτή πιθανότητα (να έχουν δηλαδή σημαντική διαφορά) είναι ίση με $0.5 / p_g(t)$. Έτσι, βρίσκοντας το κατάλληλο $N_g(t)$ θα έπρεπε να επιτρέπει στο ρυθμό μάθησης να παραμένει σχετικά σταθερός. Η συνάρτηση που χρησιμοποιείται για τον υπολογισμό της εκ των υστέρων συνάρτησης πυκνότητας πιθανότητας μπορεί να συνοψιστεί σαν ένα παράγωγο των ψυχομετρικών συναρτήσεων :

$$PDF_g(t) = PDF_g(t-1)^{\frac{1}{\lambda\sqrt{2}}} \cdot \Psi(N_g(t), 0.5 / p_g(t), S_g(t)) \tag{3.8}$$

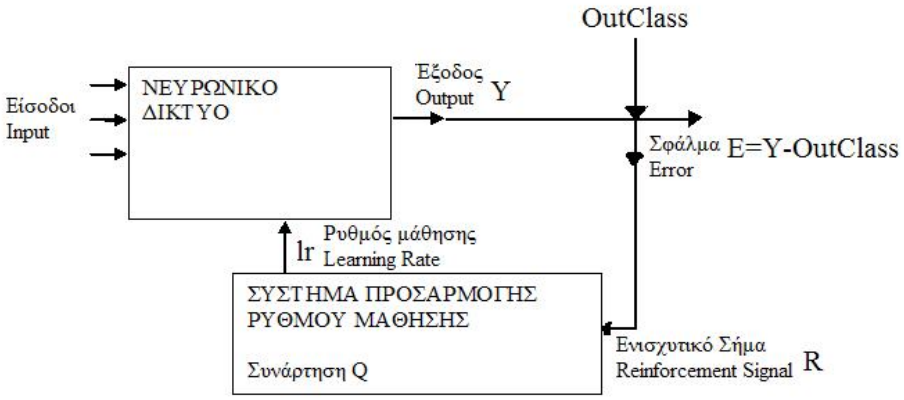
όπου λ είναι ο χρόνος ημίσειας ζωής (ο αριθμός των προσπαθειών που εξασθενούν το βάρος της δοσμένης ψυχοφυσικής συνάρτησης κατά 50%) και Ψ είναι μία ψυχοφυσική συνάρτηση που τροποποιεί την ΣΠΠ ανάλογα με την καινούρια προσπάθεια (t). Η Ψ εξαρτάται από τον αριθμό των επαναλήψεων ($N_g(t)$), το κατώφλι της πιθανότητας να υπάρχει μια επιθυμητή διαφορά ($0.5/p_g(t)$) και από το αν υπήρχε ή όχι μια σημαντική διαφορά τη στιγμή t ($S_g(t)$). Η σημαντική παράμετρος είναι το λ, γιατί σταθμίζει τις διάφορες προσπάθειες. Αν το λ είναι μεγάλο, πολλές προηγούμενες προσπάθειες θα χρησιμοποιηθούν για τον υπολογισμό της εκ των υστέρων ΣΠΠ, το οποίο μπορεί να οδηγήσει σε καλύτερη προσέγγιση της ΣΠΠ αν το κατώφλι (επιθυμητός αριθμός επαναλήψεων) είναι σταθερό. Όμως, αν ο επιθυμητός

αριθμός των επαναλήψεων αλλάζει, η ΣΠΠ θα απαιτεί περισσότερο χρόνο για να ξαναπροσαρμοστεί στο νέο κατώφλι. Με ένα σχετικά χαμηλό λ, έχουμε την ακριβώς αντίθετη συμπεριφορά : η εκ των υστέρων ΣΠΠ προσαρμόζεται πιο εύκολα σε ένα νέο κατώφλι και η προσέγγισή της είναι λιγότερο ακριβής.

4 *Μεθοδολογίες και τεχνικές – Ανάπτυξη του μοντέλου*

4.1 Εισαγωγή

Όπως αναφέρθηκε ήδη στα προηγούμενα, το μοντέλο το οποίο μελετάμε είναι ένας ταξινομητής. Ο ταξινομητής αυτός δέχεται εισόδους που απαρτίζονται από διαφορετικά χαρακτηριστικά στοιχεία κάποιου συνόλου δεδομένων και οι οποίες καταλήγουν στους νευρώνες του κρυμμένου στρώματος του νευρωνικού δικτύου. Κατόπιν, οι νευρώνες αυτοί συνδυάζονται και ορίζουν την έξοδο, η οποία αντιστοιχεί στην κατηγορία που ανήκουν κάθε ένα από τα πρότυπα εισόδου. Το μοντέλο το οποίο δημιουργήσαμε έχει τη μορφή που φαίνεται στο σχήμα 12 (κατά τα πρότυπα της ενισχυτικής μάθησης).



ΣΧΗΜΑ 12 : ΔΟΜΗ ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΣΑΡΜΟΓΗΣ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ

Περισσότερες λεπτομέρειες για την παραπάνω δομή αναλύονται στα ακόλουθα.

4.2 Αλγόριθμος ανάστροφης διάδοσης

Το νευρωνικό δίκτυο στο οποίο δουλεύουμε επιλέχτηκε να έχει 7 νευρώνες ώστε να είναι δύσκολο το έργο της εφαρμογής αλλά και να έχει αυτή αποτελέσματα σε ένα περιβάλλον πιο δύσκολο. Επιπλέον η παρουσία παραπάνω νευρώνων θα επιβάρυνε την ταχύτητα της εφαρμογής, παράγοντας ο οποίος θεωρείται σημαντικός.

Η συνάρτηση εκπαίδευσης που χρησιμοποιούμε βασίζεται στον κανόνα της πιο απότομης κλίσης. Τα βάρη(weights) και οι πολώσεις (biases) αρχικοποιούνται σε τυχαίες τιμές ενώ η ανανέωση των τιμών τους γίνεται στην κατεύθυνση της αρνητικής παραγώγου του τετραγωνικού σφάλματος. Ο ρυθμός μάθησης πολλαπλασιάζεται με την αρνητική αυτή κλίση για να καθοριστούν οι αλλαγές στα βάρη και τις πολώσεις. Όσο μεγαλύτερος γίνεται ο ρυθμός μάθησης, τόσο μεγαλύτερο είναι το βήμα. Χρησιμοποιούμε τη μέθοδο της φουρνιάς προτύπων (batch training), δηλαδή η εκπαίδευση βασίζεται σε ολόκληρο το σύνολο των παραδειγμάτων εκπαίδευσης που ονομάζεται εποχή (epoch). Κατά αυτή την έννοια η ανανέωση των βαρών γίνεται αφού παρουσιαστούν στο δίκτυο όλα τα διατιθέμενα παραδείγματα εκπαίδευσης [3].

Η έξοδος του νευρωνικού δικτύου συγκρίνεται για κάθε πρότυπο και έτσι βρίσκεται το μέσο τετραγωνικό σφάλμα για κάθε εποχή-φουρνιά. Στόχος φυσικά είναι το σφάλμα αυτό να ελαχιστοποιηθεί. Ακριβώς για αυτό το λόγο επιλέγεται και το σφάλμα κάθε φουρνιάς να αποτελέσει και τον καθοριστικό παράγοντα της ανταμοιβής από το περιβάλλον.

4.3 Το σύστημα ενισχυτικής μάθησης

4.3.1 Το ενισχυτικό σήμα

Σε ένα σύστημα ενισχυτικής μάθησης προέχει να καθοριστεί το σήμα ενίσχυσης από το περιβάλλον και πως αυτό θα ορίζεται και θα δίνεται. Στην περίπτωσή μας, μία καλή περίπτωση είναι η χρήση του σφάλματος της εξόδου σαν ανταμοιβή από το περιβάλλον. Ο πράκτορας θα βλέπει αν το σφάλμα που έχει στην έξοδό του το μοντέλο, ύστερα από την εκπαίδευση είναι μεγάλο ή μικρό και ανάλογα θα δίνει την ανταμοιβή ή την τιμωρία. Επειδή όμως είναι αρκετά ασαφές ποιο σφάλμα θεωρείται μεγάλο και ποιο σφάλμα θεωρείται μικρό, χρησιμοποιείται ένα πιο συγκεκριμένο μέτρο σύγκρισης το οποίο είναι η διαφορά σφάλματος ανάμεσα στην τρέχουσα και στην προηγούμενη εποχή. Αυτό σημαίνει πως αν το σφάλμα μειωθεί ανάμεσα σε δύο εποχές, τότε αυτό είναι ένα σημάδι πως η δράση η οποία επιλέχθηκε από το σύστημα με τη βοήθεια του πράκτορα ήταν σωστή και είχε καλό αποτέλεσμα για το δίκτυο, συνεπώς και πρέπει να ανταμειφθεί. Αντίστοιχα, αν το σφάλμα αυξηθεί ανάμεσα σε δύο εποχές, τότε η προηγούμενη επιλεχθείσα πράξη δεν ήταν καλή και συνεπώς πρέπει να δοθεί μία ποινή τιμωρίας.

Το επόμενο ζήτημα που έπρεπε να αντιμετωπιστεί ήταν ποιο θα ήταν το μέγεθος αυτής της ανταμοιβής. Αρχικά επιλέχτηκε η «σταθερή» ανταμοιβή για κάθε περίπτωση, δηλαδή σε περίπτωση καλής επιλογής πράξης (μείωση σφάλματος) να υπάρχει ανταμοιβή της τάξης του 1 ή του 10 και ανάλογα σε περίπτωση κακής επιλογής πράξης (αύξηση σφάλματος) η τιμή αυτή να είναι -1 ή -10. Γρήγορα όμως, έγινε κατανοητό πως καλύτερο αποτέλεσμα θα υπήρχε αν η ανταμοιβή (ή τιμωρία) θα είχε μεγαλύτερη επίδραση αν ήταν ανάλογη και του αποτελέσματός της. Δηλαδή, όσο μεγαλύτερη μείωση σφάλματος είχαμε, τόσο μεγαλύτερη να είναι και η ανταμοιβή (αντίστοιχα αν είχαμε μεγάλη αύξηση σφάλματος, η ποινή τιμωρίας θα έπρεπε να

είναι σχετικά μικρή). Αυτό μας οδηγεί σε μία ανταμοιβή από το περιβάλλον ανάλογη της διαφοράς του σφάλματος των δύο τελευταίων εποχών, επομένως έως τώρα διαχωρίζουμε τρεις κατηγορίες του σήματος ενίσχυσης :

- Μείωση σφάλματος : ανταμοιβή ανάλογη της απόλυτης τιμής του μεγέθους της μείωσης αυτής
- Αύξηση σφάλματος : τιμωρία ανάλογη της αύξησης αυτής
- Σταθερό σφάλμα : Η περίπτωση αυτή σπάνια παρατηρείται μιας και είναι σχετικά δύσκολο το σφάλμα σε δύο διαδοχικές εποχές να μείνει απόλυτα σταθερό, παρόλα αυτά πρέπει να καλυφθεί. Σε αυτή την περίπτωση η ανταμοιβή είναι μηδενική.

Μία τελευταία βελτίωση που έγινε στο θέμα του σήματος ενίσχυσης ήταν να γίνει ακόμη μεγαλύτερος διαχωρισμός των κατηγοριών ανάλογα με το ποσοστό της αύξησης ή μείωσης του σφάλματος. Αν η μεταβολή του σφάλματος ξεπερνούσε κάποιο ποσοστό, το οποίο εμείς καθορίζουμε, τότε σημαίνει πως η μεταβολή που επήλθε είναι πολύ σημαντική σε μέγεθος και επομένως πρέπει το σύστημα να ανταμειφθεί ή να τιμωρηθεί περισσότερο απ' ότι αν η μεταβολή ήταν μικρότερη. Έτσι τελικά, καταλήγουμε σε 5 κατηγορίες του σήματος ενίσχυσης ανάλογα με τη διαφορά σφάλματος (ΔΣ) και το ποσοστό της μεταβολής (ΠΜ) :

Με Χ συμβολίζουμε το ποσοστό που εμείς ορίζουμε ότι πάνω από αυτό θεωρείται μία μεταβολή μεγάλη.

Διαφορά Σφάλματος	Ποσοστό Μεταβολής	Ενισχυτικό Σήμα
0	0	0
< 0	> Χ	Μεγάλο θετικό
< 0	< Χ	Μικρό θετικό
> 0	> Χ	Μεγάλο αρνητικό
> 0	< Χ	Μικρό αρνητικό

ΠΙΝΑΚΑΣ 1 : ΔΙΑΡΘΡΩΣΗ ΕΝΙΣΧΥΤΙΚΟΥ ΣΗΜΑΤΟΣ

Προφανώς το ενισχυτικό σήμα, προκύπτει με πολλαπλασιασμό της διαφοράς σφάλματος με κάποιον παράγοντα, ανάλογα με το σε ποια κατηγορία από αυτές του πίνακα 1 βρισκόμαστε. Η πειραματική διαδικασία με διάφορα σύνολα δεδομένων υπέδειξε σαν τιμές ικανές να δημιουργήσουν "καλές" και "κακές" ενισχύσεις, τους παράγοντες 3,2,-3,-2 αντίστοιχα για τις κατηγορίες του πίνακα 1. Επίσης, χρειάζεται μία σημαντική τάξη μεγέθους, ώστε να μπορέσουμε να έχουμε τιμές της συνάρτησης Q που θα είναι πιο κοντά σε φυσικούς αριθμούς, διαδικασία που διευκολύνει την παρατήρηση και τους υπολογισμούς και έτσι καταλήγουμε στην ακόλουθη ακριβή μορφή για το ενισχυτικό σήμα :

$$\mathrm{Reinf} = \begin{cases} 0, \Delta\Sigma = 0 \\ 30 \cdot |\Delta\Sigma|, \Delta\Sigma < 0 \quad \kappa\alpha\iota \quad \Pi\Mu > \mathrm{X} \\ 20 \cdot |\Delta\Sigma|, \Delta\Sigma < 0 \quad \kappa\alpha\iota \quad \Pi\Mu < \mathrm{X} \\ -20 \cdot |\Delta\Sigma|, \Delta\Sigma > 0 \quad \kappa\alpha\iota \quad \Pi\Mu < \mathrm{X} \\ -30 \cdot |\Delta\Sigma|, \Delta\Sigma > 0 \quad \kappa\alpha\iota \quad \Pi\Mu > \mathrm{X} \end{cases} \tag{4.1}$$

όπου χρησιμοποιούμε τους συμβολισμούς (κατά τον πίνακα 1):

Reinf : για το ενισχυτικό σήμα

ΔΣ : διαφορά σφάλματος

ΠΜ : ποσοστό μεταβολής

4.3.2 Οργάνωση του πίνακα τιμών της συνάρτησης Q

Έπειτα από τον καθορισμό του ενισχυτικού σήματος, ακολουθεί ο καθορισμός της συνάρτησης Q και του πίνακα τιμών της, ώστε να ανταποκριθεί στο σύστημά μας. Από τις μεθοδολογίες ανάπτυξης του μοντέλου μάθησης Q που αναφέρθηκαν στην παράγραφο 2.9.1 θα χρησιμοποιήσουμε τη μέθοδο αποθήκευσης των τιμών σε πίνακα.

Δύο είναι τα βασικά πράγματα τα οποία πρέπει να καθοριστούν: ο χώρος των καταστάσεων και ο χώρος των δράσεων. Το δεύτερο είναι πιο εύκολο να οριστεί μιας και αφού το σύστημά μας προσαρμόζει το ρυθμό μάθησης, το προφανές είναι πως η επιλογή των δράσεων θα έχει να κάνει με τη μεταβολή αυτής της παραμέτρου του νευρωνικού δικτύου. Έτσι, αρχικά καθορίσαμε δύο δράσεις που είχαν να κάνουν με

την αύξηση ή τη μείωση του ρυθμού μάθησης αντίστοιχα. Από την πειραματική διαδικασία έγινε κατανοητό πως η ύπαρξη μιας δράσης για την αύξηση και μιας δράσης για τη μείωση δεν αποδίδει απόλυτα μιας και υπάρχουν περιπτώσεις που ο ρυθμός μάθησης πρέπει να αυξηθεί ή να μειωθεί πολύ (συνήθως όταν το σφάλμα είναι αρκετά μεγάλο) και αντίστοιχα περιπτώσεις που ο ρυθμός μάθησης πρέπει να αυξηθεί ή να μειωθεί λίγο (όταν το σφάλμα έχει ήδη μειωθεί αρκετά και μια μεγάλη μεταβολή του ρυθμού μάθησης θα οδηγήσει σε αστάθεια το σύστημα). Έτσι τελικά, καταλήγουμε στον καθορισμό τεσσάρων δράσεων ως ακολούθως :

Δράση	Μεταβολή ρυθμού μάθησης	Αποτέλεσμα
1	$lr * sigma^2$	Μεγάλη αύξηση
2	$lr * sigma$	Μικρή αύξηση
3	$lr / sigma$	Μικρή μείωση
4	$lr / sigma^2$	Μεγάλη μείωση

ΠΙΝΑΚΑΣ 2 : ΧΩΡΟΣ ΤΩΝ ΔΡΑΣΕΩΝ

Όπου συμβολίζουμε με :

- *lr* : τον προηγούμενο ρυθμό μάθησης
- *sigma* : την παράμετρο που θα καθορίσει το μέγεθος της μεταβολής του ρυθμού μάθησης και η οποία αποτελεί αντικείμενο πειραματικής μελέτης.

Σχετικά με την παράμετρο *sigma*, μία προσπάθεια που έγινε ήταν να ακολουθηθεί τακτική παρόμοια με αυτή της ενίσχυσης, δηλαδή να είναι ανάλογη της διαφοράς σφάλματος ή του ίδιου του σφάλματος. Παρόλα αυτά η τακτική αυτή δεν ήταν ιδιαίτερα αποδοτική λόγω του μεγάλου εύρους των τιμών του σφάλματος και οδηγούσε το σύστημα σε αστάθειες και ταλαντώσεις μεταξύ μιας αέναης αύξησης και μείωσης του ρυθμού μάθησης με καμία επίδραση στο σύστημα. Έτσι επιλέχτηκε η παράμετρος *sigma* να έχει μία σταθερή τιμή, όπως για παράδειγμα 1.2 ή 1.25, τιμή η οποία φαίνεται πως δε δημιουργεί τα προηγούμενα προβλήματα.

Ένα άλλο θέμα το οποίο πρέπει να αντιμετωπιστεί στην πράξη είναι αυτό που αναφέρθηκε στην παράγραφο 2.2 και έχει να κάνει με την επιλογή δράσεων και τον τρόπο που αυτή θα γίνει. Υπενθυμίζεται πως ο πράκτορας μπορείτε να επιλέγει, είτε βάσει της γνώσης του για το ποια δράση του αποφέρει μεγαλύτερη ενίσχυση και να εξαντλεί αυτή του την επιλογή είτε βάσει εξερεύνησης για άλλες δράσεις που μπορεί να του αποφέρουν μεγαλύτερη ανταμοιβή. Στην περίπτωσή μας αντιμετωπίζουμε αυτό το ζήτημα, βάσει όσων αναφέρθησαν στην παράγραφο 2.9.2, ως εξής : Ορίζουμε έναν παράγοντα εξερεύνησης *exf* (**ex**ploration **f**actor) με αρχική τιμή κοντά στο 0.9. Επιλέγοντας έναν τυχαίο αριθμό μεταξύ 0 και 1, θεωρούμε πως αν αυτός είναι μικρότερος από τον παράγοντα *exf* τότε το σύστημα θα προχωρήσει σε εξερεύνηση δηλαδή θα επιλέξει μία τυχαία δράση. Σε αντίθετη περίπτωση, ο πράκτορας θα επιλέξει μία δράση βάσει του υπάρχοντα πίνακα τιμών της συνάρτησης Q, δηλαδή τη δράση που βάσει των προηγούμενων αποτελεσμάτων δίνει τη μεγαλύτερη ανταμοιβή για τη συγκεκριμένη κατάσταση. Ο παράγοντας *exf* φθίνει σε κάθε εποχή ώστε καθώς περνάει ο χρόνος, ο πράκτορας να επιλέγει όλο και περισσότερο βάσει του πίνακα τιμών Q, επομένως βάσει της εμπειρίας που έχει για τα αποτελέσματα των δράσεων, εμπειρία που προφανώς μεγαλώνει με το πέρασμα των εποχών.

Ας έρθουμε τώρα στο δυσκολότερο πρόβλημα του καθορισμού του χώρου των καταστάσεων. Όπως έχει ήδη αναφερθεί στην παράγραφο 2.9.5, το πρόβλημα ενός μεγάλου χώρου καταστάσεων είναι ένα ζήτημα το οποίο πρέπει να αντιμετωπιστεί. Τα προβλήματα που ως τώρα έχουν αντιμετωπιστεί από τη μάθηση Q (όπως το [9]) είχαν να κάνουν με συγκεκριμένα θέματα στα οποία ο χώρος καταστάσεων μπορούσε να οριστεί σαφώς και με ακρίβεια. Στη συγκεκριμένη περίπτωση δεν ισχύει κάτι τέτοιο. Ο χώρος των καταστάσεων έπρεπε να οριστεί με κάποιο έμμεσο τρόπο ώστε να αντιπροσωπεύει σε κάθε στιγμή το σύστημα. Η αρχική σκέψη ήταν ο χώρος των καταστάσεων να μοντελοποιείται από τις εποχές στις οποίες τρέχουμε το σύστημα. Αυτό σημαίνει πως κάθε εποχή θα αντιπροσωπεύει μία κατάσταση. Έτσι θα είχαμε έναν πίνακα τιμών Q ο οποίος θα είχε σαν γραμμές τις εποχές και σαν στήλες τις δράσεις και οι τιμές του θα αντανακλούσαν το αν ο συνδυασμός δράσης-κατάστασης είναι καλός ή όχι για το σύστημα. Παρόλα αυτά, τα πειραματικά αποτελέσματα έδειξαν πως ο πράκτορας με αυτό το χώρο καταστάσεων δεν αποδίδει σωστά. Κινούμενοι από το γεγονός ότι οι καταστάσεις πρέπει να αντιπροσωπεύουν φάσεις

του συστήματος και μάλιστα φάσεις στις οποίες θέλουμε το σύστημα να καταλαβαίνει αν είναι καλό ή κακό να βρίσκεται εκεί, καταλήξαμε στο ότι είναι θεμιτό οι καταστάσεις του συστήματος να καθορίζονται από τη μεταβολή του σφάλματος που είχαμε στις δύο τελευταίες εποχές. Έτσι, ορίζουμε τρεις καταστάσεις ανάλογα με το αν η διαφορά σφάλματος ανάμεσα στις δύο τελευταίες εποχές αυξήθηκε, μειώθηκε ή παρέμεινε ίδια. Όπως έγινε και προηγούμενα με το ενισχυτικό σήμα και τις δράσεις έτσι και εδώ, γίνεται διαχωρισμός ανάλογα με το αν το σφάλμα μεγάλωσε ή μίκρυνε πολύ. Ο διαχωρισμός γίνεται με βάση το ίδιο ποσοστό X που καθορίζεται και το μέγεθος της ανταμοιβής / τιμωρίας (και καθορίζεται από το χρήστη). Έτσι για το χώρο καταστάσεων του προβλήματος έχουμε :

Κατάσταση	Διαφορά σφάλματος	Ποσοστό μεταβολής	Περιγραφή
1	< 0	> X	Μεγάλη μείωση
2	< 0	< X	Μικρή μείωση
3	0	0	Σταθερό σφάλμα
4	> 0	< X	Μικρή αύξηση
5	> 0	> X	Μεγάλη αύξηση

ΠΙΝΑΚΑΣ 3 : ΧΩΡΟΣ ΚΑΤΑΣΤΑΣΕΩΝ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

Έτσι έχουμε ορίσει πλήρως τη μορφή του πίνακα τιμών της συνάρτησης Q. Θα αποτελείται από γραμμές που θα αντιπροσωπεύουν τις καταστάσεις του συστήματος και από στήλες που θα αντιπροσωπεύουν τις δράσεις που δύναται να επιλεγούν σε κάθε κατάσταση κάθε εποχή. Πρόκειται δηλαδή βάσει και των πινάκων 1 και 3 για μία δομή πίνακα 5X4, ως ακολούθως :

$$Q(s,a) = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \\ q_{51} & q_{52} & q_{53} & q_{54} \end{bmatrix} \tag{4.2}$$

Τέλος, αξίζει να γίνει μία αναφορά στη συνάρτηση Q και στον τρόπο με τον οποίο γίνεται η ανανέωσή της. Η μορφή της είναι ως εξής :

$$Q(s,a) = Q(s,a) + alpha\left(REINF + gamma \cdot \max_{a' \in A} Q(s',a') - Q(s,a)\right) \tag{4.3}$$

όπου :

- *Q(s,a)* : συνάρτηση τιμών Q
- *alpha* : ρυθμός μάθησης πράκτορα
- *REINF* : ενισχυτικό σήμα
- *gamma* : παράγοντας έκπτωσης
- *s'* η επόμενη κατάσταση από την *s*, με επιλογή της δράσης *α*.

4.3.3 Διαδικασία εκτέλεσης εφαρμογής

Η διαδικασία που ακολουθείται στην εκτέλεση της εφαρμογής είναι η ακόλουθη :

1. Απαραίτητες αρχικοποιήσεις μεταβλητών
2. Αρχικοποίηση νευρωνικού δικτύου και παραμέτρων του
3. Εκπαίδευση νευρωνικού δικτύου
4. Λήψη σφάλματος, σήματος ενίσχυσης
5. Υπολογισμός νέας κατάστασης
6. Ανανέωση συνάρτησης Q
7. Επιλογή δράσης – προσαρμογή ρυθμού μάθησης
8. Επανάληψη βημάτων **3-7** για τον ορισμένο αριθμό εποχών
9. Επανάληψη βημάτων **2-8** για τον ορισμένο αριθμό προσπαθειών
10. Εύρεση βέλτιστης πολιτικής

ΣΧΗΜΑ 13 : ΑΛΓΟΡΙΘΜΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΕΦΑΡΜΟΓΗΣ

Όπως φαίνεται και από τον τρόπο λειτουργίας της εφαρμογής, στο τέλος της εκτέλεσης όλων των προσπαθειών, παίρνουμε μία βέλτιστη πολιτική. Πρόκειται για μία μονοδιάστατη δομή 5 στοιχείων, όπου το κάθε στοιχείο της αντιπροσωπεύει τη βέλτιστη δράση που πρέπει να επιλεγεί σε κάθε κατάσταση :

$$BestPolicy = \begin{bmatrix} bp_1 & bp_2 & bp_3 & bp_4 & bp_5 \end{bmatrix} \tag{4.4}$$

Οι 5 καταστάσεις είναι αυτές που παρουσιάστηκαν στον πίνακα 3, ενώ οι δράσεις είναι 4 και περιγράφονται στον πίνακα 2.

Η παραπάνω διαδικασία εφαρμόζεται στο σύνολο δεδομένων εκπαίδευσης και μας δίνει τη βέλτιστη πολιτική, που πρέπει να ακολουθηθεί σε κάθε εποχή ανάλογα με την κατάσταση στην οποία βρίσκεται το σύστημα. Κατόπιν, έχοντας αυτή τη βέλτιστη πολιτική, μπορούμε να προχωρήσουμε στον έλεγχο της ορθής λειτουργίας του συστήματος και τη σύγκριση με τον απλό αλγόριθμο.

5 Πειραματική Μελέτη

5.1 Σύνολα Δεδομένων

Η εφαρμογή και η αξιολόγηση της μεθόδου που αναπτύχθηκε έγινε σε διαφορετικά σύνολα δεδομένων τα οποία παρουσιάζουν διαφορετικά χαρακτηριστικά ώστε να εκτιμηθούν οι δυνατότητες και οι αδυναμίες της. Τα σύνολα δεδομένων μπορούν να βρεθούν στη βάση δεδομένων μηχανικής μάθησης UCI [18].

5.1.1 Pima Indians Diabetes (Διαβητικοί Ινδιάνοι φυλής Pima)

Πρόκειται για δεδομένα τα οποία προέρχονται από μία φυλή Ινδιάνων, από την οποία έχουν δειγματοληφθεί 8 χαρακτηριστικά με συνεχείς τιμές (ηλικία, διαστολική πίεση αίματος, φορές έγκυος κλπ) και βάσει αυτών τα άτομα χαρακτηρίζονται διαβητικά ή όχι, άρα θέλουμε να επιτύχουμε ταξινόμηση σε 2 κατηγορίες. Τα δείγματα είναι 768 από τα οποία 268 είναι θετικά στο διαβήτη και 500 αρνητικά. Χρησιμοποιούμε τα 468 πρότυπα για εκπαίδευση και τα 300 για έλεγχο.

5.1.2 Breast cancer (Καρκίνος του μαστού)

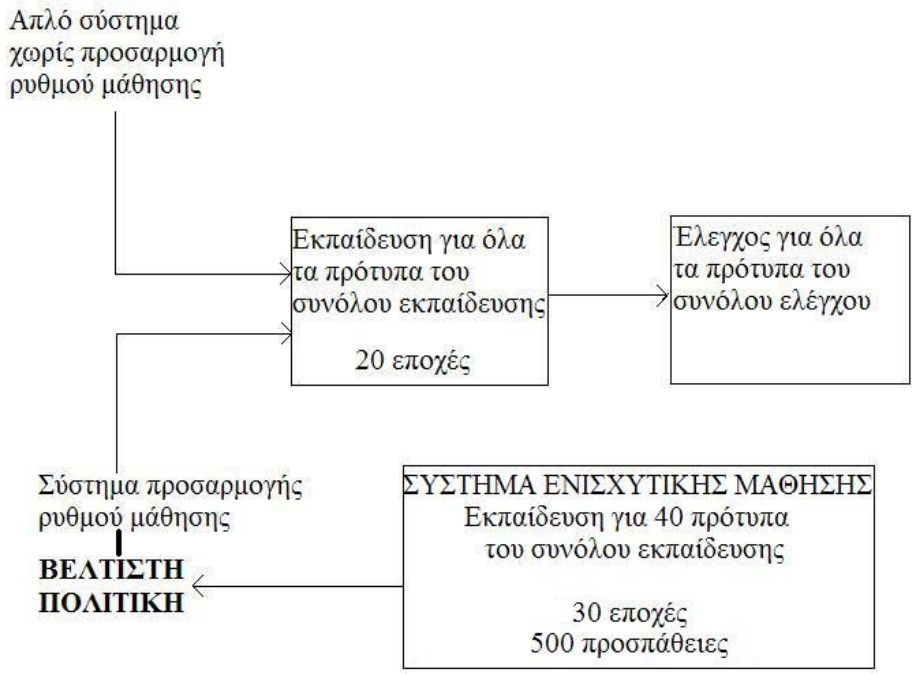
Πρόκειται για δεδομένα τα οποία προέρχονται από ένα νοσοκομείο του Wisconsin. Στη διάθεσή μας υπάρχουν 10 γνωρίσματα και βάσει αυτών φαίνεται αν ο όγκος είναι καλοήθης ή κακοήθης. Τα δείγματα είναι 699, 458 καλοήθη και 241 κακοήθη. Χρησιμοποιούμε τα 400 πρότυπα για εκπαίδευση και τα 299 για έλεγχο.

5.1.3 Ionosphere (Ιονόσφαιρα)

Πρόκειται για δεδομένα που προέρχονται από ένα σύστημα ραντάρ και περιγράφουν σήματα που είτε βρήκαν κάποιο σωματίδιο στην ιονόσφαιρα και χαρακτηρίζονται ως «καλά» ή δεν βρήκαν τίποτα και χαρακτηρίζονται «άσχημα». Έχουμε 34 χαρακτηριστικά σε ένα σύνολο 351 προτύπων. Από αυτά τα δείγματα, 201 χρησιμοποιούνται για εκπαίδευση και τα 150 για έλεγχο.

5.2 Πειραματική διαδικασία

Αξίζει να αναφερθεί πως η πειραματική έρευνα που πραγματοποιήσαμε ακολουθεί μια ιδιαίτερη διαδικασία που περιγράφεται στο σχήμα 14.



ΣΧΗΜΑ 14 : ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ ΓΙΑ ΈΛΕΓΧΟ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Από το παραπάνω σχήμα είναι σαφές πως το σύστημα της ενισχυτικής μάθησης (Q-learning) προηγείται της διαδικασίας εκπαίδευσης και καθορίζει τη βέλτιστη πολιτική μεταβολής του ρυθμού μάθησης. Κατόπιν, συγκρίνεται η επίδοση των δύο συστημάτων, του απλού συστήματος το οποίο δεν παρουσιάζει καμία μεταβολή στο

ρυθμό μάθησης και του συστήματος που έχουμε φτιάξει, το οποίο θα μεταβάλλει το ρυθμό μάθησης με βάση τη βέλτιστη πολιτική που έχει ήδη υπολογιστεί.

Το σύστημα ενισχυτικής μάθησης χρησιμοποιεί μόνο 40 πρότυπα, τυχαία επιλεγμένα, για την εύρεση της βέλτιστης πολιτικής καθώς η διαδικασία θα έπαιρνε πολύ παραπάνω χρόνο για την εκπαίδευση όλων των προτύπων, ενώ τα αποτελέσματα θα ήταν παρόμοια. Η διαδικασία αυτή περιλαμβάνει 500 προσπάθειες (Trials) εκμάθησης, όπου η κάθε μία περιλαμβάνει 30 εποχές εκπαίδευσης του δικτύου, , ώστε να εξασφαλίσουμε πως η πολιτική στην οποία έχουμε καταλήξει είναι η ορθότερη δυνατή. Να σημειωθεί πως στο τέλος κάθε προσπάθειας η συνάρτηση Q μένει αναλλοίωτη και δεν μηδενίζεται.

Το κοινό κομμάτι της εκπαίδευσης εκτελείται για 20 εποχές εκπαιδεύοντας όλα τα πρότυπα για να έχουμε καλύτερες επιδόσεις ταχύτητας και να φανεί η λειτουργία του συστήματος υπό πιο δύσκολες συνθήκες. Μεγαλύτερος αριθμός εποχών θα οδηγούσε σε σαφώς καλύτερα αποτελέσματα αλλά θα χρειαζόταν περισσότερο χρόνο και το απλό δίκτυο ίσως προλάβαινε να συγκλίνει.

5.3 Παρουσίαση αποτελεσμάτων

Στα πλαίσια της διπλωματικής εργασίας έγιναν πολλά πειράματα με διαφορετικές μορφές κώδικα και διαφορετικές τιμές στις διάφορες κρίσιμες παραμέτρους της μάθησης (αριθμός εποχών, προσπαθειών, κλπ). Θεωρείται σκόπιμο να παρουσιαστούν εδώ τα τελικά αποτελέσματα για τις παραμέτρους που κρίνουμε ότι βελτιστοποιούν τον τρόπο λειτουργίας της εφαρμογής. Τα αποτελέσματα αφορούν το σύνολο των δεδομένων και εκφράζονται σε ποσοστά επί του συνολικού πλήθους προτύπων (ΕΠ=επιτυχημένα πρότυπα).

Σε κάθε περίπτωση τα αποτελέσματα συγκρίνονται με την απλή περίπτωση όπου δε γίνεται προσαρμογή ρυθμού μάθησης (ΑΣ=απλό σύστημα) για να φανεί η διαφορά αλλά και η χρησιμότητα του συστήματος που δημιουργήσαμε. Επίσης, το σύστημα ενισχυτικής μάθησης (ΣΕΜ) δοκιμάστηκε για διάφορες αρχικές τιμές του ρυθμού μάθησης (σε σχέση πάντα με το απλό σύστημα) ώστε να παρουσιαστεί και η δυνατότητα προσαρμογής του.

5.3.1 Αποτελέσματα από το σύνολο δεδομένων Pima

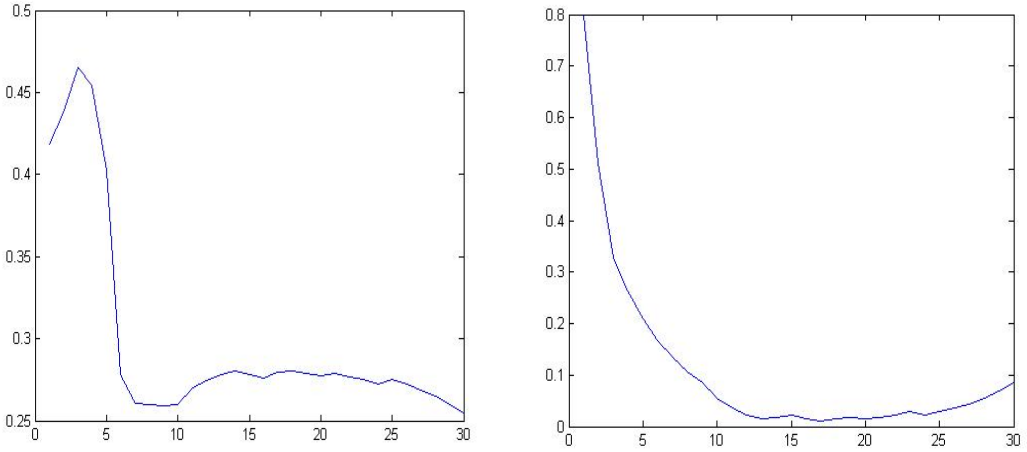
Το σύστημα ενισχυτικής μάθησης που σχεδιάσαμε και υλοποιήσαμε, έδωσε τα παρακάτω αποτελέσματα, όσον αφορά τις βέλτιστες πολιτικές που πρέπει να ακολουθηθούν σε κάθε περίπτωση.

Υπενθυμίζεται η σχέση (4.4) η οποία δίνει τη μορφή της βέλτιστης πολιτικής. Πρόκειται για έναν πίνακα-γραμμή, τα στοιχεία του οποίου δείχνουν τη δράση που πρέπει να επιλεγεί βάσει του πίνακα 2(τιμές 1 έως 4) και οι 5 δείκτες δείχνουν την κατάσταση στην οποία βρισκόμαστε βάσει του πίνακα 3.

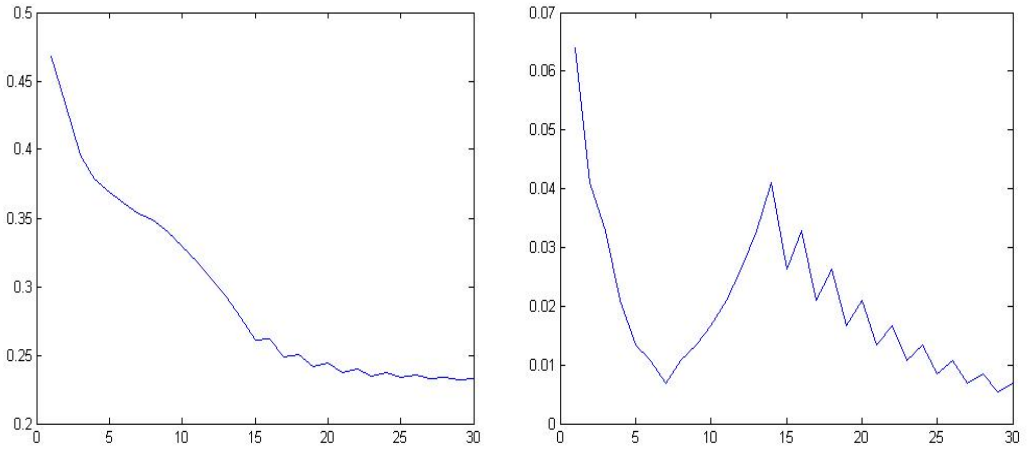
αρχικός ρυθμός μάθησης	επιλογή δράσης από κατάσταση 1	επιλογή δράσης από κατάσταση 2	επιλογή δράσης από κατάσταση 3	επιλογή δράσης από κατάσταση 4	επιλογή δράσης από κατάσταση 5
1	3	2	3	2	3
0.1	2	2	4	4	3
0.01	3	4	3	4	2
0.001	1	1	2	4	4
0.0001	2	2	3	3	4

ΠΙΝΑΚΑΣ 4 : ΒΕΛΤΙΣΤΕΣ ΠΟΛΙΤΙΚΕΣ ΓΙΑ ΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ PIMA

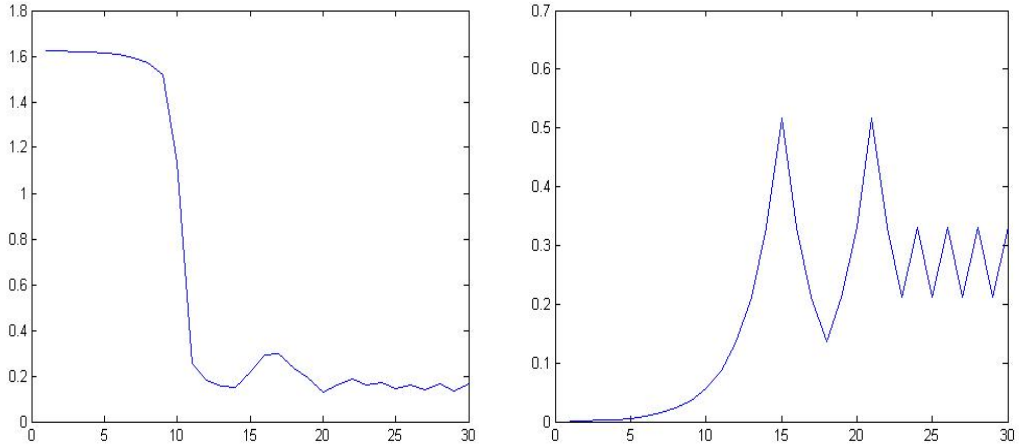
Όπως αναφέρθηκε και στα προηγούμενα (και φαίνεται στο σχήμα 14) έγιναν 500 προσπάθειες προκειμένου να διαπιστωθεί ότι η συγκεκριμένη πολιτική είναι η καλύτερη για αυτό το σύνολο δεδομένων. Κάποια ενδεικτικά αποτελέσματα έπειτα από την τελευταία προσπάθεια (για τις 30 εποχές), όσον αφορά την πορεία του ρυθμού μάθησης και του σφάλματος είναι τα ακόλουθα :



ΣΧΗΜΑ 15 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 1 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ PIMA



ΣΧΗΜΑ 16 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.1 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ PIMA



ΣΧΗΜΑ 17 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ PIMA

Στη συνέχεια ακολουθούν τα αποτελέσματα από το κοινό κομμάτι εκπαίδευσης, τόσο από το απλό σύστημα, όσο και από το σύστημα ενισχυτικής μάθησης (βάσει της βέλτιστης πολιτικής).

	Απλό σύστημα		Σύστημα ενισχυτικής μάθησης	
Αρχικός ρυθμός μάθησης	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων
1	--	--	78,1	0.00009
0.1	77,7	0.00006	79,3	0.00009
0.01	76,9	0.00043	78	0.00010
0.001	69	0.00044	75,8	0.00084
0.0001	62,9	0.00562	77,2	0.00042
Μέσος όρος διασποράς		0.001638		0.000307

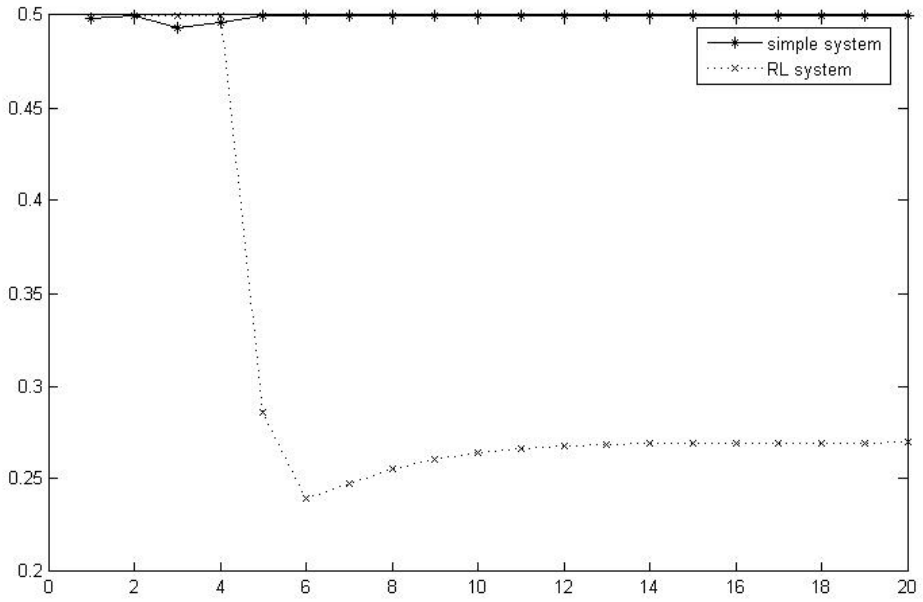
ΠΙΝΑΚΑΣ 5 : ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΙΤΥΧΙΑΣ ΠΡΟΤΥΠΩΝ ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ PIMA

Αξίζει να σημειωθούν τα ακόλουθα :

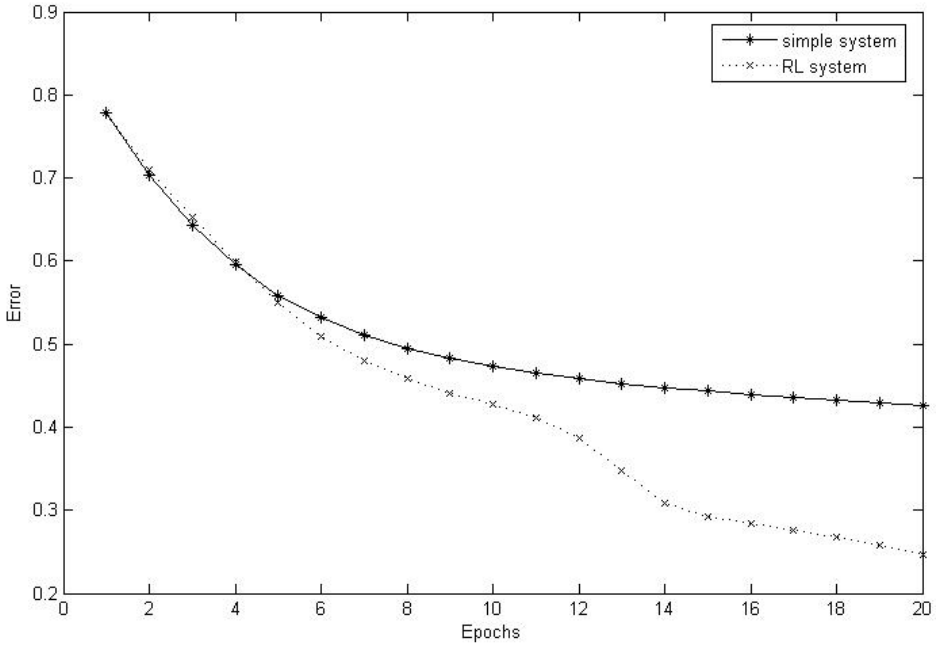
- το απλό σύστημα δεν εκπαιδεύεται καθόλου με ρυθμό μάθησης 1, σε αντίθεση με το δικό μας σύστημα το οποίο μειώνοντας το ρυθμό μάθησης καταφέρνει και εκπαιδεύεται κατάλληλα

- η διασπορά των τιμών των πειραμάτων που έγιναν με το σύστημα ενισχυτικής μάθησης ήταν μία τάξη μεγέθους μικρότερη από αυτή του απλού συστήματος γεγονός που δείχνει μεγαλύτερη σταθερότητα στη λειτουργία.

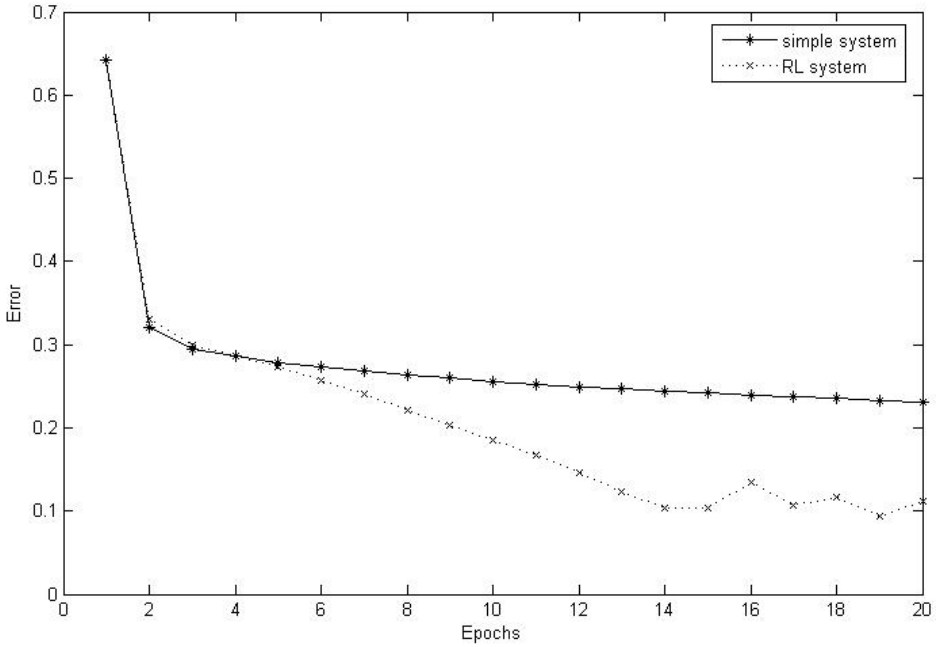
Μια εποπτική παρουσίαση των αποτελεσμάτων έχουμε από τα ακόλουθα σχήματα που δείχνουν τη μεταβολή του σφάλματος κατά την εκπαίδευση του συνόλου δεδομένων με τα δύο συστήματα για κάποιες αρχικές τιμές του ρυθμού μάθησης.



ΣΧΗΜΑ 18 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ RIMA ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 1



ΣΧΗΜΑ 19 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ ΡΙΜΑ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001



ΣΧΗΜΑ 20 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ ΡΙΜΑ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.0001

Από τα παραπάνω σχήματα γίνεται φανερό, πως το σύστημα ενισχυτικής μάθησης λειτουργεί καλύτερα από το απλό σύστημα μιας και μέσα στο μικρό διάστημα των 20 εποχών καταφέρνει και αφενός μειώνει το σφάλμα με γρηγορότερο ρυθμό, αφετέρου καταλήγει σε μικρότερη τιμή αυτού.

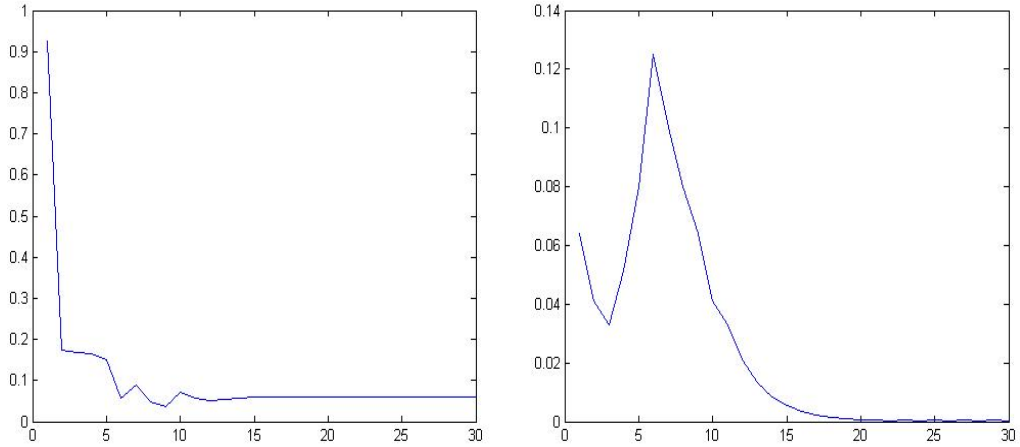
5.3.2 Αποτελέσματα από το σύνολο δεδομένων Breast cancer

Το σύνολο δεδομένων breast cancer αποτέλεσε ένα σύνολο δεδομένων το οποίο εκπαιδευόταν αρκετά καλά, ακόμα και με το απλό σύστημα. Οι πολιτικές που πήραμε (βάσει της (4.4)) φαίνονται στον ακόλουθο πίνακα :

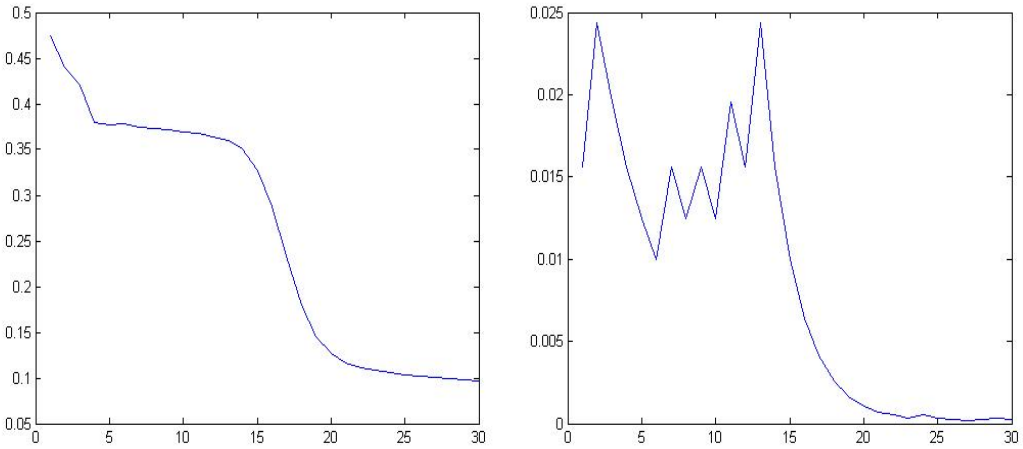
αρχικός ρυθμός μάθησης	επιλογή δράσης από κατάσταση 1	επιλογή δράσης από κατάσταση 2	Επιλογή δράσης από κατάσταση 3	επιλογή δράσης από κατάσταση 4	επιλογή δράσης από κατάσταση 5
1	4	3	1	3	1
0.1	4	1	4	4	4
0.01	3	3	1	1	4
0.001	2	1	4	4	4
0.0001	4	2	1	4	4

ΠΙΝΑΚΑΣ 6 : ΒΕΛΤΙΣΤΕΣ ΠΟΛΙΤΙΚΕΣ ΓΙΑ ΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ BREAST CANCER

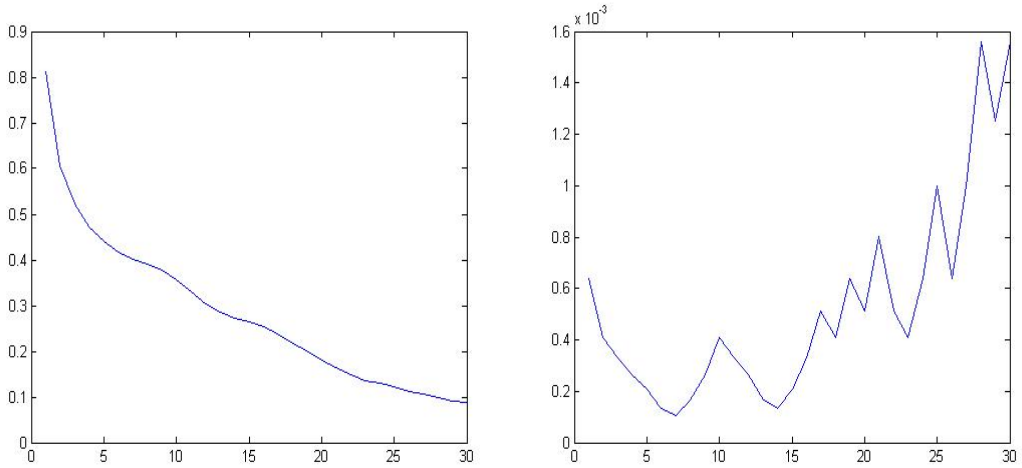
Κάποια ενδεικτικά αποτελέσματα για την πορεία του σφάλματος και του ρυθμού μάθησης καθώς περνάνε οι εποχές, για την τελευταία από τις 500 προσπάθειες είναι τα ακόλουθα :



ΣΧΗΜΑ 21 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.1 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ BREAST CANCER



ΣΧΗΜΑ 22 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.01 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ BREAST CANCER



ΣΧΗΜΑ 23 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ BREAST CANCER

Στη συνέχεια ακολουθούν τα αποτελέσματα από το κοινό κομμάτι εκπαίδευσης, τόσο από το απλό σύστημα, όσο και από το σύστημα ενισχυτικής μάθησης (βάσει της βέλτιστης πολιτικής).

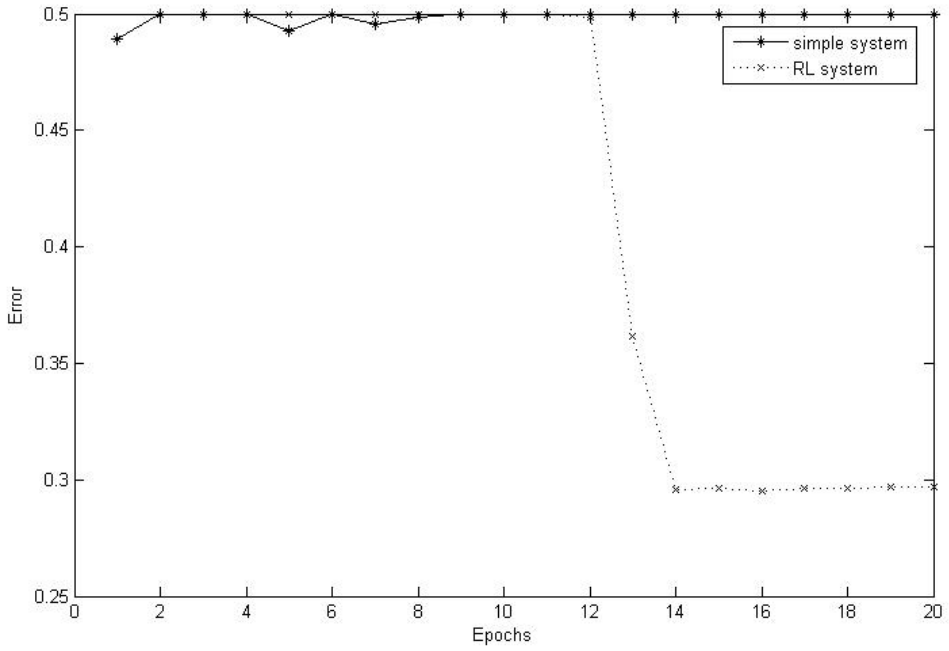
	Απλό σύστημα		Σύστημα ενισχυτικής μάθησης	
Αρχικός ρυθμός μάθησης	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων
1	--	--	98,6	0,000008
0.1	93,4	0,002046	97,7	0,000072
0.01	98,0	0,000042	97,1	0,000430
0.001	98,2	0,000008	97,3	0,000327
0.0001	97,4	0,000191	95,8	0,001079
Μέσος όρος διασποράς		0,000572		0,000383

ΠΙΝΑΚΑΣ 7 : ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΙΤΥΧΙΑΣ ΠΡΟΤΥΠΩΝ ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ BREAST CANCER

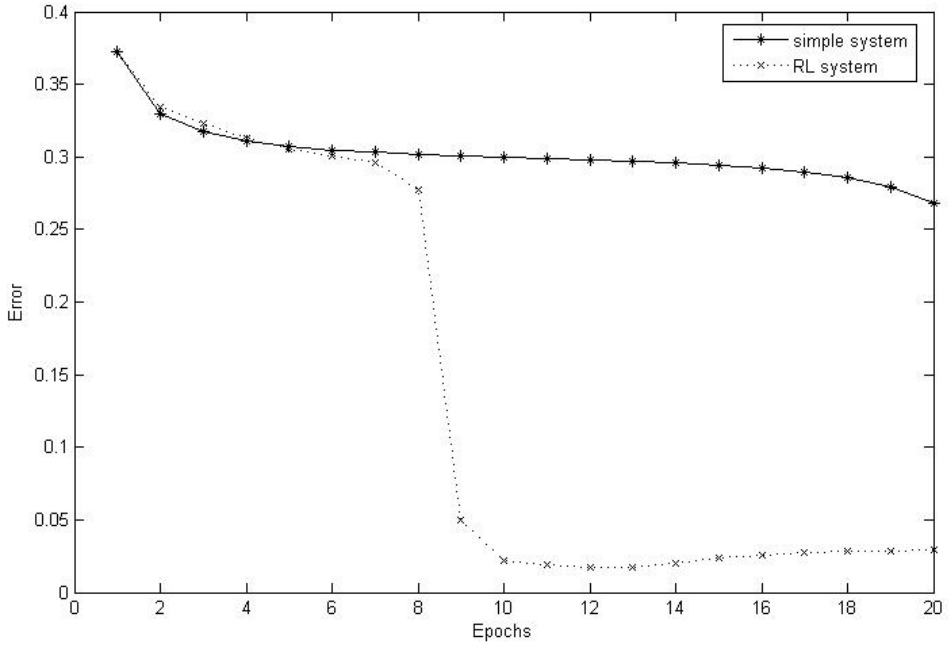
Αξίζει να σημειωθούν τα ακόλουθα :

- Όπως και στο σύνολο δεδομένων pima έτσι και εδώ, το απλό σύστημα δεν εκπαιδεύεται καθόλου με ρυθμό μάθησης 1, σε αντίθεση με το δικό μας σύστημα το οποίο μειώνοντας το ρυθμό μάθησης καταφέρνει και εκπαιδεύεται κατάλληλα
- η διασπορά των τιμών των πειραμάτων που έγιναν με το σύστημα ενισχυτικής μάθησης είναι μικρότερη από αυτή του απλού συστήματος γεγονός που δείχνει μεγαλύτερη σταθερότητα στη λειτουργία.
- Το γεγονός της σταθερότητας στην εφαρμογή ενισχύεται και από την παρατήρηση, ότι σε αρκετά από τα πειράματα στο απλό σύστημα θεωρήθηκαν αποτυχημένα, δηλαδή το δίκτυο δεν κατάφερε να εκπαιδευτεί. Ειδικά για τους μικρούς αρχικούς ρυθμούς μάθησης (0.0001 και 0.001) το ποσοστό των αποτυχημένων πειραμάτων ήταν τουλάχιστον διπλάσιο για το απλό σύστημα.

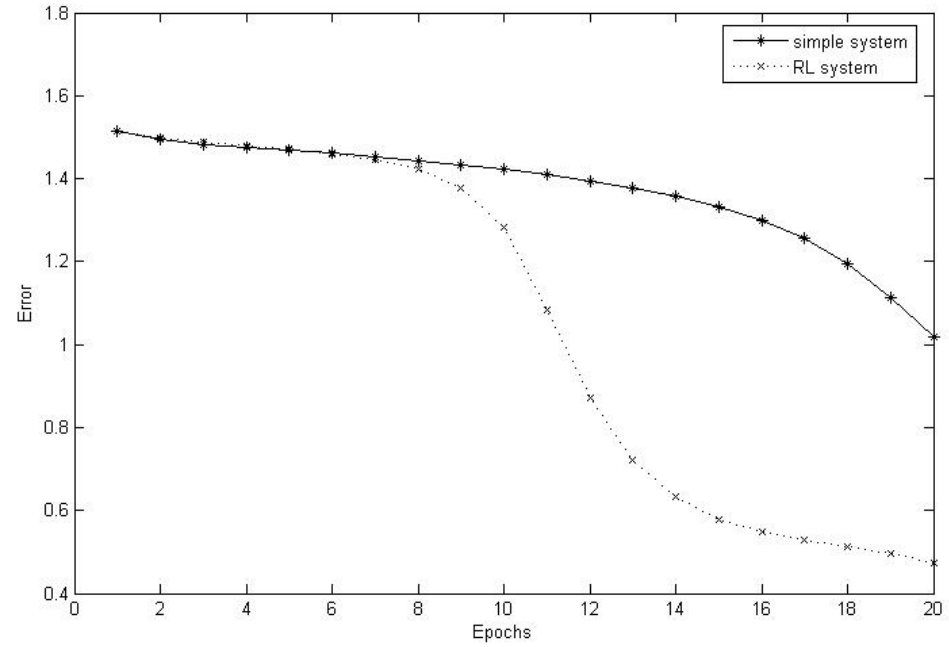
Μια εποπτική παρουσίαση των αποτελεσμάτων έχουμε από τα ακόλουθα σχήματα που δείχνουν τη μεταβολή του σφάλματος κατά την εκπαίδευση του συνόλου δεδομένων με τα δύο συστήματα για κάποιες αρχικές τιμές του ρυθμού μάθησης.



ΣΧΗΜΑ 24 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ BREAST CANCER ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 1



ΣΧΗΜΑ 25 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ BREAST CANCER ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001



ΣΧΗΜΑ 26 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ BREAST CANCER ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.0001

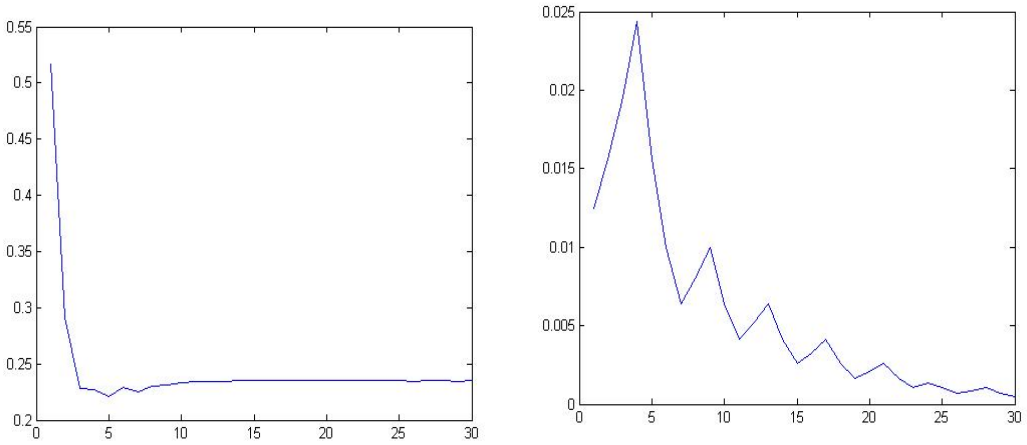
5.3.3 Αποτελέσματα από το σύνολο δεδομένων Ionosphere

Το σύνολο δεδομένων ionosphere ήταν ένα δύσκολο σύνολο δεδομένων μιας και για αρχικούς ρυθμούς μάθησης με σχετικά μεγάλες τιμές (1 και 0.1) δεν εκπαιδεύτηκε ούτε με το απλό σύστημα, ούτε με το σύστημα ενισχυτικής μάθησης. Η αιτία είναι πως το δίκτυο παγιδεύεται σε τοπικά ελάχιστα, παρότι η ιδέα του συστήματος που φτιάξαμε εφαρμόζεται και εδώ : ο ρυθμός μάθησης μειώνεται και από κάποια εποχή και μετά ξεκινά μία "ταλάντωση" μεταξύ κάποιας τιμής. Οι πολιτικές που πήραμε (βάσει της (4.4)ι της (4.4)) φαίνονται στον ακόλουθο πίνακα :

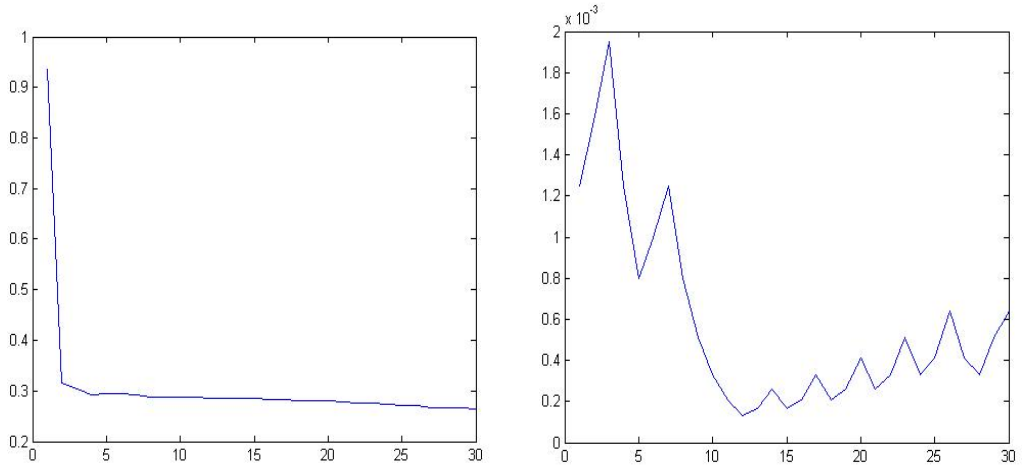
αρχικός ρυθμός μάθησης	επιλογή δράσης από κατάσταση 1	επιλογή δράσης από κατάσταση 2	Επιλογή δράσης από κατάσταση 3	επιλογή δράσης από κατάσταση 4	επιλογή δράσης από κατάσταση 5
0.01	4	4	2	2	4
0.001	2	1	3	3	4
0.0001	4	1	2	2	4

ΠΙΝΑΚΑΣ 8 : ΒΕΛΤΙΣΤΕΣ ΠΟΛΙΤΙΚΕΣ ΓΙΑ ΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ IONOSPHERE

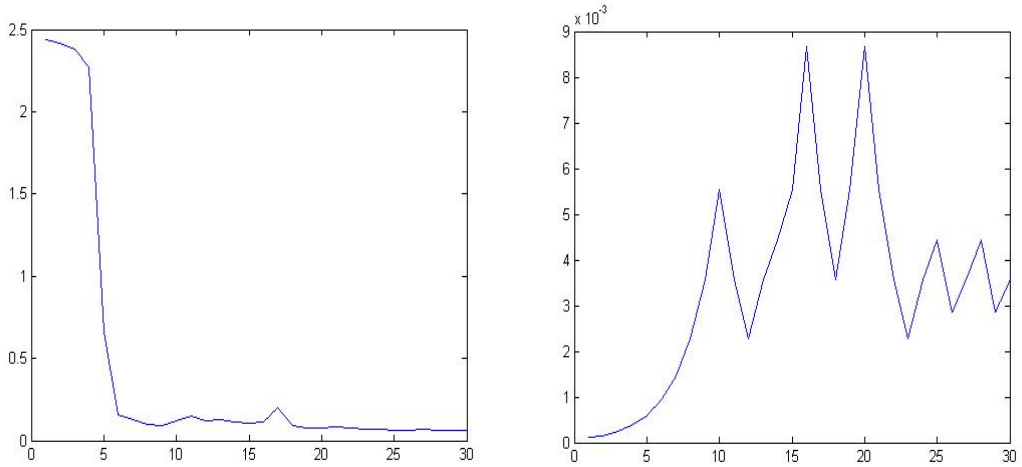
Κάποια ενδεικτικά αποτελέσματα για την πορεία του σφάλματος και του ρυθμού μάθησης καθώς περνάνε οι εποχές, για την τελευταία από τις 500 προσπάθειες είναι τα ακόλουθα :



ΣΧΗΜΑ 27 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.01 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ IONOSPHERE



ΣΧΗΜΑ 28 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ IONOSPHERE



ΣΧΗΜΑ 29 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΡΥΘΜΟΥ ΜΑΘΗΣΗΣ ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.0001 ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ IONOSPHERE

Στη συνέχεια ακολουθούν τα αποτελέσματα από το κοινό κομμάτι εκπαίδευσης, τόσο από το απλό σύστημα, όσο και από το σύστημα ενισχυτικής μάθησης (βάσει της βέλτιστης πολιτικής).

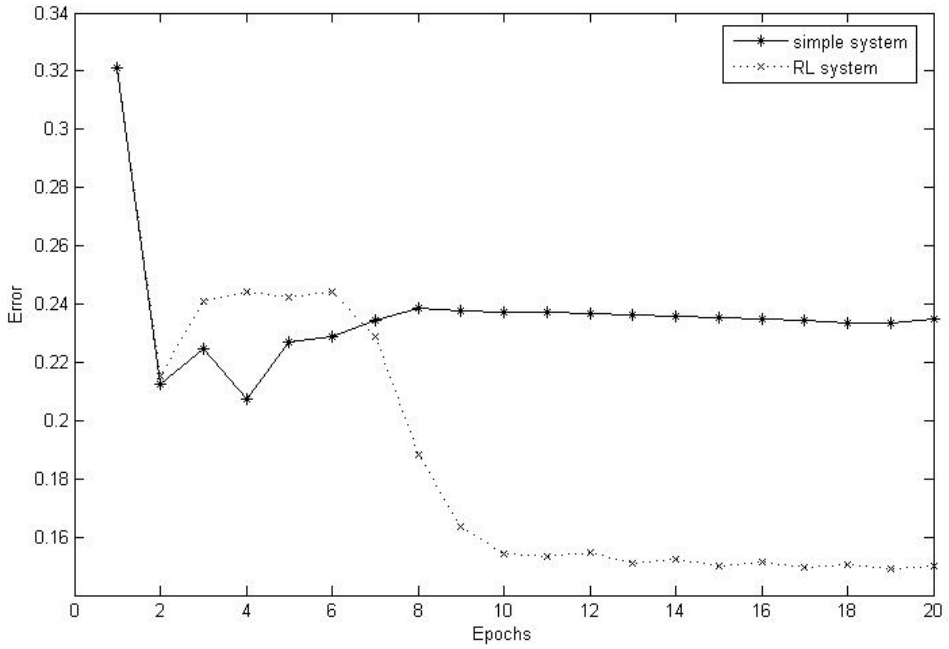
	Απλό σύστημα		Σύστημα ενισχυτικής μάθησης	
Αρχικός ρυθμός μάθησης	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων	Επιτυχημένα πρότυπα (%)	Διασπορά τιμών πειραμάτων
0.01	86,2	0,0070	92,5	0,0013
0.001	88,7	0,0130	90,6	0,0022
0.0001	85,4	0,0045	93,6	0,0009
Μέσος όρος διασποράς		0,0082		0,0015

ΠΙΝΑΚΑΣ 9 : ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΠΙΤΥΧΙΑΣ ΠΡΟΤΥΠΩΝ ΣΤΟ ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ IONOSPHERE

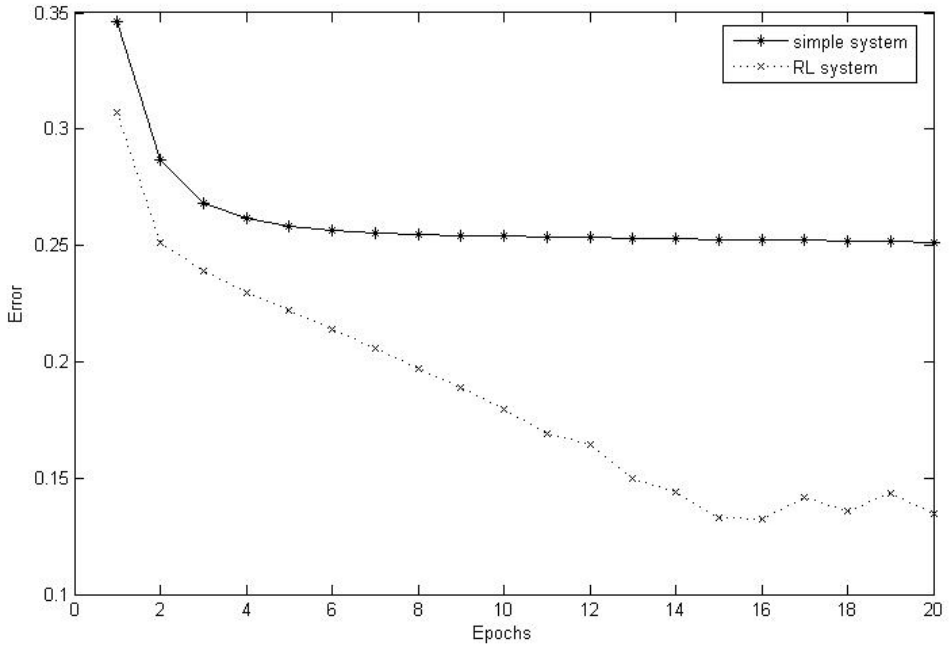
Αξίζει να σημειωθεί ότι :

- Όπως και στα άλλα δύο σύνολα δεδομένων η διασπορά των τιμών των πειραμάτων που έγιναν με το σύστημα ενισχυτικής μάθησης είναι μικρότερη από αυτή του απλού συστήματος γεγονός που δείχνει μεγαλύτερη σταθερότητα στη λειτουργία.

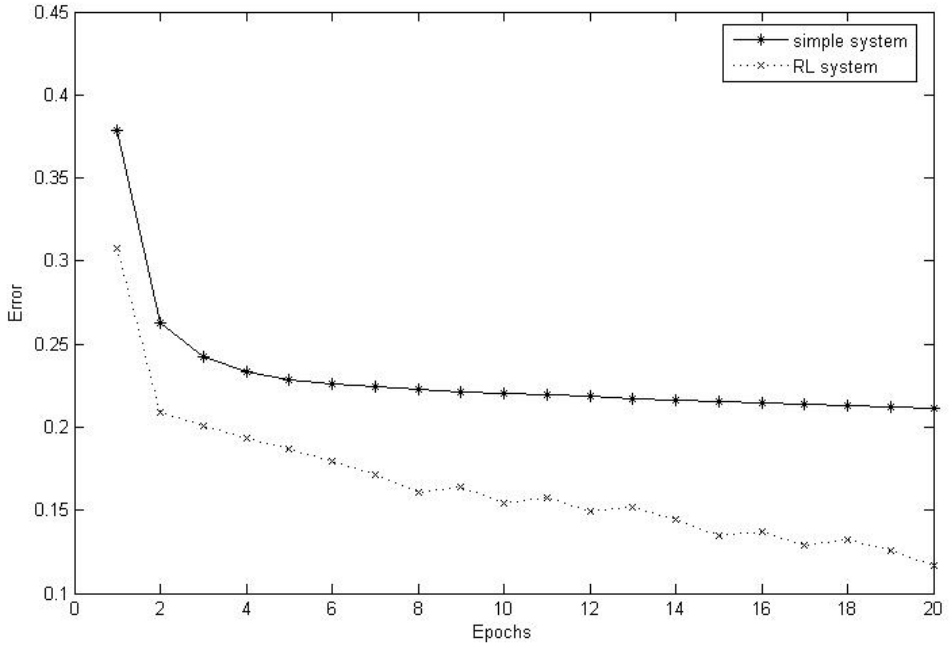
Μια εποπτική παρουσίαση των αποτελεσμάτων έχουμε από τα ακόλουθα σχήματα που δείχνουν τη μεταβολή του σφάλματος κατά την εκπαίδευση του συνόλου δεδομένων με τα δύο συστήματα για κάποιες αρχικές τιμές του ρυθμού μάθησης.



ΣΧΗΜΑ 30 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ IONOSPHERE ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.01



ΣΧΗΜΑ 31 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ IONOSPHERE ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.001



ΣΧΗΜΑ 32 : ΜΕΤΑΒΟΛΗ ΣΦΑΛΜΑΤΟΣ ΚΑΤΑ ΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ IONOSPHERE ΜΕ ΑΡΧΙΚΟ ΡΥΘΜΟ ΜΑΘΗΣΗΣ 0.0001

Στα παραπάνω σχήματα φαίνεται η υπεροχή του συστήματος ενισχυτικής μάθησης όσον αφορά τη μείωση του σφάλματος. Ιδιαίτερα αισθητή είναι η διαφορά για αρχικούς ρυθμούς μάθησης 0.0001 και 0.001 ενώ για αρχικό ρυθμό μάθησης 0.01, μία αρχική απόκλιση που φαίνεται να δημιουργείται, εύκολα εξομαλύνεται και τελικά το σφάλμα μειώνεται πολύ περισσότερο απ’ ότι στο απλό σύστημα.

6 *Συμπεράσματα – Το μέλλον*

6.1 Σύνοψη και συμπεράσματα από την εφαρμογή της μεθόδου

Όπως αναφέρθηκε στην εισαγωγή, είναι γεγονός πως η ενισχυτική μάθηση δεν έχει εφαρμοστεί ευρέως έως τώρα στους τομείς της μάθησης μηχανής και της αναγνώρισης προτύπων. Στην παρούσα διπλωματική εργασία υποστηρίζεται, και αποδεικνύεται πειραματικά, πως οι τεχνικές της ενισχυτικής μάθησης με κατάλληλη προσαρμογή και επιλογή στόχου, μπορούν να βρουν πρόσφορο έδαφος για εφαρμογή στα παραπάνω πεδία υπολογιστικής νοημοσύνης. Μία τέτοια τεχνική είναι και η μάθηση Q, που μέσα από την ερευνητική μας μελέτη, φάνηκε πως μπορεί να βελτιώσει τις διαδικασίες στην ταξινόμηση προτύπων.

Η μεθοδολογία μεταβολής του ρυθμού μάθησης κατά την εκπαίδευση εφαρμόστηκε και παρουσίασε γενικά θετικά αποτελέσματα και στα τρία σύνολα δεδομένων. Η εξαίρεση της εφαρμογής της μεθόδου ήταν για δύο περιπτώσεις του ρυθμού μάθησης (1, 0.1) στο σύνολο δεδομένων ionosphere, στο οποίο δυστυχώς η παγίδευση σε τοπικά ελάχιστα με τον αλγόριθμο της κατάβασης κλίσης, δεν μας έδωσε ικανοποιητικά αποτελέσματα, παρότι η μέθοδος λειτούργησε σωστά, μεταβάλλοντας το ρυθμό μάθησης , ώστε να μπορέσει το νευρωνικό δίκτυο να εκπαιδευτεί κατάλληλα. Στις υπόλοιπες περιπτώσεις η μέθοδος εκτελέστηκε χωρίς πρόβλημα και η συμπεριφορά του δικτύου, καθώς και τα αποτελέσματα, ήταν σε συμφωνία με το

αναμενόμενο. Σε περιπτώσεις που το απλό σύστημα, δεν κατάφερνε να εκπαιδευτεί (αρχικός ρυθμός μάθησης 1), το σύστημα ενισχυτικής μάθησης κατάφερε, μειώνοντας αρχικά το ρυθμό μάθησης και κατόπιν μεταβάλλοντάς τον ανάλογα με την τιμή του σφάλματος, να καταλήξει σε μία καλή κατηγοριοποίηση των προτύπων. Στη συντριπτική πλειοψηφία των περιπτώσεων δε, το τελικό σφάλμα ήταν αρκετά μικρότερο από εκείνο που καταλήγει το απλό σύστημα μέσα σε λίγες μόνο εποχές, παράμετρος επίσης σημαντική όσον αφορά την ταχύτητα με την οποία εκπαιδεύεται το νευρωνικό δίκτυο.

Άλλη μία παρατήρηση που σχετίζεται με τη μέθοδο που αναπτύχθηκε και την εφαρμογή της, έχει να κάνει με τις τιμές των διαφόρων παραμέτρων του κώδικα (και έχουν να κάνουν π.χ. με το νευρωνικό δίκτυο ή με τη μάθηση Q) που πρέπει να χρησιμοποιηθούν με στόχο το καλύτερο αποτέλεσμα όσον αφορά το δίπτυχο σωστή εκπαίδευση και ταχύτητα. Πέραν από κάποιες γενικές παρατηρήσεις που αναφέρθηκαν στο κεφάλαιο 4 και είχαν να κάνουν με τη γενική μορφή και διάρθρωση του ενισχυτικού σήματος, του χώρου καταστάσεων και των δράσεων, και κυρίως είχαν να κάνουν με την κλιμάκωση των αλλαγών αυτών των παραμέτρων, ένα συγκεκριμένο συμπέρασμα για τις αριθμητικές τιμές των υπολοίπων παραμέτρων δε μπορεί να βγει. Ο αριθμός των εποχών, οι κρυμμένοι νευρώνες, τα πρότυπα που θα δοθούν προς εκπαίδευση από το σύστημα ενισχυτικής μάθησης, η τάξη μεγέθους των ενισχυτικών σημάτων αλλά και παράγοντες της εξίσωσης μάθησης Q όπως ο ρυθμός μάθησης του πράκτορα ή ο παράγοντας έκπτωσης δε μπορούν να καθοριστούν μονοσήμαντα, αλλά απαιτούν διαφορετική ρύθμιση και ίσως πειραματισμό για τα διάφορα δεδομένα που έχουμε να αντιμετωπίσουμε. Στην παρούσα διπλωματική εργασία και στα λογικά πλαίσια της γενίκευσης, επιλέξαμε τιμές οι οποίες να είναι σύμφωνες με τις απαιτήσεις που θέσαμε για «γρήγορη μάθηση» αλλά και μπορούσαν να λειτουργήσουν σωστά με διάφορα σύνολα δεδομένων.

6.2 Μελλοντικές επεκτάσεις

Από την εφαρμογή της μεθόδου που αναπτύχθηκε στα προηγούμενα, αποδείχθηκε πως η ενισχυτική μάθηση μπορεί να βρει εφαρμογή στον τομέα της μάθησης μηχανής και της αναγνώρισης προτύπων, όπως άλλωστε ήταν και ο στόχος της διπλωματικής εργασίας. Ως εφαρμογή της μάθησης Q και αντικείμενο της μελέτης μας επιλέξαμε το

ρυθμό μάθησης του νευρωνικού δικτύου, σαν μία από τις σημαντικότερες παραμέτρους μεταβολής με καθοριστικό ρόλο στη διαδικασία της εκπαίδευσης.

Φυσικά αυτό ήταν μόνο το πρώτο βήμα. Σαν "φυσική συνέχεια" αυτής της διπλωματικής εργασίας, έρχονται και επιπλέον ζητήματα που έχουν να κάνουν με την επέκταση της μεθόδου της μάθησης Q σε άλλες παραμέτρους της διαδικασίας. Τέτοιες παράμετροι μπορούν να είναι ο αριθμός των εποχών, ο αριθμός των κρυμμένων νευρώνων, ο όρος ορμής και άλλες (δυνητικά κάθε όρος που εισάγεται στη διαδικασία της εκπαίδευσης), εκτιμώντας ότι τα αποτελέσματα θα είναι παρόμοια ή και καλύτερα από την εφαρμογή της μάθησης Q για το ρυθμό μάθησης.

Τελικός στόχος όλης της παραπάνω διαδικασίας είναι η κατασκευή ενός συστήματος το οποίο θα μας δίνει τις προτεινόμενες-βέλτιστες τιμές για τις διάφορες παραμέτρους του δικτύου βάσει του οποίου ταξινομούμε πρότυπα, σύστημα το οποίο φυσικά, θα εξοικονομεί πολύτιμο χρόνο από τον πειραματισμό για την εύρεση των διαφόρων τιμών αλλά και θα καταλήγει σε καλύτερα αποτελέσματα όσον αφορά στην εκπαίδευση του δικτύου. Δε μένει παρά μόνο να δούμε το σύστημα αυτό στην πράξη...

Παράρτημα

Κώδικας εκπαίδευσης νευρωνικού δικτύου με στόχο την εύρεση της βέλτιστης πολιτικής

```
function y = RLsys (dataset,dataset_test,lrate,X,Trials)

% MAIN V9 - RL SYSTEM WITH ADAPTION LEARNING RATE

k=0;

% Loading data
load (dataset);

Constants
if nargin<5
    Trials=500;
end
if nargin<4
    X=5;
end
if nargin<3
    initLR=1;
else
    initLR=lrate;
```

```
end

Epochs=30;
alpha=0.8;                % Agent's Learning Rate
gamma=0.65;               % Discount Factor gamma
sigma=1.25;               % LR Deviation
exploration=0.9;          % Exploration factor (initial value)
all_actions=4;            % 4 actions for LR : increase high, increase low, decrease low, decrease high
all_states=5;             % 5 states : error decreasing much, error decreasing, error stable, error increasing, error increasing much
goal=0;                   % error = zero


% Prepare Data
input=DataIn;
dim1=size(input,1);     % arithmos protypwn
dim2=size(input,2);     % xaraktiristika protypwn

input1=zeros(dim2,2);

minall=min(DataIn);
maxall=max(DataIn);

for k=1:1:dim2
    input1(k,1)=minall(k);
    input1(k,2)=maxall(k);
end %for min-max timwn protypwn
Out=zeros(size(OutClass));

% Set exploration factor which will decay through time
exf=exploration;
```

```
% initialize Q(states,actions) with an arbitrary value
Q=ones(all_states,all_actions);
for i=1:all_states
    for j=1:all_actions
        Q(i,j)=Q(i,j)*rand;
    end
end
[v1,P]=max(Q');
%*************************************************************
% Begin Trials
%*************************************************************
for n=1:Trials
    if (mod(n, 50)==0)
        Tr=n
        BestPolicy = P(n-1,:)
        Q
    end
    %kataskevi neurwnikou
    net    =    newff(input1,[7    size(OutClass,2)],{'tansig'
'tansig'},'learngdm');
    net.trainFcn='traingdx';
    net.trainParam.epochs = 1;
    net.trainParam.goal = 0;
    net.trainParam.show = NaN;

    %Initialize parameters
    lr=initLR;
    net=init(net);
    t=1; %observe state
    ActionOrder=randperm(all_actions);
    a=ActionOrder(1);
```

```
    % Start Environment
    while (t <= Epochs) %&& (Error>0)       % batch
        % Observe State

        net.trainParam.lr=lr;
        lr;
        % Select action and compute Q(n-1)
        for i=1:1:40      % gia ola ta protypa

            [net,tr,Y,E]                                          =
train(net,input(i,:)',OutClass(i,:)');

            Out(i,:)=Y';

            ER(i)=mse(E);

        end %if protypwn


        ER1=mean(ER(:));
        Error(t)=ER1;
        if t==1
            DE=Error(t);
            percent=1;
            s=3;
        else
            DE=Error(t) - Error(t - 1);
            percent=100*abs(DE)/Error(t-1);
        end %if sfalmatos

        %YPOSYSTHMA EKPAIDEUSHS RYTHMOU MATHISIS
```

```
        if DE==0            % Error stable
            REINF=0;
            s_new=3;
        elseif (DE<0 & percent>X) % Error reduced much
            s_new=1;
            REINF=30*abs(DE);
        elseif DE<0               % Error reduced
            s_new=2;
            REINF=20*abs(DE);
        elseif (DE>0 & percent>X) % Error increased much
            s_new=5;
            REINF=-30*abs(DE);
        elseif DE>0               % Error increased
            s_new=4;
            REINF=-20*abs(DE);
        end %if

        %   update Q

        Q(s,a)=Q(s,a)  +  alpha*(REINF  +  gamma*max(Q(s_new,:))-
Q(s,a));



        % Determine best action
        %*********************   Exploitation   -   Exploration
***************
        if rand<=exf
            BestAction=randperm(all_actions);
        else
            [MaxVal, BestAction] = max(Q(s,:));
        end
```

```
    cur_action=BestAction(1);
    pre_lr=net.trainParam.lr;
    lr=0;


    % Adapt learning rate
    switch (cur_action)
        case 1, lr=pre_lr*sigma^2;  % increase lr much
        case 2, lr=pre_lr*sigma;  % increase lr little
        case 3, lr=pre_lr/sigma;    % decrease lr little
        case 4, lr=pre_lr/(sigma^2);  % decrease lr much
    end %switch
    % if lr>1 lr=1; end
    LRATE(t)=lr;
    a=cur_action;
    t=t+1;
    s=s_new;

end %  of States

TrialError(n) = 100*(Error(1)-Error(t-1))/Error(1);
TrialLR(n) = mean(LRATE);
Q;
% Determine Best Policy P(n)
[QValue, P(n,:)] = max(Q');

% Reduce alpha
alpha=alpha - (alpha/Trials);

% Reduce exploration factor
if exf>0.01
    exf = exf-(3*exploration/Trials);
```

```
    end

end %  of Trials
Q
BestPolicy = P(Trials,:)
disp('1 (blue)=increase much, 2 (green)= increase little, 3
(red)= decrease little, 4 (cyan)= decrease much')
lr
exf;
% Error(iter)
figure;plot(TrialError);
figure;plot(Error);
figure;plot(LRATE);
figure;plot(Q);
save('LR1');
clear all;
close all;
clc;
```

Κώδικας απλού συστήματος ταξινόμησης (χωρίς προσαρμογή ρυθμού μάθησης)

```
function [perfo] = simplesys (dataset,dataset_test,initLR)

% MAIN V8 TEST - SYSTEM WITHOUT ADAPTIVE LEARNING RATE

Epochs=20;
mm=0;


% Loading data
```

```
load (dataset);

%2
% DataIn
input=DataIn;
dim1=size(input,1);     % arithmos protypwn
dim2=size(input,2);     % xaraktiristika protypwn
trainpatterns=dim1;

input1=zeros(dim2,2);

minall=min(DataIn);
maxall=max(DataIn);

for k=1:1:dim2
    input1(k,1)=minall(k);
    input1(k,2)=maxall(k);
end %for min-max timwn protypwn

Out=zeros(size(OutClass));
ER=zeros(size(Out));
Error=[0 0];

%kataskevi neurwnikou
net     =     newff(input1,[7     size(OutClass,2)],{'tansig'
'tansig'},'learngdm');
net.trainFcn='traingdx';
net.trainParam.epochs = 1;
net.trainParam.goal = 0;
net.trainParam.show = NaN;
net.trainParam.lr=initLR;
net=init(net);
```

```
for j=1:1:Epochs       % batch

    for i=1:1:trainpatterns  % protypa

        [net,tr,Y,E] = train(net,input(i,:)',OutClass(i,:)');

        Out(i,:)=Y';

        ER(i)=mse(E);

    end %if protypwn


    ER1=mean(ER(i));
    Error(j)=ER1;

end %if batch

Error;
figure;plot(Error);

% TESTING DATASET
load(dataset_test);
input=DataIn;
dim1=size(input,1);     % arithmos protypwn
dim2=size(input,2);     % xaraktiristika protypwn

for i=1:1:dim1
```

```
    %                                          [net,tr,Y,E]        =
train(net,input(i,:)',OutClass(i,:)');

    [Y,Pf,Af,E,perf]=sim(net,input(i,:)');

    Out(i,:)=Y';

    ERRR=OutClass(i,:)-Out(i,:);



    R1=max(Out(i,:));
    index1=find(Out(i,:)==R1);
    R2=max(OutClass(i,:));
    index2=find(OutClass(i,:)==R2);

    if index1==index2 mm=mm+1; end

    ER(i)=mse(ERRR);

end %if protypwn

ER1=mean(ER(i));

perfo=mm/dim1
save (sprintf('performNOLR%d.mat',repeat));

% clear all;
clear net;
close all;
clc;
```

Κώδικας συστήματος ταξινόμησης με χρήση της βέλτιστης πολιτικής μεταβολής του ρυθμού μάθησης

```
function [perfo] = testRLsys(environment)

% ======================================================
% ======================================================
% loading environment with the determined best policy
load(environment);

load(dataset);
input=DataIn;
dim1=size(input,1);     % arithmos protypwn
dim2=size(input,2);     % xaraktiristika protypwn
mm=0;
trainpatterns=dim1;
X=5;
Epochs=20;
perform=[0];

net=init(net);
lr=initLR;

%Initialize parameters
t=1; %observe state
ActionOrder=randperm(all_actions);
a=ActionOrder(1);

% Start Environment
while (t <= Epochs) %&& (Error>0)       % batch
```

```
% Observe State

net.trainParam.lr=lr;
% Select action and compute Q(n-1)
for i=1:1:trainpatterns      % gia ola ta protypa

    [net,tr,Y,E] = train(net,input(i,:)',OutClass(i,:)');

    Out(i,:)=Y';

    ER(i)=mse(E);

end %if protypwn

ER1=mean(ER(:));
Error(t)=ER1;
if t==1
    DE=Error(t);
    percent=1;
else
    DE=Error(t) - Error(t - 1);
    percent=100*abs(DE)/Error(t-1);
end %if sfalmatos


% Define new state
if DE==0            % Error stable
    s_new=3;
elseif (DE<0 & percent>X) % Error reduced much
    s_new=1;
elseif DE<0               % Error reduced
    s_new=2;
```

```
    elseif (DE>0 & percent>X) % Error increased much
        s_new=5;
    elseif DE>0               % Error increased
        s_new=4;
    end %if

    % Determine next action according to best policy
    cur_action=BestPolicy(s_new);

    pre_lr=net.trainParam.lr;

    % Adapt learning rate
    switch (cur_action)
        case 1, lr=pre_lr*(sigma^2);  % increase lr much
        case 2, lr=pre_lr*sigma;  % increase lr little
        case 3, lr=pre_lr/sigma;    % decrease lr little
        case 4, lr=pre_lr/(sigma^2);  % decrease lr much
    end %switch
    % if lr>1 lr=1; end
    LRATE(t)=lr;
    STATE(t)=s_new;
    a=cur_action;
    t=t+1;
    s=s_new;
%     net.trainParam.lr=lr;

end %  of States

% TESTING DATASET
load(dataset_test);
input=DataIn;
dim1=size(input,1);     % arithmos protypwn
```

```
dim2=size(input,2);     % xaraktiristika protypwn
Out=zeros(size(OutClass));

for i=1:1:dim1

    [Y,Pf,Af,E,perf]=sim(net,input(i,:)');

    Out(i,:)=Y';

    ERRR=OutClass(i,:)-Out(i,:);

    R1=max(Out(i,:));
    index1=find(Out(i,:)==R1);
    R2=max(OutClass(i,:));
    index2=find(OutClass(i,:)==R2);

    if index1==index2 mm=mm+1; end

    ER(i)=mse(ERRR);

end %if protypwn

figure;plot(ER);

perfo=mm/dim1

clear net;
close all;
clc;
```

Βιβλιογραφία

[1] Σ. Τζαφέστας, Υπολογιστική Νοημοσύνη Τόμος Α : Μεθοδολογίες, 2002

[2] Βλαχάβας Ι., Κεφάλας Π., Βασιλειάδης Ν., Κόκκορας Φ., Σακελλαρίου Η., Τεχνητή Νοημοσύνη, Β' έκδοση, 2005

[3] S. Haykin, Neural Networks – A Comprehensive Foundation, Prentice-Hall, 1999

[4] R. S. Sutton, A. G. Barto, Reinforcement Learning : An introduction, The MIT Press, 2005

[5] J. D. Cowan, G. Tesauro, and J. Alspector, Advances in Neural Information Processing Systems(Convergence of indirect adaptive asynchronous value iteration algorithms), volume 6, pages 695-702, 1994

[6] D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Advances in Neural Information Processing Systems(Temporal difference learning in continuous time and space) 8, MIT Press, 1996

[7] L. Kaebling, M. L. Littman, A. W. Moore, Reinforcement Learning : A survey, Journal of Artificial Intelligence Research 4 pp 237-285, 1996

[8] C. J. C. H. Watkins and P. Dayan, Machine Learning, 8 (3):279-292, 292

[9] N. Jan van Eck, M. van Wezel, Reinforcement Learning and its Application to Othello, Elsevier Science, 2004

[10] N. J. Nilsson, Introduction to Machine Learning, 1996

[11] T. Mitchell, Machine Learning, McGraw-Hill, 1997.

[12] M. Herrmann, R. Der, Efficient Q-learning by division of labor, 1998

[13] R. Allard, J. Faubert, Neural Networks : Different problems require different learning rate adaptive methods, 2003

[14] M. Riedmiller, H. Braun, A direct adaptive method for faster backpropagation learning : The RPROP Algorithm, 2004

[15] Z. Zainuddin, N. Mahat, Y. Abu Hassan, Improving the convergence of the backpropagation algorithm using local adaptive techniques, Transactions of Engineering, Computing and Technology Volume 1, 2004

[16] D.Michie, D.J. Spiegelhalter, C.C. Taylor, Machine Learning, Neural and Statistical Classification, 1994

[17] G. Orr, N. Schraudolph, F. Cummins, Neural Networks, 1999

[18] J. Mertz and P. M. Murphy, UCI Machine Learning Repository [Online] http://www.ics.uci.edu/~mlearn/MLSummary.html