



Πανεπιστήμιο Πειραιώς
Τμήμα Ψηφιακών Συστημάτων
Π.Μ.Σ. " Ψηφιακές Επικοινωνίες & Δίκτυα "

Διπλωματική Εργασία

Μηχανική μάθηση σε ασύρματα δίκτυα

Γιώργος Κούρσος
ΜΨΕ 1803

Επιβλέπων
Αγγελική Αλεξίου
Καθηγήτρια

Πειραιάς , Ιούνιος 2021

Περίληψη

Η παρούσα εργασία εξετάζει τεχνικές μηχανικής μάθησης με εφαρμογή σε ασύρματα δίκτυα. Σκοπός της παρούσας εργασίας είναι να παρουσιάσει τις βασικές τεχνικές για μηχανική μάθηση, καθώς και το πώς μπορούν να εφαρμοστούν σε ασύρματες επικοινωνίες. Η μηχανική μάθηση είναι ένα νέο ανερχόμενο επιστημονικό πεδίο με πολλές εφαρμογές, ενώ η ικανότητα της να προβλέπει και να λύνει όλα και πιο σύνθετα προβλήματα την έχουν εντάξει σχεδόν σε κάθε επιστημονικό πεδίο τεχνολογικό ή μη. Από την άλλη, οι ασύρματες επικοινωνίες αποτελούν πλέον αναπόσπαστο κομμάτι της καθημερινότητας μας έχοντας ως αποτέλεσμα οι κινητές συσκευές αλλά και οι απαιτήσεις των χρηστών να αυξάνονται συνεχώς. Τα ασύρματα δίκτυα νέας γενιάς αναμένεται να υποστηρίξουν εξαιρετικά υψηλούς ρυθμούς δεδομένων αλλά και νέες εφαρμογές οι οποίες θα έχουν υψηλές απαιτήσεις (QOS-quality of service) και θα φέρουν νέες προκλήσεις στην ασύρματη τεχνολογία. Η δυναμική ανάθεση πόρων και η συνεχόμενη αλλαγή του ασύρματου καναλιού ήταν ανέκαθεν ένα πεδίο έρευνας. Η πρόκληση λοιπόν είναι να δημιουργηθεί ένα έξυπνο δίκτυο που θα μαθαίνει και θα προσαρμόζεται στα δεδομένα αλλά και θα παίρνει αποφάσεις από μόνο του σκοπεύοντας στο καλύτερο QOS του τελικού χρήστη αλλά και στην καλύτερη αξιοποίηση των πόρων του δικτύου. Η τεχνητή νοημοσύνη και πιο συγκεκριμένα η μηχανική μάθηση είναι οι καταλληλότερες για την δημιουργία τέτοιων δικτύων, προσφέροντας την έξυπνάδα που χρειάζονται με ίσως και άριστα αποτελέσματα. Είναι λοιπόν αναπόφευκτο ότι τα δίκτυα επόμενης γενιάς θα υποβοηθούνται από αυτές τις τεχνολογίες, ενώ όσο οι γνώσεις μας αυξάνονται καλύτερες εφαρμογές θα δημιουργούνται.

Στο πρώτο κεφάλαιο γίνεται μια εισαγωγή στην μηχανική μάθηση, αναφερόμαστε στα τρία είδη της, όπου αναλύονται οι βασικές τεχνικές, οι αλγόριθμοι που χρησιμοποιούνται, τα πλεονεκτήματα αλλά και τα μειονεκτήματά τους. Στο δεύτερο κεφάλαιο αναφερόμαστε σε υπάρχοντα συστήματα ασύρματων επικοινωνιών δίνοντας τα βασικά χαρακτηριστικά τους. Για κάθε σύστημα παρουσιάζεται μια μέθοδος μηχανικής μάθησης η οποία προσαρμόζεται στο εκάστοτε σύστημα ενώ παρουσιάζονται πειραματικά αποτελέσματα που έχουν γίνει συγκρίνοντας την απόδοση τους με τις ως τώρα παραδοσιακές τεχνικές. Στο τρίτο και τελευταίο παρουσιάζονται αποτελέσματα από προσομοιώσεις που έγιναν σε python για δυο διαφορετικά συστήματα δείχνοντας την απόδοση τους.

Abstract

The scope of this thesis is to examine machine learning techniques applied to wireless networks. The purpose of this paper is to present the basic techniques for machine learning, as well as how they can be applied in wireless communications. Machine learning is a new emerging scientific field with many applications, while it has the ability to predict and solve more and more complex problems that they have integrated into almost every scientific field, technological or not. On the other hand, wireless communications are now an integral part of our daily lives, resulting in mobile devices and the demands of users are constantly increasing. The new generation wireless networks are expected to support extremely high data rates but also new applications which will have high requirements (QoS-quality of service) and will bring new challenges in wireless technology. Dynamic resource allocation and the constant change of the wireless channel has always been a field of research. So the challenge is to create a smart network that will learn and adapt to the data but also will make decisions on its own with the aim of the best QoS of the end user but also the best use of network resources. Artificial intelligence and more specifically machine learning are the most suitable for creating such networks, offering the intelligence they need with perhaps excellent results. It is therefore inevitable that next generation networks will be assisted by these technologies, while as our knowledge increases better applications will be created.

In the first chapter there is an introduction to machine learning, we refer to the three types of machine learning, which analyze the basic techniques, the algorithms that use, their advantages and disadvantages. In the second chapter we refer to existing wireless communication systems giving their basic characteristics. For each system, a machine learning method is presented, which is adapted to the respective system, while experimental results are presented, comparing their performance with the traditional techniques so far. In the third and last one chapter, results from python simulations for two different systems are presented, showing their performance.

Ευχαριστίες

Ολοκληρώνοντας την εκπόνηση της διπλωματικής εργασίας , θα ήθελα να ευχαριστήσω την καθηγήτρια μου κα. Αγγελική Αλεξίου για την δυνατότητα που μου έδωσε να ασχοληθώ με αυτό το επιστημονικό αντικείμενο καθώς και την καθοδήγηση που μου πρόσφερε. Επίσης θα ήθελα να ευχαριστήσω το Πανεπιστήμιο Πειραιώς καθώς και όλους τους καθηγητές μου για την πολύτιμη γνώση που μου πρόσφεραν ώστε να ολοκληρωθεί το μεταπτυχιακό πρόγραμμα.

Σε όλους αυτούς που άδικα έφυγαν
από την πανδημία (Covid-19)

Περιεχόμενα

Κεφάλαιο 1	8
Μηχανική μάθηση	8
1.1.1 Εισαγωγή στη μηχανική μάθηση	8
1.1.2 Είδη μηχανικής μάθησης	9
1.2 Εισαγωγή στην μάθηση με επίβλεψη	12
1.2.1 Παλινδρόμηση	12
1.2.1.1 Μέθοδος ridge	15
1.2.1.2 Μέθοδος Lasso	17
1.2.1.3 Μέθοδος Elastic Net	17
1.2.1.4 Κριτήρια επίδοσης στην παλινδρόμηση	18
1.2.2 Ταξινόμηση	19
1.2.2.1 Λογιστική παλινδρόμηση	22
1.2.2.2 Γραμμική διακριτική ανάλυση (Linear discriminant analysis-LDA)	23
1.2.2.3 Μάθηση κατά Bayes	25
1.2.2.4 K-πλησιέστεροι γείτονες	28
1.2.2.5 Δέντρα Αποφάσεων	30
1.2.2.6 Μηχανές Διανυσμάτων υποστήριξης - Support Vector Machines (SVM)	33
1.2.2.6.1 Παλινδρόμηση με διανύσματα υποστήριξης	39
1.2.2.7 Κριτήρια επίδοσης στην ταξινόμηση	41
1.3 Εισαγωγή στη μη επιτηρούμενη μάθηση	44
1.3.3 Ο αλγόριθμος των K-μέσων	46
1.3.4 Αλγόριθμοι Ιεραρχικής Ομαδοποίησης	48
1.3.5 Μέθοδος DBSCAN (Density-based spatial clustering of applications with noise)	49
1.4 Εισαγωγή στην μάθηση με ενίσχυση	52
1.4.1 Διαδικασίες αποφάσεων Markov	54
1.4.2 Πολιτική Κέρδος και αξία	54
1.4.3 Q-learning	56
1.5 Εισαγωγή στο νευρωνικό δίκτυο	58
1.4.1 Το δίκτυο Perceptron	62
1.4.2 Νευρωνικά δίκτυα πολλών στρωμάτων	63
1.6 Βαθιά μηχανική μάθηση	66
Κεφάλαιο 2	67
Μηχανική μάθηση σε ασύρματα δίκτυα	67
Εισαγωγή 2.1	67
2.2 Εκτίμηση καναλιού MIMO με SVM	68
2.3 Προσδιορισμός παραμέτρων ασύρματων δικτύων μέσω H-SVM	75
2.4 Κατανομή πόρων με χρήση cloud Computing και μηχανική μάθηση	79
2.5 Χαμηλό latency σε δίκτυο 5G με μάθηση χωρίς επίβλεψη	87
2.6 Προσαρμοστική διαμόρφωση και κωδικοποίηση με μηχανική μάθηση	91
2.6.1 k-πλησιέστεροι γείτονες για AMC	93
2.6.2 SVM για AMC	97

2.6.3 Μάθηση με ενίσχυση για AMC	104
2.7 Μηχανική μάθηση για μη γραμμικούς ανιχνευτές MIMO	111
2.7.1 Ανιχνευτής MIMO με επιβλεπόμενη μάθηση	114
2.7.1.1 Μη παραμετρική μάθηση	114
2.7.1.2 Παραμετρική μάθηση	116
2.7.2 Low-Complexity SL (LCSL) Detector	119
2.8 OFDM με νευρωνικά δίκτυα	125
2.8.1 Εκτίμηση καναλιού σε σύστημα OFDM με νευρωνικό δίκτυο	125
2.8.2 Νευρωνικό δίκτυο σε MIMO OFDM	132
2.8.3 Νευρωνικό δίκτυο για OFDM χωρίς CP	141
2.9 Βαθιά μάθηση για MIMO-NOMA Downlink	148
Κεφάλαιο 3	159
Αποτελέσματα από προσομοιώσεις σε python	159
3.1 Εύρεση τοποθεσία χρήστη σε εσωτερικούς χώρους	159
3.2 Προσαρμοστική διαμόρφωση	165
Κεφάλαιο 4	172
Συμπεράσματα και προτάσεις για έρευνα	172
Βιβλιογραφία και ηλεκτρονικές πηγές	173

Κεφάλαιο 1

Μηχανική μάθηση

1.1.1 Εισαγωγή στη μηχανική μάθηση

Η μηχανική μάθηση είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη. Το αντικείμενο της μηχανική μάθηση είναι η μελέτη και η κατασκευή αλγορίθμων (προγραμμάτων), που μπορούν να μαθαίνουν και να βελτιώνονται αυτόματα από τα δεδομένα αλλά και να κάνουν προβλέψεις σχετικά με αυτά. Το είδος αυτών των αλγορίθμων λειτουργούν δημιουργώντας μοντέλα από πειραματικά δεδομένα, προκειμένου να κάνουν προβλέψεις βασιζόμενες σε αυτά τα δεδομένα ή να εξάγουν αποφάσεις που εκφράζονται ως αποτέλεσμα. Έτσι λοιπόν ένα σύστημα μηχανικής μάθησης, βελτιώνει την απόδοσή του κατά την εκτέλεση μιας συγκεκριμένης εργασίας, χωρίς να υπάρχει ανάγκη να προγραμματιστεί εκ νέου.

Για να υπάρξει μάθηση απαιτούνται τρία βασικά συστατικά:

1. ένα περιβάλλον το οποίο να προσφέρει δεδομένα υπό μορφή παραδειγμάτων στο σύστημα
2. ένα κριτήριο αξιολόγησης της επίδοσης του συστήματος
3. μια συγκεκριμένη εργασία την οποία το σύστημα καλείται να εκτελέσει.

Είναι προφανές ότι η διαδικασία της μάθησης δεν μπορεί να υλοποιηθεί με απλή απομνημόνευση όλων των πιθανών εκδοχών ενός αντικειμένου. Για παράδειγμα, σε ένα σύστημα αναγνώρισης αντικειμένων π.χ. ειδών ρουχισμού, (παντελόνια, φούστες, φανέλες κ.τ.λ.), διαπιστώνουμε ότι υπάρχουν άπειρα διαφορετικά αντικείμενα σε σχήματα, χρώματα, σχέδια και διαστάσεις. Αλλά ακόμη και το ίδιο αντικείμενο (π.χ. φούστα) έχει άπειρες όψεις ανάλογα με τη γωνία που θα την δούμε, τον φωτισμό, το χρώμα, το σχέδιο, το μήκος και την απόσταση από εμάς. Δεν είναι πρακτικά εφικτό να δημιουργήσουμε μια βάση δεδομένων με φωτογραφίες από αντικείμενα και να περιμένουμε να αναγνωρίσουμε το αντικείμενο εκτελώντας απλώς αναζήτηση σε αυτή τη βάση δεδομένων. Είναι σίγουρο ότι η συγκεκριμένη εικόνα δεν θα βρεθεί.

Από τα παραπάνω γίνεται προφανές, ότι στόχος της μάθησης, δεν είναι η συμβατική απομνημόνευση των δεδομένων που μπορεί να επιτευχθεί με τη χρήση ενός απλού πίνακα αποθήκευσης. Αντιθέτως, στόχος της μάθησης είναι η δυνατότητα παραγωγής σωστών εκτιμήσεων σχετικά με δεδομένα τα οποία αντιμετωπίζονται για πρώτη φορά από το σύστημα.

Η αξία της μηχανικής μάθησης εντοπίζεται ακριβώς στην παραπάνω ιδιότητα. Σε πάρα πολλές

εφαρμογές είναι πολύ σημαντικό να μπορούμε να εκτιμήσουμε την κατηγορία στην οποία ανήκουν άγνωστα μέχρι στιγμής δεδομένα ή να προβλέψουμε την αξία στην οποία αντιστοιχούν τα δεδομένα χωρίς να τα έχουμε ξαναδεί. Για να μπορέσει το σύστημα να ανταπεξέλθει στην παραπάνω απαίτηση, είναι απαραίτητη η παρουσίαση παρόμοιων αντικειμένων ή δεδομένων με τέτοιο τρόπο ώστε να αποκαλύπτεται η κρυμμένη σχέση μεταξύ των μεταβλητών.

Η μηχανική μάθηση είναι ένα υψηλό και συνεχώς εξελισσόμενο διεπιστημονικό πεδίο που δανείζεται και αλληλεπιδρά με εφαρμογές και επιστήμες όπως η στατιστική, η μηχανική, η επιστήμη των υπολογιστών και η γνωσιακή επιστήμη περιλαμβάνοντας ως επί το πλείστον αλγόριθμους βελτιστοποίησης. Δεν γνωρίζουμε ακόμη ακριβώς πώς να κάνουμε τους υπολογιστές να μαθαίνουν όπως μαθαίνουν οι άνθρωποι. Ωστόσο, έχουν επινοηθεί αλγόριθμοι που είναι αποτελεσματικοί για ορισμένα είδη μαθησιακών εργασιών ώστε να αρχίζει να προκύπτει μια θεωρητική κατανόηση της μάθησης. Πολλά πρακτικά προγράμματα υπολογιστών έχουν αναπτυχθεί για να παρουσιάσουν χρήσιμους τύπους μάθησης και έχουν αρχίσει να εμφανίζουν σημαντικές εμπορικές εφαρμογές. Για προβλήματα όπως η αναγνώριση ομιλίας, οι αλγόριθμοι που βασίζονται στη μηχανική μάθηση ξεπερνούν όλες τις άλλες προσεγγίσεις που έχουν μέχρι σήμερα επιχειρηθεί.

Στο πεδίο γνωστό ως εξόρυξη δεδομένων (data mining), χρησιμοποιούνται αλγόριθμοι μηχανικής μάθησης για να ανακαλύψουν πολύτιμες γνώσεις από μεγάλες εμπορικές βάσεις δεδομένων που περιέχουν αρχεία συντήρησης εξοπλισμού, αιτήσεις δανείων, οικονομικές συναλλαγές, ιατρικά αρχεία και άλλα παρόμοια. Καθώς, η κατανόησή μας για τους υπολογιστές συνεχίζει να ωριμάζει, φαίνεται αναπόφευκτο ότι η μηχανική μάθηση θα διαδραματίσει έναν όλο και πιο κεντρικό ρόλο στην επιστήμη και την τεχνολογία των υπολογιστών τα επόμενα χρόνια.

1.1.2 Είδη μηχανικής μάθησης

Ακριβώς όπως υπάρχουν διαφορετικοί τρόποι με τους οποίους μαθαίνουν οι άνθρωποι από το περιβάλλον τους, το ίδιο ισχύει και για τη μηχανική μάθηση.

Οι εργασίες μηχανικής μάθησης συνήθως ταξινομούνται σε τρεις μεγάλες κατηγορίες, ανάλογα με τη φύση του εκπαιδευτικού «σήματος» ή την «ανατροφοδότηση» που είναι διαθέσιμη σε ένα σύστημα εκμάθησης. Αυτές είναι:

1. **Επιτηρούμενη μάθηση, (αλλιώς επιβλεπόμενη μάθηση ή μάθηση με επίβλεψη) (supervised learning)** : Σε αυτό το είδος της μάθησης το υπολογιστικό πρόγραμμα δέχεται τις παραδειγματικές εισόδους καθώς και τα επιθυμητά αποτελέσματα από έναν «δάσκαλο», και ο στόχος είναι να μάθει έναν γενικό κανόνα προκειμένου να αντιστοιχίσει τις εισόδους με τα αποτελέσματα. Κατά την διαδικασία μάθησης με επίβλεψη ο αλγόριθμος κατασκευάζει μια συνάρτηση που απεικονίζει δεδομένες εισόδους (σύνολο εκπαίδευσης) σε γνωστές επιθυμητές εξόδους, με απώτερο στόχο τη γενίκευση της συνάρτησης αυτής και για εισόδους με άγνωστη έξοδο.

Τα προβλήματα μάθησης με επίβλεψη είναι ίσως τα συνηθέστερα προβλήματα μηχανικής μάθησης. Χωρίζονται σε δύο μεγάλες κατηγορίες.

- Προβλήματα ταξινόμησης (classification problems):

Σε αυτά τα προβλήματα οι στόχοι είναι διακριτές τιμές, π.χ. $t_i \in \{0, 1\}$, $t_i \in \{0, 1, 2, \dots, C\}$ κ.λπ. Οι στόχοι αντιστοιχούν σε κλάσεις αντικειμένων. Για παράδειγμα, αν το πρόβλημα είναι η ταξινόμηση μηνυμάτων σε δύο κλάσεις, spam ή ham, θα μπορούσαμε να θέσουμε ως στόχο το $t_i = 1$ αν το πρότυπο x_i αντιστοιχεί σε spam, ή το $t_i = 0$ αν το πρότυπο αντιστοιχεί σε ham.

- Προβλήματα παλινδρόμησης (regression problems):

Σε αυτά τα προβλήματα οι στόχοι είναι συνεχείς τιμές ή απεριόριστο πλήθος διακριτών τιμών, π.χ. $t_i \in \mathbb{R}$. Οι στόχοι αντιστοιχούν σε τιμές ή αξίες κάποιων ποσοτήτων. Για παράδειγμα, αν το πρόβλημα είναι η εκτίμηση των τιμών των κατοικιών μιας πόλης, οι στόχοι θα είναι πραγματικές θετικές τιμές.

Χρησιμοποιείται σε προβλήματα:

- Ταξινόμησης (Classification)
- Πρόγνωσης (Prediction)
- Διερμηνείας (Interpretation)

2. Μη επιτηρούμενη μάθηση (αλλιώς μη επιβλεπόμενη μάθηση ή μάθηση χωρίς επίβλεψη (unsupervised learning) : Χωρίς να παρέχεται κάποια εμπειρία στον αλγόριθμο μάθησης, πρέπει να βρει την δομή των δεδομένων εισόδου. Ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο εισόδων υπό μορφή παρατηρήσεων χωρίς να γνωρίζει τις επιθυμητές εξόδους. Η μη επιτηρούμενη μάθηση μπορεί να είναι αυτοσκοπός (ανακαλύπτοντας κρυμμένα μοτίβα σε δεδομένα) ή μέσο για ένα τέλος (χαρακτηριστικό της μάθησης). Στη μη επιβλεπόμενη μάθηση ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο εισόδων υπό μορφή παρατηρήσεων χωρίς να γνωρίζει τις επιθυμητές εξόδους.

Η κύρια κατηγορίες στην οποία βρίσκει εφαρμογή είναι :

- Ομαδοποίησης (Clustering)

Η ομαδοποίηση συνιστάται στην ανακάλυψη ομάδων από αντικείμενα που παρουσιάζουν μεταξύ τους κοινά χαρακτηριστικά. Οι κατηγορίες στην περίπτωση αυτή δεν είναι γνωστές εκ των προτέρων (δεν δίνονται δηλαδή από τον χρήστη) αλλά προκύπτουν δυναμικά κατά την εκτέλεση του αλγορίθμου ομαδοποίησης. Στην εκτίμηση παραμέτρων κατασκευάζουμε ένα στατιστικό μοντέλο του προς αντιμετώπισης προβλήματος, που διαθέτει ένα σύνολο παραμέτρων, τις τιμές των οποίων επιθυμούμε να προσδιορίσουμε. Για το σκοπό αυτό χρησιμοποιούμε έναν αλγόριθμο μάθησης που εκπαιδεύεται σε αντιπροσωπευτικά δεδομένα του προβλήματος μας, προκειμένου να προσεγγίσει τις ζητούμενες παραμέτρους.

Χρησιμοποιείται σε προβλήματα:

- Ανάλυσης Συσχετισμών (Association Analysis)
- Εκτίμηση παραμέτρων (parameter estimation)

3. Ενισχυτική μάθηση (Reinforcement Learning): Στην μάθηση με ενίσχυση, η εκμάθηση μιας αντιστοίχισης εισόδου-εξόδου εκτελείται μέσω συνεχούς αλληλεπίδρασης με το περιβάλλον, και έχει ως στόχο την ελαχιστοποίηση ενός βαθμωτού δείκτη απόδοσης. Στην μάθηση με ενίσχυση το πρόγραμμα υπολογιστή αλληλεπιδρά με το δυναμικό περιβάλλον στο οποίο ένας συγκεκριμένος στόχος πρέπει να επιτευχθεί (όπως η οδήγηση ενός οχήματος), χωρίς κάποιος δάσκαλος να του λέει ρητά αν έχει φτάσει κοντά στο στόχο του.

Η τεχνική αυτή λαμβάνει χώρα σε προβλήματα :

Σχεδιασμού (Planning), όπως για παράδειγμα ο έλεγχος κίνησης ρομπότ και η βελτιστοποίηση εργασιών σε εργοστασιακούς χώρους ή σε παιχνίδι εναντίον κάποιου αντιπάλου.

➤ Μεταξύ της επιτηρούμενης και της μη επιτηρούμενης μάθησης είναι η ημι-επιτηρούμενη μάθηση, στην οποία ο δάσκαλος δίνει μερικά ελλιπείς δεδομένα , δηλαδή ένα σύνολο εκπαίδευσης με κάποια (συχνά πολλά) από τα αποτελέσματα στόχους να λείπουν. Η μεταγωγή είναι μια υποκατηγορία της μάθησης αυτής, στην οποία το σύνολο των καταστάσεων του προβλήματος είναι γνωστό κατά το χρόνο εκμάθησης, όμως ένα μέρος των στόχων λείπουν.

➤ Η μετα-εκμάθηση (meta-learning) αποτελεί ένα άλλο υποπεδίο της μηχανικής μάθησης όπου οι αλγόριθμοι αυτόματης μάθησης εφαρμόζονται στα μεταδεδομένα που σχετίζονται με το πείραμα. Ο στόχος εδώ είναι να μαθαίνει η μηχανή (να αναπτύσσει) τις δικές της επαγωγικές μεθόδους, βασιζόμενη στην προηγούμενη εμπειρία.

1.2 Εισαγωγή στην μάθηση με επίβλεψη

Σε ένα σύστημα μηχανικής μάθησης με επίβλεψη, η είσοδος δίνεται υπό τη μορφή διανύσματος X το οποίο ονομάζεται πρότυπο. Το διάνυσμα περιέχει εκείνα τα χαρακτηριστικά ή στοιχεία που απαιτούνται για την εκτίμηση της ζητούμενης τιμής t . Υπάρχουν δύο τύποι αλγορίθμων μάθησης με επίβλεψη: Παλινδρόμηση και ταξινόμηση. Ο πρώτος προβλέπει τις συνεχείς τιμές εξόδου ενώ ο δεύτερος προβλέπει διακριτές τιμές εξόδου.

1.2.1 Παλινδρόμηση

Ο στόχος ενός μοντέλου παλινδρόμησης είναι να βρεθεί μια σχέση μεταξύ ενός ή περισσότερων χαρακτηριστικών (ανεξάρτητων μεταβλητών) και μιας συνεχούς μεταβλητής στόχου (εξαρτώμενη μεταβλητή). Όταν υπάρχει ένα χαρακτηριστικό (μεταβλητή), ονομάζεται απλή παλινδρόμηση, ενώ αν υπάρχουν περισσότερα από ένα, ονομάζεται πολλαπλή παλινδρόμηση. Η μέθοδος αυτή αποτελεί μία στατιστική διαδικασία για την εκτίμηση της σχέσης μεταξύ μιας εξαρτημένης μεταβλητής και μιας ή περισσότερων ανεξάρτητων μεταβλητών. Πιο συγκεκριμένα, μέσω της διαδικασίας της παλινδρόμησης μελετάται η μεταβολή της τιμής της εξαρτημένης μεταβλητής όταν μεταβάλλεται οποιαδήποτε από τις ανεξάρτητες μεταβλητές. Πιο αναλυτικά, έστω ότι έχουμε ένα σύνολο N δεδομένων εκπαίδευσης που περιλαμβάνει τις παρατηρήσεις $\{X_n\}$, όπου $n = 1, \dots, N$, μαζί με τις αντίστοιχες τιμές στόχου $\{t_n\}$, ο στόχος είναι να προβλέψουμε την τιμή του t για μια νέα τιμή του x . Στην απλούστερη προσέγγιση, αυτό μπορεί να γίνει με άμεση κατασκευή μιας κατάλληλης συνάρτησης $y(x)$ των οποίων οι τιμές για τις νέες εισόδους x συνιστούν τις προβλέψεις για τις αντίστοιχες τιμές του t . Γενικότερα, από πιθανοτικής πλευράς, στοχεύουμε να μοντελοποιήσουμε την προβλεπόμενη κατανομή $p(t|x)$ διότι αυτό εκφράζει την αβεβαιότητά μας σχετικά με την τιμή t για κάθε τιμή του x . Από αυτή την υπό όρους κατανομή μπορούμε να κάνουμε προβλέψεις t , για κάθε νέα τιμή του x , με τέτοιο τρόπο ώστε να ελαχιστοποιούμε την αναμενόμενη τιμή μιας κατάλληλα επιλεγμένης συνάρτησης. Στην τεχνική της παλινδρόμησης ανάγονται και πρόβλημα προσαρμογής καμπύλης (curve fitting problem). Έτσι λοιπόν δεδομένου ενός συνόλου σημείων, στόχος είναι η προσαρμογή μιας ομάδας σημείων ενός διαγράμματος σε μια καμπύλη και η εύρεση της αντίστοιχης εξίσωσης που να ταιριάζει καλύτερα με τα δεδομένα μας. Μόλις βρεθεί μια τέτοια συνάρτηση, όταν έρθει ένα άγνωστο στοιχείο μπορούμε να προβλέψουμε την τιμή του.

Ένα από τα πιο συνηθισμένα μοντέλα παλινδρόμησης είναι η γραμμική παλινδρόμηση, που προϋποθέτει ότι η έξοδος είναι γραμμική συνάρτηση των εισόδων. Αυτή η σχέση μπορεί να εκφραστεί ως εξής (πολλαπλή):

$$Y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

Y : είναι η προβλεπόμενη τιμή (εξαρτώμενη μεταβλητή)

θ_0 : είναι ο όρος bias (τομή με άξονα y)

$\theta_1, \dots, \theta_n$: είναι οι παράμετροι του μοντέλου

x_1, x_2, \dots, x_n : οι τιμές των χαρακτηριστικών (ανεξάρτητη μεταβλητή).

Έτσι, αυτή η τεχνική παλινδρόμησης βρίσκει μια γραμμική σχέση μεταξύ των x (input) και y (output).

Η παραπάνω εξίσωση μπορεί να γραφτεί συντομότερα ως:

$$Y = \theta^T x$$

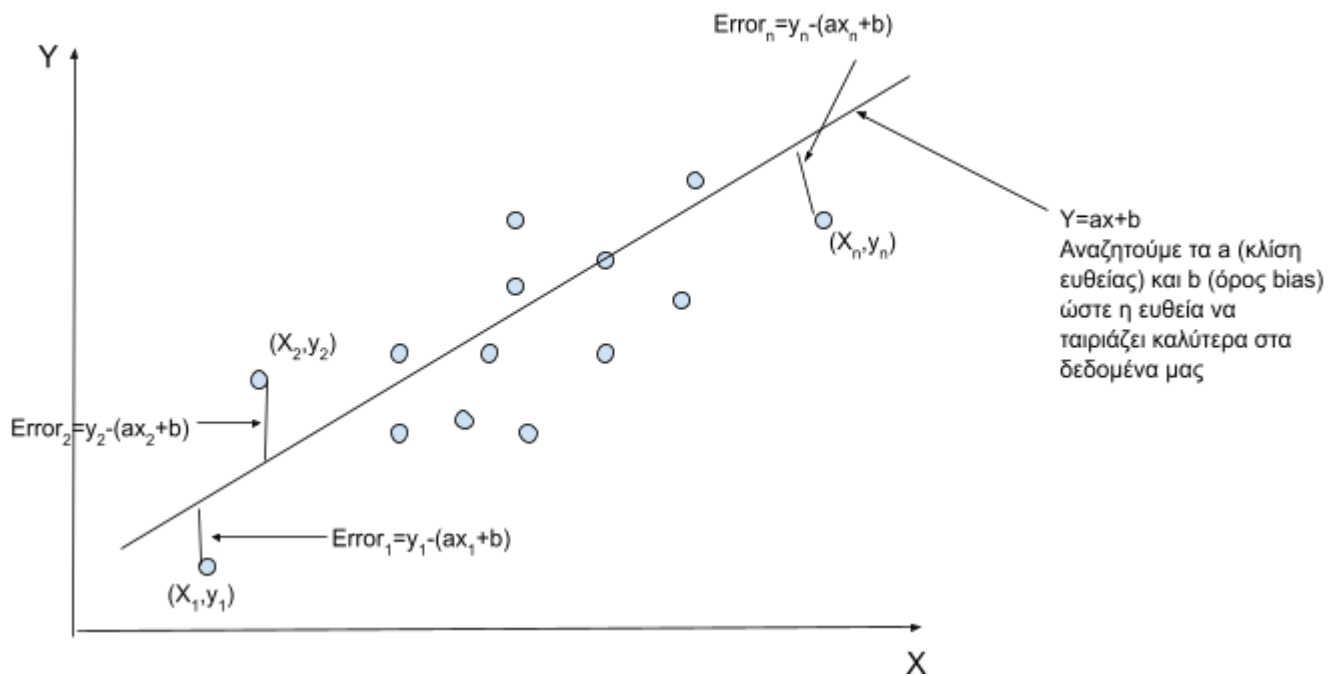
θ : είναι ένα διάνυσμα με τις παραμέτρους του διανύσματος συμπεριλαμβανομένου του όρου bias

x : είναι ένα διάνυσμα με τις τιμές του x_1, x_2, \dots, x_n .

Συνήθως στις εξισώσεις προσθέτουμε μια επιπλέον μεταβλητή ϵ , που αποτελεί το τυχαίο σφάλμα που υποθέτουμε ότι ακολουθεί κανονική κατανομή (Gaussian) $\epsilon \sim N(\mu, \sigma^2)$ όπου μ είναι ο μέσος όρος και σ^2 η διασπορά.

Αυτό που κάνει ο αλγόριθμος γραμμικής παλινδρόμησης είναι ότι ταιριάζει με πολλαπλές γραμμές τα σημεία δεδομένων και επιστρέφει τη γραμμή που οδηγεί στο ελάχιστο σφάλμα. Η γραμμή για την οποία το σφάλμα μεταξύ των προβλεπόμενων τιμών και των παρατηρούμενων τιμών είναι ελάχιστη ονομάζεται γραμμή καλύτερης προσαρμογής ή γραμμή παλινδρόμησης. Από μαθηματικής πλευράς ο αλγόριθμος λύνει το πρόβλημα $\min \|y - X_{\theta}\|_2^2$ που αποτελεί την ευκλείδεια απόσταση ή νόρμα δυο διανυσμάτων. Ελαχιστοποιήσουμε δηλαδή αυτή την εξίσωση για όλα μας τα δεδομένα ώστε να βρούμε την καλύτερη δυνατή γραμμή μεταξύ των δεδομένων μας ώστε να έχουμε και το μικρότερο σφάλμα. Για την εκτίμηση των παραμέτρων μας εφαρμόζουμε την μέθοδο των ελαχίστων τετραγώνων. Το πρόβλημα μπορεί να αποτελείται από δύο ή περισσότερες παραμέτρους.

Ας υποθέσουμε την πιο απλή περίπτωση που το πρόβλημα αποτελείται από δύο παραμέτρους. Ας απεικονίσουμε τα N σημεία μας στον δισδιάστατο χώρο \mathbb{R}^2 όπως στο επόμενο σχήμα. Αυτό που θέλουμε να πετύχουμε είναι να βρούμε μια ευθεία που ελαχιστοποιεί την τετραγωνική απόσταση μεταξύ αυτών των σημείων.



Σχήμα 1.1. Ευθεία παλινδρόμησης και δεδομένα

Θέλουμε να ελαχιστοποιήσουμε το τετραγωνικό σφάλμα του κάθε σημείου από τη γραμμή (SE).

$$SE_{LINE} = (y_1 - (ax_1 + b))^2 + (y_2 - (ax_2 + b))^2 + \dots + (y_n - (ax_n + b))^2$$

$$= \sum_{i=1}^n (y_i - a - bx_i)^2$$

Θέλουμε να βρούμε τα a και b που ελαχιστοποιούν την παραπάνω εξίσωση.

$$SE_{LINE} = y_1^2 + y_2^2 + \dots + y_n^2 - 2a(x_1y_1 + x_2y_2 + \dots + x_ny_n) - 2b(y_1 + y_2 + \dots + y_n) + a^2(x_1^2 + x_2^2 + \dots + x_n^2) + 2ab(x_1 + x_2 + \dots + x_n) + na^2$$

$$\overline{y^2} = \frac{y_1^2 + y_2^2 + \dots + y_n^2}{n} \quad \overline{xy} = \frac{x_1y_1 + x_2y_2 + \dots + x_ny_n}{n}$$

$$SE_{LINE} = n\overline{y^2} - 2an\overline{xy} - 2bn\overline{y} + m^2n\overline{x^2} + 2abn\overline{x} + nb^2$$

$$\frac{\partial SE}{\partial a} = 0 \implies -2n\overline{xy} + 2an\overline{x^2} + 2bn\overline{x} = 0 \implies a\overline{x^2} + b\overline{x} = \overline{xy} \implies a\frac{\overline{x^2}}{\overline{x}} + b = \frac{\overline{xy}}{\overline{x}}$$

$$\frac{\partial SE}{\partial b}=0 \implies -2n\bar{y}+2an\bar{x}+2bn=0 \implies a\bar{x}+b=\bar{y}$$

αφαιρώντας κατά μέλη τις δυο παραπάνω εξισώσεις παίρνουμε :

$$a \left(\bar{x} - \frac{\bar{x}^2}{\bar{x}} \right) = \bar{y} - \frac{\bar{xy}}{\bar{x}} \implies a = \frac{\bar{y} - \frac{\bar{xy}}{\bar{x}}}{\bar{x} - \frac{\bar{x}^2}{\bar{x}}} = \frac{\bar{x}\bar{y} - \bar{xy}}{\bar{x}^2 - \bar{x}^2}$$

$$b = \bar{y} - a\bar{x}$$

Γενικεύοντας:

$$a = \frac{(\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i) - \sum_{i=1}^n x_i y_i}{(\sum_{i=1}^n x_i)^2 - \sum_{i=1}^n x_i^2}$$

$$b = \sum_{i=1}^n y_i - a \sum_{i=1}^n x_i$$

1.2.1.1 Μέθοδος ridge

Η μέθοδος των ελαχίστων τετραγώνων επιτυγχάνει καλύτερα όταν οι ανεξάρτητες μεταβλητές που αποτελούν το μοντέλο είναι μεταξύ τους ασυσχέτιστες. Όταν υπάρχουν μεγάλες συσχετίσεις μεταξύ των μεταβλητών είναι δύσκολο, αν όχι αδύνατο, να αξιολογηθεί η ουσιαστική προσφορά μιας συγκεκριμένης ανεξάρτητης μεταβλητής επί της εξαρτημένης μεταβλητής που οφείλεται αποκλειστικά στη συγκεκριμένη ανεξάρτητη μεταβλητή. Υπάρχει δηλαδή ενδεχόμενο οι εκτιμώμενοι συντελεστές παλινδρόμησης να είναι εξαιρετικά ασταθείς και οι τιμές τους να υφίστανται δραματικές αλλαγές όταν κάποια νέα μεταβλητή προστίθεται ή απομακρύνεται ή όταν συμβαίνουν μικρές μεταβολές στα δεδομένα του προβλήματος.

Η κατάσταση η οποία δημιουργείται όταν υπάρχουν ισχυρές συσχετίσεις μεταξύ των ανεξάρτητων μεταβλητών στην πολλαπλή παλινδρόμηση ονομάζεται πολυσυγγραμμικότητα (multicollinearity).[4]. Αν οι μεταβλητές ενός προβλήματος που θέλουμε να αναλύσουμε παρουσιάζουν πολυσυγγραμμικότητα (multicollinearity) οι τιμές της παλινδρόμησης με την τεχνική μέθοδο των ελαχίστων τετραγώνων παρουσιάζουν αρκετά μεγάλη διασπορά, γεγονός που σημαίνει ότι οι τιμές τους μπορεί να απέχουν πολύ από την πραγματική τιμή. Η

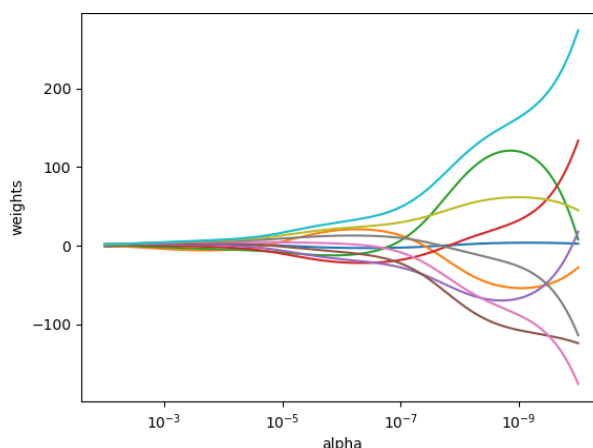
πολυσυγγραμμικότητα είναι ένα φαινόμενο κατά το οποίο μία ανεξάρτητη μεταβλητή ενός μοντέλου πολλαπλής παλινδρόμησης μπορεί να προβλεφθεί γραμμικά από τις υπόλοιπες με μεγάλη ακρίβεια. Στο μοντέλο της γραμμικής παλινδρόμησης η ύπαρξη πολυσυγγραμμικότητας οδηγεί σε υψηλά τυπικά σφάλματα για τις εκτιμήτριες ελαχίστων τετραγώνων και γι' αυτό το λόγο είναι δύσκολο να βρεθούν οι στατιστικά σημαντικές μεταβλητές που θα συμμετέχουν στο μοντέλο. Μια εναλλακτική μέθοδος για την αντιμετώπιση της πολυσυγγραμμικότητας είναι η μέθοδος της αμφικλινούς παλινδρόμησης (ridge regression). Η εναλλακτική αυτή μέθοδος έχει ως στόχο να παρακάμψει την δυσκολία που δημιουργείται από την ύπαρξη μεγάλων συσχετίσεων μεταξύ μεταβλητών. Η παλινδρόμηση ridge, επομένως, μειώνει εξίσου τις τιμές των συντελεστών των ανεξάρτητων μεταβλητών, προσθέτει, δηλαδή, ένα ποσοστό σφάλματος (bias), έτσι ώστε να μειώσει τα σφάλματα. Ο αντίστοιχος εκτιμητής Ridge παλινδρόμησης επιλύει το πρόβλημα παλινδρόμησης ελαχιστοποιώντας την

$$\|y - xa\|_2^2 + \lambda \|a\|_2^2$$

όπου :

$\|y - xa\|_2^2 = \sum_{i=1}^n (y_i - x_i^T a)^2$ είναι τετραγωνική (quadratic) ή L2-norm συνάρτηση κόστους, η οποία λαμβάνει το άθροισμα των τετραγωνισμένων τιμών των σφαλμάτων. x_i^T συμβολίζεται η i -οστή γραμμή του πίνακα X , ενώ ο όρος $\lambda \|a\|_2^2 = \lambda \sum_{i=1}^n a_i^2$ αποτελεί τον όρο πέναλτι της συνάρτησης κόστους (τιμωρεί το μοντέλο όταν τείνει να υπερ εκπαιδευτεί)

Ο ρόλος της παραμέτρου $\lambda > -1$ είναι να ελέγχει την ποσότητα της συρρίκνωσης και καλείται παράμετρος κανονικοποίησης. Αν η παράμετρος λ είναι ίση με μηδέν, τότε πρόκειται για απλή παλινδρόμηση, όπου δεν εφαρμόζεται κανονικοποίηση. Ωστόσο, όσο αυξάνεται το λ τόσο πρέπει να μειώνεται η τιμή των συντελεστών, η οποία πρέπει να προσεγγίζει το μηδέν, έτσι ώστε το μέγεθος του όρου πέναλτι να είναι αρκετά μικρό, έχει δηλαδή την ιδιότητα να συρρικνώνει τους συντελεστές ώστε να προσεγγίζουν το μηδέν. Στο παρακάτω διάγραμμα κάθε χρώμα αντιπροσωπεύει ένα διαφορετικό χαρακτηριστικό του συντελεστή διανύσματος. $\lambda \|a_1\|$



Σχήμα 1.2. Συντελεστής Ridge ως συνάρτηση της παραμέτρου κανονικοποίησης [6]

1.2.1.2 Μέθοδος Lasso

Ένα πρόβλημα που παρουσιάζεται με την χρήση της μεθόδου Ridge, είναι ότι δεν γίνεται επιλογή των μεταβλητών που συμμετέχουν στο μοντέλο. Όταν ο αριθμός των επεξηγηματικών μεταβλητών είναι μεγάλος, το μοντέλο δεν είναι εύκολο να ερμηνευθεί, καθώς χρησιμοποιούνται όλες οι ανεξάρτητες μεταβλητές, χωρίς να γίνεται επιλογή ενός υποσυνόλου αυτών. Λύση σε αυτό το πρόβλημα έρχεται να δώσει η παλινδρόμηση Lasso, η οποία επιλέγει τις μεταβλητές που έχουν τη σημαντικότερη επίδραση στην εξαρτημένη μεταβλητή. Η Lasso παλινδρόμηση χρησιμοποιείται σε εφαρμογές που υπάρχει τεράστιος όγκος δεδομένων, καθώς απαιτούνται πολύ αποτελεσματικοί και γρήγοροι αλγόριθμοι. Ωστόσο, η Lasso παλινδρόμηση δε δίνει αξιόπιστα αποτελέσματα όταν υπάρχει υψηλή συσχέτιση μεταξύ των ανεξάρτητων μεταβλητών, καθώς θα επιλέξει αυθαίρετα μία και θα αγνοήσει όλες τις υπόλοιπες. Το πέναλτι Lasso υποθέτει ότι πολλοί συντελεστές θα έχουν τιμή κοντά στο μηδέν και μόνο ένα μικρό υποσύνολο αυτών θα λαμβάνει μεγαλύτερες τιμές. Ο αντίστοιχος εκτιμητής Lasso παλινδρόμησης επιλύει το πρόβλημα παλινδρόμησης ελαχιστοποιώντας την

$$\|y - xa\|_2^2 + \lambda \|a\|_1$$

όπου $\lambda \|a\|_1 = \lambda \sum_{i=1}^n |a_i|$ είναι ο όρος πέναλτι της συνάρτησης κόστους και $\lambda > -1$ είναι η παράμετρος κανονικοποίησης όπως και στην παλινδρόμηση Ridge. Το πέναλτι της L1-norm κανονικοποιεί τη συνάρτηση ελαχίστων τετραγώνων και μηδενίζει τις τιμές ορισμένων συντελεστών για ένα κατάλληλα επιλεγμένο λ , σε αντίθεση με τη Ridge παλινδρόμηση, όπου όλες οι τιμές είναι μη-μηδενικές. Όσο αυξάνεται η τιμή του λ , τόσο συρρικνώνονται οι συντελεστές παλινδρόμησης.

1.2.1.3 Μέθοδος Elastic Net

Η Elastic Net μέθοδος παλινδρόμησης συνδυάζει γραμμικά τα penalty από τις L1 και L2 νόρμες των μεθόδων Lasso και Ridge και χρησιμοποιείται για να ξεπεραστούν οι περιορισμοί της Lasso μεθόδου, όπως για παράδειγμα η περίπτωση πολύ λίγων δειγμάτων σε δεδομένα πολλών διαστάσεων, όπου η Lasso επιλέγει το πολύ τόσες μεταβλητές όσος ο αριθμός των δειγμάτων. Ακόμα, στην περίπτωση που οι ανεξάρτητες μεταβλητές συσχετίζονται σε μεγάλο βαθμό, η μέθοδος Lasso τείνει να επιλέγει μόνο μία μεταβλητή και να αγνοεί τις υπόλοιπες. Για να ξεπεραστούν αυτές οι δυσκολίες στη μέθοδο παλινδρόμησης Elastic Net χρησιμοποιούνται οι όροι πέναλτι τόσο της Ridge όσο και της Lasso μεθόδου ελαχιστοποιώντας την παρακάτω εξίσωση:

$$\|y - xa\|_2^2 + \lambda_1 \|a\|_2^2 + \lambda_2 \|a\|_1$$

Από τα παραπάνω παρατηρούμε ότι η σημασία των μοντέλων γραμμικής παλινδρόμησης είναι αρκετά μεγάλη, τόσο στην επιστήμη της στατιστικής, όσο και στην επιστήμη των υπολογιστών. Παρόλα' αυτά, κάποια βασικά μειονεκτήματά τους μας οδηγούν στη χρήση πιο πολύπλοκων μοντέλων. Η γραμμική παλινδρόμηση περιορίζεται σε γραμμικά προβλήματα. Από τη φύση της η γραμμική παλινδρόμηση κοιτάζει τις γραμμικές σχέσεις μεταξύ των εξαρτημένων και των ανεξάρτητων μεταβλητών. Συνεπώς υποθέτει ότι η σχέση τους μπορεί να απεικονιστεί με τη μορφή μιας ευθείας γραμμής. Αυτό φυσικά δεν είναι πάντα σωστό, καθώς στα περισσότερα προβλήματα οι ευθείες που συσχετίζουν τις μεταβλητές δεν είναι γνησίως μονότονες. Η γραμμική παλινδρόμηση εστιάζει μόνο στη μέση τιμή της εξαρτημένης μεταβλητής. Όπως η μέση τιμή δεν αντιπροσωπεύει πάντα πλήρως μία μεταβλητή, έτσι και η γραμμική παλινδρόμηση δεν αντιπροσωπεύει πλήρως τις σχέσεις μεταξύ των μεταβλητών. Η γραμμική παλινδρόμηση επηρεάζεται από ακραίες τιμές παρατηρήσεων. Στα περισσότερα δεδομένα παρατηρούνται πολλές φορές ακραίες τιμές εξόδου σε ορισμένες παρατηρήσεις. Στην περίπτωση της γραμμικής παλινδρόμησης οι τιμές αυτές επηρεάζουν σε μεγάλο βαθμό την κλίση της ευθείας. Οι παρατηρήσεις αυτές ονομάζονται παρατηρήσεις με ισχυρή επιρροή (influential observations) και η διαγραφή τους μπορεί να αυξήσει την απόδοση του μοντέλου μας. Αυτό συμβαίνει διότι δεν επιθυμούμε η καμπύλη μας να επηρεάζεται από τις ακραίες τιμές, αλλά από τιμές οι οποίες είναι αντιπροσωπευτικές του συνόλου που μελετάμε. Η γραμμική παλινδρόμηση υποθέτει ότι τα δεδομένα είναι ανεξάρτητα μεταξύ τους. Αυτό σημαίνει ότι η τιμή μιας ανεξάρτητης μεταβλητής δε θα έπρεπε να επηρεάζει την τιμή μιας άλλης. Αυτό φυσικά δεν είναι πάντα σωστό. Πολλές φορές οι ανεξάρτητες μεταβλητές σε ένα πρόβλημα συσχετίζονται, γεγονός που επιφέρει αρνητικές συνέπειες στα αποτελέσματά μας. Με την ανάλυση συσχέτισης (correlation analysis) μπορούμε να μετρήσουμε και να ερμηνεύσουμε το κατά πόσο η γραμμική ή μη γραμμική σχέση μεταξύ δύο συνεχόμενων μεταβλητών είναι ισχυρή.

1.2.1.4 Κριτήρια επίδοσης στην παλινδρόμηση

Όπως αναφέρθηκε στην παλινδρόμηση οι επιθυμητές τιμές είναι πραγματικοί αριθμοί. Ως κριτήριο μέτρησης της επίδοσης έχουμε την απόσταση μεταξύ της εξόδου του μοντέλου και της επιθυμητής τιμής. Έστω ότι έχουμε N πρότυπα εισόδου x_p , $p=1..N$, όπου η επιθυμητή τιμή-στόχος t_p για το p -οστό πρότυπο είναι ένα διάνυσμα διάστασης m , ενώ αντίστοιχα η έξοδος του μοντέλου για το ίδιο πρότυπο είναι το διάνυσμα y_p .

Ένα από τα πιο γνωστά κριτήρια αποτελεί το μέσο τετραγωνικό σφάλμα (mean squared error-MSE). Το κριτήριο αυτό μετράει τη μέση ευκλείδεια απόσταση μεταξύ του διανύσματος εξόδου y_p του μοντέλου και του διανύσματος-στόχου t_p . Το κριτήριο παίρνει πάντα μη αρνητικές τιμές με βέλτιστη την τιμή μηδέν.

$$J_{MSE} = \sum_{p=1}^N \|t_p - y_p\|^2 = \sum_{p=1}^N \sum_{i=1}^m (t_{p,i} - y_{p,i})^2$$

Ένα άλλο δημοφιλές κριτήριο για την επίδοση στην παλινδρόμηση είναι το μέσο απόλυτο σφάλμα (mean absolute error-MAE), το οποίο μετράει τη μέση απόλυτη απόσταση μεταξύ των διανυσμάτων y_p και t_p . Όπως και το προηγούμε έτσι και αυτό το κριτήριο παίρνει μη αρνητικές τιμές με βέλτιστη το μηδέν.

$$J_{MAE} = \sum_{p=1}^N \sum_{i=1}^m |t_{p,i} - y_{p,i}|$$

1.2.2 Ταξινόμηση

Η ταξινόμηση είναι η διαδικασία της αντιστοίχισης ενός αντικειμένου σε μία κατηγορία ή αλλιώς κλάση, ανάμεσα σε διάφορες προκαθορισμένες κλάσεις. Σε αντίθεση με τα προβλήματα παλινδρόμησης, σε αυτή την τεχνική οι τιμές στόχοι είναι διακριτές τιμές. Σκοπός της ταξινόμησης είναι δοθέντος ενός συνόλου δεδομένων να τα αντιστοιχίσει τα δεδομένα σε μία ή περισσότερες ξεχωριστές κλάσεις. Στο πιο συνηθισμένο σενάριο, οι κλάσεις θεωρούνται διακριτές, ώστε κάθε πρότυπο να αντιστοιχεί σε μόνο μια κλάση. Στις περισσότερες εφαρμογές αναγνώρισης προτύπων αντιμετωπίζουμε ένα πρόβλημα ταξινόμησης δύο κλάσεων, όπου συνήθως η μια κλάση περιέχει πρότυπα για τα οποία ισχύει μια κατάσταση ή συνθήκη X (κλάση 1), ενώ η άλλη κλάση περιέχει πρότυπα για τα οποία αυτή η κατάσταση ή συνθήκη δεν ισχύει (κλάση 0). Με άλλα λόγια, η ετικέτα της κλάσης είναι μια δυαδική (boolean) μεταβλητή (0/1- \rightarrow true/false) που αντιστοιχεί στην ύπαρξη ή στην μη ύπαρξη της κατάστασης ή συνθήκης X . Η κωδικοποίηση των κλάσεων σε τιμές μπορεί να γίνει με διαφορετικούς τρόπους. Στον παρακάτω πίνακα παρουσιάζονται οι πιο συχνά χρησιμοποιούμενοι τρόποι κωδικοποίησης στόχων (t) σε προβλήματα ταξινόμησης ανάλογα το πλήθος των κλάσεων.

Κλάσεις (c)	Στόχοι (t)
c=2	<ul style="list-style-type: none"> • $t \in \{0,1\}$ ή • $t \in \{-1,1\}$
c>2	<p>Μπορούμε να επιλέξουμε ανάμεσα στις παρακάτω κωδικοποιήσεις:</p> <ul style="list-style-type: none"> • $t \in \{0,1,\dots,\omega-1\}$ • $t \in \{1,2,\dots,\omega\}$ • $t \in [0.0 \ 1 \ 0 \dots 0]$ <p>Στη τελευταία περίπτωση στόχος είναι ένα διάνυσμα στο οποίο το μοναδικό 1 βρίσκεται στη θέση k: ο αύξον αριθμός της κλάσης του</p>

	αντικειμένου. π.χ. αν $t \in [0 \ 1 \ 0]$ το αντικείμενο ανήκε στην κλάση 2 Η κωδικοποίηση αυτή είναι γνωστή ως κωδικοποίηση μονού σημαντικού (one-hot encoding)
--	--

Σχήμα 1.3. Κωδικοποίηση στόχων σε προβλήματα ταξινόμησης [1]

Για την μέθοδο της ταξινόμησης θα παρουσιάσουμε δύο διαφορετικές προσεγγίσεις σχεδίασης ταξινομητών. Θα αναφερθούμε σε ταξινομητές που βασίζονται σε γραμμικές συναρτήσεις αλλά και ταξινομητές που βασίζονται σε συναρτήσεις πυκνότητας πιθανότητας ή συναρτήσεις πιθανότητας.

- **Περιπτώσεις δύο κλάσεων**

Στην πιο απλή περίπτωση θεωρούμε μια γραμμική συνάρτηση του διανύσματος εισόδου ώστε:

$$y(x) = w^T x + w_0$$

όπου $w = [w_1, w_2, \dots, w_n]^T$ καλείτε διάνυσμα βαρών (weight vector) και w_0 είναι ένα κατώφλι.

Ένα διάνυσμα εισόδου x ανατίθεται στην κλάση C1 αν $y(x) \geq 0$ αλλιώς ανατίθεται στην κλάση C2. Το αντίστοιχο όριο απόφασης καθορίζεται δηλαδή από την σχέση $y(x) = 0$.

Υποθέτοντας δύο σημεία x_1 και x_2 τα οποία βρίσκονται στο επίπεδο απόφασης και επειδή $y(x_1) = y(x_2) = 0$ προκύπτει ότι $w^T(x_1 - x_2) = 0$ και κατά συνέπεια το διάνυσμα w είναι ορθογώνιο σε κάθε διάνυσμα που βρίσκεται μέσα στο επίπεδο απόφασης, και το w καθορίζει το όριο (γραμμή) απόφασης (επιλογή κλάσης). Παρομοίως, δεδομένου ενός σημείου x στο επίπεδο απόφασης παίρνουμε $y(x) = 0$, και έτσι η απόσταση από την προέλευση της επιφάνειας απόφασης είναι :

$$\frac{w^T x}{\|w\|} = -\frac{w_0}{\|w\|}.$$

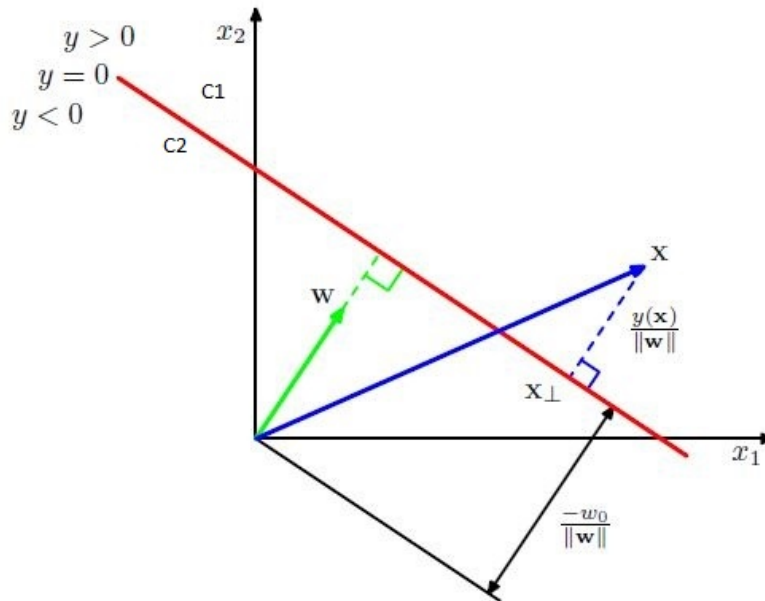
Επομένως φαίνεται ότι η παράμετρος bias w_0 καθορίζει τη θέση της επιφάνειας απόφασης, ενώ όταν η παράμετρος αυτή είναι ίση με το μηδέν το υπερεπίπεδο διέρχεται από την αρχή των αξόνων. Επιπλέον, παρατηρούμε ότι η τιμή του $y(x)$ δίνει ένα μέτρο (signed measure) της κάθετης απόστασης r του σημείου x από την επιφάνεια απόφασης. Ας υποθέσουμε σημείο x και έστω ότι το x_{\perp} είναι η ορθογώνια προβολή του στην επιφάνεια απόφασης, έτσι ώστε

$$x = x_{\perp} + r \frac{w}{\|w\|}$$

Πολλαπλασιάζοντας και τα δύο μέλη με w^T , προσθέτοντας w_0 και κάνοντας χρήση των σχέσεων $y(x) = w^T x + w_0$ και $y(x_{\perp}) = w^T x_{\perp} + w_0 = 0$ παίρνουμε

$$r = \frac{y(\mathbf{x})}{\|\mathbf{w}\|}$$

μια γεωμετρική απεικόνιση των οποίων φαίνεται παρακάτω.



Σχήμα 1.4. Απεικόνιση της γεωμετρίας δυο διαχωρίσιμων κλάσεων. Η επιφάνεια απόφασης φαίνεται με κόκκινο χρώμα , είναι κάθετη ως προς το w . [6]

Στην μια πλευρά όπου η $y(x)$ λαμβάνει θετικές τιμές έχουμε τα δεδομένα που θα ανήκουν στην πρώτη κλάση ενώ όταν η $y(x)$ παίρνει αρνητικές τιμές έχουμε τα δεδομένα που ανήκουν στην δεύτερη κλάση. Στην περίπτωση όπου η $y(x)=0$ τα δεδομένα μας θα ανήκουν ακριβώς πάνω στην διαχωριστική επιφάνεια. Η ορθογώνια απόσταση ενός τυχαίου σημείου x από την γραμμή απόφασης δίνεται από $\frac{y(x)}{\|\mathbf{w}\|}$

- **Περιπτώσεις πολλαπλών κλάσεων**

Η προηγούμενη μέθοδος που περιγράψαμε αφορά την ταξινόμηση των δεδομένων μας σε μόνο δύο κλάσεις. Στα περισσότερα πραγματικά προβλήματα οι κλάσεις είναι συνήθως πολύ περισσότερες. Αξίζει να σημειωθεί ότι όσες και να είναι οι κλάσεις μας κάθε χαρακτηριστικό ανήκει σε μόνο μια κλάση , ενώ δεν θα πρέπει να συγχέεται με την ταξινόμηση πολλαπλών ετικετών (multi-label classification) , στην οποία πολλαπλές ετικέτες μπορούν να αποδοθούν σε ένα χαρακτηριστικό.

Μία μέθοδος για επίλυση προβλημάτων πολλαπλών κλάσεων είναι ο διαχωρισμός του συνόλου δεδομένων ταξινόμησης πολλαπλών κατηγοριών σε πολλαπλά σύνολα δεδομένων δυαδικής ταξινόμησης (δύο κλάσεων) και η προσαρμογή ενός μοντέλου δυαδικής ταξινόμησης σε κάθε

ένα , έχοντας ως πιο γνωστές τις προσεγγίσεις One-vs-Rest και One-vs-One. Στην πρώτη περίπτωση έχουμε τον διαχωρισμό του συνόλου δεδομένων πολλαπλών κατηγοριών σε πολλαπλά προβλήματα δυαδικής ταξινόμησης , ενώ ένας δυαδικός ταξινομητής εκπαιδεύεται σε κάθε πρόβλημα δυαδικής ταξινόμησης και οι προβλέψεις γίνονται χρησιμοποιώντας το μοντέλο που είναι καλύτερο. Σε αντίθεση με αυτό το μοντέλο η τεχνική One-vs-One χωρίζει τα δεδομένων σε ένα σύνολο δεδομένων για κάθε κατηγορία έναντι κάθε άλλης κλάσης.

Στην πρώτη περίπτωση απαιτούνται $K-1$ ταξινομητές ενώ στην δεύτερη $K(K-1)/2$ [6] , ενώ η τεχνική έχει αυξημένη πολυπλοκότητα αν το σύνολο δεδομένων είναι πολύ μεγάλο.

Στην πιο απλή περίπτωση μπορούμε να θεωρήσουμε K κλάσεις που περιλαμβάνουν γραμμικές συναρτήσεις της μορφής :

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0}$$

ένα χαρακτηριστικό ταξινομείται στην κλάση C_k αν $y_k(x) > y_j(x)$ για κάθε $j \neq k$. Η διαχωριστική επιφάνεια δίνεται πλέον από $y_k(x) = y_j(x)$.[6]

1.2.2.1 Λογιστική παλινδρόμηση

Η λογιστική παλινδρόμηση αποτελεί τεχνική γραμμικής ταξινόμησης ενώ μοιάζει με αυτή της γραμμικής παλινδρόμησης με την διαφορά ότι σε αυτή την τεχνική οι τιμές είναι διακριτές (κλάσεις). Η λογιστική παλινδρόμηση είναι ένας αλγόριθμος προγνωστικής ανάλυσης και βασίζεται στην έννοια της πιθανότητας , ενώ η ονομασία της προέρχεται από την πολυπλοκότητα της συνάρτησης που χρησιμοποιεί η οποία είναι η σιγμοειδής [11] γνωστή και ως ‘λογική συνάρτηση’ [10]. Σε αντίθεση με τη γραμμική παλινδρόμηση στην οποία προσαρμόζουμε μια ευθεία γραμμή στα δεδομένα , στη λογιστική παλινδρόμηση προσαρμόζουμε συνήθως ένα ‘S’ (από την σιγμοειδή συνάρτηση) , για αυτό χρησιμοποιείται κυρίως για την περίπτωση δύο κλάσεων καθώς οι τιμές της συνάρτησης είναι μεταξύ 0 και 1.

Η λογιστική παλινδρόμηση είναι μια γραμμική τεχνική ταξινόμησης , έτσι χρησιμοποιείται μια γραμμική συνάρτηση της μορφής :

$$f(x) = b_0 + b_1x_1 + \dots + b_nx_n$$

Οι μεταβλητές b_0, b_1, \dots, b_n είναι οι εκτιμητές των συντελεστών παλινδρόμησης, οι οποίοι καλούνται επίσης βάρη πρόβλεψης (predicted weights) ή απλά συντελεστές.

Η συνάρτηση λογιστικής παλινδρόμησης $p(\mathbf{x})$ είναι η σιγμοειδής συνάρτηση

$$f(\mathbf{x}): p(\mathbf{x}) = 1 / (1 + \exp(-f(\mathbf{x}))).$$

η οποία συχνά καλείται ως η προβλεπόμενη πιθανότητα ή έξοδος για ένα δεδομένο να ισούται

με 1. Επομένως, το $1 - p(x)$ είναι η πιθανότητα η έξοδος είναι 0. Η λογιστική παλινδρόμηση καθορίζει τα καλύτερα προβλεπόμενα βάρη b_0, b_1, \dots, b_n έτσι ώστε η συνάρτηση $p(x)$ να είναι όσο το δυνατόν πιο κοντά σε όλες τις πραγματικές αποκρίσεις $y_i, i = 1, \dots, n$, όπου n είναι ο αριθμός των παρατηρήσεων (εκπαίδευση μοντέλου). Έτσι, για προβλήματα ταξινόμησης (δύο κλάσεων 0 ή 1) μπορούμε να ορίζουμε την παρακάτω εξίσωση η οποία προκύπτει από τον συνδυασμό των δυο προηγούμενων ως εξής :

$$P(f(x)) = \frac{1}{1 + \exp(-(b_0 + b_1 x_1 + \dots + b_n x_n))}, [0,1]$$

Η συνάρτηση κόστους βασισμένη στο εκτιμητή maximum likelihood (γνωστό και ως log-likelihood function-LLF λόγω χρήσης του φυσικού λογαρίθμου) λαμβάνει χώρα, ώστε να βελτιστοποιηθούν οι παράμετροι-βάρη και κατά συνέπεια ο αλγόριθμος, από την εξίσωση :

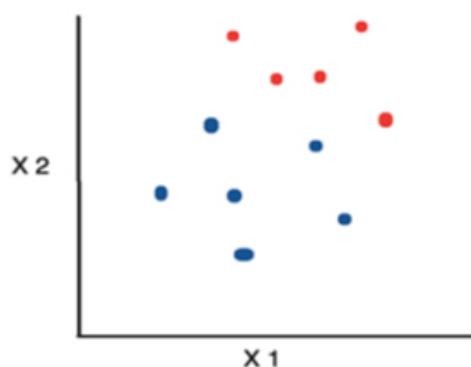
$$LLF = \sum_i (y_i \log(p(x_i)) + (1 - y_i) \log(1 - p(x_i))) . [12][13]$$

(Υπάρχουν αρκετές μαθηματικές προσεγγίσεις που μπορούν υπολογίσουν τα καλύτερα βάρη που αντιστοιχούν στο μέγιστο LLF, ο αναγνώστης μπορεί να τις αναζητήσει στην βιβλιογραφία [1][13][14][15][16][17][18]).

Όταν $y_i = 0$, το LLF για την αντίστοιχη παρατήρηση ισούται με το $\log(1 - p(x_i))$ ενώ αν το $p(x_i)$ είναι κοντά στο $y_i = 0$, τότε $\log(1 - p(x_i))$ είναι κοντά στο 0. Εάν το $p(x_i)$ απέχει πολύ από το 0, τότε το $\log(1 - p(x_i))$ μειώνεται σημαντικά έχοντας ως αποτέλεσμα μη αποδεκτό αποτέλεσμα καθώς ο σκοπός είναι η αποκτήσει μέγιστο LLF. Από την άλλη πλευρά, όταν $y_i = 1$, το LLF για αυτήν την παρατήρηση είναι $y_i \log(p(x_i))$. Εάν το $p(x_i)$ είναι κοντά στο $y_i = 1$, τότε $\log(p(x_i))$ είναι κοντά στο 0. Εάν το $p(x_i)$ απέχει πολύ από το 1, τότε $\log(p(x_i))$ είναι ένας μεγάλος αρνητικός αριθμός.

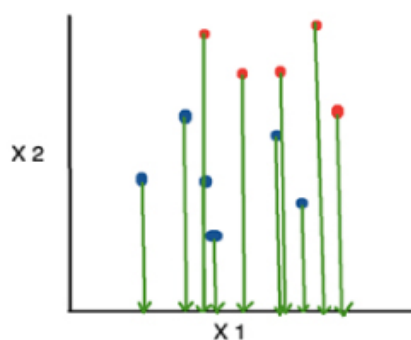
1.2.2.2 Γραμμική διακριτική ανάλυση (Linear discriminant analysis-LDA)

Η γραμμική διακριτική ανάλυση ή η ανάλυση διακριτικής λειτουργίας είναι μια τεχνική μείωσης διαστάσεων που μπορεί να χρησιμοποιηθεί και για τα προβλήματα ταξινόμησης. Κύριος στόχος της είναι η μοντελοποίηση διαφορετικών χαρακτηριστικών σε ομάδες, δηλαδή διαχωρισμό δύο ή περισσότερων κλάσεων. Ένα από τα πλεονεκτήματα της είναι η προβολή των χαρακτηριστικών από ένα χώρο υψηλότερης διάστασης σε χώρο χαμηλότερης διάστασης, έχοντας ως αποτέλεσμα να μπορεί να χρησιμοποιηθεί ακόμα και σε περιπτώσεις μη διαχωρίσιμων κλάσεων ξεπερνώντας έτσι τον περιορισμό που θέτει η λογιστική παλινδρόμηση. Βασικός στόχος της είναι η τροποποίηση των δεδομένων ώστε να μεγιστοποιηθεί η απόσταση μεταξύ των κλάσεων αλλά και να ελαχιστοποιηθεί η διασπορά εντός των κλάσεων ώστε τα δεδομένα κάθε κλάσης να είναι συγκεντρωμένα γύρω από τη μέση τιμή τους. Υποθέτουμε ότι έχουμε δύο ομάδες σημείων δεδομένων που ανήκουν σε δύο διαφορετικές κατηγορίες που θέλουμε να τα ταξινομήσουμε. Όπως παρουσιάζεται στο επόμενο σχήμα, για τον διαχωρισμό των δεδομένων δεν υπάρχει κάποια ευθεία γραμμή ώστε να μπορεί να υπάρξει ξεκάθαρος διαχωρισμός μεταξύ μπλε και κόκκινων σημείων.



Σχήμα 1.5. Γράφημα (2D) με τα δεδομένα να μην μπορούν να ταξινομηθούν πλήρως [22]

Κάνοντας χρήση της τεχνικής LDA μπορεί να μειωθεί το γράφημα (από 2D σε 1D) προκειμένου να μεγιστοποιηθεί η διαχωριστικότητα μεταξύ των δύο κατηγοριών. Κάνοντας χρήση και των δύο αξόνων για να δημιουργήσει έναν νέο άξονα, ώστε να προβάλλει τα δεδομένα σε έναν νέο άξονα με τέτοιο τρόπο ώστε να μεγιστοποιήσει τον διαχωρισμό των δύο κατηγοριών και ως εκ τούτου, μειώνοντας τη διάσταση του γραφήματος. Σε αυτή την περίπτωση μπορούμε να προβάλλουμε κάθε χαρακτηριστικό στο άξονα x_1 .

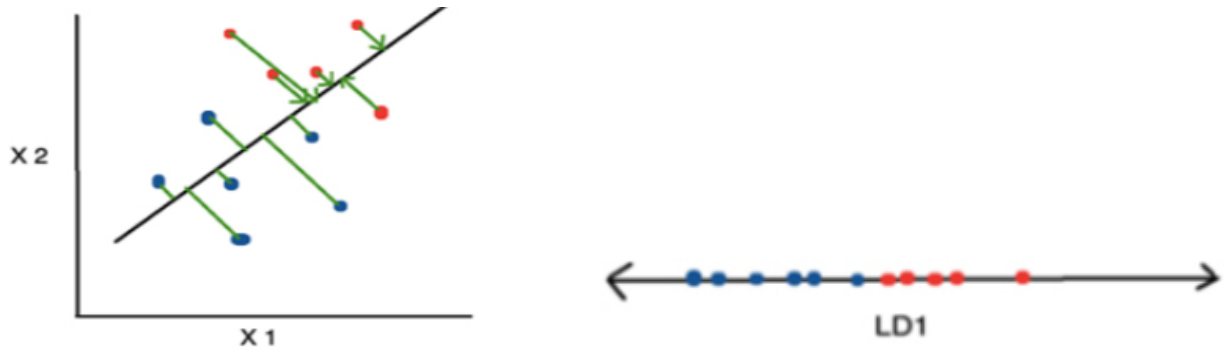


Σχήμα 1.6 Προβολή δεδομένων στον άξονα x_1 [21]

Δύο κριτήρια χρησιμοποιούνται από την LDA για τη δημιουργία ενός νέου άξονα: [21]

1. Μεγιστοποιήστε την απόσταση μεταξύ των μέσων των δύο κατηγοριών.
2. Ελαχιστοποιήστε την διακύμανση σε κάθε κατηγορία

Στο παραπάνω γράφημα, μπορεί να φανεί ότι ένας νέος άξονας δημιουργείται και σχεδιάζεται στο γράφημα 2D έτσι ώστε να ικανοποιούνται τα παραπάνω κριτήρια, ο άξονας δηλαδή που δημιουργήθηκε πρόσφατα αυξάνει το διαχωρισμό μεταξύ των σημείων των δύο κατηγοριών. Μετά την εφαρμογή των κριτηρίων και την μείωση της διάστασης τα δεδομένα πλέον βρίσκονται στον νέο άξονα που είναι εμφανώς διαχωρίσιμα.



Σχήμα 1.7. Μεγιστοποίηση μεταξύ των σημείων (αριστερά) , μείωση διαστάσεων και διαχωρισμών δεδομένων (δεξιά) [22]

Από μαθηματικής πλευράς :

1. Αρχικά, υπολογίζεται η διαχωριστικότητα μεταξύ των τάξεων που είναι η απόσταση μεταξύ του μέσου όρου των διαφορετικών τάξεων (διακύμανση μεταξύ κλάσης) :

$$S_b = \sum_{i=1}^g N_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T$$

2. Στη συνέχεια ,υπολογίζεται η απόσταση μεταξύ του μέσου και του δείγματος κάθε τάξης , (διακύμανση εντός της τάξης) :

$$S_w = \sum_{i=1}^g (N_i - 1) S_i = \sum_{i=1}^g \sum_{j=1}^{N_i} (x_{ij} - \bar{x}_i)(x_{ij} - \bar{x}_i)^T$$

3. Τέλος, μειώνουμε τον χώρο διαστάσεων σύμφωνα με τα προηγούμενα. Το P είναι lower-dimensional space projection που ονομάζεται επίσης κριτήριο του Fisher[23].

$$P = \arg \max_p \frac{|P^T S_b P|}{|P^T S_w P|}$$

1.2.2.3 Μάθηση κατά Bayes

Οι μέθοδοι ταξινόμησης Naive Bayes είναι αλγόριθμοι μηχανικής μάθησης βασισμένοι στην εφαρμογή του θεωρήματος του Bayes [24] , βασίζονται δηλαδή σε στατιστικά στοιχεία (πιθανοτικές κατανομές) , υποθέτοντας ανεξαρτησία μεταξύ των χαρακτηριστικών , ενώ έχει εφαρμογή σε μεγάλη ποικιλία προβλημάτων. Το θεώρημα Bayes περιγράφει την πιθανότητα να συμβεί ένα γεγονός έχοντας λάβει υπόψη την προγενέστερη γνώση η οποία σχετίζεται με το υπάρχον γεγονός [25] [17]:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

όπου A και B είναι γεγονότα με $P(B) \neq 0$

$P(A)$, $P(B)$ είναι η πιθανότητα των A και B αντίστοιχα

$P(B|A) = \frac{P(B,A)}{P(A)}$ είναι η υπό συνθήκη πιθανότητα του B δοθέντος του A , με $P(B,A)$ να είναι η από κοινού πιθανότητα των δύο ενδεχομένων.

Ο όρος Naive 'αφελής' προήλθε από την υπόθεση της ανεξαρτησίας των χαρακτηριστικών. Με τον Naive Bayes απλοποιούμε του υπολογισμούς υπολογίζοντας την πιθανότητα υπό όρους για κάθε χαρακτηριστικό και στη συνέχεια τα πολλαπλασιάζουμε μαζί.[26]

Ας πάρουμε ένα παράδειγμα δύο κλάσεων k_1, k_2 αντίστοιχα. Θεωρούμε ότι η εκ των προτέρων πιθανότητες $P(k_1)$, και $P(k_2)$ είναι γνωστές , αλλά ακόμα και αν δεν είναι μπορούν να εκτιμηθούν από τα διαθέσιμα διανύσματα χαρακτηριστικών ως εξής :

$P(k_1) \approx N_1/N$ και $P(k_2) \approx N_2/N$ με N να είναι ο συνολικός αριθμός των χαρακτηριστικών και N_1, N_2 αυτά που ανήκουν στην κλάση k_1 και k_2 αντίστοιχα. Γνωστές έχουν θεωρηθεί επίσης οι υπό συνθήκη συναρτήσεις πυκνότητας των κλάσεων $p(x|k_1)$ και $p(x|k_2)$ οι οποίες περιγράφουν την κατανομή των διανυσμάτων χαρακτηριστικών σε κάθε μια από τις κλάσεις (μπορούν να υπολογιστούν αν δεν είναι γνωστές από τα δεδομένα εκπαίδευσης) . Τα διανύσματα χαρακτηριστικών μπορούν να πάρουν οποιαδήποτε τιμή στον 1-διάστατο χώρο ,στην περίπτωση που τα διανύσματα χαρακτηριστικών μπορούν να λάβουν μόνο διακριτές τιμές , οι συναρτήσεις πυκνότητας $p(x|k_i)$ γίνονται πιθανότητες ($P(x|k_i)$). Σύμφωνα με το κανόνα του Bayes :

$$P(k_i|x) = \frac{p(x|k_i)P(k_i)}{p(x)} , \text{ με } p(x) = \sum_{i=1}^2 p(x|k_i)P(k_i) \text{ (σ.π.π του } p(x))$$

Ο Κανόνας ταξινόμησης κατά Bayes μπορεί να οριστεί ως :

Αν $P(k_1|x) > P(k_2|x)$ τότε το x ανήκει στην κλάση k_1

Αν $P(k_1|x) < P(k_2|x)$ τότε το x ανήκει στην κλάση k_2

Στην ειδική περίπτωση της ισότητας το πρότυπο μπορεί να ταξινομηθεί σε οποιαδήποτε από τις δύο κλάσεις [17].

Γενικεύοντας για την ταξινόμηση Naive Bayes μπορούμε να πούμε :

Δοθέντος μια κλάσης y και ενός χαρακτηριστικού x_i (εξαρτώμενου) από το σύνολο χαρακτηριστικών x_n , από το θεώρημα Bayes έχουμε :

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n | y)}{P(x_1, \dots, x_n)}$$

κάνοντας χρήση της naive υπόθεσης ανεξαρτησίας :

$$P(x_i|y, x_1, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i|y) ,$$

για κάθε i η αρχική σχέση μπορεί να υλοποιηθεί ως εξής

$$P(y|x_1, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i|y)}{P(x_1, \dots, x_n)}$$

Καθώς το $P(x_1, \dots, x_n)$ είναι σταθερό δεδομένης της εισόδου μπορεί να χρησιμοποιηθεί ο ακόλουθος κανόνας ταξινόμησης :

$$P(y|x_1, \dots, x_n) \approx P(y) \prod_{i=1}^n P(x_i|y) \Rightarrow \hat{y} = \underset{p}{\operatorname{argmax}} P(y) \prod_{i=1}^n P(x_i|y)$$

Για να προβλέψουμε τη μεταβλητή απόκρισης ανάλογα με τις διαφορετικές τιμές που

λαμβάνονται, πρέπει να ληφθεί υπόψη μια πιθανή τιμή ή το μέγιστο των τιμών. Αυτή η μέθοδος είναι γνωστή ως MAP (Maximum A Posteriori) και μπορεί να χρησιμοποιηθεί για την εκτίμηση $P(y)$ και $P(x_i | y)$. Εφόσον έχουμε θεωρήσει τον παρανομαστή σταθερό και τον έχουμε αφαιρέσει από την εξίσωση ο υπολογισμός του MAP ανάγεται σε πρόβλημα μεγιστοποίησης πιθανοφάνειας (Maximizing Likelihood) [27][28].

Υπάρχουν διάφοροι αλγόριθμοι ταξινόμησης που βασίζονται στο μοντέλο naïve Bayes οι οποίοι διαφέρουν από τις υποθέσεις που κάνουν σχετικά με την κατανομή του $P(x_i | y)$, οι πιο γνωστοί είναι: [27]

1. Gaussian Naive Bayes. Ο αλγόριθμος αυτός υποθέτει την κατανομή των χαρακτηριστικών ως Gaussian:

$$P(x_i | y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

οι παράμετροι σ_y και μ_y υπολογίζονται με maximum likelihood.

2. Multinomial Naive Bayes

Η παραλλαγή αυτή χρησιμοποιείται για πολυμεσικά κατανεμημένα δεδομένα και είναι μία από τις δύο κλασικές παραλλαγές που χρησιμοποιούνται στην ταξινόμηση κειμένου όπου τα δεδομένα αντιπροσωπεύονται συνήθως ως μετρήσεις διανυσμάτων λέξεων.

3. Complement Naive Bayes (CNB)

Η παραλλαγή αυτή αποτελεί μια προσαρμογή του τυπικού αλγορίθμου multinomial naïve Bayes (MNB) και είναι κατάλληλος για μη ισορροπημένα δεδομένων. Ο CNB χρησιμοποιεί στατιστικά στοιχεία από το συμπλήρωμα κάθε τάξης για να υπολογίσει τα βάρη του μοντέλου. Οι εφευρέτες του CNB δείχνουν εμπειρικά ότι οι εκτιμήσεις των παραμέτρων για το CNB είναι πιο σταθερές από αυτές του MNB. Επιπλέον, το CNB ξεπερνά τακτικά το MNB (συχνά με σημαντικό περιθώριο) σε εργασίες ταξινόμησης κειμένου.

4. Bernoulli Naive Bayes

Χρησιμοποιείται για δεδομένα που ακολουθούν κατανομή Bernoulli (multivariate Bernoulli distributions), δηλαδή μπορεί να υπάρχουν πολλά χαρακτηριστικά αλλά το καθένα θεωρείται ότι είναι μεταβλητή δυαδικής αξίας (Bernoulli, boolean). Επομένως, αυτή η τάξη απαιτεί τα δείγματα να εκπροσωπούνται ως διανύσματα χαρακτηριστικών δυαδικής αξίας.

Ο κανόνας απόφασης βασίζεται στην παρακάτω εξίσωση

$$P(x_i | y) = P(i | y)x_i + (1 - P(i | y))(1 - x_i)$$

Στην περίπτωση ταξινόμησης κειμένου, μπορούν να χρησιμοποιηθούν διανύσματα εμφάνισης λέξεων (και όχι διανύσματα μέτρησης λέξεων), η εκπαίδευση και χρήση αυτού του ταξινομητή μπορεί να έχει καλύτερη απόδοση σε ορισμένα σύνολα δεδομένων, ειδικά σε αυτά με μικρότερα έγγραφα.

1.2.2.4 K-πλησιέστεροι γείτονες

Η μέθοδος των k πλησιέστερων γειτόνων ανήκει στην κατηγορία των μη παραμετρικών μεθόδων, δηλαδή δεν κάνει χρήση κάποιου μοντέλου έχοντα ως στόχο την ελαχιστοποίηση κάποια συνάρτησης [1]. Η μέθοδος μπορεί να χρησιμοποιηθεί τόσο για παλινδρόμηση όσο και για ταξινόμηση με συνηθέστερη την δεύτερη, ενώ μπορεί να χρησιμοποιηθεί και σε μη γραμμικά δεδομένα.

Η βασική λειτουργία του αλγορίθμου είναι η αποθήκευση των δεδομένων εκπαίδευσης και η σύγκριση τους με το νέο δεδομένο προς ταξινόμηση, για αυτό το λόγο ο αλγόριθμος έχει ονομαστεί και ως 'τεμπέλης'. [29][30]

Οι δύο κατηγορίες που μπορεί να διακριθούν είναι: [31]

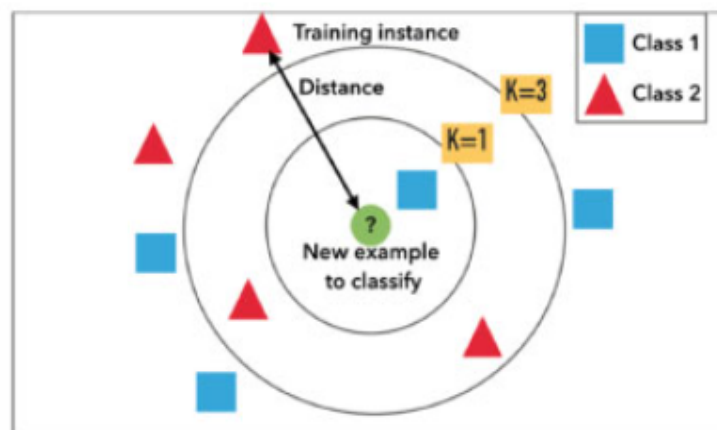
1. k πλησιέστεροι γείτονες, όπου ο αλγόριθμος λειτουργεί ανιχνεύοντας τους k πλησιέστερους γείτονες $k > 0$
2. πλησιέστεροι γείτονες ακτίνας r , όπου ο αλγόριθμος εφαρμόζεται σε μια σταθερή ακτίνα $r > 0$

Η λειτουργία του αλγορίθμου βασίζεται στην εξής απλή μέθοδο [1]:

Έστω ότι έχουμε δύο κλάσεις k_1 και k_2 και θέλουμε να ταξινομήσουμε το νέο χαρακτηριστικό x . Παρατηρώντας τα πιο κοντινά πρότυπα στο x μπορούμε να συμπεράνουμε ότι:

- Αν τα περισσότερα κοντινότερα χαρακτηριστικά στο x ανήκουν στην κλάση k_1 τότε το x ανήκει στην κλάση k_1
- Αν τα περισσότερα κοντινότερα χαρακτηριστικά στο x ανήκουν στην κλάση k_2 τότε το x ανήκει στην κλάση k_2

Η μέθοδος ονομάζεται k πλησιέστεροι γείτονες καθώς πρέπει να επιλεγεί μια τιμή για το k η οποία ορίζει πόσα κοντινά από τα ήδη υπάρχοντα χαρακτηριστικά θα πρέπει να εξετάσει ο αλγόριθμος ώστε να αποφασίσει σε ποια κλάση ανήκει το νέο στοιχείο. Η κλάση με το μεγαλύτερο αριθμό χαρακτηριστικών είναι τελικά αυτή που θα επιλεγεί, ενώ για την αποφυγή ισοψηφία το k επιλέγεται να είναι περιττός αριθμός. [1] Η επιλογή του αριθμού k αποτελεί αρκετά κρίσιμη επιλογή καθώς μπορεί να οδηγήσει σε λάθος συμπεράσματα, ας το δούμε στο επόμενο παράδειγμα. Έστω ότι έχουμε το σενάριο που παρουσιάζεται στο επόμενο σχήμα:



Σχήμα 1.8. Παράδειγμα σεναρίου k πλησιέστερου γείτονα [29]

Έστω ότι θέλουμε να ταξινομήσουμε το νέο χαρακτηριστικό (πράσινο). Αν επιλέξουμε ως $k=1$ το χαρακτηριστικό ανήκει στην κλάση 1 (μπλε τετράγωνο) ενώ εάν επιλέξουμε ως $k=3$ τότε ανήκει στην κλάση 2 (κόκκινο τρίγωνο). Μη συνηθισμένη επιλογή του k αποτελεί η τιμή $k=5$. Πως όμως θα υπολογίσουμε ποιο χαρακτηριστικό ανήκει πιο κοντά; Η πιο συνηθισμένη μέθοδος υπολογισμού της απόστασης του νέου χαρακτηριστικού μας με τα υπάρχοντα αποτελεί η Ευκλείδεια απόσταση ενώ η απόσταση Manhattan και Minkowski μπορούν να χρησιμοποιηθούν ως εναλλακτικές, ενώ για κατηγορηματικές μεταβλητές μπορεί να χρησιμοποιηθεί η απόσταση Hamming [29].

Αλγόριθμοι k πλησιέστερου γείτονα :[31]

1. Αλγόριθμος Brute Force

Ο αλγόριθμος περιλαμβάνει τον υπολογισμό των αποστάσεων μεταξύ όλων των ζευγών σημείων στο σύνολο δεδομένων: Για N δείγματα σε D διαστάσεις, αυτή η προσέγγιση θα έχει πολυπλοκότητα $O[DN^2]$. Οι αναζητήσεις brute-force μπορεί να είναι πολύ ανταγωνιστικές και αποτελεσματικές για μικρά δείγματα δεδομένων. Ωστόσο, καθώς N αυξάνεται, η προσέγγιση brute-force καθίσταται γρήγορα ανέφικτη καθώς αυξάνεται πολύ το υπολογιστικό κόστος.

2. K-D Tree

Για να αντιμετωπιστεί η μεγάλη υπολογιστική πολυπλοκότητα της προσέγγισης brute-force, εφευρέθηκε μια ποικιλία δομών δεδομένων που βασίζονται σε δέντρα, έχοντας ως στόχο να μειώσουν τον απαιτούμενο αριθμό υπολογισμών απόστασης κωδικοποιώντας αποτελεσματικά τις πληροφορίες συνολικής απόστασης για το δείγμα. Η βασική ιδέα είναι ότι εάν το σημείο A είναι πολύ μακρινό από το σημείο B και το σημείο B είναι πολύ κοντά στο σημείο C , τότε γνωρίζουμε ότι τα σημεία A και C είναι πολύ μακριά, χωρίς να χρειάζεται να υπολογίσουμε ρητά την απόστασή τους. Με αυτόν τον τρόπο, το υπολογιστικό κόστος μιας αναζήτησης πλησιέστερων γειτόνων μπορεί να βελτιωθεί $O[DN \log(N)]$. Για την εκμετάλλευση αυτού του πλεονεκτήματος αρχικά χρησιμοποιήθηκαν τα δέντρα K-D (*K-dimensional tree*). Το δέντρο KD είναι μια δυαδική δομή δέντρου που χωρίζει διαδοχικά τον χώρο παραμέτρων κατά μήκος των αξόνων δεδομένων, χωρίζοντάς τον σε ένθετες ορθοτοπικές περιοχές στις οποίες αρχειοθετούνται σημεία δεδομένων. Η κατασκευή του είναι αρκετά γρήγορη ενώ η πολυπλοκότητα του μπορεί να προσδιοριστεί ως $O[\log(N)]$.

3. Ball Tree

Η προηγούμενη μέθοδος είναι αναποτελεσματική σε υψηλότερες διαστάσεις. για αυτό κρίθηκε απαραίτητο η χρήση των δέντρων-μπάλα. Ένα δέντρο μπάλα διαιρεί αναδρομικά τα δεδομένα σε κόμβους που ορίζονται από ένα κεντροειδές C και ακτίνας r , ώστε κάθε σημείο του κόμβου να βρίσκεται εντός της υπερ-σφαίρας που ορίζεται από r και C . Ο αριθμός των πόντων για αναζήτηση γειτόνων μειώνεται μέσω της χρήσης της *ανισότητας του τριγώνου* :

$$|x + y| \leq |x| + |y|$$

με αυτή την ρύθμιση, ένας υπολογισμός απόστασης μεταξύ ενός χαρακτηριστικού προς αναζήτηση και του κεντροειδούς είναι επαρκής για τον προσδιορισμό ενός κατώτερου και ανώτερου ορίου στην απόσταση από όλα τα σημεία εντός του κόμβου.

Πλεονεκτήματα : [29][30]

- Είναι σχετικά γρηγορότερος από άλλους αλγορίθμους
- Δεν απαιτεί την εκπαίδευση ενός μοντέλου με στόχο την γενίκευση
- Σχετικά υψηλή ακρίβεια
- Μπορεί να χρησιμοποιηθεί τόσο σε γραμμικά όσο και σε μη γραμμικά δεδομένα
- Έχει ανοχή σε δεδομένα με θόρυβο
- Χρησιμοποιείται για ταξινόμηση αλλά και παλινδρόμηση

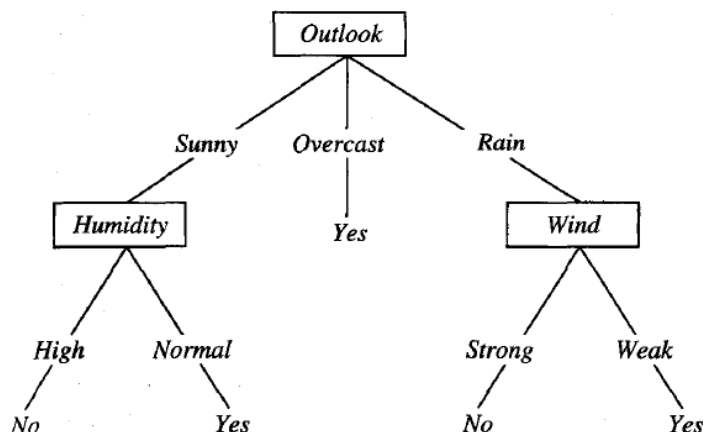
Μειονεκτήματα : [29][30]

- Η εκπαίδευση των δεδομένων είναι αργή σε σύγκριση το χρόνο και την μνήμη
- Χρειάζεται αρκετή μνήμη για την αποθήκευση του συνόλου εκπαίδευσης
- Πρέπει να λαμβάνεται υπόψη η επιλογή του κ
- Το κόστος υπολογισμού είναι υψηλό λόγω του υπολογισμού της απόστασης μεταξύ των σημείων δεδομένων και των δειγμάτων εκπαίδευσης.
- Δεν είναι κατάλληλο για μεγάλο αριθμό δεδομένων.

1.2.2.5 Δέντρα Αποφάσεων

Τα δέντρα αποφάσεων αποτελούν άλλη μια μέθοδος μη παραμετρικής μάθησης που μπορεί να χρησιμοποιηθούν τόσο για παλινδρόμηση όσο και για ταξινόμηση, [32] και για αυτό έχει επικρατήσει η ονομασία CART-Classification and Regression Trees. Η λογική πίσω από τα δέντρα αποφάσεων είναι να δημιουργηθούν ομοιογενείς περιοχές (homogeneous) [1], δηλαδή να γίνει κατάτμηση του χώρου των δεδομένων σε ορθογώνιες περιοχές έτσι ώστε η κάθε περιοχή να αποτελείται από πρότυπα μια κλάσης, ενώ έχουν σκοπό να δημιουργηθεί ένα μοντέλο που προβλέπει την αξία μιας μεταβλητής στόχου μαθαίνοντας απλούς κανόνες απόφασης που προέρχονται από τα δεδομένα. Η ονομασία δέντρο προήλθε από την δομή που κατασκευάζει αυτή η μέθοδος η οποία μοιάζει με δέντρο με τον πρώτο κόμβο (κορυφή) να ονομάζεται ρίζα, και τους υπόλοιπους φύλλα ή παιδιά, τα οποία αποτελούν κατηγορίες ταξινόμησης. Η διαδρομή από κάποιο κόμβο προς κάποιο φύλλο ονομάζεται μονοπάτι, ενώ η διαδρομή από την ρίζα του δέντρου προς το τελευταίο φύλλο καθορίζει το βάθος του δέντρου, και κατά

συνέπεια την πολυπλοκότητα του. Σε κάθε κόμβο του δέντρου γίνεται προσπάθεια επιπλέον διαχωρισμού των δειγμάτων που δεν έχουν κατηγοριοποιηθεί, έχοντας ως τελικό στόχο των διαχωρισμό όλων των δειγμάτων ώστε στα φύλλα να υπάρχουν χαρακτηριστικά από την ίδια κατηγορία (ομοιογένεια δεδομένων), και να μπορεί να γίνει η εξαγωγή ενός συμπεράσματος.



Σχήμα 1.9. Παράδειγμα ταξινόμησης δέντρου αποφάσεων, που επιστρέφει την απόφαση που σχετίζεται με το φύλλο (εδώ Yes ή No) [33]

Ο διαχωρισμός του χώρου ώστε να επιτευχθεί η ομοιογένεια αποτελεί την καρδιά του αλγορίθμου, ενώ αν μια περιοχή είναι ήδη ομοιογενής δεν χρειάζεται να γίνει κάποια ενέργεια. Αν μια περιοχή όμως δεν είναι ομοιογενής, τότε επιλέγουμε μια διάσταση j του χώρου των χαρακτηριστικών και διαιρούμε την περιοχή σε δυο υποπεριοχές, $x_j < s$ και $x_j > s$ με το s να αποτελεί κάποιο κατώφλι. Το κατώφλι s και το χαρακτηριστικό x_j επιλέγεται ώστε να οδηγεί στην μεγαλύτερη μείωση της ανομοιογένειας, ενώ απαιτούνται κάποια κριτήρια ώστε αυτό να βγει επιτυχής. Τα πιο συνηθισμένα κριτήρια μέτρησης της ανομοιογένειας δεδομένης μια περιοχής R που περιέχει N πρότυπα και K κλάσεις είναι: [1][33]

- Εντροπία

$$E_{ent}(R) = - \sum_{k=1}^K P(k) \log_2 P(k)$$

όπου $P(k)$ είναι το ποσοστό των προτύπων του R που ανήκουν στην κλάση K :

$$P(k) = \frac{1}{N} \sum_{i=1}^N I(t_i = k)$$

- Κόστος Gini

$$E_{Gini}(R) = \sum_{k \neq l} P(k)P(l)$$

Αφού διαιρέσουμε την περιοχή R σε δυο υποπεριοχές R_1 και R_2 με N_1 και N_2 να είναι τα αντίστοιχα πρότυπα, η ανομοιογένεια υπολογίζεται ως:

$$E_{split} = \frac{N_1}{N} E(R_1) + \frac{N_2}{N} E(R_2)$$

Αλγόριθμοι Δέντρων Αποφάσεων: [32]

- ID3
Ο αλγόριθμος αυτός ανήκει στην κατηγορία των άπληστων αλγορίθμων , δημιουργώντας δέντρο πολλαπλών διαδρομών , εντοπίζοντας για κάθε κόμβο το κατηγορηματικό χαρακτηριστικό που θα αποφέρει το μεγαλύτερο κέρδος πληροφοριών για κατηγορικούς στόχους. Το δέντρο μεγαλώνει στο μέγιστο βάθος τους και στη συνέχεια εφαρμόζεται ένα βήμα κλαδέματος για τη βελτίωση της ικανότητας του δέντρου να γενικεύει σε μη ορατά δεδομένα.
- C4.5
Ο αλγόριθμος αυτός αποτελεί μια εξέλιξη του αλγορίθμου ID3 , και αφαίρεσε τον περιορισμό ότι τα χαρακτηριστικά πρέπει να είναι κατηγορηματικά ορίζοντας δυναμικά ένα διακριτό χαρακτηριστικό (βάσει αριθμητικών μεταβλητών) που χωρίζει την συνεχή τιμή χαρακτηριστικού σε ένα διακριτό σύνολο διαστημάτων. Η τεχνική αυτή μετατρέπει τα δέντρα σε σύνολα κανόνων if-then. Αυτή η ακρίβεια κάθε κανόνα στη συνέχεια αξιολογείται για να προσδιοριστεί η σειρά με την οποία πρέπει να εφαρμοστούν. Το κλάδεμα γίνεται με την κατάργηση της προϋπόθεσης ενός κανόνα εάν η ακρίβεια του κανόνα βελτιωθεί χωρίς αυτήν.
- C5.0
Ο αλγόριθμος αυτός είναι μια παραλλαγή του C4.5 υπό την έννοια ότι χρησιμοποιεί λιγότερη μνήμη , κάνοντας χρήση μικρότερων συνόλων ενώ έχει βελτιωμένη ακρίβεια.
- CART
Η εκδοχή αυτού του αλγορίθμου μοιάζει πολύ με τον C4.5, αλλά διαφέρει στο ότι υποστηρίζει αριθμητικές μεταβλητές στόχου (παλινδρόμηση) και δεν υπολογίζει σύνολα κανόνων. Το CART κατασκευάζει δυαδικά δέντρα χρησιμοποιώντας τη δυνατότητα και το όριο που αποδίδει το μεγαλύτερο κέρδος πληροφοριών σε κάθε κόμβο.

Πλεονέκτημα :[1][34][35][36]

- Χειρισμό γραμμικών και μη γραμμικών δεδομένων
- Επεξεργασία αριθμητικών και κατηγορηματικών δεδομένων
- Εύκολο στην οπτικοποίηση
- Εύκολη στην χρήση χωρίς πολύπλοκους μαθηματικούς τύπους
- Είναι πολυ ευπροσάρμοστος στα δεδομένα-πρόβλημα σε σύγκριση με άλλους αλγορίθμους
- Ανθεκτικό σε ανωμαλίες δεδομένων: Ενώ οι περισσότεροι αλγόριθμοι μηχανικής μάθησης αποτυγχάνουν ακόμη και με ένα μικρό σύνολο ανωμαλιών, τα Δέντρα Απόφασης είναι πολύ ανθεκτικά και είναι σε θέση να χειριστούν ένα αρκετά μεγάλο ποσοστό τέτοιων ανωμαλιών αρκετά καλά χωρίς να αλλάξουν τα αποτελέσματα
- Δεν απαιτούν μεγάλη προετοιμασία δεδομένων
- Μπορούν να χειριστούν μεγάλο όγκο δεδομένων

Μειονεκτήματα :[1][34][35][36]

- Είναι επιρρεπή στην υπερμοντελοποίηση, δηλαδή μαθαίνουν τόσο καλά τα δεδομένα αλλά δεν αποδίδουν στα δεδομένα δοκιμής
- Είναι επιρρεπής στην υπο μοντελοποίηση , δηλαδή τα μοντέλο είναι πολύ απλό ώστε ο αλγόριθμος να μάθει και να είναι αποτελεσματικός στα δεδομένα
- Η εύρεση του βέλτιστου δέντρου ανήκει στην κατηγορία Np-πλήρη
- Ευαισθησία, από την άποψη ότι αν τα δεδομένα δεν είναι ισορροπημένα δηλαδή μια κλάση έχει περισσότερα δεδομένα από κάποια άλλη τότε δημιουργείται δέντρο με τα δεδομένα να ταιριάζουν στην μεγαλύτερη κλάση σε βάρος της μικρής
- Οι άπληστοι αλγόριθμοι δεν εγγυώνται την παραγωγή του βέλτιστου δέντρου αποφάσεων , ενώ σε μεγάλα δεδομένα αυξάνουν την πολυπλοκότητα και την υπολογιστική ισχύ.

1.2.2.6 Μηχανές Διανυσμάτων υποστήριξης - Support Vector Machines (SVM)

Οι μηχανές διανυσμάτων υποστήριξης αποτελούν την πιο δημοφιλή μέθοδος για προβλήματα μάθησης με επίβλεψη ,ενώ μπορούν να χρησιμοποιηθούν για ταξινόμηση , παλινδρόμηση αλλά και για τον υπολογισμό ακραίων τιμών , τόσο σε γραμμικά όσο και σε μη γραμμικά προβλήματα. Βασικό τους πλεονέκτημα αποτελεί η μέριμνα ώστε ο διαχωρισμός των κλάσεων να είναι ο καλύτερος δυνατός όσον αφορά την ικανότητα γενικεύσεις.[1].Ο στόχος του αλγορίθμου SVM είναι να δημιουργήσει το καλύτερο όριο γραμμής ή απόφασης που μπορεί να διαχωρίσει τον n-διάστατο χώρο σε τάξεις (υπερπλάνο), ώστε να μπορούμε εύκολα να βάλουμε το νέο σημείο δεδομένων στη σωστή κατηγορία.

Το πόσο καλός θα είναι ο διαχωρισμός που θα επιτύχει το μοντέλο σε μελλοντικά πρότυπα ελέγχου εξαρτάται από την απόσταση των προτύπων από τη διαχωριστική επιφάνεια.Όσο πιο μικρή είναι η απόσταση τόσο πιο μεγάλη είναι η πιθανότητα ένα μελλοντικό πρότυπο ελέγχου να βρεθεί σε λάθος κλάση. Η μικρότερη απόσταση που αντιστοιχεί στο πλησιέστερο πρότυπο λέγεται περιθώριο(margin) [1].

Η τεχνική SVM (όπως και άλλες , λογιστική παλινδρόμηση , Perceptron) έχουν σχεδιαστεί ώστε να χρησιμοποιούνται για δυαδική ταξινόμηση , δηλαδή για προβλήματα που έχουμε μόνο δύο κλάσεις και όχι περισσότερες. Ωστόσο έχουν επινοηθεί διάφορες τεχνικές ώστε η μέθοδος αυτή να μπορεί να επεκταθεί και στον διαχωρισμό περισσότερων κλάσεων. Οι δυο πιο γνωστές τεχνικές είναι :[162][163]

1. One vs rest : χωρίζει μια ταξινόμηση πολλαπλών κλάσεων σε ένα πρόβλημα δυαδικής ταξινόμησης ανά τάξη , δηλαδή N διαφορετικοί ταξινομητές κατασκευάζονται ένας ταξινομητής για κάθε κλάση.Ένα πιθανό μειονέκτημα αυτής της προσέγγισης είναι ότι απαιτεί να δημιουργηθεί ένα μοντέλο για κάθε τάξη. Για παράδειγμα, τρεις κλάσεις απαιτούν τρία μοντέλα. Αυτό θα μπορούσε να είναι ένα πρόβλημα για μεγάλο σύνολο δεδομένων .

2. One vs one : χωρίζει μια ταξινόμηση πολλαπλών κλάσεων σε ένα δυαδικό πρόβλημα ταξινόμησης ανά κάθε ζεύγος τάξεων δηλαδή για κάθε πιθανό ζεύγος ταξινόμησης ένας δυαδικός ταξινομητής χρησιμοποιείται. Αυτό έχει ως αποτέλεσμα να χρησιμοποιούνται περισσότερα σύνολα δεδομένων. Ο τύπος για τον υπολογισμό του αριθμού των δυαδικών συνόλων δεδομένων, και κλάσεων είναι :

$$(\text{Αριθμός Κλάσεων} * (\text{Αριθμός Κλάσεων} - 1))/2$$
, για παράδειγμα , για τέσσερις κλάσεις θέλουμε 6 δυαδικούς ταξινομητές.
3. Ιεραρχική ταξινόμηση : η δομή που χρησιμοποιείται είναι ένα δέντρο με ρίζα , κάθε κόμβος του δέντρου αντιπροσωπεύει έναν δυαδικό ταξινομητή , το βάθος του δέντρου εξαρτάται από το σύνολο των χαρακτηριστικών που χωρίζονται σε ομάδες. Τα τελικά φύλλα (κόμβοι) αποτελούν τις κλάσεις ενώ κάθε κόμβος έχει δύο παιδιά.

A. Διαχωρίσιμες κλάσεις

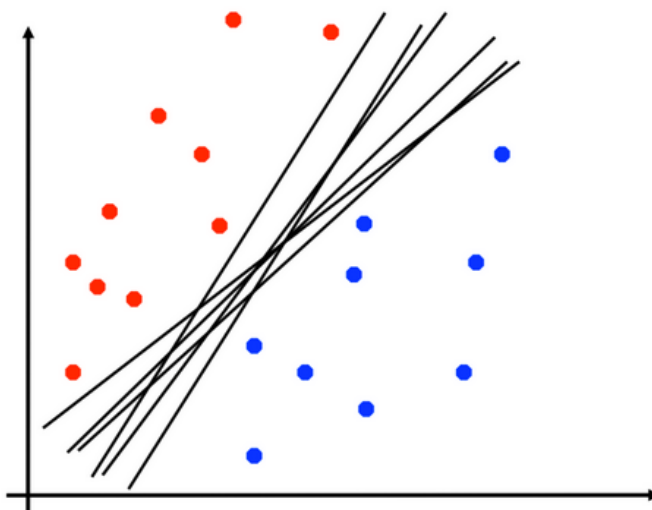
Ας θεωρούμε την περίπτωση δύο γραμμικών διαχωρίσιμων κλάσεων C_1 και C_2 , ο στόχος μας είναι να σχεδιάσουμε ένα υπερεπίπεδο που να ταξινομεί τα δεδομένα σωστά. Υποθέτουμε ότι υπάρχει μια γραμμική συνάρτηση διαχωρισμού : [17]

$$g(x) = w^T x + w_0$$

τέτοια ώστε για κάποιο διάνυσμα w και πόλωση w_0 να έχουμε

$$g(x) = \begin{cases} < 0, & \text{αν } x \in C_0 \\ > 0, & \text{αν } x \in C_1 \end{cases}$$

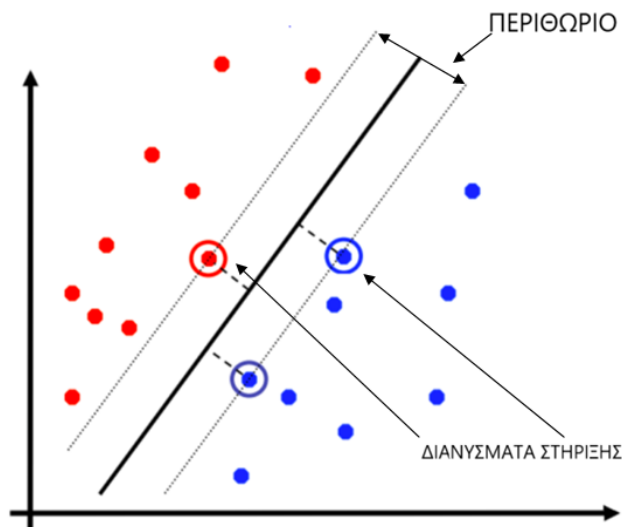
Το σύνολο των σημείων x για τα οποία ισχύει $g(x) = 0$ ορίζουν την διαχωριστική επιφάνεια (είναι τα σημεία τα οποία δεν μπορούμε να αποφασίσουμε σε ποιιά από τις δύο κλάσεις ανήκουν). Ας δούμε το επόμενο σχήμα , εδώ τα πρότυπα μας ανήκουν είτε στην κλάση 'κόκκινα' είτε στην κλάση 'μπλε'. Είναι εμφανές ότι δεν υπάρχει μόνο μια γραμμή που μπορεί να διαχωρίσει τις δύο κλάσεις αλλά άπειρες :



Σχήμα 1.10. Υπάρχουν άπειρες γραμμές που διαχωρίζουν τα δεδομένα [37]

Αυτό έχει ως αποτέλεσμα να πρέπει να βρεθεί κάποιο κριτήριο αξιολόγησης το οποίο ονομάζεται περιθώριο ταξινόμησης (margin) , m , μεταξύ των δύο κλάσεων , το οποίο ορίζεται ως η ελάχιστη απόσταση, d , οποιουδήποτε προτύπου από την διαχωριστική επιφάνεια. Ο

αλγόριθμος SVM πρέπει να βρει την καλύτερη διαχωριστής γραμμής (ή υπερπίεδο σε υψηλές διαστάσεις) με το μέγιστο περιθώριο, και ιδανικά κάθε χαρακτηριστικό θα πρέπει να βρίσκεται εκτός του περιθωρίου. Όσο μεγαλύτερο είναι το περιθώριο, τόσο πιο πιθανό είναι ένα στοιχείο να ταξινομηθεί σωστά.



Σχήμα 1.11. Περιθώριο και διανύσματα στήριξης [37]

Η απόσταση ενός προτύπου από την διαχωριστική επιφάνεια ορίζεται ως :

$$d(x;w, w_0) = \frac{|w^T x + w_0|}{\|w\|},$$

και το περιθώριο ταξινόμησης ως:

$$m(w, w_0) = \min_x d(x;w, w_0)$$

Το διάνυσμα w καθορίζει την κλίση του υπερεπιπέδου, ενώ η πόλωση w_0 καθορίζει τη μετατόπισή του στο χώρο χωρίς να επηρεάζει την κλίση: [1]

- αν η πόλωση είναι μηδέν, το υπερεπίπεδο διέρχεται από την αρχή των αξόνων
- αν η πόλωση είναι θετική, το υπερεπίπεδο μετατοπίζεται προς την κατεύθυνση $+w$
- αν η πόλωση είναι αρνητική, το υπερεπίπεδο μετατοπίζεται προς την κατεύθυνση $-w$

Αν βρεθεί το βέλτιστο διάνυσμα βαρών w , η βέλτιστη πόλωση w_0 είναι αυτή που τοποθετεί το υπερεπίπεδο στο κέντρο μεταξύ των δύο κλάσεων, έτσι ώστε

$$w^T x + w_0 = \begin{cases} \leq -1, & \text{αν } x \in C_0 \\ \geq 1, & \text{αν } x \in C_1 \end{cases} \quad (\text{εξίσωση 1})$$

και επιπλέον να υπάρχει τουλάχιστον ένα πρότυπο x^0 της κλάσης C_0 καθώς και ένα πρότυπο x^1 της κλάσης C_1 , ώστε να ισχύουν οι σχέσεις:

$$w^T x^0 + w_0 = -1$$

και

$$w^T \chi^1 + w_0 = 1$$

Όλα τα πρότυπα των κλάσεων C_0 και C_1 που ικανοποιούν τις προηγούμενες σχέσεις ονομάζονται διανύσματα υποστήριξης και αποτελούν τα ακραία σημεία ώστε να δημιουργηθεί το υπερπλάνο, δίνοντας έτσι το όνομα στον αλγόριθμο.

Τα πρότυπα χ^0 και χ^1 έχουν την ελάχιστη απόσταση από την διαχωριστική επιφάνεια, ενώ η σχέση μεταξύ περιθωρίου και διανύσματος βαρών είναι:

$$m = \frac{|w^T x^1 + w_0|}{\|w\|} = \frac{|w^T x^0 + w_0|}{\|w\|} = \frac{1}{\|w\|}$$

Όσο πιο μεγάλο είναι το $\|w\|$ τόσο πιο μικρό είναι το περιθώριο m , ενώ όσο η τιμή του περιθωρίου πλησιάζει πιο κοντά στο μηδέν τόσο πιο οριακή είναι η σωστή ταξινόμηση των προτύπων, αφού τα πρότυπα που θα εμφανιστούν στο μέλλον έχουν μεγαλύτερη πιθανότητα να βρεθούν στην λάθος πλευρά της διαχωριστικής επιφάνειας. [1]

Γενικεύοντας υποθέτουμε ότι διαθέτουμε N ζεύγη προτύπων x_i και στόχων t_i , και υποθέτοντας ότι οι στόχοι είναι :

$$t_i = \begin{cases} -1, & \text{αν } x_i \in C_0 \\ 1, & \text{αν } x_i \in C_1 \end{cases}$$

τότε οι ανισότητες της εξίσωσης 1 απλοποιούνται ως: $t_i(w^T x_i + w_0) \geq 1, i = 1, \dots, N$.

Η μεγιστοποίηση του περιθωρίου m ισοδυναμεί με την ελαχιστοποίηση της νόρμας $\|w\|$ ή του τετραγώνου αυτής. Η λύση του προηγούμενου προβλήματος ανήκει στην κατηγορία του τετραγωνικού προγραμματισμού και γίνεται με χρήση πολλαπλασιαστών Lagrange. [17][1]

B. Μη διαχωρίσιμες κλάσεις

Όταν οι κλάσεις δεν είναι διαχωρίσιμες τα προηγούμενα δεν είναι πλέον έγκυρα, καθώς κάθε προσπάθεια να σχεδιαστεί ένα υπερεπίπεδο δεν θα δημιουργήσει μια ζώνη διαχωρισμού κλάσεων χωρίς πρότυπα στο εσωτερικό της. Έτσι η προηγούμενη μέθοδος μπορεί να επεκταθεί για να αντιμετωπίσει μη διαχωρίσιμα προβλήματα. Τροποποιώντας το αρχικό πρόβλημα τετραγωνικού προγραμματισμού εισάγοντας μεταβλητές χαλαρότητας (slack variables), ξ , για κάθε έναν από τους προηγούμενους περιορισμούς και έτσι έχουμε:

$$t_i(w^T x_i + w_0) \geq 1 - \xi_i, i = 1, \dots, N, \xi \geq 0$$

Αν η μεταβλητή ξ είναι μεγαλύτερη από ένα το πρότυπο ταξινομείται λάθος. Συνεπώς το άθροισμα των μεταβλητών χαλαρότητας είναι ένα άνω όριο του πλήθους των προτύπων που

ταξινομούνται εσφαλμένα $\sum_{i=1}^N \xi_i >$ πλήθος προτύπων που ταξινομούνται εσφαλμένα.

Έτσι προσθέτουμε ένα κόστος ανάλογο του παραπάνω αθροίσματος στην αρχική μας συνάρτηση κόστους (που θέλαμε να ελαχιστοποιήσουμε $\|w\|^2$). Η συνάρτηση που καλούμαστε τώρα να ελαχιστοποιήσουμε με πρόβλημα τετραγωνικού προγραμματισμού είναι:

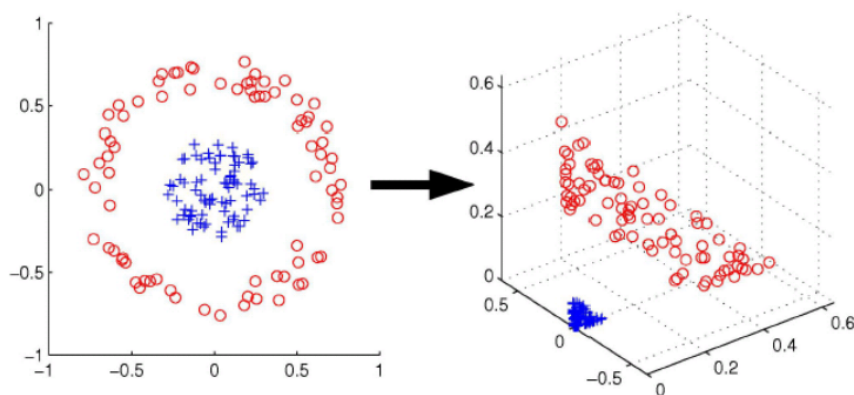
$$J(w, w_0) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i$$

(η σταθερά $\frac{1}{2}$ που συμπεριλαμβάνεται καθαρά για λόγους απλούστευσης της παραγώγισης, και δεν παίζει κάποιο ρόλο)

Η παράμετρος C είναι μια θετική σταθερά που ελέγχει τη σχετική επιρροή των δύο ανταγωνιστικών όρων.[17][1]

- Συναρτήσεις πυρήνα

Ένας σημαντικός περιορισμός των SVM είναι το μεγάλο υπολογιστικό κόστος που απαιτείται τόσο κατά την εκπαίδευση όσο και κατά την δοκιμή. Μια απλή υλοποίηση ενός αλγορίθμου τετραγωνικού προγραμματισμού χρειάζεται $O(N^3)$ πράξεις, ενώ οι απαιτήσεις σε μνήμη είναι της τάξης $O(N^2)$. Για προβλήματα μικρού μεγέθους μπορεί να χρησιμοποιηθεί οποιοσδήποτε αλγόριθμος βελτιστοποίησης γενικού σκοπού ενώ για μεγαλύτερα προβλήματα μια απλοϊκή υλοποίηση δεν κλιμακώνεται ικανοποιητικά. Ενδεικτικά για ένα σύνολο προτύπων $N=100000$ και αν χρησιμοποιήσουμε 8 byte για την αναπαράσταση κάθε αριθμού τότε απαιτούνται 80 GB μνήμης. Αυτό έχει ως αποτέλεσμα να πρέπει να βρεθούν αποδοτικότεροι τρόποι επίλυσης των προβλημάτων. Στα περισσότερα προβλήματα οι κλάσεις είναι μη διαχωρίσιμες από γραμμικές επιφάνειες. Ένας τρόπος αντιμετώπισης μη γραμμικά διαχωρίσιμων κλάσεων είναι με μετασχηματισμό των διανυσμάτων εισόδου σε ένα χώρο μεγαλύτερης διάστασης. Στόχος μας είναι μέσω μιας συνάρτησης $\kappa(x,y)$, η οποία αποτελεί την συνάρτηση πυρήνα (kernel function)-μη γραμμική συνάρτηση, να προβάλουμε τα δεδομένα σε έναν χώρο μεγαλύτερης διάστασης, ώστε να πετύχουμε έναν γραμμικό διαχωρισμό. Ας δούμε το επόμενο σχήμα (αριστερά), είναι εμφανές ότι δεν μπορεί να υπάρξει κάποια γραμμή που να διαχωρίζει τα δεδομένα μας. Αν προσθέσουμε μια τρίτη διάσταση $z = x^2 + y^2$ (συνάρτηση πυρήνα) τα δεδομένα είναι πλέον διαχωρίσιμα (δεξιά).



Σχήμα 1.12. Αριστερα μη διαχωρίσιμα δεδομένα, δεξιά δεδομένα μετά την εφαρμογή συνάρτησης πυρήνα. [38]

Για να γίνει αυτό πηγαίνουμε αρχικά από τον χώρο εισόδου (input space), που περιέχει τα δεδομένα του training set, σε έναν μετασχηματισμένο χώρο χαρακτηριστικών (feature space) υψηλότερης διάστασης μέσω μιας μη γραμμικής απεικόνισης $\Phi(x)$ που ορίζεται ως:

$k(x, y) = \Phi^T(x)\Phi(y) = \langle \Phi(x), \Phi(y) \rangle$, όπου $\langle \rangle$ αναπαριστά εσωτερικό γινόμενο, του καινούριου χώρου χαρακτηριστικών Πηγαίνοντας σε ένα χώρο υψηλότερων διαστάσεων τα δεδομένα μας γίνονται περισσότερο διαχωρίσιμα.

Αν επιπλέον επιλέξουμε κατάλληλα τη συνάρτηση πυρήνα, δεν είναι ανάγκη να υπολογίσουμε αναλυτικά αυτή την απεικόνιση σε υψηλότερες διαστάσεις, αλλά μόνο το εσωτερικό γινόμενο. Στην περίπτωση αυτή ισχύουν τα ίδια αποτελέσματα όπως περιγράψαμε παραπάνω με την διαφορά ότι κάθε πρότυπο x_i αντικαθίσταται από το $\Phi(x_i)$, οπότε η βέλτιστη διαχωριστική επιφάνεια θα είναι [1][17]

$$g^*(x) = w^T \Phi(x) + w_0.$$

Υπάρχουν πολλοί τύποι συναρτήσεων πυρήνα μερικοί από αυτούς είναι: [1][17]

- Πολυωνυμικές: $k(x, y) = (x^T y)^M = \langle x^T y \rangle^M$
- Ανομοιογενείς πολυωνυμικές: $k(x, y) = (x^T y + c)^M$
- Gauss/ ακτινικής βάσης: $k(x, y) = \exp(-\|x - y\|^2 / 2)$
- Σιγμοειδής: $k(x, y) = \tanh(ax^T y + b)$ (συνάρτηση υπερβολικής εφαπτομένης. Η επιλογή της ως πυρήνας παραπέμπει σε μια ειδική περίπτωση perceptron δύο επιπέδων.)
- Αντίστροφη πολυτετραγωνική: $k(x, y) = \frac{1}{\sqrt{\|x - y\|^2 + c^2}}$
- RBF (Radial Basis Function) Μια λειτουργία ακτινικής βάσης (RBF) είναι μια συνάρτηση του οποίου η τιμή εξαρτάται μόνο από την απόσταση μεταξύ της εισόδου και κάποιου σταθερού σημείου, απόσταση είναι συνήθως Ευκλείδεια απόσταση ,
 $k(x, y) = \frac{\|x - y\|^2}{2\sigma^2}$ [60][61]

Για να υπολογίσουμε το μετασχηματισμό δεν είναι απαραίτητη η γνώση του $\Phi(x)$, παρά μόνο η γνώση της συνάρτησης πυρήνα. Η συνάρτηση πυρήνα με άλλα λόγια υπολογίζει το εσωτερικό γινόμενο στο χώρο των χαρακτηριστικών $\Phi(x)$ απευθείας, χωρίς γνώση ή/και υπολογισμό της απεικόνισης Φ .

Μειονεκτήματα χρήσης των συναρτήσεων πυρήνα:

- Καθιστούν το μοντέλο ευάλωτο σε υπερμοντελοποίηση
- Προβάλλοντας τα δεδομένα σε πολλές διαστάσεις μπορεί να καταλήξουμε σε υπερβολική εξειδίκευση
- Σε οποιοδήποτε πρόβλημα είναι δύσκολο να βρεθεί ποιά είναι η καταλληλότερη συνάρτηση πυρήνα.

1.2.2.6.1 Παλινδρόμηση με διανύσματα υποστήριξης

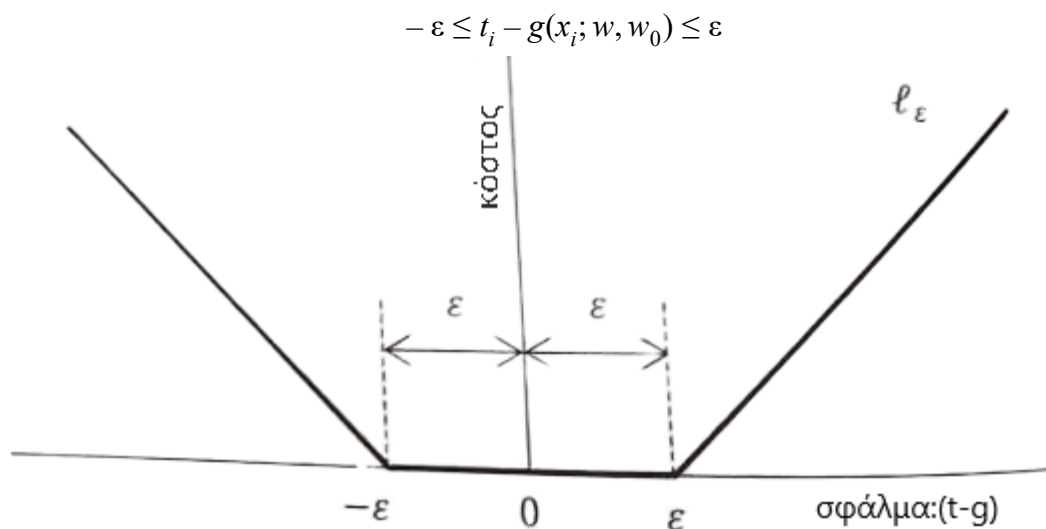
Η παλινδρόμηση μοιάζει με την ταξινόμηση υπό την έννοια ότι για κάθε πρότυπο υπάρχει μια τιμή-στόχος, διαφέρει ωστόσο στο γεγονός ότι η τιμή-στόχος ανήκει σε ένα συνεχές σύνολο τιμών. Με δεδομένο ένα σύνολο δειγμάτων εκπαίδευση στόχος είναι να υπολογιστεί μια συνάρτηση η οποία εκτιμά με βέλτιστο τρόπο την τιμή της εξόδου. Σε έναν αριθμό περιπτώσεων τα κόστη τύπου μέσου τετραγώνου ή ελαχίστων τετραγώνων δεν είναι κατάλληλα, όπως για παράδειγμα σε περιπτώσεις όπου η στατιστική κατανομή των δεδομένων παρουσιάζει μακριές ουρές ή όταν τα δεδομένα έχουν επισημειωθεί εσφαλμένα. Θεωρούμε αρχικά μια γραμμική περίπτωση και ας υποθέσουμε ένα σύνολο προτύπων x_i και στόχων $t_i, i = 1..N$, όπου οι τιμές των στόχων ανήκουν στην γενική περίπτωση στο σύνολο των πραγματικών αριθμών. Αναζητούμε λοιπόν μια συνάρτηση της μορφής: [1]

$$g(x) = w^T x + w_0$$

έτσι ώστε οι τιμές $g(x_i)$ να βρίσκονται όσο πιο κοντά γίνεται στους στόχους t_i . Η σύνδεση του προβλήματος με τις μηχανές διανυσματικής υποστήριξης γίνεται με χρήση της συνάρτησης σφάλματος που είναι γνωστή ως συνάρτηση σφάλματος με ανοχή ϵ :

$$l_\epsilon(t, g) = \begin{cases} 0, & \text{αν } |t-g| \leq \epsilon \\ |t-g| - \epsilon, & \text{αν } |t-g| > \epsilon \end{cases}$$

όπου ϵ είναι μια μικρή θετική σταθερά που επιλέγεται από τον χρήστη. Η συνάρτηση αυτή τιμωρεί τη διαφορά μεταξύ του στόχου t και της εκτιμώμενης τιμής g μόνο εφόσον η απόλυτη τιμή αυτής της διαφοράς είναι μεγαλύτερη από ϵ . Έχουμε ανοχή στο σφάλμα, δηλαδή δεν πληρώνουμε κάποιο κόστος, αν το σφάλμα είναι μικρότερο από της ϵ κατ' απόλυτη τιμή:



Σχήμα 1.13. Η συνάρτηση κόστους με ανοχή ϵ έχει εξ ορισμού τιμή μηδέν αν η απόλυτη τιμή της διαφοράς της τιμής στόχου t από την εκτιμώμενη τιμή g είναι μικρότερη από ϵ . [1]

Το πρόβλημα βελτιστοποίησης εδώ μοιάζει πολύ με το πρόβλημα βελτιστοποίησης των SVM στην περίπτωση της ταξινόμησης, με διαφορά στις ανισότητες περιορισμού. Θέλουμε να βρούμε το ελάχιστο της συνάρτησης $L(w, w_0) = 0.5 \|w\|^2$ υπό τους περιορισμούς $t_i - w^T x_i - w_0 \leq \varepsilon$ και $t_i - w^T x_i - w_0 \geq -\varepsilon$ που ανάγεται πάλι σε ένα πρόβλημα τετραγωνικού προγραμματισμού.

Όπως και στην περίπτωση της ταξινόμησης, έτσι και στην περίπτωση της παλινδρόμησης μπορούμε να επεκτείνουμε τη μέθοδο όταν η λύση δεν είναι εφικτή, όταν δεν υπάρχει δηλαδή ζεύγος (w, w_0) ώστε το συνολικό κόστος να είναι μηδέν. Η επέκταση γίνεται με την εισαγωγή μεταβλητών χαλαρότητας, όμως θα εισαχθούν δύο μεταβλητές χαλαρότητας για κάθε πρότυπο:

- η μεταβλητή $\xi_i \geq 0$ η οποία αντιστοιχεί στην απόκλιση της τιμής στόχου t_i πάνω από το όριο $w^T x_i + w_0 + \varepsilon$
- η μεταβλητή $\hat{\xi}_i \geq 0$ η οποία αντιστοιχεί στην απόκλιση της τιμής στόχου t_i κάτω από το όριο $w^T x_i + w_0 - \varepsilon$

Έτσι προκύπτει:

$$\xi_i \geq t_i - w^T x_i - w_0 - \varepsilon \text{ και } \hat{\xi}_i \geq w^T x_i + w_0 - \varepsilon - t_i .$$

Η μέθοδος αυτή της παλινδρόμησης SVM μπορεί να επεκταθεί και σε μη γραμμικά προβλήματα κάνοντας επίσης χρήση συναρτήσεων πυρήνα.[1]

➤ Τα πλεονεκτήματα των μηχανών διανυσμάτων υποστήριξης είναι : [41][42][1][17]

- Αποτελεσματικά σε χώρους μεγάλης διάστασης.
- Αποτελεσματικά στις περιπτώσεις όπου ο αριθμός των διαστάσεων είναι μεγαλύτερος από τον αριθμό των δειγμάτων.
- Είναι πιο αποτελεσματικός σε σχέση με άλλους αλγορίθμους
- Χρησιμοποιούν ένα υποσύνολο σημείων εκπαίδευσης στη συνάρτηση λήψης αποφάσεων (που ονομάζεται φορέα υποστήριξης), συνεπώς είναι αποδοτικά σε θέματα μνήμης.
- Το SVM λειτουργεί καλά με ένα καθαρό περιθώριο διαχωρισμού ακόμα και σε χώρο υψηλών διαστάσεων.
- Ευέλικτο: μπορούν να καθοριστούν διαφορετικές συναρτήσεις πυρήνα για τη συνάρτηση απόφασης. Γενικώς παρέχονται κοινοί πυρήνες (kernels) αλλά ο χρήστης μπορεί να θέσει τους δικούς του πυρήνες.

➤ Τα μειονεκτήματα των μηχανών διανυσμάτων υποστήριξης είναι : [41][42][1][17]

- Εάν ο αριθμός των χαρακτηριστικών είναι πολύ μεγαλύτερος από τον αριθμό των δειγμάτων, υπάρχει κίνδυνος για υπερβολικό overfitting και αυτό πρέπει να αποφευχθεί μέσω της επιλογής σωστής συνάρτησης πυρήνα.
- Τα SVM δεν παρέχουν απευθείας εκτιμήσεις πιθανότητας.
- Για μεγάλο σύνολο δεδομένων μπορούν να γίνουν ακατάλληλα εξαιτίας του υψηλού χρόνου εκπαίδευσης
- Είναι ευαίσθητος και εξαρτώμενα από τον πυρήνα που χρησιμοποιείται

1.2.2.7 Κριτήρια επίδοσης στην ταξινόμηση

Στην ταξινόμηση (δυναμική) συνήθως η μια κλάση περιέχει πρότυπα για τα οποία ισχύει μια κατάσταση (κλάση 1- true), ενώ η άλλη κλάση περιέχει πρότυπα για τα οποία η κατάσταση αυτή δεν ισχύει (κλάση 0-false).

Αν το πρότυπο x_i βγει Θετικό ($y_i = 1$), τότε λέμε ότι ταξινομείται στην Κλάση 1
αν το πρότυπο βγει αρνητικό ($y_i = 0$), τότε λέμε ότι ταξινομείται στην Κλάση 0

Ιδανικά θα έπρεπε, για κάθε πρότυπο x_i , το test να δίνει έξοδο y_i που να αντιστοιχεί με την ετικέτα t_i της κλάσης στην οποία ανήκει το πρότυπο, όμως αυτά δεν ισχύει πάντα. Υπάρχουν 4 περιπτώσεις: [1]

- Το πρότυπο βγήκε αρνητικό, δηλαδή $y_i = 0$, και όντως ανήκει στην κλάση 0, ισχύει δηλαδή $t_i = 0$. Τέτοια πρότυπα καλούνται true negatives (πραγματικά αρνητικά).
- Το πρότυπο βγήκε αρνητικό, δηλαδή $y_i = 0$, αλλά ανήκει στην Κλάση 1, δηλαδή $t_i = 1$. Τέτοια πρότυπα καλούνται false negatives (εσφαλμένα αρνητικά).
- Το πρότυπο βγήκε θετικό, δηλαδή $y_i = 1$, αλλά ανήκει στην Κλάση 0, δηλαδή $t_i = 0$. Τέτοια πρότυπα καλούνται false positives (εσφαλμένα Θετικά).
- Το πρότυπο βγήκε θετικό, δηλαδή $y_i = 1$, και όντως ανήκει στην Κλάση 1, δηλαδή $t_i = 1$. Τέτοια πρότυπα καλούνται true positives (πραγματικά θετικά).

Υποθέτοντας ένα σύνολο προτύπων $X = \{x_1, \dots, x_n\}$, αξιολογούμε την επίδοση ενός ταξινομητή χρησιμοποιώντας κάποια χαρακτηριστικά του λεγόμενου πίνακα σύγχυσης (confusion matrix), στην ιδανική περίπτωση ο πίνακας θα πρέπει να έχει μηδενικά διαγώνια στοιχεία, δηλαδή να μην υπάρχουν εσφαλμένα θετικά πρότυπα (FP) ούτε εσφαλμένα αρνητικά πρότυπα (FN). Ένα πρώτο μέτρο εκτίμησης της επίδοσης είναι ο λόγος των ορθώς ταξινομημένων προτύπων προς το σύνολο των προτύπων. Ο λόγος αυτός είναι γνωστός ως ακρίβεια (Accuracy):

$$Accuracy = \frac{TN+TP}{TN+TP+FN+FP}$$

Το κριτήριο αυτό έχει ως μειονέκτημα την γενικότητα του και δεν προσφέρει πληροφορίες σχετικά με την επίδοση του ταξινομητή όσον αφορά την ταξινόμηση προτύπων της κλάσης 1. Αυτό έχει ως αποτέλεσμα να έχουν προταθεί δύο άλλα δημοφιλή κριτήρια,

- Ευστοχία (Precision):[175]

$$Precision = \frac{TP}{TP+FP}$$

μας δείχνει το ποσοστό των προτύπων που κατηγοριοποιούνται ως θετικά και ανήκουν πράγματι στην Κλάση 1, (ικανότητα του ταξινομητή να μην επισημαίνει ως θετικό, δείγμα που είναι αρνητικό) Για παράδειγμα, αν ισχύει Precision = 0.9, αυτό σημαίνει ότι 90 στα 100 πρότυπα είναι θετικά

- Ανάκληση (Recall) [174]

$$Recall = \frac{TP}{TP+FN}$$

δείχνει το ποσοστό των προτύπων που ανήκουν στην Κλάση 1 και κατηγοριοποιούνται ως θετικά (η ικανότητα του ταξινομητή να βρει όλα τα θετικά δείγματα). Για παράδειγμα, αν ισχύει $Recall = 0.7$, αυτό σημαίνει ότις 70 στα 100 πρότυπα είναι θετικά.

Οι τιμές και των δύο κριτηρίων κυμαίνονται μεταξύ 0 και 1. Ιδανικά θα θέλαμε να μην υπάρχουν εσφαλμένα αρνητικά πρότυπα (FN) και εσφαλμένα θετικά πρότυπα (FP), οπότε θα ισχύει $Precision = Recall = 1$, αυτό δεν είναι πάντα εφικτό. Επίσης, είναι δυνατόν κάποιος ταξινομητής να έχει καλή επίδοση στο κριτήριο Precision αλλά όχι στο Recall, και το αντίστροφο. [1]. Αυτό έχει ως αποτέλεσμα τα κριτήρια αυτά να χρησιμοποιούνται σε συνδυασμό με το κριτήριο f1-score (F-measure) το οποίο ορίζεται ως το ηλίκο του γεωμετρικού μέσου προς τον αλγεβρικό μέσο όρο των δύο κριτηρίων :

$$F1 - score = \frac{Precision * Recall}{(Precision+Recall)/2} ,$$

το οποίο παίρνει τιμές μεταξύ 0 και 1.

Το πλήθος των πραγματικά αρνητικών προτύπων (TN) δεν εμπεριέχεται στον ορισμό των προηγούμενων κριτηρίων. Για το λόγο αυτό έχουν οριστεί άλλα δύο μέτρα επίδοσης που δίνουν ίση βαρύτητα και στις δυο κλάσεις :

- Ευαισθησία (Sensitivity)

$$Sensitivity = \frac{TP}{TP+FN} ,$$

Είναι ίδιο με το Recall

- Ειδικότητα (Specificity)

$$Specificity = \frac{Tn}{TN+FP} ,$$

το κριτήριο αυτό δείχνει σε τι ποσοστό το test ταξινομεί σωστά τα πρότυπα της κλάσης 0 .

Τα κριτήρια αυτά παίρνουν τιμές μεταξύ 0 και 1 ενώ ένα συνδυαστικό κριτήριο αυτών αποτελεί η καμπύλη ROC [1]

Σε περίπτωση που το πλήθος των κλάσεων είναι περισσότερες από δύο μπορούμε να ελεγκτήσουμε τα προηγούμενα κριτήρια .Ο πίνακας σύγχυσης έχει μέγεθος $C \times C$ όπου C ο αριθμός των κλάσεων , κάθε στοιχείο n_{ij} του πίνακα καταγράφει το πλήθος των προτύπων που ανήκουν στην κλάση $t = i$ αλλά έχουν ταξινομηθεί στην κλάση $y = j$. Τα πιο γνωστά κριτήρια αξιολόγησης ταξινομητών είναι : [1]

Μέση Ακρίβεια (average accuracy)	$Acc_m = \frac{1}{C} \sum_{i=1}^C \frac{tp_i+tn_i}{tp_i+tn_i+fp_i+fn_i}$
Μέσο Σφάλμα (Average error rate)	$Err_m = \frac{1}{C} \sum_{i=1}^C \frac{fp_i+fn_i}{tp_i+tn_i+fp_i+fn_i}$
Μέση Ευστοχία (average precision)	$P_m = \frac{1}{C} \sum_{i=1}^C \frac{tp_i}{tp_i+fp_i}$
Μέση ανάκληση (average recall)	$R_m = \frac{1}{C} \sum_{i=1}^C \frac{tp_i}{tp_i+fn_i}$
Μέσο F-measure (f1) (average F-measure)	$\frac{2(P_m * R_m)}{P_m+R_m}$

1.3 Εισαγωγή στη μη επιτηρούμενη μάθηση

Ένα σύστημα με μάθηση χωρίς επίβλεψη οφείλει από μόνο του να ανακαλύψει συσχετίσεις ή ομάδες σε ένα σύνολο δεδομένων, δημιουργώντας πρότυπα, χωρίς αρχικά να είναι γνωστό αν υπάρχουν, πόσα και ποιά είναι. Σαν αποτέλεσμα προκύπτουν πρότυπα (περιγραφές), κάθε ένα από τα οποία περιγράφει ένα μέρος από τα δεδομένα. Η μη επιτηρούμενη μάθηση είναι χρήσιμη σε προβλήματα συσταδοποίησης (clustering) όπου ζητείται να εντοπιστούν συστάδες δεδομένων, δηλαδή περιοχές με μεγάλη πυκνότητα πιθανότητας, αλλά και σε προβλήματα εκτίμησης κατανομής. Το μόνο που είναι διαθέσιμο σε αυτά τα προβλήματα είναι τα δείγματα δεδομένων, $\{x_i\}_{i=1}^N$.

Στην περίπτωση του προβλήματος της εκτίμησης μιας κατανομής το ζητούμενο είναι να εκτιμηθεί η πιθανότητα $P(x)$ αν το x παίρνει διακριτές τιμές ή η πυκνότητα πιθανότητας $p(x)$ αν το x παίρνει συνεχείς τιμές.

Στην περίπτωση της συσταδοποίησης το ζητούμενο είναι η εύρεση k σημείων $\{c_k\}_{k=1}^k$, τα οποία θα αποκαλούνται κέντρα των συστάδων, έτσι ώστε η κατανομή πιθανότητας να εκτιμάται

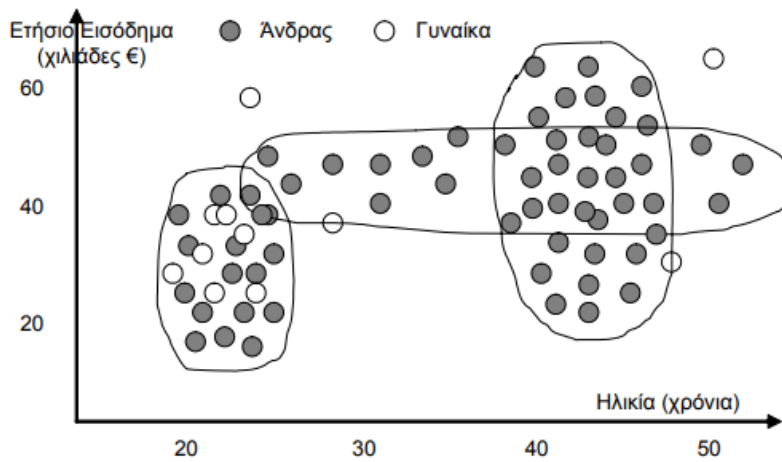
ικανοποιητικά από το άθροισμα $\hat{P}(x) = \sum_{k=1}^k w(x - c_k; \theta_k)$ για κάποια παραμετρική συνάρτηση

$w(\cdot; \theta)$ με παραμέτρους θ . Η συνάρτηση w ονομάζεται συνάρτηση παραθύρου και είναι θετική με μέγιστο τιμή $w(0)$. Αφού υπολογιστούν τα κέντρα c και οι παράμετροι θ για όλες τις συστάδες, η k -οστή συστάδα είναι η περιοχή του χώρου όπου το παράθυρο k υπερτερεί όλων των υπολοίπων ισχύει δηλαδή $w(x - c_k; \theta_k) > w(x - c_j; \theta_j)$ για όλα τα $j \neq k$. [1]

1.3.2 Ομαδοποίηση

Η ομαδοποίηση είναι η διαδικασία κατά την οποία ένα σύνολο παρατηρήσεων χωρίζεται σε υποσύνολα έχοντας ως αποτέλεσμα στην ίδια ομάδα να είναι οι όμοιες παρατηρήσεις, σύμφωνα με ορισμένα κριτήρια που έχουν οριστεί αρχικά, ενώ οι παρατηρήσεις που προέρχονται από διαφορετικά υποσύνολα να είναι ανόμοιες. Τα πρότυπα πληροφόρησης που προκύπτουν με ομαδοποίηση (clustering) δηλαδή διαχωρισμό ενός συνόλου (συνήθως πολυδιάστατων) δεδομένων σε ομάδες έτσι ώστε, σημεία που ανήκουν στην ίδια ομάδα να μοιάζουν όσο το δυνατόν περισσότερο και σημεία που ανήκουν σε διαφορετικές ομάδες να διαφέρουν όσο το δυνατόν περισσότερο.

Στο επόμενο σχήμα απεικονίζεται γραφικά μία υποθετική ομαδοποίηση σε δεδομένα αγοραστών σπορ αυτοκινήτων, με βάση την ηλικία (άξονας x), το ετήσιο εισόδημα (άξονας y) και το φύλο.



Σχήμα 1.14 ομαδοποίηση δεδομένων [43]

Διακρίνονται τρεις ομάδες:

1. αγοραστές νεαρής ηλικίας ανεξαρτήτως φύλλου,
2. άνδρες αγοραστές με υψηλό εισόδημα, όλων των ηλικιών μέχρι τα 53 χρόνια
3. άνδρες αγοραστές ηλικίας περίπου 44 ετών ανεξαρτήτως εισοδήματος.

Διαφορετικές τεχνικές κατηγοριοποίησης οδηγούν σε διαφορετικές υποθέσεις σχετικά με τη δομή των δεδομένων, οι οποίες συχνά καθορίζονται από κάποιο μέτρο ομοιότητας και αξιολογούνται για παράδειγμα ως προς την εσωτερική συνοχή (ομοιότητα μεταξύ των μελών του ίδιου cluster) και το διαχωρισμό ανάμεσα σε διαφορετικές ομάδες.

Υπάρχουν τρεις γενικές κατηγορίες αλγορίθμων ομαδοποίησης :[43]

- Οι αλγόριθμοι βασισμένοι σε διαχωρισμούς (partition based), που προσπαθούν να βρουν τον καλύτερο διαχωρισμό ενός συνόλου δεδομένων σε ένα συγκεκριμένο αριθμό ομάδων
- Οι ιεραρχικοί (hierarchical) αλγόριθμοι, που προσπαθούν με ιεραρχικό τρόπο να ανακαλύψουν τον αριθμό και τη δομή των ομάδων.
- Οι πιθανοκρατικοί (probabilistic) αλγόριθμοι, που βασίζονται σε μοντέλα πιθανοτήτων.

Η ομαδοποίηση απαιτεί κάποιο μέτρο ομοιότητας ή διαφοράς μεταξύ των προς επεξεργασία δεδομένων, όπου τις περισσότερες φορές υπολογίζεται η απόσταση μεταξύ των δεδομένων. Έστω ένα σύνολο δεδομένων D , και δύο δεδομένα του, x, y που περιγράφονται από m χαρακτηριστικά $(x_1, x_2, \dots, x_m), (y_1, y_2, \dots, y_m)$.

Τυπικά μέτρα απόστασης αυτών των δύο δεδομένων είναι η απόσταση Μανχάταν και η Ευκλείδεια απόσταση:

$$d(x, y) = \sum_i |x_i - y_i|$$

Απόσταση Μανχάταν

$$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

Ευκλείδεια απόσταση

Αν κάποια χαρακτηριστικά είναι διακριτά, τότε η απόσταση των τιμών τους θεωρείται 0 αν πρόκειται για την ίδια τιμή και 1 αν πρόκειται για διαφορετικές τιμές. □ Τα αριθμητικά χαρακτηριστικά θα πρέπει να ομογενοποιούνται ώστε η απόστασή τους να πέφτει μέσα στο διάστημα [0,1]

1.3.3 Ο αλγόριθμος των K-μέσων

Ο αλγόριθμος αυτός αποτελεί τον πιο κλασικό αλγόριθμο συσταδοποίησης . Ο αλγόριθμος δεν υποθέτει κάποια συγκεκριμένη κατανομή αλλά θεωρεί ότι ο αριθμός K των ομάδων-συστάδων έχει καθοριστεί πριν την εκτέλεση του αλγορίθμου. □ Ο αλγόριθμος ξεκινά διαλέγοντας K τυχαία σημεία από τα δεδομένα ως τα κέντρα των ομάδων. □ Έπειτα αναθέτει κάθε σημείο στην ομάδα της οποίας το κέντρο είναι πιο κοντά (μικρότερη απόσταση) σε αυτό το σημείο. □ Στη συνέχεια, υπολογίζει για κάθε ομάδα το μέσο όρο όλων των σημείων της (μέσο διάνυσμα) και ορίζει αυτό ως νέο κέντρο της. □ Τα δύο τελευταία βήματα επαναλαμβάνονται για ένα προκαθορισμένο αριθμό βημάτων ή μέχρι να μην υπάρχει αλλαγή στο διαχωρισμό των σημείων σε ομάδες

Πιο συγκεκριμένα , ο αλγόριθμος είναι αναδρομικός , αφού αρχικοποιηθούν τα κέντρα κάθε επανάληψη που ακολουθεί αποτελείται από δύο βήματα : [1]

1. Για κάθε πρότυπο βρίσκουμε ποιο κέντρο βρίσκεται πιο κοντά σε αυτό και το τοποθετούμε στην συστάδα του συγκεκριμένου κέντρου , δίνουμε δηλαδή στο πρότυπο μια ετικέτα από το 1 ως το k η οποία δείχνει σε ποια συστάδα ανήκει :

$$label[i] = arg \min_i ||x_i - c_k|| , x_i \rightarrow \text{πρότυπο} , c_k \rightarrow \text{κέντρο συστάδας}$$

2. Ενημερώνουμε την τιμή κάθε κέντρου θέτοντας το ίσο με την μέση τιμή των προτύπων που ανήκουν στη συστάδα του :

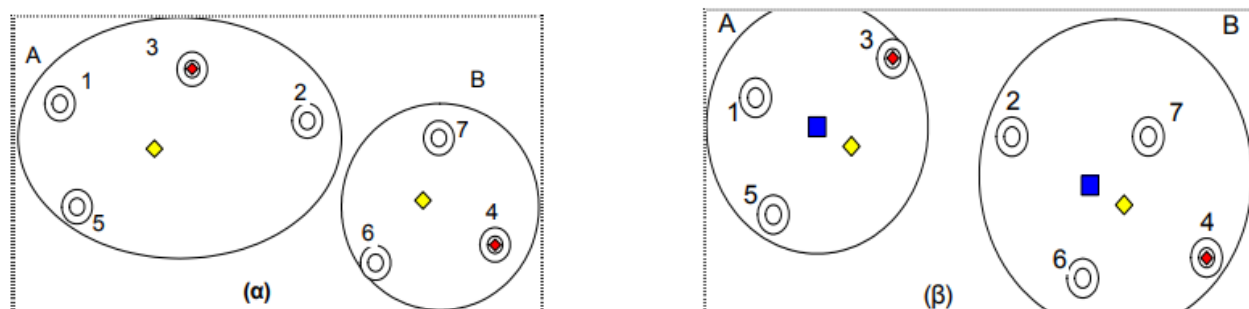
$$c_k = \frac{1}{N_k} \sum_{x_i \in G_k} x_i , N_k \rightarrow \text{πληθος προτύπων που ανήκουν στην συστάδα } G_k$$

Ο αλγόριθμος τερματίζει όταν σε δύο διαδοχικές επαναλήψεις δεν υπάρχει καμία αλλαγή στις ετικέτες των προτύπων , έχοντας ως αποτέλεσμα να μην υπάρχει και αλλαγή ούτε στα κέντρα c_k .

Παράδειγμα αλγορίθμου : [43]

Έστω ότι ο αλγόριθμος εκτελείται με $k=2$ για τα 7 σημεία του σχήματος. □

- Αρχικά επιλέγονται τυχαία, έστω τα σημεία 3 και 4, ως κέντρα για τις δύο ομάδες A και B □
- Κάθε σημείο από τα υπόλοιπα ανατίθεται στην ομάδα στις οποίες το κέντρο είναι πιο κοντά, άρα τα σημεία 1, 2, 3 και 5 θα ανήκουν στην ομάδα A ενώ τα 4, 6 και 7 στην B.
- □ Ξαναυπολογίζονται τα κέντρα κάθε ομάδας και με αυτόν τον τρόπο ολοκληρώνεται ένας κύκλος υπολογισμών. □ Τα νέα κέντρα απεικονίζονται στο σχήμα (α) με ρόμβο



Σχήμα 1.15. Παράδειγμα συσταδοποίησης [43]

Ο παραπάνω κύκλος υπολογισμών επαναλαμβάνεται χρησιμοποιώντας αυτήν τη φορά τις αποστάσεις των σημείων από τα νέα κέντρα.

- Μια μεταβολή που συντελείται είναι ότι το σημείο 2 αλλάζει ομάδα και ανήκει πλέον στη B.
- Επιπλέον, τα κέντρα των ομάδων μετατοπίζονται σε νέες θέσεις που στο σχήμα (β) σημειώνονται με τετράγωνο.
- Κατά την τρίτη επανάληψη των παραπάνω υπολογισμών, δε συντελείται καμία μεταβολή στον πληθυσμό κάθε ομάδας, οπότε η διαδικασία τερματίζει.

Βασικό μειονεκτήματα του αλγορίθμου k -means είναι το γεγονός ότι δεν υπάρχει κάποιος αυτοματοποιημένος τρόπος επιλογής του k , δηλαδή του αριθμού των συστάδων. Ο αριθμός των συστάδων μπορεί να δοθεί ως είσοδος από τον χρήστη αυτό έχει ως αποτέλεσμα η διαδικασία επιλογής του αριθμού συστάδων, να απαιτήσει την εξερεύνηση και μελέτη των δεδομένων, για παράδειγμα, μέσα από οπτικοποιήσεις, προκειμένου να καταλήξουμε στο σωστό αριθμό συστάδων .[44] Άλλοι εμπειρικοί κανόνες που βασίζονται σε δεδομένα μπορούν να βοηθήσουν για την καλύτερη επιλογή του k απο το χρήστη με τις ποιες γνωστές να είναι ο 'κανόνας του αγκώνα' καθώς και η μέθοδος Silhouette .[45]

Η επιλογή του κέντρου μπορεί να γίνει είτε με τυχαίο τρόπο αναθέτοντες τυχαίες ετικέτες στα δεδομένα και κάθε κέντρο c_k αρχικοποιείται με τον μέσο όρο των προτύπων που ανήκουν στην αντίστοιχη συστάδα είτε με τυχαία επιλογή προτύπων όπου το κάθε κέντρο αρχικοποιείται με τυχαίο πρότυπο.

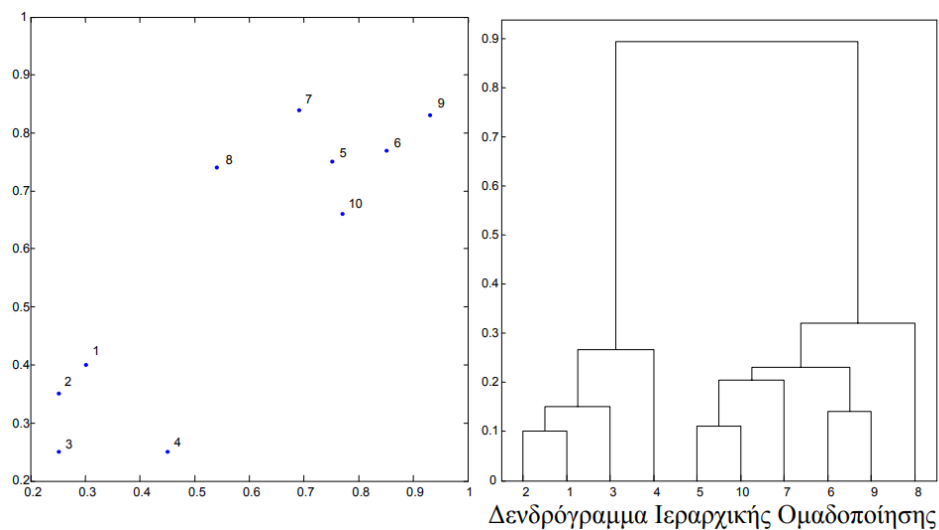
1.3.4 Αλγόριθμοι Ιεραρχικής Ομαδοποίησης

Οι αλγόριθμοι ιεραρχικής ομαδοποίησης συνδυάζουν ομάδες σε μεγαλύτερες ομάδες ή διαιρούν μεγάλες ομάδες σε μικρότερες. Η μέθοδος βασίζεται στην έννοια της απόστασης. Ο όρος ιεραρχική προήλθε καθώς οι συστάδες εμπεριέχουν συστάδες αλλά και μεμονωμένα στοιχεία, ενώ οι συστάδες μπορεί να περιέχουν και αυτές άλλες μικρότερες συστάδες, δημιουργώντας έτσι τα επίπεδα της ιεραρχίας. Οι αλγόριθμοι μπορούν να αναπαρασταθούν πλήρως με δενδρογράμματα, τα οποία παρουσιάζουν τη διάταξη των συστάδων που δημιουργήθηκαν από την ιεραρχική συσταδοποίηση, κάθε επίπεδο ορίζει ένα βήμα του αλγορίθμου. Ένα από τα βασικότερα πλεονεκτήματα της τεχνικής αυτής είναι ότι δεν χρειάζεται να υποθέσουμε ένα συγκεκριμένο αριθμό συστάδων, αφού οποιοσδήποτε αριθμός μπορεί να επιτευχθεί, απλά κόβοντας το δενδρόγραμμα στο κατάλληλο επίπεδο. [44][1]

Το αποτέλεσμα των αλγορίθμων αυτών είναι μια ιεραρχία από διαφορετικές ομαδοποιήσεις των δεδομένων στο ένα άκρο της οποίας βρίσκεται μια μόνο ομάδα με όλα τα δεδομένα, και στο άλλο τόσες ομάδες όσες και ο αριθμός των δεδομένων

Με βάση την κατεύθυνση ανάπτυξης της ιεραρχίας που ακολουθούν, οι ιεραρχικοί αλγόριθμοι ομαδοποίησης χωρίζονται στους αλγορίθμους συγχώνευση-συσσώρευσης (agglomerative) και στους αλγορίθμους διαίρεσης (divisive).

- Στη συσσωρευτική συσταδοποίηση κάθε σημείο αποτελεί μια ξεχωριστή συστάδα, ενώνουμε αναδρομικά τις δύο συστάδες που έχουν την μικρότερη απόσταση μεταξύ τους και τερματίζεται όταν οι πιο κοντινές συστάδες έχουν απόσταση μεγαλύτερη από ένα κατώφλι. Η απόσταση μεταξύ συστάδων ονομάζεται σύνδεση και οι πιο βασικοί μέθοδοι είναι:
 - Απλής σύνδεσης : Η απόσταση μεταξύ των συστάδων ισούται με την απόσταση των κοντινότερων σημείων των δυο συστάδων
 - Πλήρους σύνδεσης : Η απόσταση μεταξύ των συστάδων ισούται με την απόσταση των μακρινότερων σημείων των δυο συστάδων
 - Μέσης σύνδεσης : Η απόσταση μεταξύ των συστάδων ισούται με την μέση απόσταση όλων των ζευγών σημείων των δυο συστάδων
 - Απόσταση κεντρικών σημείων : Η απόσταση κεντρικών σημείων είναι η απόσταση μεταξύ των κέντρων των συστάδων.
- Στη διαιρετική συσταδοποίηση αρχίζουμε με όλα τα σημεία σε μια συστάδα και αναδρομικά δημιουργούνται υποσυστάδες έτσι ώστε να επιτύχουμε την μεγαλύτερη απόσταση μεταξύ των τμημάτων.
- Οι ιεραρχίες που προκύπτουν από τους αλγορίθμους ιεραρχικής ομαδοποίησης μπορεί να απεικονιστούν με έναν πρακτικό και εύκολο τρόπο μέσω ενός γραφήματος δενδρικής μορφής, το οποίο ονομάζεται δενδρόγραμμα. Η πολυπλοκότητα του αλγορίθμου είναι μεγάλη ενώ η επιλογή βέλτιστης συστάδας είναι σχεδόν αδύνατη ακόμα και για μικρό αριθμό δειγμάτων, έχοντας ως αποτέλεσμα ο αλγόριθμος να μην έχει πρακτική εφαρμογή.



Σχήμα : 1.16. Αριστερά φαίνονται κάποια σημεία στο διδιάστατο χώρο, δεξιά είναι το δενδρόγραμμα που προκύπτει μέσω ιεραρχικής ομαδοποίησης.[43]

1.3.5 Μέθοδος DBSCAN (Density-based spatial clustering of applications with noise)

Η μέθοδος αυτή αποτελεί ίσως την πιο γνωστή μέθοδο συσταδοποίησης στη εξόρυξη δεδομένων , ενώ όπως και στην ιεραρχική συσταδοποίηση δεν χρειάζεται η εκ των προτέρων γνώση του αριθμού των συστάδων. Ο αλγόριθμος καθορίζει αυτόματα το πλήθος , το σχήμα και τη θέση των συστάδων με βάση τη συγκέντρωση των προτύπων στο χώρο.[1]

Η μέθοδος DBSCAN χρησιμοποιεί δύο παραμέτρους, το ϵ και το MinPts, δηλαδή τον ελάχιστο αριθμό σημείων που απαιτούνται για τη δημιουργία μιας πυκνής περιοχής. Ο αλγόριθμος ξεκινάει με την εύρεση όλων των προτύπων τα οποία έχουν αρκετή πυκνότητα γειτόνων , τα οποία ονομάζονται σημεία πυρήνες.

Έστω ένα σύνολο από σημεία στον χώρο , και έστω ϵ μια παράμετρος που καθορίζει την ακτίνα ενός γειτονικού σημείου σε σχέση με κάποιο άλλο , ενώ όλα τα σημεία κατηγοριοποιούνται ως κεντρικά.

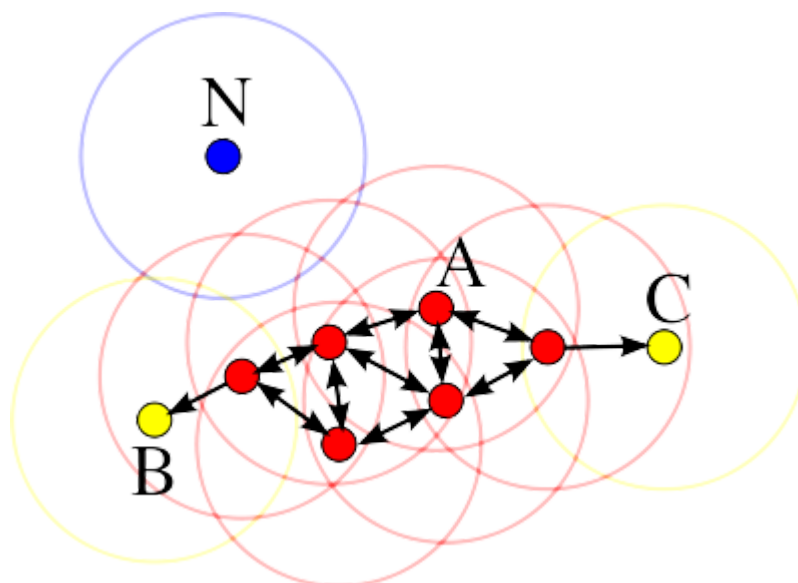
Αλγόριθμος [46][44]

- Ένα σημείο p είναι βασικό-κεντρικό , αν τουλάχιστον MinPts σημεία βρίσκονται σε απόσταση ϵ από αυτό συμπεριλαμβανομένου και του p
- Ένα σημείο q είναι άμεσα προσβάσιμο από το p , αν υπάρχει μια διαδρομή p_1, \dots, p_n , με $p_1 = p$ και $p_n = q$, όπου κάθε p_{i+1} είναι άμεσα προσεγγίσιμο από το p_i , δηλαδή το αρχικό και όλα τα σημεία της διαδρομής πρέπει να είναι κεντρικά, με πιθανή εξαίρεση το q .
- Όλα τα άλλα σημεία που δεν είναι προσβάσιμα από κανένα άλλο σημείο είναι σημεία θόρυβος ή ακραία.

Αν το σημείο p είναι κεντρικό σημείο, τότε σχηματίζει μια συστάδα μαζί με όλα τα σημεία (σημεία πυρίνα ή όχι), τα οποία είναι προσβάσιμα από αυτό , ενώ κάθε συστάδα περιέχει τουλάχιστον ένα κεντρικό σημείο. Τα σημεία που δεν είναι πυρήνες μπορούν να ανήκουν σε μια συστάδα αλλά αποτελούν το 'άκρο' καθώς δεν μπορούν να χρησιμοποιηθούν για την επίτευξη περισσότερων σημείων. Η δυνατότητα προσεγγισιμότητας δεν είναι συμμετρική, καθώς εξ

ορισμού μόνο τα βασικά σημεία μπορούν να φτάσουν σε μη βασικά σημεία. Το αντίθετο δεν ισχύει, επομένως ένα μη βασικό σημείο μπορεί να είναι προσβάσιμο, αλλά τίποτα δεν μπορεί να επιτευχθεί από αυτό, έχοντας ως αποτέλεσμα να απαιτείται μια περαιτέρω έννοια *συνδετικότητας* για να προσδιοριστεί επίσημα η έκταση των συστάδων που βρέθηκαν από το DBSCAN. Έτσι δύο σημεία p και q πυκνά συνδεδεμένα εάν υπάρχει ένα σημείο o έτσι ώστε και τα p και q να είναι προσβάσιμα από το o . Η πυκνή-συνδεσιμότητα είναι συμμετρική οπότε έχουμε τις ιδιότητες :

- Όλα τα σημεία της συστάδας μεταξύ τους πυκνά-συνδεδεμένα.
- Εάν ένα σημείο είναι προσβάσιμο από κάποιο σημείο της συστάδας, τότε είναι και αυτό σημείο της συστάδας.



Σχήμα 1.17. Δημιουργία συστάδων με $\text{minPts}=4$. Το σημείο A και τα άλλα κόκκινα είναι σημεία-πυρήνες, επειδή η περιοχή που περιβάλλει αυτά τα σημεία σε ακτίνα ϵ περιέχει τουλάχιστον 4 σημεία. [46]

Πλεονεκτήματα DBSCAN: [1][44]

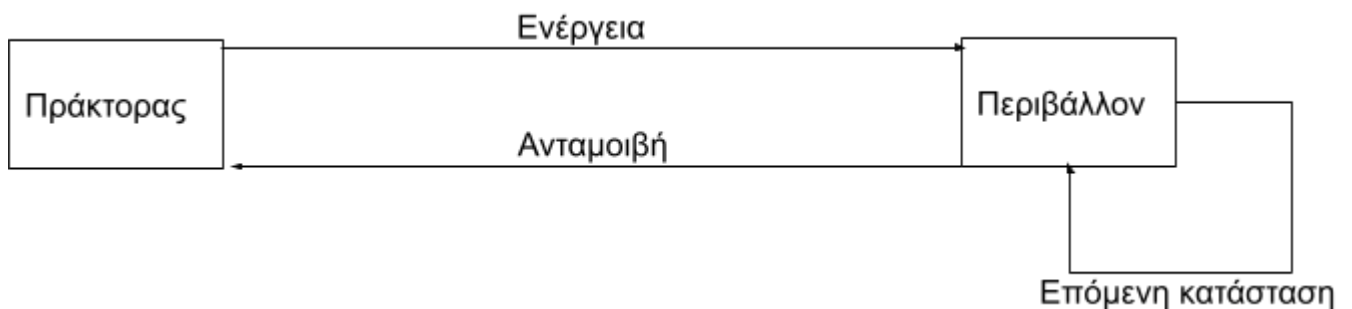
1. Σε αντίθεση με αλγορίθμους, όπως ο k -means, ο DBSCAN δεν απαιτεί τον εκ των προτέρων προσδιορισμό του αριθμού των συστάδων.
2. Μπορεί να ανακαλύψει αυθαίρετα σχήματα συστάδων. Μπορεί να εντοπίσει ακόμα και μια συστάδα, η οποία βρίσκεται γύρω από κάποια άλλη.
3. Έχει καλή ευαισθησία στον θόρυβο και δεν επηρεάζεται από ακραίες τιμές.
4. Δεν εξαρτάται από την σειρά εμφάνισης των δεδομένων
5. Δεν απαιτεί εκ των προτέρων γνώση του πλήθους των συστάδων

Μειονεκτήματα DBSCAN:[1][44]

1. Η ποιότητα των αποτελεσμάτων εξαρτάται από τη μετρική απόστασης που θα χρησιμοποιηθεί. Η πιο κοινή μετρική απόστασης είναι η Ευκλείδεια απόσταση. Ειδικά για πολυδιάστατα δεδομένα, η συγκεκριμένη μετρική είναι σχεδόν άχρηστη. Ωστόσο, αυτό μπορεί να συμβεί σε οποιονδήποτε αλγόριθμο χρησιμοποιεί την Ευκλείδεια απόσταση.
2. Δεν μπορεί να συσταδοποιήσει καλά σύνολα από δεδομένα με μεγάλες διαφορές πυκνότητας, καθώς δεν μπορεί να εντοπιστεί κάποιος συνδυασμός $\text{MinPts-}\epsilon$, που να είναι κατάλληλος για όλες τις συστάδες.
3. Αν τα δεδομένα δεν έχουν γίνει κατανοητά, η επιλογή ενός κατωφλίου ϵ που να έχει νόημα μπορεί να είναι δύσκολη.
4. Έχει ευαισθησία στις παραμέτρους N και ϵ
5. Δεν βρίσκει συστάδες αν τα δεδομένα είναι αραιά

1.4 Εισαγωγή στην μάθηση με ενίσχυση

Ενισχυτική μάθηση (Reinforcement Learning): η ενισχυτική μάθηση ή μάθηση με ενίσχυση εφαρμόζεται σε ένα σύστημα που συνήθως καλείται πράκτορας , αλληλεπιδρά με το περιβάλλον εκτελώντας ενέργειες από τις οποίες αποκομίζει ανταμοιβή (θετική ή αρνητική). [1]
Αν το περιβάλλον είναι δυναμικό τότε με κάθε ενέργεια που εκτελεί ο πράκτορας το περιβάλλον μεταβαίνει σε μια επόμενη κατάσταση .



Σχήμα 1.18. Βασικό σενάριο εφαρμογής μαθησης με ενίσχυση[1]

Η ανταμοιβή εξαρτάται τόσο από την επιλεγμένη ενέργεια όσο και από την τωρινή κατάσταση του περιβάλλοντος. Αν το περιβάλλον είναι στατικό αυτό σημαίνει ότι βρίσκεται πάντα στην ίδια κατάσταση , και η ανταμοιβή που αποκομίζει ο πράκτορας εξαρτάται μόνο από την ενέργεια που εκτελεί. Η ανταμοιβή μπορεί να είναι διαθέσιμη είτε άμεσα μετά από κάθε ενέργεια είτε μετά από μια σειρά ενεργειών.

Οι στόχοι ενός προβλήματος μάθησης με ενίσχυση είναι οι ακόλουθοι:[1]

- Η εκτίμηση της καλύτερης πολιτικής για την επιλογή ενεργειών έτσι ώστε να μεγιστοποιούνται οι ανταμοιβές.
- Η εκτίμηση της κατανομής πιθανότητας των ανταμοιβών.
- Αν το σύστημα είναι δυναμικό , η αποτίμηση της κατάστασης στην οποία βρίσκεται το σύστημα ή η εκτίμηση της πιθανότητας μετάβασης στην επόμενη κατάσταση.

Δύο κεντρικές ιδέες που χαρακτηρίζουν τη μάθηση με ενίσχυση είναι η εξερεύνηση και η εκμετάλλευση.[1]

- Εξερεύνηση : η δοκιμή πλήθους διαφορετικών επιλογών-ενεργειών ώστε να διερευνηθεί η αντίδραση του περιβάλλοντος και η ανταμοιβή που προκύπτει από κάθε ενέργεια.
- Εκμετάλλευση: η χρήση της γνώσης που αποκομίζουμε από την εξερεύνηση των διαφορετικών επιλογών ώστε να επιλέξουμε την καλύτερη ενέργεια σε κάθε κατάσταση.

Όσο μεγαλώνει το διάστημα στο οποίο εξερευνούμε διαφορετικές επιλογές τόσο πιο καλή εκτίμηση έχουμε των πιθανών ανταμοιβών που μπορούμε να πάρουμε από την επιλογή , ωστόσο καθυστερούμε να εκμεταλλευτούμε τη γνώση που αποκομίσουμε ώστε να κάνουμε καλύτερες επιλογές.

Στην πιο απλή περίπτωση ,στατικό περιβάλλον, το σύστημα βρίσκεται σε μια σταθερή κατάσταση και δεν έχει μνήμη , δηλαδή η ανταμοιβή που προσφέρει στον πράκτορα εξαρτάται μόνο από την ενέργεια που αυτός εκτελεί, δηλαδή:

Σε κάθε χρονική στιγμή t ($t=1, \dots, T$) πρέπει να λάβουμε μια απόφαση επιλέγοντας μια ενέργεια ανάμεσα από k πιθανές ενέργειες. Αν συμβολίσουμε με $A(t)$ την ενέργεια που επιλέγουμε τη στιγμή t , κάθε ενέργεια a δίνει μια ανταμοιβή r η οποία εξαρτάται μόνο από το a και όχι από το t . Επομένως έχουμε μια χρονική ακολουθία από ανταμοιβές $R(t)$:

$$A(t) := a \Rightarrow R(t) = q$$

Αν η ανταμοιβή είναι ντετερμινιστική , τότε δοκιμάζουμε όλες τις δυνατές ενέργειες από μια φορά και αφού καταγράψουμε όλες τις δυνατές ενέργειες , επιλέγουμε στη συνέχεια πάντα την ενέργεια με τη μεγαλύτερη ανταμοιβή. Εάν η ανταμοιβή q είναι τυχαία μεταβλητή με άγνωστη κατανομή πιθανότητας , στην περίπτωση αυτή θέλουμε να βρούμε τη μέση ανταμοιβή για κάθε ενέργεια a :

$$\bar{q}(a) = E \{R(t) | A(t) = a\} = \int q p(q|a) dq$$

Στόχος εδώ είναι να επιλέξουμε την κατάλληλη ακολουθία ενεργειών ώστε να μεγιστοποιήσουμε τη συνολική μέση ανταμοιβή στο χρονικό διάστημα T .

Όταν οι αναμενόμενες ανταμοιβές είναι άγνωστες , προκύπτει το δίλημμα μεταξύ εξερεύνησης και εκμετάλλευσης.

Εξερεύνηση : επειδή αρχικά δεν γνωρίζουμε την κατανομή πιθανότητας των ανταμοιβών που προσφέρονται από κάθε μηχανή , πρέπει να τις εκτιμήσουμε δοκιμάζοντας.

Εκμετάλλευση: πρέπει να χρησιμοποιήσουμε την γνώση που αποκομίσαμε επιλέγοντας την καλύτερη αναμενόμενη ανταμοιβή.

Η εξερεύνηση είναι απαραίτητη για να συλλεχθούν στατιστικά δεδομένα και να εκτιμηθεί ποια επιλογή προσφέρει τη μεγαλύτερη αναμενόμενη ανταμοιβή. Από την άλλη μεριά , κατά την διάρκεια της εξερεύνησης μπορεί να χάσουμε χρόνο δοκιμάζοντας μη βέλτιστες ενέργειες.

- Χωρίς εξερεύνηση , δοκιμάζουμε στα τυφλά
- Εξερευνώντας παραπάνω από όσο χρειάζεται , κάνουμε πολλές μη βέλτιστες επιλογές.

Αυτό έχει ως αποτέλεσμα να πρέπει να βρεθεί μια ισορροπία μεταξύ εξερεύνησης/εκμετάλλευσης ώστε να μην επιλέγουμε τυχαία αλλά ούτε και να χάνουμε πάρα πολύ χρόνο σε δοκιμές.

Μια απλή προσέγγιση στη λύση του προβλήματος είναι να γίνει πρώτα μια εξερεύνηση κάθε πιθανής ενέργειας a για ένα συγκεκριμένο χρονικό διάστημα N επαναλήψεων έτσι ώστε να εκτιμηθεί η αναμενόμενη ανταμοιβή $\hat{q}(a)$ για αυτή την ενέργεια. Αφού γίνει αυτό , τότε για πάντα επιλέγουμε την ενέργεια a^* με το μέγιστο $\hat{q}(a^*)$. Το βασικό πρόβλημα της προσέγγισης αυτής είναι η πιθανή σπατάλη μεγάλου αριθμού δοκιμών σε ενέργειες που δεν είναι βέλτιστες καθώς όλες εξετάζονται N φορές. Ένα άπληστος αλγόριθμος θα μπορούσε να αντιμετωπίσει το παραπάνω πρόβλημα από την άμεση εκμετάλλευση της γνώσης που αποκομίζεται από τη στατιστική ανάλυση των ενεργειών που έχει πραγματοποιηθεί μέχρι στιγμής. Με τον τρόπο

αυτό γίνεται μεγιστοποίηση της εκμετάλλευσης καθώς χρησιμοποιούνται οι εκτιμήσεις $\hat{q}_i(a)$ από την πρώτη χρονική στιγμή $t=1$. Ταυτόχρονα όμως γίνεται ελαχιστοποίηση της εξερεύνησης, αφού είναι δυνατόν να υπάρξουν ενέργειες που δεν θα εξερευνηθούν ποτέ επειδή θα ισχύει $\hat{q}_i(x^*)$ για όλες τις χρονικές στιγμές t . Αυτή η ελαχιστοποίηση της εξερεύνησης αποτελεί μειονέκτημα του άπληστου αλγορίθμου καθώς ενδέχεται να αγνοήσουμε ενέργειες που δεν έχουν εξερευνηθεί ποτέ. Μια εναλλακτική προσέγγιση για τον μετριάσμο αυτού του προβλήματος είναι να μην επιλέγουμε πάντα την βέλτιστη τρέχουσα ενέργεια αλλά να την επιλέγουμε με πιθανότητα $1-\epsilon$, και να επιλέγουμε μια τυχαία ενέργεια με πιθανότητα ϵ , όπου ϵ μικρός θετικός αριθμός. Αυτό μας δίνει την δυνατότητα να εξερευνήσουμε και άλλες ενέργειες πλην της τρέχουσας βέλτιστης ενέργειας.

1.4.1 Διαδικασίες αποφάσεων Markov

Μια πιο συνηθισμένη περίπτωση είναι όταν ο πράκτορας αλληλεπιδρά με ένα περιβάλλον που έχει <<μνήμη>>. Οι βασικές υποθέσεις του προβλήματος είναι οι εξής:[1]

- Ο πράκτορας αλληλεπιδρά με το περιβάλλον σε διακριτές χρονικές στιγμές $t=1,2,\dots$
- Σε κάθε χρονική στιγμή t :

Το περιβάλλον βρίσκεται σε μια κατάσταση $S(t)$ από N δυνατές καταστάσεις.

Ο πράκτορας επιλέγει μια ενέργεια $A(t)$ από K δυνατές ενέργειες.

Ο πράκτορας λαμβάνει μια ανταμοιβή $R(t)$ η οποία εξαρτάται από την κατάσταση $S(t)$ και την ενέργεια $A(t)$.

Το περιβάλλον μεταβαίνει σε μια νέα κατάσταση $S(t+1)$ η οποία επίσης εξαρτάται από την προηγούμενη κατάσταση $S(t)$ και την επιλεγμένη ενέργεια $A(t)$.

Το πρόβλημα που διατυπώνεται με βάση τις παραπάνω υποθέσεις είναι γνωστό ως διαδικασία αποφάσεων. Επειδή η κατάσταση του περιβάλλοντος εξαρτάται μόνο από την αμέσως προηγούμενη χρονική στιγμή, λέμε ότι το σύστημα έχει την ιδιότητα Markov και το πρόβλημα ονομάζεται διαδικασία αποφάσεων Markov (Markov Decision Process-MDP). Η διαδικασία περιγράφεται από την πιθανότητα το σύστημα να βρίσκεται στην κατάσταση s' και να προσφέρει ανταμοιβή r στη χρονική στιγμή $t+1$ δεδομένου ότι στην χρονική στιγμή t το περιβάλλον βρισκόταν στην κατάσταση s και επιλέχθηκε η ενέργεια a :
 $p(s',r|s,a)=Pr(S(t+1)=s', R(t+1)=r|S(t)=s, A(t)=a)$

1.4.2 Πολιτική Κέρδος και αξία

Κεντρική έννοια στις διεργασίες αποφάσεων Markov είναι η πολιτική, η οποία ορίζεται ως η μέθοδος επιλογής της ενέργειας $A(t)=a$ του πράκτορα βασιζόμενη στην κατάσταση $S(t)=s$ του περιβάλλοντος. Η πολιτική μπορεί είτε να επιλέγεται εκ των προτέρων είτε να διαμορφώνεται δυναμικά με κάποια μέθοδο μάθησης βασισμένη στις προηγούμενες εμπειρίες του πράκτορα. Η πολιτική περιγράφεται από τη συνάρτηση υπό συνθήκη πιθανότητας της ενέργειας a με δεδομένη την τρέχουσα κατάσταση s . Η συνάρτηση πολιτικής είναι η εξής:

$$\pi(a,s)=p(A(t)=a|S(t)=s)$$

Αφού η συνάρτηση π αποτελεί κατανομή πιθανότητας ισχύει ότι

$$\sum_{\alpha} \pi(\alpha, s) = 1$$

Ολόκληρη η σειρά των ενεργειών που εκτελεί ο πράκτορας για $t=1,2,..T$ ονομάζεται εργασία. Σε κάποιες περιπτώσεις οι εργασίες έχουν πεπερασμένο χρόνο και ονομάζονται εργασίες πεπερασμένου ορίζοντα ή επεισοδιακές, ενώ σε άλλες περιπτώσεις που η εργασία δεν τελειώνει ποτέ καλούνται εργασίες άπειρου ορίζοντα ή συνεχείς εργασίες.

Καθώς ο πράκτορας εκτελεί ενέργειες, εισπράττει ανταμοιβές $R(t)$, για $t=1,2,..,T$. Το κριτήριο αξιολόγησης μιας πολιτικής είναι το κέρδος $G(t)$, το οποίο για οποιαδήποτε χρονική στιγμή ορίζεται ως το άθροισμα των ανταμοιβών από εκείνη τη στιγμή ως το πέρας της εργασίας. Επειδή η εργασία μπορεί να είναι άπειρου ορίζοντα, συνήθως ορίζουμε το κέρδος χρησιμοποιώντας ένα συντελεστή λήθης (forgetting factor), $0 < \gamma < 1$, ώστε το άθροισμα να συγκλίνει ακόμα και αν έχουμε άπειρους όρους:

$$G(t) = R(t+1) + \gamma R(t+2) + \gamma^2 R(t+3) + \dots = \sum_{k=0}^{\infty} \gamma^k R(t+k+1)$$

ή αναδρομικά

$$G(t) = R(t+1) + \gamma G(t+1)$$

Στη συνέχεια θα ορίσουμε ως αξία $v_{\pi}(s)$ μιας κατάστασης s , υπό την πολιτική π , το αναμενόμενο κέρδος αν ξεκινήσουμε από την κατάσταση s και εφαρμόσουμε την πολιτική μας από το σημείο αυτό και μετά: $v_{\pi}(s) = E_{\pi} \{G(t) | S(t) = s\}$ όπου $E_{\pi} \{.\}$ συμβολίζουμε τη μέση τιμή μιας τυχαίας μεταβλητής με δεδομένο ότι ο πράκτορας ακολουθεί την πολιτική π σε κάθε χρονική στιγμή t .

Το πρόβλημα που αντιμετωπίζουμε στη θεωρία των διεργασιών αποφάσεων Markov είναι η επιλογή της βέλτιστης πολιτικής π η οποία μεγιστοποιεί το αναμενόμενο κέρδος από οποιαδήποτε αρχική κατάσταση s . Η συνάρτηση που ζητάμε να μεγιστοποιήσουμε, για κάθε s είναι: $J(s) = E_{\pi} \{G(0) | S(0) = s\} = v_{\pi}(s)$

Η βέλτιστη πολιτική για την κατάσταση s είναι αυτή που μεγιστοποιεί την αξία της s :

$$\pi^*(s) = \arg \max_{\pi} v_{\pi}(s) \text{ και η βέλτιστη αξία της } s \text{ είναι: } v^*(s) = \max_{\pi} v_{\pi}(s) = v_{\pi^*}(s).$$

Για την επίλυση του προβλήματος ορίζουμε την συνάρτηση Q , η οποία δίνει το αναμενόμενο κέρδος όταν ξεκινάμε από την κατάσταση s επιλέγοντας την ενέργεια a υπό την πολιτική π :

$$q_{\pi}(s, a) = E_{\pi} \{G(t) | S(t) = s, A(t) = a\}. \text{ Η σχέση που συνδέει τη συνάρτηση αξίας της κατάστασης } s \text{ την συνάρτηση } q \text{ και την πολιτική } \pi \text{ είναι: [1]}$$

$$v_{\pi}(s) = \sum_{\alpha} \pi(\alpha, s) q_{\pi}(s, \alpha).$$

Με βάση αυτή την εξίσωση προκύπτει ότι εάν γνωρίζουμε την συνάρτηση $q_{\pi}(s, a)$ για την βέλτιστη πολιτική π , τότε η αξία $v_{\pi}(s)$ μεγιστοποιείται αν επιλέξουμε την ενέργεια a για την οποία η $q_{\pi}(s, a)$ παίρνει τη μέγιστη τιμή. Δηλαδή η βέλτιστη πολιτική π^* είναι:

$$\pi^*(\alpha^*, s) = 1 \text{ αν } \alpha^* = \arg \max_{\alpha} (q_{\pi^*}(s, \alpha)) \text{ διαφορετικά } 0.$$

Ο συνδυασμός των δύο προηγούμενων εξισώσεων δίνει τη μέγιστη αξία s :

$$v_{\pi^*}(s) = \max_{\alpha} q_{\pi^*}(s, \alpha).$$

Η παραπάνω μέθοδος εφαρμόζεται ευρέως σε συνδυασμό με διαφορετικούς αλγορίθμους ώστε

να υπολογίσει την μέγιστη αξία , μέθοδος αυτή είναι γνωστή ως μάθηση Q (Q learning) από την χρήση της συνάρτησης Q. Η τεχνική Q-learning θεωρείται ως αλγόριθμος ‘χωρίς μοντέλο’ για να μάθει την αξία μιας δράσης σε μια συγκεκριμένη κατάσταση , αντλεί δεδομένα και πληροφορίες από δράσεις που βρίσκονται εκτός της τρέχουσας πολιτικής , ενώ μπορεί να χειριστεί προβλήματα με στοχαστικές μεταβάσεις και ανταμοιβές χωρίς να απαιτείται προσαρμογή. Για οποιαδήποτε διαδικασία Markov , η τεχνική Q -learning βρίσκει μια βέλτιστη πολιτική με την έννοια της μεγιστοποίησης της αναμενόμενης αξίας της συνολικής ανταμοιβής σε οποιοδήποτε και σε όλα τα διαδοχικά βήματα, ξεκινώντας από την τρέχουσα κατάσταση. Το Q -learning μπορεί να προσδιορίσει μια βέλτιστη επιλογή πολιτικής δράσης για κάθε δεδομένο Μαρκοβιανής διαδικασίας, δεδομένου του απεριόριστου χρόνου εξερεύνησης και μιας μερικώς τυχαίας πολιτικής. [47]

1.4.3 Q-learning

Στο μοντέλο MDP η επανάληψη πολιτικής και η επανάληψη αξίας είναι δύο μέθοδοι δυναμικού προγραμματισμού για την εύρεση της καλύτερης πολιτικής. Ωστόσο, αυτές οι δύο μέθοδοι απαιτούν προηγούμενη γνώση του δυναμικού περιβάλλοντος, η οποία είναι δύσκολο να εκτιμηθεί εκ των προτέρων από πρακτικής άποψης. Είναι δύσκολο δηλαδή να αποκτήσουμε προηγούμενη γνώση για ένα σενάριο . Σε αυτήν την περίπτωση, η εξερεύνηση της σχέσης κατάσταση-ενέργειας-ανταμοιβής απαιτείται για το περιβάλλον. Για να επιτευχθεί αυτή η πρόκληση , η τεχνική Q-learning φαίνεται ως εξαιρετική επιλογή μάθησης με ενίσχυση λόγω της απλότητας και του χαμηλού υπολογιστικού κόστους.

Στη μέθοδο Q-learning η τιμή της συνάρτησης κατάσταση-ενέργεια (Q-function) χρησιμοποιείται για να χαρακτηρίσει τις πολιτικές. Ξεκινά από μια δεδομένη κατάσταση και, στη συνέχεια, συσσωρεύει την αξία της σύμφωνα με τις ανταμοιβές που έλαβε ο πράκτορας όταν έκανε μια σειρά ενεργειών σύμφωνα με την πολιτική. Η συνάρτηση Q ορίζεται ως :

$$Q^{\pi}(s, a) = E \left(\sum_{k=0}^{\infty} \gamma^k R_k | s_0 = s, a_0 = a \right)$$

Η βέλτιστη συνάρτηση Q , $Q^*(s, a)$, είναι εκείνη που μεγιστοποιεί τις μειωμένες αθροιστικές ανταμοιβές, $Q^*(s, a) = \max_{\pi} Q^{\pi}(s, a)$. Σύμφωνα με την αρχή της βελτιστοποίησης του Bellman μια άπληστη πολιτική μπορεί να χρησιμοποιηθεί για την επίλυση αυτού του προβλήματος βελτιστοποίησης. Στην άπληστη πολιτική, επιλέγεται μια ενέργεια με τη μεγαλύτερη τιμή Q σε κάθε κατάσταση. Αυτό σημαίνει ότι για κάθε κατάσταση $s \in S$, όταν το $Q^*(s, a)$ είναι γνωστό , η βέλτιστη πολιτική είναι να ενέργεια a με την υψηλότερη τιμή $Q^*(s, a)$: $\pi^*(s) = \max_{a \in A} Q^*(s, a)$. Στον αλγόριθμο Q-learning , το $Q^*(s, a)$ ενημερώνεται αναδρομικά χρησιμοποιώντας την τρέχουσα κατάσταση s , την επόμενη κατάσταση s' , την ενέργεια a στην κατάσταση s , και η ανταμοιβή R που λαμβάνεται από το περιβάλλον λόγω της ενέργειας a στην τρέχουσα κατάσταση. Ο τύπος ενημέρωσης δίνεται ως :

$$Q(s, a) \leftarrow Q(s, a) + \alpha [R + \gamma + \max_{a'} Q(s', a') - Q(s, a)] \text{ (εξίσωση q-learning)}$$

όπου α είναι το ποσοστό εκμάθησης.

Ο αλγόριθμος RL έχει ένα δίλημμα μεταξύ εξερεύνησης και εκμετάλλευσης. Η εξερεύνηση σημαίνει τη συλλογή ενημερωτικών δεδομένων για το περιβάλλον, λαμβάνοντας νέες ενέργειες για τον δεδομένο χώρο. Εκμετάλλευση σημαίνει διατήρηση της διαδικασίας αρκετά καλά σύμφωνα με τις διαθέσιμες γνώσεις που αποκτήθηκαν προηγουμένως. Για την εξισορρόπηση της ανάγκης για εξερεύνηση και εκμετάλλευση, δύο υιοθετημένες στρατηγικές είναι η ϵ -άπληστη αναζήτηση (e-greedy search) και η μέθοδος επιλογής δράσης softmax (softmax action-selection method), στις οποίες θα αναφερθούμε εν συντομία στην συνέχεια.

ϵ -greedy search : Στον αλγόριθμο αυτό, ο πράκτορας χρησιμοποιεί την πιθανότητα ϵ για να αποφασίσει εάν θα εκμεταλλευτεί τον πίνακα Q ή θα εξερευνήσει τις νέες επιλογές. Επειδή το ϵ είναι συνήθως μικρό, ο πράκτορας τείνει να επιλέξει την ενέργεια που ικανοποιεί το μέγιστο Q (s, a) ($\max_a Q(s, a)$). Σε σπάνιες περιπτώσεις, με πιθανότητα ϵ , ο πράκτορας λαμβάνει μια τυχαία ενέργεια με σκοπό να βιώσει όσο το δυνατόν περισσότερες ενέργειες και αποτελέσματα.

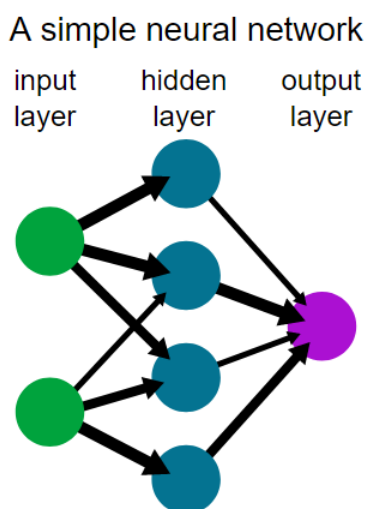
softmax action-selection: Λόγω της μικρής πιθανότητας τυχαίας επιλογής, η αναζήτηση ϵ -greedy μπορεί να πετύχει μόνο μια μη βέλτιστη πολιτική. Για την επίλυση αυτού του προβλήματος, η τυχαία επιλογή εγκαταλείπεται στη μέθοδο softmax action-selection. Αντ' αυτού, η ενέργεια a που λαμβάνεται στο στάδιο s επιλέγεται με πιθανότητα

$$P_r(a) = \frac{e^{Q(s,a)/\tau}}{\sum_{i=1}^m e^{Q(s,a_i)/\tau}} \quad (\text{εξίσωση softmax})$$

όπου τ καθορίζει την τάση επιλογής. Αν το τ είναι κοντά στο 0, ο πράκτορας προτιμά να επιλέγει ενέργειες με την υψηλότερη τιμή Q. Εάν το τ γίνει μεγαλύτερο, οι τάσεις της πολιτικής προς την τυχαία επιλογή, καθώς όλες οι ενέργειες τείνουν να έχουν την ίδια πιθανότητα επιλογής.

1.5 Εισαγωγή στο νευρωνικό δίκτυο

Νευρωνικό δίκτυο ονομάζεται ένα κύκλωμα διασυνδεδεμένων νευρώνων , [48] , παίρνοντας την ονομασία από τους ανθρώπινους βιολογικούς νευρώνες. Στην περίπτωση τεχνητών νευρώνων, πρόκειται για ένα αφηρημένο αλγοριθμικό κατασκεύασμα το οποίο εμπίπτει στον τομέα της υπολογιστικής νοημοσύνης, όταν στόχος του νευρωνικού δικτύου είναι η επίλυση κάποιου υπολογιστικού προβλήματος, καθώς και η υπολογιστική προσομοίωση της λειτουργίας των βιολογικών νευρωνικών δικτύων με βάση κάποιο μαθηματικό μοντέλο τους. Το νευρωνικό δίκτυο είναι ένα δίκτυο από απλούς υπολογιστικούς κόμβους (νευρώνες, νευρώνια), διασυνδεδεμένους μεταξύ τους.[48]. Αποτελούνται από διασυνδεδεμένα στοιχεία που έχουν την ικανότητα να ανταποκρίνονται σε ερεθίσματα που δέχονται στην είσοδο τους και να μαθαίνουν να προσαρμόζονται στο περιβάλλον τους.



Σχήμα 1.19. Παράδειγμα τεχνητού νευρωνικού δικτύου [48]

Οι νευρώνες είναι τα δομικά στοιχεία του δικτύου. Κάθε τέτοιος κόμβος δέχεται ένα σύνολο αριθμητικών εισόδων από διαφορετικές πηγές (είτε από άλλους νευρώνες, είτε από το περιβάλλον), επιτελεί έναν υπολογισμό με βάση αυτές τις εισόδους και παράγει μία έξοδο. Η εν λόγω έξοδος είτε κατευθύνεται στο περιβάλλον, είτε τροφοδοτείται ως είσοδος σε άλλους νευρώνες του δικτύου. Υπάρχουν τρεις τύποι νευρώνων:

- οι νευρώνες εισόδου (δενδρίτες)
- οι νευρώνες εξόδου (άξονας)
- οι υπολογιστικοί νευρώνες ή κρυμμένοι νευρώνες (συνάψεις- σημεία στα οποία ενώνονται οι διακλαδώσεις του άξονα ενός νευρώνα με τους δενδρίτες άλλων νευρώνων).

Οι νευρώνες εισόδου δεν επιτελούν κανέναν υπολογισμό, μεσολαβούν απλώς ανάμεσα στις περιβαλλοντικές εισόδους του δικτύου και στους υπολογιστικούς νευρώνες. Οι νευρώνες εξόδου διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου. Οι υπολογιστικοί νευρώνες πολλαπλασιάζουν κάθε είσοδό τους με το αντίστοιχο συναπτικό βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων. Το άθροισμα αυτό τροφοδοτείται ως όρισμα στη συνάρτηση ενεργοποίησης, την οποία υλοποιεί εσωτερικά κάθε κόμβος. Η τιμή που

λαμβάνει η συνάρτηση για το εν λόγω όρισμα είναι και η έξοδος του νευρώνα για τις τρέχουσες εισόδους και βάρη.[48]

- Το μοντέλο του τεχνητού νευρώνα

Ο τεχνητός νευρώνας είναι ένα υπολογιστικό μοντέλο τα μέρη του οποίου μπορεί να αντιστοιχιστούν άμεσα με αυτά του βιολογικού νευρώνα. Οι επιστήμονες McCulloch και Pitts όρισαν ένα απλό μοντέλο της δραστηριότητας του νευρώνα η κατάσταση του οποίου περιγράφεται από έναν δυαδικό αριθμό :[1]

- 0-> ο νευρώνας είναι αδρανής (δεν πυροδοτεί παλμούς)
- 1-> ο νευρώνας πυροδοτεί παλμούς στην μέγιστη ταχύτητα

Ένα τεχνητός νευρώνας δέχεται κάποια σήματα εισόδου X_1, X_2, \dots, X_n και κάθε τέτοιο σήμα μεταβάλλεται από μια τιμή βάρους W_i (συναπτικά βάρη), ο ρόλος της οποίας είναι αντίστοιχος του ρόλου της σύναψης στον βιολογικό νευρώνα. Η τιμή του βάρους μπορεί να είναι θετική (ενισχυτικές συνάψεις) ή αρνητική (ανασταλτικές συνάψεις).

Αν ισχύει $w_1x_1 + \dots + w_nx_n > \theta$ τότε ο νευρώνας πυροδοτεί παλμούς αλλιώς μένει αδρανής, όπου θ είναι ένα κατώφλι.

Από μαθηματικής πλευράς αυτό ισοδυναμεί με την ποσότητα :

$$u = \sum_{i=1}^n w_i x_i - \theta$$

να είναι θετική ή αρνητική. Οπότε η έξοδος του νευρώνα θα είναι :

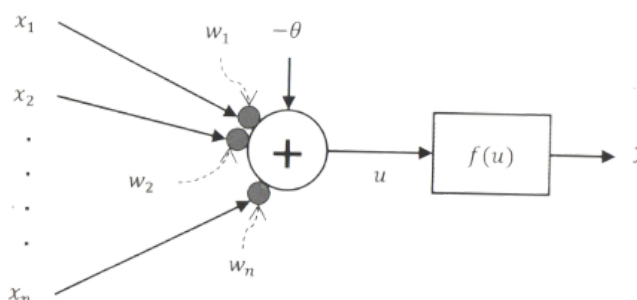
$$y = f(u) = 0 \text{ αν } u \leq 0 \text{ και } y = f(u) = 1 \text{ αν } u > 0$$

Η ποσότητα u καλείται διέγερση του νευρώνα και προκύπτει από τις εισόδους, ενώ η συνάρτηση $f(u)$ ονομάζεται βηματική συνάρτηση 0/1. Ο τεχνητός νευρώνας περιλαμβάνει επίσης την εφαρμογή ενός εξωτερικού παράγοντα $w_0 = -\theta$ ή (b) που ονομάζεται πόλωση (bias) και ισούται με το συναπτικό βάρος της σταθερής εισόδου $x_0 = 1$, οπότε μπορούμε να ξαναγράψουμε την προηγούμενη εξίσωση ως [1] :

$$u = \sum_{i=0}^n w_i x_i$$

Η πόλωση μπορεί να επιφέρει την αύξηση ή την μείωση της διέγερσης της συνάρτησης ενεργοποίησης ανάλογα με το αν είναι θετική ή αρνητική.

Στο επόμενο σχήμα παρουσιάζεται το μοντέλο ενός νευρώνα.



Σχήμα 1.20. Μοντέλο νευρώνα κατά McCulloch και Pitts [1]

Το σώμα του τεχνητού νευρώνα χωρίζεται σε δύο μέρη , τον αθροιστή , ο οποίος προσθέτει τα επηρεασμένα από τα βάρη σήματα εισόδου παράγοντας την ποσότητα u και την συνάρτηση ενεργοποίησης (ένα είδος φίλτρου που διαμορφώνει την τελική τιμή εξόδου y) , σε συνάρτηση με την ποσότητα u και την τιμή κατωφλίου της συνάρτησης ενεργοποίησης.

Στόχος του νευρωνικού δικτύου είναι να προσεγγίσει τη συνάρτηση μέσω της οποίας σχετίζεται η είσοδος με την έξοδο , με όσο το δυνατόν μεγαλύτερη ακρίβεια. Η εκπαίδευση του δικτύου γίνεται συνήθως με επίβλεψη ή χωρίς επίβλεψη ή κάποιες φορές με ενίσχυση.

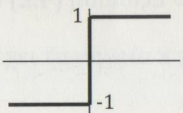
Η μάθηση γίνεται μέσω κάποιων παραδειγμάτων εκπαίδευσης και ενός αλγόριθμου εκπαίδευσης και έχει ως αποτέλεσμα τη βελτίωση της απόδοσης του δικτύου. Ο αλγόριθμος αυτός είναι μια επαναληπτική διαδικασία με την οποία αλλάζουν κάποιοι παράμετροι του δικτύου (π.χ. βάρη , κατώφλια) έτσι ώστε να μειωθεί το σφάλμα μεταξύ επιθυμητής και πραγματικής εξόδου.

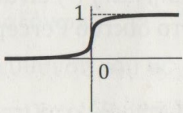
Υπάρχουν αρκετά μοντέλα για ένα νευρώνα που αποκλίνουν από αυτό του McCulloch και Pitts , με την πιο σημαντική διαφορά να εντοπίζεται στη μη γραμμική συνάρτηση που χρησιμοποιείται στην έξοδο που ονομάζεται συνάρτηση ενεργοποίησης του νευρώνα εναλλακτικές μορφές είναι: [1]

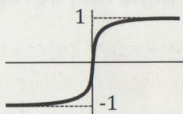
- Βηματική $-1/1$ (step function $-1/1$)

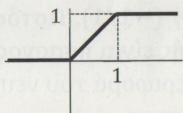
$$f(u) = \begin{cases} -1 & \text{αν } u \leq 0 \\ 1 & \text{αν } u > 0 \end{cases}$$

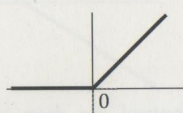
Η συνάρτηση αυτή συμβολίζεται επίσης με sgn από τη λατινική λέξη "signum", που σημαίνει πρόσημο.

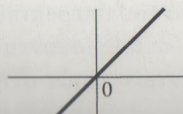

- Σιγμοειδής (sigmoid)

$$f(u) = \frac{1}{1 + e^{-u}}$$

- Υπερβολική εφαπτομένη (hyperbolic tangent)

$$f(u) = \tanh(u) = \frac{1 - e^{-u}}{1 + e^{-u}}$$

- Συνάρτηση κατωφλίου (threshold function)

$$f(u) = \begin{cases} 0 & \text{αν } u \leq 0 \\ u & \text{αν } 0 < u < 1 \\ 1 & \text{αν } u \geq 1 \end{cases}$$

- Συνάρτηση ράμπας (ramp function), γνωστή και ως ανορθωμένη γραμμική μονάδα (Rectified Linear Unit - ReLU)

$$f(u) = \begin{cases} -1 & \text{αν } u \leq 0 \\ u & \text{αν } u > 0 \end{cases}$$

- Γραμμική (linear)

$$f(u) = u$$


Τα νευρωνικά δίκτυα έχουν τα ακόλουθα πλεονεκτήματα:

1. Μη γραμμικότητα. Ένας τεχνητός νευρώνας μπορεί να είναι είτε γραμμικός είτε όχι. Ένα νευρωνικό δίκτυο αποτελούμενο από διασυνδεδεμένους μη γραμμικούς νευρώνες είναι, από τη φύση του μη γραμμικό. Επιπλέον αυτή η μη γραμμικότητα είναι «ειδικού τύπου» υπό την έννοια ότι είναι κατανεμημένη σε όλη την έκταση του δικτύου.
2. Προσαρμοστικότητα. Τα νευρωνικά δίκτυα έχουν δυνατότητα να προσαρμόζουν τα συνοπτικά βάρη τους ανάλογα με τις μεταβολές που γίνονται στο περιβάλλον τους. Ένα νευρωνικό δίκτυο που είναι εκπαιδευμένο να λειτουργεί σε συγκεκριμένο περιβάλλον, μπορεί εύκολα να επανεκπαιδευτεί ώστε να λειτουργεί και σε μεταβαλλόμενο περιβάλλον.
3. Αντιστοίχιση εισόδου-εξόδου. η αντιστοίχιση εισόδου εξόδου δημιουργείται μέσω μιας διαδικασίας μάθησης μέσω παραδειγμάτων προσπαθώντας να μειωθούν οι διαφορές της επιθυμητής απόκρισης και της πραγματικής στην προσπάθεια να περιοριστούν οι αυθαίρετες εκτιμήσεις στο χώρο του σήματος εισόδου.
4. Ενδεικτική απόκριση. Ένα νευρωνικό δίκτυο μπορεί να σχεδιαστεί ώστε να παρέχει πληροφορία όχι μόνο για το ποιό πρότυπο θα επιλεγεί, αλλά και σχετικά με τον βαθμό εμπιστοσύνης στην ληφθείσα απόφαση. Ο βαθμός εμπιστοσύνης μπορεί να χρησιμοποιηθεί για να απορρίψει την ακρίβεια της αναγνώρισης.
5. Ανοχή σε βλάβες. Λόγω της φύσης και της δομής τους τα νευρωνικά δίκτυα παρουσιάζουν μεγάλη ανθεκτικότητα καθώς και ποιοτική απόδοση σε ακραίες ή επισφαλείς συνθήκες. Ένα νευρωνικό δίκτυο μπορεί να παρουσιάσει μειωμένη απόδοση αλλά είναι εξαιρετικά δύσκολο να παρουσιάσει καταστροφική αποτυχία στην λειτουργία του.
6. Δυνατότητα υλοποίησης με τεχνολογία VLSI. .
7. Σχετική με το περιεχόμενο πληροφορία. Η γνώση ή πληροφορία που παρουσιάζεται στην δομή του κάθε νευρώνα είναι επηρεασμένη από την συνολική δραστηριότητα όλων των άλλων νευρώνων του δικτύου.
8. Ομοιομορφία ανάλυσης και σχεδίασης. Υπάρχει μια γενική αρχή ότι οι νευρώνες ως δομικά στοιχεία είναι κοινοί σε όλα τα νευρωνικά δίκτυα. Σαν αποτέλεσμα αυτού καθίσταται εφικτή η χρήση των ίδιων θεωριών και αλγορίθμων μάθησης σε διαφορετικές εφαρμογές των νευρωνικών δικτύων.
9. Ικανότητα γενίκευσης , δηλαδή την ικανότητα του δικτύου να αποδίδει εξίσου καλά να εκτιμά δηλαδή με επιτυχία τους στόχους για πρότυπα εισόδου που δεν έχει ξαναδεί. Η ικανότητα αυτή αποτελεί το πιο βασικό χαρακτηριστικό ενός νευρωνικού δικτύου καθώς στα περισσότερα προβλήματα δεν γνωρίσουμε από πριν τις καταστάσεις. Η ικανότητα γενίκευσης επηρεάζεται από :
 - α) Τον αριθμό των διανυσμάτων εκπαίδευσης και από τον αν αυτά αναπαριστούν ικανοποιητικά τον υποχώρο από τον οποίο προέρχονται.
 - β) Την πολυπλοκότητα του προβλήματος.
 - γ) Το μέγεθος του τεχνικού νευρωνικού δικτύου.

Είδη Δικτύου :

1. Δίκτυα πρόσθιας τροφοδότησης. Είναι δίκτυα στα οποία το σήμα διαδίδεται έτσι ώστε να μην υπάρχει νευρώνας που η έξοδος του είναι είσοδος κάποιου νευρώνα του ίδιου ή προηγούμενου στρώματος, το σήμα δηλαδή μεταφέρεται μόνο προς την μια κατεύθυνση από την είσοδο του προς την έξοδο.
2. Δίκτυα με ανατροφοδότηση. Στα δίκτυα με ανατροφοδότηση τα σήματα ταξιδεύουν και προς τις δύο κατευθύνσεις καθώς υπάρχουν βρόγχοι στο δίκτυο. Τα δίκτυα αυτά είναι αρκετά ισχυρά αλλά και πολύπλοκα, ενώ η κατάσταση τους αλλάζει συνεχώς μέχρι να φτάσουν σε μια κατάσταση ισορροπίας.

1.4.1 Το δίκτυο Perceptron

Το perceptron (αισθητήρας) είναι η πιο απλή μορφή ενός νευρωνικού δικτύου (αποτελείται από ένα μεμονωμένο νευρώνα με προσαρμόσιμα συνοπτικά βάρη) που χρησιμοποιείται για την ταξινόμηση προτύπων τα οποία είναι γραμμικά διαχωρίσιμα. Το ζητούμενο του δικτύου perceptron είναι η αυτόματη εκμάθηση των παραμέτρων του συστήματος ώστε να επιτυγχάνεται ο επιθυμητός στόχος. Ο Rosenblatt απέδειξε ότι εάν τα πρότυπα (διανύσματα) που χρησιμοποιούνται για την εκπαίδευση του Perceptron αντλούνται από δύο γραμμικά διαχωρίσιμες κλάσεις, τότε ο αλγόριθμος του Perceptron συγκλίνει και τοποθετεί τη διαχωριστική επιφάνεια απόφασης με τη μορφή ενός υπερεπίπεδου μεταξύ των δύο κλάσεων. Η απόδειξη της σύγκλισης του Perceptron είναι γνωστή ως θεώρημα σύγκλισης του Perceptron.[1] Το perceptron που βασίζεται σ' ένα μεμονωμένο νευρώνα περιορίζεται στην ταξινόμηση προτύπων που ανήκουν σε μόνο δύο κλάσεις. Ελεγκτώντας το επίπεδο εξόδου του perceptron ώστε να περιλαμβάνει περισσότερους του ενός νευρώνες, **μπορούμε ν εκτελέσουμε** ταξινόμηση προτύπων που ανήκουν σε περισσότερες από δύο κλάσεις εφόσον αυτές είναι γραμμικά διαχωρίσιμες.

Στο μοντέλο Perceptron ο ένας και μοναδικός νευρώνας μπορεί να χαρακτηριστεί από την συνάρτηση : [1]

$$y = f\left(\sum_{i=1}^n w_i x_i + b\right)$$

όπου y είναι η έξοδος , x_i είναι οι είσοδοι , w_i είναι τα συναπτικά βάρη , b είναι η πόλωση και $f()$ η συνάρτηση ενεργοποίησης , η οποία είναι είτε η βηματική 0/1 είτε η βηματική -1/1. Η έξοδος είναι ένας δυαδικός αριθμός ανάλογος της συνάρτησης ενεργοποίησης , ενώ οι παράμετροι που ρυθμίζουν την συμπεριφορά του νευρώνα είναι το διάνυσμα των συναπτικών βαρών και η πόλωση. Ο αλγόριθμος στην ουσία δημιουργεί ένα υπερεπίπεδο ώστε να χωρίζονται οι κλάσεις , πρότυπα με θετικές τιμές βρίσκονται στην μια πλευρά του επιπέδου ενώ πρότυπα με αρνητικές τιμές στην άλλη πλευρά.

Στόχος του νευρωνικού δικτύου είναι η αυτόματη εκμάθηση των παραμέτρων του συστήματος ώστε να επιτυγχάνεται ο επιθυμητός στόχος (εύρεση διαχωριστικού επιπέδου). Στην μάθηση με εκπαίδευση υπάρχει η τιμή στόχος $t^{(p)}$ για κάθε πρότυπο εκπαίδευσης p . Το δίκτυο μαθαίνει να

προσαρμόζει τις παραμέτρους w_n λαμβάνοντας υπόψη του τα επαυξημένα πρότυπα εκπαίδευσης x_n και τους στόχους των προτύπων αυτών χρησιμοποιώντας κάποιον επαναληπτικό αλγόριθμο. Ο κανόνας εκπαίδευσης Perceptron είναι γνωστός ως κανόνας σταθερής αύξησης [1], και είναι επαναληπτικός. Τα πρότυπα παρουσιάζονται στο δίκτυο με κυκλική σειρά και όταν τελειώνουν επαναλαμβάνονται από την αρχή. Ένας πλήρης κύκλος ονομάζεται εποχή. Ο κανόνας τροποποιεί το επαυξημένο διάνυσμα των συναπτικών βαρών w μόνο όταν υπάρχει σφάλμα ταξινόμησης, δηλαδή όταν ο στόχος $t^{(p)}$ για το πρότυπο p διαφέρει από την έξοδο του δικτύου :

$$y^{(p)} = f(w(k-1)^T x^{(p)})$$

όπου $k-1$ είναι το επαυξημένο διάνυσμα των συναπτικών βαρών μετά την επανάληψη $k-1$. Όταν υπάρχει σφάλμα η διόρθωση των βαρών γίνεται με την προσθήκη ή την αφαίρεση ενός ποσοστού του προτύπου $x^{(p)}$. Αν κατά την επανάληψη k εισάγεται το πρότυπο p , ο κανόνας διόρθωσης είναι :

$$w(k) = w(k-1) + \beta + (t^{(p)} - y^{(p)})x^{(p)}$$

όπου β είναι μια παράμετρος που καλείται βήμα η ρυθμός εκπαίδευσης και ρυθμίζει το μέγεθος διόρθωσης, και συνήθως είναι μικρός θετικός αριθμός.

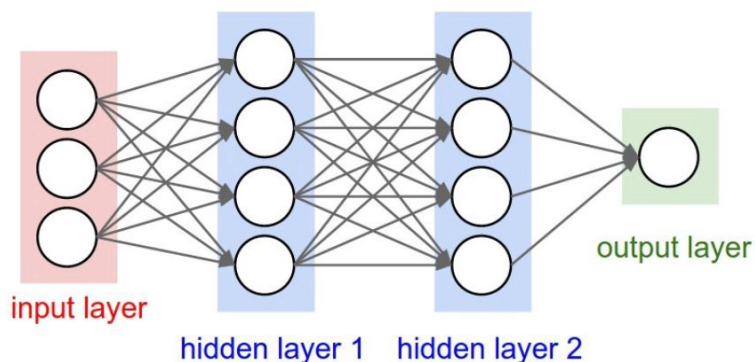
Η λογική του κανόνα μάθησης Perceptron είναι η εξής:

- Εφαρμόζουμε διαδοχικά στην είσοδο του δικτύου τα παραδείγματα που θέλουμε να «μάθει» το σύστημα.
- Για κάθε παράδειγμα, μετράμε την έξοδο του συστήματος, και την συγκρίνουμε με την επιθυμητή έξοδο. Αν η έξοδος είναι μικρότερη από την επιθυμητή, προσπαθούμε να ξεπεράσουμε το κατώφλι, αυξάνοντας τα βάρη που αντιστοιχούν στα σήματα εισόδου με θετική τιμή.
- Ομοίως, κοιτάμε να μειώσουμε αυτά τα βάρη εφόσον το μετρούμενο σήμα στην έξοδο είναι μεγαλύτερο του επιθυμητού

1.4.2 Νευρωνικά δίκτυα πολλών στρωμάτων

Το δίκτυο Perceptron που είδα έχει τον περιορισμό να μπορεί να αναπαραστήσει μόνο επίπεδες επιφάνειες. Αυτός ο αλγόριθμος βασίζεται σε ένα μεμονωμένο γραμμικό νευρώνα με προσαρμόσιμα βάρη, έχοντας ως αποτέλεσμα να περιορίζει την υπολογιστική του ισχύ. Για να ξεπεραστούν αυτοί οι περιορισμοί θα εισάγουμε μια δομή νευρωνικού δικτύου με περισσότερα στρώμα, το δίκτυο Perceptron πολλών επιπέδων [MLP] [50], τα βασικά χαρακτηριστικά του οποίου είναι :

- Το μοντέλο κάθε νευρώνα στο δίκτυο περιλαμβάνει μια μη γραμμική συνάρτηση ενεργοποίησης, η οποία είναι διαφορίσιμη
- Το δίκτυο περιέχει ένα ή περισσότερα επίπεδα τα οποία παραμένουν κρυφά (hidde) για κόμβους των επιπέδων εισόδου και εξόδου
- Το δίκτυο επιδεικνύει μεγάλη διασυνδεσιμότητα, ο βαθμός της οποίας καθορίζεται από τα συναπτικά βάρη του δικτύου.



Σχήμα 1.21. MLP με 2 κρυφά επίπεδα (πλήρως συνδεδεμένο, πρόσθιας τροφοδοσίας) [51]

Η παρουσία μιας κατανεμημένης μορφής μη γραμμικότητας και η υψηλή διασύνδεση του δικτύου καθιστούν την θεωρητική ανάλυση ενός Perceptron πολλών επιπέδων δύσκολη, ενώ η χρήση κρυφών νευρώνων καθιστά δύσκολη την σχηματοποίηση της διαδικασίας της μάθησης. Επομένως η διαδικασία της μάθησης οφείλει να αποφασίσει ποια χαρακτηριστικά του προτύπου εισόδου θα πρέπει να αναπαρίσταται από τους κρυφούς νευρώνες.

Το βασικό χαρακτηριστικό ενός MLP είναι ότι κάθε στρώμα l τροφοδοτείται από τους νευρώνες του προηγούμενου στρώματος $l-1$. Το στρώμα εισόδου αναφέρεται συνήθως ως μηδενικό στρώμα (είσοδος δεδομένων), όλα τα ενδιάμεσα αναφέρονται ως κρυφά επίπεδα-νευρώνες, ενώ η έξοδος αποτελεί το επίπεδο εξόδου. Οι κρυφοί νευρώνες δρουν ως ανιχνευτές χαρακτηριστικών. [50] Καθώς προχωρά η διαδικασία της μάθησης, οι κρυφοί νευρώνες αρχίζουν σταδιακά να 'ανακαλύπτουν' τα εξέχοντα χαρακτηριστικά δηλαδή αυτά που χαρακτηρίζουν τα δεδομένα εκπαίδευσης. Αυτό το κάνουν εκτελώντας ένα μη γραμμικό μετασχηματισμό στα δεδομένα εισόδου σε έναν χώρο που καλείται χώρος χαρακτηριστικών, στον οποίο οι κλάσεις χαρακτηριστικών που ενδιαφέρουν μια εργασία ταξινόμησης προτύπων, μπορούν να ξεχωρίσουν από οτιδήποτε άλλο θα μπορούσε να υπάρχει στον αρχικό χώρο δεδομένων εισόδου. Οπότε τώρα μπορούμε να ορίζουμε τα εξής :

- L είναι το πλήθος των στρωμάτων του δικτύου εκτός από το πρώτο στρώμα που θεωρείται το μηδενικό
- $N(0), \dots, N(L)$ είναι το πλήθος των νευρώνων των στρωμάτων $0, 1, \dots, L$. Συνεπώς $N(0)$ είναι το πλήθος των εισόδων και $N(L)$ είναι το πλήθος των εξόδων.
- $a_i(l)$ είναι οι ενεργοποιήσεις των νευρώνων του στρώματος l
- $w_{ij}(l)$ είναι το συναπτικό βάρος που συνδέει τον νευρώνα $a_j(l-1)$ του $l-1$ στρώματος με τον νευρώνα $a_i(l)$ του l στρώματος
- $a_{i0}(l)$ είναι η πόλωση του νευρώνα $a_i(l)$ του l στρώματος
- $x_i = a_i(0)$ είναι οι εισοδοί του δικτύου
- $y_i = a_i(L)$ είναι οι εξοδοί του δικτύου

Η διαδικασία υπολογισμού των τιμών όλων των νευρώνων με δεδομένες τιμές των εισόδων $x_1 \dots x_{N(0)}$ ονομάζεται ανάκληση [1], και για οποιοδήποτε στρώμα $l=1, \dots, L$ δίνονται από :

$$a_i(l) = f\left(\sum_{j=1}^{N(l-1)} w_{ij}(l)a_j(l-1) + w_{i0}(l)\right)$$

Αυτό αποτελεί και τον τύπο ενεργοποίησης ενός κλασικού νευρώνα. Ο νευρώνας i του στρώματος l δέχεται ως εισόδους τις ενεργοποιήσεις $a_i(l-1)$ των νευρώνων από το στρώμα $l-1$ και ως πόλωση την τιμή $w_{i0}(l)$. Κατά την ανάκληση μας δίνονται οι τιμές x_i των εισόδων του δικτύου, οπότε με βάση τις εισόδους υπολογίζουμε πρώτα τις ενεργοποιήσεις των νευρώνων του στρώματος 1 , στη συνέχεια με βάση αυτές υπολογίζουμε τις ενεργοποιήσεις του στρώματος 2 , και έτσι συνεχίζουμε για τις υπόλοιπες.

Η εκπαίδευση ενός δικτύου πολλών στρωμάτων είναι η διαδικασία καθορισμού των συναπτικών βαρών του έτσι ώστε να ικανοποιείται κάποιο κριτήριο (σκοπός του δικτύου). Έτσι αν έχουμε το κατάλληλο δίκτυο σε μέγεθος μπορούμε να το εκπαιδεύσουμε να μάθει οποιαδήποτε συνάρτηση, με τον αλγόριθμο Back-Propagation (οπισθοδιάδοση) να αποτελεί τον πιο γνωστό αλγόριθμο.[50][1].

Άλλοι μέθοδοι που χρησιμοποιούνται είναι οι :

- Μέθοδος Gradient Descent (Κατιούσας κλίσης)
- Μέθοδος Back propagation (Πίσω διάδοση-οπισθοδρόμησης)
- Στοχαστική βαθμωτή κατάβαση
- Παλινδρόμηση
- Μέθοδος Newton
- Μέθοδος Steepest Descent (Μεγίστης κλίσης)

Ως μέτρο απόδοσης για το σύστημα χρησιμοποιείται συνήθως το μέσο τετραγωνικό σφάλμα ή το άθροισμα των τετραγώνων των σφαλμάτων ορισμένο ως συνάρτηση των συναπτικών βαρών.

Στη μάθηση με εκπαίδευση το νευρωνικό δίκτυο και ο εκπαιδευτής λαμβάνουν ως είσοδο ένα διάνυσμα εκπαίδευσης και ο εκπαιδευτής παρέχει στο νευρωνικό δίκτυο έναν στόχο για το συγκεκριμένο διάνυσμα εκπαίδευσης. Η διαφορά μεταξύ επιθυμητής και πραγματικής εξόδου αποτελεί το σφάλμα. Έτσι οι παράμετροι του δικτύου μεταβάλλονται με επαναληπτικό τρόπο και αποθηκεύονται με τη μορφή σταθερών συναπτικών βαρών τα οποία αποτελούν την μνήμη του δικτύου.

Για εκμάθηση χωρίς επίβλεψη, το δίκτυο αυτοδιοργανώνεται μόνο του, και όχι μέσω ανάδρασης από το περιβάλλον. Αυτή η εσωτερική οργάνωση γίνεται έτσι ώστε σε συγκεκριμένο σύνολο εισόδων να αντιδρά ισχυρά ένας συγκεκριμένος νευρώνας. Απαραίτητη σε αυτή την περίπτωση είναι η ύπαρξη μεγάλου πλήθους δεδομένων. Σε αυτή την περίπτωση δίνεται απλώς η πληροφορία στο δίκτυο και δεν υπάρχει κάποιος εκπαιδευτής που να δίνει την επιθυμητή απόκριση και να επιβλέπει την διαδικασία. Τα δείγματα εκπαίδευσης αποτελούνται μόνο από δείγματα εισόδου και δεν περιέχουν δείγματα επιθυμητής εξόδου. Το δίκτυο χρησιμοποιεί ένα εσωτερικό έλεγχο και εκπαιδεύεται ώστε οι έξοδοι να έχουν τα ίδια χαρακτηριστικά με τις εισόδους. Έτσι το δίκτυο ελέγχει τον εαυτό του και διορθώνει τα σφάλματα στα δεδομένα μέσω ανάδρασης. Όταν το δίκτυο σταματήσει να αλλάζει τις τιμές των βαρών τότε η εκπαίδευση έχει τελειώσει.

Οι μέθοδοι σε αυτή την περίπτωση μπορεί να είναι:

- Ο νόμος του Hebb
- Ο διαφορικός νόμος του Hebb
- Ο ανταγωνιστικός νόμος
- Ο διαφορικός ανταγωνιστικός

Στη μάθηση με ενίσχυση η έξοδος χαρακτηρίζεται ως καλή ή κακή με βάση μια αριθμητική κλίμακα και τα βάρη αναπροσαρμόζονται με βάση αυτό το χαρακτηρισμό. Σε αυτόν τον τύπο μάθησης το δίκτυο τροφοδοτείται με δειγματικά πρότυπα εισόδου αλλά δεν τροφοδοτείται με τις επιθυμητές αποκρίσεις σε αυτές τις εισόδους. Χρησιμοποιείται ένα συνολικό μέτρο - δείκτης συμπεριφοράς το οποίο είναι γνωστό ως ενισχυτικό σήμα και ανατροφοδοτείται στο δίκτυο έτσι ώστε να επιβραβεύσει τις σωστές συμπεριφορές και να τιμωρεί τις λανθασμένες.

1.6 Βαθιά μηχανική μάθηση

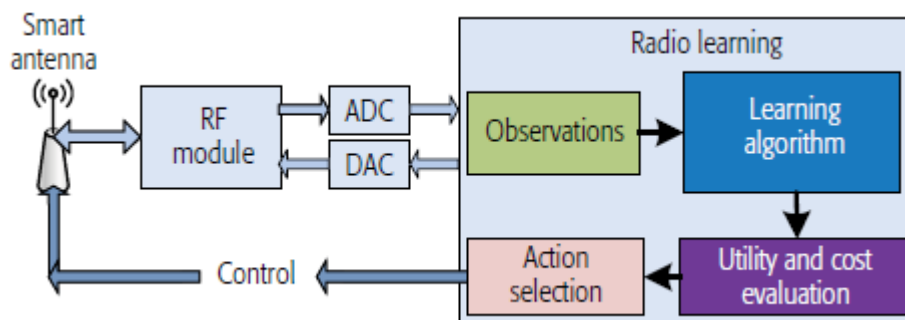
Εφόσον έχει οριστεί η έννοια του τεχνητού νευρωνικού δικτύου, καθίσταται δυνατό να αποτυπωθεί η έννοια της βαθιάς μηχανικής μάθησης. Η βαθιά μηχανική μάθηση είναι η διαδικασία εφαρμογής «μαθησιακών διεργασιών» σε νευρωνικά δίκτυα πολλών επιπέδων. Αποτελεί κομμάτι μιας ευρύτερης οικογένειας μεθόδων μηχανικής μάθησης βασιζόμενων στην αναπαράσταση δεδομένων, σε αντίθεση με αλγορίθμους επικεντρωμένους σε υπολογιστικές εργασίες.

Κεφάλαιο 2

Μηχανική μάθηση σε ασύρματα δίκτυα

Εισαγωγή 2.1

Τα ασύρματα δίκτυα επόμενης γενιάς αναμένεται να υποστηρίξουν εξαιρετικά υψηλές ταχύτητες δεδομένων καθώς και νέες εφαρμογές, οι οποίες απαιτούν ένα νέο πρότυπο ασύρματης τεχνολογίας. Η πρόκληση είναι να βοηθήσουμε την ασύρματη τεχνολογία στην έξυπνη προσαρμοστική μάθηση και τη λήψη αποφάσεων, ώστε να ικανοποιηθούν οι διαφορετικές απαιτήσεις των ασύρματων δικτύων επόμενης γενιάς. Τα μελλοντικά έξυπνα κινητά τερματικά αναμένεται να έχουν αυτόνομη πρόσβαση στις πιο αξιόλογες φασματικές ζώνες με τη βοήθεια εκμάθησης για την καλύτερη φασματικής απόδοσης, προκειμένου να ελέγχουν την ισχύ μετάδοσης, ενώ θα βασίζονται στην εκμάθηση της ενεργειακής τους απόδοσης και ταυτόχρονα θα προσαρμόζουν τα πρωτόκολλα μετάδοσης για τη απόκτηση της μέγιστης ποιότητας των υπηρεσιών. Μια από τις μεγαλύτερες προκλήσεις των δικτύων επόμενης γενιάς είναι η παροχή πρωτοφανών ρυθμών δεδομένων μεταδοσεις ώστε να μπορέσει να υποστηρίξει σύγχρονες εφαρμογές και τεχνολογίες. Πιο συγκεκριμένα, τα σύγχρονα δίκτυα καλούνται να μάθουν τόσο το περιβάλλον χρήστη όσο και την συμπεριφορά αλλά και τις απαιτήσεις του περιβάλλοντος προκειμένου να διαμορφώσουν το σύστημα στο μέγιστο. Αυτές οι έξυπνες συσκευές λοιπόν πρέπει να βασίζονται σε ένα πιο έξυπνο τρόπο λήψης αποφάσεων. Έτσι λοιπόν ο τομέας της μηχανικής μάθησης καλείται να παίξει καθοριστικό ρόλο στην εύρεση έξυπνων αλλά και βέλτιστων λύσεων σε συνδυασμό με τις σύγχρονες τεχνολογίες (π.χ. MIMO, cloud computing) ώστε να δημιουργηθεί ένα σύγχρονο και ευέλικτο σύστημα τηλεπικοινωνιών. Στο επόμενη σχήμα παρουσιάζεται ένα ραδιοσύστημα με χρήση μηχανικής μάθησης που θα είναι σε θέση να έχει αυτόνομη προσαρμογή και πρόσβαση στο διαθέσιμο φάσμα, έλεγχο ισχύος για εξοικονόμηση ενέργειας προσαρμογή του πρωτοκόλλου μετάδοσης, καθώς και ότι άλλο απαιτηθεί.



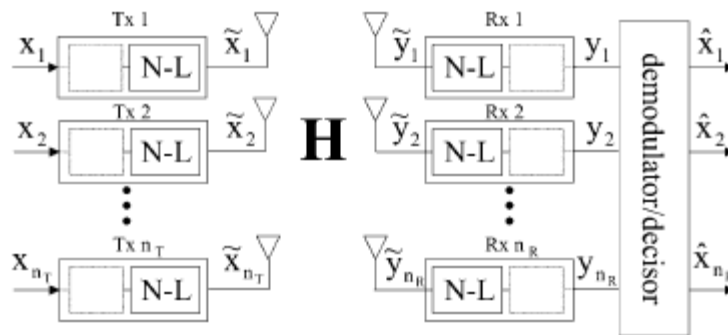
Σχήμα 2.1. Παράδειγμα έξυπνου ραδιοσυστήματος [55]

2.2 Εκτίμηση καναλιού MIMO με SVM

Σε αυτή την ενότητα παρουσιάζεται μια τεχνική για την εκτίμηση καναλιού με επίπεδες διαλείψεις για ένα σύστημα MIMO, κάνοντας χρήση παλινδρόμησης πολλαπλών μεταβλητών (multiregression) βασιζόμενη στις μηχανές διανυσματικής υποστήριξης (Support Vector Machines-SVM) που θα την αναφέρουμε ως M-SVM, ενώ θα συγκριθούν με άλλες τεχνικές. Οι μηχανές διανυσματικής υποστήριξης χρησιμοποιούνται τόσο για δεδομένα με γραμμική σχέση εισόδου-εξόδου όσο και για δεδομένα με μη γραμμική σχέση, τόσο για προβλήματα ταξινόμησης αλλά και παλινδρόμησης.

Οι πιο συνηθισμένες τεχνικές και έρευνες για την εκτίμηση καναλιού MIMO ήταν κυρίως για επίπεδες διαλείψεις και βασιζόνταν είτε σε εκτιμητή καναλιού maximum likelihood (ML) είτε στο ελάχιστο τετραγωνικό σφάλμα (minimum mean square error -MMSE). Στην ενότητα αυτή θα εκμεταλλευτούμε την πολυδιάστατη φύση ενός καναλιού MIMO ώστε να εκτιμήσουμε το κανάλι κάνοντας χρήση παλινδρόμησης με μηχανές διανυσματικής υποστήριξης, ενώ για απλότητα έχουμε υποθέσει ότι: δεν υπάρχει διασυμβολική παρεμβολή στο κανάλι, καθώς και ότι οι διαλείψεις είναι αρκετά αργές ώστε το κανάλι να παραμένει σταθερό κατά την διάρκεια του διαστήματος εκτίμησης ενώ οι μη γραμμικότητες καναλιών θα μπορούσαν να ληφθούν υπόψη είτε κατά την μετάδοση είτε κατά τη λήψη.

Το μοντέλο καναλιού που θα εστιάζουμε βασίζεται σε ένα γραμμικό σύστημα MIMO με n_T κεραιές μετάδοσης και n_R κεραιές λήψης. Χρησιμοποιούμε ένα πίνακα \mathbf{H} , ανεξάρτητων συντελεστών Gaussian για να μοντελοποιήσουμε ένα κανάλι Rayleigh (επίπεδων διαλείψεων), μεγέθους $n_R \times n_T$ όπου το στοιχείο h_{ij} αντιστοιχεί στην εξασθένιση μεταξύ της j εκπεμπόμενης και της i λαμβανόμενης κεραιάς (σχήμα 2.1). Η γενική υπόθεση για συντελεστές καναλιού είναι ένα σύνολο ανεξάρτητων και πανομοιότυπων κατανομημένων μεταβλητών (independent and identically distributed-i.i.d). Ο λευκός Gaussian θόρυβος που υιοθετείται στο κανάλι μοντελοποιείται μέσω ενός διανύσματος n διάστασης n_R .



Σύστημα 2.2. MIMO με n_T πομπούς και n_R δέκτες.[57]

Στο μοντέλο διάδοσης, οι μη γραμμικότητες θα θεωρούνται πανομοιότυπες είτε σε κάθε πομπό είτε σε κάθε δέκτη από το σύστημα MIMO. Έτσι, και οι δύο περιπτώσεις θα αντιμετωπίζονται ξεχωριστά, οδηγώντας σε δύο διαφορετικά μοντέλα καναλιών.

Υποθέτοντας ότι $\mathbf{x} = [\chi_1, \dots, \chi_{n_T}]^T$ είναι το σήμα πληροφορίας σε κάθε δείγμα χρόνου με διαμόρφωση QPSK, οι μη γραμμικές στη μετάδοση οδηγούν σε εξίσωση όπου τα μεταδιδόμενα σύμβολα ακολουθούν τους μη γραμμικούς κανόνες : [58]

$$\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \vdots \\ \tilde{x}_{n_T} \end{bmatrix} = \begin{bmatrix} x_1 + \alpha_1 x_1^2 + \alpha_2 x_1^3 \\ x_2 + \alpha_1 x_2^2 + \alpha_2 x_2^3 \\ \vdots \\ x_{n_T} + \alpha_1 x_{n_T}^2 + \alpha_2 x_{n_T}^3 \end{bmatrix}$$

Τα λαμβανόμενα σύμβολα $\tilde{\mathbf{y}}$ είναι γραμμικά συνδυασμένα με τα μεταδιδόμενα σύμβολα μέσω της διάδοσης καναλιού:

$$\tilde{\mathbf{y}} = \frac{1}{n_T} \mathbf{H} \tilde{\mathbf{x}} + \mathbf{n}.$$

Σε αυτό το μοντέλο καναλιού με μη γραμμικότητες στη μετάδοση, η διαδικασία λήψης

θεωρείται γραμμική $\mathbf{y} = \tilde{\mathbf{y}}$

Το δεύτερο κανάλι, με μη γραμμικότητες στη λήψη, οδηγεί σε ένα διαφορετικό μοντέλο καναλιού. Τα σύμβολα $\tilde{\mathbf{x}} = \mathbf{x}$ πληροφοριών που μεταδίδονται συνδυάζονται γραμμικά μέσω της εξίσωσης:

$$\tilde{\mathbf{y}} = \frac{1}{n_T} \mathbf{H} \mathbf{x} + \mathbf{n}.$$

Σε αυτό το μοντέλο καναλιού, οι μη γραμμικότητες τροποποιούν τα λαμβανόμενα σύμβολα ως εξής:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n_R} \end{bmatrix} = \begin{bmatrix} \tilde{y}_1 + \alpha_1 \tilde{y}_1^2 + \alpha_2 \tilde{y}_1^3 \\ \tilde{y}_2 + \alpha_1 \tilde{y}_2^2 + \alpha_2 \tilde{y}_2^3 \\ \vdots \\ \tilde{y}_{n_R} + \alpha_1 \tilde{y}_{n_R}^2 + \alpha_2 \tilde{y}_{n_R}^3 \end{bmatrix}$$

Τα προτεινόμενα μοντέλα δύο καναλιών αντιμετωπίζουν δύο διαφορετικά προβλήματα.

Στο πρώτο μοντέλο καναλιού, στο άκρο του δέκτη, υπάρχει ένας γραμμικός συνδυασμός των μεταδιδόμενων συμβόλων, παρόλο που καθένα από αυτά είναι μια μη γραμμική συνάρτηση των συμβόλων πληροφοριών, ο θόρυβος που θεωρείται εδώ είναι Gaussian και λευκός. Από την άλλη πλευρά, το δεύτερο μοντέλο καναλιού οδηγεί σε μη γραμμικό συνδυασμό όλων των μεταδιδόμενων συμβόλων στο δέκτη και ο θόρυβος δεν μπορεί πλέον να θεωρηθεί Gaussian.

Η εκτίμηση καναλιού για συστήματα MIMO, βασίζεται σε πιλοτικά φέροντα που αποτελούν μέρος των ωφέλιμων δεδομένων, και μπορεί να αντιμετωπιστούν με παλινδρόμηση δεδομένων. Σε ένα πρόβλημα μηχανικής μάθησης με παλινδρόμηση ψάχνουμε να βρούμε την άγνωστη σχέση μεταξύ των δεδομένων εισόδου και των δεδομένων εξόδου ($y=f(x)$). Έτσι σε ένα σύστημα MIMO με n_T κεραιές εκπομπής και n_R κεραιές λήψης, το κανάλι μπορεί να εκτιμηθεί από την παλινδρόμηση των n_R λαμβανόμενων συμβόλων σε κάθε μεταδιδόμενη λέξη στην αντίστοιχη πιλοτική ακολουθία.

Μια πιο συνηθισμένη μέθοδος για την εκτίμηση του καναλιού χρησιμοποιεί έναν γραμμικό συνδυασμό των στοιχείων εισόδου x επιλέγοντας το βάρος τους προκειμένου να ελαχιστοποιηθεί το τετραγωνικό σφάλμα E , (MMSE):

$$E = \sum_{i=1}^n \left(\hat{f}(\mathbf{x}_i) - y_i \right)^2$$

Παρόλο που η μέθοδος MMSE είναι η βέλτιστη (και αρκετά αποτελεσματική αν χρησιμοποιούνται σύντομες πιλοτικές ακολουθίες) για την εκτίμηση γραμμικών καναλιών με πρόσθετο θόρυβο Gauss, η απόδοσή της μπορεί να είναι πολύ κακή.

Αυτό έχει ως αποτέλεσμα να χρησιμοποιείται μια προσαρμοστική έκδοση ενός δικτύου ακτινικής βάσης (radial basis function network-RBFN)[59][62] για την εκτίμηση του καναλιού σε συστήματα OFDM. Ο RBFN εφαρμόζει μια συνάρτηση της μορφής

$$\hat{f}(\mathbf{x}) = y = \sum_{m=1}^M w_m h_m (\|\mathbf{x} - \mathbf{c}_m\|_2)$$

όπου M είναι ο αριθμός των κόμβων στο δίκτυο, $h_m(x)$ είναι η συνάρτηση ακτινικής βάσης (οι τιμές τους εξαρτώνται μόνο από την απόσταση μεταξύ του x και ενός σταθερού σημείου (κέντρου) ή ενός προτύπου c_m)[60][61], και w_m είναι τα βάρη που αντιστοιχούν σε καθεμία από αυτές τις συναρτήσεις. Διαφορετικές μορφές της συνάρτησης ακτινικής βάσης μπορούν να χρησιμοποιηθούν όπως η Gaussian:

$$h_m(x) = \exp\left(-\frac{x^2}{2\sigma_m^2}\right)$$

Αφού επιλεγεί το M οι παράμετροι c_m , σ_m και w_m είναι ελεύθερες και πρέπει να υπολογιστούν από τα δεδομένα εκπαίδευσης. Αυτή η φάση συνήθως περιλαμβάνει την ελαχιστοποίηση του τετραγωνικού σφάλματος. Ένας στοχαστικός αλγόριθμος [62] χρησιμοποιείται για να παρέχει στο δίκτυο δυνατότητες προσαρμογής ενώ συμπεριλάβετε έναν όρο κανονικοποίησης της συνάρτησης κόστους:

$$E = \sum_{i=1}^n \left(\hat{f}(\mathbf{x}_i) - y_i \right)^2 + \frac{\lambda}{\omega} (\hat{f}).$$

όπου ο όρος $\Omega(\hat{f})$ είναι ένας κανονικοποιητής που “τιμωρεί” τις μεγάλες αξίες της \hat{f} ενώ η παράμετρος λ ελέγχει την ανταλλαγή μεταξύ σφάλματος και ομαλότητας της λύσης.

Σε ένα σύστημα MIMO η εκτίμηση καναλιού μπορεί να γίνει με ένα σύνολο n_R μονοδιάστατου παλινδρομητών. Στη συνέχεια θα εξετάσουμε ένα σύστημα στο οποίο κάθε ένας από τους

παλινδρομητές υλοποιείται χρησιμοποιώντας SVR. Αντί να δημιουργήσουμε έναν διαφορετικό παλινδρόμηση για κάθε μεταβλητή στον δέκτη, τους θεωρούμε όλους ταυτόχρονα, που οδηγεί σε απλούστερες λύσεις (η πολυπλοκότητα του οποίου δεν αυξάνεται με τον αριθμό λήψης των κεραιών) διατηρώντας παράλληλα την υψηλή απόδοση των μεθόδων SVM.

Θα εισάγουμε μια γενίκευση του SVR για παλινδρόμηση πολλαπλών μεταβλητών, θα το αναφέρουμε ως M-SVR, το οποίο θα ανακαλύπτει τις εξαρτήσεις μεταξύ μεταδιδόμενων και ληφθέντων σημάτων σε ένα σύστημα MIMO. Ο συνδυασμός όλων των διαδρομών του καναλιού θα επιτρέψει την ακριβέστερη εκτίμηση όταν έχουμε στην διάθεση μας λιγοστά δεδομένα και η συνάρτηση κόστους, ϵ , θα βελτιώσει την ευρωστία του σχήματος όταν εμφανίζονται διαφορετικά είδη θορύβου και μη γραμμικότητες στο σύστημα. Το κλασικό SVR λύνει το πρόβλημα βρίσκοντας το παλίνδρομο w και b το οποίο ελαχιστοποιεί την εξίσωση

$$\|w\|^2/2 + C \sum_{i=1}^n L_v(y_i - (\phi^T(x_i)w + b))$$

όπου $\Phi()$ είναι ένας μη γραμμικός μετασχηματισμός (συνάρτηση πυρήνα) σε ένα χώρο υψηλών διαστάσεων (χώρος χαρακτηριστικών), $\Phi(x) \in R^H$ και $H > d$. Το SVR μπορεί να λυθεί χρησιμοποιώντας το εσωτερικό γινόμενο (inner product) ανάμεσα στο $\Phi()$, όπου δεν χρειάζεται να γνωρίζουμε τη μη γραμμική απεικόνιση επομένως, πρέπει να καθορίσουμε μόνο μια συνάρτηση πυρήνα $\kappa(X_i, X_j) = \Phi^T(X_i)\Phi(X_j)$ [1]. Η $L_u()$ ονομάζεται Vapnik συνάρτηση κόστους (Vapnik-insensitive loss-function), υποδεικνύει την αναγκαιότητα περιορισμού των κλάσεων των συναρτήσεων στις οποίες μπορεί να ανήκει η μηχανή μάθησης) η οποία είναι ίση με το μηδέν για $|y_i - (\Phi^T(X_i)w + b)| < 0$ και ίση με $|y_i - (\Phi^T(X_i)w + b)| - \epsilon$ για $|y_i - (\Phi^T(X_i)w + b)| \geq \epsilon$. Η λύση για τα w και b σχηματίζεται από έναν γραμμικό συνδυασμό των εκπαιδευτικών δειγμάτων στον χώρο που παρουσιάζει απόλυτο σφάλμα ίσο ή μεγαλύτερο από ϵ . Στην περίπτωση που η έξοδος είναι ένας διάνυσμα $y \in R^Q$, πρέπει να λύσουμε ένα πολυδιάστατο πρόβλημα παλινδρόμησης στο οποίο πρέπει να βρούμε έναν παλίνδρομο w^j και b^j , ($j = 1, \dots, Q$) για κάθε έξοδο. Μπορούμε να γενικεύσουμε το μονοδιάστατο SVR για να λύσουμε την πολυδιάστατη περίπτωση, οδηγώντας στην ελαχιστοποίηση της εξίσωσης:

$$L_P(W, b) = \frac{1}{2} \sum_{j=1}^Q \|w^j\|^2 + C \sum_{i=1}^n L(u_i)$$

$u_i = \|e_i\| = \sqrt{e_i^T e_i}$, $e_i^T = y_i^T - \Phi^T(x_i)W - b^T$, $W = [w^1, \dots, w^Q]$, $b = [b^1, \dots, b^Q]^T$ και $\Phi()$ είναι ένας μη γραμμικός μετασχηματισμός στο χώρο των χαρακτηριστικών.

Η συνάρτηση ϵ -Vapnik μπορεί να επεκταθεί σε πολλαπλές διαστάσεις, και κάνοντας χρήση της ευκλείδεια νόρμας όλες οι διαστάσεις μπορούν να ληφθούν υπόψη με ένα μοναδικό περιορισμό, αποδίδοντας έτσι ένα διάνυσμα υποστήριξης για όλες τις διαστάσεις, και μπορούμε να κάνουμε χρήση της

$$L(u) = \begin{cases} 0, & u < \varepsilon \\ u^2 - 2u\varepsilon + \varepsilon^2, & u \geq \varepsilon \end{cases}$$

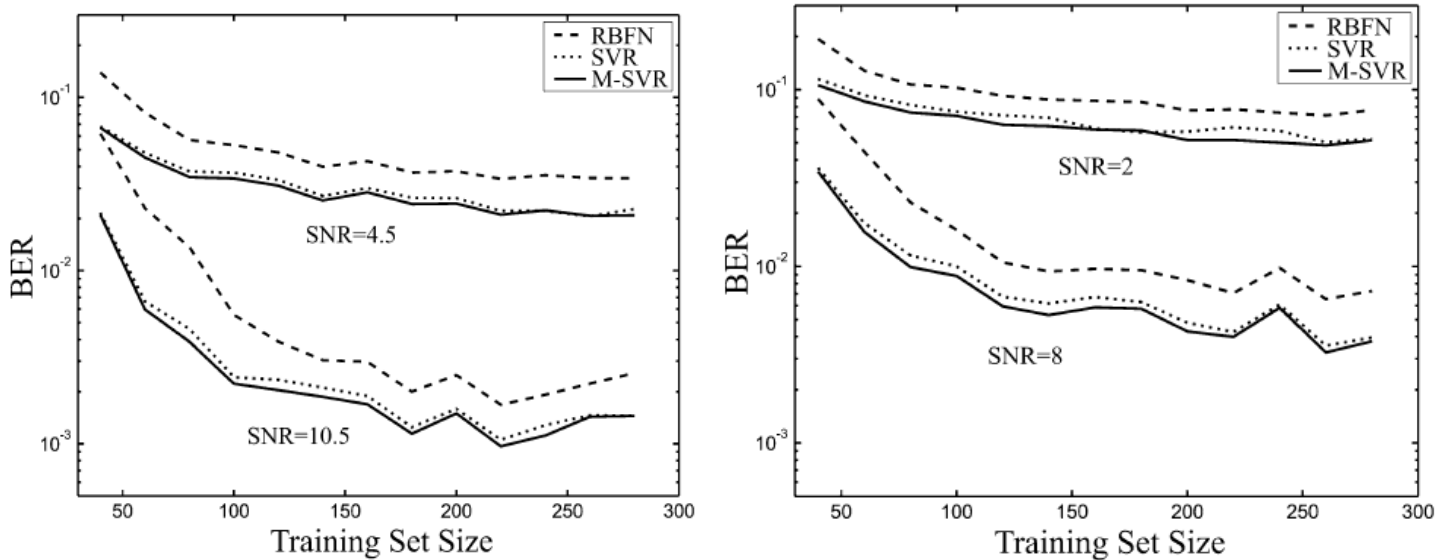
Για $\varepsilon=0$, το πρόβλημα γίνεται έναν ανεξάρτητο πρόβλημα τετραγωνικής παλινδρόμησης για κάθε στοιχείο, αλλά για ένα μη μηδενικό ε , η λύση θα λάβει υπόψη όλες τις εξόδους για την κατασκευή κάθε μεμονωμένου παλίνδρομου και θα είναι σε θέση να κάνει πιο ισχυρές προβλέψεις. Το τίμημα που πρέπει να καταβληθεί είναι ότι η επίλυση του προτεινόμενου προβλήματος δεν μπορεί να γίνει άμεσα και θα πρέπει να βασιστούμε σε μια επαναληπτική διαδικασία για να επιτύχουμε την επιθυμητή λύση. Έτσι επινοήθηκε μια προσέγγιση quasi-Newton[63] στην οποία κάθε επανάληψη έχει την ίδια υπολογιστική πολυπλοκότητα όπως τα ελάχιστα τετράγωνα για κάθε στοιχείο. Είναι ένα πρόβλημα σταθμισμένων ελαχίστων τετραγώνων όπου ο αριθμός των επαναλήψεων είναι μικρότερος καθιστώντας την διαδικασία λίγο πιο απαιτητική από αυτή των ελαχίστων τετραγώνων για κάθε στοιχείο. Η διαδικασία αυτή αναφέρεται στην βιβλιογραφία ως IRWLS.[64]

➤ Πειραματικά αποτελέσματα

Ο στόχος στο πρόβλημα εκτίμησης καναλιού είναι να επιτευχθεί μια καλή προσέγγιση με το πραγματικό κανάλι, μοντελοποιώντας την εξάρτηση μεταξύ των μεταδιδόμενων και των ληφθέντων σημάτων. Με τη μέθοδο MMSE, αυτή η σχέση περιορίζεται να είναι γραμμική και η εκτίμηση καναλιού \hat{H} μπορεί να δοθεί ρητά. Αυτό ισχύει για τις μεθόδους M-SVR και SVR με γραμμικό πυρήνα, αλλά δεν είναι δυνατό όταν χρησιμοποιείτε μη γραμμικός μετασχηματισμός σε χώρο υψηλότερης διάστασης (δηλαδή, όταν εφαρμόζεται άλλο πυρήνα διαφορετικό από τον γραμμικό) επειδή σε αυτήν την περίπτωση, μπορούμε να υπολογίσουμε μόνο τους πυρήνες. Η μέθοδος RBFN λειτουργεί ανάλογα με τους πυρήνες παλινδρόμησης.

Σε ένα σύστημα MIMO 4×3 ($n_t = 4$, $n_r = 3$) και με το κανάλι να έχει τα χαρακτηριστικά όπως περιγράφηκαν προηγουμένως, θα παρουσιαστούν αποτελέσματα για μη γραμμικότητα στο πομπό και στον δέκτη.

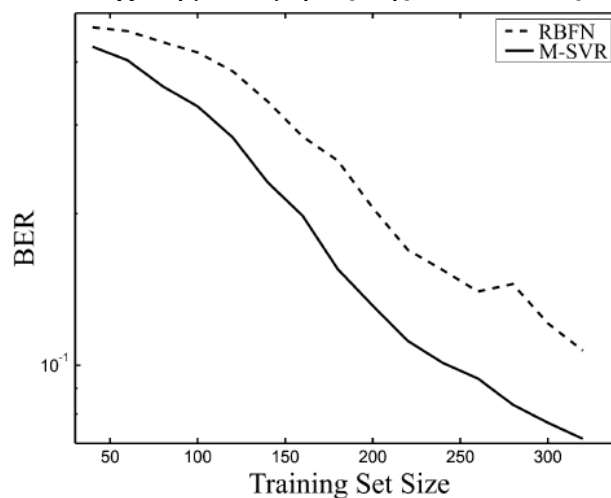
Η τεχνική M-SVR έχει καλύτερα αποτελεσματικά (μη γραμμικότητα στο πομπό), όπως φαίνεται στο επόμενο σχήμα, και λαμβάνει χαμηλότερο BER από το RBFN για μεταβλητό SNR.



Σχήμα 2.3 : M-SVR , SVR , RBFN και BER ως συνάρτηση της ακολουθίας εκπαίδευσης [57]

Η βελτίωση αυτής της μεθόδου είναι ιδιαίτερα αντιπροσωπευτική για μικρά σετ εκπαίδευσης, αν και η διαφορά μειώνεται μόνο ελαφρώς για τα μεγαλύτερα σετ. Το σημείο κορεσμού των καμπυλών, για το οποίο το BER δεν βελτιώνεται πλέον, αυξάνεται καθώς αυξάνεται το SNR. Παρά το γεγονός ότι η απόδοση του αλγορίθμου SVR είναι παρόμοια με τη μέθοδο M-SVR, η πολυπλοκότητα είναι μεγαλύτερη. Ενώ το M-SVR απαιτεί λίγες επαναλήψεις του IRWLS για σύγκλιση (περίπου πέντε βήματα), το SVR χρειάζεται περίπου δύο τάξεις μεγέθους περισσότερες επαναλήψεις. Εκτός αυτού, η πολυπλοκότητα του SVR αυξάνεται τόσο με το n_R όσο και με το μήκος της ακολουθίας εκπαίδευσης, ενώ αυτή του M-SVR δεν εξαρτάται από το n_R .

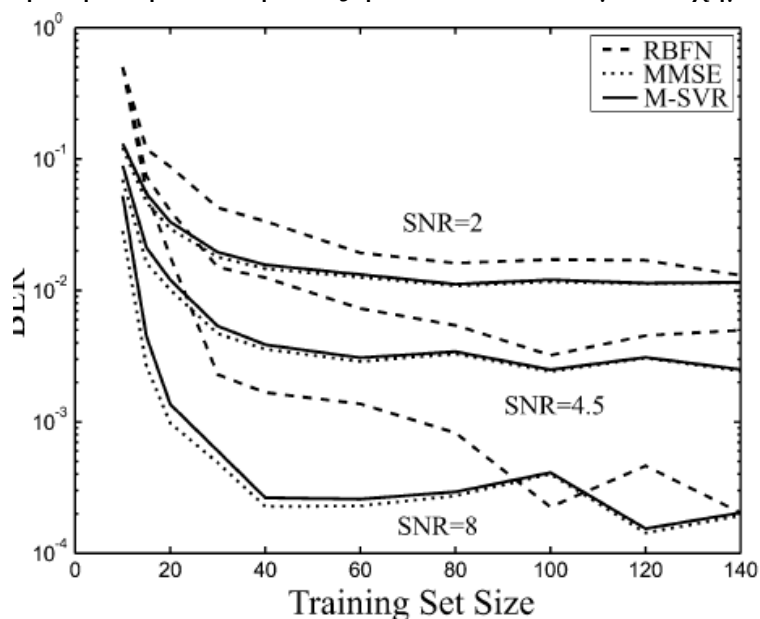
Η μη γραμμικότητα στο δέκτη είναι πιο δύσκολη να αντιμετωπιστεί. Στο επόμενο σχήμα φαίνονται οι καμπύλες BER σε σχέση με το μήκος της ακολουθίας.



Σχήμα 2.4 : M-SVR , RBFN και BER ως συνάρτηση της ακολουθίας εκπαίδευσης για SNR 10.5dB [57]

Για αυτό το μη γραμμικό κανάλι, η εκτίμηση που δίνεται από το RBFN είναι σαφώς και σταθερά χειρότερη από τη λύση M-SVR, η οποία φαίνεται να διαμορφώνει το πρόβλημα με μεγαλύτερη ακρίβεια.

Τέλος, παρουσιάζονται αποτελέσματα από πειράματα για ένα γραμμικό κανάλι με πρόσθετο θόρυβο Gauss και βλέπουμε την απόδοση του M-SVR σε σύγκριση με το MMSE, το οποίο είναι είναι βέλτιστο σε αυτήν την περίπτωση όπως φαίνεται στο επόμενο σχήμα.



Σχήμα 2.5: M-SVR, MMSE και RBFN BER σε ένα γραμμικό κανάλι MIMO για SNR 2-8 [57]

Τα αποτελέσματα και για τις δύο μεθόδους είναι σχεδόν παρόμοια, ενώ το μικρό πλεονέκτημα του MMSE οφείλεται στο γεγονός ότι το M-SVR δεν κάνει εκ των προτέρων παραδοχή σχετικά με τη γραμμικότητα του καναλιού. Εάν χρησιμοποιήσαμε αυτές τις πληροφορίες, θα μπορούσαμε να χρησιμοποιήσουμε έναν γραμμικό πυρήνα.

➤ Συμπεράσματα

Σε αυτήν την ενότητα, είδαμε την εκτίμηση καναλιών για συστήματα MIMO με έναν αλγόριθμο παλινδρόμησης που βασίζεται σε SVM. Ο αλγόριθμος M-SVR εκμεταλλεύεται το spatial diversity και είναι σε θέση να ανακαλύψει τις εξαρτήσεις μεταξύ των μεταδιδόμενων και ληφθέντων σημάτων. Το M-SVR μπορεί να χρησιμοποιηθεί με μη γραμμικούς πυρήνες, προκειμένου να αντιμετωπιστεί αποτελεσματικά η μη γραμμική εκτίμηση καναλιών. Ο M-SVR απαιτεί υπολογιστικό φορτίο που είναι συγκρίσιμο με αυτό άλλων γνωστών μεθόδων όπως ο εκτιμητής MMSE. Επιπλέον, η ανάλυση M-SVR βρίσκεται στον αλγόριθμο IRWLS, ο οποίος μπορεί εύκολα να τροποποιηθεί ώστε να χρησιμοποιεί διαφορετικές λειτουργίες κόστους ή να του παρέχει προσαρμοστικές ιδιότητες. Έγινε σύγκριση με το βέλτιστο αλγόριθμο MMSE αποδίδοντας παρόμοια αποτελέσματα. Για πιο ρεαλιστικά αποτελέσματα θα πρέπει να λάβουμε υπόψη το ISI διαφορετικά είδη θορύβου και είδη πυρήνα κατάλληλα για συστήματα MIMO, ενώ αυτό το σχήμα ίσως μπορέσει να χρησιμοποιηθεί και σε άλλα συστήματα.

2.3 Προσδιορισμός παραμέτρων ασύρματων δικτύων μέσω H-SVM

Σε αυτή την ενότητα θα αναφερθούμε στην εκτίμηση παραμέτρων για δύο προβλήματα ασύρματων δικτύων. Πιο συγκεκριμένα θα θεωρήσουμε α) το πρόβλημα εκτίμησης της θέσης των φυσικών κόμβων ενός ασύρματου εσωτερικού δικτύου και β) την εκτίμηση θορύβου του καναλιού ενός συστήματος MIMO. Η γνώση των παραμέτρων αυτών λαμβάνει μέρος σε πολλές εργασίες του δικτύου όπως η διαχείριση δικτύου, η ανίχνευση συμβάντων καθώς και υπηρεσίες βάση της τοποθεσίας, και δρομολόγησης. Όπως έχει ήδη αναφερθεί η τεχνική SVM είναι για δυαδική ταξινόμηση, έχοντας ως αποτέλεσμα διαφορετικές τεχνικές να χρειάζονται για περισσότερες κλάσεις. Η τεχνική που θα ακολουθήσουμε βασίζεται σε ένα ιεραρχικό σύστημα (δέντρο)[163] μηχανών διανυσματικής υποστήριξης - Hierarchical Support Vector Machines, H-SVM, καθώς παρουσιάζει τα ακόλουθα πλεονεκτήματα: [62]

1. Προσφέρει μια αποτελεσματική διαδικασία αξιολόγησης με καταναμημένο τρόπο λόγω της ιεραρχικής δομής
2. Μπορεί να προσδιορίσει παραμέτρους δικτύου βασιζόμενο σε απλούστερες πληροφορίες του δικτύου όπως, αριθμός κόμβων, χωρίς να απαιτείται εξειδικευμένο υλικό.
3. Προκύπτει ακριβέστερος μέσος όρος και διακύμανση του σφάλματος εκτίμησης.
4. Με χρήση ενός παράλληλου αλγορίθμου μάθησης και την τεχνική γρήγορης διαγωνοποίηση πίνακα, μειώνεται ο χρόνος υπολογισμού από την παραδοσιακή πολυπλοκότητα μάθησης ενός SVM από $O(n^3)$ σε $O(n^2)$, όπου n το μέγεθος του δείγματος εκπαίδευσης.

Η ανάπτυξη και η εκτεταμένη χρήση συσκευών εσωτερικών ασύρματων δικτύων (π.χ. Laptop, smartphones, IoT-Internet of Things-IoT) έχει δημιουργήσει ενδιαφέρον για μεθόδους αποτελεσματικής παρακολούθησης τόσο των ανθρώπων όσο και των πραγμάτων. Η γνώση των γεωγραφικών τοποθεσιών τέτοιων συσκευών σε εσωτερικά ασύρματα δίκτυα είναι κρίσιμη για πολλές εργασίες ενός εσωτερικού ασύρματου δικτύου (π.χ. ανίχνευση συμβάντων), ενώ το σύστημα GPS σε πολλές περιπτώσεις δεν λειτουργεί για συσκευές εντός κτιρίων. Επομένως, ένα σημαντικό πρόβλημα είναι να προταθεί μια μέθοδος για την αποτελεσματική εκτίμηση των τοποθεσιών των συσκευών, δεδομένου ότι οι πραγματικές πληροφορίες θέσης που λαμβάνονται από εξοπλισμό τύπου GPS είναι ελάχιστες ή άγνωστες.

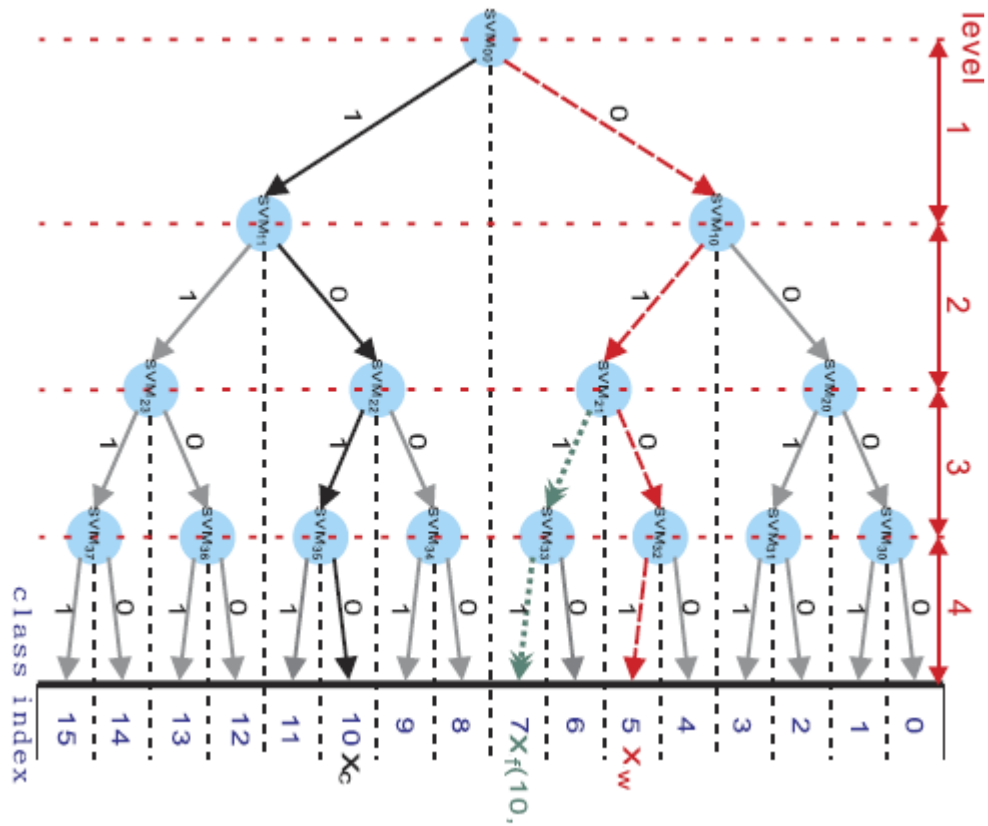
Η εξασθένιση του σήματος καθώς και πολυδιαδρομική διάδοση είναι ένα κεντρικό πρόβλημα στις ασύρματες επικοινωνίες επειδή προκαλεί στο σήμα ταχείς και βαθιές διαλείψεις αλλά και εξάπλωση της συχνότητας. Ο ακριβής προσδιορισμός του θορύβου του καναλιού μπορεί να βοηθήσει στην υιοθέτηση προσαρμοστικών αλγορίθμων αποδιαμόρφωσης ώστε να βελτιωθεί η απόδοση του καναλιού αλλά και να παρέχει πληροφορίες σχετικά με την ποιότητα του καναλιού για handoff, έλεγχο ισχύος και εκχώρηση καναλιών. Προηγούμενες τεχνικές για την εκτίμηση καναλιού ήταν κυρίως για επίπεδες διαλείψεις και βασιζόνταν είτε σε εκτίμηση καναλιού με maximum likelihood (ML) είτε στο ελάχιστο τετραγωνικό σφάλμα (minimum mean square error -MMSE).

- Δομή H-SVM

Η τεχνική SVM είναι ένας ισχυρός μηχανισμός ταξινόμησης. Προκειμένου να γενικεύσουμε τη λειτουργία SVM για προβλήματα ταξινόμησης δεδομένων πολλαπλών κλάσεων, πρέπει να υιοθετήσουμε περισσότερους από έναν ταξινομητές SVM σε μια πολυεπίπεδη δομή. Η τεχνική H-SVM για ταξινόμηση δεδομένων πολλαπλών κλάσεων, μπορεί να εφαρμοστεί. Η δομή αυτή αποτελείται από διαφορετικά επίπεδα, και κάθε επίπεδο αποτελείται από έναν πεπερασμένο αριθμό ταξινομητών SVM. Δεδομένου ενός θετικού ακεραίου m , η λεπτομερής δομή για το H-SVM με επίπεδα m περιγράφεται ως εξής:[62]

- H-SVM με m επίπεδα έχει $2^m - 1$ SVM ταξινομητές και είναι διατεταγμένοι ως κόμβοι στους κλάδους ενός δυαδικού δέντρου με ύψος m .
- Στο i -οστό επίπεδο, $1 \leq i \leq m$, υπάρχουν $2^i - 1$ SVM ταξινομητές για να ξεχωρίσουν τα δεδομένα εισόδου σε 2^i κλάσεις με επίπεδο i .
- Το σύμβολο SVM_{ij} αντιπροσωπεύει τον j -οστό ταξινομητή στο επίπεδο i , όπου $1 \leq i \leq m$ και $0 \leq j \leq 2^{i-1} - 1$. Η αντίστοιχη συνάρτηση υπερ-επιπέδου απόφασης για SVM_{ij} είναι $D_{i,j}(x)$ όπου $D_{i,j}(x)$ [63] είναι μια συνάρτηση απόφασης SVM (μη γραμμικού ταξινομητή). Κάθε $SVM_{i,j}$, θα ταξινομήσει τα δεδομένα από το επίπεδο i σε δύο κλάσεις με επίπεδο $i+1$. Ακόμα σε μια κλάση με επίπεδο $i+1$ από αυτές τις δύο κλάσεις θα εκχωρηθεί bit 0 και στην άλλη κλάση θα εκχωρηθεί bit 1.
- Η τεχνική H-SVM πρέπει να εκτελέσει ταξινόμηση SVM m φορές για να προσδιορίζει την κλάση με το επίπεδο m στην οποία ανήκουν τα δεδομένα εισόδου. Ας υποθέσουμε ένα διάνυσμα d , m στοιχείων, το στοιχείο i του d συμβολίζεται ως d_i , που αποθηκεύει την δυαδική τιμή του bit (0 ή 1) της κλάσης με επίπεδο i , όπου η δυαδική τιμή του bit βασίζεται στην έξοδο του ταξινομητή SVM στο $i-1$ επίπεδο. Τέλος, χρησιμοποιούμε τον αριθμό $\sum_{j=1}^m d_j 2^{m-j}$ για να αναπαραστήσουμε το δείκτη της κλάσης στο m -οστό επίπεδο στο οποίο ανήκουν τα δεδομένα.

Στο επόμενο σχήμα παρουσιάζεται η δομή ενός H-SVM με $m=4$ επίπεδα. Υπάρχουν συνολικά $2^m - 1 = 15$ SVM ταξινομητές.



Σχήμα 2.6. Δομή H-SM με m=4 επίπεδα[62]

➤ Εκτίμηση τοποθεσίας

Στόχος της εκτίμησης τοποθεσίας των κόμβων του δικτύου είναι η εκτίμηση συναρτήσεων που ονομάζονται συναρτήσεις εκτίμησης τοποθεσίας για κάθε συντεταγμένη, η οποία συσχετίζει τα δεδομένα εκπαίδευσης με την ακριβή τρισδιάστατη θέση ενός δικτυακού κόμβου:

$$s_k = L_k(x), \quad 1 \leq k \leq 3$$

όπου s_k (διάνυσμα) είναι η τιμή της κ-οστής συντεταγμένης που αντιπροσωπεύει τη θέση ενός χρήστη, δεδομένου ότι τα δεδομένα της εκπαίδευσης του χρήστη είναι x . Τέτοια δεδομένα μπορεί να σχετίζονται με τον αριθμό των κόμβων σε σχέση με τους γειτονικούς κόμβους ή άλλες πληροφορίες του δικτύου που μπορεί να δώσουν πληροφορίες για την εκτίμηση της θέσης. Οι παράμετροι του H-SVM μοντέλου που χρησιμοποιούνται για τη συνάρτηση απόφασης $D_{ij}(x)$ μεταδίδονται πρώτα από τον κόμβο φάρο σε όλους τους κόμβους στους οποίους πρόκειται να καθοριστούν οι θέσεις τους. Για δεδομένο εισόδο x' που αποκτήθηκε από έναν κόμβο για την εκτίμηση της θέσης του, ο επόμενος αλγόριθμος (αλγόριθμος 1) χρησιμοποιείται για την δημιουργία του εκτιμώμενου δείκτη της κλάσης που μπορεί να καθορίσει την συνάρτηση εκτίμησης θέσης $L_k(x)$. Μόλις ληφθεί η εκτιμώμενη κλάση, η συνάρτηση απόφασης μπορεί να

ληφθεί ανάλογα. Δεδομένου ότι η συνάρτηση $\sum_{i=1}^n a_i^* y_i K(x_i, x) + b$ αποτελεί το όριο ενός

ταξινομημένου συνόλου δεδομένων εισόδο, η συνάρτηση $\sum_{i=1}^n a_i^* y_i K(x_i, x)$ μπορεί να θεωρηθεί

ως η μεσαία γραμμή μιας τέτοια περιοχής, δηλαδή η περιοχή μεταξύ δύο συνόλων δεδομένων εισόδου στο χώρο των χαρακτηριστικών. Έτσι, το όριο για τα άλλα ταξινομημένα δεδομένων εισόδου θα είναι $\sum_{i=1}^n a_i^* y_i K(x_i, x) - b$. Κάνοντας χρήση αυτών των συναρτήσεων ορίων για δυο ταξινομημένα σύνολο ώστε να εκτιμήσουμε την τιμή ενός δεδομένου x . Επομένως, η συνάρτηση εκτίμησης τοποθεσίας για την k -οστή συντεταγμένη σχετίζεται με αυτή την κλάση με επίπεδο m και μπορεί να εκφραστεί ως:

$$s = \sum_{i=1}^n a_i^* y_i K(x_i, x) + b \quad \text{αν } d_{m-1} = 1$$

$$s = \sum_{i=1}^n a_i^* y_i K(x_i, x) - b \quad \text{αν } d_{m-1} = 0$$

όπου η συνάρτηση πυρήνα $K(x_i, x)$ είναι συνάρτηση ακτινικού τύπου (Radial Basis Function (RBF) και ορίζεται ως $K(x_i, x) = \exp(-\rho |x_i - x|^2)$. Ο λόγος που γίνεται χρήση αυτής της συνάρτησης πυρήνα είναι για το χαμηλό ποσοστό σφάλματος ταξινόμησης. Οι τιμές a_i^*, y_i και $K(x_i, x)$ σχετίζονται με τον ταξινομητή SVM $D_{m-1}, t(x)$ όπου $t = \sum_{i=1}^{m-1} 2^{m-1-i} d_i$. Ο παραπάνω αλγόριθμος μπορεί να εφαρμοστεί σε άλλα στοιχεία συντεταγμένων για τη λήψη όλων των τιμών συντεταγμένων s_k .

Αλγόριθμος 1:

1. Είσοδος: χ' και η συνάρτηση απόφασης $D_{i,j}(x)$ του H-SVM
2. Αρχικοποίηση: $p=1$ το οποίο χρησιμοποιείται για μετρητής επιπέδων H-SVM, $q=0$ που χρησιμοποιείται για μετρητής ταξινομητών SVM στο ίδιο επίπεδο.
3. Αν $D_{p-1}, q(x') \geq 1$ τότε $d_p = 1$. Ενημέρωσε $p=p+1$ και $q=2q+1$. Εκτέλεσε το βήμα 3.
4. Αλλιώς $D_{p-1}, q(x') < 1$ τότε $d_p = 0$. Ενημέρωσε $p=p+1$ και $q=2q$. Εκτέλεσε το βήμα 3.
5. Αν $p > m$, σταμάτα.
6. Έξοδος: Ο εκτιμώμενος δείκτης τάξης στο επίπεδο m καθορίζεται ως $\sum_{i=1}^m 2^{m-1-i} d_i$.

Τα βήματα 3 και 4 μεταφέρουν τα δεδομένα εισόδου από τον ταξινομητή SVM στο επίπεδο p , στο SVM ταξινομητή επόμενου επιπέδου. Εάν η έξοδος του ταξινομητή SVM στο επίπεδο p είναι 1, τότε τα δεδομένα εισόδου μεταφέρονται στο δεξί παιδί του ταξινομητή SVM που είναι στο επίπεδο p . Επομένως, ο δεύτερος δείκτης για τον SMV ταξινομητή αλλάζει από q σε $2q+1$ που είναι η εκτέλεση του βήματος 3. Παρομοίως, η εκτέλεση του βήματος 4 είναι να ανακατευθύνει τα δεδομένου εισόδου στο αριστερό παιδί του αρχικού ταξινομητή SVM στο επίπεδο p .

➤ Εκτίμηση θορύβου MIMO

Θεωρούμε τις συνθήκες για το γραμμικό κανάλι με πρόσθετο λευκό Gaussian θόρυβο σε ένα περιβάλλοντα MIMO με κεραιές μετάδοσης t και κεραιές λήψης r . Υποθέτοντας ότι το μοντέλο καναλιού είναι:

$$y = H s + n$$

όπου s είναι διάνυσμα $t \times 1$, $[s_1, s_2, \dots, s_t]^T$, n και y είναι διανύσματα $r \times 1$, $[n_1, n_2, \dots, n_r]^T$, $[y_1, y_2, \dots, y_r]^T$ αντίστοιχα. Ο πίνακας H είναι ένας πίνακας με διαστάσεις r, t , το στοιχείο στη σειρά i και τη στήλη j είναι το κέρδος καναλιού μεταξύ της j κεραιάς εκπομπής και της i κεραιάς δέκτη. Στόχος μας εδώ είναι η εκτίμηση της τιμής n του θορύβου του καναλιού. Για τον προσδιορισμό της συνάρτησης απόφασης $D_{ij}(x)$ κάθε βαθμίδας n_k , $1 \leq k \leq r$, τα δεδομένα εκπαίδευσης χ μπορούν να αναπαριστούν ως διάνυσμα $(t+1) \times 1$ ως $\chi' = [s_1, s_2, \dots, s_t, y_k]^T$ και ο πραγματικός θόρυβος n_k . Μόλις το H-SVM είναι καλά εκπαιδευμένο, μπορούμε να χρησιμοποιήσουμε τον προηγούμενο αλγόριθμο ώστε να εκτιμήσει κάθε n_k , $1 \leq k \leq r$ για δεδομένα εισόδου $\chi' = [s_1, s_2, \dots, s_t, y_k]^T$.

2.4 Κατανομή πόρων με χρήση cloud Computing και μηχανική μάθηση

Σε αυτή την ενότητα θα παρουσιάσουμε πως μια δομή cloud σε συνδυασμό με μηχανική μάθηση μπορεί να αλλάξει τελειώς τα έως τώρα δεδομένα για την ανάθεση πόρων σε ένα ασύρματο δίκτυο, ενώ θα αναφέρουμε και ένα παράδειγμα. Η κίνηση δεδομένων και οι απαιτήσεις των χρηστών κινητών συσκευών αυξάνονται συνεχώς, ενώ ο όγκος των κινητών συσκευών τείνει να ξεπεράσει τον πληθυσμό, αποτελώντας αναπόσπαστο κομμάτι της καθημερινότητας μας, είναι βέβαια ότι πιο αποδοτικές μέθοδοι ανάθεσης πόρων πρέπει να εφαρμοστούν.

Η κατανομή πόρων διαμορφώνεται θεωρητικά ως πρόβλημα βελτιστοποίησης και συνήθως γίνεται χρήση πληροφοριών πραγματικού χρόνου, ενώ συχνά χρησιμοποιούνται άπληστες μέθοδοι, οι οποίες έχουν ως αποτέλεσμα απώλεια απόδοσης. Υποβοηθούμενη από το cloud computing, μια τεράστια ποσότητα ιστορικών δεδομένων για σεναρία μπορεί να συλλεχθεί για την εξαγωγή ομοιοτήτων μεταξύ των σεναρίων, χρησιμοποιώντας μηχανική μάθηση. Επιπλέον η βέλτιστη ή σχεδόν βέλτιστες λύσεις ιστορικών σεναρίων μπορούν να αναζητηθούν εκτός σύνδεσης και να αποθηκευτούν, για μεταγενέστερη χρήση.

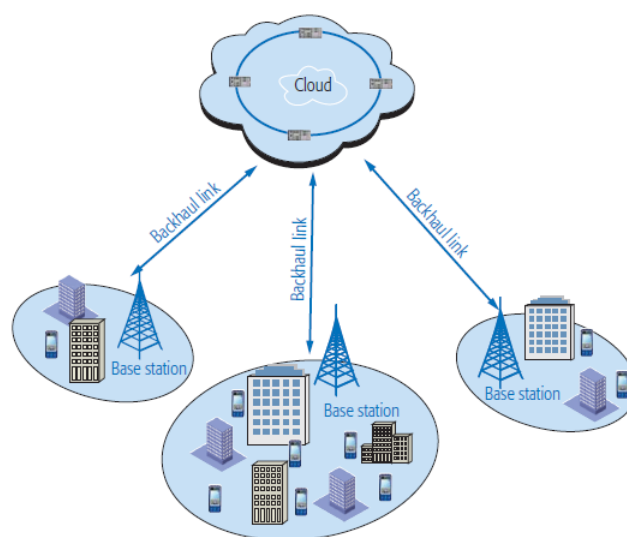
Η συνολική απόδοση των ασύρματων δικτύων εξαρτάται από τον τρόπο εκμετάλλευσης του ασύρματου καναλιού καθώς και την σωστή διαχείριση των πόρων (ζώνη συχνοτήτων, ορθογώνιοι κώδικες, σχήμα διαμόρφωσης, ισχύς εκπομπής κτλ), την δίκαιη κατανομή τους στους χρήστες, καθώς όλοι οι χρήστες μοιράζονται το ίδιο φάσμα και την ίδια ασύρματη υποδομή. Αφού συλλεχθεί η κατάσταση του καναλιού (CST-Channel State Information) και οι απαιτήσεις του χρήστη (QOS-quality of service) το πρόβλημα βελτιστοποίησης λύνεται online, ενώ η λύση πρέπει να ληφθεί αμέσως καθώς το κανάλι αλλάζει γρήγορα. Ένα σενάριο με πολλαπλούς χρήστες θα δυσκόλευε ακόμα περισσότερο την εύρεση της βέλτιστης λύσης, ενώ το QOS αυξάνεται συνεχώς. Σε πρακτικά συστήματα τηλεπικοινωνιών οι ραδιοπόροι εκχωρούνται

δυναμικά ανάλογα τις στιγμιαίες λαμβανόμενες πληροφορίες συμπεριλαμβανομένου της κατάστασης του καναλιού και τις απαιτήσεις QOS των χρηστών. Η φθηνή αποθήκευση των δεδομένων σε cloud το καθιστά πολύ εύκολο να αποθηκεύσει τις πληροφορίες ως δεδομένα για ιστορικά σενάρια που προηγουμένως θα αγνοούσαμε και θα τα πετάγαμε στα σκουπίδια. Χρησιμοποιώντας τις ομοιότητες μεταξύ σεναρίων, οι λύσεις κατανομής πόρων στα ιστορικά σενάρια μπορούν να αξιοποιηθούν για να βελτιώσουν την κατανομή πόρων του τρέχοντος σεναρίου, κάνοντας χρήση μηχανικής μάθησης με επίβλεψη. Οι λύσεις κατανομής πόρων για ιστορικά σενάρια μπορούν να αναζητηθούν εκτός σύνδεσης και να αποθηκεύεται εκ των προτέρων. Όταν οι μετρήσεις του τρέχοντος σεναρίου εμφανιστούν, δεν είναι απαραίτητο να χρησιμοποιήσουμε συμβατικές μεθόδους (Lagrangian relaxation, άπληστοι αλγόριθμοι) για την επίλυση της κατανομής πόρων στο διαδίκτυο. Αντ' αυτού, πρέπει μόνο να συγκρίνουμε το τρέχον σενάριο με ιστορικά σενάρια και να βρούμε το πιο παρόμοιο. [67] Έτσι χρησιμοποιούμε η λύση του πιο παρόμοιου ιστορικού σεναρίου για την κατανομή των πόρων για το τρέχον σενάριο. Η offline επεξεργασία των δεδομένων καθιστά δυνατή τη χρήση της υπολογιστικής ισχύς του cloud χρησιμοποιώντας τεχνικές για να βρει τη βέλτιστη ή τη σχεδόν βέλτιστη λύση κατανομής πόρων για ιστορικά σενάρια, το οποίο μπορεί να βελτιώσει την απόδοση του.

➤ Μαθηματικό μοντέλο για απόδοση πόρων

Όπως φαίνεται στο επόμενο σχήμα, η αρχιτεκτονική των ασύρματων δικτύων επικοινωνιών υποβοηθούμενα από cloud computing αποτελείται από τρία κύρια μέρη:

- Συγκεντρωμένη αποθηκευτική και υπολογιστική ισχύ σε μορφή cloud
- Σταθμός βάσης (BS) με λειτουργία ασύρματης πρόσβασης
- Συνδέσεις Backhaul που παρέχουν τις μετρήσεις δεδομένων πραγματικών σεναρίων από το BS στο cloud και μεταφέρουν το σχήμα μηχανικής μάθησης για κατανομή πόρων στο BS.



Σχήμα 2.7. Ασύρματη μετάδοση υποβοηθούμενη από cloud computing.[68]

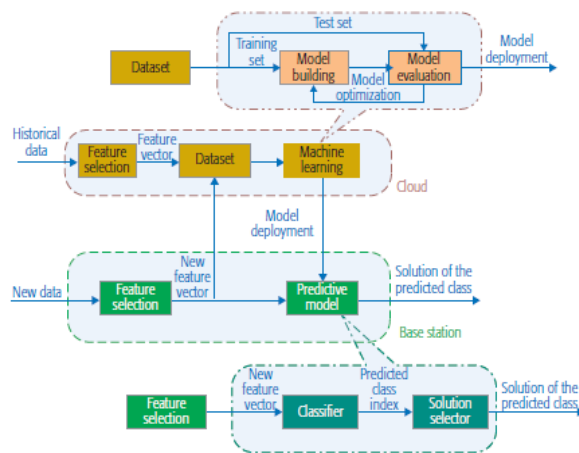
Η κατανομή πόρων στα BS μπορεί να διατυπωθεί ως μαθηματικό πρόβλημα ελαχιστοποίησης της συνάρτησης :

$$f(x,a), \chi \in s, \text{ (εξίσωση 1)}$$
$$\text{υπό τους περιορισμούς, } g_i(\chi,a) \leq 0 \quad i = 1, \dots, m \text{ και } h_i(\chi,a) = 0 \quad i = 1, \dots, p$$

όπου χ είναι το διάνυσμα μεταβλητών του προβλήματος, $f()$ είναι η συνάρτηση βελτιστοποίησης (που πρέπει να ελαχιστοποιηθεί ή να μεγιστοποιηθεί), a είναι ένα διάνυσμα παραμέτρων που καθορίζει το πρόβλημα (παραμέτροι που πρέπει να επιλεγούν), $\{g_i\}_{i=1}^m$ (inequality constraint), $\{H_i\}_{i=1}^m$ (equality constraint) αποτελούν τους περιορισμούς του προβλήματος ενώ s είναι το σύνολο των περιορισμών. Οι μεταβλητές του διανύσματος x θα περιέχουν στοιχεία σχετικά με την κατανομή πόρων όπως π.χ. ισχύ εκπομπής, οι μεταβλητές του διανύσματος a , θα περιέχουν πληροφορίες σχετικά με παραμέτρους της μετάδοσης (ή του συστήματος) όπως bandwidth επίπεδο θορύβου κτλ ενώ οι συναρτήσεις $\{g_i\}_{i=1}^m$, $\{H_i\}_{i=1}^m$ θα χρησιμοποιηθούν για να καθορίσουν το εκάστοτε σενάριο και τους περιορισμούς του όπως, περιορισμούς στους πόρους, απαίτηση QOS χρηστών, παρεμβολές κτλ.

➤ Δομή πλαισίου μηχανικής μάθησης

Για ένα σύστημα ασύρματης μετάδοσης με cloud computing ένας μεγάλος όγκος από ιστορικά δεδομένα θα έχουν συλλεχθεί και αποθηκευτεί. Η ισχυρή υπολογιστική ικανότητα του cloud αξιοποιείται για την αναζήτηση βέλτιστων ή σχεδόν βέλτιστων λύσεων για αυτά τα ιστορικά σενάρια. Με την χρήση μεθόδων ταξινόμησης αυτών των λύσεων, οι ομοιότητες που κρύβονται σε αυτά τα ιστορικά σενάρια εξάγονται ως ένα πρόγραμμα-σχήμα κατανομής πόρων που βασίζεται σε μηχανική μάθηση. Το σύστημα κατανομής πόρων που βασίζεται στη μηχανική μάθηση θα οδηγηθεί σε έναν BS (Base Station-σταθμός βάσης) ώστε να τον οδηγήσει στην πιο αποτελεσματική κατανομή πόρων. Όταν ένα BS τοποθετηθεί σε μια νέα περιοχή όπου προφανώς δεν υπάρχουν ιστορικά δεδομένα σεναρίων, τα αρχικά ιστορικά δεδομένα μπορούν να δημιουργηθούν από ένα αφηρημένο μαθηματικό μοντέλο με ρεαλιστικές τοποθεσίες BS, ακριβή αποτυπώματα κτιρίου, υποθετική κατανομή χρηστών και απαιτήσεις, καθώς και ασύρματα μοντέλα διάδοσης. Όταν το νέο BS τεθεί σε λειτουργία, τα δεδομένα σεναρίων σε πραγματικό χρόνο θα συλλεχθούν και θα χρησιμοποιηθούν ως ιστορικά δεδομένα για μάθηση, βελτιώνοντας την απόδοση ανάθεσης πόρων στον νέο BS. Στο επόμενο σχήμα μια δομή μηχανικής μάθησης παρουσιάζεται.



Σχήμα 2.8. Δομή μηχανικής μάθησης για ανάθεση πόρων [68]

Η επιλογή χαρακτηριστικών είναι κρίσιμη καθώς επιπλέον χαρακτηριστικά θα αυξήσουν το μέγεθος του συνόλου δεδομένων έχοντας ως αποτέλεσμα να πρέπει να αυξηθούν και οι υπολογιστική πόροι αλλά μπορεί και να οδηγήσουν σε απρόσμενα αποτελέσματα. Τα ιστορικά δεδομένα σεναρίων περιέχουν χαρακτηριστικά όπως CSI, SNR, τον αριθμό του χρήστη το IMIS (International mobile subscriber identification number) κτλ. Δεδομένα όπως IMSIS δεν είναι χρήσιμα για την απόδοση των πόρων, και μπορούν να μην συμπεριληφθούν στο διάνυσμα παραμέτρων a στο πρόβλημα βελτιστοποίησης. Έτσι το αρχικό διάνυσμα a , μπορεί επιπλέον να χωριστεί σε δύο κατηγορίες, δυναμικά χρονικά μεταβαλλόμενη (dynamic time-variant) και στατική χρονικά αμετάβλητη (static time-invariant). Ορισμένα χαρακτηριστικά μένουν σταθερά όπως ο αριθμός των υποφερόντων, μέγιστη ισχύ εκπομπής, αριθμός κεραιών (static time-invariant) και δεν χρειάζεται να συμπεριληφθούν στα διάνυσμα χαρακτηριστικών, ενώ άλλα χαρακτηριστικά όπως κατάσταση καναλιού, επίπεδα παρεμβολών αλλάζουν συνέχεια (dynamic time variant) και πρέπει οπωσδήποτε να συμπεριληφθούν στο διάνυσμα χαρακτηριστικών. Ο παραπάνω διαχωρισμός θα έχει ως αποτέλεσμα την μείωση του διανύσματος χαρακτηριστικών και συνεπώς την αποδοτικότερη αποτελεσματικότητα του συστήματος. Ωστόσο, λόγω του μεγάλου μεγέθους σε δεδομένα μια προεπεξεργασία θα πρέπει να υπάρχει ώστε να αποφεύγονται δεδομένα που ήδη υπάρχουν ή δεδομένα που μπορεί να οδηγήσουν σε απρόσμενα αποτελέσματα.

Έχοντας το σύνολο εκπαίδευσης (ιστορικά δεδομένα), ένας αλγόριθμος μηχανικής μάθησης με εκπαίδευση υιοθετείται για εύρεση των ομοιοτήτων που κρύβονται στα ιστορικά δεδομένα. Με αυτόν τον τρόπο, μπορεί να δημιουργηθεί ένα μοντέλο πρόβλεψης που θα χρησιμοποιηθεί για τη λήψη μιας απόφασης κατανομής πόρων για ένα μελλοντικό σενάριο, ενώ με τη βοήθεια του cloud computing, μπορούν να χρησιμοποιηθούν προηγμένες τεχνικές υπολογιστών για την αναζήτηση λύσεων για το πρόβλημα βελτιστοποίησης. Όλα τα εκπαιδευτικά διανύσματα με τις ίδιες λύσεις ταξινομούνται σε μία τάξη και κάθε τάξη σχετίζεται με τη δική της λύση. Το πρόβλημα της κατανομής πόρων τώρα μετατρέπεται σε ένα πρόβλημα ταξινόμησης πολλαπλών κλάσεων, και χρειάζεται δύο συναρτήσεις ώστε να μπορέσει να λειτουργήσει το μοντέλο πρόβλεψης. Για την πρόβλεψη της κλάσης για ένα μελλοντικό σενάριο που μπορεί να περιγραφεί ως ταξινομητής $I = \text{classifier}(F_T)$, όπου F_T είναι το διάνυσμα που προκύπτει από το σενάριο και I είναι ο δείκτης της κλάσης, δείχνει ότι τα υπάρχον σενάρια ανήκει στην

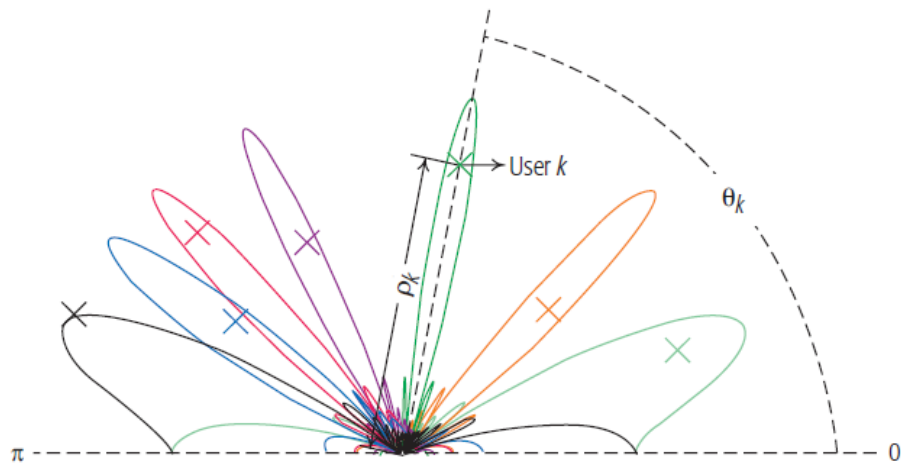
I-κλάση. Στη συνέχεια η σχετική λύση της κλάσης I επιλέγεται ώστε να γίνει η κατάλληλη ανάθεση πόρων , ενώ κάνοντας χρήση της ζεύξης backhaul τα δεδομένα μεταφέρονται στον σταθμό βάσης. Ο σταθμός βάσης μπορεί με τα δεδομένα που μετράει σε πραγματικό χρόνο να δημιουργήσει ένα καινούργιο διάνυσμα χαρακτηριστικών δίνοντας τα στο μοντέλο μάθησης ώστε να γίνει ανανέωση των δεδομένων του που σχετίζονται με πραγματικά σενάρια , βελτιώνοντας το μοντέλο μάθησης σε χρονικές στιγμές που δεν είναι υπερφορτωμένο. Όλα τα χαρακτηριστικά με την ίδια λύση τοποθετούνται στην ίδια κλάση , που χαρακτηρίζεται από έναν μη αρνητικό ακέραιο αριθμό. Η κάθε κλάση σχετίζεται με μια μοναδική λύση. Καινούργια δεδομένα πραγματικού σεναρίου επεξεργάζονται και να ταξινομούνται στην συνέχεια στην σωστή κλάση ώστε να γίνει η κατανομή πόρων. Έτσι μπορεί να δημιουργηθεί ένα μοντέλο πρόβλεψης.

Από μαθηματικής πλευράς αυτό μπορεί χαρακτηριστεί από ένα σύνολο διανυσμάτων στο χώρο. Αν η τάξη του διανύσματος χαρακτηριστικών είναι δυο αυτό μπορεί να θεωρηθεί ως ένα δισδιάστατο επίπεδο το οποίο μπορεί να χωριστεί σε υπερεπίπεδα. Καθώς τα χαρακτηριστικά μεταξύ των διαφόρων σεναρίων παρουσιάζουν ομοιότητες τα διανύσματα θα βρίσκονται κοντά και στον ίδιο υποχώρο , έτσι οι κρυμμένες ομοιότητες μπορεί να εξευρενηθούν από ένα ταξινομητή , που προβλέπει την κλάση -τον υποχώρο του νέου χαρακτηριστικού. Το πρόβλημα τώρα ανάγεται σε ένα πρόβλημα ταξινόμηση πολλαπλών κλάσεων μηχανικής μάθησης. Η επιλογή του αλγορίθμου εξαρτάται από πολλούς παράγοντες , όπως την υπολογιστική ισχύ , το χρόνο απόκρισης την κατανάλωση πόρων την ακρίβεια πρόβλεψης , και το μέγεθος των χαρακτηριστικών. Υιοθετώντας τον αλγόριθμο κ-πλησιέστερου γείτονας (όπως περιγράφηκε στην εισαγωγή) , όταν ένα νέο χαρακτηριστικό διάνυσμα φτάσει ο αλγόριθμος κ-πλησιέστερου γείτονα θα επιλέξει όλους τους κ πλησιέστερους γείτονες για το νέο διάνυσμα από το σύνολο εκπαίδευσης (εξαρτώμενος απο το κ). Στη συνέχεια το νέο διάνυσμα κρίνεται ότι ανήκει στην πιο κοινή κλάση σύμφωνα με του γείτονές του.

➤ Κατανομή δέσμης σε massive MIMO

Σε αυτό το σημείο θα δώσουμε ένα παράδειγμα μηχανικής μάθησης για το πρόβλημα της κατανομής δέσμης σε ένα σύστημα MIMO single-cell.

Υποθέτουμε ότι ο σταθμός βάσης βρίσκεται στο κέντρο της κυψέλης και κ χρήστες είναι ομοιόμορφα κατανεμημένοι μέσα στην κυψέλη με μια κεραία. Ένας μεγάλος σταθερός αριθμός $N \gg K$ από δέσμες σχηματίζονται στο δίκτυο, από μια συστοιχία N ιστροπικών κεραιών. Ένα τέτοιο σύστημα παρουσιάζεται στο επόμενο σχήμα, όπου κάθε δέσμη με το ίδιο χρώμα αντιστοιχεί σε έναν χρήστη , ενώ τα (p_k, θ_k) (ακτίνα, μοίρες) είναι οι πολικές συντεταγμένες [69] του χρήστη κ.



Σχήμα 2.9. Απεικόνιση της κατανομής δέσμης σε συστήματα MIMO. Κάθε σταυρός αντιπροσωπεύει έναν χρήστη. [68]

Για να εξυπηρετήσουμε πολλούς χρήστες ταυτόχρονα πρέπει να καταναείμουμε αποτελεσματικά τις δέσμες ώστε να μεγιστοποιηθεί ο ρυθμός μετάδοσης.

Καθώς ο αριθμός από τις δέσμες N είναι πολύ μεγαλύτερος από τους χρήστες, και σε κάθε χρήστη αντιστοιχεί μια δέσμη μόνο μερικές από τις δέσμες θα είναι ενεργές, οπότε αρχικά θα πρέπει να αποφασιστεί ποιές δέσμες θα είναι ενεργές. Αυτό μπορεί να επιλυθεί κάνοντας χρήση του συστήματος μηχανικής μάθησης, καθώς η δέσμη που θα αντιστοιχεί στο κάθε χρήστη θα είναι η έξοδος του μοντέλου πρόβλεψης. Υποθέτοντας ένα κανάλι οπτικής επαφής (line of sight-LOS), και καθώς τα κέρδη των N δεσμών για τους K χρήστες καθορίζονται από την τοποθεσία των K χρηστών, η θέση του κάθε χρήστη $u = [(\rho_1, \theta_1), \dots, (\rho_K, \theta_K)]$ δίνεται ως δεδομένα εισόδου η οποία περιέχει πληροφορίες για την απόσταση-ακτίνα του χρήστη αλλά και την φάση του. Δεδομένου ότι το κέρδος της κάθε δέσμης διαφέρει για κάθε χρήστη ανάλογα την φάση του, ο συνολικός ρυθμός μετάδοσης καθορίζεται από τις πληροφορίες φάσης του χρήστη. Επομένως το διάνυσμα χαρακτηριστικών F_u της διάταξης του χρήστη u καθορίζεται από

$$F_T = [\cos\theta^{(1)}, \cos\theta^{(2)}, \dots, \cos\theta^{(K)}],$$

όπου $\theta^{(1)} \leq \dots \theta^{(K)}$.

Πριν την ανάθεση πόρων πρέπει πρώτα να εκπαιδύσουμε το μοντέλο μας από ένα μεγάλο σύνολο δεδομένων διάταξης χρηστών το οποίο μπορεί να παραχθεί και από υπολογιστή, ανάλογα τη κατανομή. Για κάθε δεδομένο εκπαιδευσης χρήστη το διάνυσμα χαρακτηριστικών που σχηματίζεται, σχετίζεται με μια δέσμη η οποία μπορεί να καθοριστεί από τον αλγόριθμό μας. Χάρη στην ισχυρή υπολογιστική ικανότητα του νέφους, μπορούν να υιοθετηθούν βέλτιστες εξαντλητικές αναζητήσεις ή σχεδόν βέλτιστοι metaheuristics αλγόριθμοι. Αφού κάθε χαρακτηριστικό εκπαιδευσης συσχετιστεί με την δικιά του δέσμη-λύση, όλα τα διανύσματα χαρακτηριστικών ταξινομούνται σε κλάσεις ανάλογα την ενεργή δέσμη τους. Τα διάνυσμα χαρακτηριστικών που βρίσκονται στην ίδια κλάση μοιράζονται την ίδια δέσμη. Ένα μοντέλο πρόβλεψης δέσμης μπορεί να δημιουργηθεί κάνοντας χρήση ενός αλγορίθμου ταξινόμησης (π.χ. κ-πλησιέστερου γείτονα), ο οποίος μπορεί στη συνέχεια να αξιολογηθεί και να

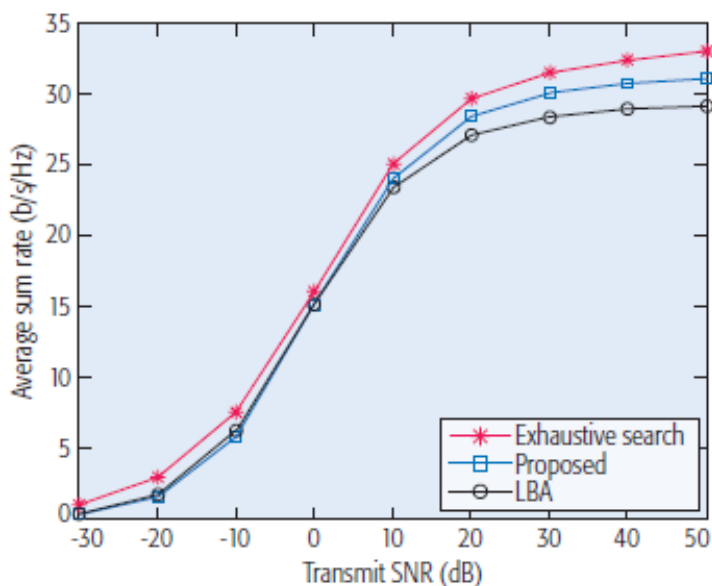
βελτιστοποιηθεί για εγγύηση της απόδοσή τους.

Το νέο μοντέλο πρόβλεψης μπορεί να αποδοθεί σε έναν σταθμό βάσης για την κατανομή της δέσμης στους χρήστες. Για ένα νέο χρήστη με διάταξη u_i και διάνυσμα χαρακτηριστικών F_{u_i} ορίζουμε την απόσταση του από ένα διάνυσμα εκπαίδευσης F_{T_j} , τότε η απόσταση είναι :

$$d_{u_i, T_j} = \|F_{u_i} - F_{T_j}\|^2,$$

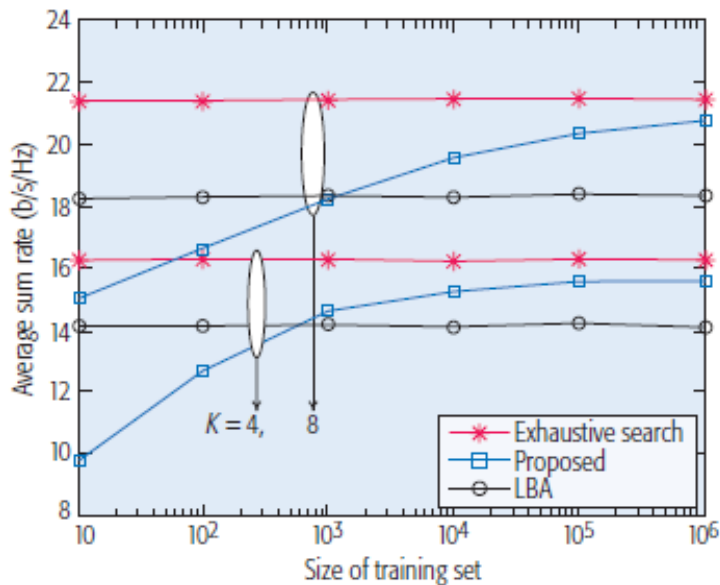
και το διάνυσμα του k πλησιέστερου γειτονικά με την μικρότερη απόσταση επιλέγεται από τον αλγόριθμο, για το νέο αυτό χαρακτηριστικό. Σύμφωνα με το αλγόριθμο αυτό η πιο κοινή κλάση επιλέγεται ως κλάση προβλέψεις ανάλογα με τα δεδομένα εισόδου και στη συνέχεια αποδίδεται στο χρήστη η δέσμη που αντιστοιχεί στην κλάση επιλογής. Με βάση τις πληροφορίες ενεργής δέσμης, κάθε ενεργή δέσμη κατανέμεται όσο τον δυνατόν καλύτερα στο χρήστη αποδίδοντάς του το υψηλότερο SINR, υποθέτοντας ίση ισχύ κατανομής σε κάθε χρήστη. Το νέο διάνυσμα F_{u_i} που θα συλλεχθεί μπορεί να αποθηκεύεται για ενημέρωση του συνόλου δεδομένων εκπαίδευσης και βελτιστοποίησης του αλγορίθμου.

Στο παρακάτω διάγραμμα παρουσιάζεται ο μέσος ρυθμός μετάδοσης (b/s/Hz) για τον αλγόριθμο μηχανικής μάθησης και συγκρίνεται με τους αλγορίθμους εξαντλητικής αναζήτησης και τον αλγόριθμο low-complexity beam allocation (LBA) [70]



Σχήμα 2.10. Σύγκριση αλγορίθμων ,με τον αλγόριθμο k πλησιέστερου γείτονα με $k=1$, $N=16$, 10^6 δεδομένα εκπαίδευσης.[68]

Μπορεί επίσης να παρατηρηθεί από το επόμενο σχήμα ότι με ένα μεγαλύτερο εκπαιδευτικό σύνολο, ο αλγόριθμός μας ξεπερνά τον αλγόριθμο LBA και πλησιάζει τον βέλτιστο αλγόριθμο εξαντλητικής αναζήτησης αποδεικνύοντας ότι το προτεινόμενο σύστημα μηχανικής μάθησης για κατανομή πόρων ξεπερνά τις συμβατικές τεχνικές.



Σχήμα 2.11. Σύγκριση ρυθμού μετάδοσης με των αριθμό δεδομένων εκπαίδευση για τον αλγόριθμο κ-πλησιέστερου γείτονα με $\kappa=1$, $\kappa=8$ και εκπεμπόμενο SNR=20 dB [68]

Αξίζει να σημειωθεί ότι ο αλγόριθμος κ-πλησιέστερου γείτονα παρέχει καλά αποτελέσματα αλλά οι αποστάσεις μεταξύ νέων δεδομένων και υπαρχόντων δεδομένων εκπαίδευσης υπολογίζονται σε πραγματικό χρόνο. Ως αποτέλεσμα, με μεγάλο αριθμό δεδομένων εκπαίδευσης, η πολυπλοκότητα του υπολογισμού θα γίνει πολύ υψηλή σε πρακτικά συστήματα. Είναι επομένως σημαντικό να σχεδιαστεί ένας ταξινομητής πολλαπλών κλάσεων χαμηλής πολυπλοκότητας, με πιο προηγμένες τεχνικές.

Μια από τις πολλές υποσχόμενες τεχνικές είναι να μετασχηματιστεί το πρόβλημα ταξινόμησης πολλών κλάσεων σε ένα σύνολο δυαδικών προβλημάτων που επιλύονται αποτελεσματικά χρησιμοποιώντας δυαδικούς ταξινομητές. Η μηχανική μάθηση με διανύσματα υποστήριξης (SVM) έχει θεωρηθεί ως ένας από τους πιο ισχυρούς και επιτυχημένους αλγόριθμους για το σχεδιασμό δυαδικών ταξινομητών χαμηλής πολυπλοκότητας, ο οποίος καθορίζει την κλάση ενός νέου διανύσματος χαρακτηριστικών χρησιμοποιώντας γραμμικά όρια (υπερεπίπεδα) σε χώρο υψηλότερων διαστάσεων, έτσι οι κλάσεις χωρίζονται με λίγα μόνο υπερεπίπεδα. Η κλάση καθορίζεται με βάση τις πλευρές των υπερεπίπεδων στο οποίο εμπίπτει το νέο διάνυσμα χαρακτηριστικών. Σε σύγκριση με τον αλγόριθμο κ-πλησιέστερου γείτονα, η πολυπλοκότητα των δυαδικών ταξινομητών που βασίζονται σε SVM είναι πολύ χαμηλότερη.

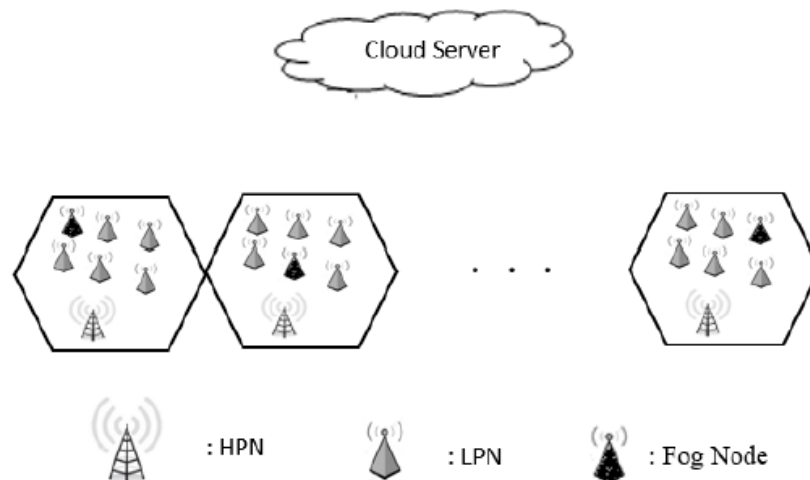
Για το προαναφερθέν παράδειγμα κατανομής δέσμης, ο συνολικός αριθμός ενεργών δεσμών είναι 2^N , υπάρχουν δηλαδή το πολύ 2^N κλάσεις, που σημαίνει ότι η πολυπλοκότητα είναι $O(2^N)$ για καθορισμός της κλάσης του σεναρίου, ενώ η πολυπλοκότητα της εξαντλητικής αναζήτησης είναι $O(N^K)$. Το, προτεινόμενο σύστημα μηχανικής μάθησης μπορεί να προσεγγίσει το βέλτιστο της απόδοσης της εξαντλητικής αναζήτησης με χαμηλή πολυπλοκότητα. Σε πραγματικά σενάριο όπου το πλήθος των χαρακτηριστικών είναι πολύ μεγαλύτερο (πολύ μεγάλο πλήθος χρηστών, συναυλίες εμπορικά κέντρα) αλγόριθμοι χαμηλότερης πολυπλοκότητας και καλύτερη απόδοσης πρέπει να χρησιμοποιηθούν. Ωστόσο σε πραγματικά συστήματα περισσότεροι από έναν σταθμό βάσης μπορεί να εξυπηρετούν το χρήστη ειδικά αν βρίσκεται στην άκρη της κυψέλης. Αυτό έχει ως αποτέλεσμα νέες προκλήσεις

να δημιουργούνται σχετικά με το τι θα πρέπει να περιέχει το διάγραμμα χαρακτηριστικών, σε ποιο σύνολο εκπαίδευσης πρέπει να αναζητήσουμε την βέλτιστη λύση και ποιος αλγόριθμος πρέπει να χρησιμοποιηθεί. Πρόσφατες τεχνικές βαθιάς μηχανικής μάθησης έχουν δείξει καλά αποτελέσματα σε αποδόσεις συστήματος. [71][72]

2.5 Χαμηλό latency σε δίκτυο 5G με μάθηση χωρίς επίβλεψη

Σε αυτή την ενότητα θα παρουσιάσουμε τον συνδυασμό ενός fog-edge network (FN) με κυψελωτό δίκτυο ώστε να μειώσουμε το latency, με τη βοήθεια μάθησης χωρίς εκπαίδευση και πιο συγκεκριμένα με συσταδοποίηση. Ένα δίκτυο fog είναι μια αρχιτεκτονική που χρησιμοποιεί συσκευές (edge devices) για την πραγματοποίηση ενός σημαντικού αριθμού υπολογισμών αποθήκευσης και επικοινωνίας μέσω του δικτύου κορμού.[192] Το δίκτυο αυτό προορίζεται για καταναμημένους υπολογιστές όπου πολλές περιφερειακές συσκευές συνδέονται σε ένα cloud. Πολλές από αυτές τις συσκευές θα παράγουν ογκώδη ακατέργαστα δεδομένα και αντί να προωθούν όλα αυτά τα δεδομένα σε διακομιστές που βασίζονται σε σύννεφο για επεξεργασία, θα κάνουν όσο το δυνατόν περισσότερη επεξεργασία χρησιμοποιώντας υπολογιστικές μονάδες που βρίσκεται μαζί με τις συσκευές δημιουργίας δεδομένων, έτσι ώστε η επεξεργασία και όχι τα ανεπεξέργαστα δεδομένα να προωθούνται, και οι απαιτήσεις του εύρους ζώνης μειώνονται. Ένα επιπλέον πλεονέκτημα είναι ότι τα επεξεργασμένα δεδομένα είναι πιθανότατα απαραίτητα από τις ίδιες συσκευές που δημιούργησαν τα δεδομένα, έτσι ώστε με την επεξεργασία τοπικά και όχι απομακρυσμένα, έτσι το latency μεταξύ της εισαγωγής και της απόκρισης ελαχιστοποιείται. Το fog δίκτυο αποτελείται από κόμβους υψηλής ισχύος (high power node-HPN) και από κόμβους χαμηλής ισχύος (low power nodes-LPN). Σε ένα fog network κάποιοι κόμβοι (π.χ. access point, routers, small cells) χαρακτηρίζονται ως κόμβοι fog στο δίκτυο cloud των πραγμάτων (IOT) (cloud-to-things)[193][194] για τον έλεγχο και παροχή υπηρεσιών στις τελικές συσκευές. Μια νέα προοπτική είναι η ενσωμάτωση του fog network σε ένα ετερογενές δίκτυο (heterogeneous networks -HetNets) που θα αποτελείται από έναν κόμβο HPN και πολλούς κόμβους LPN, όπου κάποιοι LPN θα αναβαθμίζονται σε κόμβους fog. Αυτό μπορεί να βελτιώσει πολύ την απόδοση ενός ετερογενούς δικτύου τόσο στην μείωση παρεμβολών μεταξύ των LPN όσο και στην ικανότητα ελέγχου των κόμβων fog αλλά και στην παροχή υπηρεσιών στις τελικές συσκευές. Οι τοποθεσίες των LPN οι οποίοι αναβαθμίζονται σε κόμβους fog θεωρούνται, υποθέτοντας ότι ο αριθμός των κόμβων fog έχει δοθεί εξ αρχής καθώς ότι είναι γνωστές και οι θέσεις όλων των fog κόμβων μέσα σε μια κυψέλη. Πιο συγκεκριμένα, τα LPN που αναβαθμίζονται σε fog nodes καθορίζονται με βάση έναν αλγόριθμο μη επιτηρούμενης μάθησης (soft-clustering). Τα LPN συσταδοποιούνται έτσι ώστε ο κύριος κόμβος (leader node) κάθε συστάδας να αναβαθμίζεται σε fog κόμβο. Η μετρική για την απόφαση της ομαδοποίησης θα είναι η μείωση του latency. Ένα δίκτυο ομίχλης (fog network) αποτελείται από δύο επίπεδα, ένα επίπεδο δεδομένων και ένα επίπεδο ελέγχου. Στο επίπεδο δεδομένων, ο fog υπολογιστής και οι συναφείς υπηρεσίες υπολογιστών προσπαθούν να επιτύχουν τους στόχους

των πελατών μέσω των μοναδικών χαρακτηριστικών του, όπως η πυκνή γεωγραφική κατανομή, η συγκέντρωση πόρων, η μείωση καθυστέρησης και η εξοικονόμηση εύρους ζώνης για την επίτευξη καλύτερης ποιότητας υπηρεσίας (QoS). Στο επίπεδο ελέγχου, το fog δίκτυο μπορεί να συντονίσει πολλές συσκευές για τον μετρισμό των παρεμβολών, οι οποίες είναι τα LPN. Στο επόμενο σχήμα φαίνεται μια δομή ετερογενούς δικτύου μαζί με τους κόμβους HPN και LPN, στο οποίο φαίνεται η προκλήση της επιλογής θέσης των fog κόμβων.



Σχήμα 2.12. Δίκτυο fog σε ετερογενές δίκτυο [195]

Για την περιοχή εξυπηρέτησης των κόμβων fog μια προσέγγιση θα μπορούσε να ήταν η χρήση του μοντέλου Voronoi tessellation [197] έτσι ώστε κάθε LPN να επιλέγει έναν fog κόμβο σύμφωνα με την πλησιέστερη Ευκλείδεια απόσταση. Η προσέγγιση αυτή θα μπορούσε να υλοποιηθεί με τον αλγόριθμο μηχανικής μάθησης K-mean (hard k-mean), έχοντας ως βασικό μειονέκτημα ότι η πλησιέστερη Ευκλείδεια απόσταση μπορεί να επιλέξει καλάνι με χαμηλή ποιότητα το οποίο θα υποβαθμίσει την απόδοσης επικοινωνίας έχοντας ως αποτέλεσμα να αυξήσει το latency. Έτσι ένας άλλος παρόμοιος αλγόριθμος k-mean (soft k-mean) που να μειώνει το latency πρέπει να βρεθεί, ώστε κάθε LPN να μπορεί να είναι ένας fog κόμβος σύμφωνα με την ποιότητα του καναλιού, και ένα LPN να μπορεί να συνδεθεί με πολλούς κόμβους fog. Στο hard-clustering, κάθε σημείο δεδομένων είτε ανήκει πλήρως σε μια συστάδα είτε όχι. Στο soft-clustering, αντί να τοποθετείται κάθε σημείο δεδομένων σε μια ξεχωριστή συστάδα, εκχωρείται μια πιθανότητα (probability) ή μέγιστη πιθανοθάνεια (likelihood) αυτού του σημείου δεδομένων σε αυτές τις ομάδες. [196]

Υποθέτοντας ότι ο αριθμός των κόμβων fog και τα LPN είναι γνωστά εξ αρχής και υπάρχει ένα σύνολο δεδομένων που αποτελείται από τις γεωγραφικές τοποθεσίες των LPN :

$X = \{x_1, x_2, \dots, x_N\} \in R^2$ όπου N είναι ο συνολικός αριθμός των LPN και K από αυτά θα αναβαθμιστούν σε fog notes. Ως εκ τούτου, αυτό το σύνολο δεδομένων συγκεντρώνεται με την ακόλουθη αντικειμενική συνάρτηση έτσι ώστε ο κύριος (leader) των συστάδων (cluster-header) να δίνει τις θέσεις των fog κόμβων ως $F = \{f_1, \dots, f_n\} \in R^2$.

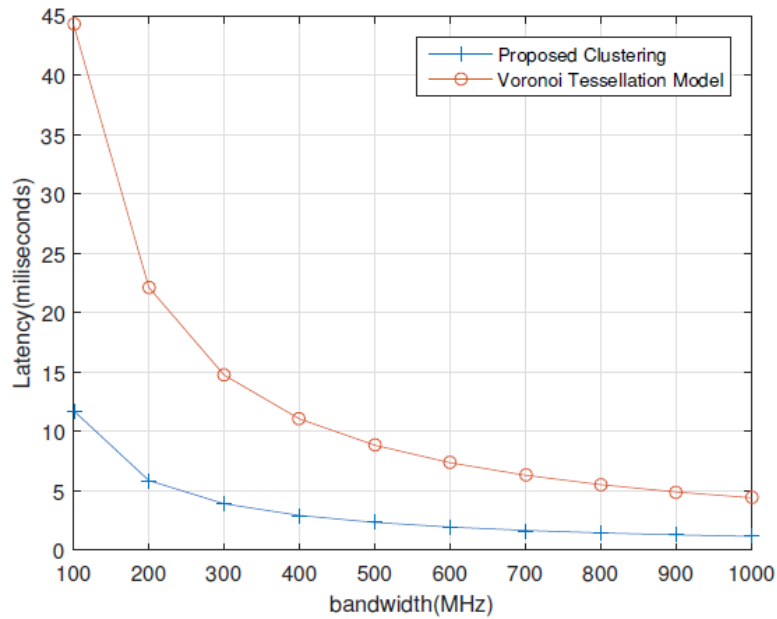
$$J = \sum_{n=1}^N \sum_{k=1}^K F(\gamma_{nk} f(x_n, f_k))$$

Η βελτιστοποίηση της εξίσωσης σχηματίζει τις συστάδες όπου το γ_{nk} δείχνει την πιθανότητα σύνδεσης μεταξύ ενός LPN και ενός fog κόμβου $\gamma_{nk} \in [0, 1]$. Εάν αυτός ήταν ένας αλγόριθμος hard K-mean γ_{nk} θα είναι είτε 0 ή 1. Το $f(x_n, f_k)$ μετρά την ομοιότητα οποιουδήποτε σημείου δεδομένων x_n $n=1,2,\dots,N$ με κόμβο ομίχλης f_k $k=1,2,\dots,K$ και πολλαπλασιάζοντας το με γ_{nk} αποτελεί την αντικειμενική συνάρτηση.

Ο πρωταρχικός στόχος της μεγιστοποίησης της εξίσωση είναι να προσδιοριστεί μια ομαδοποίηση που μειώνει την καθυστέρηση εντός του δικτύου. Για να γίνει αυτό, τα LPN συσταδοποιούνται σύμφωνα με την ισχύ του καναλιού τους, πράγμα που σημαίνει ότι κάθε LPN συσχετίζεται με κόμβους fog που έχουν κανάλια πάνω από μια συγκεκριμένη ποιότητα. Είναι σημαντικό να τονιστεί ότι κάθε LPN μπορεί να συσχετιστεί από πολλούς κόμβους ομίχλης ανάλογα με τον αλγόριθμο soft-clustering ο οποίος είναι : [195]

- A. Ορίστε τον αριθμό των κόμβων ομίχλης που θα αναβαθμιστούν από τα LPNs και τον αριθμό των LPN που δίνονται ως εκ των προτέρων πληροφορίες.
- B. Καθορίστε την ποιότητα των καναλιών μεταξύ όλων των LPN.
- C. Βρείτε τους κόμβους fog σύμφωνα με την ποιότητα του καναλιού.
- D. Προσδιορίστε την πιθανότητα σύνδεσης μεταξύ fog κόμβων και των LPNs σύμφωνα με την ποιότητα του καναλιού.

Πραγματοποιείται προσομοίωση για την αξιολόγηση της αποτελεσματικότητας του προτεινόμενου αλγορίθμου. Μέσα σε αυτό το πεδίο, θεωρείται ότι υπάρχουν 8 κόμβοι ομίχλης και 100 LPN που δεν είναι εξειδικευμένοι ως κόμβοι ομίχλης σε ένα μόνο κελί. Σύμφωνα με τον προτεινόμενο αλγόριθμο, κάθε LPN συνδέεται με όλους τους κόμβους ομίχλης ανάλογα με τις συνθήκες του καναλιού. Αυτό το σχήμα συγκρίνεται με το απλούστερο μοντέλο Voronoi[197] με το οποίο κάθε LPN είναι συνδεδεμένο σε έναν μόνο κόμβο fog που έχει την πλησιέστερη απόσταση. Στο επόμενο σχήμα συγκρίνονται τα δύο μοντέλα σε σχέση με το bandwidth και το latency έχοντας υποθέση SNR=5



Σχήμα 2.13. Μοντέλο μηχανικής μάθησης vs μοντέλο Voronoi[195]

Όπως φαίνεται, ο προτεινόμενος αλγόριθμος συσταδοποίησης έχει χαμηλότερο latency κυρίως στο χαμηλό εύρος ζώνης, ενώ ο αλγόριθμος επιτυγχάνει latency 1ms που απαιτούνται για εφαρμογών 5G σε εύρος ζώνης 1 GHz στα 5dB SNR. Η αύξηση του SNR βελτιώνει το latency. Είναι σαφές ότι ο χρόνος latency θα διπλασιαστεί αν υπάρχει ACK για τα μεταδιδόμενα πακέτα (σύστημα κλειστού βρόχου) καθώς εδώ έχουμε υπόθεση σύστημα ανοικτού βρόχου, ενώ εάν απαιτηθεί μια επαναμετάδοση πακέτου άρα και επιπλέον ACK το latency θα αυξηθεί περισσότερο. Ο χρόνος του latency μειώνεται με το εύρος ζώνης ενώ από ένα σημείο και μετά υπάρχει κορεσμός και δεν μειώνεται περαιτέρω.[195] Για την επίτευξη λοιπόν χρόνου latency 1ms πιθανόν να πρέπει να υιοθετηθεί ένα σύστημα ανοικτού βρόχου με πιθανό πρόβλημα αξιοπιστίας ή να μελετηθούν άλλες τεχνικές να πρέπει να μελετηθούν όπως Automatic repeat request (AQR)[198], hybrid automatic repeat request (HARQ)[199], Diversity Coding-Network Coding (DCNC) [200][201]

2.6 Προσαρμοστική διαμόρφωση και κωδικοποίηση με μηχανική μάθηση

Η προσαρμοστική διαμόρφωση και κωδικοποίηση (Adaptive modulation and coding-AMC) έχουν χρησιμοποιηθεί ευρέως σε ασύρματα συστήματα επικοινωνιών προκειμένου να προσαρμοστούν στις παραλλαγές των συνθηκών καναλιού. Τα σχήματα AMC μπορούν να βοηθήσουν τα συστήματα επικοινωνιών να επιτύχουν υψηλότερη απόδοση φάσματος και καλύτερη αντιστάθμιση μεταξύ του ρυθμού δεδομένων και της αξιοπιστίας του καναλιού σε πραγματικό χρόνο. Δεδομένου ότι το ασύρματο κανάλι είναι αρκετά χρονο-μεταβλητό, τα περισσότερα πρότυπα για ασύρματη επικοινωνία (IEEE 802.11), αλλά και οι διαφορετικές τεχνικές (MIMO, OFDM) έχουν υιοθετήσει σχήματα AMC. Όταν τα σχήματα AMC συνδυάζονται με MIMO και OFDM, η μοντελοποίηση του κοινού αποτελέσματος της διαμόρφωσης OFDM, της συνελκτικής κωδικοποίησης και της επεξεργασίας MIMO είναι ιδιαίτερα δύσκολη λόγω των ακόλουθων ζητημάτων:

Πρώτον, σε ένα πρακτικό σύστημα, υπάρχουν διάφορες ατέλειες, όπως η μη γραμμική παραμόρφωση των ενισχυτών ισχύος, το σφάλμα κβαντοποίησης των αναλογικών-ψηφιακών μετατροπών (quantization error of the analog-digital converters -ADCs) καθώς και μη Gaussian θόρυβος.

Δεύτερον, στο σχεδιασμό των συστημάτων επικοινωνίας, είναι σύνηθες, να χωριστεί η επεξεργασία σήματος σε πολλαπλά ανεξάρτητα μπλοκ, με το καθένα να εκτελεί μια συγκεκριμένη και απομονωμένη λειτουργία. Κατά συνέπεια, η από κοινού βελτιστοποίηση αυτών των στοιχείων οδηγεί σε υπολογιστικά πολύπλοκα συστήματα, και κάνει δύσκολη την ακριβή μοντελοποίηση ενός συστήματος τηλεπικοινωνιών. Αυτό έχει ως αποτέλεσμα οι τεχνικές που χρησιμοποιούνται να είναι προσεγγιστικές και να έχουν μεγάλο υπολογιστικό κόστος. Προκειμένου λοιπόν να εξοικονομηθεί το κόστος υπολογισμού καθώς και να αποφευχθούν μη μη πραγματικές προσεγγίσεις, όπως η προσέγγιση Gauss για μη-Gaussian κατανομές, είναι επιθυμητή μια προσέγγιση για να καταστεί δυνατή η λειτουργία των συστημάτων AMC με μαθηματικά μοντέλα που χρειάζονται όσο λιγότερες υποθέσεις είναι δυνατόν. Η μηχανική μάθηση έχει τη δυνατότητα να βελτιστοποιήσει τα από κοινού μπλοκ χρησιμοποιώντας ένα ενοποιημένο μη γραμμικό πλαίσιο. Ως εκ τούτου, μπορεί να χρησιμεύσει ως καλός υποψήφιος για τη βελτιστοποίηση των ασύρματων συστημάτων που υποστηρίζονται AMC μέσω ασύρματων καναλιών. Έτσι παρουσιάζουμε δύο τεχνικές μάθησης, με επίβλεψη και με ενίσχυση.

Οι προσεγγίσεις της μάθησης με επίβλεψη έχουν τα ακόλουθα πλεονεκτήματα ώστε να διευκολύνουν τις εφαρμογές τους στα συστήματα AMC. Πρώτον, κατά τη λειτουργία του δικτύου, τα δεδομένα εκπαίδευσης μπορούν να συλλεχθούν online για την ενημέρωση του συνόλου δεδομένων. Δεύτερον, υπάρχουν άφθονα σύνολα δεδομένων εκπαίδευσης, συμπεριλαμβανομένων ελέγχων CRC (cyclic redundancy checks), πληροφοριών κατάστασης καναλιού (CSI-channel state information), σχήματα διαμόρφωσης και κωδικοποίησης (MCS) κ.λπ. Αυτά τα δεδομένα μπορούν να χρησιμοποιηθούν για την ενημέρωση του σετ εκπαίδευσης για εκμάθηση AMC σε ένα συγκεκριμένο ασύρματο δίκτυο, ή για επαναχρησιμοποίηση για

εκπαίδευση σε άλλα δίκτυα. Ένα μειονέκτημα των προσεγγίσεων μάθησης με επίβλεψη είναι ότι τα δείγματα δεδομένων που λαμβάνονται δεν μπορούν να αναπαραστήσουν με ακρίβεια όλες τις καταστάσεις που ενδέχεται να αντιμετωπίσουν τα σήματα, συμπεριλαμβανομένων των συνθηκών του ασύρματου καναλιού που αλλάζουν κάθε χρονική στιγμή, της μη γραμμικής συμπεριφοράς του ενισχυτή, των θορύβων και των παρεμβολών που δεν έχουν Gaussian κατανομή, κ.λπ. Ως εκ τούτου, οι προσεγγίσεις μπορεί να είναι ανέφικτες για online μάθηση, όπου το σύνολο χαρακτηριστικών μπορεί να είναι ασυνεπές και η συλλογή μεγάλου αριθμού δειγμάτων εκπαίδευσης θα μπορούσε να είναι μη πρακτική. Αντίθετα, οι μέθοδοι μάθησης με ενίσχυση καθώς και οι διαδικασίες απόφασης Markov είναι ικανές για να μάθουν άμεσα από το περιβάλλον, το οποίο παρέχει έτσι λύσεις για την online μάθηση που απαιτείται από τα συστήματα επικοινωνιών, σε αντίθεση με την μάθηση με επίβλεψη όπου χρειάζεται δεδομένα ώστε να μάθει. Ένα βασικό μειονέκτημα της μάθησης με ενίσχυση για τεχνικές AMC είναι ότι ο χρόνος αλληλεπίδρασης με το περιβάλλον μπορεί να είναι πολύ μεγάλος για να επιλέξει το κατάλληλο σχήμα. Αυτό είναι σαφώς ανεπιθύμητο για λειτουργία σε ασύρματα κανάλια, και προκειμένου να μετριαστεί αυτό το πρόβλημα, μερική offline εκπαίδευση μπορεί να απαιτείται. Στη μάθηση με επίβλεψη, οι πιο ευρέως χρησιμοποιούμενοι αλγόριθμοι μάθησης περιλαμβάνουν, μηχανές διανυσματικής υποστήριξης (SVM), δέντρα αποφάσεων, k-πλησιέστερο γείτονα (K-NN), και νευρωνικά δίκτυα. Μπορούν όλοι να χρησιμοποιηθούν ως ταξινομητές μετά από κατάλληλη εκπαίδευση. Προκειμένου να εφαρμόσουμε μηχανική μάθηση σε AMC θα παρουσιάσουμε μόνο τους k-NN και SVM. Για την μάθηση με ενίσχυση, θα αναφερθούμε στον αλγόριθμο Q-learning.

MCS _i	N _{ss}	Modulation type	Coding rate	Date rate (Mbps)
0	1	BPSK	1/2	6.5
1	1	QPSK	1/2	13
2	1	QPSK	3/4	19.5
3	1	16QAM	1/2	26
4	1	16QAM	3/4	39
5	1	64QAM	2/3	52
6	1	64QAM	3/4	58.5
7	1	64QAM	5/6	65
8	2	BPSK	1/2	13
9	2	QPSK	1/2	26
10	2	QPSK	3/4	39
11	2	16QAM	1/2	52
12	2	16QAM	3/4	78
13	2	64QAM	2/3	104
14	2	64QAM	3/4	117
15	2	64QAM	5/6	130

Πίνακας 2.1. διαμόρφωσης και κωδικοποίησης ορισμένο από το πρότυπο 802.11n

2.6.1 k-πλησιέστεροι γείτονες για AMC

Με στόχο την μεγιστοποίηση του throughput δεδομένου ενός PER (packet error rate), η προσαρμογή ζεύξης που βασίζεται σε AMC απαιτείται για την μέτρηση ενός συγκεκριμένου συνόλου χαρακτηριστικών ώστε να προσδιοριστούν οι σχεδόν βέλτιστες παράμετροι AMC όπως διαμόρφωση και ρυθμός κωδικοποίησης. Το PER επιλέγεται συχνά ως μέτρηση της απόδοσης για τον προσδιορισμό ενός δείκτη MCS, ο οποίος μπορεί να διατυπωθεί ως: [89]

$$i^* = \underset{i}{\operatorname{argmax}} \{ R_i : PER_i \leq PER_{target} \}$$

όπου i^* είναι ο σχεδόν βέλτιστος δείκτης MCS, που προσδιορίζεται με την επίλυση αυτού του προβλήματος, και ο δείκτης i που αντιστοιχεί στο MCS $_i$ καθορίζεται από τον πίνακα 1. Το R_i είναι το throughput όταν χρησιμοποιείται το MCS $_i$, το PER $_i$ είναι το αντίστοιχο PER και το PER $_{target}$ είναι ο περιορισμός στο PER. Επομένως, εάν το PER $_i$ είναι το μεγαλύτερο PER από μικρότερο ή ίσο PER $_{target}$ για μια συγκεκριμένη κατάσταση καναλιού, τότε το MCS $_i$ επιλέγεται για μετάδοση δεδομένων.

Όταν το k-NN χρησιμοποιείται για AMC, το προηγούμενο πρόβλημα βελτιστοποίησης μετατρέπεται σε πρόβλημα ταξινόμησης το οποίο στην συνέχεια επιλύεται χρησιμοποιώντας τον αλγόριθμο K-NN. Ο αλγόριθμος k-NN είναι ένας μη παραμετρικός αλγόριθμος μηχανικής μάθησης με επίβλεψη που μπορεί να χρησιμοποιείται τόσο για την ταξινόμηση όσο και για την παλινδρόμηση.

➤ Αλγόριθμος K-NN για AMC

Όπως αναφέρθηκε, στις ασύρματες επικοινωνίες το AMC είναι ένα πρόβλημα ταξινόμησης. Προκειμένου να πραγματοποιήσουμε ταξινόμηση για ένα επιθυμητό MCS, ένα σύνολο χαρακτηριστικών εκπαίδευσης πρέπει αρχικά να εξαχθεί. Υποθέτουμε ότι υπάρχουν W ξεχωριστές παρατηρήσεις στο σύνολο εκπαίδευσης, που εκφράζονται ως $W = \{0, 1, \dots, w-1\}$. Στη συνέχεια μετά την εκπαίδευση του συστήματος K-NN κάθε παρατήρηση $\omega \in W$ ανατίθεται σε μια κλάση. Πιο συγκεκριμένα, έστω ότι Z_ω αντιπροσωπεύει το σύνολο χαρακτηριστικών που αποκτήθηκε από την εκπαίδευση, χρησιμοποιώντας την παρατήρηση του W που μπορεί να εκφραστεί ως: $\{Z_\omega\} \Rightarrow \{i(\omega)\}$, όπου $i(\omega)$ είναι η κλάση MCS $_i$ από την παρατήρηση του ω . Έτσι, μετά την εκπαίδευση μπορεί να ληφθεί ένας πίνακας αντιστοιχίας μεταξύ του συνόλου χαρακτηριστικών και των αντίστοιχων MCS. Με τη χρήση αυτού του πίνακα, μόλις ένα νέο Z_ω εμφανιστεί ο αλγόριθμος K-NN μπορεί να καθορίσει ένα αντίστοιχο σχήμα MCS για την επίτευξη μετάδοσης AMC. Αξίζει να σημειωθεί ότι σε πρακτικές εφαρμογές ταξινόμησης AMC, ο αλγόριθμος k-NN έχει βελτιωθεί ώστε να βελτιώσει την αποτελεσματικότητα της εκπαίδευσης.

Ο επόμενος αλγόριθμος συνοψίζει τον αλγόριθμο k-NN για AMC χρησιμοποιώντας ένα

χαρακτηριστικών q προς ταξινόμηση. Σε αυτόν τον αλγόριθμο, k πλησιέστεροι γείτονες ω_a ($a \in \{1, 2, \dots, k\}$) παρατίθενται πρώτα. Στη συνέχεια, οι τάξεις που εμφανίζονται πιο συχνά μεταξύ $\{\omega_a\}$ επιλέγονται. Τέλος, η τάξη που έχει το μικρότερο i , το οποίο οδηγεί στην επιλογή του χαμηλότερου throughput R_i , επιλέγεται προκειμένου να διασφαλιστεί το απαιτούμενο PER.[89]

Algorithm 1 k -NN algorithm for AMC (Daniels et al., 2008)

Require: $n_{i(\omega)} \leftarrow 0 \forall i(\omega) \in \mathcal{I}$

1: **for** $a = 1 \rightarrow k$ **do**

2: $\omega_a \leftarrow \arg \min\{d(z_\omega, \mathbf{q}) : \omega \notin \{\omega_1, \dots, \omega_{a-1}\}\}$

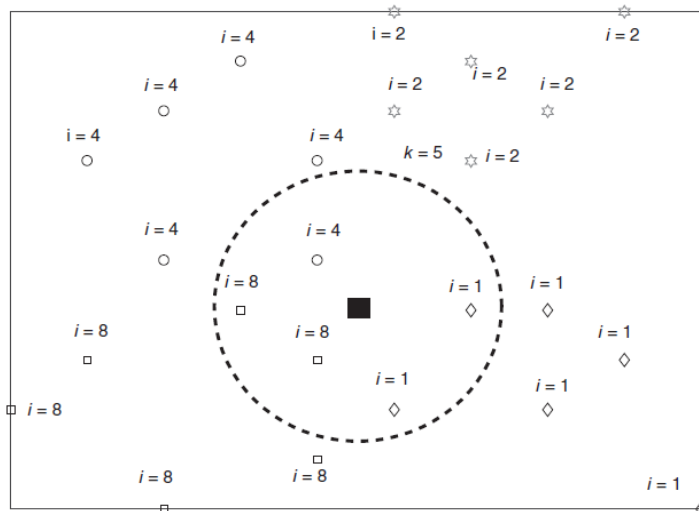
3: $n_{i(\omega)} \leftarrow n_{i(\omega)} + 1$

4: **end for**

5: $modeset = \arg \max_i \{n_i\}$

6: **return** $\min_i \{\arg \min\{R_i : i \in modeset\}\}$

Ένα παράδειγμα απεικονίζεται στο επόμενο σχήμα όπου θεωρείται ότι το $k = 5$.



Σχήμα 2.14. Απεικόνιση της επιλογής κατηγορίας MCS σε έναν αλγόριθμο 5-NN [89]

Σύμφωνα με τον προηγούμενο αλγόριθμο, οι πρώτοι πέντε πλησιέστεροι γείτονες αναγνωρίζονται, όπως φαίνεται από τα πέντε σημεία, μαζί με τις ετικέτες τους που βρίσκονται μέσα στον κύκλο. Στη συνέχεια, οι τάξεις που εμφανίζονται πιο συχνά μεταξύ του νέου χαρακτηριστικού q προς ταξινόμηση αναγνωρίζονται. Σε αυτό το παράδειγμα, το χαρακτηριστικό προς ταξινόμηση αντιπροσωπεύεται από το μαύρο τετράγωνο. Όπως φαίνεται από τις κλάσεις στον κύκλο, υπάρχουν δύο γείτονες με MCS_1 ($i = 1$), δύο γείτονες με MCS_8 ($i = 8$) και ένας γείτονας με MCS_4 ($i = 4$). Ως εκ τούτου, οι επιλεγμένες τάξεις είναι $i = 1$ και $i = 8$. Τέλος, επιλέγεται η κλάση που χαρακτηρίζεται από το μικρότερο i η οποία έχει και το μικρότερο

throughput R_i , ώστε να εγγυηθεί το απαραίτητο PER. Σε αυτό το παράδειγμα το throughput είναι $R_i = R_8$ όπως φαίνεται και από τον προηγούμενο πίνακα 1 αλλά αυτό δεν είναι απαραίτητα αληθές. Επιπλέον, όπως φαίνεται στον πίνακα, υπάρχουν τάξεις $i' < i$, αλλά το throughput είναι $R_i < R_{i'}$. Σε αυτή την περίπτωση, αν και οι κλάσεις $\{i, i'\}$ επιλέγονται από το πρώτο βήμα, η κλάση i επιλέγεται ως η τελική. Συνεπώς, δεδομένου ενός επιλεγμένου συνόλου κλάσεων $\{i_1, i_2, \dots, i_n\}$, η τελική τάξη που επιλέγεται είναι η κλάση που έχει τη χαμηλότερη απόδοση. Αυτό συμβαίνει επειδή το χαμηλότερο throughput έχει την υψηλότερη αξιοπιστία. Επομένως, ο προηγούμενος κανόνας επιλογής θα μπορούσε να εγγυηθεί την απαιτούμενη αξιοπιστία.

➤ Ανάλυση απόδοσης συστήματος k-NN-Assisted AMC

Σε ασύρματο σύστημα MIMO-OFDM (802.11n) με N_t κεραιές εκπομπής και N_r κεραιές λήψης, μετά την εφαρμογή του discrete Fourier transform (DFT), το σήμα που θα ληφθεί από το n^{th} ($n \in \{1, 2, \dots, N\}$) subcarrier μπορεί να αναπαρασταθεί ως :

$$y_n = \sqrt{E_s} H_n x_n + v_n,$$

Όπου N είναι το μέγεθος του DFT και επίσης ο αριθμός των subcarriers, x_n είναι το διάνυσμα με τα μεταδιδόμενα σύμβολα, $V_n \sim \text{CN}(0, N_o I)$ είναι το διάνυσμα του Gaussian θορύβου (complex Gaussian), όπου κάθε στοιχείο έχει μέση τιμή 0 και διακύμανση N_o , H_n αντιπροσωπεύει τον πίνακα του καναλιού

$N_r \times N_t$ του n^{th} subcarrier, E_s είναι η αναμενόμενη συνολική ενέργεια του λαμβανόμενου σήματος, επομένως $E[|x_n|^2] = 1$ για κάθε n . Στην ανάλυση απόδοσης, υποθέτουμε ότι το ασύρματο κανάλι H_n υφίσταται σχεδόν στατικό block fading, που σημαίνει ότι οι καταστάσεις καναλιών παραμένουν σταθερές σε όλα τα σύμβολα OFDM μέσα σε ένα πακέτο, αλλά είναι ανεξάρτητες για διαφορετικά πακέτα. Υποθέτουμε επίσης :

- Σταθερό μήκος πακέτου: Όλα τα πακέτα έχουν οριστεί σε 128 byte. Ως εκ τούτου, για ένα δεδομένο ποσοστό σφάλματος bit, όλα τα πακέτα έχουν το ίδιο αναμενόμενο PER.
- Τέλειος συγχρονισμός: Όλα τα πακέτα συγχρονίζονται τέλεια τόσο στο πεδίο χρόνου όσο και στο πεδίο της συχνότητας. Ως εκ τούτου, οποιαδήποτε μετατόπιση χρόνου και συχνότητας μπορεί να αγνοηθεί.
- Εκτίμηση: Υποθέτουμε ότι η ισχύς θορύβου και η εκτίμηση καναλιών υπολογίζονται στους δέκτες.
- Linear equalization: Ένας γραμμικός zero-forcing (ZF) ισοσταθμιστής χρησιμοποιείται στο δέκτη για την καταπολέμηση της διασυμβολικής παρεμβολής (inter-symbol interference -ISI).

Όπως περιγράψαμε στον προηγούμενο αλγόριθμο ένα σύνολο χαρακτηριστικών πρέπει πρώτα να εξαχθεί, και στη συνέχεια χρησιμοποιείται για τον υπολογισμό της απόστασης μεταξύ του νέου χαρακτηριστικού που θέλουμε να ταξινομήσουμε και των δεδομένων εκπαίδευσης ώστε να μπορέσουμε να καθορίσουμε τους k -πλησιέστερους γείτονες. Για ένα σύστημα AMC

MIMO-OFDM ειδικά στον δέκτη μετά την μετα-επεξεργασία μια μετρήσιμη ποιότητας σύνδεσης (link quality metric-QM) για την περιγραφή της απόδοσης της ζεύξης μπορεί να αξιολογηθεί με βάση τα H_n , E_s και N_0 . Το LQM θεωρείται ως το σύνολο χαρακτηριστικών που εξάγονται από H_n , E_s και N_0 . Όταν χρησιμοποιείται η μετα-επεξεργασία ZF, για κάθε μια χωρική ροή (spatial stream) $a \in \{1, \dots, N_s\}$, όπου N_s είναι ο αριθμός των spatial stream και $N_s \leq \min(N_r, N_t)$, το SNR του υποφέροντος $n \in \{1, \dots, N\}$ δίνεται από :

$$\gamma[a, n] = \frac{E_s}{N_0 \sum_{a'=1}^{N_s} |[\mathbf{G}_{ZF}[n]]_{a,a'}|^2},$$

όπου $\mathbf{G}_{ZF} = (\mathbf{H}_n)^+ = (\mathbf{H}_n^H \mathbf{H}_n)^{-1} \mathbf{H}_n^H$, όπου $(\cdot)^+$ και $(\cdot)^{-1}$ είναι ανάστροφος ψευδοαντίστροφος Ερμιτιανός και τον αντίστροφο αντίστοιχα, ενώ η διάσταση του SNR για κάθε subcarrier ισούται με τον αριθμό του spatial stream του. Σύμφωνα με την προηγούμενη εξίσωση για κάθε subcarrier μπορούμε να λάβουμε ένα χαρακτηριστικό από το SNR του κάθε spatial stream. Για να απλοποιήσουμε το AMC σύστημα θα υποθέσουμε ότι κάθε πακέτο παράγει μόνο ένα νέο χαρακτηριστικό i αντί για κάθε subcarrier, καθώς επίσης και ότι κάθε subcarrier του πακέτου θα έχει το ίδιο MCS ανάλογα το χαρακτηριστικό ενώ θα αναλύσουμε ξεχωριστά τις περιπτώσεις επίπεδων και συχνοεπιλεκτικών διαλείψεων καθώς είναι διαφορετικός ο τρόπος που εξάγουμε το χαρακτηριστικό προς ταξινόμηση.

Για ζεύξης με επίπεδες διαλείψεις, όλα τα subcarrier μοιράζονται το ίδιο κανάλι. Επομένως, ο μέσος όρος του SNR, μετά την επεξεργασία όλων των subcarrier, είναι διαθέσιμος για το χαρακτηριστικό προς ταξινόμηση. Μπορούμε έτσι να αποκτήσουμε ένα μονοδιάστατο σύνολο χαρακτηριστικών για MCS 0-7 με ένα spatial stream και ένα δισδιάστατο σύνολο χαρακτηριστικών για το MCS 8-15 με δύο spatial steam, σύμφωνα με την προηγούμενη εξίσωση. Δεδομένου ότι υπάρχουν δύο σύνολα δυνατοτήτων με διαφορετικές διαστάσεις, εκτελούμε τον αλγόριθμο 1 σε κάθε σύνολο δυνατοτήτων. Μόλις λάβουμε το προτεινόμενο MCS για κάθε σύνολο δυνατοτήτων, το MCS με την υψηλότερη τιμή επιλέγεται.

Για ζεύξης με συχνο-επιλεκτική διαλείψεις, ο μέσος SNR μετά την επεξεργασία δεν μπορεί να αντικατοπτρίζει αποτελεσματικά τις παραλλαγές των συνθηκών καναλιού, οι οποίες καθορίζονται μοναδικά από τον πίνακα καναλιού που περιέχει την ενέργεια σήματος και τη διακύμανση θορύβου. Σε αυτήν την περίπτωση, ο χώρος χαρακτηριστικών του σετ εκπαίδευσης μπορεί να περιλαμβάνει H_n , E_s και N_0 , που οδηγεί σε υψηλότερη διάσταση του χώρου των χαρακτηριστικών. Για ένα σύστημα μηχανικής μάθησης με επίβλεψη οι προηγούμενες διαστάσεις απαιτούν εξαιρετικά μεγάλα εκπαιδευτικά σετ ακόμη και για τυπικές τιμές. Προκειμένου να μειωθεί η διάσταση του χώρου των χαρακτηριστικών, προτείνετε μια μέθοδος subcarrier ordering. Σε αυτήν τη μέθοδο, υπό την προϋπόθεση ότι η θέση του subcarrier δεν επηρεάζει τη συνολική απόδοση του πακέτου, μπορεί να προσδιοριστεί η απόδοση του πακέτου με ανάλυση της κατανομής SNR μετά την επεξεργασία ανά subcarrier. Σύμφωνα με αυτή την μέθοδο ορίζουμε $SNR_s^{ZF}[n]$ ως το SNR μετά την επεξεργασία για το spatial stream a , για το subcarrier $n \in \{1, 2, \dots, N_{ds}\}$ όπου N_{ds} είναι ο αριθμός των δεδομένων (data bearing) subcarrier, και $N_{ds} = 52$ για 200 MHz bandwidth καναλιού. Όλα τα SNR μπορούν να συνθέσουν ένα σύνολο SNR, το οποίο εκφράζεται ως:

$$\left\{ \{SNR_a^{ZF}[n]\}_{a=1}^{N_s} \right\}_{n=1}^N$$

Στη συνέχεια ορίζουμε το $SNR_{(n)}^{ZF}$ να είναι το n-οστό μικρότερο στοιχείο του συνόλου, όπου $n \in \{1, 2, \dots, N_{ds}N\}$. Από παρατηρήσεις στα κανάλια IEEE 802.11n, αυτή η μέθοδος προτείνει ότι το SNR_s^{ZF} για μερικές τιμές n καθορίζει συχνά την απόδοση των πακέτων και μπορεί να λειτουργήσει ως το χαρακτηριστικό. Προκειμένου να ληφθούν οι κατάλληλες τιμές n, χρησιμοποιούνται εκτεταμένες αναζητήσεις σε υπολογιστές και ένα χαρακτηριστικό τεσσάρων διαστάσεων με $n = 1, 2, 6, 26$ είναι απαραίτητο για το σύστημα MCS 0-7 και MCE 8-15 που βασίζονται σε συστήματα IEEE 802.11n, για συχνο-επιλεκτικά κανάλια, τα οποία δίνονται από:

$$\mathbf{q} = [SNR_{(1)}^{ZF}, SNR_{(2)}^{ZF}, SNR_{(6)}^{ZF}, SNR_{(26)}^{ZF}]^T$$

Στο τέλος, ο αλγόριθμος εκτελείται στο σύνολο χαρακτηριστικών αποτελούμενος από αυτά των τεσσάρων διαστάσεων χαρακτηριστικών και το προτεινόμενο MCS επιλέγεται.

2.6.2 SVM για AMC

Η τεχνική k-NN για συστήματα που AMC μπορεί να βοηθήσει αποτελεσματικά το σύστημα να προσαρμοστεί στις παραλλαγές καναλιών και να βελτιώσει την απόδοση του συστήματος. Ωστόσο, παρόμοια με άλλους συμβατικούς αλγόριθμους μηχανικής μάθησης με επίβλεψη, αυτή η μέθοδος δεν μπορεί να κατασκευάσει όρια μεταξύ των κλάσεων, ενώ η μέθοδος k-NN έχει μεγάλη υπολογιστική πολυπλοκότητα και απαιτεί μια μεγάλη βάση δεδομένων για εκπαίδευση εκτός σύνδεσης, ως αποτέλεσμα να μην ενδείκνυται για αποτελεσματική χρήση σε AMC συστήματα. Οι συμβατικές μέθοδοι χρησιμοποιούν μια offline βάση δεδομένων με το PER (packet error rate) να υποβάλλεται σε διαφορετικές παρατηρήσεις σταθερού καναλιού. Ωστόσο, η offline βάση δεδομένων είναι συνήθως πολύ μεγάλη για την εφαρμογή σε online AMC. Επομένως, θα είναι επιθυμητό να συλλέξουμε τα δεδομένα εκπαίδευσης online αντί να κατασκευάσουμε μια μεγάλη offline βάση δεδομένων. Δεδομένου λοιπόν, ότι απαιτείται μεγάλος όγκος δεδομένων PER που θα πρέπει να συλλεχθεί για κάθε κανάλι ώστε να μειωθεί η διακύμανση του PER και να αυξηθεί η ακρίβεια εκτίμησης ένας πιο αποτελεσματικός τρόπος πρέπει να βρεθεί. Μια καλή εναλλακτική μέθοδος προς αυτήν την κατεύθυνση είναι μια online AMC μέθοδος η οποία θα μπορούσε να πετύχει ένα γρήγορο AMC χρησιμοποιώντας έναν αλγόριθμο support vector machine.[89][90]

➤ Αλγόριθμος SVM

Ο αλγόριθμος SVM χρησιμοποιεί μία μόνο μέτρηση του πλαισίου, όπως η επιτυχία/αποτυχία, καθώς και την μέτρηση της κατάστασης του καναλιού για την εκπαίδευση των ταξινομητών SVM. Σε σύγκριση με άλλους γρήγορους online αλγόριθμους AMC που βασίζονται στην ταξινόμηση του πλησιέστερου γείτονα (K-nn), ο γρήγορος online αλγόριθμος AMC που βασίζεται σε SVM μπορεί να επιτύχει την ίδια απόδοση με λιγότερες απαιτήσεις μνήμης και υπολογιστικής ισχύς, ενώ έρευνες έχουν δείξει ότι η πολυπλοκότητα τους είναι χαμηλότερης τάξης, ενώ επιπλέον οι αλγόριθμοι SVM έχουν βελτιώσει περαιτέρω την απόδοση του κλοντας χρήση συναρτήσεων πυρήνα για την κατασκευή μη γραμμικού ορίου επιλέγοντας κατάλληλο εκπαιδευτικό σετ αλλά και σύνολο χαρακτηριστικών για εξισορρόπηση της απόδοσης και της πολυπλοκότητας του. Στην συνέχεια θα δούμε πως η τεχνική SVM μπορεί να χρησιμοποιηθεί για την εκμάθηση των συνθηκών του καναλιού, έχοντας ως σκοπό να παρέχετε μια αξιόπιστη λύση για συστήματα AMC. Για τη δημιουργία δυαδικών online ταξινομητών AMC, το SVM μπορεί να διακρίνει καλύτερα τα όρια περιοχής ακόμα και για μη γραμμικές κλάσες έχοντας καλύτερη πολυπλοκότητα σε σύγκριση με άλλους αλγορίθμους μηχανικής μάθησης με επίβλεψη, και ακόμα από αυτών των νευρωνικών δικτύων.

Ο αλγόριθμος SVM μεγιστοποιεί το περιθώριο μεταξύ των διαφορετικών κλάσεων των δεδομένων εκπαίδευσης, ενώ χαρακτηρίζει κάθε δεδομένο τοποθετώντας του μια ετικέτα $y \in \{+1, -1\}$ ανάλογα με απόσταση από τα διαφορετικά όρια περιθωρίου της κάθε κλάσης, προσπαθεί δηλαδή να φτιάξει το μέγιστο περιθώριο για να προσδιορίσει την περιοχή της κλάσης για το σύνολο των χαρακτηριστικών. Όταν εφαρμόζουμε SVM για AMC, οι περιοχές κλάσης αντιστοιχούν στην επιτυχία ή την αποτυχία των μεταδόσεων, αντιπροσωπεύουν δηλαδή εάν η αποκωδικοποίηση πλαισίου στον δέκτη είναι επιτυχής ή όχι, ενώ συνήθως η κλάση επιτυχίας παίρνει την ετικέτα +1 ενώ η κλάση αποτυχίας της ετικέτα -1. Για το σύνολο των χαρακτηριστικών προς επεξεργασίας $\chi \in R^p$, όπου p είναι η διάσταση του διανύσματος χαρακτηριστικών και R είναι το σύνολο των πραγματικών αριθμών, το περιθώριο SVM δίνεται από την συνάρτηση $h(x) = w^T \Phi(x) + b$, όπου w μπορεί να οριστεί ως ένα γραμμικό διάνυσμα για εκπαίδευση, $\Phi(x)$ είναι ο μετασχηματισμός των χαρακτηριστικών SVM που εκτελείται στο σύνολο των χαρακτηριστικών για τη δημιουργία του SVM συνόλου και το b είναι το περιθώριο bias. Το αποτέλεσμα της συνάρτησης $h(x)$ λειτουργεί ως είσοδος στην σύνθετη συνάρτηση $g(h(x))$ με $g(z)=1$ όταν $z>0$ και $g(z)=-1$ όταν $z<0$.

Κάθε SVM παρέχει τη δυαδική ταξινόμηση, έτσι μ SVMs χρειάζονται για να διακρίνουν μ κλάσεις κωδικοποίησης και διαμόρφωσης. Ο ταξινομητής m th χρησιμοποιείται για την επιλογή του κατάλληλου μεταξύ των m κλάσεων. Όταν ο περιορισμός $PER(m, x) \leq F$ δεν ικανοποιείται, όπου F είναι το κατώφλι του PER, καμία κλάση δεν επιλέγεται. Όταν ένα πλαίσιο κωδικοποιείται με MCS το i επιλέγεται από τον ταξινομητή ως επιτυχημένη μετάδοση, ο ταξινομητής ενισχύεται για να επιλέξει MCS i . Από την άλλη πλευρά, όταν ένα πλαίσιο μετάδοσης χαρακτηριστεί ως αποτυχία, ο ταξινομητής ενισχύεται ώστε να μην επιλέγει τάξεις MCS. Έτσι, το κατάλληλο MCS για την μετάδοση πλαισίου μπορεί να επιλεγεί από μ κλάσεις MCS μετά την εφαρμογή των ταξινομητών μ (one vs none).

Η προηγούμενη μέθοδος ταξινόμηση δεν μπορεί να αντιμετωπίσει τον περιορισμό PER. Για να ταιριάζουν με την αλλαγή του PER τα δεδομένα εκπαίδευσης, το σύνολο των δεδομένων

εκπαίδευσης σταθμίζεται εξίσου. Οι διαφορετικές συναρτήσεις (αν έχουμε ένα σύνολο κλάσεων K , τότε μπορούμε να ορίσουμε ένα σύνολο διαφορετικών συναρτήσεων $K(x)$, μια για κάθε κλάση) με παλινδρόμηση PER αντικαθίστανται, με συναρτήσεις παλινδρόμησης και ορίζονται αντίστοιχα με διαφορετικές κλάσεις MCS έτσι ώστε να υπολογίζεται η εκ των υστέρων πιθανότητα κάθε κλάσεις. Οι συναρτήσεις παλινδρόμησης ορίζονται ως $r_m : R \rightarrow [0, 1]$, η οποία διαφέρει ανάλογα με το m . Ο αλγόριθμος SVM για online AMC χρησιμοποιείται για να ταξινομήσει διαφορετικά χαρακτηριστικά x με ένα πλαίσιο ταξινόμησης δύο σταδίων (one-versus-none ταξινομητές και συναρτήσεις παλινδρόμησης) μπορεί να οριστεί ως: [89]

1. Υπολογίστε το $h_m(x)$ σύμφωνα με τα δεδομένα εκπαίδευσης :
2. Υπολογίστε $r_m(h_m(x))$ σύμφωνα με το $h_m(x)$ και τα δεδομένα εκπαίδευσης
3. Αντιστοίχισε $r_m(h_m(x)) \forall m \in \{0, 1, \dots, \mu - 1\}$
4. Βρείτε τα βελτιστοποιημένα $m^* = \operatorname{argmin}_m \{(1 - r_m(h_m(x)))/T : r_m(h_m(x)) \leq F\}$
5. Αν $m^* = 1$, $m^* \leftarrow \operatorname{argmin}_{m \in m^*}$
6. Αν το m^* δεν βρεθεί επέλεξε τυχαία ένα.

όπου T_m είναι ο χρόνος που απαιτείται για τη μετάδοση των bit δεδομένων με MCS. Τα βημάτων 5 και 6 εγγυώνται ότι επιλέγεται μόνο ένα βέλτιστο m for για τη μετάδοση. Τα βήματα 3-6 χρησιμοποιούνται για την εύρεση του καταλληλότερου MCS για μια άγνωστη δομή καναλιού. Από το βήμα 4 μπορούμε να συμπεράνουμε το $r_m(h_m(x))$ παρέχει μια εκτίμηση του $PER(m, x)$. Το n - στοιχείο του m - ταξινομητή είναι $x_{m,n} \forall n \in \{0, 1, \dots, N - 1\}$. Η βελτιστοποίηση SVM γίνεται με τη βοήθεια καθενός από τους μ δυαδικούς ταξινομητές. Στον αλγόριθμο SVM το γεωμετρικό περιθώριο για κάθε x_n ορίζεται ως :

$$\gamma_n = y_n \left(\left(\frac{\mathbf{w}^T}{\|\mathbf{w}\|} \right) \phi(x_n) + \frac{b}{\|\mathbf{w}\|} \right)$$

Το γεωμετρικό περιθώριο του w, b που αντιστοιχεί στο σύνολο εκπαίδευσης είναι το μικρότερο περιθώριο του x_n , το οποίο μπορεί να οριστεί ως :

$$\gamma = \min_{n=0,1,\dots,N-1} \gamma_n$$

Ο στόχος του αλγορίθμου SVM είναι να μεγιστοποιήσει το γεωμετρικό περιθώριο το οποίο μπορεί να είναι :

$$\min_{\mathbf{w} \in \mathbb{R}^p, b \in \mathbb{R}} \frac{\hat{\gamma}}{\|\mathbf{w}\|},$$

για το οποίο ισχύει

$$y_n(\mathbf{w}^T \phi(x_n) + b) \geq \hat{\gamma}$$

όπου $\gamma = \hat{\gamma}/\|\mathbf{w}\|$ με $\hat{\gamma}$ να είναι το περιθώριο. Λαμβάνοντας υπόψη ότι w και b μπορούν να προστεθούν με αυθαίρετο περιορισμό κλιμάκωσης που σημαίνει ότι το περιθώριο $\hat{\gamma}$ του w και b μπορούν να οριστούν ως 1. Συνεπώς, το προηγούμενο πρόβλημα μπορεί να οριστεί ως

$$\min_{\mathbf{w} \in \mathbb{R}^p, b \in \mathbb{R}} \frac{1}{2} \|\mathbf{w}\|^2 \quad y_n(\mathbf{w}^T \phi(\mathbf{x}_n) + b) \geq 1.$$

Αν υποθέσουμε ότι C είναι το σφάλμα για το σετ χαρακτηριστικών απαιτείται $\xi_n > 0$ για τις περιπτώσεις προβλημάτων που είναι μη διαχωρίσιμες οι κλάσεις, οπότε το πρόβλημα μπορεί να γραφεί ως :

$$\min_{\mathbf{w} \in \mathbb{R}^p, b \in \mathbb{R}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{n=0}^{N-1} \xi_n \quad \begin{aligned} y_n(\mathbf{w}^T \phi(\mathbf{x}_n) + b) &\geq 1 - \xi_n \\ \xi_n &\geq 0. \end{aligned}$$

Ο ταξινομητής one-versus-none θα γίνει μη διαχωρίσιμος όταν το διάνυσμα-χώρος των χαρακτηριστικών δεν μπορούν να οδηγήσουν σε ντετερμινιστικά αποτελέσματα των κλάσεων. Οπότε το C δρα ως παράμετρος για τη βελτιστοποίηση της απόδοσης του ταξινομητή καθώς και να ελαχιστοποιήσει την πιθανότητα επιλογής λάθος MCS.

Για να ξεφύγουμε από τις μη διαχωρίσιμες κλάσεις και να μειώσουμε τις διαστάσεις του χώρου των χαρακτηριστικών χρησιμοποιούμε συναρτήσεις πυρήνα ώστε να μειωθούν οι διαστάσεις. Η συνάρτηση πυρήνα για ταξινομητή SVM μπορεί να οριστεί ως $k(x_n, x_{n'}) : \Phi(x_n)^T \Phi(x_{n'})$.

Επιπλέον ορίζουμε $Q \in \mathbb{R}^{N \times N}$ με :

$$[Q]_{n,n'} := y_n y_{n'} \kappa(\mathbf{x}_n, \mathbf{x}_{n'}),$$

και με τη βοήθεια των συνθηκών Lagrangian dual και Karush-Kuhn-Tucker (KKT), το μέγιστο περιθώριο μπορεί να είναι ,

$$\max_{\alpha \in \mathbb{R}^N} \sum_{n=0}^{N-1} \alpha_n - \frac{1}{2} \sum_{n,n'=0}^{N-1} \alpha_n \alpha_{n'} y_n y_{n'} \mathbf{x}_n^T \mathbf{x}_{n'} \quad \begin{aligned} \sum_{n=0}^{N-1} y_n \alpha_n &= 0 \\ 0 &\leq \alpha_n \leq C \end{aligned}$$

Χρησιμοποιώντας την συνάρτηση πυρήνα ο όρος $\mathbf{x}_n^T \mathbf{x}_{n'}$ μπορεί να ενημερωθεί ως $\Phi(x_n)^T \Phi(x_{n'})$ και η προηγούμενη εξίσωση θα μετατραπεί ως εξής:

$$\max_{\alpha \in \mathbb{R}^N} \mathbf{e}^T \alpha - \frac{1}{2} \alpha^T Q \alpha \quad \begin{aligned} \sum_{n=0}^{N-1} y_n \alpha_n &= 0 \\ 0 &\leq \alpha_n \leq C \end{aligned}$$

όπου \mathbf{e} είναι ένα διάνυσμα με όλα τα στοιχεία να είναι "1", α_n είναι το n-οστό στοιχείο στο διάνυσμα α , ενώ η συνάρτηση συμπερασμάτων γίνεται :

$$h(\mathbf{x}) = \sum_{n=0}^{N-1} y_n \alpha_n \kappa(\mathbf{x}_n, \mathbf{x}_{n'}) + b.$$

Η συνάρτηση πυρήνα χρησιμοποιείται για να γενικεύσει τα όρια μεταξύ των κλάσεων. Οι πιο

συχνά χρησιμοποιούμενες συναρτήσεις πυρήνα είναι οι εξής:

γραμμική: $k_{lin}(x_n, x_{n'}) := x_n^T x_{n'}$

πολυωνυμική: $k_{poly}(x_n, x_{n'}) := (x_n^T x_{n'})^d$, όπου d είναι η διάσταση του πολυωνύμου

Gaussian : $k_{gau}(x_n, x_{n'}) := e^{-\gamma \|x_n - x_{n'}\|^2}$ όπου $\gamma > 0$ ορίζει τα όρια σύμφωνα με την ευκλείδεια απόσταση, και μπορεί να παρέχει διαφορετικά είδη ορίων.

Εάν η μεταβλητή $a_n = 0$, το x_n δεν είναι διάνυσμα υποστήριξης, εάν $0 < a_n < C$, το x_n είναι διάνυσμα υποστήριξης πάνω στο περιθώριο ενώ εάν $a_n < C$ το x_n είναι διάνυσμα υποστήριξης αλλά βρίσκεται μέσα στο περιθώριο.

Στον online αλγόριθμο SVM η συνάρτηση παλινδρόμησης $r_m(h_m(x))$, ($h_m(x)$ η συνάρτηση συμπεράσματος) εφαρμόζεται σε m MCS κλάσεις για την αντιμετώπιση του περιορισμού PER για κάθε ταξινομητή one-versus-none. Η πιθανότητα εξόδου της κανονικής κατανομής μπορεί να εκφραστεί σε σιγμοειδή μορφή ως :

$$\Pr[y = 1|h(\mathbf{x})] = (1 + e^{A_1 h(\mathbf{x})^2 + A_2 h(\mathbf{x}) + A_3})^{-1}$$

όπου A_1, A_2, A_3 είναι σταθερές.

Λαμβάνοντας υπόψη ότι η τετραγωνική συνάρτηση δεν είναι μονότονη, η έξοδος μπορεί να απλοποιηθεί, και οι αντίστοιχες προγενέστερες πιθανότητες μπορούν να περιγραφούν ως :

$$\Pr[y = 1|h(\mathbf{x})] = (1 + e^{B_1 h(\mathbf{x}) + B_2})^{-1} \quad (\text{εξίσωση 1})$$

όπου B_1, B_2 είναι σταθερές.

Η έξοδος του ταξινομητή SVM παρέχει αξιόπιστα και ακριβή αποτελέσματα επιλογής της κατάλληλης κλάσης MCS. Έτσι για κάθε ταξινόμηση one-versus-none, η συνάρτηση παλινδρόμησης για την αντιμετώπιση του περιορισμού PER μπορεί να είναι :

$$r_m(h_m(\mathbf{x})) = (1 + e^{B_{1,m}^* h_m(\mathbf{x}) + B_{2,m}^*})^{-1}$$

όπου $B_{1,m}^*, B_{2,m}^*$ μπορεί να υπολογιστεί από τον αλγόριθμο ταιριάσματος σύμφωνα

με τη εξίσωση 1. Επιπλέον, η μεταβλητή m θα μπορούσε παραλειφθεί, δεδομένου ότι το ταιρίασμα θα μπορούσε να είναι εκτελεστεί με κάθε MCS.

Προκειμένου να επιτευχθούν B_1^*, B_2^* , η συνάρτηση κόστους μπορεί να κατασκευαστεί και να βελτιστοποιηθεί, και τα B_1^*, B_2^* να οριστούν ως :

$$\{B_1^*, B_2^*\} = \arg \min_{B_1, B_2} \left\{ - \sum_{n=0}^{N-1} \lambda_n \log((1 + e^{B_1 h(\mathbf{x}) + B_2})^{-1}) + (1 - \lambda_n) \log(1 - (1 + e^{B_1 h(\mathbf{x}) + B_2})^{-1}) \right\}$$

όπου λ_n ορίζεται ως εξής :

$$\lambda_n = \begin{cases} (N_p + 1)/(N_p + 2) & \text{if } y_n = +1 \\ 1/(N_m + 2) & \text{if } y_n = -1 \end{cases}$$

με $N_p = \sum_{n=0}^{N-1} (1 + y_n)/2$ και $N_m = N - N_p$. Η βελτιστοποίηση για την επιλογή των καλύτερων B_1^* ,

B_2^* μπορεί να εφαρμοστεί αποτελεσματικά από τον αλγόριθμο Newton που μπορεί να συγκλίνει γρήγορα στο βέλτιστη τιμή (Lin et al., 2007).

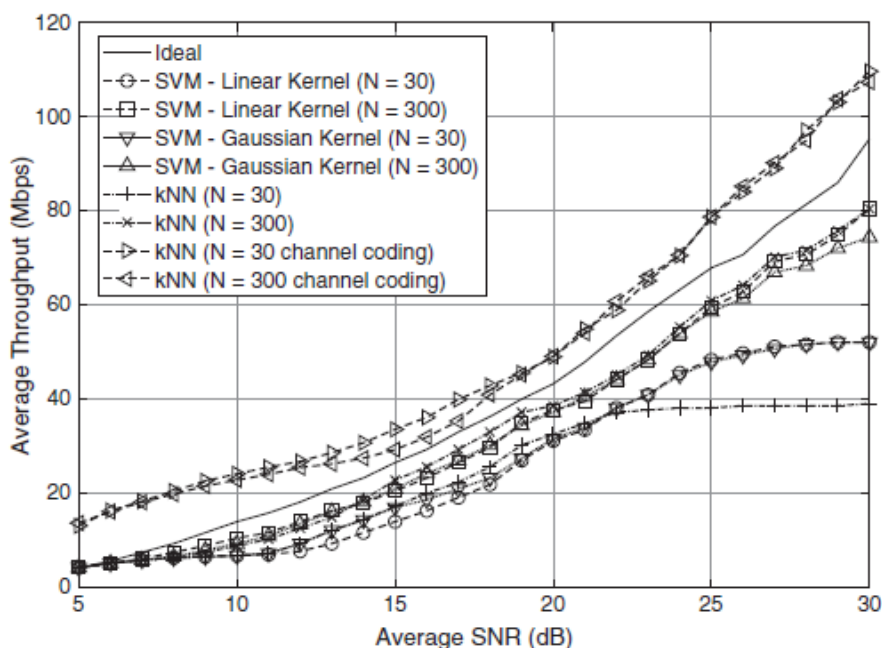
Στον επόμενο πίνακα γίνεται σύγκριση της πολυπλοκότητας των μεθόδων που βασίζονται σε μηχανική μάθηση με επίβλεψη και MCS. Μπορεί να παρατηρηθεί ότι η πολυπλοκότητα του online αλγορίθμου AMC που βασίζεται σε SVM με γραμμικό πυρήνα έχει το χαμηλότερο κόστος επεξεργασίας και το κόστος μνήμης. Δεδομένου ότι τα εκπαιδευτικά δεδομένα που απέχουν πολύ από το περιθώριο πρέπει να διατηρηθούν για τις ακόλουθες βελτιστοποιήσεις, το AMC που βασίζεται στο SVM με πυρήνα Gaussian και το k-NN θα καταναλώσει πολύ περισσότερους πόρους επεξεργασίας.

Algorithm	Processing	Memory(bits)
SVM(linear)	$\mu(p + 2)$ multiplications μ divisions $\mu(p + 4)$ additions μ exponential maps μ -length sort	$\mu b(8p + 4)$
SVM(Gaussian)	$\mu(p + 303)$ multiplications μ divisions $\mu(p + 304)$ additions 2μ exponential maps μ -length sort	$\mu b(300p + 5)$
kNN	$\mu(pN + 1)$ multiplications μ divisions $\mu(2pN + k + 1)$ additions μ N -length sorts μ -length sort	$\mu b(300p + 1)$

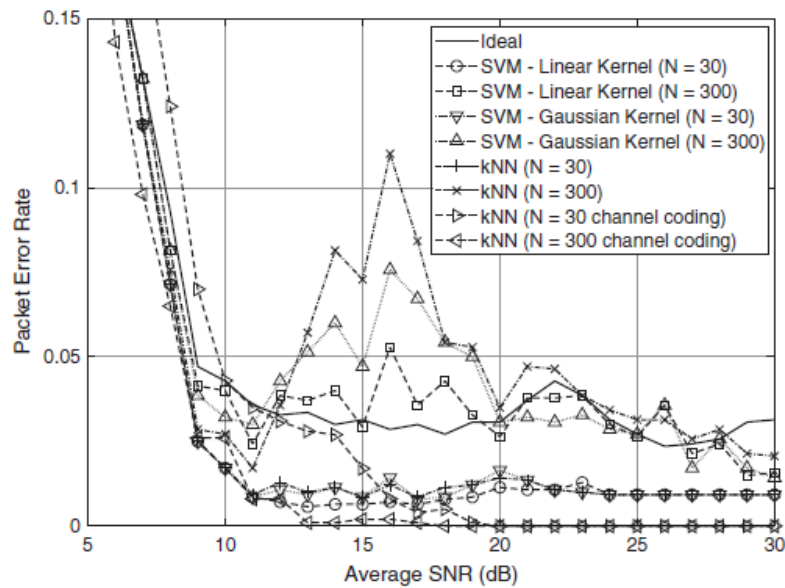
➤ Προσομοίωση και αποτελέσματα

Σε αυτό το σημείο θα παρουσιαστούν αποτελέσματα που έγιναν με τις προαναφερθέντες μεθόδους.

Οι ρυθμίσεις παραμέτρων προσομοίωσης δίνονται ως εξής : το εύρος ζώνης του καναλιού είναι 20MHz, ο αριθμός των κεραιών λήψης και κεραιών μετάδοσης είναι 2, ο αριθμός των κλάσεων MCS είναι 16 (MCS0– MCS15), ο περιορισμός PER είναι 0,1 και το μήκος πλαισίου είναι 128 bytes. Στις προσομοιώσεις, έχει υποθετηθεί ότι ο συγχρονισμός και η εκτίμηση καναλιού στον δέκτη εκτελούνται τέλεια , ενώ για equalizations χρησιμοποιείται ο αλγόριθμος zero-forcing. Ο online αλγόριθμος AMC που βασίζεται σε SVM χρησιμοποιεί γραμμική ή Gaussian συνάρτηση πυρήνα , ενώ στον αλγόριθμο πλησιέστερου γείτονα το $k=10$. Το σύνολο χαρακτηριστικών SNR είναι καλά ταξινομημένα μετά από επεξεργασία , ενώ έχουμε 32000 κανάλια με SNR 0–30 dB. Η παράμετρος μ του ταξινομητή SVM έχει καθοριστεί από εμπειρικά αποτελέσματα εδώ καθορίζουμε $C=10$ για όλους τους πυρήνες και $\gamma=0,005$ για τον Gaussian, ενώ έχει παρατηρηθεί ότι όταν ο αριθμός των παραδειγμάτων εκπαίδευσης ορίζεται σε $N = 60$, ο ταξινομητής μπορεί να προσφέρει ισχυρές επιδόσεις χωρίς να χρειάζεται να αυξησει τον αριθμό του δείγματος εκπαίδευσης. Το επόμενο σχήμα περιγράφει την σχέση μεταξύ του throughput και του μέσου SNR για όλα τα κανάλια ενώ το επόμενο σχήμα δείχνει το PER σε διάφορα SNR.



Σχήμα 2.15. Throughput vs. average SNR για διαφορετικά κανάλια [89]



Σχήμα 2.16. PER vs. SNR για διαφορετικά κανάλια[89]

Από τα δύο παραπάνω σχήματα μπορούμε να συμπεράνουμε ότι ένα σύστημα AMC που βασίζεται σε αλγόριθμο μηχανικής μάθησης με επίβλεψη μπορεί να βοηθήσει το ασύρματο σύστημα επικοινωνίας ώστε να έχει ένα καλό συμβιβασμό μεταξύ υψηλού ρυθμού μετάδοσης και υψηλής αξιοπιστίας χρησιμοποιώντας πληροφορίες που έχουν έρθει από δεδομένα εκπαίδευσης. Όταν το N αυξάνεται, τόσο το throughput όσο και το PER αυξάνονται, ενώ όταν το N είναι μικρό και το SNR μεγάλο ο αλγόριθμος SVM μπορεί να επιτύχει καλύτερο throughput από τον αλγόριθμο K-NN. Ως εκ τούτου, η χρήση του γραμμικού πυρήνα μπορεί να προσφέρει ικανοποιητικές επιδόσεις για το σύστημα και δεν χρειάζεται να σπαταλήσουμε επιπλέον ενέργεια για τη χρήση Gaussian πυρήνα. Αξίζει να σημειωθεί ότι η απόδοση του αλγορίθμου AMC που βασίζεται στον αλγόριθμο πλησιέστερου γείτονα εξαρτάται σε μεγάλο βαθμό από την επιλογή των δεδομένων και το k , καθώς στον αλγόριθμο k-NN, ο k -πλησιέστερος γείτονας καθορίζεται με τον υπολογισμό της απόστασης μεταξύ του αντικειμένου χωρίς ετικέτα (άγνωστα) και όλων των αντικειμένων με ετικέτα (γνωστά). Έτσι, σε σύγκριση με τον αλγόριθμο AMC που βασίζεται σε SVM, η αποδοτικότητα του αλγορίθμου AMC που βασίζεται στον αλγόριθμο k-NN είναι χειρότερη.

2.6.3 Μάθηση με ενίσχυση για AMC

Η μάθηση με ενίσχυση σε αντίθεση με την μάθηση με επίβλεψη μπορεί να αλληλεπιδρά άμεσα με το περιβάλλον δίνοντας την δυνατότητα στον πράκτορα να επιλέξει τις ενέργειες του σύμφωνα με τις αλληλεπιδράσεις και τις ανταμοιβές που λαμβάνει εκτελώντας τις ενέργειες του, χωρίς να απαιτείται η ύπαρξη κάποιου εκπαιδευτή. Κατά την εφαρμογή μάθησης με ενίσχυση για ένα σύστημα AMC ο πράκτορας μπορεί να μάθει και να διατυπώσει το καλύτερο σχήμα διαμόρφωσης και κωδικοποίησης χρησιμοποιώντας την προηγούμενη εμπειρία που αποκτήθηκε

σε πραγματικό χρόνο από τις καταστάσεις του καναλιού , κάνοντας ελάχιστες υποθέσεις σχετικά με το σενάριο επικοινωνίας .Με στόχο τη μεγιστοποίηση της αποτελεσματικότητας του φάσματος , μια διαδικασία απόφασης Markov κατασκευάζεται για να αποφασίσει ποια διάταξη διαμόρφωσης και κωδικοποίησης θα πρέπει να χρησιμοποιηθεί .[89] Πιο συγκεκριμένα η μάθηση με ενίσχυση για AMC δεν απαιτεί offline εκπαίδευση που πρέπει να λαμβάνει υπόψη όλες τις πιθανές καταστάσεις που ενδέχεται να αντιμετωπίσουν οι φυσικές μεταδόσεις και, συνεπώς, αυτός ο τύπος μάθησης AMC μπορεί να παρέχει πιο προσαρμοστικές υπηρεσίες από ότι η μάθηση με επίβλεψη. Ας το θυμηθούμε εν συντομία :

Διαδικασίες αποφάσεων Markov

Η μέθοδος της μηχανικής μάθησης με ενίσχυση μπορεί να αναπαρασταθεί από τη θεωρία αποφάσεων Markov (Markov Decision Process-MDP). Υπάρχουν τέσσερα βασικά στοιχεία στην θεωρία αποφάσεων Markov:

1. η κατάσταση S
2. η ενέργεια A
3. η συνάρτηση μετάβασης P
4. η ανταμοιβή R

Πιο συγκεκριμένα $S=\{s_1, s_2, \dots, s_n\}$ αντιπροσωπεύει το σύνολο των N πιθανών καταστάσεων που περιγράφει τις δυναμικές μεταβολές του περιβάλλοντος , $A=\{a_1, a_2, \dots, a_m\}$ το οποίο αντιπροσωπεύει το σύνολο των m πιθανών ενεργειών του πράκτορα , $P: S \times A \times S \rightarrow [0, 1]$ είναι μια συνάρτηση πιθανότητας μετάβασης στην οποία $P(s,a,s')$ είναι η πιθανότητα μετάβασης από την κατάσταση $s \in S$ στην κατάσταση $s' \in S$ όταν εκτελείται η ενέργεια $a \in A$, και R είναι η συνάρτηση ανταμοιβής με $R(s,a)$ είναι η άμεση ανταμοιβή του περιβάλλοντος όταν γίνεται η ενέργεια a στην κατάσταση s . Στο k -οστό στάδιο της διαδικασίας μάθησης ο πράκτορας εκτελεί μια ενέργεια $a_k \in A$ στην κατάσταση $s_k \in S$. Στο επόμενο στάδιο , πριν η κατάσταση αλλάξει σε $s_{k+1} \in S$ με πιθανότητα $P(s_k, a_k, s_{k+1})$, ο πράκτορας λαμβάνει μια ανταμοιβή $R_k(s_k, a_k)$ παραγόμενη από το περιβάλλον. Αργότερα, πραγματοποιείται μια παρόμοια διαδικασία , και ο πράκτορας λαμβάνει μια σειρά από επακόλουθες ανταμοιβές R_{k+1} , R_{k+2} , R_{k+3} ,..... από το περιβάλλον.

Στόχος του πράκτορα είναι να βρει την πολιτική π που καθορίζει τη συμπεριφορά από τις αντίστοιχες καταστάσεις των ενεργειών , $\pi: S \rightarrow A$ για να λάβει όσο το δυνατόν μεγαλύτερη ανταμοιβή.

Χρησιμοποιώντας τη διαδικασία Markov, τα συστήματα AMC που υποστηρίζονται από ML μπορούν να κάνουν ενέργειες βασισμένο στην συλλογή των καταστάσεων περιβάλλοντος και των ανταμοιβών για προσαρμογή στις δυναμικές συνθήκες καναλιού. Πιο συγκεκριμένα :

Ενέργειες : Ο στόχος του AMC είναι η μεγιστοποίηση της απόδοσης για μια δεδομένη

κατάσταση του περιβάλλοντος υιοθετώντας τα σχήματα διαμόρφωσης και κωδικοποίησης. Σε πρακτικά πρωτόκολλα όπως το IEEE 802.11n, υπάρχει μόνο ένας πεπερασμένος αριθμός σχημάτων διαμόρφωσης και κωδικοποίησης. Κάθε σχήμα θεωρείται ενέργεια και ο πράκτορας επιλέγει το βέλτιστο σχήμα βάσει προηγούμενων εμπειριών λίγο πριν από τη μετάδοση πακέτων.

Καταστάσεις : Η δημιουργία μιας ζεύξης επιτρέπει στον πράκτορα να συλλέγει πληροφορίες, όπως προηγούμενες και τρέχουσες καταστάσεις καναλιών, χαρακτηριστικά δικτύου κ.α.. Παρόλο που η κατάσταση του περιβάλλοντος μπορεί να προσδιοριστεί από διάφορα χαρακτηριστικά, το λαμβανόμενο SNR κατά μέσο όρο σε όλα τα υποφέροντα χρησιμοποιείται για τον προσδιορισμό της κατάστασης του περιβάλλοντος. Λόγω του περιορισμού ότι η τιμή της κατάστασης πρέπει να ανήκει σε ένα πεπερασμένο σύνολο, θεωρούμε τις τιμές SNR στην περιοχή από $-2-20$ dB με το βήμα 1 dB για να αποφύγουμε έναν άπειρο αριθμό καταστάσεων. Ως αποτέλεσμα, ένα σύνολο $n = 23$ καταστάσεων χρησιμοποιούνται για τον προσδιορισμό του περιβάλλοντος. Σε κάθε κατάσταση, ο πράκτορας επιλέγει μία ενέργεια από ένα σύνολο πεπερασμένων ενεργειών και διαφορετικές ενέργειες οδηγούν σε διαφορετικές ανταμοιβές. Δεδομένου ότι η ζεύξη καναλιού μπορεί να είναι χρονομεταβλητή, ο πράκτορας πρέπει συνεχώς να παρακολουθεί και να ενημερώνει την τιμή του $Q(s, a)$, σε πραγματικό χρόνο. Για να μεγιστοποιήσει την επιτεύξιμη απόδοση, ο πράκτορας πρέπει να αναζητήσει και να βρει το βέλτιστο MCS για μια δεδομένη κατάσταση του καναλιού.

Ανταμοιβές : Για να διασφαλιστεί η αποτελεσματικότητα της εφαρμογής της προσέγγισης RL σε ένα πρακτικό πρόβλημα, η συνάρτηση ανταμοιβής R πρέπει να οριστεί κατάλληλα σε ένα συγκεκριμένο πλαίσιο. Λαμβάνοντας υπόψη το σενάριο προσαρμογής ζεύξης, το throughput που επιτυγχάνεται με την ανάληψη δράσης a σε κατάσταση s χρησιμοποιείται για τον καθορισμό της ανταμοιβής της συνάρτησης R :

$$R(s, a) = \log_2(M_a) \rho_a [1 - PER(s, a)]$$

όπου το M_a δηλώνει το βαθμό διαμόρφωσης της ενέργειας a , P_a είναι ο ρυθμός κωδικοποίησης κατά την ενέργεια a , και το $PER(s, a)$ είναι ο ρυθμός σφάλματος πακέτου κατά την ενέργεια a σε μια κατάσταση καναλιού s . Χρησιμοποιώντας CRC σε κάθε πακέτο, ο δέκτης μπορεί να ανιχνεύσει πακέτα με σφάλματα και το PER μπορεί να εκτιμηθεί από τις μετρήσεις πακέτων σφαλμάτων. Αυτές οι πληροφορίες στη συνέχεια μεταφέρονται στον πομπό για να ενημερώσουν τον αλγόριθμο εκμάθησης προκειμένου να αυξηθεί η απόδοση του φάσματος.

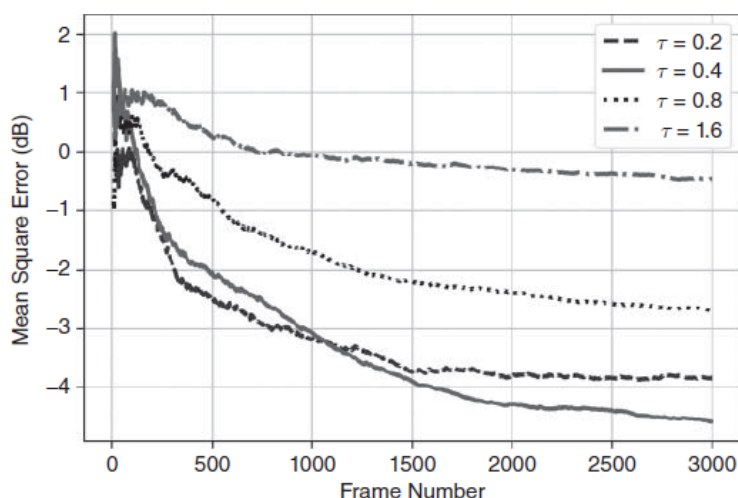
➤ Ανάλυση απόδοσης από προσομοιώσεις

Λόγω της εξερεύνησης, το πλαίσιο RL δεν είναι τόσο ισχυρό όσο το SL. Στη συνέχεια, παρουσιάζονται προσομοιώσεις για να διερευνήσουμε την απόδοση του AMC με υποστήριξη RL.

Στις προσομοιώσεις, εξετάζουμε μόνο τον συνδυασμό διαμόρφωσης $m = 8$ και κωδικοποίησης με $N_{ss} = 1$. Έχουμε ορίσει $\tau = 0,4$ στη μέθοδο επιλογής softmax που δίνεται από την εξίσωση softmax ενώ το ποσοστό εκμάθησης του αλγορίθμου Q-learning, που δίνεται από την (εξίσωση q-learning), έχει οριστεί σε $\alpha = 0,4$ από προεπιλογή.

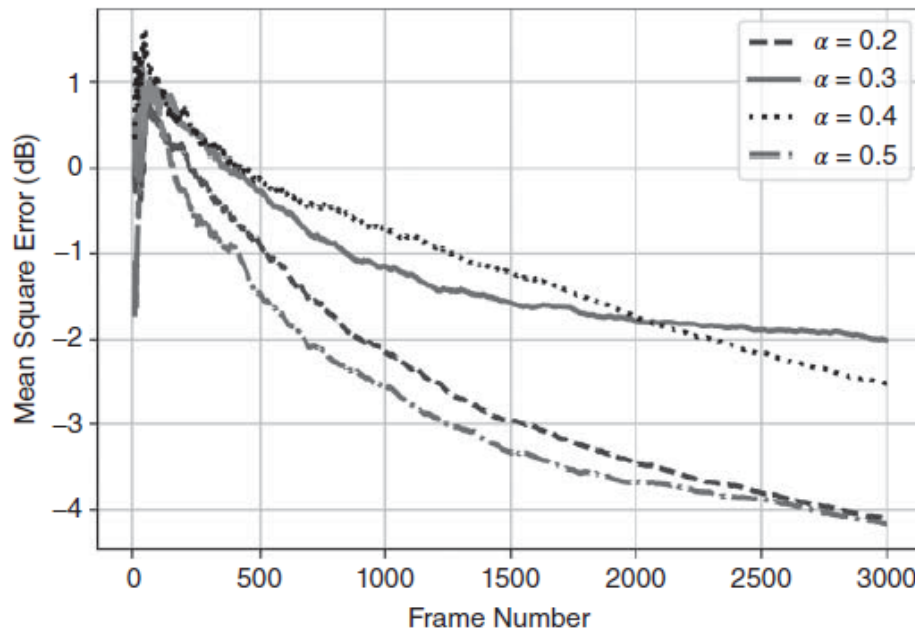
Τα επόμενα δύο σχήματα δείχνουν τα αποτελέσματα των παραμέτρων, συμπεριλαμβανομένης της παραμέτρου softmax και του ρυθμού εκμάθησης, στη συμπεριφορά σύγκλισης του αλγορίθμου. Για να δούμε τις διαφορές μεταξύ των πολιτικών RL και της διαφορετικής διαμόρφωσης και κωδικοποίησης για ένα δεδομένο SNR, το μέσο τετράγωνο σφάλμα (MSE) υπολογίζεται πάνω από τον αριθμό των πλαισίων (frames).

Από το επόμενο σχήμα μπορούμε να δούμε ότι όσο χαμηλότερη είναι η τιμή της παραμέτρου softmax, τόσο πιο γρήγορη είναι η σύγκλιση. Ο λόγος είναι ότι για $\tau = 0$, προτιμώνται ενέργειες με υψηλότερη τιμή Q. Ωστόσο, το χαμηλό τ μπορεί να οδηγήσει σε κακή απόδοση του συστήματος επειδή η διαδικασία εκμάθησης ολοκληρώνει την εξερεύνησή της πολύ νωρίς. Για παράδειγμα, $\tau = 0,2$ οδηγεί σε MSE περίπου -1dB , το οποίο είναι μεγαλύτερο από αυτό του σεναρίου $\tau = 0,4$ ή $\tau = 0,8$.



Σχήμα 2.17. Επίδραση της παραμέτρου softmax τ στη σύγκλιση του αλγορίθμου RL για $\alpha = 0,4$. [89]

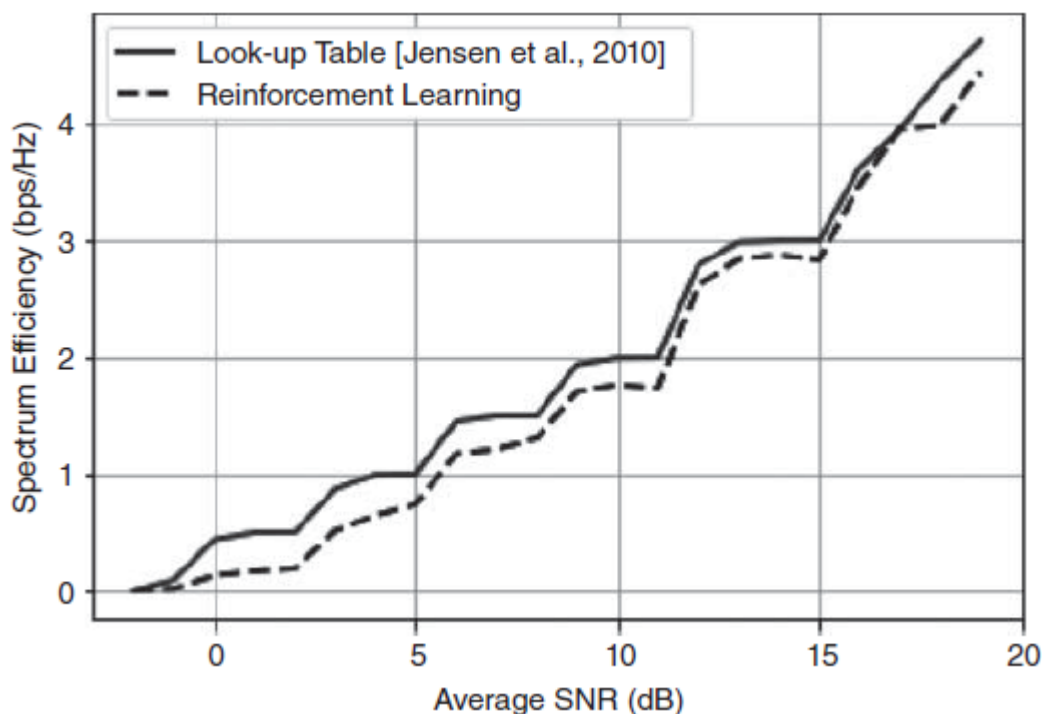
Επιπλέον, μπορεί να παρατηρηθεί από το επόμενο σχήμα ότι όταν το ποσοστό εκμάθησης α έχει μεγαλύτερη αξία, η μηχανή εκμάθησης μπορεί να λάβει πιο άμεσες ανταμοιβές από τις συσσωρευμένες ανταμοιβές και αντίστροφα. Ως εκ τούτου, το ποσοστό εκμάθησης μπορεί να χρησιμοποιηθεί για την ισορροπία της εκμετάλλευσης και της εξερεύνησης και το βέλτιστο ποσοστό μάθησης εξαρτάται από συγκεκριμένα συστήματα. Για παράδειγμα, το σύστημα προσομοίωσης έχει την καλύτερη απόδοση όταν $\alpha = 0,4$



Σχήμα 2.18. Επίδραση του ρυθμού εκμάθησης α στη σύγκλιση του αλγόριθμου RL για $\tau = 0,4$. [89]

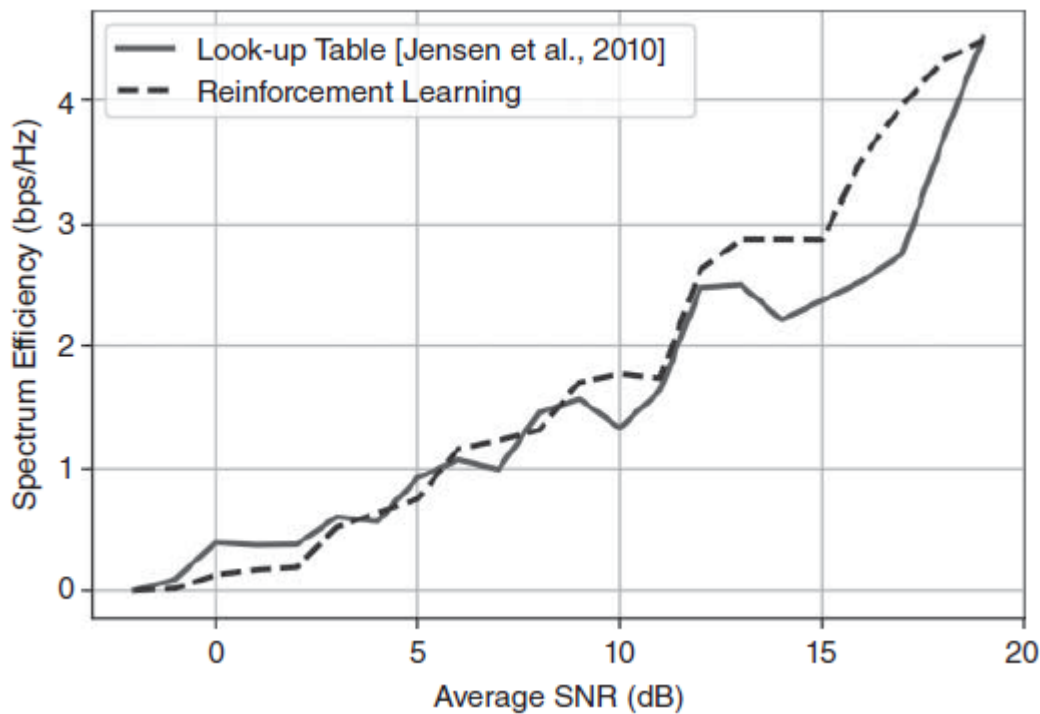
Στη συνέχεια συγκρίνουμε την απόδοση του συστήματος χρησιμοποιώντας την προσέγγιση πίνακα αναζήτησης, η οποία είναι επίσης γνωστή ως χαρτογράφηση RawBER [91] που χρησιμοποιείται για προσαρμογές ζεύξεων. Στην αντιστοίχιση RawBER, χρησιμοποιείται η πιθανότητα μη κωδικοποιημένων σφαλμάτων bit σε κάθε δευτερεύον υποφέρον για τον υπολογισμό του LQM. Συγκεκριμένα, η σχέση μεταξύ AWBER και PER μπορεί να προσδιοριστεί από παλινδρόμηση που δημιουργείται από προσομοιώσεις που πραγματοποιούνται μέσω του καναλιού λευκού Gaussian θορύβου (AWGN), το οποίο μπορεί να προετοιμαστεί εκ των προτέρων. Το μειονέκτημα της μεθόδου αυτής είναι ότι απαιτεί μεγάλη ποσότητα μνήμης και χρόνο προσομοίωσης για κάθε σενάριο. Για παράδειγμα, σε ορισμένα σενάρια, η υπόθεση Gauss - δηλ. Η παρεμβολή μαζί με τον θόρυβο Gauss μπορεί να θεωρηθεί ως ενιαία κατανομή Gauss - μπορεί να είναι παράλογη [92]. Επομένως, ένα είδος παραγόμενων δεδομένων δεν μπορεί να εγγυηθεί την κάλυψη όλων των πιθανών καταστάσεων. Κατά συνέπεια, οι πίνακες αναζήτησης ενδέχεται να οδηγήσουν σε υπο βέλτιστες λύσεις.

Ας εξετάσουμε το σενάριο όπου το SNR καναλιού μπορεί να ληφθεί ιδανικά. Το επόμενο σχήμα δείχνει ότι η φασματική απόδοση είναι συνάρτηση του μέσου SNR σε αυτό το σενάριο. Μπορεί να παρατηρηθεί ότι η προσέγγιση RL επιτυγχάνει χαμηλότερη απόδοση σε σύγκριση με την προσέγγιση πίνακα αναζήτησης, καθώς η τελευταία προσέγγιση βελτιστοποιείται στο συγκεκριμένο κανάλι, ενώ η RL προσαρμόζεται σταδιακά στο περιβάλλον.



Σχήμα 2.19. Μέση απόδοση φάσματος του πίνακα αναζήτησης και της τεχνικής RL σε ένα κανάλι AWGN με ιδανική εκτίμηση SNR.[89]

Αντίθετα, όπως φαίνεται στο επόμενο σχήμα, όταν το κανάλι SNR δεν μπορεί να ληφθεί ιδανικά λόγω του σφάλματος εκτίμησης, το οποίο είναι ένα πιο κοινό σενάριο σε πρακτικά συστήματα επικοινωνίας, το προτεινόμενο σχήμα RL μπορεί να ξεπεράσει την προσέγγιση του πίνακα αναζήτησης. Αυτό συμβαίνει επειδή το RL είναι σε θέση να μάθει από το περιβάλλον και να προσαρμοστεί στο περιβάλλον, συμπεριλαμβανομένου του σφάλματος εκτίμησης SNR, ενώ η προσέγγιση του πίνακα αναζήτησης πρέπει να βελτιστοποιηθεί σε ένα συγκεκριμένο περιβάλλον. Επιπλέον, το RL μπορεί να λειτουργεί online, και ως εκ τούτου δεν απαιτεί ειδικούς ή εκτεταμένες προσομοιώσεις για προσαρμογή σε διαφορετικά σενάρια. Η καλύτερη επιλογή MCS δεν απαιτεί εξαντλητική διαδικασία δοκιμής και σφάλματος, αλλά μικρή προσπάθεια προγραμματισμού για την κατασκευή του συστήματος.



Σχήμα 2.20. Μέση απόδοση φάσματος του πίνακα αναζήτησης και της τεχνικής RL μέσω ενός καναλιού AWGN με σφάλμα εκτίμησης SNR.[89]

Από τα αποτελέσματα και τις αναλύσεις της προσομοίωσης φαίνεται ότι το πλαίσιο RL μπορεί να προσφέρει μια πολλά υποσχόμενη λύση για το πρόβλημα AMC. Σε αυτήν τη μέθοδο, η μεγιστοποίηση της φασματικής απόδοσης θεωρείται ως διαδικασία απόφασης Markov και ο μέσος SNR χρησιμοποιείται για τον προσδιορισμό της κατάστασης. Μια σχέση μεταξύ των τιμών SNR, διαμόρφωσης και σχήματα κωδικοποίησης μπορεί να καθοριστεί υιοθετώντας -μάθηση για την επίλυση του προβλήματος της διαδικασίας απόφασης του Markov. Από τις προσομοιώσεις, μπορούμε να δούμε περαιτέρω ότι το AMC με τη βοήθεια RL μπορεί να χρησιμοποιηθεί για τη δημιουργία ραδιοζευξέων καθώς η εκπαίδευση εκτός σύνδεσης δεν απαιτείται από την RL. Επιπλέον, αυτή η μέθοδος είναι προσαρμοσμένη στις αλλαγές του περιβάλλοντος. Έτσι, με τη βοήθεια του RL, η AMC μπορεί να προσαρμόσει δυναμικά τις παραμέτρους και να παρέχει υπηρεσίες σε χρήστες με υψηλότερη απόδοση και αξιοπιστία.

➤ Συμπεράσματα

Για να επαληθευτεί η αποτελεσματικότητα της εφαρμογής αυτών των αλγορίθμων εκμάθησης στο AMC, έχουν γίνει προσομοιώσεις με ρυθμίσεις παραμέτρων που αναφέρονται στο πρότυπο IEEE 802.11n για να συγκρίνουν τα παρουσιαζόμενα σχήματα AMC που υποστηρίζονται από ML με παραδοσιακές προσεγγίσεις. Αξίζει να σημειωθεί ότι τα συστήματα AMC που βασίζονται σε SL είναι κατάλληλα για το σενάριο όπου τα δεδομένα εκπαίδευσης είναι αντιπροσωπευτικά όλων των καταστάσεων στις οποίες ενδέχεται να εκτεθεί ο πομπός. Αντιθέτως, ο αλγόριθμος RL μπορεί να μάθει άμεσα από το περιβάλλον αλληλεπίδρασης και μπορεί σταδιακά να επιτύχει

ικανοποιητική απόδοση χωρίς εκπαίδευση εκτός σύνδεσης όπως απαιτείται από το SL. Επομένως, αυτοί οι δύο τύποι μηχανισμών μάθησης μπορούν να επιλεγούν ανάλογα με το εάν η εκπαίδευση πραγματοποιείται offline ή online με βάση διαφορετικές απαιτήσεις χρηστών σε διαφορετικά σενάρια εφαρμογών.

Χάρη στους μηχανισμούς της μάθησης και της εκπαίδευσης, οι προσεγγίσεις AMC με υποστήριξη ML έχουν επιτύχει έξυπνη και εξαιρετική αξιοπιστία σε ασύρματα συστήματα. Οι μελλοντικές ερευνητικές κατευθύνσεις μπορεί να περιλαμβάνουν τον συνδυασμό RL και SL, για το πως θα ενισχύσει την ανθεκτικότητα των συστημάτων AMC που υποστηρίζονται από ML και πώς να επιτύχει καλύτερες αντισταθμίσεις μεταξύ της πολυπλοκότητας και της απόδοσης του συστήματος, με στόχο την παροχή πιο προσαρμοστικών, έξυπνων και καλύτερων υπηρεσιών για τελικούς χρήστες.

2.7 Μηχανική μάθηση για μη γραμμικούς ανιχνευτές MIMO

Τα ασύρματα συστήματα καθιστούν δυνατή την παροχή συστημάτων επικοινωνίας με ρυθμούς δεδομένων της τάξης των Gbps χρησιμοποιώντας μεγάλο αριθμό από συστοιχία κεραιών ή / και χρησιμοποιώντας μεγάλο bandwidth (πολλών gigahertz). Η κοινή αδυναμία και των δύο προσεγγίσεων είναι η μεγάλη κατανάλωση ενέργειας στο δέκτη, που προκαλείται από υψηλής ακρίβειας (π.χ. 8~14-bit ακριβείας) μετατροπείς αναλογικού σε ψηφιακό σήμα (ADC), επειδή η συνολική ισχύς καταναλώνεται από ADC κλιμακώνεται γραμμικά με τον αριθμό των επιπέδων ακρίβειας, το εύρος ζώνης και τον αριθμό των ADC. Για παράδειγμα, η κατανάλωση ενέργειας των ADC φαίνεται να είναι ανάλογη τόσο με τον αριθμό των επιπέδων ακρίβειας όσο και με το εύρος ζώνης, κατά την δειγματοληψία Nyquist. Επομένως, η χρήση ADC υψηλής ακρίβειας στο δέκτη καθίσταται ανέφικτος όταν χρησιμοποιείται μεγάλη συστοιχία κεραιών ή / και μεγάλο εύρος ζώνης. Οι ADC χαμηλής ανάλυσης (π.χ. ακρίβεια 1~3 bit) έχουν θεωρηθεί ως οικονομικά αποδοτικότερη λύση για τη μείωση της κατανάλωσης ισχύος για ασύρματων συστημάτων massive MIMO αλλά και για συστήματα επικοινωνίας μεγάλου εύρους ζώνης.

Ένα μειονέκτημα της χρήσης ADC χαμηλής ανάλυσης είναι ότι είναι δύσκολο να ληφθούν ακριβείς πληροφορίες κατάστασης καναλιού στον δέκτη (channel-state information at the receiver (-CSIR) με συμβατικές τεχνικές εκτίμησης καναλιού που βασίζονται σε πιλοτικά κανάλια. Επιπλέον, οι συμβατικές μέθοδοι ανίχνευσης δεδομένων, που αναπτύχθηκαν για γραμμικά συστήματα MIMO, παρέχουν χαμηλή απόδοση ανίχνευσης λόγω της μη γραμμικότητας στους ADC.

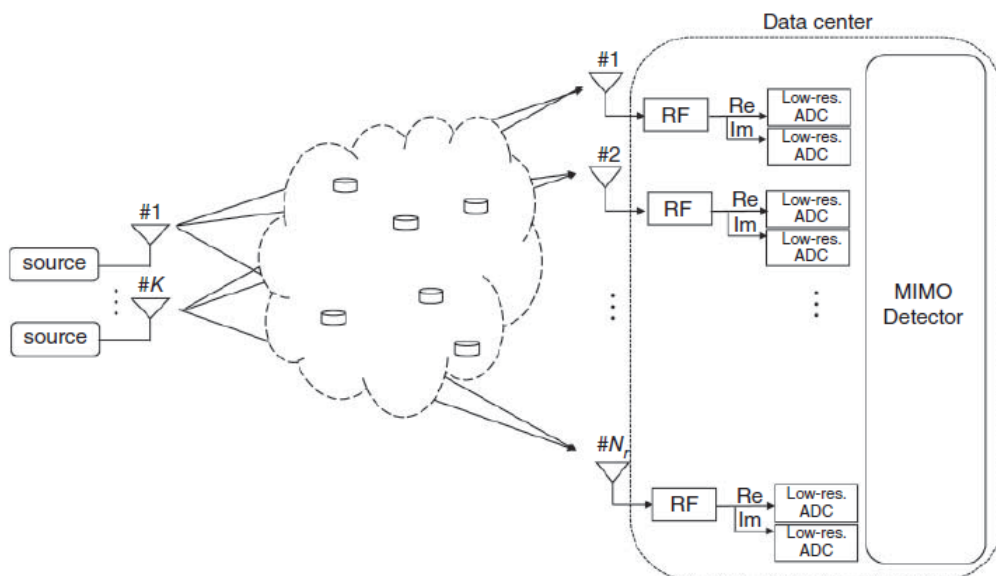
Έχει πραγματοποιηθεί αρκετή έρευνα για την επίλυση προβλημάτων εκτίμησης καναλιού και ανίχνευσης δεδομένων για το uplink σε συστήματα massive MIMO με ADC ενός bit. Για το πρόβλημα εκτίμησης καναλιών σε τέτοια συστήματα, έχουν αναπτυχθεί πολλές μέθοδοι για τη βελτίωση της ακρίβειας του CSIR. Οι μέθοδοι αυτοί περιλαμβάνουν μεθόδους όπως τα ελάχιστα τετράγωνα (least-squares (LS)), την μέγιστη πιθανοφάνεια (maximum-likelihood-ML) καθώς

και την αποσύνθεση Bussgang (Bussgang decomposition) , ενώ έχουν αναπτυχθεί και άλλοι μέθοδοι λιγότερο αποδοτικοί.

Οι τεχνικές αυτές μπορούν να επεκταθούν σε ένα σύστημα πολλαπλών διαδρομών-κόμβων MIMO (multihop MIMO), που ονομάζουμε καταναμημένο σύστημα MIMO uplink (distributed uplink MIMO system), όπου η λειτουργία μεταφοράς δεδομένων-καναλιών από άκρο σε άκρο μεταξύ πηγών K και του κέντρου δεδομένων είναι εξαιρετικά μη γραμμική έχοντας ως αποτέλεσμα να είναι δύσκολη η εκτίμηση του καναλιού με περιορισμένο αριθμό πιλοτικών καναλιών. Αυτό έχει ως αποτέλεσμα την εξέταση ενός ανιχνευτή βασισμένο σε δεδομένα μηχανικής μάθησης με επίβλεψη (SL), όπου τα πιλοτικά σήματα (ή τα δεδομένα εκπαίδευσης) αξιοποιούνται για να μάθουν άμεσα έναν ανιχνευτή MIMO, αντί να εκτιμήσουν μια σύνθετη λειτουργία μεταφοράς καναλιών.

➤ Το μοντέλο ενός καναλιού Multihop MIMO

Στο επόμενο σχήμα παρουσιάζεται ένα σύστημα multihop distributed uplink MIMO με K πομπούς , οι οποίοι εκπέμπουν ανεξάρτητα σε ένα data center με ενδιάμεσους αναμεταδότες. Το data center είναι εξοπλισμένο με $N_r \geq K$ κεραιές λήψης με one-bit ADCs (uplink).



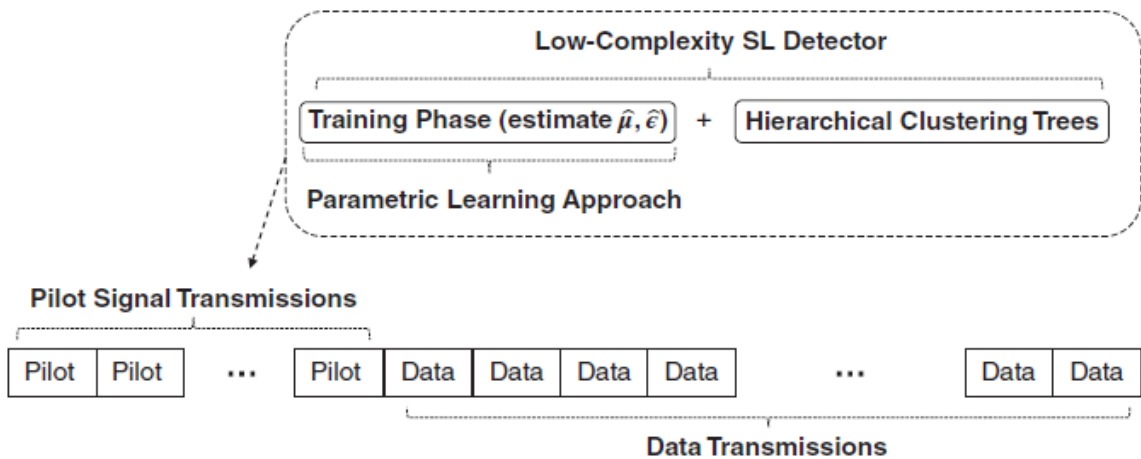
Σχήμα 2.21. Σύστημα multihop[98]

Ορίζουμε $w_k \in \{0, \dots, m-1\}$ ως το μήνυμα από την πηγή-κεραία $k \in \{1, \dots, K\}$ καθένα από τα οποία περιέχει \log_m bit πληροφορίας. Ορίζουμε επίσης το m -ary constellation set ως $S = \{s_0, \dots, s_{m-1}\}$ με ισχύ $\frac{1}{m} \sum_{i=0}^{m-1} |s_i|^2 = P_t$. Το σύμβολο $\text{sign}(\cdot): \mathbb{R} \rightarrow \{1, -1\}$ αντιπροσωπεύει το one bit ADC με $\text{sign}(u)=1$, όταν $u \geq 0$ και $\text{sign}(u)=-1$ σε κάθε άλλη περίπτωση. Το μεταδιδόμενο σύμβολο της πηγής k , \hat{x}_k λαμβάνεται από την συνάρτηση διαμόρφωσης $f: W \rightarrow S$ as $\hat{x}_k = f(w_k) \in S$. Στη συνέχεια το πραγματικό και το φανταστικό μέρος λαμβάνεται από το data center :

$$r = \text{sign}(\Phi(\tilde{x}) + \tilde{z}) \in \{-1, 1\}^N,$$

όπου $N=2 N_t$, $\Phi(\cdot)$ αντιπροσωπεύει μια μιγαδική μη γραμμική συνάρτηση (ονομάζεται end-to-end channel transfer function) και $\tilde{z} = [\tilde{z}_1, \dots, \tilde{z}_N] \in \mathbb{R}^N$ αντιπροσωπεύει το διάνυσμα θορύβου τα στοιχεία του οποίου είναι κατανέμοντα ανεξάρτητα ως Gaussian τυχαίες μεταβλητές $\tilde{z}_i \sim (0, \sigma_z^2/2)$, ενώ η $\Phi(\cdot)$ καταγράφει όλες τις λειτουργίες των ενδιάμεσων αναμεταδοτών και όλα τα τοπικά ασύρματα κανάλια στο δίκτυο.

Το πλαίσιο επικοινωνίας αποτελείται από φάσεις εκπαίδευσης και μετάδοσης δεδομένων (επόμενο σχήμα), ενώ διέρχεται αυτές τις δύο φάσεις, το κανάλι θεωρείται ότι είναι σταθερό εντός του χρόνου συνοχής.



Σχήμα 2.22. Φάση εκπαίδευσης και μετάδοσης δεδομένων [98]

Φάση εκπαίδευσης: Κατά τη διάρκεια αυτής της φάσης, K πηγές μεταδίδουν «γνωστές» ακολουθίες (δηλαδή πιλοτικά σήματα) έτσι ώστε το κέντρο δεδομένων να μπορεί να μάθει μια μη γραμμική συνάρτηση $\Phi(\cdot)$. Από την άποψη της μηχανικής μάθησης, το κέντρο δεδομένων συλλέγει τα δεδομένα και τις αντίστοιχες ετικέτες. Έστω $\mathcal{M} = \{0, \dots, m-1\}^K$ αντιπροσωπεύει το σύνολο όλων των πιθανών μηνυμάτων από τις K πηγές. Για κάθε κλάση $c \in \mathcal{M}$, οι K πηγές εκπέμπουν T πιλοτικά σήματα x_i^c για $i=1, \dots, T$. Το κέντρο δεδομένων μπορεί να συλλέξει το σύνολο δεδομένων (labelled data) D ως :

$$D = \{\tilde{r}_i^c : c \in \mathcal{M}, i = 1, \dots, T\}$$

όπου $r_i^c \in \{-1, 1\}^N$. Μετά την αποστολή των πιλοτικών σημάτων, ο δέκτης δημιουργεί εμπειρικές (υπό συνθήκη) συνάρτηση μάζας πιθανότητας (η λειτουργία μαζικής Πιθανότητας) (probability mass functions-PMFs) χρησιμοποιώντας τα ληφθέντα σήματα που παρατηρήθηκαν κατά τη διάρκεια της πιλοτικής μετάδοσης. Μια εμπειρική υπό συνθήκη PMFs για κάθε μήνυμα (κλάση) c μπορεί να δοθεί ως:

$$\hat{p}(r|c) = \frac{1}{T} \sum_{i=1}^T \mathbf{1}_{\{r=\tilde{r}_i^c\}} \text{ for } c \in 1, \dots, \mathcal{M},$$

(εξίσωση 1)

όπου $1_{\{A\}}$ είναι μια ενδεικτική συνάρτηση που ισούται με 1 αν το A είναι αληθές και μηδέν αλλιώς.

Φάση μετάδοσης δεδομένων: Δεδομένου του D και μιας νέας παρατήρησης r , το data center εντοπίζει την κλάση c (δηλαδή μηνύματα χρηστών $\hat{w} = (\hat{w}_1, \dots, \hat{w}_k)$) ως $\Psi(r) = c \in M$.

2.7.1 Ανιχνευτής MIMO με επιβλεπόμενη μάθηση

Το πρόβλημα της ανίχνευσης με μηχανική μάθησης μπορεί να ταξινομηθεί σε δυο προσεγγίσεις, μη παραμετρική μάθηση και παραμετρική μάθηση. Η μη παραμετρική μάθηση όπως, κ-πλησιέστερος γείτονας (k-NN), δέντρα απόφασης, μηχανές διανυσματικής υποστήριξης (SVM), δεν απαιτεί την προγενέστερη γνώση του συνόλου των δεδομένων D (π.χ. κατανομή), ενώ η παραμετρική μάθηση όπως, λογιστική παλινδρόμηση, native Bayes και νευρωνικά δίκτυα τα δεδομένα υποτίθεται ότι παράγονται από ένα πιθανοτικό μοντέλο με ορισμένες παραμέτρους (π.χ. μοντέλο Gaussian), και στη συνέχεια, οι παράμετροι βελτιστοποιούνται χρησιμοποιώντας το σύνολο δεδομένων D

2.7.1.1 Μη παραμετρική μάθηση

Η μη παραμετρική μάθηση χαρακτηρίζεται από PMF που βασίζεται στο σύνολο δεδομένων. Η μη παραμετρική μάθηση κατηγοριοποιείται σε ανιχνευτές μέγιστης πιθανότητας (empirical-maximum-likelihood-eMLD) και ανίχνευσης ελάχιστης μέσης απόστασης (minimum-mean-distance detection-MMD), ως εξής:

1) eMLD : Η βασική ιδέα είναι η επιλογή ενός δείκτη για το διάνυσμα συμβόλων εισόδου που μεγιστοποιεί την εξίσωση 1 ως εξής :

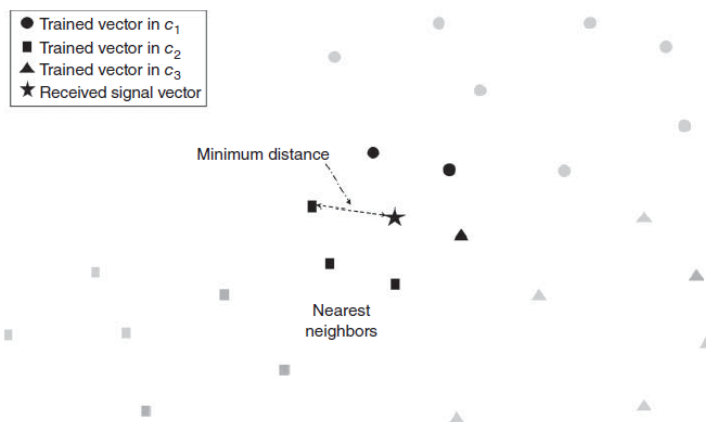
$$\hat{c} = \underset{c \in \{1, \dots, M\}}{\operatorname{argmax}} \hat{p}(r|c).$$

Όταν η εκπαίδευση των επαναλήψεων T αυξάνεται στο άπειρο, η ανίχνευση είναι ισοδύναμη με τη βέλτιστη μέθοδο ανίχνευσης μέγιστης πιθανότητας (maximum-likelihood), καθώς η εμπειρική κατανομή συγκλίνει στην αντίστοιχη πραγματική κατανομή από το νόμο των μεγάλων αριθμών. Όταν το T είναι ανεπαρκές, η εμπειρική κατανομή δεν μπορεί να αντιπροσωπεύει την πραγματική κατανομή, με αποτέλεσμα σφάλματα εντοπισμού.

Πιο συγκεκριμένα, υπάρχει μη μηδενική πιθανότητα που οδηγεί σε $\hat{p}(r|c) = 0$ για όλα τα μηνύματα $c \in M$. Για την επίλυση αυτού του προβλήματος ο δέκτης βρίσκει $\tilde{r} \in D$ που είναι πιο κοντά στο διάνυσμα r . Αν $N(r)$ είναι το σύνολο που είναι πιο κοντά στο διάνυσμα, τότε η μέθοδος eMLD $\Psi_{eMLD} : r \rightarrow C$ δίνεται από

$$\Psi_{eMLD}(r) = \operatorname{argmax}_{r \in \mathcal{N}} \sum_{\tilde{r} \in \mathcal{N}} \hat{p}(\tilde{r}|c).$$

Όπως φαίνεται και στο επόμενο σχήμα η τεχνική μοιάζει με αυτή της ταξινόμησης πλησιέστερου γείτονα K-NN με την έννοια ότι συγκρίνουν απλώς τον αριθμό των γειτονικών ετικετών. Μία αξιοσημείωτη διαφορά είναι ότι το eMLD χρησιμοποιεί το γειτονικό σύνολο των στοιχείων που απέχουν εξίσου από τον λαμβανόμενο διάνυσμα .



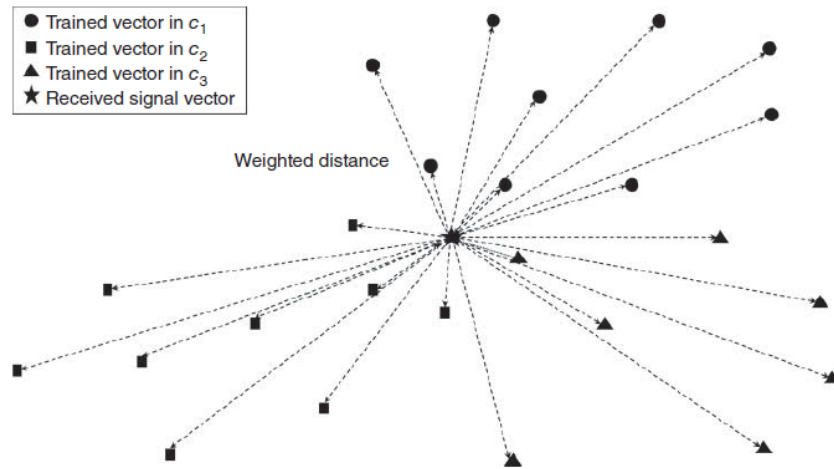
Σχήμα 2.23. eMLD[98]

2) MMD : Το μειονέκτημα του eMDL είναι ότι χρησιμοποιεί εν μέρη την εμπειρική υπό συνθήκη PMFs επειδή το σύνολο περιορίζεται μόνο στους πλησιέστερους γείτονες (N) . Για την πλήρη εκμετάλλευση των εμπειρικών PMFs, εισάγουμε μια εναλλακτική μέθοδο ανίχνευσης. Όπως φαίνεται στο επόμενο σχήμα η μέθοδος MMD $\Psi_{MMD} : r \rightarrow C$, επιλέγει τον δείκτη του διανύσματος συμβόλου που δίνει την ελάχιστη μέση απόσταση :

$$\Psi_{MMD}(r) = \operatorname{argmin}_{c \in \mathcal{M}} E_{\tilde{r}_i} [||r - \tilde{r}_i||_2 | c] = \operatorname{argmin}_{c \in \mathcal{M}} \sum_{\tilde{r}_i} ||r - \tilde{r}_i||_2 \hat{p}(\tilde{r}_i | c).$$

Το MMD βρίσκει το μήνυμα που ελαχιστοποιεί το σταθμισμένο άθροισμα των εμπειρικών PMF, όπου τα βάρη είναι η απόσταση μεταξύ του λαμβανόμενου διανύσματος και του διανύσματος εκπαίδευσης, $||r - \tilde{r}_i||_2$

Παρόλο που η βέλτιστη απόδοση του MMD δεν είναι εγγυημένη, μπορεί να έχει καλύτερη απόδοση από το eMLD όταν το L είναι ανεπαρκές, επειδή το MMD χρησιμοποιεί επιπλέον πληροφορίες αξιοπιστίας που καταγράφονται από την απόσταση μεταξύ του ληφθέντος σήματος και του εκπαιδευμένου σήματος



Σχήμα 2.24. MMD[98]

2.7.1.2 Παραμετρική μάθηση

Η αδυναμία των μη παραμετρικών προσεγγίσεων (eMLD, MMD) είναι ότι προκαλούν μεγάλη υπολογιστική πολυπλοκότητα. Και στις δύο μεθόδους, ο δέκτης πρέπει να υπολογίσει όλες τις αποστάσεις του σετ εκπαίδευσης ($|D| = M \times T$). Όταν το μέγεθος του D είναι μεγάλο (π.χ. $T \gg 1$), μια μη παραμετρική προσέγγιση είναι δύσχρηστη στα πρακτικά συστήματα. Ως εκ τούτου, με την εκμάθηση παραμέτρων κατά τη διάρκεια της εκπαίδευσης, μπορούμε να μειώσουμε σημαντικά της πολυπλοκότητα.

- 1) Μηχανική μάθηση βασισμένη σε μοντέλο Gauss (SL based on a Gaussian model-SLGM)

Σε αυτήν την προσέγγιση, είναι πολύ σημαντικό να προσδιοριστεί ένα κατάλληλο μοντέλο πιθανότητας για ένα δεδομένο σύνολο δεδομένων D . Το Gaussian μοντέλο είναι το πιο συχνά χρησιμοποιούμενο όπου τα δεδομένα $r \in D$ θεωρείται ότι δημιουργούνται από την κατανομή πιθανότητας $p(r|c, \theta_c) = N(\mu_c, \Sigma_c)$ όπου $c \in M$ αντιπροσωπεύει την κλάση (μήνυμα) των K πηγών και θ_c αντιπροσωπεύει το διάνυσμα παραμέτρων για την κλάση c . Χρησιμοποιώντας τα δοθέντα δεδομένα $\{\tilde{r}_t^c : t = 1 \dots T\}$, μπορούμε να βελτιστοποιήσουμε το $\theta_c = (\hat{\mu}_c, \hat{\Sigma}_c)$ μέσω του εκτιμητή maximum likelihood ως εξής :

$$\hat{\mu}_c = \frac{1}{T} \sum_{t=1}^T \tilde{r}_t^c$$

$$\hat{\Sigma}_c = \frac{1}{T} \sum_{t=1}^T (\tilde{r}_t^c - \hat{\mu}_c)(\tilde{r}_t^c - \hat{\mu}_c)^T, \quad (\text{εξίσωση 1})$$

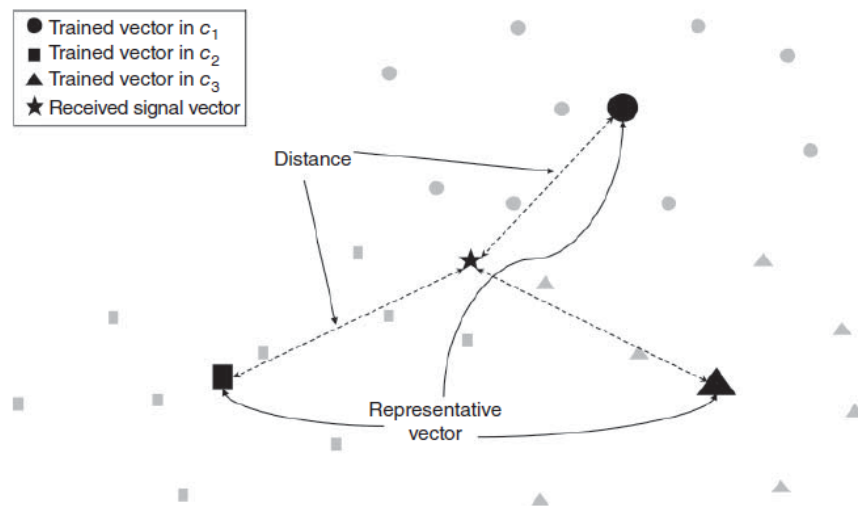
όπου $\hat{\mu}_c, \hat{\Sigma}_c$ αντιπροσωπεύουν τη μέση τιμή και τη συνδιακύμανση των δεδομένων εκπαίδευσης

της κλάσης c . Το $\hat{\mu}_c$ θα μπορούσαν να θεωρηθούν αντιπροσωπευτικά διανύσματα της κάθε κλάσης c . Όταν τα δεδομένα εκπαίδευσης είναι ανεπαρκής ο πίνακας συνδιακύμανσης τείνει να είναι ανεπαρκής κατάταξης (rank-deficient) και γίνεται ill-conditioned.

Αυτό το πρόβλημα μπορεί να επιλυθεί με έναν εκτιμητή συρρίκνωσης. Δοθέντος $\hat{\theta}_c = (\hat{\mu}_c, \hat{\Sigma}_c)$ ο βέλτιστος ανιχνευτής maximum-likelihood δίνεται ως

$$\Psi_{PGD}(r) = \underset{c \in \mathcal{M}}{\operatorname{argmin}} (r - \hat{\mu}_c)^T \hat{\Sigma}_c^{-1} (r - \hat{\mu}_c).$$

Το μέτρο μέτρησης αυτής της απόστασης ονομάζεται απόσταση Mahalanobis, και ο αντίστροφος του $\hat{\Sigma}_c$ τις εξισώσεις 1 ονομάζεται πίνακας ακριβείας. Όταν $\Sigma_c = I$ για όλα τα c , ως ειδική περίπτωση, ο ανιχνευτής που προκύπτει είναι ισοδύναμος με τον ανιχνευτή ελάχιστης κεντρικής απόστασης (minimum-centered-distance-MCD). Δεδομένου ότι η πολυπλοκότητα του eMLD είναι μεγαλύτερη από το MCD, ο MCD προτείνεται.



Σχήμα 2.25. MCD[98]

2) Ανιχνευτής με μοντέλο Bernoulli (SL based on a Bernoulli-like model-SLBM)

Σε αυτό το μοντέλο τα δεδομένα υποτίθεται ότι παράγονται από την ακόλουθη κατανομή πιθανότητας :

$$P(r|c, \theta_c) = \prod_{i=1}^N e_{c,i}^{1_{\{r_i \neq \mu_{c,i}\}}} (1 - e_{c,i})^{1_{\{r_i = \mu_{c,i}\}}},$$

όπου $\theta_c = (\mu_c, \varepsilon_c)$ $c \in M$, $\varepsilon_{c,j} < 0.5$ για κάθε i και $1_{\{A\}}$ υποδηλώνει μια συνάρτηση δείκτη με $1_{\{A\}} = 1$ αν A αληθείς και $1_{\{A\}} = 0$ σε άλλη περίπτωση. Δοθέντος των δεδομένων εκπαίδευσης για την κλάση c ($D = \{r_c^t : t = 1 \dots T\}$), η παράμετρος θ_c βελτιστοποιείται χρησιμοποιώντας εκτιμητή ML ως

$$(\hat{\mu}_c, \hat{\epsilon}_c) = \operatorname{argmax}_{\mu_c, \epsilon_c} \prod_{t=1}^T P(\tilde{r}_t^c | \mu_c, \epsilon_c).$$

ενώ συνδυάζοντας τις δυο προηγούμενες εξισώσεις παίρνουμε

$$(\hat{\mu}_c, \hat{\epsilon}_c) = \operatorname{argmax}_{\mu_c, \epsilon_c} \prod_{i=1}^N \prod_{t=1}^T \epsilon_{c,i}^{\mathbf{1}_{\{\tilde{r}_{t,i}^c \neq \mu_{c,i}\}}} (1 - \epsilon_{c,i})^{\mathbf{1}_{\{\tilde{r}_{t,i}^c = \mu_{c,i}\}}}.$$

Για κάθε $\epsilon_{c,i} < 0.5$ η συνάρτηση μπορεί να μεγιστοποιηθεί παίρνοντας

$$\hat{\mu}_{c,i} = \operatorname{sign} \left(\sum_{t=1}^T \tilde{r}_{t,i}^c \right),$$

για $i=1, \dots, N$. Αν υποθέσουμε ότι

$$N_d = \sum_{t=1}^T \mathbf{1}_{\{\tilde{r}_{k,i}^c \neq \hat{\mu}_{c,i}\}} \text{ and } N_s = \sum_{t=1}^T \mathbf{1}_{\{\tilde{r}_{k,i}^c = \hat{\mu}_{c,i}\}},$$

μπορούμε να βρούμε ένα βέλτιστο $\epsilon_{c,i}$ ανεξάρτητα από τα άλλα $\epsilon_{c,j}$, $i \neq j$ παίρνοντας τη λύση

$$\operatorname{argmax}_{\epsilon_{c,i}} \epsilon_{c,i}^{N_d} (1 - \epsilon_{c,i})^{N_s}$$

$$\frac{\partial (\epsilon_{c,i}^{N_d} (1 - \epsilon_{c,i})^{N_s})}{\partial \epsilon_{c,i}} = 0,$$

το βέλτιστο $\epsilon_{c,i}$ θα είναι :

$$\hat{\epsilon}_{c,j} = \frac{\sum_{t=1}^T \mathbf{1}_{\{\hat{\mu}_{c,j} \neq \tilde{r}_{t,j}^c\}}}{T}.$$

Συνδυάζοντας τις προηγούμενες εξισώσεις ο βέλτιστος εκτιμητής maximum-likelihood θα είναι:

$$\Psi_D(\mathbf{r}) = \operatorname{argmin}_{c \in \mathcal{M}} (\mathbf{r} - \hat{\mu}_c)^T \begin{bmatrix} -\log \hat{\epsilon}_{c,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & -\log \hat{\epsilon}_{c,N} \end{bmatrix} (\mathbf{r} - \hat{\mu}_c).$$

Όταν χρησιμοποιούνται ADC ενός bit στους δέκτες, φαίνεται ότι το SLBM ξεπερνά το SLGM δεδομένου ότι το πρώτο έχει αναπτυχθεί και είναι πιο κατάλληλο για την επεξεργασία δυαδικών δεδομένων. Η αρχή της παραμετρικής μάθησης είναι πολύ κοντά στην αρχή του πλησιέστερου κέντρου ταξινόμησης (nearest-centroid classifier-NCC), που είναι μια απλή λύση του προβλήματος ταξινόμησης σε SL. Το NCC εκχωρεί την ετικέτα τάξης ενός μη επισημασμένου διανύσματος χρησιμοποιώντας τα κεντρικά διανύσματα που αντιπροσωπεύουν τις κλάσεις τους. Ομοίως, στην παραμετρική μάθηση, ο ανιχνευτής SL λειτουργεί επιλέγοντας ένα μήνυμα που έχει την ελάχιστη απόσταση από το μέσο διάνυσμα των πιλοτικών σημάτων. Αυτή η

ομοιότητα είναι ένα καλό παράδειγμα για να δείξει μια σύνδεση μεταξύ ενός προβλήματος ανίχνευσης δεδομένων στο ασύρματο κανάλι και ενός προβλήματος ταξινόμησης μηχανικής μάθησης.

2.7.2 Low-Complexity SL (LCSL) Detector

Αν και ένας παραμετρικός ανιχνευτής μάθησης απαιτεί χαμηλότερη πολυπλοκότητα από έναν μη παραμετρικό ανιχνευτή μάθησης, η υπολογιστική πολυπλοκότητα είναι απαγορευτική ενώ το μέγεθος του χώρου αναζήτησης ($|M| = m^k$) μεγαλώνει εκθετικά με το K . Έτσι, όπως και στα συμβατικά συστήματα MIMO, μπορεί να ληφθεί υπόψη η σφαιρική αποκωδικοποίηση, η οποία βρίσκει αποτελεσματικά έναν μειωμένο χώρο αναζήτησης, ωστόσο, η συμβατική σφαιρική αποκωδικοποίηση δεν μπορεί να χρησιμοποιηθεί άμεσα λόγω της μη γραμμικότητας των εξεταζόμενων μοντέλων καναλιών. Οι λεγόμενες αποκωδικοποιήσεις σφαίρας ενός bit (OSD), οι οποίες είναι κατάλληλες για ποσοτικοποιήσεις ενός bit, έχουν προταθεί (Jeon et al. (2018β), Kim et al. (2017)). Σε αυτήν την ενότητα, εισάγεται OSD που βασίζεται σε ένα ιεραρχικό δάσος ομαδοποίησης (OSD-HCF), το οποίο μπορεί να αποδώσει υψηλότερης ποιότητας μειωμένο χώρο αναζήτησης. Σε αυτήν τη μέθοδο, χρησιμοποιείται ένας γρήγορος δυαδικός αλγόριθμος αναζήτησης πλησιέστερου γείτονα [99], ο οποίος βρίσκει έναν μειωμένο χώρο αναζήτησης αποτελεσματικά, αξιοποιώντας ιεραρχικές δομές ομαδοποίησης (μηχανική μάθηση χωρίς επίβλεψη). Σε συνδυασμό με τον ανιχνευτή SL, ονομάζεται ανιχνευτής SL (LCSL) χαμηλής πολυπλοκότητας.

Οι συνολικές διαδικασίες του ανιχνευτή LCSL παρέχονται ως εξής:

1. Πρώτα αποσυνθέτουμε τον (δυαδικό) χώρο Hamming ιεραρχικά έτσι ώστε να χτιστεί μια δομή δέντρου. Ξεκινάμε με την επιλογή J στοιχείων από \hat{U} τυχαία, τα οποία δρουν ως J κέντρα της συστάδας-συμπλέγματος.
2. Το προηγούμενο βήμα σχηματίζει τις συστάδες J γύρω από αυτά τα κεντροειδή και η διαδικασία αποσύνθεσης επαναλαμβάνεται αναδρομικά (επόμενος αλγόριθμος-Αλγόριθμος 1).
3. Αυτή η διαδικασία εκτελείται W φορές για την κατασκευή δέντρων $\{T_1, \dots, T_w\}$.

Algorithm 1 Constructing a hierarchical clustering tree $h(C_i, J)$

Input: $\hat{U} = \{\hat{\mu}_1, \dots, \hat{\mu}_{|M|}\}$

Output: hierarchical clustering tree \mathcal{T}_i

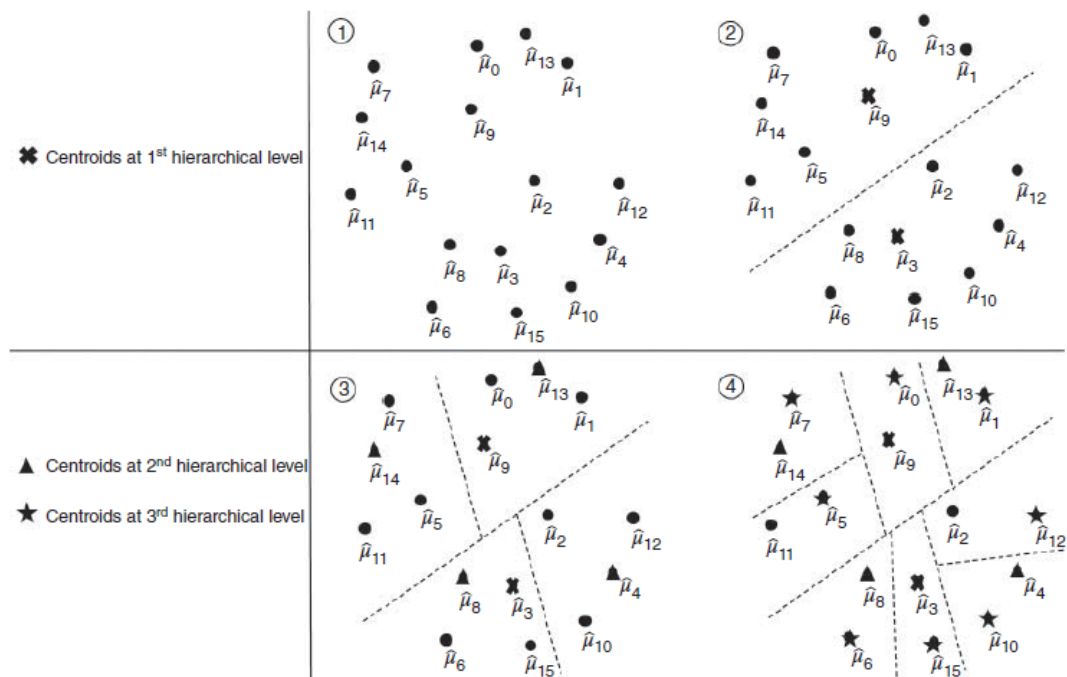
Parameters: J (branching factor and maximum leaf size)

- 1: **if** $|\hat{U}| \leq J$ **then**
 - 2: *create leaf node with the elements in \hat{U}*
 - 3: **else**
 - 4: $\mathcal{P} \leftarrow$ *select J elements randomly from \hat{U} (centroids)*
 - 5: $\mathcal{C} \leftarrow$ *cluster the elements in \hat{U} w. r. t. the centroid $i \in \mathcal{P}$*
 - 6: **for** each cluster $C_i \in \mathcal{C} = \{C_1, \dots, C_J\}$ **do**
 - 7: *create non-leaf node with centroid \mathcal{P}_i*
 - 8: *recursively apply the algorithm $h(C_i, J)$ (with the updated C_i)*
 - 9: **end for**
 - 10: **end if**
-

Τα προκύπτοντα δέντρα θα χρησιμοποιηθούν στη φάση μετάδοσης δεδομένων για να μειώσουν

αποτελεσματικά τον χώρο αναζήτησης σύμφωνα με μια νέα παρατήρηση r . Ο αλγόριθμος κατασκευάζει πολλαπλά ιεραρχικά δέντρα που έχουν πιθανώς διαφορετικές δομές αποσύνθεσης, και έτσι μπορεί να βελτιώσει την ποιότητα ενός προκύπτοντος μειωμένου χώρου αναζήτησης.

Στο επόμενο σχήμα φαίνεται αυτή η διαδικασία (Αλγόριθμος 1) δίνοντας ένα απλό παράδειγμα ($J = 2, |M| = 16$) στην περίπτωση ενός μόνο δέντρου. Η διαδικασία αποσύνθεσης εφαρμόζεται σε δυαδικό χώρο ιεραρχικά μέχρι το μέγεθος του συμπλέγματος να είναι μικρότερο από το J , το οποίο ονομάζεται μέγεθος του κόμβου των φύλλων. (η δομή εξόδου αυτής της διαδικασίας μπορεί να θεωρηθεί ως δέντρο). Εάν εκτελέσουμε αυτήν τη διαδικασία πολλές φορές, κατασκευάζονται ένα δάσος.



Σχήμα 2.26. Παράδειγμα μιας διαδικασίας αποσύνθεσης.[98]

Φάση μετάδοσης δεδομένων: Λαμβάνοντας υπόψη μια τρέχουσα παρατήρηση r , ο αλγόριθμος αναζήτησης ξεκινά με τη διάσχιση πολλαπλών δέντρων παράλληλα (Επόμενος αλγόριθμος -Αλγόριθμος 2). Τα πολλαπλά δέντρα W μοιράζονται μια ουρά προτεραιότητας (Q) και οι κόμβοι στην ουρά προτεραιότητας είναι διατεταγμένοι με τη μικρότερη απόσταση hamming σε σχέση με την παρατήρηση r . Στη συνέχεια, μπορεί να παράγει αποτελεσματικά τον μειωμένο χώρο αναζήτησης $S(r) \subseteq M$, ο οποίος περιέχει μόνο τα πλησιέστερα $\hat{\mu}_c$'s στο r . Χρησιμοποιώντας αυτό, ο ανιχνευτής LCSL εκτελείται ως:

$$\Psi_D(r) = \underset{c \in S(r)}{\operatorname{argmin}} (r - \hat{\mu}_c)^T \begin{bmatrix} -\log \hat{\epsilon}_{c,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & -\log \hat{\epsilon}_{c,N} \end{bmatrix} (r - \hat{\mu}_c).$$

Algorithm 2 Searching parallel hierarchical clustering trees

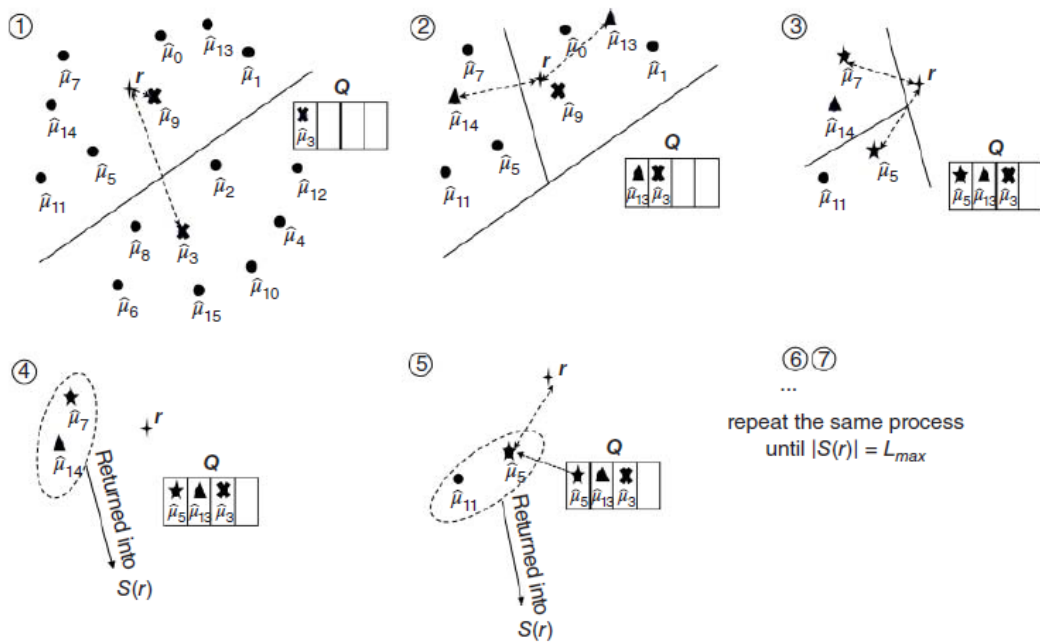
Input: hierarchical clustering trees $\{\mathcal{T}_i : i = 1, \dots, W\}$ and a new observation \mathbf{r}

Output: $S(\mathbf{r})$ (reduced search space associated with \mathbf{r})

Parameters: L_{\max} (the desired size of a reduced search space, e.g. $|S(\mathbf{r})| = L_{\max}$)

```
1:  $L \leftarrow 0$  { $L =$  number of points  $\hat{\mu}_c$  searched}
2:  $Q \leftarrow$  empty priority queue
3:  $\mathcal{R} \leftarrow$  empty priority queue
4: for each tree  $T_i$  do
5:   call  $\text{TraverseTree}(T_i, Q, \mathcal{R})$ 
6: end for
7: while  $|Q| \neq 0$  and  $L < L_{\max}$  do
8:    $j \leftarrow$  top index of  $Q$ 
9:   call  $\text{TraverseTree}(j, Q, \mathcal{R})$ 
10: end while
11: return  $K$  top points from  $\mathcal{R}$ 
1: procedure  $\text{TRAVERSE TREE}(j, Q, \mathcal{R})$ 
2:   if node  $j$  is a leaf node then
3:      $S \stackrel{\Delta}{=} \{\text{all the elements in leaf node } j\}$ 
4:      $\mathcal{R} = \mathcal{R} \cup S$ 
5:      $L \leftarrow L + |S|$ 
6:   else
7:      $C \leftarrow$  child nodes of  $N$ 
8:      $i \leftarrow$  closest node of  $C$  to observation  $\mathbf{r}$ 
9:      $C_p \leftarrow C \setminus \{i\}$ 
10:    add all nodes in  $C_p$  to  $Q$ 
11:    call  $\text{TraverseTree}(i, Q, \mathcal{R})$ 
12:   end if
13: end procedure
```

Στο επόμενο σχήμα, παρέχουμε ένα παράδειγμα που βρίσκει μειωμένο χώρο αναζήτησης με βάση τη δομή που δημιουργήθηκε στο προηγούμενο παράδειγμα (σχήμα). Κατά την πρώτη αναζήτηση, η παρατήρηση συνεχίζει να επιλέγει το πλησιέστερο κεντροειδές σε κάθε επίπεδο έως ότου φτάσει σε έναν κόμβο φύλλων.



Σχήμα 2.27. Σύντομο παράδειγμα αλγορίθμου αναζήτησης δέντρων.[98]

Τα στοιχεία που περιλαμβάνονται στον κόμβο των φύλλων επιστρέφονται στον μειωμένο χώρο αναζήτησης ($S(r)$). Επιπλέον, τα μη επιλεγμένα κεντροειδή τοποθετούνται στην ουρά προτεραιότητας που ταξινομεί τα κεντροειδή με τέτοια σειρά που καθιστά δυνατή μια αποτελεσματική αναζήτηση πλησιέστερου γείτονα. Για τη δεύτερη αναζήτηση, η ίδια μέθοδος αναζήτησης αρχίζει από τον πρώτο κόμβο στην ουρά προτεραιότητας. Ο αλγόριθμος τελειώνει όταν ικανοποιεί το προκαθορισμένο μέγεθος του μειωμένου χώρου αναζήτησης ($|S(r)|$). Πέρα από την αναζήτηση ενός δέντρου, μια αναζήτηση δέντρου επεκτείνεται σε έναν αλγόριθμο αναζήτησης δασών (Αλγόριθμος 2) όπου κατασκευάζονται πολλές διαφορετικές δομές δέντρων.

➤ Προσομοιώσεις και αποτελέσματα

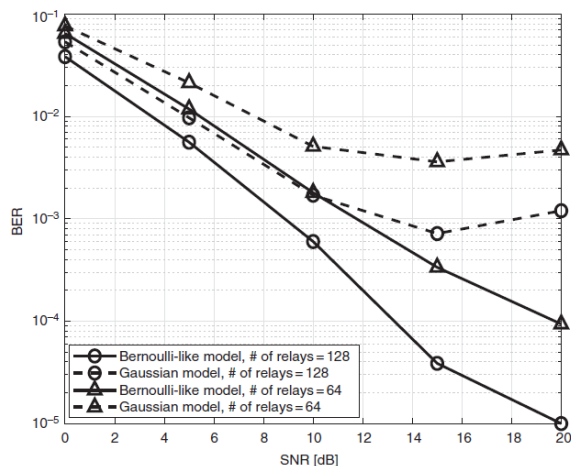
Σε αυτήν την ενότητα, αξιολογούμε τις μέσες επιδόσεις bit-error rate (BER) των SLGM και SLBM ανιχνευτών. Επιπλέον, φαίνεται ότι ο ανιχνευτής LCSL μπορεί να επιτύχει απόδοση με πολύ χαμηλότερη πολυπλοκότητα σε συστήματα MIMO.

Για τις προσομοιώσεις, έχει θεωρηθεί ένα κανάλι Rayleigh για τα ασύρματα κανάλια όπου κάθε στοιχείο του πίνακα καναλιού αντλείται από ένα ανεξάρτητο και ταυτόσημα κατανομημένο (independent and identically distributed-i.i.d.) κυκλικά συμμετρικό σύμπλεγμα Gaussian τυχαίας μεταβλητής με μηδενική μέση τιμή και διακύμανση μονάδα. Επιπλέον, οι λειτουργίες των αναμεταδοτών (μιας κεραίας) θεωρείται ότι ενισχύουν και προωθούν (amplify-and-forward-AF) το σήμα ενώ υποθέτουμε διαμόρφωση QPSK. Όταν η εκπαίδευση είναι μικρή (π.χ. το T είναι μικρό), η πιθανότητα εμπειρικού σφάλματος (ϵ_{cj}) μπορεί να υποτιμηθεί ως μηδέν (αν και στην πραγματικότητα δεν ισχύει). Αυτό μπορεί να προκαλέσει σοβαρό πρόβλημα σφάλματος. Για να αποφευχθεί αυτό, εκχωρούμε μια ελάχιστη τιμή $\hat{\epsilon}_{cj} = 10^{-3}$.

Το επόμενο σχήμα δείχνει τις επιδόσεις BER των παραμετρικών ανιχνευτών μάθησης με εκπαίδευση $T = 15$. Ακόμα λαμβάνονται υπόψη τα ακόλουθα δύο σενάρια:

- (i) 64 ενδιάμεσοι αναμεταδότες.
- (ii) 128 ενδιάμεσοι αναμεταδότες .

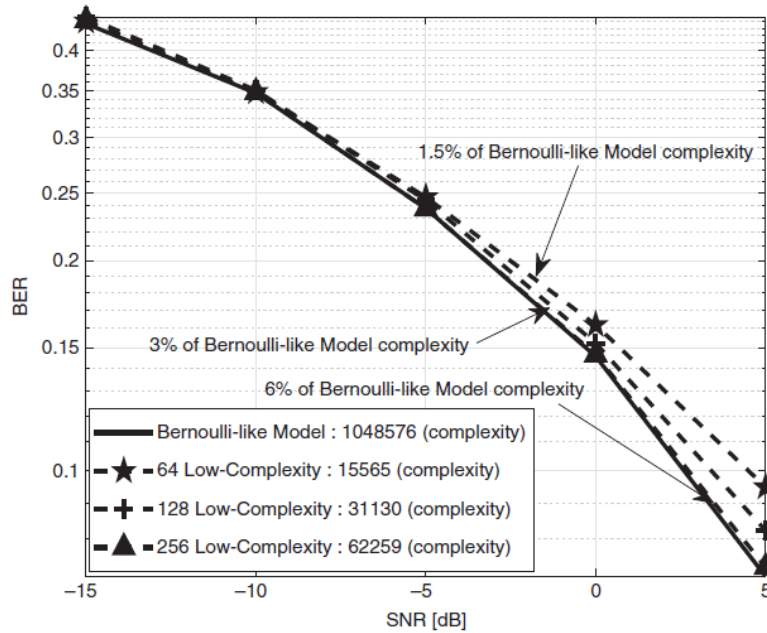
Το σχήμα δείχνει ότι ο ανιχνευτής SLBM ξεπερνά το SLGM ανιχνευτής, που σημαίνει ότι το μοντέλο Bernoulli είναι πιο κατάλληλο για δυαδικά δεδομένα από ένα Gaussian μοντέλο.



Σχήμα 2.28. $K = 8$, $N_r = 64$, and $T = 15$. Συγκρίσεις απόδοσης των ανιχνευτών SL με βάση τον Bernoulli και μοντέλα Gauss, ανάλογα με τον αριθμό των ενδιάμεσων αναμεταδοτών. [98]

Το επόμενο σχήμα δείχνει την απόδοση BER του ανιχνευτή χαμηλής πολυπλοκότητας σύμφωνα με το L_{max} στον Αλγόριθμο 2. Επίσης, ορίζουμε το $J = 32$ στον Αλγόριθμο 1. Όπως φαίνεται, η πολυπλοκότητα είναι εξαιρετικά υψηλή όταν ο αριθμός των χρηστών γίνεται μεγάλος.

Παρατηρούμε ότι ο ανιχνευτής χαμηλής πολυπλοκότητας μπορεί να επιτύχει βέλτιστη απόδοση με μόνο το 6% της αρχικής πολυπλοκότητας. Επίσης, αναμένεται ότι η χρήση μιας τεχνικής χαμηλής πολυπλοκότητας είναι αποδοτικότερη για ένα σύστημα διανομής μεγάλης κλίμακας (π.χ. μεγάλο K).



Σχήμα 2.29. $K = 14$, $N_r = 64$, and $T = 15$. Απόδοση του ανιχνευτή LCSL ανάλογα με το μέγεθος του μειωμένου χώρος αναζήτησης (L_{max}) [98]

➤ Συμπεράσματα:

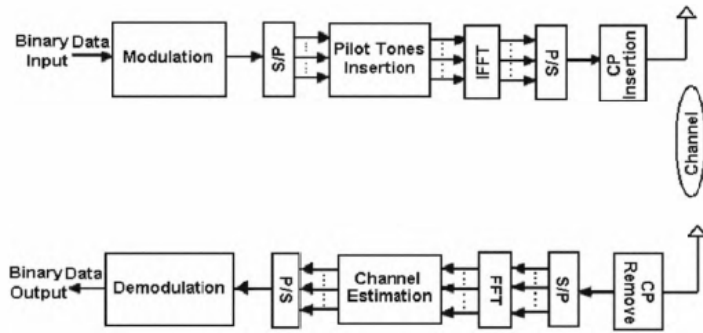
Σε αυτή την ενότητα, περιγράφηκαν ανιχνευτές για συστήματα Multihop MIMO, συγκεκριμένα όταν ο δέκτης είναι εξοπλισμένος με ADC ενός bit. Οι ανιχνευτές κατηγοριοποιούνται ως μη παραμετρικά ή παραμετρικά. Οι πρώτοι έχουν το πλεονέκτημα εκπαίδευσης μεγάλου συνόλου δεδομένων (π.χ. ένας μεγάλος αριθμός πιλοτικών σημάτων) Ωστόσο, αυτή η προσέγγιση πάσχει από εξαιρετικά υψηλή πολυπλοκότητα. Οι παραμετρικοί ανιχνευτές μπορούν να μειώσουν αυτήν την πολυπλοκότητα εκτιμώντας αντιπροσωπευτικά διανύσματα (ονομάζονται κωδικές λέξεις) που αντιστοιχούν σε κάθε τάξη (μήνυμα) κατά τη διάρκεια της φάσης εκπαίδευσης. Επί πλέον, συνδυάζοντάς τα με σφαιρική αποκωδικοποίηση ενός bit, μπορούν να αποδώσουν πολύ καλά με καλή πολυπλοκότητα. Έτσι, μπορούν να ληφθούν υπόψη οι ανιχνευτές αυτοί για ένα μη γραμμικό σύστημα επικοινωνίας.

2.8 OFDM με νευρωνικά δίκτυα

2.8.1 Εκτίμηση καναλιού σε σύστημα OFDM με νευρωνικό δίκτυο

Σε αυτή την ενότητα , παρουσιάζεται ένα νευρωνικό δίκτυο πολλαπλών επιπέδων perceptrons (MLP) με αλγόριθμο εκμάθησης backpropagation (BP) ως εκτιμητής καναλιών για συστήματα OFDM. Ο προτεινόμενος εκτιμητής με νευρωνικό δίκτυο MLP συγκρίνεται με τον αλγόριθμο ελαχίστου τετραγώνου (LS), τον αλγόριθμο ελάχιστου μέσου τετραγωνικού σφάλματος (MMSE) και το νευρωνικό δίκτυο ακτινικής βάσης (RBF) σε σχέση με κριτήρια σφάλματος bit (bit error rate-BER) και το μέσο τετραγωνικό σφάλμα (MSE) για την αξιολόγηση των επιδόσεων.

Η OFDM είναι ένα είδος τεχνικής διαμόρφωσης πολλαπλών φερόντων που διαιρεί το συνολικό εύρος ζώνης σήματος σε αριθμό των υποφερόντων , και στη συνέχεια μεταδίδει τις πληροφορίες για καθένα από αυτά τα υποφέροντα. Η φασματική αποδοτικότητα , η αντοχή στο multipath, η αντίσταση σε παρεμβολές μεταξύ συμβόλων (intersymbol interference-ISI) και η μετάδοση δεδομένων υψηλού ρυθμού είναι τα πλεονεκτήματα των συστημάτων OFDM. Για να εκτιμηθούν οι παράμετροι καναλιού σε κάθε υποφέρον, υπάρχουν δύο τύποι τεχνικών όπως η τυφλή εκτίμηση καναλιών (blind channel estimation) και η εκτίμηση καναλιού με πιλοτικά κανάλια. Οι τεχνικές εκτίμησης τυφλών καναλιών προσπαθούν να εκτιμήσουν τα CIR (Channel impulse response) χωρίς καμία γνώση των μεταδιδόμενων δεδομένων. Είναι ελκυστικά λόγω της πιθανής εξοικονόμησης στην εκπαίδευση, ωστόσο είναι αποτελεσματικά μόνο όταν μπορεί να συλλεχθεί μεγάλος όγκος δεδομένων. Αυτό είναι σαφώς ένα μειονέκτημα στην περίπτωση των κινητών ασύρματων συστημάτων λόγω της χρονικής διακύμανσης του καναλιού.[141] Η εκτίμηση καναλιού βάσει πιλότου είναι η άλλη προσέγγιση, στην οποία η ακολουθία εκπαίδευσης αποτελείται από γνωστά σύμβολα δεδομένων (πιλότοι) που μεταδίδεται συνεχώς στην αρχή της μετάδοσης και η αρχική εκτίμηση των παραμέτρων καναλιού πραγματοποιείται χρησιμοποιώντας τα ληφθέντα πιλοτικά σήματα. Οι τεχνικές εκτίμησης καναλιού που βασίζονται σε πιλότους έχουν καλύτερη απόδοση από τις τεχνικές τυφλής εκτίμησης. Οι αλγόριθμοι Least square (LS), minimum mean square error (MMSE) and least mean square (LMS) μπορούν να χρησιμοποιηθούν για την εκτίμηση καναλιών με χρήση πιλότων. Οι αλγόριθμοι αυτοί παρουσιάζουν αρκετά καλό αποτέλεσμα ο καθένας ξεχωριστά για την εκτίμηση του καναλιού , ενώ έχουν αρκετούς περιορισμούς σε σχέση με την απόδοση , τα υπολογιστικά κόστη ενώ γίνονται ίσως απαγορευτικοί για πιο σύνθετα κανάλια και απαιτητικά συστήματα επικοινωνιών.[142][133] Τα νευρωνικά δίκτυα έρχονται για να ξεπεράσουν αυτές τις δυσκολίες έχοντας καλύτερες προσεγγίσεις αλλά και μεγάλη ικανότητα μάθησης.



Σχήμα 2.30. OFDM Block [144]

Ένα τυπικό μπλοκ διάγραμμα OFDM φαίνεται στο προηγούμενο σχήμα. Η σειριακή δυαδική ροή δεδομένων ομαδοποιείται και αντιστοιχίζονται στην φάση του σήματος για να μεταδοθούν στο ασύρματο κανάλι. Μετά την εισαγωγή πιλοτικών συμβόλων που χρησιμοποιούνται για τη λήψη CIR, τα διαμορφωμένα σύμβολα $X_m(k)$ μετατρέπονται στο πεδίο του χρόνου $x_m(k)$ από το IFFT :

$$x_m(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_m(k) e^{j2\pi nk/N} \quad n = 0, 1, \dots, N-1$$

Όπου N είναι ο αριθμός των υποφερόντων, k είναι ο δείκτης του υποφέροντος m_{th} OFDM συμβόλου και n ορίζει τον αριθμό δείγματος του m_{th} OFDM συμβόλου. Για να αποφευχθεί η διασυμβολική παρεμβολή (inter symbol interference - ISI) εισάγουμε κυκλικό πρόθεμα και έτσι προκύπτει :

$$x_f(n) = \begin{cases} x_m(N+n) & n = -N_G, \dots, -1 \\ x_m(n) & n = 0, 1, \dots, N-1 \end{cases}$$

όπου $x_f(n)$ είναι το σήμα στο πεδίο του χρόνου συμπεριλαμβάνοντας το κυκλικό πρόθεμα και N_G είναι ο αριθμός δειγμάτων στο διάστημα του κυκλικού προθέματος. Στην συνέχεια το σήμα μεταδίδεται στο κανάλι. Το λαμβανόμενο σήμα είναι :

$$y_f(n) = x_f(n) \otimes h_m(n) + w_m(n)$$

όπου $h_m(n)$ είναι η συνάρτηση μεταφοράς του καναλιού στο πεδίο του χρόνου και $w_m(n)$ είναι λευκός Gaussian θόρυβος. Στο δέκτη αφού αφαιρέσουμε το κυκλικό πρόθεμα και μετά την εφαρμογή FFT παίρνουμε :

$$Y_m(k) = \frac{1}{N} \sum_{n=0}^{N-1} y_m(n) e^{-j2\pi kn/N} \quad k = 0, 1, \dots, N-1$$

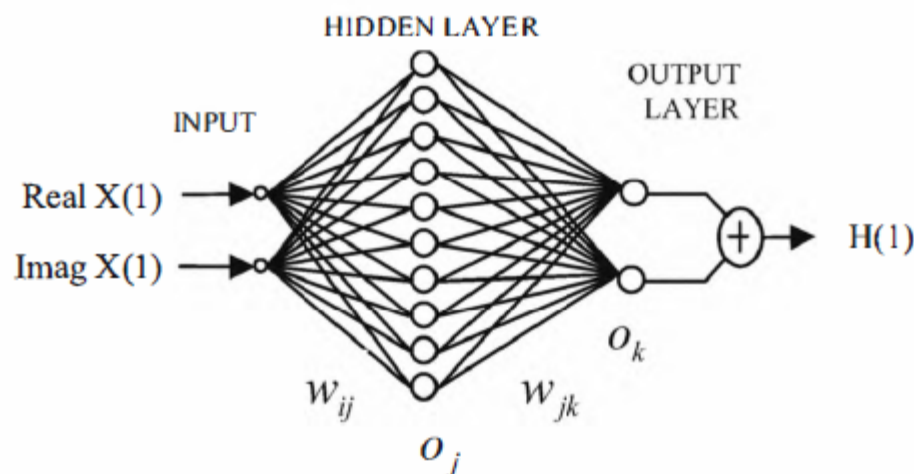
Το σήμα αντιπροσωπεύεται από :

$$Y_m(k) = X_m(k)H_m(k) + W_m(k)$$

όπου $H_m(k)$ είναι η συνάρτηση μεταφοράς του καναλιού στο πεδίο της συχνότητας. Έτσι η $H_m(k)$ εκτιμάται στο μπλοκ εκτίμησης καναλιού.

- Εκτίμηση καναλιού με τον αλγόριθμο back propagation

Το νευρωνικό δίκτυο που θα χρησιμοποιήσουμε είναι νευρωνικό δίκτυο MLP με τον αλγόριθμο back propagation για τη λήψη CIR. Ο εκτιμητής φαίνεται στο επόμενο σχήμα. Όπως φαίνεται στο σχήμα το δίκτυο MLP που προτείνεται για την εκτίμηση καναλιού έχει δύο εισόδους και δύο εξόδους. Για να υιοθετήσει το νευρικό δίκτυο στο OFDM, κάθε σήμα χωρίζεται σε πραγματικό και φανταστικό μέρος, καθώς τα σύμβολα OFDM συνίστανται από μιγαδικά σήματα. Στη συνέχεια, τα διαχωρισμένα σήματα εισάγονται στο δίκτυο και οι εξόδοι του δικτύου θα εκτιμήσουν την απόκριση του καναλιού.



Σχήμα 2.31 : Δομή MLP για εκτίμηση καναλιού [144]

Κατά την διαδικασία της επεξεργασίας του εκτιμητή, το πραγματικό και το φανταστικό μέρος των σημάτων τροφοδοτούνται μέσω του δικτύου και υπολογίζονται από το δίκτυο. Ο υπολογισμός γίνεται από τους υπολογισμούς του αθροίσματος βαρών που εισέρχεται στους κόμβους και εφαρμόζεται η σιγμοειδής συνάρτησης. Η συνάρτηση ενεργοποίησης του το κρυφού στρώματος είναι :

$$net_j = \sum_{i=1}^d X_i w_{ij} \quad o_j = f(net_j) = \frac{1}{1 + e^{-net_j}}$$

όπου d είναι ο αριθμός των σημάτων εισαγωγής X_i είναι τα εισαγόμενα δεδομένα στο δίκτυο και w_{ij} είναι τα βάρη του κρυμμένου επιπέδου στο κρυφό κόμβο j όπου εφαρμόζουμε την σιγμοειδή συνάρτηση.

Κάθε κόμβος εξόδου υπολογίζει την ενεργοποίηση του δικτύου του με βάση τα κρυφά σήματα εξόδου :

$$net_k = \sum_{j=1}^{n_H} o_j w_{jk} \quad o_k = f(net_k)$$

όπου ο δείκτης k της μονάδας εξόδου και n_H είναι αριθμός των κρυφών μονάδων. Στην περίπτωση της ενεργοποίησης, η συνάρτηση του επιπέδου εξόδου είναι γραμμική, η έξοδος του συστήματος o_k είναι:

$$o_k = f\left(\sum_{j=1}^{n_H} w_{jk} f\left(\sum_{i=1}^d w_{ij} X_i\right)\right)$$

Στην διαδικασία της εκπαίδευσης, τα βάρη της εισόδου των κρυφών επιπέδων w_{ij} και των κρυφών επιπέδων εξόδου w_{jk} υπολογίζονται ελαχιστοποιώντας την :

$$E(w) = \frac{1}{2} \sum_{k=1}^c (t_k - o_k)^2$$

όπου t_k είναι κ-στή επιθυμητή έξοδος και c είναι ο αριθμός των σημείων εξόδου. Ο κανόνας εκμάθησης του αλγορίθμου back propagation βασίζεται στη 'βαθμωτή κατάβαση' (gradient descent.) [1][17]

Τα βάρη αρχικοποιούνται με ψευδο-τυχαίες τιμές και αλλάζουν σε μια κατεύθυνση που θα μειώσει το σφάλμα:

$$\Delta w = -\eta \frac{\partial E}{\partial w}$$

όπου $\langle\langle \eta \rangle\rangle$ είναι το ποσοστό μάθησης που επιλέγεται μεταξύ 0 ή 1, το οποίο καθορίζει το πόσο αλλάζουμε τα βάρη σε κάθε βήμα. Αν το $\langle\langle \eta \rangle\rangle$ είναι πολύ μικρό ο αλγόριθμος θα χρειαστεί πολύ χρόνο για σύγκλιση. Αντίθετα αν είναι πολύ μεγάλο το δίκτυο θα εκπαιδευτεί πολύ γρήγορα και θα καταλήξουμε σε απόκλιση του αλγορίθμου -κορεσμό.

Η ενημέρωση βάρους (ή κανόνας εκμάθησης) για τα βάρη κρυμμένων επιπέδων εξόδου υπολογίζεται ως :

$$\Delta w_{jk} = \eta(t_k - o_k) f'(net_k) o_j$$

Ο κανόνας μάθησης για την είσοδο σε κρυμμένα βάρη είναι :

$$\Delta w_{ji} = \eta \left[\sum_{k=1}^c w_{jk} (t_k - o_k) f'(net_k) \right] f'(net_j) X_i$$

Η διαδικασία εκπαίδευσης ολοκληρώνεται όταν η τιμή Ε φτάσει τον επιθυμητό στόχο, ο οποίος καθορίζεται στην αρχή.

➤ Προσομοίωση και αποτελέσματα

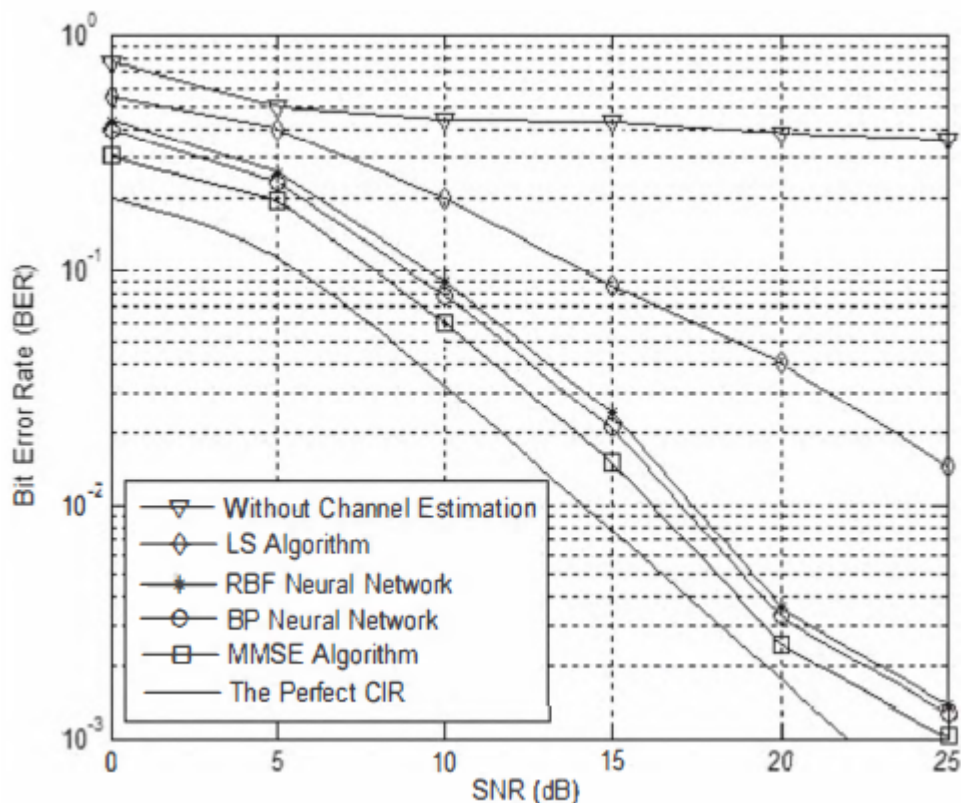
Στην προσομοίωση γίνεται σύγκριση των επιδόσεων του αλγορίθμου νευρωνικού δικτύου MLP, της λειτουργίας ακτινικής βάσης (radial basis function neural network - RBF) και του αλγορίθμου LS λαμβάνοντας υπόψη τον ρυθμό σφάλματος bit (BER) έναντι του λόγου σήματος προς θόρυβο (SNR) και του μέσου τετραγώνου σφάλμα (mean square error-MSE) έναντι κριτηρίων του λόγου σήματος προς θόρυβο. Οι παράμετροι προσομοίωσης δίνονται στον επόμενο πίνακα:

Parameter	Value
Carrier frequency (f_c)	5 GHz
FFT size	64
Number of used subcarriers	54
Sampling frequency (f_s)	3 MHz
Symbol part duration	$64 T_s = 21.33 \mu s$
Cyclic prefix duration	$T_{FFT} / 4 = 5.33 \mu s$
Modulation type	QPSK
Channel type	COST 207 TU

Πίνακας 2.2. παράμετροι προσομοίωσης OFDM. [144]

Η μετάδοση δεδομένων πάνω από ένα κανάλι με μοντέλο εξασθένησης COST 207 TU , έχοντας τις εξής καθυστερήσεις σημάτων στο κανάλι [0, 200, 600, 1600, 2400, 5000] ns , και ισχύς σημάτων είναι [-3 0 -2 -6 -8 -10] dB , ενώ ο τύπος της διαμόρφωσης που χρησιμοποιούμε είναι QPSK.

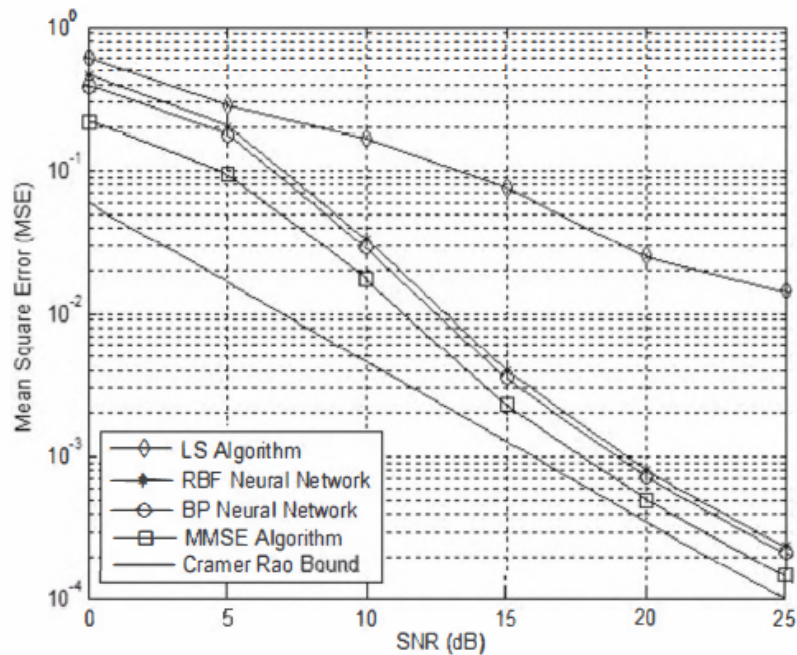
Το νευρωνικό δίκτυο MLP έχει ένα κρυφό επίπεδο και 10 νευρώνες , ενώ η συνάρτηση ενεργοποίησης είναι σιγμοειδής , και στο επίπεδο εξόδου γραμμική. Το νευρωνικό δίκτυο RBF έχει ένα κρυφό επίπεδο και 25 νευρώνες και συνάρτηση ενεργοποίησης Gauss. Το επόμενο σχήμα δείχνει το BER έναντι του SNR.



Σχήμα 2.32. Τιμές BER για την εκτίμηση καναλιού.[144]

Φαίνεται ότι ο αλγόριθμος LS δεν μπορεί να κάνει καλή εκτίμηση του καναλιού , ως αποτέλεσμα να έχει το χειρότερο BER από τους υπόλοιπους. Ωστόσο σε υψηλότερες τιμές SNR το BER είναι ικανοποιητικό. Αντίθετα ο αλγόριθμος MLP έχει καλή απόδοση τόσο στο χαμηλό SNR όσο και στις υψηλές τιμές του SNR, και είναι καλύτερος από τον LS. Πιο συγκεκριμένα στις τιμές του SNR 20 και 25 dB οι διαφορές στο BER μεταξύ των δύο αυτών αλγορίθμων είναι περισσότερο από 10^{-1} . Από το σχήμα μπορεί επίσης να φανεί ότι το νευρωνικό δίκτυο MLP έχει καλύτερη απόδοση από το RBF , το BER του BP είναι πολύ κοντά στο MMSE , ενώ σε αυξημένες τιμές SNR έχουν το ίδιο BER. Ωστόσο ο αλγόριθμος MMSE χρειάζεται στατιστικά στοιχεία καναλιού και πληροφορίες για τον θόρυβο , που είναι πρόκληση για τις πραγματικές μεταδόσεις , ενώ το νευρωνικό δίκτυο έχει μη αναγκαίες πληροφορίες για τα στατιστικά του καναλιού που είναι απαραίτητες για τον αλγόριθμο MMSE.

Στο επόμενο σχήμα φαίνεται το μέσο τετράγωνο σφάλμα (MSE) έναντι του SNR. Το σφάλμα εκτίμησης καναλιού του αλγορίθμου LS είναι περισσότερο από τους υπόλοιπους για κάθε τιμή του SNR. Μετά την τιμή του SNR στα 15 dB το MSE του εκτιμητή μας είναι κοντά σε αυτό του αλγορίθμου MMSE.



Σχήμα 2.33. τιμές MSE για εκτίμηση καναλιού. [144]

➤ Συμπεράσματα

Σε αυτήν την ενότητα προτείνεται μια μέθοδος εκτίμησης καναλιού που βασίζεται στο νευρωνικό δίκτυο MLP για συστήματα OFDM. Αφού τα δίκτυα εκπαιδεύονται χρησιμοποιώντας αποκρίσεις καναλιού που αποκτήθηκαν με τη βοήθεια πιλοτικών συμβόλων, αυτά τα εκπαιδευμένα δίκτυα χρησιμοποιούνται ως εκτιμητές καναλιών. Καθώς τα δίκτυα είναι εκπαιδευμένα, η πιλοτική μετάδοση δεν χρειάζεται να είναι διαφορετική από άλλους αλγόριθμους που λαμβάνουν CIR με βάση πιλοτικά σήματα. Έτσι το εύρος ζώνης χρησιμοποιείται αποτελεσματικότερα. Το νευρικό δίκτυο MLP είναι καλύτερο από τον αλγόριθμο LS και το νευρικό δίκτυο RBF σε σχέση με τα κριτήρια BER και MSE. Το MLP δίκτυο φαίνεται να έχει απόδοση τόσο καλή όσο και την περίπτωση μετάδοσης με τέλεια αποκρίση καναλιού. Θα μπορούσαμε να πούμε ότι είναι ένας κατάλληλος εκτιμητής για το OFDM, καθώς η απόδοσή του είναι καλύτερη από το νευρωνικό δίκτυο LS και RBF που είναι κοινός αλγόριθμος για το κανάλι εκτίμηση σε OFDM, ενώ ακόμα δεν υπάρχει ανάγκη για στατιστικά στοιχεία καναλιού, πληροφορίες θορύβου και αποστολή πιλοτικών συμβόλων που χρησιμοποιούνται από τον αλγόριθμο MMSE.

2.8.2 Νευρωνικό δίκτυο σε MIMO OFDM

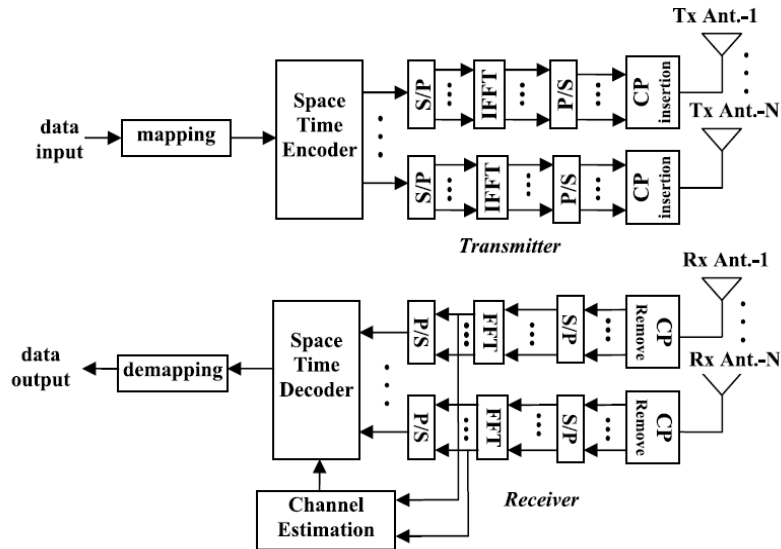
Σε αυτήν τη ενότητα , θα δούμε ένα νευρωνικό δίκτυο πολλαπλών επιπέδων πρόσθιας [147] τροφοδοσίας (MLP) εκπαιδευμένο με τον αλγόριθμο Levenberg – Marquardt (LM) για την εκτίμηση των παραμέτρων καναλιού σε συστήματα MIMO-OFDM.[148][149] Οι επιδόσεις του BER και του MSE των αλγορίθμων LS και LMSE (least mean square error) συγκρίνονται με το νευρωνικό δίκτυο . Ο εκτιμητής καναλιών νευρικού δικτύου έχει πολύ καλύτερη απόδοση από τους αλγόριθμους LS και LMS. Επιπλέον, δεν χρειάζεται στατιστικά στοιχεία καναλιού και αποστολή πιλοτικών τόνων, σε αντίθεση με τους κλασικούς αλγόριθμους.

Οι προσδοκίες από τη σημερινή και την επόμενη γενιά ασύρματων επικοινωνιών είναι η υψηλής ταχύτητας μετάδοση δεδομένων, η μεγάλη χωρητικότητα συστήματος και ποιότητα υπηρεσίας. Για να πετύχουμε αυτούς τους σκοπούς χωρίς την επιπλέον κατανάλωση φάσματος και ισχύς μετάδοσης η τεχνική διαμόρφωσης OFDM συνδυάστηκε με συστήματα MIMO , χωρική πολυπλεξίας (SP-Spatial Multiplexing) [151] και χωροχρονική κωδικοποίηση (STBC-space time block coded).[150]. Στην πρώτη τεχνική ο πομπός χωρίζει την ακολουθία συμβόλων προς μετάδοση σε N_t υποακολουθίες και τις μεταδίδει ξεχωριστά από την κεραία του έχοντας ως αποτέλεσμα να αυξηθεί ο ρυθμός των συμβόλων ενώ ο δέκτης θα πρέπει να αποπολυμπλέξει αυτή την ακολουθία. Στην δεύτερη τεχνική (STBC) ο πομπός κάνει κωδικοποίηση του καναλιού πάνω από την αρχική ακολουθία και παράγει N_t ακολουθίες προς μετάδοση, στην συνέχεια ο δέκτης πρέπει να επεξεργαστεί τις ακολουθίες λαμβάνοντας υπόψη ότι η αρχική ακολουθία έχει φτάσει σε αυτών σε N_t αντίγραφα συσχετισμένα. Στην τεχνική SP αυξάνεται ο ρυθμός μετάδοσης συμβόλων , ενώ στην τεχνική STBC παραμένει ίδιος βελτιώνεται όμως το SNR και κατά συνέπεια τα σφάλματα.

Στην πλευρά του δέκτη η απόκριση καναλιού πρέπει να εκτιμηθεί για την αποδιαμόρφωση του MIMO-OFDM σήματος από τον εκτιμητή καναλιού , αλλιώς είναι αδύνατη η λήψη του σήματος. Για την εκτίμηση της απόκρισης του καναλιού σε κάθε υποφέρων χρησιμοποιούνται αλγόριθμοι εκτίμησης καναλιού που βασίζονται σε πιλοτικά κανάλια όπως ο LS και ο LMS. Η εφαρμογή αυτών των αλγορίθμων έχει χρησιμοποιηθεί ευρέως και είναι αρκετά εύκολη αλλά έχει κακή απόδοση σε χρονομεταβλητά κανάλια. Για να ξεπεραστούν αυτά τα προβλήματα άλλες μέθοδοι όπως νευρωνικά δίκτυα , ασαφής λογική καθώς και συνδυασμός αυτών έχουν προταθεί και εφαρμοστεί για την εκτίμηση καναλιού τόσο SISO όσο και σε MIMO OFDM συστήματα επειδή έχουν ισχυρή προσέγγιση αλλά και μαθησιακή ικανότητα.

➤ Σύστημα MIMO OFDM (STBC)

Το επόμενο σχήμα παρουσιάζει ένα χωροχρονικό σύστημα MIMO OFDM (space time block coded MIMO-OFDM).



Σχήμα 2.34. MIMO-OFDM σύστημα [153]

Το σύστημα έχει N υποφέροντα N_T πομπούς και N_R δέκτες.

Στον πομπό, η ροή σειριακών δεδομένων αντιστοιχίζεται λαμβάνοντας υπόψη τον τύπο διαμόρφωσης, ενώ εισάγονται πιλοτικά σήματα για την λήψη CIR, στην συνέχεια τα μπλοκ δεδομένων $[X(2k), X(2k+1)]$ κωδικοποιούνται από τον κωδικοποιητή χωροχρόνου (space time block encoder) ως εξής : [154]

$$\begin{pmatrix} S_1(m) & S_1(m+1) \\ S_2(m) & S_2(m+1) \end{pmatrix} = \begin{pmatrix} X(2k) & -X^*(2k+1) \\ X(2k+1) & X^*(2k) \end{pmatrix}$$

όπου $S_j(m)$ είναι το σήμα που εκπέμπεται από την J -οστή κεραία στο m -οστό time slot και (*) είναι ο συζυγής μιγαδικός. Για την μετάδοση του σήματος από την i -οστή κεραία, το σήμα μετατρέπεται από σειριακό σε παράλληλο και διαμορφώνεται σε σύμβολα OFDM εφαρμόζοντας IFFT ως :

$$\begin{aligned} s_i(n) &= IFFT\{S_i(k)\} \\ &= \sum_{k=0}^{N-1} S_i(k)e^{j(2\pi kn/N)}, \quad n = 0, 1, \dots, N-1 \end{aligned}$$

όπου $S_i(k)$ είναι το χωροχρονικό διαμορφωμένο σήμα (space time block coded signal) το οποίο εκπέμπεται από την i -οστή κεραία στο k -οστό υποφέρον. Στην συνέχεια το κυκλικό πρόθεμα εισάγεται για την αποφυγή διασυμβολικής παρεμβολής, και το τελικό σήματα οδηγείται στην κεραία-πομπό. Στον j -οστό δέκτη, το λαμβανόμενο σήμα είναι :

$$y_j(n) = s_1(n) \otimes h_{j1}(n) + s_2(n) \otimes h_{j2}(n) + n_j(n)$$

όπου $h_{ij}(n)$ είναι η απόκριση του καναλιού στο πεδίο του χρόνου μεταξύ του i -οστού πομπού και του j -οστού δέκτη ενώ $n_j(n)$ είναι λευκός Gaussian θόρυβος. Μετά την αφαίρεση του κυκλικού

προθέματος στο δέκτη το σήμα μετατρέπεται από σειριακό σε παράλληλο και ο FFT είναι (εξίσωση 1):

$$Y_j(k) = FFT\{y_j(n)\} \\ = \frac{1}{N} \sum_{n=0}^{N-1} y_j(n) e^{-j(2\pi mn/N)}, \quad k = 0, 1, \dots, N-1$$

ενώ στην συνέχεια τα πιλοτικά σύμβολα εξαγονται.

Στο μπλοκ της εκτίμησης του καναλιού στο πεδίο της συχνότητας εκτιμάται το $H_{ij}(k)$ και τα μεταδιδόμενα δεδομένα λαμβάνονται ως εξής (εξίσωση 2 & 3):

$$X_j(2k) = \frac{H_{j1}^*(k)Y_j(2k) + H_{j2}(k)Y_{j2}^*(2k+1)}{|H_{j1}(k)|^2 + |H_{j2}(k)|^2} \\ X_j(2k+1) = \frac{H_{j2}^*(k)Y_j(2k) - H_{j1}(k)Y_j^*(2k+1)}{|H_{j1}(k)|^2 + |H_{j2}(k)|^2}$$

Τέλος, τα δεδομένα δυαδικών πληροφοριών λαμβάνονται μετά την αποδιαμόρφωση.

➤ Νευρωνικό δίκτυο πολλαπλών στρωμάτων (Multilayer perceptron)

Το τεχνητό νευρικό δίκτυο αποτελείται από νευρώνες (μη γραμμικά μπλοκ) που συνδέονται μεταξύ τους με παράλληλα συναπτικά βάρη. Το νευρικό δίκτυο έχει μαθησιακή ικανότητα που σημαίνει ότι τα συναπτικά βάρη προσαρμόζονται με τον αλγόριθμο εκμάθησης, έτσι ώστε το νευρικό δίκτυο να αντιδρά σε μια δεδομένη είσοδο για μια επιθυμητή έξοδο. Ένας από τους συνήθεις χρησιμοποιούμενους τύπους νευρωνικών δικτύων είναι το δίκτυο πρόσθιας τροφοδότης πολλαπλών στρωμάτων (feed-forward multilayered perceptron-MLP). Το MLP αποτελείται από στρώματα εισόδου, κρυφά στρώματα αλλά και στρώματα εξόδου όπου οι νευρώνες συνδέονται μεταξύ τους. Ο αλγόριθμος μάθησης που θα χρησιμοποιήσουμε για την εκπαίδευση του δικτύου είναι ο Levenberg–Marquardt.

- Αλγόριθμος Levenberg–Marquardt (LM)

Ο αλγόριθμος αυτός είναι επίσης γνωστός ως η μέθοδος με τα χαμηλά τετράγωνα [139], χρησιμοποιείται για την επίλυση προβλημάτων μη γραμμικών ελαχίστων τετραγώνων με τη βοήθεια του πίνακα-αλγορίθμου Hessian.[146] Ο αλγόριθμος εκπαίδευσης LM, για να προσαρμόσει τα συνολικά βάρη της είσοδο στο κρυφό στρώμα και από κρυμμένο στρώμα στην έξοδο, η συνάρτηση $E(w)$ που θα τα ελαχιστοποιήσει είναι:

$$E(w) = \frac{1}{2} \sum_{k=1}^c (d_k - o_k)^2$$

όπου d_k είναι η k-οστή επιθυμητή έξοδος, o_k είναι η k-οστή πραγματική έξοδος και c είναι ο αριθμός των σημείων εξόδου.

Προκειμένου να ελαχιστοποιηθεί η συνάρτηση $E(w)$, τα βάρη του νευρωνικού δικτύου ενημερώνονται ως εξής :

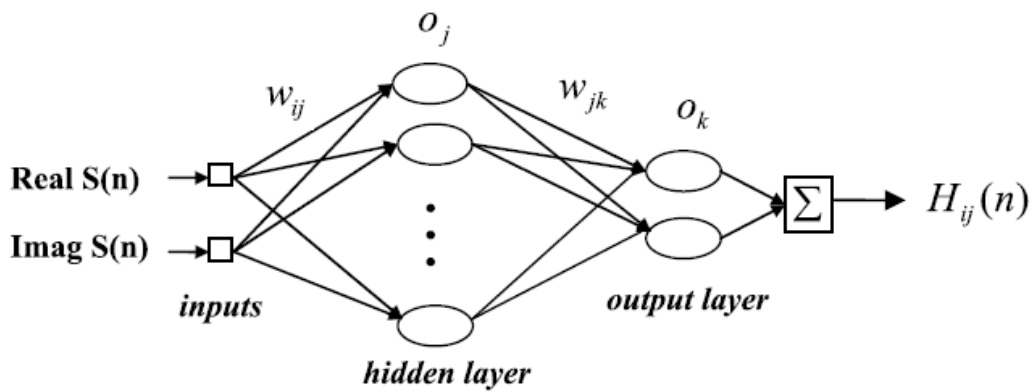
$$\Delta w = -[J^T J + \mu I]^{-1} J^T E$$

όπου j είναι ο πίνακας Jacobian [147] ο οποίος περιέχει την πρώτη παράγωγο των σφαλμάτων δικτύου σε σχέση με τα βάρη, μ είναι η παράμετρος-ρυθμός μάθησης που καθορίζει πόσο τα βάρη w μπορούν να αλλάξουν σε κάθε βήμα. Όταν χρησιμοποιείται υψηλότερο ποσοστό εκμάθησης (μέγιστο 1.0), το δίκτυο εκπαιδεύεται γρηγορότερα. Αν και με μεγαλύτερες τιμές μ επιταχύνουν τη διαδικασία εκπαίδευσης, μπορεί να προκαλέσουν 'ταλαντώσεις' που μπορεί να επιβραδύνουν τη σύγκλιση, ως αποτέλεσμα ο αλγόριθμος να αποκλίνει. Αντίθετα, εάν χρησιμοποιηθεί το μικρότερο ποσοστό εκμάθησης (ελάχιστο 0.0), ο αλγόριθμος θα πάρει πολύ χρόνο για να συγκλίνει. Για το λόγο αυτό συνήθως επιλέγεται μια ενδιάμεση τιμή (π.χ.0.05).

Βήματα εκπαίδευσης :[153]

1. Αρχικοποιήστε το βάρος και το ποσοστό εκμάθησης (παράμετρος μάθησης)
2. Υπολόγισε τη συνάρτηση που είναι το άθροισμα των τετραγώνων των σφαλμάτων μεταξύ των επιθυμητών και των πραγματικών εξόδων χρησιμοποιώντας την εξίσωση 1
3. Λύστε τον τύπο ενημέρωσης βαρών (εξίσωση 2)
4. Υπολογίστε ξανά το τετράγωνο σφάλμα χρησιμοποιώντας το $w+\Delta w$. Εάν το σφάλμα είναι μικρότερο από αυτό που υπολογίστηκε στο βήμα 2, τότε μειώστε το ποσοστό μάθησης (μ) κατά 0.1 φορές, εάν το σφάλμα δεν μειωθεί αυξήστε το ποσοστό μάθησης (μ) κατά 10 φορές και πηγαίνετε στο βήμα 3.
5. Ολοκληρώστε την εκπαίδευση όταν το σφάλμα είναι μικρότερο από την προκαθορισμένη τιμή.

Η δομή ενός νευρωνικού δικτύου για την εκτίμηση του καναλιού σε ένα σύστημα MIMO-OFDM φαίνεται στο επόμενο σχήμα.



Σχήμα 2.35. MLP νευρωνικό δίκτυο για εκτίμηση καναλιού MIMO-OFDM [153]

Σε αυτή την δομή έχουμε δύο επίπεδα MLP νευρωνικού δικτύου τα οποία έχουν ένα κρυφό και ένα επίπεδο εξόδου για την εκτίμηση της απόκρισης του καναλιού. Στο κρυφό στρώμα η συνάρτηση υπερβολικής εφαπτομένης με πέντε νευρώνες χρησιμοποιείται ενώ στο επίπεδο εξόδου χρησιμοποιείται η γραμμική συνάρτηση ενεργοποίησης. Η δομή δύο σημάτων εισόδου και δύο σημάτων εξόδου χρησιμοποιείται για την υιοθέτηση σημάτων OFDM στο νευρωνικό δίκτυο. Δεδομένου ότι τα σύμβολα OFDM αποτελούνται από μιγαδικά σήματα και το νευρωνικό δίκτυο χρησιμοποιεί πραγματικά σήματα, προκειμένου να υιοθετηθεί το νευρωνικό δίκτυο στο σύστημα OFDM, κάθε σήμα χωρίζεται σε πραγματικά και φανταστικά μέρη. Μετά την εκπαίδευση του δικτύου, κατά την διαδικασία επεξεργασία σήματος από τον εκτιμητή εισάγονται στο δίκτυο τα πραγματικά και τα φανταστικά μέρη του σήματος και οι εξοδοί του νευρωνικού δικτύου θα εκτιμήσουν την απόκριση του καναλιού (CIR). Οι μαθηματικές πράξεις που γίνονται στα επίπεδα του δικτύου κατά τη διάρκεια της διαδικασίας είναι οι εξής: [144] Τα μέρη σήματος εισόδου που χωρίζονται σε πραγματικό και φανταστικό και το άθροισμα των βαρών που εισέρχεται στους κόμβους υπολογίζονται εφαρμόζοντας τη συνάρτηση ενεργοποίησης. Εάν η συνάρτηση υπερβολικής εφαπτομένης χρησιμοποιείται στο πρώτο κρυφό στρώμα, η συνάρτηση ενεργοποίησης υπολογίζεται ως :

$$net_j = \sum_{i=0}^{L_1} S_i w_{ij}$$

$$o_j = f(net_j) = \frac{e^{2net_j} - 1}{e^{2net_j} + 1}$$

όπου w_{ij} είναι το βάρος της εισαγωγής στο κρυφό επίπεδο στο j -οστό κόμβο και L_1 είναι ο αριθμός των νευρώνων εισόδου. Στη συνέχεια στο επίπεδο εξόδου, για κάθε κόμβο εξόδου η συνάρτηση ενεργοποίησης υπολογίζεται ως εξής:

$$net_k = \sum_{j=1}^{L_2} o_j w_{jk}$$

$$o_k = f(net_k)$$

όπου w_{jk} είναι το βάρος του κρυφού στρώματος προς το επίπεδο εξόδου στον κ-οστό κόμβο και L_2 είναι ο αριθμός των κόμβων των κρυφών επιπέδων, μία από τις εξόδους του νευρωνικού εκτιμητή είναι :

$$o_k = f(net_k) = f\left(\sum_{j=0}^{L_2} w_{jk} f\left(\sum_{i=0}^{L_1} w_{ij} f(w_{ij} S_i)\right)\right)$$

Προσθέτοντας κάθε τμήμα των εξόδων, θα εκτιμηθεί η απόκριση καναλιού $H_{ij}(n)$.

➤ Προσομοίωση και αποτελέσματα

Σε αυτό το σημείο θα αναφερθούμε σε κάποια αποτελέσματα προσομοιώσεων που έγιναν με το παραπάνω σύστημα.[144] Η προσομοίωση που εκτελέστηκε είναι για συστήματα 2×2 και 4×4 MIMO-OFDM. Η απόδοση του νευρωνικού δικτύου συγκρίνεται με τον αλγόριθμο LS και LMS που βασίζονται σε πιλοτικά κανάλια χρησιμοποιώντας το BER και μέσο τετραγωνικό σφάλμα (mean square error -MSE) έναντι κριτηρίων SNR. Η επίδραση της χωροχρονικής κωδικοποίησης εξετάστηκε σε προσομοιώσεις λαμβάνοντας υπόψη το μη κωδικοποιημένο σύστημα MIMO-OFDM με βάση την εκτίμηση καναλιών από νευρικό δίκτυο. Σε προσομοιώσεις χρησιμοποιούνται πιλοτικά σύμβολα τύπου μπλοκ για την εκτίμηση των CIR. Οι παράμετροι του συστήματος MIMO-OFDM που χρησιμοποιήσαμε στις προσομοιώσεις μας φαίνονται στον επόμενο πίνακα:

Parameter	Value
Carrier frequency (f_c)	5 GHz
Sampling frequency (f_s)	3 MHz
FFT size	64
Symbol part duration	$64T_s = 21.33 \mu s$
Cyclic prefix duration	$T_{FFT}/4 = 5.33 \mu s$
Modulation type	QPSK
Channel type	COST 207 TU
Type of neural network	Feed-forward MLP
Number of neuron (hidden layer)	5

Πίνακας 2.3. παράμετροι συστήματος προσομοίωσης

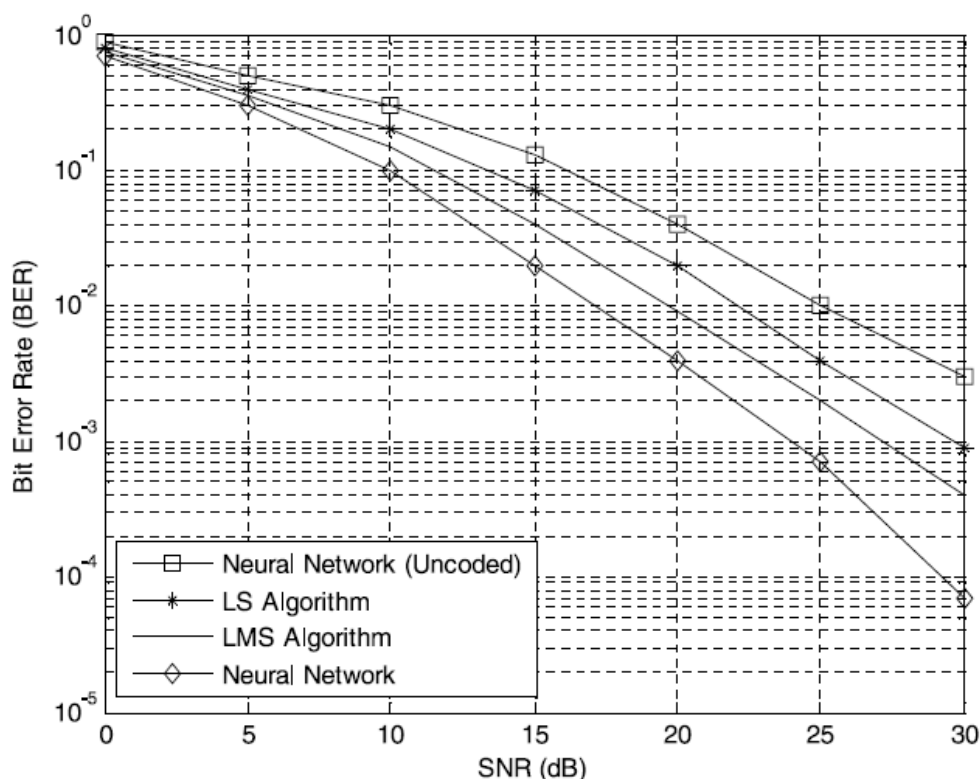
Το μοντέλο καναλιού COST 207 που χρησιμοποιείται έχει τα εξής χαρακτηριστικά:
[0, 200, 600, 1600, 2400, 5000] ns relative delays

[-3, 0, -2, -6, -8, -10] dB power paths

Στατικό στατικό περιβάλλον, υποθέτοντας ότι δεν υπήρχαν αλλαγές στο Doppler.

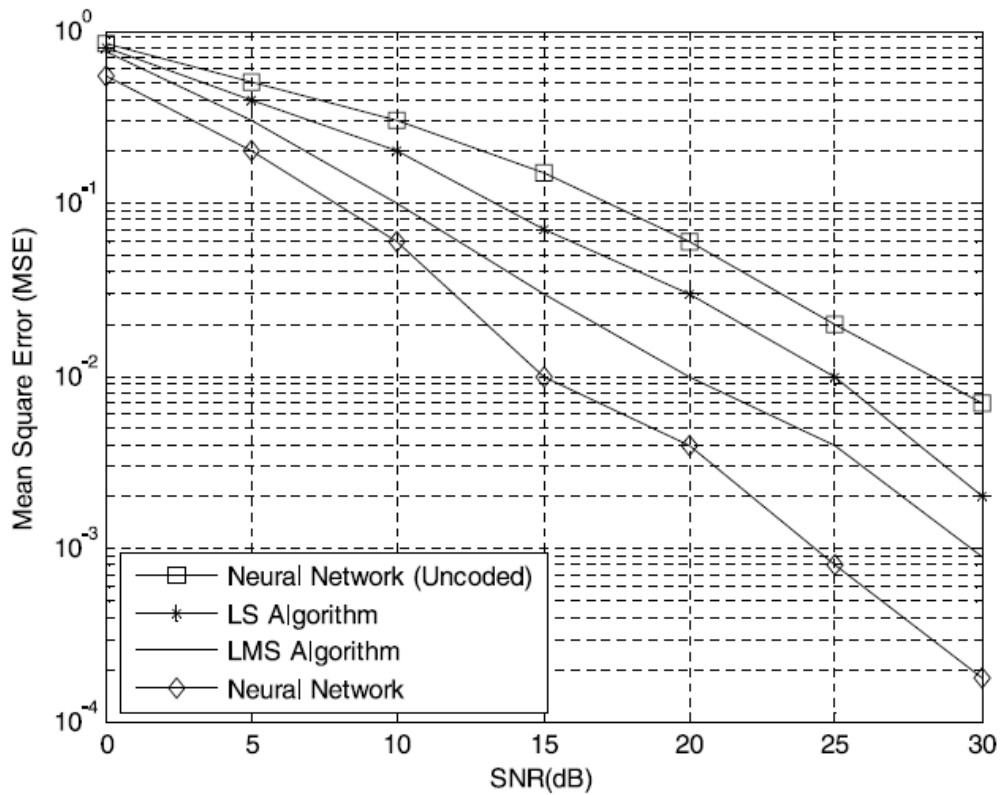
Στην εκπαιδευτική διαδικασία του νευρικού δικτύου χρησιμοποιούνται 500 εποχές-κύκλοι.

Το επόμενο δείχνει την απόδοση του ρυθμού σφάλματος bit (BER) των εκτιμητών καναλιού σε σχέση με το SNR για σύστημα 2x2.



Σχήμα 2.36. BER vs SNR για MIMO 2x2[153]

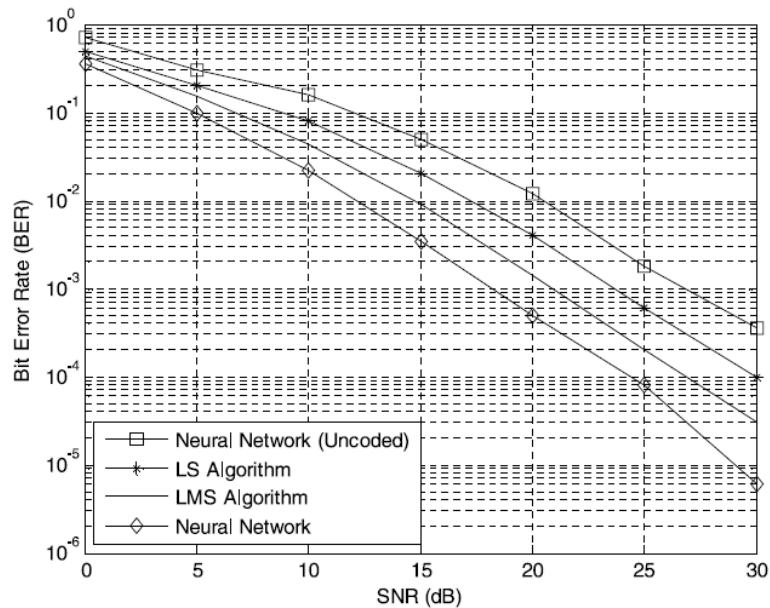
Όπως φαίνεται από το σχήμα, τόσο σε υψηλές τιμές SNR όσο και σε χαμηλές τιμές SNR, η απόδοση BER του νευρικού εκτιμητή είναι καλύτερη από τον αλγόριθμο LS και τον αλγόριθμο LMS. Επιπλέον, σε υψηλότερες τιμές του SNR, το BER του νευρικού δικτύου γίνεται καλύτερο όχι μόνο τον αλγόριθμο LS αλλά και από τον αλγόριθμο LMS. Στην τιμή SNR 20 dB, η διαφορά BER του LS και του νευρικού δικτύου είναι κοντά στα 10^{-1} και στην τιμή SNR 30 dB η διαφορά είναι μεγαλύτερη από 10^{-1} ενώ στα 30 dB SNR η διαφορά του BER μεταξύ του κωδικοποιημένου χωροχρονικού συστήματος για την εκτίμησης καναλιού με νευρωνικό δίκτυο και του μη κωδικοποιημένου συστήματος είναι σχεδόν 10^{-2} . Αν και η εφαρμογή του αλγορίθμου LS είναι αρκετά πιο εύκολη σε σχέση με τον αλγόριθμο νευρωνικών δικτύων και τον LMS, η απόδοση του αλγορίθμου LS είναι ανεπαρκής. Εκτός αυτού, το σχήμα δείχνει ότι σημαντικά οφέλη μπορούν να επιτευχθούν με χωροχρονική κωδικοποίηση. Λαμβάνοντας υπόψη τα αποτελέσματα της προσομοίωσης, μπορούμε να πούμε ότι μπορεί να χρησιμοποιηθεί χωροχρονική κωδικοποίηση για να βελτιωθεί η απόδοση των συστημάτων MIMO-OFDM. Επίσης, η απόδοση των εκτιμητών αξιολογείται με μέσο τετράγωνο σφάλμα (MSE) σε σχέση με το SNR και τα αποτελέσματα φαίνονται στο επόμενο σχήμα.



Σχήμα 2.37. MSE vs SNR για σύστημα 2x2.[153]

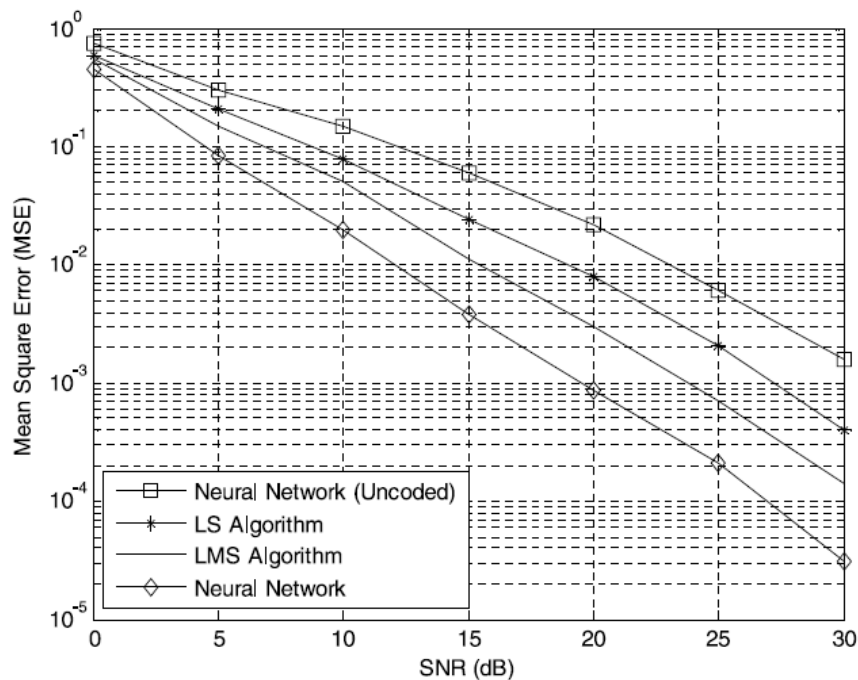
Το σφάλμα εκτίμησης του εκτιμητή νευρωνικού δικτύου είναι μικρότερο από τον αλγόριθμο LS και τον αλγόριθμο LMS για κάθε τιμή SNR. Από το σχήμα φαίνεται ότι ο αλγόριθμος LS απαιτεί 25 dB, ο αλγόριθμος LMS απαιτεί 20 dB ενώ το νευρωνικό δίκτυο απαιτεί 15 dB για να έχει $MSE = 10^{-2}$. Ένα σύστημα με κωδικοποίηση δείχνει περίπου 13,5 dB πλεονεκτήματα σε σχέση με τον νευρωνικό εκτιμητή χωρίς κωδικοποίηση και BER 10^{-2} , προσφέροντας έτσι μια μεγάλη βελτίωση SNR σε σύγκριση με τους άλλους εκτιμητές.

Ο εκτιμητής καναλιού νευρικών δικτύου προσομοιώθηκε και σε σύστημα 4x4 MIMO-OFDM. Στο επόμενο σχήμα φαίνεται ότι το μη κωδικοποιημένο νευρωνικό δίκτυο χρειάζεται 27,5 dB SNR, ο αλγόριθμος LS χρειάζεται 23,5 dB SNR και ο αλγόριθμος LMS χρειάζεται 21 dB SNR ενώ το κωδικοποιημένο νευρωνικό δίκτυο χρειάζεται 18 dB SNR για να φτάσει στο BER = 10^{-3} .



Σχήμα 2.38. BER vs SNR για σύστημα 4x4 [153]

Σύμφωνα με αυτό το σχήμα, το κωδικοποιημένο νευρωνικό δίκτυο χρειάζεται χαμηλότερες τιμές SNR σε σύγκριση με τους άλλους εκτιμητές ώστε να φτάσει το μεγάλο ποσό των τιμών BER. Στο επόμενο σχήμα φαίνεται ότι η διαφορά του SNR μεταξύ κωδικοποιημένου νευρικού δικτύου και αλγορίθμου LMS είναι 4 dB για $MSE 10^{-3}$.



Σχήμα 2.39. MSE vs SNR για σύστημα 4x4 [153]

Η διαφορά μεταξύ του κωδικοποιημένου νευρικού δικτύου και του αλγορίθμου LS είναι περίπου 7,5 dB για $MSE 10^{-3}$, ως αποτέλεσμα ο εκτιμητής καναλιού με νευρικό δίκτυο να έχει καλύτερη απόδοση από τους άλλους εκτιμητές για συστήματα MIMO-OFDM τόσο για συστήματα 2×2 όσο και για συστήματα 4×4 .

➤ Συμπεράσματα

Σε αυτήν την ενότητα είδαμε μια νέα τεχνική εκτίμησης καναλιού που βασίζεται σε νευρωνικά δίκτυα MLP για το σύστημα MIMO-OFDM με χωροχρονική κωδικοποίηση. Τα νευρωνικά δίκτυα MLP χρησιμοποιήθηκαν ως εκτιμητές καναλιών αφού τα δίκτυα εκπαιδεύτηκαν με σύνολα που αποτελούνται από σωστές αποκρίσεις καναλιών. Για την αξιολόγηση των επιδόσεων, τα συγκρίναμε με τους αλγόριθμους εκτίμησης καναλιών LS και LMS που βασίζονται σε πιλοτικούς τόνους από προσομοιώσεις υπολογιστών. Σύμφωνα με τα αποτελέσματα της προσομοίωσης, η απόδοση του νευρικού δικτύου ήταν καλύτερη από τους αλγόριθμους LS και LMS. Επιπλέον, η μη αναγκαιότητα αποστολής πιλοτικών τόνων ήταν ένα άλλο πλεονέκτημα του εκτιμητή νευρωνικών δικτύων.

2.8.3 Νευρωνικό δίκτυο για OFDM χωρίς CP

Η τεχνική OFDM χρησιμοποιείται ευρέως σε συστήματα ασύρματων επικοινωνιών για μετάδοση δεδομένων υψηλού ρυθμού, ενώ έχει συνδυαστεί με συστήματα MIMO για ακόμα μεγαλύτερη απόδοση. Το multipath του ασύρματου καναλιού προκαλεί διασυμβολική παρεμβολή (inter-symbol interference-ISI) σε συστήματα OFDM. Η προσθήκη κυκλικού προθέματος (cyclic prefix-CP) μπορεί αποτελεσματικά να ξεπεράσει το πρόβλημα της διασυμβολικής παρεμβολής για συχνοεπιλεκτικά κανάλια και multipath. Ωστόσο, η εισαγωγή CP προκαλεί μείωση της φασματικής απόδοσης.

Σε αυτή την ενότητα, θα δούμε μια μετάδοση OFDM χωρίς CP για σύστημα SISO. Το μεταδιδόμενο σήμα OFDM δεν έχει CP για υψηλότερη απόδοση φάσματος. Ωστόσο, το μεγάλο ISI υποβαθμίζει την απόδοση της μετάδοσης. Στον δέκτη, χρησιμοποιείται ένα βαθύ νευρωνικό δίκτυο για την εκτίμηση του CP για τα μεταδιδόμενα σήματα OFDM στο πεδίο του χρόνου. Τα δεδομένα για την εκπαίδευση του δικτύου δημιουργήθηκαν αναλύοντας σήματα OFDM στο πεδίο του χρόνου ώστε να ακυρώσουν το ISI. Το πρόβλημα της ακύρωσης παρεμβολών το θεωρούμε ως πρόβλημα παλινδρόμησης (multi-regression).

➤ Μετάδοση OFDM χωρίς CP

Έστω N ο συνολικός αριθμός των υποφερόντων (subcarriers) και K ο αριθμός των υποφερόντων που χρησιμοποιούνται. Το εκπεμπόμενο σήμα στο n -οστό δείγμα (sample) του m -οστού OFDM συμβόλου είναι:

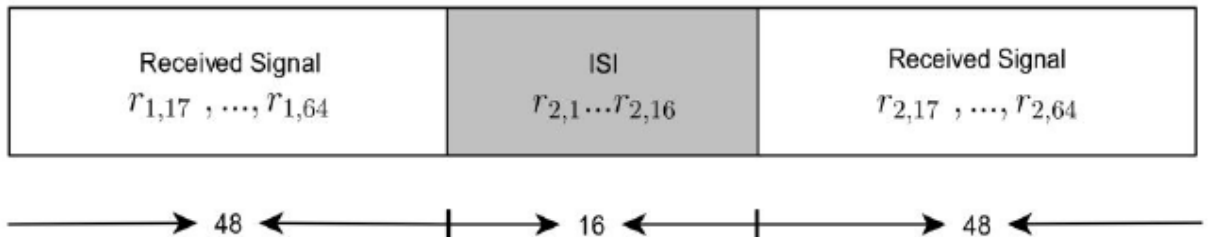
$$S_{m,n} = \frac{1}{\sqrt{N}} \sum_{k=-K/2}^{K/2} S_m(k) e^{j2\pi nk/N}$$

όπου $S_m(k)$ είναι το σύμβολο της διαμόρφωσης QPSK για το k -οστό υποφέρων του m -οστού OFDM συμβόλου. Το λαμβανόμενο σήμα είναι :

$$r_{m,n} = S_{m,n} \otimes h_1$$

όπου h_1 για $l=1, \dots, N_L$ είναι η απόκριση του καναλιού και \otimes είναι η συνέλιξη, ενώ υποθέτουμε ότι δεν υπάρχει θόρυβος.

Στον δέκτη η ISI υπάρχει στο σήμα λήψης OFDM όπως φαίνεται στο επόμενο σχήμα, όπου $m=1,2$, $N_L = 16$ και $N = 64$.



Σχήμα 2.40. OFDM χωρίς CP [157]

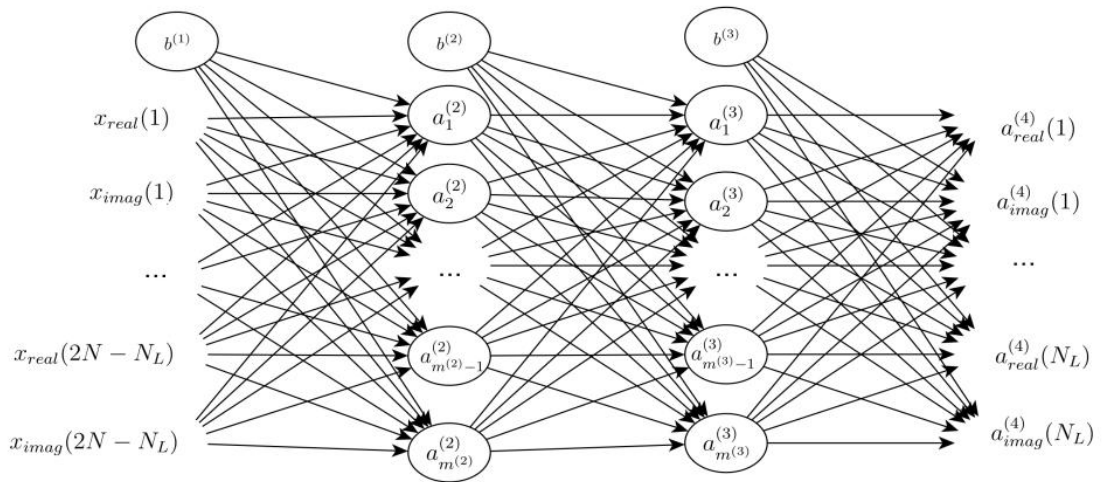
Χρησιμοποιούμε το διάνυσμα $r = [r(1), \dots, r(2N - N_L)]^T$ για την ακύρωση ISI, όπου $r(i) = r_{1, N_L+i}$ για $1 \leq i \leq N_L$ είναι το λαμβανόμενο σήμα από το πρώτο OFDM σύμβολο, $r(i + N + N_L)$ για $1 \leq i \leq N_L$ είναι το σήμα ISI και $r(i + N)$ για $1 \leq i \leq N - N_L$ είναι το λαμβανόμενο σήμα για το δεύτερο OFDM σύμβολο. Μετά την ακύρωση του ISI με την χρήση του νευρωνικού δικτύου, γίνεται αποδιαμόρφωση συμβόλου με maximum-likelihood.

- Μοντέλο νευρωνικού δικτύου για παλινδρόμηση

Θα χρησιμοποιήσουμε ένα μοντέλο νευρωνικού δικτύου ώστε να ακυρώσουμε το ISI. Το σήμα εισόδου στο δίκτυο ορίζεται ως εξής :

$$x = [x_{real}(1), x_{imag}(1), \dots, x_{real}(2N - N_L), x_{imag}(2N - N_L)]^T$$

όπου $[x_{real}(i), x_{imag}(i)]$ για $1 \leq i \leq 2N - N_L$ είναι το πραγματικό και το φανταστικό μέρος του $r(i)$. Το επόμενο σχήμα δείχνει την αρχιτεκτονική ενός βαθύ νευρωνικού δικτύου το οποίο αποτελείται από δύο κρυφά στρώματα με μη γραμμικές συναρτήσεις ενεργοποίησης, συνάρτηση σιγμοειδούς στο πρώτο στρώμα και συνάρτηση υπερβολικής εφαπτομένης στο δεύτερο, ενώ τέλος έχει ένα άλλο επίπεδο εξόδου χωρίς συνάρτηση ενεργοποίησης για παλινδρόμηση.



Σχήμα 2.41. μοντέλο νευρωνικού δικτύου [157]

Η έξοδος είναι το σήμα ακύρωση του ISI και ορίζεται ως :

$$y = [y_{real}(1), y_{imag}(1), \dots, y_{real}(N_L), y_{imag}(N_L)]^T$$

όπου $[y_{real}(i), y_{imag}(i)]$ είναι το πραγματικό και το φανταστικό μέρος του σήματος ακύρωσης ISI. Καθε κύκλο-κόμβος σε κάθε επίπεδο ονομάζεται μονάδα, η οποία ορίζεται ως η είσοδος στο επόμενο επίπεδο. Ας υποθέσουμε ότι n_l είναι ο αριθμός των επιπέδων στο δίκτυο (εδώ 4). Οι απλοί νευρώνες συνδέονται μεταξύ τους έτσι ώστε η έξοδος του ενός νευρώνα να είναι είσοδος του άλλου. Στο κρυφό επίπεδο 1, χαρακτηρίζουμε την είσοδο ως $z^{(l)}$, και την έξοδο της κάθε μονάδας ως $a^{(l)}$. Η έξοδος της i -οστής μονάδας στο επίπεδο 1 χαρακτηρίζεται ως $a_i^{(l)}$. Όλη η ισχύς του σήματος για ένα επίπεδο αποθηκεύεται σε ένα πίνακα που δηλώνεται ως πίνακας βαρών $w^{(l-1)} = w_{ij}^{(l-1)}$ και το διάνυσμα bias $b^{(l-1)} = b_j^{(l-1)}$ όπου $j \in \{1 : m^l\}$ με m^l να υποδηλώνει τον αριθμό των κόμβων στο επίπεδο 1. Ο υπολογισμός του μοντέλου στο πρώτο κρυφό επίπεδο δίνεται από :

$$\begin{aligned} a_1^{(2)} &= f(W_{11}^{(1)} x_{real}(1) + W_{12}^{(1)} x_{imag}(1) + \dots + b_1^{(1)}) \\ a_2^{(2)} &= f(W_{21}^{(1)} x_{real}(1) + W_{22}^{(1)} x_{imag}(1) + \dots + b_2^{(1)}) \\ &\dots \\ a_{m^{(2)}}^{(2)} &= f(W_{m^{(2)}1}^{(1)} x_{real}(1) + W_{m^{(2)}2}^{(1)} x_{imag}(1) + \dots + b_{m^{(2)}}^{(1)}) \end{aligned}$$

Επιπλέον τον διάνυσμα $z^{(l)} = [z_j^{(l)}]$ να αντιπροσωπεύει το συνολικό βάρος της εισόδου στην μονάδα στο επίπεδο 1 συμπεριλαμβανομένου του όρους bias :

$$z_j^{(l)} = \sum_{j=1}^{m^{(l)}} W_{ij}^{(l-1)} a_j^{(l-1)} + b_j^{(l-1)}$$

$$a_j^{(l)} = f(z_j^{(l)})$$

όπου $f(\cdot)$ είναι η μη γραμμική συνάρτηση ενεργοποίησης .

Στο πρώτο κρυφό επίπεδο η συνάρτηση ενεργοποίησης που χρησιμοποιείται είναι η σιγμοειδής :

$$f(z_j^{(2)}) = \sigma(z_j^{(2)}) = \frac{1}{\exp(-z_j^{(2)}) + 1}$$

η οποία παίρνει ένα πραγματικό αριθμό μεταξύ 0 και 1.

Στο δεύτερο κρυφό επίπεδο η συνάρτηση ενεργοποίησης που χρησιμοποιείται είναι η υπερβολική εφαπτομένη που ορίζεται ως:

$$f(z_j^{(3)}) = \tanh(z_j^{(3)}) = \frac{\exp(2z_j^{(3)}) - 1}{\exp(2z_j^{(3)}) + 1}$$

και παίρνει τιμές μεταξύ -1 και 1 .

Στο τελευταίο επίπεδο προβλέπουμε 32 τιμές ως :

$$a_{real}^{(4)}(1) = W_{11}^{(3)} a_1^{(3)} + W_{12}^{(3)} a_2^{(3)} + \dots + b_1^{(3)}$$

$$a_{imag}^{(4)}(1) = W_{21}^{(3)} a_1^{(3)} + W_{22}^{(3)} a_2^{(3)} + \dots + b_2^{(3)}$$

...

$$a_{imag}^{(4)}(N_L) = W_{(2N_L)1}^{(3)} a_1^{(3)} + W_{(2N_L)2}^{(3)} a_2^{(3)} + \dots + b_{(2N_L)}^{(3)}$$

- Συνάρτηση κόστους:

Η συνάρτηση κόστους είναι το μέτρο του σφάλματος πρόβλεψης που υπάρχει στα δεδομένα εκπαίδευσης. Παρόλο που υπήρχαν πολλές μορφές συναρτήσεων κόστους , οι περισσότερες βασίζονται στην απόσταση μεταξύ της προβλεπόμενης τιμής και της πραγματική τιμής. Στη περίπτωση μας η συνάρτηση κόστους είναι το μέσο τετραγωνικό σφάλμα (mean square error) που υπολογίζεται ως :

$$L = \frac{1}{N_L} \sum_{i=1}^{N_L} (y_{real}(i) - a_{real}^{(4)}(i))^2 + (y_{imag}(i) - a_{imag}^{(4)}(i))^2$$

- Μέθοδος βαθμωτής κατάβασης

Για την εκπαίδευση του μοντέλου θα χρησιμοποιήθηκαν 500000 μεταδόσεις σήματος ,

χρησιμοποιώντας την μέθοδο βαθμωτής κατάβασης σε ομάδες (mini-batch gradient descent).[1][158] Αρχικά αρχικοποιούνται οι παράμετροι- πίνακες βαρών του νευρωνικού δικτύου με μια από τις μεθόδους , μηδέν , τυχαία , ομοιόμορφα κτλ. Στην συνέχεια , εκπαιδεύουμε το μοντέλο , λαμβάνοντα στην έξοδο τις προβλεπόμενες τιμές. Στόχος είναι η ελαχιστοποίηση της συνάρτησης κόστους L ενημερώνοντας κατάλληλο των πίνακα βαρών. Για να βελτιστοποιήσουμε τον πίνακα βαρών , χρησιμοποιείται η μέθοδος της βαθμωτής κατάβαση, υπολογίζοντας την κλίση της συνάρτησης κόστους σε σχέση με όλες τις παραμέτρους , δημιουργούμε δηλαδή ένα διάνυσμα το οποίο περιέχει την κλίση της συνάρτησης κόστους κατά μήκος κάθε άξονα. Οι επιλογές για την εκπαίδευση του αλγορίθμου περιλαμβάνουν τρεις μεθόδους :

1. ομάδες : χρήση ολόκληρων των δεδομένων εκπαίδευσης σε κάθε επανάληψη , το οποίο έχει μεγάλο υπολογιστικό κόστος
2. στοχαστικό : χρησιμοποιεί ένα σημείο για να μοντελοποιήσει το σφάλμα , το οποίο καθιστά την σύγκληση αργή και με μεγάλη διακύμανση
3. υποομάδες (Mini-batch): χρησιμοποιεί διάφορα μικρά τυχαία υποσύνολα του συνόλου δεδομένων κάθε φορά που εκπαιδεύουμε το δίκτυο , φαίνεται η πιο αποδοτική λύση για μεγάλα σύνολα δεδομένων , οι υποομάδες θα μπορούσαν για παράδειγμα να χωριστούν σε σύνολα 1000 ανάλογα το μέγεθος των δεδομένων ώστε να επιτευχθεί η καλύτερη απόδοση .

Για να υπολογιστεί η κλίση της συνάρτησης κόστους χρησιμοποιείται ο αλγόριθμος Backward Propagation. [148] Για το επίπεδο l , μπορούμε να γράψουμε το γραμμικό μέρος ως :

$z^{(l)} = w^{(l-1)}a^{(l-1)} + b^{(l-1)}$. Από τον υπολογισμό της παραγώγου $dz^{(l)} = \frac{\partial L}{\partial z^{(l)}}$ θα πάρουμε τρεις τιμές :

$$d\mathbf{W}^{(l-1)} = \frac{\partial L}{\partial \mathbf{W}^{(l)}} = \frac{1}{N_L} (\mathbf{a}^{(l-1)})^T d\mathbf{z}^{(l)}$$

$$d\mathbf{b}^{(l-1)} = \frac{\partial L}{\partial \mathbf{b}^{(l-1)}} = \frac{1}{N_L} \sum_{i=1}^{N_L} d\mathbf{z}^{(l)(i)}$$

$$d\mathbf{a}^{(l-1)} = \frac{\partial L}{\partial \mathbf{a}^{(l-1)}} = (\mathbf{W}^{(l-1)})^T d\mathbf{z}^{(l)}$$

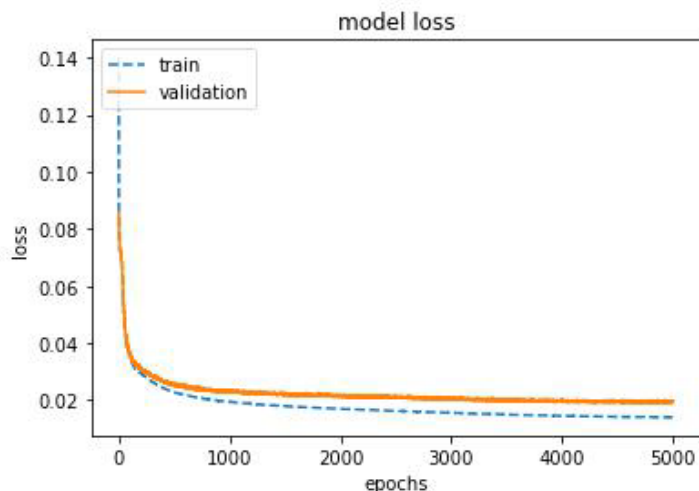
όπου a είναι η παράμετρος μάθησης.

➤ Απόδοση

Σε αυτή το σημείο θα δούμε την πειραματική απόδοση ενός νευρωνικού δικτύου για ένα σύστημα OFDM χωρίς CP. Το μέγεθος του FFT είναι N=64 , το μέγεθος των διαμορφωμένων υποφερώντων είναι K=52 και το μέγεθος του multipath είναι $N_L = 16$, ενώ έχουν παραχθεί 5000000 OFDM σήματα μετάδοσης για την προσομοίωση και την εκπαίδευση. Τα δεδομένα χωρίζονται σε τρία μέρη , δεδομένα εκπαίδευσης , επικύρωσης και δοκιμής. Το 80% των δεδομένων είναι για εκπαίδευση , 10% για επικύρωση και 10% για δοκιμή.

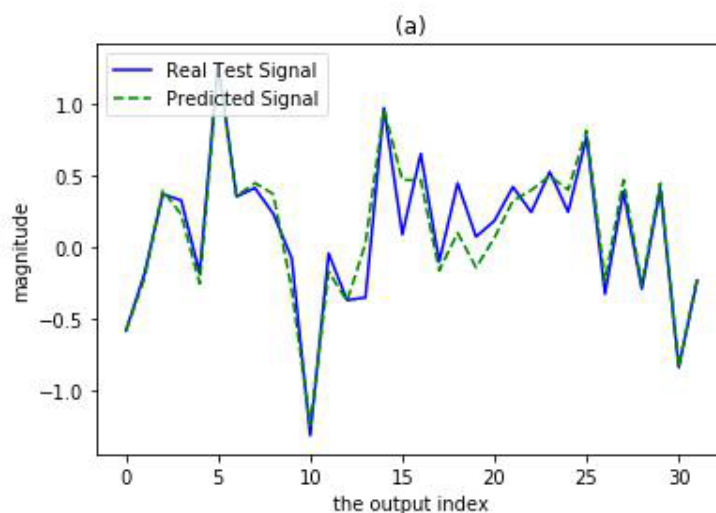
Στο επόμενο σχήμα απεικονίζει την απόδοση της συνάρτησης κόστους ανά τις εποχές σε σχέση

με το σύνολο δεδομένων εκπαίδευσης και την επικύρωση, ενώ η απώλεια τείνει να μειώνεται με τις εποχές.

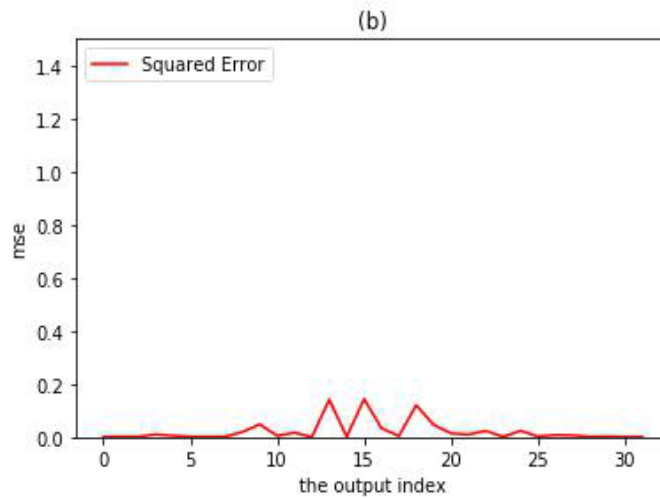


Σχήμα 2.42. συνάρτηση κόστους ανα εποχές.[157]

Τα επόμενα σχήματα απεικονίζουν τα αποτελέσματα από την προσομοίωση . Στο πρώτο σχήμα η μπλε γραμμή απεικονίζει τη σωστή έξοδο, η πράσινη γραμμή απεικονίζει την προβλεπόμενη τιμή και η κόκκινη γραμμή είναι το μέσο τετράγωνο σφάλμα (MSE).



Σχήμα 2.43. Προβλεπόμενο και πραγματικό σήμα ISI-cancelled [157]



Σχήμα 2.44. MSE μεταξύ προβλεπόμενου και πραγματικού σήματος ISI-cancelled [157]

Στο μοντέλο μας το μήκος του σήματος που χρειάζεται να προβλέψουμε είναι $N_L = 16$, με 16 πραγματικά και 16 φανταστικά μέρη. Από τα προηγούμενα σχήματα φαίνεται ότι η μπορεί να προβλεφθεί αποτελεσματικά το σήματα από τα δεδομένα εκπαίδευσης και δοκιμής. Τα συνολικά δεδομένα από τα αποτελέσματα της παλινδρόμησης φαίνονται στο επόμενο πίνακα.

MSE in training set	MSE in validation set	MSE in test set	SER (Symbol Error Rate)
0.0157	0.0176	0.0175	0.0088

Πίνακας 2.4 με τα αποτελέσματα για τα συνολικά δεδομένα.

Ο ρυθμός σφάλματος συμβόλου (Symbol error rate - SER) υπολογίζεται ως ο λόγος μεταξύ του αριθμού των συμβόλων σε σφάλμα και του συνολικού αριθμού των μεταδιδόμενων συμβόλων. Μετά την ακύρωση του ISI από το νευρωνικό δίκτυο το SER είναι 0.0088.

➤ Συμπεράσματα

Σε αυτή την ενότητα είδαμε ένα σύστημα OFDM χωρίς CP για την αποδοτικότερη χρήση φάσματος. Το ISI μπορεί να προβλεφθεί αποτελεσματικά με βαθύ νευρωνικό δίκτυο. Η συγκεκριμένη μέθοδος μπορεί να επεκταθεί και σε άλλα ασύρματα συστήματα ενώ μπορεί να χρησιμοποιηθεί σε συστήματα με θόρυβο αλλά και σε νέας γενιάς συστήματα MIMO.

2.9 Βαθιά μάθηση για MIMO-NOMA Downlink

➤ Βασικά στοιχεία NOMA

Η NOMA (Μη ορθογωνική πολλαπλή πρόσβαση - Non Orthogonal Multiple Access - NOMA)[162] αποτελεί βασική αρχή για τον σχεδιασμό τεχνικών ράδιο-πρόσβασης για τα 5G δίκτυα.[159][160][161] Η NOMA επιτρέπει στο κανάλι μιας συχνότητας να αποδίδεται σε πολλούς χρήστες την ίδια χρονική στιγμή στο ίδιο κελί προσφέροντας βελτιωμένη φασματική αποδοτικότητα, υψηλότερο ρυθμό-απόδοση μεταξύ κελιών, αποδοτικότερο κανάλι ανατροφοδότησης (όπου μόνο η λαμβανόμενη ισχύ σημάτων απαιτείται και όχι πληροφορίες σχετικά με την κατάσταση καναλιού (Channel State Information - CSI)), χαμηλή καθυστέρηση. Οι βασικές τεχνικές NOMA μπορούν να χωριστούν σε 2 βασικές κατηγορίες, την PD (Τομέα ισχύος - Power Domain – PD) και την CD (Τομέα κώδικα Code Domain – CD) πολυπλεξία. Η τεχνική PD υποστηρίζει πολλούς χρηστών, όπου χρησιμοποιεί έναν συγκεκριμένο χρήστη για να εξαπλώσει τις ακολουθίες, ενώ CD τεχνική μοιάζει με την τεχνική CDMA καθώς μοιράζεται όλους τους διαθέσιμους πόρους (χρόνου/ συχνότητας). Κύριος σκοπός της NOMA είναι να αντιμετωπίσει τα διάφορα προβλήματα που δεν επιλύονται με τις τεχνικές OMA (Ορθογωνική πολλαπλή πρόσβαση Orthogonal Multiple Access) όπως για παράδειγμα η τεχνική OFDM.

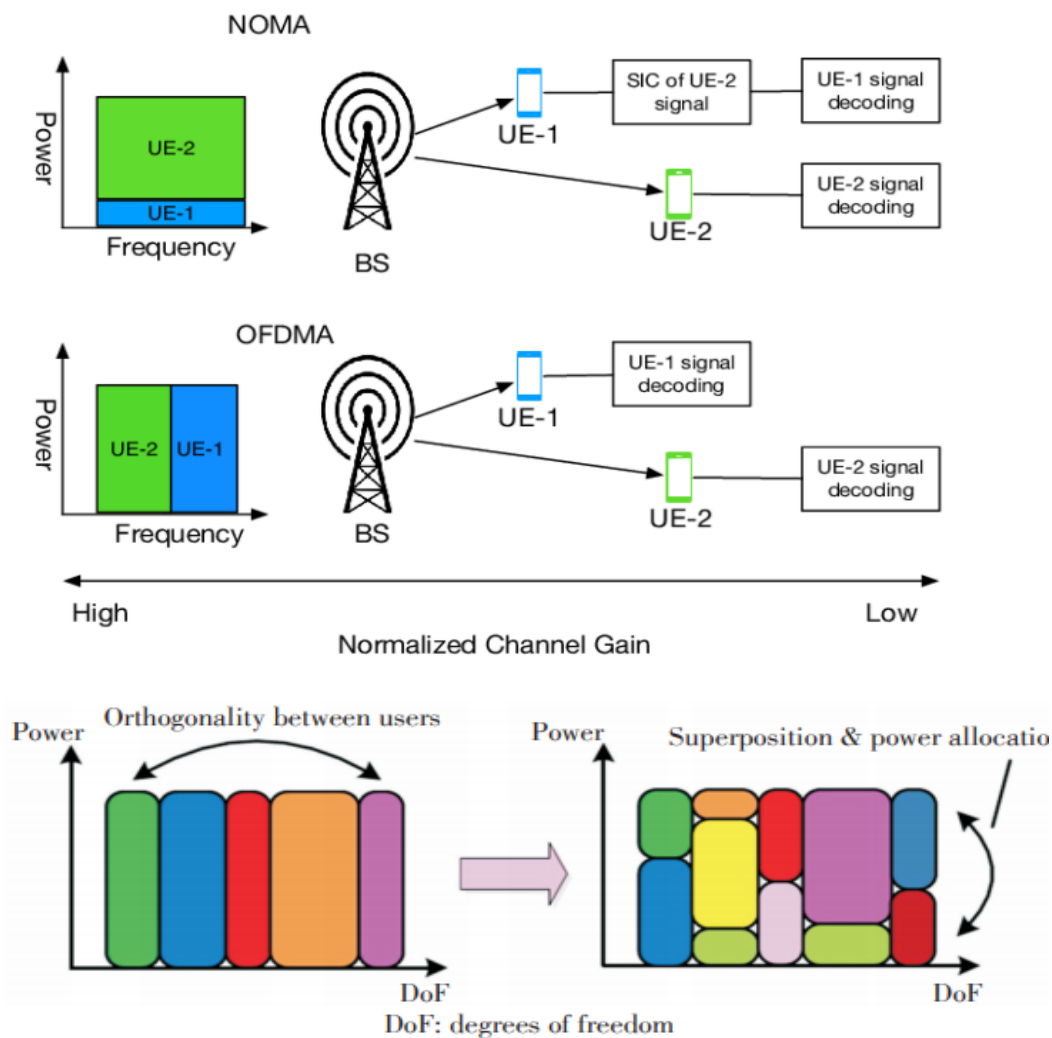
Πιο συγκεκριμένα:

PD πολυπλεξία: Σε διαφορετικούς χρήστες κατανέμονται διαφορετικά επίπεδα ισχύος ανάλογα με τις συνθήκες των καναλιών τους, με σκοπό την βέλτιστη απόδοση του συστήματος.

Χρησιμοποιώντας την τεχνική της διαδοχικής ακύρωσης παρεμβολών (Successive Interference Cancellation - SIC) μεταξύ των χρηστών, πραγματοποιείται και η κατανομή της ισχύος.

CD πολυπλεξία: Αυτή είναι όμοια με την CDMA και την πολλαπλών φερόντων CDMA (Multi Carrier CDMA - MC-CDMA), όπου διαφορετικοί χρήστες έχουν διαφορετικούς κώδικες, οι οποίοι πολυπλέκονται πάνω στους ίδιους πόρους (χρόνου/ συχνότητας).

Η διαφορά μεταξύ της PD και της CD πολυπλεξίας έγκειται στο γεγονός πως η CD πολυπλεξία μπορεί να επιτύχει κάποιο κέρδος εξάπλωσης (spreading gain) και διαμόρφωσης, ενώ και για τα δυο υπάρχει αύξηση στο εύρος ζώνης του σήματος. Απο τις πιο σημαντικές διαφορές μεταξύ OMA και NOMA αφορά αυτή της κατανομής πόρων, όπου η OMA κατανέμει τους πόρους αποκλειστικά για κάθε χρήστη περιορίζοντας έτσι τον αριθμό χρηστών ανάλογα τους πόρους, ενώ η NOMA διαθέτει τον ίδιο πόρο σε πολλούς χρήστες μέσω της PD τεχνικής, εκμεταλλευόμενη έτσι τα χαρακτηριστικά του καναλιού, το οποίο επιτρέπει στους χρήστες που έχουν σημαντική διαφορά κέρδους καναλιού να μοιράζονται τον ίδιο πόρο, ενώ δεν υπάρχει αυστηρός περιορισμός χρηστών.



Σχήμα 2.45. OMA VS NOMA

Βασικά πλεονεκτήματα NOMA :

1. Βελτιωμένη φασματική απόδοση:
 - a. Στην άνω ζεύξη, η NOMA πετυχαίνει το όριο της χωρητικότητας, ενώ η OMA όχι.
 - b. Στην κάτω ζεύξη, η NOMA είναι βέλτιστη, ενώ η OMA είναι μη βέλτιστη εάν ο δέκτης γνωρίζει μόνο το CSI.
2. Πόροι-Χρήστες:
 - a. Η μη ορθογωνική κατανομή των πόρων με την NOMA υποδηλώνει ότι ο αριθμός των χρηστών που μπορεί να υποστηρίξει δεν περιορίζεται στο πλήθος των διαθέσιμων πόρων αλλά από τον χρονοπρογραμματισμό. Πετυχαίνει μέσω της κοινόχρηστης πρόσβασης πολλαπλών χρηστών (Multi User Shared Access - MUSA) καλύτερη κατανομή πόρων και έτσι υποστηρίζει περισσότερους χρήστες σε σχέση με την OMA.
3. Latency :
 - a. Σε αντίθεση με την OMA, σε μερικά σενάρια για την άνω ζεύξη της NOMA δεν απαιτείται δυναμικός χρονοπρογραμματισμός, καθιστώντας την έτσι, πιο ελεύθερη άνω μετάδοση η οποία μπορεί να μειώσει την καθυστέρηση της μετάδοσης και της σηματοδοσίας (signaling).

➤ Μοντέλο βαθιάς μάθησης για MIMO-NOMA

Σε αυτήν την ενότητα, θεωρούμε έναν νέο ανιχνευτή που υιοθετεί ένα DNN σε ένα σύστημα MIMO-NOMA. Σε αντίθεση με το κλασικό μπλοκ SIC, το οποίο χωρίζει τη διαδικασία ανίχνευσης σε ξεχωριστά μπλοκ, συμπεριλαμβανομένου της εκτίμησης καναλιών, την ανίχνευση MMSE, την αποδιαμόρφωση, και την αποκωδικοποίηση καναλιού, η μέθοδος βαθιάς μάθησης μπορεί να εκτελέσει όλες αυτές τις διαδικασίες ως μία μόνο διαδικασία. Οι βέλτιστες παράμετροι μπορούν να αποκτηθούν με συνεχή επανάληψη για τον προσδιορισμό των κανόνων που σχετίζονται με την έξοδο.

Θεωρούμε ένα σύστημα MIMO με M κεραίες εκπομπής και N κεραίες λήψης, ενώ υποθέτουμε K ο αριθμός των UE. Το σήμα μετάδοσης MIMO-NOMA μπορεί να εκφραστεί με την μορφή πίνακα ως εξής: [168]

$$S = (S_1, S_2, \dots, S_M),$$

S_m ($m \in [1, M]$) να είναι η m -οστή κεραία μετάδοσης και μπορεί να εκφραστεί ως εξής:

$$S_m = \sum_{k=1}^K \sqrt{P_k} S_m^k, \quad (p \rightarrow \text{power}, s \rightarrow \text{signal})$$

Για μια διαμόρφωση M-PSK, $S_m^k \in X_i$ είναι σήμα μετάδοσης από το k -οστό UE της m -οστής κεραίας και X_i είναι το σύνολο των M μεταδιδόμενων σημάτων:

$$X_i = A e^{j(\omega t + \phi_i)}, \quad i = (1, 2, \dots, M)$$

Το MIMO-NOMA χρησιμοποιεί την ισχύς σήματος για να βελτιώσει την χωρητικότητα του καναλιού, έτσι ο πίνακας καναλιού είναι ένας τανυστής [121][124] τρίτης τάξης:

$H \in \mathbb{R}^{N \times M \times K}$. h_{nm}^k , ($k \in [1, K]$, $m \in [1, M]$, και $n \in [1, N]$) είναι το κέρδος του καναλιού του k -οστού UE από την m -οστή κεραία εκπομπής στην n -οστή κεραία λήψης. Με αναδιαμόρφωση τανυστών προκύπτει ο πίνακας [157]

$$H_{(3)} = \begin{bmatrix} h_{11}^1 & \dots & h_{1N}^1 & \dots & h_{11}^K & \dots & h_{1N}^K \\ h_{21}^1 & \dots & h_{2N}^1 & \dots & h_{21}^K & \dots & h_{2N}^K \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{M1}^1 & \dots & h_{MN}^1 & \dots & h_{M1}^K & \dots & h_{MN}^K \end{bmatrix}$$

ενώ το λαμβανόμενο σήμα είναι: $Y_{(3)} = SH_{(3)} + N_{(3)}$

Το κανάλι και η λήψη σήματος του k -οστού UE επισημαίνονται με H^k και Y^k ως εξής:

$$H^k = \begin{bmatrix} h_{11}^k & h_{12}^k & \dots & h_{1N}^k \\ h_{21}^k & h_{22}^k & \dots & h_{2N}^k \\ \vdots & \vdots & \ddots & \vdots \\ h_{M1}^k & h_{M2}^k & \dots & h_{MN}^k \end{bmatrix},$$

$$Y^k = (y_1^k \quad y_2^k \quad \dots \quad y_M^k).$$

Η ανίχνευση SIC χρειάζεται μια συνεχή διαδικασία αποκωδικοποίησης, ανακατασκευής και ακύρωσης σήματος. Υποθέτουμε ότι η ισχύς που κατανέμεται στα UE μειώνεται σταδιακά ($P_1 > P_2 \dots > P_k$), l_q^k είναι η εκτίμηση εξόδου για το σήμα του q-οστού UE στο k-οστό UE δέκτη ενώ η κατάσταση του καναλιού είναι γνωστή. Για το πρώτο UE_1 η διαδικασία ανίχνευσης MMSE μπορεί να εκφραστεί ως :

$$l_1^1 = \arg \min_i \|Y^1 \hat{H}^1 - \sqrt{P_1} X_i\|,$$

$$\hat{H}^1 = (H^1)^H \left(H^1 (H^1)^H + \sigma_1^2 I \right)^{-1}$$

Για το δεύτερο UE_2 , οι πληροφορίες του UE_1 πρέπει πρώτα να εξαχθούν και η διαδικασία υπολογισμού είναι παρόμοια, ενώ στην συνέχεια το ανακατασκευασμένο σήμα αφαιρείται από το ληφθέν σήμα :

$$\hat{Y}^1 = Y^2,$$

$$\hat{Y}^2 = \hat{Y}^1 - \sqrt{P_1} S_{l_1^1} H^2.$$

ενώ το αποκωδικοποιημένο σήμα του

$$l_2^2 = \arg \min_i \|\hat{Y}^2 \hat{H}^2 - \sqrt{P_2} X_i\|.$$

Γενικεύοντας, για το k-οστό UE η ανίχνευση μπορεί να γίνει ως :

$$\hat{Y}^K = \hat{Y}^{K-1} - \sqrt{P_{K-1}} S_{l_{K-1}^K} H^K,$$

$$l_K^K = \arg \min_i \left\| \left(\hat{Y}^K \right) \hat{H}^K - \sqrt{P_2} X_i \right\|.$$

Από τις προηγούμενες εξισώσεις, το τελικό αποτέλεσμα ταξινόμησης μπορεί να εκφραστεί στη μορφή:

$$f_K \cdots f_2 (Y_2 f_1 (Y_1 A + b_1) + b_2) \cdots + b_K$$

όπου Y_k είναι σταθερός πίνακας, b_k είναι ένα σταθερό διάνυσμα ενώ η f_i αντιπροσωπεύει κάποια μορφή μη γραμμικής συνάρτησης.

Ο ανιχνευτής DNN είναι ικανός να αντικαταστήσει την παραδοσιακή μέθοδο SIC σε ένα

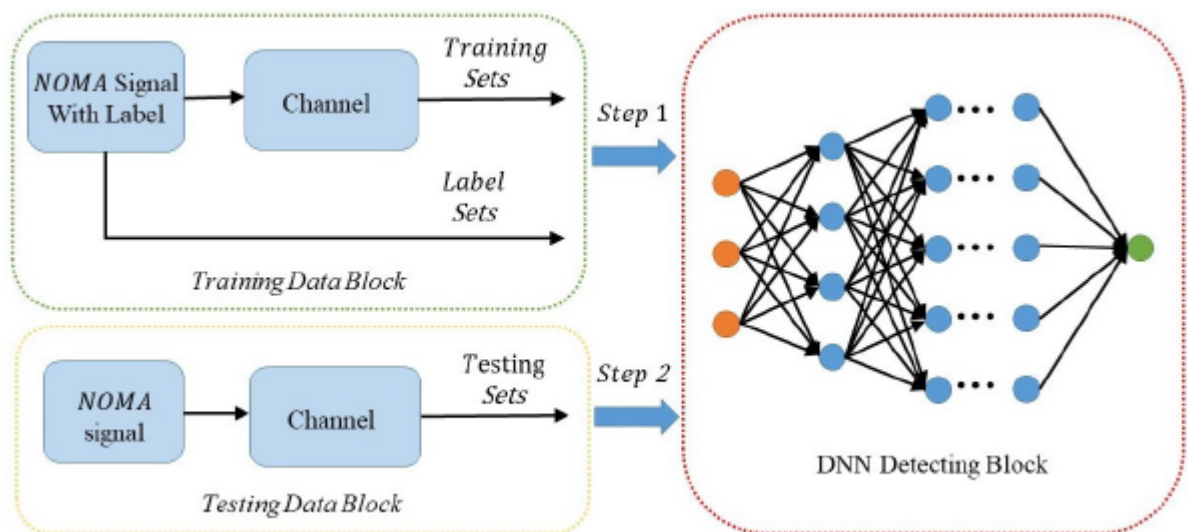
σύστημα MIMO-NOMA και είναι ακόμη πιο ισχυρός λόγω της ικανότητάς του να βρει τη βέλτιστη λύση.

➤ Σύστημα MIMO-NOMA με deep learning (DL)

Σε αυτήν την υποενότητα, παρουσιάζεται ένας νέος ανιχνευτής DL για ανίχνευση σήματος MIMO-NOMA, ο οποίος χωρίς επιπλέον επεξεργασία σήματος, δέχεται απευθείας το σήμα από τις κεραίες λήψης, δημιουργώντας έτσι ένα πιο εύκολο δέκτη από τον δέκτη SIC. Το καινούργιο αυτό σχήμα αποτελείται συνολικά από τρία μπλοκ:

1. Μπλοκ εκπαίδευσης
2. Μπλοκ δοκιμών
3. Μπλοκ DNN ανίχνευσης

Η απεικόνιση του μοντέλου φαίνεται στο επόμενο σχήμα.



Σχήμα 2.46. Δομή συστήματος MIMO-NOMA DL[168]

Το τμήμα εκπαίδευσης είναι υπεύθυνο για την παραγωγή του σήματος MIMO-NOMA και την παροχή των ετικετών στο DNN. Σε αυτό το μπλοκ, για να αποκτήσουμε το σήμα MIMO-NOMA για κεραίες N_t , θα πρέπει να παράγουμε δύο εκπαιδευτικές ακολουθίες UE1 και UE2 για κάθε κεραία. Στη συνέχεια διαμορφώνονται με κωδικοποίηση υπέρθεσης (superposition coding) με διαφορετικούς συντελεστές ισχύος κατανομής, ενώ το σήμα που έχει περάσει και αλλοιωθεί (θόρυβος) από το κανάλι λαμβάνεται στο δέκτη. (Αυτές οι ακολουθίες είναι γνωστές από τον δέκτη ως ετικέτες, και είναι παρόμοια με την πιλοτική ακολουθία.)

Το μπλοκ δοκιμών χρησιμοποιείται για την προσομοίωση μετάδοσης MIMO-NOMA σε πραγματικό χρόνο. Σε αυτό το μπλοκ, παράγεται πρώτα το σήμα MIMO-NOMA, ενώ δεν απαιτούνται ετικέτες. Το SNR στο μπλοκ εκπαίδευσης δημιουργείται τυχαία με το time slot να κυμαίνεται στο εύρος ενδιαφέροντος, ενώ το SNR στο μπλοκ δοκιμών είναι σταθερό έτσι ώστε η απόδοση σφάλματος του DNN να μπορεί να αξιολογηθεί σε ορισμένες συνθήκες SNR. Το μπλοκ DNN είναι το κύριο μπλοκ ανίχνευσης για την αποκωδικοποίηση του λαμβανόμενου σήματος.

Τα χαρακτηριστικά του καναλιού και ο αλγόριθμος αποκωδικοποίησης MIMO-NOMA μπορούν να μελετηθούν βελτιστοποιώντας τις παραμέτρους του νευρωνικού δικτύου. Τα δύο πρώτα μπλοκ παρέχουν ετικέτες και σήματα από το κανάλι και το τελευταίο μπλοκ ανακτά τα αρχικά δεδομένα, η διαδικασία ανίχνευσης μπορεί να χωριστεί σε δύο στάδια:

1. στάδιο εκπαίδευσης

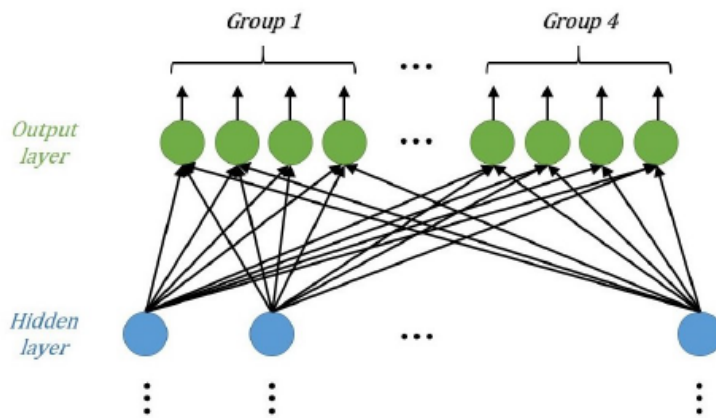
Στο στάδιο εκπαίδευσης, το (offline) block εκπαίδευσης είναι ενεργό. Η είσοδος του συστήματος εκπαίδευσης DNN περιλαμβάνει το ληφθέν MIMO-NOMA σήμα ως το επίπεδο εισόδου του συστήματος DNN, και οι ετικέτες ως δεδομένα (εκπαίδευσης με επίβλεψη) για να βοηθήσουν το DNN στην βελτιστοποίηση των παραμέτρων.

2. στάδιο δοκιμών

Το στάδιο δοκιμής ενεργοποιείται μετά την εκπαίδευση του DNN. Σε αυτό το στάδιο το online μπλοκ έχει ενεργοποιηθεί (το offline μπλοκ απενεργοποιείται) και η απόδοση του συστήματος αξιολογείται..

Το μοντέλο DNN που παρουσιάζεται για την ανίχνευση MIMO-NOMA περιλαμβάνει επτά επίπεδα: ένα στρώμα εισόδου, ένα στρώμα εξόδου και πέντε κρυφά επίπεδα. Το επίπεδο εισόδου και το κρυφό στρώμα είναι πλήρως συνδεδεμένα, ενώ τα επίπεδα εξόδου χωρίζονται σε ομάδες για την αποκωδικοποίηση των σημάτων πολλαπλών κεραιών.

Στο επίπεδο εισόδου λαμβάνεται το σήμα το οποίο στην συνέχεια χωρίζεται σε πραγματικό και φανταστικό μέρος ώστε να επεξεργαστεί. Τα κρυφά στρώματα αποτελούνται από πέντε πλήρως συνδεδεμένα επίπεδα. Λόγω των πολλών επιπέδων δεν μπορούν να χρησιμοποιηθούν ως συναρτήσεις ενεργοποίησης η υπερβολική εφαιπτομένη ή η σιγμοειδής λόγω του προβλήματος της εξαφάνισης της κλίσης (vanishing gradient problem)(η κλίση εξαφανίζεται λόγω των επιπέδων περιοχών του σιγμοειδούς) για αυτό γίνεται χρήση της συνάρτησης ράμπας-ReLu (Rectified Linear Unit)[1] ως μια πιο αποτελεσματική συνάρτηση ενεργοποίησης του νευρώνα.[159] Το στρώμα εξόδου χρησιμοποιείται για την αναφορά των τελικών αποτελεσμάτων ανίχνευσης. Το επίπεδο εξόδου DNN είναι συνήθως πλήρως συνδεδεμένο και έχει κωδικοποίηση one-hot με τη συνάρτηση softmax. Ωστόσο, στην ανίχνευση σήματος MIMO-NOMA, τα σήματα από πολλές κεραιές πρέπει να αποκωδικοποιηθούν σε ένα slot. Έτσι, το επίπεδο εξόδου σχεδιάζεται για να σχηματίζει ομάδες. Ο αριθμός των ομάδων είναι ίσος με τον αριθμό των κεραιών εκπομπής N_t και ο αριθμός των νευρώνων σε κάθε ομάδα είναι ίσος με τον αριθμό των one-hot κωδικοποιήσεων.



Σχήμα 2.47. παράδειγμα για MIMO-NOMA 4X4[168]

Η επιλογή της συνάρτησης κόστους και του αλγόριθμου βελτιστοποίησης είναι ένα βασικό κομμάτι για το σύστημα MIMO-NOMA-DL. Η συνάρτηση κόστους μετρά την απόσταση μεταξύ των προβλέψεων και των ετικετών. Η κλασική συνάρτηση απώλειας είναι η συνάρτηση μέσου τετραγώνου σφάλματος (MSE). Στη λογιστική παλινδρόμηση, το MSE αποδίδει από την άποψη της ακρίβειας. Ωστόσο, σε προβλήματα ταξινόμησης πολλαπλών κλάσεων, το MSE έχει αργή ταχύτητα σύγκλισης. Σε αυτό το σύστημα εξετάζεται η συνάρτηση εγκάρσιας εντροπίας (cross-entropy)[170]. Η απόκλιση Kullback – Leibler (KLD) [169] μπορεί να χρησιμοποιηθεί για την αναπαράσταση της διαφοράς μεταξύ δύο κατανομών πιθανότητας και η έκφραση γράφεται ως:

$$D_{KL} (P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}(x).$$

Η διαδικασία ελαχιστοποίησης του KLD ισοδυναμεί με την ελαχιστοποίηση της εγκάρσιας εντροπίας $H(P, Q)$, η οποία ορίζεται ως:

$$H(P, Q) = D_{KL} (P||Q) + H(P)$$

Η συνάρτηση εγκάρσιας εντροπίας έχει γρήγορη σύγκλιση και χαμηλή πολυπλοκότητα στη διαδικασία επαναληπτικής βελτιστοποίησης, ενώ η μέθοδος Adam[1] χρησιμοποιείται ως αλγόριθμος βελτιστοποίησης.

➤ Προσομοίωση και ανάλυση

Σε αυτήν την ενότητα, έχουμε το σύστημα ανίχνευσης σήματος NOMA με βάση τη μέθοδο βαθιάς μάθησης (με επίβλεψη) και παρουσιάζουμε τα αριθμητικά αποτελέσματα για διαφορετικές παραμέτρους. Οι παράμετροι της προσομοίωσης και ο αλγόριθμος παρουσιάζονται στην συνέχεια: [168]

Παράμετροι συστήματος:

Channel Fading	MIMO channel/AWGN Rayleigh distribution
Number of UEs per cluster	2
Number of transmit antennas	4
Number of receive antennas	4
Modulation	PSK
Number of training samples	409,600
Total transmitted power per antenna	1 W
Power allocation factor	0.8
Hidden layer	ReLU
Output later	Sigmoid

Αλγόριθμος :

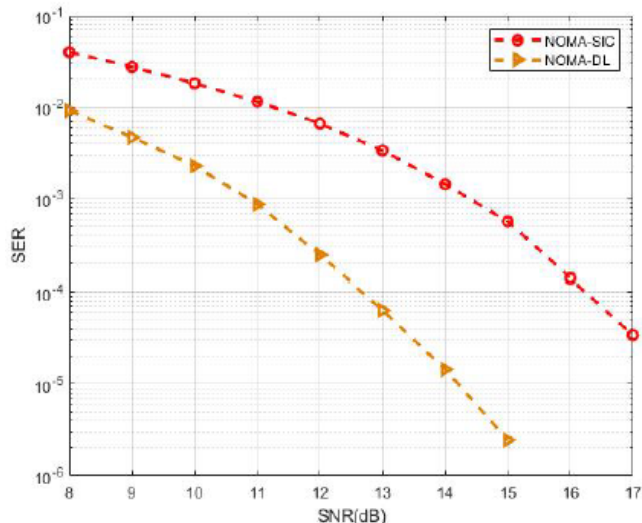
1. Αρχικοποίηση μοντέλου DNN
2. Δημιουργία και προσαρμογή των δεδομένων εκπαίδευσης. Υποθέτουμε ότι ο αριθμὸν των slot είναι N , τα δεδομένα εισόδου δηλώνονται ως εξής : $x = x^{[1]}, x^{[2]}, \dots, x^{[N]}$. Κάθε $x^{[i]}$ είναι ένα MIMO-NOMA δiάνυσμα στήλης στο slot i .
3. Ορισμός βασικών παραμέτρων , συμπεριλαμβανομένων των υποομάδων (mini-batch) , του ρυθμού μάθησης , των συναρτήσεων εξόδου του κρυφού επιπέδου και του επιπέδου εξόδου, την αρχικοποίηση των βαρών και του όρου bias για κάθε επίπεδο DNN.
4. Εφαρμογή της διαδικασίας προώθησης του DNN (forward DNN process) και λήψη των αποτελεσμάτων από το επίπεδο εξόδου που δηλώνονται ως εξής : $\hat{y} = y_i^{[1]}, y_i^{[2]}, \dots, y_i^{[M]}$
5. Υπολογισμός της συνάρτησης κόστους (εγκάρσιας εντροπίας) , (y, \hat{y}) ως εξής :

$$Loss(y, \hat{y}) = \sum_i y_i \log \frac{y_i}{\hat{y}_i} + H(y) + \lambda \cdot \sum_i \|\omega_i\|_2.$$

6. Υπολογίστε τη διορθωτική παράμετρο με τον αλγόριθμο βελτιστοποίησης Adam. Ενημερώστε τις παραμέτρους με τον αλγόριθμο για να αναζητήσετε τη βέλτιστη λύση.
7. Επιστροφή στο βήμα 4 εάν η συνάρτηση κόστους δεν είναι αρκετά μικρή, διαφορετικά προχωρήστε στο επόμενο βήμα. Εάν η συνάρτηση κόστους δεν πληροί τις απαιτήσεις, το DNN με τις ενημερωμένες παραμέτρους θα πρέπει να επανεκπαιδευτεί.
8. Δοκιμή στο εκπαιδευμένο DNN με τα δεδομένα δοκιμής και σχεδιασμός καμπύλης SER-SNR.

Η μέθοδος ανίχνευση σήματος MIMO-NOMA-DL συγκρίνεται με το παραδοσιακό σχήμα MIMO-NOMA-SIC. Υποθέτουμε ότι το SIC γνώριζε τέλεια τις παραμέτρους καναλιού και ότι ο τύπος διαμόρφωσης και των δυο UE στον πομπό είναι BPSK. Στο κλασικό σχήμα ανίχνευσης MIMO-NOMA-SIC, το σήμα UE1 , το οποίο αντιμετωπίζει το σήμα UE2 ως παρεμβολή, πρέπει πρώτα να αποδιαμορφωθεί. Στη συνέχεια, μπορούμε να αποδιαμορφώσουμε το σήμα UE2 μετά την αφαίρεση του διαμορφωμένου σήματος UE1 από το λαμβανόμενο NOMA-MIMO σήμα. Ωστόσο, στο σχήμα MIMO-NOMA-DL, το ληφθέν σήμα αποστέλλεται στο DNN και οι επικείμενες επιλέγονται μόνο για την ακολουθία UE2 κατά τη διάρκεια της εκπαίδευσης.

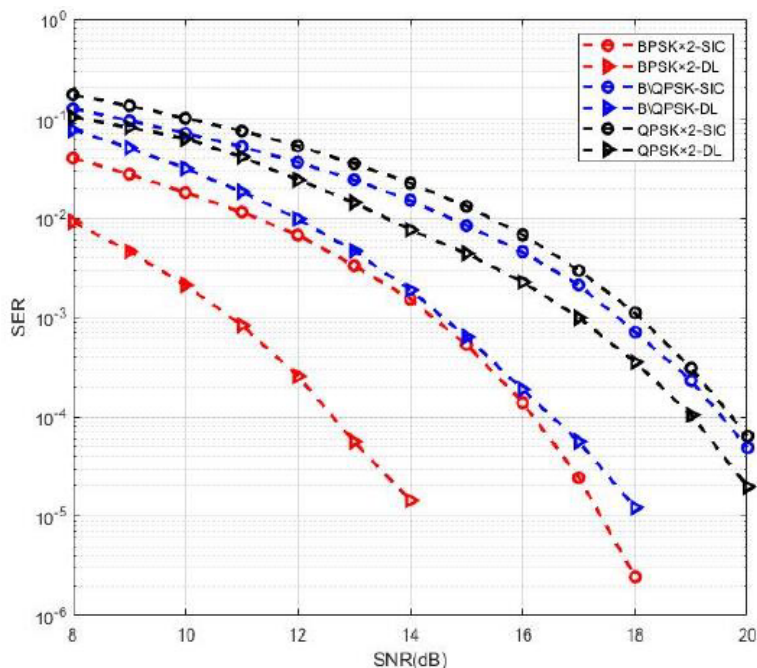
Το επόμενο σχήμα δείχνει την καμπύλη SER-SNR.



Σχήμα 2.48. SER VS SNR για NOMA-DL και NOMA-SIC[168]

Αν πάρουμε ως σημείο αναφοράς το $SER = 10^{-4}$ βλέπουμε μια διαφορά 3.6 dB μεταξύ του NOMA-DL (12.6 dB) και του NOMA-SIC (16.2 dB), ενώ δεν πραγματοποιήθηκε προεπεξεργασία ούτε μετα-επεξεργασία, αντί της παραδοσιακής σύνθετης επεξεργασίας σήματος για εκτίμηση καναλιών και αποδιαμόρφωση σήματος, έγινε χρήση ισχυρών εργαλείων βαθιάς μάθησης για να γίνει ακριβή ανίχνευση σήματος.

Το σύστημα MIMO-NOMA υιοθετεί κωδικοποίηση υπέρθεσης, στην οποία τα σήματα από διαφορετικά UE επικαλύπτονται με συγκεκριμένη αναλογία ισχύος. Μπορούν να χρησιμοποιηθούν διαφορετικοί τύποι διαμόρφωσης για διαφορετικά σήματα UE. Επειδή το NOMA είναι μια μη ορθογώνια τεχνολογία η παρεμβολή από άλλα UE είναι ένας σημαντικός παράγοντας για τον προσδιορισμό της απόδοσης αποδιαμόρφωσης.

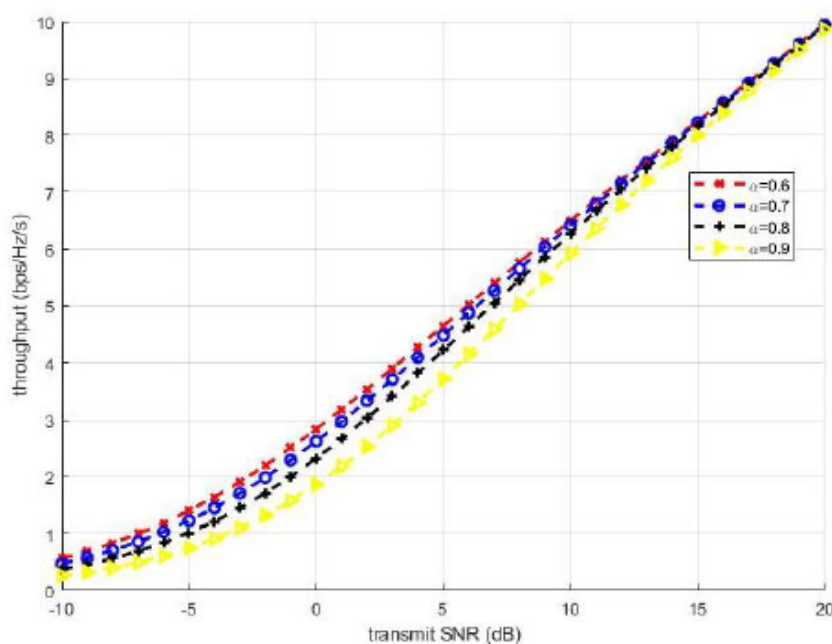


Σχήμα 2.49. SER VS SNR για NOMA-DL και NOMA-SIC για διαφορετικά σχήματα διαμόρφωσης[168]

Ένας άλλος σημαντικός παράγοντας για το σύστημα NOMA είναι η κατανομή ισχύος μετάδοσης των EUs στο BS καθώς επηρεάζει το throughput. Το συνολικό throughput για δυο UEs αναφέρεται ως :

$$R_{sum}(\alpha) = \log_2 \frac{(1 + (1 - \alpha) \rho |h_2|^2) (1 + \rho |h_1|^2)}{1 + (1 - \alpha) \rho |h_1|^2}$$

όπου ρ είναι το εκπεμπόμενο SNR, $h_i (i = 1, 2)$ είναι το κέρδος καναλιού, $h_1 < h_2$. Η απόδοση μπορεί να αποδειχθεί ότι είναι μια μονότονα μειούμενη συνάρτηση του α για $\frac{dR_{sum}}{d\alpha} < 0$, $\alpha \in [0, 1]$. Το επόμενο σχήμα δείχνει το throughput για ένα σύστημα NOMA με σταθερό συντελεστή κατανομής ισχύος στο 0.6, 0.7, 0.8. Το throughput αυξήθηκε καθώς το SNR μετάδοσης αυξήθηκε. Επιπλέον, όσο μεγαλύτερος ήταν ο συντελεστής κατανομής ισχύος, τόσο μικρότερο ήταν το throughput.



Σχήμα 2.50. NOMA throughput για διαφορετικές τιμές του α . [168]

➤ Συμπεράσματα :

Η εφαρμογή της βαθιάς μάθησης στα συστήματα επικοινωνίας MIMO-NOMA είναι μια πολλά υποσχόμενη προσέγγιση για να αντιμετωπίσει τα μειονεκτήματα της μεθόδου SIC. Αντί για τον περίπλοκο σχεδιασμό αλγορίθμων και τις διαδικασίες ακύρωσης παρεμβολών, η προσέγγιση βαθιάς μάθησης μπορεί να αναζητήσει τη βέλτιστη λύση των παραμέτρων του πολυεπίπεδου νευρωνικού δικτύου με μηχανική εκμάθηση. Σε αυτή την ενότητα, είδαμε

ένα σύστημα ανίχνευσης σήματος MIMO-NOMA με βαθιά μάθηση για την εκτέλεση ανάκτησης σήματος, η τεχνική αυτή μπορεί ταυτόχρονα να ολοκληρώσει τις διαδικασίες εκτίμησης καναλιού και ανίχνευσης σήματος MIMO-NOMA. Απο τις συγκρίσεις των συστημάτων βλέπουμε ότι το DL σύστημα πετυχαίνει αρκετά καλά αποτελέσματα σε σχέση με το κλασικό SIC, σημαντικό ρόλο παίζει και ο τύπος διαμόρφωσης που θα χρησιμοποιηθεί, ενώ δοθηκαν τα βασικά βήματα του αλγορίθμου σχεδιασμού για ένα τέτοιο σύστημα.

Κεφάλαιο 3

Αποτελέσματα από προσομοιώσεις σε python

Σε αυτό το κεφάλαιο θα παρουσιαστούν αποτελέσματα από προσομοιώσεις σε python. Πιο συγκεκριμένα, θα αναφερθούμε στο πρόβλημα εύρεσης τοποθεσίας σε εσωτερικούς χώρους καθώς και την προσαρμοστική διαμόρφωση, σε όλα τα πειράματα έγινε χρήση της γλώσσα python με την βιβλιοθήκη scikit-learn.

3.1 Εύρεση τοποθεσία χρήστη σε εσωτερικούς χώρους

Η εύρεση θέσης ενός χρήστη έχει δείξει ιδιαίτερη άνθηση τα τελευταία χρόνια, και ενσωματώνεται πλέον σχεδόν σε κάθε φορητή συσκευή και όχι μόνο. Αρκετές τεχνικές έχουν υιοθετηθεί για την εύρεση της θέσης ενός χρήστη κυρίως, για εξωτερικούς χώρους. Το πλέον γνωστό σύστημα εντοπισμού θέσης είναι το GPS (Global Positioning System) [165], το οποίο αποτελείται από πολυάριθμους (24 - 32) δορυφόρους, καλύπτοντας σχεδόν κάθε σημείο της γης, εφοδιασμένος με ειδικές συσκευές εντοπισμού (πομποδέκτες GPS). Οι πομποδέκτες αυτοί παρέχουν ακριβείς πληροφορίες για τη θέση ενός σημείου, το υψόμετρό του, την ταχύτητα και την κατεύθυνση της κίνησης του ενώ μπορούν να χρησιμοποιηθούν με ειδικό λογισμικό χαρτογράφησης. Βασικό του μειονέκτημα αποτελεί η χρήση του σε εσωτερικούς χώρους λόγω του αδύναμου σήματος δορυφόρου που δεν είναι επαρκές ώστε να φτάσει μέσα σε κτίρια. Άλλες τεχνικές όπως Bluetooth και Wi-Fi θα μπορούσαν να χρησιμοποιηθούν για εσωτερικούς χώρους ενώ θα μπορούν να συμβάλουν και να συνδυαστούν με άλλες πολλά υποσχόμενες μελλοντικές εφαρμογές όπως τα έξυπνα σπίτι (smart home), εντοπισμός θέσης σε περιπτώσεις καταστροφών κτλ.

Για τον εντοπισμό της τοποθεσίας ενός χρήστη σε εσωτερικούς χώρους ένα μοντέλο πρέπει δημιουργηθεί και να εκπαιδευτεί λαμβάνοντας δεδομένα από κάποια συσκευή. Πιο συγκεκριμένα θα γίνει χρήση της ισχύος του σήματος Wi-Fi από διαφορετικά router ενός εσωτερικού χώρου ανάλογα την τοποθεσία του χρήστη (δωμάτιο). Μειονεκτήματα αυτής της τεχνικής είναι η απαραίτητη χαρτογράφηση κάθε εσωτερικού χώρου. Θεωρούμε την εγκατάσταση ενός γραφείου [166] το οποίο αποτελείται από τέσσερα δωμάτια καθώς και επτά routers. Οι μετρήσεις λαμβάνονται από συσκευή android κάθε 1s για διαφορετικά δωμάτια. Τα router αναπαρίσταται ως WS_i (Wi-Fi Station) όπου i αναπαριστά το i -οστό router ενώ τα δωμάτια με $room_j$ όπου j είναι το j -οστό δωμάτιο. Κάθε κλάση περιέχει 500 πρότυπα. Τα πρώτα δεδομένα φαίνονται στον επόμενο πίνακα:

WS1	WS2	WS3	WS4	WS5	WS6	WS7	class
-64	-56	-61	-66	-71	-82	-81	room1
-68	-57	-61	-65	-71	-85	-85	room1
-63	-60	-60	-67	-76	-85	-84	room1
-61	-60	-68	-62	-77	-90	-80	room1
-63	-65	-60	-63	-77	-81	-87	room1
-64	-55	-63	-66	-76	-88	-83	room1
-65	-61	-65	-67	-69	-87	-84	room1
-61	-63	-58	-66	-74	-87	-82	room1
-65	-60	-59	-63	-76	-86	-82	room1
-62	-60	-66	-68	-80	-86	-91	room1
-67	-61	-62	-67	-77	-83	-91	room1
-65	-59	-61	-67	-72	-86	-81	room1

Πίνακας 3.1 Δεδομένα μετρήσεων

Το πρόβλημα πλέον ανάγεται σε πρόβλημα ταξινόμησης (μάθηση με επίβλεψη) , δέχοντας ως είσοδο την ισχύ από τα Wi-Fi και έξοδο την κλάση (δωμάτιο).

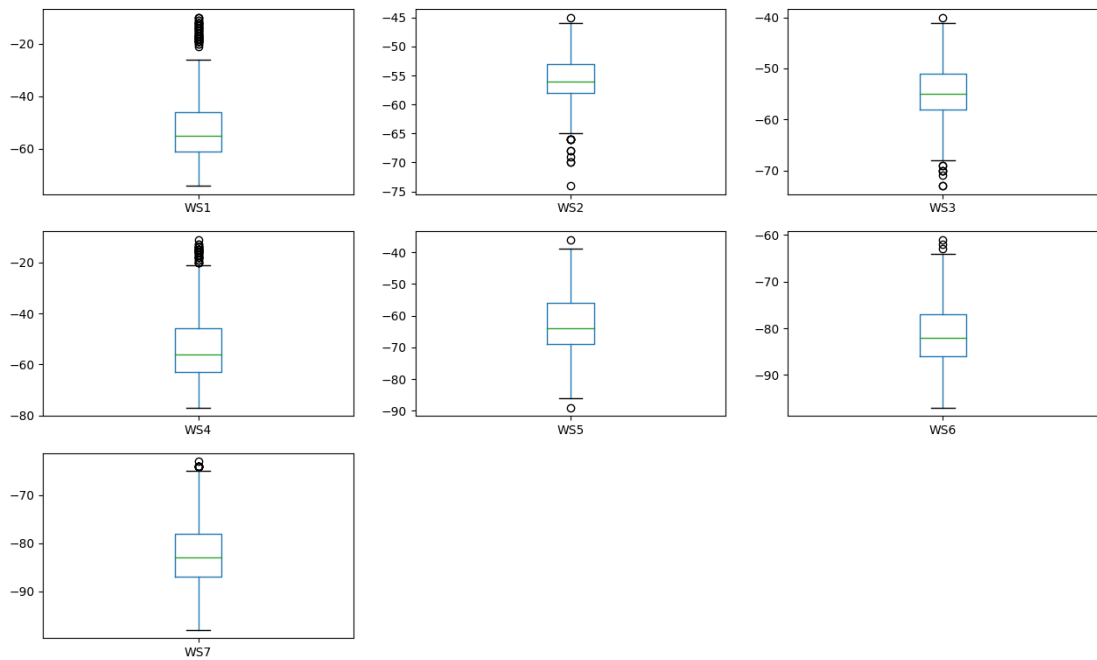
Το μοντέλο θα εκπαιδευτεί με έξι αλγόριθμους (οι οποίοι έχουν παρουσιαστεί στο κεφάλαιο 1) ώστε να δούμε ποιος έχει καλύτερη απόδοση , οι αλγόριθμοι είναι:

- Logistic Regression (one-vs-rest) [168]
- Linear Discriminant Analysis [169]
- K Neighbors Classifier ($k=5$) [170]
- Decision Tree Classifier [171]
- Gaussian NB [27]
- SVM (one-vs-one) (kernel->rbf)[172]

Το 20% των δεδομένων είναι για δοκιμή και το 80% για εκπαίδευσης, ενώ για την επίδοση του μοντέλου χρησιμοποιούμε τη μέθοδο διασταυρωμένης επικύρωσης (kfold , $k=10$) [173].

Σύμφωνα με αυτή τη μέθοδο , ελέγχουμε την επίδοση ενός μοντέλου ως προς την ικανότητα γενίκευσης του, εκπαιδεύοντας το μοντέλο σε ένα σύνολο δεδομένων (σύνολο εκπαίδευσης) και ελέγχοντας την επίδοσή του σε ένα διαφορετικό σύνολο δεδομένων (σύνολο ελέγχου). Τα δεδομένα του συνόλου ελέγχου δεν πρέπει να χρησιμοποιηθούν κατά την εκπαίδευση. [1] Η μέθοδος που χρησιμοποιείται είναι ο διαμοιρασμός των δεδομένων , χωρίζονται σε K ομάδες και εκτελούνται K πειράματα. Στο i -οστό πείραμα εκπαιδεύουμε τον αλγόριθμο χρησιμοποιώντας όλες τις ομάδες εκτός από την i -οστή που είναι για έλεγχο. [181]

Μια γραφική απεικόνιση των τιμών των δεδομένων μας φαίνεται στο επόμενο διάγραμμα (box and whisker plots - Θηκόγραμμα). Το διάγραμμα δίνει πληροφορίες σχετικά με τις μεγαλύτερες τις μικρότερες αλλά και τη μέση τιμή για κάθε κλάση. [176][177][178]



Σχήμα 3.1. box and whisker plots για κάθε χαρακτηριστικό

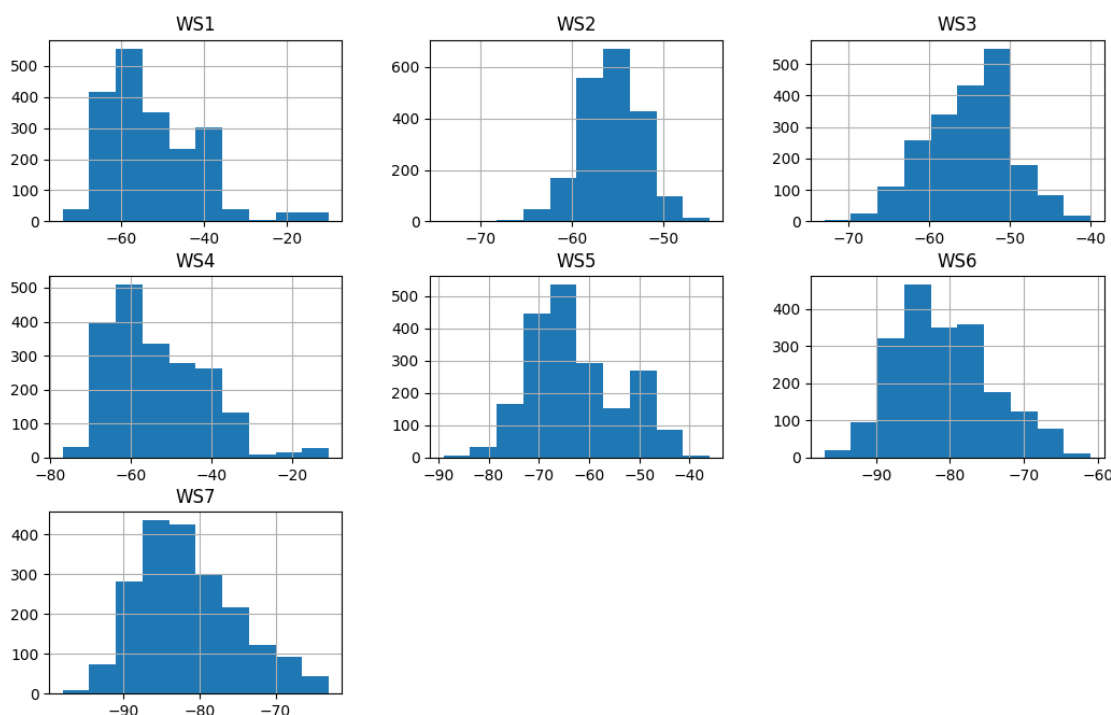
Μια αριθμητή αναπαράσταση του γραφήματος φαίνεται παρακάτω :

	WS1	WS2	...	WS6	WS7
count	2000.000000	2000.000000	...	2000.000000	2000.000000
mean	-52.330500	-55.623500	...	-80.985000	-81.726500
std	11.321677	3.417688	...	6.516672	6.519812
min	-74.000000	-74.000000	...	-97.000000	-98.000000
25%	-61.000000	-58.000000	...	-86.000000	-87.000000
50%	-55.000000	-56.000000	...	-82.000000	-83.000000
75%	-46.000000	-53.000000	...	-77.000000	-78.000000
max	-10.000000	-45.000000	...	-61.000000	-63.000000

Ο μέσος όρος για την κλάση 1 (WS1) είναι -52 , από το διάγραμμα είναι η μεσαία πράσινη γραμμή , ενώ άλλες επιπλέον πληροφορίες μπορούν να εξαχθούν. Οι τιμές 25% 50% κτλ. αποτελούν τα εκατοστημόρια δηλαδή είναι οι τιμές κάτω από τις οποίες βρίσκεται ένα ορισμένο ποσοστό δεδομένων.[191]

Το παραπάνω διάγραμμα δεν μας δίνουν πληροφορίες σχετικά με την συχνότητα και την κατανομή των δεδομένων μας , για αυτό μπορούμε να χρησιμοποιήσουμε ιστόγραμμα για την οπτικοποίηση της κατανομής. Το ιστόγραμμα είναι γραφική απεικόνιση στατιστικών συχνοτήτων περιοχών τιμών ενός μεγέθους, δείχνει δηλαδή τον αριθμό εμφανίσεων διαφορετικών τιμών σε ένα σύνολο δεδομένων. Η επιφάνεια κάθε ορθογωνίου είναι μέτρο της συχνότητας εμφάνισης της συγκεκριμένης περιοχής τιμών ενώ το ύψος του ισούται με το λόγο

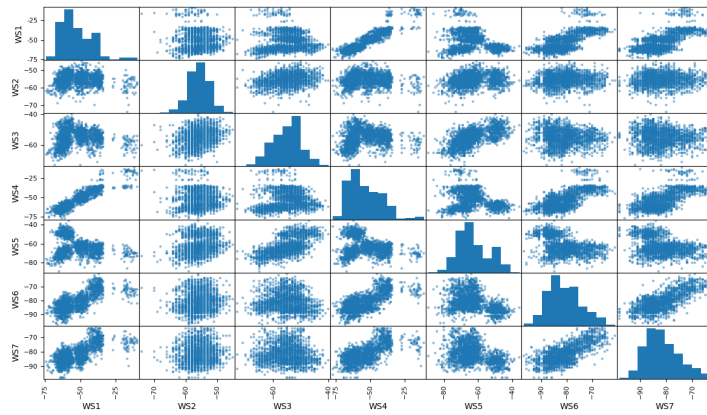
της συχνότητας προς το εύρος των τιμών που αντιπροσωπεύει το ορθογώνιο. [179]. Τα ιστογράμματα είναι χρήσιμα καθώς μπορούν να μας δώσουν πληροφορίες σχετικά με την κατανομή που πιθανόν να ακολουθούν τα χαρακτηριστικά ώστε να χρησιμοποιήσουμε αλγορίθμους που βασίζονται σε κατανομές.



Σχήμα 3.2. Ιστογράμματα για κάθε χαρακτηριστικό

Μπορεί να παρατηρηθεί ότι κάποιο χαρακτηριστικό (WS7) μοιάζει να ακολουθεί κατανομή Gaussian.

Τώρα μπορούμε να δούμε τις αλληλεπιδράσεις μεταξύ των μεταβλητών. Κατ' αρχάς, ας δούμε τα διασκορπισμένα σημεία όλων των ζευγών χαρακτηριστικών. Αυτό μπορεί να σας βοηθήσει να εντοπίσετε δομημένες σχέσεις μεταξύ των μεταβλητών εισαγωγής. Όσο πιο πυκνά είναι τόσο μεγαλύτερη συσχέτιση υπάρχει. Τα ιστογράμματα παρουσιάζουν κάθε χαρακτηριστικό ξεχωριστά και δεν μας δίνουν πληροφορίες για την συσχέτιση των χαρακτηριστικών μεταξύ τους. Μια μέθοδος οπτικοποίησης των δεδομένων αυτών είναι μέσω του πίνακα διασποράς scatter plots matrix. [180]



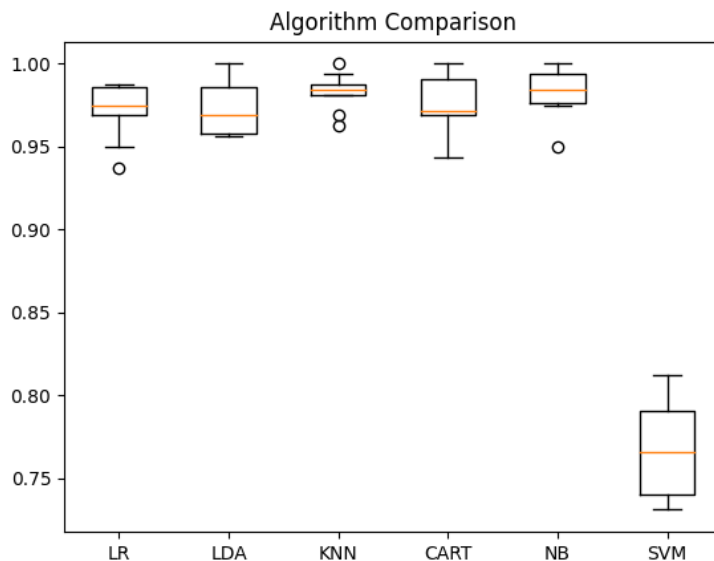
Σχήμα 3.3. scatterplots

Η διαγώνια δείχνει την κατανομή των δεδομένων μας σε ιστόγραμμα , ενώ στα άλλα κελιά του , έχουμε την γραφική παράσταση συσχέτισης κάθε μεταβλητού συνδυασμού του πλαισίου δεδομένων μας.

Στην συνέχεια μπορούμε να αφήσουμε το μοντέλο μας να εκπαιδευτεί με του 6 αλγορίθμους και να επιλέξουμε τον αλγόριθμο μας την καλύτερη ακρίβεια (προφανώς μετά το cross validation , $k=10$) , ενώ μπορούμε να τα οπτικοποιήσουμε με (box and whisker plot) .

Αριθμητική ακρίβεια αλγορίθμους, μέση ακρίβεια (ακρίβεια τυπικής απόκλισης) :

```
LR: 0.971875 (0.015873)
LDA: 0.972500 (0.015104)
KNN: 0.983125 (0.010477)
CART: 0.975000 (0.016771)
NB: 0.983125 (0.013707)
SVM: 0.766250 (0.027272)
```



Σχήμα 3.4. Box and whisker plot για κάθε αλγόριθμο

Από τα παραπάνω είναι εμφανές ότι ο καλύτερος αλγόριθμος είναι ο K-πλησιέστερος γείτονας (KNN) με ακρίβεια (μέση) 98%.

Μετά την εκπαίδευση του αλγορίθμου μας μπορούμε να κάνουμε πρόβλεψη (στα ήδη υπάρχοντα δεδομένα) ενώ τιμές για την ακρίβεια του μοντέλου (πίνακας σύγχυσης, ευστοχία, ανάκληση f1-measure-όπως περιγράφηκαν στην εισαγωγή) δίνονται και βρίσκονται κοντά στο 1. Δηλαδή έχουμε ένα πολύ καλό μοντέλο πρόβλεψης.

```

0.985
[[ 95  0  0  0]
 [  0 97  3  0]
 [  1  0 99  2]
 [  0  0  0 103]]
      precision    recall  f1-score   support

 room1      0.99      1.00      0.99         95
 room2      1.00      0.97      0.98        100
 room3      0.97      0.97      0.97        102
 room4      0.98      1.00      0.99        103

 accuracy          0.98         400
 macro avg      0.99      0.99      0.99         400
 weighted avg   0.99      0.98      0.98         400

```

Για το room1 παρατηρούμε ότι το precision είναι 0.99, δηλαδή 99 στα 100 δεδομένα μας που

ανιχνεύτηκαν να ανήκουν στο room1 όντως ανήκουν (το μέτρο είναι κατάλληλο για την ελαχιστοποίηση των ψευδών θετικών). Το recall (sensitivity) ισούται με 1 δηλαδή κατάφερε να βρει όλα τα δεδομένα που ανήκουν στο room1 (Κατάλληλο για την ελαχιστοποίηση των ψευδών αρνητικών είναι το επίκεντρο.) Ας υπολογίσουμε το f1-score $\frac{2(0.99*1)}{0.99+1} = \frac{1.98}{1.99} = 0.99$. Η τελευταία στήλη (support) αναφέρεται στον αριθμό των δεδομένων προς επαλήθευση. Όπως είπαμε μόλις 20% από το σύνολο είναι για επαλήθευση δηλαδή 400 , ενώ μπορούμε να δούμε πόσα επιλέχθηκαν από κάθε κλάση. Για τον πίνακα σύγχυσης (C) κάθε στοιχείο C_{ij} είναι ίσο με τον αριθμό των παρατηρήσεων που ταξινομήθηκαν στην i κλάση και προβλέφθηκαν να είναι στην j κλάση. Κάθε γραμμή του πίνακα αντιστοιχεί στο άθροισμα του συνόλου των παρατηρήσεων.

3.2 Προσαρμοστική διαμόρφωση

Η επιλογή της διαμόρφωσης για ένα σύστημα αποτελεί κρίσιμος παράγοντας για το ρυθμό μετάδοσης του τελικού χρήστη. Μια ασύρματη ζεύξη επηρεάζεται από διάφορους παράγοντες με του κυριότερους να αποτελούν την πολυδιαδρομική διάδοση τον θόρυβο, και την εξασθένηση. Χρήστες που έχουν καλύτερο κανάλι και συνήθως βρίσκονται κοντά στο σταθμό βάσης θα πάρουν υψηλότερη διαμόρφωση και κατά συνέπεια υψηλότερο ρυθμό μετάδοσης , ενώ σε χρήστες με χαμηλά ποιοτικό κανάλι δεν έχει νόημα να δοθεί υψηλή διαμόρφωση. Τα προσαρμοστικά αυτά συστήματα φέρουν καλή εμπειρία χρήστη (QOS) αλλά έχουν κόστος στον υπολογισμό του καναλιού , δημιουργώντας overhead υπό τον όρο ότι γίνεται ακριβής μοντελοποίηση καναλιών και πραγματοποιείται συνεχής εκτίμηση κατάστασης καναλιού. Σε αυτό το σημείο αποτελέσματα από μηχανικής μάθησης με επίβλεψη θα παρουσιαστούν για ένα τέτοιο σενάριο. Τα δεδομένα δημιουργήθηκαν από προσομοίωση σε matlab. Η επιλογή διαμόρφωσης βασίζεται στην τιμή του SNR. Πιο συγκεκριμένα έχουμε υπάρχοντες τιμές SNR μεταξύ -20 ως και 45 db. Η διαμόρφωση επιλέγεται ως εξής :

-20<SNR<-10	BPSK
-11<SNR<8	QPSK
7<SNR<14	8QAM
13<SNR<21	16QAM
20<SNR<34	32QAM
SNR>33	64QAM

Πίνακας : 3.2

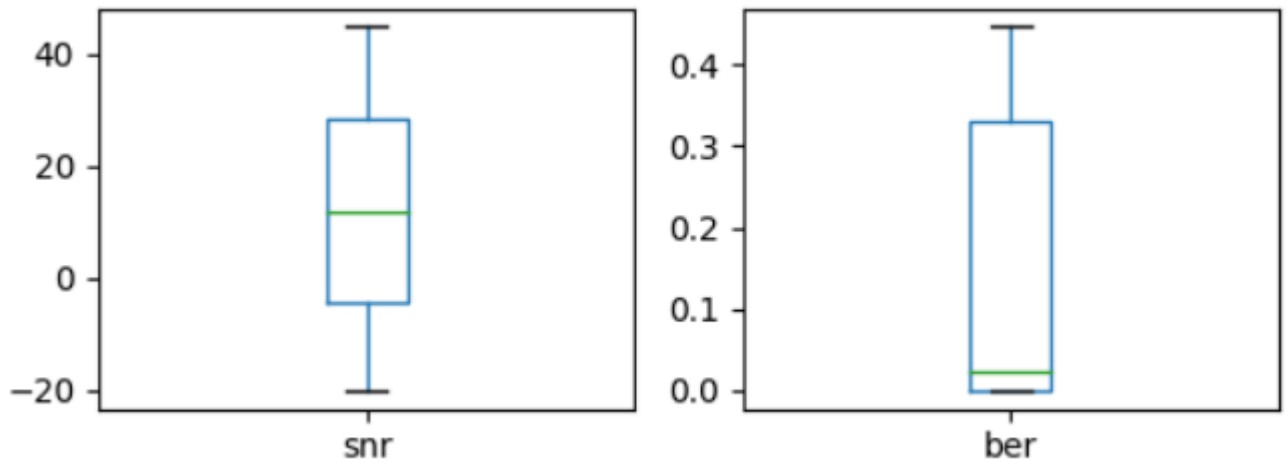
Η εκπαίδευση του αλγορίθμου θα γίνει με παρόμοιο τρόπο όπως στο προηγούμενο σενάριο , χρησιμοποιούνται δηλαδή έξι αλγόριθμοι , αξιολογείται η απόδοσή τους και επιλέγεται ο καλύτερος. Να σημειωθεί ότι κάθε τύπος διαμόρφωσης αντιπροσωπεύει μια κλάση , ενώ τα δεδομένα εκπαίδευσης είναι 4096. Ας δούμε μερικά από τα δεδομένα μας :

snr	ber	class
14.089368	0.023600	16QAM
-19.199540	0.439529	BPSK
31.088104	0.000000	32QAM
-2.287161	0.281845	QPSK
-5.705465	0.360207	QPSK
40.451162	0.000000	64QAM
-7.623942	0.392159	QPSK
-14.120377	0.390015	BPSK
-2.922519	0.299065	QPSK
35.361214	0.000000	64QAM
33.949193	0.000000	32QAM
-13.962365	0.390213	BPSK
4.186631	0.075333	QPSK
24.786281	0.000000	32QAM
12.194418	0.033989	8QAM
3.341585	0.102753	QPSK
35.409720	0.000000	64QAM
-3.681282	0.317490	QPSK
11.206430	0.053368	8QAM
18.837617	0.000443	16QAM

Πρότυπα κάθε κλάσης :

class	count
16QAM	433
32QAM	816
64QAM	674
8QAM	386
BPSK	657
QPSK	1130

Στο επόμενο διάγραμμα μπορούμε να δούμε μια αναπαράσταση των χαρακτηριστικών μας (SNR ,BER)



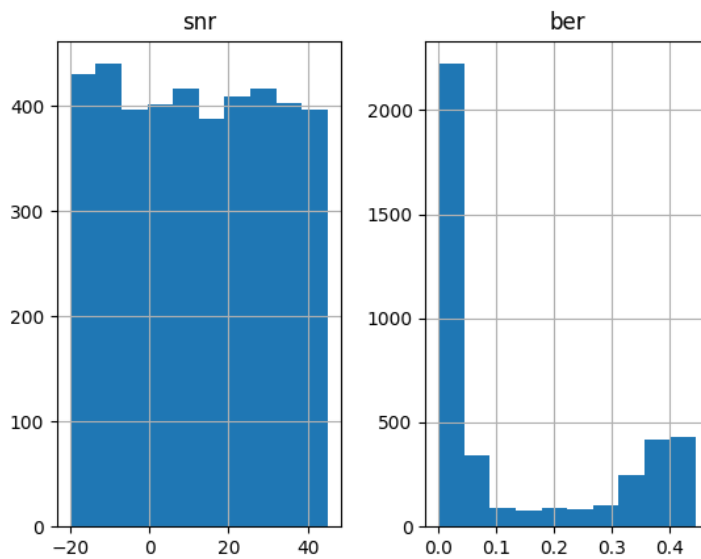
Σχήμα 3.5. box and whisker plots για κάθε χαρακτηριστικό

Αριθμητική αναπαράσταση του γραφήματος :

count	4096.000000	4096.000000
mean	12.115787	0.133415
std	18.846011	0.167493
min	-19.999589	0.000000
25%	-4.323706	0.000000
50%	11.868006	0.022945
75%	28.610822	0.330095
max	44.992880	0.447021

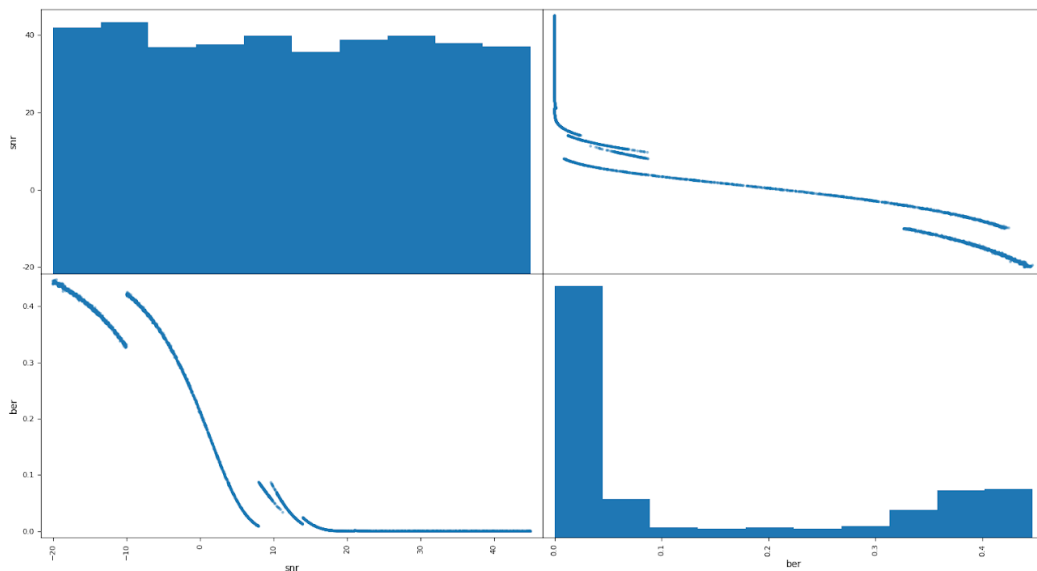
Η μέσος όρος για το SNR είναι περίπου 12 (στο διάγραμμα πράσινη γραμμή) ενώ όπως είναι αναμενόμενο η μικρότερη τιμή είναι -19.9 και η μέγιστη 44.9.

Στο επόμενο σχήμα παρουσιάζεται ένα ιστόγραμμα για τα χαρακτηριστικά μας. Δεν φαίνεται να ακολουθούν κάποια γνωστή κατανομή, οπότε δεν περιμένουμε κάποιος παραμετρικός αλγόριθμος (LR, LDA NB) να έχει την καλύτερη απόδοση.



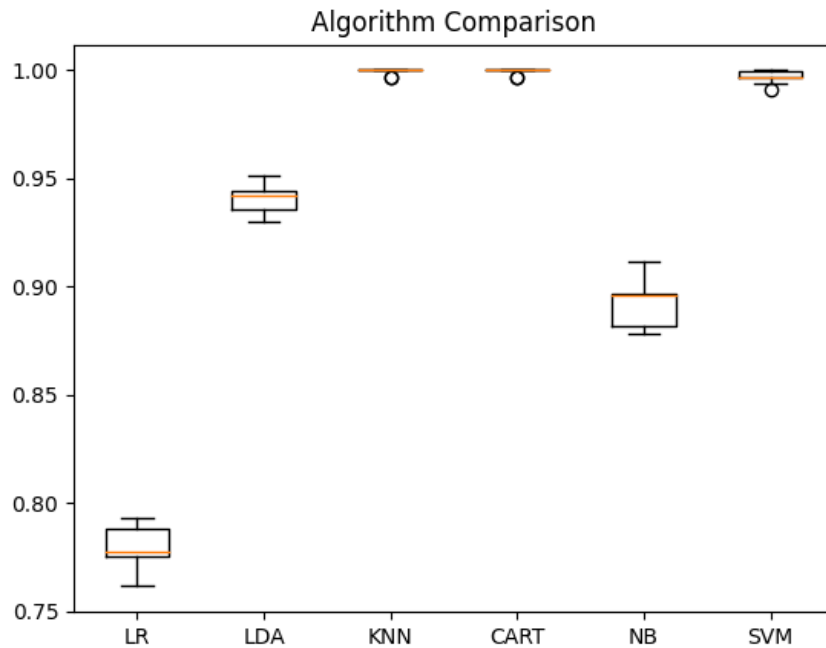
Σχήμα 3.6. ιστόγραμμα χαρακτηριστικών.

Ένα scatterplot φαίνεται στο επόμενο σχήμα. Φαίνεται να υπάρχει υψηλή συσχέτιση μεταξύ των χαρακτηριστικών SNR και BER. Αυτό είναι λογικό καθώς όσο το SNR είναι καλύτερο έχουμε λιγότερα λάθη.



Σχήμα 3.7. scatterplots

Εφόσον έχουμε τα δεδομένα θα εκπαιδεύσουμε το σύστημα μας. Όπως αναμέναμε και είναι εμφανές στο παρακάτω διάγραμμα οι παραμετρικοί αλγόριθμοι δεν έχουν καλή απόδοση.

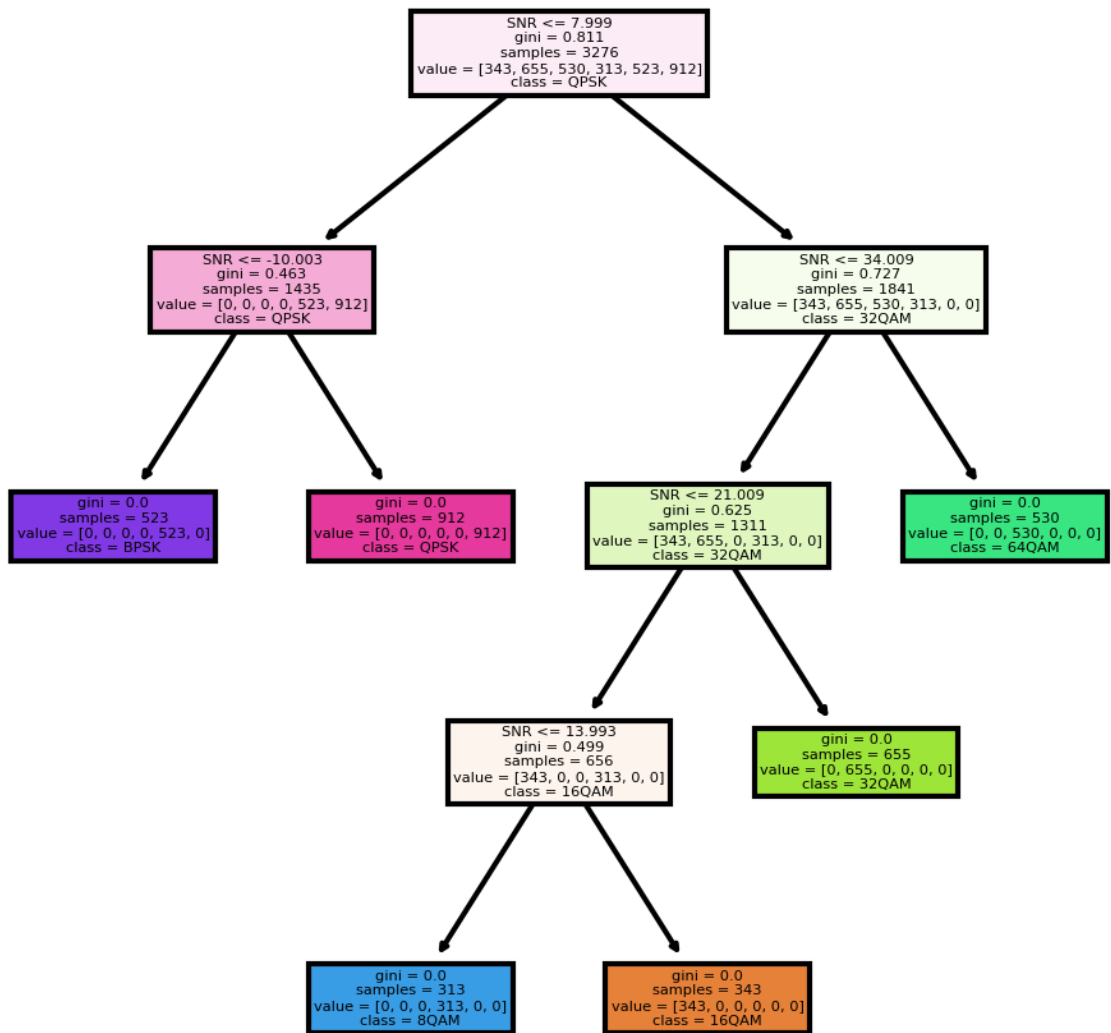


Σχήμα 3.8. box and whisker plots για κάθε αλγόριθμο

Το διάγραμμα με αριθμούς :

```
LR: 0.779602 (0.008896)
LDA: 0.940170 (0.006137)
KNN: 0.999389 (0.001221)
CART: 0.999389 (0.001221)
NB: 0.892863 (0.010979)
SVM: 0.996949 (0.002727)
```

Ο καλύτερος αλγόριθμος για αυτό το σύστημα είναι ο CART (δέντρα απόφασης) ένα σχηματικό διάγραμμα του , φαίνεται στο επόμενο σχήμα :



Σχήμα 3.9. Σχεδιάγραμμα CART

samples: Συνολικά δεδομένα προς εκπαίδευσης. Εφόσον το 20% είναι για δοκιμή δηλαδή 820 μένουν 3276.

Values: Δεδομένα κάθε κλάσης.

Αν ο δείκτης GINI είναι μηδέν σημαίνει ότι όλα τα δεδομένα ανήκουν σε μια κατηγορία έχουν δηλαδή ταξινομηθεί.

Από τα επόμενα αποτελέσματα διαπιστώνουμε ότι η εκπαίδευση του μοντέλου μας είναι ιδανική καθώς όλα τα κριτήρια απόδοσης (F1-score) είναι στο 1 .

```
0.998780487804878
[[ 90  0  0  0  0  0]
 [  0 161  0  0  0  0]
 [  0  0 144  0  0  0]
 [  0  0  0  73  0  0]
 [  0  0  0  0 133  1]
 [  0  0  0  0  0 218]]
      precision    recall  f1-score   support

   16QAM         1.00      1.00      1.00         90
   32QAM         1.00      1.00      1.00        161
   64QAM         1.00      1.00      1.00        144
    8QAM         1.00      1.00      1.00         73
    BPSK         1.00      0.99      1.00        134
    QPSK         1.00      1.00      1.00        218

 accuracy                   1.00         820
 macro avg           1.00      1.00      1.00         820
 weighted avg        1.00      1.00      1.00         820
```

Κεφάλαιο 4

Συμπεράσματα και προτάσεις για έρευνα

Στην παρούσα εργασία παρουσιάσαμε μεθόδους μηχανικής μάθησης εξετάζοντας τους αλγόριθμους που την αναδεικνύουν , ενώ τους συνδυάζαμε με σύγχρονες μεθόδους ασύρματων δικτύων δείχνοντας την αποτελεσματικότητα τους έναντι των συμβατικών τεχνικών. Τόσα τα ασύρματα δίκτυα όσο και οι απαιτήσεις των χρηστών (QOS) εξελίσσεται με ταχείς ρυθμούς έχοντας ως αποτέλεσμα νέες προκλήσεις να δημιουργούνται. Η αποτελεσματικότερη κατανομή πόρων φαίνεται να απασχολεί ολοένα και περισσότερους τους ερευνητές των ασύρματων συστημάτων ενώ φαίνεται πως η μηχανική μάθηση αποτελεί μονόδρομος για ένα σύγχρονο σύστημα επικοινωνιών. Η νέα γενιά δικτύων είναι βέβαια ότι θα αποτελείται από IOT (Internet of things) δηλαδή οποιαδήποτε συσκευή θα είναι συνδεδεμένη στο δίκτυο , κάνοντας ακόμα πιο ανταγωνιστικά τα ασύρματα δίκτυα , ενώ τεχνικές cloud computing θα συμβάλλουν σε αυτή. Πολλές τεχνικές μηχανικής μάθησης έχουν μελετηθεί ώστε να τα κάνουν εξυπνότερα τα δίκτυα, αποδίδοντας στον χρήστη την καλύτερη δυνατή εμπειρία , ωστόσο σε πολλές περιπτώσεις έχουν γίνει υποθέσεις που δεν ανταποκρίνονται σε πραγματικά σενάρια και χρήζουν περαιτέρω διερεύνησης. Όσο ο αριθμός των κεραιών αυξάνεται και μειώνεται η ακτίνας της κάθε κυψέλης έχουμε αύξηση της χωρητικότητας και του ρυθμού μετάδοσης , ωστόσο νέες προκλήσεις πρέπει να μελετηθούν όσα αναφορά την κατάλληλη επιλογή/αλλαγή κυψέλης (handover) καθώς και την καλύτερη εξοικονόμηση ενέργεια από πλευράς κινητής συσκευής καθώς οι μπαταρίες των κινητών συσκευών φαίνεται να υποφέρουν πλέον από κατανάλωση ενέργειας που μπορεί να δημιουργήσουν αντιθετα αποτελέσματα, ενώ σε αυτό έρχονται να συμβάλουν και τα ετερογενή δίκτυα. Όλες αυτές οι νέες προκλήσεις καθώς και εφαρμογή μεθόδων σε πραγματικά δεδομένα ώστε να αξιολογηθούν οι αλγόριθμοι της μηχανικής μάθησης χρήζουν περαιτέρω διερεύνηση.

Βιβλιογραφία και ηλεκτρονικές πηγές

1. Μηχανική Μάθηση , Κων. Διαμαντάρης , Δημ. Μπότσης.
2. https://en.wikipedia.org/wiki/Machine_learning
3. https://repository.kallipos.gr/bitstream/11419/3382/1/02_chapter_04.pdf
4. <https://www.csc.com.gr/machine-learning-%CE%BC%CE%B7%CF%87%CE%B1%CE%BD%CE%B9%CE%BA%CE%AE-%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7-%CF%84%CE%B9-%CE%B5%CE%AF%CE%BD%CE%B1%CE%B9/>
5. <http://www2.stat-athens.aueb.gr/~jpan/grammika-montela/chapter-III-7.pdf> (ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΠΟΛΥΣΥΓΓΡΑΜΜΙΚΟΤΗΤΑΣ)
6. Bishop - Pattern Recognition And Machine Learning
7. https://en.wikipedia.org/wiki/Multiclass_classification
8. https://en.wikipedia.org/wiki/Multi-label_classification
9. <https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/>
10. <https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148>
11. https://en.wikipedia.org/wiki/Sigmoid_function
12. https://en.wikipedia.org/wiki/Likelihood_function
13. <https://realpython.com/logistic-regression-python/>
14. <https://christophm.github.io/interpretable-ml-book/logistic.html>
15. https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html
16. https://datacamp.com/community/tutorials/understanding-logistic-regression-python?utm_source=adwords_ppc&utm_campaignid=898687156&utm_adgroupid=48947256715&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adposition=&utm_creative=332602034349&utm_targetid=aud-392016246653:dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=9061582&gclid=EAIaIQobChMIq8izsdGP8QIVhehRCh3sflw5EAAYASAAEgJnXPD_BwE
17. Αναγνώριση Προτύπων , Θεοδωρίδης, Κουτρούμπας
18. Kevin P Murphy. Machine learning: a probabilistic perspective. The MIT Press, 2012.
19. https://scikit-learn.org/stable/auto_examples/linear_model/plot_ridge_path.html
20. Roth, V. and V. Steinhage (2000). Nonlinear discriminant analysis using kernel functions.
21. <https://www.geeksforgeeks.org/ml-linear-discriminant-analysis/>
22. <https://www.knowledgehut.com/blog/data-science/linear-discriminant-analysis-for-machine-learning>
23. https://en.wikipedia.org/wiki/Linear_discriminant_analysis
24. https://scikit-learn.org/stable/modules/naive_bayes.html
25. https://en.wikipedia.org/wiki/Bayes%27_theorem
26. <https://towardsdatascience.com/5-minute-machine-learning-naive-bayes-f48472670fdd>
27. https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html

28. <https://www.knowledgehut.com/blog/data-science/naive-bayes-in-machine-learning>
29. <https://www.knowledgehut.com/blog/data-science/knn-for-machine-learning>
30. <https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning>
31. <https://scikit-learn.org/0.15/modules/neighbors.html#neighbors>
32. <https://scikit-learn.org/stable/modules/tree.html>
33. Machine learning Tom M. Mitchell
34. <https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052>
35. https://en.wikipedia.org/wiki/Decision_tree_learning
36. <https://www.knowledgehut.com/blog/data-science/classification-and-regression-trees-in-machine-learning>
37. <https://liyanxu.blog/2018/11/02/overview-svm-optimization/>
38. <https://people.eecs.berkeley.edu/~jordan/courses/281B-spring04/lectures/lec3.pdf>
39. https://en.wikipedia.org/wiki/Support_vector_machine
40. <https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm>
41. <https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python>
42. <https://scikit-learn.org/stable/modules/svm.html>
43. Τεχνητή Νοημοσύνη - Β' Έκδοση Ι. Βλαχάβας, Π. Κεφαλάς, Ν. Βασιλειάδης, Φ. Κόκκορας, Η. Σακελλαρίου
44. https://repository.kallipos.gr/bitstream/11419/2972/1/02_chapter_06.pdf
45. <https://medium.com/analytics-vidhya/how-to-determine-the-optimal-k-for-k-means-708505d204eb>
46. <https://en.wikipedia.org/wiki/DBSCAN>
47. <https://en.wikipedia.org/wiki/Q-learning>
48. https://el.wikipedia.org/wiki/%CE%9D%CE%B5%CF%85%CF%81%CF%89%CE%BD%CE%B9%CE%BA%CF%8C_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF
49. https://en.wikipedia.org/wiki/Artificial_neural_network
50. Νευρωνικά δίκτυα και μηχανική μάθηση , Simon Haykin
51. <https://medium.com/analytics-vidhya/multi-layer-perceptron-using-keras-on-mnist-dataset-for-digit-classification-problem-relu-a276cbf05e97>
52. https://en.wikipedia.org/wiki/Multilayer_perceptron
53. https://en.wikipedia.org/wiki/Vertical_handover
54. N. Ding et. al., “Characterizing and modeling the impact of wireless signal strength on smartphone battery drain,” ACM SIGMETRICS Perf..
55. Machine Learning Paradigms for Next-Generation Wireless Networks ,Chunxiao Jiang, Haijun Zhang, Yong Ren, Zhu Han, Kwang-Cheng Chen, and Lajos Hanzo
56. Machine Learning Regression-Based Approach for Dynamic Wireless Network Interface Selection , Lucas M. F. Harada and Daniel C. Cunha
57. SVM Multiregression for Nonlinear Channel Estimation in Multiple-Input Multiple-Output Systems ,Matilde Sánchez-Fernández, , Mario de-Prado-Cumplido,Jerónimo Arenas-García, Fernando , IEEE TRANSACTIONS ON SIGNAL PROCESSING
58. D. J. Sebald and J. A. Bucklew, “Support Vector Machines techniques for nonlinear

- equalization,” IEEE Trans. Signal Processing
59. https://en.wikipedia.org/wiki/Radial_basis_function_network
 60. https://en.wikipedia.org/wiki/Radial_function
 61. https://en.wikipedia.org/wiki/Radial_basis_function
 62. X. Zhou and X. Wang, “Channel estimation for OFDM systems using adaptive radial basis function networks,” IEEE Trans. Veh. Technol.
 63. https://en.wikipedia.org/wiki/Quasi-Newton_method
 64. An IRWLS procedure for SVR .F. Pérez-Cruz; A. Navia- Vázquez; P. L. Alarcón-Diana; A. Artés-Rodríguez ,IEEE
 65. Determination of Wireless Networks Parameters through Parallel Hierarchical Support Vector Machines Vin-sen Feng and Shih Yu Chang, Member, IEEE
 66. <http://doi.ieeecomputersociety.org/10.1109/TPDS.2011.156>
 67. S. Bi, R. Zhang, Z. Ding, and S. Cui, “Wireless Communications in the Era of Big Data,” IEEE Commun. Mag., vol. 53, no. 10, Oct. 2015, pp. 190–99.
 68. A Machine Learning Framework for Resource Allocation Assisted by Cloud Computing Jun-Bo Wang, Junyuan Wang, Yongpeng Wu, Jin-Yuan Wang, Huiling Zhu, Min Lin, and Jiangzhou Wang
 69. https://en.wikipedia.org/wiki/Polar_coordinate_system
 70. J. Wang et al., “Low-Complexity Beam Allocation for Switched-Beam Based Multiuser Massive MIMO Systems,” IEEE Trans. Wireless Commun..
 71. N. Kato et al., “The Deep Learning Vision for Heterogeneous Network Traffic Control: Proposal, Challenges, and Future Perspective,” IEEE Wireless Commun
 72. F. Tang et al., “On Removing Routing Protocol from Future Wireless Networks: A Real-Time Deep Learning Approach for Intelligent Traffic Control,” IEEE Wireless Commun.
 73. http://users.teiath.gr/vmouss/ebooks/optimee/docs/matlab_optimfunctions.pdf
 74. <http://www2.stat-athens.aueb.gr/~jpan/grammika-montela/chapter-III-7.pdf>
 75. <https://towardsdatascience.com/a-beginners-guide-to-machine-learning-5d87d1b06111>
 76. <https://en.wikipedia.org/wiki/Perceptron#Variants>
 77. https://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt_algorithm
 78. Pilot Based Channel Estimation Technique in MIMO-OFDM System
(<http://ijarece.org/wp-content/uploads/2015/01/IJARECE-VOL-4-ISSUE-1-110-117.pdf>
)
 79. X. Zhou and X. Wang, “Channel estimation for OFDM systems using adaptive radial basis function networks,” IEEE Trans. Veh. Technol., vol. 52, pp. 48–59, Jan. 2003.
 80. <https://scikit-learn.org/stable/modules/svm.html#svm-classification>
 81. http://www.eng.ucy.ac.cy/gmitsis/ece795/lectures/ECE795_Lectures19_20_2012.pdf
 82. <https://dSPACE.lib.ntua.gr/xmlui/bitstream/handle/123456789/43439/%ce%94%ce%b9%cf%80%ce%bb%cf%89%ce%bc%ce%b1%cf%84%ce%b9%ce%ba%ce%ae%20%ce%95%cf%81%ce%b3%ce%b1%cf%83%ce%af%ce%b1%20%ce%9c%ce%b9%cf%87%ce%b1%ce%b%ce%bf%cf%80%ce%bf%cf%8d%ce%bb%ce%bf%cf%85%20%ce%95.%20final3.pdf?sequence=1&isAllowed=y>
 83. <https://slideplayer.gr/slide/11365300/>

84. T.L. Jensen, S. Kant, J. Wehinger, and B.H. Fleury. Fast link adaptation for MIMO OFDM. *IEEE Transactions on Vehicular Technology*, 59 (8): 3766–3778, Oct 2010.
85. <https://el.wikipedia.org/wiki/%CE%91%CE%BD%CE%AC%CE%B4%CF%81%CE%B1%CF%83%CE%B7>
86. https://el.wikipedia.org/wiki/%CE%95%CF%80%CE%B9%CE%B2%CE%BB%CE%B5%CF%80%CF%8C%CE%BC%CE%B5%CE%BD%CE%B7_%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7
87. https://el.wikipedia.org/wiki/%CE%9C%CE%B7-%CE%B5%CF%80%CE%B9%CE%B2%CE%BB%CE%B5%CF%80%CF%8C%CE%BC%CE%B5%CE%BD%CE%B7_%CE%9C%CE%AC%CE%B8%CE%B7%CF%83%CE%B7
88. https://el.wikipedia.org/wiki/%CE%95%CE%BD%CE%B9%CF%83%CF%87%CF%85%CF%84%CE%B9%CE%BA%CE%AE_%CE%BC%CE%AC%CE%B8%CE%B7%CF%83%CE%B7
89. Machine Learning–Based Adaptive Modulation and Coding Design Lin Zhang and Zhiqiang Wu
90. ONLINE ADAPTIVE MODULATION AND CODING WITH SUPPORT VECTOR MACHINES ,Robert Daniels and Robert W. Heath, Jr.
91. T.L. Jensen, S. Kant, J. Wehinger, and B.H. Fleury. Fast link adaptation for MIMO OFDM. *IEEE Transactions on Vehicular Technology*
92. M. Aljuaid and H. Yanikomeroglu. Investigating the validity of the gaussian approximation for the distribution of the aggregate interference power in large wireless networks.
93. <https://scikit-learn.org/stable/index.html>
94. <http://aibook.csd.auth.gr/include/slides/Chap18.pdf>
95. <https://ieeexplore.ieee.org/document/4042295>
96. http://dione.lib.unipi.gr/xmlui/bitstream/handle/unipi/11028/Tsakarelos_Christos.pdf?sequence=1&isAllowed=y
97. Machine Learning–Based Nonlinear MIMO Detector Song-Nam Hong and Seonho Kim
98. Machine Learning–Based Nonlinear MIMO Detector ,Fa-Long Luo, 2020 IEEE (<https://ieeexplore.ieee.org/document/8958860>)
99. Marius Muja and David G. Lowe. Fast matching of binary features. In *Computer and Robot Vision* , 2012 IEEE.
100. Yo-Seb Jeon, Namyoon Lee, Song-Nam Hong, and Robert W. Heath. One-bit sphere decoding for uplink massive MIMO systems with one-bit ADCs. *IEEE Transactions on Wireless Communications*, 2018b.
101. Seonho Kim, Namyoon Lee, and S-N Hong. A low-complexity soft-output wmd decoding for uplink MIMO systems with one-bit ADCs. arXiv preprint arXiv:1707.02868, 2017.
102. https://en.wikipedia.org/wiki/Radial_basis_function
103. https://el.wikipedia.org/wiki/%CE%9D%CE%B5%CF%85%CF%81%CF%89%CE%BD%CE%B9%CE%BA%CF%8C_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF
104. <http://aibook.csd.auth.gr/include/slides/Chap19.pdf>
105. http://auto.teipir.gr/sites/default/files/dialexeis_tnd_3-5-2017.pdf

106. Back Propagation Neural Network Approach for Channel Estimation in OFDM System, Necmi Taspınar, M. Nuri Seyman.
107. L. J. Cimini Analysis and simulation of digital mobile channel using orthogonal frequency division multiplexing. IEEE
108. A. Jeremic, T.A. Thomas OFDM channel estimation in presence of interference, IEEE Trans. on Signal Proc., 2004
109. J. H. Wen, C. Y. Chang, G. R. Lee, C. Y. Huang, OFDM channel prediction using fuzzy update LMS algorithm in time-variant mobile channel In Proceedings of the IEEE 64th Vehicular Technology Conference, Montreal, Canada, September 2006
110. O. Simione, Y. Bar-Ness, U. Spagnolini. Pilot based channel estimation for OFDM systems by tracking the delay subspace, IEEE Transaction on Wireless Communications 2004
111. T. Cui, C. TeJlambura, Channel estimation for OFDM systems based on adaptive radial basis function network. In Proceeding of the IEEE 60th Vehicular Technology Conference, Los Angeles, USA, September 2004
112. J. J. Van de Beek, O. Edfords, M. Sandel, S. K. Wilson and P. O. Borjesson, On channel estimation in OFDM systems, Proc. of the IEEE 45th Veh. Technol Conf, Chicago, U.S.A, July 1995
113. M.T. Hagan, B. Menhaj, Training feed forward networks with the Marquardt algorithm, IEEE Trans. Neural Networks 5 (6) (1994) 989–993.
114. M.N. Seyman, N. Taspınar, Channel estimation based on adaptive neuro-fuzzy inference system in OFDM, IEICE Trans. Commun. E91-B (7) (2008) 2426–2430.
115. Channel estimation based on neural network in space time block coded MIMO–OFDM system . Muhammet Nuri Seyman, Necmi Taspınar
116. I. Harjula, A. Mammela, Channel estimation algorithm for space-time block coded OFDM system, in: Proc. of the IEEE Global Commun.
117. I. Barhumi, G. Leus, M. Moonen, Optimal training design for MIMO OFDM systems in mobile wireless channels, IEEE Trans. Signal Process
118. Sun, D.F. Yuan, Neural network channel estimation based on least mean error algorithm in the OFDM systems.
119. Deep neural network for OFDM multiplexing systems without cyclic prefix transmission. Viet-Hung Nguyen, Minh-Tuan Nguyen, and Yong-Hwa Kim
120. A Subspace Blind Channel Estimation Method for OFDM Systems Without Cyclic Prefix Sumit Roy, Senior Member, IEEE and Chengyang Li
121. Roy S and Li C 2002 A subspace blind channel estimation method for OFDM systems without cyclic prefix IEEE Trans. Wireless Commun
122. Sebastian R 2016 An overview of gradient descent optimization algorithm
123. Sutskever I, Martens J, Dahl G and Hilton G 2013 On the importance of initialization and momentum in deep learning 30th Int. Conf. on Machine Learning
124. LeCun Y, Bottou L, Orr GB and Muller KR 1998 Efficient BackProp, Neural Networks: Tricks of the Trade Neural Networks: Tricks of the Trade (Berlin: Springer)
125. Darken C, Chang J and Moody J 1992 Learning rate schedules for faster stochastic gradient search Neural Networks for Signal Process II IEEE.

126. A Survey of Downlink Non-Orthogonal Multiple Access for 5G Wireless Communication Networks , WEI Zhiqiang , YUAN Jinhong , Derrick Wing Kwan Ng , Maged ElKashlan , and DING Zhiguo
127. Non-orthogonal Multiple Access , Zekun Zhang , Haijian Sun , Xianfu Lei
128. Y. Liu, M. ElKashlan, Z. Ding, and G. K. Karagiannidis, “Fairness of user clustering in MIMO non-orthogonal multiple access systems,” IEEE Commun. Lett., vol. 20, no. 7, pp. 1465–1468, July 2016.
129. J. Mei, L. Yao, H. Long, and K. Zheng, “Joint user pairing and power allocation for downlink non-orthogonal multiple access systems,” in IEEE Proc. of Inter. Commun. Conf. (ICC), Kuala Lumpur, Malaysia, pp. 1–6, May 2016.
130. https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence
131. <https://en.wikipedia.org/wiki/Tensor>
132. Unsupervised Multiway Data Analysis: A Literature Survey , Evrim Acar , Bulent Yener
133. https://en.wikipedia.org/wiki/Tensor_reshaping
134. https://en.wikipedia.org/wiki/Softmax_function
135. Aceto, G.; Ciunzo, D.; Montieri, A.; Pescape, A. Mobile Encrypted Traffic Classification Using Deep Learning
136. https://en.wikipedia.org/wiki/Vanishing_gradient_problem
137. Recurrent Neural Networks (RNN) - The Vanishing Gradient Problem (<https://www.superdatascience.com/blogs/recurrent-neural-networks-rnn-the-vanishing-g-gradient-problem>)
138. <https://en.wikipedia.org/wiki/One-hot>
139. Why One-Hot Encode Data in Machine Learning? (<https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/>)
140. https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence
141. A. Jeremic, T.A. Thomas OFDM channel estimation in presence of interference, IEEE Trans. on Signal Proc., 2004
142. S. Coleri, M. Ergen, A. Puri, A. Bahai Channel estimation techniques based on pilot arrangement in OFDM systems IEEE Trans. on Broad., 2002.
143. J. J. Van de Beek, O. Edfords, M. Sandel, S. K. Wilson and P. O. Borjesson, On channel estimation in OFDM systems,
144. Back Propagation Neural Network Approach for Channel Estimation in OFDM System , Necmi Taspimar , M. Nuri Seyman
145. T. Cui, C. TeJlambura, Channel estimation for OFDM systems based on adaptive radial basis function network. IEEE
146. D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning internal representations by error propagation, D. Rumelhart and J. McClelland Editors, Parallel Data Processing, The M.L.T. Press, Cambridge, 1986.
147. https://en.wikipedia.org/wiki/Feedforward_neural_network
148. https://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt_algorithm
149. Levenberg–Marquardt Training , Hao Yu Auburn University Bogdan , M. Wilamowski Auburn University

150. https://en.wikipedia.org/wiki/Space%E2%80%93time_block_code
151. https://en.wikipedia.org/wiki/Space%E2%80%93time_code
152. https://en.wikipedia.org/wiki/Spatial_multiplexing
153. Channel estimation based on neural network in space time block coded MIMO–OFDM system , Muhammet Nuri Seyman , Necmi Ta , spinar
154. S.M. Alamouti, A simple transmit diversity scheme for wireless communications, IEEE J. Select. Areas Commun. (1998)
155. https://en.wikipedia.org/wiki/Hessian_matrix
156. https://en.wikipedia.org/wiki/Jacobian_matrix_and_determinant
157. Deep neural network for orthogonal frequency division multiplexing systems without cyclic prefix transmission ,Viet-Hung Nguyen, Minh-Tuan Nguyen, and Yong-Hwa Kim
158. Sebastian R 2016 An overview of gradient descent optimization algorithm
159. NOMA in 5G Systems: Exciting Possibilities for Enhancing Spectral Efficiency ,By S. M. Riazul Islam, Member, IEEE, Ming Zeng, Student Member, IEEE, Octavia A. Dobre, Senior Member, IEEE
160. Analysis of non-orthogonal multiple access for 5G , Yingmin Wang; Bin Ren; Shaohui Sun; Shaoli Kang; Xinwei Yue , IEEE
161. Non-Orthogonal Multiple Access: Common Myths and Critical Questions , Mojtaba Vaezi; Robert Schober; Zhiguo Ding; H. Vincent Poor ,IEEE
162. https://en.wikipedia.org/wiki/Non-orthogonal_frequency-division_multiplexing
163. Power Domain Non Orthogonal Multiple Access: A Review , Aiman Kassir; Rudzidatul Akmam Dziyauddin; Hazilah Mad Kaidi; Mohd Azri Mohd Izhar ,IEEE
164. Power-Domain Non-Orthogonal Multiple Access (NOMA) in 5G Systems: Potentials and Challenges ,S.M. Riazul Islam, Nurilla Avazov, Octavia A. Dobre, and Kyung-Sup Kwak
165. Uplink non-orthogonal multiple access for 5G wireless networks , Mohammed Al-Imari; Pei Xiao; Muhammad Ali Imran; Rahim Tafazolli ,IEEE
166. Code-domain Non Orthogonal Multiple Access for 5G networks , Mai Thi-Phuong Le
167. Kolda, T.G.; Bader, B.W. Tensor decompositions and applications. SIAM Rev. 2009, 51, 455–500
168. A Deep Learning Approach for MIMO-NOMA Downlink Signal Detection , Chuan Lin , Qing Chang1, and Xianxu Li
169. https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence
170. https://en.wikipedia.org/wiki/Cross_entropy
171. <https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/>
172. Hierarchical Support Vector Machines for Multi-Class Pattern Recognition ,Friedhelm Schwenker ,University of Ulm, D-89069 Ulm, Germany
173. Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning series) , Kevin P. Murphy
174. https://el.wikipedia.org/wiki/Global_Positioning_System
175. User Localization in an Indoor Environment Using Fuzzy Hybrid of Particle Swarm Optimization & Gravitational Search Algorithm with Neural Networks , Jayant G. Rohra1,

- Boominathan Perumal(&), Swathi Jamjala Narayanan, Priya Thakur, and Rajen B. Bhatt
176. <https://archive.ics.uci.edu/ml/datasets/Wireless+Indoor+Localization>
 177. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
 178. https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html
 179. <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>
 180. <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>
 181. <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>
 182. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html
 183. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.recall_score.html
 184. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html
 185. https://en.wikipedia.org/wiki/Box_plot
 186. <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.describe.html>
 187. https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.boxplot.html
 188. <https://el.wikipedia.org/wiki/%CE%99%CF%83%CF%84%CF%8C%CE%B3%CF%81%CE%B1%CE%BC%CE%BC%CE%B1>
 189. <https://www.marsja.se/pandas-scatter-matrix-pair-plot/>
 190. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html
https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html
 191. <https://en.wikipedia.org/wiki/Percentile>
 192. https://en.wikipedia.org/wiki/Fog_computing
 193. <https://www.geeksforgeeks.org/introduction-of-cloud-of-things-cot/>
 194. The Cloud-to-Thing Continuum Opportunities and Challenges in Cloud, Fog and Edge Computing Editors: Lynn, P.T., Mooney, J., Lee, B., Endo
 195. Unsupervised Machine Learning in 5G Networks for Low Latency Communications, Eren Balevi, Member, IEEE and Richard D. Gitlin
 196. <http://arnoldit.com/wordpress/2016/11/17/hard-and-soft-clustering-explained/>
 197. https://en.wikipedia.org/wiki/Voronoi_diagram
 198. https://en.wikipedia.org/wiki/Automatic_repeat_request
 199. https://en.wikipedia.org/wiki/Hybrid_automatic_repeat_request
 200. N. I. Sulieman, E. Balevi, K. Davaslioglu, and R. D. Gitlin, "Diversity and Network coded 5G fronthaul wireless networks for ultra-low latency communication
 201. Diversity maximal combining for transparent protocol with Cooperative Network Coding (CNC) IEEE ,Muhammad Iqbal; Suwadi; Wirawan; Rina Pudjiastuti