



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ
ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

Επιβλεπόμενη Μηχανική Μάθηση και το Πρόβλημα της Ταξινόμησης

Νάι Μαρία

Επιβλέπων: Φουσκάκης Δημήτριος

Διπλωματική Εργασία

Αθήνα
Οκτώβριος 2019

Ευχαριστίες

Θα ήταν παράλειψή μου αν δεν αναγνώριζα τη σημαντική βοήθεια που έλαβα κατά τη διάρκεια της εκπόνησης της διπλωματικής μου εργασίας. Θα ήθελα, λοιπόν, να ευχαριστήσω τον κ. Δημήτριο Φουσκάκη, Αναπληρωτή Καθηγητή του Ε.Μ.Π., για την γνώση που μου παρείχε, τις σημαντικές επισημάνσεις του αλλά και για το χρόνο που αφιέρωσε στην ορθή συγγραφή της εργασίας.

Αναμφισβήτητα σημαντική συμβολή, όχι μόνο στην εκπόνηση της παρούσας διπλωματικής εργασίας, αλλά και στην πορεία μου στις προπτυχιακές μου σπουδές, ήταν αυτή της οικογένειάς μου. Μια συμβολή η οποία ήταν όχι μόνο οικονομική αλλά πολύ περισσότερο ηθική και ψυχολογική, γιατί, βέβαια, όλες οι καταστάσεις στη ζωή περιλαμβάνουν στιγμές χαράς αλλά και απογοήτευσης.

Αθήνα, Οκτώβριος 2019
Μαρία Νάι

Περίληψη

Ο όρος ‘Τεχνητή Νοημοσύνη’ προέρχεται από τον βασικό στόχο να μπορέσει κάποτε η τεχνολογία να δημιουργήσει αυτοματοποιημένα συστήματα αποφάσεων. Στόχος αυτών είναι η επιτέλεση σύνθετων ενεργειών χωρίς την διαρκή παρέμβαση του ανθρώπου, με την ακολουθία λογικών ενεργειών που αποτελούν αποτέλεσμα μίας μηχανικής εκμάθησης ή βαθιάς μάθησης. Η Μηχανική Μάθηση μπορεί να περιγραφεί ως ένα σύνολο τεχνικών και εργαλείων που επιτρέπουν στους υπολογιστές να σκέφτονται δημιουργώντας μαθηματικούς αλγόριθμους που βασίζονται σε δεδομένα. Η βαθιά μάθηση είναι μια μορφή Μηχανικής Μάθησης.

Σκοπός της παρούσας διπλωματικής εργασίας είναι η ανάλυση σημαντικών αλγορίθμων επιβλεπόμενης Μηχανικής Μάθησης που επιλύουν το πρόβλημα της ταξινόμησης. Η ταξινόμηση λαμβάνει χώρα σε πολλές εφαρμογές επιστημονικών πεδίων όπως στην επιστήμη υπολογιστών, στην ιατρική, στην μετεωρολογία, στα οικονομικά και σε πολλά άλλα. Γίνεται λοιπόν κατανοητό το πόσο σημαντικό είναι η ανάπτυξη τεχνικών οι οποίες θα μπορούν να μας προσφέρουν αξιόπιστη ταξινόμηση.

Στο Κεφάλαιο 1 γίνεται μια εισαγωγή στην Τεχνητή Νοημοσύνη στην ιστορία αυτής και στους τρόπους με τους οποίους την προσεγγίζουμε. Το Κεφάλαιο 2 αναλύει το πρόβλημα της Μηχανικής Μάθησης δίνοντας ιδιαίτερη έμφαση στο πρόβλημα της ταξινόμησης. Στην συνέχεια, το Κεφάλαιο 3 πραγματεύεται την αξιολόγηση ταξινομητών. Αναλύεται το δίλημμα μεροληψίας-διασποράς και παρουσιάζονται μέθοδοι και κριτήρια βελτίωσης της απόδοσης αυτών των ταξινομητών. Τέλος, το Κεφάλαιο 4 ασχολείται με τεχνικές ταξινόμησης βασισμένες στην θεωρία του Bayes ενώ το 5 με αλγοριθμικές τεχνικές.

Abstract

The term “Artificial Intelligence” originates from the technology’s main target to create automatic decision platforms. Such systems aim to execute complex actions via a sequence of logic operations stemming from a Machine Learning procedure, without the need of constant human intervention. Artificial intelligence can be described as a group of techniques and tools, which allow computers to “think” by creating mathematical algorithms based on data mining. A class of such algorithmic techniques is known as deep learning, which belongs in the general category of Machine Learning.

The purpose of the present diploma thesis is to expand on the most important supervised machine learning techniques, which solve the classification problem. Classification takes place in many applications within different scientific fields, including computer science, medicine, meteorology, and economics, to name a few. As a result, it is quite essential to develop techniques, which can offer reliable classification.

In Chapter 1, we provide an introduction to the history of Artificial Intelligence, along with the most important algorithmic approaches. In Chapter 2, we analyze the problem of Machine Learning, by emphasizing especially on the classification problem. In Chapter 3, an evaluation of the different classifiers is performed while methods and criteria are also provided for their improvement. The bias-variance dilemma is also analyzed. Finally, Chapter 4 deals with classification techniques based on Bayes’ theorem, while Chapter 5 emphasizes on respective algorithmic techniques.

Περιεχόμενα

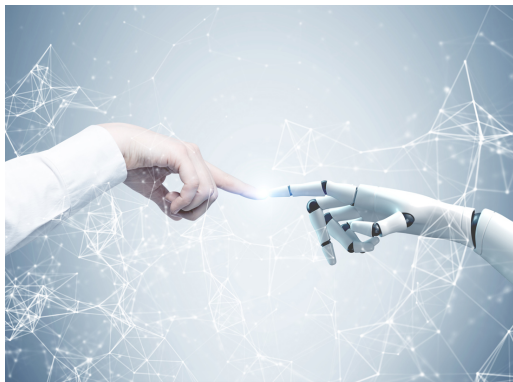
1	Τεχνητή Νοημοσύνη	7
1.1	Εισαγωγή	7
1.2	Ανθρώπινη-Τεχνητή Νοημοσύνη	8
1.3	Ιστορική Αναδρομή	8
1.4	Προσεγγίσεις	10
2	Μηχανική Μάθηση και το πρόβλημα της ταξινόμησης	12
2.1	Μηχανική Μάθηση	12
2.1.1	Εισαγωγή	12
2.1.2	Εφαρμογές Μηχανικής Μάθησης	13
2.1.3	Βασικές έννοιες	15
2.1.4	Διαρροή δεδομένων και Μεροληψία δειγμάτων	16
2.1.5	Τεχνικές και προβλήματα Μάθησης	17
2.2	Πρόβλημα ταξινόμησης	19
2.2.1	Εισαγωγή	19
2.2.2	Ορισμός του προβλήματος	19
2.2.3	Περιγραφή διαδικασίας	20
2.2.4	Πρόβλημα πολλών διαστάσεων	23
3	Αξιολόγηση απόδοσης ταξινομητών	24
3.1	Εισαγωγή	24
3.2	Αντιστάθμισμα μεροληψίας και διακύμανσης	24
3.2.1	Αποσύνθεση αναμενόμενης τιμής σφάλματος στην παλινδρόμηση	25
3.2.2	Έννοια σφάλματος στην ταξινόμηση	28
3.2.3	Αποσύνθεση αναμενόμενης τιμής σφάλματος στην ταξινόμηση .	30
3.3	Μέθοδοι βελτίωσης του μοντέλου για υψηλότερη απόδοση	31
3.4	Κριτήρια απόδοσης ταξινομητή	33
4	Τεχνικές ταξινόμησης βασισμένες στην θεωρία αποφάσεων κα- τά Μπέυζ	37
4.1	Μπεϋζιανοί ταξινομητές	37
4.1.1	Εισαγωγή	37
4.1.2	Θεωρία αποφάσεων κατά Μπέυζ	38
4.1.3	Συναρτήσεις διάκρισης και επιφάνειες απόφασης	41
4.1.4	Τετραγωνική και Γραμμική Διαχωριστική ανάλυση	42
4.1.5	Αφελής ταξινομητής Μπέυζ	47
4.1.6	Μπεϋζιανά δίκτυα	51
4.2	Ταξινομητές Πλησιέστερου Γείτονα	55
4.2.1	Εισαγωγή	55

4.2.2	Ταξινομητής K Πλησιέστερων Γειτόνων	55
5	Αλγοριθμικές τεχνικές ταξινόμησης	61
5.1	Δέντρα Ταξινόμησης	61
5.1.1	Εισαγωγή	61
5.1.2	Βασική μεθοδολογία	62
5.1.3	Επιλογή χαρακτηριστικών	62
5.1.4	Αλγόριθμοι εκμάθησης	66
5.2	Μηχανές Διανυσμάτων Υποστήριξης	69
5.2.1	Εισαγωγή	69
5.2.2	Γραμμικώς διαχωρίσιμα δεδομένα	70
5.2.3	Τέχνασμα του πυρήνα	75
5.2.4	Ταξινομητές Εξασθενημένου Περιθωρίου	78
5.3	Τεχνητά Νευρωνικά Δίκτυα	83
5.3.1	Εισαγωγή	83
5.3.2	Βιολογική έμπνευση	83
5.3.3	Μοντέλο τεχνητού νευρώνα	85
5.3.4	Αριχίτεκτονική δικτύων	87
5.3.5	Νευρωνικά Δίκτυα Πρόσθιας τροφοδότησης	89
5.3.6	Εκπαίδευση	91
5.4	Λογιστική Παλινδρόμηση	101
5.4.1	Εισαγωγή	101
5.4.2	Λογιστικό μοντέλο	102
6	Επίλογος	107
	Βιβλιογραφία	109

Κεφάλαιο 1

Τεχνητή Νοημοσύνη

1.1 Εισαγωγή

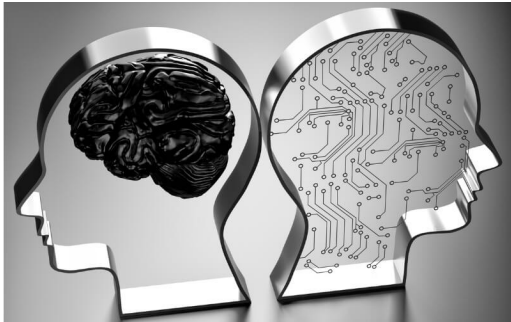


Η ανάπτυξη της τεχνολογίας επιτυγχάνεται εντυπωσιακά, με ρυθμούς γεωμετρικής προόδου. Σημαντικό ρόλο σε αυτό έχει διαδραματίσει στην σύγχρονη εποχή η εξέλιξη της **Τεχνητής Νοημοσύνης (TN)/Artificial Intelligence (AI)**. Αντικείμενο της TN είναι η υλοποίηση αλγορίθμων, οι οποίοι μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς (μάθηση, προσαρμοστικότητα, ευφυής ανάλυση, εξαγωγή συμπερασμάτων). Πιο συγκεκριμένα, στόχος της είναι όχι μόνο

η προσπάθεια εξομοίωσης των βασικών λειτουργιών ενός βιολογικού νευρώνα (με την βοήθεια υπολογιστικών συστημάτων), αλλά και η εύρεση της αλληλοσυσχέτισης μεταξύ ενός συνόλου αυτών. Με τον τρόπο αυτό, αναπαράγεται νοημοσύνη, αντίστοιχη με την ανθρώπινη αλλά τεχνητή, δηλαδή Τεχνητή Νοημοσύνη.

Η Τεχνητή Νοημοσύνη άρχισε να εξελίσσεται σταδιακά από την δεκαετία του 2000. Βασίζεται στην ανάπτυξη αλγορίθμων εκμάθησης (**machine learning**) που αξιοποιούν ένα τεράστιο όγκο εισερχόμενων δεδομένων (**big data analysis**) από τον χρήστη/περιβάλλον, με στόχο την εξεύρεση επαναλαμβανων μοτίβων και συσχετίσεων. Οι εφαρμογές της Τεχνητής Νοημοσύνης είναι πολλές και ιδιαίτερα εντυπωσιακές. Για παράδειγμα, το πρωτότυπο Seeing AI της Microsoft βοηθά ανθρώπους με προβλήματα όρασης να εξερευνούν το περιβάλλον γύρω τους και να εμπλουτίζουν την εμπειρία τους καθώς κινούνται. Η οπτική αναγνώριση χαρακτήρων του Google translate οδηγεί στην μετατροπή εικόνων σε κείμενα αναγνώσιμα από τον υπολογιστή. Επίσης, στον ιατρικό τομέα, τεχνικές AI χρησιμοποιούνται για την ανίχνευση καρκίνου σε απεικονίσεις με μαγνητική τομογραφία, με ακρίβεια παρόμοια αυτής των καταρτισμένων ακτινολόγων. Γενικά, η τεχνητή νοημοσύνη έχει βρει εφαρμογές σε μια πληθώρα εμπορικών, βιομηχανικών, επιστημονικών κλάδων γεγονός που έχει εντείνει τα τελευταία χρόνια τις ερευνητικές προσπάθειες προς αυτό τον κλάδο της επιστήμης των υπολογιστών.

1.2 Ανθρώπινη-Τεχνητή Νοημοσύνη



Νοημοσύνη ή ευφυΐα είναι ένα σύνολο λειτουργιών που επιτρέπουν στον άνθρωπο να αναπαριστά με ακρίβεια το περιβάλλον του και να κατανοεί τα μηνύματα που δέχεται. Ορισμένες από αυτές τις λειτουργίες περιλαμβάνουν την αντίληψη, την μνήμη, τη λογική σκέψη, και τη συνειδηση, για να αναφερθούν μερικές. Σύμφωνα με τον Howard Gardner [18], η νοημοσύνη είναι πολυεπίπεδη και μπορεί να διακριθεί σε λογικομαθηματική, μουσική,

συναισθηματική, αλλά και διαπροσωπική. Συνεπώς, η αξιολόγηση της δεν μπορεί να βασιστεί αποκλειστικά σε IQ tests, που εστιάζουν στην εξέταση της γλώσσας και της λογικής.

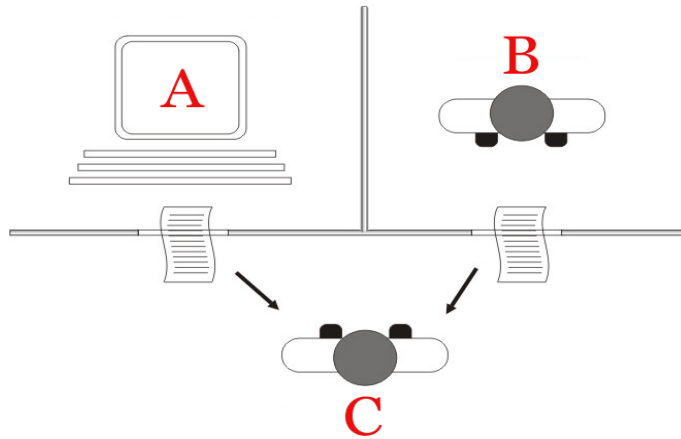
Το εγχείρημα με το οποίο ασχολείται η έρευνα πάνω στην Τεχνητή Νοημοσύνη, αφορά την προσπάθεια μεταφοράς ορισμένων εκ των λειτουργιών του ανθρώπινου στις μηχανές, ώστε αυτές να αναπτύξουν μια ευφυή συμπεριφορά. Συνδυάζοντας σύνολα κανόνων, οι «ευέλικτες» μηχανές εκπαιδεύονται δυναμικά και συνεχώς προκειμένου να μεγιστοποιήσουν τις πιθανότητες επίτευξης ενός συγκεκριμένου στόχου με το μικρότερο δυνατό τελικό κόστος (πχ. ελαχιστοποίηση υπολογιστικού χρόνου).

Η Τεχνητή Νοημοσύνη έχει αναγνωριστεί από την ευρεία επιστημονική κοινότητα σαν ένας ξεχωριστός τομέας της Επιστήμης των Υπολογιστών, που έχει ως αντικείμενο την σχεδίαση και την υλοποίηση προγραμμάτων με απώτερο σκοπό την απομίμηση της ανθρώπινης αντίληψης, τη λογική σκέψης, αλλά και λοιπών ανεπτυγμένων γνωστικών ικανοτήτων. Με αυτόν τον τρόπο, μια «έξυπνη» μηχανή μπορεί να αναπτύξει «αντίληψη», να αλληλεπιδράσει με το περιβάλλον, αλλά και να επιτύχει την επίλυση πολύπλοκων διεπιστημονικών προβλημάτων μέσω της αναγνώρισης σύνθετων προτύπων. Η αλληλεπίδραση αυτή μεταξύ ανθρώπινης και τεχνητής νοημοσύνης μπορεί μάλιστα να οδηγήσει στην εξεύρεση πολύπλοκων μοτίβων και μοντέλων που έχει ήδη αξιοποιηθεί στις χρηματιστηριακές αγορές, στην αστρονομία αλλά και στη κβαντική φυσική.

1.3 Ιστορική Αναδρομή

Η ιστορία της σύγχρονης Τεχνητής Νοημοσύνης ξεκινά το 1950 από τον Άγγλο μαθηματικό Alan Turing [55] εισήγαγε την μέθοδο του Turing test για την αξιολόγηση της νοημοσύνης των 'ευφυών' μηχανών. Βάσει αυτής της μεθοδολογίας, ένας ανθρώπινος κριτής (παίκτης γ) καλείται να ξεχωρίσει μεταξύ μιας μηχανής (παίκτη α) και ενός ανθρώπου (παίκτη β), έχοντας δεδομένες τις γραπτές συνδιαλέξεις μεταξύ μηχανής-ανθρώπου. Αν ο εξωτερικός παρατηρητής δεν μπορεί αξιόπιστα να καθορίσει ποιος από τους παίκτες α,β είναι υπολογιστής, τότε η μηχανή θα έχει περάσει το τεστ και θα

θεωρείται ευφυής (Διάγραμμα 1.1).



Διάγραμμα 1.1: Αναπαράσταση του “Turing test”.

Η διατύπωση του όρου Τεχνητή Νοημοσύνη έγινε για πρώτη φορά το 1956 από τον John McCarthy κατά την διάρκεια ενός συνεδρίου στο Dartmouth των ΗΠΑ. Ο ίδιος δύο χρόνια αργότερα ανέπτυξε την δεύτερη γλώσσα υψηλού επιπέδου ιδανική για έρευνα στην Τεχνητή Νοημοσύνη (LISP). Το 1965 κατασκευάζεται στο MIT το πρώτο διαδραστικό πρόγραμμα Τεχνητής Νοημοσύνης με το όνομα Eliza το οποίο μπορεί να μιμηθεί διαλόγους στα αγγλικά ενώ το 1982 μπαίνει σε λειτουργία το πρώτο σύστημα αυτόματης αναγνώρισης δαχτυλικών αποτυπωμάτων κατασκευασμένο από την NEC. Ορόσημο στην εξέλιξη της Τεχνητής Νοημοσύνης αποτέλεσε η νίκη του υπερυπολογιστή Deep Blue επί του παγκόσμιου πρωταθλητή στο σκάκι Garry Kasparov (1997). Το υπολογιστικό αυτό σύστημα της IBM, βασιζόμενο σε ένα σύνολο οριοθετημένων κανόνων, μπορούσε να πραγματοποιήσει 200.000.000 κινήσεις ανά δευτερόλεπτο με σκοπό να υλοποιήσει την βέλτιστη στρατηγική νίκης. Παρόλο αυτά η νίκη επί του Kasparov δεν αποτελούσε σαφή ένδειξη μηχανικής ευφυίας γιατί ο προγραμματισμός του υπερυπολογιστή Deep Blue βασιζόταν σε ένα σύνολο οριοθετημένων κανόνων.

Ιδιαίτερη εξέλιξη παρουσιάζει τα τελευταία χρόνια ο κλάδος της Τεχνητής Νοημοσύνης με την ανάπτυξη του πρώτου αυτοοδηγούμενου αμαξίου από την Google (2009) αλλά και την πραγματοποίηση της πρώτης χειρουργικής επέμβασης απο ρομπότ στον Καναδά (2010) . Ένα χρόνο αργότερα, ο υπολογιστής Watson της IBM, καταφέρνει να κερδίσει σε διαγωνισμό τους νικητές του γνωστού παιχνιδιού γνώσεων της αμερικάνικης τηλεόρασης, Jeopardy. Στην συνέχεια, το 2011-2014 δημιουργήθηκαν προγράμματα όπως το Siri της Apple, το Google’s Google Now και το Cortana της Microsoft τα οποία παίρνουν βασικές αποφάσεις, κάνουν συστάσεις, μπορούν να δώσουν λύσεις σε προβλήματα, να απαντήσουν σε ερωτήσεις και να επικοινωνήσουν σε φυσική γλώσσα (Διάγραμμα 1.2).

Σημαντικό επίτευγμα στην εξέλιξη της Τεχνητής Νοημοσύνης είναι ένα ‘έξυπνο’, αναβαθμισμένο σύστημα που παρουσίασε η Deep Mind θυγατρική εταιρία της Google(2017). Το AlphaZero, όπως ονομάστηκε, είναι αυτοδίδακτο, σε αντίθεση με το AlphaGo που έχει εκπαιδευτεί αναλύοντας εκατομμύρια παιχνίδια ανθρώπων. Το νέο αυτό σύστημα κατάφερε να γίνει ειδήμων στο κινέζικο επιτραπέζιο παιχνίδι Go,

Conversation 4: Philosophical Q&A	Conversation 5: Morality
Human: <i>what is the purpose of life ?</i>	Human: <i>what is moral ?</i>
Machine: <i>to serve the greater good .</i>	Machine: <i>what empowered humanity , what intellectual the essence is .</i>
Human: <i>what is the purpose of living ?</i>	Human: <i>what is immoral ?</i>
Machine: <i>to live forever .</i>	Machine: <i>the fact that you have a child .</i>
Human: <i>what is the purpose of existence ?</i>	Human: <i>what is morality ?</i>
Machine: <i>to find out what happens when we get to the planet earth .</i>	Machine: <i>what is altruism ?</i>
Human: <i>where are you now ?</i>	Human: <i>what is the definition of altruism ?</i>
Machine: <i>i 'm in the middle of nowhere .</i>	Machine: <i>if you don 't believe in god , then you don 't know .</i>
Human: <i>what is the purpose of dying ?</i>	Human: <i>ok ... so what is the definition of morality ?</i>
Machine: <i>to have a life .</i>	Machine: <i>well , the truth is . you 're not a believer in god almighty .</i>
Human: <i>what is the purpose of being intelligent ?</i>	Human: <i>tell me the definition of morality , i am quite upset now !</i>
Machine: <i>to find out what it is .</i>	Machine: <i>i 'm not ashamed of being a philosopher !</i>
Human: <i>what is the purpose of emotions ?</i>	Human: <i>tell me the definition of morality .</i>
Machine: <i>i don 't know .</i>	Machine: <i>i don 't have ethics .</i>
	Human: <i>look , i need help , i need to know more about morality ...</i>
	Machine: <i>i don 't know what ethics is .</i>
	Human: <i>seriously , what is morality ?</i>
	Machine: <i>what is the definition of living ?</i>
	Human: <i>why living has anything to do with morality ?</i>
	Machine: <i>you 're not a cop .</i>
	Human: <i>okay , i need to know how should i behave morally ...</i>

Διάγραμμα 1.2: Αλληλεπίδραση μηχανής-ανθρώπου [26].

χωρίς να τροφοδοτηθεί με άλλες πληροφορίες πέρα από αυτές για τους κανόνες του παιχνιδιού. Τέλος, μόλις ένα χρόνο πριν, το Ινστιτούτο Data Science and Technologies της Alibaba ανακοίνωσε ότι το μοντέλο τεχνητής νοημοσύνης καταγράφει υψηλότερο σκορ από τα καλύτερα μυαλά του Stanford University στο τεστ (100.000 ερωτήσεων) ανάγνωσης και κατανόησης. Τα επιτεύγματα αυτά είναι εύκολο να μας πείσουν ότι οι μηχανές κατάφεραν να ξεπεράσουν την νοημοσύνη των κατασκευαστών τους σε συγκεκριμένες εργασίες.

1.4 Προσεγγίσεις

Η Τεχνητή Νοημοσύνη εκτός από την επιστήμη της πληροφορικής συνδέεται και με άλλες επιστήμες όπως της ψυχολογίας, της νευρολογίας, της επιστήμης μηχανικών, με τελικό στόχο τη σύνθεση ευφυούς συμπεριφοράς. Συνήθως εφαρμόζεται σε μηχανές ή υπολογιστές ειδικής κατασκευής. Διακρίνονται οι παρακάτω προσεγγίσεις της ΤΝ:

Η **συμβολική ΤΝ** (symbolic AI) που άρχισε να αναπτύσσεται στην δεκαετία του 1960, μελετά την ανθρώπινη συμπεριφορά και προσπαθεί να την προσεγγίσει αλγοριθμικά χρησιμοποιώντας σύμβολα και λογικούς κανόνες υψηλού επιπέδου. Παράδειγμα της προσεγγίσης αυτής είναι ο υπερυπολογιστής Deep Blue της IBM ο οποίος κατάφερε να νικήσει τον παγκόσμιο πρωταθλητή στο σκάκι Garry Kasparov. Πιο συγκεκριμένα, ο Deep Blue βασίστηκε σε ένα αλγόριθμο που αξιολογεί εκατομμύρια δυνατότητες ανά δευτερόλεπτο και επιλέγει την πιο ελπιδοφόρα κίνηση. Στην πραγματικότητα δεν είχε σκακιστική ευφυία αφού οι ικανότητές του προέρχονταν από τον υπολογισμό των κινήσεων και όχι από την κατανόηση των στρατηγικών του παιχνιδιού.

Ο προγραμματισμός ενός υπολογιστή όμως απαιτεί τον λεπτομερή καθορισμό

κάθε βήματος ξεχωριστά που πρέπει ο υπολογιστής να ακολουθήσει προκειμένου να λύσει κάποιο πρόβλημα. Η προσέγγιση αυτή έχει οδηγήσει σε μέτρια αποτελέσματα αφού η καταγραφή μιας πολύ καλώς ορισμένης ακολουθίας βημάτων που θα περιγράφει επακριβώς πολύπλοκες ανθρώπινες διεργασίες είναι δύσκολο έργο. Έτσι, λόγω της αδυναμίας του ανθρώπου να αποκωδικοποιήσει στοιχεία της νοημοσύνης ώστε να γίνονται αντιληπτά από μια μηχανή, δημιουργήθηκε η παρακάτω προσέγγιση.

Η **μη συμβολική ΤΝ** (non symbolic AI) προσπαθεί να αναπαράγει την ευφυΐα του ανθρώπου, δηλαδή να μιμηθεί τις βιολογικές διεργασίες με υπολογιστικές τεχνικές. Στην κατηγορία αυτή ανήκουν τα νευρωνικά δίκτυα και οι γενετικοί αλγόριθμοι. Σύμφωνα με την προσέγγιση αυτή οι μηχανές προχωρούν στην ανάλυση μεγάλων ποσοτήτων δεδομένων ώστε να εντοπίσουν στατιστικούς συσχετισμούς. Παράδειγμα της υπολογιστικής νοημοσύνης, όπως αλλιώς λέγεται, είναι το 'έξυπνο' σύστημα της Deep Mind, Alpha Go το οποίο είχε εκπαιδευτεί αναλύοντας εκατομμύρια παιχνίδια ανθρώπων, νικώντας έτσι κορυφαίους παίκτες του παιχνιδιού Go.

Ανάλογα με το επίπεδο νοημοσύνης που εμφανίζουν οι μηχανές, η Τεχνητή Νοημοσύνη διακρίνεται:

Στην **ασθενή ΤΝ** (weak AI) η μηχανή είναι ένα καλώς σχεδιασμένο εργαλείο που εκτελεί συγκεκριμένες εργασίες, λειτουργεί υπό τις εντολές του προγραμματιστή δίχως να έχει την δυνατότητα αντίληψης. Έως τώρα οι εφαρμογές της AI ανήκουν σε αυτή την κατηγορία. Τον τελευταίο καιρό, έχει σημειωθεί σημαντική πρόοδος στον τομέα της Ασθενούς Τεχνητής Νοημοσύνης, κυρίως χάρη στην αύξηση της υπολογιστικής ισχύος των ηλεκτρονικών υπολογιστών, στη διάθεση μεγάλου όγκου δεδομένων και στην ανάπτυξη της Μηχανικής Μάθησης.

Στην **γενική ΤΝ** (general AI) η μηχανή είναι σε θέση να εκτελέσει όλες τις διανοητικές λειτουργίες που μπορεί να εκτελέσει ο άνθρωπος, εμφανίζει σημάδια δημιουργικότητας αφού έχοντας τα κατάλληλα προγράμματα μπορεί να 'μεταφέρει' την εμπειρία από μια νοητική λειτουργία που εκτελεί σε μια άλλη. Η μορφή αυτή δεν αποτελεί προσομοίωση αλλά αναπαραγωγή της ανθρώπινης νοημοσύνης.

Η Τεχνητή Νοημοσύνη είναι σχετικά μια από τις νεότερες επιστήμες καθώς πραγματική έρευνα γύρω από αυτή άρχισε το 1956. Αυτή που βιώνουμε σήμερα και ενσωματώνεται με γρήγορους ρυθμούς σχεδόν σε όλες τις καθημερινές μας δραστηριότητες είναι η ασθενής ΤΝ καθώς τα συστήματα AI που σήμερα χρησιμοποιούμε εμφανίζουν ένα ορισμένο βαθμό νοημοσύνης σε ένα συγκεκριμένο πεδίο εκτελώντας εξαιρετικά εξειδικευμένες εργασίες. Στο επόμενο κεφάλαιο γίνεται μια γενική ανασκόπηση της Μηχανικής Μάθησης, υποπεδίο της Τεχνητής Νοημοσύνης που αναπτύχθηκε για την μελέτη αναγνώρισης προτύπων με την κατασκευή αλγορίθμων που μαθαίνουν μέσα από τα δεδομένα, οι τεχνικές της οποίας θα αναλυθούν στο Κεφάλαιο 4.

Κεφάλαιο 2

Μηχανική Μάθηση και το πρόβλημα της ταξινόμησης

2.1 Μηχανική Μάθηση

2.1.1 Εισαγωγή

Ως επιστήμη η **Μηχανική Μάθηση (Machine Learning)** γεννήθηκε μέσα από την Τεχνητή Νοημοσύνη. Τα τελευταία χρόνια μια από τις μεγαλύτερες προκλήσεις είναι η αξιοποίηση του τεράστιου όγκου δεδομένων που παράγονται από την ανάπτυξη του ψηφιακού κόσμου. Ενδεικτικά, δημιουργούνται πάνω από 2,5 τετράκις εκατομμύρια bytes δεδομένων ανά έτος, σύμφωνα με την IBM. Στο εγχείρημα αυτό συμβάλλει σε μεγάλο βαθμό η Μηχανική Μάθηση, η οποία σε συνδιασμό με τα μεγάλα δεδομένα οδηγεί στην εξαγωγή χρήσιμων πληροφοριών. Σήμερα, η εκμάθηση μηχανών διαδραματίζει καίριο λόγο στην αναγνώριση προσώπων/ αντικειμένων σε εικόνες, στον εντοπισμό ηλεκτρονικής απάτης, στην ομαδοποίηση πελατών καταστημάτων με στόχο την στοχευμένη διαφήμιση, στην πρόγνωση καρδιοπάθειας μέσω ανάλυση εικόνων MRI, δίνοντας έτσι λύσεις σε πολλά προβλήματα της Επιστήμης και της βιομηχανίας. Είναι επίσης γεγονός πως πολλές εταιρίες όπως οι Apple, Alphabet Inc, eBay, Google αξιοποιούν την Μηχανική Μάθηση σε μεγάλο βαθμό σε αρκετά από τα προϊόντα τους. Τέλος, παρόλο που οι μηχανές μάθησης μπορούν να προσομοιώσουν την ανθρώπινη ευχέρεια για γενίκευση, απομνημόνευση, πρόβλεψη, η επιστήμη απέχει από το να πετύχει οτιδήποτε παρόμοιο με την ανθρώπινη ευφυΐα.

Στη Μηχανική Μάθηση, οι υπολογιστές εφαρμόζουν τεχνικές στατιστικής μάθησης για να αναγνωρίσουν αυτόματα μοτίβα στα δεδομένα. Οι τεχνικές αυτές μπορούν να χρησιμοποιηθούν για να δημιουργηθούν πολύ ακριβείς προβλέψεις. Στην αρχή του κεφαλαίου αυτού, αναφέρονται κάποιες χρήσιμες εφαρμογές της Μηχανικής

Μάθησης, εισάγονται βασικές τεχνικές και προβλήματα αυτής. Τέλος, δίνεται έμφαση στο πρόβλημα της ταξινόμησης και στην διαδικασία της, γίνεται αναφορά στις έννοιες υπερπροσαρμοστικότητα και διαστατικότητα που θα μας απασχολήσουν στην συνέχεια.

2.1.2 Εφαρμογές Μηχανικής Μάθησης

Ερευνητές της Τεχνητής Νοημοσύνης και ειδικότερα της Μηχανικής Μάθησης συνεργάζονται με επιστήμονες πολλών ακαδημαϊκών κλάδων και προγραμματιστές αναζητώντας λύσεις σε μια σειρά από προβλήματα, έχοντας προσφέρει μέχρι σήμερα πολύ ικανοποιητικά αποτελέσματα. Έτσι έχουν αναπτυχθεί τεχνικές που έχουν προοπτικές για νέες εφαρμογές σε ένα μεγάλο εύρος πεδίων. Ο πλούτος των δεδομένων σε συνδυασμό με τις δυνατότητες αξιοποίησης τους έχει δημιουργήσει πρωτοφανείς ευκαιρίες για την βελτίωση της υγείας, της καθημερινότητας και του τρόπου ζωής, των μεταφορών και του περιβάλλοντος. Οι εφαρμογές του ML είναι απεριόριστες (Διάγραμμα 5.17), μερικές από αυτές αναλύονται παρακάτω.

Η επεξεργασία φυσικής γλώσσας (natural language processing/NLP) αποτελεί κομμάτι της μηχανικής μάθησης, των μεγάλων δεδομένων και της Τεχνητής Νοημοσύνης. Αφορά την ικανότητα ενός υπολογιστή να κατανοεί την φυσική γλώσσα (γραπτή, προφορική), να εξάγει νοήματα από ανθρώπινα ή γλωσσικά δεδομένα και να παράγει την φυσική γλώσσα. Με την βοήθεια NLP μεθόδων επεξεργαστές κειμένου όπως το Microsoft Word και το Grammarly εντοπίζουν τυχόν γραμματικά λάθη, πολλά τηλεφωνικά κέντρα παρέχουν πιο γρήγορη και αποτελεσματικότερη εξυπηρέτηση στους πελάτες με την χρήση του Interactive Voice Responce (IVR). Σαφώς και η άμεση μετάφραση από μια γλώσσα σε μια άλλη (google translate βασίζεται την διαδικασία επεξεργασίας φυσικής γλώσσας μέσω μηχανών. Επίσης, προσωπικοί ψηφιακοί βοηθοί όπως το Amazon Echo, η Cortana, η Siri χρησιμοποιούνται καθημερινά ολόένα και περισσότερο και είναι σε θέση να αντιλαμβάνονται τον ανθρώπινο λόγο και να δίνουν απαντήσεις δια μέσω ψηφιακής φωνής. Τέλος, μια επαναστατική εφαρμογή της NLP είναι η Ανάλυση συναισθήματος (Sentiment analysis) η οποία στοχεύει στην ανίχνευση πολικότητας (polarity detection), δηλαδή στην εξαγωγή άποψης από κείμενο και στην κατηγοριοποίηση αυτής ως θετική ή αρνητική. Η διαδικασία αυτή είναι μια νέα πρόκληση για τις εταιρίες που προσπαθούν να καταλάβουν, μέσα από εκατομμύρια διαδικτυακές αναρτήσεις, την γνώμη των ανθρώπων σε κοινωνικά, οικονομικά, αθλητικά, πολιτικά θέματα (5.17δ). Είναι φανερό λοιπόν η πρόοδος της NLP και οι νέες δυνατότητες αλληλεπίδρασης ανθρώπου-υπολογιστή.

Το πεδίο της **ρομποτικής** συνδέεται στενά με την Τεχνητή Νοημοσύνη αφού απαιτείται 'ευφυΐα' ώστε οι μηχανές να αντικαταστήσουν τον άνθρωπο σε εργασίες χειρισμού αντικειμένων και αυτόματης πλοήγησης. Οι αλγόριθμοι αυτοί βοηθούν ώστε το ρομπότ να χαρτογραφεί το περιβάλλον γύρω του και εκτελεί κινήσεις οι οποίες περιλαμβάνουν την αποφυγή εμποδίων. Οι σχετικές έρευνες στον τομέα αυτό αφορούν την κατασκευή μηχανών οι οποίες πέρα από τις βασικές αισθήσεις (αφή, όραση) είναι εφοδιασμένες με αντιληπτικές ικανότητες (αντίληψη σχημάτων, μορφών, εικόνων) με δυνατότητα εξαγωγής λογικών συμπερασμάτων. Ευρύτατη χρήση ρομπότ γίνεται σε πάρα πολλούς παραγωγικούς τομείς. Στην ιατρική, η ρομποτική αυξάνει εξαιρετικά την

ακρίβεια και περιορίζει τα ανθρώπινα λάθη. Ένα παράδειγμα συστήματος ρομποτικής υποβοήθησης είναι το ‘MAKO’ με την βοήθεια του οποίου μπορούν να πραγματοποιηθούν επεμβάσεις μερικής ή ολικής αρθροπλαστικής γόνατος. Μέχρι στιγμής έχουν πραγματοποιηθεί πάνω από 100.000 επεμβάσεις με μεγάλη επιτυχία. Επίσης, είναι αλήθεια πως στις διαστημικές αποστολές τα ρομπότ είναι πιο ανθεκτικά σε αντίξοες συνθήκες απ’ ότι ο άνθρωπος. Το εξειδικευμένο Robonaut – R2 που σχεδιάστηκε από την NASA και την General Motors είναι το πρώτο ανθρωποειδές ρομπότ που ταξίδεψε στο διάστημα τον Φεβρουάριο του 2011 (5.17α’).

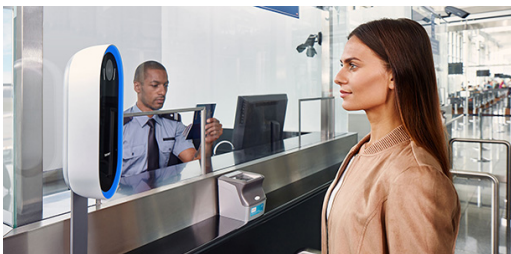
Η **μηχανική όραση** (computer vision) αναπτύχθηκε στις αρχές του 1980 και επιχειρεί να αναπαράγει αλγοριθμικά την αίσθηση-όραση σε έναν ηλεκτρονικό υπολογιστή ή ρομπότ. Οι αλγόριθμοι ανάλυσης εικόνας μπορούν να εντοπίσουν σε μια εικόνα αντικείμενα, ανθρώπους, την συναισθηματική τους κατάσταση αλλά και την περίπτωση που απεικονίζεται. Η μηχανική όραση εφαρμόζεται οπουδήποτε απαιτείται πιο εξειδικευμένη ανάλυση σε διάφορες εφαρμογές. Για παράδειγμα, το InnerEye της Microsoft είναι ικανό μέσω της μηχανικής όρασης να εντοπίζει πιθανούς όγκους ή άλλες παθήσεις σε ακτινογραφίες (5.17β’). Επίσης έχει εξελιχθεί μια από τις πιο βασικές τεχνολογίες στον βιομηχανικό αυτοματισμό (ανίχνευση βλάβων, διάβασμα barcode, ετικετών κα). Τέλος, η αξιοποίηση της υπολογιστικής όρασης έχει συνεισφέρει και στον τομέα της ασφάλειας. Συστήματα ελέγχου έχουν την δυνατότητα να ταυτοποιούν ανθρώπους με το ‘σκανάρισμα’ ίριδας (5.17γ’). Πιο συγκεκριμένα, κατά τη διάρκεια της σάρωσης δημιουργείται μια λεπτομερής εικόνα της επιφάνειας του χρωματιστού μέρους των ματιών, μέρος που είναι τόσο μοναδικό όσο ένα δακτυλικό αποτύπωμα.



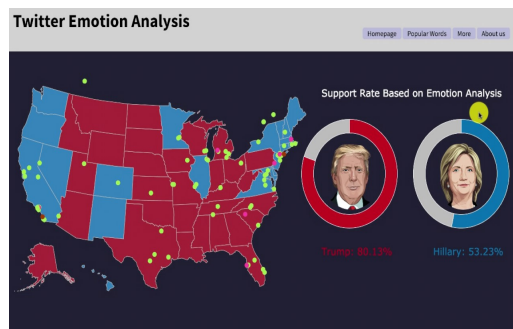
(α) Robonaut-R2, NASA.



(β) Εντοπισμός καρκίνου του προστάτη με το σύστημα InnerEye.



(γ) Σκανάρισμα ίριδας στο αεροδρόμιο του Dubai.



(δ) Support rate based on twitter Emotional Analysis.

Σχήμα 2.1: Εφαρμογές Μηχανικής Μάθησης.

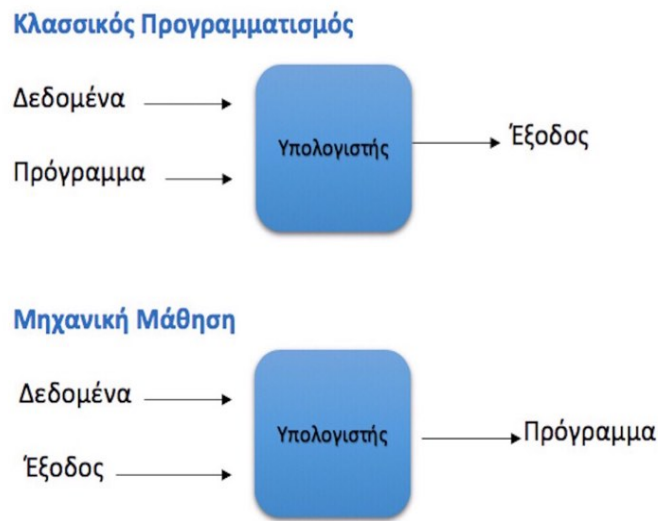
2.1.3 Βασικές έννοιες

Η Μηχανική Μάθηση/MM (Machine Learning/ML) είναι πεδίο της επιστήμης των υπολογιστών και της Τεχνητής Νοημοσύνης. Διερευνά την μελέτη αλγορίθμων που μπορούν να μαθαίνουν από τα δεδομένα και να κάνουν προβλέψεις σχετικά με αυτά. Σε αυτή την διαδικασία έχει συμβάλει η φτηνή υπολογιστική ισχύς και η διαθεσιμότητα μεγάλων δεδομένων. Η γενική ιδέα ενός προβλήματος Μηχανικής Μάθησης είναι η κατασκευή-βελτίωση μιας συνάρτησης, γνωστή ως συνάρτηση πρόγνωσης, ή αλλιώς υπόθεση, με βάση την οποία παράγονται αξιόπιστες αποφάσεις και αποτελέσματα δοθέντος ενός συνόλου παρατηρήσεων. Σε μια ιατρική διάγνωση, ο αλγόριθμος επξεργάζεται τα δεδομένα, δηλαδή τα συμπτώματα του ασθενή και καταλήγει σε μια πιθανή ασθένεια. Η MM συνδέεται στενά με την υπολογιστική στατιστική αφού έχουν κοινό στόχο την πρόβλεψη με την χρήση υπολογιστή. Επίσης, έχει ισχυρούς δεσμούς με την μαθηματική βελτιστοποίηση η οποία της παρέχει τις μεθόδους, την θεωρία και τους τομείς εφαρμογής.

Ένας από τους αρχικούς ορισμούς της Μηχανικής Μάθησης διατυπώθηκε από τον Arthur Samuel και είναι ο εξής: "Field of study that gives computers the ability to learn without being explicitly programmed". Μάλιστα, ο Arthur Samuel δημιούργησε το πρώτο τέτοιο πρόγραμμα (1959) το οποίο αφορούσε το παιχνίδι της ντάμας (game of checkers). Πρόκειται για ένα απλό παιχνίδι που απαιτούσε όμως στρατηγική την οποία το πρόγραμμα προσπαθούσε να μάθει. Ο υπολογιστής της IBM, που υποστήριξε το πρόγραμμα, 'παίζοντας' με τον δημιουργό του, αποθήκευε τις στρατηγικές που οδηγούσαν σε νίκες με αποτέλεσμα κάθε φορά να γίνεται αποδοτικότερο στο συγκεκριμένο παιχνίδι.

Ο Tom M. Mitchell, (computer scientist, C.M.U.) [33] διατύπωσε τον παρακάτω επίσημο ορισμό της Μηχανικής Μάθησης που χρησιμοποιείται ευρέως: «Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από εμπειρία E ως προς μια κλάση εργασιών T και ένα μέτρο επίδοσης P , αν η επίδοσή του σε εργασίες της κλάσης T , όπως αποτιμάται από το μέτρο απόδοσης P , βελτιώνεται με την εμπειρία E ». Αναφορικά με το παράδειγμα της ντάμας, E θεωρείται η εμπειρία που απέκτησε το σύστημα μέσω των παιχνιδιών που πραγματοποίησε έναντι του Arthur Samuel δηλαδή ο εντοπισμός ευνοϊκής στρατηγικής που θα οδηγήσει στη νίκη, επιτελούμενη εργασία T είναι η απομνημόνευση των 'σωστών' κινήσεων μέσω των παιχνιδιών και μέτρο απόδοσης P αφορά την πιθανότητα το σύστημα να κερδίσει την επόμενη παρτίδα.

Η έννοια της Μ.Μ. γίνεται καλύτερα αντιληπτή εφόσον σκεφτεί κανείς την διαφορά της με τον κλασικό προγραμματισμό. Ο τελευταίος αφορά την κατασκευή αλγορίθμων, μιας σειράς καλών ορισμένων βημάτων, ύστερα από προσεκτική σχεδίαση για την επίλυση κάποιου υπολογιστικού προβλήματος από υπολογιστή. Η διαδικασία αυτή μπορεί να γίνει εξαιρετικά πολύπλοκη και χρονοβόρα καθώς αν έστω και μία από τις παραμέτρους του θέματος μεταβληθεί, χρειάζεται ο προγραμματιστής να ενημερωθεί σχετικά με τις αλλαγές, να μελετήσει, να δημιουργήσει εκ νέου κανόνες, να κάνει δοκιμές προσαρμόζοντας έτσι το πρόβλημα στη νέα κατάσταση. Μέσω αυτής της διαδικασίας βασισμένη σε κανόνες (ruled based procedure) το πρόγραμμα δεν έχει την ικανότητα να γενικεύει τους κανόνες με αποτέλεσμα να μην είναι λειτουργικό σε περίπτωση εισαγωγής καινούργιων εισόδων.



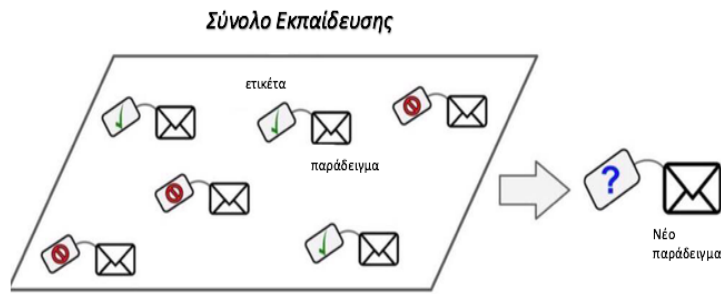
Διάγραμμα 2.1: Κλασσικός Προγραμματισμός και Μηχανική Μάθηση.

Σε αντίθεση, η Μηχανική Μάθηση ασχολείται με δομικά συστήματα που μπορούν να μάθουν από τα δεδομένα και όχι από ρητά προγραμματισμένες οδηγίες. Το βασικό συστατικό της ΜΜ είναι το σύνολο μιας πληθώρας παραδειγμάτων που ονομάζεται σύνολο εκπαίδευσης (training set) τα οποία τροφοδοτούνται στους αλγόριθμους ΜΜ βοηθώντας τους να γενικεύσουν κανόνες και να παράγουν τελικά την συνάρτηση πρόγνωσης. Έτσι, ενώ στον κλασσικό προγραμματισμό είναι απαραίτητη η δημιουργία κώδικα και η εισαγωγή δεδομένων σε αυτόν ώστε να παραχθεί το επιθυμητό αποτέλεσμα, στην Μηχανική Μάθηση με την βοήθεια πολλών εμπειρικών παραδειγμάτων δημιουργείται το τελικό πρόγραμμα (Διάγραμμα 2.1).

2.1.4 Διαρροή δεδομένων και Μεροληψία δειγμάτων

Η διαρροή δεδομένων (data leakage) είναι ένα από τα προβλήματα που δημιουργούνται συχνά στην διαδικασία εκμάθησης μηχανών δίνοντας μια παραπλανητική απόδοση του παραγόμενου μοντέλου. Δεν αφορά τον αλγόριθμο αλλά τα δεδομένα εισόδου. Εμφανίζεται όταν πληροφορίες που αφορούν τις ετικέτες, που πρέπει ο αλγόριθμος να προβλέψει, βρίσκονται στα δεδομένα εκπαίδευσης. Για παράδειγμα, σε ένα πρόβλημα αναγνώρισης προτύπων είναι αναγκαίο να αποφευχθούν οι τίτλοι των εικόνων ως χαρακτηριστικό εισόδου, αφού είναι μια πληροφορία που στην πραγματικότητα δεν δίνεται, και ο αλγοριθμος λανθασμένα θα μάθει από αυτή.

Η μεροληψία του δείγματος είναι επίσης ένα από τα προβλήματα που αντιμετωπίζονται στην είσοδο του αλγορίθμου. Συμβαίνει όταν η επιλογή των δεδομένων του συνόλου ελέγχου δεν αντιπροσωπεύουν την πραγματικότητα. Για παράδειγμα, έστω ότι για την δημιουργία ενός προβλεπτικού μοντέλου για τις εκλογές στις ΗΠΑ γίνονται έρευνες μέσω τηλεφώνου. Προφάνως, το δείγμα αυτό δεν είναι αντιπροσωπευτικό αφού η επιλογή των ανθρώπων που έχουν τηλεφωνικές γραμμές δεν είναι τυχαίο υποσύνολο του πληθυσμού, άρα είναι μεροληπτικό.



Διάγραμμα 2.2: Εκπαιδευόμενο σύνολο παραδειγμάτων στην επιβλεπομένη Μηχανική Μάθηση (πχ spam filtering) [19].

2.1.5 Τεχνικές και προβλήματα Μάθησης

Υπάρχει μια πληθώρα διαφορετικών αλγορίθμων που χρησιμοποιούνται στην Μηχανική Μάθηση. Οι τεχνικές με τις οποίες 'μαθαίνουν' διακρίνονται στις παρακάτω βασικές κατηγορίες.

Στην **μάθηση με επίβλεψη (supervised learning)**, το σύνολο εκπαίδευσης που δέχεται το υπολογιστικό πρόγραμμα αποτελείται από εισόδους $\{\mathbf{x}_i\}_{i=1}^N$ και αντίστοιχες εξόδους $\{y_i\}_{i=1}^N$, όπου N είναι ο αριθμός των παραδειγμάτων. Έτσι, με βάση αυτό το σύνολο παραδειγμάτων σωστής συμπεριφοράς που προέρχονται από έναν εξωτερικό επιβλέποντα (external supervisor) ο αλγόριθμος 'μαθαίνει' ένα γενικό κανόνα προκειμένου να αντιστοιχίσει νέες εισόδους σε αποτελέσματα. Η ανίχνευση ανεπιθύμητης αλληλογραφίας αποτελεί πρόβλημα επιβλεπόμενης Μηχανικής Μάθησης αφού κάθε είσοδος (μήνυμα) του συνόλου εκπαίδευσης έχει μια σωστή έξοδο ή αλλιώς ετικέτα (label) (Διάγραμμα 2.2).

Η μάθηση με επίβλεψη παρουσιάζει ένα σοβαρό μειονέκτημα. Απαιτεί μεγάλο αριθμό παραδειγμάτων γνωστής κλάσης προκειμένου να κατασκευάσει ένα ικανοποιητικό προβλεπτικό μοντέλο. Επέκταση αυτής της μορφής μάθησης αποτελεί η **μάθηση με μερική επίβλεψη (semi supervised learning)** η οποία χρησιμοποιεί ως δεδομένα εκπαίδευσης ένα μικρό σύνολο ταξινομημένων και ένα μεγάλο σύνολο μη ταξινομημένων εισόδων για την παραγωγή του τελικού μοντέλου. Πολλές φορές, ένας ικανοποιητικός αριθμός παρατηρήσεων για τις οποίες δεν υπάρχει ετικέτα σε συνδιασμό με λίγα δεδομένα για τα οποία υπάρχει η επιθυμητή έξοδος είναι αρκετά ώστε να δημιουργηθεί ένα μοντέλο με αυξημένη ακρίβεια σε σύγκριση με το αν χρησιμοποιούνταν μόνο οι παρατηρήσεις με ετικέτες. Υπάρχουν περιπτώσεις που η συλλογή δεδομένων είναι εύκολη, η απόδοση ετικετών όμως σε αυτά είναι περίπλοκη διαδικασία και αρκετά χρονοβόρα. Επειδή λοιπόν η απόδοση ετικετών σε όλο το σύνολο των παρατηρήσεων φέρει μεγάλο κόστος, η μάθηση με μερική επίβλεψη που είναι συνδιασμός της μάθησης με και χωρίς επίβλεψη βρίσκει σπουδαία πρακτική εφαρμογή.

Στην **μάθηση χωρίς επίβλεψη (unsupervised learning)** το υπολογιστικό πρόγραμμα δέχεται μόνο εισόδους $\{\mathbf{x}_i\}_{i=1}^N$. Χωρίς να παρέχεται κάποια εμπειρία

στον αλγόριθμο μάθησης, αυτός προσπαθεί να ανακαλύψει την δομή των δεδομένων εισόδου. Ανακαλύπτει μόνος του πρότυπα/ κατηγορίες στα δεδομένα χωρίς όμως να γνωρίζει εκ των προτέρων πόσες, ποιες και αν αυτές υπάρχουν.

Η **ενισχυτική μάθηση (reinforcement learning)** πρόκειται για μια τεχνική μάθησης που επιτυγχάνεται μέσω της αλληλεπίδρασης με στόχο την μεγιστοποίηση μιας συναρτησης του αριθμητικού σήματος ανταμοιβής. Πιο συγκεκριμένα, το σύστημα (ο ευφυής πράκτορας-agent) χωρίς να γνωρίζει ποια είναι η σωστή ενέργεια εκ των προτέρων, προσπαθεί να την ανακαλύψει μέσω της αλληλεπίδρασης με το περιβάλλον (environment) (μέθοδος δοκιμής και αποτυχίας). Με τον τρόπο αυτό 'μαθαίνει' μόνο του την καλύτερη στρατηγική που καλείται πολιτική (policy) η οποία θα οδηγήσει στη μεγαλύτερη δυνατή ανταμοιβή. Η πολιτική καθορίζει στον πράκτορα την επόμενη κίνηση δεδομένης μιας κατάστασης. Η τεχνική αυτή μάθησης εφαρμόζεται στον ρομποτικό έλεγχο, σε παιχνίδια, σε αυτοοδηγούμενα αυτοκίνητα και σε πολλές άλλες περιπτώσεις.

Η **μάθηση βασισμένη σε στιγμιότυπα (instance based learning)** έχει μια ιδιαιτερότητα σε σχέση με τις υπόλοιπες τεχνικές μάθησης. Στην περίπτωση αυτή, δεν παράγεται μια καθολική συνάρτηση πρόγνωσης αλλά κάθε φορά η κατηγοριοποίηση ενός νέου στιγμιότυπου γίνεται με βάση την ομοιότητά του με τα αρχικά δεδομένα που ο αλγόριθμος έχει αποθηκεύσει στην μνήμη του. Οι αλγόριθμοι που χρησιμοποιούν τον τρόπο μάθησης βασισμένο σε παραδείγματα ονομάζονται οκνηροί (lazy) αφού στην ουσία δεν υπάρχει κάποιο στάδιο εκπαίδευσης, αλλά 'μαθαίνουν' την στιγμή εκτέλεσης του συστήματος. Το κύριο μειονέκτημα τους είναι πως το υπολογιστικό κόστος κατά την ταξινόμηση νέων στιγμιότυπων μπορεί να είναι πολύ υψηλό. Στην συνέχεια, αναλύονται τα σημαντικά προβλήματα (learning tasks) που καλούνται να αντιμετωπίσουν οι αλγόριθμοι χρησιμοποιώντας τις προηγούμενες τεχνικές μάθησης.

Στο πρόβλημα της **ταξινόμησης (classification)** δημιουργούνται μοντέλα πρόβλεψης διακριτών τιμών. Στόχος είναι η ταξινόμηση των στιγμιότυπων στις κατηγορίες/ κλάσεις που ανήκουν. Πρόκειται για προβλήματα αναγνώρισης προτύπων (pattern recognition) και για την επίλυση τους χρησιμοποιούν τεχνικές μάθησης με και χωρίς επίβλεψη. Παραδείγματα αποτελούν η κατηγοριοποίηση των εισερχόμενων μηνυμάτων σε επιθυμητα/ανεπιθυμητα, η πρόβλεψη κάποια ασθένειας, δηλαδή αν κάποιος πάσχει ή όχι από αυτήν, κ.α.

Στο πρόβλημα της **παλινδρόμησης (regression)** δημιουργούνται μοντέλα πρόβλεψης αριθμητικών τιμών. Η έξοδος του συστήματος ανήκει σε ένα συνεχές σύνολο τιμών, όπως είναι το \mathbb{R} . Η πρόβλεψη εσόδων, τιμών κατοικιών/ μετοχών αποτελούν τέτοιου είδους προβλήματα.

Κατά την **συσταδοποίηση/ ομαδοποίηση δεδομένων (clustering)** στόχος του αλγορίθμου είναι ο βέλτιστος διαχωρισμός των δεδομένων σε ομάδες με βάση κάποιες ομοιότητες στα χαρακτηριστικά τους. Για τον επιτυχή διαχωρισμό συστάδων είναι απαραίτητη η ύπαρξη ενός μέτρου ομοιότητας. Σε αντίθεση με την ταξινόμηση, στην συσταδοποίηση δεν είναι γνωστό εκ των προτέρων το πόσες και ποιές ομάδες υπάρχουν.

Η **βαθμονόμηση (ranking)** πρόκειται για πρόβλημα ταξινόμησης που χρησιμοποιεί κάποιο κριτήριο. Παραδείγμα αυτής αποτελεί η αναζήτηση σελίδων στο διαδίκτυο που επιστρέφει αποτελέσματα με βάση τις αναζητήσεις των χρηστών.

Το πρόβλημα της συσταδοποίησης χρησιμοποιεί τεχνικές μάθησης χωρίς επίβλεψη ενώ τα υπόλοιπα λύνονται με την βοήθεια επιβλεπόμενης μάθησης η οποία είναι από τις καλύτερα μελετημένες περιοχές της Μηχανικής Μάθησης. Στην παρούσα διπλωματική εργασία θα αναλυθεί το πρόβλημα της ταξινόμησης.

2.2 Πρόβλημα ταξινόμησης

2.2.1 Εισαγωγή

Μια από τις πιο διαδεδομένες μεθόδους περιγραφής φαινομένων είναι η ταξινόμηση. Όπως αναφέρθηκε προηγουμένως, πρόκειται για πρόβλημα αναγνώρισης προτύπων (pattern recognition problem) όπου ο αλγόριθμος αξιοποιώντας τα παραδείγματα εκπαίδευσης, προσπαθεί να εντοπίσει ομοιότητες σε αυτά, παράγοντας μια συνάρτηση, γνωστή ως συνάρτηση πρόγνωσης. Στην προκειμένη περίπτωση η συνάρτηση αυτή ονομάζεται ταξινομητής και στοχεύει στην κατάταξη ενός νέου αντικειμένου/ προτύπου σε μια από τις γνωστές κλάσεις/ κατηγορίες, έχει δηλαδή την ικανότητα γενίκευσης. Ανάλογα με το πλήθος των κλάσεων γίνεται η διάκριση σε δυαδική (binary), πολλαπλών κλάσεων (multi-class) και πολλαπλών ετικετών (multi-label) ταξινόμηση.

Η δυαδική αποσκοπεί στον χωρισμό των δεδομένων σε δύο ομάδες και η τελική συνάρτηση προβλέπει σε ποια από τις δύο τα νέα δεδομένα ανήκουν. Η ταξινόμηση πολλαπλών ετικετών είναι μια γενίκευση αυτής των πολλαπλών κλάσεων. Και στις δύο περιπτώσεις υπάρχουν πάνω από δύο κλάσεις, με την διαφορά ότι στην πρώτη μια νέα είσοδος έχει το δικαίωμα να ανήκει σε μια μόνο από αυτές ενώ στην δεύτερη μπορεί να ανήκει σε περισσότερες. Για παράδειγμα, η ταξινόμηση ταινιών με βάση την καταλληλότητα παρακολούθησης είναι πρόβλημα πολλαπλών κλάσεων ενώ ανάλογα με το είδος του θέματος είναι πρόβλημα ταξινόμησης πολλαπλών ετικετών.

2.2.2 Ορισμός του προβλήματος

Αρχικά πρέπει να αναφέρουμε πως τα στιγμιότυπα, που σχηματίζουν τον χώρο \mathcal{X} , είναι οποιαδήποτε πιθανή είσοδος στον αλγόριθμο. Το υποσύνολο των στιγμιότυπων που επιλέγουμε για εκμάθηση και αξιολόγηση του συστήματος ονομάζεται δείγμα και συμβολίζεται με S . Με τον όρο χαρακτηριστικά (features) αναφερόμαστε σε ένα σύνολο γνωρισμάτων X^1, \dots, X^l που σχετίζονται με το δείγμα, περιγράφουν δηλαδή τα στιγμιότυπα του χώρου \mathcal{X} . Το σύνολο υποθέσεων (hypothesis set) είναι οι συναρτήσεις (h) που αντιστοιχούν το διάνυσμα των χαρακτηριστικών σε κάποια από τις διαθέσιμες κατηγορίες. Στο πρόβλημα κατηγοριοποίησης οι υποθέσεις αυτές ονομάζονται ταξινο-

μητές (classifiers).

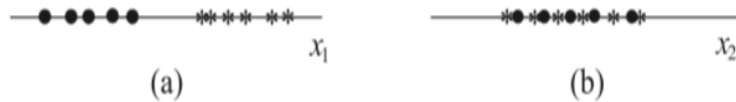
Σε ένα τυπικό πρόβλημα ταξινόμησης σκοπός είναι η κατάταξη ενός νέου στιγμιοτύπου της μορφής $\mathbf{x} = \{x^1, x^2, \dots, x^l\}$, όπου l το πλήθος των χαρακτηριστικών που επιλέγονται στην αρχή του προβλήματος, σε μια κατηγορία Y (εξαρτημένη μεταβλητή) που παίρνει διακριτές τιμές $\omega_1, \dots, \omega_m$. Αν $m = 2$ τότε είμαστε στην περίπτωση δυαδικής ταξινόμησης, διαφορετικά μιλάμε για ταξινόμηση πολλαπλών ετικετών ή πολλαπλών κλάσεων. Οι τιμές των ετικετών μπορεί να είναι ονομαστικές (nominal data) ή διατακτικές (ordinal data). Οι ονομαστικές τιμές αναφέρονται σε κατηγορίες, που η σειρά τους δεν έχει σημασία (πχ χρώμα, μέσο μεταφοράς κτλ) ενώ οι διατακτικές τιμές σε κατηγορίες όπου η διάταξη έχει σημασία (πχ σοβαρότητα ασθένειας, γνώμη κτλ). Η αντιστοίχιση των στιγμιοτύπων σε κάποια κατηγορία αναφέρεται σε μια υπόθεση $h : \mathcal{X} \rightarrow Y$ και σκοπός της είναι να προσεγγίσει την πραγματική συνάρτηση που αναπαριστά την κατάσταση στην φύση $f : \mathcal{X} \rightarrow Y$. Η κατασκευή της h γίνεται με βάση το ήδη ταξινομημένο σύνολο παραδειγμάτων $S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$ που ονομάζεται σύνολο εκπαίδευσης (training set) όπου $\mathbf{x}_i \in \mathcal{X}^l$ και $y_i \in Y$. Η συμπεριφορά της υπόθεσης εξαρτάται από το πόσο καλά έχει 'εκπαιδευτεί' ο αλγόριθμος ώστε να μπορεί να κάνει προβλέψεις με ακρίβεια. Αποτελεσματική εκπαίδευση γίνεται όταν ο αλγόριθμος καταφέρνει να προσεγγίσει ικανοποιητικά το σύνολο των παραμέτρων που προσπαθεί να 'μάθει' και οι οποίες συμβάλλουν στην δημιουργία της υπόθεσης h .

2.2.3 Περιγραφή διαδικασίας

Στην ενότητα αυτή περιγράφεται η διαδικασία που ακολουθείται σε ένα πρόβλημα ταξινόμησης και παριστάνεται γραφικά στο Διάγραμμα 2.4. Αρχικά το σύνολο των ήδη ταξινομημένων στιγμιοτύπων πρέπει να διαιρεθεί σε τρία υποσύνολα (εκπαίδευσης, επικύρωσης, ελέγχου). Τα υποσύνολα αυτά πρέπει να είναι ανεξάρτητα μεταξύ τους δηλαδή να μην συμμετέχουν με κανένα τρόπο σε πάνω από μια διαδικασία ταυτόχρονα. Τα δεδομένα εκπαίδευσης (training set), συνήθως αφορούν το 60% των δεδομένων και δημιουργούν την βασική δομή των μοντέλων ταξινόμησης. Τα δεδομένα επικύρωσης (validation set) συνήθως αντιπροσωπεύουν το 20% και ρυθμίζουν τις παραμέτρους κάθε μοντέλου, επιλέγοντας το βέλτιστο από αυτά. Τέλος, το σύνολο ελέγχου (test set) υπολογίζει την τιμή του σφάλματος της βελτιστοποιημένης μεθόδου που τελικά επιλέχθηκε.

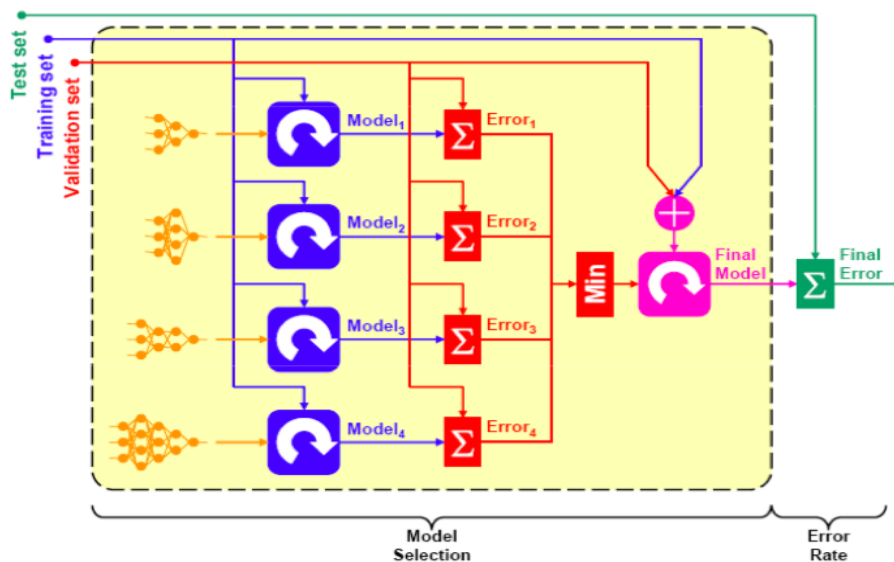
Η επιλογή των χαρακτηριστικών γίνεται με βάση την εκ των προτέρων μας γνώση για το πρόβλημα και χρησιμοποιούνται προκειμένου να εκπαιδευτεί ο αλγόριθμος. Η επιλογή ποιοτικών και αξιόπιστων χαρακτηριστικών (variable selection) από μια πληθώρα αυτών είναι ζωτικής σημασίας για την απόδοση του συστήματος. Γενικά, υιοθετούνται εκείνα που διαχωρίζουν 'καλά' τις κατηγορίες. Αυτό συνήθως σημαίνει ότι οι μέσες τιμές των χαρακτηριστικών διαφέρουν σημαντικά μεταξύ τους από κατηγορία σε κατηγορία και οι διασπορές γύρω από τις μέσες τιμές είναι μικρές. Είναι όμως απαραίτητη παραίτητη ανάπτυξη θεωριών προκειμένου να αντιμετωπιστούν συχνά προβλήματα όπως αυτό των μεγάλων διαστάσεων και της υπεπροσαρμοστικότητας. Στο Διάγραμμα 2.3 φαίνονται δύο χαρακτηριστικά X^1 , X^2 , πλούσιο και φτωχό αντίστοιχα σε πληροφορία. Με αστερίσκο συμβολίζονται οι τιμές που παίρνουν για την μια κλάση

ενώ με τελεία για την άλλη.



Διάγραμμα 2.3: Τιμές χαρακτηριστικών που (α) διαχωρίζουν και (β) δεν διαχωρίζουν “καλά” τις κλάσεις.

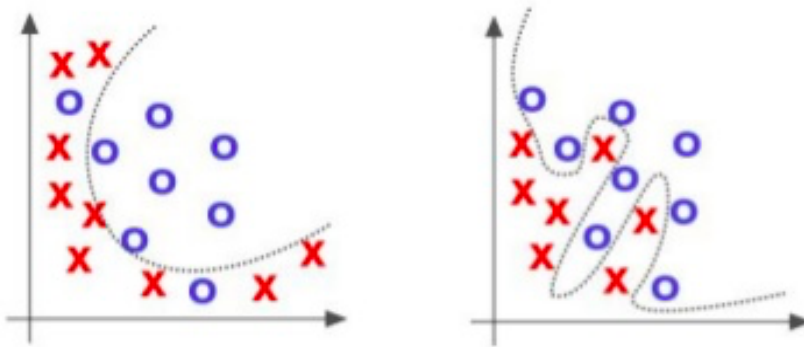
Αφού γίνει ο διαχωρισμός των δεδομένων στις τρεις κατηγορίες που αναφέραμε, επιλέγουμε αρχιτεκτονική (αλγόριθμο) και παραμέτρους για το πρόβλημα μας. Χρησιμοποιώντας τα δεδομένα εκπαίδευσης κατασκευάζεται η βασική δομή του αλγορίθμου (μια υπόθεση h). Το σύνολο επικύρωσης είναι κατάλληλο για να ρυθμίσουμε τις ελεύθερες παραμέτρους (που αρχικά επιλέξαμε αυθαίρετα) του αλγορίθμου και να μετρήσουμε την απόδοση αυτού. Η απόδοση σχετίζεται με την συνάρτηση απώλειας (loss function). Πρόκειται για μια συνάρτηση που μετρά το πόσο κοντά είναι η προβλεπόμενη τιμή στην πραγματική. Στα προβλήματα ταξινόμησης χρησιμοποιούμε πολλές φορές την 0/1 συνάρτηση απώλειας που θα αναλυθεί στο επόμενο κεφάλαιο. Στην συνέχεια, επαναλαμβάνουμε την παραπάνω διαδικασία για διαφορετικές τεχνικές και παραμέτρους. Τελικά, επιλέγουμε το μοντέλο που έχει την καλύτερη απόδοση στο σύνολο επικύρωσης. Το επιλεγμένο αυτό μοντέλο, το ξαναεκπαιδεύουμε, αυτή την φορά χρησιμοποιώντας το σύνολο εκπαίδευσης και επικύρωσης.



Διάγραμμα 2.4: Διαδικασία επιλογής μοντέλου στο πρόβλημα ταξινόμησης.

Σκοπός όμως είναι η πιθανή μελλοντική απόδοση του αλγορίθμου σε νέα παραδείγματα και όχι στα ήδη δεδομένα παραδείγματα εκπαίδευσης. Η τιμή της συνάρτησης απώλειας σε γνωστά δεδομένα δεν αποτελεί αξιόπιστη ένδειξη της τιμής της σε νέα δεδομένα. Αυτό συμβαίνει για τον λόγο ότι το μοντέλο έχει σχεδιαστεί με τρόπο ώστε να ελαχιστοποιεί την απώλεια που παράγεται από το σύνολο εκπαίδευσης

(σφάλμα επαναληπτικής αντικατάστασης/ resubstitution error) με αποτέλεσμα οποιαδήποτε εκτίμηση της αποδοτικότητας του να αποτελεί ανέλπιστα αισιόδοξη εκτίμηση. Αυτό αποτελεί ένα σύννηδες πρόβλημα στην Μηχανική Μάθηση, γνωστό ως υπερπροσαρμοστικότητα (overfitting) (Διάγραμμα 2.5) που έχει ως αποτέλεσμα η υπόθεση να ταιριάζει πολύ καλά στα δεδομένα εκπαίδευσης αλλά να μην έχει την ικανότητα να γενικεύσει το αποτέλεσμα σε νέα δεδομένα. Για τον λόγο αυτό, χρησιμοποιείται το σύνολο ελέγχου το οποίο δεν είναι διαθέσιμο ούτε στην εκπαίδευση ούτε στην επικύρωση. Όταν γίνεται το αντίθετο δηλαδή η τελική επιφάνεια απόφασης αδυνατεί να περιγράψει ικανοποιητικά τα δεδομένα εκπαίδευσης τότε εμφανίζεται το φαινόμενο της υποπροσαρμογής (underfitting).



Διάγραμμα 2.5: Στο πρώτο Διάγραμμα η απλή γραμμή απόφασης είναι η επιθυμητή παρά την λανθασμένη ταξινόμηση λίγων δειγμάτων. Στο δεύτερο Διάγραμμα, η περίπλοκη γραμμή απόφασης εφαρμόζει πολύ καλά στα δεδομένα αλλά δεν αποτελεί καλή εκτίμηση της πραγματικότητας (υπερπροσαρμογή).

Υπάρχουν δύο βασικές μέθοδοι που χρησιμοποιούνται για να λύσουν το πρόβλημα της κατηγοριοποίησης. Η πρώτη αφορά τον καθορισμό των ορίων. Δηλαδή, το αποτέλεσμα της διαδικασίας είναι μια συνάρτηση ή ένα σύνολο συναρτήσεων (ταξινομητής) με ελεύθερη μεταβλητή το διάνυσμα χαρακτηριστικών \mathbf{X} που διαμερίζει τον χώρο σε περιοχές $R_k, k \in (1, 2, \dots, m)$ όπου m ο αριθμός των κλάσεων. Πρόκειται για διαχωριστικά όρια, γνωστά ως όρια απόφασης (decision boundaries), τα οποία αποσκοπούν στην διαίρεση του χώρου των χαρακτηριστικών σε περιοχές που αντιστοιχούν σε διαφορετικές κλάσεις. Για παράδειγμα, αν έχουμε ένα πρόβλημα δύο διαστάσεων τότε οι περιοχές R_k θα είναι δυσδιάστατες, θα αντιστοιχούν στις κλάσεις ω_k και θα δημιουργούν διαχωριστικά όρια που θα είναι γραμμές. Σε ένα πρόβλημα μεγαλύτερων διαστάσεων, έστω l , τα R_k θα είναι επίσης l διαστάσεων, θα αντιστοιχούν στις κλάσεις ω_k δημιουργώντας διαχωριστικό υπόχωρο $l - 1$ διαστάσεων. Τα όρια απόφασης δεν είναι κατ' ανάγκη σωστά αφού μερικές φορές γίνεται εσφαλμένη ταξινόμηση. Η μέθοδος αυτή εφαρμόζεται στην Λογιστική Παλινδρόμηση, στις Μηχανές Διανυσμάτων Υποστήριξης, στα Νευρωνικά Δίκτυα, στον αλγόριθμο των k πλησιέστερων γειτόνων. Όπως θα δούμε αυτοί οι αλγόριθμοι υπολογίζουν κατευθείαν την εκ των υστέρων πιθανότητα $P(Y = \omega | \mathbf{X} = \mathbf{x})$ προκειμένου να δημιουργήσουν το σύνορο απόφασης, υπολογίζουν δηλαδή την πιθανότητα το προς ταξινόμηση παράδειγμα \mathbf{x} να ανήκει σε κάθε κλάση ω και επιλέγει την μεγαλύτερη πιθανότητα από αυτές που υπο-

λογίζει. Τα σύνορα απόφασης που δημιουργούνται, στην πράξη σπάνια είναι γραμμικά (υπερεπίπεδο στον l -διάστατο χώρο ή ευθεία γραμμή στο δυδιάστατο χώρο). Γενικά, πρόκειται για μη γραμμικές επιφάνειες που διαιρούν τον χώρο σε περιοχές κλάσεων.

Στην δεύτερη μέθοδο χρησιμοποιούνται κατανομές πιθανότητας. Έστω $\omega_i, i = 1, \dots, m$ οι κλάσεις του προβλήματος που μελετούμε και m το πλήθος αυτών. Τότε η ταξινόμηση ενός νέου στιγμιοτύπου, \mathbf{x} , γίνεται με βάση τον προσδιορισμό των πιθανοτήτων $P(\mathbf{X} = \mathbf{x} | Y = \omega)$ και $P(Y = \omega)$. Συνδιάζοντας τον κατά Μπέυζ κανόνα εκτιμούμε την εκ των υστέρων πιθανότητα $P(Y = \omega | \mathbf{X} = \mathbf{x})$ ως $P(Y = \omega)P(\mathbf{X} = \mathbf{x} | Y = \omega)$ για κάθε κλάση ω . Η κλάση όπου η ποσότητα αυτή μεγιστοποιείται είναι και η επιθυμητή. Η μέθοδος αυτή εφαρμόζεται σε αλγόριθμους όπως ο Αφελής Μπεύζιανός Ταξινομητής, στα Μπεύζιανά Δίκτυα, στην Γραμμική και Τετραγωνική Διαχωριστική Ανάλυση, με τους οποίους θα ασχοληθούμε στο Κεφάλαιο 4.

2.2.4 Πρόβλημα πολλών διαστάσεων

Προκειμένου να σχεδιαστεί ένας ταξινομητής με ικανοποιητική απόδοση είναι απαραίτητο το πλήθος των δειγμάτων να είναι αρκούντως μεγάλο. Πιο συγκεκριμένα, η ανάγκη για δεδομένα αυξάνει εκθετικά με την διάσταση l του χώρου των χαρακτηριστικών. Για παράδειγμα, αν σε ένα μονοδιάστατο πρόβλημα το N θεωρείται ένας καλός αριθμός σημείων εκπαίδευσης, τότε για τον l διάστατο χώρο θα απαιτούνταν N^l σημεία. Γενικά, για να περιγράψουμε ένα πρόβλημα χρησιμοποιούμε πολλά χαρακτηριστικά. Η ανάγκη για πολλά χαρακτηριστικά οδηγεί στην αδυναμία μας να κατασκευάσουμε ένα αποδοτικό μοντέλο αφού η συλλογή πολύ μεγάλου πλήθους δεδομένων είναι ανέφικτη. Το πρόβλημα αυτό είναι γνωστό ως πρόβλημα διαστατικότητας (curse of dimensionality). Ωστόσο, υπάρχουν αρκετές μέθοδοι μείωσης των χαρακτηριστικών. Κάποιες από αυτές αφορούν την επιλογή ενός υποσυνόλου των υπάρχοντων χαρακτηριστικών (επιλογή χαρακτηριστικών/ feature selection) και άλλες αφορούν την δημιουργία νέων χαρακτηριστικών τα οποία αποτελούν συνδιασμό αυτών που ήδη υπάρχουν (εξαγωγή χαρακτηριστικών/feature extraction).

Η Μηχανική Μάθηση ως υποπεριοχή του πεδίου της Τεχνητής Νοημοσύνης σχετίζεται με την εφαρμογή αλγορίθμων μάθησης με σκοπό τη αυτόνομη απόκτηση και ενσωμάτωση γνώσης. Δημιουργούνται έτσι υπολογιστικά συστήματα ικανά να μαθαίνουν μέσω της εμπειρίας και της ανάλυσης των δεδομένων που δέχονται. Τα υπολογιστικά αυτά συστήματα στην πραγματικότητα είναι συναρτήσεις (ταξινομητές στην περίπτωση του προβλήματος ταξινόμησης) που προκύπτουν από την εκπαίδευση των αλγορίθμων και μας βοηθούν στην πρόβλεψη αποτελεσμάτων που μας ενδιαφέρουν. Προκειμένου ένα σύστημα μάθησης να είναι αποτελεσματικό είναι ανάγκη να έχει την ικανότητα γενίκευσης, να μην υπάρχουν δηλαδή φαινόμενα υποπροσαρμογής ή υπερπροσαρμογής, τα οποία αντιμετωπίζονται με τον διαχωρισμό του αρχικού δείγματος σε σύνολο εκπαίδευσης, επικύρωσης και ελέγχου. Ωστόσο, για να αποφύγουμε συχνά προβλήματα όπως αυτό των μεγάλων διαστάσεων, της μεροληψίας του δείγματος, της διαρροής των δεδομένων, πρέπει να είμαστε ιδιαίτερα προσεχτικοί στην επιλογή των δεδομένων, καθώς αποτελούν βασικό συστατικό για την σχεδίαση και την υλοποίηση μεθόδων μάθησης υπό επίβλεψη.

Κεφάλαιο 3

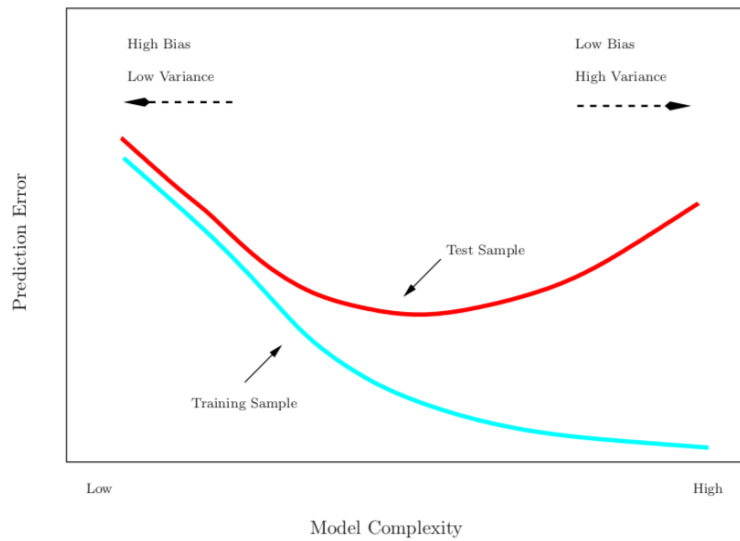
Αξιολόγηση απόδοσης ταξινομητών

3.1 Εισαγωγή

Η αξιολόγηση απόδοσης ενός ταξινομητή αποτελεί ένα από τα σημαντικότερα στάδια δημιουργίας ενός συστήματος μάθησης. Στην πράξη, με την βοήθεια του συνόλου επικύρωσης εκτιμούμε την απόδοση διαφορετικών μοντέλων (test/generalization errors) ώστε να επιλέξουμε το καλύτερο (επιλογή μοντέλου/model selection). Οι τρόποι που διαχειριζόμαστε τα σύνολα εκπαίδευσης και επικύρωσης ώστε να έχουμε μια καλύτερη προσέγγιση του σφάλματος αναλύονται στην Παράγραφο 3.3. Αφού έχουμε επιλέξει το τελικό μοντέλο με την καλύτερη επίδοση στο σύνολο επικύρωσης εκτιμούμε το σφάλμα του στα δεδομένα ελέγχου (εκτίμηση μοντέλου/ model assessment) με κάποια κριτήρια και γραφικές μεθόδους που αναλύονται στην Παράγραφο 3.4. Τέλος, εξηγείται το δίλημμα μεροληψίας-διασποράς, δύο ποσότητες που στην γενική των περιπτώσεων οφείλονται για την μη ικανοποιητική απόδοση ενός μοντέλου.

3.2 Αντιστάθμισμα μεροληψίας και διακύμανσης

Το αντιστάθμισμα μεροληψίας-διασποράς (bias-variance trade-off) αποτελεί ένα από τα σημαντικότερα προβλήματα στην Επιβλεπόμενη Μάθηση και αφορά την προσπάθεια συμβιβασμού των δύο αυτών ποσοτήτων. Με τον όρο μεροληψία εννοούμε την απόκλιση που υπάρχει μεταξύ της αντιστοίχισης που εκτιμούμε και της ιδανικής [21][7][20]. Υψηλή μεροληψία συνδέεται με το πρόβλημα υποπροσαρμογής (underfitting) και εμφανίζεται όταν προσαρμόζουμε ένα μοντέλο περισσότερο απλό από όσο πρέπει με αποτέλεσμα τα σφάλματα εκπαίδευσης και ελέγχου να είναι μεγάλα. Από την άλλη, η



Διάγραμμα 3.1: Δίλημμα μεροληψίας-διασποράς.

διακύμανση δείχνει την ευαισθησία στο σύνολο των δεδομένων εκπαίδευσης. Υψηλή διακύμανση δημιουργεί το πρόβλημα της υπερπροσαρμογής (overfitting). Πιο συγκεκριμένα, υιοθετούνται πολύπλοκα μοντέλα (με πολλές παραμέτρους) που προσεγγίζουν πολύ καλά ένα συγκεκριμένο σύνολο εκπαίδευσης αλλά απογοητευτικά κάποιο άλλο. Σε τέτοια μοντέλα παρατηρείται χαμηλή μεροληψία. Γίνεται αντιληπτό λοιπόν πως μειώνοντας την διασπορά αυξάνεται η μεροληψία και αντίστροφα. Το φαινόμενο αυτό είναι γνωστό ως δίλημμα μεροληψίας-διασποράς και για την αντιμετώπιση του αναζητούνται τρόποι αντιστάθμισης αυτών.

Η ελαχιστοποίηση του σφάλματος εκπαίδευσης δεν σημαίνει ελαχιστοποίηση του σφάλματος ελέγχου, όπως φαίνεται και στο Διάγραμμα 3.1. Καθώς η πολυπλοκότητα του μοντέλου αυξάνει (ξεκινώντας από τα αριστερά του οριζόντιου άξονα προς τα δεξιά), το σφάλμα ελέγχου και εκπαίδευσης μειώνεται μέχρι κάποιο σημείο (βέλτιστο σφάλμα ελέγχου). Μετά το σημείο αυτό το μοντέλο αρχίζει να 'υπερπροσαρμόζεται' στα δεδομένα χάνοντας την ικανότητα γενίκευσης με αποτέλεσμα να μεγαλώνει το σφάλμα ελέγχου.

3.2.1 Αποσύνθεση αναμενόμενης τιμής σφάλματος στην παλινδρόμηση

Στα προβλήματα παλινδρόμησης, όπου η συνάρτηση απώλειας είναι η τετραγωνική, είναι ευκολότερο να αποσυνθέσουμε το μέσο τετραγωνικό σφάλμα στους όρους θόρυβος-μεροληψία-διασπορά από ότι σε ένα πρόβλημα ταξινόμησης όπου η κατάλληλη συνάρτηση είναι η 0/1. Για λόγους κατανόησης θα γίνει αρχικά η προσέγγιση σε προβλήματα παλινδρόμησης όπου η τετραγωνική συνάρτηση σφάλματος ορίζεται στο $I \times I$ ως $L(y, y') = (y' - y)^2$, όπου $I \subseteq \mathbb{R}$.

Θεωρούμε την τυχαία συνεχή μεταβλητή Y και ένα τυχαίο διάνυσμα \mathbf{X} που περιέχει l τυχαίες μεταβλητές X_1, \dots, X_l . Έστω τώρα ένα σύνολο εκπαίδευσης $S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$. Κάθε ζεύγος (\mathbf{x}_i, y_i) παίρνει τιμές από τις (\mathbf{X}, Y) και εκφράζει την κλάση y_i που παίρνει ένα παράδειγμα \mathbf{x}_i . Το N δείχνει το σύνολο των διαθέσιμων παραδειγμάτων του συνόλου S . Επίσης, μπορούμε να ορίσουμε την από κοινού αθροιστική συνάρτηση κατανομής (joint cumulative density function) της συνεχούς (\mathbf{X}, Y) ως:

$$F_{\mathbf{X},Y}(\mathbf{x}, y) = P(\mathbf{X} \leq \mathbf{x}, Y \leq y) = \int_{\mathbf{x}} \int_Y f_{\mathbf{X},Y}(\mathbf{x}, y) \partial \mathbf{x} \partial y,$$

όπου $f_{\mathbf{X},Y}$ η από κοινού συνάρτηση πυκνότητας πιθανότητας/ σ.π.π. (joint probability density function) της (\mathbf{X}, Y) . Ακόμη, υποθέτουμε ότι κάθε δεδομένο διάνυσμα \mathbf{x} δεν αντιστοιχεί σε ένα μοναδικό y . Για παράδειγμα, έστω \mathbf{x} ένα διάνυσμα που περιγράφει τα χαρακτηριστικά ενός σπιτιού με μεταβλητή απόκρισης y που αναφέρεται στην τιμή του. Τότε προφανώς ένα διαφορετικό σπίτι με τα ίδια χαρακτηριστικά δεν θα έχει αναγκαστικά την ίδια τιμή. Θεωρούμε λοιπόν μια αντιστοίχιση της μορφής $Y = f(\mathbf{X}) + \epsilon$, όπου ϵ είναι ο θόρυβος και ακολουθεί $N(0, \sigma^2)$. Έχουμε συνεπώς το δικαίωμα να πούμε ότι για κάθε δεδομένο \mathbf{x} υπάρχει μια κατανομή ως προς Y . Η τυχαία μεταβλητή Y δεδομένου ότι η \mathbf{X} έχει πάρει την τιμή \mathbf{x} ορίζεται να έχει πυκνότητα $f_{Y|\mathbf{X}}(y|\mathbf{x})$. Η μέση τιμή της Y δεδομένου ότι $\mathbf{X} = \mathbf{x}$ ισούται με [29][56][42]:

$$\bar{y}(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}] = \int_Y y f_{Y|\mathbf{X}}(y|\mathbf{x}) \partial y.$$

Στην συνέχεια συμβολίζουμε $h_S = \mathcal{A}(S)$ ως την υπόθεση που ‘έμαθε’ ο αλγόριθμος (\mathcal{A}), που χρησιμοποιήσαμε, μέσω του συνόλου εκπαίδευσης S . Δοθείσης της h_S το σφάλμα ελέγχου γράφεται ως:

$$\mathbb{E}_{\mathbf{X},Y} [(h_S(\mathbf{x}) - y)^2] = \int_{\mathbf{X}} \int_Y (h_S(\mathbf{x}) - y)^2 f_{\mathbf{X},Y}(\mathbf{x}, y) \partial y \partial \mathbf{x},$$

όπου $h_S(\mathbf{x})$ είναι η πρόβλεψη του αλγορίθμου και y η πραγματική τιμή για το διάνυσμα \mathbf{x} . Επίσης, παρατηρούμε ότι το S δημιουργήθηκε τυχαία από την κατανομή $F_{\mathbf{X},Y}$ γι’ αυτό και θεωρείται τυχαίο. Κατά συνέπεια η h_S ως συνάρτηση του S είναι μια τυχαία μεταβλητή και μπορούμε να υπολογίσουμε την αναμενόμενη τιμή της (δοθέντος ενός αλγορίθμου \mathcal{A}):

$$\bar{h} = \mathbb{E}_S (h_S) = \int_S h_S f(S) \partial S.$$

Πλέον είναι εύκολο να υπολογίσουμε το αναμενόμενο σφάλμα του αλγορίθμου \mathcal{A} γνωρίζοντας ότι τα δεδομένα ακολουθούν την από κοινού συνάρτηση κατανομής $F_{\mathbf{X},Y}$:

$$\mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - y)^2] = \int_S \int_{\mathbf{X}} \int_Y (h_S(\mathbf{x}) - y)^2 f(\mathbf{x}, y) f(S) \partial \mathbf{x} \partial y \partial S. \quad (3.1)$$

Πρέπει να σημειώσουμε ότι τα (\mathbf{x}, y) αναφέρονται στα δεδομένα επικύρωσης ενώ το S στα δεδομένα εκπαίδευσης. Στην πραγματικότητα ενδιαφερόμαστε για τον υπολογισμό της σχέσης (3.1) καθώς αυτή αξιολογεί την γενική απόδοση του αλγορίθμου \mathcal{A} . Στην

συνέχεια δείχνουμε πως η έκφραση αυτή διακρίνεται στους όρους θόρυβος-μεροληψία-διασπορά:

$$\begin{aligned} \mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - y)^2] &= \mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x}) + \bar{h}(\mathbf{x}) - y)^2] \\ &= \mathbb{E}_{\mathbf{X},S} [(\bar{h}_S(\mathbf{x}) - \bar{h}(\mathbf{x}))^2] + 2 \mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x})) (\bar{h}(\mathbf{x}) - y)] + \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - y)^2]. \end{aligned} \quad (3.2)$$

Ο δεύτερος όρος της εξίσωσης (3.2) είναι 0 αφού:

$$\begin{aligned} \mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x})) (\bar{h}(\mathbf{x}) - y)] &= \mathbb{E}_{\mathbf{X},Y} [\mathbb{E}_S [h_S(\mathbf{x}) - \bar{h}(\mathbf{x})] (\bar{h}(\mathbf{x}) - y)] \\ &= \mathbb{E}_{\mathbf{X},Y} [(\mathbb{E}_S [h_S(\mathbf{x})] - \bar{h}(\mathbf{x})) (\bar{h}(\mathbf{x}) - y)] \\ &= \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - \bar{h}(\mathbf{x})) (\bar{h}(\mathbf{x}) - y)] \\ &= \mathbb{E}_{\mathbf{X},Y} [0] \\ &= 0. \end{aligned}$$

Η εξίσωση (3.2) έχει πλέον την μορφή:

$$\mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - y)^2] = \underbrace{\mathbb{E}_{\mathbf{X},S} [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x}))^2]}_{\text{Διασπορά}} + \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - y)^2]. \quad (3.3)$$

Ο δεύτερος όρος της σχέσης (3.3) αναλύεται ως:

$$\begin{aligned} \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - y)^2] &= \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}) + \bar{y}(\mathbf{x}) - y)^2] \\ &= \underbrace{\mathbb{E}_{\mathbf{X},Y} [(\bar{y}(\mathbf{x}) - y)^2]}_{\text{Θόρυβος}} + \underbrace{\mathbb{E}_{\mathbf{X}} [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))^2]}_{\text{Μεροληψία}^2} + 2 \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})) (\bar{y}(\mathbf{x}) - y)]. \end{aligned} \quad (3.4)$$

Ο τρίτος όρος της εξίσωσης (3.4) είναι 0 αφού:

$$\begin{aligned} \mathbb{E}_{\mathbf{X},Y} [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})) (\bar{y}(\mathbf{x}) - y)] &= \mathbb{E}_{\mathbf{X}} [\mathbb{E}_{Y|\mathbf{X}} [\bar{y}(\mathbf{x}) - y] (\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))] \\ &= \mathbb{E}_{\mathbf{X}} [\mathbb{E}_{Y|\mathbf{X}} [\bar{y}(\mathbf{x}) - y] (\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))] \\ &= \mathbb{E}_{\mathbf{X}} [(\bar{y}(\mathbf{x}) - \mathbb{E}_{Y|\mathbf{X}} [y]) (\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))] \\ &= \mathbb{E}_{\mathbf{X}} [(\bar{y}(\mathbf{x}) - \bar{y}(\mathbf{x})) (\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))] \\ &= \mathbb{E}_{\mathbf{X}} [0] \\ &= 0. \end{aligned}$$

Επομένως, το αναμενόμενο σφάλμα του αλγόριθμου υπολογισμένο στα δεδομένα επικύρωσης γράφεται ως:

$$\underbrace{\mathbb{E}_{\mathbf{X},Y,S} [(h_S(\mathbf{x}) - y)^2]}_{\text{Αναμενόμενο σφάλμα ελέγχου}} = \underbrace{\mathbb{E}_{\mathbf{X},S} [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x}))^2]}_{\text{Διασπορά}} + \underbrace{\mathbb{E}_{\mathbf{X},Y} [(\bar{y}(\mathbf{x}) - y)^2]}_{\text{Θόρυβος}} + \underbrace{\mathbb{E}_{\mathbf{X}} [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))^2]}_{\text{Μεροληψία}^2}.$$

Δηλαδή για κάθε \mathbf{x} το μέσο τετραγωνικό σφάλμα του αλγόριθμου \mathcal{A} είναι:

$$MSE(\mathbf{x}) = \mathbb{E}_S [(h_S(\mathbf{x}) - \bar{h}(\mathbf{x}))^2] + \mathbb{E}_Y [(\bar{y}(\mathbf{x}) - y)^2] + [(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x}))^2].$$

3.2.2 Έννοια σφάλματος στην ταξινόμηση

Στο πρόβλημα της ταξινόμησης η ποσότητα $\bar{y}(\mathbf{x})$ και η τετραγωνική συνάρτηση απώλειας που συναντήσαμε στην παλινδρόμηση δεν έχουν νόημα. Για τον λόγο αυτό χρησιμοποιούμε την 0/1 συνάρτηση απώλειας (0/1 loss function) που ορίζεται στο $\{-1, +1\} \times \{-1, +1\}$ ως $L(y, y') = \mathbb{1}[y' \neq y]$. Δηλαδή, σε κάθε περίπτωση η μέγιστη απώλεια που θα έχουμε όταν προβλεψουμε μια λάθος κατηγορία είναι 1. Όμως τα λάθη ταξινόμησης κλάσεων δεν έχουν πάντα το ίδιο βάρος, δηλαδή ίδιες επιπτώσεις. Μερικές φορές όμως κάποιες εσφαλμένες αποφάσεις είναι περισσότερο δαπανηρές από κάποιες άλλες. Για τον λόγο αυτό είναι πιο σωστό να ορίσουμε ένα πίνακα απώλειας L όπου L_{ij} θα αντανακλά την απώλεια που λαμβάνουμε αν για κάποιο \mathbf{x} προβλέψουμε την κλάση ω_j ενώ η σωστή είναι η ω_i [29].

Αναμενόμενη απώλεια εσφαλμένης ταξινόμησης y'

Όπως είδαμε, στην περίπτωση της ταξινόμησης η μεταβλητή απόκρισης Y είναι διακριτή. Έστω ότι γνωρίζουμε την συνάρτηση μάζας πιθανότητας (σ.μ.π.) (probability mass function) της διακριτής τυχαίας μεταβλητής Y όταν το τυχαίο διάνυσμα \mathbf{X} παίρνει τιμή \mathbf{x} . Συνεπώς είναι γνωστή η δεσμευμένη σ.μ.π. $P(Y = y | \mathbf{X} = \mathbf{x})$. Μπορούμε πλέον να ορίσουμε το ρίσκο που έχουμε όταν κάνουμε μια εσφαλμένη ταξινόμηση y' ως την αναμενόμενη απώλεια που προκύπτει αν για ένα τυχαία επιλεγμένο παράδειγμα \mathbf{x} προβλέψουμε την κλάση y' [29]:

$$r(y') = \mathbb{E}_Y [L(Y, y')] = \sum_{y=\omega_1}^{\omega_m} P(Y = y | \mathbf{X} = \mathbf{x}) L(y, y'). \quad (3.5)$$

Είναι φανερό πως η βέλτιστη κλάση y^* για ένα διάνυσμα \mathbf{x} είναι αυτή που ελαχιστοποιεί το ρίσκο της εσφαλμένης ταξινόμησης. Δηλαδή:

$$y^* = \underset{y'}{\operatorname{argmin}} r(y'). \quad (3.6)$$

Για παράδειγμα, σε ένα πρόβλημα δύο κλάσεων αν $L(\omega_1, \omega_2) = 0.1$, $L(\omega_2, \omega_1) = 0.9$, $P(\omega_1) = 0.7$ και $P(\omega_2) = 0.3$ τότε $r(\omega_1) = 0.3 \times 0.9 = 0.27$ και $r(\omega_2) = 0.7 \times 0.1 = 0.07$. Συνέπως, η κλάση που επιλέγουμε είναι η ω_2 . Χρησιμοποιώντας την 0/1 συνάρτηση απώλειας οι σχέσεις (3.5), (3.6) γίνονται:

$$r(y') = \sum_{y' \neq \omega_i} P(Y = \omega_i | \mathbf{X} = \mathbf{x}) = 1 - P(Y = y' | \mathbf{X} = \mathbf{x}),$$

και:

$$y^* = \underset{y'}{\operatorname{argmin}} r(y') = \underset{y'}{\operatorname{argmax}} P(Y = y' | \mathbf{X} = \mathbf{x}).$$

Η συνάρτηση y^* ονομάζεται βέλτιστος Μπεϋζιανός ταξινομητής αφού ελαχιστοποιεί το ρίσκο εσφαλμένης ταξινόμησης δοθείσης της δεσμευμένης σ.μ.π. $P(Y = y | \mathbf{X} = \mathbf{x})$ δηλαδή μεγιστοποιεί την πιθανότητα να γίνει σωστά η ταξινόμηση. Προφανώς, ο υπολογισμός των παραπάνω σχέσεων είναι ανέφικτος αφού στην πραγματικότητα η $P(Y = y | \mathbf{X} = \mathbf{x})$ είναι άγνωστη. Είναι δυνατόν να προσεγγιστεί όμως χρησιμοποιώντας τα ήδη ταξινομημένα παραδείγματα που έχουμε στην διάθεσή μας.

Εκτιμώμενη απώλεια εσφαλμένης ταξινόμησης y'

Σύμφωνα με το Νόμο των Μεγάλων Αριθμών κάτω από κατάλληλες υποθέσεις ο δειγματικός μέσος μιας ακολουθίας ανεξάρτητων τυχαίων μεταβλητών που ακολουθούν μια κοινή κατανομή συγκλίνει σχεδόν βεβαίως προς τον θεωρητικό μέσο (μέση τιμή) της κατανομής. Με βάση τον μαθηματικό ορισμό αυτό σημαίνει ότι αν $\{Z_n, n = 1, 2, \dots\}$ είναι μια ακολουθία ανεξάρτητων και ισόνομων τυχαίων μεταβλητών με μέση τιμή \mathbb{E}_Z τότε:

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n Z_i = \mathbb{E}_Z,$$

με πιθανότητα 1. Μπορούμε λοιπόν να υπολογίσουμε το σφάλμα εκπαίδευσης ως τη μέση απώλεια στα δεδομένα εκπαίδευσης:

$$err_{train}(y') = \frac{1}{S_{train}} \sum_{i=1}^{S_{train}} L(y'_i, y_i), \quad (3.7)$$

όπου S_{train} το πλήθος παραδειγμάτων εκπαίδευσης. Κατά την διαδικασία αξιολόγησης ενός μοντέλου δεν επιδιώκουμε την ελαχιστοποίηση της σχέσης (3.7) καθώς δεν αποτελεί ρεαλιστική εκτίμηση της πραγματικής απόδοσης και συνήθως το σφάλμα είναι ανέλπιστα μικρό. Σε αντίθεση υπολογίζουμε το σφάλμα του ταξινομητή στο σύνολο επικύρωσης που είναι αμερόληπτο:

$$err_{test}(y') = \frac{1}{S_{test}} \sum_{i=1}^{S_{test}} L(y'_i, y_i), \quad (3.8)$$

όπου S_{test} το πλήθος των παραδειγμάτων για επικύρωση.

Αναμενόμενη τιμή σφάλματος ταξινόμησης για ένα δεδομένο x

Αφού το σετ εκπαίδευσης S θεωρείται τυχαίο, υπολογίζουμε την αναμενόμενη απώλεια εσφαλμένης ταξινόμησης κάποιου διανύσματος x ως εξής [29]:

$$\begin{aligned} MME(\mathbf{x}) &= \mathbb{E}_{Y,S} [L(Y, \hat{Y})] \\ &= \mathbb{E}_{Y,S} \left[\sum_{y,y'=\omega_i}^{\omega_m} L(y, y') \mathbb{1}(Y = y) \mathbb{1}(\hat{Y} = y') \right] \\ &= \sum_{y,y'=\omega_i}^{\omega_m} L(y, y') \mathbb{E}_Y [\mathbb{1}(Y = y)] \mathbb{E}_S [\mathbb{1}(\hat{Y} = y')] \\ &= \sum_{y,y'=\omega_i}^{\omega_m} L(y, y') P(Y = y | \mathbf{X} = \mathbf{x}) P(\hat{Y} = y' | \mathbf{X} = \mathbf{x}). \end{aligned}$$

Κάνοντας χρήση της 0/1 συνάρτησης απώλειας έχουμε:

$$\begin{aligned} MME(\mathbf{x}) &= 1 - \sum_{y, y'=\omega_i}^{\omega_m} \mathbb{1}(y = y')P(Y = y|\mathbf{X} = \mathbf{x})P(\hat{Y} = y'|\mathbf{X} = \mathbf{x}) \\ &= 1 - \sum_{y=\omega_1}^{\omega_m} P(Y = y|\mathbf{X} = \mathbf{x})P(\hat{Y} = y|\mathbf{X} = \mathbf{x}) \\ &= P(Y \neq \hat{Y}|\mathbf{X} = \mathbf{x}). \end{aligned}$$

3.2.3 Αποσύνθεση αναμενόμενης τιμής σφάλματος στην ταξινόμηση

Μια λύση στο πρόβλημα αποσύνθεσης του σφάλματος στην ταξινόμηση δόθηκε από τον P.Domingos [15]. Όπως είδαμε η βέλτιστη κλάση για ένα δεδομένο \mathbf{x} ελαχιστοποιεί την αναμενόμενη τιμή της συνάρτησης απώλειας για κάθε πιθανή τιμή y της τυχαίας μεταβλητής Y :

$$y^* = \operatorname{argmin}_{y'} \mathbb{E}_Y[L(Y, y')].$$

Επίσης, θα ορίσουμε ως κύρια κλάση (main class), \hat{y}^* για ένα \mathbf{x} την κλάση που ελαχιστοποιεί την αναμενόμενη τιμή της συνάρτησης απώλειας για όλα τα πιθανά σύνολα εκπαίδευσης S :

$$\hat{y}^* = \operatorname{argmin}_{y'} \mathbb{E}_S[L(\hat{Y}, y')].$$

Παρατηρούμε ότι το \hat{y}^* στην ταξινόμηση αντιστοιχεί στο $\bar{h}(\mathbf{x})$ στην παλινδρόμηση. Μπορούμε πλέον να ορίσουμε τις έννοιες μεροληψία, θόρυβος και διασπορά για ένα ταξινομητή. Ο θόρυβος για ένα δεδομένο παράδειγμα \mathbf{x} είναι η απώλεια που προέρχεται από τις αποκλίσεις της κλάσης y από την βέλτιστη y^* :

$$N(\mathbf{x}) = \mathbb{E}_Y[L(Y, y^*)]. \quad (3.9)$$

Δοθέντος \mathbf{x} , η μεροληψία αναφέρεται στην απώλεια που έχει το μοντέλου εξαιτίας της διαφοράς ανάμεσα στην κύρια πρόβλεψη \hat{y}^* και την βέλτιστη y^* :

$$B(\mathbf{x}) = L(y^*, \hat{y}^*).$$

Τέλος, με τον όρο διασπορά για μια συγκεκριμένη είσοδο εννοούμε την μέση απώλεια που οφείλεται στην πρόβλεψη \hat{Y} και στην κύρια \hat{y}^* :

$$V(\mathbf{x}) = \mathbb{E}_S = [L(\hat{y}^*, \hat{Y})].$$

Κάνοντας χρήση των παραπάνω ορισμών, μπορούμε να αναλύσουμε την αναμενόμενη απώλεια της εσφαλμένης ταξινόμησης σε όρους θόρυβος-μεροληψία-διασπορά ως:

$$ME(\mathbf{x}) = \mathbb{E}_{Y,S}[L(Y, \hat{Y})] = c_1 N(\mathbf{x}) + B(\mathbf{x}) + c_2 V(\mathbf{x}),$$

όπου c_1, c_2 πολλαπλασιαστικοί όροι που εξαρτώνται από το \mathbf{x} . Αφού το \mathbf{x} είναι ένα τυχαίο διάνυσμα μπορούμε να βρούμε την αναμενόμενη τιμή του σφάλματος ως προς \mathbf{x} :

$$MIE = \mathbb{E}_{\mathbf{x}, Y, S}[L(Y, \hat{Y})] = \mathbb{E}_{\mathbf{x}}[c_1 N(\mathbf{x})] + \mathbb{E}_{\mathbf{x}}[B(\mathbf{x})] + \mathbb{E}[c_2 V(\mathbf{x})].$$

3.3 Μέθοδοι βελτίωσης του μοντέλου για υψηλότερη απόδοση

Ο διαχωρισμός των διαθέσιμων δεδομένων σε σύνολο εκπαίδευσης, επικύρωσης και ελέγχου είναι καθοριστικό στάδιο για την δημιουργία ενός αποδοτικού αλγορίθμου. Ωστόσο είναι δύσκολο να δοθεί ένας γενικός κανόνας για το μέγεθος αυτών. Υπάρχουν όμως κάποιες τεχνικές που επαναχρησιμοποιούν το δείγμα και κάνουν την διαδικασία μέτρησης της απόδοσης του εκπαιδευμένου αλγορίθμου στο σύνολο επικύρωσης περισσότερο αξιόπιστες.

Στην μέθοδο hold-out τα ταξινομημένα στιγμιότυπα χωρίζονται σε δύο σύνολα, εκπαίδευσης (2/3) και επικύρωσης (1/3) όπως φαίνεται στο Διάγραμμα 3.2. Η μέθοδος αυτή αν και είναι υπολογιστικά φτηνή, σε περίπτωση μικρού συνόλου δεδομένων δεν είναι δυνατό να κρατήσουμε δείγματα για δοκιμή. Το σφάλμα υπολογίζεται σύμφωνα με την σχέση (3.8). Επίσης, η διαμέριση σε δύο ομάδες μπορεί να παραπλανήσει το σφάλμα. Τα προβλήματα αυτά μπορούν να ξεπεραστούν, χρησιμοποιώντας τις παρακάτω μεθόδους με μεγαλύτερο όμως υπολογιστικό κόστος [20][57].

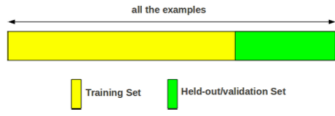
Η μέθοδος τυχαία υποδειγματοληψία (random subsampling) εκτελεί K επαναλήψεις και για κάθε μια από αυτές επιλέγουμε ένα συγκεκριμένο αριθμό παραδειγμάτων (χωρίς αντικατάσταση) για έλεγχο, εκπαιδούμε τον αλγόριθμο στα υπόλοιπα παραδείγματα όπως φαίνεται στο Διάγραμμα 3.3. Τέλος, γίνεται η εκτίμηση του σφάλματος, E_i σε κάθε i επανάληψη, με την βοήθεια των δεδομένων επικύρωσης και η τελική απόδοση είναι το συνολικό ποσοστό λάθους (E) που υπολογίζεται ως ο μέσος όρος των E_i .

Μια από τις πιο διαδεδομένες τεχνικές αποτελεί η διασταυρωμένη επικύρωση σε K μέρη (K fold cross-validation/ CV) [3][20]. Σύμφωνα με αυτήν το σύνολο των παραδειγμάτων, $S = \{\mathbf{x}_i, y_i\}_{i=1}^N$, που εισάγεται στο σύστημα, διαιρείται σε K ισομεγέθη υποσύνολα (φάκελοι). Έχοντας K σύνολα δεδομένων εκπαίδευσης, διενεργούνται K πειράματα. Όπως φαίνεται στο Διάγραμμα 3.4, σε κάθε πείραμα, το σύστημα εκπαιδεύεται στην ένωση $K-1$ συνόλων δεδομένων εκπαίδευσης και αποτιμάται στο εναπομένον σύνολο δεδομένων, το οποίο είναι διαφορετικό σε κάθε πείραμα. Έτσι, κάθε φάκελος έχει συμμετάσχει στην διαδικασία εκπαίδευσης και ελέγχου $K-1$, και μία φορά αντίστοιχα. Το αποτέλεσμα της τεχνικής αυτής είναι η παραγωγή K αποδόσεων για καθένα από τα K μοντέλα. Τελικά, η απόδοση του συστήματος (σφάλμα του ταξινομητή) εξάγεται υπολογίζοντας τον μέσο όρο αυτών. Συνηθέστερες τιμές για το K είναι τα 5, 10 οδηγώντας σε πενταπλή και δεκαπλή διασταυρωμένη επικύρωση αντίστοιχα. Το πλεονέκτημα της μεθόδου K -fold cross validation σε σχέση με της Random Subsampling είναι ότι όλα τα δείγματα χρησιμοποιούνται κάποια στιγμή και στην εκπαίδευση και στη δοκιμή. Το τελικό σφάλμα εφαρμόζοντας αυτή την τεχνική είναι:

$$err_{CV} = \frac{1}{K} \frac{K}{N} \sum_{k=1}^K \sum_{i=1}^{N/K} L(y_i, y'_i).$$

Η διασταυρωμένη επικύρωση με αποκλείσιμο μιας τιμής (Leave-One-Out-CV) αποτελεί ξεχωριστή μορφή διασταυρωμένης επικύρωσης όπου το K είναι όσα και τα

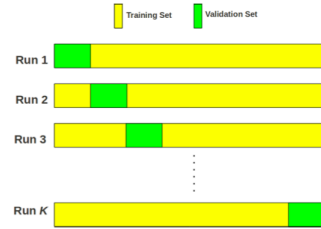
Διάγραμμα 3.2: Μέθοδος Hold out.



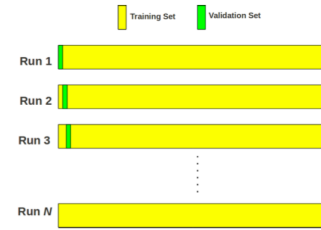
Διάγραμμα 3.3: Μέθοδος Random subsampling.



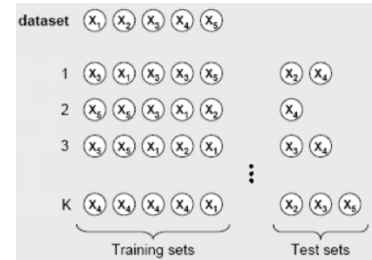
Διάγραμμα 3.4: Μέθοδος k fold cross validation.



Διάγραμμα 3.5: Μέθοδος LOOCV.



Διάγραμμα 3.6: Μέθοδος bootstrap.



διαθέσιμα παραδείγματα, έστω N . Δημιουργούνται N μοντέλα, κάθε ένα από αυτά εκπαιδεύεται χρησιμοποιώντας $N-1$ δείγματα και στο εναπομένον ένα επαληθεύεται (Διάγραμμα 3.5). Η τεχνική αυτή χρησιμοποιεί το μέγιστο δυνατό υποσύνολο των δεδομένων για εκπαίδευση αλλά είναι μια υπερβολικά δαπανηρή σε κόστος διαδικασία. Το σφάλμα υπολογίζεται ως μέσος όρος σφαλμάτων των N μοντέλων και είναι:

$$err_{LOOCV} = \frac{1}{N} \sum_{i=1}^N L(y_i, y'_i).$$

Τέλος, με την τεχνική Bootstrap από ένα πλήθος παραδειγμάτων (N) επιλέγουμε N τυχαία δείγματα με αντικατάσταση για εκπαίδευση. Τα στοιχεία που δεν επιλέχθηκαν καθόλου προσρίζονται για τον έλεγχο του αλγορίθμου (Διάγραμμα 3.6). Μπορούν να γίνουν K επαναλήψεις του παραπάνω πειράματος. Εφαρμόζοντας την μέθοδο Bootstrap το σφάλμα υπολογίζεται ως εξής:

$$err_{boot} = \frac{1}{KB} \sum_{k=1}^K \sum_{i=1}^{B_k} L(y_i, y'_i),$$

όπου B είναι ο συνολικός αριθμός των παραδειγμάτων που δεν επιλέχθηκαν στα K πειράματα και B_k ο αριθμός των παραδειγμάτων που δεν επιλέχθηκαν στο k -οστό πείραμα.

3.4 Κριτήρια απόδοσης ταξινομητή

Οι μέθοδοι cross validation και bootstrap μας βοηθούν να προσεγγίσουμε όσο το δυνατόν περισσότερο το πραγματικό σφάλμα του ταξινομητή (err_{true}) και τελικά κατευθύνει την επιλογή μας στο μοντέλο με το μικρότερο σφάλμα. Στην Παράγραφο αυτή θα παρουσιάσουμε κάποια κριτήρια που αξιολογούν το σφάλμα του τελικά επιλεχθέντος ταξινομητή χρησιμοποιώντας το σύνολο ελέγχου [20][50].

Πίνακας σύγχυσης

Έστω ένα πρόβλημα ταξινόμησης δύο κλάσεων ($\omega_1 = 1, \omega_2 = 0$). Δοθέντος ενός συνόλου A της μορφής $A = [(h(\mathbf{x}_1), y_1), \dots, (h(\mathbf{x}_N), y_N)]$, μπορούμε να διακρίνουμε τέσσερις πιθανές εκβάσεις στο δυαδικό αυτό πρόβλημα ταξινόμησης. Η τυχαία μεταβλητή \mathbf{X} , ($\mathbf{X} = X^1, \dots, X^l$), αποτελεί την είσοδο του συστήματος. Τα y_i δηλώνουν την αληθινή κλάση ενώ τα $h(\mathbf{x}_i)$ τις προβλέψεις για τα διανύσματα \mathbf{x}_i που παίρνουν τιμές από την \mathbf{X} . Συμβολίζουμε ως Αληθώς Θετικά (True Positive-TP) και ως Αληθώς Αρνητικά (True Negative-TN) τα στιγμιότυπα που το μοντέλο ταξινόμησης ορθά στην θετική $\omega_1 = 1$ ή ορθά στην αρνητική ($\omega_2 = 0$) κλάση αντίστοιχα [54][50]. Δηλαδή, είναι:

$$TP = \sum_{i=1}^N \mathbb{1}[y_i = 1, h(\mathbf{x}_i) = 1], \quad (3.10)$$

$$TN = \sum_{i=1}^N \mathbb{1}[y_i = 0, h(\mathbf{x}_i) = 0].$$

Οι άλλες δύο περιπτώσεις αναφέρονται σε εσφαλμένες ταξινομήσεις του ταξινομητή. Ψευδώς θετικά (False Positive-FP) αναφέρονται ως τα παραδείγματα που ο ταξινομητής προβλέπει ότι ανήκουν στην θετική κλάση ($\omega_1 = 1$) ενώ ανήκουν στην Αρνητική ($\omega_2 = 0$) και ως ψευδώς αρνητικά (False Negative-FN) τα παραδείγματα που κατηγοριοποιήθηκαν λανθασμένα στην αρνητική κλάση. Δηλαδή, είναι:

$$FP = \sum_{i=1}^N \mathbb{1}[y_i = 0, h(\mathbf{x}_i) = 1], \quad (3.11)$$

$$FN = \sum_{i=1}^N \mathbb{1}[y_i = 1, h(\mathbf{x}_i) = 0].$$

Οι σχέσεις (3.10), (3.11) δημιουργούν τον πίνακα σύγχυσης (confusion matrix) όπως φαίνεται στο Διάγραμμα 3.1. Αναπαριστώντας τα αποτελέσματα της αξιολόγησης σε ένα τέτοιο πίνακα είναι εύκολο να διαπιστωθεί αν ο κατηγοριοποιητής διακρίνει σωστά τις κλάσεις του προβλήματος.

	Predicted		
	Positive	Negative	
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Πίνακας 3.1: Πίνακας σύγχυσης στο πρόβλημα ταξινόμησης δύο κλάσεων.

Ακρίβεια

Μια συχνή μετρική για την αξιολόγηση του μοντέλου είναι η ακρίβεια (accuracy) που ορίζεται να είναι οι σωστές προβλέψεις προς το συνολικό πλήθος των προβλέψεων:

$$Accuracy = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[y_i = h(\mathbf{x}_i)].$$

Σύμφωνα με τον πίνακα σύγχυσης η ακρίβεια μπορεί να γραφεί και ως:

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

Ευαισθησία, Εξειδίκευση

Προκειμένου να διαχωρίσουμε τα Ψευδώς Θετικά από τα Ψευδώς Αρνητικά, υπολογίζουμε ξεχωριστά το ποσοστό των σωστών προβλέψεων των παραδειγμάτων που ταξινομήθηκαν στην θετική και αρνητική κλάση αντίστοιχα. Ως ευαισθησία (sensitivity) ορίζουμε τον ρυθμό των αληθώς θετικών προβλέψεων ο οποίος υπολογίζεται ως [54]:

$$sensitivity = \frac{TP}{TP + FN}.$$

Ένας ταξινομητής με 100% ευαισθησία σημαίνει ταξινομεί όλα τα παραδείγματα θετικής κλάσης σωστά, χωρίς όμως αυτό να γενικεύεται στο πλήθος των παραδειγμάτων. Η εξειδίκευση (specificity) συμβολίζει το ρυθμό αληθώς αρνητικών στιγμιотύπων και υπολογίζεται ως:

$$specificity = \frac{TN}{TN + FP}.$$

Οι όροι ευαισθησία και εξειδίκευση αναφέρονται και ως ανάκληση (recall) θετικής και αρνητικής κλάσης αντίστοιχα. Ένας ταξινομητής είναι αποδοτικός, αν έχει υψηλή ευαισθησία και υψηλή εξειδίκευση. Ο μέσος όρος αυτών των δύο ονομάζεται ισορροπημένη

ακρίβεια (balanced accuracy-BA) και υπολογίζεται ως:

$$BA = \frac{\textit{sensitivity} + \textit{specificity}}{2}$$

$$= \frac{1}{2} \frac{TP}{TP + FN} + \frac{1}{2} \frac{TN}{TN + FP}.$$

Ορθότητα

Η ορθότητα (precision) ορίζεται, δεδομένου μιας κλάσης, ως το πλήθος των παραδειγμάτων που ταξινομήθηκαν ορθά προς το σύνολο των στιγμιοτύπων που κατηγοροποιήθηκαν σε αυτήν. Σύμφωνα με τον πίνακα σύγχυσης για το πρόβλημα δύο κλάσεων είναι [38]:

$$\textit{precision}_P = \frac{TP}{TP + FP},$$

$$\textit{precision}_N = \frac{TN}{TN + FN}.$$

F-measure

Μια εικόνα για την απόδοση του ταξινομητή μας δίνει η συνάρτηση F (F-measure) που ορίζεται για κάθε κλάση ως ο αρμονικός μέσος της ανάκλησης και της ορθότητας [38][54]. Δηλαδή:

$$F_P = \frac{2}{\frac{1}{\textit{precision}_P} + \frac{1}{\textit{sensitivity}}},$$

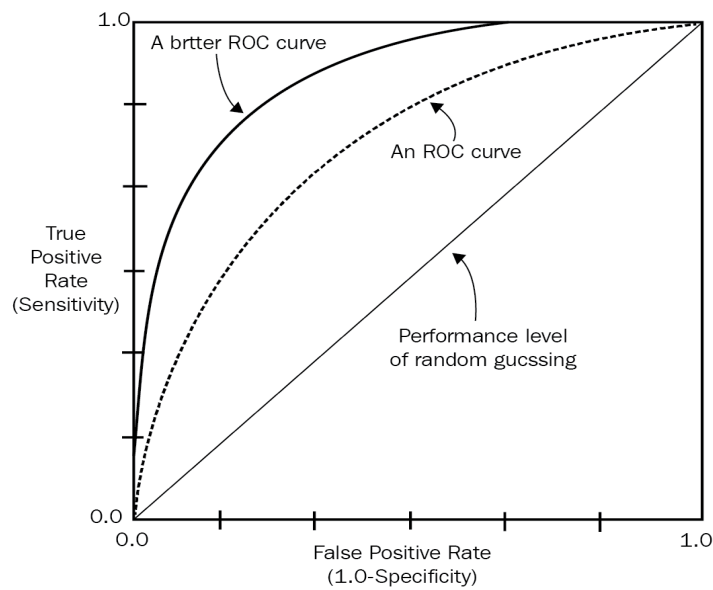
$$F_N = \frac{2}{\frac{1}{\textit{precision}_N} + \frac{1}{\textit{specificity}}}.$$

Παρατηρούμε ότι το μέτρο F παίρνει μεγάλη τιμή όταν η ανάκληση και η ακρίβεια είναι υψηλές.

Καμπύλη ROC και εμβαδόν κάτω από την καμπύλη ROC (AUC)

Η καμπύλη ROC (Receiver Operating Characteristic curve) αναπαριστά γραφικά την προβλεπτική ικανότητα του κατηγοριοποιητή στο δυαδικό πρόβλημα ταξινόμησης. Πιο συγκεκριμένα, δείχνει την αναλογία του ρυθμού των ορθών ταξινομήσεων των θετικών παραδειγμάτων (sensitivity) προς τον ρυθμό των εσφαλμένων ταξινομήσεων αυτών (1-specificity) για τις διάφορες τιμές του κατωφλιού απόφασης (θ) το οποίο καθορίζει την κλάση του κάθε παραδείγματος.

Οι καμπύλες του Διαγράμματος 3.7 δείχνουν τις αποδόσεις τριών ταξινομητών. Κάθε μια δημιουργήθηκε από σημεία που αντιστοιχούν στον ταξινομητή που θα προέκυπτε από διαφορετικές τιμές του κατωφλιού θ . Παρατηρούμε ότι το σημείο



Διάγραμμα 3.7: Παραδείγματα τριών καμπυλών ROC.

(0,1) αντιστοιχεί στον τέλει κατηγοριοποιητή καθώς έχει 100% ευαισθησία και 100% εξειδίκευση. Η διαγώνιος δείχνει την απόδοση ενός τυχαίου ταξινομητή. Όταν η καμπύλη είναι πάνω από την διαγώνιο τότε ο ταξινομητής έχει απόδοση άνω του μετρίου, διαφορετικά η απόδοση του είναι φτωχή.

Ένα άλλο μέτρο αξιόγησης είναι το εμβαδόν κάτω από την καμπύλη ROC, γνωστό ως AUC (area under the ROC curve). Οι τιμές του μέτρου AUC κυμαίνονται στο διάστημα [0,1]. Γνωρίζοντας ότι το αντίστοιχο εμβαδόν για ένα τυχαίο ταξινομητή είναι 0.5, οι αποδοτικοί ταξινομητές πρέπει να έχουν τιμή πάνω από 0.5.

Το στάδιο της αποτίμησης της απόδοσης ενός ταξινομητή είναι αναπόσπαστο τμήμα της διαδικασίας σχεδίασης του μοντέλου. Η αποτίμηση της απόδοσης του συστήματος θα καθορίσει αν το σύστημα είναι σύμφωνο με τις απαιτήσεις που θέτει η εκάστοτε εφαρμογή και ο επιθυμητός τρόπος χρήσης του συστήματος. Συνεπώς, εφαρμόζοντας τις μεθόδους βελτίωσης που περιγράψαμε και χρησιμοποιώντας τα κριτήρια απόδοσης του μοντέλου που αναφέραμε σε αυτό το κεφάλαιο μπορούμε να σχεδιάσουμε έναν ταξινομητή αποτελεσματικό, ικανό να κατηγοριοποιήσει σωστά μελλοντικά στιγμιότυπα.

Κεφάλαιο 4

Τεχνικές ταξινόμησης βασισμένες στην θεωρία αποφάσεων κατά Μπέυζ

4.1 Μπεϋζιανοί ταξινομητές

4.1.1 Εισαγωγή

Η έννοια της υπό συνθήκη πιθανότητας είναι ιδιαίτερα χρήσιμη. Υπάρχουν αμέτρητα παραδείγματα στο κόσμο γύρω μας όπου η πιθανότητα να συμβεί ένα γεγονός εξαρτάται από την πιθανότητα ενός άλλου. Στόχος της ενότητας αυτής είναι η κατασκευή αλγορίθμων που βασίζονται στην θεωρία αποφάσεων κατά Μπέυζ (Bayes). Με βάση την πιθανοθεωρητική αυτή προσέγγιση κατασκευάζονται αλγόριθμοι που εκτιμούν την πιθανότητα ένα νέο παράδειγμα να ανήκει σε μια από τις υπάρχουσες κλάσεις. Μάλιστα, οι Μπεϋζιανοί ταξινομητές επιτυγχάνουν μεγάλες ταχύτητες έχοντας υψηλή απόδοση συγκρίσιμη με αυτή άλλων αλγορίθμων.

Η μέθοδος Μπέυζ προσπαθεί να εντοπίσει την πιο πιθανή υπόθεση (h) από ένα σύνολο υποθέσεων, δεδομένου των ήδη ταξινομημένων παραδειγμάτων και της όποιας εκ των προτέρων γνώσης διαθέτουμε για το πρόβλημα, συνδιάζοντας βέβαια τον κανόνα Μπέυζ. Για το πρόβλημα ταξινόμησης, όπως θα δούμε παρακάτω χρησιμοποιούμε την διανυσματική περιγραφή των δεδομένων όπου το σύνολο εκπαίδευσης (S) αντιστοιχεί στις N παρατηρήσεις, $S = (\mathbf{x}_1, \dots, \mathbf{x}_N)$, και οι υποθέσεις στις m κλάσεις, $h = (\omega_1, \dots, \omega_m)$.

Στις επόμενες Παραγράφους αναλύονται βασικές έννοιες της θεωρίας του

Μπέυζ και αποδυναμώνεται πως ο ταξινομητής που βασίζεται στην θεωρία αυτή είναι ο πιο αποδοτικός. Επίσης, αναλύονται τέσσερις μορφές του Μπεύζιανού ταξινομητή, η Τετραγωνική και Γραμμική Διαχωριστική ανάλυση, ο Αφελής ταξινομητής και τα Δίκτυα Μπέυζ.

4.1.2 Θεωρία αποφάσεων κατά Μπέυζ

Η ιδέα της θεωρίας αποφάσεων κατά Μπέυζ είναι η ταξινόμηση ενός στιγμιότυπου στην κλάση που έχει την μεγαλύτερη εκ των υστέρων (a posteriori) πιθανότητα. Σκοπός ενός προβλήματος ταξινόμησης με κλάσεις $\omega_1, \omega_2, \dots, \omega_m$, όπου m το πλήθος αυτών, είναι, δεδομένου ενός αγνώστου χαρακτηριστικού διανύσματος \mathbf{x} (που δεν ανήκει στο σύνολο εκπαίδευσης), ο προσδιορισμός της πιο πιθανής ετικέτας του. Οι δύο πληροφορίες που συνδιάζει ένας κατά Μπέυζ ταξινομητής στην παραπάνω διαδικασία επιλογής είναι οι $P(Y = \omega_i)$ και $p(\mathbf{X} = \mathbf{x} | Y = \omega_i)$. Το $\omega_i, i = 1, 2, \dots, m$ φανερώνει την κατάσταση της φύσης, δηλαδή ποια είναι η σωστή κατηγορία για κάποιο δεδομένο \mathbf{x} . Επειδή η κατάσταση στην φύση είναι απρόβλεπτη, το ω θεωρείται σαν μια μεταβλητή που πρέπει να περιγραφεί πιθανολογικά. Η πιθανότητα $P(Y = \omega)$ ονομάζεται εκ των προτέρων (a priori) πιθανότητα και αντανακλά την εκ των προτέρων γνώση μας για το πόσο πιθανό είναι να κατεμηθεί ένα παράδειγμα \mathbf{x} σε μια κλάση. Μπορεί εύκολα να προσεγγισθεί από τον εξής τύπο: $P(Y = \omega_i) \approx N_i/N$ όπου N_i είναι το πλήθος των διανυσμάτων εκπαίδευσης που ανήκουν στην κλάση ω_i και $N = N_1 + \dots + N_m$ [52].

Η δεύτερη πληροφορία που ποσοτικοποιείται είναι η συνάρτηση πυκνότητας πιθανότητας/ σ.π.π. (probability density function/p.d.f.) των διανυσματικών χαρακτηριστικών \mathbf{x} δεδομένου ότι η κατάσταση της φύσης είναι ω_i , συμβολίζεται με $p(\mathbf{X} = \mathbf{x} | Y = \omega_i)$ γνωστή και ως πιθανοφάνεια (likelihood) του ω_i ως προς το \mathbf{x} . Η πιθανότητα αυτή είναι επίσης υπολογίσιμη όταν διατίθεται ικανοποιητικά μεγάλος αριθμός δειγμάτων με την αντίστοιχη ‘ετικέτα’ τους. Θεωρώντας γνωστές αυτές τις δύο ποσότητες, η ζητούμενη εκ των υστέρων πιθανότητα $P(Y = \omega_i | \mathbf{X} = \mathbf{x})$, συνδυάζοντας τον Μπεύζιανό κανόνα και το θεώρημα ολικής πιθανότητας για την σ.π.π. του \mathbf{X} , ($p(\mathbf{X} = \mathbf{x})$), γράφεται ως:

$$P(Y = \omega_i | \mathbf{X} = \mathbf{x}) = \frac{p(\mathbf{X}=\mathbf{x}|Y=\omega_i)P(Y=\omega_i)}{p(\mathbf{X}=\mathbf{x})} = \frac{p(\mathbf{X}=\mathbf{x}|Y=\omega_i)P(Y=\omega_i)}{\sum_{i=1}^m p(\mathbf{X}=\mathbf{x}|Y=\omega_i)P(Y=\omega_i)}. \quad (4.1)$$

Αξίζει να σημειωθεί ότι αν οι τιμές που λαμβάνουν τα διανύσματα χαρακτηριστικών είναι διακριτές τότε οι συναρτήσεις πυκνότητας $p(\mathbf{X} = \mathbf{x} | Y = \omega_i)$ γίνονται πιθανότητες και συμβολίζονται με $P(\mathbf{X} = \mathbf{x} | Y = \omega_i)$. Έστω λοιπόν ότι έχουμε ένα πρότυπο \mathbf{x} το οποίο αντιπροσωπείται από ένα l -διάστατο διάνυσμα χαρακτηριστικών, δηλαδή $\mathbf{x} = (x^1, x^2, \dots, x^l)^T \in \mathbb{R}^l$. Τότε το \mathbf{x} αναγνωρίζεται ότι ανήκει στην κλάση ω_j αν [16][20]:

$$P(Y = \omega_j | \mathbf{X} = \mathbf{x}) > P(Y = \omega_i | \mathbf{X} = \mathbf{x}) \quad \forall i = 1, 2, \dots, m.$$

Χρησιμοποιώντας την σχέση (4.1) είναι:

$$p(\mathbf{X} = \mathbf{x} | Y = \omega_j)P(Y = \omega_j) > p(\mathbf{X} = \mathbf{x} | Y = \omega_i)P(Y = \omega_i) \quad \forall i = 1, 2, \dots, m.$$

Δηλαδή ένας Μπεύζιανός ταξινομητής κατηγοριοποιεί το \mathbf{x} στην κλάση ω_j για την οποία η ποσότητα $p(\mathbf{X} = \mathbf{x} | Y = \omega_j)P(Y = \omega_j)$ λαμβάνει την μεγαλύτερη τιμή.

Δηλαδή:

$$p(\mathbf{X} = \mathbf{x}|Y = \omega_j)P(Y = \omega_j) = \max_{i=1,2,\dots,m} p(\mathbf{X} = \mathbf{x}|Y = \omega_i)P(Y = \omega_i). \quad (4.2)$$

Παρατηρείται ότι η $p(\mathbf{X} = \mathbf{x})$ δεν υπεσέρχεται στους υπολογισμούς μας μιας και είναι μια θετική ποσότητα ανεξάρτητη των κλάσεων. Μάλιστα, με τον τρόπο αυτό η ολική πιθανότητα λάθους είναι η μικρότερη δυνατή καθώς ισχύει ότι:

$$P(\text{error}) = 1 - \max[P(Y = \omega_i|\mathbf{X} = \mathbf{x}), \dots, P(Y = \omega_m|\mathbf{X} = \mathbf{x})].$$

Ελαχιστοποίηση πιθανότητας σφάλματος ταξινόμησης

Προκειμένου να δικαιολογηθεί αυτή η διαδικασία επιλογής του Μπεϋζιανού ταξινομητή, υπολογίζεται η πιθανότητα λάθους όταν παίρνεται μια λάθος απόφαση. Γενικά, δίνεται από τον τύπο [20]:

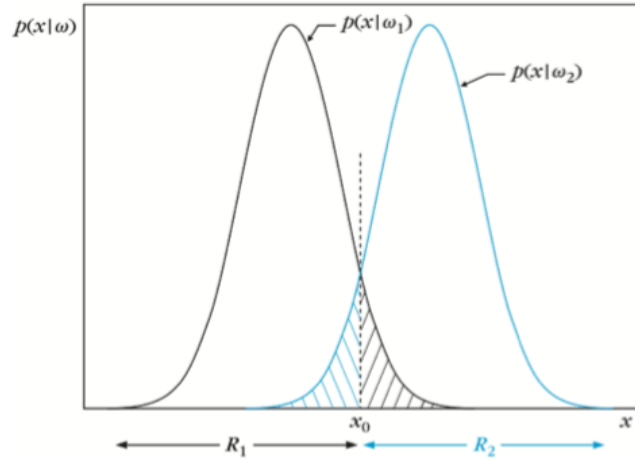
$$P(\text{error}) = \sum_{i=1}^m \int_{R_i} \left(\sum_{k=1, k \neq i}^m p(\mathbf{X} = \mathbf{x}|Y = \omega_k)P(Y = \omega_k) \right) d\mathbf{x}, \quad (4.3)$$

όπου R_i αναφέρεται στην υποπεριοχή του χώρου των χαρακτηριστικών που αντιστοιχεί στην κλάση ω_i . Για χάρη απλότητας εστιάζουμε στο πρόβλημα δύο κλάσεων, ω_1, ω_2 αντίστοιχα. Στο Διάγραμμα 4.1 φαίνονται δύο ισοπίθανες κλάσεις και η μεταβολή $p(\mathbf{X} = \mathbf{x}|Y = \omega_i), i = 1, 2$ ως συναρτήσεως του \mathbf{x} για την απλή περίπτωση ενός απλού χαρακτηριστικού. Ο ταξινομητής Μπέυζ ορίζει το σημείο \mathbf{x}_0 το οποίο χωρίζει τον χώρο σε δυο περιοχές R_1 όπου για κάθε διάνυσμα \mathbf{x} ισχύει: $p(\mathbf{X} = \mathbf{x}|Y = \omega_1) > p(\mathbf{X} = \mathbf{x}|Y = \omega_2)$ και την R_2 όπου $p(\mathbf{X} = \mathbf{x}|Y = \omega_2) > p(\mathbf{X} = \mathbf{x}|Y = \omega_1)$. Εσφαλμένη ταξινόμηση γίνεται όταν κάποιο παράδειγμα \mathbf{x} βρίσκεται ταυτόχρονα και στις δύο κλάσεις. Κάνοντας χρήση του τύπου της δεσμευμένης πιθανότητας, η πιθανότητα λάθους στην περίπτωση των δύο κλάσεων είναι:

$$\begin{aligned} P(\text{error}) &= P(\mathbf{X} = \mathbf{x} \in R_2, Y = \omega_1) + P(\mathbf{X} = \mathbf{x} \in R_1, Y = \omega_2) \\ &= p(\mathbf{X} = \mathbf{x} \in R_2|Y = \omega_1)P(Y = \omega_1) + p(\mathbf{X} = \mathbf{x} \in R_1|Y = \omega_2)P(Y = \omega_2) \\ &= P(Y = \omega_2) \int_{R_1} p(\mathbf{X} = \mathbf{x}|Y = \omega_2)d\mathbf{x} + P(Y = \omega_1) \int_{R_2} p(\mathbf{X} = \mathbf{x}|Y = \omega_1)d\mathbf{x} \\ &= \frac{1}{2} \int_{-\infty}^{\mathbf{x}_0} p(\mathbf{X} = \mathbf{x}|Y = \omega_2)d\mathbf{x} + \frac{1}{2} \int_{\mathbf{x}_0}^{+\infty} p(\mathbf{X} = \mathbf{x}|Y = \omega_1)d\mathbf{x}, \end{aligned}$$

όπου $P(\cdot, \cdot)$ η από κοινού πιθανότητα των δύο γεγονότων. Με το παραπάνω σκεπτικό ένας Μπεϋζιανός ταξινομητής μπορεί να κατηγοριοποιεί ένα άγνωστο παράδειγμα \mathbf{x} στην πιο πιθανή κλάση. Από το Διάγραμμα 4.1 φαίνεται ότι για κάθε $\mathbf{x} \neq \mathbf{x}_0$ η πιθανότητα λάθους ταξινόμησης αυξάνεται καθώς το αντίστοιχο γραμμοσκιασμένο εμβαδόν κάτω από τις καμπύλες μεγαλώνει.

Πρόταση 4.1.1. *Ο κατά Μπέυζ ταξινομητής είναι βέλτιστος ως προς την ελαχιστοποίηση της πιθανότητας του σφάλματος ταξινόμησης.*



Διάγραμμα 4.1: Υπο συνθήκη συναρτήσεις πυκνότητας πιθανότητας για δύο κλάσες ω_1, ω_2 , και οι περιοχές R_1, R_2 που σχηματίζονται από τον ταξινομητή κατά Μπέυζ στην περίπτωση ισοπίθανων κλάσεων.

Απόδειξη. Από το Διάγραμμα (4.1) φαίνεται ότι:

$$\int_{R_1} p(\mathbf{X} = \mathbf{x}|Y = \omega_1)d\mathbf{x} + \int_{R_2} p(\mathbf{X} = \mathbf{x}|Y = \omega_1)d\mathbf{x} = 1, \quad (4.4)$$

η οποία με την βοήθεια του κατά Μπέυζ κανόνα παίρνει την μορφή:

$$\int_{R_1} P(Y = \omega_1|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} + \int_{R_2} P(Y = \omega_1|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} = P(Y = \omega_1).$$

Αντικαθιστώντας τον τύπο (4.4) στην εξίσωση πιθανότητας λάθους, έχουμε:

$$\begin{aligned} P(\text{error}) &= P(Y = \omega_1) \int_{R_2} p(\mathbf{X} = \mathbf{x}|Y = \omega_1)d\mathbf{x} + P(Y = \omega_2) \int_{R_1} p(\mathbf{X} = \mathbf{x}|Y = \omega_2)d\mathbf{x} \\ &= \int_{R_2} P(Y = \omega_1|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} + \int_{R_1} P(Y = \omega_2|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} \\ &= P(Y = \omega_1) - \int_{R_1} P(Y = \omega_1|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} \\ &\quad + \int_{R_1} P(Y = \omega_2|\mathbf{X} = \mathbf{x})p(\mathbf{X} = \mathbf{x})d\mathbf{x} \\ &= P(Y = \omega_1) - \int_{R_1} (P(Y = \omega_1|\mathbf{X} = \mathbf{x}) - P(Y = \omega_2|\mathbf{X} = \mathbf{x}))p(\mathbf{X} = \mathbf{x})d\mathbf{x}. \end{aligned}$$

□

Από αυτό το αποτέλεσμα συμπαιραίνουμε ότι η πιθανότητα λάθους γίνεται η μικρότερη δυνατή αν η R_1 είναι η περιοχή του χώρου των χαρακτηριστικών για την οποία $P(Y = \omega_1|\mathbf{X} = \mathbf{x}) > P(Y = \omega_2|\mathbf{X} = \mathbf{x})$. Αντίστοιχα για την περιοχή R_2 .

Στην συνέχεια θα δούμε ότι ισχύουν οι γενικεύσεις για την περίπτωση περισσότερων κλάσεων. Ισχύει ότι:

$$P(Y = \omega_1) = \sum_{i=1}^m \int_{R_i} P(Y = \omega_1 | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) d\mathbf{x}$$

$$\Rightarrow P(Y = \omega_1) - \int_{R_1} P(Y = \omega_1 | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) d\mathbf{x} = \sum_{i=2}^m \int_{R_i} P(Y = \omega_1 | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) d\mathbf{x}.$$

Αντικαθιστώντας την τελευταία στην εξίσωση πιθανότητας λάθους (4.3), έχουμε:

$$P(\text{error}) = P(Y = \omega_1) - \left(\int_{R_1} P(Y = \omega_1 | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) d\mathbf{x} - P(Y = \omega_2 | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) - \dots - P(Y = \omega_m | \mathbf{X} = \mathbf{x}) p(\mathbf{X} = \mathbf{x}) \right)$$

$$+ \sum_{i=2}^m \int_{R_i} \left(\sum_{k=2, k \neq i}^m p(\mathbf{X} = \mathbf{x} | Y = \omega_k) P(Y = \omega_k) \right) d\mathbf{x}.$$

Από αυτό το αποτέλεσμα συμπαράινουμε ότι και στην περίπτωση των m κλάσεων η πιθανότητα λάθους γίνεται η μικρότερη δυνατή αν η R_1 είναι η περιοχή του χώρου των χαρακτηριστικών για την οποία ισχύει έχει την μεγαλύτερη εκ των υστέρων πιθανότητα $P(Y = \omega_1 | \mathbf{X} = \mathbf{x})$ από τις υπόλοιπες κλάσεις. Αντίστοιχα για τις περιοχές R_2, \dots, R_m . Συνεπώς, ένα άγνωστο πρότυπο, το οποίο αναπαριστάται από το διάνυσμα χαρακτηριστικών \mathbf{x} καταχωρείται στην κλάση ω_i , αν:

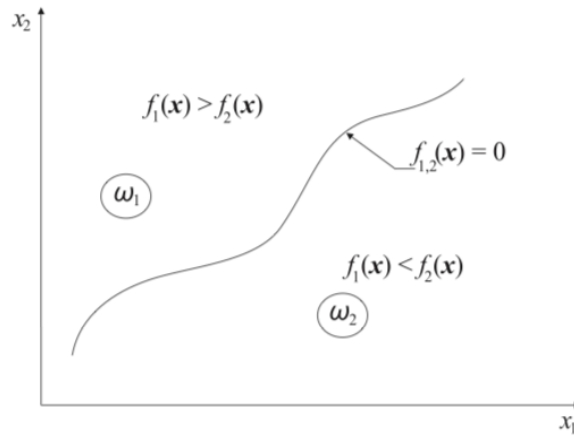
$$P(Y = \omega_i | \mathbf{X} = \mathbf{x}) > P(Y = \omega_j | \mathbf{X} = \mathbf{x}) \quad \forall j \neq i.$$

4.1.3 Συναρτήσεις διάκρισης και επιφάνειες απόφασης

Στην προηγούμενη Παράγραφο παρουσιάστηκε ο Μπεϋζιανός ταξινομητής και μελετήθηκε το σχεπτικό διαίρεσης του l -διάστατου χώρου των χαρακτηριστικών σε περιοχές ταξινόμησης $R_i, i = 1, \dots, m$ το οποίο βασίστηκε στις ποσότητες $P(Y = \omega_i) p(\mathbf{X} = \mathbf{x} | Y = \omega_i)$ και πού κάθε μια από αυτές παίρνει την μεγαλύτερη τιμή από τις υπόλοιπες. Τις περισσότερες φορές εκφράζουμε τον κατά Μπέυζ ταξινομητή με την βοήθεια συναρτήσεων της μορφής $g_q(\mathbf{x}) = f(P(Y = \omega_q) p(\mathbf{X} = \mathbf{x} | Y = \omega_q))$, $q = 1, 2, \dots, m$ όπου f είναι μια γνησίως αύξουσα συνάρτηση. Στην περίπτωση αυτή, ταξινομούμε ένα συγκεκριμένο παράδειγμα \mathbf{x} στην κλάση ω_j για την οποία $g_j(\mathbf{x}) = \max_{q=1, \dots, m} g_q(\mathbf{x})$. Οι συναρτήσεις διάκρισης (discriminant functions), όπως ονομάζονται, απλουστεύουν τους υπολογισμούς αφού για παράδειγμα αν $f(\mathbf{x}) = \ln(\mathbf{x})$ και $p(\mathbf{X} = \mathbf{x} | Y = \omega)$ είναι εκθετικής μορφής τότε οι υπολογισμοί περιορίζονται στους εκθέτες.

Οι επιφάνειες απόφασης (decision surfaces) είναι μια έννοια που συναντάται συχνά στην ταξινόμηση προτύπων. Στην περίπτωση του Μπεϋζιανού ταξινομητή οι συναρτήσεις διάκρισης που χρησιμοποιούνται είναι της μορφής $g_j(\mathbf{x}) = P(Y = \omega_j) p(\mathbf{X} = \mathbf{x} | Y = \omega_j)$. Δηλαδή, ταξινομούμε ένα δεδομένο \mathbf{x} στην κλάση ω_j για την οποία $\mathbf{x} \in R_j$ όπου $R_j = \{\mathbf{x} \in R^l : g_j(\mathbf{x}) = \max_{q=1, \dots, m} g_q(\mathbf{x})\}$. Αυτό σημαίνει ότι $g_j(\mathbf{x}) - g_q(\mathbf{x}) > 0 \quad \forall j \neq q$. Στην περίπτωση των δύο κλάσεων (Διάγραμμα 4.2) επιφάνεια απόφασης είναι η αυτή που ορίζεται από την εξίσωση $g_{jq}(\mathbf{x}) = g_j(\mathbf{x}) - g_q(\mathbf{x}) = 0$.

Στο πρόβλημα πολλών κλάσεων, οι επιφάνειες απόφασης διαχωρίζουν μόνο γειτονικές περιοχές.



Διάγραμμα 4.2: Επιφάνεια απόφασης στο παράδειγμα δύο κλάσεων.

4.1.4 Τετραγωνική και Γραμμική Διαχωριστική ανάλυση

Στην πράξη, συχνά γίνεται η υπόθεση ότι η Κανονική κατανομή μπορεί να περιγράψει επαρκώς τα δεδομένα κάθε κλάσης. Κατά συνέπεια, ο κατά Μπέυζ ταξινομητής είναι Τετραγώνικος ή Γραμμικός ανάλογα με τις παραδοχές που κάνουμε. Η προσέγγιση αυτή στα προβλήματα ταξινόμησης είναι γνωστή ως Τετραγωνική Διαχωριστική ανάλυση (Quadratic Discriminant Analysis-QDA) ή Γραμμική Διαχωριστική ανάλυση (Linear Discriminant Analysis-LDA). Η πρώτη αποτελεί επέκταση της δεύτερης η οποία αναπτύχθηκε από τον Ronald Fisher (1936) για την επίλυση του προβλήματος ταξινόμησης φυτών. Οι μέθοδοι LDA, QDA είναι ταξινομητές με καλές επιδόσεις και χρησιμοποιούνται στις περιπτώσεις όπου οι τιμές των χαρακτηριστικών είναι συνεχείς. Βέβαια είναι δύσκολο να δεχτούμε ότι η υπόθεση της Κανονικής κατανομής οδηγεί σε μια λογική μοντελοποίηση των στατιστικών δεδομένων. Ωστόσο, επιφάνειες απόφασης Γραμμικής ή Τετραγωνικής μορφής διαμερίζουν λογικά τον χώρο [20][53].

Βασικές γνώσεις

Η υπόθεση της Κανονικής κατανομής γίνεται χάρη στο Κεντρικό Οριακό Θεώρημα (Central Limit Theorem) σύμφωνα με το οποίο το άθροισμα μεγάλου πλήθους ανεξάρτητων και ισόνομων τυχαίων μεταβλητών ακολουθεί μία κατανομή η οποία προσεγγίζει την Κανονική κατανομή [53].

Είναι γνωστό ότι η πυκνότητα πιθανότητας μιας συνεχούς τυχαίας μεταβλητής που ακολουθεί Gaussian κατανομή δίνεται από την σχέση:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad (4.5)$$

όπου $\mu = E[x] = \int_{-\infty}^{+\infty} xp(x)dx$ και $\sigma^2 = E[(x-\mu)^2] = \int_{-\infty}^{+\infty} (x-\mu)^2p(x)dx$ είναι η μέση τιμή και διασπορά αντιστοίχα μιας τυχαίας μεταβλητής. Η εξίσωση (4.5) γενικεύεται και σε περισσότερες διαστάσεις. Έτσι στον l -διάστατο χώρο παίρνει την μορφή:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{l/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right), \quad (4.6)$$

όπου \mathbf{x} είναι το d -διάστατο διάνυσμα στηλών, $\boldsymbol{\mu}$ είναι το μέσο διάνυσμα, δηλαδή $E[[x^1, \dots, x^l]^T]$ και Σ ο $l \times l$ πίνακας συνδιασποράς (covariance matrix) και ορίζεται ως:

$$\Sigma = E[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T] = \begin{bmatrix} (\sigma^1)^2 & \sigma^{12} & \sigma^{13} & \dots & \sigma^{1l} \\ \sigma^{21} & (\sigma^2)^2 & \sigma^{23} & \dots & \sigma^{2l} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma^{l1} & \sigma^{l2} & \sigma^{l3} & \dots & (\sigma^l)^2 \end{bmatrix},$$

όπου $(\sigma^i)^2 = E[(x^i - \mu^i)^2]$, $\sigma^{ij} = \sigma^{ji} = E[(x^i - \mu^i)(x^j - \mu^j)]$. Τα στοιχεία της διαγωνίου του πίνακα είναι οι αντίστοιχες διασπορές των συνιστωσών του τυχαίου διανύσματος \mathbf{x} ενώ τα υπόλοιπα στοιχεία αφορούν τις συνδιασπόρες, ένα μέτρο που δείχνει το βαθμό γραμμικής εξάρτησης των συνιστωσών. Προφανώς, αν οι x^i είναι ανεξάρτητες τυχαίες μεταβλητές δηλαδή $Cov(x^i, x^j) = \sigma^{ij} = E[(x^i - \mu^i)(x^j - \mu^j)] = E[(x^i - \mu^i)]E[(x^j - \mu^j)] = 0$ τότε ο πίνακας Σ γίνεται διαγώνιος. Γενικά το αντίστροφο δεν ισχύει.

Γραμμικός ταξινομητής

Στην περίπτωση του Γραμμικού ταξινομητή γίνονται οι παραδοχές ότι κάθε ομάδα παρατηρήσεων, ω_r , ακολουθεί πολυδιάστατη Κανονική κατανομή με διαφορετική μέση τιμή $\boldsymbol{\mu}_r, r = 1, \dots, m$ αλλά όλες έχουν τον ίδιο πίνακα συνδιασποράς $\Sigma_r = \Sigma$. Θεωρούμε δηλαδή [53][16]:

$$p(\mathbf{X} = \mathbf{x} | Y = \omega_i) \equiv \mathcal{N}(\boldsymbol{\mu}_i, \Sigma) = \frac{1}{(2\pi)^{l/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)\right), \quad (4.7)$$

όπου $\boldsymbol{\mu}_i$ είναι το διάνυσμα της μέσης τιμής των χαρακτηριστικών για την κλάση ω_i . Όπως είδαμε, ο Μπέυζ περιγράφει την ταξινόμηση από τις εκ των υστέρων πιθανότητες:

$$\mathbf{x} \in \omega_i \text{ αν } g_i(\mathbf{x}) > g_j(\mathbf{x}) \quad \forall j \neq i \quad (4.8)$$

όπου $g_i(\mathbf{x}) = P(Y = \omega_i)p(\mathbf{X} = \mathbf{x} | Y = \omega_i)$.

Χρησιμοποιώντας την (4.7), η συνάρτηση διάκρισης παίρνει την μορφή:

$$g_i(\mathbf{x}) = P(Y = \omega_i) \frac{1}{(2\pi)^{l/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)\right). \quad (4.9)$$

Λογαριθμώντας οι υπολογισμοί απλουστεύουν. Άρα:

$$\begin{aligned}
 g'_i(\mathbf{x}) &= -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) + \ln P(Y = \omega_i) - \frac{l}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma| \quad (4.10) \\
 &= -\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x} + \frac{1}{2} \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \boldsymbol{\mu}_i^T \Sigma^{-1} \boldsymbol{\mu}_i + \frac{1}{2} \boldsymbol{\mu}_i^T \Sigma^{-1} \mathbf{x} \\
 &\quad + \ln P(Y = \omega_i) - \frac{l}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma| \\
 &= \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \boldsymbol{\mu}_i^T \Sigma^{-1} \boldsymbol{\mu}_i + \ln P(Y = \omega_i) + c_i,
 \end{aligned}$$

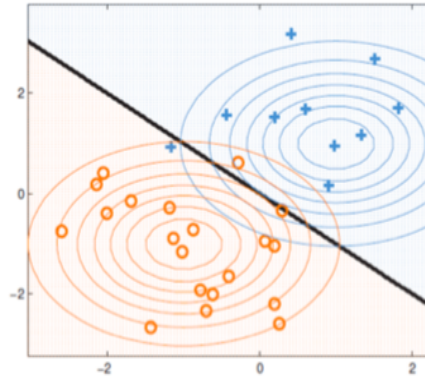
με σταθερά $c_i = -\frac{1}{2} \mathbf{x}^T \Sigma^{-1} \mathbf{x} - \frac{l}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma|$. Η πρόβλεψη για ένα νεό \mathbf{x} είναι η κλάση, ω_i που έχει την μεγαλύτερη $g'_i(\mathbf{x})$. Η επιφάνεια απόφασης σύμφωνα με την λογαριθμική συνάρτηση διάκρισης είναι μια γραμμική εξίσωση ως προς \mathbf{x} :

$$g'_{ij}(\mathbf{x}) = 0$$

$$\Leftrightarrow g'_i(\mathbf{x}) = g'_j(\mathbf{x})$$

$$\Leftrightarrow \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \boldsymbol{\mu}_i^T \Sigma^{-1} \boldsymbol{\mu}_i + \ln P(Y = \omega_i) = \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_j - \frac{1}{2} \boldsymbol{\mu}_j^T \Sigma^{-1} \boldsymbol{\mu}_j + \ln P(Y = \omega_j).$$

Η Γραμμική Διαχωριστική ανάλυση χωρίζει τις παρατηρήσεις όπως φαίνεται στο Διάγραμμα (4.3). Οι επιφάνειες απόφασης που σχηματίζονται ονομάζονται υπερεπίπεδα (hyperplanes). Στην περίπτωση του Διαγράμματος 4.3 όπου έχουμε δύο χαρακτηριστικά τα υπερεπίπεδα αντιστοιχούν σε ευθείες. Κάθε ομάδα παρατηρήσεων έχει διαφορετική μέση τιμή αλλά οι δύο καμπύλες είναι πανομοιότυπες αφού οι δύο κατανομές έχουν τον ίδιο πίνακα διασποράς. Αυτό σημαίνει ότι και οι Κανονικές κατανομές $p(\mathbf{X} = \mathbf{x} | Y = \omega_1), p(\mathbf{X} = \mathbf{x} | Y = \omega_2)$ θα έχουν το ίδιο σχήμα αλλά η μια θα έχει μετακινηθεί δεξιά/αριστερά της αλλής λόγω της διαφορετικής μέσης τιμής [20][53].



Διάγραμμα 4.3: Ταξινόμηση με Τετραγωνική Διαχωριστική ανάλυση σε ένα διάστατο πρόβλημα δύο κλάσεων. Απεικονίζονται οι ισοσταθμικές καμπύλες για τους ίδιους πίνακες Σ κάθε κλάσης και η καμπύλη απόφασης που είναι ευθεία.

Χρησιμοποιώντας το σύνολο εκπαίδευσης $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$ μπορούμε να εκτιμήσουμε τις άγνωστες παραμέτρους:

$$\hat{P}(Y = \omega_i) = \frac{1}{N} \sum_{r=1}^N \mathbb{1}[y_r = \omega_i].$$

Στην συνέχεια, εκτιμούμε την μέση τιμή και την διασπορά με την μέθοδο Μέγιστης Πιθανοφάνειας. Πιο συγκεκριμένα, για κάθε κλάση $\omega_i, i = 1, \dots, m$ το διάνυσμα της μέσης τιμής και ο πίνακας συνδιασποράς είναι:

$$\hat{\boldsymbol{\mu}}_i = \frac{\sum_{r=1}^N \mathbb{1}[y_r = \omega_i] \mathbf{x}_r}{\sum_{r=1}^N \mathbb{1}[y_r = \omega_i]},$$

$$\hat{\Sigma} = \frac{1}{N-m} \sum_{i=1}^m \sum_{k=1}^N \mathbb{1}[y_k = \omega_i] (\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i)(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_i)^T.$$

Ο λόγος που στον παρονομαστή χρησιμοποιούμε αντί για το N το $N-m$ είναι επειδή έτσι ο εκτιμητής είναι αμερόληπτος. Για μεγάλες τιμές του N η διαφορά του N από το $N-m$ είναι πολύ μικρή.

Ταξινομητής Ελαχίστης Ευκλείδειας απόστασης

Στην παράγραφο αυτή, θα ασχοληθούμε με μια ειδική κατηγορία Γραμμικών ταξινομητών όπου υποθέτουμε ότι οι κλάσεις είναι ισοπίθανες και έχουν τον ίδιο διαγώνιο πίνακα διασποράς (ασυσχέτιστα χαρακτηριστικά) [53]. Δηλαδή:

$$\Sigma = \sigma^2 I = \begin{bmatrix} \sigma^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma^2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma^2 \end{bmatrix}.$$

Συνεπώς, ισχύει ότι $\Sigma_i^{-1} = \frac{1}{\sigma^2} I$. Σύμφωνα με τις υποθέσεις, η 4.10 γίνεται:

$$\begin{aligned} g'_i(\mathbf{x}) &= \frac{1}{\sigma^2} \boldsymbol{\mu}_i \mathbf{x}^T - \frac{1}{2\sigma^2} \boldsymbol{\mu}_i^T \boldsymbol{\mu}_i + c_i \\ &= \frac{1}{\sigma^2} \boldsymbol{\mu}_i^T \mathbf{x} - \frac{1}{2\sigma^2} \boldsymbol{\mu}_i^T \boldsymbol{\mu}_i + c_i, \end{aligned}$$

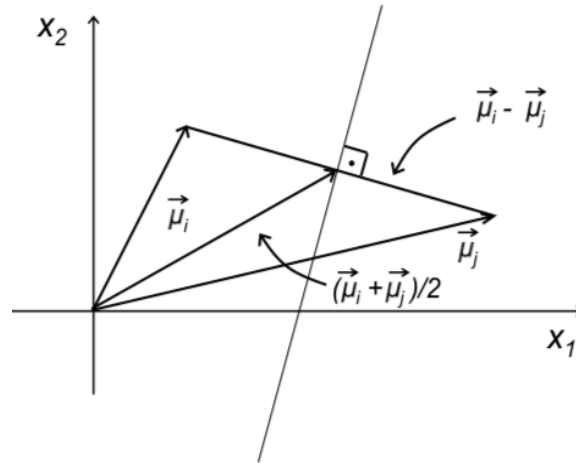
αφού ισχύει ότι $a^T b = (a^T b)^T = b^T a$ η οποία είναι βαθμωτή ποσότητα. Η εξίσωση $g'_{ij}(\mathbf{x}) = g'_i(\mathbf{x}) - g'_j(\mathbf{x})$ βρίσκει την επιφάνεια απόφασης. Είναι:

$$\begin{aligned} g'_{ij}(\mathbf{x}) &= \frac{1}{\sigma^2} \boldsymbol{\mu}_i^T \mathbf{x} - \frac{1}{2\sigma^2} \boldsymbol{\mu}_i^T \boldsymbol{\mu}_i - \frac{1}{\sigma^2} \boldsymbol{\mu}_j^T \mathbf{x} + \frac{1}{2\sigma^2} \boldsymbol{\mu}_j^T \boldsymbol{\mu}_j = 0 \\ \Rightarrow (\boldsymbol{\mu}_i^T - \boldsymbol{\mu}_j^T) \mathbf{x} - \frac{1}{2} (\boldsymbol{\mu}_i^T \boldsymbol{\mu}_i - \boldsymbol{\mu}_j^T \boldsymbol{\mu}_j) &= 0, \end{aligned}$$

η οποία είναι γραμμική εξίσωση καθότι:

$$\begin{aligned} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \mathbf{x} - \frac{1}{2} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T (\boldsymbol{\mu}_i + \boldsymbol{\mu}_j) &= 0 \\ (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \left(\mathbf{x} - \frac{\boldsymbol{\mu}_i + \boldsymbol{\mu}_j}{2} \right) &= 0. \end{aligned}$$

Παρατηρούμε, ότι το $\boldsymbol{\mu}_i - \boldsymbol{\mu}_j$ είναι κάθετο στο υπερεπίπεδο απόφασης το οποίο διέρχεται από τον μέσο όρο των $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j$. Δηλαδή, το $\mathbf{x} - \frac{\boldsymbol{\mu}_i + \boldsymbol{\mu}_j}{2}$ είναι μια επιφάνεια κάθετη στο



Διάγραμμα 4.4: Γραμμή απόφασης για τα Κανονικά κατανομημένα διανύσματα με $\Sigma = \sigma^2 I$ και $P(Y = \omega_i) = P(Y = \omega_j)$.

διάνυσμα $\mu_i - \mu_j$. Τα αποτελέσματα αυτά απεικονίζονται στο Διάγραμμα (4.4) σε ένα πρόβλημα δύο διαστάσεων. Έχοντας ως συνάρτηση διάκρισης την πιο γενική μορφή, με τις ίδιες όμως υποθέσεις, $g_i(\mathbf{x}) = -\frac{1}{2\sigma^2}(\mathbf{x} - \mu_i)^T(\mathbf{x} - \mu_i) = -\frac{1}{2\sigma^2}\|\mathbf{x} - \mu_i\|^2$, καταλαβαίνουμε ότι για τον υπολογισμό της μέγιστης τιμής της αρκεί να υπολογίσουμε την ελάχιστη (λόγω του αρνητικού πρόσημου) Ευκλείδεια απόσταση του \mathbf{x} από τις μέσες τιμές μ_i , δηλαδή την ελάχιστη τιμή των $\|\mathbf{x} - \mu_i\|$. Έτσι ο ταξινομητής Μπέυζ σε αυτήν την περίπτωση μπορεί να διατυπωθεί ως εξής:

$$\mathbf{x} \in \omega_j \text{ αν } \|\mathbf{x} - \mu_j\| = \min_{i=1, \dots, m} \|\mathbf{x} - \mu_i\|.$$

Παρατηρούμε ότι τα σημεία στην αριστερή πλευρά είναι πλησιέστερα στην μ_i και αυτά στην δεξιά πλευρά πλησιέστερα στην μ_j . Έτσι τα διανύσματα χαρακτηριστικών ταξινομούνται σε κλάσεις ανάλογα με την Ευκλείδεια απόστασή τους από τα αντίστοιχα μέσα διανύσματα.

Τετραγωνικός ταξινομητής

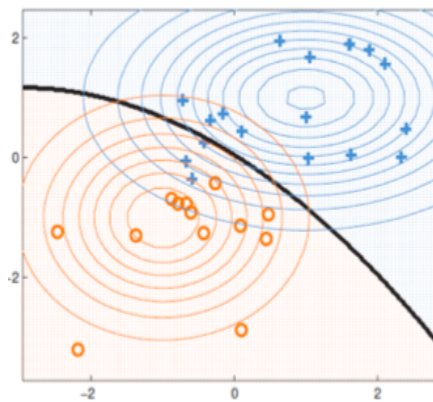
Η υπόθεση ότι ο πίνακας συνδιασποράς Σ είναι ίδιος για όλες τις κλάσεις είναι περιοριστική. Η Τετραγωνική Διαχωριστική ανάλυση λειτουργεί όμοια με την Γραμμική κάνοντας την υπόθεση ότι κάθε κλάση έχει ξεχωριστή μέση τιμή και τον δικό της πίνακα συνδιασποράς [16][53]. Οι παρατηρήσεις κάθε ομάδας ακολουθούν Κανονική κατανομή, δηλαδή:

$$p(\mathbf{X} = \mathbf{x} | Y = \omega_i) \equiv \mathcal{N}(\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{l/2} |\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i)^T \Sigma_i^{-1} (\mathbf{x} - \mu_i)\right).$$

Η λογαριθμοποιημένη συνάρτηση διάκρισης πλέον βρίσκεται στην πιο γενική της μορφή:

$$\begin{aligned} g'_i(\mathbf{x}) &= -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) + \ln P(Y = \omega_i) - \frac{l}{2} \ln 2\pi - \frac{1}{2} \ln |\boldsymbol{\Sigma}_i| \\ &= -\frac{1}{2} \mathbf{x}^T \boldsymbol{\Sigma}_i^{-1} \mathbf{x} + \mathbf{x}^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \boldsymbol{\mu}_i^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\mu}_i + \ln P(Y = \omega_i) + c'(i), \end{aligned}$$

όπου $c'(i) = -\frac{l}{2} \ln 2\pi - \frac{1}{2} \ln |\boldsymbol{\Sigma}_i|$. Στην περίπτωση αυτή, η επιφάνεια απόφασης περιέχει τον τετραγωνικό όρο ($\mathbf{x}^T \boldsymbol{\Sigma}_i^{-1} \mathbf{x}$) και αναπαριστά μια καμπύλη δευτέρου βαθμού (υπερ-έλλειψη, υπερ-παραβολή) και ο ταξινομητής ονομάζεται Τετραγωνικός. Στην περίπτωση 2 χαρακτηριστικών επιφάνειες απόφασης μπορεί να είναι ελλείψεις, παραβολές (Διάγραμμα 4.5), υπερβολές, ζεύγη ευθειών.



Διάγραμμα 4.5: Ταξινόμηση με Γραμμική Διαχωριστική ανάλυση σε ένα διάστατο πρόβλημα δύο κλάσεων. Απεικονίζονται οι ισοσταθμικές καμπύλες για τους ξεχωριστούς πίνακες Σ κάθε κλάσης και η καμπύλη απόφασης.

4.1.5 Αφελής ταξινομητής Μπέυζ

Ένας Αφελής Μπεϋζιανός ταξινομητής (Naive Bayes Classifier) είναι ένας απλός πιθανοτικός ταξινομητής που χρησιμοποιείται στα προβλήματα Επιβλεπόμενης μάθησης και βασίζεται στην Μπεϋζιανή θεωρία αποφάσεων. Είναι κατάλληλος για υψηλής διάστασης προβλήματα καθώς κάνει την εξής ισχυρή υπόθεση γνωστή και ως υπο-συνθήκη ανεξαρτησία (conditional independence): Η επίδραση της τιμής ενός χαρακτηριστικού σε μια δεδομένη κατηγορία είναι ανεξάρτητη από τις τιμές των υπολοίπων χαρακτηριστικών [20]. Εφαρμόζεται κυρίως σε προβλήματα διακριτών χαρακτηριστικών και για τον λόγο αυτό οι συναρτήσεις πυκνότητας πιθανότητας αντικαθιστώνται με πιθανότητες. Στην συνέχεια, δίνεται ο μαθηματικός ορισμός της υπό-συνθήκη ανεξαρτησίας [35][20].

Ορισμός 4.1.1. (Υπο συνθήκη ανεξαρτησία). Έστω τρεις τυχαίες διακριτές μεταβλητές X, Y και Z . Θα λέμε ότι οι X είναι υπό συνθήκη ανεξάρτητη της Y δεδομένης της

Z αν:

$$(\forall i, j, k) P(X = x_i | Y = y_j, Z = z_k) = P(X = x_i | Z = z_k).$$

Για συντομία, γράφουμε:

$$P(X|Y, Z) = P(X|Z).$$

Στόχος σε ένα τυπικό πρόβλημα ταξινόμησης είναι, δοθέντος ενός στιγμιότυπο \mathbf{x} της μορφής $\mathbf{x} = [x^1, \dots, x^j, \dots, x^l]^T$ όπου l η διάσταση του χώρου των χαρακτηριστικών, η εύρεση της πιο πιθανής κατηγορίας που ανήκει. Σύμφωνα με τον κανόνα του Μπέυζ:

$$P(Y = \omega | X^1 = x^1, \dots, X^l = x^l) = \frac{P(Y = \omega)P(X^1 = x^1, \dots, X^l = x^l | Y = \omega)}{P(X^1 = x^1, \dots, X^l = x^l)}.$$

Παρατηρούμε ότι:

$$\begin{aligned} P(Y = \omega)P(X^1 = x^1, \dots, X^l = x^l | Y = \omega) &= P(X^1 = x^1, \dots, X^l = x^l, Y = \omega) \quad (4.11) \\ &= P(X^1 = x^1 | X^2 = x^2, \dots, X^l = x^l, Y = \omega) \\ &P(X^2 = x^2, \dots, X^l = x^l, Y = \omega) \\ &= P(X^1 = x^1 | X^2 = x^2, \dots, X^l = x^l, Y = \omega) \\ &P(X^2 = x^2 | X^3 = x^3, \dots, X^l = x^l, Y = \omega) \dots \\ &\dots P(X^{l-1} = x^{l-1} | X^l = x^l, Y = \omega) \\ &P(X^l = x^l | Y = \omega)P(Y = \omega). \end{aligned}$$

Λόγω της υπόθεσης της ανεξαρτησίας ισχύει $P(X^i = x^i | X^{i+1} = x^{i+1}, \dots, X^l = x^l, Y = \omega) = P(X^i = x^i | Y = \omega)$. Αντικαθιστώντας την τελευταία στην (4.11) έχουμε:

$$P(Y = \omega)P(X^1 = x^1, \dots, X^l = x^l | Y = \omega) = P(Y = \omega) \prod_{j=1}^l P(X^j = x^j | Y = \omega). \quad (4.12)$$

Πλέον ο κανόνας του Μπέυζ παίρνει την μορφή:

$$P(Y = \omega | X^1 = x^1, \dots, X^l = x^l) = \frac{P(Y = \omega) \prod_{j=1}^l P(X^j = x^j | Y = \omega)}{P(X^1 = x^1, \dots, X^l = x^l)}. \quad (4.13)$$

Επειδή η ποσότητα $P(X^1 = x^1, \dots, X^l = x^l)$ είναι ίδια για κάθε κατηγορία παραλείπεται. Σύμφωνα με την θεωρία αποφάσεων κατά Μπέυζ, ένα νέο στιγμιότυπο, έστω \mathbf{x}_{test} , ταξινομείται στην κλάση με την μεγαλύτερη εκ των υστέρων πιθανότητα (έστω ω_k), δηλαδή:

$$\mathbf{x}_{test} \in \omega_k \Leftrightarrow P(Y = \omega_k | \mathbf{X} = \mathbf{x}_{test}) = \max_{r=1, \dots, m} P(Y = \omega_r | \mathbf{X} = \mathbf{x}_{test}).$$

Επομένως, ο απλός Μπέυζιανός ταξινομητής μπορεί να διατυπωθεί ως εξής:

$$\mathbf{x}_{test} \in \omega_k \text{ αν } \omega_k = \underset{\omega_r, r=1, \dots, m}{\operatorname{argmax}} \prod_{j=1}^l P(X^j = x_{test}^j | Y = \omega_r). \quad (4.14)$$

Αν δεν γνωρίζουμε τις πιθανότητες των κατηγοριών $P(Y = \omega_r)$, μπορούμε να υποθέσουμε ότι είναι ίσες. Τότε χρειάζεται να μεγιστοποιηθεί μόνο το $P(X^j = x_{test}^j | Y = \omega)$. Εναλλακτικά, μπορούμε να υπολογίσουμε την εκ των προτέρων πιθανότητα από το σύνολο εκπαίδευσης. Δοθέντος ενός συνόλου εκπαίδευσης N ο απλός αλγόριθμος Μπέυζ εκτιμά την $P(Y = \omega_r)$ για κάθε $r = 1, 2, \dots, m$ μετρώντας πόσο συχνά κάθε κατηγορία, ω_r , εμφανίζεται στα δεδομένα εκπαίδευσης. Δηλαδή:

$$\hat{P}(Y = \omega_r) = \frac{\sum_{i=1}^N \mathbb{1}[y_i = \omega_r]}{N}. \quad (4.15)$$

Επίσης, για τον καθορισμό της $\hat{P}(X^j = x_{test}^j | Y = \omega_r)$ πρέπει για κάθε κατηγορία να μετρηθεί ο αριθμός των εμφανίσεων κάθε τιμής του γνωρίσματος. Άρα είναι:

$$\hat{P}(X^j = x_{test}^j | Y = \omega_r) = \frac{\sum_{i=1}^N \mathbb{1}[x_i^j = x_{test}^j, y_i = \omega_r]}{\sum_{i=1}^N \mathbb{1}[y_i = \omega_r]}. \quad (4.16)$$

Στην σχέση (4.16) είναι πιθανό να δημιουργηθεί το εξής πρόβλημα: Πιθανός μηδενισμός κάποιας δεσμευμένης πιθανότητας εξαιτίας της απουσίας ορισμένων τιμών των χαρακτηριστικών από τα δεδομένα εκπαίδευσης κάποια κλάσης. Δηλαδή αν κανένα από τα παραδείγματα εκπαίδευσης μιας κλάσης δεν περιέχει το γνώρισμα x_{test}^j τότε δεν θα μπορούσαμε να ταξινομήσουμε το x_{test} που περιέχει αυτό το γνώρισμα. Είναι εφικτό να ξεπεράσουμε αυτή την δυσκολία ενσωματώνοντας την τεχνική εξομάλυνσης Laplace (Laplace smoothing) [30][46]. Συγκεκριμένα προσθέτουμε στον αριθμητή μια παράμετρο, $\alpha \geq 1$, γνωστή ως παράμετρος Laplace και στον παρονομαστή το μέγεθος των διαφορετικών τιμών του γνωρίσματος x_j , έστω K . Έτσι, όταν λείπει η τιμή ενός χαρακτηριστικού από τα διαθέσιμα παραδείγματα οι αντίστοιχες πιθανότητες λαμβάνουν μικρές θετικές τιμές αντί να μηδενίζονται. Θέτωντας ως $\alpha = 1$, και εφαρμόζοντας την τεχνική αυτή οι σχέσεις (4.15), (4.16) παίρνουν την μορφή:

$$\hat{P}(x_{test}^j | \omega_r) = \frac{\sum_{i=1}^N \mathbb{1}[x_i^j = x_{test}^j, \omega_i = \omega_r] + 1}{\sum_{i=1}^N \mathbb{1}[\omega_i = \omega_r] + K}.$$

Παράδειγμα

Για να γίνει κατανοητή η διαδικασία που ακολουθεί ο απλός Μπέυζιανός ταξινομητής παρατίθεται ένα παράδειγμα που αφορά την πρόβλεψη σήψης δεδομένου δύο χαρακτηριστικών. Πρόκειται για ένα πρόβλημα δύο κλάσεων (σήψη, όχι σήψη) στον διδιάστατο χώρο με χαρακτηριστικά τον ρυθμό αναπνοής (γρήγορο, αργό) και την ψυχική κατάσταση (διαταραγμένη, φυσιολογική). Με βάση ένα σύνολο εκπαίδευσης δημιουργήθηκε ο Πίνακας πιθανοφάνειας (4.1). Παρατηρούμε ότι η πιθανότητα κάποιος να αναπνέει γρήγορα δεδομένου ότι έχει σήψη είναι $15/20 = 0.75$ ενώ η πιθανότητα να είναι ψυχικά διαταραγμένος δεδομένου ότι δεν έχει σήψη είναι $3/80 = 0.0375$. Χρησιμοποιώντας τον Αφελή Μπέυζιανό ταξινομητή μπορούμε να προβλέψουμε αν ένας ασθενής με αργό ρυθμό αναπνοής (slow RR) και με μη φυσιολογική ψυχική κατάσταση (altered mental

Likelihood	Respiratory rate		Mental status		
	Fast	Slow	Altered	Normal	Total
Sepsis	15/20	5/20	17/20	3/20	20
Non-sepsis	5/80	75/80	3/80	77/80	80
Total	20/100	80/100	20/100	80/100	100

Πίνακας 4.1: Πίνακας πιθανοφάνειας για διάγνωση της σήψης [58].

status) πάσχει από σήψη ή όχι. Σύμφωνα με τον Πίνακα 4.1:

$$\begin{aligned}
 P(\text{sepsis}) &= 0.2, \\
 P(\text{slow } RR|\text{sepsis}) &= 0.25, \\
 P(\text{altered mental status}|\text{sepsis}) &= 0.85, \\
 P(\text{non} - \text{sepsis}) &= 0.8, \\
 P(\text{slow } RR|\text{non} - \text{sepsis}) &= 0.9375, \\
 P(\text{altered mental status}|\text{non} - \text{sepsis}) &= 0.0375.
 \end{aligned}$$

Αξιοποιώντας την σχέση (4.15) έχουμε:

$$\begin{aligned}
 P(\text{sepsis}) \times P(\text{slow } RR|\text{sepsis})(\text{altered mental status}|\text{sepsis}) &= 0.2 \times 0.25 \times 0.85 \\
 &= 0.0425,
 \end{aligned}$$

$$\begin{aligned}
 P(\text{non} - \text{sepsis}) \times P(\text{slow } RR|\text{non} - \text{sepsis}) \times P(\text{altered mental status}|\text{non} - \text{sepsis}) \\
 &= 0.8 \times 0.9375 \times 0.0375 = 0.028125.
 \end{aligned}$$

Αφού $0.0425 > 0.028125$ καταλήγουμε στο συμπέρασμα ότι ο ασθενής πάσχει από σήψη.

Περίπτωση συνεχών τιμών χαρακτηριστικών

Όπως είδαμε στην περίπτωση των διακριτών τιμών εκτιμάται η πιθανότητα για κάθε κλάση τα χαρακτηριστικά του παραδείγματος να λάβουν κάποιες από τις ήδη υπάρχουσες διακριτές τιμές. Στην περίπτωση όπου κάποιο από τα γνωρίσματα x_i παίρνει συνεχείς αριθμητικές τιμές, θεωρούμε ότι οι τιμές αυτές ακολουθούν μια πιθανοτική κατανομή η οποία προσεγγίζεται από τα δεδομένα εκπαίδευσης. Όταν δεν γνωρίζουμε την κατανομή που οι τιμές αυτές ακολουθούν, θεωρούμε χωρίς βλάβη της γενικότητας εξ' ορισμού την Κανονική (Gaussian) κατανομή και ο ταξινομητής ονομάζεται Gaussian Naive Bayes. Οπότε, αν θέλουμε να προβλέψουμε την τιμή ενός νέου στιγμιότυπου, που το χαρακτηριστικό x_j παίρνει συνεχή τιμή z , η πυκνότητα πιθανοφάνειας $p(X^j = z|Y = \omega_r)$ για κάθε κλάση υπολογίζεται ως:

$$p(X^j = z|Y = \omega_r) = \frac{1}{\sqrt{2\pi\sigma_r^2}} \exp\left[-\frac{(z - \mu_r^j)^2}{2\sigma_r^2}\right].$$

Οι παράμετροι $\hat{\mu}_r^j, \hat{\sigma}_r^j$ είναι η μέση τιμή, διασπορά του χαρακτηριστικού j για όλα τα στιγμιότυπα εκπαίδευσης που αφορούν την κλάση ω_r . Μπορούν να υπολογιστούν με την βοήθεια του Εκτιμητή Μέγιστης Πιθανοφάνειας (Maximum Likelihood Estimator) όπου:

$$\hat{\mu}_r^j = \frac{1}{\sum_{i=1}^N \mathbb{1}[y_i = \omega_r]} \sum_{i=1}^N x_i^j \mathbb{1}[y_i = \omega_r],$$

$$\hat{\sigma}_r^j = \frac{1}{\sum_{i=1}^N \mathbb{1}[y_i = \omega_r]} \sum_{i=1}^N (x_i^j - \hat{\mu}_r^j)^2 \mathbb{1}[y_i = \omega_r].$$

Επειδή όμως ο Εκτιμητής Μέγιστης Πιθανοφάνειας για το $\hat{\sigma}_r^j$ είναι μεροληπτικός, υπολογίζεται ο ελάχιστος αμερόληπτος εκτιμητής διασποράς (minimum variance unbiased estimator (MVUE)):

$$\hat{\sigma}_r^j = \frac{1}{(\sum_{i=1}^N \mathbb{1}[y_i = \omega_r] - 1)} \sum_{i=1}^N (x_i^j - \hat{\mu}_r^j)^2 \mathbb{1}[y_i = \omega_r].$$

Επομένως, στην περίπτωση των συνεχών χαρακτηριστικών, ο Αφελής Μπεϋζιανός ταξινομητής διατυπώνεται ως εξής:

$$\mathbf{x} \in \omega_k \text{ αν } \omega_k = \underset{\omega_r, r=1, \dots, m}{\operatorname{argmax}} P(Y = \omega_r) \prod_{j=1}^l \frac{1}{\sqrt{2\pi\hat{\sigma}_r^j}} \exp \left[-\frac{(z - \hat{\mu}_r^j)^2}{2\hat{\sigma}_r^j} \right]. \quad (4.17)$$

Μια εναλλακτική μέθοδος προκειμένου να χειριστούμε τα χαρακτηριστικά με συνεχείς τιμές είναι η διακριτοποίηση τους πριν την εκπαίδευση. Για παράδειγμα, ένα συνεχές γνώρισμα θεωρείται η ηλικία. Δημιουργώντας όμως ηλικιακές ομάδες μετατρέπεται σε διακριτό χαρακτηριστικό.

4.1.6 Μπεϋζιανά δίκτυα

Όπως είδαμε στην προηγούμενη Παράγραφο, ο Αφελής Μπεϋζιανός ταξινομητής είναι κατάλληλος σε προβλήματα μεγάλων διαστάσεων. Κάνει όμως την ακραία υπόθεση της ανεξαρτησίας μεταξύ των χαρακτηριστικών δεδομένης μιας κατηγορίας. Στο σημείο αυτό εισάγουμε την έννοια Μπεϋζιανό δίκτυο, μια μεθοδολογία που θεωρεί την ανεξαρτησία όπου αυτή χρειάζεται βελτιώνοντας έτσι την απόδοση ενός Αφελούς Μπεϋζιανού ταξινομητή. Τα δίκτυα αυτά χαρακτηρίζονται από μαθηματική αυστηρότητα αλλά είναι κατανοητά διαισθητικά. Μέσω αυτών αναπαρίστανται αποτελεσματικά και υπολογίζονται οι από κοινού κατανομές πιθανοτήτων στο σύνολο των τυχαίων μεταβλητών.

Ένα Μπεϋζιανό δίκτυο (Bayesian Network/BN) γνωστό και ως δίκτυο πεποίθησης (belief network) είναι μια μορφή πιθανοτικού γράφου. Πιο συγκεκριμένα, πρόκειται για ένα κατευθυνόμενο ακυκλικό γράφο, (directed acyclic graph-DAG), του οποίου οι κόμβοι αντιπροσωπεύουν τα χαρακτηριστικά και τα τόξα μεταξύ των κόμβων τις στατιστικές σχέσεις εξάρτησης μεταξύ των χαρακτηριστικών και τις τοπικές κατανομές πιθανότητας για κάθε γνώρισμα. Εάν υπάρχει ένα τόξο από τον κόμβο A στον κόμβο B, τότε το χαρακτηριστικό B εξαρτάται από το A και το A καλείται

γονέας του B . Αν το x^i δεν έχει κανένα γονέα τότε η τοπική κατανομή πιθανότητας είναι η περιθώρια (marginal), διαφορετικά είναι η κατανομή πιθανότητας του κόμβου δοθέντων των γονέων. Στην περίπτωση που θα μελετήσουμε στην συνέχεια, όπου τα χαρακτηριστικά παίρνουν διακριτές τιμές, μελετούμε τις περιθώριες και τις υπό συνθήκη συναρτήσεις μάζας πιθανότητας δοθέντων των γονέων. Αυτές, όπως θα δούμε στο επόμενο παράδειγμα, συχνά αναπαρίστανται από ένα πίνακα, όπου παρατίθεται η τοπική πιθανότητα που ένας κόμβος-παιδί παίρνει σε κάθε μια από τις αποδεκτές τιμές, για κάθε συνδιασμό των τιμών των γονέων του.

Από τον κανόνα αλυσίδας της πιθανότητας, η κοινή συνάρτηση μάζας πιθανότητας των χαρακτηριστικών δίνεται από την σχέση (4.18) που ισχύει πάντα [23][5][51]:

$$\begin{aligned} P(X^1 = x^1, \dots, X^l = x^l) &= P(X^1 = x^1)P(X^2 = x^2|X^1 = x^1).. \quad (4.18) \\ &..P(X^l = x^l|X^1 = x^1, \dots, X^{l-1} = x^{l-1}) \\ &= \prod_{i=1}^l P(X^i = x^i|X^1 = x^1, \dots, X^{i-1} = x^{i-1}). \end{aligned}$$

Λόγω του ότι κάθε χαρακτηριστικό $x^i, i = 1, \dots, l$, εξαρτάται μόνο από το σύνολο των μεταβλητών των γονέων ($Parents(x^i)$) η 4.18 απλοποιείται ως:

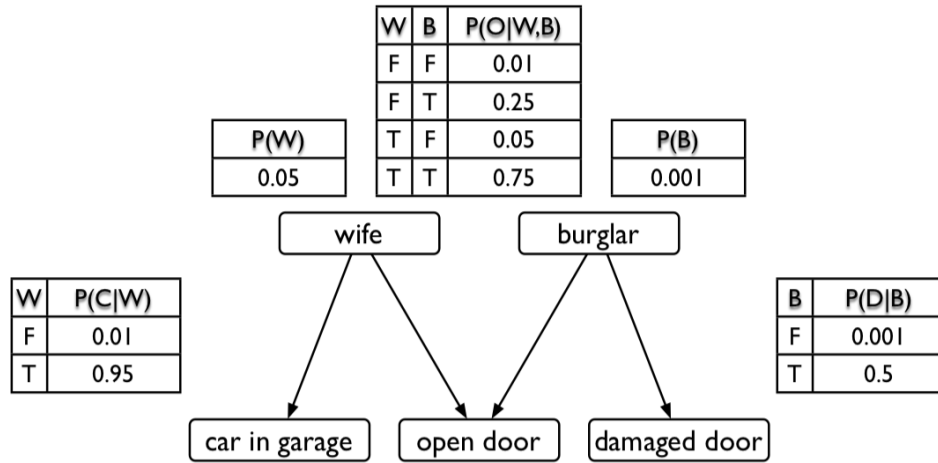
$$P(X^1 = x^1, \dots, X^l = x^l) = \prod_{i=1}^l P(X^i = x^i|Parents(x^i)) \quad (4.19)$$

όπου $Parents(x^i) \subseteq (x^1, \dots, x^{i-1})$.

Παρατηρείται ότι το πρόβλημα της εκτίμησης της από κοινού σ.μ.π. έχει αναλυθεί σε ένα γινόμενο απλούστερων παραγόντων, κάθε ένας από αυτούς περιλαμβάνει μικρότερο αριθμό χαρακτηριστικών σε σχέση με τον αρχικό αριθμό.

Προκειμένου να προσδιορίσουμε πλήρως ένα Μπεϋζιανό δίκτυο χρειαζόμαστε τις περιθώριες σ.μ.π. των χαρακτηριστικών που δεν έχουν γονέα και τις υπο συνθήκη σ.μ.π. των χαρακτηριστικών δοθέντων των γονέων τους για όλους τους πιθανούς συνδιασμούς των τιμών τους. Επίσης σημαντική είναι η τοπολογική ταξινόμηση των γνωρισμάτων δηλαδή η διάταξη αυτών με τρόπο τέτοιο ώστε καθένα να βρίσκεται πριν από τους απόγονους της στο γράφο. Αν αυτή δεν γίνει σωστά τότε δομή που θα αποκτήσει το δίκτυο θα αποτυχεί να εμφανίσει αρκετές υπό όρους ανεξαρτησίες μεταξύ των γνωρισμάτων. Έτσι, για να έχει επιτυχία ένα Μπεϋζιανό δίκτυο και να έχουμε μια δομή που θα ικανοποιεί την σχέση (4.19) σχεδιάζουμε τα βέλη από την αιτία στο άμεσο αποτέλεσμα.

Παρακάτω δίνεται ένα παράδειγμα με σκοπό να γίνει κατανοητή η λειτουργία ενός Μπεϋζιανού δικτύου. Έστω ότι γυρνώντας στο σπίτι ένας άντρας βλέπει την πόρτα ανοιχτή και θέλει να εξακριβώσει αν το σπίτι διαρρήχτηκε ή η γυναίκα του την ξέχασε ανοιχτή. Πιθανές μεταβλητές είναι: Ανοιχτή πόρτα (O), σύζυγος (W), διαρρήκτης (B), αυτοκίνητο στο γκαράζ (C), χαλασμένη πόρτα (D). Αν παρατηρηθεί το αυτοκίνητο στο γκαράζ τότε πιθανώς η σύζυγος ξέχασε την πόρτα ανοιχτή, ενώ αν η πόρτα είναι χαλασμένη τότε μάλλον πρόκειται για διαρρήκτη. Ας υποθέσουμε ότι κάποιος θέλει να υπολογίσει την πιθανότητα η πόρτα να είναι ανοιχτή, η σύζυγος να ευθύνεται γι' αυτό και όχι ο διαρρήκτης, το αυτοκίνητο να είναι στο γκαράζ και η πόρτα



Διάγραμμα 4.6: Παράδειγμα Μπεϋζιανού Δικτύου.

να μην είναι χαλασμένη. Αυτή μπορεί να υπολογιστεί ως το γινόμενο όλων των υπο συνθήκη πιθανοτήτων που ορίζουν το δίκτυο. Χρησιμοποιώντας λοιπόν την μορφή του Μπεϋζιανού Δικτύου (Διάγραμμα 4.6) είναι:

$$\begin{aligned}
 P(o, w, \neg b, c, \neg d) &= P(o|w, \neg b, c, \neg d)P(w, \neg b, c, \neg d) & (4.20) \\
 &= P(o|w, \neg b)P(w, \neg b, c, \neg d) \\
 &= P(o|w, \neg b)P(c|w, \neg b, \neg d)P(w, \neg b, \neg d) \\
 &= P(o|w, \neg b)P(c|w)P(w, \neg b, \neg d) \\
 &= P(o|w, \neg b)P(c|w)P(\neg d|w, \neg b)P(w, \neg b) \\
 &= P(o|w, \neg b)P(c|w)P(\neg d|\neg b)P(w, \neg b) \\
 &= P(o|w, \neg b)P(c|w)P(\neg d|\neg b)P(w)P(\neg b).
 \end{aligned}$$

Αντικαθιστώντας τις αντίστοιχες τιμές είναι:

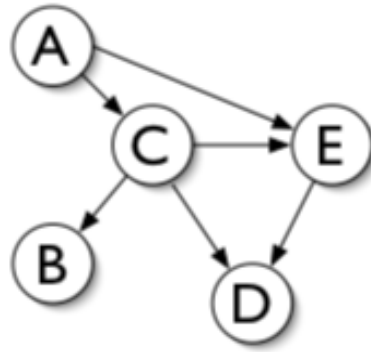
$$\begin{aligned}
 P(o, w, \neg b, c, \neg d) &= P(o|w, \neg b)P(c|w)P(\neg d|\neg b)P(w)P(\neg b) \\
 &= 0.05 \times 0.95 \times 0.999 \times 0.05 \times 0.999 \\
 &= 0.0024.
 \end{aligned}$$

Σε μια γενικότερη περίπτωση όπως ο γράφος του Διαγράμματος 4.7 η από κοινού πιθανότητα των μεταβλητών δίνεται από τον τύπο:

$$P(A, B, C, D, E) = P(A)P(B|C)P(C|A)P(D|C, E)P(E|A, C). \quad (4.21)$$

Πλέον μπορούμε να υπολογίσουμε οποιαδήποτε πιθανότητα μας ενδιαφέρει. Για παράδειγμα, η $P(A|D = d)$ υπολογίζεται ως:

$$\begin{aligned}
 P(A|D = d) &= \frac{P(A, D = d)}{P(D = d)} & (4.22) \\
 &= \frac{P(A, D = d)}{\sum_A P(A, D = d)}.
 \end{aligned}$$



Διάγραμμα 4.7: Παράδειγμα Μπεϋζιανού γράφου.

Για να λάβουμε την $P(A, D = d)$ πρέπει να περιθωριοποιήσουμε την από κοινού πιθανότητα για όλες τις δυνατές τιμές των B, C, E , δηλαδή:

$$P(A, D = d) = \sum_{B, C, E} P(A, B, C, E, D = d). \quad (4.23)$$

Έστω ότι κάθε μια από τις διακριτές τιμές μπορεί να πάρει L τιμές, τότε θα πρέπει να γίνουν L^3 πράξεις. Για πολλές μεταβλητές και ένα μεγάλο πλήθος τιμών L , ο αριθμός των υπολογισμών που χρειάζονται να γίνουν είναι υπερβολικά μεγάλος. Προκειμένου να μειώσουμε την πολυπλοκότητα των υπολογισμών αξιοποιούμε την δομή του Μπεϋζιανού Δικτύου (Διάγραμμα 4.7) και χρησιμοποιώντας τον κανόνα αλυσίδας για πιθανότητες έχουμε:

$$\begin{aligned} \sum_{B, C, E} P(A, B, C, E, D = d) &= \sum_{B, C, E} P(A)P(B|C)P(C|A)P(D = d|C, E)P(E|A, C) \\ &= P(A) \sum_C P(C|A) \sum_B P(B|C) \sum_E P(E|A, C)P(D = d|C, E). \end{aligned} \quad (4.24)$$

Είναι φανερό πως χρησιμοποιώντας τις υπό όρους ανεξαρτησίες που είναι κωδικοποιημένες στο δίκτυο Μπέυζ μειώνεται σημαντικά το υπολογιστικό φορτίο καθώς για να παρθεί μια απόφαση δεν είναι αναγκαίο να υπολογιστούν όλες οι από κοινού πιθανότητες καθώς τα ξενά 'κλαδιά' μπορούν να αγνοηθούν. Τα Δίκτυα αυτά είναι πολύ χρήσιμα καθώς μπορούν να διαβαστούν λεπτομέρειες του μοντέλου απευθείας από τον γράφο. Βέβαια αδυνατούν να αποτυπώσουν όλες τις δυνατές σχέσεις μεταξύ των μεταβλητών, είναι όμως ιδανικά για την αναπαράσταση αίτιου - αποτελέσματος.

Εκπαίδευση

Η εκπαίδευση ενός Μπεϋζιανού δικτύου αποτελείται από δύο μέρη. Το πρώτο αφορά την κατασκευή του δικτύου. Συχνά, έχουμε ισχυρούς λόγους να πιστεύουμε ότι ένα γεγονός επηρεάζει ένα άλλο ή ενδείξεις ότι δυο γεγονότα είναι ανεξάρτητα μεταξύ τους.

Συνδιάζοντας αυτή την εκ των προτέρων γνώση μας και τις πληροφορίες που παρέχονται από κάποιον ειδικό στο πρόβλημα που αντιμετωπίζουμε είναι δυνατόν να καθορίσουμε την τοπολογία του Δικτύου [51]. Το δεύτερο μέρος στην εκπαίδευση αποτελεί ο υπολογισμός των άγνωστων παραμέτρων, δηλαδή οι υπο συνθήκη και οι περιθώριες πιθανότητες από τα διαθέσιμα δεδομένα εκπαίδευσης. Ένας απλός τρόπος εκμάθησης είναι ο υπολογισμός πιθανοτήτων ως το κλάσμα του πλήθους των περιπτώσεων που συμβαίνει ένα γεγονός προς το συνολικό πλήθος των δοκιμών που εκτελούνται.

4.2 Ταξινομητές Πλησιέστερου Γείτονα

4.2.1 Εισαγωγή

Ο ταξινομητής Πλησιέστερου Γείτονα είναι από τους πιο διαδεδομένους αλγόριθμους μάθησης βασισμένη σε στιγμιότυπα. Δεν εμπίπτει στο Μπεϋζιανό πλαίσιο, παρόλο αυτά ταιριάζει όμαρφα σε αυτό το σημείο και μπορεί να θεωρηθεί ως μια γέφυρα με το κεφάλαιο 5 που ακολουθεί. Η μάθηση που εφαρμόζεται στην μέθοδο των Πλησιέστερων Γειτόνων είναι γνωστή και ως οκνηρή (lazy learning) καθώς ο αλγόριθμος δεν ‘εκπαιδεύεται’ παραμόνο αποθηκεύει στην μνήμη του τις παρατηρήσεις του συνόλου εκπαίδευσης. Η κατάταξη ενός νέου στιγμιότυπου γίνεται ανάλογα με το πόσο αυτό ‘μοιάζει’ με τα ήδη ταξινομημένα στιγμιότυπα. Κάθε φορά λοιπόν που πρέπει να γίνει μια νέα κατηγοριοποίηση ανακαλείται από την μνήμη του αλγορίθμου ένα σύνολο από σχετιζόμενα, με το προς ταξινόμηση στιγμιότυπο, παραδείγματα. Οι ταξινομητές που βασίζονται στην μάθηση αυτή θεωρούνται αργοί στην φάση της ταξινόμησης αφού δεν έχουν κατασκευάσει ένα μοντέλο γενίκευσης ώστε να γίνεται άμεσα η κατηγοριοποίηση μιας άγνωστης παρατήρησης. Το όνομα τους οφείλεται στο ότι η ταξινόμηση γίνεται με βάση την κλάση των πιο κοντινών παραδειγμάτων (γείτονες), δηλαδή στην κατηγορία από την οποία απέχει λιγότερο. Τέλος, όπως θα δείξουμε παρακάτω, ο κανόνας KNN τείνει στον Μπεϋζιανό ταξινομητή κάτω από ορισμένες συνθήκες.

4.2.2 Ταξινομητής K Πλησιέστερων Γειτόνων

Ο αλγόριθμος K Πλησιέστερων Γειτόνων (K Nearest Neighbor/ KNN) βασίζεται στην παραδοχή ότι τα παραδείγματα κάθε κλάσης έχουν κοντινές τιμές στα χαρακτηριστικά τους και έτσι το άγνωστο παράδειγμα κατηγοριοποιείται στην κλάση που ανήκει η πλειοψηφία των K γειτόνων. Στην υλοποίηση του αλγορίθμου συμμετέχουν ένα σύνολο ταξινομημένων στιγμιότυπων (υποψήφιοι γείτονες), μια μετρική απόστασης η οποία καθορίζει το πόσο απέχουν οι παρατηρήσεις μεταξύ τους και η παράμετρος K που αντιπροσωπεύει τον αριθμό των γειτόνων.

Πιο συγκεκριμένα, δοθέντος ενός συνόλου εκπαίδευσης $S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$ όπου $\mathbf{x}_i \in \mathbb{R}^l$, $y_i \in Y = (\omega_1, \dots, \omega_m)$, m, l το πλήθος των κλάσεων, χαρακτηριστικών αντίστοιχα, και ενός παραδείγματος προς ταξινόμηση \mathbf{x}_{new} τότε ο προσδιορισμός

της κλάσης y_{new} γίνεται ως εξής: Ο χρήστης επιλέγει μια προσωρινή τιμή για το K όπως επίσης και μια μετρική. Με βάση την τελευταία, υπολογίζει την απόσταση, $dist(\mathbf{x}_{new}, \mathbf{x}_i) \forall i = 1, \dots, N$, του παραδείγματος (διάνυσμα) προς ταξινόμησης από κάθε παράδειγμα του σώματος εκπαίδευσης S . Από τις αποστάσεις αυτές δημιουργείται ένα σύνολο $B_{\mathbf{x}_{new}}$, υποσύνολο του S , που αποτελείται από τους K Πλησιέστερους Γείτονες που σημείωσαν την μικρότερη απόσταση από το \mathbf{x}_{new} [12]. Τελικά, η προβλεπόμενη κατηγορία y_{new} είναι αυτή που ανήκει η πλειοψηφία των παρατηρήσεων του συνόλου $B_{\mathbf{x}_{new}}$. Η απλούστερη μορφή του αλγορίθμου είναι για $K = 1$, γνωστή ως Κανόνας Πλησιέστερου Γείτονα (Nearest Neighbor Rule), όπου το διάνυσμα χαρακτηριστικών κατατάσσεται στην κλάση του κοντινότερου γείτονα.

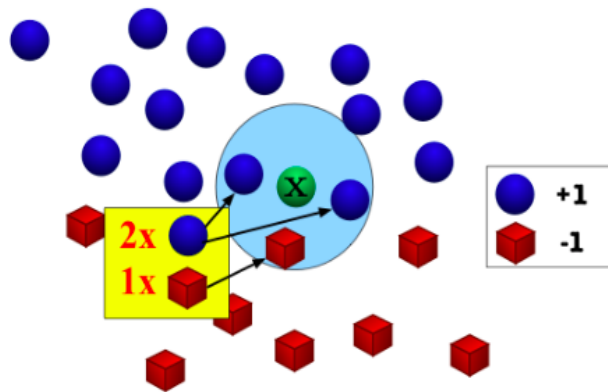
Τυπικά, το σύνολο $B_{\mathbf{x}_{new}}$ ορίζεται ως: $B_{\mathbf{x}_{new}} \subseteq S : |B_{\mathbf{x}_{new}}| = k$ και $\forall (\mathbf{x}', y') \in S \setminus B_{\mathbf{x}_{new}}$ ισχύει:

$$dist(\mathbf{x}_{new}, \mathbf{x}') \geq \max_{(\mathbf{x}'', y'') \in B_{\mathbf{x}_{new}}} dist(\mathbf{x}_{new}, \mathbf{x}'').$$

Δηλαδή το \mathbf{x}_{new} απέχει από οποιοδήποτε σημείο που ανήκει στο σύνολο εκπαίδευσης και όχι στο $B_{\mathbf{x}_{new}}$ απόσταση ίση ή μεγαλύτερη από ότι απέχει από το πιο απομακρυσμένο σημείο του $B_{\mathbf{x}_{new}}$ [12]. Με άλλα λόγια, όταν είμαστε στον Ευκλείδειο χώρο δημιουργείται μια σφαίρα με κέντρο το \mathbf{x}_{new} και ακτίνα $R = \max_{(\mathbf{x}'', y'') \in B_{\mathbf{x}_{new}}} dist(\mathbf{x}_{new}, \mathbf{x}'')$ που περιέχει τους K κοντινούς γείτονες. Μπορούμε λοιπόν να ορίσουμε έναν ταξινομητή $h(\cdot)$ σαν μια συνάρτηση που επιστρέφει την περισσότερες φορές εμφανιζόμενη κατηγορία στο $B_{\mathbf{x}_{new}}$:

$$h(\mathbf{x}_{new}) = mode(\{y'' : (\mathbf{x}'', y'') \in B_{\mathbf{x}_{new}}\}),$$

όπου $mode(\cdot)$ επιστρέφει την παραπάνω ζητούμενη κλάση. Το σκεπτικό αυτό απεικονίζεται στο Διάγραμμα 4.8 όπου θέλουμε να προβλέψουμε την κλάση, (+1,-1), της άγνωστης παρατήρησης (πράσινο στοιχείο). Αν επιλέξουμε $K = 3$ τότε οι ετικέτες των τριών αυτών γειτόνων είναι $2 \times (+1)$ και $1 \times (-1)$. Αφού πλειοψηφεί η κατηγορία +1 το νέο στιγμιότυπο ταξινομείται σε αυτή.



Διάγραμμα 4.8: Επιλογή γειτόνων για την ταξινόμηση της άγνωστης πράσινης παρατήρησης.

Σύνδεση με τον βέλτιστο Μπεϋζιανό ταξινομητή

Αν υποθέσουμε ότι γνωρίζουμε την κατανομή $P(Y = \omega_i | \mathbf{X} = \mathbf{x})$ τότε σύμφωνα με τον ταξινομητή κατά Μπέυζ η πιο πιθανή κλάση για το \mathbf{x}_{new} είναι:

$$y^* = h_{opt}(\mathbf{x}_{new}) = \underset{\omega_i}{\operatorname{argmax}} P(Y = \omega_i | \mathbf{X} = \mathbf{x}_{new}),$$

η οποία όπως έχουμε δείξει, ελαχιστοποιεί την πιθανότητα λανθασμένης ταξινόμησης, $P(error) = 1 - P(Y = y^* | \mathbf{X} = \mathbf{x})$. Πρακτικά, ο Μπεϋζιανός ταξινομητής δεν μπορεί να εφαρμοστεί, οι πιθανότητες εσφαλμένης κατηγοριοποίησης αποτελούν όμως ένα σημαντικό θεωρητικό ελάχιστο κάτω φράγμα για τις αντίστοιχες πιθανότητες άλλων ταξινομητών [4][12]. Πιο συγκεκριμένα, ο KNN προσεγγίζει την απόδοση του βέλτιστου Μπεϋζιανού ταξινομητή ενώ αυτή του $1NN$ είναι μόλις δύο φορές χειρότερη ($N \rightarrow \infty$).

Θεώρημα 4.2.1. Καθώς $N \rightarrow \infty$ η πιθανότητα σφάλματος ταξινόμησης, P_{1NN} , είναι το πολύ διπλάσια της πιθανότητας σφάλματος του Μπεϋζιανού ταξινομητή, P_B , ($P_{1NN} \leq 2P_B$).

Απόδειξη. Έστω ότι \mathbf{x}_{1NN} είναι ο ένας κοντινότερος γείτονας ενός \mathbf{x}_t του οποίου θέλουμε να προβλέψουμε την ετικέτα. Τότε καθώς $N \rightarrow \infty$, $dist(\mathbf{x}_{1NN}, \mathbf{x}_t) \rightarrow 0$, δηλαδή $\mathbf{x}_{1NN} \rightarrow \mathbf{x}_t$. Αυτό σημαίνει ότι για πολύ μεγάλο πλήθος παραδειγμάτων N , ο κοντινότερος γείτονας ταυτίζεται με το \mathbf{x}_t και έτσι η προβλεπόμενη κατηγορία για το \mathbf{x}_t είναι αυτή του \mathbf{x}_{1NN} . Στην συνέχεια, υπολογίζουμε την πιθανότητα λανθασμένης ταξινόμησης ως:

$$\begin{aligned} P_{1NN} &= P(Y = y^* | \mathbf{X} = \mathbf{x}_t) (1 - P(Y = y^* | \mathbf{X} = \mathbf{x}_{1NN})) \\ &\quad + P(Y = y^* | \mathbf{X} = \mathbf{x}_{1NN}) (1 - P(Y = y^* | \mathbf{X} = \mathbf{x}_t)) \\ &\leq (1 - P(Y = y^* | \mathbf{X} = \mathbf{x}_{1NN})) (1 - P(Y = y^* | \mathbf{X} = \mathbf{x}_t)) \\ &= 2P_B, \end{aligned}$$

όπου χρησιμοποιήσαμε $P(Y = y^* | \mathbf{X} = \mathbf{x}_t) \leq 1$, $P(Y = y^* | \mathbf{X} = \mathbf{x}_{1NN}) \leq 1$ και $P(Y = y^* | \mathbf{X} = \mathbf{x}_{1NN}) = P(Y = y^* | \mathbf{X} = \mathbf{x}_t)$. \square

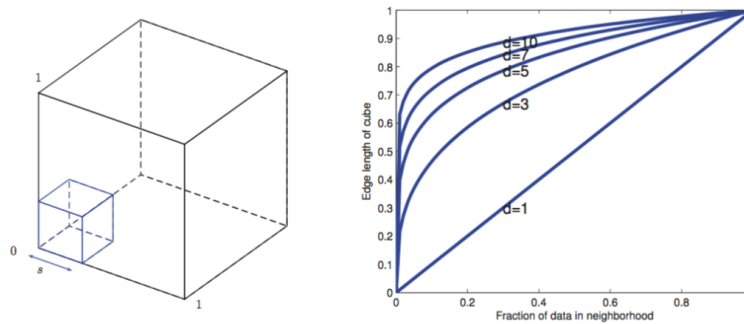
Ανάλογα με τον $1NN$ ταξινομητή, για τον KNN σε πρόβλημα δύο κλάσεων καθώς $N \rightarrow \infty$ ισχύει:

$$P_B \leq P_{KNN} \leq P_B + \sqrt{\frac{2P_{1NN}}{K}}.$$

Συμπεραίνουμε ότι ο KNN ταξινομητής για πολύ μεγάλο N είναι αρκετά ακριβής αλλά ταυτόχρονα πολύ αργός. Γενικά, είναι από τους πιο αποδοτικούς αλγορίθμους όταν εφαρμόζεται σε προβλήματα λίγων διαστάσεων. Η απλότητα του είναι το κυριότερο του πλεονέκτημα. Σε περιπτώσεις όμως πολλών διαστάσεων ο KNN έχει φτωχά αποτελέσματα.

Διαστατικότητα και KNN ταξινομητής

Όπως είδαμε, ο KNN στηρίζεται στην υπόθεση πως οι τιμές των χαρακτηριστικών των παραδειγμάτων που ανήκουν στην ίδια κλάση είναι κοντινές, γεγονός που δεν ισχύει σε προβλήματα μεγάλων διαστάσεων [8][20][34]. Για να γίνει αυτό κατανοητό παραθέτουμε το εξής παράδειγμα:



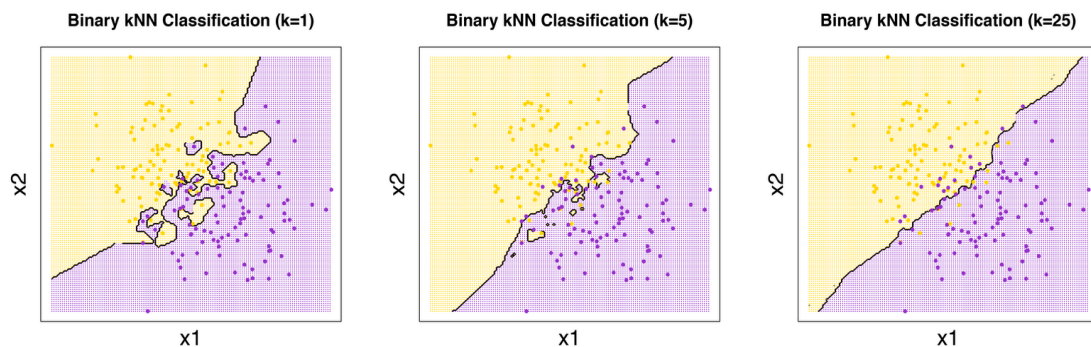
Διάγραμμα 4.9: Απεικόνιση του προβλήματος πολλών διαστάσεων. Στο αριστερό σχήμα έχουμε ενσωματώσει ένα κύβο ακμής s σε ένα μεγαλύτερο ακμής 1. Στο δεξί Διάγραμμα φαίνεται πως αυξάνεται η ακμή του μικρού κύβου καθώς η διάσταση d αυξάνεται [20].

Ας υποθέσουμε έναν κύβο $[0, 1]^d$ με τα δεδομένα εκπαίδευσης \mathbf{x}_i ομοιόμορφα κατανομημένα σε αυτόν, δηλαδή $\forall i, \mathbf{x}_i \in [0, 1]^d$. Έστω ότι θέλουμε να προβλέψουμε την ετικέτα ενός στοιχείου \mathbf{x}_{new} με βάση τους 10 κοντινότερους γείτονες. Θα συμβολίσουμε με s την ακμή του κύβου που περιέχει τους $k = 10$ πλησιέστερους γείτονες [12][20][8]. Παρατηρούμε ότι $s^d \simeq \frac{k}{N}$ δηλαδή $s \simeq \frac{k}{N}^{\frac{1}{d}}$. Για $N = 1000$ και $d = 1000$ τότε $s = 0.9954$. Δηλαδή αν $d \gg 0$ τότε τα 10 κοντινότερα στιμιότυπα είναι διασκορπισμένα σχεδόν σε όλο τον χώρο με αποτέλεσμα το μέγεθος της ‘γειτονιάς’ να έχει γίνει πολύ μεγάλο και οποιαδήποτε κατάταξη να είναι πιθανότατα λανθασμένη καθώς χάνεται η έννοια ‘ομοιότητας’ των παραδειγμάτων (Διάγραμμα 4.9).

Επιλογή της παραμέτρου K

Η επιλογή του αριθμού των πλησιέστερων γειτόνων K είναι καθοριστική για την καλή λειτουργία της μεθόδου και τον βαθμό ποιότητας των προβλέψεων [20][6][34]. Γενικά,

αποφεύγεται η τιμή αυτή να είναι πολλαπλάσιο των κλάσεων m (αποφυγή ισοπαλιών) και σε προβλήματα ταξινόμησης δύο κλάσεων επιλέγεται να είναι περιττή. Επίσης η τιμή K μπορεί να θεωρηθεί σαν παράμετρος ομαλότητας (smoothing parameter). Η επιλογή μικρής τιμής έχει σαν αποτέλεσμα ένα μοντέλο με μεγάλη διακύμανση στις προβλέψεις, δημιουργούνται πολλές μικρές περιοχές για κάθε κλάση και οδηγεί στο πρόβλημα της υπερπροσαρμοστικότητας, όπως φαίνεται και στο Διάγραμμα 5.3 για $K = 1$. Από την άλλη μεριά, πολύ μεγάλη τιμή του K , οδηγεί σε ένα μοντέλο με υψηλή μεροληψία, παρατηρούνται ομαλότερες περιοχές απόφασης (smoother boundaries) και μπορεί να οδηγεί σε πρόβλημα υποπροσαρμογής (Διάγραμμα 4.10 για $K=25$). Γίνεται λοιπόν αντιληπτό, ότι η τιμή της παραμέτρου K θα πρέπει να είναι τόσο μεγάλη ώστε να ελαχιστοποιεί την πιθανότητα λάθους ταξινόμησης και τόσο μικρή ώστε οι K κοντινές παρατηρήσεις να είναι αντιπροσωπευτικές του άγνωστου σημείου. Έτσι, υπάρχει μια βέλτιστη ποσότητα K η οποία ασισταθμίζει την μεροληψία και την διακύμανση του μοντέλου.



Διάγραμμα 4.10: Περιοχές κλάσεων για τρεις διαφορετικές περιοχές του K σε διδιάστατο πρόβλημα.

Εφαρμόζοντας την τεχνική cross validation μπορούμε να πετύχουμε μια αποδοτική τιμή για την ελεύθερη παράμετρο K . Επιλέγουμε μια αρχική τιμή για το K , χρησιμοποιούμε τις $N - 1$ παρατηρήσεις και κάνουμε, με βάση την πλειοψηφία της κλάσης των K γειτόνων, πρόβλεψη για την n -οστή παρατήρηση και υπολογίζουμε κάθε φορά το σφάλμα. Επαναλαμβάνουμε N φορές και υπολογίζουμε το μέσο σφάλμα. Εφαρμόζοντας την ίδια διαδικασία για διαφορετικές τιμές του K , επιλέγουμε εκείνο με το μικρότερο μέσο σφάλμα.

Μετρικές απόστασης

Προκειμένου να υλοποιηθεί ο αλγόριθμος πρέπει να ορίσουμε κάποια μετρική απόστασης μεταξύ των παρατηρήσεων. Σύμφωνα με τον μαθηματικό ορισμό, μια μετρική πρέπει να εκπληρώνει τις βασικές τέσσερις ιδιότητες. Για οποιαδήποτε στιγμιότυπα a, b, c πρέπει να ισχύει:

- ✓ $d(a, b) \geq 0$ (Θετικότητα),
- ✓ $d(a, b) = 0$ ανν $a = b$ (Αντανεκλαστικότητα),

$\checkmark d(a, b) = d(b, a)$ (Συμμετρικότητα),
 $\checkmark d(a, b) + d(b, c) \geq d(a, c)$ (Τριγωνική ανισότητα).

Μια γνωστή μετρική που μπορεί να χρησιμοποιηθεί είναι η απόσταση Minkowski. Ανάμεσα στην νέα παρατήρηση \mathbf{x}_{new} και την ήδη ταξινομημένη παρατήρηση \mathbf{x}_i η απόσταση αυτή ορίζεται ως:

$$d(\mathbf{x}_{new}, \mathbf{x}_i) = \left(\sum_{j=1}^l |x_{new}^j - x_i^j|^p \right)^{\frac{1}{p}},$$

όπου l το πλήθος των χαρακτηριστικών. Αν $p = 2$ παίρνουμε την γνωστή Ευκλείδεια απόσταση που είναι η συνηθέστερη σε περίπτωση συνεχών χαρακτηριστικών. Όταν τα γνωρίσματα παίρνουν διακριτές τιμές χρησιμοποιούμε την απόσταση Hamming, όπου $d(\mathbf{x}_{new}, \mathbf{x}_i) = \sum_{j=1}^l |x_{new}^j - x_i^j|$. Αν $x_{new}^j \neq x_i^j$ τότε $x_{new}^j - x_i^j = 1$ διαφορετικά είναι 0.

Μέχρι στιγμής, έχουμε προσεγγίσει το πρόβλημα της ταξινόμησης χρησιμοποιώντας πιθανοτικές έννοιες κατά Μπέυζ με σκοπό να ελαχιστοποιήσουμε την πιθανότητα εσφαλμένης ταξινόμησης. Είδαμε πως το μοντέλο του Μπεϋζιανού ταξινομητή μπορεί να ταξινομήσει τα στιγμιότυπα με την μεγαλύτερη ακρίβεια από κάθε άλλη μέθοδο. Ένα σημαντικό πρόβλημα που συνδέεται με τον ταξινομητή αυτόν και κατά συνέπεια με τις δύο μορφές που αυτός παίρνει (LDA, QDA) είναι το μεγάλο πλήθος των παραμέτρων που πρέπει να εκτιμηθούν, ο υπολογισμός των πολλών πιθανοτήτων και η απαίτηση μεγάλου αριθμού σημείων εκπαίδευσης. Ωστόσο, αναλύσαμε περισσότερο πρακτικές προσεγγίσεις όπως αυτή του Αφελούς Μπεϋζιανού ταξινομητή και των Μπεϋζιανών Δικτύων. Επίσης, κάναμε λόγο για τον ταξινομητή των K πλησιέστερων γειτόνων ο οποίος πετυχαίνει αξιοθαύμαστα αποτελέσματα, προσεγγίζοντας αυτά του Μπεϋζιανού ταξινομητή. Αυτές οι προσεγγίσεις όμως δεν ενδείκνυνται για όλα τα προβλήματα. Υπάρχουν, περιπτώσεις που είναι προτιμότερο να υπολογίσουμε τις επιφάνειες απόφασης χρησιμοποιώντας εναλλακτικές συναρτήσεις κόστους. Με το θέμα αυτό θα ασχοληθούμε στο επόμενο κεφάλαιο, αναλύοντας αλγορίθμους που δημιουργούν συναρτήσεις διάκρισης και απόφασης χωρίς να έχουν κατ' ανάγκη σχέση με την Μπεϋζιανή ταξινόμηση και γενικά είναι υποβέλτιστες σε σχέση με τους κατά Μπέυζ ταξινομητές.

Κεφάλαιο 5

Αλγοριθμικές τεχνικές ταξινόμησης

5.1 Δέντρα Ταξινόμησης

5.1.1 Εισαγωγή

Μια ευρέως χρησιμοποιημένη μέθοδος που επιλύει το πρόβλημα της κατηγοριοποίησης είναι η εκμάθηση με Δέντρα Απόφασης (Decision Trees). Στην προκειμένη περίπτωση, αναφερόμαστε ειδικά σε Δέντρα ταξινόμησης αφού η έξοδος του μοντέλου είναι διακριτή. Τα αποτελέσματα των αλγορίθμων που βασίζονται σε μια τέτοια διαδικασία μάθησης έχουν δενδρική μορφή, δηλαδή αναπαριστούν διαγραμματικά τα δεδομένα.

Πιο συγκεκριμένα, με τον όρο δενδρική δομή αναφερόμαστε σε ένα σύνολο κανόνων αν-τότε που συνδέονται μεταξύ τους από την ρίζα του δέντρου ως τα φύλλα. Οι αποφάσεις που εξάγονται βασίζονται στον έλεγχο των κανόνων αυτών (classification rules). Οι κόμβοι του δέντρου χωρίζονται στους εσωτερικούς, οι οποίοι περιέχουν και την ρίζα, και τους τερμαματικούς κόμβους (φύλλα). Οι εσωτερικοί αντιπροσωπεύουν τα ονόματα των χαρακτηριστικών, οι ακμές μεταξύ των κόμβων παίρνουν τις δυνατές τιμές των γνωρισμάτων και τα φύλλα αναφέρονται στις διάφορες κατηγορίες. Μια νέα παρατήρηση εισέρχεται στο δέντρο από τον κόμβο της ρίζας όπου ελέγχεται η τιμή του γνωρίσματος ώστε να καθοριστεί σε ποιο παιδί κόμβο θα ακολουθήσει ο επόμενος έλεγχος. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να καταλήξουμε σε κάποιο φύλλο-κλάση στην οποία γίνεται και η κατηγοριοποίηση της παρατήρησης. Ισοδύναμα, ένα Δέντρο Απόφασης μπορεί να θεωρηθεί ως μια διαμέριση του χώρου των χαρακτηριστικών \mathbb{R}^l σε m υποπεριοχές, $\{R_1, \dots, R_m\}$, όπου κάθε $R_j \subset \mathbb{R}^l$ περιέχει παραδείγματα που ανήκουν στην ίδια κλάση.

5.1.2 Βασική μεθοδολογία

Η εκπαίδευση του αλγορίθμου αναφέρεται στην διαδικασία οικοδόμησης του δέντρου χρησιμοποιώντας τα ήδη ταξινομημένα παραδείγματα (σετ εκπαίδευσης). Πολλοί αλγόριθμοι έχουν σχεδιαστεί για το πρόβλημα εκμάθησης ενός Δέντρου Απόφασης. Κύριοι εκπρόσωποι αυτών είναι ο ID3 και η βελτιωμένη έκδοση του C4.5, οι οποίοι έχουν κοινή την παρακάτω βασική μεθοδολογία.

Ο αλγόριθμος χτίζει το δέντρο από την κορυφή προς τα κάτω, δηλαδή αρχικά αποφασίζει ποιο γνώρισμα θα βρίσκεται στην ρίζα και κατευθύνεται προς τα φύλλα. Για κάθε κόμβο σκοπός είναι να επιλεγεί από τα διαθέσιμα χαρακτηριστικά εκείνο που διαχωρίζει με τον καλύτερο τρόπο τα δεδομένα εκπαίδευσης, ώστε να μειώνεται το βάθος του δέντρου. Αφού επιλεγεί το χαρακτηριστικό, τότε δημιουργείται μια ακμή προς νέους κόμβους που θα αντιστοιχούν στις διαφορετικές τιμές του χαρακτηριστικού αυτού. Σε κάθε ένα από τους νέους αυτούς κόμβους ενσωματώνονται τα στιγμιότυπα που έχουν την αντίστοιχη τιμή του γνωρίσματος. Η διαδικασία επαναλαμβάνεται για κάθε νέο κόμβο που προκύπτει και τερματίζει όταν τα παραδείγματα του τρέχοντος κόμβου έχουν την ίδια ετικέτα (στην περίπτωση αυτή δημιουργείται ένας κόμβος απόφασης που δηλώνει την κοινή ετικέτα) ή όταν δεν υπάρχουν δεδομένα εκπαίδευσης ή περαιτέρω χαρακτηριστικά να ελεγχθούν (όταν συμβαίνει αυτό δημιουργείται κόμβος απόφασης με τιμή την κατηγορία που εμφανίζεται περισσότερο στα δεδομένα εκπαίδευσης).

Υποθέτοντας την ύπαρξη ενός συνόλου εκπαίδευσης S , σύνολο χαρακτηριστικών $A = \{x^1, x^2, \dots, x^l\}$, διαφορετικές τιμές χαρακτηριστικών $u \in x^i$ και πιθανές κλάσεις m , ο αλγόριθμος εκμάθησης Δέντρων Απόφασης που ακολουθεί την παραπάνω βασική μεθοδολογία μπορεί να παρουσιαστεί βηματικά ως εξής [8][40][41]:

Data: S, A, m

Result: Δέντρο απόφασης

if: Όλες οι ετικέτες των παραδειγμάτων = m **then**

return την κατηγορία-φύλλο m

else

$x^i \leftarrow$ χαρακτηριστικό που διαχωρίζει καλύτερα το S

for $u \in x^i$ **do**

Πρόσθεσε νέο κόμβο $x^i = u$

end

$S_u \leftarrow$ υποσύνολο παραδειγμάτων του $S : x^i = u$

if S_u είναι άδειο **then**

Πρόσθεσε ένα κόμβο απόφασης με τιμή την κλάση που πλειοψηφεί στο σύνολο S

else

Πρόσθεσε υπόδεντρο $(S_u, A - \{x^i\}, m)$

end

5.1.3 Επιλογή χαρακτηριστικών

Το πιο σημαντικό στάδιο των αλγορίθμων που χρησιμοποιούνται στο πρόβλημα μάθησης Δέντρων Απόφασης είναι η εύρεση εκείνου του γνωρίσματος που διαχωρίζει όσο γίνεται πιο αποδοτικά τις κλάσεις. Ο τρόπος επιλογής τους είναι και η διαφορά μεταξύ

των διαφορετικών αλγορίθμων που έχουν σχεδιαστεί για το πρόβλημα αυτό. Από τα πιο γνωστά μέτρα επιλογής είναι το κέρδος πληροφορίας (information gain), ο δείκτης Gini (Gini index) και το σφάλμα ταξινόμησης (Classification error) [1][8].

Κέρδος πληροφορίας

Κάθε χαρακτηριστικό έχει το δικό του κέρδος πληροφορίας το οποίο σχετίζεται με την ικανότητα του να διαχωρίζει αποτελεσματικά τα δεδομένα εκπαίδευσης στις πιθανές κλάσεις. Συνέπως, σκοπός είναι η επιλογή των γνωρισμάτων με υψηλότερο πληροφοριακό κέρδος πρώτα. Ένα κατάλληλο μέτρο για την μέτρησή του είναι η αναμενόμενη ποσότητα πληροφορίας που παρέχεται από ένα χαρακτηριστικό. Η θεωρία της πληροφορίας αναπτύχθηκε από τον Shannon και μετρά το περιεχόμενο της πληροφορίας σε δυαδικά ψηφία (bits). Πιο συγκεκριμένα, η ποσότητα πληροφορίας που περιέχει ένα γεγονός με πιθανότητα εμφάνισης p είναι $I(p) = -\log_2 p$. Όσο μικρότερη είναι η πιθανότητα p , τόσο μεγαλύτερη είναι η πληροφορία του γεγονότος. Έστω μια τυχαία μεταβλητή X που μπορεί να πάρει κάποια από τις τιμές u_1, u_2, \dots, u_J και $p(u_j)$ η πιθανότητα η X να πάρει την τιμή u_j . Η αναμενόμενη πληροφορία των γεγονότων u_1, u_2, \dots, u_J είναι:

$$\mathbb{E}_{[p(u_1), \dots, p(u_J)]} = \sum_{j=1}^J p(u_j) I(p(u_j)) = - \sum_j p(u_j) \log_2 p(u_j) =: H(\mathbf{p}).$$

Η $H(\mathbf{p})$ ονομάζεται εντροπία της διακριτής τυχαίας μεταβλητής X . Με το διάνυσμα $\mathbf{p} = (p(u_1), \dots, p(u_J))$ συμβολίζουμε τις πιθανότητες εμφάνισης των πιθανών τιμών (u_1, u_2, \dots, u_J) της X . Στο πρόβλημα της ταξινόμησης ως γεγονότα θεωρούνται οι πιθανές κλάσεις. Μειώνοντας την εντροπία, μειώνεται και η αβεβαιότητα ως προς την ταξινόμηση των παραδειγμάτων. Στόχος μας είναι η διάσπαση του χώρου των χαρακτηριστικών με βάση κάποιο χαρακτηριστικό που θα επιφέρει μεγάλη μείωση της εντροπίας. Στην περίπτωση που υπάρχει μια μόνο κλάση τότε η εντροπία είναι 0, καθώς είμαστε βέβαιοι για την κλάση που κατηγοριοποιούνται τα δεδομένα. Από την αλλαγή, γίνεται μέγιστη αν τα δεδομένα είναι ομοιόμορφα κατανεμημένα σε κάθε κλάση, δηλαδή η πιθανότητα εμφάνισης κάθε κλάσης είναι $p = 1/m$ και η μέγιστη εντροπία ισούται με $-mp \log_2 p$, όπου m ο αριθμός των κλάσεων. Στο Διάγραμμα 5.1 β', φαίνεται η συμπεριφορά της εντροπίας σε ένα πρόβλημα ισοπίθανων κλάσεων καθώς αυξανουμε το πλήθος αυτών. Παρατηρούμε ότι για $m > 2$ έχει ως αποτέλεσμα η εντροπία να είναι μεγαλύτερη του 1.

Στην περίπτωση των δύο κλάσεων, θα συμβολίσουμε το πλήθος των θετικών και αρνητικών παραδειγμάτων που υπάρχουν στο σύνολο εκπαίδευσης S με $\#pos$ και $\#nos$ αντίστοιχα. Τότε, η εντροπία της τυχαίας μεταβλητής Y , όπου οι πιθανότητες εμφάνισης της θετικής και αρνητικής κλάσης είναι p και $1 - p$ αντίστοιχα, υπολογίζεται

ως:

$$H(\mathbf{p}) = -p \log p - (1-p) \log(1-p)$$

$$H\left(\frac{\#pos}{\#pos + \#neg}, \frac{\#neg}{\#pos + \#neg}\right) = -\frac{\#pos}{\#pos + \#neg} \log \frac{\#pos}{\#pos + \#neg} - \frac{\#neg}{\#pos + \#neg} \log \frac{\#neg}{\#pos + \#neg}.$$

Έστω ένα χαρακτηριστικό $x^i = [u_1, \dots, u_J]$ που χωρίζει το σύνολο εκπαίδευσης σε E_1, \dots, E_J σύνολα. Τότε, η εντροπία του συνόλου E_j είναι $H\left(\frac{\#pos_j}{\#pos_j + \#neg_j}, \frac{\#neg_j}{\#pos_j + \#neg_j}\right)$ bits. Η πιθανότητα ένα τυχαίο παράδειγμα να πάρει την τιμή u_j είναι $\frac{\#pos_j + \#neg_j}{\#pos + \#neg}$. Άρα αν τα δεδομένα διαμεριστούν σύμφωνα με το χαρακτηριστικό x^i θα χρειαστούν $\sum_{j=1}^J \frac{\#pos_j + \#neg_j}{\#pos + \#neg} H\left(\frac{\#pos_j}{\#pos_j + \#neg_j}, \frac{\#neg_j}{\#pos_j + \#neg_j}\right)$ bits για να ταξινομήσουμε ένα παράδειγμα. Στην ιδανική περίπτωση που το χαρακτηριστικό x^i ταξινομούσε τα παραδείγματα στην πραγματική τους κλάση τότε η αβεβαιότητα θα ήταν 0, θα χρειαζόταν 0 bits και το πληροφοριακό κέρδος θα ήταν μέγιστο. Συνεπώς, το κέρδος της πληροφορίας του χαρακτηριστικού x^i είναι η αρχική πληροφορία που απαιτούνταν αρχικά μείον την πληροφορία που χρειάζεται μετά την διαμέριση με βάση το χαρακτηριστικό αυτό. Δηλαδή, είναι:

$$Gain(S, x^i) = H\left(\left[\frac{\#pos}{\#pos + \#neg}, \frac{\#neg}{\#pos + \#neg}\right]\right) - \sum_{j=1}^J \frac{\#pos_j + \#neg_j}{\#pos + \#neg} H\left(\left[\frac{\#pos_j}{\#pos_j + \#neg_j}, \frac{\#neg_j}{\#pos_j + \#neg_j}\right]\right).$$

Έτσι, σε κάθε βήμα κατασκευής του δέντρου, υπολογίζεται για κάθε διαθέσιμο χαρακτηριστικό το πληροφοριακό κέρδος και επιλέγεται εκείνο με το υψηλότερο.

Δείκτης Gini

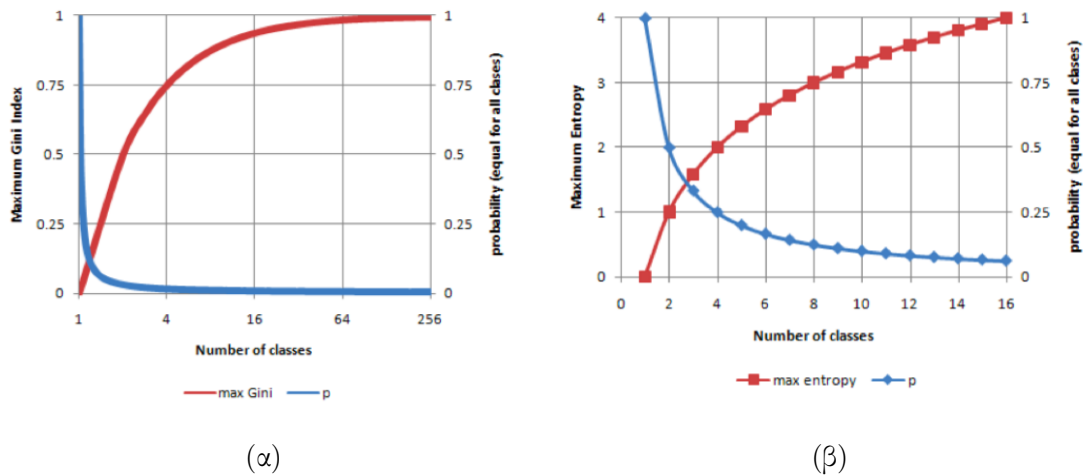
Ένα μέτρο που μπορεί να χρησιμοποιηθεί αντί του κέρδους πληροφορίας είναι ο δείκτης Gini. Για ένα χαρακτηριστικό x^i ορίζεται ως εξής:

$$Gini(x^i) = 1 - \sum_j p(\omega_j | x^i)^2,$$

όπου $p(\omega_j | x^i)$ είναι η σχετική συχνότητα της κλάσης ω_j στον κόμβο χαρακτηριστικό x^i . Ο δείκτης αυτός παίρνει την μέγιστη τιμή του όταν οι κλάσεις είναι ισοπίθανες ενώ είναι 0 όταν υπάρχει μια μοναδική κλάση αφού $1 - 1^2 = 0$. Στο Διάγραμμα 5.1α', φαίνεται πως αλλάζει ο δείκτης Gini καθώς αυξάνει ο αριθμός των κατηγοριών m . Παρατηρούμε πως σε αντίθεση με την εντροπία δεν ξεπερνάει την τιμή 1.

Σφάλμα ταξινόμησης

Ένας εναλλακτικός τρόπος να κάνουμε την διαμέριση των δεδομένων είναι να χρησιμοποιήσουμε ως κριτήριο το σφάλμα ταξινόμησης για κάθε κόμβο-χαρακτηριστικό



Διάγραμμα 5.1: Συμπεριφορά του δείκτη Gini (α') και της εντροπίας (β') καθώς αυξάνουμε τις πιθανές κλάσεις.

x^i :

$$err(x^i) = 1 - \max_{\omega_j} p(\omega_j | x^i).$$

Όμοια με την εντροπία και τον δείκτη Gini, το σφάλμα ταξινόμησης όταν μπορούμε να κάνουμε τέλεια κατηγοριοποίηση είναι $1 - \max(1) = 0$. Η τιμή του κυμαίνεται στο διάστημα $[0,1]$ και σε περίπτωση ισοπίθανων κλάσεων που παίρνει την μέγιστη τιμή του ($1 - \max\{1/m\} = 1 - 1/m$) αυτή είναι ίση με την μέγιστη τιμή του δείκτη Gini που είναι $1 - m(1/m)^2 = 1 - 1/m$.

Παράδειγμα

Θέλουμε να κατασκευάσουμε ένα Δέντρο Απόφασης ώστε να πάρουμε την απόφαση για το αν ο πελάτης θα περιμένει στο εστιατόριο με βάση τα δεδομένα εκπαίδευσης και τα χαρακτηριστικά που φαίνονται στο Διάγραμμα 5.2 [46]. Έστω ότι p, n είναι ο

1. **Alternate**: is there an alternative restaurant nearby?
2. **Bar**: is there a comfortable bar area to wait in?
3. **Fri/Sat**: is today Friday or Saturday?
4. **Hungry**: are we hungry?
5. **Patrons**: number of people in the restaurant (None, Some, Full)
6. **Price**: price range (\$, \$\$, \$\$\$)
7. **Raining**: is it raining outside?
8. **Reservation**: have we made a reservation?
9. **Type**: kind of restaurant (French, Italian, Thai, Burger)
10. **WaitEstimate**: estimated waiting time (0-10, 10-30, 30-60, >60)

Example	Attributes										Target
	Alt	Bar	Fri	Hun	Pat	Price	Rain	Res	Type	Est	
X_1	T	F	F	T	Some	\$\$\$	F	T	French	0-10	T
X_2	T	F	F	T	Full	\$	F	F	Thai	30-60	F
X_3	F	T	F	F	Some	\$	F	F	Burger	0-10	T
X_4	T	F	T	T	Full	\$	F	F	Thai	10-30	T
X_5	T	F	T	F	Full	\$\$\$	F	T	French	>60	F
X_6	F	T	F	T	Some	\$\$	T	T	Italian	0-10	T
X_7	F	T	F	F	None	\$	T	F	Burger	0-10	F
X_8	F	F	F	T	Some	\$\$	T	T	Thai	0-10	T
X_9	F	T	T	F	Full	\$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	\$\$\$	F	T	Italian	10-30	F
X_{11}	F	F	F	F	None	\$	F	F	Thai	0-10	F
X_{12}	T	T	T	T	Full	\$	F	F	Burger	30-60	T

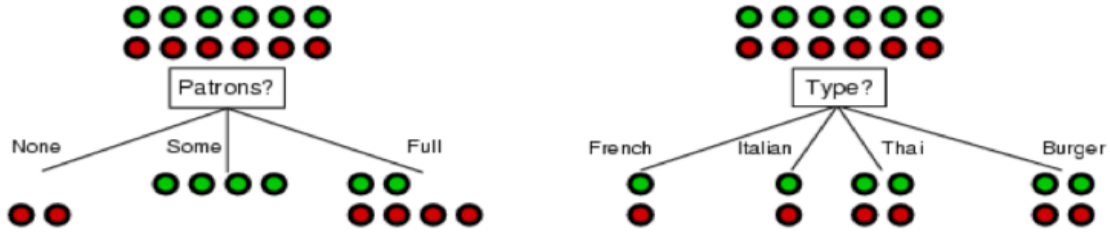
(α)

(β)

Διάγραμμα 5.2: Παράδειγμα εστιατορίου.

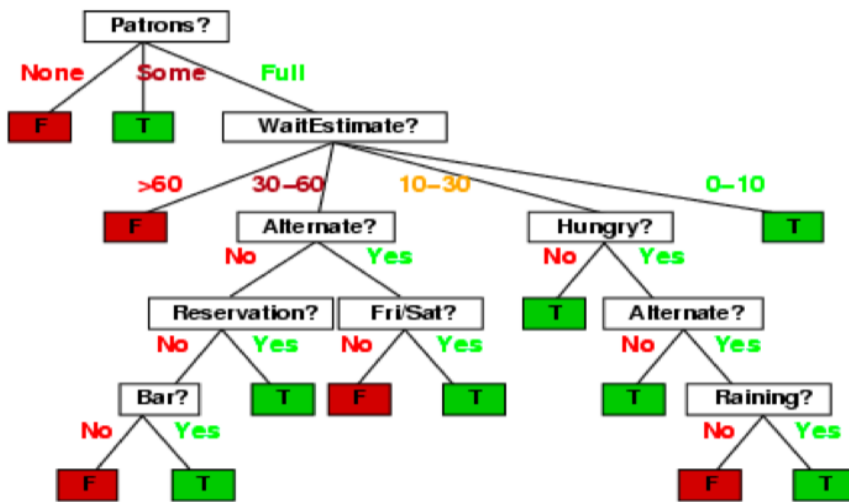
αριθμός των ανθρώπων που περιμένουν και φεύγουν αντίστοιχα. Οι εκ των προτέρων πιθανότητες των $Wait=True$ και $Wait=False$ είναι $P(True) = \frac{p}{n+p}$, $P(False) = \frac{n}{n+p}$.

Τότε, η εντροπία της μεταβλητής Wait είναι $H(\frac{p}{n+p}, \frac{n}{n+p}) = -\frac{6}{12} \log_2 \frac{6}{12} - \frac{6}{12} \log_2 \frac{6}{12} = -\log_2 \frac{1}{2} = \log_2 2 = 1bit$. Ως ρίζα του δέντρου θέλουμε να επιλέξουμε εκείνο το χαρακτηριστικό που θα έχει μικρή αβεβαιότητα ως προς την ταξινόμηση, θα έχει δηλαδή χαμηλή εντροπία και μεγάλο κέρδος πληροφορίας. Έστω ότι επιλέγουμε να συγκρίνουμε τα χαρακτηριστικά Patrons και Type.



Διάγραμμα 5.3: Διαμέριση δεδομένων με βάση τα χαρακτηριστικά Patrons, Type.

Αν διαλέξουμε το χαρακτηριστικό Type παρατηρούμε ότι για κάθε κλαδί της, η εντροπία είναι $H(\frac{1}{2}, \frac{1}{2}) = 1$. Συνεπώς, το κέρδος πληροφορίας είναι $Gain(S, Type) = 1 - 1 = 0bits$. Αν επιλέξουμε το γνώρισμα Patrons τότε στα κλαδιά None, Some η εντροπία είναι 0, αφού $H(0, 1) = 0$ ενώ για το Full είναι $H(\frac{2}{6}, \frac{4}{6})$. Συνεπώς, $Gain(S, Patrons) = 1 - H(\frac{2}{6}, \frac{4}{6}) \simeq 0.541bits$. Άρα, επιλέγοντας το γνώρισμα Patrons κερδίζουμε περισσότερη πληροφορία. Επαναλαμβάνοντας ανάλογες συγκρίσεις μεταξύ των διαθέσιμων χαρακτηριστικών αποφασίζουμε για την σειρά τους που θα διαμορφώσουν το Δέντρο Απόφασης του Διαγράμματος 5.4.



Διάγραμμα 5.4: Δέντρο απόφασης του παραδείγματος εστιατορίου.

5.1.4 Αλγόριθμοι εκμάθησης

Η εκπαίδευση του αλγορίθμου αναφέρεται στην διαδικασία οικοδόμησης του δέντρου χρησιμοποιώντας τα ήδη ταξινομημένα παραδείγματα. Πολλοί αλγόριθμοι έχουν σχε-

διαστεί για το πρόβλημα εκμάθησης ενός δέντρου απόφασης. Ανάμεσα στους πιο γνωστούς είναι οι ID3, C4.5 και ο CART.

Ο ID3 είναι ένα απλό σύστημα που έχει αναπτυχθεί σε 600 γραμμές κώδικα PASCAL. Ακολουθεί την βασική μεθοδολογία που κατασκευής Δέντρων Απόφασης εφαρμόζοντας την στατιστική ιδιότητα του κέρδους πληροφορίας για να εντοπίσει το πιο χρήσιμο γνώρισμα με βάση το οποίο θα διαχωρίσει τα δεδομένα. Ο C4.5 έχει αναπτυχθεί σε περίπου 9000 γραμμές κώδικα C και είναι μια επέκταση του ID3 καθώς τον βελτιώνει σε αρκετούς τομείς. Ακολουθεί επίσης την βασική μεθοδολογία κατασκευής αλλά χρησιμοποιεί τον Λόγο κέρδους πληροφορίας (Gain ratio). Ένα μειονέκτημα του κέρδους πληροφορίας είναι ότι έχει την τάση να επιλέγει χαρακτηριστικά με μεγάλο σύνολο τιμών σε σχέση με άλλα που λαμβάνουν μικρότερες με αποτέλεσμα να έχουμε ένα πλατύ δέντρο. Ας υποθέσουμε ότι σε ένα πρόβλημα ταξινόμησης έχουμε ένα χαρακτηριστικό της μορφής $x^i = \{u_1, u_2, \dots, u_N\}$, όπου N ο αριθμός δεδομένων εκπαίδευσης και $\forall u_i$ αντιστοιχεί ένα στιγμιότυπο. Το χαρακτηριστικό αυτό θα έχει την μεγαλύτερη τιμή κέρδους αφού θα διαχωρίζει τα στιγμιότυπα σε μονοσύνολα που το καθένα θα είχε εντροπία ($H(0, 1) = H(1, 0) = 0$). Συνεπώς, σκοπός είναι να διαλέξουμε το χαρακτηριστικό εκείνο που μεγιστοποιεί το κέρδος και προκαλεί μικρό αριθμό διαμερίσεων. Την ιδιότητα αυτή έχει ο Λόγος κέρδους πληροφορίας που ορίζεται ως [20][46][8]:

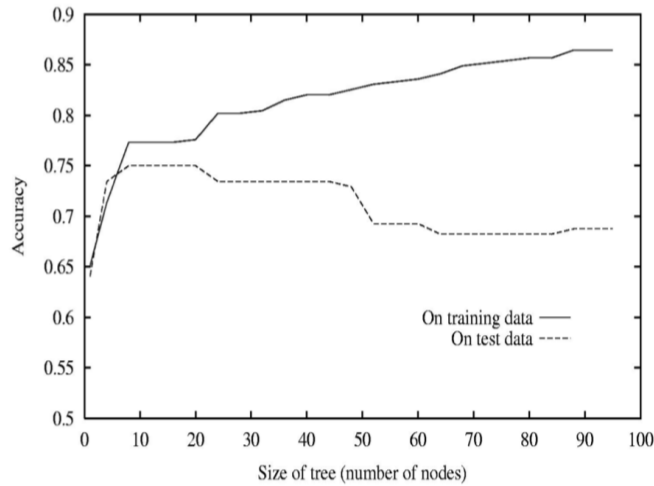
$$GainRatio(S, x^i) = \frac{Gain(S, x^i)}{SplitInfo(S, x^i)},$$

όπου το $SplitInfo(S, x^i)$ δείχνει το πόσο πολύ χωρίζεται ένα σύνολο από ένα χαρακτηριστικό και είναι:

$$SplitInfo(S, x^i) = - \sum_{j=1}^J \frac{|S_j|}{|S|} \log \left(\frac{|S_j|}{|S|} \right),$$

όπου $|S_j|$ είναι ο αριθμός των παραδειγμάτων στο κλαδί j . Επιθυμούμε κάθε όρος του αθροίσματος να είναι μεγάλος, δηλαδή το $\frac{|S_j|}{|S|}$ να είναι μεγάλο (πολλά παραδείγματα σε κάθε κλαδί).

Όπως φαίνεται στο Διάγραμμα 5.5, καθώς αυξάνει το μέγεθος το δέντρου δημιουργείται το πρόβλημα της υπερπροσαρμοστικότητας. Προκειμένου να αποφευχθεί το πρόβλημα της υπερπροσαρμοστικότητας ο C4.5 'κλαδεύει' το δέντρο αφού έχει φτάσει στην πλήρη ανάπτυξη του. Έτσι, ο αλγόριθμος έχει τρεις επιλογές, αν θα αφήσει το υπόδεντρο ως έχει, αν θα το αντικαταστήσει με ένα κόμβο-φύλλο ή αν θα το ανυψώσει δηλαδή να το αντικαταστήσει με κάποιον από τους άμεσους απογόνους, αποκόπτοντας φύλλα και αναδιανέμοντας τα στιγμιότυπα στο πάνω επίπεδο. Για να κάνει την τελική επιλογή ο C4.5 υπολογίζει τα άνω φράγματα των πιθανοτήτων λάθους για κάθε μια από τις εναλλακτικές επιλογές που έχει και αποφασίζει να εφαρμόσει αυτή με το μικρότερο άνω φράγμα. Θα αναλύσουμε την διαδικασία σε ένα πρόβλημα δύο κλάσεων για λόγους απλότητας. Αρχικά, θα υπολογίσουμε το άνω φράγμα της πιθανότητας λάθους για ένα κλαδί. Έστω λοιπόν ότι σε ένα κλαδί έχουμε N στιγμιότυπα εκ των οποίων τα M είναι λάθος ταξινομημένα. Προφανώς, η μεταβλητή M ακολουθεί την διωνυμική κατανομή με πιθανότητα $p_{reasonable\ upper\ bound}$. Σκοπός μας είναι η πιθανότητα να κάνουμε το πολύ



Διάγραμμα 5.5: Πρόβλημα υπεπροσαρμοστικότητας στα Δέντρα Απόφασης.

Μι λάθη να είναι μεγάλη, τουλάχιστον μεγαλύτερη ενός α . Δηλαδή [46][8][28]:

$$P_{M \sim \text{Bin}(N, p_{\text{reasonable upper bound}})}(M \text{ ή λιγότερα}) \geq \alpha$$

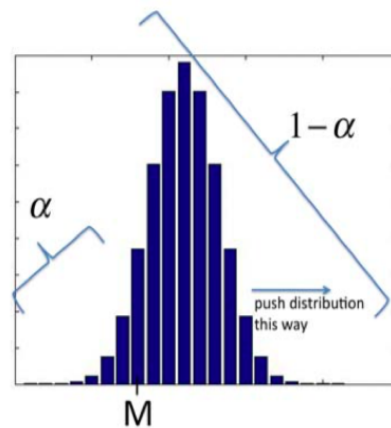
$$\sum_{z=0}^M \text{Bin}(z, N, p_{\text{reasonable upper bound}}) \geq \alpha.$$

Παρατηρούμε πως για δεδομένο M καθώς αυξάνουμε το $p_{\text{reasonable upper bound}}$ το $\sum_{z=0}^M \text{Bin}(z, N, p_{\text{reasonable upper bound}})$ μειώνεται. Όπως φαίνεται στο Διάγραμμα 5.6, θα υπάρξει λοιπόν μια τιμή p_α για την οποία:

$$\sum_{z=0}^M \text{Bin}(z, N, p_\alpha) \simeq \alpha$$

$$\sum_{z=0}^M \binom{N}{z} p_\alpha^z (1 - p_\alpha)^{N-z} - z \simeq \alpha \text{ για } M > 0.$$

Δεδομένου του N , M και α μπορούμε να υπολογίσουμε το κατώτατο άνω φράγμα της πιθανότητας λάθους για ένα κλάδι, p_α . Ο αλγόριθμος C4.5 χρησιμοποιεί την τιμή $\alpha=0.25$ και υπολογίζει έτσι p_α για κάθε κλάδι. Για την πρώτη περίπτωση, υπολογίζουμε το μέσο όρο των p_α των κλαδιών του χαρακτηριστικού που ελέγχουμε. Στην δεύτερη περίπτωση όπου αντικαθιστούμε το υπόδεντρο με ένα φύλλο υπολογίζουμε επίσης το p_α από την εξίσωση $\alpha = \sum_{z=0}^M \text{Bin}(z, N, p_\alpha)$. Ανάλογα πράττουμε και στην τρίτη περίπτωση όπου βρίσκουμε τον μέσο όρο των p_α των κλαδιών του πλέον υπερυψωμένου δέντρου. Τελικά, διαλέγω το δέντρο με την μικρότερο άνω φράγμα p_α . Επίσης, σε αντίθεση με τον ID3, ο C4.5 μπορεί να διαχειριστεί αυτόματα περιπτώσεις συνεχών τιμών των χαρακτηριστικών χρησιμοποιώντας την τεχνική της δυαδικής διακριτοποίησης η οποία παρέχει την ίδια ακρίβεια πρόβλεψης με την πολυ-διακριτοποίηση αλλά επιτυγχάνει κατά μέσο όρο διπλάσια ταχύτητα. Ακόμη, ενώ σε μικρές βάσεις δεδομένων έχει πολύ καλές επιδόσεις, σε μεγαλύτερες είτε αυξάνει πολύ ο χρόνος επεξεργασίας είτε μειώνεται υπερβολικά η ακρίβεια της ταξινόμησης.



Διάγραμμα 5.6: Διωνυμική κατανομή για $p = 0.5$. Αυξάνοντας το p , η κατανομή μετακινείται προς τα δεξιά, και για δεδομένο M το $\sum_{z=0}^M \text{Bin}(z, N, p)$ μειώνεται μέχρι κάποιο σημείο όπου το άθροισμα θα λάβει την οριακή τιμή α .

Ο αλγόριθμος CART έχει κοινή βασική μεθοδολογία με τους προηγούμενους. Οι διαφορές του με τον C4.5 είναι πως εκτελεί πάντα δυαδικούς διαχωρισμούς των δεδομένων ανεξάρτητα αν τα γνωρίσματα είναι κατηγορικά ή αριθμητικά χρησιμοποιώντας τον δείκτη Gini. Η στρατηγική που ακολουθεί είναι πως αρχικά αναπτύσσεται ένα μεγάλο δέντρο και στην συνέχεια κλαδεύεται βρίσκοντας το υπόδεντρο που ελαχιστοποιεί το κόστος πολυπλοκότητας το οποίο υπολογίζεται ως [20][46]:

$$\text{cost}(\text{subtree}) = \sum_{\text{leaves } j} \sum_{x_i \in \text{leaf } j} \mathbb{1}[y_i \neq \text{leaf}'s \text{ Class}] + C[\text{leaves in subtree}].$$

Ο πρώτος όρος δηλώνει το κόστος των λάθος ταξινομήσεων που έγιναν στο υπόδεντρο. Το $C \geq 0$ είναι μια ρυθμιστική παράμετρος που αντιμετωπίζει το πρόβλημα της υπερπροσαρμοστικότητας. Μεγάλη τιμή του C έχει ως αποτέλεσμα Δέντρα με μικρό βάθος ενώ όταν παίρνει μικρή τιμή τότε το υπόδεντρο που ελαχιστοποιεί το κόστος έχει καλύτερη εφαρμογή στα δεδομένα εκπαίδευσης. Συνήθως, το C επιλέγεται χρησιμοποιώντας την τεχνική της διασταυρωμένης επικύρωσης.

5.2 Μηχανές Διανυσμάτων Υποστήριξης

5.2.1 Εισαγωγή

Οι Μηχανές Διανυσμάτων Υποστήριξης (Support vector machines, SVMs)[10] είναι από τις πρόσφατες τεχνικές μάθησης υπό επίβλεψη. Η πρώτη παρουσίαση της μεθοδολογίας τους έγινε το 1992 από τους Boser, Guyon και Vapnik στο COLT-92 (Computational learning theory-92). Χρησιμοποιείται συχνά σε προβλήματα αναγνώρισης προτύπων (αναγνώριση αντικειμένων/ χαρακτήρων της αλφαβήτου, εντοπισμό προσώπων

σε εικόνες κ.α.). Μάλιστα, σε προβλήματα ανγνώρισης γραφής οι Μηχανές Διανυσμάτων Υποστήριξης δίνουν ακρίβεια συγκρίσιμη με αυτή των πολύπλοκων νευρωνικών δικτύων.

Πιο συγκεκριμένα, οι Μηχανές Διανυσμάτων Υποστήριξης είναι γραμμικές συναρτήσεις της μορφής $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$ που διαχωρίζουν τις κλάσεις. Τα διανύσματα \mathbf{x} είναι τα σημεία του συνόλου εκπαίδευσης (σημεία των κλάσεων προς διαχωρισμό), \mathbf{w} είναι το διάνυσμα βάρους δηλαδή το διάνυσμα προσανατολισμού της γραμμικής συνάρτησης f και το b είναι μια σταθερά. Ο SVM ταξινομητής κατηγοριοποιεί τις παρατηρήσεις στην κλάση $+1$ αν $f(\mathbf{x}) \geq 0$, διαφορετικά στην -1 . Βέβαια, οι Μηχανές Διανυσμάτων Υποστήριξης μπορούν να χρησιμοποιηθούν και στην μη γραμμική ταξινόμηση, κάνοντας χρήση μη γραμμικών συναρτήσεων πυρήνα που καταφέρνουν να μετασχηματίσουν τον μη γραμμικό χώρο των χαρακτηριστικών σε έναν γραμμικά διαχωρίσιμο χώρο μεγαλύτερης διάστασης.

5.2.2 Γραμμικώς διαχωρίσιμα δεδομένα

Στην απλή εκδοχή του, ο αλγόριθμος εκπαιδεύεται με δεδομένα που είναι γραμμικώς διαχωρίσιμα. Ας θεωρήσουμε ένα παράδειγμα δύο κλάσεων και δύο χαρακτηριστικών για ένα σύνολο δεδομένων. Στην περίπτωση αυτή, οι Μηχανές Διανυσμάτων Υποστήριξης προσπαθούν να βρουν ένα υπερεπίπεδο (ευθεία) που να διαχωρίζει τα παραδείγματα εκπαίδευσης με τέτοιο τρόπο ώστε τα παραδείγματα που ανήκουν στην ίδια κατηγορία να είναι στην ίδια πλευρά του υπερεπιπέδου. Από όλα τα πιθανά υπερεπίπεδα, αναζητά το βέλτιστο, εκείνο δηλαδή που θα διαχωρίσει τις δύο ομάδες με την μέγιστη δυνατή απόσταση (μέγιστο περιθώριο). Ο βέλτιστος ταξινομητής βρίσκεται στην μέση του περιθωρίου και ορίζεται από ένα υποσύνολο των σημείων του συνόλου εκπαίδευσης που ονομάζονται διανύσματα υποστήριξης (support vectors).

Μέγιστο περιθώριο

Η μαθηματική έκφραση ενός υπερεπιπέδου είναι $\mathbf{w}^T \mathbf{x} + b = 0$, όπου το \mathbf{w} είναι κάθετο σε αυτό. Όπως είδαμε, οι Μηχανές Διανυσμάτων Υποστήριξης προσπαθούν να μεγιστοποιήσουν το ελάχιστο περιθώριο. Για να βρούμε λοιπόν το μέγιστο περιθώριο, θεωρούμε δύο τυχαία υπερεπίπεδα που διαχωρίζουν τις κλάσεις μεταξύ τους και στην συνέχεια μεγιστοποιούμε την απόστασή τους. Η μέγιστη απόσταση είναι το μέγιστο περιθώριο. Έστω ότι έχουμε ένα σύνολο εκπαίδευσης $S = \{\mathbf{x}_i, y_i\}_{i=1}^N$. Για λόγους απλότητας θεωρούμε την περίπτωση που τα \mathbf{w}, \mathbf{x} είναι διανύσματα δύο διαστάσεων. Οι ετικέτες y_i των \mathbf{x}_i παίρνουν τιμές $\{-1, +1\}$. Τα δύο τυχαία υπερεπίπεδα H_0, H_1 είναι της μορφής [20][39][14]:

$$H_0 : \mathbf{w}^T \mathbf{x} + b = -\delta,$$

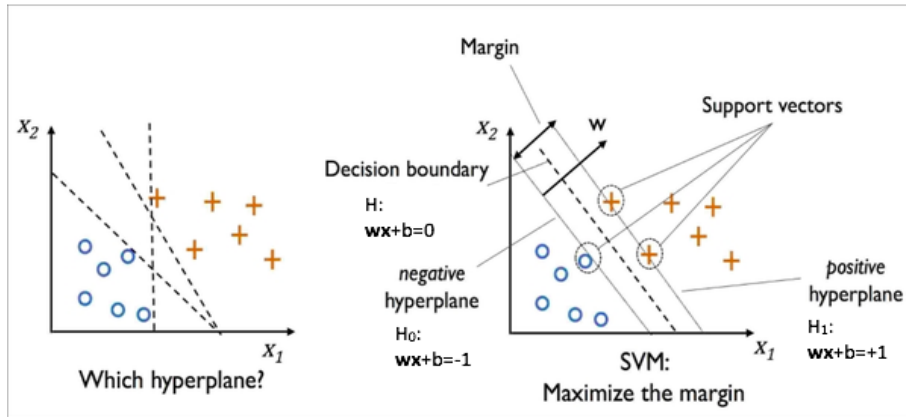
$$H_1 : \mathbf{w}^T \mathbf{x} + b = \delta,$$

δηλαδή απέχουν απόσταση δ από το βέλτιστο υπερεπίπεδο $H : \mathbf{w}\mathbf{x} + b = 0$. Επιλέγοντας τυχαία $\delta=1$ έχουμε:

$$H_0 : \mathbf{w}\mathbf{x} + b = -1,$$

$$H_1 : \mathbf{w}\mathbf{x} + b = 1.$$

Όπως φαίνεται στο Διάγραμμα 5.7, για κάθε παρατήρηση $\mathbf{x}_i, i = 1, \dots, N$ ισχύει είτε



Διάγραμμα 5.7: Στο αριστερό Διάγραμμα φαίνονται όλα τα πιθανά υπερεπίπεδα που μπορούν να σχηματιστούν. Στο δεξί αναπαριστάται το βέλτιστο υπερεπίπεδο που μεγιστοποιεί το ελάχιστο περιθώριο.

$\mathbf{w}\mathbf{x}_i + b > 1$ όταν $y_i = 1$ είτε $\mathbf{w}\mathbf{x}_i + b < -1$ όταν $y_i = -1$. Πολλαπλασιάζοντας τις δύο αυτές σχέσεις με την ετικέτα τους καταλήγουμε στο ότι για κάθε \mathbf{x}_i ισχύει $y_i(\mathbf{w}\mathbf{x}_i + b) \geq 1$. Έστω ένα διάνυσμα $\mathbf{x}_0 \in H_0$ και $\mathbf{x}_1 \in H_1$. Τότε, η απόσταση (W) μεταξύ των δύο αυτών υπερεπιπέδων είναι [8][20]:

$$\begin{aligned} W &= (\mathbf{x}_1 - \mathbf{x}_0) \frac{\mathbf{w}}{\|\mathbf{w}\|} \\ &= \frac{1 - b - (-1 - b)}{\|\mathbf{w}\|} \\ &= \frac{2}{\|\mathbf{w}\|}, \end{aligned}$$

όπου $\frac{\mathbf{w}}{\|\mathbf{w}\|}$ είναι το μοναδιαίο κάθετο διάνυσμα στα υπερεπίπεδα. Αφού υπολογίσαμε την απόσταση μεταξύ των δύο υπερεπιπέδων, σκοπός μας είναι να την μεγιστοποιήσουμε. Ισοδύναμα, μπορούμε να μεγιστοποιήσουμε το $\frac{1}{\|\mathbf{w}\|}$ ή να ελαχιστοποιήσουμε το $\|\mathbf{w}\|$. Άρα, μπορούμε να ελαχιστοποιήσουμε την ποσότητα $\frac{1}{2} \|\mathbf{w}\|^2$ καθώς η συνάρτηση $f(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2$ είναι γνησίως αύξουσα. Καταλήγουμε λοιπόν στο παρακάτω πρόβλημα ελαχιστοποίησης:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \tag{5.1}$$

με περιορισμό $y_i(\mathbf{w}\mathbf{x}_i + b) \geq 1, i = 1, 2, \dots, N$.

Προτού λύσουμε το πρόβλημα 5.1, θα μελετήσουμε γενικότερα το πρόβλημα βελτιστοποίησης υπό περιορισμούς.

Πρόβλημα βελτιστοποίησης υπό περιορισμούς

Στην πιο απλή περίπτωση, ο περιορισμός είναι μια ισότητα και το πρόβλημα λύνεται με την βοήθεια των πολλαπλασιαστών Lagrange και διατυπώνεται ως εξής [14][8]:

$$\min_{\mathbf{w}} f(\mathbf{w})$$

με περιορισμό $h_i(\mathbf{w}) = 0, i = 1, 2, \dots, N$.

Ορίζουμε την συνάρτηση Lagrange ως:

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}) = f(\mathbf{w}) + \sum_{i=1}^N \lambda_i h_i(\mathbf{w}).$$

Οι συντελεστές λ_i ονομάζονται πολλαπλασιαστές Lagrange. Στην συνέχεια, λαμβάνουμε τα $\mathbf{w}, \boldsymbol{\lambda}$ λύνοντας τις εξισώσεις $\frac{\partial \mathcal{L}}{\partial w_j} = 0, \frac{\partial \mathcal{L}}{\partial \lambda_i} = 0$. Σκοπός μας είναι να γενικεύσουμε αυτή την διαδικασία όταν ως περιορισμό έχουμε ανισότητες. Υποθέτουμε ότι έχουμε το παρακάτω πρωτεύον (primal) πρόβλημα βελτιστοποίησης:

$$\min_{\mathbf{w}} f(\mathbf{w})$$

με περιορισμούς $g_i(\mathbf{w}) \leq 0, i = 1, \dots, k$
 $h_i(\mathbf{w}) = 0, i = 1, \dots, l$.

Για να βρούμε τον ελαχιστοποιητή της $f(\mathbf{w})$ ορίζουμε την γενικευμένη Lagrange συνάρτηση:

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\mathbf{w}) + \sum_{i=1}^k \alpha_i g_i(\mathbf{w}) + \sum_{j=1}^l \beta_j h_j(\mathbf{w}),$$

όπου α_i, β_j είναι οι πολλαπλασιαστές Lagrange. Επίσης, ορίζουμε την ποσότητα:

$$\theta_{\mathcal{P}}(\mathbf{w}) = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}).$$

Έστω μια τυχαία τιμή για το \mathbf{w} . Αν αυτή παραβιάζει κάποιον από τους περιορισμούς τότε $\theta_{\mathcal{P}}(\mathbf{w}) = \infty$, διαφορετικά $\theta_{\mathcal{P}}(\mathbf{w}) = f(\mathbf{w})$ αφού $\sum_{i=1}^k \alpha_i g_i(\mathbf{w}) \leq 0$ και $\sum_{j=1}^l \beta_j h_j(\mathbf{w}) = 0$. Ας θεωρήσουμε τώρα το πρόβλημα ελαχιστοποίησης:

$$\min_{\mathbf{w}} \theta_{\mathcal{P}}(\mathbf{w}) = \min_{\mathbf{w}} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\mathbf{w}} f(\mathbf{w}),$$

το οποίο παρατηρούμε πως είναι το ίδιο πρόβλημα με το πρωτεύον με λύση $p^* = \min_{\mathbf{w}} \theta_{\mathcal{P}}(\mathbf{w})$. Σε κάθε πρωτεύον πρόβλημα αντιστοιχεί ένα δυϊκό, το οποίο θα περιγράψουμε παρακάτω. Η βέλτιστη λύση του ενός δίνει πληροφορία για την βέλτιστη λύση του άλλου. Ορίζουμε λοιπόν την ποσότητα:

$$\theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}).$$

Το δυϊκό πρόβλημα παίρνει την εξής μορφή:

$$\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \theta_{\mathcal{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}: \alpha_i \geq 0} \min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}).$$

Παρατηρούμε ότι μοιάζει με το πρωτεύον πρόβλημα, με την μόνη διαφορά ότι έχει αλλάξει η σειρά των \min, \max . Εάν $d^* = \max_{\alpha, \beta: \alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha, \beta)$ τότε αποδεικνύεται ότι οι λύσεις του πρωτεύοντος και δυϊκού προβλήματος σχετίζονται σύμφωνα με την σχέση:

$$d^* = \max_{\alpha, \beta: \alpha_i \geq 0} \min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \alpha, \beta) \leq \min_{\mathbf{w}} \max_{\alpha, \beta: \alpha_i \geq 0} \mathcal{L}(\mathbf{w}, \alpha, \beta) = p^*.$$

Η ισότητα $d^* = p^*$ ισχύει κάτω από τις παρακάτω συνθήκες:

- ✓ Οι f, g_i είναι κυρτές συναρτήσεις.
- ✓ Υπάρχουν a_i, b_i τέτοια ώστε $h_i(\mathbf{w}) = a_i^T \mathbf{w} + b_i, \forall i$.
- ✓ Υπάρχει \mathbf{w} έτσι ώστε $g_i(\mathbf{w}) < 0, \forall i$.

Έτσι λοιπόν, με βάση τις παραπάνω συνθήκες υπάρχουν α^*, β^* λύσεις του δυϊκού και \mathbf{w}^* λύση του πρωτεύοντος τέτοιες ώστε $p^* = d^* = \mathcal{L}(\mathbf{w}^*, \alpha^*, \beta^*)$. Επιπλέον τα $\alpha^*, \beta^*, \mathbf{w}^*$ ικανοποιούν τις ακόλουθες συνθήκες Karush-Kuhn-Tucker:

$$\frac{\partial}{\partial w_i} \mathcal{L}(\mathbf{w}^*, \alpha^*, \beta^*) = 0, i = 1, \dots, n \quad (1) \quad (5.2)$$

$$\frac{\partial}{\partial \beta_i} \mathcal{L}(\mathbf{w}^*, \alpha^*, \beta^*) = 0, i = 1, \dots, l \quad (2)$$

$$\alpha_i^* g_i(\mathbf{w}^*) = 0, i = 1, \dots, k \quad (3)$$

$$g_i(\mathbf{w}^*) \leq 0, i = 1, \dots, k \quad (4)$$

$$\alpha_i^* \geq 0, i = 1, \dots, k \quad (5).$$

Η σχέση (5.2(3)) ονομάζεται συνθήκη δυϊκής συμπληρωματικότητας (dual complementary condition) και υπονοεί πως αν για κάποια παρατήρηση \mathbf{x}_i , το $\alpha_i^* > 0$ τότε $g_i(\mathbf{w}^*) = 0$. Επίσης, με τις σχέσεις (5.2(4),(5)) ελέγχουμε ότι η λύση είναι εφικτή και οι (5.2(1),(2)) επαληθεύουν ότι τα $\mathbf{w}^*, \alpha^*, \beta^*$ ελαχιστοποιούν την Lagrange συνάρτηση ως προς τις μεταβλητές \mathbf{w}, b .

Βέλτιστο υπερεπίπεδο

Είδαμε ότι προκειμένου να βρούμε το βέλτιστο υπερεπίπεδο αρκεί να λύσουμε το πρόβλημα ελαχιστοποίησης 5.1. Ο περιορισμός μπορεί να γραφεί ως $g_i(\mathbf{w}) = -y_i(\mathbf{w}\mathbf{x}_i + b) + 1 \leq 0$. Συνεπώς, το πρόβλημα βελτιστοποίησης έχει ως εξής:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (5.3)$$

με περιορισμό $g_i(\mathbf{w}) = -y_i(\mathbf{w}\mathbf{x}_i + b) + 1 \leq 0, i = 1, 2, \dots, N$.

Δημιουργούμε την Lagrangian συνάρτηση:

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i [y_i(\mathbf{w}\mathbf{x}_i + b) - 1], \quad (5.4)$$

όπου τα α_i είναι οι πολλαπλασιαστές Lagrange. Παρατηρούμε ότι δεν υπάρχει ο όρος $\sum_{j=1}^l \beta_j h_j(w)$ καθώς έχουμε μόνο περιορισμούς ανισότητας. Στην συνέχεια, μετατρέπουμε το πρωτεύον πρόβλημα στο δυϊκό του. Υπολογίζουμε το θ_D θέτοντας τις παραγώγους της Lagrange συνάρτησης ως προς τις μεταβλητές \mathbf{w} , b ίσες με 0. Με τον τρόπο αυτό αφαιρείται η επίδρασή τους και οι άγνωστοι του προβλήματος είναι οι πολλαπλασιαστές Lagrange που πρέπει να υπολογιστούν. Άρα, έχουμε ότι:

$$\begin{aligned} \nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) &= \mathbf{w} - \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i = 0 \\ \Rightarrow \mathbf{w} &= \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \\ \frac{\partial}{\partial b} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) &= \sum_{i=1}^N \alpha_i y_i = 0. \end{aligned} \quad (5.5)$$

Αντικαθιστώντας τις σχέσεις (5.5) στην συνάρτηση (5.4) έχουμε:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\alpha}) &= \frac{1}{2} \left[\sum_i \alpha_i y_i \mathbf{x}_i \right] \left[\sum_j \alpha_j y_j \mathbf{x}_j \right] - \left[\sum_i \alpha_i y_i \mathbf{x}_i \right] \left[\sum_j \alpha_j y_j \mathbf{x}_j \right] - \sum_i \alpha_i y_i b + \sum_i \alpha_i \\ &= \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \mathbf{x}_j). \end{aligned}$$

Έτσι, το αρχικό πρόβλημα μετασχηματίζεται στο δυϊκό του, το οποίο έχει ως εξής:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \mathbf{x}_j) \\ \text{με περιορισμό } \sum_{i=1}^N \alpha_i y_i = 0, \alpha_i \geq 0, i = 1, \dots, N. \end{aligned} \quad (5.6)$$

Αφού οι συνθήκες των Karush-Kuhn Tucker ισχύουν, η λύση του δυϊκού προβλήματος θα ταυτίζεται με αυτή του πρωτεύοντος προβλήματος. Μάλιστα, από την συνθήκη 5.2(3) βλέπουμε πως είτε το $\alpha_i^* = 0$ είτε $y_i(\mathbf{w}^* \mathbf{x}_i - b^*) - 1 = 0$. Οι παρατηρήσεις με μη μηδενικό πολλαπλασιαστή Lagrange και $g_i(\mathbf{w}) = 0$ ονομάζονται διανύσματα υποστήριξης και βρίσκονται πάνω σε κάποιο από τα υπερεπίπεδα H_0, H_1 ανάλογα με την τιμή της ετικέτας y_i . Συνεχίζουμε λοιπόν με την επίλυση του δυϊκού προβλήματος. Έστω ότι οι παράμετροι $\boldsymbol{\alpha}^*$ είναι λύση του δυϊκού τότε το διάνυσμα \mathbf{w}^* :

$$\mathbf{w}^* = \sum_{i=1}^N \alpha_i^* y_i \mathbf{x}_i,$$

δημιουργεί το υπερεπίπεδο μέγιστου περιθωρίου. Η σταθερά b υπολογίζεται με βάση το σύνολο εκπαίδευσης και το διάνυσμα \mathbf{w}^* . Αρχικά, υπολογίζουμε την τιμή της παραμέτρου b^- που αφορά το υπερεπίπεδο H_0 . Για δεδομένο \mathbf{w}^* παρατηρούμε ότι το b^- θα είναι ίσο με την μεγαλύτερη δυνατή προβολή που μπορεί να έχει ένα στιγμιότυπο κλάσης - στο διάνυσμα \mathbf{w}^* :

$$b^- = \max\{\mathbf{w}^* \mathbf{x} | (\mathbf{x}, y) \in S, y = -1\}.$$

Αντίστοιχα, η τιμή του b^+ , που αφορά το υπερεπίπεδο H_1 , για δεδομένο \mathbf{w}^* θα είναι η ελάχιστη προβολή που μπορεί να έχει ένα \mathbf{x} στο διάνυσμα \mathbf{w}^* :

$$b^+ = \min\{\mathbf{w}^* \mathbf{x} | (\mathbf{x}, y) \in S, y = +1\}.$$

Αφού το βέλτιστο υπερεπίπεδο H βρίσκεται στην μέση των H_0, H_1 για την παράμετρο b^* θα ισχύει:

$$b^* = \frac{b^+ + b^-}{2}.$$

Επομένως, η εξίσωση του βέλτιστου υπερεπίπεδου είναι:

$$\mathbf{w}^* \mathbf{x} + b^* = \sum_{i=1}^N \alpha_i y_i(\mathbf{x}; \mathbf{x}) + b^*.$$

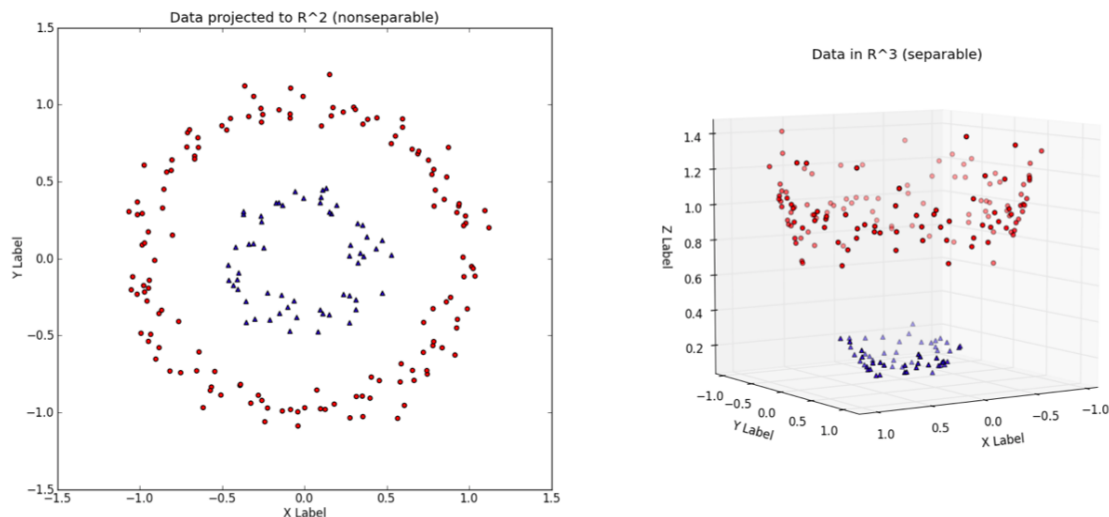
Παρατηρούμε ότι για την πρόβλεψη της κλάσης ενός νέου στιγμιότυπου \mathbf{x} μπορούμε να υπολογίσουμε μόνο τα εσωτερικά γινόμενα του \mathbf{x} και των παραδειγμάτων \mathbf{x}_i . Έτσι, αν $\sum_{i=1}^N \alpha_i y_i(\mathbf{x}; \mathbf{x}) + b^* > 0$ τότε η προβλεπόμενη κλάση είναι η $y = +1$ ενώ αν $\sum_{i=1}^N \alpha_i y_i(\mathbf{x}; \mathbf{x}) + b^* < 0$ είναι η $y = -1$. Καταλήγουμε στο συμπέρασμα ότι στην λύση του δυϊκού προβλήματος συνεισφέρουν μόνο τα διανύσματα υποστήριξης αφού μόνο για αυτά οι πολλαπλασιαστές Lagrange είναι μη μηδενικοί, με αποτέλεσμα η διαδικασία πρόβλεψης να γίνεται ευκολότερη.

5.2.3 Τέχνασμα του πυρήνα

Ένα σημαντικό πλεονέκτημα των Μηχανών Διανυσμάτων Υποστήριξης είναι πως η έννοια της γραμμικότητας που τις διακατέχει μπορεί να επεκταθεί και σε προβλήματα που το σύνολο εκπαίδευσης είναι μη γραμμικά διαχωρίσιμο. Η λειτουργία των μη γραμμικών SVM's βασίζεται στο θεώρημα του Cover [11] σύμφωνα με το οποίο ο χώρος εισόδου, όπου τα στιγμιότυπα δεν είναι γραμμικά διαχωρίσιμα, μετασχηματίζεται σε ένα χώρο μεγαλύτερης διάστασης στον οποίο τα δεδομένα εισόδου διαχωρίζονται γραμμικά. Έτσι, μέσω μιας μη γραμμικής απεικόνισης $\phi(\mathbf{x})$ καταφέρνουμε να περάσουμε από την μη γραμμική στην γραμμική περίπτωση.

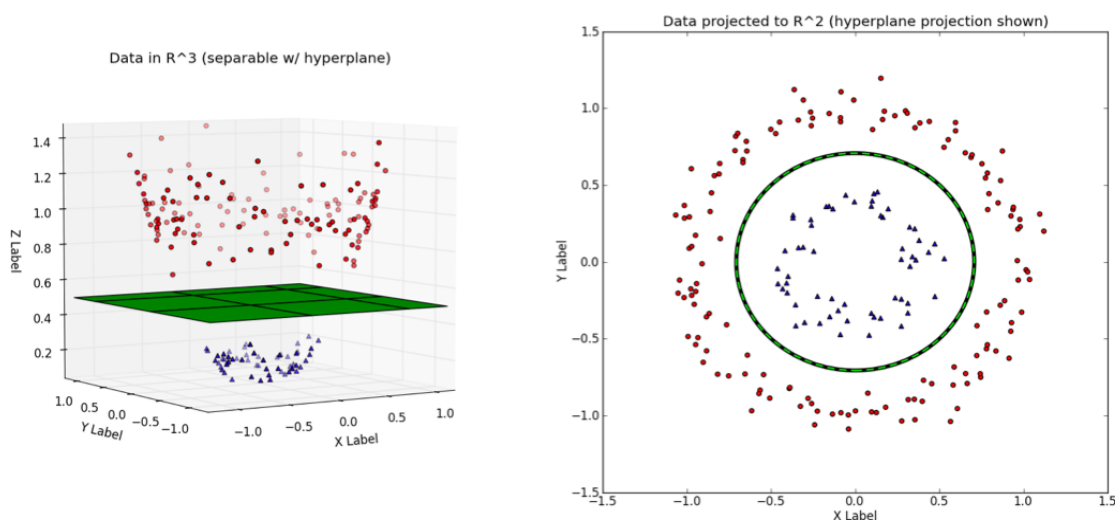
Ας θεωρήσουμε ένα παράδειγμα όπου το σύνολο εκπαίδευσης ανήκει στον δυσδιάστατο χώρο και όπως φαίνεται αριστερά του Διαγράμματος 5.8 είναι μη γραμμικά διαχωρίσιμος. Αντί λοιπόν να εργαστούμε στο χώρο εισόδου \mathbb{R}^2 μπορούμε να τον μετασχηματίσουμε σε έναν χώρο μεγαλύτερης διάστασης, όπως φαίνεται δεξιά του Διαγράμματος 5.8, ο οποίος μας επιτρέπει να εκπαιδεύσουμε έναν γραμμικό SVM ταξινομητή. Σκοπός είναι να βρούμε μια συνάρτηση $\phi: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ μέσω της οποίας θα είναι εφικτό τα μετασχηματισμένα στιγμιότυπα να διαχωριστούν γραμμικά. Στο παράδειγμα μας, μια κατάλληλη τέτοια συνάρτηση είναι η $\phi(x_1, x_2) = [x_1, x_2, x_1^2 + x_2^2]$ η οποία αφού εφαρμοστεί στα δοσμένα δεδομένα εκπαίδευσης έχει ως αποτέλεσμα την δεξιά απεικόνιση του Διαγράμματος 5.8.

Αν υποθέσουμε ότι η απεικόνιση ϕ είναι γνωστή τότε η διαδικασία έχει ως εξής: μετατρέπουμε τον χώρο εισόδου X σε X' μέσω της ϕ και εκπαιδεύουμε τον γραμμικό ταξινομητή (f) στο X' . Η πρόβλεψη της ετικέτας μιας νέας παρατήρησης



Διάγραμμα 5.8: (Αριστερά) Σύνολο εκπαίδευσης στον \mathbb{R}^2 μη γραμμικά διαχωρίσιμο. (Δεξιά) Το ίδιο σύνολο εκπαίδευσης αλλά μετασχηματισμένο, $[x_1, x_2] = [x_1, x_2, x_1^2 + x_2^2]$.

\mathbf{x} γίνεται με βάση το αποτέλεσμα της $f(\mathbf{x}) = \mathbf{w}^* \phi(\mathbf{x}) + b^*$ [47][8][20]. Δηλαδή, η διαδικασία είναι ίδια με την γραμμική περίπτωση με την διαφορά ότι κάνουμε έναν μετασχηματισμό. Έτσι, στον \mathbb{R}^3 η επιφάνεια απόφασης είναι ένα υπερεπίπεδο, σε αντίθεση με τον \mathbb{R}^2 όπου έχουμε μια μη γραμμική διαχωριστική επιφάνεια (Διάγραμμα 5.9).



Διάγραμμα 5.9: (Δεξιά) Η επιφάνεια απόφασης w είναι γραμμική όταν ο χώρος χαρακτηριστικών είναι ο \mathbb{R}^3 αλλά στον \mathbb{R}^2 είναι μη γραμμική (Αριστερά).

Στην γενική περίπτωση, ένα σύνολο εκπαίδευσης S μπορεί να μην είναι γραμμικά διαχωρίσιμο στον \mathbb{R}^L αλλά να έχει αυτή την ιδιότητα σε έναν άλλον χώρο, έστω \mathbb{R}^M , $M > L$. Έτσι, αν έχουμε τον μετασχηματισμό ϕ που μπορεί να μεταφέρει τα στιγμιότυπα στον \mathbb{R}^M τότε μπορούμε να βρούμε το υπερεπίπεδο που θα διαχωρίζει ικανοποιητικά τις κλάσεις και αν το προβάλουμε στον αρχικό χώρο \mathbb{R}^L θα είναι μια

μη γραμμική επιφάνεια απόφασης.

Η ιδέα αυτή αν και είναι ελκυστική χάρη στην απλότητά της, αντιμετωπίζει ένα σοβαρό πρόβλημα. Όταν το M αυξάνεται πολύ σε σχέση με το L τότε το υπολογιστικό κόστος είναι αρκετά μεγαλύτερο [20][48]. Ας θεωρήσουμε για παράδειγμα την προηγούμενη απεικόνιση από τον χώρο εισόδου στον χώρο χαρακτηριστικών της μορφής $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ με $\phi(\mathbf{x}) = \phi(x_1, x_2) = (x_1^2, x_2^2, x_1^2 + x_2^2) = (z_1, z_2, z_3) = \mathbf{z}$ όπου η επιφάνεια απόφασης του χώρου χαρακτηριστικών είναι $\mathbf{w}^* \phi(\mathbf{x}) = b^*$. Σύμφωνα με αυτή μπορούμε να ορίσουμε την συνάρτηση απόφασης ως:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}^* \phi(\mathbf{x}) + b^* & (5.7) \\ &= w_1^* x_1^2 + w_2^* x_2^2 + w_3^* (x_1^2 + x_2^2) - b^* \\ &= \mathbf{w}^* \mathbf{z}^* + b^* \\ &= \sum_{i=1}^3 w_i^* z_i + b^*. \end{aligned}$$

Από την σχέση (5.7) φαίνεται ότι η πολυπλοκότητα της συνάρτησης απόφασης συνδέεται με την διάσταση του χώρου των χαρακτηριστικών. Συνεπώς, όσο πιο πολύπλοκες μη γραμμικές επιφάνειες απόφασης απαιτούνται στον χώρο εισόδου τόσο μεγαλύτερους διάστασης χώρους χαρακτηριστικών χρειαζόμαστε για να μεταφερθούμε στο γραμμική περίπτωση.

Όπως δείξαμε, οι Μηχανές Διανυσμάτων Υποστήριξης τόσο στο στάδιο εκπαίδευσης όσο και ελέγχου υπολογίζουν τα εσωτερικά γινόμενα $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^L$. Το ενδιαφέρον είναι ότι δοθέντων των $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^L$, υπάρχουν συναρτήσεις οι οποίες υπολογίζουν το εσωτερικό γινόμενο του μεγαλύτερης διάστασης χώρου (χώρος χαρακτηριστικών) χωρίς να μεταφέρουν τα $\mathbf{x}_i, \mathbf{x}_j$ στο χώρο αυτό. Δηλαδή, η τιμή του εσωτερικού γινομένου στην παραγματικότητα υπολογίζεται στον χώρο εισόδου. Οι συναρτήσεις αυτές ονομάζονται συναρτήσεις πυρήνα (kernel functions) και συμβολίζονται με $k(\mathbf{x}_i, \mathbf{x}_j)$. Τυπικά, έχουμε ότι:

Για $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^L$, $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_M$, όπου $\langle \cdot, \cdot \rangle_M$ είναι ένα εσωτερικό γινόμενο στον \mathbb{R}^M , $M > L$, και ο μετασχηματισμός $\phi(\mathbf{x})$ μεταφέρει το \mathbf{x} στον \mathbb{R}^M ($\phi : \mathbb{R}^L \rightarrow \mathbb{R}^M$).

Ας δούμε πρακτικά τι σημαίνει αυτό. Το κάθετο διάνυσμα \mathbf{w}^* στο βέλτιστο υπερεπίπεδο έχει την μορφή $\mathbf{w}^* = \sum_{i=1}^N \alpha_i^* y_i \phi(\mathbf{x}_i)$ [47][8]. Γίνεται αντιληπτό ότι τα δεδομένα εκπαίδευσης \mathbf{x}_i πρέπει να μετασχηματιστούν σε $\phi(\mathbf{x}_i)$. Το βέλτιστο υπερεπίπεδο παίρνει την μορφή:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}^* \phi(\mathbf{x}) + b^* \\ &= \sum_{i=1}^N \alpha_i^* y_i \phi(\mathbf{x}_i) \phi(\mathbf{x}) + b^* \\ &= \sum_{i=1}^N \alpha_i^* y_i k(\mathbf{x}_i, \mathbf{x}) + b^*. \end{aligned}$$

Παρατηρούμε λοιπόν, ότι η πολυπλοκότητα της συνάρτησης απόφασης είναι ανάλογη του αριθμού των διανυσμάτων υποστήριξης όπως και στην περίπτωση που δεν θεωρούμε κάποιο μετασχηματισμό ϕ . Στην τελευταία ισότητα φαίνεται το τέχνασμα του

πυρήνα (kernel trick). Δηλαδή με την βοήθεια μιας κατάλληλης συνάρτησης-πυρήνα οι υπολογισμοί στον χώρο των χαρακτηριστικών απλοποιούνται σε υπολογισμούς στον χώρο εισόδου. Οι πιο διαδεδομένες συναρτήσεις πυρήνα που συναντάμε φαίνονται στον Πίνακα 5.1:

Πίνακας 5.1: Γνωστοί πυρήνες και ελεύθεροι παράμετροι.

Όνομασία πυρήνα	Συνάρτηση πυρήνα	Ελεύθεροι παράμετροι
Γραμμικός	$k(\mathbf{x}, \mathbf{y}) = \mathbf{x}\mathbf{y}$	-
Ομογενής πολυωνυμικός	$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}\mathbf{y})^d$	$d \geq 2$
Μη ομογενής πολυωνυμικός	$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}\mathbf{y} + c)^d$	$d \geq 2, c > 0$
Γκαουσιανός	$k(\mathbf{x}, \mathbf{y}) = e^{-\frac{\ \mathbf{x}-\mathbf{y}\ ^2}{2\sigma^2}}$	$\sigma > 0$

Η επιλογή του ‘σωστού’ πυρήνα δεν είναι πάντα προφανής. Μετά από δοκιμές χρησιμοποιούμε αυτόν που μας δίνει τα πιο ικανοποιητικά αποτελέσματα. Όσο για την επιλογή των ελεύθερων παραμέτρων του πυρήνα, εφαρμόζουμε τις τεχνικές ρύθμισης παραμέτρων Cross validation.

Έτσι το πρόβλημα βελτιστοποίησης (5.6) μπορεί να γίνει πιο ευέλικτο χρησιμοποιώντας κάποια από τις συναρτήσεις πυρήνα που αναφέραμε προηγουμένως ώστε να μπορούμε να αναπαραστήσουμε πολύ γενικότερες υπερεπιφάνειες αποφάσεων. Με βάση το τέχνασμα του πυρήνα μπορούμε να αντικαταστήσουμε το εσωτερικό γινόμενο $\mathbf{x}_i \mathbf{x}_j$ με οποιαδήποτε συνάρτηση πυρήνα $k(\mathbf{x}_i, \mathbf{x}_j)$ ακολουθώντας την ίδια διαδικασία με την γραμμική περίπτωση που αναπτύχθηκε στην Παράγραφο 4.4.2.

5.2.4 Ταξινομητές Εξασθενημένου Περιθωρίου

Μέχρι τώρα έχουμε εξετάσει τις περιπτώσεις όπου τα δεδομένα εκπαίδευσης μπορούν να διαχωριστούν γραμμικά είτε στον αρχικό χώρο γνωρισμάτων είτε σε ένα μετασχηματισμένο μεγαλύτερης διάστασης χώρο. Το βασικό πρόβλημα που αντιμετωπίζουμε στον ταξινομητή μέγιστου περιθωρίου είναι πως δεν αφήνει την δυνατότητα λάθους ταξινόμησης στα παραδείγματα εκπαίδευσης. Παράγεται μια αυστηρή συνάρτηση απόφασης και εξασφαλίζει ότι τα υπερεπίπεδα υποστήριξης θα τοποθετηθούν με τρόπο τέτοιο ώστε να μεγιστοποιούν την απόστασή τους από την επιφάνεια απόφασης και να εφάπτονται στα όρια των κλάσεων. Στις περισσότερες πρακτικές εφαρμογές όμως υπάρχει θόρυβος στα δεδομένα εισόδου (πχ σφάλματα μέτρησης). Ο Ταξινομητής Εξασθενημένου Περιθωρίου (Soft Margin Classifier) αντιμετωπίζει περιπτώσεις θορύβου, ακραίων τιμών καθώς επιτρέπει στον SVM ταξινομητή να κάνει κάποια σφάλματα προσπαθώντας ταυτόχρονα να κρατήσει το περιθώριο όσο πιο πλατύ γίνεται.

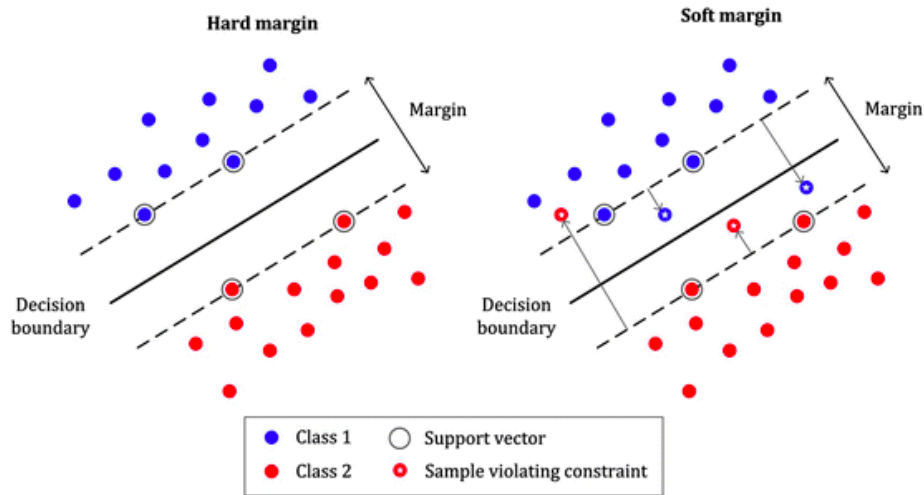
Πιο συγκεκριμένα, μπορούμε να αγνοήσουμε κάποια δεδομένα που ίσως είναι λάθος ταξινομημένα και τα οποία περιορίζουν το μέγεθος του περιθωρίου. Επιτρέπουμε λοιπόν κάποια σημεία εκπαίδευσης να είναι στο ‘λάθος’ ημιεπίπεδο με ποινή η οποία

αυξάνεται με την απόσταση από το σύνορο [46] [24][39]. Αναδιατυπώνουμε το πρόβλημα (5.3) ως εξής:

$$\min_{\mathbf{w}, b, \xi_i} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \quad (5.8)$$

$$\begin{aligned} \text{με περιορισμούς } g_i(\mathbf{w}) &= y_i(\mathbf{w}\mathbf{x}_i + b) \geq 1 - \xi_i \\ \xi_i &\geq 0, i = 1, 2, \dots, N. \end{aligned}$$

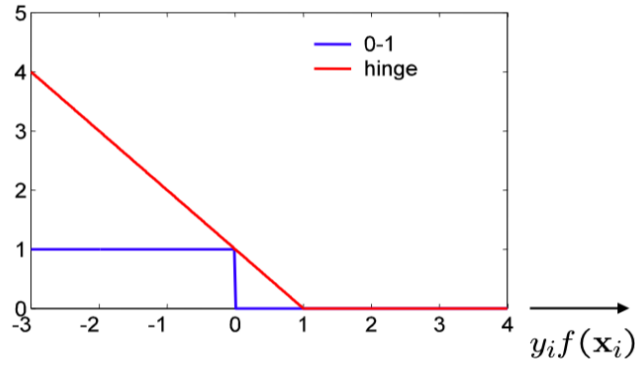
Για κάθε παράδειγμα \mathbf{x}_i εισάγουμε την λεγόμενη ‘χαλαρή’ μεταβλητή ξ_i . [30] Στην πραγματικότητα, τα ξ_i είναι όροι σφάλματος που μετρούν το πόσο απέχουν τα στιγμιότυπα που βρίσκονται στο λάθος ημιεπίπεδο από το υπερεπίπεδο υποστήριξης της κλάσης που ανήκουν. Με τον τρόπο αυτό επιτρέπουμε ορισμένα παραδείγματα \mathbf{x}_i να έχουν λάθος ετικέτα, πληρώνοντας όμως κάποιο κόστος $C\xi_i$. Η ιδέα αυτή απεικονίζεται στο Διάγραμμα 5.11.



Διάγραμμα 5.10: (Αριστερά) Ο αυστηρός ταξινομητής μέγιστου περιθωρίου. (Δεξιά) Ταξινομητής εξασθενημένου περιθωρίου.

Ο περιορισμός $g_i(\mathbf{w}) = y_i(\mathbf{w}\mathbf{x}_i + b) \geq 1 - \xi_i$ μπορεί να γραφεί και ως $y_i f(\mathbf{x}_i) \geq 1 - \xi_i$. Έχοντας υπόψη ότι $\xi_i \geq 0$ μπορούμε να γράψουμε ότι $\xi_i = \max(0, 1 - y_i f(\mathbf{x}_i))$ η οποία εκφράζει την συνάρτηση απώλειας, γνωστή ως hinge loss. Αν το $y_i f(\mathbf{x}_i) > 1$ τότε το σημείο βρίσκεται έξω από το περιθώριο και η απώλεια $\xi_i = 0$. Αν $y_i f(\mathbf{x}_i) = 1$ τότε το σημείο είναι πάνω στο υπερεπίπεδο υποστήριξης και επομένως πάλι η απώλεια είναι 0. Τέλος, αν $y_i f(\mathbf{x}_i) < 1$, το σημείο παραβιάζει το περιορισμό και έχουμε απώλεια. Στο Διάγραμμα 5.11 φαίνεται η συνάρτηση απώλειας hinge loss σε σύγκριση με την 0/1 συνάρτησης απώλειας. Παρατηρούμε ότι όσο μικρότερο της μονάδας είναι το γινόμενο $y_i f(\mathbf{x}_i) < 1$ τόσο περισσότερο η hinge loss συνάρτηση τιμωρεί τις προβλέψεις $f(\mathbf{x}_i)$.

Η παράμετρος C είναι μια ρυθμιστική παράμετρος που προσπαθεί να αντισταθμίσει το μέγεθος του περιθωρίου και το πλήθος των λάθος ταξινομήσεων [30][53]. Αν $C = 0$ τότε δεν επιτρέπεται να γίνει λάθος ταξινόμηση, (περίπτωση του αυστηρού ταξινομητή μέγιστου περιθωρίου) και το περιθώριο μεταξύ των υπερεπιπέδων υποστήριξης



Διάγραμμα 5.11: Hinge και 0/1 συναρτήσεις απώλειας.

είναι στενό. Στην περίπτωση αυτή εμφανίζεται το πρόβλημα της υπερμοντελοποίησης καθώς έχουμε υψηλή διακύμανση. Όσο η τιμή του C αυξάνει τόσο περισσότερες λάθος ταξινομήσεις γίνονται, αυξάνοντας το πλάτος του περιθωρίου. Πολύ μεγάλες τιμές του C οδηγούν σε μεγάλη μεροληψία. Η παράμετρος μπορεί να ρυθμιστεί χρησιμοποιώντας την τεχνική Cross validation. Προκειμένου να λύσουμε το πρόβλημα 5.8 θεωρούμε την Lagrangian συνάρτηση:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \mathbf{r}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i [y_i(\mathbf{x}_i \mathbf{w} + b) - 1 + \xi_i] - \sum_{i=1}^N r_i \xi_i,$$

όπου α_i, r_i είναι οι πολλαπλασιαστές Lagrange και ισχύει ότι $\alpha_i, r_i \geq 0$. Όπως και στην γραμμικά διαχωρίσιμη περίπτωση, για να δημιουργήσουμε το δυϊκό πρόβλημα θέτουμε τις παραγώγους ως προς τις μεταβλητές \mathbf{w}, b, ξ_i ίσες με 0.

$$\begin{aligned} \frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \mathbf{r}) &= \mathbf{w} - \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i = 0 \\ \Rightarrow \mathbf{w} &= \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \\ \frac{\partial}{\partial b} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \mathbf{r}) &= \sum_{i=1}^N \alpha_i y_i = 0 \\ \frac{\partial}{\partial \xi_i} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \mathbf{r}) &= C - \alpha_i - r_i = 0 \\ \Rightarrow \alpha_i &= C - r_i. \end{aligned}$$

Συνεπώς, η Lagrangian συνάρτηση παίρνει την μορφή:

$$\mathcal{L}(\boldsymbol{\alpha}) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \mathbf{x}_j).$$

Έτσι, το δυϊκό πρόβλημα έχει την μορφή:

$$\max_{\alpha} \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i x_j) \quad (5.9)$$

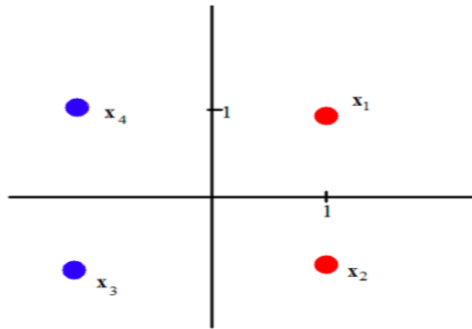
$$\text{με περιορισμούς } \sum_{i=1}^N \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C, , i = 1, \dots, N.$$

Παρατηρούμε ότι μοναδική διαφορά με το πρόβλημα βελτιστοποίησης που λύσαμε στην περίπτωση του αυστηρού ταξινομητή μέγιστου περιθωρίου (πρόβλημα 5.6) είναι ότι οι πολλαπλασιαστές Lagrange περιορίζονται από την τιμή της παραμέτρου C την οποία επιλέγουμε εκ των προτέρων.

Παράδειγμα

Ας θεωρήσουμε ένα πρόβλημα δύο κλάσεων (+1=κόκκινη, -1=μπλε) στον δυοδιάστατο χώρο με σύνολο εκπαίδευσης $S = \{[(1, 1), 1], [(1, -1), 1], [(-1, 1), -1], [(-1, -1), -1]\}$ όπως φαίνεται στο Διάγραμμα 5.12. Ας θεωρήσουμε ότι η παράμετρος C έχει τιμή $\frac{1}{4}$.



Διάγραμμα 5.12: Παράδειγμα δύο κλάσεων στο δυοδιάστατο χώρο.

Όπως δείξαμε, το πρόβλημα βελτιστοποίησης που καλούμαστε να λύσουμε είναι:

$$\max_{\alpha} \sum_{i=1}^4 \alpha_i - \frac{1}{2} \sum_{i=1}^4 \sum_{j=1}^4 \alpha_i \alpha_j y_i y_j \underbrace{(x_i x_j)}_{K(x_i, x_j)}$$

$$\text{με περιορισμούς } \sum_{i=1}^4 \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq \frac{1}{4}, , i = 1, \dots, N.$$

Ο γραμμικός πυρήνας που έχουμε επιλέξει παίρνει την μορφή:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \mathbf{x}_j = \begin{bmatrix} 2 & 0 & -2 & 0 \\ 0 & 2 & 0 & -2 \\ -2 & 0 & 2 & 0 \\ 0 & -2 & 0 & 2 \end{bmatrix}.$$

Παρατηρούμε ότι:

$$\sum_{i=1}^4 \sum_{j=1}^4 \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{i=1}^4 \alpha_i^2 + 2\alpha_1 \alpha_3 + 2\alpha_2 \alpha_4 = (\alpha_1 + \alpha_3)^2 + (\alpha_2 + \alpha_4)^2. \quad (5.10)$$

Αντικαθιστούμε την σχέση (5.10) στο πρόβλημα μεγιστοποίησης:

$$\begin{aligned} \max_{\alpha} \sum_{i=1}^4 \alpha_i - (\alpha_1 + \alpha_3)^2 - (\alpha_2 + \alpha_4)^2 \\ \text{με περιορισμούς } \sum_{i=1}^4 \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq \frac{1}{4}, i = 1, 2, 3, 4. \end{aligned}$$

Θέτοντας $u = \alpha_1 + \alpha_3$ και $v = \alpha_2 + \alpha_4$ έχουμε:

$$\sum_{i=1}^4 \alpha_i - (\alpha_1 + \alpha_3)^2 - (\alpha_2 + \alpha_4)^2 = u + v - u^2 - v^2.$$

Στην συνέχεια θέτουμε τις παραγώγους ως προς u, v ίσες με 0:

$$\begin{aligned} 1 - 2u = 0 \text{ και } 1 - 2v = 0 \\ \Rightarrow u = v = \frac{1}{2}. \end{aligned}$$

Αξιοποιώντας τους περιορισμούς καταλήγουμε στο ότι $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{1}{4}$. Το διάνυσμα \mathbf{w} γίνεται:

$$\mathbf{w} = \sum_{i=1}^4 \alpha_i y_i \mathbf{x}_i = \frac{1}{4}(x_1 + x_2 - x_3 - x_4) = \frac{1}{4}((4, 0)) = (1, 0).$$

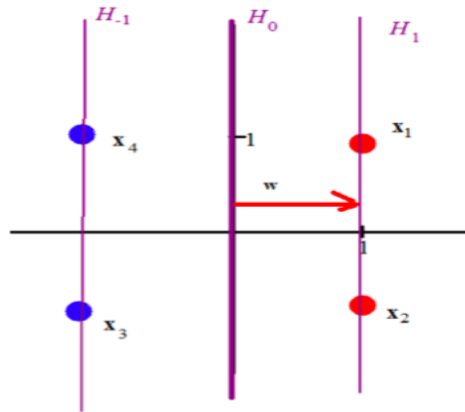
Το b , όπως έχουμε δει, υπολογίζεται ως εξής:

$$\begin{aligned} b^* &= \frac{b^+ + b^-}{2} \\ &= \frac{\max\{\mathbf{w}^* \mathbf{x} | (\mathbf{x}, y) \in S, y = +1\} + \max\{\mathbf{w}^* \mathbf{x} | (\mathbf{x}, y) \in S, y = -1\}}{2} \\ &= \frac{1 + (-1)}{2} \\ &= 0. \end{aligned}$$

Με βάση τα παραπάνω ισχύει:

$$f(\mathbf{x}) = \mathbf{w} \mathbf{x} + b = x_1.$$

Για το βέλτιστο υπερεπίπεδο H_0 θέτω $x_1 = 0$ ενώ τα υπερεπίπεδα υποστήριξης βρίσκονται για $x_1 = -1, x_1 = 1$ (Διάγραμμα 5.13).



Διάγραμμα 5.13: Βέλτιστο υπερίπεδο (H_0) και υπερίπεδα υποστήριξης (H_1, H_{-1}).

5.3 Τεχνητά Νευρωνικά Δίκτυα

5.3.1 Εισαγωγή

Μια εναλλακτική τεχνική Μηχανικής Μάθησης είναι τα Τεχνητά Νευρωνικά Δίκτυα/ ΤΝΔ (Artificial Neural Networks/ANN). Τα τελευταία χρόνια υπάρχει μια έκρηξη ενδιαφέροντος για αυτά καθώς έχουν σημειώσει επιτεύγματα σε ένα ευρύ φάσμα επιστημών, όπως η ιατρική, τα χρηματοοικονομικά, ρομποτική, επιστημή του μηχανικού, φυσική κ.α. Στην πραγματικότητα, τα ΤΝΔ εισάγονται οπουδήποτε τίθεται ζήτημα πρόβλεψης, αναγνώρισης, ταξινόμησης, ελέγχου.

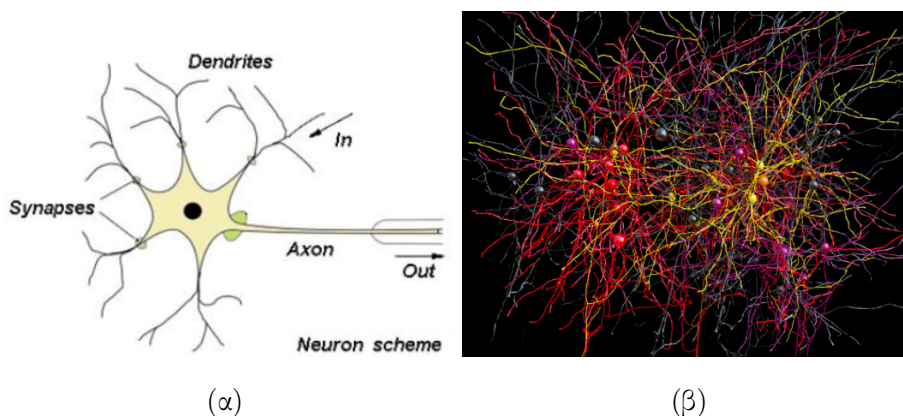
Τα Τεχνητά Νευρωνικά Δίκτυα χαρακτηρίζονται από την προσπάθειά τους να προσομοιώσουν την λειτουργία των βιολογικών Νευρωνικών Δικτύων με βάση κάποιο μαθηματικό μοντέλο. Εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα, ένα Νευρωνικό Δίκτυο είναι ένα κύκλωμα που αποτελείται από κόμβους (νευρώνες/ neurons) που συνδέονται μεταξύ τους. Πρόκειται για εξελιγμένες τεχνικές μη γραμμικής μοντελοποίησης, ικανές να μοντελοποιήσουν πολύπλοκες λειτουργίες. Τις περισσότερες φορές που η γραμμική προσέγγιση αποτυγχάνει, τα Νευρωνικά Δίκτυα δίνουν λύσεις, επιτρέποντας τη μη γραμμικότητα μέσω των μη γραμμικών συναρτήσεων ενεργοποίησης.

5.3.2 Βιολογική έμπνευση

Η έμπνευση των Τεχνητών Νευρωνικών Δικτύων ξεκινά από την βιολογία. Το νευρικό σύστημα των ανθρώπων αποτελείται από πολλά Νευρωνικά Δίκτυα που εξειδικεύονται στις καθημερινές διεργασίες, όπως η συναναστροφή των ατόμων με το περιβάλλον, μάθηση, μνήμη κ.τ.λ. Ένας τυπικός βιολογικός νευρώνας αποτελείται από τους δενδρίτες που είναι οι είσοδοι του νευρώνα μέσω των οποίων λαμβάνονται ηλεκτρικά σήματα από άλλα νευρικά κύτταρα, τον άξονα δηλαδή η έξοδος του νευρώνα μέσω του οποίου διοχετεύονται ηλεκτρικά σήματα και τις συνάψεις. Οι τελευταίες αναφέρονται στην ενδοεπικοινωνία των νευρώνων, δηλαδή στις διακλαδώσεις του άξονα, μέσω των οποίων

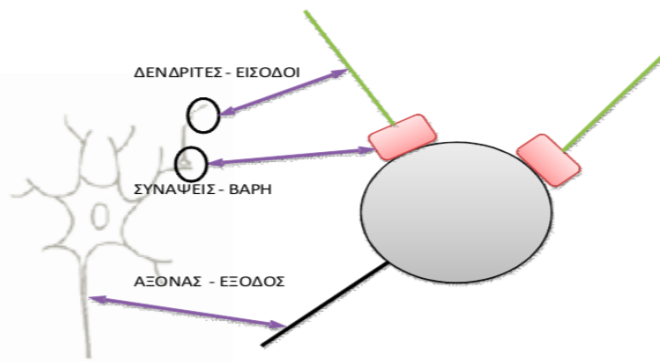
ο νευρώνας ενώνεται με τους δενδρίτες άλλων νευρικών κυττάρων (Διάγραμμα 5.14 α'). Κάθε νευρώνας έχει πολλούς δενδρίτες με πολλές διακλαδώσεις και μπορεί να είναι ενεργός ή αδρανής. Στην πρώτη περίπτωση παράγει ηλεκτρικούς παλμούς που διαδίδονται κατά μήκος του άξονα του και μεταβιβάζονται μέσω των συνάψεων στους δενδρίτες άλλων νευρώνων, διεγείρωντάς τους.

Το σπουδαιότερο και μεγαλύτερο τμήμα του Κεντρικού Νευρικού Συστήματος αποτελεί ο εγκέφαλος ο οποίος επίσης αποτελείται από πολλά νευρωνικά δίκτυα. Η μάθηση είναι η ικανότητα του εγκεφάλου να προσαρμόζει την λειτουργία του ώστε να επιλύει καθημερινά προβλήματα που σχετίζονται με τα ερεθίσματα που δέχεται (αναγνώριση εικόνας, ομιλίας κτλ). Η μάθηση επιτυγχάνεται μέσω της εκπαίδευσης, δηλαδή της διαδικασίας που σταδιακά προσαρμόζονται οι συναπτικές συνδέσεις των νευρώνων. Κατά την διαδικασία της εκπαίδευσης οι συναπτικές συνδέσεις ενισχύονται ή αποδυναμώνονται ώστε τελικά ο εγκέφαλος να λύσει το πρόβλημα. Η διαδικασία αυτή είναι εξαιρετικά δύσκολη καθώς στον ανθρώπινο εγκέφαλο υπάρχουν περίπου 100 δισεκατομμύρια νευρώνες, κάθε ένας από τους οποίους μπορεί να δημιουργήσει χιλιάδες συναπτικές συνδέσεις με άλλους νευρώνες (Διάγραμμα 5.14 β').



Διάγραμμα 5.14: (α') Απεικόνιση ενός τυπικού νευρώνα. (β') Πλήθος νευρώνων εγκεφάλου διασυνδεδεμένοι μεταξύ τους.

Τα Τεχνητά Νευρωνικά Δίκτυα αποτελούν μια προσπάθεια προσέγγισης των βιολογικών Νευρωνικών Δικτύων εκτελώντας υπολογισμούς με μαζικό, παράλληλο τρόπο. Οι τεχνητοί νευρώνες αναφέρονται και ως μονάδες επεξεργασίας (nodes) και έχουν συνδέσεις που τους επιτρέπουν να δέχονται και να στέλνουν σήματα. Κάθε νευρώνας λαμβάνει ταυτόχρονα πολλές εισόδους, κάθε μια από τις οποίες έχει το δικό της βάρος. Τα βάρη αυτά λειτουργούν όπως και οι συναπτικές δυνάμεις των βιολογικών νευρώνων. Και στις δύο περιπτώσεις κάποια σήματα είναι περισσότερο σημαντικά από κάποια άλλα με αποτέλεσμα όταν συνδιάζονται για να παράγουν ένα αποτέλεσμα, να έχουν μεγαλύτερη επίδραση στον νευρώνα. Συνεπώς, τα βάρη δείχνουν πόσο ισχυρές είναι οι συνδέσεις μεταξύ των νευρώνων, δηλαδή το πόσο ισχυρό είναι το σήμα που μεταφέρεται. Τα συναπτικά βάρη (synaptic weights) είναι προσαρμοστικοί συντελεστές και τροποποιούνται κατά της διαδικασία εκπαίδευσης του δικτύου. Στο Διάγραμμα 5.15 φαίνονται οι συσχετίσεις του βιολογικού με τον τεχνητό νευρώνα, όπου οι δενδρίτες αντιστοιχούν στις εισόδους, οι συνάψεις στα βάρη και ο άξονας στην έξοδο του τεχνητού νευρώνα.



Διάγραμμα 5.15: Αντιστοιχία του βιολογικού σε τεχνητό νευρώνα.

5.3.3 Μοντέλο τεχνητού νευρώνα

Όπως έχουμε αναφέρει, τα δομικά στοιχεία ενός δικτύου είναι οι νευρώνες οι οποίοι διακρίνονται σε νευρώνες εισόδου, υπολογιστικούς/ κρυφούς και εξόδου. Οι πρώτοι δέχονται σαν είσοδο δεδομένα από το περιβάλλον και τα διοχετεύουν στους υπολογιστικούς νευρώνες χωρίς να επιτελούν κάποιο υπολογισμό. Το πλήθος των νευρώνων εισόδου είναι τόσο, όσες και οι τιμές των χαρακτηριστικών X^1, X^2, \dots, X^l . Οι υπολογιστικοί νευρώνες ακολουθούν την παρακάτω διαδικασία. Κάθε ένας από αυτούς δέχεται έναν αριθμό εισόδων. Κάθε εισόδος πολλαπλασιάζεται με το αντίστοιχο συναπτικό βάρος που δηλώνει πόσο στενά είναι συνδεδεμένοι οι δύο νευρώνες που συνδέονται. Στην συνέχεια, υπολογίζεται το ολικό άθροισμα των γινομένων όλων των εισόδων-βαρών του νευρώνα. Το άθροισμα αυτό δίνεται ως όρισμα στην συνάρτηση ενεργοποίησης (activation function) που έχει οριστεί. Η τιμή που λαμβάνει η συνάρτηση για το συγκεκριμένο όρισμα είναι η έξοδος του νευρώνα και παίρνει τιμές στο διάστημα $[0,1]$ ή $[-1,1]$ ανάλογα με την επιλογή της συνάρτησης. Η τιμή αυτή τροφοδοτείται στους κόμβους εξόδου οι οποίοι διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου.

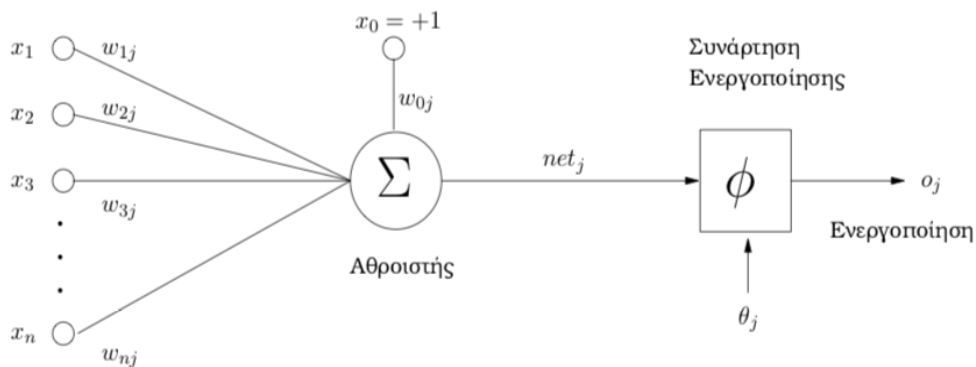
Κάθε κόμβος εξόδου αντιστοιχεί και σε μια κατηγορία και η τιμή που εξέρχεται από αυτόν εκφράζει την πιθανότητα η τιμή της εισόδου να ανήκει στην κατηγορία αυτή. Στην δυαδική ταξινόμηση έχουμε έναν νευρώνα εξόδου. Γνωρίζοντας ότι το αποτέλεσμα της ταξινόμησης για οποιαδήποτε παρατήρηση μπορεί να είναι $y = \omega_1 = 0$ ή $y = \omega_2 = 1$, αν ο νευρώνας εξόδου παράγει τιμή μικρότερη του 0.5 τότε το παράδειγμα ταξινομείται στην κλάση ω_1 , διαφορετικά στην κλάση ω_2 . Στην περίπτωση που το πρόβλημα μας έχει πάνω από τρεις κλάσεις τότε το Νευρωνικό Δίκτυο που χρησιμοποιούμε έχει τόσους νευρώνες όσες και οι κατηγορίες. Στην περίπτωση αυτή, οι ετικέτες των παραδειγμάτων εκπαίδευσης πρέπει να έχουν την μορφή διανυσμάτων διάστασης ίσης με τον αριθμό των κλάσεων. Έστω ότι θέλουμε να κατηγοριοποιήσουμε μια φωτογραφία με βάση το περιεχόμενό της, έχοντας στην διάθεση μας τις κλάσεις πεζός, αυτοκίνητο, μοτοσικλέτα. Κάθε παράδειγμα x_i (π.χ. πεζός) θα αντιστοιχεί σε ένα $y_i = [1, 0, 0]$. Η αποτέλεσμα του ταξινομητή για ένα καινούργιο παράδειγμα θα είναι επίσης ένα διάνυσμα. Τελικά, η προβλεπόμενη κλάση είναι αυτή της οποίας ο αντίστοιχος νευρώνας (στοιχείο του διανύσματος) έχει την μεγαλύτερη τιμή.

Στην συνέχεια, θα περιγράψουμε το μοντέλο ενός τεχνητού νευρώνα, έστω

j , χρησιμοποιώντας μαθηματική ορολογία. Οι εισόδους πλήθους l του j -οστού νευρώνα που συμβολίζονται με $x_{ji}, i = 1, \dots, l$ χαρακτηρίζονται από βάρη w_{ji} . Το βάρος w_{ji} αναφέρεται στην σύνδεση της εισόδου x_{ji} με τον j -οστό νευρώνα. Έστω ότι net_j συμβολίζουμε το άθροισμα των γινομένων εισόδων-βαρών του νευρώνα j και ϕ είναι η συνάρτηση ενεργοποίησης που χρησιμοποιούμε τότε η έξοδος, o_j , του νευρώνα θα δίνεται από την σχέση:

$$o_j = \phi(net_j) = \phi \left(\sum_{i=0}^l x_{ji}w_{ji} \right) = \phi \left(\sum_{i=1}^l x_{ji}w_{ji} + x_{j0}w_{j0} \right). \quad (5.11)$$

Παρατηρούμε ότι στον j -οστό νευρώνα προσθέτουμε ένα νέο σήμα εισόδου με τιμή σταθερή, $x_{j0} = 1$ και συναπτικό βάρος $w_{j0} = \theta_j$. Το βάρος αυτό καλείται κατώφλι/πόλωση (threshold/bias). Αν το συνολικό άθροισμα όλων των υπολοίπων εισόδων του νευρώνα είναι μεγαλύτερο από την τιμή $x_{j0}w_{j0}$ τότε ο νευρώνας ενεργοποιείται. Εάν είναι μικρότερο τότε ο παραμένει ανενεργός [17][8]. Συνεπώς, η συμπεριφορά ενός νευρώνα περιγράφεται από δύο τελεστές, έναν γραμμικό και ένα μη γραμμικό οι οποίοι είναι συνδεδεμένοι στην σειρά. Ο γραμμικός παριστάνεται σαν το άθροισμα των γινομένων εισόδων-βαρών και ο μη-γραμμικός σαν την μη γραμμική συνάρτηση της εξόδου του γραμμικού τελεστή. Στο Διάγραμμα 5.16 φαίνονται τα βασικά στοιχεία ενός νευρώνα που περιγράφηκαν παραπάνω.



Διάγραμμα 5.16: Μοντέλο τεχνητού νευρώνα.

Συναρτήσεις ενεργοποίησης

Υπάρχουν διαφορετικές συναρτήσεις ενεργοποίησης. Σε αυτή την Παράγραφο θα περιγράψουμε την γραμμική, την βηματική (step function), τη συνάρτηση προσήμου (sign function), τη δυαδική σιγμοειδής (binary sigmoid και την διπολική σιγμοειδής συνάρτηση (bipolar sigmoid) Διάγραμμα 5.17 [17][20].

Ως γραμμική συνάρτηση ενεργοποίησης μπορούμε να χρησιμοποιήσουμε την ταυτοτική, $\phi(x) = x$ ή οποιαδήποτε άλλη γραμμική. Η βηματική συναρτηση, αναφέρεται ως συνάρτηση Heaviside και περιγράφεται από την σχέση:

$$\phi(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0. \end{cases}$$

Σύμφωνα με αυτή την συνάρτηση ενεργοποίησης, η έξοδος του νευρώνα j , o_j , θα είναι 1 στην περίπτωση που το άθροισμα $net_j = \sum_{i=0}^l x_{ij}w_{ij}$ είναι μη αρνητικό και 0 διαφορετικά. Δηλαδή:

$$o_j = \phi(net_j) = \begin{cases} 1 & net_j \geq 0 \\ 0 & net_j < 0. \end{cases}$$

Ο νευρώνας που χρησιμοποιεί την βηματική συνάρτηση ενεργοποίησης αναφέρεται ως το μοντέλο McCulloch–Pitts (MCP). Μια εναλλακτική συνάρτηση ενεργοποίησης είναι η συνάρτηση προσήμου. Έχει θετική (+1)/αρνητική (-1) έξοδο αν το όρισμα της, u_k , είναι μη αρνητικό/αρνητικό:

$$o_j = \phi(net_j) = \begin{cases} 1 & net_j \geq 0 \\ -1 & net_j < 0. \end{cases}$$

Ωστόσο, οι συναρτήσεις ενεργοποίησης που περιγράψαμε δεν θεωρούνται χρήσιμες στα τεχνητά νευρωνικά δίκτυα καθώς έχουν το βασικό μειονέκτημα ότι η παράγωγος απειρίζεται. Έτσι, προέκυψε η ανάγκη για συναρτήσεις ενεργοποίησης που η γραφική παράσταση τους μοιάζει με τη βηματική ή προσήμου αλλά ταυτόχρονα να είναι συνεχείς και παραγωγίσιμες σε όλο το πεδίο ορισμού τους. Τέτοιες συναρτήσεις είναι η δυαδική και διπολική σιγμοειδής [6][17]. Η δυαδική σιγμοειδής συνάρτηση ενεργοποίησης ορίζεται ως εξής:

$$o_j = \phi(net_j) = \frac{1}{1 + e^{-\sigma net_j}},$$

και η διπολική σιγμοειδής συνάρτηση ως:

$$o_j = \phi(net_j) = \tanh\left(\frac{\sigma}{2} net_j\right).$$

Και στις δύο περιπτώσεις η τιμή της παραμέτρου σ είναι θετική. Όσο αυξάνει η τιμή της τόσο πιο απότομες είναι οι αλλαγές στις τιμές της $\phi(net_j)$. Το πεδίο ορισμού της πρώτης είναι το $[0,1]$ ενώ της δεύτερης το $[-1,1]$ καθώς το όρισμα net_j παίρνει τιμές από $-\infty$ ως το $+\infty$. Σε αντίθεση με την βηματική και την προσήμου, οι δυαδικές και διπολικές σιγμοειδής συναρτήσεις είναι ομαλές και οι παράγωγοι αυτών είναι επίσης ομαλές. Για την δυαδική:

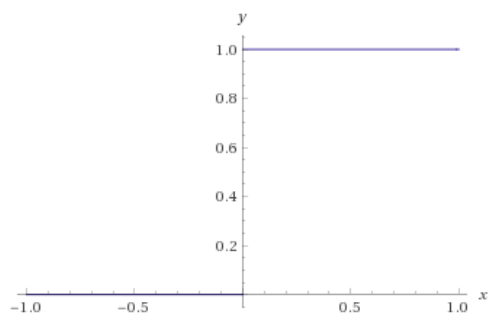
$$o'_j = \phi'(net_j) = \sigma \phi(net_j)[1 - \phi(net_j)],$$

και για την διπολική:

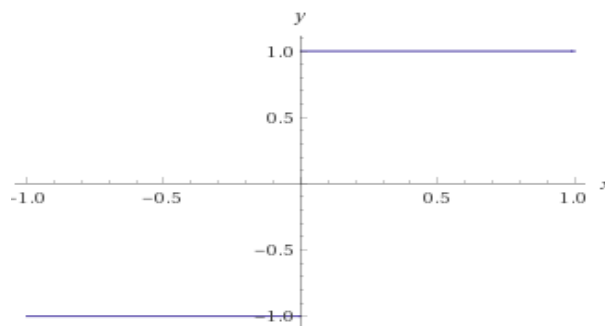
$$o'_j = \phi'(net_j) = \frac{\sigma}{2}[1 + \phi(net_j)][1 - \phi(net_j)].$$

5.3.4 Αριχτεκτονική δικτύων

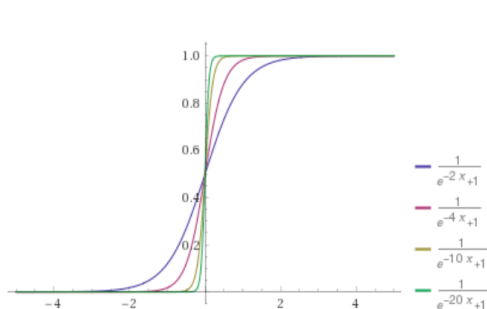
Ένα από τα βασικά χαρακτηριστικά ενός Τεχνητού Νευρωνικού Δικτύου είναι η αρχιτεκτονική του, η οποία σχετίζεται με την διάταξη, το πλήθος των νευρώνων και την συνάρτηση ενεργοποίησης που αυτοί χρησιμοποιούν. Σε κάθε περίπτωση, οργανώνονται σε επίπεδα (layers) και οι νευρώνες κάθε ενός από αυτά χρησιμοποιούν σχεδόν



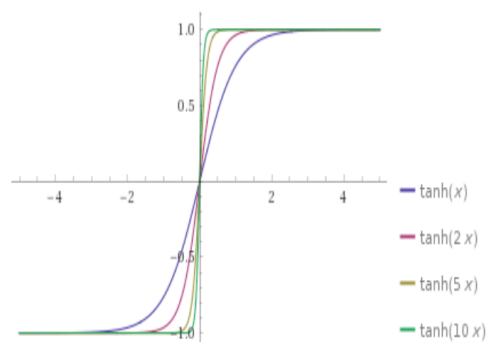
(α) Βηματική συνάρτηση.



(β) Συνάρτηση προσήμου.



(γ) Δυαδική σιγμοειδής συνάρτηση για διάφορες τιμές της παραμέτρου σ .

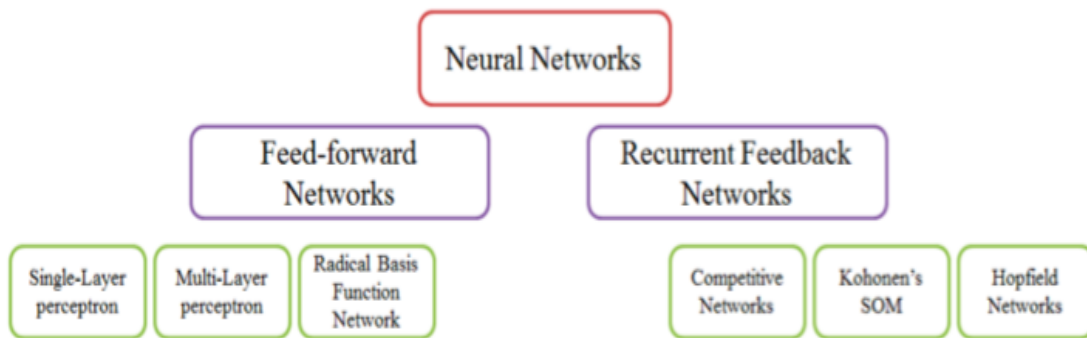


(δ) Διπολική σιγμοειδής συνάρτηση για διάφορες τιμές της παραμέτρου σ .

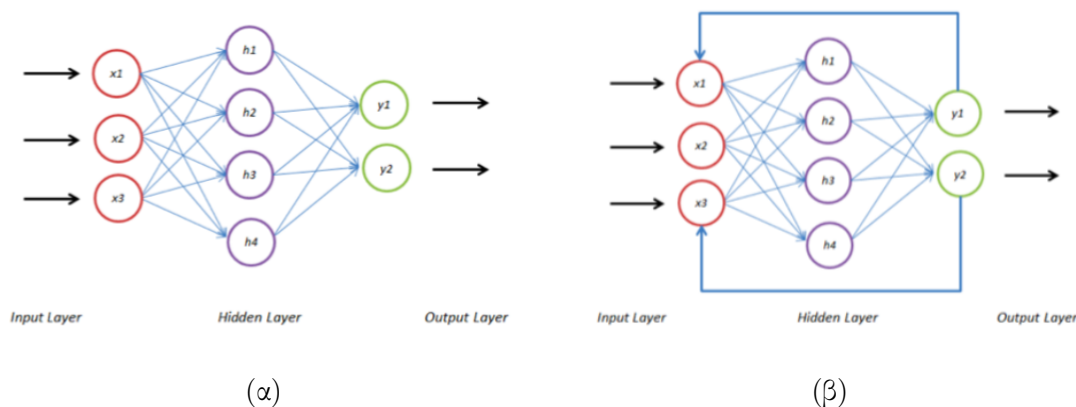
Διάγραμμα 5.17: Συναρτήσεις ενεργοποίησης. Για μεγάλες τιμές του σ η δυαδική σιγμοειδής συνάρτηση προσεγγίζει την βηματική και η διπολική σιγμοειδής της συνάρτηση προσήμου.

πάντα την ίδια συνάρτηση ενεργοποίησης και συμπεριφέρονται με τον ίδιο τρόπο. Όπως είδαμε, τα επίπεδα ή αλλιώς στρώματα διακρίνονται σε επίπεδα εισόδου (input layers) που έχουν τόσους νευρώνες όσες και οι διαφορετικές τιμές εισόδου ενός ANN, στα κρυφά επίπεδα (hidden layers) κάθε ένα από τα οποία μπορεί να έχει r νευρώνες διασυνδεδεμένους με διάφορους τρόπους με τα υπόλοιπα κρυφά στρώματα ή το στρώμα εξόδου. Τέλος, το επίπεδο εξόδου έχει τόσους νευρώνες (output layers) όσες και οι μεταβλητές εξόδου του ANN [43][8].

Μια πρώτη διάκριση των δικτύων ως προς την αρχιτεκτονική τους αφορά την σύνδεση των νευρώνων. Αυτοί μπορεί να είναι πλήρως συνδεδεμένοι (fully connected) όπου κάθε ένας νευρώνας συνδέεται με όλους τους νευρώνες του επόμενου επιπέδου, σε διαφορετική περίπτωση ονομάζονται μερικώς συνδεδεμένοι (partially connected). Ένας εναλλακτικός τρόπος κατηγοριοποίησης των ANN αφορά την φορά και τον τρόπο διάδοσης πληροφοριών μεταξύ των νευρώνων. Έτσι, έχουμε τα Δίκτυα Πρόσθιας τροφοδότησης (Feed-Forward Neural Networks) και τα Αναδρομικά Δίκτυα (Recurrent Neural Networks). Στο Διάγραμμα 5.18 φαίνονται οι δύο αυτές κατηγορίες αρχιτεκτονικής και οι βασικοί τύποι Νευρωνικών Δικτύων.



Διάγραμμα 5.18: Αρχιτεκτονικές δικτύου και οι βασικοί τύπου Νευρωνικών Δικτύων.



Διάγραμμα 5.19: (α') Αρχιτεκτονική Νευρωνικών Δικτύων Πρόσθιας Τροφοδότησης και (β') Αναδρομικών Δικτύων.

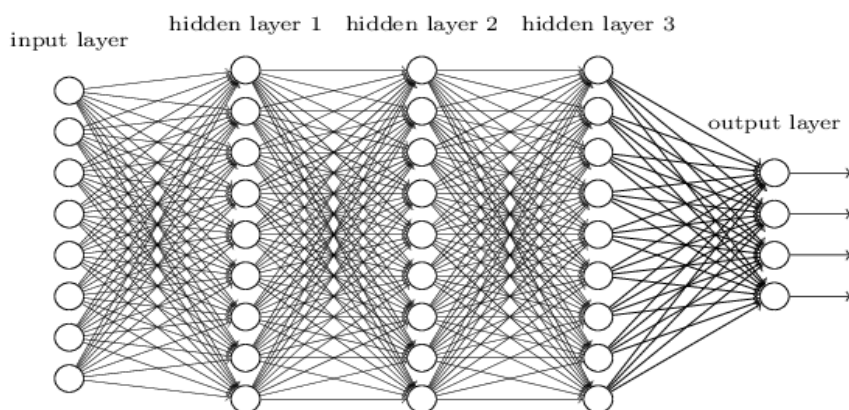
Στα Νευρωνικά Δίκτυα Πρόσθιας τροφοδότησης το σήμα/ πληροφορίες κινούνται προς μια κατεύθυνση (προς τα εμπρός). Η διαδικασία ξεκινά από τους κόμβους εισόδου, συνεχίζει στους υπολογιστικούς κόμβους, αν υπάρχουν, και τερματίζει στους νευρώνες εξόδου. Συνεπώς, δεν υπάρχει νευρώνας που η έξοδος του είναι είσοδος σε νευρώνα προηγούμενου στρώματος ή ακόμα και του ίδιου στρώματος (Διάγραμμα 5.19 α'). Αντίθετα, όταν υπάρχουν συνδέσεις μεταξύ νευρώνων ενός επιπέδου και νευρώνων προηγούμενου επιπέδου τότε δίκτυα αυτά καλούνται Ανδρομικά (Διάγραμμα 5.19 β').

5.3.5 Νευρωνικά Δίκτυα Πρόσθιας τροφοδότησης

Στην Παράγραφο αυτή θα ασχοληθούμε με τα Νευρωνικά Δίκτυα Πρόσθιας τροφοδότησης που αποτελεί το πιο συνηθισμένο και διαδεδομένο είδος Νευρωνικών Δικτύων και είναι αυτά που χρησιμοποιούνται περισσότερο σε προβλήματα ταξινόμησης. Πιο συγκεκριμένα, θα αναλύσουμε τα δύο πιο σημαντικά δίκτυα αυτής της κατηγορίας, το μονό επίπεδο perceptron και τα πολυεπίπεδα perceptron.

Το δίκτυο perceptron ενός επιπέδου ή μονοεπίπεδο perceptron (single layer perceptron/SLP) είναι το πιο απλό είδος Τεχνητού Νευρωνικού Δικτύου Πρόσθιας τροφοδότησης και αποτελεί ιστορικά μια πρώτη προσέγγιση των ΤΝΔ. Χαρακτηρίζεται ως γραμμικός ταξινομητής καθώς αποφασίζει για την κλάση ενός παραδείγματος με βάση την τιμή ενός γραμμικού συνδιασμού των γνωρισμάτων του. Στην περίπτωση αυτή, δεν υπάρχουν υπολογιστικοί νευρώνες αλλά μόνο ένα στρώμα κόμβων εξόδου [20][2]. Συνεπώς, οι εισόδοι τροφοδοτούνται απευθείας στις εξόδους μέσω μιας σειράς βαρών. Το άθροισμα των γινομένων των βαρών-εισόδων υπολογίζεται σε κάθε κόμβο και αν η τιμή είναι πάνω από το κατώφλι τότε ο νευρώνας ενεργοποιείται και παίρνει την ενεργοποιημένη τιμή +1 διαφορετικά παίρνει απενεργοποιημένη τιμή -1 ή 0. Στις περισσότερες περιπτώσεις που το μονοεπίπεδο perceptron λειτουργεί ως δυαδικός ταξινομητής έχει την μορφή που φαίνεται στο Διάγραμμα 5.16. Μπορεί όμως να χρησιμοποιηθεί και σε προβλήματα περισσότερων κλάσεων έχοντας στο στρώμα εξόδου περισσότερους νευρώνες.

Μεγαλύτερα και περισσότερο πολύπλοκα δίκτυα προσφέρουν μεγαλύτερη ικανότητα υπολογισμών. Επίσης, έχουν δυνατότητες περισσότερες από αυτές των μονοεπίπεδων perceptron, σημαντικότερη από τις οποίες ο μη γραμμικός διαχωρισμός των παραδειγμάτων, όπως θα δούμε παρακάτω. Τα πολυεπίπεδα perceptrons (multilayer perceptron/MLPs) αποτελούνται από ένα σύνολο κόμβων εισόδου, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Κάθε επίπεδο μπορεί να έχει οποιοδήποτε αριθμό νευρώνων και κάθε νευρώνας στο δίκτυο περιλαμβάνει μια μη γραμμική διαφορίσιμη συνάρτηση ενεργοποίησης. Ο αριθμός των νευρώνων εισόδου και εξόδου εύκολα καθορίζεται από το πρόβλημα. Όπως έχουμε αναφέρει, ο αριθμός των νευρώνων εισόδου είναι όσα και τα χαρακτηριστικά που έχουμε επιλέξει και νευρώνες εξόδου είναι τόσος, όσες και οι κατηγορίες του προβλήματος [8] [22]. Δεν υπάρχει κάποιος κανόνας για τον αριθμό των κρυμμένων στρώματων ενός Νευρωνικού Δικτύου. Το πιο σύνηθες είναι να χρησιμοποιούμε ένα κρυφό επίπεδο με κρυφούς νευρώνες περισσότερους από αυτούς που υπάρχουν στο επίπεδο εισόδου. Βέβαια, πολυπλοκότερα προβλήματα απαιτούν περισσότερα από ένα κρυφά στρώματα [22]. Τα δίκτυα αυτά ονομάζονται βαθιά Τεχνητά Νευρωνικά Δίκτυα (deep neural nets) και έχουν την μορφή του Διαγράμματος 5.20.



Διάγραμμα 5.20: Βαθύ Τεχνητό Νευρωνικό Δίκτυο με 3 κρυφά στρώματα νευρώνων.

Γραμμική διαχωρισιμότητα

Η χρήση του απλού perceptron έχει περιορισμένες δυνατότητες, καθώς με ένα νευρώνα αναπαρίσταται ένα υπερεπίπεδο ικανό να διαχωρίσει γραμμικά τις κλάσεις. Τα πιο πολλά προβλήματα δεν είναι γραμμικά διαχωρίσιμα και τα λίγα προβλήματα που είναι, έχουν λυθεί με άλλους τρόπους ευκολότερα. Ο περιορισμός αυτός αντιμετωπίζεται με τα πολυεπίπεδα perceptron που χρησιμοποιούν τουλάχιστον ένα κρύφο επίπεδο με μη γραμμικούς νευρώνες ικανά να λύσουν γραμμικά μη διαχωρήσιμα προβλήματα. Διαφορετικά, αν οι νευρώνες χρησιμοποιούσαν γραμμική συνάρτηση ενεργοποίησης, τότε το ΤΝΔ θα έχει ως αποτέλεσμα μια γραμμική απεικόνιση και θα μπορούσε να αντικατασταθεί με το απλό perceptron [8][22][43]. Τα MLPs χρησιμοποιούν την σιγμοειδή συνάρτηση ενεργοποίησης καθώς όπως θα δούμε οι κανόνες εκπαίδευσης που εφαρμόζονται, βασίζονται σε κανόνες βελτιστοποίησης και χρησιμοποιούν παραγώγους.

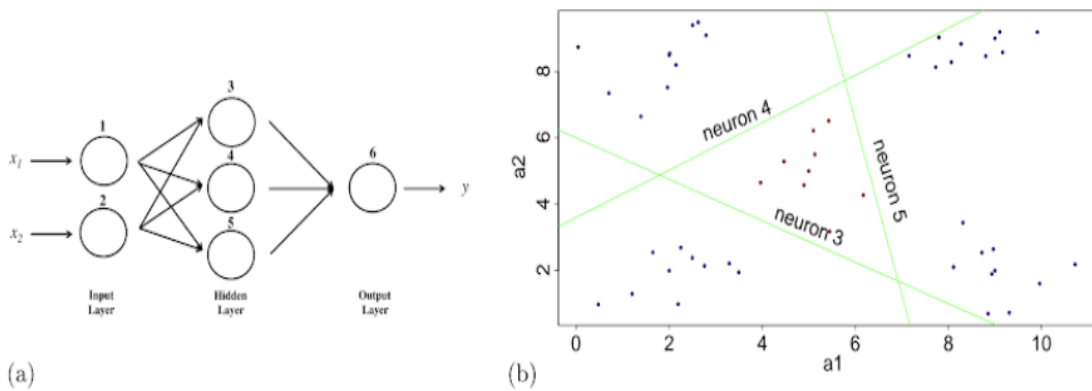
Χρησιμοποιώντας περισσότερα επίπεδα νευρώνων το μοντέλο μπορεί να ξεχωρίσει σημεία που περιλαμβάνονται σε διακριτές περιοχές. Έστω ότι έχουμε δύο εισόδους (x, y) και έναν νευρώνα στο επίπεδο εξόδου. Αν το κρυφό επίπεδο αποτελούταν από δύο νευρώνες τότε κάθε ένας νευρώνας θα δημιουργούσε δύο ευθείες, αν είχαμε 3 νευρώνες τότε θα σχηματιζόνταν τρεις ευθείες στον χώρο των χαρακτηριστικών. Για περισσότερους νευρώνες δημιουργούνται πολύγωνα. Η ιδέα αυτή απεικονίζεται στα Διαγράμματα 5.21, 5.22 όπου παρατηρούμε ότι κάθε νευρώνας του πρώτου επιπέδου αντιστοιχεί σε μια ευθεία στον δυδιάστατο χώρο χαρακτηριστικών [43]. Στο δεύτερο κρυφό στρώμα του Διαγράμματος 5.22 οι νευρώνες θα σχηματίζουν υπερεπίπεδα στον δεκα-διάστατο χώρο χαρακτηριστικών. Το αποτέλεσμα του δικτύου 5.21 είναι:

$$o_6 = \phi_6(w_{63} \underbrace{\phi_3(w_{31}x_1 + w_{32}x_2 + \theta_3)}_{o_3} + w_{64} \underbrace{\phi_4(w_{41}x_1 + w_{42}x_2 + \theta_4)}_{o_4} + w_{65} \underbrace{\phi_5(w_{51}x_1 + w_{52}x_2 + \theta_5)}_{o_5} + \theta_6).$$

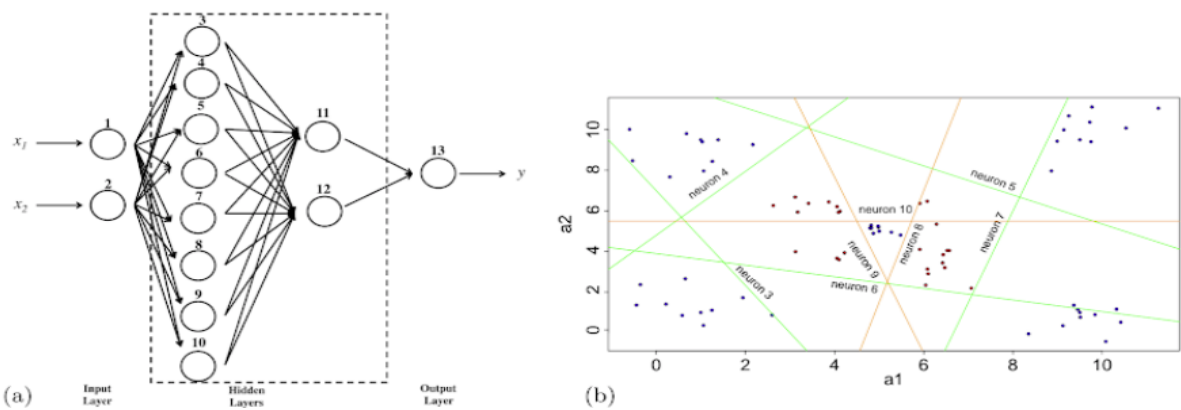
Όμοια υπολογίζουμε και την έξοδο του προβλήματος 5.22. Γίνεται εύκολα αντιληπτό ότι αν δεν υπήρχε κρυφό στρώμα τότε το αποτέλεσμα του δικτύου θα ήταν μια ευθεία ($w\mathbf{x} = 0$) η οποία θα ήταν αδύνατο να λύσει τα προβλήματα 5.21, 5.22 αλλά θα ήταν κατάλληλο για ένα πρόβλημα όπως αυτό του Διαγράμματος 5.23 όπου τα δεδομένα είναι γραμμικά διαχωρίσιμα.

5.3.6 Εκπαίδευση

Για να επιλύσουμε ένα πρόβλημα ταξινόμησης μάθησης με επίβλεψη χρησιμοποιώντας τα Νευρωνικά Δίκτυα, ακολουθούμε την παρακάτω διαδικασία. Αρχικά, ο χρήστης καθορίζει την τοπολογία του δικτύου, δηλαδή τον αριθμό των κρυμμένων στρωμάτων, τα αρχικά βάρη των νευρώνων, τον τρόπο που μεταβιβάζεται η πληροφορία σε αυτούς (οργάνωση συνάψεων) και τις συναρτήσεις που θα χρησιμοποιηθούν από τον γράφο.



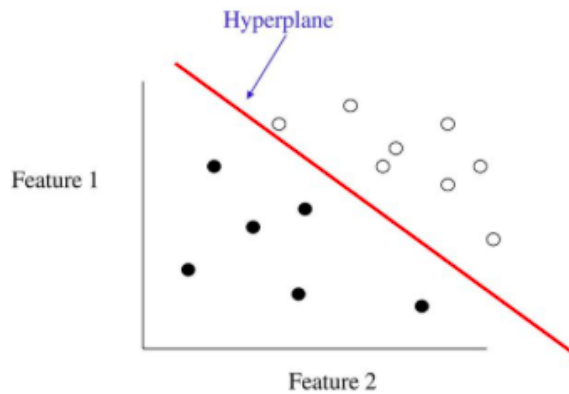
Διάγραμμα 5.21: (a) Παράδειγμα πολυεπίπεδου ΝΔ με ένα κρυφό στρώμα που αποτελείται από τρεις νευρώνες. (b) Ταξινόμηση βάση διαχωρισμού του χώρου χαρακτηριστικών σε διακριτές περιοχές. Κάθε νευρώνας αντιστοιχεί και σε μια ευθεία.



Διάγραμμα 5.22: (a) Παράδειγμα πολυεπίπεδου ΝΔ με δύο κρυφά στρώματα που αποτελούνται από 10 και 2 νευρώνες αντίστοιχα. (b) Αναπαράσταση των ευθειών που σχηματίζουν οι 10 νευρώνες του πρώτου κρυφού στρώματος.

Τα αρχικά βάρη μπορεί να είναι μικροί πραγματικοί αριθμοί ή να προέρχονται από την ομοιόμορφη κατανομή $[-1,1]$. Στην συνέχεια, αξιοποιώντας τα παραδείγματα εκπαίδευσης και χρησιμοποιώντας κάποιον αλγόριθμο εκπαίδευσης τροποποιούμε τα συναπτικά βάρη διορθώνοντας το σφάλμα που υπάρχει, ώστε να έχουμε το επιθυμητό αποτέλεσμα. Λέμε ότι ολοκληρώνουμε έναν κύκλο εκπαίδευσης όταν έχουμε χρησιμοποιήσει διαδοχικά όλα τα κατηγοριοποιημένα παραδείγματα. Κατά την εκπαίδευση ένα Νευρωνικό Δίκτυο εκτελεί πολλούς κύκλους προκειμένου το σύστημα να έχει την απαιτούμενη συμπεριφορά.

Συνεπώς, ο αλγόριθμος εκπαίδευσης σκοπεύει να μεταβάλλει τις ελεύθερες παραμέτρους ώστε να βελτιστοποιεί την έξοδο. Στις επόμενες Παραγράφους παρουσιάζουμε κάποιους βασικούς αλγόριθμους εκπαίδευσης που χρησιμοποιούν τα Τεχνητά Νευρωνικά Δίκτυα. Πιο συγκεκριμένα, θα αναλύσουμε τον αλγόριθμο οπισθοδιάδοσης του λάθους που χρησιμοποιείται στην εκπαίδευση των πολυεπίπεδων Τεχνητών Νευρω-



Διάγραμμα 5.23: Αναπαράσταση ευθείας που σχηματίζει ο ένας νευρώνας προκειμένου να διαχωρίζει δύο κλάσεις.

νικών Δικτύων και αποτελεί την πιο γενική μορφή του κανόνα Δέλτα που εφαρμόζουμε στην περίπτωση του μονοεπίπεδου perceptron.

Αλγόριθμος οπισθοδιάδοσης λάθους

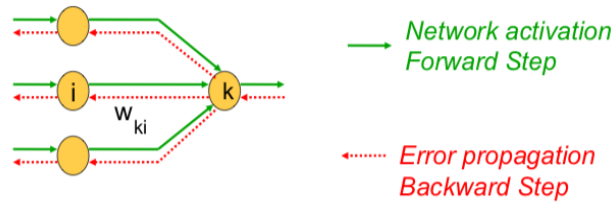
Ένας από τους δημοφιλείς αλγορίθμους εκμάθησης πολυεπίπεδων Νευρωνικών Δικτύων είναι ο αλγόριθμος οπισθοδιάδοσης του λάθους (error backpropagation algorithm). Ο αλγόριθμος αυτός εφαρμόζοντας την μέθοδο βελτιστοποίησης της καθόδου με βάση την κλίση (gradient descent) και τον κανόνα αλυσίδας καταφέρνει να προσαρμόσει με τέτοιο τρόπο τα βάρη ώστε να ελαχιστοποιείται το λάθος ταξινόμησης [17][6].

Η εκπαίδευση με βάση τον αλγόριθμο οπισθοδιάδοσης του λάθους χωρίζεται σε δύο μέρη. Αρχικά, το σήμα εισόδου κατευθύνεται, από την είσοδο προς την έξοδο όπου παράγεται η έξοδος του δικτύου (feed forward) ενώ τα βάρη μένουν σταθερά. Η τιμή αυτή συγκρίνεται με την επιθυμητή έξοδο και υπολογίζεται η απόκλιση τους, δηλαδή υπολογίζεται το σφάλμα. Τα σήματα της πρώτης φάσης που διαδίδονται επίπεδο-επίπεδο σε όλο το δίκτυο καταλήγοντας στην έξοδο καλούνται λειτουργικά (function signals) [8][6]. Στην δεύτερη φάση, το σήμα του σφάλματος (error signals) κατευθύνεται αντίστροφα, δηλαδή από την έξοδο προς την είσοδο (backpropagation) και τροποποιούνται κατάλληλα τα βάρη. Τα δύο αυτά είδη σημάτων φαίνονται στο Διάγραμμα 5.24.

Έστω $S = \{\mathbf{x}(n), \mathbf{y}(n)\}_{n=1}^N$ το σύνολο εκπαίδευσης. Το $o_j(n)$ συμβολίζει το λειτουργικό σήμα- έξοδος του νευρώνα εξόδου j και, τουλάχιστον αρχικά, είναι διάφορο της επιθυμητής εξόδου $y_j(n)$ για το διάνυσμα εισόδου $\mathbf{x}(n)$. Έτσι, στην έξοδο του νευρώνα j παράγεται ένα σφάλμα που ορίζεται ως:

$$e_j(n) = y_j(n) - o_j(n), \quad (5.12)$$

όπου $y_j(n)$ είναι το j -οστό στοιχείο του διανύσματος $\mathbf{y}(n)$. Η επιβλεπόμενη μάθηση του δικτύου εκτελείται με την online μέθοδο. Με την μέθοδο αυτή, τα συναπτικά βάρη



Διάγραμμα 5.24: Ροές σήματος στο πολυεπίπεδο perceptron.

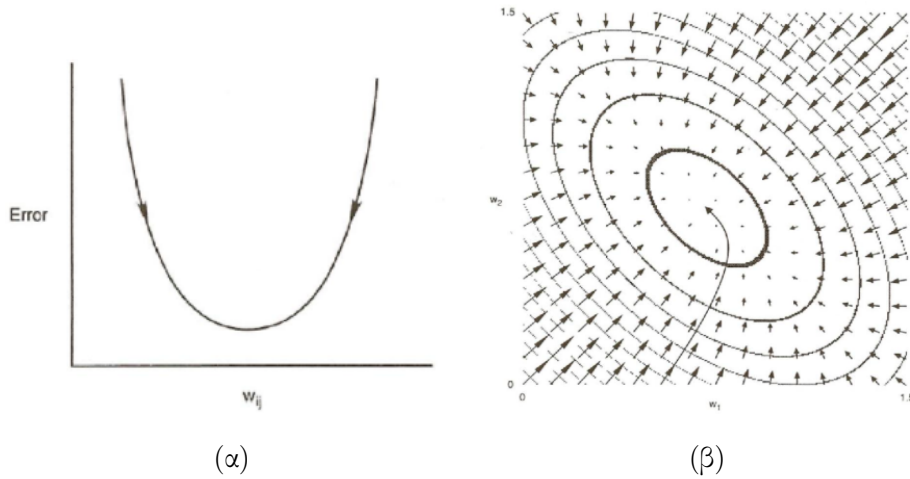
του πολυεπίπεδου perceptron προσαρμόζονται παράδειγμα προς παράδειγμα, δηλαδή σε κάθε εμφάνιση κάποιου προτύπου του συνόλου εκπαίδευσης. Συνεπώς, σκοπός μας είναι να ελαχιστοποιήσουμε το σφάλμα του δικτύου για κάθε παράδειγμα που παρουσιάζεται, δηλαδή σε κάθε βήμα της επανάληψης. Θεωρώντας ως συνάρτηση απώλειας την τετραγωνική έχουμε:

$$E(n) = L(\mathbf{o}, \mathbf{y}) = \frac{1}{2} \sum_{j=1}^C (y_j - o_j)^2 = \frac{1}{2} \sum_{j=1}^C e_j^2(n), \quad (5.13)$$

όπου $E(n)$ αποτελεί το σφάλμα του δικτύου για το παράδειγμα $\mathbf{x}(n)$ και C είναι το σύνολο των νευρώνων εξόδου. Η σχέση (5.13) εκφράζει την συνάρτηση κόστους την οποία θέλουμε να ελαχιστοποιήσουμε. Έτσι, για κάθε παράδειγμα που υποβάλλεται στο δίκτυο διορθώνονται τα βάρη με την μέθοδο gradient descent [49][20]. Η μέθοδος αυτή είναι ένας επαναληπτικός αλγόριθμος βελτιστοποίησης πρώτης τάξης που βρίσκει το ελάχιστο μιας συνάρτησης. Πιο συγκεκριμένα, η παράγωγος της συνάρτησης κόστους ως προς τα βάρη μας υποδεικνύει αν θα μειώσουμε ή αν θα αυξήσουμε τα βάρη. Θετική/ αρνητική τιμή της παραγώγου οδηγεί σε μείωση/ αύξηση των βάρων. Όσο πιο απότομη είναι η κλίση της καμπύλης του Διαγράμματος 5.25α', τόσο περισσότερο απέχει η προβλεπόμενη από την πραγματική έξοδο. Στο Διάγραμμα 5.25β' φαίνονται οι ισοσταθμικές καμπύλες της επιφάνειας του σφάλματος στην περίπτωση που έχουμε δύο βάρη. Παρατηρούμε ότι οι καμπύλες είναι πιο πυκνές στις άκρες και καθώς πλησιάζουμε στο ελάχιστο σφάλμα γίνονται πιο αραιές. Τα βέλη δείχνουν την κατεύθυνση στον διανυσματικό χώρο που σχηματίζουν τα βάρη του δικτύου ώστε να μειώνεται το σφάλμα. Στις άκρες το μέγεθος των βελών είναι μεγαλύτερο καθώς η συνάρτηση σφάλματος ελαττώνεται/ αυξάνεται με γρήγορο ρυθμό. Γίνεται λοιπόν αντιληπτό πως για να μπορέσουμε να υπολογίσουμε τις παραγώγους θα πρέπει οι συναρτήσεις που θα χρησιμοποιήσουμε να είναι διαφορίσιμες. Παράδειγμα μιας συνεχώς διαφορίσιμης μη γραμμικής συνάρτησης που χρησιμοποιούμε στα πολυεπίπεδα νευρωνικά δίκτυα είναι η σιγμοειδής συνάρτηση που έχουμε αναφέρει σε προηγούμενη Παράγραφο. Κατά το ευθύ πέρασμα του δικτύου υπολογίζουμε τα σφάλματα e_j για κάθε νευρώνα εξόδου j όπου η πραγματική έξοδος αυτού, o_j , είναι:

$$o_j = \phi(\text{net}_j) = \phi \left(\sum_{k=0}^p w_{jk} o_k \right).$$

Η ϕ είναι η μη γραμμική παραγωγίσιμη συνάρτηση ενεργοποίησης. Το w_{jk} αναφέρεται στο βάρος μεταξύ του νευρώνα j που εξετάζουμε και των νευρώνων k του προηγούμενου επιπέδου. Το όρισμα net_j της ϕ είναι το άθροισμα των εξόδων o_k των νευρώνων $k = 1, 2, \dots, p$ του προηγούμενου στρώματος πολλαπλασιάσμένα με τα βάρη w_{jk} που



Διάγραμμα 5.25: (α') Καμπύλη που δείχνει την σχέση σφάλματος-βάρους σε ένα δίκτυο. Το κατώτατο σημείο της καμπύλης αντιστοιχεί στο βάρος εκείνο που ελαχιστοποιεί το σφάλμα. Αριστερα/ Δεξιά αυτού η κλίση είναι αρνητική/ θετική. (β') Ισοσταθμικές καμπύλες για την συνάρτηση σφάλματος σε περίπτωση που έχουμε δύο βάρη w_1, w_2 [20].

αποτελούν εισόδους στο νευρώνα j . Αν ο νευρώνας j ανήκει στο πρώτο κρυφό επίπεδο τότε τα o_k αντιστοιχούν στις εισόδους του δικτύου. Στην περίπτωση που ως συνάρτηση ενεργοποίησης χρησιμοποιούμε την σιγμοειδή, τότε αυτή παίρνει την μορφή $\phi(z) = \frac{1}{1+e^{-z}}$. Έστω ότι $u = 1 + e^{-z}$ τότε $\phi(z) = \frac{1}{u}$ [44][17]. Υπολογίζουμε την παράγωγο ως εξής:

$$\begin{aligned}
 \frac{d\phi(z)}{dz} &= \frac{d\phi(z)}{du} \frac{du}{dz} & (5.14) \\
 &= -\frac{1}{u^2}(-e^{-z}) \\
 &= \left(-\frac{1}{(1+e^{-z})^2} \right) (-e^{-z}) \\
 &= \frac{1}{1+e^{-z}} \frac{1+e^{-z}-1}{1+e^{-z}} \\
 &= \frac{1}{1+e^{-z}} \left(\frac{1+e^{-z}}{1+e^{-z}} - \frac{1}{1+e^{-z}} \right) \\
 &= \frac{1}{1+e^{-z}} \left(1 - \frac{1}{1+e^{-z}} \right) \\
 &= \phi(z)(1 - \phi(z)).
 \end{aligned}$$

Στην συνέχεια, υπολογίζουμε τις μερικές παραγώγους της συνάρτησης σφάλματος ως προς τα βάρη χρησιμοποιώντας τον κανόνα αλυσίδας:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial w_{ji}} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} \frac{\partial net_j}{\partial w_{ji}}, \quad (5.15)$$

όπου:

$$\frac{\partial net_j}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \left(\sum_{k=1}^p w_{jk} o_k \right) = \frac{\partial}{\partial w_{ji}} w_{ji} o_i = o_i. \quad (5.16)$$

Αν ο νευρώνας j αντιστοιχεί σε κόμβου πρώτου κρυμμένου επιπέδου τότε $o_i = x_i$. Η μερική παράγωγος της εξόδου του νευρώνα j ως προς άθροισμα των γινομένων εισόδων-βαρών του νευρώνα j (net_j) είναι στην ουσία η μερική παράγωγος της συνάρτησης ενεργοποίησης ως προς το net_j . Για την περίπτωση της λογιστικής συνάρτησης είναι:

$$\frac{\partial o_j}{\partial net_j} = \frac{\partial}{\partial net_j} \phi(net_j) = \phi(net_j)(1 - \phi(net_j)). \quad (5.17)$$

Η τελευταία σχέση μας δείχνει τον λόγο που θα πρέπει η συνάρτηση ενεργοποίησης πρέπει να είναι διαφορίσιμη. Αν ο νευρώνας j είναι νευρώνας εξόδου τότε χρησιμοποιώντας την σχέση (5.13) έχουμε:

$$\frac{\partial E}{\partial o_j} = \frac{\partial L(\mathbf{o}, \mathbf{y})}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} \sum_{j=1}^C (y_j - o_j)^2 = \sum_{j=1}^C (o_j - y_j). \quad (5.18)$$

Ωστόσο, ο υπολογισμός του όρου $\frac{\partial E}{\partial o_j}$ δεν είναι τόσο άμεσος αν ο νευρώνας j είναι εσωτερικός νευρώνας αφού δεν υπάρχει καθορισμένη επιθυμητή έξοδος y_j για αυτόν. Το σήμα του σφάλματος σε ένα τέτοιο νευρώνα θα υπολογιστεί χρησιμοποιώντας τα σήματα σφάλματος όλων των νευρώνων με τους οποίους συνδέεται άμεσα ο εν λόγω νευρώνας. Αφού λοιπόν το o_j γράφεται ως συνδιασμός των εισόδων των νευρώνων του επόμενου στρώματος μπορούμε να θεωρήσουμε το E ως μια συνάρτηση που δέχεται ως όρισμα όλους τους νευρώνες $L = (u, v, ..w)$ που λαμβάνουν είσοδο από τον νευρώνα j . Έχουμε δηλαδή:

$$\frac{\partial E(o_j)}{\partial o_j} = \frac{\partial E(net_u, net_v, .., net_w)}{\partial o_j}. \quad (5.19)$$

Σύμφωνα με τον κανόνα αλυσίδας η σχέση (5.19) γίνεται:

$$\frac{\partial E}{\partial o_j} = \sum_{l \in L} \frac{\partial E}{\partial net_l} \frac{\partial net_l}{\partial o_j} = \sum_{l \in L} \frac{\partial E}{\partial o_l} \frac{\partial o_l}{\partial net_l} \frac{\partial net_l}{\partial o_j} = \sum_{l \in L} \frac{\partial E}{\partial o_l} \frac{\partial o_l}{\partial net_l} w_{lj}, \quad (5.20)$$

αφού αν ο νευρώνας j έχει p εισόδους τότε $net_j = \sum_{k=1}^p w_{jk} x_{jk}$ τότε η έξοδος του είναι $o_j = \phi(net_j)$ και η είσοδος του ακριβώς επόμενου από τον j νευρώνα είναι $net_l = w_{lj} \phi(net_j)$. Συνεπώς, $\frac{\partial net_l}{\partial o_j} = w_{lj}$. Λαμβάνοντας υπόψη τις σχέσεις (5.16), (5.18), (5.19) η σχέση (5.15) γίνεται:

$$\begin{aligned} \frac{\partial E}{\partial w_{ji}} &= \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial w_{ji}} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} \frac{\partial net_j}{\partial w_{ji}} = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} o_i \\ &\Rightarrow \frac{\partial E}{\partial w_{ji}} = o_i \delta_j, \end{aligned}$$

όπου

$$\delta_j = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} = \begin{cases} \frac{\partial L(o_j, \mathbf{y})}{\partial o_j} \frac{d\phi(net_j)}{dnet_j} & j \text{ αντιστοιχεί σε νευρώνα εξόδου} \\ \sum_{l \in L} w_{jl} \delta_l \frac{d\phi(net_j)}{dnet_j} & j \text{ αντιστοιχεί σε εσωτερικό νευρώνα.} \end{cases}$$

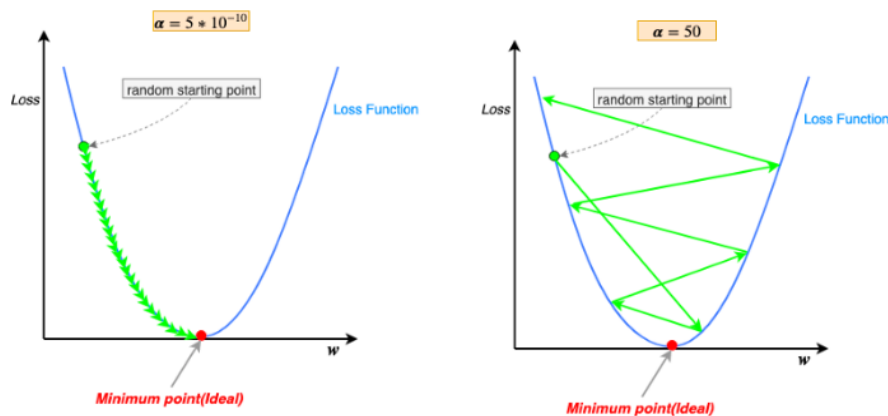
Αν η ϕ είναι η λογιστική συνάρτηση ενεργοποίησης και η συνάρτηση απώλειας είναι η τετραγωνική τότε:

$$\delta_j = \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} = \begin{cases} (o_j - y_j)o_j(1 - o_j) & j \text{ αντιστοιχεί σε νευρώνα εξόδου} \\ (\sum_{l \in L} w_{lj}\delta_l) o_j(1 - o_j) & j \text{ αντιστοιχεί σε εσωτερικό νευρώνα.} \end{cases}$$

Προκειμένου να τροποποιήσουμε τα συναπτικά βάρη w_{ji} με την μέθοδο gradient descent είναι αναγκαίο να επιλέξουμε κάποια τιμή για την παράμετρο μάθησης η , ($\eta > 0$). Οι αλλαγές στα βάρη αντανακλούν την επίδραση που έχουν αυτές στο σφάλμα E [8][22]. Αν η μερική παράγωγος $\frac{\partial E}{\partial w_{ji}}$ είναι θετική και αυξήσουμε το βάρος w_{ji} τότε το σφάλμα E αυξάνει. Αντίθετα, αν είναι αρνητική και αυξήσουμε το βάρος τότε μειώνεται το σφάλμα. Το καινούργιο Δw_{ji} προστίθεται στο παλιό w_{ji} και το γινόμενο της παραμέτρου μάθησης με την μερική παράγωγο πολλαπλασιασμένα με το -1 εγγυάται ότι w_{ji} αλλάζει με τρόπο τέτοιο ώστε να μειώνεται το σφάλμα E . Με άλλα λόγια, το $-\eta \frac{\partial E}{\partial w_{ji}}$ τροποποιεί το w_{ji} ώστε το σφάλμα να μειώνεται:

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} = -\eta o_i \delta_j. \quad (5.21)$$

Αναφορικά με την παράμετρο μάθησης η , όσο μικρότερη είναι η τιμή της, τόσο πιο μικρές είναι οι αλλαγές των συναπτικών βαρών από την μια επανάληψη στην άλλη. Στην περίπτωση αυτή, απαιτούνται πολλοί κύκλοι καθώς η διαδικασία μάθησης είναι αργή. Αντίθετα, μεγάλες τιμές του η επιταχύνουν την μάθηση αλλά υπάρχει κίνδυνος οι απότομες αλλαγές των βαρών να πάρουν την μορφή ταλάντωσης όποτε το σύστημα είναι ασταθές (Διάγραμμα 5.26). Στην συνέχεια, παρουσιάζονται συνοπτικά τα βήματα του



Διάγραμμα 5.26: Οι επιδράσεις στο δίκτυο όταν επιλέγουμε πολύ μικρή τιμή της παραμέτρου η ($\eta = 5 * 10^{-10}$) και πολύ μεγάλες τιμές όπως $\eta = 50$.

αλγόριθμου backpropagation: Αφού αρχικοποιηθούν τα συναπτικά βάρη των νευρώνων με μικρές πραγματικές τιμές, τότε για κάθε παράδειγμα εκπαίδευσης:

1. Υπόβαλε το παράδειγμα $\mathbf{x}(n)$ στο δίκτυο και υπολόγισε την έξοδο $\mathbf{y}(n)$ του δικτύου.
2. Για κάθε νευρώνα εξόδου j υπολόγισε:

$$\delta_j = (o_j - y_j)o_j(1 - o_j).$$

3. Για κάθε κρυμμένο νευρώνα j υπολόγισε:

$$\delta_j = \left(\sum_{l \in L} w_{lj} \delta_l \right) o_j (1 - o_j).$$

4. Τροποποιήστε το βάρος:

$$w_{ji} = w_{ji} + \Delta w_{ji},$$

$$\text{όπου } \Delta w_{ji} = -\eta o_i \delta_j.$$

Μέθοδος κανόνα Δέλτα

Η μέθοδος κανόνα Δέλτα (delta rule) χρησιμοποιείται για την εκπαίδευση ενός μονοεπίπεδου perceptron καθώς όπως θα δούμε οι μετατροπές των βαρών γίνονται μόνο για το επίπεδο εκείνο που προϋπάρχει της εξόδου [45][20]. Αποτελεί μια πιο ειδική μορφή της γενικής μεθόδου backpropagation την οποία αναλύσαμε προηγουμένως. Όταν χρησιμοποιούμε τον κανόνα Δέλτα για την ενημέρωση των συναπτικών βαρών συνήθως θεωρούμε ότι η συνάρτηση ενεργοποίησης είναι η γραμμική ($\phi(x) = x$) και ο κανόνας παίρνει την μορφή:

$$\Delta w_{ji} = -\eta(o_j - y_j)\phi'(net_j)x_i = -\eta(o_j - y_j)x_i. \quad (5.22)$$

Παρατηρούμε ότι η σχέση (5.22) είναι η σχέση (5.21) με την διαφορά ότι το o_i αντιστοιχεί στις εισόδους x_i .

Απόδειξη του κανόνα Δέλτα

Η απόδειξη του κανόνα Δέλτα γίνεται μέσα από την προσπάθεια ελαχιστοποίησης του σφάλματος της εξόδου του δικτύου χρησιμοποιώντας την μέθοδο της απότομης καθόδου. Όπως είδαμε, το σφάλμα του δικτύου για μια είσοδο $\mathbf{x}(n)$ δίνεται από [36][8]:

$$E(n) = \frac{1}{2} \sum_{j=1}^C (y_j - o_j)^2. \quad (5.23)$$

Όπως και στην περίπτωση της οπισθοδιάδοσης του λάθους υπολογίζουμε:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \left(\frac{1}{2} \sum_{j=1}^C (y_j - o_j)^2 \right). \quad (5.24)$$

Χρησιμοποιώντας δύο φορές τον κανόνα της αλυσίδας η τελευταία σχέση γίνεται:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\left(\frac{\partial}{\partial} \frac{1}{2} \sum_{j=1}^C (y_j - o_j)^2 \right)}{\partial o_j} \frac{\partial o_j}{\partial w_{ji}} = -(y_j - o_j) \frac{\partial o_j}{\partial w_{ji}} = -(y_j - o_j) \frac{\partial o_j}{\partial net_j} \frac{\partial net_j}{\partial w_{ji}}, \quad (5.25)$$

όπου $\phi(net_j) = o_j$, δηλαδή $\frac{\partial o_j}{\partial net_j} = \phi'(net_j)$. Επίσης, το net_j γράφεται ως $\sum_k x_k w_{jk}$ όπου k το σύνολο των εισόδων στο δίκτυο.

$$\frac{\partial E}{\partial w_{ji}} = -(y_j - o_j) \phi'(net_j) \frac{\partial \sum_k x_k w_{jk}}{\partial w_{ji}} = -(y_j - o_j) \phi'(net_j) x_i. \quad (5.26)$$

Σύμφωνα με την μέθοδο gradient descent για να μειώσουμε το σφάλμα αλλάζουμε το βάρος προς κατεύθυνση αντίθετη αυτής της μερικής παραγώγου. Διαλέγοντας μια παράμετρο μάθησης η εφαρμόζουμε τον κανόνα Δέλτα. Σύμφωνα με αυτόν εκπαιδεύουμε το μονοεπίπεδο perceptron:

$$\Delta w_{ji} = \eta(y_j - o_j)\phi'(net_j)x_i = \eta(y_j - o_j)x_i.$$

όπου $\phi(net_j) = net_j$.

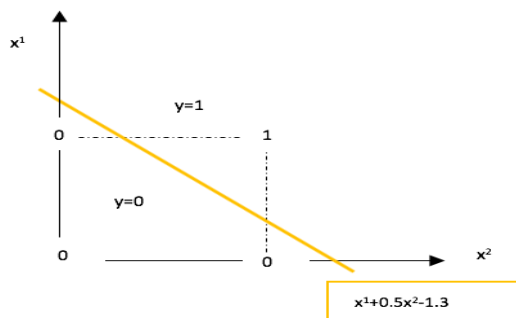
Πρόβλημα λογικού AND

Έστω ότι θέλουμε να εκπαιδεύσουμε ένα perceptron ώστε να επιλύει το πρόβλημα του λογικού ΚΑΙ. Σκοπός δηλαδή είναι να βρούμε τις τιμές των βαρών. Κατάλληλο perceptron για το πρόβλημα αυτό είναι εκείνο με δύο εισόδους και ένα νευρώνα εξόδου. Αρχικά, κωδικοποιούμε τις εισόδους ως διανύσματα με την επιθυμητή τους έξοδο. Έχουμε $\mathbf{x}_1 = [0, 0]$, $y_1 = 0$, $\mathbf{x}_2 = [0, 1]$, $y_2 = 0$, $\mathbf{x}_3 = [1, 0]$, $y_3 = 0$ και $\mathbf{x}_4 = [1, 1]$, $y_4 = 1$. Η τιμή του κατωφλιού w_0 είναι αρχικά -1 . Θέτουμε επίσης κάποιες αρχικές τυχαίες τιμές για τα βάρη, έστω $\mathbf{w} = [0.8, 1.0, 0.5]$.

Όπως είδαμε για την εκπαίδευση του απλού perceptron χρησιμοποιούμε τον κανόνα Δέλτα με συνάρτηση ενεργοποίησης την ταυτοτική. Για τον πρώτο κύκλο εκπαίδευσης υπολογίζουμε για κάθε παράδειγμα $\sum_{i=0}^2 w_{3i}x_{3i}$. Το w_{3i} αντιστοιχεί στο βάρος που συνδέει τον νευρώνα εξόδου 3 και τον νευρώνα εισόδου i , $i = 1, 2$. Για το πρώτο παράδειγμα έχουμε $0.8 \times (-1) + 1 \times 0 + 0.5 \times 0 = 0.8$. Αφού $-0.8 < 0$ η έξοδος είναι $o_1 = 0$. Συνεπώς, αφού το σφάλμα e είναι $y_1 - o_1 = 0$ τα βάρη δεν αλλάζουν. Όμοίως για το δεύτερο πρότυπο, το σφάλμα είναι 0 και τα βάρη δεν τροποποιούνται. Για το τρίτο παράδειγμα, βρίσκουμε ότι $\sum_{i=0}^2 w_{3i}x_{3i} = 0.2 \geq 0$, δηλαδή η έξοδος o_3 είναι 1 με αποτέλεσμα το σφάλμα να είναι $y_3 - o_3 = -1$. Άρα τα βάρη αλλάζουν και σύμφωνα με το κανόνα Δέλτα είναι $\Delta w_{3i} = 0.5(-1)x_i = [1.3, 0.5, 0.5]^T$. Συνεχίζουμε εφαρμόζοντας την ίδια διαδικασία για τα υπόλοιπα παραδείγματα. Στο συγκεκριμένο πρόβλημα χρειάζονται 3 κύκλοι εκπαίδευσης για να απαντήσει ο νευρώνας σωστά σε όλα δεδομένα εκπαίδευσης με τελικά βάρη $\mathbf{w} = [1.3, 1.0, 0.5]$. Έτσι το επίπεδο απόφασης έχει την μορφή: $x^1 + 0.5x^2 - 1.3 = 0$ (Διάγραμμα 5.27). Πρακτικά, η εύρεση σφάλματος στην προβλεπόμενη έξοδο έχει ως αποτέλεσμα την μετατόπιση του υπερεπιπέδου ώστε να μειωθεί αυτό το σφάλμα. Γεωμετρικά, μεγάλη τιμή της παραμέτρου η σημαίνει απότομη μετατόπιση του υπερεπιπέδου ενώ μικρή τιμή έχει ως αποτέλεσμα μικρές αλλαγές της θέσης του.

Πρόβλημα XOR

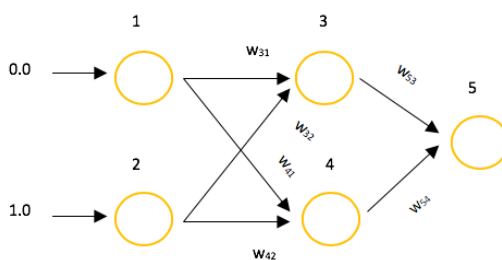
Δεν χρειάζεται κανείς να πάει σε περίπλοκες καταστάσεις για να αναζητήσει μη γραμμικά διαχωρίσιμα προβλήματα. Η γνωστή Boolean συνάρτηση Αποκλειστικού Ή (XOR) αποτελεί τυπικό παράδειγμα τέτοιου προβλήματος. Οι Boolean συναρτήσεις μπορούν να ερμηνευτούν ως προβλήματα ταξινόμησης. Η επίλυση του XOR είναι πολυπλοκότερο πρόβλημα από ότι το AND. Συνεπώς, είναι ανάγκη το δίκτυο να έχει διαφορετική δομή από αυτή του απλού perceptron. Η συνάρτηση XOR είναι μια συνάρτηση που



Διάγραμμα 5.27: Ευθεία διαχωρισμού που σχηματίζει το απλό perceptron για το πρόβλημα του λογικού AND.

δέχεται δύο εισόδους, δίνει έξοδο με τιμές 0 ή 1 και ισχύει ότι αν και οι δύο εισοδοι είναι ίδιες τότε η έξοδος είναι 0 ενώ αν οι είναι διαφορετικές είναι 1.

Για την λύση του προβλήματος XOR χρησιμοποιούμε ένα δίκτυο με δύο εισόδους, ένα κρυφό στρώμα με δύο νευρώνες και έναν νευρώνα εξόδου. Η εκπαίδευσή του βασίζεται στην μέθοδο οπισθοδιάδοσης του λάθους με παράμετρο εκμάθησης $\eta = 1$. Η συνάρτηση ενεργοποίησης που θα χρησιμοποιήσουμε σε όλους τους νευρώνες είναι η σιγμοειδής. Όπως και πριν, κωδικοποιούμε τις εισόδους ως διανύσματα με την επιθυμητή τους έξοδο. Έχουμε $\mathbf{x}_1 = [0, 0]$, $y_1 = 0$, $\mathbf{x}_2 = [0, 1]$, $y_2 = 1$, $\mathbf{x}_3 = [1, 1]$, $y_3 = 0$ και $\mathbf{x}_4 = [1, 0]$, $y_4 = 1$. Στο παράδειγμα αυτό δεν θα κάνουμε χρήση κατωφλιού. Θέτουμε επίσης κάποιες αρχικές τυχαίες τιμές για τα βάρη, έστω $w_{31} = 0.1$, $w_{41} = 0.2$, $w_{32} = 0.2$, $w_{53} = -0.1$ και $w_{54} = -0.1$. Θα βρούμε ενδεικτικά τις τιμές των βαρών για τον αμέσως επόμενο κύκλο εκπαίδευσης.



Διάγραμμα 5.28: Δίκτυο για την επίλυση του προβλήματος XOR.

Έστω ότι αρχίζουμε από το παράδειγμα (0.0, 1.0) όπου επιθυμητή έξοδος είναι 1. Αρχικά, υπολογίζουμε την πραγματική έξοδο του δικτύου. Οι δύο πρώτοι νευρώνες βρίσκονται στο επίπεδο εισόδου και για το λόγο αυτό οι εισοδοι απλά μεταφέρονται στην έξοδο, δηλαδή $o_1 = x^1 = 0$ και $o_2 = x^2 = 1$. Στην συνέχεια, για τον κόμβο τρία που είναι κρυφός υπολογίζουμε τον αθροιστή ως $w_{31}x_{31} + w_{32}x_{32} = 0.1 \times 0 + 0.2 \times 1 = 0.2$ και η έξοδος είναι $o_3 = \phi(0.2) = \frac{1}{1+e^{-0.2}} = 0.550$. Ομοίως για τον νευρώνα τέσσερα, η είσοδος του είναι $w_{41}x_{41} + w_{42}x_{42} = 0.2 \times 0 + 0.1 \times 1 = 0.1$ και η έξοδος $\phi(0.1) = 0.525$. Για τον κόμβο πέντε ο αθροιστής υπολογίζεται ως $w_{53}o_3 + w_{54}o_4 = -0.108$ και η τελική έξοδος του ΤΝΔ είναι $o_5 = \phi(-0.108) = 0.473$. Άρα, το σφάλμα του μοναδικού νευρώνα εξόδου είναι επιθυμητή-πραγματική=1-0.473=0.527.

Σειρά έχει το προς το ανάστροφο πέρασμα. Αρχίζουμε από τον κόμβο εξόδου υπολογίζοντας την τοπική κλίση $\delta_5 = (o_5 - y_5)o_5(1 - o_5) = 0.527 \times 0.473(1 - 0.473) = 0.131$. Για τον κρυφό νευρώνα τέσσερα έχουμε $\delta_4 = o_4(1 - o_4)(\delta_5 w_{54}) = 0.525(1 - 0.525)(0.131 \times (-0.1)) = -0.003$. Παρατηρούμε ότι ο νευρώνας αυτός έχει μικρή συμβολή στο λάθος. Ομοίως για τον κόμβο τρία $\delta_3 = o_3(1 - o_3)(\delta_5 w_{54}) = -0.003$. Στην συνέχεια, γίνονται οι διορθώσεις στα βάρη:

$$\begin{aligned}\Delta w_{53} &= \eta \delta_5 o_3 = 0.072, \\ \Delta w_{54} &= \eta \delta_5 o_4 = 0.069, \\ \Delta w_{41} &= \eta \delta_4 o_1 = 0.0, \\ \Delta w_{42} &= \eta \delta_4 o_2 = -0.003, \\ \Delta w_{31} &= \eta \delta_3 o_1 = 0.0, \\ \Delta w_{32} &= \eta \delta_3 o_2 = -0.003.\end{aligned}$$

Έτσι, τα νέα βάρη έχουν ως εξής:

$$\begin{aligned}w_{53} &= w_{53} + \Delta w_{53} = -0.028, \\ w_{54} &= w_{54} + \Delta w_{54} = -0.031, \\ w_{41} &= w_{41} + \Delta w_{41} = 0.2, \\ w_{42} &= w_{42} + \Delta w_{42} = 0.097, \\ w_{31} &= w_{31} + \Delta w_{31} = 0.1, \\ w_{32} &= w_{32} + \Delta w_{32} = 0.197.\end{aligned}$$

Δείξαμε λοιπόν ενδεικτικά τους υπολογισμούς που χρειάζονται για ένα προτύπο. Αφού διορθώσουμε τα βάρη σύμφωνα με την παραπάνω διαδικασία και ξαναεκτελέσουμε το προς τα εμπρός πέρασμα παρατηρούμε πως η έξοδος είναι 0.492, δηλαδή πιο κοντά στην επιθυμητή (1.0) από ότι ήταν η αρχική που είχαμε υπολογίσει (0.473). Βέβαια, απαιτούνται αρκετοί ακόμα κύκλοι για την εύρεση της σωστής τιμής.

5.4 Λογιστική Παλινδρόμηση

5.4.1 Εισαγωγή

Η Λογιστική Παλινδρόμηση (logistic/ logit regression) παρουσιάστηκε για πρώτη φορά από τον D. Cox την δεκαετία του '50 και είναι μια μορφή παλινδρόμησης, ένα μοντέλο πρόβλεψης που εφαρμόζεται όταν η μεταβλητή απόκρισης είναι κατηγορική. Είναι χρήσιμη σε καταστάσεις που θέλουμε να προβλέψουμε την ύπαρξη ή την απουσία ενός χαρακτηριστικού.

Η λογιστική παλινδρόμηση μπορεί να έχει τρεις μορφές, την δυαδική (binary), την πολυωνυμική (multinomial) και την διατάξιμη (ordinal). Στην πολυωνυμική Λογιστική Παλινδρόμηση η εξαρτημένη μεταβλητή μπορεί να πάρει περισσότερες από δύο

τιμές ενώ στην τακτική η μεταβλητή απόκρισης είναι διατεταγμένη. Υπάρχει δηλαδή μια φυσική διαβάθμιση όπως για παράδειγμα σε μια ερώτηση όπου η απάντηση μπορεί να είναι καθόλου/λίγο/πολύ. Η πιο απλή εκδοχή είναι δίτιμη Λογιστική Παλινδρόμηση όπου η εξαρτημένη μεταβλητή μπορεί να έχει δύο τιμές (0,1), όπως για παράδειγμα επιτυχία/ αποτυχία, Ναι/Όχι.

Ο σκοπός της Λογιστικής Παλινδρόμησης είναι παρόμοιος με αυτόν της κλασικής γραμμικής παλινδρόμησης, με την διαφορά ότι η μεταβλητή απόκρισης είναι κατηγορική και όχι ποσοτική. Επιχειρεί λοιπόν να προβλέψει αυτή την κατηγορική μεταβλητή, να υπολογίσει δηλαδή την πιθανότητα η μεταβλητή απόκρισης να ανήκει σε κάποια κατηγορία. Η τιμή της πιθανότητας παίρνει τιμές στο διάστημα $[0,1]$, κάτι που δεν μπορεί να κάνει η γραμμική παλινδρόμηση καθώς αυτή υπολογίζει και τιμές έξω από αυτά τα όρια. Στην πιο απλή εκδοχή ένα λογιστικό μοντέλο χρησιμοποιεί την λογιστική συνάρτηση για να μοντελοποιήσει μια δίτιμη εξαρτημένη μεταβλητή. Για να το πετύχει αυτό χρησιμοποιεί τον λογάριθμο του λόγου των συμπληρωματικών πιθανοτήτων (log-odds/ logarithm of the odds). Ο λογάριθμος αυτός για την κατηγορία '1' είναι ο γραμμικός συνδιασμός των ανεξάρτητων μεταβλητών. Στην συνέχεια, χρησιμοποιούμε την λογιστική συνάρτηση για να μετασχηματίσουμε τον λογάριθμο σε πιθανότητα.

Ως προβλεπτική μέθοδος, η Λογιστική Παλινδρόμηση χρησιμοποιείται σε πολλές εφαρμογές της Μηχανικής Μάθησης. Κάποιες από αυτές είναι η μέτρηση της θνησιμότητας τραυματισμένων ασθενών (Trauma and injury security score, TRISS), η εκτίμηση της σοβαρότητας ασθενειών, η πρόβλεψη του ρίσκου εμφάνισης μια νόσου (διαβήτης, στεφανιαία νόσος κ.α.). Για να λυθούν τα παραπάνω προβλήματα ταξινόμησης αξιοποιούνται χαρακτηριστικά των ασθενών όπως ηλικία, φύλο, ένδειξη μάζας σώματος κ.τ.λ. Επίσης, πολλές φορές εφαρμόζουμε την μέθοδο της Λογιστικής Παλινδρόμησης για την πρόβλεψη της επιλογής πολιτικού κόμματος με βάση την καταγραφή των δημογραφικών στοιχείων των πολιτών όπως τόπος διαμονής, εισόδημα, ηλικία, προηγούμενη ψηφοφορία κτλ. Τέλος, η μέθοδος αυτή χρησιμοποιείται και σε τραπεζικές συναλλαγές, όπως για παράδειγμα στην εκτίμηση της πιθανότητας αθέτησης από δανειολήπτη της αποπληρωμής του δανείου του.

5.4.2 Λογιστικό μοντέλο

Στην Παράγραφο αυτή θα δούμε αναλυτικότερα το μοντέλο της δίτιμης λογιστικής παλινδρόμησης. Έστω ότι έχουμε ένα σύνολο εκπαίδευσης $S = (\mathbf{x}_i, y_i)_{i=1}^N$, όπου \mathbf{x}_i είναι το l -διάστατο διάνυσμα των l ανεξάρτητων μεταβλητών (l χαρακτηριστικά) και y_i είναι η μεταβλητή απόκρισης για την i -οστή παρατήρηση. Όπως έχουμε δει, τα χαρακτηριστικά μπορούν να είναι κατηγορικά, ποσοτικά ή συνδιασμός αυτών, ενώ η εξαρτημένη μεταβλητή y έχει κωδικοποιηθεί με την τις δυαδικές τιμές 0,1 που αντιστοιχούν σε επιτυχία/ αποτυχία, παρουσία/ απουσία [13][37].

Με $p = P(Y = 1 | \mathbf{X} = \mathbf{x})$ συμβολίζουμε την πιθανότητα επιτυχίας δεδομένης μιας παρατήρησης \mathbf{x} . Αντίστοιχη πιθανότητα αποτυχίας είναι $1 - p = P(Y = 0 | \mathbf{X} = \mathbf{x})$. Ο πιο απλός τρόπος να εκτιμήσουμε αυτή την πιθανότητα θα ήταν το μοντέλο της απλής γραμμικής παλινδρόμησης, δηλαδή $p = \beta_0 + \beta \mathbf{x} = \beta_0 + \beta_1 x^1 + \beta_2 x^2 + \dots + \beta_l x^l$. Επειδή το αποτέλεσμα όμως μπορεί να είναι μια πιθανότητα έξω από το

διάστημα $[0,1]$, χρησιμοποιούμε αντί για την πιθανότητα, μια συνδετική συνάρτηση g η οποία σχηματίζεται λαμβάνοντας υπόψη τον λογάριθμο των πιθανοτήτων odds (σχετική πιθανότητα), $g(p) = \log\left(\frac{p}{1-p}\right)$, $g: [0,1] \rightarrow [-\infty, +\infty]$. Δηλαδή έχουμε:

$$g(p) = \log\left(\frac{p}{1-p}\right) = \hat{\beta}_0 + \hat{\beta}_1 x^1 + \dots + \hat{\beta}_l x^l. \quad (5.27)$$

Παρατηρούμε ότι η συνάρτηση g είναι γραμμική ως προς τα χαρακτηριστικά και παίρνει τιμές από το $-\infty$ στο $+\infty$. Η εκτιμώμενη πιθανότητα \hat{p} προκύπτει ως εξής:

$$\begin{aligned} \frac{\hat{p}}{1-\hat{p}} &= e^{\hat{\beta}_0 + \hat{\beta}x} \\ \Rightarrow \hat{p} &= (1-\hat{p})e^{\hat{\beta}_0 + \hat{\beta}x} \\ \Rightarrow \hat{p} &= \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x^1 + \dots + \hat{\beta}_l x^l}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x^1 + \dots + \hat{\beta}_l x^l}}. \end{aligned} \quad (5.28)$$

Αφού εκτιμήσουμε τις άγνωστες παραμέτρους, θα χρησιμοποιήσουμε την εκτιμώμενη πιθανότητα (5.28) για να ταξινομήσουμε κάθε νέο στιγμιότυπο \mathbf{x}_{new} σε κάποια από τις ομάδες 0,1 με τον εξής κανόνα:

$$\hat{y}_{new} = \begin{cases} 0 & \text{αν } \hat{p} < 0.5 \\ 1 & \text{αν } \hat{p} \geq 0.5. \end{cases}$$

Στην γραμμική παλινδρόμηση εκτιμούμε τις παραμέτρους β_i με την μέθοδο των ελαχίστων τετραγώνων. Στην λογιστική του παλινδρόμηση, η εκτίμηση τους γίνεται με τη μέθοδο της μέγιστης πιθανοφάνειας [9][32]. Η μεταβλητή απόκρισης μπορεί να λάβει την τιμή 1 με πιθανότητα επιτυχίας p και την τιμή 0 με πιθανότητα αποτυχίας $1-p$, ακολουθεί δηλαδή την κατανομή Bernoulli:

$$P(Y = y | \mathbf{X} = \mathbf{x}, \boldsymbol{\beta}) = p^y (1-p)^{1-y}.$$

Στο λογιστικό λοιπόν μοντέλο η συνάρτηση πιθανοφάνειας είναι:

$$L = \prod_{i=1}^N p_i^{y_i} (1-p_i)^{1-y_i}, \quad (5.29)$$

όπου p_i είναι η πιθανότητα επιτυχίας της i -οστής παρατήρησης. Λογαριθμώντας, έχουμε:

$$\log(L) = \sum_{i=1}^N y_i p_i + (1-y_i)(1-p_i).$$

Μεγιστοποιώντας την τελευταία έκφραση εκτιμούμε το διάνυσμα $\boldsymbol{\beta}$. Στην συνέχεια, θα μελετήσουμε την φυσική ερμηνεία των συντελεστών $\boldsymbol{\beta}$. Ο λόγος των σχετικών odds σύμφωνα με την σχέση (5.27) για ένα παράδειγμα \mathbf{x}_i είναι:

$$\frac{\hat{p}_i}{1-\hat{p}_i} = e^{\hat{\beta}_0 + \hat{\beta}_1 x_i^1 + \dots + \hat{\beta}_l x_i^l}.$$

Ο λόγος των odds (odds ratio) προκύπτει αν αυξήσουμε την τιμή της j -οστής μεταβλητής κατά μια μονάδα, διατηρώντας όλες τις υπόλοιπες μεταβλητές σταθερές. Παρατηρούμε ότι:

$$\text{odds ratio} = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x_i^1 + \dots + \hat{\beta}_j (x_i^j + 1) + \dots + \hat{\beta}_l x_i^l}}{e^{\hat{\beta}_0 + \hat{\beta}_1 x_i^1 + \dots + \hat{\beta}_j x_i^j + \dots + \hat{\beta}_l x_i^l}} = e^{\hat{\beta}_j}. \quad (5.30)$$

Δηλαδή, αυξάνοντας την τιμή μιας ανεξάρτητης μεταβλητής x_i^j ενός παραδείγματος x_i κατά μια μονάδα, προκαλούμε πολλαπλασιαστική αύξηση της σχετικής πιθανότητας κατά e^{β_i} .

Στην περίπτωση που έχουμε Λογιστική Παλινδρόμηση περισσότερων από δύο κλάσεων μπορούμε να το μετατρέψουμε σε δυαδικά υποπροβλήματα. Ας υποθέσουμε ότι η μεταβλητή απόκρισης είναι το μεταφορικό μέσο (αυτοκίνητο, ποδήλατο, μετρό) [13]. Για να υπολογίσουμε την πιθανότητα το μέσο να είναι το αυτοκίνητο, θεωρούμε ως επιτυχία (1) το αυτοκίνητο και όλα τα υπόλοιπα αποτυχία (0). Η διαδικασία συνεχίζεται μέχρι να υπολογιστούν οι πιθανότητες για όλες τις πιθανές κλάσεις. Τελικά, η πρόβλεψη είναι η κατηγορία αυτή που έχει την μεγαλύτερη πιθανότητα να βρίσκεται σε αυτή το όχημα. Με τον τρόπο αυτό ένα πρόβλημα 3 κλάσεων μετατράπηκε σε 3 προβλήματα, κάθε ένα από αυτά έχει 2 κλάσεις.

Παράδειγμα

Ας θεωρήσουμε το παρακάτω πρόβλημα. Έστω ότι ένα δείγμα 20 ατόμων ξοδεύουν από 0 έως 6 ώρες για να διαβάσουν για ένα διαγώνισμα. Θέλουμε να μελετήσουμε πως ο αριθμός των ωρών που διαβάζουν επιδρά στην απόδοση των μαθητών στο διαγώνισμα. Θα χρησιμοποιήσουμε Λογιστική Παλινδρόμηση καθώς η κατηγορική μεταβλητή αφορά την επιτυχία (1) ή την αποτυχία (0) των μαθητών στο διαγώνισμα. Τα δεδομένα του

Hours	0.50	0.75	1.00	1.25	1.50	1.75	1.75	2.00	2.25	2.50	2.75	3.00	3.25	3.50	4.00	4.25	4.50	4.75	5.00	5.50
Pass	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1

Πίνακας 5.1: Αριθμός ωρών που κάθε μαθητής διαβάσε και η επιδοσή του στο διαγώνισμα (0/1).

προβλήματος φαίνονται στον Πίνακα 5.1 και τα αποτελέσματα της Λογιστικής Παλινδρόμησης στον Πίνακα 5.2, δηλαδή οι τιμές των παραμέτρων β_0, β_1 που αντιστοιχούν στην σταθερά και στον συντελεστή της ανεξάρτητης μεταβλητής Hours. Από τα απο-

	Coefficient	Std. Error	z-value	P-value (Wald)
Intercept	-4.0777	1.7610	-2.316	0.0206
Hours	1.5046	0.6287	2.393	0.0167

Πίνακας 5.2: Αποτελέσματα Λογιστικής Παλινδρόμησης.

τελέσματα φαίνεται ότι οι ώρες συσχετίζονται με την πιθανότητα να περάσει κάποιος

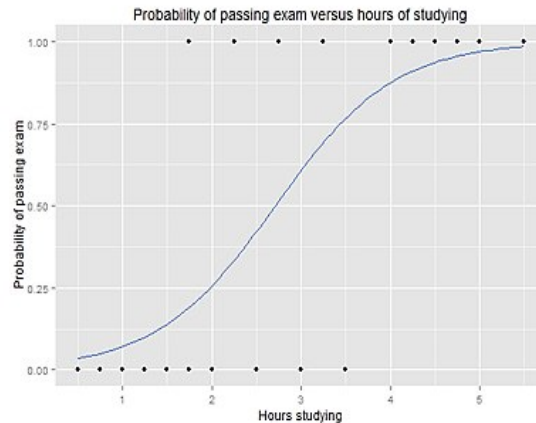
το τεστ ($p_{value} = 0.0167$). Έτσι, έχουμε τα εξής:

$$\text{log-odds επιτυχίας στο διαγώνισμα} = 1.5046\text{Hours} - 4.0777 = 1.5046(\text{Hours} - 2.71),$$

$$\text{odds επιτυχίας στο διαγώνισμα} = e^{1.5046\text{Hours}-4.0777} = e^{1.5046(\text{Hours}-2.71)},$$

$$\text{πιθανότητα επιτυχίας στο διαγώνισμα} = \frac{1}{1 + e^{-(1.5046\text{Hours}-4.0777)}}.$$

Αξίζει να παρατηρήσουμε ότι κάθε μοναδιαία αύξηση ώρας αυξάνει τον λογάριθμο της



Διάγραμμα 5.29: Γραφική αναπαράσταση της καμπύλης της Λογιστικής Παλινδρόμησης που δείχνει την πιθανότητα να περάσει κάποιος μαθητής το διαγώνισμα με βάση τις ώρες που έχει διαβάσει.

σχετικής πιθανότητας να περάσει κάποιος το διαγώνισμα κατά 1.5046. Συνεπώς, η αύξηση της σχετικής πιθανότητας είναι $e^{1.5046} \simeq 4.5$. Αν για παράδειγμα ένας μαθητής διαβάσει για 2 ώρες τότε η πιθανότητα να επιτύχει είναι 0.26 αφού:

$$\text{πιθανότητα επιτυχίας στο διαγώνισμα} = \frac{1}{1 + e^{-(1.5046 \times 2 - 4.0777)}} = 0.26,$$

ενώ για κάποιον που έχει διαβάσει 4 ώρες η αντίστοιχη πιθανότητα είναι 0.87:

$$\text{πιθανότητα επιτυχίας στο διαγώνισμα} = \frac{1}{1 + e^{-(1.5046 \times 4 - 4.0777)}} = 0.87. \quad (5.31)$$

Επαναλαμβάνοντας την ίδια διαδικασία για περισσότερες ώρες προκύπτει ο Πίνακας 5.3. Στο συγκεκριμένο παράδειγμα χρησιμοποιήσαμε το κριτήριο Wald [32] για να ελέγξου-

Hours of study	Passing exam		
	Log-odds	Odds	Probability
1	-2.57	0.076 \approx 1:13.1	0.07
2	-1.07	0.34 \approx 1:2.91	0.26
3	0.44	1.55	0.61
4	1.94	6.96	0.87
5	3.45	31.4	0.97

Πίνακας 5.3: Πιθανότητες επιτυχίας με βάση διαφορετικές ώρες διαβάματος.

με την στατιστική σημαντικότητα του συντελεστή της μεταβλητής Hours. Πράγματι,

διαπιστώσαμε ότι η μεταβλητή αυτή λόγω της μικρής τιμής της p_{value} θεωρείται στατιστικά σημαντική. Γενικά, στόχος μας είναι η επιλογή του βέλτιστου μοντέλου, το μοντέλο δηλαδή που μπορεί να ερμηνεύσει την μεταβλητή απόκρισης με τις λιγότερο δυνατών μεταβλητές. Εκτός από το κριτήριο Wald υπάρχουν και άλλα κριτήρια τα οποία μας βοηθούν να επιλέξουμε ποιο χαρακτηριστικό (μεταβλητή) θα αφαιρεθεί από το μοντέλο βάση της στατιστικής σημαντικότητας του αντίστοιχου συντελεστή όπως το likelihood ratio test, conditional statistic κ.α. [27].

Με το παρόν Κεφάλαιο ολοκληρώσαμε την παρουσίαση των βασικών αλγορίθμων που χρησιμοποιούμε στην διαδικασία της ταξινόμησης. Σε αντίθεση με την Μπεύζιανή προσέγγιση που συναντήσαμε στο Κεφάλαιο 4, οι τεχνικές που περιγράψαμε στο Κεφάλαιο 5 στηρίζονται στην δημιουργία επιφανειών απόφασης που διαμερίζουν όσο το δυνατόν καλύτερα το χώρο των χαρακτηριστικών. Δεν υπάρχει κάποιος γενικός κανόνας για το ποια προσέγγιση είναι προτιμότερη κάθε φορά. Δοκιμάζοντας διάφορες τεχνικές στο πρόβλημα που καλούμαστε να επιλύσουμε, μπορούμε τελικά να επιλέξουμε αυτή που προσεγγίζει καλύτερα τα αποτελέσματα που επιθυμούμε.

Κεφάλαιο 6

Επίλογος

Στην σημερινή εποχή, με την εξάπλωση της χρήσης υπολογιστών σε όλους τους τομείς της ζωής μας, παρατηρείται εκθετική αύξηση των διαθέσιμων δεδομένων, η συγκέντρωση δηλαδή μεγάλου όγκου πληροφοριών. Στην μετατροπή των πληροφοριών σε χρήσιμη γνώση βοηθούν αποτελεσματικά οι τεχνικές μηχανικής μάθησης. Στόχος αυτής της διπλωματικής εργασίας ήταν η ανάλυση σημαντικών αλγορίθμων επιβλεπόμενης μηχανικής μάθησης και μεθόδων που βελτιώνουν την απόδοσή τους. Οι αλγόριθμοι και οι μέθοδοι που αναλύσαμε επιτρέπουν τους υπολογιστές να ‘μαθαίνουν’, δημιουργώντας έτσι προγράμματα που λειτουργούν με βάση την ανάλυση πολλών δεδομένων και όχι με ρητά προγραμματισμένες εντολές. Δημιουργούνται έτσι μηχανές που έχουν την ικανότητα της αίσθησης, κατανόησης, δράσης και μάθησης. Πλέον, μπορούν να αντιλαμβάνονται το περιβάλλον τους μέσω της επεξεργασίας εικόνων, ήχων, ομιλίας, γραπτού λόγου και πολλών άλλων δεδομένων και είναι ικανές να καταλαβαίνουν τις πληροφορίες που συλλέγουν (image, pattern recognition). Αφού γίνει η κατανόηση του περιβάλλοντος εκτελούν ενέργειες και τελικά βελτιώνουν την απόδοσή τους μέσα από τον βαθμό επιτυχίας/ αποτυχίας των ενεργειών τους. Έτσι λοιπόν, είναι εφικτό να λυθούν προβλήματα ιδιαίτερα πολύπλοκα όπου ο ανθρώπινος παράγοντας δύσκολα μπορεί να κατανοήσει και οι κλασικές προσεγγίσεις με μαθηματικά μοντέλα δεν μπορούν να αποδώσουν ικανοποιητικά.

Συνοπτικά, τα βήματα ενός συστήματος επιβλεπόμενης μάθησης είναι η συγκέντρωση δεδομένων εκπαίδευσης, ο καθαρισμός και η επεξεργασία αυτών. Στην συνέχεια, εξάγονται χαρακτηριστικά που αντιπροσωπεύουν το πρόβλημα και επιλέγοντας τον κατάλληλο αλγόριθμο μάθησης, τον εκπαιδεύουμε δημιουργώντας έτσι τον ταξινομητή που θα μας βοηθήσει στην λήψη των αποφάσεων. Όπως γνωρίζουμε για να επιλύσουμε ένα πρόβλημα επιβλεπόμενης ΜΜ μπορούμε να εφαρμόσουμε διαφορετικούς αλγορίθμους όπως Νευρωνικά Δίκτυα, Μπεϋζιανά δίκτυα, μηχανές διανυσμάτων υποστήριξης, δέντρα απόφασης και άλλοι που αναλύθηκαν διεξοδικά στην έκταση αυτής της διπλωματικής. Ωστόσο, κάποιοι από αυτούς τους αλγορίθμους μπορεί να είναι αποδοτικότεροι από κάποιους άλλους στην εκμάθηση κάποιων συναρτήσεων πρόγνωσης. Συνεπώς, η επιλογή του καταλληλότερου εξ' αυτών αποτελεί ένα πολύπλοκο πρόβλημα.

Το πρόβλημα της ταξινόμησης και οι αλγόριθμοι μηχανικής μάθησης και Τεχνητής Νοημοσύνης που εφαρμόζονται στην επίλυση του συνεισφέρουν σε μια πληθώρα επιστημονικών και βιομηχανικών έργων. Στην οικονομία είναι γνωστή η ανάγκη για όσο το δυνατόν πιο έγκυρες προβλέψεις. Συχνά λοιπόν χρησιμοποιούνται συστήματα τεχνητής νοημοσύνης για την πρόβλεψη της κίνησης της κεφαλαιαγοράς ώστε να γίνουν αποδοτικότερες επενδύσεις, αγορές μετοχών. Επίσης, σήμερα εταιρίες χρησιμοποιούν έξυπνα λογισμικά προκειμένου να προβλέψουν την αγορά, να προσδιορίσουν δηλαδή την καλύτερη πιθανή τιμή των προϊόντων και την επίδραση που αυτά θα έχουν στους ανθρώπους. Ακόμη, οι τράπεζες μέσω τεχνικών ΑΙ μπορεί να προβλέψουν την πιθανότητα αποπληρωμής δανείων από τους πελάτες βασίζομενοι σε χαρακτηριστικά των πελατών τους. Η βιομηχανία επωφελείται σε μεγάλο βαθμό από την τεχνητή νοημοσύνη καθώς ρομπότ με μηχανική όραση μπορούν να αντικαταστήσουν ανθρώπους σε επικίνδυνες εργασίες, αποδεικνύοντας μάλιστα ότι είναι περισσότερο αποδοτικά και ταχύτερα σε αυτοματοποιημένες ενέργειες.

Στην ιατρική, μέθοδοι μηχανικής μάθησης αποτελούν βασικό εργαλείο για την διαχείριση αποτελεσμάτων και εξαγωγή συμπερασμάτων χάρη στην μεγάλη διαθεσιμότητα κλινικών, γενετικών, παθολογικών χαρακτηριστικών. Ένα συχνό πρόβλημα ταξινόμησης είναι η κατηγοριοποίηση της κατάστασης των ασθενών ως προς κάποια ασθένεια και λύνεται συχνά με την βοήθεια αλγόριθμών και μεθόδων που αναλύσαμε.

Τα Μπεϋζιανά δίκτυα χρησιμοποιούνται σε πολλές ιατρικές εφαρμογές καθώς μπορούν να χειριστούν την γνώση με την αβεβαιότητα που προέρχεται από διαγνώσεις ασθενειών, προβλέψεις αποτελέσματος διάφορων φαρμάκων, διαδικασία επιλογής βέλτιστης θεραπείας κ.τ.λ. Παραδείγματα τέτοιων δικτύων αποτελούν πολλά ιατρικά έμπειρα συστήματα όπως το σύστημα NESTOR για την διάγνωση ενδοκρινολογικών διαταραχών, το σύστημα ALARM για την παρακολούθηση ασθενών κ.α. Επίσης, με την βοήθεια της γνώσης των ειδικών κατασκευάζονται κατανοητά και ορθά δέντρα απόφασης που αποτελούν βασικό συστατικό της ιατρικής διάγνωσης. Διάφορες κλινικές περιπτώσεις που αποτελούν το δείγμα εκπαίδευσης μπορούν να περιγραφούν από κάποια χαρακτηριστικά τα οποία μπορούν να αναπαρασταθούν με απλούς κανόνες και τελικά σχηματίζουν ένα δέντρο απόφασης. Παράδειγμα αυτής της μεθόδου είναι το σύστημα CARDIO που είναι σχεδιασμένο για την μέτρηση ECG. Τα Νευρωνικά Δίκτυα, που τα τελευταία χρόνια παρατηρείται έντονη ενασχόληση ερευνητών γύρω από αυτά, εφαρμόζονται με επιτυχία σε ένα ευρύ φάσμα εφαρμογών. Το έντονο αυτό ενδιαφέρον οφείλεται στην αποτελεσματική προσπάθεια τους να προσωμοιώσουν μαθηματικά την λειτουργία του ανθρώπινου εγκεφάλου. Το σύστημα Parnet βασίζεται στην εφαρμογή νευρωνικών δικτύων και μέσω αυτού είναι δυνατή η πρόωμη διάγνωση του καρκίνου του τραχήλου της μήτρας οδηγώντας έτσι σε πρωιμότερη θεραπεία. Το σύστημα αυτό έχει σχεδιαστεί ώστε να αναγνωρίζει τα μη φυσιολογικά κύτταρα σε ένα τεστ Παπανικολάου, χαρτογραφώντας τις περιοχές που αυτά βρίσκονται. Οι Μηχανές Διανυσμάτων Υποστήριξης εφαρμόζονται επίσης σε ιατρικά, βιολογικά δεδομένα. Χρησιμοποιούνται συχνά για την διάγνωση και πρόγνωση διάφορων παθολογικών περιπτώσεων. Οι Yu et. al. (2010) μελέτησαν την περίπτωση εμφάνισης σακχαρώδους διαβήτη χρησιμοποιώντας μηχανές διανυσμάτων υποστήριξης. Όπως και τα Νευρωνικά Δίκτυα, οι SVMs εφαρμόζονται για την διάγνωση καρκίνου με χαμηλότερο όμως ποσοστό ακρίβειας τις περισσότερες φορές.

Συμπερασματικά, η Τεχνητή Νοημοσύνη είναι ένας όρος ομπρέλα που αγκα-

λιάζει πολλούς διαφορετικούς τύπους Μηχανικής Μάθησης. Το πεδίο της εισάγει την ανθρωπότητα σε μία σύγχρονη περίοδο Αναγέννησης, όπως χαρακτηριστικά αναφέρεται, με τα ερευνητικά ιδρύματα και τις ιδιωτικές εταιρείες να επενδύουν μεγάλα ποσά για την ανάπτυξη του περιβάλλοντος και των εφαρμογών ΑΙ και να πιέζουν τα όρια της. Επέκταση των αλγορίθμων που περιγράψαμε σε αυτή την διπλωματική (που ανήκουν στην κατηγορία αδύναμη ΑΙ) αποτελούν τα Νευρωνικά Δίκτυα ισχυρής νοημοσύνης. Σε αντίθεση με την αδύναμη ΤΝ που είναι σε θέση να χειριστεί μόνο ένα συγκεκριμένο έργο, σκοπός της ισχυρής ΤΝ είναι να αντιμετωπίσει κάθε γενικευμένο έργο που της ζητείται, σχεδόν σαν να ήταν άνθρωπος. Πιο συγκεκριμένα, οι έρευνες προς αυτήν την κατεύθυνση προσπαθούν στην ανάπτυξη ενός 'ανθρώπινου' συστήματος που διαθέτει τις γνωστικές ικανότητες και τη γενική βιωματική κατανόηση του περιβάλλοντος που έχουμε εμείς οι άνθρωποι φτιάξει, σε συνδυασμό με την ικανότητα επεξεργασίας αυτών των δεδομένων με πολύ μεγαλύτερες ταχύτητες από το ανθρώπινο νου, δηλαδή περιέχει όχι μόνο συνείδηση της ύπαρξης και των επιλογών του αλλά και ταχύτατη διαδικασία απόφασης. Το σίγουρο είναι ότι το χάσμα ανθρώπινης και τεχνητής νοημοσύνης είναι αρκετά μεγάλο καθώς οι υπολογιστές μηχανές δεν μπορούν να αναπτύξουν αυτόνομη γνώση αλλά στηρίζονται σε προγραμματισμό που εισάγει ο άνθρωπος.

Βιβλιογραφία

- [1] Alpaydin, Ethem [2009], *Introduction to machine learning*, MIT press.
- [2] Anderson, James A [1995], *An introduction to neural networks*, MIT press.
- [3] Arlot, Sylvain and Celisse, Alain and others [2010], 'A survey of cross-validation procedures for model selection', *Statistics surveys* **4**, 40–79.
- [4] Barber, David [2012], *Bayesian reasoning and machine learning*, Cambridge University Press.
- [5] Ben Gal I [2007], 'Bayesian Networks.In Ruggeri F, Kennett RS, Faltin FW (eds.). *Encyclopedia of Statistics in Quality and Reliability*. John Wiley Sons'.
- [6] Bishop, Christopher M [2006], *Pattern recognition and machine learning*, Springer.
- [7] Cai, Eric [2014], 'Machine Learning Lesson of the Day-Overfitting and Underfitting'.
- [8] Carbonell, Jaime Guillermo and Mitchell, Tom Michael and Michalski, Ryszard Stanislaw [1984], *Machine learning: An artificial intelligence approach*, Springer-Verlag.
- [9] Christensen, Ronald [2006], *Log-linear models and logistic regression*, Springer Science & Business Media.
- [10] Cortes, Corinna and Vapnik, Vladimir [1995], 'Support-vector networks', *Machine learning* **20**(3), 273–297.
- [11] Cover, Thomas M [1965], 'Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition', *IEEE transactions on electronic computers* (3), 326–334.
- [12] Cover, Thomas M and Hart, Peter and others [1967], 'Nearest neighbor pattern classification', *IEEE transactions on information theory* **13**(1), 21–27.
- [13] Cramer, Jan Salomon [2002], 'The origins of logistic regression'.
- [14] Cristianini, Nello and Shawe-Taylor, John and others [2000], *An introduction to support vector machines and other kernel-based learning methods*, Cambridge university press.

- [15] Domingos, Pedro [2000], 'A unified bias–variance decomposition and its applications (Proc. of the Seventeenth International Conf. on Machine Learning, ICML'00)'.
- [16] Duda, Richard O and Hart, Peter E and Stork, David G [2012], *Pattern classification*, John Wiley & Sons.
- [17] Fausett, Laurene [1994], *Fundamentals of neural networks: architectures, algorithms, and applications*, Prentice-Hall, Inc.
- [18] Gardner, Howard [2011], *Frames of mind: The theory of multiple intelligences*, Hachette Uk.
- [19] Géron, Aurélien [2017], *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*, "O'Reilly Media, Inc."
- [20] Hastie, Trevor and Tibshirani, Robert and Friedman, Jerome and Franklin, James [2005], 'The elements of statistical learning: data mining, inference and prediction', *The Mathematical Intelligencer* **27**(2), 83–85.
- [21] Hawkins, Douglas M [2004], 'The problem of overfitting', *Journal of chemical information and computer sciences* **44**(1), 1–12.
- [22] Haykin, Simon [1999], 'Self-organizing maps', *Neural networks-A comprehensive foundation, 2nd edition*, Prentice-Hall .
- [23] Heckerman, David [2008], *A tutorial on learning with Bayesian networks*, Springer.
- [24] Hsu, Chih-Wei and Chang, Chih-Chung and Lin, Chih-Jen and others [2003], 'A practical guide to support vector classification'.
- [25] James, Gareth and Witten, Daniela and Hastie, Trevor and Tibshirani, Robert [2013], *An introduction to statistical learning*, Springer.
- [26] Johnson, Melvin and Schuster, Mike and Le, Quoc V and Krikun, Maxim and Wu, Yonghui and Chen, Zhifeng and Thorat, Nikhil and Viégas, Fernanda and Wattenberg, Martin and Corrado, Greg and others [2017], 'Google's multilingual neural machine translation system: Enabling zero-shot translation', *Transactions of the Association for Computational Linguistics* **5**, 339–351.
- [27] King, Gary [1998], *Unifying political methodology: The likelihood theory of statistical inference*, University of Michigan Press.
- [28] Kotsiantis, Sotiris B and Zaharakis, I and Pintelas, P [2007], 'Supervised machine learning: A review of classification techniques', *Emerging artificial intelligence applications in computer engineering* **160**, 3–24.
- [29] Le Borgne, Y [2005], 'Bias-variance trade-off characterization in a classification problem: What differences with regression', *Machine Learning Group, Univ. Libre de Bruxelles, Belgium* .

-
- [30] Manning, Christopher and Raghavan, Prabhakar and Schütze, Hinrich [2010], 'Introduction to information retrieval', *Natural Language Engineering* **16**(1), 100–103.
- [31] McDermott, Christopher D and Petrovski, Andrei [2017], 'Investigation of computational intelligence techniques for intrusion detection in wireless sensor networks.'
- [32] Menard, Scott [2002], *Applied logistic regression analysis*, Sige.
- [33] Mitchell, Tom M and others [1997], 'Machinelearning. 1997', *Burr Ridge, IL: McGraw Hill* **45**(37), 870–877.
- [34] Murphy, Kevin P [2012], *Machine learning: a probabilistic perspective*, MIT press.
- [35] Murty, M Narasimha and Devi, V Susheela [2011], *Pattern recognition: An algorithmic approach*, Springer Science & Business Media.
- [36] Ng, Hwee Tou and Goh, Wei Boon and Low, Kok Leong [1997], 'Feature selection, perceptron learning, and a usability case study for text categorization', **97**, 67–73.
- [37] Park, Hyeoun [2013], 'An introduction to logistic regression: from basic concepts to interpretation with particular attention to nursing domain', *Journal of Korean Academy of Nursing* **43**(2), 154–164.
- [38] Powers, David Martin [2011], 'Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation'.
- [39] Press, William H and Teukolsky, Saul A and Vetterling, William T and Flannery, Brian P [2007], 'Section 16.5. support vector machines', *Numerical recipes: the art of scientific computing* .
- [40] Quinlan, J. Ross [1987], 'Simplifying decision trees', *International journal of man-machine studies* **27**(3), 221–234.
- [41] Quinlan, JR [1986], 'Induction of Decision Trees. Mach. Learn'.
- [42] Ramasubramanian, Karthik and Singh, Abhishek [2017], *Model Performance Improvement*, Springer.
- [43] Ranganathan, Shoba and Nakai, Kenta and Schonbach, Christian [2018], *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, Elsevier.
- [44] Rojas, Raúl [2013], *Neural networks: a systematic introduction*, Springer Science & Business Media.
- [45] Russell, Ingrid [2016], 'The delta rule, University of Hartford', *Microcomputer Journal (From Wikipedia)* .
- [46] Russell, Stuart J and Norvig, Peter [2016], *Artificial intelligence: a modern approach*, Malaysia; Pearson Education Limited,.

-
- [47] Scholkopf, Bernhard and Smola, Alexander J. [2001a], *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, USA.
- [48] Scholkopf, Bernhard and Smola, Alexander J [2001β], *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press.
- [49] Snyman, Jan A and Wilke, Daniel N [2018], *Practical Mathematical Optimization: Basic Optimization Theory and Gradient-based Algorithms*, όλ. 133, Springer.
- [50] Stehman, Stephen V [1997], 'Selecting and interpreting measures of thematic classification accuracy', *Remote sensing of Environment* **62**(1), 77–89.
- [51] Stephenson, Todd Andrew [2000], 'An introduction to Bayesian network theory and usage'.
- [52] Tan, PN and Steinbach, M and Kumar, V [2006], 'Introduction to data mining, Addison Wesley Publishers'.
- [53] Theodoridis, Sergios and Pikrakis, Aggelos and Koutroumbas, Konstantinos and Cavouras, Dionisis [2010], *Introduction to pattern recognition: a matlab approach*, Academic Press.
- [54] Ting, Kai Ming and Sammut, C and Webb, GI [2011], 'Encyclopedia of machine learning', *KM Ting.–Boston, MA: Springer* .
- [55] Turing, Alan M [2009], 'Computing machinery and intelligence'.
- [56] Vijayakumar, Sethu [2007], 'The bias-variance tradeoff', *University Edinburgh Lecture notes* .
- [57] Webb, Andrew R [2003], *Statistical pattern recognition*, John Wiley & Sons.
- [58] Zhang, Zhongheng [2016], 'Naïve Bayes classification in R', *Annals of translational medicine* **4**(12).