ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόβλεψη Δεικτών Επίδοσης και Ποιότητας Υπηρεσιών Υπολογιστικού Νέφους με χρήση Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γιάννης Σ. Μπούρας

Επιβλέπουσα : Θεοδώρα Βαρβαρίγου

Καθηγήτρια Ε.Μ.Π.

Αθήνα, Οκτώβριος 2018

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόβλεψη Δεικτών Επίδοσης και Ποιότητας Υπηρεσιών Υπολογιστικού Νέφους με χρήση Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γιάννης Σ. Μπούρας

Επιβλέπουσα : Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 5η Οκτωβρίου 2018.

.....	.....	.....
Θεοδώρα Βαρβαρίγου Καθηγήτρια Ε.Μ.Π	Δημήτριος Ασκούνης Καθηγητής Ε.Μ.Π.	Συμεών Παπαβασιλείου Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβρης 2018

.....

Γιάννης Μπούρας

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

Περίληψη

Η Υπολογιστική Νέφους αποτελεί την πλέον σύγχρονη και απαραίτητη τεχνολογία στον κλάδο της Τεχνολογίας της Πληροφορίας (Information Technology - IT). Η ραγδαία ανάπτυξη της Υπολογιστικής Νέφους, έχει ως αποτέλεσμα την δημιουργία όλο και περισσότερων εφαρμογών και υπηρεσιών βασισμένων και εξαρτώμενων από το μοντέλο Λογισμικό-Πλατφόρμα-Υποδομή ή αλλιώς μοντέλο SPI (Software-Platform-Infrastructure). Ωστόσο η περιορισμένη ποσότητα πληροφοριών που μεταφέρονται μεταξύ των παραπάνω τριών στρωμάτων δημιούργησε νέες προκλήσεις σε παραδοσιακούς τομείς, όπως η εκτίμηση της επίδοσης των εφαρμογών και η αντιστοίχιση παραμέτρων λογισμικού στους απαιτούμενους υπολογιστικούς πόρους.

Πλέον όμως, με την ανάπτυξη της Μηχανικής Μάθησης ,μια τεχνολογία που επωφελείται από τη συνεχόμενη συλλογή και διάθεση μεγάλων ποσοτήτων πληροφορίας που χαρακτηρίζει την σημερινή εποχή, μας δίνεται η δυνατότητα να ξεπεράσουμε τα παραπάνω εμπόδια χρησιμοποιώντας εξελιγμένους αλγόριθμους μάθησης για την ακριβή πρόβλεψη των παραμέτρων ποιότητας υπηρεσίας (QoS) της εκάστοτε εφαρμογής. Τα τελευταία χρόνια η Μηχανική Μάθηση εξελίσσεται με ραγδαίους ρυθμούς και αποκτά ολοένα και ευρύτερο πεδίο εφαρμογής καθιστώντας τις έξυπνες μηχανές και εφαρμογές καθημερινό φαινόμενο, βοηθώντας μας έτσι να κάνουμε ακριβέστερες προβλέψεις και να λαμβάνουμε σοφότερες αποφάσεις.

Σε αυτή την εργασία, χρησιμοποιούνται αλγόριθμοι αιχμής από τον τομέα της Μηχανικής Μάθησης με σκοπό την πρόβλεψη παραμέτρων ποιότητας υπηρεσίας μιας εμπορικής εφαρμογής Διαχείρισης Πελατειακών Σχέσεων η οποία διατίθεται ως Λογισμικό ως Υπηρεσία (SaaS). Σκοπός της παρούσας διπλωματικής είναι, τόσο η πρόταση μιας προσέγγισης Μηχανικής Μάθησης στο τομέα της εκτίμησης επιδόσεων των σύγχρονων υπηρεσιών, όσο και η παρουσίαση μιας αναλυτικής συγκριτικής μελέτης των αλγορίθμων μάθησης που χρησιμοποιήθηκαν.

Στην διπλωματική εργασία μελετήθηκαν και χρησιμοποιήθηκαν για την εκτέλεση των πειραμάτων, τρεις βασικοί αλγόριθμοι στο τομέα της Ανάλυσης Παλινδρόμησης, τα Τεχνητά Νευρωνικά Δίκτυα, τα Τυχαία Δάση και Μηχανές Διανυσμάτων Υποστήριξης. Τα παραπάνω μοντέλα εκπαιδευτήκαν και ρυθμίστηκαν με σύγχρονους μεθόδους χρησιμοποιώντας το παραχωρηθέν σύνολο δεδομένων μιας εμπορικής εφαρμογής Διαχείρισης Πελατειακών Σχέσεων.

Λέξεις κλειδιά

Μηχανική Μάθηση, Ανάλυση Παλινδρόμησης, Υπολογιστική Νέφους,SPI,QoS, Λογισμικό ως Υπηρεσία, Τεχνητά Νευρωνικά Δίκτυα, Τυχαία Δάση , Μηχανές Διανυσμάτων Υποστήριξης

Abstract

Cloud Computing have become a state of the solution in the IT industry. With the rapid growth of Cloud Computing a more and more applications and services are based on the SPI model, Software-Platform-Infrastructure. However, the limited amount of information transferred between the above three layers has created new challenges in traditional areas of computing such us the application performance estimation and the mapping of software related parameters to the required computing resources.

But today, because of the constant growth of Machine Learning, a technology that benefits from the continuous collection and disposal of large amounts of information which characterizes today's era, we are given the opportunity to overcome these obstacles by using sophisticated learning algorithms to accurately predict Quality of Service parameters. In recent years, Machine Learning has evolved rapidly and is becoming increasingly wider in scope, making smart machines and applications a daily phenomenon and helping us to make more accurate forecasts and wiser decisions.

In this work, cutting-edge Machine learning algorithms are used to predict QoS parameters of a commercial Customer Relationship Management (CRM) application that is available as Software as a Service (SaaS). The aim of this diploma thesis is both to propose a Machine Learning approach in the field of performance assessment of modern Web Services and to present an analytical comparative study of the learning algorithms used.

In this thesis, three basic algorithms in the field of Regression Analysis, Artificial Neural Networks, Random Forests and Support Vector Machines were studied and used to perform the experiments. The above models were trained and configured using modern methods on the granted data set of a commercial Customer Relationship Management application

Key Words

Cloud Computing, IT, SPI, Quality of Service QoS, Customer Relationship Management (CRM), Machine Learning, Regression Analysis, Artificial Neural Networks, Random Forests, Support Vector Machines

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε στον τομέα Επικοινωνιών, Ηλεκτρονικής και Συστημάτων Πληροφορικής, στη σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Ηλεκτρονικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου.

Σε αυτό το σημείο θα ήθελα να απευθύνω θερμές ευχαριστίες στην επιβλέπουσα καθηγήτρια μου, κυρία Θεοδώρα Βαρβαρίγου για την καθοδήγηση και την ουσιαστική βοήθεια που μου παρείχε τόσο κατά τη διάρκεια των σπουδών μου στο Πολυτεχνείο, όσο και κατά τη διάρκεια της εκπόνησης αυτής της διπλωματικής εργασίας. Στη συνέχεια θα ήθελα να ευχαριστήσω εγκάρδια τον δόκτορα Φώτη Αίσωπο και τον υποψήφιο διδάκτορα Αλέξανδρο Ψύχα ,για την πολύτιμη βοήθεια και καθοδήγηση που μου παρείχαν κατά τη διάρκεια της συνεργασίας μας, βοηθώντας ουσιαστικά στη περάτωση της παρούσας διπλωματικής.

Ένα τεράστιο ευχαριστώ αξίζουν οι γονείς μου, αλλά και οι δύο αδερφές μου, Νάσια και Κατερίνα που με στηρίζουν όλα αυτά τα χρόνια και βρίσκονται πάντα δίπλα μου. Επίσης θα ήθελα να ευχαριστήσω όλους μου τους φίλους.

Τέλος δεν μπορώ να μην αναφερθώ στο διδακτικό προσωπικό του Εθνικού Μετσόβιου Πολυτεχνείου για τις γνώσεις και τις αρχές που μου μετέδωσαν κατά τη διάρκεια των σπουδών μου.

Γιάννης Μπούρας
Αθήνα, Οκτώβρης 2018

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

1 Εισαγωγή

Στη σύγχρονη εποχή το Διαδίκτυο, διακατέχει εξέχουσα θέση στον τρόπο ανάπτυξης και διάθεσης υπηρεσιών και εφαρμογών. Η Υπολογιστική Νέφους αποτελεί την πλέον διαδεδομένη και χρησιμοποιούμενη τεχνολογία στον τομέα του IT (Information Technology). Το μοντέλο SPI (Software -Platform-Infrastructure) οδήγησε στην καθιέρωση των SOA αρχιτεκτονικών στο τομέα της Σχεδίασης Λογισμικού. Η πληθώρα των προσφερόμενων υπηρεσιών σήμερα , διατίθενται μέσω Internet και συγκεκριμένα μέσω της τεχνολογίας του Υπολογιστικού Νέφους. Οι υπηρεσίες αυτές δεν περιορίζονται σε υπηρεσίες παροχής Λογισμικού αλλά ,όπως επιβάλλει και το μοντέλο SPI, περιλαμβάνουν ακόμα και την παροχή Υποδομών (Infrastructure).

Ένα παραδοσιακό πρόβλημα στον χώρο του IT είναι η πρόβλεψη των παράμετρών ποιότητας υπηρεσίας (QoS Parameters) των εφαρμογών. Επιπλέον, η αλλαγή της αρχιτεκτονικής των εφαρμογών και η μεταφορά τους στο Υπολογιστικό Νέφος δυσχέρανε ακόμη περισσότερο τη δυνατότητα πρόβλεψης της απόδοσης τους, καθώς η πληροφορία που μεταφέρεται από το ένα επίπεδο του μοντέλου SPI στο άλλο είναι περιορισμένη, συνήθως για λόγους εμπιστευτικότητας.

Στη σημερινή εποχή η ανάγκη για την ακριβή πρόβλεψη της ποιότητας υπηρεσίας των εφαρμογών είναι αυξημένη, τόσο για λόγους εξοικονόμησης πόρων, όσο και για την πιστή τήρηση των συμβολαίων υπηρεσίας SLAs , τονίζοντας έτσι την αναγκαιότητα για αποτελεσματικές μεθόδους πρόβλεψης του Quality of Service (QoS) a priori της δέσμευσης πόρων.

Ωστόσο η παραπάνω μετάβαση από τα τοπικά συστήματα και εφαρμογές σε υπηρεσίες διατιθέμενες μέσω της τεχνολογίας του Υπολογιστικού Νέφους στον παγκόσμιο ιστό, οδήγησε σε μια πραγματικά σημαντική αλλά και παράλληλα ακούσια συνέπεια. Πλέον μπορούμε να συλλέγουμε τεράστιες ποσότητες δεδομένων για σχεδόν τα πάντα που αφορούν την ανθρώπινη δραστηριότητα αλλά και να τα μεταφέρουμε με υψηλές ταχύτητες χάρη στην σημαντική εξέλιξη των δικτυακών πρωτοκόλλων αλλά και του αντίστοιχου απαιτούμενου υλισμικού. Αυτό οδήγησε στην ραγδαία άνθιση της Μηχανικής Μάθησης καθώς και στην αναδιοργάνωση της ως ξεχωριστό επιστημονικό πεδίο (ξεχωριστά από τη Τεχνητή Νοημοσύνη).

Η Μηχανική Μάθηση είναι ένας όρος που επινοήθηκε πρώτα από τον Αμερικανό ερευνητή της IBM Arthur Samuel το 1959. Ως επιστημονική προσπάθεια, η Μηχανική Μάθηση εξελίχθηκε από την αναζήτηση Τεχνητής Νοημοσύνης. Ωστόσο η σημαντική έλλειψη διαθέσιμων δεδομένων εκείνη την εποχή δεν βοήθησε στη ανάπτυξη και ευρύτερη εφαρμογή του συγκεκριμένου τομέα. Το γεγονός πως στην τωρινή περίοδο ,όπως εξηγήσαμε και παραπάνω, υπάρχει

άφθονη πληροφορία διαθέσιμη μέσω του Internet και άμεσα αξιοποιήσιμη , έχει οδηγήσει στην ραγδαία ανάπτυξη του τομέα της Μηχανικής Μάθησης που όλοι μας βιώνουμε σήμερα συνειδητά ή ασυνείδητα.

Σκοπός της Μηχανικής Μάθησης είναι η επίλυση προβλημάτων των οποίων η λύση είναι πολύ δύσκολο ή ακόμα και αδύνατο να προγραμματιστεί και να διατυπωθεί ρητά. Αντ. 'αυτού, παρέχουμε ένα μεγάλο όγκο δεδομένων σε έναν αλγόριθμο μάθησης και αφήνουμε τον αλγόριθμο να το επεξεργαστεί και να δημιουργήσει ένα μοντέλο που θα επιτύχει αυτό που οι προγραμματιστές του έχουν αναθέσει. Ορισμένα από τα παραπάνω δύσκολα προβλήματα είναι η αναγνώριση ήχου και εικόνας , η πρόβλεψη τιμών μετοχών στο χρηματιστήριο, αλλά και η ενίσχυση των αλγόριθμων που χρησιμοποιούν οι Μηχανές Αναζήτησης.

Στο πλαίσιο λοιπόν της παρούσας διπλωματικής εργασίας θα μελετήσουμε την εφαρμογής της Μηχανικής Μάθησης στο εξαιρετικά δύσκολο πρόβλημα της πρόβλεψης της απόδοσης των εφαρμογών (QoS Prediction).

1.1 Αντικείμενο της Διπλωματικής Εργασίας

Η συγκεκριμένη διπλωματική εργασία ,όπως έχει ήδη αναφερθεί παραπάνω, ασχολείται κυρίως με αλγόριθμους και τεχνολογίες από τον τομέα της Μηχανικής Μάθησης και ιδιαίτερα του επιστημονικού υποπεδίου αυτής, την Ανάλυση Παλινδρόμησης. Πεδίο εφαρμογής των παραπάνω τεχνολογιών θα είναι , η Υπολογιστική Νέφους και συγκεκριμένα εφαρμογές σχεδιασμένες και διατιθέμενες με βάση το μοντέλο αδειοδότησης και παράδοσης λογισμικού, Λογισμικό ως Υπηρεσία (SaaS).

Η παρούσα διπλωματική εργασία προτείνει τη χρήση και σύγκριση αλγορίθμων και τεχνικών Μηχανικής Μάθησης για την πρόβλεψη μετρικών ποιότητας υπηρεσίας εφαρμογών και υπηρεσιών του Υπολογιστικού Νέφους, αλλά και την αντιστοίχιση αυτών στον φόρτο εργασίας και στους διατιθέμενους υπολογιστικούς πόρους της εκάστοτε εφαρμογής. Η παρουσιαζόμενη λύση αξιολογείται για ένα πραγματικό και εμπορικό σύστημα Διαχείρισης Πελατειακών Σχέσεων, δημιουργώντας ένα σύνολο ρεαλιστικών εργασιών και μετρήσεων ποιότητας υπηρεσίας απεικονίζοντας την αποτελεσματικότητα της παρούσας προσέγγισης.

Η εξασφάλιση πόρων στο Υπολογιστικό Νέφος, είτε μέσω της χρήσης του μοντέλου Λογισμικό ως Υπηρεσία (SaaS) ,Πλατφόρμα ως Υπηρεσία (PaaS) είτε μέσω της παροχής Υποδομών ως Υπηρεσία (IaaS) δίνει τη δυνατότητα στον εκάστοτε ιδιοκτήτη μιας εφαρμογής να επωφεληθεί από τα πλεονεκτήματα που προσφέρει η Υπολογιστική Νέφους, όπως η επεκτασιμότητα και η αξιοπιστία. Ωστόσο σε πολλές περιπτώσεις ο πάροχος Λογισμικού ως Υπηρεσίας (SaaS)

μετατρέπεται σε υιοθετών υπηρεσιών του Υπολογιστικού Νέφους, όπως η Υποδομή ως Υπηρεσία (IaaS) προκειμένου να εξασφαλίσει τους απαραίτητους υπολογιστικούς πόρους για την εφαρμογή-υπηρεσία του. Η εξασφάλιση της ποιότητας υπηρεσίας της εφαρμογής επηρεάζεται σε μεγάλο βαθμό από τους διαθέσιμους υπολογιστικούς πόρους. Το κύρος προς στους πελάτες του, καθώς και η απαίτηση για την ικανοποίηση των πιθανών συμβολαίων σε επίπεδο υπηρεσίας με αυτούς (Service Level Agreement) καθιστά τη διαδικασία επιλογής υπολογιστικών πόρων εξαιρετικά σημαντική για τον πάροχο Λογισμικού ως Υπηρεσία. Ο παραδοσιακός τρόπος επιλογής των συγκεκριμένων παραμέτρων βασίζεται στην εμπειρία του εκάστοτε μηχανικού, σε ιστορικά δεδομένα ή ακόμα και στη μέθοδο Δοκιμής και Λάθους. Η καινοτομία της παρούσας εργασίας έγκειται στο γεγονός πως προτείνονται και χρησιμοποιούνται αλγόριθμοι Μηχανικής Μάθησης για την ανίχνευση των συσχετίσεων των παραμέτρων φόρτου εργασίας , υλισμικού και ποιότητας υπηρεσίας ώστε ο πάροχος του Λογισμικού ως Υπηρεσία να μπορεί να προβλέπει την συμπεριφορά της εφαρμογής του a priori της τελικής δέσμευσης πόρων στο Υπολογιστικό Νέφος. Το παραπάνω προσφέρει την απαραίτητη ενόραση που χρειάζεται ο κάτοχος της εφαρμογής ώστε να επιλέξει σοφά τους υπολογιστικού πόρους που θα δεσμεύσει.

Για την εξέταση της παραπάνω πρότασης ,χρησιμοποιήσαμε αλγόριθμους αιχμής στο πεδίο της επίλυσης προβλημάτων Ανάλυσης Παλινδρόμησης, όπως τα Νευρωνικά Δίκτυα ,τα Τυχαία Δάση και οι Μηχανές Διανυσμάτων Υποστήριξης τα οποία ρυθμίσαμε κατάλληλα με σύγχρονους αλγόριθμους Ρύθμισης Παραμέτρων. Τα παραπάνω μοντέλα εκπαιδεύτηκαν και αξιολογήθηκαν σε ένα σύνολο δεδομένων παραχθέν από μια σύγχρονη εφαρμογή Διαχείρισης Πελατειακών Σχέσεων της εταιρίας Cognity AE.

1.2 Οργάνωση κειμένου

Το κείμενο της παρούσας διπλωματικής εργασίας αποτελείται από 7 Κεφάλαια και 1 παράρτημα .

Το Κεφάλαιο 1, δηλαδή το παρόν Κεφάλαιο το οποίο αποτελεί την Εισαγωγή της διπλωματικής εργασίας.

Το Κεφάλαιο 2, στο οποίο παρουσιάζεται αναλυτικά το θεωρητικό υπόβαθρο της εργασίας. Πιο αναλυτικά, αρχικά γίνεται μια εισαγωγή στη Μηχανική Μάθηση και στα διάφορα είδη και υποπεδία που την απαρτίζουν. Στη συνέχεια παρουσιάζουμε αναλυτικά κάθε μια από τις χρησιμοποιούμενες τεχνικές Μηχανικής Μάθησης ,δηλαδή τα Τεχνητά Νευρωνικά Δίκτυα , τις Μηχανές Διανυσμάτων Υποστήριξης και τα Τυχαία Δάση, όπου γίνεται αρχικά η παρουσίαση των Δέντρων Απόφασης καθώς και ο τρόπος με τον οποίο

οδηγούμαστε από ένα απλό μοντέλο όπως το Δέντρο Απόφασης σε ένα περίπλοκο και αποδοτικό μοντέλο όπως είναι τα Τυχαία Δάση. Μετά την ανάλυση των Νευρωνικών Δικτύων παρουσιάζεται αναλυτικά η εξελικτική μέθοδος του Γενετικού Αλγόριθμου καθώς και πως αυτή χρησιμοποιείται για την ρύθμιση των υπερπαραμέτρων ενός Νευρωνικού Δικτύου.

Το Κεφάλαιο 3, στο οποίο παρουσιάζονται τα εργαλεία που χρησιμοποιήθηκαν για την πραγματοποίηση της παρούσας εργασίας. Συγκεκριμένα παρουσιάζουμε το ελεύθερο λογισμικό Gnu Octave το οποίο είναι το περιβάλλον ανάπτυξης του εργαλείου που χρησιμοποιήθηκε για την υλοποίηση των Τεχνητών Νευρωνικών Δικτύων ,των Mapping Models, ύστερα παρουσιάζεται η σύγχρονη βιβλιοθήκη της γλώσσας Python scikit-learn η οποία χρησιμοποιήθηκε για την δημιουργία των Τυχαίων Δασών και των Μηχανών Διανυσμάτων Υποστήριξης καθώς και για τη διαδικασία προπεξεργασίας των δεδομένων. Τέλος ,αναφέρουμε τη βιβλιοθήκη Pandas, επίσης της γλώσσας python, η οποία χρησιμοποιήθηκε για την διαχείριση και φόρτωση των συνόλων δεδομένων στα παραπάνω δύο μοντέλα.

Το Κεφάλαιο 4, όπου γίνεται αναλυτική παρουσίαση της υλοποίησης των μοντέλων. Πιο αναλυτικά παρουσιάζεται η δομή και ο τρόπος λειτουργίας των Mapping Models ,το εργαλείο που χρησιμοποιήθηκε για την δημιουργία ,τη ρύθμιση και την εκπαίδευση των Νευρωνικών Δικτύων. Στο συγκεκριμένο χωρίο γίνεται εκτενής ανάλυση του τρόπου με τον οποίο ο γενετικός αλγόριθμος χρησιμοποιήθηκε για τη ρύθμιση των υπερπαραμέτρων του Νευρωνικού Δικτύου καθώς και ποιες παράμετροι ρυθμίστηκαν. Στη συνέχεια παρουσιάζουμε τον τρόπο με τον οποίο χρησιμοποιήσαμε τη βιβλιοθήκη scikit-learn για την σχεδίαση και την εκπαίδευση των Τυχαίων Δασών και των Μηχανών Διανυσμάτων Υποστήριξης. Για κάθε ένα από αυτά τα μοντέλα γίνεται αναλυτική παρουσίαση του τρόπου με τον οποίο ρυθμίστηκαν οι παράμετροι καθώς και διαισθητικές παρατηρήσεις για τη σωστή ρύθμιση αυτών.

Το Κεφάλαιο 5, στο οποίο αρχικά παρουσιάζεται η εφαρμογή Διαχείρισης Πελατειακών Σχέσεων που χρησιμοποιήθηκε για τη δημιουργία του συνόλου δεδομένων. Παρουσιάζεται περιληπτικά η αρχιτεκτονική της αλλά και ο τρόπος με τον οποίο παράχθηκαν τα δεδομένα που χρησιμοποιήθηκαν για την εκτέλεση των πειραμάτων. Στη συνέχεια παρουσιάζεται εποπτικά η δομή και το περιεχόμενο του συνόλου δεδομένων καθώς και ο τρόπος χωρισμού του σε σύνολο εκπαίδευσης και σύνολο δοκιμής. Ακολουθεί η παρουσίαση των πειραμάτων η οποία χωρίζεται στη παρουσίαση των τελικών ρυθμισμένων υπερπαραμέτρων των τριών μοντέλων και στη συγκριτική μελέτη των μοντέλων μέσω της παρουσίασης των αποτελεσμάτων αξιολόγησης τους δίνοντας έμφαση στη διαγραμματική απεικόνιση αυτών.

Το Κεφάλαιο 6, σε αυτό το κεφάλαιο παρουσιάζουμε τα συμπεράσματα στα οποία καταλήξαμε ύστερα από την εκτίμηση και ανάλυση των πειραμάτων που εκτελέσαμε. Ταυτόχρονα γίνεται μια εκτενής αναφορά στις πιθανές μελλοντικές

ενέργειες, οι οποίες θα έχουν σκοπό την αναβάθμιση της έρευνας τόσο σε επίπεδο της συγκριτικής μελέτης μεταξύ των αλγόριθμων Μηχανικής Μάθησης, όσο και σε επίπεδο ποιότητας των τελικών αποτελεσμάτων.

Το Κεφάλαιο 7 το οποίο αποτελείται από την βιβλιογραφία που χρησιμοποιήθηκε για την σύνταξη του κειμένου της διπλωματικής εργασίας.

Τέλος στο Παράρτημα 1 υπάρχει ένα μέρος του πηγαίου κώδικα που χρησιμοποιήθηκε για την εκπαίδευση και ρύθμιση των μοντέλων, καθώς και ένας υπερσύνδεσμος για τον υπόλοιπο κώδικα και το σύνολο δεδομένων που χρησιμοποιήθηκε.

2 Θεωρητικό Υπόβαθρο

Στο κεφάλαιο αυτό γίνεται μια αναλυτική παρουσίαση του θεωρητικού υπόβαθρου στο οποίο βασίστηκε η παρούσα διπλωματική εργασία. Αρχικά γίνεται μια εισαγωγή στην Μηχανική Μάθηση ως ένα αυτοτελές επιστημονικό πεδίο. Παρουσιάζονται τα διάφορα είδη και οι διάφορες τεχνικές που χρησιμοποιούνται για την δημιουργία σύγχρονων αλγόριθμων μάθησης, καθώς και δίνονται διαισθητικές απαντήσεις στο ερώτημα «Γιατί χρησιμοποιούμε Μηχανική Μάθηση». Στη συνέχεια παρουσιάζεται μια αλγοριθμική και μαθηματική ανάλυση για κάθε ένα από έναν από τους τρεις αλγόριθμους Τεχνητά Νευρωνικά Δίκτυα, Τυχαία Δάση, Μηχανές Διανυσμάτων Υποστήριξης αλλά και σύγχρονων μεθόδων ρύθμισης υπερπαραμέτρων όπως οι Γενετικοί Αλγόριθμοι.

2.1 Μηχανική μάθηση

Η μάθηση όπως και η νοημοσύνη καλύπτουν ένα ευρύ φάσμα διαδικασιών και για αυτό το λόγο είναι πολύ δύσκολο να οριστούν με ακρίβεια. Το ανθρώπινο γένος έχει πορευτεί προσπαθώντας πάντα να κατανοήσει το περιβάλλον του χρησιμοποιώντας απλουστευμένα μοντέλα του κόσμου που τον περιβάλει. Η διαδικασία αυτή ονομάζεται επαγωγική μάθηση ή αλλιώς επαγωγή (induction). Όσον αφορά τις μηχανές ,μπορούμε να πούμε ότι μια μηχανή μαθαίνει ,κάθε φορά που αλλάζει η δομή της ,το πρόγραμμα ,καθώς και τα δεδομένα της (με βάση τις εισόδους που δέχεται από το περιβάλλον) με τέτοιο τρόπο ώστε στο αναμενόμενο μέλλον, η απόδοση της μηχανής να βελτιώνεται. Έχουν προταθεί διάφοροι τυπικοί ορισμοί για τη μηχανική μάθηση όπως αυτός του **Tom M. Mitchell**:

Ένα πρόγραμμα υπολογιστών λέγεται ότι «μαθαίνει» από την εμπειρία E σε σχέση με ένα σύνολο εργασιών T και μια μετρική απόδοσης σε αυτές τις εργασίες P ,αν η απόδοσή του στις εργασίες T όπως υπολογίζεται από το P ,βελτιώνεται με την εμπειρία E.

Συνεπώς , πιο περιγραφικά , η μηχανική μάθηση αποτελεί ένα υποσύνολο του τομέα της Τεχνητής Νοημοσύνης το οποίο προσδίδει στους υπολογιστές τη δυνατότητα να εκτελούν συγκεκριμένες διεργασίες «μαθαίνοντας από δεδομένα» χωρίς να προγραμματιστούν ρητά.

Η σημασία της μηχανικής μάθησης προκύπτει από το γεγονός πως ο άνθρωπος δεν είναι πάντα ικανός να σχεδιάζει ρητά, μηχανές που να λειτουργούν εξαρχής με τον επιθυμητό τρόπο. Μερικοί από τους λόγους χρήσης της μηχανικής μάθησης αναφέρονται παρακάτω:

- Ορισμένες εργασίες μπορούν να οριστούν καλά μόνο με τη χρήση του παραδείγματος, δηλαδή μπορούμε να καθορίσουμε ζεύγη εισόδου/εξόδου αλλά όχι μια μεστή σχέση μεταξύ των εισόδων και των αντίστοιχων επιθυμητών εξόδων. Συνεπώς εδώ είναι απαραίτητη η προσαρμογή των μηχανών – μοντέλων στα διαθέσιμα παραδείγματα - δεδομένα.
- Σε πολλές περιπτώσεις τα δεδομένα μπορεί να κρύβουν σημαντικές σχέσεις και συσχετισμούς. Η εξαγωγή αυτών των σχέσεων ονομάζεται εξόρυξη δεδομένων (data mining) και μπορεί να πραγματοποιηθεί με τη χρήση τεχνικών μηχανικής μάθησης.
- Τα περιβάλλοντα εργασίας των μηχανών και των μοντέλων αλλάζουν με την πάροδο του χρόνου. Μηχανές ικανές να προσαρμόζονται στις αλλαγές αυτές απαλλάσσονται από την ανάγκη του συχνού και κοστοβόρου επανασχεδιασμού τους.

Στην ανάπτυξη της μηχανικής μάθησης έχουν συμβάλλει αρκετά γνωστικά αντικείμενα όπως:

- ***Η στατιστική***
  Χαρακτηριστικό πρόβλημα στη στατιστική είναι η προσέγγιση της τιμής μιας άγνωστης συνάρτησης σε ένα σημείο, δοθέντος ενός συνόλου τιμών αυτής της συνάρτησης για διάφορα σημεία εισόδου. Μέθοδοι που χρησιμοποιούνται σε αυτά τα προβλήματα θεωρούνται τεχνικές μηχανικής μάθησης.

- ***Μοντέλα εγκεφάλου***
  Τα νευρωνικά δίκτυα αποτελούν μοντέλα μηχανικής μάθησης τα οποία προσπαθούν να προσεγγίσουν τον τρόπο με τον οποίο ο ανθρώπινος λειτουργεί και μαθαίνει.

- ***Ψυχολογικά μοντέλα***
  Έρευνες ψυχολόγων για το πώς τα ερεθίσματα ανταμοιβής επηρεάζουν την εκμάθηση συμπεριφοράς επιδίωξης στόχου στα ζώα, έχουν χρησιμοποιηθεί από τους επιστήμονες της μηχανικής μάθησης για την ανάπτυξη μοντέλων ενισχυτικής μάθησης(Reinforcement Learning).

- ***Τεχνητή Νοημοσύνη***
  Η μηχανική μάθηση ως υποσύνολο της τεχνητής νοημοσύνης παραμένει πάντα ως ένα από τα εργαλεία της τελευταίας , για την ανάπτυξη έξυπνων συστημάτων.

- ***Εξελικτικά μοντέλα***
  Στη φύση εκτός από την μάθηση ,η οποία βοηθάει τα ζώα και τους ανθρώπους να βελτιώνονται στις διάφορες εργασίες, σημαντικό ρόλο

διαδραματίζει και η εξέλιξη η οποία βοηθά τα ζώα να προσαρμόζονται στο περιβάλλον τους. Μοντέλα που μιμούνται τη βιολογική εξέλιξη ,όπως οι γενετικοί αλγόριθμοι , έχουν εφαρμοστεί με επιτυχία σε συστήματα μηχανικής μάθησης.

2.1.1 Είδη Μηχανικής Μάθησης

Υπάρχουν αρκετές τεχνικές μηχανικής μάθησης οι οποίες χρησιμοποιούνται ανάλογα με το είδος του προβλήματος που προσπαθούν να επιλύσουν αλλά όλες εμπίπτουν σε μια από τις παρακάτω κατηγορίες.

1. *Μάθηση με Επίβλεψη (Supervised Learning)*
2. *Μάθηση χωρίς Επίβλεψη (Unsupervised Learning)*
3. *Ενισχυτική Μάθηση (Reinforcement Learning)*

Μάθηση με Επίβλεψη

Στην μάθηση με επίβλεψη η μηχανή πρέπει να μάθει μια συνάρτηση «στόχο» (target function) η οποία θα εκφράζει το μοντέλο που περιγράφει τα δεδομένα. Συγκεκριμένα δοθέντων των τιμών της συνάρτησης στόχου f για ένα σύνολο σημείων-δειγμάτων m ,τα οποία ορίζουν το σύνολο εκπαίδευσης Ξ, υποθέτουμε πως αν βρούμε μια συνάρτηση, η οποία θα ονομάζεται «υπόθεση h» και θα προσεγγίζει καλά τη συνάρτηση στόχο για ένα αρκετά μεγάλο σύνολο των παραδειγμάτων τότε η h θα προσεγγίζει τη συνάρτηση στόχο και για περιπτώσεις που δεν έχει εξεταστεί. Συνεπώς, στόχος της Μάθησης με Επίβλεψη είναι h εύρεση της συνάρτησης που συσχετίζει τα δεδομένα εκπαίδευσης.

Τα προβλήματα που επιλύονται με Μάθηση με Επίβλεψη μπορούν κατηγοριοποιηθούν περαιτέρω σε δύο είδη προβλημάτων μάθησης ,στα *προβλήματα ταξινόμησης* και στα *προβλήματα ανάλυσης παλινδρόμησης*.

Ταξινόμηση (Classification)

Η κατασκευή μοντέλων που επιλύουν προβλήματα ταξινόμησης ,ανάγεται στη προσέγγιση μιας συνάρτησης αντιστοίχισης f για την οποία ισχύει f (X)=Y όπου X οι μεταβλητές εισόδου και Y είναι οι διακριτές μεταβλητές εξόδου του συνόλου εκπαίδευσης. Οι μεταβλητές εξόδου Y συνήθως ονομάζονται ετικέτες των δεδομένων ή κατηγορίες.

Υπάρχουν αρκετοί τρόποι να αξιολογήσουμε ένα μοντέλο ταξινόμησης αλλά το επικρατέστερο είναι ο υπολογισμός της ακρίβειας ταξινόμησης για το σύνολο των δειγμάτων εκπαίδευσης.

Ανάλυση Παλινδρόμησης (Regression Analysis)

Η κατασκευή μοντέλων που επιλύουν προβλήματα παλινδρόμησης ,ανάγεται στη προσέγγιση μιας συνάρτησης αντιστοίχισης f για την οποία ισχύει f(X)=Y όπου X οι μεταβλητές εισόδου και Y είναι οι συνεχείς μεταβλητές εξόδου του συνόλου εκπαίδευσης. Οι μεταβλητές εξόδου Y συνήθως αντιστοιχούν σε ποσότητες όπως ποσά και μεγέθη.

Υπάρχουν επίσης αρκετοί τρόποι αξιολόγησης ενός μοντέλου παλινδρόμησης αλλά ίσως το πιο συχνά χρησιμοποιούμενο είναι το κριτήριο της ρίζας του μέσου τετραγωνικού σφάλματος (RMSE).

Μάθηση χωρίς Επίβλεψη

Στη συγκεκριμένη κατηγορία προβλημάτων το σύνολο εκπαίδευσης αποτελείται μόνο από διανύσματα εισόδου χωρίς κάποια τιμή αντιστοίχισης για αυτά. Σκοπός σε αυτή τη περίπτωση είναι ο διαχωρισμός του συνόλου εκπαίδευσης σε υποσύνολα $\Xi_1, \dots, \Xi_R$ με κατάλληλο μη τυχαίο τρόπο. Συνεπώς οι αλγόριθμοι που ανήκουν στη συγκεκριμένη κατηγορία έχουν σκοπό την ανακάλυψη συσχετίσεων και ομάδων δεδομένων βασιζόμενοι μόνο στις ιδιότητες και τα χαρακτηριστικά των διανυσμάτων εισόδου.

Ενισχυτική Μάθηση

Η ενισχυτική μάθηση είναι ένας γενικός όρος που αναφέρεται σε τεχνικές που αφορούν τη κατασκευή πρακτόρων λογισμικού, οι οποίοι λειτουργούν με μοναδικό σκοπό την μεγιστοποίηση του συνολικού σήματος ενίσχυσης – ανταμοιβής. Πιο συγκεκριμένα ο αλγόριθμος δεν καθοδηγείται ούτε επιβλέπεται αλλά αναζητά τις ενέργειες εκείνες που θα του αποφέρουν το μεγαλύτερο κέρδος. Εφαρμογές της παραπάνω τεχνικής συναντάμε στη πλοήγηση robot ,στη μάθηση επιτραπέζιων παιχνιδιών, κτλ.

2.1.2 Ρύθμιση Υπερπαραμέτρων Αλγόριθμων Μηχανικής Μάθησης

Πολλοί από τους αλγόριθμους μάθησης βασίζονται στη σωστή ρύθμιση των παραμέτρων τους ώστε να λειτουργήσουν αποτελεσματικά.

Οι υπερπαράμετροι (Hyperparameters) είναι οι μεταβλητές εκείνες που διέπουν τη διαδικασία εκπαίδευσης. Για παράδειγμα, μέρος της δημιουργίας ενός νευρωνικού δικτύου είναι επιλογή του πλήθους των κρυφών επιπέδων μεταξύ

του στρώματος εισόδου και του στρώματος εξόδου καθώς και το πλήθος των κόμβων-νευρώνων που θα αποτελούν κάθε στρώμα. Αυτές οι μεταβλητές δε σχετίζονται με τα δεδομένα εκπαίδευσης. Πρόκειται για μεταβλητές διαμόρφωσης των μοντέλων (Configuration Variables). Σημειωτέο είναι πως οι υπερπαράμετροι παραμένουν σταθεροί κατά τη διάρκεια της εκπαίδευσης και κατά κάποιο τρόπο χαρακτηρίζουν το συγκεκριμένο μοντέλο.

Η επιλογή των σωστών υπερπαραμέτρων είναι καθοριστική για την αποδοτική λειτουργία του εκάστοτε μοντέλου. Ωστόσο δεν υπάρχουν συγκεκριμένοι κανόνες που να προσδιορίζουν πως θα γίνει η συγκεκριμένη επιλογή. Όλοι οι αλγόριθμοι που επιτελούν την ρύθμιση αυτών των παραμέτρων λειτουργούν σύμφωνα με τη μέθοδο Δοκιμής-Σφάλματος. Δηλαδή προβαίνουν σε συνεχόμενες προσαρμογές των υπερπαραμέτρων κατά τη διάρκεια πολλών εκπαιδευτικών κύκλων μέχρις ότου φτάσουν στις βέλτιστες τιμές.

Σημαντικό βήμα στη διαδικασία ρύθμισης των υπερπαραμέτρων είναι η επιλογή εκείνων που θα υποβληθούν στη διαδικασία της βελτιστοποίησης-ρύθμισης. Κάθε αλγόριθμος Μηχανικής Μάθησης μπορεί να περιέχει από καμία μέχρι και δεκάδες υπερπαραμέτρων. Ωστόσο υπάρχουν πολύ λίγες καθολικές συμβουλές για το πώς να επιλεγούν αυτές που θα ρυθμιστούν. Η επιλογή αυτών, πρέπει να γίνεται με σύνεση, καθώς κάθε παράμετρος που επιλέγεται να ρυθμιστεί μπορεί να αυξήσει εκθετικά τον αριθμό των δοκιμών που απαιτούνται για μια επιτυχημένη εργασία συντονισμού και εξαρτάται κυρίως από την εμπειρία του μηχανικού.

Αφού επιλεχθούν οι παράμετροι προς ρύθμιση εφαρμόζονται αλγόριθμοι οι οποίοι προελαύνουν τον χώρο αναζήτησης που δημιουργείται, το μέγεθος του οποίου εξαρτάται από το πλήθος και το εύρος των υπερπαραμέτρων που έχουν επιλεχθεί να ρυθμιστούν.

Αλγόριθμοι Ρύθμισης Υπερπαραμέτρων

- *Αναζήτηση Πλέγματος* (Grid Search)

  Η αναζήτηση πλέγματος αποτελεί τον πιο απλό αλγόριθμο βελτιστοποίησης υπερπαραμέτρων. Ο συγκεκριμένος αλγόριθμος εκτελεί μια εξαντλητική αναζήτηση στον προκαθορισμένο χώρο αναζήτησης που δημιουργείται. Η αναζήτηση πλέγματος όμως, πάσχει από την κατάρα των διαστάσεων (Curse Of Dimensionality),καθώς ο χώρος αναζήτησης μπορεί να καταλήξει να αποτελεί ένα υπερεπίπεδο δεκάδων διαστάσεων , ανάλογα πάντα με το πλήθος των προς ρύθμιση παραμέτρων. Ωστόσο, αποτελεί έναν τέλεια παράλληλο αλγόριθμο (Embarrassingly Parallel) λόγω του ότι οι ρυθμίσεις για τις διάφορες υπερπαραμέτρους συνήθως είναι ανεξάρτητες μεταξύ τους,.

- *Τυχαία Αναζήτηση* (Random Search)

  Η τυχαία αναζήτηση διασχίζει τυχαία τον χώρο αναζήτησης έως ότου ικανοποιηθεί ένα κριτήριο τερματισμού όπως ,ο αριθμός των επαναλήψεων ή η εύρεση ενός επαρκώς καλού μοντέλου. Ο συγκριμένος αλγόριθμος δεν εγγυάται πως θα βρει τη βέλτιστη λύση αλλά λειτουργεί ικανοποιητικά σε προβλήματα που το πλήθος των υπερπαραμέτρων είναι μικρό.

- *Μπεϋζιανή Βελτιστοποίηση* (Bayesian Optimization)

  Έστω $f(X)$ το σφάλμα του εξεταζόμενου μοντέλου ρυθμισμένο με την πλειάδα υπερπαραμέτρων X. Όπως έχουμε αναφέρει σκοπός μας είναι η εύρεση του ελαχίστου της f σε κάποιο οριακό σύνολο X. Η Μπεϋζιανή Βελτιστοποίηση κατασκευάζει ένα πιθανοτικό μοντέλο για την f από προηγούμενες παρατηρήσεις - εκτελέσεις πειραμάτων για διαφορετικές τιμές υπερπαραμέτρων. Χρησιμοποιώντας το συγκεκριμένο μοντέλο, του οποίου η ελαχιστοποίηση είναι τετριμμένη ,επιλέγεται η επόμενη πλειάδα παραμέτρων προς εξέταση. Η βασική φιλοσοφία είναι η αξιοποίηση όλης της διαθέσιμης πληροφορίας από προηγούμενες εκτελέσεις του αλγόριθμου και η έξυπνη επιλογή της επόμενης πλειάδας X η οποία δε θα εξαρτάται μόνο από τη τοπική κλίση και της Χεσιανές προσεγγίσεις

- *Εξελικτική Βελτιστοποίηση* (Evolutionary Optimization)

  Στην Εξελικτική Βελτιστοποίηση χρησιμοποιούνται εξελικτικοί αλγόριθμοι για την διερεύνηση του χώρου αναζήτησης και την εύρεση της βέλτιστης λύσης. Οι εξελικτικοί αλγόριθμοι είναι αλγόριθμοι εμπνευσμένοι από τη βιολογικοί εξέλιξη και χρησιμοποιούν τεχνικές όπως η μετάλλαξη ,η αναπαραγωγή, ο συνδυασμός και η επιλογή.

2.2 Δέντρα Απόφασης

Οι αλγόριθμοι μάθησης ή επαγωγής δέντρων απόφασης είναι από τους πιο δημοφιλείς αλγόριθμους μάθησης και έχουν εφαρμοστεί αποτελεσματικά σε διάφορους τομείς, όπως η διάγνωση ιατρικών περιστατικών, ή η πρόβλεψη της συμπεριφοράς καταναλωτή. Είναι μια μέθοδος που χρησιμοποιεί ένα Δέντρο Απόφασης για να μεταβεί από τις παρατηρήσεις για ένα στοιχείο (που αντιπροσωπεύεται στους κλάδους) σε συμπεράσματα σχετικά με τη τιμή του στόχου του αντικειμένου (που αντιπροσωπεύεται στα φύλλα).Τα δέντρα απόφασης χρησιμοποιούνται για να προβλέψουν με κάποιο βαθμό ακρίβειας την τιμή της μεταβλητής που μοντελοποιούν , με βάση τις τιμές των θεωρούμενων ανεξάρτητων μεταβλητών (χαρακτηριστικών).

Τα δέντρα αποφάσεων που χρησιμοποιούνται στην εξόρυξη δεδομένων είναι δύο βασικών τύπων:

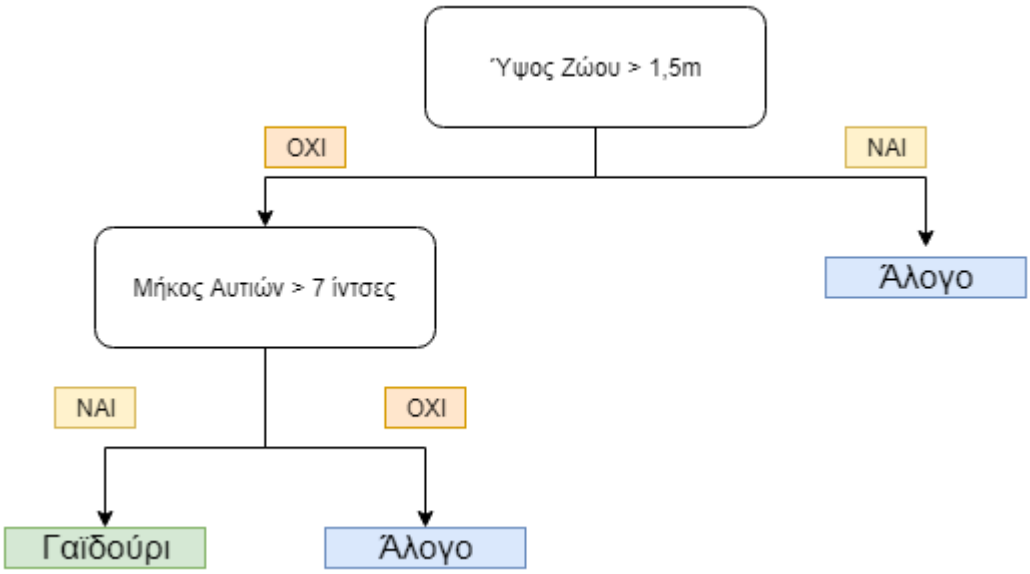
- **Δέντρα ταξινόμησης ,** τα οποία χρησιμοποιούνται όταν το προβλεπόμενο αποτέλεσμα είναι η κλάση-κατηγορία στην οποία ανήκουν τα δεδομένα.

- **Δέντρα παλινδρόμησης ,** τα οποία χρησιμοποιούνται όταν το προβλεπόμενο αποτέλεσμα μπορεί να θεωρηθεί πραγματικός αριθμός (π.χ. η τιμή ενός σπιτιού ή διάρκεια εκτέλεσης μια διαδικασίας).

Ο όρος **Classification And Regression Tree (CART)** χρησιμοποιείται όταν αναφερόμαστε στη εφαρμογή δέντρων απόφασης, τόσο στη ανάλυση παλινδρόμησης όσο και στη ταξινόμηση δεδομένων και εισήχθη από τον **Leo Breiman**.

Η μάθηση με χρήση δέντρων απόφασης βασίζεται στη δημιουργία ενός δέντρου χρησιμοποιώντας ένα κατάλληλα διατεταγμένο σύνολο πλειάδων-περιπτώσεων. Κάθε κόμβος στο δέντρο ορίζει μια συνθήκη ελέγχου της τιμής κάποιου χαρακτηριστικού (attribute ή feature) των περιπτώσεων (του συνόλου δεδομένων) και κάθε κλαδί που φεύγει από τον κόμβο αυτό , αντιστοιχεί σε μια διαφορετική διακριτή τιμή του χαρακτηριστικού αυτού. Αρχίζοντας από τη ρίζα, σε κάθε κόμβο ελέγχεται η τιμή του στιγμιότυπου εισόδου για το χαρακτηριστικό του κόμβου και ακολουθείται το αντίστοιχο κλαδί προς κάποιο φύλλο του δέντρου, το οποίο περιέχει την πρόβλεψη της εξόδου η οποία είναι μια διακριτή τιμή της κατηγορίας της περίπτωσης αν πρόκειται για δέντρο ταξινόμησης, ή μια συνεχή τιμή της μεταβλητής εξόδου η οποία αντιπροσωπεύει την πρόβλεψη για τη συγκεκριμένη περίπτωση, αν πρόκειται για πρόβλημα Ανάλυσης Παλινδρόμησης.

Παρακάτω φαίνεται ένα δέντρο απόφασης που δημιουργήθηκε για το πρόβλημα ταξινόμησης που περιλαμβάνει δύο κατηγορίες, τα Άλογα και τα Γαϊδούρια. Το σύνολο δεδομένων περιλαμβάνει στιγμιότυπα με δύο χαρακτηριστικά, το ύψος του ζώου και το μήκος των αυτιών του.



Εικόνα 2-1 Παράδειγμα Ταξινόμησης με τη χρήση Δέντρων Απόφασης

2.3 Αλγοριθμικό πλαίσιο για Δέντρα Απόφασης

Οι επαγωγικοί αλγόριθμοι για τα δέντρα απόφασης είναι αλγόριθμοι που κατασκευάζουν αυτόματα μια απόφαση από ένα συγκεκριμένο σύνολο δεδομένων. Συνήθως ο στόχος είναι να βρεθεί η βέλτιστη απόφαση με την ελαχιστοποίηση του σφάλματος. Ωστόσο, μπορούν να οριστούν και άλλες συναρτήσεις στόχοι, για παράδειγμα, την ελαχιστοποίηση του αριθμού των κόμβων ή την ελαχιστοποίηση του μέσου βάθους του δέντρου.

Η επαγωγή μιας βέλτιστης απόφασης από ένα σύνολο δεδομένων θεωρείται ένα δύσκολο έργο. Ο **Hancock** (1996) έχει δείξει ότι η εξεύρεση ενός ελάχιστου δέντρου απόφασης το οποίο να συνάδει με το σύνολο των δεδομένων εκπαίδευσης είναι NP-Hard, ενώ οι **Hyafil** και **Rivest** (1976) έδειξαν ότι η κατασκευή ενός δυαδικού δέντρου απόφασης ελάχιστο ως προς τον αναμενόμενο αριθμό των δοκιμών που απαιτούνται για την ταξινόμηση ενός ενδεχόμενου είναι NP-Complete. Ακόμη και η εύρεση του ελάχιστου ισοδύναμου δέντρου απόφασης ,**Zantema και Bodlaender (2000),** ή η κατασκευή του βέλτιστου δέντρου απόφασης από πίνακες αποφάσεων είναι γνωστό ότι είναι NP-Hard **Naumov (1991).**

Αυτά τα αποτελέσματα δείχνουν ότι η χρήση ενός βέλτιστου αλγόριθμου δένδρου απόφασης είναι εφικτή μόνο σε μικρά προβλήματα. Κατά συνέπεια, απαιτούνται ευρετικοί αλγόριθμοι για την επίλυση του προβλήματος. Σε γενικές

γραμμές, αυτές οι μέθοδοι χωρίζονται σε δύο ομάδες:

- Αλγόριθμοι από πάνω προς τα κάτω (Top-Down)
- Αλγόριθμοι από κάτω προς τα πάνω (Bottom-Up)

με σαφή προτίμηση ωστόσο στη βιβλιογραφία για την πρώτη ομάδα.

Υπάρχουν διάφοροι αλγόριθμοι για επαγωγικά δέντρα απόφασης από πάνω προς τα κάτω όπως ID3 **[Quinlan(1986)]**, C4.5 **[Quinlan (1993)],** CART **[Breiman et al. (1984)]**. Μερικοί επαγωγικοί αλγόριθμοι αποτελούνται από δύο εννοιολογικές φάσεις: Ανάπτυξη του δέντρου και «κλάδεμα» (C4.5 και CART). Άλλοι επαγωγικοί αλγόριθμοι εκτελούν μόνο τη φάση ανάπτυξης.

Ο αλγόριθμος ID3 κατασκευάζει το δέντρο άπληστα από πάνω προς τα κάτω επιλέγοντας αρχικά το πιο κατάλληλο χαρακτηριστικό για έλεγχο στη ρίζα. Η επιλογή βασίζεται σε κάποιο στατιστικό μέτρο που υπολογίζεται από τα δεδομένα. Ένας από τους πιο διαδεδομένους μηχανισμούς διαχωρισμού και αυτός που χρησιμοποιείται από τον ID3, είναι αυτός της εντροπίας της πληροφορίας ο οποίος επιλέγει εκείνη την ανεξάρτητη μεταβλητή που οδηγεί σε περισσότερο συμπαγές δέντρο. Ο αλγόριθμος C4.5 και CART αποτελούν από τις περισσότερο διαδεδομένες βελτιώσεις του ID3 χρησιμοποιώντας τεχνικές κλαδέματος ,πριν την ολοκλήρωση της κατασκευής του δέντρου.

Σημειώστε ότι αυτοί οι αλγόριθμοι είναι άπληστοι από τη φύση τους και κατασκευάζουν το δέντρο απόφασης σε ένα top-down, αναδρομικό τρόπο (επίσης γνωστή ως διαίρει και βασίλευε). Σε κάθε επανάληψη, ο αλγόριθμος θεωρεί το διαμέρισμα του συνόλου των εκπαιδευτικών δεδομένων, χρησιμοποιώντας το αποτέλεσμα των διακριτών χαρακτηριστικών εισόδου. Η επιλογή του κατάλληλου χαρακτηριστικού γίνεται με τη χρήση διάφορων συναρτήσεων κόστους. Μετά την επιλογή μιας κατάλληλης διάσπασης, κάθε κόμβος υποδιαιρεί τα δεδομένα εκπαίδευσης σε μικρότερα υποσύνολα, μέχρι ένα κριτήριο τερματισμού “stopping criteria” να ικανοποιείται.

Ο Αλγόριθμος CART

Ο αλγόριθμος δέντρων ταξινόμησης και παλινδρόμησης (CART) είναι ένας από τους πιο δημοφιλείς και επιτυχημένους αλγόριθμους επαγωγικής διαδικασίας δέντρων απόφασης. Η αναπαράσταση του μοντέλου CART είναι ένα δυαδικό δέντρο όπου κάθε εσωτερικός κόμβος είναι ένα σημείο διάσπασης ως προς ένα χαρακτηριστικό του συνόλου δεδομένων. Κάθε κόμβος «φύλλο» περιέχει την μεταβλητή εξόδου η οποία χρησιμοποιείται για να γίνει μια πρόβλεψη.

Η δημιουργία ενός δυαδικού δέντρου με τη χρήση του εν λόγω αλγόριθμου βασίζεται στην επιλογή χαρακτηριστικών (μεταβλητών εισόδου) και σημείων διαχωρισμού σε αυτά τα χαρακτηριστικά. Η επιλογή αυτών των μεταβλητών

καθώς και των αντίστοιχων σημείων κοπής γίνεται «άπληστα» προσπαθώντας να ελαχιστοποιηθεί μια συνάρτηση κόστους. Η κατασκευή του δέντρου σταματά όταν ικανοποιηθεί ένα κριτήριο τερματισμού όπως για παράδειγμα ,ο ελάχιστος αριθμός περιπτώσεων που έχουν ανατεθεί σε έναν κόμβο «φύλλο» του δέντρου.

Η δημιουργία ενός δυαδικού δέντρου είναι στην ουσία μια διαδικασία διαχωρισμού του χώρου εισόδου. Η άπληστη στρατηγική που χρησιμοποιείται ονομάζεται αναδρομικός δυαδικός διαχωρισμός (recursive binary splitting). Αυτή είναι μια αριθμητική διαδικασία κατά την οποία όλες τιμές των χαρακτηριστικών εισόδου παρατάσσονται και διαφορετικά σημεία κοπής αξιολογούνται σύμφωνα με συνάρτηση κόστους όπως έχουμε ήδη αναφέρει. Το σημείο με το καλύτερο κόστος επιλέγεται.

Σε προβλήματα ανάλυσης παλινδρόμησης, η συνάρτηση κόστους που ελαχιστοποιείται είναι συνήθως το μέσο αθροιστικό σφάλμα των προβλέψεων, έτσι όπως εμπίπτουν από τη διαδικασία διαχωρισμού του χώρου εισόδου από το επιλεγμένο σημείο κοπής ,για όλα τα δείγματα εκπαίδευσης n που υπάγονται στον εκάστοτε κόμβο του δυαδικού δέντρου.

$$\frac{1}{n} * \sum_{n}^{1} [(y - p)\text{^}2]$$

Όπου y η μεταβλητή εξόδου των περιπτώσεων εκπαίδευσης και p οι τιμές των προβλέψεων του μοντέλου.

Σε προβλήματα ταξινόμησης ακολουθείται ακριβώς η ίδια διαδικασία με μόνη διαφορά στη συνάρτηση κόστους που βελτιστοποιείται σε κάθε διαχωρισμό του χώρου εισόδου. Στη συγκεκριμένη περίπτωση χρησιμοποιείται ο συντελεστής Gini ο οποίος είναι μέτρο ανομοιογένειας των δεδομένων που αντιστοιχούν σε κόμβο του δέντρου. Ο δείκτης Gini για ένα σύνολο από J κλάσεις ταξινόμησης δίνεται από το τύπο:

$$I_G(p) = \sum_{i=1}^{J} p_i \sum_{k \neq i} p_k = \sum_{i=1}^{J} p_i(1 - p_i) = \sum_{i=1}^{J} (p_i - {p_i}^2) = \sum_{i=1}^{J} p_i - \sum_{i=1}^{J} {p_i}^2 = 1 - \sum_{i=1}^{J} {p_i}^2$$

Όπου p_i το ποσοστό των περιπτώσεων εκπαίδευσης που ανήκουν στη κλάση i.

Ένας κόμβος με περιπτώσεις μιας μόνο κλάσης πετυχαίνει την απόλυτη ομοιογένεια και θα έχει Gini δείχτη G=0 ενώ ένας κόμβος ,με περιπτώσεις που είναι ισοκατανεμημένες σε όλες τις διαφορετικές κλάσεις που υπάρχουν στα δεδομένα του συγκεκριμένου κόμβου, θα έχει Gini δείχτη G=0.5.

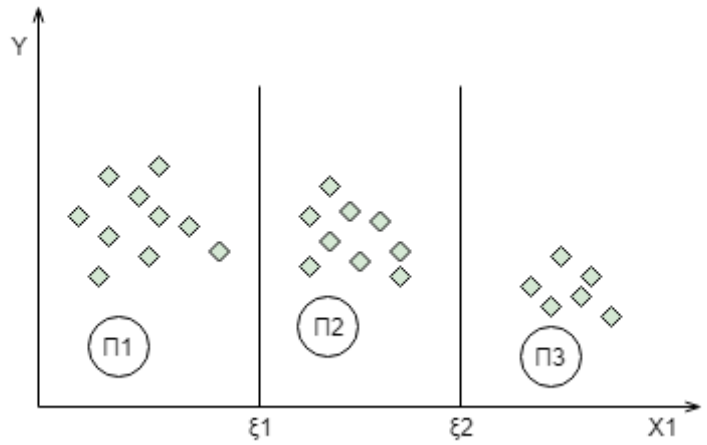
Συνεπώς κατά τη δημιουργία του δέντρου απόφασης, σε κάθε σημείο κοπής του

χώρου εισόδου αντιστοιχίζεται μια συνάρτηση κόστους ελαχιστοποίησης η οποία είναι ο σταθμισμένος μέσος των δεικτών Gini των δύο κόμβων «παιδιών» που θα δημιουργηθούν αν επιλεγεί το συγκεκριμένο σημείο κοπής. Ο δείκτης Gini κάθε κόμβου σταθμίζεται από τον αριθμό των περιπτώσεων που ανήκουν στο κόμβο «πατέρα». Ο τύπος για τον υπολογισμό της συνάρτησης κόστους Gini ή αλλιώς Gini score είναι:

$$G_{score} = IG_1 * \frac{ng1}{n} + IG2 * \frac{ng2}{n}$$

Όπου $IG_1, IG2$ είναι οι δείκτες Gini των δύο κόμβων που δημιουργούνται αν επιλέξουμε το συγκεκριμένο σημείο κοπής και ng1,ng2 είναι το συνολικό πλήθος των περιπτώσεων που ανήκουν στους δύο κόμβους «παιδιά» αντίστοιχα.

Παρακάτω φαίνεται ο τρόπος με το οποίον ο αλγόριθμος CART διαχωρίζει ένα απλό σύνολο δεδομένων εισόδου σε τρεις περιοχές. Όπως διακρίνεται και από το σχήμα πρόκειται για πρόβλημα παλινδρόμησης που τα δεδομένα ορίζονται από ένα χαρακτηριστικό εισόδου X και μια μεταβλητή εξόδου Y. Σε κάθε μια περιοχή θα αντιστοιχεί μια τιμή (συνήθως ο σταθμισμένος μέσος των μεταβλητών εξόδου των δειγμάτων της περιοχής) ως η τιμή πρόβλεψης y_π για κάθε νέο στιγμιότυπο που θα υπάγεται στη εκάστοτε περιοχή.



Εικόνα 2-2 Αλγόριθμος CART. Παράδειγμα διαχωρισμού συνόλου δεδομένων

2.3.1 Οι Τεχνικές του Bootstrap Aggregating και Bagging

Η μέθοδος της «ενίσχυσης» ή αλλιώς Boosting και η μέθοδος της ενσάκισης ή αλλιώς Bagging είναι τεχνικές που ανήκουν στην ευρύτερη οικογένεια των αλγόριθμων συνολικής μάθησης (Ensemble Methods). Οι μέθοδοι συνολικής

μάθησης χρησιμοποιούν πολλαπλούς αλγόριθμους μάθησης για να επιτύχουν καλύτερη απόδοση από αυτή που θα μπορούσε να επιτευχθεί από τους υποψήφιους αλγόριθμους ξεχωριστά.

Bootstrap Aggregating

Η μέθοδος την ενσάκισης είναι μια μέθοδο συνολικής μάθησης σχεδιασμένη να μειώνει την διακύμανση των προβλέψεων και το overfitting. Αποτελεί μια μέθοδο «ψήφου» όπου οι αδύναμοι μηχανισμοί εκμάθησης διαφοροποιούνται εκπαιδεύοντάς τους σε σύνολα δεδομένων που διαφέρουν ελάχιστα μεταξύ τους. Με τη μέθοδο Bootstrap προκύπτουν τα L δείγματα ως εξής : Από το σύνολο δεδομένων εκπαίδευσης X μεγέθους n , λαμβάνεται ένα τυχαίο δείγμα μεγέθους n΄, με επανάθεση. Λόγω της δειγματοληψίας με επανάθεση είναι πιθανό ορισμένες παρατηρήσεις να επιλεγούν περισσότερες από μια φορές και άλλες να μην επιλεγούν ποτέ. Κατόπιν ,οι αδύναμοι μηχανισμοί εκμάθησης έστω ε_j εκπαιδεύονται στα L δείγματα X_j , j = 1 . . . , L ,χρησιμοποιώντας μια ασταθή διαδικασία εκμάθησης, και στο τέλος συνδυάζονται ώστε να προκύψει η τελική πρόβλεψη.

Η μέθοδος Bagging εφαρμόζεται τόσο σε προβλήματα ταξινόμησης όσο και παλινδρόμησης. Στη περίπτωση της ταξινόμησης χρησιμοποιείται η διαδικασία της ψηφοφορίας ενώ στη περίπτωση της παλινδρόμησης χρησιμοποιείται ο η διάμεσος των επιμέρους προβλέψεων για τον σχηματισμό της τελικής πρόβλεψης.

Boosting

Οι αλγόριθμοι Boosting χτίζουν πολλαπλά μοντέλα από ένα σύνολο δεδομένων. Η βασική ιδέα του Boosting (και η κύρια διαφορά της με τη μέθοδο της ενσάκισης) είναι η σύνδεση κάθε παρατήρησης του συνόλου δεδομένων με ένα βάρος.

Χτίζεται λοιπόν ένα πλήθος μοντέλων και τα βάρη ενισχύονται, εξ ‘ου και ο αγγλικός όρος Boosted, εάν ένα μοντέλο ταξινομεί λανθασμένα την παρατήρηση. Τα βάρη των παρατηρήσεων γενικά παρουσιάζουν διακυμάνσεις προς τα πάνω και προς τα κάτω από το ένα μοντέλο στο επόμενο. Το τελικό μοντέλο είναι αθροιστικό, κατασκευασμένο από μία ακολουθία μοντέλων, με το αποτέλεσα καθενός εξ ’αυτών να έχει λάβει κάποιο βάρος.

Τα πλεονεκτήματα της μεθόδου Boosting είναι ότι απαιτείται μικρή προσαρμογή (tuning) και ότι η μόνη υπόθεση που κάνουμε για τον κατασκευαστή του μοντέλου είναι ότι πρέπει να είναι σχετικά αδύναμος. Τα μειονεκτήματα του Boosting είναι ότι μπορεί να αποτύχει είτε όταν τα δεδομένα δεν είναι επαρκή, είτε όταν τα αδύναμα μοντέλα είναι εξαιρετικά πολύπλοκα, επίσης ένα άλλο μειονέκτημα της συγκεκριμένης μεθόδου πως είναι ευαίσθητη στα τυχαία σφάλματα.

2.4 Τυχαία Δάση

Τα Τυχαία Δάση ή Τυχαία Δάση απόφασης είναι μια μέθοδος συνολικής μάθησης (ensemble method) που χρησιμοποιείται σε προβλήματα ταξινόμησης και ανάλυσης παλινδρόμησης. Ο συγκεκριμένος αλγόριθμος μάθησης λειτουργεί με τη κατασκευή ενός πλήθους δέντρων αποφάσεων κατά τη διάρκεια εκπαίδευσης και τη συλλογή των ατομικών προβλέψεων κάθε δέντρου για την παραγωγή της τελικής πρόβλεψης του μοντέλου. Πιο συγκεκριμένα στη περίπτωση της ταξινόμησης η έξοδος του τυχαίου δάσους είναι η επικρατούσα τιμή (mode) ,ενώ στη παλινδρόμηση είναι η μέση πρόβλεψη των εξόδων, των μεμονωμένων δέντρων. Τα Τυχαία Δάση διορθώνουν το φαινόμενο της υπερπροσαρμογής στο σύνολο εκπαίδευσης από το οποίο πάσχουν τα δέντρα απόφασης. Συγκεκριμένα, τα δέντρα που αναπτύσσονται πολύ βαθιά (δέντρα με μεγάλο ύψος),τείνουν να «μαθαίνουν» άκρως ακανόνιστα μοτίβα με αποτέλεσμα να έχουν χαμηλή μεροληψία (bias) αλλά πολύ υψηλή διακύμανση (variance).

Ο αλγόριθμος των Τυχαίων Δασών δημιουργήθηκε πρώτα από τον **Tin Kam Ho** και επεκτάθηκε από τον **Leo Breiman** εισάγοντας την ιδέα της ενσάκισης (bagging) και τη τυχαία επιλογή των χαρακτηριστικών του συνόλου δεδομένων.

2.4.1 Αλγοριθμικό πλαίσιο Τυχαίων Δασών

Όπως έχει ήδη αναφερθεί τα Τυχαία Δάση είναι ένας τρόπος να εξομαλυνθούν και να γενικευτούν οι αποκρίσεις βαθιών Δέντρων Απόφασης. Η παραπάνω διαδικασία επιτυγχάνεται με τη χρήση δύο βασικών ιδεών:

- *Την ενσάκιση των δέντρων απόφασης(Tree Bagging)*
- *Την τυχαία επιλογή χαρακτηριστικών(Random subset of Features)*

2.4.1.1 Η Διαδικασία του Bagging στα Τυχαία Δάση

Ο αλγόριθμος εκπαίδευσης των Τυχαίων Δασών εφαρμόζει την τεχνική του bootstrap aggregating ή Bagging η οποία αναλύθηκε παραπάνω. Πιο συγκεκριμένα, δοθέντος ενός συνόλου δεδομένων εκπαίδευσης έστω X=$x_1, \dots, x_n$ και με μεταβλητές εξόδου Y = $y_1, \dots, y_n$ η ενσάκιση επιλέγει επαναλαμβανόμενα (έστω Β φορές) ,με επανατοποθέτηση, ένα τυχαίο δείγμα από το σύνολο δεδομένων και με κάθε ένα από αυτά εκπαιδεύει διαφορετικά δέντρα αποφάσεων.

Για b = 1, ..., B:

1) *Πάρε, με αντικατάσταση, n δείγματα εκπαίδευσης από τα X, Y;*

2) *Εκπαίδευσε ένα δέντρο ταξινόμησης ή παλινδρόμησης f_b χρησιμοποιώντας τα παραπάνω δείγματα X_b, Y_b;*

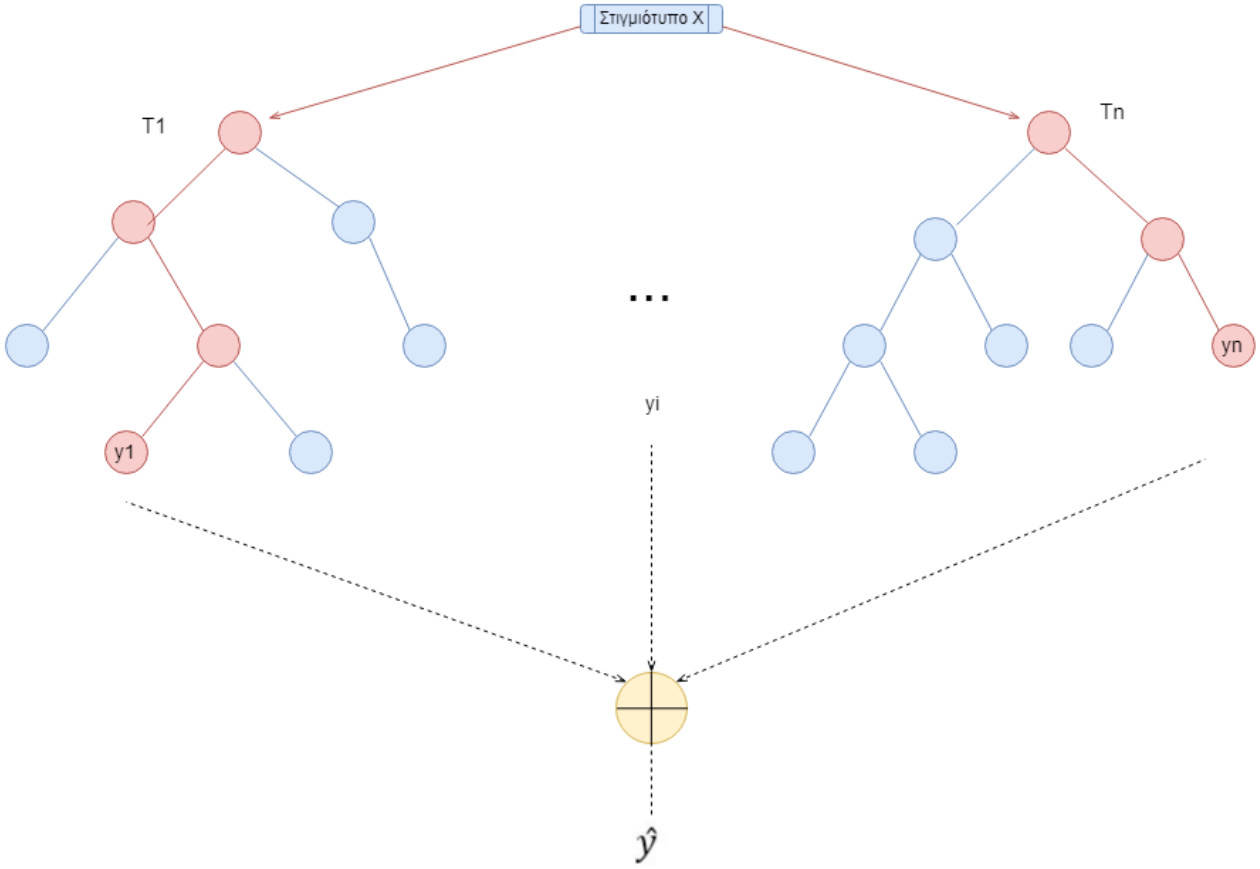
Ο παραπάνω ψευδοαλγόριθμος παρουσιάζει τη διαδικασία εκπαίδευσης ενός τυχαίου δάσους.

Μετά την ολοκλήρωση της εκπαίδευσης οι προβλέψεις του τυχαίου δάσους, $\hat{f}$, γίνονται παίρνοντας το μέσο όρο των προβλέψεων των δέντρων παλινδρόμησης (Ανάλυση Παλινδρόμησης)

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(x')$$

ή χρησιμοποιώντας την επικρατούσα τιμή των προβλέψεων των δέντρων ταξινόμησης (Ταξινόμηση).

Παρακάτω απεικονίζεται ο τρόπος με τον οποίο ένα Τυχαίο Δάσος παράγει τις προβλέψεις του. Τα στιγμιότυπα εισόδου τροφοδοτούνται σε κάθε Δέντρο Απόφασης του δάσους και οι αποκρίσεις τους συνδυάζονται ώστε να δημιουργηθεί η τελική τιμή πρόβλεψης της μεταβλητής εξόδου. Η τελική πρόβλεψη παράγεται από την επικρατούσα τιμή των επιμέρους προβλέψεων, αν πρόκειται για πρόβλημα ταξινόμησης, ή τον μέσο όρο αυτών, αν πρόκειται για Ανάλυση Παλινδρόμησης.



Εικόνα 2-3 Ανάκληση στα Τυχαία Δάση

2.4.1.2 Τυχαία επιλογή χαρακτηριστικών στα Τυχαία Δάση

Η παραπάνω διαδικασία αναλύει τον τρόπο που τα Τυχαία Δάση χρησιμοποιούν τη διαδικασία της ενσάκισης για να εκπαιδεύσουν ένα σύνολο από δέντρα απόφασης.

Στις περισσότερες εφαρμογές των τυχαίων δασών και ειδικότερα σε περιπτώσεις που το σύνολο δεδομένων περιέχει πολλά χαρακτηριστικά (predictors), γίνεται χρήση μιας ακόμα τεχνικής συνολικής μάθησης (ensemble learning), η οποία ονομάζεται τυχαία μέθοδο υποσυνόλου ή ενσάκιση χαρακτηριστικών (Feature Bagging). Πιο συγκεκριμένα χρησιμοποιείται ένας τροποποιημένος αλγόριθμος εκπαίδευσης δέντρων, που επιλέγει ένα τυχαίο υποσύνολο των χαρακτηριστικών εισόδου, ώστε να βρεθεί το βέλτιστο σημείο κοπής του χώρου εκπαίδευσης , κατά τη διαδικασία εκμάθησης των μεμονωμένων δέντρων απόφασης του Τυχαίου Δάσους.

Ο λόγος που χρησιμοποιείται η συγκεκριμένη τεχνική είναι η πιθανή συσχέτιση των μεμονωμένων δέντρων απόφασης. Στη περίπτωση που ένα ή μερικά χαρακτηριστικά του χώρου εκμάθησης αποτελούν ισχυρούς παράγοντες πρόβλεψης για τη μεταβλητή απόκρισης, αυτά τα χαρακτηριστικά θα επιλεγούν ως εσωτερικοί κόμβοι «κοπής» από τη πλειοψηφία των δέντρων, προκαλώντας έτσι μια ανεπιθύμητη συσχέτιση μεταξύ αυτών, η οποία επιδρά αρνητικά στη δυνατότητα γενίκευσης των Τυχαίων Δασών. Το πλήθος και η τυχαία κατανομή που θα ακολουθήσει η επιλογή των χαρακτηριστικών αποτελούν σημαντικές παραμέτρους ρύθμισης των Τυχαίων Δασών.

2.4.2 Επιλογή και Εξαγωγή χαρακτηριστικών με τη χρήση Τυχαίων Δασών

Συχνά στην επιστήμη των δεδομένων συναντώνται σύνολα δεδομένων τα οποία περιέχουν εκατοντάδες ή ακόμα και χιλιάδες χαρακτηριστικά (Features). Στις περισσότερες όμως περιπτώσεις το μοντέλο που θα δημιουργηθεί είναι επιθυμητό να περιλαμβάνει τα σημαντικότερα χαρακτηριστικά και όχι το σύνολο αυτών. Υπάρχουν τρεις βασικοί λόγοι για να γίνει αυτό. Πρώτον, τα μοντέλα γίνονται πιο απλά και συνεπώς είναι πιο εύκολο να ερμηνευτούν. Επίσης μειώνοντας το πλήθος των χαρακτηριστικών στα σύνολα εκμάθησης μειώνουμε τη διακύμανση των μοντέλων (variance) και συνεπώς το φαινόμενο της υπερ-προσαρμοστηκότητας (overfitting). Τέλος, το μοντέλο που δημιουργείται έχοντας ένα πιο απλό σύνολο εκμάθησης (λιγότερα χαρακτηριστικά) συγκλίνει γρηγορότερα αφού ο χρόνος εκπαίδευσης μειώνεται αισθητά.

Τα χαρακτηριστικά με τη μεγαλύτερη βαρύτητα θεωρούνται εκείνα που «τακτοποιούν» ή χαρακτηρίζουν καλύτερα το σύνολο δεδομένων επηρεάζοντας σε μεγαλύτερο βαθμό τη μεταβλητή εξόδου κάθε περίπτωσης (instance).

Τα Τυχαία Δάση χρησιμοποιούνται συχνά για την επιλογή χαρακτηριστικών σε προβλήματα Μηχανικής Μάθησης και εξόρυξης δεδομένων. Όπως έχει ήδη αναφερθεί σκοπός των αλγόριθμων κατασκευής δέντρων απόφασης είναι ο διαδοχικός διαχωρισμός του συνόλου εκμάθησης ,με τη χρήση κατάλληλων σημείων κοπής, σε υποσύνολα με όσο το δυνατόν μεγαλύτερη καθαρότητα (purity). Συνήθως οι τεχνικές με τις οποίες μετριέται αυτή η καθαρότητα είναι το κύριο χαρακτηριστικό που κάνει αυτούς τους αλγόριθμους να διαφέρουν μεταξύ τους.

Λόγω της άπληστης φύσης αυτών των αλγόριθμων τα χαρακτηριστικά εκείνα που μειώνουν την πρόσμιξη (impurity) σε κάθε κόμβο θα επιλέγονται πρώτα, και συνεπώς θα αντιστοιχούν σε κόμβους κοντινούς στη ρίζα του δέντρου. Συνεπώς εκπαιδεύοντας ένα δέντρο η μείωση της πρόσμιξης που επιφέρει κάθε

χαρακτηριστικό ,σε κάθε κόμβο-σύνολο ,μπορεί να υπολογιστεί από κατάλληλους δείκτες μέτρησης όπως ο Gini δείκτης ή η εντροπία που έχουμε αναφέρει στο κεφάλαιο των δέντρων απόφασης.

Έχοντας λοιπόν υπολογίσει τη μέση τιμή των δεικτών αυτών για κάθε χαρακτηριστικό-κόμβο στα μεμονωμένα δένδρα μπορούμε να δημιουργήσουμε ένα υποσύνολο των πιο σημαντικών χαρακτηριστικών.

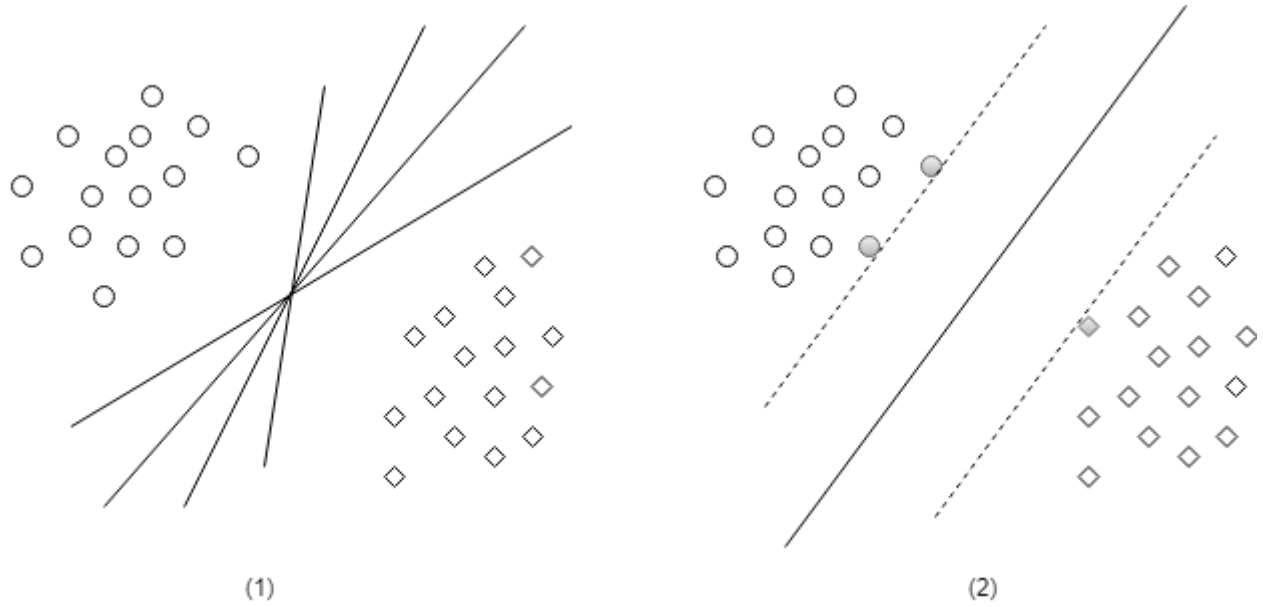
2.5 Μηχανές Διανυσμάτων Υποστήριξης

Οι Μηχανές Διανυσμάτων Υποστήριξης είναι ένα μοντέλο που ανήκει στη κατηγορία των αλγόριθμων επιβλεπόμενης μηχανικής μάθησης το οποίο χρησιμοποιείται τόσο για ταξινόμηση δεδομένων, όσο και για Ανάλυση Παλινδρόμησης. Το συγκεκριμένο μοντέλο εκτός από την εκτέλεση γραμμικής ταξινόμησης δηλαδή τον διαχωρισμό γραμμικά διαχωρίσιμων κλάσεων, μπορεί εξίσου αποδοτικά να διαχωρίσει μη γραμμικά διαχωρίσιμες κλάσεις αντικειμένων, χρησιμοποιώντας τις λεγόμενες **συναρτήσεις πυρήνα** (Kernel Functions).

Διαισθητικά, το μοντέλο των Μηχανών Διανυσμάτων Υποστήριξης, ή αλλιώς ταξινομητής μέγιστου περιθωρίου ,όπως συχνά ονομάζεται, προσπαθεί να διαχωρίσει τα δεδομένα εισόδου με το καλύτερο δυνατό τρόπο ,για το σκοπό αυτό, κατασκευάζει ένα υπερεπίπεδο το οποίο έχει τη μεγαλύτερη απόσταση από τα πλησιέστερο σημείο εκπαίδευσης οποιασδήποτε κλάσης των δεδομένων εισόδου, το λεγόμενο λειτουργικό περιθώριο (Functional Margin) διαχωρίζοντας έτσι τον χώρο εισόδου με το βέλτιστο τρόπο. Όσο μεγαλύτερη είναι αυτή η απόσταση τόσο μικρότερο θα είναι το σφάλμα γενίκευσης του τελικού μοντέλου.

Όταν το παραπάνω υπερεπίπεδο έχει καθοριστεί μαθηματικά, δηλαδή όταν το μοντέλο έχει εκπαιδευτεί, τότε δοθέντος ενός νέου σημείου μπορούμε εύκολα να προβλέψουμε τη κλάση ανάλογα αν συνάρτηση του υπερεπιπέδου δίνει τιμή μεγαλύτερη ή μικρότερη του μηδενός.

Χάρη στην απλή υλοποίηση, αλλά και το συνδυασμό υψηλής ταχύτητας εκπαίδευσης με πολύ μικρό σφάλμα γενίκευσης, οι Μηχανές Διανυσμάτων Υποστήριξης αποτελούν από τους πλέον δημοφιλείς αλγόριθμους μηχανικής μάθησης με εφαρμογές σε πλήθος προβλημάτων, όπως η ταξινόμηση εικόνων και η κατηγοριοποίηση κειμένου.



Εικόνα 2-4 1)Μη Βέλτιστες ευθείες διαχωρισμού
2)Βέλτιστη ευθεία διαχωρισμού

Όπως φαίνεται από το παραπάνω *σχήμα (1)* υπάρχουν πολλά διαφορετικά υπερεπίπεδα, (στη περίπτωση των 2 διαστάσεων ευθείες), που μπορούν να διαχωρίσουν δύο γραμμικά διαχωρίσιμες κλάσεις δεδομένων.

Ωστόσο οι Μηχανές Διανυσμάτων Υποστήριξης διαχωρίσουν το χώρο εισόδου με το βέλτιστο δυνατό τρόπο μεγιστοποιώντας το λειτουργικό περιθώριο. Τα πρότυπα του συνόλου δεδομένων που απέχουν την ελάχιστη απόσταση από το υπερεπίπεδο διαχωρισμού, ονομάζονται διανύσματα υποστήριξης, όπως φαίνεται και από *το σχήμα (2)* της παραπάνω εικόνας.

2.5.1 Μαθηματική Ανάλυση των Μηχανών Διανυσμάτων Υποστήριξης

Έστω το δυαδικό πρόβλημα ταξινόμησης που ορίζεται από το διάνυσμα εισόδου $\boldsymbol{x}$ και τις ετικέτες των κλάσεων y ∈ {−1, 1}. Υποθέτοντας αρχικά πως οι δύο αυτές κλάσεις δεδομένων είναι γραμμικά διαχωρίσιμες, θα υπάρχει ένα διάνυσμα $\boldsymbol{w}$ και ένα κατώφλι w_0 τέτοιο ώστε για κάθε στιγμιότυπο εκπαίδευσης (**x ,** y) ναι ισχύει :

$$\boldsymbol{w}^T\boldsymbol{x} + w_0 \begin{cases} < 0 \ \alpha\nu \ y = 1 \\ > 0 \ \alpha\nu \ y = -1 \end{cases}$$

Δηλαδή θα υπάρχει ένα υπερεπίπεδο τέτοιο ώστε να διαχωρίζει της κλάσεις

δεδομένων. Στη περίπτωση των χώρου των δύο διαστάσεων ένα τέτοιο υπερεπίπεδο είναι μια ευθεία.

Γίνεται γρήγορα όμως σαφές πως δεν υπάρχει μια μόνο λύση σε αυτό το πρόβλημα καθώς υπάρχουν άπειρα ζεύγη $(\boldsymbol{w}, w_0)$ που ικανοποιούν την παραπάνω συνθήκη.

Σε αυτό το σημείο είναι χρήσιμο να εισάγουμε τις έννοιες του λειτουργικού καθώς και του γεωμετρικού περιθωρίου ταξινόμησης.

Λειτουργικό Περιθώριο

Δοθέντος ενός στιγμιότυπου εκπαίδευσης $(\boldsymbol{x}^{(i)}, y^{(i)})$ ορίζουμε το λειτουργικό περιθώριο σε σχέση με το στιγμιότυπου εκπαίδευσης ως:

$$\hat{\gamma}^{(i)} = y^{(i)}(\boldsymbol{w}^T \boldsymbol{x}^{(i)} + w_0).$$

Όταν το $y^{(i)} = 1$,δηλαδή όταν το στιγμιότυπο εκπαίδευσης ανήκει στη θετική κλάση y=1 τότε, αν το λειτουργικό περιθώριο είναι μεγάλο δηλαδή αν ο όρος $\boldsymbol{w}^T \boldsymbol{x}^{(i)} + w_0$ είναι ένας αρκετά μεγάλος θετικός αριθμός τότε η πρόβλεψη μας θα έχει υψηλό βαθμό βεβαιότητας καθώς το δείγμα μας θα βρίσκεται χωρίς μεγάλη αμφιβολία στα θετικά δείγματα. Αντίστοιχα για τη περίπτωση που το $y^{(i)} = -1$.

Γενικεύοντας ,δοθέντος ενός συνόλου εκπαίδευσης S = $\{\boldsymbol{x}^{(i)}, y^{(i)}$); i = 1, . . . , m} ορίζεται ως λειτουργικό περιθώριο του συνόλου S ως το μικρότερο των μεμονωμένων περιθωρίων των στιγμιότυπων εκπαίδευσης:

$$\hat{\gamma} = \min_i \hat{\gamma}^{(i)}$$

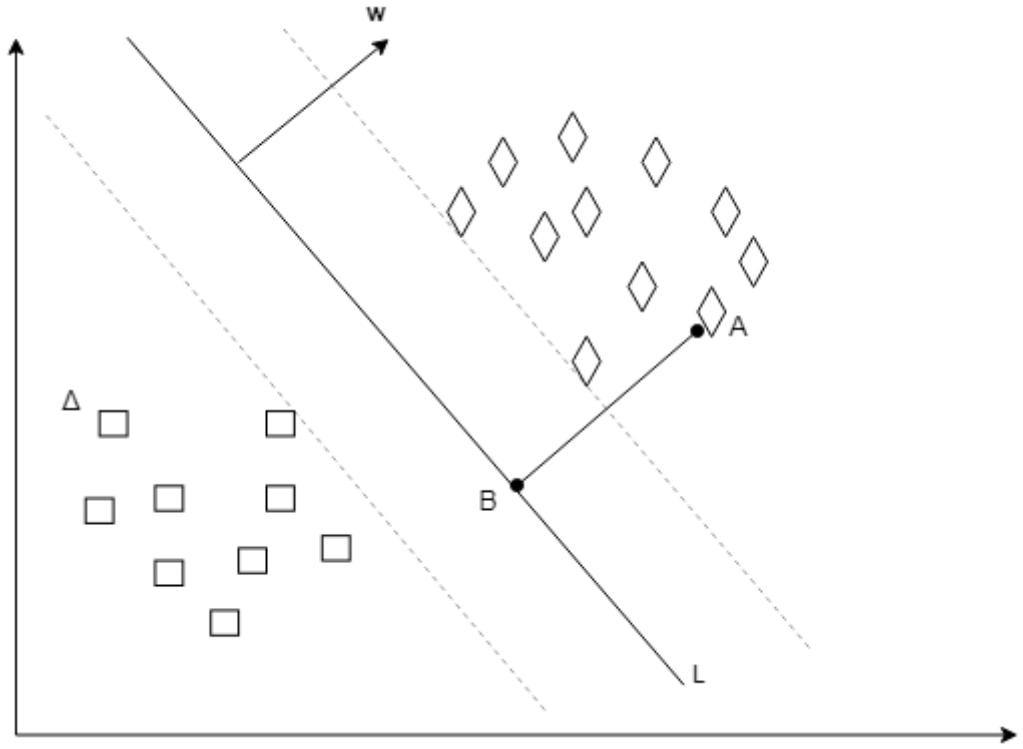
Εκ πρώτης όψεως η εξίσωση $\boldsymbol{w}^T \boldsymbol{x} + w_0 = 0$ φαίνεται να προσδιορίζει το υπερεπίπεδο διαχωρισμού των κλάσεων δεδομένων και το λειτουργικό περιθώριο να αποτελεί ένα αξιόπιστο μέτρο της βεβαιότητας των προβλέψεων. Παρατηρώντας όμως την παραπάνω εξίσωση διαπιστώνουμε ότι περιέχει αρκετό πλεονασμό και δεν προσδιορίζει μια συγκεκριμένη λύση, καθώς τα ζεύγη $(\boldsymbol{w}, w_0)$, $(\alpha \boldsymbol{w}, \alpha w_0)$ περιγράφουν ακριβώς το ίδιο υπερεπίπεδο στο χώρο και μάλιστα για κάθε τιμή της σταθεράς α.

Επιπλέον τα ζεύγη $(\boldsymbol{w}, w_0)$, $(\alpha \boldsymbol{w}, \alpha w_0)$ μας οδηγούν σε διαφορετικά λειτουργικά περιθώρια. Συνεπώς εκμεταλλευόμενοι τον βαθμό ελευθέριας της εξίσωσης μπορούμε να καταλήξουμε σε υπερβολικά μεγάλα λειτουργικά περιθώρια χωρίς να αλλάζει κάτι ουσιώδες στο πρόβλημα μας.

<u>Οι παραπάνω παρατηρήσεις μας οδηγούν διαισθητικά στο συμπέρασμα πως ίσως χρειάζεται να κανονικοποιήσουμε την εξίσωσή μας επιβάλλοντας συγκεκριμένους περιορισμούς.</u>

<u>Γεωμετρικό Περιθώριο</u>

Η έννοια του γεωμετρικού περιθωρίου είναι ευκολότερο να παρουσιαστεί σχηματικά.



Εικόνα 2-5 Γεωμετρικό Περιθώριο

Έστω L η διαχωριστική ευθεία των δύο κλάσεων δεδομένων που αντιστοιχεί στο ζεύγος $(\boldsymbol{w}, w_0)$, και στην οποία επιθυμούμε να καταλήξουμε μετά τη διαδικασία βελτιστοποίησης. Στο σχήμα παρουσιάζεται και το διάνυσμα $\boldsymbol{w}$ το οποίο είναι γνωστό πως είναι κάθετο στο υπερεπίπεδο διαχωρισμού καθώς και τα σημεία Α, Δ τα οποία αντιπροσωπεύουν στιγμιότυπα εκπαίδευσης με χαρακτηριστικά εισόδου $\boldsymbol{x}^{(i)}$ και $\boldsymbol{x}^{(j)}$ αντίστοιχα και για κάθε ένα ισχύει $y^{(i)} = 1$, $y^{(j)} = -1$.

Η απόσταση $\gamma^{(i)}$ του Α από τη ευθεία διαχωρισμού δίνεται από το μήκος του ευθύγραμμου τμήματος ΑΒ. Εφόσον το Α αναπαριστά το $\boldsymbol{x}^{(i)}$ το Β θα αναπαριστά το $\boldsymbol{x}^{(i)} - \gamma^{(i)} \frac{\boldsymbol{w}}{||\mathbf{w}||}$ όπου $\frac{\boldsymbol{w}}{||\mathbf{w}||}$ το μοναδιαίο διάνυσμα του $\boldsymbol{w}$, κάθετο στη ευθεία διαχωρισμού L.

Το σημείο Β ανήκει στη ευθεία διαχωρισμού συνεπώς ικανοποιεί τη εξίσωση :

$$\boldsymbol{w}^T\boldsymbol{x} + w_0 = 0$$

Επομένως θα ισχύει:

$$\boldsymbol{w}^T\left(\boldsymbol{x}^{(i)} - \gamma^{(i)}\frac{\boldsymbol{w}}{||\mathbf{w}||}\right) + w_0 = 0$$

Και επιλύοντας ως προς $\gamma^{(i)}$:

$$\gamma^{(i)} = \frac{\boldsymbol{w}^T\,\boldsymbol{x}^{(i)} + w_0}{||\mathbf{w}||}$$

Με την ίδια ακριβώς λογική καταλήγουμε πως:

$$\gamma^{(j)} = \frac{\boldsymbol{w}^T\,\boldsymbol{x}^{(j)} + w_0}{||\mathbf{w}||}$$

Συνεπώς, έστω δύο κλάσεις δεδομένων C_0, C_1 οι οποίες αντιστοιχούν στα δεδομένα για τα οποία ισχύει $y^{(i)} = -1\,, y^{(i)} = 1$ αντίστοιχα. Ορίζουμε ως γεωμετρικό περιθώριο ταξινόμησης μεταξύ των δύο κλάσεων το άθροισμα:

$$\gamma = \gamma_0 + \gamma_1$$

Όπου:

$$\gamma_0 = \min_{x\epsilon C_0}\frac{|\boldsymbol{w}^T\mathbf{x} + w_0|}{||\mathbf{w}||}$$

$$\gamma_1 = \min_{x\epsilon C_1}\frac{|\boldsymbol{w}^T\mathbf{x} + w_0|}{||\mathbf{w}||}$$

Όπως γίνεται αντιληπτό από τις παραπάνω εξισώσεις οι αριθμοί γ_0, γ_1 είναι θετικοί αριθμοί και αντιστοιχούν στην απόσταση το πιο κοντινού προτύπου της

κάθε κλάσης από τη διαχωριστική επιφάνεια (στη περίπτωση μας ευθεία). Όσο τα δύο αυτά περιθώρια μικραίνουν τόσο πιο απαισιόδοξες γίνονται οι προβλέψεις του μοντέλου αφού για τα στιγμιότυπα που βρίσκονται κοντά στη διαχωριστική επιφάνεια, μια μικρή μετατόπιση λόγω θορύβου μπορεί να αλλάξει την απόφαση ταξινόμησης τους.

Τα πρότυπα εκπαίδευσης για τα οποία επιτυγχάνεται η ελάχιστη απόσταση γ_0 ή γ_1 ,ανάλογα τη κλάση στη οποία ανήκουν, καλούνται
Διανύσματα Υποστήριξης.

<u>Κανονικό Διαχωριστικό Επίπεδο</u>

Η απαλλαγή του πλεονασμού που εισάγει η εξίσωση $\boldsymbol{w}^T\boldsymbol{x} + w_0 = 0$ θα πραγματοποιηθει με την εισαγωγή περιορισμών οι οποίοι θα ορίσουν την έννοια του κανονικού διαχωριστικού επιπέδου.

Συνεπώς κανονικό διαχωριστικό επίπεδο ορίζεται ως εκείνο που:

- Τοποθετείται ακριβώς στη μέση ανάμεσα στις δύο κλάσεις C_0, C_1 , οπότε $\gamma_0 = \gamma_1$.
- Τα $(\boldsymbol{w}, w_0)$ είναι τέτοια ώστε να ισχύει:

$$\boldsymbol{w}^T\boldsymbol{x} + w_0 \begin{cases} \leq -1 \ \alpha\nu \ x\epsilon C_0 \\ > 1 \ \ \alpha\nu \ x\epsilon C_1 \end{cases}$$

Από τους παραπάνω περιορισμούς έχουμε πως $\min_{x\epsilon C_0}|\boldsymbol{w}^T\boldsymbol{x} + w_0| = \min_{x\epsilon C_1}|\boldsymbol{w}^T\boldsymbol{x} + w_0| = 1$ και $\gamma_0 = \gamma_1 = 1/||\mathbf{w}||$.

Συνεπώς το γεωμετρικό περιθώριο ταξινόμησης γράφεται ως:

$$\gamma = \frac{2}{||\mathbf{w}||}$$

Η μεγιστοποίηση του γεωμετρικού περιθωρίου ανάγεται στη ελαχιστοποίηση της νόρμας $||\mathbf{w}||$ ή $||\mathbf{w}||^2$, έτσι η αναζήτηση της καταλληλότερη λύσης μετατρέπεται στο παρακάτω πρόβλημα βελτιστοποίησης :

$$\min_{\gamma,w,w_0} \frac{1}{2}||\mathbf{w}||^2$$

$$\text{s.t. } y^{(i)}\left(\boldsymbol{w}^T\boldsymbol{x}^{(i)} + w_0\right) \geq 1 \ i = 1, \dots, m$$

Όπου m το πλήθος των στιγμιότυπων εκπαίδευσης.

Το παραπάνω πρόβλημα ανήκει στη περιοχή του τετραγωνικού προγραμματισμού για το οποία υπάρχει πληθώρα αλγορίθμων και πακέτων λογισμικού που προσφέρουν λύσεις από το πεδίο της αριθμητικής ανάλυσης.

Συνεπώς έχοντας προσδιορίσει τα $(\boldsymbol{w}, w_0)$,το υπερεπίπεδο που ορίζεται από την εξίσωση $\boldsymbol{w}^T\boldsymbol{x} + w_0 = 0$ θα χρησιμοποιηθεί για τη πραγματοποίηση των προβλέψεων του μοντέλου.

Πιο συγκεκριμένα , δοθέντος ενός σημείου Τ με χαρακτηριστικά εισόδου $\boldsymbol{x}^{(i)}$ σε ένα πρόβλημα ταξινόμησης δύο κλάσεων C_0, C_1 έχουμε:

$$Αν\ \ \boldsymbol{w}^T\boldsymbol{x}^{(i)} + w_0 > 0\ τότε\ T \epsilon C_1$$
$$Αν\ \ \boldsymbol{w}^T\boldsymbol{x}^{(i)} + w_0 < 0\ τότε\ T \epsilon C_0$$

2.5.2 Συναρτήσεις Πυρήνα και μη Γραμμική Κατηγοριοποίηση

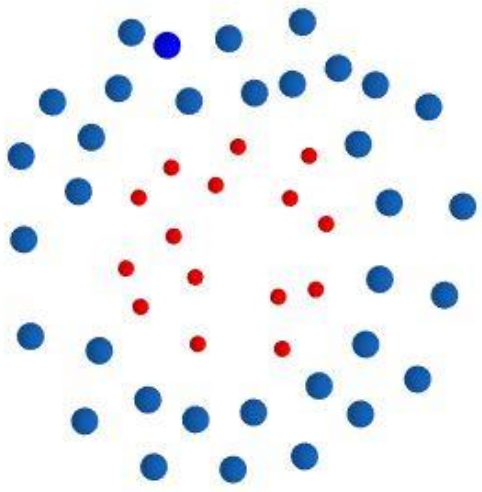
Οι τεχνικές που παρουσιάστηκαν παραπάνω μπορούν να εφαρμοστούν μόνο στη επίλυση γραμμικά διαχωρίσιμων προβλημάτων. Ωστόσο τα περισσότερα προβλήματα που συναντάμε στο πραγματικό κόσμο είναι μη γραμμικώς διαχωρίσιμα και συνεπώς η χρήση ενός κανονικού υπερεπιπέδου για τη ταξινόμηση των δεδομένων δε μπορεί να εφαρμοστεί.

Για την επίλυση του συγκεκριμένου προβλήματος χρησιμοποιείται η λογική και η επίγνωση που προκύπτει από το θεώρημα του ***Cover.***

Θεώρημα Cover

Το θεώρημα Cover αποτελεί μια σημαντική διατύπωση στη θεωρία της υπολογιστικής μάθησης και είναι ένα από τα κύρια θεωρητικά κίνητρα για τη χρήση μη γραμμικών μεθόδων πυρήνα στις εφαρμογές της μηχανικής μάθησης, Το θεώρημα δηλώνει ότι δοθέντος ενός συνόλου δεδομένων εκπαίδευσης ,σε ένα πρόβλημα ταξινόμησης ,τα οποία δεν είναι γραμμικά διαχωρίσιμα ,τότε μπορεί κανείς με μεγάλη πιθανότητα να μετασχηματίσει αυτό το σύνολο σε ένα αντίστοιχο, το οποίο όμως θα είναι γραμμικά διαχωρίσιμο ,προβάλλοντάς το σε ένα χώρο υψηλότερων διαστάσεων χρησιμοποιώντας κάποιον μη γραμμικό μετασχηματισμό.

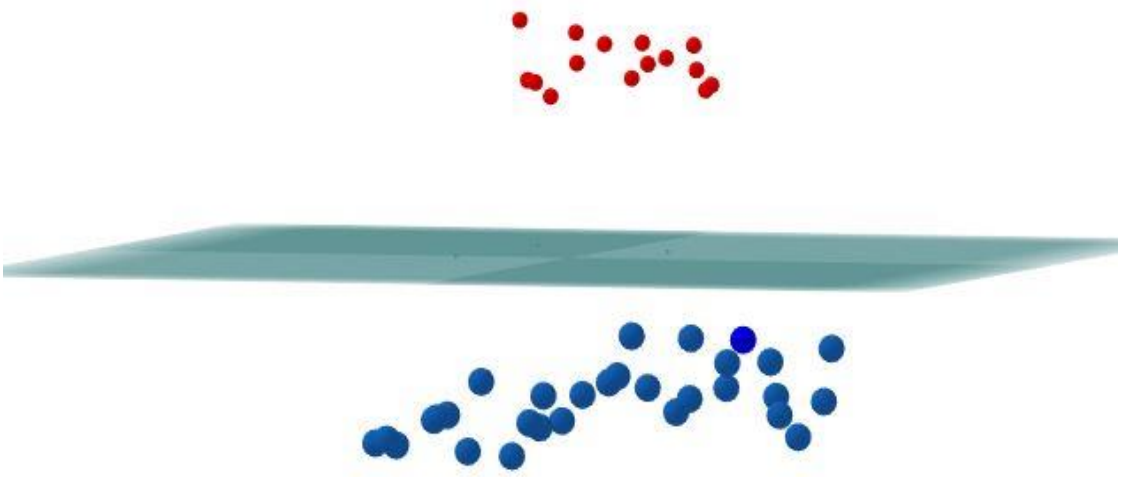
Συνεπώς χρησιμοποιώντας ένα κατάλληλο μη-γραμμικό μετασχηματισμό έστω Φ(.) στα πρότυπα εκπαίδευσης οδηγούμαστε σε ένα χώρο που το σύνολο δεδομένων θα είναι γραμμικά διαχωρίσιμο και συνεπώς θα μπορεί να χρησιμοποιηθεί η μέθοδος που παρουσιάστηκε παραπάνω. Οπότε η βέλτιστη διαχωριστική επιφάνεια είναι πλέον της μορφής $\boldsymbol{w}^T\boldsymbol{\Phi}(\boldsymbol{x}) + w_0 = 0$.



Εικόνα 2-6 Μη διαχωρίσιμο σύνολο δεδομένων

Παραπάνω παρουσιάζονται δύο μη γραμμικά διαχωρίσιμες κλάσεις δεδομένων. Όπως γίνεται εύκολα αντιληπτό δεν υπάρχει υπερεπίπεδο ,στις δυο διαστάσεις ευθεία , που να μπορεί να διαχωρίσει τις συγκεκριμένες κλάσεις.

Ωστόσο σύμφωνα και με το Θεώρημα του Cover παρατηρούμε στο παρακάτω σχήμα πώς το σύνολο δεδομένων μετατρέπεται σε γραμμικά διαχωρίσιμο, ύστερα από κατάλληλο μετασχηματισμό σε χώρο τριών διαστάσεων.



Εικόνα 2-7 Διαχωρίσιμο σύνολο δεδομένων ύστερα από την εφαρμογή του θεωρήματος Cover

Οικονομία πράξεων με χρήση συναρτήσεων πυρήνα

Κατά τη διαδικασία βελτιστοποίησης του προβλήματος της εύρεσης του βέλτιστου υπερεπιπέδου διαχωρισμού ,κάνοντας χρήση των πολλαπλασιαστών Langranze ,εμφανίζονται εσωτερικά γινόμενα της μορφής $\boldsymbol{x}^{(i)^T}$, $\boldsymbol{x}^{(j)}$ και κατά συνέπεια κατά τη διαδικασία του μη-γραμμικού μετασχηματισμού θα εμφανίζονται εσωτερικά γινόμενα της μορφής $\boldsymbol{\Phi}(\boldsymbol{x}^{(i)})^T, \boldsymbol{\Phi}(\boldsymbol{x}^{(j)})$. Έχοντας υπόψιν λοιπόν αυτή τη παρατήρηση ορίζουμε τη συνάρτηση :

$$K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right) = \boldsymbol{\Phi}(\boldsymbol{x}^{(i)})^T \boldsymbol{\Phi}(\boldsymbol{x}^{(j)})$$

Η συνάρτηση K ονομάζεται συνάρτηση πυρήνα. Όπως γίνεται φανερό από τη παραπάνω σχέση προκειμένου να υπολογίσουμε τα γινόμενα $\boldsymbol{\Phi}(\boldsymbol{x}^{(i)})^T \boldsymbol{\Phi}(\boldsymbol{x}^{(j)})$ τα οποία εμφανίζονται στη επίλυση του προβλήματος βελτιστοποίησης δεν χρειάζεται να υπολογίσουμε όλους τους μετασχηματισμούς $\boldsymbol{\Phi}(\boldsymbol{x}^{(j)})$ για κάθε $\boldsymbol{x}^{(j)}$ του συνόλου εκπαίδευσης αλλά αρκεί να χρησιμοποιήσουμε τη συνάρτηση πυρήνα, εξοικονομώντας με αυτό τον τρόπο υπολογιστικούς πόρους.

Σύμφωνα με το θεώρημα του **Mercer** κάθε συνεχής συμμετρική και μη αρνητική συνάρτηση πυρήνα $\mathrm{K}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$ μπορεί να γραφτεί σαν το εσωτερικό γινόμενο $\boldsymbol{\Phi}(\boldsymbol{x}^{(i)})^T \boldsymbol{\Phi}(\boldsymbol{x}^{(j)})$ δύο διανυσμάτων απείρων διαστάσεων.

Μερικές από τις πιο συχνά χρησιμοποιούμενες συναρτήσεις πυρήνα είναι:

- $e^{-\|x-y\|^2/2\sigma^2}$ Γκαουσσιανή RBF

- $[\boldsymbol{x}^T\boldsymbol{y}+\boldsymbol{\theta}]^P$ Πολυωνυμική

- $\tanh(a\,\boldsymbol{x}^T\boldsymbol{y}+\boldsymbol{\theta})$ Σιγμοειδής

Χαρακτηριστικό παράδειγμα αποτελεί η Γκααουσιανή συνάρτηση πυρήνα για την οποία η συνάρτηση Φ() που την παράγει είναι άγνωστη και μάλιστα αρκετά δύσκολο να υπολογιστεί. Η συνάρτηση Φ() στη προκείμενη περίπτωση απεικονίζει ένα διάνυσμα **x** σε ένα διάνυσμα άπειρων διαστάσεων καθώς όμως στο πρόβλημα βελτιστοποίησης των Μηχανών Διανυσμάτων Υποστήριξης χρησιμοποιείται παντού η συνάρτηση πυρήνα δεν χρειάζεται ο υπολογισμός της περίπλοκης συνάρτησης Φ().

2.5.3 Ανάλυση Παλινδρόμησης και Μηχανές Διανυσμάτων Υποστήριξης

Η παραπάνω ανάλυση παρουσιάζει και εξηγεί τον τρόπο με τον οποίο η Μηχανές Διανυσμάτων Υποστήριξης χρησιμοποιούνται για την επίλυση προβλημάτων ταξινόμησης. Ο συγκεκριμένος αλγόριθμος όμως είναι ευρέως χρησιμοποιούμενος και σε προβλήματα Ανάλυσης Παλινδρόμησης με εξαιρετικές επιδόσεις.

Με τον ίδιο τρόπο, όπως και με τη προσέγγιση ταξινόμησης υπάρχει επίσης κίνητρο να αναζητηθούν και να βελτιστοποιηθούν τα όρια γενίκευσης που δίνονται και για την παλινδρόμηση.

Πιο συγκεκριμένα, δοθέντος ενός συνόλου εκπαίδευσης $(\boldsymbol{x}^{(1)}, y^{(1)},\ldots, \boldsymbol{x}^{(n)}, y^{(n)})$ ο στόχος είναι η εύρεση μιας συνάρτησης f(**x**) η οποία θα έχει το πολύ ε απόκλιση από όλους τους στόχους $y^{(i)}$ του συνόλου εκπαίδευσης και θα είναι όσο πιο επίπεδη (flat function) γίνεται. Δηλαδή το μοντέλο δεν επηρεάζεται από σφάλματα πρόβλεψης μικρότερα του ε αλλά δεν επιτρέπει και την ύπαρξη σφαλμάτων μεγαλύτερα του ε.

Χάριν απλότητας θα υποθέσουμε αρχικά την αναζήτηση μιας γραμμικής συνάρτησης.

Έστω λοιπόν η γραμμική συνάρτηση:

$$f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0 \quad \text{με } w_0 \in \mathrm{R}$$

Για να ικανοποιείται ο περιορισμός πως η f πρέπει να είναι όσο το δυνατόν πιο επίπεδη χρειάζεται να έχουμε μικρά βάρη $\boldsymbol{w}$ δηλαδή χρειάζεται να ελαχιστοποιήσουμε τη νόρμα $\left|\left|\mathbf{w}\right|\right|^2$.

Συνεπώς εκφράζοντας και τους περιορισμούς που αφορούν την απόκλιση της συνάρτησης από τους στόχους εκπαίδευσης $y^{(i)}$ μαθηματικά καταλήγουμε στο παρακάτω πρόβλημα βελτιστοποίησης:

$$\min_{w,w_0} \frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2$$

$$y^{(i)} - \boldsymbol{w}^T\boldsymbol{x}^{(i)} - w_0 \leq \varepsilon \;\; i = 1, \dots, n$$
$$\boldsymbol{w}^T\boldsymbol{x}^{(i)} + w_0 - y^{(i)} \leq \varepsilon \;\; i = 1, \dots, n$$

Ωστόσο, η παραπάνω διατύπωση του προβλήματος κάνει την παραδοχή, πως θα υπάρχει η συνάρτηση f η οποία θα προσεγγίζει όλα τα ζεύγη $(\boldsymbol{x}^{(i)}, y^{(i)})$ με ακρίβεια ε. Δηλαδή υποθέσαμε ότι το παραπάνω κυρτό πρόβλημα βελτιστοποίησης θα έχει λύση. Αυτό όμως δεν είναι πάντα εφικτό και συνεπώς θα πρέπει να χαλαρώσουμε τους περιορισμούς επιτρέποντας μεγαλύτερα σφάλματα.

Για το λόγο αυτό εισάγονται οι μεταβλητές χαλάρωσης (slack variables) $\xi^{(i)}, \xi^{(i)^*}$ ώστε να αντιμετωπιστούν λιγότερο αυστηρά οι ανέφικτοι περιορισμού του προβλήματος. Συνεπώς καταλήγουμε στο πρόβλημα:

$$\min_{w,w_0,\xi} \left[\frac{1}{2}\left|\left|\mathbf{w}\right|\right|^2 + C\sum_{i=1}^{n}(\xi^{(i)} + \xi^{(i)^*})\right]$$

$$y^{(i)} - \boldsymbol{w}^T\boldsymbol{x}^{(i)} - w_0 \leq \varepsilon + \xi^{(i)} \; i = 1, \dots, n$$
$$\boldsymbol{w}^T\boldsymbol{x}^{(i)} + w_0 - y^{(i)} \leq \varepsilon + \xi^{(i)^*} \;\; i = 1, \dots, n$$
$$\xi^{(i)}, \xi^{(i)^*} \geq 0$$

Η σταθερά C καθορίζει την αντιστάθμιση μεταξύ του πόσο επίπεδη θα είναι η συνάρτηση f και του συνολικού μεγέθους (αθροιστικά) των αποκλίσεων που θα είναι μεγαλύτερες του ε και είναι μια σταθερά που καθορίζεται συνήθως από τον χρήστη -δημιουργό του μοντέλου. Δηλαδή η σταθερά ε καθορίζει τον βαθμό στον οποίο είναι επιτρεπτό το μοντέλο να αποκλίνει από τους στόχους $y^{(i)}$ περισσότερο από ε.

Το παραπάνω είναι ένα καλά ορισμένο πρόβλημα βελτιστοποίησης, όπως και στη περίπτωση της ταξινόμησης, δεν θα προχωρήσουμε σε περισσότερες λεπτομέρειες που αφορούν την αναλυτική λύση του.

2.6 Τεχνητά Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα ΤΝΔ, είναι υπολογιστικά μοντέλα μάθησης τα οποία είναι εμπνευσμένα από τα βιολογικά νευρωνικά δίκτυα που συνιστούν τους ζωικούς εγκεφάλους. Η βασική διαφορά του συγκεκριμένου μοντέλου με άλλα, είναι η μη ρητή αναπαράσταση γνώσης καθώς και η έλλειψη ενός ειδικά σχεδιασμένου αλγόριθμου αναζήτησης προτύπων στα δεδομένα. Αντιθέτως τα ΤΝΔ βασίζονται σε βιολογικά πρότυπα χρησιμοποιώντας δομές και διαδικασίες που προσπαθούν να μιμηθούν αυτές του ζωικού εγκεφάλου.

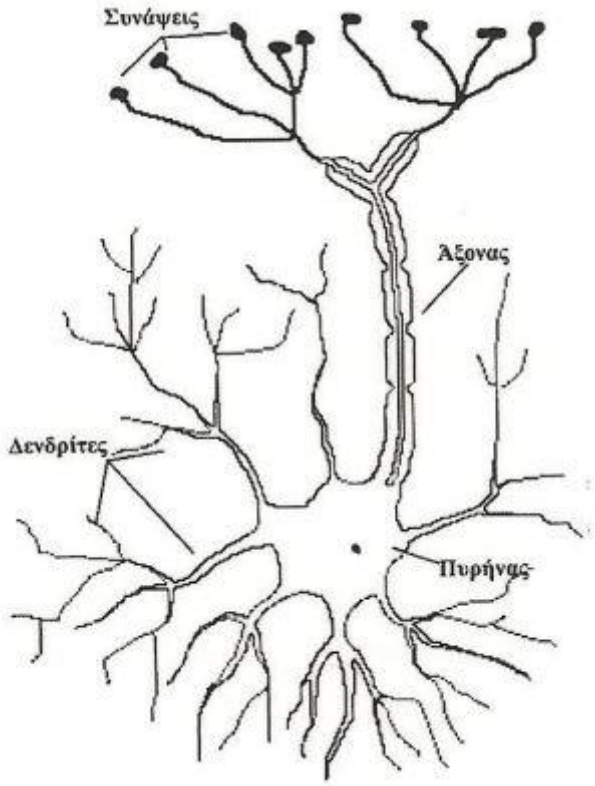
2.6.1 Βιολογικός και Τεχνητός Νευρώνας

Για να κατανοήσουμε καλύτερα το μαθηματικό μοντέλο του τεχνητού νευρώνα είναι χρήσιμο να περιγράψουμε απλοϊκά την λειτουργία ενός βιολογικού νευρώνα πάνω στην οποία έχει στηριχθεί το συγκεκριμένο μοντέλο.

2.6.1.1 Βιολογικός Νευρώνας

Η δομική μονάδα του νευρικού συστήματος είναι ο νευρώνας. Ο εγκέφαλος ενός ενήλικου ανθρώπου αποτελείται από δισεκατομμύρια νευρώνες κάθε ένας από τους οποίους συνδέεται με ένα μεγάλο πλήθος γειτονικών ή μη, νευρώνων ,δημιουργώντας έτσι ένα πολύπλοκο βιολογικό δίκτυο, τα δομικά στοιχεία του οποίου αλληλοεπιδρούν μεταξύ τους με χρόνους απόκρισης της τάξης των msec.

Σημαντικό στάδιο στη κατανόηση του μαθηματικού - υπολογιστικού μοντέλου των τεχνητών νευρωνικών δικτύων είναι η κατανόηση των 3 βασικών λειτουργικών τμημάτων ενός βιολογικού νευρώνα.



Εικόνα 2-8 Βιολογικός Νευρώνας
(Πηγή Διαμανταράς 2007).

Οι βιολογικοί νευρώνες αποτελούνται από:

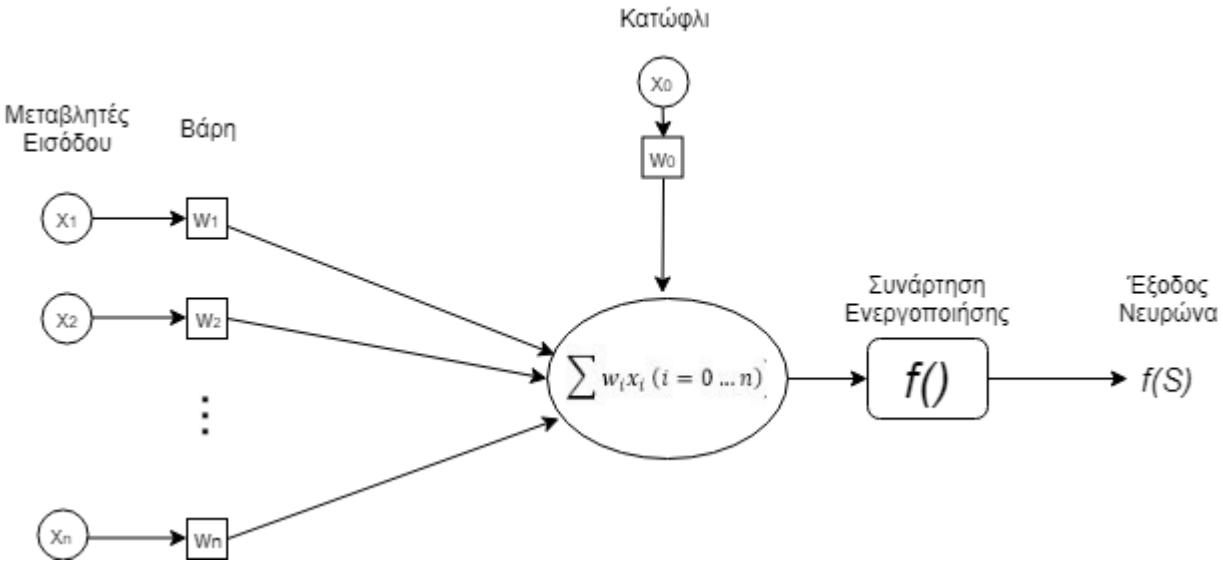
- Τους ***δενδρίτες*** ,οι οποίοι αποτελούν το δίκτυο με το οποίο ο νευρώνας προσλαμβάνει σήματα από άλλα κύτταρα.

- Τον ***νευράξονα,*** ο οποίος είναι μια λεπτή ίνα συνήθως μεγάλου μήκους, υπεύθυνος για την μεταφορά σημάτων προς άλλους νευρώνες υπό τη μορφή ηλεκτρικών παλμών.

- Τις ***συνάψεις*** , οι οποίες είναι τα μέσα με τα οποία πραγματοποιείται η μετάδοση των σημάτων μεταξύ των νευρώνων. Αποτελούν τα σημεία ένωσης μεταξύ του άξονα ενός νευρώνα και των δενριτών του άλλου. Οι συνάψεις αποτελούν σημαντικό συστατικό κάθε βιολογικού νευρώνα καθώς το πλάτος τους, και η απόσταση τους από τον δενδρίτη επηρεάζουν την ευκολία με την οποία οι ηλεκτρικοί παλμοί μεταδίδονται στον γειτονικό νευρώνα. Το ποσοστό αυτής της ηλεκτρικής δραστηριότητας που μεταδίδεται στον δενρτίτη καλείται ***συναπτικό βάρος*** και αποτελεί βασική έμπνευση για τη δημιουργία του μαθηματικού μοντέλου των τεχνητών νευρώνων.

2.6.1.2 Μοντέλο Τεχνητού Νευρώνα

Ο τεχνητός νευρώνας είναι όπως έχουμε ήδη αναφέρει ένα υπολογιστικό μοντέλο βασισμένο στη λειτουργά και δομή των βιολογικών νευρώνων. Όπως γίνεται αντιληπτό και από το σχήμα, στο τεχνητό νευρώνα τα σήματα εισόδου είναι συνεχής μεταβλητές ,αντί για ηλεκτρικοί παλμοί, και το ρόλο των συνάψεων παίζουν οι μεταβλητές w_i οι οποίες ονομάζονται βάρη και καθορίζουν τη συμβολή των εισόδων στην έξοδο του νευρώνα. Σε αρκετές περιπτώσεις υπάρχει ένα συγκεκριμένο βάρος w_0 το οποίο ονομάζεται πόλωση ή κατώφλι και είναι καθοριστικό για τη τιμή ενεργοποίησης του νευρώνα.



Εικόνα 2-9 Μοντέλο Τεχνητού Νευρώνα

Τα σημαντικότερα δομικά στοιχεία του τεχνητού νευρώνα είναι ο αθροιστής, ο οποίος παράγει το άθροισμα $S = \sum w_i x_i \ (i = 0 \dots n)$,και η συνάρτηση ενεργοποίησης $f(S)$ η οποία διαμορφώνει την τελική έξοδο του νευρώνα, πάντα συναρτήσει της τιμή της ποσότητας S και του κατωφλιού w_0.

Το βάρος w_0 επιδρά πάντα στη είσοδο $x_0 = 0$ και είναι ένα εξωτερικό ερέθισμα το οποίο χρησιμοποιείται συνήθως για να καθορίσει τη θέση της συνάρτησης ενεργοποίησης στο καρτεσιανό επίπεδο όπως γίνεται φανερό από την παρακάτω εξίσωση:

$$f(S) = f\left(\sum w_i x_i \ (i = 0 \dots n)\right) = f\left((\sum w_i x_i \ (i = 1 \dots n)) + w_0\right)$$

Η συνάρτηση ενεργοποίησης παίζει καθοριστικό ρόλο στη λειτουργία του

τεχνητού νευρώνα καθώς διαμορφώνει τη τελική τιμή του σήματος εξόδου του νευρώνα συναρτήσει των σημάτων εισόδου και των βαρών.

Παρακάτω παρουσιάζονται μερικές τυπικές και συχνά χρησιμοποιούμενες συναρτήσεις ενεργοποίησης:

- Η <u>*συνάρτηση πρόσημου*</u>

  - $f(S) = +1\ \alpha\nu\ S > 0$
  - $f(S) = -1\ \ \alpha\nu\ S < 0$
  - $f(S) = 0{,}5\ \ \alpha\nu\ S = 0$

- Η <u>*βηματική συνάρτηση,*</u> η οποία δίνει μη μηδενική έξοδο (συνήθως 1) μόνο αν S>0.

  - $f(S) = \ 1\ \alpha\nu\ S > 0$
  - $f(S) = 0\ \alpha\nu\ S \leq 0$

- Η <u>*λογιστική συνάρτηση*</u> για την οποία γράφουμε $f(S) = \frac{1}{1+e^{-aS}}$.
  Η συγκεκριμένη συνάρτηση ανήκει στην οικογένεια των σιγμοειδών συναρτήσεων οι οποίες είναι μαθηματικές συναρτήσεις ο οποίες σχηματίζουν μια καμπύλη σε σχήμα S, και οι οποίες είναι αρκετά δημοφιλείς ως συναρτήσεις ενεργοποίησης, αλλά και ως συστατικά στοιχεία στατιστικών κατανομών.

- Η <u>*γραμμική συνάρτηση*</u> της οποίας η έξοδος είναι γραμμικός συνδυασμός της εισόδου. Η συγκεκριμένη συνάρτηση ,σε αντίθεση με τις άλλες ,χρησιμοποιείται κυρίως σε προβλήματα Ανάλυσης Παλινδρόμησης.

2.6.2 Αρχιτεκτονικές Τεχνητών Νευρωνικών Δικτύων.

Τα τεχνητά νευρωνικά δίκτυα είναι υπολογιστικές δομές Μηχανικής Μάθησης που αποτελούνται από ένα πλήθος τεχνητών νευρώνων οργανωμένων σε συνθέσεις παρόμοιες με αυτές του ανθρώπινου εγκεφάλου.

Συνήθως τα ΤΝΔ είναι οργανωμένα σε επίπεδα τεχνητών νευρώνων που επιτελούν διαφορετικές λειτουργίες ανάλογα με τη συνάρτηση ενεργοποίησης που χρησιμοποιείται σε κάθε ένα από αυτά.

Πιο συγκεκριμένα τα ΤΝΔ αποτελούνται από:

- ***Το επίπεδο εισόδου***, το οποίο είναι υπεύθυνο για την λήψη των σημάτων

εισόδου χωρίς να παίζει ρόλο στη εκπαίδευση του δικτύου αφού τα δομικά στοιχεία που ανήκουν σε αυτό δεν επιτελούν κάποιο υπολογισμό.

- ***Το επίπεδο εξόδου***, αποτελείται από νευρώνες οι οποίοι εκτός της συνεισφοράς τους στην εκπαίδευση του δικτύου παρέχουν και τα σήματα εξόδου στη κατάλληλη μορφή (π.χ διακριτά ,συνεχή σήματα).

- ***Τα κρυφά ή ενδιάμεσα επίπεδα*** ,είναι προαιρετικές δομές τεχνητών νευρώνων τα οποία διαδραματίζουν σημαντικό ρόλο στην εκπαίδευση του νευρωνικού δικτύου και στην ικανότητα του να 'μαθαίνει' από τα δεδομένα.

Υπάρχουν δύο κύριες κατηγορίες ΤΝΔ , τα **δίκτυα πρόσθιας τροφοδότησης** και τα **δίκτυα με ανατροφοδότηση ή αναδρομικά δίκτυα.**

Δίκτυα πρόσθιας τροφοδότησης

Σε ένα δίκτυο πρόσθιας τροφοδότησης το σήμα εισόδου ρέει από το επίπεδο εισόδου προς το επίπεδο εξόδου. Δηλαδή δεν υπάρχουν συνδέσεις νευρώνων ενός επιπέδου με νευρώνες του ίδιου ή προηγούμενου επιπέδου. Συνεπώς σε ένα δίκτυο πρόσθιας τροφοδότησης η έξοδος αποτελεί συνάρτηση του τρέχοντος σήματος εισόδου.

Αναδρομικά δίκτυα

Στα αναδρομικά δίκτυα υπάρχει τουλάχιστον ένας βρόγχος ανάδρασης μεταξύ των τεχνητών νευρώνων. Συνεπώς η έξοδος των συγκεκριμένων δικτύων δεν εξαρτάται μόνο από τον τρέχον σήμα εισόδου αλλά και από προηγούμενες, χρονικά, εισόδους. Άρα τα αναδρομικά δίκτυα μπορούν να υποστηρίξουν βραχυπρόθεσμη μνήμη η οποία είναι ιδιαίτερα χρήσιμη σε προβλήματα όπως η αναγνώριση-ανάλυση κειμένου.

2.6.3 Το Δίκτυο Perceptron

Το δίκτυο Perceptron αποτελεί μια απλή τοπολογία δικτύου που ανήκει στη οικογένεια των δικτύων πρόσθιας τροφοδότησης. Στην πιο απλή του μορφή το μοντέλο αποτελείται από μόνο ένα νευρώνα και χρησιμοποιείται ως γραμμικός ταξινομητής.

Έστω $x_1, x_2, \dots, x_n$ οι είσοδοι του νευρώνα και - θ το κατώφλι όπως φαίνεται και στο σχήμα. Το Perceptron παράγει το άθροισμα $u = \sum_1^n w_i x_i - \theta$ όπου w_i τα συναπτικά βάρη του νευρώνα.

Η συνάρτηση ενεργοποίησης του συγκεκριμένου νευρώνα είναι η βηματική

συνάρτηση:

$$f(u) = \begin{cases} 1 \; \alpha\nu \; u > 0 \\ 0 \; \alpha\nu \; u \leq 0 \end{cases}$$

Συνεπώς σε ένα δυαδικό πρόβλημα ταξινόμησης η έξοδος $y = 1$ θα αντιστοιχεί σε στιγμιότυπα της μια κλάσης έστω C_1 και η έξοδος $y = 0$ θα αντιστοιχεί σε στιγμιότυπα της άλλης κλάσης έστω C_0.

Από τη παραπάνω ανάλυση καταλήγουμε πως ένα Perceptron n εισόδων αναπαριστά ένα υπερεπίπεδο $n-1$ διαστάσεων που διαχωρίζει τον $n-$διαάστατο χώρο σε δύο περιοχές που η μια αντιστοιχεί σε στιγμιότυπα για τα οποία ισχύει $y = 1$ και η άλλη σε στιγμιότυπα για τα οποία ισχύει $y = 0$. Η εύρεση του παραπάνω επιπέδου είναι δυνατή μόνο αν το πρόβλημα ταξινόμησης είναι γραμμικά διαχωρίσιμο.

Συνεπώς η λύση του προβλήματος ταξινόμησης ανάγεται στην εύρεση των βαρών w_i και του κατωφλίου θ.

2.6.3.1 Εκπαίδευση του Δικτύου Perceptron

Ο αλγόριθμος μάθησης των Perceptrons ανήκει στην ευρύτερη οικογένεια των αλγόριθμων μάθησης με επίβλεψη. Ο συγκεκριμένος αλγόριθμος είναι καθοδηγούμενος από το σφάλμα και καταλήγει σε τιμές βαρών w_i τέτοιες ώστε το υπερεπίπεδο που ορίζεται από τη εξίσωση $\sum_1^n w_i x_i - \theta = 0$ να διαχωρίζει πλήρως όλα τα δεδομένα εκπαίδευσης ανάλογα με τη κλάση στη οποία ανήκουν.

Πιο συγκεκριμένα ο αλγόριθμος είναι επαναληπτικός και σε κάθε επανάληψη τροποποιεί το επαυξημένο διάνυσμα βαρών $\boldsymbol{w}$ κάθε φορά που υπάρχει σφάλμα στη ταξινόμηση. Τα επαυξημένα διανύσματα $\boldsymbol{w}, \boldsymbol{x}$ είναι τα διανύσματα που προκύπτουν με την απορρόφηση του κατωφλίου θ ως βάρος στη είσοδο $x_0 = -1$ όπως έχουμε αναφέρει σε προηγούμενο κεφάλαιο.

Έστω ότι ο αλγόριθμος μάθησης βρίσκεται στη k επανάληψη στην οποία εισάγεται το πρότυπο p με τιμή πρόβλεψης $y = f(\boldsymbol{w}_{k-1}^T \boldsymbol{x}^{(p)})$ και τιμή στόχο $d^{(p)}$. Τότε το διάνυσμα των επαυξημένων βαρών μεταβάλλεται κατά $\Delta \boldsymbol{w} = \beta(d^{(p)} - y)\boldsymbol{x}^{(p)}$ δηλαδή ισχύει:

$$\boldsymbol{w}_k = \boldsymbol{w}_{k-1} + \beta(d^{(p)} - y)\boldsymbol{x}^{(p)}$$

Όπου β η παράμετρος που ρυθμίζει τη ταχύτητα εκπαίδευσης ή καλύτερα το μέγεθος της διόρθωσης και καλείται **βήμα ή ρυθμός εκπαίδευσης.**

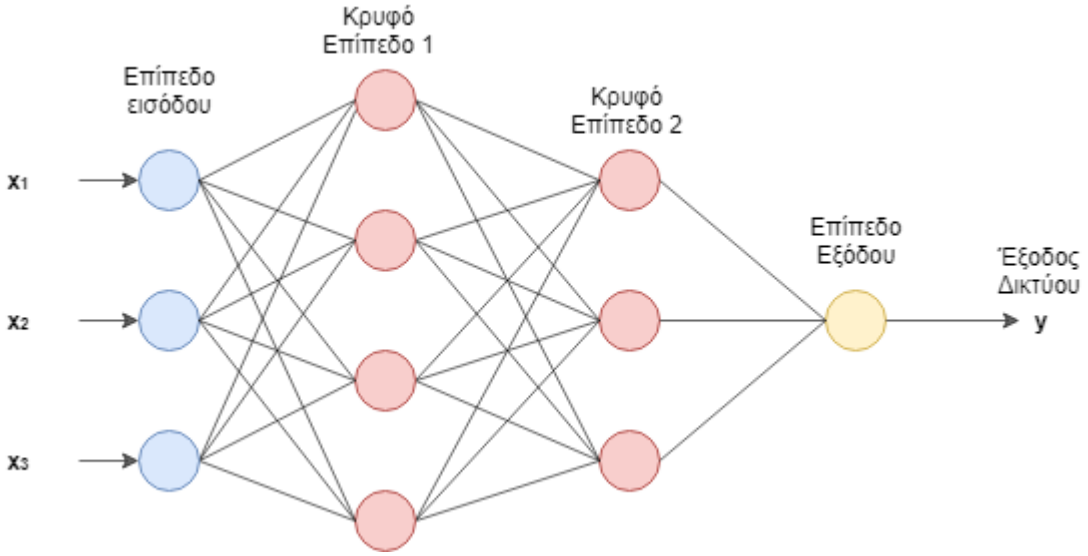
Ένας πλήρης κύκλος χρήσης των προτύπων στη διαδικασία εκπαίδευσης καλείται **εποχή**. Όταν λοιπόν το διάνυσμα των συναπτικών βαρών παραμείνει αμετάβλητο για μια ολόκληρη εποχή τότε ο αλγόριθμος έχει συγκλίνει και η αναλυτική εξίσωση του υπερεπιπέδου διαχωρισμού των δεδομένων είναι γνωστή.

Αποδεικνύεται πως αν το πρόβλημα ταξινόμησης είναι γραμμικά διαχωρίσιμο, τότε η παραπάνω διαδικασία εκπαίδευσης συγκλίνει σε πεπερασμένο αριθμό εποχών.

2.6.4 Το Μοντέλο Perceptron Πολλών Στρωμάτων – MLP

Τα πολυεπίπεδα Perceptron αποτελούν μια γενίκευση των απλών Perceptron που αναλύσαμε στο προηγούμενο κεφάλαιο. Πρόκειται για δίκτυα πρόσθιας τροφοδότησης τα οποία περιέχουν τουλάχιστον ένα κρυφό επίπεδο εκτός από τα επίπεδα εισόδου και εξόδου.

Όπως διακρίνεται και από το παρακάτω σχήμα, οι νευρώνες ενός επιπέδου l τροφοδοτούν αποκλειστικά τους νευρώνες του επιπέδου $l+1$ αλλά και κάθε νευρώνας σε ένα επίπεδο είναι συνδεδεμένος με κάθε νευρώνα του επόμενου επιπέδου για αυτό το λόγο ονομάζονται και **πλήρως διασυνδεδεμένα δίκτυα.**



Εικόνα 2-10 MLP τριών στρωμάτων

Όπως διαπιστώθηκε και στη προηγούμενη ενότητα τα δίκτυα Perceptron μπορούν να αναπαραστήσουν μόνο επίπεδες επιφάνειες περιορισμός που αίρεται με τη χρήση περισσότερων επιπέδων.

Η δημιουργία των MLP είχε ως σκοπό την επίλυση πολύπλοκων και μη γραμμικών προβλημάτων τα οποία όπως είδαμε παραπάνω δεν μπορούσαν να λύσουν τα απλά δίκτυα ενός επιπέδου. Για αυτό το λόγο αλλά και επειδή οι περισσότεροι αλγόριθμοι εκπαίδευσης, όπως η **κατάβαση δυναμικού** που θα δούμε παρακάτω , χρησιμοποιούν παραγώγους, οι συναρτήσεις ενεργοποίησης των νευρώνων πρέπει να είναι παραγωγίσιμες στο πεδίο ορισμού τους αλλά και μη γραμμικές.

Τέτοιες συναρτήσεις είναι που ανήκουν στην οικογένεια των σιγμοειδών συναρτήσεων όπως :

- Η λογιστική συνάρτηση $f(S) = \frac{1}{1+e^{-aS}}$
- Αλλά και η υπερβολική εφαπτομένη $\tanh(S) = \frac{e^{S}-e^{-S}}{e^{S}+e^{-S}}$

Οι παραπάνω συναρτήσεις έχουν την ιδιότητα να μοιάζουν αρκετά στη βηματική συνάρτηση που χρησιμοποιούν τα Perceptrons αλλά λόγω της μορφής τους μπορούν δημιουργήσουν ομαλές επιφάνειες χωρίς απότομες μεταβολές, ιδιότητα χρήσιμη για τη προσέγγιση μη γραμμικών συναρτήσεων.

Τα Πολυεπίπεδα Δίκτυα Perceptron ως Καθολικοί Προσεγγιστές

Σύμφωνα με το θεώρημα καθολικής προσέγγισης ένα Νευρωνικό Δίκτυο πρόσθιας τροφοδότησης με τουλάχιστον ένα κρυφό επίπεδο το οποίο περιέχει πεπερασμένο πλήθος νευρώνων με μη γραμμική σιγμοειδή συνάρτηση ενεργοποίησης μπορεί να προσεγγίσει οποιαδήποτε ομαλή συνάρτηση όσο κοντά επιθυμούμε.

Τα MLP είναι δίκτυα που ικανοποιούν τις παραπάνω συνθήκες και για αυτό το λόγο αναφέρονται και ως καθολικοί προσσεγγιστές.

2.6.4.1 Ανάκληση στα Πολυεπίπεδα Perceptron

Ανάκληση είναι η διαδικασία κατά την οποία, δοθέντος ενός διανύσματος εισόδου $\boldsymbol{x}$ υπολογίζονται οι ενεργοποιήσεις όλων των νευρώνων του δικτύου. Υπενθυμίζεται, πως κάθε νευρώνας του στρώματος l τροφοδοτείται αποκλειστικά από τους νευρώνες του στρώματος $l-1$,συνεπώς ο υπολογισμός των ενεργοποιήσεων των νευρώνων εξόδου απαιτεί πρώτα τον υπολογισμό των εξόδων των νευρώνων των προηγούμενων επιπέδων.

Έστω :

- L το πλήθος των επιπέδων του δικτύου.
- N(i)το πλήθος των νευρώνων του επιπέδου i.
- $a_i(l)$ η ενεργοποιήση του νευρώνα i στο επίπεδο l .
- $w_{ij}(l)$ το συναπτικό βάρος που συνδέει τον νευρώνα j του $l-1$ επιπεδου και του νευρώνα i του l επιπέδου.
- $w_{i0}(l)$ το κατώφλιτου νευρώνα i.
- $x_i = a_i(0)$ οι είσοδοι του δικτύου και $y_i = a_i(L)$ οι έξοδοι.

Γενικεύοντας λοιπόν τη διαδικασία της ανάκλησης του απλού δικτύου Perceptron,οι ενεργοποιήσεις των νευρώνων για κάθε επίπεδο του δικτύου δίνονται από τη σχέση :

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

(Όπου f η συνάρτηση ενεργοποίησης του αντίστοιχου νευρώνα.)

2.6.4.2 Εκπαίδευση Δικτύων MLP

Η διαδικασία εκπαίδευσης των πολυεπίπεδων Perceptron, περιλαμβάνει την ρύθμιση των συναπτικών βαρών του δικτύου με σκοπό την ικανοποίηση κάποιου κριτήριου καταλληλόλητας.

Ο κυριότερος και πιο δημοφιλής αλγόριθμος εκπαίδευσης δικτύων MLP είναι ο αλγόριθμος της πίσω διάδοσης λάθους (Back Propagation Algorithm). Ο συγκεκριμένος αλγόριθμος ανήκει στην ευρύτερη οικογένεια των αλγορίθμων επιβλεπόμενης μάθησης.

Συνεπώς η εκπαίδευση ενός MLP απαιτεί ένα σύνολο δεδομένων το οποίο θα αποτελείται από στιγμιότυπα της μορφής $(\boldsymbol{x}^{(P)}, \boldsymbol{d}^{(P)})$ **,** όπου $\boldsymbol{x}^{(P)}$ το διάνυσμα εισόδου και **,** $\boldsymbol{d}^{(P)}$ το διάνυσμα «στόχος» εξόδου του P-στού στιγμιότυπου εκπαίδευσης.

Έστω $\boldsymbol{y}^{(P)}$ η απόκριση του μοντέλου στο διάνυσμα εισόδου $\boldsymbol{x}^{(P)}$**,** η οποία προκύπτει από τη διαδικασία ανάκλησης όπως αναφέρουμε παραπάνω.

Σκοπός του συγκεκριμένου αλγόριθμου εκπαίδευσης είναι να ελαχιστοποιηθεί το μέσο τετραγωνικό σφάλμα J:

$$J = \frac{1}{P}\sum_{p=1}^{P}\left\|\boldsymbol{d}^{(P)} - \boldsymbol{y}^{(P)}\right\|^2$$

το οποίο αποτελεί το κριτήριο κόστους που χρησιμοποιείται ευρέως σε πολλά προβλήματα μηχανικής μάθησης.

2.6.4.2.1 Κατάβαση Δυναμικού

Από τη σχέση της ανάκλησης των νευρωνικών δικτύων και το γεγονός πως $y_i = a_i(L)$ και $\boldsymbol{d}^{(P)}$ γνωστό διάνυσμα «στόχος» για κάθε πρότυπο εισόδου, καταλήγουμε πως τα συναπτικά βάρη w_{ij} ,είναι οι παράμετροι που πρέπει να διορθωθούν ώστε να ελαχιστοποιηθεί το σφάλμα.

Συνεπώς η εκπαίδευση του νευρωνικού δικτύου ανάγεται ,στο γνωστό πρόβλημα του μαθηματικού λογισμού, αυτό της *ελαχιστοποίησης μιας συνάρτησης πολλών μεταβλητών .*

Η μέθοδος που χρησιμοποιείται συνήθως σε τέτοια πολύπλοκα προβλήματα (καθώς η συνάρτηση J αποτελεί συνήθως συνάρτηση χιλιάδων διαστάσεων**)** είναι η επαναληπτική μέθοδος της **Κατάβασης Δυναμικού.**

Η μέθοδος αυτή ξεκινάει από ένα τυχαίο σημείο του χώρου εισόδου και κινείται επαναληπτικά στην αντίθετη κατεύθυνση της κλίσης της συνάρτησης J, μαθηματικά στην κατεύθυνση $-\boldsymbol{\nabla J}$ και με βήμα ανάλογο του μέτρου της.

Έστω συνάρτηση στον τρισδιάστατο χώρο $f(x, y)$ τότε για τη κλίση f της ισχύει:

$$\nabla f = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j} + \frac{\partial f}{\partial z}\mathbf{k}$$

Εφαρμόζοντας τη παραπάνω ιδέα στο πρόβλημα ελαχιστοποίησης της συνάρτησης κόστους J ενός Νευρωνικού Δικτύου, ως προς τα συναπτικά βάρη, η Κατάβαση Δυναμικού εξομοιώνεται με μια διακριτή εξίσωση :

$$\boxed{\mathrm{w_{ij}}(l,k+1)-\mathrm{w_{ij}}(l,k)=-\beta\frac{\partial J}{\partial \mathrm{w_{ij}}(l,k)}}$$

- Όπου k, ο αριθμός της επανάληψης που βρίσκεται ο αλγόριθμος, όπως εξετάσαμε και στην ενότητα του δικτύου Perceptron. Κάθε επανάληψη, αντιστοιχεί στην εφαρμογή ενός στιγμιότυπο εκπαίδευσης στο Δίκτυο καθώς και στην αλλαγή που αυτό επιφέρει στα βάρη του δικτύου.

- Η παράμετρος β είναι ένας μικρός θετικός αριθμός (συνάρτηση του k) και ονομάζεται βήμα καθώς επηρεάζει το ρυθμό μεταβολής των βαρών.

2.6.4.2.2 Ο Αλγόριθμος πίσω διόρθωσης σφάλματος

- Για την ανάλυση του αλγόριθμου είναι χρήσιμο να θυμίσουμε πως η τιμή ενεργοποίησης του νευρώνα i στο επίπεδο l και κατά την επανάληψη k είναι :

$$a_i{}^{(k)}(l)=f\left(u_i{}^{(k)}(l)\right)$$

όπου :

$$\boxed{u_i{}^{(k)}(l)=\sum_{j=1}^{N(l-1)}\mathrm{w_{ij}}(l,k)\mathrm{a}_j{}^{(k)}(l-1)+\mathrm{w_{i0}}(l,k)}$$

Γνωστή και ως δικτυακή διέγερση του νευρώνα.

- Επίσης χρησιμοποιώντας τον κανόνα αλυσίδας από τον διαφορικό λογισμό έχουμε:

$$\frac{\partial J}{\partial \mathrm{w_{ij}}(l,k)}=\frac{\partial J}{\partial u_i{}^{(k)}(l)}\frac{\partial u_i{}^{(k)}(l)}{\partial \mathrm{w_{ij}}(l,k)}=\delta_i^{(k)}(l)\,\frac{\partial u_i{}^{(k)}(l)}{\partial \mathrm{w_{ij}}(l,k)}$$

Όπου $\frac{\partial u_i{}^{(k)}(l)}{\partial \mathrm{w_{ij}}(l,k)}=\begin{cases}a_j{}^{(k)}(l-1) & \alpha\nu\ j\neq 0\\ 1 & \alpha\nu\ \ j=0\end{cases}$

- Χρησιμοποιώντας τον κανόνα αλυσίδας για συναρτήσεις πολλών μεταβλητών καταλήγουμε στις εξής εκφράσεις για τη παράμετρο δ_i :

  - $\delta_i^{(k)}(L) = \left(d_i^{(k)} - y_i^{(k)}\right) f'(u_i{}^{(k)}(L))$ *για το επίπεδο εξόδου* $l = L$.

  - $\delta_i^{(k)}(l) = f'(u_i{}^{(k)}(l)) \sum_{\mu=1}^{N(l+1)} w_{\mu i}{}^{(k)}(l+1)\delta_\mu{}^{(k)}(l+1)$ *για* $l = 1, \dots, L-1$

Έχοντας διατυπώσει μαθηματικά τις απαραίτητες παραμέτρους, παρουσιάζουμε παρακάτω την αλγοριθμική διαδικασία που ακολουθείται κατά την εκπαίδευση ενός νευρωνικού δικτύου:

Εκπαίδευση ενός MLP Νευρωνικού Δικτύου

1. *Το πρώτο βήμα είναι η επιλογή της αρχιτεκτονικής του δικτύου και η αρχικοποίηση των συναπτικών βαρών. Κοινή πρακτική αποτελεί η αρχικοποίηση των βαρών με μικρού τυχαίους αριθμούς μηδενικής μέση τιμής.*

2. *Κατά τη δεύτερη φάση της εκπαίδευσης το δίκτυο τροφοδοτείται με τα στιγμιότυπα εκπαίδευσης ,δηλαδή με ζεύγη της μορφής* $\left(\boldsymbol{x}^{(P)}, \boldsymbol{d}^{(P)}\right)$. *Για κάθε πρότυπο* p *ακολουθείται η διαδικασία της ανάκλησης (υπολογισμός προς τα εμπρός) και ο υπολογισμός των παραμέτρων* δ_i *για κάθε επίπεδο.*

3. *Όταν ολοκληρωθούν οι παραπάνω υπολογισμοί ,έπεται η διαδικασία ενημέρωσης των βαρών με χρήση του τύπου :*

$$\mathrm{w_{ij}}(l, k+1) = \mathrm{w_{ij}}(l, k) + \beta \delta_i^{(k)}(l)\, \mathrm{a}_j{}^{(k)}(l-1)\,,$$

όπως έχει αναλυθεί παραπάνω.

Η παραπάνω διαδικασία επαναλαμβάνεται για κάθε πρότυπο εκπαίδευσης μέχρι να ικανοποιηθεί το κριτήριο τερματισμού, το οποίο μπορεί να είναι:

- Η συμπλήρωση ενός συγκεκριμένου αριθμού εποχών εκπαίδευσης.

- Η μείωση του σφάλματος J κάτω από μια τιμή ε.

- Η αμελητέα μείωση του σφάλματος μεταξύ δύο διαδοχικών εποχών.
- Η αμελητέα τροποποίηση των συναπτικών βαρών στο τέλος μιας εποχής.

Ο παραπάνω αλγόριθμος πάσχει από αργή σύγκλιση ,αλλά και από τον εγκλωβισμό σε τοπικά ακρότατα γεγονός που μπορεί να οδηγήσει σε λύσεις χαμηλής ποιότητας αλλά και σε υψηλούς χρόνους εκπαίδευσης.

Ωστόσο έχουν αναπτυχθεί διάφορες μέθοδοι που αποσκοπούν στη βελτίωση της απόδοσης του συγκεκριμένου αλγορίθμου, όπως ***η χρήση της ορμής (momentum), η αναζήτηση σε ευθεία (line search)*** **και** ***η συζυγής κατάβαση δυναμικού (conjugate gradient)***, δε θα προχωρήσουμε σε περαιτέρω ανάλυση αυτών.

2.7 Γενετικοί Αλγόριθμοι

Σε αρκετές περιπτώσεις το μέγεθος ενός προβλήματος καθιστά αδύνατη τη χρήση συμβατικών μεθόδων αναζήτησης. Το παραπάνω ζήτημα εμφανίζεται συχνά σε προβλήματα βελτιστοποίησης όπου συνήθως μια μέθοδος εξαντλητικής αναζήτησης θα οδηγούσε σε υπερβολικά μεγάλους χρόνους τερματισμού. Σε αυτές τις περιπτώσεις γίνεται χρήση πιθανοκρατικών και ευριστικών αλγόριθμων οι οποίοι δεν εγγυώνται την εύρεση της βέλτιστης λύσης αλλά είναι ικανοί να επιστρέψουν μια αρκετά καλή λύση ,σε ένα σχετικά σύντομο χρονικό διάστημα.

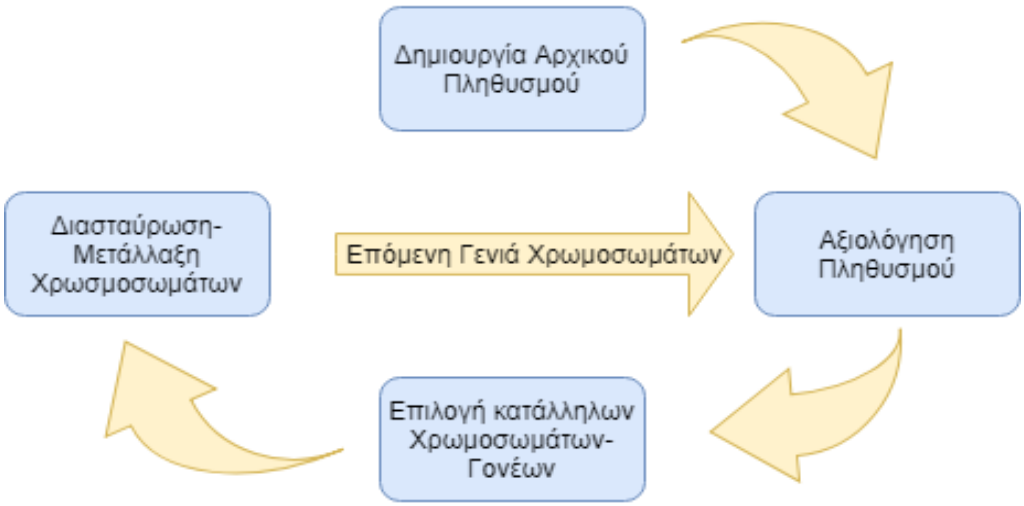
Οι Γενετικοί Αλγόριθμοι (ΓΑ) είναι μια ευριστική τεχνική αναζήτησης, η οποία ανήκει στην ευρύτερη οικογένεια των εξελεγκτικών αλγόριθμων ,αρχικά υποκινημένη από τη δαρβινική αρχή της εξέλιξης μέσω της γενετικής επιλογής. Ένας Γενετικός Αλγόριθμος χρησιμοποιεί μια άκρως αφηρημένη έκδοση των εξελικτικών διαδικασιών, για την εξεύρεση λύσεων σε συγκεκριμένα προβλήματα. Κάθε ΓΑ λειτουργεί πάνω σε ένα γνωστό σύνολο-πληθυσμό από τεχνητά χρωμοσώματα, που αντιπροσωπεύουν τις πιθανές λύσεις του προβλήματος. Τα χρωμοσώματα είναι στη ουσία συμβολοσειρές ενός πεπερασμένου αλφαβήτου συνήθως του δυαδικού (0 ή 1) και κάθε ένα από αυτά , βαθμολογείται από μια συνάρτηση καταλληλόλητας (Fitness Function).

Η φιλοσοφία λοιπόν των Γενετικών Αλγόριθμων είναι η εξής:

Ξεκινώντας με έναν τυχαία δημιουργημένο πληθυσμό χρωμοσωμάτων ένας ΓΑ διεξάγει μια διαδικασία επιλογής, βασισμένη στη συνάρτηση καταλληλόλητας, και μια διαδικασία συνδυασμού-αναπαραγωγής χρωμοσωμάτων για να παραχθεί ένας διάδοχος πληθυσμός. Κατά τη διάρκεια του συνδυασμού,

επιλέγονται γονικά χρωμοσώματα και το γενετικό τους υλικό διασταυρώνεται, για την παραγωγή νέων απογόνων χρωμοσωμάτων.

Καθώς αυτή η διαδικασία επαναλαμβάνεται, μια ακολουθία διαδοχικών γενεών εξελίσσεται και η μέση καταλληλόλητα των χρωμοσωμάτων τείνει να αυξάνεται μέχρις ότου ικανοποιηθεί το κριτήριο του τερματισμού . Με αυτόν τον τρόπο, ένας Γενετικός Αλγόριθμος "εξελίσσει" μια καλή λύση σε ένα συγκεκριμένο πρόβλημα.



Εικόνα 2-11 Λειτουργία Γενετικών Αλγόριθμών

2.7.1 Λειτουργία των Γενετικών αλγόριθμων

Όπως γίνεται αντιληπτό και από τα παραπάνω ένας Γενετικός Αλγόριθμος, για ένα συγκεκριμένο πρόβλημα περιλαμβάνει πέντε βασικά συστατικά-λειτουργίες:

- *Δημιουργία αρχικού πληθυσμού λύσεων και κωδικοποίηση χρωμοσωμάτων.*
- *Δημιουργία μιας συνάρτησης καταλληλόλητας*
- *Ένα μηχανισμό επιλογής χρωμοσωμάτων-γονέων.*
- *Δημιουργία μηχανισμού διασταύρωσης και μετάλλαξης χρωμοσωμάτων*
- *Εξέλιξη και σύγκλιση*

Κωδικοποίηση χρωμοσωμάτων

Ένας Γενετικός Αλγόριθμος χειρίζεται πληθυσμούς χρωμοσωμάτων τα οποία είναι συμβολοσειρές λύσεων για ένα συγκεκριμένο πρόβλημα. Τα συγκεκριμένα χρωμοσώματα είναι αφαιρέσεις χρωμοσωμάτων γενετικού υλικού-DNA τα οποία μπορούν να αναπαρασταθούν με συμβολοσειρές ενός αλφάβητου π.χ. {Α,Β,Δ,Γ}.

Κάθε διαφορετική θέση σε ένα χρωμόσωμα ονομάζεται γονίδιο. Η αλφάβητος που θα επιλέγει για να αναπαραστήσει τα γονίδια ονομάζεται κωδικοποίηση του Γενετικού Αλγόριθμου για ένα συγκεκριμένο πρόβλημα. Η πιο κλασική και δημοφιλής προσέγγιση είναι οι συμβολοσειρές δυαδικών χαρακτήρων, δηλαδή το αλφάβητο {0-1}.
Όπως γίνεται αντιληπτό το χρωμόσωμα από μόνο του περιέχει περιορισμένες πληροφορίες για το συγκεκριμένο πρόβλημα .Μεγάλο μέρος της σημασίας ενός συγκεκριμένου χρωμοσώματος κωδικοποιείται στο δεύτερο συστατικό του γενετικού αλγόριθμου τη συνάρτηση καταλληλόλητας.

Συνάρτηση καταλληλόλητας

Η συνάρτηση καταλληλόλητας αποτελεί το κριτήριο αξιολόγησης των χρωμοσωμάτων, δηλαδή των πιθανών λύσεων. Σε αναλογία με τη βιολογία ένα χρωμόσωμα μπορεί να αναφερθεί ως γενότυπο ενώ η λύση που εκπροσωπεί ως φαινότυπο. Η συνάρτηση καταλληλόλητας παίρνει σαν είσοδο ένα χρωμόσωμα και επιστρέφει ένα αριθμό που συμβολίζει το βαθμό καταλληλόλητας της εν λόγω λύσης. Η συνάρτηση καταλληλόλητας μπορεί να είναι από πολύ απλή έως και ιδιαίτερα πολύπλοκη πράγμα που εξαρτάται κυρίως από το εκάστοτε πρόβλημα. Στη ιδανικότερη περίπτωση η συγκεκριμένη συνάρτηση θα είναι συνεχής και μονότονη και συνεπώς μια απλή αναζήτηση αναρρίχησης λόφου θα αρκούσε για να βρεθεί η βέλτιστη λύση. Προφανώς αυτό σπάνια συμβαίνει και η προσπάθεια επικεντρώνεται στη εύρεση μιας συνάρτησης με λίγα τοπικά μέγιστα ή ενός σχετικά απομονωμένου ολικού μεγίστου.

Επιλογή χρωμοσωμάτων-γονέων

Η διαδικασία της επιλογής έχει τη ικανότητα να χρησιμοποιεί την συνάρτηση καταλληλόλητας ως οδηγό για την εξέλιξη των χρωμοσωμάτων.

Πιο συγκεκριμένα η διαδικασία επιλογής χρωμοσωμάτων-γονέων σχετίζεται με την απόδοση πιθανοτήτων αναπαραγωγής στα χρωμοσώματα του πληθυσμού. Όσο μεγαλύτερη είναι η τιμή της συνάρτησης καταλληλόλητας για ένα χρωμόσωμα τόσο πιο πιθανό να επιλέγει για αναπαραγωγή. Για τη συγκεκριμένη διαδικασία χρησιμοποιούνται αρκετές τεχνικές με μια από τις πιο δημοφιλείς αυτή της επιλογής ρουλέτας (roulette wheel selection). Αυτή αποδίδει σε κάθε χρωμόσωμα πιθανότητα να επιλέγεται, ανάλογη με τη σχετική φυσική κατάσταση του, η οποία είναι η φυσική του κατάσταση ως ποσοστό του συνόλου

των φυσιολογικών όλων των χρωμοσωμάτων στον πληθυσμό. Φυσική κατάσταση ενός χρωμοσώματος είναι ο βαθμός καταλληλόλητας της λύσης που αντιπροσωπεύει.

<u>Διασταύρωση-Μετάλλαξη χρωμοσωμάτων</u>

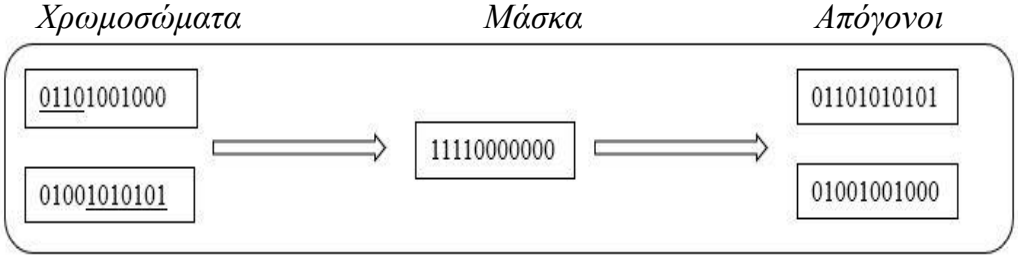
Η διαδικασία της αναπαραγωγής στους γενετικούς αλγορίθμους είναι εκείνη κατά την οποία χρωμοσώματα ,δηλαδή αναπαραστάσεις πιθανών λύσεων, από ένα πληθυσμό διασταυρώνονται και μεταλλάσσονται ώστε να δημιουργηθεί ένας νέος καταλληλότερος πληθυσμός απογόνων. Η παραπάνω διαδικασία προσομοιώνει την ανάμειξη του γενετικού υλικού που συμβαίνει όταν αναπαράγονται ζωντανοί οργανισμοί. Στη διαδικασία της αναπαραγωγής εμπλέκονται ένα σύνολο από τελεστές οι οποίοι αντιστοιχούν σε διαδικασίες βιολογικής εξέλιξης. Οι πιο κοινοί τελετές είναι αυτοί της διασταύρωσης (crossover) και της μετάλλαξης (mutation).

Η μετάλλαξη είναι μια διαδικασία κατά την οποία αλλάζει τυχαία ένα δυαδικό ψηφίο σε ένα χρωμόσωμα και εφαρμόζεται συνήθως μετά τη διασταύρωση. Σκοπός της μετάλλαξης είναι να εισάγει μια μορφή τυχαιότητας στους απογόνους, ώστε να αποφευχθούν οι συγκλίσεις σε τοπικά μέγιστα.

Υπάρχουν αρκετές τεχνικές διασταύρωσης, με την πιο ευρέως χρησιμοποιούμενη να είναι αυτή που χρησιμοποιεί την μάσκα διασταύρωσης ,δηλαδή μια δυαδική συμβολοσειρά που υποδεικνύει ποιος γονέας θα συνεισφέρει το κάθε δυαδικό ψηφίο στο χρωμόσωμα απόγονο. Οι πιο γνωστοί τελεστές είναι: η διασταύρωση ενός σημείου, η διασταύρωση δύο σημείων και τέλος η ομοιόμορφη διασταύρωση.

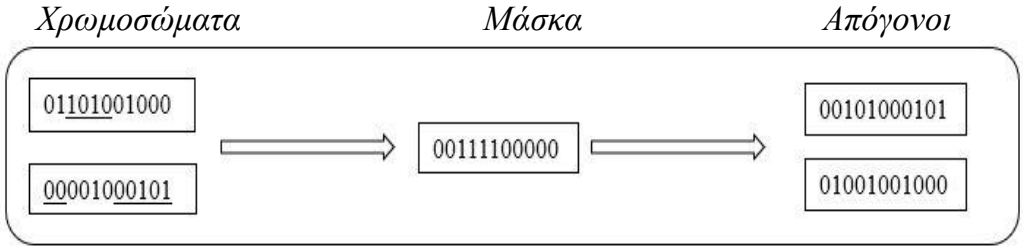
- ***Διασταύρωση ενός σημείου***

  Στη διασταύρωση ενός σημείου η μάσκα ξεκινά με μια σειρά συνεχόμενων άσσων «1» και καταλήγει με συνεχόμενα μηδενικά. Κάθε απόγονος παίρνει μόνο τα δυαδικά ψηφία που δε χρησιμοποιήθηκαν για τη δημιουργία του άλλου. Η διαδικασία φαίνεται παρακάτω:



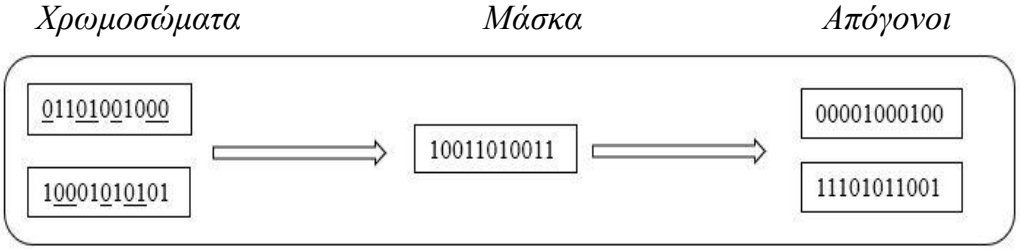
- ***Διασταύρωση δύο σημείων***

Στη διασταύρωση δύο σημείων οι απόγονοι δημιουργούνται από τον συνδυασμό των κεντρικών δυαδικών ψηφίων του ενός γονέα και πλευρικών ψηφίων του άλλου όπως θα καθορίζει η μάσκα διασταύρωσης . Η διαδικασία φαίνεται παρακάτω:



- ***Ομοιόμορφη διασταύρωση***

Στην ομοιόμορφη διασταύρωση, τα δυαδικά ψηφία των γονέων μοιράζονται ομοιόμορφα στους δύο απογόνους, σύμφωνα με τη τυχαία δημιουργημένη μάσκα διασταύρωσης.



2.7.2 Εξέλιξη και Σύγκλιση

Μετά τη διασταύρωση ,τα χρωμοσώματα που προκύπτουν δημιουργούν τον νέο πληθυσμό απογόνων-πιθανών λύσεων του γενετικού αλγόριθμου. Η διαδικασία της επιλογής της διασταύρωσης και της δημιουργίας νέων πληθυσμών επαναλαμβάνεται για ένα πλήθος γενεών μέχρι να ικανοποιηθούν τα όποια κριτήρια τερματισμού. Αυτά μπορεί να είναι, ένας προκαθορισμένος αριθμός γενεών ,σύγκλιση στη καλύτερη και μόνο λύση, ή η σύγκλιση σε μια ικανοποιητική γενιά λύσεων που ικανοποιούν συγκεκριμένες συνθήκες. Υπάρχουν αρκετές διαφορετικές τεχνικές εξέλιξης ενός πληθυσμού όπως η ολική αντικατάσταση ,όπου όλα τα μέλη ενός πληθυσμού αντικαθίστανται στη επόμενη γενιά ή ελιτιστική επιλογή κατά την οποία το καλύτερο χρωμόσωμα ή

τα δύο καλύτερα μεταφέρονται αμετάβλητα στην επόμενη γενιά εξασφαλίζοντας ότι η ποιότητα δε θα μειώνεται από γενιά σε γενιά.

Η επιλογή του εξελεγκτικού σχεδίου που θα ακολουθηθεί είναι μια σημαντική πτυχή της σχεδίασης γενετικών αλγόριθμων και θα εξαρτηθεί από τη φύση του χώρου λύσεων του προβλήματος.

2.7.3 Ρύθμιση Νευρωνικών Δικτύων με Γενετικούς Αλγόριθμους

Στα περισσότερα προβλήματα η απόδοση των νευρωνικών δικτύων εξαρτάται σε μεγάλο βαθμό από τη ρύθμιση των σχεδιαστικών παραμέτρων τους, όπως το πλήθος των κρυφών επιπέδων και ο αριθμός των νευρώνων σε κάθε ένα από αυτά. Η επιλογή των παραμέτρων αυτών δεν γίνεται βάση συγκεκριμένων κανόνων, αλλά κυρίως στηρίζεται σε ενδελεχή πειράματα ή στη εμπειρία και το ένστικτο του μηχανικού. Όπως γίνεται γρήγορα αντιληπτό ο χώρος αναζήτησης που δημιουργείται από τον εξαντλητικό συνδυασμό
όλων αυτών των παραμέτρων είναι εξαιρετικά μεγάλος ώστε να διασχιστεί αποδοτικά χρησιμοποιώντας μεθόδους ωμής βίας.

Εφαρμόζοντας ένα γενετικό αλγόριθμο με στόχο την εύρεση των υπερπαραμέτρων των νευρωνικών δικτύων, μπορούμε να καταλήξουμε σε μια καλή λύση σε πολύ μικρό χρονικό διάστημα εξοικονομώντας έτσι υπολογιστικούς πόρους.

Η διαδικασία που ακολουθείται είναι η εξής:

- *Δημιουργία ενός πληθυσμού νευρωνικών δικτύων*
- *Εκπαίδευση των νευρωνικών δικτύων του πληθυσμού*
- *Υπολογισμός της καταλληλόλητας κάθε νευρωνικού δικτύου.*
- *Επιλογή νευρωνικών δικτύων*
- *Διασταύρωση νευρωνικών δικτύων*
- *Δημιουργία επόμενης γενιάς*
- *Επανάληψη μέχρι να ικανοποιηθεί το κριτήριο τερματισμού.*

***Υπολογισμός της καταλληλόλητας κάθε νευρωνικού δικτύου*.**

Για κάθε Νευρωνικό Δίκτυο που δημιουργείται από τη διαδικασία της διασταύρωσης του γενετικού αλγόριθμου χρειαζόμαστε μια συνάρτηση καταλληλόλητας η οποία θα αξιολογεί τα δίκτυα και θα καθορίζει την πιθανότητα διασταύρωσης τους. Η συνάρτηση αυτή θα είναι το σφάλμα του νευρωνικού δικτύου σε ένα σύνολο δεδομένων γνωστό ως σύνολο επαλήθευσης ή επικύρωσης.

Συνεπώς το σύνολο των δεδομένων εισόδου χωρίζεται σε σύνολο εκπαίδευσης, σύνολο επαλήθευσης και σύνολο ελέγχου ή δοκιμής το οποίο χρησιμοποιείται για την τελική επιλογή και αξιολόγηση των νευρωνικών δικτύων.Ο λόγος που το σφάλμα υπολογίζεται στο σύνολο επαλήθευσης ,το οποίο είναι ξένο με τα σύνολο δοκιμής, είναι η επιθυμία για τη διατήρηση της αμεροληψίας του συνόλου δοκιμής το οποίο χρησιμοποιείται για τη τελική αξιολόγηση του μοντέλου.

Επιλογή παραμέτρων ρύθμισης

Βασικό σημείο της εξέλιξης ενός νευρωνικού δικτύου, είναι η επιλογή των υπερπαραμέτρων που θα ρυθμιστούν από τον Γενετικό Αλγόριθμο καθώς και το εύρος του διαστήματος επιλογής αυτών. Κάθε Νευρωνικό Δίκτυο θα αποτελεί ένα χρωμόσωμα το οποίο αποτελείται από τα γονίδια, το πλήθος των οποίων εξαρτάται από το πλήθος και το εύρος των υπερπαραμέτρων.

Συνεπώς γίνεται εύκολα αντιληπτό πως ο χρόνος τερματισμού της όλης διαδικασίας αλλά και το αποτέλεσμα αυτής καθορίζεται σε μεγάλο βαθμό από το μέγεθος του χώρου αναζήτησης που θα δημιουργηθεί. Κοινή πρακτική είναι οι παράμετροι ρύθμισης να είναι :

1. *Το πλήθος των επιπέδων του νευρωνικού δικτύου*
2. *Το πλήθος των νευρώνων σε κάθε επίπεδο*
3. *Οι συναρτήσεις μεταφοράς κάθε επιπέδου*

3 Εργαλεία και Τεχνολογίες

Στο Κεφάλαιο αυτό παρουσιάζονται οι βασικές τεχνολογίες που χρησιμοποιήθηκαν για την διεκπεραίωση της συγκεκριμένης διπλωματικής εργασίας. Πιο συγκεκριμένα, παρουσιάζεται η γλώσσα GNU Octave η αποτελεί την γλώσσα υλοποίησης του ερευνητικού εργαλείου Mapping Models. Στη συνέχεια αναλύεται η σύγχρονη βιβλιοθήκη Μηχανικής Μάθησης της γλώσσας Python, scikit-learn η οποία χρησιμοποιήθηκε για τον σχεδιασμό των Τυχαίων Δασών και των Μηχανών Διανυσμάτων Υποστήριξης ,αλλά και για την οργάνωση και εκτέλεση των πειραμάτων. Τέλος παρουσιάζεται η βιβλιοθήκη επίσης της γλώσσας Python Pandas η οποία χρησιμοποιήθηκε για την διαχείριση των συνόλων δεδομένων.

3.1 GNU Octave

Το GNU Octave αποτελεί μια γλώσσα υψηλού επιπέδου η οποία προορίζεται κυρίως για αριθμητικούς υπολογισμούς. Χρησιμοποιείται κυρίως για προβλήματα όπως η επίλυση γραμμικών και μη γραμμικών εξισώσεων, η αριθμητική ανάλυση, η στατιστική ανάλυση αλλά και για την εκτέλεση άλλων αριθμητικών πειραμάτων. Μπορεί επίσης να χρησιμοποιηθεί και για λειτουργίες αυτοματοποιημένης επεξεργασίας δεδομένων.

Οι σύγχρονες εκδόσεις του Octave παρέχουν προγραμματιστικό περιβάλλον το οποίο διαθέτει μια γραφική διεπαφή χρήστη για ην ευκολότερη διαχείριση των εφαρμογών του προγραμματιστή. Η διεπαφή αυτή φιλοξενεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών, το οποίο περιλαμβάνει έναν επεξεργαστή κώδικα με επισημάνσεις σύνταξης ,ενσωματωμένο αλγόριθμο εντοπισμού σφαλμάτων καθώς και έναν εξελιγμένο διερμηνέα.

Το GNU Octave προσφέρει επίσης ένα λογισμικό πακέτο για τον σχεδιασμό και την εκπαίδευση Πολυεπίπεδων Νευρωνικών Δικτύων Perceptron προσφέροντας έτσι τα απαραίτητα εργαλεία για την γρήγορη και αποτελεσματική δημιουργία MLPs.

Το Octave είναι το εργαλείο –γλώσσα που χρησιμοποιήθηκε για την ανάπτυξη του ερευνητικού εργαλείου Mapping Models (Νευρωνικά Δίκτυα ρυθμιζόμενα από έναν Γενετικό Αλγόριθμο) το οποίο χρησιμοποιήσαμε για την εκτέλεση των πειραμάτων της συγκεκριμένης διπλωματικής.

Το GNU Octave είναι ελεύθερα αναδιανεμητέο λογισμικό και μπορεί να αναδιανεμηθεί και να τροποποιηθεί σύμφωνα με τους όρους της Γενικής Δημόσιας Άδειας GNU όπως δημοσιεύεται από το Ίδρυμα Ελεύθερου Λογισμικού.

3.2 Η Βιβλιοθήκη Scikit-learn

Η γλώσσα προγραμματισμού Python έχει καθιερωθεί ως μία από τις πιο δημοφιλείς γλώσσες στον κλάδο που ονομάζεται υπολογιστική επιστήμη (Scientific Computing). Χάρη στη διαδραστική υψηλού επιπέδου φύση της και των ώριμων βιβλιοθηκών στο συγκεκριμένο κλάδο αποτελεί μια ελκυστική επιλογή για την αλγοριθμική ανάπτυξη και διερεύνηση και ανάλυση δεδομένων. Ωστόσο η Python ως μια γλώσσα γενικού σκοπού χρησιμοποιείται ,εκτός από τον ακαδημαϊκό χώρο, και στη βιομηχανία.

Η βιβλιοθήκη scikit-learn αξιοποιεί αυτό το πλούσιο προγραμματιστικό περιβάλλον που προσφέρει η εν λόγω γλώσσα προγραμματισμού ,για να παρέχει αποδοτικές και σύγχρονες υλοποιήσεις αλγόριθμων μηχανικής μάθησης, διατηρώντας παράλληλα μια εύκολη στη χρήση προγραμματιστική διεπαφή (API) στενά ενσωματωμένη στη γλώσσα Python.

Η ανάπτυξη των παραπάνω ανταποκρίνεται στην αυξανόμενη ανάγκη για στατιστική ανάλυση δεδομένων από μη ειδικούς στην επιστήμη των υπολογιστών καθώς και από επιστήμονες διαφόρων επιστημών όπως η βιολογία και η φυσική.

Το scikit-learn είναι μια βιβλιοθήκη αφοσιωμένη στη μοντελοποίηση δεδομένων και την υλοποίηση αλγορίθμων και όχι στη φόρτωση και διαχείριση αυτών, όπως είανι η Pandas (βλ. παρακάτω),είναι γραμμένη σε Python αλλά και Cython ώστε να επιτευχθεί μεγαλύτερη απόδοση και διανέμεται υπό την άδεια BSD.

Ορισμένα παραδείγματα αλγόριθμων ταξινόμησης και παλινδρόμησης είναι οι Μηχανές Διανυσμάτων Υποστήριξης ,τα Τυχαία Δάση, και η Γραμμική Παλινδρόμηση.

Υποκείμενες Τεχνολογίες

Οι τεχνολογίες στις οποίες είναι βασισμένη η βιβλιοθήκη του scikit-learn είναι:

Scipy: Λογισμικό πακέτο που περιέχει αποδοτικούς αλγόριθμους γραμμικής άλγεβρας και βασικές λειτουργίες από το κλάδο της στατιστικής. Το Scipy έχει συνδέσεις για πολλά αριθμητικά πακέτα βασισμένα στη γλώσσα Fortran.

Numpy: Το Numpy αποτελεί το θεμελιώδες λογισμικό πακέτο επιστημονικής υπολογιστικής της γλώσσας Python. Χρησιμοποιείται για την αναπαράσταση πολυδιάστατων πινάκων, περιέχει εργαλεία για τη διασύνδεση με τις γλώσσες C.C++,Fortran. Επίσης προσφέρει υλοποιήσεις αλγόριθμων γραμμικής άλγεβρας ,μετασχηματισμών (Fourier, Laplace) , αλλά και παρέχει δυνατότητες διαχείρισης και δημιουργίας τυχαίων αριθμών.

Cython: Η Cython είναι μια γλώσσα που χρησιμοποιείται για τον συνδυασμό των γλωσσών C και Python. Η διευκολύνει την επίτευξη της απόδοσης των μεταγλωττισμένων γλωσσών προγραμματισμού χρησιμοποιώντας το συντακτικό και της λειτουργίες που προσφέρει μια διερμηνευμένη (interpreted) γλώσσα σαν την Python.

Μερικά από τα μοντέλα και τους αλγόριθμους που προσφέρει το scikit-learn είναι:

- *Αλγόριθμοι για τη συλλογή και συσπείρωση δεδομένων (Clustering Algorithms)*
- *Τεχνικές για την σωστή εκτίμηση της απόδοσης εποπτευόμενων μοντέλων (Cross Validation,k-fold Cross Validation κτλ)*
- *Παροχή συνόλων δεδομένων για εκτέλεση δοκιμών.*
- *Τεχνικές για την μείωση των διαστάσεων των δεδομένων (Dimensionality Reduction) για την μείωση των χαρακτηριστικών εισόδου των δεδομένων ,όπως ο αλγόριθμος της Ανάλυσης Κύριων Συνιστωσών (Principal Component Analysis).*
- *Αλγόριθμοι για την επιλογή και την εξαγωγή χαρακτηριστικών από τα σύνολα δεδομένων.*
- *Τεχνικές και αλγόριθμοι για την ρύθμιση παραμέτρων.*
- *Πληθώρα υλοποιήσεων για μοντέλα επιβλεπόμενης ή μη μάθησης.*

3.3 Η Βιβλιοθήκη Pandas

Η βιβλιοθήκη Pandas προσφέρει ένα σύνολο εργαλείων για την διαχείριση και επεξεργασία δομημένων συνόλων δεδομένων. Η βιβλιοθήκη παρέχει ενσωματωμένες και διαισθητικές ρουτίνες για την εκτέλεση κοινών λειτουργιών διαχείρισης σε δεδομένα από διάφορους τομείς όπως η στατιστική, τα

οικονομικά, οι κοινωνικές επιστήμες και πολλά άλλα επιστημονικά πεδία.

Το Pandas αποτελεί ένα ισχυρό συμπλήρωμα της στοίβας των εργαλείων που προσφέρει η Python ενώ ταυτόχρονα υλοποιεί και βελτιώνει λειτουργίες και εργαλεία που προσφέρονται από άλλες γλώσσες όπως η γλώσσα R. Αν και το Pandas ειδικεύεται στη διαχείριση δεδομένων και συνόλων από ανομοιογενή δεδομένα ,παρέχει ωστόσο χρήσιμα εργαλεία για ανάλυση δεδομένων (Data Analysis), παρέχοντας wrappers γύρω από τις τυποποιημένες στατιστικές μεθόδους και μεθόδους γραφικών της βιβλιοθήκης matplotlib.

4 Σχεδιασμός και Υλοποίηση Μοντέλων

Στο παρόν Κεφάλαιο θα παρουσιάσουμε λεπτομερείς υλοποίησης τόσο για το ερευνητικό εργαλείο Mapping Models το οποίο χρησιμοποιήθηκε για την σχεδίαση ,εκπαίδευση και ρύθμιση των Νευρωνικών Δικτύων, αλλά και για τα μοντέλα που αναπτύχθηκαν με τη χρήση της βιβλιοθήκης scikit-learn ,συγκεκριμένα τα Τυχαία Δάση και οι Μηχανές Διανυσμάτων Υποστήριξης.

Για κάθε έναν από τους παραπάνω αλγόριθμους θα παρουσιάζεται αναλυτικά η διαδικασία ρύθμισης των υπερπαραμέτρων του.

4.1 Σχεδίαση και Ρύθμιση Νευρωνικών Δικτύων - Mapping Models

Για την δημιουργία και ρύθμιση των Νευρωνικών Δικτύων χρησιμοποιήσαμε το ερευνητικό εργαλείο Mapping Models το οποίο είναι υλοποιημένο στη γλώσσα GNU Octave. Το συγκεκριμένο εργαλείο κατασκευάζει και εκπαιδεύει Πολυεπίπεδα Perceptron MLPs, χρησιμοποιώντας έναν γενετικό αλγόριθμο για την ρύθμιση των παραμέτρων των Νευρωνικών Δικτύων. Όπως έχουμε αναφέρει και σε παραπάνω κεφάλαια η ρύθμιση των παραμέτρων αποτελεί σημαντική διαδικασία η οποία θα καθορίσει σε μεγάλο βαθμό την απόδοση του εκάστοτε μοντέλου.

Η δημιουργία των Νευρωνικών Δικτύων, πραγματοποιείται με τη χρήση της συνάρτησης **'newff'** η οποία προσφέρεται από το ενσωματωμένο λογισμικό πακέτο Νευρωνικών Δικτύων που διαθέτει το GNU Octave. Το συγκεκριμένο πακέτο χρησιμοποιείται για τη δημιουργία δικτύων πρόσθιας τροφοδότησης η εκπαίδευση των οποίων γίνεται με τη χρήση του αλγόριθμου Back Propagation.

Η ρύθμιση των υπερπαραμέτρων γίνεται χρησιμοποιώντας έναν Γενετικό Αλγόριθμο προκειμένου να βελτιστοποιηθεί η επιλογή τους. Γενικά η επιλογή αυτών των παραμέτρων ,όπως έχει ήδη αναφερθεί αποτελεί μια δύσκολη και επίπονη διαδικασία και βασίζεται κυρίως στην ανθρώπινη εμπειρία. Η χρήση του γενετικού αλγορίθμου μας απαλλάσσει από τις παραπάνω δυσκολίες ενώ ταυτόχρονα αυτοματοποιεί τη διαδικασία επιλογής παραμέτρων, ανεξάρτητα από το εκάστοτε dataset στοιχίζοντας ωστόσο σε χρόνο εκπαίδευσης, αλλά και σε απόδοση του τελικού μοντέλου.

Συγκεκριμένα στα Mapping Models οι σχεδιαστικές παράμετροι που χρησιμοποιούνται στη εξελικτική διαδικασία είναι :

- *Το πλήθος των κρυφών επιπέδων του Νευρωνικού Δικτύου*
- *Το πλήθος των νευρώνων σε κάθε δίκτυο*
- *Η συνάρτηση ενεργοποίησης κάθε επιπέδου*

Η διαδικασία βελτιστοποίησης ξεκινάει με έναν αρχικό πληθυσμό, τυχαία ρυθμισμένων Νευρωνικών Δικτύων. Σε κάθε γενιά ο γενετικός αλγόριθμος επιλέγει τα καταλληλότερα Νευρωνικά Δίκτυα με βάση ένα πιθανοτικό σύστημα-κατανομή σχετικό με την απόδοση τους και τα διασταυρώνει.

Από αυτή τη διαδικασία δημιουργούνται νέες αρχιτεκτονικές δικτύων οι οποίες ύστερα από την κατάλληλη διαδικασία μετάλλαξης καθιστούν τον νέο πληθυσμό ο οποίος με τη σειρά του θα εκπαιδευτεί και θα αξιολογηθεί. Η παραπάνω διαδικασία συνεχίζεται μέχρι ο προκαθορισμένος από το χρήστη αριθμός γενεών ολοκληρωθεί.

4.1.1 Τεχνικά Χαρακτηριστικά Υλοποίησης

Παρακάτω παρουσιάζονται ορισμένα τεχνικά χαρακτηριστικά του Γενετικού Αλγόριθμου και των Νευρωνικών Δικτύων που αυτός ρυθμίζει.

Τεχνικά Χαρακτηριστικά του Γενετικού Αλγορίθμου

- Συνάρτηση Αξιολόγησης:
  Η συγκεκριμένη υλοποίηση χρησιμοποιεί το σφάλμα κάθε Νευρωνικού Δικτύου στο σύνολο εκπαίδευσης ως συνάρτηση αξιολόγησης του γενετικού αλγορίθμου

- Παράμετροι Ρύθμισης-Γονίδια**:**
  Οι μεταβλητές που αναπαριστούν τα γονίδια των χρωμοσωμάτων είναι:

  - *Πλήθος κρυφών επιπέδων*: Το μέγιστο επιτρεπτό πλήθος κρυφών επιπέδων είναι τα
    10 επίπεδα.

  - Πλήθος νευρώνων σε κάθε επίπεδο: Το πλήθος των νευρώνων σε κάθε επίπεδο μπορεί να εκτείνεται από έναν νευρώνα ως 30 νευρώνες

  - Συναρτήσεις μεταφοράς-ενεργοποίησης: Η συνάρτηση ενεργοποίησης κάθε επιπέδου επιλέγεται από τις παρακάτω τρεις υλοποιήσεις του GNU Octave:

- ***tansig:*** Μη γραμμική υπερβολική σιγμοειδή συνάρτηση ενεργοποίησης

- ***purelin:*** Γραμμική συνάρτηση ενεργοποίησης

- ***logsig***: Μη γραμμική σιγμοειδή συνάρτηση ενεργοποίησης (log-sigmoid)

Τεχνικά Χαρακτηριστικά των Ρυθμιζόμενων Νευρωνικών Δικτύων

- Αλγόριθμος εκπαίδευσης:
  Για την εκπαίδευση των Νευρωνικών Δικτύων χρησιμοποιείται η συνάρτηση του GNU Octave **trainlm** η οποία υλοποιεί την τεχνική Back Propagation χρησιμοποιώντας τον Levenberg-Marquardt αλγόριθμο.

- Παράμετροι εκπαίδευσης (Σταθεροί)

  - *Ο μέγιστος αριθμός εποχών εκπαίδευσης:*150

  - *Μέγιστος χρόνος εκπαίδευσης: 500 sec (προσαρμοσμένος στο πλήθος των δεδομένων)*

4.1.2 Κλιμάκωση των Χαρακτηριστικών στα Νευρωνικά Δίκτυα

Η κανονικοποίηση των δεδομένων εισόδου ή αλλιώς κλιμάκωση των χαρακτηριστικών εισόδου (Feature Scaling) αναφέρεται στη διαδικασία κατά την οποία οι κλίμακες των διαφορετικών χαρακτηριστικών εισόδου προσαρμόζονται σε μια θεωρητικά κοινή κλίμακα. Δηλαδή η κλιμάκωση των χαρακτηριστικών είναι η διαδικασία τυποποίησης του εύρους των ανεξάρτητων μεταβλητών-χαρακτηριστικών η οποία ανήκει στο σύνολο εκείνων των ενεργειών που στη Μηχανική Μάθηση ονομάζουμε προ-επεξεργασία δεδομένων.

Σημασία της κλιμάκωσης των χαρακτηριστικών εισόδου στα Νευρωνικά Δίκτυα

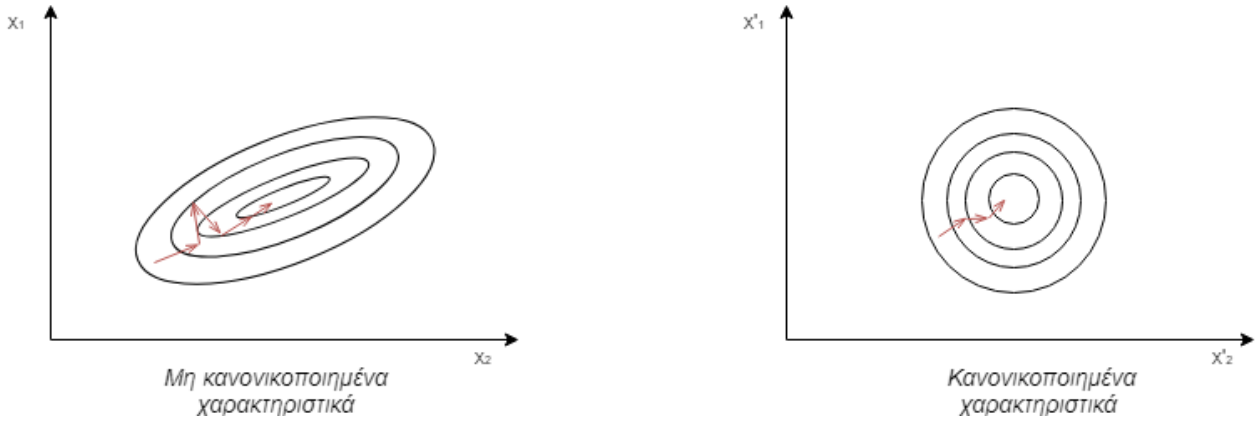
Όπως έχει ήδη αναφερθεί κατά την εκπαίδευση των νευρωνικών δικτύων χρησιμοποιείται η μέθοδος της Κατάβασης Δυναμικού, ώστε να βρεθεί το ελάχιστο της συνάρτησης κόστους του εκάστοτε δικτύου.

Πιο γενικά ,η Κατάβαση Δυναμικού είναι μια αναζήτηση στη επιφάνεια της συνάρτησης κόστους - σφάλματος του δικτύου σε μια προσπάθεια να βρεθούν οι τιμές για κάθε παράμετρο εισόδου έτσι ώστε να ελαχιστοποιηθεί η συγκεκριμένη συνάρτηση. Δηλαδή αναζητείται η μικρότερη τιμή της επιφάνειας

της συνάρτησης σφάλματος. Εξασφαλίζοντας ότι όλα τα χαρακτηριστικά - μεταβλητές εισόδου παίρνουν τιμές ίδιου εύρους-κλίμακας, αποδεικνύεται πως μπορούμε να βελτιώσουμε σημαντικά τη λειτουργία των αλγορίθμων εκπαίδευσης βασισμένων στη Κατάβαση Δυναμικού.

Το παραπάνω γίνεται εύκολα κατανοητό λαμβάνοντας υπόψιν πως η συνάρτηση σφάλματος χαρακτηρίζεται από τις μεταβλητές εισόδου και συνεπώς η τοπολογία της εξαρτάται άμεσα από το εύρος τιμών που αυτές εκτείνονται. Αυτό γίνεται αντιληπτό αν σκεφτούμε πως κάθε χαρακτηριστικό εισόδου θα αντιπροσωπεύει μια από τις διαστάσεις στις οποίες θα εκτείνεται η συνάρτηση σφάλματος.

Η μέθοδος της Κατάβασης Δυναμικού σε κάθε βήμα, υπολογίζει της απαραίτητές μεταβολές στις τιμές των παραμέτρων εισόδου ώστε να ελαχιστοποιηθεί η συνάρτηση σφάλματος. Ωστόσο στη περίπτωση που τα εύρη των παραμέτρων διαφέρουν σε μεγάλο βαθμό, η τιμή της κλίσης ως προς την μεγαλύτερη σε εύρος μεταβλητή, θα επηρεάζει σε μεγάλο βαθμό τη τελική ανανέωση των τιμών των παραμέτρων όπως γίνεται φανερό και από το παρακάτω σχήμα.



Εικόνα 4-1Κατάβαση Δυναμικού και Κανονικοποίηση Δεδομένων

Συνεπώς κανονικοποιώντας τα δεδομένα σε μια κοινή τυπική κλίμακα το δίκτυο συγκλίνει πιο γρήγορα στη βέλτιστη λύση. Επίσης, είναι σημαντικό η κλίμακα αυτή να κυμαίνεται ανάμεσα στο -1 και το 1 ώστε να αποφεύγονται τα πλαστά αποτελέσματα που προκύπτουν από τις πράξεις πολύ μεγάλων ή πολύ μικρών αριθμών κινητής υποδιαστολής. Το συγκεκριμένο εύρος, διευκολύνει στη σωστή λειτουργία των περισσότερων συναρτήσεων ενεργοποίησης οι οποίες λειτουργούν με πεδίο ορισμού το [-1,1].

Υπάρχουν αρκετοί μέθοδοι κλιμάκωσης χαρακτηριστικών με τις πιο γνωστές να είναι:

- *Η κανονικοποίηση Μεγίστου – Ελαχίστου*
- *Η κανονικοποίηση σταθμισμένου μέσου*
- *Η Τυποποίηση χαρακτηριστικών*
- *Η κλιμάκωση στη μονάδα*

Κλιμάκωση Χαρακτηριστικών στο εργαλείο Mapping Models

Τα Mapping Models εκτελούν κανονικοποίηση μεγίστου ελαχίστου στο εύρος [-1,1]. Έστω Min_i , Max_i το ελάχιστο και το μέγιστο του χαρακτηριστικού i. Τότε για κάθε μεταβλητή εισόδου x του $i-οστού$ χαρακτηριστικού έχουμε:

$$x' = -1 + \frac{(x - Min_i * 2)}{Max_i - Min_i}$$

Τα Mapping Models εκτός από την κανονικοποίηση των χαρακτηριστικών εισόδου εφαρμόζουν την ίδια τεχνική και για τη μεταβλητή εξόδου. Η κανονικοποίηση των μεταβλητών εξόδου βοηθάει στη δημιουργία καλών αρχικών βαρών και δεν αποτελεί αναγκαιότητα.

Το γεγονός όμως πως το Δίκτυο εκπαιδεύεται με βάση την κανονικοποιημένη μεταβλητή εξόδου δημιουργεί την ανάγκη ,κυρίως στα προβλήματα Ανάλυσης Παλινδρόμησης, για την αντίστροφη διαδικασία από την κλιμάκωση του χαρακτηριστικού. Αυτό προκύπτει από το γεγονός πως οι προβλέψεις του Νευρωνικού Δικτύου πρέπει να δίνονται στη αρχική κλίμακα του χαρακτηριστικού ώστε να μπορούν να χρησιμοποιηθούν σε πραγματικά σενάρια και όχι στο εύρος κλιμάκωσης [-1,1].

Η αντίστροφή αυτή διαδικασία δίνεται από την εξίσωση που προκύπτει αν λύσουμε την παραπάνω σχέση ως προς x.

$$x = \frac{1 + x' * (Max_i - Min_i)}{2} + Min_i$$

Οι παραπάνω διαδικασίες υλοποιούνται από τις συναρτήσεις :

- ***normalize*** (Υπεύθυνη για τη διαδικασία της κανονικοποίησης)
- ***denormalize*** (Υπεύθυνη για την επαναφορά της μεταβλητής εξόδου στην αρχική της κλίμακα)

οι υλοποιούνται από το εργαλείο Mapping Models.

4.2 Σχεδιασμός και Ρύθμιση Τυχαίων Δασών και Μηχανών Διανυσμάτων Υποστήριξης

Όπως είδαμε παραπάνω τα Mapping Models χρησιμοποιούν μια εξελικτική μέθοδο, συγκεκριμένα έναν γενετικό αλγόριθμο, για την ρύθμιση των υπερπαραμέτρων τους. Τα Νευρωνικά Δίκτυα αποτελούν περίπλοκα μοντέλα τα οποία χαρακτηρίζονται από τους υψηλούς χρόνους εκπαίδευσης. Επίσης το πλήθος των πλειάδων που προκύπτουν από τον συνδυασμό των υπερπαραμέτρων τους είναι αρκετά μεγάλο. Λόγω των παραπάνω γίνεται φανερό πως ο χώρος αναζήτησης που δημιουργείται είναι απαγορευτικά μεγάλος ,ώστε μέθοδοι όπως η εξαντλητική αναζήτηση, να είναι απαγορευτικοί.

Ωστόσο μοντέλα όπως τα Τυχαία Δάση και οι Μηχανές Διανυσμάτων Υποστήριξης χαρακτηρίζονται από συγκριτικά χαμηλότερους χρόνους εκπαίδευσης και οι υπερπαράμετροι αυτών μπορούν να περιοριστούν σε σχετικά μικρό εύρος χωρίς να θυσιάζεται η απόδοση των μοντέλων.

Για τους παραπάνω λόγους η μέθοδος που χρησιμοποιήθηκε για την ρύθμιση των παραμέτρων των παρακάτω μοντέλων είναι η Αναζήτηση Πλέγματος (Grid Search) η οποία όπως είχαμε αναφέρει σε προηγούμενο κεφάλαιο αποτελεί την απλούστερη μέθοδο αναζήτησης υπερπαραμέτρων.

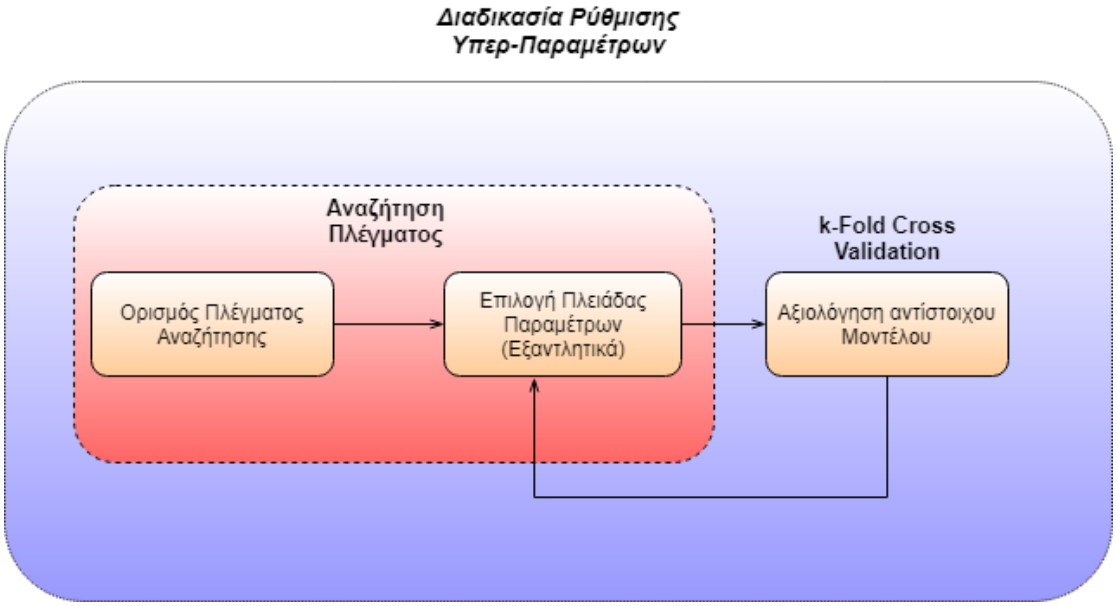
4.2.1 Ρύθμιση Τυχαίων Δασών και Μηχανών Διανυσμάτων Υποστήριξης.

Στη παραπάνω διαδικασία καίρια σημεία είναι τόσο η επιλογή των υπερπαραμέτρων που θα ρυθμιστούν από τον αλγόριθμο αναζήτησης όσο και το εύρος αυτών. Μια παραπάνω παράμετρος μπορεί να επιβραδύνει κατά πολύ τη διαδικασία, χωρίς όμως να επηρεάζει σε σημαντικό βαθμό θετικά την απόδοση του εκάστοτε μοντέλου ,συνεπώς η επιλογή των υπό ρύθμιση παραμέτρων και το αντίστοιχο εύρος πρέπει να γίνεται με σύνεση.

Ωστόσο εξίσου σημαντικό ρόλο έχει και η επιλογή του τρόπου με τον οποίο ένα μοντέλο θα αξιολογείται, καθώς θα επηρεάσει σε μεγάλο βαθμό την επιλογή των τελικών παραμέτρων.

Ο αλγόριθμος αξιολόγησης των παραγόμενων μοντέλων (από τη αναζήτηση πλέγματος) που χρησιμοποιήθηκε στη υλοποίηση μας είναι η μέθοδος k-fold Cross Validation.

Παρακάτω παρουσιάζεται σχηματικά η διαδικασία ρύθμισης υπερπαραμέτρων που ακολουθήθηκε για τα μοντέλα Τυχαία Δάση, Μηχανές Διανυσμάτων Υποστήριξης



Εικόνα 4-2 Ρύθμιση παραμέτρων για Τυχαία Δάση και ΜΔΥ

K-Fold Cross Validation

Η μέθοδος k-Fold Cross Validation είναι μια διαδικασία επαναλαμβανόμενης δειγματοληψίας η οποία χρησιμοποιείται για την αξιολόγηση μοντέλων μηχανικής μάθησης. Η παράμετρος k , αναφέρεται στον αριθμό των ομάδων στις οποίες πρόκειται να χωριστεί το δείγμα – σύνολο δεδομένων πάνω στο οποίο θα αξιολογηθεί το εκάστοτε μοντέλο.

Παρακάτω παρουσιάζεται αναλυτικά σε μορφή ψευδοκώδικα ο συγκεκριμένος αλγόριθμος:

1. *Ανακατάταξε τυχαία το εν λόγω σύνολο δεδομένων*

2. *Διαχώρισε το σύνολο δεδομένων σε k ξένα υποσύνολα.*

3. *Για κάθε ένα υποσύνολο:*

a. *Θεώρησε το συγκεκριμένο υποσύνολο ως το σύνολο δοκιμής.*

b. *Συνένωσε τα υπόλοιπα υποσύνολα σε ένα ενιαίο σύνολο εκπαίδευσης.*

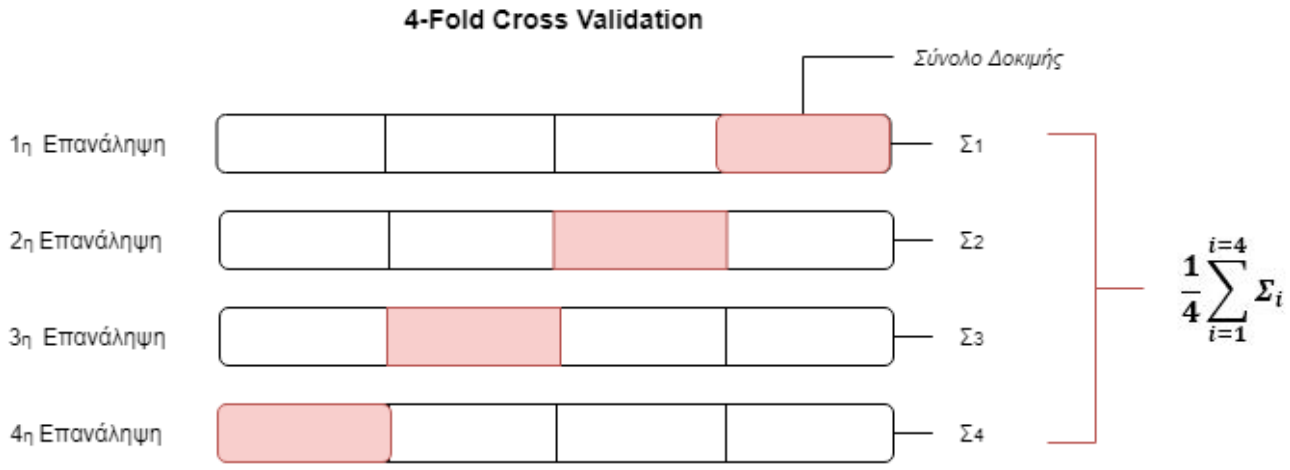
c. *Εκπαίδευσε το μοντέλο στο σύνολο εκπαίδευσης.*

d. *Αξιολόγησε το χρησιμοποιώντας το παραπάνω σύνολο δοκιμής (χρησιμοποιώντας ένα μια συνάρτηση σφάλματος).*

e. *Αποθήκευσε το σφάλμα και απέρριψε το μοντέλο .*

f. *Επανέλαβε.*

4. *Συγκέντρωσε το τελικό σφάλμα του μοντέλου λαμβάνοντας υπόψιν την απόδοση του στη κάθε μια επανάληψη. Η συνηθέστερη μετρική είναι η μέση τιμή των σφαλμάτων του μοντέλου σε κάθε επανάληψη.*



Εικόνα 4-3 4-Fold Cross Validation

Στην περίπτωση μας η μέθοδος k-Fold Cross Validation χρησιμοποιείται κατά τη διαδικασία ρύθμισης των υπερπαραμέτρων των εν λόγω μοντέλων. Συνεπώς το αρχικό σύνολο δεδομένων χωρίζεται τυχαία σε σύνολο εκπαίδευσης και σύνολο δοκιμής. Το σύνολο εκπαίδευσης είναι αυτό που θα χρησιμοποιηθεί από τη παραπάνω μέθοδο για τη επιλογή των παραμέτρων και προφανώς και για την εκπαίδευση του τελικού «επιλεγμένου μοντέλου».

Η τεχνική k-Fold Cross Validation ,αποτελεί μια αμερόληπτη μέθοδος αξιολόγησης μοντέλων Μηχανικής Μάθησης η οποία μας βοηθάει να

αποφύγουμε την πιθανή μεροληπτική επιλογή που μπορεί να προκύψει από τους τυχαίους διαχωρισμούς του αρχικού συνόλου σε σύνολο εκπαίδευσης και σύνολο δοκιμής.

Αξίζει επίσης να σημειωθεί πως το σύνολο δεδομένων που θα χρησιμοποιηθεί για τη διαδικασία ρύθμισης των υπερπαραμέτρων πρέπει να είναι ξένο από το τελικό σύνολο δοκιμής που θα χρησιμοποιηθεί για τη αξιολόγηση της βέλτιστης αρχιτεκτονικής ,ώστε διατηρηθεί η αμεροληψία της επιλογής.

4.2.2 Σχεδιασμός και Υλοποίηση Τυχαίων Δασών

Για τη δημιουργία των Τυχαίων Δασών χρησιμοποιήθηκε η συνάρτηση της βιβλιοθήκης scikit-learn **`RandomForestRegressor().`** Η εκπαίδευση των Τυχαίων Δασών έγινε χρησιμοποιώντας μια βελτιστοποιημένη έκδοση του αλγόριθμου CART που έχουμε αναλυτικά παρουσιάσει σε προηγούμενο κεφάλαιο.

Η βιβλιοθήκη scikit-learn μας δίνει τη δυνατότητα να ρυθμίσουμε ένα μεγάλο πλήθος παραμέτρων σχετικά με την αρχιτεκτονική και την εκπαίδευση των τυχαίων δασών όπως :

- *Το πλήθος των Δέντρων – Εκτιμητών του δάσους*
- *Το μέγιστο βάθος του κάθε Δέντρου*
- *Τον μέγιστο αριθμό φύλλων των Δέντρων Απόφασης*
- *Τον βαθμό τυχαιότητας στη διαδικασία διαχωρισμού του συνόλου δεδομένων*
- *Τον αριθμό των παράλληλων εργασιών κατά τη διαδικασία της εκπαίδευσης και της πρόβλεψης*

Για τα πειράματα μας επιλέχθηκαν να ρυθμιστούν οι παράμετροι:

- **`n_estimators`**: Η παράμετρος αυτή ορίζει τον πλήθος των Δέντρων Απόφασης που θα αποτελέσουν το λεγόμενο Δάσος. Η παράμετρος αυτή είναι ιδιαίτερα σημαντική καθώς θα επηρεάσει σε μεγάλο βαθμό τη μεροληψία και τη διακύμανση του μοντέλου σε μεγάλο βαθμό. Ωστόσο μετά από κάποιο σημείο η αύξηση του αριθμού των Δέντρων-Εκτιμητών δεν επηρεάζει σε μεγάλο βαθμό το σφάλμα του Δάσους αλλά αυξάνει εκθετικά τον χρόνο εκπαίδευσης του μοντέλου, συνεπώς οι τιμές που εξετάστηκαν για τη συγκεκριμένη παράμετρο κατά τη διαδικασία

ρύθμισης είναι:

`n_estimators =` {10,100,200,…,900,1000}

- **`max_depth:`** Η παράμετρος αυτή ορίζει το μέγιστο επιτρεπτό βάθος ενός Δέντρου Απόφασης του Τυχαία Δάσους.

  Πιο συγκεκριμένα ,η παραπάνω παράμετρος ορίζει σε ένα βαθμό τον αριθμό των διαχωρισμών που θα πραγματοποιηθούν για το συγκεκριμένο Δέντρο. Ένας πολύ μεγάλος αριθμός διαχωρισμών οδηγεί σε Δέντρα υπερπροσαμοσμένα στο σύνολο εκπαίδευσης, με χαμηλή δυνατότητα γενίκευσης. Ένας πολύ χαμηλός αριθμός διαχωρισμών μπορεί να οδηγήσει σε πτωχή εκπαίδευση και να δημιουργήσει μεγάλη διακύμανση στο μοντέλο μας.

  Για τους παραπάνω λόγους κατά τη διαδικασία ρύθμισης πρέπει να αποφεύγονται οι ακραίες τιμές στο εύρος της συγκεκριμένης παραμέτρου.

- **min_samples_leaf:** Η συγκεκριμένη παράμετρος ορίζει τον μικρότερο επιτρεπόμενο αριθμό δειγμάτων ενός κόμβου «φύλλου» σε ένα Δέντρο Απόφασης. Περιορίζοντας το συγκεκριμένο αριθμό σε ένα συγκεκριμένο εύρος αποτρέπουμε την υπερπροσαρμογή του μοντέλου μας.

  Οι παράμετροι **min_samples_leaf** και **`max_depth`** είναι υπεύθυνοι για την ρύθμιση των κύριων συνιστωσών ενός Τυχαίου Δάσους που είναι τα Δέντρα Απόφασης που το αποτελούν. Παρότι δεν ορίζουν αυστηρούς κανόνες ρύθμισης θέτουν ωστόσο τα όρια σημαντικών αρχιτεκτονικών παραμέτρων ρυθμίζοντας έτσι έμμεσα και τη συμπεριφορά του τελικού μοντέλου.

4.2.3 Σχεδιασμός και Υλοποίηση Μηχανών Διανυσμάτων Υποστήριξης

Παρακάτω παρουσιάζεται η διαδικασία σχεδιασμού και εκπαίδευσης των Μηχανών Διανυσμάτων Υποστήριξης. Αρχικά παρουσιάζεται ο τρόπος με τον οποίο κανονικοποιήσαμε το σύνολο δεδομένων αλλά και διαισθητικές παρατηρήσεις για την σημασία της συγκεκριμένης διαδικασίας. Στη συνέχεια γίνεται αναλυτική παρουσίαση του αλγόριθμου ρύθμισης των ΜΔΥ.

4.2.3.1 Προεπεξεργασία και Κανονικοποίηση Δεδομένων

Η κλιμάκωση (Scaling) των δεδομένων εισόδου πριν από την εκπαίδευση των Μηχανών Διανυσμάτων Υποστήριξης αποτελεί διαδικασία καίριας σημασίας

για τη σωστή λειτουργία του τελικού μοντέλου. Όπως έχουμε ήδη αναφέρει στο κεφάλαιο των Mapping Models , το κύριο πλεονέκτημα της κλιμάκωσης , είναι η αποφυγή της επικράτησης των χαρακτηριστικών με μεγάλη αριθμητική κλίμακα σε βάρος αυτών με μικρότερο αριθμητικό εύρος.

Ωστόσο η κλιμάκωση στις Μηχανές Διανυσμάτων Υποστήριξης είναι επιτακτική για την απλοποίηση των μαθηματικών υπολογισμών που προκύπτουν από τη χρήση των συναρτήσεων πυρήνα. Οι τιμές των συναρτήσεων Πυρήνα εξαρτώνται κατά κύριο λόγο από τα εσωτερικά γινόμενα των διανυσμάτων των διαφόρων χαρακτηριστικών του συνόλου δεδομένων. Οι πολύ υψηλές τιμές των μεταβλητών εισόδου μπορεί να προκαλέσουν αριθμητικά προβλήματα.

Για τους παραπάνω λόγους στην υλοποίηση μας προχωρήσαμε σε κλιμάκωση των χαρακτηριστικών στο εύρος [0,1] χρησιμοποιώντας τη κλιμάκωση Μεγίστου-Ελαχίστου.

Το παραπάνω πραγματοποιήθηκε χρησιμοποιώντας τη συνάρτηση **MinMaxScaler()** της βιβλιοθήκης scikit-learn() και μόνο για τα χαρακτηριστικά εισόδου.

```
min_max_scaler = MinMaxScaler()
scaled_inputs  = min_max_scaler.fit_transform(inputs)
```

4.2.3.2 Σχεδιασμός και Υλοποίηση

Για την δημιουργία μοντέλων στη περίπτωση των Μηχανών Διανυσμάτων Υποστήριξης χρησιμοποίησα τη συνάρτηση **`SVR()`** της βιβλιοθήκης scikit-learn.

Λόγω της μη γραμμικότητας και της πολυπλοκότητας του συνόλου δεδομένων των πειραμάτων ,χρησιμοποιήθηκε μια μη γραμμική συνάρτηση Πυρήνα και συγκεκριμένα η Γκαουσιανή Συνάρτηση RBF (Radial Basis Function)

`SVR(kernel ='rbf')`

Στις περισσότερες περιπτώσεις που δε γνωρίζουμε τη δομή και τις συσχετίσεις των χαρακτηριστικών στο σύνολα δεδομένων, ο πυρήνας RBF αποτελεί τη μια λογική πρώτη επιλογή.

Η συγκεκριμένη συνάρτηση πυρήνα απεικονίζει ,μη γραμμικά , τα δείγματα εισόδου σε ένα χώρο περισσότερων διαστάσεων ώστε να μπορεί να διαχειριστεί μη γραμμικές συσχετίσεις μεταξύ των χαρακτηριστικών εισόδου και εξόδου.

Ο Πυρήνας RBF χρειάζεται σαν παραμέτρους εισόδου το **C** και το **gamma**. Η επιλογή των συγκεκριμένων υπερπαραμέτρων είναι πολύ σημαντική , καθώς ρυθμίζουν σε μεγάλο βαθμό την ικανότητα γενίκευσης και κατά συνέπεια την απόδοση του τελικού μοντέλου.

Διαισθητικά η παράμετρος **gamma** καθορίζει πόσο μακριά θα φτάνει η επιρροή ενός στιγμιότυπου εκπαίδευσης. Πιο συγκεκριμένα, έστω K η συνάρτηση πυρήνα τότε ισχύει

$$K(x, x') = \exp(-gamma * \|x - x'\|^2)$$

Όπως μπορούμε να διακρίνουμε από τη παραπάνω εξίσωση, η παράμετρος **gamma** είναι η ελεύθερη παράμετρος της Γκαουσιανής Συνάρτησης RBF **.** Μια μικρή **gamma** σημαίνει πως η Γκαυσιανή επιφάνεια θα είναι επιφάνεια με μεγάλη διακύμανση, έτσι, έστω πως το x' αποτελεί διάνυσμα υποστήριξης ,αν η τιμή του **gamma** είναι μικρή αυτό θα σημαίνει πως η κλάση του συγκεκριμένου διανύσματος θα επηρεάζει τη κλάση του διανύσματος x ακόμα και αν η απόσταση μεταξύ τους είναι μεγάλη **.**

Όσο μεγαλύτερη είναι η παράμετρος **gamma** τόσο στενότερη θα είναι η λεγόμενη Γκαουσιανή «καμπάνα» συνεπώς η διακύμανση θα είναι μικρή και για αυτό το λόγο τα διανύσματα υποστήριξης δε θα έχουν εκτεταμένη επίδραση στη κατηγοριοποίηση άλλων διανυσμάτων εισόδου. .

Η παράμετρος **C** καθορίζει το κατά πόσο επιτρεπτή είναι η λανθασμένη κατηγοριοποίηση των δεδομένων παίρνοντας ως αντάλλαγμα μια πιο απλή κα λεία επιφάνεια διαχωρισμού. Μια χαμηλή τιμής της παραμέτρου **C** δημιουργεί μια ομαλή επιφάνεια διαχωρισμού με δυνατότητες γενίκευσης ,αντίθετα μια υψηλή τιμή στοχεύει στην ορθή ταξινόμηση όλων των διανυσμάτων εισόδου δίνοντας την ελευθερία στο μοντέλο να επιλέξει περισσότερα διανύσματα υποστήριξης ενισχύοντας ωστόσο το φαινόμενο της υπερπροσαρμογής στο σύνολο εκπαίδευσης.

Σε αυτό το σημείο αξίζει να σημειωθεί πως η συμπεριφορά του μοντέλου είναι πολύ ευαίσθητη στην τιμή της παραμέτρου **gamma.** Αν η συγκριμένη παράμετρος ρυθμιστεί σε μεγάλη τιμή τότε το εύρος επιρροής των διανυσμάτων υποστήριξης περιλαμβάνει μόνο τα ίδια τα διανύσματα υποστήριξης και δεν υπάρχει τρόπος να ρυθμιστεί το τελικό μοντέλο με χρήση της παραμέτρου **C**. Σε αυτή τη περίπτωση η υπερπροσαρμογή στο σύνολο εκπαίδευσης είναι αναπόφευκτη.

Αντίστοιχα, όταν η παράμετρος **gamma** έχει πολύ μικρή τιμή , το μοντέλο θα είναι πολύ περιορισμένο και δε θα μπορεί να καταγράψει την πολυπλοκότητα και τις συσχετίσεις που κρύβονται στα δεδομένα. Η περιοχή επιρροής κάθε διανύσματος υποστήριξης θα περιλαμβάνει όλο το σύνολο δεδομένων εκπαίδευσης και συνεπώς το τελικό μοντέλο θα συμπεριφέρεται παρόμοια με ένα γραμμικό μοντέλο, μη αξιοποιώντας τις δυνατότητες που προσφέρει η μέθοδος της συνάρτησης Πυρήνα (Kernel Trick).

Συνεπώς λαμβάνοντας τα παραπάνω υπόψιν, η επιλογή των υπερπαραμέτρων για το μοντέλο των Μηχανών Διανυσμάτων Υποστήριξης πραγματοποιήθηκε με σύνεση και ιδιαίτερη προσοχή στο εύρος τιμών της παραμέτρου **gamma.**

Παρακάτω παρουσιάζεται ο κώδικας δημιουργίας του πλέγματος αναζήτησης των παραπάνω υπερπαραμέτρων.

```
Cs = 10. ** np.arange(-3, 8)

gammas= 10. ** np.arange(-5, 4)

param_grid = {'C': Cs, 'gamma': gammas}
```

Δηλαδή έχουμε $\mathbf{Cs} = \{10^{-3}, \dots, 10^{8}\}$ και $\boldsymbol{gammas} = \{10^{-5}, \dots, 10^{4}\}$

5 Εκπαίδευση Μοντέλων και Παρουσίαση Αποτελεσμάτων

Όπως έχει ήδη αναφερθεί, κατά την επιλογή ενός παρόχου Υποδομής ως Υπηρεσία (IaaS provider), για την εγκατάσταση και λειτουργία μιας εφαρμογής, ο κάτοχος της εφαρμογής και υιοθετών της εκάστοτε υποδομής ,θα πρέπει να πάρει μια πολύ σημαντική απόφαση η οποία σχετίζεται με το πλήθος και το είδος των υπολογιστικών πόρων που θα δεσμευτούν έτσι ώστε να διασφαλιστεί η ομαλή λειτουργία της εφαρμογής.

Όπως είναι γνωστό η παραπάνω δεν αποτελεί μια απλή απόφαση, καθώς οι υπολογιστικοί πόροι που θα δεσμευτούν, σε συνδυασμό με τον υπολογιστικό φόρτο της εφαρμογής σε διάφορα χρονικά σημεία αποτελούν καθοριστικούς παράγοντες που επηρεάζουν πολλούς βασικούς δείκτες επίδοσης όπως ο χρόνος απόκρισης της εφαρμογής σε αναμενόμενα αιτήματα.

Η πρόταση της συγκεκριμένης διπλωματικής, είναι χρήση της Μηχανικής Μάθησης για την πρόβλεψη της ποιότητας υπηρεσίας της εφαρμογής δοθέντων των διαθέσιμων υπολογιστικών πόρων και του τρέχοντος υπολογιστικού φορτίου. Προφανώς το παραπάνω κάνει φανερή την αίτηση για μια προεργασία πριν από τη επιλογή των πόρων η οποία περιλαμβάνει την δημιουργία ενός επαρκούς ,τόσο ποσοτικά όσο και ποιοτικά, συνόλου δεδομένων για την εκπαίδευση των διάφορων μοντέλων Μηχανικής Μάθησης ώστε να καθίσταται δυνατή η σωστή πρόβλεψη που θα αφορά το είδος και το πλήθος των υπολογιστικών πόρων a priori της εγκατάστασης της εφαρμογής.

Για την επικύρωση της παραπάνω άποψης χρησιμοποιήσαμε ένα σύνολο δεδομένων το οποίο προέκυψε από διαδοχικές εκτελέσεις μιας εμπορικής εφαρμογής Διαχείρισης Πελατειακών Σχέσεων (**C**ustomer **R**elationship **M**anagement) της εταιρίας Cognity A.E. Η συγκεκριμένη εφαρμογή ,αρχικά αναπτυγμένη σε μια εμπορική υποδομή εντός της εγκατάστασης (On-Premise) , μεταφέρεται στο στρώμα Υποδομής ως Υπηρεσία (IaaS) και εγκαθίσταται εν μέρει ή ολοκληρωτικά στο Υπολογιστικό Νέφος.

Στη συνέχεια, χρησιμοποιήθηκαν τρεις από τις επικρατέστερες τεχνολογίες για την επίλυση προβλημάτων στο τομέα της ***Ανάλυσης Παλινδρόμησης*** ,η οποία όπως έχουμε ήδη αναφέρει ανήκει στο ευρύτερο πεδίο της ***Μηχανικής Μάθησης,*** τα Νευρωνικά Δίκτυα, οι Μηχανές Διανυσμάτων Υποστήριξης και τα Τυχαία Δάση. Η διαδικασία ανάπτυξης και δημιουργίας του συνόλου δεδομένων, όπως αυτή παραγοντοποιήθηκε από την εταιρία Cognity A.E. ,αλλά και των πειραματικών αποτελεσμάτων παρουσιάζονται παρακάτω.

5.1 Το Σύνολο Δεδομένων – Εφαρμογής ΔΠΣ

Η παραπάνω εφαρμογή ΔΠΣ (Διαχείρισης Πελατειακών Σχέσεων) ακολουθεί τον αρχιτεκτονικό σχεδιασμό των Microservices ,δηλαδή η λογική της εφαρμογής είναι διασπασμένη σε αυτόνομες υπηρεσίες κάθε μια από τις οποίες είναι υπεύθυνη για την εκτέλεση μιας συγκεκριμένης λειτουργίας. Στη περίπτωση μας έχουμε τις εξής αυτόνομες υπηρεσίες:

- *Την βάση δεδομένων*
- *Τη γραφική διεπαφή χρήστη*
- *Την Υπηρεσία ανακάλυψης (Eureka)*
- *Τέσσερις διαφορετικές υπηρεσίες Χρήστες, Προϊόντα, Πελάτες Παραγγελίες (Users,Products,Customers,Order) οι οποίες χρησιμοποιούνται από τη γραφική διεπαφή για την εκτέλεση των διάφορων αιτημάτων χρήστη.*

Ο κύριος δείκτης απόδοσης για τη συγκεκριμένη εφαρμογή, καθώς και αυτός που χρησιμοποιήθηκε ως παράμετρος ποιότητας υπηρεσίας (QoS parameter) για τα πειράματα μας είναι ο χρόνος απόκρισης στο αίτημα του χρήστη. Ωστόσο ο χρήστης αλληλοεπιδρά με την γραφική διεπαφή η οποία με της σειρά της δημιουργεί τα αντίστοιχα αιτήματα στην εκάστοτε υπηρεσία.

Διαφορετικά φορτία εργασίας και διαφορετικοί υπολογιστικοί πόροι επηρεάζουν κυρίως την απόδοση κάθε μιας από τις παρακάτω υπηρεσίες: Χρήστες, Προϊόντα, Πελάτες Παραγγελίες.

Κάθε μια από αυτές αποτελεί διαφορετικό λογισμικό, συνεπώς θα συμπεριφέρονται διαφορετικά ως προς την αλλαγή στο φόρτο εργασίας ή το πλήθος και είδος των υπολογιστικών πόρων.

Για τους παραπάνω λόγους το τελικό σύνολο δεδομένων χωρίστηκε σε τέσσερα ξένα υποσύνολα δεδομένων, ένα για κάθε διαφορετική υπηρεσία, τα οποία χρησιμοποιήθηκαν ξεχωριστά για την εκπαίδευση των μοντέλων Μηχανικής Μάθησης. Στη περίπτωση που ένα αίτημα χρησιμοποιεί παραπάνω από μια υπηρεσίες διαδοχικά, τότε οι προβλέψεις των μοντέλων μπορούν κλιμακωτά να παρέχουν ενόραση για το τελικό αθροιστικό χρόνο απόκρισης της εφαρμογής

Για τη δημιουργία του συνόλου δεδομένων, κάθε μία από τις παραπάνω τέσσερις

υπηρεσίες εγκαταστάθηκε σε ένα ξεχωριστό Docker Container τα οποία με τη σειρά τους τροφοδοτήθηκαν από την ίδια εικονική μηχανή (Host Virtual Machine). Στη συνέχεια ένα ειδικά κατασκευασμένο λογισμικό της εταιρίας Cognity A.E δημιούργησε εικονική κίνηση σε κάθε μια από αυτές τις υπηρεσίες ώστε να συγκεντρωθούν οι απαραίτητες εγγραφές που θα σχημάτιζαν το τελικό σύνολο δεδομένων.

Όπως έχει αναφερθεί παραπάνω, σκοπός των πειραμάτων είναι η αντιστοίχιση του φόρτου εργασίας και των υπολογιστικών πόρων ,σε μια μετρική ποιότητας υπηρεσίας , που στη συγκεκριμένη περίπτωση είναι **ο χρόνος απόκρισης στο αίτημα κάθε χρήστη**.

Ως υπολογιστικός πόρος χρησιμοποιήθηκε **η μνήμη RAM** που έχει διοχετευθεί στο αντίστοιχο Docker Container που είναι εγκατεστημένη η κάθε υπηρεσία. Ενώ ως παράμετροι φόρτου εργασίας ορίστηκαν για κάθε υπηρεσία ξεχωριστά:

- *Το πλήθος των εικονικών χρηστών* της εφαρμογής που δημιουργήθηκαν κατά τη διάρκεια της εκτέλεσης
- *Το πλήθος των συνολικών αιτημάτων* που πραγματοποιήθηκαν σε όλες τις υπηρεσίες.
- *Το πλήθος των αιτημάτων που εξυπηρετήθηκαν* από τη συγκεκριμένη υπηρεσία.
- *Τα αιτήματα ανά δευτερόλεπτο* που προωθήθηκαν προς την υπηρεσία

Συγκεντρωτικά το συνολικό παραχθέν σύνολο δεδομένων για όλες τις υπηρεσίες μαζί ,που μας παραχωρήθηκε είχε πεδία της μορφής:

- **Όνομα Υπηρεσίας** - (Service Component)
- **Μνήμη Ram (Docker Ram)** – (RAM)
  [Η οποία αποτελεί τον μεταβαλλόμενο υπολογιστικό πόρο (Hardware Parameters)]
- **Το πλήθος των εικονικών χρηστών** (#Users)
- **Το πλήθος των συνολικών αιτημάτων** (#Global Requests)
- **Το πλήθος των αιτημάτων ανά υπηρεσία** (#Request)
- **Τα αιτήματα ανά δευτερόλεπτο που προωθήθηκαν προς την υπηρεσία** (#RPS)
  [Οι οποίες αποτελούν τις παραμέτρους του φόρτου εργασίας (Workload Parameters)]
- **Μέσος χρόνος απόκρισης αιτήματος ανά υπηρεσία** –(Latency AVG)
  [Η οποία μετρική αποτελεί την παράμετρο ποιότητας υπηρεσίας]

Όπως γίνεται αντιληπτό από την παραπάνω ανάλυση, κατά την εκπαίδευση των μοντέλων η μεταβλητή εξόδου-πρόβλεψης θα είναι ο Μέσος χρόνος απόκρισης αιτήματος ανά υπηρεσία , ενώ η Μνήμη RAM αλλά και οι παράμετροι του φόρτου εργασίας της κάθε υπηρεσίας, αποτελούν τις μεταβλητές εισόδου.

5.2 Αξιολόγηση Μοντέλων και Παρουσίαση Αποτελεσμάτων

Η εκτέλεση των πειραμάτων αποτελείται από δύο ξεχωριστές αλλά αλληλένδετες διαδικασίες:

- *Την ρύθμιση των παραμέτρων των μοντέλων Μηχανικής Μάθησης και για τα τέσσερα διαφορετικά σύνολα δεδομένων (ένα για κάθε διαδικασία)*
- *Την εκπαίδευση και αξιολόγηση των βέλτιστα ρυθμισμένων μοντέλων*.

Τα πειραματικά αποτελέσματα των παραπάνω διαδικασιών παρουσιάζονται αναλυτικά παρακάτω.

5.2.1 Αποτελέσματα Ρύθμισης Παραμέτρων

Όπως έχουμε αναφέρει, οι τρεις αλγόριθμοι Μηχανικής Μάθησης, που χρησιμοποιήθηκαν για την επίλυση του συγκεκριμένου προβλήματος Ανάλυσης Παλινδρόμησης αλλά και για την εκτέλεση των πειραμάτων (Συγκριτική Μελέτη) , είναι :

- ***Τα Τεχνητά Νευρωνικά Δίκτυα***
- ***Οι Μηχανές Διανυσμάτων Υποστήριξης***
- ***Τα Τυχαία Δάση***

Σε κάθε έναν από τους παραπάνω αλγόριθμους τροφοδοτήσαμε τα τέσσερα διαφορετικά σύνολα δεδομένων (ένα για κάθε υπηρεσία) και χρησιμοποιώντας την εκάστοτε τεχνική ρύθμισης παραμέτρων καταλήξαμε στις τελικές πλειάδες υπερπαραμέτρων για κάθε ξεχωριστό σύνολο.

Υπενθυμίζουμε σε αυτό το σημείο πως στη περίπτωση των Νευρωνικών Δικτύων χρησιμοποιήθηκε μια εξελικτική μέθοδος για την ρύθμιση του μοντέλου και συγκεκριμένα ένας *Γενετικός Αλγόριθμος.*

Στη περίπτωση των Τυχαίων Δασών και των Μηχανών Διανυσμάτων Υποστήριξης, η ρύθμιση των υπερπαραμέτρων πραγματοποιήθηκε, όπως έχουμε αναλύσει σε προηγούμενο κεφάλαιο, με τη χρήση της *Αναζήτησης Πλέγματος*.

Παρακάτω παρουσιάζονται αναλυτικά τα αποτελέσματα της διαδικασίας ρύθμισης παραμέτρων για κάθε αλγόριθμο και κάθε σύνολο δεδομένων:

- **Αλγόριθμος**: *Τεχνητά Νευρωνικά Δίκτυα*
  Μέθοδος Ρύθμισης Παραμέτρων: *Γενετικός Αλγόριθμος*

Επιλεγμένοι Υπερπαράμετροι ΤΝΔ			
Υπηρεσία	**Αριθμός Επιπέδων**	**Νευρώνες ανά Επίπεδο**	**Συνάρτηση Μεταφοράς ανά επίπεδο**
Παραγγελίες	4	5-3-2-1	tansig-logsig-logsig-purelin
Πελάτες	3	5-3-1	tansig-logsig-purelin
Προϊόντα	3	5-3-1	tansig-logsig-purelin
Χρήστες,	3	5-3-1	tansig-tansig-purelin

Πίνακας 1: Υπερπαράμετροι ΤΝΔ

- **Αλγόριθμος**: *Μηχανές Διανυσμάτων Υποστήριξης*
  Μέθοδος Ρύθμισης Παραμέτρων: *Αναζήτηση Πλέγματος*

Επιλεγμένοι Υπερπαράμετροι ΜΔΥ		
Υπηρεσία	**gamma**	**C**
Παραγγελίες	100	1000
Πελάτες	100	1000
Προϊόντα	100	1000
Χρήστες,	100	100

Πίνακας 2: Υπερπαράμετροι ΜΔΥ

- **Αλγόριθμος**: *Τυχαία Δάση*
  Μέθοδος Ρύθμισης Παραμέτρων: *Αναζήτηση Πλέγματος*

Επιλεγμένοι Υπερπαράμετροι ΤΔ			
Υπηρεσία	**Πλήθος Δέντρων Απόφασης**	**Μέγιστο βάθος Δέντρου Απόφασης**	**Ελάχιστο Πλήθος Δειγμάτων στα φύλλα**
Παραγγελίες	100	21	11
Πελάτες	100	41	9
Προϊόντα	100	91	11
Χρήστες,	100	11	1

Πίνακας 3: Υπερπαράμετροι ΤΔ

5.2.2 Αξιολόγηση Ρυθμισμένων Μοντέλων

Ύστερα από τη διαδικασία ρύθμισης των παραμέτρων ,έπονται οι διαδικασίες της δημιουργίας, εκπαίδευσης και ρύθμισης των βέλτιστα ρυθμισμένων μοντέλων.

Πιο συγκεκριμένα, για κάθε ένα από τα τέσσερα σύνολα δεδομένων χωρίσαμε με τυχαίο τρόπο τα δεδομένα στο σύνολο εκπαίδευσης (Training Set 65%) και σύνολο δοκιμής (Test Set 35%) χρησιμοποιώντας τη παρακάτω εντολή της βιβλιοθήκης scikit-learn:

```
X_train, X_test, y_train, y_test = train_test_split(inputs,
output, test_size=0.35, random_state=42)
```

Όπως έχουμε αναλύσει παραπάνω οι μεταβλητές εξόδου στο σύνολο δεδομένων θα είναι:

- *Όνομα Υπηρεσίας - (Service Component)*
- *Μνήμη Ram (Docker Ram) – (RAM)*
- *Το πλήθος των εικονικών χρηστών (#Users)*
- *Το πλήθος των συνολικών αιτημάτων (#Global Requests)*
- *Το πλήθος των αιτημάτων ανά υπηρεσία (#Request)*
- *Τα αιτήματα ανά δευτερόλεπτο που προωθήθηκαν προς την υπηρεσία (#RPS)*

Ενώ η μεταβλητή εξόδου δηλαδή το μέγεθος που θα προβλεφθεί από τα μοντέλα

είναι:

- *Μέσος χρόνος απόκρισης αιτήματος ανά υπηρεσία –(Latency AVG)*

Ύστερα από την δημιουργία των βέλτιστα ρυθμισμένων μοντέλων και χρησιμοποιώντας το τελικό σύνολο εκπαίδευσης , τα μοντέλα αξιολογήθηκαν ως προς την ικανότητα πρόβλεψης και γενίκευσης χρησιμοποιώντας το τελικό σύνολο δοκιμής.

Η μετρική που χρησιμοποιήθηκε για την αξιολόγηση των προβλέψεων των μοντέλων είναι:

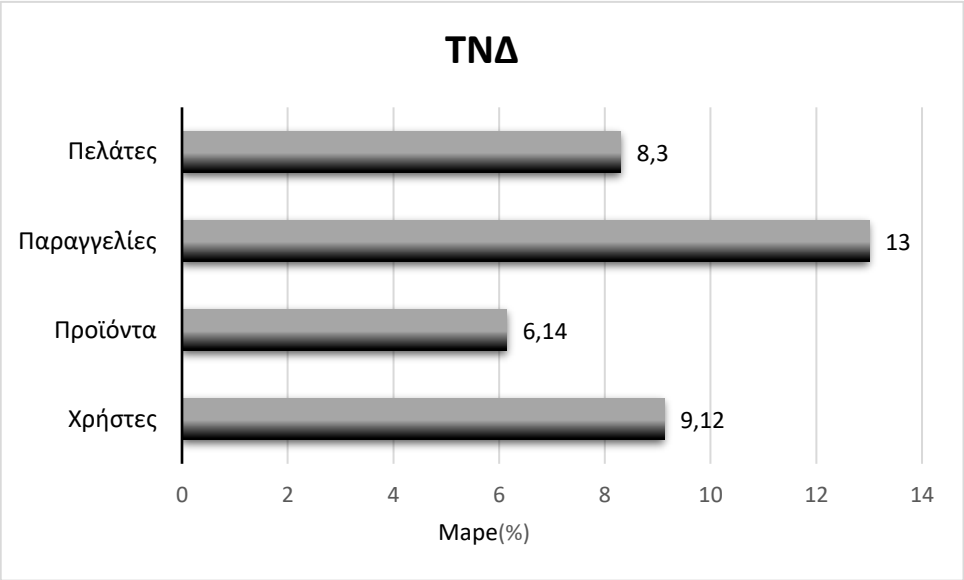
Η **Μέση Απόλυτη Εκατοστιαία Απόκλιση** (MAEA) ή αλλιώς το **Ποσοστιαίο Μέσο Απόλυτο Σφάλμα** (Mean Absolute Percentage Error MAPE). Ο τύπος για το συγκεκριμένη μετρική σφάλματος είναι:

$$\frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

, όπου A_t η πραγματική τιμή της μεταβλητής εξόδου , F_t η *προβλεφθείσα* τιμή της μεταβλητής εξόδου και n το πλήθος των εγγραφών του συνόλου δοκιμής.

Παρακάτω παρουσιάζονται σε διαγραμματική κορφή τα αποτελέσματα των πειραμάτων για κάθε μια από τις υπηρεσίες Χρήστες, Προϊόντα, Πελάτες Παραγγελίες.

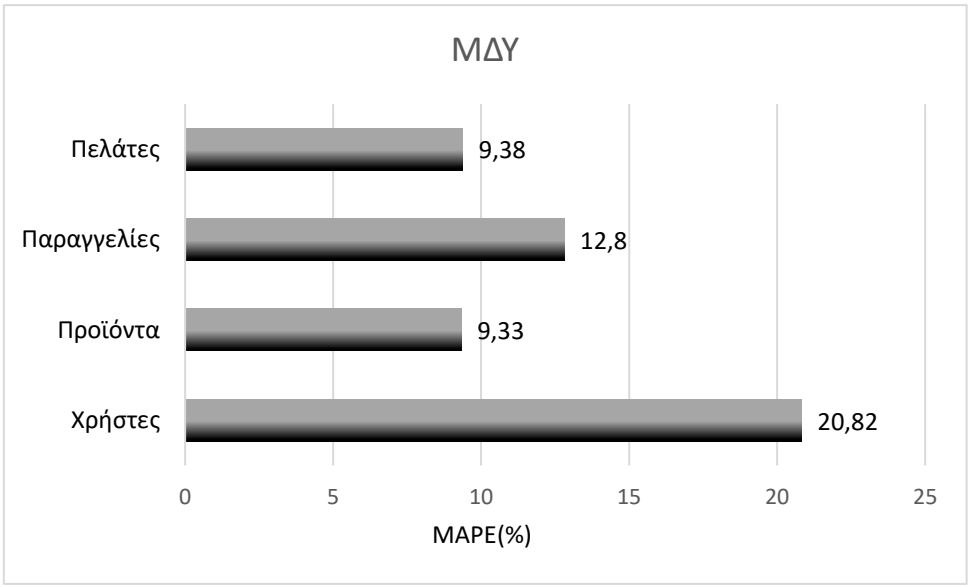
- **Αλγόριθμος**: *Τεχνητά Νευρωνικά Δίκτυα*
  Μέθοδος Ρύθμισης Παραμέτρων: *Γενετικός Αλγόριθμος*



Εικόνα 5-1:Αξιολόγηση Τεχνητών Νευρωνικών Δικτύων

Τις λεπτομέρειες για τις παραμέτρους των ΤΝΔ μπορείτε να τις αναζητήσετε στον Πίνακα 1

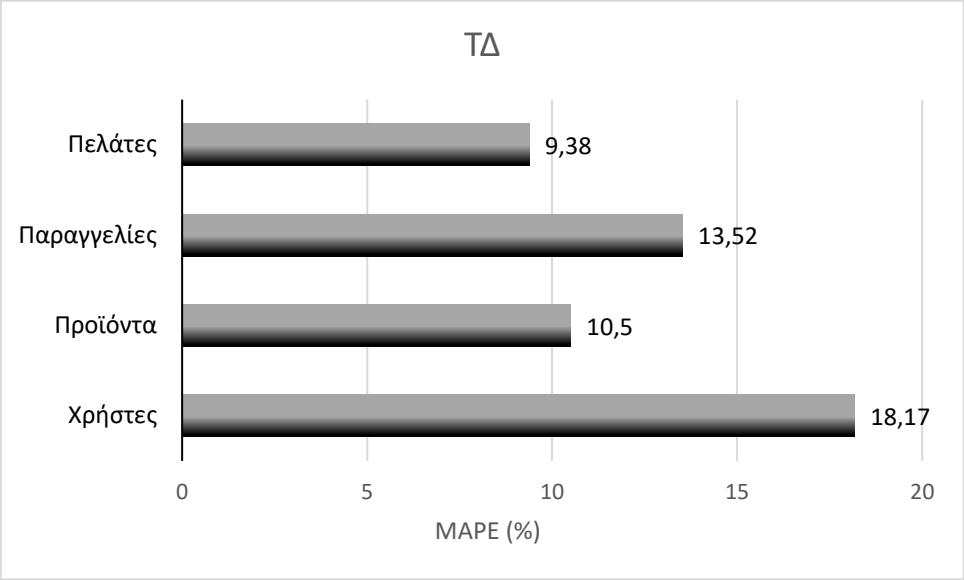
- **Αλγόριθμος**: *Μηχανές Διανυσμάτων Υποστήριξης*
  Μέθοδος Ρύθμισης Παραμέτρων: *Αναζήτηση Πλέγματος*



Εικόνα 5-2 Αξιολόγηση Μηχανών Διανυσμάτων Υποστήριξης

Τις λεπτομέρειες για τις παραμέτρους των ΤΝΔ μπορείτε να τις αναζητήσετε στον Πίνακα 2

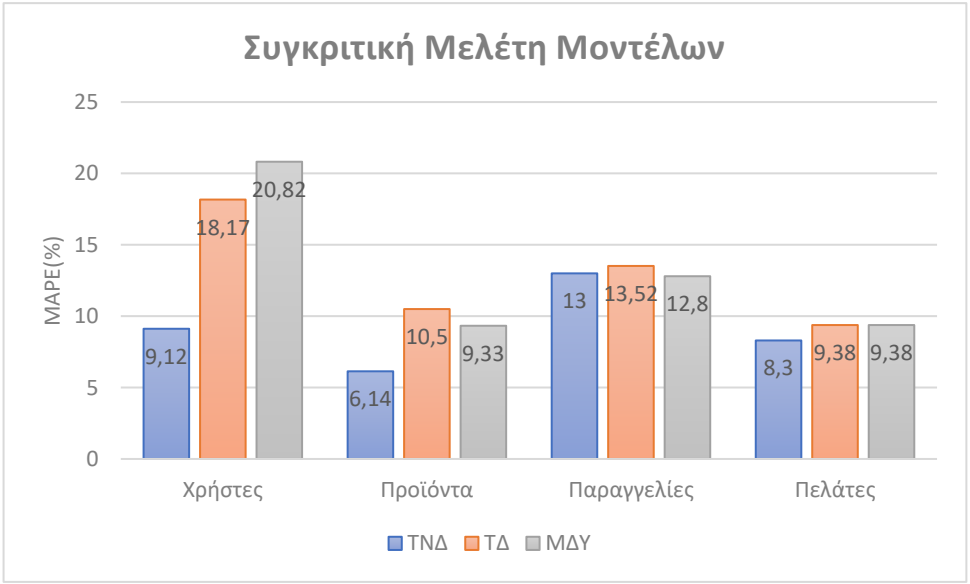
- **Αλγόριθμος**: *Τυχαία Δάση*
  Μέθοδος Ρύθμισης Παραμέτρων: *Αναζήτηση Πλέγματος*



Εικόνα 5-3:Αξιολόγηση Τυχαίων Δασών

Τις λεπτομέρειες για τις παραμέτρους των ΤΔ μπορείτε να τις αναζητήσετε στον Πίνακα 3

Παρακάτω παρουσιάζονται τα αποτελέσματα συγκεντρωτικά ώστε να διευκολυνθεί η συγκριτική μελέτη των τριών αλγορίθμων.



Εικόνα 5-4 Συγκριτική Μελέτη Μοντέλων

Όπως γίνεται φανερό από τα παραπάνω διαγράμματα, το σφάλμα και των τριών μοντέλων είναι σχετικά μικρό, γεγονός που φανερώνει πως πράγματι υπάρχουν συσχετίσεις μεταξύ των παραμέτρων : Φόρτου εργασίας, Υλισμικού και Ποιότητας Υπηρεσίας

Επίσης διακρίνεται πως τα Νευρωνικά Δίκτυα αποδίδουν καλύτερα και στα τέσσερα διαφορετικά σύνολα δεδομένων. Περισσότερη ανάλυση ακολουθεί στο επόμενο κεφάλαιο.

6 Επίλογος

Το παρόν Κεφάλαιο αποτελεί τον επίλογο της διπλωματικής εργασίας. Αρχικά παραθέτουμε τα συμπέρασμα στα οποία καταλήξαμε κατά τη διαδικασία εκτέλεσης των πειραμάτων αλλά και ανάλυσης των τελικών αποτελεσμάτων. Στη συνέχεια παρουσιάζουμε τις πιθανές επεκτάσεις που μπορούν να πραγματοποιηθούν τόσο σε επίπεδο συνόλου δεδομένων όσο και σε επίπεδο αλγόριθμων Μηχανικής Μάθησης.

6.1 Σύνοψη και Συμπεράσματα

Σκοπός της παρούσας διπλωματικής εργασίας ήταν, η ανάλυση των πιθανών συσχετίσεων που δημιουργούνται μεταξύ των παραμέτρων φόρτου εργασίας (Workload Parameters) ,των παραμέτρων υλισμικού ή αλλιώς υπολογιστικών πόρων (Hardware Parameters), και των παραμέτρων που καθορίζουν την ποιότητα υπηρεσίας μιας εφαρμογής, η οποία διατίθεται ως υπηρεσία σε Υπολογιστικό Νέφος (SaaS) .

Η σύνηθες τακτική ανεύρεσης των παραπάνω συσχετίσεων και παραμέτρων, βασίζεται στην εμπειρία και τη γνώση του εκάστοτε μηχανικού ή ομάδας μηχανικών ή ακόμα και στην μέθοδο Δοκιμής και Λάθους ώστε να βρεθούν οι κατάλληλοι συνδυασμοί παραμέτρων. Η παραπάνω διαδικασία είναι επίπονη και πολλές φορές καταλήγει σε λανθασμένες αποφάσεις.

Η πρόταση μας είναι η χρήση τεχνικών Μηχανικής Μάθησης ώστε να αποκτηθεί η παραπάνω ενόραση για τις πιθανές συσχετίσεις που χαρακτηρίζουν τις προαναφερθείσες παραμέτρους. Αυτό εκτιμήθηκε χρησιμοποιώντας ένα πραγματικό σύνολο δεδομένων μιας εμπορικής εφαρμογής Διαχείρισης Πελατειακών Σχέσεων το οποίο παράχθηκε δημιουργώντας την κατάλληλη εικονική κίνηση για μια πληθώρα συνδυασμών παραμέτρων υλισμικού και παραμέτρων φόρτου εργασίας.

Τα συμπεράσματα που καταλήγουμε από τα πειράματα μπορούν να παρουσιαστούν υπό το πρίσμα δύο διαφορετικών κατευθύνσεων.

Αρχικά παρατηρήσαμε ότι, και οι τρεις αλγόριθμοι που χρησιμοποιήσαμε πέτυχαν ικανοποιητική απόδοση δείχνοντας μας πως πράγματι, οι συσχετίσεις μεταξύ των παραμέτρων φόρτου εργασίας , υπολογιστικών πόρων και ποιότητας υπηρεσίας είναι παρούσες και ανιχνεύσιμες από τα μοντέλα μας. Η συγκεκριμένη παρατήρηση καθιστά τη Μηχανική Μάθηση ως μια ελκυστική προσέγγιση για την επίλυση του προβλήματος της αντιστοίχισης των παραπάνω παραμέτρων.

Η δεύτερη κατεύθυνση των συμπερασμάτων επικεντρώνεται στην συγκριτική μελέτη των αλγόριθμων Μηχανικής Μάθησης και των μεθόδων ρύθμισης παραμέτρων. Τα αποτελέσματα αξιολόγησης που παρουσιάστηκαν αναλυτικά στο προηγούμενο κεφάλαιο ανέδειξαν την αποτελεσματικότητα της λύσης των Νευρωνικών Δικτύων σε αντίθεση με τις άλλες τεχνικές. Η εφαρμογή του Γενετικού αλγορίθμου για την ρύθμιση των παραμέτρων των Νευρωνικών Δικτύων καθιστά τη εκπαίδευση και ρύθμιση της τρέχουσας λύσης λιγότερο αποδοτική, όσον αφορά τη χρονική πολυπλοκότητα, ωστόσο κατά τη διάρκεια του χρόνου εκτέλεσης δεν παρατηρείται σημαντική χρονική απόκλιση από τις άλλες προσεγγίσεις.

6.2 Μελλοντικές Επεκτάσεις

Η Μηχανική Μάθηση διαδραματίζει σήμερα σπουδαίο ρόλο σε πολλούς τομείς της επιστήμης και της τεχνολογίας. Είναι βέβαιο πως στο εγγύς μέλλον θα χρησιμοποιείται όλο και περισσότερο τόσο για την ανάπτυξη Τεχνητής Νοημοσύνης αλλά κυρίως για την επίλυση δύσκολων προβλημάτων που συναντώνται συχνά στη χώρο της βιομηχανίας και ων επιχειρήσεων.

Η γρήγορη ανάπτυξη και εξέλιξη των Νευρωνικών Δικτύων, χρησιμοποιώντας αρχιτεκτονικές πολλών νευρώνων και κρυφών επιπέδων, η λεγόμενη και βαθιά μάθηση ή Deep Learning, μας προϊδεάζει για το μέλλον της Μηχανικής Μάθησης και των Νευρωνικών Δικτύων ως πιθανού κυρίαρχου αλγορίθμου στο συγκεκριμένο τομέα.

Οι πιθανές μελλοντικές επεκτάσεις της παρούσας εργασίας είναι:

- Να συμπεριλάβουμε περισσότερες παραμέτρους υπολογιστικών πόρων όπως η ταχύτητα του επεξεργαστή ,ή η ταχύτητα των πιθανών συσκευών εισόδου εξόδου (σκληρός δίσκος κτλ.),ώστε να μελετηθεί περεταίρω η επιρροή των παραμέτρων υλισμικού στην απόκριση της κάθε εφαρμογής.

- Να προηγείται αποτελεσματική διαδικασία προεπεξεργασίας του συνόλου δεδομένων ώστε να απορρίπτονται παράμετροι που δεν επηρεάζουν τον χρόνο απόκρισης της εφαρμογής, βελτιώνοντας έτσι την απόδοση των χρησιμοποιούμενων μοντέλων.

- Η δημιουργία μεγαλύτερου συνόλου δεδομένων. Με αυτό τον τρόπο επιτυγχάνεται η καλύτερη απόκριση των αλγορίθμων εκπαίδευσης αλλά και βελτιώνεται η δυνατότητα γενίκευσης των τελικών μοντέλων.

- Η εκτέλεση πειραμάτων χρησιμοποιώντας περισσότερους αλγόριθμους

εκπαίδευσης αλλά και περισσότερους αλγόριθμους ρύθμισης παραμέτρων όπως η Μπεϋζιανή Βελτιστοποίηση. Αυτό θα μας δώσει τη δυνατότητα για την εκτέλεση αναλυτικότερης συγκριτικής μελέτης και πιθανών την εύρεση αποδοτικότερων μεθόδων πρόβλεψης για τη συγκεκριμένη εφαρμογή.

- Η συλλογή δεδομένων από νέες εφαρμογές και κατά συνέπεια η ανάλυση τους, οι οποίες εκθέτουν την λειτουργικότητα τους ως Λογισμικό ως υπηρεσία ή σκοπεύουν να μεταβούν στο Υπολογιστικό Νέφος στο εγγύς μέλλον.

Όλα τα παραπάνω θα μας βοηθήσουν να αποκτήσουμε μια πιο εμπεριστατωμένη άποψη για τις συσχετίσεις που ενυπάρχουν μεταξύ των παραμέτρων ποιότητας υπηρεσίας, φόρτου εργασίας και υπολογιστικών πόρων και πως γνωρίζοντας κάποιες από αυτές να μπορούμε να προβλέψουμε τις υπόλοιπες. Επίσης τα παραπάνω θα οδηγήσουν σε μια αναλυτική συγκριτική μελέτη μεταξύ αλγορίθμων και για εφαρμογές διαφορετικού σκοπού και αντικειμένου.

7 Βιβλιογραφία

1]. CHIKALOV, Igor. Algorithms for Decision Tree Construction. In: *Average Time Complexity of Decision Trees*. Springer, Berlin, Heidelberg, 2011. p. 61-78.

2]. ΣΤΑΛΈΝΤΣΗΣ, Βλαδίμηρος. Θεωρία δέντρων αποφάσεων και εφαρμογές. 2015.

3]. https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052

4]. QUINLAN, J.. Ross . Simplifying decision trees. *International journal of man-machine studies*, 1987, 27.3: 221-234.

5]. KAMIŃSKI, Bogumił; JAKUBCZYK, Michał; SZUFEL, Przemysław. A framework for sensitivity analysis of decision trees. *Central European journal of operations research*, 2018, 26.1: 135-159.

6]. ROKACH, Lior; MAIMON, Oded Z. *Data mining with decision trees: theory and applications*. World scientific, 2008.

7]. QUINLAN, J.. Ross . Induction of decision trees. *Machine learning*, 1986, 1.1: 81-106.

8]. BREIMAN, Leo. Bagging predictors. *Machine learning*, 1996, 24.2: 123-140.

9]. STROBL, Carolin; MALLEY, James; TUTZ, Gerhard. An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological methods*, 2009, 14.4: 323.

10]. HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H. The Elements of Statistical Learning, Springer, New York. 2001.

11]. JAMES, Gareth, et al. *An introduction to statistical learning*. New York: springer, 2013.

12]. BARANDIARAN, Iñigo. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 1998, 20.8.

13]. BREIMAN, Leo. Random forests. *Machine learning*, 2001, 45.1: 5-32.

14]. ARLOT, Sylvain; GENUER, Robin. Analysis of purely random forests bias. *arXiv preprint arXiv:1407.3939*, 2014.

15]. CORTES, Corinna; VAPNIK, Vladimir. Support-vector networks. *Machine learning*, 1995, 20.3: 273-297.

16].PRESS, William H., et al. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

17].BEN-HUR, Asa, et al. Support vector clustering. *Journal of machine learning research*, 2001, 2.Dec: 125-137.

18].BOUBOULIS, Pantelis, et al. Complex support vector machines for regression and quaternary classification. *IEEE transactions on neural networks and learning systems*, 2015, 26.6: 1260-1274.

19].http://cs229.stanford.edu/notes/cs229-notes3.pdf

20].SMOLA, Alex J.; SCHÖLKOPF, Bernhard. A tutorial on support vector regression. *Statistics and computing*, 2004, 14.3: 199-222.

21].CSÁJI, Balázs Csanád. Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, 2001, 24: 48.

22].SHARMA, Vidushi; RAI, Sachin; DEV, Anurag. A comprehensive study of artificial neural networks. *International Journal of Advanced research in computer science and software engineering*, 2012, 2.10.

23].FAUSETT, Laurene V., et al. *Fundamentals of neural networks: architectures, algorithms, and applications*. Englewood Cliffs: Prentice-Hall, 1994.

24].LIOU, Daw-Ran; LIOU, Jiun-Wei; LIOU, Cheng-Yuan. *Learning Behaviors of Perceptron*. ISBN 978-1-477554-73-9. iConcept Press, 2013.

25].ROSENBLATT, Frank. *Principles of neurodynamics. perceptrons and the theory of brain mechanisms*. CORNELL AERONAUTICAL LAB INC BUFFALO NY, 1961.

26].ΒΛΑΧΆΒΑΣ, Ι.; ΚΕΦΑΛΑΣ, Π.; ΒΑΣΙΛΕΙΆΔΗΣ, Ν. Φ. Κόκκορας και Η. Σακελλαρίου," Τεχνητή Νοημοσύνη"-Γ'Έκδοση 2006, εκδ. *Μ. Γκιούρδας.[2] Κεραυνού. Ε Τεχνητή Νοημοσύνη και Έμπειρα Συστήματα. Ελληνικό Ανοιχτό Πανεπιστήμιο. Πάτρα.*

27].ΔΙΑΜΑΝΤΆΡΑΣ, Κωνσταντίνος. Τεχνητά νευρωνικά δίκτυα. *Κλειδάριθμος. Αθήνα*, 2007.

28].RUSSEL, Stuart; NORVIG, Peter. Τεχνητή νοημοσύνη, μια σύγχρονη προσέγγιση. *Β'έκδοση*, 2005, 31-69.

29].BUITINCK, Lars, et al. API design for machine learning software: experiences from the scikit-learn project. *arXiv preprint arXiv:1309.0238*, 2013.

30].PEDREGOSA, Fabian, et al. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 2011, 12.Oct: 2825-2830.

31].GOLDBERG, David E. *Genetic algorithms*. Pearson Education India, 2006.

32].MITCHELL, Melanie. *An introduction to genetic algorithms*. MIT press, 1998.

33].SNOEK, Jasper; LAROCHELLE, Hugo; ADAMS, Ryan P. Practical bayesian

optimization of machine learning algorithms. In: *Advances in neural information processing systems*. 2012. p. 2951-2959.

34]. NILSSON, Nils J. Introduction to machine learning: An early draft of a proposed textbook. 1996.

35]. SMOLA, Alex; VISHWANATHAN, S. V. N. Introduction to machine learning. *Cambridge University, UK*, 2008, 32: 34.

36]. MANNILA, Heikki. Data mining: machine learning, statistics, and databases. In: *ssdbm*. IEEE, 1996. p. 2.

37]. MAN, Kim-Fung; TANG, Kit-Sang; KWONG, Sam. Genetic algorithms: concepts and applications [in engineering design]. *IEEE transactions on Industrial Electronics*, 1996, 43.5: 519-534.

38]. KOUSIOURIS, George, et al. Translation of application-level terms to resource-level attributes across the cloud stack layers. In: *Computers and Communications (ISCC), 2011 IEEE Symposium on*. IEEE, 2011. p. 153-160.

39]. KOUSIOURIS, George; CUCINOTTA, Tommaso; VARVARIGOU, Theodora. The effects of scheduling, workload type and consolidation scenarios on virtual machine performance and their prediction through optimized artificial neural networks. *Journal of Systems and Software*, 2011, 84.8: 1270-1291.

40]. MCKINNEY, Wes. pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, 2011, 1-9.

41]. TEAM, Python Core. Python: A dynamic, open source programming language. *Wilmington, DE: Python Software Foundation*, 2015.

42]. EATON, John Wesley; BATEMAN, David; HAUBERG, Søren. *GNU Octave version 3.0. 1 manual: a high-level interactive language for numerical computations*. SoHo Books, 2007.

Παράρτημα 1 : Πηγαίος Κώδικας

Στο παράρτημα αυτό βρίσκεται ο κώδικας που χρησιμοποιείται για την εκπαίδευση και ρύθμιση των αλγόριθμων : Τυχαία Δάση και Μηχανές Διανυσμάτων Υποστήριξης.

Πιο συγκεκριμένα παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων αλλά και για την εκτέλεση της διαδικασίας k-Fold Cross Validation με χρήση της αναζήτησης πλέγματος. Ο συνολικός πηγαίος κώδικας (που χρησιμοποιήθηκε για την αυτοματοποίηση της εκτέλεσης των πειραμάτων) αλλά και το σύνολο δεδομένων που χρησιμοποιήθηκε για τα πειράματα βρίσκονται στη σελίδα: https://gitlab.com/Mundusspawn/thesis_implementation.git

Πηγαίος Κώδικας Συναρτήσεων Εκπαίδευσης και Ρύθμισης Μηχανών Διανυσμάτων Υποστήριξης

```
def gridSVM(inputs, output, param_grid):

    X_train, X_test, y_train, y_test = train_test_split(inputs,
    output, test_size=0.20, random_state=43)

    grid_search = GridSearchCV(SVR(kernel='rbf'), param_grid, cv=5)

    grid_search.fit(X_train, y_train)

    return grid_search.best_params_




def checkSVM(c, g, inputs, output):

    X_train, X_test, y_train, y_test = train_test_split(inputs,
    output, test_size=0.35, random_state=42)

    X_test, X_val, y_test, y_val = train_test_split(X_test, y_test,
    test_size=0.4285, random_state=42)

    svr_rbf = SVR(kernel='rbf', C=c, gamma=c)

    y_rbf = svr_rbf.fit(X_train, y_train).predict(X_test)

    y_test = y_test.iloc

    return ce.mape(y_test, y_rbf)


def graph3d(inputs, y):

    fig = plt.figure()

    ax = fig.add_subplot(111, projection='3d')

    ax.scatter(inputs.iloc[:, 0], inputs.iloc[:, 1], output)

    ax.plot(inputs.iloc[:, 0], inputs.iloc[:, 1], y, color='red')

    plt.show()

    return
```

Πηγαίος Κώδικας Συναρτήσεων Εκπαίδευσης και Ρύθμισης Τυχαίων Δασών

```
def gridforest(inputs, output, param_grid):

    X_train, X_test, y_train, y_test = train_test_split(inputs,
    output, test_size=0.20, random_state=42)

    rgr = RandomForestRegressor()

    grid_clf = GridSearchCV(rgr, param_grid, cv=5)

    grid_clf.fit(X_train, y_train)

    return grid_clf.best_params_


def checkforest(estimators, samples, depth, inputs, output):

    X_train, X_test, y_train, y_test = train_test_split(inputs,
    output, test_size=0.35, random_state=42)

    X_test, X_val, y_test, y_val = train_test_split(X_test, y_test,
    test_size=0.4285, random_state=42)

    rgr = RandomForestRegressor(n_estimators=estimators,
    min_samples_leaf=samples, max_depth=depth)

    rgr.fit(X_train, y_train)

    prediction = rgr.predict(X_test)

    fig = plt.figure()

    y_test = y_test.iloc

    return ce.mape(y_test, prediction)
```

Σχεδίαση Πλέγματος Αναζήτησης και Διάβασμα Συνόλου Δεδομένων.

```
path = input('Enter Dataset Path')

inputs = pd.read_csv(path,header=None)

output = inputs.iloc[:,5]

inputs = inputs.iloc[:,:5]


param_grid = {
                    'n_estimators':   [5,10]   +   [x   for   x   in
range(100,1000,100)],
                     'max_depth':      sum([[j      for      j      in
range(1,100,10)],[None]],[]),
                    'min_samples_leaf':[i for i in range(1,50,5)]

                }

params=gridforest(inputs,output,param_grid)


checkforest(params.get("n_estimators","none"),params.get("min_samples
_leaf","none"),params.get("max_depth","none"),inputs,output)



Cs = 10. ** np.arange(-3, 8)

gammas= 10. ** np.arange(-5, 4)

param_grid = {'C': Cs, 'gamma': gammas}

inputs = pd.read_csv(path, header=None)

output = inputs.iloc[:, 5]

inputs = inputs.iloc[:, :5]

min_max_scaler = MinMaxScaler()

inputs = min_max_scaler.fit_transform(inputs)

params = gridSVM(inputs, output, param_grid)

checkSVM(params.get("C","none"),params.get("gamma","none"),inputs,out
put)
```

Συνάρτηση Υπολογισμού Σφάλματος Mape

```
def mape(r, p):
    sum = 0
    k = 0

    for i in p:
        sum = sum + abs((r[k] - p[k]) / r[k])
        k = k + 1

    return 100 * sum / len(p)
```